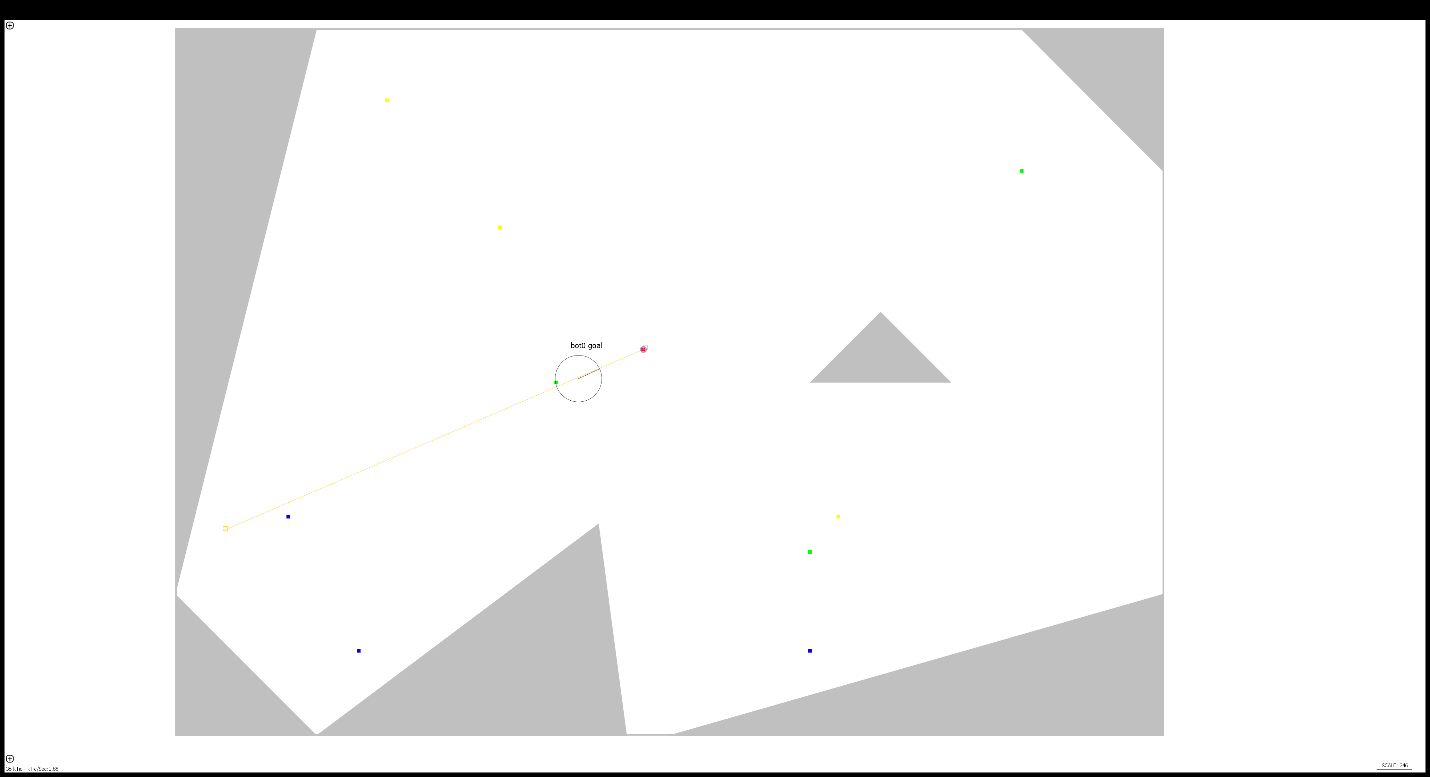
**CSE4340/CSE5349, Spring 2015, Final Project**

