
Bilkent University
Department of Computer Engineering

# Senior Design Project

*Project short-name: Prexcel*

## Analysis Report

Burak Yiğit Uslu, Can Kırımca, Can Kırşallıoba, Alper Sarı, Barış Tiftik

Supervisor: Dr. Özcan Öztürk

Jury Members: Dr. Shervin Arashloo and Dr. Hamdi Dibeklioğlu

Analysis Report
November 15, 2021

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

# Table of Contents

# Analysis Report

*Project Short-Name: Prexcel*

## 1. Introduction

In the rapidly changing world, and especially considering the COVID-19 pandemic; the work of many occupations and companies have transitioned towards an online workspace. Thus, the popularity of online presentations has been increasing rapidly. Following these new trends, our app, named Prexcel, is a multifunctional online presentation assistant that aims to help users give better presentations, and improve their presentation ability, especially in an online environment.

In general, giving online presentations have a tendency to be more difficult compared to giving in-person presentations. For example, engaging the audience's attention is more challenging in online presentations since the audience can be distracted much easier when they attend the presentations online. Therefore, giving effective online presentations requires effective communication skills and the ability to make a succinct, easy-to-follow delivery. Prexcel aims to help its users improve themselves in these regards. Prexcel as an online presentation assistant, helps the user to improve their online presentation skills by evaluating their presentations and giving feedback on how to improve them. It also has the functionality of giving live feedback and -when stuck- word recommendations to the user during a live presentation.

In the rest of this report, under section 2, the technical details of the project such as the non-functional, functional and the functional requirements, and various system models are discussed, under their respective subsections. General discussion and description of the app is located in section 2.1, "Overview". Under section 3, various other analysis elements such as the consideration of various factors in the engineering design, risks and alternatives to the current development plan, project plan, our strategy on how to ensure proper teamwork, various ethical and professional responsibilities and our learning strategies are discussed in their relevant subsections. The report also contains a glossary section.

# 2. Proposed System

## 2.1. Overview

Prexcel is a presentation assistance application that helps its users improve their presentation abilities and their presentations. The app accomplishes this by giving feedback to user’s presentations and by providing live feedback to the user when presenting. It is geared towards online presentations in particular.

App has two main functionalities. The first main functionality is analysing the user’s presentation from beginning to the end and providing detailed feedback that contains the app's grades for the presentation, recommendations and feedback regarding how to improve the presentation as well as various relevant statistics. This analysis is based on the user’s speech, as well as his/her body language and facial orientation. All of these are presented to the user in a detailed analysis report. The grading is done for 3 categories. The first category is the delivery, which is based on the number and length of pauses in the presentation and how fast or slow the user's speech is. Second category is the use of language, which is based on how often the user repeats the same words, and the amount of filler words in his/her speech. Last category is body language, which is graded based on how much the user does not look at the screen, and whether he/she closes his face with his/her hand. All of the filler words, and words that are frequently repeated and such are also marked on the transcript of the speech, which is also provided in the detailed analysis report. For creating the transcript of the speech, speech recognition is used to transcribe the presenters words into text, from there the program recognises filler words and phrases as well as noting down extended pauses used in their speech. The end report provides alternative word recommendations to the user for words he/she uses excessively, as well as gives feedback to the user on talking faster or slower. It provides this feedback based on statistical data, such as “If you had made only half of the pauses you made, the presentation could have been 5 minutes shorter.” or that “If you used %30 less filler words, the presentation could have been shortened by 10 minutes.” and such. The user has the option to download the detailed analysis report.

Second main functionality of the app is the live feedback functionality. If the users choose to perform the presentation live, while using the app, the app can be used to give instantaneous feedback to the user regarding his/her voice level, and “word per minute” speed. It also recommends words to the user based on the flow of the sentence when the user gets stuck or has an extended pause.

The users have the option to either upload a prerecorded video of a presentation, or capture the video of their presentation via the app. The live feedback functionality is available only on the latter case.

Users can also track their progress across different presentations (or different trials for a particular presentation) using the progress report functionality. Fundamentally, users can select a subset of their previous presentations, and view using charts and tables how their grades have changed over these presentations. They can download these progress reports as well.

It is important that we can provide the user with a tangible set of data regarding their presentation. Therefore, we would like to emphasize that the report is mainly based on statistical and mathematical phrasing of the presenters' behaviors, such as for how long have they looked in other places in total, or how much could they shorten their presentations if they got rid of the extended pauses in their speech. For the convenience of the user, this analysis data and their compiled reports will be held in their user accounts, but of course no personal or audiovisual data will be kept to comply with user safety.

## 2.2. Functional Requirements

### 2.2.1. User-Side Requirements

- The user should be able to upload the video recordings of their presentations.
- The user should be able to upload the audio recordings of their speeches.
- The user should be able to start live recordings and record their presentations while giving them.
- The user should be able to receive live feedback based on their live presentations.
    - The user should see recommended words that are relevant to the presentation’s context during the presentation.
    - The user should be able to see feedback about their pace of speech.
    - The user should be able to receive warning if they fail to look at the direction of presenters for a particular duration.
    - The user should be able to receive warning if their voice volume is above or below the normal threshold.
- After uploading or recording a presentation, the user should be able to receive system-generated reports containing the analysis of their presentations based on various metrics.
    - The report should contain the transcript of the speech that the user gave.
    - The report should contain the evaluation of the speech based on various metrics such as words per minute, repetitive words and their frequencies etc.
- The user should be able to view and download their reports on the system.

- The user should be able to create accounts using their username, email addresses and a password.
- The user should be able to sign in to its account with their login information.
- The user should be able to change their account details.
- The user should be able to delete their account.

### 2.2.2. System-Side Requirements

- The system should be able to transcribe the user's speech.
- The system should be able to conduct an analysis of the transcript based on the determined metrics.
- The system should be able to generate word recommendations relevant to the presentation's context and flow of the sentences.
- The system should be able to analyze the visual input based on the user's face orientation.
- The system should keep track of the registered users.
- The system should be able to retrieve and modify the user's presentation data as requested by the user.

## 2.3. Non-functional Requirements

### 2.3.1. Usability

- The user interface should be simple to use and user-friendly.
- The live feedback given by the system should not distract the presenter during the live presentation.
- The application should be usable by anyone who has an English level above A1.

### 2.3.2. Reliability

- The application should continue to function even if the voice input cannot be transcribed or some issue prevents the use of instant feedback from working within the appropriate time frame.
- A catastrophic failure of the application should not cause a loss of data for the user, nor should it require any low level manipulation from the user's side other than restarting the application itself.

### 2.3.3. Performance

- The live feedback model should be able to make recommendations in at most 3 seconds after the presenter stops speaking.
- Processing of the audio files (when the user chooses to upload an audio file) should not take more than 5 seconds for a presentation of 10 minutes.

- Processing of the video files (when the user chooses to upload a video file) should not take more than 10 seconds for a presentation of 10 minutes

#### 2.3.4. Supportability

- Prexcel must support Windows, Linux, and Mac desktop operating systems.
- Any recording hardware (Microphone and camera etc.) that the users utilise should be supported by the application, assuming that the hardware components are compatible with the system on which the application is running on.

#### 2.3.5. Security

- Users' processed data should be deleted right away after its usage.
- Users' password must be encrypted when it is stored in the database.

#### 2.3.6. Scalability

- The application should work even when the user count exceeds projected counts, and the user experience should not suffer due to this fact.

### 2.4. Pseudo Requirements

1. Prexcel will be a desktop application.
2. GitHub will be used for version control, as well as tracking changes made across development.
3. DeepSpeech [1] will be used to convert speech to text.
4. OpenCV [2] will be used to implement the Facial Recognition Model.
5. JavaScript [3] programming language will be used to program the user interface.
6. Electron [4] and React [5] framework will be used for developing the applications UI.
7. Python [6] programming language will be used to implement processes and machine learning models.
8. There will be a database used for user accounts, utilising MySQL [7].

### 2.5. System Models

#### 2.5.1. Scenarios

**Use Case #1**

**Use case name:** StartLiveRecording

**Participating actor:** Initiated by User

**Flow of Events:**

1- The user chooses the UI element that allows them to make a live recording.

2- The user performs the presentation, with their device recording the audio-visual data

3- The presentation is finished, and the recording comes to an end

4- Alternatively, there is no more storage space for the data to be held and the recording is terminated early.

**Entry Conditions:**

The user has chosen to start a recording with the app, by selecting the relevant UI option.

**Exit Conditions:**

The recording is stopped by the user OR

Maximum length of video that is possible to store in memory is reached.

---

**Use Case #2**

**Use case name:** UploadRecording

**Participating actor:** Initiated by User

**Flow of Events:**

1- The user chooses the UI element that allows them to upload an existing recording file.

2- The presentation file is received by the system

3- The data gets analysed and an analysis is run.

**Entry Conditions:**

The user has chosen to upload an already finished recording to the app, by selecting the relevant UI option.

**Exit Conditions:**

The uploaded data is sent to the analysis module.

---

**Use Case #3**

**Use case name:** ViewUserSettings

**Participating actor:** Initiated by User

**Flow of Events:**

1- The user chooses the UI element that allows them to interact with the user settings screen.

2- The user can view their settings and account details on this screen.

**Entry Conditions:**

The user has chosen to interact with the "User Settings" screen.

**Exit Conditions:**

The user presses the button to go back to the main menu screen.

---

**Use Case #4**

**Use case name:** ChangeAccountDetails

**Participating actor:** Initiated by User

**Flow of Events:**

1- The user chooses the UI element that allows them to interact with the user settings screen.

2- The user, if they wish, can modify their account details.

3- The user then chooses to confirm their changes by pressing the "confirm changes" button.

4- The user exits the page, backing up into the main menu screen

**Entry Conditions:**

The user has chosen to view the "User Settings" AND changed their account details.

**Exit Conditions:**

The user chooses to exit the screen, returning to the previous screen.

---

**Use Case #5**

**Use case name:** LogOut

**Participating actor:** Initiated by the user

**Flow of Events:**

1- The user logs out of their account by pressing the "Log Out" button.

2- The user is directed to the login screen of Prexcel.

**Entry Conditions:**

The user is on the "User Settings" screen AND the user presses the "Log Out" button.

**Exit Conditions:**

The logout process is completed.

---

**Use Case #6**

**Use case name:** DeleteAccount

**Participating Actor:** Initiated by the user

**Flow of Events:**

1- The user deletes their account by pressing the "Delete Account" button.

2- The user is directed to the login screen of Prexcel.

**Entry Conditions:**

The user is on the "User Settings" screen AND the user presses the "Delete Account" button.

**Exit Conditions:**

The deletion process of the account is completed.

---

**Use Case #7**

**Use case name:** ViewPresentations

**Participating actor:** Initiated by User

**Flow of Events:**

1- The user has chosen the UI element that allows them to view their existing presentations.

2- The user chooses to exit the page, backing up into the main menu screen.

**Entry Conditions:**

The user has chosen to view previous presentations on "My Presentations" screen.

**Exit Conditions:**

The user chooses to exit the screen, returning to the previous screen.

---

**Use Case #8**

**Use case name:** ViewReport

**Participating actor:** Initiated by User

**Flow of Events:**

1- The user views a presentation report by choosing the desired report on the presentations screen

2- Alternatively, users can see their report after the analysis of an uploaded or recorded presentation.

3- The user chooses to exit the screen after viewing.

**Entry Conditions:**

The user is on the "My Presentations" screen AND has chosen to view an existing analysis report of one of their previous presentations.

**Exit Conditions:**

The user chooses to exit the screen, returning to the previous screen.

---

**Use Case #9**

**Use case name:** SeeTranscript

**Participating actor:** Initiated by User

**Flow of Events:**

1- The user has chosen the UI element that allows them to see the presentation details.

2- The user chooses to press the See Text Transcript button.

**Entry Conditions:**

The user is on the viewing screen of the selected report AND has chosen to display the transcript of the selected presentation.

**Exit Conditions:**

The user chooses to exit the screen, returning to the previous screen.

---

**Use Case #9**

**Use case name:** DownloadReport

**Participating actor:** Initiated by the user

**Flow of events:**

1- The user downloads the report which is being viewed currently.

**Entry Conditions:**

The user is on the viewing screen of the selected report AND has pressed the “Download Report” button.

**Exit Conditions:**

The download is complete.

---

**Use Case #10**

**Use case name:** DeleteReport

**Participating actor:** Initiated by the user

**Flow of events:**

1- The user chooses the report to be deleted on the “My Presentations” screen.

2- The user clicks the “Delete Report” button.

3- The report is deleted from the list of user's presentation reports.

**Entry conditions:**

The user is in the “My Presentations” screen AND has chosen to delete a particular report.

**Exit Conditions:**

The deletion of the report is completed.

---

**Use Case #11**

**Use case name:** ViewProgressReport

**Participating actor:** Initiated by the user

**Flow of events:**

1- The user chooses to see the progress report by pressing the “See Progress” button on the “My Presentations” screen.

2-The user views the progress report containing past presentation scores and a graph illustrating the progress.

**Entry Conditions:**

The user is in the “My Presentations” screen AND has chosen to see the progress report

**Exit Conditions:**

The user exits the “Progress Report” screen to return to the previous screen

---

**Use Case #12**

**Use case name:** DownloadProgressGraph

**Participating actor:** Initiated by the user

**Flow of events:**

1- The user downloads the progress graph by pressing the “Download Graph” button.

**Entry Conditions:**

The user is in the “Progress Report” screen AND has chosen to download the graph

**Exit Conditions:**

The download operation of the progress graph is complete

---

**Use Case #13**

**Use case name:** ReceiveLiveFeedback

**Participating actor:** Initiated by User

**Flow of Events:**

1- The user has chosen the UI element that allows the system to give live feedback to the user.

**Entry Conditions:**

The user has chosen to start a live presentation.

**Exit Conditions:**

The user has chosen to end the live presentation.

---

**Use Case #14**

**Use case name:** GenerateWordRecommendation

**Participating actor:** Initiated by the System

**Flow of Events:**

1- The system feeds required data into the model.

2- The system initiates the Word Recommendation Model.

**Entry Conditions:**

The system has the required data for the model to start AND

The user has started a live presentation.

**Exit Conditions:**

The system terminates the process when the live presentation is over.

---

**Use Case #15**

**Use case name:** DetectFace

**Participating actor:** Initiated by the System

**Flow of Events:**

1- The system feeds required data into the model.

2- The system initiates the Face Recognition data.

**Entry Conditions:**

The system has the required data for the model to start AND

(The user has started a live presentation OR

The user has uploaded a recorded presentation).

**Exit Conditions:**

The system terminates the process when the live presentation is over OR

The system terminates the process when the processing of the recorded presentation is over.

---

**Use Case #16**

**Use case name:** TranscribeVoice

**Participating actor:** Initiated by the System

**Flow of Events:**

1- The system feeds required data into the model.

2- The system initiates the Speech to Text Model.

**Entry Conditions:**

The system has the required data for the model to start AND

(The user has started a live presentation OR

The user has uploaded a recorded presentation).

**Exit Conditions:**

The system terminates the process when the live presentation is over OR

The system terminates the process when the processing of the recorded presentation is over.

---

**Use Case #17**

**Use case name:** GenerateReport

**Participating actor:** Initiated by the System

**Flow of Events:**

1- The system generates the report by receiving data from analyzer units.

**Entry Conditions:**

(The user uploads a presentation OR

The user finishes a live presentation recording) AND

The system has sufficient data to produce the reports.

**Exit Conditions:**

The system successfully generates the report and the process is terminated.

---

**Use Case #18**

**Use case name:** ShowReport

**Participating actor:** Initiated by the System

**Flow of Events:**

1- The system sends the content of the report to the related screen.

**Entry Conditions:**

The System has generated the report.

**Exit Conditions:**

The system successfully generates the report and the process is terminated.

---

**Use Case #19**

**Use case name:** DisplayFeedback

**Participating actor:** Initiated by the System

**Flow of Events:**

1- The system receives processed data from the models.

2- The system sends the feedback to the related screen.

**Entry Conditions:**

The system has generated the related data for displaying feedback AND

The user is recording a live presentation.

**Exit Conditions:**

The system successfully generates the live feedback and the process is terminated.

