

# Introduction to Software Engineering

## Software Testing

The student team is required to complete the **Software Testing** documentation for the assigned course project, following the attached template.



Software Engineering Department  
Faculty of Information and Technology  
University of Science

# Table of Contents

<b>Objectives</b>	<b>1</b>
<b>1 Member Contribution Assessment</b>	<b>2</b>
<b>2 Test plan</b>	<b>3</b>
<b>3 Test cases</b>	<b>5</b>
3.1 List of test cases	5
3.2 Test case specifications	9
3.2.1 Verify User Registration	9
3.2.2 Verify Valid Login	11
3.2.3 Verify Invalid Login	12
3.2.4 Verify Microphone Access	13
3.2.5 Verify Speech Analysis & Feedback	14
3.2.6 Verify Complete Lesson Workflow	16
3.2.7 Verify Random Practice Mode	18
3.2.8 Verify Dashboard Statistics Update	20
3.2.9 Verify Session History Recording	22
3.2.10 Verify Admin Add Topic	24





# Software Testing

## Objectives

This document focus on the following topics:

- ✓ Completing the Software Testing document with the following sections:
  - Test Plan
  - Test Cases
- ✓ Understanding the Software Testing document.

# 1 Member Contribution Assessment

ID	Name	Contribution (%)	Signature
23127241	Đoàn Thành Phát	25%	
23127089	Nguyễn Quang Minh	25%	
23127085	Phạm Phát Lộc	25%	
23127102	Lê Quang Phúc	25%	

# 2 Test plan

## 2.1 Objectives

The primary objective of this testing plan is to verify that the **English Speaking Learning System** functions correctly according to the requirements specified in Section 3, ensures data integrity, and meets the performance constraints (response time < 5 seconds). The testing process aims to identify and fix defects before the final deployment.

## 2.2 Testing Levels and Techniques

### 2.2.1 Unit Testing

- **Technique:** White-box testing.
- **Objects:**
  - + **Backend Functions (Node.js):** Specifically the algorithms for calculating pronunciation scores and processing JSON feedback from the AI engine.
  - + **Frontend Components (ReactJS):** Individual UI components like the *VoiceRecorder*, *AudioPlayer*, and *ResultChart*.
- **Goal:** Ensure that each specific module of code performs its logic correctly in isolation.

### 2.2.2 Integration Testing

- **Technique:** Black-box testing (API Testing).
- **Objects:**
  - + **RESTful APIs:** Testing endpoints for User Authentication (Login/Register), Lesson Retrieval, and Audio Upload.
  - + **External Service Integration:** Verifying the communication between the Backend and **Google Cloud Speech-to-Text API** to ensure audio is sent and transcripts are received correctly.
  - + **Database Interaction:** Verifying that User, PracticeSession, and ExerciseAttempt data is correctly created, read, updated, and deleted in MySQL.
- **Goal:** Verify that different modules and services work together seamlessly.

### 2.2.3 System & Performance Testing

- **Technique:** Load testing and Functional testing.
- **Objects:** The entire web application deployed on the staging environment.
- **Goal:**
  - + Validate the Non-Functional Requirement that voice analysis returns results within 3–5 seconds
  - + Ensure the system handles concurrent users without crashing.
  - + Verify the application is responsive on both Desktop and Mobile devices.

### 2.2.4 User Acceptance Testing (UAT)

- **Technique:** Beta testing with real users.
- **Objects:** End-to-end user flows (e.g., A user logs in -> Selects a Topic -> Records Audio -> Receives Feedback -> Checks History).
- **Goal:** Confirm the system meets the "Definition of Done" and satisfies the needs of real English learners.

## 2.3 Tools and Environment

- **Unit Testing Framework:** Jest (for React and Node.js logic).
- **API Testing:** Postman (for testing backend endpoints).
- **Performance Testing:** Google Lighthouse (for Frontend performance) and Apache JMeter (for load testing).
- **Browser Environment:** Google Chrome, Microsoft Edge (latest versions).

