



CS 353 Database Systems
Project Proposal
Online Language Learning Platform

Group 31

Erengazi Mutlu - 21803676

Tuna Öğüt - 21803492

Bilgehan Yılmaz Sandıkçı - 21902354

Ömer Burak Yıldırım - 21901425

https://bilgehansandikci.github.io/CS353_Group31/

Table Of Contents

Table Of Contents	2
1. Introduction	3
2. Project Description	3
3. Requirements	4
3.1 Functional Requirements	4
3.2 Non-Functional Requirements	5
3.3 Constraints (Pseudo Requirements)	6
4. Limitations	6
5. Entity Relationship Model	7

1. Introduction

This project proposal is about an Online Language Learning platform. This proposal will include a brief description about the project, functional and non-functional requirements, limitations of our application's database and finally the entity relationship model of our database.

Project description will include information about the project, why the database is an essential part of the project and how we will use it.

In the requirements part, functional, non-functional and pseudo requirements will be introduced. Functional requirements part plays an essential role to the project because it includes functionalities and properties of the project. In the non-functional part we will be discussing the performance, security, reliability and usability goals for our application. Lastly, in the pseudo requirements, we will introduce which technologies are going to be used in our project.

Limitations part will include the boundaries and constraints that our application will have. At the end, we will add our E/R diagram, which is the fundamental of our project.

2. Project Description

This project is aimed to be an online language learning platform that is used by students, teachers, and native speakers of a particular language. The main goal of the application is to establish a bridge between students and teachers or native speakers. Students will request classes from teachers depending on the language they prefer and their level of knowledge will be considered for the classes they request. Students can also request speaking sessions with native speakers of a particular language. So, to make students see those lists of classes, we need a database. Also, with the help of the database, teachers will see the number of students and their information who are enrolled in their class. If teachers give assignments to the students, they will be stored on the database and also, grading of these assignments. Lastly,

there will be a platform admin who is capable of monitoring the grading average of the students and information about the teachers.

3. Requirements

3.1 Functional Requirements

3.1.1 User

- Each user except admin needs to sign up and login to the system.
- Users can modify their personal info.

3.1.2 Admin

- Admin status will be granted by the system.
- Admins can see the statistics of grades, exams, and classes.
- Admins can access the entire database and manipulate it.

3.1.3 Teacher

- Teachers can assign homework.
- Teachers can grade exams and homeworks.
- Teachers can see how many students are attending their classes.
- Teachers can see which activities that students are attending.

3.1.4 Native Speakers

- Native speakers can hold speaking sessions.
- Native speakers can grade speaking sessions.
- Native speakers can teach if they are also teachers.

3.1.5 Students

- Students can request classes from teachers according to their level.

- Students can take exams and homeworks.
- Students can attend activities.
- Students can request speaking exercises from native speakers.

3.2 Non-Functional Requirements

3.2.1 Security

Username and password will be unique to the user therefore, users have their own account.

With the various account types, users can only manipulate things that their account type allows them to do. Also, we will be using a database server which will enhance the security against malicious action.

3.2.2 Performance

Our system will return each response immediately. Thanks to this, there will be no noticeable delay in our system.

3.2.3 Usability

The UI should be advanced enough to meet the needs of the users and also it should be easy enough to use. Therefore, there will be a balance between those particular trade-offs.

3.2.4 Scalability

This application needs to be modular and modifiable enough to grow in case of data size. So that in the future, the project can handle the data size.

3.2.5 Maintainability

The system should be developed in a way that if a problem or error occurs, it must be easy enough to handle. Also, if there is an update needed, it should be easy enough to deploy to the production.

3.3 Constraints (Pseudo Requirements)

- NodeJS, Javascript, HTML and CSS will be used in backend and frontend.
- PostgreSQL will be used for Structured Query Language

4. Limitations

- Each user must sign up with a password and unique username.
- Students can not change their grades.
- Native speakers can not teach unless they are also Teachers.
- Only teachers can give homework and exams to students.
- Native speakers can grade speaking sessions.
- Homeworks that is uploaded from teachers and students can not exceed 20mb.
- Username and password can not exceed 32 characters.
- A student can not see other students' grades.

5. Entity Relationship Model

