Lepic

supporting the assessment and management of reading fluency

Sponsors

Fabiana Mendes - University of Brasília (UnB), Brazil
George Marsicano - University of Brasília (UnB), Brazil
Letícia Celeste - University of Brasília (UnB), Brazil
Luciana Mendonça Alves - Federal University of Minas Gerais (UFMG), Brazil

Project Abstract

Reading skills development opens up the world to the students, and, beyond that, it expands and enhances learning development. If someone has good reading skills, he/she can also improve the learning skills because he/she will be able to interpret texts and questions better. This project's main goal is to develop a tool to support reading fluency assessment and management, as shown with manual analysis in previous work (Celeste et al., 2018). The software aims to help professionals that work with students' literacy and kid's relatives.

Reference: Celeste, L. C., Pereira, E. S., Pereira, N. R. R., & Alves, L. M. (2018). Prosodic parameters of reading in 2nd to 5th grade students. In CoDAS (Vol. 30, No. 1). Sociedade Brasileira de Fonoaudiologia.

Project Description

For this project, each team should develop a solution to the scope presented in the next section. The core features are the following: (1) CRUD of class, user, student, text, and questionnaire; (2) data collection (audio, mistakes, and questionnaire answers); and (3) report (measures, tables, and graphics).

Project Scope

This project aims to develop a mobile app to support reading fluency assessment and management in a semiautomatic way. The main functionalities are:

1. **CRUD of class** with level, class, teacher.
2. **CRUD of user** with a name, profile
3. **CRUD of a student** with name and class.
4. **CRUD of text.** A teacher can register and edit a text which will be read by students.
5. **Identify words in a text.** The software must identify automatically the beginning and ending of each word in each registered text.
6. **Record reading audio.** The teacher should record audio of her/his students reading the texts registered in the software.
7. **Indicate student's mistakes made during her/his reading.** The teacher should be able to indicate the mistakes made by the students and link them to a specific word in the text.

8. **Calculate measures related to the student's reading fluency.** The software must calculate the following measures automatically:
    a. Total reading time (automatic)
    b. Number of words read per minute (automatic)
    c. Number of words read in the first minute (automatic)
    d. Number of correct words read per minute (semi-automatic)
    e. Total number word incorrectly read (semi-automatic)
9. **Calculate Z-Score of the number of words read correctly per minute (ZSNW).** For each class, the software needs to associate a Z-Score of the number of words read correctly per minute.
10. **Generate an individual report from a template.** The template is a simple text like the following one:

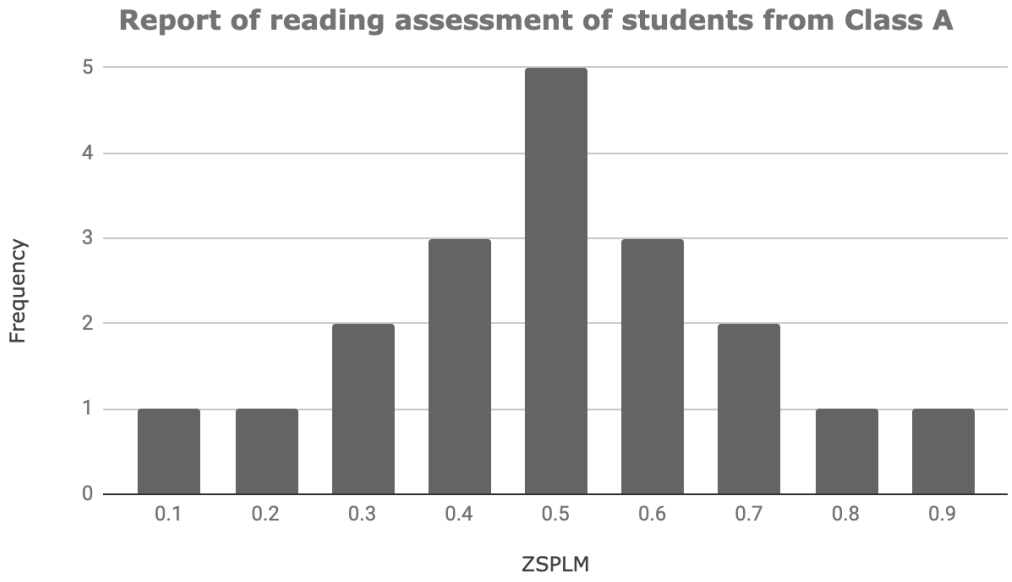
    The student _____ had his/her reading fluency assessed in dd/mm/yyyy. He/She read the text _____ in XX seconds.

