

Universidad “Politécnica Salesiana”

Carrera de Ingeniería de Sistemas

Título:

Desarrolló un sistema con Arquitectura JEE para la creación de un software del tipo financiero.

Integrantes del equipo:

Edison

Huiñaizaca

Cuenca, 16 de febrero de 2021

1. Chárter:

- OE1. Crear sistema web.

No.	Actividad
1	Estudiar y analizar la arquitectura Java EE para el desarrollo de páginas web, y manejo de páginas JSF.
2	Estudiar el uso del servidor WildFly
3	Prototipado de subsistema de administración web.
4	Usar persistencia en la aplicación para almacenamiento en la base de datos.
5	Validación de capa de negocio a nivel standalone.
6	Testear el app

Tabla 1 Actividades del Objetivo Específico

- OE2. Crear el sistema móvil

No.	Actividad
1	Estudiar api o protocolos [POP3, SMTP e IMAP] para envío de correo desde java.
3	Prototipado de subsistema de banca móvil
6	Testeo del la app
7	Validaciones

Tabla 2. Actividades del Objetivo Específico

Glosario

Web	La World Wide Web (del inglés, Telaraña Mundial), la Web o WWW, es un sistema hipertexto que funciona sobre Internet. Para ver la información se utiliza una aplicación llamada navegador Web para extraer elementos de información
Servidor	Un servidor es un tipo de software que realiza ciertas tareas en nombre de los usuarios. El término servidor ahora también se utiliza para referirse al ordenador físico en el cual funciona ese software, una máquina cuyo propósito es proveer datos de modo que otras máquinas puedan utilizar esos datos.
JEE	(Java Platform, Enterprise Edition) es una plataforma de programación, para desarrollar y ejecutar software de aplicaciones en el lenguaje de programación de java.

JSF	(JavaServer Faces)es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.
Usuario/Cliente	Persona quien va a ser beneficiaria de del desarrollo de la aplicación
Sistema	Un sistema es un objeto compuesto cuyos componentes se relacionan con al menos algún otro componente; puede ser material o conceptual
Api	Interfaz de programación de una aplicación, es un conjunto de procedimientos que son utilizados por otro software como una capa de abstracción
Software	Soporte lógico de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición

Justificación (Motivación)

Los servicios en línea hoy en día han adquirido gran popularidad, debido a que se ha generado una dependencia en las personas, ya que facilita realizar actividades comerciales, bancarias o de distinta índole sin importar donde se encuentre, debido a esto Simón Bank ofrece servicios mediante la creación de una aplicación web haciendo uso de tecnologías y tendencias, creando una aplicación web que sea segura, rápida y brinda mayor confiabilidad a los usuarios, facilitando de cierta manera realizar transacciones, ver información de sus cuentas,etc.

Objetivos

- **General**

Construcción de un sistema con Arquitectura JEE para SimonBank, contando con procesos administrativos de la cooperativa, y servicios para los clientes en la web y en una aplicación móvil.

- **Específicos**

- OE1. Crear el subsistema de administración web, el cual nos servirá para la administración de los distintos roles existentes dentro nuestra aplicación,
- OE2. Crear el subsistema de banca móvil en la web, esto nos servirá para ofrecer nuestros servicios mediante el uso de una página web, compatible con cualquier navegador haciendo uso de distintas tecnologías, como html y bootstrap.
- OE3. Crear el subsistema de banca móvil en app, mediante esta aplicación nos permitirá gestionar desde los distintos dispositivos móviles transferencias bancarias desde nuestras cuentas hacia otras cuentas.

Propuesta de solución

El proyecto está conformado por 4 etapas, la **primera** consiste en diseñar un prototipo que sea de fácil interacción con el usuario, para de acuerdo a esto poder crear la aplicación.

La **segunda** etapa consiste en identificar las entidades y desarrollar la aplicación para gestionar las transacciones, haciendo uso del modelo de programación definido por entidades, DAO(Object Access data) , ON(objeto de negocio) y la interfaz que consume dicha lógica.

Donde las entidades son las tablas que existen en la base de datos, estas poseen atributos propios, cada fila en la base de datos es un atributo.

