California State University, Sacramento
College of Engineering and Computer Science
CSC 131-05: Computer Software Engineering

SOFTWARE VALIDATION DOCUMENT

(Verification & Testing Phase - Waterfall Methodology)

for

PROJECT BRAVO

Prepared By: Sparsh Saraiya,Mukhammad Abdusamadov

Table Of Contents

SACRAMENTO STATE

CSC 131-05: Computer Software Engineering

Project Bravo: Software Validation Document

Chapter 1: Introduction

1.1 Purpose & Scope:

The goal of this document is to outline the structured testing and validation process for the **Expense Tracker** desktop application. Testing is where we measure the systematic quality through different processes that check whether the software solution contains and satisfies all functional and non-functional requirements provided in the Software Requirements Specification (SRS). The scope of this document is limited to unit testing, integration testing, validation testing, and system testing. Each level of testing verifies that the system functions correctly, both at the component and system levels, confirming that the **Expense Tracker** operates correctly and meets end user expectations. This document matches the verification testing

1.2 Objectives of Testing:

The main goal of testing is to confirm that the **Expense Tracker** software works correctly, efficiently and according to specification. Testing is intended to confirm that each individual part is functioning properly (verification), and that the overall system meets the requirements specified (validation). It is also a way of assessing usability, reliability and performance across the full range of the application (data entry, reporting, administration etc). Testing will provide a basis to find bugs early, confirm that all functions work accurately, and validate the overall quality of software as according to the Software Specification document.

Chapter 2: Testing Methods and Approach

2.1 Unit Testing:

Unit tests were conducted on every major component of the application to ensure that every feature functioned correctly on its own. Specific windows such as Add Entry, View Summary, and Admin Login were tested to verify all functions performed correctly and without errors. Also, each module was tested with valid and invalid inputs to verify correct calculations, data handling, and to ensure the accuracy of displayed error messages prior to integration testing.

2.2 Integration Testing:

Integration testing was done using the bottom-up approach where smaller modules of the Expense Tracker application were integrated and tested separately. The data handling and database module was integrated first to ensure that financial entries could be saved and retrieved properly. Once it was confirmed that the lower level modules could store and retrieve financial data reliably, the lower level modules were then integrated with the lower level interface windows, such as Add Entry and Summary, to check the correct passage of values back and forth from the lower-level modules to the interface windows. All modules interacted properly together without errors at this point, so higher level features (the Administrative Dashboard module) were integrated and tested. Each phase of the integration confirmed that the modules worked together appropriately and that no new issues would be introduced by combining modules from different levels of the integration model.

2.3 Validation Testing:

Validation testing needs to be conducted to ensure that the Expense Tracker application was built following all of the requirements outlined in the Software Requirements Specification (SRS) document. This part of the process focused on making sure the software would successfully execute the tasks found in the SRS document and provide the end user with the expected results. Here, team members acted in the role of an end user and evaluated the system in a realistic scenario. Examples include adding several income entries and expense reports, checking the summaries of categorized expenses, and generating monthly reports. The administrative function of the application was validated by logging in with the test administrator log-in credentials to verify that an administrator could view, edit and generate reports as intended. Each test validated that the user outputs confirmed the requirements within accuracy, reliability, and ease of use. The application functioned without bugs in displaying correct financial summaries, classification of expense entries was done correctly, and invalid entry criteria would be handled as expected. The results from the validation testing confirmed that the Expense Tracker application meets all key functional and non-functional requirements made within the SRS.

2.4 System Testing:

System testing was done to test the **Expense Tracker** as a fully functioning system. This phase checked everything from the user windows to data storage and administrative roles worked well together. The testing team ran the program on standard desktop computers and performed normal tasks such as adding income and expense entries, creating reports, and switching back and forth between multiple open windows. The purpose of these tests was to check if the software solution remained stable, responsive, and functioning as intended. More tests were done using larger amounts of sample data to see how the system would manage larger data workloads. The solution continued to provide correct results and remained responsive to actions taken by the development team during these tests. All the main functionalities worked properly, and no crashes or major failures were noted. The Expense Tracker proved to be stable, dependable, and suitable for final validation and submission.

Chapter 3: Test Cases & Results

3.1 Summary of Test Cases:

The testing process focused on testing and validating both the functional and non-functional requirements of the Expense Tracker Application to check each function worked as intended and delivered the expected results. The testing process broadly focused on income and expense tracking, editing/deleting data, generating reports, and administrative roles. Each test focused on the software solution's ability to handle both valid and invalid data, as well as accurately present financial summaries and visuals if needed.

3.2 Test Results Table:

Test Number	Functionality Tested	Expected Result	Actual Result	Test Result
1	Add Income Entry	Entry is saved correctly and displayed in the table	Works as expected	Pass
2	Add Expense Entry	Expense recorded/categorized	Works as expected	Pass

3	Edit Existing Entry	Updated information replaces old values	Works as expected	Pass
4	Delete Entry	Selected item is permanently removed	Works as expected	Pass
5	Generate Monthly Report	Report shows accurate totals and charts	Works as expected	Pass
6	Admin Login	Access granted only with correct credentials	Works as expected	Pass
7	Invalid Input	Proper error handling	Works as expected	Pass
8	User Login	Login succeeds/fails properly	Works as expected	Pass
9	Filter Transactions	Correct filtered results	Works as expected	Pass
10	Contact form	Form submits/validates	Works as expected	Pass
11	Navigation + Icons	Links function correctly	Works as expected	Pass

Chapter 4: Validation Summary

4.1 Evaluation of Results:

In every phase of testing, the Expense Tracker application behaved and operated according to its expected functionality and non-functional requirements. Each feature of the application worked as designed, whether that involved entering data, editing, or generating reports and summaries. The application verified user input, displayed and responded to errors in a reasonable manner for all associated windows, and every calculation displayed correctly and consistently. The link between both user and administrator portions of the application functioned as intended. Also, the data reflected in the database aligned with what is shown in the application.

4.2 Confirmation of Requirements

The final software solution follows and completes all the requirements in the Software Requirements Specification (SRS) document. Each of the major features includes income and expense tracking, report generation, and access to the administrative side, has been tested and confirmed as working. The system has met non-functional performance and reliability goals. Since all tests we planned for have completed with passing results, the project has been validated to be working as expected and ready for final submission.