

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Manejo e Implementación de Archivo
Ing. Oscar Paz
Aux. Carlos Eduardo Hernandez Molina



Proyecto 2

Objetivos

- **General**
 - Implementar un modelo de base de datos eficaz y escalable
- **Específicos**
 - Crear una plataforma web con una conexión a una base de datos relacional
 - Implementar una arquitectura personalizada
 - Construir una base de datos a través del uso de archivos de scripts SQL

Descripción.

Se desarrollará una aplicación web similar a Facebook o Twitter, esta permitirá compartir publicaciones a los usuarios registrados. Tendrá las funcionalidades de login, registro de usuarios, ver publicaciones, crear publicaciones, chatear con amigos y bots para obtener información y gráficas sobre los datos del COVID-19 proporcionados.

Los datos que utiliza el Bot para mostrar la información al usuario se obtendrán siempre de la siguiente fuente: https://narek-t.github.io/covid19/timeseries_global.json

La fuente muestra los datos con una estructura en formato JSON que facilita la obtención de datos, descrita a continuación:

```

{
  "pais": {
    "lat": "33.93911",
    "lng": "67.709953",
    "code": "AF",
    "flag": "AF",
    "timeSeries": [
      {
        "date": "1/22/20",
        "confirmed": 0,
        "deaths": 0,
        "recovered": 0
      },
      {
        "date": "1/23/20",
        "confirmed": 0,
        "deaths": 0,
        "recovered": 0
      },
      ...
    ]
  },
  ...
}

```

Las cantidades de casos confirmados, recuperados y muertos, son acumulativos, es decir que en determinada fecha se tiene el valor del total de casos (ya sea confirmados, recuperados o muertos) hasta esa fecha.

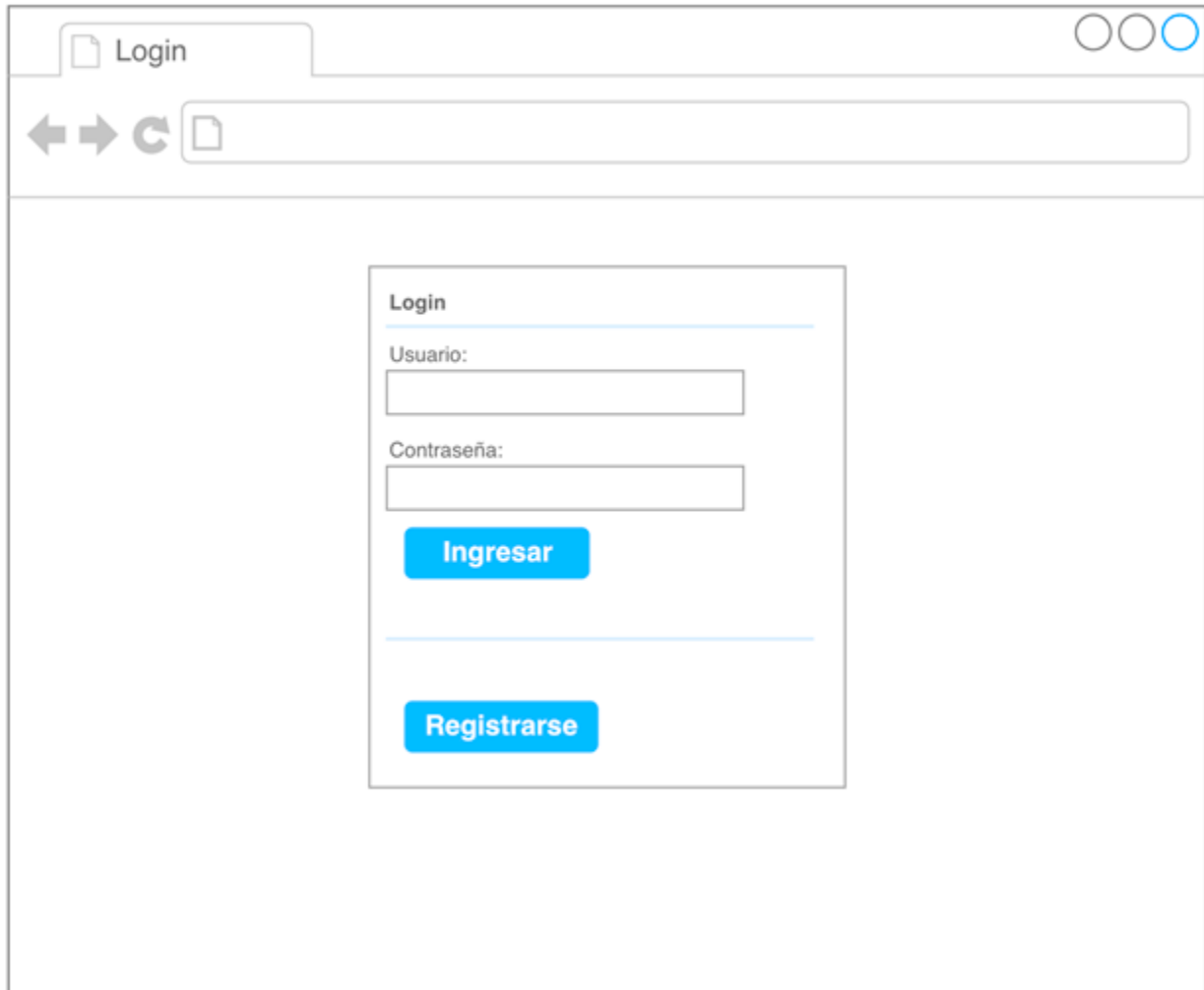
Los elementos que posee el arreglo con clave "<País>", están ordenados según el atributo "<date>", es decir, el último elemento de este arreglo contiene la información más reciente del total de casos de ese "<País>".

