ELECTRONIC ASSIGNMENT COVERSHEET

Murdoch UNIVERSITY

Team Number	Team 2	
member	Jing Ren, Tan	34309148
member	Hui Shi, Lim	34248791
member	Ka Jie, Au	34207159
member	Guangkuo, Tian	34162031

Unit Code	ICT376
Unit name	Mobile Application Development
Enrollment mode	Internal
Date	06/04/2022
Assignment number	A2
Assignment name	Assignment two
Tutor	Aaron Yeo Sze Wee

Student's Declaration:

- Except where indicated, the work I am submitting in this assignment is my own work and has not been submitted for assessment in another unit.

- This submission complies with Murdoch University's academic integrity commitments. I am aware that information about plagiarism and associated penalties can be found at http://www.murdoch.edu.au/teach/plagiarism/. If I have any doubts or queries about this, I am further aware that I can contact my Unit Coordinator prior to submitting the assignment.

- I acknowledge that the assessor of this assignment may, for the purpose of assessing this assignment:
  - Reproduce this assignment and provide a copy to another academic staff member; and/or
  - Submit a copy of this assignment to a plagiarism-checking service. This web-based service may retain a copy of this work for the sole purpose of subsequent plagiarism checking, but has a legal agreement with the University that it will not share or reproduce it in any form.
- I have retained a copy of this assignment.

I am aware that in making this declaration, by submitting this document electronically and by using my Murdoch ID and password, it is deemed equivalent to executing this declaration with my written signature.

Optional Comments to Tutor:

Please insert this completed form into the body of the assignment you submit. Follow the instructions in the Unit Guide about how to submit your file(s) and how to name them, so that the Unit Coordinator knows whose work it is.

A summary describing the project

The application for assignment two is called Photo Diary. The application allows users to upload photos with descriptions to the platform to record their life in a journal form. It helps us for collecting rich memories, not just by photo also by locations.

Design including the class diagram

UserModel
- id: int
- email: String
- name: String
- password: String
- gender: String
- dob: Date
- profilePhotoPath: String

+ getId(): int
+ setId(int id): woid
+ getName(): String
+ setName(String name): void
+ getPassword(): String
+ setPassword(String password): void
+ getProfilePhotoPath(): String
+ setProfilePhotoPath(String profilePhotoPath): void
+ getEmail(): String
+ setEmail(String email): void
+ getGender(): String
+ setGender(String gender): void
+ getDob(): Date
+ setDob(Date date): void

DiaryModel
- id: int
- title: String
- bdate: String
- time: String
- location: String
- description: String
- fileName: String
- imageUri: String
- userId: int

+ toString(): String
+ getFileName(): String
+ setFileName(String fileName): void
+ getId(): int
+ setId(): void
+ getTitle(): String
+ setTitle(): void
+ getDate(): String
+ setDate(): void
+ getTime(): String
+ setTime(): void
+ getLocation(): String
+ setLocation(): void
+ getDescripton(): String
+ setDescription(): void
+ getimageUri(): String
+ setImageUri(): void
+ getUserId(): void
+ setUserId(): void

MainActivity
+ etEmail: EditText;
+ etPassword: EditText

+ nCreate(Bundle savedInstanceState): void
+ login(View view): void
+ newUser(View view): void

NewUser
- REQUEST_IMAGE_CAPTURE: int
- REQUEST_GALLERY_IMAGE: int
- profilePhotoPath: String
- ivProfile: ImageView
- etEmail: EditText
- etName: EditText
- etDob: EditText
- etPassword: EditText
- etGender: EditText

onCreate(Bundle savedInstanceState): void
+ createUser(View view): void
# onActivityResult(int requestCode...): void

EditProfile
+ USER_ID: String
- REQUEST_IMAGE_CAPTURE: int
- REQUEST_GALLERY_IMAGE: int
- ivProfile: ImageView
- btnSave: Button
- etEmail: EditText
- etName: EditText
- etPassword: EditText
- etGender: EditText
- etDob: EditText
- imageFragment: ImageListDialogFragment
- userId: int
- db: DatabaseHelper

onCreate(Bundle savedInstanceState): void
- populateProfile(): void
+ saveUser(): void
# onActivityResult(int requestCode,...): void
- saveImage(String filename, Bitmap bitmapImage): String

UserLanding
- recyclerView: RecyclerView
- databaseHelper: DatabaseHelper
- userId: int

onCreate(Bundle savedInstanceState): void
onResume(): void
- showUserDiaries(): void

DatePickerFragment
+ onCreateDialog(Bundle savedInstanceState): Dialog
+ onDateSet(DatePicker view, int year, int month, int day): void

DiariesAdapter
- diaryModelList: List<DiaryModel>

+ onCreateViewHolder(...): DiariesAdaper.MyViewHolder
+ onBindViewHolder(...): void
+ getItemCount(): int

ImageListDialogFragment
- binding: FragmentItemListDialogListDialogBinding

+ newInstance(int itemCount): ImageListDialogFragment
+ onCreateView(...): View
+ onViewCreate(...): void
+ onDestroyView(): void
+ onBindViewHolder(....): void
+ getItemCount(): int

TimePickerFragment
+ onCreateDialog(Bundle savedInstanceState): Dialog
+ onTimeSet(TimePicker view, int hourOfDay, int minute): void

DatabaseHelper
- USER_TABLE: string
- USER_ID: string
- USER_EMAIL: string
- USER_NAME: string
- USER_PASSWORD: string
- USER_GENDER: string
- USER_DOB: string
- USER_PROFILE_PHOTO_PATH: string
- DIARY_TABLE: string
- DIARY_ID: string
- DIARY_TITLE: string
- DIARY_DATE: string
- DIARY_TIME: string
- DIARY_LOCATION: string
- DIARY_DESCRIPTION: string
- DIARY_FILE_NAME: string
- DIARY_IMAGE_URI: string

+ onCreate(SQLiteDatabase db): void
+ onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion): void
+ checkIfUserExists(String email): boolean
+ addUser(String email ......): boolean
+ getUserById(int id): UserModel
+ getUserByEmail(String email): UserModel
+ getAllUserDiaries(int userID): List<DiaryModel>
+ addDiary(DiaryModel diaryModel): boolean

