



Facultad Regional Mártires de Artemisa

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

Título: Colaboración en línea en tiempo real
usando el marco de trabajo jWebSocket.

Autor:

Daimi Mederos Llanes

Tutores:

MSc. Yamila Vigil Regalado

Lic. Gilberto Ramón Justiniani Fernández

Artemisa, Junio 2012

DECLARACIÓN DE AUTORÍA

Declaro que soy la única autora de este trabajo y autorizo a la Facultad Regional “Mártires de Artemisa” de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Daimi Mederos Llanes

Msc. Yamila Vigil Regalado

Firma del Autor

Firma del Tutor

Lic. Gilberto R. Justiniani Fernández

Firma del Tutor



"Los estudiantes son en su mayoría revolucionarios. Revolucionarios por naturaleza, porque pertenecen a ese estrato de jóvenes que se abren a la vida y que adquieren todos los días conocimientos nuevos".

Ernesto Che Guevara

AGRADECIMIENTOS

Agradezco a mi familia, por el apoyo incondicional que me han brindado en todos estos años, en especial a mis padres que siempre han estado a mi lado y me han dado todo su cariño y a mi hermana del alma Dainerys.

Agradezco a todos mis amigos que me han acompañado a lo largo de estos 5 años de la carrera.

Agradezco a los tutores de esta investigación que me ayudaron a desarrollar la misma con la calidad requerida.

Agradezco a todos los profesores que de una manera u otra han aportado a mi formación como profesional.

En fin agradezco a todo aquel que ha hecho posible la realización de este sueño.

DEDICATORIA

A mis padres por educarme como una persona de bien, por su incondicional e infinito amor, por la confianza que siempre depositaron en mí, por inspirarme en todo momento y guiarme a ser todo lo que soy.

A mi hermana por ser mi amiga y darme un sobrino tan bello.

A mi sobrinito Javier que lo quiero con la vida.

A mi familia en general por apoyarme y darme todo su cariño.

A mis amigos, no hace falta especificar, ellos saben quiénes son.

RESUMEN

El surgimiento de la Web 2.0 trajo consigo el inicio de las comunidades de usuarios y una gama de servicios que fomentan la colaboración y el intercambio ágil de información. Surge así la colaboración en línea que ofrece posibilidades favorables para los usuarios. Las aplicaciones de colaboración en línea constituyen un software que une a varias personas con el fin de facilitar el flujo de información y la realización de trabajos en conjunto. Permiten la sincronización, la edición colaborativa, el uso compartido de archivos, los paneles de discusión, entre otros.

Actualmente existen numerosas herramientas web que posibilitan este tipo de colaboración. Pero debido a las nuevas exigencias de interoperabilidad y conectividad permanente en la Web, se hace necesaria la colaboración en línea en tiempo real. Para esto surge el protocolo de comunicación WebSocket que proporciona un canal de comunicación bidireccional, permitiendo la entrada y salida de datos de manera simultánea. Uno de los marcos de trabajo que utilizan este protocolo es jWebSocket. Este constituye una tecnología orientada al desarrollo de aplicaciones basadas en WebSockets. Proporciona altos niveles de velocidad, escalabilidad, seguridad y el trabajo en tiempo real, que es el elemento clave para la colaboración en línea hoy en día.

En esta investigación se presenta el proceso de desarrollo de Shared Canvas Demo en su primera versión. Esta es una aplicación web demostrativa que garantiza el posible aumento del nivel de usabilidad del marco de trabajo jWebSocket en el desarrollo de aplicaciones de colaboración en línea en tiempo real.

ÍNDICE DE CONTENIDO

| | |
|---|----|
| INTRODUCCIÓN | 1 |
| CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA | 9 |
| 1.1. Conceptos Asociados al Dominio del Problema..... | 9 |
| 1.2. Análisis de Soluciones Existentes | 11 |
| 1.3. Metodología usada para el desarrollo de la solución | 17 |
| 1.4. Herramientas y Tecnologías usadas para el desarrollo de la solución..... | 20 |
| 1.4.1. Marco de Trabajo del lado del Servidor..... | 20 |
| 1.4.2. Marco de Trabajo del lado del Cliente | 22 |
| 1.4.3. Lenguajes de Programación y Modelado | 23 |
| 1.4.4. Otros Lenguajes | 24 |
| 1.4.5. Herramienta CASE | 25 |
| 1.4.6. Herramienta de Control de Versiones | 26 |
| 1.4.7. Cliente de Control de Versiones..... | 27 |
| 1.4.8. Entorno Integrado de Desarrollo | 28 |
| CAPÍTULO 2. CARACTERÍSTICAS, ANÁLISIS Y DISEÑO DEL SISTEMA | 30 |
| 2.1. Propuesta de Solución | 30 |
| 2.2. Planificación del Proyecto por Roles | 32 |
| 2.3. Modelo de Historias de Usuario del Negocio..... | 35 |
| 2.4. Lista de Reserva del Producto (LRP) | 36 |
| 2.5. Historias de Usuario y Tareas de Ingeniería | 38 |
| 2.6. Plan de Releases | 43 |
| 2.7. Descripción de la Arquitectura..... | 44 |
| 2.8. Diseño con Metáforas..... | 45 |
| 2.9. Diagrama de Componentes..... | 46 |
| CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DEL SISTEMA | 48 |
| 3.1. Diagrama de Despliegue | 48 |
| 3.2. Validación de la Solución Propuesta | 49 |
| 3.3. Resultados Obtenidos | 57 |
| 3.4. Funcionalidades Obtenidas | 57 |
| 3.5. Aporte Social y Económico..... | 58 |

| | |
|--|----|
| CONCLUSIONES | 59 |
| RECOMENDACIONES | 60 |
| BIBLIOGRAFÍA REFERENCIADA | 61 |
| BIBLIOGRAFÍA CONSULTADA | 67 |
| ANEXOS | 69 |
| GLOSARIO DE TÉRMINOS | 70 |

ÍNDICE DE TABLAS Y FIGURAS

| | |
|---|----|
| Tabla 1: Planificación del Proyecto por Roles | 32 |
| Tabla 2: Lista de Reserva del Producto..... | 36 |
| Tabla 3: Descripción de la Historia de Usuario Dibujar | 39 |
| Tabla 4: Descripción de la Tarea de Ingeniería 1.1 | 39 |
| Tabla 5: Descripción de la Tarea de Ingeniería 1.2 | 39 |
| Tabla 6: Descripción de la Tarea de Ingeniería 1.3 | 40 |
| Tabla 7: Descripción de la Historia de Usuario Insertar Imágenes | 40 |
| Tabla 8: Descripción de la Tarea de Ingeniería 2.1 | 41 |
| Tabla 9: Descripción de la Tarea de Ingeniería 2.2 | 41 |
| Tabla 10: Descripción de la Tarea de Ingeniería 2.3 | 41 |
| Tabla 11: Descripción de la Historia de Usuario Limpiar Lienzos..... | 42 |
| Tabla 12: Descripción de la Tarea de Ingeniería 3.1 | 42 |
| Tabla 13: Descripción de la Tarea de Ingeniería 3.2 | 43 |
| Tabla 14: Plan de Releases..... | 43 |
| Tabla 15: Descripción del Caso de Prueba # 1 para la HU Dibujar | 50 |
| Tabla 16: Descripción del Caso de Prueba # 2 para la HU Dibujar..... | 50 |
| Tabla 17: Descripción del Caso de Prueba # 3 para la HU Dibujar | 51 |
| Tabla 18: Descripción del Caso de Prueba # 1 para la HU Insertar Imágenes | 51 |
| Tabla 19: Descripción del Caso de Prueba # 2 para la HU Insertar Imágenes | 52 |
| Tabla 20: Descripción del Caso de Prueba # 3 para la HU Insertar Imágenes | 52 |
| Tabla 21: Descripción del Caso de Prueba # 1 para la HU Limpiar Lienzos | 53 |
| Tabla 22: Descripción del Caso de Prueba # 2 para la HU Limpiar Lienzos | 54 |
| | |
| Fig. 1 Propuesta de Solución..... | 31 |
| Fig. 2 Modelo de Historias de Usuario del Negocio | 35 |
| Fig. 3 Arquitectura de la solución | 44 |
| Fig. 4 Diagrama de Paquetes | 45 |
| Fig. 5 Diagrama de Componentes..... | 46 |
| Fig. 6 Diagrama de Despliegue | 48 |

INTRODUCCIÓN

La Web desde su surgimiento ha estado en constante evolución. Esta ha transitado de la complejidad técnica para publicar a la facilidad para producir y compartir. Inicialmente, en el período de 1994 a 1997, estuvo vigente la Web 1.0 caracterizada por ser estática. El usuario jugaba un papel pasivo pues solo consultaba la información y las páginas web eran documentos estáticos que no se actualizaban. Luego surgió la Web 1.5, presente de 1997 al 2003, esta es conocida como la Web Dinámica. Las páginas web eran construidas dinámicamente a partir de una o varias bases de datos.(PAZ, 2005)

Esta versión dio paso a la Web 2.0 que emergió en el 2003 y se caracterizó por ser colaborativa, dinámica y participativa. El usuario es protagonista activo, creando y compartiendo contenidos. La Web 2.0 está basada en comunidades de usuarios y una gama especial de servicios que fomentan la colaboración y el intercambio ágil de información entre los usuarios, convertidos en contribuidores que publican las informaciones y realizan cambios en los datos.(PAZ, 2005) (Portal de Marketing Directo, 2011)(BAROJA, 2011)

Con el término Web 2.0 se subraya un cambio de paradigma sobre la concepción de Internet y sus funcionalidades. Estas dejan de ser unidireccionales y se orientan a facilitar la máxima interacción entre los usuarios y el desarrollo de redes sociales donde puedan expresarse, opinar, buscar y recibir información de interés, colaborar, crear conocimiento y compartir contenidos.(Conferencia Web 2.0 "Revvig the Web", 2004) La Web 2.0 es la representación de la evolución de las aplicaciones tradicionales hacia aplicaciones web enfocadas al usuario final.(¿Qué es la Web 2.0?, 2005) Las nuevas herramientas de la Web 2.0 están caracterizadas por la interacción y colaboración entre los usuarios, contribuyendo a una socialización de la red.

Estas potencialidades que brinda la Web 2.0 permitieron el surgimiento del trabajo colaborativo en línea. Sincronización, uso compartido de archivos, edición

colaborativa y paneles de discusión son sólo algunas de las características que puede tener una herramienta de colaboración en línea. Estas son aplicaciones amigables que facilitan el trabajo sin requerir de mucho entrenamiento ni conocimientos tecnológicos.

Una herramienta colaborativa es un software que une a varias personas con el fin de facilitar el flujo de información y la realización de trabajos en conjunto. Las primeras fueron las soluciones de *groupware* que aparecieron a finales de los 80 de la mano de Lotus Notes¹. El término *groupware* hace referencia a los métodos y herramientas de software que facilitan el trabajo en grupo, mejorando su rendimiento y contribuyendo a que personas que están localizadas en puntos geográficos diferentes puedan trabajar a la vez, ya sea directamente o de forma anónima a través de las redes. Hoy en día la mayoría de las herramientas colaborativas están basadas en Internet.(BAROJA, 2011)

La colaboración en línea se convierte en un elemento cada vez más importante en la economía y la sociedad de hoy. El compartir información y conocimientos, son componentes vitales de una verdadera relación de colaboración. El *groupware* busca apoyar el trabajo que se realiza por grupos y equipos, teniendo en cuenta los aspectos de la colaboración que son necesarios para cosechar el máximo de ventajas. Es una herramienta tecnológica muy útil en estos tiempos ya que se pueden integrar los conocimientos teóricos con la tecnología. Con el crecimiento de las empresas surge la necesidad de aumentar el número de personas que deben contar con información actualizada. Es por ello que compartir archivos, sincronizar documentos y tareas y administrar canales de comunicación se vuelve cada vez más importante para alcanzar el éxito en el sector empresarial.

Hoy día no solo se consulta e intercambia información a través de la Web, sino que constituye el medio de comunicación primordial. Al aumentar considerablemente el número de usuarios en línea y el acceso a la Web desde distintos dispositivos,

¹Lotus Notes: Sistema cliente/servidor de colaboración y correo electrónico, desarrollado por Lotus Software, filial de la empresa multinacional International Business Machines (IBM).

tanto estacionarios como móviles, la comunicación web exige mayor interactividad, interoperabilidad y tiempo real. La interactividad en la Web exige un intercambio de experiencias en tiempo real, los usuarios desean recibir retroalimentación inmediata de todas las acciones que realizan, conocer al instante los cambios del estado del tiempo, información de sus amigos, de sus negocios. Esto hace que la comunicación no se establezca siempre ante una solicitud del usuario, sino que la comunicación sea proactiva, que la Web pueda comunicarse con ellos sin una acción precedente.

Es por esto que surge la Web 3.0 o Web Semántica, que apareció por primera vez en el año 2006 en un artículo de Jeffrey Zeldman, con la idea de añadir metadatos a la World Wide Web. Esas informaciones adicionales que describen el contenido, el significado y la relación de los datos, se deben proporcionar de manera formal, para que así sea posible evaluarlas automáticamente por máquinas de procesamiento. El objetivo es mejorar Internet ampliando la interoperabilidad entre los sistemas informáticos usando agentes inteligentes².(VELASCO, 2011)

En consecuencia las herramientas de colaboración en línea necesitan evolucionar para que cumplan con las nuevas exigencias de la Web. Es por esto que se hace necesaria la colaboración en línea en tiempo real. Pero las herramientas actuales que propician este tipo de colaboración trabajan sobre el protocolo de comunicación HTTP, que no brinda esta posibilidad. HTTP fue diseñado como un simple protocolo de solicitud-respuesta. Este no proporciona una comunicación bidireccional entre el cliente y el servidor. De esta forma si se genera un nuevo evento en el servidor, este no es capaz de enviarle directamente un mensaje al cliente de manera asíncrona. En todos los casos el cliente debe realizar una solicitud para obtener una respuesta del servidor, ocasionando demoras en la entrega de datos actualizados al usuario final.(Comet-HTML5-Websocket, 2011)

El tiempo real en la Web se posibilita cuando el cliente recibe mensajes del servidor sin solicitud previa y cuando los usuarios finales reciben actualizaciones de forma

²Agentes inteligentes: Programas en las computadoras que buscan información sin operadores humanos.

simultánea.(Framework Approach for WebSockets, 2011) Para brindar solución a la comunicación web en tiempo real se crea el protocolo WebSocket. Este es una tecnología que proporciona un canal de comunicación bidireccional y *full-duplex*³ sobre un único *socket*⁴ TCP. Está diseñada para ser implementada en navegadores y servidores web, pero puede utilizarse por cualquier aplicación cliente/servidor.

WebSocket soluciona las limitaciones del protocolo HTTP al establecer una comunicación *full-duplex* (TCP) entre el cliente y el servidor, sustituyendo la comunicación *half-duplex*⁵ (HTTP). Se reduce en grandes proporciones el tráfico en la red teniendo en cuenta que al establecer la comunicación WebSocket entre el cliente y el servidor solo hay un envío de 2 bits, eliminando las cabeceras HTTP.(HTML5 Websockets and communication, 2010)

Ante las nuevas exigencias de la Web se pronostica que en poco tiempo WebSocket y sus beneficios reemplazarán al protocolo HTTP y sus limitaciones.(HTML5 Websockets and communication, 2010)(Framework Approach for WebSockets, 2011) Algunos de los servidores que soportan WebSocket para el desarrollo de aplicaciones son, la Pasarela WebSockets de Kaazing⁶, Jetty WebSocketServlet⁷, Socket.IO⁸, django-websocket del proyecto Python⁹ y jWebSocket¹⁰.

El marco de trabajo jWebSocket es una nueva tecnología orientada al desarrollo de aplicaciones basadas en WebSockets que proporcionan altos niveles de velocidad, escalabilidad, seguridad y el trabajo en tiempo real, elemento clave para la colaboración en línea hoy día.(Framework Approach for WebSockets, 2011) El proyecto jWebSocket, que impulsa el desarrollo de este marco de trabajo, fue creado en el año 2009 por lo que se considera un proyecto reciente. Esto trae

³ Full-Duplex: Cualidad de los elementos que permiten la entrada y salida de datos de forma simultánea.