## 2.4 Success Criteria

The testing phase will be considered successful when:

- All critical and high-priority bugs are resolved.
- The speech recognition and scoring accuracy reaches **>= 85%**.
- The average system response time for feedback is under **5 seconds**.

# 3 Test cases

## 3.1 List of test cases

Seq	Test case	Target	Description
1	Verify User Registration	Register Account	Verify that a new Learner can successfully create an account by providing a valid name, email, and password. The system should create a new profile in the database.
2	Verify Register Duplicate Email	Register Account	Verify that the system displays an error message ("Email already in use") when a user attempts to register with an email address that already exists in the database.
3	Verify Valid Login	Log In	Verify that a registered user (Learner or Admin) can log in successfully with correct credentials and be redirected to the appropriate Dashboard.
4	Verify Invalid Login	Log In	Verify that the system displays an error

			message ("Invalid email or password") when a user attempts to log in with incorrect credentials.
5	Verify Microphone Access	Perform Speaking Exercise	Verify that the system correctly prompts the user for microphone permissions and starts capturing audio when the "Record" button is clicked.
6	Verify Mic Permission Denied	Perform Speaking Exercise	Verify the system's behavior when the browser denies microphone access. It should display a helpful instruction to enable permissions instead of crashing.
7	Verify Speech Analysis & Feedback	Analyze Speech	Verify that after recording, the system sends audio to the Speech Service and displays the correct Score, Fluency metrics, and highlights mispronounced words.
8	Verify No Speech Detected	Analyze Speech	Verify that if the user clicks Record but stays silent, the system detects the silence and prompts "We couldn't hear you" instead of returning a zero score or error.
9	Verify Complete Lesson Workflow	Complete Speaking Lesson	Verify that a learner can select a topic, complete all exercises in a lesson, and see the "Lesson



			Summary Report" upon completion.
10	Verify Random Practice Mode	Practice Randomly	Verify that the system can fetch a random sentence from the database and allow the user to practice it without selecting a specific topic.
11	Verify Dashboard Statistics Update	Dashboard	Verify that the "Streak Counter" and "Overall Score" on the Dashboard are updated correctly immediately after a user completes a practice session.
12	Verify Search Topic Function	Dashboard / Topic List	Verify the search functionality. Enter a keyword (e.g., "Travel") into the search bar on the Dashboard. The system must update the topic grid immediately and display only topics containing the keyword in their title or description.
13	Verify Search No Results	Dashboard	Verify the system behavior when a search yields no results. Enter a random string (e.g., "xyz123"). The system must display a friendly message (e.g., "No topics found") instead of showing a blank screen or crashing.

14	Verify Clear Filter/Search	Dashboard	Verify the reset functionality. After applying a search or filter, click the "Clear" button or delete the search text. The system must restore the full list of topics or history sessions to its default state.
15	Verify Session History Recording	Session History Page	Verify that a completed practice session is saved to the database and appears in the "History" tab with the correct date, topic name, and score.
16	Verify Skill Radar Chart	View Pronunciation Score	Verify that the Radar Chart in the Profile correctly visualizes the 5 skill metrics (Pronunciation, Fluency, Intonation, Grammar, Vocabulary) based on historical data.
17	Verify Admin Add Topic	Manage Topics	Verify that an Administrator can successfully add a new Topic (with Name, Description, Thumbnail) and it appears in the Learner's topic grid.

18	Verify Admin Delete Topic Constraint	Manage Topics	Verify data integrity: An Administrator cannot delete a Topic if it currently contains Lessons. The system must show an error message preventing the deletion.
19	Verify Promote User Role	Assign Roles	Verify that a Super Admin can promote a "Learner" account to "Admin". Confirm that the promoted user gains access to the Admin Dashboard upon next login.
20	Verify Delete User Account	User Management	Verify that an Administrator can permanently delete a specific user account. Upon confirmation, the user should be removed from the system list, and their login credentials should no longer be valid.