Sitio web

A continuación se describen las secciones que contendrá el sitio web:

Login

Para la autenticación se realizará por medio de las credenciales: nombre de usuario y contraseña.



The image shows a mockup of a web browser window. The browser has a single tab titled "Login". The address bar contains navigation icons (back, forward, refresh) and a document icon. The main content area displays a login form centered on the page. The form is titled "Login" and includes two input fields: "Usuario:" and "Contraseña:". Below the "Contraseña:" field is a blue button labeled "Ingresar". A horizontal line separates the login section from the registration section, which features a blue button labeled "Registrarse".

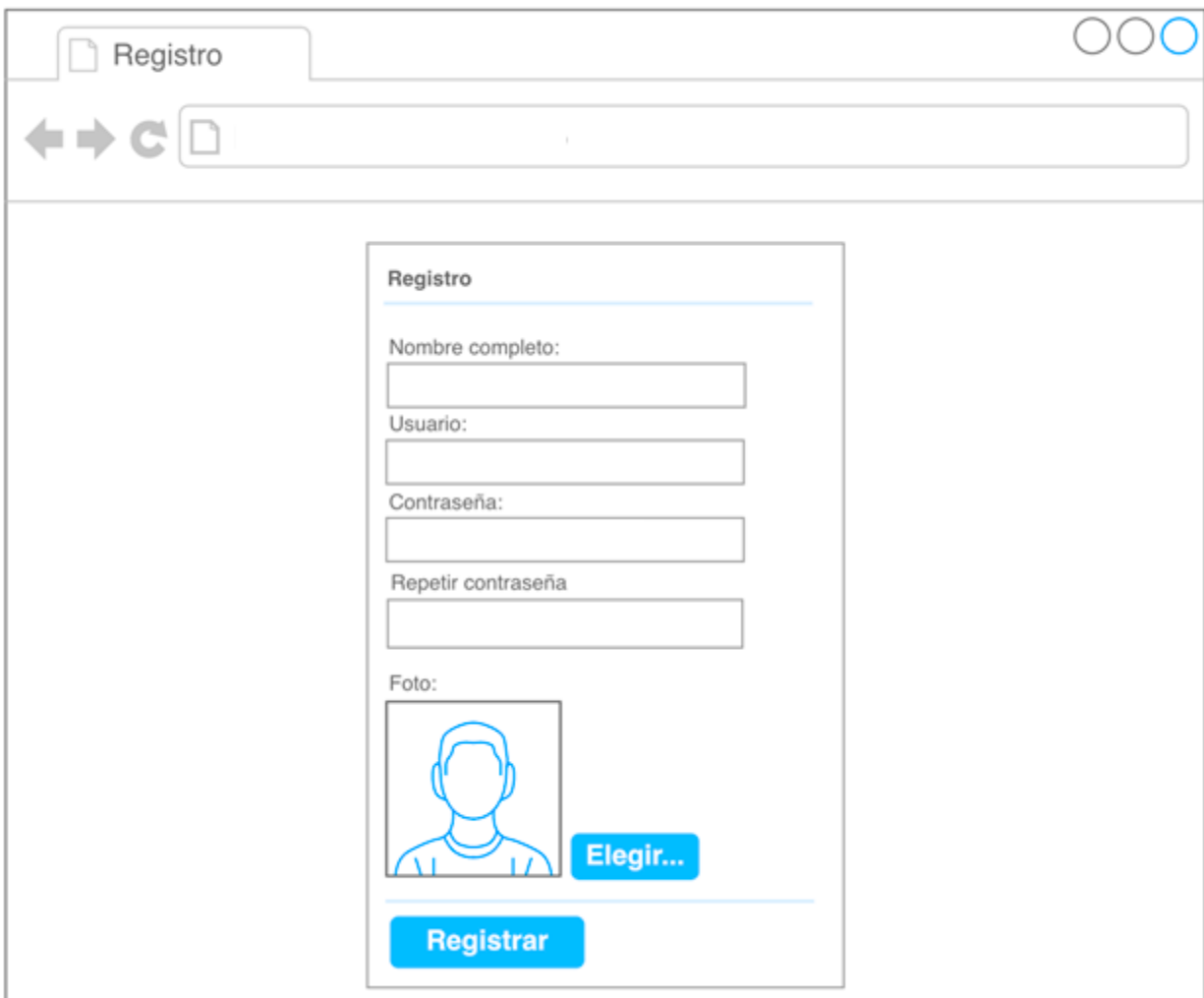
Registro

Para registrar un nuevo usuario obligatoriamente se pedirán los datos:

- Nombre completo
- Usuario
- Contraseña
- Repetir contraseña

La foto del usuario será opcional, esta se realizará eligiendo una imagen de la computadora.

Para el caso de la contraseña, se debe de utilizar un algoritmo de encriptación al guardarse en la base de datos, para que no pueda ser legible por ninguna persona, se puede utilizar MD5, SHA1 o algún otro.



The image shows a web browser window with a single tab titled "Registro". The browser's address bar is empty. The main content area displays a registration form with the following elements:

- Registro**: A section header with a horizontal line below it.
- Nombre completo:**: A text input field.
- Usuario:**: A text input field.
- Contraseña:**: A text input field.
- Repetir contraseña**: A text input field.
- Foto:**: A label above a square image placeholder showing a person's outline.
- Elegir...**: A blue button next to the photo placeholder.
