

Statement of Work

-Zach Pulichino, Nicholas Gillespie, Jason King, Silas Nevstad

Project Name: Husksheets

Project Duration

May 13, 2024 - June 15, 2024

Project Description

Husksheets is designed to be a user-friendly, robust, collaborative spreadsheet application. It consists of a server with a persistent store, a client able to create and open spreadsheets, and a user interface that displays sheets and allows editing them.

Project Objectives

- Develop a server supporting REST API requests for managing spreadsheets
 - register, createSheet, getSheets, deleteSheet, getUpdatesForSubscription, getUpdatesForPublished, updatePublished, updateSubscription
- Develop a client application that can create, open and interact with spreadsheets
- Develop a UI allowing users to see, edit and collaborate on spreadsheets
- Implement authentication and authorization

MVP

Broad Goals

- Server that can handle basic requests for managing spreadsheets
- Simple client that can create and open spreadsheets (interacting with the server)
- Basic UI that can display spreadsheets and allow basic editing (numeric/textual data, basic formulas, etc)
- Authentication and data persistence

User Stories

- User is able to authenticate their identity to access sheet
 - Acceptance Criteria
 - User can enter a password and gain access to editing the sheet
- User is able to save and access sheet at a later date
 - Acceptance Criteria
 - User can click a save button at any time during editing and the sheet will save to a file
 - User can load and edit an existing sheet from a previously saved file
- User is able to enter data into sheet and have it appear in specified cell
 - Acceptance Criteria
 - Data entered by user is displayed properly in cell it is entered in
- User is able to enter a formula and have the result displayed in cell

- Acceptance Criteria
 - Correct result of formula is displayed
 - Error is displayed if formula is unable to be evaluated
- User is able to decide on application startup whether to create a new sheet or edit an existing one
 - Acceptance Criteria
 - User can create new spreadsheet from home page
 - User can choose to edit existing spreadsheet from home page

Desirable Features

As a user I want to collaboratively edit a sheet with other users in real time, so that changes are instantly visible and can be jointly managed.

Acceptance Criteria(conditions of satisfaction):

- **Real-Time Updates:** Changes made by one user should be visible to all other users in real time without needed to refresh the page
- **Concurrent Editing:** Multiple users should be able to edit the sheet at the same time without data being overwritten unintentionally.
- **User Presence:** The application should display which users are currently active on the sheet and where they are editing
- **Change History:** Users should be able to view a history of changes made to the sheet, including who made each change and when

As a user, I want to apply various formatting options to cells(like font size, color, and style), so that I can emphasize certain data and make the sheet easier to read.

Acceptance Criteria(conditions of satisfaction):

- **Formatting Options:** Users should be able to select cells and apply formatting such as font size, font style, text color, and background color.
- **Multiple Selection:** Users should be able to select multiple cells at once to apply formatting collectively.
- **Preview:** Changes should be previewed in real time as they are applied to ensure accuracy

As a user, I want to merge multiple cells, so that I can group data logically and improve the visual layout of my spreadsheets.

Acceptance Criteria(conditions of satisfaction):

- **Merging Functionality:** Users should be able to select two or more adjacent cells and merge them into one cell.
- **Unmerge Option:** Users should also have the option to unmerge previously merged cells.
- **Content Alignment:** The content of the merged cell should be aligned according to the users preference (e.g., left, center, right).

As a user, I want to use complex formulas in cells, so that I can perform advanced data analysis and calculations directly within the spreadsheet.

Acceptance Criteria(conditions of satisfaction):

- **Formula Support:** The application should support a wide range of mathematical and logical formulas.
 - **Autocomplete and Suggestions:** As users type formulas, the systems should provide autocomplete suggestions and syntax help.
 - **Error Handling:** The system should provide clear error messages if a formula is incorrect or cannot be executed
- User-friendly, polished UI

As a user new to using spreadsheets to manage task, I want the U.I. to be user friendly, polished, and easy to understand.

Acceptance Criteria :

- **Modern GUI :** The graphical user interface we use to build the U.I. should be modern, polished and well supported so as to make the best U.I.

Additional Features

- User Story : As an event planner, I want to be able to easily import and export .csv files related to the current event I'm planning so I can easily share and incorporate new data.
 - Acceptance Criteria : Export and import for external files
- User Story : As a user, I want the ability to store sheets in the cloud, as well as be ensured that my sheets are secure and need some form of authentication to be accessed.
 - Acceptance Criteria : Third-party services (cloud storage, authentication etc)