



SOFTWARE DESIGN DOCUMENT

**Presentation by Cogitater Sigauke,
Merry Mekonnen,
Redi Negash,
Pyungkang Hong**



MyRecipes

A recipe sharing web application

A recipe search web application

An application develop your cooking skills

An application place to meet new friends

A place to be inspired



Problem

Some cooks are eager to share their recipes
People don't know how to cook

Third party APIs and Technologies

Spring Boot

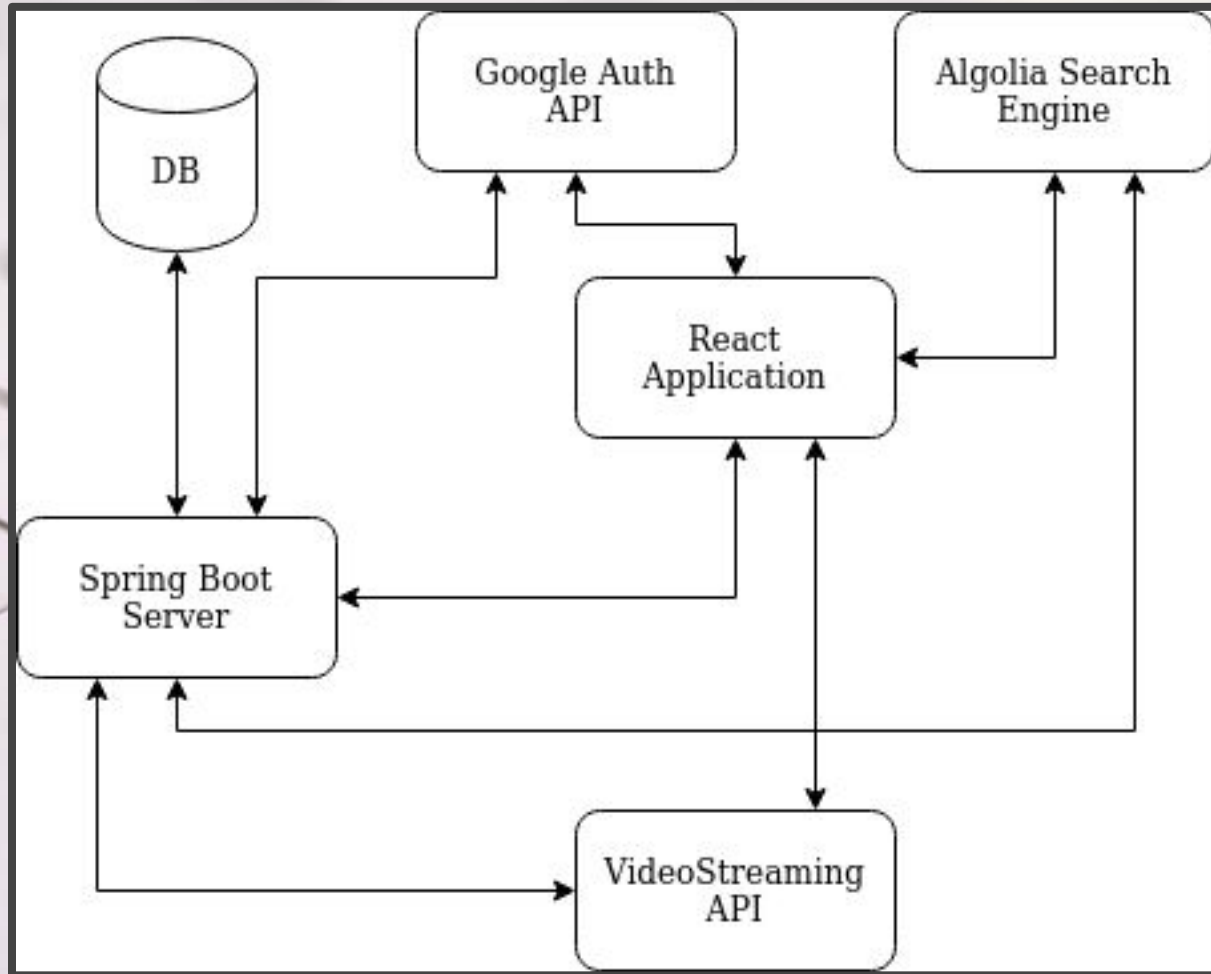
React.js

Algolia Search Engine

WebRCT

Google Auth

General Architecture View





Major Features

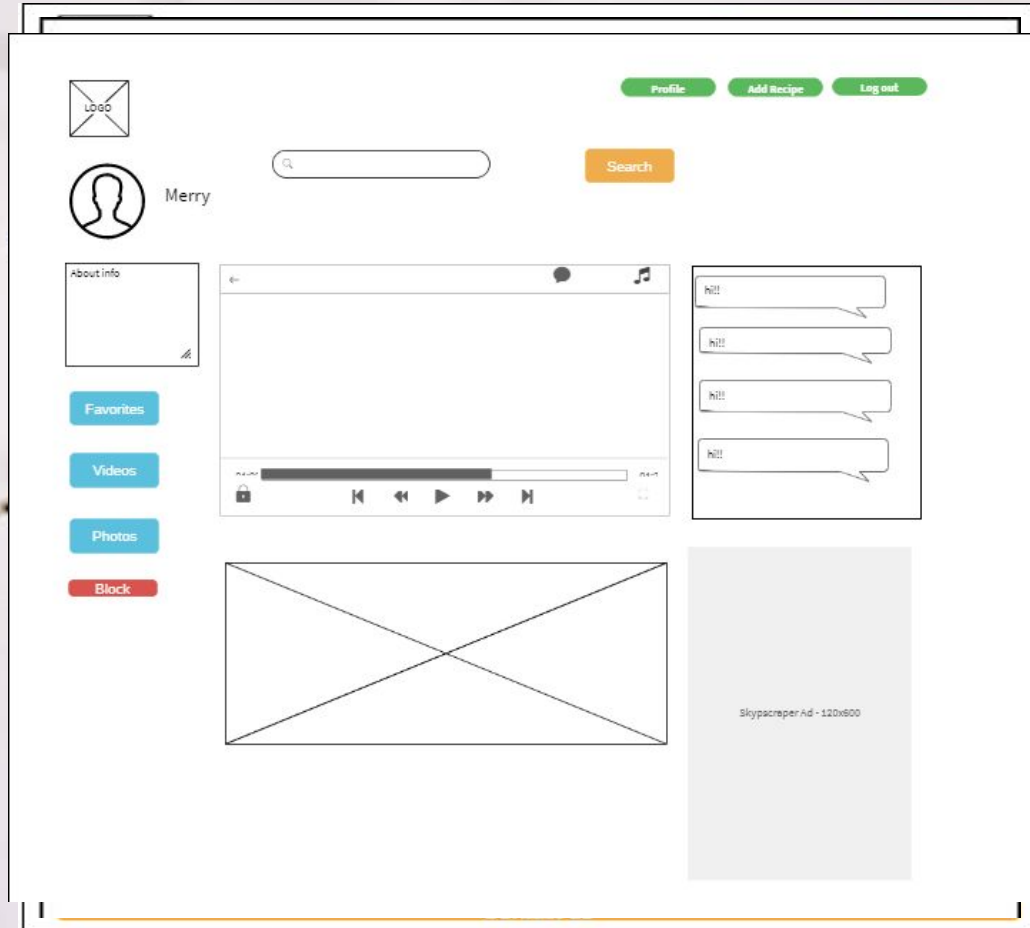
General Text Based Search with different filters

