Software Requirements Specification (SRS) Document

Teaching Assistance tool for English Language Learning, Team
24 and Ayush Agarwal, Janapati Sai Akash, Polavarapu Neeraj,
Rahul Mallavarapu

Note: This is a "living document", meaning its content will change with the implementation of the project. Use it to capture key project requirements and make sure that your product features match the requirements exactly – if you wish to add any features, they must be added first to the requirements. Changes in the document must be approved by the customer (mentor) and the instructor or TA Mentor.

Remove this text and the descriptive paragraphs in each section stating what to do before you turn it in.

Brief problem statement

Replace this text and the instructions below with your statement in black.
(2-5 lines describing the problem being addressed. Note that even if you are simply restating what is already in the project description, you must rephrase it in your words. This gives an opportunity for the customer to identify and provide feedback on differences in interpretation, if any).

The developed app aims to help teachers in a classroom environment by providing them digital tools to conduct English classes. The app shall act as a learning assistance platform and interface for an existing speech evaluation API (voisTUTOR) that, evaluates the performance of the student speech under metrics such as pronunciation, intonation, pauses, etc.

System requirements

Replace this text and the instructions below with your statement in black.
(Identify the system requirements for your solution. If you require particular technologies, languages and libraries, list them as well).

- It is a web app, which can be accessed through any browser.
- A smart device with internet connection, microphone.

Users profile

Replace this text and the instructions below with your statement in black.
(Identify who will be using the system, in what mode, and their profile in terms of familiarity with using computers and such software).

The developed application is to be deployed in a school setting with the following two class of users:

1. Admin -. Should have familiarity with usage of website and Microsoft Office Suite (Excel). Will perform tasks such as teacher account maintenance and course and student roster updates.

2. Teacher - Should have a basic understanding of how to operate smartphones/computers and work with websites and Microsoft Office Suite (Excel).

Feature requirements (described using use cases)

Read the instructions below and fill in the table. Delete all the blue text turning it in.

(This is a numbered list of use cases that are the features of the system to be implemented. Each use case is an operation that the user can perform on/with the system. For each use case, provide a description (2-3 sentences) so you know what to build and so you can write a test case to demonstrate that your system provides that feature. For each use case, you will identify (during release planning) the release in which it will be implemented: R1 or R2. Typically, your project will have 10-15 use cases, but feel free to add or delete table rows if you decide to use finer-grain or socoarse-grain use cases).

Admin Use Cases:

No.	User Case Name	Description	Release
1.	Admin - (login)	Can login to the platform with an email and password combination.	
2.	Admin - (teacher registration)	Add teachers to the platform and assign/update their corresponding classes (as in courses as well as student rosters)	
3.	Admin - (student registration)	Add the student roster to the platform and assign them to their corresponding classes.	
4.	Admin – (class registration)	Add new classes to the platform.	
5.	Admin - (edit accounts)	Can edit all accounts associated with the platform.	

Teacher Use-Cases

No.	User Case Name	Description	Release
1.	Teacher - (login)	Can login to the platform with an email and password combination.	
2.	Teacher's landing page after login	• View list of classes assigned to him/her. • Update user information and password.	
3.	Teacher - Student reports page	Performance reports of the individual's performance throughout the term of the class.	
4.	Teacher - Class page	• Display average analytics for the whole class. • Course plan/material – allowing to plan the content to be used in a particular lesson ahead of time. • Class student roster.	

Use case diagram

Read the instructions below and fill in the table. Delete all the blue text before adding this to your repository or turning it in to your instructor.

Draw the UML use case diagram for the system. Make sure the use cases shown in the diagram correspond to the use cases described in the previous section.

Use case description

Delete all the blue text and fill-in the template before adding this to your repository or turning it in to your instructor.

Use Case Number:	UC-XX (Replace XX with a number)
Use Case Name:	Enter the name of Use Case
Overview:	Describe the purpose of the Use Case and give a 1-2 line description. This could be the same as the description provided in feature requirements section.
Actors:	List all actors that participate in this Use Case.
Pre condition:	Enter the condition that must be true before the main flow is executed.
Flow:	Main (success) Flow: Steps should be numbered.
	Alternate Flows: Include the post condition for each alternate flow if different from the main flow.
Post Condition:	Enter the condition that must be true when the main flow is completed.

Test Backlog

Read the instructions below and fill in the table. Delete all the blue text turning it in.

You need to document and create test cases for your system. This would be in-line with the test cases created in gitlab.