Software Requirements Specification

for

Independent Learning App

Version 1.0 approved

Prepared by Allison Ho, Ezekiel Pulido, Hunter Pope, Claire Justis

CSE3213 Group 10

10/6/2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
All Authors	10/6/24	Initial Documentation	1.0

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification Document is to provide a comprehensive overview of Independent Learning App Version 1.0. This document serves as a detailed guideline for the foundation defining different requirements, functional and non-functional. The overarching goal of this document is also to allow the different parties involved to have a clear comprehensive understanding of the project in full.

1.2 Document Conventions

This document in organized into different sections to make sure that there is a clear understanding of the different aspects of Independent Learning App.
'#. Title' – Intended to separate the overall sections of this document.
'#.# Subtitle' – Subtitle of the overarching section.
'#.#.# Subsection' - Subcategories of the subsection.
'Bold' – Items to be paid attention to within the different subsections.
Unless stated otherwise each requirement stated has its own priority.

1.3 Intended Audience and Reading Suggestions

The SRS document is designed to help the different parties involved in the creation of Independent Learning App. The parties involved are as follows.

- Development
- Project Managers
- Marketing Staff
- Users
- Testers
- Documentation writers

The recommended reading for each party is as follows.

Introduction: For all parties to understand the project.
Overall Description: Developers, Project Managers, Users
External Interface Requirements: Developers, Project Managers, Marketing Staff, Testers, Documentation Writers
System Features: Developers, Project Managers, Marketing, Testers, Documentation Writers
Other Nonfunctional Requirements: Developers, Project Managers, Testers, Documentation Writers
Other Requirements: All parties to understand the document

Following these reading requirements ensures that each party fully understands the development process of Independent Learning App.

1.4 Product Scope

Our client has requested a functioning interactive website that allows her students to practice the MAAP-A standardized test. The application will enable students to practice the tests in a controlled environment on electronic devices, without the need for a physical proctor. The scope of this first iteration of the application will be a complete test environment for third grade students.

1.5 References

"How to meet WCAG (quick reference)," How to Meet WCAG (Quickref Reference), https://www.w3.org/WAI/WCAG22/quickref/?currentsidebar=%23col_customize (accessed Oct. 6, 2024).

"Mississippi Academic Assessment Program-Alternate (MAAP-A)," The Mississippi Department of Education, https://www.mdek12.org/OSA/SP/MAAP-A (accessed Oct. 6, 2024).

Roger S. Pressman_Bruce R. Maxim – Software Engineering_A Practitioner's Approach-McGraw_Hill Education (2019)

"What is Ferpa?," What is FERPA? | Protecting Student Privacy, https://studentprivacy.ed.gov/faq/what-ferpa (accessed Oct. 6, 2024).

W. W. A. I. (WAI), "WCAG 2 Overview," Web Accessibility Initiative (WAI), https://www.w3.org/WAI/standards-guidelines/wcag/ (accessed Oct. 6, 2024).

2. Overall Description

2.1 Product Perspective

This product is a new, interactive website designed for educational purposes. It allows students to practice the MAAP-A standardized test. This website specifically targets students in the third grade. However, the system may expand in the future to accommodate students from grade four and five. Key components of the system include interactive tests materials meant to simulate an environment of the standardized test. Here is a simple diagram of how our website will function.

2.2 Product Functions

The product will allow students the following functions. Further detail will be described in the upcoming sections.

- Listen and read along within the story comprehension section.
- Listen and read along answers following this section.
- Listen and view math problems in the math comprehension section.
- Listen and view math answers following this section.
- Select answers from given options from both comprehension sections.
- Receive scores based on the accuracy of the answers.
- Track performance with a scorecard or saved on the student's account.

2.3 User Classes and Characteristics

The website will cater to two distinct user classes

- **Students**
  - Technical Expertise: Students must be familiar with using electronic devices.
  - Frequency of Use: As frequent as the student wants for practice purposes.
  - Primary User: Students will be allowed to use the interactive features of this website by answering questions, listening to the questions, and receiving immediate feedback based off the student's accuracy.