⁴Socket: Método para la comunicación entre un programa del cliente y un programa del servidor en una red.

⁵Half-Duplex: Significa que el método o protocolo de envío de información es bidireccional pero no simultáneo.

⁶<http://kaazing.com>

⁷<http://www.eclipse.org/jetty>

⁸<http://socket.io>

⁹<http://pypi.python.org/pypi/django-websocket>

¹⁰<https://jwebsocket.org>

consigo que las potencialidades brindadas por WebSocket para la colaboración en línea no se aprovechen en aplicaciones o herramientas web de este tipo. En consecuencia muchas empresas y desarrolladores no utilizan el marco de trabajo como vía para desarrollar aplicaciones de colaboración en línea que se caractericen por trabajar en tiempo real.

Atendiendo a los elementos anteriores, se describe la siguiente **situación problemática**:

WebSocket es un proyecto de reciente creación. Por su poca madurez no ha logrado promoverse en el desarrollo de aplicaciones de colaboración en línea. Es por esto que en la actualidad no se cuenta con una vía para mostrar la capacidad del marco de trabajo WebSocket para desarrollar aplicaciones web de colaboración en línea en tiempo real. En consecuencia los desarrolladores no identifican las potencialidades y facilidades que ofrece el marco para este tipo de colaboración. Ante esto muchas empresas que desarrollan aplicaciones de colaboración en línea, no visualizan el uso de WebSocket como vía para lograr este fin. Por ello no se aprovecha el marco para mejorar el proceso de colaboración en línea y la disminución de los costos que esto trae consigo. El desconocimiento de las potencialidades de WebSocket para la colaboración en línea en tiempo real trae consigo además un bajo nivel de usabilidad y pérdidas de cuotas de mercado para el marco de trabajo.

Para darle solución a la situación descrita, se plantea como **problema científico**, ¿Cómo garantizar mayor nivel de usabilidad del marco de trabajo WebSocket en el desarrollo de aplicaciones de colaboración en línea en tiempo real? Del problema anterior se puede definir como **objeto de estudio** de esta investigación, la colaboración en línea. Acotando el **campo de acción** a la colaboración en línea en tiempo real con el marco de trabajo WebSocket.

Para ello se plantea como **objetivo general** de la investigación, desarrollar una aplicación web demostrativa de colaboración en línea en tiempo real que garantice

mayor nivel de usabilidad del marco de trabajo jWebSocket en el desarrollo de este tipo de aplicaciones.

Para complementar este objetivo general se definen las siguientes **preguntas científicas**:

1. ¿Cuáles son los fundamentos teórico-metodológicos de la colaboración en línea?
2. ¿Cuál es la situación actual de la colaboración en línea?
3. ¿Cómo desarrollar una aplicación web demostrativa de colaboración en línea en tiempo real que garantice mayor nivel de usabilidad del marco de trabajo jWebSocket en el desarrollo de este tipo de aplicaciones?
4. ¿Cómo comprobar la capacidad y potencialidad de la aplicación web demostrativa de colaboración en línea en tiempo real utilizando el marco de trabajo jWebSocket?

Para dar respuesta a las preguntas anteriores se plantean las siguientes **tareas de la investigación**:

- ✓ Fundamentación teórico-metodológica de la colaboración en línea.
- ✓ Análisis de la situación actual de la colaboración en línea.
- ✓ Desarrollo de una aplicación web demostrativa de colaboración en línea en tiempo real que garantice mayor nivel de usabilidad del marco de trabajo jWebSocket en el desarrollo de este tipo de aplicaciones.
- ✓ Comprobación de la capacidad y potencialidad de la aplicación web demostrativa de colaboración en línea en tiempo real utilizando el marco de trabajo jWebSocket.

Para llevar a cabo esta investigación se utilizan los siguientes **métodos teóricos**:

Histórico-Lógico: Se utiliza para realizar un estudio histórico de la colaboración en línea que permite identificar deficiencias de esta y proponer soluciones acordes a las necesidades.

Análisis Documental: Mediante este método se realiza un estudio de la documentación referente a la colaboración en línea o cualquier otra temática que proporcione conocimientos necesarios que pudieran ser incorporados a la investigación.

Analítico-Sintético: Mediante este método se analiza toda la teoría recopilada que permita lograr un mejor desarrollo de la solución propuesta y poder aplicar estos conocimientos en la práctica, de manera que se adquiriera una mayor preparación sobre el tema en cuestión.

Modelación: Se utiliza para realizar una representación de la situación que se analiza. Permite obtener mediante diagramas y objetos una mayor comprensión del problema y desarrollar un modelo para la aplicación en la que se está trabajando.

Se cuenta además con las siguientes variables:

Variable Independiente: Aplicación web demostrativa de colaboración en línea en tiempo real con el marco de trabajo `WebSocket`.

Variable Dependiente: Nivel de usabilidad del marco de trabajo `WebSocket` para el desarrollo de aplicaciones de colaboración en línea en tiempo real.

Al concluir el trabajo se contará con el siguiente **resultado**:

- ✓ Aplicación web demostrativa de colaboración en línea en tiempo real que garantice mayor nivel de usabilidad del marco de trabajo `WebSocket` en el desarrollo de este tipo de aplicaciones

Para una mejor comprensión de la investigación el contenido ha sido desglosado en tres capítulos, además de las conclusiones generales, recomendaciones, referencias bibliográficas, bibliografía utilizada, anexos y glosario de términos.

Los capítulos han sido estructurados de la siguiente manera:

- ✓ **Capítulo 1. Fundamentación Teórica:** Se realiza la fundamentación teórico-metodológica de la investigación. Se expone un estudio del estado del arte de la colaboración en línea en la actualidad, tanto a nivel nacional como internacional.
- ✓ **Capítulo 2. Características, Análisis y Diseño del Sistema:** Brinda una fundamentación de la solución propuesta, a partir de la cual se describen las actividades de análisis de la solución, seguidas por la descripción de los procesos del sistema y de la etapa de diseño.
- ✓ **Capítulo 3. Implementación y Validación del Sistema:** Se describe la etapa de implementación que conlleva a la obtención de la aplicación. Además como parte de la estrategia de validación trazada se elaboran y documentan las pruebas funcionales, certificación de calidad de software y valoración de la solución propuesta por parte del cliente.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Introducción

El presente capítulo tiene como objetivo, presentar los conceptos y aspectos más significativos abordados en diferentes fuentes bibliográficas acerca del tema que se investiga. Se proporciona una panorámica general acerca de la colaboración en línea y en específico de las aplicaciones web de colaboración en línea en la actualidad. Además se analiza la metodología, tecnologías y herramientas utilizadas para el desarrollo de la solución, siempre teniendo en cuenta la necesidad del uso de herramientas de código abierto.

1.1. Conceptos Asociados al Dominio del Problema

Trabajar en un mismo proyecto con personas que viven en países y continentes distintos es posible gracias a las herramientas o aplicaciones web de colaboración en línea. Estas permiten compartir archivos e información sin mucho entrenamiento ni conocimientos tecnológicos. El objeto de estudio y campo de acción de la presente investigación están enmarcados en este tema. Es por eso que los conceptos fundamentales que orientaron la presente investigación estuvieron referidos a la colaboración en línea, al tiempo real en la Web y al protocolo de comunicación WebSocket que lo posibilita.

La definición de colaboración en línea ha sido dada por varios autores. Entre ellos se encuentra el enfoque brindado por Evan Rosen, para el cual la colaboración en línea es trabajar juntos para crear valores añadidos al compartir un espacio virtual. (ROSEN, 2007) Michael Weinzettl plantea que la colaboración en línea permite a un grupo de personas con un objetivo común, trabajar juntos a través de Internet.(WEINZETTL, 2010) Por último se puede citar a Magnus Ramage para el cual la colaboración en línea combina habilidades, aptitudes y experiencias a través de las tecnologías de la comunicación y la colaboración; permitiendo intercambiar, compartir y crear valores agregados efectivos para los usuarios.(RAMAGE, 2010) Esta última definición es asumida como concepto de colaboración en línea para la

presente investigación porque brinda elementos más completos y enfocados al objetivo de la investigación.

En los últimos años han surgido aplicaciones web de colaboración en línea cada vez más participativas. En consecuencia muchas de las herramientas que posibilitan el trabajo colaborativo en la Web necesitan utilizar tiempo real. Los preceptos que sustentan el tiempo real en la Web han sido expuestos por varias personalidades reconocidas en el tema. Lambert M. Surhone y otros autores plantean que el tiempo real en la Web se brinda a través de un conjunto de tecnologías y prácticas que permiten a los usuarios recibir información tan pronto como se publique por sus autores, en lugar de comprobar una fuente de actualizaciones periódicamente.(SURHONE, y otros, 2010) Se puede citar además el enfoque dado por Alexander Schulze para el cual el tiempo real en la Web se posibilita cuando el cliente recibe mensajes del servidor sin solicitud previa y los usuarios finales reciben actualizaciones de forma simultánea.(Framework Approach for WebSockets, 2011) Esta última definición es asumida como concepto de tiempo real en la Web para la presente investigación porque fue dada por el líder internacional del proyecto `jQueryWebSocket` y plantea elementos más concretos.

En la actualidad se utiliza en la Web el protocolo de comunicación HTTP que no proporciona una comunicación bidireccional entre el cliente y el servidor. Con el uso de este protocolo si se genera un nuevo evento en el servidor, este no es capaz de enviarle directamente un mensaje al cliente de manera asíncrona. En todos los casos el cliente debe realizar una solicitud, para obtener una respuesta del servidor, ocasionando demoras en la entrega de datos actualizados al usuario final.(Comet-HTML5-Websocket, 2011) Para poder brindar tiempo real en la Web es necesario introducir el uso de un protocolo que brinde esta posibilidad. Es por esto que se crea el protocolo `WebSocket`.

Para la IETF `WebSocket` permite la comunicación bidireccional entre un cliente que ejecuta código no confiable en un ambiente controlado y un anfitrión remoto que ha optado por comunicaciones desde ese código. El modelo de seguridad utilizado

para ello es el modelo de seguridad basado en el origen comúnmente utilizado por los navegadores web. Este protocolo consiste en un apretón de manos de apertura, seguida por la estructuración en capas del mensaje básico a través de TCP. El objetivo de esta tecnología es proporcionar un mecanismo para aplicaciones basadas en navegadores que necesitan una comunicación bidireccional con los servidores que no dependen de la apertura de múltiples conexiones HTTP.(FETTE, y otros, 2011)

Según Petter Lubbers WebSocket es una tecnología que proporciona un canal de comunicación bidireccional y *full-duplex* sobre un único *socket* TCP. Este protocolo soluciona las limitaciones de HTTP, al establecer una comunicación *full-duplex* (TCP) entre el cliente y el servidor, sustituyendo la comunicación *half-duplex* (HTTP). Se reduce en grandes proporciones el tráfico en la red teniendo en cuenta que al establecer la comunicación WebSocket entre el cliente y el servidor solo hay un envío de 2 bits, eliminando las cabeceras HTTP.(HTML5 Websockets and communication, 2010) Esta última definición es asumida como concepto de WebSocket para la presente investigación porque expone claramente las ventajas que tiene este nuevo protocolo sobre HTTP.

1.2. Análisis de Soluciones Existentes

La colaboración a través de Internet ha alcanzado un gran protagonismo en los últimos años, especialmente con el auge de las redes sociales, la Web 2.0 y más recientemente la Web 3.0. El surgimiento de la Web 2.0 representó la evolución de las aplicaciones tradicionales hacia aplicaciones web enfocadas en el usuario final. No se trata de una nueva tecnología sino de una actitud de colaboración y participación de las personas para proporcionar mejores datos, nuevos servicios y aplicaciones en línea.(FLORES, y otros, 2009)

El principal valor que ofrecen las aplicaciones Web 2.0 es la simplificación de la lectura y escritura en línea. Esto se traduce en dos acciones sustantivas del proceso de aprendizaje: generar contenidos y compartirlos.(Planeta Web 2.0. Inteligencia colectiva o medios fast food, 2007) En este sentido se considera la

posibilidad de utilizar aplicaciones Web 2.0 para el desarrollo de proyectos colaborativos en línea que faciliten la colaboración y participación activa de las personas en la generación y publicación de contenido.

Estas aplicaciones han permitido que se multipliquen las posibilidades de aprender y compartir conocimientos de forma colectiva, provocando una conmoción en el ámbito educativo y del conocimiento, así como nuevas formas de evaluar objetivos y competencias.(FLORES, y otros, 2009) Además el término Web 3.0 del que se ha comenzado a hablar hace unos años, introduce el incremento de los usuarios en la Web y la necesidad de los mismos de compartir y colaborar. Esto trae consigo que la comunicación instantánea, la interoperabilidad, la interactividad y la conectividad permanente sean requisitos fundamentales para la nueva colaboración en línea.

En los últimos años han surgido multitud de aplicaciones web para permitir que equipos que no trabajan físicamente juntos puedan colaborar, comunicarse, compartir información y documentos. Este tipo de aplicaciones resulta de gran utilidad en muchas situaciones tales como la comunicación entre equipos de trabajo situados en distintas oficinas, teletrabajo, colaboración en línea con socios, clientes y proveedores. Algunas de las facilidades que puede brindar una herramienta de colaboración en línea son: uso compartido de archivos, administración de proyectos, acceso a la misma información en el mismo momento y trabajar sobre ella, sincronización, edición colaborativa y paneles de discusión. Cada aplicación tiene sus particularidades que la hacen más aconsejable para un cierto tipo de equipos o trabajos, pero en líneas generales destacan las siguientes:

Herramientas para el Trabajo Colaborativo

Este tipo de herramientas permiten gestionar proyectos en forma colaborativa. Son redes privadas virtuales que usan protocolos de Internet para compartir de forma segura información con empleados, proveedores, clientes, socios, entre otros.(LANGER, 2010) A continuación se presentan algunos ejemplos de herramientas para el trabajo colaborativo:

- ✓ **Teambox**¹¹: Herramienta de colaboración y comunicación en línea. Incluye gestión de proyectos, tareas y archivos. Es una solución de uso muy intuitivo. Permite centralizar y organizar las comunicaciones de un grupo de manera sencilla. Tiene versión gratuita y en idioma español. No requiere instalación. Cuenta con algunas restricciones: la versión gratuita es bastante limitada y no tiene funciones avanzadas de gestión de proyectos y estadísticas.
- ✓ **Basecamp**¹²: Probablemente es una de las aplicaciones de colaboración más utilizadas en el mundo. Es muy eficaz a la hora de gestionar proyectos en forma colaborativa. Constituye una aplicación sencilla que funciona desde un navegador, permitiendo a los distintos usuarios conectarse desde cualquier lugar con acceso a Internet. Adicionalmente se pueden gestionar aplicaciones para equipos móviles, celulares y hasta clientes de correo como Gmail. Tiene versión en idioma español y no requiere instalación. Cuenta con restricciones tales como: la versión gratuita es limitada y no incluye funciones avanzadas de gestión de proyectos y documentos.
- ✓ **FengOffice**¹³: Es una herramienta completa y flexible, pero su curva de aprendizaje es mayor por lo que cuesta algo más aprovechar todas sus funcionalidades. Es una plataforma de colaboración, gestión de proyectos, gestión de clientes, facturación, finanzas y archivo de documentos. Permite gestionar eficientemente gran cantidad de proyectos y documentos simultáneamente. Tiene versión gratuita y en idioma español. No requiere instalación. Cuenta con algunas limitaciones tales como: la versión gratuita para código abierto requiere de instalación en el servidor del usuario.
- ✓ **CentralDesktop**¹⁴: Esta plataforma brinda la posibilidad de crear conversaciones privadas, proteger esa red privada a través de contraseñas, gestionar contenidos y hacer un seguimiento de los proyectos con los

¹¹<http://teambox.com>

¹²<http://basecamphq.com>

¹³<http://www.fengoffice.com/web/community>

¹⁴<http://www.centraldesktop.com>

clientes en un entorno seguro y reservado. Se pueden organizar documentos, compartirlos, editarlos y archivarlos en línea. Es un software de colaboración en línea y una solución de gestión de proyectos para las empresas. Permite gestionar equipos, proyectos, documentos, tareas y clientes.