En los DAO, se realiza la lógica para mandar a insertar , actualizar, leer o remover y estas son gestionadas en el servidor, mediante un archivo de persistencia contra la base de datos.

La tercera etapa es desarrollo de la aplicación web, la misma que va a estar desarrollada en lenguaje java orientada a arquitectura JEE es decir usaremos clases Servlet y Jsp ayudadas con javascript. Destacamos que para el diseño usaremos las librerías de bootstrap. Mediante la interfaz web podrán acceder usuarios finales tales como cajeras, jefe de créditos y clientes de la cooperativa.

La cuarta fase es el diseño e implementación de una aplicación móvil la cual será desarrollada en Android y consumirá un api de las aplicaciones ya creadas en las etapas anteriores, se seguirán usando los mismos objetos y los modelos de negocio. Por lo que la mitad de esta etapa de enfocar en la creación del api y la otra mitad será la implementación de la misma pero ya en la app desarrollada en Android studio.

Hemos estado hablando sobre el uso de una aplicación distribuida desarrollada por capas como lo es: Capa de presentación, Capa de negocios y Capa de acceso a datos, por lo que a continuación explicaremos cómo serán implementadas dichas capas en nuestra aplicación.

Interfaz de interacción con el usuario o capa de presentación. Conformado por la página web y aplicación móvil, es la capa encargada de recopilar y mostrar la información a los usuarios, la capa de presentación se encargará de consultar en la nube mediante web services según la interacción del usuario, esta será la interfaz que usará el usuario final del sistema.

Lógica de negocio o capa de negocio. De acuerdo al uso de la capa de presentación esta capa toma la información y de acuerdo a reglas y procedimientos válida dicha información o establece ciertos parámetros para esta información la misma que puede ser enviada a la capa de datos o regresar a ser presentada al usuario de otra manera en la capa de presentación. Se destaca que esta capa es la que obtiene los datos de capa de datos y envía a la capa de presentación, siendo a veces filtrada la **Capa de servicio de datos o capa de datos**. La capa de servicios de datos es la encargada de persistir o dicho de otra manera de guardar la

información en una base de datos pudiendo ser almacenada en servidores locales o servidores en la nube. Pero esta capa se encarga de que los datos sean almacenados, borrados o actualizados según lo define la capa presentación. En nuestra aplicación utilizaremos tecnología hibernate y JPA que nos ayudan a gestionar los datos, la cual explicaremos a continuación.

Persistencia Orientada Objetos. Por lo general cuando un desarrollador quiere hacer un ingreso de datos lo hace mediante consulta con una propiedad llamada INSERT o cuando necesita extraer datos lo hace mediante Querys(Consultas) lo que le permite extraer datos para así generar objetos y trabajar con los mismos, siendo esto un trabajo tedioso y estar pendiente de la sintaxis, es donde nace la persistencia la cual nos permite insertar objetos y a su vez extraer dichos objetos con los cuales se está trabajando ya en las diferentes capas mencionadas anteriormente. Haciendo que sea más rápido el desarrollo de las aplicaciones y más ágil.

Restricciones

Las siguientes restricciones son aplicadas a este proyecto:

- El sistema de banca móvil “SimonBank” deben ser compatibles con las plataformas operativas a nivel web.
- Las etapas del proyecto deben cumplirse en los tiempos estimados.
- El performance a la hora de las transacciones
- Un mismo usuario usando varias interfaces que podrían alterar lo datos. (Ejm. Un usuario usando el cajero automático a la vez que hace un transacción en línea)
- Los módulos deben cumplir con los requerimientos al 100%
- Control de bitácora de los usuarios.

Supuestos

A continuación, se lista los hechos que se consideran supuestos para el éxito del proyecto:

- Se cuenta con la documentación necesaria, y la ayuda de los docentes.
- Los desarrolladores conocen la metodología y la visión - misión de que el producto.
- Control de versiones, a través de la herramienta git, mediante el uso de github.
- El uso de SCRUM para el trabaja en equipo.

Metodología SCRUM

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa. Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación e innovación. Aplicando a nuestro proyecto diríamos que

esta metodología nos ayuda a que el equipo trabaje ordenadamente con lo requerimientos más importantes, lo a que al mismo tiempo permite que se cumplan los tiempos establecidos. Esta metódica de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades.

