HonorsBook

A Digital Phonebook for Student Connection

(Maintenance Manual)

By

Colby Holloman

B.S. - Computer Science Degree

Anticipated May 22, 2021

Mentors:

Dr. David Marshall

Director, University Honors Program

Dr. Ernesto Gomez

Professor, Computer Science Department

A PROJECT PRESENTED TO THE FACULTY OF

CALIFORNIA STATE UNIVERSITY SAN BERNARDINO

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

THE UNIVERSITY HONORS PROGRAM

Abstract

High-achieving students like those in the University Honors Program often have trouble making connections with their peers. This social disconnect stands against the program's goal of creating a welcoming academic cohort. To address this issue, the idea of Honorsbook, a web app created for honors students, was posed. The application aims to create an environment where UHP students can be connected with peers of similar interests. A team was assembled to accomplish the separate aspects of imagery and design, and of programming and data management. The design philosophy aims for a layout that is intuitive, and creates a feeling similar to being in a library or coffee shop. Using CSUSB colors and design as a basis, further decisions behind color palettes and typographic elements were inspired by the designs used in competing social media websites and other forms of digital media. For the development of the application, HTML/Javascript are the primary languages, allowing for straightforward implementation of the design team's ideas. Additional research was required to accomplish data storage with Google Firebase, as well as the development of key features such as user profiles, and a searchable tag system. In addition to the approximation of real-world job experience for the team, the app will have a significant impact for UHP students once deployed, and well into the future. With a tailored platform for high-achieving students, it is hoped that the social disconnect experienced by many of these students can be greatly reduced.

Acknowledgements

This project would not have been possible without the original inception and continued guidance from the Honors Program Director, Dr. Marshall. Dr. Marshall offered myself and the other members of the HonorsBook development team the opportunity to experience the real-world process of a multidisciplinary project centered around the development of a computer program, which will be invaluable in our future career paths.

I would also like to thank the other members of the development team this year, who made the completion of the project possible:

Samuel Jacuinde

Daniel Martinez

Celeste Soto

Regan Reynolds

Angelica Gutierrez

Logan Ashbaugh

Lastly, I would like to thank the faculty and staff of the Honors Program as well as the Computer Science Department, for creating supportive and inclusive environments that provided me with everything I needed to contribute to this project.

HonorsBook Maintenance Manual

Version 1.1

Colby Holloman - Project Manager & Software Engineer

Samuel Jacuinde - Project Manager & Software Engineer

Daniel Martinez - Project Manager & Software Engineer

Celeste Soto - Graphics Designer

Regan Reynolds - Graphics Designer

Angelica Gutierrez - Web Designer

Logan Ashbaugh - Web Designer

1.) File Structure

The project's file structure is handled and organized on our github repository, saved under the root directory. Because we are designing our source code to be as re-usable as possible, we organize our files by keeping the most closely related files together. Sign-up related code is kept in the sign-up folder, menu related in menus folder, and home page/primary code saved directly in the root directory. The parent directory and all subdirectories are accessible to all team members, where each member can upload, download, and view files. Each change/push onto the github includes a brief description of what was changed.

style.css

This primary style sheet for every page of the webpage. It's the basis for text and background color, flexible stylines and bordered objects, and more. This should be utilized by every page as the starting basis, and anything more specific than the template style should be formed in its own .css file.

index.html

Is the first page that any non-logged-in user will be sent to. This page will allow the user to log-in if they have an existing account, or click on a sign-up button if they need to create an account.

login-form.css

The css file for the index page, defining the lay-out for the login form.

login-page-script.js

This file handles the logging-in functions of index.html. It will take in the email/password pair the user inputs and logs in the user if the credentials exist. The user will then be redirected to home.html.

signup-form.html

This page is for users to create their account on the website. Here the user will type in their username or "displayname", email, password, etc. When they complete and submit the sign-up form, they will be redirected to the main home-page.

signup-form.css

This is the formatting file for the sign-up form page.

sign-up-script.js

This is the script that runs signup-form.html. When the user clicks on the "submit" button, it will take in the entered information, and create a user authentication profile and database entry for the user.

