**Proyecto StraviaTEC**

Área de Ingeniería en Computadores

Instituto Tecnológico de Costa Rica

CE 3101 - Bases de datos

**Estudiantes**

Allan Josué Calderón Quirós - 2018114634

Ronny Josué Santamaria Vargas - 2018109283

Antony Fabián Fallas Elizondo - 2018178906

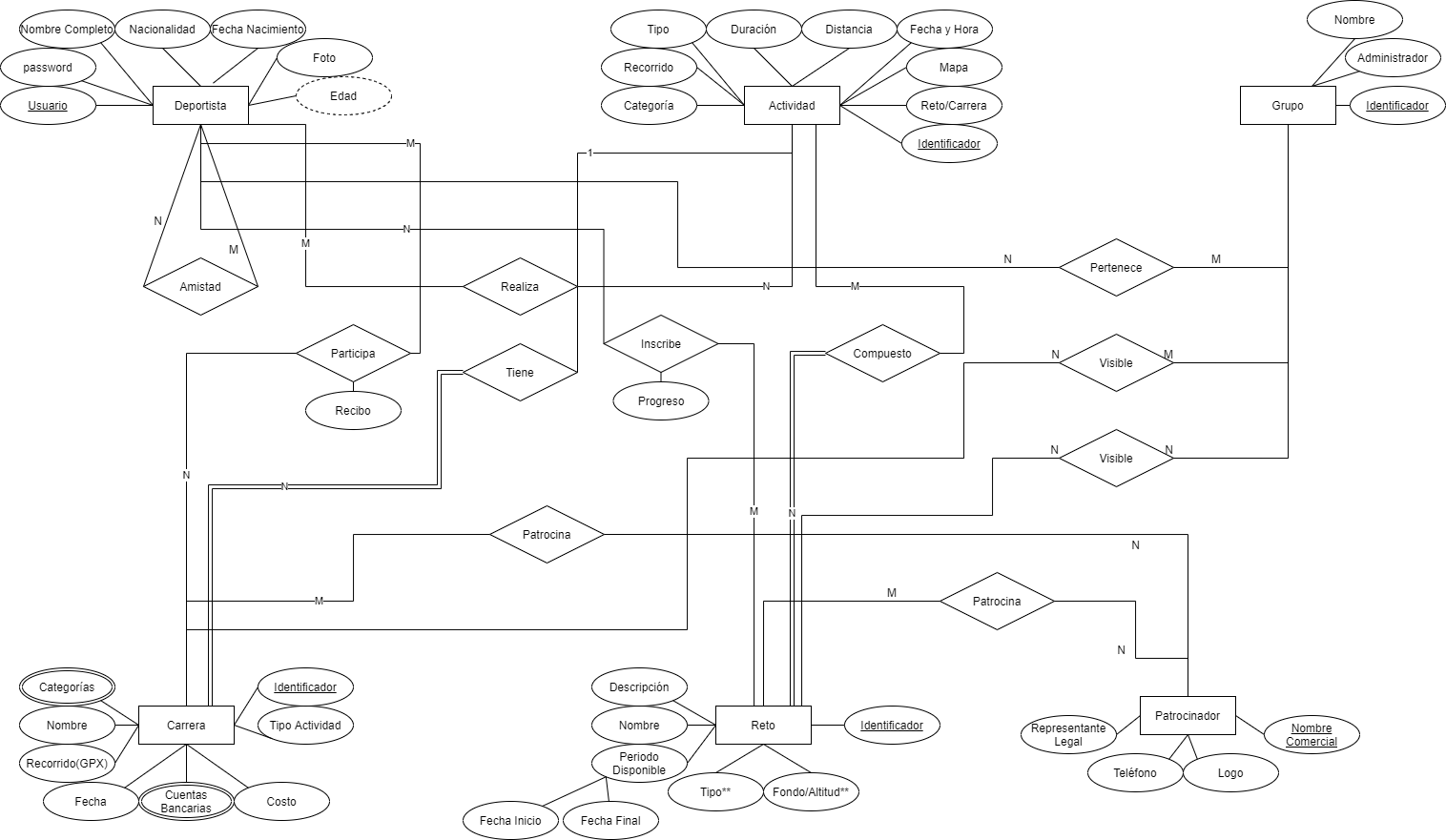
Edgar Alberto Gonzalez Chacon - 2018156158

