



Bilkent University
Department of Computer Engineering

Senior Design Project
T2302
NEUTLAN

Analysis and Requirement Report

21801347, Akmuhammet, akmuhammet@ug.bilkent.edu.tr

21803313, Sila Sarağlu, sila.saraoglu@ug.bilkent.edu.tr

21703920, Berke Ceran, berke.ceran@ug.bilkent.edu.tr

21801781, Lara Merdol, lara.merdol@ug.bilkent.edu.tr

Can Alkan

Erhan Dolak, Tagmac Topal

17.11.2022

Contents

1. Introduction	4
2. Current System	5
3. Proposed System	5
3.1. Overview	5
3.2. Functional Requirements	5
3.2.1. Web Application	6
3.2.2. Extension	6
3.2.3. Web Application and Extension	6
3.3. Non-functional Requirements	7
3.3.1. User-friendliness	7
3.3.2. Reliability	7
3.3.3. Performance	7
3.3.4. Extensibility	7
3.3.5. Scalability	8
3.4. Pseudo Requirements	8
3.4.1. Implementation Requirements	8
3.4.2. Economic Requirements	8
3.4.3. Language Requirements	9
3.4.4. Ethical Requirements	9
3.4.5. Social Requirements	9
3.5. System Models	9
3.5.1. Scenarios	9
3.5.2. Use-Case Model	20
3.5.3. Object and Class Model	21
3.5.4. Dynamic Models	23
3.5.4.1. Activity Diagram	23
3.5.4.1.1. Activity Diagram for Web Page	23
3.5.4.1.2. Activity Diagram for Extension	25
	2

3.5.4.2. State Diagram	26
3.5.5. User Interface	32
4. Other Analysis Elements	37
4.1. Consideration of Various Factors in Engineering Design	37
4.2. Risks and Alternatives	39
4.3. Project Plan	40
4.4. Ensuring Proper Teamwork	44
4.5. Ethics and Professional Responsibilities	45
4.6. Planning for New Knowledge and Learning Strategies	45
5. Glossary	46
6. References	47

Analysis and Requirement Report

NEUTLAN: Neutralize Language

1. Introduction

In today's contemporary world, gender equality is one of the outstanding topics. It is an international concept which many international organizations or even countries are trying to raise awareness about. Therefore, there are several actions taken to this issue however there are many aspects of it such as economic, language, and daily life. For example, to close the gender pay gap the globe needs one hundred and thirty-two years [\[1\]](#). To achieve gender equality in language, we need a Gender-Fair language (GFL). One principle can be obtained to have a gender fair language: neutralization. Neutralization is changing the words with masculine forms to gender unmarked forms [\[2\]](#). For example, today in the formal language and contents, he or she is used to represent a person. By replacing this with they or them is turning the words in an unmarked gender form or instead of using policeman (a masculine form word) police officer can be used to neutralize the word and job in terms of gender. Another significant concept of today's world in terms of gender equality is gender biased artificial intelligence (AI). Today, we see many examples of gender biased AI in many technological applications. A classic example is Google Translate. For example, Turkish is gender neutral language in terms of pronouns so if "This person is president, and this person is cooking" written in Translate, it puts "He's president and she's cooking" [\[3\]](#). It transforms the sentence in a biased way.

From these two significant approaches, we decided to contribute to make an improvement. Our project's name is NEUTLAN which comes from the slogan neutralize the language. In our project, the aim is to neutralize the content of any written text in terms of language by implementing the neutralization principle. In this way, we will act against gender biased ai by forming an unbiased one. Our target group for this project is mainly content creators nevertheless; everyone can use it to write more unbiased content.

In this document, constraints of the project in aspects of many fields such as social context will be provided. Then systems models such as use case diagrams, and activity diagrams will be given. Finally, other analysis of elements will be provided for the project.

2. Current System

NEUTLAN is a greenfield project which indicates that there is no example application like this one. However, there is much research that investigates the datasets about gender-biased content [4, 5]. However, those works are not turned into useful applications that help people in their daily life. Furthermore, some applications provide users with tools for checking their language in terms of grammar such as the Grammarly application which recommends corrections for grammatical errors in the content. In this project we have inspired from those technologies while developing our idea for our project.

3. Proposed System

3.1. Overview

In this project NEUTLAN, we plan to develop a browser extension and a web application which will help users to neutralize their language. It will check the written content and suggest GFL interpretation of gender-biased words or phrases. NEUTLAN will be a product performance innovation because, as far as we know, there are no similar applications, and it holds substantial and sustainable value [6]. To implement this system, we plan to use Natural Language Processing (NLP) technique to capture the gender-biased content in the user input and provide suggestions [7]. We aim to make our application useful by developing both an extension and a web application. Customers will be able to use NEUTLAN while sending e-mails, using translators, or creating social media content by using the extension. In addition, by using the web application, users will be able to check their writings and save them for future use. Users can also upload files instead of using the text editor via web application to check the content of the file. Users can inspect their saved contents later and change the content of them again also. The application will recommend neutral phrases to use as suggestions to the biased phrases in the content. Moreover, the user will see the grade of their written content which depends on the number of biased phrases in the content.

3.2. Functional Requirements

The functionalities listed below will be provided by NEUTLAN.

3.2.1. Web Application

- The user can sign into the application to use, or he/she can use the application without sign in.
- The user can log out from the application.
- The user can type some text content or upload a document to use the application.
- The user can delete her/his account from the system.
- The user can look at the previous content he/she uploaded or typed to analyze before.
- The users can see the content's grade in terms of the corrections and mistakes on their texts or uploaded files.

3.2.2. Extension

- The user can activate the extension to use the application by clicking the validate button on the processing page content.
- If the user wants to use NEUTLAN, he/she should use Mozilla or Chrome browser to use extensions.
- The user can deactivate the extension by removing it.

3.2.3. Web Application and Extension

- The application will underline the mistakes which are sexist contents, meanings or words and give recommendations to change them in a proper way.
- According to the recommendation, the user can choose to change the content but if the user does not click the recommendation the app won't change the text of that sentence or word.
- According to the result of the application, if there is no sexist phrase or meaning in the user's text or the uploaded file and nothing to change, the user will be informed in this regard.
- Before first use of the application, users must confirm a confirmation text for data security.

3.3. Non-functional Requirements

This section will provide information about the non-functional requirements of the NEUTLAN project, namely user-friendliness, reliability, performance, extensibility and scalability.

3.3.1. User-friendliness

- The user interface should be simple and effective in a way that can be easily understood and used by the user. The user interface should not be too complicated because NEUTLAN aims to appeal to literate users of all ages.

3.3.2. Reliability

- The project to be done should correctly distinguish sexist patterns, meanings, and words in sentences or texts, and suggest proper corrections. The sexist structures detected should have a high proper accuracy rate.
- NEUTLAN's servers should not be down frequently or for long periods of time because NEUTLAN will become part of users' daily lives.
- When an error occurs on the program or the operation that the user wants to perform on the program cannot be realized, the user should be informed adequately.

3.3.3. Performance

- The runtime of the application to be made should give feedback to the user as quickly as possible (~20-30s).

3.3.4. Extensibility

- The application should be able to be extensible in terms of adding or developing new features according to the feedback from the innovation experts and supervisors.
- Extension will be developed primarily for NEUTLAN and then the web application will be developed.

3.3.5. Scalability

- NEUTLAN will be introduced in the school first, but after the developments are completed, it is aimed to reach a wide audience and be used by a wide audience. Therefore, NEUTLAN should be scalable in a way that can reach many users.

3.4. Pseudo Requirements

3.4.1. Implementation Requirements

- GitHub platform and Git will be used to sustain consistency and create a development environment for all members. Progress will be monitored through commits [\[8\]](#).
- GitHub Issues will be used to assign tasks for each member and monitor the progress.
- GitHub Actions will be used for automation of CI pipeline for the server. The deployment will be done for the NLP model training.
- Twitter gender biased datasets and Kaggle Sexist Workplace Statements datasets will be used for training. Other datasets found and proved to be useful can be used [\[9\]](#).
- Google cloud servers will be utilized for NLP model training.
- Chrome will be used for testing the extension environment.
- The Firebase product of Google will be used for deployment of web applications.
- To store the new data coming from users will be stored in an external database.

3.4.2. Economic Requirements

- Kaggle does not charge the usage of datasets. However, if a useful database will be found that requires charge, it can be bought.
- Google provides \$ 300 cloud usage for newcomers for the first 3 months [\[10\]](#). Hence, for the first 3 months, it is not expected to be paid for the servers. Later, with optimized usage, servers can be used with charges.
- Deployment of extension will not require a fee.
- Deployment of web applications on Firebase might require some fee after a 3- month trial period.
- The application will be free for the first version.

3.4.3. Language Requirements

- First version of this application will support English language for both in-app and detection of phrases.

3.4.4. Ethical Requirements

- The data collected by the users, who admitted sending, anonymously will not be shared to third parties.
- The data will not be stored in the form where they can be linked with its source.
- The development team will abide by the code of Ethics by the National Society of Professional Engineers [\[11\]](#).
- The data which may be a debated content whether it is sexist or not, several experts may be consulted for determination.

3.4.5. Social Requirements

- Application will not provide any interaction between users.
- The recommendation system provided by the NLP model will be autonomous and not provided by people.
- The processing of texts and recommendations will not be visible by other users.

3.5. System Models

3.5.1. Scenarios

Scenario 1 - Sign Up

Use Case:	Sign Up
Actor:	User
Entry Conditions:	The user opens the website whether from the website link or extension link.
Exit Condition:	The user's account has been successfully created.

Flow of Events:	<ol style="list-style-type: none"> 1. The user presses the sign up button. 2. The user enters her/his credentials. 3. The user presses the submit button. 4. The user will be asked to activate their account from their email.
Alternative Flow of Events:	<ol style="list-style-type: none"> 1. Sign Up Using Google Account <ol style="list-style-type: none"> a. User presses the Google icon. b. The user is redirected to the browser to select the Google account they want to use. c. The user is redirected back to the sign up page.

Scenario 2 - Sign In

Use Case:	Sign In
Actor:	User
Entry Conditions:	<p>The user is signed out previously.</p> <p>The user enters the right credentials.</p>
Exit Condition:	The user is redirected to the welcome page if the sign is successful.
Flow of Events:	<ol style="list-style-type: none"> 1. The user presses the sign in button. 2. The user enters their credentials. 3. The user presses the submit button. 4. The user is redirected to the welcome page.
Alternative Flow of Events:	<ol style="list-style-type: none"> 1. Sign In Using Google Account <ol style="list-style-type: none"> a. The user will sign in using google account. b. The user enters their google account credentials. c. The user clicks continue. d. The user is redirected to welcome page

	<ol style="list-style-type: none"> 2. Sign In is not successful <ol style="list-style-type: none"> a. The user enters wrong credentials. b. The user is informed that they entered wrong credentials. c. The user is redirected to the sign in page.
--	---

Scenario 3 - UserHasNoAccount

Use Case:	UserHasNoAccount
Actor:	User
Entry Conditions:	This use case extends the SignIn use case. It is initiated by the system whenever the user is not registered to the system.
Exit Condition:	The user goes back to the Sign In use case.
Flow of Events:	<ol style="list-style-type: none"> 1. System navigates the user to the signing up process. 2. System asks for the information needed for the signing up process. 3. User enters his/her information. 4. Systems registers the user to the system and displays the acknowledgement to the user.

