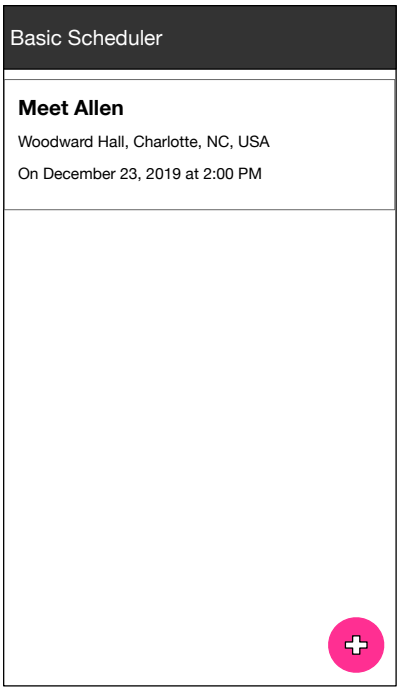
ITIS/ITCS 5180 Mobile Application Development
Final Exam

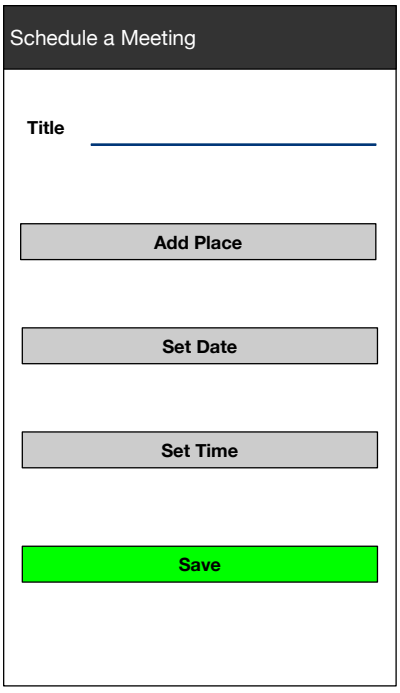
Basic Instructions:

1. This is the Final Exam, which will count for 20% of the total course grade.
2. This Final is an individual effort. Each student is responsible for her/his own Final and its submission.
3. Once you have picked up the exam, you may not discuss it in any way with anyone until the exam period is over.
4. During the exam, you are allowed to use the course videos, slides, and your code from previous home works and in class assignments. You can use the internet to search for answers. You are NOT allowed to use code from other students or solicit help from other online persons.
5. Answer all the exam parts, all the parts are required.
6. Please download the support files provided with the Final and use them when implementing your project.
7. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will loose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
8. Export your Android project and create a zip file which includes all the project folder and any required libraries. The file name is very important and should follow the following format: **800#_Final.zip.** Submit the exported file using the provided canvas submission link.
9. **Do not try to use any Social Messenger apps, Emails, Or Cloud File Storage services in this exam, except for your own materials.**
10. **The required Android Virtual Device (AVD) should have minimum SDK version set to 26 and target SDK at 28+.**
11. **Failure to follow the above instructions will result in point deductions.**
12. **Any violation of the rules regarding consultation with others will not be tolerated and will result disciplinary action and failing the course.**

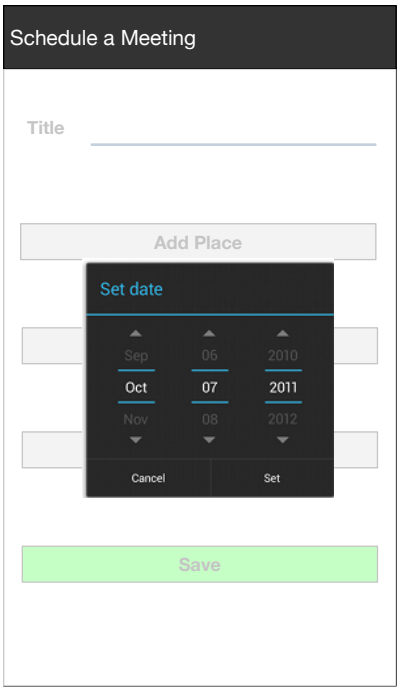
Final Exam (100 points)



