



Universidad de las Ciencias Informáticas
FACULTAD 4

Trabajo de diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Título:

Componente de obtención de datos estadísticos de bateadores a la
ofensiva para Strike Zone.

Autor: Julio Cesar Betancourt Ramos

Tutor/a: Ing. Adis Mirtha Canet Aliaga

La Habana, 2022

“Año 64 de la Revolución”

“Es difícil vencer a una persona que nunca se rinde.”

Babe Ruth.

Dedicatoria

Dedico este trabajo principalmente a mi padre Julio Cesar Betancourt Calderín que siempre me acompaña desde el cielo, a mi madre y a mi hermana que me han apoyado durante todo este proceso y me han aguantado mis malos momentos.

Agradecimientos

Agradecerles principalmente a mis amigos de la universidad: Samira, Dayana, Jesús, Daniel, Carlos, Clarke, Jaydy y a los que ya no están Marlon, Dainier, Abraham, Yanela y a mi tutora Adis y a Álvaro por aconsejarme siempre y por estar presentes siempre que lo necesité. Al Joe, el entrenador de fitness, que estuvo pendiente a mi docencia durante mis años de universidad. A Andy por ayudarme con la aplicación cuando me hizo falta. A mi profesora de inglés Silvia por todos los consejos que siempre me dio y por las tardes con música y alcohol que me sirvieron para desahogarme siempre que estuve hasta el cuello de todo y agradecerles a todas esas personas que de una forma u otra aportaron su granito de arena para que hoy estuviera donde estoy. Muchas Gracias a todos de corazón.

Declaración de autoría

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales sobre esta, con carácter exclusivo.

Para que así conste firmamos la presente a los días 7 del mes diciembre del año 2022.



Julio Cesar Betancourt Ramos

Autor

Adis Mirtha
Canet
Aliaga

Digitally signed by Adis Mirtha Canet Aliaga
DN:
c:CO, o:U. 2342, ou:100.1.1=adis
canet, cn=Adis Mirtha Canet Aliaga,
SERIALNUMBER=E137171, G=Adis
Mirtha, SN=Canet Aliaga, C=CU
Reason: I am approving this document
with my legally binding signature
Location: UCI
Date: 2022.12.05 09:12:10-05'00'
Post PDF Reader Version: 11.2.1

Ing. Adis Mirtha Canet Aliaga

Tutor

Resumen

Desde el comienzo del desarrollo de las tecnologías relacionadas con la informática la industria del deporte ha buscado estrategias para llegar a los televidentes y garantizar que toda persona que disfruta de una transmisión deportiva reciba más información de lo que sucede dentro del juego y también han buscado la manera de apoyar al arbitraje. Con este objetivo se han creado diferentes sistemas informáticos ajustados a las necesidades y características de cada deporte.

Dentro de este contexto Strike Zone es un proyecto perteneciente al Centro de Entornos Interactivos VERTEX de la Universidad de las Ciencias Informáticas el cual fue creado para el enriquecimiento de los videos de las trasmisiones televisivas del Béisbol con el objetivo principal de determinar si un lanzamiento es bola o *strike*. Actualmente, Strike Zone cuenta con un módulo que permite construir la zona de *strike* para cada bateador de forma manual. La propuesta de esta investigación es un componente de obtención de datos estadísticos de bateadores a la ofensiva para Strike Zone. Para la realización de la propuesta de solución se hace uso del lenguaje C++, del entorno de desarrollo integrado Qt Creator 5.2.1. El componente propuesto se basa en el análisis que se les realiza a los bateadores para analizar su rendimiento y saber si el bateador rinde o no en el equipo y a su vez si el mismo bateador está avanzado con respecto al equipo.

Palabras clave: componente, estadística, béisbol, zona de *strike*.

Tabla de contenido

INTRODUCCIÓN	17
CAPÍTULO 1: Fundamentación teórica del Componente de obtención de datos estadísticos para Strike Zone	21
1.1 Conceptos asociados al dominio de la investigación	21
1.1.1 Béisbol.....	21
1.1.2 Sabermetría	22
1.1.2 Estadística	23
1.1.3 Sistemas informáticos estadísticos.	24
1.2 Estudio de sistemas homólogos.....	24
1.3 Metodologías, Herramientas y Tecnologías	28
1.3.1 Metodología del desarrollo de software.....	29
1.3.2 Entorno de desarrollo integrado	30
1.3.3 Lenguajes de programación.....	30
1.3.4 Sistema gestor de base de datos	30
1.4 Conclusiones parciales	30
CAPÍTULO 2: Descripción de la propuesta de solución para el componente de obtención de datos estadísticos para Strike Zone.....	32
2.1 Descripción de la propuesta de solución.....	32
2.2 Especificación de requisitos de software.....	33
2.2.1 Requisitos funcionales	33
2.2.2 Requisitos no funcionales	35
2.3 Descripción de las historias de usuario	36
2.4 Fase de Planificación	52
2.4.1 Plan de iteraciones.....	52
2.4.2 Plan de entrega.....	56
2.4.3 Tarjetas CRC	57

2.5 Estilo arquitectónico.....	59
2.5.1 Patrón arquitectónico.....	60
2.6 Diagrama de clase del diseño.....	60
2.7 Modelo de datos.....	61
2.8 Patrones de diseño.....	63
2.9 Conclusiones parciales.....	64
Capítulo 3: Implementación y pruebas del componente de obtención de datos estadísticos para Strike Zone.....	65
3.1 Estándares de codificación.....	65
3.2 Tareas de Ingeniería.....	67
3.2.1 Descripción Tareas de Ingeniería.....	71
3.3 Pruebas de software.....	72
3.3.1 Pruebas Unitarias.....	72
3.3.2 Pruebas de aceptación.....	77
3.4 Resultados de las pruebas aplicadas.....	¡Error! Marcador no definido.
3.4 Conclusiones parciales.....	93
Conclusiones Generales.....	94
Recomendaciones.....	95
Bibliografía.....	96
Anexos.....	100
A.1 Estadísticos saber métricos para el béisbol.....	100
A.1.1 Resumen de los principales estadísticos saber métricos.....	100
A.1.2 Estadísticos saber métricos de bateo.....	101
A.2 Historias de usuario.....	105
A.3 Tareas de Ingeniería.....	159
A.4 Casos de Prueba de Aceptación.....	182

Índice de tablas

Tabla 1: Requisitos funcionales del sistema.....	33
Tabla 2: Historia de usuario 1 "Registrar deportista"	36
Tabla 3: Historia de usuario 2 "Modificar deportista"	37
Tabla 4: Historia de usuario 3 "Eliminar deportista".....	39
Tabla 5: Historia de usuario 4 "Listar deportista".....	40
Tabla 6: Historia de usuario 5 "Visualizar información OBP"	42
Tabla 7: Historia de usuario 6 "Ingresar información OBP"	42
Tabla 8: Historia de usuario 7 "Calcular datos OBP"	43
Tabla 9: Historia de usuario 8 "Guardar información OBP"	45
Tabla 10: Historia de usuario 9 "Mostrar información OBP"	47
Tabla 11: Historia de usuario 10 "Limpiar información OBP".....	48
Tabla 12: Historia de usuario 11 "Visualizar resultado OBP".....	49
Tabla 13: Historia de usuario 12 "Visualizar comentario OBP"	50
Tabla 14: Historia de usuario 13 "Eliminar OBP "	51
Tabla 15: Plan de iteraciones.....	53
Tabla 16: Plan de entrega.....	56
Tabla 17: Tarjeta CRC "Registrar Deportista"	57
Tabla 18: Tarjeta CRC "Modificar deportista"	57
Tabla 19: Tarjeta CRC "Main Window"	58
Tabla 20: Estándares de codificación.....	65
Tabla 21: Tareas de Ingeniería	67
Tabla 22: Tarea de ingeniería 1 "Implementar Registrar deportista"	71
Tabla 23: Tarea de ingeniería 2 "Implementar Modificar deportista"	71
Tabla 24: Tarea de ingeniería 3 "Implementar Eliminar deportista"	71
Tabla 25: Tarea de ingeniería 4: "Implementar Listar deportista"	72
Tabla 26: Cálculo de la complejidad ciclomática del método on_pushButton_clicked de la clase mainwindow.cpp	74
Tabla 27: Caminos del grafo de flujo.....	76
Tabla 28: Caso de prueba para el camino básico 5	76
Tabla 29: Caso de prueba 1 "Gestionar deportista"	78
Tabla 30: Caso de prueba 2: "Registrar deportista"	78
Tabla 31: Caso de prueba 3 "Modificar deportista"	79

Tabla 32: Caso de prueba 4 "Listar deportista"	79
Tabla 33: Caso de prueba 5 "Guardar deportista"	80
Tabla 34: Caso de prueba 6 "Eliminar deportista"	81
Tabla 35: Caso de prueba 6 "Administrar OBP"	81
Tabla 36: Caso de prueba 7 "Visualizar OBP"	82
Tabla 37: Caso de prueba 8 "Ingresar información OBP"	82
Tabla 38: Caso de prueba 9 "Calcular datos OBP"	83
Tabla 39: Caso de prueba 10 "Guardar información"	83
Tabla 40: Caso de prueba 11 "Mostrar información OBP"	84
Tabla 41: Caso de prueba 12 "Limpiar campos OBP"	84
Tabla 42: Caso de prueba 13 "Visualizar resultado OBP"	85
Tabla 43: Caso de prueba 14 "Visualizar comentario OBP"	85
Tabla 44: Caso de prueba 15 "Eliminar OBP"	86
Tabla 45: Caso de prueba 15 "Administrar SLG"	87
Tabla 46: Caso de prueba 16 "Visualizar SLG"	87
Tabla 47: Caso de prueba 17 "Ingresar información SLG"	88
Tabla 48: Caso de prueba 18 "Calcular datos SLG"	88
Tabla 49: Caso de prueba 19 "Guardar información SLG"	89
Tabla 50: Caso de prueba 20 "Mostrar información SLG"	89
Tabla 51: Caso de prueba 21 "Limpiar casos SLG"	90
Tabla 52: Caso de prueba 22 "Visualizar resultado SLG"	90
Tabla 53: Caso de prueba 23 "Visualizar comentario SLG"	91
Tabla 54: Caso de prueba 24 "Eliminar SLG"	91
Tabla 55: Estadísticos sabermétricos de bateo	101
Tabla 56: Historia de usuario 14 " Visualizar información SLG"	105
Tabla 57: Historia de usuario 15 "Ingresar información SLG"	106
Tabla 58: Historia de usuario 16 "Calcular datos SLG "	107
Tabla 59: Historia de usuario 17 "Guardar información SLG"	108
Tabla 60: Historia de usuario 18 "Mostrar información SLG "	109
Tabla 61: Historia de usuario 19 "Limpiar información SLG"	110
Tabla 62: Historia de usuario 20 "Visualizar resultado SLG"	111
Tabla 63: Historia de usuario 21 "Visualizar comentario SLG "	112
Tabla 64: Historia de usuario 22 "Eliminar SLG"	113

Tabla 65: Historia de usuario 23 " Visualizar información OPS "	114
Tabla 66: Historia de usuario 24 "Ingresar información OPS"	115
Tabla 67: Historia de usuario 25 "Calcular datos OPS"	116
Tabla 68: Historia de usuario 26 "Guardar información OPS"	117
Tabla 69: Historia de usuario 27 "Mostrar información OPS"	118
Tabla 70: Historia de usuario 28 "Limpiar información OPS "	119
Tabla 71: historia de usuario 29 "Visualizar resultado OPS "	120
Tabla 72: Historia de usuario 30 "Visualizar comentario OPS"	121
Tabla 73: Historia de usuario 31 "Eliminar OPS"	122
Tabla 74: Historia de usuario 5 " Visualizar información BABIP "	123
Tabla 75: Historia de usuario 33 "Ingresar información BABIP"	124
Tabla 76: Historia de usuario 34 "Calcular datos BABIP "	125
Tabla 77: Historia de usuario 35 "Guardar información BABIP"	126
Tabla 78: Historia de usuario 36 "Mostrar información BABIP"	127
Tabla 79: Historia de usuario 37 "Limpiar información BABIP"	128
Tabla 80: Historia de usuario 38 "Visualizar resultado BABIP"	129
Tabla 81: Historia usuario 39 "Visualizar comentario BABIP"	130
Tabla 82: Historia usuario 40 "Eliminar BABIP"	131
Tabla 83: Historia de usuario 6 " Visualizar información BB%"	132
Tabla 84: Historia de usuario 42 "Ingresar información BB% "	133
Tabla 85: Historia de usuario 43 "Calcular datos BB% "	134
Tabla 86: Historia de usuario 44 "Guardar información BB%"	135
Tabla 87: Historia de usuario 45 "Mostrar información BB% "	136
Tabla 88: Historia de usuario 46 "Limpiar información BB% "	137
Tabla 89: Historia de usuario 47 "Visualizar resultado BB% "	138
Tabla 90: Historia de usuario 48 "Visualizar comentario BB%"	139
Tabla 91: Historia de usuario 49 "Eliminar BB%"	140
Tabla 92: Historia de usuario 7 " Visualizar información ISO "	141
Tabla 93: Historia de usuario 51 "Ingresar información ISO"	142
Tabla 94: Historia de usuario 52 "Calcular datos ISO"	143
Tabla 95: Historia de usuario 53 "Guardar información ISO"	144
Tabla 96: Historia de usuario 55 "Limpiar información ISO"	146
Tabla 97: Historia de usuario 56 "Visualizar resultado ISO"	147

Tabla 98: Historia de usuario 57 "Visualizar comentario ISO"	148
Tabla 99: Historia de usuario 8 "Administrar RC"	150
Tabla 100: Historia de usuario 60 "Ingresar información RC "	151
Tabla 101: Historia de usuario 61 "Calcular datos RC"	152
Tabla 102: Historia de usuario 62 "Guardar información RC"	153
Tabla 103: historia de usuario 63 "Mostrar información RC"	154
Tabla 104: Historia de usuario 64 "Limpiar información RC "	155
Tabla 105: Historia de usuario 65 "Visualizar resultado RC"	156
Tabla 106: Historia de usuario 66 "Visualizar comentario RC"	157
Tabla 107: Historia de usuario 67 "Eliminar RC"	158
Tabla 108: Tarea de ingeniería 5 " Implementar Visualizar información OBP "	159
Tabla 109: tarea de ingeniería 6 "Implementar Ingresar información OBP"	159
Tabla 110: Tarea de ingeniería 7 " Implementar Calcular datos OBP "	160
Tabla 111: Tarea de ingeniería 8 "Implementar Guardar información OBP"	160
Tabla 112: Tarea de ingeniería 9 "Implementar Mostrar información OBP"	160
Tabla 113: Tarea de ingeniería 10 "Implementar Limpiar campos OBP"	161
Tabla 114: Tarea de ingeniería 11 "Implementar Visualizar resultado OBP"	161
Tabla 115: tarea de ingeniería 12 "Implementar Visualizar comentario OBP"	162
Tabla 116: Tarea de ingeniería 13 "Implementar Eliminar OBP"	162
Tabla 117: Tarea de ingeniería 14 "Implementar Visualizar información SLG"	162
Tabla 118: Tarea de ingeniería 15 "Implementar Ingresar información SLG"	163
Tabla 119: Tarea de ingeniería 16 "Implementar Calcular datos SLG"	163
Tabla 120: Tarea de ingeniería 17 "Implementar Guardar información SLG"	163
Tabla 121: Tarea de ingeniería 18 "Implementar Mostrar información SLG"	164
Tabla 122: Tarea de ingeniería 19 "Implementar Limpiar campos SLG"	164
Tabla 123: Tarea de ingeniería 20 "Implementar Visualizar resultado SLG"	164
Tabla 124: Tarea de ingeniería 21 "Implementar Visualizar comentario SLG"	165
Tabla 125: Tarea de ingeniería 22 "Implementar Eliminar SLG"	165
Tabla 126: Tarea de ingeniería 23 "Implementar Visualizar información OPS"	165
Tabla 127: Tarea de ingeniería 23 "Implementar Ingresar información OPS"	166
Tabla 128: Tarea de ingeniería 24 "Implementar Calcular datos OPS"	166
Tabla 129: tarea de ingeniería 25 "Implementar Guardar información OPS"	167
Tabla 130: Tarea de ingeniería 26 "Implementar Mostrar información OPS"	167

Tabla 131: Tarea de ingeniería 27 "Implementar Limpiar campos OPS"	167
Tabla 132: Tarea de ingeniería 28 "Implementar Visualizar resultado OPS"	168
Tabla 133: Tarea de ingeniería 29 "Implementar Visualizar comentario OPS"	168
Tabla 134: Tarea de ingeniería 30 "Implementar Eliminar OPS"	168
Tabla 135: tarea de ingeniería 31 " Implementar Visualizar información BABIP"	169
Tabla 136: Tarea de ingeniería 32 "Implementar Ingresar información BABIP"	169
Tabla 137: tarea de ingeniería 33 "Implementar Calcular datos BABIP"	169
Tabla 138: Tarea de ingeniería 34 "Implementar Guardar información BABIP"	170
Tabla 139: Tarea de ingeniería 35 "Implementar Mostrar información BABIP"	170
Tabla 140: Tarea de ingeniería 36 "Implementar Limpiar campos BABIP"	171
Tabla 141: Tarea de ingeniería 37 "Implementar Visualizar resultado BABIP"	171
Tabla 142: Tarea de ingeniería 38 " Implementar Visualizar comentario BABIP"	171
Tabla 143: Tarea de ingeniería 40 "Implementar Eliminar BABIP"	172
Tabla 144: Tarea de ingeniería 41 "Implementar Visualizar información BB%"	172
Tabla 145: Tarea de ingeniería 42 "Implementar Ingresar información BB%"	172
Tabla 146: Tarea de ingeniería 43 "Implementar Calcular datos BB%"	173
Tabla 147: Tarea de ingeniería 44 "Implementar Guardar información BB%"	173
Tabla 148: Tarea de ingeniería 45 "Implementar Mostrar información BB%"	173
Tabla 149: Tarea de ingeniería 46 "Implementar Limpiar campos BB%"	174
Tabla 150: Tarea de ingeniería 47 "Implementar Visualizar resultado BB%"	174
Tabla 151: Tarea de ingeniería 48 "Implementar Visualizar comentario BB%"	175
Tabla 152: Tarea de ingeniería 49 "Implementar Eliminar BABIP"	175
Tabla 153: Tarea de ingeniería 49 "Implementar Visualizar información ISO"	175
Tabla 154: Tarea de ingeniería 51 "Implementar Ingresar información ISO"	176
Tabla 155: Tarea de ingeniería 52 "Implementar Calcular datos ISO"	176
Tabla 156: Tarea de ingeniería 48 "Implementar Guardar información ISO"	176
Tabla 157: Tarea de ingeniería 54 "Implementar Mostrar información ISO"	177
Tabla 158: Tarea de ingeniería 55 "Implementar Limpiar campos ISO"	177
Tabla 159: Tarea de ingeniería 56 "Implementar Visualizar resultado ISO"	177
Tabla 160: Tarea de ingeniería 57 "Implementar Visualizar comentario ISO"	178
Tabla 161: Tarea de ingeniería 58 "Implementar Eliminar ISO"	178
Tabla 162: Tarea de ingeniería 59 "Implementar Visualizar información RC"	179
Tabla 163: Tarea de ingeniería 60 "Implementar Ingresar información RC"	179

Tabla 164: Tarea de ingeniería 61 "Implementar Calcular datos RC"	179
Tabla 165: Tarea de ingeniería 62 "Implementar Guardar información RC"	180
Tabla 166: Tarea de ingeniería 63 "Implementar Guardar información RC"	180
Tabla 167: Tarea de ingeniería 64 "Implementar Limpiar campos RC"	180
Tabla 168: Tarea de ingeniería 65 "Implementar Visualizar resultado RC"	181
Tabla 169: Tarea de ingeniería 66 "Implementar Visualizar comentario OPS"	181
Tabla 170: Tarea de ingeniería 67 "Implementar Eliminar RC"	181
Tabla 171: Caso de prueba 24 "Administrar OPS"	182
Tabla 172: Caso de prueba 25 "Visualizar OPS"	183
Tabla 173: Caso de prueba 26 "Ingresar Información OPS"	183
Tabla 174: caso de prueba 27 "Ingresar Información OPS"	183
Tabla 175: Caso de prueba 28 "Guardar información OPS"	184
Tabla 176: Caso de prueba 29 "Mostrar información OPS"	185
Tabla 177: Caso de prueba 30 "Limpiar campos OPS"	185
Tabla 178: Caso de prueba 31 "Visualizar resultado OPS"	186
Tabla 179: Caso de prueba 32 "Visualizar comentario OPS"	186
Tabla 180: Caso de prueba 33 "Eliminar OPS"	187
Tabla 181: Caso de prueba 34 "Visualizar BABIP"	187
Tabla 182: Caso de prueba 35 "Ingresar Información BABIP"	188
Tabla 183: Caso de prueba 36 "Calcular datos BABIP"	188
Tabla 184: Caso de prueba 37 "Guardar información BABIP"	189
Tabla 185: Caso de prueba 38 "Mostrar información BABIP"	189
Tabla 186: Caso de prueba 39 "Limpiar campos BABIP"	190
Tabla 187: Caso de prueba 40 "Visualizar resultado BABIP"	190
Tabla 188: Caso de prueba 41 "Visualizar comentario BABIP"	190
Tabla 189: Caso de prueba 42 "Eliminar BABIP"	191
Tabla 190: Caso de prueba 43 "Administrar BB%"	192
Tabla 191: Caso de prueba 44 "Visualizar BB%"	192
Tabla 192: Caso de prueba 45 "Ingresar Información BB%"	193
Tabla 193: Caso de prueba 46 "Calcular datos BB%"	193
Tabla 194: caso de prueba 47 "Guardar información BB%"	194
Tabla 195: caso de prueba 48 "Mostrar información BB%"	194
Tabla 196: Caso de prueba 49 "Limpiar campos BB%"	195

Tabla 197: Caso de prueba 50 "Visualizar resultado BB%"	195
Tabla 198: Caso de prueba 51 "Visualizar comentario BB%"	196
Tabla 199: Caso de prueba 52 "Eliminar BB%"	196
Tabla 200: Caso de prueba 53 "Administrar ISO"	197
Tabla 201: Caso de prueba 54 "Visualizar ISO"	197
Tabla 202: Caso de prueba 55 "Ingresar Información ISO"	198
Tabla 203: Caso de prueba 56 "Calcular datos ISO"	198
Tabla 204: Caso de prueba 57 "Guardar información ISO"	199
Tabla 205: Caso de prueba 58 "Mostrar información ISO"	199
Tabla 206: Caso de prueba 59 "Limpiar campos ISO"	200
Tabla 207: Caso de prueba 60 "Visualizar resultado ISO"	200
Tabla 208: Caso de prueba 61 "Visualizar comentario ISO"	201
Tabla 209: Caso de prueba 62 "Eliminar ISO"	202
Tabla 210: Caso de prueba 63 "Administrar RC"	202
Tabla 211: Caso de prueba 64 "Visualizar RC"	203
Tabla 212: Caso de prueba 65 "Ingresar Información RC"	203
Tabla 213: Caso de prueba 66 "Calcular datos RC"	204
Tabla 214: Caso de prueba 67 "Guardar información RC"	204
Tabla 215: Caso de prueba 68 "Mostrar información RC"	205
Tabla 216: Caso de prueba 69 "Limpiar campos RC"	205
Tabla 217: Caso de prueba 70 "Visualizar resultado RC"	206
Tabla 218: Caso de prueba 71 "Visualizar comentario RC"	206
Tabla 219: Caso de prueba 72 "Eliminar RC"	207

Índice de ilustraciones

Ilustración 1: Sistema PITCHf/x	25
Ilustración 2: Herramienta Rapsodo	26
Ilustración 3: Sistema Statcast	27
Ilustración 4: Sistema K-Zone	28
Ilustración 5: “Arquitectura en Capas”	60
Ilustración 6: Diagrama de clases del diseño	61
Ilustración 7: “Modelado de datos”	62
Ilustración 8: Resumen de los principales estadísticos sabermétricos	100

INTRODUCCIÓN

El aprendizaje y desarrollo del análisis deportivo ha sido uno de los factores fundamentales para la investigación novedosa en la industria del deporte. La predicción de futuros movimientos ha acaparado la atención en el contexto deportivo debido a los intereses del mercado y que en las transmisiones deportivas se reciban la mayor cantidad de información de lo que acontece respetando y apoyando las decisiones arbitrarias; dichas predicciones se ven reflejadas con mayor énfasis en los deportes colectivos.

Los deportes colectivos son aquellos que se practican entre equipos opuestos, donde los jugadores interactúan entre ellos de manera simultánea para alcanzar un objetivo, existe cooperación entre dos o más compañeros y oposición a los deportistas contrarios, realizando una participación simultánea o alternativa de los jugadores, pudiendo compartir un espacio común y utilizar un objeto móvil. Año tras año, el deporte en equipo va ganando popularidad, convirtiéndose incluso en un fenómeno seguido por las masas y en referentes e ídolos para muchas personas (Bou, 2019). Ejemplo de deportes colectivos son el fútbol que es el deporte colectivo más famoso del mundo y el más practicado. Este deporte se practica en un terreno rectangular de césped natural o artificial y en el que en cada extremo hay una portería; el objetivo es desplazar la pelota a través del campo con cualquier parte del cuerpo, mayoritariamente con los pies, que no sean los brazos o las manos e introducirla en la portería contraria, el baloncesto se juega entre dos equipos de 5 jugadores cada uno. El partido se divide en cuatro períodos de 10 o 12 minutos cada uno. El objetivo de este juego es anotar el máximo de puntos posibles introduciendo un balón por la canasta, un aro que se encuentra a 3,05 metros de altura del que cuelga una red, el voleibol es un deporte colectivo que se juega con pelota. Dos equipos, formados por 6 jugadores cada uno, se enfrentan en una cancha de juego que se encuentra dividida en la mitad por una red. El objetivo de este juego es pasar el balón por encima de esta, intentando que la pelota llegue al suelo del equipo contrario. Mientras, el otro equipo intentará evitarlo con su defensa, procurando devolver la pelota, béisbol entre otros (Bou, 2019).

El béisbol es un deporte colectivo, de equipo, de carácter variable y contextualizado entre el grupo de “deportes con pelota”, se clasifica como un deporte con un fin táctico. El aumento del número de países que lo practican, el incremento de complejas y dinámicas formas de jugar, las modificaciones de la reglamentación y la participación de profesionales junto a amateurs en las competencias fundamentales implican transformaciones. Este puede desarrollarse solo con desplazamientos, lanzamientos, alcances, bateo y las combinaciones de ellos: atrapar-lanzar, correr-lanzar, correr-saltar, batear-correr, correr-atrapar. Independientemente de que la acción que más frecuentemente se realiza es el correr,

la más importante es el bateo, pues constituye la primera acción del equipo que está a la ofensiva y da la posibilidad de comenzar ganando. Debido a la gran cantidad de datos y de información que se puede recopilar solamente de este deporte se decidió crear la sabermetría como ciencia que estudia las actividades que se suscitan dentro del campo de juego ya que el béisbol es reconocido por ser el deporte que más datos estadísticos genera durante un partido.

En Cuba el béisbol no es solo un deporte, es la prolongación cultural de todo un pueblo. Se caracteriza por el empleo de la fuerza a la ofensiva y el poco uso de la rapidez en combinaciones tácticas entre el bateador y los corredores, jugadas ofensivas y defensivas sustentadas en esquemas tácticos tradicionales, entre otras características. El desarrollo de las tecnologías asociadas a la mejora del rendimiento deportivo de los atletas en béisbol a nivel mundial, entre otros aspectos, ha hecho que Cuba pierda los planos estelares con que contaba históricamente. (Fuentes, Alfredo Ríos; Sánchez Córdova, Beatriz; Rojas Duque de Estrada, Enrique, 2019). En la actualidad, con el objetivo de apoyar el proceso de perfeccionamiento del béisbol como deporte, se avanza con pequeños pasos en el desarrollo científico-tecnológico en el área de la sabermetría.

La Universidad de las Ciencias Informáticas (UCI), es un centro de estudios que tiene como una de sus misiones fundamentales: producir aplicaciones y servicios informáticos que sirvan de soporte a la industria cubana del software y la informatización del país. Para esto cuenta con una red de centros que se encargan de la producción de software, entre estos se encuentra el Centro de Tecnologías Interactivas (VERTEX).

El proyecto **Strike Zone**, perteneciente a VERTEX, ha logrado avances significativos en el béisbol como deporte. Ejemplo de ello se puede encontrar en el reconocimiento de la trayectoria que sigue la bola lanzada por el pitcher hacia el *home*, a partir de imágenes digitales o videos, y cuenta con un método para analizar el tipo de lanzamiento que se realizó, así como la velocidad del mismo. Este sistema está basado en la primera herramienta que la Major League of Baseball (MLB) usó con este objetivo, pero no cuenta con un componente estadístico que tenga implementado las funcionalidades necesarias para analizar a los jugadores de manera individual, dígase, análisis del rendimiento de los jugadores, características de bateo y/o picheo, cantidad de movimientos que realiza el jugador y las condiciones en que lo realiza, entre otros, por lo que para una mayor explotación de la aplicación se hace necesario agregarle dicho componente.

Teniendo en cuenta la situación problemática antes descrita, se identifica como **problema de investigación**: ¿Cómo automatizar el proceso de análisis estadístico de los jugadores de béisbol?

Como **objeto de estudio** se tiene el análisis de datos estadísticos en el béisbol. Dentro del objeto de estudio de la investigación se precisa como **campo de acción** componente de almacenamiento de

datos para Strike Zone. Para dar solución al problema de investigación planteado, se define como **objetivo general:** Desarrollar un componente de obtención de datos estadísticos para Strike Zone que se pueda introducir nuevos elementos e información referente al jugador y manejar algunas variables para el entendimiento del mismo.

Para dar cumplimiento al objetivo de la investigación se desglosan las siguientes **tareas de investigación:**

- Elaboración del marco teórico de la investigación a través del estudio del estado del arte.
- Selección de metodología, herramientas de software a utilizar y lenguajes de programación para el desarrollo del sistema
- Caracterización de diferentes sistemas de captura de datos estadísticos de jugadores de béisbol.
- Identificación de los requisitos de software para determinar las funcionalidades y restricciones que el sistema debe tener.
- Descripción de una propuesta de aplicación que permita almacenar los datos estadísticos de los jugadores de béisbol y a su vez el almacenamiento de los mismos.
- Desarrollo de los artefactos ingenieriles asociados al diseño de la propuesta de solución, a partir de la definición de la arquitectura.
- Implementación de los elementos necesarios para cumplir con las funcionalidades identificadas para el sistema de recopilación de datos estadísticos para los jugadores de béisbol.
- Validación y prueba de los resultados obtenidos para asegurar la calidad de la solución.

Métodos empíricos:

Observación: Se utiliza con el objetivo de diagnosticar el problema de modo que se pueda obtener información de cada elemento tratado en la investigación.

Entrevista: Se emplea con el objetivo de conocer la estructura organizativa del proceso de análisis estadístico de los jugadores de béisbol en Cuba.

Métodos teóricos:

Histórico-lógico: Empleado en el estudio de los datos almacenados en bibliografías para enriquecer el estudio y ayudar en la toma de decisiones en un partido de béisbol.

Analítico-Sintético: Se emplea para identificar los aspectos fundamentales relacionados al análisis estadístico en los partidos de béisbol.

La presente investigación está estructurada de la siguiente manera:

Capítulo 1: Fundamentación teórica del componente de obtención de datos estadísticos para Strike Zone: Este capítulo contiene toda la base teórica de la investigación realizada y también abarca el análisis de sistemas homólogos a Strike Zone. Se hace una descripción de las técnicas, estrategias y herramientas empleadas con dicho objetivo.

Capítulo 2: Descripción de la propuesta de solución para el componente de obtención de datos estadísticos para Strike Zone: En este capítulo se hace una breve descripción de la propuesta de solución, además se exponen las técnicas que utiliza la metodología escogida.

Capítulo 3: Implementación y pruebas del componente de obtención de datos estadísticos para Strike Zone: El contenido de este capítulo está relacionado a la validación del algoritmo propuesto, además se realizan distintos tipos de pruebas para el correcto funcionamiento del componente.

CAPÍTULO 1: Fundamentación teórica del Componente de obtención de datos estadísticos para Strike Zone

En este capítulo se describen los fundamentos teóricos que sustentan la presente investigación y se abordan términos que servirán para lograr una mejor comprensión del trabajo. También se realiza un estudio del arte, ofreciendo una descripción de las principales características de los lenguajes, tecnologías, herramientas y metodología de desarrollo de software que se pueden utilizar en la implementación del sistema propuesto, con el propósito de seleccionar la mejor opción para el desarrollo del mismo. Se realiza además un análisis de otras soluciones existentes con características similares al prototipo que se pretende desarrollar.

1.1 Conceptos asociados al dominio de la investigación

Para ayudar a entender el desarrollo de la investigación es necesario definir y analizar ciertos conceptos de relevancia, que a su vez brindan soporte teórico-metodológico y conceptual a la misma

1.1.1 Béisbol

El béisbol es considerado uno de los deportes, estadísticamente hablando, que más análisis se realizan entre los deportes colectivos; por ejemplo, cuando un pitcher hace un lanzamiento se analizan cosas como la velocidad de lanzamiento, el tipo de lanzamiento, si es strike o es bola, entre otros aspectos que se miden a la hora del pitcheo. Este deporte se practica en un campo cuadrado de 30 m de lado entre dos equipos de nueve jugadores cada uno; se trata de golpear con un bate una pequeña pelota lanzada con la mano por un contrario y recorrer el perímetro de un cuadrado interior del campo pasando por las cuatro esquinas o bases antes que el rival recupere el control del juego; gana el equipo que más veces consigue a lo largo de las nueve partes o entradas de que consta el partido.

Dicho deporte goza de gran popularidad en países caribeños en su mayoría y del norte de América, como por ejemplo Venezuela, República Dominicana, Puerto Rico, Cuba, Estados Unidos, México y Canadá, sin embargo, en algunos países asiáticos es también un deporte muy seguido. El béisbol es bastante antiguo sin embargo el juego tal y como se le conoce hoy en día fue desarrollado a inicios del siglo XVIII, tanto por niños como por jugadores amateurs. En Cuba específicamente este deporte es de los más populares, atrayendo a cientos de miles de espectadores a los estadios y a millones de seguidores mediante los medios de comunicación como la radio y la televisión. (Adrián, 2021)

La revolución del análisis estadístico surgida en las grandes ligas americanas procuró “La búsqueda del conocimiento objetivo del béisbol”, como lo definió Bill James, uno de los grandes gurús en el análisis matemático del deporte, trayendo consigo el surgimiento de una nueva tendencia en el proceso de toma de decisiones en el béisbol moderno conocida como sabermetría.

1.1.2 Sabermetría

La sabermetría tiene origen en el análisis estadístico de los datos generados por la Liga de Béisbol de los Estados Unidos (MLB, por sus siglas en inglés), obtiene la perspectiva del béisbol a través de evidencia objetiva, específicamente a través de estadísticas, cuyo fin es medir de manera eficaz las actividades que se suscitan dentro del campo de juego (César Soto Valero, 2015). El término es derivado del acrónimo SABR (por sus siglas en inglés de Society for American Baseball Research), el cual hace referencia a la Sociedad para la Investigación del Béisbol Americano, entidad que fue fundada en 1971 por Bob Davis. (Leyva, 2014)

Dicha ciencia ha sido definida como el análisis empírico del juego de béisbol a través del estudio de la evidencia objetiva obtenida, específicamente usando técnicas estadísticas (Soto Valero & González Castellanos, 2015), busca predecir el desempeño de un jugador en el futuro y así poder estimar su competitividad con fines de uso o para su contratación en equipos a través de agencias libres entre otros, analiza conceptos arraigados en el mundo del béisbol que no han sido lo suficientemente estudiados, con el fin de estimar su veracidad y eficacia. Ejemplo de esto sería el estudio del efecto que tiene un estadio determinado en las estadísticas de un jugador en particular, o la medida en que contribuye la defensa a la conquista de victorias de un equipo determinado (César Soto Valero, 2015). La sabermetría se concentra fundamentalmente en evaluar cómo afectan las estadísticas individuales y colectivas de los jugadores al récord de ganados y perdidos de los equipos de béisbol. Siguiendo esta lógica, para que un equipo sea exitoso debe ganar más juegos que sus oponentes, lo cual se logra anotando carreras o por medio de la prevención de éstas. Es por ello que la sabermetría se enfoca mayormente en la medición del aporte individual de cada jugador en términos de anotación de carreras, lo cual a la postre se traduce en victorias para todo el equipo (César Soto Valero, 2015).

Entre los usos que tiene la sabermetría están: evaluar el desempeño de diferentes jugadores, la comparación entre jugadores de distintas épocas, la predicción del desempeño de un jugador en el futuro y así poder estimar su competitividad con fines de uso y analizar conceptos arraigados en el mundo del béisbol que no han sido lo suficientemente estudiados con el fin de estimar su veracidad y eficacia. Por ejemplo, el estudio del efecto que tiene un estadio determinado en las estadísticas de un

jugador en particular y a su vez la medida en que contribuye la defensa a la conquista de victorias de un equipo determinado. (César Soto Valero, 2015)

A continuación, se detallan algunos de los principales estadísticos que utiliza la sabermetría el resto se encuentran en los anexos (anexos A.1):

Estadísticas de bateadores

- **Porcentaje de embasado (OBP):** Gracias a *Moneyball* y *Billy Beane*, el OBP es la estadística que comenzó la revolución saberométrica en el béisbol, demostrando que es más importante evitar que te hagan out (lo que mide el OBP) a dar un hit (lo que mide el AVG). Los jugadores con buen OBP (de .350 para arriba) suelen estar en los primeros lugares del lineup (1-4). Para obtener un buen OBP ayuda que el bateador sea paciente y tome un buen número de boletos (Corredor, 2013).
- **Slugging (SLG):** Mide el alcance de los batazos de un jugador. Cada base equivale a mil puntos (HR=4 000, 2B=2 000, etc); un turno fallido es 0. Un buen Slugging depende del poder del jugador y de su capacidad para mantener un buen promedio al bate (Corredor, 2013).
- **OBP+SLG (OPS):** Es simplemente la suma del OBP y el SLG para unir la utilidad de ambos en una sola estadística. El OPS promedio del béisbol es .750 (Corredor, 2013).

La sabermetría constituye una forma poderosa de análisis estadístico en el béisbol y busca maximizar los juegos ganados, basados en la optimización de recursos económicos por medio de la contratación de jugadores con base a la estadística; y no en la popularidad, imagen u otros factores comúnmente utilizados. (Leyva, 2014)

Cuando se hace un análisis saberométrico se tienen en cuenta todos los aspectos que envuelven al jugador en cuestión. Se analizan las cualidades físicas del mismo, sus porcentajes en el pasado y el aporte económico que este jugador puede significar para el equipo. Si bien desde hace años se han utilizado medidas para evaluar algunos factores de desempeño en los beisbolistas, lo que la sabermetría busca es incluir en esas calificaciones factores que pudieron hacer la diferencia en ese desempeño. De forma más específica, la sabermetría se centra en cómo afectan las **estadísticas** individuales y colectivas de los jugadores al récord de ganados y perdidos de un equipo de béisbol.

1.1.2 Estadística.

La estadística es una disciplina que se ocupa de la obtención, orden y análisis de un conjunto de datos con el fin de obtener explicaciones y predicciones sobre fenómenos observados (Roldán, 2017).

Consiste en métodos, procedimientos y fórmulas que permiten recolectar información para luego analizarla y extraer de ella conclusiones relevantes.

Se puede decir que es la Ciencia de los Datos y que su principal objetivo es mejorar la comprensión de los hechos a partir de la información disponible. Debido a la gran importancia que estas presentan, nacen los sistemas estadísticos para recoger y tener almacenada toda la información en un solo lugar (Roldán, 2017)

Tradicionalmente el análisis estadístico se ha hecho de forma manual donde el investigador tenía que efectuar por sí mismo los cálculos. Esto entrañaba una dificultad añadida en el proceso, pero también ofrecía ciertas ventajas, pues el investigador sabía en cada momento lo que estaba haciendo y cómo lo estaba haciendo (García, 2021).