Herramientas para Compartir Archivos

Estas herramientas constituyen aplicaciones con carácter de necesidad básica. Sirven para intercambiar archivos que quedan almacenados en la nube¹⁵ virtual. Algunas de estas herramientas permiten no sólo compartir archivos en línea, sino también crearlos y editarlos de esta manera. La gran ventaja de estas aplicaciones es que se puede acceder a los documentos desde cualquier lugar y computadora, e inclusive desde teléfonos inteligentes. (LANGER, 2010) Seguidamente se presentan varios ejemplos de herramientas para compartir archivos:

- ✓ **GoogleDocs**¹⁶: Aplicación ofimática con la que se puede compartir y crear documentos con procesadores de texto, planillas de cálculo y presentaciones de diapositivas. Acepta sus equivalentes de Microsoft Office.
- ✓ **GoogleCalendar**¹⁷: Esta es una aplicación que consiste en una agenda en línea. Se pueden planificar tareas, recordatorios de citas importantes, cumpleaños, entre otros. Incluso es posible configurar alertas para que notifiquen esos recordatorios. Esta información se puede compartir y enviar por correo electrónico.
- ✓ **4shared**¹⁸: Es una herramienta que posibilita el almacenamiento de información. La misma es publicada en un servidor para luego compartir con quien se desee, mediante correo electrónico o enlaces en la propia red. Se pueden almacenar archivos de texto, audio, video y fotos. La diferencia con

¹⁵Nube: Se refiere a plataformas que ofrecen servicios que son accesibles a través de internet. Estos servicios pueden ir desde el almacenamiento de archivos, a la edición de documentos ofimáticos, reproducción de música, entre otros. Basta con tener un equipo con acceso a internet para acceder a ellos a través de un navegador.

¹⁶<http://www.docs.google.com>

¹⁷<http://www.google.com/calendar>

¹⁸<http://www.4shared.com>

otras herramientas es que la información que se publica es guardada en un lugar físico, ya sea la computadora del usuario, el teléfono o la cámara fotográfica. Para acceder a los archivos se debe descargar la información a la computadora, una vez allí se pueden ver y editar.

- ✓ **Dropbox**¹⁹: Esta aplicación permite almacenar, sincronizar y compartir documentos. Al igual que la anterior no es una aplicación para crear documentos. Estos archivos se comparten de forma instantánea, accediendo de forma inmediata a ellos. La otra persona con la que se comparte la información puede editarla, visualizándose el cambio inmediatamente.

Wikis

Las Wikis son una de las herramientas de colaboración y participación más poderosas de la Web 2.0. Permiten generar y publicar contenido fácilmente. Este puede ser actualizado en línea por cualquier persona que sea miembro de un equipo de trabajo o una comunidad de aprendizaje, desde cualquier lugar con acceso a Internet. Todos los miembros se encargan de la supervisión y control del contenido publicado, favoreciendo de esta forma la autoría en equipo.(PARKER, y otros, 2008) A continuación se presentan algunos ejemplos de wikis:

- ✓ **Wikipedia**²⁰: Autodefinida como un esfuerzo colaborativo por crear una enciclopedia gratis, libre y accesible por todos. Permite revisar, escribir y solicitar artículos.(Fundación Wikimedia, Inc., 2001)
- ✓ **Wikcionario**²¹: Proyecto colaborativo para producir un diccionario multilingüe libre en cada lengua, con significados, etimologías y pronunciaciones.(Fundación Wikimedia, Inc., 2002)
- ✓ **Wikilibros**²²: Tiene por objetivo poner a la disposición de cualquier persona libros de texto, manuales, tutoriales u otros textos pedagógicos de contenido libre y de acceso gratuito.(Fundación Wikimedia, Inc., 2003)

¹⁹<https://www.dropbox.com>

²⁰<http://es.wikipedia.org>

²¹<http://es.wiktionary.org>

Estas herramientas de colaboración en línea han llegado también a diferentes sectores de nuestro país. Se pueden citar algunas como:

- ✓ **Proyectos UTEEDA²³**: Es una aplicación web puesta en marcha en el año 2007. Está destinada a la planificación del trabajo y la gestión en línea de proyectos, orientado al trabajo colaborativo de equipos virtuales. Incluye todas las herramientas necesarias para la creación, control y monitoreo de proyectos, pero también para la gestión de los equipos que trabajan en ellos. Posee una variedad de herramientas anexas dirigidas a facilitar el trabajo colaborativo como son: calendario grupal, chat, fórum, gestión de archivos, entre otras. El usuario sólo requiere un navegador web para acceder a toda la información sin necesidad de almacenar ningún documento en su computadora. Con respecto al trabajo colaborativo, permite el control y seguimiento de las versiones de los documentos almacenados en el sistema.(Las tecnologías de información y comunicación en el seguimiento y evaluación de los desastres naturales: Estudio de un caso: La plataforma informática de la red UTEEDA para la gestión de la información sobre desastres, 2007)
- ✓ **Infomed 2.0²⁴**: Es una nueva etapa del Portal de Salud de Cuba que surgió en el año 2007. Trabaja para construir colectivamente un ecosistema de personas, servicios y fuentes de información para la salud, sostenido por herramientas de información, comunicación y colaboración de excelencia. Brinda servicios como Infomed Wiki para construir contenidos de manera colectiva y documentar procesos. Además incluye documentos normativos, glosarios, libros, entre otros.(URRA, 2007)
- ✓ **Aulas Hermanas²⁵**: Proyecto colaborativo internacional dirigido a estudiantes entre 12 y 17 años. Promueve y fomenta contactos educativos a partir de la interacción y la construcción colectiva entre dos aulas de

²²<http://es.wikibooks.org>

²³<http://uteeda.isch.edu.cu>

²⁴<http://www.sld.cu>

²⁵<http://aulashermanas.org>

diferentes países, a través de las plataformas y tecnologías de Internet. Su objetivo es que aulas de diferentes zonas geográficas trabajen conjuntamente y en forma virtual durante varios meses, en torno a un tema concreto de investigación. A través de una estrategia de trabajo cooperativo, las aulas comparten conocimientos, ideas, experiencias y aprenden a trabajar junto a otros equipos hermanos por medio de Internet. Su primera versión fue en el año 2010, participaron escuelas de países como México, Cuba y Nicaragua. (Ministerio de Informática y Comunicaciones, 2012)

De manera general estas herramientas o aplicaciones de colaboración en línea presentan algunas restricciones. Utilizan el protocolo de comunicación HTTP que no proporciona tiempo real durante el proceso de colaboración. Lo que trae consigo que el cliente deba realizar una solicitud para obtener una respuesta del servidor. Este hecho ocasiona demoras en la entrega de datos actualizados al usuario final durante el proceso de generación y publicación de contenidos, sincronización, uso compartido de archivos, gestión de proyectos, entre otros. Por esto se hace necesario iniciar una investigación que logre diseñar e implementar una aplicación web de colaboración en línea con el nuevo protocolo WebSocket que brinde la posibilidad de trabajar en tiempo real, en este caso utilizando el marco de trabajo jWebSocket.

1.3. Metodología usada para el desarrollo de la solución

Todo proceso de desarrollo de una aplicación informática debe estar regido y orientado por una metodología de desarrollo de software, que guíe los procesos y permita tener un registro detallado del avance de la investigación. Las metodologías pueden ser robustas o ágiles. Las robustas o pesadas están concebidas para guiar el proceso de desarrollo de software de gran envergadura. Por su parte las metodologías ágiles se enfocan en los clientes y los resultados. Se basan en promover iteraciones en el desarrollo a lo largo de todo el ciclo de vida del proyecto y logran que se minimicen los riesgos desarrollando software en corto tiempo.(Metodologías Tradicionales Vs. Metodologías Ágiles, 2011) Teniendo en

cuenta los aspectos anteriores se realizó un análisis de la metodología robusta RUP y las ágiles SCRUM, XP y SXP para identificar cual es la más apropiada para el desarrollo de la aplicación web demostrativa que posibilita la colaboración en línea en tiempo real.

Proceso Unificado de Desarrollo

RUP es un proceso formal que provee un acercamiento disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo. Es una de las metodologías robustas. Su objetivo es asegurar la producción de software de alta calidad que satisfaga los requerimientos de los usuarios finales, respetando cronograma y presupuesto. Puede ser adaptado y extendido para satisfacer las necesidades de la organización que lo adopte. Es guiado por casos de uso, centrado en la arquitectura, iterativo e incremental.(FIGUEROA, y otros, 2011)

Consta de 4 fases principales:

- ✓ **Inicio:** el objetivo en esta etapa es determinar la visión del proyecto.
- ✓ **Elaboración:** en esta etapa el objetivo es determinar la arquitectura óptima.
- ✓ **Construcción:** en esta etapa el objetivo es llegar a obtener la capacidad operacional inicial.
- ✓ **Transición:** el objetivo es llegar a obtener el despliegue del proyecto.

Programación Extrema

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.(WELLS, 2009)

SCRUM

SCRUM es un proceso ágil y liviano que sirve para administrar y controlar el desarrollo de software. El desarrollo se realiza de forma iterativa e incremental (una iteración es un ciclo corto de construcción repetitivo). Cada ciclo o iteración termina con una pieza de software ejecutable que incorpora nuevas funcionalidades. SCRUM se enfoca en priorizar el trabajo en función del valor que tenga para el negocio, maximizando la utilidad de lo que se construye y el retorno de inversión. Sus principales características son: desarrollo mediante iteraciones y reuniones sistemáticas a lo largo del proyecto.(SCHWABER, y otros, 2011)

SXP

SXP está compuesta por las metodologías SCRUM y XP, ofreciendo una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva. Esta metodología fomenta el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo.

SXP consta de 4 fases principales:

- ✓ **Planificación-Definición:** se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.
- ✓ **Desarrollo:** es donde se realiza la implementación del sistema hasta que esté listo para ser entregado.
- ✓ **Entrega:** es la puesta en marcha.
- ✓ **Mantenimiento:** se realiza el soporte para el cliente.

De cada una de estas fases se realizan numerosas actividades tales como el levantamiento de requisitos, la priorización de la Lista de Reserva del Producto, definición de las Historias de Usuario, Diseño, Implementación, Pruebas, entre otras; de donde se generan artefactos para documentar todo el proceso. Las

entregas son frecuentes, y existe una refactorización continua, lo que permite mejorar el diseño cada vez que se le añada una nueva funcionalidad.

SXP está especialmente indicada para proyectos con pequeños equipos de trabajo, un constante cambio de requisitos o requisitos imprecisos, donde existe un alto riesgo técnico y se orienta a una entrega rápida de resultados y una alta flexibilidad. Fomenta el trabajo en equipo, con un objetivo claro, permitiendo el seguimiento y control de las tareas a realizar.(SXP, Metodología Ágil para el Desarrollo de Software, 2010)

Debido a las grandes ventajas que proporcionan las metodologías ágiles, además de que el proyecto está formado por un equipo de trabajo pequeño, los requisitos son imprecisos y el desarrollo está orientado a una entrega rápida de resultados, se propone para esta investigación el uso de la metodología SXP.

1.4. Herramientas y Tecnologías usadas para el desarrollo de la solución

1.4.1. Marco de Trabajo del lado del Servidor

jWebSocket

Es un marco de trabajo de código abierto para el desarrollo de aplicaciones Web estacionarias y móviles. Este utiliza el lenguaje de programación Java en el lado del servidor permitiendo que los clientes puedan ser desarrollados en múltiples lenguajes tales como JavaScript, C# y Python. jWebSocket establece un modelo de *token*. Los *tokens* son datos abstractos que, a través de una estructura jerárquica y una API, proporcionan métodos de acceso a los contenidos. Con el objetivo de realizar una abstracción en la manipulación de los diferentes formatos, el marco de trabajo convierte los paquetes de datos entrantes y salientes en *tokens*. El cliente nativo soporta el intercambio de paquetes en los formatos JSON, XML y CSV, que en entornos específicos se pueden utilizar sin la necesidad de manejarlos a través de *tokens*. El cliente jWebSocket tiene una arquitectura de complementos que permite aumentar con facilidad sus funcionalidades.

El servidor jWebSocket está diseñado para funcionar como servidor de comunicaciones o como servidor Web, brindando total flexibilidad. En la primera opción jWebSocket proporciona un archivo .jar, ofreciendo la ventaja de ejecutarse fácilmente desde una línea de comandos e integrarse a la biblioteca de una aplicación existente de Java. En la actualidad hay algunos servidores que ya soportan WebSocket y otros que no, por lo que jWebSocket se integra a servidores como Tomcat o Apache para lograr una comunicación WebSocket. En caso de que los servidores soporten de manera nativa WebSocket, como el caso de Jetty o GlassFish, se incluyen las funciones de comunicación del marco de trabajo jWebSocket, pero los motores internos se apagan y el anfitrión se utiliza. Esto asegura que no haya mecanismos de seguridad adicionales.

jWebSocket como servidor Web proporciona un conjunto importante de funcionalidades y su arquitectura extensible mediante complementos permite añadir fácilmente características adicionales a un sistema independiente. Por otra parte los administradores pueden configurar el servidor exactamente como sea necesario y dejar a un lado todos los módulos que no necesiten. En un clúster los componentes se pueden utilizar como servicios, por lo que jWebSocket perfectamente es compatible con SOA en un entorno totalmente basado en eventos. Estas características muestran la fortaleza y flexibilidad del marco de trabajo para el desarrollo de aplicaciones Web estacionarias y móviles, multiplataforma, multisectorial y compatible con todos los navegadores.

El marco de trabajo jWebSocket, a través de la comunicación WebSocket, proporciona altos niveles de velocidad, escalabilidad, seguridad y el trabajo en tiempo real, elemento clave para la colaboración en línea hoy día. Se pronostica que jWebSocket, como paradigma para el desarrollo Web, establezca una nueva dirección de la colaboración y educación en línea y nuevas oportunidades de participación y colaboración social. (Framework Approach for WebSockets, 2011)

jWebSocket es un proyecto del cual un equipo de estudiantes de la Facultad Regional de la UCI “Mártires de Artemisa”, constituye alrededor de un 50% de los

integrantes del grupo de desarrollo de este marco de trabajo. La presente investigación es desarrollada para el proyecto jWebSocket que a su vez forma parte del campo de acción de la misma. Por estas razones y por las potencialidades que brinda jWebSocket para la colaboración en línea, la aplicación a desarrollar se lleva a cabo sobre este marco de trabajo.

1.4.2. Marco de Trabajo del lado del Cliente

jQuery

Es una biblioteca o marco de trabajo de JavaScript. Permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos y desarrollar animaciones, alcanzando grandes resultados en menos tiempo y espacio.(MURPHEY, 2010)

Su característica principal es que permite cambiar el contenido de una página Web sin necesidad de recargarla, mediante la manipulación del árbol DOM y peticiones AJAX. Además posibilita la interactividad y modificaciones del árbol DOM, incluyendo soporte para CSS 1-3. Posee varias utilidades para obtener información del navegador, operar con objetos, vectores, funciones, entre otras. Es compatible con los navegadores Mozilla Firefox 2.0+, Internet Explorer 6+, Safari 3+, Opera 10.6+ y Google Chrome 8+.(Proyecto jQuery, 2010)

Ventajas:(jQuery, la librería Javascript por excelencia. Un framework Javascript lleno de ventajas, 2011)

- ✓ Como buen JavaScript se puede trabajar separado del HTML, haciéndolo más fácil de mantener y enriquecer, aumentando la productividad del proyecto.
- ✓ Su sintaxis se considera liviana, teniendo en cuenta que la sencillez y poca extensión de código es fundamental para los desarrollos.
- ✓ Es un producto serio, estable, bien documentado y con un gran equipo de desarrolladores a cargo de la mejora y actualización del marco de trabajo.

- ✓ Tiene una comunidad de creadores de componentes muy amplia, lo que hace fácil encontrar soluciones ya creadas en jQuery para implementar interfaces de usuario, galerías, votaciones, efectos diversos, entre otros.

