ATM Simulator Program Report

Introduction

Automated Teller Machines (ATMs) are machines used by banks to allow customers to withdraw money from their bank accounts. Problems can arise when multiple people attempt to make a transaction from the same account. This can lead to inconsistencies in the balance of a bank account. To demonstrate this, we have created a simulation using multi-threading in C# to show the situation with this problem and without this problem to gain an understanding of how this situation should and should not occur.

Program Features

Our program starts with a ‘launch’ GUI that allows the user to select the simulation they want to start and the number of ATMs they want to use during the simulation. Once the user clicks the start simulation button, they will receive one window for the bank main computer and ATM windows depending on how many the user asked for. The user can then interact with the ATM windows to perform the basic functionality of a typical ATM. If the user wants to exit the program, they can select the ‘Menu’ tab and click Exit.

Our Approach

We began the project with great urgency. We quickly got together for a team meeting through Discord. We laid out everything we needed to do for this project in this meeting, creating a class diagram to give us a clear vision of what our program would look like. This meeting was a great way for us to start the project as it allowed everyone in the team to gain a clear understanding of the work needing to be done. As time went on though, we did not communicate with each other for a long time. This led to confusion and disarray within the team as not everyone was on the same page. At one point, a team member was working on something that had already been finished. To prevent this happening again, we started to organise frequent meetings where we would delegate more clearly and ask on others progress to see if they needed any assistance. We immediately saw an improvement to our workflow as more tasks were being complete daily.

It was also said during our initial meeting that we would use GitHub to store all our work. Having all had some experience with GitHub it made sense to use it. It made access to our files very painless and easy. For the most part GitHub was very useful and great to use, it allowed everyone to be able to access our work. We did encounter some problems during project development. The biggest problem we encountered with GitHub was keeping up to date with the most recent files. Sometimes team members would take a break and forget to update their local repositories which led to them working on files that were redundant. This wasted a lot of our time since any work that was done on out of date files was useless to us and thrown away. To mitigate this problem, we started to remind each other to update their local repositories. Once we all understood how much this was affecting or progress, we started getting into the habit of updating every time we logged on. It dramatically improved our progression as we had more time to use working on other tasks.

Conclusion

We believe our simulation provides an accurate representation of the project requirements. Problems that we encountered during the development of the project were dealt with to ensure that they never came up again during future development. Both these can be attributed to the hard work carried out by everyone involved with this project.

Word Count: 599