Design Document for "Recommend Me"

Group <1_yn_6>

Member1 Name:  Caelan Herzberg

Contribution: Block Diagram classes and design description

Member2 Name: Josh Podlich

Contribution: Block Diagram classes and design description
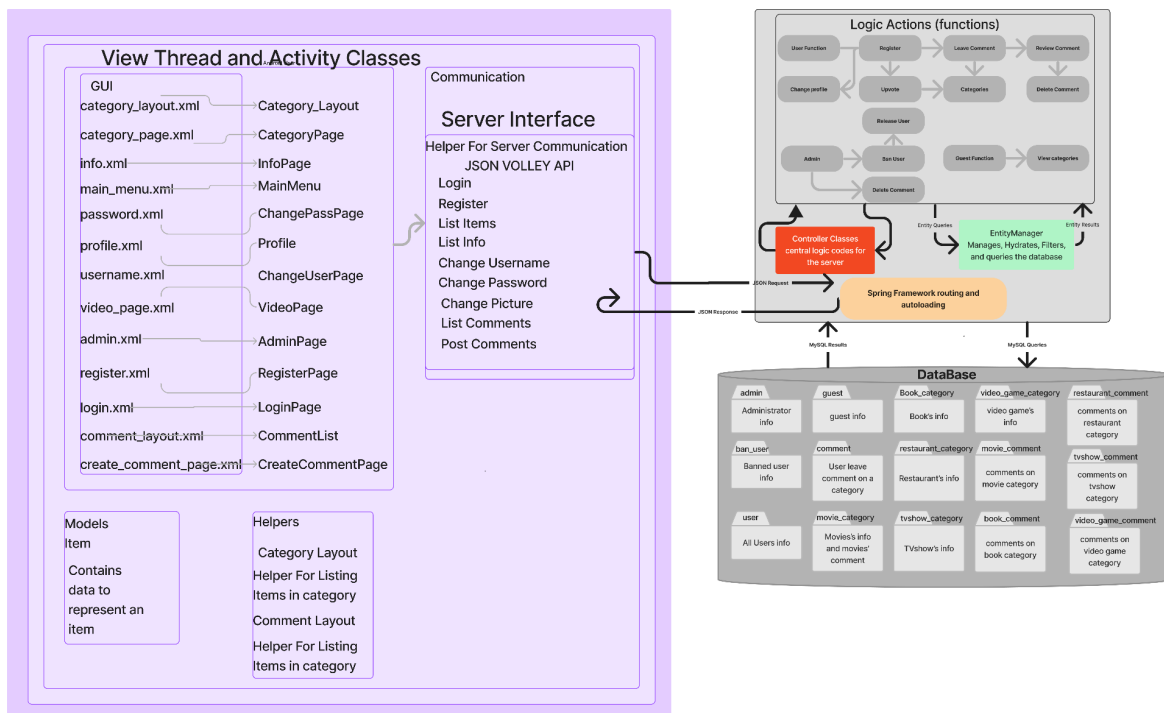
Member3 Name: Jian Shi

Contribution: Block Diagram database&logic action formatting and design description

Member4 Name: Wenqin Wu

Contribution: Block Diagram logic action and tables list

Page 2:



**View Thread and Activity Classes**

GUI

| XML | Class |
|---|---|
| category_layout.xml | → Category_Layout |
| category_page.xml | → CategoryPage |
| info.xml | → InfoPage |
| main_menu.xml | → MainMenu |
| password.xml | → ChangePassPage |
| profile.xml | → Profile |
| username.xml | → ChangeUserPage |
| video_page.xml | → VideoPage |
| admin.xml | → AdminPage |
| register.xml | → RegisterPage |
| login.xml | → LoginPage |
| comment_layout.xml | → CommentList |
| create_comment_page.xml | → CreateCommentPage |

Communication

**Server Interface**

Helper For Server Communication
JSON VOLLEY API

Login
Register
List Items
List Info
Change Username
Change Password
Change Picture
List Comments
Post Comments

Models
Item

Contains data to represent an item

Helpers

Category Layout
Helper For Listing Items in category
Comment Layout
Helper For Listing Items in category

**Logic Actions (functions)**

| User Function | Register | Leave Comment | Review Comment |
| Change profile | Upvote | Categories | Delete Comment |
| | Release User | | |
| Admin | Ban User | Guest Function | View categories |
| | Delete Comment | | |