**Profesor**

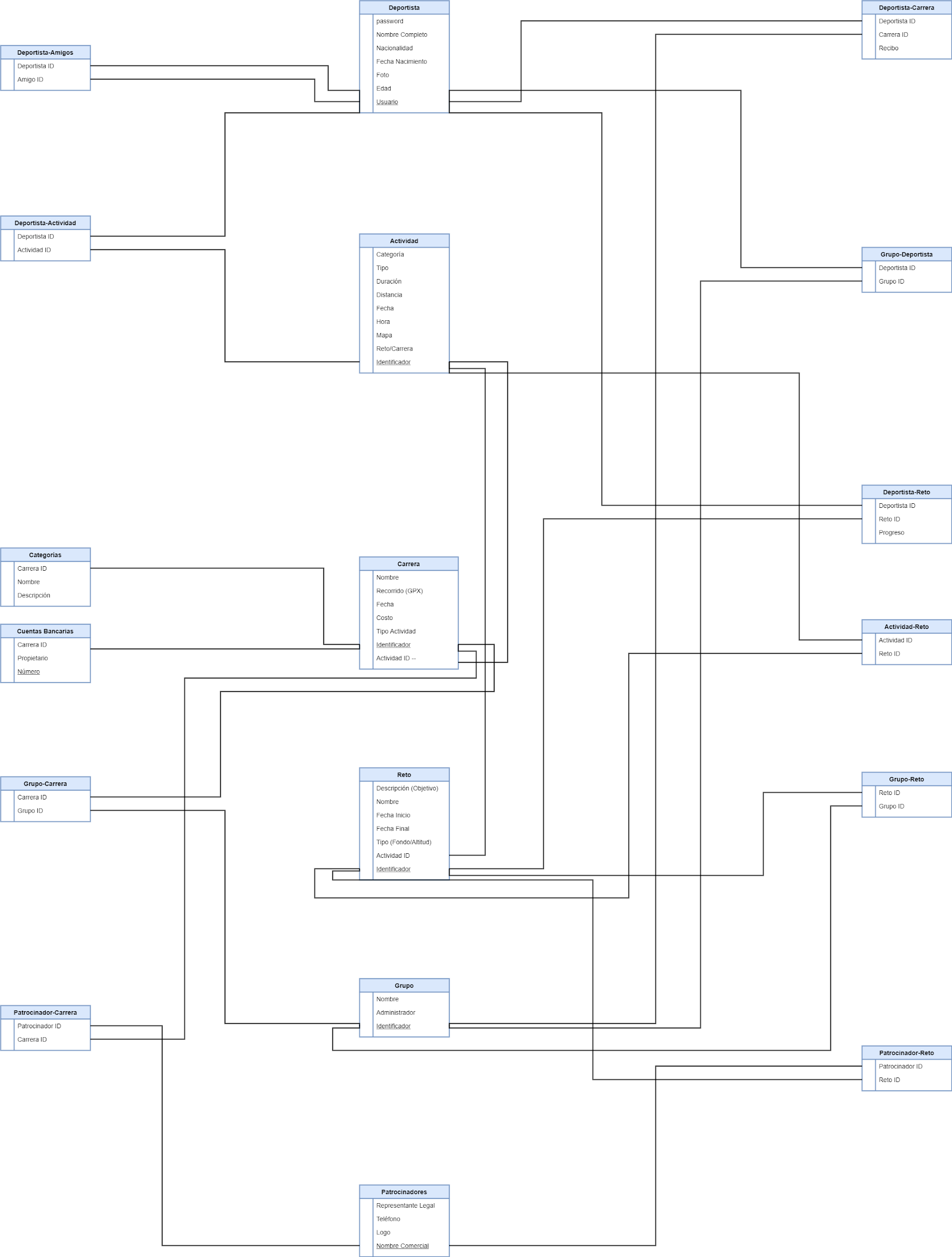
Marco Rivera Meneses

II semestre, 2020.

Modelo conceptual utilizando la notación de Chen.



Modelo relacional.



Descripción de las estructuras de datos desarrolladas (Tablas).

**Para las tablas creadas en Postgresql**

**ATHLETE:** Almacena los datos de los deportistas que contendrá los datos del nombre, contraseña, nacionalidad, fecha de nacimiento, foto, edad y usuario (atributo llave).

**SPONSOR:** Contiene los datos de representante legal, número de teléfono, logo y nombre comercial de la tabla de patrocinadores (atributo llave).

**ACTIVITY:** Almacena los datos referentes a la tabla actividad. Esta tabla contiene el tipo de actividad, si es de fondo o altitud, la duración de la actividad, la distancia determinada para la actividad, la fecha de la actividad, el mapa que debe contener la ruta a un archivo .gpx, si la actividad pertenece a una carrera o a un reto y el identificador de la actividad (atributo llave).

**RACE:** Almacena los datos de la tabla carrera. Esta tabla contiene nombre de la carrera, el trayecto de la carrera, la fecha en que se realizó la carrera, costo de inscripción, el tipo de actividad que compone la carrera, el identificador de la carrera (atributo llave) y el identificador de la actividad que compone la carrera (llave foránea).

**BANK\_ACCOUNT:** Tabla creada con el fin de mantener los datos de las cuentas bancarias vinculadas una carrera. Esta tabla contiene: identificador de la carrera con la que se relaciona (llave foránea), el dueño de la cuenta bancaria y el número de cuenta (atributo llave).

**CATEGORY:** Esta tabla contiene los valores de la tabla de categorías que se relaciona con las carreras. La categoría contiene: identificador de la carrera (llave foránea), nombre de la categoría (atributo llave) y una descripción de la categoría.

**CHALLENGE:** La tabla contiene los datos de los retos que sean creados. Entre los valores de esta tabla se encuentran: descripción del reto, nombre del reto, fecha de inicio, fecha final, tipo de reto, identificador de actividad que compone el reto (llave foránea) y el identificador del reto (atributo llave).

**TEAM:** Tabla que contiene los equipos, entre la información que se almacena en esta tabla está: el nombre del equipo, el administrador y el identificador del equipo (atributo llave).

**SPONSOR\_RACE:** Representa la relación entre los patrocinadores y las carreras, en este caso sponsor\_id referencia al patrocinador y race\_id referencia a la carrera que patrocina.

**SPONSOR\_CHALLENGE:** Representa la relación entre los patrocinadores y los retos, en este caso sponsor\_id referencia al patrocinador y challenge\_id referencia al reto que patrocina.

**TEAM\_RACE:** Representa la relación entre las carreras y los equipos a los que se puede mostrar la carrera, en este caso team\_id referencia al equipo y race\_id referencia a la carrera que patrocina.

**TEAM\_CHALLENGE:** Representa la relación entre los retos y los equipos a los que se puede mostrar el reto, en este caso team\_id referencia al equipo y challenge\_id referencia al reto que patrocina.

**TEAM\_ATHLETE:** Representa la relación de pertenencia de un deportista a un equipo, en este caso team\_id referencia al equipo al que pertenece el deportista y athlete\_id referencia al deportista que está dentro del equipo.