1.1.3 Sistemas informáticos estadísticos.

La utilización de sistemas informáticos de análisis de datos ha supuesto una revolución porque se simplifica enormemente los procedimientos matemáticos y como consecuencia se emplean técnicas modernas que antes eran de muy difícil aplicación, se pasa del cálculo manual tradicional al cálculo por ordenador lo que supone una mayor rapidez en la obtención de los resultados, una mayor fiabilidad, un mejor manejo de estos procedimientos y una gran facilidad de aplicación y propicia una nueva forma de enseñanza del análisis de datos en los alumnos sin necesidad de una gran información matemática pueden efectuar operaciones que en ocasiones son muy complejas (García, 2021).

1.2 Estudio de sistemas homólogos

En el mundo existen muchas herramientas y sistemas que utilizan el estudio Sabermétrico para el mejoramiento de los resultados de los deportistas de forma individual y de forma colectiva como un equipo. A continuación, se realiza un estudio de algunas de ellas:

Sistema PITCHf/X

PITCHf / x, creado y mantenido por Sportvision, es un sistema que rastrea las velocidades y trayectorias de pelotas de béisbol lanzadas. Este sistema, que hizo su debut en los playoffs de la MLB de 2006, está instalado en todos los estadios de la MLB. Los radiodifusores suelen utilizar los datos del sistema para mostrar una representación visual del lanzamiento y si un lanzamiento entró o no en la zona de strike. PITCHf / x también se usa para determinar el tipo de lanzamiento lanzado, como una

bola rápida, una curva o un control deslizante. MLB usa los datos de PITCHf/x en su Sistema de Evaluación de Zona que se usa para calificar y proporcionar retroalimentación a los árbitros. Los analistas de Sabermetría señalan que la precisión de los árbitros ha mejorado después de que se introdujo la tecnología en MLB. PITCHf/x es un sistema que usa tres cámaras montadas permanentemente en el estadio para rastrear la velocidad y la ubicación de una pelota de béisbol lanzada desde el montículo del lanzador hasta el plato de home con una precisión de mejor de una milla por hora y una pulgada. Con PITCHf/x, se pueden analizar estadísticas como el lanzador con la bola rápida más rápida o el lanzador con la curva de ruptura más aguda, etc. Calcula el "movimiento" de los tonos provocados por la fuerza de Magnus. (Spantip, s.f.)

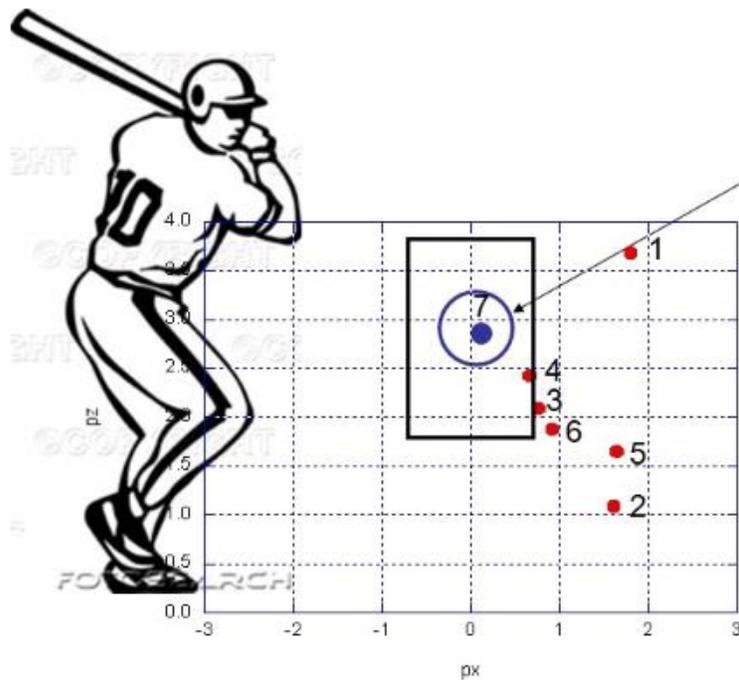
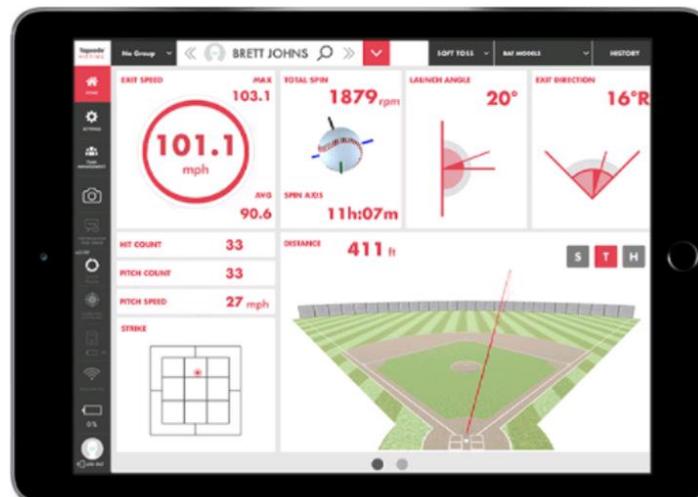


Ilustración 1: Sistema PITCHf/x

Herramienta Rapsodo

Rapsodo, la compañía de datos deportivos conocida por ayudar a los atletas a analizar métricas en tiempo real para perfeccionar su juego. Es un dispositivo móvil que rastrea el vuelo de cada lanzamiento utilizando tecnología de seguimiento óptico, está configurado a 15 pies y 6 pulgadas frente a la placa de inicio y luego se conecta aun iPad a través de una señal inalámbrica, para proporcionar retroalimentación instantánea en un tablero de fácil lectura en la aplicación. Rapsodo PITCHING utiliza

tecnología de cámara y radar que mide y analiza todos los puntos de datos posibles de un tono en vuelo para brindarle un análisis de tono como nunca antes. Esta unidad se integra directamente en la aplicación Diamond, donde puede ver la velocidad de giro, el eje de giro, la eficiencia, la rotura horizontal, vertical y la velocidad de cada lanzamiento. Con características como Auto Pitch Tagging, análisis de la zona de strike, información de lanzamiento y retroalimentación de video inmediata, los jugadores y entrenadores pueden analizar cada representante de un lanzamiento por individual, o ver sesiones completas de bullpen y datos históricos dentro de Rapsodo Cloud. Rapsodo se ha convertido en un líder en la industria por ayudar en el proceso de diseño del tema, brindando una tonelada de información en cada toma, brinda retroalimentación instantánea crucial para el proceso de diseño de la cancha y actualmente Rapsodo está siendo utilizada en Japón comprometido a brindar un apoyo excepcional a entrenadores y Sabermétricos. (Rapsodo, 2019)



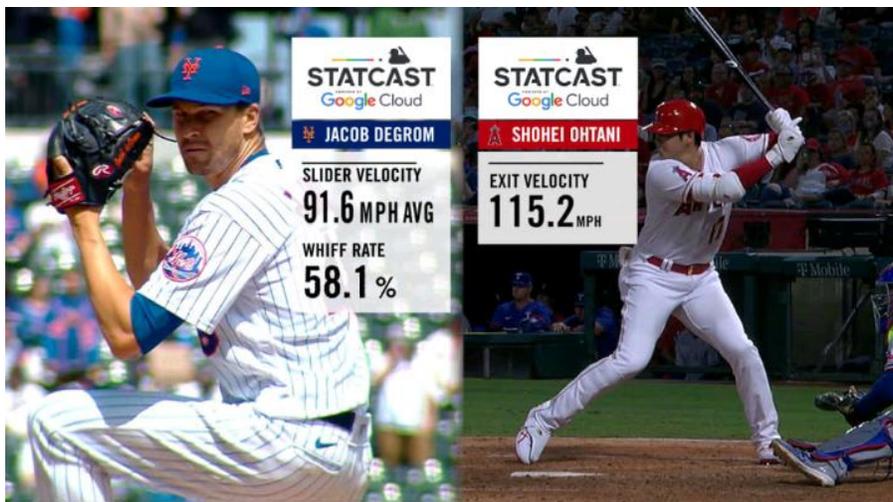
Rapsodo Hitting

Ilustración 2: Herramienta Rapsodo

Tecnología Statcast

La tendencia actualmente en el béisbol es tratar de recolectar la mayor cantidad de datos posibles, para poder tomar decisiones más acertadas. Statcast está compuesto por dos sistemas Trackman y Chyronhego cameras, estos dos sistemas están instalados en los 30 estadios de la MLB. La primera parte de este sistema es Trackman, este es responsable de medir todo lo que tiene que ver con la pelota. Esto incluye lo rápido que está rotando, lo rápido que fue lanzada y lo lejos que fue bateada y

hacia que parte del terreno. Trackman usa un radar Dopple de tecnología de grado militar, reutilizado desde el mismo sistema que usa la armada para rastrear balas y misiles. Luego están los Chyronhego cameras responsables de todo lo que suceda con respecto al movimiento de los jugadores. Estas instaladas en dos bancos de tres cámaras, en las líneas de foul. Su función es medir velocidad, distancia en el desplazamiento y la ubicación de cada jugador en el terreno de juego. De aquí es donde proviene la calificación de las jugadas. Utiliza las variables anteriormente nombradas para determinar la probabilidad de una atrapada, y otorga una calificación a cada una. Estos dos sistemas se combinan para crear siete terabytes de datos por juego. Esta información es recolectada y digerida para analizar en profundidad el rendimiento de los jugadores. (MLB, 2021)



Statcast Awards for 2021 season

Ilustración 3: Sistema Statcast

K Zone

Una Herramienta que comenzó su proceso de prueba en 2006 y ya en 2007 era una de las más utilizadas en la MLB, ESPN presentó "K Zone 2.0", una herramienta utilizada para el apoyo de los entrenadores y Sabermétrico de las grandes ligas MLB en la toma de decisiones, la cual agregó la capacidad de rastrear el camino de una pelota desde la mano del lanzador hasta el plato de home y la capacidad de mostrar una secuencia completa de lanzamientos a un bateador en orden numérico. Algunos pueden hacerlo por el ritmo de juego, otros por la duración de los juegos. K-Zone de ESPN, es un cuadrado superpuesto que muestra una zona de strike no oficial basada en la entrada de tres

cámaras de seguimiento, la cual ha sido visible durante la mayoría de las transmisiones de la cadena y actualmente usada en los 30 estadios de la MLB. (InsideHook, 2018)



Why ESPN's new K-Zone should be given a chance

Ilustración 4: Sistema K-Zone

Estas herramientas son importantes en su dominio de aplicación, sin embargo, por su tema y filosofía de enseñanza-aprendizaje no se adapta a las necesidades de esta investigación. No se cuenta con el código fuente para su modificación y no cumple con las políticas de soberanía tecnológica definidas en Cuba para la informatización del país y aunque no está acorde a lo que se necesita, motivan el desarrollo de una solución propia ajustada a la medida de las necesidades identificadas en la problemática de esta investigación.

En Cuba se tiene muy poco desarrollo con respecto a la sabermetría lo cual sumado con otros procesos han condicionado obtener un bajo nivel en el béisbol y a pesar de que se han desarrollado sistemas como STRIKE ZONE no son capaces de recopilar suficiente información ya que es necesaria la obtención de un gran número de datos.

1.3 Metodologías, Herramientas y Tecnologías

Uno de los factores primordiales que permiten un buen desarrollo de software son las herramientas y las tecnologías que se utilizan brindando calidad y eficiencia a la hora de trabajar con los mismos. En la actualidad existen muchas herramientas que facilitan el proceso de desarrollo de software como por ejemplo los entornos de desarrollo, el tipo de lenguaje de programación, sistemas gestores de bases de datos, entre otros. A continuación, se muestran algunas de las herramientas y tecnologías antes

mencionadas el cual están sujetas a las políticas de desarrollo del proyecto PITCH UCI del cual se obtienen datos para la descripción de la propuesta de solución.

1.3.1 Metodología del desarrollo de software

Metodología se denomina a la serie de métodos y técnicas de rigor científico que se aplican sistemáticamente durante un proceso de investigación para alcanzar un resultado teóricamente válido; funciona como el soporte conceptual que rige la manera en que aplicamos los procedimientos en una investigación (Coelho, 2019)

La metodología de desarrollo de software que se selecciona para guiar el desarrollo de la propuesta de solución es Programación Extrema (XP). En esta metodología todos los requerimientos se expresan como escenarios (llamados historias de usuarios), los cuales se implementan directamente como una serie de tareas. Los programadores trabajan en parejas y desarrollan pruebas para cada tarea antes de escribir el código, el desarrollo incremental se lleva a cabo a través de entregas pequeñas y frecuentes, la participación del cliente se lleva a cabo a través del compromiso a tiempo completo del cliente con el equipo de desarrolladores, el cambio se lleva a cabo a partir de las entregas regulares, entre otros.

Para modelar los artefactos que se generan se utilizará el Lenguaje de Modelado Unificado (UML, por sus siglas en inglés), dicho lenguaje fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento y como herramienta de Ingeniería de Software Asistida por Computadora (CASE, por sus siglas en inglés) Visual Paradigm for UML, el cual, es una herramienta para el desarrollo de aplicaciones que utiliza el lenguaje UML, ofreciendo confiabilidad y estabilidad en el proceso de desarrollo. Soporta el ciclo de vida completo del software: análisis y desarrollo orientados a objetos, construcción, prueba y despliegue. Brinda un diseño centrado en casos de uso y enfocado al negocio, lo que genera un software de mayor calidad, además permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y la generación de documentación. Ambos elementos son los que monopolizarán el desarrollo de los proyectos productivos junto a la metodología seleccionada.

1.3.2 Entorno de desarrollo integrado

Como entorno de desarrollo se escogió para el desarrollo de la aplicación QT Creator ya que es un IDE multiplataforma programado en C++, JavaScript y QML creado por Trolltech el cual es parte de SDK para el desarrollo de aplicaciones con interfaces gráficas de Usuario con las bibliotecas de Qt, es un editor de código con soporte C++, QML y ECMAscript, proporciona resultado de sintaxis y autocompletado de código, control estático de código y estilo a medida que se escribe, ayuda sensitiva al contexto, entre otras (DesdeLinux, 2012).

1.3.3 Lenguajes de programación

Se utilizará C++ como lenguaje de programación debido al poder y la flexibilidad que posee. Entre las principales características que presenta este lenguaje de programación es la sintaxis heredada del lenguaje C, tiene un standard ISO, conocido como ANSI-C++, lenguaje fuertemente tipado, programación orientada a objetos, sobre carga de operadores, soporta expresiones Lambda también llamadas funciones anónimas, control de excepciones, biblioteca estándar, compatibilidad de C, uso de punteros y eficiencia con el hardware. Gracias al poder y la flexibilidad que posee y a que es orientado a objeto, se escoge C++ como lenguaje de programación para el desarrollo de la aplicación informática, ya que, es un lenguaje fuertemente tipado, soporta extensiones Lambda también llamadas funciones anónimas, presenta un gran control de las excepciones, entre otras características. (Content, 2019)

1.3.4 Sistema gestor de base de datos

Debido a que el gestor de base de datos es uno de los pilares para el desarrollo de la aplicación informática se decidió utilizar SQLite para el desarrollo del mismo ya que es un motor de base de datos SQL incorporado, no tiene proceso de servidor separado y es capaz de leer y escribir directamente en archivos de discos ordinarios. Es una base de datos completa con múltiples tablas, índices, activadores y vistas que está contenida en un solo archivo de disco. Es multiplataforma por lo que se puede copiar libremente entre sistemas de 32 y 64 bits y contiene una biblioteca compacta por lo que sus funcionalidades están habilitadas.

1.4 Conclusiones parciales

Durante este capítulo se abordaron los elementos teóricos que sustentan a la propuesta de solución del problema planteado, en este sentido se concluye:

- El análisis de los sistemas homólogos permitió identificar las principales funcionalidades que debe tener el sistema a desarrollar: consultas a la base de datos y gestión de la información.
- El estudio de las principales tendencias en el desarrollo de este dominio de aplicaciones, permitió la selección de las herramientas, tecnologías y metodología a utilizar en la presente investigación.
- La selección de las herramientas y lenguaje de programación a utilizar aseguró el ambiente de desarrollo necesario para la implementación de la propuesta de solución, en el cual se definió como lenguaje de programación C++, como entorno de desarrollo se escogió Qt Creator, como gestor de base de datos SQLite.

CAPÍTULO 2: Descripción de la propuesta de solución para el componente de obtención de datos estadísticos para Strike Zone

El presente capítulo aborda los principales aspectos relacionados con las características de la propuesta de solución. Se identifican los requisitos funcionales y no funcionales con los que debe cumplir la solución propuesta, así como estilo arquitectónico y los patrones de diseño para lograr buenas prácticas en el diseño y posterior implementación del sistema. Igualmente se muestran los principales artefactos de ingeniería de software propuestos para la metodología utilizada.

2.1 Descripción de la propuesta de solución

Dadas las necesidades planteadas en la situación problemática de la presente investigación, la solución propuesta constituye un componente de obtención de datos estadísticos para el software Strike Zone. El componente contará con una ventana principal, la misma, con un sub menú en la barra superior en el lado izquierdo que tendrá las ventanas para registrar y modificar a un deportista; estas presentarán tres campos editables en las que el usuario ingresará la información del deportista, dos botones: el primero para agregar o modificar la información ingresada y el otro para cancelar en caso de no querer realizar dichas acciones. Luego de interactuar con las ventanas de registrar y modificar deportista, el componente mostrará la información añadida en una tabla ubicada en el lado derecho del sistema en la que el usuario tendrá la opción de seleccionar o no a un deportista; al lado izquierdo de la tabla se encontrarán un conjunto de botones ordenados uno debajo del otro, que, dichos botones, abrirán cada uno las ventanas correspondientes a los cálculos que se desea realizar; cabe recalcar que para acceder a alguna de estas ventanas se deberá seleccionar previamente a un deportista. Las ventanas de cálculos contarán con un campo no editable donde se mostrará el nombre del deportista seleccionado, debajo de este los campos editables para ingresar los valores de las variables que se van a utilizar; las ventanas contendrán cuatro botones principales para realizar las acciones de “Calcular” para el cálculo de los valores ingresados en los campos mostrando a su vez el resultado y el comentario referente a dicho resultado, “Guardar” para guardar la información del cálculo y mostrar la misma en la tabla, “Actualizar” para actualizar la información de la tabla y “Eliminar” para eliminar la información de la misma. La tabla antes mencionada estará en el lado derecho de los campos a editar. El componente de gestión se desarrollará sobre la base de una aplicación de escritorio facilitando a los usuarios los consecuentes beneficios que estas aplicaciones reportan, los datos están más centralizados, menor tiempo de desarrollo y no depende de tener internet. La gestión de la información se realizará a través de una base de datos localizada en el servidor.

2.2 Especificación de requisitos de software

La especificación de requisitos del sistema, SyRS por sus siglas en inglés, es un esquema detallado de los requerimientos necesarios para crear un sistema completo. Estos requisitos se documentan en un esfuerzo por definir la funcionalidad completa, la disponibilidad, el rendimiento y las necesidades de seguridad de un sistema. (Garcia, 2020)

Con el objetivo de que el sistema propuesto se desarrolle con las necesidades dictadas por el cliente final a partir del uso de la tormenta de ideas y la entrevista con el cliente, se obtuvieron los requisitos funcionales y no funcionales.

2.2.1 Requisitos funcionales

Teniendo en cuenta las características de la propuesta en cuanto a prestaciones y operatividad se identifican 60 requisitos funcionales. A continuación, la tabla 1 muestra los requisitos del sistema.

Tabla 1: Requisitos funcionales del sistema

N.º	Nombre	Prioridad	Complejidad
RF1	Visualizar información OBP	Alta	Baja
RF2	Ingresar información OBP	Alta	Media
RF3	Calcular datos OBP	Alta	Alta
RF4	Guardar información OBP	Alta	Media
RF5	Mostrar información OBP	Media	Baja
RF6	Limpiar campos OBP	Baja	Baja
RF7	Visualizar resultado OBP	Media	Baja
RF8	Visualizar comentario OBP	Media	Baja
RF9	Eliminar OBP	Baja	Baja
RF10	Visualizar información SLG	Alta	Baja
RF11	Ingresar información SLG	Alta	Media
RF12	Calcular datos SLG	Alta	Alta
RF13	Guardar información SLG	Alta	Media
RF14	Mostrar información SLG	Media	Baja
RF15	Limpiar campos SLG	Baja	Baja
RF16	Visualizar resultado SLG	Media	Baja

RF17	Visualizar comentario SLG	Media	Baja
RF18	Eliminar SLG	Baja	Baja
RF19	Visualizar información OPS	Alta	Baja
RF20	Ingresar información OPS	Alta	Media
RF21	Calcular datos OPS	Alta	Alta
RF22	Guardar información OPS	Alta	Media
RF23	Mostrar información OPS	Media	Baja
RF24	Limpiar campos OPS	Baja	Baja
RF25	Visualizar resultado OPS	Media	Baja
RF26	Visualizar comentario OPS	Media	Baja
RF27	Eliminar OPS	Baja	Baja
RF28	Visualizar información BABIP	Alta	Baja
RF29	Ingresar información BABIP	Alta	Media
RF30	Calcular datos BABIP	Alta	Alta
RF31	Guardar información BABIP	Alta	Media
RF32	Mostrar información BABIP	Media	Baja
RF33	Limpiar campos BABIP	Baja	Baja
RF34	Visualizar resultado BABIP	Media	Baja
RF35	Visualizar comentario BABIP	Media	Baja
RF36	Eliminar BABIP	Baja	Baja
RF37	Visualizar información BB%	Alta	Baja
RF38	Ingresar información BB%	Alta	Media
RF39	Calcular datos BB%	Alta	Alta
RF40	Guardar información BB%	Alta	Media
RF41	Mostrar información BB%	Media	Baja
RF42	Limpiar campos BB%	Baja	Baja
RF43	Visualizar resultado BB%	Media	Baja
RF44	Visualizar comentario BB%	Media	Baja
RF45	Eliminar BB%	Baja	Baja
RF46	Visualizar información ISO	Alta	Baja
RF47	Ingresar información ISO	Alta	Media

RF48	Calcular datos ISO	Alta	Alta
RF49	Guardar información ISO	Alta	Media
RF50	Mostrar información ISO	Media	Baja
RF51	Limpiar campos ISO	Baja	Baja
RF52	Visualizar resultado ISO	Media	Baja
RF53	Visualizar comentario ISO	Media	Baja
RF54	Eliminar ISO	Baja	Baja
RF55	Visualizar información RC	Alta	Baja
RF56	Ingresar información RC	Alta	Media
RF57	Calcular datos RC	Alta	Alta
RF58	Guardar información RC	Alta	Media
RF59	Mostrar información RC	Media	Baja
RF60	Limpiar campos RC	Baja	Baja
RF61	Visualizar resultado RC	Media	Baja
RF62	Visualizar comentario RC	Media	Baja
RF63	Eliminar RC	Baja	Baja
RF64	Registrar deportista	Alta	Media
RF65	Modificar deportista	Alta	Media
RF66	Eliminar deportista	Media	Media
RF67	Listar deportista	Media	Baja

2.2.2 Requisitos no funcionales

Los requisitos no funcionales se originan en la necesidad del usuario, debido a las restricciones presupuestarias, políticas organizacionales, la necesidad de interoperabilidad con otros sistemas de software o hardware, o factores externos tales como regulaciones de seguridad, políticas de, privacidad, entre otros (blog, 2018).

- **Usabilidad**

- RnF1 Tipo de aplicación informática: La herramienta debe ser una aplicación de escritorio.
- RnF2 El sistema debe poseer una interfaz fácil de utilizar para cualquier tipo de usuario con conocimientos básicos de computación en el manejo de ordenadores.

- RnF3 El tamaño de la letra en el componente se debe encontrar entre 16 y 20.
- RnF4 Los botones deben estar distribuidos de manera tal que el usuario interactúe con ellos de manera intuitiva, proporcionando consistencia a las interfaces.
- **Hardware y software requerido para utilizar la aplicación**
 - RnF5 Sistema Operativo: Podrá utilizar Windows como sistema operativo.
 - RnF6 La computadora que se utilice debe contar mínimo con 2 GB de RAM y de almacenamiento 100 GB. La computadora servidor debe contar con una RAM de 2 GB y un almacenamiento de al menos 250 GB
 - RnF7 Se debe garantizar un consumo de RAM y CPU inferior al 80%.

2.3 Descripción de las historias de usuario

Para el encapsulamiento de los requisitos de software, tal como lo plantea la metodología XP y utilizando el estudio de sistemas homólogos y la entrevista con el cliente, se generaron las historias de usuario para cada uno de los requisitos funcionales adquiridos y para un mejor entendimiento de los mismos se agruparon dichos requisitos en diferentes CRUD (Crear, Eliminar, Actualizar, Guardar) A continuación, se muestran las correspondientes a Gestionar deportista y Administrar OBP; el resto se encuentra en los anexos (Ver Anexo A.2).

Tabla 2: Historia de usuario 1 "Registrar deportista"

Historia de usuario	
Número: 1	Nombre: Registrar deportista
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción:	

El componente debe permitir al usuario ingresar la información de un deportista y a su vez guardar la misma teniendo en cuenta de que todos los campos en la ventana son obligatorios: nombre y apellidos, carné de identidad y número de jugador

Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se presione el botón de “Registrar deportista”.

En caso de que no se llene uno de los campos en la ventana, el sistema mostrará un mensaje de alerta con el mensaje “Faltan campos por llenar”

El usuario tiene la opción de agregar la información introducida o cancelarla en caso de que no quiera realizar la acción de registrar un deportista.

Interfaz



The screenshot shows a window titled "Registrar deportista" with a light blue header and standard window controls (minimize, maximize, close). The main content area has a light gray background and is titled "Registrar deportista" in bold black text. Below the title, there are three input fields, each with a label to its left: "Nombre y apellidos" containing "Julio Cesar Betancourt Ramos", "Carné de identidad" containing "99030607241", and "Número de jugador" containing "54". At the bottom of the form, there are two buttons: "Agregar" and "Cancelar".

Tabla 3: Historia de usuario 2 "Modificar deportista"

Historia de usuario	
Número: 2	Nombre: Modificar deportista
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
<p>Descripción:</p> <p>El componente debe permitir al usuario modificar la información de un deportista y a su vez guardar la misma teniendo en cuenta de que todos los campos en la ventana son obligatorios: nombre y apellidos, carné de identidad y número de jugador.</p>	
<p>Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y se presione el botón de “Modificar deportista”.</p> <p>En caso de haber algún campo vacío, se muestra un mensaje de alerta con el mensaje “Faltan campos por llenar”</p> <p>El usuario tiene la opción de modificar la información de un deportista o cancelar, en caso de que no desee realizar esta acción.</p>	
Interfaz	

Tabla 4: Historia de usuario 3 "Eliminar deportista"

Historia de usuario	
Número: 3	Nombre: Eliminar deportista
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 0.3	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción:	
El componente debe permitir al usuario eliminar la información de un deportista.	

Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un jugador y se presione el botón eliminar en la pantalla principal.

En caso de que no se seleccione a un jugador, se muestra una alerta con el mensaje “Debe seleccionar un deportista para poder eliminar”

Luego de realizar todas las acciones satisfactoriamente el componente muestra una alerta con el mensaje “El deportista ha sido eliminado satisfactoriamente”

Interfaz

The screenshot shows a window titled 'SZ' with a sidebar on the left containing several options: 'Porcentaje de embasado' (OBP), 'Slugging' (SLG), 'OBP + SLG' (OPS), 'Promedio de pelotas en juego' (BABIP), 'Frecuencia de bases por bola' (BB%), 'Poder aislado' (ISO), and 'Carreras creadas' (RC). The main area is titled 'ESTADÍSTICOS' and contains a table with the following data:

Nombre y Apellidos	Carné de identidad	Numero de jugador	OBP
Julio Cesar Betancourt Ramos	99030607241	16	0.356
Jesus Francisco Vázquez Biltre	98100125478	53	0.222
Daniel Aguilar Torres	99101326487	18	1.24
Luis Ernesto Clarke Samuels	99110469874	3	5.2
Carlos Enrique González Prieto	98021456897	5	0.1
Yunikuo Ernesto Beltrán Miranda	91032547896	47	1.56
Robert Maceira	8201236547	8	0.89
Marcos Estrada Pérez	8501236547	69	0.78

At the bottom right of the window are two buttons: 'Eliminar' and 'Actualizar'.

The dialog box titled 'Alerta' contains the text: "Debe seleccionar un deportista para poder eliminar". At the bottom, there are two buttons: 'Aceptar' and 'Cancelar'.

The dialog box titled 'Alerta' contains the text: "El deportista ha sido eliminado satisfactoriamente". At the bottom, there are two buttons: 'Aceptar' and 'Cancelar'.

Tabla 5: Historia de usuario 4 "Listar deportista"

Historia de usuario

Número: 4	Nombre: Listar deportista																																				
Usuario: usuario																																					
Prioridad: Alta	Riesgo en desarrollo: Alto																																				
Puntos estimados: 0.3	Iteración: 1																																				
Programador a cargo: Julio Cesar Betancourt Ramos																																					
<p>Descripción:</p> <p>El componente debe permitir al usuario listar y mostrar la información de un deportista en la tabla de la pantalla principal.</p>																																					
<p>Observaciones: Esta historia de usuario se ejecutará desde que se abre el componente y la información se actualiza luego de presionar el botón actualizar en la pantalla principal.</p>																																					
<p>Interfaz</p>  <p>The screenshot shows a window titled 'SZ' with a sidebar of options and a main table of statistics. The options include: Porcentaje de embasado (OBP), Slugging (SLG), OBP + SLG (OPS), Promedio de pelotas en juego (BABIP), Frecuencia de bases por bola (BB%), Poder aislado (ISO), and Carreras creadas (RC). The table, titled 'ESTADÍSTICOS', has columns for 'Nombre y Apellidos', 'Carné de identidad', 'Numero de jugador', and 'OBP'. The data rows are as follows:</p> <table border="1"> <thead> <tr> <th>Nombre y Apellidos</th> <th>Carné de identidad</th> <th>Numero de jugador</th> <th>OBP</th> </tr> </thead> <tbody> <tr> <td>Julio Cesar Betancourt Ramos</td> <td>99030607241</td> <td>16</td> <td>0.356</td> </tr> <tr> <td>Jesus Francisco Vázquez Biltre</td> <td>98100125478</td> <td>53</td> <td>0.222</td> </tr> <tr> <td>Daniel Aguilar Torres</td> <td>99101326487</td> <td>18</td> <td>1.24</td> </tr> <tr> <td>Luis Ernesto Clarke Samuels</td> <td>99110469874</td> <td>3</td> <td>5.2</td> </tr> <tr> <td>Carlos Enrique González Prieto</td> <td>98021456897</td> <td>5</td> <td>0.1</td> </tr> <tr> <td>Yunikuo Ernesto Beltrán Miranda</td> <td>91032547896</td> <td>47</td> <td>1.56</td> </tr> <tr> <td>Robert Maceira</td> <td>8201236547</td> <td>8</td> <td>0.89</td> </tr> <tr> <td>Marcos Estrada Pérez</td> <td>8501236547</td> <td>69</td> <td>0.78</td> </tr> </tbody> </table> <p>At the bottom right of the window are buttons for 'Eliminar' and 'Actualizar'.</p>		Nombre y Apellidos	Carné de identidad	Numero de jugador	OBP	Julio Cesar Betancourt Ramos	99030607241	16	0.356	Jesus Francisco Vázquez Biltre	98100125478	53	0.222	Daniel Aguilar Torres	99101326487	18	1.24	Luis Ernesto Clarke Samuels	99110469874	3	5.2	Carlos Enrique González Prieto	98021456897	5	0.1	Yunikuo Ernesto Beltrán Miranda	91032547896	47	1.56	Robert Maceira	8201236547	8	0.89	Marcos Estrada Pérez	8501236547	69	0.78
Nombre y Apellidos	Carné de identidad	Numero de jugador	OBP																																		
Julio Cesar Betancourt Ramos	99030607241	16	0.356																																		
Jesus Francisco Vázquez Biltre	98100125478	53	0.222																																		
Daniel Aguilar Torres	99101326487	18	1.24																																		
Luis Ernesto Clarke Samuels	99110469874	3	5.2																																		
Carlos Enrique González Prieto	98021456897	5	0.1																																		
Yunikuo Ernesto Beltrán Miranda	91032547896	47	1.56																																		
Robert Maceira	8201236547	8	0.89																																		
Marcos Estrada Pérez	8501236547	69	0.78																																		

Tabla 6: Historia de usuario 5 "Visualizar información OBP"

Historia de usuario																	
Número: 5	Nombre: Visualizar información OBP																
Usuario: usuario																	
Prioridad: Alta	Riesgo en desarrollo: Medio																
Puntos estimados: 0.4	Iteración: 1																
Programador a cargo: Julio Cesar Betancourt Ramos																	
Descripción: El componente debe permitir al usuario visualizar la información de los cálculos de un deportista																	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "OBP".																	
Interfaz																	
<p>The screenshot shows a window titled 'SZ' with a form for calculating OBP. On the left, there are input fields for 'Nombre' (Julio Cesar Betancourt Ramos), 'H (Hits)' (5), 'BB (Bases por bola)' (6), 'HBP (Golpeado por levantaiento)' (3), 'AB (Veces al bate)' (8), and 'SF (Batazos elevados de sacrificio)' (1). Below these are 'Calcular' and 'Guardar' buttons. The 'Resultado' is 0.356, and a message says 'El rendimiento del bateador es bajo'. On the right, a table displays the data:</p> <table border="1"> <thead> <tr> <th>Nombre</th> <th>H</th> <th>BB</th> <th>HBP</th> <th>AB</th> <th>SF</th> <th>Resu...</th> <th>Co...</th> </tr> </thead> <tbody> <tr> <td>Julio Cesar B...</td> <td>5</td> <td>6</td> <td>3</td> <td>8</td> <td>1</td> <td>0.356</td> <td>El ren...</td> </tr> </tbody> </table> <p>At the bottom of the window are 'Borrar', 'Eliminar', and 'Actualizar' buttons.</p>		Nombre	H	BB	HBP	AB	SF	Resu...	Co...	Julio Cesar B...	5	6	3	8	1	0.356	El ren...
Nombre	H	BB	HBP	AB	SF	Resu...	Co...										
Julio Cesar B...	5	6	3	8	1	0.356	El ren...										

Tabla 7: Historia de usuario 6 "Ingresar información OBP"

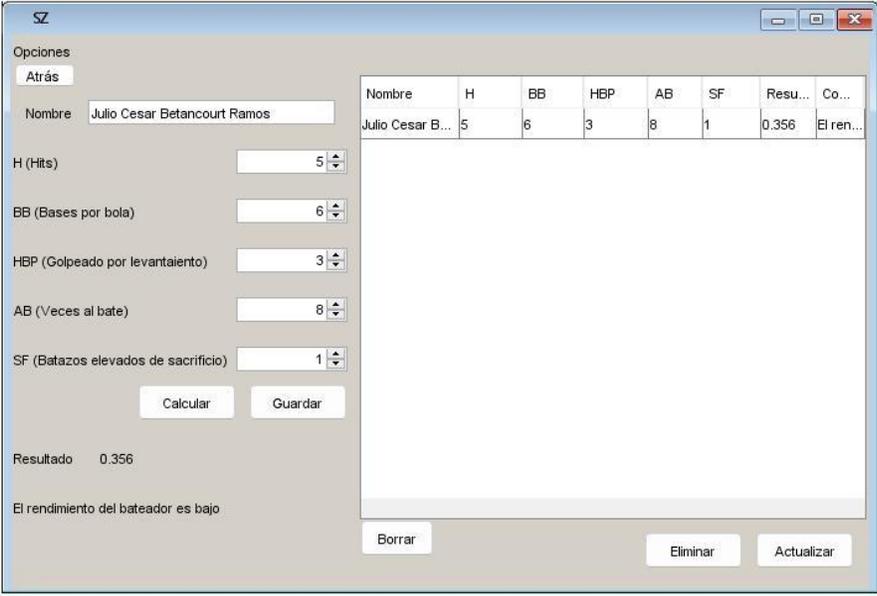
Historia de usuario																	
Número: 6	Nombre: Ingresar información OBP																
Usuario: usuario																	
Prioridad: Alta	Riesgo en desarrollo: Medio																
Puntos estimados: 0.4	Iteración: 1																
Programador a cargo: Julio Cesar Betancourt Ramos																	
<p>Descripción:</p> <p>El componente debe permitir al usuario ingresar la información de los cálculos de un deportista teniendo en cuenta de que el campo nombre no es editable y los campos H, BB, HBP, AB y SF son obligatorios.</p> <p>Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de “OBP”.</p>																	
<p>Interfaz</p>  <table border="1" data-bbox="743 1178 1252 1619"> <thead> <tr> <th>Nombre</th> <th>H</th> <th>BB</th> <th>HBP</th> <th>AB</th> <th>SF</th> <th>Resu...</th> <th>Co...</th> </tr> </thead> <tbody> <tr> <td>Julio Cesar B...</td> <td>5</td> <td>6</td> <td>3</td> <td>8</td> <td>1</td> <td>0.356</td> <td>El ren...</td> </tr> </tbody> </table>		Nombre	H	BB	HBP	AB	SF	Resu...	Co...	Julio Cesar B...	5	6	3	8	1	0.356	El ren...
Nombre	H	BB	HBP	AB	SF	Resu...	Co...										
Julio Cesar B...	5	6	3	8	1	0.356	El ren...										

Tabla 8: Historia de usuario 7 "Calcular datos OBP"

Historia de usuario	
Número: 7	Nombre: Calcular datos OBP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
<p>Descripción:</p> <p>El componente debe permitir al usuario realizar el cálculo del estadístico a partir de los datos introducidos teniendo en cuenta de que el campo nombre no es editable y los campos H, BB, HBP, AB y SF son obligatorios.</p>	
<p>Observaciones: Esta historia de usuario se ejecutará luego de haber introducido la información del cálculo en el sistema y a su vez después de haber presionado el botón de "Calcular".</p> <p>En caso de que alguno de esos campos no esté lleno, se muestra el mensaje "Para realizar el cálculo debe llenar los campos"</p>	
Interfaz	

Opciones

 Nombre: Julio Cesar Betancourt Ramos

Nombre	H	BB	HBP	AB	SF	Resu...	Co...
Julio Cesar B...	5	6	3	8	1	0.356	El ren...

H (Hits): 5
 BB (Bases por bola): 6
 HBP (Golpeado por levantaiento): 3
 AB (Veces al bate): 8
 SF (Batazos elevados de sacrificio): 1

Resultado: 0.356
 El rendimiento del bateador es bajo

Alerta
 Para realizar el cálculo debe llenar los campos

Tabla 9: Historia de usuario 8 "Guardar información OBP"

Historia de usuario	
Número: 8	Nombre: Guardar información OBP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	

Descripción:

El componente debe permitir al usuario guardar la información del cálculo estadístico.

Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo y a su vez después de haber presionado el botón de “Guardar”.

En caso de que el usuario desee guardar la información si haber llenado los campos o faltando alguno de ellos por llenar, el componente muestra el mensaje “Para guardar los datos debe llenar los campos”.

En caso de que la información se guarde satisfactoriamente, el componente muestra el mensaje “Los datos se han guardado satisfactoriamente”

Interfaz

The screenshot shows a window titled "SZ" with a form for entering player statistics. The form includes a "Nombre" field with the value "Julio Cesar Betancourt Ramos", and five numeric input fields: "H (Hits)" with value 5, "BB (Bases por bola)" with value 6, "HBP (Golpeado por levantaiento)" with value 3, "AB (Veces al bate)" with value 8, and "SF (Batazos elevados de sacrificio)" with value 1. Below these fields are "Calcular" and "Guardar" buttons. The "Resultado" is displayed as 0.356, and a message states "El rendimiento del bateador es bajo". A table on the right shows the following data:

Nombre	H	BB	HBP	AB	SF	Resu...	Co...
Julio Cesar B...	5	6	3	8	1	0.356	El ren...

At the bottom of the window are "Borrar", "Eliminar", and "Actualizar" buttons.

An alert dialog box titled "Alerta" with the message "Para guardar los datos debe llenar los campos". It contains "Aceptar" and "Cancelar" buttons.

An alert dialog box titled "Alerta" with the message "Los datos se han guardado satisfactoriamente". It contains "Aceptar" and "Cancelar" buttons.