Existen otros marcos de trabajo JavaScript como Moo Tools, Prototype, YUI y Ext JS pero para la presente investigación se decide utilizar jQuery como marco de trabajo del lado del cliente. Este contiene una amplia gama de complementos disponibles para cubrir necesidades específicas y para esta investigación se necesitan utilizar algunos de ellos. Además provee de un excelente mecanismo para la captura de eventos, soporta una gran cantidad de selectores como los CSS y permite agregar o quitar fácilmente los atributos de cualquier elemento HTML.

1.4.3. Lenguajes de Programación y Modelado

Lenguaje de Modelado Unificado

UML (Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. Se utiliza para definir un sistema de software y detallar sus artefactos y documentos. (RUMBAUGH, y otros, 2007)

Se selecciona UML porque es el lenguaje utilizado por la herramienta de modelado Visual Paradigm seleccionada para la presente investigación.

Java

Es un lenguaje de programación moderno, orientado a objetos y de alto nivel. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir muchos errores, como la manipulación directa de punteros o memoria. Actualmente este lenguaje es uno de los más utilizados para el desarrollo de aplicaciones

porque es orientado a objetos, robusto, independiente de la plataforma que se utilice, permite una fácil creación y eliminación de objetos, mayor seguridad y fácil portabilidad.(GOSLING, y otros, 2005)

Se selecciona el lenguaje Java por ser utilizado en el lado del servidor del marco de trabajo jWebSocket seleccionado para la presente investigación.

JavaScript

Es un lenguaje de programación interpretado que se utiliza principalmente para crear páginas Web dinámicas. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.(EGUÍLUZ, 2009)

Se selecciona el lenguaje JavaScript por ser utilizado en el lado del cliente del marco de trabajo jWebSocket seleccionado para la presente investigación.

1.4.4. Otros Lenguajes

Lenguaje de Marcado de Hipertexto

HTML (Hyper Text Markup Language) es el lenguaje con el que se crean la mayoría de las páginas Web. Es usado para describir la estructura y el contenido de las páginas en forma de texto, así como para complementar el texto con objetos tales como imágenes. Como se trata de un estándar reconocido por todas las empresas relacionadas con el mundo de Internet, una misma página HTML se visualiza de forma muy similar en cualquier navegador de cualquier sistema operativo. Desde su creación, el lenguaje HTML ha pasado de ser un lenguaje utilizado exclusivamente para crear documentos electrónicos a ser un lenguaje que se utiliza en muchas aplicaciones electrónicas como buscadores, tiendas online y banca electrónica. (EGUÍLUZ, 2009)

Se selecciona HTML porque es necesario para crear las interfaces o vistas de la presente investigación.

Hojas de Estilo en Cascada

CSS (Cascading Style Sheets) es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas Web complejas. Una vez creados los contenidos de una página Web, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, entre otros.(EGUÍLUZ, 2009)

Se selecciona CSS por ser de gran utilidad para controlar el aspecto de las interfaces o vistas de la presente investigación.

1.4.5. Herramienta CASE

Las herramientas CASE propician un conjunto de métodos y técnicas automatizadas que brindan ayuda y dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todo el ciclo de desarrollo de un software. (Herramientas Case, 2011)

Visual Paradigm

Es una poderosa herramienta CASE que hace uso del lenguaje de modelado UML. Es multiplataforma, proporciona un ciclo de vida completo del desarrollo de software, excelentes facilidades de interoperabilidad con otras aplicaciones y compatibilidad entre versiones. Permite crear todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y documentación.

Se integra con herramientas Java como NetBeans que es el IDE seleccionado para la presente investigación. Visual Paradigm permite la generación de código y la ingeniería inversa para un conjunto de lenguajes entre los que se encuentran Java. Cualquiera de los cambios en el código existente puede reflejarse en el modelo y viceversa.(Visual Paradigm International Ltd, 2011)

Se analizaron otras de las herramientas CASE existentes para el modelado de software como Rational Rose, pero para esta investigación se selecciona la herramienta Visual Paradigm. Esta a pesar de no ser libre, es multiplataforma y cuenta con una licencia comercial la cual posee la Universidad de las Ciencias Informáticas.

1.4.6. Herramienta de Control de Versiones

Un sistema de control de versiones permite la gestión de archivos y directorios. Su principal característica es mantener el historial de cambios y modificaciones que se han realizado sobre dichos archivos a lo largo del tiempo. Los sistemas de control de versiones guardan toda la información en un repositorio central accesible a través de la red, permitiendo el trabajo colaborativo entre varios puestos de trabajo y a su vez proporcionando una mayor seguridad y disponibilidad de los datos.

Subversion

Es un sistema de control de versiones completamente equipado. En este sistema los directorios son versionados, la resolución de conflictos es interactiva, gestiona de manera eficaz los archivos binarios, está vinculado con varios repositorios y lenguajes de programación. Además posee bloqueo de archivos, soporte para desarrolladores y desarrollo paralelo.

Dentro de las características planteadas anteriormente se profundiza en varias de ellas debido a la relevancia de las mismas. (Apache Software Foundation, 2011)

- ✓ **Soporte para desarrolladores:** los desarrolladores tienen acceso a todas las funcionalidades de Subversion desde diferentes IDE como Eclipse, NetBeans, entre otros.
- ✓ **Desarrollo Paralelo:** permite el desarrollo paralelo, para que miembros individuales del equipo de trabajo puedan completar las diferentes partes y versiones de un proyecto al mismo tiempo, sin necesidad de esperar por que otro compañero termine las tareas que está realizando.

Teniendo en cuenta las características expuestas, se selecciona Subversion como herramienta de control de versiones para ser utilizado en el desarrollo de la aplicación, ya que permite la integración con NetBeans y brinda la posibilidad de desarrollar en paralelo. También influyó en la decisión, la experiencia de trabajo por parte del equipo de desarrollo con esta herramienta y su fácil uso.

1.4.7. Cliente de Control de Versiones

Permiten la gestión de cambios que se realizan sobre los elementos de algún producto. Permitiendo una conexión entre él como cliente y el servidor de control de versiones, para un mejor manejo de los archivos locales.

RapidSVN

Es una plataforma de interfaz gráfica de usuario, para el sistema de revisión de Subversion. Este proyecto también incluye un cliente de Subversion C++ API. RapidSVN está licenciado bajo la v3 de GPL. Utiliza las mejores características de los clientes de otras arquitecturas de control de versiones.(RapidSVN, 2011)

Se caracteriza por ser:

- ✓ **Simple:** Proporciona una interfaz fácil de usar.
- ✓ **Eficiente:** Simple para los principiantes pero lo suficientemente flexible como para aumentar la productividad para los usuarios de Subversion.
- ✓ **Portátil:** Se ejecuta en cualquier plataforma: Linux, Windows, Mac OS / X, Solaris.
- ✓ **Rápido:** Completamente escrito en C++.

Se tuvo en cuenta otras de las herramientas cliente para el Control de Versiones como TortoiseSVN, pero se decide usar RapidSVN. Esta herramienta es libre y fácil de usar, tanto por quienes ya conocen Subversion como para quienes empiezan a utilizarlo, pudiendo acceder a direcciones SVN, subir y descargar contenido, sincronizarlo con el servidor original, comprobar su estado, crear y fusionar direcciones.

1.4.8. Entorno Integrado de Desarrollo

Un IDE es un entorno de programación que integra varias herramientas con el objetivo de facilitar el desarrollo de software sobre uno o varios lenguajes de programación. Cuentan con herramientas tales como: editor de código, herramientas para el rastreo de código, compilador, depurador y constructor de interfaz gráfica.(NOURIE, 2005)

NetBeans

Está escrito completamente en Java, por lo que puede ser utilizado desde cualquier sistema operativo compatible con la máquina virtual de Java. Permite el desarrollo de aplicaciones de escritorio, Web y móviles. Brinda soporte a varios lenguajes de programación como Java, que es el lenguaje que se utiliza en el lado del servidor en la presente investigación. Es un producto libre y gratuito sin restricciones de uso. Constituye una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Su misión consiste en evitar tareas repetitivas, facilitar la escritura correcta de código, disminuir el tiempo de depuración e incrementar la productividad del desarrollador. Cuenta con un depurador, perfilador de integración, herramientas para refactorizaciones, completamiento de código y control de versiones de archivos. (NetBeans, 2011)

Existen otros IDE que brindan potencialidades para el desarrollo de aplicaciones Java como Eclipse SDK. Sin embargo, para el desarrollo de la solución propuesta se selecciona como entorno integrado de desarrollo a NetBeans debido a que se tiene una mayor experiencia y familiarización con esta herramienta. Además su versión 7.0.1 introduce un soporte para el desarrollo con la especificación JavaSE7 (Java Standard Edition) con las características de JDK7 (Java Development Kit). Esta versión también ofrece una integración mejorada con el servidor Oracle WebLogic, así como soporte para Oracle Database y GlassFish3.1. Otros puntos destacados incluyen soporte para Maven3 y HTML5, así como mejoras en el editor de Java.

Conclusiones del Capítulo

En el presente capítulo se abordaron los conceptos fundamentales que orientan la presente investigación. Se consolidó el basamento teórico mediante un análisis de la actualidad de las aplicaciones web de colaboración en línea, evidenciándose las dificultades de estas. Además se muestran los resultados de un estudio realizado con el objetivo de seleccionar las herramientas, tecnologías y metodología a utilizar, permitiendo sentar las bases para el desarrollo de la aplicación web demostrativa que posibilita la colaboración en línea en tiempo real usando el marco de trabajo `WebSocket`.

CAPÍTULO 2. CARACTERÍSTICAS, ANÁLISIS Y DISEÑO DEL SISTEMA

Introducción

En el presente capítulo se describe cómo debe funcionar el sistema, destacando sus características distintivas. Se realiza el modelado de las historias de usuario del negocio con el objetivo de comprender su contexto. Además se detallan brevemente los artefactos de la metodología de desarrollo SXP, propuesta en el capítulo anterior. Describiéndose entre otros aspectos los requisitos funcionales y no funcionales del sistema, las historias de usuario y las tareas de ingeniería asociadas a las mismas.

2.1. Propuesta de Solución

En la actualidad el proceso de colaboración en línea no se realiza en tiempo real, debido a que la Web utiliza el protocolo de comunicación HTTP que lo imposibilita. Esto impide aprovechar las potencialidades y facilidades que brindarían las aplicaciones de colaboración en línea si fueran de respuesta inmediata. Para demostrar esto en la presente investigación se desarrolla una aplicación web de colaboración en línea que utilice el protocolo de comunicación WebSocket, el cual permite el trabajo en tiempo real.

La aplicación Shared Canvas Demo permite a los usuarios conectados pintar e insertar imágenes sobre lienzos compartidos en tiempo real en la Web. Lo que pinte un usuario o las imágenes que inserte pueden ser vistas por todos los demás. Puede ser usada para colaborar desde la Web en un proyecto en común para los usuarios que estén conectados o para apoyar el proceso de enseñanza-aprendizaje en sus diferentes modalidades y aspectos. Se diferencia de otras aplicaciones web de colaboración en línea porque esta trabaja en tiempo real a través del nuevo protocolo de comunicación WebSocket que lo hace posible. Será desarrollada sobre el marco de trabajo jWebSocket, permitiendo que el usuario compruebe el potencial de jWebSocket para la colaboración en línea en tiempo real. De esta

manera aumentan las posibilidades de lograr mayor nivel de usabilidad del marco de trabajo para este fin.

El usuario puede dibujar a mano alzada utilizando el color que desee e insertar imágenes luego de haberlas seleccionado. Con estas potencialidades que brinda la aplicación se pueden escribir fórmulas matemáticas, realizar diagramas explicativos o simplemente compartir ideas de forma gráfica en tiempo real. Esto permite facilitar el intercambio de conocimientos y mejorar de forma general la calidad del proceso de enseñanza-aprendizaje en la Web. Entre otros aspectos importantes la aplicación podrá ser visualizada desde disímiles lugares del mundo, por miles de personas y en el mismo instante de tiempo, permitiendo mayor alcance y repercusión. A continuación en la Figura 1 se presenta un diagrama de la solución propuesta:

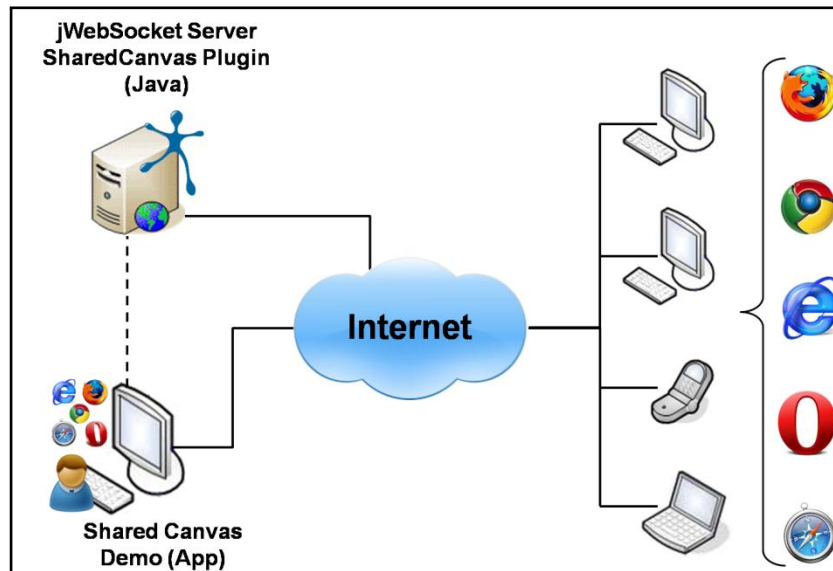


Fig. 1 Propuesta de Solución

Para acceder a Shared Canvas Demo el cliente debe utilizar un navegador que soporte el protocolo WebSocket. Cada acción que realice el usuario sobre el lienzo compartido de la aplicación será enviada al servidor jWebSocket. Este es el encargado de distribuirla a través de Internet hacia los demás clientes, para que puedan ver en tiempo real las acciones que realiza cada usuario. Estos últimos

clientes también deben acceder a la aplicación utilizando navegadores que soporten WebSocket.

