**User Acceptance Testing (UAT) – SmartSDLC**

**Date**: 25 JUNE 2025  
**Team ID**: LTVIP2025TMID33777  
**Project Name**: SmartSDLC  
**Maximum Marks**: 10

**Project Overview**

* **Project Name**: SmartSDLC
* **Project Description**:  
  SmartSDLC is an intelligent, AI-powered tool that automates the classification of software engineering documents into SDLC (Software Development Life Cycle) phases. By leveraging Generative AI models (simulated using Google Gemini, targeting IBM Granite in future phases), SmartSDLC extracts text from uploaded PDF documents, classifies paragraphs into appropriate SDLC stages like Requirements, Design, Implementation, Testing, and Maintenance, and presents the results through a clean, interactive Streamlit interface. This solution assists students, developers, and educators by reducing manual effort and providing clear, phase-wise documentation insights.

**Project Version: 1.0**

**Testing Period: 20 June 2025 – 22 June 2025**

**Testing Scope**

**Functionalities to be Tested**:

* **PDF Upload and Validation**: Verifying file type restrictions and feedback for unsupported formats.
* **Text Extraction**: Accurate parsing of content using PyMuPDF from multipage documents.
* **AI-Based SDLC Classification**: Correct assignment of text blocks to SDLC phases.
* **Phase-Wise Output Display**: Proper tagging and visualization of each paragraph with its SDLC phase.
* **Downloadable Summary**: Generating a downloadable report of SDLC-classified content.
* **Application Stability**: Tab transitions, error handling, and consistent performance within user sessions.

**User Stories to be Tested**

* As a user (Rayapudi), I want to upload an SRS or Design Document PDF and receive clear SDLC phase classification.
* As a user (Harika), I want to verify if AI correctly identifies each paragraph's SDLC stage.
* As a user (Ram Prasad), I want to view an intuitive, color-coded display of my document’s classified structure.
* As a user (Giridhar), I want to download a categorized summary of my document to use for review or project reporting.

**Testing Environment**

* **URL/Location**: Localhost (via streamlit run app.py)
* **Credentials**: Not Required
* **Browser/OS**: Google Chrome (latest) on Windows 10 Pro

**Test Cases**

**TC-001 | PDF Upload Validation**  
**Steps**:

* Navigate to the Upload tab.
* Upload both a valid PDF and an invalid file type (e.g., .jpg).  
  **Expected**:
* PDF is accepted; other types trigger an error.  
  **Actual**:
* Valid PDF accepted, invalid file rejected with message.  
  **Result**: ✅ PASS

**TC-002 | Content Extraction Accuracy**  
**Steps**:

* Upload a multi-page software document.
* View extracted content in the raw display.  
  **Expected**:
* All text is extracted correctly and split into paragraphs.  
  **Actual**:
* Text is parsed and displayed correctly.  
  **Result**: ✅ PASS

**TC-003 | SDLC Classification (AI-Based)**  
**Steps**:

* After uploading, click “Classify” and observe phase tagging for each paragraph.  
  **Expected**:
* Each section is labelled with SDLC phases and confidence level.  
  **Actual**:
* Classification is accurate, and confidence is clear.  
  **Result**: ✅ PASS

**TC-004 | Download Phase-wise Summary**  
**Steps**:

* Click “Download Summary” after classification.  
  **Expected**:
* File downloaded with content grouped under SDLC phase headers.  
  **Actual**:
* Download successful and content properly organized.  
  **Result**: ✅ PASS

**TC-005 | Session State Consistency**  
**Steps**:

* Update input, navigate between tabs, and return.  
  **Expected**:
* User session data (input, classification) remains persistent.  
  **Actual**:
* Inputs persist correctly across tabs.  
  **Result**: ✅ PASS

**Bug Tracking**

**BG-001 | Delay in Classification with Large Documents**

* **Steps**: Upload 20+ page documents for classification.
* **Severity**: Medium
* **Status**: Under Optimization
* **Feedback**: AI takes longer to respond. Future improvements include content chunking or background processing.

**Sign-off**

* **Tester Name**: Rayapudi D V K Ramya Sri
* **Date**: 22 June 2025
* **Signature**: *Rayapudi D V K Ramya Sri* (electronic)

**Notes**

* All major functionalities were tested and validated under typical usage conditions.
* Gemini API was used as a simulation for IBM Granite, and the behavior aligns with project expectations.
* Only one medium-severity bug noted, with mitigation planned.
* SmartSDLC is ready for the next development phase involving persistent storage, model fine-tuning, and deployment on Streamlit Cloud or a private server.