11. **Register questionnaires to assess the interpretation skills of each registered text.** The teacher can create multiple choice questions for each registered text. The software will collect the answers and calculate a grade (from 0 to 100).
12. **Generate comparative reports.** This report will use the measure ZSNW to compare students from the same class.
13. The reports mentioned in the previous item should be in the **format of table and graphic** according to the following examples:

Example of a comparative report of students from the same class on a tabular format

Student	Level	Class	ZSNW
Student 1	1	A	0.12
Student2	1	A	0.25
Student 3	1	A	0.37
Student 4	1	A	0.41
Student 5	1	A	0.59
Student 6	1	A	0.67
Student 7	1	A	0.72
Student 8	1	A	0.82
Student 9	1	A	0.94

Example of a comparative report of students from the same class on a graphical format



The potential users of this software are software administrador (A), parents or other kids' relatives (Pa), teachers (T), speech therapists (ST), psychopedagogist (PP), pedagogues (Pe), and researchers (Re). Table 1 presents the maximum access level for each user and for each functionality presented before. In this table, **N** indicates that the user has no access to the functionality, **R** indicates that the user has read access and **E** indicates edition access.

Table 1: Access level of each potential user

Functionality	User						
	A	**Pa**	**T**	**ST**	**PP**	**Pe**	**Re**
1	E	N	E	E	E	E	R
2	E	E	E	E	E	E	E
3	E	R[1]	E	E	E	E	N
4	E	R	E	E	E	E	N
5	E	R	R	R	R	R	R
6	E	E	E	E	E	E	N
7	E	E	E	E	E	E	N
8	E	R	R	R	R	R	R
9	E	R	R	R	R	R	R
10	E	R	R	R	R	R	N

11	E	R	E	E	E	E	N
12	E	N	R^2	R	R	R	R
13	E	N	R^2	R	R	R	R

Notes: *1. Only the student that the parent is responsible for*
2. Only classes that the teacher is responsible for

Process Requirements

All development will happen in a public GitHub repository. Any agile-like process is fine, as long as it is adequately explained and motivated. It is required for the team to provide the user stories with acceptance criteria. Furthermore, the team needs to employ practices of clean code.

Environmental Constraints

The application must run on Android and iOS smartphones and written in Flutter. The backend should be built in Django and Phyton.

Project Restrictions

The code must be available on Github.

Project License

The software license must be the MIT License.

Level of Sponsor Involvement

Teams can contact sponsors via two different channels: (1) Google Form and (2) Skype.

The team can send **questions** to the sponsors using this form. We will attempt to answer questions within a week throughout the contest period. Selected questions deemed important by the sponsors will be gathered and answered in a FAQ at the bottom of this page. If the sponsors do not answer within a week, the team is free to make their own assumptions. However, these assumptions should be registered in the project report.

Meetings can be proposed by the team to the sponsors through this form at the beginning of the contest. The team will send a schedule with a maximum of four dates and three different times for each date. The sponsors will select two dates among those suggested by the team. The meetings' duration is 30min, and they will happen via Skype. The meetings should happen from Monday to Friday, between 8 am and 6 pm in the Brazilian timezone (Brasília). The sponsor will not be able to provide technical assistance.