**Integrantes**

WILSON EDUARDO TOMAS AlCANTARA 20193550

YEIMELIN ENCARNACION GALVEZ 2019-0373

FRANCISCO JAVIER FEBLES JIMENEZ 2018-3659

Practica Final de Proyecto Integrador 2

2022

**DISEÑO DE UNA APLICACIÓN PARA CALCULO PRESUPUESTARIO DE CURSOS VIRTUAL EN EL ÁREA DE LA INGENIERÍA DE SOFTWARE EN REPUBLICA DOMINICANA.**

**Tabla De Contenido**

[Introducción 6](#_Toc104716023)

[1. ESTADO DEL ARTE 7](#_Toc104716024)

[2. PLANTEAMINETO DEL PROBLEMA 9](#_Toc104716025)

[3. Objetivos 10](#_Toc104716026)

[3.1. Objetivo General 10](#_Toc104716027)

[3.2. Objetivos Específicos 10](#_Toc104716028)

[**4. MARCO CONCEPTUAL** 11](#_Toc104716029)

[**4.1. Generalidades** 11](#_Toc104716030)

[**4.2. Definición de conceptos** 12](#_Toc104716031)

[**4.2.1. Plataforma** 12](#_Toc104716032)

[**4.2.2. Contenido educativo** 12](#_Toc104716033)

[**4.2.3. E-Learning** 12](#_Toc104716034)

[**4.2.4. Calculo Presupuestario** 12](#_Toc104716035)

[**4.2.5. Requisitos del sistema** 13](#_Toc104716036)

[**4.2.6. Base de datos** 14](#_Toc104716037)

[**4.2.7. Interfaz de usuario** 14](#_Toc104716038)

[**4.2.8. Patrones de Diseño de Interfaz de Usuario** 15](#_Toc104716039)

[**4.2.9. Inteligencia de Negocios** 15](#_Toc104716040)

[4.2.10. Arquitectura de Software 16](#_Toc104716041)

[**4.2.11. Diagrama de arquitectura de Software** 17](#_Toc104716042)

[**4.2.12. Front-end** 17](#_Toc104716043)

[**4.2.13. Back-end** 18](#_Toc104716044)

[**4.2.14. Control de versiones** 18](#_Toc104716045)

[**4.2.15. Git** 19](#_Toc104716046)

[**4.2.16. GitHub** 19](#_Toc104716047)

[**4.2.17. Proyecto** 19](#_Toc104716048)

[**4.2.18. Gestión de Proyectos** 19](#_Toc104716049)

[**4.2.19. Metodologías de Gestión de Proyectos** 19](#_Toc104716050)

[**4.2.20. Herramientas de Gestión de Proyectos** 20](#_Toc104716051)

[5. Levantamiento de Datos y Análisis 20](#_Toc104716052)

[5.2 Análisis 21](#_Toc104716053)

[6. Método 22](#_Toc104716054)

[6.1 Definición de etapas 23](#_Toc104716055)

[6.2 Estudio de viabilidad 25](#_Toc104716056)

[6.3 Diseño de la investigación 25](#_Toc104716057)

[6.4 Locación de estudio 25](#_Toc104716058)

[6.5 Estrategia de muestreo 25](#_Toc104716059)

[6.6 Recolección de datos 26](#_Toc104716060)

[6.7 Diseño del sistema 27](#_Toc104716061)

[6.8 Implementación del sistema 27](#_Toc104716062)

[6.9 Pruebas del sistema 27](#_Toc104716063)

[7. Propuesta de Valor 27](#_Toc104716064)

[7.1 ¿Como se Realizará? 27](#_Toc104716065)

[7.2 ¿Qué Hará el Software? 28](#_Toc104716066)

[7.3.1 Modelos Canvas 28](#_Toc104716067)

[7.3.2 Matriz de Marco Lógico 29](#_Toc104716068)

[7.4 Herramienta de Gestión de Proyectos. 29](#_Toc104716069)

[7.4.1 Trello 29](#_Toc104716070)

[7.5 Diseño del lado de cliente 31](#_Toc104716071)

