



Informe de estado del proyecto

VitaBox Software

Resumen del proyecto

FECHA DEL INFORME	NOMBRE DEL PROYECTO	PREPARADO POR
06/10/2024	ARETEO	Irving Geyler Cupul UC

Resumen del estado

El proyecto sufrió un importante retraso debido a problemas en el almacenamiento de la información en la base de datos. Se ha continuado con el ajuste en el modulo de preguntas para la creación de instrumentos, completado el diseño de la base de datos, se ha logrado el convertir nodos de la aplicación a imágenes para la exportación de graficas, se ha implementando una nueva biblioteca de graficas , el registro y acceso al administrador y lograr proteger las entrada a menos de iniciar sesión. Actualmente, el equipo está trabajando en la exportación de reportes como PDF y en formato Docx., el registro de pacientes. El progreso ha sido bastante corto con respecto al mes anterior, debido a los problemas con establecer y gestionar la base de datos al momento de la compilación de la aplicación lo que habría incentivado a una migración demasiado grande lograr nuestros objetivos.

Información general del proyecto

Tarea:

Realización del clon de formularios con las siguientes funcionalidades:

- Agregar secciones a los formularios

% **Hecho:** 40 por ciento

Fecha de vencimiento: 15 de octubre de 2024

Conductor: Joar Honorio Ruiz Peraza

Retrasos

- Aún no se han aprobado las últimas correcciones y añadidos en los tipos de preguntas del formulario. Este apartado sigue en revisión por parte del líder del proyecto.
- Aun siguen trabajando en la parte de guardado de las preguntas.

Complicaciones

- Necesidad de reestructurar los archivos de tipos de preguntas en la carpeta "Formulario" para cumplir con los nuevos requerimientos del proyecto.
- Adaptación a nuevos archivos de código para agregar más opciones de tipos de preguntas, lo que implicó ajustes adicionales en la organización del código.

Logros

- Satisfactoria implementación de nuevos tipos de preguntas requeridas por el proyecto.

Tarea:

Creación de tipos según las tablas de la base de datos del sistema

% Hecho: 100 por ciento

Fecha de vencimiento: 1 de octubre de 2024

Conductor: Jesus Alejandro Be Hau

Retrasos

- Debido a cuestiones de tiempo, se ha visto un retraso de 1 día de trabajo.

Complicaciones

- Ninguna, a pensar del cambio repentino de las definiciones en las IDs de las tablas.

Logros

- Implementar los tipos por tablas al lenguaje de programación Typescript para agilizar el desarrollo de la aplicación.

Tarea:

Desarrollar la funcionalidad de guardar archivos en Docx de la aplicación.

- Implementación de la librería
- Desarrollo de la funcionalidad, trabajando con la lógica básica del código.

% Hecho: 25 por ciento

Fecha de vencimiento: 12 de octubre de 2024

Conductor: Didier Andrey Tec Esquivel y Juan Carlos Conde Marrufo

Retrasos

- Aun no se completan todos los casos de prueba para el uso de la librería.

Complicaciones

- Dificultades técnicas al intentar reflejar en la aplicación la funcionalidad de descargar un archivo Docx. Esto ha implicado revisiones adicionales en la estructura del código.

Logros

- Implementación de la librería .
- Reconocer dificultades y necesidades de esta parte del proyecto, optando por modificar y ajustar el código, así como la incorporación de un nuevo miembro.

Tarea:

Creación de registro y acceso de administrador

- Protección de rutas
- Seguridad al registrar los datos del administrador

% Hecho: 100 por ciento

Fecha de vencimiento: 4 de octubre de 2024

Conductor: Irving Geyler Cupul Uc

Retrasos

- No se encontraba la respuesta a la base de datos, provocando un retraso de una semana.
- A la espera de que se subieran los tipos de la tabla administrador para agilizar el desarrollo.

Complicaciones

- Durante la protección de las ventanas del dashboard, esta tomo mas tiempo debido al manejo de la sesión.

Logros

- Se completo el registro e inicio de sesión del administrador.
- Se logro la protección a las ventanas del dashboard, si no ha iniciado sesión.
- Se protegió las credenciales del administrador.

Incidentes:

1. Durante la implementación de las tablas creadas en SQL para la base de datos relacional que funcionaría de manera local al instalar la aplicación de escritorio usando SQLite, surgió el inconveniente de que no se podía hacer ningún registro y validación de la existencia de la base de datos, aun al seguir la documentación oficial para el desarrollo con estas características. Acudiendo a diversos desarrollos similares con la misma tecnología, no dio resultados, por lo que mediados de septiembre se decidió por migrar todo el desarrollo de Tauri hecha en Rust, a Electrón hecha con JavaScript y usando Nodejs para convertir aplicaciones web a aplicaciones de escritorio.
2. La documentación para comenzar la migración a Electrón fue mas confusa que la de Tauri, debido a que esta utilizaba conceptos bastante antiguas a las implementaciones modernas de EcmaScript, y el buscar plantillas que fueran compatibles a nuestras librerías usadas para crear la aplicación, las cuales nunca dieron problemas con Tauri, ya que esta estaba construida específicamente para solventar los problemas de Electrón, pero la documentación y los ejemplos no fueron suficientes. Esto provoco que luego de una búsqueda al fin encontrar una plantilla que se ajustara en medida para usar nuestras librerías.
3. El problema fue el mismo, pero mas grave con Electrón a la hora de la creación con la base de datos SQLite, que los problemas de compatibilidad con el uso de estándares distintos al usar JavaScript y el acceso de permisos para la modificación de archivos en los dispositivos de escritorio, lo volvió aun mas complicado, provocando así hacer un ultimo intento por implementar la aplicación con Tauri.
4. Los resultados esta segunda vez dieron buenos resultados a encontrar un proyecto open source que implementaba de manera sencilla el uso y acceso a una base de datos relacional con SQLite de manera local, lo cual nos permitió reconocer lo que nos faltaba para lograr utilizar nuestra base de datos, y no hacer una migración de gran escala a estas alturas.

Plan para la siguiente semana

El proyecto presenta un contratiempo demasiado grave, por lo que todas las fechas han sufrido retrasos significativos, por lo que se ha decidido que los tareas sean asignadas a dos personas, que una, para aprovechar una rápida retroalimentación. Esta semana se busca completar la gestión de los pacientes, y el uso de la librería para crear documentos Docx, y para la próxima semana trabajar todo el equipo en la implementación y pruebas del modulo de la creación de formularios y al mismo tiempo realizar las pruebas.