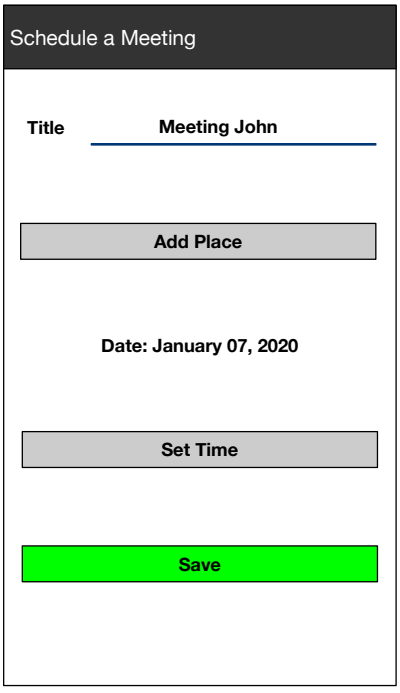
(a) Home screen



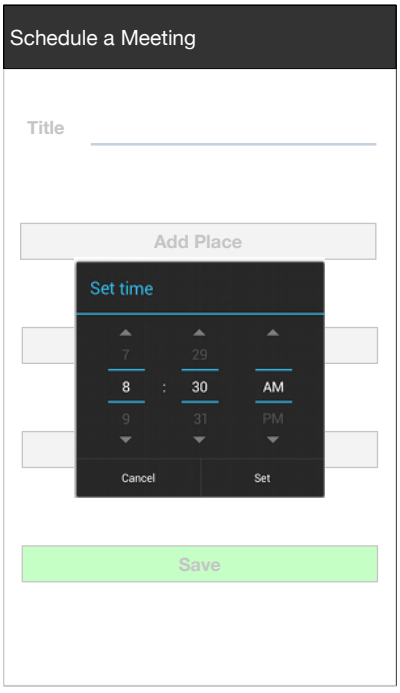
(b) Add button clicked



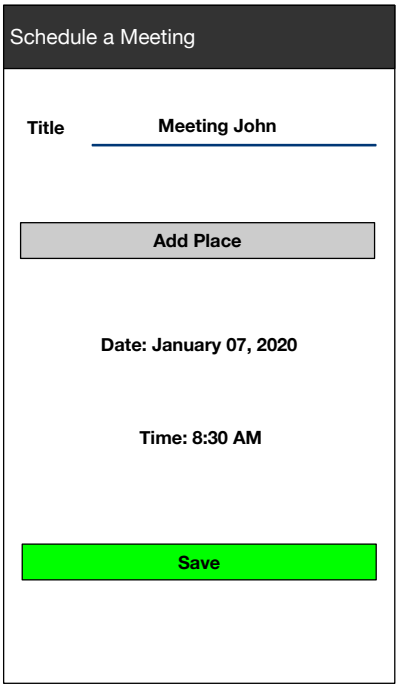
(c) Date picker to pick date



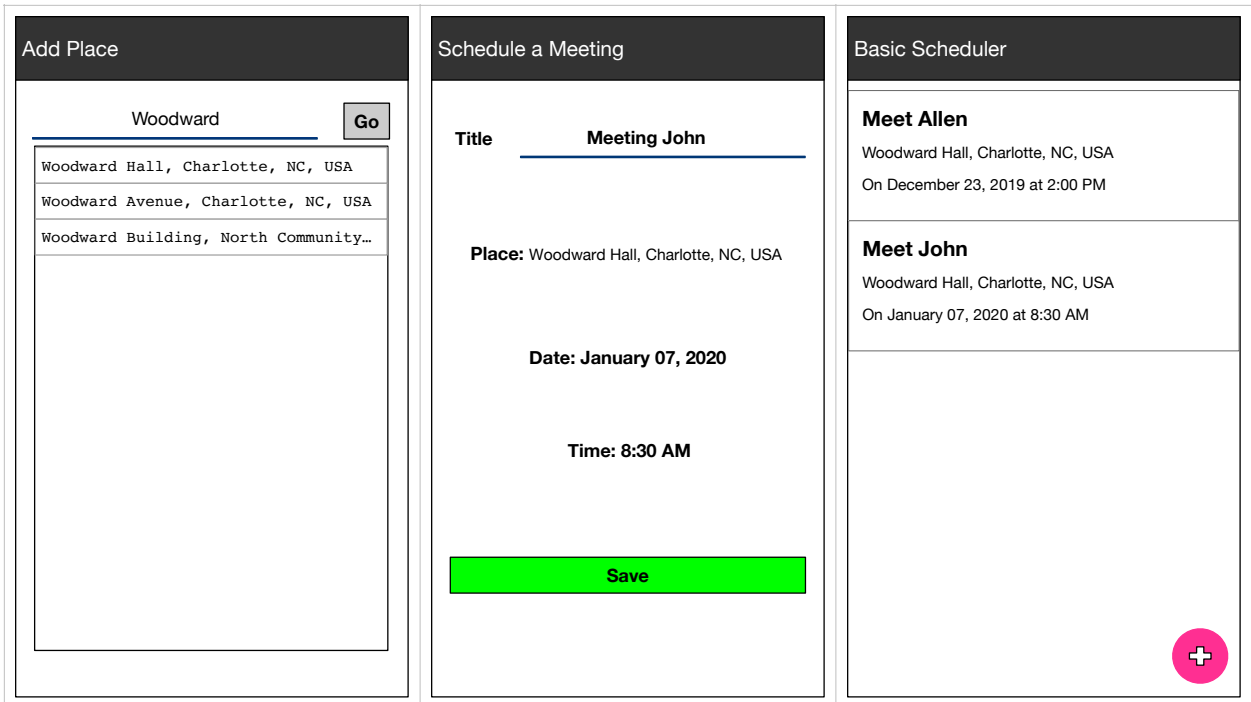
(d) Date is set



(e) Time picker to pick time



(f) Time is set



(g) Add Place screen	(h) Place is added	(i) Schedule added, sorted by earlier to later

Figure 1: App screens

In this assignment, you need to build a basic meeting scheduler application. Please read the requirements very carefully:

Basic instructions:

1. For screens, you can use either Activities or Fragments.
2. You must save the data in Firebase Firestore or Realtime Database.
3. You must use RecyclerView. There will be a 12 points penalty if you do not use RecyclerView.
4. You have to use Google places autocomplete API to search the places.
5. You have to use Date and Time pickers to pick Date and Time. Please visit https://developer.android.com/guide/topics/ui/controls/pickers#java for more info.
6. You must take care of the corner cases and validations.

Main Screen (40 points):

1. This is your app's main screen, see figure 1(a). It should display,
    1. A list of schedules, implemented by RecyclerView.
    2. Each item in the RecyclerView contains:
        1. Title.
        2. Place.
        3. Date and Time.
    3. The list must be sorted by Date and Time in the order of the earlier to the later schedules.

2. When the app loads, it must retrieve the saved schedules from Firebase and display the list.
3. There should be an Add button to add new schedules at the bottom of the screen.
4. **Long click on any of the item in the schedule list must ask the user if they want to delete the schedule.** And if they click Yes, then it should remove the schedule from Firebase.

