

BICISHARE - Bicicletas compartidas

DOCUMENTO DE ARQUITECTURA

| Autor | Correo | Fecha | Versión |
|----------------------------|----------------------------|------------|---------|
| Efrain Aperador Mancipe | e.aperador@uniandes.edu.co | 17/04/2018 | 1.0 |

TABLA DE CONTENIDO

| | |
|---------------------------------------|----|
| TABLA DE CONTENIDO..... | 2 |
| 1. CONTEXTO DE LA APLICACIÓN | 3 |
| 1.1. Factores Externos | 3 |
| 1.2. Aplicación | 3 |
| 1.3. Capa de Servicios Web | 3 |
| 1.4. Servidor de Aplicaciones | 3 |
| 1.5. Servidor de Base de Datos | 3 |
| 2. PERSONAS | 3 |
| 3. ESCENARIOS FUNCIONALES | 4 |
| 1. ESCENARIOS DE CALIDAD | 4 |
| a. Atributos de Calidad | 4 |
| b. Escenarios de Calidad..... | 5 |
| 2. MATRIZ..... | 8 |
| 3. DISEÑO | 9 |
| a. MODELO DE DOMINIO..... | 9 |
| b. MODELO DE DESPLIEGUE | 11 |
| c. MODELO DE COMPONENTES..... | 11 |
| d. SET TECNOLÓGICO..... | 13 |
| Proyecto del lado del cliente:..... | 14 |
| Servicios REST:..... | 14 |
| Proyecto del lado del servidor: | 14 |

1. CONTEXTO DE LA APLICACIÓN

Bicishare es un programa que posee dos tipos de usuario, por un lado los administradores manejan un conjunto de herramientas que les permiten poner en alquiler o préstamo bicicletas o accesorios y tener un control total sobre el estado de las mismas; de otro lado los usuarios generales pueden realizar solicitudes de préstamo de las bicicletas y poseen un conjunto de otras utilidades que mejoran la experiencia de usuario dentro de la aplicación.

1.1. Factores Externos

En esta rama se encuentran representados los principales factores externos que van a tener relación con la aplicación estos son:

- **Usuarios:** serán las personas que presentan mayor cercanía y relación con la aplicación para llevar a cabo los diferentes procesos.
- **Sistemas Externos:** Tendremos comunicación a través de la aplicación a redes sociales para poder publicar o extraer contenido de las mismas.

1.2. Aplicación

Bicishare será una aplicación web la cual será consumida por los diferentes usuarios a través de internet.

1.3. Capa de Servicios Web

La capa de servicios servirá de intermediaria entre el cliente y la lógica del negocio; a través de esta lo que se logra es poder realizar el procesamiento de las solicitudes realizadas por un cliente, a través de un comportamiento de **petición y respuesta**.

1.4. Servidor de Aplicaciones

En el servidor de aplicaciones se encontrará principalmente la lógica de negocio de la aplicación, capa en la que se realizará el procesamiento y preparación de la respuesta a la solicitud realizada por el cliente.




1.5. Servidor de Base de Datos




Finalmente, en este servidor se va a alojar la base de datos en donde se van a realizar las transacciones relacionadas con cada módulo de la aplicación.

2. PERSONAS

Los principales usuarios de la aplicación serán:

| | |
|---------------------|---|
| ID | USU-001 |
| PERSONA | ADMINISTRADOR |
| DESCRIPCIÓN GENERAL | El administrador es el encargado de manejar el sistema de control de la aplicación, entre sus tareas se permite la creación, modificación o borrado de datos. |

| | | |
|--------------------|------------------------|---|
| ENTORNO DE TRABAJO | NAVEGADORES |    |
| | RESOLUCIÓN DE PANTALLA | |
| | SISTEMA OPERATIVO | |

| | | |
|---------------------|--|---|
| ID | USU-002 | |
| PERSONA | BICI-USUARIOS | |
| DESCRIPCIÓN GENERAL | Los biciusuarios son personas que necesitan realizar solicitudes de préstamo o alquiler de bicicletas, también pueden registrarse en la aplicación o modificar sus datos personales. | |
| ENTORNO DE TRABAJO | NAVEGADORES |    |
| | RESOLUCIÓN DE PANTALLA | |
| | SISTEMA OPERATIVO | |

3. ESCENARIOS FUNCIONALES

Bicishare va a ser desarrollado modularmente, por tal motivo los escenarios funcionales se especifican por módulo a trabajar.