### 2.5.2. Use-Case Model

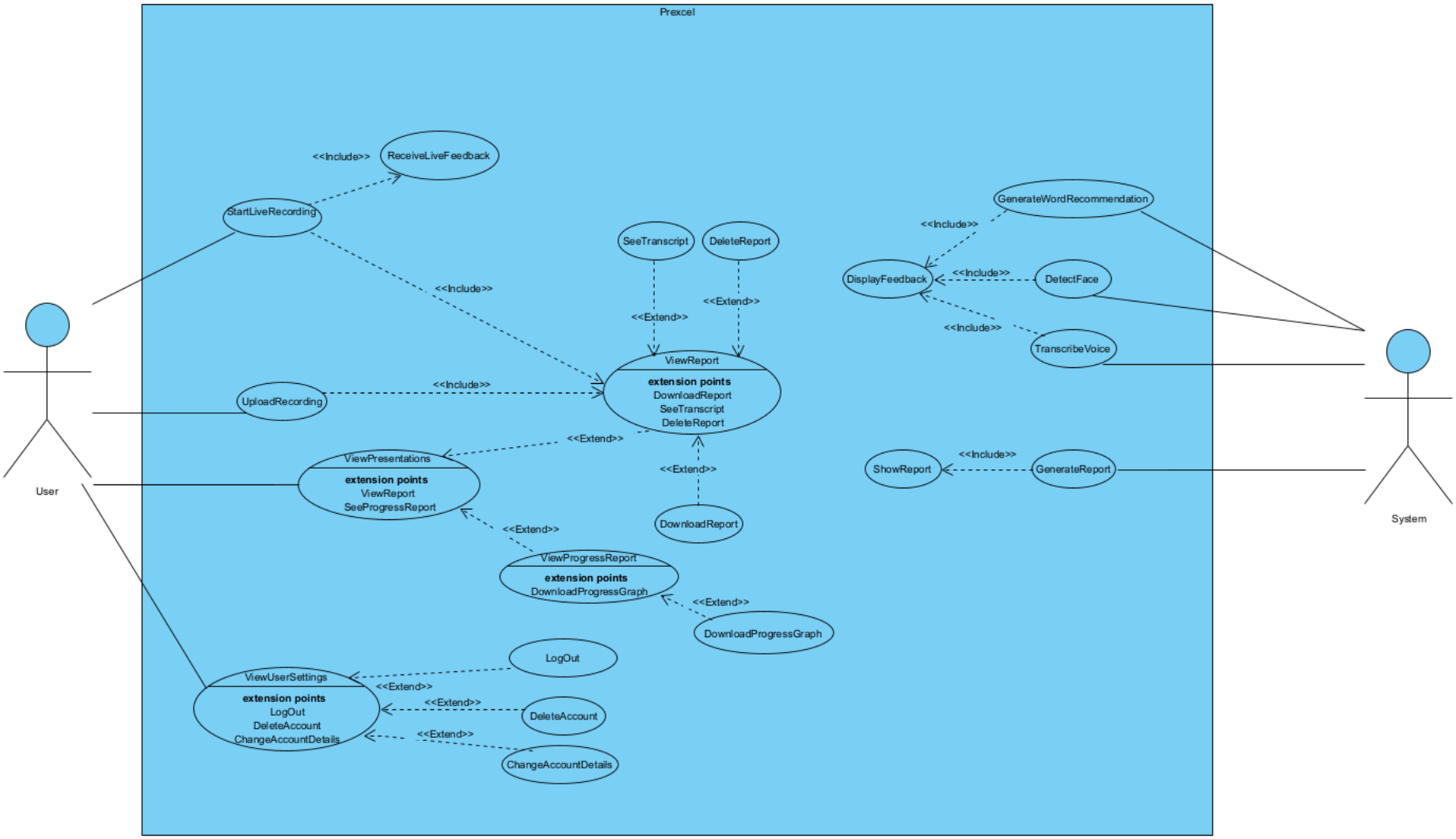


**Figure 1.** Use Case Diagram of Prexcel

### 2.5.3. Object and Class Model

#### 2.5.3.1. Class Diagram

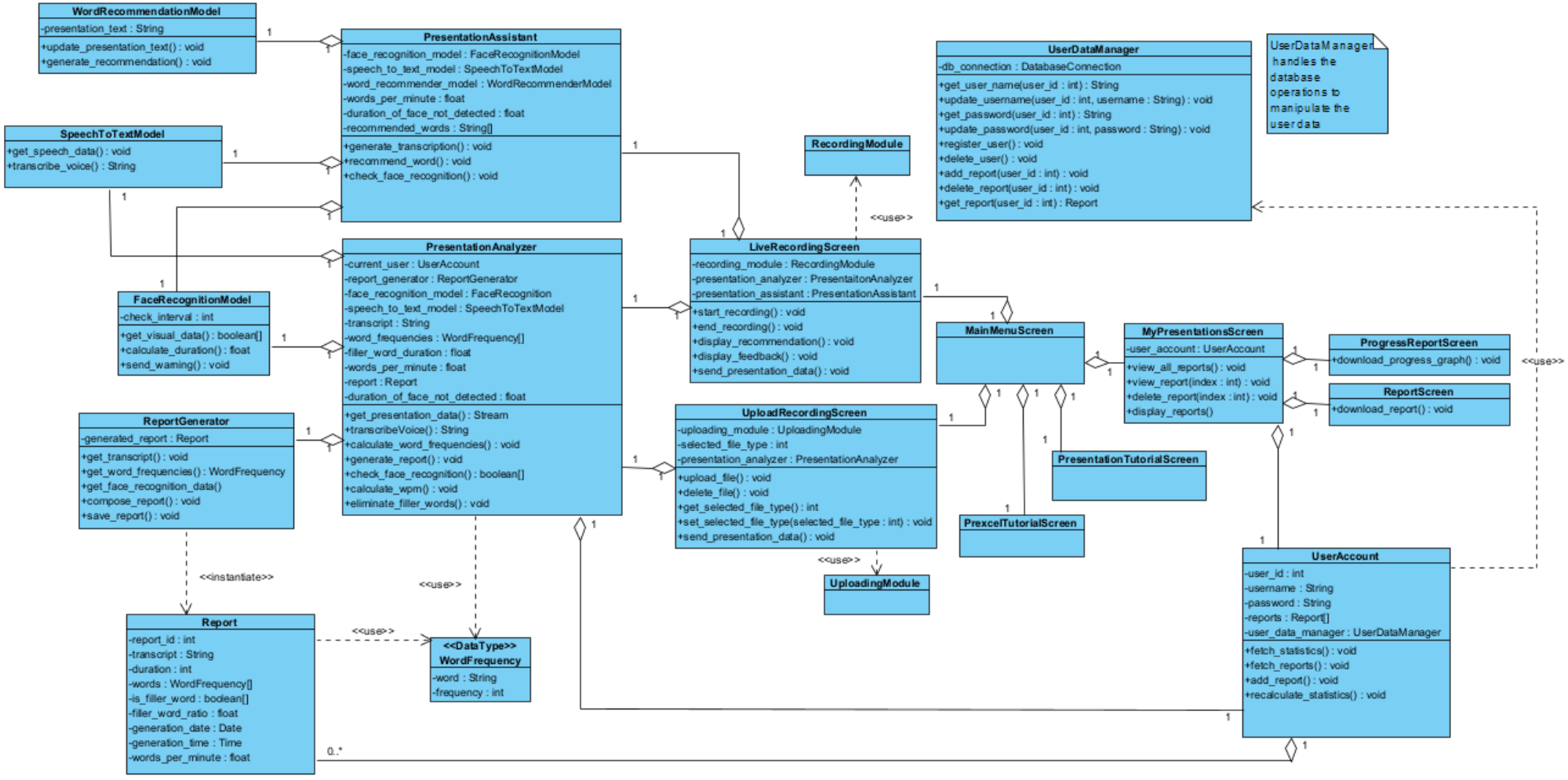


**Figure 2.** Class Diagram of Prexcel

#### 2.5.3.2. Explanations of the Class Diagram

**MainMenuScreen:** This class communicates with the main menu screen in the user interface. It enables the navigation to other user interface components based on the action of the user.

**MyPresentationsScreen:** This class represents the screen where the user can view their past presentations and their reports.

**ReportScreen:** This class represents the user interface component where the users can view and download each individual report.

**ProgressReportScreen:** This class represents the user interface component where the users can view and download their progress reports.

**PresentationTutorialScreen:** This class represents the screen where the user can view the tutorial on presentations.

**PrexcelTutorialScreen:** This class represents the screen where the user can view tutorial on how to use Prexcel.

**LiveRecordingScreen:** This class represents the window in which the user starts and presents his or her presentation. In addition to the presentation, live feedback such as word recommendation, facial rotation warnings etc. are given.

**UploadRecordingScreen:** This class represents the window in which the user uploads a recording of their presentation either in audio files or in video files.

**RecordingModule:** This class contains the necessary functions to enable the users' to record their live presentations.

**UploadingModule:** This class contains the necessary functions to enable the users' to upload their audio and/or video files.

**UserAccount:** This is the class that holds the user information and modifies it based on the operations performed by the user. This class also communicates with the database based on the addition, modification and deletion of user data.

**UserDataManager:** This class acts as an intermediate class between the user account and the database. The class performs the queries on the database based on the requested operations

**PresentationAnalyzer:** This class takes various data from the various models represented in the class diagram (speech to text model: transcript and timestamps of the

speech, facial recognition model: if the presenter is facing the screen or not) and processes them. The processes include calculating various statistics such as the words per minute, calculating the word frequencies. This class acts as an intermediate agent between the upload recording screen and the live recording screen, and the backhand processes such as the machine learning models, and the processes.

**PresentationAssistant:** This class takes various data from the machine learning models that we are going to use (speech to text model: words per minute statistic, facial recognition model: if the user is looking at the screen or not, word recommendation model: recommends words to the user when he or she is stuck) and processes them. The process includes computing the WPM statistics and informing the user if he or she is going too fast or too slow, warning the user about the times when the user is not looking at the screen, and the word recommendation feature is going to give the user.

**WordRecommendationModel:** This class includes the word recommendation machine learning model. This model is going to analyze the user's speech and when the user gets stuck while presenting, this model is going to give him or her word recommendation by analyzing the previously mentioned words. Several machine learning algorithms will be implemented and combined in this class to produce relevant and logical word recommendations.

**SpeechToTextModel:** This class includes the implementation of the model that converts speech to text. Currently, usage of the DeepSpeech model is going to conduct the speech-to-text conversion. This class will get the speech data, apply the conversion and return the transcribed text to the analyzer and assistant classes.

**FaceRecognitionModel:** This class includes the implementation of the face recognition model. This model is going to detect the presenter’s face with the aid of OpenCV, and when it does not detect the presenter’s face, it is going to flag it as the presenter is not looking at the screen. Later it is going to give this information to the PresentationAnalyzer to be processed.

**ReportGenerator:** This class is going to generate the report which includes the transcript of the speech and the statistics which are calculated in the PresentationAnalyzer class.

**Report:** This class contains the measured characteristics of the presentation to be presented to the user in the report.

**<<DataType>> WordFrequency:** This class defines an associative array which holds the frequency of the words as an integer, and the word that the frequency is related to as String.

### 2.5.4. Dynamic Models

#### 2.5.4.1. Sequence Diagram of a User Initiating Live Recording Module

Scenario:

The user starts recording on the Live Recording Screen. Then, the Presentation Assistant starts to get the data from the user. The Face Recognition Model collects data, then returns it to the Presentation Assistant. Also, the Speech to Text Model transcribes the voice of the user which it also returns to the Presentation Assistant. Lastly, the Word Recommendation Model generates words which are also returned to the Presentation Assistant. Then the user ends the live recording. Then, the presentation analyzer starts to get data from the models. Similarly to the Presentation Assistant's case, it also gets data from the machine learning models. The sequence ends when the analysis process is finished.

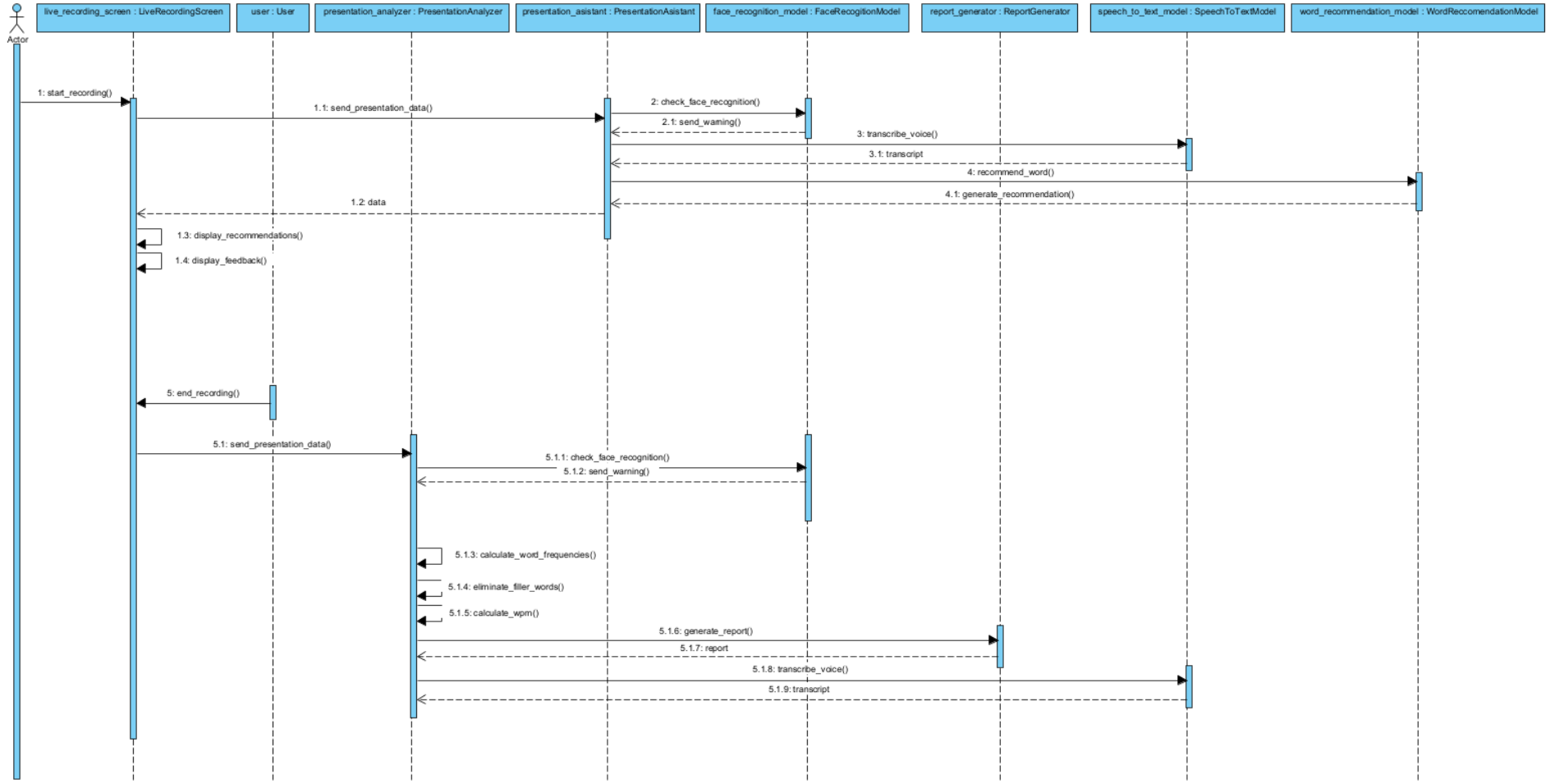


**Figure 3.** Sequence Diagram showing the sequence of events for a user initiating a live recording module

#### 2.5.4.2. Sequence Diagram of a User Initiating Upload Recording Module

Scenario:

The user uploads the recording of the presentation to the Upload Presentation Screen. Then, the Presentation Analyzer starts to get the data from the user. The Face Recognition Model collects data then returns it to the Presentation Analyzer. Also, the Speech to Text Model transcribes the voice of the user which it also returns to the Presentation Analyzer. The sequence ends when the processing of the file is done.

