

PROJECT DOCUMENTATION

Title: - Shopping App with Virtual Personal Assistant (VPA) .

Technology Used: -

- i) Spring Boot and JAVA (Back-End)
- ii) Angular (Front-End)
- iii) Heroku/Firebase → Deployment of the backend in cloud server (Acting as a server part).
- iv) Netlify → Deployment of the frontend as a cloud server.
- v) Python → Development of the Virtual Personal Assistant.
- vi) Jython → Integrating part between Python and Spring Boot with JAVA and successfully binding it.

Objective: -

A full stack application of Shopping App with a personalized Virtual Personal Assistant. The objective is that we will give a shopping app in virtual platform by covering the features as much as possible like the other famous shopping app.

In addition to this, we would like to add one more feature i.e., Virtual Personal Assistant which is not actually present in any shopping website. So, a website is memory efficient as it does not consume the hard disk or any kind of permanent memory.

We would like to build a VPA, a voice automated website, so that the client can get a feel that they are guided by an assistant in any other book shops that we visit physically.

We would also like to increase the scalability, flexibility as well as the performance of the website.

Synopsis: -

Spring Boot with JAVA is used to create the backend part of the project. The main purpose is to create REST API endpoints. Now why Spring Boot? Spring Boot has an embedded server like Tomcat/Jetty as well as embedded database 'H2'. Now, with Spring we can change our database as H2 is a volatile database and is only for testing purpose. The benefit of the spring boot is that it can store the data in JSON format as well as it supports Junit/ Selenium Testing which is very essential in any kind of project. Not only this but also, we can create database or rather manipulate database by writing few lines of code. So, spring boot gives a complete control over the backend and any changes can be easily altered by changing few lines of codes and the annotations features saves us from boiler's plate code approach.

We use Angular for the presentation part in the frontend, but we can apply Object Based Programming and can apply JWT, OAuth Guard and HTTP Interceptor simply by coding. We can do the navigation part with redirect method in and can do the approach of Single Paging Application (SPA) and its lightweight as rendering is very easy in implementation too.

Now we can use either MongoDB or MySQL by replacing volatile H2 database and can be used as database in server to act as a permanent database and will remain even after reloading the database.

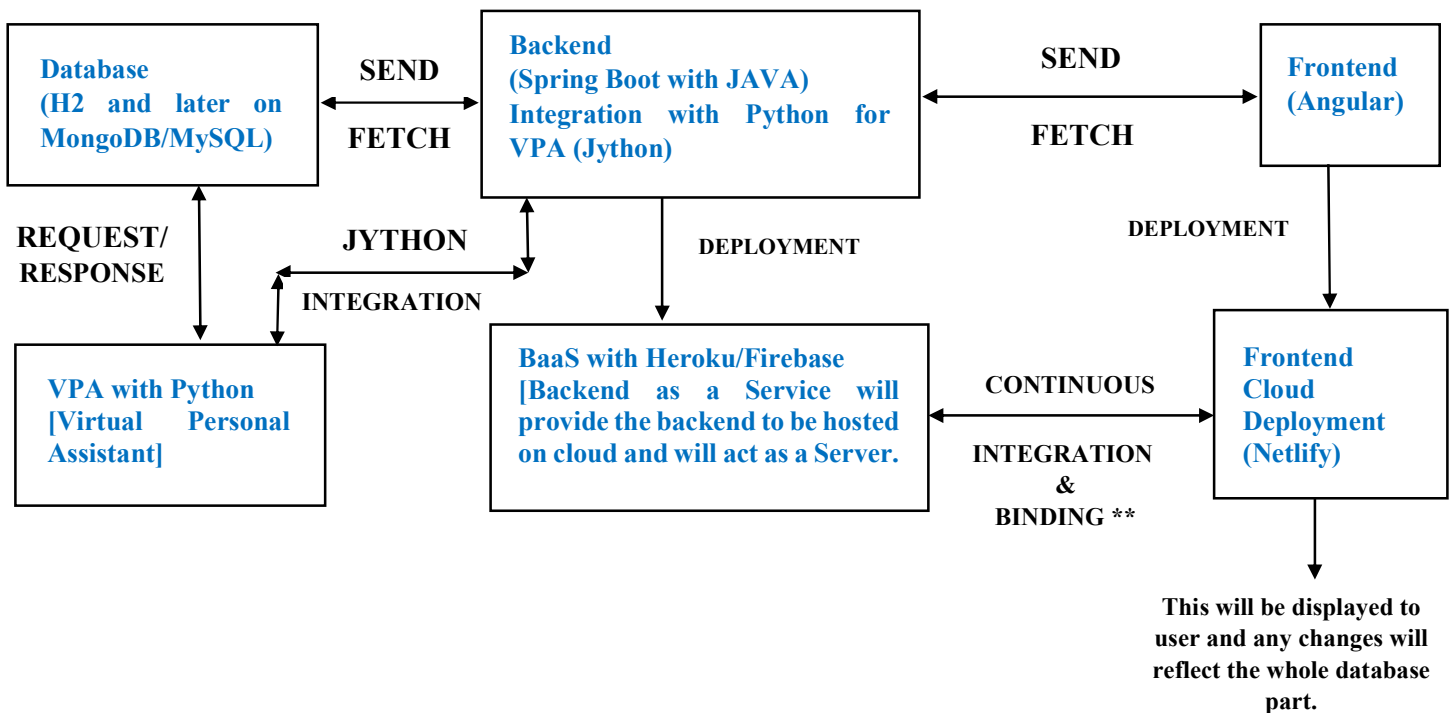
The deployment will be gone in two steps:

- i) BaaS (Backend as a Service) → The backend will be deployed either in Heroku or Firebase (Firebase is more preferred for the web socket for instant reloading of the effect done in website by fetching API). Now this deployment will act as a Server running independently irrespective to the frontend deployment.
- ii) Frontend Deployment (Netlify) → The frontend part will be deployed w.r.t. the backend server and will perform the actions as per the user and admin end choices.

Now, comes the integration part with the backend part. The main and innovative thing is to develop VPA with Python and later on integrating with Spring Boot with JAVA, with Jython dependency package [Check for the referral]. So the VPA will automatically deployed with the backend part and so the VPA will run irrespective to frontend and can easily fetch data from the JSON format.

Application: -

A project architecture will be best describing the Synopsis: -



**Data binding must take place in smooth functionality.