



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN SISTEMAS

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA WEB PARA SIMULAR
LOS FENÓMENOS ONDULATORIOS DE D’ALAMBERT Y FOURIER EN 2D
Y 3D APLICANDO MODELOS MATEMÁTICOS”**

TRABAJO DE TITULACIÓN

TIPO: Proyecto Técnico

Presentado para optar al grado académico de:

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTORES: LUIS MANUEL BARRAGÁN GONZÁLEZ

EFRAÍN AMADO COLCHA CEPEDA

TUTOR: DR. ALONSO WASHINGTON ÁLVAREZ OLIVO

Riobamba – Ecuador

2019

©2019, Luis Manuel Barragán González, Efraín Amado Colcha Cepeda

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SISTEMAS

El Tribunal del trabajo de titulación certifica que: El trabajo de investigación: Tipo Técnico “**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA WEB PARA SIMULAR LOS FENÓMENOS ONDULATORIOS DE D’ALAMBERT Y FOURIER EN 2D Y 3D APLICANDO MODELOS MATEMÁTICOS**”, de responsabilidad de los señores Luis Manuel Barragán González y Efraín Amado Colcha Cepeda, ha sido minuciosamente revisado por los Miembros del Tribunal del trabajo de titulación, quedando autorizada su presentación.

FIRMA

FECHA

Dr. Julio Santillán

**VICEDECANO FACULTAD DE
INFORMÁTICA Y ELECTRÓNICA**

Ing. MSc. Patricio Moreno

**DIRECTOR ESCUELA DE
INGENIERÍA EN SISTEMAS**

Dr. Alonso Álvarez

**DIRECTOR DE TRABAJO
DE TITULACIÓN**

Ing. Omar Gómez PhD.

MIEMBRO DEL TRIBUNAL

“Nosotros, **LUIS MANUEL BARRAGÁN GONZÁLEZ Y EFRAÍN AMADO COLCHA CEPEDA** somos responsables de las ideas, doctrinas y resultados expuestos en este trabajo de titulación; y el patrimonio intelectual del trabajo de titulación pertenece a la Escuela Superior Politécnica de Chimborazo”

Luis Manuel Barragán González

Efraín Amado Colcha Cepeda

DEDICATORIA

Dedicatoria A Dios por darme la oportunidad de vivir y por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente y por haber puesto en mi camino a aquellas personas que han sido mi soporte y compañía durante todo el periodo de estudio. A mis padres Juan y Dolores quienes con su amor, paciencia y esfuerzo me han permitido llegar a cumplir hoy un sueño más, gracias por inculcar en mí el ejemplo de esfuerzo y valentía, de no temer las adversidades.

Efraín

El presente trabajo de titulación va dedicado a mis padres: Mauro Barragán y Doris González, quienes me brindaron su apoyo incondicional no solo en mi carrera universitaria sino durante toda mi vida, dándome un gran ejemplo de perseverancia para que yo pueda conseguir mis metas tanto personales como profesionales.

Luis

AGRADECIMIENTO

Agradezco a Dios por protegerme durante todo mi camino y darme fuerzas para superar obstáculos y dificultades a lo largo de toda mi vida y a mis padres por su apoyo incondicional.

Efraín

A la Escuela Superior Politécnica de Chimborazo y a todos sus servidores politécnicos por brindarme todos los conocimientos necesarios y buenos valores éticos y morales para ser un excelente profesional.

Luis

TABLA DE CONTENIDO

| | |
|-------------------------|------|
| ÍNDICE DE TABLAS..... | x |
| ÍNDICE DE FIGURAS..... | xi |
| ÍNDICE DE GRÀFICOS..... | xii |
| ÍNDICE DE ANEXOS | xiii |
| RESUMEN | xiv |
| ABSTRACT..... | xv |
| INTRODUCCIÓN | 1 |

CAPITULO I

| | |
|---|---|
| 1. MARCO REFERENCIAL | 2 |
| 1.1 Antecedentes..... | 2 |
| 1.2 Formulación del problema | 3 |
| 1.3 Sistematización del problema..... | 4 |
| 1.4 Justificación del trabajo de titulación | 4 |
| 1.4.1 <i>Justificación teórica</i> | 4 |
| 1.4.2 <i>Justificación aplicativa</i> | 5 |
| 1.5 Objetivos..... | 6 |
| 1.5.1 <i>Objetivo general</i> | 6 |
| 1.5.2 <i>Objetivos específicos</i> | 6 |

CAPITULO II

| | |
|--|---|
| 2 MARCO TEÓRICO | 7 |
| 2.1. Computación gráfica | 7 |
| 2.1.1 <i>Introducción</i> | 7 |
| 2.1.2 <i>Antecedentes</i> | 7 |
| 2.1.3 <i>Animación informática</i> | 9 |

| | | |
|-------|---|-----------|
| 2.2 | Modelos matemáticos..... | 11 |
| 2.2.1 | <i>Concepto.....</i> | 11 |
| 2.2.2 | <i>Las TIC en la enseñanza de las matemáticas</i> | 11 |
| 2.2.3 | <i>Métodos numéricos</i> | 12 |
| 2.2.4 | <i>Ondas</i> | 12 |
| 2.2.5 | <i>Método de D’Alambert.....</i> | 15 |
| 2.2.6 | <i>Método de Fourier</i> | 16 |
| 2.3 | Herramientas de desarrollo..... | 19 |
| 2.3.1 | <i>Java</i> | 19 |
| 2.3.2 | <i>Glassfish</i> | 22 |
| 2.3.3 | <i>Bootstrap 4</i> | 23 |
| 2.3.4 | <i>JavaScript.....</i> | 24 |
| 2.3.5 | <i>PostgreSQL</i> | 25 |
| 2.3.6 | <i>NetBeans 8.2</i> | 28 |
| 2.4 | Modelo, vista, controlador..... | 28 |
| 2.5 | Metodología ágil Scrum..... | 29 |
| 2.5.1 | <i>Conceptos</i> | 29 |
| 2.5.2 | <i>Características</i> | 30 |
| 2.5.3 | <i>Roles</i> | 31 |
| 2.5.4 | <i>Ciclo de vida</i> | 32 |
| 2.5.5 | <i>Scrum vs otras metodologías ágiles</i> | 32 |
| 2.6 | Estándar ISO/IEC 9126..... | 34 |
| 2.6.1 | <i>Portabilidad.....</i> | 35 |
| 2.6.2 | <i>Eficiencia</i> | 35 |

CAPITULO III

| | | |
|-------|-----------------------------------|-----------|
| 3 | MARCO METODOLÓGICO..... | 36 |
| 3.1 | Fase de planificación..... | 36 |
| 3.1.1 | <i>Requerimientos</i> | 38 |
| 3.1.2 | <i>Roles del proyecto</i> | 40 |
| 3.1.3 | <i>Product backlog</i> | 40 |
| 3.1.4 | <i>Sprint backlog</i> | 41 |
| 3.1.5 | <i>Gestión de riesgos</i> | 42 |

| | | |
|------------|---|-----------|
| 3.2 | Fase de diseño..... | 46 |
| 3.2.1 | <i>Diagrama de casos de uso</i> | 46 |
| 3.2.2 | <i>Diagrama de clases</i> | 47 |
| 3.2.3 | <i>Diagrama de secuencia</i> | 48 |
| 3.2.4 | <i>Diagrama de estados</i> | 49 |
| 3.2.5 | <i>Diagrama de actividad</i> | 50 |
| 3.2.6 | <i>Arquitectura del sistema</i> | 50 |
| 3.2.7 | <i>Diagrama de despliegue</i> | 51 |
| 3.2.8 | <i>Estándar de codificación.....</i> | 52 |
| 3.2.9 | <i>Diseño de la interfaz de usuario</i> | 52 |
| 3.2.10 | <i>Diseño de la base de datos</i> | 52 |
| 3.3 | Fase de desarrollo | 52 |
| 3.3.1 | <i>Sprints</i> | 53 |
| 3.3.2 | <i>Historias de usuarios</i> | 53 |
| 3.3.3 | <i>Tablero de tareas</i> | 54 |
| 3.3.4 | <i>Pruebas de aceptación</i> | 55 |
| 3.4 | Fase de finalización | 56 |
| 3.4.1 | <i>Manual técnico</i> | 56 |
| 3.4.2 | <i>Manual de usuario</i> | 56 |
| 3.4.3 | <i>Implantación</i> | 56 |
| 3.4.4 | <i>Gráfico burndown chart</i> | 57 |

CAPITULO IV

| | | |
|----------|--|-----------|
| 4 | MARCO DE RESULTADOS..... | 58 |
| 4.1 | Generalidades..... | 58 |
| 4.2 | Portabilidad..... | 59 |
| 4.3 | Eficiencia | 61 |
| 4.3.1 | <i>Utilización de recursos</i> | 61 |
| 4.4 | Resultados..... | 63 |
| 4.5 | Respuesta a la sistematización del problema | 63 |

| | |
|-----------------------------|-----------|
| CONCLUSIONES..... | 65 |
| RECOMENDACIONES..... | 66 |
| BIBLIOGRAFÍA | |
| ANEXOS | |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 1-2: Comparativa lenguajes de programación. | 19 |
| Tabla 2-2: Características de Glassfish | 23 |
| Tabla 3-2: Características, ventajas y limitaciones. | 25 |
| Tabla 4-2: Ventajas y desventajas de PostgreSQL. | 26 |
| Tabla 5-2: Comparación gestores de base de datos. | 27 |
| Tabla 6-2: Sub-características portabilidad. | 35 |
| Tabla 7-2: Sub-características eficiencia. | 35 |
| Tabla 8-3: Roles del Proyecto. | 40 |
| Tabla 9-3: Product backlog. | 41 |
| Tabla 10-3: Sprint backlog. | 42 |
| Tabla 11-3: Identificación de riesgos. | 43 |
| Tabla 12-3: Resultados análisis de riesgos. | 43 |
| Tabla 13-3: Hoja de gestión del riesgo R1. | 44 |
| Tabla 14-3: Presupuesto del proyecto. | 45 |
| Tabla 15-3: Caso de uso - Graficar onda 2D caso finito. | 47 |
| Tabla 16-3: Sprint2 - Acciones de onda en dos dimensiones. | 53 |
| Tabla 17-3: Historia de usuario – Generar histórico ondas 2D. | 54 |
| Tabla 18-3: Tarea de ingeniería – Generar histórico ondas 2D. | 54 |
| Tabla 19-3: Prueba de aceptación – Generar histórico ondas 2D. | 55 |
| Tabla 20-4: Parámetros e indicadores. | 58 |
| Tabla 21-4: Criterios de evaluación. | 59 |
| Tabla 22-4: Casos de estudio. | 59 |
| Tabla 23-4: Resultados portabilidad. | 60 |
| Tabla 24-4: Casos de estudio. | 61 |
| Tabla 25-4: Resultados utilización recursos. | 62 |

ÍNDICE DE FIGURAS

| | |
|--|----|
| Figura 1-2: Primer Programa para dibujar Sketchpad. | 8 |
| Figura 2-2: Animación computarizada | 9 |
| Figura 3-2: Vibración de una cuerda. | 12 |
| Figura 4-2: Vibración de una membrana. | 12 |
| Figura 5-2: Java JDK | 22 |
| Figura 6-2: Sistema de rejillas Bootstrap 4..... | 24 |
| Figura 7-2: Patrón MVC. | 29 |
| Figura 8-2: Proceso Scrum. | 30 |
| Figura 9-2: Comparativa metodologías ágiles. | 33 |
| Figura 10-2: Características ISO 9126. | 34 |
| Figura 11-3: Actividades planificadas. | 37 |
| Figura 12-3: Diagrama Gantt. | 37 |
| Figura 13-3: Diagrama casos de uso..... | 46 |
| Figura 14-3: Diagrama de clases. | 47 |
| Figura 15-3: Diagrama de objetos. | 48 |
| Figura 16-3: Diagrama de secuencia - Graficar onda 2D caso finito. | 48 |
| Figura 17-3: Diagrama de colaboración – Graficar onda 2D caso finito. | 49 |
| Figura 18-3: Diagrama de estados..... | 49 |
| Figura 19-3: Diagrama de actividad - Graficar onda 2D caso finito. | 50 |
| Figura 20-3: Arquitectura del sistema. | 51 |
| Figura 21-3: Diagrama de despliegue..... | 51 |
| Figura 22-3: Gráfico burndown chart..... | 57 |
| Figura 23-4: Monitor de recursos. | 61 |

ÍNDICE DE GRÁFICOS

| | |
|--|----|
| Gráfico 1-4: Resultados portabilidad..... | 60 |
| Gráfico 2-4: Resultados utilización recursos. | 62 |

ÍNDICE DE ANEXOS

- ANEXO A:** Gestión de riesgos
- ANEXO B:** Diagrama casos de uso
- ANEXO C:** Diagrama de secuencia
- ANEXO D:** Diagrama de colaboración
- ANEXO E:** Diagrama de actividad
- ANEXO F:** Sprint del proyecto
- ANEXO G:** Estándar de codificación
- ANEXO H:** Diseño de interfaces
- ANEXO I:** Diseño base de datos
- ANEXO J:** Manual técnico
- ANEXO K:** Manual de usuario

RESUMEN

En el presente trabajo de titulación se realizó el desarrollo del sistema web para simular los fenómenos ondulatorios en 2D Y 3D aplicando los modelos matemáticos de ecuaciones diferenciales en derivadas parciales y las soluciones de D’Alambert y Fourier utilizando el lenguaje de programación java, Bootstrap para el diseño de las interfaces web y PostgreSQL como gestor de base de datos. El sistema web fue realizado en base a la metodología de desarrollo ágil SCRUM, la cual ayudó a gestionar las actividades de manera óptima generando así un ambiente de trabajo agradable para cliente y el equipo de desarrollo. Se utilizaron herramientas tecnológicas como: JavaScript, Bootstrap 4.1 Framework, PostgreSQL 10.1, Glassfish Server 4.1 y NetBeans IDE 8.2. El producto final se evaluó de acuerdo con las métricas de evaluación que indica la norma ISO/IEC 9126 en base a la portabilidad y eficiencia, teniendo como resultado que el sistema web posee un 100% de portabilidad y un 12% de utilización de recursos. De esta manera se llegó a la conclusión de que el sistema es portable y eficiente, cumpliendo así con la calidad de software y aportando al proceso de enseñanza- aprendizaje del cálculo vectorial en la Facultad de Informática y Electrónica de la Escuela Superior Politécnica de Chimborazo de la ciudad de Riobamba. Se recomienda revisar el manual de usuario adjunto al documento para una correcta manipulación del sistema.

Palabras Clave: <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>;<INGENIERÍA DE SOFTWARE>;<COMPUTACIÓN GRÁFICA>;<METODOS NUMÉRICOS><MODELOS MATEMÁTICOS>;<DESARROLLO DE APLICACIONES WEB>;<SCRUM (METODOLOGÍA DE DESARROLLO ÁGIL) >;<CALIDAD DE SOFTWARE>.

ABSTRACT

In the present work of degree the development of the web system was developed to simulate the wave phenomena in 2D and 3D applying the mathematical models of differential equations in partial derivatives and the solutions of D'Alambert and Fourier using the programming language java, Bootstrap for the design of web interfaces and PostgreSQL as a database manager. The web system was made based on the agile development methodology SCRUM, which helped to manage the activities in an optimal way, generating a pleasant work environment for the client and the development team. Technological tools were used such as: JavaScript, Bootstrap 4.1 Framework, PostgreSQL 10.1, Glassfish Server 4.1 and NetBeans IDE 8.2. The final product was evaluated according to the evaluation metrics indicated by ISO / IEC 9126 based on portability and efficiency, resulting in the web system having 100% portability and 12% resource utilization. In this way, it was concluded that the system is portable and efficient, thus complying with the quality of software and contributing to the teaching-learning process of vectorial calculation in the Faculty of Informatics and Electronics of the Polytechnic School of Chimborazo in the city of Riobamba. It is recommended to review the user manual attached to the document for a correct manipulation of the system.

Keywords: <TECHNOLOGY AND ENGINEERING SCIENCE>; <SOFTWARE ENGINEERING>; <GRAPHIC COMPUTATION>; <NUMERICAL METHODS> <MATHEMATICAL MODELS>; <WEB APPLICATION DEVELOPMENT>; <SCRUM (AGILE DEVELOPMENT METHODOLOGY)>; < SOFTWARE QUALITY>.

INTRODUCCIÓN

El constante avance tecnológico ha hecho de las TIC una potente herramienta para el análisis de contenido científico, hoy en día, la creación de sistemas científicos se constituye como un gran soporte para la investigación y toma de decisiones en campos científicos y tecnológicos, las instituciones de educación superior usan las TIC para potenciar el proceso de enseñanza aprendizaje y brindar apoyo en los campos científicos con elevado nivel de complejidad.

El estudio de los fenómenos ondulatorios es una parte fundamental para entender el mundo que nos rodea, con la ayuda de las ondas podemos percibir el sonido la luz y en general la mayor parte de información que se obtiene es gracias a las pequeñas perturbaciones de las ondas.

Con el desarrollo del sistema web se pretende ofrecer una herramienta informática para el estudio de los fenómenos ondulatorios en 2D y 3D aplicando las fórmulas de D’Alambert y Fourier, de esta manera facilitar el estudio de las ondas y apoyar en la enseñanza de las aplicaciones del cálculo vectorial en la facultad de informática y electrónica de la ESPOCH.

El presente trabajo de titulación está dividido en cuatro capítulos en donde el capítulo I Marco referencial se detallan los aspectos técnicos y la información base para el desarrollo del Sistema web.

En el capítulo II Marco teórico se describe la información sobre las herramientas tecnológicas aplicadas en el desarrollo del Sistema de web.

En el capítulo III Marco metodológico se detalla el desarrollo del sistema web denominado “Sistema web para simular los fenómenos ondulatorios de D’Alambert y Fourier en 2D y 3D”, aplicando la metodología de desarrollo ágil SCRUM.

