

ITIS/ITCS 4180/5180 Mobile Application Development
Spring 2016
Homework 2

Date Posted: 02/02/2016 at 9:30
Due Date: 02/08/2014 at 23:55

Basic Instructions:

1. In every file submitted you MUST place the following comments:
 - a. Assignment #.
 - b. File Name.
 - c. Full name of all students in your group.
2. Each group should submit only one assignment. Only the group leader is supposed to submit the assignment on behalf of all the other group members.
3. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will lose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
4. Please download the support files provided with this assignment and use them when implementing your project.
5. Export your Android project and create a zip file which includes all the project folder and any required libraries.
6. Submission details: The file name should follow the following format:
Group#_HW02.zip
7. Failure to follow the above instructions will result in point deductions.

Homework 2 (100 Points)

In this assignment you will get familiar with Android intents (Explicit/Implicit), data passing between activities, using `startActivityForResult`, and alert dialogs. This is a Ticket Reservation App, where the user gets to create a ticket with source, destination and departure and return dates, along with other ticket details. User should be able edit, display and delete tickets. This is a multi-screen application; the Main Activity is displayed in Figure 1.

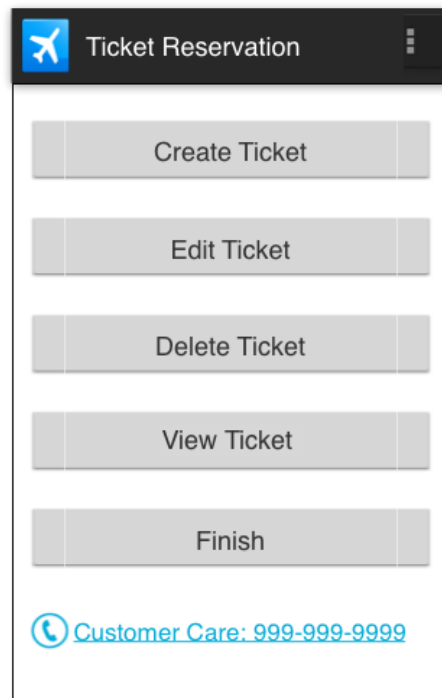


Figure 1, Main Activity Wireframe

Main Activity (10 Points)

The Main Activity contains the 5 buttons arranged as displayed in Figure 1. The Main Activity should maintain a `LinkedList` of the current Ticket Objects, the `LinkedList` should be shared with the other activities through intents when needed. The activity requirements are as follows:

- The ticket list should be stored in the Main Activity. The Create Ticket, Edit Ticket, Delete Ticket, View Tickets, and Finish buttons should start the Create Ticket Activity, Edit Ticket Activity, Delete Ticket Activity, View Ticket Activity and finish the activity respectively.
- The Customer Care number should be clickable and should start the Android's Calling Intent when clicked on it.

CreateTicket & Print Ticket Activity (20 Points)

The CreateTicket Activity is displayed in Figure 2. Implement the following:

- When the user selects “Create Ticket” button in the MainActivity, the CreateTicket Activity should be started as shown in Figure 2.
- The Name field should not exceed 20 characters.
- The source, destination, date and time EditTexts should not be editable, you should set their key listeners to null, (setKeyListener(null)).
- The Source and Destination will contain a list of cities as shown in Table 1. An error toast message should be displayed if the source and destination are the same.
- When the “One-Way” button is checked, the Returning date and time should be hidden and when the user checks the “Round-trip” button, the Returning date and time should be made visible.
- When the Departure and Returning date and time EditTexts are selected, the corresponding date picker or time picker dialog boxes should be displayed to enable the user to pick the date or time respectively. Figure 2, shows the date and time picker displayed when the date and time EditText is clicked. The selected time and date should be displayed in their corresponding edit text as shown in Figure 2.
- Clicking the Print Summary button should save the fields as a Ticket object and start the PrintTicket Activity and send it as a result to the Print Ticket Activity so that it can be stored in the ticket list and be displayed in the PrintTicket Activity in various TextViews.
- The Finish Button should stop the PrintTicket Activity and start the MainActivity.

