

SENG 438 Lab 1 Report



B01

Nicole Heather: 30163278

Jennifer Jay: 30115782

Nora Mellik: 30148251

Nelson Thompson: 30163519

```
> **SENG 438 - Software Testing, Reliability, and Quality**
```

```
**Lab. Report \#1 - Introduction to Testing and Defect Tracking**
```

```
| Group: 23 |
|-----|
| Nora Mellik |
| Nicole Heather |
| Nelson Thompson |
| Jennifer Jay |
```

```
**Table of Contents**
```

(When you finish writing, update the following list using right click, then "Update Field")

```
[1 Introduction 1](#Introduction)
```

```
[2 High-level description of the exploratory testing plan 1](#High-  
level description of the exploratory testing plan)
```

```
[3 Comparison of exploratory and manual functional  
testing 1](#Comparison of exploratory and manual functional testing)
```

```
[4 Notes and discussion of the peer reviews of defect  
reports 1](#Notes and discussion of the peer reviews of defect reports)
```

```
[5 How the pair testing was managed and team work/effort was divided  
1](#How the pair testing was managed and team work/effort was divided)
```

```
[6 Difficulties encountered, challenges overcome, and lessons  
learned 1](#Difficulties encountered, challenges overcome, and lessons learned  
)
```

```
[7 Comments/feedback on the lab and lab document itself  
1](#Comments/feedback on the lab and lab document itself)
```

```
[5 How the pair testing was managed and team work/effort was  
divided 1](#_Toc439194681)
```

```
[6 Difficulties encountered, challenges overcome, and lessons  
learned 1](#_Toc439194682)
```

[7 Comments/feedback on the lab and lab document itself 1](#Toc439194683)

Introduction

Testing is a critical aspect of software engineering; it helps to ensure the quality and efficiency of a product while also providing a pathway of communication between developer and client. The purpose of this lab is to explore the practical procedures and introduce proper bug reporting in a team environment. This group's knowledge of exploratory and manual testing was purely theoretical before the execution of this lab, but, after, we now have a better understanding of how testing plans operate in real development environments. There are also differences in the procedure that we did not think of before, such as the more humanlike interaction in the system in exploratory testing, compared to the heavily standardized manual testing.

High-level_descriptionof_the_exploratory_testing_plan

Test Strategy:

We wanted to focus our attention on the functionalities that are the most used and if the system could handle incorrect inputs, as these kinds of inputs are a majority of what the ATM must deal with. These inputs should mostly be sequential, unless a bug causes the system to become unresponsive. For this specific system there is also a lot for the program to keep track of, such as the total balance and amount available. It is important to test for those values being correct, while also recording them in the receipt and log.

Features To Be Tested:

Functionality	Actors/Users	Description
Account Withdraw	Customer	Customer: Can withdraw from a chosen account
Account Transfer	Customer	Customer: can transfer from one account to another
Account Deposit	Customer	Customer: Can deposit from a chosen account
Account Inquiry	Customer	Customer: Can see the total balance and amount available for a certain account
Card/Pin Authentication	Customer	Customer: A customer can re-enter their pin or card number if incorrect
Transaction Canceling	Customer	Customer: can cancel transaction, then either continue or quit ATM

Receipt Retrieval	Customer	Customer: can take a receipt that has all transactions up until that point
-------------------	----------	--

The functionalities not accounted for are:

- User interface functionality
- Hardware Interface
- Database Logic
- Communication interface
- Security
- Performance

Test Type: We are purely using exploratory testing for this lab.

Risk and Issues:

Risk	Mitigation
The project schedule could be too strict	Ensure open communication to possibly reschedule or re-assign tasks
Lack of knowledge about Azure	Take the time to educate team on the functionalities and benefits of Azure

Who is testing?

The main four members of this lab group will complete this testing together. There are no cost or time constraints within this lab.

When will the tests occur?

This group took about 2 hours to plan and organize our exploratory testing before execution. After we took about 30 minutes to fully examine and record everything.

Test Objective:

The purpose of this examination is to ensure that the system is efficient, accurate, and provides all the functionalities specified in its requirements. But to also become familiar with test planning and execution at an industry level.