Finalmente, en el capítulo IV Marco de Resultados se detalla el proceso de evaluación de la calidad sistema en base a las métricas de eficiencia y portabilidad, aplicando la norma ISO/IEC 9126.

CAPITULO I

1. MARCO REFERENCIAL

1.1 Antecedentes

El constante avance tecnológico y científico ha venido revolucionando al mundo a un ritmo acelerado, a lo largo de la historia de la humanidad, se ha procurado asegurar y mejorar la calidad de vida gracias al conocimiento de todo lo que nos rodea, es decir a través de un desarrollo incesante de la ciencia.

La aplicación de las TIC ha resultado primordial para el análisis del contenido científico que se genera actualmente, dando a lugar a la creación de sistemas de información científicas y técnicas que se constituyen como un gran soporte para la investigación y toma de decisiones en campos científicos y tecnológicos.

Hoy en día el internet se ha convertido en una herramienta indispensable para el acceso y compartición de la información, gracias a ello se ha desarrollado tecnologías y herramientas que facilitan la creación de sistemas informáticos, logrando así obtener resultados tanto óptimos como eficaces.

El desarrollo de sistemas informáticos es una de las ramas de la ingeniería que se encarga principalmente de la creación de software con la ayuda de herramientas de desarrollo, Estas herramientas tienen como objetivo disminuir los tiempos en el desarrollo, así como el generar mejores propuestas para el usuario final. El desarrollar software implica una actividad compleja y extensa, por ello utilizar algunas de las herramientas disponibles ayudan a reducir el estrés y aumentar el desempeño tanto en desarrolladores como en las aplicaciones resultantes.

Desde sus inicios las herramientas de desarrollo han venido desempeñando un papel importante dentro del desarrollo de software, con los continuos cambios consecuencia de un constante avance tecnológico, existen gran cantidad de estas herramientas que apoyan en múltiples formas dentro del desarrollo de software en general.

El conocimiento es una de las bases de la sociedad actual, el combinar las tecnologías con el conocimiento da lugar a sistemas educativos especializados en mejorar los procesos de enseñanza y aprendizaje en donde los estudiantes y docentes salen altamente beneficiados. La aplicación de un simulador de ondas aportara en el proceso de la enseñanza de los métodos numéricos, proceso con un alto grado de complejidad debido a los modelos matemáticos que ahí se trabajan.

La Escuela Superior Politécnica de Chimborazo, es una institución con personería jurídica de derecho público totalmente autónoma, se rige por la Constitución Política del Estado ecuatoriano, la ley de educación superior y por su propio estatuto y reglamentos internos y tiene su domicilio principal en la ciudad de Riobamba.

Dentro de los fines que la Escuela Superior Politécnica de Chimborazo ofrece se encuentra realizar investigación científica y tecnológica para garantizar la generación, asimilación y adaptación de conocimientos que sirvan para solucionar los problemas de la sociedad ecuatoriana. el objetivo de realizar este trabajo de titulación es el de desarrollar una aplicación web que mejore el proceso de enseñanza de los modelos matemáticos de las ecuaciones de ondas y de esta manera fomentar los procesos de investigación, generación y asimilación del conocimiento.

1.2 Formulación del problema

¿LOS MÉTODOS NUMÉRICOS PODRÁN REPRESENTAR Y RESOLVER LA SOLUCIÓN DEL MODELO MATEMÁTICO DE LA PROPAGACIÓN DE ONDAS EN 2D Y 3D Y LAS FÓRMULAS DE D'ALAMBERT Y FOURIER CON LA AYUDA DE UN SISTEMA WEB?

1.3 Sistematización del problema

¿Cuáles son las soluciones de los modelos matemáticos para las ecuaciones de ondas de D’Alambert y Fourier en 2D Y 3D?

¿El desarrollo del sistema ayudará en la enseñanza de las aplicaciones del cálculo vectorial en la ESPOCH?

¿Cómo se puede representar las fórmulas de D’Alambert y Fourier a través de un sistema web?

¿Cuál es la importancia de desarrollar sistemas para la simulación de modelos matemáticos en la ESPOCH?

1.4 Justificación del trabajo de titulación

1.4.1 Justificación teórica

La aplicación de simuladores en el ámbito educativo ayuda a los estudiantes y docentes a crear un marco para la exploración y practica de contenidos a probarse en un entorno sin riesgos.

En todas las áreas de la educación, el uso de las tecnologías ha generado grandes beneficios apoyando directamente en el proceso de enseñanza y aprendizaje. “El simulador permite al estudiante aprender de manera práctica, a través del descubrimiento y la construcción de situaciones hipotéticas.” (Brusquetti 2011).

La ventaja que ofrecen los simuladores a las estudiantes y docentes es la de generar destrezas tanto mentales como físicas en situaciones sin riesgos, además estimulan el trabajo en equipo y mejoran la comprensión y comunicación entre docentes y estudiantes (Brusquetti 2011).

Para el desarrollo del simulador de fenómenos ondulatorios en 2D y 3D se pretende utilizar modelos matemáticos con una complejidad elevada que son imposibles de resolver con modelos analíticos, y así poder graficarlos a través un sistema utilizando los gráficos por computadora.

A través de este, se podrá estudiar el comportamiento de las ecuaciones de ondas en 2D y 3D y de esa manera facilitar la enseñanza de las aplicaciones del cálculo vectorial en la ESPOCH.

1.4.2 Justificación aplicativa

Para el desarrollo del simulador de fenómenos ondulatorios en 2D Y 3D se aplicará las fórmulas de D’Alambert y Fourier y las ecuaciones de ondas en dos y tres dimensiones, utilizando el lenguaje de programación java y PostgreSQL como gestor de base de datos, además este sistema será evaluado en calidad interna según las métricas de eficiencia y portabilidad mediante la ISO/IEC 9126, este sistema contará con los siguientes módulos:

Módulo de diseño

- Diseño y maquetación del sistema.
- Preparación del área de trabajo.

Módulo ondas en dos dimensiones

- Identificación de condiciones iniciales.
- Análisis del modelo matemático del problema.
- Resolución analítica y numérica.

Módulo ondas en tres dimensiones

- Identificación de condiciones iniciales.
- Análisis del modelo matemático del problema.
- Resolución analítica y numérica.

Este proyecto de desarrollo aplicativo se sustenta en la Líneas de investigación de la ESPOCH en la de Tecnología de la Información, Comunicación, Procesos Industriales y Biotecnológicos, en el Programa para el desarrollo de aplicaciones de software, hardware y telecomunicaciones. Cuyo eje

transversal es el de Aplicación de Ingeniería en la línea de investigación para Proceso de desarrollo de software en el ámbito de Construcción de Software.

1.5 Objetivos

1.5.1 Objetivo general

DISEÑAR E IMPLEMENTAR UN SISTEMA WEB PARA SIMULAR LOS FENÓMENOS ONDULATORIOS DE D'ALAMBERT Y FOURIER EN 2D Y 3D APLICANDO MODELOS MATEMÁTICOS.

1.5.2 Objetivos específicos

- Analizar y describir la solución de los modelos matemáticos de D'Alambert y Fourier para 2D y 3D.
- Definir los requerimientos para el desarrollo del sistema.
- Desarrollar e implementar el sistema web para simular los fenómenos ondulatorios y representar la solución de modelos matemáticos de las ecuaciones de ondas de D'Alambert y Fourier para 2D y 3D en la ESPOCH.
- Evaluar la calidad en eficiencia y portabilidad del simulador, de acuerdo con la norma ISO/IEC 9126.

CAPITULO II

2 MARCO TEÓRICO

2.1. Computación gráfica

2.1.1 *Introducción*

La computación grafica es una rama de la informática que se encarga del estudio de los métodos y técnicas que ayudan en la representación de imágenes del mundo real y no real a través de un computador, se le puede definir como el arte de transmitir emociones e intercambiar información mediante imágenes generadas por el computador.

2.1.2 *Antecedentes*

La computación grafica tiene su origen con la aparición de los computadores digitales, las primeras aplicaciones realizadas con la ayuda de los gráficos por computadora fueron para visualizar gráficos de datos que, comúnmente, se imprimían con el apoyo de impresoras de caracteres.

Para los años de 1960 un estudiante del MIT creó un programa al cual denominó Sketchpad, mediante el cual podía dibujar trazos en la pantalla de un computador con la ayuda de una pluma de luz, más tarde en el año de 1961, se creó el primer videojuego guerra espacial que fue todo un rotundo éxito.

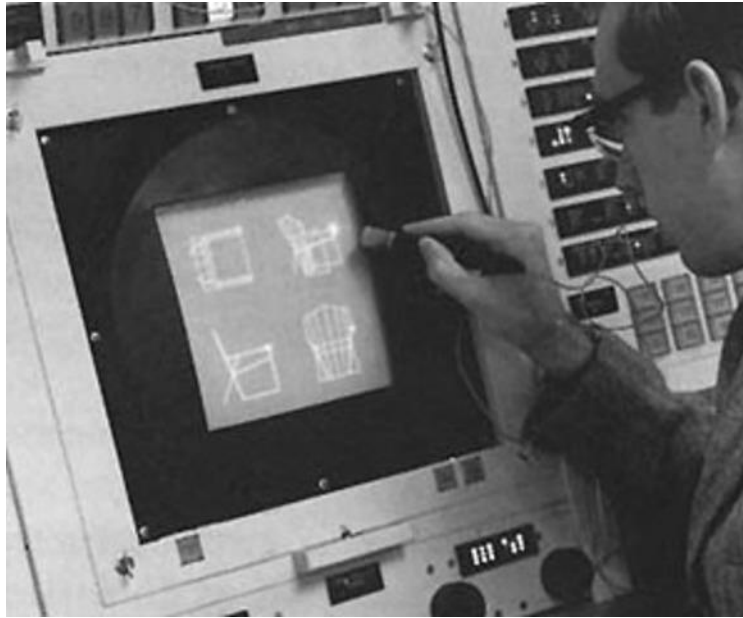


Figura 1-2: Primer Programa para dibujar Sketchpad.

Fuente: <https://www.timetoast.com/timelines/antecedentes-y-la-evolucion-de-la-graficacion-por-computadora>

En los años 70 la computación grafica se introdujo en la producción de películas y series de televisión, generando así un gran impacto en el mundo de la animación, un claro ejemplo de esto es la película de Jurassic Park en donde se crearon dinosaurios con la ayuda de efectos visuales nunca vistos en la industria de la animación y el cine.

Otro de las aplicaciones de la computación grafica es la representación de datos, “Todavía la representación gráfica de datos es una de las aplicaciones más comunes, pero hoy podemos generar fácilmente gráficos que muestren complejas relaciones entre datos para realizar informes escritos o para presentarlos mediante diapositivas, transparencias o animaciones en video”. (Donald Hearn and M. Pauline Baker, 2006).

2.1.3 Animación informática

Según (Gutiérrez, Cerezo y Pulido, 2006) “La animación es el arte. Los ordenadores son sólo el medio”.

La animación por computadora se la puede definir como la presentación de una secuencia de imágenes en movimiento generadas por el computador, es usada comúnmente en las películas animadas y series de tv (Suarez, 2003).

La animación por computadora se refiere, a toda secuencia temporal donde se generen cambios visuales en una imagen. Así como cambios en las posiciones de los objetos ya sean traslaciones o rotaciones, una animación por computadora puede mostrar cambios temporales que afecten al tamaño de los objetos como son: su color, transparencia y texturas (Donald Hearn and M. Pauline Baker, 2006).



Figura 2-2: Animación computarizada

Fuente: <http://Insinformatica10.blogspot.com/2017/02/animaciones.html>

2.1.3.1 Animación 2D

La animación 2D es una técnica informática que permite agregar movimientos a dibujos en dos dimensiones, para esto es necesario generar una cantidad considerable de imágenes en las cuales pueden variar el color, ubicación y el tamaño (Quijano, 2013).

Con la ayuda de la tecnología la animación 2D se popularizó en el siglo 19 siendo las técnicas mejoradas por Walt Disney y utilizadas en sus dibujos animados, la animación 2D es una forma de arte con grandes capacidades que dieron el origen a la animación 3D la que se utiliza para crear efectos especiales en las películas (Jones, 2018).

2.1.3.2 Animación 3D

La ventaja principal de la animación 3D es la posibilidad de poder girar la cámara y así ver otro ángulo de una escena, a diferencia de la animación 2D este se debe realizar por un computador con la ayuda de software especializados (Suarez, 2003).

Las técnicas de la animación 3D, hoy en día tienen un papel altamente importante en el mundo de la animación. Desde sus orígenes, a mediados de los años 80, la técnica se ha ido perfeccionando, y a la vez combinándose con otras técnicas para crear complejos procesos capaces de llegar a un grado de absoluto realismo en una imagen. Estas técnicas de la animación 3D tienen un rango de estética alta y abarca estilos tan diferentes como los dibujos animados, imágenes realistas o los videojuegos (Cuesta, 2015).

2.2 Modelos matemáticos

2.2.1 *Concepto*

Se define como modelo matemático al conjunto de herramientas que ayudan a entender un sistema complejo que es afectado por muchas variables de maneras diversas, estos modelos ayudan al ser humano a percibir de mejor manera cálculos que no se pueden resolver con aritmética simple, además son utilizados como herramientas de predicción y estimación de posibles escenarios y así ayudar en el proceso de toma de decisiones (Oviedo Rondón, 2014).

2.2.2 *Las TIC en la enseñanza de las matemáticas*

La evolución de las tecnologías ha favorecido a muchas áreas como es la educación, según (Granados Ospina, 2014) los estudiantes en la actualidad están inmersos en las tecnologías al igual que las institución educativas, y esto facilita el proceso de enseñanza -aprendizaje.

El estudio de las matemáticas no puede dejar a un lado las herramientas que la tecnología nos brinda, por lo que es necesario la incorporación de nuevas técnicas o herramientas por medio de la cual el proceso de enseñanza aprendizaje mejore (Granados Ospina, 2014).

El uso de un software especializado facilitaría la comprensión de las matemáticas y sobre todo en la ejecución de fórmulas complejas, además con la ayuda de las TIC se puede potenciar y enriquecer el aprendizaje.

2.2.3 Métodos numéricos

Los métodos numéricos son una parte de la matemática que ayuda en la investigación y operaciones de ingeniería, según (Carlos y Soler, 2010), el objetivo de los métodos numéricos es ayudar a dar respuestas a problemas mediante aproximaciones suficientemente exactas con un esfuerzo mínimo cuando no es posible obtener la solución por métodos analíticos.

En palabras de Carlos y Soler (2010) “En todas las labores para el desarrollo científico y tecnológico se utiliza hoy en día el computador, con un software especializado que es ajustado a la medida según alguno de los métodos numéricos”. El poder determinar el mejor software va a depender de las condiciones del problema y la utilización de los métodos para la resolución.

2.2.4 Ondas

En palabras de (Castaños, 2016) “Una onda es una perturbación que se propaga a través de un determinado medio o en el vacío, con transporte de energía pero sin transporte de materia”. Se puede decir que la onda es la propagación de una perturbación desde un punto de origen hasta el medio que lo rodea.



Figura 3-2: Vibración de una cuerda.
Realizado por: Barragán Luis, Colcha Efraín. 2018



Figura 4-2: Vibración de una membrana.
Realizado por: Barragán Luis, Colcha Efraín. 2018

Según (Jiménez Quintana, 2015) las ecuaciones de ondas son modelos que se describen en términos de ecuaciones diferenciales en derivadas parciales, ya que en ellas interviene la mecánica, física e ingeniería, es un opuesto exacto a la ecuación de calor ya que es un sistema reversible en el tiempo y su velocidad de propagación es finita.

En palabras de (Zuazua, 2009) en las aplicaciones físicas la dimensión espacial es normalmente $n = 1, 2$ y 3 ; en donde las ecuaciones de ondas con dimensiones $n = 1$ y 2 permite representar vibraciones de cuerdas y membranas, mientras que las ecuaciones de ondas con $n = 3$ representa la propagación de un campo acústico.

2.2.4.1 Ecuación de ondas en dos dimensiones

Para la obtención de la ecuación de ondas en dos dimensiones se toma como punto de partida una cuerda uniforme flexible con densidad lineal ρ estirada mediante una tensión de T entre los puntos fijos $x = 0$ y $x = l$, mientras la cuerda vibra en el plano en su posición de equilibrio, cada punto se mueve paralelo al eje y , lo que se puede representarse por $y(x, t)$, es decir el comportamiento de x en cada instante de tiempo t , además se considera que aparte de las fuerzas internas de tensión actuando tangencialmente a la cuerda, se encuentra otra fuerza vertical externa con densidad lineal $f(x)$ (Blanchard, Devaney y Hall 2009).

Al aplicar la segunda ley de Newton $F = ma$ sobre un pequeño segmento de cuerda de masa $\rho\Delta x$ perteneciente a un intervalo $[x, x + \Delta x]$, siendo a la aceleración vertical de $u(x, t)$ en su punto medio, se obtiene la ecuación diferencial que describe las vibraciones verticales de una cuerda flexible con densidad lineal constante ρ y tensión T bajo la influencia de una fuerza vertical externa con densidad lineal $f(x)$, además si se considera $\alpha^2 = T/\rho$ y $f(x) = 0$ se obtiene la ecuación de ondas en dos dimensiones: $\frac{\partial^2 u(x,t)}{\partial t^2} = c^2 \frac{\partial^2 u(x,t)}{\partial x^2}$ (Blanchard, Devaney y Hall 2009).

Según (Blanchard, Devaney y Hall 2009) “El movimiento de la cuerda estará determinado si se especifican tanto su función de posición inicial $u(x, 0) = f(x)$ y su función de velocidad inicial $u_t(x, 0) = g(x)$ ”, en donde se obtiene el problema con valores de frontera para la función de desplazamiento $u(x, t)$ de una cuerda vibrando libremente con extremos fijos, posición inicial $f(x)$ y velocidad inicial $g(x)$.