2.2. Planificación del Proyecto por Roles

La selección de los roles que intervienen en el ciclo de desarrollo de la solución propuesta es vital para la organización de las responsabilidades y mantener una buena comunicación entre los miembros del proyecto. Estos comparten una visión común del proyecto y se enfocan en implementar la solución con altos estándares de calidad. Seguidamente en la Tabla 1 se presenta la planificación de roles realizada para lograr el desarrollo de la solución propuesta:

Tabla 1: Planificación del Proyecto por Roles

| Rol | Responsabilidades | Nombre |
|--------------------|--|-----------------------|
| Gerente | Es el responsable de tomar las decisiones finales, acerca de estándares y convenciones a seguir durante el proyecto. Participa en la selección de objetivos y requerimientos. Tiene la responsabilidad de controlar el progreso y trabaja junto con el Jefe de Proyecto en la reducción de la Lista de Reserva del Producto. | Alexander Schulze |
| Líder del Proyecto | Debe asegurar que el proyecto se está llevando a cabo de acuerdo con las prácticas y que todo funciona según lo planeado. Su principal trabajo es remover impedimentos y reducir riesgos del producto. | Yamila Vigil Regalado |

| | | |
|-------------------------------------|--|--------------------------------------|
| Especialista (Diseñador Gráfico) | Conoce a fondo el proceso, ya sea de la metodología utilizada o cualquier otro proceso o elementos de gran importancia para el desarrollo de software. Particularmente es una especialización que está activa, el miembro del grupo de trabajo que la desempeña siempre está ejecutándola y alcanzando un grado mayor de conocimientos en el tema. | Rebecca Schulze |
| Consultores | Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto en el que puedan surgir problemas. Además aportan ideas y experiencias para el beneficio del sistema en desarrollo. Esta es una especialización menos activa, quien la ejecuta funciona en este rol por un corto período de tiempo. | Alexander Schulze Rebecca Schulze |
| Cliente | Participa en las tareas que involucran la lista de reserva del producto. | Alexander Schulze |
| Programador | Es el encargado de producir el código y escribir las pruebas | Daimi Mederos Llanes |

| | | |
|----------------------|--|----------------------|
| | unitarias. Debe existir una comunicación y coordinación adecuada entre los programadores y otros miembros del equipo. | |
| Analista | Es el encargado de escribir las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio, todo esto lo realiza junto con el cliente. | Daimi Mederos Llanes |
| Diseñador | Encargado del diseño del sistema; así como de los prototipos de interfaces, es el máximo responsable de la realización del diseño de las metáforas y supervisa el proceso de construcción. | Daimi Mederos Llanes |
| Encargado de Pruebas | Es el encargado de ayudar al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas. | Daimi Mederos Llanes |

| | | |
|------------|--|----------------------|
| Arquitecto | Se vincula directamente con el analista y el diseñador debido a que su trabajo tiene que ver con la estructura y el diseño en grande del sistema. Ayuda en el diseño de las metáforas. | Daimi Mederos Llanes |
|------------|--|----------------------|

2.3. Modelo de Historias de Usuario del Negocio

Para todo sistema se debe establecer una técnica para la especificación de los requisitos más importantes que van a dar soporte al negocio. Dentro de los artefactos que genera la metodología SXP se encuentra la plantilla del Modelo Historias de Usuario del Negocio. En esta plantilla se modelan las características específicas del proceso de colaboración en línea que constituye el negocio en cuestión. De esta manera se establece una panorámica general de este proceso, logrando un mayor entendimiento del mismo por parte del equipo de proyecto. En la

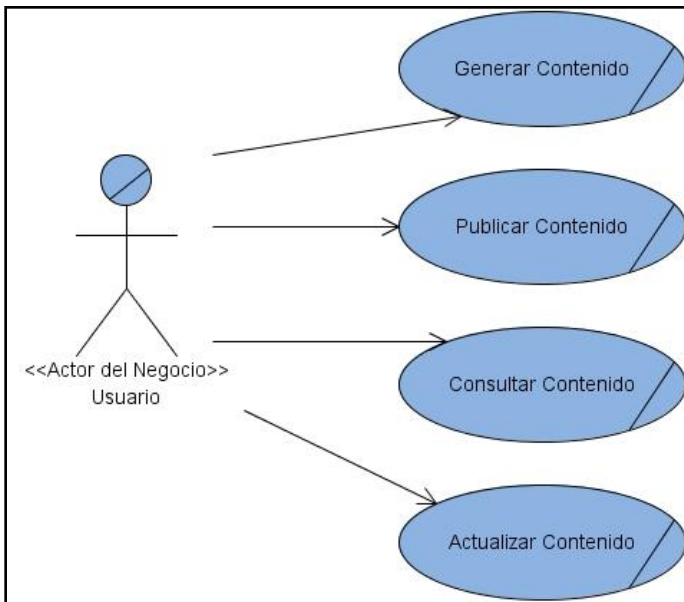


Fig. 2 se muestra el modelo que define las características específicas del negocio en cuestión.

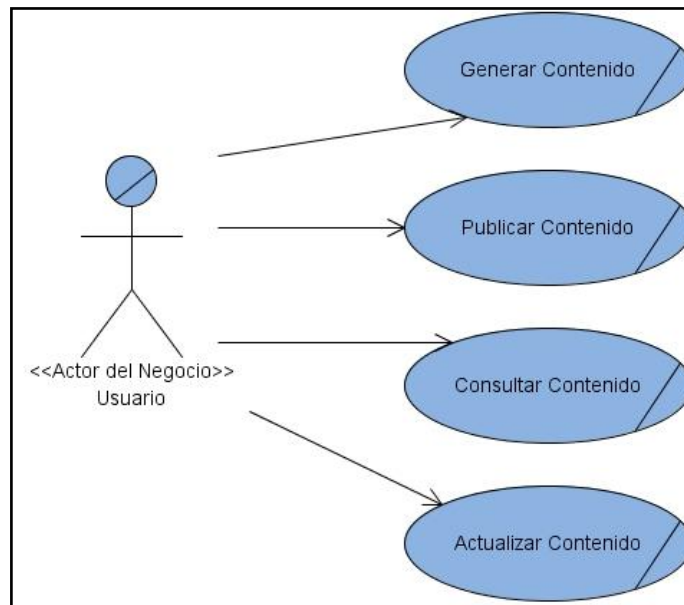


Fig. 2 Modelo de Historias de Usuario del Negocio

El Usuario es quien inicia y para el cual está dirigido el proceso de colaboración en línea. Este inicializa las Historias de Usuario del Negocio: Generar Contenido, Publicar Contenido, Consultar Contenido y Actualizar Contenido que constituyen los pilares fundamentales del proceso de colaboración en línea.

2.4. Lista de Reserva del Producto (LRP)

Recoge en una lista priorizada todo el trabajo a desarrollar en el proyecto. Su objetivo es asegurar que el producto definido al terminar la lista es el más correcto, útil y competitivo posible. Para esto la lista debe acompañar los cambios en el entorno y el producto. Puede estar conformada por requerimientos técnicos y del negocio, funciones, errores a reparar, defectos, mejoras y actualizaciones tecnológicas requeridas. A continuación se presenta en la Tabla 2 la lista priorizada contenida en el LRP de la solución propuesta:

Tabla 2: Lista de Reserva del Producto

| Prioridad | Ítem * | Descripción | Estimación | Estimado por |
|-----------------|--------|----------------------------|------------|--------------|
| Muy Alta | | | | |
| | 1 | Crear lienzo para dibujar. | 5 días | Analista |
| | 2 | Seleccionar color. | 5 días | Analista |
| | 3 | Dibujar a mano alzada. | 20 días | Analista |

| | | | | |
|--|----|--|---------|----------|
| | 4 | Crear lienzo para insertar imágenes. | 5 días | Analista |
| | 5 | Seleccionar imagen. | 5 días | Analista |
| | 6 | Insertar imagen. | 20 días | Analista |
| | 7 | Limpiar los lienzos. | 5 días | Analista |
| Alta | | | | |
| | | | | |
| Media | | | | |
| | | | | |
| Baja | | | | |
| | | | | |
| RNF (Requisitos No Funcionales) | | | | |
| | 8 | La aplicación está dirigida a usuarios con conocimientos básicos de informática y un correcto dominio del negocio en cuestión, por lo cual debe estar diseñada para ser de fácil manejo por el cliente. | | |
| | 9 | El Sistema es una aplicación web. | | |
| | 10 | Se debe realizar la aplicación de forma versionable que permita darle mantenimiento al sistema a fin de aumentar las funcionalidades y/o corregir los errores del mismo a través de versiones posteriores. | | |
| | 11 | Se documentará la aplicación con los manuales de usuario, instalación y desarrollador con el objetivo de explicar el uso de la aplicación y garantizar el soporte de la misma. | | |
| | 12 | En el lado del servidor el lenguaje de programación a utilizar es Java, empleando el marco de trabajo jWebSocket y el IDE NetBeans 7.0.1. | | |
| | 13 | En el lado del cliente se hará uso de los lenguajes | | |

| | | | | |
|--|----|--|--|--|
| | | JavaScript, HTML y CSS, empleando el marco de trabajo jQuery y el IDE NetBeans 7.0.1. | | |
| | 14 | Para la modelación se utilizará el lenguaje UML y la herramienta Visual Paradigm 6.4. | | |
| | 15 | Todos los textos y mensajes en pantalla aparecerán en idioma inglés. | | |
| | 16 | La interfaz deberá ser consistente con el mundo real, de manera que los conceptos manejados sean conocidos por los usuarios, para que les sea fácil su uso y aprendizaje. | | |
| | 17 | Lograr un diseño amigable y de fácil uso. | | |
| | 18 | Todas las interfaces de usuario que se definan para el sistema respetarán los patrones de diseño establecidos para el proyecto. | | |
| | 19 | Los requisitos mínimos de hardware que tiene que tener el cliente para el correcto funcionamiento de la aplicación son: 512 MB de memoria RAM, microprocesador Pentium IV, tarjeta red 100 Mbps. | | |
| | 20 | El cliente debe contar con un navegador que soporte el protocolo WebSocket. | | |
| | 21 | El servidor de aplicaciones debe tener instalado la máquina virtual de Java OpenJDK 7 y el servidor de jWebSocket. | | |
| | 22 | El código de la aplicación será liberado bajo la Licencia | | |

| | | | | |
|--|--|---|--|--|
| | | Pública General Reducida de GNU (LGPL). | | |
|--|--|---|--|--|

2.5. Historias de Usuario y Tareas de Ingeniería

Las historias de usuario, en la metodología de desarrollo SXP, son las que describen las tareas que el sistema debe hacer, cuestión que depende en gran medida de las especificaciones realizadas por el cliente. Se escriben con un lenguaje natural y con palabras concisas. Constituyen la guía para la construcción posterior de las pruebas de aceptación comprobando de esta manera su correcta implementación. Para lograr el desarrollo de una historia de usuario esta es desglosada en varias tareas de ingeniería que describen con más detalle la forma en que se implementan cada una de las funcionalidades contenidas en la historia de usuario. A continuación se presentan las historias de usuario con sus correspondientes tareas de ingeniería:

Tabla 3: Descripción de la Historia de Usuario Dibujar

| Historia de Usuario | |
|--|--|
| Número: HU_1 | Nombre Historia de Usuario: Dibujar |
| Modificación de Historia de Usuario Número: Ninguna | |
| Usuario: Daimi Mederos Llanes | Iteración Asignada: 2 |
| Prioridad en Negocio: Muy Alta | Puntos Estimados: 6 semanas |
| Riesgo en Desarrollo: Alto | Puntos Reales: 6 semanas |
| Descripción: La presente historia de usuario tiene como objetivo crear un lienzo compartido sobre el cual el usuario pueda seleccionar un color y dibujar a mano alzada. | |
| Observaciones: Para que esto sea posible el usuario debe haber accedido a la aplicación utilizando un navegador que soporte el protocolo de comunicación WebSocket y el servidor de la aplicación tiene que estar ejecutándose. | |
| Prototipo de interfaz: Ver Anexo # 1: Dibujar. | |

Tabla 4: Descripción de la Tarea de Ingeniería 1.1

| Tarea de Ingeniería | |
|---|---|
| Número Tarea: 1.1 | Número Historia de Usuario: HU_1 |
| Nombre Tarea: Crear lienzo para dibujar | |
| Tipo de Tarea : Desarrollo | Puntos Estimados: 5 días |
| Fecha Inicio: 28/10/2011 | Fecha Fin: 3/11/2011 |
| Programador Responsable: Daimi Mederos Llanes | |
| Descripción: Utilizando el Canvas de HTML5 se crea un Canvas o lienzo en el cual se podrá dibujar a mano alzada. | |

Tabla 5: Descripción de la Tarea de Ingeniería 1.2

| Tarea de Ingeniería | |
|--|---|
| Número Tarea: 1.2 | Número Historia de Usuario: HU_1 |
| Nombre Tarea: Seleccionar color | |
| Tipo de Tarea : Desarrollo | Puntos Estimados: 5 días |
| Fecha Inicio: 4/11/2011 | Fecha Fin: 10/11/2011 |
| Programador Responsable: Daimi Mederos Llanes | |
| Descripción: Desarrollar la funcionalidad que permite seleccionar el color con el que se pintan en el lienzo. | |

Tabla 6: Descripción de la Tarea de Ingeniería 1.3

| Tarea de Ingeniería | |
|--|---|
| Número Tarea: 1.3 | Número Historia de Usuario: HU_1 |
| Nombre Tarea: Dibujar a mano alzada | |
| Tipo de Tarea : Desarrollo | Puntos Estimados: 20 días |
| Fecha Inicio: 11/11/2011 | Fecha Fin: 8/12/2011 |
| Programador Responsable: Daimi Mederos Llanes | |
| Descripción: Desarrollar el método doPaint que hará uso de los eventos mousedown, mousemove y mouseup para dibujar a mano alzada sobre el lienzo. | |

Tabla 7: Descripción de la Historia de Usuario Insertar Imágenes

| Historia de Usuario | |
|--|--|
| Número: HU_2 | Nombre Historia de Usuario: Insertar Imágenes |
| Modificación de Historia de Usuario Número: Ninguna | |
| Usuario: Daimi Mederos Llanes | Iteración Asignada: 2 |
| Prioridad en Negocio: Muy Alta | Puntos Estimados: 6 semanas |
| Riesgo en Desarrollo: Alto | Puntos Reales: 6 semanas |
| Descripción: La presente historia de usuario tiene como objetivo crear un lienzo compartido sobre el cual el usuario pueda seleccionar una imagen e insertarla en el lienzo. | |
| Observaciones: Para que esto sea posible el usuario debe haber accedido a la aplicación utilizando un navegador que soporte el protocolo de comunicación WebSocket y el servidor de la aplicación tiene que estar ejecutándose. | |
| Prototipo de interfaz: Ver Anexo # 2: Insertar Imágenes. | |

Tabla 8: Descripción de la Tarea de Ingeniería 2.1

| Tarea de Ingeniería | |
|--|---|
| Número Tarea: 2.1 | Número Historia de Usuario: HU_2 |
| Nombre Tarea: Crear lienzo para insertar imágenes | |
| Tipo de Tarea : Desarrollo | Puntos Estimados: 5 días |
| Fecha Inicio: 9/12/2011 | Fecha Fin: 15/12/2011 |
| Programador Responsable: Daimi Mederos Llanes | |
| Descripción: Utilizando el Canvas de HTML5 se crea un Canvas o lienzo en el cual se podrán insertar imágenes. | |

Tabla 9: Descripción de la Tarea de Ingeniería 2.2

| Tarea de Ingeniería | |
|---|---|
| Número Tarea: 2.2 | Número Historia de Usuario: HU_2 |
| Nombre Tarea: Seleccionar imagen | |

| | |
|---|---------------------------------|
| Tipo de Tarea : Desarrollo | Puntos Estimados: 5 días |
| Fecha Inicio: 16/12/2011 | Fecha Fin: 11/01/2012 |
| Programador Responsable: Daimi Mederos Llanes | |
| Descripción: Desarrollar la funcionalidad que permite seleccionar una imagen para luego insertarla en el lienzo. | |

Tabla 10: Descripción de la Tarea de Ingeniería 2.3

| Tarea de Ingeniería | |
|--|---|
| Número Tarea: 2.3 | Número Historia de Usuario: HU_2 |
| Nombre Tarea: Insertar imagen | |
| Tipo de Tarea : Desarrollo | Puntos Estimados: 20 días |
| Fecha Inicio: 12/01/2012 | Fecha Fin: 8/02/2012 |
| Programador Responsable: Daimi Mederos Llanes | |
| Descripción: Implementar el método drawImage para poder insertar la imagen seleccionada, en la posición y con el tamaño especificado. | |

Tabla 11: Descripción de la Historia de Usuario Limpiar Lienzos

| Historia de Usuario | |
|--|--|
| Número: HU_3 | Nombre Historia de Usuario: Limpiar Lienzos |
| Modificación de Historia de Usuario Número: Ninguna | |
| Usuario: Daimi Mederos Llanes | Iteración Asignada: 2 |
| Prioridad en Negocio: Muy Alta | Puntos Estimados: 1 semana |
| Riesgo en Desarrollo: Alto | Puntos Reales: 1 semana |
| Descripción: La presente historia de usuario tiene como objetivo permitir que el usuario pueda limpiar totalmente el lienzo de dibujar y el de insertar imágenes. | |
| Observaciones: Para que esto sea posible el usuario debe haber accedido a la aplicación utilizando un navegador que soporte el protocolo de | |

comunicación WebSocket y el servidor de la aplicación tiene que estar ejecutándose.

