Verification and Validation Report: Plutos

Team #10, Plutos
Payton Chan
Eric Chen
Fondson Lu
Jason Tan
Angela Wang

March 10, 2025

1 Revision History

Date	Version	Notes
Date 1	1.0	Notes
Date 2	1.1	Notes

2 Symbols, Abbreviations and Acronyms

Refer to Section 1.3 of the Software Requirements Specification (SRS) document for the list of symbols, abbreviations, and acronyms.

In addition, the following abbreviations are used in this document:

Table 1: Symbols, Abbreviations, and Acronyms

symbol	**description**
V&V	Verification and Validation
UI	User Interface
OCR	Optical Character Recognition
SQL	Structured Query Language
GDPR	General Data Protection Regulation

Contents

1 Revision History i

2 Symbols, Abbreviations and Acronyms ii

3 Functional Requirements Evaluation 1

4 Nonfunctional Requirements Evaluation 2

5 Comparison to Existing Implementation 3

6 Unit Testing 4

6.1 Front-end unit tests 4

7 Changes Due to Testing 4

8 Automated Testing 4

9 Trace to Requirements 4

10 Trace to Modules 4

11 Code Coverage Metrics 4

List of Tables

1 Symbols, Abbreviations, and Acronyms ii

2 Functional Requirements Evaluation 1

3 Nonfunctional Requirements Evaluation 2

4 Unit Testing Table 6

List of Figures

This document reports the results of the Verification and Validation (V&V) process for the Plutossoftware. The V&V plan is documented in the Verification and Validation Plan document.

3 Functional Requirements Evaluation

The functional system tests can be found in Section 4.1 of the Verification and Validation Plan document. These tests are all performed manually.

Table 2: Functional Requirements Evaluation

Test ID	Pass/Fail	Comments
test-UAM-1	Pass	
test-UAM-2	Pass	
test-UAM-3	Pass	
test-UAM-4	Fail	Not yet implemented
test-UAM-5	Pass	
test-UAM-6	Pass	
test-IP-1	Pass	
test-IP-2	Pass	
test-IP-3	Fail	Not yet implemented
test-MIS-1	Pass	
test-DM-1	Pass	
test-DM-2	Pass	
test-RS-1	Pass	
test-RS-2	Pass	
test-RS-3	Pass	
test-FT-1	Fail	Not yet implemented
test-FT-2	Pass	
test-FT-3	Fail	Not yet implemented

4 Nonfunctional Requirements Evaluation

The nonfunctional system tests can be found in Section 4.2 of the Verification and Validation Plan document. **Tests without comments are performed as described in the plan.**

Table 3: Nonfunctional Requirements Evaluation

Test ID	Pass/Fail	Comments
test-ACC-1	Pass	Actual output; accuracy is 47/57 = 82.46%, which meets the threshold of 80%. Test can be found here.
test-ACC-2	Pass	By manually comparing the input (set of receipt images) with the resulting output, and calculating the accuracy as descripted in the V&V Plan, current accuracy is 80%, which meets the threshold of 80%. • *foodbasics_1.jpg*: 13.5/15 = 90% • *foodbasics_2.jpg*: 7/9 = 77.78% • *walmart_1.jpg*: 7/10 = 70% • *costco_1.jpg*: 18.5/23 = 76.09% • Overall accuracy: 46/57 = 80.70%
test-ACC-3	Pass	
test-ACC-4	Pass	
test-ACC-5	Pass	
test-PERF-1	Pass	
test-PERF-2	Pass	
test-PERF-3	Fail	Load testing has not yet been performed

test-USAB-1	Pass	
test-USAB-2	Pass	
test-USAB-3	Pass	
test-USAB-4	Pass	
test-SEC-1	Pass	
test-MTB-1	Pass	System stability has been tested, but application is not backward compatible since it is still under active development.
test-MTB-2	Pass	
test-MTB-3	Pass	
test-PORT-1	Pass	
test-PORT-2	Pass	
test-PORT-3	Pass	
test-REUS-1	Pass	Code walkthrough/review was performed with the team. See meeting minutes.
test-REUS-2	Pass	See REUS-1
test-UND-1	Pass	See REUS-1
test-UND-2	Pass	
test-UND-3	Pass	
test-LEGAL-1	Fail	The application is still under active development, so it is still using the testing environment and not all security features are active.

5 Comparison to Existing Implementation

This section will not be appropriate for every project.

6 Unit Testing

6.1 Front-end unit tests

All front-end unit tests can be found in the test directory
Refer to Table 4 for unit test traceability table

7 Changes Due to Testing

[This section should highlight how feedback from the users and from the supervisor (when one exists) shaped the final product. In particular the feedback from the Rev 0 demo to the supervisor (or to potential users) should be highlighted. —SS]

8 Automated Testing

9 Trace to Requirements

10 Trace to Modules

11 Code Coverage Metrics

References

Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Reflection.

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. What went well while writing this deliverable?

2. What pain points did you experience during this deliverable, and how did you resolve them?

3. Which parts of this document stemmed from speaking to your client(s) or a proxy (e.g. your peers)? Which ones were not, and why?

4. In what ways was the Verification and Validation (VnV) Plan different from the activities that were actually conducted for VnV? If there were differences, what changes required the modification in the plan? Why did these changes occur? Would you be able to anticipate these changes in future projects? If there weren't any differences, how was your team able to clearly predict a feasible amount of effort and the right tasks needed to build the evidence that demonstrates the required quality? (It is expected that most teams will have had to deviate from their original VnV Plan.)

Test	**Testing plan**
test-UAM-1: Account creation	
test-UAM-2: User login	
test-UAM-3: User logout	
test-UAM-4: Account update	
test-UAM-5: Authorization access	
test-UAM-6: Password reset	Manual testing
test-IP-1: Image upload	Manual testing
test-IP-2: Image preview	
test-IP-3: Image upload file size limit	
test-MIS-1: Manual input expense	AddExpenseView.test.tsx, AddExpenseModal.test.tsx
test-FT-1: View spending history and trends	ExpensesList.test.tsx, HomePageMetricsBox.test.tsx, SpendingDetails.test.tsx
test-FT-2: Set and track budget	BudgetBoxDetails.test.tsx, MyBudgetsBox.test.tsx, NewBudgetModal.test.tsx
test-FT-3: Notification when user approaching limit	Not implemented

Table 4: Unit Testing Table