[8 Análisis Económico De La Propuesta 31](#_Toc104716072)

[9 Resultados 32](#_Toc104716073)

[10 Discusión de los Resultados. 33](#_Toc104716074)

[11 Conclusión 34](#_Toc104716075)

[12 Referencias Bibliográficas 34](#_Toc104716076)

[13 Anexos 35](#_Toc104716077)

[14 Glosario de Términos 35](#_Toc104716078)

# Introducción

# ESTADO DEL ARTE

**Plan de trabajo**

Tabla 1 (Tabla del plan de trabajo)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Acciones por realizar*** | ***Semana 1*** | ***Semana 2*** | ***Semana 3*** | ***Semana 4*** | ***Semana 5*** | ***Semana 6*** | ***Semana***  ***7*** | ***Semana 8*** | ***Semana 9*** |
| Selección del Tema |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| introducción |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Definición del estado del arte |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Plan de trabajo |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Planteamiento del problema |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Objetivos |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Marco Conceptual |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Levantamiento de datos y análisis |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Método |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| ***Acciones por realizar*** | ***Semana 10*** | ***Semana 11*** | ***Semana 12*** | ***Semana 13*** | ***Semana 14*** | ***Semana 15*** | ***7*** |  |  |
| Defensa Oral |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Propuesta |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Análisis Económico |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Resultados |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Conclusión |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

# PLANTEAMINETO DEL PROBLEMA

# Objetivos

## Objetivo General

.

## Objetivos Específicos

**4. MARCO CONCEPTUAL**

**4.1. Generalidades**

**4.2. Definición de conceptos**

**4.2.1. Plataforma**

**4.2.2. Contenido educativo**

**4.2.3. E-Learning**

**4.2.4. Calculo Presupuestario**

**4.2.5. Requisitos del sistema**

**4.2.6. Base de datos**

**4.2.7. Interfaz de usuario**

**4.2.8. Patrones de Diseño de Interfaz de Usuario**

**4.2.9. Inteligencia de Negocios**

### 4.2.10. Arquitectura de Software

**4.2.11. Diagrama de arquitectura de Software**

**4.2.12. Front-end**

**4.2.13. Back-end**

**4.2.14. Control de versiones**

**4.2.15. Git**

**4.2.16. GitHub**

**4.2.17. Proyecto**

**4.2.18. Gestión de Proyectos**

**4.2.19. Metodologías de Gestión de Proyectos**

***4.2.19.1. Metodología en Cascada***

***4.2.19.2. Metodología Agile***

***4.2.19.3. Metodología Scrum***

***4.2.19.4. Metodología Kanban***

**4.2.20. Herramientas de Gestión de Proyectos**

***4.2.20.1. Trello***

# Levantamiento de Datos y Análisis

# 5.2 Análisis

# Método

Figura 1 (Metodología de Software en Cascada).

Fuente: Propia.

## 6.1 Definición de etapas

* ***Análisis***. El equipo de desarrollo de productos analiza los requisitos y comprende plenamente los problemas. Se trata de una fase de investigación que no incluye la construcción. El equipo intenta hacer todas las preguntas y obtener todas las respuestas que necesita para construir el requisito del producto.
* *Diseño*. Los desarrolladores de software diseñan una solución técnica a los problemas planteados por los requisitos del producto, incluyendo escenarios, diseños y modelos de datos. Esta fase suele ir acompañada de documentación para cada requisito, lo que permite a otros miembros del equipo revisarla para su validación.
* ***Implementación***. Una vez aprobado el diseño, comienza la ejecución técnica. Esta suele ser la fase más corta porque la investigación y el diseño se han realizado con antelación.
* ***Pruebas***. Una vez finalizada la implementación completa, es necesario realizar pruebas antes de que el producto pueda ser entregado a los clientes. El equipo de pruebas de software utilizará los documentos de diseño, los personajes y los escenarios de casos de usuario entregados por el director de producto para crear sus casos de prueba.
* ***Verificación***. En esta fase el cliente revisa el producto para asegurarse de que cumple los requisitos establecidos al principio del proyecto. Para ello, se entrega el producto terminado al cliente.
* ***Mantenimiento***. El cliente utiliza regularmente el producto durante la fase de mantenimiento, descubriendo fallos, características inadecuadas y otros errores que se produjeron durante la producción. El equipo de producción aplica estas correcciones según sea necesario hasta que el cliente esté satisfecho.

