Mobile Project and Senior Design Project Application Participation Break Down

Mobile:

Jesse:

* Activity 1:
  + Created and Edited Globus\_Welcome\_Screen.java
  + Created and Edited activity\_globus\_\_application.xml
  + Created and Edited Welcome\_Screen.java
  + Created and Edited fragment\_globus\_\_welcome\_\_screen.xml
  + Created and Edited Login\_Screen.java
  + Created and Edited fragment\_login\_\_screen.xml
  + Created and Edited Create\_Account\_Screen.java
  + Created and Edited fragment\_create\_\_account\_\_screen.xml
* Activity 2:
  + Edited Group\_Select\_Screen.java
  + Edited joined\_group\_listview.xml
  + Edited fragment\_group\_\_select\_\_screen.xml
* Activity 3:
  + Edited Globus\_Applicaion.java
  + Edited activity\_globus\_\_applicaion.xml
  + Edited Whiteboard\_Button\_Screen.java
  + Edited fragment\_whiteboard\_\_button\_\_screen.xml
  + Edited Whiteboard\_Screen.java
  + Edited fragment\_whiteboard\_\_screen.xml
  + Edited whiteboard\_listview.xml
  + Created and Edited Calendar\_Screen.java
  + Edited fragment\_add\_\_calendar\_\_event.xml
  + Created and Edited fragment\_calendar\_\_screen.xml
  + Created and Edited message\_creation\_layout.xml

User’s log (stardate: 92540.07):

For activity 1, I wrote the basic fragments including the text views, edit text fields and buttons. These fragments weren’t pretty. Just the basics (the ui was later redesigned by Taylor). Towards the end of development, I fixed a few bugs stemming from our prewritten login information and tested the final versions of the fragments.

For activity 2, I wrote a custom array list adapter called “groupAdapter” to display a list of groups. In order to make this work, I wrote a custom class type called “groupEntity” and I redesigned the joined\_group\_listview layout xml file to utilize the groupEntity and to fit the UI design. I also added in an onItemClickListener for the specific group entries in the list so the user can select a premade/joined group to view.

For activity 3, I wrote two more custom classes called calendarItem and whiteboardItem. These classes both serve to make it easier to store/utilize calendar events and whiteboard items (messages and events). While I did write a lot of code for the calendar\_screen fragment, not much of it can be used in this version of the Globus app. This is because much of what the calendar does is interact with databases, and until the Globus database is functioning, I cannot fully implement or test these functions. Aside from this, however, I did create a basic dialog in this fragment which, in the future, will allow users to create new events. In the whiteboard\_screen fragment, I made many modifications to the ui, added a second custom list adapter to display both events and messages (each of which is displayed with slight modifications to help users differentiate) and wrote a custom dialog that will add user made messages to the whiteboard. For the two dialog, I also needed to create the layout xml file titled message\_creation\_layout and modify the beginnings of the add\_calendar\_event fragment layout file (while this was originally intended to be a fragment all on its own, I thought it would be better suited as a dialog as it would keep the user from leaving the calendar view and it would allow us to use functions from the calendar class without having to re-write them in a new file). Regrettably, I forgot to write an onItemClickListener for the events and messages in the whiteboard fragment before we submitted the project. One other thing worth noting that I did on this activity was that I compressed the images used as the whiteboard buttons. These icons were at the time causing a particularly nasty problem. They were causing my phone to run out of memory because they ranged in size from 22.3 KB to 92.4 KB each. To fix this, I resized the icons in paint, and compressed them further using tiny png (<https://tinypng.com/>). The images now range from 659 bytes to 1.53 KB.

