



SISTEMA INFORMÁTICO DE TÉCNICAS DE PROCESAMIENTO DE DATOS CUANTITATIVOS

**Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas**

Autor:

Rafael Alexander Mendoza Fernández

Tutores:

Ing. Harold García Alfonso

M.Sc. Eylín Hernández Luque

La Habana, junio de 2016
"Año 58 de la Revolución"

Declaración de Autoría

Declaro por este medio que yo, Rafael Alexander Mendoza Fernández, con carné de identidad 92070946242 soy el autor principal del trabajo titulado “Sistema informático de técnicas de procesamiento de datos cuantitativos” y autorizo a la Universidad de las Ciencias Informáticas y al Centro de Innovación y Calidad de la Educación, a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Y para que así conste, firmo la presente declaración jurada de autoría en La Habana a los _____ días del mes de _____ del año _____.

Autor: Rafael Alexander Mendoza Fernández

Tutor: Ms.C. Eylín Hernández Luque

Tutor: Ing. Harold García Alfonso

Tutor: Ing. Harold García Alfonso

Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas en Julio de 2014.

Profesor de Programación en la Facultad Introdutoria de Ciencias Informáticas.

Centro de Trabajo: Universidad de las Ciencias Informáticas (UCI). Centro de Innovación y Calidad de la Educación (CICE).

Cargo: Profesor del Centro de Innovación y Calidad de la Educación (CICE). Vicerrectoría de Formación.

Teléfono Oficina: +53 – 7 – 837 – 2522.

Correo electrónico: halfonso@uci.cu.

Tutora: Ms.C. Eylín Hernández Luque

Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas en Julio de 2007.

Máster en Gestión de Proyectos Informáticos en el año 2013.

Categoría docente: Asistente.

Profesor de las asignaturas de pregrado: Metodología de la Investigación Científica, Ingeniería y Gestión de Software, Base de Datos 1 y 2, Introducción a las Ciencias Informáticas, Proyecto de Investigación de desarrollo. Imparte en postgrado el curso de la Formación del Profesional en la Universidad Cubana.

Centro de Trabajo: Universidad de las Ciencias Informáticas (UCI). Centro de Innovación y Calidad de la Educación (CICE).

Cargo: Metodóloga de investigación del Centro de Innovación y Calidad de la Educación (CICE). Vicerrectoría de Formación.

Teléfono Oficina: +53 – 7 – 837 – 2522.

Correo electrónico: ehernandezl@uci.cu.

Terminar mi formación académica era una de mis metas, ahora, ya estoy al punto de alcanzarla, y todo gracias a aquellas personas que me ayudaron a cumplirla. Agradezco a mis padres por darme la vida, hacer de mí una persona mejor cada día, por encaminarme en todos los caminos de la vida, por brindarme todo su amor y compromiso, por hacer todo sacrificio para hacerme sentir mejor. A mi familia por darme todo el apoyo del mundo en todos los momentos de mi vida, por estar ahí cuando los necesité, por cuidarme y amarme. A Luis Carlos por todo el apoyo que me dio, por la ayuda en estos momentos finales de mi carrera y por su sacrificio. A mis tutores por brindarme su ayuda y dedicación en la tesis. Harold eres uno de los mejores tutores que he visto, gracias por todo el apoyo que me has dado, por estar arriba de mí para que hiciera las cosas, por ayudarme con los problemas que presentara, Eysín, aunque no pudiste estar presente en todos los momentos, las veces que necesitamos de ti hacías tiempo para ayudarnos. Yordanis gracias por todo el tiempo que me dedicaste en los momentos finales del documento cuando necesite ayuda con su revisión. A mi novia yai, por estar ahí, apoyándome y dándome todo su amor y cariño, por guiarme todo este tiempo para que fuese mejor. A la gente del barrio que siempre se preocupan por mí, siempre averiguando cómo estoy y como me va. A las amistades que me ayudaron y acompañaron estos 5 años en la carrera, la gente del grupo, del apartamento, a Zapato, Robert, Olga, el Magua, Adrian, Alexo, Sanchez, Hosford, el Emito, Acebo, Sergio, Lili, Eduardo, Ángel. A Raúl y Mayra por todo ese apoyo que me dieron desde primer año, por tratarme como otro hijo más.

Le quiero dedicar la tesis a mi abuelo que ya no esté presente en este mundo, por todos los valores que formo en mí, por todo el amor que me dio, por su cariño y su carácter, por enseñarme que todo se puede en esta vida. A mi hermana para que coja un ejemplo a seguir y lleve sus estudios hasta el final y se encamine en la vida.

En el Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas, se procesan encuestas que tributan a la calidad del proceso de formación integral en el área de las Ciencias Pedagógicas y Ciencias de la Educación. Este procedimiento se realiza de forma manual y depende de las habilidades y experiencias adquiridas por los especialistas. Esta investigación propone el desarrollo de un sistema basado en técnicas de procesamiento de datos cuantitativos, para contribuir a la disminución del tiempo de procesamiento y al aumento de la fiabilidad del análisis estadístico de la información generada por el sistema de encuestas *LimeSurvey*. Para el desarrollo de la propuesta, se utilizaron los métodos científicos analítico-sintético, histórico-lógico e inductivo-deductivo, para fundamentar los principales conceptos del desarrollo de sistemas basados en técnicas de procesamiento de datos cuantitativos. Se seleccionó XP como metodología de desarrollo de software, *Symfony2* como marco de trabajo, *Doctrine2* como capa de acceso a datos para el mapeo Objeto-Relacional y *PostgreSQL* como gestor de base de datos. Como parte de la propuesta se incluyen las técnicas estadísticas de la moda, media, mediana, varianza, desviación estándar, distribución de frecuencia, distribución de porcentajes y frecuencia acumulativa. Se realizaron pruebas unitarias y de aceptación al sistema basado en técnicas de procesamiento de datos cuantitativos, lo cual proporcionó una herramienta fiable en el procesamiento de las encuestas.

Palabras claves: Datos cuantitativos, encuestas, procesamiento, técnicas.

Surveys leading to the quality of the comprehensive training process in the fields of Pedagogical Sciences and Education are processed at the Center of Innovation and Quality of Education (CICE for its Spanish acronym). This procedure is manually operated and it depends on the skills and expertise of specialists. This research aims for developing a system based on quantitative data processing techniques in order to save time and improve statistical analysis reliability of data generated by LimeSurvey. The theoretical foundations of computer system development based on quantitative data processing were obtained by putting into practice some research methods: Analysis-synthesis, induction-deduction and the logical-historical one. During the research, it was also taken into consideration: XP as a software development methodology, Symfony2 as a web application framework, Doctrine2 as a data access layer for the object-relational mapping, and PostgreSQL as a Database Management System. The proposal also includes some statistical techniques: mode, mean, median, variance, standard deviation, frequency distribution, percentage distribution and cumulative frequency. The unit and acceptance tests based on quantitative data processing techniques were carried out on the system, providing a reliable tool in surveys processing.

Keywords: Quantitative data, surveys, processing, techniques.

Contenido de la memoria escrita de la investigación

| | |
|--|----|
| INTRODUCCIÓN..... | 1 |
| CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA..... | 6 |
| 1.1 INTRODUCCIÓN..... | 6 |
| 1.2 ANÁLISIS DE DATOS | 6 |
| 1.2.1 <i>Análisis de datos cuantitativos</i> | 7 |
| 1.2.2 <i>Análisis de datos cualitativos</i> | 8 |
| 1.3 TÉCNICAS DE ANÁLISIS DESCRIPTIVO DE DATOS..... | 10 |
| 1.4 HERRAMIENTAS DE PROCESAMIENTO DE DATOS | 16 |
| 1.5 METODOLOGÍAS DE DESARROLLO DE SOFTWARE | 19 |
| 1.6 TECNOLOGÍAS Y HERRAMIENTAS DE DESARROLLO..... | 24 |
| CONCLUSIONES PARCIALES | 29 |
| CAPÍTULO 2: CARACTERÍSTICAS DE LA PROPUESTA DE SOLUCIÓN..... | 31 |
| 2.1 INTRODUCCIÓN..... | 31 |
| 2.2 FLUJO ACTUAL DEL PROCESO..... | 31 |
| 2.3 PROPUESTA DE SOLUCIÓN | 32 |
| 2.3.1 <i>Fase de exploración</i> | 33 |
| 2.3.2 <i>Fase de planificación</i> | 34 |
| 2.3.3 <i>Fase de diseño</i> | 37 |
| 2.3.4 <i>Fase de codificación</i> | 40 |
| 2.3.5 <i>Requisitos no funcionales</i> | 42 |
| 2.4 PATRONES DE DISEÑO | 44 |
| 2.4.1 <i>Patrones GRASP</i> | 44 |
| 2.4.2 <i>Patrones GoF</i> | 45 |
| 2.5 ARQUITECTURA DEL SISTEMA | 46 |
| CONCLUSIONES PARCIALES | 48 |
| CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS..... | 49 |
| 3.1 INTRODUCCIÓN..... | 49 |

| | | |
|-------|---|----|
| 3.2 | ESTÁNDARES DE CODIFICACIÓN..... | 49 |
| 3.3 | DIAGRAMA DE DESPLIEGUE | 50 |
| 3.4 | PRUEBAS | 51 |
| 3.4.1 | <i>Pruebas unitarias</i> | 52 |
| 3.4.2 | <i>Pruebas de aceptación</i> | 53 |
| 3.4.3 | <i>Registro de No conformidades</i> | 54 |
| 3.5 | VALORACIÓN DE LA CONTRIBUCIÓN DE LA PROPUESTA | 56 |
| | CONCLUSIONES PARCIALES | 57 |
| | CONCLUSIONES | 58 |
| | RECOMENDACIONES | 59 |
| | BIBLIOGRAFÍA..... | 60 |
| | GLOSARIO DE TÉRMINOS | 64 |

Índice de Figuras

| | | |
|-----------|---|----|
| Figura 1: | Uso Personal de IDE y editores de PHP. [tomado de (Okeke Oluchukwu, 2015)]..... | 27 |
| Figura 2: | Uso en Negocios de IDE y editores de PHP. [tomado de (Okeke Oluchukwu, 2015)] | 28 |
| Figura 3: | Modelo conceptual..... | 32 |
| Figura 4: | <i>Diagrama entidad - relación del sistema de análisis</i> | 40 |
| Figura 5: | Patrón MVC [tomado de (Potencier, y otros, 2008)] | 48 |
| Figura 6: | Utilización de los estándares de codificación | 50 |
| Figura 7: | Diagrama de despliegue | 51 |
| Figura 8: | Pruebas unitarias procesamientosTest | 53 |
| Figura 9: | Pruebas unitarias procesamientosControllerTest | 53 |

Índice de Tablas

| | | |
|---------|--|----|
| Tabla 1 | Frecuencia absoluta del nivel de escolaridad. [tomado de (Briones, 1996)] | 12 |
| Tabla 2 | Distribución de porcentajes del nivel de escolaridad. [tomado de (Briones, 1996)]..... | 12 |
| Tabla 3 | Porcentajes de variables cualitativas. [tomado de (Briones, 1996)]..... | 13 |
| Tabla 4 | Frecuencia acumulativa de las edades. [tomado de (Briones, 1996)]..... | 14 |

| | |
|---|----|
| Tabla 5 Comparación entre las herramientas de análisis de datos | 19 |
| Tabla 6 Comparación entre las metodologías Ágiles y Tradicionales. [tomado de: (Cános, 2005)]..... | 20 |
| Tabla 7 Comparación de las metodologías ágiles. [tomado de (Criterios de selección de metodologías de desarrollo de software, 2010)]..... | 22 |
| Tabla 8 Comparación de las metodologías DSDM, Scrum y XP. [tomado de (Gimson, 2012)] | 23 |
| Tabla 9 Lista de IDE más mencionados. [tomado de (Okeke Oluchukwu, 2015)]..... | 26 |
| Tabla 10 HU Procesar encuestas utilizando técnicas estadísticas | 33 |
| Tabla 11 Plan de iteraciones..... | 35 |
| Tabla 12 Plan de entregas | 36 |
| Tabla 13 Tarjeta CRC procesamientosController | 37 |
| Tabla 14 Tarjeta CRC Procesamientos..... | 38 |
| Tabla 15 Descripción de las tablas de la Base de Datos..... | 39 |
| Tabla 16 TI Obtener respuestas de las preguntas con técnica seleccionada de la BD del sistema de encuesta | 41 |
| Tabla 17 TI Procesar las respuestas con la técnica seleccionada..... | 42 |
| Tabla 18 Prueba de aceptación HU18_PA18..... | 54 |
| Tabla 19 Registro de No conformidades | 55 |
| Tabla 20 Comparación del procesamiento con y sin el sistema informático basado en técnicas de procesamiento de datos cuantitativos | 56 |

INTRODUCCIÓN

Las tecnologías de la información han experimentado crecimientos desde los años 50, a un ritmo en el que la potencia de la informática crece exponencialmente todos los años. A este crecimiento de la informática le ha acompañado el de la gestión de la información, cuyos volúmenes hacen que sea indescifrable por sí sola. Según (Sarduy Domínguez, 2013) *«las empresas y entidades de información se dedicaban hasta hace poco al almacenamiento de datos para que los usuarios las utilizaran cuando y como pudieran. Ahora, con una visión más agresiva, los especialistas de información brindan no sólo datos o grandes volúmenes de información, sino que entregan informes y productos de análisis, con los cuáles les ayudan a convertir tantos datos en información sintetizada y confiable. Estos análisis de información ayudan a la toma de decisiones, que es una tarea que se hace cada vez más dinámica y requiere de un basamento informativo bien sustentado»*. El objetivo del análisis de información de acuerdo a (Dulzaides Iglesias, y otros, 2004) *«es obtener ideas relevantes, de las distintas fuentes de información, lo cual permite expresar el contenido sin ambigüedades, con el propósito de almacenar y recuperar la información contenida»*.

La Universidad de las Ciencias Informáticas (UCI) con motivo de elevar la calidad del proceso de formación integral, aplica diferentes encuestas a estudiantes y profesores. A los resultados obtenidos de estas encuestas se le realizan un proceso de análisis por parte de especialistas de la universidad. A través del análisis de la información encontrada, se deducen problemas actuales que atentan contra la calidad del proceso de formación. Además, permite realizar comparaciones con resultados de otros años, permitiendo esto, conocer si las metas propuestas son cumplidas. Principalmente, a través de ello es posible diagnosticar y brindar apoyo pedagógico a las facultades docentes de la universidad. El Centro de Innovación y Calidad de la Educación (CICE), perteneciente a la UCI, es el responsable de procesar la información obtenida de las encuestas. Actualmente, la mayor cantidad de información se obtiene a través de las respuestas encontradas en el sistema de encuestas de la universidad (*LimeSurvey*).

Los resultados que se obtienen de la herramienta *LimeSurvey* son previamente analizados por el módulo de Respuestas y Estadísticas del sistema de encuestas. Una vez obtenido el análisis del sistema, es exportado a una hoja de cálculo Excel donde se realiza un procesamiento estadístico a mayor profundidad. El procesamiento se realiza de forma manual por los especialistas del centro, donde van ordenando y realizando cálculos necesarios para que la información se convierta en una fuente de dato útil y comparable.

Luego se realizan comparaciones con resultados anteriores para llegar a conclusiones y tomar medidas según los resultados arrojados. El análisis realizado presenta dificultades, debido a los altos niveles de volúmenes de datos y por el módulo de Respuestas y Estadísticas del sistema de encuesta.

Por lo que plantea (Pérez, 2007) *LimeSurvey*, aunque posee un módulo de Respuestas y Estadísticas, no es un sistema especializado en arrojar buenos resultados estadísticos. El módulo, sólo cuenta con funcionalidades que permiten:

- Mostrar todas las respuestas.
- Generar estadísticas de respuestas cuantitativas.
- Exportar resultados a una aplicación (Procesador de texto, hoja de cálculo, fichero CSV).
- Exportar resultados a archivos de comando SPSS/PASW.
- Exportar resultados a archivos de datos R.

