Term Project

Functional Spec

CS 151: Object-Oriented Design
Prof: Ahmad Yazdankhah

Team Number	01	**Section**	04
Team Members	1. Ajay Singh 2. Landon Shishido 3. Shun Furuya		
Software Name	Recommend-Easy		

1 Introduction

1.1 Objective

This document has a detailed description of our software's vision and a sequence of use-cases, and the audiences of this document are the design team and our client. In the end, we have included some mock-ups to better illustrate our software.

1.2 References

The only tools we have used to create this document are Adobe XD to design our mock-ups and the functional specification template provided.

1.3 Acronyms, and Abbreviations

N/A	

2 Software Overview

2.1 Problem Statement

Writing recommendations for students who want to apply for higher education is one of the faculties' responsibilities at any level.

Faculties usually spend some time to compile all required information they need for preparing a suitable recommendation. Furthermore, a student may apply for several schools.

Therefore, faculties need to keep track of the recommendation, written for a student, to be able to reuse it later if the student applies for other schools.

There is no open source and free personal tool for compiling recommendations.

2.2 Scope

Our software's target audience are faculties working at high schools, community colleges and universities.

2.3 Functional Requirements

1. Faculties shall be able to sign-up easily using their email address.
2. Faculty shall be able to change the provided default password once they login.
3. Faculty shall be able to create/edit/delete a letter of recommendation using the template.
4. Faculty shall be able to edit a text field if it needs correction.
5. Faculty shall be able to search a letter of recommendation from the home screen.
6. At last, they shall be able to save their work, and log out of the software to return to the login page.

3 Use Cases

Use Case Name	UC-01: Login

Goal

A description for how a user login into the system

Participating Actors

user and system

Glossary

N/A

Primary Flow of Events

Trigger

User launches the application

Steps	Action	System Response
1	Open the application	Display user login screen
2	User enter the password	1. Checks user password in database
		2. Display main window to user

Alternate Flow of Events

Alternate Trigger

User enters wrong value for password

Steps	Action	System Response
1	Enters wrong password	Display “wrong password” message
2	User re-enters correct password	User login succeeds
3		

<table>
<tr><td><b>Use Case Name</b></td><td colspan="2">UC-02: Log out</td></tr>
<tr><td colspan="3"><b>Goal</b></td></tr>
<tr><td colspan="3">A description for how a user log out the system</td></tr>
<tr><td colspan="3"><b>Participating Actors</b></td></tr>
<tr><td colspan="3">user and system</td></tr>
<tr><td colspan="3"><b>Glossary</b></td></tr>
<tr><td colspan="3">N/A</td></tr>
<tr><td colspan="3"><b>Primary Flow of Events</b></td></tr>
<tr><td colspan="3"><b>Trigger</b></td></tr>
<tr><td colspan="3">User presses the “Logout” button</td></tr>
<tr><td><b>Steps</b></td><td><b>Action</b></td><td><b>System Response</b></td></tr>
<tr><td>1</td><td>Press the “Logout” button from Main window</td><td>Display the login window again</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td colspan="3"><b>Alternate Flow of Events</b></td></tr>
<tr><td colspan="3"><b>Alternate Trigger</b></td></tr>
<tr><td colspan="3"></td></tr>
<tr><td><b>Steps</b></td><td><b>Action</b></td><td><b>System Response</b></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
</table>

Use Case Name	UC-03: Delete Recommendation

Goal

A description for how a user can delete a completed recommendation

Participating Actors

User, System

Glossary

N/A

Primary Flow of Events

Trigger

User searches for the recommendation letter using Student’s last name

Steps	Action	System Response
1	User enters Student’s last name in Search option	Shows matching list of letters in a ListView
2	User clicks the letter that needs to be deleted	System enables “Edit” and “Delete” option for user
3	User pressed the “Delete” button	Draft and Edited version is permanently deleted from “letters” directory matching Student’s last name
4		

Alternate Flow of Events

Alternate Trigger

User clicks “Edit” button instead of “Delete” button

Steps	Action	System Response
1	User clicks “Edit” button instead	Shows the details window to the user
2	User clicks “Go back” button	Shows the main window again
3	User clicks “Delete” button now	Deletes the selected letter

