



Universidad Tecnológica de Cancún

ORGANISMO PÚBLICO DESCENTRALIZADO DEL GOBIERNO DEL ESTADO DE QUINTANA ROO

Sistema de Gestión de Estadías

Versión 1.0

**T.S.U. en Tecnologías de la Información y
Comunicación área Desarrollo de Software
Multiplataforma**

P R E S E N T A N:

Valier Sánchez Carlos Francisco

22393231

Moreno Godinez Olga Elizabeth

22393144

Hau Pech Marco Antonio

22393169

Zamora Gorospe Alexis Alberto

22393152

Generación: 2022 - 2024

Asesor académico: Mtra. Mayra Guadalupe Fuentes Sosa

Empresa: Universidad Tecnológica de Cancún

Asesor empresarial: Nombre

Cancún; Quintana Roo

Marzo de 2024

CONTENIDO

INTRODUCCIÓN	3
I. PLANTEAMIENTO DEL PROYECTO.....	4
a. Descripción del problema.....	4
b. Descripción de la solución	4
II. MARCO TEÓRICO	5
III. DESARROLLO DEL PROYECTO	15
1. Diagrama de la aplicación web.....	16
2. Lista de los requerimientos funcionales de la aplicación	16
3. Definición de requerimientos funcionales asignados al equipo	18
4. Desarrollo de Diagramas UML	20
a) Diagramas de Casos de uso	20
b) Diagramas de secuencia	21
c) Diagrama de distribución.....	23
5. Diseño de la Base de datos	24
a) Descripción de la base de datos	24
b) Presentación y descripción de la base de datos	26
c) Presentación y descripción del diccionario de datos	27
IV. RESULTADOS O PRODUCTOS OBTENIDOS.....	29
V. CONCLUSIONES.....	32
BIBLIOGRAFÍA	33

INTRODUCCIÓN

Si preguntan una razón por la cual los requerimientos son importantes, se puede decir, porque definen el alcance del proyecto, pero no solamente hay una razón, existe una gran variedad de razones por la cual especificar los requerimientos son importantes y más en software, su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes [1]. En este documento podrás visualizar el desglosamiento de cada uno de los requerimientos con los que contará el sistema de gestión de estadías, de igual forma, se definirá el alcance que tendrá el SGE y quienes participarán para su desarrollo. Para entrar un poco más en contexto, los requerimientos son una descripción de una condición o capacidad que debe cumplir un sistema, ya sea derivada de una necesidad de usuario identificada u otro documento formalmente impuesto al inicio del proceso [1]. Del cual parten dos tipos de requerimientos, los requerimientos funcionales y los no funcionales; el primero define las funciones que el sistema será capaz de realizar, describen las transformaciones que el sistema realiza sobre las entradas para producir salidas [1]; mientras que el segundo tiene que ver con características que de una u otra forma puedan limitar el sistema, por ejemplo, el rendimiento, fiabilidad, mantenimiento, seguridad, portabilidad, estándares [1]. Como se puede observar, los requerimientos funcionales especifican que debe de hacer el sistema, mientras que los no funcionales definen como deben de hacerlo. Para realizar la obtención de los requerimientos se hizo uso de herramientas como lo son las entrevistas, observaciones y encuestas, las cuales tienen como objetivo, conocer los documentos que se expiden y quienes tienen la obligación de realizarlo, de igual forma, conocer las inquietudes de los usuarios con el SGE anterior. Como se puede observar, la importancia de la especificación de requerimientos de software radica en que genera una visión más clara para empezar a desarrollar el proyecto, teniendo una guía de lo que se necesita y poder solventar esas necesidades.

I. PLANTEAMIENTO DEL PROYECTO

Nombre de la empresa: Universidad Tecnológica de Cancún

Objetivo general del proyecto

Las reglas para redactar un objetivo general son: Iniciar con un verbo en infinitivo indicando en un solo párrafo que van a hacer, para que y como lo van a hacer.

Innovar el Sistema de Gestión de Estadías profesionales de los estudiantes que cursen 6to Y 11vo cuatrimestre, esto para facilitar los procesos que conlleva la realización de la Cédula de Anteproyecto. Haciendo uso de tecnologías como lo es Laravel, MySQL y MongoDB, de igual manera, utilizando PHP Y JavaScript como lenguajes de programación con las cuales se realiza una mejora de forma exponencial y adaptativa del sistema.

Justificación del proyecto

a. Descripción del problema

Identificar y explicar las causas que originan el problema

En el Sistema de Gestión de Estadías (SGE), se enfrentan diversos desafíos para su digitalización. La ausencia de una plataforma centralizada dificulta que los profesores puedan revisar y retroalimentar a los alumnos de manera ordenada y eficaz. Esto no solo genera retrasos en la evaluación, sino también riesgos de pérdida de documentos físicos y una falta de seguimiento adecuado del progreso de los estudiantes con respecto a sus células de anteproyecto.

b. Descripción de la solución

Explicar como proponen solucionar las causas que originaron el problema y las necesidades identificadas.

La propuesta que se tiene para solventar los problemas y necesidades del SGE, es desarrollar una herramienta de comunicación la cual se implementara entre el asesor y estudiante en la elaboración de su anteproyecto, que consiste en hacer comentarios y a su vez, estos tengan una respuesta; otro punto a abordar es, generar la digitalización de los documentos dentro del sistema, así mismo, la implementación de un calendario, con la que los estudiantes estarán pendientes de sus próximas actividades.

II. MARCO TEÓRICO

En esta sección del documento deben proporcionar conceptos e información sobre las herramientas y técnicas utilizadas durante el desarrollo del proyecto, recuerden que cada concepto proporcionado debe ir acompañado al final con los corchetes cuadrados y un número interno, ejemplo [4] de acuerdo al formato IEEE indicando la fuente de información consultada al investigarlo, estas referencias pueden ser tomadas de libros, links profesionales, artículos arbitrados, etc.

1. Técnicas de recolección de requerimientos aplicadas

Explicar que es una técnica de recolección, porque es importante utilizarlas y los conceptos de cada una de las técnicas de recolección que utilizaron.

Una técnica de recolección se refiere a un método específico utilizado para recopilar datos de manera sistemática y estructurada, para obtener información de las partes interesadas (stakeholders) sobre las necesidades y expectativas que tienen para un proyecto de software. Es un proceso fundamental en la investigación y recopilación de información con el objetivo de obtener resultados significativos y respaldar conclusiones [2].

Importancia de utilizar técnicas de recolección:

El uso de técnicas de recolección adecuadas contribuye a la precisión y fiabilidad de los datos, asegurando que la información recopilada sea representativa y válida; así mismo, si estas son eficientes permiten ahorrar tiempo y recursos, lo que es crucial en cualquier investigación, también, al seleccionar las técnicas correctas, se puede asegurar que los datos recopilados sean pertinentes para los objetivos de la investigación; utilizar técnicas consistentes asegura que los datos se recopilen de manera uniforme a lo largo del tiempo y entre diferentes observadores o lugares, lo que mejora la coherencia de los resultados. Las técnicas de recolección bien diseñadas permiten extrapolar los resultados a poblaciones más amplias, mejorando la capacidad de generalización de los hallazgos [2].

Conceptos de técnicas de recolección:

Entrevistas: Conversaciones directas con los stakeholders para comprender sus necesidades y expectativas. Se puede definir como un “intento sistemático de recoger información de otra persona” a través de una comunicación interpersonal que se lleva a cabo por medio de una conversación estructurada [3].

Encuestas: consisten en un conjunto prediseñado de preguntas normalizadas, dirigidas a una muestra socialmente representativa de individuos, con el fin de conocer sus opiniones; pueden ser estructuradas (preguntas cerradas) o no estructuradas (preguntas abiertas). La clave para el éxito es que tengan un propósito y audiencia claramente definida, establecer fechas tope para llenar la encuesta, con preguntas claras y concisas [3].

2. Metodología de desarrollo Design Thinking

Describir que es la metodología design of thinking, explicar sus fases y presentar su gráfico.

El Design Thinking es una metodología que se centra en comprender las necesidades de los usuarios, desafiar suposiciones, redefine problemas y crea soluciones innovadoras para mejorar productos, servicios o procesos.