Kelsey:

* Activity 1:
  + Edited Globus\_Welcome\_\_Screen.java
  + Edited Login\_Screen.java
  + Edited Create\_Account\_Screen.java
* Activity 2:
  + Created and Edited Globus\_Group\_Selection\_Screen.java
  + Created and Edited activity\_group\_screen.xml
  + Created and Edited Group\_Select\_Screen.java
  + Created and Edited joined\_group\_listview.xml
  + Created and Edited fragment\_group\_\_select\_\_screen.xml
  + Created and Edited Create\_Group\_Screen.java
  + Created and Edited fragment\_create\_\_group\_\_screen.xml
  + Created and Edited Join\_Group\_Screen.java
  + Created and Edited fragment\_join\_\_group\_\_screen.xml
* Activity 3:
  + Created and Edited Globus\_Application.java
  + Created and Edited activity\_globus\_applicaiton.xml
  + Created and Edited Whiteboard\_Button\_Screen.java
  + Created and Edited fragment\_whiteboard\_\_button\_\_screen.xml
  + Created and Edited Whiteboard\_Screen.java
  + Created and Edited fragment\_whiteboard\_\_screen.xml
  + Created and Edited whiteboard\_listview.xml
  + Create fragment\_add\_\_calendar\_\_event.xml
  + Created and Edited Google\_Drive\_Screen.java
  + Edited fragment\_google\_drive\_screen.xml

Description of what group member did:

Activity 1: I just added the fragments and activity connections between activity 1 and activity 2 in the fragments that Jesse created in activity 1. I also set up some of the code to prove that the login works.

Activity 2: I created the second activity as well as ended the first activity and I created all of the fragments in activity two and then linked all of the fragments to activity 1 where appropriate. I also made sure to pass the appropriate information between activity 1 and 2 in intent lines. I also did the majority of the xml for activity 2 which included changing colors, formatting, etc.. I then linked the appropriate fragments to the next activity and next fragments.

Activity 3: I created the third activity as well as the multi pane fragment that displays the buttons and the whiteboard. I linked all of these pages together. I also linked all of the buttons to the appropriate fragments as well as ‘writing’ most of the google drive code. I got this code from an example from class so I use the term ‘writing’ loosely as Jim Ward really wrote about 98% of it in the webView example. I then passed that code off to Taylor to change the web destination of the code. I also created the skeleton of fragment\_add\_\_calendar\_\_event.xml for Jesse to edit and then linked the appropriate calendar fragments together. The majority of what I did for this project is specifically for the mobile project verses senior design as it shows that the fragments and activities can interact with each other.

Taylor:

* Activity 1:
  + Edited fragment\_globus\_\_welcome\_\_screen.xml
  + Edited fragment\_login\_\_screen.xml
  + Edited fragment\_create\_\_account\_\_screen.xml
* Activity 2:
  + Edited fragment\_group\_\_select\_\_screen.xml
  + Edited fragment\_join\_\_group\_\_screen.xml
  + Edited fragment\_create\_\_group\_\_screen.xml
* Activity 3:
  + Edited Globus\_Application.java
  + Edited fragment\_whiteboard\_\_button\_\_screen.xml
  + Edited whiteboard\_listview.xml
  + Edited activity\_globus\_\_application.xml
  + Created and Edited Google\_Drive\_Screen.java
  + Created and Edited fragment\_google\_drive\_screen.xml

Description of what group member did:

Within Activity 1, after Jesse wrote the fragments for the three screens, I drew up the interface mockup, and after getting the group’s approval, formatted the fragments to look that way. This ended up being the primary design scheme for Globus. The same happened for Activity 2 fragments.

Beginning Activity 3, I designed the initial mockup for the Whiteboard design.

For Activity 3, I spent a lot of time trying to get the Globus database to work. Working with David, our other Senior Design group member, we attempted to get the database on the server to communicate with the application. Long story short, it ended up being an issue with android not allowing the connection out. We decided to attempt to get the database to be local on the phone. I spent many hours attempting this, but just could not get it to work with the system in time. Finally, my other duties for Activity 3 included button art and the Google Drive fragment. The Google Drive fragment is based on Jim Ward’s code. I implemented it and formatted it to fit the UI design of Globus, and it opens to the Google Drive login. We also decided to include a place on the fragment for the Group’s username and password for the Google Drive to appear. This will be implemented later. For the button art, I drew each button as an .png image at 1000x1000 pixels, so that when they were scaled down, they still look nice. With the spacing of the whiteboard, the buttons are squished a bit. I plan to implement a “Globus” icon at the top, which will help with size and spacing issues of the buttons. I didn’t do that now because there was a memory leak caused, and I couldn’t figure out where with the amount of time I had left.