

Universidad de San Carlos de Guatemala
Facultad de ingeniería
Escuela de Ciencias y Sistemas
Seminario de Sistemas 1
Segundo Semestre 2022

Sección A

Ing. Manuel Fernando López Fernández
Aux. Aldair Estrada García

Sección B

Ing. Ludwing Federico Altan Sac
Aux. Edgar Rolando Herrera Rivas



Proyecto 1

Super-Storage

Objetivos

- **General**
 - Cloud permite la innovación a través del acceso a nuevas tecnologías de forma rápida, segura y fácil.
- **Específicos**
 - Aplicar los conocimientos adquiridos tanto de la clase magistral y el laboratorio.
 - Implementar una arquitectura utilizando servicios en la nube.
 - Integrar servicios de AWS
 - Aplicar las tecnologías de la nube a un entorno real.

Descripción

Amazon Web Services (AWS) es una plataforma de servicios de nube que ofrece potencia de cómputo, almacenamiento de bases de datos, entrega de contenido y otra funcionalidad para ayudar a las empresas a escalar y crecer.

Actualmente existen diversas formas y situaciones donde se necesita e implementan los servicios de la nube, como estudiantes de Seminario de sistemas 1 y teniendo los conocimientos básicos de diferentes servicios que nos brinda AWS se requiere que defina, desarrolle e implemente una aplicación para el Almacenamiento de archivos públicos y privados con las opciones de agregar amigos y poder ver sus archivos compartidos y con la posibilidad de compartir nuestros propios archivos para que estos los puedan ver y también tener la opción de guardar archivos en modo secreto.

Los servicios de AWS que se utilizaran para este proyecto son:

- IAM
- EC2
- Load Balancer
- S3
- RDS

SITIO WEB

Las siguientes imágenes son ejemplo de cómo es la aplicación visualmente, junto con su descripción de que debe de realizar cada pantalla, queda al **criterio del estudiante el diseño de su aplicación**.

- **Login**
 - El usuario podrá ingresar a su cuenta con su **Nombre de Usuario o Correo electrónico**.
 - Verificar que el usuario exista.

El diagrama muestra una ventana de navegador con la URL `www.seminario1.com`. Dentro de la ventana, hay un formulario de login con el título "Login". El formulario contiene dos campos de entrada: "Usuario" y "Contraseña:". Debajo de estos campos hay un botón "Entrar" y un enlace "Registrarse!!!".

- **Registro**

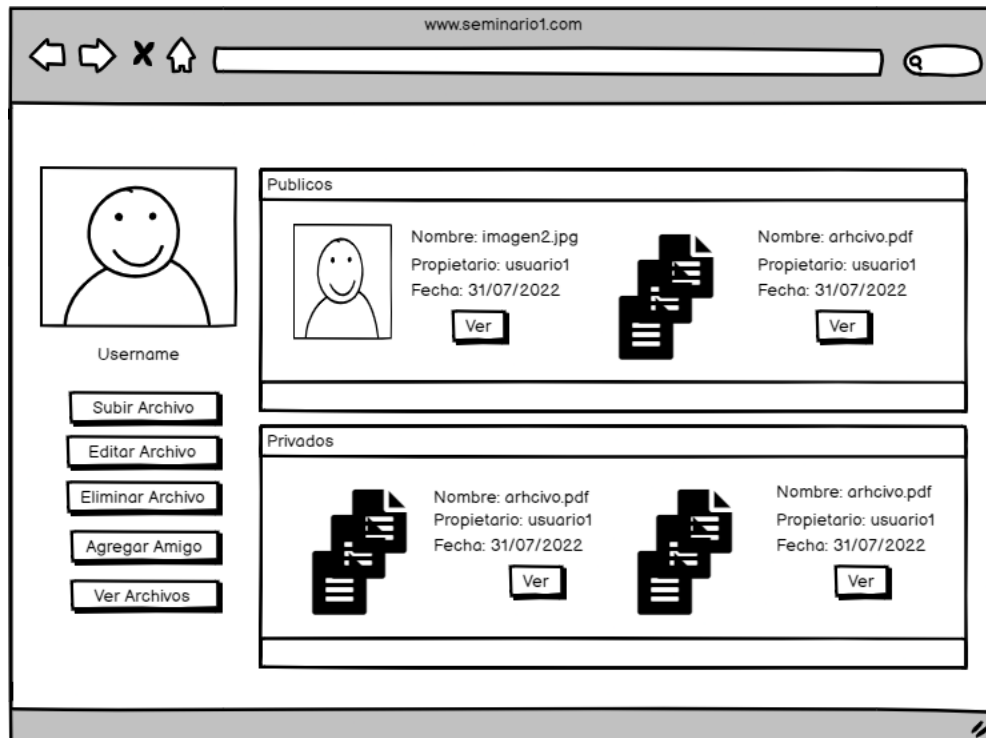
- La pantalla de registro debe de pedir lo siguiente:
 - Nombre de usuario
 - Correo electrónico
 - Contraseña
 - Fotografía de perfil
- Se debe de tener un cuadro para confirmar la contraseña y validar que las 2 coincidan para poder crear el usuario.
- La foto de perfil es un campo **Obligatorio**.
- La contraseña debe de estar **ENCRYPTADA** en la base de datos.

