INSTITUTO TECNOLÓGICO DE COSTA RICA

Campus Tecnológico Central Cartago

A picture containing font, graphics, screenshot, typography

Description automatically generated

Análisis de Resultados

Ingeniería en Computación IC-2101

Bases de Datos I

Realizado por: Ángel Zeledón, Pablo Ruiz

Profesor: Franco Quirós

Semestre II 2023

# Índice

* [Introducción al Proyecto](#_Introducción_al_Proyecto)
* [Ambiente](#_Ambiente)
* [Estados](#_Estados)
* [Métricas del Proyecto](#_Métricas_del_Proyecto)

# Introducción al Proyecto

Uno de los grandes retos de muchas pequeñas empresas hoy en día, es que no se digitalizan, se quedan cómodamente en el lugar que están, talvez con una que otra modernización en sus sistemas, pero esto por necesidad.

Pero muchas de estas pequeñas empresas descuidan la facilidad de acceso a si mismas. Un cliente busca la comodidad, por eso las empresas que facilitan esto tiene un boom, pongamos por ejemplo Uber, el concepto no es nuevo, la entrega a domicilio ha existido por años, pero nuca había sido centralizada ni digitalizada, antes dde el boom de Uber y otras empresas similares, uno o se tenia que saber el teléfono de el restaurante, o esperar que estuviera en Google. Uber quito esta necesidad por completo, solo era necesario tener la app y que ellos tuvieran un convenio con el lugar, se removió el factor humano a la hora de hacer el pedido. Y le facilito a el empresario la entrega, porque ya no tiene que pedir direcciones de manera hablada, la app provee una dirección en un mapa.

Nuestro proyecto no se acerca a la capacidad de estas aplicaciones, es un simple sistema de inventario, ‘barebones’ se podría decir.

Como Usarlo

Al iniciar nuestra aplicación se va a presentar una ventana de inicio de sesión/registro.

A screenshot of a computer screen

Description automatically generated

El usuario deberá introducir sus credenciales o crearlas.

Una vez cumplido este paso se mostrará la siguiente ventana:

A screenshot of a computer

Description automatically generated

El usuario puede hacer una serie de operaciones, entre ellas:

* Insertar Artículos
* Actualizar artículos
* Borrarlos
* Encontrar por nombre
* Mostrar los primeros n artículos
* Mostrar por clase.

Estas ultimas 3 se pueden encontrar en la parte de arriba:

A white background with black squares

Description automatically generated

Debajo de esto se verán los artículos filtrados por lo que el usuario haya solicitado.

A screenshot of a computer

Description automatically generatedPara insertar artículos aparecerá el pop siguiente

Aquí se deberá llenar todos los campos para

# Ambiente

A screenshot of a computer

Description automatically generated El ambiente de desarrollo es una mezcla de MSSQL server para SP y servidor SQL, una API hecha con las librerías “Morgan”,” Sequelize”, “Express”, … en JavaScript, y un Frontend usando el framework “React” también con JavaScript.

Es el mejor ambiente de desarrollo, probablemente no, no es el más seguro, no es el mas robusto, Ni tampoco creamos el Frontend mas bonito que existe, y mientras que JavaScript tiene sus peculiaridades, que nos ha causado horas de mirar código haciendo cambios pequeños con esperanza de arreglar un error, sirve.

A computer screen shot of a black screen

Description automatically generated

Como IDE para API y Frontend usamos VSCode, la versión ‘universal’ de Visual Studio. Elegimos este IDE porque ya ambos(Pablo y Angel) estamos familiarizados con él, y lo tenemos personalizado a nuestro gusto y necesidades. Puede que sea un IDE pesado y hambriento por RAM pero las funcionalidades, Herramientas y personalización que ofrece, es superior a muchos IDE’s.

Problemas

Entre los muchos problemas que tuvimos están incluidos:

* Carácter ‘N’ agregado por JavaScript en ciertas ejecuciones de SP, la solución fue separar los String de llamado de SP, Razón de problema: ni idea
* Problemas con sintaxis de envió de datos entre Frontend y API, Problemas: resulta que Angel y yo teníamos ideas diferentes de como enviar los datos, y creemos que el otro si había entendido, por lo que pasamos un buen rato dibujando hasta darnos cuenta de que no nos habíamos entendido.
* Frontend no enviaba info bien, Problema: envío de información mal, Solución: cambio de unos envíos de info que estaban alambrados y no nos acordábamos.
* Correcciones menores de algunos errores de sintaxis que tuvimos al duplicar ciertas funciones para agregar funcionalidad.
* React no da un error en especifica, indica en sus librerías que se cayó, no en que parte del código del usuario se cayó, Solución: debugeo ☹

# Estados

|  |  |  |
| --- | --- | --- |
| Item | Resultado | Cometarios |
| 1) Documentación | 100 |  |
| 2) Creación de Base de datos. | 100 | Todas las tablas creadas y con sus restricciones respectivas |
| 3) Llenado de Base de datos por XML | 0 |  |
| 4) SP creados. | 100 | Todos creados, con errores en EventLog |
| 5) Conexión a la Base de datos, del código en capa lógica. | 60 | Para la fecha de entrega (25) nos apareció un error en la parte de insertar artículo, básicamente JavaScript decidió agregarle a la llamada de un SP un N que hacia que fallara el llamado. Seguimos sin saber porque.  Los artículos se muestran por código, no nos dio tiempo de arreglarlo  Java script no esta dando problemas con los strings. Algunas veces Ejecuta los SP bien, otras veces hace cambios raros y jode todo. |
| 6) Corrección del funcionamiento de los requerimientos programa | 90 | No logeamos Logout de Usuario |

# Métricas del Proyecto

|  |  |  |
| --- | --- | --- |
| Nombre | Valor | Comentario |
| Entradas git | 25 | En conjunto y branches |
| Entradas Bitacora | 8 |  |
| Primera entrada de Bitacora | 5/9/2023 |  |
| Primera entrada Git | 4/9/2023 |  |
| Ttl mins | 1195 | En conjunto |
| Horas pruebas | 3 hrs |  |
| Logeos de eventos en BD antes de entrega | 1000+ |  |