

**20CE 504L Programming Skills**

**Development-II Laboratory**

**(2022-23 Sem-I)**

**SYNOPSIS FORMAT FOR MINI-PROJECT**

**PSDL-II LAB SYNOPSIS**

1. **Group Member Roll No and C no. Name**

| **Name** | **Roll No.** | **C No.** |
| --- | --- | --- |
| Bhaktee Ugale | 3461 | C22020221461 |
| Yashvi Bhutada | 3472 | C22020221473 |
| Tejal Wakchaure | 3479 | C22021222405 |

1. **Problem Statement.**

Fitness Application

1. **Technology Selected :**

Technology Used - Android

Programming Language Used - Java

Database Used - SQLite

1. **Keywords :**
2. **Libraries Used Keywords:**

**i)** GIFLoading Libraries:

* Glide
* Dependency Used for Glide: Glide.*with*(this).load("<https://www.spotebi.com/wp-content/uploads/2014/10/push-up-exercise-illustration.gif>").into(imageView);

**ii) ImageLoading Libraries:**

* **CircularImageView :** To show Images, status, stories in android.

**iii) UI Component Libraries:**

* **LinearLayout** : To arrange and organize components in Linear order.
* **RelativeLayout** : To arrange and organize components in Relative order.
* **RecycleView** : To recycle components.

1. **UI Components Keywords Used:**

i) Imageview

ii) CardView

iii) Button

iv) TextView and others..

1. **Database Connectivity Keywords:**

i) SqliteDatabase

ii) SqlOpenHandler

ii) Adapter

iv) ViewHandler

v) and its methods…

1. **Abstract(150 words approx.)**

Through this app, we have tried to create a working fitness application. The app consists of different categories of workouts- for weight loss, body toning and muscle workouts. In these categories, we have created multiple workouts that work on timers. It also displays your profile, where your basic information is stored. Users can also set a reminder to get reminded of fitness activities they have to do. You can rate the app, and it has a privacy policy for users’ assurance. All the exercises and workouts are stored in the database. We can track our workouts to keep a track of history. This Application is developed for easy and effective workouts and exercises the user wants to do depending on their workout choices.

1. **Module wise Scope :**

**We have divided our Project into 4 modules:**

1. **FrontEnd Module :**

This Module includes all the Frontend Libraries, Activities we have done in our Fitness Application.

This module will then Interact with the Model Module to get access to the database.

---------- # Activities # ----------

**MainActivity** - used for displaying splash screen in which "Fitness App" is displayed

**HomeActivity** – used to display goals which include weight reduction,

**ReportActivity** – used to display the details of exercises, workouts done by user

**MeActivity** – used to display button for my profile, remainder, set rest time, set countdown time, rate us, privacy policy etc

**WorkoutHome** – after selecting any goal user will be directed to WorkoutHome which displays workouts included in selected goal through cardview

**ExerciseHome** – after choosing workout which user wishes to do he/she will be directed to ExerciseHome which will display exercises included in selected workout

**RestActivity** – after clicking on the start button from ExerciseHome the workout will begin. As exercises will be executed one by one loop has to applied to it, that is done in this activity

**ExecuteExercise** – through RestActivity for each exercise user will be directed towards this activity which will display gif, name and countdown of exercise

**ProfieActivity** – used to display user’s health profile. User can change the details, can view them in either kgcm or lbft.

**RemainderPage** – used to display remainders created by user through cardview.

**Remainder** – used to create remainder and add it to user’s database.

**RateUs** – used to take rating from user

**Feedback** – used to take feedback from user

**PrivacyPolicy** – used to display privacy policy of our app

**NotificationMessage** – used when notification of remainder is displayed

1. **Model Class:**

This module includes all the data and attributes frontend activities want to access and then this module will interact with another class to get DB access.

------------ # Model Classes # ------------

Model classes are used to represent data for example in case of exercise data ModelExercise is created which has attributes of exercise.

**ModelExercise.java** – used for exercise

**ModelWorkout.java** – used for workout

**ModelGoal.java** – used for goal

**ModelWorkoutExercise.java** – used to store data of WorkoutExercise table

**ModelGoalWorkout.java** – used to store data of GoalsWorkouts table

**ModelRemainder.java** – used for remainder

**ModelHealthDetails.java** – used for health details

**ModelDailyWorkoutDetails.java** – used for storing daily workout details

**ModelCountdownRestTime.java** – used for storing countdown and rest time details

1. **DatabaseHelper Class:**

This Module will interact with the Database and implement the Sqlite DB and include all the data the database wants to give access to. We created a Database , Tables its attributes inside this Module.

**DbHelper.java** – Used to store functions which connect to database and get data

1. **Database Module :**

This Module Contains the Actual Database Created for the Fitness Application.

----------- # Database # -----------

**Database Name** – FITNESS\_DB

**Database tool used** – SQLite

Since our application required some tables to created from our side and then upload it to user we did not create database in user’s device rather used already created database.