## 3.2 Test case specifications

### 3.2.1 Verify User Registration

Test case	Verify User Registration
<i>Related Use case</i>	U001 - Register Account
<i>Context</i>	The user is accessing the web application for the first time. The email address used for registration does not currently exist in the database. The user is on the Registration Page (/register).

<i>Input Data</i>	<ul style="list-style-type: none"><li>- <b>Full Name:</b> "Test Student A"</li><li>- <b>Email:</b> "student.new@example.com"</li><li>- <b>Password:</b> "StrongPass@123"</li><li>- <b>Confirm Password:</b> "StrongPass@123"</li></ul>
<i>Expected Output</i>	<ol style="list-style-type: none"><li>1. The system creates a new record in the users table.</li><li>2. The system creates a linked record in userstats table.</li><li>3. The user is automatically redirected to the <b>Dashboard</b> page.</li><li>4. A welcome message "Registration successful" is displayed.</li></ol>
<i>Test steps</i>	<ol style="list-style-type: none"><li>1. Open the web browser and launch the application.</li><li>2. Click on the <b>"Sign Up"</b> or <b>"Register"</b> button.</li><li>3. Enter the <b>Full Name</b>, <b>Email</b>, <b>Password</b>, and <b>Confirm Password</b> into the respective fields.</li><li>4. Click the <b>"Create Account"</b> button.</li><li>5. Verify the redirection to the Dashboard and check the database for the new user.</li></ol>
<i>Actual Output</i>	The system successfully created the account, redirected the user to the Dashboard, and the new user ID was found in the database.
<i>Result</i>	Passed / Failed

### 3.2.2 Verify Valid Login

Test case	Verify Valid Login
Related Use case	U002 - Log In
Context	The user has a valid account (Learner or Admin) and is currently on the Login Page (/login). The system is ready to accept credentials.
Input Data	- <b>Email:</b> "student.existing@example.com"  - <b>Password:</b> "CorrectPassword123"
Expected Output	1. The system verifies the credentials against the database.  2. The user is authenticated and a session token is generated.  3. The user is redirected to the appropriate Dashboard (Learner or Admin).  4. Examples: Learner -> /dashboard; Admin -> /admin.
Test steps	1. Open the web browser and navigate to the Login page.  2. Enter the valid Email address into the email field.  3. Enter the corresponding correct Password into the password field.  4. Click the <b>Log In</b> button.  5. Observe the URL redirection and the presence of user-specific data on the landing page.
Actual Output	The system authenticated the user, issued a token, and redirected the browser to /dashboard without errors.
Result	Passed

### 3.2.3 Verify Invalid Login

Test case	Verify Invalid Login
Related Use case	U002 - Log In
Context	The user is on the Login Page. The user attempts to access the system using incorrect credentials (wrong password or non-existent email).
Input Data	<b>- Email:</b> "student.existing@example.com" <b>- Password:</b> "WrongPassword!!!"
Expected Output	<ol style="list-style-type: none"><li>1. The system rejects the login attempt.</li><li>2. The user remains on the Login Page.</li><li>3. An error message "Invalid email or password" is displayed clearly to the user.</li></ol>
Test steps	<ol style="list-style-type: none"><li>1. Navigate to the Login page.</li><li>2. Enter a valid or invalid Email address</li><li>3. Enter an incorrect Password.</li><li>4. Click the "Log In" button.</li><li>5. Observe the error message and ensure no redirection occurs.</li></ol>
Actual Output	The system displayed "Invalid email or password" in red text below the input fields and did not redirect the user.
Result	Passed