Para el completo análisis de la información, los especialistas necesitan de una gran cantidad de resultados para emitir los informes valorativos según la solicitud que se realice, por lo que todo el proceso es engorroso y el procesamiento de los datos está expuesta a errores manuales. Por otra parte, los resultados estadísticos obtenidos del sistema, no permiten una adecuada interpretación por parte de los especialistas por dos elementos fundamentales, estructura ineficiente para presentar los datos y uso de una sola técnica estadística para procesarlos. En consecuencia, los especialistas plantean que es lento el avance porque tienen que realizar luego de los análisis manuales diversas transformaciones a los datos. De modo que, en el CICE, el procesamiento de la información obtenida de las encuestas se caracteriza por presentar insuficientes resultados estadísticos arrojados por el módulo de Respuestas y Estadísticas del *LimeSurvey* y el análisis de grandes volúmenes de información de forma manual.

Lo anterior evidencia la necesidad del CICE de procesar la información recopilada en el sistema de encuestas, utilizando técnicas de procesamiento que satisfagan la necesidad de los especialistas del centro, con el objetivo de que dediquen la mayor parte del tiempo a la interpretación de los resultados y no al procesamiento de los datos.

Por lo antes expuesto se identifica como **problema científico** ¿Cómo contribuir al tiempo de procesamiento y a la fiabilidad del análisis estadístico de la información generada por el *LimeSurvey*?

En correspondencia con el problema planteado, el **objeto de estudio** lo constituye el proceso de análisis estadístico de la información que se genera en el Centro de Innovación y Calidad de la Educación, el **campo de acción** se enmarca en los sistemas informáticos que aplican técnicas de procesamiento de datos cuantitativos.

Para dar solución al problema se propone el siguiente **objetivo general**: Desarrollar un sistema informático basado en técnicas de procesamiento de datos cuantitativos, que contribuya a la disminución del tiempo de procesamiento y al aumento de la fiabilidad del análisis estadístico de la información generada por el *LimeSurvey* en el Centro de Innovación y Calidad de la Educación.

Objetivos específicos:

1. Sistematizar los fundamentos teórico-metodológicos de los sistemas informáticos que aplican técnicas de procesamiento de datos cuantitativos.
2. Caracterizar el proceso de análisis estadístico de la información que se genera en el Centro de Innovación y Calidad de la Educación.
3. Implementar el sistema informático basado en técnicas de procesamiento de datos cuantitativos.
4. Valorar la contribución de la propuesta, a través de la comprobación del tiempo de procesamiento y fiabilidad del análisis estadístico del sistema informático basado en técnicas de procesamiento de datos cuantitativos.

Idea a defender: Si se introduce un sistema informático basado en técnicas de procesamiento de datos cuantitativos en el Centro de Innovación y Calidad de la Educación, se contribuye a la disminución del tiempo de procesamiento y al aumento de la fiabilidad del análisis estadístico de la información generada por el *LimeSurvey*.

Para dar cumplimiento al objetivo anteriormente planteado se definen las siguientes **tareas de investigación**:

1. Comprobación de la validez del problema científico planteado.
2. Sistematización de los fundamentos teórico-metodológico de los sistemas informáticos que aplican técnicas de procesamiento de datos cuantitativos.

3. Caracterización del proceso de análisis estadístico de la información que se genera en el Centro de Innovación y Calidad de la Educación.
4. Fundamentación de las tecnologías, herramientas, metodología y lenguaje necesarios para el desarrollo del producto.
5. Modelación del ciclo de desarrollo del software teniendo en cuenta la metodología seleccionada.
6. Implementación del sistema informático basado en técnicas de procesamiento de datos cuantitativos, que contribuya a la disminución del tiempo de procesamiento y al aumento de la fiabilidad del análisis estadístico de la información generada por el *LimeSurvey* en el Centro de Innovación y Calidad de la Educación.
7. Validación de la propuesta a partir de las pruebas realizadas.
8. Valoración de la propuesta a través de la comprobación del tiempo de procesamiento y fiabilidad del análisis estadístico del sistema informático basado en técnicas de procesamiento de datos cuantitativos.

Para poder guiar el desarrollo de la investigación, se utilizaron diferentes **métodos científicos**.

Dentro de los **métodos teóricos** se tendrá en cuenta el método **Analítico – Sintético**, el cual permite un análisis de forma general y específica de los diferentes conceptos que conforman el objeto de estudio y el campo de acción.

El empleo del método **Histórico – Lógico**, basado en la extracción de lo teóricamente más importante de proyectos similares se utilizará para determinar el devenir histórico de la temática tratada, lo que permite contar con referencias teóricas sobre los sistemas informáticos de técnicas de procesamiento de datos cuantitativos.

El empleo del método **Inductivo – Deductivo**, basado en el razonamiento a través del cual se pasa de un conocimiento de cosas particulares a un conocimiento más general para el análisis de la información que se necesita por los especialistas del centro, que va a reflejar lo que hay de común en los procesos que se llevan a cabo.

Dentro de los **métodos empíricos** se utilizará la **entrevista** con el objetivo de adquirir nuevos conocimientos sobre el tema que se está investigando, así como para especificar las funcionalidades necesitadas. Además,

se utilizará para valorar la contribución de la propuesta a través de la comprobación del tiempo de procesamiento y la fiabilidad del análisis estadístico del sistema informático basados en técnicas de procesamiento de datos cuantitativos.

La presente investigación se divide en 3 capítulos organizados de la siguiente forma:

Capítulo 1. Fundamentación teórica: En este capítulo se exponen los principales conceptos asociados al dominio del problema. Contiene la base teórica para entender el problema a solucionar, además de los distintos sistemas de análisis de datos existentes. Al final, se define la metodología para guiar el proceso de desarrollo del software, así como las tecnologías y herramientas que serán utilizadas para la construcción del sistema.

Capítulo 2: Características de la propuesta de solución: En este capítulo se presenta la solución propuesta con todos los aspectos definidos en la fundamentación teórica. Esta solución contiene los artefactos necesarios para desarrollar el sistema a construir, entre los que se encuentran: historias de usuarios, tarjetas CRC, modelo de datos, tareas de ingeniería, entre otros.

Capítulo 3: Implementación y pruebas: En este capítulo se abordan aspectos relacionados con la implementación, especificando para ello los estándares de codificación utilizados y el diagrama de despliegue que permitirá tener un mayor conocimiento acerca de la estructura y funcionamiento del sistema. Por último, se realizan pruebas unitarias y de aceptación que validan el cumplimiento de las funcionalidades definidas.

Para finalizar, se presentan las **conclusiones**, **recomendaciones**, **bibliografía** y **anexos** que complementan la memoria escrita de la investigación.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se realiza una descripción de conceptos que se contemplan dentro del estado del arte de la investigación desarrollada. Además, se estudia el análisis de datos cuantitativos, cualitativos y sus diferentes técnicas con el objetivo de obtener la base para la propuesta de solución. Por otra parte, se realiza un estudio de los sistemas existentes para analizar si es factible la adaptación de alguno o que evidencie la necesidad de desarrollar una nueva solución. Para culminar, se estudian las metodologías y herramientas actuales para realizar una selección de las que serán utilizadas en la propuesta de solución.

1.2 Análisis de datos

Según (Kerlinger, 1983) *«el análisis de datos es el precedente para la actividad de interpretación. La interpretación se realiza en términos de los resultados de la investigación. Esta actividad consiste en establecer inferencias sobre las relaciones entre las variables estudiadas para extraer conclusiones y recomendaciones»*. El propio autor plantea que la interpretación se realiza en dos etapas:

- a) Interpretación de las relaciones entre las variables y los datos que las sustentan con fundamento en algún nivel de significancia estadística.
- b) Establecer un significado más amplio de la investigación, es decir, determinar el grado de generalización de los resultados de la investigación.

Por otra parte (Rouse, 2012) plantea que *«es la ciencia que examina datos en bruto con el propósito de sacar conclusiones sobre la información. Es usado en varias industrias con el objetivo de tomar mejores decisiones empresariales y verificar o reprobar modelos o teorías existentes. Se distingue de la extracción de datos por su alcance, su propósito y su enfoque sobre el análisis. Además, se centra en la inferencia, el proceso de derivar una conclusión basándose solamente en lo que conoce el investigador»*.

Tomando la idea que expone (Rouse, 2012), a continuación se muestran los tipos de análisis de datos, que se consideran significativos para el objeto de estudio de la investigación.

1.2.1 Análisis de datos cuantitativos

De acuerdo con (Sabino, 1992) «este tipo de operación se efectúa, naturalmente, con toda la información numérica resultante de la investigación». En consecuencia, (Briones, 1996) destaca que «hay numerosos tipos de análisis cuantitativos que resultan de la aplicación de uno o más criterios de clasificación. Así, según el objetivo principal, los análisis pueden clasificarse en descriptivas y en explicativas. Según el tiempo durante el cual se realiza el estudio se distinguen las investigaciones sincrónicas o transversales, que se refieren al objeto de investigación en un mismo periodo de tiempo, y las investigaciones diacrónicas o longitudinales, en las cuales los individuos se analizan durante un cierto tiempo, de manera más o menos continua».

Autores como (Moya Pardo, y otros, 2008) enuncian que, «en muchos casos, cuando se requieren técnicas estadísticas muy complejas es conveniente solicitar el apoyo de especialistas (que pueden conocer mejor las técnicas, en particular sus alcances y limitaciones)». Además, abordan la existencia de dos niveles de análisis cuantitativos.

Análisis descriptivos

- Consiste en asignar un atributo a cada una de las variables del modelo teórico.
- Los atributos pueden ser estadísticos descriptivos como la media, la mediana, la moda o la varianza, sobre cuyas propiedades existe gran conocimiento, experiencia y consenso, por lo que no es necesario realizar análisis de validez y fiabilidad. Pero en estadísticos menos conocidos (como por ejemplo la covarianza) puede ser necesario realizar este tipo de análisis.
- Es necesario tener definidos los criterios a seguir en caso de porcentajes elevados de no respuesta.
- El análisis descriptivo suele realizarse mediante la utilización de software estadístico como el SPSS, Systat, etc.

Análisis ligado a las hipótesis

Cada una de las hipótesis planteadas en el estudio debe ser objeto de una verificación. Cuando los datos recolectados son de naturaleza cuantitativa, esta verificación se realiza con la ayuda de herramientas estadísticas que se definen sobre la base de 3 aspectos principales:

- Las hipótesis que se desea verificar.
- Los diseños de investigación (experimental, cuasi experimental¹, experimental invocado).
- Distribución estadística de las variables.

1.2.2 Análisis de datos cualitativos

Los propios autores antes mencionados, (Moya Pardo, y otros, 2008) definen que «*no existen reglas formales (al estilo de los métodos estadísticos) para la realización de análisis cualitativos. Sin embargo, estos estudios suelen realizarse en las siguientes cuatro etapas*»:

1. Preparación y descripción del material bruto.
2. Reducción de los datos.
3. Elección y aplicación de los métodos de análisis.
4. Análisis transversal de los casos estudiados (si hubiera más de uno).

Preparación y descripción del material bruto

Consiste en preparar la base documental completa y fácilmente accesible.

- La información debe ser detectable (saber que existe), ubicable (dónde se encuentra) y trazable (dónde y cómo se obtuvo, cuáles son sus fuentes).
- La información suele ser voluminosa por lo que en muchos casos se requiere bastante trabajo de preparación.
- La prueba del éxito de esta etapa sería que un investigador ajeno a la investigación pudiera ejecutar las fases siguientes del análisis de datos a partir de la base documental.

Reducción de los datos

Se intenta reducir el volumen de los datos, despejando los componentes (las variables) de interés para la investigación. Existen tres formas de realizar la reducción de datos:

¹ estudio empírico para estimar el impacto causal de una intervención sobre la población objetivo

- **Redacción de resúmenes:** reduce la masa de información, pero no utiliza métodos muy específicos (no es replicable por otros investigadores). En el resumen se procura identificar los conceptos relevantes y cómo éstos se relacionan entre sí.
- **Codificación:** es el modo más desarrollado de reducción de datos. Consiste en atribuir categorías o conceptos a porciones del material bien circunscriptas y que presentan una alta unidad conceptual. Un buen sistema de codificación debe ser: Inclusivo, exhaustivo (abarcar todas las posibilidades) y permitir que cada elemento tenga tantos códigos como sea necesario para la investigación; Adaptativo, debe permitir generar nuevos códigos cuando la investigación lo requiera; Abarcar varios niveles de abstracción, permitir categorías descriptivas y analíticas.
- **Inducción:** consiste en identificar temas a partir de la base de datos y luego realizar reagrupamientos a partir de estos temas.

Elección y aplicación de los métodos de análisis

En esta etapa se procede a la interpretación de los datos utilizando tres posibles métodos de análisis para detectar “patrones” a partir de los datos previamente organizados.

- **Método de emparejamiento:** compara una configuración teórica predicha con una configuración empírica observada. (Requiere teoría previa y elección cuidadosa del caso o casos adecuados para poner la teoría a prueba).
- **Método iterativo:** Abordaje de los datos con mínima formalización teórica y construcción progresiva de una explicación. (Requiere conocimiento de las diferentes teorías que pueden explicar el fenómeno y la realización de un trabajo reiterado sobre los datos).
- **Método de análisis histórico (series temporales):** Consiste en formular predicciones sobre la evolución en el tiempo de un fenómeno. Es un caso particular del método de emparejamiento en el que la teoría es la predicción sobre el futuro.

Los tres métodos pueden utilizarse conjuntamente.

Análisis transversal

El análisis transversal apunta esencialmente a verificar si hay réplica de resultados entre varios casos o situaciones. Se agrega a las etapas precedentes cuando los datos cualitativos recolectados se refieren a varios casos del fenómeno (organizaciones, situaciones, individuos). Procede por comparación dónde cada situación es analizada de acuerdo al o los modos de análisis descritos precedentemente, de manera que se pueda captar si los modelos o patrones observados se reproducen. En general, el análisis de los datos recogidos consiste en determinar, mediante técnicas apropiadas, las formas, magnitudes y relaciones que se dan en esos datos, las cuales son buscadas de acuerdo con los objetivos propuestos.

Los objetivos propuestos en la presente investigación, la estructura e información de los datos y los resultados necesarios para elevar la calidad del proceso de formación integral, constituyen rasgos para determinar las técnicas apropiadas. Sobre la base de lo propuesto por (Briones, 1996), basándose en que *«el análisis descriptivo, constituye el primer nivel de análisis, y sus funciones son las de establecer cuál es la forma de distribución de una, dos o más variables en el ámbito global, cuántas unidades se distribuyen en categorías naturales, cuál es la magnitud de ella expresada en forma de una síntesis de valores, cuál es la dispersión con que se da entre las unidades del conjunto»*. Se determina utilizar el análisis de datos cuantitativos a un nivel de análisis descriptivo para dar cumplimiento a la propuesta de solución.

1.3 Técnicas de análisis descriptivo de datos

Profesores del Departamento de Matemáticas de la Facultad de Ciencias Exactas y Naturales de la Universidad de Buenos Aires (UBA, 2004) plantean que *«los métodos de Análisis Exploratorio o Estadística Descriptiva ayudan a comprender la estructura de los datos, de manera de detectar tanto un patrón de comportamiento general como apartamientos del mismo. Una forma de realizarlo, es mediante gráficos de sencilla realización e interpretación. Otra forma de describir los datos es resumiendo los datos en uno, dos o más números que caractericen al conjunto de datos con fidelidad»*.

Según (Briones, 1996) *«el análisis descriptivo cumple la función principal de caracterizar a un colectivo con una o más de esas expresiones de la variable analizada. Por ejemplo, el promedio de edad de un grupo es una característica, que puede tener significación por sí sola o permitir ser comparada con los promedios de otros grupos»*. Abordando las siguientes técnicas.

- Distribución de frecuencia absoluta.
- Distribución de porcentajes.
- Frecuencia acumulativa.
- Moda.
- Mediana.
- Medio aritmético.
- Varianza.
- Desviación estándar.

Distribución de frecuencia absoluta

Para elaborar la distribución de frecuencias se comienza por elegir un número reducido de categorías en las cuales se podrá agrupar la muestra. Posteriormente se obtienen los límites de cada categoría, dividiendo la diferencia de los límites de la muestra con el número de categoría elegido. La parte entera del valor resultante indica el tamaño del intervalo.