Diagrama organizacional

2. Cronograma

Nombre de tarea	Comienzo	Fin
Proyecto		
Tarea 1		
OE.1		
ACT. 5. Implementación de prototipado a páginas web. .	23/12/20	1/1/21
ACT. 6. Validación de capa de negocio a nivel standalone.	25-12-20	1-1-21
OE.2	25-12-20	1-1-21
ACT. 3. Prototipado de subsistema de banca móvil en la web.	25-12-20	1-1-21
ACT. 4. Generalización de bitácoras de acuerdo al ingreso y administración del usuario final del sistema.	25-12-20	1-1-21
OE. 3	25-12-20	1-1-21
ACT. 3. Implementación de prototipo en la interfaz	25-12-20	1-1-21
ACT. 4. Implementación de las ultimas funcionalidades	25-12-20	1-1-21

REVISIÓN / CORRECIÓN 2		
Tarea 2		
Inicio Revisión 3		
OE. 1		
ACT. 7. Testear el uso de la interfaz con los diferentes roles.	12-1-21	14-1-21
OE. 2	1-1-21	
ACT.5. Testear a gran escala para verificar el performance en transacciones bancarias	12-1-21	16-1-21
OE. 3		
ACT.5. Realización de la app móvil	12-1-21	20-1-21
OE. 3		
ACT. 7. Probar todos los módulos del sistema de SimosBank.	15-1-21	20-1-21

REVISIÓN / CORRECIÓN 5	15-1-21	20-1-21
------------------------	---------	---------

Prototipo

Loguin

En esta sección se puede apreciar los componen del logan al iniciar sesión



Sign In

User Name:
johndoe

Password:

SIGN IN

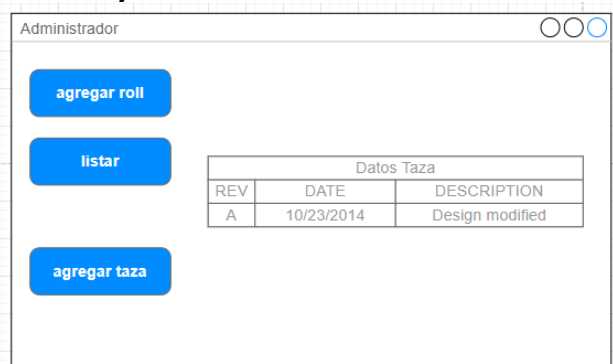
[Recuperrar clave](#)

New User

SIGN UP

Administrador

En la siguiente sección se puede apreciar la lista de la taza con sus debidos porcentajes



Administrador

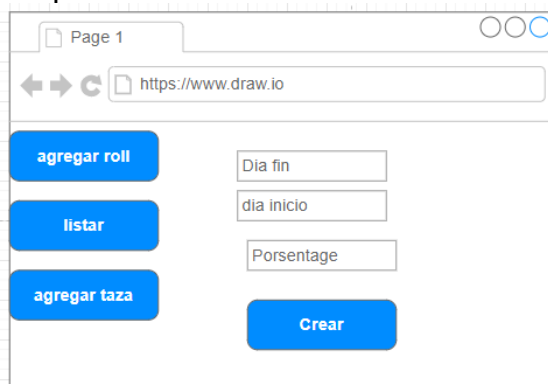
agregar roll

listar

Datos Taza		
REV	DATE	DESCRIPTION
A	10/23/2014	Design modified

agregar taza

En la sigiente seccion se podra apresiar el in greso de los datos consernientes ala polaiza



Page 1

https://www.draw.io

agregar roll

listar

agregar taza

Día fin

día inicio

Porcentage

Crear

Esta sección don de se muestra los componentes que conforman la parte de la creación de los usuarios administrador, cajero, asistente

Page 1

https://www.draw.io

agregar roll

listar

agregar taza

nombre

email

cedula

Telefono

Password

☐ Administrador

☐ Cajero

☐ Asistente

Crear

Cliente

Pagina que listara las pólizas y transaccione además de mostrar por fecha las transacciones