Design Thinking mezcla las capacidades del pensamiento creativo y el pensamiento racional. De la creatividad y la razón [1].

Fases de la Metodología de Design Thinking:

Empatizar: En esta fase, los equipos buscan comprender las necesidades y deseos de los usuarios. Esto implica la observación directa, la interacción y la empatía con los usuarios para obtener una comprensión profunda de sus experiencias y desafíos [4].

Definir: En esta etapa, se sintetizan y analizan los hallazgos de la fase de empatía. Se definen claramente los problemas y se establecen puntos de vista específicos para guiar el proceso de diseño [4].

Idear: Durante esta fase, se fomenta la generación libre y abierta de ideas. Los equipos buscan soluciones creativas a los problemas identificados sin preocuparse inicialmente por su viabilidad. Se alienta la creatividad y la colaboración [4].

Prototipar: En esta etapa, se desarrollan prototipos de las soluciones propuestas. Estos prototipos pueden ser modelos simples, bocetos o representaciones tangibles que permitan al equipo y a los usuarios experimentar con las ideas de manera concreta [4].

Probar: Los prototipos se ponen a prueba con los usuarios para obtener retroalimentación. Esta fase proporciona información valiosa sobre la efectividad de las soluciones propuestas y ayuda a identificar áreas de mejora [4].

Gráfico del Design Thinking:

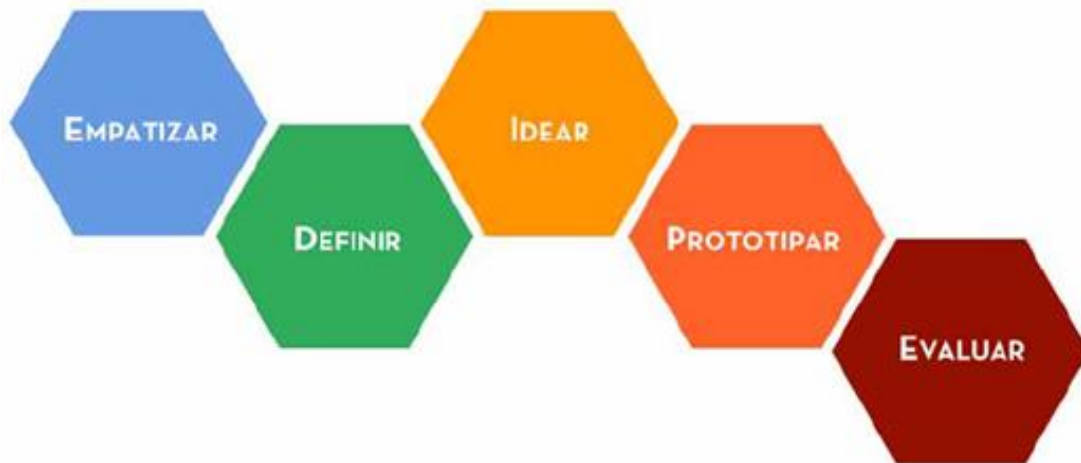


Ilustración 1 Gráfico obtenido de internet

3. Aplicación web

Investigar y definir el concepto de aplicación web

La aplicación web es un software que se ejecuta en un navegador web y que se accede a través de internet. Las empresas tienen que intercambiar información y proporcionar servicios de forma remota. Utilizan aplicaciones web para comunicarse con los clientes cuando lo necesiten y de una forma segura [5].

Algunas características comunes de las aplicaciones web incluyen:

Acceso a través de un Navegador: Los usuarios pueden acceder a la aplicación web utilizando un navegador web estándar, como Google Chrome, Mozilla Firefox, Safari o Microsoft Edge [5].

Las aplicaciones web suelen tener interfaces de usuario intuitivas y fáciles de usar, diseñadas para facilitar la interacción del usuario; así mismo, muchas utilizan tecnologías que permiten la actualización en tiempo real de la información, lo que mejora la colaboración y la respuesta rápida a los cambios. La mayoría de las

aplicaciones web almacenan datos y realizan procesamiento en servidores remotos, aprovechando servicios en la nube para ofrecer escalabilidad y flexibilidad, por ende, requieren una conexión a Internet para funcionar correctamente; debido a su naturaleza basada en el navegador, son accesibles desde una variedad de dispositivos, como computadoras, tabletas y teléfonos inteligentes; al estar alojadas en servidores, las actualizaciones pueden implementarse de manera centralizada, lo que facilita la entrega de nuevas características y correcciones de seguridad de manera eficiente [6].

4. Herramientas de desarrollo utilizadas

Investigar y definir el concepto de cada una de las tecnologías utilizadas para desarrollar la aplicación (PHP, Laravel, HTML, etc.)

- a. **HTML:** Lenguaje que laravel utiliza junto con un motor de plantillas llamado blade para desarrollar la estructura y contenido que forma la vista de la Aplicación web.
- b. **PHP:** Las funciones de la aplicación web, esto nos ayudará hacer la aplicación dinámica del lado del servidor igual hacer consultas.
- c. **Laravel:** Framework de desarrollo web para php que proporciona herramientas y estructura para la creación de la aplicación web.
- d. **MongoDB:** La base de datos NoSQL MongoDB para almacenar datos o información de manera flexible y escalable ya que se utilizará una aplicación web que requiere una alta escalabilidad y flexibilidad en el modelo de datos.
- e. **Tailwind + vite:** Diseños de las interfaces de los usuarios de la aplicación web con framework de CSS y Vite es una herramienta de compilación de JavaScript rápida y flexible esto nos ayudara para renderizar las interfaces de usuario en la aplicación web.
- f. **Github:** Alojamiento del código fuente esta nos ayudará a guardar código la aplicación web, facilitar la colaboración entre los desarrolladores, revisión de código y despliegue de la aplicación web.

- g. **Composer:** Administrar las bibliotecas y paquetes de terceros en proyectos de PHP. ayuda en la gestión de dependencias y la integración de bibliotecas externas en el proyecto.
- h. **Visual Studio Code:** Desarrollar la aplicación web ya que es un editor de código fuente ya que nos ayudara hacer la lógica, diseño, consultas de la aplicación.
- i. **MySQL:** MySQL es un sistema de gestión de bases de datos relacionales de código abierto ampliamente utilizado. Es una herramienta que permite a los usuarios almacenar, organizar y recuperar datos de manera eficiente mediante consultas estructuradas.

5. Modelado de la aplicación utilizando Diagramas UML

a. Diagramas de Casos de Uso

Proporcionar el concepto y la simbología de los diagramas de Casos de Uso, recordar que cada caso de uso (elipse) representa un RF,

El diagrama de casos de uso es una representación visual de los requisitos funcionales de un sistema desde la perspectiva de los actores que interactúan con él [13].

Simbología:

- **Actor:** representa un rol, persona, u otro sistema que interactúa con el sistema que se está modelando. Se representa como una figura de persona o un símbolo genérico que identifica el tipo de actor. Los actores se colocan fuera del límite del sistema [13].
- **Caso de Uso:** describe una funcionalidad o una acción específica que el sistema debe realizar para cumplir con los objetivos de un actor. Se representa como una elipse dentro del límite del sistema. El nombre del caso de uso describe la acción que se realiza [13].

Relaciones entre Actores y Casos de Uso:

- **Asociación:** Indica que un actor interactúa con un caso de uso específico [13].
- **Inclusión:** Indica que un caso de uso incluye la funcionalidad de otro caso de uso [13].
- **Extensión:** Indica que un caso de uso puede ser extendido por otro caso de uso en ciertas circunstancias [13].
- **Sistema:** representa el sistema que se está modelando. Puede ser una caja o un rectángulo que contiene los casos de uso y los actores [13].

b. Diagramas de secuencia

Proporcionar el concepto y la simbología para elaborar diagramas de secuencia.

El diagrama de secuencia es una representación gráfica que muestra cómo los objetos en un sistema interactúan entre sí en una secuencia específica de eventos. Estos diagramas son útiles para visualizar el flujo de mensajes entre los diferentes componentes o entidades de un sistema a lo largo del tiempo, lo que permite comprender mejor el comportamiento dinámico de un sistema [14].