Por ejemplo, suponiendo que se tienen las escolaridades de 150 personas, con números de años de estudio como los siguientes:

12, 14, 11, 13, 8, 7, 9

Seleccionando el 5 como número de categorías y posteriormente calculando el tamaño del intervalo $[(14-7)/5]$, las categorías para agrupar los datos son:

7 - 8; 9 - 10; 11 - 12; 13 - 14

A continuación, se procede a anotar el número de personas que, según su nivel de escolaridad, cae en cada uno de los intervalos. Se supone que se da esta distribución de frecuencias que es la buscada según nuestro objetivo de investigación:

Tabla 1 Frecuencia absoluta del nivel de escolaridad. [tomado de (Briones, 1996)]

| Nivel de escolaridad | Frecuencia |
|----------------------|------------|
| 7-8 | 77 |
| 9-10 | 45 |
| 11-12 | 19 |
| 13-14 | 9 |
| Total | 150 |

La distribución del ejemplo muestra que el mayor número de personas tiene escolaridades de 7 a 8 años, seguido por las personas que tienen de 9 a 10 años. Se puede decir que la distribución encontrada es unimodal: hay una categoría en la cual hay un mayor número de casos.

Distribución de porcentajes

Un procedimiento que habitualmente acompaña al anterior consiste en el cálculo de porcentajes, a partir de las frecuencias absolutas distribuidas en intervalos de clase o categorías. Si se toma el ejemplo dado en la (Tabla 1) se tendría:

Tabla 2 Distribución de porcentajes del nivel de escolaridad. [tomado de (Briones, 1996)]

| Nivel de escolaridad | Frecuencia | Porcentajes |
|----------------------|------------|-------------|
| 7-8 | 77 | 51.3 |
| 9-10 | 45 | 30.0 |
| 11-12 | 19 | 12.7 |
| 13-14 | 9 | 6.9 |

| | | |
|--------------|-----|-----|
| Total | 150 | 100 |
|--------------|-----|-----|

Como se sabe, los porcentajes se calculan dividiendo cada frecuencia absoluta por el total de casos (77/150) y multiplicando el resultado por 100.

En el ejemplo anterior se trataba de la distribución de una variable cuantitativa continua. Para ello se hizo necesario agrupar los valores en ciertas categorías o intervalos. En otras ocasiones, las categorías están dadas de manera natural o son de naturaleza cualitativa, como sería el caso de las variables: sexo, ocupación, etc. Los procedimientos de cálculos son similares a los ya presentados.

Tabla 3 Porcentajes de variables cualitativas. [tomado de (Briones, 1996)]

| Proyectos al salir de la educación media | Frecuencia | Porcentajes |
|---|-------------------|--------------------|
| Continuar estudios | 35 | 24.6 |
| Trabajar | 84 | 59.2 |
| Estudiar y Trabajar | 15 | 10.5 |
| No sabe | 8 | 5.6 |
| Totales | 142 | 99.9 |

Frecuencia acumulativa

Las frecuencias acumuladas o acumulativas que se presentan en un cuadro permiten ver el número o porcentajes de casos que quedan en un cierto intervalo de la distribución cuando se le han sumado los casos de los intervalos anteriores. Así se muestra en el ejemplo que sigue:

Tabla 4 Frecuencia acumulativa de las edades. [tomado de (Briones, 1996)]

| Edades | Frecuencia | Porcentajes | Frecuencias acumuladas | Porcentajes acumulados |
|---------|------------|-------------|------------------------|------------------------|
| 7 a 8 | 62 | 51.7 | 62 | 51.7 |
| 9 a 10 | 35 | 29.1 | 97 | 80.8 |
| 11 a 12 | 14 | 11.7 | 111 | 92.5 |
| 13 a 14 | 9 | 7.5 | 120 | 100 |
| Totales | 120 | 100 | 120 | 100 |

Los datos indican que 111 de los 120 alumnos se encuentran hasta el intervalo 11- 12, y que, hasta ese mismo intervalo, se encuentra el 92,5 % de los alumnos.

Moda

La moda (o modo) es el valor de una serie que se da con mayor frecuencia entre los miembros de un colectivo. Puede ser utilizado con variables nominales ya que solamente es necesario contar los números de sujetos que hay en cada categoría de una variable. De igual forma ocurre con las variables cualitativas donde se realiza un conteo de las repeticiones de cada resultado y se lleva a una variable nominal. La determinación de la moda es muy fácil y por ello se le emplea como una primera medida de tendencia central. En la serie siguiente se observa que el modo es el número 20:

8, 7, 6, 10, 15, 16, 20, 20, 20, 21, 23

Algunas veces hay más de un modo:

6, 7, 8, 9, 9, 9, 10, 11, 12, 12, 12

En las cifras anteriores hay dos modos, los números 9 y 12, una distribución como esa se denomina bimodal.

Mediana

La mediana es el valor que ocupa el lugar central de una distribución ordenada de valores, habitualmente en orden ascendente. Si el número de valores es impar, la mediana es el valor central, si es par, es la semisuma de los dos valores centrales.

Ejemplos.

- Número impar de valores, ya ordenados: 10, 12, 14, 16, 19. La mediana es 14.
- Número par de valores, ya ordenados: 12, 14, 15, 16, 18, 20. La mediana es la semisuma de los valores centrales 15 y 16, es decir, 15,5.

Medio aritmético

El medio aritmético (o media aritmética) es una de las medidas de tendencia central más utilizada para caracterizar a un colectivo mediante un sólo valor. Ese valor es la suma de los valores de una variable cuantitativa continua, dividida por el número de valores sumados. Su fórmula para datos no agrupados es la siguiente:

$$\bar{X} = \frac{\sum X_i}{N}$$

Ejemplo: Si las cifras siguientes indican el número de horas de cada uno de 6 niños que ven televisión al día (2,5; 3; 3; 3,5; 2), el medio aritmético de esa actividad es la suma de las horas dividida por 6:

$$\bar{X} = \frac{2,5 + 3 + 3 + 2,5 + 2}{6} = 2,5$$

Varianza y Desviación estándar

Son medidas de dispersión o de variabilidad de los datos de una serie de valores. Indican la homogeneidad o heterogeneidad de ellos y, por lo tanto, la semejanza o diferencia que existe entre los individuos de un colectivo con relación a una cierta variable cuantitativa (la edad, los ingresos, etc.). Las principales de esas medidas son la varianza y la desviación estándar.

La varianza (S^2) es el promedio de las desviaciones elevadas al cuadrado de cada uno de los valores de una serie respecto del medio aritmético de ella. La desviación estándar (S), a su vez, es la raíz cuadrada de la varianza:

$$S^2 = \frac{\sum(X_i - \bar{x})^2}{N}$$

$$S = \sqrt{S^2}$$

1.4 Herramientas de procesamiento de datos

Autores como (Buendía Eisman, y otros, 1997) plantean que *«desde los años ochenta venimos experimentando una invasión del mundo de la informática en el campo de la investigación educativa a todos los niveles; bases de datos, bases bibliográficas... pero muy especialmente en el análisis de los datos. Las aplicaciones de la informática a la investigación son importantes, teniendo en cuenta que cada vez más la cantidad de datos que se manejan y la variedad de análisis que se realizan rebasan la capacidad del cálculo manual»*. Actualmente en el mundo existen numerosos sistemas de análisis, los cuales se han convertido en una solución a los problemas referentes a la comprensión y el análisis de la información. A continuación, se mencionan algunos de los sistemas más utilizados, los cuales se consideran significativos para el objeto de estudio de la investigación.

GenStat

Según (Ltd, 2009) es una herramienta omnímoda² del análisis de datos, que ofrece un fácil manejo por el sistema de menús exhaustivos, apoyado por la flexibilidad de un lenguaje de programación sofisticado. Todo esto está disponible en un ambiente enfocado a fomentar los prototipos y el desarrollo, mientras conduce a los usuarios nuevos y menos experimentados a través del sistema de menús bien elaborados. Se puede utilizar para manejar e ilustrar sus datos, resumir y comparar, modelar relaciones, estructurar investigaciones y por supuesto analizar sus experimentos desde lo más fácil ANOVA hasta lo más complejo REML. Entre sus características se encuentran.

² Según (Dictionary, 2007) *adj.* Que lo abraza y comprende todo.

- Tablas de resumen y menús de tablas de frecuencia que incluyen la capacidad de trazar una tabla.
- Test de permutación para el test chi-cuadrado del menú de independencia.
- Menú de control de compatibilidad para datos QTL.
- Gráficos bidimensionales con una distribución de los márgenes de las matrices.

Se utiliza en muchas disciplinas o ámbitos, como por ejemplo la sociología, las ciencias de la educación, la economía, el marketing, la religión, la etnología, la arquitectura, la planificación urbana, la sanidad pública y la medicina.

SPSS

Programa estadístico informático muy usado en las ciencias sociales y las empresas de investigación de mercado. Originalmente SPSS fue creado como el acrónimo de *Statistical Package for the Social Sciences* (Paquete Estadístico para las Ciencias Sociales). Es uno de los programas estadísticos más conocidos teniendo en cuenta su capacidad para trabajar con grandes bases de datos y una interfaz sencilla para la mayoría de los análisis (Bausela Herreras, 2005).

SYSTAT

Software de estadística que tiene todos los procesos y aplicaciones necesarios para cualquier procedimiento estadístico que necesite emplear para un análisis eficiente de sus datos. Ofrece desde las características de la estadística más elemental hasta la más compleja, utilizando los más sofisticados algoritmos. A los usuarios nuevos, brinda una interfaz amigable y fácil de comprender y usar. Para los usuarios con experiencia posee un lenguaje de comandos para analizar los datos con mayor eficiencia. En cualquiera de los dos casos, podrá explotar las técnicas para analizar diferentes tipos de datos basados en sistemas lineales, lineales generales y lineales mezclados.

Patdown (Centro de Información y Gestión Tecnológica (CIGET), Villa Clara)

El software permite la descarga de los documentos de patentes localizados en la base *Espacenetny*, importándolas a una base de datos en *Access*, donde se realiza el procesamiento primario, hasta llegar a la homogeneización de los datos. Permite extraer principales países generadores de patentes en la temática, países designados, registros por años, principales inventores, principales firmas, temáticas más

destacadas dentro de la tecnología y su evolución en el tiempo. Los gráficos de salida se pueden editar, pero sólo estéticamente, no conceptualmente. La herramienta permite además relacionar campos, por ejemplo: inventor-país y compañía-país (Cossío Cárdenas, 2009).

ProIntec (Universidad de Pinar del Río)

Por lo que plantean (Giráldez Reyes, y otros, 2008) es un sistema modular *ad hoc* que integra las etapas para el análisis de la información proveniente de bases de datos de patentes. Se desarrolló con lenguaje PHP, PostgreSQL como gestor de base de datos y Apache como servidor Web. Además, utiliza de forma transparente para el usuario el programa *NetDraw 2.062* para mostrar las representaciones visuales. Permite la búsqueda, recuperación, procesamiento, análisis y presentación de información contenida en documentos de patentes procedentes de bases de datos internacionales.

ToolInf (Consultoría Biomundi/IDICT)

Herramienta para el análisis y procesamiento de la información bibliográfica, mediante opciones de conteo, identificación de datos, clasificación de registros y creación y normalización de matrices de coocurrencia³. Su ventaja principal se enfoca en la realización de conteos automatizados, que pudieran hacerse mediante las tablas dinámicas de Excel, con la particularidad de que se cuentan por separado elementos que se encuentran dentro de una misma celda y se realizan conteos clasificatorios. En estos conteos no sólo se obtiene la cantidad de registros en que aparece cada elemento, sino también el número de los registros. Además, es muy útil en la realización de matrices de coocurrencia para campos con más de un elemento por celda, ya que se pueden combinar dos hojas de conteos clasificatorios en una matriz. Es una herramienta que se instala como un complemento a *Microsoft Excel*, por lo que su utilización es sencilla y eficaz (Cossío Cárdenas, 2009).

El estudio comparativo de los sistemas descritos, teniendo en cuenta los parámetros para seguir contribuyendo a la soberanía tecnológica de nuestro país y según las necesidades de la propuesta de solución, arrojó los resultados que se muestran en la (Tabla 5).

³ Por lo que plantea (Alonzo Ramos, 2001) se refiere a la utilización conjunta de dos unidades léxicas en una unidad superior, como una palabra o documento. Existe la suposición de que los términos son mutuamente dependientes cuando su utilización conjunta es muy común.

Tabla 5 Comparación entre las herramientas de análisis de datos

| Herramienta | Multiplataforma | Tipo de Licencia | Obtención de los datos | Código abierto |
|-------------|-----------------|------------------|-------------------------------|----------------|
| GenStat | No | Pagada | Entrada Manual | No |
| SPSS | No | Pagada | Entrada Manual, Ficheros SPSS | No |
| SYSTAT | No | Pagada | Entrada Manual | No |
| Patdown | - | Libre | Base Espacenet | No |
| ProIntec | Si | Libre | Base de datos PostgreSQL | No |
| ToolInf | Si | - | Fichero Excel | No |

El análisis de dichas herramientas arrojó el resultado de que constituyen un gran aporte para la realización e implementación de la propuesta de solución, no para su utilización, a consecuencia del uso de licencias pagadas en su mayoría. No brindan opciones ni permisos para realizar configuraciones sobre la aplicación y poder ser adaptados a diferentes ámbitos, por lo que sólo pueden ser utilizados para el fin con el que se crearon.

1.5 Metodologías de desarrollo de software

Autores como (Avison, 1995) presentan una definición de las metodologías de desarrollo muy clara, destacando sus principales componentes, fases, herramientas y técnicas. «Una metodología es una colección de procedimientos, técnicas, herramientas y documentos auxiliares que ayudan a los desarrolladores de software en sus esfuerzos por implementar nuevos sistemas de información. Una

metodología está formada por fases, cada una de las cuales se puede dividir en sub-fases, que guiarán a los desarrolladores de sistemas a elegir las técnicas más apropiadas en cada momento del proyecto y también a planificarlo, gestionarlo, controlarlo y evaluarlo». Considerando su filosofía de desarrollo se pueden agrupar en: metodologías ágiles o ligeras y metodologías pesadas.

Para la selección de uno de estos enfoques se identificaron los siguientes indicadores:

- Desarrollo del producto
- Cambios en la marcha
- Normas y reglas
- Interacción con el cliente
- Tamaño del equipo
- Artefactos generados
- Tiempo de desarrollo

Tabla 6 Comparación entre las metodologías Ágiles y Tradicionales. [tomado de: (Cános, 2005)]

| Indicadores | Metodologías Ágiles | Metodologías Tradicionales |
|--------------------------------|---|---|
| Desarrollo del producto | Se basan en heurísticas provenientes de prácticas de producción de código | Se basan en normas provenientes de estándares seguidos por el entorno de desarrollo |
| Cambios en la marcha | Preparados para cambios durante el proyecto | Cierta resistencia a los cambios |
| Normas y reglas | Impuestas internamente por el equipo | Impuestas externamente |

| | | |
|--|------------------------------------|---|
| Interacción con el cliente | El cliente es parte del desarrollo | Cliente interactúa con el equipo de desarrollo mediante reuniones |
| Tamaño del equipo de desarrollo | Grupos pequeños (<10) | Grupos grandes |
| Artefactos | Pocos artefactos | Muchos artefactos |
| Tiempo de desarrollo | Limitado | Amplio |

La investigación responde a condiciones y necesidades reales del equipo, determinadas por una elevada disposición y capacidad antes los nuevos cambios surgidos durante la marcha de la investigación. El cliente se encuentra altamente comprometido y motivado con el desarrollo del sistema. Además, el equipo de trabajo es pequeño y dispone de un tiempo limitado de desarrollo.