Tabla 10: Historia de usuario 9 "Mostrar información OBP"

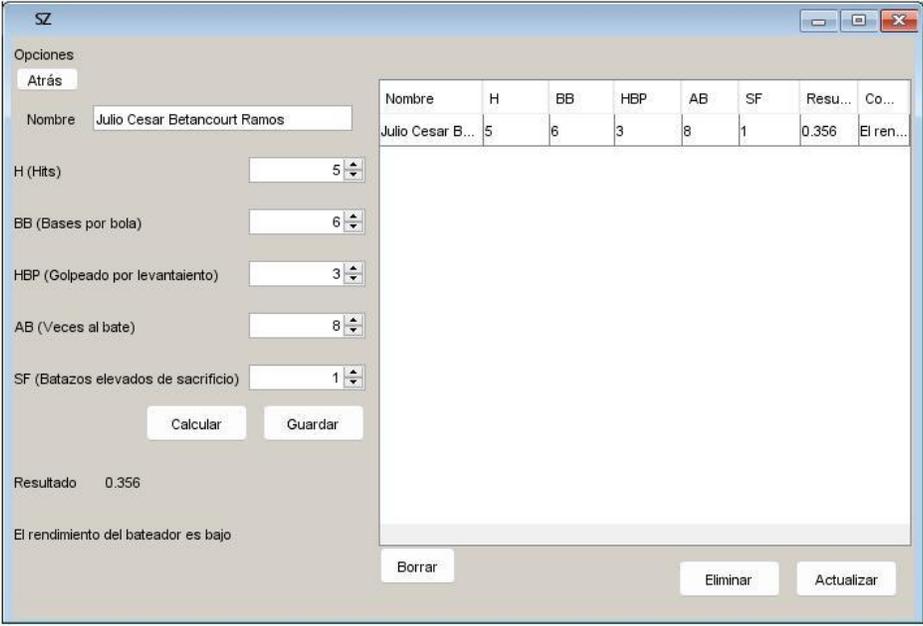
Historia de usuario	
Número: 9	Nombre: Mostrar información OBP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: <p>El componente debe permitir al usuario mostrar la información guardada del cálculo estadístico.</p>	
Observaciones: Esta historia de usuario se ejecutará luego de haber guardado la información del cálculo estadístico y a su vez después de haber presionado el botón de "Actualizar".	
Interfaz 	

Tabla 11: Historia de usuario 10 "Limpiar información OBP"

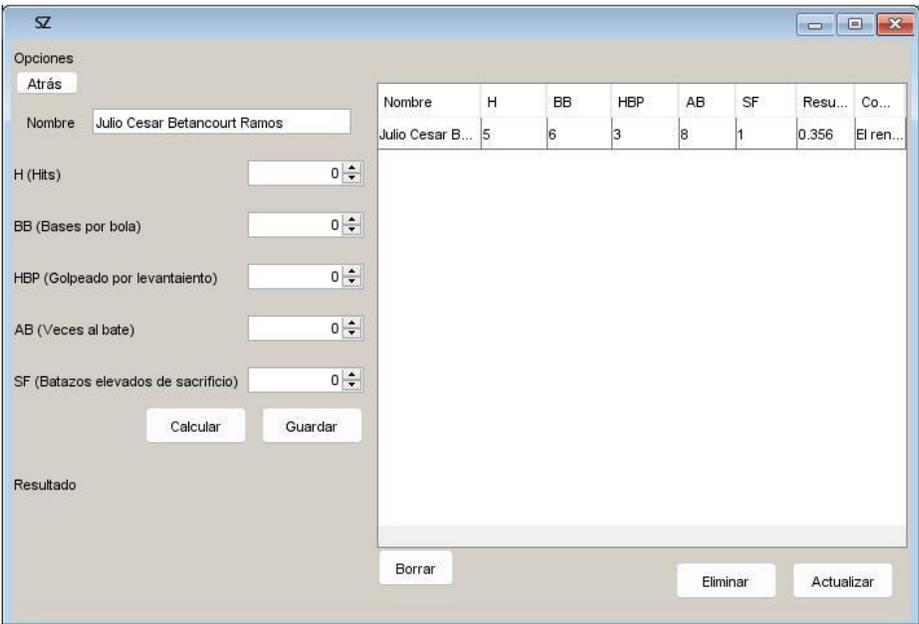
Historia de usuario																	
Número: 10	Nombre: Limpiar información OBP																
Usuario: usuario																	
Prioridad: Alta	Riesgo en desarrollo: Medio																
Puntos estimados: 0.1	Iteración: 1																
Programador a cargo: Julio Cesar Betancourt Ramos																	
<p>Descripción:</p> <p>El componente debe permitir al usuario limpiar los campos editables para introducir nuevos datos.</p>																	
<p>Observaciones: Esta historia de usuario se ejecutará luego de haber introducido información del cálculo en los campos editables y a su vez después de haber presionado el botón con un ícono que indica la acción de "Borrar".</p>																	
<p>Interfaz</p>  <p>The screenshot shows a window titled 'SZ' with a form for calculating baseball statistics. The form includes a 'Nombre' field with the value 'Julio Cesar Betancourt Ramos'. Below it are five input fields for 'H (Hits)', 'BB (Bases por bola)', 'HBP (Golpeado por levantaiento)', 'AB (Veces al bate)', and 'SF (Batazos elevados de sacrificio)', each with a value of '0'. There are 'Calcular' and 'Guardar' buttons. To the right is a table with the following data:</p> <table border="1"> <thead> <tr> <th>Nombre</th> <th>H</th> <th>BB</th> <th>HBP</th> <th>AB</th> <th>SF</th> <th>Resu...</th> <th>Co...</th> </tr> </thead> <tbody> <tr> <td>Julio Cesar B...</td> <td>5</td> <td>6</td> <td>3</td> <td>8</td> <td>1</td> <td>0.356</td> <td>El ren...</td> </tr> </tbody> </table> <p>At the bottom of the window are 'Borrar', 'Eliminar', and 'Actualizar' buttons.</p>		Nombre	H	BB	HBP	AB	SF	Resu...	Co...	Julio Cesar B...	5	6	3	8	1	0.356	El ren...
Nombre	H	BB	HBP	AB	SF	Resu...	Co...										
Julio Cesar B...	5	6	3	8	1	0.356	El ren...										

Tabla 12: Historia de usuario 11 "Visualizar resultado OBP"

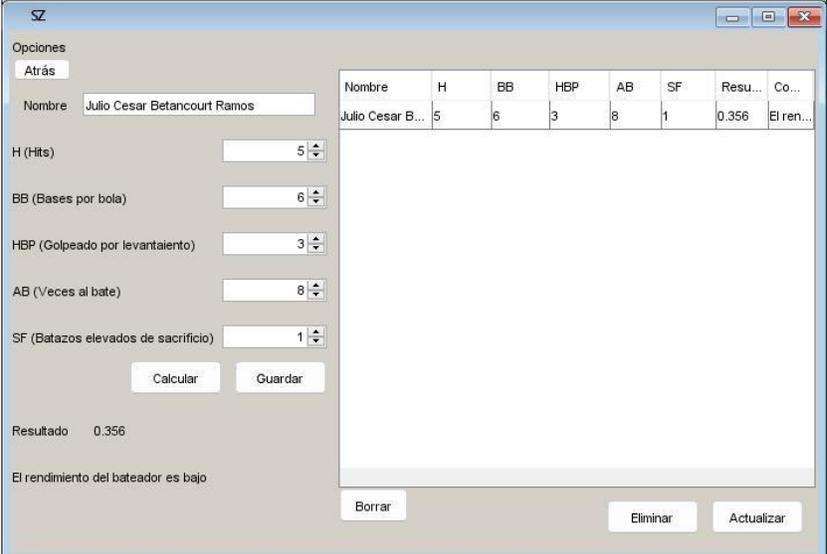
Historia de usuario	
Número: 11	Nombre: Visualizar resultado OBP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
<p>Descripción:</p> <p>El componente debe permitir al usuario visualizar el resultado del cálculo estadístico que se realizó.</p>	
<p>Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.</p>	
<p>Interfaz</p> 	

Tabla 13: Historia de usuario 12 "Visualizar comentario OBP"

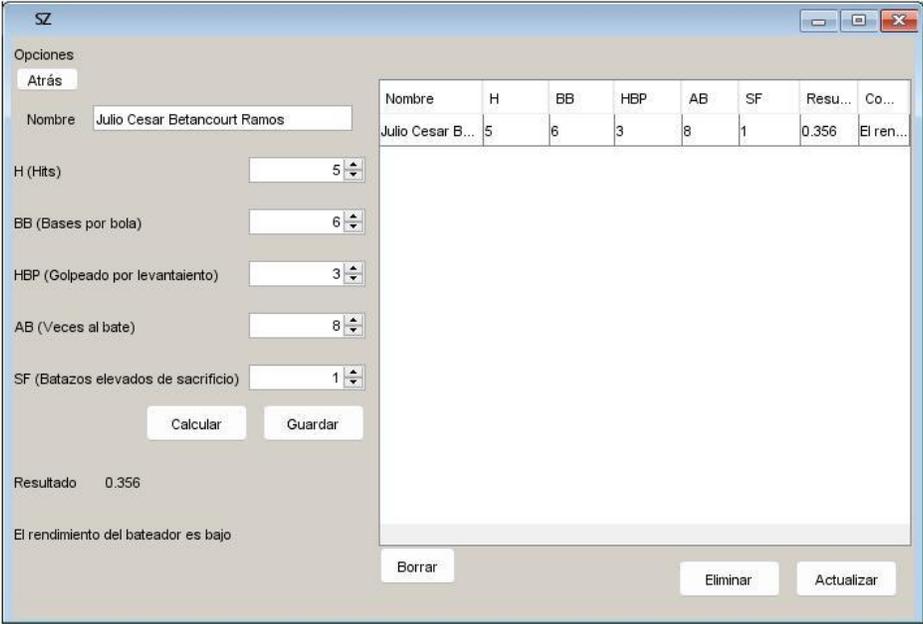
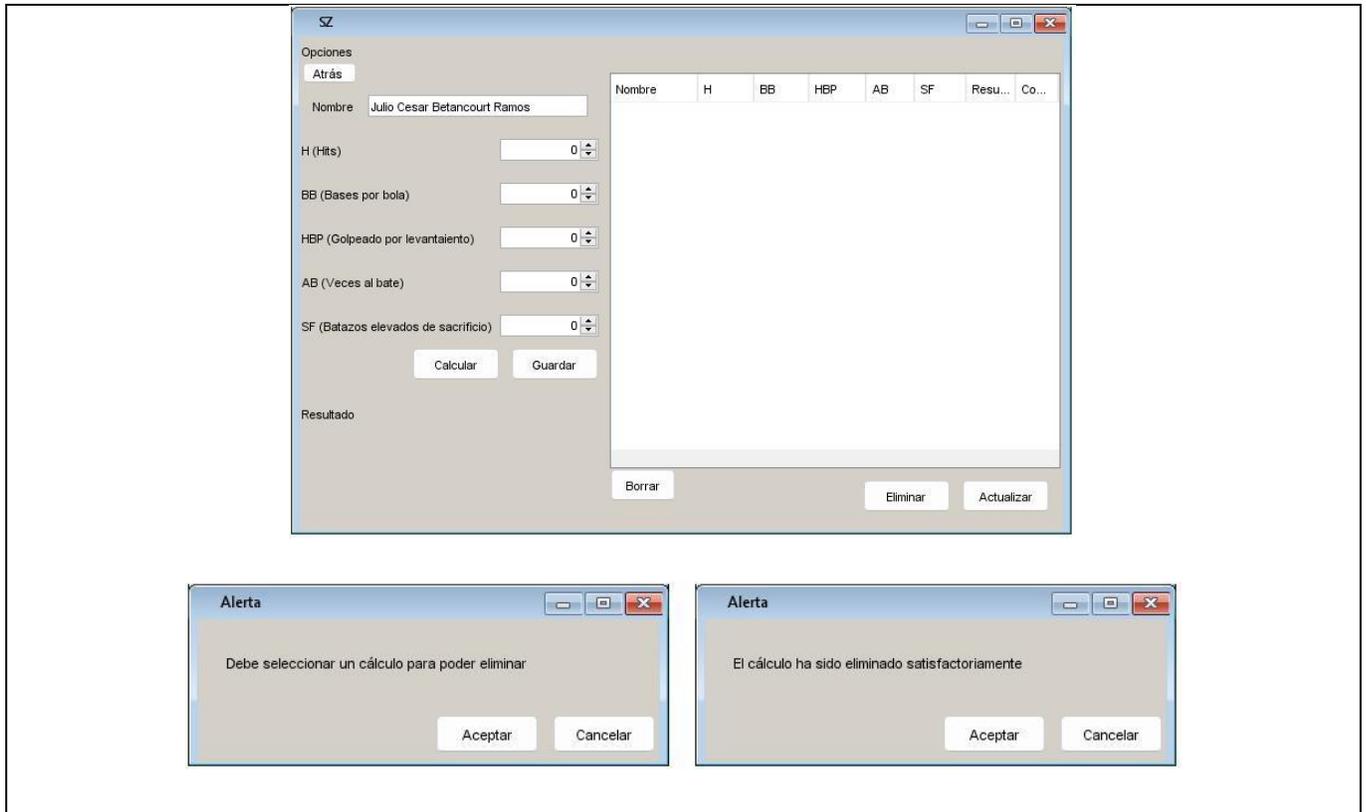
Historia de usuario																	
Número: 12	Nombre: Visualizar comentario OBP																
Usuario: usuario																	
Prioridad: Alta	Riesgo en desarrollo: Medio																
Puntos estimados: 0.1	Iteración: 1																
Programador a cargo: Julio Cesar Betancourt Ramos																	
<p>Descripción:</p> <p>El componente debe permitir al usuario visualizar el comentario del cálculo estadístico que se realizó a partir del resultado.</p>																	
<p>Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.</p>																	
<p>Interfaz</p>  <p>The screenshot shows a window titled 'SZ' with a 'Opciones' section containing several input fields with spinners: 'Nombre' (Julio Cesar Betancourt Ramos), 'H (Hits)' (5), 'BB (Bases por bola)' (6), 'HBP (Golpeado por levantaiento)' (3), 'AB (Veces al bate)' (8), and 'SF (Batazos elevados de sacrificio)' (1). Below these are 'Calcular' and 'Guardar' buttons. The 'Resultado' is 0.356, and a message states 'El rendimiento del bateador es bajo'. A table on the right shows the input data for 'Julio Cesar B...':</p> <table border="1"> <thead> <tr> <th>Nombre</th> <th>H</th> <th>BB</th> <th>HBP</th> <th>AB</th> <th>SF</th> <th>Resu...</th> <th>Co...</th> </tr> </thead> <tbody> <tr> <td>Julio Cesar B...</td> <td>5</td> <td>6</td> <td>3</td> <td>8</td> <td>1</td> <td>0.356</td> <td>El ren...</td> </tr> </tbody> </table> <p>At the bottom of the window are 'Borrar', 'Eliminar', and 'Actualizar' buttons.</p>		Nombre	H	BB	HBP	AB	SF	Resu...	Co...	Julio Cesar B...	5	6	3	8	1	0.356	El ren...
Nombre	H	BB	HBP	AB	SF	Resu...	Co...										
Julio Cesar B...	5	6	3	8	1	0.356	El ren...										

Tabla 14: Historia de usuario 13 "Eliminar OBP "

Historia de usuario	
Número: 13	Nombre: Eliminar OBP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.3	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
<p>Descripción:</p> <p>El componente debe permitir al usuario eliminar la información guardada.</p>	
<p>Observaciones: Esta historia de usuario se ejecutará luego de que se muestre y se visualice la información del cálculo estadístico y a su vez que se seleccione uno de los mismos y se presione el botón "Eliminar".</p> <p>En caso de que no se seleccione uno de los cálculos, el componente muestra el mensaje "Debe seleccionar un cálculo para poder eliminar".</p> <p>En caso de que todos los pasos para eliminar un cálculo se cumplan, el componente muestra el mensaje "El cálculo ha sido eliminado correctamente".</p>	
Interfaz	



2.4 Fase de Planificación

Durante esta fase se realiza una estimación del esfuerzo que costará implementar todas las historias de usuario, se parte del tiempo asignado a cada una en la fase de exploración. Una vez terminado, se procede a organizarlas en las iteraciones correspondientes, teniendo en cuenta la prioridad especificada por el cliente y el tiempo de desarrollo de cada una (Penadés & Letelier Torres, 2006).

2.4.1 Plan de iteraciones

Una iteración es una parte esencial del proyecto que cuando culmina el cliente obtiene un resultado parcial. Al concluir la última iteración, el mismo quedará completamente satisfecho, pues culmina la producción del proyecto. En la organización de las iteraciones se debe evitar extender más de un mes laboral; por lo general se proponen entre 20 y 30 días. La demora de un producto puede atentar con los objetivos que tenga el cliente con la aplicación. Una forma de evitar que esto ocurra se lleva a cabo mediante las pruebas de los usuarios finales, las correcciones a las que son sometidas la aplicación en las revisiones y así se podrá definir si el software va por un buen camino (Penadés & Letelier Torres,

2006). En la siguiente tabla se pueden observar las iteraciones que se determinaron necesarias para el desarrollo de la solución.

Tabla 15: Plan de iteraciones

Iteración	Historia de usuario	Duración en semanas
1	Registrar deportista	5
	Modificar deportista	
	Eliminar deportista	
	Listar deportista	
	Visualizar información OBP	
	Ingresar información OBP	
	Calcular datos OBP	
	Guardar información OBP	
	Mostrar información OBP	
	Limpiar campos OBP	
	Visualizar resultado OBP	
	Visualizar comentario OBP	
	Eliminar OBP	
	Visualizar información SLG	
	Ingresar información SLG	
	Calcular datos SLG	
Guardar información SLG		

	Mostrar información SLG	
	Limpiar campos SLG	
	Visualizar resultado SLG	
	Visualizar comentario SLG	
	Eliminar SLG	
2	Visualizar información OPS	5
	Ingresar información OPS	
	Calcular datos OPS	
	Guardar información OPS	
	Mostrar información OPS	
	Limpiar campos OPS	
	Visualizar resultado OPS	
	Visualizar comentario OPS	
	Eliminar OPS	
	Visualizar información BABIP	
	Ingresar información BABIP	
	Calcular datos BABIP	
	Guardar información BABIP	
	Mostrar información BABIP	
	Limpiar campos BABIP	
	Visualizar resultado BABIP	

	Visualizar comentario BABIP	
	Eliminar BABIP	
	Visualizar información BB%	
	Ingresar información BB%	
	Calcular datos BB%	
	Guardar información BB%	
	Mostrar información BB%	
	Limpiar campos BB%	
	Visualizar resultado BB%	
	Visualizar comentario BB%	
	Eliminar BB%	
3	Administrar ISO	3
	Administrar RC	
	Visualizar información ISO	
	Ingresar información ISO	
	Calcular datos ISO	
	Guardar información ISO	
	Mostrar información ISO	
	Limpiar campos ISO	
	Visualizar resultado ISO	
	Visualizar comentario ISO	

	Eliminar ISO	
	Visualizar información RC	
	Ingresar información RC	
	Calcular datos RC	
	Guardar información RC	
	Mostrar información RC	
	Limpiar campos RC	
	Visualizar resultado RC	
	Visualizar comentario RC	
	Eliminar RC	
TOTAL		13

2.4.2 Plan de entrega

El plan de entrega es el compromiso final del equipo de desarrollo con el cliente, pues en él se define cuando será entregado el producto. Representa un factor importante para el proyecto, pues la demora en la entrega del proyecto trae consigo insatisfacción con el cliente. En el plan de entrega se lleva a cabo la estimación del tiempo necesario para entregar al cliente las versiones del producto a medida que se cumplan los requisitos del software (Pressman 2013).

Tabla 16: Plan de entrega

Iteraciones	Fecha de inicio	Fecha fin
1	29 de agosto de 2022	3 de octubre de 2022
2	4 de octubre de 2022	8 de noviembre de 2022
3	9 de noviembre de 2022	30 de noviembre de 2022

2.4.3 Tarjetas CRC

El principal objetivo de las tarjetas CRC es dejar el pensamiento procedimental para enfocarse al orientado a objetos. Cada tarjeta representa una clase con su nombre en la parte superior, en la sección inferior izquierda están descritas las responsabilidades y a la derecha las clases que le sirven de soporte (Penadés & Letelier Torres, 2006). A continuación, se presentan las tarjetas que fueron necesarias crear para el desarrollo de la solución:

Tabla 17: Tarjeta CRC "Registrar Deportista"

Tarjeta CRC	
Clase: Registrar Deportista	
Responsabilidades	Clases relacionadas
<p>+void</p> <p><i>RegistrarDeportista::on_pushButton_2_clicked()</i> : Este método se utiliza para cerrar la ventana "Registrar Deportista".</p> <p>+ void</p> <p><i>RegistrarDeportista::on_pushButton_clicked()</i> : Contiene el código para guardar la información introducida en la base de datos.</p>	<ul style="list-style-type: none"> • MainWindow.cpp • registrardeportista.h

Tabla 18: Tarjeta CRC "Modificar deportista"

Tarjeta CRC	
Clase: Modificar Deportista	
Responsabilidades	Clases relacionadas
<p>+ void</p> <p><i>Modificardeportista::on_pushButton_3_clicked()</i> :</p>	<ul style="list-style-type: none"> • MainWindow.cpp • modificardeportista.h

<p>Contiene el código para actualizar y guardar la información introducida en la base de datos..</p> <p>+ <i>void</i></p> <p><i>Modificardeportista::on_pushButton_4_clicked():</i></p> <p>Este método se utiliza para cerrar la ventana “Registrar Deportista”.</p>	
---	--

Tabla 19: Tarjeta CRC "Main Window"

Tarjeta CRC	
Clase: Main Window	
Responsabilidades	Clases relacionadas
<p>+ <i>MainWindow::MainWindow(QWidget *parent):</i></p> <p>Método principal donde reside el constructor de la clase, se realiza la conexión a la base de datos y se crean las tablas de la misma.</p> <p>+ <i>void</i></p> <p><i>MainWindow::on_pushButton_27_clicked():</i></p> <p>Método que se utiliza para eliminar a un deportista que esté seleccionado en la tabla “Estadísticos”</p> <p>+ <i>void</i></p> <p><i>MainWindow::on_pushButtonOBP_clicked():</i></p> <p>Método del botón OBP que se utiliza para cambiar de la ventana principal a la ventana “OBP”.</p> <p>+ <i>void</i></p> <p><i>MainWindow::on_pushButton_10_clicked():</i></p>	<ul style="list-style-type: none"> • Registrar Deportista • Modificar Deportista • mainwindow.h

<p>Método del botón atrás que se utiliza para cambiar de la ventana OBP a la ventana principal.</p> <p>+ <i>void MainWindow::on_pushButton_clicked():</i> Método del botón “Calcular” que se utiliza para realizar el cálculo de OBP.</p> <p>+ <i>void MainWindow::mostrarTabla():</i> Método que se utiliza para mostrar la información guardada en base de datos en la tabla OBP.</p> <p>+ <i>void MainWindow::on_pushButton_2_clicked():</i> Método del botón “Actualizar” que se utiliza para actualizar la información que muestra la tabla OBP.</p> <p>+ <i>void MainWindow::on_pushButton_4_clicked():</i> Método del botón “Guardar” que se utiliza para guardar la información que introduce el usuario en la base de datos y la muestra en la tabla OBP.</p> <p>+ <i>void MainWindow::on_pushButton_3_clicked():</i> Método que se utiliza para limpiar los campos escritos por el usuario.</p>	
---	--

2.5 Estilo arquitectónico

Un estilo arquitectónico es un conjunto de decisiones que son aplicables en un contexto de desarrollo específico, restringen las decisiones de diseño de un sistema a ese contexto y plantean como objetivo ciertas cualidades para el sistema resultante (Ecarri, 2016).

Para el diseño e implementación del sistema se utilizará el estilo Llamada y Retorno. Este tipo de arquitectura permite que los usuarios puedan acceder a los mismos datos al establecer conexión con

el servidor, esta brinda un acceso transparente a las aplicaciones que se ejecutan en él, controlando el acceso a los recursos.

2.5.1 Patrón arquitectónico.

Con el objetivo de establecer una estructura para todos los componentes del sistema se selecciona el patrón arquitectónico en capas. El uso de este patrón permite un mejor soporte para futuras actualizaciones del sistema, dado que está destinado a ser auto-independiente; los componentes están interconectados pero no dependen unos de otros ya que cada capa tiene un papel y una responsabilidad específicos en la aplicación siendo esta una de las características más importantes de este patrón porque los componentes dentro de una capa específica se ocupan sólo de la lógica que pertenece a esa capa (Novoselseva, 2020).

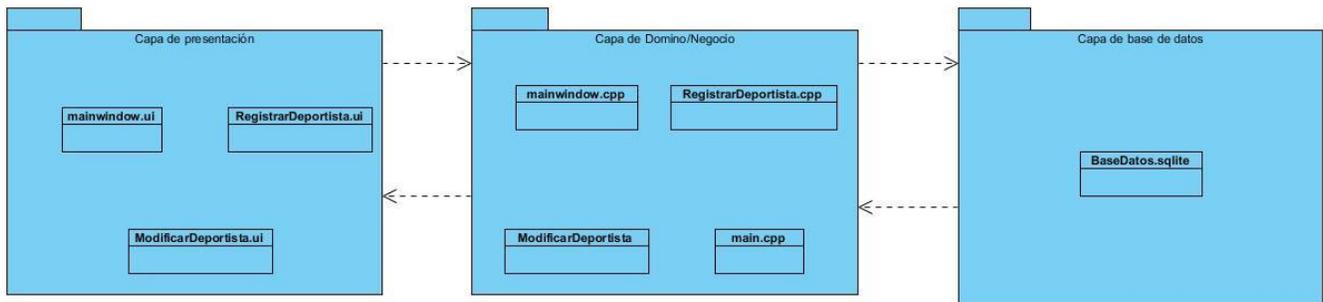


Ilustración 5: "Arquitectura en Capas"

2.6 Diagrama de clase del diseño

A partir de la arquitectura seleccionada se diseña el diagrama de clases del diseño para describir de forma visual las clases utilizadas dentro del sistema, las funcionalidades y las relaciones entre ellas. Permitiendo para la posterior implementación la vista del diseño y una interpretación de las funcionalidades del sistema propuesto.

La ilustración 6 muestra el diagrama de clases del diseño.

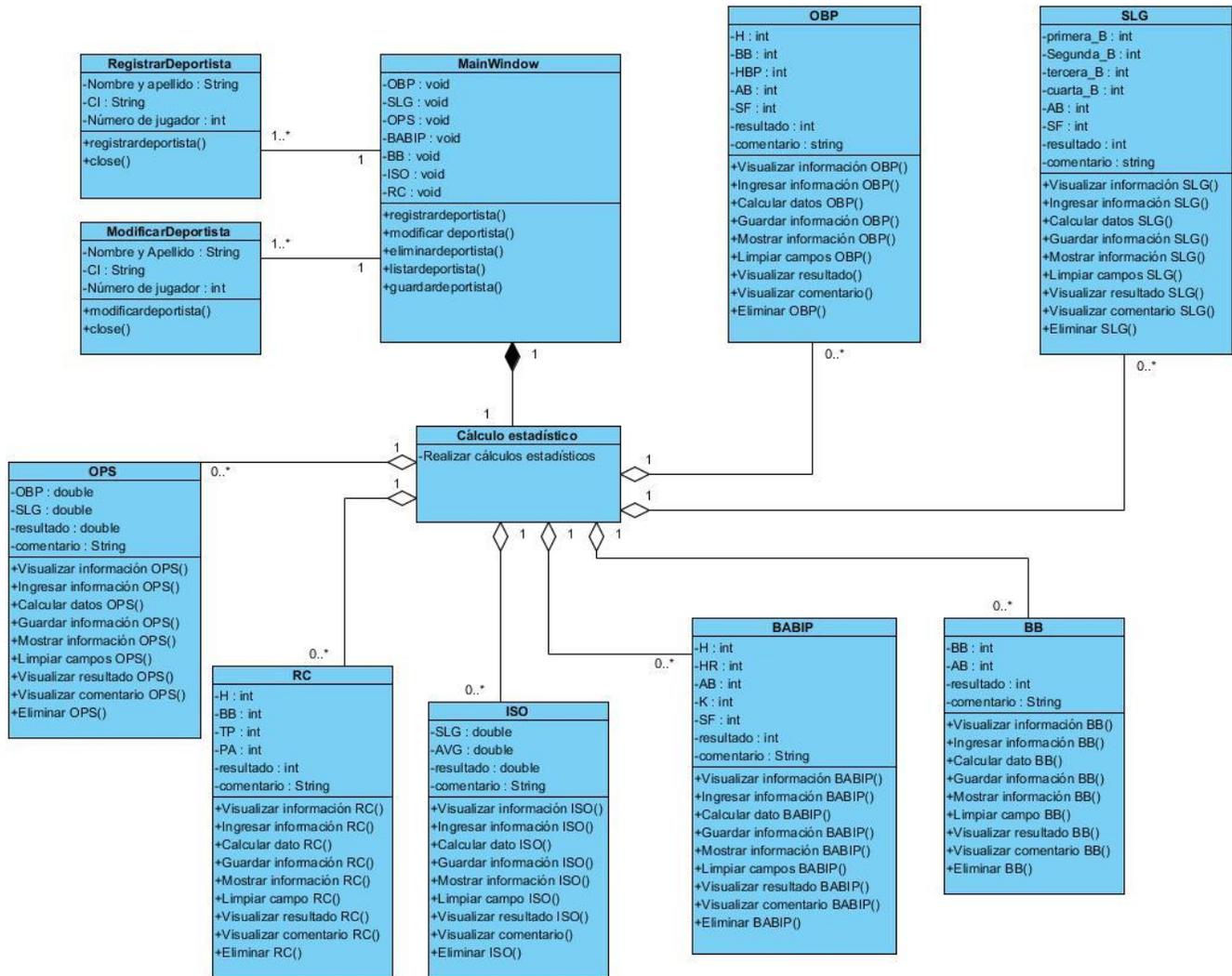


Ilustración 6: Diagrama de clases del diseño

2.7 Modelo de datos

Para la determinación de una estructura lógica de una base de datos y el modo de almacenar, organizar y manipular los datos se emplea el modelo de datos. Con el objetivo de definir las clases persistentes se identifiquen los conceptos, en el dominio del negocio, que persisten en el tiempo.

A continuación, se muestra el diagrama entidad-relación, el cual posee un conjunto de tablas correspondientes a cada componente:

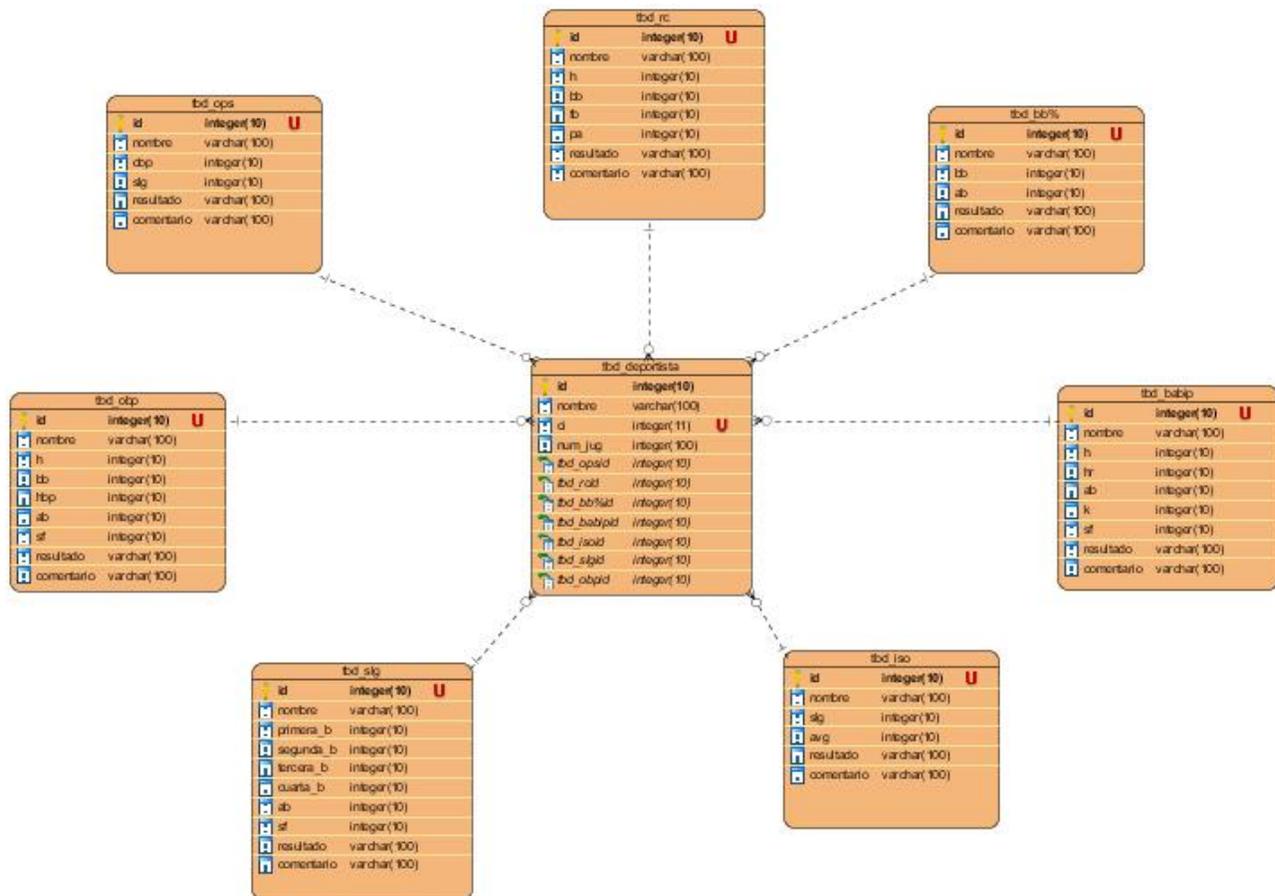


Ilustración 7: "Modelado de datos"

Descripción de las tablas

1. **tbd_deportista:** Tabla principal del modelo de datos en la que se registran las actividades con sus características.
2. **tbd_OBP:** Tabla nomencladora de los cálculos de OBP a realizar.
3. **tbd_SLG:** Tabla nomencladora de los cálculos de SLG a realizar.
4. **tbd_OPS:** Tabla nomencladora de los cálculos de OPS a realizar.
5. **tbd_BABIP:** Tabla nomencladora de los cálculos de BABIP a realizar.
6. **tbd_BB%:** Tabla nomencladora de los cálculos de BB% a realizar.
7. **tbd_ISO:** Tabla nomencladora de los cálculos de ISO a realizar.
8. **tbd_RC:** Tabla nomencladora de los cálculos de RC a realizar.

2.8 Patrones de diseño

Para un mejor entendimiento del contexto en el que se desarrolla la aplicación se utilizan los patrones de diseño, puesto que, son plantillas para soluciones a problemas comunes en el desarrollo de software que se pueden usar en distintos momentos. (iitalomorales, 2005)

Patrones Generales de Software de Asignación de Responsabilidad (GRASP, por sus siglas en inglés): Los patrones GRASP, más que patrones propiamente dichos, son una serie de “buenas prácticas” de aplicación recomendable en el diseño de software. Entre ellos, los patrones Experto, creador, bajo acoplamiento y Alta cohesión pueden ser tratados en un curso de lógica de programación porque guardan directa relación con la creación. (Tabares, 2019)

- Patrón Creador: El empleo de este patrón asigna a determinado objeto la responsabilidad de crear instancias de otros objetos. El uso de este patrón se evidencia utilizando los repositorios asociados a cada entidad por medio de la clase `mainwindow.h`
- Bajo Acoplamiento: El acoplamiento mide el grado en que una clase está conectada, tiene conocimiento o depende de otra. En el desarrollo de la aplicación se evidencia al realizar modificaciones en clases como `mainwindow.cpp`, `registrardeportista.cpp` y `modificardeportista.cpp`, que las mismas no están fuertemente relacionadas, por tanto, la repercusión sobre las demás será mínima.
- Alta Cohesión: Verifica que la información que almacena una clase debe ser coherente y debe estar, en la medida de lo posible, relacionada con la clase. En la solución propuesta las clases clientes y servidoras, se encargan de realizar una función específica, para evitar la saturación. Además, la asignación de responsabilidades garantiza la alta cohesión en las clases existentes.
- Experto: Este patrón implica que cada objeto es responsable de almacenar su propia información, no la de los demás. Este patrón se ve evidenciado principalmente en la clase `mainwindow.cpp`

Patrones Pandilla de cuatro (GOF, por sus siglas en inglés)

- Observer (Observador): Define una dependencia de uno a muchos, de forma que cuando el objeto escuchado cambia su estado, el patrón se encarga de que todos los oyentes sean notificados de esos cambios. El patrón observador garantiza la comunicación entre los componentes. El presente patrón es utilizado en el sistema propuesto en los mensajes. Esto se evidencia en el manejo de la aplicación ya que muestra mensajes de alerta como “El campo “Nombre” es obligatorio” cuando se va a realizar algún cálculo y no se ha seleccionado previamente un deportista.

2.9 Conclusiones parciales

- Los requisitos del sistema y la arquitectura propuesta permitieron obtener un diseño robusto de la aplicación.
- Las definiciones de historias de usuario permitieron tener la base para la implementación de la aplicación.
- Las definiciones de las tarjetas CRC aportaron en la identificación de las responsabilidades de las clases y las colaboraciones de las mismas.
- Se aplicaron los patrones necesarios para lograr un diseño flexible y eficiente.

Capítulo 3: Implementación y pruebas del componente de obtención de datos estadísticos para Strike Zone

En el presente capítulo se abordan los principales aspectos relacionados con la implementación de la solución propuesta. Se definen los estándares de codificación para lograr una mejor comprensión y estandarización del código resultante. Se realiza la verificación del correcto funcionamiento del software mediante la aplicación de las pruebas correspondientes y se presentan los resultados de dichas pruebas.

3.1 Estándares de codificación

Con el objetivo de que el trabajo de los desarrolladores sea más fácil para saber estructurar su código y pueda ser revisado por otras personas se utilizan los estándares de codificación. A continuación, se muestran los estándares empleados durante el desarrollo. (Ichi pro, 2022)

Tabla 20: Estándares de codificación

Descripción	Ejemplo
Definición de Objetos, Clases, funcionalidades y atributos	
Todos los nombres de las clases implementadas comenzarán con la letra mayúscula. En caso de poseer un nombre compuesto se escribirán de acuerdo a la normativa CamelCase-UpperCamelCase	<code>MainWindow::MainWindow(QWidget *parent)</code> { /*Cuerpo de la clase*/ }
Siempre se declara para todas las clases implementadas su respectivo destructor de clase	<code>MainWindow::~~MainWindow()</code>
La declaración de las funciones o métodos siempre comenzarán en letra inicial minúscula. En caso de ser un nombre compuesto se registrará por la normativa CamelCase-lowerCamelCase	<code>void MainWindow::mostrarTabla()</code> { /*Cuerpo del método*/ }
Los atributos siempre estarán escritos con letra minúscula. En caso de ser un nombre compuesto	<code>QSqlQuery crear;</code>

se regirá por la normativa CamelCase-lowerCamelCase	<code>MainWindow *mainw = new MainWindow();</code>
Definición de los parámetros dentro de las funciones y constructores de clases	
Los nombres de los identificadores de los parámetros en las funciones deben estar escritos con minúscula separados a un espacio cada uno u en vaso de ser compuesto utilizar la normativa CamelCase-lowerCamelCase	<code>QBarSet *MainWindow::createSerie(QString name, QList<qreal> resultado)</code>
Los identificadores de los parámetros dentro de los constructores de las clases deben estar escritos con minúscula separados a un espacio cada uno y en caso de ser compuesto utilizar la normativa CamelCase-lowerCamelCase	<code>QMainWindow(parent), ui(new Ui::MainWindow)</code>
Definición de expresiones	
Para una mejor comprensión en la lectura y legibilidad del código los operadores binarios exceptuando los punteros, función de llamado de miembros, escritura de un arreglo y paréntesis de una función se escribirán con un espacio entre ellos.	<code>double a = ui->spinBoxH->value() + ui->spinBoxBB->value(); double b = ui->spinBoxAB->value() + ui->spinBoxBB->value(); double c = a/b;</code>
Definición de estructuras de control y bucles	
Las estructuras de control y los bucles estarán definidos de igual manera en ambos casos siguiendo el estándar determinado por el framework de Qt	Para las estructuras if, else y else if: <code>if (condición) { } else if (condición) { } else { }</code>
	Para los bucles while, for, do while y otros:

	for (condición) { }
Comentarios en el código según c++	
Comentarios pequeños	/* comentario sencillo*/
Otros comentarios	/* *comentario */
Comentario de versión, descripción de clase y otras características de la clase o paquete	/* *comentario amplio* */

3.2 Tareas de Ingeniería

En la tabla 10 se muestran de manera general las correspondientes tareas de ingeniería y en las tablas 11-14 se detallan algunas una de ellas. (Ver anexo A.3).