### 3.2.4 Verify Microphone Access

Test case	Verify Microphone Access
Related Use case	U005 - Perform Speaking Exercise
Context	The Learner is in a Practice Session and attempts to record their answer for a speaking exercise. The browser has not yet granted microphone permission for this site.
Input Data	- User Action: Click "Record" icon/button
Expected Output	<ol style="list-style-type: none"><li>1. The browser displays a permission prompt asking to allow microphone access.</li><li>2. Upon clicking "Allow", the system begins capturing audio input.</li><li>3. Visual feedback (e.g., a waveform or "Recording..." status) is shown to the user.</li></ol>
Test steps	<ol style="list-style-type: none"><li>1. Log in as a Learner.</li><li>2. Navigate to a Topic and start a Lesson to enter the Practice Session screen.</li><li>3. Locate the Voice Recorder component.</li><li>4. Click the microphone icon / "Record" button.</li><li>5. If prompted by the browser, select "Allow".</li><li>6. Speak into the microphone and check for visual indicators of active recording.</li></ol>
Actual Output	The browser requested permission. After allowing, the recording timer started counting up, and the visualizer reacted to voice input.
Result	Passed

### 3.2.5 Verify Speech Analysis & Feedback

Test case	Verify Speech Analysis & Feedback
<i>Related Use case</i>	<b>U009 - Analyze Speech</b> (Included in U005 - Perform Speaking Exercise)
<i>Context</i>	The Learner is performing a speaking exercise. Microphone permissions are granted , and the device has an active Internet connection to communicate with the Speech Service.
<i>Input Data</i>	<ul style="list-style-type: none"><li>- <b>Action:</b> Click the "Record" button.</li><li>- <b>Audio Input:</b> The user speaks the target sentence displayed on the screen (e.g., "Tell me about your workday") .</li><li>- <b>Action:</b> Click "Stop" (or wait for silence detection).</li></ul>



<i>Expected Output</i>	<ol style="list-style-type: none"><li>1. The system captures the audio and sends it to the Speech Service API.</li><li>2. The system receives a structured JSON response containing the Transcript and Scores.</li><li>3. The UI displays the <b>Overall Score</b> (e.g., 82%).</li><li>4. The UI displays detailed metrics: <b>Fluency</b> and <b>Pronunciation</b> scores.</li><li>5. Mispronounced words are highlighted (e.g., in red) with suggestions for improvement.</li></ol>
<i>Test steps</i>	<ol style="list-style-type: none"><li>1. Navigate to a speaking lesson and open an exercise.</li><li>2. Click the <b>Record</b> button.</li><li>3. Speak the prompt text clearly into the microphone.</li><li>4. Click <b>Stop</b> to finish recording.</li><li>5. Wait for the processing (verify it takes less than 3-5 seconds).</li><li>6. Verify that the score, error highlights, and feedback are displayed correctly.</li></ol>

<i>Actual Output</i>	The system processed the audio in 2.5 seconds, displayed a score of 85%, and highlighted the word "workday" as mispronounced.
<i>Result</i>	Passed

### 3.2.6 Verify Complete Lesson Workflow

Test case	Verify Complete Lesson Workflow
<i>Related Use case</i>	<b>U003 - Complete Speaking Lesson</b>
<i>Context</i>	The Learner is logged in and viewing the Topic list. The user intends to complete a full lesson sequence.
<i>Input Data</i>	<ul style="list-style-type: none"> <li>- <b>Topic Selection:</b> Select a Topic (e.g., "Greetings") .</li> <li>- <b>Lesson Selection:</b> Select a specific Lesson from the list.</li> <li>- <b>Exercise Interaction:</b> Complete all sequential exercises (Pronunciation, Conversation, etc.).</li> <li>- <b>Navigation:</b> Click "Next" after each exercise.</li> </ul>