Display of uploaded recipes in different categories

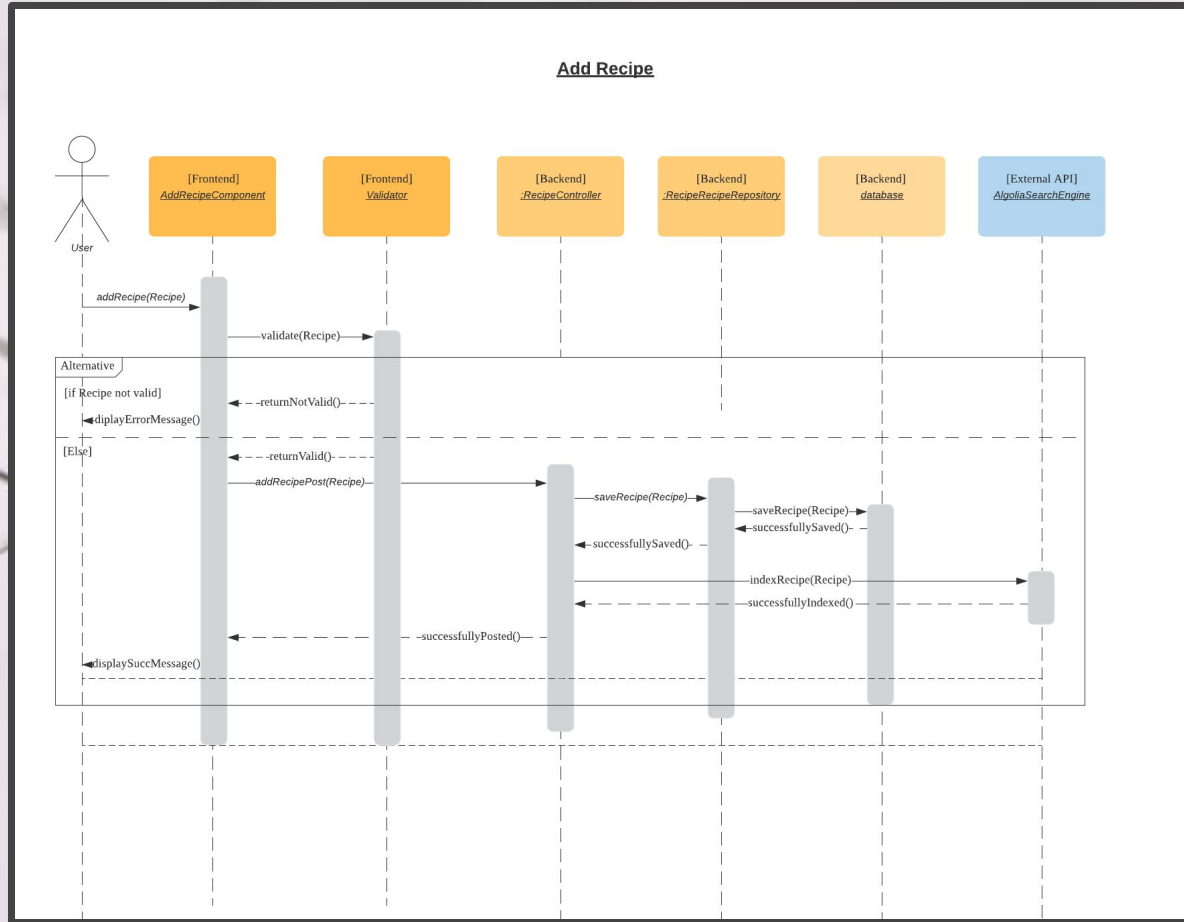
Live streaming

Messages and inbox

User Interface Diagram



Sequence Diagram

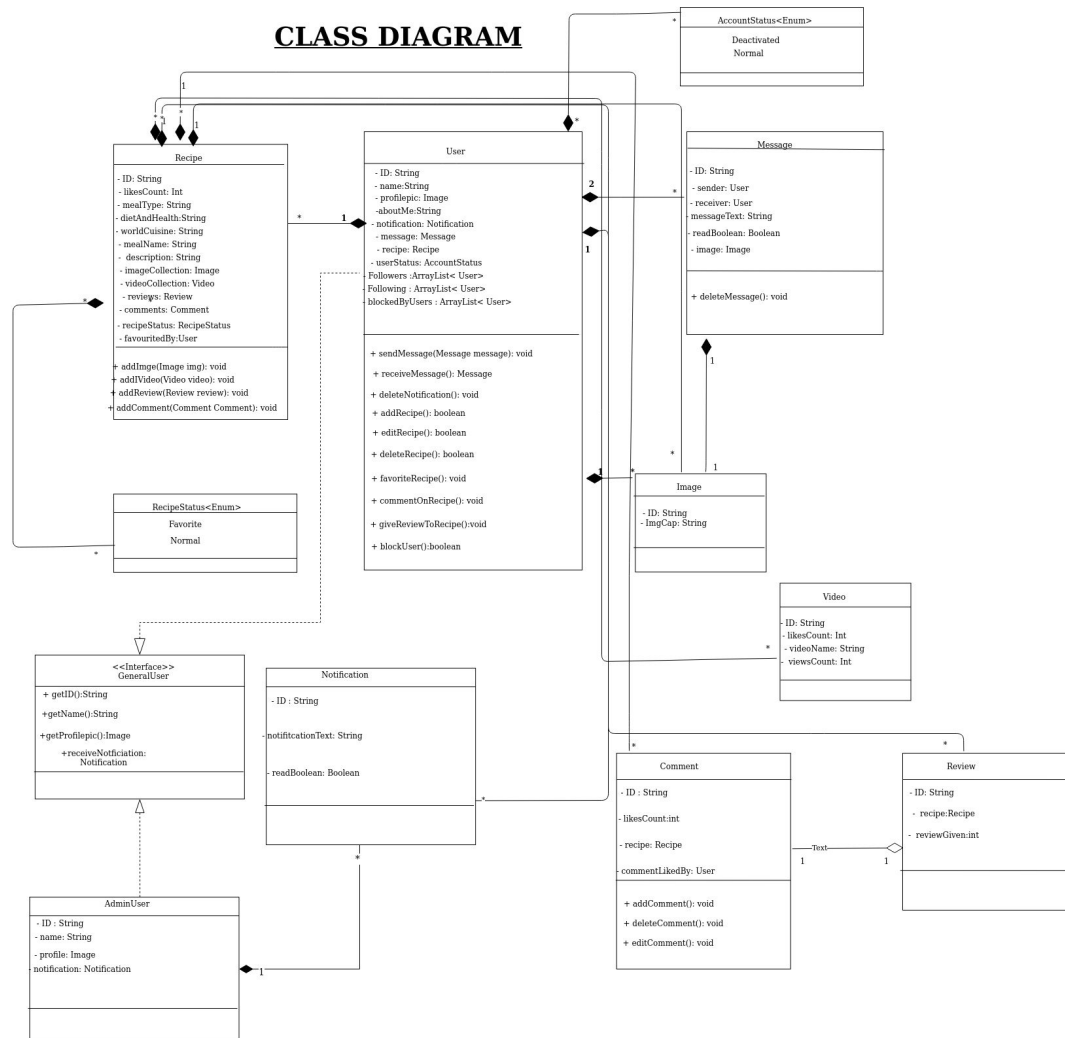


CLASS DIAGRAM

```

classDiagram
    class Recipe {
        -ID: String
        -likesCount: Int
        -mealType: String
        -dietAndHealth: String
        -worldCuisine: String
        -mealName: String
        -description: String
        -imageCollection: Image
        -videoCollection: Video
        -reviews: Review
        -comments: Comment
        -recipeStatus: RecipeStatus
        -favouriteBy: User
        +addImage(Image img): void
        +addVideo(Video video): void
        +addReview(Review review): void
        +addComment(Comment Comment): void
    }
    class User {
        -ID: String
        -name: String
        -profilePic: Image
        -aboutMe: String
        -notification: Notification
        -message: Message
        -recipe: Recipe
        -userStatus: AccountStatus
        -followers: ArrayList<User>
        -following: ArrayList<User>
        -blockedBy: ArrayList<User>
        +sendMessage(Message message): void
        +receiveMessage(): Message
        +deleteNotifications(): void
        +addRecipe(): boolean
        +editRecipe(): boolean
        +deleteRecipe(): boolean
        +favoriteRecipe(): void
        +commentOnRecipe(): void
        +giveReviewToRecipe(): void
        +blockUser(): boolean
    }
    class Message {
        -ID: String
        -sender: User
        -receiver: User
        -messageText: String
        -readBoolean: Boolean
        -image: Image
        +deleteMessage(): void
    }
    class Image {
        -ID: String
        -imgCap: String
    }
    class Video {
        -ID: String
        -likesCount: Int
        -videoName: String
        -viewsCount: Int
    }
