**Your Name \_Liman Chang\_ Your Partner’s Name \_Kenneth Christian\_**

1. **Meeting/Communication Logs with the partner(s)**

|  |  |  |
| --- | --- | --- |
| **Time and Date** | **Agenda** | **Action Items (who will do what)** |
| 10/2 at 12:40pm | Delegate roles. | Liman will work on Transaction Manager and Date class. Kenneth will work on Account Database and the Account classes. |
| 10/6 at 12:00pm | Update on project progress. | Liman will add exception handling to Transaction Manager. Kenneth will working on fixing methods for Account subclasses. |
| 10/7 at 1:30pm | Implement JUnit tests and debug remainder of project. | Liman will implement Date JUnit test class. Kenneth will implement Checking and MoneyMarket JUnit test classes. |
| 10/8 at 11:00am | Final run through of project. | Fix whatever remaining issues are left in the program. |

1. **Your Personal Time Logs**

|  |  |  |
| --- | --- | --- |
| **Date and Time** | **Total Time in minutes** | **Tasks** |
| 9/29 at 12:00pm | 5 | Created repository for project 2. |
| 9/29 at 12:45pm | 25 | Set up class files and class methods. |
| 10/2 at 12:45pm | 70 | Coded TransactionManager and Date |
| 10/6 at 12:30pm | 60 | Debug TransactionManager and Date |
| 10/7 at 1:35pm | 90 | Implement exception handling and debug TransactionManager |
| 10/7 at 6:00pm | 60 | Implement JUnit testing for Date |
| 10/7 at 7:30pm | 90 | Debug Project 2 |
| 10/8 at 1:00pm | 60 | Started Class Diagram |
| 10/8 at 2:00pm | 95 | Debug TransactionManager and AccountDatabase |
| 10/8 at 4:50pm | 140 | Debug Accounts and AccountDatabase |
| 10/8 at 7:30pm | 30 | Finalize Documentation |
| **Total Time:** | 725 |  |

1. **Issues or Comments** (write the issues you had and how you addressed them; if no issues, write what you’ve learned or reflection)

There was no real major issue this project, the only problem that can be considered is the amount of time taken to debug the program due to the scale of the code. Since many of the methods have repeating functionalities with slight changes depending the type of account being used, debugging took up a large amount of time as issues can overlap across multiple methods and classes due to the nature of the program. In reflection, while making the class diagram, it made me wish that the program could be streamlined more as the multitude of methods made the program appear more bloated then when we were initially writing it. Seeing how every method and class related to each other in the diagram made it such that I wished we approached the diagram first so that during planning, we could better structure the program.