Controller Classes
central logic codes for the server

EntityManager
Manages, Hydrates, Filters, and queries the database

Entity Queries / Entity Results

Spring Framework routing and autoloading

JSON Request
JSON Response

MySQL Results / MySQL Queries

**DataBase**

| admin | guest | Book_category | video_game_category | restaurant_comment |
| Administrator info | guest info | Books info | video game's info | comments on restaurant category |
| ban_user | comment | restaurant_category | movie_comment | tvshow_comment |
| Banned user info | User leave comment on a category | Restaurant's info | comments on movie category | comments on tvshow category |
| user | movie_category | tvshow_category | book_comment | video_game_comment |
| All Users info | Movies's info and movies' comment | TVshow's info | comments on book category | comments on video game category |

Page 3:

## Android User Gui:

The android gui is composed of xml files paired with corresponding java class files that define an activity. WIthin the box labeled View Thread and Activity Classes, there are listed 13 activities each defined by a java class for logic and xml file for layout.

## Android Helpers:

The program will use 2 helpers. The first helper is Category Layout, which defines the correct dynamic layout to list all items in a category. The second helper is Comment Layout, which dynamically lists all the comments from an item within a category. These helpers allow the app to list as much data as is required on the fly.

## Android model:

The app will use one model, named item, to represent the data associated with a specific item. This model helps to conveniently be able to get and set data values with ease within multiple classes.
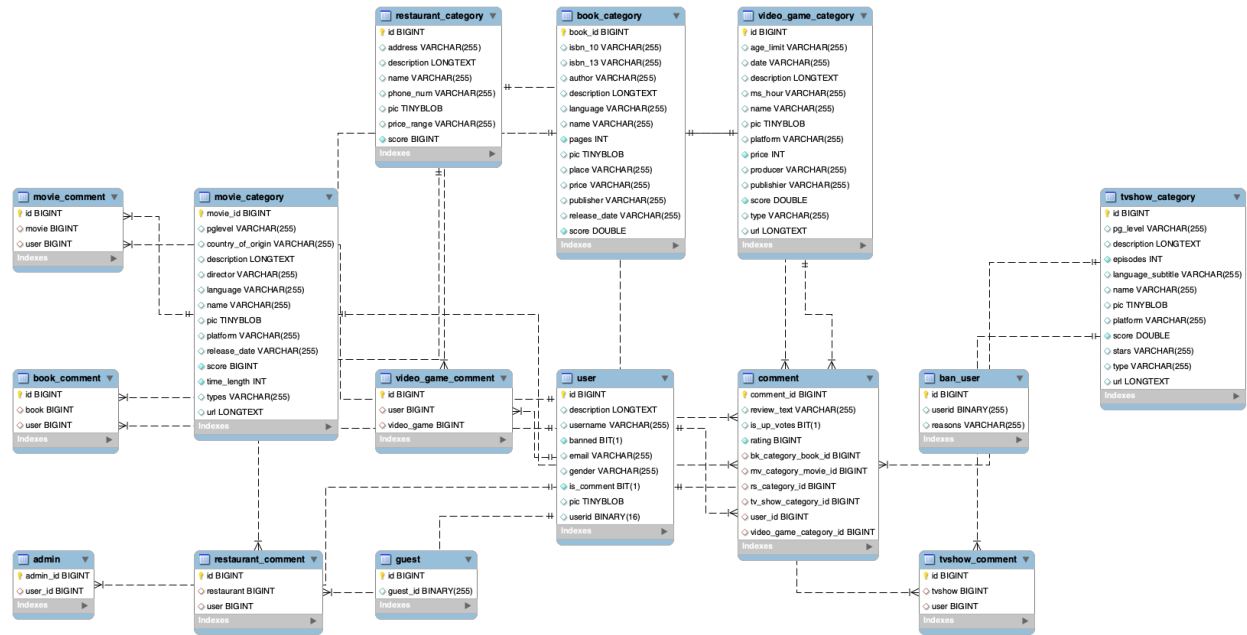
## Android Communication:

The app will use the Volley API in order to send and receive JSON requests asynchronously with the MySql database.

## Spring Framework

Spring framework is a lightweight Java development framework that provides a platform for application development. It was created to address the complexities of enterprise application development. One of the main advantages of the framework is the layered architecture, which allows the user to choose which components to use, while providing an inherited framework for J2EE application development.

In our project, most functions were written into the controller package, and these functions will be able to handle data creation and queries. Variables definitions were written in the model package to define the type of the item in the SQL tables.

