CSC301, Final Team Demo

Monday, April 2nd - Tuesday, April 3rd

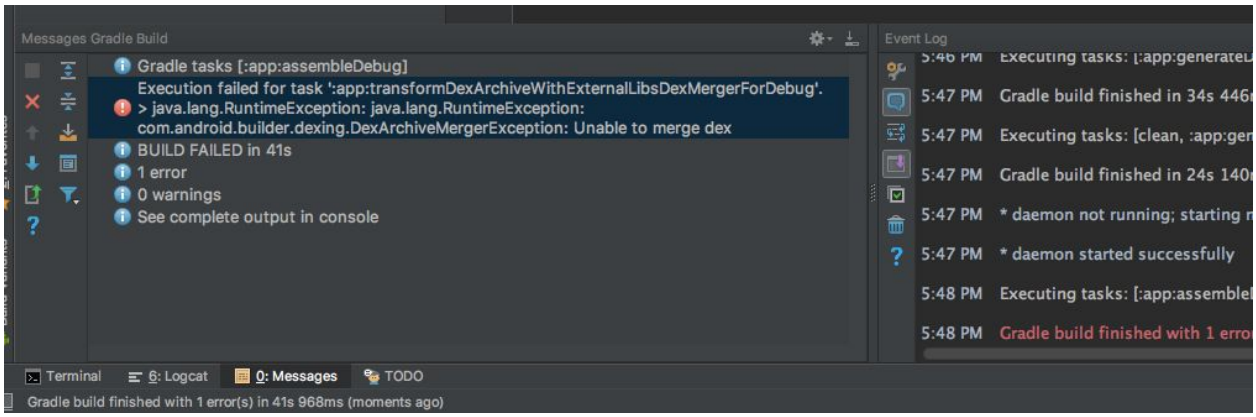
Your final demo will be similar to the videos you submitted for deliverables, but there are a few differences:

- It is a live demo, not a recorded video.
- It will be evaluated by the instructor(s), not the TA's.
- It is slightly longer and should include additional discussions (see details below).

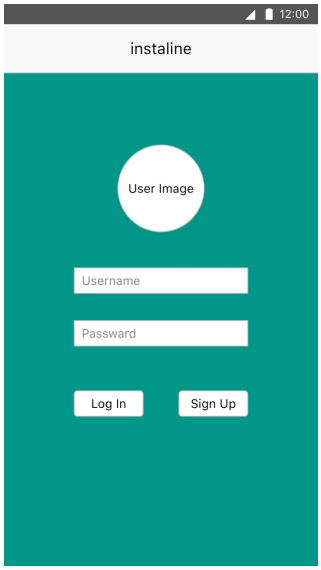
To be clearer, your presentation should include the following:

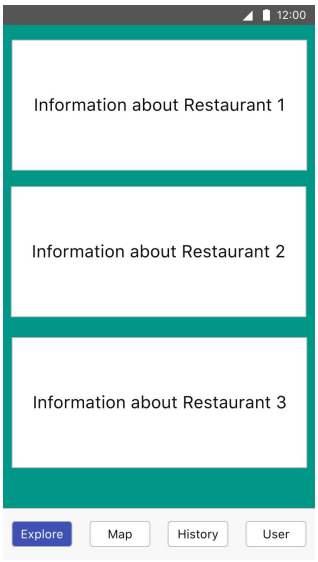
- Demo (max' 5 minutes)
    - Introduce the problem you are solving (and the users you are solving it for).
      Our android application project, Instaline, will provide a virtual lining service to solve variety of lining up inconveniences. While online/phone reservation is not flexible and waiting for a single restaurant is too frustrating, Instaline intends to free users from being physically lining up in the restaurant. Users can start or quit lining up at anytime, in anywhere, for whatever restaurants they wish.
    - Demonstrate the solution you built.
      Follow the video script.
    - This part should be similar to the videos you submitted for previous deliverables.
- Technical Discussion (max' 3 minutes)
  (从这里开始往下，用ppt演示的效果应该会比直接讲来得好)
    - Describe 1 - 3 technical items that came up while working on your project.
    - A technical item can be an interesting bug, design change, lesson you learned from using a certain technology/framework, etc.
    - Be specific and feel free (and encouraged) to include useful artifacts (e.g. code snippets, screen shots, etc.)