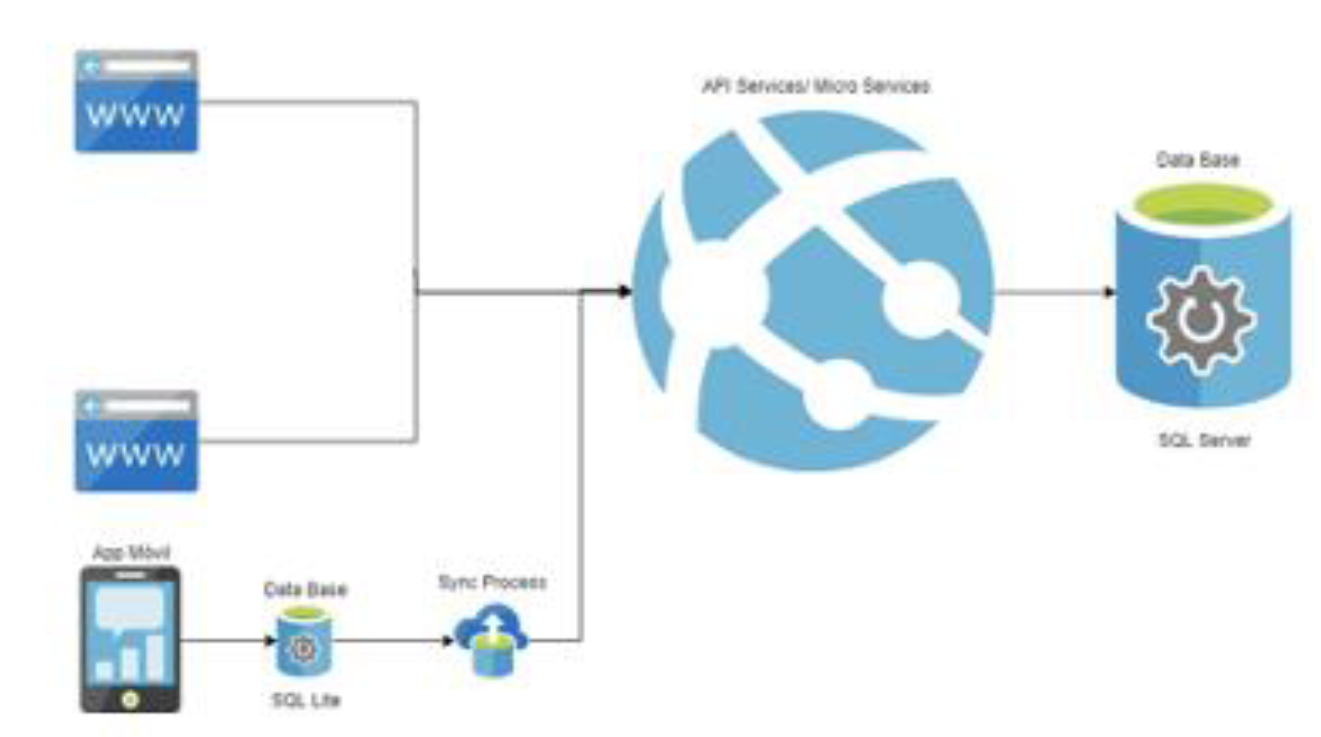
**ATHLETE\_ACTIVITY:** Representa la relación entre un deportista que realiza una actividad. Los valores son athlete\_id que hace referencia al deportista y activity\_id que hace referencia a la actividad realizada.

**ATHLETE\_CHALLENGE:** Representa la relación entre un deportista que realiza un reto. Los valores son athlete\_id que hace referencia al deportista, challenge\_id que hace referencia al reto que realiza y .progress que significa cuánto porcentaje del reto ha completado.

**ATHLETE\_RACE:** Representa la relación entre un deportista que participa en una carrera. Los valores son athlete\_id que hace referencia al deportista, race\_id que hace referencia a la carrera en la que participa y .payment\_receipt que contiene el recibo del pago de la inscripción a la carrera.

**ATHLETE\_FRIEND:** Representa la relación de amigos donde athlete\_id hace referencia a un deportista y friend\_id toma el valor de otro deportista que se desee tener como amigo.

**ACTIVITY\_CHALLENGE:** Esta tabla contiene los valores que representan la relación entre las actividades y los retos, el valor activity\_id hace referencia a la actividad que compone el reto y challenge\_id referencia al reto que está compuesto por la actividad.

Descripción detallada de la arquitectura desarrollada. 

**Base de datos (Postgresql):**

Esta parte de la arquitectura es la encargada de almacenar las tablas y los datos mencionados en el apartado anterior. Recibe solicitudes desde el backend y retorna los datos necesarios para cumplir con las operaciones que son recibidas. Permite la manipulación de los datos, recibe datos y los inserta en las tablas siempre y cuando estos cumplan con las restricciones descritas en el apartado anterior, además retorna datos a las solicitudes ingresadas desde el backend.

**Backend:**

Se implementó como un medio de comunicación entre el frontend y la base de datos. Es la parte que se encarga de administrar las solicitudes enviadas desde el frontend, “traducir” esas operaciones para hacer la petición en el motor de base de datos. Las solicitudes que se encarga de manejar el backend son la lectura, inserción, actualización y eliminación de los datos contenidos en la base de datos según la necesidad y conveniencia del usuario que utilice el frontend.

**FrontEnd:**

Se desarrolló la aplicación utilizando la biblioteca de Google Angular, la cual permite el desarrollo de SPA’s. El desarrollo se basó en la creación de vistas según la especificación, donde las funcionalidades estaban basadas en las clases de usuarios Administrador y deportista. Para la aplicación, se realizaron CRUD’s en las diferentes vistas, cada uno trabajando sobre los datos solicitados y enviados. Se utilizaron forms de angular para obtener datos por parte del usuario, los cuales son procesados y enviados al BackEnd mediante los métodos POST, GET, PUT, DELETE , en typescript.

**FrontEnd AppMovilSatraviaTEC:**

Se desarrolló con la biblioteca de Android Studio y Google Angular. Su función es visualizar en modo individual la ruta óptima para el ciclista y poder cronometrar el tiempo de duración con sus respectivos kilómetros por dirección y la distancia total de la ruta. Su enfoque es totalmente para el uso del celular con la ubicación en tiempo real.

Descripción de roles:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| nombre | Apellido | Puesto: |  |  |
| Allan Josué | Calderón Quirós | Frontend Developer |  |  |
| Antony Fabián | Fallas Elizondo | Project Manager/Developer |  |  |
| Edgar Alberto | González Chacón | Android Developer |  |  |
| Ronny Josué | Santamaría Vargas | Backend Developer |  |  |
|  |  |  |  |  |

Minutas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instituto Tecnológico de Costa Rica | | | | |
|  |  |  |  |  |
| Proyecto | Stravia Tec | | | |
| Fecha | 23/10/2020 | | | |
| Medio | Discord | | | |
| Periodo | 17:00 - 19:00 | | | |
|  |  |  |  |  |
| Tema | Descripción | Comentarios | | |
| Indicaciones sobre avances de proyecto | Se realiza la lectura del documento del proyecto, para posteriormente realizar el plan de trabajo, donde se describen todos ellos pasos a realizar durante las siguientes semanas y de esta manera llevar un control | Las dudas que surgieron durante la lectura serán preguntadas al profesor durante la próxima lección | | |
| Se comentan errores en busca de solución | Luego de leer El plan se dividió el proyecto basado en las habilidades y destrezas de cada uno. |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instituto Tecnológico de Costa Rica | | | | |
|  |  |  |  |  |
| Proyecto | Stravia Tec | | | |
| Fecha | 06/13/2020 | | | |
| Lugar |  | | | |
| Periodo | 17:00 - 19:00 | | | |
|  |  |  |  |  |
| Tema | Descripción | Comentarios | | |
| Se busca solución en los problemas durante esta semana | Los integrantes comentan los errores que están teniendo |  | | |
| Se muestran los avances y las tareas que no se lograron hacer | Cada estudiante indica cuáles tareas aún no logra cumplir, además de indicar el porqué de esta situación |  | | |
| Avances y solución a problemas | Se buscan solución en conjunto para los problemas sin solución y las tareas con un porcentaje de tiempo sin terminar mayor |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instituto Tecnológico de Costa Rica | | | | |
|  |  |  |  |  |
| Proyecto | Stravia Tec | | | |
| Fecha | 13/10/2020 | | | |
| Medio | Discord | | | |
| Periodo | 17:30 - 19:00 | | | |
|  |  |  |  |  |
| Tema | Descripción | Comentarios | | |
| Organización de l  as uniones BackEnd-FrontEnd | Una vez que se logró conectar el BackEnd con el FrontEnd, se procedió a proponer el formato de envío, la implementación, los requerimientos y los datos que se debían obtener | Existió un atraso sustancial en lograr la comunicación, Existen funciones que son similares en varias partes de la especificación, por lo que implementarla en algunas de estas partes demuestra la capacidad de integrar la misma función en otros lugares | | |
| Avance en la aplicación móvil | Se discute la implementación de Angular en la aplicación y de los mapas en la página web |  | | |

