# Introducción

Cada ves que se inicia un proyecto es importante considerar que tendrá aspectos diferentes a proyectos anteriores, a pesar de compartir procesos de negocio. Por esto es imporante realizar un estudio sobre la factibilidad del sistema que se desea desarrollar, para lo cuál se hará lo siguiente:

1. Planteamiento del problema: Identificar el problema de negocio que se desea solucionar con un sistema de softwareNivel de impacto Definición
2. 0 Sin impacto
3. 1 Imp
4. Arquitectura general: Se describe un modelo de la arquitectura planeada y se toman consideraciones técnicas
5. Tareas a realizar por roles: Se identifican usuarios y las tareas que necesitan realizar estos para cumplir con el objetivo
6. Organigrama de actividades: Se identifican las fases del proyecto, las tareas que cada etapa requiere y el tiempo necesario para resolverlas.
7. Factibilidad técnica: Se hará un análisis para resolver que tan viable e es realizar el sistema con las tecnologías disponibles.
8. Factibilidad económica: Se establecerán tareas y sus respectivos costos para tener establecer un presupuesto y conocer sí el costo-beneficio es asumible.
9. Factibilidad operativa: Se realizará un análisis sobre esfuerzo que emplea ejecutar el software y la capacidad de la empresa y el personal para responder.
10. Conclusiones: Se establece y fundamenta que tan viable es realizar el sistema.

# Planteamiento del problema

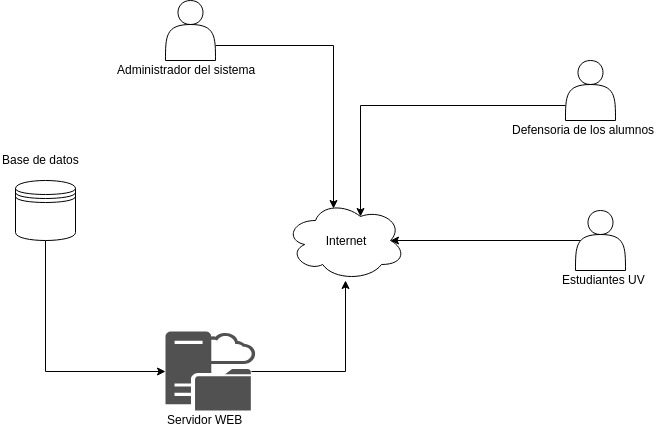
Actualmente, en la sociedad mexicana, no se tiene una costumbre de denunciar cualquier suceso que lo amerite ya que se tiene una idea de que no se tomará en cuenta y el hecho quedará en inpunidad. Sumado a esto, el panorama de actual indica que aún existen muchos problemas en universidades públicas y privadas; negligencia, acoso o abuso son ejemplos de esto, por lo que se plantea la realizar un sistema que permita iniciar un proceso de denuncia en línea y dar seguimiento de este a través de este.

Por lo tanto, el sistema contemplará las siguientes funciones:

* Permitir iniciar una denuncia a través del sistema por parte de un estudiante
* Consultar las denuncias iniciadas a los directivos encargados
* Contestar una denuncia por parte de los directivos encargados
* Consultar el estado de la denuncia y las respuestas por parte de un estudiante
* Consultar el estatuto de la institución por parte de un estudiante

# Arquitectura general

La siguiente figura representa un modelo general del funcionamiento del sistema de denuncias estudiantiles. En la que se puede apreciar la distribución del sistema y los usuarios identificados



## Consideraciones el software

Para desarrollar el tomamos las siguientes consideraciones para elegir las tecnologías que usaremos.

Lenguaje:

PHP:

* Lenguaje totalmente libre y abierto
* Curva de aprendizaje muy baja
* Los entornos de desarrollo son de rápida y fácil configuración
* Fácil despliegue: paquetes totalmente autoinstalables que integran PHP
* Fácil acceso a bases de datos
* Comunidad muy grande

Python:

* Facililidad de uso
* Legibilidad de código
* Multiplataforma
* Facilidad de escritura de código asíncronico
* Abundancia de bibiotecas
* Gran base de usuaios

### Sistema operativo:

Ubuntu live server:

Ubuntu está basado en la distribución Debian GNU/Linux y soporta oficialmente dos arquitecturas de hardware: Intel x86, AMD64. Sin embargo ha sido portada extraoficialmente a cinco arquitecturas más: PowerPC, SPARC (versión "alternate"), IA-64, Playstation 3 y HP

PA-RISC. Al igual que casi cualquier distribución basada en Linux, Ubuntu es capaz de actualizar a la vez todas las aplicaciones instaladas en la máquina a través de repositorios, a diferencia de otros sistemas operativos comerciales, donde esto no es posible.