- **Teachers/ Administrators**
  - Technical Expertise: Responsible for managing student accounts as well as viewing performance data.
  - Frequency: Viewing test scores and reviewing students' progress.
  - Privilege Level: Higher than students to access multiple performance data.

2.4 Operating Environment

This website will be accessible to all web-based platforms (e.g., Chrome, Firefox, Safari) as well as via common electronic devices such as tablets, laptops, or PC. Because it is a web-based system it will not require any additional hardware beyond the ability to handle audio playback and visual display. Alongside this our software is also compatible with the most common operating systems such as Windows or macOS.

2.5 Design and Implementation Constraints

Key constraints for the development of this product include:

- **Technology Constraints:** The product must be web-based, compatible with HTML5 for universal access across different devices and browsers.

- **Audio Integration:** The platform should support text-to-speech APIs or recorded audio files for the reading comprehension and math sections.

- **User Experience Considerations:** The system should cater to students with varying levels of technical literacy, requiring a simple and intuitive interface.

- **Grading System:** The grading system should function accurately without manual intervention and must handle automated scoring.
- **Scalability:** While initially built for third-grade students, the system must be designed to potentially support additional grade levels in the future.

- **Security and Privacy:** The system must comply with student privacy regulations such as FERPA, ensuring secure handling of user data and scores.

2.6 User Documentation

The format of the application will be simple so that one will not need any manuals to understand how to use our website. If anything is not self-explanatory there will be onscreen prompts to guide the user.

2.7 Assumptions and Dependencies

Assumptions

- The students and teachers will have access to devices with functioning audio output (speakers or headphones).
- Internet connectivity will be stable during the usage of the platform, as the system is web-based.

Dependencies

- The system will rely on third-party APIs for text-to-speech functionalities.
- The project may depend on existing web frameworks or libraries for building interactive elements, such as buttons and scorecards.
- Potential future expansion to higher grades will depend on feedback from third-grade implementation and any additional requirements that arise.

3. External Interface Requirements

3.1 User Interfaces

The software is designed for third graders with learning disabilities, so the interface will prioritize simplicity, accessibility, and engagement. The layout will be clear and simple, using large, easy-to-read fonts and a clean design. A limited number of elements per screen will help prevent distractions. All screens will follow a consistent structure, with a header containing navigation options (home, help, settings, profile information etc.) and a central section for content. Error messages will also be limited to appearing in the center of the screen to avoid confusion and provide clarity.

3.2 Hardware Interfaces

The software is a web-based application that will be compatible with a wide range of devices, including desktop computers, laptops, smartphones, and tablets, provided they are accessed through standard browsers. The tool does not require any external hardware devices or assistive technologies to function. Users will interact with the tool using common input devices such as a keyboard, mouse, touchscreen, and possibly a microphone for voice input. The software relies on standard HTTP/HTTPS protocols for communication, with no external devices or assistive technologies required. Since it runs entirely in the browser, there are no specific hardware or performance requirements for the user’s device.

3.3 Software Interfaces

The software will connect to a local SQLite database for storing user information, including quiz results, progress, and account data. The database will be accessed directly by the application using standard SQL queries. The software does not depend on or interact with external operating systems beyond what is required by standard web browsers.
The development process will rely on GitHub for version control and collaboration, ensuring that changes to the codebase are tracked and managed efficiently.
The front-end of the application will be developed using React, which will handle user interface rendering and interactions. All communication between the front-end and the SQLite database will occur through standard HTTP requests handled by the back-end service, with data being retrieved or stored as needed.
All data communications within the system will use standard HTTP/HTTPS protocols, ensuring secure transmission of user data between the front-end, back-end, and database. Since SQLite is a file-based database, there are no additional communication protocols or external databases to connect with.

3.4 Communications Interfaces

The state testing study tool will use standard communication protocols, specifically HTTP and HTTPS, for data transmission between the client's browser and the server, with HTTPS ensuring secure communication. Data encryption will be implemented, though the specific method is yet to be determined, to protect sensitive information such as user credentials and quiz results. The system will operate at standard web speeds, with no special bandwidth requirements. In some cases, emails will be sent to users or administrators, formatted in HTML, for notifications or password resets. All data submissions will occur through traditional asynchronous web form submissions, without real-time synchronization mechanisms.