Cities		
Albany, NY	Houston, TX	Portland, OR
Atlanta, GA	Las Vegas, NV	Raleigh, NC
Boston, MA	Los Angeles, CA	San Jose, CA
Charlotte, NC	Miami, FL	Washington, DC
Chicago, IL	Myrtle Beach, SC	
Greenville, SC	New York, NY	

Table 1, List of cities

Edit Ticket Activity (25 Points)

The EditTicketActivity is displayed in Figure 3. Implement the following requirements:

- When the user selects “Edit Ticket” button in the Main Activity, the EditTicket Activity should be started as shown in Figure 3.
- Clicking the Select Ticket button should show an Alert Dialog displaying the names of persons, which enables the user to pick which ticket to edit. Upon selecting a name, the ticket information should be populated in the EditTicket activity as shown in Fig 3.

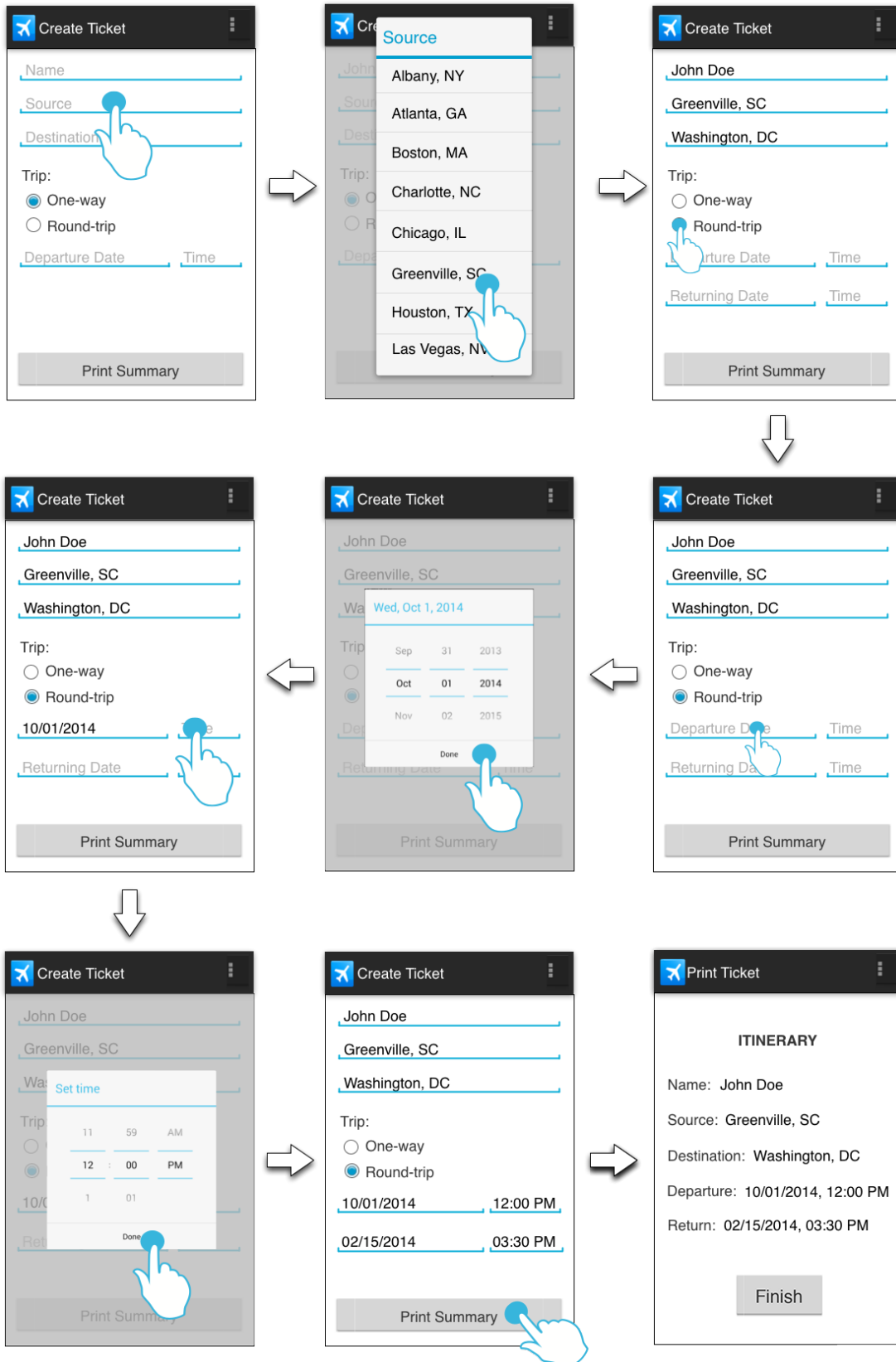


Figure 2, CreateTicket Activity

- The UI for this activity is similar to the Create Ticket Activity and should follow the same text size, and time/date picker implementation and restrictions.
- Clicking the Save button should save the fields as a Ticket object and send it back as a result to the Print Ticket Activity so that it can be updated and stored in the ticket list.
- The Finish Button should stop the PrintTicket Activity and start the MainActivity.