Bitácora

**Ronny Santamaria Vargas**

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Tiempo** | **Actividad** |
| 23/01/20 | 1 h | División del proyecto, reconocimiento de las principales características del proyecto y distribución de roles. |
| 24/10/20 | 3 h | Creación del diagrama conceptual. Investigación acerca del principales características del funcionamiento de postgresql |
| 25/10/20 | 3.5 h | Implementación de las tablas en el motor de base de datos con el fin de empezar a realizar pruebas y las conexiones entre el rest service y postgresql. |
| 26/10/20 | 2 h | Creación de métodos para obtener los datos de la base de datos. |
| 03/11/2020 | 5 h | Reunión de recuento de lo avanzado durante la semana. Borrar datos de la base de datos. |
| 06/11/2020 | 4 h | Implementación del método que recibe las solicitudes get en el rest service. (Pruebas realizadas con postman) |
| 07/11/2020 | 1 h | Reunión para verificar el progreso del proyecto. |
| 08/11/2020 | 4 h | Implementación del método que recibe las solicitudes post. (Pruebas realizadas con postman). |
| 11/11/2020 | 2 h | Pruebas de comunicación del backend para ser mostrado en la red local. |
| 12/11/2020 | 7 h | Envío de datos desde el frontend hacia el backend. |
| 13/11/2020 | 4 h | Pruebas del envío de datos desde el frontend hacia el backend. |
| 14/11/2020 | 6 h | Creación de los métodos necesarios para procesar los datos provenientes del frontend hasta la base de datos. |
| 15/11/2020 | 4 h | Finalización de los métodos de comunicación de la base de datos y el backend. |
| 16/11/2020 | 3 h | Se habilitó el IIS para acceder a los servicios del api de forma remota. Pruebas con la serialización y el envío de datos del Frontend a el Backend. |
| 17/11/2020 | 8 h | Creación de consultas en las tablas para algunas de las vistas requeridas. |
| 18/11/2020 | 10 h | Muestra, actualización, borrado e inserción de datos de todas las tablas creadas en la base de datos. |
| 19/11/2020 | 13 h | Creación de consultas para algunas de las vistas. Pruebas de comunicación entre el frontend y el backend. |
| 20/11/2020 | 13 h | Subir imágenes al backend y actualizar la url de la foto donde se sube la imagen. Creación del método para la búsqueda de personas. Finalización de la documentación externa. |
| 21/11/2020 | 4 h | Pruebas del método de búsqueda del backend y el frontend. Verificación de la completitud de la documentación externa. |
| Total | 97.5 h |  |

Bitácora

Edgar Alberto González Chacón

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Tiempo** | **Actividad** |
| 23/01/20 | 1 h | División del proyecto, reconocimiento de las principales características del proyecto y distribución de roles. |
| 01/11/2020 | 5 h | Se logra instalar Angular, a pesar de tener errores con Node.JS, y se corrigen los errores. Y Se crea una aplicación de prueba para angular |
| 04/11/2020 | 4 h | Se crea el componente para solicitar usuario y contraseña para ingresar a la aplicación móvil. |
| 08/11/2020 | 6 h | Investigación para conectar angular con google maps, además, la instalación de las librerías correspondientes |
| 9/11/2020 | 5 h | Cargar el Api Key de google maps al proyecto, comprando la licencia sin inversión de $0 para acceder a los permisos de google maps y habilitar los diferentes scripts de la librería de google, además, de la instalación de la librerías específicas para poder visualizar una ruta. |
| 10/11/2020 | 7 h | Visualizar el mapa gps de Google Maps en una ubicación en específico |
| 11/11/2020 | 11.5 h | Mostrar la ruta |
| 11/11/2020 | 6 h | Mostrar indicaciones de la ruta seleccionada |
| 12/11/2020 | 15 h | Mostrar la ruta óptima y dibujarla. |
| 14/11/2020 | 7 h | Se indaga para crear un path entre los componentes y se crean las rutas para poder “viajar entre rutas”. Gracias al compañero Allan se facilitaron las investigaciones del archivo .routes |
| 16/11/2020 | 5 h | Mostrar el cronómetro del recorrido, colocando y separando los tiempos por tramo del ciclista. |
| 17/11/2020 | 10 h | Con la aplicación web, se crea una vista en android Studio para visualizar el localHost de la página en el teléfono android con el puerto localhost:4200 0.0.0.0 |
| 18/11/2020 | 7 h | Se re-acomoda la aplicación ya que en los dispositivos móviles no se logró ingresar la ubicación actual por permisos de google maps con estos dispositivos. |
| 19/11/2020 | 5 h | Mostrar el kilometraje por recorrer del ciclista, se crea un array el cual contiene todos los datos importantes de la ruta entre ellos el kilometraje total por recorrer del ciclista. |
| 20/11/2020 | 3 h | Se re-acomoda la vista de la aplicación .html de los componentes en angular para ver más atractivo el kilometraje y tiempo del recorrido del ciclista. |
| Total | 97.5 h |  |