El diagrama muestra una interfaz de usuario para el registro en un navegador web. La barra de direcciones superior muestra la URL "www.seminario1.com". El formulario principal, titulado "Registro", contiene los siguientes elementos:

- Campos de entrada para "Usuario" y "Nombre".
- Campos de entrada para "Contraseña:" y "Repetir Contraseña:".
- Un cuadro de imagen para la fotografía de perfil, que contiene un icono de una persona sonriente.
- Un botón "Subir Foto" ubicado a la derecha del cuadro de imagen.
- Un botón "Registrar" ubicado a la derecha del botón "Subir Foto".

- **Dashboard Pantalla Principal**

- En la pantalla principal del usuario se debe mostrar la foto de perfil como el nombre del usuario.
- Dentro de la pantalla principal se pueden ver los **archivos del usuario** divididos en 2 secciones. Archivos Públicos y archivos privados. Esto solo para diferenciarlos. Queda a criterio del estudiante como diferenciar estos archivos.
- Se tendrán 5 opciones dentro de esta pantalla.
 - Subir archivo
 - Editar Archivo
 - Eliminar Archivo
 - Agregar Amigo
 - Ver Archivos esta función se mostrarán los archivos públicos como privados, el cual deberá contar con un espacio propio para cada archivo



- **Subir Archivo**

- En esta pantalla se pedirá seleccionar un archivo a subir desde el ordenador.
- Se mostrará la ruta del archivo seleccionado.
- Se podrá modificar el nombre del archivo como será guardado.
- Se puede elegir la visibilidad del archivo en modo **público o privado**.
- Para poder subir un archivo se pedirá confirmar la contraseña esto es **obligatorio**.
- **Únicamente se podrán subir Imágenes, archivos de textos y documentos pdf.**

The image shows a web browser window with the address bar displaying "www.seminario1.com". The main content area contains a form titled "Subir Archivo". The form includes the following fields and controls:

- Nombre Archivo:** A text input field for the filename.
- Archivo Seleccionado:** A text input field showing the selected file's path.
- Tipo Archivo:** Two radio buttons labeled "Publico" and "Privado" for selecting the file's visibility.
- Contraseña:** A password input field with a masked display (asterisks).
- Subir Archivo:** A button located below a placeholder image of a person.
- Cancelar:** A button to cancel the upload process.
- Subir:** A button to confirm and upload the file.

- **Eliminar Archivo**

- En esta pantalla únicamente se pedirá seleccionar el archivo que se quiera eliminar.
- Para confirmar la eliminación de un archivo se necesita confirmar la contraseña del usuario.

www.seminario1.com

Subir Archivo

Archivos Seleccionado:

Archivo1.pdf

Cancelar

Contraseña:

Eliminar

- **Editar Archivo**

- En esta pantalla se podrá cambiar el nombre del archivo.
- Se podrá cambiar el **modo de visibilidad** del archivo de privado a publico y viceversa.
- Se necesita confirmación de contraseña para editar el archivo.

www.seminario1.com

Editar

Seleccionar Archivo: notas.pdf

Nombre Archivo: NotasActuales

Tipo de Archivo: ☐ Publico ☐ Privado

Confirmar Contraseña: *****

Editar

< Regresar

- **Agregar Amigos**

- En la sección de agregar amigos, aparecerán todos los usuarios creados en la plataforma.
- Tendrá una opción de buscar a un usuario por su **username**.
- También contendrá un **contador** de cuantos archivos públicos tiene ese usuario y la opción de agregar a ese amigo.

