

UNIVERSIDAD NACIONAL DE LANÚS



DEPARTAMENTO DE DESARROLLO PRODUCTIVO Y TECNOLÓGICO

LICENCIATURA EN SISTEMAS

Materia: Sistemas Distribuidos

Trabajo Práctico N°4

Web Services.

Docentes

Ing Diego Andrés Azcurra

Lic. Marcos Amaro

Alumnos:

Goday Carlos Nicolas

DNI 33.401.014

Pizarro Maximiliano

DNI 36.771.843

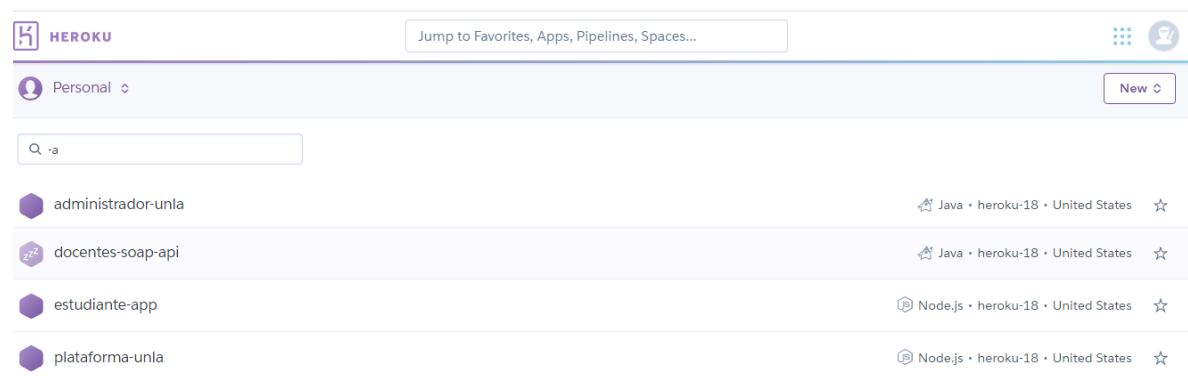
ÍNDICE



















1. [Plataforma UNLa](#)
2. [Arquitectura](#)
3. [Tecnologías](#)

 - 3.1. [Frontend](#)
 - 3.2. [Administrador Api Rest](#)
 - 3.3. [Estudiantes Api Rest](#)
 - 3.4. [Docentes Soap Web Service](#)
4. [Autenticación y Autorización](#)
5. [Base de Datos](#)
6. [Variables de Entorno](#)
7. [Repositorios](#)
8. [Link Pruebas](#)

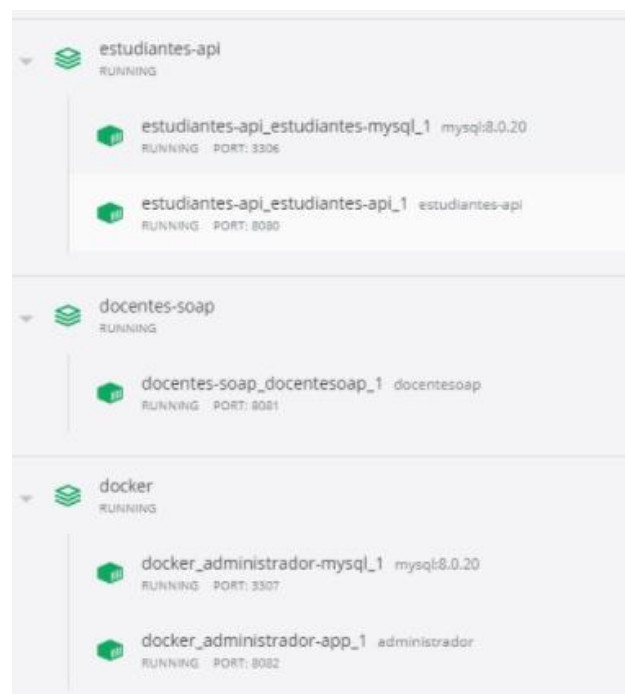
Plataforma UNLa.









Para el desarrollo de la actividad se produjeron contenedores **Docker** y servicios free tier de Heroku Content Platform para complementar el procesamiento distribuido en la nube durante el desarrollo de toda la solución que denominamos “**Plataforma UNLa.**”, generando de esta manera un espacio de trabajo colaborativo y reutilizable para la exposición final del Trabajo Práctico N°4 Web Services al equipo Docente según la planilla de los casos de prueba y el enunciado.



 HEROKU	Jump to Favorites, Apps, Pipelines, Spaces...	 
 Personal 		
<input type="text" value="Q -a"/>		
 administrador-unla		 Java • heroku-18 • United States 
 docentes-soap-api		 Java • heroku-18 • United States 
 estudiante-app		 Node.js • heroku-18 • United States 
 plataforma-unla		 Node.js • heroku-18 • United States 

Contenedores Java-Heroku18 y Node js-Heroku18 de la Plataforma UNLa.