Scenario 4 - WrongUserNameOrPassword

Use Case:	WrongUserNameOrPassword
Actor:	User
Entry Conditions:	This use case extends the SignIn use case. It is initiated by the system whenever the user is not registered to the system.

Exit Condition:	The user goes back to the Sign In use case.
Flow of Events:	1. System sends acknowledgement to the user that the email or password is wrong.

Scenario 5 - ActivateAsExtension

Use Case:	ActivateExtension
Actor:	User
Entry Conditions:	The user has an account and signed in. The user downloaded the extension of the project.
Exit Condition:	The extension is activated.
Flow of Events:	<ol style="list-style-type: none"> 1. The user downloads the extension. 2. The user presses the extension icon. 3. The user signs in. 4. The user activates the extension.

Scenario 6 - DeactivateAsExtension

Use Case:	DeactivateExtension
Actor:	User
Entry Conditions:	The user already activated the extension.
Exit Condition:	The extension is deactivated.
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the extension icon. 2. The user deactivates the icon.
Alternative Flow of Events:	<ol style="list-style-type: none"> 1. The user logs out from their account.

Scenario 7 - Logout

Use Case:	Logout
Actor:	User
Entry Conditions:	The user is signed in.
Exit Condition:	The user is logged out from their account.
Flow of Events:	<ol style="list-style-type: none">1. The user clicks the logout button.2. The user is redirected to the sign in page.

Scenario 8 - ChangeSettings

Use Case:	ChangeSettings
Actor:	User
Entry Conditions:	The user has an account.
Exit Condition:	A change in settings is done.
Flow of Events:	<ol style="list-style-type: none">1. The user confirms their account by their email or by signing in.2. The user changes a setting.3. The system sends acknowledgement to the user that modification is done.

Scenario 9 - ForgetPassword

Use Case:	ForgetPassword
Actor:	User
Entry Conditions:	The user already has an account.

Exit Condition:	Password is reset.
Flow of Events:	<p>The user forgets their password.</p> <p>The user presses the reset password button.</p> <p>The system sends an email to the user.</p> <p>The user enters the link in the email.</p> <p>The user enters their new password which is different from the last three passwords.</p>

Scenario 10 - ChangePassword

Use Case:	ChangePassword
Actor:	User
Entry Conditions:	This use case extends the ChangeSettings use case. The user is signed in.
Exit Condition:	The password is changed.
Flow of Events:	<ol style="list-style-type: none"> 1. The user goes to the profile page. 2. The user enters their new password. 3. The system sends an email to confirm the change. 4. The user clicks to the link. 5. The system acknowledges the user that change has been made. 6. The user is redirected to the sign in page.

Scenario 11 - DeleteAccount

Use Case:	DeleteAccount
Actor:	User
Entry Conditions:	This use case extends the ChangeSettings use case. The

	user is signed in.
Exit Condition:	The user's account is deleted.
Flow of Events:	<ol style="list-style-type: none"> 1. The user goes to the profile page. 2. The user presses the delete account button. 3. The system sends an email for modification. 4. User enters the link in the email. 5. The system acknowledges the user and deletes the account.. 6. The user is directed to the sign up page.

Scenario 12 - CreateNewDocument

Use Case:	CreateNewDocument
Actor:	User
Entry Conditions:	The user is signed in.
Exit Condition:	A new document is added to the user's account.
Flow of Events:	<ol style="list-style-type: none"> 1. The user presses the create new document button. 2. The user is directed to the document content page.

Scenario 13 - InspectOldDocuments

Use Case:	InspectOldDocuments
Actor:	User
Entry Conditions:	The user is signed in.
Exit Condition:	The user is in one of the old documents pages.

Flow of Events:	<ol style="list-style-type: none"> 1. The user is on the welcome page. 2. The user sees a view of old documents. 3. The user clicks one of the views. 4. The user is directed to an old document page. 5. The user sees the old document's content.
-----------------	--

Scenario 14 - SeeGradeOfContent

Use Case:	SeeGradeOfcontent
Actor:	User
Entry Conditions:	This use case extends the ProcessContent use case. The user is signed in. The user entered a document.
Exit Condition:	The user sees the grade of content.
Flow of Events:	<ol style="list-style-type: none"> 1. The user enters a document via uploading or typing a text. 2. The system gets the content and trains the model. 3. The system calculates a grade. 4. The user sees the grade of the content.

Scenario 15 - ChangeContentAccordingToRecommendations

Use Case:	ChangeContentAccordingToRecommendations
Actor:	User
Entry Conditions:	This use case extends the ProcessContent use case. The user is signed in and created a new document.
Exit Condition:	The user changes the content.
Flow of Events:	<ol style="list-style-type: none"> 1. The user types or uploads a content.

	<ol style="list-style-type: none"> The user sees the recommendations and the grade of content. The user changes one of the parts to the recommended one.
--	--

Scenario 16 - UseTextEditor

Use Case:	UseTextEditor
Actor:	User
Entry Conditions:	The user is signed in. The user is in one of the new documents pages.
Exit Condition:	The user types text content.
Flow of Events:	<ol style="list-style-type: none"> The user presses the create new document. The user is in the create new document page. The user types the content.

Scenario 17 - UploadFile

Use Case:	UploadFile
Actor:	User
Entry Conditions:	The user is signed in. The user is in one of the new documents pages.
Exit Condition:	The user uploads a file.
Flow of Events:	<ol style="list-style-type: none"> The user presses the create new document. The user is in the create new document page. The user uploads a file from their device.

Scenario 18 - GetDocumentsAsData

Use Case:	GetDocumentsAsData
Actor:	Administrator
Entry Conditions:	The system has several users and their documents.
Exit Condition:	The system uses the data to train the model
Flow of Events:	<ol style="list-style-type: none"> 1. The administrator gets the document of users as data. 2. The administrator uses that data to train the model again.

Scenario 19 - TrainModel

Use Case:	TrainModel
Actor:	Administrator
Entry Conditions:	This use case extends the GetDocumentsAsData use case. The system has several users and their documents.
Exit Condition:	The system trains the model
Flow of Events:	<ol style="list-style-type: none"> 1. The system trains the model.

Scenario 20 - MonitorAnalytics

Use Case:	MonitorAnalytics
Actor:	Administrator
Entry Conditions:	The system has several users and their documents.
Exit Condition:	The system analyzes the data it has.

Flow of Events:	<ol style="list-style-type: none"> 1. The administrator collects the data. 2. The administrator calculates analytics. 3. The administrator shares the analytics with the users via email.
-----------------	--

Scenario 21 - ProcessContent

Use Case:	ProcessContent
Actor:	User
Entry Conditions:	The user is signed in.
Exit Condition:	The content user enters is started to process both by the user and the model.
Flow of Events:	<ol style="list-style-type: none"> 1. The user signs in. 2. The user clicks to create a new document or starts to inspect an old document. 3. The user starts to process/manipulate the content and the model starts to evaluate the result.