- Registrar**: A large blue button at the bottom of the form.

Página principal

Información del usuario

En esta sección, el usuario puede modificar sus datos, tales como:

- Nombre completo
- Usuario
- Foto de perfil
- Modo Bot (desactivada por defecto, esta funcionalidad se explica en la sección de Chat)

Se debe ingresar la contraseña correcta para que se guarden los datos modificados.

The image shows a web browser window with a tab labeled 'Info'. The browser's address bar contains navigation icons (back, forward, refresh) and a document icon. In the top right corner of the browser window, there are three window control buttons (minimize, maximize, close). The main content area of the browser displays a user profile form titled 'Mi Información'. The form includes the following elements: a 'Nombre completo:' label followed by a text input field; a 'Usuario:' label followed by a text input field; a 'Foto:' label followed by a square placeholder for a profile picture and a blue 'Elegir...' button; a 'Modo Bot:' label followed by a blue toggle switch currently set to 'ON'; and a 'Confirmar contraseña' label followed by a password input field with masked characters. A large blue 'Guardar' button is positioned at the bottom of the form. In the top right corner of the page content, there is a blue button labeled 'Inicio'.

Inicio



Usuarios y amigos

Para visualizar a los otros usuarios registrados, se debe de mostrar un apartado donde se liste a los usuarios que no son amigos. Los usuarios que son amigos solo se mostrarán en el Chat, que se describe más adelante.

La forma para agregar amigos queda a discreción del estudiante. Se debe implementar un sistema de solicitudes de amistad

Publicaciones

En este apartado el usuario puede ver y crear publicaciones.

