INTRODUCCIÓN

ghp\_ZksZYEyciPNnEl3XlkkZzSmTE9xrBy1qdkEJ

Para el desarrollo de la aplicación una aplicación que facilite la colaboración remota entre equipos. Puedes incorporar funciones de chat, compartición de documentos, y seguimiento de tareas, que tenga un IDE integrado para programación, se va a realizar para grupos de desarrollo donde internamente tenemos que reunirnos de manera eficiente y por proyecto, para una modalidad de tener un cronograma de tareas, asignación y cumplimiento del equipo, hacer procesos de retroalimentación y en llegados casos presentaciones de código para organizar commitar y demás, por tal motivo vamos a realizar las siguientes actividades.

MARCO TEÓRICO

acceso del usuario:

Se debe crear una cuenta desde el admin para la inicialización del proyecto para el ingreso, de ahí para el grupo de desarrolladores se va a crear una instancia donde las personas que ingresen con su correo electrónico se le entregara una contraseña; la cual debe cambiar y se le da acceso a su proyecto, ya que se diseñara por proyecto cada instancia.

Acceso al repositorio:

Se entregará el espacio del repositorio donde se va a manejar todo el versiona miento del proyecto.

Videoconferencias:

Un espacio donde se puede crear reuniones de videoconferencias con compartición de pantalla integrado y se puede generar un banco de ideas y nuevas formas de hacer las tareas

Centro de tareas:

Un centro donde está el porcentaje de avance del proyecto están las tareas a realizar y un tiempo de cada tarea con opción de alargue con observaciones y permisos adquiridos, un diagrama de como se debe iniciar el proyecto en proceso y un diagrama de la ejecucion, ademas del centro de nombres de quien es el responsables y los cambios que se pueden verificar el proceso.

El IDE:

Un IDE que está integrado con los plugins necesarios para poder realizar las actividades según las necesidades del proyecto.

Bloc de notas:

Un bloc de notas para realizar notas que pueden ser privadas o públicas de suugerencias

Centro de mensajes:

El centro de mensajes es un chat que se realiza con un grupo donde estarán todos los responsables y el personal adscrito al proyecto, además de poder crear grupos extras con compañeros, y chats privados con demás miembros del proyecto; también se pueden compartir documentos.

Centro de correo:

Un centro de correos respecto al proyecto y que se conecta con el listado de correos de todo el equipo, donde también se puede hacer envío de documentos a otros usuarios

## EXPLICACIÓN

Para el \*\*informe de entregables\*\* de tu proyecto de desarrollo de software, aquí tienes una estructura que puedes seguir, junto con ejemplos de entregables y consideraciones relevantes. Esto te ayudará a organizar y presentar de manera efectiva la información sobre tu aplicación de comunicación remota y gestión de documentos.

Introducción

- Breve descripción del proyecto y su objetivo.

- Importancia de los entregables en el proceso de desarrollo de software.

Características del Software

Identifica y describe las características clave que tu aplicación debe tener, como por ejemplo:

- \*\*Comunicación Remota\*\*: Funcionalidades de chat y videoconferencia.

- \*\*Compartición de Documentos\*\*: Opciones para subir, compartir y editar documentos en tiempo real.

- \*\*Gestión de Tareas\*\*: Funcionalidades para crear, asignar y seguir el progreso de tareas en proyectos.

- \*\*Desarrollo Colaborativo\*\*: Herramientas para permitir la colaboración en código y documentación entre desarrolladores.

### Plataformas de Desarrollo

Define las plataformas que usarás para el desarrollo, considerando las siguientes opciones:

- \*\*Lenguajes de Programación\*\*: Por ejemplo, JavaScript (para frontend y backend),Python (para backend), etc.

- \*\*Frameworks y Librerías\*\*: Como React o Angular para el frontend, Node.js para el backend.

- \*\*Base de Datos\*\*: MariaDB para almacenamiento de datos.

- \*\*Entornos de Desarrollo\*\*: IDEs como Neovim o VSCode.

Entregables del Proyecto

Aquí hay una lista de los entregables que puedes incluir, junto con descripciones y estados posibles:

|  |  |  |  |
| --- | --- | --- | --- |
| ***Entregable*** | ***Descripción*** | ***Estado Posible*** | ***Fuente*** |
| **Especificación de Requisitos** | Documento que detalla los requisitos funcionales y no funcionales del software. | | Borrador, Revisión, Aprobado | Reuniones con usuarios y análisis de requerimientos |
| **Diseño de Software** | | Diagramas de arquitectura, diagramas de clases y diagramas de flujo que representan el diseño del software. | Borrador, Revisión, Aprobado | Herramientas de diseño (como UML) |
| **Registro de Trazabilidad** | Matriz que vincula los requisitos a los casos de prueba y entrega del software. | Completo, En Progreso | Análisis de requisitos y pruebas |
| **Prototipo de Interfaz de Usuario** | Mockups o prototipos que muestran la interfaz de usuario y la experiencia de usuario | Borrador, Revisión, Aprobado | Herramientas de diseño (como Figma) |
| **Código Fuente** | Código que implementa las funcionalidades del software según las especificaciones y el diseño. | En Desarrollo, Revisión, Completo | Repositorio de GitHub u otro control de versiones |
| **Documentación del Usuario** | Guía de usuario y manual de instalación para ayudar a los usuarios a entender el software. | Borrador, Revisión, Completo | Manuales y tutoriales |
| **Plan de Pruebas** | Documento que describe cómo se probarán las funcionalidades y la calidad del software. | Borrador, Revisión, Aprobado | Análisis de requisitos y pruebas |
| **Informe Final del Proyecto** | Resumen de todo el proceso de desarrollo, resultados y lecciones aprendidas. | Borrador, Revisión, Completo | Revisión de todo el proyecto |

### Consideraciones Finales

- Asegúrate de establecer un \*\*cronograma de entrega\*\* para cada uno de los entregables.

- Considera la \*\*revisión y validación\*\* de cada entregable con partes interesadas y usuarios finales para garantizar que se alineen con las expectativas y necesidades.

### Conclusión

Un resumen de la importancia de seguir este enfoque estructurado para el desarrollo de software, asegurando que se aborden las necesidades del usuario y se mantenga la calidad del producto.

### Ejemplo de Entregable: Especificación de Requisitos

Puedes detallar algunos requisitos específicos, como:

- El sistema debe permitir la creación de grupos de chat.

- Los usuarios deben poder subir y compartir documentos con un límite de tamaño de 10 MB.

- Debe haber un sistema de notificación para las tareas asignadas.

Con esta estructura, puedes desarrollar un informe completo que aborde todos los aspectos importantes del proyecto. Si necesitas más detalles sobre algún punto en específico, ¡no dudes en preguntar!