Cliente

Historial

Historial de ingreso

Poliza

Datos cuenta

REV	DATE	DESCRIPTION
A	10/23/2014	Design modified

Historial cuenta

REV	DATE	DESCRIPTION
A	10/23/2014	Design modified

Desde

Option 1

Hasta

Option 1

Listar

Historial

sección que mostrara el historial de ingreso del usuario

Cliente

Historial ingreso

Historial

Poliza

historial ingreso

REV	DATE	DESCRIPTION
A	10/23/2014	Design modified

Listar

Simulador de pólizas

Cliente

Poliza

Historial

Historial Ingreso

Días plazo

monto

Simular

Polisa simulador

REV	DATE	DESCRIPTION
A	10/23/2014	Design modified

Generar Póliza

Pagina que cargara los datos pdf

Cliente

Poliza

Historial

Historial Ingreso

Cargar datos pdf

Datos subidos

REV	DATE	DESCRIPTION
A	10/23/2014	Design modified

Finalizar Transaccion

Asistente

Ventana que listara todas las pólizas las cuales serán aprobadas posterior mente

Asistente

listar

Polizas

REV	DATE	DESCRIPTION
A	10/23/2014	Design modified

Ventana donde se aprobará o rechazara una póliza

Asistente

listar

Id

Estado

Pazo

Aprovar

Denegar

Actualizar

Cajero

Sección donde se depositara en una cuenta

The 'Cajero' window has a title bar with the text 'Cajero' and three window control buttons. On the left is a vertical menu with four items: 'Deposito' (highlighted in blue), 'Retiro', 'Crear Usuario', and an empty box. To the right of the menu are two input fields labeled 'numero de cuenta' and 'monto'. At the bottom center is a blue button labeled 'Depositar'.

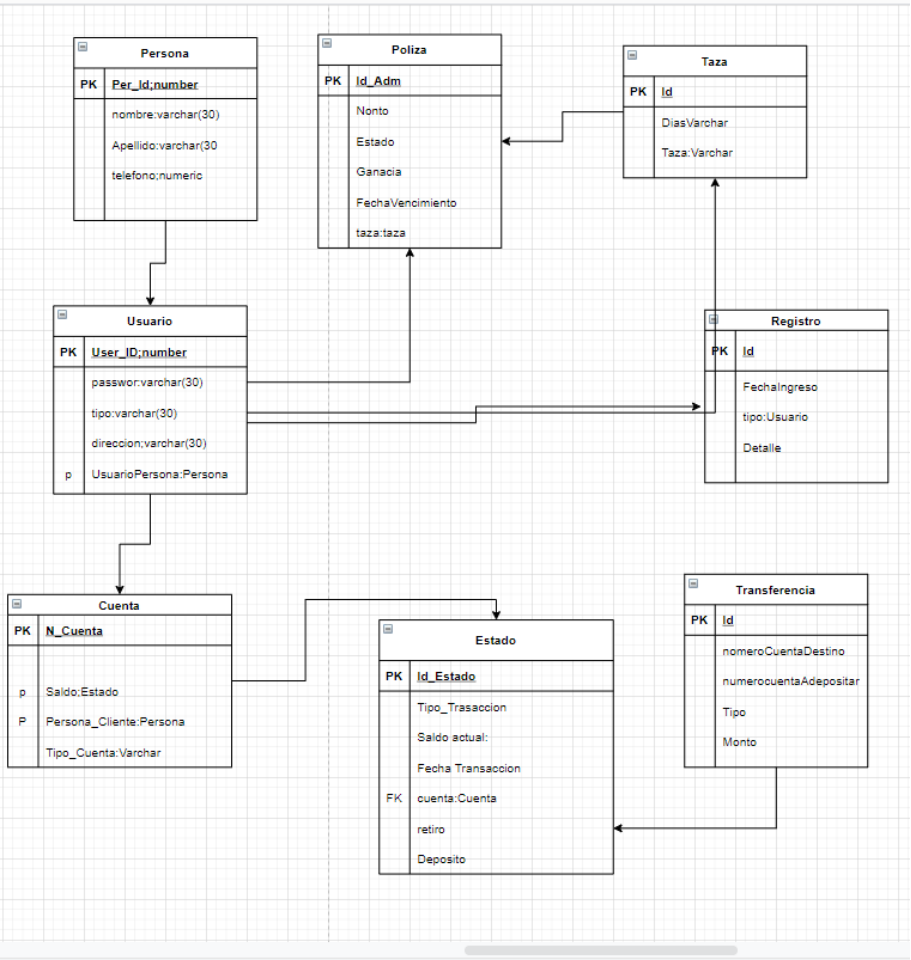
sección donde se hará un retiro de una cuenta