    class Comment {
        -ID: String
        -likesCount: Int
        -recipe: Recipe
        -commentLikedBy: User
        +addComment(): void
        +deleteComment(): void
        +editComment(): void
    }
    class Review {
        -ID: String
        -recipe: Recipe
        -reviewGiven: Int
    }
    class AdminUser {
        -ID: String
        -name: String
        -profile: Image
        -notification: Notification
    }
    class GeneralUser {
        <<interface>>
        +getID(): String
        +getName(): String
        +getProfilePic(): Image
        +receiveNotification: Notification
    }
    class RecipeStatus {
        <<enum>>
        Favorite
        Normal
    }
    class AccountStatus {
        <<enum>>
        Deactivated
        Normal
    }
    class Notification {
        -ID: String
        -notificationText: String
        -readBoolean: Boolean
    }

    GeneralUser <|-- AdminUser
    RecipeStatus <|-- Recipe
    AccountStatus <|-- User
    Recipe "1" *-- "*" User
    User "1" *-- "*" Message
    User "1" *-- "*" Image
    User "1" *-- "*" Video
    User "1" *-- "*" Comment
    User "1" *-- "*" Review
    AdminUser "1" *-- "*" Recipe
    AdminUser "1" *-- "*" Notification
    
```



Design Decisions

Choosing Google Auth over implementing our own login system

Using algolia search engine rather than writing our own

Choosing video streaming API over coding our own from scratch

Choosing mongoDb over MySql

Schedule and Deliverables

The screenshot displays a Trello board titled "Schedule" in a green-themed interface. The board is organized into four columns, each representing a project milestone. The top navigation bar includes icons for home, boards, and search, along with a notification banner that reads: "Hey cogitater.sigauke, we need you to confirm your email address. [Confirm email](#) · [Remind me later](#)".