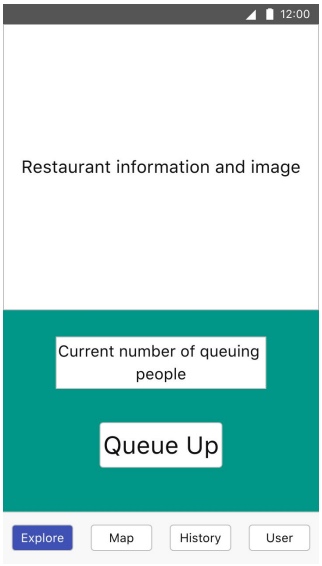
First, the android studio environment setting. We met various android studio environment setting problems. Android studio is a highly advanced android software development IDE, built by Intellij. The advantages of android studio are fully functional, multi-platform, intelligent, and similar to java. The disadvantages, which affect the development, are mostly related to the environment setting of android studio. We tried hard to fix this kind of bugs, for example, we clean the project before running, created a new project with old code, restarted android studio and even the computer. At the end, by the help of stack overflow, we reset gardle files, which solved almost all bugs.



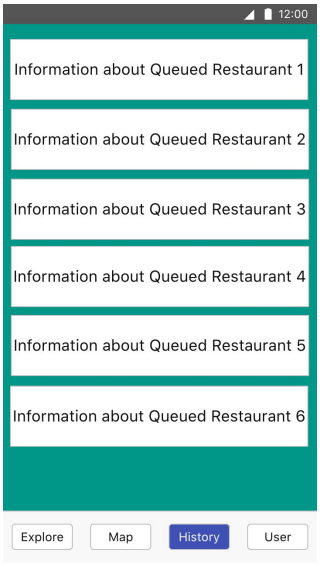
Second, the front-end design changes during the development. For example, after completed the restaurant list of customer side, we found out that if there was no search function, it would be hard for users to find their ideal restaurant(s) once there existing too many restaurants in the list. So we added the search function for the customer side app. Except this, we also improved some details of the front-end design, you can see the difference between the mock-ups and the real application.

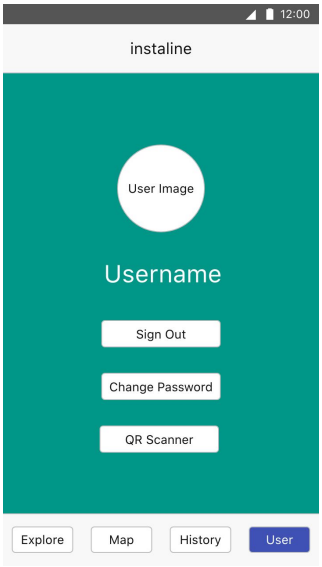

12:00
instaline
User Image
Username
Password
Log In
Sign Up

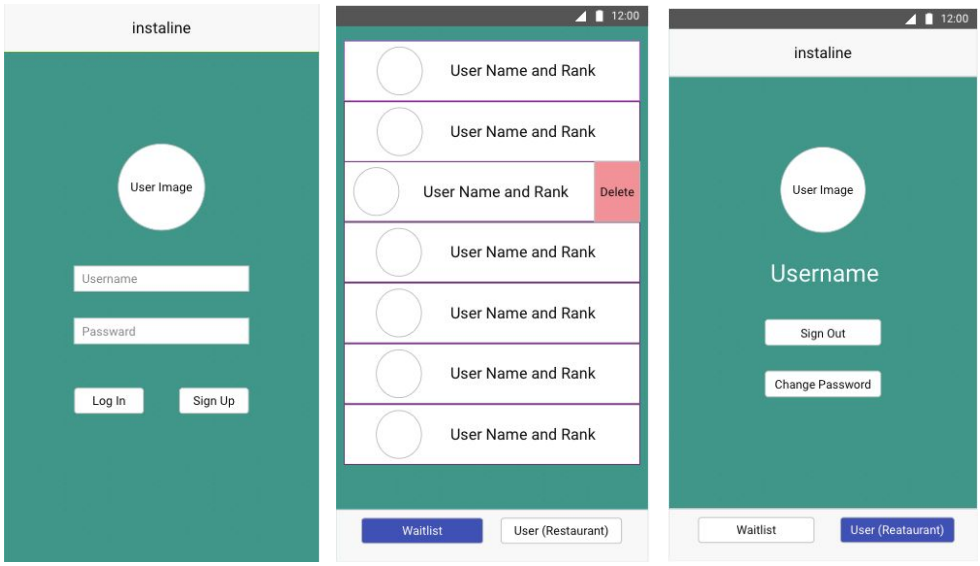

12:00
Information about Restaurant 1
Information about Restaurant 2
Information about Restaurant 3
Explore
Map
History
User


12:00
Restaurant information and image
Current number of queuing people
Queue Up
Explore
Map
History
User