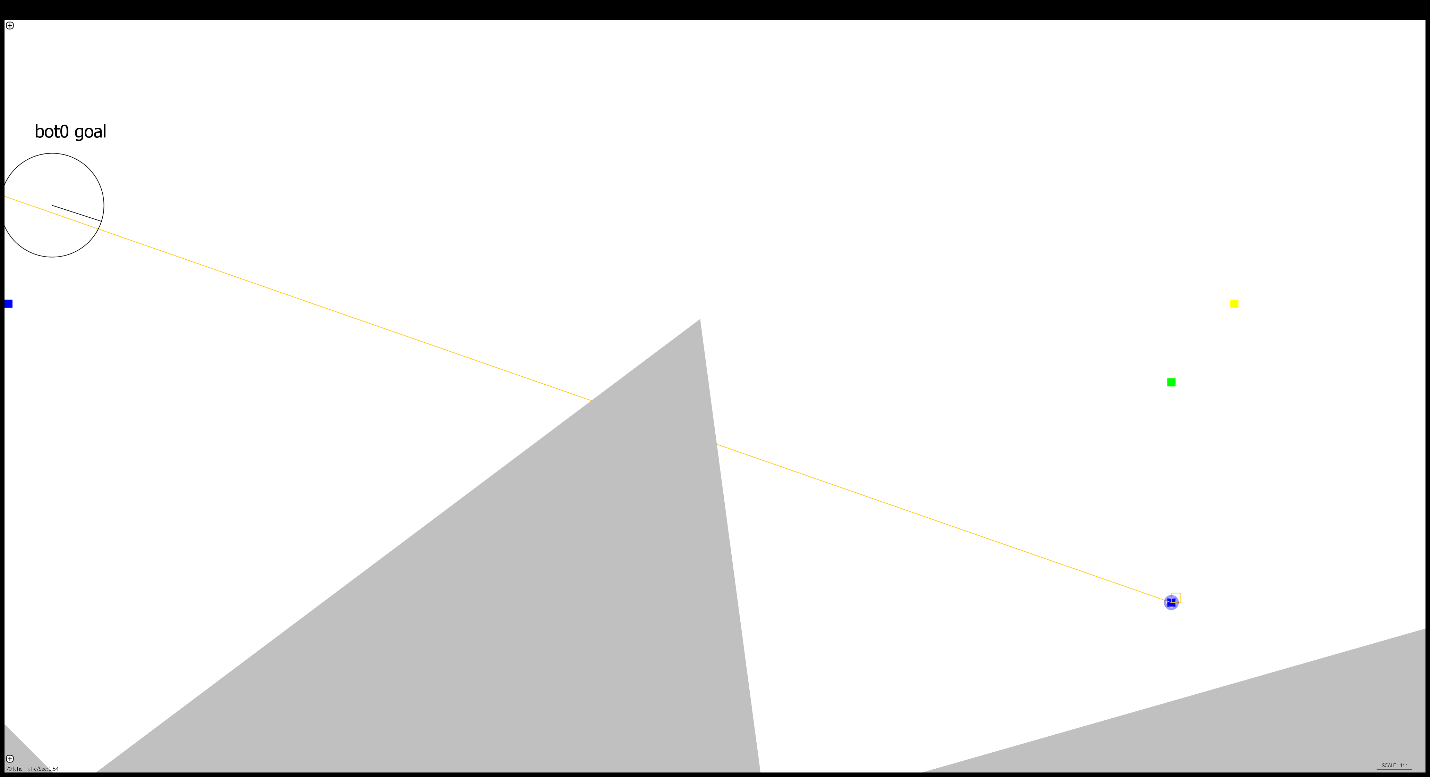
**Instructions for Given Files**

**Overview:**

The given files are specified in ProjectApp, this will provide an environment like the one shown in slides in class. Additionally, we included another app called CircleApp that you may find useful to look at, although it doesn’t solve the problem for the project. ProjectApp will show the following environment you can use to test. In your tests of your solution, you should definitely try varying the numbers of robots and changing the waypoint sets (both numbers in each set, the total number of sets, and the specific locations), and you may want to play with creating other test environments.