El modelo en cascada es un modelo muy popular en el desarrollo de software. Se ha estado utilizando con éxito durante mucho tiempo, pero recientemente sólo se utiliza para proyectos pequeños en los que las especificaciones son claras.

Fuente: McConnell, S. (2006).

## 6.2 Estudio de viabilidad

## 6.3 Diseño de la investigación

## 6.4 Locación de estudio

## 6.5 Estrategia de muestreo

## 6.6 Recolección de datos

## 6.7 Diseño del sistema

## 6.8 Implementación del sistema

## 6.9 Pruebas del sistema

# Propuesta de Valor

## 7.1 ¿Como se Realizará?

—

## ¿Qué Hará el Software?

—

—

## Modelos Canvas

Figura 2 (modelo canvas)

Fuente: Propia

## 7.3.2 Matriz de Marco Lógico

Tabla 2 (matriz de marco lógico)

Fuente: Propia.

## Herramienta de Gestión de Proyectos.

### Trello

**Explicación**

1. Se crea el tablero del proyecto.

Figura 3 (Tablero del proyecto)

Fuente (https://trello.com/)

1. Se crean las tareas que se van a realizar.

Figura 4 (Tabla de tareas)

Fuente: (<https://trello.com/>)

1. Se lleva un registro de las tareas en curso.

Figura 5 (Tabla de tareas en proceso)

Fuente: (https://trello.com/)

1. Se enlistan las tareas ya realizadas

Figura 6 (Tabla de tareas realizadas)

Fuente: (https://trello.com/)

# Diseño del lado de cliente

# Análisis Económico De La Propuesta

Fuente: Propia.

**Tabla 5 (costos fijos)**

**Fuente: Propia.**

**Tabla 6 (costos fijos anuales)**

**Fuente: Propia.**

# Resultados

**Requisitos funcionales y no funcionales**

**Funcionales (F)**

F1. El sistema debe permitir la autenticación de usuarios.

F2. El sistema debe permitir el registro de nuevos usuarios.

**No Funcionales (NF)**

NF1. El sistema debe validar todos los campos de entrada de datos.

NF2. El sistema debe tener trazas de auditoria en la base de datos.

NF3. El sistema debe poseer un diseño amigable para el usuario.

**Arquitectura del sistema**

La arquitectura usada en este proyecto es la famosa arquitectura cliente-servidor en la que la mayoría de las aplicaciones web se basan. La arquitectura cliente-servidor, se basa en que muchos clientes solicitan y reciben servicios de un servidor centralizado. Los ordenadores clientes proporcionan una interfaz que permite a un usuario de ordenador solicitar servicios del servidor y mostrar los resultados que éste devuelve. Los servidores esperan a que lleguen las peticiones de los clientes y luego responden a ellas.

**Gestor de base de datos**

**Diagrama de Entidad Relación**

Figura 16 (diagrama de entidad relación)

**Diagrama de Casos de Uso**

Figura 17 (diagrama de casos de uso)

**Front-End**

**Back-End**

**Pruebas de Unidad**

**Patrón de diseño y principios.**

**Patrón repositorio**

# Discusión de los Resultados.

# Conclusión

# Referencias Bibliográficas

# Anexos

# Glosario de Términos

**Cliente:** El cliente es una aplicación informática o un ordenador que consume un servicio remoto en otro ordenador conocido como servidor, normalmente a través de una red de telecomunicaciones.​

**Software:** Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas.

**Diagrama:** Representación gráfica de las variaciones de un fenómeno o de las relaciones que tienen los elementos o las partes de un conjunto.

**Programa:** Un programa informático o programa de computadora es una secuencia de instrucciones, escritas para realizar una tarea específica en una computadora.

**Ventanas:** Una ventana es un área visual, normalmente de forma rectangular, que contiene algún tipo de interfaz de usuario.