Simbología utilizada en el Diagrama de Secuencia:

- **Actores:** representados por cajas con la palabra "actor" seguida del nombre del actor. Representan entidades externas al sistema que interactúan con él, como usuarios o sistemas externos [14].
- **Participantes (Objetos):** representados por cajas con el nombre del participante. Representan cualquier objeto o entidad dentro del sistema que participa en la interacción [14].
- **Línea de Vida:** representada por una línea vertical que se extiende desde la parte superior de un participante. Indica la vida útil de un objeto durante una interacción [14].
- **Mensaje:** representado por flechas que van desde el emisor al receptor, etiquetadas con el contenido del mensaje. Indica la comunicación entre objetos y puede ser un mensaje de solicitud, respuesta u otra forma de interacción [14].
- **Activación/Desactivación:** representada por las palabras "activate" y "deactivate" seguidas del nombre del participante. Indica el inicio y el final de la participación de un objeto en una interacción [14].
- **Notas de Activación/Desactivación:** indican el inicio y el final de la participación de un componente en una interacción [14].

- **Restricción de Tiempo:** representada por una línea vertical punteada con una restricción de tiempo. Indica restricciones temporales en una interacción [14].

c. Diagramas de componentes

Proporcionar el concepto y la simbología para elaborar un diagrama de componentes.

El diagrama de componentes es una representación visual de la estructura interna de un sistema de software, mostrando cómo sus partes o componentes están organizados y conectados entre sí. Este tipo de diagrama se centra en los componentes del sistema y sus interacciones, permitiendo comprender la arquitectura del sistema y la distribución de sus elementos [15].

Simbología:

- **Componente:** representa una parte modular y reutilizable del sistema. Puede ser una clase, un módulo, una biblioteca, un archivo o cualquier otro elemento que encapsule funcionalidades relacionadas. Se representa como un rectángulo con el nombre del componente en su interior [15].
- **Interfaz:** define cómo interactúan dos componentes entre sí. Puede ser una interfaz proporcionada por un componente para que otros lo utilicen (interfaz proporcionada) o una interfaz requerida por un componente para funcionar correctamente (interfaz requerida). Se representa como un rectángulo con dos partes: una etiquetada como "Interfaz" y otra como "Nombre", indicando el tipo de interfaz y su nombre respectivamente [15].
- **Dependencia:** indica una relación en la que un componente depende de otro para su funcionamiento. Se representa con una línea punteada con una flecha que va desde el componente que depende hacia el componente del que depende [15].
- **Relaciones de uso:** indican cómo los componentes del sistema utilizan otros componentes o interfaces para llevar a cabo sus funciones [15]. Puede ser relaciones de realización, donde un componente implementa una interfaz, o relaciones de asociación, donde un componente utiliza otro componente.

6. Diseño de la base de datos del sistema

a. Sistema Gestor de Bases de datos utilizado (SGBD)

Explicar el concepto de un SGBD.

Un Sistema Gestor de Base de Datos (SGBD) o DataBase Management System (DBMS) es un sistema que permite la creación, gestión y administración de bases de datos, así como la elección y manejo de las estructuras necesarias para el almacenamiento y búsqueda de información del modo más eficiente posible [7]. Un SGBD es un software que sigue un modelo de sistema de base de datos y, por lo tanto, resulta decisivo a la hora de configurarla, administrarla y utilizarla. Solo cuando el sistema gestor de base de datos está instalado y configurado, los usuarios pueden introducir y consultar los datos. El más conocido de estos lenguajes es SQL (Structured Query Language, lenguaje de consulta estructurada).

Los elementos básicos que lo conforman son tres: el diccionario de datos, el lenguaje de definición de datos y el lenguaje de manipulación de datos.

- **Diccionario de datos:** consiste en una lista de metadatos que reflejan las características de los diversos tipos de datos incluidos en la base de datos [8].
- **Lenguaje de definición de datos:** el lenguaje de definición de datos, también llamado lenguaje de base de datos o DDL (data definition language), sirve para estructurar el contenido de la base de datos [8].
- **Lenguaje de manipulación de datos:** mediante el lenguaje de manipulación de datos o DML (data manipulation language), se pueden introducir nuevos registros en la base de datos, así como eliminar, modificar y consultar los que ya contiene. Este lenguaje también permite comprimir y extraer los datos [8].

Existen multitud de SGBD y pueden ser clasificados según la forma en que administran los datos en:

- **Relacionales (SQL):** Este modelo se basa fundamentalmente en establecer relaciones o vínculos entre los datos, imaginando una tabla aparte por cada relación existente con sus propios registros y atributos [7].

- **No relacionales (NoSQL)**

Se utilizan en entornos distribuidos que han de estar siempre disponibles y operativos y que gestionan un importante volumen de datos [7].

b. Base de datos (BD)

Explicar el concepto de una BD.

Una base de datos (database) almacena datos y los conecta en una unidad lógica junto a los metadatos necesarios para su procesamiento. Las bases de datos son instrumentos de gran utilidad para gestionar grandes ficheros y facilitar la consulta de información. Puede definirse un esquema de permisos que establece qué personas o programas pueden acceder a los datos, y a cuáles, con el objetivo de presentar el contenido de forma adecuada y clara [9].

c. Base de datos relacional

Explicar el concepto de una BD

Una base de datos relacional es un tipo de base de datos que almacena y proporciona acceso a puntos de datos relacionados entre sí. Las bases de datos relacionales se basan en el modelo relacional, una forma intuitiva y directa de representar datos en tablas. En una base de datos relacional, cada fila en una tabla es un registro con una ID única, llamada clave. Las columnas de la tabla contienen los atributos de los datos y cada registro suele tener un valor para cada atributo, lo que simplifica la creación de relaciones entre los puntos de datos [10].

d. Diccionario de la base de datos

Explicar el concepto de un diccionario de datos, y cuáles son sus elementos.

Un diccionario de datos es una colección centralizada de información sobre los datos, como su significado, relaciones, origen, uso y formato. Actúa como un recurso de referencia para el manejo de bases de datos y sistemas de información, facilitando la comprensión y el uso efectivo de los datos dentro de una organización [12].

- **Nombre del Campo:** Identifica de manera única cada elemento de datos (columna en una base de datos) en un sistema [11].
- **Tipo de Dato:** Especifica el tipo de datos (por ejemplo, entero, decimal, texto, fecha) que se espera para el campo, definiendo cómo se puede manipular y almacenar el dato [11].

- **Descripción:** Ofrece una explicación clara del campo, incluyendo para qué se usa y qué representa en el contexto del negocio o la aplicación [11].
- **Formato:** Detalla el formato en que se deben ingresar los datos (por ejemplo, DD/MM/AAAA para fechas), asegurando la consistencia en todo el sistema [11].
- **Restricciones:** Incluye cualquier regla o límite aplicado a los datos, como valores mínimos/máximos, si el campo puede estar vacío (null), y restricciones de unicidad [11].
- **Relaciones:** Describe cómo se relaciona el campo con otros campos o tablas dentro de la base de datos, incluyendo claves foráneas y relaciones entidad-relación [11].
- **Origen de Datos:** Indica la fuente de donde provienen los datos, lo que es crucial para trazabilidad, calidad de datos y procesos de integración.
- **Responsable:** Identifica quién es responsable de gestionar y mantener el campo, lo que puede ser relevante para cuestiones de gobernanza de datos y calidad [11].
- **Uso:** Proporciona detalles sobre cómo se utilizan los datos en el análisis, reportes y otros procesos de negocio, ayudando a los usuarios a entender su importancia y aplicaciones [11].

III. DESARROLLO DEL PROYECTO

Nombre del proyecto: Gestión de Cédulas de Anteproyectos de Estadía

Descripción de la aplicación desarrollada

Explicar el sistema que desarrollaron mencionando los beneficios para alumnos, Asesores académicos, presidente de academia. direcciones de división y asistente de la dirección que proporcionará su implementación en la empresa., recuerden, un párrafo mínimo debe tener cuatro líneas de texto.

El Sistema de Gestión de Estadías ha sido diseñado para optimizar el proceso de revisión de anteproyectos realizados por estudiantes durante su estancia en la empresa. Este sistema ofrece una serie de beneficios para todos los actores involucrados:

Beneficios para los estudiantes:

- **Mejora la comunicación con el asesor:** El sistema facilita la comunicación fluida entre el estudiante y su asesor, permitiendo un seguimiento más cercano del anteproyecto.