....//TODO

1. ESCENARIOS DE CALIDAD

Para poder garantizar la calidad del sistema a desarrollar se deben tener en cuenta atributos de calidad de software necesarios para garantizar un buen producto y evitar errores en producción.

a. Atributos de Calidad

Los atributos de calidad son las cualidades o propiedades de calidad que el sistema debe satisfacer; los cuales se pueden medir de forma cuantitativa y/o cualitativa.

En la tabla mostrada a continuación se exponen los atributos de calidad que serán tenidos en cuenta para el desarrollo de la aplicación.

| ATRIBUTOS DE FUNCIONALIDAD | | | |
|----------------------------|--|-------------|--|
| CÓDIGO | AC-1 | CÓDIGO | AC-2 |
| ATRIBUTO | Interoperabilidad | ATRIBUTO | Seguridad |
| DESCRIPCIÓN | Capacidad del sistema para interactuar con uno o más sistemas. | DESCRIPCIÓN | El sistema debe proteger el acceso los datos, evitando intrusiones externas. |
| CÓDIGO | AC-3 | CÓDIGO | AC-4 |

| ATRIBUTO | Integridad | ATRIBUTO | Disponibilidad |
|-------------------------|---|-------------|---|
| DESCRIPCIÓN | El sistema debe de estar bien protegido, contra modificaciones de personas no privilegiadas, asegurando así la consistencia de datos. | DESCRIPCIÓN | El sistema debe de estar disponible las 24 horas, se deberá intentar minimizar todos los fallos en la medida de lo posible. |
| ATRIBUTOS DE EFICIENCIA | | | |
| CÓDIGO | AC-5 | CÓDIGO | AC-6 |
| ATRIBUTO | Tiempos de respuesta | ATRIBUTO | Escalabilidad |
| DESCRIPCIÓN | Capacidad del sistema para proporcionar tiempos de respuesta, procesos y potencia apropiados bajo condiciones determinadas. | DESCRIPCIÓN | El sistema debe estar preparado para el aumento de continuo de usuarios. |
| ATRIBUTOS DE USABILIDAD | | | |
| CÓDIGO | AC-7 | | |
| ATRIBUTO | Usabilidad | | |
| DESCRIPCIÓN | El sistema debe ser todo lo intuitivo posible, facilitando su uso a nuevos usuarios. | | |

b. Escenarios de Calidad

Para identificar los escenarios de calidad se seguirá la siguiente estructura:

1. **Fuente del estímulo:** quien o que genera el estímulo.
2. **Estímulo:** lo que se quiere llevar a cabo.
3. **Ambiente:** condiciones dentro de las cuales se presenta el estímulo.
4. **Artefacto:** parte del sistema que recibe el estímulo.
5. **Respuesta:** actividad que ocurre luego de la llegada del estímulo.
6. **Medida de la Respuesta:** criterio para testear el requerimiento.
7. **Atributo de calidad afectado:** Atributo de calidad relacionado con el escenario

Los escenarios de calidad a tener en cuenta para el desarrollo de la aplicación se encuentran descritos a continuación:

QAESC-01: El sistema debe permitir comunicación con otros sistemas de los cuales requiera información o los cuales requieran información de este.

| | |
|--------|----------------------------------|
| ID | QAESC-01 |
| Fuente | Comunicación con otros sistemas. |

| | | | |
|------------------------------|--|-------------------------------------|------|
| Estímulo | Necesidad de adquirir o proporcionar información a otros sistemas. | | |
| Ambiente | Ambiente de desarrollo, producción y pruebas. | | |
| Artefacto | Sistema, servidor y base de datos. | | |
| Respuesta | El sistema puede comunicarse satisfactoriamente con otros sistemas. | | |
| Medida de la Repuesta | Servicios del sistema para comunicarse con otros o para exponer información que otro sistema necesite de éste. | Atributo de calidad afectado | AC-1 |

QAESC-02: El sistema deberá impedir, en la medida de todo lo posible , los fallos de seguridad como intrusos que puedan acceder al sistema haciéndose pasar por usuarios con determinados permisos dentro de esta, o el acceso a los datos, pudiendo modificarlos, borrarlos o extraerlos.