**Bitacora**

**Antony Fabián Fllas Elizondo**

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Tiempo** | **Actividad** |
| 23/10/20 | 1h | División del proyecto, reconocimiento de las principales características del proyecto y distribución de roles. |
| 24/10/20 | 1.5h | Creación del diagrama conceptual |
| 01/11/20 | 4h | Búsqueda de información sobre SQLite y sobre su implementación |
| 3/11/20 | 4h | Se descargan las librerías necesarias para utilizar el SQLite en Visual Studio 2019 y se comienza a implementar |
| 4/11/20 | 3h | Implementación de la base de datos SQLite, creación y modificación |
| 6/11/20 | 5h | Corrección de errores con la creación de la base de datos, debido a que no se creaba la base, solo muestra un error que dice que no se logra generar pero no muestra |
| 9/11/20 | 6h | Implementación de Insertar, eliminar y cargar datos a la base de datos SQlite |
| 11/11/20 | 4h | Prueba de la base de datos y búsqueda de información para corregir los errores que se presentan |
| 12/11/20 | 3h | Se comienza a unir el SQlite con la aplicación . |
| 13/11/20 | 6h | Busqueda de informacion sobre formas u opciones de concertar SQlite con Angular para que logre funcionar,debido a que al intentar unirlas genera grandes errores |
| 14/11/20 | 6h | Pruebas para lograr unir SQLite y Angular, además de búsqueda de información |
| 16/11/20 | 9h | Se busca información detallada sobre los errores generados a la hora de intentar unirlos, además de información sobre opción alternativa para lograr crear la unión |
| 17/11/20 | 5h | Se descarga Angular, y sus librerías correspondientes para buscar crear la base de datos SQLite directamente en el Angular, demas de busqueda de informacion |
| 18/11/20 | 12h | Se crea la primera prueba de la base de datos SQLite en Angular,Con la función de generar la base y cargarla . |
| 19/11/20 | 15h | Se crean las funciones de insertar, eliminar y actualizar, pero todas las anteriores generan errores, por lo cual se empieza a buscar soluciones a dichos errores, con la desventaja de que un error trae consigo otro más grande. |
| 20/11/20 | 13h | Se busca información para lograr solucionar errores extraños que se generan debido al Angular, ya que al correr la base de datos en otro espacio que no sea Angular si funciona bien. |
| 21/11/20 | 10h | Se Buscan e implementan distintas maneras para crear la base de datos SQLite con distintas librerías, pero más que todo la complicación se encuentra en lograr unirle al Angular el esta base de datos en específico. |
|  |  |  |
| TOTAL: | 108.5h |  |

**Bitacora**

**Allan Josué Calderón Quirós**

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Tiempo** | **Actividad** |
| 23/01/20 | 1 h | División del proyecto, reconocimiento de las principales características del proyecto y distribución de roles. |
| 24/10/20 | 1.5 h | Creación del diagrama conceptual |
| 31/10/2020 | 4.5 h | Creación del proyecto y las dependencias de angular. Debug de problemas con Bootstrap. Creación de vistas |
| 01/11/2020 | 1.75 h | Debug de bootstrap, creación de vistas |
| 02/11/2020 | 0.5 h | Creación de ventanas para las diferentes funciones, proposición de unión con BackEnd |
| 03/11/2020 | 2 h | Creación de forms html |
| 04/11/2020 | 2 h | Instalación de herramientas del backend para iniciar debug, investigación sobre comunicación backend-frontend |
| 06/11/2020 | 2.5 h | Instalación de herramientas del backend, investigación sobre la utilización de las herramientas del backend y de programación en C# , investigación sobre comunicación backend-frontend |
| 07/11/2020 | 3.5 h | investigación sobre comunicación backend-frontend, reunión de avance del proyecto |
| 09/11/2020 | 5 h | investigación sobre comunicación backend-frontend, instalación de postman. Investigación de los métodos de HTTP |
| 10/11/2020 | 3.5 h | investigación sobre comunicación backend-frontend |
| 11/11/2020 | 10 h | investigación sobre comunicación backend-frontend, debug del backend, optimización de laptop debido al consumo de la herramienta Visual Studio Community 2019 |
| 12/11/2020 | 3 h | investigación sobre comunicación backend-frontend |
| 13/11/2020 | 3 h | investigación sobre comunicación backend-frontend, conclusión exitosa de una comunicación backend-frontend |
| 14/11/2020 | 2.5 h | Formato de comunicación entre backend y frontend |
| 15/11/2020 | 2 h | Comunicación con los miembros del equipo para conocer los requerimientos de formato de parte del backend. Empieza parseo de las comunicaciones entre el FrontEnd y el Backend. Documentación |
| 16/11/2020 | 5 h | Formato de comunicación backend-frontend. Creación de nuevos componentes con nuevos formatos siguiendo el mismo esquema, implementación de mapas en ventana “página de inicio” |
| 17/11/2020 | 15.75 h | implementación de mapas en ventana “página de inicio”, implementación de todas las entradas del frontend “visuales” de los requerimientos del proyecto,proyección de comunicación con el backend, documentación interna, creación de forms, inicia implementación final con backend. Intento de instalación de postgres (fallido). Al hacer un pull mediante la aplicación GitHub desktop en otra branch, en vez de hacer pull primero se sobreescribió el código, por lo que hubo que reinstalar el proyecto de angular con todas las dependencias. Hacer el login y redirigir a nueva página. Pasar datos a la nueva ventana. Generación de la página de inicio con los datos de usuario y mapas. Analizar la respuesta del server. Creación de ventanas para los requerimientos del administrador. Documentación interna |
| 18/11/2020 | 6 h | Conclusión de la ventana de Login. Envío de imágenes al server. Intento de solución de ejecución de Pgadmin. Scripts de población.Instalación de SQL server con la esperanza de que permitiría el funcionamiento de la aplicación pgAdmin, no tuvo éxito |
| 19/11/2020 | 12 h | Mostrar reportes desde la base de datos, agregar amigos, script de población.Algoritmo de Login, debug, documentación |
| 20/11/2020 | 11.5 h | Función de eliminar en tablas, mostrar gestión de retos, actualizar valores en las tablas (CRUD completo), algoritmo de búsqueda de amigos |
| 21/11/2020 | 7.5 h | Algoritmo de búsqueda de amigos, debug,documentación |
| Total | 105 |  |

**Problemas encontrados**

**Problemas de conexión entre BackEnd y FrontEnd:**

Se encontraron múltiples problemas para establecer comunicación entre el backend y el frontend. Para resolver estos problemas, se requirió casi una semana de dedicación exclusiva a la resolución del problema, lo que afecta importantemente el desempeño en los demás requerimientos del proyecto.