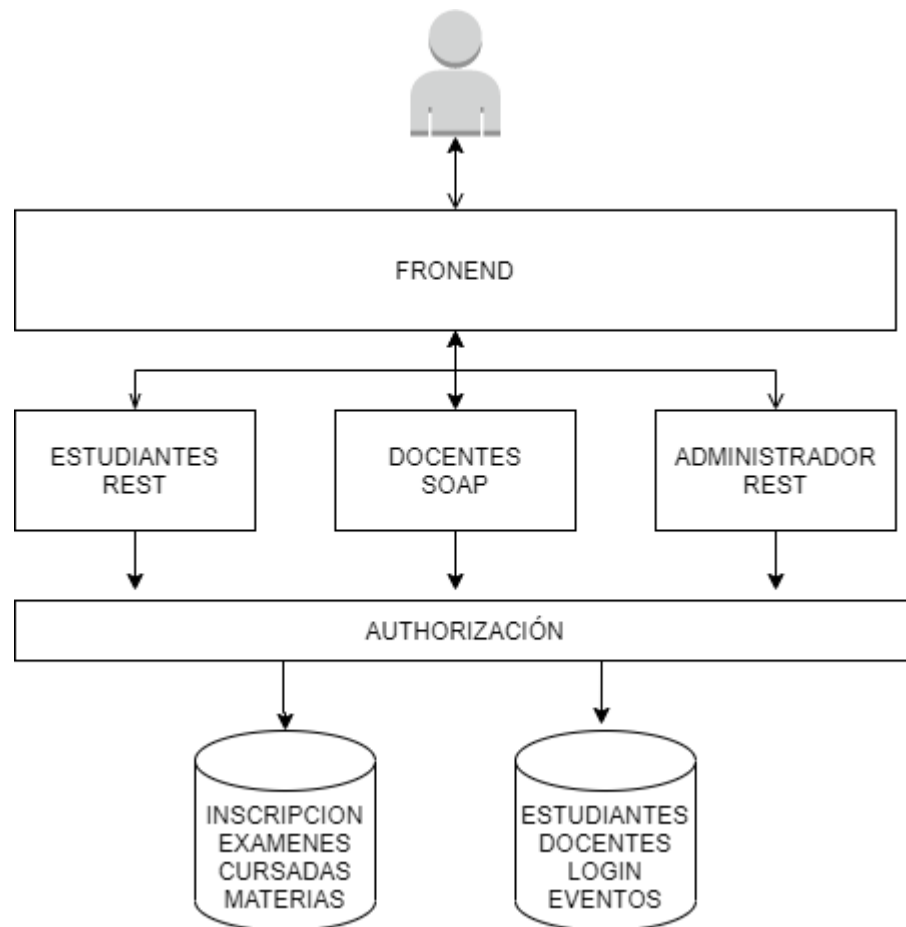


 estudiantes-api RUNNING	
 estudiantes-api_estudiantes-mysql_1	mysql:8.0.20 RUNNING PORT: 3306
 estudiantes-api_estudiantes-api_1	estudiantes-api RUNNING PORT: 8080
 docentes-soap RUNNING	
 docentes-soap_docentessoap_1	docentessoap RUNNING PORT: 8081
 docker RUNNING	
 docker_administrador-mysql_1	mysql:8.0.20 RUNNING PORT: 3307
 docker_administrador-app_1	administrador RUNNING PORT: 8082

Contenedores Docker Local de la Plataforma UNLa.

Arquitectura

Se tomó de referencias para la construcción y elección de los componentes de la arquitectura los principios de diseño del material bibliográfico aportado por el equipo docente en las clases “Microsoft Application Architecture Guide 2nd Edition” y “Aplicaciones, servicios y procesos distribuidos” de Patricia Bazán que fueron de gran utilidad para la división de responsabilidades, coherencia y cohesión, soporte y mantenimiento, resolución de problemas comunes del trabajo en Sistemas Distribuidos. El estilo de arquitectura definido para el desarrollo es “Basado en Componentes”.



Mapa de Arquitectura

Tecnologías

FrontEnd

Se implementó el framework de desarrollo **Angular 10** con servidor **Express** en lenguaje **Node JS v12.18.2** con un setup inicial basado en buenas prácticas del generador **JHipster** que implementa por defecto los componentes necesarios para el login y administración de cuentas. A partir de ese setup y en relación a los web service se respetó el principio de acceso a los componentes por medio de Interfaces siendo Angular un framework tipado y

orientado a las buenas prácticas del desarrollo de software. Podemos encontrar los conceptos del “Modelo de Objetos usados en Sistemas de Objetos Distribuidos” propuesto en el libro de Patricia Bazan ya que todos los mensajes se intercambian por medio de Publicación y Suscripción.

