**Description of Issues**

For the final deliverable, the only roadblocks I encountered along the way was when I was writing the java program and later trying to create the executable jar application. When beginning to write the java program, I wasn’t sure if it would have been better to use objects, as we did with the second deliverable, or if it wouldn’t make a difference if I chose to use methods. In the end, I decided use methods since it was cleaner, straight-forward to implement and since I am not completing Junit testing on the application for this deliverable, I won’t need to create objects for mocking. For this deliverable, I created the test plan prior to writing the java program which proved to be very helpful writing the java program since I had a clear outline on exactly what the java program needed to complete given each command from the user. After learning how to create an executable jar application using Eclipse, it was fairly straight-forward and didn’t take me very long to complete.

The only additional test I would use when testing this application would be more test cases for each requirement. For example, I realized after completing the test plan that the requirement TOUR\_WIN could have two test cases rather than one. I completed a test case testing paying the manager after the user has collected $100 and the second test case I could’ve completed would test paying the manager after the user has collected greater than $100. This would’ve been a valid test case, testing a single requirement, since the requirement specifically states that the user wins if the manager is paid once the balance has reached $100 or greater. There are no failed tests or problem areas with this application or test plan. All the test cases return as expected and after testing, there should be no defects. I would recommend that this product is ready to be released. There are no noticeable, or test case caught, defects and the application satisfies all the requirements listed.