2.2.4.2 Ecuación de onda en tres dimensiones

Para la obtención de la ecuación de onda en tres dimensiones se toma como punto de partida el análisis de las fuerzas que actúan sobre una pequeña porción de una membrana, de ahí se deduce la ecuación diferencial que rige el movimiento ondulatorio de la membrana. siendo las deflexiones de la membrana y los ángulos de inclinación pequeños, las fuerzas que actúan sobre los lados equivalen a $T\Delta x$ y $t\Delta y$, que son fuerzas tangentes a la membrana (Melgarejo, Gonzales y Ramírez-Martín 2013).

Según (Melgarejo, Gonzales y Ramírez-Martín 2013) “Al obtener la proyección de las fuerzas sobre el eje horizontal, resultan afectadas por el coseno del ángulo de inclinación, como son ángulos pequeños su coseno tiende a uno; por ende las componentes horizontales de las fuerzas en lados opuestos son iguales y se anulan”, es decir que las partículas de la membrana se mueven verticalmente.

Al aplicar la segunda ley de Newton, en donde la suma de la componente vertical de las fuerzas es igual al producto de la masa $\rho\Delta A$ de esa porción pequeña de membrana y la aceleración $\delta^2 u / \delta t^2$, se tiene que ρ es la masa y $\Delta A = \Delta x \Delta y$ es el área de la porción de la membrana no flexionada, a esto se suma la evaluación de Δx y Δy cuando tienden a cero; se obtiene la ecuación de ondas en tres dimensiones: $\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} = \frac{1}{c^2} \frac{\partial^2 w}{\partial t^2}$ (Melgarejo, Gonzales y Ramírez-Martín 2013).

2.2.5 Método de D'Alambert

2.2.5.1 Solución en dos dimensiones

Para la solución a la ecuación de onda D'Alambert se considera una cuerda de longitud fija en sus dos extremos, puesta en movimiento para determinar la función que da la ordenada, es decir la excursión de cada punto de la cuerda de abscisa x , a cada instante t (Campo Bedoya 2014).

Consideremos el problema de condiciones iniciales:

$$\frac{\partial^2 u(x,t)}{\partial t^2} = c^2 \frac{\partial^2 u(x,t)}{\partial x^2} \text{ Ecuación de onda para dos dimensiones.}$$

$u(x, 0) = f(x)$ función de la posición inicial.

$u_t(x, 0) = g(x)$ función de la velocidad final.

Para encontrar la solución se utiliza el cambio de variables: $v(w, z) = u(x, t)$, donde $w = x + ct$ y $z = x - ct$. A partir de lo anterior se puede reescribir la función $u(x, t)$ como $u(x, t) = F(x + ct) + G(x - ct)$ y al imponer las condiciones iniciales se obtiene que $f(x) = F(x) + G(x)$ y $g(x) = c(F'(x) - G'(x))$.

Finalmente se puede reescribir la solución $u(x, t) = F(x + ct) + G(x - ct)$ en función de posición inicial y velocidad inicial como: $u(x, t) = \frac{1}{2} (f(x + ct) + f(x - ct)) + \frac{1}{2}c \int_{x-ct}^{x+ct} g(s)ds$, a esta solución se la conoce como la Formula de D'Alambert de la Ecuación de Ondas (Jiménez Quintana 2015).

Si $f(x)$ es el perfil inicial para la perturbación inicial de los distintos puntos de la cuerda, $f(x + ct)$ representa un perfil idéntico al inicial que se desplaza hacia la derecha, De la misma manera, $f(x - ct)$ representa un perfil idéntico al inicial que se desplaza con velocidad c hacia la izquierda, es decir, que el comportamiento de una onda se puede analizar al dividir la onda en sus componentes (Benguria 2012).

2.2.6 Método de Fourier

2.2.6.1 Solución en dos dimensiones

Consideremos el problema con valores frontera:

$$\frac{\partial^2 u(x,t)}{\partial t^2} = c^2 \frac{\partial^2 u(x,t)}{\partial x^2} \quad 0 < x < L \text{ Ecuación de onda para dos dimensiones.}$$

$u(x, 0) = f(x), 0 < x < L, t = 0$; siendo $f(x)$ la función de la posición inicial.

$u_t(x, 0) = g(x), 0 < x < L, t = 0$; siendo $g(x)$ la función de la velocidad final.

Usando el método de separación de variables, e ignorando momentáneamente las condiciones iniciales se puede decir $u(x, t) = X(x) - T(t)$, en donde resolviendo el problema con valores de frontera anterior, la solución viene dada por la forma $x_n(x) = C1\cos(\sqrt{\lambda x}) + C2\sin(\sqrt{\lambda x})$, en donde los autovalores son de la forma $\lambda_n = \left(\frac{n\pi}{L}\right)^2$, reemplazando λ_n en la solución de la ecuación y por el principio de superposición se obtiene la ecuación conocida como la solución por el método de Fourier: $u(x, t) = \sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right) \left(\frac{a_n \cos(n\pi ct)}{L} + \frac{b_n \sin(n\pi ct)}{L}\right)$ (Jiménez Quintana 2015).

Considerando ahora las condiciones iniciales, se determina los coeficientes a_n y b_n , a partir del desplazamiento inicial $f(x)$ y la velocidad inicial $g(x)$, siendo $a_n = \frac{2}{L} \int_0^L \frac{f(x) \sin(n\pi x)}{L} dx$ y $b_n = \frac{2}{n\pi c} \int_0^t \frac{g(x) \sin(n\pi x)}{L} dx$ (Jiménez Quintana 2015).

Según (Blanchard, Devaney y Hall 2009), con los coeficientes a_n calculados la solución al coeficiente b_n se obtiene de la serie seno de Fourier de $f(x)$ insertando simplemente el factor $\cos\left(\frac{n\pi ct}{L}\right)$ en el n-ésimo término.

2.2.6.2 Solución en tres dimensiones

Considérese la ecuación de ondas para la vibración de una membrana rectangular:

$$\frac{\delta^2 w}{\delta x^2} + \frac{\delta^2 w}{\delta y^2} = \frac{1}{c^2} \frac{\delta^2 w}{\delta t^2}$$

Siendo las condiciones de frontera:

$$u(x, 0, t) = 0, \quad u(x, l_2, t) = 0, \quad x \in [0, l_1]$$

$$u(0, y, t) = 0, \quad u(l_1, y, t) = 0, \quad y \in [0, l_2]$$

Las condiciones iniciales:

$$u(x, y, 0) = f(x, y) \text{ Función de la posición inicial}$$

$$u_y(x, y, 0) = g(x, y) \text{ Función de la velocidad inicial}$$

Aplicando el método de Fourier para la resolución de la ecuación de onda bidimensional se obtiene la ecuación $u(x, y, t) = X(x)Y(y)T(t)$, en donde imponiendo las condiciones de frontera se llega a: $X(0) = X(l_1) = 0$, $Y(0) = Y(l_2) = 0$ y aplicando el mismo proceso que se utiliza en la ecuación de onda en una cuerda se obtiene: $X_n(x)Y_m(y) = \text{sen}\left(\frac{n\pi x}{l_1}\right)\text{sen}\left(\frac{m\pi y}{l_2}\right)$.

Finalmente resolviendo la ecuación $\frac{d^2 T}{dt^2} + a^2 \lambda_{n,m} T = 0$ se obtiene la solución general a la ecuación de onda en tres dimensiones conocida como la solución por el método de Fourier para ondas en 3D: $u(x, y, t) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \left(a_{n,m} \cos(a\sqrt{\lambda_{n,m}}t) + b_{n,m} \text{sen}(a\sqrt{\lambda_{n,m}}t) \right) \text{sen}\left(\frac{n\pi x}{l_1}\right) \text{sen}\left(\frac{m\pi y}{l_2}\right)$ (Peiró 2008).

Aplicando las condiciones iniciales se obtiene las ecuaciones de los coeficientes $a_{n,m}$ y $b_{n,m}$ siendo:

$$a_{n,m} = \frac{4}{l_1 l_2} \int_0^{l_1} dx \int_0^{l_2} dy f(x, y) \text{sen}\left(\frac{n\pi x}{l_1}\right) \text{sen}\left(\frac{m\pi y}{l_2}\right)$$

$$b_{n,m} = \frac{4}{l_1 l_2 a \sqrt{\lambda_{n,m}}} \int_0^{l_1} dx \int_0^{l_2} dy g(x, y) \text{sen}\left(\frac{n\pi x}{l_1}\right) \text{sen}\left(\frac{m\pi y}{l_2}\right)$$

2.3 Herramientas de desarrollo

2.3.1 Java

Dentro del desarrollo de aplicaciones existen una variedad de lenguajes de programación, tecnologías que han sido fuente de grandes debates sobre cuál de ellos es el mejor, sin embargo desde el punto de vista del desarrollador no existe una opinión fija, cada uno posee sus seguidores y detractores (Saray, 2015). En la **tabla 1-2** se presenta una comparativa de los lenguajes de programación más utilizados.

Tabla 1-2: Comparativa lenguajes de programación.

| Lenguaje | Paradigma | Sistema operativo | Características | Limitaciones |
|---------------|---|---|---|--|
| JAVA | Es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90 (Saray, 2015). | Lenguaje multiplataforma (Gomez, 2018). | Lenguaje fuertemente tipado (Saray, 2015). Rápido, seguro y fiable (Gomez, 2018). Predominante e integral a la web moderna (Saray, 2015). | Necesita de una máquina virtual java para su ejecución (Villa Martínez, 2015). |
| PHP | Es un lenguaje de programación interpretado en el servidor (Saray, 2015). | Lenguaje multiplataforma (Gomez, 2018). | Lenguaje de tipado débil (Saray 2015). Seguro, robusto y estable (Gomez, 2018). Conocido por su velocidad (Gomez, 2018). | Falta de normas en la programación orientada a objetos (Saray, 2015). |
| PYTHON | Es una lenguaje de programación interpretado, orientado a objetos (Gomez, 2018). | Lenguaje multiplataforma (Gomez 2018). | Simple, versátil y rápido (Gomez 2018). Lenguaje con mayor facilidad de aprendizaje (Gomez, 2018) | No es recomendable en programas de múltiples hilos (Garcia, 2017). |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Como se puede observar en la **tabla 1-2**, cada uno de los lenguajes tienen sus fortalezas y debilidades, sin embargo, se debe tomar en cuenta los requisitos del sistema para seleccionar uno y de esa manera elegir el más conveniente, en este caso debido a la seguridad, paradigma, facilidad y rapidez que Java ofrece, lo hace óptimo para el desarrollo de este proyecto.

2.3.1.1 Antecedentes

Java es un lenguaje de programación orientado a objetos y concurrente, apareció en los años 90 cuando la empresa Sun Microsystems empezó a incursionar en la industria de la electrónica y desarrollo de programas para controlar dispositivos electrónicos (Villa Martínez, 2015).

Los programas desarrollados en Java presentan grandes ventajas frente a otros lenguajes de programación, la ejecución de programas en Java tiene varias posibilidades como la ejecución de aplicación, applet o servlet, además Java permite el desarrollo en arquitecturas como cliente servidor y n-capas, aunque otros lenguajes también permiten esto, Java incorpora en su propia API la posibilidad de ejecutar tareas en varios ordenadores simultáneamente repartiendo el trabajo (García de Jalón et al., 1999).

Actualmente los programadores ocupan Java en diferentes ambientes, todavía se insertan programas Java en las páginas web (applets). “La popularidad inicial de los applets ayudó a posicionar a Java como uno de los lenguajes de programación líderes en el mundo. A pesar de que los applets aún tienen un papel significativo en el éxito actual de Java, otros tipos de programas Java han venido a superarlos en términos de popularidad” (Dean y Dean, 2009).

2.3.1.2 Características

- **Java es neutral a la arquitectura:** El compilador se ejecuta en una máquina virtual conocida como JVM (Java Virtual Machine), en donde el intérprete ejecuta los códigos, es decir, es portable al nivel de código objeto (Villa Martínez, 2015).
- **Java es totalmente orientado a objetos:** Java soporta los mecanismos de la programación orientación a objetos tales como la encapsulación, herencia y polimorfismo (Villa Martínez, 2015).
- **Java es seguro:** Posee un chequeo de datos muy estricto, además el manejo de la memoria es sencillo, Cuando un objeto ya no se utiliza, el recolector de basura automáticamente libera dicho espacio (Villa Martínez, 2015).
- **Java es distribuido:** Posee librerías para programación con TCP/IP protocolos de internet (Villa Martínez, 2015).
- **Java es concurrente:** Posee librerías para hacer programas con múltiples hilos de ejecución simultáneamente (Villa Martínez, 2015).

2.3.1.3 Ambiente de desarrollo

Java está disponible en 3 distribuciones: la edición empresarial, standard y la micro, en donde el standard es la versión original de java, la micro está orientada a el desarrollo de programas para dispositivos de poca memoria y la empresarial que posee distintas librerías para el desarrollo de programas distribuidos. El ambiente de desarrollo está constituido por el compilador, el intérprete de la máquina virtual JRE (Java Run-time Environment), el manual en línea y otras herramientas, además de un programa para correr los Applets (Villa Martínez, 2015).

Para realizar la instalación es necesario descargar el JDK correspondiente para el sistema operativo en uso, disponible para Windows, Linux y Solaris.

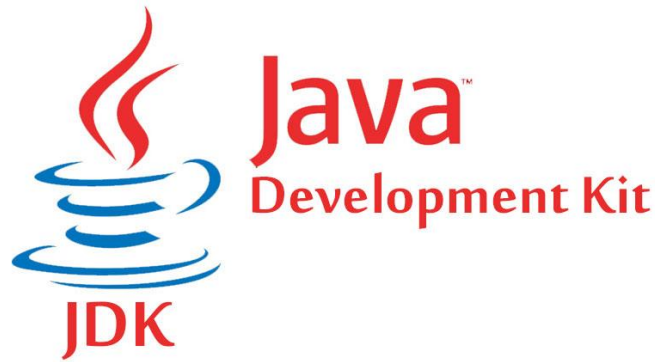


Figura 5-2: Java JDK

Fuente: <http://somebooks.es/instalar-jdk-windows-10/>

2.3.2 *Glassfish*

2.3.2.1 *Conceptos*

Glassfish es un servidor de aplicaciones diseñado para la plataforma de java EE, elaborado por la empresa Sun Microsystems, tiene soporte para las versiones más recientes de tecnologías tales como: JSP, JSF, Servlets, Ejbs, web services y XML.

Según (Solano, 2009) en su blog menciona “Glassfish además de ser un servidor de aplicaciones, es una comunidad de usuarios, que descargan y utilizan libremente”, a esto se le suma el equipo de ingenieros y testers que trabajan para solucionar cada fallo que se presente y la transparencia en la entrega de código fuente, datos de descarga y documentación.

2.3.2.2 *Características*

Glassfish posee una gran cantidad de características que lo hacen más confiable a la hora de elegir un servidor de aplicaciones, según (Maestre, 2016) “Glassfish es un servidor de aplicaciones desarrollado por Sun Microsystems para plataforma java EE” y posee características como se puede observar en la **tabla 2-2**.

Tabla 2-2: Características de Glassfish

| Características de Glassfish | |
|---|---|
| Balanceo de carga y colas de espera | Administración poco sencilla |
| Web services | Consume gran cantidad de recursos del CPU |
| Servidor web | Servidor de JEE5 |
| Distintos modos de inicio (normal, debug y profile) | |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Fuente: (Maestre, 2016).

2.3.3 Bootstrap 4

Bootstrap es un conocido framework para el diseño web basado en HTML, CSS y JS, es el más popular entre los desarrolladores de código abierto en todo el mundo, fue creado por el equipo de Twitter con el objetivo de ofrecer la mejor experiencia en cuanto a diseño de trata.

Bootstrap está preparado para funcionar en los navegadores de Chrome, Firefox, Opera, Safari e Internet Explorer a partir de la versión 8, Además de funcionar en PCs y portátiles con cualquier dimensión de pantalla también está disponible para tablets y smartphones de dimensiones más reducidas (Gallego, 2018).

El sistema de rejilla que Bootstrap utiliza permite visualizar el contenido de una web en diferentes dimensiones de pantalla, además de ofrecer variedad de funciones que optimizan el tiempo de diseño de una web responsiva.

En la **figura 6-2** se puede observar el sistema de rejilla de Bootstrap 4, su funcionamiento de acuerdo con la dimensión de la pantalla del dispositivo y como llamarlas con la ayuda de las clases CSS.

| Pantalla | Dimensiones | Prefijo de la clase | Ancho del contenedor |
|----------------------|-------------|---------------------|----------------------|
| Tamaño extra pequeño | < 576 px | .col- | Ninguno (automático) |
| Tamaño pequeño | ≥ 576 px | .col-sm- | 540px |
| Tamaño medio | ≥ 768 px | .col-md- | 720px |
| Tamaño grande | ≥ 992 px | .col-lg- | 960px |
| Tamaño extra grande | ≥ 1200 px | .col-xl- | 1140px |

Figura 6-2: Sistema de rejillas Bootstrap 4.

Fuente: (Gallego, 2018).

2.3.4 JavaScript

2.3.4.1 Antecedentes

JavaScript es un robusto lenguaje de programación que ayuda a los desarrolladores a realizar acciones dentro de sus páginas web, fue creado por Brendan Eich en el año de 1995 adoptando el nombre de LiveScript que posteriormente se lo conoció como JavaScript. Actualmente es considerado uno de los lenguajes de programación más importantes que se utiliza en casi todos los sitios web alrededor del mundo.

Existe una tendencia a confundir los lenguajes de Javascript con Java, pero estos son completamente distintos y tienen características propias. (Pérez, 2007) en su web nos menciona que JavaScript tiene la ventaja de ser incorporado en cualquier página web y ser ejecutado sin la necesidad de instalar otro programa para ser visualizado.

JavaScript actualmente es soportado por la mayor cantidad de navegadores entre ellos: Chrome, Mozilla, Opera, Netscape y Internet Explorer, aunque JavaScript sea soportado en gran cantidad de navegadores los usuarios pueden elegir la opción de Activar/Desactivar el JavaScript en los mismos (Pérez, 2007).

