# Development of Motion Fitness Application for College Students

## 1 Introduction

Development of Motion Fitness Application for College Students is based on the Android system to develop an Android application for college students' exercise and fitness. The development of this app is mainly for the convenience of college students who lack exercise experience and lack of funds and cannot go to the gym to exercise regularly and healthily.The application uses Android Studio as a development tool and it bases on the C/S structure.This software is developed by Android System and uses MVC design pattern. The foreground handles the interface by using the Fragment. Background data storage uses MySQL as this application's database. This software is developed in strict accordance with the basic ideas of software engineering, through the problem definition, requirements analysis, system design, system implementation and program testing five stages.

## 2 Demand Analysis

### 2.1 Functional Requirement

### 2.1.1 Requirement statement

The specific functions are described as follows.

(1) Login and registration functions

New users can click the "register" button to register, and fill in the account, password, height and weight (allowing modified after login) on the registration interface. After successful registration, they can return to the login interface and login with the account password, and then enter the main interface ('Training' interface) of the system.

(2) Fitness Daily clocking function

Users can clock in on the calendar interface to mark that the work-out has been completed on one day. As long as there is a clock-in record date will be marked with a red spot. Users can intuitively observe the daily work-out through this function.

(3) 'Social News' interface (the release of news functions)

Users can see the shared information (including title, picture and content) released by different users on the 'Social News' interface. Meanwhile, users can also release news to share their own work-out process. Click "Publish new news" to upload the shared information to the social news platform for other users to browse.

(4) Comments and favorites function

Users can browse other users' news detail, and they can comment below the 'News' details. After successful comment, they can choose "favorites" to collect their favorite comment content, and can also collect comments posted by other users.

(5) Motion guidance and timing functions

On the 'Training' interface, users can choose different stages of training guidance (elementary, advanced and extreme) according to their personal conditions. The training guidance includes timing and training guidance videos. With the improvement of the training stage, the training difficulty of the video is increasing.