Las características que presenta el equipo de trabajo, determinan en gran medida la utilización de una metodología ágil, pues según estudios de (Abrahamsson, y otros, 2002) *«las metodologías de desarrollo de software tradicionales dirigidas por planes no se usan en la práctica. Se ha argumentado que las metodologías tradicionales son demasiado mecánicas para usarse en detalle. Los métodos de desarrollo de software ágil, que oficialmente comenzaron con la publicación del manifiesto ágil, hacen un intento por hacer un cambio en el campo de la ingeniería de software. Los métodos ágiles claman por colocar más énfasis en las personas, la interacción, el software funcionando, la colaboración del cliente, y el cambio más que en los procesos, las herramientas, los contratos y los planes»*.

Según plantean (Tinoco Gómez, y otros, 2010) *«los diseñadores de software tienen interés de trabajar con metodologías lo suficientemente documentadas, que nos faciliten la obtención de información, pero también es interesante trabajar con metodologías que dispongan de algún tipo de certificación y training»*. Según estas condiciones, han determinado seis clasificaciones que permiten seleccionar una metodología.

Las clasificaciones son:

- La metodología con mayor presencia en Internet. (1)
- La metodología mejor documentada. (2)
- Metodologías certificadas⁴ y con *training*⁵. (3)
- Metodologías con comunidades⁶. (4)
- Metodología más utilizada por empresas. Presencia empresarial. (5)
- Metodología más utilizada en proyectos software. (6)

En la (Tabla 7), se evidencia una comparación entre algunas metodologías ágiles en relación con los aspectos mencionados anteriormente. (Tinoco Gómez, y otros, 2010) plantean que «a un grupo de programadores profesionales en el medio local (10), se le ha aplicado una encuesta, sobre recordación, conocimiento y uso de metodologías, quedando un grupo de 5 metodologías, que se han evaluado, según este criterio de selección». Además, «la evaluación es asignada con una puntuación de 1 a 5 según las evidencias encontradas en internet».

Tabla 7 Comparación de las metodologías ágiles. [tomado de (Criterios de selección de metodologías de desarrollo de software, 2010)]

| Metodología | (1) | (2) | (3) | (4) | (5) | (6) | Total |
|--|-----|-----|-----|-----|-----|-----|-------|
| Agile Project Management (APM) | 2 | 1 | 3 | 5 | 1 | 1 | 13 |
| Dynamic Systems development methods (DSDM) | 1 | 3 | 5 | 5 | 4 | 4 | 22 |
| Scrum | 5 | 2 | 5 | 5 | 5 | 5 | 27 |

⁴ Se considera como metodologías certificadas aquellas que emiten un certificado que aseguran el cumplimiento y seguimiento de la metodología, así como sus técnicas y prácticas.

⁵ Una metodología dispone de *training*, si se encuentra alguna institución, organización o compañía que ofrezca formación de la metodología

⁶ Se considera que una metodología tiene comunidad, contemplando si se ha formado una comunidad relevante o si está asociada a la *Agile Alliance*, soportándola y cumpliendo sus principios.

| | | | | | | | |
|---------------------------------|---|---|---|---|---|---|-----------|
| Test Driven Developme | 3 | 4 | 3 | 2 | 2 | 2 | 16 |
| Extreme Programming (XP) | 4 | 5 | 3 | 2 | 3 | 3 | 20 |

A partir de la selección del tipo de metodología ágil y tomando en cuenta la investigación realizada por (Tinoco Gómez, y otros, 2010), se analizan las metodologías DSDM, Scrum y XP. Para ello se tomaron en cuenta un conjunto de indicadores:

- Participación del cliente.
- Realización de pruebas.
- Nivel de documentación.
- Desarrollo.
- Tamaño del proceso.

Tabla 8 Comparación de las metodologías DSDM, Scrum y XP. [tomado de (Gimson, 2012)]

| Indicadores | DSDM | Scrum | XP |
|---------------------------|---------------------------------|-------------------|--|
| Participación del cliente | Implicación y relación estrecha | En cada iteración | Activa, el cliente tiene que estar presente y disponible todo el tiempo para el equipo |
| Realización de pruebas | A lo largo del ciclo | En cada iteración | Ejecutadas constantemente ante cada modificación del sistema |
| Nivel de documentación | Media | Poca | Se sustituye la documentación por la comunicación |

| | | | |
|--------------------|------------|-------------------------|-------------------------|
| Desarrollo | Por ciclos | Iterativo e incremental | Iterativo e incremental |
| Tamaño del proceso | Medio | Medio | Pequeño |

El equipo de desarrollo seleccionó la metodología XP por las ventajas que ofrece. Permite realizar pruebas constantemente, lo cual garantiza una aplicación estable y sin errores. La activa participación y comunicación del cliente aumenta las posibilidades del éxito de la aplicación, lo cual permite que todo esté correcto y a la medida. El desarrollo iterativo e incremental, permite desarrollar pequeñas mejoras, realizándose consecutivamente. Además, brinda la posibilidad de organizar el trabajo y determinar la mejor forma de lograr los objetivos, motivando al equipo de desarrollo.

1.6 Tecnologías y Herramientas de desarrollo

Para el desarrollo de un proyecto es necesario el estudio de las tecnologías y herramientas a utilizar, así como definir el lenguaje de modelado y programación, herramientas de modelado e implementación y técnicas para elevar la calidad del producto a desarrollar.

Para la selección de las herramientas y tecnologías se definieron cinco indicadores generales (1) conocimientos y habilidades del equipo de desarrollo, (2) tiempo limitado, (3) herramientas de software libre o licencias sin costo, (4) tecnologías que garanticen un sistema multiplataforma y configurable y (5) aceptación y bibliografías existentes en la comunidad universitaria.

Lenguaje de modelado

Los lenguajes de modelado, según (Magaña, 2009), «*permiten representar un software y describir las actividades a realizar para desarrollarlo*». En la actualidad existen dos lenguajes que son los más populares y ampliamente utilizados, BPMN y UML. BPMN, de acuerdo a los criterios de (Milestone, 2009) «*es una notación gráfica que describe la lógica de los pasos de un proceso de negocio. Esta ha sido especialmente diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes de las diferentes actividades*». Por otra parte, el lenguaje UML facilita a los desarrolladores la especificación,

visualización y documentación de modelos de sistemas de software. Según (Jacobson, y otros, 2000) «*permite modelar analizar y diseñar sistemas orientados a objetos y es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad*».

La relación entre UML y BPMN viene dada por el hecho de que ambos definen una notación gráfica para los procesos de negocio, aunque, usan enfoques diferentes para modelarlos. En la investigación se utiliza UML, teniendo en cuenta las habilidades del equipo de desarrollo y el tiempo limitado para la propuesta de solución. Además, la realización de un sistema orientado a objetos.

Herramienta de modelado

La calidad de un proyecto está determinada en gran medida por una adecuada decisión a la hora de escoger una herramienta de modelado con UML. Es por ello que el equipo de desarrollo además de los indicadores generales utilizó otros para la selección de la herramienta de modelado con UML. Los otros elementos a valorar fueron (1) el modelado de datos y manejo de diagramas, (2) el apoyo metodológico y soporte completo UML.

La herramienta de modelado que responde con mayor potencial a los indicadores evaluados anteriormente, es *Visual Paradigm (VP)*. VP es una herramienta multiplataforma para el modelado de aplicaciones, sobre todo en proyectos donde se apliquen intensivamente los conceptos avanzados de orientación a objeto (Canales Mora, 2004). Además, facilita a los ingenieros de software diseñar, integrar y modelar visualmente los distintos diagramas que se generan a lo largo del desarrollo del software. Presenta un generador de código que soporta más de 10 lenguajes (Java, C++, PHP, etc.) y proporciona la ingeniería inversa (Paradigm, 2013).

Lenguaje de programación

Desde los inicios de Internet, fueron surgiendo diferentes demandas por los usuarios y se dieron soluciones mediante lenguajes estáticos. A medida que pasó el tiempo, las tecnologías fueron desarrollándose y surgieron nuevos problemas a dar solución. Esto dio lugar a desarrollar lenguajes de programación para la web dinámicos, que permitieran interactuar con los usuarios y utilizaran sistemas de Bases de Datos (Pérez

Váldez, 2007). Según el propio autor PHP, ASP, ASP.net, JSP y Python son ejemplos de los más utilizados en la actualidad.

A partir de los indicadores generales se seleccionó el lenguaje PHP ya que es un poderoso lenguaje e intérprete y es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. Por otro lado (Pérez Váldez, 2007) plantea que:

- Es muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta la orientación a objetos. Clases y herencia.
- Es un lenguaje multiplataforma.
- Presenta conexión con la mayor cantidad de manejadores de bases de datos.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Posee amplia documentación y una gran comunidad de desarrolladores.

Entorno de desarrollo integrado

A partir de la selección del lenguaje de programación, es necesario definir un entorno de desarrollo integrado (IDE, por sus siglas en inglés) que permita desarrollar aplicaciones utilizando el lenguaje PHP. Estudios de (Okeke Oluchukwu, 2015) arrojaron los resultados que se muestran en la (Tabla 9).

Tabla 9 Lista de IDE más mencionados. [tomado de (Okeke Oluchukwu, 2015)]

| IDE | Mencionado |
|-------------|------------|
| PHPStorm | 81 veces |
| Eclipse PDT | 33 veces |
| NetBeans | 29 veces |
| Zend Studio | 17 veces |

| | |
|--------------------|---------|
| Komodo IDE | 9 veces |
| phpDepsiner | 1 vez |

Basándose en los indicadores generales, la tendencia del mundo actual y los resultados arrojados en las Figura (1 y 2), se selecciona *PHPStorm* como IDE de desarrollo.

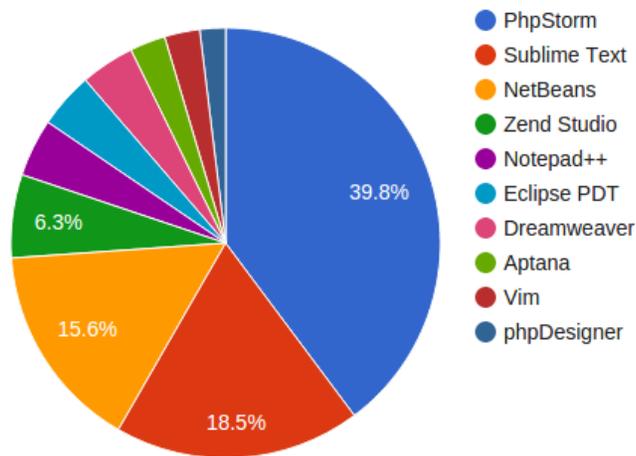


Figura 1: Uso Personal de IDE y editores de PHP. [tomado de (Okeke Oluchukwu, 2015)]

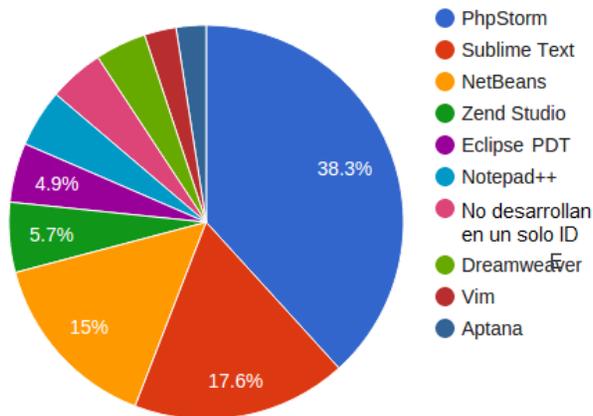


Figura 2: Uso en Negocios de IDE y editores de PHP. [tomado de (Okeke Oluchukwu, 2015)]

Según (Gadja, 2007) «*PHPStorm es un IDE diseñado específicamente para desarrollar aplicaciones de internet escritas en PHP. Las principales características se dividen en las siguientes categorías*».

- Navegación.
- Sincronización de ficheros.
- VCS.
- Depuración.
- Edición y refactorización de código.

El propio autor aborda también que «*las aplicaciones modernas, consisten en muchos archivos, librerías, módulos y clases almacenados en diferentes carpetas. PHPStorm, ayuda a la navegación entre ellas. Los archivos abiertos se encuentran distribuidos en pestañas y es posible navegar entre ellos con el mouse y los accesos del teclado*».

Framework de desarrollo

Un *framework* (marco de trabajo) como plantea (Johnson, y otros, 1998) es un «*diseño abstracto orientado a objetos para un determinado tipo de aplicación, que se compone de una clase abstracta para cada componente principal del diseño; contendrá normalmente una librería de subclases que pueden ser utilizadas como componentes del diseño*». Aunque, como resumen se puede decir según (Larman, 2004) que «*es un conjunto cohesivo de interfaces y clases que colaboran para proporcionar los servicios de la parte central e invariable de un subsistema lógico (frozen spot)*». Tomando como partida los elementos definidos por (Booch, y otros, 1999) de que un *framework* es un «*patrón arquitectónico que proporciona una plantilla extensible para aplicaciones dentro de un dominio*». Además, teniendo en cuenta los indicadores generales de selección y los *framework* de desarrollo para PHP más utilizados en la actualidad, el equipo de desarrollo selecciona el *framework* *Symfony2*.

Symfony2 es una de las versiones más recientes de *Symfony*, el popular *framework* para desarrollar aplicaciones PHP. Se anunció por primera vez a principios de 2009 y supone un cambio radical tanto en arquitectura interna como en filosofía de trabajo respecto a sus versiones anteriores. *Symfony2* ha sido ideado para explotar al límite todas las nuevas características de PHP 5.3 y por eso es uno de los

frameworks PHP con mejor rendimiento. Su arquitectura interna está completamente desacoplada, lo que permite reemplazar o eliminar fácilmente aquellas partes que no encajan en el proyecto que se esté desarrollando. *Symfony2* también es el *framework* que más ideas incorpora del resto de *frameworks*, incluso de aquellos que no están programados con PHP.

Sistema Gestor de Base de Datos (SGBD)

Según (Cobo, 2007) «un SGBD permite el almacenamiento, manipulación y consulta de datos pertenecientes a una base de datos (BD) organizada en uno o varios ficheros. En el modelo más extendido (BD relacional) la BD consiste, de cara al usuario, en un conjunto de tablas entre las que se establecen relaciones». En la propuesta de solución no existe la necesidad de la selección de un SGBD específico. *Symfony*, está configurado para soportar tanto MySQL como SQLite, PostgreSQL, Oracle, IBM DB2 y SQLServer. Además, incorpora ORM Doctrine que según (Dunglas, 2013) la «*biblioteca Doctrine's ORM* permite una fácil persistencia y recuperación de gráfico de objetos PHP, sin necesidad de escribir consultas SQL a mano. También proporciona un lenguaje potente orientado a objetos similar a SQL, una herramienta de generación de esquema de base de datos, un sistema de eventos y mucho más». Por otra parte, entre los *plugging* del IDE seleccionado se encuentra una herramienta para BD y SQL que permite la conexión y el manejo de los SGBD como DB2, Derby, H2, HSQLDB, MySQL, Oracle, PostgreSQL, SQLServer, SQLite y Sysbase (Vink, 2012). Sin embargo, por parte del equipo de desarrollo se selecciona PostgreSQL para la administración y se evidencia necesario también el uso MySQL, SGBD utilizado en el sistema de encuestas de la universidad.

Conclusiones parciales

- El análisis y valoración realizado en el marco teórico de la investigación arrojó que, de los diferentes tipos de análisis de datos descritos, se selecciona el análisis de datos cualitativos a un nivel de análisis descriptivo. Además, que las técnicas de análisis de datos descriptivas seleccionadas son distribución de frecuencia absoluta, distribución de porcentajes, frecuencia acumulativa, moda, mediana, medio aritmético, varianza y desviación estándar.
- La caracterización realizada a los sistemas homólogos, permitió constatar que las herramientas de procesamiento de datos analizadas, poseen limitaciones para darle solución a la investigación, por

el uso de licencias pagadas, no ser multiplataforma en su mayoría y no presentar permisos para configurar su código.

- El estudio y análisis de las tecnologías a utilizar para el ambiente de desarrollo de la propuesta corroboró que para el proceso de desarrollo de software se utilizará la metodología XP, atendiendo a las características presentadas por el equipo de desarrollo. Así como, se determinó el uso del *framework* de desarrollo *Symfony2*; los lenguajes de programación PHP y UML y las herramientas *Visual Paradigm* y *PHPStorm*.