2.3.4.2 Característica, ventajas y limitaciones

JavaScript tiene su fuerte en el lado de frontend dando a la web una mayor interactividad, también se puede usar librerías y frameworks que ayudan a mejorar la experiencia de los sitios webs, en la **tabla 3-2** se mencionan algunas de sus características, ventajas y limitaciones.

Tabla 3-2: Características, ventajas y limitaciones.

| | Características | Ventajas | Limitaciones |
|-------------------|---|---|--|
| JavaScript | Lenguaje orientado a eventos. Débilmente tipado. Trabaja con prototipos. Es funcional. Imperativo y estructurado. | Contenido interactivo. Interacción con el DOM. Facilita las validaciones. Recomendable para animaciones. | Demora en la ejecución de los scripts. |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Fuente: (Mora, 2017).

2.3.5 PostgreSQL

2.3.5.1 Concepto

PostgreSQL es un potente gestor de base de datos relacionales orientadas a objetos, los cuales son empleados en grandes empresas o negocios con gran cantidad de información para almacenar, en donde es indispensable una base de datos. Según (Rubinos Carvajal y Nuevo León, 2011) “Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular”.

2.3.5.2 Antecedentes

El sistema gestor base de datos PostgreSQL tiene su origen en los años de 1975 en la universidad de california, a partir de 1978 se crearon varias extensiones logrando así, soportar hasta base de datos distribuidas, está derivado del paquete Postgres escrito en Berkeley. Es el gestor de bases de datos de código abierto más avanzado actualmente, ofreciendo control de concurrencia multi-versión, soportando la mayoría de la sintaxis SQL, contando también con varios enlaces con diferentes lenguajes de programación como son C, C++, Java, perl, tcl y python (Vélez et al., 2016).

2.3.5.3 Características, limitaciones, y ventajas

PostgreSQL es el gestor de base de datos más avanzado hoy en día, con una gran robustez y soporte por una amplia comunidad de desarrolladores, en general se caracteriza por ser un sistema estable con alto rendimiento y una buena flexibilidad, funciona bien con sistemas Unix y tiende a ser escalables (Lockhart, 2018).

Una de sus principales características, es la implementación de los famosos rollbacks, subconsultas y transacciones, logrando así obtener un rendimiento óptimo de lo que sus competidores podrían ofrecer, pero posee una gran limitación que se refiere a la gran cantidad de recursos que consume al realizar sus operaciones (Lockhart, 2018).

En la **tabla 4-2** podemos observar las ventajas y desventajas de PostgreSQL.

Tabla 4-2: Ventajas y desventajas de PostgreSQL.

| Ventajas | Desventajas |
|---|--|
| Alto nivel de escalabilidad | Consume gran cantidad de recursos |
| Gran capacidad de almacenamiento | Sintaxis poco intuitiva en algunos comandos |
| Posee herramientas gráficas para el diseño y la administración de base de datos | No cuenta con un soporte completo para la orientación a objetos. |
| Es multiplataforma | |
| Posee un buen sistema de seguridad | |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Fuente: (Lockhart, 2018).

2.3.5.4 PostgreSQL vs otros gestores

Según (Florencio, 2017) “A la hora de la realización un proyecto web, tenemos que tener muy claro que Sistema Gestor de Base de Datos elegir para optimizar dicho proyecto”.

En la actualidad existen gran variedad de gestores de bases cada una con sus características, seguidores y detractores, en la **tabla 5-2** se puede observar las principales características, ventajas e inconvenientes de los gestores de bases más populares en el desarrollo web.

Tabla 5-2: Comparación gestores de base de datos.

| Gestor | Características | Ventajas | Inconvenientes |
|-------------------|--|---|--|
| Access | Es un sistema de gestión de bases de datos Relacional creado por Microsoft (Iruela, 2016). Es muy gráfico, simple y directo para trabajar con la información (Florencio, 2017). | Accesible para personas con poco manejo de base de datos (Florencio, 2017). Las bases de datos pueden ser consultados por otros programas (Iruela, 2016). | No funciona para bases de datos grandes (Florencio, 2017). No es multiplataforma (Florencio, 2017). |
| MySQL | Pertenece a Oracle con licencia GPL (Florencio, 2017). Es un sistema de gestión de base de datos relacional, multihilo y multiusuario (Iruela, 2016). | Velocidad al realizar las operaciones (Iruela, 2016). Instalación sencilla (Florencio, 2017). | Capacidad limitada, sin soporte (Florencio, 2017). |
| PostgreSQL | Es un sistema de gestión de base de datos relacional orientada a objetos y libre (Iruela, 2016). Tiene la extensión POSTGIS para base de datos espaciales (Florencio, 2017). | Estabilidad y confiabilidad (Iruela, 2016). Código abierto y gratuito, multiplataforma (Florencio, 2017). Soporta transacciones, disparadores y afirmaciones (Florencio, 2017). | Poco intuitivo, respuesta lenta (Florencio, 2017). |
| Oracle | Es un sistema de gestión de base de datos relacional (Iruela 2016). Soporta base de datos de gran tamaño (Florencio, 2017). | Intuitiva y fácil de usar (Florencio 2017). Soporte para transacciones, estable y escalable (Iruela, 2016). | No es gratuito y tiene precios elevados (Florencio, 2017). |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Debido a la estabilidad y confiabilidad que PostgreSQL ofrece a sus usuarios y la ventaja de ser código abierto y gratuito se seleccionó como gestor de base de datos para el desarrollo del presente proyecto.

2.3.6 *NetBeans 8.2*

NetBeans es un exitoso proyecto de código abierto con un gran soporte constante de la comunidad de desarrolladores, es un entorno de desarrollo muy famoso entre los desarrolladores de java debido a sus características que nos permiten, escribir, compilar, depurar y ejecutar programas, está escrito en el lenguaje de programación java pero se lo puede utilizar para desarrollar en varios lenguajes de programación, una de sus ventajas es que es libre y gratuito sin ningún tipo de restricciones de uso.

NetBeans 8.2 da soporte completo para la última versión del lenguaje de programación web PHP, además de una configuración especial para proyectos y soporte para varias herramientas como Symfony y PHPunit, soporte para ECMAScript 6 y node.js, así como compatibilidad con los contenedores Docker (Velasco, 2016).

2.4 **Modelo, vista, controlador**

Según (Bahit, 2014) “MVC es un patrón de arquitectura de software encargado de separar la lógica de negocio de la interfaz del usuario y es el más utilizado en aplicaciones Web, ya que facilita la funcionalidad, mantenibilidad y escalabilidad del sistema, de forma simple y sencilla”.

MVC divide a las aplicaciones en tres capas conocidas como modelo, vista y el controlador cada una de ellas encargada de una función en específico.

- **Modelo:** se encarga de las operaciones con datos, trabajando con la base de datos, es decir: consultas, inserciones, actualizaciones etc.
- **Controlador:** recibe las peticiones del usuario y se encarga de solicitar los datos al modelo.
- **Vista:** es la encargada de dar la respuesta de forma visual a las peticiones del usuario.

En la **figura 7-2** se puede observar el funcionamiento del patrón MVC.

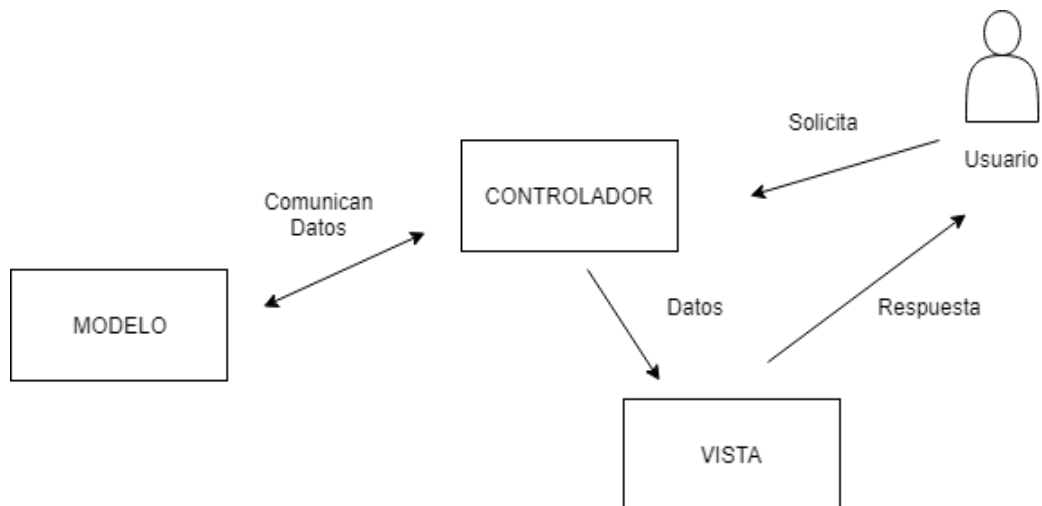


Figura 7-2: Patrón MVC.
Realizado por: Barragán Luis, Colcha Efraín. 2018

2.5 Metodología ágil Scrum

2.5.1 *Conceptos*

Según (Dimes, 2015) “Scrum es un marco de referencia para crear software complejo y entregarlo a tiempo de una forma mucho más sencilla”, tiene su aparición en los años 1990 y en la actualidad en una de las metodologías con más popularidad en el área de desarrollo de software.

(Schwaber y Sutherland, 2013) mencionan que scrum es “Un marco de trabajo por el cual las personas pueden acometer problemas complejos adaptativos, a la vez que entregar productos del máximo valor posible productiva y creativamente”, además que es ligero, fácil de entender y difícil de llegar a dominar.

2.5.2 Características

Scrum trabaja en ciclos iterativos llamados Sprints, los cuales tienen una duración máxima de hasta 4 semanas, estas iteraciones son asignadas por el equipo de trabajo y no se deben alargar jamás, los miembros del equipo asignan los requerimientos del software al product backlog y de ahí los organizan por sprints, cuando estos terminan se realiza la reunión de revisión en donde se habla del progreso del software. Según (Dimes, 2015) “Las iteraciones en scrum se asemejan a las iteraciones en Agile ya que estas iteraciones también requieren tener un prototipo listo para entregar”.

Scrum trabaja con una serie de roles, artefactos y eventos los cuales se pueden observar en la **figura 8-2**.

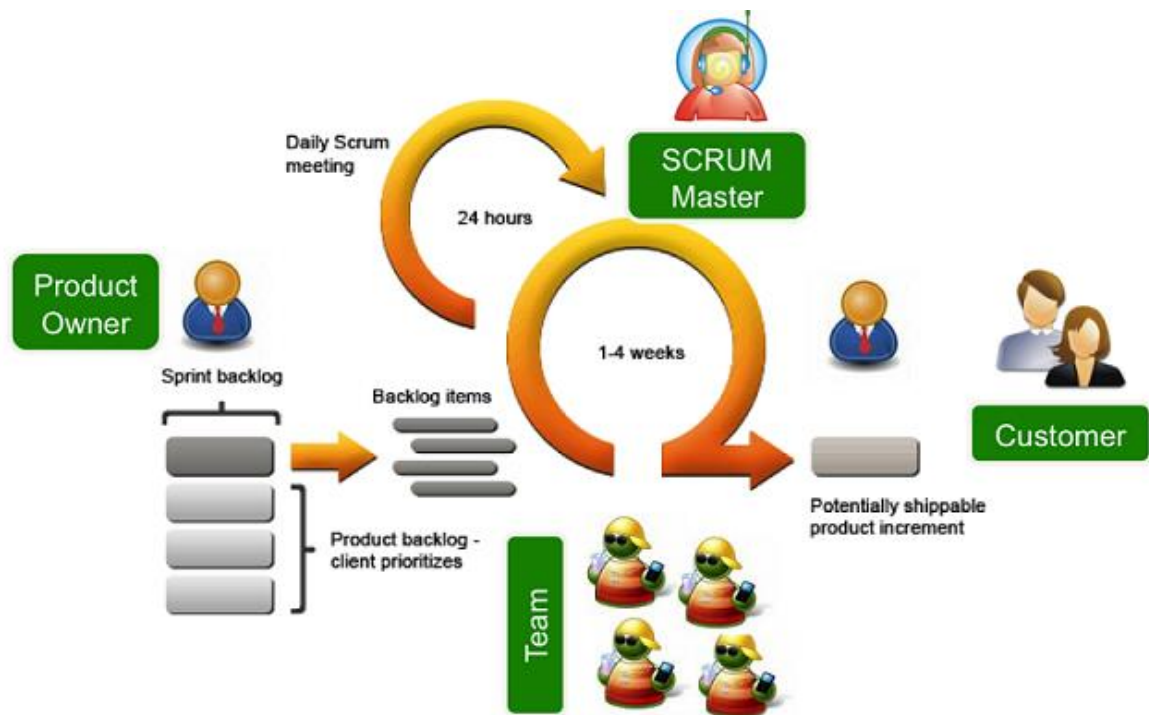


Figura 8-2: Proceso Scrum.

Fuente: <http://kulkansystems.com/blog/metodologia-agil-scrum-proyecto-de-fundamentacion-parte-iii/>

2.5.3 Roles

Dentro del equipo de trabajo Scrum cada uno de los miembros desempeña un rol con una función definida, los principales roles de scrum son:

Scrum master

Es la persona encargada de liderar al equipo de trabajo, dentro de sus funciones se encuentran: velar por el cumplimiento de las reglas y procesos de la metodología, gestionar riesgos del proyecto y garantizar el cumplimiento de las actividades.

Product owner

Es la persona encargada de velar por los intereses del cliente, dentro de sus funciones se encuentra el crear las historias de usuario, priorizarlas y asignarlas dentro del product backlog.

Team

Es el personal encargado de llevar a cabo las historias de usuario y de entregar el producto, por lo general son profesionales con conocimientos técnicos, suelen ser grupos de 3 a 8 personas.

Stakeholders

Son las personas que se benefician directamente con el desarrollo del software, dentro de sus funciones se encuentra la revisión de cada uno de los sprints.

2.5.4 *Ciclo de vida*

El ciclo de vida Scrum viene dado por 4 fases las cuales se detallan a continuación:

Fase de planificación

En esta fase se realiza el análisis de los requerimientos, así como la asignación de los roles de cada uno de los miembros del equipo, además se determina todas las tareas a realizar.

Fase de diseño

En esta fase se realizan tareas técnicas tales como, diseño de diagramas UML, base de datos estándares de codificación, diseño de interfaces y demás herramientas a utilizar.

Fase de desarrollo

En esta fase se realizan las tareas de cada uno de los sprints, generando así al final un producto entregable.

Fase de pruebas

En esta fase se realizan las pruebas necesarias para asegurar el correcto funcionamiento del producto, se lo realiza al final de cada sprint, pero también se puede hacer una revisión general al final del proyecto.

2.5.5 *Scrum vs otras metodologías ágiles*

Según (Montoya, Sepúlveda y Jiménez, 2017) “El desarrollo de software es un proceso riesgoso y difícil de controlar, y más si no se lleva una metodología para esta actividad, el resultado que se va a obtener son clientes insatisfechos, mala calidad del producto, exceso de tiempo y mal uso del presupuesto”. La finalidad de las metodologías ágiles es mejorar los resultados de los productos en desarrollo.

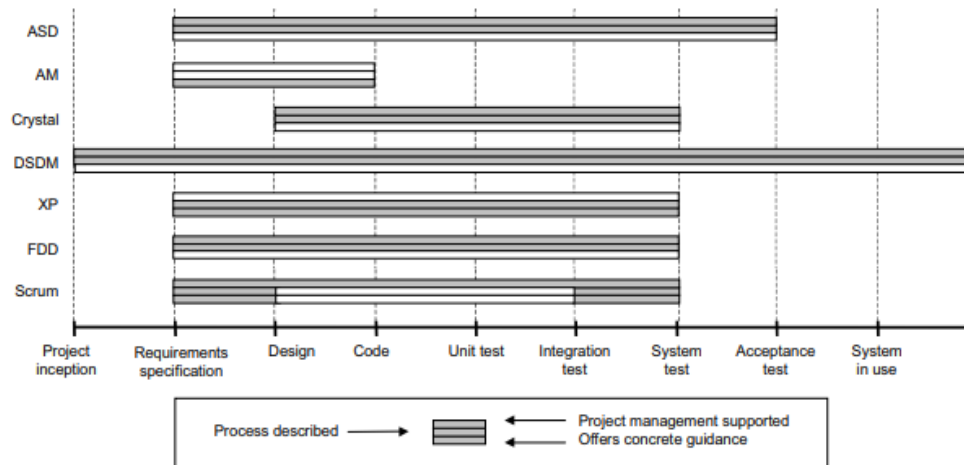


Figura 9-2: Comparativa metodologías ágiles.
Fuente: (Rodríguez, 2015).

En la **figura 9-2** se puede observar la comparativa de las metodologías ágiles más populares en donde se las compara por la gestión del proyecto, ciclo de vida y abstracción vs concreto, según (Rodríguez, 2015) DSDM ofrece soporte para gestión de proyectos y cumple con el ciclo de vida, AM solo ofrece guías concretas pero en general todas carecen de orientaciones concretas.

SCRUM sobresale por su notoriedad y se enfoca en las fases de desarrollo del proyecto asegurando la calidad y satisfacción del usuario en palabras de (Rodríguez, 2015) “Tal vez su éxito se debe a que si tiene en cuenta la gestión del proyecto y da direcciones concretas en dos fases cruciales del proyecto. ”

2.6 Estándar ISO/IEC 9126

Según (Molina et al., 2016) “La calidad del software son aquellas características que son propias del software, el desarrollo de este no se lo fabrica sino más bien de desarrolla a media como incrementa la productividad en el desarrollo de aplicaciones web”, para evaluar la calidad de software existen ciertas normas que respaldan y verifican la calidad de un producto software”.

El estándar ISO 9126 es una norma internacional para la evaluación de software que fue publicado en 1992, este estándar describe 6 características generales (Largo y Marin, 2009).