Tabla 21: Tareas de Ingeniería

Número de Tarea	Número de Historia	Nombre de la Tarea
1	1	Implementar Registrar deportista
2	1	Implementar Modificar deportista
3	1	Implementar Eliminar deportista
4	1	Implementar Listar deportista
5	2	Implementar Visualizar información OBP

6	2	Implementar Ingresar información OBP
7	2	Implementar Calcular datos OBP
8	2	Implementar Guardar información OBP
9	2	Implementar Mostrar Información OBP
10	2	Implementar Limpiar campos OBP
11	2	Implementar Visualizar resultado OBP
12	2	Implementar Visualizar comentario OBP
13	2	Implementar Eliminar OBP
14	3	Implementar Visualizar información SLG
15	3	Implementar Ingresar información SLG
16	3	Implementar Calcular datos SLG
17	3	Implementar Guardar información SLG
18	3	Implementar Mostrar Información SLG
19	3	Implementar Limpiar campos SLG
20	3	Implementar Visualizar resultado SLG
21	3	Implementar Visualizar comentario SLG
22	3	Implementar Eliminar SLG
23	4	Implementar Visualizar información OPS
24	4	Implementar Ingresar información OPS
25	4	Implementar Calcular datos OPS
26	4	Implementar Guardar información OPS

27	4	Implementar Mostrar Información OPS
28	4	Implementar Limpiar campos OPS
29	4	Implementar Visualizar resultado OPS
30	4	Implementar Visualizar comentario OPS
31	4	Implementar Eliminar OPS
32	5	Implementar Visualizar información BABIP
33	5	Implementar Ingresar información BABIP
34	5	Implementar Calcular datos BABIP
35	5	Implementar Guardar información BABIP
36	5	Implementar Mostrar Información BABIP
37	5	Implementar Limpiar campos BABIP
38	5	Implementar Visualizar resultado BABIP
39	5	Implementar Visualizar comentario BABIP
40	5	Implementar Eliminar BABIP
41	6	Implementar Visualizar información BB%
42	6	Implementar Ingresar información BB%
43	6	Implementar Calcular datos BB%
44	6	Implementar Guardar información BB%
45	6	Implementar Mostrar Información BB%
46	6	Implementar Limpiar campos BB%
47	6	Implementar Visualizar resultado BB%

48	6	Implementar Visualizar comentario BB%
49	6	Implementar Eliminar BB%
50	7	Implementar Visualizar información ISO
51	7	Implementar Ingresar información ISO
52	7	Implementar Calcular datos ISO
53	7	Implementar Guardar información ISO
54	7	Implementar Mostrar Información ISO
55	7	Implementar Limpiar campos ISO
56	7	Implementar Visualizar resultado ISO
57	7	Implementar Visualizar comentario ISO
58	7	Implementar Eliminar ISO
59	8	Implementar Visualizar información RC
60	8	Implementar Ingresar información RC
61	8	Implementar Calcular datos RC
62	8	Implementar Guardar información RC
63	8	Implementar Mostrar Información RC
64	8	Implementar Limpiar campos RC
65	8	Implementar Visualizar resultado RC
66	8	Implementar Visualizar comentario RC
67	8	Implementar Eliminar RC

3.2.1 Descripción Tareas de Ingeniería

Iteración 1

Tabla 22: Tarea de ingeniería 1 "Implementar Registrar deportista"

Tarea de Ingeniería	
Número de tarea: 1	Número de historia: 1
Nombre de la tarea: Implementar Registrar deportista	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad que permita registrar un deportista en el sistema	

Tabla 23: Tarea de ingeniería 2 "Implementar Modificar deportista"

Tarea de Ingeniería	
Número de tarea: 2	Número de historia: 1
Nombre de la tarea: Implementar Modificar deportista	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita modificar, ya sea el nombre, carnet de identidad o número de jugador, de algún deportista ya registrado en el sistema.	

Tabla 24: Tarea de ingeniería 3 "Implementar Eliminar deportista"

Tarea de Ingeniería	
Número de tarea: 3	Número de historia: 1

Nombre de la tarea: Implementar Eliminar deportista	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita eliminar a algún deportista ya registrado en el sistema.	

Tabla 25: Tarea de ingeniería 4: "Implementar Listar deportista"

Tarea de Ingeniería	
Número de tarea: 4	Número de historia: 1
Nombre de la tarea: Implementar Listar deportista	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar en pantalla la lista de deportistas que ya están registrados en el sistema.	

3.3 Pruebas de software

Las pruebas de software son los procesos que permiten verificar y revelar la calidad de un producto software antes de su puesta en marcha, se puede decir que, básicamente, es una fase de desarrollo de software que consiste en probar las aplicaciones construidas (Somerville, 2005).

3.3.1 Pruebas Unitaria

Se aplican a un componente del software. Podemos considerar como componente, a una función, una clase, una librería, etc. Estas pruebas las ejecuta el desarrollador, cada vez que va probando fragmentos de código o scripts para ver si todo funciona como se desea. El objetivo de las pruebas unitarias es aislar cada parte del programa y mostrar que las partes individuales son correctas.

Proporcionan un contrato escrito, que el fragmento de código debe satisfacer. El método utilizado para realizar este tipo de prueba se denomina caja blanca (Somerville, 2005).

Método de caja blanca

Las pruebas de caja blanca intentan garantizar que:

- Se ejecutan al menos una vez todos los caminos independientes de cada módulo.
- Se utilizan las decisiones en su parte verdadera y en su parte falsa.
- Se ejecuten todos los bucles en sus límites.
- Se utilizan todas las estructuras de datos internas.

Para la realización de las pruebas unitarias, se le aplicó la técnica de prueba de camino básico a las unidades código que responden a funcionalidades críticas del software, lo cual permitió generar el grafo de flujo, calcular la Complejidad Ciclomática (CC) para determinar los caminos linealmente independientes y el número mínimo de escenarios de los casos de prueba para forzar la ejecución de cada camino del conjunto básico.

Luego en apoyo a las pruebas se usó el módulo QTest que ofrece el entorno de desarrollo QtCreator, para la automatización de las pruebas unitarias. Con él se probó cada módulo desarrollado, y gracias a la aplicación de la técnica de camino básico, en aquellas funcionalidades críticas, se pudieron automatizar pruebas para cada uno de los escenarios o caminos posibles, garantizando probar todo el código en cuestión. Entre los elementos de código que fueron probadas se encuentra el referente al método `on_pushButton_clicked` de la clase `mainwindow.cpp`, que se encarga de controlar la lógica de calcular el OBP de un deportista, así como mostrar el resultado de dicho cálculo y hacer un comentario a partir del mismo.

Tabla 26: Cálculo de la complejidad ciclomática del método `on_pushButton_clicked` de la clase `mainwindow.cpp`

Método

```
void MainWindow::on_pushButton_clicked()
{
    int obp = ui->spinBoxH->value();
    int obp1= ui->spinBoxBB->value();
    int obp2= ui->spinBoxHBP->value();
    int obp3= ui->spinBoxAB->value();
    int obp4= ui->spinBoxSF->value();

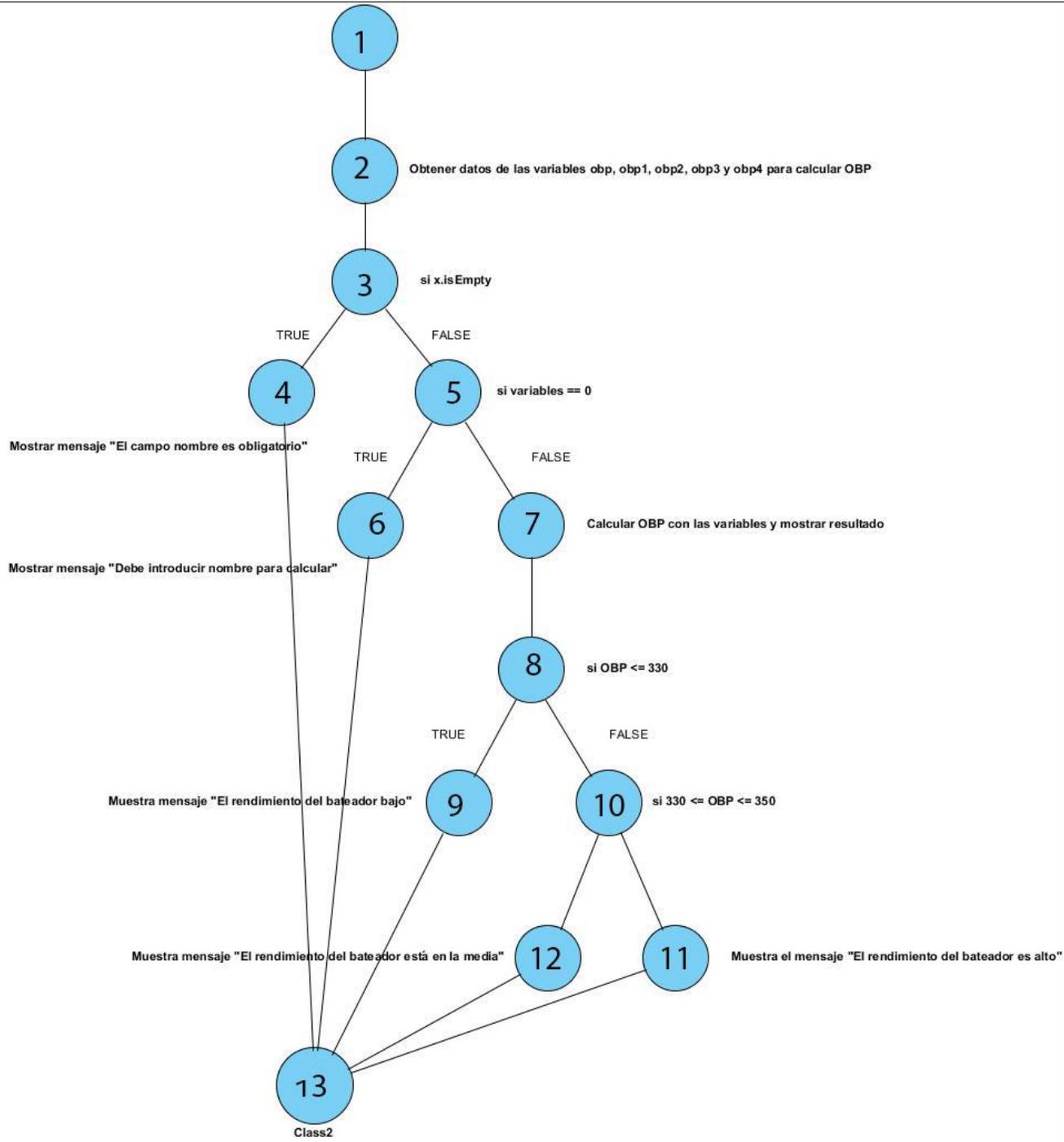
    QString x = ui->lineEditNombre->text();
    if(x.isEmpty()){

        QMessageBox::information(this,"Mensaje","El campo 'Nombre' es obligatorio");
    } else if (obp==0&obp1==0&obp2==0&obp3==0&obp4==0){
        QMessageBox::information(this,"Mensaje","Debe introducir números para poder calcular");
    }else {

        double a = ui->spinBoxH->value() + ui->spinBoxBB->value() + ui->spinBoxHBP->value();
        double b = ui->spinBoxAB->value() + ui->spinBoxBB->value() + ui->spinBoxHBP->value() + ui->spinBoxSF->value();
        double c = a/b;
        ui->resultados->setText(QString::number(c));

        if (c <= 330)
        {
            ui->comentario->setText("El rendimiento del bateador es bajo");
        }else if (330 <= c <= 350) {
            ui->comentario->setText("El rendimiento del bateador está en la media");
        }else {
            ui->comentario->setText("El rendimiento del bateador es alto");
        }
    }
}
```

Grafo resultante



Complejidad Ciclomática

$V(G) = \# \text{ de regiones}$

$V(G) = A - N + 2$

$V(G) = 16 - 13 + 2$

$V(G) = P + 1$

$V(G) = 4 + 1$

$V(G) = 5$	$V(G) = 5$	$V(G) = 5$
------------	------------	------------

Luego de la determinación de los nodos y flujos de control del código se obtuvo un grafo de flujo y se calculó la complejidad ciclomática del algoritmo.

Como resultado se obtuvo que la CC es igual a 5, lo que significa que existen cinco posibles caminos linealmente independientes y hay que diseñar un mínimo de 5 casos de prueba para el algoritmo. La tabla*** muestra los caminos existentes.

Tabla 27: Caminos del grafo de flujo

N.º	Camino
1	1,2,3,4,13
2	1,2,3,5,6,13
3	1,2,3,5,7,8,9,13
4	1,2,3,5,7,8,10,12,13
5	1,2,3,5,7,8,10,11,13

Los casos de prueba para las pruebas de caja de caja blanca por la técnica de camino básico se ejecutan por cada camino independiente que se determine en un algoritmo específico. A continuación, se muestra el caso de prueba para el camino básico independiente 5 del algoritmo.

Tabla 28: Caso de prueba para el camino básico 5

Proceso:
Calcular OBP.
Casos de prueba:

Calcular datos OBP
Camino independiente:
1,2,3,5,7,8,10,11,13
Entradas:
<ul style="list-style-type: none"> • H: 54, BB: 23, HBP: 15, AB: 65, SF: 8
Resultados esperados
<ul style="list-style-type: none"> • Realiza el cálculo correctamente • Muestre el resultado del cálculo • Muestre el comentario referente a ese resultado • Se muestre en la tabla el cálculo realizado
Condiciones de ejecución:
<ul style="list-style-type: none"> • Que se haya seleccionado previamente al deportista

Con la realización de los casos de prueba diseñados se probó la ejecución de cada sentencia del código al menos una vez, teniendo en cuenta todas las condiciones lógicas en sus variantes verdaderas y falsas. La obtención de la CC de valor 5 del método post ejemplificado, permitió determinar que existen 5 caminos linealmente independientes, suficientes para probar el código al menos una vez.

Los resultados del método de caja blanca fueron satisfactorios. Se automatizaron un total de 72 casos de prueba con el uso de la biblioteca QTest, de los cuales a 7 se le aplicó la técnica del camino básico, permitiendo que su automatización garantice probar todos los caminos con un mínimo de escenarios diseñados, y no obteniendo errores.

3.3.2 Pruebas de aceptación

Para la revisión final de las especificaciones del diseño y de la implementación se realizaron las pruebas de aceptación con el objetivo de determinar diferentes tipos de errores y resolverlos (Silva).

A continuación, se muestran las pruebas de aceptación realizadas a la primera iteración, las otras se encuentran ubicadas en los anexos (Ver anexo A.4)

Tabla 29: Caso de prueba 1 "Gestionar deportista"

Caso de prueba de aceptación	
Código: HU1_P1	Historia de Usuario: 1
Nombre: Gestionar Deportista	
Descripción: Trabaja con las funcionalidades principales para gestionar un deportista	
Condiciones de Ejecución: No presenta	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar las ventanas de registrar o modificar. 2. Llena los campos de texto de las ventanas y le da a los botones agregar o modificar. 3. Se muestra la pantalla principal con la información del deportista luego de tocar el botón modificar. 4. Selecciona un deportista de la tabla y toca el botón eliminar. 5. Se muestra la tabla sin el usuario seleccionado. 	
Resultado: No Satisfactorio	

Tabla 30: Caso de prueba 2: "Registrar deportista"

Caso de prueba de aceptación	
Código: HU1_P2	Historia de Usuario: 1
Nombre: Registrar deportista	
Descripción: Trabaja con las funcionalidades principales para registrar un deportista	
Condiciones de Ejecución: No presenta	

<p>Pasos de Ejecución:</p> <ol style="list-style-type: none"> 1. Debe seleccionar las ventanas de registrar. 2. Llena los campos de texto de la ventana y le da a el botón agregar. 3. Se muestra la pantalla principal con la información del deportista luego de tocar el botón actualizar.
<p>Resultado: Satisfactorio</p>

Tabla 31: Caso de prueba 3 "Modificar deportista"

Caso de prueba de aceptación	
Código: HU1_P3	Historia de Usuario: 1
Nombre: Modificar deportista	
Descripción: Trabaja con las funcionalidades principales para Modificar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
<p>Pasos de Ejecución:</p> <ol style="list-style-type: none"> 1. Debe seleccionar las ventanas de modificar. 2. Llena los campos de texto de la ventana y le da a el botón modificar. 3. Se muestra la pantalla principal con la información del deportista luego de tocar el botón actualizar. 	
Resultado: No Satisfactorio	

Tabla 32: Caso de prueba 4 "Listar deportista"

Caso de prueba de aceptación

Código: HU1_P4	Historia de Usuario: 1
Nombre: Listar deportista	
Descripción: Trabaja con las funcionalidades principales para Listar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se muestra la pantalla principal. 2. Toca el botón Actualizar. 3. Se muestra la pantalla principal con la información del deportista. 	
Resultado: Satisfactorio	

Tabla 33: Caso de prueba 5 "Guardar deportista"

Caso de prueba de aceptación	
Código: HU1_P5	Historia de Usuario: 1
Nombre: Guardar deportista	
Descripción: Trabaja con las funcionalidades principales para guardar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se muestra la pantalla principal de modificar o registrar deportista. 2. Llena los campos vacíos. 3. Toca el botón aceptar 4. Se muestra la pantalla principal con la información del deportista y guardada en la base de datos. 	
Resultado: Satisfactorio	

Tabla 34: Caso de prueba 6 "Eliminar deportista"

Caso de prueba de aceptación	
Código: HU1_P6	Historia de Usuario: 1
Nombre: Eliminar deportista	
Descripción: Trabaja con las funcionalidades principales para eliminar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se muestra la pantalla principal. 2. Selecciona al deportista. 3. Toca el botón "Eliminar" 4. Se muestra la pantalla principal con el deportista eliminado. 	
Resultado: Satisfactorio	

Tabla 35: Caso de prueba 6 "Administrar OBP"

Caso de prueba de aceptación	
Código: HU2_P7	Nº de Historia de Usuario: 2
Historia de usuario: Administrar OBP	
Descripción: Trabaja con las funcionalidades principales para realizar el cálculo del OBP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar a un deportista. 2. Toca el botón "OBP" que abre la ventana para el cálculo del mismo. 3. Se muestra la ventana OBP. 4. El usuario introduce los datos y realiza el cálculo. 	

<ol style="list-style-type: none"> 5. Se muestra en pantalla el resultado y el comentario de dicho cálculo. 6. El usuario toca el botón guardar 7. Se muestra en la tabla los cálculos realizados.
Resultado: Satisfactorio

Tabla 36: Caso de prueba 7 "Visualizar OBP"

Caso de prueba de aceptación	
Código: HU2_P8	Nº de Historia de Usuario: 2
Historia de usuario: Visualizar OBP	
Descripción: Muestra en pantalla la ventana OBP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar a un deportista. 2. Toca el botón "OBP" que abre la ventana para el cálculo del mismo. 3. Se muestra la ventana OBP. 	
Resultado: Satisfactorio	

Tabla 37: Caso de prueba 8 "Ingresar información OBP"

Caso de prueba de aceptación	
Código: HU2_P9	Nº de Historia de Usuario: 2
Historia de usuario: Ingresar Información OBP	
Descripción: Permite ingresar la información necesaria para realizar el cálculo del OBP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	

Pasos de Ejecución: 1. Debe llenar los campos vacíos editables en la ventana.
Resultado: Satisfactorio

Tabla 38: Caso de prueba 9 "Calcular datos OBP"

Caso de prueba de aceptación	
Código: HU2_P10	Nº de Historia de Usuario: 2
Historia de usuario: Calcular datos OBP	
Descripción: Permite realizar cálculos pertinentes para obtener el OBP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo OBP.	
Resultado: Satisfactorio	

Tabla 39: Caso de prueba 10 "Guardar información"

Caso de prueba de aceptación	
Código: HU2_P11	Nº de Historia de Usuario: 2
Historia de usuario: Guardar información OBP	
Descripción: Permite guardar la información de OBP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	

<p>Pasos de Ejecución:</p> <ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón “Calcular” para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo OBP.
<p>Resultado: Satisfactorio</p>

Tabla 40: Caso de prueba 11"Mostrar información OBP"

Caso de prueba de aceptación	
Código: HU2_P12	Nº de Historia de Usuario: 2
Historia de usuario: Mostrar información OBP	
Descripción: Permite mostrar en pantalla la información de OBP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
<p>Pasos de Ejecución:</p> <ol style="list-style-type: none"> 1. Muestra la información en cuanto se abre la ventana OBP. 2. Toca el botón “Actualizar” para mostrar la información que no se ha mostrado aún. 	
Resultado: Satisfactorio	

Tabla 41: Caso de prueba 12 "Limpiar campos OBP"

Caso de prueba de aceptación	
Código: HU2_P13	Nº de Historia de Usuario: 2
Historia de usuario: Limpiar campos OBP	

Descripción: Permite limpiar los datos que introdujo el usuario al sistema.
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Limpiar" y pone los campos en blanco.
Resultado: Satisfactorio

Tabla 42: Caso de prueba 13 "Visualizar resultado OBP"

Caso de prueba de aceptación	
Código: HU2_P14	Nº de Historia de Usuario: 2
Historia de usuario: Visualizar resultado OBP	
Descripción: Permite visualizar el resultado del cálculo realizado.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 	
Resultado: Satisfactorio	

Tabla 43: Caso de prueba 14 "Visualizar comentario OBP"

Caso de prueba de aceptación

Código: HU2_P15	Nº de Historia de Usuario: 2
Historia de usuario: Visualizar comentario OBP	
Descripción: Permite visualizar el comentario según el resultado que da el cálculo.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 4. Muestra el comentario de dicho resultado. 	
Resultado: Satisfactorio	

Tabla 44: Caso de prueba 15 "Eliminar OBP"

Caso de prueba de aceptación	
Código: HU1_P16	Historia de Usuario: 2
Nombre: Eliminar OBP	
Descripción: Trabaja con las funcionalidades principales para eliminar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se muestra la ventana "OBP". 2. Selecciona al deportista. 3. Toca el botón "Eliminar" 4. Se muestra la ventana OBP con el deportista eliminado. 	
Resultado: Satisfactorio	

Tabla 45: Caso de prueba 15 "Administrar SLG"

Caso de prueba de aceptación	
Código: HU3_P17	Nº de Historia de Usuario: 3
Historia de usuario: Administrar SLG	
Descripción: Trabaja con las funcionalidades principales para realizar el cálculo del SLG	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar las ventanas de registrar o modificar. 2. Llena los campos de texto de las ventanas y le da a los botones agregar o modificar. 3. Se muestra la pantalla principal con la información del deportista luego de tocar el botón modificar. 4. Selecciona un deportista de la tabla y toca el botón eliminar. 5. Se muestra la tabla sin el usuario seleccionado. 	
Resultado: Satisfactorio	

Tabla 46: Caso de prueba 16 "Visualizar SLG"

Caso de prueba de aceptación	
Código: HU3_P18	Nº de Historia de Usuario: 3
Historia de usuario: Visualizar SLG	
Descripción: Muestra en pantalla la ventana SLG	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar a un deportista. 2. Toca el botón "SLG" que abre la ventana para el cálculo del mismo. 	

3. Se muestra la ventana SLG.
Resultado: Satisfactorio

Tabla 47: Caso de prueba 17 "Ingresar información SLG"

Caso de prueba de aceptación	
Código: HU3_P19	Nº de Historia de Usuario: 3
Historia de usuario: Ingresar Información SLG	
Descripción: Permite ingresar la información necesaria para realizar el cálculo del SLG	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
1. Debe llenar los campos vacíos editables en la ventana.	
Resultado: Satisfactorio	

Tabla 48: Caso de prueba 18 "Calcular datos SLG"

Caso de prueba de aceptación	
Código: HU3_P20	Nº de Historia de Usuario: 3
Historia de usuario: Calcular datos SLG	
Descripción: Permite realizar cálculos pertinentes para obtener el SLG	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
1. Debe Llenar los campos vacíos editables de la ventana.	
2. Toca el botón "Calcular" para realizar el cálculo del mismo.	

3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo SLG.

Resultado: Satisfactorio

Tabla 49: Caso de prueba 19 "Guardar información SLG"

Caso de prueba de aceptación	
Código: HU3_P21	Nº de Historia de Usuario: 3
Historia de usuario: Guardar información SLG	
Descripción: Permite guardar la información de SLG	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none">1. Debe Llenar los campos vacíos editables de la ventana.2. Toca el botón "Calcular" para realizar el cálculo del mismo.3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo SLG.	
Resultado: Satisfactorio	

Tabla 50: Caso de prueba 20 "Mostrar información SLG"

Caso de prueba de aceptación	
Código: HU3_P22	Nº de Historia de Usuario: 3
Historia de usuario: Mostrar información SLG	
Descripción: Permite mostrar en pantalla la información de SLG	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	

1. Muestra la información en cuanto se abre la ventana SLG.
2. Toca el botón "Actualizar" para mostrar la información que no se ha mostrado aún.

Resultado: Satisfactorio

Tabla 51: Caso de prueba 21 "Limpiar casos SLG"

Caso de prueba de aceptación	
Código: HU3_P23	Nº de Historia de Usuario: 3
Historia de usuario: Limpiar campos SLG	
Descripción: Permite limpiar los datos que introdujo el usuario al sistema.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Limpiar" y pone los campos en blanco. 	
Resultado: Satisfactorio	

Tabla 52: Caso de prueba 22 "Visualizar resultado SLG"

Caso de prueba de aceptación	
Código: HU3_P24	Nº de Historia de Usuario: 3
Historia de usuario: Visualizar resultado SLG	
Descripción: Permite visualizar el resultado del cálculo realizado.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	

Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo.
Resultado: Satisfactorio

Tabla 53: Caso de prueba 23 "Visualizar comentario SLG"

Caso de prueba de aceptación	
Código: HU3_P25	Nº de Historia de Usuario: 3
Historia de usuario: Visualizar comentario SLG	
Descripción: Permite visualizar el comentario según el resultado que da el cálculo.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 4. Muestra el comentario de dicho resultado. 	
Resultado: Satisfactorio	

Tabla 54: Caso de prueba 24 "Eliminar SLG"

Caso de prueba de aceptación	
Código: HU1_P26	Historia de Usuario: 3
Nombre: Eliminar SLG	

Descripción: Trabaja con las funcionalidades principales para eliminar un deportista
Condiciones de Ejecución: Que haya al menos un deportista registrado
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se muestra la ventana “SLG”. 2. Selecciona al deportista. 3. Toca el botón “Eliminar” 4. Se muestra la ventana SLG con el deportista eliminado.
Resultado: Satisfactorio

3.3.2.1 Resultados de las pruebas de aceptación.

Como resultado final de estas pruebas se detectaron un conjunto de no conformidades relacionadas con errores de validación y funcionalidad. Los resultados se muestran en la ilustración 8, donde se evidencia la cantidad de casos de prueba ejecutados, los casos de prueba con no conformidades.

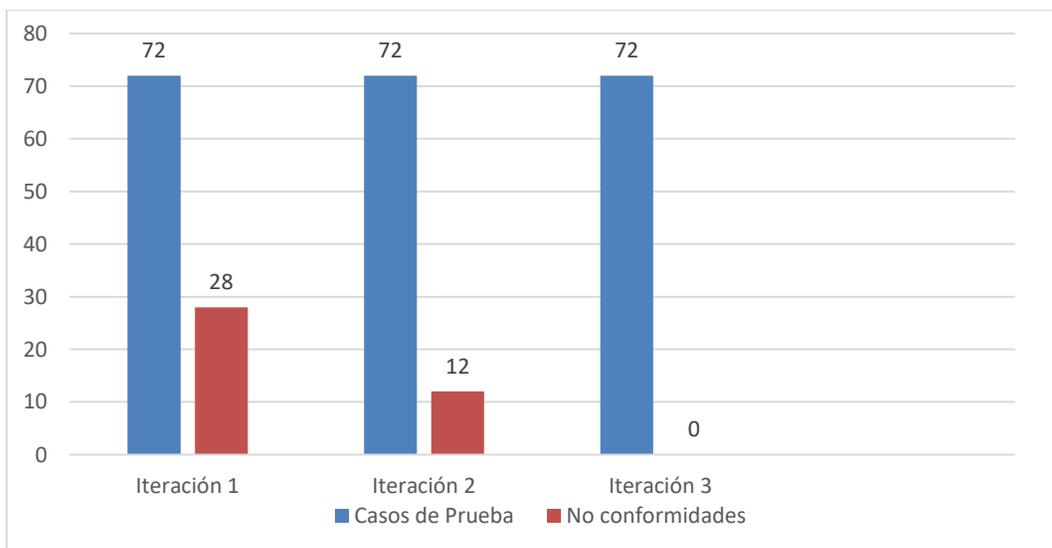


Ilustración 8: Representación del resultado de las pruebas

Se realizaron tres iteraciones donde, en una primera iteración se analizaron 72 casos de prueba de los cuales 28 resultaron no conformidades. En la segunda iteración a través de las pruebas de regresión

se verificó que las no conformidades anteriores estuviesen solucionadas, y de estas pruebas se obtuvieron 12 nuevas no conformidades, quedando resuelta en la tercera iteración y cumpliéndose correctamente todos los casos de prueba.

3.4 Conclusiones parciales.

En este capítulo se abordaron los elementos del Componente de obtención de datos estadísticos de bateadores a la ofensiva para Strike Zone, así como las pruebas realizadas al mismo y los resultados obtenidos; lo que permite concluir.

- La implementación de la propuesta de solución facilitó obtener una aplicación funcional de obtención de datos estadísticos de bateadores a la ofensiva para Strike Zone, como interfaz un repositorio con una base de datos central que contiene toda la información que se recoge del análisis estadístico que se realiza, de manera organizada.
- La validación de la propuesta de solución, mediante una estrategia de prueba de software facilitó corroborar la calidad de la misma, estableciendo así el cumplimiento de los requisitos definidos por el cliente

Conclusiones Generales

Con el desarrollo de la presente investigación se obtuvo un Componente de obtención de datos estadísticos de bateadores a la ofensiva para Strike Zone, el cual contribuye al almacenamiento de la información recopilada a partir de los archivos multimedia, con lo que se le da cumplimiento al objetivo propuesto al inicio de la investigación. Adicionalmente se concluye que:

- El estudio de las principales tendencias en el desarrollo de este dominio de aplicaciones, permitió la selección de las herramientas, tecnologías y metodología a utilizar en la presente investigación asegurando el ambiente de desarrollo necesario para la implementación de la propuesta de solución, en el cual se definió como lenguaje de programación C++, como entorno de desarrollo se escogió Qt Creator, como gestor de base de datos SQLite.
- Los requisitos del sistema y la arquitectura propuesta permitieron obtener un diseño robusto de la aplicación obteniendo la base para la implementación de la aplicación a través de las historias de usuario, además las tarjetas CRC aportaron en la identificación de las responsabilidades de las clases y las colaboraciones de las mismas y se aplicaron los patrones necesarios para lograr un diseño flexible y eficiente.
- La implementación de la propuesta de solución permitió obtener una aplicación funcional de obtención de datos estadísticos de bateadores a la ofensiva para Strike Zone, como interfaz un repositorio con una base de datos central que contiene toda la información que se recoge del análisis estadístico que se realiza, de manera organizada, la validación de la propuesta de solución y mediante una estrategia de prueba de software facilitó corroborar la calidad de la misma, estableciendo así el cumplimiento de los requisitos definidos por el cliente.

Recomendaciones

Para dar continuidad a la presente investigación se recomienda:

- Desarrollar un módulo que implemente técnicas de análisis de mayor complejidad, para garantizar la optimización del código y la fortaleza del sistema.
- Integrar el componente desarrollado a la aplicación Strike Zone y así tomar los datos de manera automática de la misma.

Referencia bibliográfica

Adrián, Y. (9 de Febrero de 2021). *Concepto definición*. Recuperado el 4 de Mayo de 2022, de Concepto definición: <https://conceptodefinicion.de/beisbol/>

Alfredo Ríos Fuentes, B. S. (2019). *Strike Zone: sistema para la recopilación y análisis de información pitch by pitch en el béisbol cubano*.

ayudaley. (s.f.). *ayudaley*. Recuperado el 22 de abril de 2022, de Qué es PostgreSQL y sus principales ventajas: https://ayudaleyprotecciondatos.es/bases-de-datos/que-es-postgresql-ventajas/#Breve_historia_de_PostgreSQL

blog, R. (20 de Abril de 2018). *Medium*. Recuperado el 1 de Mayo de 2022, de Requerimientos funcionales y no funcionales, ejemplos, tips: <https://medium.com/@requeridosblog/requerimientos-funcionales-y-no-funcionales-ejemplos-y-tips-aa31cb59b22a>

blog.centrodelearning. (15 de Diciembre de 2021). Recuperado el 21 de Abril de 2022, de https://blog.centrodelearning.com/2021/12/15/lenguajes-de-programacion-mas-utilizados/#4_C_programacion_mas_facil_y_eficiente

César Soto Valero, M. G. (2015). *Sabermetría y nuevas tendencias en el análisis estadístico del juego de béisbol*. Villa CLara, Cuba.

Coelho, F. (17 de Mayo de 2019). *Significados*. Recuperado el 30 de Abril de 2022, de Significado de metodología: <https://www.significados.com/metodologia/>

Content, R. R. (20 de Abril de 2019). *Rockcontent*. Recuperado el 21 de Abril de 2022, de Rockcontent blog: <https://rockcontent.com/es/blog/que-es-un-lenguaje-de-programacion/>

Corredor, C. D. (2013). *ABC de la Sabermetría*.

delgado, C. (29 de Marzo de 2021). *ourcodeworld.co*. Recuperado el 21 de Abril de 2022, de ourcodeworld.co: <https://ourcodeworld.co/articulos/leer/1469/que-es-un-ide-entorno-de-desarrollo-integrado>

DesdeLinux. (s.f.). Recuperado el 21 de Abril de 2022, de DesdeLinux: <https://blog.desdelinux.net/qt-creator-ide-multiplataforma-ideal-desarrolladores->

qt/?utm_source=feedburner&utm_medium=feed&utm_campaign=Feed%3A+Us
emosLinux+%28Usemos+Linux%29

Ecarri, Y. (24 de Marzo de 2016). *oSoft Blog*. Obtenido de Estilos y Patrones de Arquitectura de Software: <http://blog.osoft.es/index.php/2016/03/24/estilos-y-patrones-de-arquitecturas-de-software/#:~:text=Un%20Patr%C3%B3n%20de%20Arquitectura%20de,ese%20problema%20de%20dise%C3%B1o%20aparece.>

Facultad de Ciencias e Ingeniería. (s.f.). Obtenido de Estadística: <https://facultad.pucp.edu.pe/ingenieria/carreras/ciencias-estadisticas/que-es-la-estadistica/>

Fuentes, Alfredo Ríos; Sánchez Córdova, Beatriz; Rojas Duque de Estrada, Enrique;. (10 de Marzo de 2019). *Efdeportes.com*. Obtenido de Efdeportes.com: <https://www.efdeportes.com/efdeportes/index.php/EFDeportes/article/view/1192/632>

García, A. V. (2021). *Aplicaciones de sistemas informáticos para la enseñanza del análisis estadístico: un estudio comparativo con la enseñanza tradicional*. Obtenido de Aplicaciones de sistemas informáticos para la enseñanza del análisis estadístico: un estudio comparativo con la enseñanza tradicional: http://www.quadernsdigitals.net/datos_web/articles/edutec/congresos/edutec97/comunicaciones/experienciaseducativas/edaplicaciones.htm

Ichi pro. (2022). Obtenido de Estándares de codificación comunes que debe implementar: <https://ichi.pro/es/estandares-de-codificacion-comunes-que-debe-implementar-91480639896552#:~:text=Los%20est%C3%A1ndares%20de%20codificaci%C3%B3n%20son%20simplemente%20un%20conjunto,de%20referencia%20para%20medir%20su%20base%20de%20c%C3%B3digo.>

iiitalomorales. (2005). *Platzi*. Obtenido de ¿Qué son los patrones de diseño? Conoce para qué te sirven: <https://platzi.com/blog/patrones-de-diseno/>

(s.f.). *Ingeniería de requerimientos*.

InsideHook. (10 de Abril de 2018). *InsideHook*. Recuperado el 21 de Abril de 2022, de Does ESPN's K-Zone Ruin Baseball on TV?:

https://www.insidehook.com/daily_brief/sports/does-espns-k-zone-ruin-baseball-on-tv

Leyva, E. P. (2014). *El Análisis por envoltura de datos (DEA) aplicado a la evaluación del desempeño de los jugadores de beisbol*. Santa Clara.

Maida, E. G., Pacienza, & Julián. (2015). *Metodologías de desarrollo de software [en línea]*. Universidad Católica Argentina.

MLB. (2021). *Statcast*. Recuperado el 20 de Abril de 2022, de <https://www.mlb.com/glossary/statcast>

Novoselseva, E. (2 de Junio de 2020). *Los 5 principales patrones de arquitectura de software*. Obtenido de Apiumhub: <https://apiumhub.com/es/tech-blog-barcelona/principales-patrones-arquitectura-software/>

OpenCV, E. d. (2022). *OpenCV*. Obtenido de Acerca de: [https://opencv.org/about/#:~:text=OpenCV%20\(Open%20Source%20Computer%20Vision,perception%20in%20the%20commercial%20products.](https://opencv.org/about/#:~:text=OpenCV%20(Open%20Source%20Computer%20Vision,perception%20in%20the%20commercial%20products.)

Pérez, S. D. (18 de Octubre de 2021). *Intelequia*. Recuperado el 22 de Abril de 2022, de ¿QUÉ ES MICROSOFT SQL SERVER Y PARA QUÉ SIRVE?: <https://intelequia.com/blog/post/2948/qu%C3%A9-es-microsoft-sql-server-y-para-qu%C3%A9-sirve>

Rapsodo. (2019). *rapsodo*. Recuperado el 20 de Abril de 2022, de rapsodo: <https://rapsodo.com/who-we-are/>

Robledano, A. (22 de Julio de 2019). *OpenWebinars*. Recuperado el 21 de Abril de 2022, de OpenWebinars: <https://openwebinars.net/blog/que-es-cpp/>

Robledano, A. (24 de Septiembre de 2019). *OpenWebinars*. Recuperado el 22 de Abril de 2022, de Qué es MySQL: Características y ventajas: <https://openwebinars.net/blog/que-es-mysql/>

Roldán, P. N. (31 de Julio de 2017). *Encomipedia*. Obtenido de Estadísticas: <https://economipedia.com/definiciones/estadistica.html>

Silva, J. C. (s.f.). *academia*. Obtenido de Pruebas de software: https://www.academia.edu/8813345/Pruebas_de_Software

Soto Valero, C., & González Castellanos, M. (2015). *Sabermetría y nuevas tendencias en el análisis estadístico del juego de béisbol*. Murcia, España.

Spantip. (s.f.). *spantip.com*. Obtenido de spantip.com:
<https://www.spantip.com/wiki/PITCHf/x>

Tododxts. (2022). Recuperado el 4 de Mayo de 2022, de Educación física:
<https://www.tododxts.com/ciencias-actividad-fisica/educacion-fisica/que-son-deportes-colectivos-concepto>

Zorita, A. (14 de Octubre de 2016). *ByteCode*. Obtenido de Librería informática, ¿De qué se trata?: <https://bytecode.es/que-es-una-libreria-informatica/>

Anexos

A.1 Estadísticos saber métricos para el béisbol.