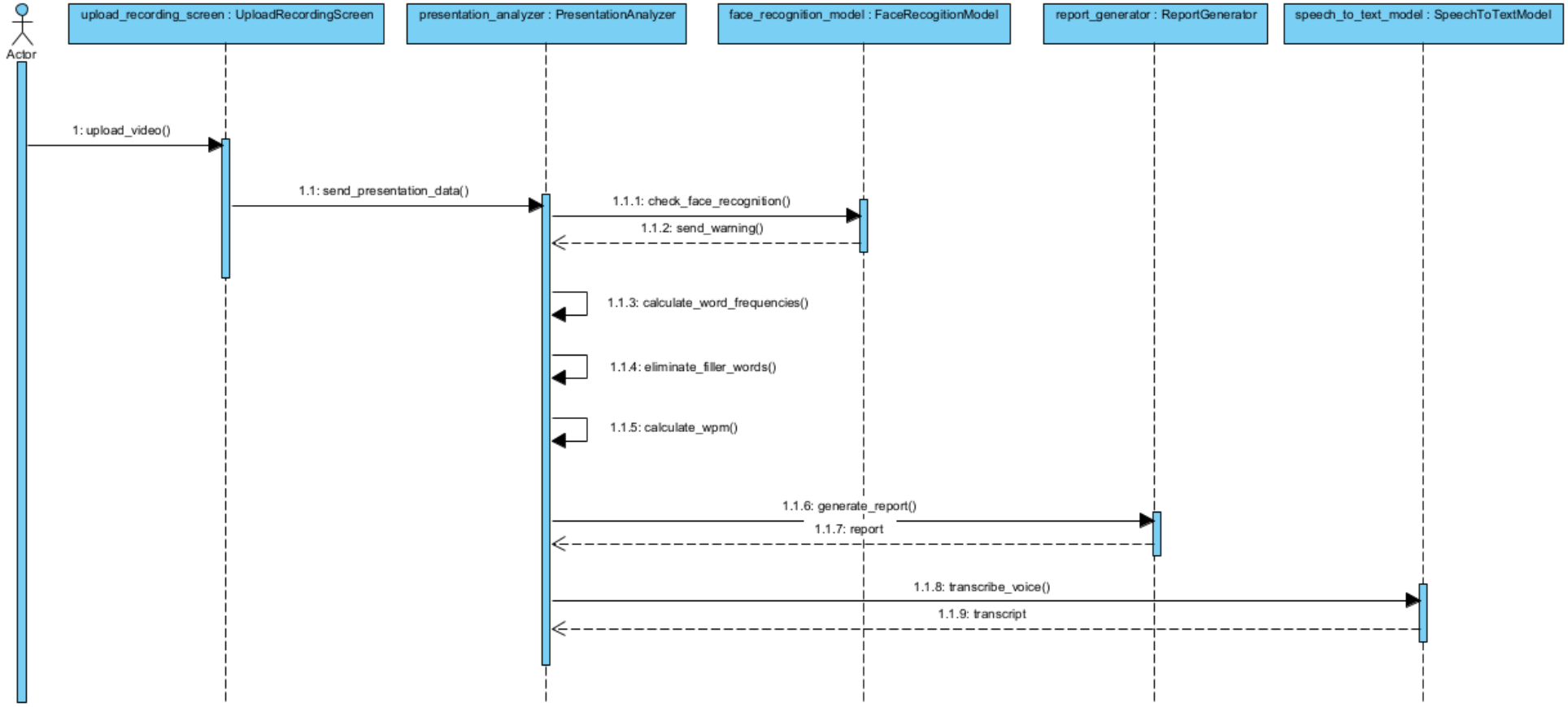


**Figure 4.** Sequence Diagram showing the sequence of events for a user initiating the upload recording module.

#### 2.5.4.3. Sequence Diagram of a User Viewing a Report

Scenario:

The user opens the presentation list. When the user selects a presentation to view its report, the report is fetched through user account -> user data manager -> database. From the presentation's report screen the user also initiates the download report function.

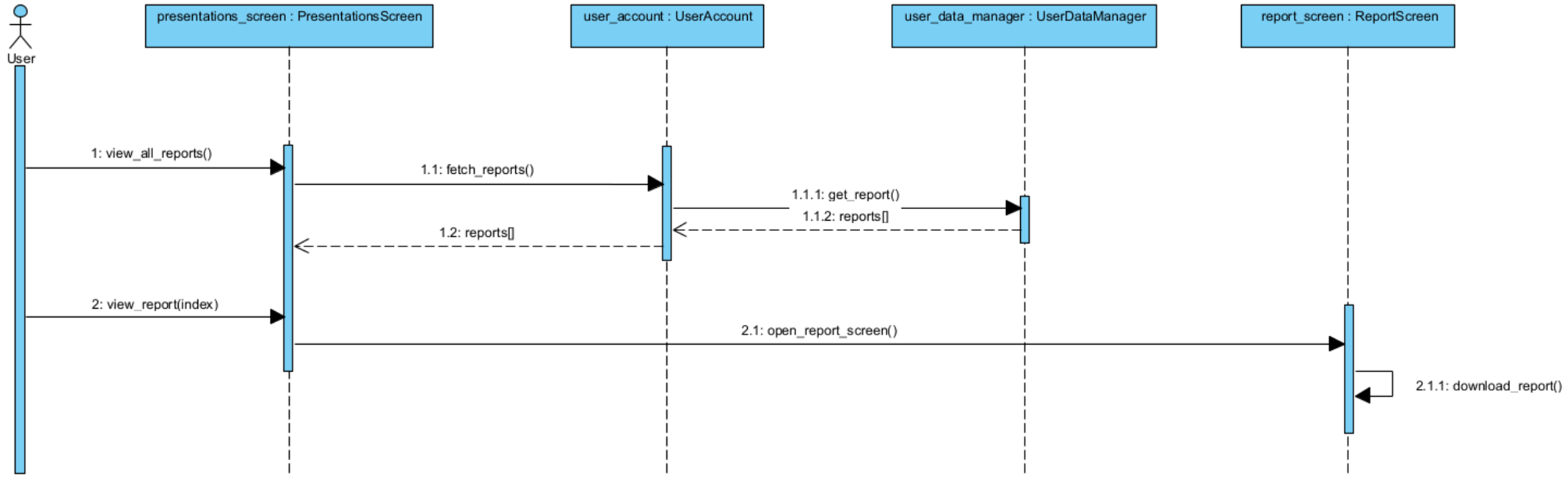


**Figure 5.** Sequence Diagram showing the sequence of events for a user viewing a report.

#### 2.5.4.4. Activity Diagram

The following activity diagram illustrates the transition between the screens and operations. When the user starts the application, the main menu screen is displayed to the user. Then, the system waits for the user's decision. If the user starts a live recording, the live recording screen is displayed. When the recording starts, the live feedback models are initiated and they exchange data continuously with the live recording screen. If the user chooses to upload a recording of a presentation. The presentation is analyzed and a report is generated. The report is then displayed to the user, and the user has the option to download the report. If the user chooses to view the tutorials of Prexcel or presentations in general, the corresponding tutorial screen is displayed to the user. The user also can view their account settings by pressing the "User Settings" button. In this screen, they have the option to modify their account details, log out of their account or delete their account. In each screen, there is an option to go back to the main menu, and if the user presses the "Exit" button on the main menu, the application is closed.

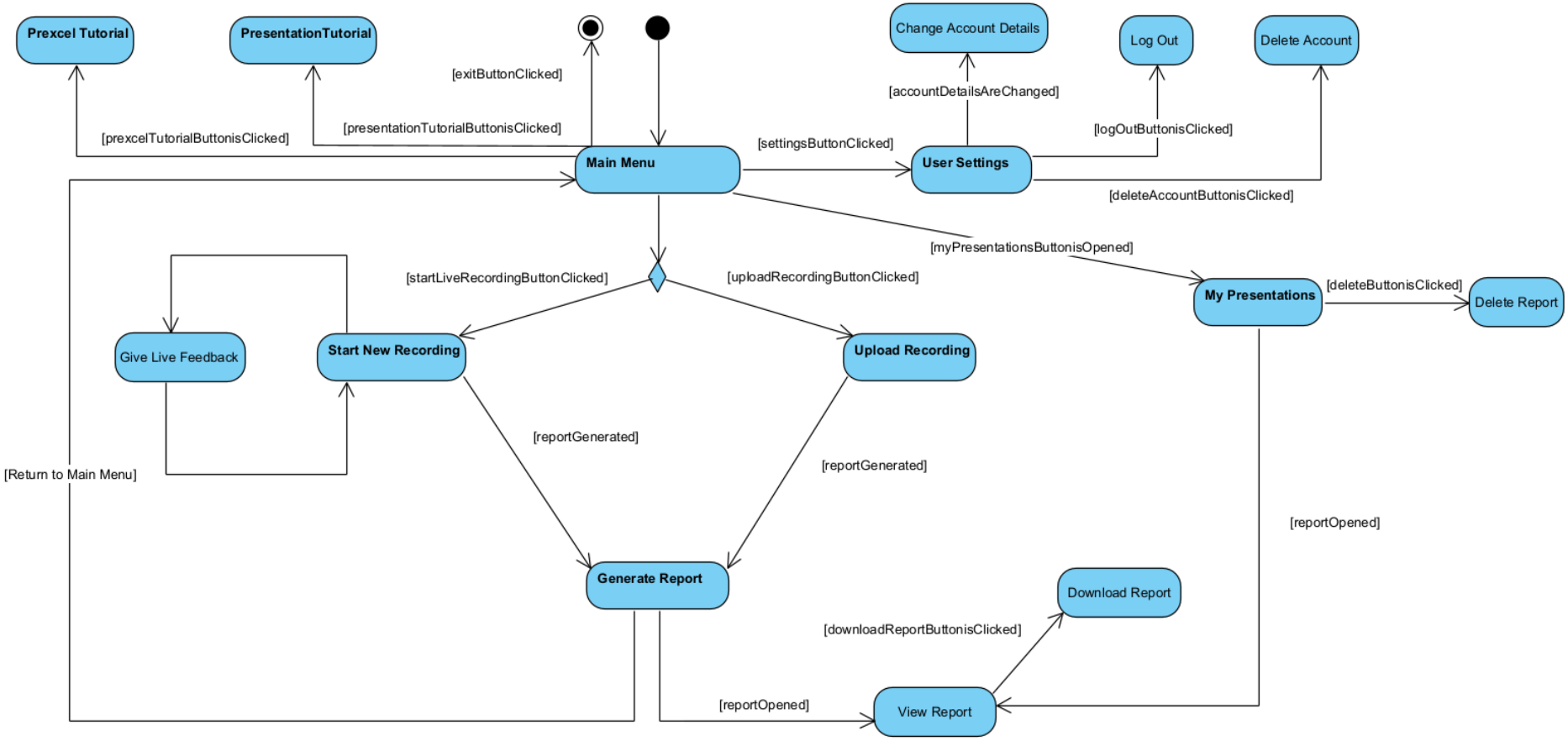


**Figure 6.** Activity Diagram of Prexcel

### 2.5.5. User Interface

#### 2.5.5.1. Navigational Path

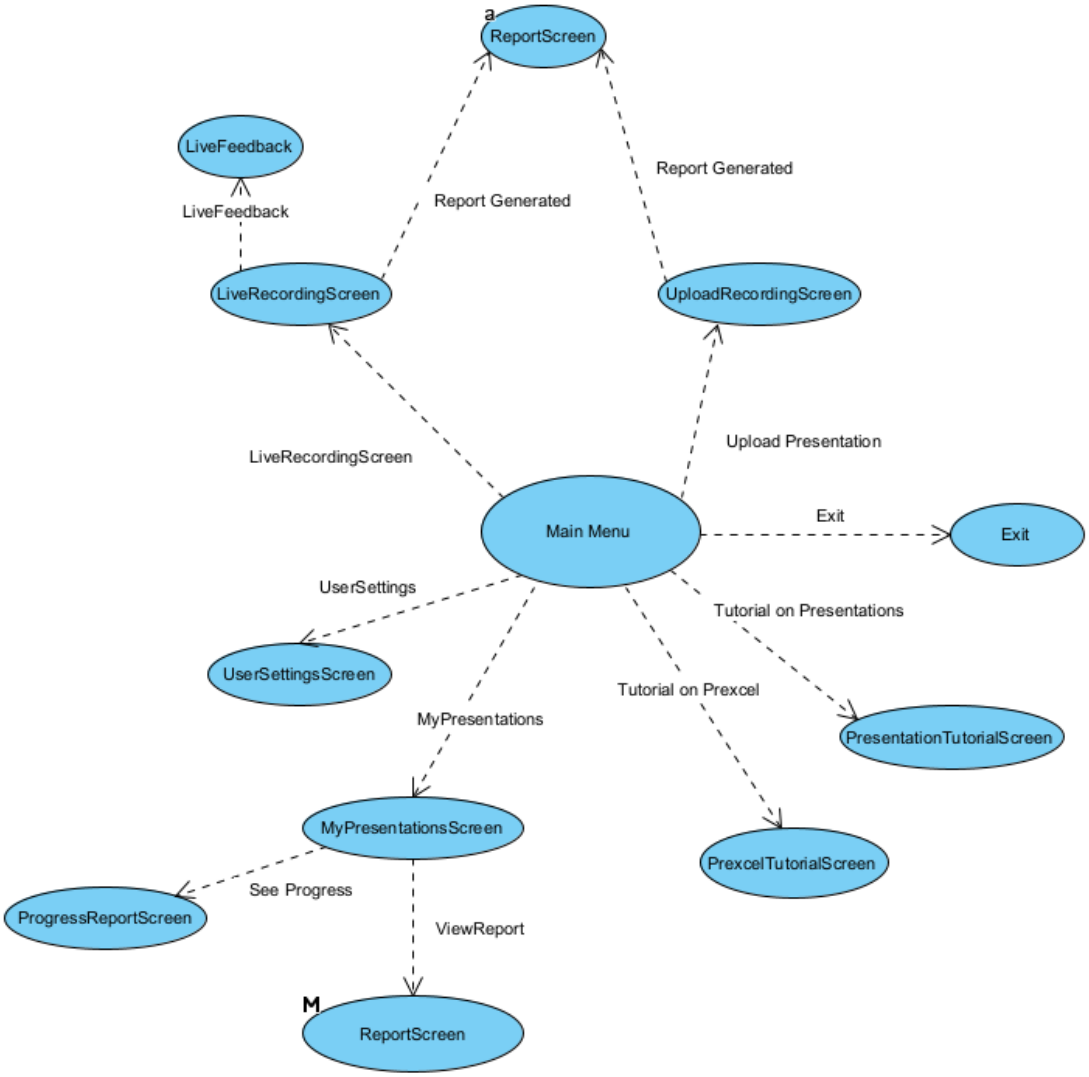


**Figure 7.** Navigational path of the user interface screen

### 2.5.5.2. UI Mock-Ups

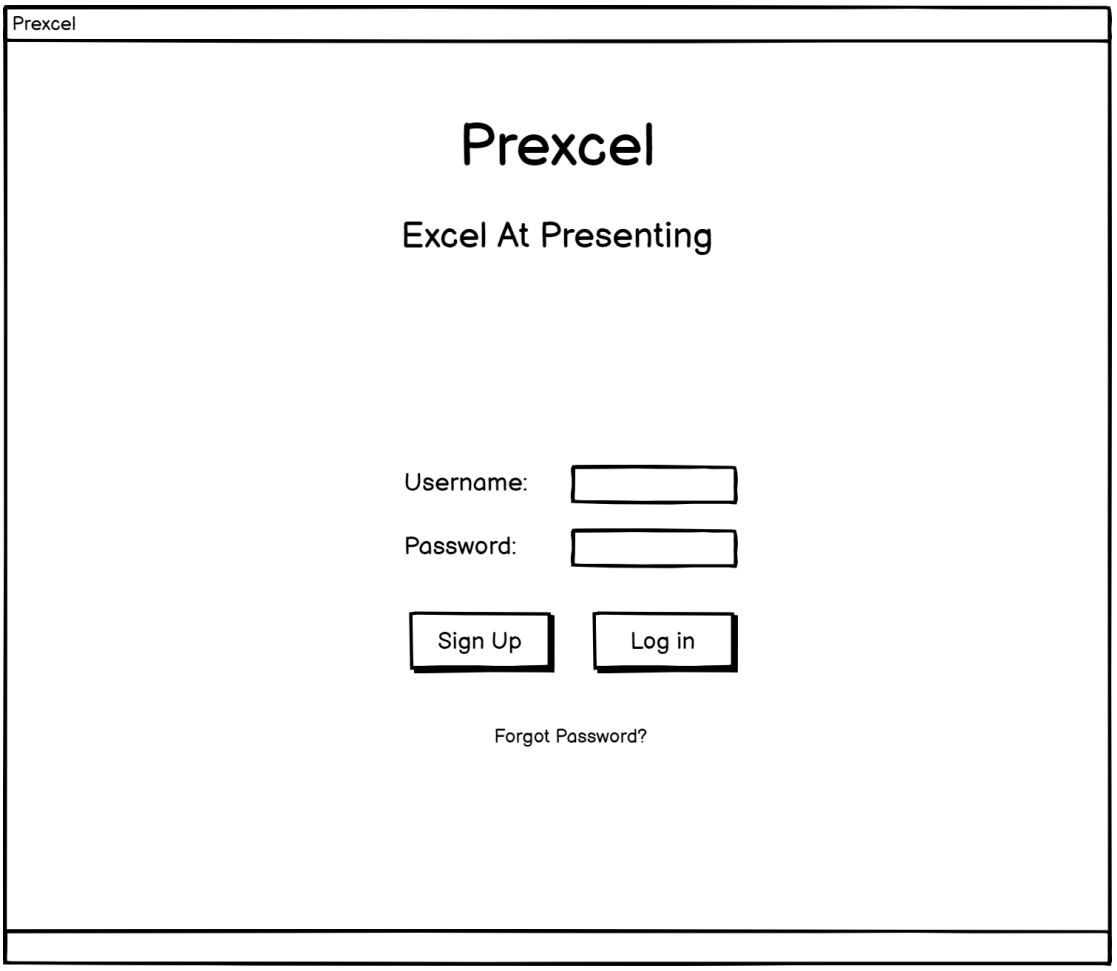


**Figure 8.** Login Screen

Login page is the first page that comes up when the application is opened. If the user does not have an account he/she can click Sign Up to go to its respective screen.

