	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTÁ		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		

PROGRAMA INGENIERÍA DE SISTEMAS PROYECTO DE AULA

NOMBRE DEL PROYECTO: SISTEMA DISTRIBUIDO CON CLIENTE MOVIL PARA EL MANEJO DE TRANSACCIONES ENTRE CUENTAS BANCARIAS

DESCRIPCIÓN DEL PROBLEMA: Un banco requiere de un sistema que les permita a sus clientes realizar transferencias entre cuentas y tener acceso a su historial de transacciones bancarias desde sus dispositivos móviles. El inconveniente es que es una entidad pequeña con una infraestructura pequeña por lo que piden usar solo una maquina física para implantar el servidor, pero sin tener la base de datos en el mismo lugar, de modo tal, en un futuro, se puedan crear más servidores que se conecten fácilmente a la base de datos.


OBJETIVO GENERAL: Construir un sistema distribuido que permita la realización de transferencias entre cuentas bancarias desde dispositivos móviles contemplando las limitaciones de infraestructura de la entidad bancaria.

OBJETIVOS ESPECÍFICOS:

- Diseñar los componentes del sistema distribuido, tales como la base de datos, API y la aplicación móvil en base a las necesidades de la entidad bancaria.
- Construir cada uno de los componentes diseñados.
- Implementar el sistema distribuido con cada uno de los componentes.
- Realizar pruebas para la mejora del sistema.

TEMÁTICAS:

1. **Cloud Computing:** Es el suministro de servicios informáticos (incluidos servidores, almacenamiento, bases de datos, redes, software, análisis e inteligencia) a través de internet (la “nube”), cuyo objetivo es ofrecer una innovación más rápida, recursos flexibles y economías de escala (Microsoft, 2021).
2. **Bases de datos:** Es una colección organizada de información estructurada, o datos, típicamente almacenados electrónicamente en un sistema de computadora. Una base de datos es usualmente controlada por un sistema de gestión de base de datos (DBMS). En conjunto, los datos y el DBMS, junto con las aplicaciones que están asociados con ellos, se conocen como un sistema de base de datos (Oracle Mexico, 2021).
3. **Servicios web:** Son aplicaciones modulares autocontenidas que puede describir, publicar, localizar e invocar a través de una red. Un escenario de servicios web típico es una aplicación empresarial que solicita un servicio de otra aplicación existente. La solicitud se procesa a través de una dirección web

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTA		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		

determinada usando mensajes SOAP a través de un transporte HTTP. El servicio recibe la solicitud, la procesa y devuelve una respuesta (IBM, 2021).


4. **Dispositivos móviles:** Son aparatos de tamaño pequeño (generalmente) que cuentan con características tales como capacidades especiales de procesamiento, conexión permanente o intermitente a una red, memoria limitada, diseños específicos para una función principal y versatilidad para el desarrollo de otras funciones. Tanto su posesión como su operación se asocian al uso individual de una persona, quien puede configurarlo según sus criterios y preferencias.
5. **Requerimientos del negocio:** Son especificaciones de alto nivel, establecidas por la organización que perciben necesidades o detectan oportunidades. Incluyen los objetivos del negocio, la visión del producto y el alcance del proyecto. Define el por qué y para qué se construye un sistema (Ferreira, 2015).
6. **Pruebas de seguridad:** Son pruebas en un sistema que evidencian las vulnerabilidades más típicas existentes en las aplicaciones, con el propósito de evitar que cualquier persona pueda aprovecharse de ellas para ingresar a un sistema.

COMPETENCIAS PARA DESARROLLAR:

1. Manejo de servicios de cloud computing
2. Desarrollo de bases de datos relacionales
3. Desarrollo de servicios web/APIs
4. Desarrollo de aplicaciones móviles con Android Studio






CONOCIMIENTOS PREVIOS REQUERIDOS:


1. Lenguaje SQL
2. Conectividad y red
3. Spring Boot
4. Android Studio
5. Conexión a maquinas remotas

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTA		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		


RECURSOS A NIVEL HARDWARE Y SOFTWARE:

Para la realización de este proyecto se definen los siguientes recursos, los cuales se emplearán en el diseño, desarrollo y verificación de los componentes del sistema distribuido:

Icono	Nombre del recurso	Descripción
	Google Cloud	Consiste en un conjunto de recursos físicos, como computadoras y unidades de disco duro, y recursos virtuales como VM, que se encuentran en los centros de datos de Google de todo el mundo.
	MySQL 5.7	Es un sistema de bases de datos relacional desarrollado bajo licencia dual por Oracle Corporation.
	MySQL Workbench 8.0	Es un entorno gráfico de MySQL que permite modelar, generar y gestionar bases de datos.
	PuTTY / PuTTYgen	Es un cliente SSH, Telnet, rlogin t TCP raw con licencia libre.
	Java 1.8	Es el estándar global para desarrollar y distribuir aplicaciones móviles y embebidas, juegos, contenido basado en web y software de empresa.