Estadísticas de bateadores:

- **BABIP (Promedio de pelotas en juego):** Es el Average sin contar los ponches ni los jonrones; en otras palabras, dice cuántas pelotas que fueron bateadas cayeron en hit.
- **BB% (Porcentaje de boletos):** Más que dar un hit, lo más importante es evitar que te hagan out y embasarte. Comparar los boletos con las apariciones legales permite ver qué tanto ayuda un bateador con su paciencia a su OBP.
- **ISO (Poder Aislado):** Se le resta el promedio al Slugging y el resultado es el poder bruto del jugador. En otras palabras, resta los sencillos a las bases conseguida con dobles, triple y jonrones. Básicamente mide la capacidad de dar extrabases.
- **RC (Carreras Creadas):** Básicamente usa el OBP y el SLG de un jugador para calcular cuántas carreras totales ha aportado a su equipo. La fórmula inicial, propuesta por Bill James, era simplemente $OBP * SLG * VG$. Sin embargo, ha tenido varias modificaciones donde se le da más importancia al OBP, así como se añade el promedio con hombres en posición anotadora y el éxito en el robo de bases.

A.1.1 Resumen de los principales estadísticos saber métricos



Ilustración 9: Resumen de los principales estadísticos saber métricos

A.1.2 Estadísticos sabermétricos de bateo

Tabla 55: Estadísticos sabermétricos de bateo

Estadísticos sabermétricos de bateo: la idea es neutralizar los factores que no dependan del bateador, como el desempeño de sus contrarios, las características del terreno donde juega, etc.		
Estadístico de bateo	¿Cómo se calcula?	Descripción
OBP (On Base Percentage o Porcentaje de Embasado)	$\frac{H + BB + HBP}{AB + BB + HBP + SF}$	El OBP es la estadística que comenzó la revolución sabermétrica en el béisbol, demostrando que es más importante evitar que al bateador lo pongan out (lo que mide el OBP) a dar un hit (lo que mide el AVG). Los jugadores con buen OBP (más de .350) suelen estar en los primeros lugares en la alineación (1-4). Para tener un buen OBP ayuda que el bateador sea paciente y se tome un buen número de boletos. Promedio de MLB (Major League Baseball): .330
SLG (Slugging)	$\frac{1B+(2*2B)+(3*3B)+(4*4B)}{AB+SF}$	Relaciona los turnos al bate con las bases alcanzadas. Cada base equivale a mil puntos (HR=4.000, 2B=2.000, etc.). Un turno fallido tiene valor cero. Un buen Slugging depende del poder del jugador y de su capacidad para mantener un buen promedio de bateo. Regularmente un bateador

		con un Slugging elevado supera la marca de .600. Promedio de MLB: .420.
OPS (On Base Plus Slugging, o OBP más SLG)	OBP + SLG	Es la suma del OBP y el SLG para unir la utilidad de ambos en una sola estadística más completa para los bateadores. Como el valor máximo de OBP es 1 y el máximo de Slugging es 4, el valor máximo de OPS posible es 5. Se estima que un bateador es realmente completo una vez que su OPS supera el valor de 1.000. Promedio de MLB: .750.
BABIP (Batting Average of Balls in Play, o Promedio de Pelotas en Juego)	$\frac{H + HR}{AB - K - HR - SF}$	Se trata de la medición del promedio de bateo tomando en consideración únicamente las pelotas puestas en juego que no resultan en un error o en un cuadrangular, descartando igualmente las bases por bolas y los ponches; en otras palabras, dice cuántas pelotas que fueron bateadas cayeron de hit. En el caso de los bateadores, ellos sí tienen control sobre su BABIP. Por ejemplo, bateadores de líneas que suelen hacer buen contacto con la pelota o bateadores rápidos tienden a superar los .300 puntos de

		<p>BABIP (y viceversa). Un BABIP en períodos cortos (hasta una temporada) se puede comparar con el BABIP vitalicio del bateador y ver qué tanto influyó la suerte en su desempeño. Un BABIP muy alto puede indicar que más pelotas de lo normal han caído de hit, por lo que se puede esperar que tanto el BABIP, como el promedio y el OBP disminuyan en el futuro. Promedio de MLB: .300.</p>
<p>BB% (Walk Rate, o Frecuencia de Bases por Bolas)</p>	$\frac{BB}{AB}$	<p>Estima con qué regularidad un jugador recibe una base por bolas cuando va a batear, lo que en cierto modo indica lo paciente que puede ser un bateador en el plato. Considera que más que dar un hit, lo más importante es evitar que pongan out al bateador, lo cual se esté consigue obteniendo bases por bolas. Comparar los boletos con las apariciones legales permite ver qué tanto ayuda un bateador con su paciencia a su OBP. Favorece a los bateadores de poder y aquellos que reciben muchos lanzamientos por turno. Promedio de MLB: 10%.</p>

<p>ISO (Isolated Power, o Poder Aislado)</p>	<p>SLG - AVG</p>	<p>Resta los sencillos a las bases conseguidas con dobles, triples y cuadrangulares con lo cual se mide la capacidad de dar extrabases, calculándose el poder bruto del bateador. Aunque lo ideal sería que el jugador lograra un equilibrio entre los aportes en promedios de bases alcanzadas y de poder, la mayoría de las veces no sucede así, por lo que el ISO constituye una medida importante para determinar los méritos ofensivos del jugador. En este sentido, se puede decir que bateadores que cuentan con un ISO por debajo del promedio son jugadores de velocidad que dependen mucho de los sencillos para mantener sus promedios de bateo. Igualmente es importante señalar que el ISO es una estadística que alcanza una significancia predictiva a los 550 turnos al bate o más, por lo que un ISO de 0.350 durante unos 50 turnos es una muestra insuficiente para predecir si este valor corresponde al talento real</p>
---	------------------	---

		del bateador o no. Promedio de MLB: .155.
RC (Runs Created, o Carreras Creadas)	$\frac{(H + BB) * TB}{PA}$	Estima cuántas carreras ha aportado un bateador a su equipo, independientemente del rendimiento de dicho equipo. Esta es la fórmula inicial propuesta por Bill James, la cual ha tenido varias modificaciones en las que se le da una mayor o menor importancia al OBP, así como se le añade el promedio con hombres en posición anotadora y el éxito en el robo de bases etc. Promedio de MLB: 75.

A.2 Historias de usuario.

Tabla 56: Historia de usuario 14 " Visualizar información SLG"

Historia de usuario	
Número: 14	Nombre: Visualizar información SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "SLG".	

Interfaz

Tabla 57: Historia de usuario 15 "Ingresar información SLG"

Historia de usuario	
Número: 15	Nombre: Ingresar información SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario ingresar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "SLG".	
Interfaz	

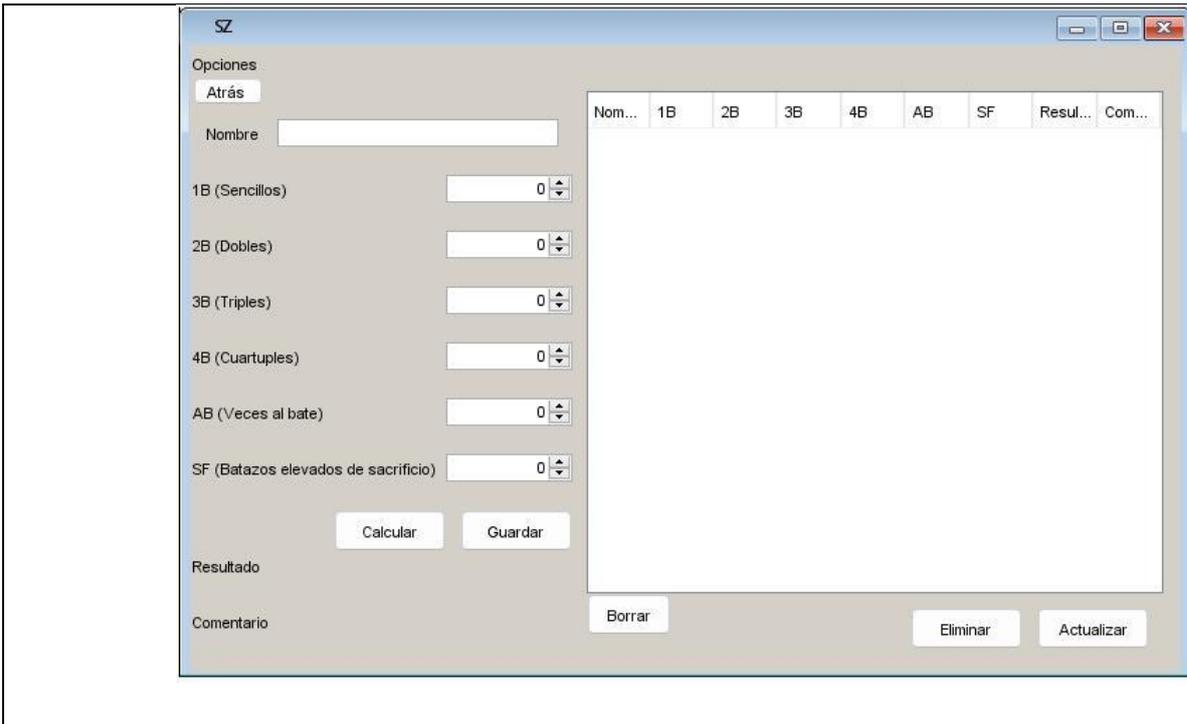


Tabla 58: Historia de usuario 16 "Calcular datos SLG "

Historia de usuario	
Número: 16	Nombre: Calcular datos SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario realizar el cálculo del estadístico a partir de los datos introducidos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido la información del cálculo en el sistema y a su vez después de haber presionado el botón de "Calcular".	
Interfaz	

Tabla 59: Historia de usuario 17 "Guardar información SLG"

Historia de usuario	
Número: 17	Nombre: Guardar información SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario guardar la información del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo y a su vez después de haber presionado el botón de "Guardar".	
Interfaz	

Tabla 60: Historia de usuario 18 "Mostrar información SLG "

Historia de usuario	
Número: 18	Nombre: Mostrar información SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario mostrar la información guardada del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber guardado la información del cálculo estadístico y a su vez después de haber presionado el botón de "Actualizar".	
Interfaz	

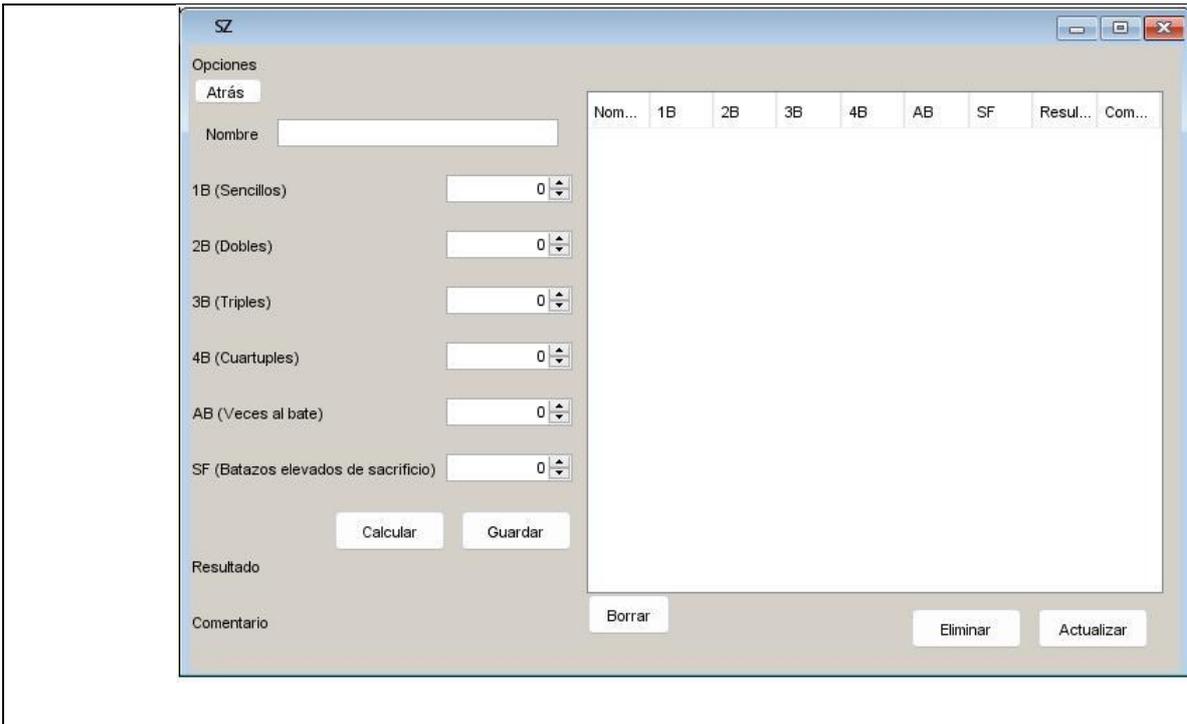


Tabla 61: Historia de usuario 19 "Limpiar información SLG"

Historia de usuario	
Número: 19	Nombre: Limpiar información SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario limpiar los campos editables para introducir nuevos datos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido información del en los campos editables y a su vez después de haber presionado el botón con un ícono que indica la acción de "Borrar".	
Interfaz	

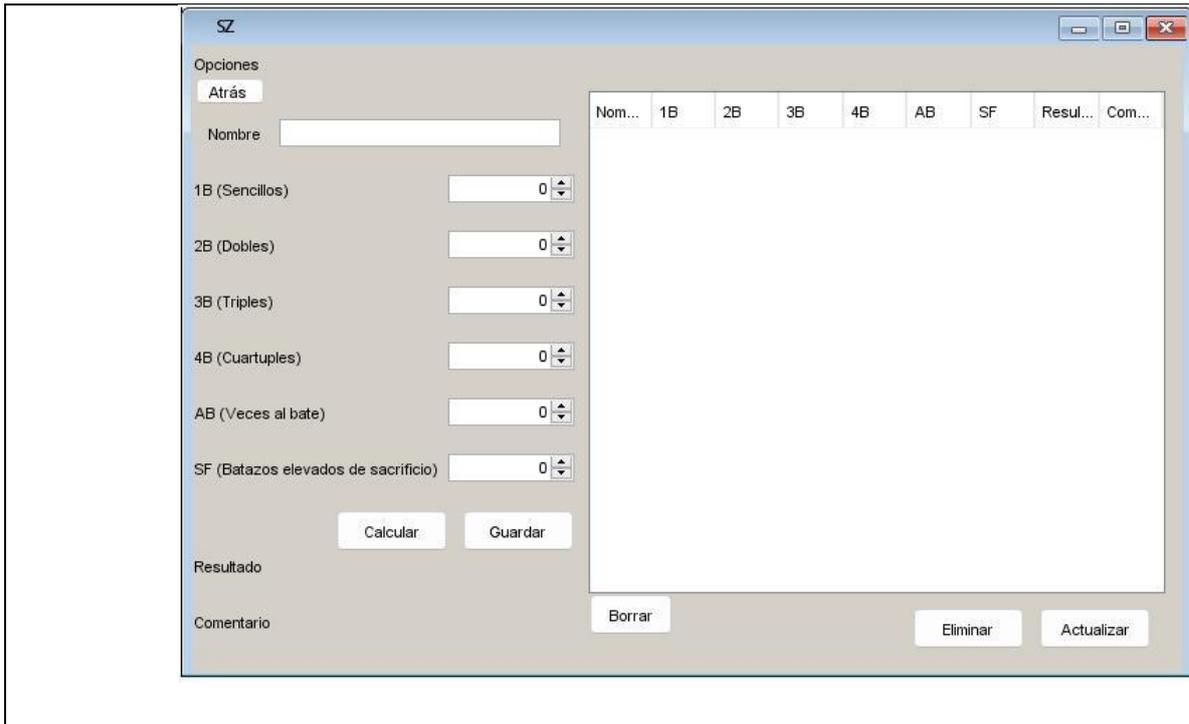


Tabla 62: Historia de usuario 20 "Visualizar resultado SLG"

Historia de usuario	
Número: 20	Nombre: Visualizar resultado SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar el resultado del cálculo estadístico que se realizó.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

Tabla 63: Historia de usuario 21 "Visualizar comentario SLG "

Historia de usuario	
Número: 21	Nombre: Visualizar comentario SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: <p>El componente debe permitir al usuario visualizar el comentario del cálculo estadístico que se realizó a partir del resultado.</p>	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

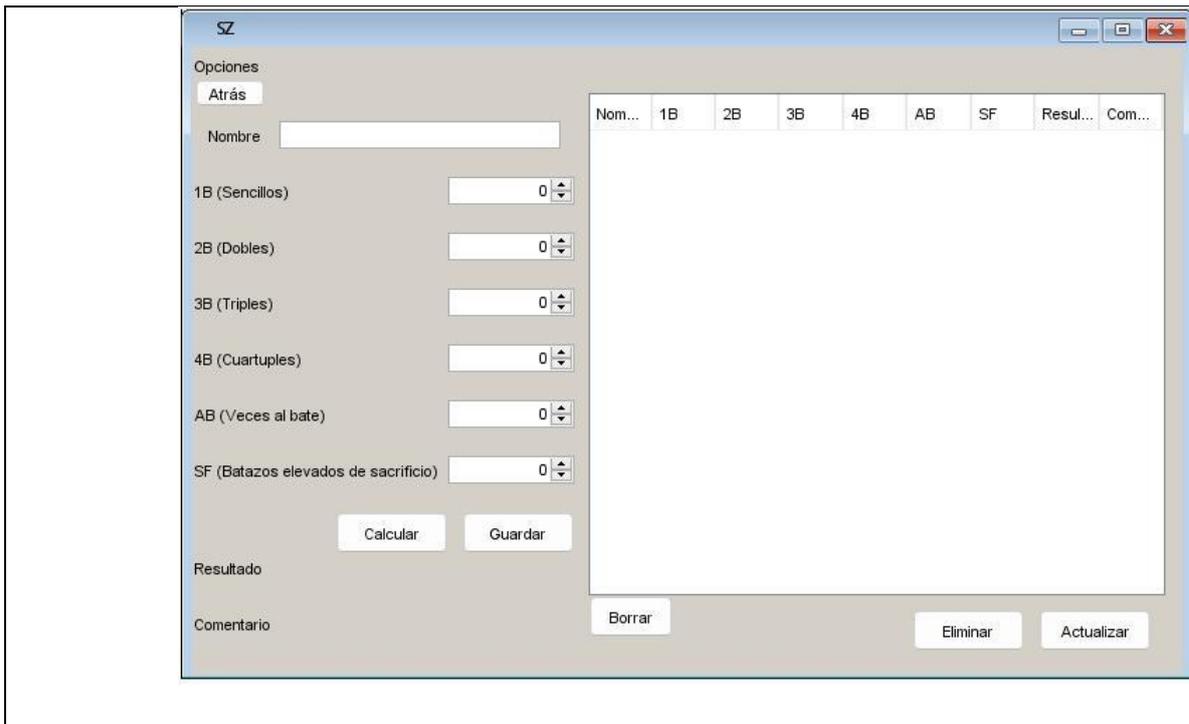


Tabla 64: Historia de usuario 22 “Eliminar SLG”

Historia de usuario	
Número: 22	Nombre: Eliminar SLG
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.3	Iteración: 1
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario eliminar la información guardada.	
Observaciones: Esta historia de usuario se ejecutará luego de que se muestre y se visualice la información del cálculo estadístico y a su vez que se seleccione uno de los mismos y se presione el botón “Eliminar”.	
Interfaz	

Tabla 65: Historia de usuario 23 " Visualizar información OPS "

Historia de usuario	
Número: 23	Nombre: Visualizar información OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "OPS".	
Interfaz	

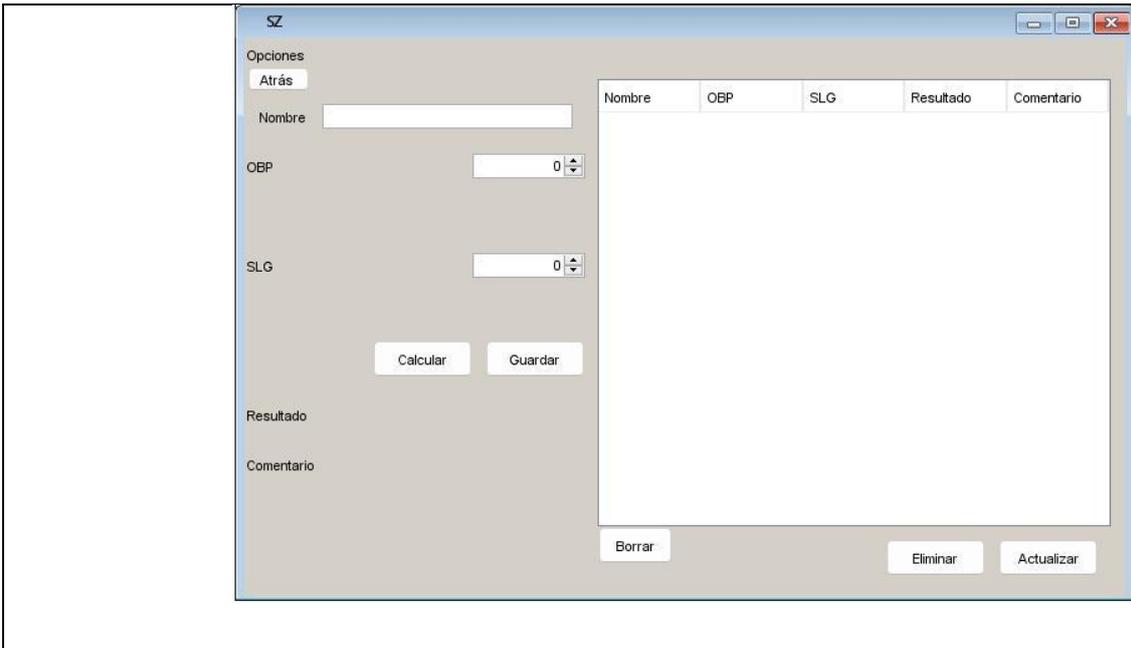


Tabla 66: Historia de usuario 24 "Ingresar información OPS"

Historia de usuario	
Número: 24	Nombre: Ingresar información OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario ingresar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "OPS".	
Interfaz	

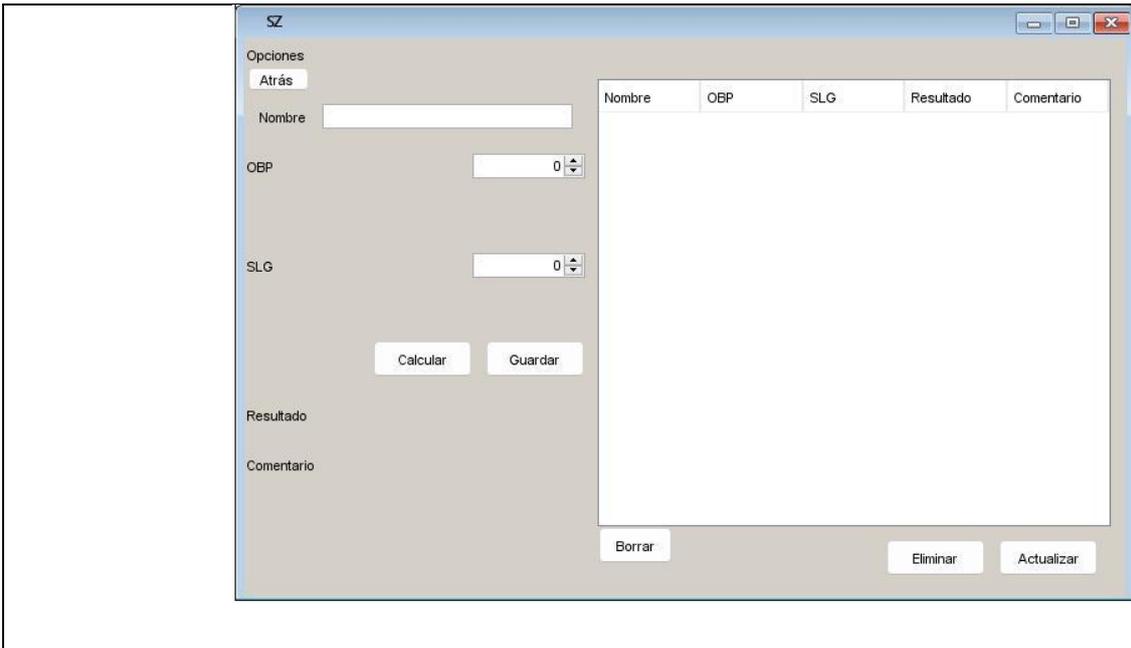


Tabla 67: Historia de usuario 25 "Calcular datos OPS"

Historia de usuario	
Número: 25	Nombre: Calcular datos OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario realizar el cálculo del estadístico a partir de los datos introducidos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido la información del cálculo en el sistema y a su vez después de haber presionado el botón de "Calcular".	
Interfaz	

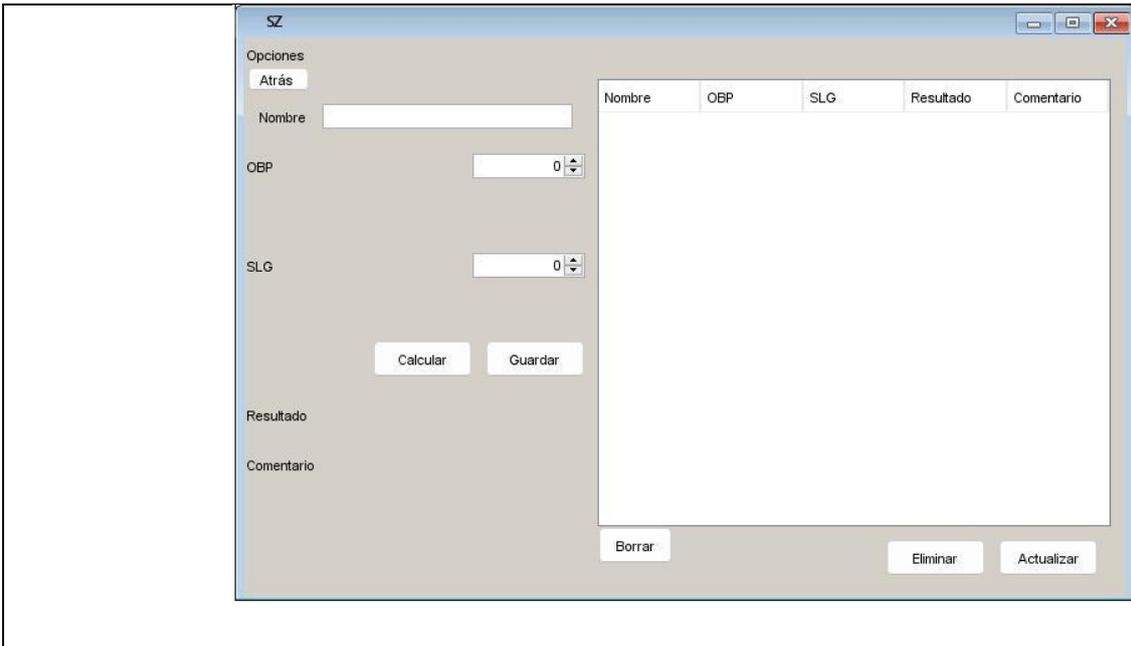


Tabla 68: Historia de usuario 26 "Guardar información OPS"

Historia de usuario	
Número: 26	Nombre: Guardar información OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario guardar la información del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo y a su vez después de haber presionado el botón de "Guardar".	
Interfaz	

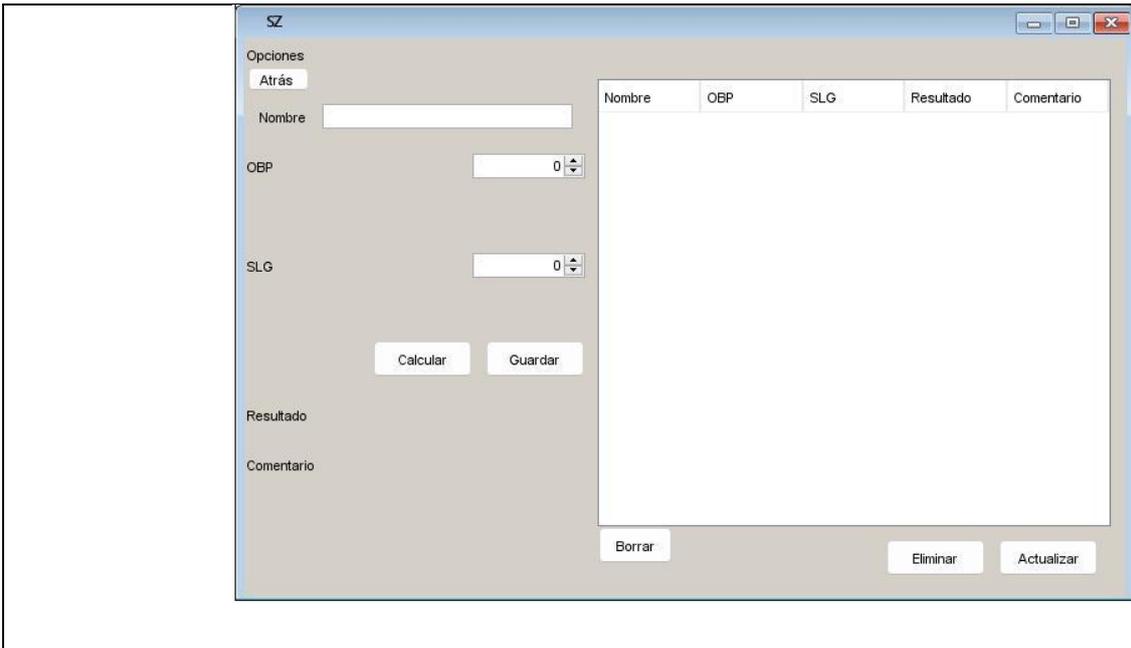


Tabla 69: Historia de usuario 27 "Mostrar información OPS"

Historia de usuario	
Número: 27	Nombre: Mostrar información OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario mostrar la información guardada del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber guardado la información del cálculo estadístico y a su vez después de haber presionado el botón de "Actualizar".	
Interfaz	

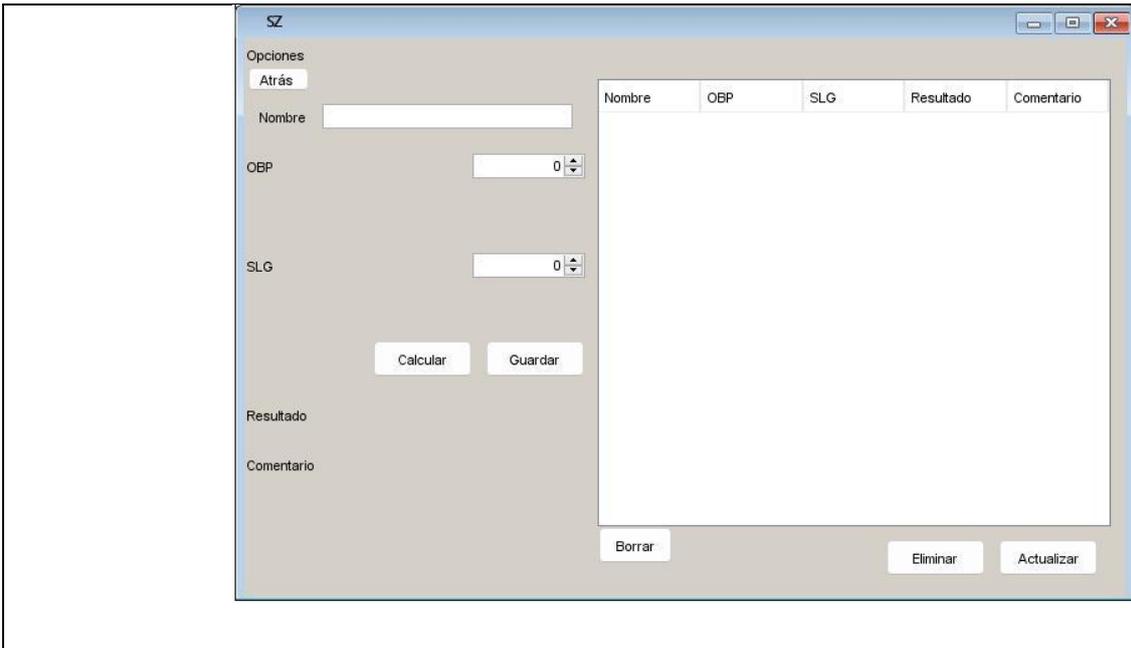


Tabla 70: Historia de usuario 28 "Limpiar información OPS "

Historia de usuario	
Número: 28	Nombre: Limpiar información OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario limpiar los campos editables para introducir nuevos datos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido información del en los campos editables y a su vez después de haber presionado el botón con un ícono que indica la acción de de "Borrar".	
Interfaz	

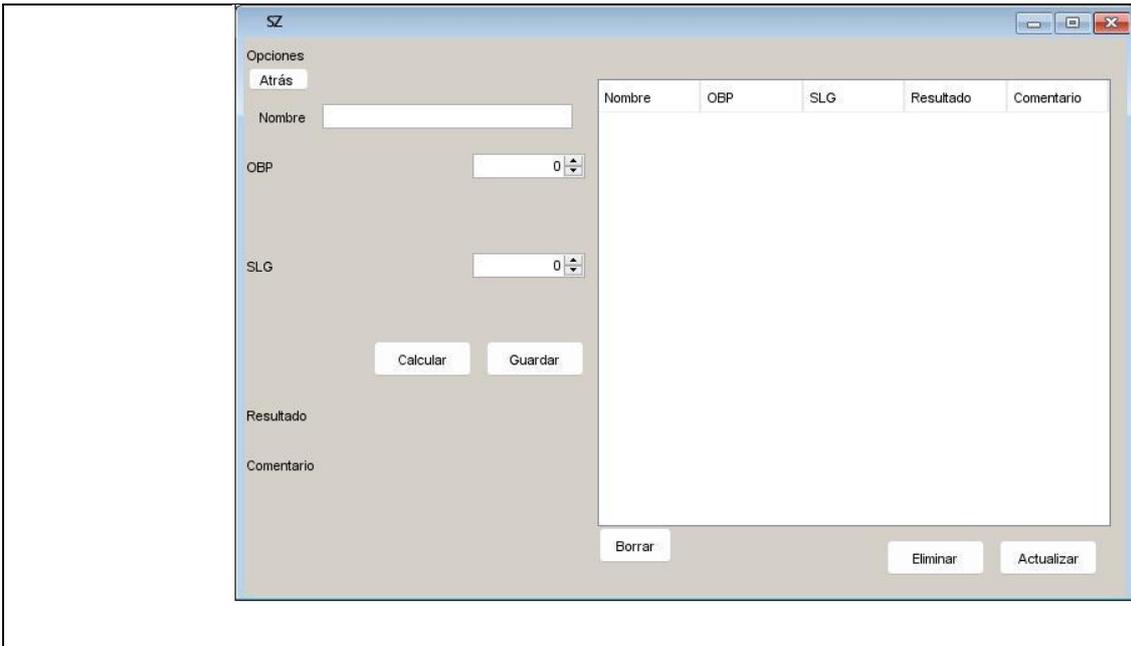


Tabla 71: historia de usuario 29 "Visualizar resultado OPS "

Historia de usuario	
Número: 29	Nombre: Visualizar resultado OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar el resultado del cálculo estadístico que se realizó.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

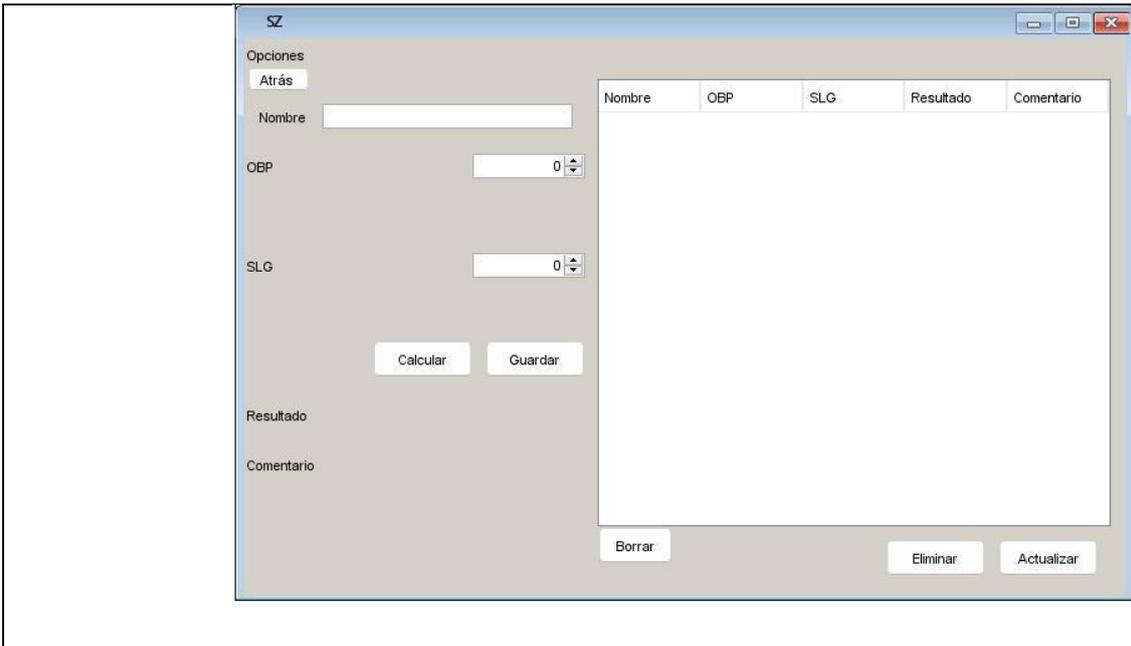


Tabla 72: Historia de usuario 30 "Visualizar comentario OPS"

Historia de usuario	
Número: 30	Nombre: Visualizar comentario OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: <p>El componente debe permitir al usuario visualizar el comentario del cálculo estadístico que se realizó a partir del resultado.</p>	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

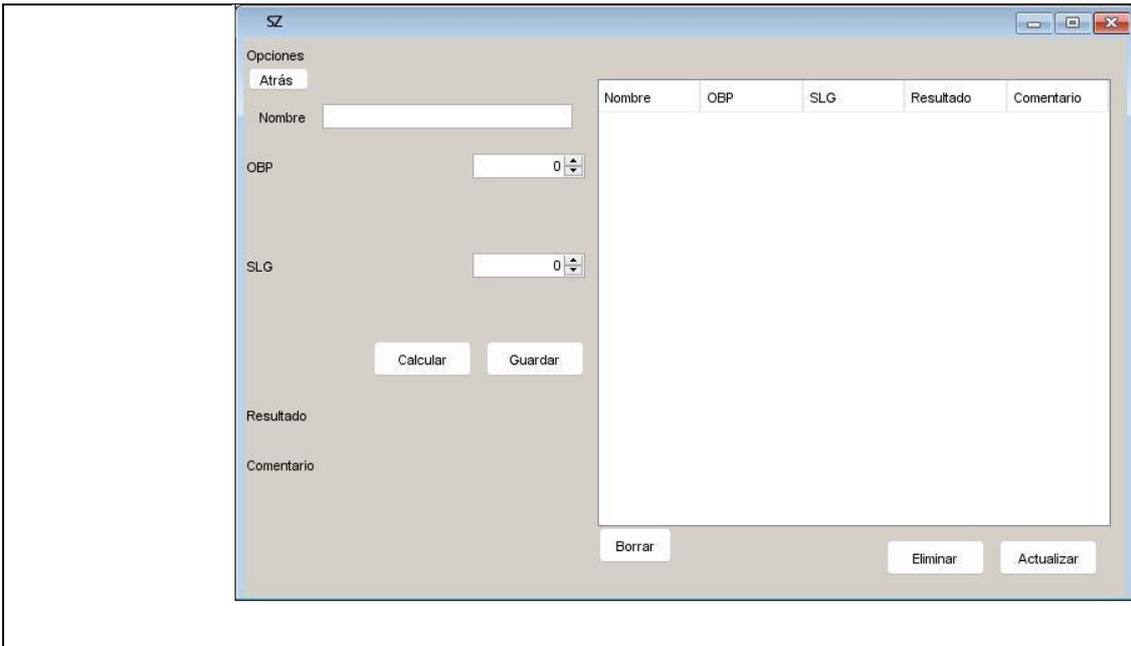


Tabla 73: Historia de usuario 31 "Eliminar OPS"