4. System Features

3rd/4th Grade Stimulus Cards

4.1.1 Description and Priority

Priority: High

The 3rd grade and 4th grade stimulus cards are similar but focus on different types of questioning. The feature allows the user to respond to a proctor's questions by selecting cards that represent possible answers. Since this is a core function of the application, it is considered a high priority. The cards will be clickable, and the interaction will be evaluated for correctness. Depending on the correctness of the selected card, the application will provide feedback and move on to the next question or deduct points if the answer is incorrect.

Priority Component Ratings:

- **Benefit:** 9 - Central feature of the application.

- **Penalty:** 8 - Missing this feature significantly impacts user engagement.
- **Cost:** 5 - Requires integration with voice or text-to-speech (TTS) systems and error handling.
- **Risk:** 4 - Moderate risk due to reliance on external systems for creation of speach.

4.1.2 Stimulus/Response Sequences

- **Proctor Reads Question:**
  - Stimulus: The proctor (via recorded voice or TTS) reads a question.
  - Response: The user is prompted to select a card that corresponds to the question.
- **User Selects a Card:**
  - Stimulus: User clicks one of the displayed cards.
  - Response (Correct): System acknowledges the correct selection, displays feedback, and moves to the next question.
  - Response (Incorrect): System deducts points, displays feedback, and allows the user to try again.
- **Error Handling:**
  - If TTS fails, the question is displayed as text.
  - If a question is unresponsive or contains an error, the system skips the question.

4.1.3 Functional Requirements

REQ-1: Implement pre-recorded voice or third-party text-to-speech (TTS) to read questions.

- **Use Case:** The system plays an audio file or synthesizes the question text using TTS when a question is initiated.
- **ERROR-1:** If the TTS or audio system fails, the text of the question is displayed on screen.

REQ-2: Create clickable cards that allow the user to select answers.

- **Use Case:** User clicks on one of the displayed answer cards, and the system checks if the answer is correct or incorrect.
- **ERROR-2:** If a card fails to load or register a click, the system automatically skips to the next question.

REQ-3: Deduct points if the selected answer is incorrect.

- **Use Case:** Upon selecting an incorrect card, the system reduces the user's score and provides feedback on the error.
- **ERROR-3:** TBD (e.g., if point deduction fails, user will receive a notification, but progress will continue).

REQ-4: Provide a submit button that locks in the answer and moves to the next question if the answer is correct.

- **Use Case:** After selecting a card, the user clicks "Submit" to finalize their choice, triggering the system to evaluate the answer and proceed.
- **ERROR-4:** TBD (e.g., if the submit button fails, a temporary message will notify the user, and an auto-advance feature might kick in).

REQ-5: Display text or images that assist the proctor in asking questions.

- **Use Case:** The system displays supporting images or text alongside the proctor's audio to aid the user in answering the question.
- **ERROR-5:** If the image or text fails to load, the system will display placeholder text indicating the missing content.

REQ-6: Provide immediate feedback on whether the user's selection was correct or incorrect.

- **Use Case:** After the user submits an answer, the system visually displays whether the selected answer was correct (e.g., green for correct, red for incorrect).
- **ERROR-6:** TBD (e.g., if feedback fails to display, the system will retry or log an error message).

4.1 Point Calculation System for the cards

4.2.1 Description and Priority

Priority: High

This feature tracks the points awarded for each question and calculates a cumulative total for the entire assessment. The user starts with 3 points for each question, with deductions based on the number of incorrect attempts. After three incorrect attempts, the question is skipped, and the points for that question are set to zero. The total score is calculated and displayed at the end of the test. This feature is essential to the functioning of the application, making it a high priority.

Priority Component Ratings:

- **Benefit:** 8 - Provides critical feedback for the user's performance.
- **Penalty:** 7 - Without this, the assessment lacks meaningful scoring.
- **Cost:** 4 - Requires point tracking and simple calculations.
- **Risk:** 3 - Low risk since the point deduction system is straightforward.