<i>Expected Output</i>	<ol style="list-style-type: none"><li>1. The learner can successfully select a topic and enter a lesson.</li><li>2. The system retrieves and displays the list of exercises for the lesson.</li><li>3. The system loops through each exercise item until all are finished.</li><li>4. Upon completion of the last item, the "<b>Lesson Summary Report</b>" is displayed.</li><li>5. The lesson is marked as "Completed" in the user's learning history.</li></ol>
<i>Test steps</i>	<ol style="list-style-type: none"><li>1. From the Dashboard, select a <b>Topic</b>.</li><li>2. Click on a <b>Lesson</b> that has not been started yet.</li><li>3. Complete the first exercise and click <b>Next</b>.</li><li>4. Repeat for all exercises in the sequence.</li><li>5. Verify that the <b>Lesson Summary Report</b> appears after the final exercise.</li><li>6. Return to the menu and check if the lesson status is updated to <b>Completed</b>.</li></ol>

<i>Actual Output</i>	The user completed all 5 exercises. The Summary Report appeared showing an average score of 80%. The lesson is now marked with a green checkmark.
<i>Result</i>	Passed

### 3.2.7 Verify Random Practice Mode

Test case	Verify Random Practice Mode
<i>Related Use case</i>	<b>U004 - Practice Randomly.</b>
<i>Context</i>	The Learner is logged in and is on the Dashboard. The user wants to practice without selecting a specific topic.
<i>Input Data</i>	<ul style="list-style-type: none"> <li>- <b>Trigger:</b> Click the "Quick Practice" (or "Random Practice") button.</li> <li>- <b>Action:</b> Record audio for the displayed sentence.</li> <li>- <b>Navigation:</b> Click the "Next" button after receiving feedback.</li> </ul>

<i>Expected Output</i>	<ol style="list-style-type: none"><li>1. The system randomly fetches a sentence from the database suitable for the user's level.</li><li>2. The system allows the user to practice the sentence and view feedback.</li><li>3. When the user clicks "Next", the system immediately fetches a <b>new</b> random sentence.</li><li>4. The session continues endlessly until the learner chooses to "End Session".</li></ol>
<i>Test steps</i>	<ol style="list-style-type: none"><li>1. On the Dashboard, click the <b>"Quick Practice"</b> button.</li><li>2. Verify that a random sentence is displayed.</li><li>3. Record speech and wait for the score.</li><li>4. Click the <b>Next</b> button.</li><li>5. Verify that a <i>different</i> random sentence is loaded immediately.</li><li>6. Click <b>End Session</b> and verify the session summary is shown.</li></ol>
<i>Actual Output</i>	The system loaded a random sentence. After clicking Next, a new sentence appeared. The session ended successfully when "End Session" was clicked.

<i>Result</i>	Passed
---------------	--------

### 3.2.8 Verify Dashboard Statistics Update

Test case	Verify Dashboard Statistics Update
<i>Related Use case</i>	<b>U005 - Complete Practice Session.</b>
<i>Context</i>	The user has just completed a practice session successfully and is currently viewing the Dashboard page (/dashboard). The system should update user statistics in real-time based on the session outcome.
<i>Input Data</i>	<ul style="list-style-type: none"> <li>- Practice Session Score: 85 (out of 100)</li> <li>- Session Topic: "Basic Greetings"</li> <li>- Session Date: Current date (e.g., December 25, 2025)</li> <li>- User's Previous Streak: 3 consecutive days</li> <li>- User's Previous Overall Score: 78.5</li> </ul>

<i>Expected Output</i>	<ol style="list-style-type: none"><li>1. The system calculates and updates the "Streak Counter" (increments if the session was completed on a consecutive day; resets otherwise).</li><li>2. The system recalculates and updates the "Overall Score" based on the new session score (weighted average or cumulative calculation).</li><li>3. The Dashboard refreshes to display the updated statistics without requiring a page reload.</li><li>4. Examples: If streak continues, counter shows 4; Overall Score updates to approximately 79.2 (based on averaging logic).</li></ol>
<i>Test steps</i>	<ol style="list-style-type: none"><li>1. Log in as a valid learner user and navigate to the Dashboard page.</li><li>2. Start and complete a practice session for a topic (e.g., "Basic Greetings") by answering questions and submitting.</li><li>3. Upon completion, observe the session score displayed (e.g., 85).</li><li>4. Return to or refresh the Dashboard page.</li><li>5. Check the "Streak Counter" and "Overall Score" values to ensure they reflect the updated calculations.</li></ol>