| | | | |
|------------------------------|--|-------------------------------------|------|
| ID | QAESC-02 | | |
| Fuente | Acceso a la aplicación, información privada o intrusión. | | |
| Estímulo | Restricción de acceso a datos a personas que no se encuentren autorizadas en el sistema. | | |
| Ambiente | Ambiente de producción y pruebas con el sistema de autenticación y restricción a datos implementado. | | |
| Artefacto | Sistema, servidor y base de datos. | | |
| Respuesta | Acceso al sistema con credenciales seguras. | | |
| Medida de la Repuesta | Número de intrusiones y acceso a datos de forma ilícita. | Atributo de calidad afectado | AC-2 |

QAESC-03: Usuarios sin acceso, no privilegiados intentan modificar datos de la base de datos del sistema.

| | | | |
|------------------------------|---|-------------------------------------|------|
| ID | QAESC-03 | | |
| Fuente | Usuarios sin acceso - no privilegiados. | | |
| Estímulo | Intento de modificar los datos de la base de datos. | | |
| Ambiente | Producción. | | |
| Artefacto | Base de datos/Sistema | | |
| Respuesta | El sistema negará la operación que se desee realizar indicando el motivo. | | |
| Medida de la Repuesta | El usuario no debe haber podido realizar los cambios que quería efectuar. | Atributo de calidad afectado | AC-3 |

QAESC-04: El sistema deberá facilitar una alta disponibilidad, el portal será accesible el 90% del tiempo, en alguna de esas ocasiones la página no será visible y en otras el servicio será denegado al usuario, pidiéndole que trate de acceder más tarde; en especial cuando se esté realizando mantenimiento del sistema.

| | | | |
|---------------|---|--|--|
| ID | QAESC-04 | | |
| Fuente | Interacción de los usuarios con el sistema. | | |

| | | | |
|------------------------------|---|-------------------------------------|------|
| Estímulo | Interacción de los usuarios con el sistema. | | |
| Ambiente | Ambiente de producción. | | |
| Artefacto | Sistema. | | |
| Respuesta | Visualización e interacción con el portal o denegación de servicio. | | |
| Medida de la Repuesta | Denegar el servicio del Sistema el menor tiempo posible. | Atributo de calidad afectado | AC-4 |

QAESC-05: El sistema debe continuar su funcionamiento con normalidad en caso de presentar algún error al realizar una función.

| | | | |
|------------------------------|---|-------------------------------------|------|
| ID | QAESC-05 | | |
| Fuente | Funcionamiento del sistema ante la presencia de errores. | | |
| Estímulo | Presencia de un error en la ejecución de alguna función de la aplicación. | | |
| Ambiente | Aplicación en producción o pruebas finales. | | |
| Artefacto | Sistema | | |
| Respuesta | El sistema sigue en funcionamiento a pesar del problema presentado. | | |
| Medida de la Repuesta | MTBF = intervalo de tiempo promedio que transcurre hasta que se produce un error en un componente y es preciso repararlo. | Atributo de calidad afectado | AC-5 |

$$MTBF = \frac{(\text{tiempo total transcurrido} - \text{suma del tiempo de inactividad})}{\text{número de fallos}}$$

QAESC-06: El sistema debe ser diseñado de forma que aunque en un inicio el número de usuarios sea bajo, si en un periodo corto de tiempo aumentasen, la disponibilidad y rendimiento de este no se vea afectado.

| | | | |
|------------------------------|--|-------------------------------------|------|
| ID | QAESC-06 | | |
| Fuente | Incremento de usuarios y transacciones. | | |
| Estímulo | Gran número de peticiones de acceso al sistema o solicitud de información. | | |
| Ambiente | Ambiente de Producción. | | |
| Artefacto | Servidor del Sistema. | | |
| Respuesta | Se incrementa la capacidad del servidor y base de datos del sistema. | | |
| Medida de la Repuesta | Tiempo en el que el sistema da respuesta al acceso al sistema de los usuarios, número de peticiones denegadas. | Atributo de calidad afectado | AC-6 |

QAESC-07: El sistema no debe tardar más de 5 segundo en creación de información.

| | | | |
|-----------|-----------------|--|--|
| ID | QAESC-07 | | |
|-----------|-----------------|--|--|

| | | | |
|------------------------------|---|-------------------------------------|------|
| Fuente | Usuario. | | |
| Estímulo | Creación de nuevos datos en el sistema. | | |
| Ambiente | Ambiente de Producción y pruebas. | | |
| Artefacto | Sistema. | | |
| Respuesta | El sistema debe realizar inserción de información en el sistema con una duración no mayor a 5 segundos. | | |
| Medida de la Repuesta | Tiempo de respuesta en operaciones de inserción de datos no mayor a 5 segundos. | Atributo de calidad afectado | AC-7 |

