



UNIVERSIDAD DE  
SAN BUENAVENTURA  
CALI

## TRABAJO FINAL ARQUITECTURA DE SOFTWARE

CRISTIAN ALFREDO MUÑOZ ALVAREZ  
JHONY SARRIA REVELO

DOCENTE:  
FERNANDO BARRAZA ALVARADO

UNIVERSIDAD DE SAN BUENAVENTURA CALI  
FACULTAD DE INGENIERIA  
ARQUITECTURA DE SOFTWARE

2018 – 1

## Tabla de contenido

DESCRIPCION DE TALLER .....	3
DESCRIPCIÓN DE LA SOLUCIÓN .....	3
ARQUITECTURA DEL SISTEMA .....	5
VISTA LOGICA .....	5
VISTA DE DESARROLLO.....	6
VISTA FISICA.....	7
VISTA ESCENARIOS.....	8
VISTA DE PROCESO .....	9

## DESCRIPCION DE TALLER

Para llegar hasta este punto del trabajo final, durante el semestre se realizaron distintos talleres, con el objetivo de entender como se diseñaba la arquitectura y los diagramas y diseños a utilizar. Para cumplir con este objetivo se llevo el sistema desde una complejidad baja a una complejidad superior. Los talleres realizados durante el semestre fueron:

- HelloName web: por medio de java se creaba una pantalla que contenía una caja de texto, dentro de esta se escribía el nombre y el sistema saludaba adjuntando el texto escrito en la caja.
- HelloName idiomas: en este segundo taller se tomó como base el primer taller, y el sistema ahora no solo saludaba, si no que tenia en cuentas el idioma del nombre y apartir de esto, saludaba en el idioma correspondiente.
- HelloName idiomas SPRING MVC: en el tercer taller se toma como base el segundo, y se realiza sobre el framework SPRING, con un patrón MVC, haciendo la identificación de cada una de las capas, y en cuál de estas interviene el framwork.
- HelloName SPRING MVC persistencia: en este cuarto taller, se toma como base el tercero y el principal cambio que se realiza es que los datos ya no están quemados en el sistema, se incluye una persistencia en este caso realizado en PostgreSQL, donde estarán los datos de idioma, nombre y saludo.
- HelloName final: En la última entrega del sistema, se le agrega un web service, donde enviando el nombre el retorna el saldo correspondiente, en caso de no tener algún saludo en la persistencia, se responderá un saludo por defecto.

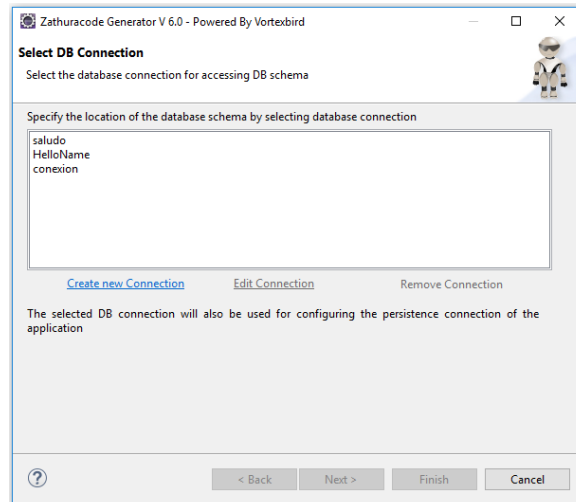
## DESCRIPCIÓN DE LA SOLUCIÓN

Para llevar a cabo el desarrollo del proyecto decidimos utilizar las siguientes tecnologías, frameworks y plug-in:

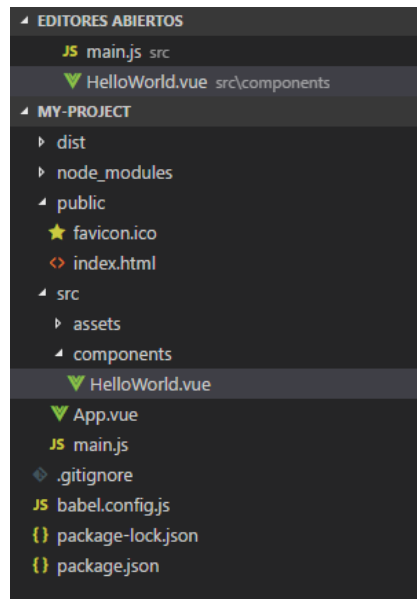
- ZathuraCode: Plug-in de eclipse para la creación de un proyecto web basado en una conexión a una base de datos. Se genero modelo, control, conexión a persistencia JPA, DAO, web services y proyecto web en primefaces.