3.5.2. Use-Case Model

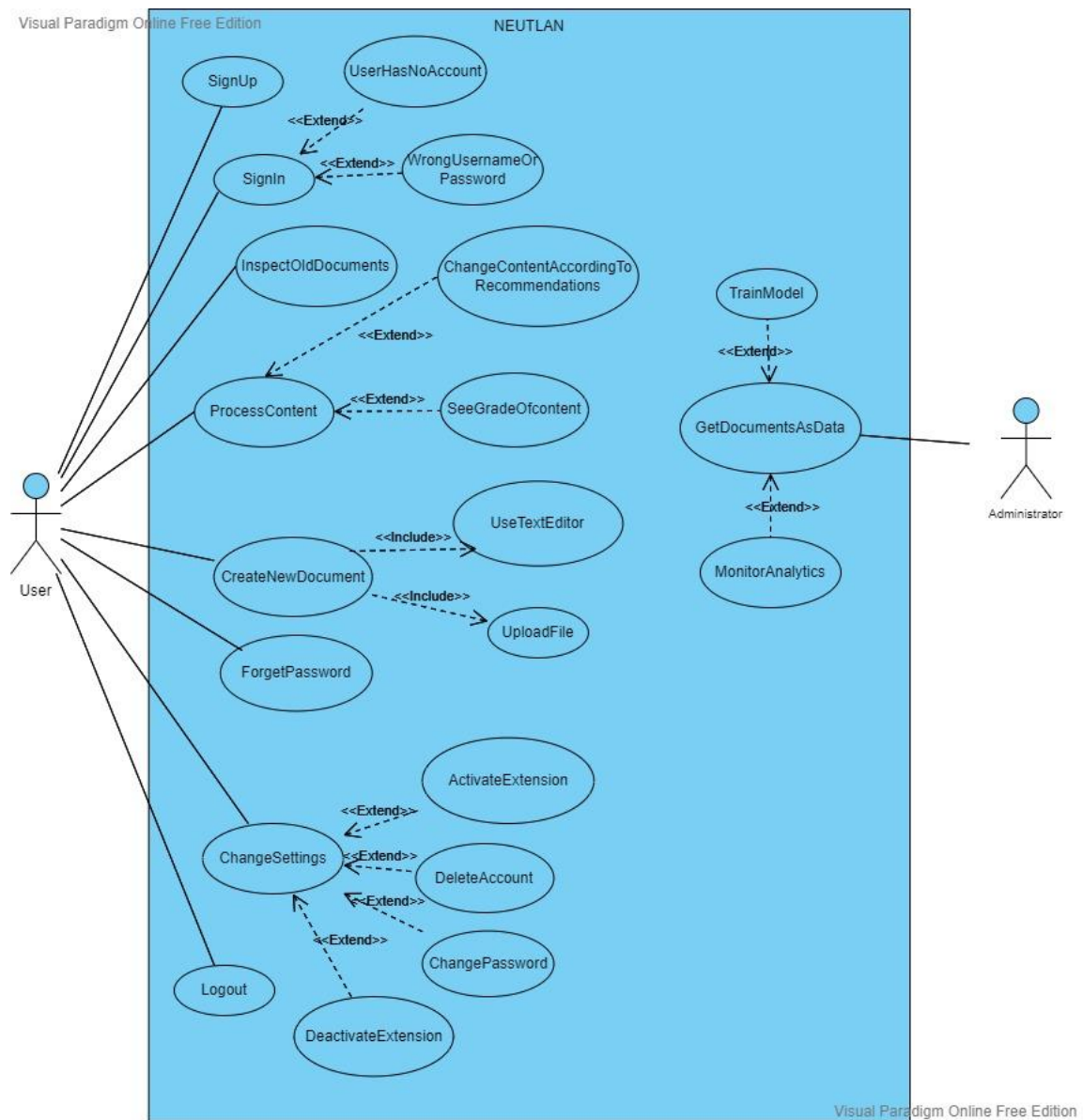


Figure 1: Use Case Diagram

3.5.3. Object and Class Model

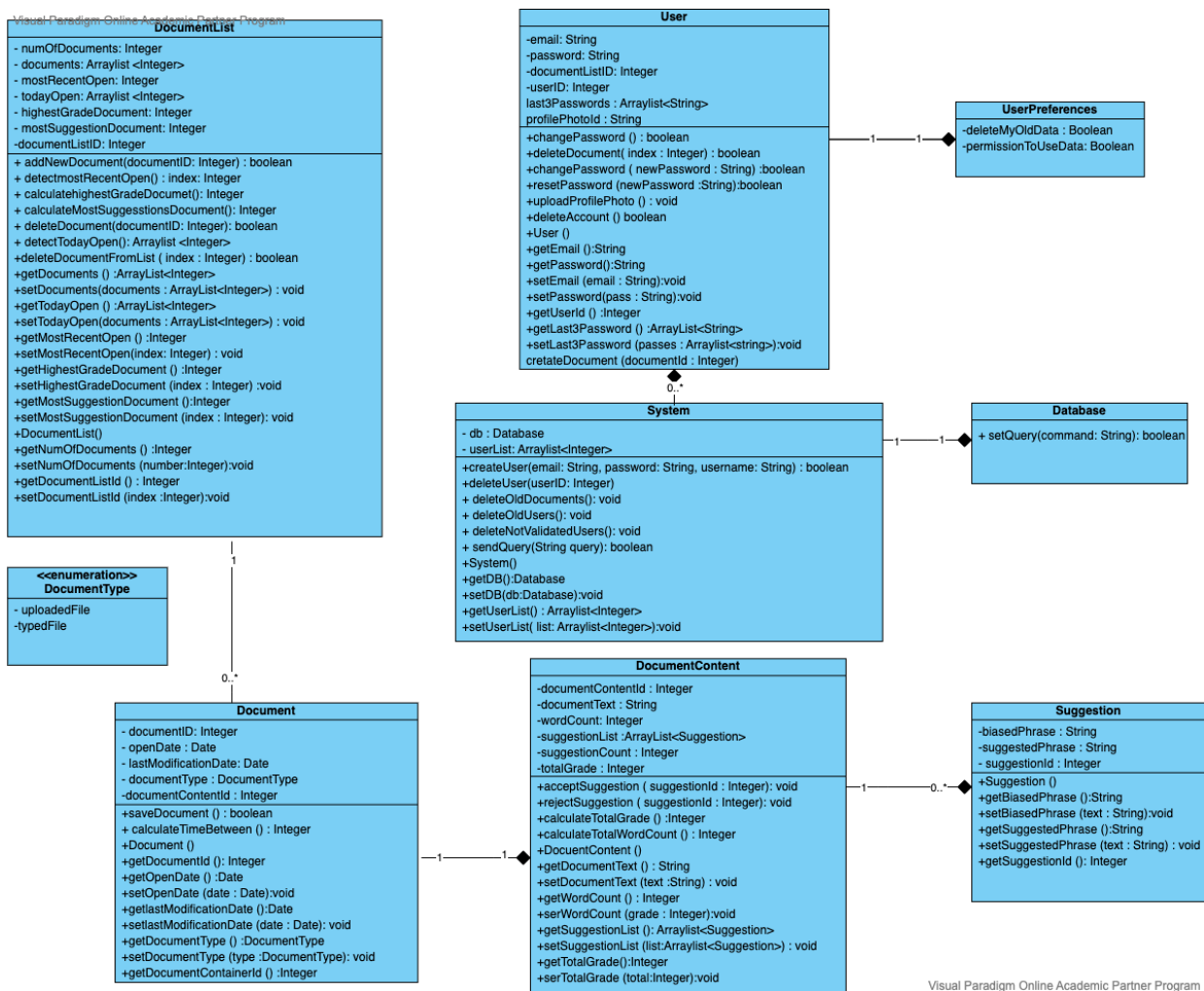


Figure 2: Class Diagram

Our class diagram consists of 9 classes. In the below part we have discussed each type separately;

- **User Class:** Our first class is the user class. Our system holds users' personal information and login information such as their email, password, name, preferences, etc to provide users with better customer service and documentList that holds users' edited documents. Users can modify that information by changing or resetting their password and also create new documents and also delete documents that they want from their document list.
- **UserPreferences Class:** Our system also holds the personal preferences of users such as "*deleteMyOldData*" which enable the system to automatically delete the documents

that the user does not perform any edit for a long time. Or other preferences like permission to use their data for training the model.

- **DocumentList Class:** The other class is the documentList class which holds a collection of users' edited documents. Each time a new user is created we create an empty documentList inside that user class. When a user creates or deletes a document the content of the documentList object changes accordingly. This object also holds statistical information like the "mostRecentOpen", "mostSuggestionDocument" or "highestGradeDocument" to provide more information to users about their collection and increase usability.
- **Document Class:** When a user creates a new document a new "Document" object is created. This object holds the date information like openDate , lastModificationDate to perform some statistics but also helps users to track their old not used documents. On the other hand, the document holds the type information and the id of the documentContent.
- **DocumentContent Class:** DocumentContent class is the one which holds the content of the documents in our application. It holds the attributes such as its' id, text, the count of the words, the suggestion list, number of suggestions and total grade. This class provides the functionalities such as accepting or rejecting suggestions, calculating the total grade and total word count.
- **Suggestion Class:** Suggestion class is the one which holds each suggestion as the old format (biased phrase) which user is written if the phrase is biased then system recommends another phrase (suggested phrase). This class also contains the suggestion id.
- **System Class:** This class is the main class which is responsible for operations in the application which controls all classes in a way. This class has attributes db, and user list. To access the database we will use this db attribute. To access users, their documents etc. we will use the user list. This class can create and delete users from the database. This class can delete some old and unvalidated users and documents which are not used for a particular long time. This class will also send queries to the database.
- **Database Class:** This class is the representation of the database class. It will directly communicate with the database of the application.

3.5.4. Dynamic Models

3.5.4.1. Activity Diagram

3.5.4.1.1. Activity Diagram for Web Page

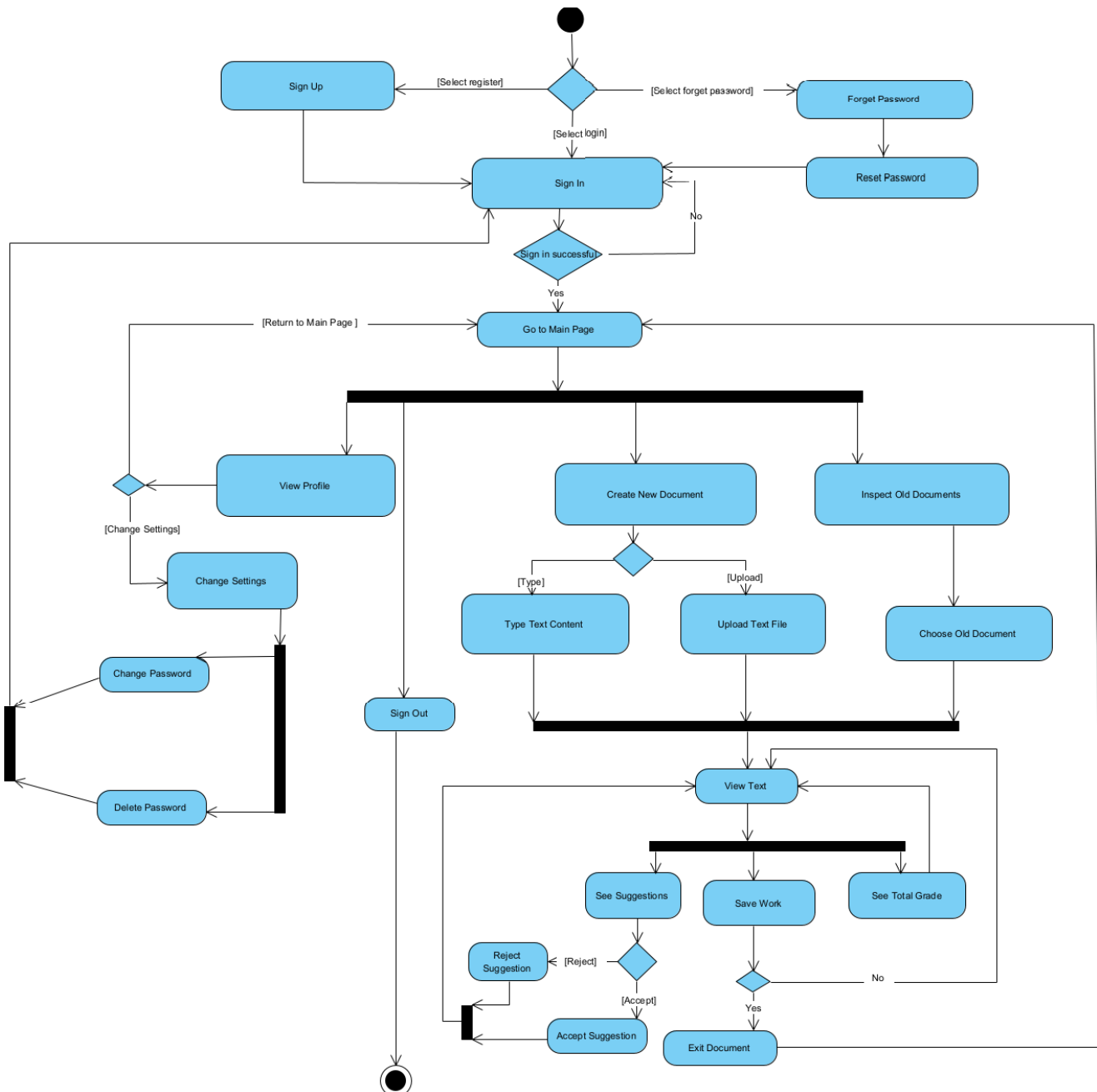


Figure 3: Activity Diagram for Web Page

The primary flow of how the NEUTLAN web application operates is described in the above activity diagram. When the user first enters, they decide to sign up or sign in. If the user already has an account they sign in and the flow begins otherwise they first register for the

application. Users can change their login settings from their profile. But if they do that they have to sign in the system again with the new login settings. The app's primary functions are to scan the given text input which can be uploaded or typed, identify the gender-biased content, and recommend suggestions. Users can accept or reject suggestions and edit the document accordingly. The web application also lets the user see a general grade in terms of reaching gender neutral language. In addition users can also see and edit their past documents which they have uploaded before.