CAPÍTULO 2: CARACTERÍSTICAS DE LA PROPUESTA DE SOLUCIÓN

2.1 Introducción

En el presente capítulo se ofrece una descripción de la propuesta de solución con el objetivo de solucionar el problema planteado inicialmente. Se construye el modelo conceptual en el que se reflejan los principales conceptos que se deben abarcar, para el desarrollo del sistema de análisis de datos cuantitativos. Posteriormente, se definen las historias de usuario, tarjetas CRC, el modelo de datos, tareas de ingeniería y requisitos no funcionales correspondientes. Para culminar se especifican los patrones de diseños y la arquitectura de software utilizada.

2.2 Flujo actual del proceso

A través de la interacción con varios especialistas del CICE y las especificaciones del cliente, fue posible obtener una visión del procesamiento de los datos obtenidos del sistema de encuestas.

El CICE es el centro encargado de administrar el sistema de encuestas de la universidad. En este sistema se encuentran almacenadas las encuestas que realiza la universidad con el objetivo de mejorar el proceso de formación integral. Además, cuenta con un módulo de análisis y estadísticas con el cual se procesan las respuestas de las encuestas. Este módulo, para procesar, solo utiliza la distribución de frecuencia como técnica estadística y resulta insuficiente en el análisis de las respuestas que responda a las necesidades de los especialistas. Esta situación provoca que los especialistas del centro realicen el procesamiento manual a través de un fichero Excel que exporta el módulo con las respuestas para cada una de las encuestas. Con los resultados del procesamiento alcanzado, los especialistas pueden brindar apoyo pedagógico a las facultades docentes de la universidad.

Para ayudar a comprender el flujo del procesamiento de las respuestas de las encuestas se utiliza el siguiente modelo conceptual (Figura 3).

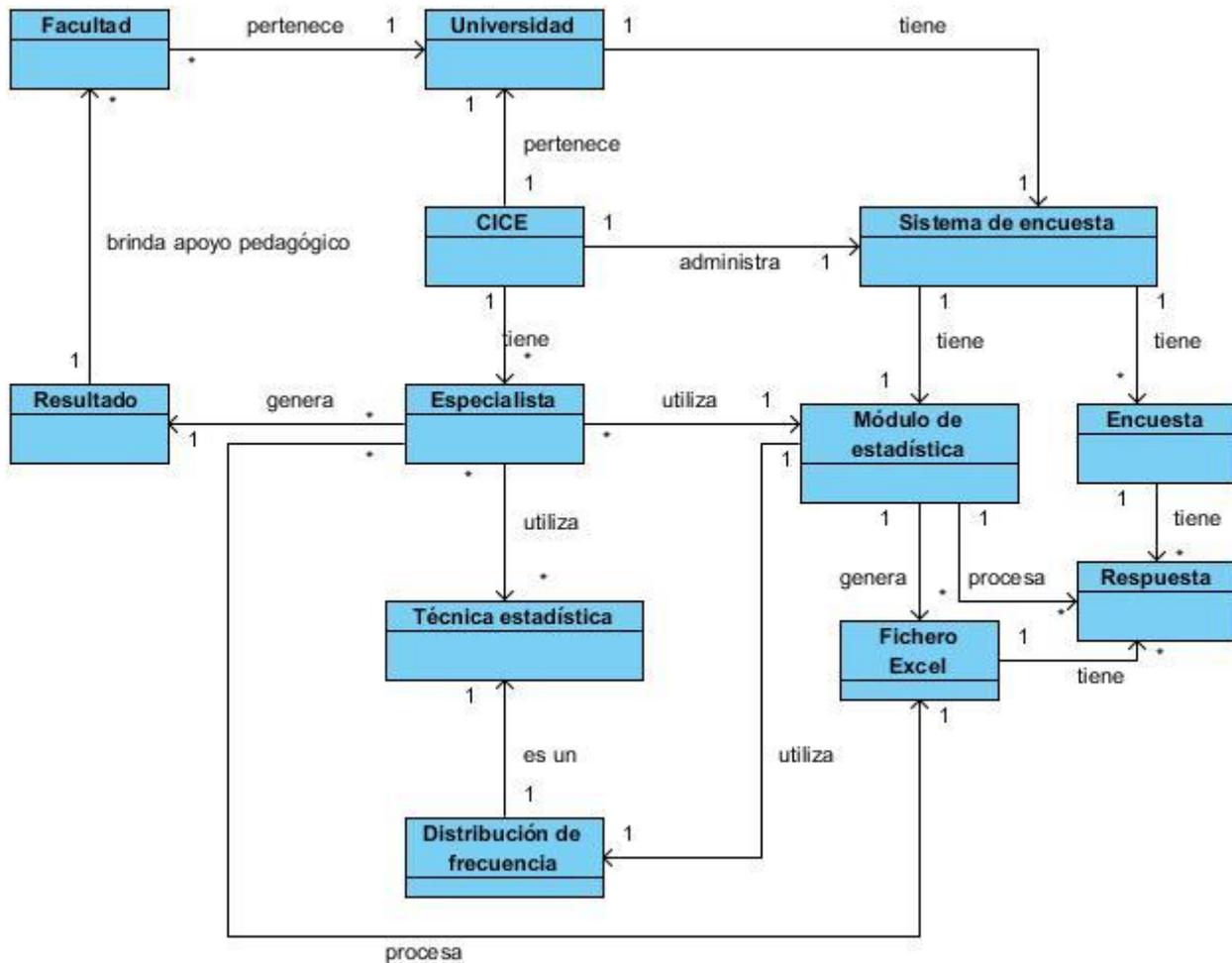


Figura 3: Modelo conceptual

2.3 Propuesta de solución

Para dar solución al problema planteado al inicio de la investigación y los problemas encontrados en el flujo actual del proceso, se propone el desarrollo de un sistema de análisis de datos que tenga entre sus funcionalidades principales el procesamiento de las respuestas almacenadas en el sistema de encuestas. El procesamiento realizado en el sistema utilizará las técnicas (Moda, Media, Mediana, Varianza, Desviación estándar, Distribución de frecuencia, Distribución de porcentajes y Frecuencia acumulativa) y las respuestas serán obtenidas a través de consultas SQL directamente de la base de datos del sistema de encuestas. En el sistema únicamente se procesarán las preguntas seleccionadas de la encuesta a procesar, pudiéndose

filtrar estas por preguntas y determinando las muestras para cada filtro. Los resultados se obtendrán directamente en el sistema o ser exportados a ficheros Excel y Pdf.

Para lograr una mejor organización en el sistema de análisis de datos se hace necesario realizar una planificación de dicho proceso de desarrollo de software, basándose en las fases que propone la metodología seleccionada.

2.3.1 Fase de exploración

Por lo que plantean (Letelier, y otros, 2012) esta es la primera fase que propone la metodología XP, en la que los clientes plantean a grandes rasgos las funcionalidades que son de interés para la elaboración del producto, transformándose las mismas en historias de usuario.

Historias de Usuario (HU)

Por lo que plantean (Penadés, y otros, 2014) «Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software. En ellas se describen brevemente las características que el sistema debe poseer. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas».

A continuación en la (Tabla 10), se describe una historia de usuario generada en la investigación, las demás pueden ser encontradas en el Anexo 1.

Tabla 10 HU Procesar encuestas utilizando técnicas estadísticas

| Historia de Usuario | |
|---|---------------------|
| Número: 18 | Usuario: Procesador |
| Nombre de historia: Procesar encuestas utilizando técnicas estadísticas | |
| Prioridad en negocio: Alta | Puntos estimados: 2 |

| | |
|--|-----------------------|
| Riesgo en desarrollo: Alta | Iteración asignada: 2 |
| Programador Responsable: Rafael Alexander Mendoza Fernández | |
| Descripción: Procesar cada pregunta de la encuesta utilizando la técnica seleccionada. | |
| Observaciones: Solo se procesan las preguntas con técnicas seleccionadas. | |

2.3.2 Fase de planificación

Según (Letelier, y otros, 2012) el propósito general de la fase de planificación es que los clientes y desarrolladores se pongan de acuerdo en qué historias de usuario deben estar listas para la primera liberación. De forma particular, es en esta fase donde los programadores obtienen la estimación del esfuerzo necesario para la elaboración de las historias de usuario.

Plan de iteraciones

Según (Joskowicz, 2008) «*las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido. Al comienzo de cada ciclo, se realiza una reunión de planificación de la iteración. Cada historia de usuario se traduce en tareas específicas de programación*».

Por lo planteado en (Computación, 2011) los elementos que deben tomarse en cuenta durante la elaboración del Plan de la Iteración (Tabla 11) son: historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Todo el trabajo de la iteración es expresado en tareas de programación y cada una de ellas es asignada a un programador como responsable.

Para la elaboración del plan de iteraciones se tuvo en cuenta la estimación de las historias de usuarios, así como la prioridad y el orden de cada una de las historias de usuario según su necesidad para el desarrollo y el negocio.

Capítulo 2: Características de la propuesta de solución

Tabla 11 Plan de iteraciones

| Iteración | Historia de usuario | Semanas estimadas |
|-----------|---|-------------------|
| 1 | Adicionar Usuario | 6.6 |
| | Modificar Usuario | |
| | Listar Usuarios | |
| | Eliminar Usuarios | |
| | Adicionar Rol | |
| | Modificar Rol | |
| | Listar Roles | |
| | Eliminar Roles | |
| | Adicionar Permiso | |
| | Modificar Permiso | |
| | Listar Permisos | |
| | Eliminar Permisos | |
| | Listar encuestas disponibles | |
| | Ver detalles de las encuestas disponibles | |
| | Seleccionar técnica de procesamiento | |

Capítulo 2: Características de la propuesta de solución

| | | |
|---|--|---|
| 2 | Guardar configuración | 7 |
| | Cargar configuración | |
| | Procesar encuestas utilizando técnicas estadísticas | |
| | Listar encuestas procesadas | |
| | Exportar resultados de la encuesta procesada a Excel | |
| | Exportar resultados de la encuesta procesada a Pdf | |
| | Ver resultados de la encuesta procesada | |
| 3 | Procesar encuesta de satisfacción | 2 |

Plan de entregas

Por lo que plantea (Cános, 2005) en esta etapa se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente, estableciendo la prioridad de cada historia de usuario.

La primera entrega permite calcular la velocidad del proyecto, optimizando de esta manera las fechas de entrega de las distintas iteraciones (Tabla 12).

Tabla 12 Plan de entregas

| Entregable | Fin iteración 1 | Fin iteración 2 | Fin iteración 3 |
|--|-----------------|-----------------|-----------------|
| Sistema informático de técnicas de procesamiento de datos cuantitativos para el Centro de Innovación y Calidad de la Educación | Enero 2016 | Marzo 2016 | Abril 2016 |

2.3.3 Fase de diseño

XP establece prácticas especializadas que inciden directamente en la realización del diseño para lograr un sistema robusto y reutilizable tratando de mantener su simplicidad, es decir, crear un diseño evolutivo que se va mejorando incrementalmente y que permite hacer entregas pequeñas y frecuentes de valor para el cliente según (Letelier, y otros, 2012). Los propios autores plantean además que, a la hora de darle cumplimiento a la actividad de diseñar, XP no especifica ninguna técnica de modelado. Pueden utilizarse indistintamente sencillos esquemas en una pizarra, diagramas de clases utilizando UML o tarjetas CRC (Clase, Responsabilidad y Colaboración) siempre que sean útiles, tributen a la comprensión y no requieran mucho tiempo en su creación.

Tarjetas CRC

La metodología XP en lugar de utilizar diagramas para desarrollar modelos representa las clases mediante tarjetas. La técnica *Class, Responsibility and Collaboration* (CRC) creada por Kent Beck y Ward Cunningham, según (Bellin, y otros, 1997) y (Kent, y otros, 1989), ayudan a realizar el análisis o el diseño orientado a objetos.

Cada tarjeta representa una clase en la programación orientada a objetos y define sus responsabilidades (lo que ha de hacer) y las colaboraciones con las otras clases (cómo se comunica con ellas). Como resultado del trabajo realizado se obtuvo un total de 14 tarjetas CRC, a continuación, se muestran dos de ellas, las demás pueden ser encontradas en el Anexo 2.

Tabla 13 Tarjeta CRC procesamientosController

| Clase procesamientosController | |
|--------------------------------|----------------|
| Responsabilidades | Colaboradores |
| Generar Pdf Generar Excel | Procesamientos |

Capítulo 2: Características de la propuesta de solución

| | |
|---|--|
| <p>Generar procesamiento</p> <p>Leer xml</p> <p>Listar procesamientos</p> <p>Listar encuestas</p> <p>Ver estadísticas de encuestas</p> <p>Obtener conexión</p> <p>Obtener respuestas</p> <p>Procesar respuestas</p> | |
|---|--|

Tabla 14 Tarjeta CRC Procesamientos

| Clase Procesamientos | |
|--|----------------|
| Responsabilidades | Colaboradores |
| <p>Obtener nombre de procesamiento</p> <p>Editar nombre de procesamiento</p> <p>Obtener fecha</p> <p>Editar fecha</p> <p>Obtener encuesta</p> <p>Editar encuesta</p> | <p>Usuario</p> |

Capítulo 2: Características de la propuesta de solución

| | |
|-----------------|--|
| Obtener usuario | |
| Editar usuario | |

Modelo de datos

El diagrama entidad - relación presentado en la (Figura 4) se corresponde con la representación física de la base de datos del sistema de análisis. Expresa las entidades relevantes para el sistema, así como sus interrelaciones y propiedades. El modelo de datos correspondiente tiene un total de 8 tablas, de las cuales 1 es nomenclador y las otras 7 constituyen las tablas que almacenan los datos necesarios para el sistema. Seguidamente se detallan las tablas de la base de datos.

