



# TO-DO Tracker

**Presented By:**

Shashank Malviya

Tamojit Das

Sai Saranya

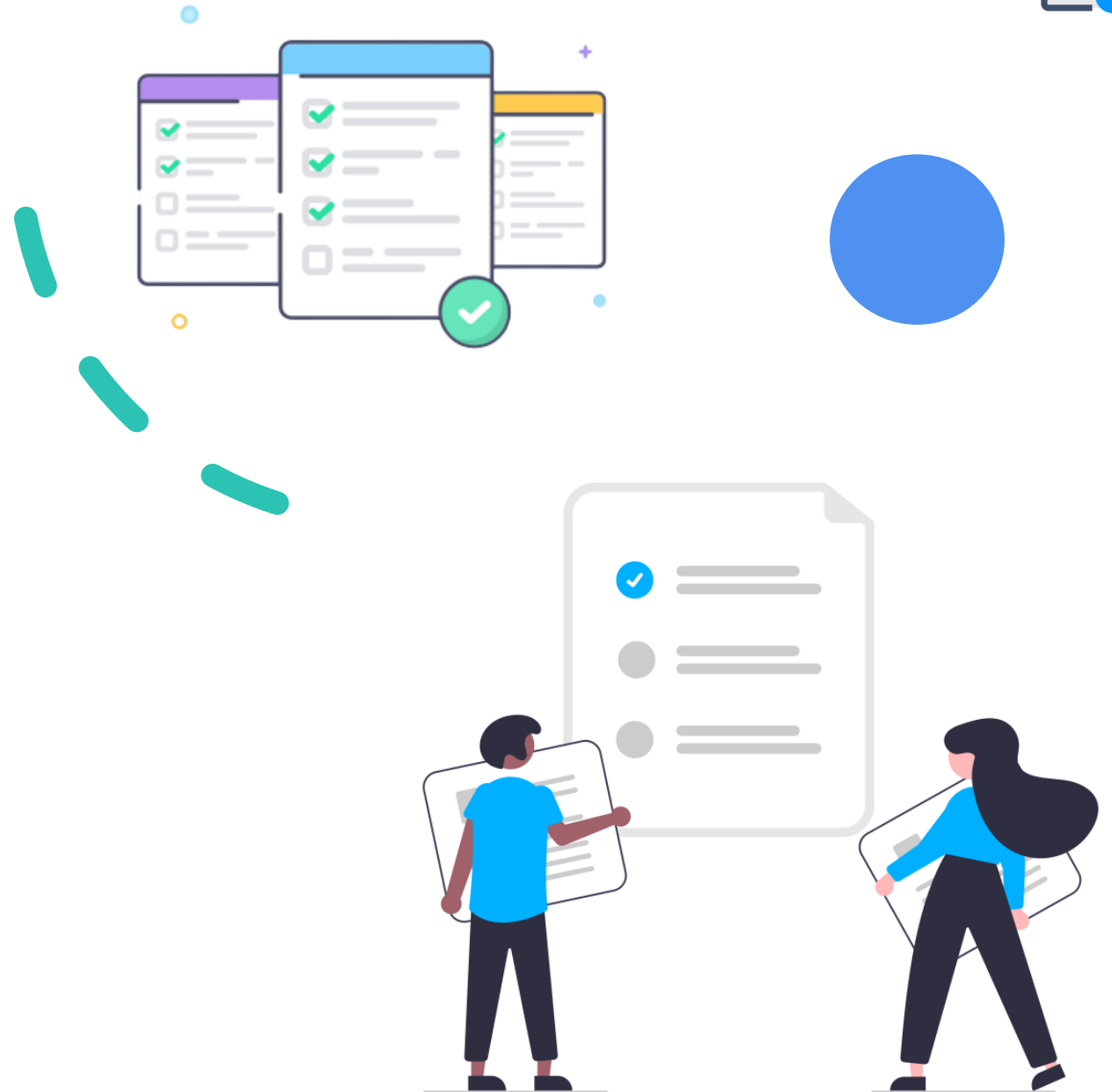
**Mentor:**

Mr. Rakesh Pandit



# About TO-DO Tracker

Captures quick ideas, add notes, check tasks off your To-Do list. One of the fundamental features for any productivity tool is a To-Do list. With the To-Do Tracker, you can create and share notes and lists. The To-Do Tracker helps you to accomplish a set of tasks in order to achieve a goal wherever you are. To-Do Tracker surfaces the right tasks at the right time, so you always know what to focus on next.