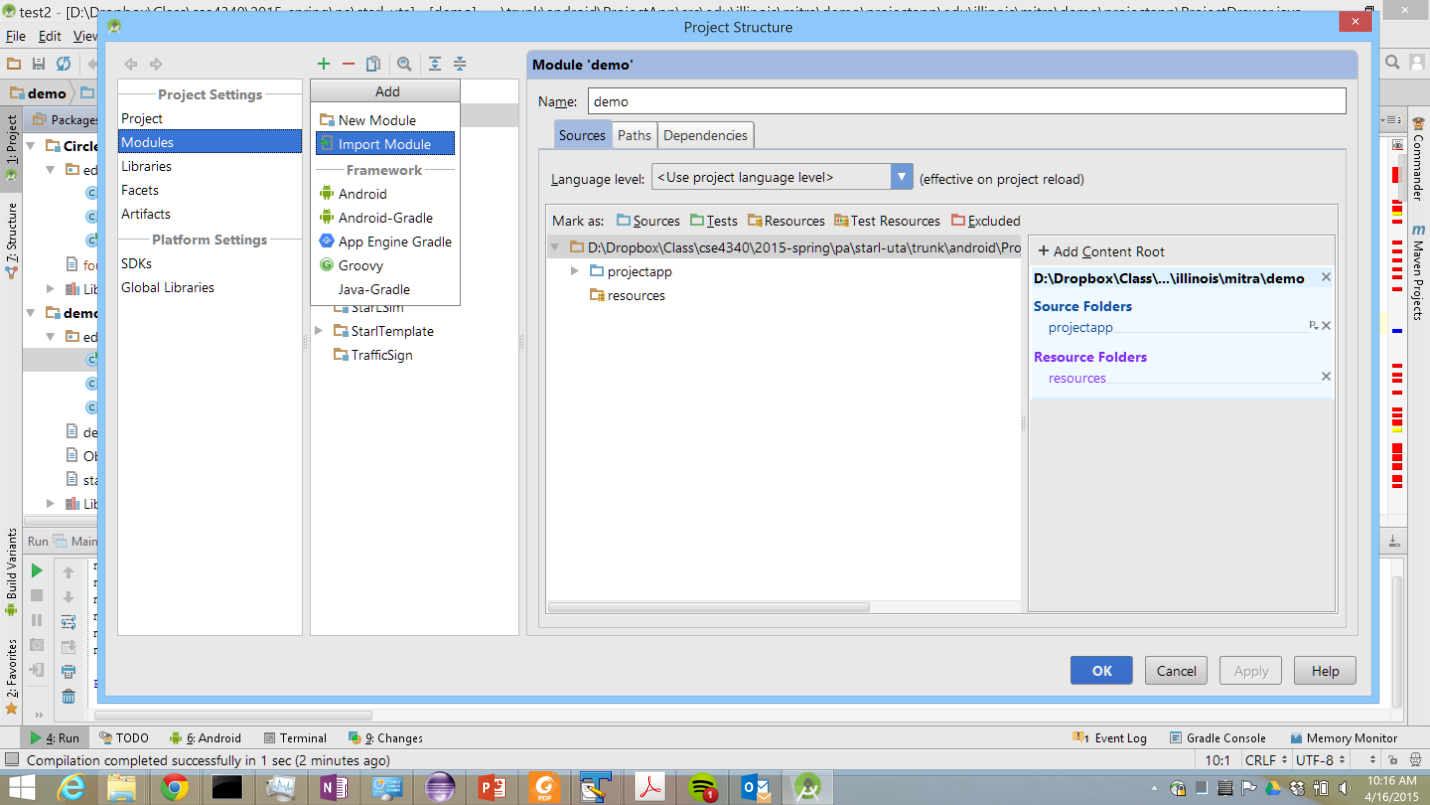


**Zoomed further:**

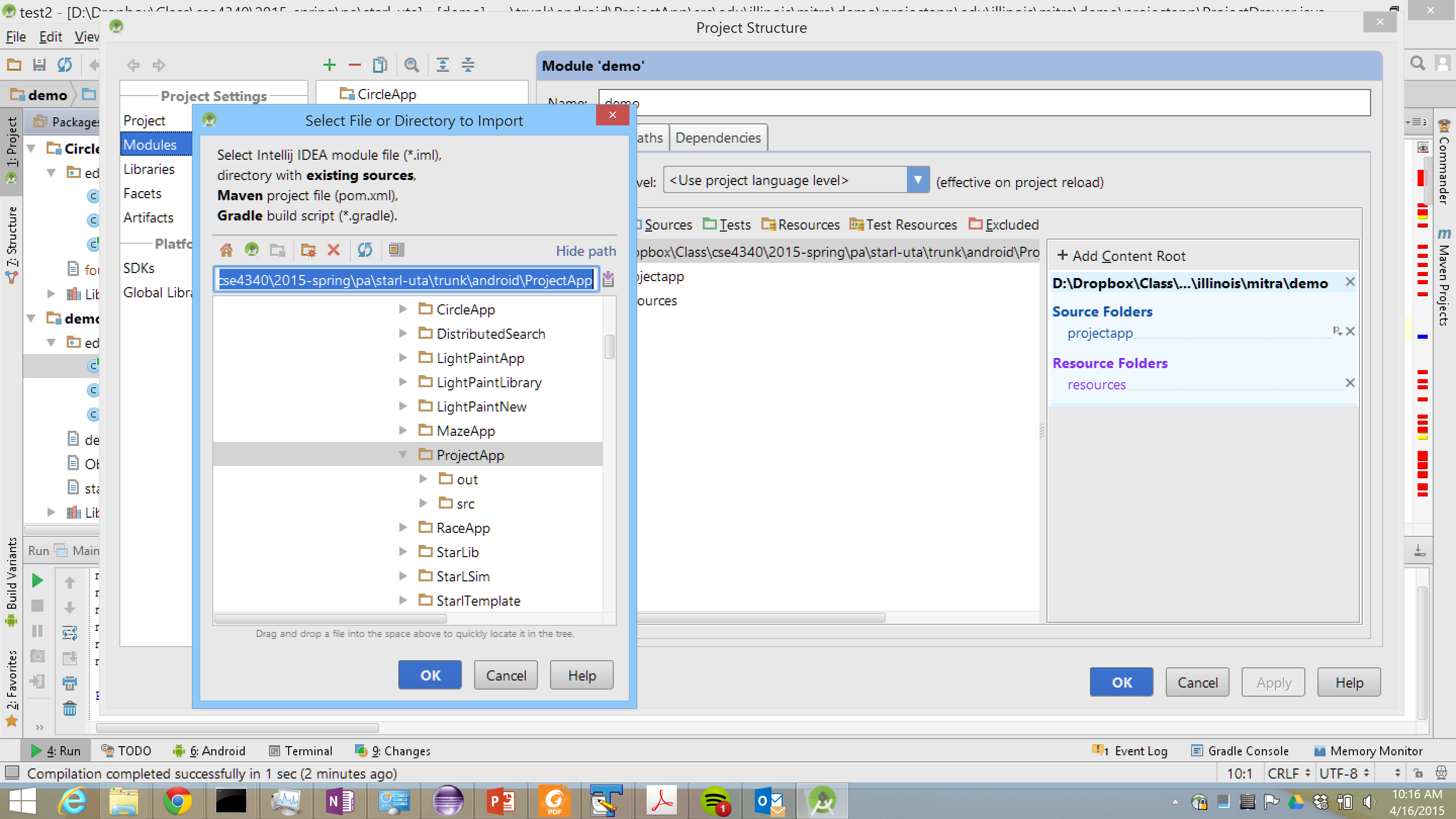


**Importing given