QAESC-08: Los usuarios del sistema realizan cualquier operación (siempre que estén autorizados) en el sistema.

| | | | |
|------------------------------|---|-------------------------------------|------|
| ID | QAESC-08 | | |
| Fuente | Usuario. | | |
| Estímulo | Realizar una operación, evento y/o acción en el sistema. | | |
| Ambiente | Ambiente de Producción. | | |
| Artefacto | Sistema. | | |
| Respuesta | El sistema va a realizar la operación de una manera rápida y sin fallos. | | |
| Medida de la Repuesta | El usuario no debe abandonar el sistema debido a un largo tiempo de espera. | Atributo de calidad afectado | AC-5 |

QAESC-09: El sistema no debe tardar más de 5 segundo en operaciones de consulta de información.

| | | | |
|------------------------------|---|-------------------------------------|------|
| ID | QAESC-09 | | |
| Fuente | Usuario. | | |
| Estímulo | Consulta de datos en el sistema. | | |
| Ambiente | Ambiente de Producción y pruebas. | | |
| Artefacto | Sistema. | | |
| Respuesta | El sistema debe mostrar información contenida en el sistema con una duración no mayor a 5 segundos. | | |
| Medida de la Repuesta | Tiempo de respuesta en operaciones de consulta de datos no mayor a 5 segundos. | Atributo de calidad afectado | AC-5 |

2. MATRIZ

Dentro del proceso de diseño de la arquitectura se realiza una matriz que permita identificar el impacto de los escenarios funcionales y de calidad con respecto a los atributos de calidad planteados en la arquitectura. Con este análisis lo que se desea es identificar los principales atributos de calidad que afectan el buen funcionamiento de la aplicación.

En cuanto a los escenarios funcionales la mayoría se encuentran relacionados con transacciones que tienen gran interacción con los datos de la aplicación y con la visualización de estos, por lo que en temas de atributos de calidad priman los temas de **seguridad, integridad, disponibilidad y tiempos de respuesta** como principales a tener en cuenta en el funcionamiento del sistema.

En la tabla mostrada a continuación se representa la relación entre los escenarios de calidad y los atributos de calidad relacionados.

| A.C E.C | AC-1 | AC-2 | AC-3 | AC-4 | AC-5 | AC-6 | AC-7 |
|------------|------|------|------|------|------|------|------|
| QAESC-01 | | | | | | | |
| QAESC-02 | | | | | | | |
| QAESC-03 | | | | | | | |
| QAESC-04 | | | | | | | |
| QAESC-05 | | | | | | | |
| QAESC-06 | | | | | | | |
| QAESC-07 | | | | | | | |
| QAESC-08 | | | | | | | |
| QAESC-09 | | | | | | | |

Partiendo de lo mostrado anteriormente es de resaltar que todos los atributos de calidad especificados en el presente documento generan una gran importancia al sistema de información, pero con este apartado el ideal es poder generar la priorización de estos para tenerlos claramente identificados en para la realización del diseño, por ende se seleccionan:

1. Escalabilidad
2. Seguridad
3. Disponibilidad
4. Latencia (tiempos de respuesta)

3. DISEÑO

En este apartado se encuentra expuesto el diseño que tendrá el sistema de información teniendo en cuenta los atributos de calidad identificados en el numeral anterior.

a. MODELO DE DOMINIO

El modelo de dominio del sistema va a ser representado a través del modelo entidad-relación referente al primer módulo que se va a desarrollar el cual es el de **monitorias**; conforme los módulos van surgiendo el modelo de dominio irá creciendo como sea necesario.