Use Case Name	UC-04: Create new Recommendations

Goal

A description for how to create a new recommendation

Participating Actors

User, System

Glossary

N/A

Primary Flow of Events

Trigger

User clicks “Create letter” button from main window

Steps	Action	System Response
1	User clicks “Create letter” button	Display Details page to input Student’s data
2	Enters all required information for a new recommended person and click “Compile” button	Shows the compiled letter in a new window
3	User clicks “Save” button	Saves the Edited and Draft version of letter in “letters” directory

Alternate Flow of Events

Alternate Trigger

Invalid/Missing Data

Steps	Action	System Response
1	Doesn’t input all necessary information like first name	Asks user to make sure they have correctly inputted all necessary information
2	User fills out missing/invalid information	New recommendation is accepted

Use Case Name	UC-05: Search Old Recommendations

Goal

A description for how users can search their saved recommendations.

Participating Actors

User, System

Glossary

N/A

Primary Flow of Events

Trigger

User clicks “Search” input field from the home screen

Steps	Action	System Response
1	Enters student's last name that the user wants to search	Shows the matching letters in a List format
2	Selects the recommendation letter that the user was searching for	Enables the “Edit” and “Delete” option for user to make changes

Alternate Flow of Events

Alternate Trigger

When no letter matching the keywords is found

Steps	Action	System Response
1	Enters a letter name that doesn’t match any saved letters	Shows “No Record !” message in the search area
2	Re-enters the name to search again	Shows the matching result

<table>
<tr><td><b>Use Case Name</b></td><td colspan="2">UC-06: Edit Recommendation</td></tr>
<tr><td colspan="3"><b>Goal</b></td></tr>
<tr><td colspan="3">A description for how a user can edit a completed recommendation</td></tr>
<tr><td colspan="3"><b>Participating Actors</b></td></tr>
<tr><td colspan="3">User, System</td></tr>
<tr><td colspan="3"><b>Glossary</b></td></tr>
<tr><td colspan="3">N/A</td></tr>
<tr><td colspan="3"><b>Primary Flow of Events</b></td></tr>
<tr><td colspan="3"><b>Trigger</b></td></tr>
<tr><td colspan="3">User searches for the recommendation letter to edit</td></tr>
<tr><td><b>Steps</b></td><td><b>Action</b></td><td><b>System Response</b></td></tr>
<tr><td>1</td><td>User clicks on recommendation they wish to edit</td><td>Enables the edit option for the user</td></tr>
<tr><td>2</td><td>Selects the “Edit” button to edit the selected letter</td><td>Allows user to now edit recommendation on the details window</td></tr>
<tr><td>3</td><td>User compiles the newly edited recommendation</td><td>Shows newly drafted version, asks user if they wish to save</td></tr>
<tr><td>4</td><td>Presses save</td><td>New recommendation overwrites the old file, instead of duplicating the same file</td></tr>
<tr><td>5</td><td></td><td></td></tr>
<tr><td colspan="3"><b>Alternate Flow of Events</b></td></tr>
<tr><td colspan="3"><b>Alternate Trigger</b></td></tr>
<tr><td colspan="3">Invalid/Missing Data</td></tr>
<tr><td><b>Steps</b></td><td><b>Action</b></td><td><b>System Response</b></td></tr>
<tr><td>1</td><td>Doesn’t input all necessary information like first name</td><td>Asks user to make sure they have correctly inputted all necessary information</td></tr>
<tr><td>2</td><td>User fills out missing/invalid information</td><td>Recommendation changes are accepted</td></tr>
<tr><td><i>3</i></td><td></td><td></td></tr>
<tr><td><i>4</i></td><td></td><td></td></tr>
</table>

Use Case Name	UC-07: Password Reset

Goal

A description for how a user can reset their password

Participating Actors

User, System

Glossary

N/A

Primary Flow of Events

Trigger

When user clicks “Edit password” button in the main window

Steps	Action	System Response
1	Clicks “Edit password” button	Shows the password reset window
2	Enter the default and new password and then click on “Confirm new password” button	Verifies and stores the newly created password in the database and takes user back to the login window
3		