Entre los problemas que se tuvieron para conectar el FrontEnd con el BackEnd se encuentran :

-Bloqueo de comunicación entre servidores internos de la computadora debido al firewall: al intentar establecer una comunicación que únicamente involucra al BackEnd, utilizando para esto la herramienta Postman, se corroboró que no se podía utilizar ninguno de los métodos de HTTP. Para solucionar el problema, se sugirió por parte de un compañero intentar la comunicación luego de apagar el firewall de windows, lo cual pareció tener un impacto positivo en la resolución del problema. Cabe destacar que se intentaron varias soluciones al mismo tiempo, por lo que varias causas se resolvieron sin saber cuál era el problema que las causaba

-Errores en el tipo del mensaje enviado desde el cliente:cuando se intentaba enviar un mensaje de prueba al server, la ejecución se detiene en un breakpoint del algoritmo del server. Esto al parecer debido al formato con el que se estaba enviado el mensaje. Para solucionarlo, se procedió a enviar un archivo JSON estático con la misma estructura que la que esperaba el server. De esta manera,el server demostró estar recibiendo el mensaje, por lo que el problema era del lado del cliente. Se procedió a verificar si el problema era a la hora de obtener los datos desde el HTML o al darle formato en Typescript. Se encontró que angular utiliza métodos propios para la obtención de

información desde forms, por lo que no se pueden usar métodos de HTML “puro”.

-Errores en importación de bibliotecas de angular: al intentar exportar el método que se comunica con el cliente a un servicio, el proyecto no compilaba. Investigando sobre el error, se logró determinar que se debía importar la biblioteca Ng Module de angular para que la exportación funcionara correctamente y el código fuera reutilizable. Cabe destacar que el método utilizado para hacer post no funcionaba implementado directamente en un componente, el componente simplemente no cargaba, pero no se obtenía ningún error desde la terminal que corría el servidor de angular ni la consola del navegador. Al exportar el componente, el método funcionó correctamente sin alteraciones, por lo que se desconoce porque dicho método no funcionaba en un componente aislado

-Obtuvimos un error 400: Bad Request al intentar contactar al servidor por primera vez. Se soluciono luego de seguir <https://stackoverflow.com/questions/55723977/how-do-i-fix-a-400-bad-request-error-in-net-core-post-operation> y pasar el tipo de dato correcto

-Posteriormente, obtuvimos un error Http 415 Unsupported Media type, luego de empezar a pasar JSON. Luego de seguir varias sugerencias de <https://stackoverflow.com/questions/22566433/http-415-unsupported-media-type-error-with-json> , se procedio a darle exactamente el mismo formato al JSON que se recibe desde el cliente. Al parecer, se debe crear un modelo, como una clase. No se pueden pasar los datos del form al componente directamente y posteriormente enviarlos a través del método POST

**Problemas en el rendimiento de las herramientas utilizadas:**

-La realización de pruebas del proyecto en múltiples ocasiones requirió la utilización de diversos programas al mismo tiempo : Visual Studio Code para el FrontEnd, Visual Studio Community para el BackEnd, CMD de windows para mantener corriendo el server de Angular, un navegador para probar el programa desde una página, Postman para verificar la conexión y discord para comunicarse con los miembros del grupo, para obtener indicaciones o mostrar resultados. Estos programas ejecutados en conjunto, consumen muchos recursos de la PC, lo que hace que los diversos programas “crasheen” o se detengan. En particular, Visual Studio Community demostró ir lento incluso cuando no se está ejecutando ningún otro programa en conjunto. En algunas ocasiones, las PC llegaron incluso a requerir un reinicio debido a la imposibilidad de utilizarlas por la lentitud. Por lo tanto, se debió optimizar el uso de los recursos de la PC, cerrando procesos en segundo plano, definiendo las aplicaciones que se ejecutan al iniciar la PC, desactivando las animaciones de windows, si se requieren pequeñas modificaciones durante el tiempo de ejecución de pruebas se procede a utilizar el bloc de notas integrado de windows en vez de utilizar Visual Studio Code. Se tuvieron que realizar varios reinicios en varias ocasiones, lo que consume tiempo valioso para el desarrollo del proyecto, sin embargo, se hizo todo lo posible por mantener el uso de los recursos más óptimo posible para una depuración lo más fluida posible. También se propuso utilizar herramientas como hamachi para ejecutar el BackEnd y el FrontEnd en las computadoras de los distintos miembros en vez de depender de solo uno, sin embargo, existieron problemas a la hora de establecer la red local

**Problema con libreria Google Maps y la Api Key:**

-Un error común al realizar la investigación del gps es la visualización del mapa de google-maps en modo developer por lo que, anteriormente el uso completo de esta librería en la aplicación móvil no se pudo realizar, solamente algunas funciones como la de view-map. Pero para poder realizar esta aplicación móvil con los requerimientos estipulados, se necesita comprar la licencia de Api Key de Google Maps. Esta llave, desbloquea por completo las funciones de la librería, llegado a observar el mapa completo.

-Al tener la api key, se obtuvo un problema con el reconocimiento de la librería, se intentó exportar directamente la api en el componente GPS pero no se logró. Al final se tuvo que colocar el script de la clave en el archivo index.html de angular.

<!-- Maps-->

<script src="https://maps.googleapis.com/maps/api/js?key=XXX"></script>

<script src="https://maps.googleapis.com/maps/api/js?key=XXX&libraries=places&language=en"></script>

**Problema con la ejecución del Angular AppStraviaTEC**

-Al descargar el repositorio surge un error de no encontrar la librería, @angular-devkit, por lo que se corrige instalando en el cmd de windows el siguiente comando: npm install --save-dev @angular-devkit/build-angular

**Problemas conocidos**

**Instalación de Postgresql**

-Se intentó instalar la aplicación de Postgresql en la computadora del integrante Allan Calderón. Al iniciar la aplicación, no cargaba la página de inicio. Se intentó reiniciar la computadora, reinstalar la aplicación, cerrar y abrir el proceso, pausar el servidor SQL, sin embargo no hubo éxito. Esto ralentizó los procesos de prueba y debug ya que no se podía probar desde todas las computadoras y tenían que conectarse dos o más miembros a realizar pruebas. Se consultó : -<https://www.postgresql.org/message-id/B75CD08C73BD3543B97E4EF3964B7D7047AF4F86%40CITESMBX1.ad.uillinois.edu>

-<https://stackoverflow.com/questions/64888525/postgresql-pgadmin-keep-loading-forever>

-postgresql - pgAdmin 4 v3.0 and v3.1 doesn't load (crashes) on Windows 10 - Stack Overflow

-<https://www.postgresql.org/message-id/B75CD08C73BD3543B97E4EF3964B7D7047AF471F%40CITESMBX1.ad.uillinois.edu>

Es preocupante para este error que ni siquiera en la página oficial de postgresql lograran dar una respuesta al respecto. Parece que la aplicación tiene errores, en vez de ser un error de la pc donde se instala.

