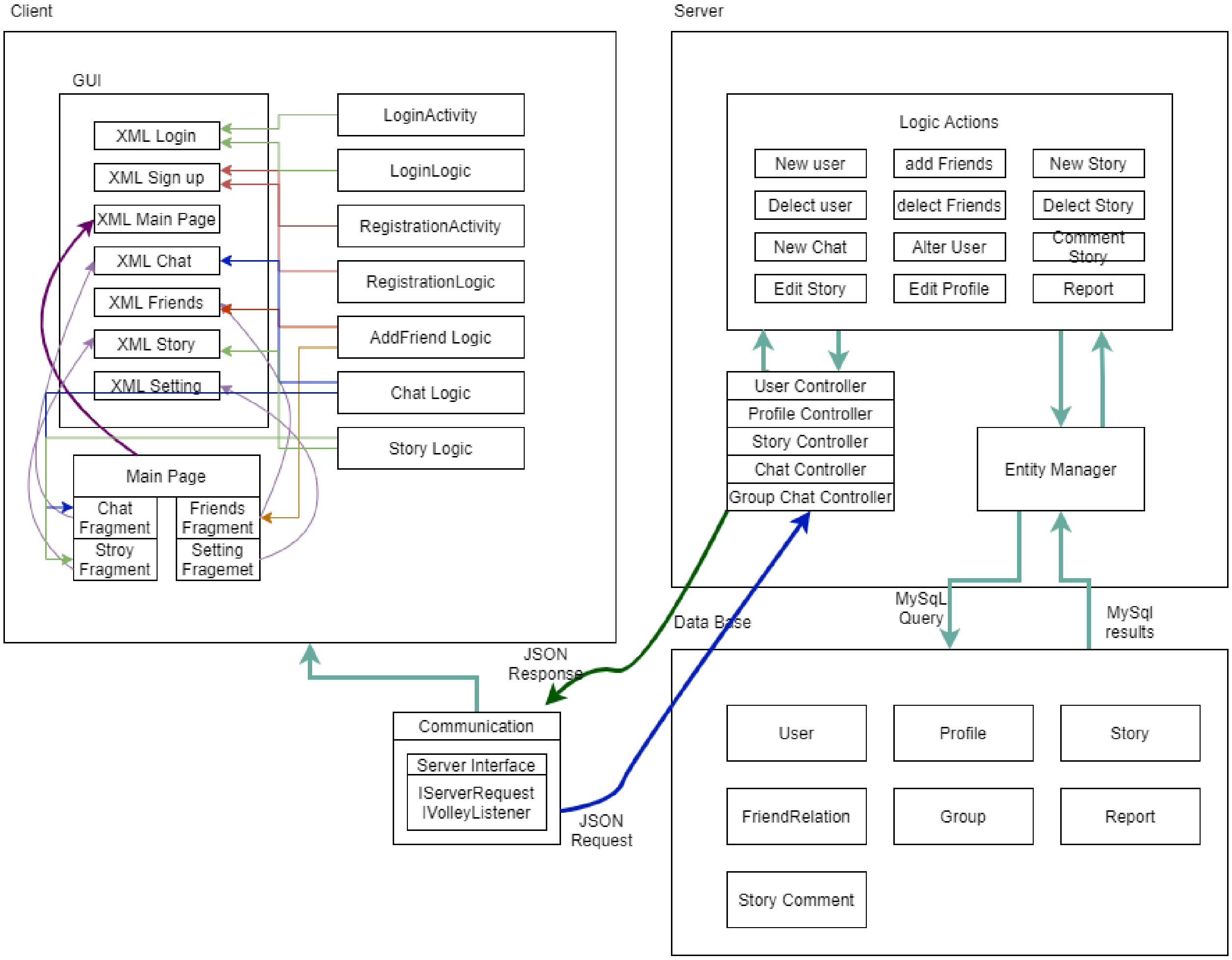
Design document for CyChat

Group VB6

Member 1 Xiwen Zhang: 25% contribution
Member 2 Chi Chang Lin: 25% contribution
Member 3 Anji Xu: 25% contribution
Member 4 Dongbo Xie: 25% contribution

Block Diagrams:



Description

Android Frontend Overview

Our CyChat application will have at least 12 different activity classes. And each activity will have a corresponding xml file which describes the styling of the page. Each activity contains corresponding logics for data communication, server processing and user interfaces. The major selling points of our application will be Stories and Friends which allows users to make posts and have live chats, yet we will build helper functions, models and interfaces to satisfy our requirements throughout the process of building the main activities.

Android Interface

We build communication interfaces(IServerRequest and IVolleyListenter) in java clients. These two interfaces will handle the network communication and obfuscate the JSON commands that will be sent to/receiver from the server. All the activities will send or receive data through the communication interfaces.