NewEntry
locationManager: LocationManager
+ imageFragment: ImageListDialogFragment
+ titleView: TextView
+ dateView: TextView
+ timeView: TextView
+ locationView: TextView
+ descriptionView: TextView
+ saveButton: Button
+ imageView: ImageView
+ datePickerFragment: DialogFragment
+ timePickerFragment: DialogFragment
+ userId: int

onCreate(Bundle savedInstanceState): void
+ imageSource(View view): void
+ pickDate(View view): void
+ pickTime(View view): void
# onActivityResult(int requestCode, ...): void
+ onLocationChanged(Location location): void
+ onProviderEnabled(String provider): void
+ onProviderDisabled(String provider): void
+ onStatusChanged(String provider, int status, Bundle extras): void
+ setLocation(): void
+ setDate(LocalDate localDate): void
+ setTime(LocalTime localTime): void
+ saveImage(String filename, Bitmap bitmapImage): String
+ save(View view): void
+ onOptionsItemSelected(MenuItem item): boolean

EditDiaryActivity
- diaryTitle: EditText
- dianyDesc: EditText
- title: String
- date: String
- time: String
- desc: String
- path: String
- imageName: String
- id: int
- databaseHelper: DatabaseHelper

onCreate(Bundle savedInstanceState): void
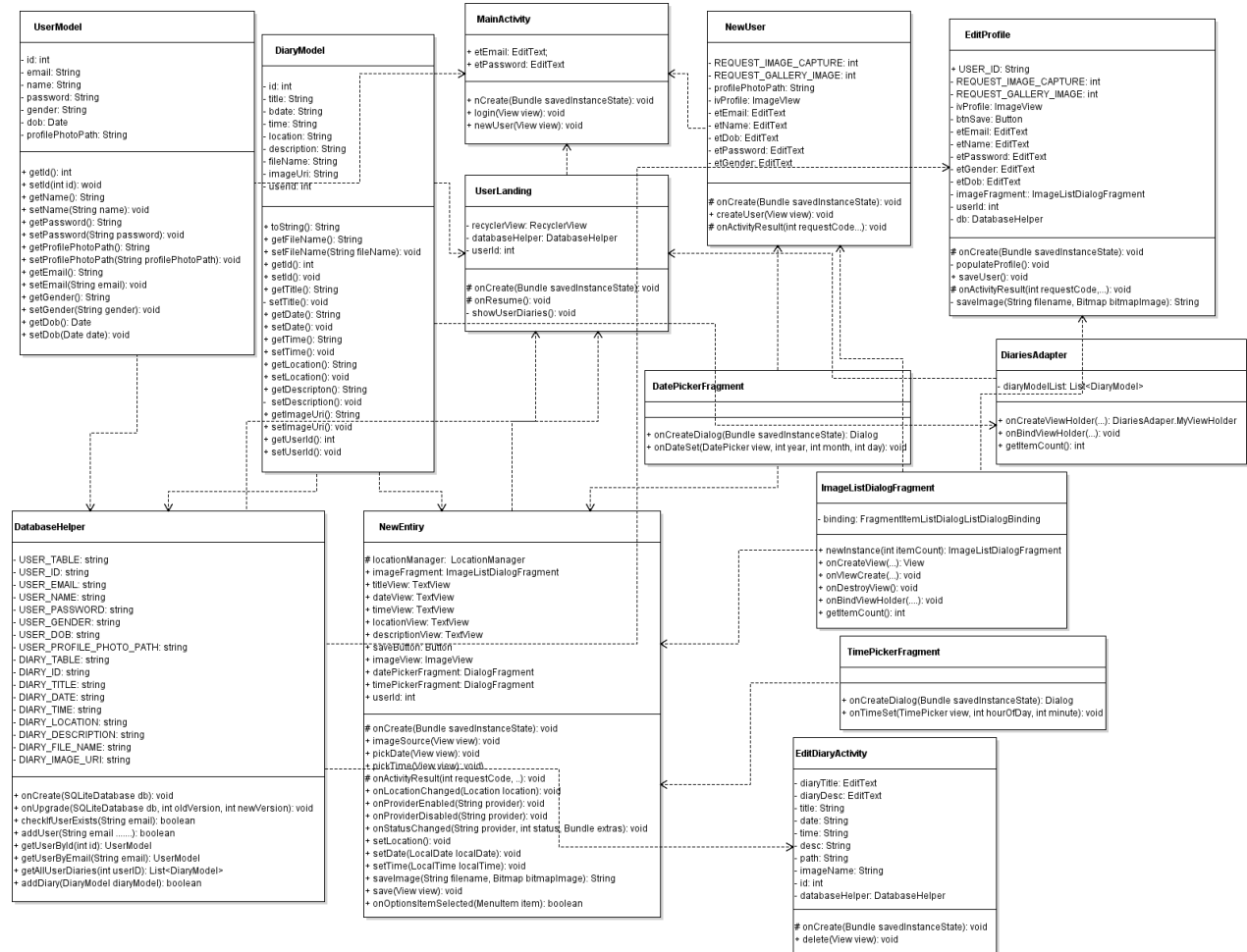+ delete(View view): void

Figure 1 Photo Diary Class diagram

As Figure 1 described when user open the application it shows login page, firstly application will trigger the MainActivity.class. At login page user can register new account and login by their credentials also we provide test account for those users who want to try without signing up.

If user use sign-up page to register a new account, user need to provide their email, name, password, gender and DOB to sign up. This function will use NewUser.class.
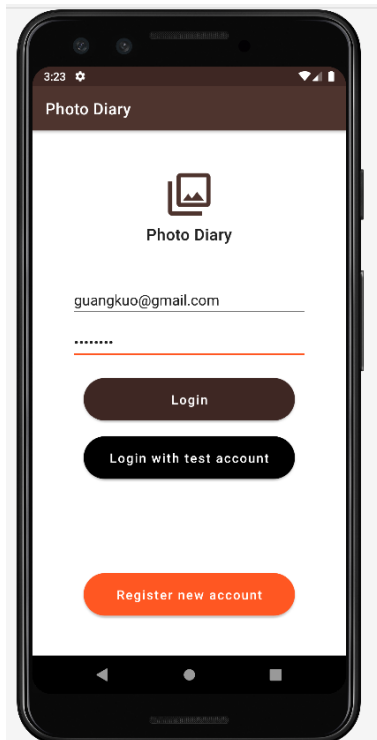
After user signed up, user can sign in. Then user can view diary lists, click one item to see the details and update or delete one diary. Also in this page user can click add button to add new photo diary.

