Corporación Centroamericana de Servicios de Navegación Aérea

(COCESNA)

Informe de proyecto

Rutinas de mantenimiento

**Elaborado por**

Héctor Josué Ardón Morga

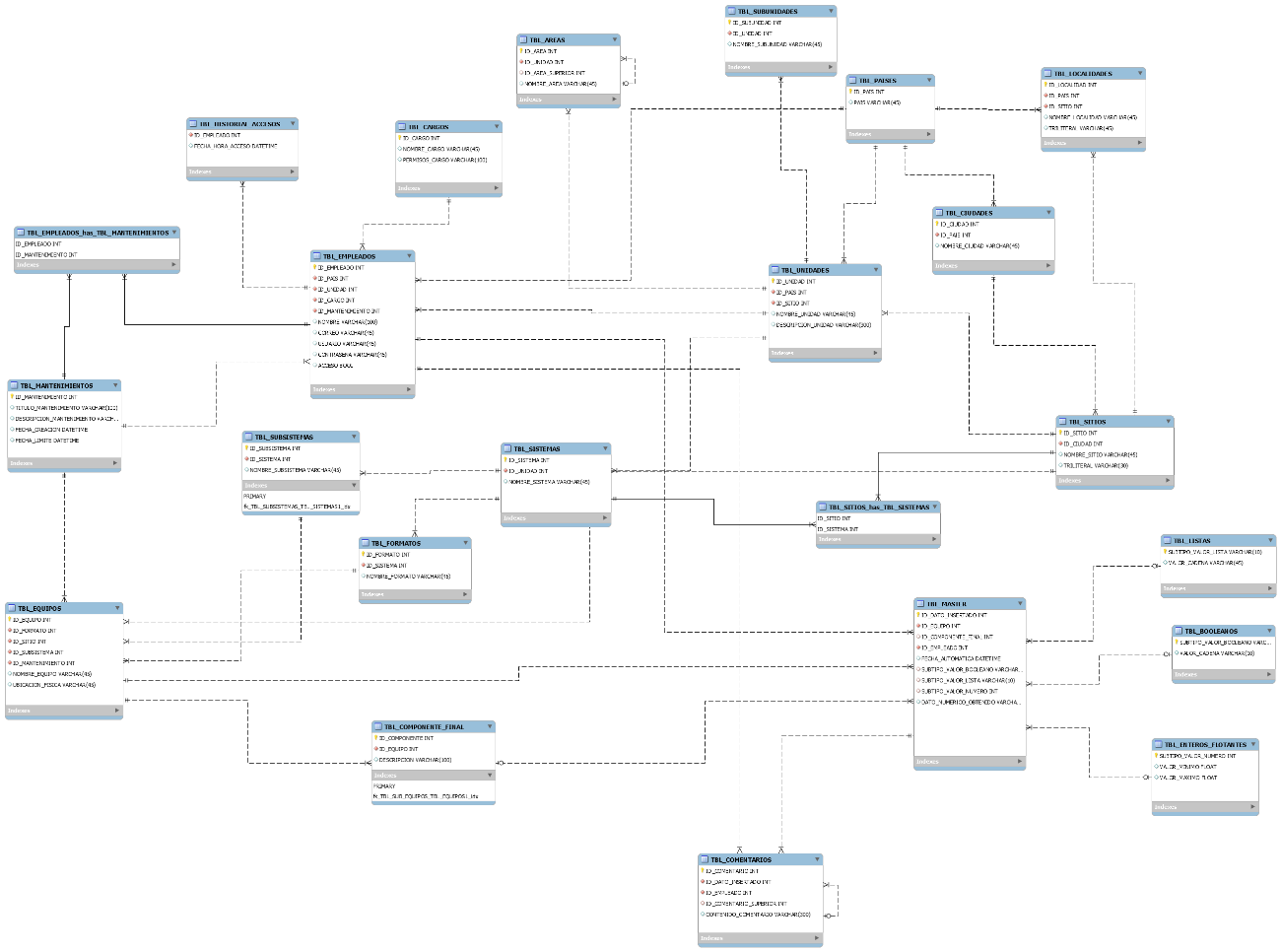
Introducción

El presente proyecto describe algunas características utilizadas para el desarrollo de este proyecto, el cual es para la empresa COCESNA. El desarrollo de este proyecto describe algunas cosas elaboradas hasta la fecha, el cual nos ha servido para comenzar a evolucionar aspectos que son importantes de implementar en la empresa, el hecho de comenzar a automatizar procesos y disponiendo de herramientas que contribuyen al desarrollo de la misma y del país.