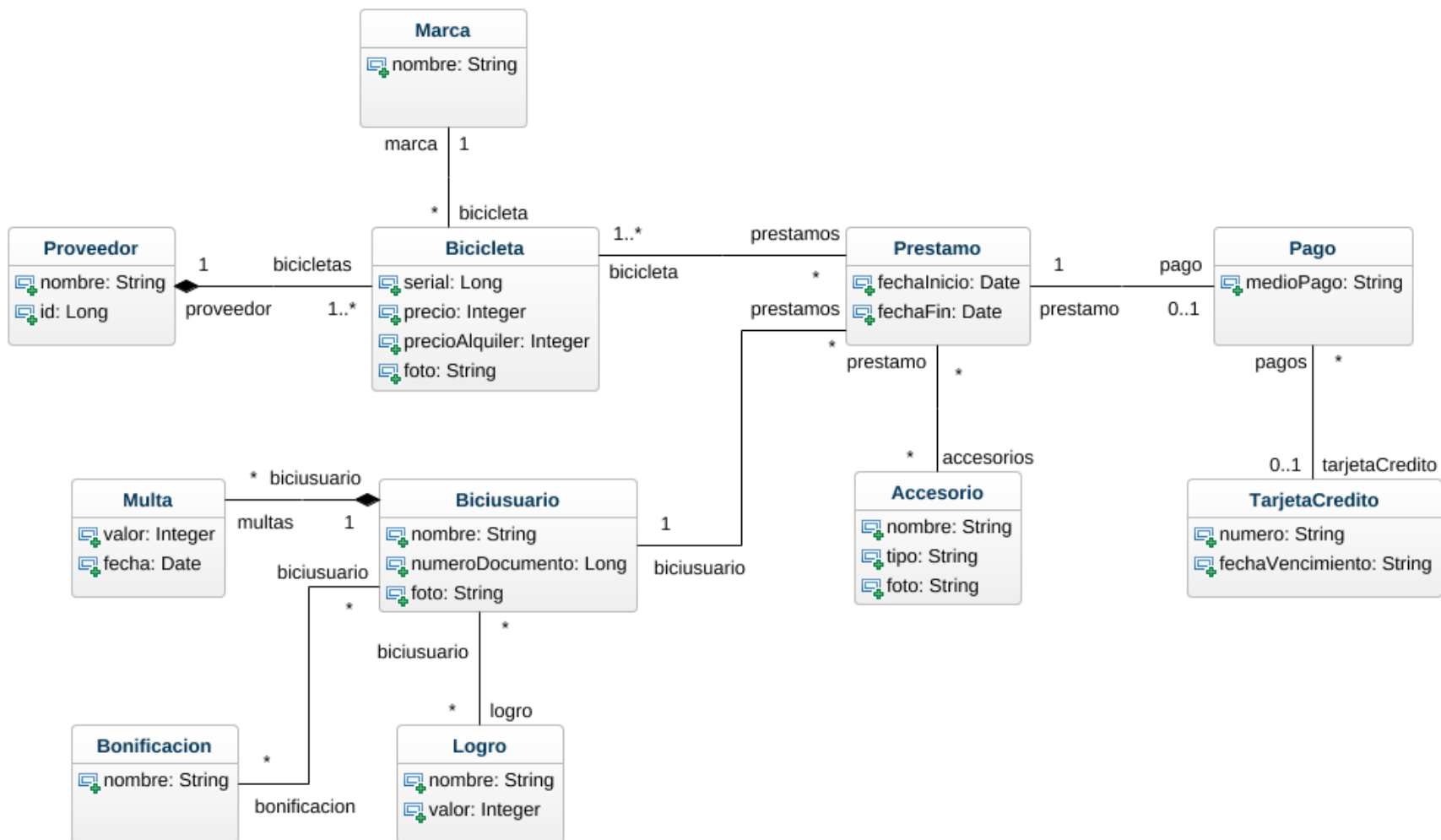


Figura 1 Modelo de dominio - Diagrama de clases

b. MODELO DE DESPLIEGUE

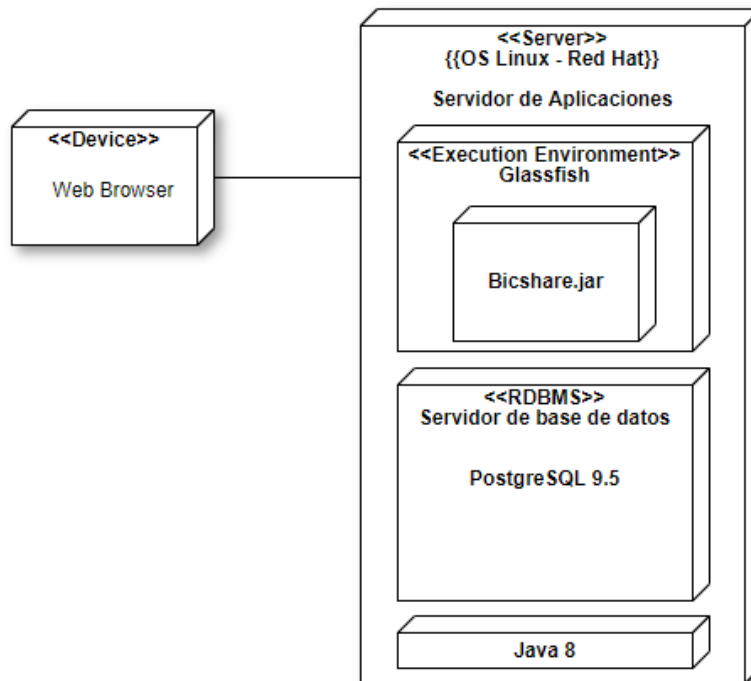


Figura 2 Modelo de despliegue - Glassfish

c. MODELO DE COMPONENTES

Mediante este apartado se muestra una estructura más específica sobre la forma como se va a realizar el desarrollo de la aplicación, teniendo en cuenta que se realiza contemplando dos frentes: la aplicación web y su desarrollo para aplicación móvil.

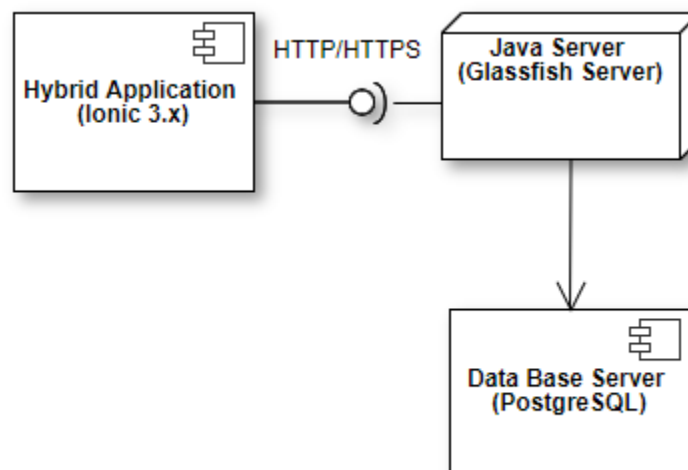


Figura 3 Modelo de Componentes simple

El framework seleccionado para el desarrollo de la aplicación web es Angular en su versión 4.2.4. junto con Ionic 3.x; Angular