After user click add button, will go into new Entity page, user can select camera to take a new photo or browser a photo shop to select a photo. At the same time user can add a title, timestamp for the diary if user give this app location services right, our app will record the location of the photo took.

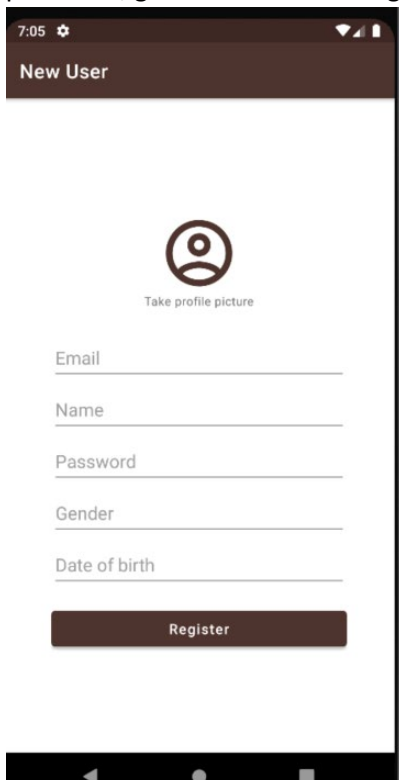# Instructions on how to use the software

1) Login page,

   At login page user can register new account or login by their credentials, also we provide a test account for those user want to try without signing up.
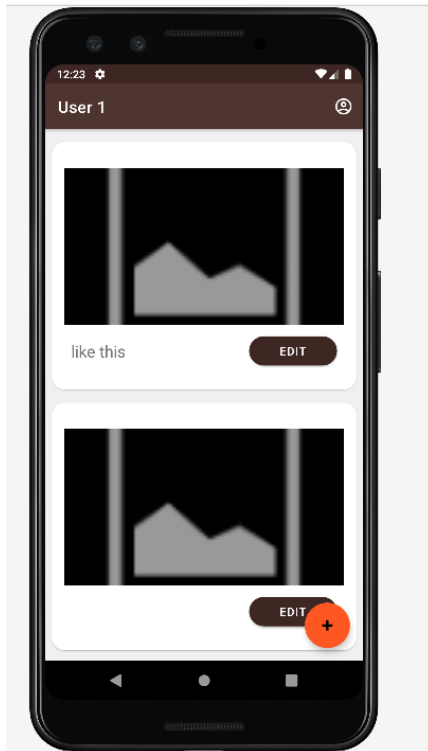

2) New User page

   If user use sign-up page to register a new account, user need to provide their email, name, password, gender and DOB to sign up.
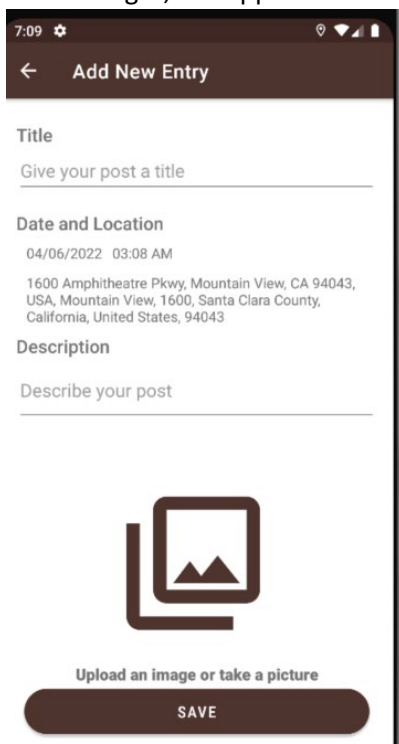
3) User landing page
   After use login in, user can view diary lists, click one item to see the details and update or delete one diary. Also in this page user can click add button to add new photo diary.

4) Add new entity page
   user can select camera to take a new photo or browser a photo shop to select a photo. At the same time user can add a title, timestamp for the diary if user give this app location services right, our app will record the location of the photo took.