Beneficios para los asesores académicos:

- **Herramientas de seguimiento y evaluación:** El sistema ofrece herramientas para el seguimiento del progreso de los estudiantes, la evaluación de sus trabajos y la identificación de áreas de mejora.
- **Apoyo personalizado:** El sistema facilita la comunicación con los estudiantes y permite brindar un apoyo más personalizado a cada uno de ellos; esto con la ayuda de los comentarios.

Beneficios para los presidentes de academia y las direcciones de división:

- **Visión general:** El sistema proporciona una visión integral del desarrollo de las actividades durante el proceso de revisión de anteproyectos.

Beneficios para el asistente de la dirección:

- **Datos precisos y actualizados:** El sistema facilita la gestión de la información, proporcionando datos precisos y actualizados, para el manejo de la documentación requerida.

El Sistema de Gestión de Estadías representa una herramienta innovadora que permite optimizar la gestión de los anteproyectos, brindándoles una experiencia educativa más enriquecedora y personalizada, con la cual se busca obtener la aprobación de una cédula de anteproyecto, que posteriormente, será tratado como proyecto para la elaboración de su respectiva memoria.

1. Diagrama de la aplicación web.

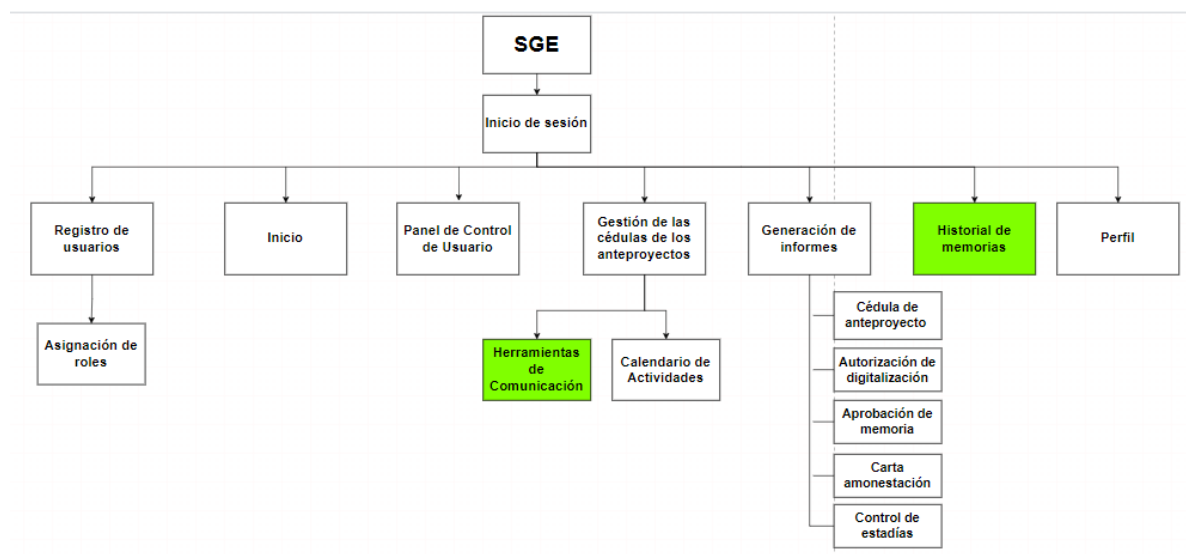


Ilustración 2 Diagrama del SGE hecho por los desarrolladores del grupo SM54

2. Lista de los requerimientos funcionales de la aplicación

Explicar la tabla que se presenta a continuación, numerando ordenadamente todos los RF que forman parte de la aplicación, por favor, sean cuidadosos en la numeración de los RF para que cada equipo documente adecuadamente los que le correspondan

A continuación, se presenta la tabla con los principales requerimientos funcionales del Sistema de Gestión de Estadías, así mismo, con los responsables de su elaboración.

Tabla 1. Lista de Requerimientos funcionales principales de la aplicación

No. RF	Nombre del RF		
RF01	Inicio de sesión:		
	No	Nombre	Autor
	RF01.1	Inicio de sesión	Marco Rocha Cauich

RF02	Registro de usuarios:		
	No	Nombre	Autor
	RF02.1	Registro de usuarios	Jomar Arias Estrella
RF03	Asignación de roles:		
	No	Nombre	Autor
	RF03.1	Gestión de Roles	Marco Rocha Cauich
	RF03.2	Gestión de libros	Marco Rocha Cauich
RF04	Perfil:		
	No	Nombre	Autor
	RF04.1	Perfil estudiante	Jomar Arias Estrella
	RF04.2	Perfil docente	Jomar Arias Estrella
RF05	Panel de control de usuario:		
	No	Nombre	Autor
	RF05.1	Cambiar contraseñas de usuarios	Josias Kumul Quetzal
	RF05.2	Dar de baja usuarios	Josias Kumul Quetzal
	RF05.3	Editar comentarios	Josias Kumul Quetzal
	RF05.4	Eliminar comentarios	Josias Kumul Quetzal
	RF05.5	Generar amonestaciones	Josias Kumul Quetzal
	RF05.6	Permitir que los docentes puedan ver documentos	Josias Kumul Quetzal
	RF05.7	Ver los detalles de los usuarios	Josias Kumul Quetzal
RF06	Gestión de las cédulas de los anteproyectos:		
	No	Nombre	Autor
	RF06.1	Registrar cédula del anteproyecto	José Coronado Cob
	RF06.2	Visualización de la cédula del anteproyecto	José Coronado Cob
	RF06.3	Visualización del estado de la cedula del anteproyecto	José Coronado Cob
	RF06.4	Modificar la información de la cedula del anteproyecto	José Coronado Cob
	RF06.5	Revisión y comentarios de la cédula del anteproyecto	José Coronado Cob
RF07	Generación de informes:		
	No	Nombre	Autor
	RF07.1	Cédula de anteproyecto	Daniel Sánchez Martínez
	RF07.2	Autorización de digitalización	Daniel Sánchez Martínez
	RF07.3	Aprobación de Memoria	Daniel Sánchez Martínez
	RF07.4	Carta Amonestación	Daniel Sánchez Martínez
	RF07.5	Control de Estadías	Daniel Sánchez Martínez

RF08	Calendario de actividades:	
	No	Nombre
	RF08.1	Gestión de fechas de entrega por los asesores
RF09	Herramientas de comunicación:	
	No	Nombre
	RF09.1	Envío de comentario
RF10	Historial de memorias:	
	No	Nombre
	RF10.1	Almacenamiento y visualización de la memoria

3. Definición de requerimientos funcionales asignados al equipo

Leer y adecuar el siguiente párrafo, presentando después del párrafo las tablas correspondientes a los RF de su equipo, explicando lo que van a presentar, recuerden deben numerar, nombrar y referenciar cada tabla.

A continuación, se presentan 2 tablas mediante las cuales se definen los requerimientos funcionales correspondientes al desarrollo del módulo **Herramientas de la comunicación** y al módulo **Historial de memorias** de la aplicación Gestión de las cédulas de Anteproyectos Estadías desarrollada por los alumnos Marco Hau, Olga Moreno, Carlos Valier y Alexis Zamora, con el propósito de definir las funciones que tendrá cada módulo y conocer las transformaciones que el sistema realiza sobre las entradas para que estas puedan producir salidas.

Presentar una tabla por cada RF que forme parte del módulo asignado al equipo. Agregar las que sean necesarias utilizando el siguiente formato:

Tabla 1. Determinación del RF09.1: Envío de comentario

Autor:	Carlos Valier Sánchez
Fecha de creación:	22/marzo/2024
Número de RF:	RF09.1
Nombre de RF:	Envío de comentario
Tipo	<input checked="" type="checkbox"/> RF <input type="checkbox"/> RNF
Actores	Asesor Académico
Prioridad del requerimiento	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/deseable <input type="checkbox"/> Baja/Opcional
Descripción del requerimiento	Requerimiento funcional (RF) que permitirá que el docente pueda escribir comentarios sobre el anteproyecto del estudiante
Precondición:	El asesor académico debe de visualizar el anteproyecto del estudiante para

	evaluarlo
Entradas	Comentario
Proceso	<ol style="list-style-type: none"> 1. El docente inicia sesión. 2. El docente visualiza el anteproyecto del estudiante. 3. El docente realiza el comentario al anteproyecto del estudiante. 4. El docente envía el comentario al estudiante.
Salidas	<ol style="list-style-type: none"> 1. Se envía el comentario al estudiante. 2. El estudiante puede ver el comentario. 3. Se realiza el registro de la comunicación en el apartado del anteproyecto.
Postcondición	El asesor académico envía correctamente el comentario al anteproyecto del estudiante.

