Logo

Description automatically generated

CMPE-492

Test Plan Report

MAI Therapist

Gülden Ünal

Project Supervisor:

Prof.Dr. Tolga Kurtuluş Çapın

Hatice Esra Yılmaz

Renas Barış Özkal

Uğur Kellecioğlu

Table of Contents

[1.Introduction 4](#_Toc135169611)

[2.Test Strategy 4](#_Toc135169612)

[2.1 Testing Levels 4](#_Toc135169613)

[2.2 Testing Types 5](#_Toc135169614)

[2.3 Testing Techniques 5](#_Toc135169615)

[2.4 Test Coverage 5](#_Toc135169616)

[2.5 Test Deliverables 5](#_Toc135169617)

[3. Test Environment 6](#_Toc135169618)

[3.1 Hardware Requirements 6](#_Toc135169619)

[3.2 Software Requirements 6](#_Toc135169620)

[3.3 Network Configuration 6](#_Toc135169621)

[3.4 Test Data 6](#_Toc135169622)

[4. Test Schedule 6](#_Toc135169623)

[4.1 Test Phases 6](#_Toc135169624)

[4.2 Milestones 7](#_Toc135169625)

[4.3 Resource Allocation 7](#_Toc135169626)

[5. Item Pass/Fail Criteria 7](#_Toc135169627)

[5.1 Usability Testing 7](#_Toc135169628)

[5.2 Unit Testing 7](#_Toc135169629)

[5.3 Acceptance Testing 8](#_Toc135169630)

[5.4 Load and Stress Testing 8](#_Toc135169631)

[6. Test Execution 8](#_Toc135169632)

[6.1 Test Execution Process 8](#_Toc135169633)

[6.2 Test Logs and Defect Tracking 8](#_Toc135169634)

[6.3 Test Results Reporting 9](#_Toc135169635)

[6.4 Test Execution Steps for Specific Features 9](#_Toc135169636)

[7.Roles and Responsibilities 10](#_Toc135169637)

[7.1 Test Team Members 10](#_Toc135169638)

[7.2 Responsibilities 10](#_Toc135169639)

[8.Risk Management 11](#_Toc135169640)

[8.1 Identified Risks 11](#_Toc135169641)

[8.2 Risk Mitigation Strategies 11](#_Toc135169642)

[8.3 Risk Monitoring 12](#_Toc135169643)

[9. Test Closure 12](#_Toc135169644)

[9.1 Criteria for Test Completion 12](#_Toc135169645)

[9.2 Lessons Learned 12](#_Toc135169646)

[9.3 Recommendations for Future Testing 13](#_Toc135169647)

[10. Glossary 13](#_Toc135169648)

# 1.Introduction

This is the Test Plan for the MAI Therapist application. The purpose of this plan is to outline the testing approach and strategies for ensuring the quality and reliability of the MAI Therapist software. The primary goal of the testing effort is to verify that the application functions as intended, meets the specified requirements, and provides an effective and user-friendly experience for both therapists and clients. all testing for the MAI Therapist application will be conducted on the DEV and TEST environments. The DEV environment will be utilized for unit testing, focusing on testing individual components and functionalities. The TEST environment, on the other hand, will be dedicated to user acceptance testing and comprehensive system testing to ensure the application's overall performance and stability.

Considering the aggressive development timeline estimated at approximately 3 months, any delays encountered during the development process may have notable impacts on the test plan. It is crucial to maintain a synchronized schedule between development and testing activities to ensure that sufficient time is allocated for thorough testing and bug resolution. Continuous collaboration and communication between the development team and the testing team will be essential to identify and address any potential issues promptly.

Efforts will be made to optimize the testing process while maintaining the required level of testing coverage to mitigate the risks associated with the tight timeline. Regular progress updates and status meetings will be conducted to monitor the development progress and adjust the test plan as necessary. Flexibility and adaptability will be key in accommodating any changes or adjustments required throughout the development and testing phases.

# 2.Test Strategy

## 2.1 Testing Levels

1. Unit Testing: Test individual components and function of web app & AI model that analyze emotions by facial expressions and voice tone.
2. Integration Testing: Test the integration of components to make sure that they work together correctly. Test the AI model with the video recording functionality to ensure that the analysis is performed correctly on the recorded session.
3. System Testing: Test the overall system to verify its performance, functionality, and usability.
4. Acceptance Testing: Test the web app with the requirements/expectations of the end users & conduct test with real therapists to validate the app’s analysis results.

