Functional and Technical Specifications Document

Project Name: StuPlan

Group Number: 7

Group Members:

1. **Krishna Parmar (200504981)**
2. **Sajan Tamang (200507132)**

Functional Requirements

Stuplan is a planner application. The purpose of the project is to develop an application in which users can view and add their deadlines for assignments, homework, and test. This app will help students to track their upcoming assignments and test according to their subject so that they can plan their studies well.

1. **Splash Screen:**
   Initially, splash screen will be loaded which will include the logo of the application.

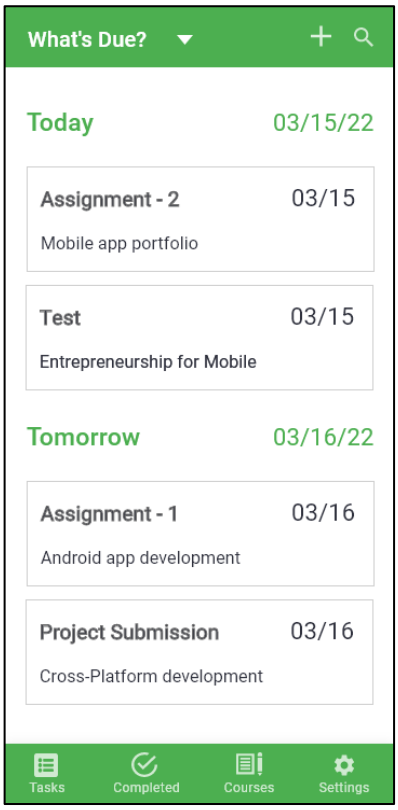
2. **Home Screen:**
   In this screen, User will be able to see what is left to do for today or upcoming days.

   Using + button, user can add new task which are needed to be done in particular subject.

   button will allow user to search list in home screen.

   When users complete the task, they can swipe left the task in order to mark as a completed.

   All the task which are mark as completed by swiping left will be added into list of completed tasks.



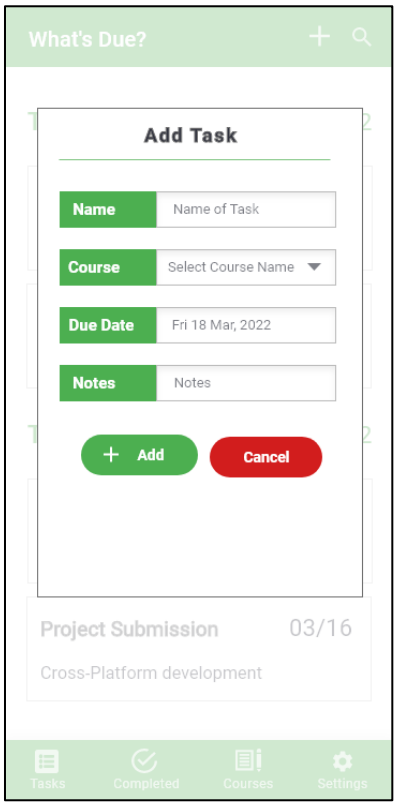
3. Add Task Screen:

In '*Home screen'* when user clicks  user will be redirect to 'Add courses screen'.

In this screen, user can add new tasks by simply entering name Of task, Courses name due dates and additional notes.

Afterwards, User can click on 

to add the task to task List which user will be able to see in 'home screen.'

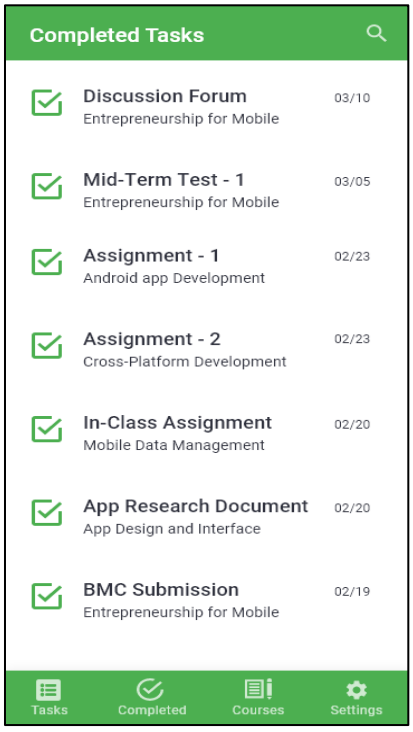


4. Completed Tasks Screen:

In bottom navigation, when user click on 

button, user will be redirected to completed task screen.