The 'Cajero' window has a title bar with the text 'Window Title' and three window control buttons. On the left is a vertical menu with four items: 'Retira' (highlighted in blue), 'Deposito', 'Crear Usuario', and an empty box. To the right of the menu are two input fields labeled 'numero de cuenta' and 'monto'. At the bottom center is a blue button labeled 'Retiro'.

sección donde se creará una cuenta corriente o ahorro

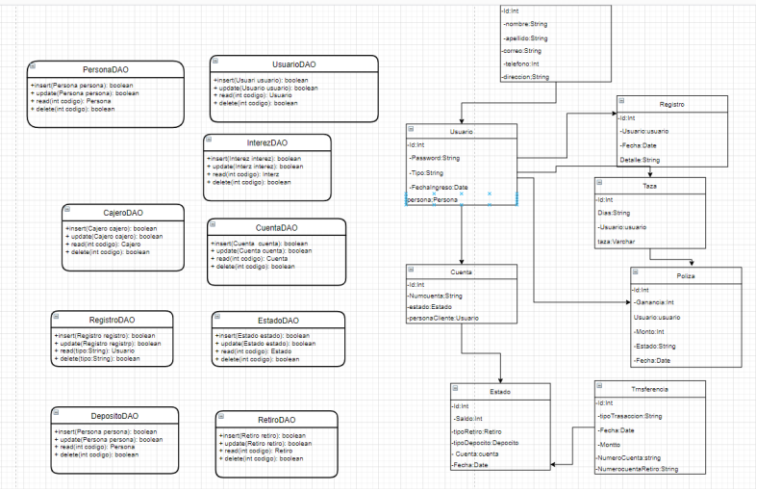
The window has a title bar with 'Page 1' and three window control buttons. Below the title bar is a browser-like address bar with navigation icons and the URL 'https://www.draw.io'. On the left is a vertical menu with four items: 'Grear Usuario' (highlighted in blue), 'Deposito', 'Retirar', and an empty box. To the right of the menu are five input fields labeled 'nombre', 'coreo', 'cedula', 'Telefono', and 'Password'. Below these fields are two radio buttons labeled 'Ahorro' and 'Coriente'. At the bottom center is a blue button labeled 'Crear'.