## 2.2 Testing Types

1. Functional Testing: Verify that AI model detects and analyzes emotions accurately, using gestures, facial expressions, and voice tone. And verify all the functions and features of the web app work well.
2. Usability Testing: Evaluate the app's user experience for therapists, ensuring it’s easy to navigate.
3. Performance Testing: Test the app’s performance under different conditions. For example, response times & generating analysis results.
4. Security Testing: Test the app’s security measures to protect the user and client data. Like session records and card information.
5. Compatibility Testing: Ensure that the app works well across different devices, browsers, and OSs.
6. Regression Testing: Re-test previously working features to ensure they still work after changes/updates.

## 2.3 Testing Techniques

1. Black Box Testing: Validate the app's functionality from an external perspective without knowledge of its internal workings.
2. White Box Testing: Test the app to verify its correctness of internal algorithms and logic.
3. Exploratory Testing: Test the app by exploring scenarios and paths to discover any unexpected issues or usability concerns.

## 2.4 Test Coverage

1. UI Coverage: Ensure that the user interface is thoroughly tested.
2. Analysis Coverage: Test the accuracy of emotion recognition.
3. Error Handling Coverage: Test the error handling and error messages.

## 2.5 Test Deliverables

1. Test Plan: Document detailing the test strategy and schedule.
2. Test Cases: Instructions of tests and results.
3. Test Reports: Summery of the test results, including any issues if found.

# 3. Test Environment

## 3.1 Hardware Requirements

A computer or device with a modern web browser.

Webcam or camera device for video recording (if the therapist intends to record sessions directly from their device).

## 3.2 Software Requirements

Operating System: Windows, macOS, or Linux

Web Browser: Google Chrome, Mozilla Firefox, Microsoft Edge, Safari, or any other modern web browser with JavaScript enabled.

## 3.3 Network Configuration

Stable internet connection with sufficient bandwidth for video upload and download.

## 3.4 Test Data

Sample patient information data for creating new patient profiles and updating existing profiles.

Sample video recordings of therapy sessions for uploading and testing the emotion detection and transcript generation features.

Test transcripts corresponding to the sample video recordings for verification of the combined results.

Test reports generated by the system for comparison and validation.

# 4. Test Schedule

## 4.1 Test Phases

1. Planning Phase: Defining the test objectives and timelines. Planning on which tests to do.
2. Preparation Phase: Setting up test environment.
3. Execution Phase: Performing of the tests & recording of the results.
4. Closure Phase: Analyzing the test results and generating test reports. Resolving of the defects.

## 4.2 Milestones

1. Completion of Test Planning and Test Plan Approval.
2. Test Environment Setup and Readiness.
3. Completion of Test Case Preparation.
4. Test Execution and Defect Reporting.
5. Completion of Test Analysis and Test Reports.
6. Review and Resolution of Reported Defects.
7. Final Acceptance Testing and Sign-off.

## 4.3 Resource Allocation

1. Video Recording Equipment: Ensure availability of suitable cameras and recording devices.
2. AI Analysis Resources: Allocate necessary computational resources and AI models for emotions, gestures, facial expressions, and voice tone analysis.
3. Time: Allocate sufficient time for planning, preparation, execution, and analysis

# 5. Item Pass/Fail Criteria

The pass or fail criteria for each test item in the MAI Therapist application will vary based on the specific type of testing being conducted.

## 5.1 Usability Testing

Usability testing focuses on gathering valuable insights to inform future design improvements. It is not strictly about pass or fail determinations. Usability testing will be conducted in two phases: one with clickable mockups and another with working software. A minimum requirement for usability testing is to test one person from each user class during each phase. However, it is ideal to test three or four users in each role to obtain a reliable System Usability Scale (SUS) score.

## 5.2 Unit Testing

Unit testing will utilize the JUnit framework, which will determine the pass or fail status based on the assert statements. Each individual unit test will be marked as passed or failed by JUnit after execution. The entire unit testing process will be considered complete when all test cases pass successfully, achieving a minimum of 70% code coverage.

## 5.3 Acceptance Testing

Acceptance testing will be conducted to evaluate the fulfillment of requirements outlined in the Requirements Document. Initially, all requirements will be marked as “not accepted.” After review by project sponsors, if a requirement has been successfully implemented, it will be marked as “accepted.” This process will also apply to earlier testing, but official acceptance will only be noted by the sponsors. Acceptance testing will be considered complete when all requirements with a “High” or “Medium” priority in the Requirements Document have been marked as “accepted.”