### 2.1.2 The Use Case modeling

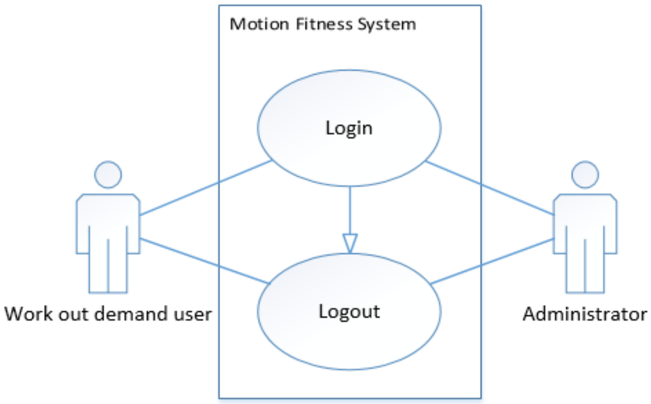


Figure 2-1 Use Case Diagram Of User's Management

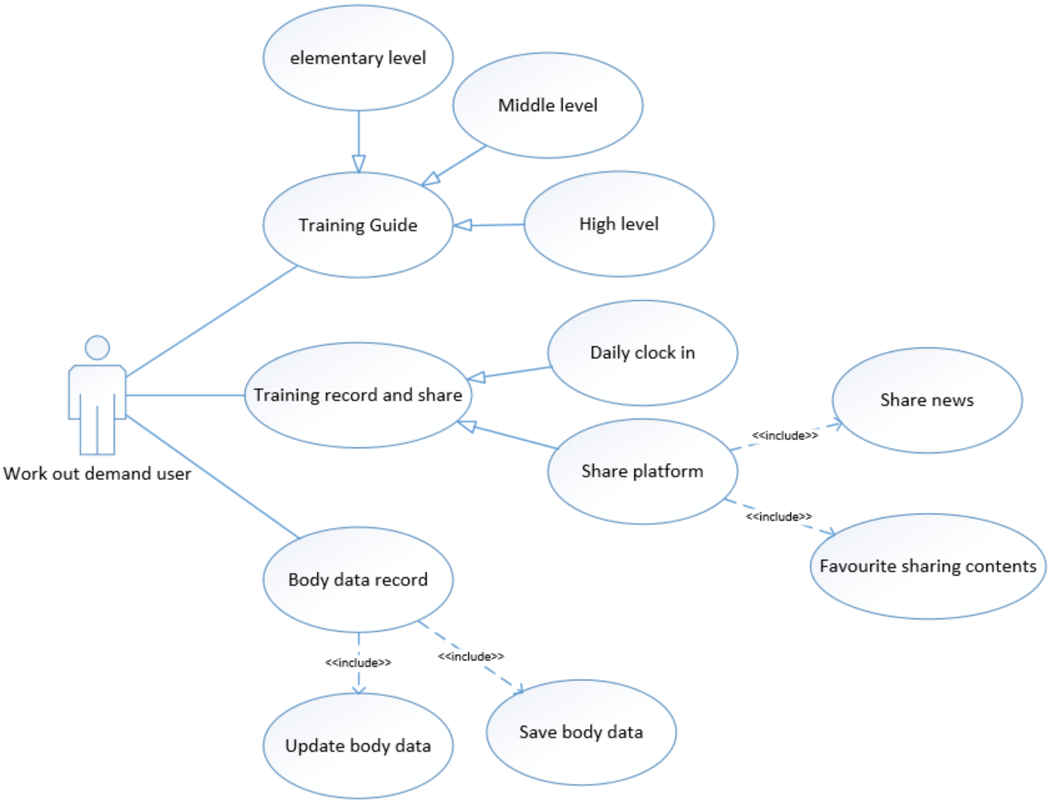


Figure 2-2 Use Case Diagram Of System

## 2.2 Development and Runtime Environment

Table 2-1 Development environment

| Client | Development Tool | Android Studio |
|---|---|---|
| | Android compiled version | Android 10 |
| | JAVA build tool | gradle:2.3.1 |
| | Android debugger | ADB 1.0.39 |
| | Data-transfer format | JSON |
| | Communication protocol | HTTP 1.1 |
| | JAVA Runtime Environment | JDK1.8 |
| Server | Development Tool | Eclipse |
| | Data-transfer format | JSON |
| | Database | MySQL8.0 |
| | Server | Tomcat8.5 |

# 3 Systematic Design

## 3.1 System Architecture Design

The design is a framework that follows the MVC pattern completely. In MVC mode, object hierarchy can be divided into three layers: view layer, control layer and model layer.

## 3.2 Class Design

### 3.2.1 Controller Class

(1) NewsFragment: Control the data interaction and functional logical of ‘News’ interface.
(2) TrainingFragment: Control the data interaction and functional logic of ‘Training’ interface.
(3) MeFragment: Control the data interaction of personal information and functional logical of ‘Me’ interface.

### 3.2.2 Service Class

(1) ‘RegisterActivity’ class & ‘LoginActivity’ class: Processing user registration and login.
(2) ‘HomePageActivity’ class: Process personal information and display function.
(3) ‘VideoPlayer’ class: Process the video player of ‘training’ function module.
(4) ‘CommentsListActivity’ class: Get user’s comments information in personal management module.
(5)‘DateCheckActivity’ class: Process time status of ‘daily clock in’ module and display the data of

‘daily clock in’ module.

(6) ‘NewsDetailActivity’ class: Process the comments data in the ‘news’ module.
(7) ‘ReleaseNewsActivity’ class: Process the text and pictures posted by user in the ‘news’ module.
(8) ‘FavorsListActivity’ class: Process the data of the ‘favourite’ function in the ‘news’ module.

## 3.3 Interaction Design

‘News’ function, ‘Comment’ function and ‘Daily clock in’ function sequence diagram shows in the following figure.

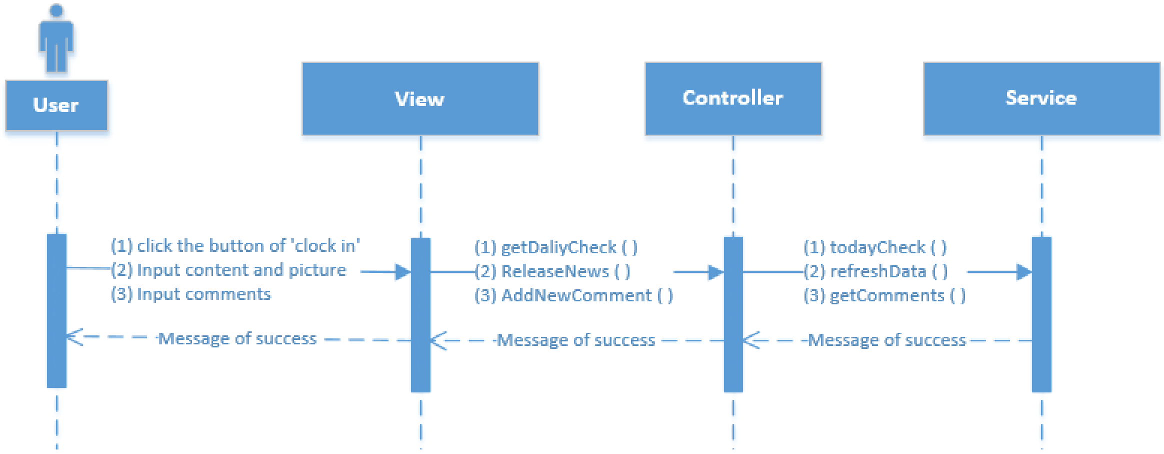
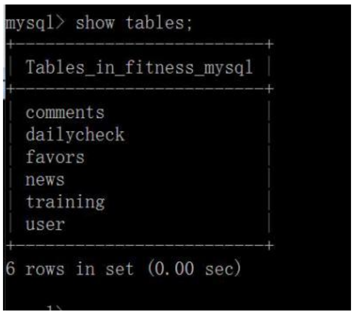