4.2.2 Stimulus/Response Sequences

- **Starting Points for a Question:**
  - Stimulus: User begins answering a question.
  - Response: The system initializes the points at 3 for the current question.
- **Incorrect Answer (First Attempt):**
  - Stimulus: User selects an incorrect card on the first attempt.
  - Response: Points for that question are reduced from 3 to 2.
- **Incorrect Answer (Second Attempt):**
  - Stimulus: User selects an incorrect card on the second attempt.
  - Response: Points for that question are reduced from 2 to 1.
- **Incorrect Answer (Third Attempt):**
  - Stimulus: User selects an incorrect card on the third attempt.

  - Response: Points for that question are set to 0, and the system automatically moves to the next question.
- **End of Test:**
  - Stimulus: User completes the assessment.
  - Response: The system calculates the total score and displays it to the user.

4.2.3 Functional Requirements

REQ-1: The user should start with 3 points by default for each question.

- **Use Case:** When a question begins, the system sets the initial score to 3 for the current question.
- **ERROR-1:** TBD (e.g., if points initialization fails, a default score of 3 will be assumed).

REQ-2: After one incorrect answer, the points should be reduced to 2.

- **Use Case:** If the user selects an incorrect card on the first attempt, the system deducts 1 point from the current question's score.
- **ERROR-2:** TBD (e.g., if point deduction fails, the system will notify the user and proceed to deduct points on the next attempt).

REQ-3: After two incorrect answers, the points should be reduced to 1.

- **Use Case:** If the user selects an incorrect card on the second attempt, the system deducts an additional point, reducing the total to 1.
- **ERROR-3:** TBD (e.g., if point deduction fails, the system logs the error and continues to the next deduction step).

REQ-4: After three incorrect answers, the points should be reduced to 0, and the system moves to the next question.

- **Use Case:** If the user selects an incorrect card on the third attempt, the system sets the score to 0 and skips to the next question.
- **ERROR-4:** TBD (e.g., if the skip function fails, the system retries or notifies the user and proceeds manually).

REQ-5: The total points should be calculated at the end of the assessment.

- **Use Case:** Upon completing all questions, the system sums up the points for each question and displays the final score to the user.
- **ERROR-5:** If an error occurs while calculating the total score, the system will display a message indicating the error and retry the calculation.

4.2 Sending Results

4.3.1 Description and Priority

Priority: High

This feature allows teachers to receive the results of a student’s test via email. Once the test is completed, the system will gather the total score and send an email to the teacher. This is crucial for teachers to evaluate student performance, making it a high-priority feature.

Priority Component Ratings:

- **Benefit:** 8 - Facilitates teacher assessment of student progress.
- **Penalty:** 7 - Without this, teachers must manually collect scores, reducing efficiency.
- **Cost:** 5 - Requires email service integration and validation.
- **Risk:** 4 - Moderate risk, mainly related to email delivery and data formatting.

4.3.2 Stimulus/Response Sequences

- **Sending Results via Email:**
  - Stimulus: User completes the test and clicks the "Send" button.
  - Response: The system gathers the total score and sends it via email to the teacher’s provided email address.
- **Teacher Email Input:**
  - Stimulus: Teacher’s email address is entered before sending results.
  - Response: The system validates the email address and confirms it before sending the test results.
- **Error Handling:**
  - If the email address is invalid or the email fails to send, the system will notify the user and prevent sending.

4.3.3 Functional Requirements

REQ-1: There should be a clickable "Send" button to trigger the sending process.

- **Use Case:** When the user finishes the test, they click the "Send" button to initiate sending the results.
- **ERROR-1:** TBD (e.g., if the button fails, an error message will notify the user, and the action will retry).

REQ-2: The "Send" button should gather the total score and send the results via email when clicked.

- **Use Case:** After clicking the "Send" button, the system retrieves the student’s total score, formats it, and emails it to the teacher.
- **ERROR-2:** TBD (e.g., if the email fails to send, a message will notify the user, and the system will attempt to resend).

REQ-3: There should be an input field where the teacher’s email can be entered.