Below the navigation bar, the board is categorized by "Personal" and "Private" tabs, with a filter set to "C MM A P RY". The right sidebar contains a "Butler" button and a "Show Menu" button.

The four milestones and their tasks are as follows:

- Project Milestone1**
 - Create the Models for the DB (Due: Apr 24, Assignees: C, P)
 - Create React Components (Due: Apr 24, Progress: 0/3, Assignees: MM, RY)
 - Prepare the APIs and Libraries we expect to use (Due: Apr 25, Assignees: MM, P)
 - Implement the HomePage (Due: Apr 25, Progress: 0/3, Assignees: C, RY)
- Project Milestone2**
 - Implement Recipe - Backend (Due: May 6, Progress: 0/5, Assignees: C, RY)
 - Implement Recipe - Frontend (Assignees: C, P)
 - Implement Comment - Backend (Due: May 11, Progress: 0/4, Assignees: MM, P)
 - Implement Reviews - Backend (Due: May 8, Progress: 0/1, Assignees: C, P)
 - Implement Comment: Frontend (Assignees: MM, P)
- Project Milestone3**
 - Implement Chat Box - Backend (Assignees: C, MM)
 - Implement Notification - Frontend (Assignees: P, RY)
 - Implement Chat Box - Frontend (Due: May 18, Assignees: MM, RY)
 - Implement Live Stream - Backend (Due: May 20, Progress: 2, Assignees: C, RY)
 - Implement Live Stream - Frontend (Assignees: MM, P)
- Project Milestone4**
 - Testing (Due: May 29, Progress: 2, Assignees: C, MM, RY)
 - Ads (Due: May 28, Progress: 1, Assignees: C, MM, P)

Each milestone column has a "+ Add another card" button at the bottom. The right sidebar also features a "+ Add another list" button.

A close-up photograph of a white puzzle piece being placed into a larger white puzzle. The puzzle piece is slightly raised, showing its interlocking edges. The background is a soft-focus view of the rest of the puzzle. Overlaid on the center of the puzzle piece is the text "Thank You !!!" in a bold, orange, sans-serif font.

Thank You !!!