Ritesh Misra

For Work Package 2

System Design

1. Deciding the details of how my system will communicate with the wayside controller and handle time – 1.5 hours
2. Actually meeting with Jeff to compare what we think our communication will consist of and then come to agreements (for the design documents) - 2 hours
3. Meeting with the entire group to finish up the Design documents – 3 hours

Test Plan

1. Come up with tests for my individual module – 1 hour

Updated Schedule – 1 hour

Actual programming

1. Finishing the layout of the Java GUI – 4 hours
2. Loading the data user enters into “Set Suggestion” menu into the program – 30 minutes
3. Prototyping sending suggestion to wayside controller – 1 hour
4. Testing suggestions with actual wayside controller (Properly sending and receiving status) – 2 hours

Depends on actual wayside controller code being functional to be able to test

1. Sending data of user-selected switches to wayside controller – 1 hour

Depends on actual wayside controller code being functional to be able to test

1. Making the “Show Schedule” button open up the table – 30 minutes
2. Have the schedule table be filled with actual data that I get from the wayside controller – 4 hours

Depends on being able to test with the wayside controller code

1. Program the functionality of the “Set Authority” button located on the schedule – 2 hours

Wayside controller should already be handling my initial suggestions properly by this point. Depends on wayside controller being able to properly send authority changes to the track model

1. Having “Edit Track” open up the block manager with the latest data from the wayside controller – 4 hours

Depends on the wayside controller working

1. Programming the block manager to be able to enable or disable a block based on user input (by selecting check boxes for each block) – 3 hours

Depends on the wayside controller code correctly handling input from the track model and then properly relaying data to the CTC

1. Programming an automatic mode – 10 hours

Should have no dependencies, since all of the possible outputs to the wayside controller will have been thoroughly tested by this point. The hard part is programming a computer to make safety-critical decisions at the CTC