
Software Requirements Specification

for

Notelee

Version 2.0 approved

Prepared by Deja Hintzen, Josh Warren, Christian King

Kennesaw State University

Fall 2023

12/3/2023

Table of Contents

1. Introduction	1
a. Purpose	1
b. Intended Audience and Reading Suggestions	1
c. Product Scope	1
d. References	1
2. Overall Description	1
a. Product Perspective	1
b. Product Functions	2
c. User Classes and Characteristics	2
d. Operating Environment	2
e. User Documentation	2
3. External Interface Requirements	2
a. User Interfaces	2
b. Hardware Interfaces	2
c. Software Interfaces	2
d. Communications Interfaces	3
4. System Features	3
a. Automatic Login	3
4.2 Login	3
4.3 Register	3
4.4 Send Password Reset	4
4.5 Delete Account	4
4.6 Log Off	5
4.7 View Folders	5
4.8 Note Creation	5
4.9 Note Deletion	6
4.10 Note Editing	6
4.11 Show Tutorial	6
5. Other Nonfunctional Requirements	7
a. Performance Requirements	7
b. Safety Requirements	7
c. Security Requirements	7
d. Software Quality Attributes	7
Appendix A: Glossary	7
Appendix C: To Be Determined List	8

Name	Date	Reason For Changes	Version
Christian King	9/10/23	Team Meeting	1.0
Deja Hintzen	9/16/23	Document Revision	1.0
Deja Hintzen	9/24/23	Document Revision	1.0
Christian King	12/3/23	Document Revision	2.0

Revision History

1. Introduction

1.1. Purpose

The purpose of this SRS report is to define the software requirements for our Google Chrome plug-in, Notelee, version 2.0. This document will provide the foundation for the architecture of the software as well as the design. This document will also be used by our team to track the progress of our project and report changes made.

1.2. Intended Audience and Reading Suggestions

The intended audience for this report is the developers, Christian King, Deja Hintzen, and Josh Warren, and the professor, Sharon Perry. This SRS will contain the description of the intended product, the functionality of Notelee, and future plans

1.3. Product Scope

Notelee will be a Google Chrome plugin with note taking capabilities through the Google Chrome browser. The plugin will be used for bookmarking documents and webpages where the user will be able to add notes. It can also be used for documentation for research or processes for a project or job.

1.4. References

“Chrome Extensions Getting Started Guides.” *Chrome for Developers*,

developer.chrome.com/docs/extensions/mv3/getstarted/. Accessed 24 Sept. 2023.

Rebelo, Miguel. *Evernote vs. Google Keep: Which Should You Use?*, Zapier, 15 Sept. 2022,

zapier.com/blog/google-keep-vs-evernote/. Accessed 24 Sept. 2023.

2. Overall Description

2.1. Product Perspective

Notelee will be a new, self-contained product for note taking in Chrome. It will be designed to effortlessly take notes using the web browser. Notelee will easily be able to work in the Google Chrome Browser for efficient note taking.

2.2.Product Functions

Notelee's functions will be:

- Allow users to create and save notes through the Google Chrome browser
- Notes can be titled
- Folders can be created and notes can be saved into those folders (can be multiple folders)
- Multiple notes can be saved in a folder
- Notes can be made and also archived
- Notelee will have functional buttons to create these folders or notes or just change pages within the plugin

2.3.User Classes and Characteristics

Anyone such as students, teachers, and researchers will be able to use Notelee to study, take notes or keep track of research. Anyone who needs to take notes will be able to use this product

2.4.Operating Environment

Notelee will operate on an updated Google Chrome browser and will use Javascript, HTML, CSS, and firebase.

3.External Interface Requirements

3.1.User Interfaces

Notelee will take up a small amount of the page that the user is on when opened. The user will be able to see the folder they've created as well as the notes within each of the folders. Each note created will have text boxes and input boxes.

3.2.Hardware Interfaces

Using a manifest.json file, the plugin should only be used by Google Chrome and interact with our database.

3.3.Software Interfaces

Firebase and chrome storage: Firebase realtime database is implemented to save data for registered users, such as their id, email, folders, and notes. Chrome storage must save the user as active or inactive for the auto login feature.

3.4.Communications Interfaces

Notelee should have quick communication with firebase using the firebase APIs to read and write data.

4.System Features

4.1Automatic Login

4.1.1 Description and Priority

This will be a high priority feature that will automatically login the last logged in user by searching for the “active” user in chrome storage.

4.1.2 Stimulus/Response Sequences

When opening Notelee, they will skip the popup.html screen to show a loading screen before their folder notes are shown.

4.1.3 Functional Requirements

1. Log in to Notelee then close it.
2. When opened again, show a loading screen before showing the notes for folder 1.

4.2 Login

4.2.1 Description and Priority

High priority feature. Allows users to login with their credentials.

4.2.3 Stimulus/Response Sequence

On popup.html, click login, enter in an email and password. If it matches an authenticated user, load notes for folder 1.

4.2.4 Functional Requirements

1. Alerts for errors.
2. If the email and password is in the system and their email is verified, take them to view their folders.
3. Make their status in chrome storage as “active.”

4.3 Register

4.3.1 Description and Priority

High priority feature. Allows users to register an account.

4.3.3 Stimulus/Response Sequence

On popup.html, click register, enter an email and password, click register. If credentials are correctly formatted, send an email verification and reload the webpage.

4.3.4 Functional Requirements

1. Alerts for errors and formatting issues.
2. Sends email verification when the register button is clicked, along with an alert on the screen.
3. Reloads Notelee.