### Manejador de base de datos:

MySQL:

es muy utilizado en aplicaciones web como, Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL- PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

## Consideraciones del hardware

Considerando que la universidad veracruzana ofrece varios servicios web y dispone de infraestructura para esto, se planea que el sistema sea instalado en los equipos de los que dispone la universidad veracruzana.

# Tareas a realizar por roles

|  |  |
| --- | --- |
| Actor | Función |
| Denunciastes | * Iniciar denuncia * Consultar estado de denuncia * Consultar estatuto académico |
| Defensoría de los alumnos | * Consultar denuncias * Responder denuncia |

# Organigrama de actividades

1. Análisis de requerimientos: En esta primera etapa se recabaran los datos necesarios para describir los servicios que debe de proporcionar el sistema, y las restricciones operativas que deberá de cumplir.
2. Diseño del sistema: El sistema a desarrollar estará compuesto por subsistemas que proporcionan algún conjunto de servicios relacionados. El proceso de diseño deberá de identificar estos subsistemas, establecer un marco de control y comunicación entre los mismos, para su elaboración independiente.
3. Desarrollo del sistema: Se elaboran las interfaces, las bases de datos y se le agregaran los datos necesarios para el funcionamiento del mismo, siguiendo el modelo diseñado previamente.
4. Pruebas del sistema y corrección de errores: En esta etapa se someterá a pruebas el sistema, tratando de someterlo a las condiciones y características más reales posibles, múltiples inscripciones, cambios, etc.
5. Instalación del sistema: Esta etapa se sujeta normalmente a ventanas de mantenimiento en las cuales se pueda integrar el sistema de tal manera que no afecte el servicio productivo. Se instalará en el ambiente productivo. Se harán prueba hasta que todo quede de manera perfecta. Vigilando durante los primeros días su funcionamiento para verificar que no haya ningún problema.

Etapa y duración de cada etapa

|  |  |
| --- | --- |
| Etapa | Duración |
| Requerimientos | 20 días |
| Diseño | 14 días |
| Desarrollo del sistema | 14 días |
| Pruebas del sistema | 7 días |
| Instalación del sistema | 7 días |

Etapa y actividades correspondientes

|  |  |
| --- | --- |
| Etapa | Actividad |
| Requerimientos | * System request * Casos de uso * Especificación de requerimientos * Análisis de factibilidad |
| Diseño | * Modelo de domino * Modelo de clases * Modelo E-R * Modelo relacional * Modelo de GUI |
| Desarrollo del sistema | * Construcción de prototipo * Construcción de la base de datos * Construcción de modulos |
| Pruebas del sistema | * Pruebas de integración * Pruebas de aceptación |
| Instalación del sistema |  |
|  |  |

Actividades de requerimientos y análisis

|  |  |
| --- | --- |
| Actividad | Duración en horas |
| System request | 1 |
| Casos de uso | 18 |
| Especificación de requerimientos | 20 |
| Análisis de factibilidad | 15 |
| Estimación de esfuerzo | 6 |
| Plan de calidad | 5 |
| Plan de control de riesgos | 5 |
| Plan de control de cambios | 4 |

Actividades de diseño

|  |  |
| --- | --- |
| Actividad | Duración en horas |
| Modelo de dominio | 4 |
| Modelo de clases | 8 |
| Modelo E-R | 8 |
| Modelo relacional | 5 |
| Modelo de interfaz | 8 |

Actividades de construcción

|  |  |
| --- | --- |
| Actividad | Duración en horas |
| Construcción de paquetes | 40 |
| Construcción de prototipo | 20 |

Actividades de pruebas

|  |  |
| --- | --- |
| Actividad | Duración en horas |
| Pruebas de integración | 20 |
| Pruebas de aceptacion | 20 |

Actividades de instalación

|  |  |
| --- | --- |
| Actividad | Duración en horas |
| Instalación | 25 |