3.5.4.1.2. Activity Diagram for Extension

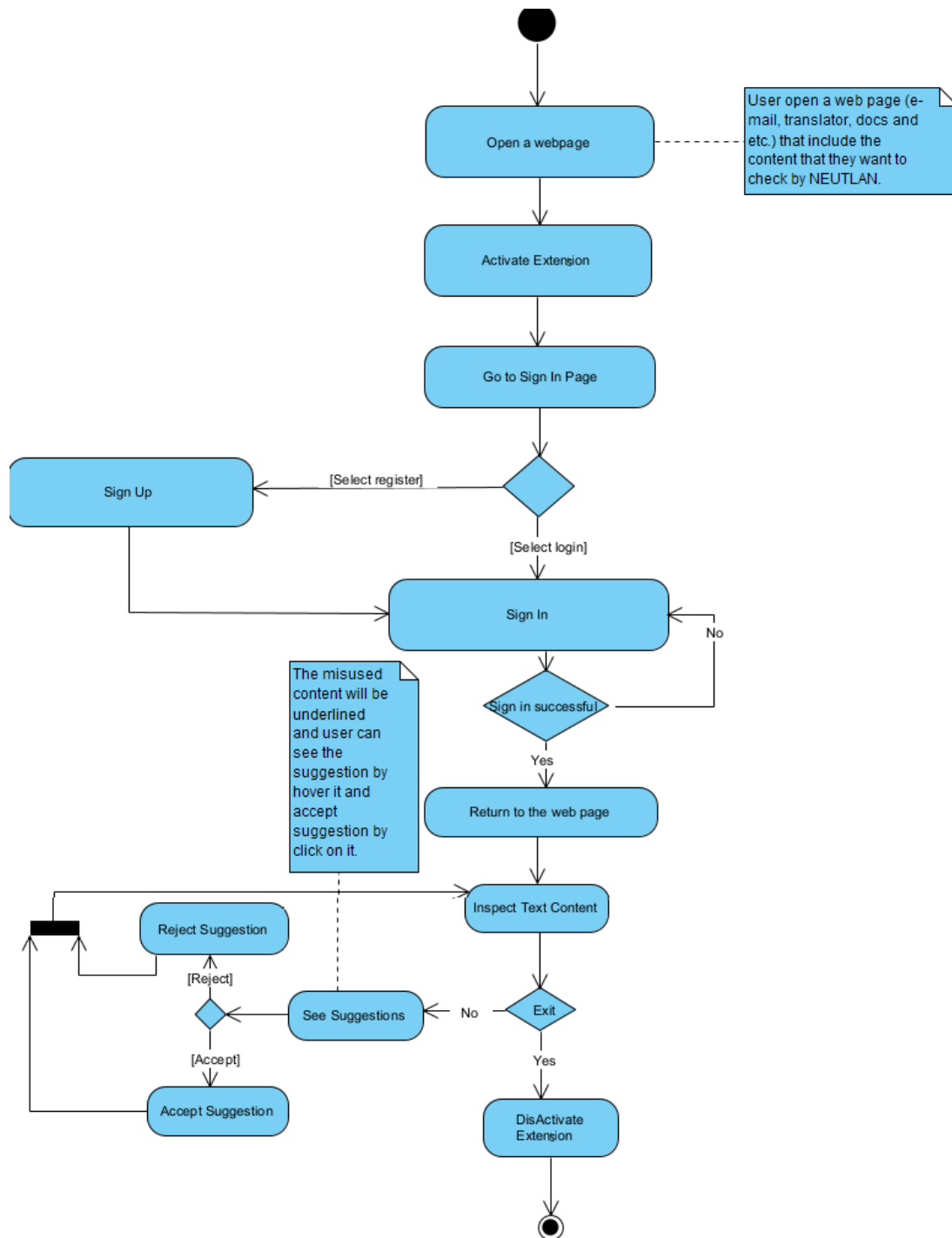


Figure 4: Activity Diagram for Extension

In addition to web applications, users can use the web extension that provides them with additional functionalities. The primary flow of how the NEUTLAN Extension operates is described in the above activity diagram. Flow begins when the user opens a webpage that they

want to neutralize its language. Then they turn on the extension from their browser extension manager. Because our system requires authenticated entries for both systems if they are not already signed in, they are directed to the same sign-in sign-up page with the web application. After the authentication step, users can use the extension for various web pages that their browser enables (Gmail, Docs, Translators, Twitter, etc) by hovering over the underlined segments to see the suggestions and clicking to accept them.

3.5.4.2. State Diagram

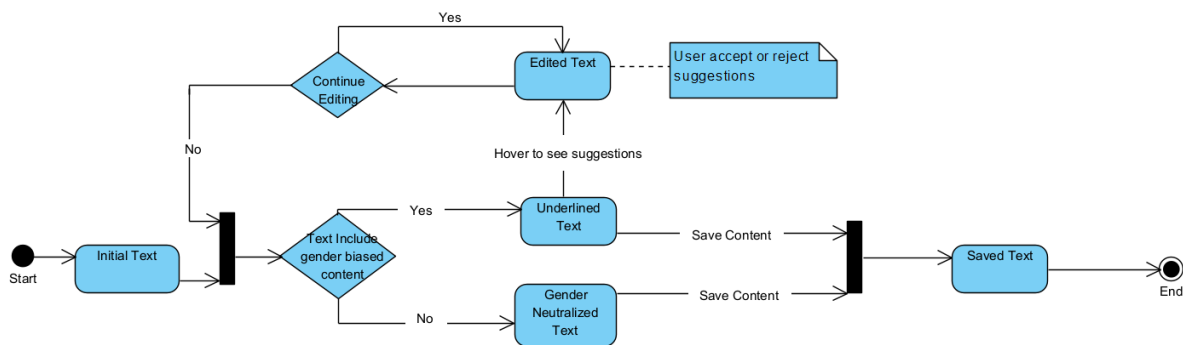


Figure 5: Text State Diagram

The above diagram shows the lifecycle of the text component which the user wants to neutralize by the NEUTLAN service. When the user input the text content by various methods described by the above parts, it comes to its first state “Initial Text”. After that state, if the text does not contain any gender-biased content it becomes “Gender Neutralized Text” otherwise it becomes “Underlined Text”. During the editing process by accepting or rejecting suggestions, text enters the state “Edited Text”. While the user continues editing, the text component stay in this state. If the user stops the editing process, and the text does not contain any gender-biased content it enters the “Gender Neutralized Text” and if the user saves the content it enters its last state “Saved Text”.

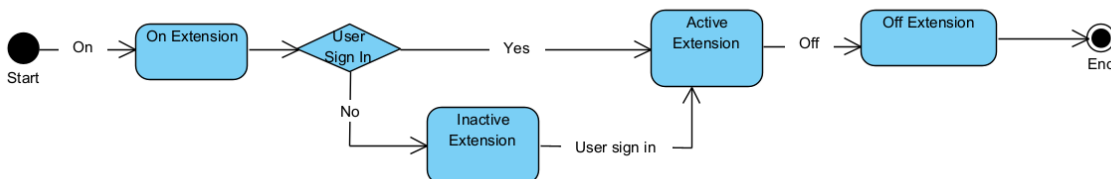


Figure 6: Extension State Diagram

The above diagram shows the lifecycle of the extension. When the user is on the extension from the extension manager, the extension starts its life cycle with its initial state “On Extension”. If a user is not signed in it enters the “Deactivate Extension” state and waits for that state until the user signs in. If the user signed in to the extension enters the “Active Extension” state and stays in that state until the user of the extension. When the user is off, the extension enters its last state “Off Extension”.

3.5.4.3. Sequence Diagram

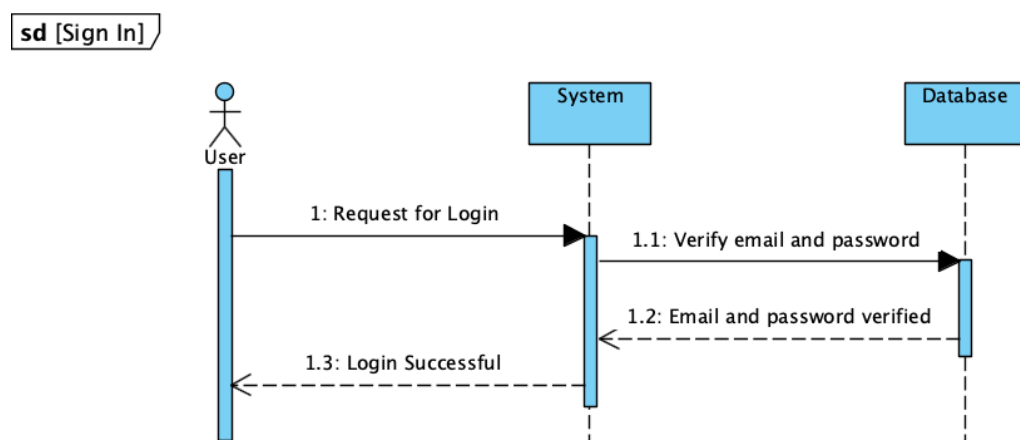


Figure 7: Sing In Sequence Diagram

This sequence diagram is the case of user signing in. First, user enter their credentials and submit it then system verifies the credentials and send the message to the user if login is successful.

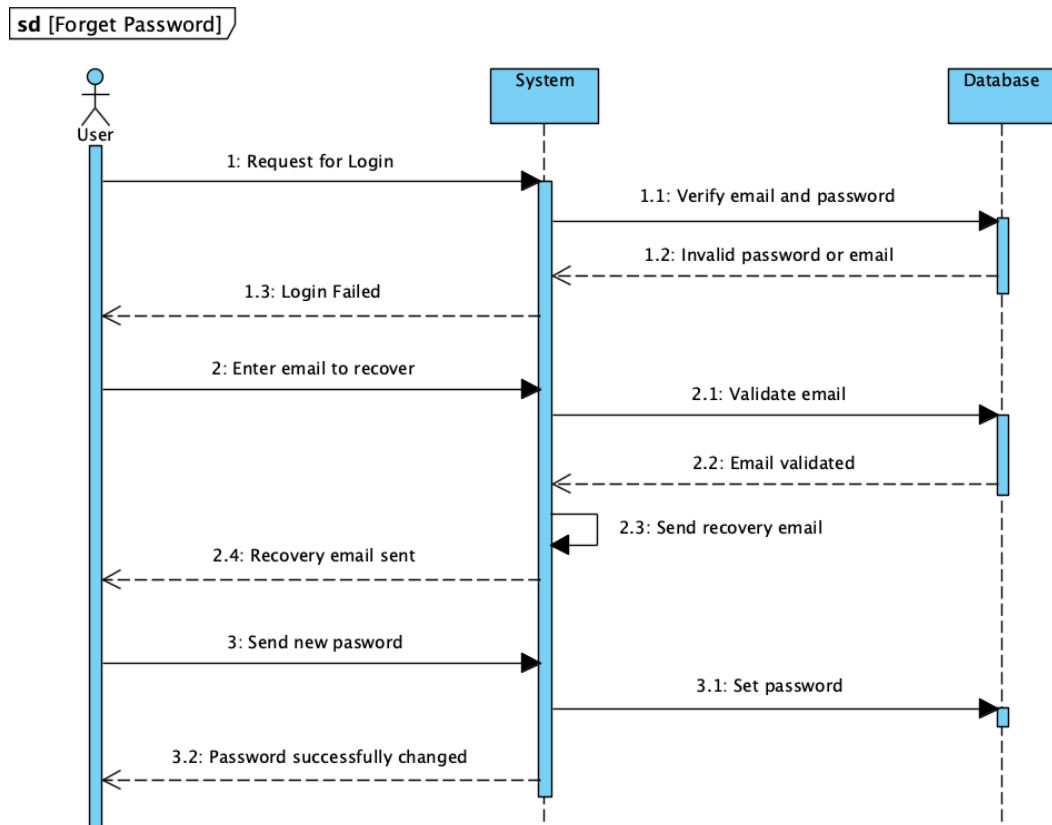


Figure 8: Forget Password Sequence Diagram

This diagram is the representation of the case when a user forgets their password and tries to get a new one. First user enters their credentials and the system sends the message that login is unsuccessful because they enter the wrong credentials. Then the user requests the forget password function. User enters their email and the system checks whether that email does exist in the database then the system sends a recovery email to the user's email. Then the user enters that email and the system lets the user enter another password. Then systems set the new password to the database and informs the user.