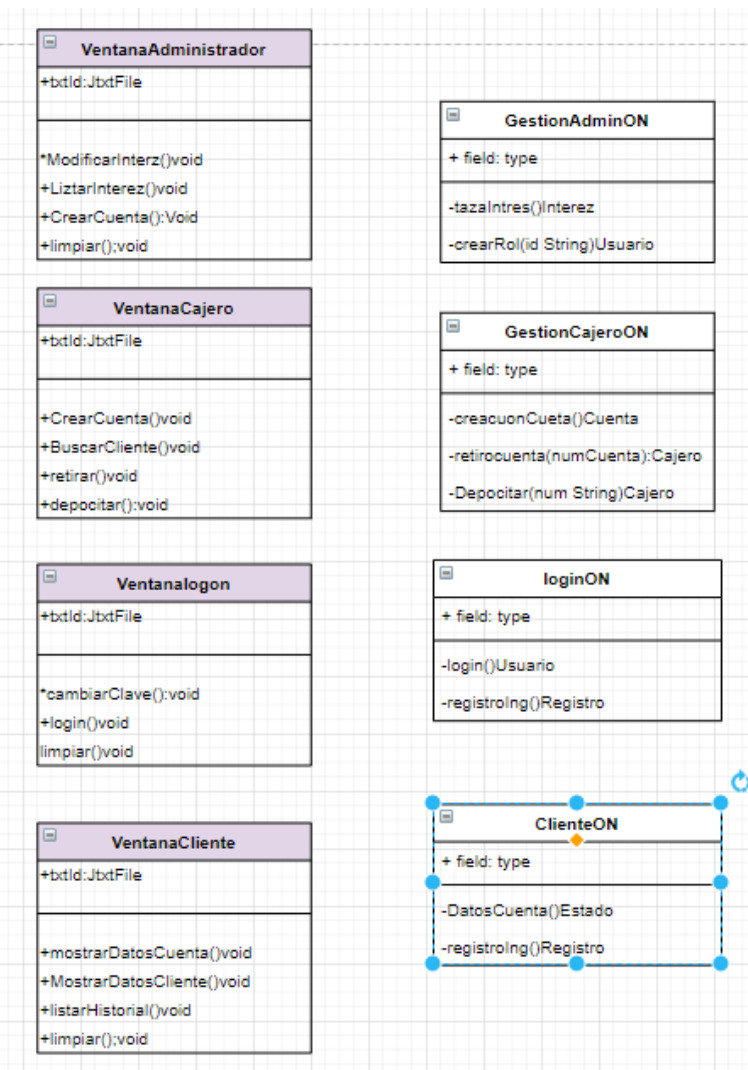
Diagrama de la base



Estructura del sistema




























Diagrama que representa la estructura del sistema siguiendo el esquema de negocios para java jee



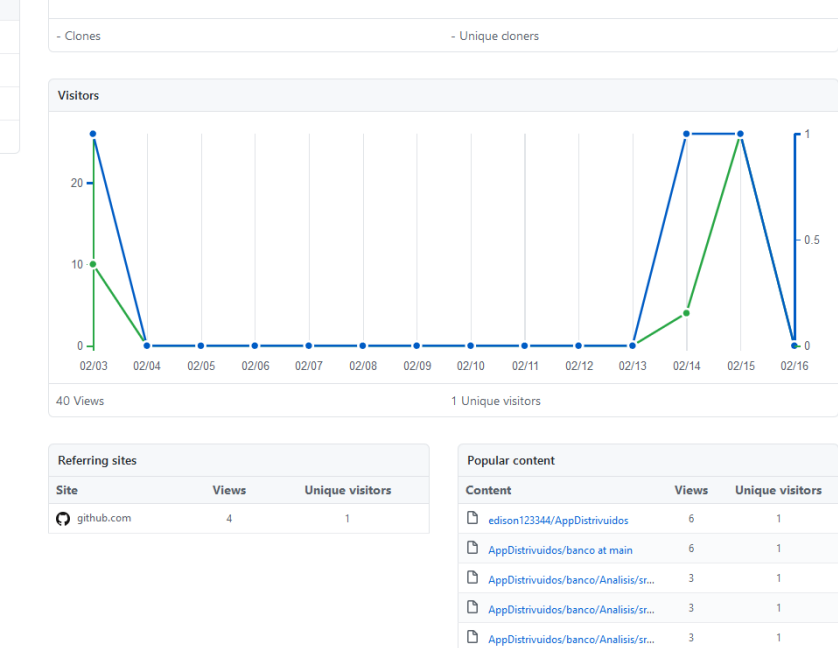


Avances hechos en gid

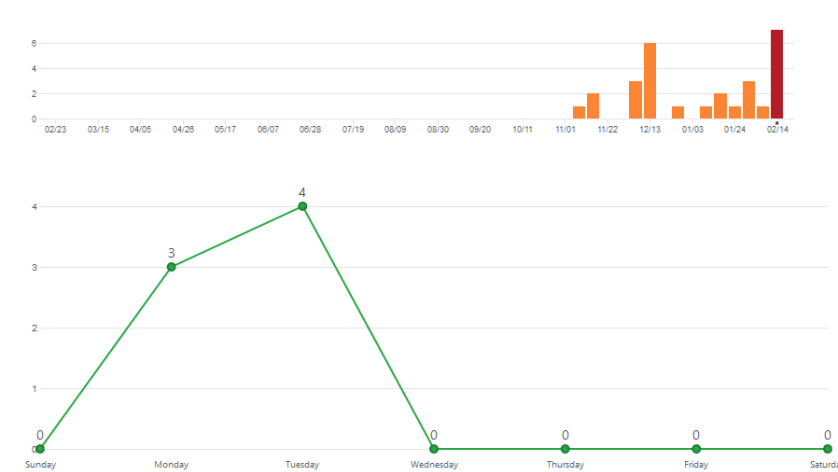
Commits echo salo largo del avance del proyecto