Tabla 15 Descripción de las tablas de la Base de Datos

| Tabla | Descripción |
|-----------------------|---|
| tecnica | Almacena los tipos de técnicas del sistema |
| usuario | Almacena los usuarios del sistema |
| rol | Almacena los roles del sistema |
| permiso | Almacena los permisos del sistema |
| rol_permiso | Almacena la relación entre los roles y los permisos |
| procesamiento | Almacena procesamientos realizados |
| configuracion | Almacena las configuraciones realizadas |
| configuracion_tecnica | Almacena la relación de las técnicas y las preguntas de las configuraciones |

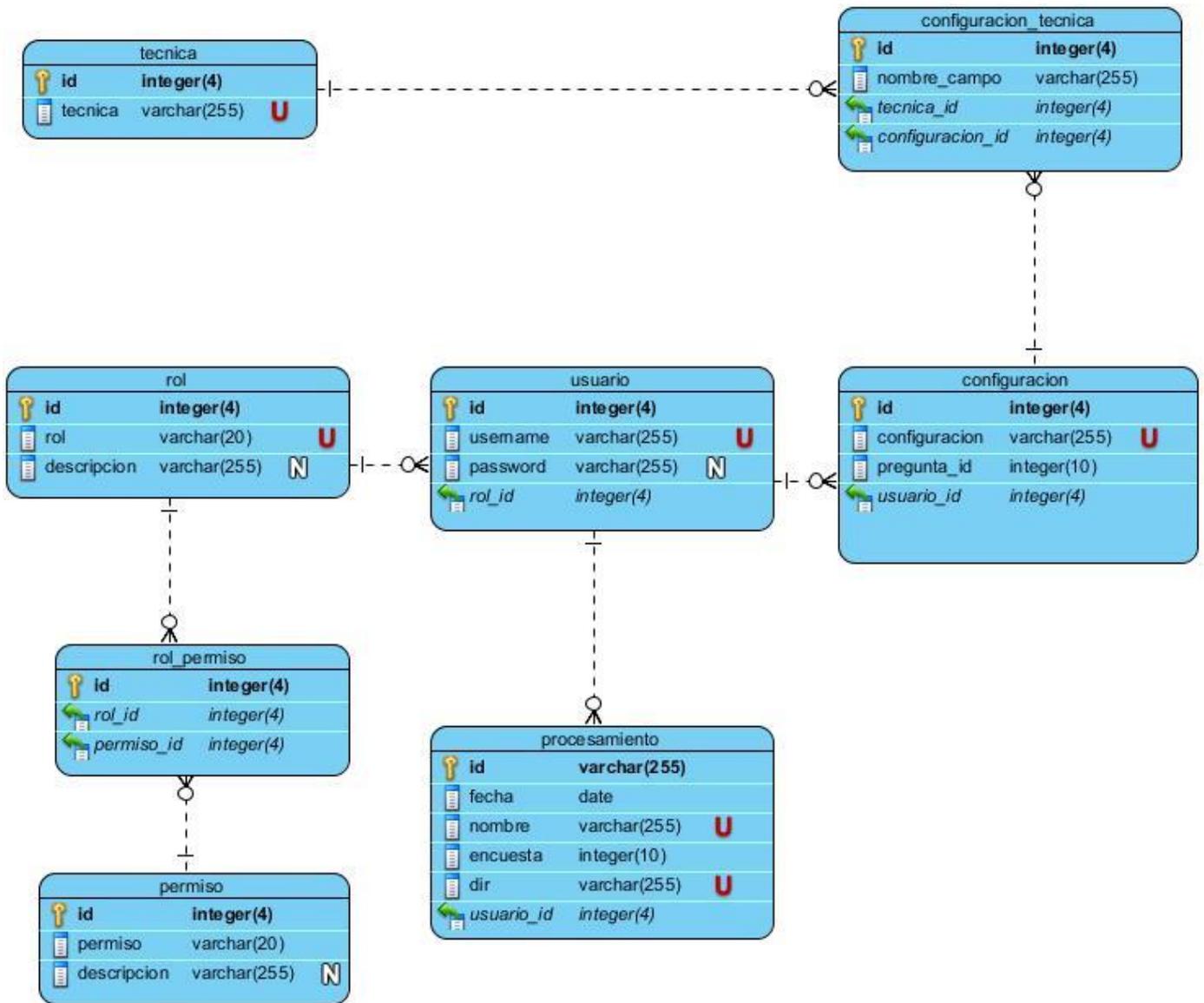


Figura 4: Diagrama entidad - relación del sistema de análisis

2.3.4 Fase de codificación

De acuerdo a (Letelier, y otros, 2012) en esta fase las historias de usuario elegidas se descomponen en tareas de programación o ingeniería, escritas en lenguaje técnico, que a su vez son convertidas posteriormente a código. La programación en parejas, refactorización, la integración continua y la

Capítulo 2: Características de la propuesta de solución

compilación de diez minutos son un subconjunto de las prácticas que sirven de guía durante esta crucial actividad.

Tareas de ingeniería

Una vez identificadas las historias de usuario los programadores proceden a descomponer cada una en tareas específicas, las denominadas tareas de programación o de ingeniería que están escritas técnicamente y que darán solución a la historia correspondiente. Las Tareas de Ingeniería a realizarse en cada iteración se agrupan en la siguiente tabla. En el Anexo 3 se especifican los detalles de cada una de las tareas.

Tabla 16 TI Obtener respuestas de las preguntas con técnica seleccionada de la BD del sistema de encuesta

| Tarea de ingeniería | |
|--|---|
| Número tarea: 40 | Historia de usuario (No. 18): Procesar encuestas utilizando técnicas estadísticas |
| Nombre tarea: Obtener respuestas de las preguntas con técnica seleccionada de la BD del sistema de encuesta | |
| Tipo de tarea: Desarrollo | Puntos estimados: 0.3 |
| Fecha inicio: 12/2/2016 | Fecha fin: 14/2/2016 |
| Programador responsable: Rafael Alexander Mendoza Fernández | |
| Descripción: Esta tarea facilita la obtención de las respuestas de la BD del sistema de encuestas. Una vez que el procesador selecciona la opción procesar se obtienen las respuestas de las preguntas con técnicas seleccionada de la BD del sistema de encuesta. | |

Tabla 17 TI Procesar las respuestas con la técnica seleccionada

| Tarea de ingeniería | |
|--|---|
| Número tarea: 41 | Historia de usuario (No. 18): Procesar encuestas utilizando técnicas estadísticas |
| Nombre tarea: Procesar las respuestas con la técnica seleccionada | |
| Tipo de tarea: Desarrollo | Puntos estimados: 1 |
| Fecha inicio: 14/2/2016 | Fecha fin: 21/2/2016 |
| Programador responsable: Rafael Alexander Mendoza Fernández | |
| Descripción: Esta tarea facilita procesar las respuestas con la técnica seleccionada. Una vez obtenidas las respuestas de las preguntas se procesan utilizando la técnica seleccionada en cada pregunta. | |

2.3.5 Requisitos no funcionales

Al concebir las historias de usuario no se detallan ciertos aspectos que se deben precisar en la elaboración de una aplicación. Estos aspectos influyen en gran medida en el funcionamiento del producto final. La metodología XP no define artefactos para los requisitos no funcionales⁷, sin embargo, propone que los detalles de implementación de la historia de usuario se tienen en cuenta en el mismo momento de la concepción. Para un mejor funcionamiento es necesario que el producto cumpla con cualidades y propiedades; por estas razones se describen a continuación, algunas de las características no funcionales que debe poseer.

⁷ Según (Sommerville, y otros, 2005) «los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Son las características que hacen al producto atractivo, usable, rápido y confiable».

Capítulo 2: Características de la propuesta de solución

RNF-1 El sistema debe permitir la fácil interacción y llegar de manera rápida y efectiva a la opción deseada. Debe poseer además una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia, una rápida adaptación.

RNF-2 Apariencia o interfaz externa: La interfaz externa del producto será de fácil navegación por el usuario, sencilla y legible. Su funcionamiento deberá ser intuitivo y requerir información mínima. Se garantizará la uniformidad de las interfaces, teniendo en cuenta que las operaciones comparables siempre se activen de la misma forma. Se utilizarán conceptos y términos obtenidos de la experiencia de las personas que más utilizarán el sistema. Incluirá mecanismos para recuperarse ante errores. De igual forma cuando se cometan errores la interfaz proporcionará características de ayuda. Se incluirán elementos visuales para la selección de información siempre que sea posible para minimizar los posibles errores.

RNF-3 El sistema debe seguir estándares para mantener una coherencia y estilo común entre todas las páginas. Interfaz sencilla con colores suaves a la vista y sin cúmulo de imágenes u objetos que distraigan al cliente del objetivo.

RNF-4 Usabilidad: El sistema podrá ser utilizado por cualquier usuario que posea conocimientos informáticos básicos. Podrán operar sobre los diferentes escenarios del sistema de acuerdo con los permisos que tengan otorgados. El software tendrá una curva de aprendizaje baja, que permita al usuario familiarizarse rápidamente con los elementos del sistema y operarlo de forma correcta en poco tiempo de uso. El usuario debe presionar un máximo de tres clics para llegar a cada de las funcionalidades del sistema y se tendrán “migas de pan” (bread crumbs) para indicar la ubicación exacta que tiene el recurso dentro del sistema.

RNF-5 El sistema como requerimientos mínimos de hardware recomienda, tanto para servidores como para estaciones de trabajo:

Equipo Servidor

- Procesador Core™2 Duo a 2 GHz.
- Memoria RAM 2 GB. La cantidad de memoria RAM varía según la cantidad de usuarios Web.
- Disco Duro con más de 10 Gb libres para los datos que se generan en los procesamientos.

- PHP v5.6 o superior.

Equipo Cliente

- Poseer un navegador web instalado, preferentemente (Mozilla Firefox, Opera, Google Chrome).

2.4 Patrones de diseño

Según (Gamma, y otros, 1995) «*un patrón de diseño abarca una idea completa dentro de un programa, y por lo tanto puede también abarcar las fases de análisis y diseño de alto nivel. Sin embargo, dado que un patrón a menudo tiene una implementación directa en código, podría no mostrarse hasta el diseño de bajo nivel o la implementación*».

Existen otras definiciones como la de los autores (Jiménez López, 2003) «*los patrones de diseño cambian la perspectiva acerca de las posibilidades de diseño de la solución de un problema, permiten que estos sean más flexibles, modulares, reutilizables y comprensibles*» y (Leonart Martín, y otros, 2006) «*un patrón de diseño es una solución probada que se puede aplicar con éxito a un determinado tipo de problemas que aparece repetidamente en el desarrollo de sistemas de software*».

Por lo que (Potencier, y otros, 2008) enuncian para la implementación de *Symfony* se utilizan varios patrones, situándolos en las capas de Modelo y Controlador que plantea el patrón arquitectónico MVC, entre los patrones utilizados se encuentran:

2.4.1 Patrones GRASP

Los patrones GRASP (Patrones Generales de Software para Asignar Responsabilidades), por lo que plantea (Grosso, 2011), describen los principios fundamentales de la asignación de responsabilidades a objetos.

- **Experto:** Es uno de los patrones que más se utiliza cuando se trabaja con *Symfony*, con la inclusión de la librería Doctrine para mapear la Base de Datos. *Symfony* utiliza esta librería para realizar su capa de abstracción en el modelo, encapsular toda la lógica de los datos y generar las clases con todas las funcionalidades comunes de las entidades, las clases de abstracción de datos (Peer del

Modelo) poseen un grupo de funcionalidades que están relacionadas directamente con la entidad que representan y contienen la información necesaria de la tabla que representan.

- **Creador:** En la clase *Controller* se encuentran las acciones definidas para el sistema y se ejecutan en cada una de ellas. En dichas acciones se crean los objetos de las clases que representan las entidades, lo que evidencia que la clase *Controller* es “creador” de dichas entidades.
- **Alta Cohesión:** *Symfony* permite la organización del trabajo en cuanto a la estructura del proyecto y la asignación de responsabilidades con una alta cohesión. Un ejemplo de ello es la clase *Controller*, la cual está formada por varias funcionalidades que están estrechamente relacionadas, siendo la misma la responsable de definir las acciones para las plantillas y colaborar con otras para realizar diferentes operaciones, instanciar objetos y acceder a las propiedades.
- **Bajo Acoplamiento:** La clase *Controller* hereda únicamente de *sfController* para alcanzar un bajo acoplamiento de clases. Las clases que implementan la lógica del negocio y de acceso a datos se encuentran en el modelo, las cuales no tienen asociaciones con las de la vista o el controlador, lo que proporciona que la dependencia en este caso sea baja.

2.4.2 Patrones GoF

Según (Gamma, y otros, 1995) «los patrones GOF (*pandilla de los cuatro, siglas en inglés*) son una forma indispensable de enfrentarse a la programación para buscar solución a un problema determinado».

- **Patrón Singleton:** Clase *sfRouting* – método *getInstance* Esta clase la utiliza el controlador frontal (*sfWebFrontController*) y se encarga de enrutar todas las peticiones que se hagan a la aplicación. El *singleton sfRouting* precisa otros métodos muy útiles para la gestión manual de las rutas.
- **Patrón Decorator:** Este método pertenece a la clase abstracta *sfView*, padre de todas las vistas, que contienen un decorador para permitir agregar funcionalidades dinámicamente. El archivo nombrado *layout.php* es el que contiene el *layout* de la página. Este archivo, conocido también como plantilla global, guarda el código HTML que es usual en todas las páginas del sistema, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el *layout*, o si se mira desde

el otro punto de vista, el *layout* decora la plantilla. Este procedimiento es una implementación del patrón *Decorator*.

- **Patrón Registry:** Este patrón es muy útil para los desarrolladores en la Programación Orientada a Objetos. Este patrón es un medio sencillo y eficiente de compartir datos y objetos en la aplicación sin la necesidad de preocuparse por conservar numerosos parámetros o hacer uso de variables globales. Este patrón se aplica en el fichero *Parameters*, que es el encargado de acumular todas las variables de uso global en el sistema.

2.5 Arquitectura del sistema

Según (Bass, y otros, 2003) «*la arquitectura de software es la estructura o las estructuras del sistema que incluyen los componentes del software, las propiedades visibles externamente de esos componentes y las relaciones entre ellos*».

Por otra parte (Pressman, 2005) enuncia que la estructura de los datos y los componentes del programa requeridos para construir un software se representa en el diseño arquitectónico. El mismo autor establece el estilo arquitectónico, la estructura y las propiedades de los componentes que comprende, y las interrelaciones que tienen lugar entre los componentes arquitectónicos del sistema. Además, sugiere para las aplicaciones web, el patrón de diseño arquitectónico Modelo-Vista-Controlador (MVC), encargado de desacoplar la interfaz de usuario de la funcionalidad y el contenido de la aplicación.

Tomando los elementos de que *Symfony* está basado, según (Potencier, y otros, 2008), en un patrón clásico del diseño web conocido como arquitectura MVC se define el uso de la propia arquitectura (Figura 5), la cual se describe a continuación según los autores (Reynoso, 2004) y (González Díaz, y otros, 2012):

Vista

Es la encargada a grandes rasgos de proveer la interacción con el usuario y facilitar funcionalidades como:

- Mostrar datos, formatearlos, ordenarlos.
- Solicitar datos, validarlos.
- Informar de los errores lógicos y de ejecución.
- Controlar la navegación entre pantallas.

Capítulo 2: Características de la propuesta de solución

Esta capa en el sistema de análisis de datos sería la encargada de visualizar al cliente toda la información de las encuestas y los resultados y especificaciones de los procesamientos realizados.

Controlador

Esta capa contiene la funcionalidad que implementa la aplicación. Involucra cálculos basados en la información dada por el usuario, datos almacenados y validaciones. Controla la ejecución de la capa de acceso a datos y servicios externos. Se puede diseñar la lógica para uso directo por parte de componentes de presentación o su encapsulamiento como servicio y llamada a través de una interfaz de servicios que coordina la conversación con los clientes del servicio o invoca cualquier flujo o componente de negocio.

Esta capa en el sistema sería la encargada de ajustar las funcionalidades para realizar los procesamientos de las encuestas seleccionadas por los clientes. Se encontrarían también las clases controladoras encargadas de manejar la comunicación entre la vista y el modelo.

Modelo

Es un grupo de clases y de componentes responsables del almacenamiento de los datos, esta incluye necesariamente un modelo de las entidades del dominio del negocio. Es la representación real de los datos y representa además la persistencia del estado del sistema.

Esta capa representaría todas las clases entidades. Cubriría también aquellas clases o repositorios que almacenan las consultas para acceder a los registros de la base de datos.

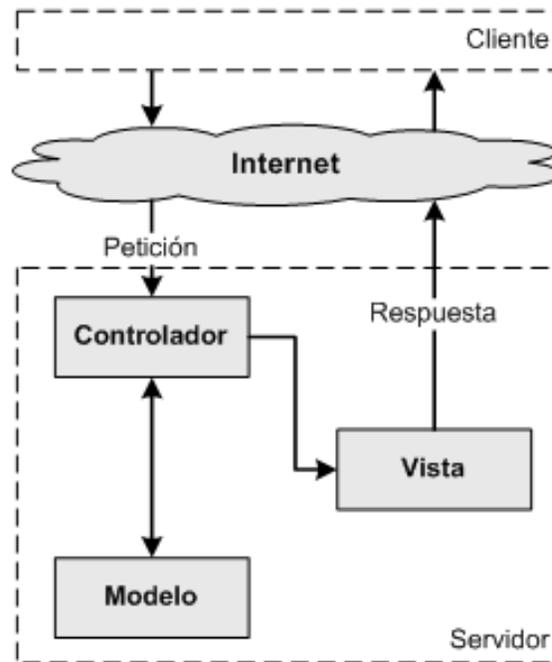


Figura 5: Patrón MVC [tomado de (Potencier, y otros, 2008)]

Conclusiones parciales

- La sistematización del procesamiento estadístico de la información que se genera en el Centro de Innovación y Calidad de la Educación permitió conocer que el *LimeSurvey* solo utiliza una técnica estadística, por lo que los especialistas realizan el procesamiento manual.
- Una vez definidas las características para el desarrollo de la solución, fueron concebidas las historias de usuario para lograr una planificación en el desarrollo de la propuesta de solución.
- La metodología XP plantea que las tarjetas CRC son elaboradas para las clases más importantes de la aplicación, con el objetivo de plasmar sus responsabilidades y colaboradores, de modo que para la propuesta de solución fue necesario desarrollar 13 tarjetas CRC.
- El levantamiento de las funcionalidades de la propuesta, así como de los requisitos no funcionales y las historias de usuarios, permitió que se definieran los patrones de diseño y la arquitectura MVC, utilizados para lograr una correcta estructuración del sistema.