Figura 10-2: Características ISO 9126.

Fuente: (Largo y Marin, 2009).

2.6.1 Portabilidad

Se define a la portabilidad como la capacidad del software para ser trasladado de un entorno a otro, según (Molina et al., 2016) “, busca cumplir con la función de evaluar la facilidad de adaptación y reemplazo”. En la **tabla 6-2** se puede observar las sub-características de la portabilidad.

Tabla 6-2: Sub-características portabilidad.

| | |
|------------------------------------|--|
| Adaptabilidad | Adaptación del software en entornos especificados. |
| Facilidad de instalación | Facilidad de instalación del sistema para el usuario final. |
| Coexistencia | Capacidad de competir por recursos comunes con otro software. |
| Reemplazabilidad | Capacidad de ser reemplazo por software de las mismas características. |
| Conformidad de portabilidad | Cumplir con normas de portabilidad. |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Fuente: (Largo y Marin, 2009).

2.6.2 Eficiencia

Según (Largo y Marin, 2009) “La eficiencia del software es la forma del desempeño adecuado, de acuerdo al número recursos utilizados según las condiciones planteadas”, es realizar una evaluación del rendimiento del sistema en un tiempo dado. En la **tabla 7-2** se puede observar las sub-características de la eficiencia.

Tabla 7-2: Sub-características eficiencia.

| | |
|----------------------------------|---|
| Comportamiento de tiempos | Rendimiento del sistema cuando se ejecuta en circunstancias dada. |
| Utilización de recursos | Capacidad del software de consumir recursos. |
| Conformidad de eficiencia | Capacidad del software para cumplir con estándares de eficiencia. |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Fuente: (Largo y Marin, 2009).

CAPITULO III

3 MARCO METODOLÓGICO

Para el desarrollo del presente trabajo de titulación se ha optado por la utilización de una metodología ágil de desarrollo de software, se seleccionó la metodología ágil SCRUM debido a que esta nos permite dar seguimiento a las actividades que se realizarán dentro del proyecto, así como también el trabajo de cada uno de los integrantes del equipo, en donde los entregables se presentarán al final de cada Sprint y deberán cumplirse de acuerdo con su planificación.

3.1 Fase de planificación

En esta fase se realiza la planificación para el desarrollo del sistema, mediante la recolección de los requerimientos, los cuales se extraen en las reuniones y/o entrevistas con el Dr. Alonso Álvarez O. docente de la Facultad de Informática y Electrónica en la ESPOCH.

La planificación del proyecto se puede observar en la **figura 11-3**, tomando en cuenta que los tiempos para cada actividad pueden cambiar durante la realización del sistema.

| | Nombre de tarea | Comienzo | Fin |
|----|---------------------------------|-------------|-------------|
| 1 | FASE INVESTIGATIVA | lun 2/4/18 | lun 16/4/18 |
| 2 | Definición de actividades | lun 2/4/18 | vie 6/4/18 |
| 3 | Recopilación de información | sáb 7/4/18 | mar 10/4/18 |
| 4 | Clasificación de información | mié 11/4/18 | jue 12/4/18 |
| 5 | Elaboración del Documento | vie 13/4/18 | lun 16/4/18 |
| 6 | FASE APLICATIVA | mar 17/4/18 | dom 15/7/18 |
| 7 | Sprint 1 | mar 17/4/18 | mar 1/5/18 |
| 8 | Sprint 2 | mié 2/5/18 | mié 16/5/18 |
| 9 | Sprint 3 | jue 17/5/18 | jue 31/5/18 |
| 10 | Sprint 4 | vie 1/6/18 | vie 15/6/18 |
| 11 | Sprint 5 | sáb 16/6/18 | sáb 30/6/18 |
| 12 | Sprint 6 | dom 1/7/18 | dom 15/7/18 |
| 13 | FASE FINAL | lun 16/7/18 | lun 30/7/18 |
| 14 | Redacción del Documento Final. | lun 16/7/18 | mié 25/7/18 |
| 15 | Conclusiones y recomendaciones. | jue 26/7/18 | sáb 28/7/18 |
| 16 | Impresión de documentos. | dom 29/7/18 | dom 29/7/18 |
| 17 | Defensa de trabajo escrito. | lun 30/7/18 | lun 30/7/18 |

DIAGRAMA DE GANTT

Figura 11-3: Actividades planificadas.
Realizado por: Barragán Luis, Colcha Efraín. 2018

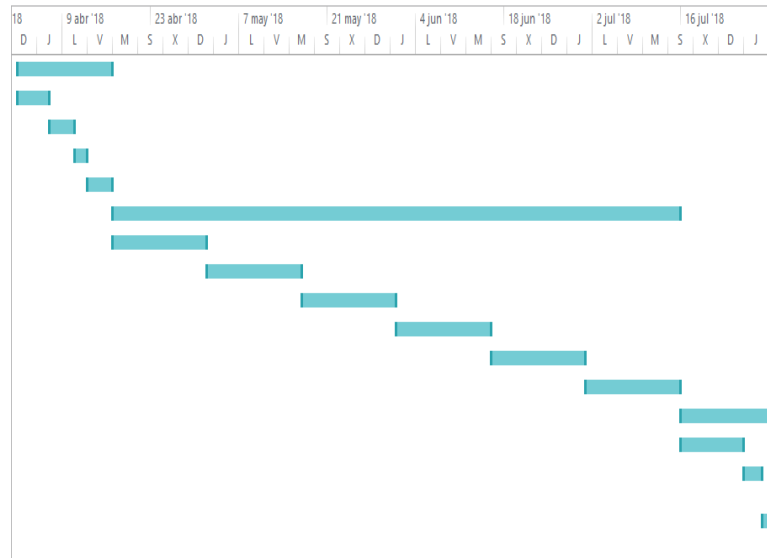


Figura 12-3: Diagrama Gantt.
Realizado por: Barragán Luis, Colcha Efraín. 2018

3.1.1 Requerimientos

Gracias a las reuniones que se realizaron se recolectaron los requerimientos para el desarrollo del sistema, para una mayor organización se lo divido en tres módulos los mismos que están definidos de la siguiente manera:

Módulo de diseño: Este módulo contendrá todas las acciones correspondientes al diseño del sistema, así como la preparación del área de trabajo.

Las tareas por realizar en este módulo son:

- Diseño y maquetación del sistema.
- Preparación del área de trabajo.

Módulo ondas en dos dimensiones: Este módulo contendrá las acciones correspondientes a las gráficas de ondas en dos dimensiones que puede realizar el usuario dentro sistema.

Las tareas por realizar en este módulo son:

- Identificación de condiciones iniciales.
- Análisis del modelo matemático del problema.
- Resolución analítica y numérica.

Módulo ondas en tres dimensiones: Este módulo contendrá las acciones correspondientes a las gráficas de ondas en tres dimensiones que puede realizar el usuario dentro sistema.

Las tareas por realizar en este módulo son:

- Identificación de condiciones iniciales.
- Análisis del modelo matemático del problema.
- Resolución analítica y numérica.

3.1.1.1 Requisitos funcionales

Como resultado de las reuniones se obtuvo un total de cuatro requerimientos los cuales se detallan a continuación:

- Graficar onda 2D caso finito.
- Graficar onda 3D caso finito.
- Graficar onda 2D caso infinito.
- Graficar onda 3D caso infinito.
- Generar histórico ondas 2D
- Generar histórico ondas 3D

3.1.1.2 Requisitos no funcionales

Con el objetivo de obtener un sistema de calidad se identificaron los requerimientos no funcionales, los cuales no describen información a gestionar, ni funciones a realizar, sino características de funcionamiento. Se identificaron cuatro requerimientos no funcionales, los cuales permitirán al sistema tener distintos atributos de calidad.

Los requerimientos no funcionales identificados para el desarrollo del sistema se detallan a continuación:

Mantenibilidad: El sistema podrá ser modificado de manera efectiva y eficiente frente a necesidades evolutivas.

Usabilidad: El sistema debe ser fácil de entender, aprender, manipular y debe ser atractivo para los usuarios.

Escalabilidad: El sistema debe estar preparado para adaptarse al crecimiento sin perder calidad en los servicios ofrecidos.

Portabilidad: El sistema debe ser capaz de ejecutarse en diferentes navegadores y sistemas operativos.

3.1.2 Roles del proyecto

Para el desarrollo del presente proyecto se identificaron los roles Scrum como se puede observar en la **tabla 8-3**.

Tabla 8-3: Roles del Proyecto.

| Rol | Nombre |
|---------------|---|
| Product Owner | Docente de la Facultad de Informática y Electrónica: Dr. Alonso Álvarez O. |
| Stakeholders | Estudiantes de la Facultad Informática y Electrónica de la ESPOCH. Docentes de la Facultad Informática y Electrónica de la ESPOCH. |
| ScrumMaster | Dr. Alonso Álvarez O. |
| Team | Luis Barragán G. Efraín Colcha C. |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.1.3 Product backlog

El Product backlog representa el conjunto de historias de usuario que conforman el desarrollo del sistema, se determinaron un total de 12 historias de usuario las cuales serán organizados en varios sprints, mismos que al final del desarrollo conformarán un producto entregable.

En la **tabla 9-3** se detallan todas las historias de usuario necesarios para el desarrollo del sistema.

Tabla 9-3: Product backlog.

| Product backlog | |
|------------------------|---|
| ID | Descripción |
| HT_01 | Como programador deseo obtener los requerimientos del sistema proporcionados por el usuario. |
| HT_02 | Como técnico deseo diseñar el modelo de la arquitectura del sistema. |
| HT_03 | Como programador deseo especificar un estándar de codificación. |
| HT_04 | Como programador deseo diseñar la base de datos para el sistema. |
| HT_05 | Cómo programador deseo crear las clases necesarias para correcto funcionamiento del sistema. |
| HT_06 | Cómo programador deseo obtener el diseño de la interfaz de usuario para el sistema. |
| HU_01 | Como estudiante/docente deseo graficar una onda en dos dimensiones aplicando el modelo matemático de D'alambert. |
| HU_04 | Como estudiante/docente deseo graficar una onda en tres dimensiones aplicando el modelo matemático de D'alambert. |
| HU_02 | Como estudiante/docente deseo graficar una onda en dos dimensiones aplicando el modelo matemático de Fourier. |
| HU_05 | Como estudiante/docente deseo graficar una onda en tres dimensiones aplicando el modelo matemático de Fourier. |
| HU_03 | Como estudiante/docente deseo observar el histórico de las ondas en dos dimensiones. |
| HU_06 | Como estudiante/docente deseo observar el histórico de las ondas en tres dimensiones. |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.1.4 Sprint backlog

El Sprint Backlog representa el conjunto de actividades que se deben realizar en cada interacción o sprint, se determinaron un total de 4 sprints como se puede observar en la **tabla 10-3**.

Tabla 10-3: Sprint backlog.

| Sprint | Descripción |
|--|--|
| Sprint 0: acciones preliminares | Corresponde a las actividades de planificación del sistema. |
| Sprint 1: acciones de configuración área de trabajo | Corresponde a las tareas de configuración y preparación del área de trabajo. |
| Sprint 2: acciones de ondas en dos dimensiones | Corresponde al manejo de toda acción que genere ondas en dos dimensiones. |
| Sprint 3: acciones de ondas caso en tres dimensiones | Corresponde al manejo de toda acción que genere ondas en tres dimensiones |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.1.5 Gestión de riesgos

Para el análisis y gestión de riesgos se tomó en cuenta todos los posibles problemas que puedan ocurrir durante el desarrollo del proyecto, se la realizó en tres etapas: la identificación de riesgos, el análisis de riesgos y el plan de reducción, supervisión y gestión de riesgos.

3.1.5.1 Identificación de riesgos

Se identificaron un total de seis riesgos, cuatro son de categoría del proyecto, uno técnico y uno del negocio, en la **tabla 11-3** se puede observar la descripción, categoría y las consecuencias de cada uno de los riesgos identificados.

Tabla 11-3: Identificación de riesgos.

| Identificación | Descripción | Categoría | Consecuencias |
|-----------------------|--|------------------|--|
| R1 | Cambio de requisitos | Del Proyecto | Retraso del proyecto, asignación de tareas extras. |
| R2 | Perdida involuntaria de la información sobre el proyecto. | Del Proyecto | Retraso del proyecto, asignación de tareas extras. |
| R3 | Mal diseño de la base de datos. | Técnico | Redundancia de datos |
| R4 | Problemas de comunicación en el equipo. | Del proyecto | Retraso en el proyecto. |
| R5 | Mala planificación del proyecto | Del proyecto | Retraso del proyecto, replanificación |
| R6 | Presupuesto insuficiente para la realización del proyecto. | Del negocio | Pérdida de ingresos económicos del equipo. |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.1.5.2 Análisis de riesgos

En la **tabla 12-3** se puede observar los diferentes resultados obtenidos después del respectivo análisis de riesgos.

Tabla 12-3: Resultados análisis de riesgos.

| Identificación | Descripción | Probabilidad | Impacto | Exposición | Prioridad |
|-----------------------|--|---------------------|----------------|-------------------|------------------|
| R1 | Cambio de requisitos | Baja | Moderado | Media | 2 |
| R2 | Perdida involuntaria de la información sobre el proyecto. | Baja | Moderado | Media | 2 |
| R3 | Mal diseño de la base de datos. | Media | Alto | Alta | 1 |
| R4 | Problemas de comunicación en el equipo. | Media | Bajo | Baja | 3 |
| R5 | Mala planificación del proyecto. | Baja | Crítico | Media | 2 |
| R6 | Presupuesto insuficiente para la realización del proyecto. | Media | Bajo | Baja | 3 |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.1.5.3 Plan de reducción, supervisión y gestión del riesgo

Para la adecuada gestión de los riesgos se desarrollaron las denominadas hojas de gestión de riesgos una por cada riesgo identificado en la etapa de análisis.

En la **tabla 13-3** se puede observar la hoja de gestión del riesgo R1, las demás se las puede encontrar en el **ANEXO A**.

Tabla 13-3: Hoja de gestión del riesgo R1.

| HOJA DE GESTIÓN DEL RIESGO | | | |
|--|--------------------------|--|---------------------|
| Identificación: R1 | | Fecha: 23/04/2018 | |
| Probabilidad: Baja | Impacto: Moderado | Exposición: Media | Prioridad: 2 |
| Descripción: Cambio de requisitos. | | | |
| Refinamiento: | | Efectos: | |
| Causas: <ul style="list-style-type: none"> • Confusión entre deseo y necesidad del cliente. • Mala interpretación de los requisitos. • Nuevas necesidades del cliente. | | <ul style="list-style-type: none"> • Retraso de la ejecución del proyecto. • Aumento o disminución de requerimientos | |
| Reducción: | | | |
| <ul style="list-style-type: none"> • Establecer correctamente las necesidades del cliente. | | | |
| Supervisión: | | | |
| <ul style="list-style-type: none"> • Verificar el cumplimiento de la planificación de actividades. • Supervisar que los nuevos requerimientos, no afecten a otros ya realizados. | | | |
| Gestión: | | | |
| <ul style="list-style-type: none"> • Replanificación del proyecto. • Negociación de los nuevos requerimientos. • Adaptarse al cambio de los nuevos requerimientos. | | | |
| Estado Actual: | | | |
| Fase de Reducción iniciada: <input checked="" type="checkbox"/> | | | |
| Fase de Supervisión iniciada: <input type="checkbox"/> | | | |
| Gestionando el riesgo: <input type="checkbox"/> | | | |
| Responsables: | | | |
| Luis Barragán, Efraín Colcha | | | |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.1.5.4 Análisis económico

Para el desarrollo e implementación del presente trabajo de titulación no se solicitó de ningún tipo de gastos de implementación ni gastos de licencia ya que las herramientas a utilizar no poseen costo de licencia y la ESPOCH dispone de hardware necesario para la puesta en funcionamiento de la aplicación.

En la **tabla 14-3** se puede observar el detalle del presupuesto del proyecto.

Tabla 14-3: Presupuesto del proyecto.

| Ítems | Cantidad | Valor Unitario | Total |
|---|-----------------|-----------------------|----------------|
| Software | | | |
| Microsoft Office 2016 | 1 | 120,00 | 120,00 |
| Hardware | | | |
| Laptop Dell Inspiron con disco duro de 500GB y RAM DE 8GB | 1 | 700,00 | 700,00 |
| Laptop Toshiba con disco duro de 1tB y RAM de 12GB | 1 | 1100,00 | 1100,00 |
| Impresora EPSON L365 | 1 | 360,00 | 360,00 |
| Otros | | | |
| Artículos de oficina | varios | 50,00 | 50,00 |
| Tinta de impresora | 2 | 30,00 | 60,00 |
| Resma hojas A4 | 1 | 05,00 | 05,00 |
| Internet | 6 meses | 35,00 | 210,00 |
| Transporte y Viáticos | 6 meses | 100,00 | 600,00 |
| Memorias USB | 1 | 12,00 | 12,00 |
| Total | | | 3217,00 |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.2 Fase de diseño

En esta fase se definen actividades previas antes de iniciar con el desarrollo, por ejemplo: determinación de la arquitectura del sistema, los estándares de diseño, codificación, el diseño de la base de datos y los diagrama UML.

3.2.1 Diagrama de casos de uso

Una vez identificados los requisitos del sistema se diseñó el diagrama de casos de uso en donde se muestra las funcionalidades del sistema, se determinó un total de 6 casos, en la **figura 13-3** se muestra los casos de uso para el desarrollo del presente proyecto.