Administrador Api REST

Se implementó el framework **Spring Boot 2.2.7.RELEASE** con un setup inicial basado en buenas prácticas del generador JHipster. El servicio es de tipo Microservicio Gateway y su principal responsabilidad fueron con los componentes de login. A partir del setup se crearon los servicios de Profesores y Estudiantes, con algunos ajustes en el core de login también para comportamientos como el set de password por defecto.

Estudiantes Api REST

Se implementó un Microservicio Gateway en el lenguaje **Node JS v12.18.2** con servidor **Express** basado en el patrón de diseño MVC. Este componente es el núcleo de la capa de negocio de la Plataforma UNLa. en conjunto con Docentes Web Service SOAP que intercambia mensajes por medio del middleware node-soap para adaptar el patrón de publicar y suscribirse que implementa el frontend.

Docentes Web Service SOAP

Se implementó el framework **Spring 4.3.19.RELEASE** con el proveedor de JAVAX. Este componente está basado en el patrón de diseño MVC con DAO para el acceso a los repositorios. Cumple con los requerimientos del negocio de calificaciones de cursadas y materias.

Autorización y Autenticación

Para la autorización y autenticación se implementó el protocolo **JWT Stateless**. Los Json Web Tokens que se intercambian por Header en cada request y contiene la información cuentan con el usuario, el role de acceso y el tiempo de expiración para su uso. El proveedor de estos token es la api del administrador que cuenta con la información del login así como también la clave privada para la codificación y decodificación, esta clave es compartida al componente de estudiantes para la securización de sus recursos.

Bases de Datos

Se implementaron dos bases de datos **MySQL 8.0** con el fin de desacoplar la lógica de negocio de los usuarios y el sistema. Para poder consumir los componentes de base de datos se implementaron los middleware de **Liquibase** en la api del administrador y

Sequelize en la api de estudiantes simplificando los mapeos a las bases de datos por medio de **ORM**. En el caso del web service del docente se implementó un adaptador **JDBC**. También se implementaron algunas consultas en la api de estudiantes con el conector **ODBC**.

Variables de Entorno

Componentes	Variable	Valor
Frontend	NODE_ENV	prod
	SERVER_API_ESTUDIANTES	https://estudiante-app.herokuapp.com
	SERVER_API_URL	https://administrador-unla.herokuapp.com
Administrador Api REST	JDBC_DATABASE_URL	jdbc:mysql://us-cdbr-east-02.cleardb.com/heroku_fee96f339c93def?reconnect=true
	JDBC_PASSWORD	c759058a
	JDBC_USER	b1926aa07752b5
	SPRING_PROFILES_ACTIVE	prod,swagger
Estudiantes Api REST	NODE_ENV	prod
	NODE_ESTUDIANTE_DB	pod3i2eblsvv0rz1
	NODE_ESTUDIANTE_DBPORT	3306
	NODE_ESTUDIANTE_HOST	sm9j2j5q6c8bpgyq.cbetxkdyhwsb.us-east-1.rds.amazonaws.com
	NODE_ESTUDIANTE_PASSWORD	yznskdvv9lj45on
	NODE_ESTUDIANTE_USER	lqo0ahe9urejc5cw
	SERVER_WS_DOCENTES	https://docentes-soap-api.herokuapp.com/docentesSoap?wsdl
Docentes SOAP Web Service	HOST_SOAP	http://0.0.0.0
	JDBC_DATABASE_URL	jdbc:mysql://sm9j2j5q6c8bpgyq.cbetxkdyhwsb.us-east-

		l.rds.amazonaws.com/pod3i2eblsvv0rz1?reconnect=true
	JDBC_DB	pod3i2eblsvv0rz1
	JDBC_PASSWORD	yznskdvvuv9lj45on
	JDBC_USER	lqo0ahe9urejc5cw
	SPRING_PROFILES_ACTIVE	prod

Repositorios

Componente	Repositorio	URL
Frontend	https://github.com/maximilianoPizarro/siu-frontend	https://plataforma-unla.herokuapp.com
Administrador Api REST	https://github.com/maximilianoPizarro/administrador	https://administrador-unla.herokuapp.com/swagger-ui.html
Estudiantes Api REST	https://github.com/Nicolasgoda/estudiantes-api	https://estudiante-app.herokuapp.com/api-docs/
Docentes SOAP Web Service	https://github.com/Nicolasgoda/Docentes-Soap	https://docentes-soap-api.herokuapp.com/docentesSoap?wsdl

Tabla de Componentes

Link Pruebas

https://drive.google.com/drive/u/0/folders/1uDRBdwQEsaRvaRhBjwPJp_uWSubaXngR