



K. J. Somaiya School of Engineering
Department of Computer Engineering

Batch: A-4 Roll No.: 16010122147

Experiment No : 04

Group No: 3

Title: Design of Test Cases

Expected Outcome of Experiment:

CO2: Implement and prototype creation for the specified application.

Books/ Journals/ Websites referred:

[Students can mention websites/ books used in their project implementation]

This write up will expect students to prepare chapter no. 4 in the format given below

Chapter 4

Design Test Cases

This chapter details the testing conducted mentioning the various test scenarios considered. This chapter in a software development project report should provide a comprehensive overview of the testing process, methodologies, and results. The purpose of a test case document is to provide a detailed, structured record of the steps and conditions needed to verify a specific functionality or feature in software testing, ensuring that the application functions correctly and meets requirements

Introduction:

Developing design test cases is crucial for identifying defects early, ensuring software quality, and validating user requirements, ultimately leading to a more reliable and user-friendly product.

Testing of a software system is a very essential step which provides benefits such as:

- **Early Defect Detection:**

Thoroughly designed test cases help uncover defects early in the development lifecycle, minimizing costly rework and delays.

- **Enhanced Software Quality:**

By systematically testing all features and functionalities, test cases ensure that the software meets user expectations and performs as intended.

- **Improved User Satisfaction:**

A well-tested product with fewer bugs leads to a better user experience and increased satisfaction.

- **Reduced Costs:**

Identifying and fixing defects early in the development process is significantly cheaper than fixing them later, saving time and resources.

- **Validation of Requirements:**

Test cases verify that the software meets the defined requirements and functionalities, ensuring that the product delivers on its promises.

- **Increased Confidence:**

Thorough testing builds confidence among stakeholders, including customers, investors, and regulatory bodies, regarding the software's quality and reliability.

- **Optimized Testing Effort:**

Well-designed test cases ensure comprehensive test coverage, minimizing redundancy and maximizing efficiency.

- **Facilitates Bug Detection:**

Test cases help identify when something in the software isn't working correctly, allowing bugs to be fixed before the software is released.

- **Ensures Coverage:**

They help to ensure that every feature of the software is tested and that all the requirements are covered.

- **Traceability:**

Test cases can be traced back to the requirements they are testing, providing a clear link between the requirements and the testing process.

- **Test Scenarios:**

Test scenarios are useful for organizing your test cases and providing high-level information on what to test in your application.

- **Test Coverage:**

Test coverage reports on areas of the codebase that are not covered by the test cases, facilitating the creation of new test cases to ensure maximum test coverage.

Students will be required to perform tests (whichever are applicable) from the below list:

1. **Validation of data:**




Checking if the entered values are in the correct format and within the range etc.




2. **Appropriate Navigation**




Accessing the various pages as per the flow of the system



3. **Verification of the results generated using the system.**

Check if the results are correct

Sr. No	Test Case Description	Prompt	Expected Output	Got Output	Pass/Fail	Screenshot
1	Verify image generated for valid prompt	A dog flying a spaceship	Image relevant to prompt	Image matched ~90%	Pass	
2	Verify different prompts give different outputs	Prompt A: A mountain during sunset Prompt B: A futuristic city underwater	Two distinct images	Different images seen	Pass	 

3	Image consistency for repeated prompt	A wizard riding a dragon	Similar but not identical images	Unique outputs with similar style	Pass	  
---	---------------------------------------	--------------------------	----------------------------------	-----------------------------------	------	---

4	Verify prompt case insensitivity	a cat wearing glasses / A CAT WEARING GLASSES	Similar images regardless of case	~95% similar outputs	Pass	 
5	Check response to ambiguous prompt	Time	Image reflecting concept of "time"	Clock-related visual	Pass	

6	Prompt with multiple elements	A panda playing guitar on the moon	All elements shown in one image	Panda, guitar, and moon visible	Pass	
7	Special character handling in prompt	cat & dog	Image showing cat and dog together	Two animals, with cat face on dog body	Pass	

Conclusion:

The experiment successfully validated the functionality and reliability of the AI Image Generator through structured test cases. It ensured that the system meets user expectations by producing accurate and meaningful image outputs for various prompt types.