files:**

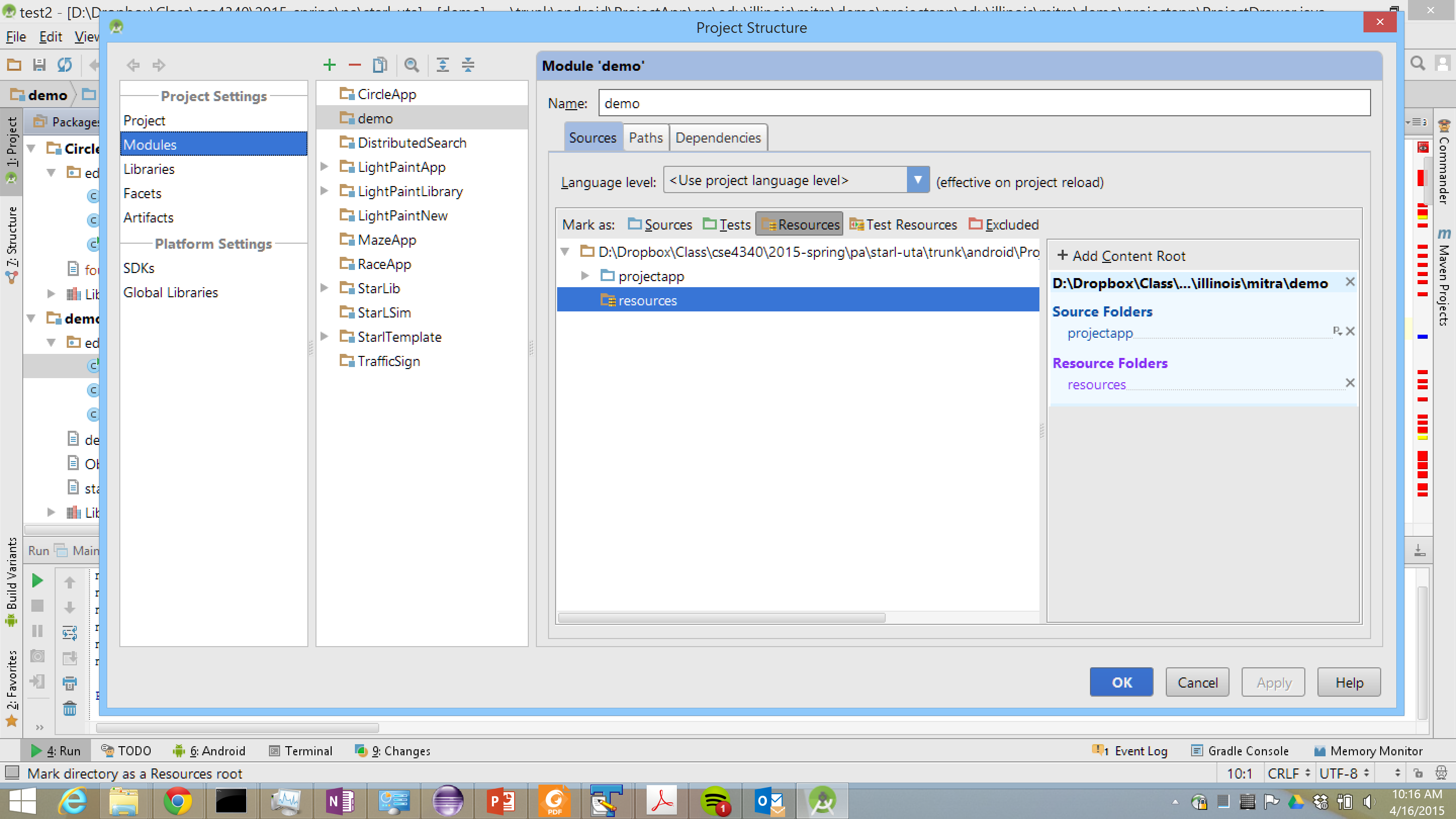
First, select open existing project from Android Studio, and open the given project. Next, go to File->Project Structure and click on Modules. Then click on the green + and select Import Module as shown below.