	UNIVERSIDAD DE SAN BUENAVENTURA SEDE BOGOTÁ		
	PROGRAMA:	ASIGNATURA:	TIEMPO: 2:00 H
	INGENIERIA DE SISTEMAS	SISTEMAS DISTRIBUIDOS	
PROYECTO DE AULA			

	Spring Boot – Spring Tools 4.2	<p>Es una herramienta que facilita el desarrollo de aplicaciones web y microservicios con el Framework Spring, de una manera más fácil y veloz.</p>
	Eclipse IDE for Enterprise Java Developers - 2020-09	<p>Es una plataforma de software compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar aplicaciones.</p>
	Android Studio 4.2	<p>Es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de apps para Android y está basado en IntelliJ IDEA.</p>
	Retrofit	<p>Librería que funciona como un cliente HTTP seguro para la comunicación con Spring Boot</p>
	Postman 8.5	<p>Es una plataforma API para construir y utilizar APIs.</p>

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTÁ		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		




	Equipos personales	<p>Son los equipos que van a servir como:</p> <ul style="list-style-type: none"> - Equipos de desarrollo. - Clientes. - Servidores.
	Github – Github Desktop	<p>Es un repositorio de proyectos mediante el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador.</p>

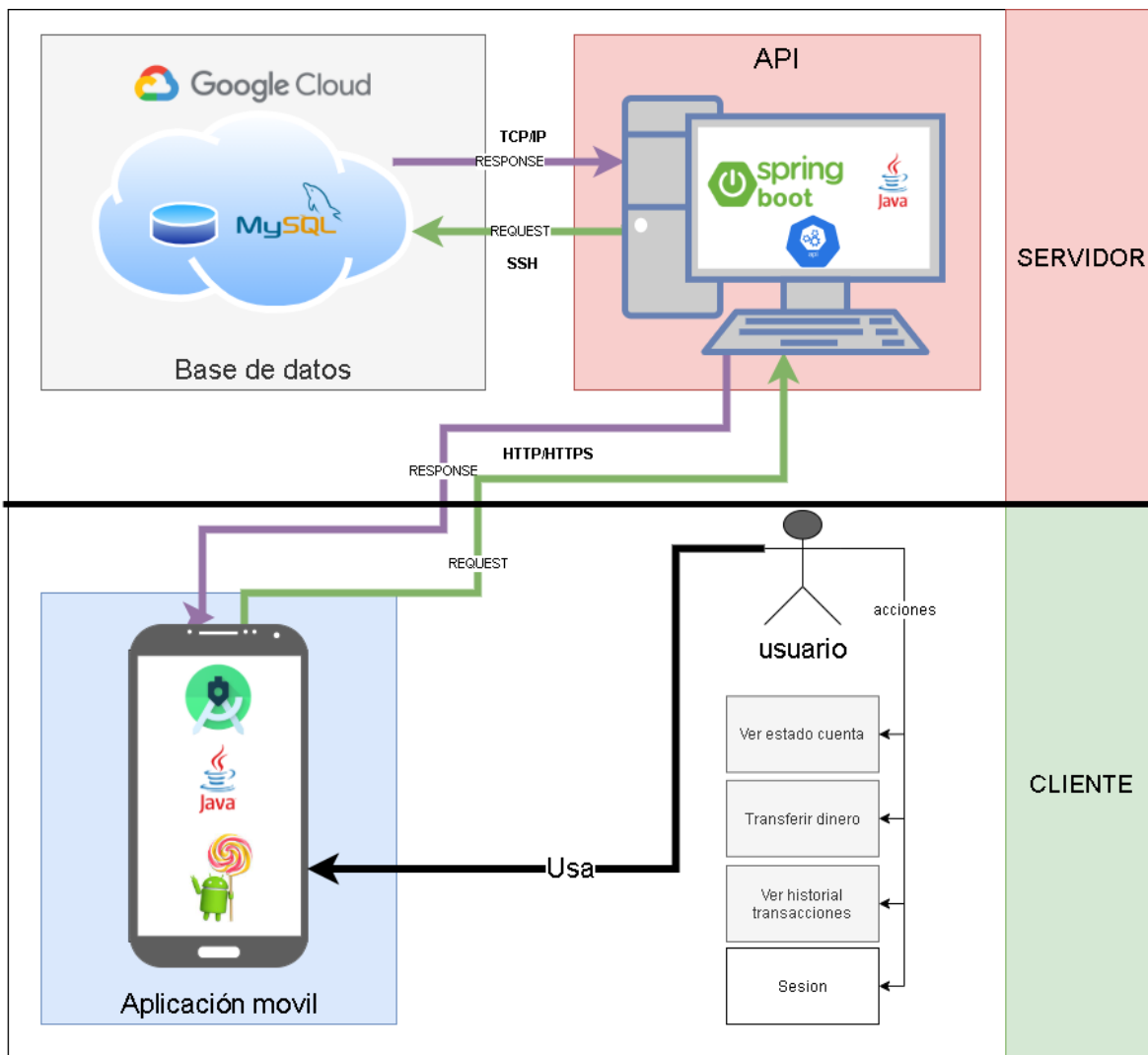
Ilustración 1 tabla de recursos de software y hardware