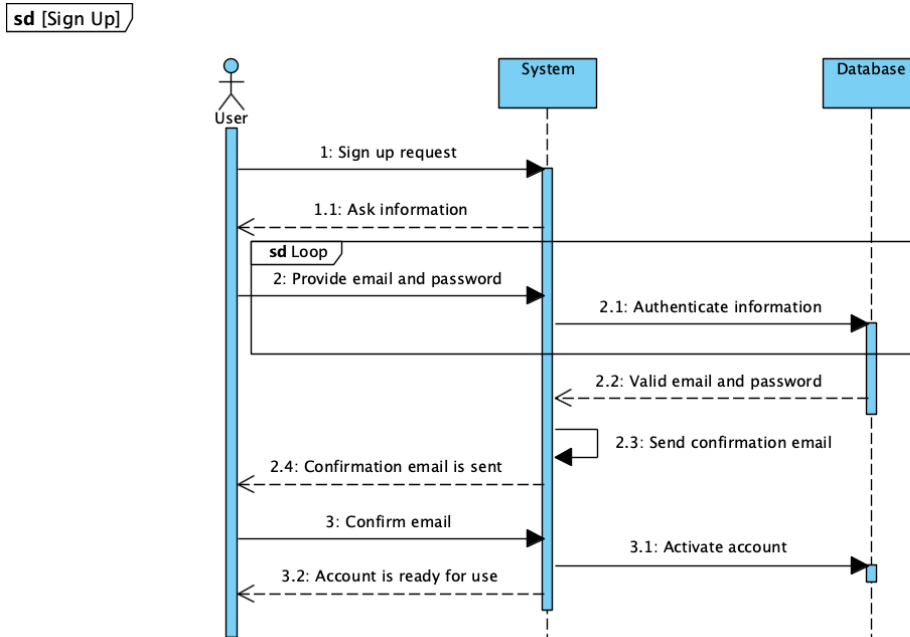


Figure 9: Sign Up Sequence Diagram

This diagram is the representation of the sign up use case. First user requests a sign up request to the system. Then the system wants users to enter their credentials (info). Then the system validates the email and password and sends the user the validation email to their email address. If user validates their email from their email page then the system activates the account and shows that account is activated information to the user.

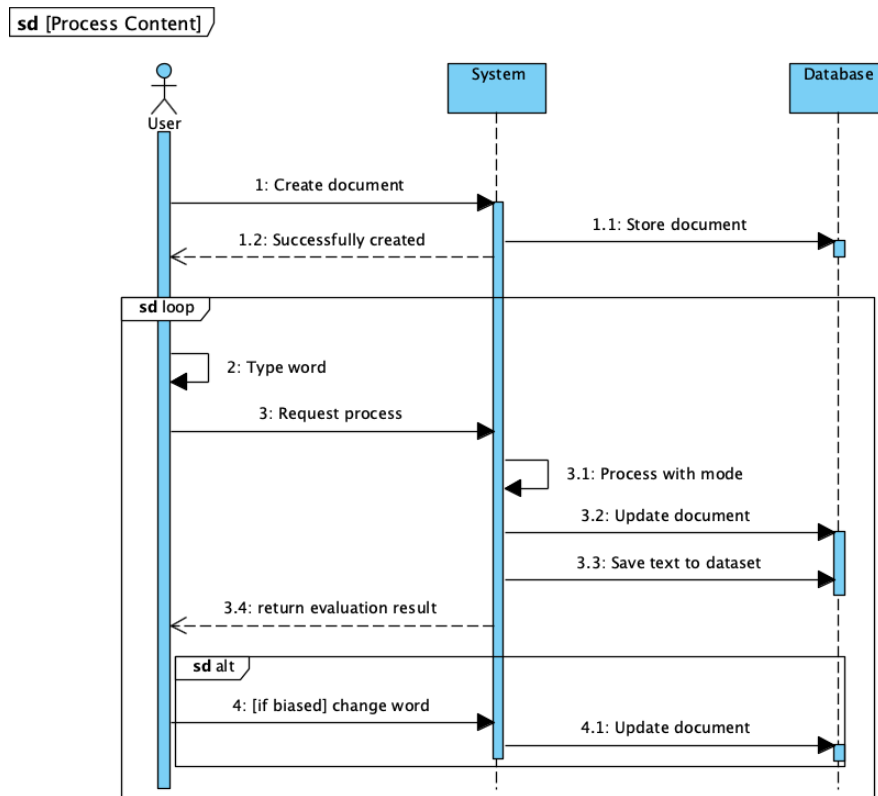


Figure 10: Process Content Sequence Diagram

This diagram represents the process content use case which is the case of manipulating and changing the content in a loop flow. First, the user creates a new document and the system stores this document in the database linked to the particular user and sends the successful message to the user. Then, users type words in the text editor and in a loop each time the user enters the words, the system trains the model to evaluate the content, update the document to database and save the text to a dataset which will be used to train the model. Then, the system returns the evaluation result to the user and the system gives recommendations. Then, if user changes the words according to the evaluation result, system updates the document again.

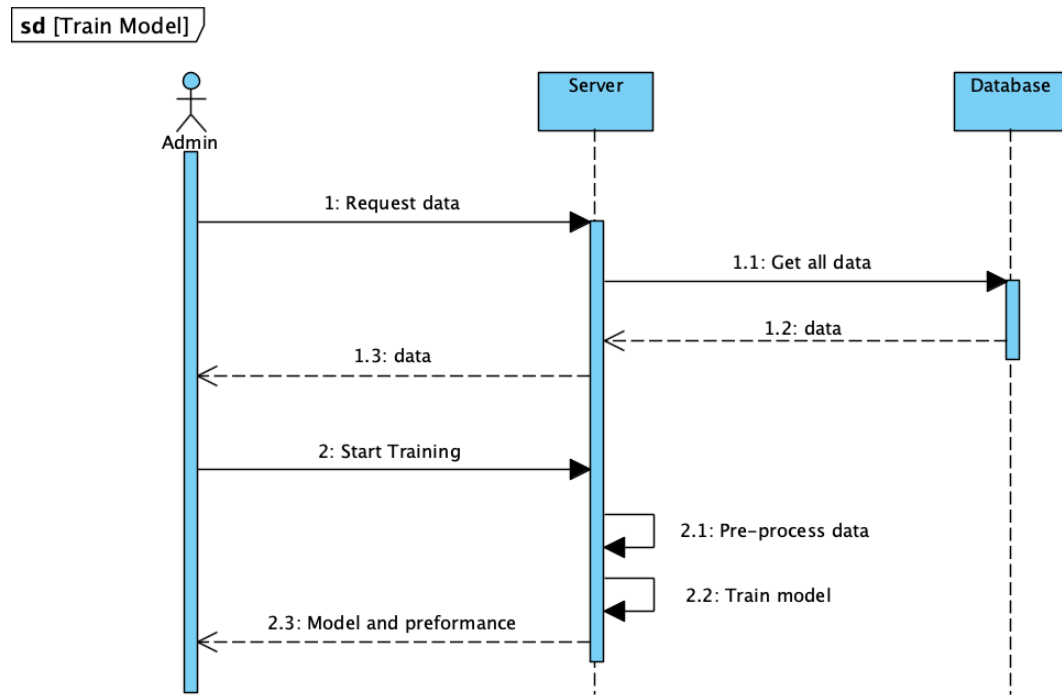
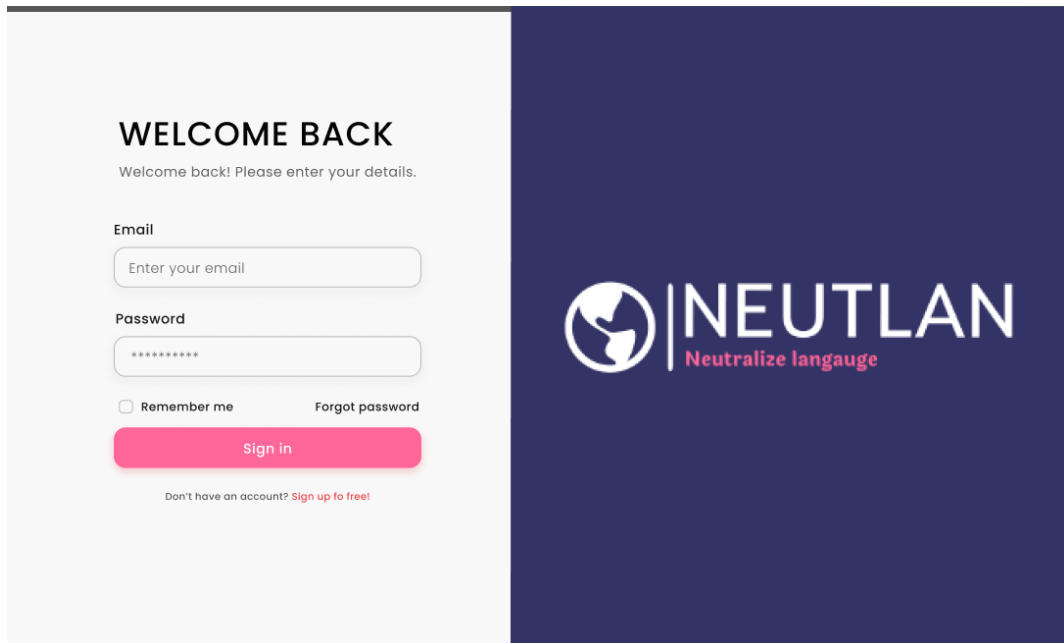


Figure 11: Train Model Sequence Diagram

This is the representation of the train model use case. This sequence is done by the admin of the application. First, admin requests data to server then server gets all the data from database then admin starts to training the model with the dataset they got. Then after the server trains the model it shows the model and performance to the admin.