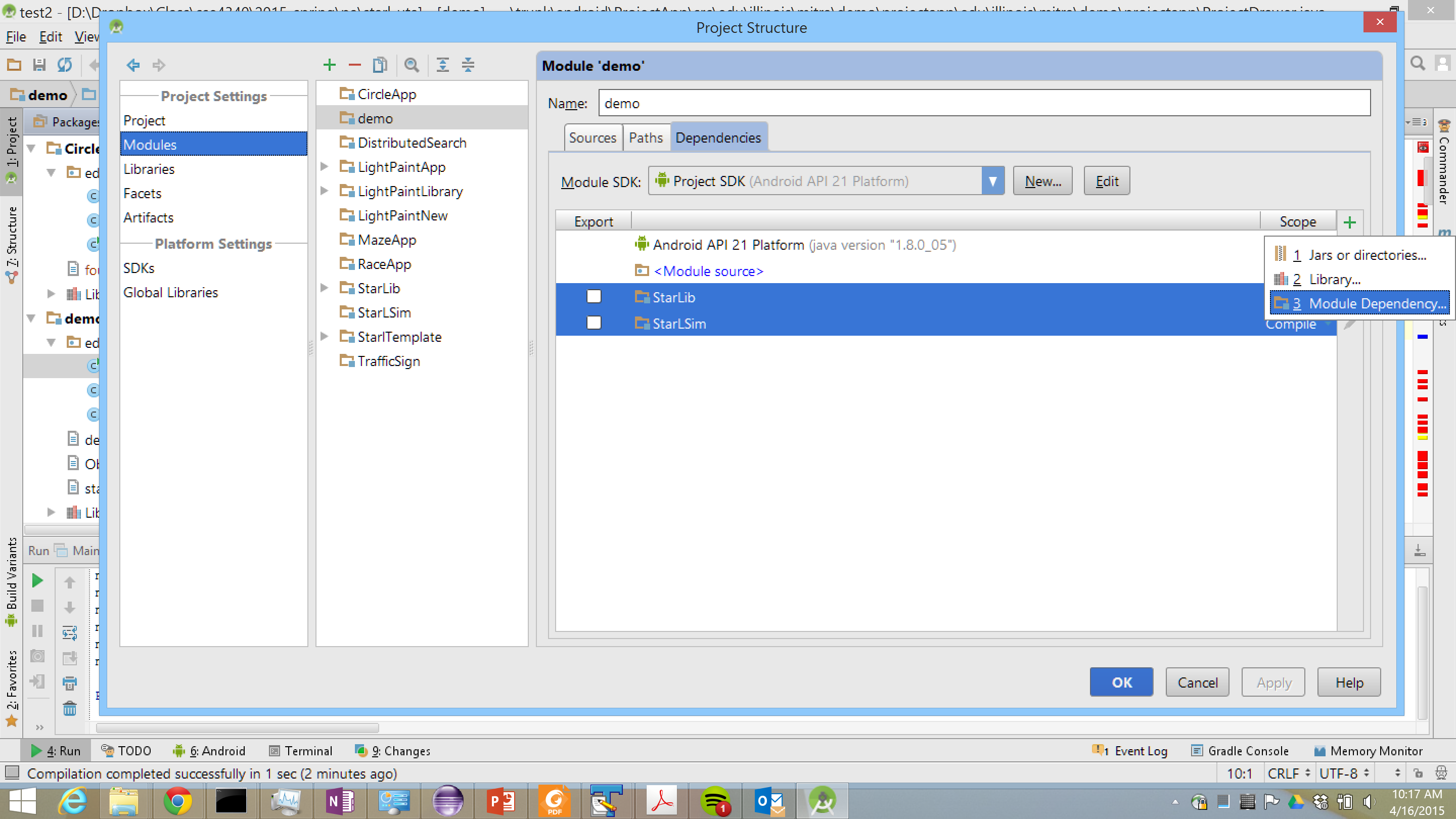
Then navigate to and select the ProjectApp folder, show below. From the root directory for the project, this will be root->trunk->android->ProjectApp.