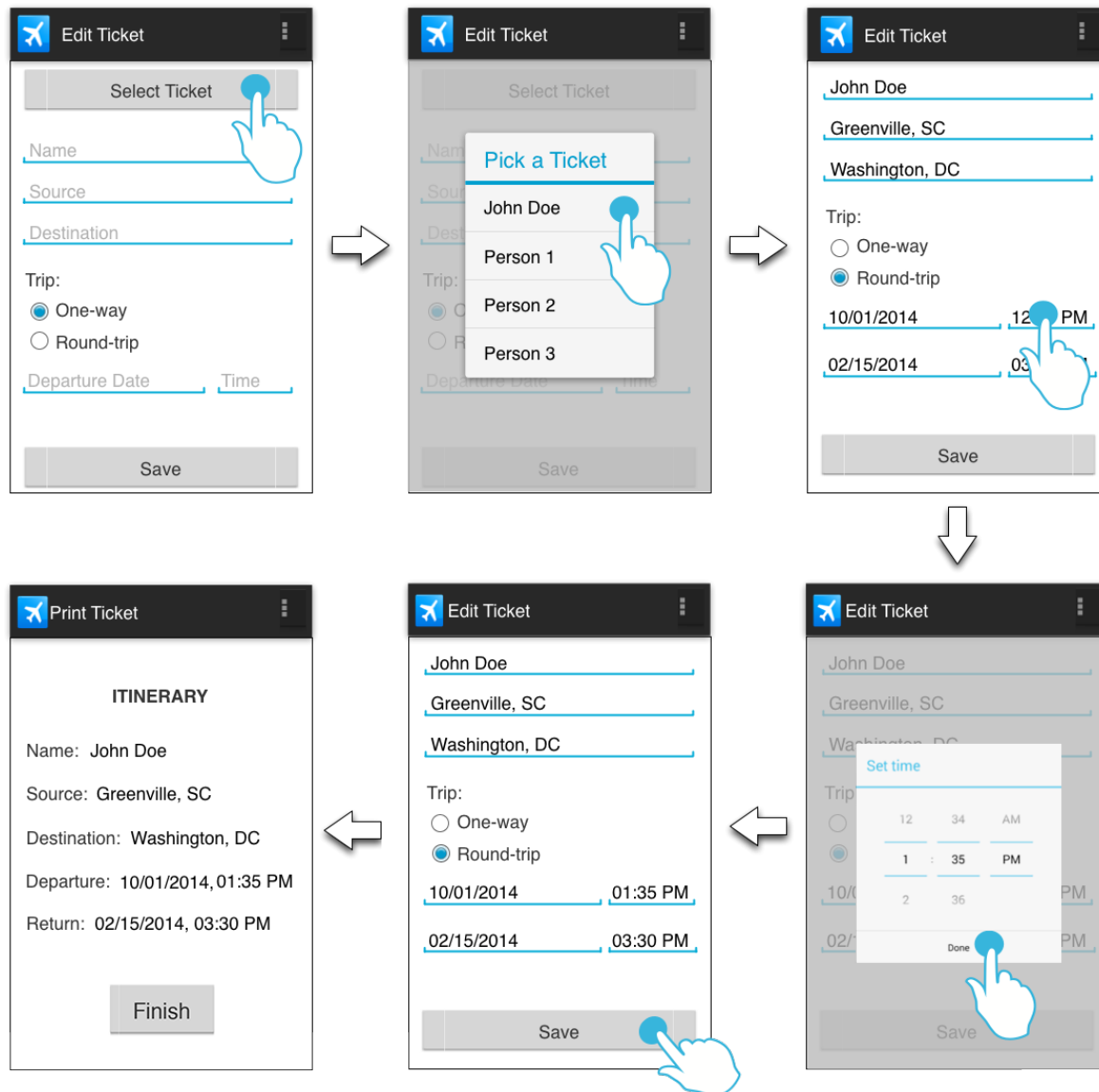


Figure 3, Edit Ticket Activity

Delete Ticket Activity (25 Points)

The DeleteTicket activity is displayed in Figure 4. Implement the following requirements:

- When the user selects “Delete Ticket” button in the MainActivity, the DeleteTicket Activity should be started as shown in Figure 4.

