# Implementation of Templates

I already have a template from the last phase, regarding the overdraft limit. I don’t like templates, maybe in the future that’ll change when I use them more but right now, I don’t like them.

# Robust Exception Handling

I implemented try-catch blocks in the withdraw and deposit functions of the account classes. I think this was a good chance, because the code is now easier to read in my opinion and it doesn’t feel as cluttered. The currently handled exceptions are just with insufficient balances or an otherwise invalid input to the system.

# Documentation and Explanation

The rationale for catching errors is to provide a sophisticated output. Why else do you catch errors? There were no challenges faced, but otherwise it’s something I do a lot with my other programming languages, and I’m happy I could finally do it with C++.

# Testing and Validation

I tested both account types and they both functioned as expected, responding to the new error handling.

# Reflection and Future Improvements

I want to try and incorporate templates more, if possible, but I can’t find anywhere to best put them. The rest of the document is the requested output images:

A screenshot of a computer screen

Description automatically generatedA black screen with white text

Description automatically generated