# Features of our Project

1

User can create as many to do he/she wishes to, categorize and organize your tasks.

2

Get reminded about self-imposed deadlines.

3

Mark the tasks as completed. Modify/Update and remove the tasks from list as required.

4

Offers user friendly interface.

5

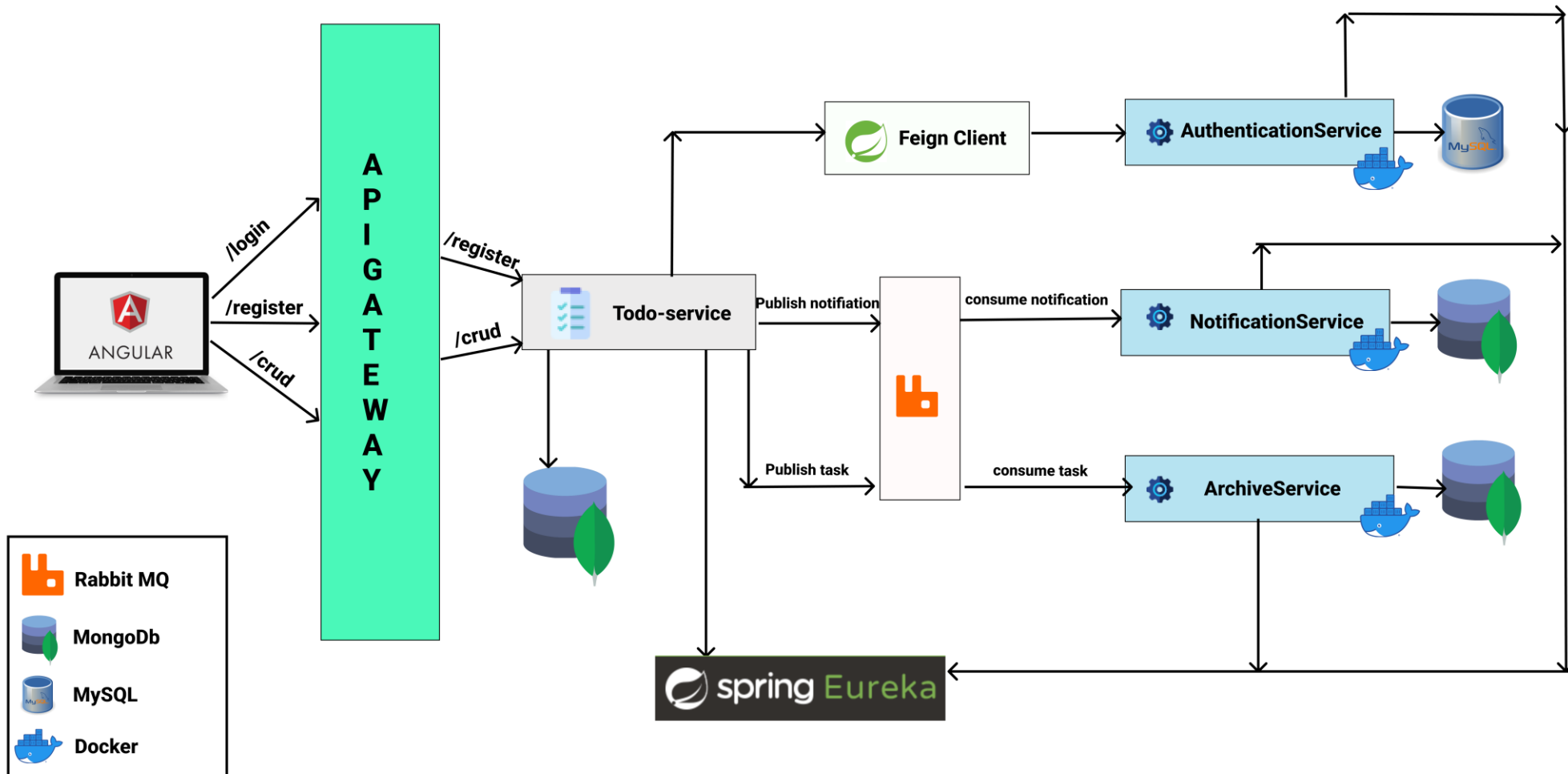
List the archived tasks as and when required.