Tables and their details:

1. **Exercises**

Use – used to store details of each exercise

Columns :-

ID – primary key, autoincremented used for identifying exercise

Name - name of exercise

Time – amount of time for which exercise is to be executed

Calories – amount of calories burned after completing exercises

Info – how to do exercise

GIF – link of exercise gif

**2) Workouts**

Use – used to store name and details of each workout

Columns –

ID – primary key, autoincremented used for identifying workout

Name – name of workout

Info – information/slogan about workout

Image – image associated with workout which is to be displayed in cardview

**3) WorkoutExercises**

Use – used to store in each workout which exercises are included

Columns –

Workout\_id – foreign key, used for workout id which references ID from workouts table

Exercise\_id – foreign key, used for exercise id which references ID from exercise table

4) **Goals**

Used to store goals in our application

Columns –

ID – primary key, autoincremented used for identifying goal

Name – name of goal

Image – image associated with goal which is to be displayed in cardview

**5) GoalsWorkouts**

Use - used to store in each goal which workouts are included

Columns –

Goal\_id – foreign key, used for goal id which references ID from Goals table

Workout\_id - foreign key, used for workout id which references ID from workouts table

**6) HealthDetails**

Use – used to store health details of user

Columns –

Date\_of\_recording – used to store date of recording of health data

Height – used to store height of user

Weight – used to store weight of user

**7) DailyWorkoutDetails**

Use – used store details of each exercise performed by user

Columns –

Workout\_id – foreign key, used for workout id which references ID from workouts table

Exercise\_id – foreign key, used for exercise id which references ID from exercise table

Date – date on which exercise was performed

Time – time for which exercise was performed

Calories – calories burned after performing exercise

**8) Remainder**

Use – used to store remainders created by user

Columns –

ID - primary key, autoincremented used for identifying remainder

Name – name of remainder

Date – date of remainder

Time – time of remainder

**9) CountdownRestTime**

Use – used to store rest time and countdown time as per requirement of user

Columns –

ID - primary key, autoincremented used for identifying user

CountdownTime – used to store countdown time user wants

RestTime – used to store rest time user wants

----------- # Java Classes Used for Database # -----------

These classes are used to store table names, column names of database. These classes are created with purpose of loose coupling. If any column name or table name is changed then only changing attributes of these classes will be enough.

**DatabaseDetails.java** – Used to store details of database such as name of database, name of it’s tables, version number

**Exercises.java** – Used to store name of columns

**Workouts.java** - Used to store name of columns

**WorkoutsExercises.java** - Used to store name of columns

**Goals.java** - Used to store name of columns

**GoalsWorkouts.java** - Used to store name of columns

**HealthDetails.java** - Used to store name of columns

**DailyWorkoutDetails.java** - Used to store name of columns

**Remainder.java** - Used to store name of columns

**CountdownRestTime.java** - Used to store name of columns

1. **Technological features covered:**

* **Technology Libraries Used :**

1. **Libraries Used :**

**i)** GIFLoading Libraries:

* **Glide** : Glide not only provides animated GIF support while handling image loading and caching but also helps in fetching, decoding, displaying video calls, images, and these GIFs.

Dependency Used for Glide: Glide.*with*(this).load("<https://www.spotebi.com/wp-content/uploads/2014/10/push-up-exercise-illustration.gif>").into(imageView);

**ii) ImageLoading Libraries:**

* **CircularImageView :** To show Images, status, stories in android.

**iii) UI Component Libraries:**

* **LinearLayout** : To arrange and organize components in Linear order.
* **RelativeLayout** : To arrange and organize components in Relative order.
* **RecycleView** : To recycle components.

1. **UI Components Used:**

i) Imageview

ii) CardView

iii) Button

iv) TextView and others..

1. **Database Connectivity Technologies:**

i) SqliteDatabase:

SQLite supports all the relational database features. In order to access this database,

SqliteDatabase Database - Package Used:

The main package is android.database.sqlite that contains the classes to manage your own databases

SqliteDatabase Database Operations Done :

i) Database Insertion

ii) Database Deletion

ii) Database Updation

iv) Database View

1. Java File Technologies Used:

i) Intent : To pass from one activity to another

ii) View : To Multiple UI Components will be present in the operations and those UI components will be instances of View and ViewGroup SubClasses.

iii) AlertManager : The Alarm Manager holds a CPU wake lock as long as the alarm receiver's onReceive() method is executing. This guarantees that the phone will not sleep until you have finished handling the broadcast. Once onReceive() returns, the Alarm Manager releases this wake lock.

iv) NotificationManager: For receiving Notifications.

v) Graphics Primitives: For shaping the cardviews and creating graphics we used graphics primitive.

vi) and other Technologies required to develop fitness app.

1. **Conclusion**

We have Developed a Fitness Application by using Android Technologies.

1. **References**
2. [**https://developer.android.com/studio**](https://developer.android.com/studio)
3. **https://abhiandroid.com/**