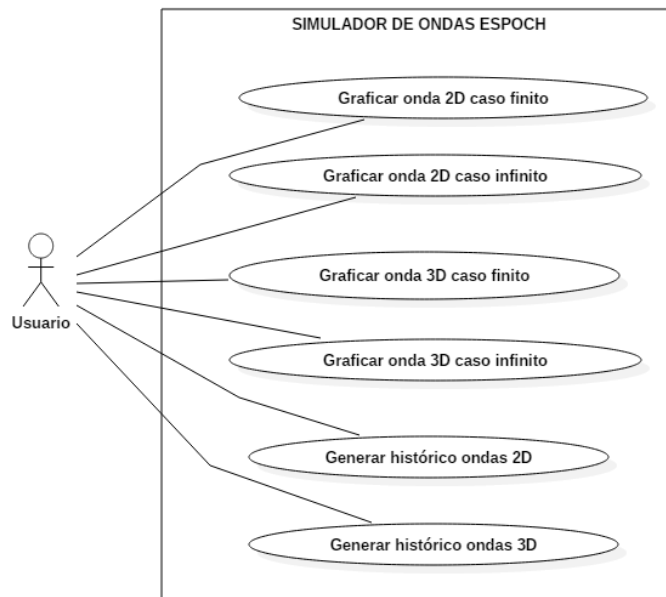


Figura 13-3: Diagrama casos de uso.
Realizado por: Barragán Luis, Colcha Efraín. 2018

En la **tabla 15-3** se puede observar la documentación del caso de uso “Graficar onda 2D caso finito” las demás se pueden encontrar en el **ANEXO B**.

Tabla 15-3: Caso de uso - Graficar onda 2D caso finito.

| Casos de Uso | | Graficar ondas 2D caso finito | |
|-------------------------|---|--|--|
| Descripción | El sistema permitirá al usuario graficar una onda en 2D utilizando el caso finito Fourier | | |
| Precondición | El usuario debe tener los datos necesarios para poder graficar la onda 2D | | |
| Secuencia Normal | Paso | Acción | |
| | 1 | El usuario debe ingresar al sistema para graficar una onda | |
| | 2 | El usuario debe escoger la opción Fourier caso finito | |
| | 3 | EL usuario debe ingresar los datos de coeficiente de elasticidad, velocidad inicial y posición inicial y seleccionar 2D. | |
| | 4 | El sistema utiliza los parámetros ingresado para generar la gráfica de la onda 2D | |
| Post Condición | El usuario podrá visualizar la simulación de la onda 2D generada. | | |
| Excepciones | Paso | Acción | |
| | 3 | Si el sistema detecta un valor no permitido se procederá a solicitar un nuevo valor | |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.2.2 Diagrama de clases

Se identificaron un total de 8 clases como se puede apreciar en la **figura 14-3**.

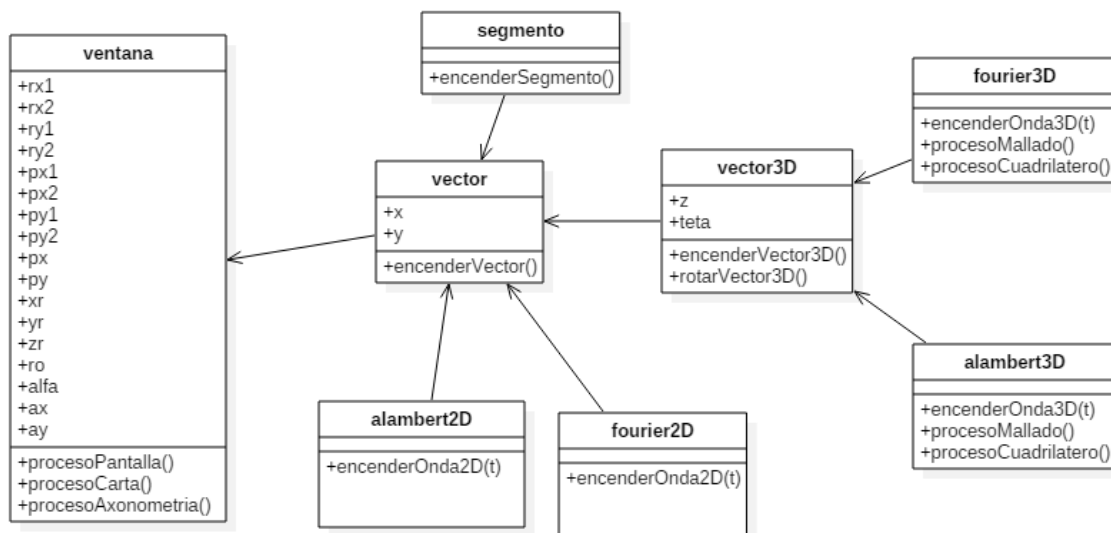


Figura 14-3: Diagrama de clases.

Realizado por: Barragán Luis, Colcha Efraín. 2018

Una vez obtenido el diagrama de clases se diseñó el diagrama de objetos como se puede observar en la **figura 15-3**.

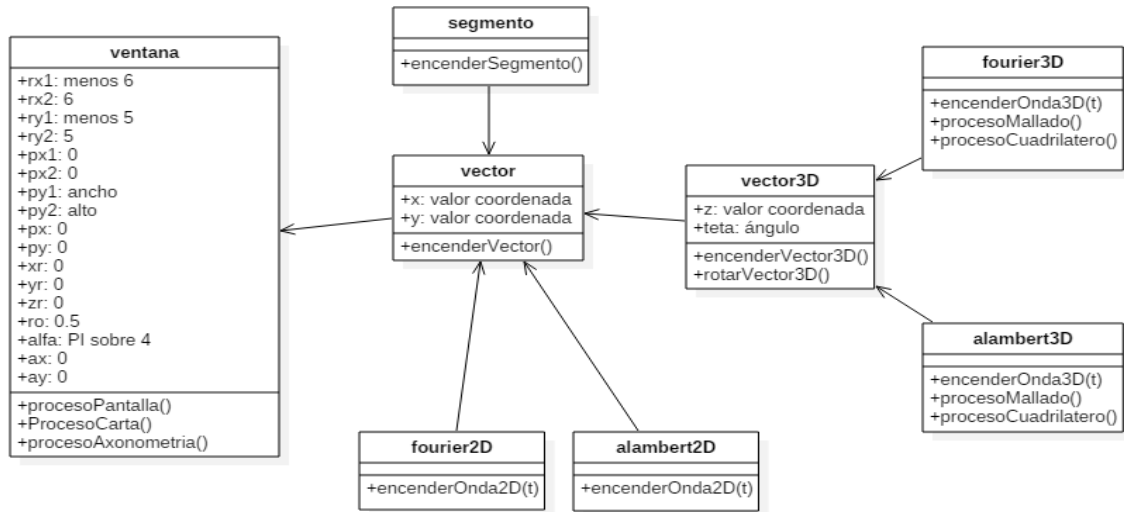


Figura 15-3: Diagrama de objetos.
 Realizado por: Barragán Luis, Colcha Efraín. 2018

3.2.3 Diagrama de secuencia

Se generaron un total de 6 diagramas de secuencia uno por cada caso de uso. En la **figura 16-3** se puede observar el diagrama de secuencia del caso de uso “Graficar onda 2D caso finito”, los demás diagramas se pueden encontrar en el **ANEXO C**.

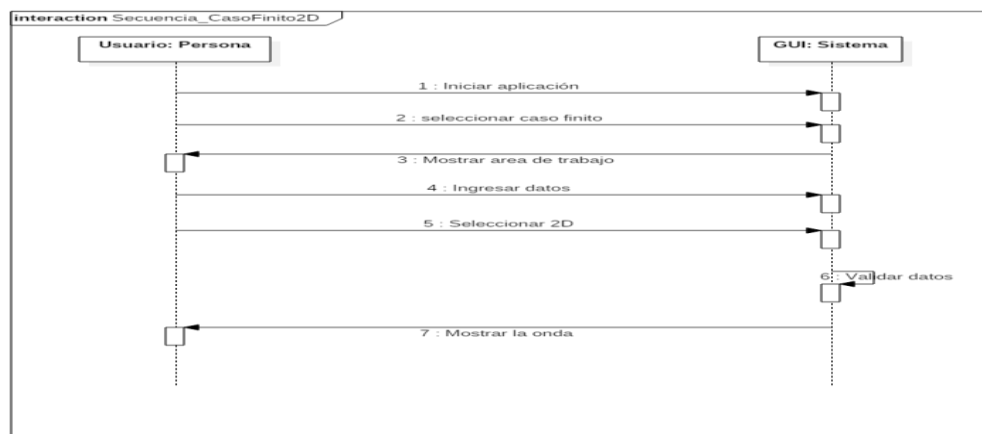


Figura 16-3: Diagrama de secuencia - Graficar onda 2D caso finito.
 Realizado por: Barragán Luis, Colcha Efraín. 2018

Partiendo de los diagramas de secuencia se diseñaron los diagramas de colaboración, se puede observar en la **figura 17-3** el diagrama de colaboración del caso de uso “Graficar onda 2D caso finito”, los demás diagramas se pueden encontrar en el **ANEXO D**.

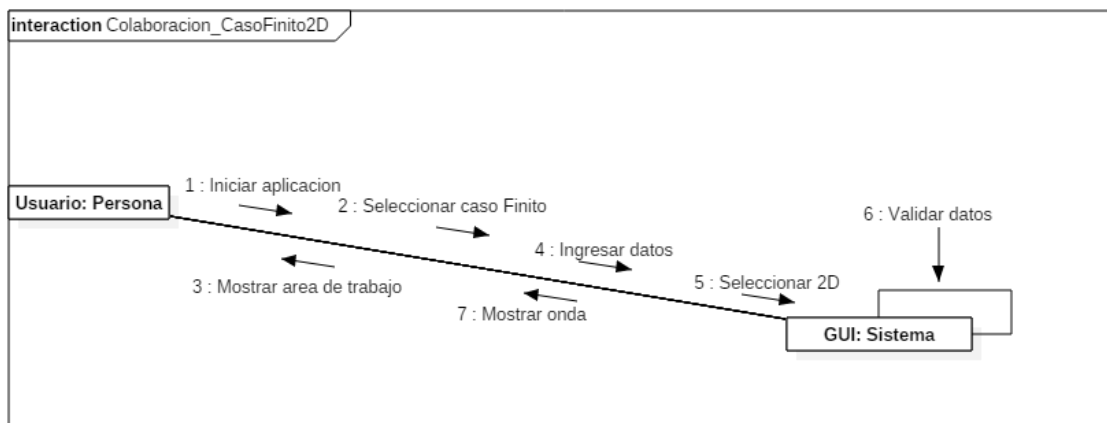


Figura 17-3: Diagrama de colaboración – Graficar onda 2D caso finito.
Realizado por: Barragán Luis, Colcha Efraín. 2018

3.2.4 Diagrama de estados

A partir del diagrama de objetos se diseñaron los diagramas de estados, en la **figura 18-3** se puede observar los diferentes estados que presenta el grafico de una onda.

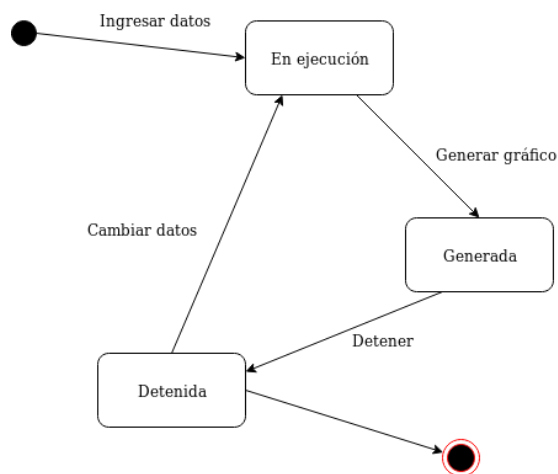


Figura 18-3: Diagrama de estados.
Realizado por: Barragán Luis, Colcha Efraín. 2018

3.2.5 Diagrama de actividad

Se diseñaron un total de 6 diagramas de actividad correspondientes a cada uno de los casos de uso identificados, en la **figura 19-3** podemos observar el diagrama de actividad del caso de uso “Graficar onda 2D caso finito”, los demás se pueden encontrar en los **ANEXO E**.

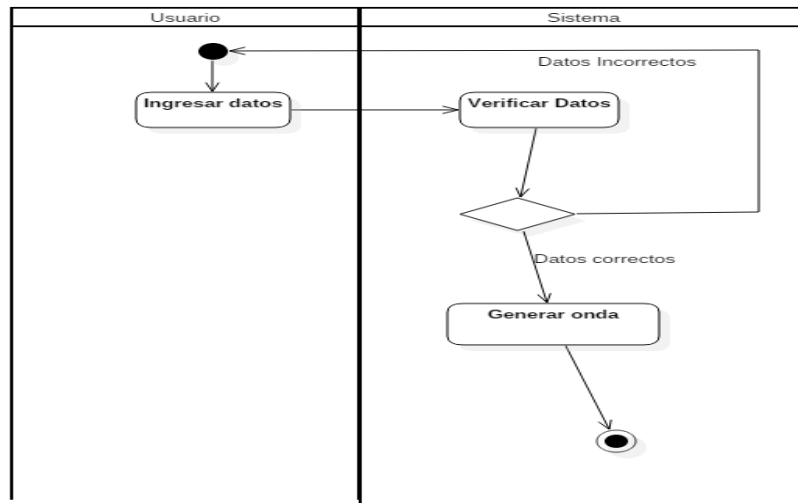


Figura 19-3: Diagrama de actividad - Graficar onda 2D caso finito.
Realizado por: Barragán Luis, Colcha Efraín. 2018

3.2.6 Arquitectura del sistema

Para el desarrollo del presente proyecto se optó por utilizar una arquitectura MVC, la cual separa los datos de la aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos, las cuales se detallan a continuación:

Modelo: es la encargada de manejar la lógica de negocio dentro del sistema, así como sus mecanismos de persistencia, trabaja directamente con la base de datos.

Controlador: En la capa media entre el modelo y la vista, encargada de recibir las peticiones del usuario y comunicarlas al modelo para realizar las operaciones necesarias y generar una respuesta.

Vista: en la encargada de interactuar con el usuario y dar las resulta a sus peticiones de forma visual.

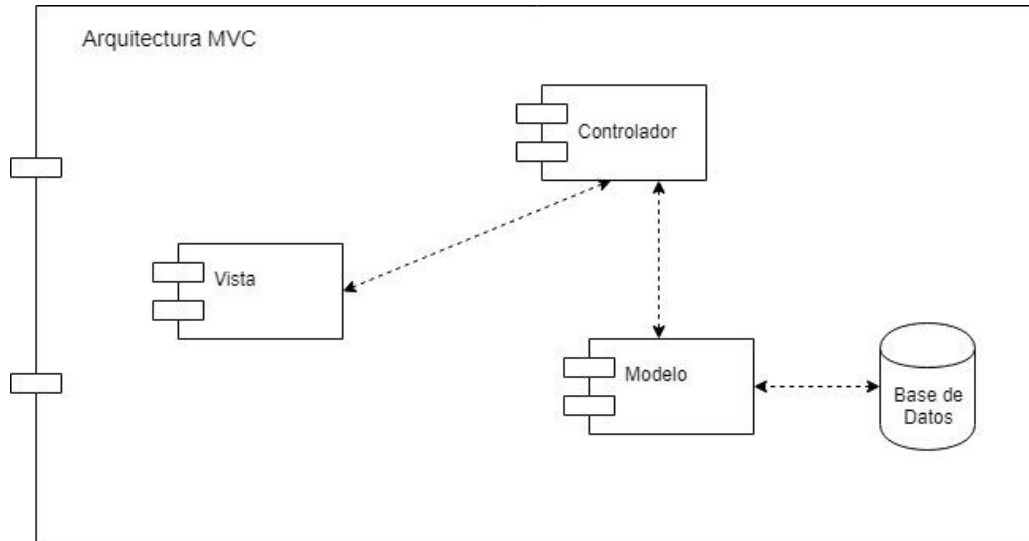


Figura 20-3: Arquitectura del sistema.
Realizado por: Barragán Luis, Colcha Efraín. 2018

3.2.7 Diagrama de despliegue

A partir del diagrama de componentes se diseñó el diagrama de despliegue que consta de dos servidores: el servidor web en los que se implementarán: lógica del negocio, acceso a datos y el servidor de base de datos, como se puede observar en la **figura 21-3**.

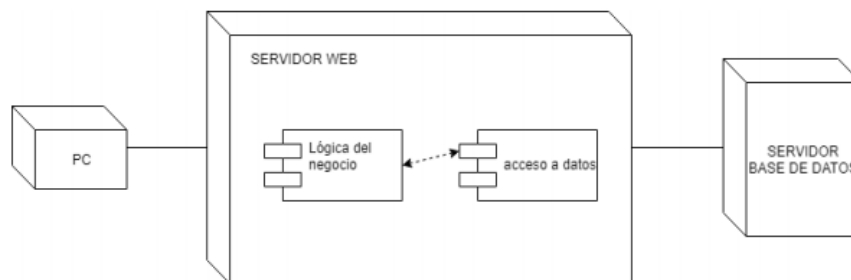


Figura 21-3: Diagrama de despliegue.
Realizado por: Barragán Luis, Colcha Efraín. 2018

3.2.8 *Estándar de codificación*

El estándar de codificación que se seleccionó para el desarrollo del presente trabajo de titulación fue el estándar lowerCamelCase, el cual permite un correcto manejo de la escritura de código y proporciona buenas prácticas de programación, los detalles se pueden observar en el **ANEXO G**.

3.2.9 *Diseño de la interfaz de usuario*

Para el desarrollo del simulador de fenómenos ondulatorios se realizó el diseño de la interfaz de usuario utilizando los colores característicos de la facultad de informática y electrónica de la ESPOCH y elementos adecuados para una mejor experiencia del usuario. El estándar de interfaz se puede observar en el **ANEXO H**.

3.2.10 *Diseño de la base de datos*

Para el desarrollo del simulador de fenómenos ondulatorios se realizó un análisis previo de las funcionalidades para poder determinar las entidades y relaciones necesarias y así garantizar el correcto almacenamiento de la información. Los diagramas de la base de datos se puede observar en el **ANEXO I**.

3.3 Fase de desarrollo

Se realizaron un total de 4 sprints, en donde el primero se refiere a actividades preliminares y las demás a tareas para desarrollar el sistema. Se obtuvieron un total de 12 historias de usuario.