Tabla 2. Determinación del RF10.1: Almacenamiento y visualización de la memoria

Autor:	Carlos Valier Sánchez
Fecha de creación:	22/marzo/2024
Número de RF:	RF10.1
Nombre de RF:	Almacenamiento y visualización de la memoria
Tipo	<input checked="" type="checkbox"/> RF <input type="checkbox"/> RNF
Actores	Alumno/Asesor Académico/Administrador
Prioridad del requerimiento	<input type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/deseable <input checked="" type="checkbox"/> Baja/Opcional
Descripción del requerimiento	Requerimiento funcional (RF) que permitirá que el estudiante suba su memoria finalizada al sistema, así mismo, permite que el docente y el administrador pueda visualizar la memoria final del estudiante
Precondición:	El estudiante debe tener finalizada la memoria
Entradas	Documento memoria
Proceso	<ol style="list-style-type: none"> 1. El estudiante inicia sesión. 2. Sube la memoria al SGE. 3. Se almacena la memoria.
Salidas	<ol style="list-style-type: none"> 1. Se guarda la memoria dentro del SGE. 2. El asesor académico y el administrador pueden visualizar la memoria finalizada.
Postcondición	El alumno subió correctamente el documento memoria, así mismo, el asesor académico y el administrador pueden visualizarlo.

4. Desarrollo de Diagramas UML

a) Diagramas de Casos de uso

Numerar, explicar, referenciar y presentar los diagramas de CU de cada módulo desarrollado por el equipo

En la Ilustración 3, se puede observar el caso de uso referente al módulo de herramientas de comunicación, en el cual se visualiza el proceso que realizan los usuarios.

Primero, el asesor inicia sesión, visualiza el anteproyecto, posteriormente, realiza el comentario, envía el comentario y finalmente cierra sesión; De igual manera, el alumno inicia sesión, ve el anteproyecto y revisa si tiene algún comentario, por último, cierra sesión.

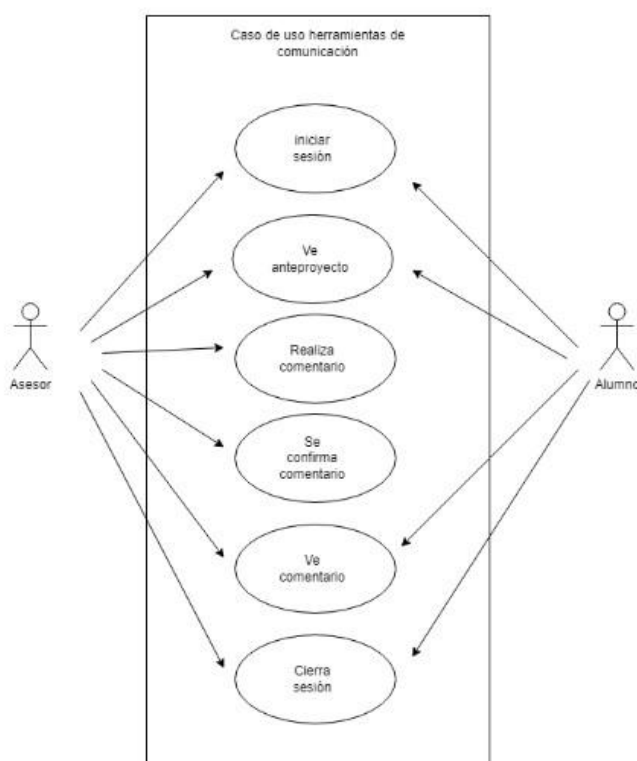


Ilustración 3 Caso de uso del Módulo Herramientas de la comunicación

En la Ilustración 4, se puede observar el caso de uso referente al módulo de historial de memoria, en el cual se visualiza el proceso que realizan los usuarios.

Primero, el alumno inicia sesión, sube su memoria finalizada, posteriormente cierra sesión; por otro lado, el asesor/administrador inicia sesión, visualiza la memoria, y finalmente cierra sesión.

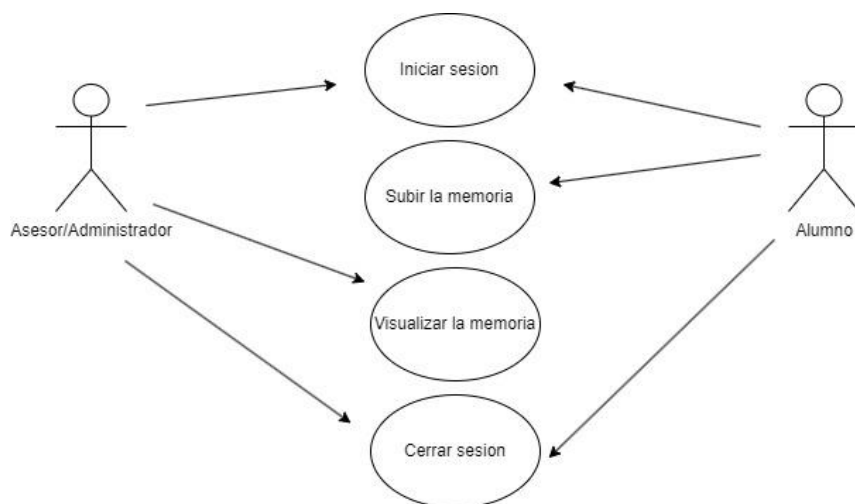


Ilustración 4 Caso de uso del Módulo Historial de Memorias

b) Diagramas de secuencia

Numerar, explicar, referenciar y presentar los diagramas de secuencia de cada diagrama de casos de uso desarrollado por el equipo.

El diagrama de secuencia que se visualiza en la Ilustración 5, describe interacciones entre actores y componentes para la revisión de anteproyectos, el docente inicia sesión en el sistema, visualiza el anteproyecto solicitado, realiza un comentario, y posteriormente, envía el comentario al sistema para ser guardado en la base de datos, se confirma la recepción del comentario, posteriormente, se envía una notificación al estudiante, finalmente, se registra la comunicación entre el asesor y el estudiante.

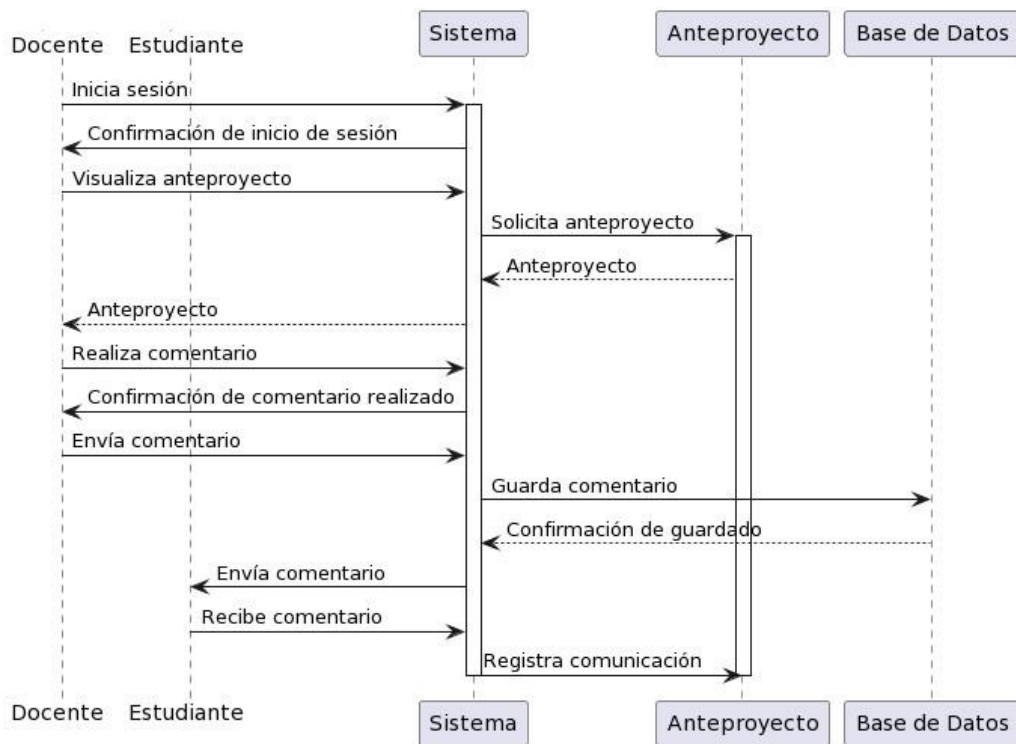


Ilustración 5 Diagrama de secuencia “Herramientas de comunicación”

El diagrama de secuencia que se visualiza en la Ilustración 6, describe interacciones entre actores y componentes para la visualización de las memorias finalizadas por los alumnos. El estudiante inicia sesión, sube la memoria finalizada al sistema, el cual el sistema almacena, para posteriormente, se pueda visualizar la memoria a los asesores y administrador.