www.seminario1.com

Agregar Amigos

Username

usuario1

Buscar

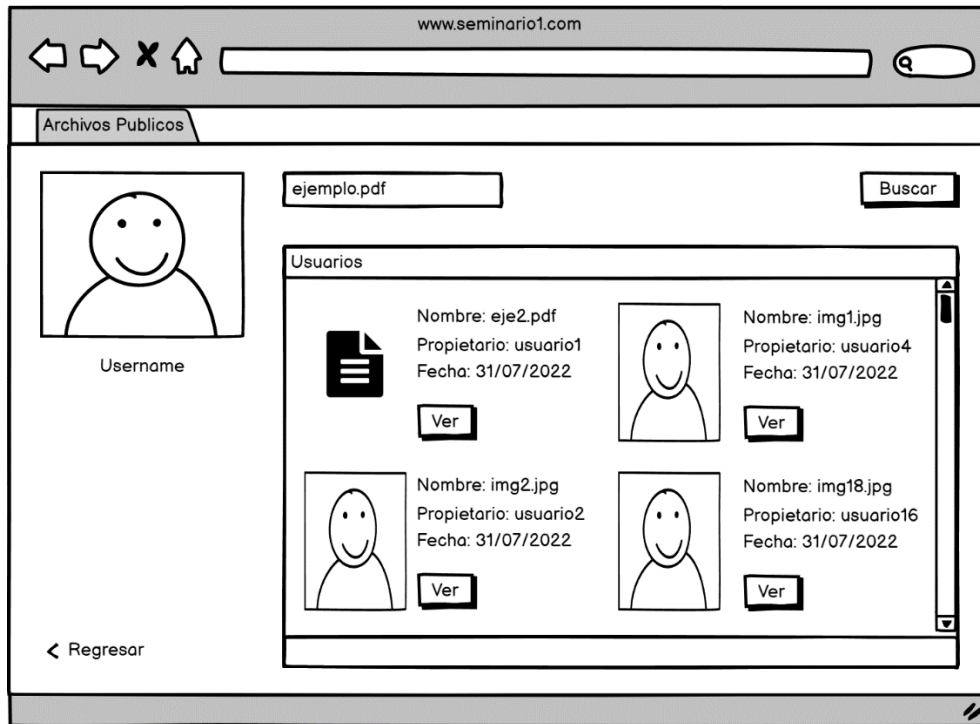
Usuarios

usuario1 5 archivos publicos Agregar	usuario5 3 archivos publicos Agregar
usuario3 20 archivos publicos Agregar	usuario4 18 archivos publicos Agregar