- **Use Case:** Before sending results, the teacher inputs their email address into a designated input field.
- **ERROR-3:** If an invalid or missing email is detected, the system prevents sending and displays an error message.

REQ-4: The email should display the student’s score in a professional format.

- **Use Case:** The email sent to the teacher should contain a well-structured summary of the student's performance, including the total points and any additional relevant information.
- **ERROR-4:** TBD (e.g., if the email formatting fails, the system will default to a simple text format and log the issue).

5. Other Nonfunctional Requirements

5.1 Performance Requirements

1. Numerical values, text, imagery, and prerecorded audio will all be required for the program's interactions. Persistent storage should be capable of handling content of all of these types.
2. System should be able to handle multiple users simultaneously with minimal lag

5.2 Safety Requirements

The system should be compliant with the WCAG 2.2. A summary of WCAG can be found at https://www.w3.org/WAI/standards-guidelines/wcag/ , and a searchable, filterable quick reference can be found at https://www.w3.org/WAI/WCAG22/quickref/?currentsidebar=%23col_customize
Ideally the system should meet as many criteria with Level A compliance, Level AA as a stretch goal.
Level A guidelines applicable to this project are detailed here, with Level AA and up guidelines as well as suggested Dos and Don'ts to fulfil each guideline available on the quick reference above. Guidelines are labeled with a number corresponding to the one in the quick reference.

Text Alternatives
1.1.1 Non-text Content: All non-text content that is presented to the user has a text alternative that serves the equivalent purpose, except for in situations of Controls/Input, Time-based Media, Tests, Sensory, CAPTCHA, or Decorative content. In these cases, content should provide

text giving descriptive identification and/or be made ignorable by assistive screen reading technology.

Time-Based Media
1.2.1 Audio-only and Video-only (Prerecorded): An alternative for time-based media is provided

that presents equivalent information for prerecorded audio-only or video-only content.
1.2.2 Captions (Prerecorded): Captions are provided for all prerecorded audio content in synchronized media, except when the media is a media alternative for text and is clearly labeled as such.
1.2.3 Audio Description or Media Alternative (Prerecorded): An alternative for time-based media or audio description of the prerecorded video content is provided for synchronized media, except when the media is a media alternative for text and is clearly labeled as such.

Adaptability
1.3.1 Info and Relationships: Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text.
1.3.2 Meaningful Sequence: When the sequence in which content is presented affects its meaning,

a correct reading sequence can be programmatically determined.

1.3.3 Sensory Characteristics: Instructions provided for understanding and operating content do not rely solely on sensory characteristics of components such as shape, color, size, visual location, orientation, or sound.

Distinguishability

1.4.1 Use of Color: Color is not used as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.

1.4.2 Audio Control: If any audio on a Web page plays automatically for more than 3 seconds, either a mechanism is available to pause or stop the audio, or a mechanism is available to control audio volume independently from the overall system volume level.

Providing Enough Time

2.2.2 Pause, Stop, Hide: For moving, blinking, scrolling, or auto-updating information that (1) starts automatically, (2) lasts more than five seconds, (3) is presented in parallel with other content, and/or (4) is presented in parallel with other content, there is a mechanism for the user to pause, stop, or hide it or to control the frequency of the update unless the content is part of an activity where it is essential.

Seizures and Physical Reactions

2.3.1 Three Flashes or Below Threshold: Web pages do not contain anything that flashes more than three times in any one second period, or the flash is below the general flash and red flash thresholds.

Navigability

2.4.1 Bypass Blocks: A mechanism is available to bypass blocks of content that are repeated on multiple Web pages.

2.4.2 Page Titled: Web pages have titles that describe topic or purpose.

2.4.3 Focus Order: If a Web page can be navigated sequentially and the navigation sequences affect meaning or operation, focusable components receive focus in an order that preserves these qualities.

2.4.4 Link Purpose (In Context): The purpose of each link can be determined from the link text alone or from the link text together with its programmatically determined link context, except where the purpose of the link would be ambiguous to users in general.

Input Modalities

2.5.1 Pointer Gestures: All functionality that uses multipoint or path-based gestures for operation can be operated with a single pointer without a path-based gesture, unless a multipoint or path-based gesture is essential.