Commits on Feb 16, 2021	
actualisacion  edison123344 committed 8 minutes ago	 18c1340 
Actualisacion  edison123344 committed 15 minutes ago	 0fc17fa 
Update slinx.sqlite  edison123344 committed 16 minutes ago	 50a942e 
actu  edison123344 committed 17 hours ago	 b622199 
Commits on Feb 15, 2021	
actualizacion  edison123344 committed 20 hours ago	 54c3801 
actual  edison123344 committed yesterday	 999f135 
actualisacion  edison123344 committed yesterday	 d76a2a8 
Commits on Feb 10, 2021	
banca  edison123344 committed 6 days ago	 97f151e 
Commits on Feb 3, 2021	
examen  edison123344 committed 13 days ago	 21720e6 
Commits on Feb 1, 2021	

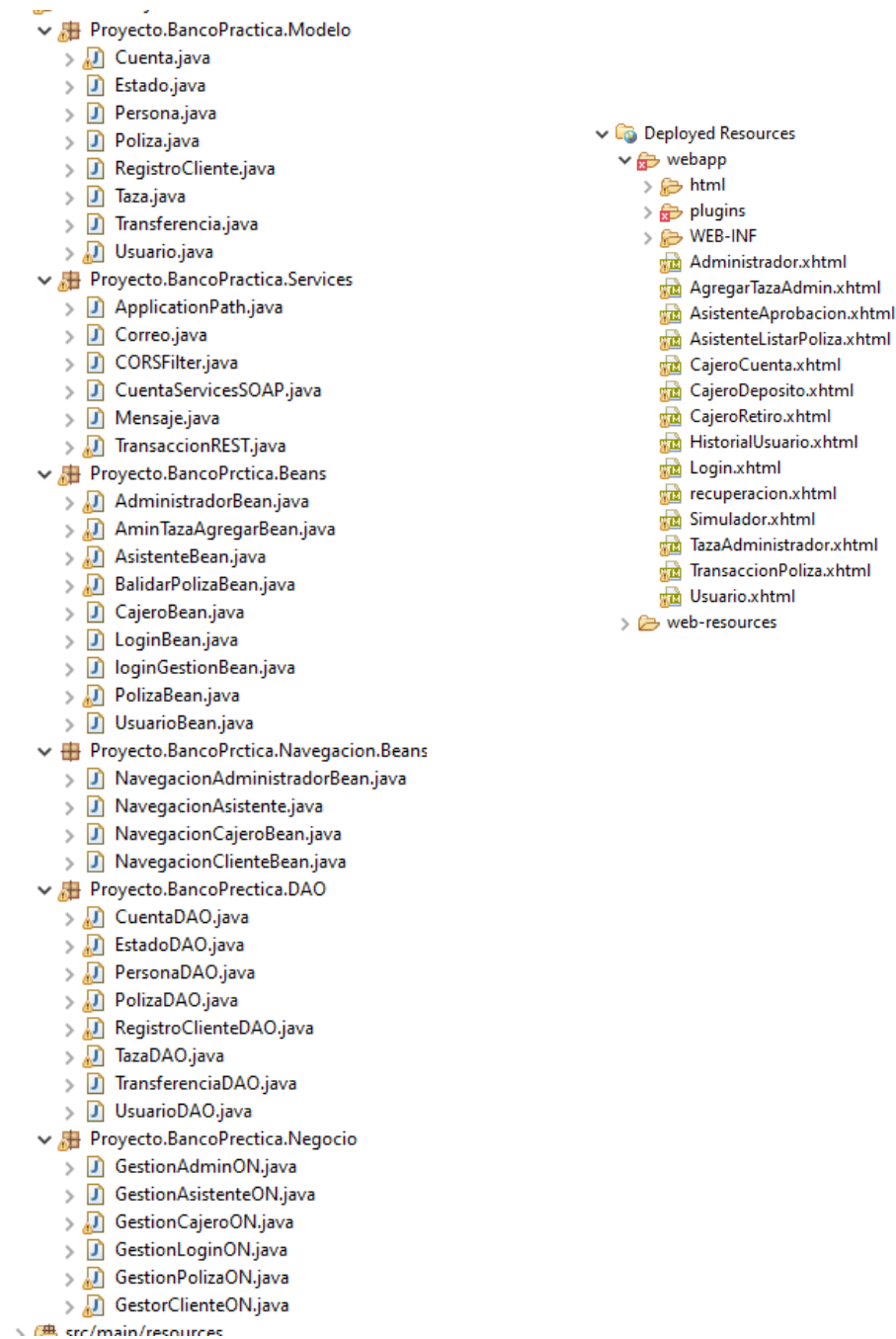
Grafica de actividad en el gid respecto al proyecto