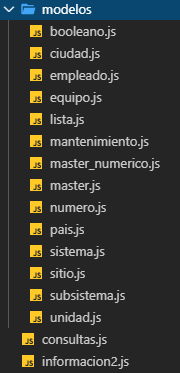
Se han utilizado herramientas de desarrollo web para implementar algunas necesidades en la empresa, cabe destacar que un proyecto de programación, no está 100% terminado, los proyectos deben ir escalando a medida que pasa el tiempo, pero dentro de lo respectivo se va a tratar de explicar aspectos importantes del mismo

**Identificación del problema**

Nos hemos visto en la necesidad de evolucionar la forma en la que se desarrollan los mantenimientos, elaborándolos en una plataforma digital, para esto hemos elaborado el siguiente modelo de base de datos My SQL.



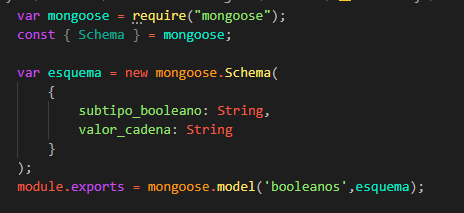
Tambien se elaboró parte de este modelo usando el gestor Mongo DB, el cual es un gestor mucho más moderno y actualmente se encuentrá agarrando más fuerza en cuanto a las bases de datos no relacionales.



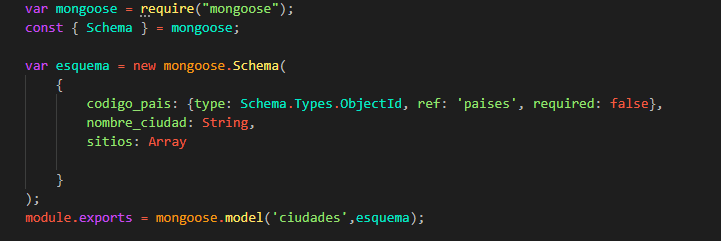
Los modelos se encuentran en archivos con extensión .js, en base a estos modelos es que se guía la aplicación, como si fuera el esquema que estamos acostumbrados a hacer en My SQL Work Bench.