Figure 3-16 Sequence Diagram

# 4 System Implementation

## 4.1 Establish database and connection

First, create database ‘Fitness_mysql’ using MySQL on Tencent server.
Second, add the data table to database ‘Fitness_mysql’ and set the corresponding field. After the database is created successfully, the data table is shown in Figure 4-1.



```
mysql> show tables;
+-----------------------+
| Tables_in_fitness_mysql |
+-----------------------+
| comments              |
| dailycheck            |
| favors                |
| news                  |
| training              |
| user                  |
+-----------------------+
6 rows in set (0.00 sec)
```

Figure 4-1 Data table

## 4.2 Programming

### 4.2.1 Implementation of ‘Daily clock in’ function

The main code is as follows:

```
/**
  *   Today’s Check-in
  */
private void todayCheck() {
    uiFlusHandler.setTip("Clocking in...");
    uiFlusHandler.sendEmptyMessage(SHOW_LOADING_DIALOG);
    String url = Constants.BASE_URL + "DailyCheck?method=check";
    OkHttpUtils
            .post()
            .url(url)
            .id(1)
            .addParams("userId", Constants.USER.getUserId() + "")
            .build()
            .execute(new MyStringCallback());
}

public class MyStringCallback extends StringCallback {
        @Override
        public void onResponse(String response, int id) {
            uiFlusHandler.sendEmptyMessage(DISMISS_LOADING_DIALOG);
            switch (id) {
                case 1:
                    if (response.contains("success")) {
                        DisplayToast("Today clock-in success! ");
                    } else {
                        DisplayToast(response);
                    }
                    break;
                case 2:
                    if (response.contains("error")) {
                        DisplayToast("Data is temporarily unavailable");
                    } else {
                        String[] dates = response.split(",");
                        for (String s: dates) {
                        }
                    }
                    break;
        }
    }
```

### 4.2.2 Implementation of ‘News’ Function

The main code is as follows:

```
private void releaseNews() {
    String titleStr = title.getText().toString();
    String contentStr = content.getText().toString();

    uiFlusHandler.sendEmptyMessage(DISMISS_LOADING_DIALOG);
    String url;

    if (imageFile != null && imageFile.exists()) {
        url = Constants.BASE_URL + "News?method=releaseNewsWithImage";
        OkHttpUtils
                .post()
                .addFile("image", imageFile.getName(), imageFile)
                .url(url)
                .id(1)
                .addHeader("content-Type", "multipart/form-data; boundary=" +
UUID.randomUUID().toString())
                .addParams("title", titleStr)
                .addParams("content", contentStr)
                .addParams("userId", Constants.USER.getUserId() + "")
                .build()
                .execute(new MyStringCallback());
    } else {
        url = Constants.BASE_URL + "News?method=releaseNewsWithoutImage";
        OkHttpUtils
                .post()
                .url(url)
                .id(1)
                .addParams("title", titleStr)
                .addParams("content", contentStr)
                .addParams("userId", Constants.USER.getUserId() + "")
                .build()
                .execute(new MyStringCallback());
    }
}

public class MyStringCallback extends StringCallback {
    @Override
    public void onResponse(String response, int id) {
        uiFlusHandler.sendEmptyMessage(DISMISS_LOADING_DIALOG);
        Gson gson = new Gson();
        switch (id) {
            case 1:
                if (response.contains("success")) {
                    DisplayToast("Successfully Release News!");
                    finish();
                } else {
                    DisplayToast("Please try again later.");
                }
                break;
            default:
                DisplayToast("what?");
                break;
        }
    }
```

### 4.2.3 Implementation of ‘Comment’ function

The main code is as follows:

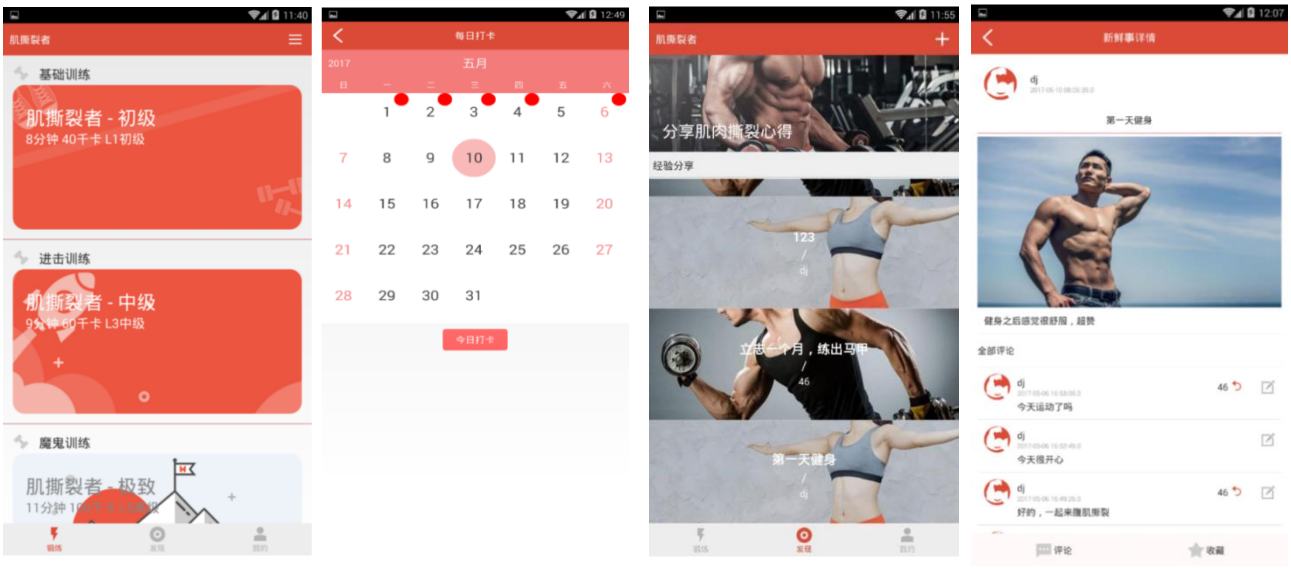
```
private void addNewComment() {
    String commentText = addCommentET.getText().toString().trim();
    if (TextUtils.isEmpty(commentText)) {
        DisplayToast("Please input comment.");
        return;
    }
    String url = Constants.BASE_URL + "Comment?method=addNewComment";
    OkHttpUtils
            .post()
            .url(url)
            .id(3)
            .addParams("newsId", newsId + "")
            .addParams("userId", Constants.USER.getUserId() + "")
            .addParams("comment", commentText)
            .addParams("replyUser", replyUsername)
            .build()
            .execute(new MyStringCallback());
}

public class MyStringCallback extends StringCallback {
    @Override
    public void onResponse(String response, int id) {
        switch (id) {
            case 1:
                uiFlusHandler.sendEmptyMessage(DISMISS_LOADING_DIALOG);
                Gson gson = new Gson();
                ...
                if (mList != null && mList.size() > 0) {
                    adapter = new NewsDetailCommnetsAdapter(mContext, mList);
                    adapter.setOnCommentButtonClickListner(new
NewsDetailCommnetsAdapter.OnCommentButtonClickListner() {

                        @Override
                        public void OnCommentButtonClicked(String replyUser) {
                            commentPane.setVisibility(View.VISIBLE);
                            addCommentET.setHint("Reply to" + replyUser);
                            replyUsername = replyUser;
                        }
                    });
                    commentsLV.setAdapter(adapter);
                }
                break;
                case 2:
                ...
}
```

### 4.2.4 Implementation of ‘Training’ function & video player

The main code is as follows:

```
private void loadVideo() {
    String uri = "android.resource://" + getPackageName() + "/";
    switch (tag) {
        case 1:
            duration = "8";
            uri += R.raw.base;
            break;
        case 2:
            duration = "9";
            uri += R.raw.enhance;
            break;
        case 3:
            duration = "11";
            uri += R.raw.acme;
            break;
    }
    videoView.setVideoURI(Uri.parse(uri));
    mediaController = new MediaController(this);
    videoView.setMediaController(mediaController);
    mediaController.setMediaPlayer(videoView);
    videoView.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {
        @Override
        public void onCompletion(MediaPlayer mp) {
            videoStop = true;
            saveTrainRecord();
        }
    });
    videoView.start();
}
```

Main UI design Shows in the follow:



# Bibliography

[1] Guo Lin. First Line of Code --- Android (edition 2). Beijing: Posts & Telecom Press, 2016.12.

[2] Guo Hongzhi. Android Application Development Details.Beijing: Publishing house of Electronics Industry, 2016.6.

[3] Reto Meier. Professional Android 4 Application Development [M]. Wrox.2012.

[4] Li Xinghua. Android Development Practice classic [M]. Beijing: Tsinghua University Press,2012.

[5] QiQi. Application and countermeasures of fitness APP in sports and fitness activities[J]. Journal of Xi 'an University of Arts and Sciences: Natural Science edition, 2020(5):99-104.

[6] Deng Fang. Research on the development of smartphone fitness APP in the field of sports and fitness [J].Journal of Longyan University,2019(3):100-105.