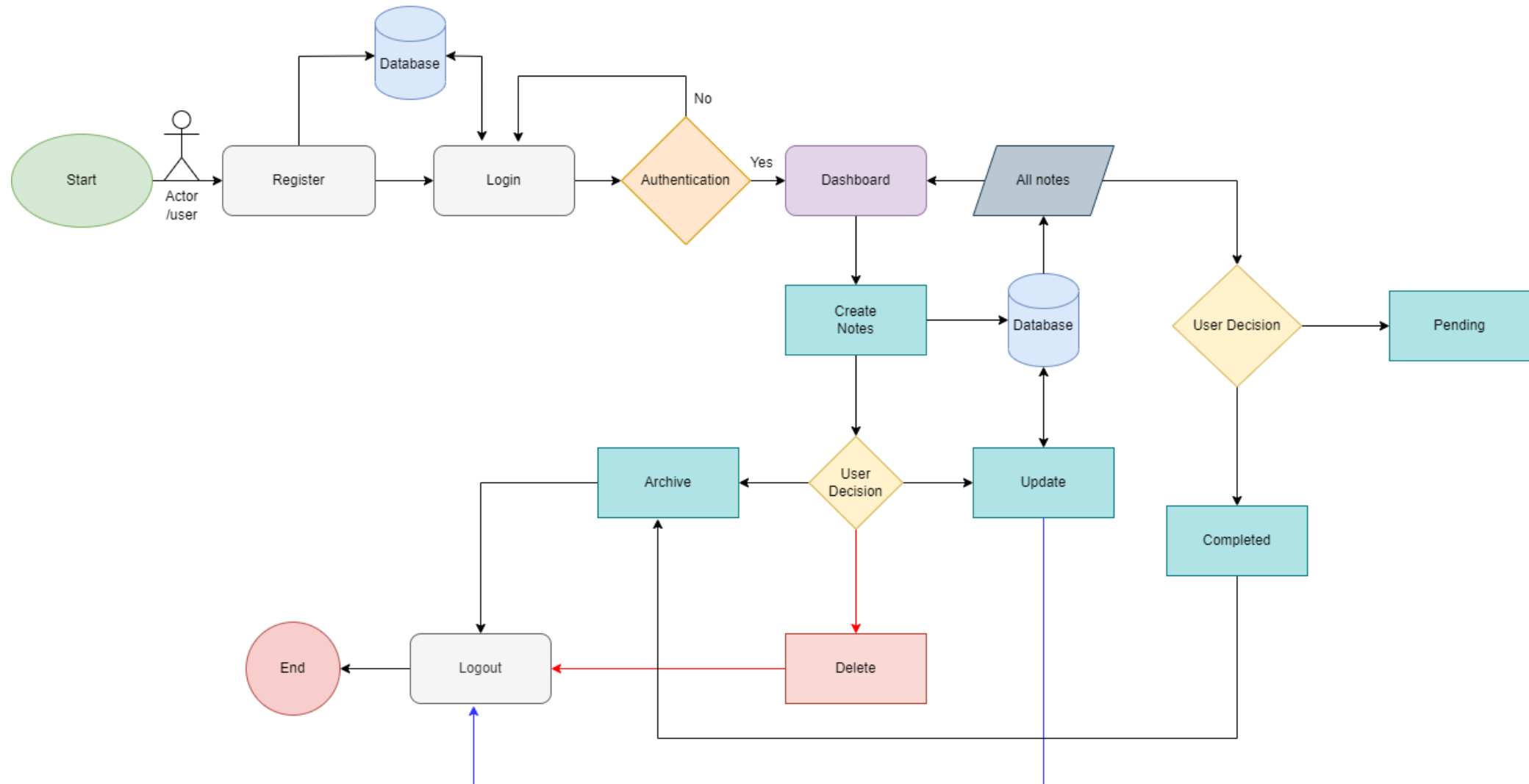
# Technologies used



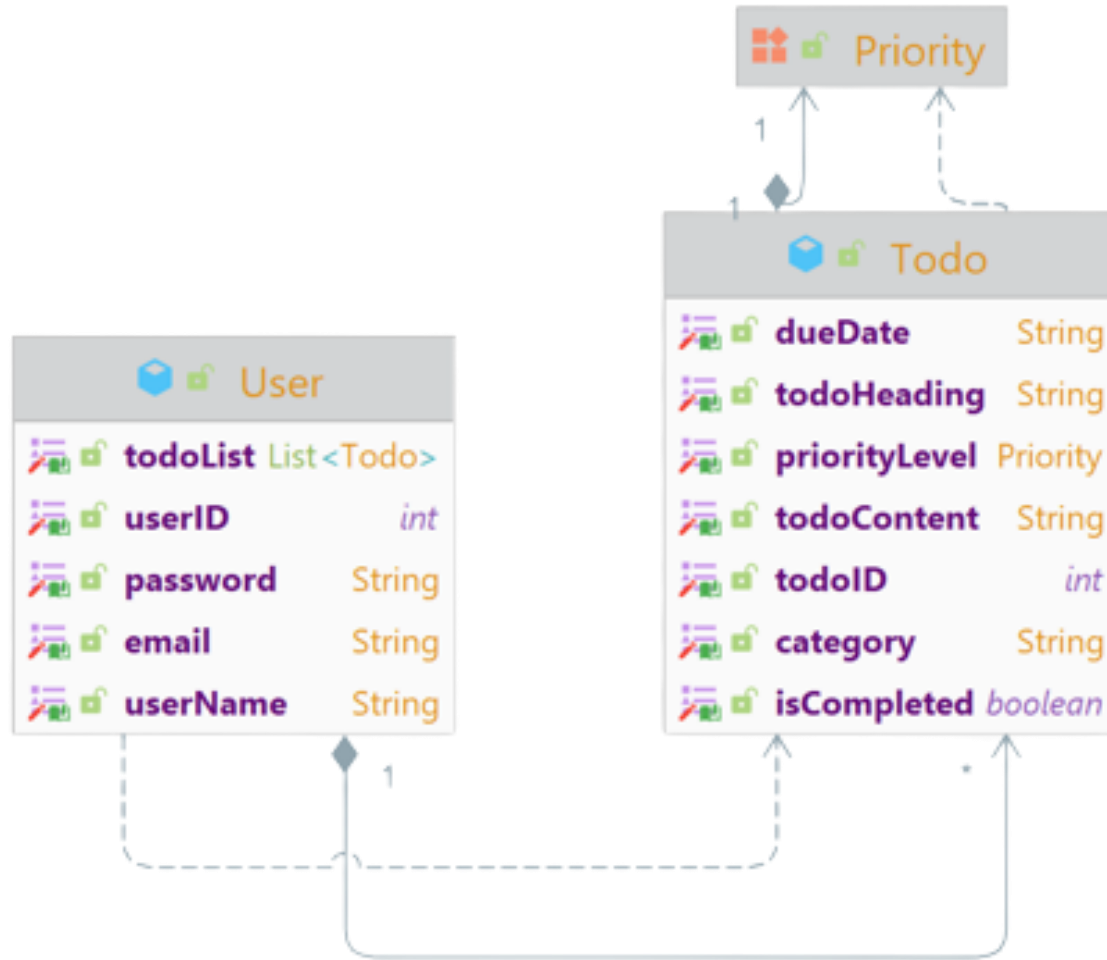
# Architecture Diagram



# Workflow Diagram



# Class Diagram





# Features that we have implemented at Backend

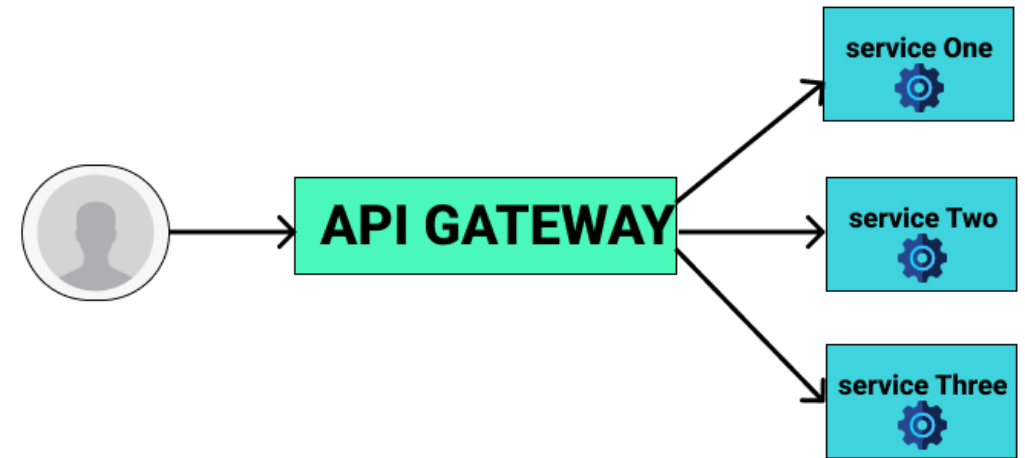
- API Gateway Service
- Eureka server
- JWT for Authentication
- Feign Client
- Rabbit MQ
- Docker





# API Gateway Service

- The API Gateway offers a reverse proxy to redirect or route requests to the endpoints of the internal microservices. We have used Spring Cloud's API Gateway Service.
- It insulates user's information on the way to the server .
- It makes it possible for the user to access microservices without knowing their actual location .





# Eureka Server

- Eureka Server is an application that holds the information about all client service applications.