A continuación, se exponen lo que son las colecciones de esta base de datos

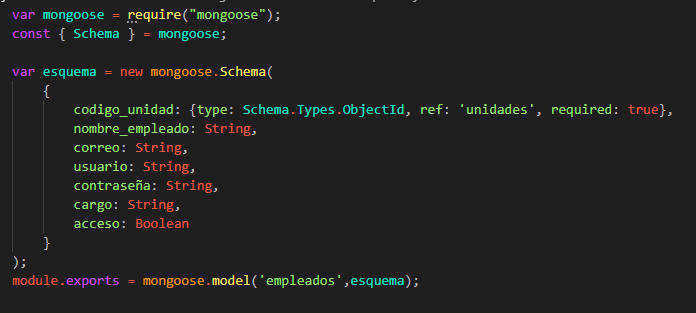
1. Colección de booleanos



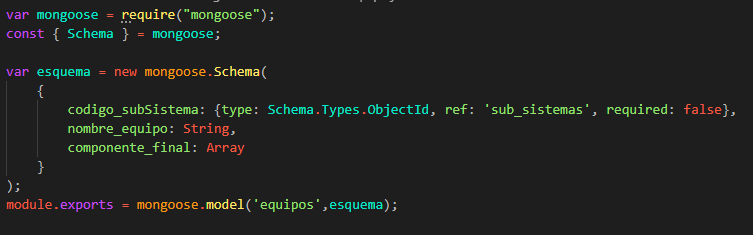
1. Colección de ciudades



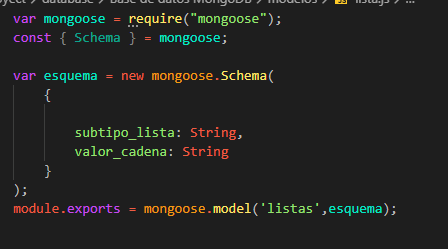
1. Colección de empleados



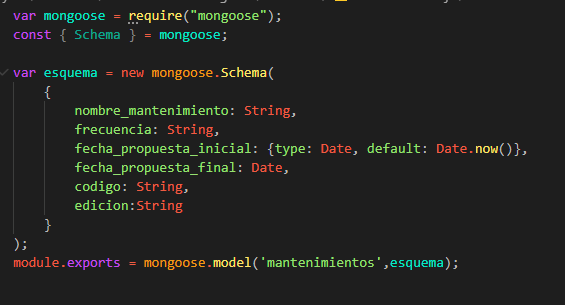
1. Colección de equipos



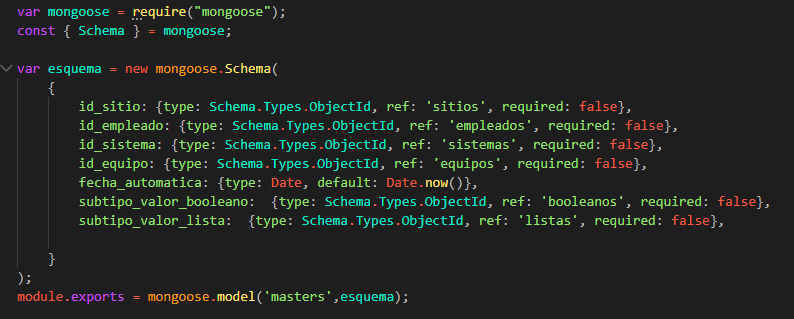
1. Colección de listas



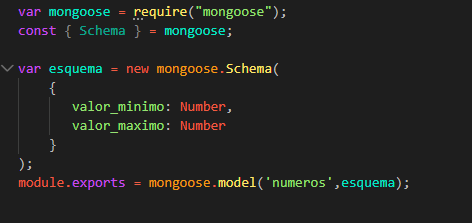
1. Colección de mantenimientos



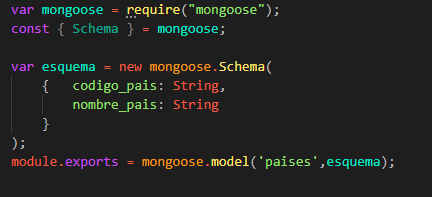
1. Colección master



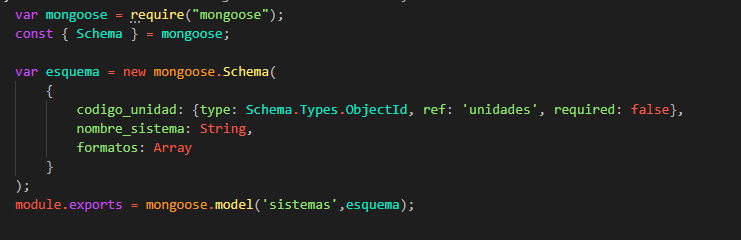
1. Colección de datos numéricos



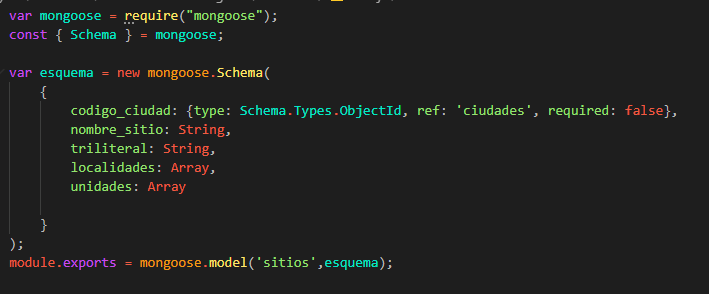
1. Colección de países



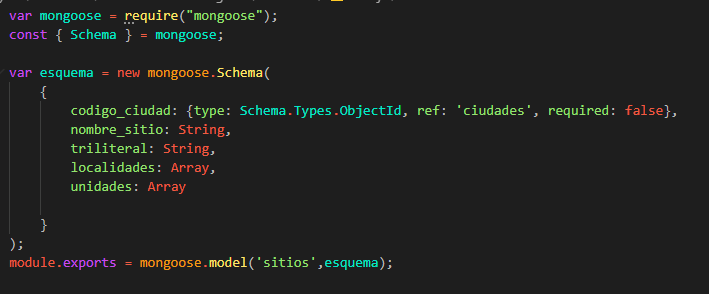
1. Colección de sistemas



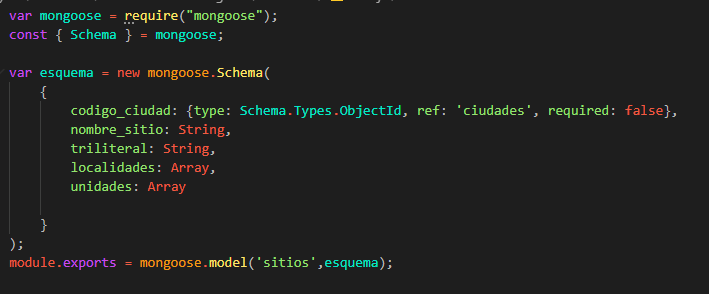
1. Colección de sitios



1. Colección de subsistemas



1. Colección de unidades

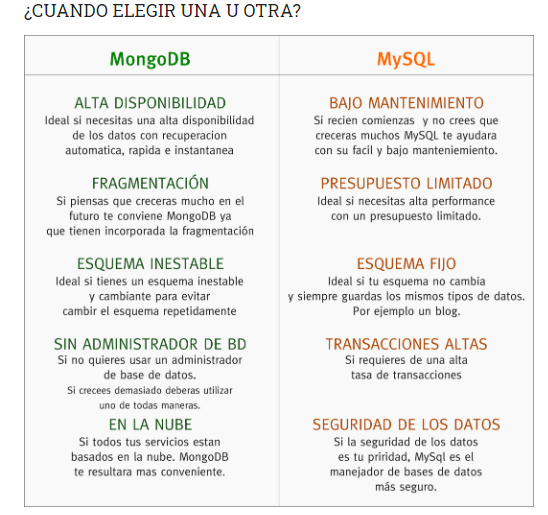


Cabe destacar que la forma que se hacen los cruces aquí, así como en My SQL usamos sentencias multitabla, ejemplos INNER JOIN, LEFT JOIN, RIGHT JOIN, etc,

Aquí los parámetros que tienen el tipo type: Schema.Types.ObjectId, indican claves foráneas, y para entender esto mejor, pongamos un ejemplo:

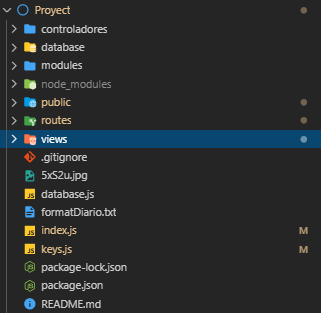
La colección o tabla paises es independiente, todas las colecciones independientes no van a tener el dato Schema.Types.ObjectId, pero las colecciones dependientes por ejemplo ciudades, las ciudades dependen de un país, por lo tanto la colección ciudades debe tener un dato con tipo Schema.Types.ObjectId, el cual almacene el \_id del país al que pertenece, \_id es el equivalente a una clave primaria de las que se usan en SQL.

Me tomo un poco más el tiempo de explicar sobre este gestor, porque SQL es bastante popular, ya muchas consultas y sentencias se conocen desde hace muchos años, y la documentación es incluso mucho más extensa.



Estructura del proyecto

El proyecto se encuentra estructurado de esta forma



El proyecto de forma original se conecta a la base de datos My SQL, pues la idea para hacerlo era usando este gestor, el lenguaje de programación utilizado es JavaScript, se usa el entorno su entorno Node JS del lado del servidor, al decir este termino del lado del servidor, es del lado de la base de datos o sea lo que se le llama el back end, que es lo que le hace que cada componente del Front End, (HTML, CSS, Java Script), pueda hacer la función para la cual haya sido establecido

**Carpeta controladores**

Esa carpeta contiene archivos que contienen funciones, esas funciones se encargan de hacer las tareas que tengan que hacer los routers, routers son las rutas que el servidor responde cuando el cliente hace una petición.

Una aplicación CRUD, es una aplicación básica de un sistema de información, CRUD significa *CREATE READ UPDATE DELETE,* Create se refiere a la operación de insertar en la base de datos, lo cual se usa la sentencia INSERT INTO.

Read es lo que se le llama leer,