Prototipo de interfaz: [Ver Anexo # 3: Limpiar Lienzos.](#)

Tabla 12: Descripción de la Tarea de Ingeniería 3.1

| Tarea de Ingeniería | |
|---|---|
| Número Tarea: 3.1 | Número Historia de Usuario: HU_3 |
| Nombre Tarea: Limpiar lienzo de dibujar a mano alzada | |
| Tipo de Tarea : Desarrollo | Puntos Estimados: 3 días |
| Fecha Inicio: 9/02/2012 | Fecha Fin: 15/02/2012 |
| Programador Responsable: Daimi Mederos Llanes | |
| Descripción: Desarrollar el método que permitirán limpiar totalmente el lienzo de dibujar a mano alzada. | |

Tabla 13: Descripción de la Tarea de Ingeniería 3.2

| Tarea de Ingeniería | |
|---|---|
| Número Tarea: 3.2 | Número Historia de Usuario: HU_3 |
| Nombre Tarea: Limpiar lienzo de insertar imágenes | |
| Tipo de Tarea : Desarrollo | Puntos Estimados: 2 días |
| Fecha Inicio: 9/02/2012 | Fecha Fin: 15/02/2012 |
| Programador Responsable: Daimi Mederos Llanes | |
| Descripción: Desarrollar el método que permitirán limpiar totalmente el lienzo de insertar imágenes. | |

2.6. Plan de Releases

El plan de releases o iteraciones permite realizar las entregas intermedias y la entrega final. Tiene como entrada la relación de historias de usuario definidas previamente. Para colocar una historia de usuario en cada iteración se tiene en cuenta la prioridad que definió el cliente para dicha historia. Como resultado de la

priorización de las historias de usuario durante la presente investigación se llegó a la planificación que se muestra en la Tabla 14:

Tabla 14: Plan de Releases

| Release | Descripción de la iteración | Orden de la HU a implementar | Duración total |
|-------------|---|------------------------------|----------------|
| Iteración 2 | En esta iteración se va a desarrollar una versión completa de la aplicación para la cual se desarrollarán todas las historias de usuario ya que las mismas tienen prioridad muy alta. | HU_1 HU_2 HU_3 HU_4 | 13 semanas |

2.7. Descripción de la Arquitectura

La arquitectura de una aplicación es la vista conceptual de su estructura. Toda aplicación contiene código de presentación, procesamiento de datos y almacenamiento de datos. La arquitectura de las aplicaciones difiere según como este distribuido su código. Para desarrollar la aplicación Shared Canvas Demo se utilizará una arquitectura en capas que tiene como objetivo primordial la separación de la lógica del negocio de la lógica del diseño. La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y en caso de que sobrevenga algún cambio, sólo se afecta al nivel requerido sin tener que revisar entre código mezclado. Además permite que cada grupo de trabajo esté totalmente abstraído del resto de los niveles. Como se evidencia en la

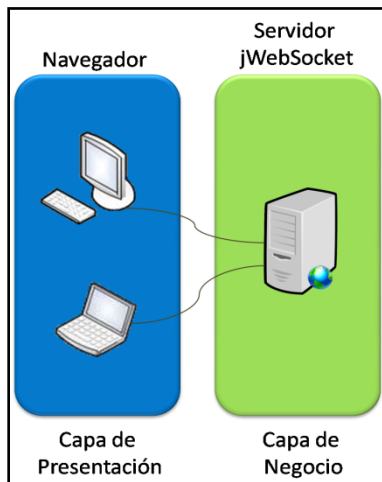


Fig. 3, específicamente se cuenta con 2 capas, la capa de presentación y la capa de negocio.

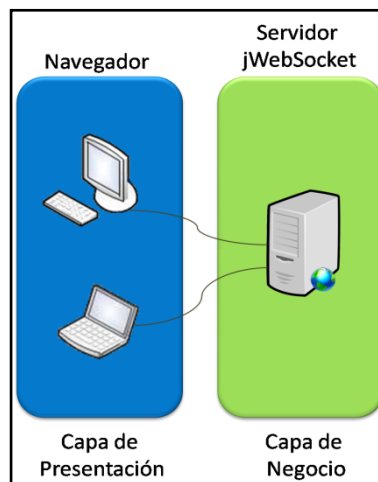


Fig. 3 Arquitectura de la solución

Capa de presentación: Está representada por el navegador. Muestra el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso. También es conocida como interfaz gráfica y debe tener la característica de ser amigable, entendible y fácil de usar para el usuario. Esta capa se comunica únicamente con la capa de negocio.

Capa de negocio: Está representada por el servidor de jWebSocket. Es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio porque es

aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados.

2.8. Diseño con Metáforas

La metodología de desarrollo SXP seleccionada para la presente investigación está basada en XP que utiliza el diseño con metáforas para describir como debe funcionar la aplicación. Además genera el artefacto conocido como Modelo de Diseño que a su vez está compuesto por un diagrama de paquetes. Este diagrama expone el diseño de la aplicación y muestra como está dividida en agrupaciones lógicas. El diagrama de paquetes correspondiente a la aplicación web demostrativa de colaboración en línea Shared Canvas Demo puede observarse en la siguiente

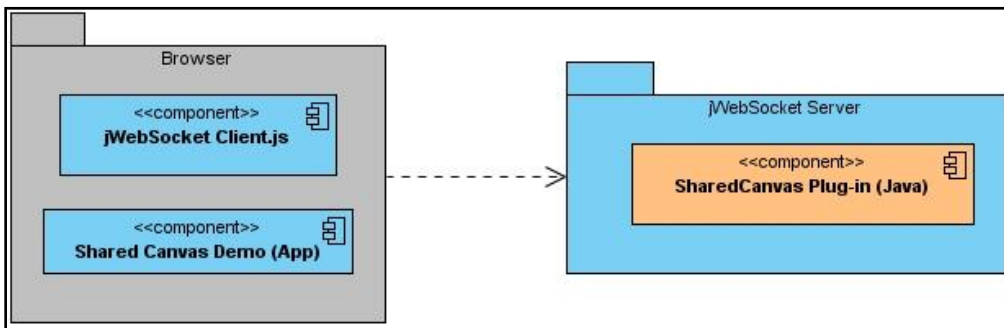


Fig. 4:

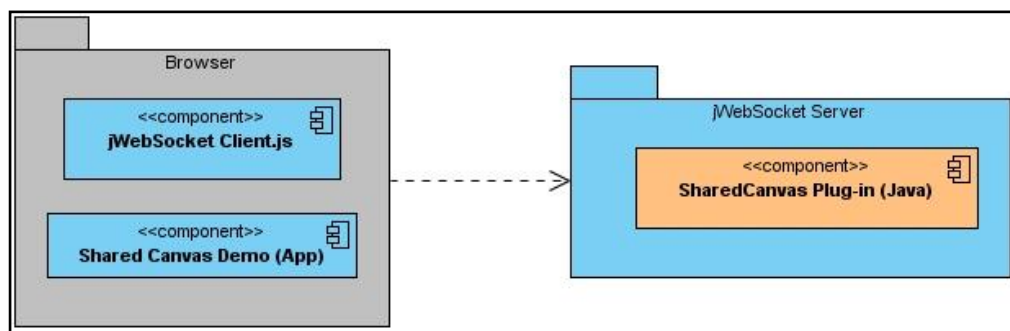


Fig. 4 Diagrama de Paquetes

El browser representa la capa de presentación. Este contiene los componentes Shared Canvas Demo que constituye la aplicación y jWebSocket Client que es el cliente del marco de trabajo jWebSocket. Esta capa del browser se relaciona con la

capa del servidor de jWebSocket. Además el servidor de jWebSocket representa la capa de negocio que contiene el *plugin*²⁶ de la aplicación.

2.9. Diagrama de Componentes

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas. Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, entre otros. Seguidamente en la

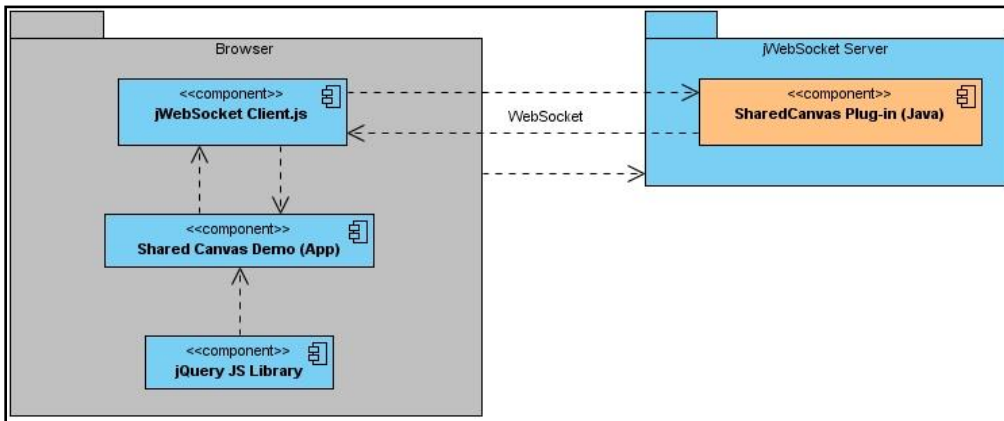


Fig. 5 se presenta el diagrama de componentes correspondiente a la aplicación web demostrativa de colaboración en línea Shared Canvas Demo:

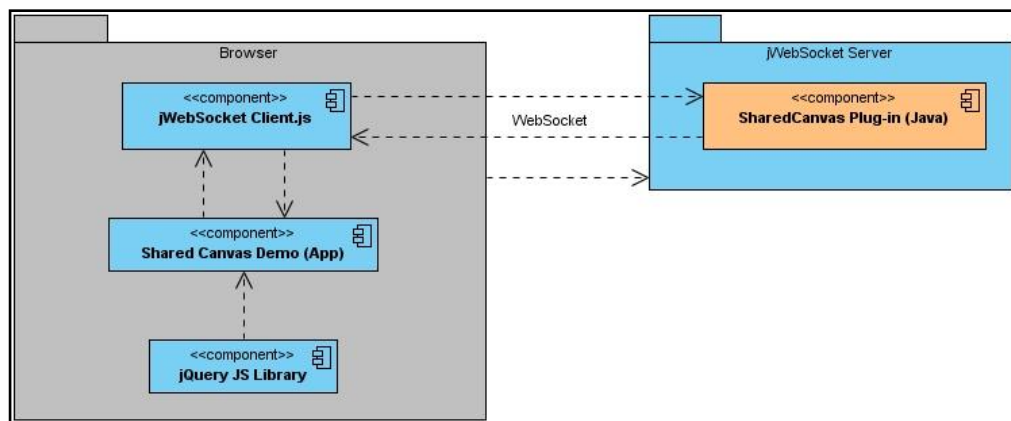


Fig. 5 Diagrama de Componentes

²⁶Plugin: Es un módulo de hardware o software que añade una característica o un servicio específico a un sistema más grande

La aplicación Shared Canvas Demo consume funcionalidades de la librería de JavaScript jQuery. Además envía y recibe los datos a través del cliente de jWebSocket que se comunica con su servidor por medio del protocolo WebSocket. El servidor de jWebSocket contiene el *plugin* de la aplicación que es el encargado de distribuir las acciones que se realicen sobre los lienzos compartidos de Shared Canvas Demo entre todos los usuarios conectados a la aplicación.

Conclusiones del Capítulo

En este capítulo se definieron las características y funcionalidades del sistema que se necesita desarrollar. Quedaron aprobados los requisitos funcionales y no funcionales necesarios para obtener un sistema eficiente. Se describieron las historias de usuario y tareas de ingeniería que se deben implementar y el plan de releases para establecer el cronograma de trabajo. Además se describió la arquitectura que se va a utilizar para el desarrollo y se realizó el modelado del diagrama de paquetes y de componentes que permiten una mejor comprensión de la solución propuesta.

CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DEL SISTEMA

Introducción

En el presente capítulo se documenta la implementación de la solución propuesta, generándose el diagrama de despliegue del sistema desarrollado. Se presentan las pruebas funcionales a las que fue sometida la aplicación, demostrando de esta forma el cumplimiento de los requerimientos definidos. Además se muestra la capacidad funcional de la aplicación mediante la certificación de calidad de software y la valoración del cliente. Se exponen los resultados obtenidos hasta el momento y el aporte social y económico de la solución.

3.1. Diagrama de Despliegue

El diagrama de despliegue es un artefacto que modela la arquitectura en tiempo de ejecución de un sistema. En él se indica la situación física de los componentes lógicos desarrollados, lo que significa que sitúa el software en el hardware que lo contiene. Representa la distribución física de la aplicación en términos de cómo se distribuirán las funcionalidades entre los nodos, donde cada nodo representa un recurso de cómputo, siendo estos procesadores o dispositivos hardware que se necesitan para el despliegue de la aplicación. A continuación se muestra en la



el diagrama de despliegue correspondiente a la aplicación web demostrativa desarrollada durante la presente investigación:

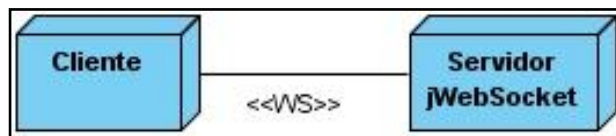


Fig. 6 Diagrama de Despliegue

El cliente se conectará por medio del protocolo WebSocket al Servidor de jWebSocket para lo cual solo necesitará un navegador que soporte WebSocket sobre cualquier sistema operativo.

3.2. Validación de la Solución Propuesta

En este epígrafe se presenta la estrategia de validación diseñada para la presente investigación. Partiendo del objetivo trazado la validación está enfocada a demostrar la capacidad de la aplicación web demostrativa desarrollada para mejorar el nivel de usabilidad del marco de trabajo jWebSocket en el desarrollo de aplicaciones de colaboración en línea en tiempo real. La estrategia está conformada por tres etapas las cuales van aumentando los niveles de validación partiendo desde el equipo de desarrollo y llegando hasta la valoración del cliente.

La primera etapa de la estrategia es la realización de pruebas funcionales a las historias de usuario por parte del equipo del proyecto. Estas pruebas se realizan con el objetivo de verificar el correcto funcionamiento de los requerimientos con que debe cumplir la aplicación desarrollada.

Seguidamente se pasa a un proceso de certificación de calidad de software por parte del Grupo de Calidad del Centro de Desarrollo de la Facultad Regional "Mártires de Artemisa". Este grupo hace una certificación basándose principalmente en la funcionalidad, estandarización, organización y limpieza del código fuente de la aplicación. Además revisa los artefactos documentales que son generados por la metodología de desarrollo SXP seleccionada para la presente investigación.

Por último se realiza una valoración de la aplicación por parte del cliente Alexander Schulze. Esta etapa de validación tiene el objetivo de obtener una certificación de calidad de la aplicación desarrollada para que pueda ser integrada al marco de trabajo jWebSocket. Seguidamente se describe con más detalle de cada una de las etapas de la estrategia de validación trazada a las cuales fue sometida la aplicación desarrollada.

