1.Contents of the electronic submission:

- We are attaching the whole android project as a zip file which includes the following package hierarchies:
  - businessLogic:
    - TransitListGenerator.java
    - TransitListPopulator.java
    - DatabaseService.java
      - Location:
        - LocationChangeListener.java
        - LocationPreferences.java
        - OnLocationChanged.java
  - objects:
    - BusRoute.java
    - BusRouteSchedule.java
    - BusStop.java
    - BusStopSchedule.java
    - BusVariant.java
    - Location.java
    - ScheduledStop.java
    - Street.java
    - Time.java
    - TransitListItem.java
  - presentation:
    - BusListViewFragment.java
    - DisplayAdapter.java
    - Mainactivity.java
  - Persistence:
    - database:
      - Database.java
      - DatabaseAccessStub.java
    - transitAPI:
      - ApiListenerCallback.java
      - TransitAPI.java
      - TransitAPIClient.java
      - TransitAPIProvider.java
      - TransitAPIResponse.java
- test/java:
  - comp3350/WinnipegTransitGo:
    - tests:
      - businessLogic:
        - TransitListGeneratorTest.java
      - Objects:
        - TransitListItemTest.java
    - AllTests.java
    - ExampleUnitTest.java

2.Log file:

The individual logs and the team meeting logs are located in the log.pdf file

3.Overview of major implemented feature:

- Showing nearby bus stops in map (Location in GUI: This can be found in the upper half of the screen which displays a map)
- Showing Real Time bus schedules (Location in GUI: Each row of the list view contains the real-time schedules of each bus)
- Showing Real Time bus status such as whether the bus is late or early or due.
- Showing real time 3 consecutive departure timings for each bus
- Showing the distance from the user to the nearby bus stops
- Ability to pin point the location of the user in the map
- Ability to refresh to get the latest bus timings by moving user's location in the map or clicking the button to locate themselves in the map.
- Automatic information refreshes based on location change.
- Showing the bus stop name, bus stop number and the bus route for each individual bus.

4.Bugs and Known issues:

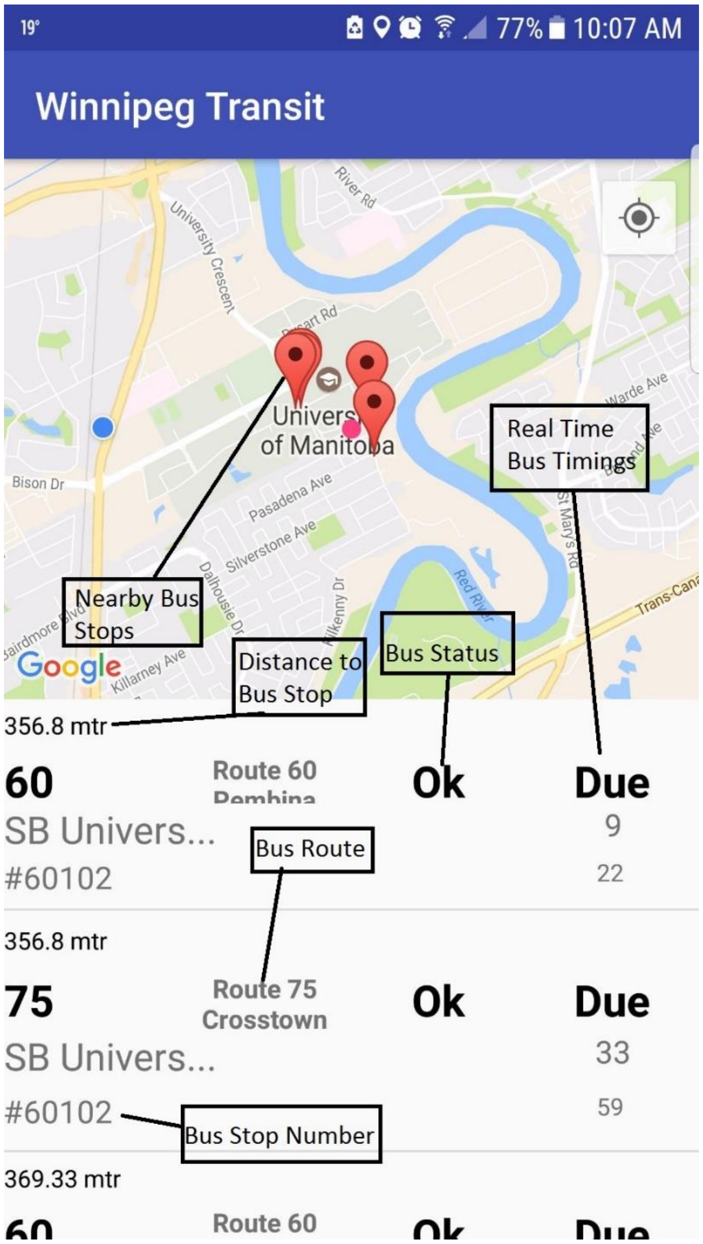
- The api call for getting the bus stop names and bus route information returns strings in a long format that we cannot fit in the list. In the next iteration we will use the short names for the bus stop and the bus route so that it fits well in the listview. We decided to push this small issue into the next iteration since we decided to put our effort into features with higher cost.

5.Test Cases:

- There are no test cases for the GUI as there are no logic in the GUI
- we don't have any test cases for the API calls since all the logic for the API calls are handled by the "RETROFIT" framework. The API code doesn't contain any logic and all the objects used to store information from the API are simple objects that only provide getter functions.

6.Tutorial on using the app:

- The app allows users to see real time departure times of nearby buses
- After opening the app the app asks user to allow the app to access user location and the user has to click allow to proceed.
- Then the user will be presented with a list of nearby buses with their departure time and a map showing the nearby bus stops in it.
- The user can click on the circle located at the top right of the screen to find buses and bus stops near their location.
- For each bus the information about the bus stop name, bus stop number, and distance from the user, bus route and schedules of 3 departure time is shown.



7.Permission to use API calls:

- We realize that for this iteration, making any API calls or using external services was not allowed, however, before iteration 1, our whole team requested permission from Franklin to use API calls and google maps since our app is light on GUI and heavy on services and logic. Franklin agreed and gave us permission to use API calls for this iteration.

8.Sketch of the overview architecture:

- The sketch of the overall architecture of the system can be found as a hard copy in the folder submitted along with other required documents of iteration 1.
- Or in the file “Architecture_Sketch_Iteration1.png”