5) Edit photo diary page

6) Edit user profile

Sample input-output screenshots to demonstrate all major operations asked

1) Login with test account
   a. Open this application
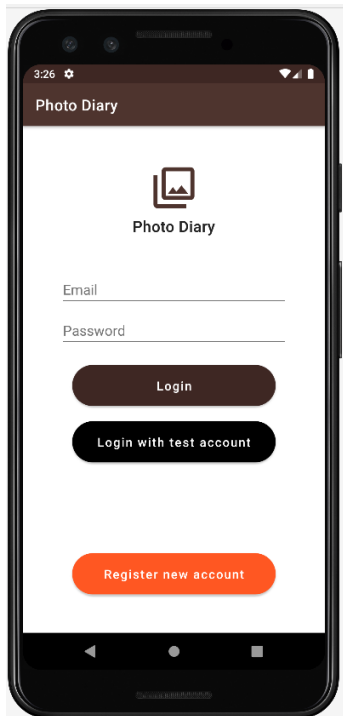
   b. Then click "Login with test account" button, test is ok
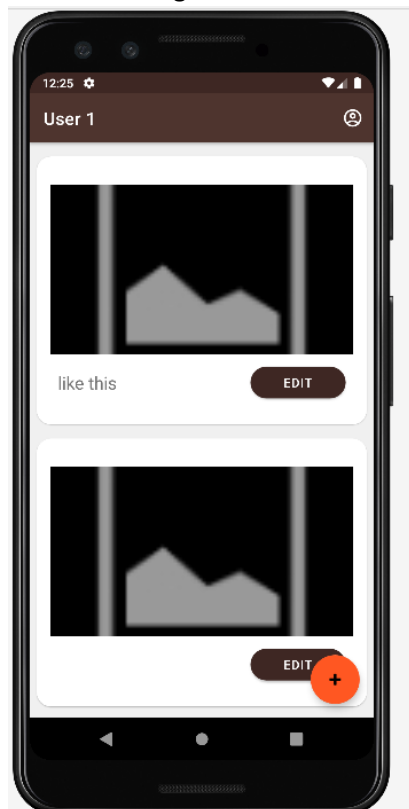
2) Register new account
   a. Click register new account
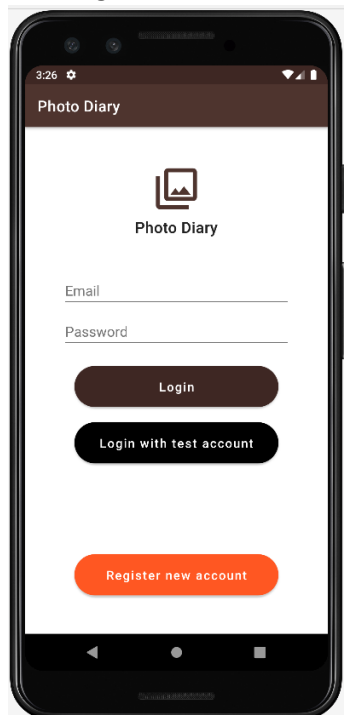
   b. Input user information and click register
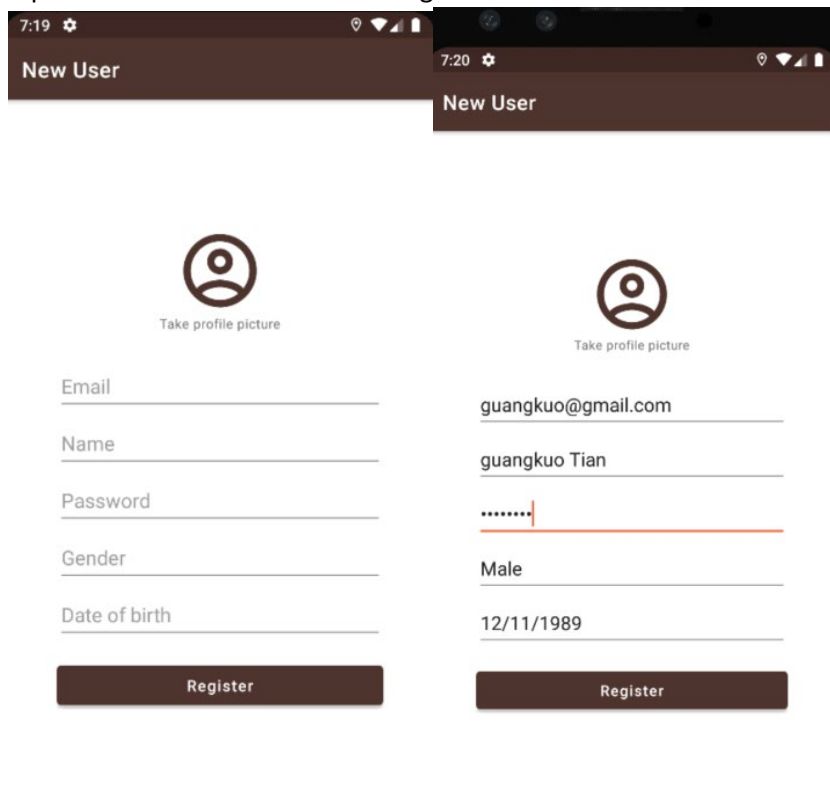
      It will show register successful