permite realizar un desarrollo más limpio al tener una estructura similar a la POO (Programación Orientada a Objetos); lo que facilita la mantenibilidad del código, la rápida detección de errores y la escalabilidad de la aplicación; de otro lado, el dejar carga al navegador puede mejorar notablemente el rendimiento del servidor al reducir la cantidad de llamados y procesos que se ejecutan desde el mismo dejando al cliente ejecutar gran parte de las funciones y el procesamiento de las pantallas.

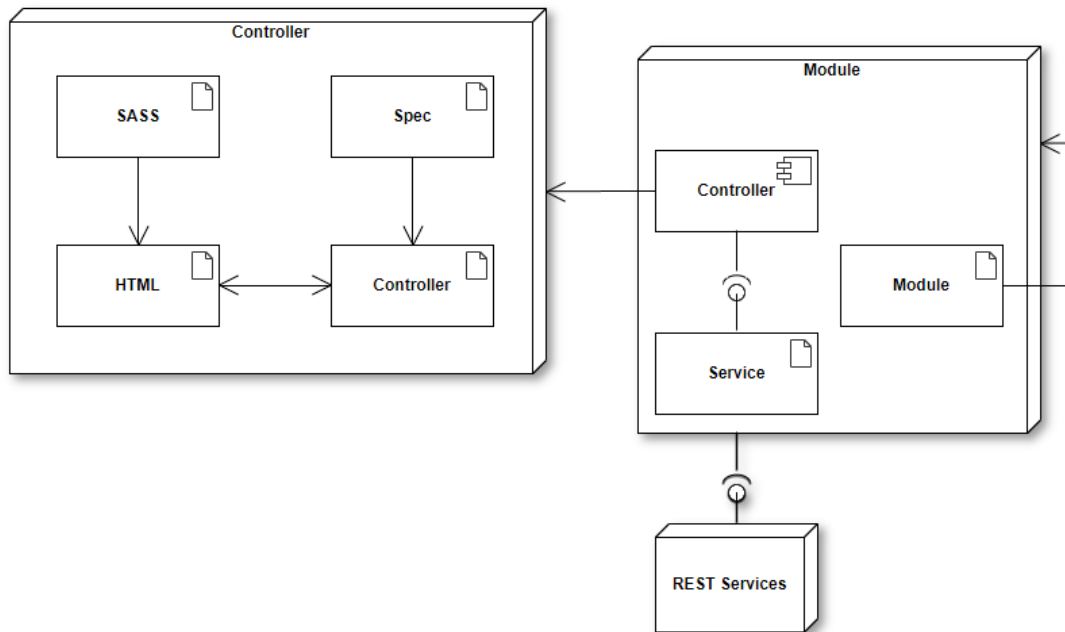


Figura 4 Modelo de Componentes Front End desplegado

El diseño elegido para el desarrollo de la aplicación consta de 3 elementos base:

- a. **Módulo (Module):** El módulo es el contenedor de una sección de la aplicación que ejecuta un proceso específico, puede hacer referencia a una o varias pantallas. Un módulo a su vez puede contener otros submódulos que hacen parte del proceso general pero necesitan un grado de independencia para mejorar la mantenibilidad de la aplicación. El módulo debe contener funciones de enrutamiento y una sección de importación/exportación de elementos necesarios para que el módulo funcione.
- b. **Controlador (Controller):** Se denomina controlador al conjunto de elementos necesarios para que una vista funcione. El controlador está compuesto de 4 elementos:
 - i. **SASS:** Contiene el código de los estilos de la vista en formato SASS (Syntactically Awesome Style Sheets). Para más información diríjase a la página <https://sass-lang.com/>
 - ii. **HTML:** Contiene todas las etiquetas necesarias que componen la estructura de la página web
 - iii. **SPEC:** Documento de especificación, contiene las pruebas unitarias de la funcionalidad de la vista.
 - iv. **CONTROLLER:** Contiene el código necesario para el correcto funcionamiento de la página, el lenguaje usado es TypeScript.

- c. **Service:** El componente de servicios se encarga de la comunicación con el servidor, en el se encuentran todos los archivos necesarios para apuntar a los end points (direcciones de los servicios) expuestos por el servidor