- Eureka Server continuously sends heartbeat to each of its client-service application and checks whether they are up and working .
- Eureka server is also known as Discovery Server and helps in load balancing and failure handling.



**Spring Eureka**



# JSON Web Token for Authentication

- It is a standard for token-based authentication.
- It uses the HMAC SHA-256 signing algorithm .
- We have implemented JWT for authentication of user in our project.





# Feign Client

- Synchronous communication between microservices is possible by using Feign Client.
- In this approach the calling service will wait until the caller service responds.
- In Synchronous communication , a chain of requests is created between the microservices while serving the client request.



**Spring Feign Client**



# Rabbit MQ Server

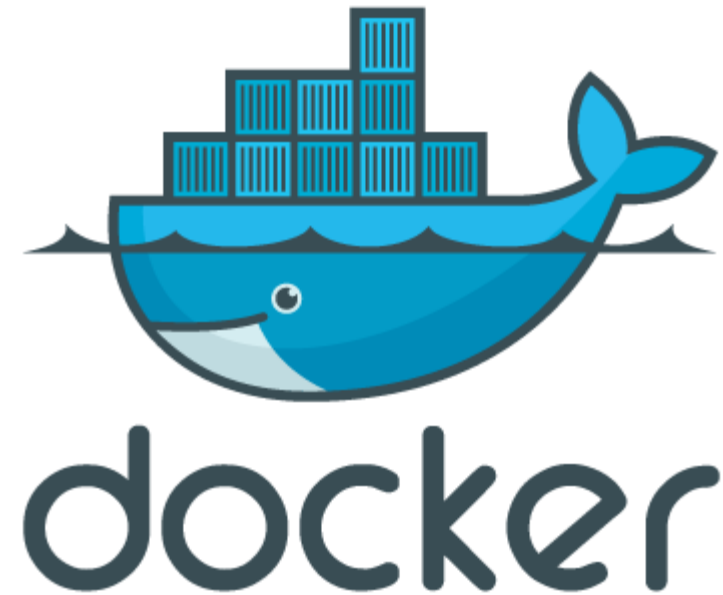
- Rabbit MQ is one of the most popular open-source message brokers that gives all applications a common platform to send and receive messages. Here the messages get safe place to live until received.
- It uses Advanced Message Queuing Protocol.
- In Asynchronous Communication , the client doesn't wait for a response from the server. The flow of communication is one way.





# Docker

- Docker is a container technology to build, ship and run any app at any operating system or platform.
- When a docker image runs , it becomes a container , which can be created , started , stopped and deleted .
- It is cost effective and helps in fast deployment.