3) Login with user account
   a) Login with last step username and password
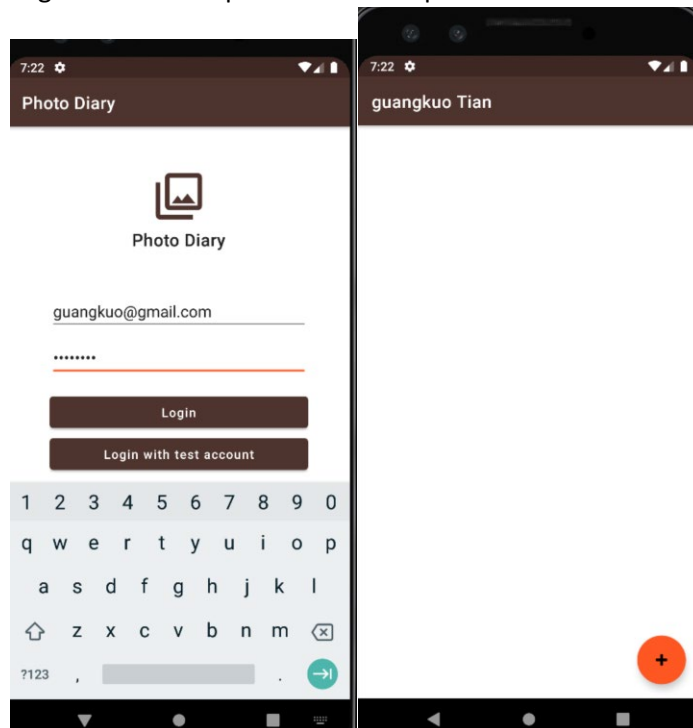
   Test is ok, login successfully.
4) Add new diary
   a. Continue from last test, click add button
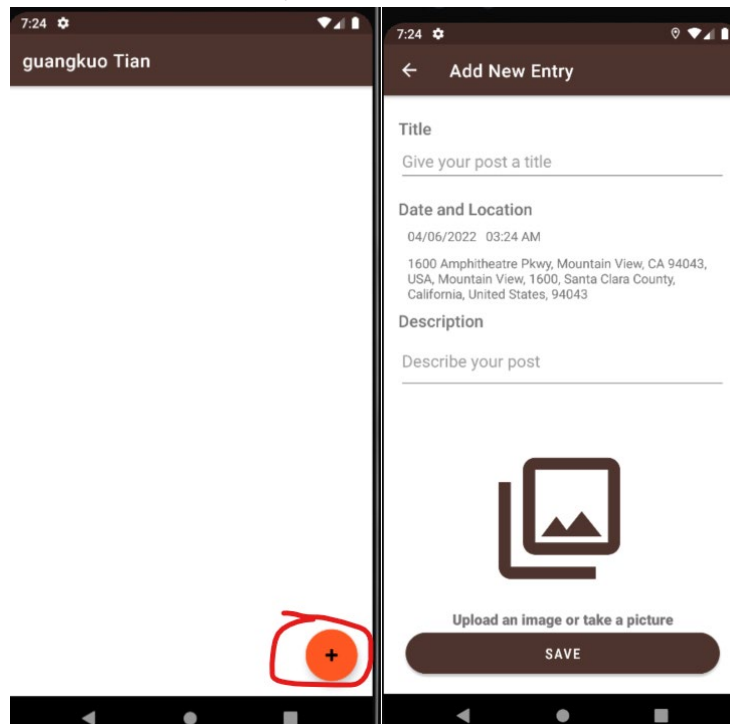
b. Title can be edited, date can be selected
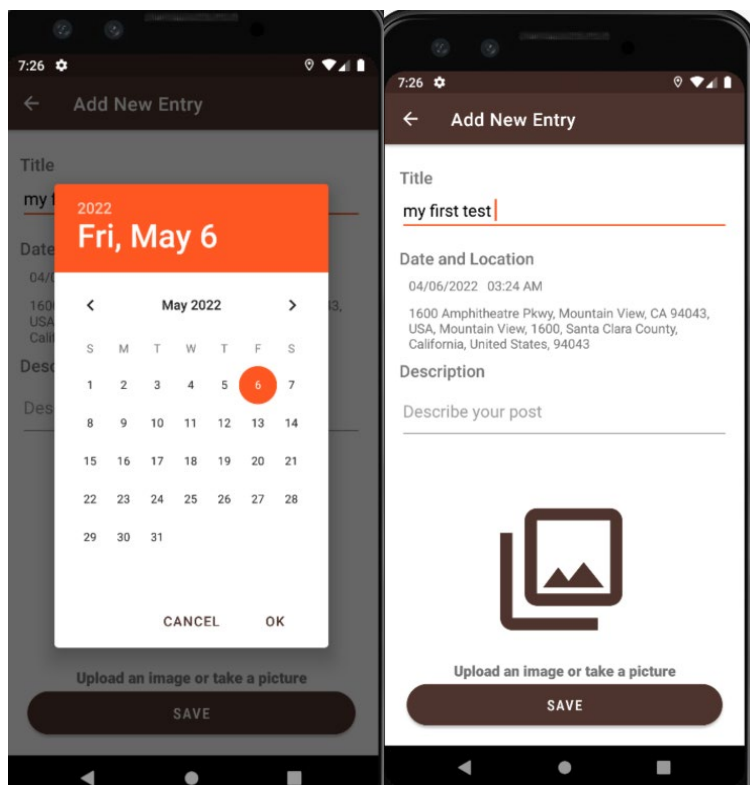
c. Description can be edited and then choose browser
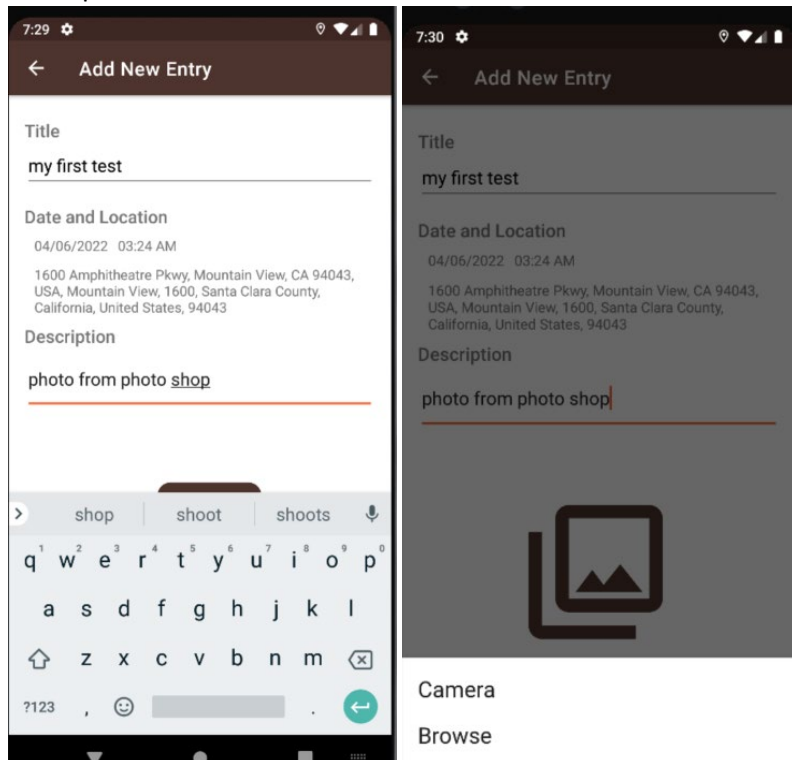
d. Select one photo