Page 4 (Table Diagram):

**restaurant_category**
- id BIGINT
- address VARCHAR(255)
- description LONGTEXT
- name VARCHAR(255)
- phone_num VARCHAR(255)
- pic TINYBLOB
- price_range VARCHAR(255)
- score BIGINT
- Indexes

**book_category**
- book_id BIGINT
- isbn_10 VARCHAR(255)
- isbn_13 VARCHAR(255)
- author VARCHAR(255)
- description LONGTEXT
- language VARCHAR(255)
- name VARCHAR(255)
- pages INT
- pic TINYBLOB
- place VARCHAR(255)
- price VARCHAR(255)
- publisher VARCHAR(255)
- release_date VARCHAR(255)
- score DOUBLE
- Indexes

**video_game_category**
- id BIGINT
- age_limit VARCHAR(255)
- date VARCHAR(255)
- description LONGTEXT
- ms_hour VARCHAR(255)
- name VARCHAR(255)
- pic TINYBLOB
- platform VARCHAR(255)
- price INT
- producer VARCHAR(255)
- publisher VARCHAR(255)
- score DOUBLE
- type VARCHAR(255)
- url LONGTEXT
- Indexes

**tvshow_category**
- id BIGINT
- pg_level VARCHAR(255)
- description LONGTEXT
- episodes INT
- language_subtitle VARCHAR(255)
- name VARCHAR(255)
- pic TINYBLOB
- platform VARCHAR(255)
- score DOUBLE
- stars VARCHAR(255)
- type VARCHAR(255)
- url LONGTEXT
- Indexes

**movie_comment**
- id BIGINT
- movie BIGINT
- user BIGINT
- Indexes

**movie_category**
- movie_id BIGINT
- pglevel VARCHAR(255)
- country_of_origin VARCHAR(255)
- description LONGTEXT
- director VARCHAR(255)
- language VARCHAR(255)
- name VARCHAR(255)
- pic TINYBLOB
- platform VARCHAR(255)
- release_date VARCHAR(255)
- score BIGINT
- time_length INT
- types VARCHAR(255)
- url LONGTEXT
- Indexes

**book_comment**
- id BIGINT
- book BIGINT
- user BIGINT
- Indexes

**video_game_comment**
- id BIGINT
- user BIGINT
- video_game BIGINT
- Indexes

**user**
- id BIGINT
- description LONGTEXT
- username VARCHAR(255)
- banned BIT(1)
- email VARCHAR(255)
- gender VARCHAR(255)
- is_comment BIT(1)
- pic TINYBLOB
- userid BINARY(16)
- Indexes

**comment**
- comment_id BIGINT
- review_text VARCHAR(255)
- is_up_votes BIT(1)
- rating BIGINT
- bk_category_book_id BIGINT
- mv_category_movie_id BIGINT
- rs_category_id BIGINT
- tv_show_category_id BIGINT
- user_id BIGINT
- video_game_category_id BIGINT
- Indexes

**ban_user**
- id BIGINT
- userid BINARY(255)
- reasons VARCHAR(255)
- Indexes

**admin**
- admin_id BIGINT
- user_id BIGINT
- Indexes

**restaurant_comment**
- id BIGINT
- restaurant BIGINT
- user BIGINT
- Indexes

**guest**
- id BIGINT
- guest_id BINARY(255)
- Indexes

**tvshow_comment**
- id BIGINT
- tvshow BIGINT
- user BIGINT
- Indexes

Page 4 (Table List):

| Table: | Relation: | Type: |
|---|---|---|
| Admin | User | OneToOne |
| BanUser | User | OneToOne |
| BookCategory | | |
| BookComment | User<br>BookCategory | ManyToOne<br>ManyToOne |
| Comment | User<br>MoiveCategory<br>BookCategory<br>ResturantCategory<br>TVShowCategory<br>VideoGameCategory | OneToOne<br>ManyToOne<br>ManyToOne<br>ManyToOne<br>ManyToOne<br>ManyToOne |
| Guest | | |
| MovieCategory | | |
| MovieComment | User<br>MovieCategory | ManyToOne<br>ManyToOne |

| RestaurantCategory | | |
|---|---|---|
| RestaurantComment | User<br>ResturantCategory | ManyToOne<br>ManyToOne |
| TVShowCategory | | |
| TVShowComment | User<br>TVShowCategory | ManyToOne<br>ManyToOne |
| User | | |
| VideoGameCategory | | |
| VideoGameComment | User<br>VideoGameCategory | ManyToOne<br>ManyToOne |