The players can request a new password for themselves if they have forgotten their password by clicking on the “Forgot Password?” button.

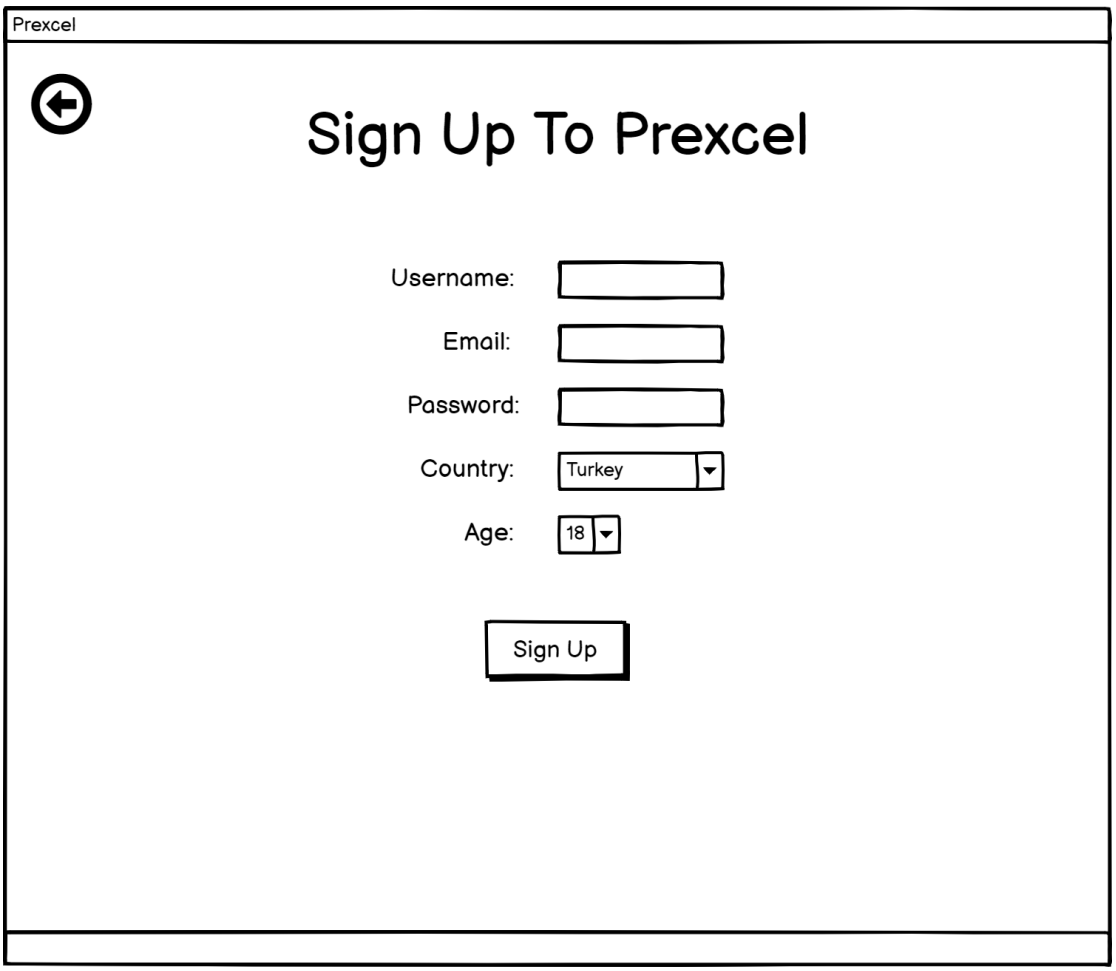


**Figure 9.** Sign Up Screen

In the Sign Up Screen the user can enter their information to respective fields, then click Sign Up to create an account.

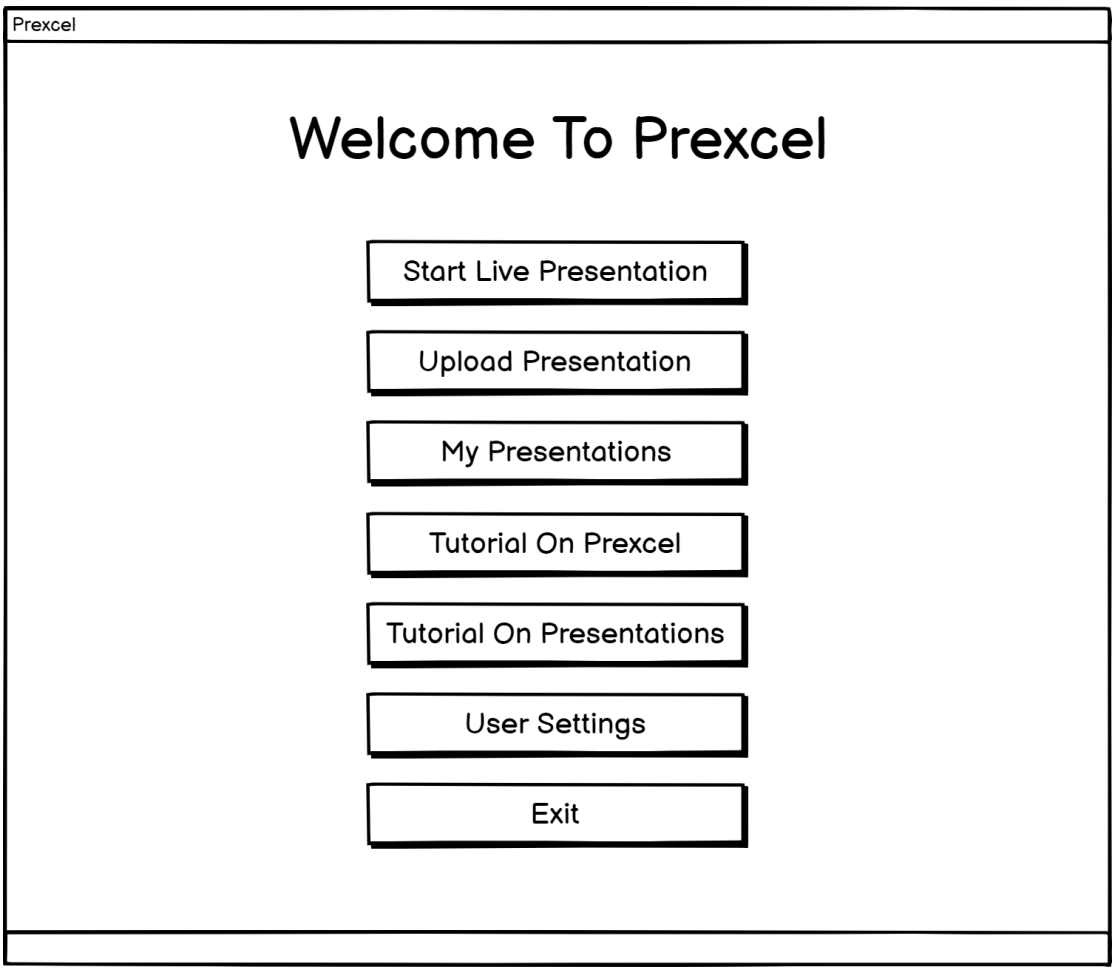


**Figure 10.** Main Menu Screen

Main Menu Screen opens after the user signs in to the application. It contains 6 buttons which lead the user to their respective pages. When the Exit button is clicked the application closes.

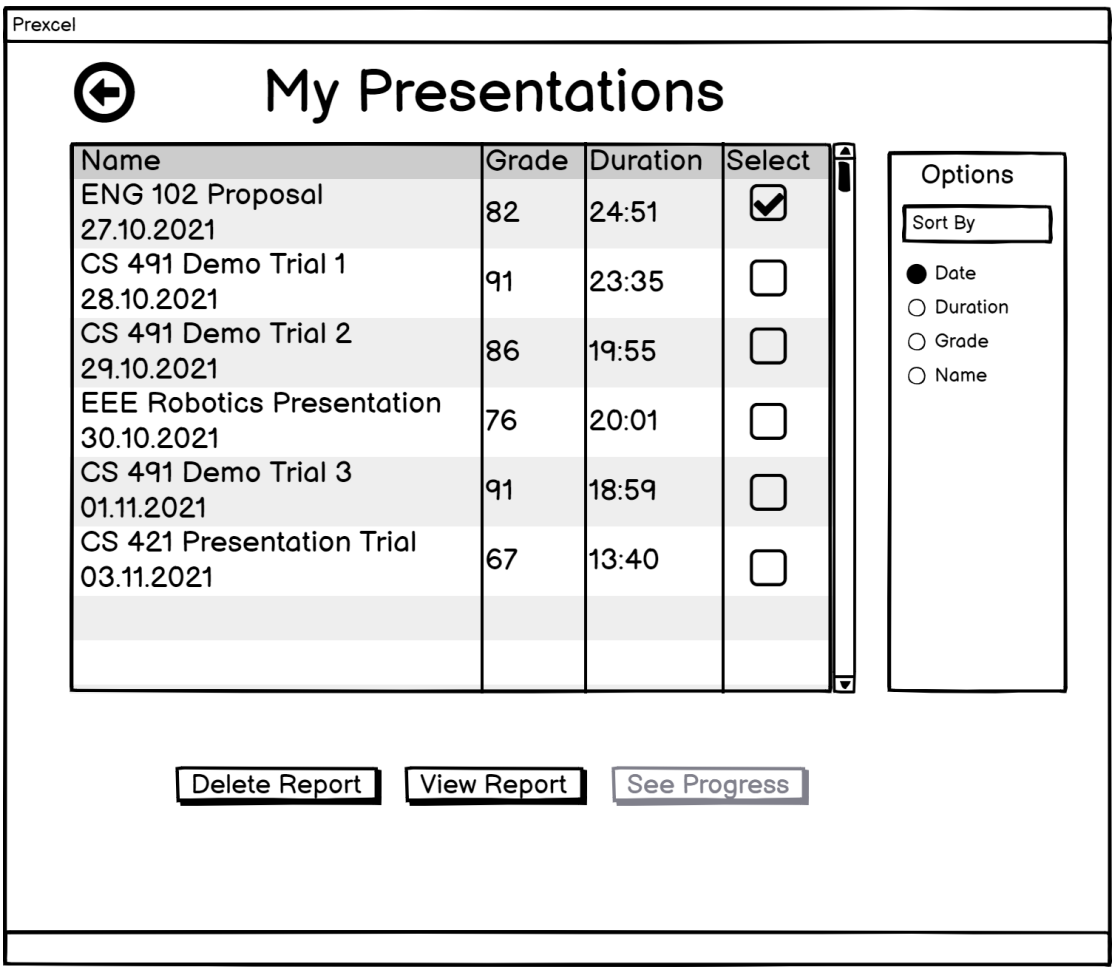


**Figure 11.** My Presentations Screen

In My Presentations Screen, the user can see their presentation history and relevant data. In the right hand side of the screen the user has the option to sort the presentations. If the user clicks on the View Report button, the report screen opens up. If more than one report is Selected on the Select column, then the See Progress button is enabled. The View Report button is enabled when there is only one presentation selected. If it is clicked its respective screen opens. Users can delete the selected presentations using the Delete Report button.

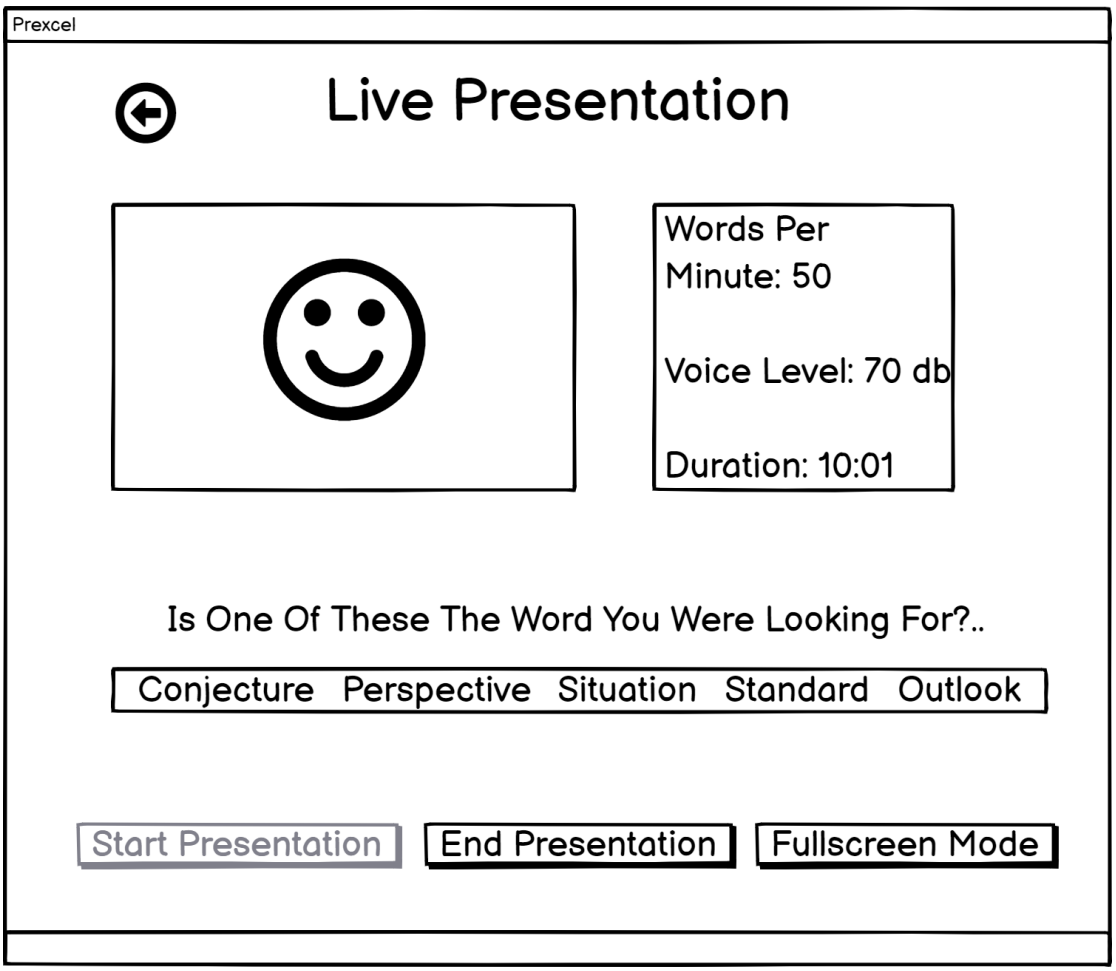


**Figure 12.** Live Presentation Screen

In the Live Presentation Screen, the user has the option to start a live presentation and end the currently live presentation. On the top left side, the user's camera feed is displayed. On the top right side, the live statistics of the presentation is displayed. And in the middle of the screen the word recommendations are displayed when the user gets stuck. Clicking the Fullscreen Mode button takes the app to the fullscreen mode, as seen in Figure 12.