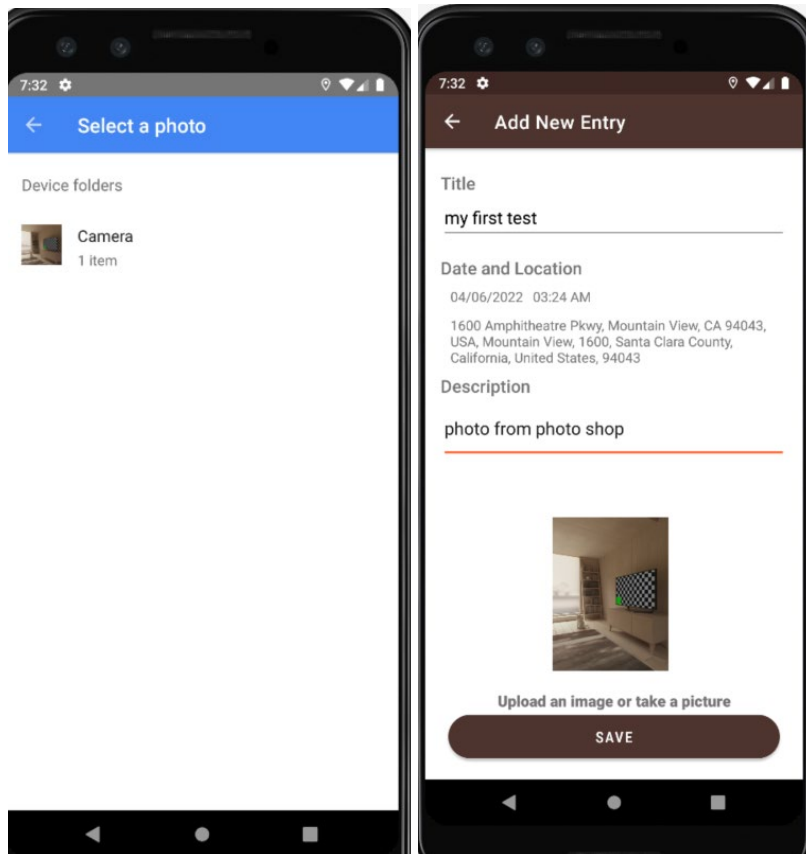
e. Click save button to save diary, and diray is saved successfully.
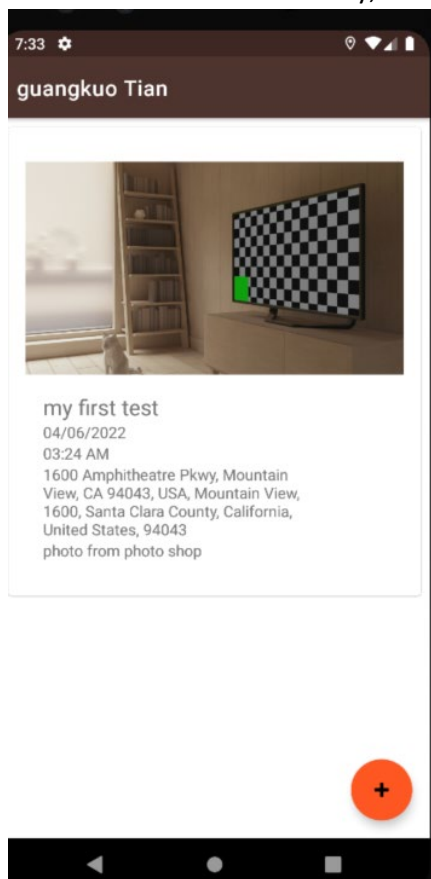
5) Edit diary with some changes
   a. Login with account
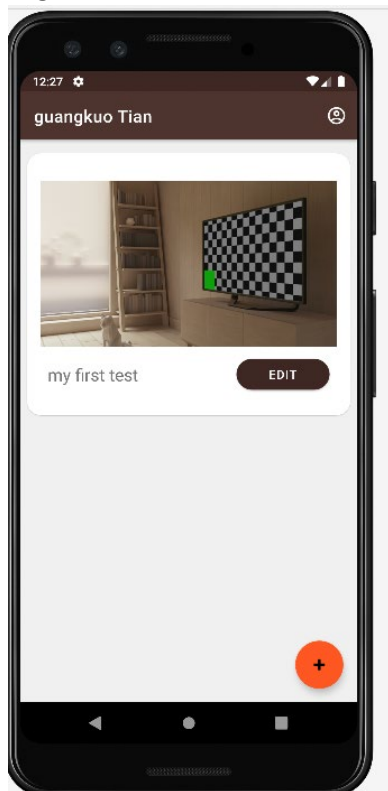
   b. Click Edit button and change title name
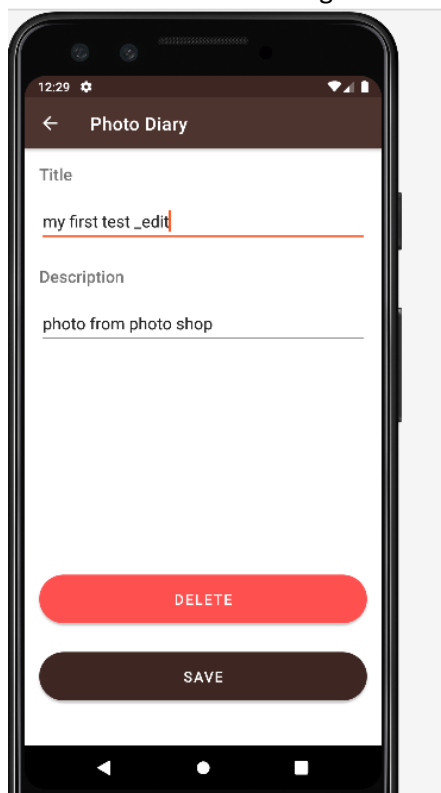
c. Click save button, we can see diary name has been changed.
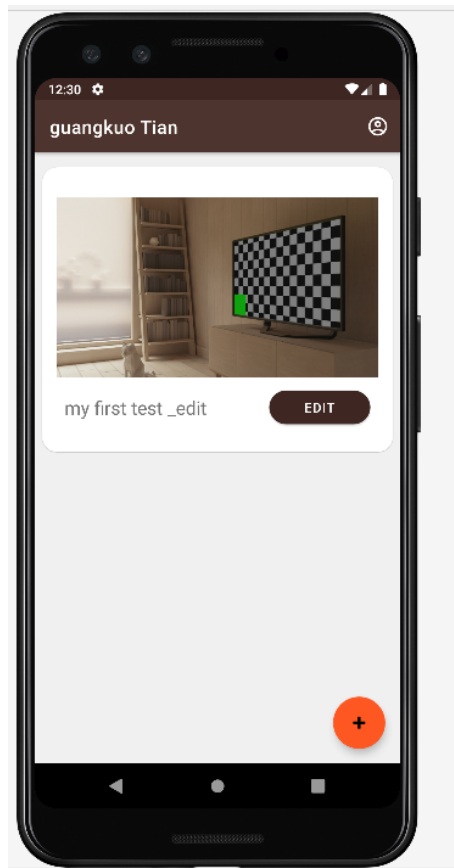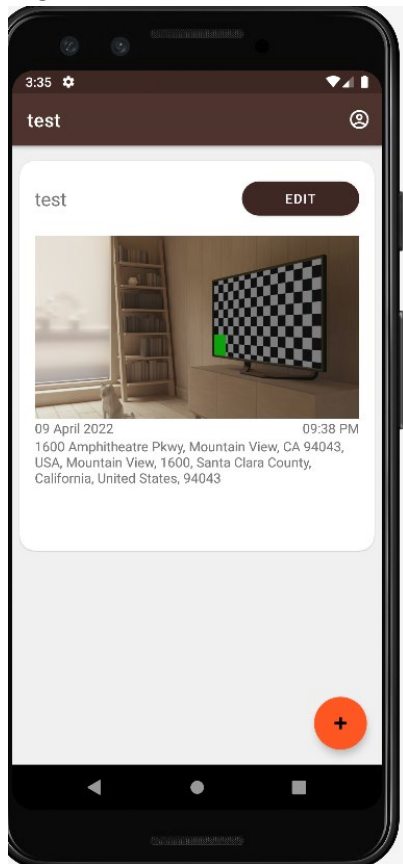