Alternate Flow of Events

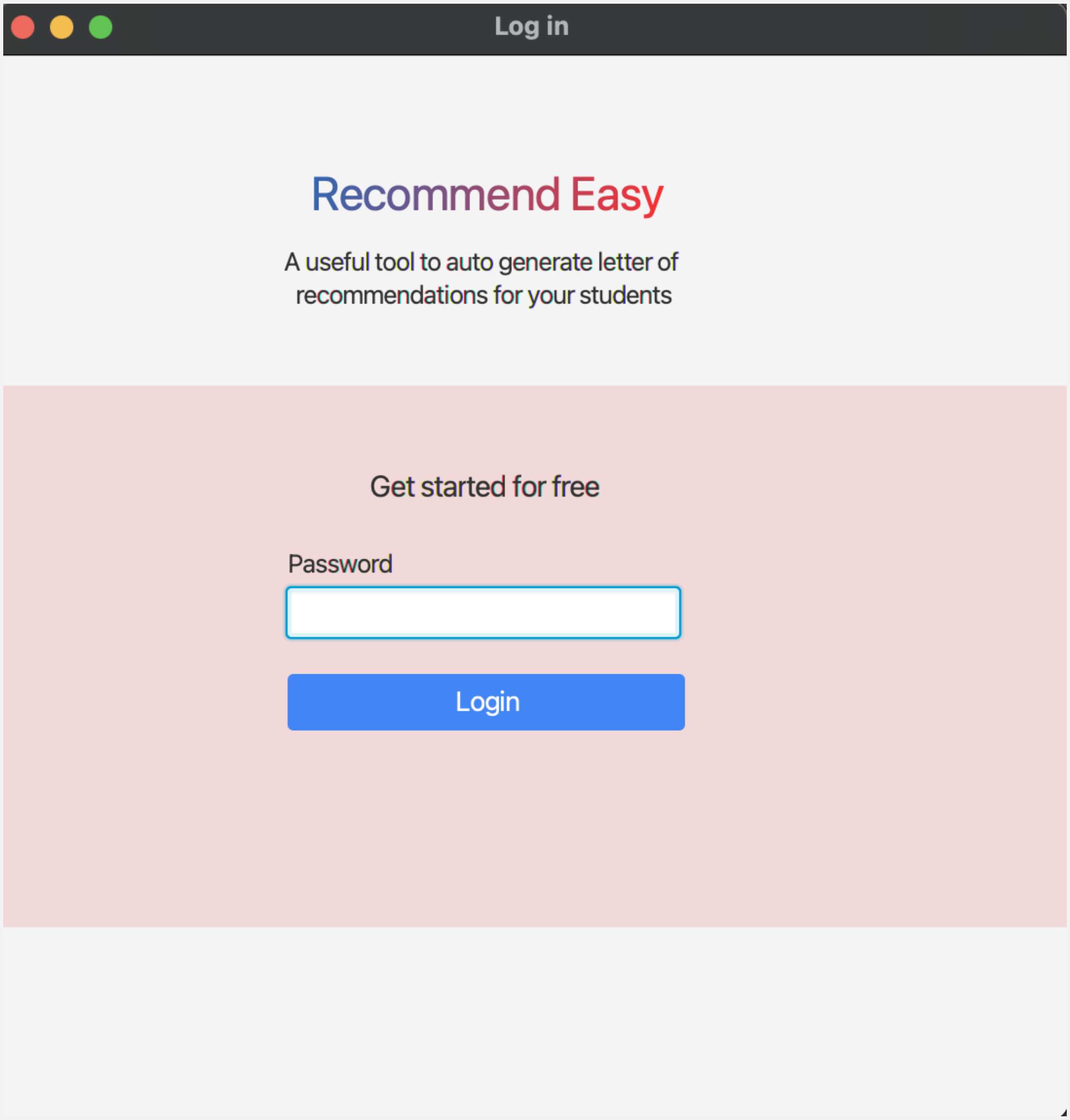
Alternate Trigger

When a user enters invalid default password to reset

Steps	Action	System Response
1	Enters wrong input for default password	Highlights the error and asks the user to type the correct default password
2	User re-enters both password correctly and click the “Confirm new password” button	Accepts both passwords and allows the user to log-in with the new password
3		