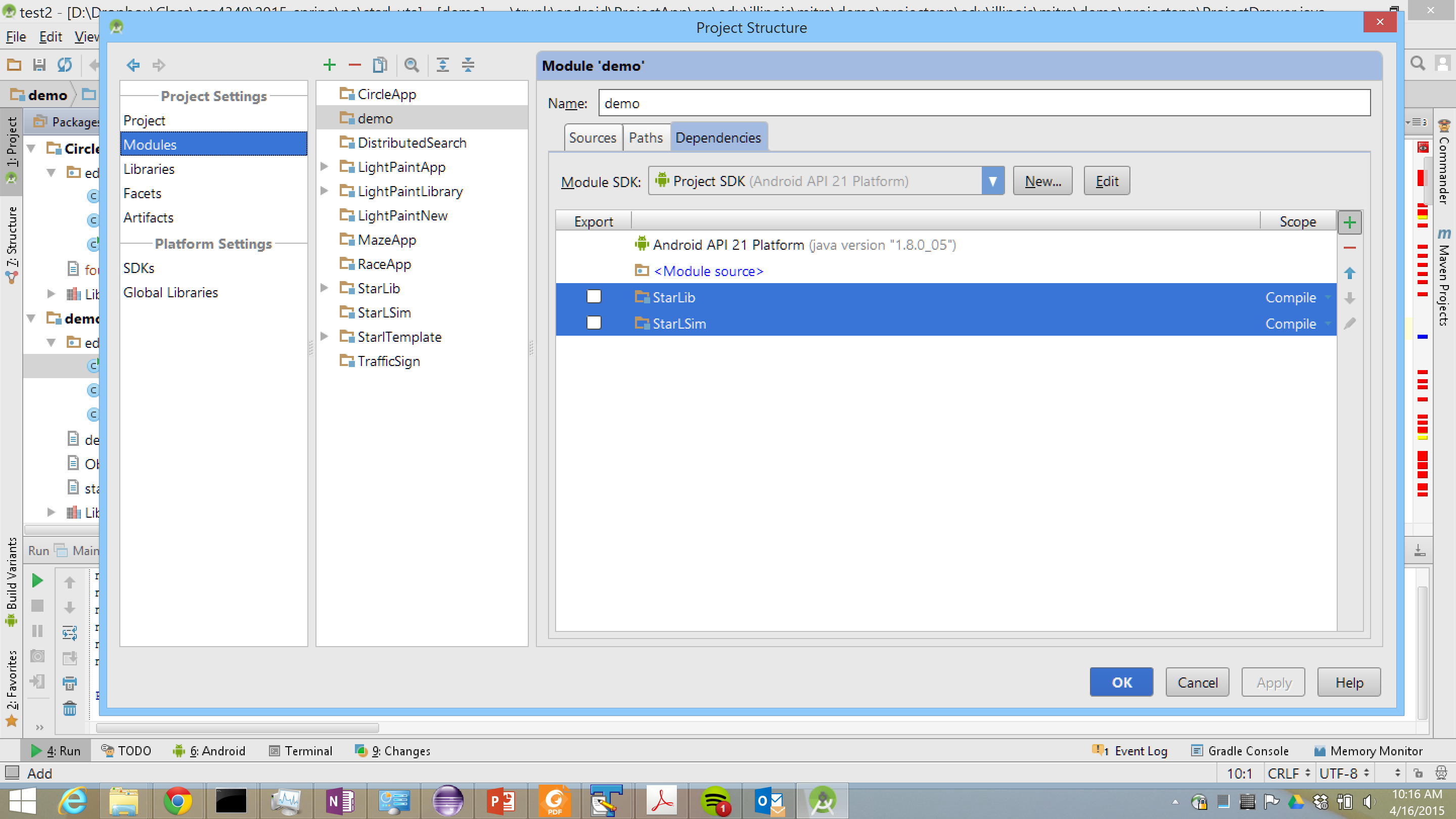
Then click on the Sources tab, select the module you just imported, select the folder named resources, and click the resources button to make the folder a resources type.



Then click the Dependencies tab and select the module you just imported. Click the green plus and select module dependencies.



You should add the StarLSim and StarLib modules so that your screen looks like the one below.



**Obstacles:**

Obstacles are listed in the obstacles.wpt file. The obstacles are defined as a list of points enclosing a polygon. For example, the following line adds a triangular obstacle with vertices (0,5000), (0,1000), and (1000,5000).

Hidden;0,5000;0,1000;1000,5000;-1

Note that you can only add horizontal and vertical lines, because listing two points not on a horizontal or vertical line draws a rectangle. You may change the environment for testing by modifying the points listed in the obstaclesl.wpt file. You should not change Hidden or the -1 at the end.

**Waypoints:**

In previous apps, waypoints were stored in a HashMap object. Waypoints will now be stored in an ArrayList object, which has HashMap elements. Each HashMap element in the array will hold one set of waypoints. Waypoints are defined in the dest.wpt file with keys 00, 01, 10, 11, 20, 21, etc. where the first number indicates the set, and the second the point within the set. There is a for loop in ProjectApp that goes through all waypoints and adds them to the proper HashMap based on the set number. The ArrayList of HashMaps holding the waypoints is called destinations.