Sinkd

Cloud storage solution for the ever moving student

CIS444 Group A
David Bright, Brandon Tilley, Sam Doroudi
Crystele Dierickx, Chris Banci, Andrew Zhang

October 22, 2017

Annotations: *Italicized*

3rd party technologies used:

jsTree → https://www.jstree.com

Bootstrap → http://getbootstrap.com

1) Describe the real life problem to be solved with a Web server.

- The real life problem to be solved is access to storage when you are on the go whenever and wherever. Instead of storing files locally on a computer, they can be stored on a cloud to be accessed anywhere. *The reason why ours is better than the other companies that have similar concepts is because many people can access files anywhere while you are on the go and it's free to cater to the students at CSUSM.*

2) Identify all the potential users of such Web site (ideally, you would have already interviewed some potential users at this stage).

- The potential users for our website are CSUSM students, faculty and staff who require cloud storage for ease of printing/presenting and accessible from all devices.

3) Illustrate how each potential user is going to interact with your Web site.

- The student will login on a secure (https) website by entering their credentials to access their cloud files. *Once the user accesses their cloud files and selects them to upload into their desired folder or adjust them as they please, they then can proceed to access them from any other device when using our webpage.*

4) Include how many Web pages are potentially needed and what each page is for, but not the actual implemented pages, and their wireframes.

- For our project, only *three* pages will potentially be required:

  1) *Create Account*

  - *This page will be used to create an account.*

2) Login page

- This page is used to login to the website securely along *with the functionally to be redirected to create a new account if the user doesn't have an account*.

3) Main page

- This page is used to display the user's files in a tree view. It will refresh dynamically upon upload and deletion of files. We value user experience, and a way to express this is with using a single page. Users would rather have their files at their fingertips, rather than different pages.

5) Indicate what users are expected to enter and receive for each page,

- 1) *Create Account*
  - *To create an account the user is expected to enter a email, username, and password and receive access their account and ready to upload files.*
- 2) Login page
  - To login, the user is expected to enter their username and password. If credentials match, the user will receive access to their files.
- 3) Main Page
  - The user is expected to upload, delete and view files on this page. The user will also be able to create folders and share their files with others.

6) Draw a diagram indicating the interactions among the Web pages.

Create Account:

This page interacts with the main page after making an account

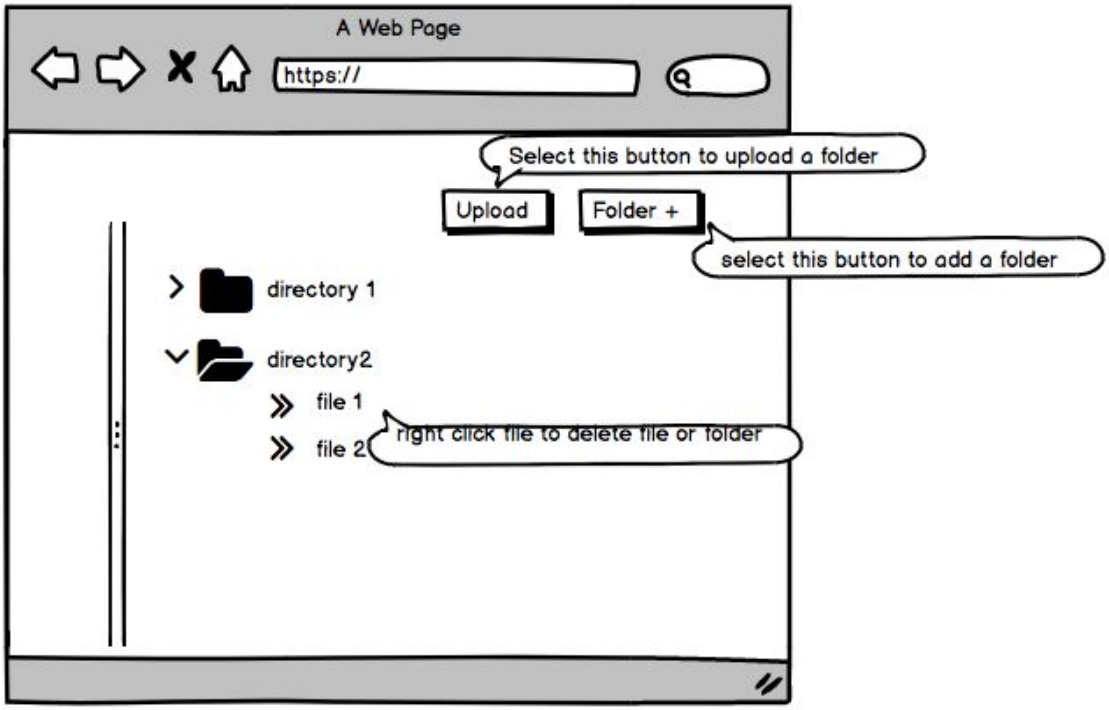
Login Page:

Login Page: This page interacts with the main page and the create account page if the user has not made an account, depending on what button the user selects.

Main Page:

Main Page: This page interacts with all the pages such as the Login page after the user has successfully login , the Create Account page when the user has made an account they are directed to the main page, and when creating , deleting, uploading a file.

Delete, Add folder, Add/Upload file:



<Page function here :)>