home.html

This is the home-page of the HonorsBook platform. Here the user is able to see any recent notifications, view messages from the University Honors Program, and see some daily recommendations for users similar to them.

homeStyle.css

This is the styling page for the home page, defining the layout, colors, etc.

navstyle.css

This file is used to format the primary navigation bar on the top of every web page. It defines its color, shape, positioning of the buttons, and more.

user-check-script.js

This script runs every time the user loads up one of the main pages of the project. If there is a user logged in, it will load up any of the necessary user-specific information. If there is no user logged in, it will redirect the user to the login-page.

sign-out-script.js

This script runs the sign-out feature of the website. The user will click on an assigned "sign-out" button, and when the user is signed out, the *user-check-script.js* will automatically send the user back to the login page.

profilePage.html

This page represents the general profile page for every user on the site. It allows users to view other user's names, profile pictures, major, class year, biography, and more.

profilePageStyle.css

The primary styling file for profilePage.html

profile.js

This script will load the logged-in user's information onto their profile page.

profile-page-edit-script.js

This script runs profile page features, such as the ability to modify the user's bio. It interacts directly with the database, and as such assumes and requires that user-check-script is running in order to function correctly. On the click of a "submitButton" element, the database is updated with the information entered into html text fields on the profile page.

interests.html
This page is where users can search for and add tags to their account. Upon hitting the search button, the system will look for any tags that contain the user's search criteria. On clicking a tag, it will add that tag to the users' account.

interestsStyle.html
The formatting file for interests.html

interests-script.js
The functional file for interests.html. Upon loading the page the system loads a copy of all of the site tags and displays the first 16 tags onto the page. There are several in-code defined functions which aid in the creation of the tags and their ability to attach themselves to users. Further descriptions in the code comments themselves.

2.) Instructions

The application can be run using one of the following options:

1. Using the URL provided by the school (recommended).
2. Emulating the website through use of Microsoft Visual Studio Code.

To run the application through any of the above methods, one must

1. Use a laptop or desktop computer that is connected to the internet.
2. Use Google Chrome to access the application.

While we have no admin permissions yet, we have a default admin account with the following login information:

- Email: admin@admin.com
- Password: 123456

Method 1: To run the application through the URL generated and hosted by Firebase:

1. Use a laptop or desktop computer that is connected to the internet.
2. Use either Google Chrome, Firefox, or Microsoft Edge.
3. Use the following URL:
   http://honorsbook.academic.csusb.edu/
4. Press enter.

Method 2:

1. Download all source code from the main branch of our github and extract into a root folder: https://github.com/SJacu/HonorsBoook. Access to the files and the repository may be requested by visiting the link or emailing 005832082@coyote.csusb.edu.

2. Install Visual Studio Code
    a. Go to https://code.visualstudio.com/
    b. Click on the drop-down arrow next to “Download for Windows” and select the installer for your machine.
    c. Run the installer and follow the installation instructions to install Visual Studio Code.
3. Open Visual Studio Code in the root folder and Install the Live Server addon
    a. Click on the extensions button on the left-hand side of Visual Studio Code
    b. In the search bar where it says “Search Extensions in Marketplace”, search up “Live Server”
    c. Select the “Live Server” addon by Ritwick Dey and install onto Visual Studio Code
4. Open the index.html file in Visual Studio Code, and right click on the main html code, and click on “Open with live server”

3.) Good Implementation

The following are what is positively implemented with the current design and are things that should be considered when building off of the current build:

- Not too many files are necessary to run the site in the current version.
- Pleasant Design and appearance, with engaging visuals and fonts
- Website navigation is intuitive and direct.
- A solid-working proof of concept.

4.) Needs Improvements

The following are things on our project that need improvement, or a complete reworking:

- Error and downtime mitigation
- User account profile customization
- Create code so that way users can not modify javascript of the website.

5.) Overall Recommendations

Should another team decide to continue this project in the future, the following suggestions may be used as a guide:

- Source code cleanup and continued modulation.
- Mobile friendly
- Move the website functionality back onto school hosting.
- Adjust tag system so it does not apply to users who already have that tag.