Completed tasks screen contains list of tasks which are already completed.
Search button will allow user to search the list of completed task.

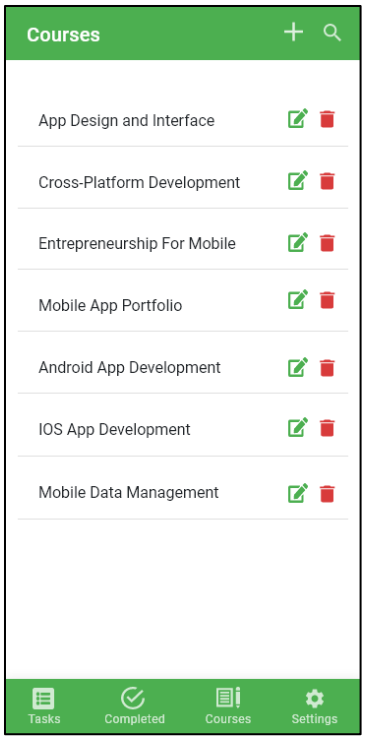


5. **Courses Screen:**

In bottom navigation, when user clicks 

button user will be redirect to courses screen.

This screen contains list of courses that are added by the user.



Using + button, user will be redirect to 'Add courses Screen' Which will be used to add courses in the application.

this button allows user to edit the course name.

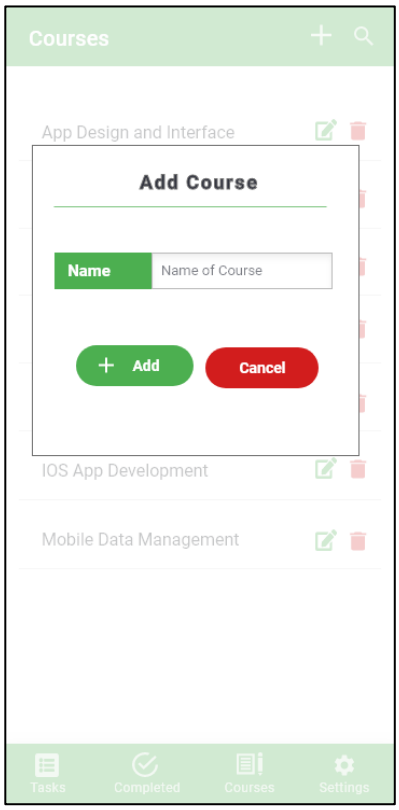
this button allows user to delete the course name.

this button allows user to search the list of courses name.

6. **Add Courses:**

In Courses screen, when user clicks + user will be redirect to 'Add courses screen'.

In this screen, User can add courses by simply entering the course Name and clicking + Add then courses will be added in the 'Courses screen'