<i>Actual Output</i>	The system updated the Streak Counter to 4 and the Overall Score to 79.2 immediately after session completion, with the Dashboard reflecting changes without errors.
<i>Result</i>	Passed

### 3.2.9 Verify Session History Recording

Test case	Verify Session History Recording
<i>Related Use case</i>	<b>U006 - View Session History</b>
<i>Context</i>	The user has completed a practice session and is currently on the History Page (/history). The system should have saved the session details to the database for retrieval.
<i>Input Data</i>	<ul style="list-style-type: none"> <li>- Practice Session Score: 92 (out of 100)</li> <li>- Session Topic: "Business English Phrases"</li> <li>- Session Date: Current date (e.g., December 25, 2025)</li> <li>- User ID: Valid learner user (e.g., student.existing@example.com)</li> </ul>



<i>Expected Output</i>	<ol style="list-style-type: none"><li>1. The system saves the completed practice session to the database with correct details.</li><li>2. The session appears in the "History" tab on the History Page.</li><li>3. The entry displays the correct date, topic name, and score.</li><li>4. Examples: History entry shows "December 25, 2025 - Business English Phrases - Score: 92".</li></ol>
<i>Test steps</i>	<ol style="list-style-type: none"><li>1. Log in as a valid learner user and navigate to the Practice page.</li><li>2. Start and complete a practice session for a topic (e.g., "Business English Phrases") by answering questions and submitting.</li><li>3. Navigate to the History Page (/history).</li><li>4. Access the "History" tab.</li><li>5. Verify the most recent entry matches the session details (date, topic name, score).</li></ol>
<i>Actual Output</i>	The system saved the session to the database, and the History tab displayed the entry with correct date "December 25, 2025", topic "Business English Phrases", and score "92" without errors.

<i>Result</i>	Passed
---------------	--------

### 3.2.10 Verify Admin Add Topic

Test case	Verify Admin Add Topic
<i>Related Use case</i>	<b>U007 - Manage Topics.</b>
<i>Context</i>	An Administrator is logged in and currently on the Admin Topic Management page (/admin/topics). The system is ready to accept new topic details for addition.
<i>Input Data</i>	<ul style="list-style-type: none"> <li>- Topic Name: "Advanced Idioms"</li> <li>- Topic Description: "Practice advanced English idioms and their usage in context."</li> <li>- Topic Thumbnail: Valid image URL or file (e.g., "idioms-thumbnail.jpg")</li> </ul>

<i>Expected Output</i>	<ol style="list-style-type: none"><li>1. The system validates and saves the new topic to the database.</li><li>2. The topic appears in the Administrator's topic management list.</li><li>3. The topic becomes visible in the Learner's topic grid on the Dashboard or Topics page.</li><li>4. Examples: New topic "Advanced Idioms" with description and thumbnail displays in both admin and learner views.</li></ol>
<i>Test steps</i>	<ol style="list-style-type: none"><li>1. Log in as a valid administrator user and navigate to the Admin Topic Management page (/admin/topics).</li><li>2. Enter the Topic Name into the name field.</li><li>3. Enter the Topic Description into the description field.</li><li>4. Upload or provide the Topic Thumbnail.</li><li>5. Click the "Add Topic" button.</li><li>6. Log out and log in as a learner user, then navigate to the Dashboard or Topics page to observe the new topic in the grid.</li></ol>
<i>Actual Output</i>	The system saved the new topic "Advanced Idioms" with description and thumbnail, and it appeared in the learner's topic grid without errors.

<i>Result</i>	Passed
---------------	--------