También se intentó instalar una versión antigua de pgAdmin, pero no se obtuvieron resultados.

Se instaló el Microsoft SQL server con la esperanza de que permitiría el funcionamiento de la aplicación, pero tampoco tuvo un resultado positivo.

Se lamenta el tiempo que se perdió tratando de solucionar este problema

**Problemas de conexión de internet:**

-A lo largo de todo el proyecto, diferentes miembros del equipo tuvieron problemas de conexión, lo que para ciertas partes del proyecto que obligatoriamente requieren colaboración entre dos o más miembros, implicó un retraso considerable. Además que sin internet, ciertas funciones , como las solución de problemas, documentación e implementación de funciones desconocidas no es posible.

**Problema entre la conexión del SQLite y la aplicación:**

-Se creó la base de datos SQLite en Visual Studio 2019, esta base se implementa con todas sus partes, y se hacen pruebas para comprobar la funcionalidad, lo cual se encuentra en perfecto estado, la aplicación se creó en Angular con todos sus componentes, A la hora de que se intenta unir comienza a generar errores extraños los cuales al buscarlos en internet no se encuentra registro de ellos, los problemas se generan a raíz de que el angular no está logrando entender la base de datos SQLite en específico, por lo que la conexión falla, al principio se cree que es por las librerías entonces el integrante Antony Fallas instala todas las librerias correspondientes del Angular en su computador para poder hacer pruebas, pero el error perdura, buscando información se encuentra una referencia que hace mención a que si deseas utilizar Angular preferiblemente no utilizar SQLite como base de datos, lo que deja un poco más claro el porqué de los errores generados, la solución que se busca es intentar generar directamente el SQLite en los componentes del Angular, pero se cuenta con muy pocos días para realizar esto, en el proceso van surgiendo errores extraños provenientes de la conexión debido a que ciertos comandos o indicaciones del SQLite el Angular nos los entiende y la sugerencia que brinca es eliminar esa línea de comando, además de que como se menciona anteriormente en internet el error no aparece entonces es muy difícil encontrarle solución a los problemas.

Conclusiones:

-Las tareas marcadas como finalizadas en la tabla donde se realizó la distribución de roles y se asignaron las tareas no se corresponde con el porcentaje de completitud del proyecto, muchas de las tareas que fueron establecidas con el fin de realizar una distribución de una forma equitativa para los miembros del grupo no fue correcta ya que se generalizó mucho en la división y la asignación de las mismas, debido a esto aunque se podía observar un avance “adecuado” durante el proyecto posteriormente se evidenció que no era el avance esperado debido a que muchas tareas estaban contenidas en otras.

-Para cada método del protocolo HTTP existen comandos diferentes dependiendo de la operación que se realice

-Por la naturaleza SPA de Angular, el mismo framework promueve la creación de un código estructurado, el cual evita las malas prácticas de programación dentro del contexto SPA tales como cargar diferentes documentos HTML (todo se debe cargar desde el index.html) , y también la creación de un código más limpio y reutilizable, por ejemplo, la utilización de servicios o la inclusión del código FrontEnd dentro de los archivos del componente generado por Angular (por ejemplo, no permite la inyección de Javascript directamente en el código HTML del documento).

**Recomendaciones**:

-Realizar una división del proyecto más detallada, que comprenda los elementos de una manera que dentro de una tarea no quede integrado de manera implícita otras tareas, por ejemplo: “Conexión del Backend con el frontend”, en lugar de proponer una tarea con esa estructura se puede plantear en varias tareas por ejemplo: “realizar post desde el frontend con el backend”, “realizar get desde el frontend con el backend”, “realizar put desde el frontend con el backend” y “realizar delete desde el frontend con el backend”, esta es una estructura más explícita y entendible que puede evitar confusiones durante el desarrollo del proyecto.

-Utilizar el método adecuado según la función CRUD que se esté realizando : POST para CREATE, GET para READ, PUT para UPDATE Y DELETE para DELETED

-Leer la documentación de Angular para conocer las prácticas de programación recomendadas, las funcionalidades que Angular Inhibe por su naturaleza SPA y la correcta estructura que debe poseer una aplicación FrontEnd generada con este framework.

Bibliografía:

<https://stackoverflow.com/questions/48747643/how-to-show-input-data-inside-modal-body>

<https://www.xspdf.com/resolution/52803304.html>

<https://www.w3schools.com/tags/att_input_type_hidden.asp>

<https://www.pluralsight.com/guides/posting-deleting-putting-data-angular>

<https://stackoverflow.com/questions/50943666/how-load-local-gpx-file-in-a-openlayer-map>

<https://www.w3schools.com/sql/trysqlserver.asp?filename=trysql_func_sqlserver_datediff>

<https://docs.microsoft.com/en-us/sql/database-engine/install-windows/install-sql-server-from-the-command-prompt?view=sql-server-ver15>

<https://www.enterprisedb.com/postgresql-tutorial-resources-training?cid=437>

<https://angular.io/guide/router#fetch-data-before-navigating>

<https://stackoverflow.com/questions/36835123/how-do-i-pass-data-to-angular-routed-components>

<https://angular.io/guide/http#adding-headers>

<https://medium.com/@yonem9/angular-qu%C3%A9-son-los-m%C3%B3dulos-y-c%C3%B3mo-se-refactoriza-una-aplicaci%C3%B3n-9457550e8e9>

<https://getbootstrap.com/docs/4.0/components/card/>

<https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Objetos_globales/JSON/stringify>

<https://material.angular.io/components/select/overview>

<https://ng-bootstrap.github.io/#/components/dropdown>

<https://www.positronx.io/angular-7-select-dropdown-examples-with-reactive-forms/>

<https://www.w3schools.com/html/html_form_elements.asp>

<https://stackoverflow.com/questions/62755093/angular-error-generic-type-modulewithproviderst-requires-1-type-arguments>

<https://stackoverflow.com/questions/51096567/generic-type-subjectt-requires-1-type-arguments-angular>

<https://stackoverflow.com/questions/45485073/typescript-date-type>

<https://angular.io/guide/user-input>

<https://www.w3schools.com/sql/sql_update.asp>

<https://www.acontracorrientech.com/#tve-jump-17125c774d3>

<https://www.acontracorrientech.com/routing-en-angular-guia-completa-parte-3/>

<https://www.programiz.com/csharp-programming/namespaces#:~:text=Namespaces%20are%20used>