6) Delete a diary
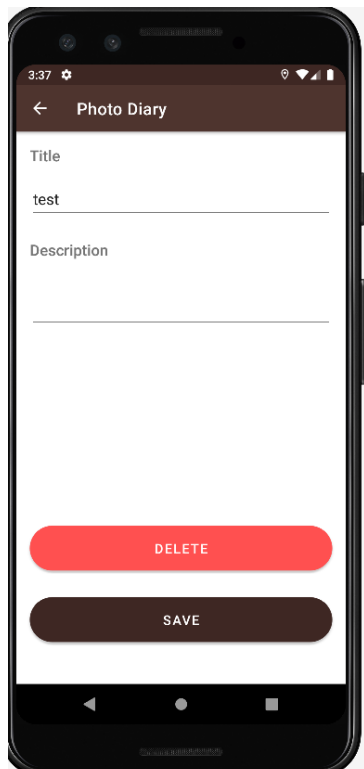   a. Login with account
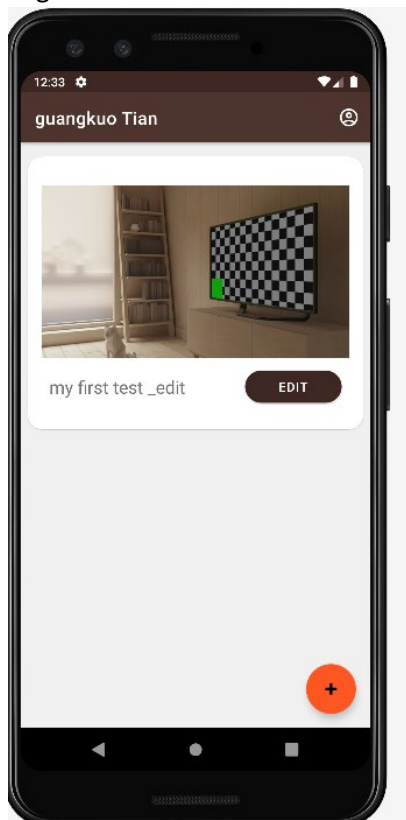
b. Click the edit button
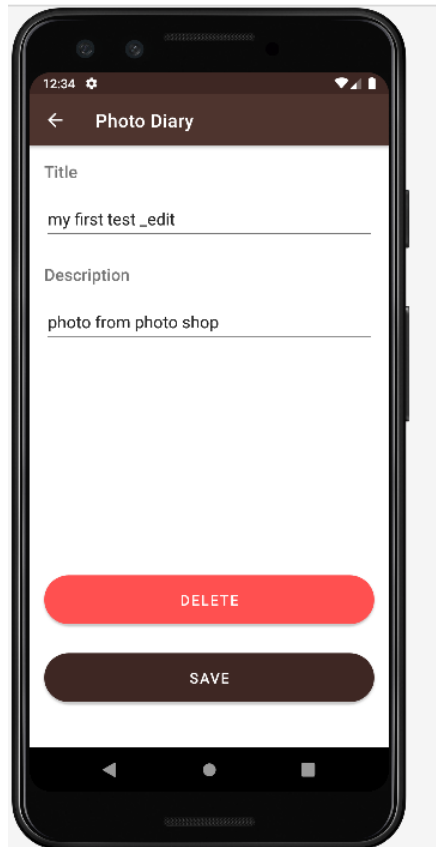
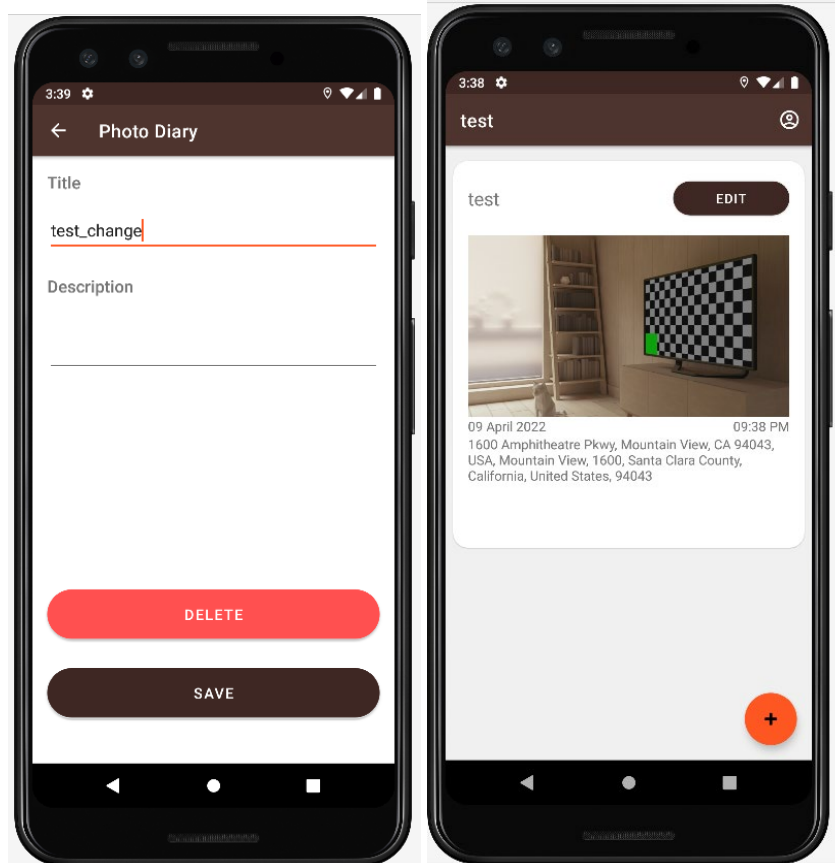
c. Click delete button, we can see diary has been deleted.