## 5.4 Load and Stress Testing

The specific pass/fail criteria for load and stress testing will be determined by ITS (Information Technology Services). Generally, if the application performs acceptably under a given capacity, it will pass the test item. Conversely, if the application underperforms, it will be considered a test item failure.

# 6. Test Execution

## 6.1 Test Execution Process

- Set up the necessary hardware and software components in the testing environment.

- To test it out, install and set up the client web interface.

- Create the accounts and access rights for AWS services (such as AWS Rekognition, AWS S3, and DynamoDB) as needed.

- Make sure the system is able to connect to the AWS cloud services.

## 6.2 Test Logs and Defect Tracking

- Create a test log template to record test execution details, including test case ID, description, actual results, and any defects encountered.

- To record and monitor any issues or defects discovered during test execution, use a defect tracking system or tool.

- Document steps to reproduce any identified defects for easier troubleshooting and resolution.

## 6.3 Test Results Reporting

- Run test cases to confirm the application's functionality.

- Log the actual outcomes of each test case in the test log, taking note of any departures from the intended course of events.

- Record any errors that occur during execution and give the proper severity levels.

- Add each test case's execution state (pass, fail, or blocked) to the test log.

- Produce test result summaries that include the number of executed, passed, failed, and blocked test cases as well as a summary of the overall test execution process.

- Include thorough details on any flaws discovered, their effects, and reproduction procedures, as well as suggestions for fixing the problems.

## 6.4 Test Execution Steps for Specific Features

1. Logging In and Registration:

- Verify that users can successfully register by creating a new account.

- Validate that registered users can log in using their credentials.

- Ensure that appropriate error messages are displayed for invalid login attempts.

1. Patient Information Management:

- Test the creation of new patient information by verifying that it is stored accurately in the DynamoDB database.

- Validate the ability to update existing patient information and verify that the changes are saved correctly.

- Ensure that appropriate validations and error handling are in place for patient information management.

1. Video Recording and Analysis:

- Test the upload functionality for session videos, ensuring that they are successfully stored in AWS S3 storage.

- Verify that uploaded videos are processed by AWS Rekognition for emotion detection accurately.

- Validate that the generated transcript is combined with emotions and stored correctly in DynamoDB.

- Confirm that the final report is received and displayed/downloaded by the user accurately.

# 7.Roles and Responsibilities

## 7.1 Test Team Members

- Test Manager: In charge of organizing, managing, and planning the complete testing process. Ensures that the test plan is successfully carried out and produces reliable findings.

- Test Lead: Assists the test manager in planning and executing the testing tasks. Collaborates with the development team and stakeholders to prioritize test cases and ensure timely execution.

- Test Analysts/Testers: In charge of designing and executing test cases, documenting test results, and reporting defects. Collaborates with the development team to identify and resolve problems.

- Automation Testers: Develops and maintains automated test scripts and frameworks. Carries out automated tests to increase test efficiency and coverage.

- AWS Specialist: Provides expertise in setting up and configuring the AWS cloud services required for the project. Assists with integration testing and ensures that AWS components are operating correctly.

- Database Administrator: Manages the DynamoDB database and makes sure that data is correctly stored and retrieved. Collaborates with the testing team to validate data integrity and perform database-related tests.

## 7.2 Responsibilities

- Test Manager:

- Creates the test strategy, timetable, and plan of tests.

- Manages the test team and allocates resources.

- Reports to the project manager on the status of testing activities.

- Shares information about test progress, risks, and problems with stakeholders.

- Test Lead:

- Assists in test planning and coordination.

- Collaborates closely with the test manager to prioritize test cases and allocate resources.

- Coordinates with the development team to ensure that bugs are fixed quickly.

- Updates the test manager on issues and reports test progress.

- Test Analysts/Testers:

- Create and carry out test cases in accordance with requirements.

- Document test results and report defects.

- Take part in meetings for defect triage and contribute ideas for fixing defects.

- Work with the development team to identify the root causes of defects.

- Automation Testers:

- Develop and maintain automated test scripts and frameworks.

- Execute automated tests and analyze test results.

- Collaborate with the test team to identify areas for test automation.

- Provide guidance on test automation best practices.

- AWS Specialist:

- Configure and set up the necessary AWS cloud services.

- Ensure that AWS component integration and functioning are correct.