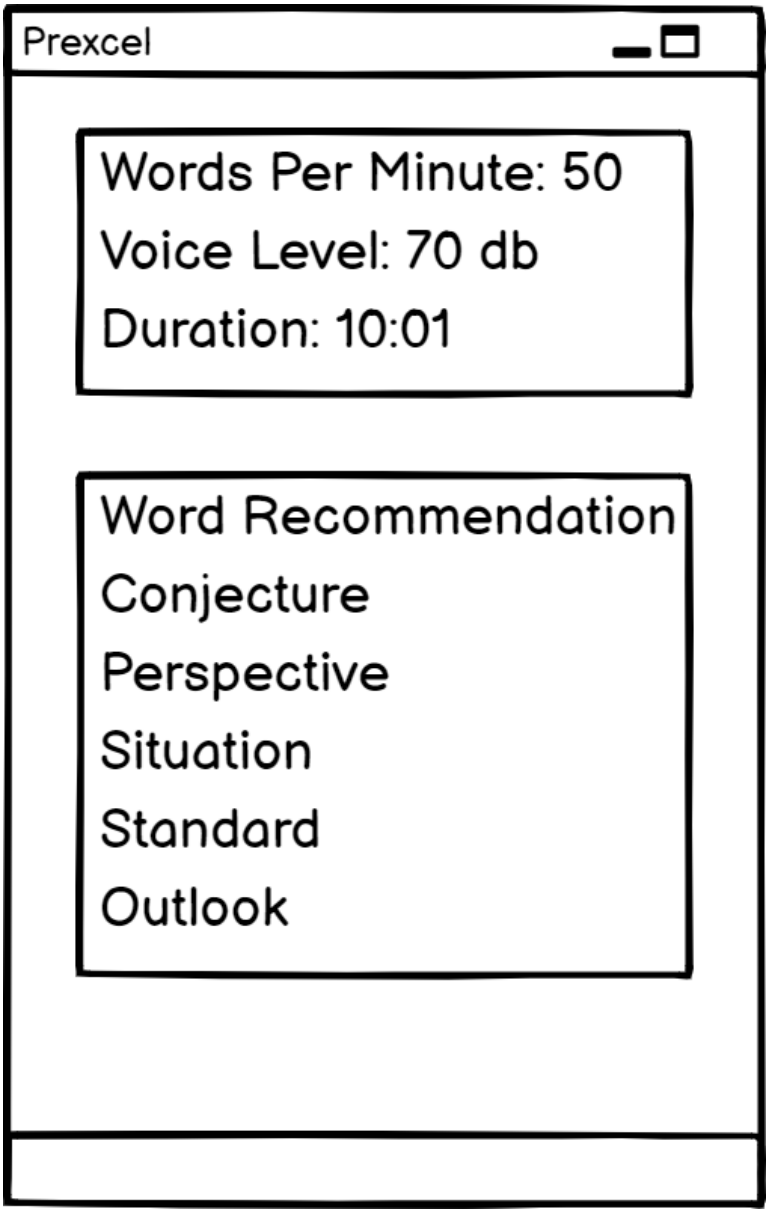


**Figure 13.** Statistics and Word Recommendation Popup Screen

This is the screen of the fullscreen mode of the live presentation feature. In this popup, users' live statistics are displayed at the top side. At the bottom side, when the user gets stuck word recommendations are displayed.

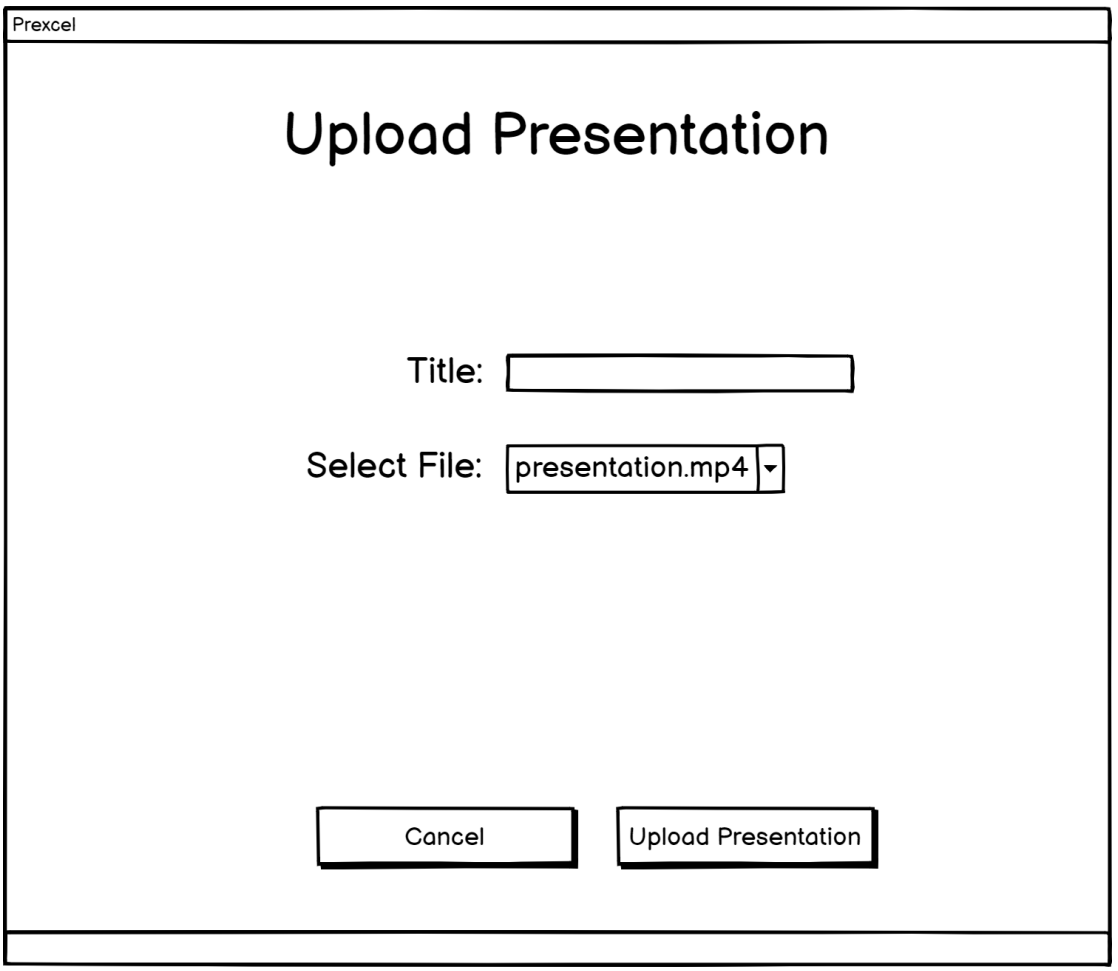


**Figure 14.** Upload Presentation Screen

In this screen the user is prompted to enter a Title for his/her presentation, then select a file. Once the user has done these he or she can upload their presentation by clicking on the Upload Presentation button. Or the user can cancel the operation by clicking the Cancel button.

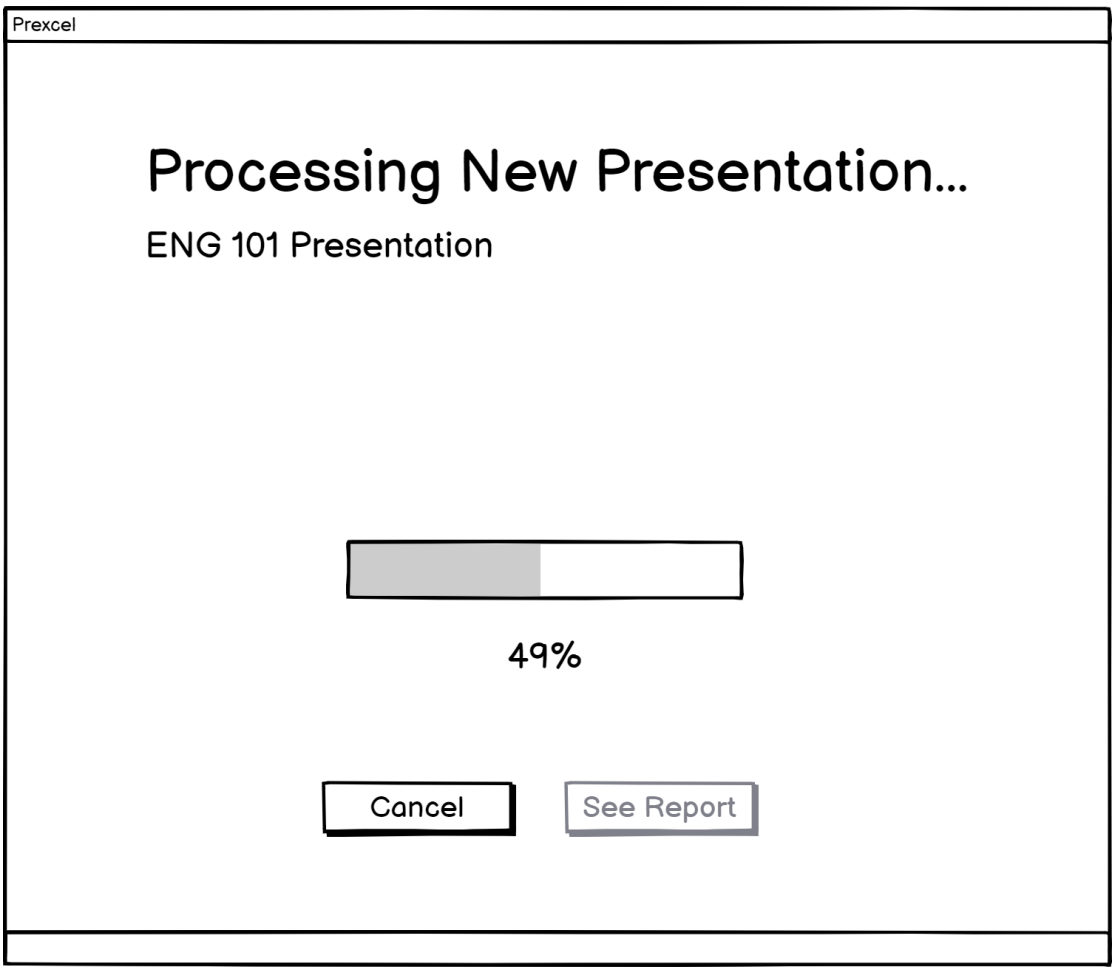


**Figure 15.** Processing Presentation Screen

This screen opens up when the user uploads a presentation. Since sometimes the file sizes may be too large, the processing may take a while, therefore the progress screen is shown for the user to be able track the processing times. If the user does not wish to continue the downloading progress he or she can cancel or when the processing is done he or she can view the report by clicking on the See Report button.

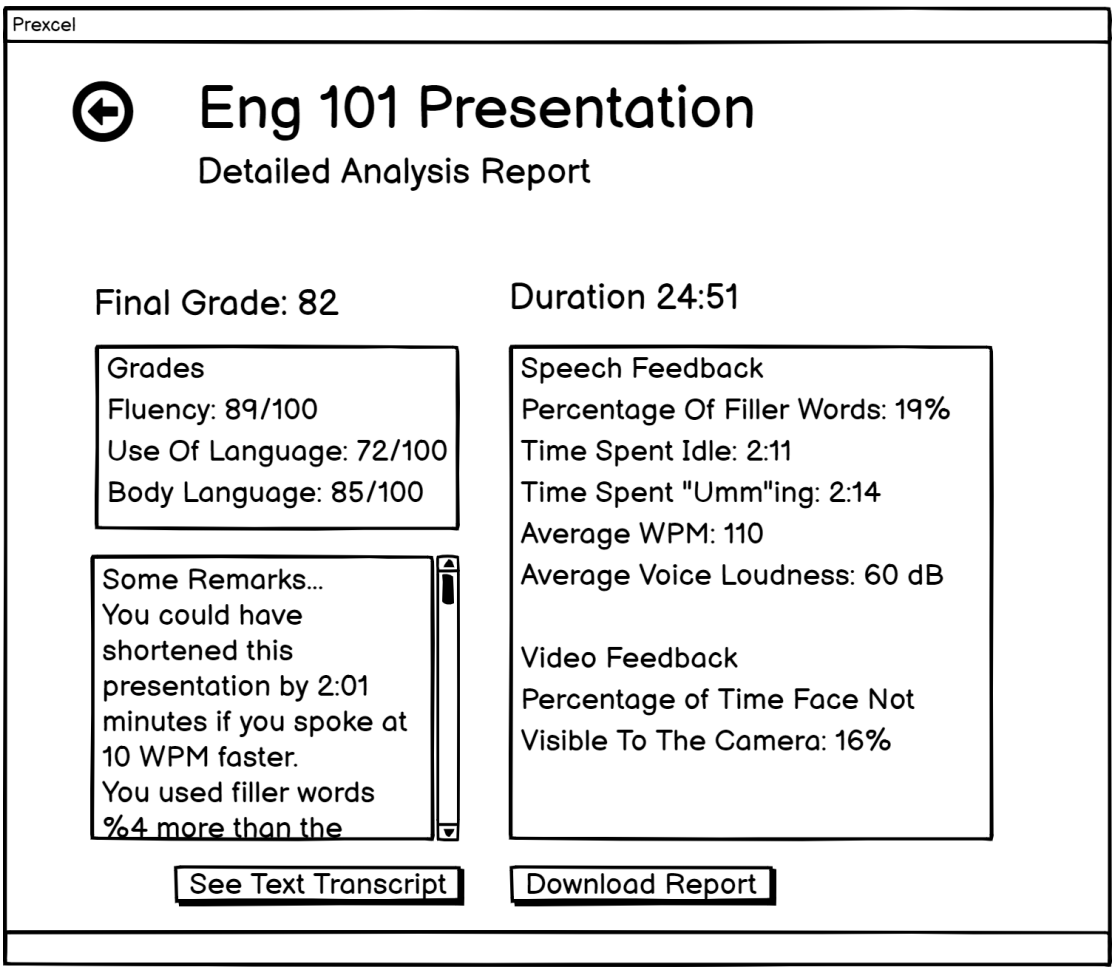


**Figure 16.** Presentation Details Screen

In this screen, the details of the user's selected presentation is displayed. On the top left corner presentations' grade information is displayed. On the bottom left side of the screen, remarks and recommendations about the presentation are displayed. On the right hand side of the screen, feedback about the two separate parts (speech and video) are displayed.