Nota: por medio de los nombres de cada uno de los recursos de software se puede tener acceso a documentación y archivos usados.


	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTA		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		

DISEÑO

Contemplando los objetivos y la descripción del problema, se definió el siguiente diseño del sistema distribuido, en el cual se definen los componentes y herramientas de software requeridas para el desarrollo, implementación y puesta en marcha de todo el sistema, en la siguiente imagen se podrán ver la forma en que el sistema se encontrara distribuido y la forma en que cada uno de los componentes se conectara y comunicaran entre sí, de modo tal se definirán también los protocolos y actividades de comunicación:




Arquitectura del sistema: este sistema distribuido contará con una arquitectura cliente-servidor, en la cual se tendrá del lado del servidor los componentes de la base de datos y la API que hará uso de esta, esta API será la que tendrá comunicación con el cliente que en este caso será una Aplicación móvil. Esta arquitectura y las tecnologías usadas

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTA		
	PROGRAMA:	ASIGNATURA:	TIEMPO: 2:00 H
	INGENIERIA DE SISTEMAS	SISTEMAS DISTRIBUIDOS	
	PROYECTO DE AULA		

garantizan que se podrá en un futuro incluir un nuevo cliente como lo sería un cliente web, y escalar en servicios la API existente.

- **Base de datos:** contemplando las limitaciones de infraestructura y la necesidad de tener los componentes de base de datos y servidor o API separados, teniendo en cuenta que solo existe un equipo físico, por esto se define que la base de datos MySQL se encontrara en una instancia de VM (Virtual Machine) en Google Cloud. Por la naturaleza de este ejercicio, se mantendrá activa la instancia de VM solo en los momentos en que se requiera tener acceso a la base de datos, y se accederá a esta por medio de protocolos TCP/IP y SSH, el primero se usara para el acceso por parte de la API y el segundo nos permitirá la manipulación de la instancia de forma remota.
- **API:** contemplando las limitaciones expuestas y la necesidad de un sistema distribuido, se construirá una API con Spring Boot, Framework para Java el cual permite la creación y manipulación de APIs de forma rápida. La API será probada con Postman con el fin de probar el correcto funcionamiento de los servicios que esta poseerá.
- **Aplicación móvil:** como cliente se tendrá una aplicación móvil desarrollada por medio de Android Studio, la cual hace manejo de lenguaje Java y XML para la manipulación de la interfaz. Para el consumo de la API Rest se hará uso de la librería Retrofit.
- **Usuario:** contemplando las necesidades del usuario, se especifican las acciones básicas del usuario para implementarlas en la aplicación móvil y definir los servicios en la API Rest.

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTÁ		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		

METODOLOGÍA

Para el desarrollo de este proyecto se implementará la metodología en cascada, la cual es útil en el desarrollo de sistemas, de modo tal se definan los elementos relacionados con los requerimientos, diseño e implementación de cada uno de los componentes, para realizar la verificación de cada uno de los componentes y luego una verificación del sistema distribuido en general. Las fases definidas en esta metodología son las siguientes:

Metodología en cascada

El modelo en cascada es la propuesta de un enfoque metodológico que consiste en ordenar de forma lineal las distintas etapas que se deben seguir al momento de desarrollar software (Royce, 1970).