<https://stackoverflow.com/questions/40978750/angular2-filesaver-js>

<https://en.it1352.com/article/53c6260591e24c929204acbccde5dbe2.html>

<https://stackoverflow.com/questions/39670360/angular2-filesaver-js-cannot-find-module-file-saver/41493844>

<https://github.com/DefinitelyTyped/DefinitelyTyped/issues/16485>

<https://stackoverflow.com/questions/57234220/module-has-no-exported-member-error-in-angular-module>

<https://stackoverflow.com/questions/50333003/could-not-find-module-angular-devkit-build-angular>

<https://github.community/t/problem-with-commiting-changes-in-desktop-github-exit-code-1-error/719/4>

<https://social.msdn.microsoft.com/Forums/es-ES/505d2414-ffd1-4185-8a62-2dbbd2cd1dd8/guardar-resultados-en-archivos-txt-en-javascript-implementar-funcion-en-este-codigo?forum=webdeves>

<https://es.stackoverflow.com/questions/139537/c%C3%B3mo-puedo-crear-un-fichero-txt-con-javascript-y-llenarlo-con-un-string>

<https://www.youtube.com/watch?v=k0-cP3nMiqw>

<https://www.google.com/search?biw=1366&bih=657&tbm=vid&sxsrf=ALeKk01--qQiQS4blLOgz3BrOrA9Xw362Q%3A1605764400604&ei=MAW2X-u5JKWB5wKKxoDQCQ&q=navigator.geolocation.watchposition+angular&oq=navigator.geolocation.watchPosition&gs_l=psy-ab.3.2.0l3j0i30k1l7.1321265.1321265.0.1325069.1.1.0.0.0.0.197.197.0j1.1.0....0...1.1.64.psy-ab..0.1.197....0.l7iO73aOYzc>

<https://progexpertos.com/q/119778/ngx-google-places-autocomplete-input-not-updating-selected-string>

<https://docs.microsoft.com/en-us/rest/api/maps/route/getroutedirections>

<https://docs.microsoft.com/en-us/rest/api/maps/route/postroutematrix>

<https://fmoralesdev.com/2020/01/03/serve-angular-app-over-https-using-angular-cli/>

<https://stackoverflow.com/questions/48196456/how-to-redirect-from-http-to-https-in-angular-5/48196942>

<https://stackoverflow.com/questions/57130901/getcurrentposition-in-js-does-not-work-on-ios>

<https://es.stackoverflow.com/questions/119444/conectarse-a-localhost-desde-un-dispositivo-que-se-encuentre-en-la-misma-red-uso>

<https://es.stackoverflow.com/questions/326656/c%C3%B3mo-puedo-configurar-localhost-para-mi-celular>

<https://getbootstrap.com/docs/4.3/components/navs/>

<https://getbootstrap.com/docs/4.3/components/jumbotron/>

<https://getbootstrap.com/docs/4.3/components/input-group/>

<https://getbootstrap.com/docs/4.3/components/buttons/>

<https://www.flipandroid.com/navigator-geolocation-getcurrentposition-no-funciona-en-android-google-chrome.html>

<https://stackoverflow.com/questions/49699067/property-has-no-initializer-and-is-not-definitely-assigned-in-the-construc>

<https://www.codingame.com/playgrounds/8012/angular-form-validation-example-tutorial>

<https://code.visualstudio.com/Docs/languages/typescript#_refactoring>

<https://www.freecodecamp.org/news/creating-an-electron-app-using-angular-and-sqlite3-24ca7d892810/>

<https://stackoverflow.com/questions/14947916/import-csv-to-sqlite>

<https://www.sqlitetutorial.net/sqlite-nodejs/>

<https://academiaandroid.com/sqlite-en-app-android-actualizar-eliminar-y-consultar-datos/>

<https://www.mclibre.org/consultar/amaya/css/css3-columnas.html>

<https://developer.mozilla.org/es/docs/Web/HTML/Elemento/input/Bot%C3%B3n>

<https://www.youtube.com/watch?v=orjkt0VHt1c>

<https://www.digitalocean.com/community/tutorials/js-geolocation-api>

<https://stackoverflow.com/questions/60502097/angular-geolocation>

<https://www.tutorialspoint.com/html5/geolocation_watchposition.htm>

<https://www.youtube.com/watch?v=fnC5lOaOc5I>

<https://www.youtube.com/watch?v=DQZTeZZXYBk&t=808s>

<https://www.youtube.com/watch?v=RSl9wYWQIcs>

<https://www.youtube.com/watch?v=lApggVS0icc&t=317s>

<https://www.youtube.com/watch?v=orjkt0VHt1c>

<https://www.youtube.com/watch?v=U8lgiEDmkv8&t=520s>

<https://www.youtube.com/watch?v=t-hZa1mPPN0&t=488s>

<http://zetcode.com/csharp/postgresql/>

<https://www.postgresqltutorial.com/>

<https://stackoverflow.com/questions/10320232/how-to-accept-a-file-post>

<https://docs.microsoft.com/en-us/aspnet/core/mvc/models/file-uploads?view=aspnetcore-5.0>

<https://www.c-sharpcorner.com/UploadFile/2b481f/uploading-a-file-in-Asp-Net-web-api/>

<https://code-maze.com/upload-files-dot-net-core-angular/>

<https://medium.com/nerd-for-tech/how-to-upload-files-in-net-core-web-api-and-react-36a8fbf5c9e8>

<https://docs.microsoft.com/en-us/aspnet/web-api/overview/security/enabling-cross-origin-requests-in-web-api>

<https://docs.microsoft.com/en-us/aspnet/core/security/cors?view=aspnetcore-5.0>

<https://developer.mozilla.org/es/docs/Web/HTTP/Access_control_CORS>

<https://www.infoworld.com/article/3173363/how-to-enable-cors-on-your-web-api.html>

<https://www.hexacta.com/how-to-enable-cors-on-your-web-api/>

<https://enable-cors.org/server_aspnet.html>

<https://stackoverflow.com/questions/18619656/enable-cors-in-web-api-2>

<https://dzone.com/articles/login-and-registration-aspnet-web-api-using-angula>

<https://fullstackmark.com/post/10/user-authentication-with-angular-and-asp-net-core>

<https://www.c-sharpcorner.com/article/login-and-crud-operations-in-asp-net-web-api-using-angular-9-web-application/>

<https://stackoverflow.com/questions/36102192/authentication-approach-to-be-use-in-asp-net-web-api-and-angular-js>

<https://stackoverflow.com/questions/27547192/how-to-use-direction-route-object-to-get-data>