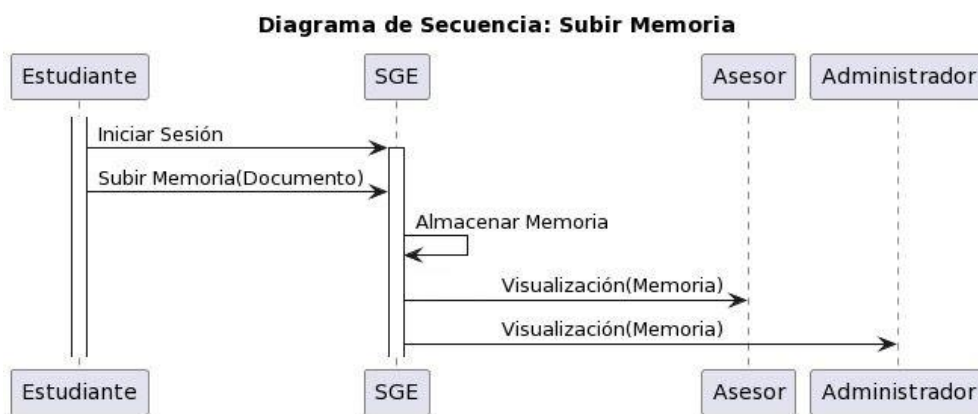


Ilustración 6 Diagrama de secuencia "Historial de memorias"

c) Diagrama de distribución

Numerar, explicar, referenciar y presentar los diagramas de secuencia de cada diagrama de casos de uso desarrollado por el equipo.

El diagrama que se observa en la Ilustración 7, representado por el paquete "Sistema", muestra la estructura de un sistema de comentarios. En él, se identifican tres componentes principales: la interfaz de usuario (UI), el controlador de comentarios (CC) y la base de datos (DB), la interfaz de usuario interactúa con el controlador de comentarios para realizar un comentario, el cual es luego almacenado en la base de datos, para finalmente, recibir una confirmación de que el comentario ha sido enviado.

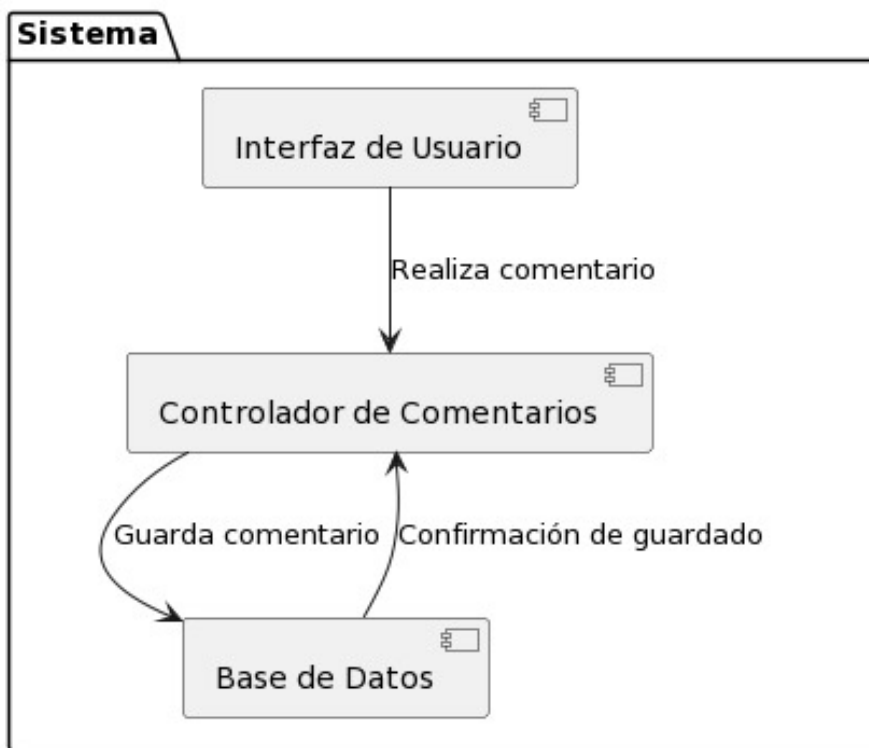


Ilustración 7 Diagrama de distribución "Herramientas de comunicación"

El diagrama que se observa en la Ilustración 8, representado por el paquete "Sistema", muestra la estructura de un sistema memorias, en el cual se podrá subir la memoria y ser visualizado. En él, se identifican 3 componentes: la interfaz del estudiante, el controlador de memorias del SGE y, la interfaz del asesor; la

interfaz del estudiante interactúa con el controlador de memorias, este último, interactúa con la interfaz del asesor, para que la memoria sea visualizada.

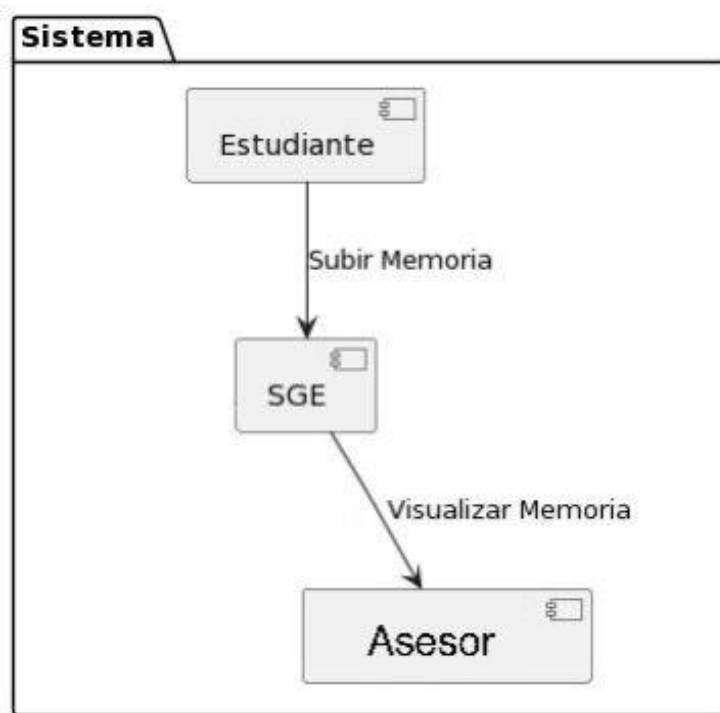


Ilustración 8 Diagrama de distribución "Historial de memorias"

5. Diseño de la Base de datos

a) Descripción de la base de datos

Explicar la base de datos, la información que se administra por medio de ella y mencionar el SGBD que utilizaron para desarrollarla, haciendo referencia a las tablas que la forman y que se describen en la siguiente sección.

MySQL como Sistema Gestor de Bases de Datos (SGBD) para desarrollar la base de datos del sistema es una opción popular y confiable para el desarrollo de bases de datos relacionales.

Mongo es un sistema de base de datos NoSQL que ofrece flexibilidad y escalabilidad para manejar grandes volúmenes de datos no estructurado.