- Perform integration testing in conjunction with the testing team.

- Provide technical expertise in resolving AWS-related issues.

- Database Administrator:

- Manage the DynamoDB database.

- Ensure accurate data retrieval, storage, and integrity.

- Work with the testing team to conduct database-related tests.

- Enhance database performance and troubleshoot issues.

# 8.Risk Management

## 8.1 Identified Risks

- Risk 1: Inadequate test coverage: Because of time restrictions, there is a chance that some functionality or edge situations won't get enough testing, which could result in system flaws.

- Risk 2: Resource constraints: The thoroughness and efficiency of the testing process may be impacted by a lack of testing resources, such as qualified testers or appropriate testing equipment.

- Risk 3: Technical challenges with AWS integration: It may be technically challenging to integrate AWS services like Rekognition, S3, and DynamoDB, which could cause delays or problems with video processing, emotion recognition, or data storage.

## 8.2 Risk Mitigation Strategies

- Risk 1: To mitigate the risk of insufficient test coverage, a thorough test plan will be designed, prioritizing crucial functionality and identifying key test cases. Test prioritization will be based on requirements analysis and consultation with stakeholders.

- Risk 2: To address resource constraints, efforts will be made to optimize resource allocation by cross-training team members, using automation tools when appropriate, and seeking additional resources as needed. To guarantee effective use of the resources at hand, the team will develop clear communication channels and collaboration.

- Risk 3: To mitigate technical challenges with AWS integration, thorough testing and verification of AWS services' interactions will be performed in a dedicated test environment. Regular communication with AWS support will be maintained to address any technical issues promptly.

## 8.3 Risk Monitoring

- Risks will be continuously monitored throughout the project lifecycle, with regular risk assessments and reviews conducted during project meetings. Accordingly, the risk log will be updated, reflecting new risks, modifications to existing risks, and the status of mitigation efforts.

# 9. Test Closure

## 9.1 Criteria for Test Completion

• The test plan's specified test cases have all been carried out.

• The web app’s features and functionality have undergone extensive testing and satisfy the criteria.

• Defects and issues discovered during testing have been addressed and fixed or a choice has been taken to postpone their fixing depending on their severity and priority. We’ve focused on the important ones first.

• It is determined that the test coverage is adequate and includes all necessary application functionality. It’s above %90.

• Performance and stability testing was done, and it passed the required standards. Test results have been reviewed and the acceptance criteria have been met.

• The testing group concurs that the application is prepared for deployment or future development.

## 9.2 Lessons Learned

• Identification of testing approaches or techniques that proved to be effective in ensuring the quality and reliability of the application.

• Identification of any gaps or areas where testing could be improved or expanded.

• Evaluation of the test data used, and any challenges faced in acquiring or generating relevant and representative test data.

• Assessment of the effectiveness of the test environment setup, including hardware, software, and network configurations.

• Evaluation of the collaboration and communication within the testing team and with other stakeholders.

• Analysis of any unforeseen issues or obstacles encountered during testing and the corresponding solutions or workarounds implemented.

## 9.3 Recommendations for Future Testing

• Implement test automation: To increase testing efficiency and repeatability, identify sections of the application that can be automated and invest in test automation frameworks and tools.

• Enhance test data management: Create a thorough plan for managing test data that addresses producing realistic and varied data sets, guaranteeing data privacy and security, and optimizing the providing of test data.

• Implement performance testing: Consider implementing performance testing to evaluate the behavior of the application under various load levels, ensuring it can manage anticipated user traffic and operate at its best.

• Expand test coverage: Review and update the test coverage on a regular basis to account for new features, edge cases, and scenarios that might have slipped through the cracks during the most recent testing cycle.

• Foster collaboration and communication: In order to establish a shared understanding of requirements, concerns, and priorities, the testing team, developers, and stakeholders should place an emphasis on good communication and collaboration.

• Establish a robust defect management process: Implement a defined procedure for prioritizing, tracking, and resolving defects to ensure that all issues are properly detected, assigned, and tracked until resolution.

Future testing initiatives can benefit from the knowledge gained by implementing these suggestions, leading to enhanced quality assurance procedures and better testing outcomes.

# 10. Glossary

AI: Artificial Intelligence

AWS: Amazon Web Services

DEV: Development Environment

TEST: Testing Environment

UI: User Interface

SUS: System Usability Scale

OS: Operating System

QA: Quality Assurance