1. **Definición de requerimientos:** Consiste en la fase de análisis primario, que contempla el estado inicial del proyecto, los recursos disponibles, los requerimientos de negocio (objetivos del proyecto, justificaciones y estado final del proyecto) y el tiempo disponible para su desarrollo.

En esta fase se obtendrá como resultado las tablas de requerimientos funcionales y no funcionales del sistema distribuido.

2. **Diseño de componentes:** Consiste en analizar los recursos disponibles del proyecto y determinar cuales se utilizarán teniendo en cuenta los requerimientos de este. Luego, se utilizan dichos recursos para construir los componentes necesarios del sistema.

En esta fase se obtendrán los siguientes productos:

- Modelo relacional de la base de datos
- Listado de servicios de la API Rest
- Diseño de interfaces graficas de la aplicación móvil

3. **Implementación:** Consiste en acoplar los componentes para realizar la aplicación funcional del sistema en el contexto del proyecto.


En esta fase se obtendrán los siguientes productos:

- Instancia de VM con la base de datos MySQL creada y configurada
- API Rest en Spring Boot configurada
- Aplicación móvil desarrollada en Android Estudio.

4. **Verificación:** Consiste en realizar las pruebas de funcionamiento sobre el sistema en cuestión, con el fin de detectar y corregir errores, implementar o inhabilitar funciones y optimizar el sistema (de ser posible).

En esta fase se obtendrán los siguientes productos:

- Pruebas realizadas a la API Rest en Postman
- Pruebas funcionales de la aplicación móvil

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTÁ		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		

5. **Mantenimiento:** Consiste en verificar el sistema funcional sobre el contexto del proyecto para verificar que funcione correctamente, y de ser necesario, corregir errores y vulnerabilidades.

En esta fase se obtendrán los siguientes productos:

- Pruebas de seguridad y conectividad a los componentes
- Revisión de componentes

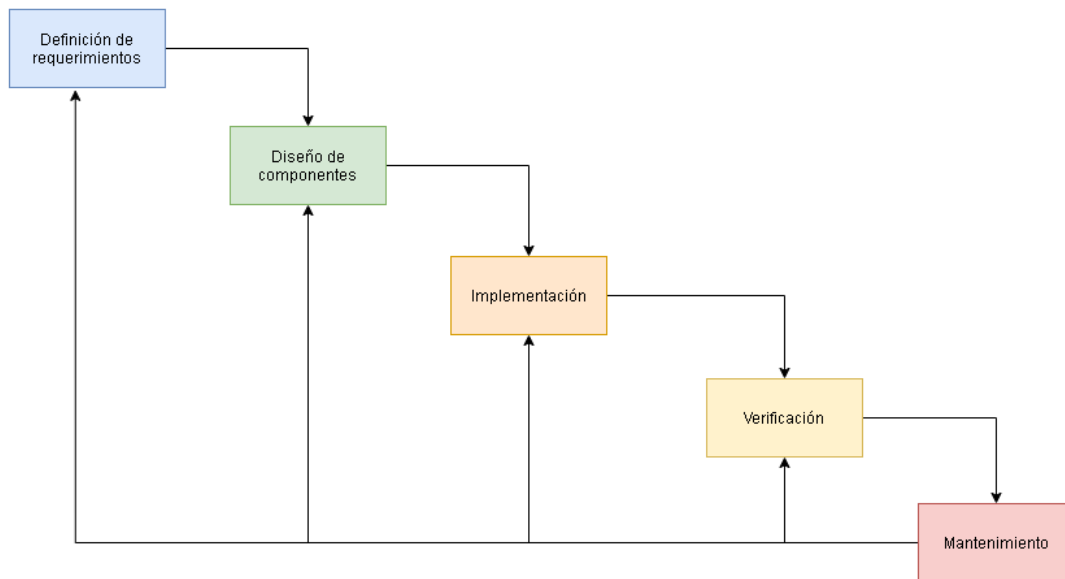



Ilustración 2 estructura de la metodología en cascada

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTA		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		

PROCEDIMIENTO O DESARROLLO INGENIERÍL


A continuación se demostrarán los productos y evidencia de actividades desempeñadas en cada una de las fases especificadas por la metodología que se va a emplear para el desarrollo de este proyecto.

Definición de requerimientos


A continuación, se definen los requerimientos funcionales y no funcionales del sistema, contemplando las necesidades expuestas por el cliente.