# Technologies used in Frontend

- Angular
- Nodejs
- HTML, CSS
- Material Design Bootstrap(MDB)



# Database used

- MySQL
- MongoDB





# Project Walkthrough

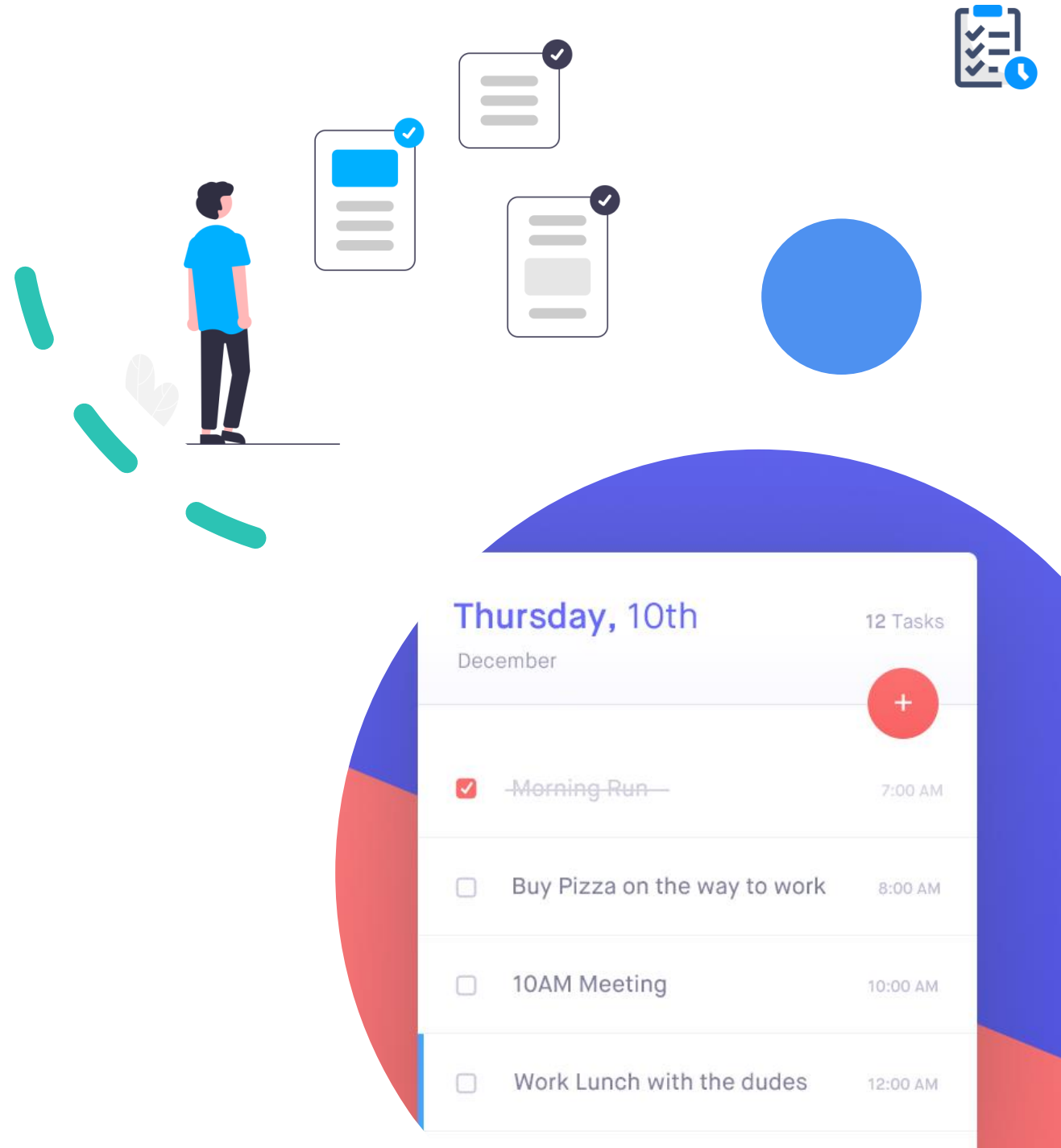


TO-DO Tracker

# Further improvements

Since we have created this product in 12 days for our initial release, in our later versions, we will incorporate the following enhancements:

- Make the UI more enriched.
- Forgot password features and more based on feedback.
- And many more.





Any Questions?





Thank you

- Shashank Malviya
- Tamojit Das
- Sai Saranya