Historia de usuario	
Número: 31	Nombre: Eliminar OPS
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.3	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario eliminar la información guardada.	
Observaciones: Esta historia de usuario se ejecutará luego de que se muestre y se visualice la información del cálculo estadístico y a su vez que se seleccione uno de los mismos y se presione el botón "Eliminar".	
Interfaz	

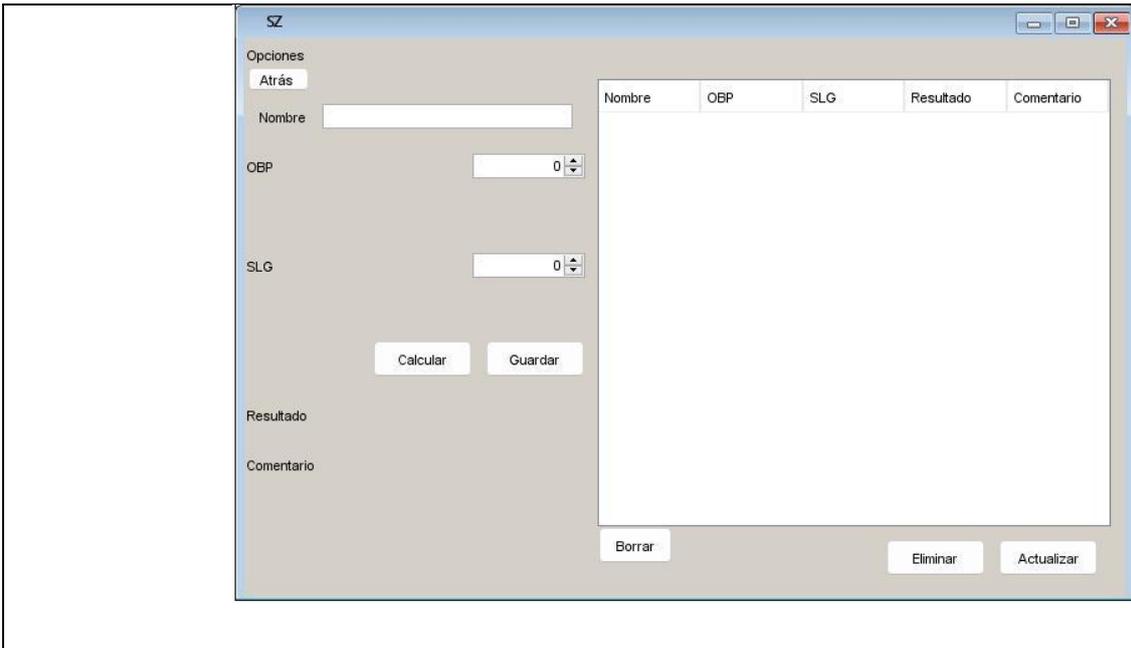


Tabla 74: Historia de usuario 5 " Visualizar información BABIP "

Historia de usuario	
Número: 32	Nombre: Visualizar información BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "BABIP".	
Interfaz	

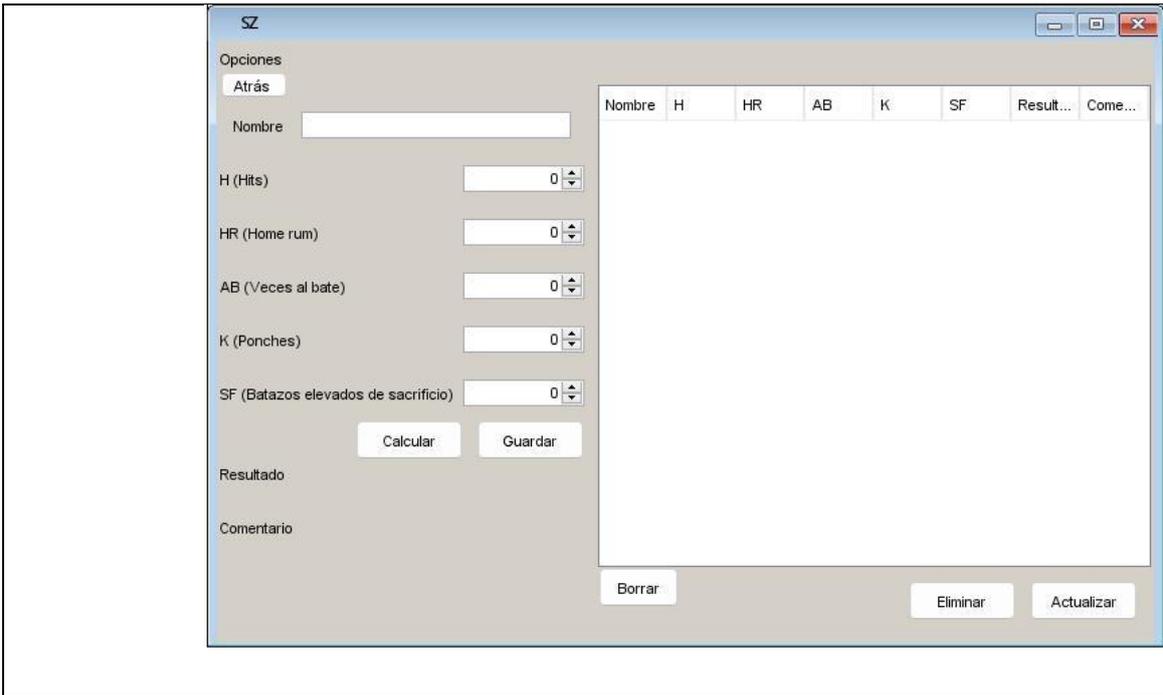


Tabla 75: Historia de usuario 33 "Ingresar información BABIP"

Historia de usuario	
Número: 33	Nombre: Ingresar información BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario ingresar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "BABIP".	
Interfaz	

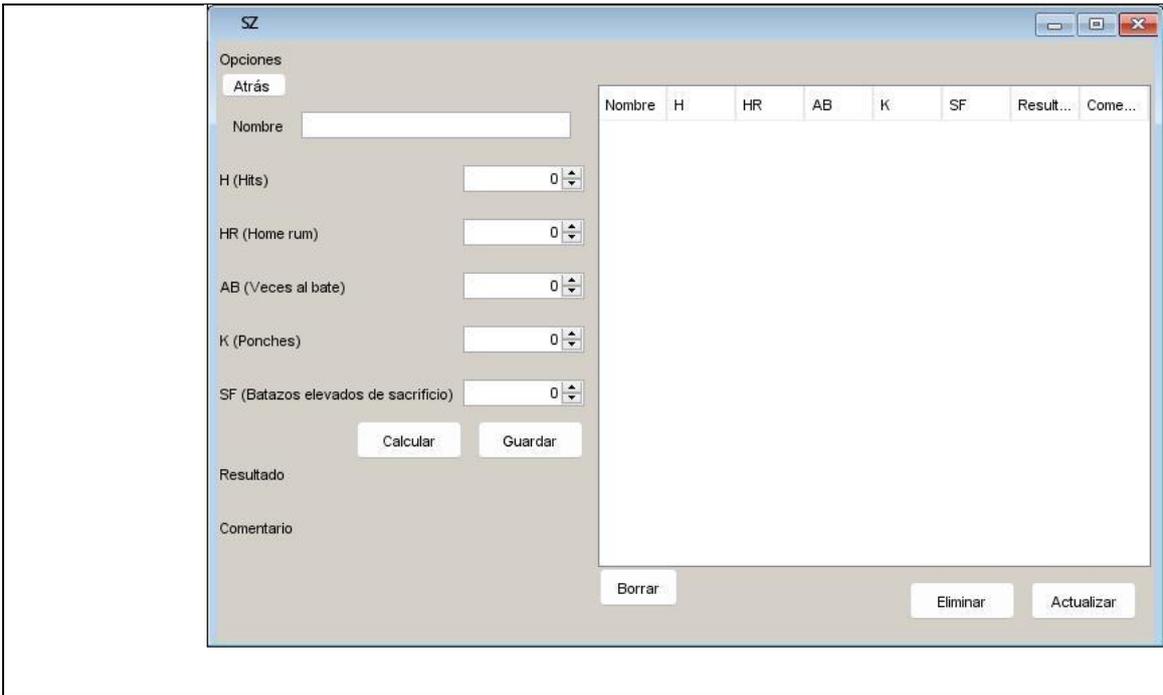


Tabla 76: Historia de usuario 34 "Calcular datos BABIP "

Historia de usuario	
Número: 34	Nombre: Calcular datos BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario realizar el cálculo del estadístico a partir de los datos introducidos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido la información del cálculo en el sistema y a su vez después de haber presionado el botón de "Calcular".	
Interfaz	

Tabla 77: Historia de usuario 35 "Guardar información BABIP"

Historia de usuario	
Número: 35	Nombre: Guardar información BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario guardar la información del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo y a su vez después de haber presionado el botón de "Guardar".	
Interfaz	

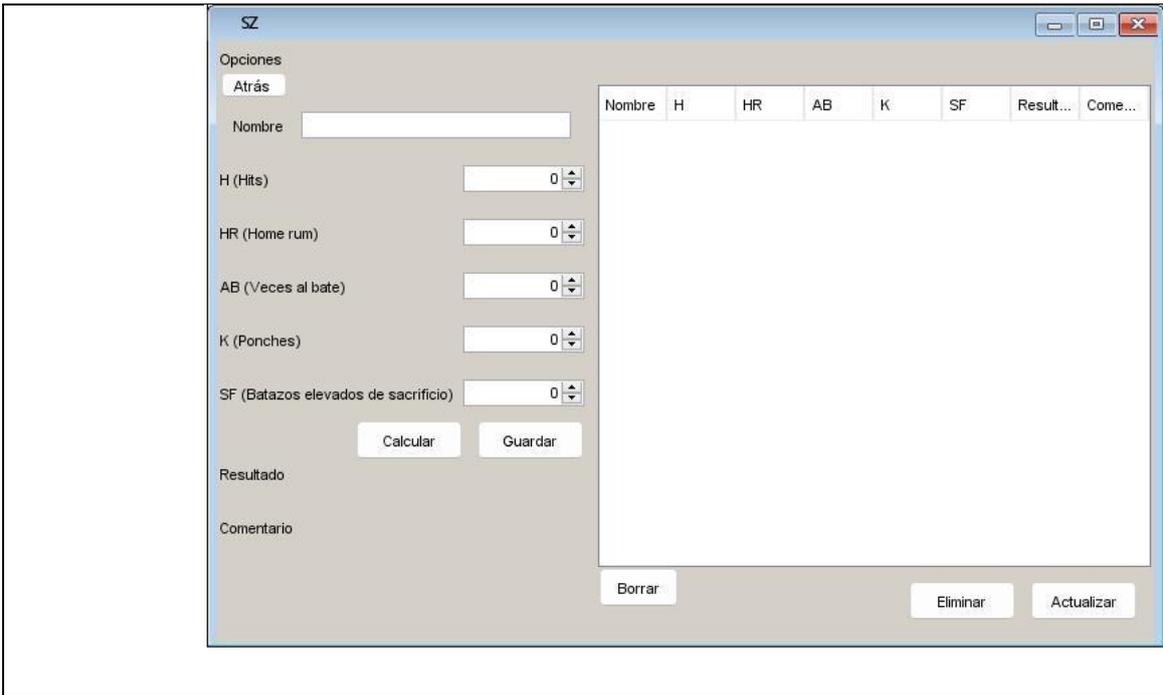


Tabla 78: Historia de usuario 36 "Mostrar información BABIP"

Historia de usuario	
Número: 36	Nombre: Mostrar información BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario mostrar la información guardada del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber guardado la información del cálculo estadístico y a su vez después de haber presionado el botón de "Actualizar".	
Interfaz	

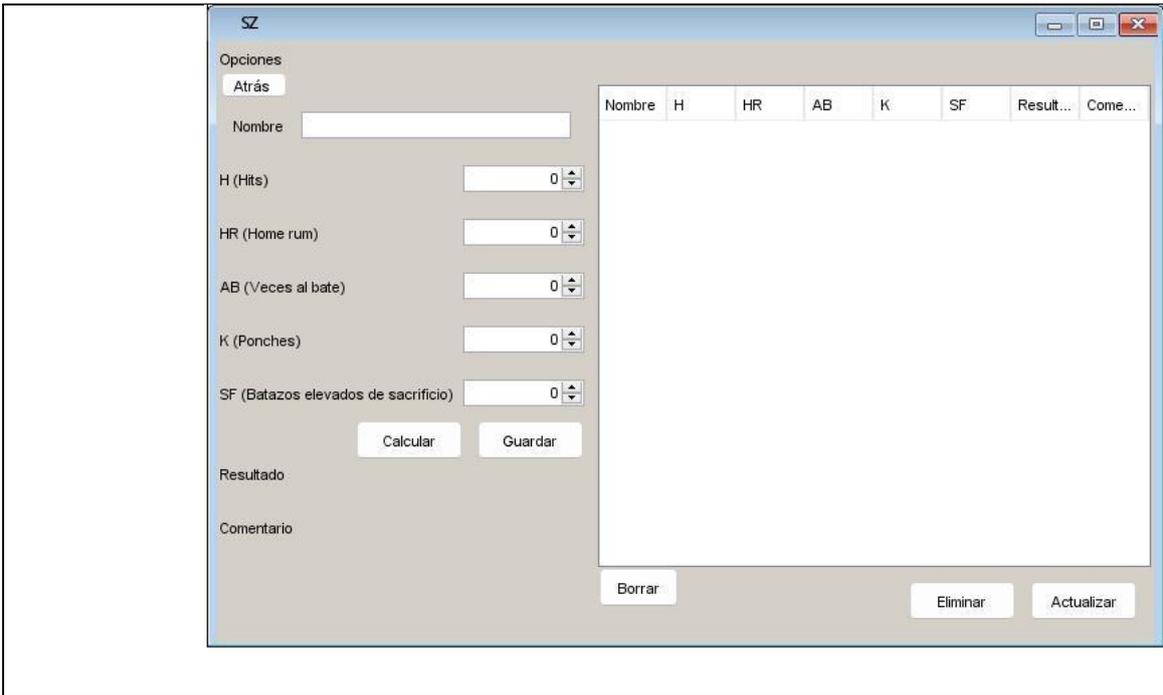


Tabla 79: Historia de usuario 37 "Limpiar información BABIP"

Historia de usuario	
Número: 37	Nombre: Limpiar información BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario limpiar los campos editables para introducir nuevos datos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido información del en los campos editables y a su vez después de haber presionado el botón con un ícono que indica la acción de de "Borrar".	
Interfaz	

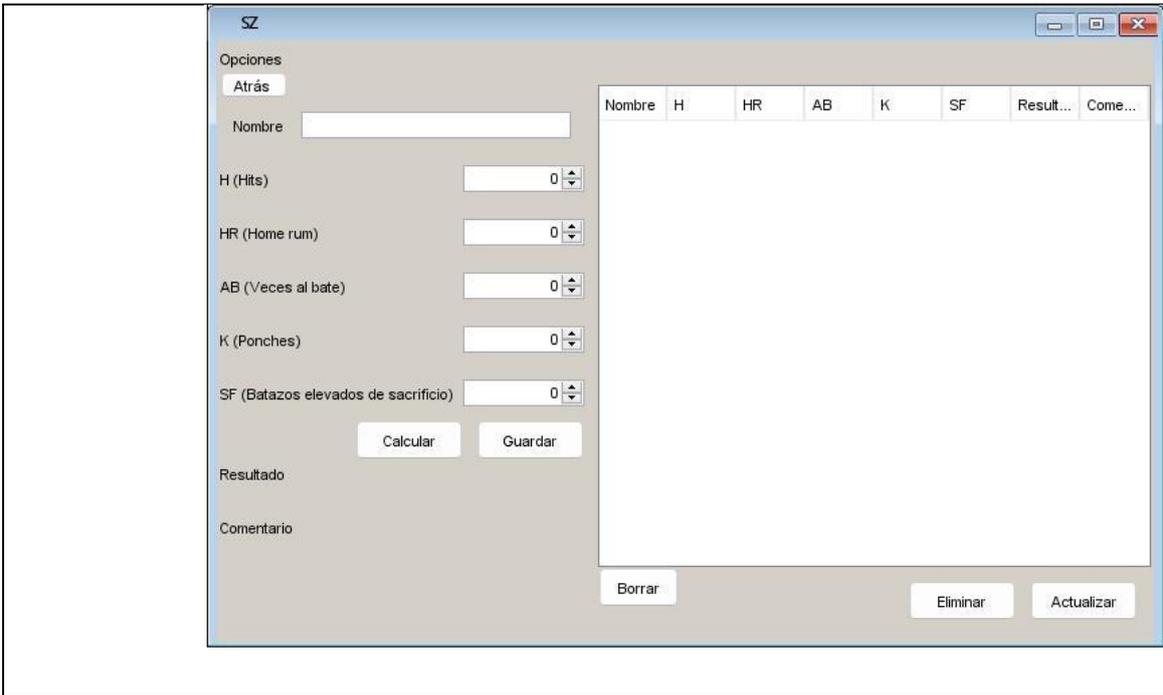


Tabla 80: Historia de usuario 38 "Visualizar resultado BABIP"

Historia de usuario	
Número: 38	Nombre: Visualizar resultado BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar el resultado del cálculo estadístico que se realizó.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

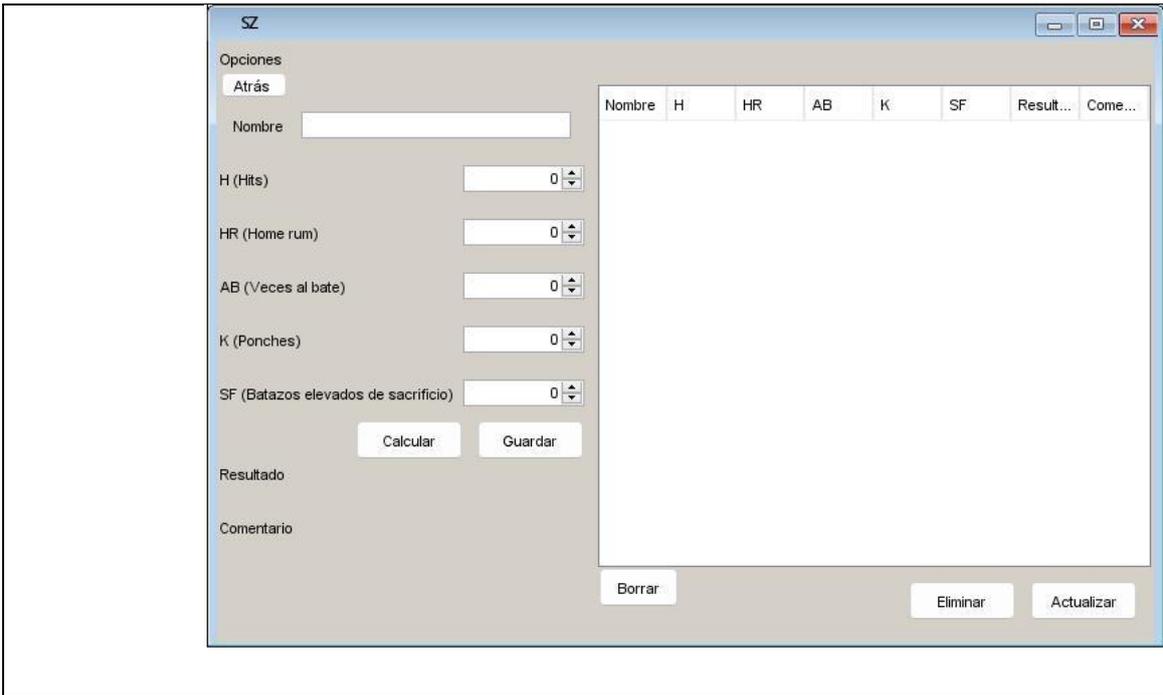


Tabla 81: Historia usuario 39 "Visualizar comentario BABIP"

Historia de usuario	
Número: 39	Nombre: Visualizar comentario BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar el comentario del cálculo estadístico que se realizó a partir del resultado.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

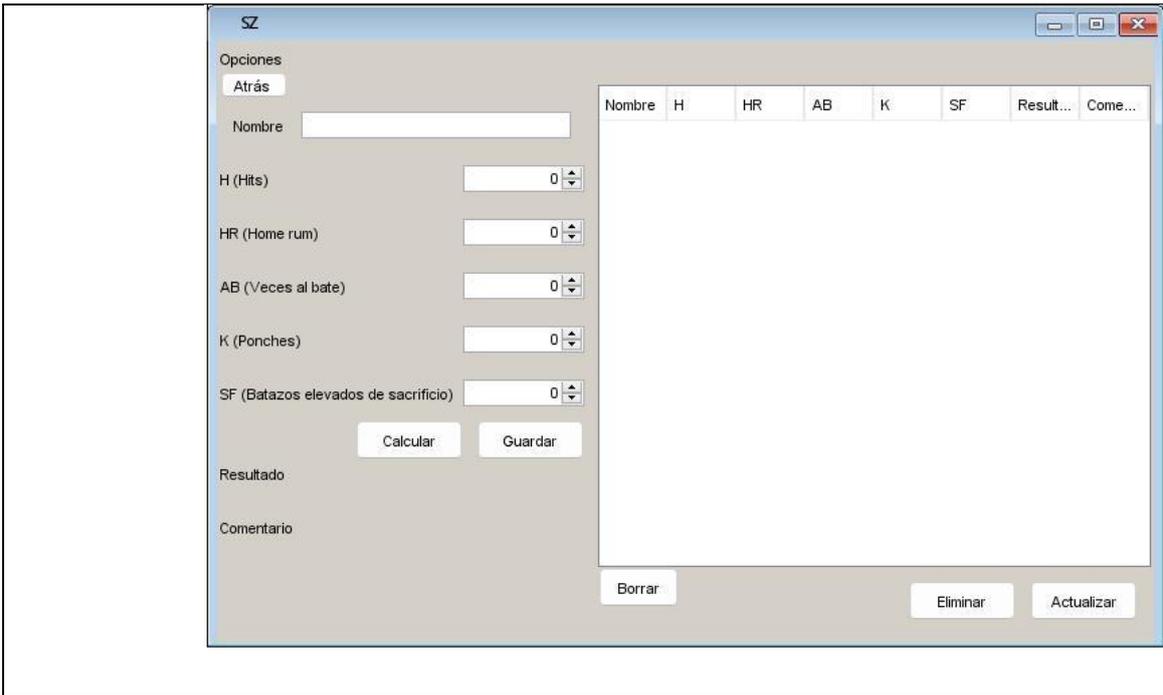


Tabla 82: Historia usuario 40 "Eliminar BABIP"

Historia de usuario	
Número: 40	Nombre: Eliminar BABIP
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.3	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario eliminar la información guardada.	
Observaciones: Esta historia de usuario se ejecutará luego de que se muestre y se visualice la información del cálculo estadístico y a su vez que se seleccione uno de los mismos y se presione el botón "Eliminar".	
Interfaz	

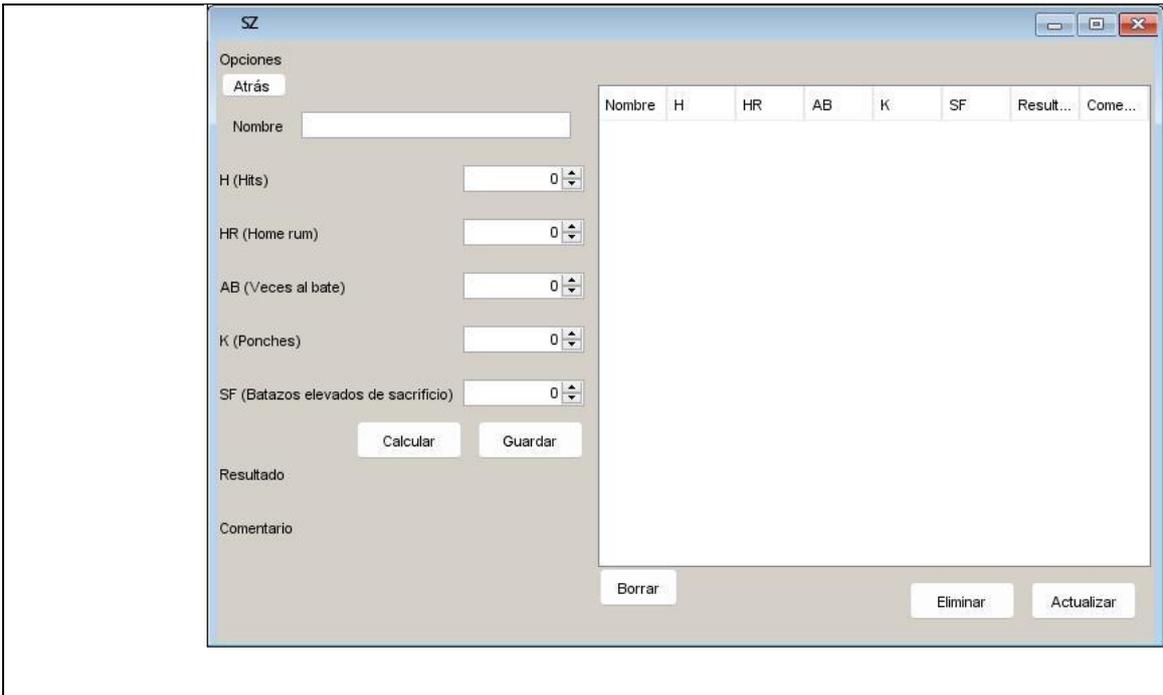


Tabla 83: Historia de usuario 6 " Visualizar información BB%"

Historia de usuario	
Número: 41	Nombre: Visualizar información BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "BB%".	
Interfaz	

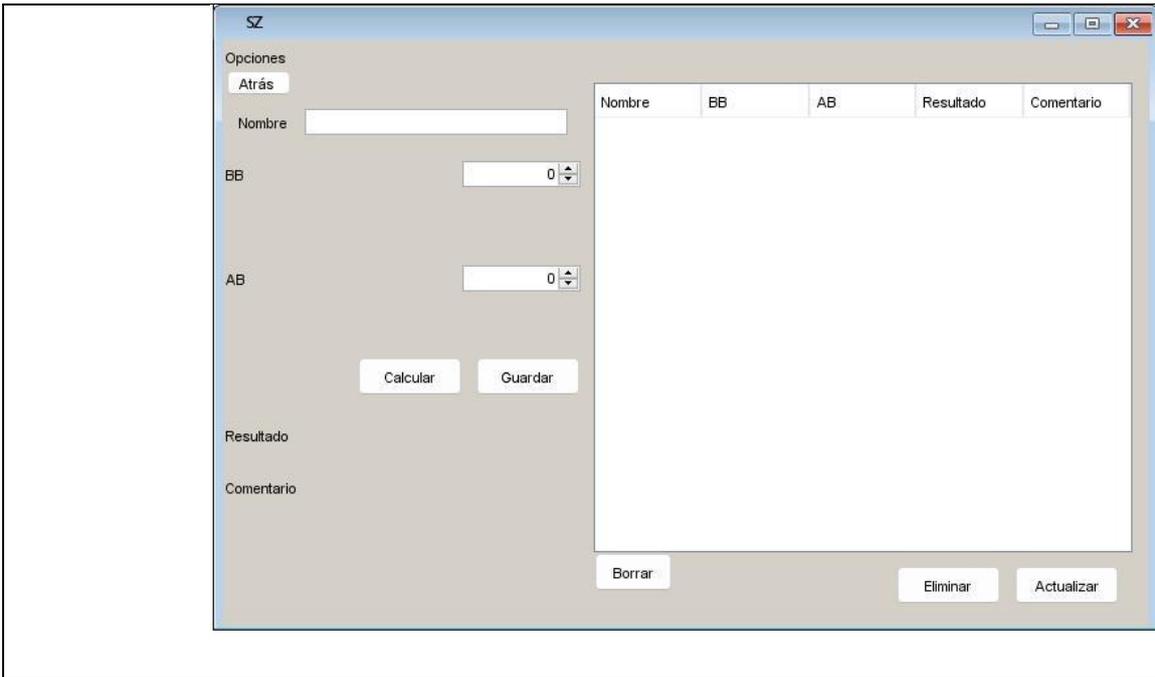


Tabla 84: Historia de usuario 42 "Ingresar información BB% "

Historia de usuario	
Número: 42	Nombre: Ingresar información BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario ingresar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "BB%".	
Interfaz	

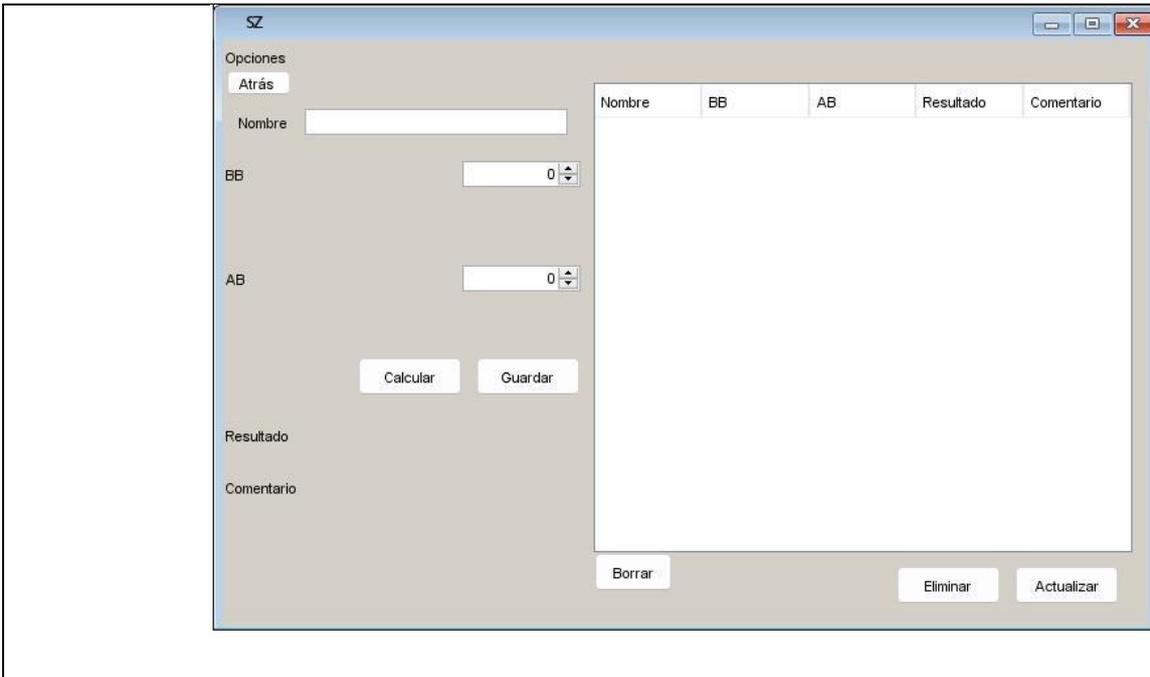


Tabla 85: Historia de usuario 43 "Calcular datos BB% "

Historia de usuario	
Número: 43	Nombre: Calcular datos BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario realizar el cálculo del estadístico a partir de los datos introducidos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido la información del cálculo en el sistema y a su vez después de haber presionado el botón de "Calcular".	
Interfaz	

The screenshot shows a window titled "SZ" with a form for saving user information. The form includes the following elements:

- Opciones:** A button labeled "Atrás".
- Nombre:** A text input field.
- BB:** A numeric input field with a dropdown arrow, currently showing "0".
- AB:** A numeric input field with a dropdown arrow, currently showing "0".
- Buttons:** "Calcular" and "Guardar" are located below the input fields. "Borrar", "Eliminar", and "Actualizar" are located at the bottom of the window.
- Table:** A table on the right side of the window with the following columns: "Nombre", "BB", "AB", "Resultado", and "Comentario". The table is currently empty.
- Labels:** "Resultado" and "Comentario" are labels for the corresponding columns in the table.

Tabla 86: Historia de usuario 44 "Guardar información BB%"

Historia de usuario	
Número: 44	Nombre: Guardar información BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario guardar la información del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo y a su vez después de haber presionado el botón de "Guardar".	
Interfaz	

The screenshot shows a software window titled "SZ" with a form for user history. The form includes the following elements:

- Opciones:** A button labeled "Atrás".
- Nombre:** A text input field.
- BB:** A dropdown menu with the value "0" selected.
- AB:** A dropdown menu with the value "0" selected.
- Buttons:** "Calcular" and "Guardar" are located below the input fields. "Borrar", "Eliminar", and "Actualizar" are located at the bottom of the window.
- Table:** A table with columns labeled "Nombre", "BB", "AB", "Resultado", and "Comentario". The table is currently empty.
- Labels:** "Resultado" and "Comentario" are labels for the bottom sections of the form.

Tabla 87: Historia de usuario 45 "Mostrar información BB% "

Historia de usuario	
Número: 45	Nombre: Mostrar información BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario mostrar la información guardada del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber guardado la información del cálculo estadístico y a su vez después de haber presionado el botón de "Actualizar".	
Interfaz	

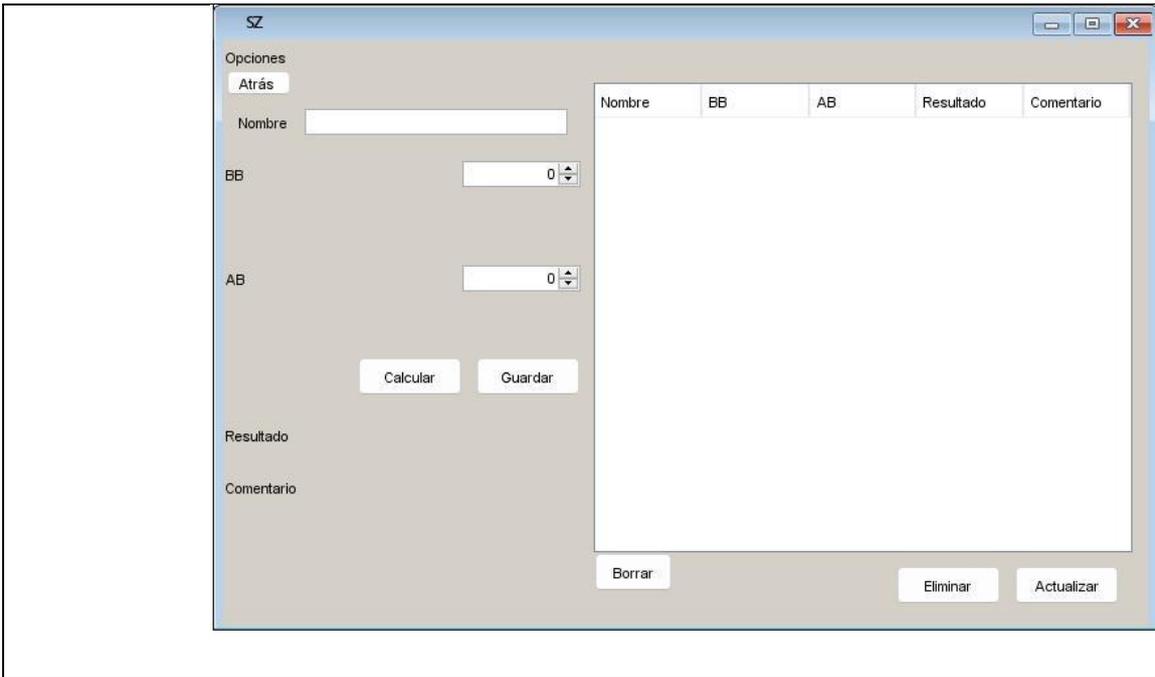


Tabla 88: Historia de usuario 46 "Limpiar información BB% "

Historia de usuario	
Número: 46	Nombre: Limpiar información BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario limpiar los campos editables para introducir nuevos datos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido información del en los campos editables y a su vez después de haber presionado el botón con un ícono que indica la acción de de "Borrar".	
Interfaz	

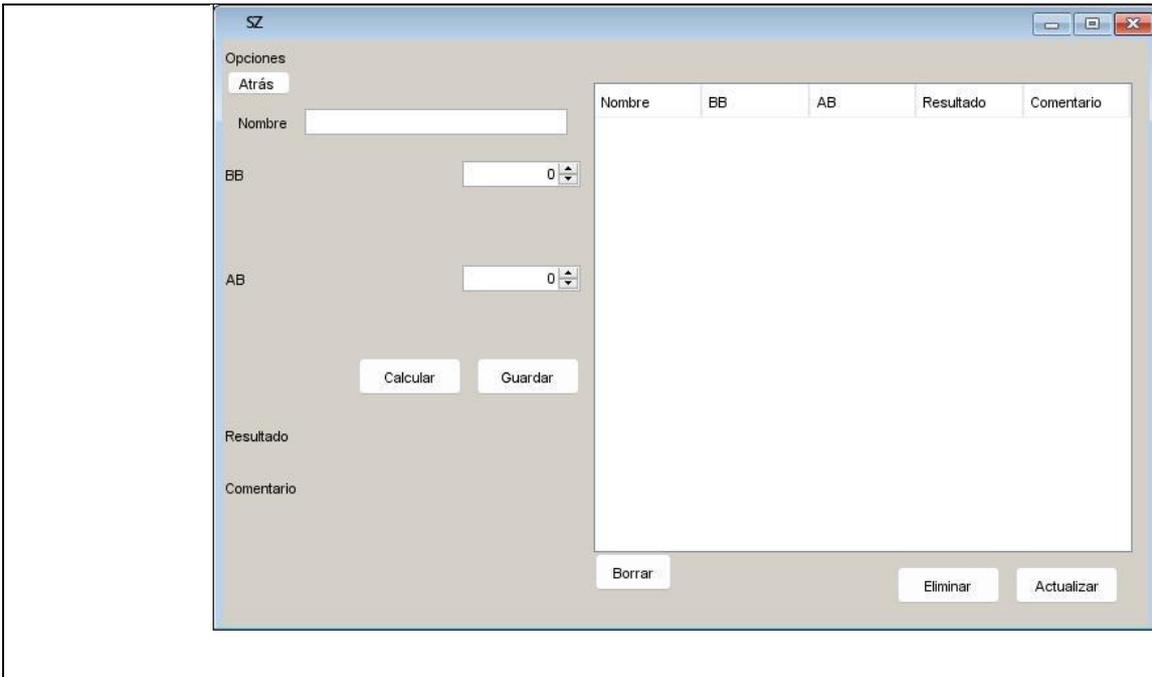


Tabla 89: Historia de usuario 47 "Visualizar resultado BB% "

Historia de usuario	
Número: 47	Nombre: Visualizar resultado BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar el resultado del cálculo estadístico que se realizó.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

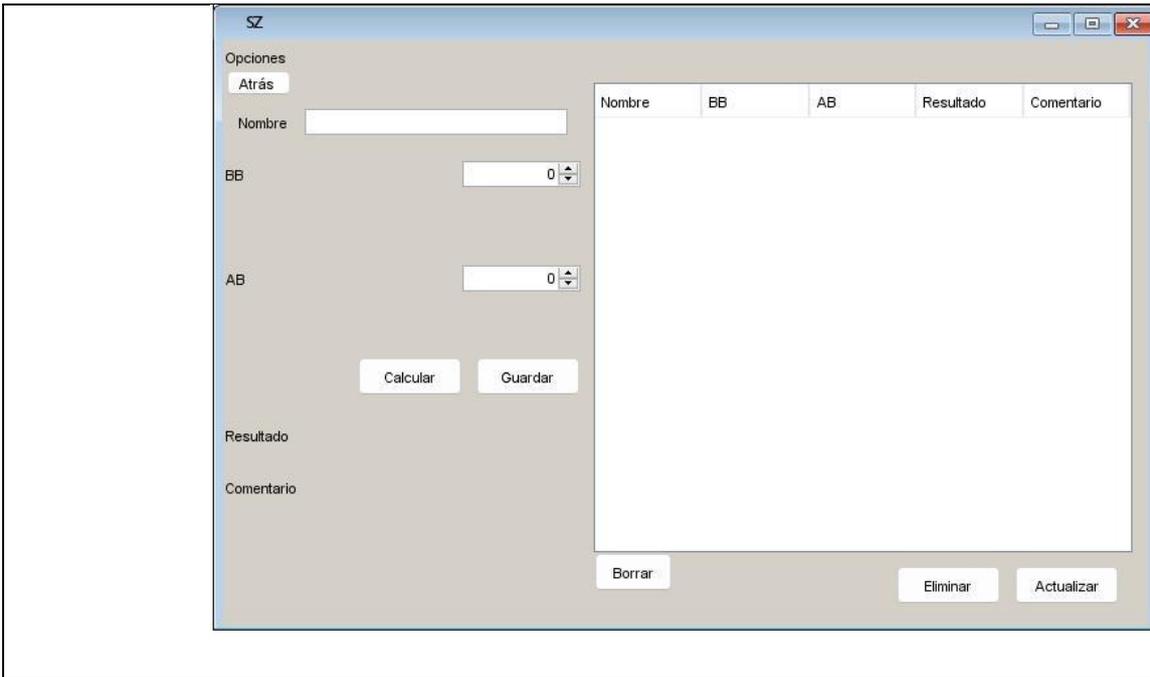


Tabla 90: Historia de usuario 48 "Visualizar comentario BB%"