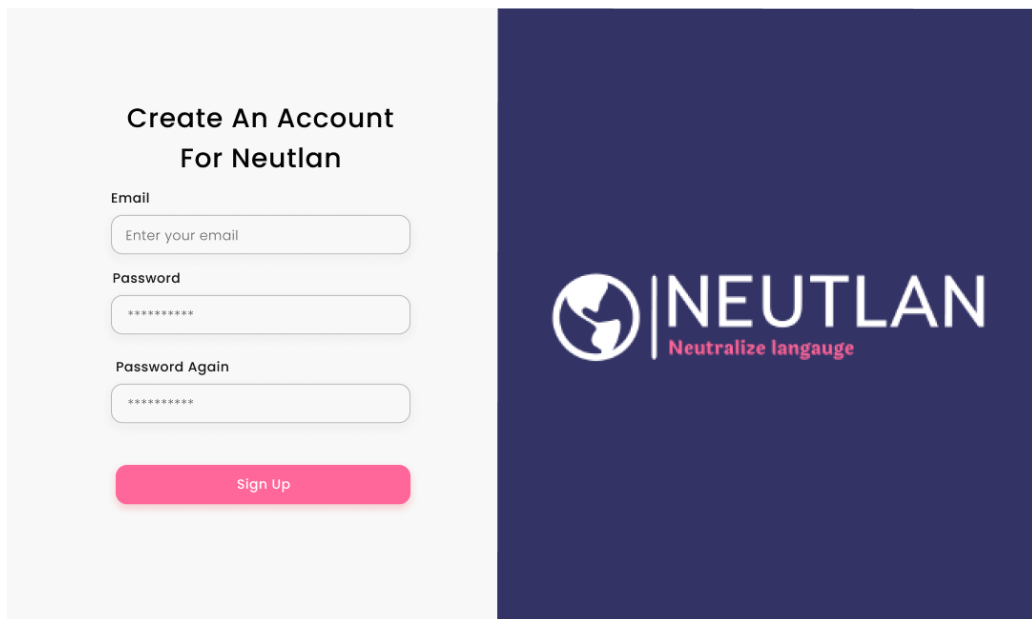
3.5.5. User Interface



The login page features a light gray background on the left and a dark blue background on the right. The right side contains the NEUTLAN logo, which consists of a white globe icon with a stylized 'N' inside, followed by the text 'NEUTLAN' in white and 'Neutralize language' in pink below it. On the left, the heading 'WELCOME BACK' is centered, followed by the subtext 'Welcome back! Please enter your details.' Below this are two input fields: 'Email' with the placeholder 'Enter your email' and 'Password' with a masked password '*****'. There is a checkbox labeled 'Remember me' and a link 'Forgot password'. A prominent pink 'Sign in' button is centered below the inputs. At the bottom, a link reads 'Don't have an account? Sign up to free!'

Figure 12: Login Page

On the login page seen above, the user can enter the application by entering his email and password. In addition, if he clicks on the remember me field, his next entries will be made automatically. In addition, if the user has forgotten his/her password, he/she can change his password by clicking the forgot password field, or if he/she does not have an account before, he/she can sign up for the application by clicking the sign-up field.



The create account page has a similar layout to the login page. The right side features the same NEUTLAN logo. The left side has the heading 'Create An Account For Neutlan'. Below it are three input fields: 'Email' (placeholder 'Enter your email'), 'Password' (masked '*****'), and 'Password Again' (masked '*****'). A pink 'Sign Up' button is centered at the bottom of the form area.

Figure 13: Sign Up Page

On this page, the user can register for the application by entering his/her e-mail and the password he/she has determined into the system.

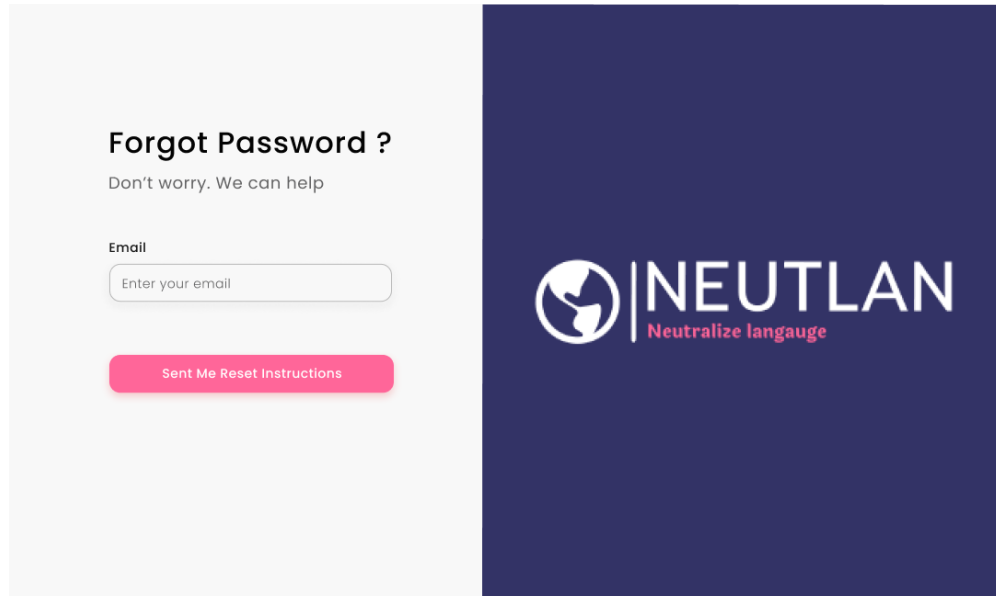
The image shows a web form for forgetting a password. It is split into two vertical panels. The left panel has a light gray background and contains the text 'Forgot Password ?' followed by 'Don't worry. We can help'. Below this is a label 'Email' and a text input field with the placeholder 'Enter your email'. At the bottom of the left panel is a pink button labeled 'Sent Me Reset Instructions'. The right panel has a dark blue background and features the NEUTLAN logo, which consists of a white globe icon and the text 'NEUTLAN' in white, with 'Neutralize language' in pink below it.

Figure 14: Forgot Password Email Sent

This page and the page below work together and allow the user to change his password with the password change link sent to the email address of the user who forgot his/her password.

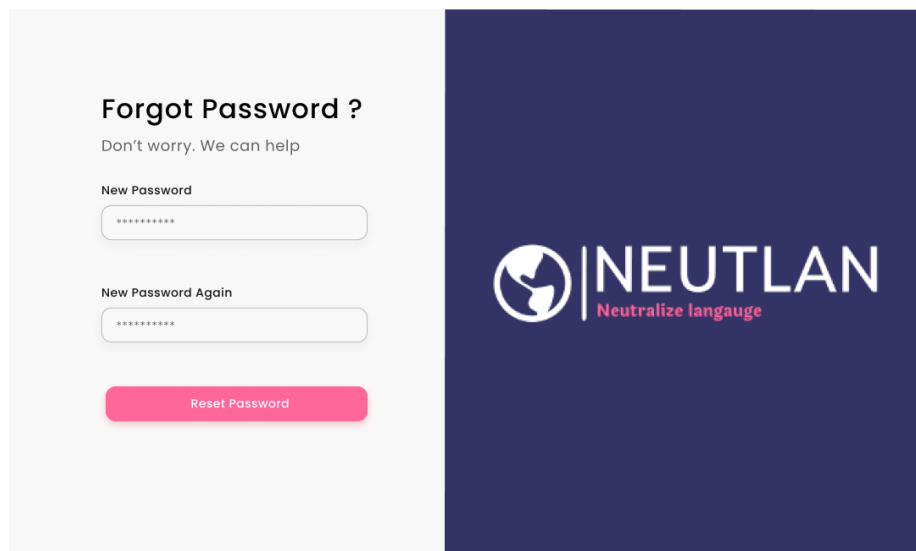
The image shows a web form for resetting a password. It is split into two vertical panels. The left panel has a light gray background and contains the text 'Forgot Password ?' followed by 'Don't worry. We can help'. Below this are two labels: 'New Password' and 'New Password Again', each followed by a text input field with a masked password (represented by dots). At the bottom of the left panel is a pink button labeled 'Reset Password'. The right panel has a dark blue background and features the NEUTLAN logo, which consists of a white globe icon and the text 'NEUTLAN' in white, with 'Neutralize language' in pink below it.

Figure 15: Forgot Password

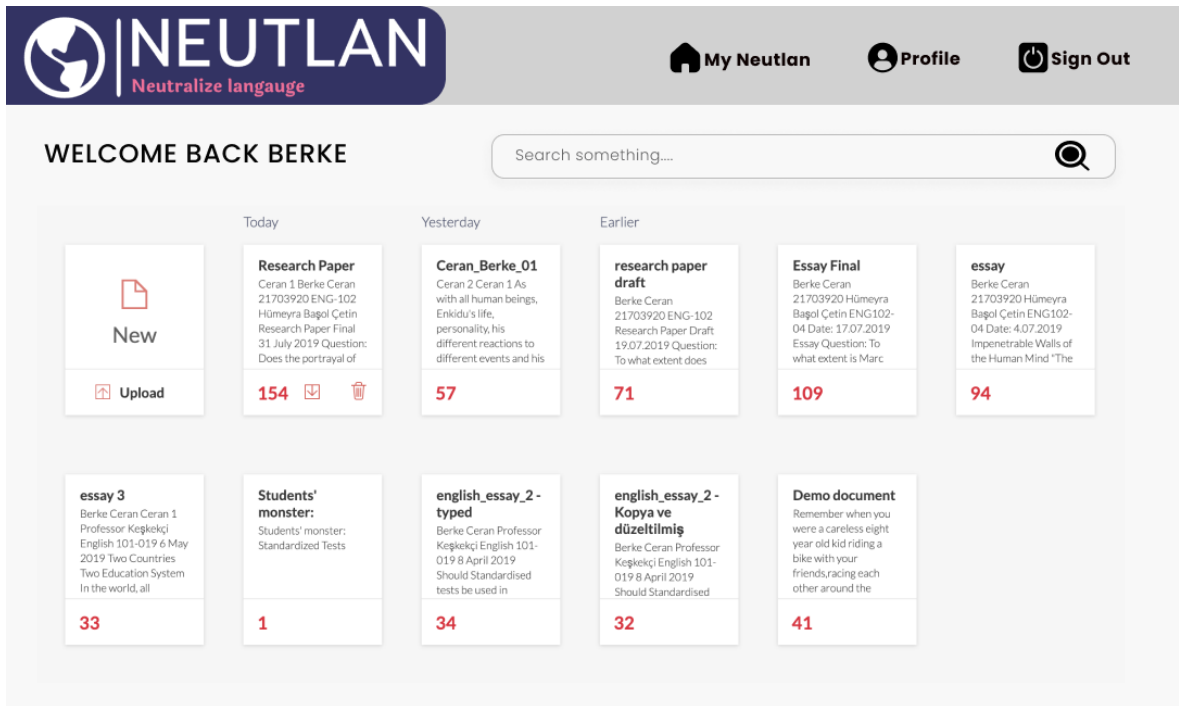


Figure 16: Home Page

This page is the main page of the application. The user can do various things on the page. You can go to the profile page, sign out or return to the main page via the menu at the top of the page. In addition, there is a welcome message specific to the user, a search bar that searches for file names so that the previous files can be found more easily, and a form of the user's previous files sorted according to certain dates. In addition to these, the user can open a window to write new articles on this page, upload an article or download old articles and delete them from the page. In addition, the number of suggestions for articles written or uploaded by the user is displayed on each article.