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

3.1 Introducción

En el siguiente capítulo se muestran las particularidades necesarias para describir el proceso de la implementación. Para ello se definen los estándares de codificación y se diseña el diagrama de despliegue para explicar la distribución de la aplicación desarrollada. Se realizan pruebas unitarias y pruebas de aceptación para comprobar el cumplimiento de los requerimientos establecidos en las historias de usuarios.

3.2 Estándares de codificación

XP enfatiza en la comunicación de los programadores a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación para mantener el código legible entre los miembros del equipo de desarrollo facilitando de esta manera los cambios que se puedan presentar.

En la solución propuesta se tendrá en cuenta los siguientes estándares de codificación limitándose al uso de los estándares utilizados por *Symfony2* (Figura 6), los cuales están, según (Potencier, y otros, 2008), definidos en los documentos PSR-0, PSR-1, PSR-2.

1. Los archivos sólo deben utilizar una codificación UTF-8.
2. Los nombres de las variables, métodos y clases deben expresar el propósito de dicho elemento.
3. Los nombres de clases deben ser escritas utilizando la técnica *StudlyCaps*.
4. Métodos y funciones deben ser escritos utilizando la técnica *camelCase*.
5. La indentación debe ser con un tabulador establecido a 4 espacios.
6. Se escribe únicamente una declaración por línea.
7. El número de caracteres por línea deben ser de 80 columnas, aunque también está aceptado que sean hasta 120.

```
class GestionarUsuariosController extends Controller
{
    /**
     * @Route("/administrar/usuarios",name="adminUser")
     * @Security("has_role('ROLE_listUser')")
     */
    public function usuariosAction()
    {
        $em=$this->getDoctrine()->getManager();
        $consulta = $em->createQuery("SELECT r FROM ucildapBundle:Roles r WHERE r.rol<>'Super Administrador'");
        $consulta->useResultCache(true);
        $roles=$consulta->getResult();
        return $this->render('ucildapBundle:Administrar:usuarios.html.twig',array('roles'=>$roles));
    }
}
```

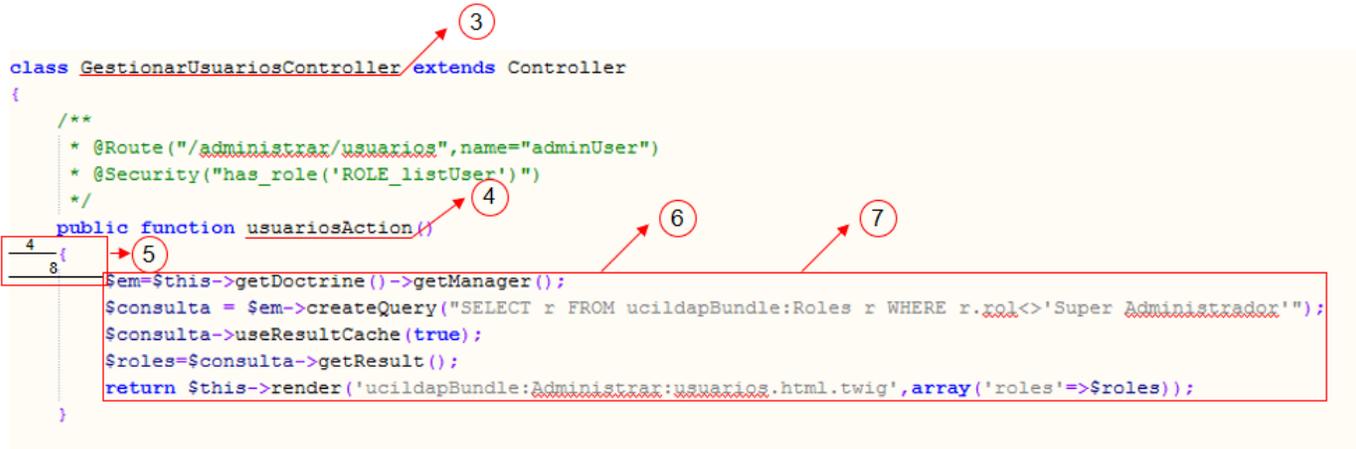
The image shows a code snippet for a Symfony controller. Red circles with numbers 3 through 7 are placed around the code, with red arrows pointing to specific elements: 3 points to the class name 'GestionarUsuariosController', 4 points to the '@Route' annotation, 5 points to the function name 'usuariosAction()', 6 points to the Doctrine EntityManager object '\$em', and 7 points to the SQL query string.

Figura 6 Utilización de los estándares de codificación

3.3 Diagrama de despliegue

Los diagramas de despliegue (Figura 6), por lo que plantean (Jacobson, y otros, 2000), son los complementos de los diagramas de componentes que unidos proveen la vista de implementación del sistema. Describen la topología del sistema la estructura de los elementos de hardware y el software que ejecuta cada uno de ellos. Los diagramas de despliegue representan a los nodos y sus relaciones. Los nodos son conectados por asociaciones de comunicación tales como enlaces de red, conexiones TCP/IP.

El usuario desde una estación de trabajo podrá acceder al sistema utilizando un navegador web, el sistema estará desplegado en el mismo servidor web donde se encuentre ubicada la aplicación del LimeSurvey. Dicho servidor estará conectado a un servidor BD PostgreSQL en el cual se almacenará la información de interés para la solución y al servidor de BD MySQL del LimeSurvey de donde obtiene los resultados de las encuestas.

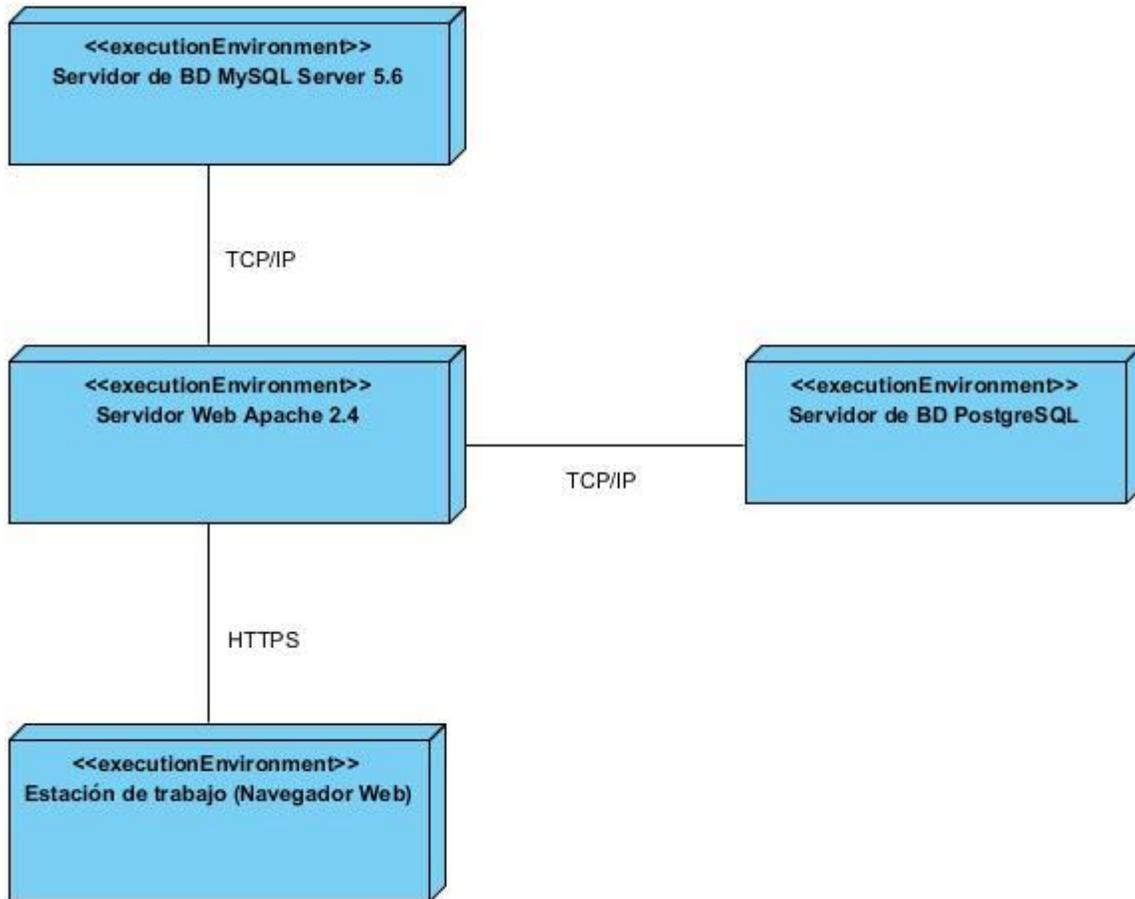


Figura 7: Diagrama de despliegue

3.4 Pruebas

Las pruebas de software, por lo que plantea (Carreiras Ruiz, 2008), «son un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación. Dichas pruebas son realizadas con el objetivo de detectar errores en el sistema, por lo que se llevan a cabo durante todo el ciclo de vida del producto. Los casos de prueba especifican una forma de probar el sistema, incluyendo las entradas con las que se ha de probar, las condiciones bajo las que ha de probarse, así como los resultados esperados».

Autores como (Myers, y otros, 2011) afirman que las «pruebas de software son el proceso de ejecución de un programa o sistema con la intención de encontrar errores». Según (Letelier, y otros, 2012) entre las prácticas de XP, se encuentra la de llevar a cabo un Desarrollo Guiado por Pruebas (TDD, por sus siglas en inglés), pues esta reduce el número de errores no detectados, así como el tiempo entre la introducción de estos en el sistema y su descubrimiento. Los propios autores plantean además que, en XP, se pueden clasificar las pruebas en 2 grupos principalmente, las pruebas unitarias y las pruebas de aceptación.

3.4. 1 Pruebas unitarias

Por lo que plantea (Jurado, 2010) las pruebas unitarias o test unitarios son lo más importantes para el practicante TDD. Cada test unitario o test unidad es un paso andado en el camino de la implementación del software. Todo test unitario debe ser: atómico, independiente, inocuo y rápido. Si no cumple estas premisas entonces no es un test unitario, aunque se ejecute con una herramienta tipo *xUnit*. Las pruebas unitarias son una forma de probar el buen funcionamiento de un módulo o una parte del sistema, con el fin de asegurar el correcto funcionamiento de todos los módulos por separado y evitar así errores futuros en el momento de la integración de todas sus partes.

Una de las actividades que conforma la fase de pruebas en XP es la refactorización, constante actividad de reestructuración del código. Dicha actividad tiene como objetivo remover la duplicidad, mejorar la legibilidad, simplificar y hacer más flexible la codificación, para facilitar los posteriores cambios.

Las pruebas unitarias son realizadas por los desarrolladores en todo momento del desarrollo según propone la metodología utilizada. Para realizar las pruebas se utilizó la librería *PHPUnit* que es la librería utilizada por el *framework* de desarrollo y de fácil integración con el IDE seleccionado. Fueron realizadas 44 pruebas de las cuales 33 fueron satisfactorias y 11 no satisfactorias, los resultados de algunas de las pruebas realizadas se muestran a continuación.

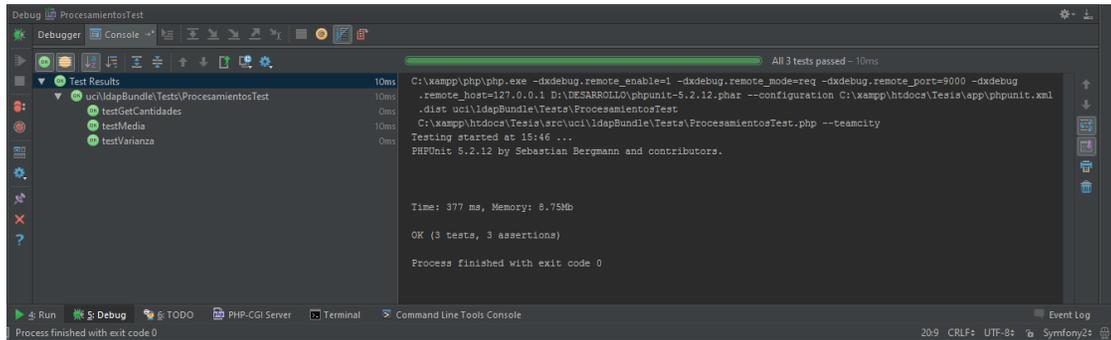


Figura 8: Pruebas unitarias *procesamientosTest*

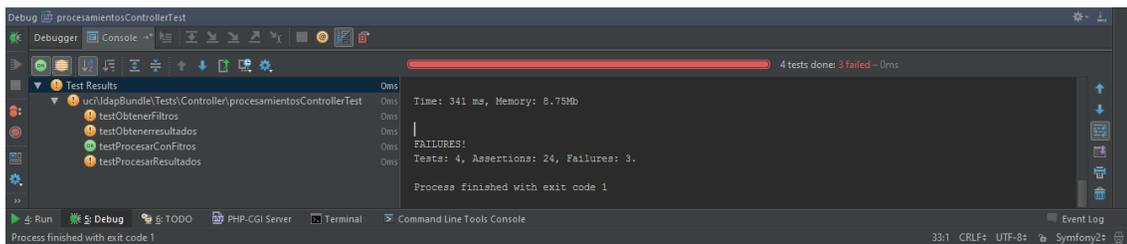


Figura 9: Pruebas unitarias *procesamientosControllerTest*

3.4.2 Pruebas de aceptación

El propio autor (Jurado, 2010) aborda que las pruebas de aceptación o test de aceptación permiten comprobar que el software cumple con un requisito de negocio. Una funcionalidad escrita con el lenguaje del cliente pero que puede ser ejecutada por la máquina. Estas pruebas son creadas a partir de las historias de usuario. Las mismas son el punto de partida del desarrollo en cada iteración. Por otra parte (Tobón Echeverry, 2007) proyecta las pruebas de aceptación como pruebas de caja negra, que representan un resultado esperado de determinada transacción con el sistema. Cada una de las pruebas representa una salida esperada del sistema, donde es responsabilidad del cliente verificar la corrección de las pruebas y tomar decisiones acerca de las mismas.

Las pruebas de aceptación son realizadas por el cliente, como propone la metodología bajo la cual se realiza el presente trabajo de diploma. Estas se realizan por casos de pruebas basándose en los requerimientos que se recogen en las historias de usuario. El objetivo de las pruebas es validar que la aplicación

desarrollada realice de forma eficiente las funciones para las que ha sido creada en base a los requerimientos planteados por el usuario. A continuación, se evidencian una de las pruebas de aceptación realizadas a la aplicación, las demás pueden ser encontradas en el Anexo 4.

Tabla 18 Prueba de aceptación HU18_PA18

| Prueba de aceptación | |
|--|---|
| Código: HU18_PA18 | H.U: Procesar encuestas utilizando técnicas estadísticas |
| Nombre: Procesar encuestas utilizando técnicas estadísticas | |
| Nombre de la persona que realiza la prueba: Harold García Alfonso | |
| Descripción de la prueba: Probar que se procesan las preguntas de las encuestas seleccionada | |
| Condición de ejecución: El usuario debe estar autenticado y pertenecer a un rol con permiso de procesar encuestas | |
| Entrada / Pasos de ejecución: El usuario selecciona la opción procesar | |
| Resultado esperado: Se procesan las respuestas y se crea un fichero xml con el resultado | |
| Evaluación: Satisfactoria | |

3.4.3 Registro de No conformidades

Al realizar las pruebas, uno de los detalles a no pasar por alto son las no conformidades detectadas, las cuales se traducen en los errores encontrados y funcionalidades no deseadas por el cliente. Al final de cada iteración se le muestra al cliente una versión funcional del software de forma que pueda detectar aquellas no conformidades que serán corregidas al inicio de la subsiguiente iteración. La presente investigación está dividida en 3 iteraciones, a continuación, se listan las no conformidades encontradas en cada una de ellas.