Ver publicaciones

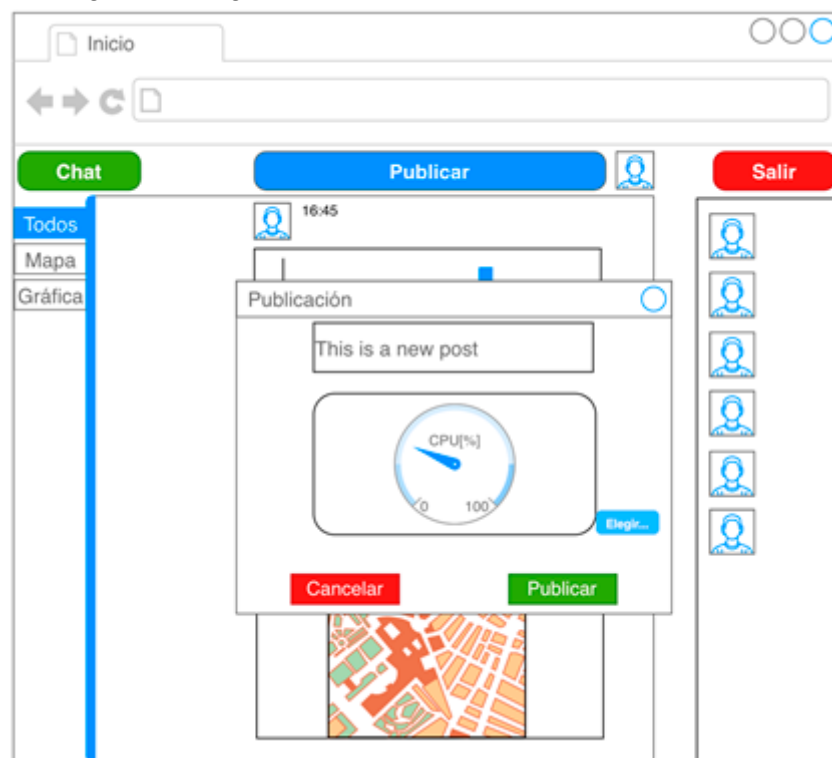
Se muestran todas las publicaciones que otros usuarios (solo amigos) han realizado, también se muestran las del propio usuario. Todas se muestran ordenadas por fecha de publicación de forma descendente (del más reciente al más antiguo).

Filtrar publicaciones

Se tendrá una lista de filtros, estas estarán basadas en la filtración de publicaciones por tags. Se deberá de buscar todas las publicaciones según el tag que indique el usuario.