Grafica de commits efectuados a lo largo dela ejecución de este proyecto



Estructura des sistema en java



Estructura del sistema móvil

Resultados de la sección Móvil Cambio clave según el correo

cambioClave
ingrese email
ENVIAR

Listado de la cuenta y polizasa

listado Informacion

≡

Informacion Cuenta

id:

saldo:0

tipo:Deposito

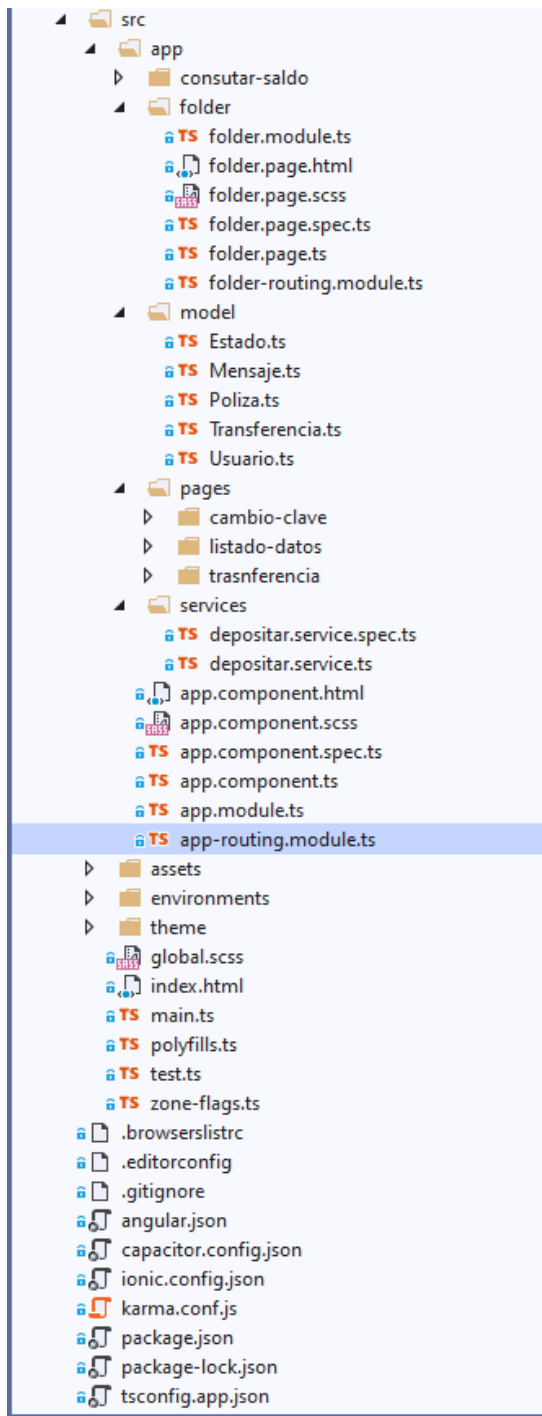
dia:1613365870000

Polizas

Transferencia

trasnferencia
Cuenta destinatario
Cuenta usuario
Monto
tipo
ENVIAR

Estructura del sistema movil



URL Repositorio:
Respaldo

<https://github.com/edison123344/AppDistrivuidos/tree/main/banco>

<https://github.com/edison123344/AppDistrivuidos.git>