A finalizar el Proyecto, se migrará el sistema gestor de base de datos de MySQL a MongoDB. Esta migración implica cambios en la implementación de la base de datos para adaptarse al modelo de datos de MongoDB.

La base de datos desarrollada para el sistema ha sido diseñada utilizando MySQL como Sistema Gestor de Bases de Datos (SGBD). Esta base de datos ha sido estructurada de manera eficiente para almacenar y gestionar la información necesaria para el funcionamiento del sistema.

A continuación, se hace referencia a las tablas que conforman la base de datos del sistema, las cuales se relacionan mediante el uso de llaves foráneas, así mismo, también se encuentran tablas pivotes que proporciona la unión de datos de datos de diferentes tablas, en una tabla nueva.

- Tabla Estudiante
- Tabla Libros
- Tabla Preguntas_seguridad
- Tabla Memorias
- Tabla reportes
- Tabla empresa
- Tabla tipo_estadias
- Tabla estatus_proyecto
- Tabla colaboradores_proyecto
- Tabla proyecto
- Tabla likes_proyectos
- Tabla asesoramiento_docente
- Tabla carta_amonestacion
- Tabla autorizacion_rol
- Tabla roles
- Tabla usuarios
- Tabla docentes
- Tabla division
- Tabla carrera

- Tabla grupo
- Tabla notificaciones
- Tabla estatus_notificacion
- Tabla Comentarios
- Tabla respuestas
- Tabla historial_actividad

b) Presentación y descripción de la base de datos

Presentar el diagrama relacional de la BD mencionando su nombre, y el número de tablas relacionadas, haciendo referencia al número de la figura.

A continuación, se observa en la Ilustración 9 el diagrama relacional de la base de datos del Sistema de Gestión de Estadías, la cual esta conformada por 25 tablas, quienes se encuentran relacionadas con llaves foráneas, así mismo, cuenta con tablas pivotes, para la agrupación de los datos de dos tablas diferentes en una nueva.

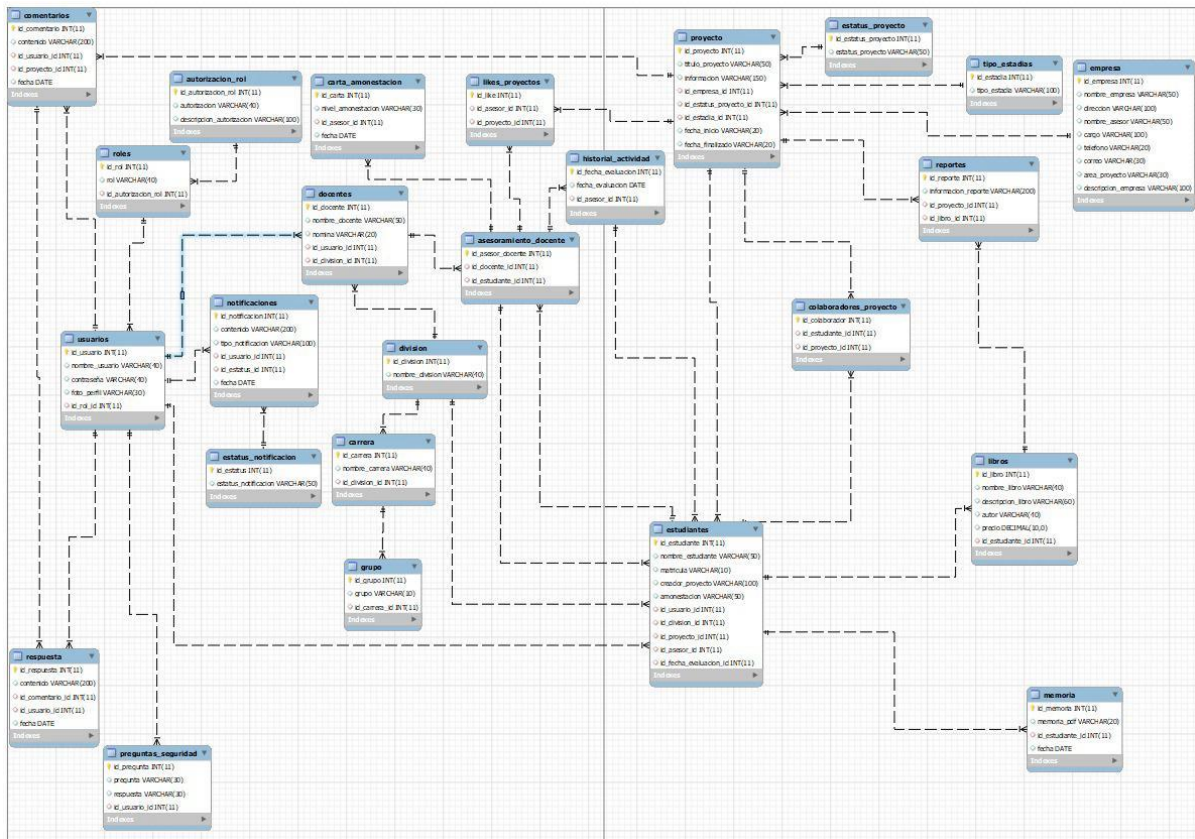


Ilustración 9 Diagrama relacional de la base de datos "sge"

c) Presentación y descripción del diccionario de datos

Presentar el diccionario de la BD mencionando el nombre, el número y cada tabla presentada en el diagrama relacional, haciendo referencia al número de la figura.

Tipos de datos: Determinar qué tipo de datos de almacenarán son INT, VARCHAR, DATE, TEXT NUMBER.

Clave Primaria (Primary key): Especifica que la columna es la clave primaria de la tabla

Auto incremento (Auto_increment): especifica para las columnas que incrementen automáticamente cada inserción.

A continuación, se observa el nombre de las tablas, que forman parte de la base de datos que se presenta en la Ilustración 9.

- Claves Foráneas (Foreign): Establece una relación con una columna en otra tabla.

- Tabla Estudiante
- Tabla Libros
- Tabla Preguntas_seguridad
- Tabla Memorias
- Tabla reportes
- Tabla empresa
- Tabla tipo_estadias
- Tabla estatus_proyecto
- Tabla colaboradores_proyecto
- Tabla proyecto
- Tabla likes_proyectos
- Tabla asesoramiento_docente
- Tabla carta_amonestacion
- Tabla autorizacion_rol
- Tabla roles
- Tabla usuarios
- Tabla docentes
- Tabla division
- Tabla carrera
- Tabla grupo
- Tabla notificaciones
- Tabla estatus_notificacion
- Tabla Comentarios
- Tabla respuestas
- Tabla historial_actividad

IV.RESULTADOS O PRODUCTOS OBTENIDOS

Presentación y descripción de las ventanas de los módulos desarrollados por cada equipo.

Módulo: Historial de Memorias

A continuación, se presenta la descripción de las ventanas correspondientes al módulo de Historial de Memorias desarrollado, el cual permite que los docentes puedan acceder y revisar las memorias finalizadas por los alumnos. A través de una interfaz intuitiva, los docentes pueden buscar, visualizar y descargar las memorias almacenadas en el sistema.

En la Ilustración 10, se presenta la ventana correspondiente al módulo Historial de Memorias, mediante el cual los docentes pueden buscar entre las memorias finalizadas. Se incluyen opciones para buscar por título. La ventana muestra los resultados de la búsqueda de manera clara y organizada, facilitando la localización de memorias específicas.

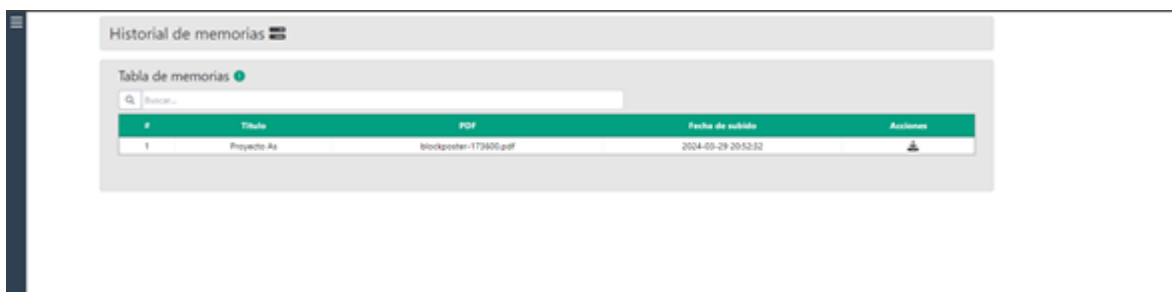


Ilustración 10 Módulo de Historial de Memorias

Módulo: Herramientas de comunicación.