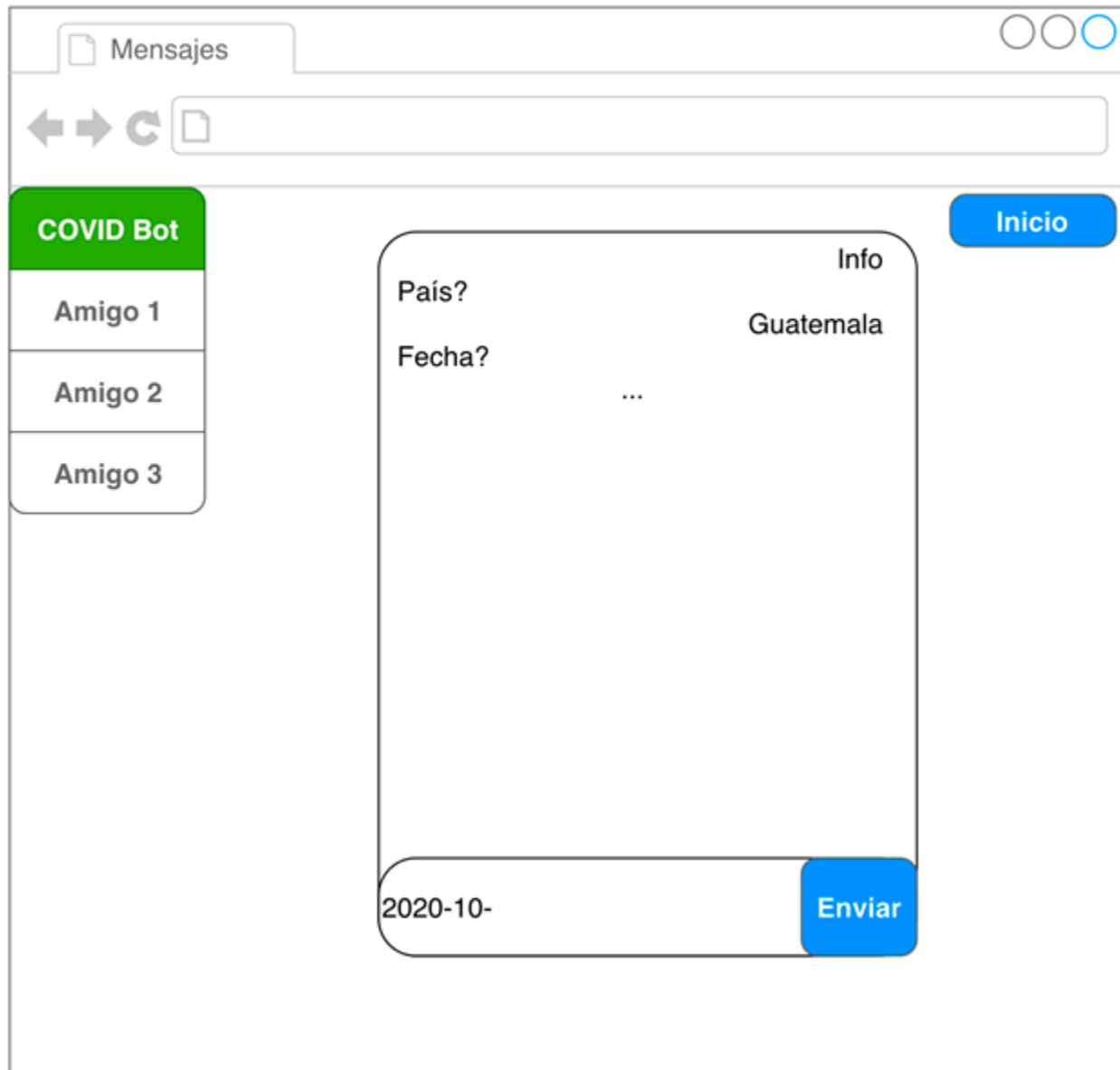
Crear publicaciones

Se tendrá una sección donde se creará una nueva publicación, esta deberá de tener siempre una imagen y el texto puede ser opcional, si se agrega una imagen tiene que brindar la opción que se le agreguen tags a la imagen.



Chat

En esta sección, el usuario puede comunicarse con sus amigos y usuarios Bots. Las conversaciones tienen que ser en tiempo real implementando Sockets.



Chat con amigos

Ambos usuarios amigos pueden enviar y recibir mensajes.

Chat con bots

Esta funcionalidad permite chatear con usuarios que tienen el Modo Bot activo, el chatbot se hará a puro código. A este se le podrán realizar las preguntas y pedirá los parámetros correspondientes:

- Casos
 - País
 - Fecha
 - Tipo de casos (confirmados, recuperados, muertes o todos)
- Gráfica de casos
 - País
 - Rango de fechas

ChatBot

A continuación se mostrarán un ejemplo para cada funcionalidad del Bot, las preguntas y la forma de responder a estar queda a elección del estudiante, pero tiene que seguir el flujo de preguntas definidas anteriormente. El bot consultará la información de los casos por COVID19 al json con la información previamente brindada.

Casos

Un ejemplo de comunicación para la funcionalidad “!Casos” por medio del chat puede ser el siguiente:

Casos	
País?	Guatemala
Fecha?	hoy (2020-04-17)
Tipo de casos (confirmados, recuperados, muertes o todos)?	todos
214 confirmados, 19 recuperados y 5 muertes en Guatemala el día 17/04/2020	

El usuario escribe envía la palabra “Casos” como mensaje al Bot, de tal forma que este empiece a realizar las preguntas y el usuario responda con los parámetros.