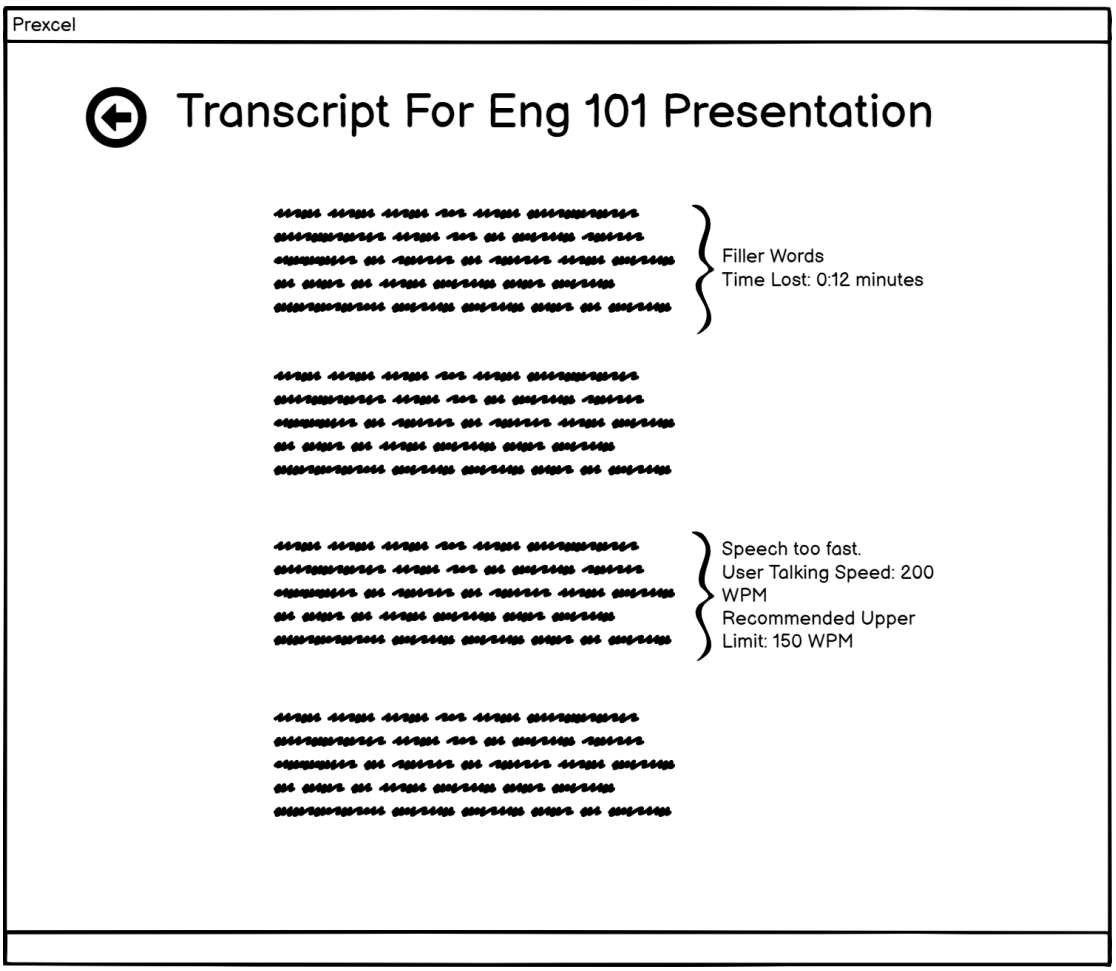


**Figure 17.** Transcript Details Screen

In the transcript details screen, users' transcribed speech is displayed. Within the transcript, filler words are highlighted and their details are indicated. When the WPM is too low or too high it is also highlighted and its details are also displayed.

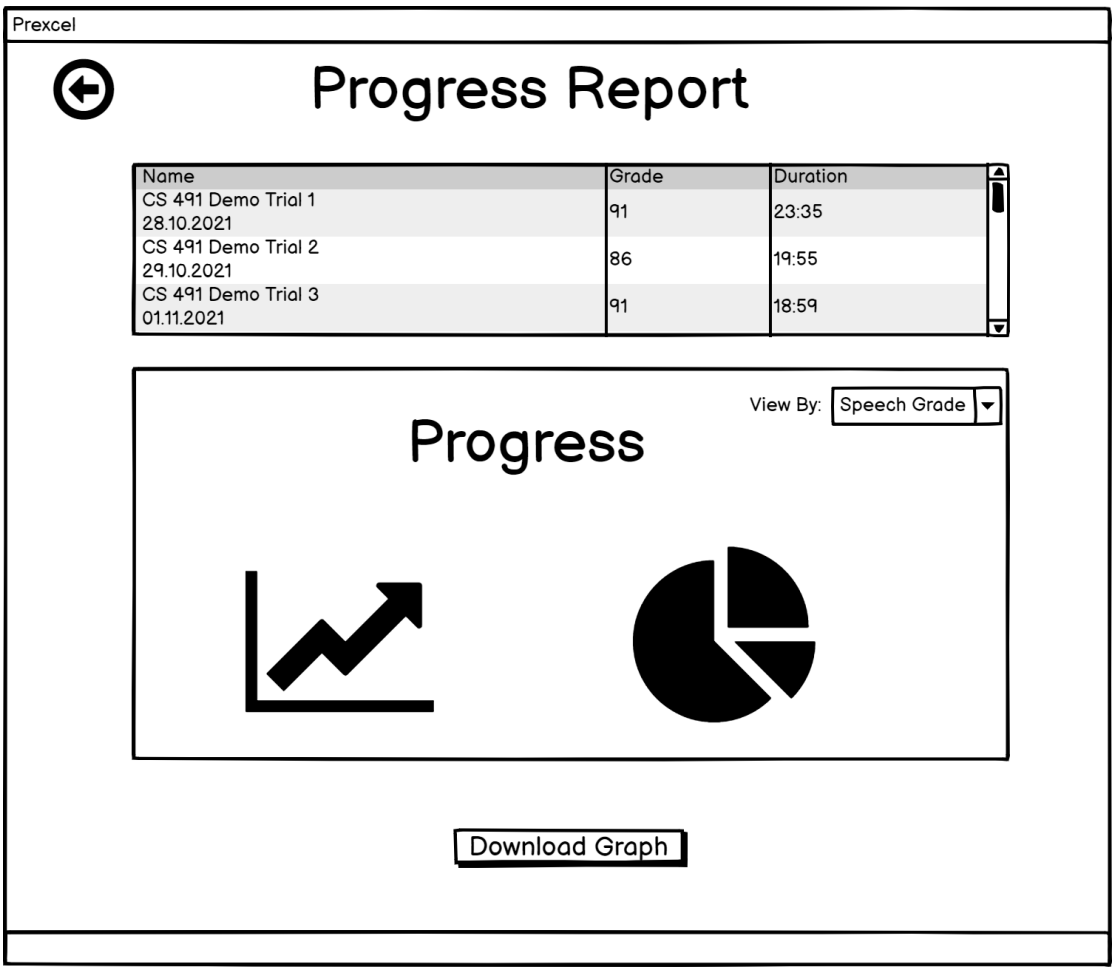


**Figure 18.** Progress Tracking Screen

In this screen, the users can track their progress based on the grades of their past presentations. The past presentation grades are illustrated by a graph. The presentations displayed in this screen are the presentations selected in figure 16.

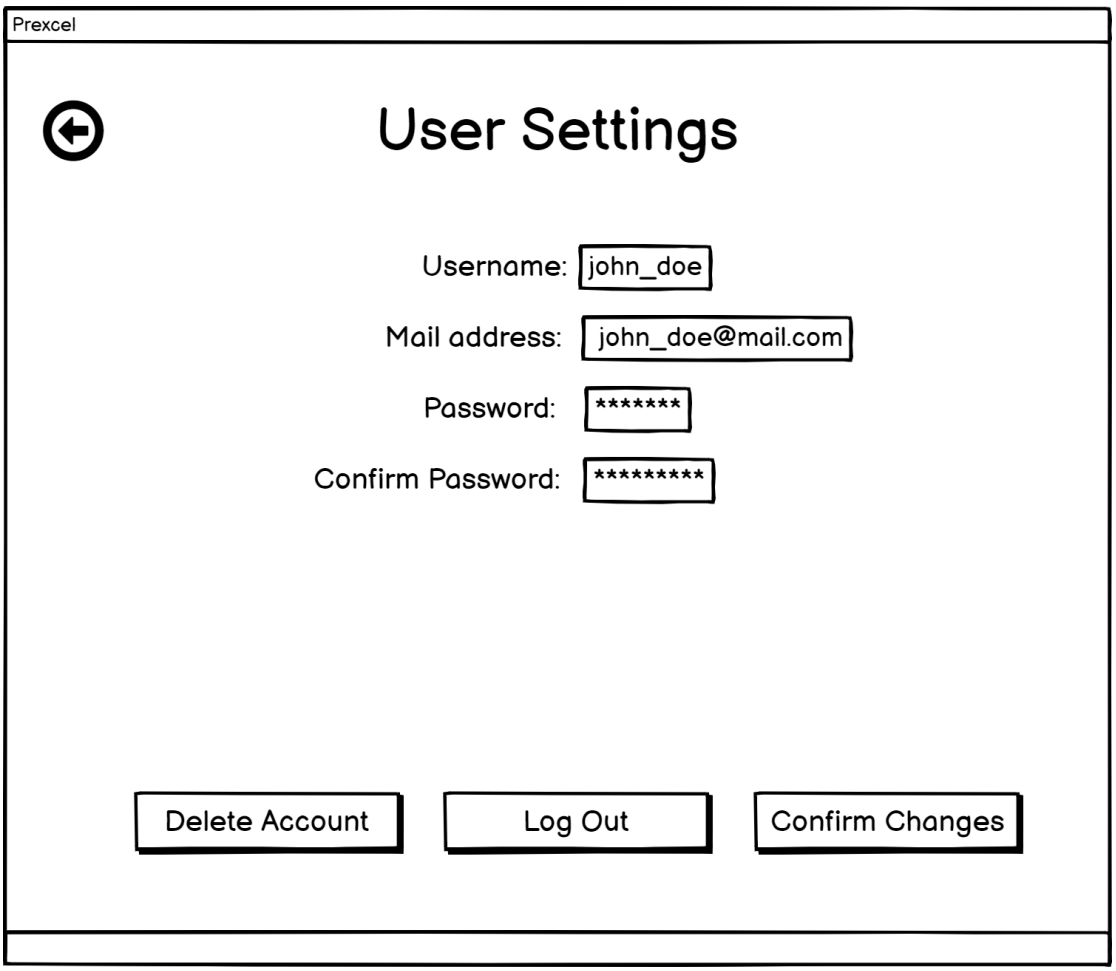


**Figure 19.** User Settings Screen

In the user settings screen, the users can see and modify their information. They can also log out of their account and delete their account permanently.

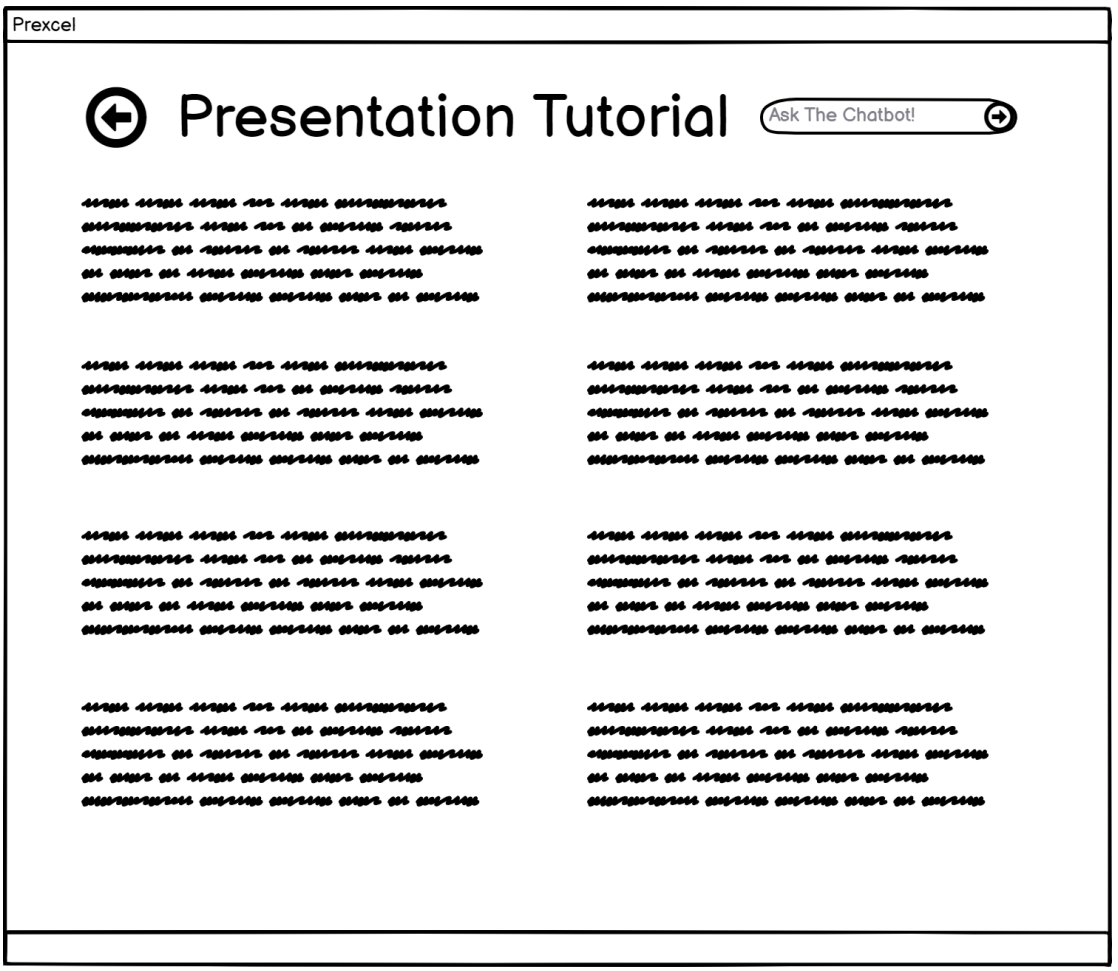


**Figure 20.** Presentation Tutorial

In the Presentation Tutorial screen, the users can learn about the metrics used by Prexcel to evaluate the presentations and more tips about how to give an effective presentation.

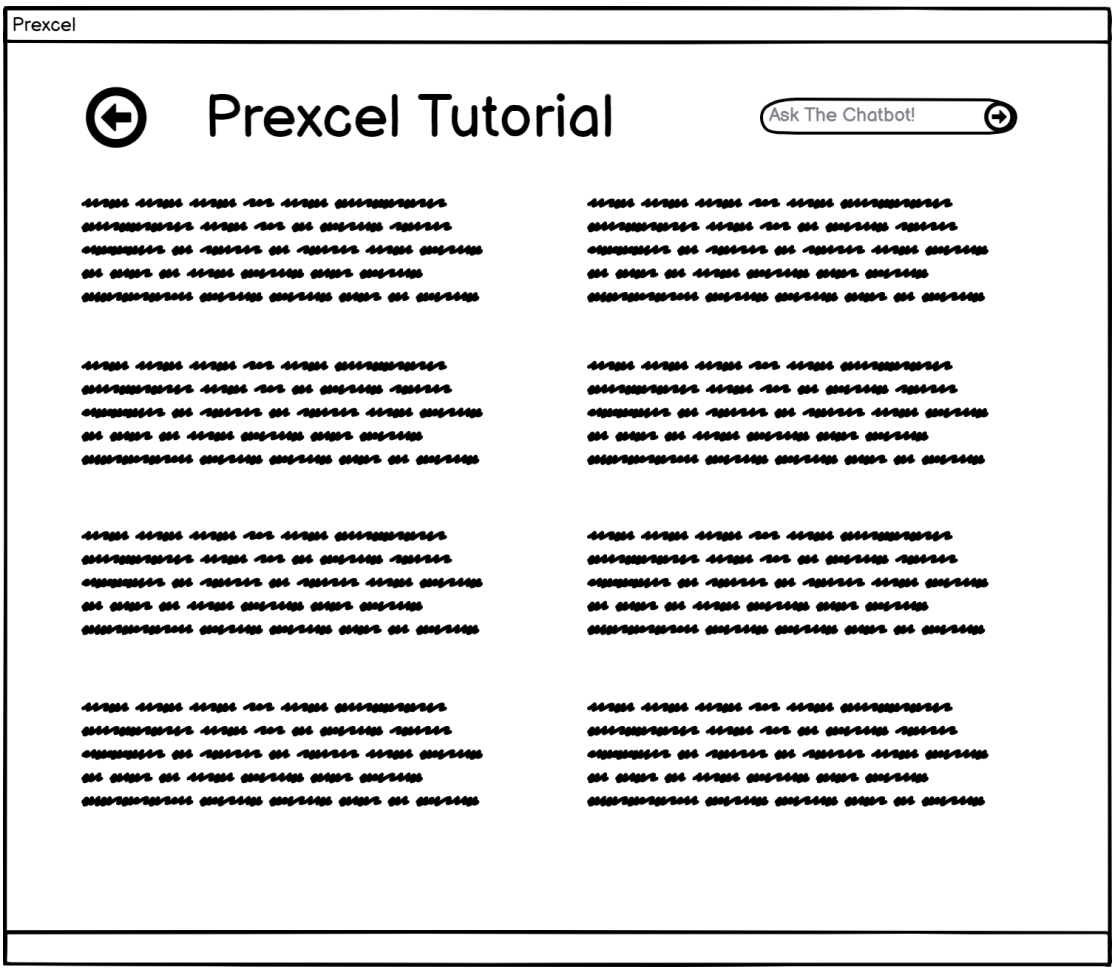


**Figure 21.** Prexcel Tutorial Screen

In the Prexcel Tutorial screen, information on the app's usage and features are provided to the users.