Tabla 19 Registro de No conformidades

| No conformidades | |
|------------------|---|
| Iteración | |
| 1 | Al adicionar un usuario permite adicionar usuarios fuera del LDAP de la universidad. |
| | Al eliminar un rol se eliminan todos los usuarios de ese rol. |
| | Al adicionar un rol no se le asignan los permisos seleccionados. |
| | Al listar los roles no salían los permisos que tienen cada uno. |
| 2 | Al ver los detalles de las encuestas disponibles los resultados salían duplicados. |
| | Al guardar las configuraciones se guardan mal las relaciones de las técnicas y las preguntas. |
| | Los resultados exportados a pdf no salían completos. |
| | Cuando se procesan las encuestas le cambia las técnicas a las preguntas. |
| 3 | Procesa todos los resultados y no la cantidad seleccionada. |
| | Procesa la cantidad seleccionada pero no de los grupos seleccionados |
| | Siempre se procesan los mismos resultados |

Como parte de la metodología seleccionada, las no conformidades encontradas en cada iteración son las primeras tareas a resolver de la iteración siguiente, siendo el cliente el encargado de ordenarlas por prioridad. Algunas de ellas al no ser críticas, son arrastradas a la siguiente iteración. Llevando a cabo este proceso, se logran minimizar los niveles de aceptación de errores. De esta manera quedaron resueltas las no conformidades detectadas en la aplicación desarrollada.

3.5 Valoración de la contribución de la propuesta

Para poseer un punto de vista del procesamiento realizado sin la utilización del sistema desarrollado se efectuaron entrevistas no estructuradas a especialistas que se encontraban inmersos en el procesamiento, ver Anexo 5. Con los resultados arrojados, fue posible realizar una comparación teniendo en cuenta los indicadores (1) Tiempo de procesamiento, (2) Especialistas involucrados, (3) Presencia de errores y (4) Tiempo para corregir los errores, mostrado en la (Tabla 21).

Tabla 20 Comparación del procesamiento con y sin el sistema informático basado en técnicas de procesamiento de datos cuantitativos

| | Procesamiento sin el sistema | Procesamiento con el sistema |
|---|-------------------------------------|-------------------------------------|
| Tiempo de procesamiento | 60-80 horas | 4 horas |
| Especialistas involucrados | 6 | 1 |
| Presencias de errores | En ocasiones | No |
| Tiempo para corregir los errores | 60-80 horas | 0 |

Con la comparación realizada, fue posible alcanzar una visión sobre las diferencias de la realización del procesamiento con y sin el sistema informático basado en técnicas de procesamiento de datos cuantitativos, demostrando así, que con la utilización del sistema realizado fue posible disminuir el tiempo de procesamiento. Además, la no existencia de errores en los resultados arrojados del sistema, permite que los especialistas empleen el tiempo destinado a la corrección de los errores en la interpretación de los resultados. Con la valoración antes realizada, queda demostrado la disminución del tiempo de procesamiento y el aumento de fiabilidad del análisis estadístico del sistema informático basado en técnicas de procesamiento de datos cuantitativos.

Conclusiones parciales

- La utilización de los diferentes estándares de codificación facilitó el uso del componente desarrollado en otras soluciones, lo que permitió la implementación del código de forma legible para que pueda ser recodificado.
- Con la realización del diagrama de despliegue se obtuvo una representación de la estructura del sistema y el entorno donde estará trabajando.
- La evaluación de la propuesta a través de las pruebas arrojó que las 11 no conformidades encontradas fueron resueltas, lo que posibilitó el correcto funcionamiento del sistema y se verificó las funcionalidades desarrolladas.
- Con la valoración realizada, se constata la disminución del tiempo de procesamiento y el aumento de la fiabilidad del análisis estadístico del sistema informático basado en técnicas de procesamiento de datos cuantitativos.

CONCLUSIONES

La presente investigación tuvo como base el análisis estadístico de la información que se genera por el *LimeSurvey* en el Centro de Innovación y Calidad de la Educación, para ello fueron cumplidas con satisfacción las tareas de investigación planificadas, arribando a las siguientes conclusiones:

- La fundamentación de los referentes teóricos que sustentan el proceso de análisis estadístico de la información que se genera en el Centro de Innovación y Calidad de la Educación y de los sistemas informáticos que aplican técnicas de procesamiento de datos cuantitativos, permitió seleccionar el análisis de datos cualitativos a un nivel de análisis descriptivo. Además, permitió identificar que las técnicas de análisis de datos descriptivas seleccionadas son distribución de frecuencia absoluta, distribución de porcentajes, frecuencia acumulativa, moda, mediana, medio aritmético, varianza y desviación estándar.
- El estudio de los sistemas informáticos que aplican técnicas de procesamiento de datos, brindaron una base de conocimiento para el desarrollo del sistema de análisis de los datos generados por el *LimeSurvey* en el Centro de Innovación y Calidad de la Educación.
- En la caracterización del procesamiento estadístico de la información que se genera del *LimeSurvey* en el Centro de Innovación y Calidad de la Educación, se constata la necesidad de disminuir el tiempo de procesamiento y aumentar la fiabilidad de los datos.
- El empleo del sistema de análisis de datos desarrollado para el procesamiento de la información generada por el *LimeSurvey* en el Centro de Innovación y Calidad de la Educación, constituye una herramienta práctica y fiable durante la etapa del procesamiento de las respuestas, demostrado a través de la valoración y las pruebas realizadas.

RECOMENDACIONES

Luego de cumplir con el objetivo trazado en la investigación y al analizar los resultados que se han obtenido, se recomienda lo siguiente:

- Incluir el procesamiento de las respuestas cualitativas al sistema de análisis de datos desarrollado.
- Incorporar comparaciones entre los procesamientos actuales y los anteriores.

BIBLIOGRAFÍA

- Abrahamsson, P, y otros.** *Agile software development methods- Review and analysis*. Universidad de Oulu. Finlandia : ESPOO, 2002.
- Alonso Ramos, M.** *Coocurrencia lexica y descripcion lexicográfica*. 2001.
- Avison, D.** *Information Systems Development: Methodologies, Techniques, and Tools*. s.l. : McGraw-Hill, 1995.
- Bass, L, Clements, P y Kazman, R.** *Software Architecture in Practice*. s.l. : Addison Wesley, 2003.
- Bausela Herreras, E.** *SPSS: UN INSTRUMENTO DE ANÁLISIS DE DATOS CUANTITATIVOS*. Departamento de Filosofía y Ciencias de la Educación, Universidad de León. España : s.n., 2005.
- Bellin, D y Suchman Simone, S.** *The CRC card book*. s.l. : Addison Weley, 1997.
- Booch, G, Rumbaugh, J y Jacobson, I.** *El Lenguaje Unificado de Modelado*. s.l. : Addison Wesley Iberoamericana, 1999.
- Briones, G.** *Metodología de la investigación cuantitativa en las ciencias sociales*. Bogota, Colombia : s.n., 1996.
- Buendía Eisman, L, Hernández Pina, F y Colás Bravo, P.** *Métodos de investigación en psicopedagogía*. España : McGraw-Hill Interamericana, 1997.
- Canales Mora, R.** *Modelado UML con Visual Paradigm. Adictos al Trabajo*. [En línea] 2 de febrero de 2004. view-source:<http://www.adictosaltrabajo.com/tutoriales/vparadigm/>.
- Cános, J.** *Métodologías Ágiles en el Desarrollo de Software*. Universidad Politécnica de Valencia. Valencia : s.n., 2005.
- Carreiras Ruiz, M.** *Estimación del Coste de la Calidad del Software a través de la Simulación de Procesos de Desarrollo*. 2008.
- Computación, Departamento de Sistemas Informáticos.** Universidad Politécnica de Valencia. Valencia : s.n., 2011.
- Cossío Cárdenas, G.** *Las herramientas de software para el análisis cuantitativo de información. Estudio preliminar de aplicaciones desarrolladas en Cuba*. 2009.
- Devijver, P y Kittler, J.** Londres : Prentice-Hall, 1982.
- Dictionary, Larousse Spanish.** *Diccionario Manual de la Lengua Española*. 2007.
- Dulzaides Iglesias, M y Molina Gómez, A.** *Análisis documental y de información: dos componentes de un mismo proceso*. Ciudad de La Habana : s.n., 2004. Vol. 12.

- Dunglas, K.** *Persistence in PHP with the Doctrine ORM*. s.l. : Packt Publishing Ltd, 2013.
- Gadja, W.** *Instant PhpStorm Starter*. Birmingham : Packt Publishing Ltd, 2007.
- Gamma, E, y otros.** *Design Patterns: Elements of Reusable Object-Oriented Software*. s.l. : Addison-Wesley Professional, 1995. Vol. 49.
- Gimson, L.** *Metodologías ágiles y desarrollo basado en conocimiento*. Facultad de informática, Universidad Nacional de La Plata. Argentina : s.n., 2012.
- Giráldez Reyes, R., Díaz Pére, M. y Armas Peña, D.** *PROInTec: un software para el tratamiento inteligente de datos sobre patentes*. 2008.
- González Díaz, Y y Fernández Romero, Y.** *Patrón Modelo-Vista-Controlador*. 2012.
- Grosso, A.** Prácticas de software. [En línea] 21 de 3 de 2011. <http://www.practicadesoftware.com.ar/2011/03/patrones-grasp/>.
- Jacobson, I, Booch, G y Rumbaugh, J.** *El proceso unificado de desarrollo de software*. 1. 2000. Vol. 7.
- Jiménez López, R.** *Análisis y Diseño Orientado a Objetos de un*. Facultat de Psicologia, Universitat de les Illes Balears. 2003. Tesis doctoral.
- Johnson, R y Foote, B.** *Designing reusable classes*. s.l. : Journal of Object-Oriented Programming, 1998. Vol. 1.
- Joskowicz, J.** *Reglas y Prácticas en eXtreme Programming*. Universidad de Vigo. España : s.n., 2008. Doctorado de Ingeniería Telemática.
- Jurado, C.** *Diseño Ágil con TDD*. 2010.
- Kent, B y Ward, C.** *A laboratory for teaching Object-Oriented thinking*. New Orleans : s.n., 1989.
- Kerlinger, F.** *Investigación del Comportamiento. Técnicas y Metodología*. México : Interamericana, 1983.
- Larman, C.** *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. s.l. : Addison Wesley Professional, 2004.
- Letelier, P y Penádez, M.** Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). [En línea] 2012. [Citado el: 13 de 2 de 2016.] <http://www.cyta.com.ar/ta0502/v5n2a1.htm>.
- Lleonart Martín, E y García-Menacho Rovira, A.** *Patrones*. España : s.n., 2006.
- Ltd, VSN International.** GenStat 12th edition. Getting Started Guide. [En línea] Julio de 2009. [Citado el: 10 de enero de 2016.] <http://www.vsni.co.uk/downloads/genstat/release12/doc/GettingStartedGuide.pdf>.
- Magaña, S.** *Estudio comparativo de Lenguajes de Modelado de Procesos de Negocio para su integración en Procesos de Desarrollo de Software dirigido por modelos*. Universidad Carlos II. Madrid : s.n., 2009.

- Milestone.** Milestone. [En línea] 2009. [Citado el: 29 de 11 de 2015.] <http://www.milestone.com.mx/CursoModeladoNegociosBPMN.htm>.
- Monroy Fernández, M.** *Modelo de comportamiento de la organización virtual: una aplicación empírica a los sistemas de franquicia*. Universidad de Las Palmas de Gran Canaria. 2003.
- Moya Pardo, M y Gómez Gómez, H.** *Análisis de datos*. 2008.
- Myers, G y Badgett, T.** *The art of software testing*. s.l. : John Wiley & Sons, 2011.
- Okeke Oluchukwu, E.** What's the best editor/IDE for PHP. *Quora*. [En línea] 10 de Octubre de 2015. <https://www.quora.com/Whats-the-best-editor-IDE-for-PHP>.
- Paradigm, Visual.** *Visual paradigm for uml. Visual Paradigm for UML-UML tool for software application development*. 2013.
- Péladeau, N.** *QDA miner. Qualitative data analysis software user's guide*. Canada : Provalis Research, 2004.
- Penadés, C y Letelier, P.** *Ingeniería del software y sistemas de información*. 2014.
- Pérez Váldez, D.** Los diferentes lenguajes de programación para la web. [En línea] 2 de Noviembre de 2007. <http://www.maestrosdelweb.com/los-diferentes-lenguajes-de-programacion-para-la-web/>.
- Pérez, C.** Manual de usuario, plataforma de encuestas en línea: LimeSurvey. 2007.
- Potencier, F y François, Z.** *Symfony, la guía definitiva*. 2008.
- Pressman, R.** *Ingeniería de Software un Enfoque Práctico*. Madrid : McGraw-Hill, 2005.
- Reynoso, C.** *Introducción al Patrón de Arquitectura por Capas*. Argentina : s.n., 2004. Vol. 33.
- Rodríguez Gómez, G, y otros.** *Análisis de datos cualitativos asistido por ordenador: AQUAD y NUDIST*. 1995.
- Rouse, M.** Análisis de datos: Definición. [En línea] 20 de Noviembre de 2012. [Citado el: 21 de 11 de 2015.] <http://searchdatacenter.techtarget.com/es/definicion/Analisis-de-Datos>.
- Sabino, C.** *El proceso de investigación*. Caracas : Panapo, 1992.
- Sarduy Domínguez, Y.** *Revista Cubana de Salud Pública*. 2013.
- Sommerville, I y Alfonso, M.** *Ingeniería del software*. s.l. : Pearson Educación, 2005.
- Tinoco Gómez, O, Rosales Lopez, P Pablo y Salas Bacalla, J.** *Criterios de selección de metodologías de desarrollo de software*. 2010.
- Tobón Echeverry, L.** *Caso práctico de la metodología ágil XP al desarrollo de software*. 2007.

UBA, Departamento de Matemática - FCEyN -. Estadística descriptiva. [En línea] 2004.
http://www.dm.uba.ar/materias/probabilidades_estadistica_C/2004/1/PyEC12.pdf.

Vallejo Morales, P. *La fiabilidad de los tests y escalas*. Facultad de Ciencias Humanas y Sociales, Universidad Pontificia Comillas. Madrid : s.n., 2007.

Vink, M. SQL support and Database tools. [En línea] 2 de Noviembre de 2012.
<http://blog.jetbrains.com/webide/2012/11/sql-support-and-database-tools/>.

Wilkinson, L. *SYSTAT for Windows: statistics, graphics, data, getting started, version 5*. Systat : s.n., 1992.

GLOSARIO DE TÉRMINOS

Algoritmo: Del nombre del matemático árabe Al-Khwarizmi (780 - 850 aprox.). Define el conjunto de instrucciones que sirven para ejecutar una tarea o resolver un problema. Los motores de búsqueda usan algoritmos para mostrar los resultados de búsquedas.

Doctrine: Es un Mapeador de Objeto Relacional (ORM) creado para PHP.

Información: Agregación de datos que tiene un significado específico más allá de cada uno de éstos. Un ejemplo: 1, 9, 8 y 7 son datos; 1987 es una información. La información ha sido siempre un recurso muy valioso, revalorizado hoy más aún por el desarrollo y la expansión de las Tecnologías de la Información y de las Comunicaciones.

HTML (Lenguaje de Marcado de Hipertexto): Lenguaje en el que se escriben las páginas a las que se accede a través de navegadores WWW. Admite componentes hipertextuales y multimedia.

Subsistema: Conjunto de elementos interrelacionados que, en sí mismo, es un sistema, pero a la vez es parte de un sistema superior.

TCP/IP: Protocolo de comunicaciones estándar en Internet. (Transmission Control Protocol/Internet Protocol).

Tecnología: Conjunto de conocimientos técnicos, científicamente ordenados, que permiten diseñar y crear bienes y servicios, para facilitar la adaptación al medio ambiente y satisfacer tanto las necesidades esenciales como los deseos de la humanidad.