- VUE : es un framework de JavaScript nuevo, si lo comparamos con otros frameworks como Backbone o Ember. Sin embargo, su facilidad de aprendizaje y uso con respecto a otros frameworks y libraries como ReactJS, su rendimiento comparado con AngularJS y la facilidad para usarlo y adaptarlo a proyectos tanto grandes como pequeños.



## Plug-in zathuraCode



## Proyecto VUE.js

## ARQUITECTURA DEL SISTEMA

Para representar la arquitectura del sistema decidimos emplear el modelo 4+1. A continuación se hará una descripción de cada una de las vistas del modelo:

### VISTA LOGICA

Lo que se busca por medio de esta vista es representar la estructura y funcionalidad general del sistema, para este caso se decidió representarlo por medio de una diagrama de clases UML, y como anexo se adjunta el modelo entidad relación correspondiente al sistema.

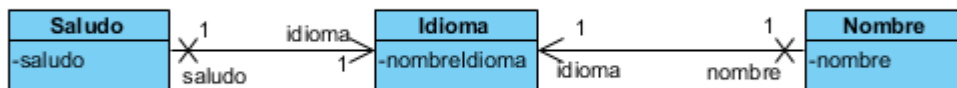
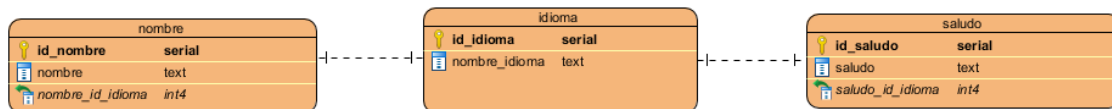


Diagrama de clases



Anexo. Modelo entidad relación

En este caso definimos tres entidades que son idioma, nombre y saludo, las cuales tiene atributos específicos que son:

- Idioma: identificador del idioma y nombre del idioma.
- Nombre: tiene el identificador del nombre, el nombre (dato que será comparado con el introducido por el usuario) y nombre\_id\_idioma (llave foránea).
- Saludo: identificador del saludo, el saludo (se consultara para que el sistema salude de acuerdo al idioma del nombre introducido), saludo\_id\_idioma (llave foranea).

## VISTA DE DESARROLLO

En la vista de desarrollo también conocida como la vista de implementación, se busca con esta vista ilustrar la perspectiva del programador, enfocándose en la administración de los artefactos de software. Para este caso se decidió hacer un diagrama de componentes, el siguiente diagrama es el correspondiente al sistema desarrollado.