7. Settings Screen:

In bottom navigation, when user clicks on 

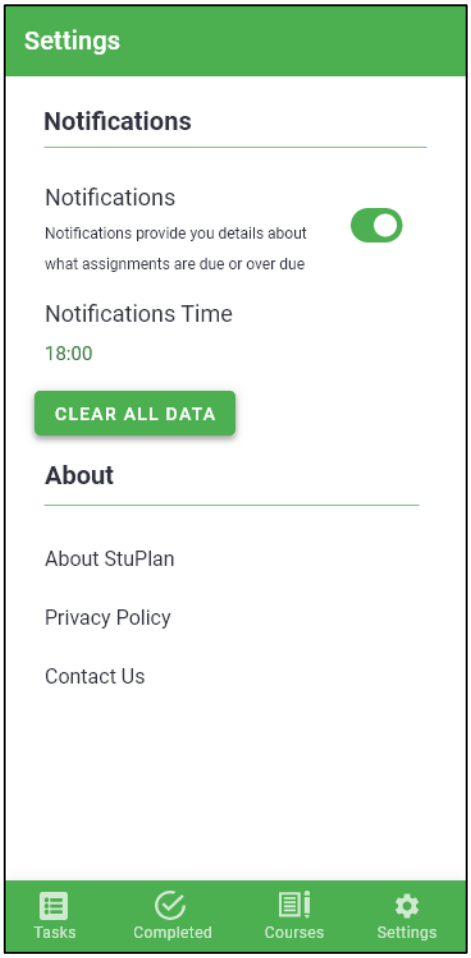
menu,

User will be redirect to ‘Setting Screen’.

This screen allows users to enable or disable notification alert And allow to change the time of notification.

button will wipe out the all the due tasks and Completed task.

About section contains the information about StuPlan app, Privacy policy and contact information.



Technical Specifications

1. **Home Screen:**
   Model class: Task
   Variables for Task class:
   - taskId which is an integer
   - taskName which is a string,
   - course which is another class,
   - dueDate which is date or string
   - note which is String.

   Functions:
   fetchTasks() -> this function queries and returns all the incomplete tasks from local database.
   updateTaskAsComplete() -> this function has queries to update the task as complete when user swipe in the list

   The data are fetched from persistent local database i.e. sqlite. In the list there is swipe feature to update the data in the database to mark it as completed. The Boolean variable in the model class denotes if the task is completed or not.

2. **Courses Screen:**
   Model class: Course
   Variables: courseName is a string
   courseId is an integer
   Functions:
   updateCourse() -> to make necessary edit in name of the course and update in local database
   deleteCourse() -> to delete the course from the list as well as local database
   The data are fetched from the local database and contains functions to fire the queries for create, update, or delete the course.

3. **Add course screen:** This screen makes use of the Course model class.
   Functions: insertCourse() -> to add the course in the local database and display in the list in course screen.

4. **Add task screen:** This screen makes use of Task class as a model class.
   Functions: insertTask() -> to insert the task in local database from the data obtained in the UI.

5. **Completed task screen:** This uses Task class.
   Functions:
   fetchCompletedTask() -> fetch all the tasks that have a completed flag from the local database to display as a list.

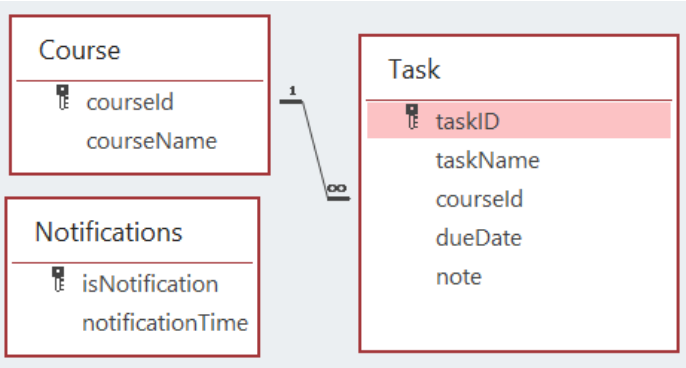
6. **Settings screen:** It has a button to clear the data that will reset all the database and other data about the user in the app. Furthermore, It has option to send daily notification in which user can turn it off/on and set the time to send notification
   Functions: setTimer() -> to set timing of notification
   deleteData() -> to delete all the data from local database

Database:

Table Name: Course
courseId (Integer) (Primary Key) (Auto_Increment)
courseName (Text)

Table Name: Notification
isNotification (Boolean)
notificationTime (Text)

Table Name: Task
taskId(Integer) (Primary Key) (Auto_Increment)
courseId (Integer) (Foreign Key)
taskName(Text)
dueDate (Text)
Note (Text)



Course table will be used to store details of course name.

Task table will be used to store details of tasks and it will use courseId as a foreign key to get the details of courses.

Notification table will be used to store the details of whether notification is enabled or disabled as well as the time of notification to be send.