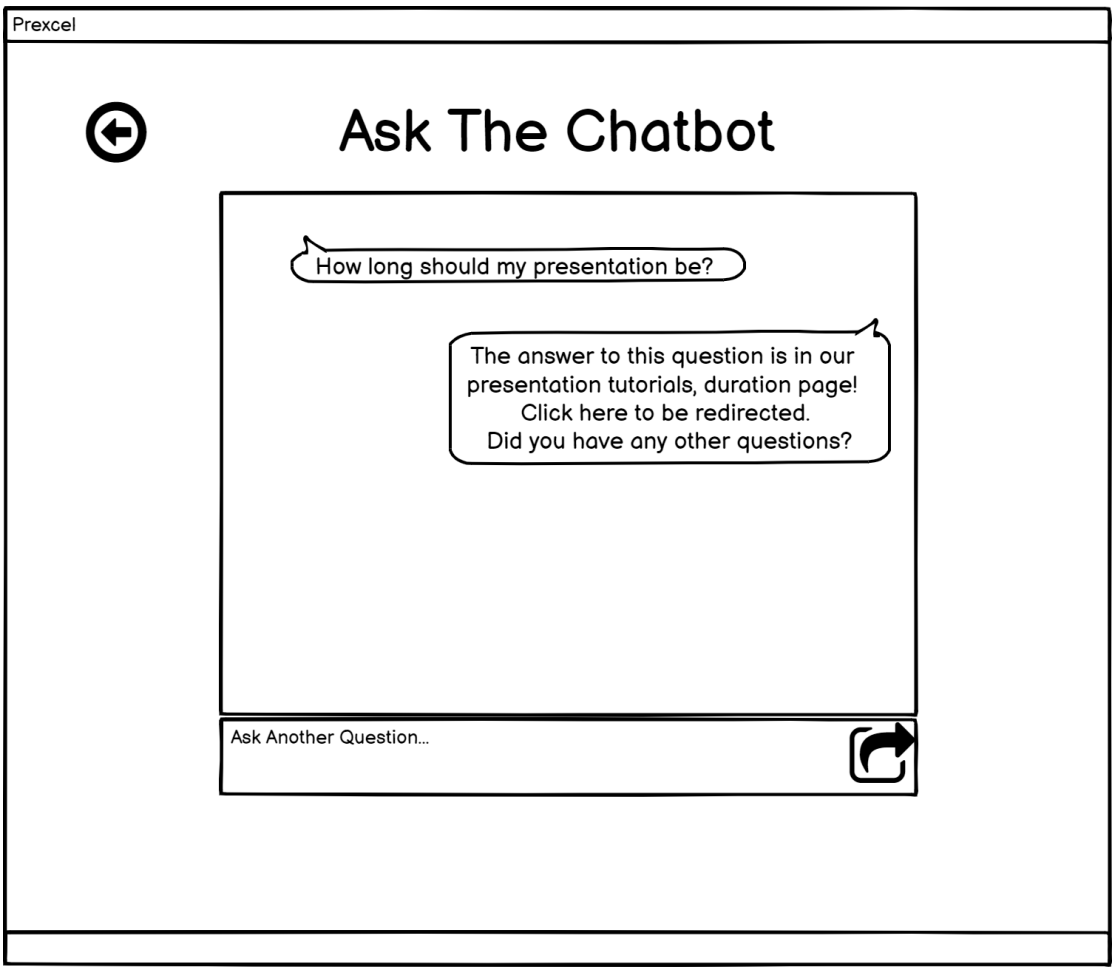


**Figure 22.** Chatbot screen

In the chatbot screen, the users can ask questions about how their presentations are evaluated and also ask simple questions on how to improve their presentations.

# 3. Other Analysis Elements

## 3.1. Consideration of Various Factors in Engineering Design

Our project affects and is affected by various factors like public health factors, cultural factors, global factors and social factors.

With regards to public health, Prexcel has had a positive impact, especially, during the pandemic. Prexcel is a tool that is optimized and geared towards online presentations, and in particular, it improves the users' online presentation abilities and enhances their online presentations with the live feedback features. The increased effectiveness of online presentations may decrease the need for face to face presentations, and help improve public health during the pandemic.

With regards to cultural factors, again, Prexcel has significant positive effects. Software development companies have a unique culture where working remotely is more accepted and significantly more prevalent. Prexcel has the potential to significantly improve remote working culture by improving remote presentations, and making remote work more desirable.

With regards to global factors, education is one of the most important global efforts. Prexcel is a great complementary educational tool that could help students improve their presentation skills. Furthermore, the word recommendation feature of the live feedback feature of Prexcel could be very beneficial for students whose native language is not English, during their presentations, if they are receiving an education in English.

With regards to social factors, establishing an effective communication and expressing one's ideas is extremely crucial. With this in mind, Prexcel enhances the user's ability to express themselves in a better way by giving them directions and statistics on how they can improve. Therefore, while giving online presentations the presenters are also going to develop themselves in their social relationships.

With regards to economic factors, effective communication and presentation skills are a very important skill in life that many people struggle with, and in general many people spend a lot of money into various effective speech & presentation courses and resources. Prexcel is a tool that will help people improve those skills for free, saving many people a considerable amount of money. In addition to this, the above mentioned effects on remote work also have significant positive economic impact.

With regards to environmental factors, as also previously mentioned, Prexcel makes remote work more feasible. Making remote work more feasible would, in time, stop migration to big cities & allow people working in smaller towns & villages, and help stop the excess crowding in cities. This would mean less time spent commuting, consequently less pollution and a net positive impact for the environment.

The project does not have a significance with regards to public safety and public welfare.

Table xx: Factors that can affect analysis and design.

| | Effect level | Effect |
|---|---|---|
| Public health | 5/10 | Prexcel helps the users improve their online presentation skills, as well as enhancing their online presentations by live feedback features. The increased effectiveness of online presentations may decrease the need for face to face |

| | | |
|---|---|---|
| | | presentations, and help improve public health during the pandemic. |
| Public safety | 0/10 | None. |
| Public welfare | 0/10 | None. |
| Global factors | 7/10 | Education is one of the most important global efforts. Prexcel could greatly help students improve their presentation skills as a complementary tool. |
| Cultural factors | 4/10 | Usually, software development companies have a unique culture where working remotely is more prevalent and accepted. Prexcel helps people develop better online presentation skills, making remote working more feasible and desirable, making a cultural impact in this aspect. |
| Social factors | 7/10 | Prexcel allows the users to explain their ideas on a deeper level, thus elevating their social skills. |
| Economical factors | 9/10 | Prexcel would help people develop their presentation skills without attending paid courses, helping many people save money. Furthermore, promoting remote work would also have a positive impact in this regard. |
| Environmental factors | 5/10 | Prexcel would have a positive impact on the environment, as it makes remote work more feasible, and makes people spend less time commuting. (Hence less pollution.) |

## 3.2. Risks and Alternatives

One of the most important objectives of Prexcel is to give accurate recommendations to the users when they get stuck in a presentation. To do this, several machine learning algorithms will be used to create a recommending system

with useful outputs. However, achieving this requires a detailed analysis of the context and flow of the sentences. One risk is that the accuracy of the recommendation model is not sufficient to be useful for the user during the presentation. In that case, the recommendations of the system would be not only useless, but also confusing for the presenter during the presentation. If the word recommendation model does not produce accurate results by the end of the implementation, we plan to change the implementation such that the recommendations will be generated based on a transcript of the presentation uploaded by the presenter before the presentation. In this way, the model can make more relevant predictions

Since the word recommendation model utilizes machine learning algorithms, the running time of those algorithms may cause delays in word recommendations. If the delay exceeds a certain amount of time such as 3 seconds, these recommendations may not be as useful and this may affect the presentation quality negatively. If the model has an unacceptable amount of delay (for example, more than 3 seconds), the word recommendation algorithm may be simplified so that a portion of recommendation quality will be traded for a reduction of running time.

Another possibility of risk is that the speech-to-text model (DeepSpeech) won't be able to transcribe the voice accurately enough to be analyzed conveniently. If we encounter such problems based on the quality of the speech-to-text tool, we may look for alternative tools in order to increase the transcription quality.

Table xx: Risks

| | Likelihood | Effect on the project | B Plan Summary |
|---|---|---|---|
| Irrelevant word recommendations | 4/10 | The program makes useless and confusing recommendations | Word recommendation based on a transcript uploaded by the presenter |
| Excessive word recommendation duration | 6/10 | The recommendations are delayed and does not help the presenter | Simplified word recommendation algorithm |
| Poor performance by the speech-to-text tool | 3/10 | The program cannot transcribe the speech correctly | Usage of alternative tools in order to increase transcription quality |

## 3.3. Project Plan

Our project plan is explained in terms of work packages, which are shown below. These packages represent the bulk of the work required for the project to function. The packages are distributed in a manner that is most convenient to us, where the package leaders and members are the ones most able to contribute meaningfully to the work required, due to their previous foundations and interests. The Gantt chart presented in the next section shows the distribution of the work packages discussed below with respect to time.

These are our project goals:

**Project Goals:**

1) Developing a responsive user-interface that can communicate and properly display the data received from the backend processes.
2) Implementing a system that takes user's presentation videos or does a live recording, and processes and analyses that recording according to the metrics.
3) Implementing a detailed end analysis that gives feedback to users based on the previously discussed metrics.
    a) Developing a system that converts speech to text and analyzes the text according to the previously discussed metrics.
    b) The program should be able to determine when the user's face is not recognized by the camera, and give feedback to the user based on that.
4) Developing a live feedback system that recommends supplementary words that will assist the users when they are presenting their presentations live.
5) Developing a database system that stores the user information such as statistics of the users previous presentations.

Below are our work packages, and our explanations for each work package:

**Work Packages:**

Table 1: List of work packages

| WP# | Work package title | Leader | Members involved |
|---|---|---|---|
| WP1 | High-Level Design Report | Can Kırımca | All Members |
| WP2 | Low-Level Design Report | Burak Yiğit Uslu | All Members |
| WP3 | Final Report | Can Kırşallıoba | All Members |
| WP4 | User-Interface | Burak Yiğit Uslu | Burak Yiğit Uslu, Can Kırımca, Can Kırşallıoba, Barış Tiftik |

| | | | |
|---|---|---|---|
| WP5 | Video Uploading. Recording Modules | Alper Sarı | Alper Sarı, Barış Tiftik, Burak Yiğit Uslu |
| WP6 | Presentation Analyzer | Can Kırşallıoba | Can Kırşallıoba, Barış Tiftik, Alper Sarı, Burak Yiğit Uslu |
| WP7 | Speech to Text Model | Can Kırşallıoba | Can Kırşallıoba, Burak Yiğit Uslu, Can Kırımca, Alper Sarı |
| WP8 | Facial Recognition Model | Barış Tiftik | Barış Tiftik, Can Kırımca, Can Kırşallıoba |
| WP9 | Presentation Assistant | Alper Sarı | Alper Sarı, Barış Tiftik, Can Kırımca |
| WP10 | Word Recommendation Model | Can Kırımca | Can Kırımca, Can Kırşallıoba |
| WP11 | Users & Presentations Database | Can Kırımca | Can Kırımca, Burak Yiğit Uslu |

Table 2: Work Package 1 describing High Level Design Report

| **WP 1:** High-Level Design Report | | | |
|---|---|---|---|
| **Start date:** *20.11.2021* **End date:** *23.12.2021* | | | |
| **Leader:** | *Can Kırımca* | **Members involved:** | All Members |
| **Objectives:** *To write a report that describes the system design and architecture.* | | | |
| **Tasks (Sections):**<br>***Task 1.1 Introduction***<br>***Task 1.2 Current Software Architecture***<br>***Task 1.3 Proposed Software Architecture***<br>***Task 1.4 Subsystem Services***<br>***Task 1.5 Consideration of Various Factors in Engineering Design***<br>***Task 1.6 Teamwork Details***<br>***Task 1.7 Glossary***<br>***Task 1.8 References*** | | | |
| **Deliverables**<br>***D1.1:*** *High-Level Design Report* | | | |

Table 3: Work Package 2 describing Low Level Design Report

| **WP 2:** Low-Level Design Report | | | |
|---|---|---|---|
| **Start date:** *01.02.2021* **End date:** *TBD* | | | |
| **Leader:** | *Burak Yiğit Uslu* | **Members involved:** | All Members |
| **Objectives:** *To write the design report that identifies the low-level design of project components such as classes, packages and interfaces.* | | | |

| |
|---|
| **Tasks (Sections):**<br>***Task 2.1 Introduction***<br>***Task 2.2 Packages***<br>***Task 2.3 Class Interfaces***<br>***Task 2.4 Glossary***<br>***Task 2.5 References*** |
| **Deliverables**<br>***D2.1:*** *Low-Level Design Report* |

Table 4: Work Package 3 describing Final Report

| | | | |
|---|---|---|---|
| **WP 3:** Final Report | | | |
| **Start date:** *TBD* **End date:** *TBD* | | | |
| **Leader:** | *Can Kırşallıoba* | **Members involved:** | All Members |
| **Objectives:** *To write the final report that evaluates the final status of the project and describes the future work related to the project.* | | | |
| **Tasks (Sections):**<br>***Task 3.1 Introduction***<br>***Task 3.2 Requirements Details***<br>***Task 3.3 Final Architecture and Design Details***<br>***Task 3.4 Development/Implementation Details***<br>***Task 3.5 Testing Details***<br>***Task 3.6 Maintenance Plan and Details***<br>***Task 3.7 Other Project Elements***<br>***Task 3.8 Conclusion and Future Work***<br>***Task 3.9 Glossary***<br>***Task 3.10 References*** | | | |
| **Deliverables**<br>***D3.1:*** *Final Report* | | | |

Table 5: Work Package 4 describing User-Interface

| | | | |
|---|---|---|---|
| **WP 4:** *User-Interface* | | | |
| **Start date:** *15.11.2021* **End date:** *01.05.2022* | | | |
| **Leader:** | *Burak Yiğit Uslu* | **Members involved:** | Burak Yiğit Uslu, Can Kırımca, Can Kırşallıoba, Barış Tiftik |
| **Objectives:** *UI is the interface through which the user interacts with and uses the program. Within this work package, we aim to develop and provide the user with pages to log in to and create their account, a page to see their previous presentation and grades, pages to perform live presentations or upload* | | | |

| |
|---|
| *presentations, and a page to see the related feedback and other minor pages. To ensure the smoothest user experience possible, the UI must be responsive, intuitive and functional.* |
| **Tasks:**<br>***Task 4.1 Development of the log-in/sign-up screens*:** *These screens should provide the functionality for the user to be able to sign up and log in., and communicate with the database for these functionalities.*<br><br>***Task 4.2 Development of live presentation and the live feedback screen*:** *These are the screens that will give momentary feedback to the user during live presentations. They should be able to display the video input on the screen, the volume of the audio inputs on the screen as well, and communicate with relevant backend processes for word recommendation.*<br><br>***Task 4.3 Development of the uploading presentation video screen*:** *This is the screen the user uploads pre-recorded presentation videos. This screen should have a file selector and a responsive progress bar that changes according to the remaining processing time.*<br><br>***Task 4.4 Development of the detailed analysis & feedback screen and the transcript screen*:** *These are the screens that show the feedback to the user about their presentations. These screens should fetch the relevant analysis data from the backend processes and display them to the user in the correct format.*<br><br>***Task 4.5 Development of the user settings screen* :** *From this screen user should be able to change his email, username, and password from this screen. Users should also be able to log off or if desired delete his/her account as well.*<br><br>***Task 4.6 Development of the tutorials and the chatbot screen* :** *These screens contain the tutorials and a simple chatbot that answer simple user questions. The chatbot should be able to communicate with the appropriate backend processes in order to answer the user's question.* |
| **Deliverables**<br>***D4.1:*** *User Manual (Descriptions in the manual will be based on the UI screens .)*<br>***D4.2:*** *Log-in/sign-up screens*<br>***D4.3:*** *Live presentation and the live feedback screen*<br>***D4.4:*** *Upload presentation video screen*<br>***D4.5:*** *Detailed analysis & feedback screen and the transcript screen*<br>***D4.6:*** *User settings screen*<br>***D4.7:*** *Tutorials and the chatbot screen*<br><br>*Note: The deliverables include the communication with the backend processes as well.* |