2.5.2 Pointer Cancellation: For functionality that can be operated using a single pointer, at least one of the following is true:

- No Down-Event: The down-event of the pointer is not used to execute any part of the function;
- Abort or Undo: Completion of the function is on the up-event, and a mechanism is available to abort the function before completion or to undo the function after completion;
- Up Reversal: The up-event reverses any outcome of the preceding down-event;
- Essential: Completing the function on the down-event is essential.

2.5.3 Label in Name: For user interface components with labels that include text or images of text, the name contains the text that is presented visually.

2.5.4 Motion Actuation: Functionality that can be operated by device motion or user motion can also be operated by user interface components and responding to the motion can be disabled to prevent accidental actuation, except when the motion is used to operate functionality through an accessibility supported interface or the motion is essential for the function and doing so would invalidate the activity.

Readability
3.1.1 Language of Page: The default human language of each Web page can be programmatically determined.

Predictability
3.2.1 On Focus: When any user interface component receives focus, it does not initiate a change of context.
3.2.2 On Input: Changing the setting of any user interface component does not automatically cause a change of context unless the user has been advised of the behavior before using the component.
3.2.6 Consistent Help: If a Web page contains any of the following help mechanisms, and those mechanisms are repeated on multiple Web pages within a set of Web pages, they occur in the same order relative to other page content, unless a change is initiated by the user:

- Human contact details;
- Human contact mechanism;
- Self-help option;
- A fully automated contact mechanism.

Input Assistance
3.3.1 Error Identification: If an input error is automatically detected, the item that is in error is identified and the error is described to the user in text.
3.3.2 Labels or Instructions: Labels or instructions are provided when content requires user input.

Compatibility
4.1.2 Name, Role, Value: For all user interface components (including but not limited to: form elements, links and components generated by scripts), the name and role can be programmatically determined; states, properties, and values that can be set by the user can be programmatically set; and notification of changes to these items is available to user agents, including assistive technologies.

5.3 Security Requirements

Information Access – Teachers should only be able to view records belonging to students attributed to them, and student should only be able to access their own records and none belonging to other students without authorization from those students.
User privacy – Only essential student information should be stored in the system, and in those cases said information must be properly encrypted to ensure security.
FERPA: https://studentprivacy.ed.gov/faq/what-ferpa

5.4 Software Quality Attributes

Usability: function over form, system should be usable for the students and instructor first and foremost. This will be important to give users a smooth experience.
Correctness: Test content and scoring should be correct what is expected at all times. This is important for both student experience and the instructor's ability to ascess student performance.
Maintainability: system should be concise and easy to maintain. This will be important both to developers and operations managers.

Portability: System must be usable from different devices and browsers with minimal differences. This is important to students who may need to access it from different devices and browsers.
Reliability: system should function reliably under most conditions such as network traffic, differing production environments, etc. This will be important for students to use the system independently at will.
Testability: System should be concise and easy to test in addition to being tested frequently. This is important to the development team to ensure high software accuracy and quality.
Availability: System should ideally be available 100% of the time to allow maximum flexibility for students and teachers,

5.5 Business Rules

Teachers/Administrators
Can:

- provide access to students
- receive score reports

Cannot:

- Access student information from students not assigned to them

Students
Can:

- access practice test
- provide name and code
- answer questions
- view their score report after taking test

Cannot:

- access the scores of other users

6. Other Requirements

Released sample items found on the Mississippi Department of Education website may be used as content for the practice test questions. Written permission from the MDE is required to use these materials and as of writing this has been successfully obtained.

Released sample items can be found at https://www.mdek12.org/OSA/SP/MAAP-A

Appendix A: Glossary

FERPA – Family Educational Rights and Privacy Act, governs the collection and disclosure of educational records for children and their families in the US
WCAG 2.2 - Web Content Accessibility Guidelines version 2.2
WCAG Levels – Levels A, AA, and AAA correspond to different standards of compliance. A refers to the bare minimum, AA is the recommended level, and AAA is the highest standard.

Appendix B: Analysis Models

At this time, N/A.

Appendix C: To Be Determined List

At this time, N/A.