Casos de Pruebas Funcionales

La realización de los casos de pruebas funcionales proporciona grandes ventajas, permitiendo a los programadores medir la calidad de su trabajo y garantizar la entrega de un producto con calidad y en correspondencia con las necesidades del

cliente. Se definieron varios casos de pruebas funcionales para la aplicación Shared Canvas Demo basados en las historias de usuario que fueron diseñadas para la aplicación. De esta manera se verifica el cumplimiento de los requisitos funcionales definidos para la misma. A continuación se explican detalladamente cada uno de los casos de pruebas funcionales realizados:

Tabla 15: Descripción del Caso de Prueba # 1 para la HU Dibujar

| Caso de Prueba Funcional | |
|---|--|
| Código Caso de Prueba: jws-1-1 | Nombre Historia de Usuario: Dibujar |
| Nombre de la persona que realiza la prueba: Daimi Mederos Llanes | |
| Descripción de la Prueba: Esta prueba consiste en verificar que el usuario pueda seleccionar el color con el cual va a dibujar sobre el lienzo. | |
| Condiciones de Ejecución: Para poder realizar la prueba se tiene que utilizar un navegador que soporte el protocolo WebSocket y el servidor de la aplicación tiene que estar ejecutándose. | |
| Entrada / Pasos de ejecución: Para seleccionar el color se debe hacer clic sobre el color deseado por el usuario. | |
| Resultado Esperado: El usuario puede seleccionar el color con el cual dibujará sobre el lienzo. | |
| Evaluación de la Prueba: Satisfactoria. | |

Tabla 16: Descripción del Caso de Prueba # 2 para la HU Dibujar

| Caso de Prueba Funcional | |
|--|--|
| Código Caso de Prueba: jws-1-2 | Nombre Historia de Usuario: Dibujar |
| Nombre de la persona que realiza la prueba: Daimi Mederos Llanes | |
| Descripción de la Prueba: Esta prueba consiste en verificar que el usuario pueda dibujar a mano alzada sobre el lienzo. | |

| |
|---|
| Condiciones de Ejecución: Para poder realizar la prueba se tiene que utilizar un navegador que soporte el protocolo WebSocket y el servidor de la aplicación tiene que estar ejecutándose. |
| Entrada / Pasos de ejecución: Para pintar a mano alzada debe dirigirse al lienzo, presionar el clic izquierdo del mouse y mover el mismo sobre el lienzo. |
| Resultado Esperado: El usuario puede pintar a mano alzada sobre el lienzo. |
| Evaluación de la Prueba: Satisfactoria. |

Tabla 17: Descripción del Caso de Prueba # 3 para la HU Dibujar

| Caso de Prueba Funcional | |
|---|--|
| Código Caso de Prueba: jws-1-3 | Nombre Historia de Usuario: Dibujar |
| Nombre de la persona que realiza la prueba: Daimi Mederos Llanes | |
| Descripción de la Prueba: Esta prueba consiste en verificar que el usuario pueda dirigirse hacia la pestaña Paint para dibujar a mano alzada sobre el lienzo. | |
| Condiciones de Ejecución: Para poder realizar la prueba se tiene que utilizar un navegador que soporte el protocolo WebSocket y el servidor de la aplicación tiene que estar ejecutándose. | |
| Entrada / Pasos de ejecución: Para dirigirse hacia la pestaña Paint el usuario debe hacer clic sobre el botón correspondiente a la pestaña "Paint". | |
| Resultado Esperado: El usuario puede dirigirse hacia la pestaña Paint para dibujar a mano alzada sobre el lienzo. | |
| Evaluación de la Prueba: Satisfactoria. | |

Tabla 18: Descripción del Caso de Prueba # 1 para la HU Insertar Imágenes

| Caso de Prueba Funcional | |
|---|--|
| Código Caso de Prueba: jws-2-1 | Nombre Historia de Usuario: Insertar Imágenes |
| Nombre de la persona que realiza la prueba: Daimi Mederos Llanes | |

| |
|---|
| Descripción de la Prueba: Esta prueba consiste en verificar que el usuario pueda seleccionar las imágenes que va a insertar sobre el lienzo. |
| Condiciones de Ejecución: Para poder realizar la prueba se tiene que utilizar un navegador que soporte el protocolo WebSocket y el servidor de la aplicación tiene que estar ejecutándose. |
| Entrada / Pasos de ejecución: Para seleccionar una imagen se debe hacer clic sobre la imagen deseada por el usuario. |
| Resultado Esperado: El usuario puede seleccionar imágenes para insertar sobre el lienzo. |
| Evaluación de la Prueba: Satisfactoria. |

Tabla 19: Descripción del Caso de Prueba # 2 para la HU Insertar Imágenes

| Caso de Prueba Funcional | |
|---|--|
| Código Caso de Prueba: jws-2-2 | Nombre Historia de Usuario: Insertar Imágenes |
| Nombre de la persona que realiza la prueba: Daimi Mederos Llanes | |
| Descripción de la Prueba: Esta prueba consiste en verificar que el usuario pueda insertar imágenes sobre el lienzo. | |
| Condiciones de Ejecución: Para poder realizar la prueba se tiene que utilizar un navegador que soporte el protocolo WebSocket y el servidor de la aplicación tiene que estar ejecutándose. | |
| Entrada / Pasos de ejecución: Para insertar una imagen luego de haberla seleccionado debe dirigirse al lienzo, presionar el clic izquierdo del mouse, arrastrarlo hasta donde se desee y soltarlo. | |
| Resultado Esperado: El usuario puede insertar imágenes sobre el lienzo. | |
| Evaluación de la Prueba: Satisfactoria. | |

Tabla 20: Descripción del Caso de Prueba # 3 para la HU Insertar Imágenes

| Caso de Prueba Funcional | |
|---|--|
| Código Caso de Prueba: jws-2-3 | Nombre Historia de Usuario: Insertar Imágenes |
| Nombre de la persona que realiza la prueba: Daimi Mederos Llanes | |
| Descripción de la Prueba: Esta prueba consiste en verificar que el usuario pueda dirigirse hacia la pestaña Insert Image para insertar imágenes sobre el lienzo. | |
| Condiciones de Ejecución: Para poder realizar la prueba se tiene que utilizar un navegador que soporte el protocolo WebSocket y el servidor de la aplicación tiene que estar ejecutándose. | |
| Entrada / Pasos de ejecución: Para dirigirse hacia la pestaña Insert Image el usuario debe hacer clic sobre el botón correspondiente a la pestaña “Insert Image”. | |
| Resultado Esperado: El usuario puede dirigirse hacia la pestaña Insert Image para insertar imágenes sobre el lienzo. | |
| Evaluación de la Prueba: Satisfactoria. | |

Tabla 21: Descripción del Caso de Prueba # 1 para la HU Limpiar Lienzos

| Caso de Prueba Funcional | |
|---|--|
| Código Caso de Prueba: jws-3-1 | Nombre Historia de Usuario: Limpiar Lienzos |
| Nombre de la persona que realiza la prueba: Daimi Mederos Llanes | |
| Descripción de la Prueba: Esta prueba consiste en verificar que el usuario pueda limpiar por completo el lienzo de dibujar a mano alzada. | |
| Condiciones de Ejecución: Para poder realizar la prueba se tiene que utilizar un navegador que soporte el protocolo WebSocket y el servidor de la aplicación tiene que estar ejecutándose. | |
| Entrada / Pasos de ejecución: Para limpiar totalmente el lienzo de dibujar a | |

| |
|--|
| mano alzada se debe hacer clic en el botón “Clear” correspondiente a la pestaña Paint. |
| Resultado Esperado: El usuario puede limpiar por completo el lienzo de dibujar a mano alzada. |
| Evaluación de la Prueba: Satisfactoria. |

Tabla 22: Descripción del Caso de Prueba # 2 para la HU Limpiar Lienzos

| Caso de Prueba Funcional | |
|---|--|
| Código Caso de Prueba: jws-3-2 | Nombre Historia de Usuario: Limpiar Lienzos |
| Nombre de la persona que realiza la prueba: Daimi Mederos Llanes | |
| Descripción de la Prueba: Esta prueba consiste en verificar que el usuario pueda limpiar por completo el lienzo de insertar imágenes. | |
| Condiciones de Ejecución: Para poder realizar la prueba se tiene que utilizar un navegador que soporte el protocolo WebSocket y el servidor de la aplicación tiene que estar ejecutándose. | |
| Entrada / Pasos de ejecución: Para limpiar totalmente el lienzo de insertar imágenes se debe hacer clic en el botón “Clear” correspondiente a la pestaña Insert Image. | |
| Resultado Esperado: El usuario puede limpiar por completo el lienzo de insertar imágenes. | |
| Evaluación de la Prueba: Satisfactoria. | |

Los casos de pruebas funcionales realizados por el equipo de desarrollo constituyen un proceso importante para garantizar que la aplicación cumple con todo los requisitos funcionales para los que fue diseñada. Sin embargo es importante obtener la valoración de personal externo al proyecto, el cual no haya estado vinculado directamente al desarrollo y pueda validar el uso y funcionamiento de la aplicación basándose en la documentación y el código fuente de la misma.

Certificación de Calidad de Software

Se realizó un estudio y revisión por parte del Grupo de Calidad del Centro de Desarrollo de la Facultad Regional "Mártires de Artemisa" de la documentación generada en el desarrollo del proyecto. Esta documentación está conformada por varios artefactos que se generan durante las fases establecidas por la metodología de desarrollo SXP seleccionada para la presente investigación. Para realizar esta valoración el grupo tiene en cuenta que la documentación sea clara, concisa, que tenga buena redacción, que refleje los resultados y las decisiones tomadas durante el desarrollo. Entre los principales artefactos evaluados se encuentran los siguientes:

- ✓ Lista de Reserva del Producto (LRP), recoge en una lista priorizada todos los requisitos funcionales y no funcionales de la aplicación.
- ✓ Historias de Usuario, aporta al grupo los elementos necesarios para conocer las tareas que el sistema debe hacer, cuestión que depende en gran medida de las especificaciones realizadas por el cliente.
- ✓ Tareas de Ingeniería, describe con más detalle la forma en que se implementan cada una de las funcionalidades contenidas en las Historias de Usuario.
- ✓ Arquitectura de Software, describe la arquitectura de la aplicación, además ofrece información sobre las herramientas y tecnologías fundamentales utilizadas para el desarrollo.
- ✓ Plan de Releases, permite establecer el cronograma de trabajo a partir de la priorización de las Historias de Usuario.
- ✓ Además se pueden mencionar los Manuales de Usuario, Instalación y Desarrollador. Estos documentos explican el funcionamiento, proceso de instalación y utilización del código fuente de la aplicación respectivamente.

Luego de haber logrado el entendimiento de las pautas tomadas en cuenta para el desarrollo de la aplicación web demostrativa, el grupo inicia la valoración del código

fuelle generado durante la etapa de implementación. Para esta revisión se tiene en cuenta la organización, entendimiento, comentarios y estandarización del código a partir del artefacto Estándar de Código definido para el desarrollo de la aplicación por parte del equipo del proyecto. La revisión de los artefactos estuvo a cargo de la Ing. Maidel Ojeda Castro Asesora de Calidad del Centro de Desarrollo y el código fuente fue revisado por el Ing. Domma Moreno Dager Asesor de Tecnología del Centro de Desarrollo.

Valoración del Cliente

La aplicación desarrollada durante la presente investigación fue sometida al criterio del cliente Alexander Schulze, líder de la comunidad internacional, fundador y principal arquitecto del proyecto jWebSocket.

Durante esta etapa de validación se almacenó en el repositorio internacional del proyecto jWebSocket el código fuente y una documentación en inglés de la aplicación Shared Canvas Demo para su posterior evaluación por parte del cliente. El código fuente se analiza con el objetivo de verificar su funcionalidad, limpieza y cumplimiento con los estándares de codificación establecidos para el marco de trabajo jWebSocket. De esta manera se garantiza la calidad de la solución así como su correcto funcionamiento en correspondencia con jWebSocket. A su vez se evalúa la usabilidad que presenta la aplicación desarrollada para la colaboración en línea en tiempo real.

La documentación en idioma inglés relacionada a la aplicación posibilita un mayor entendimiento de la solución. La misma está conformada por los manuales de desarrollador, administración y usuario, estos son evaluados con el objetivo de garantizar su calidad para facilitar el soporte de la aplicación una vez que esta sea liberada e integrada al entorno de desarrollo de jWebSocket.

3.3. Resultados Obtenidos

Debido al resultado satisfactorio de los casos de pruebas funcionales se demuestra el cumplimiento de los requerimientos identificados. Garantizando que la aplicación cumple con las funcionalidades para las que fue diseñada.

Por parte del Grupo de Certificación de Calidad se obtuvo resultados positivos referentes a la limpieza, estandarización del código fuente y los artefactos generados que permiten la disponibilidad de la aplicación para futuras mejoras (Ver Anexo # 4: Certificación de Calidad de Software). Con estos resultados se garantiza que la aplicación y su documentación poseen la calidad requerida.

A través de la valoración del cliente se obtuvo una certificación satisfactoria de la aplicación desarrollada (Ver Anexo # 5: Valoración del Cliente), comprobándose que la aplicación puede ser integrada al entorno de desarrollo de `WebSocket` a nivel internacional y utilizada por la Comunidad del proyecto.

Los resultados obtenidos a partir de cada estrategia de validación garantizan que los usuarios identifiquen las fortalezas del marco de trabajo para desarrollar aplicaciones de colaboración en línea en tiempo real. Esto brinda elementos importantes ante la posibilidad de aumentar el nivel de usabilidad de `WebSocket` para este fin. De esta manera queda disponible `Shared Canvas Demo` en su versión 1.0, esta constituye una aplicación que cumple con todas las especificaciones para permitir la colaboración en línea en tiempo real.

3.4. Funcionalidades Obtenidas

Entre las funcionalidades que posee la aplicación `Shared Canvas Demo` en su versión 1.0 se pueden mencionar:

- ✓ Seleccionar el color con el cual se va a dibujar.
- ✓ Dibujar a mano alzada sobre el lienzo.
- ✓ Seleccionar las imágenes que serán insertadas.
- ✓ Insertar imágenes sobre el lienzo.
- ✓ Borrar los elementos dibujados o insertados sobre el lienzo.

3.5. Aporte Social y Económico

Shared Canvas Demo es una aplicación de colaboración en línea en tiempo real desarrollada con el marco de trabajo `jWebSocket`. Esta puede ser utilizada para brindar apoyo al proceso de enseñanza-aprendizaje en la Web en sus diferentes modalidades y aspectos. Además muestra la capacidad de `jWebSocket` en el desarrollo de aplicaciones y plataformas educativas en tiempo real, mejorando la calidad de los procesos de colaboración y aprendizaje en línea, aulas y cursos virtuales.

Con esta aplicación se pueden escribir fórmulas matemáticas, realizar diagramas explicativos o simplemente compartir ideas de forma gráfica en tiempo real. De esta manera disminuye la curva de aprendizaje, se facilita el intercambio de conocimientos y se mejora de forma general la calidad del proceso de enseñanza-aprendizaje en la Web. Entre otros aspectos importantes, Shared Canvas Demo puede ser visualizado desde disímiles lugares del mundo por miles de personas, permitiendo un mayor alcance y repercusión de las potencialidades que brinda la aplicación.

Cualquier institución educativa que haga uso de esta aplicación Web tendrá asociada una disminución de costos. Debido a que esta aplicación es libre y constituye una herramienta que sin dudas traerá mejoras a la educación a distancia.

Conclusiones del Capítulo

En este capítulo se realizó el diagrama de despliegue que indica la situación física de los componentes lógicos desarrollados. Se realizaron pruebas funcionales que guiaron la calidad de la aplicación desarrollada, mostrando que sus funcionalidades son operativas y producen un resultado correcto. Además se verificó la calidad de la aplicación y su documentación mediante la certificación de calidad de software y la valoración del cliente. Demostrándose así que el sistema se encuentra en óptimas condiciones para el proceso de colaboración en línea en tiempo real.

CONCLUSIONES

Con la realización de la presente investigación se dio respuesta a las preguntas científicas planteadas, obteniéndose de manera general las siguientes conclusiones:

- ✓ La realización de la fundamentación teórico-metodológica permitió conceptualizar el proceso de colaboración en línea, que brinda a los usuarios la posibilidad de facilitar el flujo de información y la realización de trabajos en conjunto en la Web.
- ✓ Actualmente las aplicaciones de colaboración en línea permiten la sincronización, edición colaborativa, uso compartido de archivos, entre otras potencialidades pero no trabajan en tiempo real.
- ✓ La aplicación web demostrativa Shared Canvas Demo desarrollada permite la colaboración en línea en tiempo real utilizando jWebSocket, y a su vez garantiza un posible aumento del nivel de usabilidad del marco de trabajo en el desarrollo de este tipo de aplicaciones.
- ✓ Se comprobó la capacidad y potencialidad de la aplicación desarrollada para la colaboración en línea en tiempo real a través de la estrategia de validación trazada para la presente investigación, garantizando que su código fuente y documentación poseen la calidad requerida, además puede ser integrada al entorno de desarrollo de jWebSocket y utilizada por su comunidad.

RECOMENDACIONES

- ✓ Desarrollar una segunda versión de la aplicación Shared Canvas Demo incluyéndole nuevas funcionalidades tales como la integración con un editor de texto que permita la edición colaborativa de documentos en tiempo real, aumentando las potencialidades del proceso de colaboración en línea utilizando esta aplicación.