Table 6: Work Package 5 describing Video Uploading, Recording Modules

| |
|---|
| **WP 5:** Video Uploading. Recording Modules |
| **Start date:** *13.12.2021* **End date:** *3.04.2022* |

| **Leader:** | *Alper Sarı* | **Members involved:** | Alper Sarı, Barış Tiftik, Burak Yiğit Uslu |
|---|---|---|---|
| **Objectives:** *The video uploading and downloading modules will be necessary for the user to be able to either use the application directly when inputting video data, or for them to receive that data back from the application. Also, said video input will be used for the live feedback modules, as in, said live feedback modules will take the video input in real time before the presentation is finished, after which the full presentation video will be used for a complete analysis.* | | | |
| **Tasks:**<br>***Task 5.1 Recording video*:** *The user will be able to start a video recording of themselves, which will be the input data of the analysis modules.*<br>***Task 5.2 Uploading video:*** *The user will be able to upload an existing video of a presentation.* | | | |
| **Deliverables**<br>***D5.1:*** *Video recording module*<br>***D5.2:*** *Video uploading module* | | | |

Table 7: Work Package 6 describing Presentation Analyzer

| **WP 6:** Presentation Analyzer | | | |
|---|---|---|---|
| **Start date:** *29.11.2021* **End date:** *20.03.2022* | | | |
| **Leader:** | Can Kırşallıoba | **Members involved:** | Can Kırşallıoba, Barış Tiftik, Alper Sarı, Burak, Burak Yiğit Uslu |
| **Objectives:** *Presentation Assistant plays a crucial part in the project, It takes the data which is produced by the machine learning models. Namely, speech to text model and facial recognition model. Presentation Analyzer processes said data into vital information such as the WPM of the presenter, ratio of filler words, time stamps in which the user did not face the presentation etc.* | | | |
| **Tasks:**<br>***Task 6.1 Processing the speech to text data*:** *Processes the data which is given by the speech to text module. It re-formats and prepares the data to be used by the computations.*<br>***Task 6.2 Processing the facial recognition data*:** *Processes the data which is given by the facial recognition module. It holds a tuple which has the timestamps and if the algorithm can detect presenters face in said timestamps.*<br>***Task 6.3 Generate a report:*** *Generates a report containing the transcript of the presenters speech, and other statistics such as the WPM, ratio of filler words etc.*<br>***Task 6.4 Compute the text statistics:*** *Computes the statistics mentioned above using the data given from the machine learning algorithms.*<br>***Task 6.4 Alterations to the text:*** *Receiving the data from the machine learning algorithms it modifies the transcript.* | | | |
| **Deliverables**<br>***D6.1:*** *Presentation Analysis System* | | | |

Table 8: Work Package 7 describing Speech to Text Model

| **WP 7:** Speech to Text Model | | | |
|---|---|---|---|
| **Start date:** *13.12.2021* **End date:** *17.04.2022* | | | |
| **Leader:** | Can Kırşallıoba | **Members involved:** | Can Kırşallıoba, Burak Yiğit Uslu, Can Kırımca, Alper Sarı |
| **Objectives:** *This work package handles the machine learning model which transcribes the speech of the presenter to a transcript, which is a text file.* | | | |
| **Tasks:**<br>***Task 7.1 Plugging in the speech to text model to our codebase*** **:** *We are going to use an already implemented speech to text model, so this task is only about trying and fitting the model to our codebase.*<br>***Task 7.2 Preparing the output of the speech to text model:*** *Deepspeech gives the output in the format of a series of timestamps and the words that the presenter is speaking.*<br>***Task 7.3 Taking the timestamps to analyze the WPM statistics:*** *The model will analyze the presenters speech and take the timestamps in a periodic fashion to calculate the words per minute.* | | | |
| **Deliverables**<br>***D7.1:*** *Report Generation System*<br>***D7.2:*** *Presentation Analysis System* | | | |

Table 9: Work Package 8 describing Facial Recognition Mode

| **WP 8:** Facial Recognition Model | | | |
|---|---|---|---|
| **Start date:** *13.12.2021* **End date:** *01.05.2022* | | | |
| **Leader:** | *Barış Tiftik* | **Members involved:** | *Barış Tiftik, Can Kırımca, Can Kırşallıoba* |
| **Objectives:** *Using several face recognition libraries such as OpenCV (tentative), to develop a presentation face recognizer. It is an essential part of the live feedback mechanism of Prexcel's presentation assistant.* | | | |
| **Tasks:**<br>***Task 8.1 Recognize the presenter's face*** **:** *Implementing the system that checks and tries to recognize the presenter's face, and reports whether his face is well-oriented according to the camera or not.* | | | |
| **Deliverables**<br>***D8.1:*** *Face recognition model* | | | |

Table 10: Work Package 9 describing Presentation Assistant

| **WP 9:** *Presentation Assistant* | | | |
|---|---|---|---|
| **Start date:** *27.12.2021* **End date:** *01.05.2022* | | | |
| **Leader:** | *Alper Sarı* | **Members involved:** | *Alper Sarı, Barış Tiftik, Can Kırımca* |
| **Objectives:** *To develop a presentation assistant that gives live feedback to the user. The presentation assistant is an essential part of the live feedback mechanism of Prexcel. It helps the user correct his face orientation, continue his speech with new words and adjust his speech pace.* | | | |

| |
|---|
| **Tasks:**<br>***Task 9.1 Recognize the presenter's face and prepare an alert*** **:** *This part gets the data from Facial Recognition model about whether the presenter's face is recognized by the camera or not. Subsequently, it counts the duration of face's not being recognized properly. It warns the user after a certain time threshold is exceeded.*<br>***Task 9.2 Show word recommendations*** **:** *This part gets the recommended words from the word recommendation model and sends it to the UI component to be displayed on the screen.*<br>***Task 9.3 Measure the speech pace and warn:*** *It measures the speed of the presenter's voice. According to a certain word per minute rate, it warns the user if his pace is slower or faster than the set word per minute rate.* |
| **Deliverables**<br>***D9.1:*** *Presentation Assistant* |

Table 11: Work Package 10 describing Word Recommendation Model

| | | | |
|---|---|---|---|
| **WP 10:** *Word Recommendation Model* | | | |
| **Start date:** *13.12.2021* **End date:** *01.05.2022* | | | |
| **Leader:** | *Can Kırımca* | **Members involved:** | *Can Kırımca, Can Kırşallıoba* |
| **Objectives:** *To develop a model that analyzes the user's sentences and gives recommendations to the user. The model needs to be able to give accurate recommendations based on the concepts used in the speech and the flow of the sentences.* | | | |
| **Tasks:**<br>***Task 10.1 Obtaining the user's speech in text format generated by the speech-to-text model*** **:** *Implementing this part will form the connection between speech-to-text model and word recommendation model. This will enable the machine learning models to obtain the words as input.*<br><br>***Task 10.2 Using and combining machine learning algorithms to obtain an accurate word recommendation rate*****:** *Several machine learning algorithms will be used in the recommendation process. This task includes the choice, usage and combination of those models. This will be done in a heuristic way. Therefore, the algorithms may be added to and removed from the model to obtain more accurate predictions.*<br><br>***Task 10.3 Detecting the parts where the word recommendation will be visible:*** *The word recommendation will be displayed after the user stopped for a certain duration. This task includes detecting those parts and making the recommendations visible to the user.* | | | |
| **Deliverables**<br>***D10.1:*** *Word recommendation model* | | | |

Table 12: Work Package 11 describing Users & Presentations Database

| **WP 11:** Users & Presentations Database | | | |
|---|---|---|---|
| **Start date:** *29.11.2021* **End date:** *3.04.2022* | | | |
| **Leader:** | *Can Kırımca* | **Members involved:** | *Can Kırımca, Burak Yiğit Uslu* |
| **Objectives:** *To develop a database system to store and retrieve user's presentation data.* | | | |
| **Tasks:**<br>***Task 11.1 Creation of the database schemas:*** *The schemas that describe the entities and relationships used to represent the user data. The schemas will define the data belonging to the presentations and their relationships to the respective user.*<br><br>***Task 11.2 Designing the SQL queries:*** *The user data will be obtained, inserted, deleted and updated by executing SQL queries. Efficient SQL queries will be designed to make the data manipulation accurate and efficient.*<br><br>***Task 11.3 Implementation of the database connection class*****:** *The project contains a class that handles the data manipulation by forming the database connection, executing the SQL queries and transmitting the data between the database and user classes. This class will be implemented in the scope of this task.* | | | |
| **Deliverables**<br>***D11.1:*** *Database tables*<br>***D11.2:*** *SQL queries*<br>***D11.3:*** *Database connection class* | | | |

## 3.4. Ensuring Proper Teamwork

Below, you will see the Gantt Chart we have prepared that outlines our plans for how the work packages will be handled and developed. The lines of the chart represent the group members who will act as the package leaders for said package and be responsible for it, the rest of the members who are planned to work on that package are omitted on the chart but can be seen in the work package table within the previous section. The chart allows us to spread the workload appropriately across the two semesters that it will take for the project to be finished, therefore allowing us to make a fair allotment of work time.

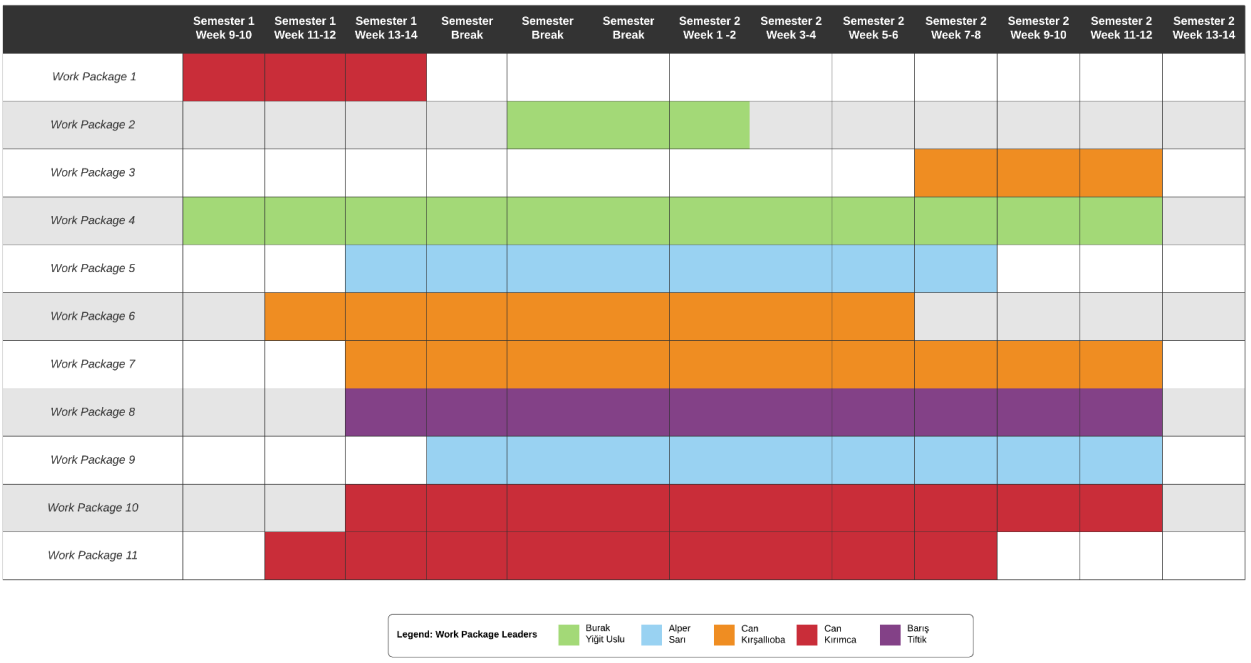


**Figure 23.** Gantt Chart that specifies the time schedule of the project.

## 3.5. Ethics and Professional Responsibilities

There is an array of professional responsibilities and ethical issues related to the development of Prexcel. A major part of these issues are rooted in the fact that the main functionality of Prexcel involves processing and temporarily storing user's personal data. This issue is subject to various laws and regulations throughout the world and these regulations were an important factor to consider during our design of Prexcel. *Kişisel Verileri Koruma Kanunu (KVKK)* [8] in Turkey and *General Data Protection Regulations (GDPR)* [9] in the European Union are the most important regulatory laws to consider in this matter for our case.

In general, a big portion of these regulations are not relevant to our design of Prexcel, because these laws deal with the long-term storing and the processing of this data. They, however, still affected our design of Prexcel because we had to design our software in such a way that we would not have to deal with these laws. Because of this reason, for example, the database we use does not store the auditory input of the user, but rather the transcript of the auditory input. Because, if we stored the auditory input in an auditory form that contains the user's voice, we would have to conform to a very strict set of regulations set by these laws, and we would have to change our design & implementation. In general, we will present the user with an End User License Agreement (EULA) during the account creation process, which openly discloses how the user's personal data is used; which is that it is only stored temporarily, and is deleted after it is processed.

Another professional responsibility related to the development of Prexcel is complying with the licensing regulations of the third party libraries we use. Especially for the backend processes that utilize machine learning techniques, we plan to utilize a number of different libraries. In general, it is important that during the development that we only use the external libraries that allow the usage of themselves for the purposes we plan to use in Prexcel. Since we do not plan on commercializing Prexcel, most libraries do not have clauses that prevent us from using them.

### 3.6. Planning for New Knowledge and Learning Strategies

We have divided the project into three learning areas. The first part is the user interface of the application. For the user interface, we have decided to use JavaScript, React, and Electron. To learn these new technologies we are thinking about using a top down approach, where we first deepen our knowledge about JavaScript then learn about React and Electron. To deepen our knowledge about JavaScript, we thought it would be best to utilize the JavaScript documentation. For React and Electron we are thinking about following an online course plan with assistance of some books that are related to the subject.

The second learning area in the project is the part where we implement the processing algorithms. These processes are going to get their data from the machine learning models and algorithms and produce relevant statistics and data to be displayed to the user. To implement them we thought it would be best to utilize Python as we are all familiar with it. Of course there may be small syntax related problems. To remedy them, we are going to use the Python documentation.

Lastly, for the machine learning implementations; for the part where we use pre-implemented models (such as DeepSpeech) or base models (such as OpenCV), to plug them into our codebase we are going to read through their documentation. And for the models that we are going to implement ourselves we are mostly going to utilize books and tutorials to learn how to do them.

Since everyone in the group has different foundations on different topics, we will follow the different learning plans and strategies. Members of the group who have experience in a particular topic will help the others in the process of learning it.

# 4. Glossary

DeepSpeech: An open source machine learning algorithm that is used for converting speech to text.

Electron: A Javascript library that is used for creating web-based apps on the desktop platform.

Model: Model within the context of this project refers to machine learning models, which is fundamentally a program that is trained to recognize patterns. Within the context of this project, machine learning models will be used for converting users speech to text, to recommend words to the user based on the flow of the sentence, and to decide whether the user is looking at the camera (for the face contact grade).

OpenCV: An open source computer vision library.

Progress-Tracking: Within the context of Prexcel, progress-tracking refers to the comparison of grades between the selected presentations and consequently an overview of the users presenting ability’s improvement.

React: A Javascript framework developed by Facebook. It is used for creating apps for browsers or mobile platforms (using React native). Within the context of the project, it will be used alongside Electron to create a desktop app.

Speech-To-Text: Conversion of auditory input to text format.

# 5. References

[1] DeepSpeech, https://deepspeech.readthedocs.io/en/r0.9/#, [Accessed: Oct. 8, 2021].

[2] OpenCV, https://opencv.org, [Accessed: Oct. 8, 2021].

[3] JavaScript, https://www.javascript.com/, [Accessed: Nov. 14, 2021].

[4] Electron, https://www.electronjs.org, [Accessed: Oct. 10, 2021].

[5] React, https://tr.reactjs.org/, [Accessed: Nov. 14, 2021].

[6] Python, https://www.python.org/, [Accessed: Nov. 14, 2021].

[7] MySQL, https://www.mysql.com/, [Accessed: Nov. 14, 2021].

[8] T.C. Cumhurbaşkanlığı Mevzuat Bilgi Sistemi, "Kişisel Verilerin Korunması Kanunu(KVKK)", Apr. 7, 2016, https://www.mevzuat.gov.tr/MevzuatMetin/1.5.6698.pdf, [Accessed: Oct. 9, 2021].

[9] The European Parliament and the Council of the European Union, "General Data Protection Regulation", Apr. 27, 2016, https://gdpr-info.eu/, [Accessed: Oct. 9, 2021].