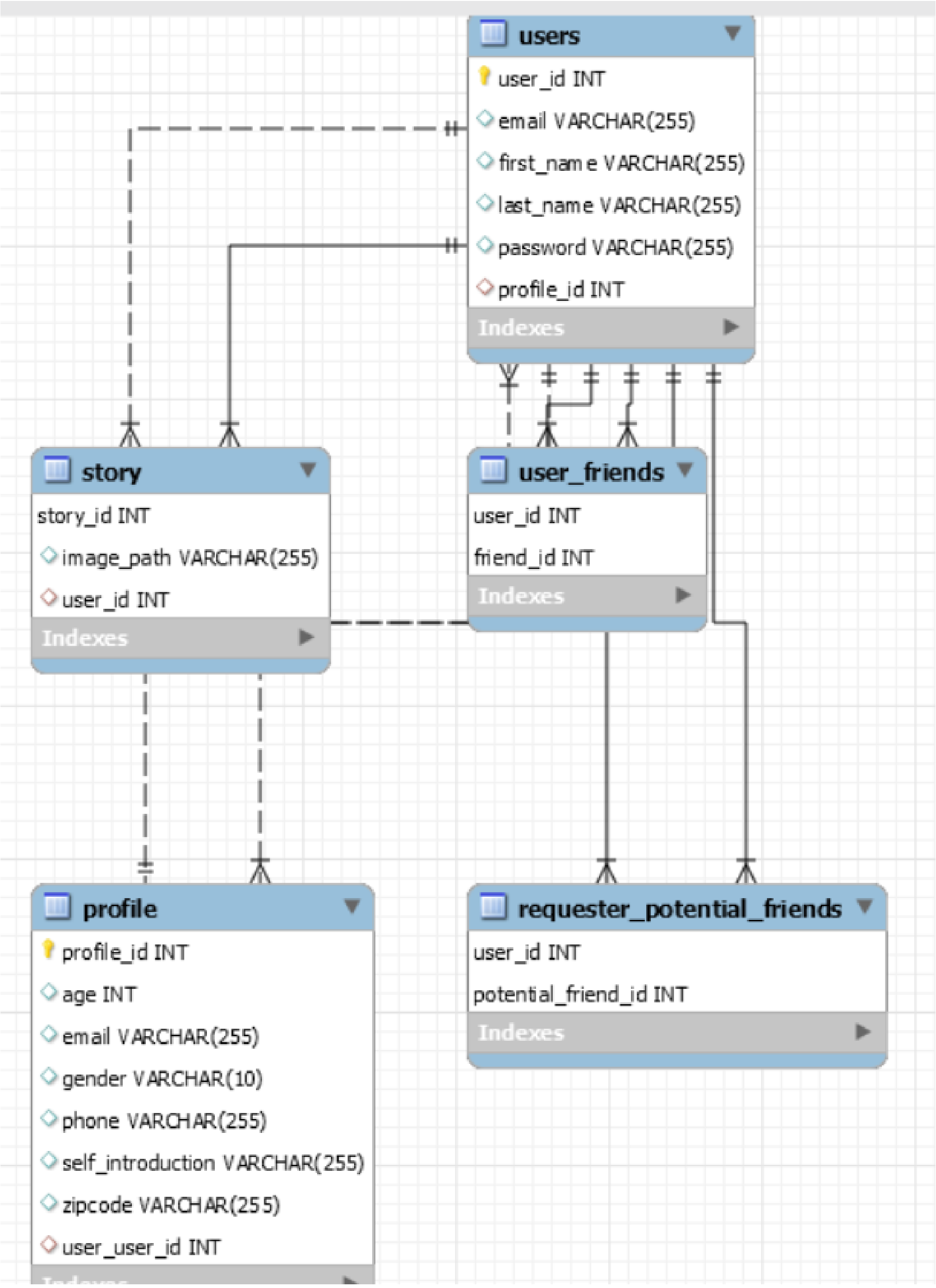
Model

Our application uses MySQL for data storage and Spring JPA repository for server side processing. We used MySQL because it’s easy to manage and access. We deployed the application on the provided Linux server and plan to use CICD and docker to automate the deployment process once all the basic functionalities of the application are satisfied. We used Volley library to assist communication between our Android client and the API server. For API request testing, we use Postman (for API creation testings) and Postman Echo (as mock server) to test our API functionalities.

Controllers

User’s information will bind with email and can be requested from the server either by Id or email. We will complete the entire route struction in each controller class with table name separation. User email/Id are utilized as a Path variable in requests and foreign key in tables for consistent information tracking. All the detailed variables are passed through JSON with post request to ensure data securities in formation passing. Unless necessary (requests without path variables or body data), post and put requests are used to ensure secure data manipulations and extractions. Two special cases in our application will be image upload and live chatting system. We will use web-socket to establish live chats and store images as binary data in MySQL. Details of the upcoming controllers will be provided and revised once we update the code for the described functionalities.

Table relations



users

user_id INT

email VARCHAR(255)

first_name VARCHAR(255)

last_name VARCHAR(255)

password VARCHAR(255)

profile_id INT

Indexes

story

story_id INT

image_path VARCHAR(255)

user_id INT

Indexes

user_friends

user_id INT

friend_id INT

Indexes

profile

profile_id INT

age INT

email VARCHAR(255)

gender VARCHAR(10)

phone VARCHAR(255)

self_introduction VARCHAR(255)

zipcode VARCHAR(255)

user_user_id INT

requester_potential_friends

user_id INT

potential_friend_id INT

Indexes