Test Criteria:

There is no specific suspension or exit criteria, only that we stop our testing after 30 minutes in accordance with the procedure of this lab. These testing processes are merely a way for us to be familiarized with standard industry practices.

Resource Planning:

System Requirements:

No.	Resources	Description
-----	-----------	-------------

1.	Computer	Needs to be able to download and run both applications. No other requirements
2.	Azure	Needs to be able to accurately maintain the information we provide about our bugs
3.	Microsoft Excel	Needs to be understandable and facilitate cooperation across team members

Human Requirements:

No.	Resources	Description
1.	Test Manager	Manages and directs the process of exploratory testing.
2.	Test Executer	Performs the actual test cases and records their progress, also reporting the bugs in Azure

Test Environment:

The testing environment remains relatively simple for the purposes of this lab. For the exploratory section there are two pairs working on one computer each. They both run the application with the previously discussed plans.

Schedule and Estimation:

All Project Tasks and Execution:

Task	Members	Estimate Effort
Perform Test Execution	Test Manager, Test Executors	4 man - 1/2 hour
Test Reporting	Test Manager, Test Executors	4 man - 36 hours
Total		4 man - 36 ½ hours

Schedule to Complete Tasks:

Task Name	Start Date	End Date
Perform Test Execution	Thursday, Jan 25 2024	Sunday, Jan 28 2024
Milestone	Monday, Jan 29 2024	Monday, Jan 29 2024
Test Reporting	Monday, Jan 29 2024	Thursday, Jan 30 2024
Milestone	Thursday, Jan 30 2024	Thursday, Jan 30 2024

Test Deliverables:

Before testing phase:

- Create testing plan, cases and specifications.

During testing phase:

- Testing data and report log.

After testing phase:

- Review and comparison between both application versions. Also continuing to create lab report.

Comparison_of_exploratory_and_manual_functional_testing

Exploratory testing provided a more freeform and adaptable procedure to test the system. We were able to identify bugs from all different angles of this program, including errors with incorrect inputs and with normal input. Exploratory testing in execution was less time consuming, but slightly less efficient than the structured approach of manual testing. Manual testing provides a far deeper exploration of the functionalities focused on, keeping track of specific system states and their input.

Notes_and_discussion_of_the_peer_reviews_of_defect_reports

Due to the way we tested these applications, many of the bugs reported were already discussed and recorded as a group. This helped make communication clear and understandable, but we did use Azure as the basis for regression testing. We also noticed that including detail in the steps was crucial. We often would go back to tests in both exploratory and manual to see if we could replicate the errors or successes to confirm the results of our testing.

How_the_pair_testing_was_managed_and_team_work/effort_was_divided

Pair tested was managed by splitting larger tasks into two. Exploratory was done simultaneously with each pair, it was also timed to ensure that an equal amount of time was given for each section. We also had each pair test different functionalities: common paths or edge cases. For manual, two people lead v1.0 and another pair lead v.1.1. Keeping the leaders for the versions different made sure that every student took part in the recording and leading process. Eventually after the completion of both testing plans, we came together as a group to discuss the results.

Difficulties_encountered,_challenges_overcome,_and_lessonslearned

The most difficult parts of this lab were the execution of exploratory testing and learning the process of proper bug reporting. Exploratory testing and its lack of clear structure sometimes made it difficult to determine if the testing we did was enough to demonstrate an understanding of the actual procedure. We overcame this challenge by dividing some of the functionalities between each pair and allowing them to create a smaller testing procedure that would eventually combine with the other pair. The other challenge was only an issue due to the new technology used, none of us had ever used Azure before so there was a slight learning process in how it was supposed to function which did delay some work.

Comments/feedback_on_the_lab_and_lab_document

itself

We do think that the lab was a good learning experience and introduction to the professional testing and bug reporting procedure. Although a small walkthrough of Azure or Atlassian would have likely been useful, also would have made clear what our reports should have looked like in that form. The other comment would be the structure of the lab document itself. We think that the inclusion of questions to answer is very helpful in providing clear structure, but it is strange that the report cannot be done in Word or Google Docs, as these platforms provide tools to make formatting and writing clearer and easy to edit. However, these are not detrimental issues.