Gipher App Case Study

1	**Name of the Project**	**Gipher App**
2	**Objective/ Vision**	Build an app to search for gif images and add them to wish list. Create a Dashboard view with header sections to display Reaction, Entertainment, Sports, Sticker GIFs, The application needs to fetch from giphy.com api - This Dashboard is the default view to be shown. - The 3 sections are: - Header section having Reaction, Entertainment, Sports, Sticker link. - Search section - Show search result
3	**Users of the System**	All Internet users
4	**Functional Requirements**	**1)** Home Page should consist of Register page through which a user can register. Upon registration, the user should be able to login into his/her account. **2)** User home page should also have options to edit the profile and change his/her password. **3)** Header view– It should have links for popular categories of GIFs which user can view from 3rd party service provider (giphy.com, in this case) under the show result section **4)** Upon click on a category of GIF in header section, search result should display in desired section **5)** Search functionality: Search for a particular category of GIF Images **6)** Logged in User should be able to add Gifs to the wishlist and manage them
5	**Non-functional requirements**	a) App should be accessible from any location with access to the Internet. b) App should be responsive to display consistently across multiple device screens. c) App should have an intuitive UI that can be operated by novice-expert Internet users
6	**Tools and Technologies to be used**	**1.** VCS : Gitlab **2.** Middleware : Spring Boot **3.** Frond end : Angular/React **4.** Data Store : MongoDB / MySQL **5.** Testing : JUnit, Mocha, Chai, Jest, Protractor **6.** Container : Docker **7.** Bug Fix : Sonarlint **8.** CI : Gitlab

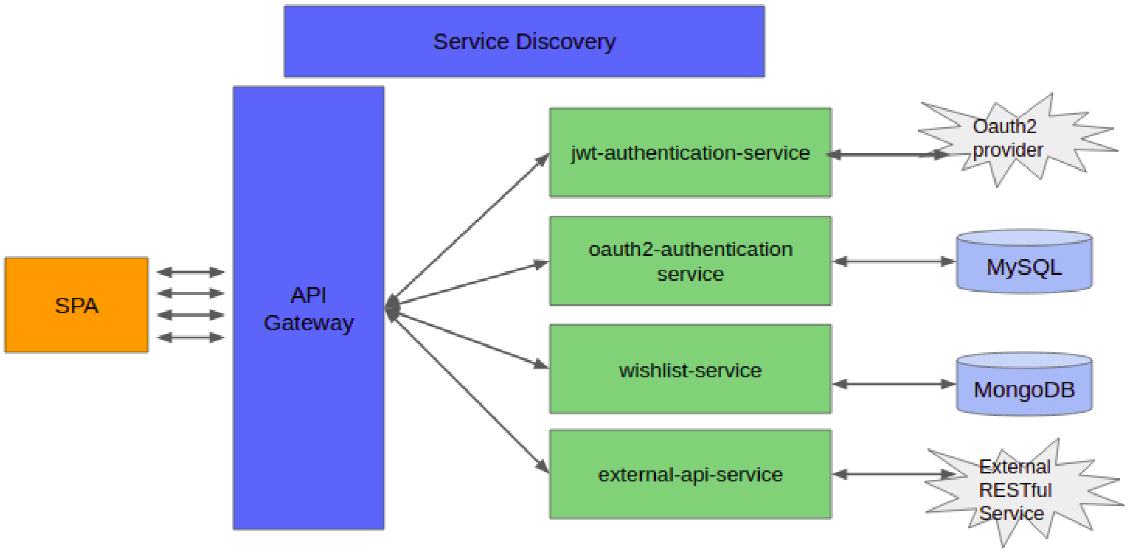
User Stories

1	**As a user I should be able to register with the application so that I can login and use the functionalities of the application.**
2	**As a user, I should be able to login with my user name and password in order to access the functionalities of the application.**
3	**As a user, I should be able to login with my Gmail account in order to access the functionalities of the application.(Optional requirement)**
4	**As a user I should be able to search resources to view their details**
5	**As a user, I should be able to save resources to a wishlist/favourite so that I can access them later**
6	**As a user, I should be able to access items saved to my wishlist/favourite**

Notes:

- The application should be based on microservices architecture
- API Gateway pattern should be implemented using Spring Cloud Gateway
- Services should register themselves with Eureka Service Discovery server.
- All layers of microservices should be covered with automated unit and integration tests
- All microservice endpoints should have API documentation

High Level Architecture Diagram


Service Discovery
SPA
API Gateway
jwt-authentication-service
oauth2-authentication service
wishlist-service
external-api-service
Oauth2 provider
MySQL
MongoDB
External RESTful Service