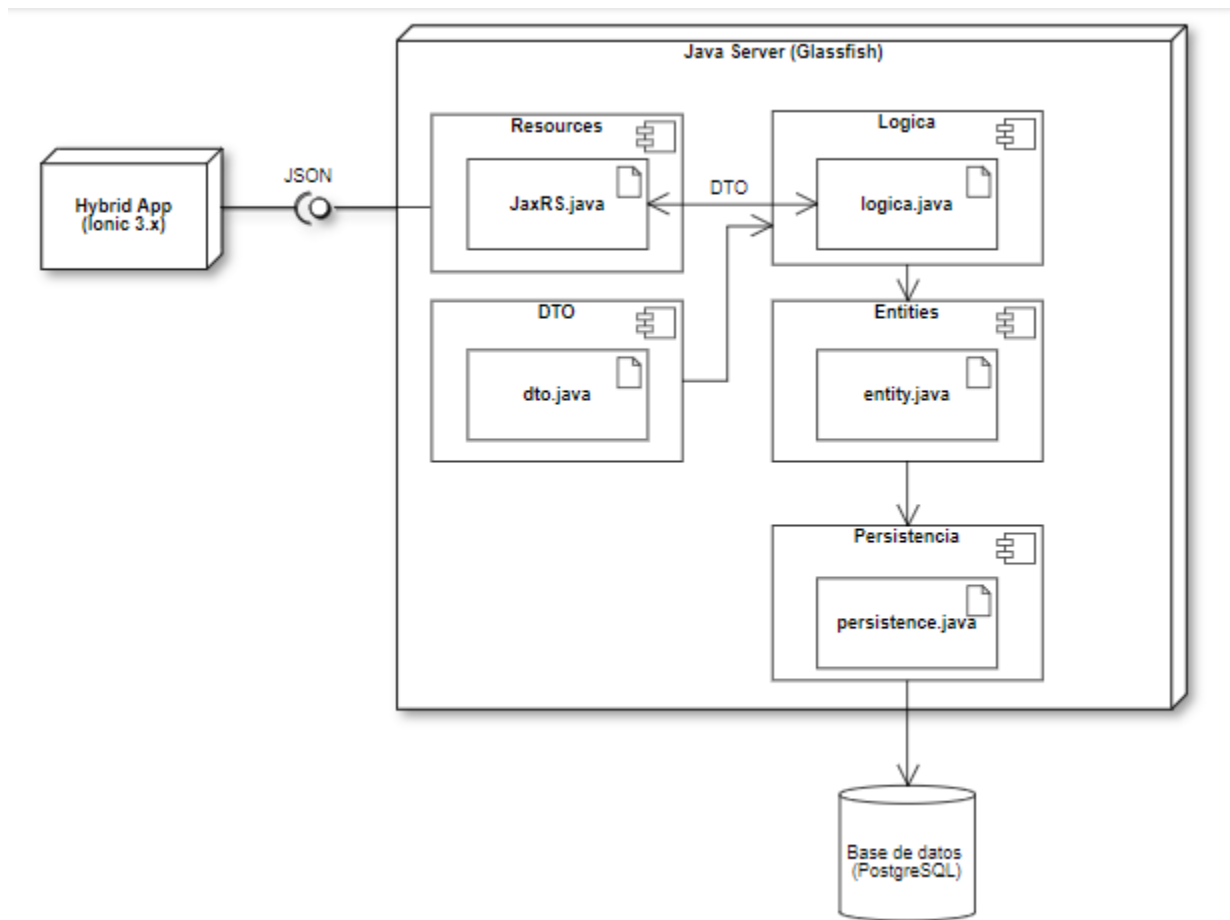


Figura 5 Modelo de Componentes Back End desplegado

Para el desarrollo del servidor se eligió la tecnología Java en su versión 8 debido a su robustez, mantenibilidad, estandarización y peso en el mercado de lenguajes de desarrollo, adicionalmente el servidor de despliegue es Glassfish 5. El diseño de la arquitectura está enfocado en un Cliente - Servidor de 3 capas, la primera capa expone servicios REST mediante JaxRS, la segunda realiza toda la lógica necesaria de la aplicación y la última realiza la conexión a la base de datos.

Seleccionamos una base de datos PostgreSQL debido a que es un BDMS de código abierto, libre y con una gran comunidad de soporte. Adicionalmente es adecuada para la cantidad de datos calculados que debemos almacenar.

d. SET TECNOLÓGICO

La aplicación tendrá dos proyectos independientes una del lado del cliente y otro del lado del servidor, las cuales estarán comunicadas mediante solicitudes HTTP usando Ajax

apuntando al servidor que expuso previamente servicios RESTful; finalmente el servidor se conecta a través del puerto 5432 a la base de datos.

Para cada una de estas secciones la tecnología a ser trabajada será:

Proyecto del lado del cliente:

En lado del cliente se va a hacer uso de tecnología **Angular 4.2.4** y **Ionic 3.1**. Con esta tecnología crear aplicaciones adaptativas con una menor cantidad de código y tiempo invertido.

Además se hará uso de **SASS** el cual es un **precompilador de CSS** el cual permite incrementar la productividad en la escritura de este lenguaje, gracias a que, automatiza tareas repetitivas y admite realizar manejo de constantes.

Finalmente, para la maquetación de la interfaz gráfica se hará uso de HTML en su versión 5.1.

Servicios REST:

La capa de **servicios REST** será implementada en Java mediante el uso del API **JaxRS**, la cual provee componentes específicos para la construcción de los servicios.

Proyecto del lado del servidor:

El servidor será desarrollado en Java, haciendo uso de:

- **JPA:** API de persistencia utilizada para la gestión y manipulación de la base de datos.
- Una estructura de **DTO (Data Transfer Object)** que permiten el mapeo objeto relacional de cada información que está siendo manipulada en el proceso.
- Y se cuenta con manejadores de la lógica de negocio, estructuras que van a ser llamadas principalmente desde los servicios dependiendo las solicitudes realizadas por lo clientes, y/o respuestas emitidas por el modelo.

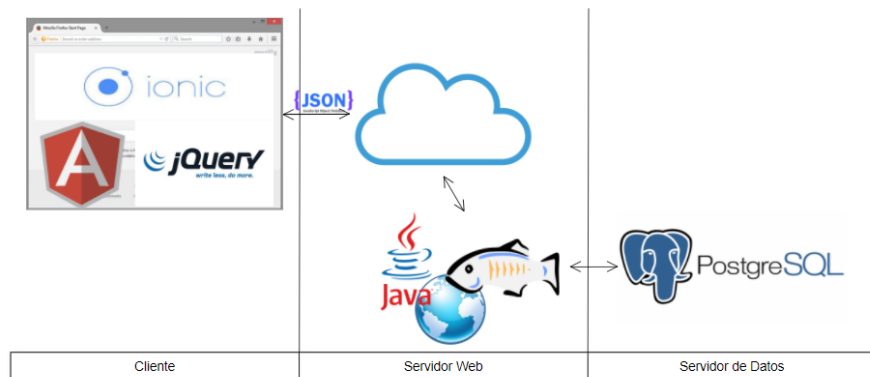


Figura 6 Modelo de Tecnología