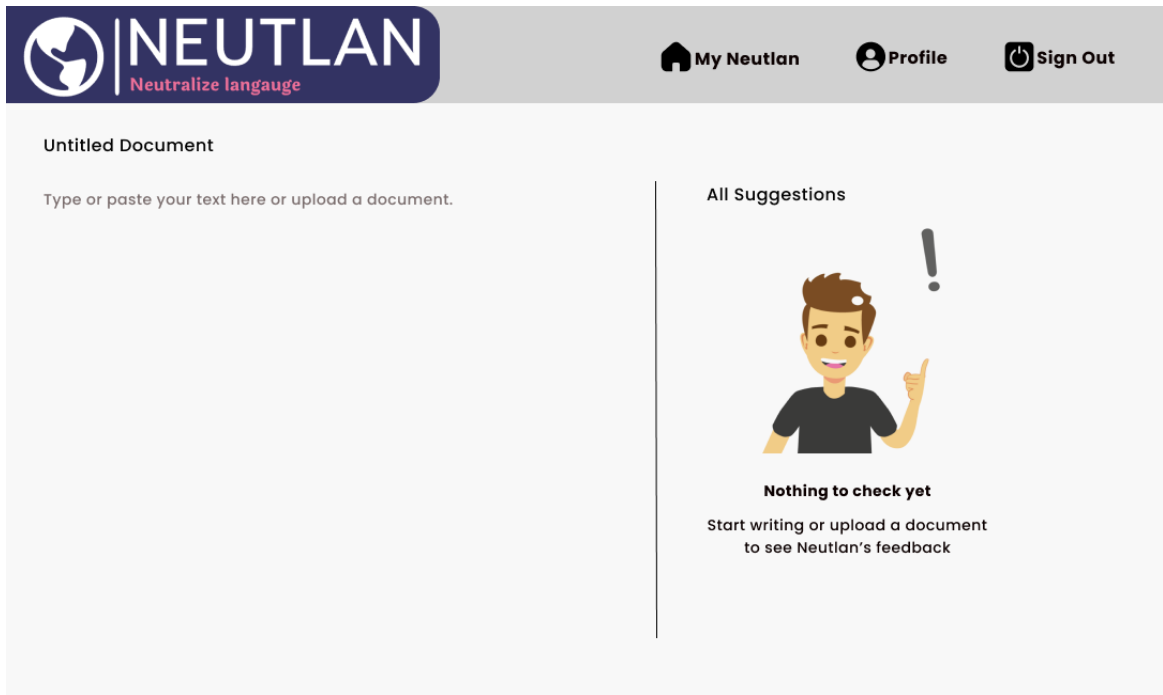


Figure 17: Writing Feature Page

The user can write the text they want to write on the left side of this page and receive feedback about their text from the suggestion section on the right side as seen in the picture below.

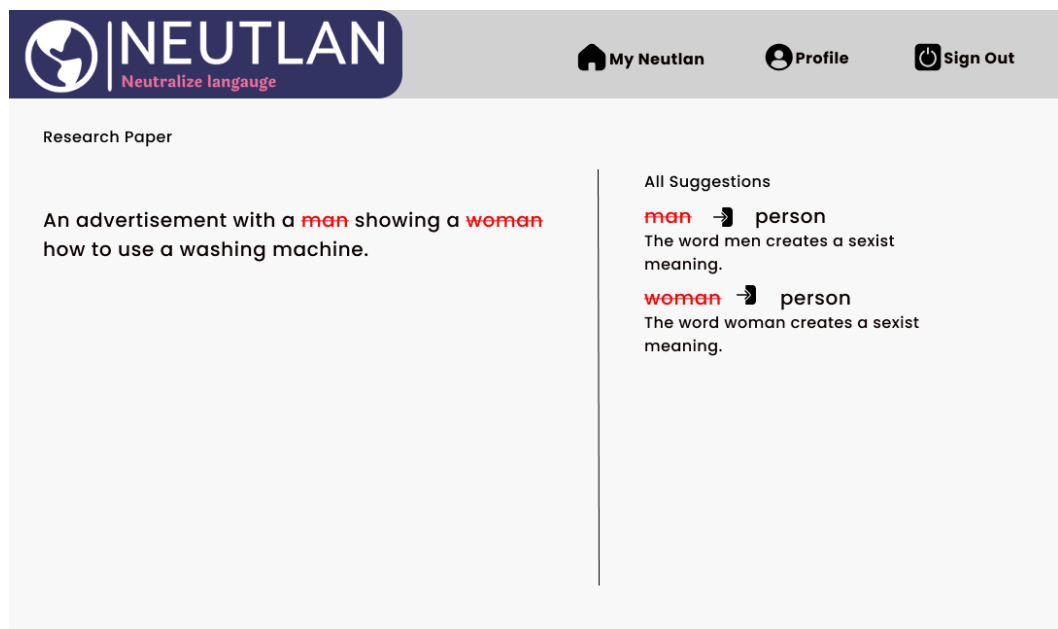


Figure 18: Suggestion Page



Enable to check for writing suggestions on Google Chrome ☒

Written language American English

+ New Document

Neutlan Home

Settings

Neutlan has been correcting your sexist text since November 5, 2022

Figure 19: Extension

Finally, this image is the extension image of the application that will be found on Google Chrome. Through this extension, the user can enable and disable the extension, change the text language, add documents by connecting to the main application from the new document section, go to the settings section and finally see how long he/she has been using the NEUTLAN application.

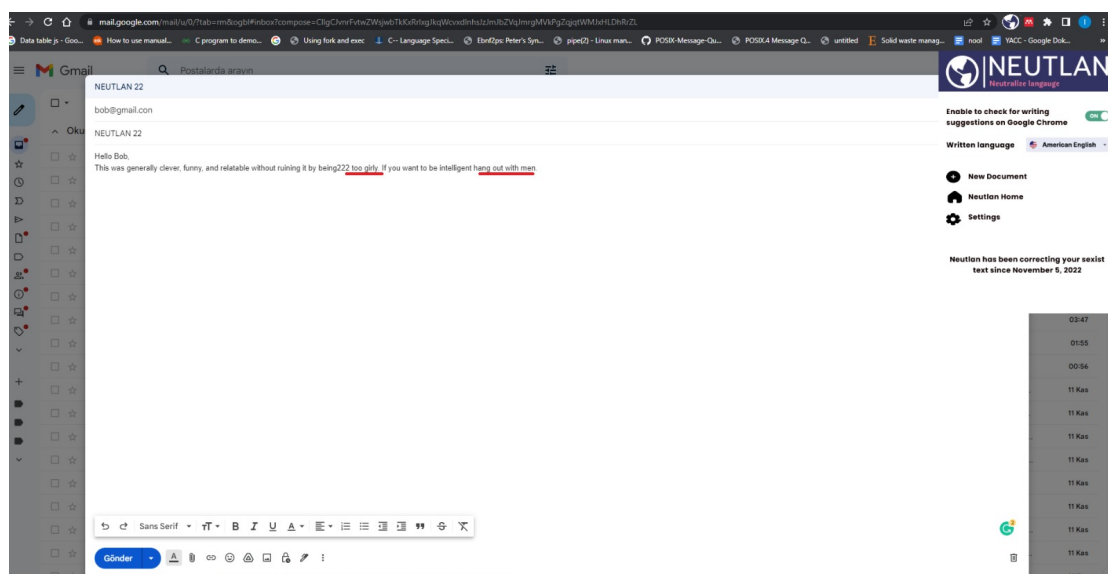


Figure 20: Extension in use

4. Other Analysis Elements

4.1. Consideration of Various Factors in Engineering Design

During the research and implementation processes, there are several factors that may have either small or significant impact to our project NEUTLAN. These factors are explained in the following sections.

4.1.1. Public Safety

NEUTLAN will detect the gender biased words and phrases for those who will be using it. By eliminating them, it will enable users to establish healthy communication and prevent partners from being offended in case of an usage of those words or phrases accidentally.

4.1.2. Individual's Privacy

According to the choice of the users, the data will be collected. However, that data should not be linkable to the user. Hence, the gathering of the data will have to be anonymous in order to ensure the privacy of individuals. Also, those who will not allow their data to be sent will, their choices will be respected and their data will not be collected.

4.1.3. Global Factors

NEUTLAN is expected to be published for global usage since English is used widely and usage of gender biased language is not a single country's problem. However, due to some differences in versions of English, such as British, US, Canadian, etc., and there are many versions of English, NEUTLAN will not differentiate among them and will be able to be used for all of them.

4.1.4. Social Factors

NEUTLAN will create social awareness by helping to reduce gender biased language usage. Due to the many Chrome browser users and the convenience of using extensions in it, many audiences will be reached.

4.1.5. Economic Factors

NEUTLAN will not pursue any profit by providing this service to the users. Hence, the registration and usage of this application will be free. However, the costs of the implementation and deployment of the application will be paid by the developers.

The factors which are described above might cause the changes in our analysis and design due to some unforeseen problems. The problems and their effect on our analysis and design are summarized in the table given below. The scale for each factor is also set to put the level of importance.

Table 1: Factors that can affect analysis and design.

	Effect level	Effect
Public Safety	10	If the model is performing with low accuracy, then a different approach needs to be investigated to prevent people being offended.
Individual's Privacy	7	Investigating some anonymization techniques for ensuring anonymity if it is found the person's privacy is at risk.
Global Factors	6	If there are huge differences in English versions, then only a limited number of them (or one of them) will be supported.
Social Factors	0	In order to reach more users, other

		browsers will not be considered for implementation.
Economic Factors	7	If the cost is high, then the implementation will be reconsidered.

4.2. Risks and Alternatives

The NEUTLAN aims to neutralize language from gender biased words and phrases. In order to achieve this, a NLP model powered by Neural Networks will be used. However, as with all other networks, it will not always produce the correct outcome as detecting the words and phrases or labeling wrong ones. Hence, with incorrect evaluation, the language cannot be fully neutralized. If the program is unable to detect the biased words or phrases, the outcome might be offensive to the partner being involved in communication or the individual's reputation damage if models are being processed in important documents such as articles or legal documents. If it is being detected as causing substantial damage to the society, as breaking relationships or causing scandals, then additional programs can be implemented with a hash table for detection of specific words or phrases. However, it will increase the memory as well as the time complexity of the application causing the slow performance. Another important issue is related to the individuals' privacy in case that the data would become linkable to the users and serious security issues will be encountered. In such cases, some anonymization techniques will be investigated. However, the implementation of the data anonymization might decrease the utility of the data as it would be processed. Last important risk is due to the costs of the implementation and deployment of the application. It is being planned to use Google services since it is providing a \$300 bonus for the new-comers. Since the project will be presented at the end of the spring in 2023, there is a possibility of payments for the training after some time. Also, if the number of users will increase substantially, then the new servers with higher RAM and storage needs to be utilized to meet the demand of the users which might create switching costs. Hence, if it is found to be too expensive to continue, then the

alternatives of the services will be considered and the number of users or functionalities (limit to the file or processing numbers) can be restricted.

The following table describes the risks as well as their likelihood of the risk mentioned above. Also, plans are presented in case those risks will turn out to be true.

Table 2: Risks

	Likelihood	Effect on the project	B Plan Summary
Wrong labeling	8	Labeling biased words/phrases as unbiased or contrary can offend or anger people.	Implementing hash table addition to the model to make sure the correct labeling.
Privacy disclosure or data leak	2	The privacy can be compromised or data can be leaked.	More advanced anonymization techniques will have to be implemented.
High costs	4	The costs of the applications, servers, platforms can be high	The alternatives might be considered. The number of users might be limited. The performance will not be improved further.