Historia de usuario	
Número: 48	Nombre: Visualizar comentario BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: <p>El componente debe permitir al usuario visualizar el comentario del cálculo estadístico que se realizó a partir del resultado.</p>	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

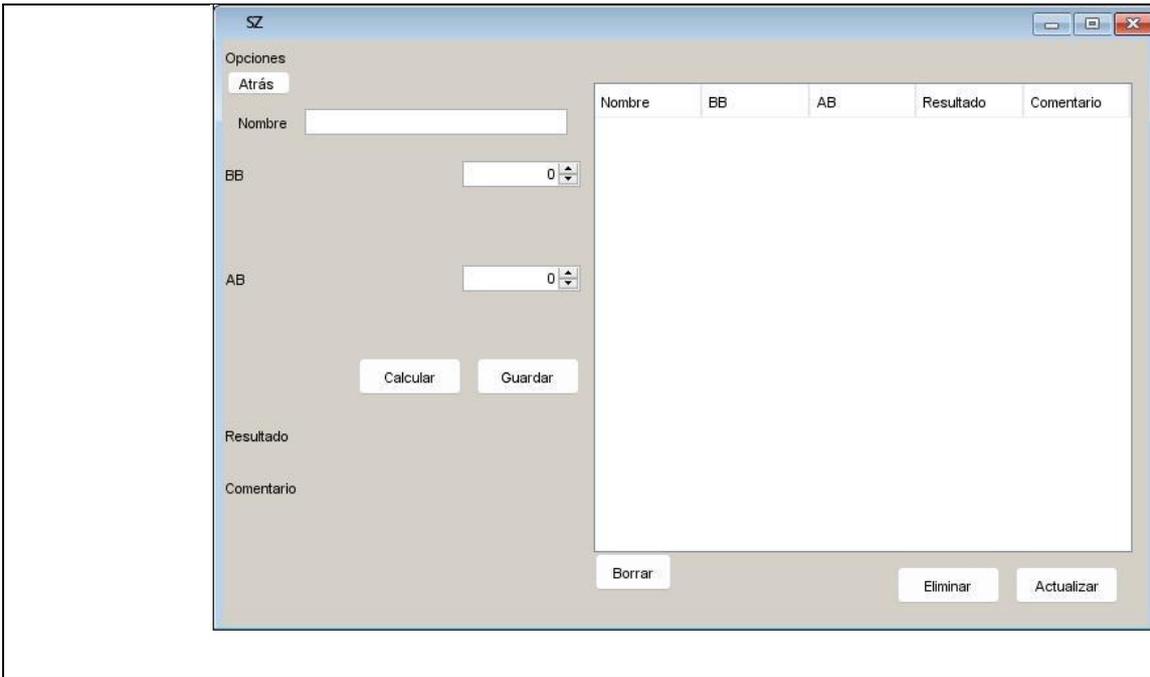


Tabla 91: Historia de usuario 49 "Eliminar BB%"

Historia de usuario	
Número: 49	Nombre: Eliminar BB%
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.3	Iteración: 2
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario eliminar la información guardada.	
Observaciones: Esta historia de usuario se ejecutará luego de que se muestre y se visualice la información del cálculo estadístico y a su vez que se seleccione uno de los mismos y se presione el botón "Eliminar".	
Interfaz	

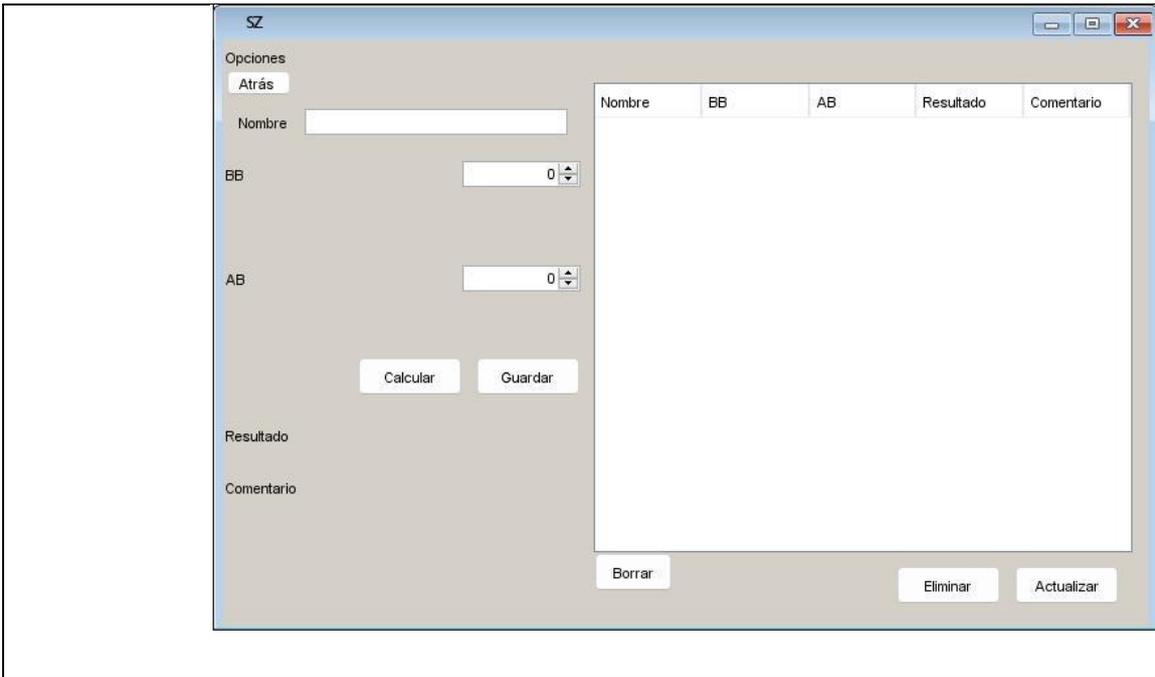


Tabla 92: Historia de usuario 7 " Visualizar información ISO "

Historia de usuario	
Número: 50	Nombre: Visualizar información ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "ISO".	
Interfaz	

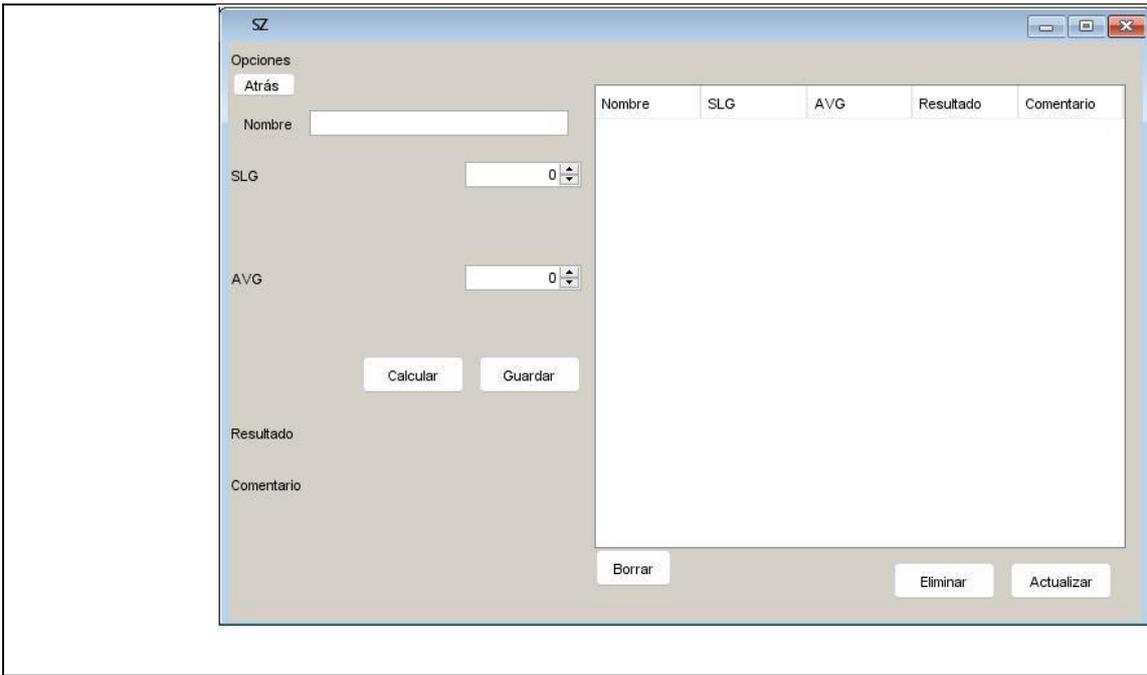


Tabla 93: Historia de usuario 51 "Ingresar información ISO"

Historia de usuario	
Número: 51	Nombre: Ingresar información ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario ingresar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "ISO".	
Interfaz	

The screenshot shows a web application window titled "SZ". On the left, there are input fields for "Nombre", "SLG", and "AVG", each with a "0" value. Below these are "Calcular" and "Guardar" buttons. On the right, there is a table with columns "Nombre", "SLG", "AVG", "Resultado", and "Comentario". At the bottom of the window, there are "Borrar", "Eliminar", and "Actualizar" buttons.

Tabla 94: Historia de usuario 52 "Calcular datos ISO"

Historia de usuario	
Número: 52	Nombre: Calcular datos ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: <p>El componente debe permitir al usuario realizar el cálculo del estadístico a partir de los datos introducidos.</p>	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido la información del cálculo en el sistema y a su vez después de haber presionado el botón de "Calcular".	
Interfaz	

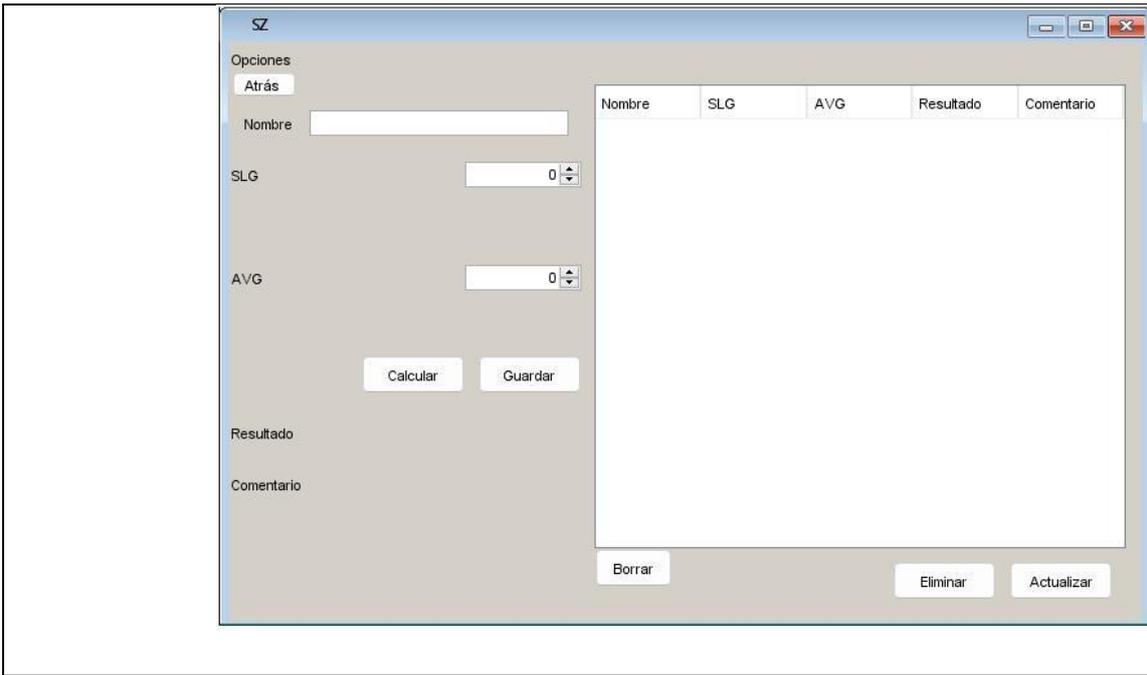
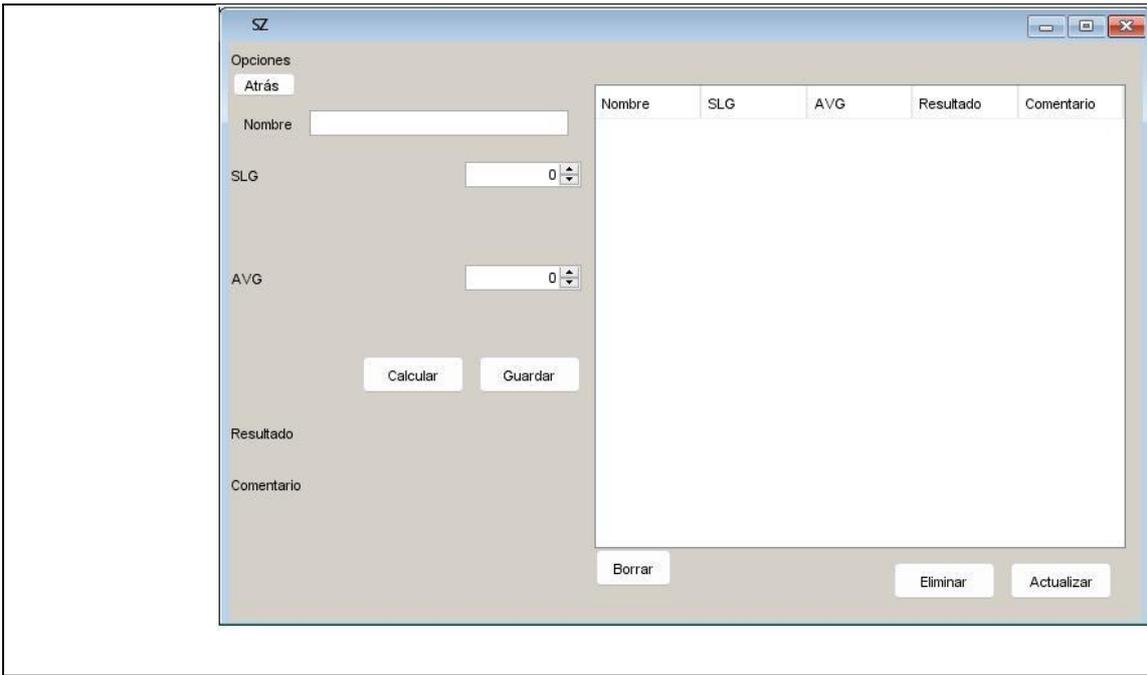


Tabla 95: Historia de usuario 53 "Guardar información ISO"

Historia de usuario	
Número: 53	Nombre: Guardar información ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario guardar la información del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo y a su vez después de haber presionado el botón de "Guardar".	
Interfaz	



Historia de usuario	
Número: 54	Nombre: Mostrar información ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario mostrar la información guardada del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber guardado la información del cálculo estadístico y a su vez después de haber presionado el botón de “Actualizar”.	
Interfaz	

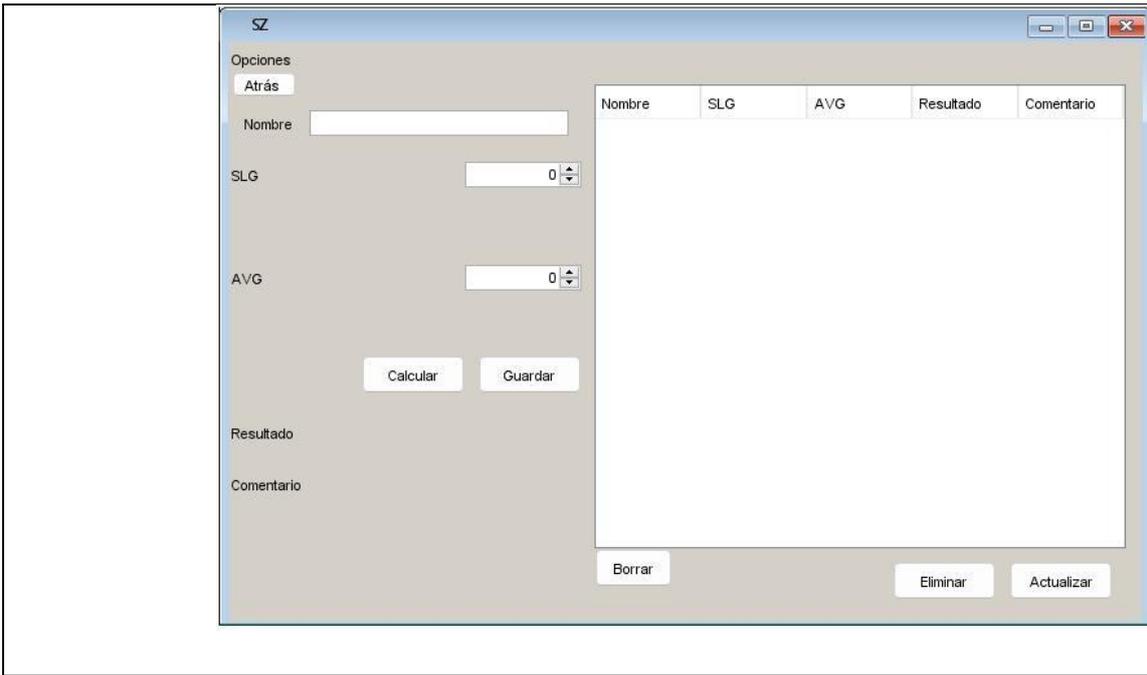


Tabla 96: Historia de usuario 55 "Limpiar información ISO"

Historia de usuario	
Número: 55	Nombre: Limpiar información ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario limpiar los campos editables para introducir nuevos datos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido información del en los campos editables y a su vez después de haber presionado el botón con un ícono que indica la acción de de "Borrar".	
Interfaz	

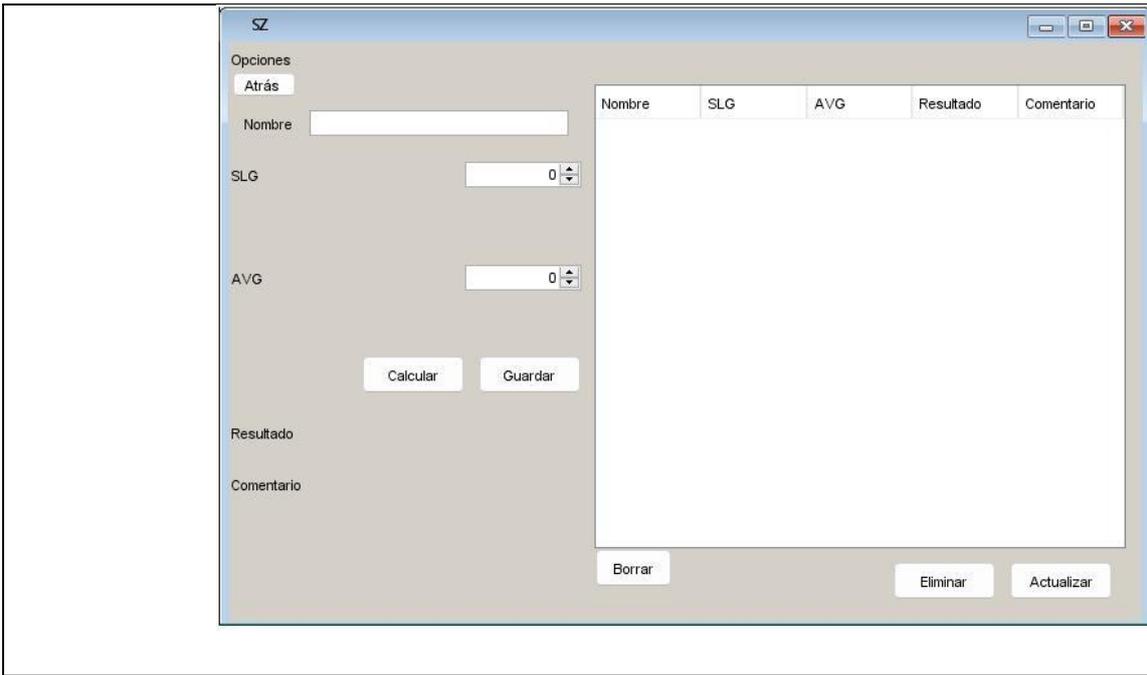


Tabla 97: Historia de usuario 56 "Visualizar resultado ISO"

Historia de usuario	
Número: 56	Nombre: Visualizar resultado ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar el resultado del cálculo estadístico que se realizó.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

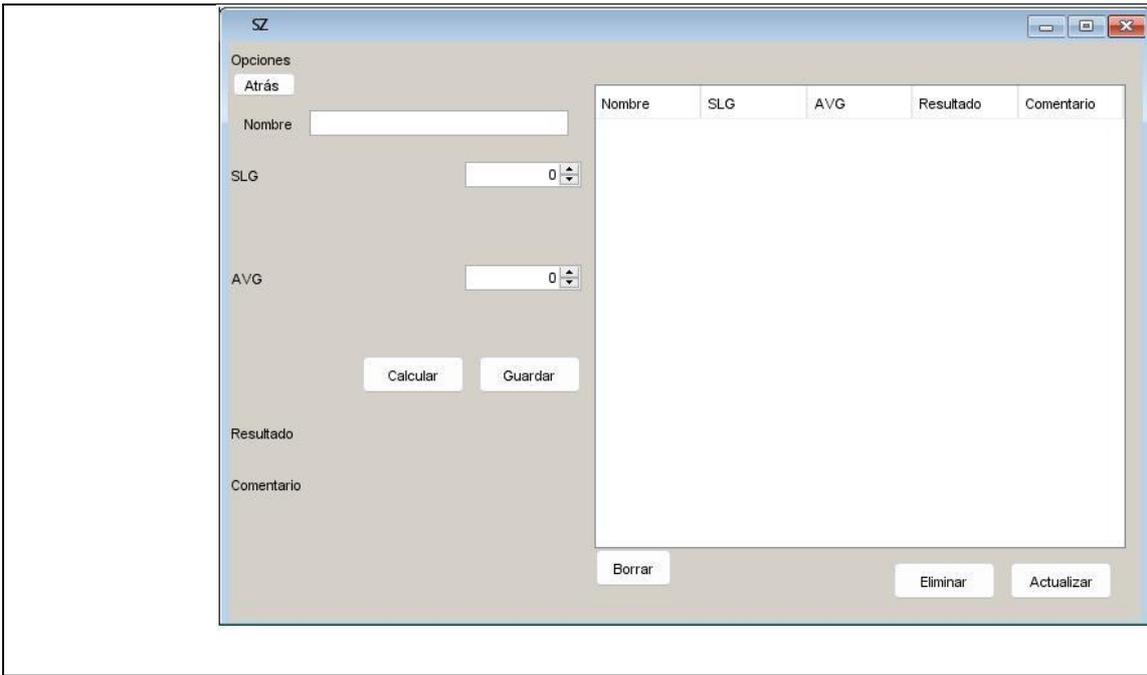
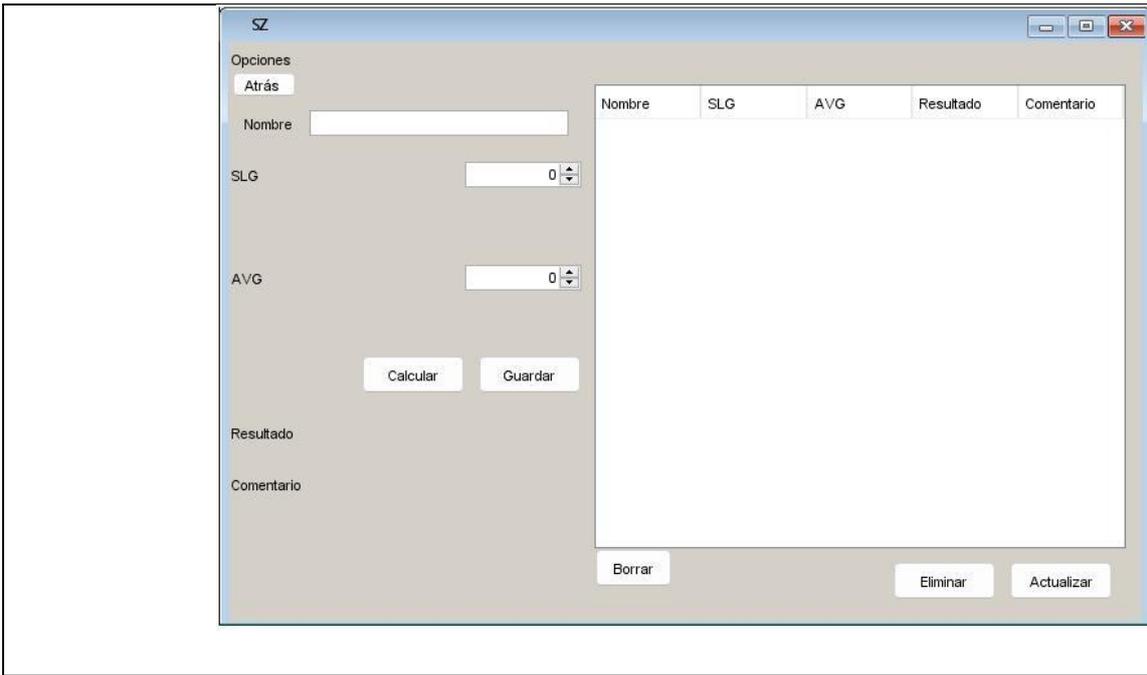


Tabla 98: Historia de usuario 57 "Visualizar comentario ISO"

Historia de usuario	
Número: 57	Nombre: Visualizar comentario ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: <p>El componente debe permitir al usuario visualizar el comentario del cálculo estadístico que se realizó a partir del resultado.</p>	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	



Historia de usuario	
Número: 58	Nombre: Eliminar ISO
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.3	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario eliminar la información guardada.	
Observaciones: Esta historia de usuario se ejecutará luego de que se muestre y se visualice la información del cálculo estadístico y a su vez que se seleccione uno de los mismos y se presione el botón "Eliminar".	
Interfaz	

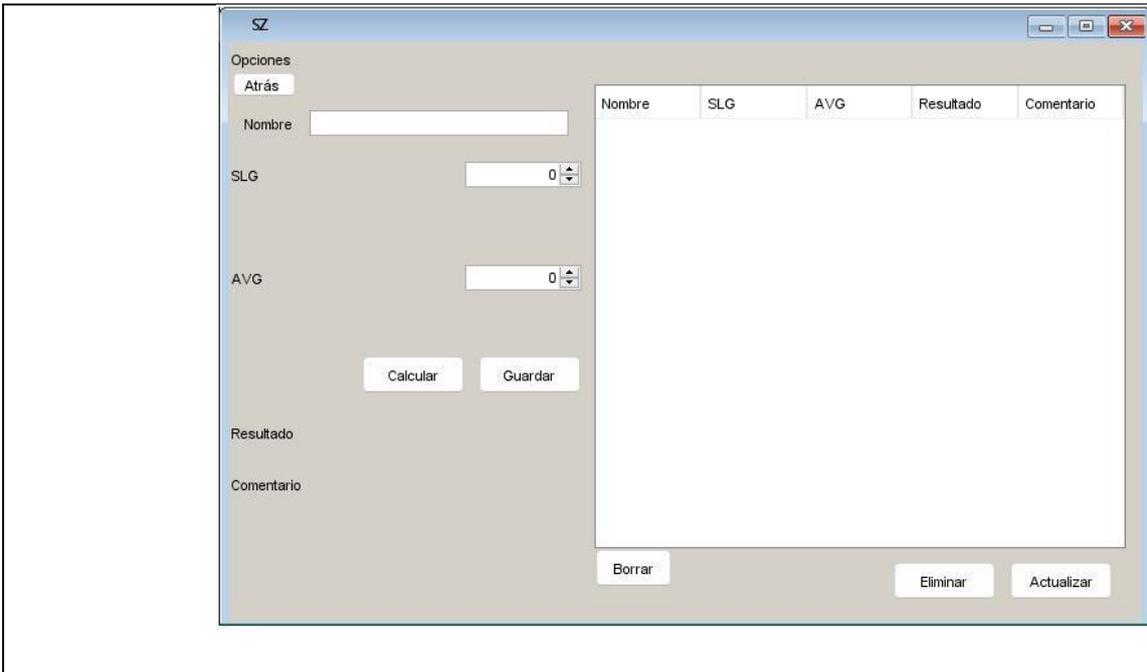


Tabla 99: Historia de usuario 8 "Administrar RC"

Historia de usuario	
Número: 59	Nombre: Visualizar información RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "RC".	
Interfaz	

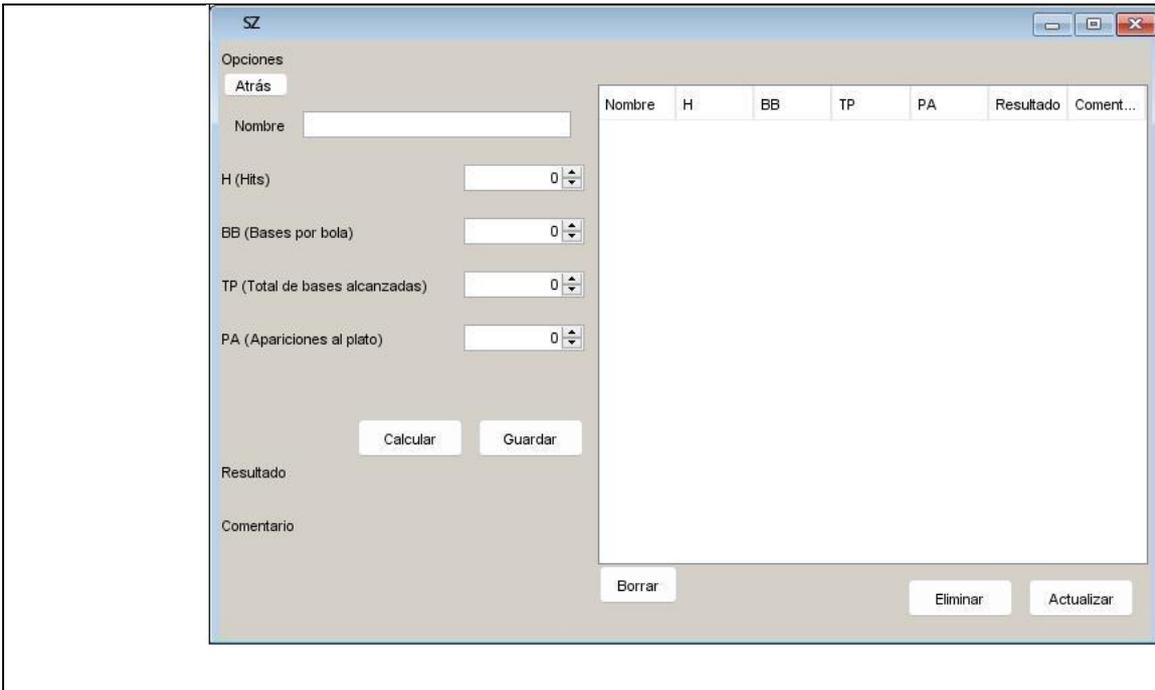


Tabla 100: Historia de usuario 60 "Ingresar información RC "

Historia de usuario	
Número: 60	Nombre: Ingresar información RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.4	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario ingresar la información de los cálculos de un deportista	
Observaciones: Esta historia de usuario se ejecutará luego de que en la ventana principal se seleccione a un deportista y a su vez se presione el botón de "RC".	
Interfaz	

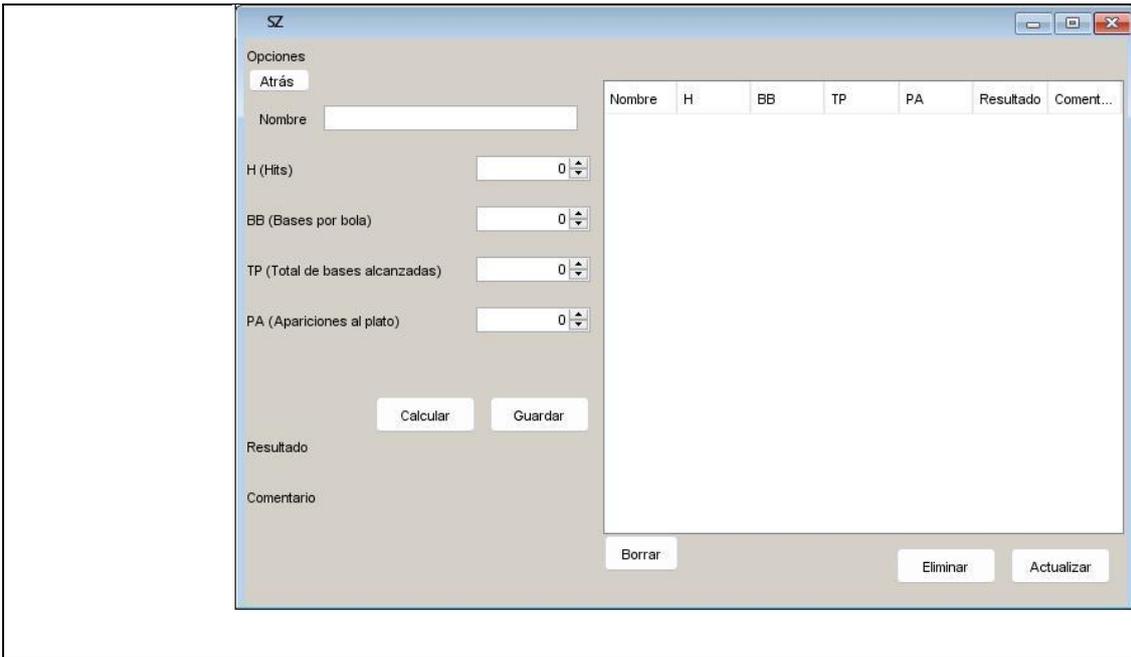


Tabla 101: Historia de usuario 61 "Calcular datos RC"

Historia de usuario	
Número: 61	Nombre: Calcular datos RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: <p>El componente debe permitir al usuario realizar el cálculo del estadístico a partir de los datos introducidos.</p>	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido la información del cálculo en el sistema y a su vez después de haber presionado el botón de "Calcular".	
Interfaz	

The screenshot shows a software window titled "SZ" with a light blue header. On the left, there is a section labeled "Opciones" containing a "Atrás" button and a "Nombre" input field. Below this are four spinners for "H (Hits)", "BB (Bases por bola)", "TP (Total de bases alcanzadas)", and "PA (Apariciones al plato)", each with a "0" value. At the bottom of this section are "Calcular" and "Guardar" buttons. Below the spinners are labels for "Resultado" and "Comentario". On the right side of the window is a table with columns: "Nombre", "H", "BB", "TP", "PA", "Resultado", and "Coment...". At the bottom right of the window are "Borrar", "Eliminar", and "Actualizar" buttons.

Tabla 102: Historia de usuario 62 "Guardar información RC"

Historia de usuario	
Número: 62	Nombre: Guardar información RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario guardar la información del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo y a su vez después de haber presionado el botón de "Guardar".	
Interfaz	

Tabla 103: historia de usuario 63 "Mostrar información RC"

Historia de usuario	
Número: 63	Nombre: Mostrar información RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario mostrar la información guardada del cálculo estadístico.	
Observaciones: Esta historia de usuario se ejecutará luego de haber guardado la información del cálculo estadístico y a su vez después de haber presionado el botón de "Actualizar".	
Interfaz	

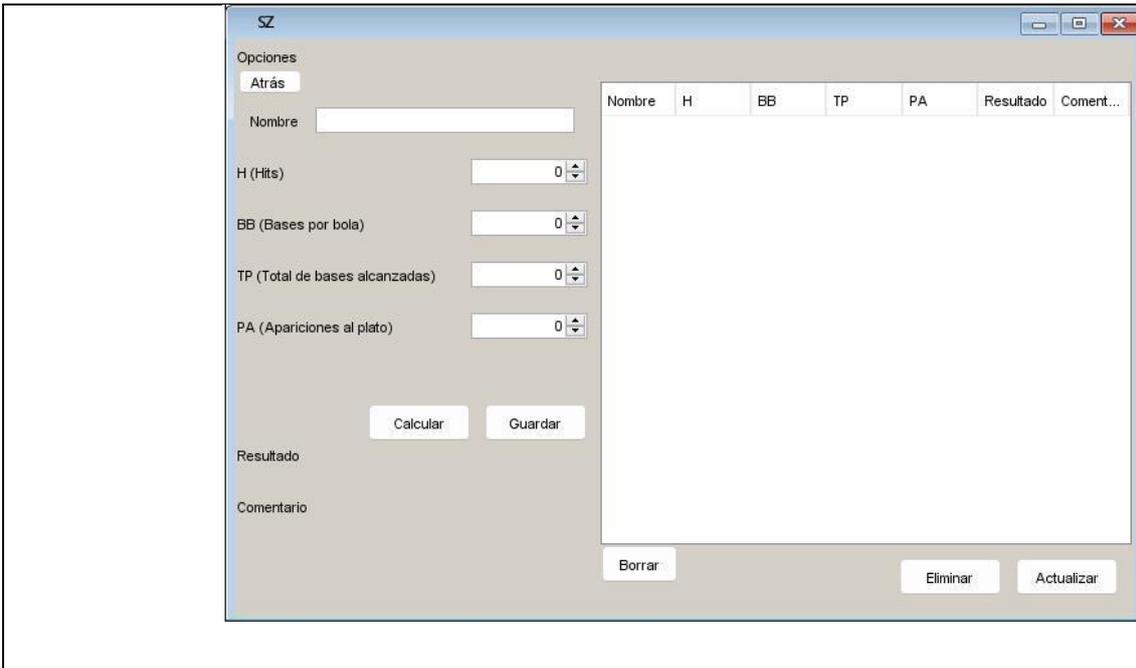


Tabla 104: Historia de usuario 64 "Limpiar información RC "

Historia de usuario	
Número: 64	Nombre: Limpiar información RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario limpiar los campos editables para introducir nuevos datos.	
Observaciones: Esta historia de usuario se ejecutará luego de haber introducido información del en los campos editables y a su vez después de haber presionado el botón con un ícono que indica la acción de de "Borrar".	
Interfaz	

Tabla 105: Historia de usuario 65 "Visualizar resultado RC"

Historia de usuario	
Número: 65	Nombre: Visualizar resultado RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar el resultado del cálculo estadístico que se realizó.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

Tabla 106: Historia de usuario 66 "Visualizar comentario RC"

Historia de usuario	
Número: 66	Nombre: Visualizar comentario RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario visualizar el comentario del cálculo estadístico que se realizó a partir del resultado.	
Observaciones: Esta historia de usuario se ejecutará luego de haber realizado el cálculo estadístico.	
Interfaz	

Tabla 107: Historia de usuario 67 "Eliminar RC"

Historia de usuario	
Número: 67	Nombre: Eliminar RC
Usuario: usuario	
Prioridad: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.3	Iteración: 3
Programador a cargo: Julio Cesar Betancourt Ramos	
Descripción: El componente debe permitir al usuario eliminar la información guardada.	
Observaciones: Esta historia de usuario se ejecutará luego de que se muestre y se visualice la información del cálculo estadístico y a su vez que se seleccione uno de los mismos y se presione el botón "Eliminar".	
Interfaz	

A.3 Tareas de Ingeniería

Iteración 2

Tabla 108: Tarea de ingeniería 5 " Implementar Visualizar información OBP "

Tarea de Ingeniería	
Número de tarea: 5	Número de historia: 2
Nombre de la tarea: Implementar Visualizar información OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar en pantalla los datos a calcular para obtener el OBP.	