- Clicking the Select Ticket button should show an Alert Dialog displaying the person names, which enables the user to pick which ticket to delete. Upon selecting a name, the ticket information should be populated in the Delete ticket activity as shown in Figure 4.
- All the EditTexts and radio buttons should not be clickable, and should only be used to present the current Ticket's information.
- Clicking the Delete button should delete the current ticket and send the Main activity the required information in order to delete this ticket from the ticket list.
- Clicking the Cancel button should finish the Delete Ticket activity without deleting the ticket.

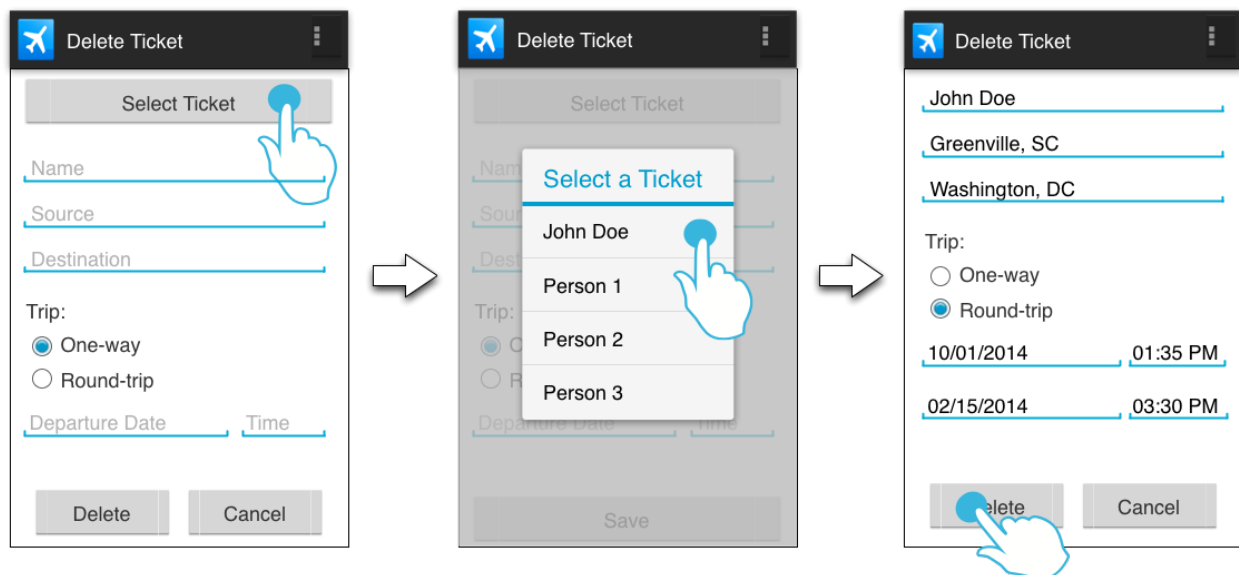



Figure 4, Delete Ticket Activity

View Ticket Activity (20 Points)

The ViewTicket Activity is displayed in Figure 5. Implement the following requirements:

- When the user selects “View Ticket” button in the Main Activity, the ViewTicketActivity should be started as shown in Figure 5.
- This activity enables the user to view all the currently stored tickets.
- All the EditTexts and radio buttons should not be clickable, and should only be used to present the current ticket's information.
- The actions required from the each of the GUI images are listed in Table 2. Upon clicking finish the activity should be finished.


View Ticket

Name

Source

Destination

Trip:

☒ One-way
☐ Round-trip

Departure Date

Time

⏮

⏪

Finish

⏩

⏭

Figure 5, View Ticket Activity

Button	Action
⏮	Show the first ticket
⏪	Show previous ticket
⏩	Show next ticket
⏭	Show last ticket

Table 2, GUI Actions