A continuación, se presenta la descripción de las ventanas correspondientes al módulo de Herramientas de comunicación desarrollado, el cual permite que los docentes puedan realizar comentarios a los anteproyectos de los alumnos. A través de una interfaz que está diseñada para proporcionar una representación clara y completa del anteproyecto, y así mismo agregar comentarios para su mejora, permitiendo su gestión eficiente.

En la Ilustración 11, se presenta la ventana correspondiente al módulo Herramientas de comunicación, la cual muestra una vista general del anteproyecto, incluyendo detalles como el título, el autor, la fecha de creación, y el

estado actual del proyecto. Desde aquí, los docentes pueden revisar el anteproyecto y realizar un comentario referente al anteproyecto de los alumnos.

Consulta individual de anteproyecto

Información de anteproyecto

CÉDULA DE ANTEPROYECTO DE ESTADÍA

División: Ingeniería y Tecnología

Programa educativo: TSU en TI Área Desarrollo de Software Multiplataforma

Título del anteproyecto: Reto Tokio

Nombre del alumno: Daniela Morantes Ricalde

Matrícula: 21393143 Grupo: SM54

Teléfono: 9988235671 Correo electrónico: 21393143@utcanuncun.edu.mx

Fecha de inicio del proyecto: 19/05/2024 Fecha de término del proyecto: 19/07/2024

Empresa: Uprify

Dirección: Calle Robalo 72, 77500 Cancún, Q.R.

Nombre del asesor Empresarial: Diego Horacio Dorantes

Cargo: Gerente

Teléfono: 9988233459 Correo electrónico: uprify23@gmail.com

Comentarios de revisión:

Información relevante

Poner información como:

- Aceptado por asesor?
- Colaborativo
- Likes

¿Qué más?

No se que más podría ir aquí

Ilustración 11 Módulo de herramientas de comunicación

Explicar si se lograron los objetivos planteados

Con respecto a los objetivos que se establecieron al desarrollar el módulo de Herramienta de comunicación, se logró generar el envío de los comentarios por parte del asesor académico, y este sea visualizado por el alumno; por otro parte, los objetivos que se establecieron para el módulo de Historial de memoria, aún no se cumple con todos, pero, no obstante, se ha generado un avance del 80% de su funcionalidad con respecto a los objetivos planteados.

CRONOGRAMA DE ACTIVIDADES																			
NOMBRE DEL PROYECTO		Sistema de Gestión de Estadías																	
OBJETIVO GENERAL		Innovar el Sistema de Gestión de Estadías profesionales de los estudiantes que cursen 6to y 11vo cuatrimestre, esto para facilitar los procesos que conlleva la realización de la Cédula de Anteproyecto. Haciendo uso de tecnologías como los es Laravel, MySQL y MongoDB, de igual manera, utilizando PHP y JavaScript como lenguajes de programación con las cuales se realiza una mejora de forma exponencial y adaptativa del sistema																	
PERIODO DE REALIZACIÓN		Enero - abril/2024																	
OBJETIVOS ESPECÍFICOS	ACTIVIDADES		Enero				Febrero				Marzo				Abril				
			1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
Diseñar las técnicas de recolección de requerimientos	Entrevista	P																	
		R																	
	Encuesta	P																	
		R																	
Especificar los requerimientos del Sistema de Gestión de Estadías	Funcionales	P																	
		R																	
	No funcionales	P																	
		R																	
Ejercer la metodología de desarrollo Design Thinking	Aplicar técnicas de recolección	P																	
		R																	
	Definir	P																	
		R																	
	Modelado UML	P																	
		R																	
	Diseño de interfaces	P																	
		R																	
	Diseño de la base de datos	P																	
		R																	
	Desarrollo del SGE	P																	
		R																	
	Probar	P																	
		R																	
	Entrega	P																	
		R																	
PROGRAMADO	P																		
REAL	R																		

V. CONCLUSIONES

Durante el transcurso del desarrollo de este proyecto, hemos comprendido la importancia que conlleva tener una buena organización en el cumplir con cada actividad que el cronograma señala, así mismo, el trabajo en conjunto de todos los equipos de programación, se ha retroalimentado, para mejorar la comunicación que debe de existir en proyectos similares o de mayor grado al actual.

Conforme se va desarrollando el sistema, se logro adquirir conocimientos sobre un nuevo framework, en este caso Laravel, así mismo, se ha ido retroalimentando los conocimientos sobre los lenguajes de programación que utiliza, como lo es PHP y JavaScript. Al trabajar con Laravel, y como ya se ha observado, se hace uso del paradigma de la Programación Orientada a Objetos (POO), la cual nos ayuda a tener mejor control al asignar roles a los usuarios, al mismo tiempo, asignarles atributos que cada uno manejará.

Al trabajar en un proyecto del tamaño como lo es el Sistema de Gestión de Estadías, que será utilizado por la división de Ingeniería de la Universidad Tecnológica de Cancún, Laravel nos ha proporcionado una base de mucha ayuda, de igual manera, herramientas útiles, que nos facilita la programación con su arquitectura de software MVC.

BIBLIOGRAFÍA

- [1] C. S. A. S. M. P. Walter A. Lucero, «Fiabilidad en la calidad del software: Modelos, Métodos y Estrategias,» *CORE*, p. 5.
- [2] C. A. Guerra, «SG,» SG#17, [En línea]. Available: <https://sg.com.mx/revista/17/obtencion-requerimientos-tecnicas-y-estrategia>.
- [3] «PMOinformatica.com,» PMOinformatica.com, 03 08 2016. [En línea]. Available: <https://www.pmoinformatica.com/2016/08/tecnicas-levantamiento-requerimientos.html>.
- [4] G. R. Madrigal, «MANUAL DE DESIGN THINKING,» [En línea]. Available: chrome-extension://efaidnbmnnnibpcajpcgclefindmkaj/http://www.utsc.edu.mx/vidaEstudiantil/pdf/pdf_pades/manual_design_thinking.pdf.
- [5] Amazon, «AWS,» [En línea]. Available: <https://aws.amazon.com/es/what-is/web-application/>.
- [6] «digital guide,» ionos, [En línea]. Available: <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/que-es-una-web-app-y-que-clases-hay/>. [Último acceso: 21 03 2024].
- [7] «INESEM,» BUSINESS SCHOOL, [En línea]. Available: <https://www.inesem.es/revistadigital/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/>. [Último acceso: 26 03 2024].
- [8] «Digital Guide,» IONOS, [En línea]. Available: <https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/sistema-gestor-de-base-de-datos-sgbd/>. [Último acceso: 26 03 2024].

- [9] «Digital guide,» IONOS, [En línea]. Available:
<https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/bases-de-datos/>. [Último acceso: 28 03 2024].
- [10] «oracle,» linux, [En línea]. Available:
<https://www.oracle.com/mx/database/what-is-a-relational-database/>. [Último acceso: 28 03 2024].
- [11] «IBM,» [En línea]. Available:
<https://www.ibm.com/docs/es/networkmanager/4.2.0?topic=database-data-dictionary>. [Último acceso: 28 03 2024].
- [12] «Conecta magazine,» webinar, [En línea]. Available:
<https://www.conectasoftware.com/magazine/glosario/diccionario-de-datos/>. [Último acceso: 28 03 2024].
- [13] «creately,» Cinergix Pty. Ltd, [En línea]. Available:
<https://creately.com/es/diagram-type/casos-uso/>. [Último acceso: 30 03 2024].
- [14] «creately,» Cinergix Pty. Ltd, [En línea]. Available:
<https://creately.com/es/diagram-type/diagrama-secuencia/>. [Último acceso: 30 03 2024].
- [15] «Crear un diagrama de componentes UML,» Microsoft, [En línea]. Available:
<https://support.microsoft.com/es-es/topic/crear-un-diagrama-de-componentes-uml-aa924ecb-e4d2-4172-976e-a78fa157b074>. [Último acceso: 30 03 2024].