Schedule a Meeting screen (40 points):

1. Clicking on the Add button for the main screen should start this screen.
2. This screen contains,
    1. 1 EditText to get the Title.
    2. 1 Button to start the Add Place screen.
    3. 1 Button to set the date.
    4. 1 Button to set the time.
    5. 1 Button to save the schedule.
3. Clicking on Add Place should open the Add place screen, see figure 1 (g).
4. Clicking on Set Date button should display the DatePicker. The user should be able to pick a date and set it. See figure 1(c).
5. Clicking on Set Time button should display the TimePicker. The user should be able to pick a time and set it. See figure 1(e).
6. Clicking on Save button should save the Schedule to Firebase, and get back to the main screen.

Add Place screen (20 points):

1. This screen should contain an EditText for the search keyword. See figure 1(g).
2. The user should write the keyword and click on Go button.
3. Upon clicking on Go button a list of places should be populated. You should use a RecyclerView to implement that.
4. Clicking on any of the item on the list should select that place, get back to 'Schedule a Meeting' screen, and set the selected place there.

Rubrics:

1. Main Screen: load the schedules when the app starts	5
2. Main Screen: managing the data structure (local and Firebase)	10
3. Main screen: implementing RecyclerView and sorting the list in the order of the earliest to the latest schedule	15
4. Input validations (getting all the inputs correctly for all the screens)	10
5. Schedule screen: Date picker	10
6. Schedule screen: Time picker	10
7. Schedule screen: Save data to Firebase	10
8. Schedule screen: Hiding the Buttons and displaying the TextViews	10

9. Add Place: RecyclerView	8
10. Add Place: Google Places API call	12
Total	**100**