4 Mockups

1. Login window

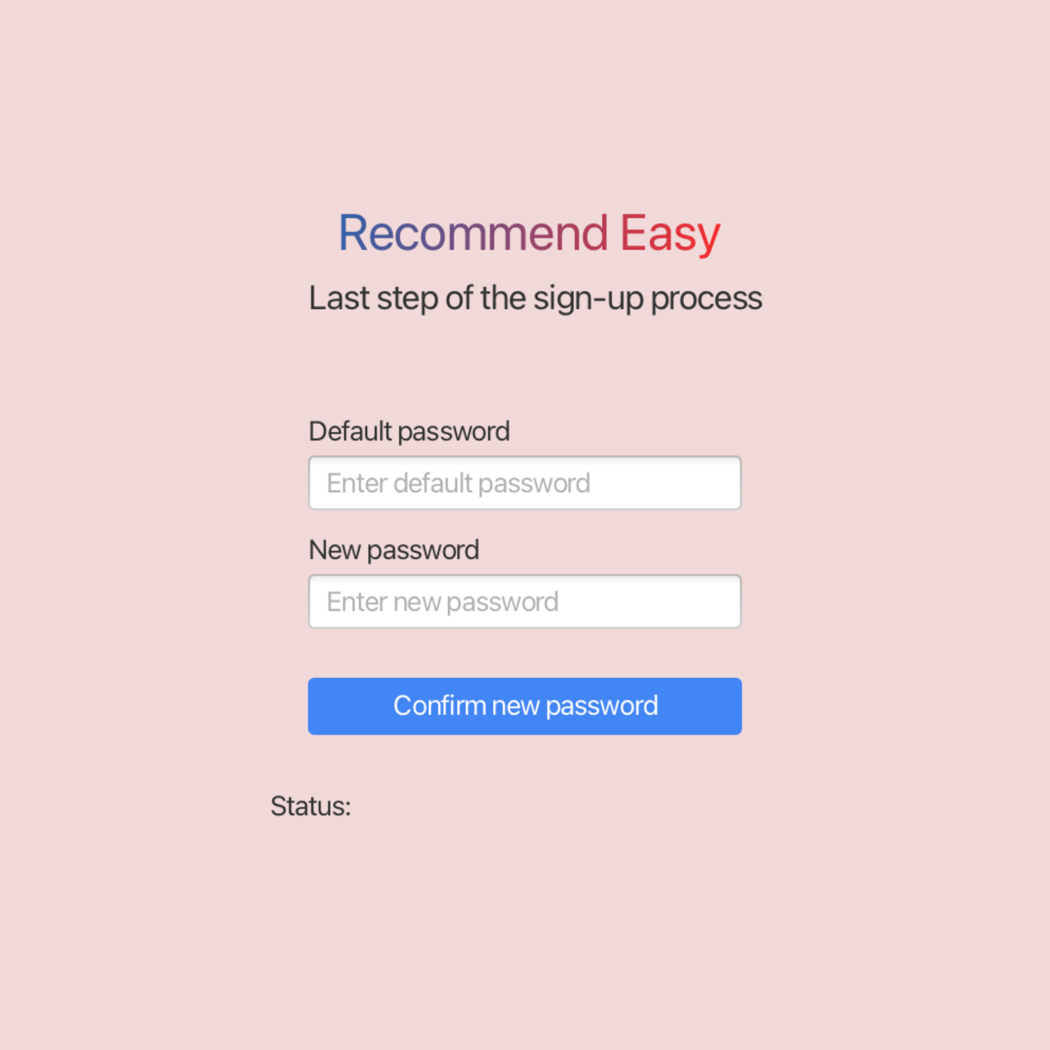


Recommend Easy

A useful tool to auto generate letter of recommendations for your students

Get started for free

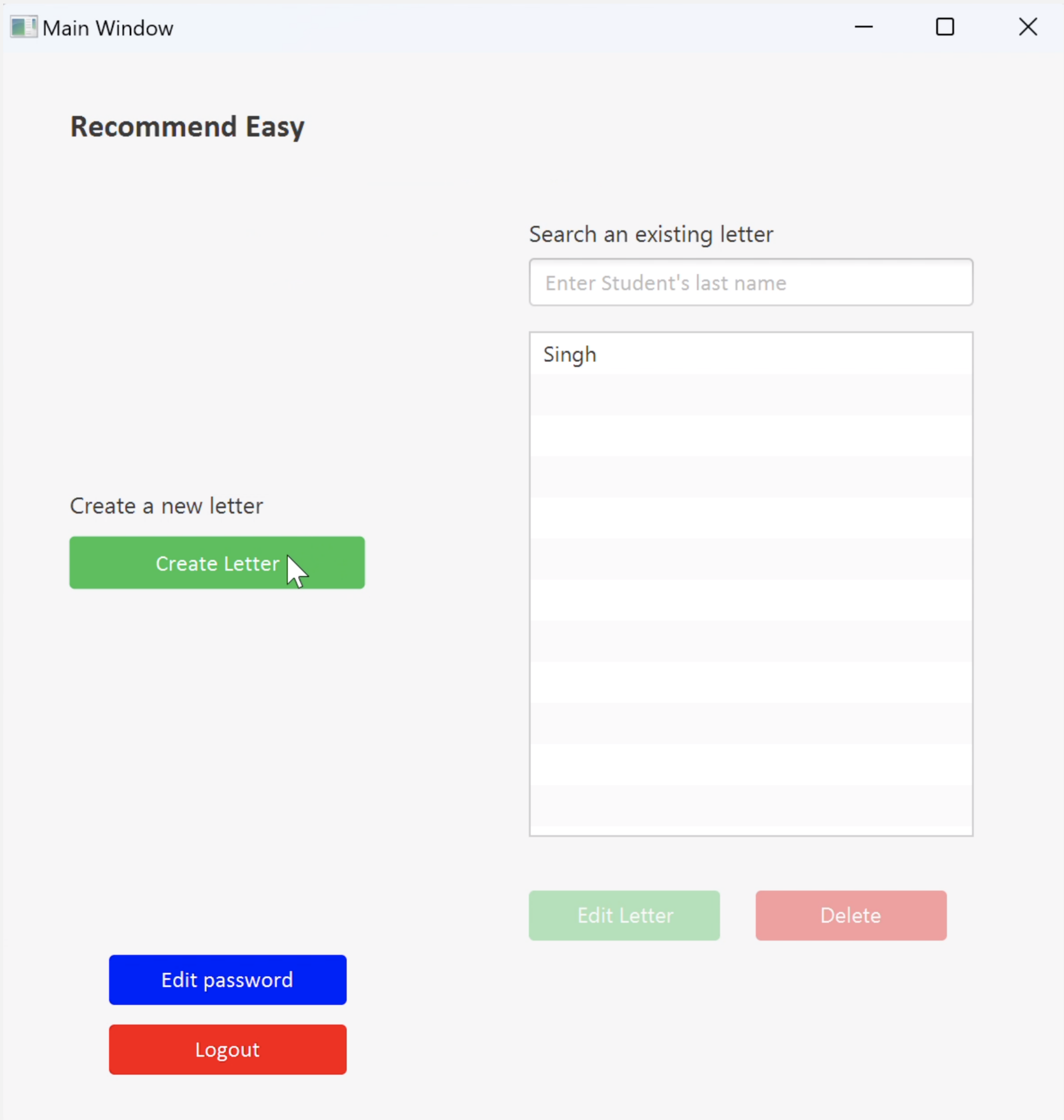
2. Password Reset window



Recommend Easy

Last step of the sign-up process

3. Home Page window


Main Window
Recommend Easy
Search an existing letter
Enter Student's last name
Singh
Create a new letter
Create Letter
Edit Letter
Delete
Edit password
Logout