4.3. Project Plan

The project is divided into the work packages that contain the plans and aims for the project. The work packages contain all sub-project it can be considered as a sprint plan lasting for different amounts of time related to the workweight of the sub-project.

Table 3: List of work packages

WP#	Work package title	Leader	Members involved
WP1	Web	Berke Ceran	Everyone
WP2	Extension	Berke Ceran	Everyone

WP3	Backend	Sıla Saraoğlu	Everyone
WP4	Model	Lara Merdol	Everyone
WP5	Testing	Sıla Saraoğlu	Everyone

WP 1: Web			
Start date: 10.01.2023 End date: 01.04.2023			
Leader:	Berke Ceran	Members involved:	Berke Crean Akmuhammet Sıla Saraoğlu Lara Merdol
Objectives: Web application will be implemented			
Tasks: Task 1.1 create web page project : Create project Task 1.2 integrate with backend : Create connection with backend Task 1.3 file management integration: Upload, modify and delete file implementation Task 1.4 model integration: integrate model and process text			
Deliverables D1.1: Web application			

WP 2: Extension			
Start date: 14.11.2022 End date: 10.01.2023			
Leader:	Berke Ceran	Members involved:	Berke Crean Akmuhammet Sıla Saraoğlu Lara Merdol
Objectives: Extension will be implemented			
Tasks: Task 2.1 create extension project : Create project Task 2.2 integrate with backend : Create connection with backend Task 2.3 modify text: Target and modification of text implementation Task 2.4 model integration: integrate model and process text			
Deliverables			

D2.1: Web extension project

WP 3: Backend

Start date: 1.12.2022 **End date:** 10.02.2023

Leader:	<i>Sıla Saraoğlu</i>	Members involved:	<i>Berke Crean Akmuhammet Sıla Saraoğlu Lara Merdol</i>
----------------	----------------------	--------------------------	---

Objectives: *Backend will be implemented*

Tasks:

Task 3.1 create backend project : *Create project*

Task 3.2 integrate with frontend : *Create connection with frontend*

Task 3.4 model integration: *integrate model and process text*

Deliverables

D3.1: *Backend application*

WP 4: Model

Start date: 14.11.2022 **End date:** 01.04.2023

Leader:	<i>Lara Merdol</i>	Members involved:	<i>Berke Crean Akmuhammet Sıla Saraoğlu Lara Merdol</i>
----------------	--------------------	--------------------------	---

Objectives: *NLP model will be trained*

Tasks:

Task 4.1 create model project : *Create project*

Task 4.2 training different models: *Train different models being investigated*

Task 4.4 model integration: *integrate model with backend*

Deliverables

D4.1: *Model*

WP 5: Testing			
Start date: 14.11.2022 End date: 01.04.2023			
Leader:	Sıla Saraoğlu	Members involved:	Berke Crean Akmhammet Sıla Saraoğlu Lara Merdol
Objectives: Testing of web and extension and backend			
Tasks: Task 5.1 Manual Testing: Consistency between reports and the application functionalities Task 5.2 UI Testing of Web and Extension Project: UI testing without integrating with the backend Task 5.3 Backend API Testing: Backend request testing			
Deliverables D5.1: Tested and Revised Application			

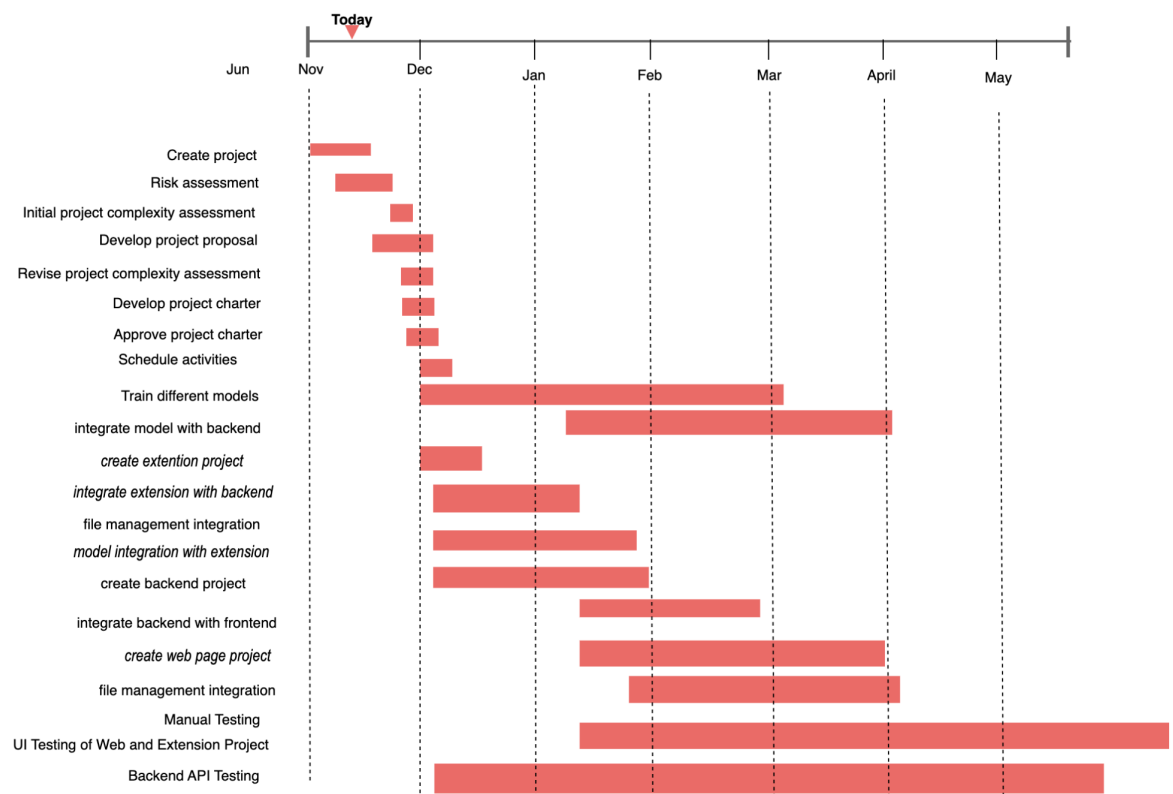


Table 5: Gantt Chart

4.4. Ensuring Proper Teamwork

The task assignment will be done through GitHub issues and the progress will be tracked from there. For convenience, the project is divided into 4 sub projects which are Extension, Web, Backend and Model. However, each sub-project will not be assigned to the single member of the group, instead, each member will have some role in every part of the project. The workweight that will be assigned to the member and the subproject will vary. The distribution of workweight to members is given below in the table.

Table 6: Work distribution to each member.

	Extension	Web	Backend	Model
Sila	0	5	50	5
Lara	10	20	5	35
Berke	85	70	5	5
Akmuhammet	5	5	40	55

As you can see, the numbers are arranged in a way that the sum of the column, representing the sub-project, is equal to 100. However, the row is not guaranteed to be equal to 100 since the weights of sub-projects are not equal.

4.5. Ethics and Professional Responsibilities

The application will store the users sensitive data such as name, surname, email and password. However, they will not be distributed or shared with 3rd parties. Also, collecting the data for model improvement will be done by the consensus of the user where he/she will be giving the application on registration. However, users will also be provided with the option to change that setting.

Since the data of the users is highly sensitive and confidential, there is a responsibility that the privacy of individuals is ensured. It is also crucial to sustain the secure deployment of the application to prevent any leakage.

In order to prevent copy-right issues, only open source applications, models and libraries will be used. Also, if there is a property right issue in utilization of application, models and libraries, appropriate permissions will be taken (in return of payment or any condition being asked to be met for the usage of them).

The agreement with users will be done for the terms being mentioned above during the registration.

4.6. Planning for New Knowledge and Learning Strategies

The aim of this course is to provide students with some sense of being software engineer and be part of a team that will try to solve the problem that society is facing. Therefore, this project will require not only using the program, languages and concepts being taught so far, but learn new ones, as well. Hence, the documentations, online platforms, articles and tutorials will be the main source of learning. Also, another important actor from whom will be advised and taken feedback will be the course instructors, Erhan Dolak, Tagman Topal and our supervisor Can Alkan.

5. Glossary

Term	Definition
GFL	Gender-fair language (GFL) aims at reducing gender stereotyping and discrimination.
NLP	Natural language processing is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data.

6. References

- [1] B. Baschuk, "Global gender gap: WEF says it's narrowing but will take 132 years to get parity," Bloomberg.com, 13-Jul-2022. [Online]. Available: <https://www.bloomberg.com/news/articles/2022-07-13/wef-gender-gap-narrows-after-pandemic-delayed-parity-by-decades?leadSource=uverify+wall> [Accessed: 16-Oct-2022].
- [2] S. Szczesny, M. Formanowicz, and F. Moser, "Can gender-fair language reduce gender stereotyping and discrimination?," Frontiers. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fpsyg.2016.00025/full> [Accessed: 16-Oct-2022].
- [3] A. Madgavkar, "A conversation on Artificial Intelligence and gender bias," McKinsey & Company, 01-Aug-2022. [Online]. Available: <https://www.mckinsey.com/featured-insights/asia-pacific/a-conversation-on-artificial-intelligence-and-gender-bias> [Accessed: 16-Oct-2022].
- [4] J. Doughman, W. Khreich, M. El Gharib, M. Wiss, and Z. Berjawi, "Gender bias in text: Origin, taxonomy, and implications," *Proceedings of the 3rd Workshop on Gender Bias in Natural Language Processing*, 2021.
- [5] M. Samory, I. Sen, J. Kohne, F. Flöck, and C. Wagner, "'call me sexist, but...': Revisiting sexism detection using psychological scales and adversarial samples," *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 15, pp. 573–584, 2021.
- [6] "Product performance innovation," *Product Performance Innovation | BA 850: Sustainability Driven Innovation*. [Online]. Available: <https://www.e-education.psu.edu/ba850/node/720>. [Accessed: 16-Oct-2022].
- [7] IBM Cloud Education, "What is natural language processing?," IBM. [Online]. Available: [https://www.ibm.com/cloud/learn/natural-language-processing#:~:text=Natural%20language%20processing%20\(NLP\)%20refers,same%20way%20human%20beings%20can](https://www.ibm.com/cloud/learn/natural-language-processing#:~:text=Natural%20language%20processing%20(NLP)%20refers,same%20way%20human%20beings%20can). [Accessed: 16-Oct-2022].
- [8] "Where the world builds software," GitHub. [Online]. Available: <https://github.com/>. [Accessed: 16-Oct-2022]

- [9] D. Grosz, "Sexist workplace statements," Kaggle, 28-Apr-2020. [Online]. Available: <https://www.kaggle.com/datasets/dgrosz/sexist-workplace-statements>. [Accessed: 16-Oct-2022].
- [10] "Cloud computing, hosting services, and apis | google cloud," Google. [Online]. Available: <https://cloud.google.com/>. [Accessed: 16-Oct-2022].
- [11] "Code of ethics," Code of Ethics | National Society of Professional Engineers. [Online]. Available: <https://www.nspe.org/resources/ethics/code-ethics>. [Accessed: 16-Oct-2022].