CEBU INSTITUTE OF TECHNOLOGY UNIVERSITY

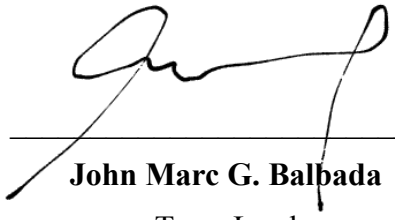
COLLEGE OF COMPUTER STUDIES

Software Test Document

for

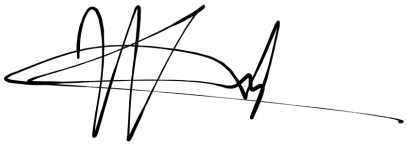
TechnoDynamic V2: AI-Enhanced Dynamic Content for Technopreneurship

Signature



John Marc G. Balbada
Team Lead

Jason Carlo O. Obiso
Team Member



Winson C. Baring
Team Member

Change History

Table of Contents

1. Introduction

1.1. System Overview

TechnoDynamic V2 is an AI-enhanced learning management system aimed at improving Technopreneurship courses by providing teachers with content recommendations based on insights gathered from student FAQs. The system enhances teaching efficiency and optimizes content by addressing common student inquiries and areas of confusion.

1.2. Test Approach

The system will undergo functional and non-functional testing. Each module will be tested individually using black-box testing techniques to verify functionality against the defined requirements. Integration testing will ensure that the interactions between modules work as expected, followed by system testing to validate the entire system's performance. The goal is to ensure that the AI-based content suggestions work correctly and that users can interact with the system without issues.

1.3. Definitions and Acronyms

- **AI**: Artificial Intelligence
- **Dynamic Content**: Adaptable educational material generated by the system based on student feedback.
- **FAQ**: Frequently Asked Questions
- **NLP**: Natural Language Processing

2. Test Plan

2.1 Features to be Tested

- **Teacher Profile Management**: Test the ability for teachers to set notification and content similarity thresholds.
- **Student Lesson Page**: Test the display and interaction with lesson content, including chatbot interactions.
- **FAQ Module**: Ensure that FAQs are generated and displayed based on student inquiries.
- **Notifications**: Verify that teachers receive notifications based on student FAQs and AI suggestions.
- **Insights Page**: Test the generation of insights from student FAQs.
- **Content Suggestion**: Ensure that AI-generated content is displayed, and teachers can accept/ignore it.
- **Revert Content**: Test the functionality for reverting AI-suggested content back to the original.

2.2 Features not to be Tested

- **UI Design Consistency**: Visual design will not be the primary focus of the testing.
- **Third-party Integrations**: Any integrations not directly related to the core functionalities (such as optional plugins or extensions).

3.3 Testing Tools and Environment

- **Testing Tools**: Postman (for API testing)
- **Environment**: Windows 10/11, Node.js environment for backend testing, PostgreSQL database, and modern browsers for frontend tests (Chrome, Firefox, Edge, or Opera).
- **Database**: Database Client (viewing of PostgreSQL database)

3. Test Cases

3.n Case-n

3.n.1 Purpose

Test the notification system when AI-generated content suggestions are available.

3.n.2 Inputs

Student inquiries leading to an FAQ; teacher's notification preferences.

3.n.3 Expected Outputs & Pass/Fail Criteria

Teachers receive a notification with the correct content suggestion and can review or ignore it. Pass if the notification is received and processed correctly; fail if there is no notification or incorrect content.

3.n.4 Test Procedure

1. Simulate student interaction to generate an FAQ.
2. Check if the teacher receives a notification about AI-generated content.
3. Validate the correctness of the content suggestion.

Appendix (Test Logs)

A.1 Log for Test n

Record of actions taken during each test.

A.2 Test Results

Document the pass/fail status of each test.

A.3 Incident Report

Any issues or incidents encountered during testing should be logged here.