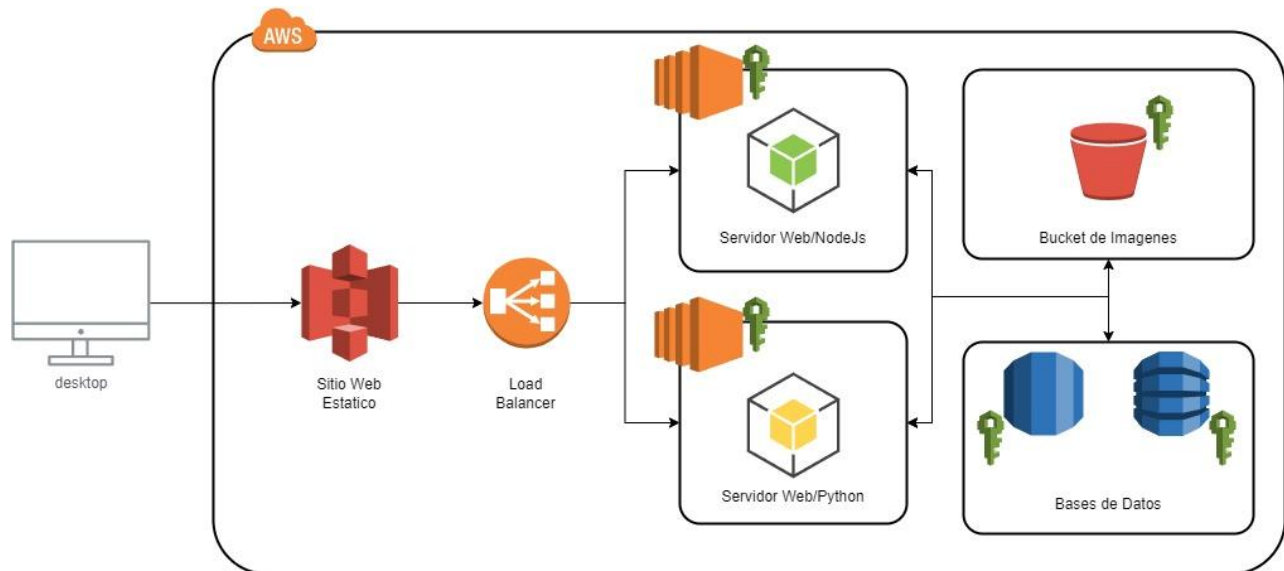
< Regresar

- **Visualización de Archivos**

- En esta sección aparecerán todos los archivos públicos de los amigos que hayamos agregado. **SOLAMENTE LOS ARCHIVOS PUBLICOS DE AMIGOS AGREGADOS.**
- Tendrán una opción de poder ver estos archivos ya sea una imagen o un archivode texto .txt o .pdf, que contara con un espacio propio para ver dichos archivos (misma pantalla que se menciona en Dashboard Pantalla Principal)
- Contendrá también una opción de buscar por el nombre del archivo.



Arquitectura e Implementación



- La arquitectura anterior demuestra como funcionarían todos los servicios de AWS internamente en nuestra aplicación.
 - **S3:**
 - Se tendrá nuestra aplicación dentro de un bucket de S3 funcionando como un **sitio web estática**. El nombre de este debe ser **appweb-grupo#-P1**
 - Se tendrá otro bucket donde se almacenarán todos los archivos y todas las imágenes tanto de perfil como las que se suban desde la app, configurado con políticas públicas para poder obtenerlos desde nuestra aplicación. El nombre de este será **archivos-#grupo-P1**.
 - **EC2:**
 - Se tendrán 2 instancias de EC2 donde se montarán 2 servidores con exactamente las mismas rutas de nuestra aplicación ya que esto nos permitirá enlazar nuestro balanceador de carga.
 - El lenguaje de los servidores debe de ser en **NODEJS y PYTHON**.
 - Asegurarse de habilitar **únicamente** los puertos necesarios para su aplicación en los **Security Groups** de cada instancia.
 - El sistema operativo de las instancias queda a discreción del estudiante.
 - **Load Balancer:**
 - Se configurará 1 balanceador de carga donde estará verificando el estado de los 2 servidores de las EC2.
 - Este es el que se tiene que consumir desde la aplicación, ya que como se sabe este redirecciona la solicitud a alguno de los servidores disponibles. Se verificará que es el que se consume desde la aplicación y no los servidores como tal.

- **RDS o DYNAMODB:**
 - Se utilizará una instancia de RDS O DYNAMODB, queda a discreción del estudiante para la base de datos.
 - Las contraseñas de los usuarios **deben de estar encriptada** con cualquier método o librería que el estudiante desee.
 - Para guardar los archivos e imágenes de un usuario solamente deben de guardar la URL de este donde esta almacenado.
 - El motor de la base de datos queda a criterio del estudiante.
- **IAM**
 - Se debe de crear los usuarios de IAM que sean necesarios para el manejo y uso de los servicios de aws que lo requieran con su política asociada.

Manual Técnico

Se necesita que se realice un manual técnico que incluya el siguiente contenido:

- Objetivos del Manual
- Explicación de Arquitectura del proyecto.
- Descripción de cada usuario de IAM creado con las políticas asociadas.
- Capturas y descripción de cómo se configuro cada servicio.
- Conclusiones

Manual de Usuario

Se necesita que se realice un manual de usuario que incluya el siguiente contenido:

- Objetivos del Manual
- Explicación y descripción de la aplicación.
- Pasos con capturas de cómo utilizar la aplicación.

CONSIDERACIONES:

- Nombre de Repositorio: **Semi1-Grupo#-Proyecto1**
- Repositorio en Github en modo privado y documentado con el formato Markdown (Manual de configuraciones y de Usuario).
- Agregar como colaborador en el repositorio al usuario dependiendo de su Sección de Laboratorio .
 - Aldair2011– Sección A
 - EH0809 – Sección B
- El proyecto puede ser en Grupos
- Usar los respectivos usuarios de IAM con sus respectivas políticas de acuerdo con el servicio que se está utilizando.
- Cualquier copia total o parcial será reportada a la escuela y se obtendrá una nota de 0.
- Fecha de entrega: 11/09/2022 a las 23:59 por **UEDI**