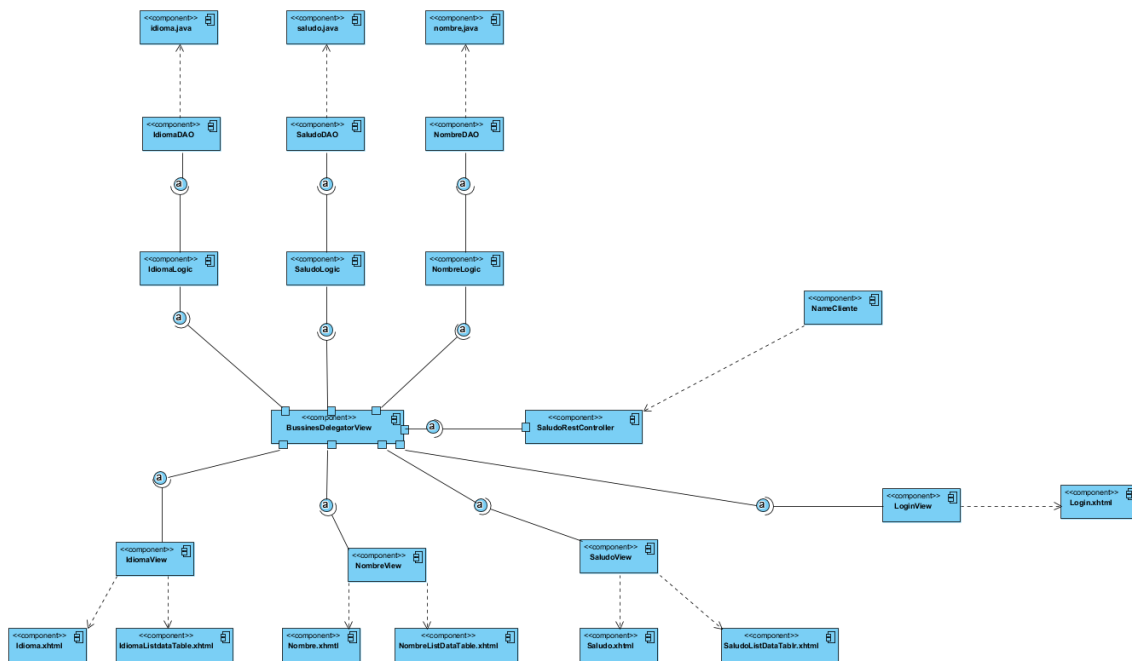
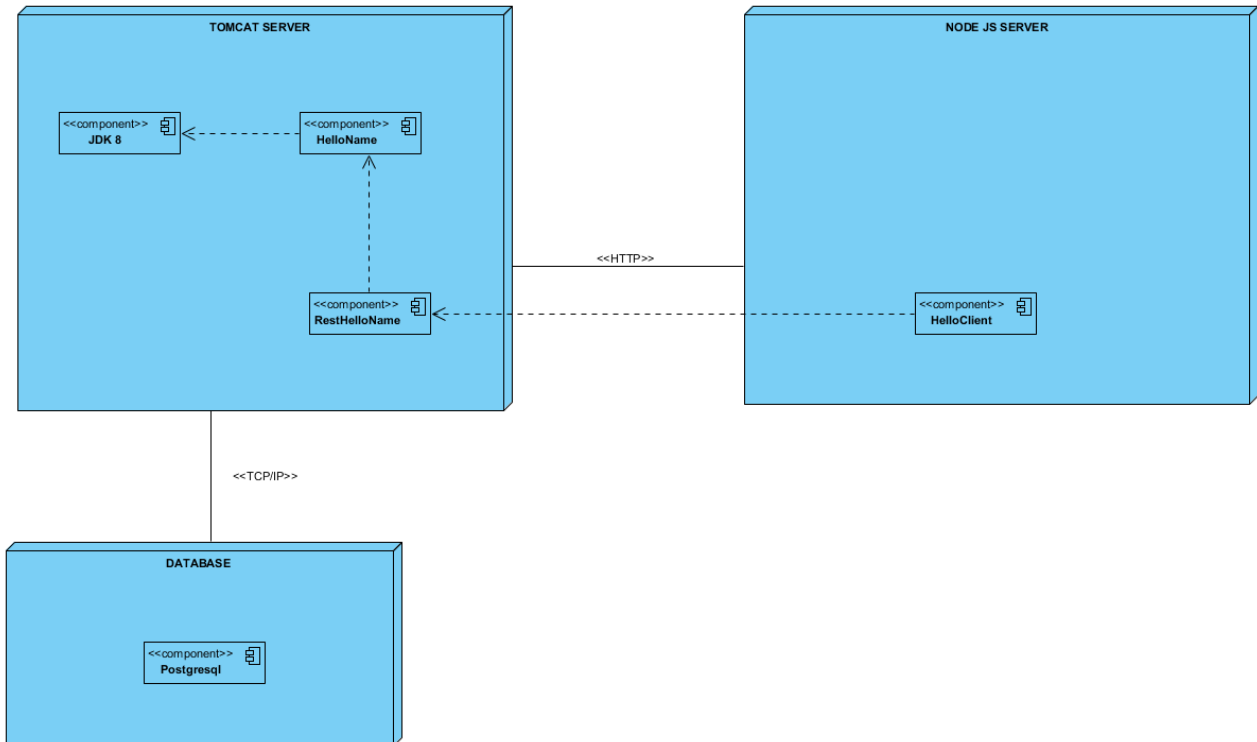