Gráfica de casos

Un ejemplo de comunicación para la funcionalidad “Gráfica de casos” por medio del chat puede ser el siguiente:

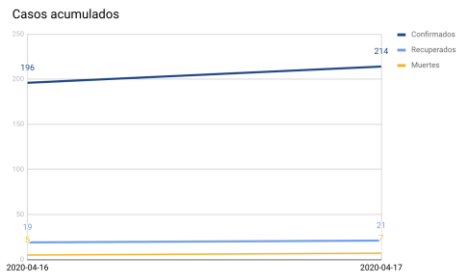
Gráfica de casos

País?

Guatemala

Rango de fechas?

2020-04-16 a 2020-04-17



El usuario escribe envía la palabra “!Gráfica de casos” como mensaje al Bot, de tal forma que este empiece a realizar las preguntas. Para la gráfica de este, se debe de utilizar una Gráfica Lineal y puede ser mostrado en el mismo chat o en una ventana emergente, queda a elección del estudiante.

*Los datos mostrados en la gráfica siempre son acumulativos, es decir son los valores de (todos los tipos de) casos totales que se tienen en esas fechas. En el eje x se debe mostrar todas las fechas (no solo el inicio y el fin) en ese rango de fechas.

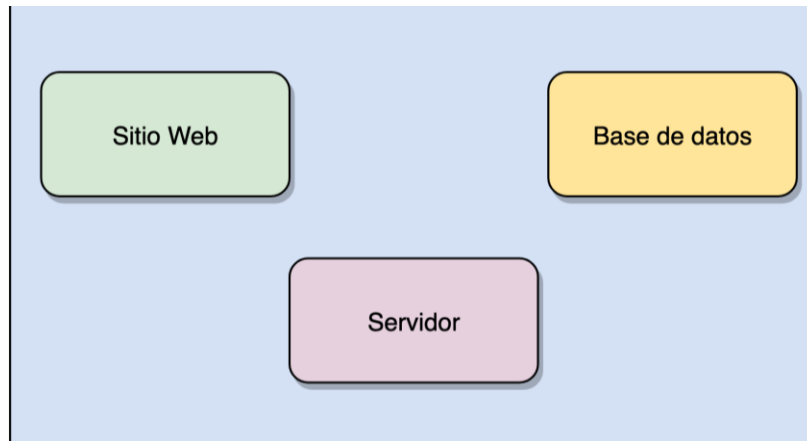
Stored Procedure

Se deberá implementar un stored procedure para el login y registro de usuarios.

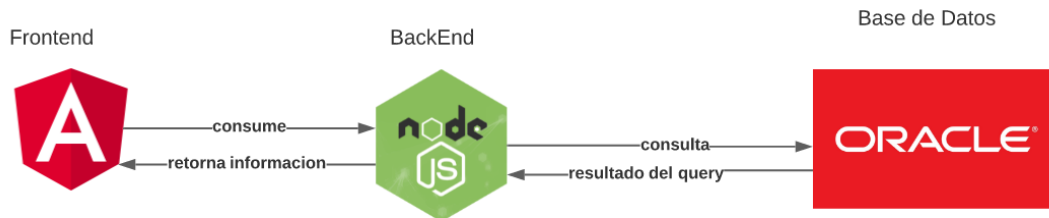
Implementación

Los servicios deben de ser implementados utilizando :

- Angular para el Sitio Web
- Nodejs para el Servidor
- Oracle para la base de datos



La arquitectura es la siguiente:



Documentación

Debe ser realizado en formato **Markdown** en el README.md del repositorio privado y debe contener lo siguiente:

- Describir la arquitectura implementada.
- Entidad Relación y descripción de cada tabla utilizada
- Describir los endpoints que se consumiran.
- Describir el Stored Procedure.
- Screen de la versión de Oracle Utilizada

ENTREGABLES:

- Documentacion en formato Markdown
- Entidad Relación
- Script con las sentencias DDL utilizadas.
- Script con las sentencias DML utilizadas
- Código Fuente de la Aplicacion

CONSIDERACIONES:

- Para el backend se utilizara Nodejs.
- Para el frontend se utilizara Angular.
- Oracle en cualquier version mayor o igual a 12c en su version Express para la base de datos. El metodo de instalacion queda a discrecion del estudiante.
- Subir el proyecto a un repositorio de github en modo **privado** y documentado con el formato [Markdown](#).
- El proyecto es individual.
- **Fecha de entrega: Miércoles 30/06/2021 hasta las 23:59:59 por UEDi/Classroom.**