3.3.1 Sprints

En este punto se ha asignado las historias de usuario a los diferentes sprints, en la **tabla 16-3** se puede observar las historias de usuario correspondiente a las acciones de onda en dos dimensiones los cuales están agrupados en el sprint 2, las demás historias de usuarios se encontrarán en el **ANEXO F**.

Tabla 16-3: Sprint2 - Acciones de onda en dos dimensiones.

| Sprint 2 – Acciones de onda en dos dimensiones | |
|---|--|
| ID | Descripción |
| HU_01 | Como estudiante/docente deseo graficar una onda en dos dimensiones aplicando el modelo matemático de D’Alambert. |
| HU_02 | Como estudiante/docente deseo graficar una onda en dos dimensiones aplicando el modelo matemático de Fourier. |
| HU_03 | Como estudiante/docente deseo observar el histórico de las ondas en dos dimensiones. |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.3.2 Historias de usuarios

Para la descripción de los requisitos se utilizó la herramienta historias de usuario las cuales sirven como guía del cómo se realizó las actividades en cada requerimiento. En la **tabla 17-3** se puede observar la descripción de la historia de usuario generar histórico ondas 2D, las demás se encuentran dentro del manual técnico adjunto en el **ANEXO J**.

Tabla 17-3: Historia de usuario – Generar histórico ondas 2D.

| HISTORIA DE USUARIO | |
|---|--|
| Número: HU_03 | Nombre de la Historia: Generar histórico ondas 2D |
| Usuario: Programador | Sprint Asignado: 2 |
| Puntos Estimados: 40 | Puntos Reales: 40 |
| Descripción: Como estudiante/docente deseo observar el histórico de las ondas en dos dimensiones para tener los registros de las ondas graficadas | |
| Pruebas de Aceptación: <ul style="list-style-type: none"> • Generar una onda en 2D y verificar el histórico • Verificar el cumplimiento del estándar de programación | |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.3.3 Tablero de tareas

Para una mejor comprensión de las historias de usuario se utilizó las tablas de tareas de ingeniería las cuales tienen el detalle de cada tarea realizada. En la **tabla 18-3** se puede observar la descripción de la tarea de ingeniería perteneciente a la historia de usuario generar histórico ondas 2D, las demás se encuentran dentro del manual técnico adjunto en el **ANEXO J**.

Tabla 18-3: Tarea de ingeniería – Generar histórico ondas 2D.

| PRUEBA DE ACEPTACIÓN | |
|--|--|
| Código: HU_03 PA_03 | Nombre de la Historia: Generar histórico ondas 2D |
| Nombre de la Prueba: Verificar el cumplimiento del estándar de programación | |
| Responsable: Alonso Álvarez | Fecha: 20-07-2018 |
| Descripción: Se verificará el cumplimiento del estándar de programación | |
| Condiciones de Ejecución: <ul style="list-style-type: none"> • El estándar de programación debe estar establecido | |
| Pasos de Ejecución: <ul style="list-style-type: none"> • Ingresar al sistema • Abrir archivos generados • Verificar estándar | |
| Resultado Esperado: Se cumple con el estándar de programación | |
| Evaluación de la Prueba: Exitosa | |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.3.4 Pruebas de aceptación

Para la validación y verificación se utilizó las tablas de pruebas de aceptación las cuales tienen como objetivo determinar el éxito o fracaso de las tareas realizadas. En la **tabla 19-3** se puede observar la descripción de la prueba de aceptación perteneciente a la historia de usuario generar histórico ondas 2D, las demás se encuentran dentro del manual técnico adjunto en el **ANEXO J**.

Tabla 19-3: Prueba de aceptación – Generar histórico ondas 2D.

| | |
|---|--|
| PRUEBA DE ACEPTACIÓN | |
| Código: HU_03 PA_01 | Nombre de la Historia: Generar histórico ondas 2D |
| Nombre de la Prueba: Generar una onda en 2D y verificar el histórico | |
| Responsable: Alonso Álvarez | Fecha: 20-07-2018 |
| Descripción: Se verificará que se grafique la onda en 2D y se genere el histórico | |
| Condiciones de Ejecución: <ul style="list-style-type: none">• El acceso datos, script, modelo, vista y controlador deben estar terminados | |
| Pasos de Ejecución: <ul style="list-style-type: none">• Ingresar al sistema• Acceder a histórico• Seleccionar 2D• Seleccionar un caso de onda• Verificar histórico | |
| Resultado Esperado: El histórico funciona correctamente | |
| Evaluación de la Prueba: Exitosa | |

Realizado por: Barragán Luis, Colcha Efraín. 2018

3.4 Fase de finalización

En esta fase se describen las actividades realizadas para la finalización del desarrollo del sistema, como son el manual técnico, manual de usuario, instalación y despliegue y el grafico BurnDown Chart.

3.4.1 *Manual técnico*

En el manual técnico se encuentra toda la información técnica acerca del desarrollo del sistema web, los detalles se los puede encontrar en el **ANEXO J**.

3.4.2 *Manual de usuario*

El manual de usuario está orientado para las personas que van a dar uso al sistema, está realizado con el objetivo de brindar apoyo al momento de utilizar el sistema, los detalles se los puede encontrar en el **ANEXO K**.

3.4.3 *Implantación*

Para la implantación se alquiló un servidor web, en donde se asignó un dominio destinado exclusivamente para el sistema, dentro de dicho servidor se subió el proyecto completo desarrollado en java con servidor Glassfish versión 4.1, java jdk 1.81 y su respectiva base de datos en el gestor PostgreSQL 10.4.

3.4.4 Gráfico burndown chart

En la **figura 22-3** se puede observar que el proceso de desarrollo del sistema se ha llevado a cabo con total normalidad existiendo leves variaciones en el tiempo estimado con el real.

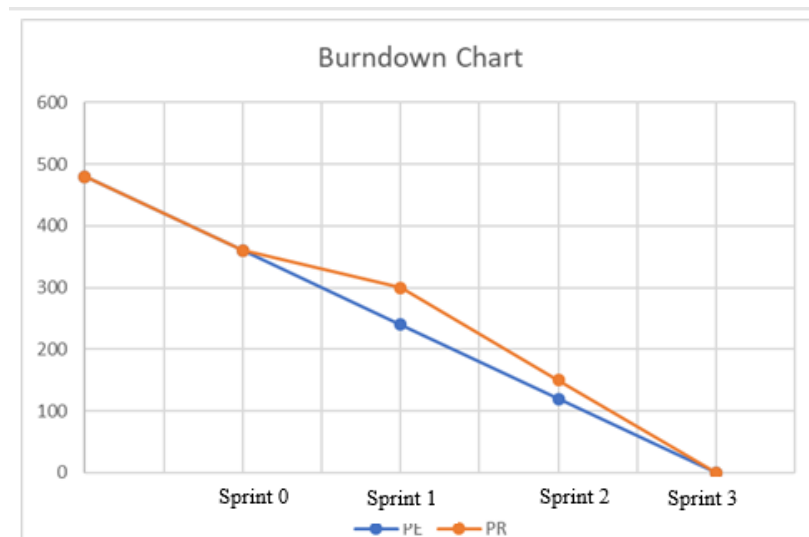


Figura 22-3: Gráfico burndown chart
Realizado por: Barragán Luis, Colcha Efraín. 2018

CAPITULO IV

4 MARCO DE RESULTADOS

En este capítulo se analizará los resultados que se obtuvieron de acuerdo con la evaluación de la portabilidad y eficiencia del SISTEMA WEB PARA SIMULAR LOS FENÓMENOS ONDULATORIOS DE D'ALAMBERT Y FOURIER EN 2D Y 3D APLICANDO MODELOS MATEMÁTICOS usando el estándar ISO 9126 descrito en el capítulo II, a continuación, se presenta el detalle de los aspectos de la evaluación.

4.1 Generalidades

Según la norma ISO/IEC 9126, para la evaluación de la eficiencia y portabilidad del sistema, se debe determinar indicadores para identificar el cumplimiento de la calidad como se puede observar en la **tabla 20-4**.

Tabla 20-4: Parámetros e indicadores.

| Parámetros | Indicador | Descripción |
|--------------|---------------------------|---|
| Portabilidad | Adaptabilidad | ¿El software se puede trasladar a otros ambientes? |
| | Capacidad de instalación | ¿El software se puede instalar fácilmente? |
| | Conformidad | ¿El sistema puede seguir funcionando si se hacen cambios? |
| Eficiencia | Comportamiento del tiempo | ¿Qué tan rápido responde el sistema? |
| | Utilización de recursos | ¿El sistema utiliza los recursos de manera eficiente? |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Además, se establece un criterio de evaluación para expresar los resultados analizados de una manera cuantitativa y cualitativa, como se puede observar en la tabla **21-4**.

Tabla 21-4: Criterios de evaluación.

| Calificación porcentual | Valor cuantitativo | Valor cualitativo |
|-------------------------|--------------------|-------------------|
| 0% – 35% | 1 | Baja |
| 36% - 70% | 2 | Media |
| 71% - 100% | 3 | Alta |

Realizado por: Barragán Luis, Colcha Efraín. 2018

4.2 Portabilidad

Para la evaluación de la portabilidad se optó por estudiar el comportamiento del sistema en dos casos de estudios, como se puede observar en la **tabla 22-4**.

Tabla 22-4: Casos de estudio.

| Características | Caso 1 | Caso 2 |
|-------------------|------------------------|------------------------|
| Sistema Operativo | Windows Home 10 | Fedora Workstation 28 |
| Memoria RAM | 8 GB | 8 GB |
| Disco duro | 500 GB | 500 GB |
| Procesador | Intel Core I5, 250 GHz | Intel Core I5, 250 GHz |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Como referencia se tomó la funcionalidad “Graficar onda 3D caso finito” en la cual se midió el comportamiento bajo los indicadores de la portabilidad (adaptabilidad, capacidad de instalación, conformidad) los resultados se los puede observar en la **tabla 23-4**.

Tabla 23-4: Resultados portabilidad.

| Indicadores | Caso 1 | | Caso 2 | |
|--------------------------|---|--------------------|---|--------------------|
| | Resultado | Valor cuantitativo | Resultado | Valor cuantitativo |
| Adaptabilidad | Se pudo acceder al sistema con normalidad | 3 | Se pudo acceder al sistema con normalidad | 3 |
| Capacidad de instalación | Se ingresa mediante la dirección web | 3 | Se ingresa mediante la dirección web | 3 |
| Conformidad | El sistema funciona con normalidad | 3 | El sistema funciona con normalidad | 3 |

Realizado por: Barragán Luis, Colcha Efraín. 2018

De acuerdo con el análisis realizado para la portabilidad se obtuvo un total de 3 puntos para cada uno de los indicadores en ambos casos de estudio dando un valor cualitativo de alta como se puede observar en el **gráfico 1-4**.

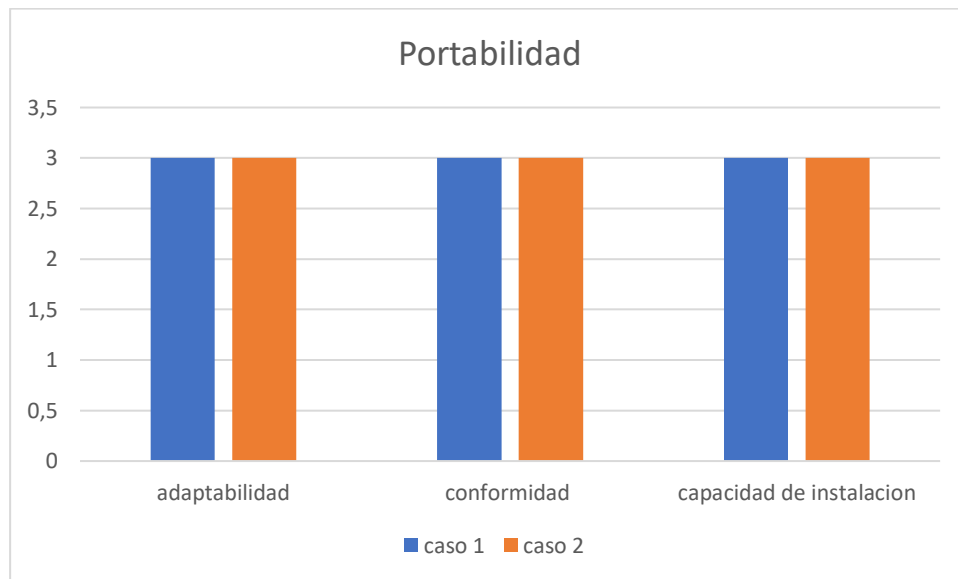


Gráfico 1-4: Resultados portabilidad.

Realizado por: Barragán Luis, Colcha Efraín. 2018

Cabe mencionar que los resultados son en base a los casos de estudios y que estos varían dependiendo a sus características, mientras mejores sean las características, mayor será el desempeño del sistema. Como resultado se obtiene un 100% de portabilidad, esto se justifica al ser un sistema web al cual se puede acceder en cualquier entorno que cuente o con un navegador web.

4.3 Eficiencia

Para la evaluación de la eficiencia se midió los tiempos de respuesta del sistema web y la cantidad de recursos que consume el mismo.

4.3.1 Utilización de recursos

Para la evaluación de la utilización de recursos se determinó la cantidad de recursos que el sistema utiliza en dos casos de estudios, como se puede observar en la **tabla 24-4**.

Tabla 24-4: Casos de estudio.

| Características | Caso 1 | Caso 2 |
|-------------------|------------------------|------------------------|
| Sistema Operativo | Windows Home 10 | Fedora Workstation 28 |
| Memoria RAM | 8 GB | 8 GB |
| Disco duro | 500 GB | 500 GB |
| Procesador | Intel Core I5, 250 GHz | Intel Core I5, 250 GHz |

Realizado por: Barragán Luis, Colcha Efraín. 2018

Como referencia se tomó la funcionalidad “Graficar onda 3D caso finito” en la cual se midió la cantidad de memoria, disco y CPU que consume el sistema web, los resultados se los puede observar en la **figura 22-4**.

| | | 39% CPU | 34% Memoria | 4% Disco | | | |
|----------------------|--------------------------------|---------|-------------|----------|-----------|---------------|------------------------|
| CASO 1 | Nombre | Estado | | | | | |
| | Aplicaciones (2) | | | | | | |
| | > Administrador de tareas | | 0,5% | 21,4 MB | | | |
| > Google Chrome (10) | | 38,0% | 314,1 MB | 0,1 MB/s | | | |
| CASO 2 | Nombre del proceso | Usuario | % CPU | ID | Memoria | Lectura total | Escritura total en dis |
| | chrome --type=renderer --field | luis | 27 | 8320 | 123,7 MIB | N/D | N/D |
| | chrome --type=gpu-process --f | luis | 1 | 8021 | 29,9 MIB | N/D | N/D |
| | chrome --type=broker | luis | 0 | 8087 | 13,2 MIB | N/D | N/D |
| chrome | luis | 0 | 7974 | 65,2 MIB | N/D | 5,0 MIB | |

Figura 23-4: Monitor de recursos.

Realizado por: Barragán Luis, Colcha Efraín. 2018

De acuerdo con el análisis realizado para la utilización de recursos se obtuvo para el caso de estudio 1 un total de: CPU= 38%, MEMORIA= 314.1 MB, DISCO= 0.1 MB/s y para el caso de estudio 2 CPU= 27%, MEMORIA=123,7 MB, DISCO= 0 MB/s, como se puede observar en el **grafico 2-4**.

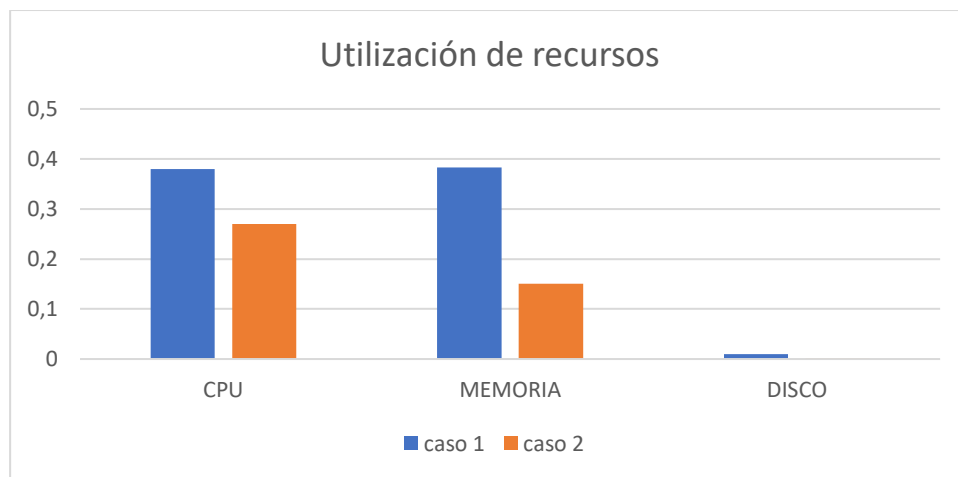


Gráfico 2-4: Resultados utilización recursos.
Realizado por: Barragán Luis, Colcha Efraín. 2018

Cabe mencionar que los resultados son en base a los casos de estudios y que estos varían dependiendo a sus características, mientras mejores sean las características, mayor será el desempeño del sistema. Como resultado se obtiene un 12% de utilización de recursos con una calificación cualitativa de baja como se puede observar en la **tabla 25-4**, siendo entre más baja mejor.

