INDICE

[Información General. 3](#_Toc10889385)

[Propósito. 3](#_Toc10889386)

[Alcance. 3](#_Toc10889387)

[Personal involucrado. 3](#_Toc10889388)

[Básico. 3](#_Toc10889389)

[Convenciones, definiciones, acrónimos 3](#_Toc10889390)

[Referencias, 3](#_Toc10889391)

[Resumen 3](#_Toc10889392)

[Perspectiva del producto 3](#_Toc10889393)

[Funcionalidad del producto 3](#_Toc10889394)

[Características del producto 3](#_Toc10889395)

[Restricciones 3](#_Toc10889396)

[Restricciones 3](#_Toc10889397)

[Suposiciones 3](#_Toc10889398)

[Información funcional y no funcional. 3](#_Toc10889399)

[Proceso Funcional. 3](#_Toc10889400)

[Requerimientos Funcionales generales 3](#_Toc10889401)

[Requisitos comunes de las interfaces 4](#_Toc10889402)

[Interfaces de usuario 4](#_Toc10889403)

[Interfaces de hardware 4](#_Toc10889404)

[Interfaces de software 4](#_Toc10889405)

[Interfaces de comunicación 4](#_Toc10889406)

[Requerimientos Funcionales específicos 4](#_Toc10889407)

[Requisito funcional 1 4](#_Toc10889408)

[Requisito funcional 2 4](#_Toc10889409)

[Requerimientos No Funcionales 5](#_Toc10889410)

[Requisitos de rendimiento 5](#_Toc10889411)

[Seguridad 5](#_Toc10889412)

[Fiabilidad 5](#_Toc10889413)

[Disponibilidad 5](#_Toc10889414)

[Mantenibilidad 5](#_Toc10889415)

[Portabilidad 5](#_Toc10889416)

[Información técnica. 6](#_Toc10889417)

[Básico. 6](#_Toc10889418)

[Base Datos. 6](#_Toc10889419)

[Front End 6](#_Toc10889420)

[Vista “Login” 6](#_Toc10889421)

[Vista “registro”. 7](#_Toc10889422)

[Backend. 7](#_Toc10889423)

[Capa de la vista 7](#_Toc10889424)

[Capa de negocio 7](#_Toc10889425)

[Capa de datos 7](#_Toc10889426)

[Capa de servicio 7](#_Toc10889427)

[Seguridad. 8](#_Toc10889428)

[Complementos. 8](#_Toc10889429)

[Pendientes de definición. 8](#_Toc10889430)

[Pendientes Técnicos. 8](#_Toc10889431)

# Información General.

## 

## Propósito.

realizar una aplicación móvil capaz de conectar al conductor habitual de una camioneta Golden Dragon con una central de asesores que le permitan obtener repuestos mecánicos y mantenimiento técnico automotriz a su vehículo, así como la posibilidad de referenciar amigos (nuevos clientes) que permitirán al usuario de la aplicación obtener puntos para futuros descuentos en productos o servicios.

El objetivo principal de la aplicación es resolver la necesidad de los conductores de camionetas Golden Dragon en el momento de sufrir una falla mecánica o necesitar asistencia automotriz.

## Alcance.

Desarrollar una Aplicación móvil que permita al usuario solicitar repuestos o piezas automotrices, solicitar el servicio de grúa o taller de reparación; y permitir referir a posibles nuevos clientes.

## Personal involucrado.

## Básico.

## Convenciones, definiciones, acrónimos

## Referencias,

## 

## Resumen

## Perspectiva del producto

Este ítem se basa en el árbol de problemas que realizaron con anticipación y de donde salió esa necesidad y que se piensa solucionar.

## Funcionalidad del producto

## Características del producto

## Restricciones

## Restricciones

## Suposiciones

# Información funcional y no funcional.

## Proceso Funcional.

|  |
| --- |
|  |

## Requerimientos Funcionales generales

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF01 |
| **Nombre del Requerimiento:** | Autentificación de Usuario. |
| **Características:** | Los usuarios deberán identificarse para acceder a cualquier parte del sistema. |
| **Descripción del requerimiento:** | El sistema podrá ser consultado por cualquier usuario dependiendo del módulo en el cual se encuentre y su nivel de accesibilidad. |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 * RNF05 * RNF08 |
| **Prioridad del requerimiento:**  Alta | |

## Requisitos comunes de las interfaces

### Interfaces de usuario

### Interfaces de hardware

### Interfaces de software

### Interfaces de comunicación

## Requerimientos Funcionales específicos

### Requisito funcional 1

• Autentificación de Usuarios: los usuarios deberán identificarse para acceder a cualquier parte del sistema.

 El sistema podrá ser consultado por cualquier usuario dependiendo del módulo en el cual se encuentre y su nivel de accesibilidad

### Requisito funcional 2

* **Consultar Información:** El sistema ofrecerá al usuario información general acerca de la Instrucción Académica, materias, Calendario de eventos.
* **Consultar Instrucción Académica:** Muestra información general sobre los cursos que la institución ofrece, en qué consiste, duración por período académico.
* **Consultar Materias:** Permite a los estudiantes ver las asignaciones que los docentes han publicado, descargar guía e información relevante a la materia y ver notas obtenidas en evaluaciones.
* **Consultar Calendario de Eventos:** Muestra a los usuarios información relevante a noticias u otros eventos planificados.