Tabla 109: tarea de ingeniería 6 "Implementar Ingresar información OBP"

Tarea de Ingeniería	
Número de tarea: 6	Número de historia: 2

Nombre de la tarea: Implementar Ingresar información OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita ingresar la información necesaria para calcular el OBP.	

Tabla 110: Tarea de ingeniería 7 " Implementar Calcular datos OBP "

Tarea de Ingeniería	
Número de tarea: 7	Número de historia: 2
Nombre de la tarea: Implementar Calcular datos OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita calcular el OBP a partir de los datos introducidos por el usuario.	

Tabla 111: Tarea de ingeniería 8 "Implementar Guardar información OBP"

Tarea de Ingeniería	
Número de tarea: 8	Número de historia: 2
Nombre de la tarea: Implementar Guardar información OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita guardar la información que se recoge después de haber realizado el cálculo del OBP.	

Tabla 112: Tarea de ingeniería 9 "Implementar Mostrar información OBP"

Tarea de Ingeniería	
Número de tarea: 9	Número de historia: 2
Nombre de la tarea: Implementar Mostrar información OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar la información del cálculo guardada en la base de datos.	

Tabla 113: Tarea de ingeniería 10 "Implementar Limpiar campos OBP"

Tarea de Ingeniería	
Número de tarea: 10	Número de historia: 2
Nombre de la tarea: Implementar Limpiar campos OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita limpiar los campos editables que fueron llenados previamente al cálculo del OBP.	

Tabla 114: Tarea de ingeniería 11 "Implementar Visualizar resultado OBP"

Tarea de Ingeniería	
Número de tarea: 11	Número de historia: 2
Nombre de la tarea: Implementar Visualizar resultado OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el resultado del cálculo que se realizó de manera previa.	

Tabla 115: tarea de ingeniería 12 "Implementar Visualizar comentario OBP"

Tarea de Ingeniería	
Número de tarea: 12	Número de historia: 2
Nombre de la tarea: Implementar Visualizar comentario OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el comentario del cálculo que se realizó de manera previa.	

Tabla 116: Tarea de ingeniería 13 "Implementar Eliminar OBP"

Tarea de Ingeniería	
Número de tarea: 13	Número de historia: 2
Nombre de la tarea: Implementar Eliminar OBP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita eliminar a algún deportista ya registrado en el sistema.	

Tabla 117: Tarea de ingeniería 14 "Implementar Visualizar información SLG"

Tarea de Ingeniería	
Número de tarea: 14	Número de historia: 3
Nombre de la tarea: Implementar Visualizar información SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	

Descripción: Se implementa una funcionalidad en el componente que permita mostrar en pantalla los datos a calcular para obtener el SLG.
--

Tabla 118: Tarea de ingeniería 15 "Implementar Ingresar información SLG"

Tarea de Ingeniería	
Número de tarea: 15	Número de historia: 3
Nombre de la tarea: Implementar Ingresar información SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita ingresar la información necesaria para calcular el SLG.	

Tabla 119: Tarea de ingeniería 16 "Implementar Calcular datos SLG"

Tarea de Ingeniería	
Número de tarea: 16	Número de historia: 3
Nombre de la tarea: Implementar Calcular datos SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita calcular el SLG a partir de los datos introducidos por el usuario.	

Tabla 120: Tarea de ingeniería 17 "Implementar Guardar información SLG"

Tarea de Ingeniería	
Número de tarea: 17	Número de historia: 3
Nombre de la tarea: Implementar Guardar información SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4

Programador Responsable: Julio Cesar Betancourt Ramos
Descripción: Se implementa una funcionalidad en el componente que permita guardar la información que se recoge después de haber realizado el cálculo del SLG.

Tabla 121: Tarea de ingeniería 18 "Implementar Mostrar información SLG"

Tarea de Ingeniería	
Número de tarea: 18	Número de historia: 3
Nombre de la tarea: Implementar Mostrar información SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar la información del cálculo guardada en la base de datos.	

Tabla 122: Tarea de ingeniería 19 "Implementar Limpiar campos SLG"

Tarea de Ingeniería	
Número de tarea: 19	Número de historia: 3
Nombre de la tarea: Implementar Limpiar campos SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita limpiar los campos editables que fueron llenados previamente al cálculo del SLG.	

Tabla 123: Tarea de ingeniería 20 "Implementar Visualizar resultado SLG"

Tarea de Ingeniería	
Número de tarea: 20	Número de historia: 3

Nombre de la tarea: Implementar Visualizar resultado SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el resultado del cálculo que se realizó de manera previa.	

Tabla 124: Tarea de ingeniería 21 "Implementar Visualizar comentario SLG"

Tarea de Ingeniería	
Número de tarea: 21	Número de historia: 3
Nombre de la tarea: Implementar Visualizar comentario SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el comentario del cálculo que se realizó de manera previa.	

Tabla 125: Tarea de ingeniería 22 "Implementar Eliminar SLG"

Tarea de Ingeniería	
Número de tarea: 22	Número de historia: 3
Nombre de la tarea: Implementar Eliminar SLG	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita eliminar a algún deportista ya registrado en el sistema.	

Tabla 126: Tarea de ingeniería 23 "Implementar Visualizar información OPS"

Tarea de Ingeniería	
Número de tarea: 23	Número de historia: 4
Nombre de la tarea: Implementar Visualizar información OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar en pantalla los datos a calcular para obtener el OPS.	

Tabla 127: Tarea de ingeniería 23 "Implementar Ingresar información OPS"

Tarea de Ingeniería	
Número de tarea: 23	Número de historia: 4
Nombre de la tarea: Implementar Ingresar información OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita ingresar la información necesaria para calcular el OPS.	

Tabla 128: Tarea de ingeniería 24 "Implementar Calcular datos OPS"

Tarea de Ingeniería	
Número de tarea: 24	Número de historia: 4
Nombre de la tarea: Implementar Calcular datos OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita calcular el OPS a partir de los datos introducidos por el usuario.	

Tabla 129: tarea de ingeniería 25 "Implementar Guardar información OPS"

Tarea de Ingeniería	
Número de tarea: 25	Número de historia: 4
Nombre de la tarea: Implementar Guardar información OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita guardar la información que se recoge después de haber realizado el cálculo del OPS.	

Tabla 130: Tarea de ingeniería 26 "Implementar Mostrar información OPS"

Tarea de Ingeniería	
Número de tarea: 26	Número de historia: 4
Nombre de la tarea: Implementar Mostrar información OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar la información del cálculo guardada en la base de datos.	

Tabla 131: Tarea de ingeniería 27 "Implementar Limpiar campos OPS"

Tarea de Ingeniería	
Número de tarea: 27	Número de historia: 4
Nombre de la tarea: Implementar Limpiar campos OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	

Descripción: Se implementa una funcionalidad en el componente que permita limpiar los campos editables que fueron llenados previamente al cálculo del OPS.

Tabla 132: Tarea de ingeniería 28 "Implementar Visualizar resultado OPS"

Tarea de Ingeniería	
Número de tarea: 28	Número de historia: 4
Nombre de la tarea: Implementar Visualizar resultado OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el resultado del cálculo que se realizó de manera previa.	

Tabla 133: Tarea de ingeniería 29 "Implementar Visualizar comentario OPS"

Tarea de Ingeniería	
Número de tarea: 29	Número de historia: 4
Nombre de la tarea: Implementar Visualizar comentario OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el comentario del cálculo que se realizó de manera previa.	

Tabla 134: Tarea de ingeniería 30 "Implementar Eliminar OPS"

Tarea de Ingeniería	
Número de tarea: 30	Número de historia: 4
Nombre de la tarea: Implementar Eliminar OPS	

Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita eliminar a algún deportista ya registrado en el sistema.	

Tabla 135: tarea de ingeniería 31 " Implementar Visualizar información BABIP"

Tarea de Ingeniería	
Número de tarea: 31	Número de historia: 5
Nombre de la tarea: Implementar Visualizar información BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar en pantalla los datos a calcular para obtener el BABIP.	

Tabla 136: Tarea de ingeniería 32 "Implementar Ingresar información BABIP"

Tarea de Ingeniería	
Número de tarea: 32	Número de historia: 5
Nombre de la tarea: Implementar Ingresar información BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita ingresar la información necesaria para calcular el BABIP.	

Tabla 137: tarea de ingeniería 33 "Implementar Calcular datos BABIP"

Tarea de Ingeniería

Número de tarea: 33	Número de historia: 5
Nombre de la tarea: Implementar Calcular datos BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita calcular el BABIP a partir de los datos introducidos por el usuario.	

Tabla 138: Tarea de ingeniería 34 "Implementar Guardar información BABIP"

Tarea de Ingeniería	
Número de tarea: 34	Número de historia: 5
Nombre de la tarea: Implementar Guardar información BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita guardar la información que se recoge después de haber realizado el cálculo del BABIP.	

Tabla 139: Tarea de ingeniería 35 "Implementar Mostrar información BABIP"

Tarea de Ingeniería	
Número de tarea: 35	Número de historia: 5
Nombre de la tarea: Implementar Mostrar información BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar la información del cálculo guardada en la base de datos.	

Tabla 140: Tarea de ingeniería 36 "Implementar Limpiar campos BABIP"

Tarea de Ingeniería	
Número de tarea: 36	Número de historia: 5
Nombre de la tarea: Implementar Limpiar campos BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita limpiar los campos editables que fueron llenados previamente al cálculo del BABIP.	

Tabla 141: Tarea de ingeniería 37 "Implementar Visualizar resultado BABIP"

Tarea de Ingeniería	
Número de tarea: 37	Número de historia: 5
Nombre de la tarea: Implementar Visualizar resultado BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el resultado del cálculo que se realizó de manera previa.	

Tabla 142: Tarea de ingeniería 38 " Implementar Visualizar comentario BABIP"

Tarea de Ingeniería	
Número de tarea: 38	Número de historia: 5
Nombre de la tarea: Implementar Visualizar comentario BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	

Descripción: Se implementa una funcionalidad en el componente que permita observar el comentario del cálculo que se realizó de manera previa.

Tabla 143: Tarea de ingeniería 40 "Implementar Eliminar BABIP"

Tarea de Ingeniería	
Número de tarea: 40	Número de historia: 5
Nombre de la tarea: Implementar Eliminar BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita eliminar a algún deportista ya registrado en el sistema.	

Tabla 144: Tarea de ingeniería 41 "Implementar Visualizar información BB%"

Tarea de Ingeniería	
Número de tarea: 41	Número de historia: 6
Nombre de la tarea: Implementar Visualizar información BB%	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar en pantalla los datos a calcular para obtener el BB%.	

Tabla 145: Tarea de ingeniería 42 "Implementar Ingresar información BB%"

Tarea de Ingeniería	
Número de tarea: 42	Número de historia: 6
Nombre de la tarea: Implementar Ingresar información BB%	

Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita ingresar la información necesaria para calcular el BB%.	

Tabla 146: Tarea de ingeniería 43 "Implementar Calcular datos BB%"

Tarea de Ingeniería	
Número de tarea: 43	Número de historia: 6
Nombre de la tarea: Implementar Calcular datos BB%	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita calcular el BB% a partir de los datos introducidos por el usuario.	

Tabla 147: Tarea de ingeniería 44 "Implementar Guardar información BB%"

Tarea de Ingeniería	
Número de tarea: 44	Número de historia: 6
Nombre de la tarea: Implementar Guardar información BB%	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita guardar la información que se recoge después de haber realizado el cálculo del BB%.	

Tabla 148: Tarea de ingeniería 45 "Implementar Mostrar información BB%"

Tarea de Ingeniería

Número de tarea: 45	Número de historia: 6
Nombre de la tarea: Implementar Mostrar información BB%	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar la información del cálculo guardada en la base de datos.	

Tabla 149: Tarea de ingeniería 46 "Implementar Limpiar campos BB%"

Tarea de Ingeniería	
Número de tarea: 46	Número de historia: 6
Nombre de la tarea: Implementar Limpiar campos BB%	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita limpiar los campos editables que fueron llenados previamente al cálculo del BB%.	

Tabla 150: Tarea de ingeniería 47 "Implementar Visualizar resultado BB%"

Tarea de Ingeniería	
Número de tarea: 47	Número de historia: 6
Nombre de la tarea: Implementar Visualizar resultado BB%	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el resultado del cálculo que se realizó de manera previa.	

Tabla 151: Tarea de ingeniería 48 "Implementar Visualizar comentario BB%"

Tarea de Ingeniería	
Número de tarea: 48	Número de historia: 6
Nombre de la tarea: Implementar Visualizar comentario BB%	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el comentario del cálculo que se realizó de manera previa.	

Tabla 152: Tarea de ingeniería 49 "Implementar Eliminar BABIP"

Tarea de Ingeniería	
Número de tarea: 49	Número de historia: 6
Nombre de la tarea: Implementar Eliminar BABIP	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita eliminar a algún deportista ya registrado en el sistema.	

Tabla 153: Tarea de ingeniería 49 "Implementar Visualizar información ISO"

Tarea de Ingeniería	
Número de tarea: 50	Número de historia: 7
Nombre de la tarea: Implementar Visualizar información ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	

Descripción: Se implementa una funcionalidad en el componente que permita mostrar en pantalla los datos a calcular para obtener el ISO.

Tabla 154: Tarea de ingeniería 51 "Implementar Ingresar información ISO"

Tarea de Ingeniería	
Número de tarea: 51	Número de historia: 7
Nombre de la tarea: Implementar Ingresar información ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita ingresar la información necesaria para calcular el ISO.	

Tabla 155: Tarea de ingeniería 52 "Implementar Calcular datos ISO"

Tarea de Ingeniería	
Número de tarea: 52	Número de historia: 7
Nombre de la tarea: Implementar Calcular datos ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita calcular el ISO a partir de los datos introducidos por el usuario.	

Tabla 156: Tarea de ingeniería 48 "Implementar Guardar información ISO"

Tarea de Ingeniería	
Número de tarea: 53	Número de historia: 7

Nombre de la tarea: Implementar Guardar información ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita guardar la información que se recoge después de haber realizado el cálculo del ISO.	

Tabla 157: Tarea de ingeniería 54 "Implementar Mostrar información ISO"

Tarea de Ingeniería	
Número de tarea: 54	Número de historia: 7
Nombre de la tarea: Implementar Mostrar información ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar la información del cálculo guardada en la base de datos.	

Tabla 158: Tarea de ingeniería 55 "Implementar Limpiar campos ISO"

Tarea de Ingeniería	
Número de tarea: 55	Número de historia: 7
Nombre de la tarea: Implementar Limpiar campos ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita limpiar los campos editables que fueron llenados previamente al cálculo del ISO.	

Tabla 159: Tarea de ingeniería 56 "Implementar Visualizar resultado ISO"

Tarea de Ingeniería	
Número de tarea: 56	Número de historia: 7
Nombre de la tarea: Implementar Visualizar resultado ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el resultado del cálculo que se realizó de manera previa.	

Tabla 160: Tarea de ingeniería 57 "Implementar Visualizar comentario ISO"

Tarea de Ingeniería	
Número de tarea: 57	Número de historia: 7
Nombre de la tarea: Implementar Visualizar comentario ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el comentario del cálculo que se realizó de manera previa.	

Tabla 161: Tarea de ingeniería 58 "Implementar Eliminar ISO"

Tarea de Ingeniería	
Número de tarea: 58	Número de historia: 7
Nombre de la tarea: Implementar Eliminar ISO	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita eliminar a algún deportista ya registrado en el sistema.	

Tabla 162: Tarea de ingeniería 59 "Implementar Visualizar información RC"

Tarea de Ingeniería	
Número de tarea: 59	Número de historia: 8
Nombre de la tarea: Implementar Visualizar información RC	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar en pantalla los datos a calcular para obtener el RC.	

Tabla 163: Tarea de ingeniería 60 "Implementar Ingresar información RC"

Tarea de Ingeniería	
Número de tarea: 60	Número de historia: 8
Nombre de la tarea: Implementar Ingresar información RC	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita ingresar la información necesaria para calcular el RC.	

Tabla 164: Tarea de ingeniería 61 "Implementar Calcular datos RC"

Tarea de Ingeniería	
Número de tarea: 61	Número de historia: 8
Nombre de la tarea: Implementar Calcular datos RC	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Programador Responsable: Julio Cesar Betancourt Ramos	

Descripción: Se implementa una funcionalidad en el componente que permita calcular el RC a partir de los datos introducidos por el usuario.

Tabla 165: Tarea de ingeniería 62 "Implementar Guardar información RC"

Tarea de Ingeniería	
Número de tarea: 62	Número de historia: 8
Nombre de la tarea: Implementar Guardar información RC	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita guardar la información que se recoge después de haber realizado el cálculo del RC.	

Tabla 166: Tarea de ingeniería 63 "Implementar Guardar información RC"

Tarea de Ingeniería	
Número de tarea: 63	Número de historia: 8
Nombre de la tarea: Implementar Mostrar información RC	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita mostrar la información del cálculo guardada en la base de datos.	

Tabla 167: Tarea de ingeniería 64 "Implementar Limpiar campos RC"

Tarea de Ingeniería	
Número de tarea: 64	Número de historia: 8
Nombre de la tarea: Implementar Limpiar campos RC	

Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita limpiar los campos editables que fueron llenados previamente al cálculo del RC.	

Tabla 168: Tarea de ingeniería 65 "Implementar Visualizar resultado RC"

Tarea de Ingeniería	
Número de tarea: 65	Número de historia: 8
Nombre de la tarea: Implementar Visualizar resultado RC	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el resultado del cálculo que se realizó de manera previa.	

Tabla 169: Tarea de ingeniería 66 "Implementar Visualizar comentario OPS"

Tarea de Ingeniería	
Número de tarea: 66	Número de historia: 8
Nombre de la tarea: Implementar Visualizar comentario OPS	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita observar el comentario del cálculo que se realizó de manera previa.	

Tabla 170: Tarea de ingeniería 67 "Implementar Eliminar RC"

Tarea de Ingeniería

Número de tarea: 67	Número de historia: 8
Nombre de la tarea: Implementar Eliminar RC	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.4
Programador Responsable: Julio Cesar Betancourt Ramos	
Descripción: Se implementa una funcionalidad en el componente que permita eliminar a algún deportista ya registrado en el sistema.	

A.4 Casos de Prueba de Aceptación

Iteración 2

Tabla 171: Caso de prueba 24 "Administrar OPS"

Caso de prueba de aceptación	
Código: HU4_P24	Nº de Historia de Usuario: 4
Historia de usuario: Administrar OPS	
Descripción: Trabaja con las funcionalidades principales para realizar el cálculo del OPS	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar las ventanas de registrar o modificar. 2. Llena los campos de texto de las ventanas y le da a los botones agregar o modificar. 3. Se muestra la pantalla principal con la información del deportista luego de tocar el botón modificar. 4. Selecciona un deportista de la tabla y toca el botón eliminar. 5. Se muestra la tabla sin el usuario seleccionado. 	
Resultado: Satisfactorio	

Tabla 172: Caso de prueba 25 "Visualizar OPS"

Caso de prueba de aceptación	
Código: HU4_P25	Nº de Historia de Usuario: 4
Historia de usuario: Visualizar OPS	
Descripción: Muestra en pantalla la ventana OPS	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar a un deportista. 2. Toca el botón "OPS" que abre la ventana para el cálculo del mismo. 3. Se muestra la ventana OPS. 	
Resultado: Satisfactorio	

Tabla 173: Caso de prueba 26 "Ingresar Información OPS"

Caso de prueba de aceptación	
Código: HU4_P26	Nº de Historia de Usuario: 4
Historia de usuario: Ingresar Información OPS	
Descripción: Permite ingresar la información necesaria para realizar el cálculo del OPS	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe llenar los campos vacíos editables en la ventana. 	
Resultado: Satisfactorio	

Tabla 174: caso de prueba 27 "Ingresar Información OPS"

Caso de prueba de aceptación	
Código: HU4_P27	Nº de Historia de Usuario: 4
Historia de usuario: Calcular datos OPS	
Descripción: Permite realizar cálculos pertinentes para obtener el OPS	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo OPS. 	
Resultado: Satisfactorio	

Tabla 175: Caso de prueba 28 "Guardar información OPS"

Caso de prueba de aceptación	
Código: HU4_P28	Nº de Historia de Usuario: 4
Historia de usuario: Guardar información OPS	
Descripción: Permite guardar la información de OPS	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo OPS. 	
Resultado: Satisfactorio	

Tabla 176: Caso de prueba 29 "Mostrar información OPS"

Caso de prueba de aceptación	
Código: HU4_P29	Nº de Historia de Usuario: 4
Historia de usuario: Mostrar información OPS	
Descripción: Permite mostrar en pantalla la información de OPS	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Muestra la información en cuanto se abre la ventana OPS. 2. Toca el botón "Actualizar" para mostrar la información que no se ha mostrado aún. 	
Resultado: Satisfactorio	

Tabla 177: Caso de prueba 30 "Limpiar campos OPS"

Caso de prueba de aceptación	
Código: HU2_P30	Nº de Historia de Usuario: 4
Historia de usuario: Limpiar campos OPS	
Descripción: Permite limpiar los datos que introdujo el usuario al sistema.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Limpiar" y pone los campos en blanco. 	
Resultado: Satisfactorio	

Tabla 178: Caso de prueba 31 "Visualizar resultado OPS"

Caso de prueba de aceptación	
Código: HU2_P31	Nº de Historia de Usuario: 4
Historia de usuario: Visualizar resultado OPS	
Descripción: Permite visualizar el resultado del cálculo realizado.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 	
Resultado: Satisfactorio	

Tabla 179: Caso de prueba 32 "Visualizar comentario OPS"

Caso de prueba de aceptación	
Código: HU2_P32	Nº de Historia de Usuario: 4
Historia de usuario: Visualizar comentario OPS	
Descripción: Permite visualizar el comentario según el resultado que da el cálculo.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 4. Muestra el comentario de dicho resultado. 	
Resultado: Satisfactorio	

Tabla 180: Caso de prueba 33 "Eliminar OPS"

Caso de prueba de aceptación	
Código: HU1_P33	Historia de Usuario: 4
Nombre: Eliminar OPS	
Descripción: Trabaja con las funcionalidades principales para eliminar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se muestra la ventana "OPS". 2. Selecciona al deportista. 3. Toca el botón "Eliminar" 4. Se muestra la ventana OPS con el deportista eliminado. 	
Resultado: Satisfactorio	

Tabla 181: Caso de prueba 34 "Visualizar BABIP"

Caso de prueba de aceptación	
Código: HU5_P34	Nº de Historia de Usuario: 5
Historia de usuario: Visualizar BABIP	
Descripción: Muestra en pantalla la ventana BABIP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar a un deportista. 2. Toca el botón "BABIP" que abre la ventana para el cálculo del mismo. 3. Se muestra la ventana BABIP. 	
Resultado: Satisfactorio	

Tabla 182: Caso de prueba 35 "Ingresar Información BABIP"

Caso de prueba de aceptación	
Código: HU5_P35	Nº de Historia de Usuario: 5
Historia de usuario: Ingresar Información BABIP	
Descripción: Permite ingresar la información necesaria para realizar el cálculo del BABIP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Debe llenar los campos vacíos editables en la ventana. 	
Resultado: Satisfactorio	

Tabla 183: Caso de prueba 36 "Calcular datos BABIP"

Caso de prueba de aceptación	
Código: HU5_P36	Nº de Historia de Usuario: 5
Historia de usuario: Calcular datos BABIP	
Descripción: Permite realizar cálculos pertinentes para obtener el BABIP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo BABIP. 	
Resultado: Satisfactorio	

Tabla 184: Caso de prueba 37 "Guardar información BABIP"

Caso de prueba de aceptación	
Código: HU5_P37	Nº de Historia de Usuario: 5
Historia de usuario: Guardar información BABIP	
Descripción: Permite guardar la información de BABIP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo BABIP. 	
Resultado: Satisfactorio	

Tabla 185: Caso de prueba 38 "Mostrar información BABIP"

Caso de prueba de aceptación	
Código: HU5_P38	Nº de Historia de Usuario: 5
Historia de usuario: Mostrar información BABIP	
Descripción: Permite mostrar en pantalla la información de BABIP	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Muestra la información en cuanto se abre la ventana BABIP. 2. Toca el botón "Actualizar" para mostrar la información que no se ha mostrado aún. 	
Resultado: Satisfactorio	

Tabla 186: Caso de prueba 39 "Limpiar campos BABIP"

Caso de prueba de aceptación	
Código: HU5_P39	Nº de Historia de Usuario: 5
Historia de usuario: Limpiar campos BABIP	
Descripción: Permite limpiar los datos que introdujo el usuario al sistema.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Limpiar" y pone los campos en blanco. 	
Resultado: Satisfactorio	

Tabla 187: Caso de prueba 40 "Visualizar resultado BABIP"

Caso de prueba de aceptación	
Código: HU5_P40	Nº de Historia de Usuario: 5
Historia de usuario: Visualizar resultado BABIP	
Descripción: Permite visualizar el resultado del cálculo realizado.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 	
Resultado: Satisfactorio	

Tabla 188: Caso de prueba 41 "Visualizar comentario BABIP"

Caso de prueba de aceptación	
Código: HU5_P41	Nº de Historia de Usuario: 5
Historia de usuario: Visualizar comentario BABIP	
Descripción: Permite visualizar el comentario según el resultado que da el cálculo.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 5. El usuario llena los campos editables mostrados en pantalla. 6. Toca el botón "Calcular". 7. Muestra el resultado del cálculo. 1. Muestra el comentario de dicho resultado. 	
Resultado: Satisfactorio	

Tabla 189: Caso de prueba 42 "Eliminar BABIP"

Caso de prueba de aceptación	
Código: HU1_P42	Historia de Usuario: 5
Nombre: Eliminar BABIP	
Descripción: Trabaja con las funcionalidades principales para eliminar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se muestra la ventana "BABIP". 2. Selecciona al deportista. 3. Toca el botón "Eliminar" 4. Se muestra la ventana BABIP con el deportista eliminado. 	
Resultado: Satisfactorio	

Tabla 190: Caso de prueba 43 "Administrar BB%"

Caso de prueba de aceptación	
Código: HU6_P43	Nº de Historia de Usuario: 6
Historia de usuario: Administrar BB%	
Descripción: Trabaja con las funcionalidades principales para realizar el cálculo del BB%	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar las ventanas de registrar o modificar. 2. Llena los campos de texto de las ventanas y le da a los botones agregar o modificar. 3. Se muestra la pantalla principal con la información del deportista luego de tocar el botón modificar. 4. Selecciona un deportista de la tabla y toca el botón eliminar. 5. Se muestra la tabla sin el usuario seleccionado. 	
Resultado: Satisfactorio	

Tabla 191: Caso de prueba 44 "Visualizar BB%"

Caso de prueba de aceptación	
Código: HU6_P44	Nº de Historia de Usuario: 6
Historia de usuario: Visualizar BB%	
Descripción: Muestra en pantalla la ventana BB%	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar a un deportista. 	

<ol style="list-style-type: none"> 2. Toca el botón "BB%" que abre la ventana para el cálculo del mismo. 3. Se muestra la ventana BB%.
Resultado: Satisfactorio

Tabla 192: Caso de prueba 45 "Ingresar Información BB%"

Caso de prueba de aceptación	
Código: HU6_P45	Nº de Historia de Usuario: 6
Historia de usuario: Ingresar Información BB%	
Descripción: Permite ingresar la información necesaria para realizar el cálculo del BB%	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Debe llenar los campos vacíos editables en la ventana. 	
Resultado: Satisfactorio	

Tabla 193: Caso de prueba 46 "Calcular datos BB%"

Caso de prueba de aceptación	
Código: HU6_P46	Nº de Historia de Usuario: 6
Historia de usuario: Calcular datos BB%	
Descripción: Permite realizar cálculos pertinentes para obtener el BB%	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 	

3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo BB%.
Resultado: Satisfactorio

Tabla 194: caso de prueba 47 "Guardar información BB%"

Caso de prueba de aceptación	
Código: HU6_P47	Nº de Historia de Usuario: 6
Historia de usuario: Guardar información BB%	
Descripción: Permite guardar la información de BB%	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo BB%. 	
Resultado: Satisfactorio	

Tabla 195: caso de prueba 48 "Mostrar información BB%"

Caso de prueba de aceptación	
Código: HU6_P48	Nº de Historia de Usuario: 6
Historia de usuario: Mostrar información BB%	
Descripción: Permite mostrar en pantalla la información de BB%	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	

<ol style="list-style-type: none"> 1. Muestra la información en cuanto se abre la ventana OBP. 2. Toca el botón "Actualizar" para mostrar la información que no se ha mostrado aún.
Resultado: Satisfactorio

Tabla 196: Caso de prueba 49 "Limpiar campos BB%"

Caso de prueba de aceptación	
Código: HU6_P49	Nº de Historia de Usuario: 6
Historia de usuario: Limpiar campos BB%	
Descripción: Permite limpiar los datos que introdujo el usuario al sistema.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Limpiar" y pone los campos en blanco. 	
Resultado: Satisfactorio	

Tabla 197: Caso de prueba 50 "Visualizar resultado BB%"

Caso de prueba de aceptación	
Código: HU6_P50	Nº de Historia de Usuario: 6
Historia de usuario: Visualizar resultado BB%	
Descripción: Permite visualizar el resultado del cálculo realizado.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 	

<ol style="list-style-type: none"> 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo.
Resultado: Satisfactorio

Tabla 198: Caso de prueba 51 "Visualizar comentario BB%"

Caso de prueba de aceptación	
Código: HU6_P51	Nº de Historia de Usuario: 6
Historia de usuario: Visualizar comentario BB%	
Descripción: Permite visualizar el comentario según el resultado que da el cálculo.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 8. El usuario llena los campos editables mostrados en pantalla. 9. Toca el botón "Calcular". 10. Muestra el resultado del cálculo. 1. Muestra el comentario de dicho resultado. 	
Resultado: Satisfactorio	

Tabla 199: Caso de prueba 52 "Eliminar BB%"

Caso de prueba de aceptación	
Código: HU1_P52	Historia de Usuario: 6
Nombre: Eliminar BB%	
Descripción: Trabaja con las funcionalidades principales para eliminar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Se muestra la ventana "BB%". 	

<ol style="list-style-type: none"> 2. Selecciona al deportista. 3. Toca el botón "Eliminar" 4. Se muestra la ventana BB% con el deportista eliminado.
Resultado: Satisfactorio

Iteración 3

Tabla 200: Caso de prueba 53 "Administrar ISO"

Caso de prueba de aceptación	
Código: HU7_P53	Nº de Historia de Usuario: 7
Historia de usuario: Administrar ISO	
Descripción: Trabaja con las funcionalidades principales para realizar el cálculo del ISO	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar las ventanas de registrar o modificar. 2. Llena los campos de texto de las ventanas y le da a los botones agregar o modificar. 3. Se muestra la pantalla principal con la información del deportista luego de tocar el botón modificar. 4. Selecciona un deportista de la tabla y toca el botón eliminar. 5. Se muestra la tabla sin el usuario seleccionado. 	
Resultado: Satisfactorio	

Tabla 201: Caso de prueba 54 "Visualizar ISO"

Caso de prueba de aceptación	
Código: HU7_P54	Nº de Historia de Usuario: 7

Historia de usuario: Visualizar ISO
Descripción: Muestra en pantalla la ventana ISO
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar a un deportista. 2. Toca el botón "ISO" que abre la ventana para el cálculo del mismo. 3. Se muestra la ventana ISO.
Resultado: Satisfactorio

Tabla 202: Caso de prueba 55 "Ingresar Información ISO"

Caso de prueba de aceptación	
Código: HU7_P55	Nº de Historia de Usuario: 7
Historia de usuario: Ingresar Información ISO	
Descripción: Permite ingresar la información necesaria para realizar el cálculo del ISO	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe llenar los campos vacíos editables en la ventana. 	
Resultado: Satisfactorio	

Tabla 203: Caso de prueba 56 "Calcular datos ISO"

Caso de prueba de aceptación	
Código: HU7_P56	Nº de Historia de Usuario: 7

Historia de usuario: Calcular datos ISO
Descripción: Permite realizar cálculos pertinentes para obtener el ISO
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo ISO.
Resultado: Satisfactorio

Tabla 204: Caso de prueba 57 "Guardar información ISO"

Caso de prueba de aceptación	
Código: HU7_P57	Nº de Historia de Usuario: 7
Historia de usuario: Guardar información ISO	
Descripción: Permite guardar la información de ISO	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo ISO. 	
Resultado: Satisfactorio	

Tabla 205: Caso de prueba 58 "Mostrar información ISO"

Caso de prueba de aceptación	
Código: HU7_P58	Nº de Historia de Usuario: 7
Historia de usuario: Mostrar información ISO	
Descripción: Permite mostrar en pantalla la información de ISO	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Muestra la información en cuanto se abre la ventana ISO. 2. Toca el botón "Actualizar" para mostrar la información que no se ha mostrado aún. 	
Resultado: Satisfactorio	

Tabla 206: Caso de prueba 59 "Limpiar campos ISO"

Caso de prueba de aceptación	
Código: HU7_P59	Nº de Historia de Usuario: 7
Historia de usuario: Limpiar campos ISO	
Descripción: Permite limpiar los datos que introdujo el usuario al sistema.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Limpiar" y pone los campos en blanco. 	
Resultado: Satisfactorio	

Tabla 207: Caso de prueba 60 "Visualizar resultado ISO"

Caso de prueba de aceptación	
Código: HU7_P60	Nº de Historia de Usuario: 7
Historia de usuario: Visualizar resultado ISO	
Descripción: Permite visualizar el resultado del cálculo realizado.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 	
Resultado: Satisfactorio	

Tabla 208: Caso de prueba 61 "Visualizar comentario ISO"

Caso de prueba de aceptación	
Código: HU7_P61	Nº de Historia de Usuario: 7
Historia de usuario: Visualizar comentario ISO	
Descripción: Permite visualizar el comentario según el resultado que da el cálculo.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 4. Muestra el comentario de dicho resultado. 	
Resultado: Satisfactorio	

Tabla 209: Caso de prueba 62 "Eliminar ISO"

Caso de prueba de aceptación	
Código: HU1_P62	Historia de Usuario: 7
Nombre: Eliminar ISO	
Descripción: Trabaja con las funcionalidades principales para eliminar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se muestra la ventana "ISO". 2. Selecciona al deportista. 3. Toca el botón "Eliminar" 4. Se muestra la ventana ISO con el deportista eliminado. 	
Resultado: Satisfactorio	

Tabla 210: Caso de prueba 63 "Administrar RC"

Caso de prueba de aceptación	
Código: HU8_P63	Nº de Historia de Usuario: 8
Historia de usuario: Administrar RC	
Descripción: Trabaja con las funcionalidades principales para realizar el cálculo del RC	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Debe seleccionar las ventanas de registrar o modificar. 2. Llena los campos de texto de las ventanas y le da a los botones agregar o modificar. 3. Se muestra la pantalla principal con la información del deportista luego de tocar el botón modificar. 	

<ol style="list-style-type: none"> 4. Selecciona un deportista de la tabla y toca el botón eliminar. 5. Se muestra la tabla sin el usuario seleccionado.
Resultado: Satisfactorio

Tabla 211: Caso de prueba 64 "Visualizar RC"

Caso de prueba de aceptación	
Código: HU8_P64	Nº de Historia de Usuario: 8
Historia de usuario: Visualizar RC	
Descripción: Muestra en pantalla la ventana RC	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Debe seleccionar a un deportista. 2. Toca el botón "RC" que abre la ventana para el cálculo del mismo. 3. Se muestra la ventana RC. 	
Resultado: Satisfactorio	

Tabla 212: Caso de prueba 65 "Ingresar Información RC"

Caso de prueba de aceptación	
Código: HU8_P65	Nº de Historia de Usuario: 8
Historia de usuario: Ingresar Información RC	
Descripción: Permite ingresar la información necesaria para realizar el cálculo del RC	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	

1. Debe llenar los campos vacíos editables en la ventana.
Resultado: Satisfactorio

Tabla 213: Caso de prueba 66 "Calcular datos RC"

Caso de prueba de aceptación	
Código: HU8_P66	Nº de Historia de Usuario: 8
Historia de usuario: Calcular datos RC	
Descripción: Permite realizar cálculos pertinentes para obtener el RC	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 2. Toca el botón "Calcular" para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo RC. 	
Resultado: Satisfactorio	

Tabla 214: Caso de prueba 67 "Guardar información RC"

Caso de prueba de aceptación	
Código: HU8_P67	Nº de Historia de Usuario: 8
Historia de usuario: Guardar información RC	
Descripción: Permite guardar la información de RC	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. Debe Llenar los campos vacíos editables de la ventana. 	

<ol style="list-style-type: none"> 2. Toca el botón “Calcular” para realizar el cálculo del mismo. 3. Se muestra el resultado del cálculo y el comentario que se crea a partir del mismo RC.
Resultado: Satisfactorio

Tabla 215: Caso de prueba 68 "Mostrar información RC"

Caso de prueba de aceptación	
Código: HU8_P68	Nº de Historia de Usuario: 8
Historia de usuario: Mostrar información RC	
Descripción: Permite mostrar en pantalla la información de RC	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Muestra la información en cuanto se abre la ventana RC. 2. Toca el botón “Actualizar” para mostrar la información que no se ha mostrado aún. 	
Resultado: Satisfactorio	

Tabla 216: Caso de prueba 69 "Limpiar campos RC"

Caso de prueba de aceptación	
Código: HU8_P69	Nº de Historia de Usuario: 8
Historia de usuario: Limpiar campos RC	
Descripción: Permite limpiar los datos que introdujo el usuario al sistema.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	

<ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón “Limpiar” y pone los campos en blanco.
Resultado: Satisfactorio

Tabla 217: Caso de prueba 70 "Visualizar resultado RC"

Caso de prueba de aceptación	
Código: HU8_P70	Nº de Historia de Usuario: 8
Historia de usuario: Visualizar resultado RC	
Descripción: Permite visualizar el resultado del cálculo realizado.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 2. Toca el botón “Calcular”. 3. Muestra el resultado del cálculo. 	
Resultado: Satisfactorio	

Tabla 218: Caso de prueba 71 "Visualizar comentario RC"

Caso de prueba de aceptación	
Código: HU8_P71	Nº de Historia de Usuario: 8
Historia de usuario: Visualizar comentario RC	
Descripción: Permite visualizar el comentario según el resultado que da el cálculo.	
Condiciones de Ejecución: Debe estar registrado en el sistema al menos un deportista	
Pasos de Ejecución:	
<ol style="list-style-type: none"> 1. El usuario llena los campos editables mostrados en pantalla. 	

<ol style="list-style-type: none"> 2. Toca el botón "Calcular". 3. Muestra el resultado del cálculo. 4. Muestra el comentario de dicho resultado.
Resultado: Satisfactorio

Tabla 219: Caso de prueba 72 "Eliminar RC"

Caso de prueba de aceptación	
Código: HU1_P72	Historia de Usuario: 8
Nombre: Eliminar RC	
Descripción: Trabaja con las funcionalidades principales para eliminar un deportista	
Condiciones de Ejecución: Que haya al menos un deportista registrado	
Pasos de Ejecución: <ol style="list-style-type: none"> 1. Se muestra la ventana "RC". 2. Selecciona al deportista. 3. Toca el botón "Eliminar" 4. Se muestra la ventana RC con el deportista eliminado. 	
Resultado: Satisfactorio	

A.5 Entrevista

Entrevista a: _____ con el objetivo de obtener información y conocimientos sobre la sabermetría en cuba.

Preguntas:

- 1- ¿Qué es la sabermetría en cuba?
- 2- ¿Qué aspectos en los deportistas cubanos, específicamente en el béisbol, analiza la sabermetría?
- 3- ¿Cuáles son las principales variables que se utilizan en Cuba para el análisis del rendimiento de los jugadores de béisbol?
- 4- ¿Existe algún sistema cubano que recoja dicha información y la almacene?

- 5- ¿Cree usted que el béisbol cubano mejoraría con las anotaciones que recogería dicho sistema? ¿Por qué?
- 6- ¿Cuáles cree usted que son las principales características de los jugadores de béisbol en cubanos?
- 7- ¿Qué factores medioambientales cree usted que son los que intervienen en el desarrollo de los jugadores de béisbol cubano?