Laynie Tierney

CS-405-R4888 Secure Coding

3 April 2024

4-1 Activity: Exceptions

A screenshot of a computer program

Description automatically generated

In the process of addressing the abrupt termination issue in the banking web application, I In the process of addressing the abrupt termination issue in the banking web application, our team undertook a systematic approach to identify, analyze, and resolve exceptions within the codebase. Initially, we recognized the need to address the issue of the application crashing without providing any warnings or information to the user. This indicated potential unhandled exceptions occurring during runtime, which required immediate attention to ensure the stability and usability of the application.

To diagnose and address the issue, the software developers conducted a thorough analysis of the codebase. Through collaborative efforts, they traced down the problem to suspect code segments but decided against making direct changes to the code. Instead, the team opted to implement exception handling mechanisms to capture errors and display informative error messages to users, thus enhancing the application's resilience to unexpected errors.

The implementation of exception handling involved the use of try-catch blocks to wrap code segments that could potentially throw exceptions. Special consideration was given to catching exceptions derived from std::exception, as well as custom exceptions specific to the application's requirements. Additionally, provisions were made to handle divide-by-zero errors and other exceptional scenarios gracefully, ensuring a more robust and user-friendly application.

Following the implementation phase, rigorous testing was conducted to validate the effectiveness of the exception handling mechanisms. Various test cases were executed to trigger different types of exceptions, allowing developers to observe and analyze the behavior of the application under exceptional circumstances. Through thorough testing and debugging, the team ensured that exceptions were caught and handled appropriately, thereby mitigating the risk of application crashes and improving overall user experience.

In documenting the exceptions testing process, a static testing summary was prepared to provide a comprehensive overview of the steps undertaken and the outcomes achieved. This summary included detailed descriptions of the exceptions encountered, along with their respective error messages, to facilitate better understanding and future reference. Overall, the systematic approach adopted by our team helped address the exception-related issues effectively, resulting in a more stable and reliable banking web application for users.