4.4 Send Password Reset

4.4.1 Description and Priority

High priority feature. Allows users to change their password if they forgot it.

4.4.3 Stimulus/Response Sequence

On popup.html, click login, click forgot password button. When clicked, allows users to input their email and send a password reset email.

4.4.4 Functional Requirements

1. Alerts for errors.
2. Sends password reset email to the email entered in the input box.
3. Alerts that the email has been sent.
4. Reloads Notelee.

4.5 Delete Account

4.5.1 Description and Priority

Low priority. Allows users to delete their account and all data.

4.5.3 Stimulus/Response Sequence

Once logged in, go to the manage account menu and click delete account. Shows two options: yes and no. When yes is clicked, all of their information is removed from firebase and chrome storage. When no is clicked, go back to viewing notes.

4.5.4 Functional Requirements

1. Clicking yes alerts the account has been successfully deleted.
2. Removes the account from the list of authenticated users.
3. Removes the user's reference in the realtime database, removing all of their folders and notes.
4. Removes their user data from chrome storage.
5. Reloads Notelee.

4.6 Log Off

4.6.1 Description and Priority

High priority feature. Allows users to sign off of Notelee.

4.6.3 Stimulus/Response Sequence

Once logged in, go to the account management menu and click sign off. Alerts that they have logged out and Notelee is reloaded.

4.6.4 Functional Requirements

1. Alerts that the user has signed off.
2. Reloads Notelee.
3. Sets their status as “inactive” in chrome storage.

4.7 View Folders

4.7.1 Description and Priority

High priority feature. Allows users to view any and all notes under their folders.

4.7.3 Stimulus/Response Sequence

Once logged in, they will see a “No Notes” text or a list of their notes on the display. Both will have a button to create a note, and clicking a different folder button changes the display.

4.7.4 Functional Requirements

1. Clicking a folder button makes it background pink while the rest are blue.
2. Hovering over the buttons changes the border color.
3. Clicking a folder with no notes shows a no notes display onto the screen with a create note button.
4. Clicking a folder with notes shows a colorful note list table with note buttons that are the title of the note, and delete buttons for each note.

4.8 Note Creation

4.8.1 Description and Priority

This will be a high priority feature which will allow users to create notes quickly and efficiently

4.8.2 Stimulus/Response Sequences

When the user wants to create a new note, they would click create note. A note page will pop up which will allow them to create a note of any length by inputting a title and any text into the text area.

4.8.3 Functional Requirements

1. Clicking “Save Note” checks if the title input field is empty.
 - a. If it is, alert the user.
 - b. If it is not, continue.

2. The note is pushed into the user's folder in firebase.
3. The user is sent to view their folders again in Notelee.

4.9 Note Deletion

4.9.1 Description and Priority

This will be a high priority feature which will allow users to delete notes.

4.9.2 Stimulus/Response Sequences

To delete a note, press the "x" button to the right of the note in the note list. This should take the user to a delete confirmation screen where the user can press yes or no.

4.9.3 Functional Requirements

1. Clicking "no" returns the user to the folder.
2. Clicking "yes" removes the note from the user's folder in firebase, and the user is taken back to their folder to view the updated list.

4.10 Note Editing

4.10.1 Description and Priority

This will be a high priority feature which will allow users to edit already created notes.

4.10.2 Stimulus/Response Sequences

To edit a note, press the note's button on the note list. This will take the user to the note creation screen with the note's text already inputted into the text areas.

4.10.3 Functional Requirements

1. Clicking "Save Note" replaces the content of the note in firebase with the new note, keeping the same note id.
2. The user is taken back to viewing their folders.

4.11 Show Tutorial

4.11.1 Description and Priority

Low priority feature. Will show a tutorial popup when clicking the question mark icon.

4.11.3 Stimulus/Response Sequence

After logging in, clicking the question mark icon opens a popup window with a tutorial webpage.

4.11.4 Functional Requirements

1. Hovering over the question mark button changes the opacity.

2. Clicking the button opens a closable popup window that displays a webpage for the Notelee tutorial.

5.Other Nonfunctional Requirements

5.1.Performance Requirements

The performance of Notelee requires a network connection for communications between Notelee and firebase. A slower network communication will result in a longer time for loading notes for folders.

5.2.Safety Requirements

Everything that the user enters into Notelee will be secure. No other website will be able to access any information in any of the notes created by the user,

5.3.Security Requirements

Users will only be able to be logged into Notelee through the Google Chrome browser. Other webpages will not be able to see any notes made by the user. Notelee does not keep track of information entered on other pages. For example, if the user enters their debit card information onto Amazon, Notelee will not be able to read this information and store it, and users can only read and write data to their own account.

5.4.Software Quality Attributes

The following are software quality attributes that will be present in Notelee:

- **Functionality:** Users will be able to create, edit, and store notes, manage their account settings
- **Customization:** Colorful appearance in the note lists and buttons throughout Notelee
- **Availability:** Will be available to user anytime as long as Chrome is open
- **Reusability:** Users can constantly reuse/open close the plugin without errors or info being deleted

Appendix A: Glossary

- **Annotation:** Highlighting and commenting on a webpage.
- **Customization:** Ability to personalize notes and folders
- **End to End Encryption:** Ensures that your data is secure and encrypted when notes are stored

- Software Requirements Specification: Document provided to outline the functional and nonfunctional requirements of a software product.
- User Interface: The elements that allow the user to interact with the software