Tabla 25-4: Resultados utilización recursos.

| | Características | Caso 1 | | Caso 2 | |
|-------------------------|-----------------|----------------|---------------------|----------------|-------------------|
| | | Valor numérico | Valor cualitativo | Valor numérico | Valor cualitativo |
| CPU | 100% | 0,38 | media | 0,27 | baja |
| MEMORIA | 8 GB | 0,0383 | media | 0,0151 | baja |
| DISCO | 500 GB | 0,01 | baja | 0,00001 | baja |
| Promedio | | 0,14 | | 0,10 | |
| Promedio General | | | | | |
| | | 0,12 | Calificación | baja | |

Realizado por: Barragán Luis, Colcha Efraín. 2018

4.4 Resultados

Una vez realizado la comparación con la ayuda de los dos casos de estudio se obtuvo como resultado un valor de 100% en la métrica de portabilidad y un valor promedio de 12% de utilización de recursos correspondiente a la métrica de eficiencia, cabe mencionar que no se realizó la medición del comportamiento en el tiempo debido a que el sistema muestra simulaciones en tiempo real y el fin de dicha simulación depende del usuario, además la generación de las ondas depende de los parámetro de entrada, entre más complejas las funciones matemáticas más tiempo se demorará en dibujar.

4.5 Respuesta a la sistematización del problema

En este apartado se muestra las respuestas a las interrogantes de la sistematización del problema:

¿Cuáles son las soluciones de los modelos matemáticos para las ecuaciones de ondas de D’Alambert y Fourier en 2D Y 3D?

Luego del análisis y estudio de los modelos matemáticos de las ecuaciones de ondas en 2D y 3D se identificó las soluciones en las fórmulas de D’Alambert y Fourier, las mismas que se encuentran en el capítulo II marco teórico.

¿El desarrollo del sistema ayudará en la enseñanza de las aplicaciones de cálculo vectorial en la ESPOCH?

Mediante la utilización del sistema se puede apreciar de mejor manera el comportamiento en el tiempo de las ondas de acuerdo a los parámetros de entrada, mejorando el proceso de enseñanza - aprendizaje debido a que tanto el estudiante como docente tendrán una herramienta de apoyo para el estudio de las ondas.

¿Cómo se puede representar las fórmulas de D’Alambert y Fourier a través de un sistema web?

Con la ayuda de la computación grafica se pudo representar de manera exitosa los procesos matemáticos de D’Alambert y Fourier en 2D y 3D a través de un sistema web utilizando java, JavaScript como herramientas de desarrollo.

¿Cuál es la importancia de desarrollar sistemas para la simulación de modelos matemáticos en la ESPOCH?

El desarrollo de sistemas para la simulación de modelos matemáticos tiene como fin el brindar una herramienta que sirva de apoyo para la enseñanza en la facultad de informática y electrónica de la ESPOCH.

¿Los métodos numéricos podrán representar y resolver la solución del modelo matemático de la propagación de ondas en 2D y 3D y las fórmulas de D’Alambert y Fourier con la ayuda de un sistema web?

Una vez realizado el sistema con la ayuda de conceptos de computación gráfica, métodos numéricos, las fórmulas D’Alambert y Fourier y las herramientas de desarrollo se pudo simular de manera exitosa la propagación de ondas en 2D y 3D.

CONCLUSIONES

- Con respecto al primer objetivo, se realizó el análisis de las fórmulas de D’Alambert y Fourier en 2D y 3D, con lo que se pudo representar las soluciones de los modelos matemáticos de ondas, las cuales se utilizaron en el desarrollo del sistema web.
- Con respecto al segundo objetivo, se obtuvo un total de 6 requerimientos para el desarrollo del sistema web, los cuales satisfacen las necesidades del usuario.
- Con respecto al tercer objetivo, fue posible desarrollar el sistema web utilizando la herramienta NetBeans 8.2, con los lenguajes de programación java y JavaScript, los cuales facilitaron la representación de las fórmulas de D’Alambert y Fourier en 2D y 3D.
- Con respecto al cuarto objetivo, se efectuó un estudio comparativo para medir la portabilidad y eficiencia del sistema, obteniendo como resultado un 100% de portabilidad y un promedio de 12% de consumo de recursos.
- Con la ayuda de la computación gráfica y los métodos numéricos fue posible representar de manera exitosa las ecuaciones de ondas en 2D y 3D en un sistema web.
- Luego del estudio comparativo para validar la eficiencia se determinó que el sistema consume menos recursos en un entorno Linux.

RECOMENDACIONES

- Se recomienda aplicar la metodología de desarrollo ágil SCRUM, ya que esta ayuda a gestionar de manera eficiente las actividades de desarrollo mediante entregables semanales.
- Para una correcta utilización del sistema y garantizar un buen funcionamiento de la herramienta, se recomienda revisar el manual de usuario.
- Para futuras incorporaciones al sistema, se recomienda investigar sobre tecnologías y herramientas para simulaciones en tiempo real.
- El estándar ISO/IEC 9126 promueve la evaluación de calidad de software, se recomienda aplicar estándares de calidad para satisfacer las necesidades del cliente y obtener software de calidad.
- Se recomienda usar el sistema en dispositivos con características similares o superiores a los casos de estudio, ya que en estos se garantiza un correcto desempeño del simulador.
- Para futuros desarrollos de sistemas de simulación, se recomienda evaluar la calidad en base a la métrica de usabilidad para determinar cuál es la facilidad con la que los usuarios usan estos tipos de sistemas.

BIBLIOGRAFÍA

ALBA, F. *Comparativa de los principales sistemas gestores de Bases de Datos (SGBD)*. [en línea], 2017. [Consulta: 4 julio 2018]. Disponible en: <https://www.cursosgis.com/comparativa-de-los-principales-sistemas-gestores-de-bases-de-datos-sgbd/>

BAHIT, E. *Poo y mvc en php*. [en línea], 2001. [Consulta: 17 junio 2018]. Disponible en: <http://www1.herrera.unt.edu.ar/biblcet/wpcontent/uploads/2014/12/eugeniabahitpooymvcenphp.pdf>

BENGURIA, R.D. *La ecuación de ondas en una dimensión; solución por el método de D'Alambert*. [en línea], 2012. [Consulta: 19 septiembre 2018]. Disponible en: [http://www.fis.puc.cl/~rbenguri/CURSOS/MMF2/CLASES/clase12\(2012\).pdf](http://www.fis.puc.cl/~rbenguri/CURSOS/MMF2/CLASES/clase12(2012).pdf)

BLANCHARD, P., DEVANEY, R. y HALL, G. *Ecuaciones diferenciales*. [en línea], Boston University: International Thomson Editores, 1999. [Consulta: 19 septiembre 2018]. ISBN 968-7529-63-6 Disponible en: <https://danielmorochoruiz.files.wordpress.com/2015/09/ecuaciones-diferenciales-blanchard.pdf>

BRUSQUETTI, C. *Los simuladores educativos y su función como herramienta de aprendizaje*. [en línea], 2011. [Consulta: 2 junio 2018]. Disponible en: <http://www.abc.com.py/edicion-impresa/suplementos/escolar/los-simuladores-educativos-y-su-funcion-como-herramienta-de-aprendizaje-287876.html>

CAMPO BEDOYA, O.L. *Ecuación de D'Alambert, de la cuerda vibrante, bajo la teoría de lie*. [en línea] (tesis). (Maestría). Universidad Nacional de Colombia, Bogotá. 2014. [Consulta: 6 julio 2018]. Disponible en: <http://bdigital.unal.edu.co/46430/1/01830488.2014.pdf>

CARLOS, L. y SOLER, T. *Métodos Numéricos*. [en línea], 2010. [Consulta: 27 agosto 2018]. Disponible en: <http://www.disi.unal.edu.co/~lctorress/MetNum/LiMetNum.pdf>

CASTAÑOS, E. *Ecuación de onda*. [blog], 2016. [Consulta: 7 julio 2018]. Disponible en: <https://lidiakonlaquimica.wordpress.com/tag/ecuacion-de-onda/>

CUESTA, J. *La reinterpretación de los principios clásicos de animación en los medios digitales.* [en línea] (tesis). (Doctoral). Universidad complutense de Madrid, Madrid. 2015. [Consulta: 10 junio 2018]. Disponible en: <http://eprints.ucm.es/32970/1/T36327.pdf>

CUEVAS LEDESMA, S.M. *Análisis y simulación de vibraciones en medios continuos mediante la plataforma adefid.* [en línea] (tesis) (pregrado). Universidad de Guanajuato, Salamanca. 2015. [Consulta: 22 septiembre 2018]. Disponible en: https://adefid.com/images/adefid/publications/Tesis_CuevasLedesmaMelissa.pdf

DEAN, J. y DEAN, R. *Introducción a la Programación con Java.* [en línea], México: The McGraw-Hill Companies, Inc., 2009. [Consulta: 11 junio 2018]. ISBN 978-970-10-7278-3. Disponible en: http://www.juntos.gob.pe/modulos/mod_legal/archivos/1INTRODUCCION_4_L4_9R06R4M4C10N_C0N_J4V4.pdf

DIMES, T. *Conceptos Básicos De Scrum.* [en línea], Babelcube, Inc., 2015. [Consulta: 1 julio 2018]. ISBN 9781507102732. Disponible en: https://books.google.es/books?hl=es&lr=lang_es&id=ETuXBgAAQBAJ&oi=fnd&pg=PT7&dq=scrum&ots=1nOlf1hEuU&sig=GhJw1PBXupAYueaLWDIVKIYdXIk#v=onepage&q=scrum&f=false

GALLEGO, A. *Bootstrap 4.* [en línea], 2018. [Consulta: 4 julio 2018]. Disponible en: <https://pdfcursos.com/pdf/0177-bootstrap-4.pdf>

GARCIA, A. *¿Cuáles son las principales debilidades de Python como lenguaje de programación?* [en línea], 2007. [Consulta: 4 julio 2018]. Disponible en: <https://es.quora.com/Cuáles-son-las-principales-debilidades-de-Python-como-lenguaje-de-programación>

GARCÍA DE JALÓN, J., RODRÍGUEZ, J.I., MINGO, I., IMAZ, A., BRÁZALEZ, A., LARZABAL, A., CALLEJA, J. y GARCÍA, J. *Aprenda Java como si estuviera en primero.* [en línea], San Sebastián: TECNUN, 2000. [Consulta: 11 junio 2018]. Disponible en: <http://www4.tecnun.es/asignaturas/Informat1/AyudaInf/aprendainf/Java/Java2.pdf>

GOMEZ, M. *Top 9 Lenguajes de Programación en 2018 - Programar, presente y futuro.* [Blog], 2018. [Consulta: 4 julio 2018]. Disponible en: <https://miguelgomezsa.com/blog/top-lenguajes-programacion-2018>

GRANADOS OSPINA, A. *Las TIC en la enseñanza de los métodos numéricos.* [en línea], 2014. [Consulta: 5 julio 2018]. ISSN: 1794-8932 Disponible en: <http://www.redalyc.org/pdf/4137/413740778003.pdf>

GUTIÉRREZ, D., CEREZO, E. y PULIDO, J.I. *Animación por ordenador.* [en línea], Zaragoza: Departamento de Informática e Ingeniería de Sistemas, 2006. [Consulta: 10 junio 2018]. ISBN: 84-689-9891-5 Disponible en: http://webdiis.unizar.es/~diegog/ficheros/teaching/libro_animacion.pdf

HEARN, D. y BAKER, M.P. *Gráficos por Computadora con OpenGL.* [en línea], Madrid: Pearson educación S.A, 2006. [Consulta: 10 junio 2018]. ISBN 978-84-205-3980-5. Disponible en: <https://ingenieriayeducacion.files.wordpress.com/2013/12/graficosporcomputadorayopengl.pdf>

IRUELA, J. *Los gestores de bases de datos más usados.* [en línea], 2016. [Consulta: 4 julio 2018]. Disponible en: <https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/>

JIMÉNEZ QUINTANA, E.W. *Ecuaciones en derivadas parciales: una introducción a la teoría clásica.* [en línea] (tesis) (pregrado). Universidad del BÍO-BÍO. Chillán, 2015. [Consulta: 7 julio 2018]. Disponible en: http://repobib.ubiobio.cl/jspui/bitstream/123456789/1912/1/Jimenez_Quintana_Edwars.pdf

JONES, O.W. *Diferencias entre la animación 3D y 2D.* [blog], 2018. [Consulta: 10 junio 2018]. Disponible en: https://techlandia.com/diferencias-animacion-3d-2d-lista_74941/

LARGO, A. y MARIN, E. *Guía técnica para evaluación de software.* [en línea], 2005. [Consulta: 5 julio 2018]. Disponible en: https://jrvargas.files.wordpress.com/2009/03/guia_tecnica_para_evaluacion_de_software.pdf

LOCKHART, T. *Manual del usuario de PostgreSQL.* [en línea]. Postgres Global Development Group. 1999. [Consulta: 16 junio 2018]. Disponible en: <http://index-of.co.uk/SERVIDORES/Postgres-User.pdf>

MAESTRE, M. *Servidor de aplicaciones Web Glassfish*. [en línea], 2016. [Consulta: 1 julio 2018]. Disponible en: <https://slidex.tips/download/que-es-glassfish-caracteristicas-de-glassfish-2-caracteristicas-de-glassfish>

MELGAREJO, M., GONZALES, F. y RAMÍREZ-MARTÍN, C. *Solución de la ecuación diferencial parcial de una membrana vibrante mediante Maple y MatLab*. [en línea], 2013. [Consulta: 22 septiembre 2018]. ISSN 1870-9095. Disponible en: http://www.lajpe.org/dec13/14-LAJPE_844_Carlos_Ramirez.pdf

MOLINA, J., LOJA, N., ZEA, M. y LOAIZA, E. *Evaluación de los Frameworks en el Desarrollo de Aplicaciones Web con Python*. [en línea], 2016, (Ecuador). [Consulta: 5 julio 2018]. ISSN 2314-2642. Disponible en: <http://revistas.unla.edu.ar/software/article/view/1149/1035>

MONTOYA, L., SEPÚLVEDA, M. y JIMÉNEZ, L. *Análisis comparativo de las metodologías ágiles en el desarrollo de software aplicadas en Colombia*. [en línea], 2016, (Colombia). [Consulta: 5 julio 2018]. Disponible en: https://www.researchgate.net/publication/317840767_Analisis_comparativo_de_las_metodologias_agiles_en_el_desarrollo_de_software_aplicadas_en_Colombia

MORA, G. *La evolución de JavaScript*. [en línea], 2017. [Consulta: 4 julio 2018]. Disponible en: http://83.39.242.131/files/1519371689_JavaScript.pdf

OVIEDO RONDÓN, E.O. *Como Usar y Desarrollar un Modelo Matemático para Nutrición Avícola*. [en línea], 2014. [Consulta: 5 julio 2018]. ISBN 0124-6690. Disponible en: https://amevea.org/RevistaPlumazos/Plumazos_047.pdf

PEIRÓ, D.G. *Modelos en EDPs*. [en línea], Universidad Politécnica de Valencia, Departamento de Matemática Aplicada, 2009. [Consulta: 23 septiembre 2018]. Disponible en: <http://personales.upv.es/dginesta/docencia/posgrado/modelos.pdf>

PÉREZ, D. *¿Qué es JavaScript?* [blog], 2007. [Consulta: 12 junio 2018]. Disponible en: <http://www.maestrosdelweb.com/que-es-javascript/>

QUIJANO, L. *Animación 2D*. [blog], 2013. [Consulta: 10 junio 2018]. Disponible en: http://quijano1993.blogspot.com/2013/03/1_18.html

RODRÍGUEZ, M. *Estudio comparativo entre las metodologías ágiles y las metodologías tradicionales para la gestión de proyectos software*. [en línea] (tesis) (Maestría). Universidad de Oviedo. 2015. [Consulta: 5 julio 2018]. Disponible en: <http://digibuo.uniovi.es/dspace/bitstream/10651/32457/6/TFMMIJGarciaRodriguezRUO.pdf>

RUBINOS CARVAJAL, A.M. y NUEVO LEÓN, H.A. *Seguridad en bases de datos*. [en línea], 2011, (Cuba), vol. 5, no. 1. [Consulta: 16 junio 2018]. Disponible en: <http://www.redalyc.org/html/3783/378343671005/>

SARAY, M. *JAVA Vs PHP: Eterno debate*. [blog], 2015. [Consulta: 4 julio 2018]. Disponible en: <https://www.facilcloud.com/noticias/java-vs-php-eterno-debate/>

SCHWABER, K. y SUTHERLAND, J. *La Guía de Scrum*. [en línea], Scrum.Org and Scrum Inc. 2013. [Consulta: 1 julio 2018]. Disponible en: <https://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-es.pdf>

SOLANO, B. *¿Qué es Glassfish?* [blog], 2009. [Consulta: 1 julio 2018]. Disponible en: <https://bannysolano.wordpress.com/2009/08/23/¿que-es-glassfish/>

SUAREZ, R. *Animación y Visualización de Fenomenos Naturales*. [en línea] (tesis). Universidad de las Américas Puebla. Cholula, Puebla, 2003. [Consulta: 10 junio 2018]. Disponible en: http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/suarez_r_pk/

VELASCO, R. *Llega el nuevo NetBeans 8.2 con soporte completo para PHP7*. [blog], 2016. [Consulta: 17 junio 2018]. Disponible en: <https://www.redeszone.net/2016/10/03/llega-nuevo-netbeans-8-2-soporte-completo-php7/>

VÉLEZ, J.F., ARISTIZÁBAL, M.L., CORONADO, R., QUICENO, W., ORTIZ, W.A. y ZAPATA VANEGAS, M. *El Postgres: un manejador de dominio público de bases de datos relacionales*. [en línea], 1999 (Colombia) no. 18, pp. 106-112. [Consulta: 16 junio 2018]. Disponible en: <http://aprendeenlinea.udea.edu.co/revistas/index.php/ingenieria/article/view/325820>

VILLA MARTÍNEZ, H.A. *Introducción a la programación en Java*. [en línea], Universidad de Sonora. 2015. [Consulta: 11 junio 2018]. Disponible en:

https://www.researchgate.net/publication/287646094_Introduccion_a_la_programacion_en_Java

ZUAZUA, E. *Métodos Numéricos de resolución de Ecuaciones en Derivadas Parciales*. [en línea], 2009. [Consulta: 8 julio 2018]. Disponible en:

http://www.bcamath.org/documentos_public/archivos/personal/comites/1_notas-05_065-complete.pdf