## Requerimientos No Funcionales

### Requisitos de rendimiento

* Garantizar que el diseño de las consultas u otro proceso no afecte el desempeño de la base de datos, ni considerablemente el tráfico de la red.

### Seguridad

* Garantizar la confiabilidad, la seguridad y el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta.
* Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean documentos, archivos y contraseñas.
* Facilidades y controles para permitir el acceso a la información al personal autorizado a través de Internet, con la intención de consultar y subir información pertinente para cada una de ellas.

### Fiabilidad

* El sistema debe tener una interfaz de uso intuitiva y sencilla
* La interfaz de usuario debe ajustarse a las características de la web de la institución, dentro de la cual estará incorporado el sistema de gestión de procesos y el inventario

### Disponibilidad

* La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 7 días por 24 horas, garantizando un esquema adecuado que permita la posible falla en cualquiera de sus componentes, contar con una contingencia, generación de alarmas.

### Mantenibilidad

* El sistema debe disponer de una documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible
* La interfaz debe estar complementada con un buen sistema de ayuda (la administración puede recaer en personal con poca experiencia en el uso de aplicaciones informáticas).

### Portabilidad

# Información técnica.

## Básico.

1. Las hojas tienen columnas escondidas. Las hojas con columnas escondidas tienen un objetivo de proceso y de formulación que ayuda al control del modelo. Es importante que cuando se requiera adicionar líneas o columnas se haga un análisis de impacto sobre el modelo y sobre las rutinas o macros que utilizan información o referencias de celdas y rangos fijos.

## Base Datos.

1. Esta base de datos es el repositorio donde se almacenan los valores comunes a los usuarios del Libro Business Case.
2. El concepto general de tener una base de datos con la información centralizada es tener el control de adición y modificación de datos utilizados en el cálculo del modelo de Business Case.
3. Modelo MER:

|  |
| --- |
|  |

1. Path: [**\\10.500.500.204\**](file:///\\10.500.500.204\AxedeBC\AxeBC.accdb)
2. Los perfiles administradores pueden modificar cualquier campo o tabla que reside en esta base datos.

Los perfiles usuarios pueden leer cualquier campo o tabla que reside en esta base datos.

1. Script

## Front End

El fornt end se realizo en asp .net core versión 2.0

### Vista “Login”

Esta vista permite el ingreso al sistema de usuarios q están registrados es la base de datos, sino están registrados, no pueden acceder alsistema. El registro se hace en la empresa cliente

Se definen los valores que impactan la funcionalidad de la vista.

1. Se definen las celdas que deben ser obligatorias al diligenciar un business case.
2. Se definen los centros de costo.
3. Se define que centros de costo componen una ayuda para celdas de selección.

### Vista “registro”.

Cada una de las tablas diseñadas en esta hoja está separada por una columna en BLANCO, esta columna no debe ser borrada, modificada.

Se encuentra toda información base de operación del libro. El objetivo de esta hoja es encontrar centralizada la adición, modificación de valores utilizados en el proceso de fabricación de un business case.

1. Se encuentra las descripciones que son utilizadas para asignar un grupo de ayuda a una celda definida por parámetros.
2. Se encuentran valores asociados a la descripción de ayuda. Este valor es utilizado para afectar una celda y a su vez comprometida en los cálculos o formulación del Business Case.

## Backend.

El backend se realizo en asp .net core 3.0 con una arquitectura de 4 capas

### Capa de la vista

En la capa de la vista se codifico la vista user para acceder a registrar los usuarios por medio de usuarios administradores y se encuentra en la ruta 10.100.100.200/user

En la capa de la vista se codifico la vista login para acceder a registrar los usuarios en la vista user por medio de usuarios administradores y se encuentra en la ruta 10.100.100.200/login

En la capa de la vista se codifico la vista user para acceder a registrar los usuarios por medio de usuarios administradores y se encuentra en la ruta 10.100.100.200/user

### Capa de negocio

Se manejó la lógica de negocio de la app por medio de las clases A, B, y la clase C solo funciona como modelo de nuestro proyecto

### Capa de datos

Se creó un modelo que se conecta a nuestra base de datos que esta en el servidor 10.100.100.255:3306 por medio del usuario aaa y el password bbb.

### Capa de servicio

Se crearon las clases A, b, c, d y e que permite el uso de la api de nuestro servidor 10.100.100.255:3306 por medio del usuario aaa y el password bbb.

La clase A nos realiza los métodos get, post, put y delete de la tabla mama

La clase B nos realiza los métodos get, post y delete de la tabla papa

La clase B nos realiza los métodos get, de la tabla tio

## Seguridad.

1. El código tiene una clase que se encarga de encriptar las contraseñas del usuario por medio del siguiente método de la clase encrip:
2. La conexión con las api maneja un nivel de autenticación basado en auth 2.0 .

## Complementos.

Para compilar el programa se debe instalar el framework Tequendama en el servidor XXX.xxx.YYY.yyy en modo administrador.

El servidor de base de datos debe tener 1 Tb de RAM para realizar el procesamiento.

# Pendientes de definición.

* Participación perfiles Comercial
* Participación perfiles Cuenta
* Participación perfiles Diferenciación
* Valor flujo de caja
* Resumen para BC Servicio
* Resumen para BC Producto
* Escenarios
* Plantilla
* Informes de análisis

# Pendientes Técnicos.

* Ocultar hojas
* Documentar Solución
* Actualizar base de datos ABC.bak