12:00
SUMMERHILL
RATHNELLY
Park Drive Reservation Lands
YORKVILLE
Restaurant 1
Restaurant 3
CHURCH AND WELLESLEY
Restaurant 2
Adega
Toronto
CHINATOWN
Sheraton Cen
Explore
Map
History
User


12:00
Information about Queued Restaurant 1
Information about Queued Restaurant 2
Information about Queued Restaurant 3
Information about Queued Restaurant 4
Information about Queued Restaurant 5
Information about Queued Restaurant 6
Explore
Map
History
User


12:00
instaline
User Image
Username
Sign Out
Change Password
QR Scanner
Explore
Map
History
User



Third, the connection between customer side and restaurant (enterprise) side. We use firebase as a server to manage the database between two apps, which means this database should monitor data changes from different apps accurately. Thanks to this development, we get more familiar with the firebase server and back-end data manipulation.

- Process Discussion (max' 3 minutes)
    - Describe your team's workflow: How did you communicate? How did you use Git/GitHub? How did you divide tasks? etc.
    - Highlights significant decisions (good or bad) that you've made along the way, and explain the rationale behind them.

We used mobile chat application to create a Instaline development group for communication. Git and GitHub are tools that we use to do the version control.

- Individual Contribution (max' 1 minute per team member)
    - At the end of your presentation, every team member will be asked to (very briefly) describe their main individual contribution(s) and/or lesson(s) learned.

- Jack (Chenjie Ni): Product Manager
    - Knowing customers' needs and main features of our app
    - Assign general tasks to every team member

  What you learned:

- Jasmine (JiaXin Wu): Scrum Master
    - Coordinate with the availability of each team member
    - Do the overall testing

  What you learned:

I learned how to communicate with all team members as well as coordinate everyone's availability. For the development, I learned that using online resources is really useful, like using knowledge on Stackoverflow to solve bugs. Also, some existing API is really useful that we can directly use like the SearchView. For the testing, I learned that it is always a good idea to test backend and frontend separately which helps locating the bug quickly.

- Jerry (Jialiang Yi): Frontend UI Designer and some full stack
    - Leader of the frontend group
    - Make decisions on how each page should look like
    - Connect front end with database

  What you learned:

Learn android development from scratch. Get familiar with android ui components and their lifecycle. Worked with firebase database, specifically, its authentication functionalities and data listeners. Communicate with backend designer to retrieve data from backend and display them accordingly in the front end. Moreover, I learn to use libraries from the internet that developed by others, import and modified them in order to use them in our project.

- Eason (Yichong Guan): Backend Firebase Designer
    - Decide how the data is stored and retrieved in a feasible way

  What you learned:

- Hao (Hao Wang): Frontend Designer
    - Frontend logic on how data is observed and what actions it will perform

  What you learned:

- Bill (Zhihong Wang): Frontend Designer
    - Frontend structure view
    - User input gesture

What you learned: I learned how to design user views by linear layout, relative layout, text view, image view, and card view; how to setup an activity files to connect the data with xml files; how to transfer the activity file to a fragment file by searching stack overflow; how to find and use helpful api and library, like the google map api and the swipe library we used; how to monitor user gestures by using android built in attributes.

When preparing your demo, think about your audience (i.e. the instructor(s)).
We don't know your project as well as your TA, and we are interested in:

- Understanding who your target users are.
- Understanding why your software is valuable to its users.
- Seeing your prototype in action and understand the user experience.
- Understanding how you worked together as a team.

During your demo, we may ask you a few questions. You can expect

- Technical questions about your software
- Questions about how you worked together as a team

Logistics

Total time (including questions from the instructor(s)): 15-20 minutes

When: Monday, April 2nd - Tuesday, April 3rd
Please register your team in this spreadsheett. It will be open for editing **Friday, March 30th at 5:00pm.**

You will be able to connect your laptop to the projector via VGA and to the audio system (headphones jack).
If you need any additional audio/video equipment, please let us know in advance.

Evaluation

We will evaluate the following aspects of your demo:

- Clarity & focus
  - We should have a very clear idea of *what* you built, *who* you built it for and *why*.
  - The value to the users should be clear.
- Quality of presentation
  - Practice your demo in advance, and make sure it has a coherent and logical flow.
  - When answering questions, make sure you understand the question first. When in doubt, ask us to rephrase the question (instead of spending time answering the wrong question.
- Overall evaluation of the project
  - Effort that was put in.
  - Quality of the work (planning & implementing).
  - Insights (technical or non-technical).