Diagrama de componentes

En el diagrama se puede ver las capas generadas por el plug-in zathura code, inicialmente tenemos el modelo, que es la representación de la persistencia de datos dentro del sistema. Esta también la capa de DAOS, en donde inicialmente se generan las CRUD de cada entidad de la base de datos, posteriormente se le agregan las consultas dependiendo de la lógica del negocio, por otro lado tenemos la capa lógica en donde se hace la lógica de negocio del sistema y validaciones, por otro lado tenemos el delegado de negocio que es la conexión de la vista con el backend del sistema y por último tenemos la capa web, que son los bean y xhtml correspondiente a cada vista.

## VISTA FISICA

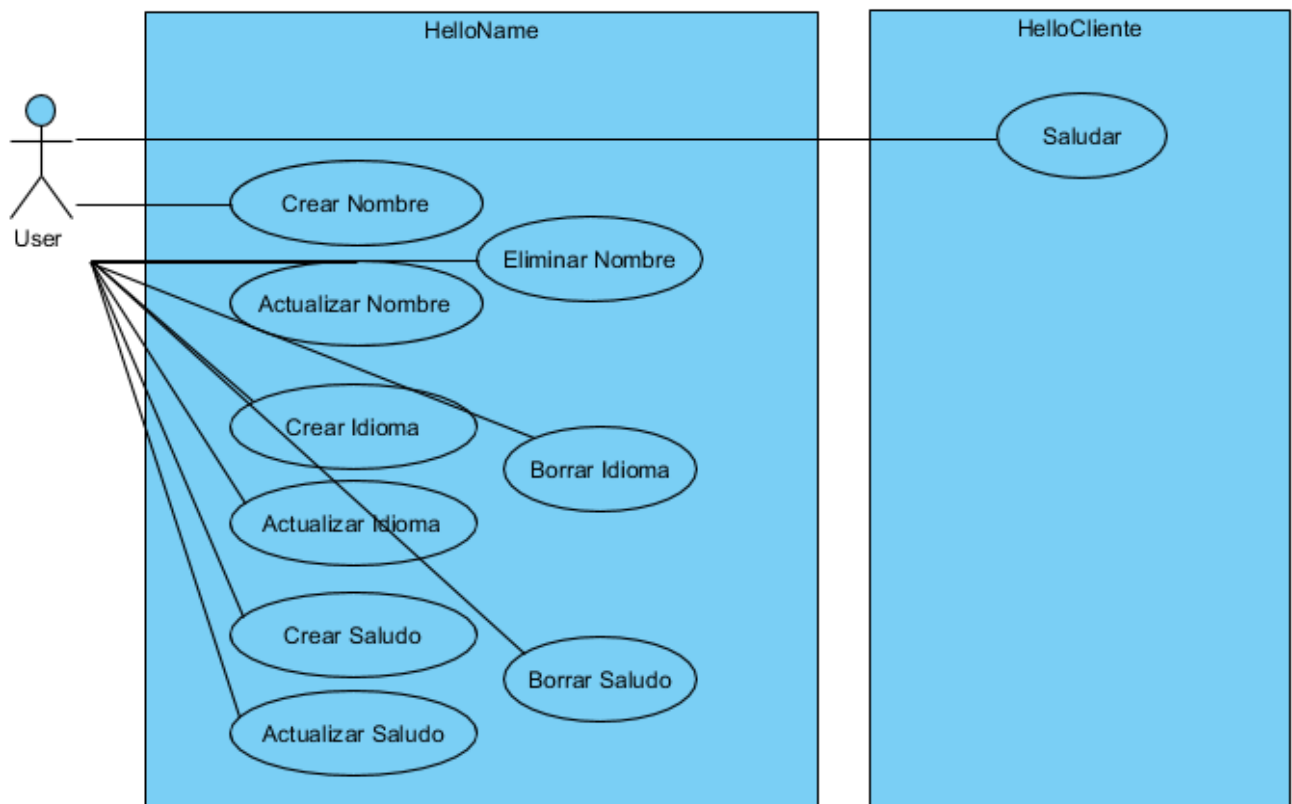
En la vista física lo que se busca es describir el sistema desde el punto de vista del ingeniero en sistemas, esta relacionada con la topología de componentes de software en la capa física, así como las conexiones físicas entre estos componentes.