7) Edit diary without changing
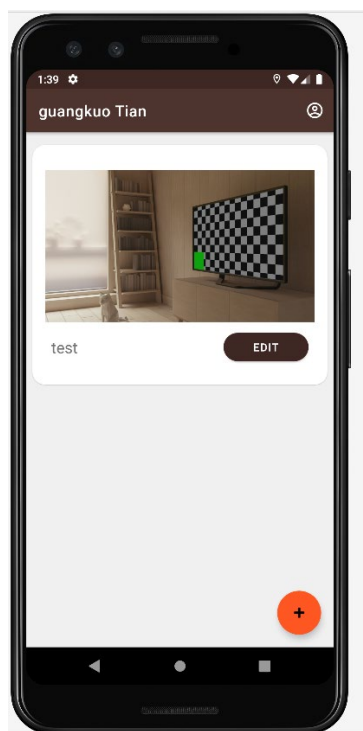   a. Login with account

   b. Click the edit button

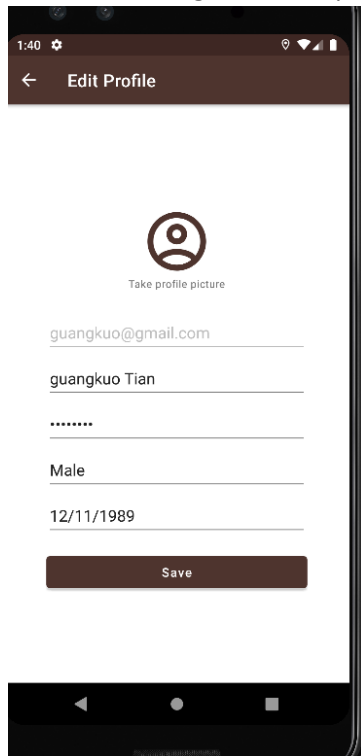c. Edit some changes but Click back button without press save button

If we don't click save button, nothing will be changed.

8) Edit user profile
    a. Login in with user profile

b. Click user icon, go into edit profile page

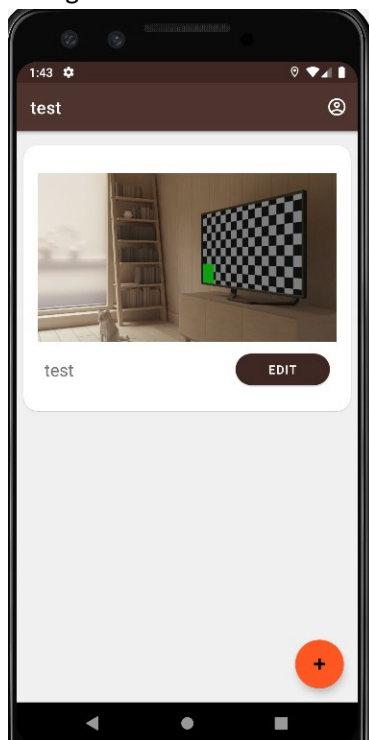c. After change username to test, click save. We can see picture below, username has changed.

Table 1: Testcase for Photo Diary

Test Description	Expected output	Outcome – Pass?
Login Page		
Login with test account	Able to login through test account button	Yes
Register new account in new page	Route to register page	Yes
Login with registered account	Able to login with registered	Yes
Register page		
Set profile picture with camera capture	Able to set profile picture with camera capture	Yes
Set profile picture with uploading an image locally	Able to set profile picture with uploading an image locally	Yes
Check if email is using the right format	Prompt error message if email does not follow the right format	Yes
Check if name is meet minimum requirement	Prompt error message if name does not meet minimum requirement	Partially, require minimum characters
Check if password meet minimum requirement	Prompt error message if password does not meet minimum requirement	Partially, require minimum characters
Check if data are stored at local database	User account is created and credentials are stored in local database	Yes
Home page		
Diary entries displaying	Diary entries displaying in card format	Yes
Entries information showing	Entries information such as Title, Description, Image, Location, Date and time is displaying properly	Yes
New entry button	Route to New Entry Page	Yes
Edit button	Route to Edit Entry Page	Yes
User profile button	Route to Edit User Profile Page	Yes
New Entry page		
Title and description Input	Able to store a title and description with entry locally	Yes
Date and time Input	Able to input date and time and stored locally	Yes
Location Input	Able to query user's current location and stored with entry locally	Yes
Save entry	Able to save the entry with all input information in local database	Yes

Check if title and description meet minimum requirement	Prompt error message if title and description does not meet minimum requirement	Partially, require minimum characters
Check if image is uploded	Prompt error message if no image is uploaded	No
Edit Diary		
Edit Title	Able to modify the title input	Yes
Edit Description	Able to modify the title input	Yes
Delete entry	Able to delete the diary entry	Yes
Save edited entry	Able to update and store the modified diary entry in local database	Yes
User Profile		
Profile image change	Able to update and store the new profile image through upload or camera capture	Yes
Update password	Able to update a new password for the user account	Yes
Update name	Able to update a new name for the user account	Yes
Update gender	Able to update a new gender for the user account	Yes
Update birthdate	Able to update a new birthdate for the user account	Yes