Project Instructions

Project 7: WebApp Dashboard

Before you start:

1.) **As with the previous projects, you'll submit your finished working using GitHub. If you need a reminder on how to use GitHub and GitHub desktop to create a new repository check out this workshop: Share Your Techdegree Projects With GitHub Desktop**

2.) **Download the project files. We've given you the following files:**
   - dashboard.png (a mockup of the finished web application page)
   - icon files (8 svg images):
       - 4 navigation buttons on the left side of the page
       - "Bell" icon used to identify alerts
       - Twitter, Facebook and Google+ icons

3.) **Find a chart library. We recommend chart.js. When adding your chart library, please use the CDN links, DO NOT install via node/bower for this project. Use the link for the chart.js library here:** https://cdnjs.com/libraries/Chart.js

4.) **As with previous Treehouse projects, this project needs to be coded with a mobile-first approach. Your base CSS code will be for the mobile view, then add 768px and 1024px min-width media queries for the larger screen sizes to match the mockup.**

5.) **Look at the Project 7 study guide**: https://s3.amazonaws.com/treehouse-techdegree/fewd/study-guides/project-7-study-guide.pdf

Instructions:

1.) Create a responsive web page from the supplied dashboard.png mockup with the following:
   - Header with app name, notification icon badge, and profile avatar and name.
   - SVG icon based navigation with the following links: Dashboard, Members, Visits, and Settings. NOTE: You only have to build out the main dashboard page, not any of the other pages.
   - Main content area where the specific dashboard widgets will go.
   - Ensure that the design responds well to mobile (320px), tablet (768px) and desktop (1024px) screen sizes.

- Use CSS grid to lay out the main elements (header, sidebar navigation, main content) on the page.

2.) Demo alert notification.
- In the dashboard.png, this appears as the purple bar near the top of the page with the word "Alert" in it. When the page loads this alert should be visible, but the user should be able to close the alert by clicking the X button.
- Include a notification icon in the navigation header. Use the icon-bell.svg file. In the mockup, this is the bell in the top right corner of the page.
- Add a CSS transition or animation to the bell icon when the user hovers over it.

3.) Chart widgets
- Using chart.js, create and include the information for the following chart widgets, as shown in the mockup for the:
- Web Traffic (line chart)
- Daily Traffic Bar Chart (bar chart)
- Mobile User Pie Chart (donut chart)
- Style the charts to match the overall style of the dashboard.
- You will need to make up this data -- you can see the mockup for sample data.

4.) Social Stats Widget
- Create a widget (or three separate widgets) to display social network stats for Facebook, Twitter, and Google+.
- Use the provided SVG icons for each of the social networks.
- SVG icons are added as inline SVG's.
- SVG fill colors have been changed to match the mockups.
- Style the social information to match the corresponding social network.
- Style to match the overall look and feel of the dashboard.

5.) New Members and Recent Activity Widgets
- Create widgets that list users for both widgets.
- Include avatars for each member (member avatars are inside images folder).
- Add the information for each user as shown in the mockup, such as Member name, email address, Sign up Date etc.

6.) Message User Widget
- Create a field for searching for a user.
- You don't have to add real search functionality, but if you attempt to get the exceeds grade, you'll need to make up some user data.

- Add a message textarea field that lets you add a message.
- Create a "Send" button and use JS to allow you to submit the form and display a confirmation the message was sent. You won't actually submit the form, just simulate the action using JavaScript.
- Use JS to display error messages if a user isn't selected or message field is empty.
- Style to match the overall look and feel of the dashboard.

7.) Settings Widget
- Create a settings widget to display various setting options using different form elements.
- Create an on/off widget for whether to send email notifications.
- Create an on/off widget for whether to set the profile to public or private.
- Create a drop-down to select the timezone.
- Add Save and Cancel buttons (these do not have to do anything functional, unless going for the Exceeds Expectations localStorage requirement).
- Style to match the overall look and feel of the dashboard.

NOTE: A good practice is to check your project for cross-browser compatibility. Making sure that it looks and functions correctly in multiple (at least three) browsers is an important part of being a top-notch developer. If you want, leave a comment to the project reviewer about which browser(s) the project was checked to ensure they are seeing things as you have designed them.

Some browser options:

- Google Chrome
- Mozilla Firefox
- Internet Explorer/Edge
- Safari

If you're having trouble with this project, make sure you take a look at this great project 7 study guide: https://s3.amazonaws.com/treehouse-techdegree/fewd/study-guides/project-7-study-guide.pdf

EXTRA CREDIT

1.) Display at least two notifications at the same time when the user clicks the alerts icon
- This could be a pop-up window or dropdown menu.

2.) Traffic chart widget

- Create a navigation similar to the one in the mockup to display different data when the Hourly, Daily, Weekly and Monthly button is selected. Add functionality to the Hourly, Daily, Weekly and Monthly buttons so that a different line chart is displayed on click.

3.) Add an "autocomplete" feature for the "Search for User" box, listing names that match the search term.

4.) Use local storage to save the settings.
- The settings are saved to local storage when the "Save" button is clicked.
- The settings are reset when the "Cancel" button is clicked.
- When page is reloaded the settings are remembered.