BIBLIOGRAFÍA REFERENCIADA

1. *¿Qué es la Web 2.0?* **VAN DER HENST, Christian. 2005.** Maestros del Web : <http://www.maestrosdelweb.com/editorial/web2/>, 2005.
2. **10gen, Inc. 2011.** Agile and Scalable . *mongoDB*. [En línea] 2011. [Citado el: 16 de 02 de 2012.] <http://www.mongodb.org/>.
3. —. **2011.** What is MongoDB? *10gen*. [En línea] 2011. [Citado el: 16 de 02 de 2012.] <http://www.10gen.com/why-mongodb>.
4. **Apache Software Foundation. 2011.** Apache Subversion. *Apache Subversion*. [En línea] 10 de 12 de 2011. [Citado el: 10 de 12 de 2011.] <http://subversion.apache.org/>.
5. **BARANOVSKIY, Dmitry. 2008.***Raphaël—JavaScript Library*. [En línea] 2008. [Citado el: 18 de 02 de 2012.] <http://raphaeljs.com>.
6. **BAROJA, Diego. 2011.** Web 2.0 Introducción. Maestría en Tecnologías para la Gestión y Práctica Docente. *SlideShare*. [En línea] 15 de 4 de 2011. [Citado el: 2 de 12 de 2011.] <http://www.slideshare.net/diegofer369/herramientas-web-20-pdf>.
7. *Comet-HTML5-Websocket*. **EGLI, Peter R. 2011.** s.l. : Indigoo, 2011, Vols. <http://www.slideshare.net/PeterREgli/comet-websockets-html5>.
8. *Conferencia Web 2.0 "Revving the Web"*. **O'REILLY, Tim. 2004.** San Francisco, California : s.n., 2004.
9. *Creating Mobile Learning Content*. **MARCUM, Chris. 2011.** New York : s.n., 2011. mLearning DevCon.
10. **EGUÍLUZ, Javier. 2009.***Introducción a CSS*. 2009.
11. —. **2009.***Introducción a JavaScript*. 2009.
12. —. **2009.***Introducción a XHTML*. 2009.

13. **FETTE, I. y MELNIKOV, A. 2011.** The WebSocket Protocol. *Internet Engineering Task Force (IETF)*. [En línea] 12 de 2011. [Citado el: 17 de 02 de 2012.] <http://tools.ietf.org/html/rfc6455>. 2070-1721.
14. **FIGUEROA, Roberth G., SOLÍS, Camilo J. y CABRERA, Armando A. 2011.** Entorno Virtual de Aprendizaje. [En línea] 2011. [Citado el: 13 de 12 de 2011.]
<http://eva.uci.cu/mod/resource/view.php?id=9303&subdir=/Metodologias/RUP>.
15. **FLORES, Juan José, BERTOLOTTI, Carmen y GONZÁLEZ, Fernando. 2009.** La Web 2.0 y las Herramientas de Colaboración y Participación. [En línea] 17 de 09 de 2009.
16. *Framework Approach for WebSockets*. **SCHULZE, Alexander. 2011.** s.l. : Web Technologies & Internet Applications (WebTech 2011), 2011.
http://dl.globalstf.org/index.php?page=shop.product_details&flypage=flypage_images.tpl&product_id=528&category_id=42&option=com_virtuemart&Itemid=4&vmcchk=1&Itemid=4..
17. **Fundación Wikimedia, Inc. 2001.** *Wikipedia*. [En línea] 2001. [Citado el: 16 de 02 de 2012.] http://es.wikipedia.org/wiki/Wikipedia:La_enciclopedia_libre.
18. —. **2002.** *Wikcionario*. [En línea] 2002. [Citado el: 16 de 02 de 2012.] http://es.wiktionary.org/wiki/Wikcionario:Acerca_de.
19. —. **2003.** *Wikilibros*. [En línea] 2003. [Citado el: 16 de 02 de 2012.] http://es.wikibooks.org/wiki/Wikilibros:Acerca_de.
20. **GOSLING, James, y otros. 2005.** *The Java language specification*. s.l. : Addison-Wesley, 2005. 0-321-24678-0.
21. **HERNAMPÉREZ, Rafael. 2010.** Análisis entre modelo relacional y modelo documental de bases de datos. *Raticos tecnológicos*. [En línea] 28 de 03 de 2010. [Citado el: 17 de 11 de 2011.]

<http://rafinquer.blogspot.com/2010/03/analisis-entre-modelo-relacional-y.html>.

22. *Herramientas Case*. **ALFARO, Félix. 2011.** 2011, COLECCION CULTURA INFORMATICA.
23. *HTML5 Websockets and communication*. **LUBBERS, Petter. 2010.** Java User Group Meeting : <http://www.slideshare.net/Kaazing/v2-peterlubberssfjugwebsocket>, 2010.
24. *jQuery, la librería Javascript por excelencia. Un framework Javascript lleno de ventajas*. **SORIANO, Javier. 2011.** Valencia : <http://www.dicreato.com/blog/jquery-la-libreria-javascript-por-excelencia-un-framework-javascript-lleno-de-ventajas/>, 2011.
25. **LANGER, Bárbara. 2010.** Herramientas de colaboración online. *Portal Tecno-Soluciones*. [En línea] 13 de 07 de 2010. [Citado el: 15 de 02 de 2012.]
<http://www.tecnosoluciones.com/modules.php?name=News&file=article&sid=192>.
26. *Las tecnologías de información y comunicación en el seguimiento y evaluación de los desastres naturales: Estudio de un caso: La plataforma informática de la red UTEEDA para la gestión de la información sobre desastres*. **MENA, Néstor. 2007.** 1, Ciudad de La Habana : Editorial Ciencias Médicas, 2007, Vol. 16.
http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352007000700004&lng=es&nrm=iso.1024-9435.
27. *Metodologías Tradicionales Vs. Metodologías Ágiles*. **FIGUEROA, Roberth G., SOLÍS, Camilo J. y CABRERA, Armando A. 2011.** Loja : <http://www.docstoc.com/docs/102328746/articulo-metodologia-de-sw-formato>, 2011.

28. **Ministerio de Informática y Comunicaciones. 2012.** Aulas hermanas. *EcuRed*. [En línea] 06 de 02 de 2012. [Citado el: 17 de 02 de 2012.] http://www.ecured.cu/index.php/Aulas_hermanas.
29. **MURPHEY, Rebecca. 2010.** *jQuery Fundamentals*. s.l. : Autoedición, 2010.
30. **NetBeans. 2011.** *NetBeans*. [En línea] Oracle Corporation, 2011. <http://netbeans.org/features/index.html>.
31. **NOURIE, Dana. 2005.** *Oracle Sun Developer Network (SDN)*. [En línea] Oracle Corporation, 24 de Mayo de 2005. <http://java.sun.com/developer/technicalArticles/tools/intro.html>.
32. **PARKER, Kevin R. y CHAO, Joseph T. 2008.** Wiki as a Teaching Tool. *Interdisciplinary Journal of Knowledge and Learning Objects*. 2008, Vol. 3, <http://www.ijello.org/Volume3/IJKLOv3p057-072Parker284.pdf>.
33. **PAZ, Eduardo. 2005.** Características principales de Web 1_0, Web 1_5 y Web 2_0. *B2B-Blog*. [En línea] 23 de 11 de 2005. [Citado el: 2 de 12 de 2011.] http://e-global.es/b2b-blog/2005/11/23/caracteristicas-principales-de-web-1_0-web-1_5-y-web-2_0/.
34. *Planeta Web 2.0. Inteligencia colectiva o medios fast food.* **COBO, C. y PARDO, H. 2007.** Barcelona/México DF : Grupo de Recerca de Interacciones Digitales, Universidad de Vic. Flacso, México, 2007.
35. **Portal de Marketing Directo. 2011.** Tim O'Reilly inaugura FICOD 2011: "el futuro es la Web 3.0, la sensorial, no la semántica". *Marketing Directo*. [En línea] 22 de 11 de 2011. [Citado el: 2 de 12 de 2011.] <http://www.marketingdirecto.com/actualidad/eventos-y-formacion/tim-oreilly-inaugura-ficod-2011-el-futuro-es-la-web-3-0-la-sensorial-no-la-semantica/>.
36. **Proyecto jQuery. 2010.** Browser Compatibility. *jQuery*. [En línea] 2010. [Citado el: 8 de 12 de 2011.] http://docs.jquery.com/Browser_Compatibility.

37. **RAMAGE, Magnus. 2010.** *Online communication and collaboration: a reader.* s.l. : Routledge, 2010. 9780415564786.
38. **RapidSVN. 2011.** RapidSVN. [En línea] 2011. [Citado el: 13 de 12 de 2011.] <http://www.rapidsvn.org/>.
39. **ROSEN, Evan. 2007.** *THE CULTURE OF COLLABORATION.* s.l. : Red Ape Publishing, 2007. 13978-0-9774617-0-7.
40. **RUMBAUGH, James y JACOBSON, Ivar. 2007.** *El Lenguaje Unificado de Modelado. Manual de Referencia.* s.l. : ADDISON-WESLEY, 2007. 9788478290871.
41. **SCHWABER, Ken y BEEDLE, Mike. 2011.** *Agile Software Development with SCRUM.* s.l. : Prentice Hall, 2011. 0130676349.
42. **SURHONE, Lambert M, T Tennoe, Mariam y HENSSONOW, Susan F. 2010.** *Real-Time Web.* s.l. : VDM Verlag, 2010. 9786133432239.
43. *SXP, Metodología Ágil para el Desarrollo de Software.* **PEÑALVER, G., MENESES, A. y GARCÍA, S. 2010.** Chile : 1er congreso Iberoamericano de Ingeniería de Proyectos, 2010.
44. **URRA, Pedro. 2007.** *¿Qué es Infomed y dónde se encuentra? Una entrevista con Pedro Urra González, director de Infomed.* [entrev.] Maura E. Díaz Antúnez. Ciudad de la Habana : Editorial Ciencias Médicas, 15 de 04 de 2007. http://bvs.sld.cu/revistas/aci/vol15_04_07/aci21407.htm.
45. **VELASCO, Carolina. 2011.** *¿Qué es la web 3.0? ... o qué dicen que es....* *wwwwhat's new.* [En línea] 04 de 04 de 2011. [Citado el: 11 de 02 de 2012.] <http://wwwwhatsnew.com/2011/04/04/%C2%BFque-es-la-web-3-0-o-que-dicen-que-es/>.
46. **Visual Paradigm International Ltd. 2011.** Visual Paradigm for UML. *Visual Paradigm.* [En línea] 2011. [Citado el: 17 de 02 de 2012.] <http://www.visual-paradigm.com/product/vpuml/>.

47. **WEINZETTL, Michael. 2010.** *Online Collaboration*. s.l. : VDM Verlag, 2010. 9783639304114.
48. **WELLS, Don. 2009.** Extreme Programming: A gentle introduction. *Extreme Programming*. [En línea] 28 de 07 de 2009. [Citado el: 16 de 02 de 2012.] <http://www.extremeprogramming.org/>.

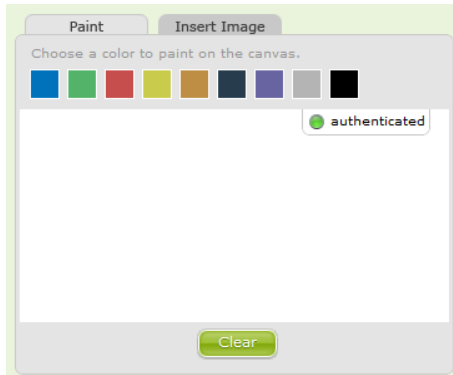
BIBLIOGRAFÍA CONSULTADA

1. *¿Qué es un 'framework'?* **SÁNCHEZ, Jordi. 2006.** Valencia : <http://jordisan.net/blog/2006/que-es-un-framework>, 2006.
2. **AQR8 Media. 2009.** Core Characteristics of Web 2.0 Services. *TechPluto*. [En línea] Junio de 2009. [Citado el: 1 de Diciembre de 2011.] <http://www.techpluto.com/web-20-services/>.
3. **CCOPACONDORI, Edwin. 2010.** Ventajas de jQuery. *no hay límites*. [En línea] 08 de 2010. [Citado el: 16 de 02 de 2012.] <http://www.nohaylimites.com/?p=141>.
4. *Comparación de Frameworks en Javascript*. **TAVÁREZ, David. 2009.** s.l. : <http://www.maestrosdelweb.com/editorial/comparacion-frameworks-javascript/>, 2009.
5. *Compare JavaScript frameworks*. **LENNON, Joe. 2010.** s.l. : <http://www.ibm.com/developerworks/web/library/wa-jsframeworks/index.html>, 2010.
6. **Eclipse. 2011.** *The Eclipse Foundation Open Source Community Website*. [En línea] Eclipse Foundation, 2011. <http://www.eclipse.org/>.
7. **FUMERO, A. y ROCA, G. 2007.** Web 2.0. *Fundación Orange España*. [En línea] 2007. http://www.fundacionorange.es/areas/25_publicaciones/publi_253_11.asp.
8. **IBM's Corporate. 2011.** Rational Rose Enterprise. *IBM*. [En línea] 2011. [Citado el: 17 de 02 de 2012.] <http://www-142.ibm.com/software/products/es/es/enterprise/>.
9. **Proyecto jQuery. 2010.** License. *jQuery*. [En línea] 2010. [Citado el: 8 de 12 de 2011.] <http://jquery.org/license/>.

10. *Qué es jQuery, para qué sirve y qué ventajas tiene el utilizar este framework Javascript.* **ALVAREZ, Miguel Angel. 2009.** s.l. : <http://www.desarrolloweb.com/articulos/introduccion-jquery.html>, 2009.
11. **SIMARD, D. 2007.** 3 reasons why I use jQuery. *JavascriptKata.* [En línea] 29 de 05 de 2007. [Citado el: 16 de 02 de 2012.] <http://www.javascriptkata.com/2007/05/29/3-reasons-why-i-use-jquery/>.
12. **TortoiseSVN Team . 2011.** TortoiseSVN. [En línea] 2011. [Citado el: 13 de 12 de 2011.] <http://tortoisesvn.net/>.

ANEXOS

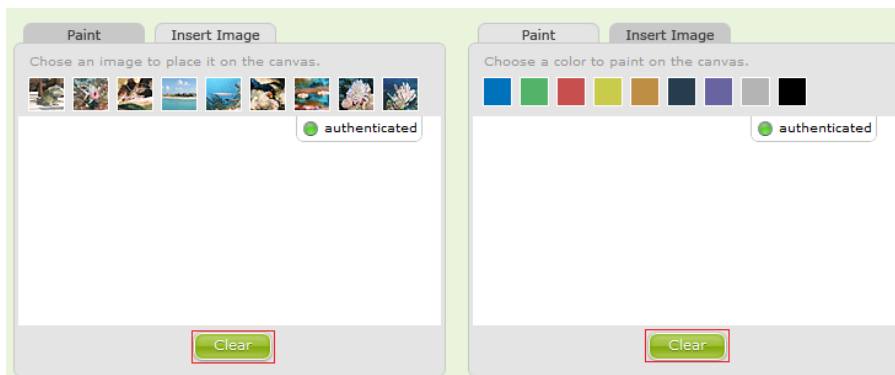
Anexo # 1: Dibujar



Anexo # 2: Insertar Imágenes



Anexo # 3: Limpiar Lienzos



Anexo # 4: Certificación de Calidad de Software

Anexo # 5: Valoración del Cliente

GLOSARIO DE TÉRMINOS

AJAX (Asynchronous JavaScript And XML): JavaScript Asíncrono y XML.

CASE (Computer Aided Software Engineering): Ingeniería de Software Asistida por Ordenador.

CSS (Cascading Style Sheets): Hojas de Estilo en Cascada.

CSV (Comma Separated Values): Valores separados por Comas.

DOM (Document Object Model): Modelo de Objetos del Documento.

GNU: GNU es el nombre de un completo sistema de software compatible con Unix. Se denominó así, según se afirma en el Proyecto GNU, principalmente para marcar la diferencia ya que GNU era un acrónimo recursivo de "GNU No es Unix".

GPL: Licencia Pública GNU.

HTML (Hyper Text Markup Language): Lenguaje de Marcado de Hipertexto.

HTTP (Hypertext Transfer Protocol): Protocolo de Transferencia de Hipertexto.

IDE (Integrated Development Environment): Entorno Integrado de Desarrollo.

IETF (Internet Engineering Task Force): Grupo Especial sobre Ingeniería de Internet.

JSON (JavaScript Object Notation): Notación de Objetos de JavaScript.

RUP (Rational Unified Process): Proceso Unificado de Desarrollo.

SOA (Service Oriented Architectures): Arquitectura Orientada a Servicios.

TCP (Transmission Control Protocol): Protocolo de Control de Transmisión.

XHTML (eXtensible Hyper Text Markup Language): Lenguaje de Marcado de Hipertexto Extensible.

XML (eXtensible Markup Language): Lenguaje de Marcas Extensible.

XP (Extreme Programming): Programación Extrema.