En este caso se tienen dos servidores principales, el tomcat donde estaría alojada a aplicación java y otro de node para la aplicación VUE.js

## VISTA ESCENARIOS

En la vista de escenarios se da una descripción de la arquitectura se ilustra utilizando un conjunto de casos de uso, o escenarios lo que genera una quinta vista. Estos escenarios describen secuencias de interacciones entre objetos, y entre procesos. Se utilizan para identificar y validar el diseño de arquitectura. También sirven como punto de partida para pruebas de un prototipo de arquitectura.

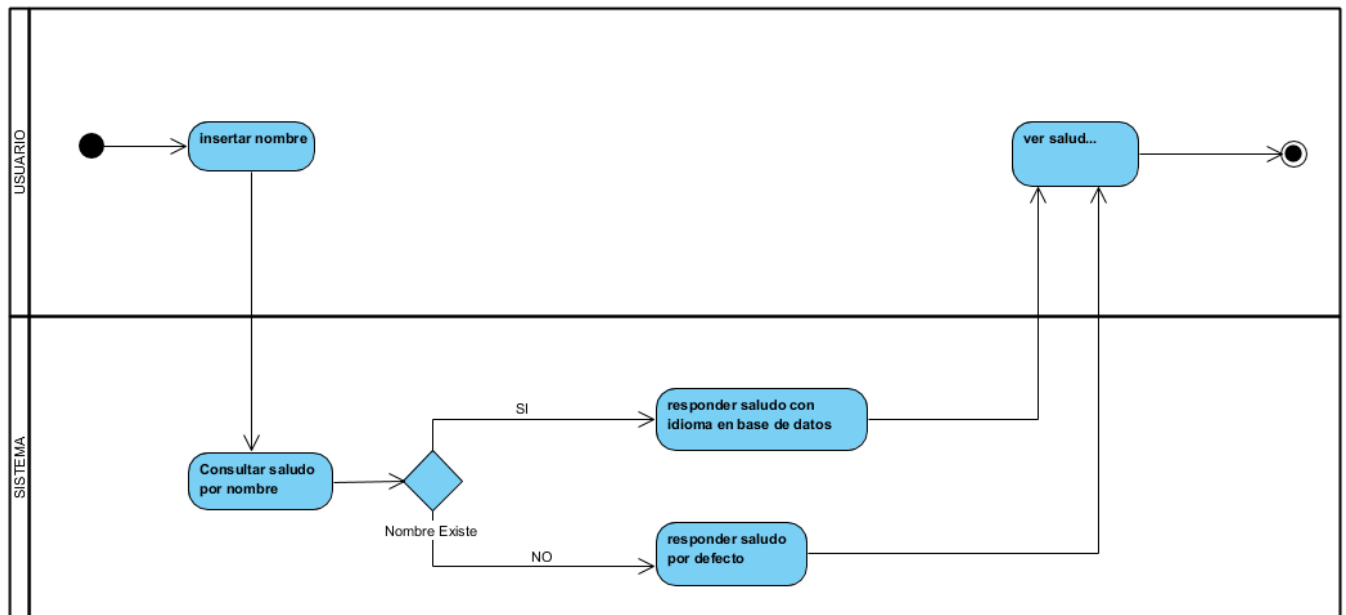


En este caso definimos un usuario que puede hacer uso del sistema por medio del aplicativo web o utilizando el web service por medio del aplicativo desarrollado en VUE.js



## VISTA DE PROCESO

En la vista de proceso trata los aspectos dinámicos del sistema, explica los procesos de sistema y cómo se comunican. se enfoca en el comportamiento del sistema en tiempo de ejecución. La vista considera aspectos de concurrencia, distribución, rendimiento, escalabilidad, etc.



En este caso el diagrama se actividades describe el proceso básico para hacer uno de la herramienta HelloName