REQUERIMIENTOS FUNCIONALES BANCAPP		
Cod	Nombre	Descripción
RW001	Ingreso a la aplicación móvil	La aplicación debe permitirles a los usuarios ingresar a ella mediante las credenciales correspondientes, y en base a ellas determinar si ingresan como clientes o como administrador.
RW002	Transferencia entre cuentas	La aplicación debe permitirles a los clientes del banco realizar transferencias entre cuentas de los otros clientes del banco y notificar si se ha presentado algún inconveniente en el proceso.
RW003	Ver historial de transacciones	La aplicación debe permitirles a los clientes del banco revisar los movimientos y transacciones (consignación, retiro, transferencia) realizados.
RW004	Cambio de datos personales y credenciales de usuario	La aplicación debe permitirles a los usuarios cambiar cierta información personal y sus credenciales de acceso, esto último como medida de seguridad.

Ilustración 3 tabla de requerimientos funcionales

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTA		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		

REQUERIMIENTOS NO FUNCIONALES BANCAPP		
Cod	Nombre	Descripción
RW00X	Privacidad de datos empresariales	Este requerimiento consta de asegurar el acceso a la información asociada a cada uno de los componentes, es decir, permitir a los clientes de la entidad bancaria acceder de forma segura a la información de sus transacciones y la ejecución de estas.
RW007	Almacenamiento en base de datos	Se requiere que la aplicación móvil posea una base de datos la cual asegure el correcto indexado de los datos, los cuales serán utilizados por ella.
RW008	Acceso remoto a la información	Se requiere que se tenga acceso a las funcionalidades de forma remota, siempre y cuando, los usuarios tengan acceso a Internet.
RW09	Interfaz de usuario simple y fluida	Debido a que se requiere una gestión ágil para la información, no se requiere dinamismo en la presentación de la información para cada uno de los usuarios, ya que lo importante aquí es tener acceso rápido a las funcionalidades que requiera cada usuario.
RW010	Seguridad de la aplicación móvil	Debido al manejo de información personal, como lo son los datos de cada uno de los usuarios y de los empleados de las empresas, se requiere que la aplicación móvil cuente con seguridad frente a vulnerabilidades que afecten su funcionamiento
RW011	Seguridad de la base de datos	Se requiere algún sistema de seguridad para la base de datos y los datos que esta contiene debido a la importancia de la información que se manejara.

	UNIVERSIDAD DE SANBUENAVENTURA SEDE BOGOTÁ		
	PROGRAMA: INGENIERIA DE SISTEMAS	ASIGNATURA: SISTEMAS DISTRIBUIDOS	TIEMPO: 2:00 H
	PROYECTO DE AULA		

RW012	Limitación en infraestructura	Los componentes de base de datos y servidor deben encontrarse distribuidos, pero solo se cuenta con un equipo
RW013	Panel de ayuda de aplicación móvil	Para que los usuarios de la aplicación móvil puedan resolver dudas de forma inmediata frente a su uso, la aplicación requiere tener un espacio con ítems relacionados al manejo de cada una de las funcionalidades de la aplicación.

Ilustración 4 tabla de requerimientos no funcionales

BIBLIOGRAFÍA:

- Ferreira, H. (2015). *Requerimientos del Negocio-Modelos*.
[http://cs.uns.edu.ar/materias/rs/downloads/PRACTICA/Clase Practica3 Requerimientos del Negocio Modelos RdeS 2015.pdf](http://cs.uns.edu.ar/materias/rs/downloads/PRACTICA/Clase%20Practica3%20Requerimientos%20del%20Negocio%20Modelos%20RdeS%202015.pdf)
- IBM. (2021, January 25). *Servicios web*. Documentación de IBM.
<https://www.ibm.com/docs/es/was/9.0.5?topic=services-web>
- Microsoft. (2021, January 12). *What Is Cloud Computing? A Beginner's Guide | Microsoft Azure*. Microsoft Azure. <https://azure.microsoft.com/en-us/overview/what-is-cloud-computing/>
- Oracle Mexico. (2021, January 16). *¿Qué es una base de datos? | Oracle México. ¿Qué Es Una Base de Datos?* <https://www.oracle.com/mx/database/what-is-database/>
- Royce, W. W. (1970). *MANAGING THE DEVELOPMENT OF LARGE SOFTWARE SYSTEMS*. <http://www-scf.usc.edu/~csci201/lectures/Lecture11/royce1970.pdf>

AUTOR: Luis Felipe Velasco Tao – Juan David González Dimaté