Volunteer Check in for Bike Co-op

Project Requirement Fulfillments

Web Developer Project Requirements
Responsive Design:
Implement responsive design using media queries, CSS Grid, Flexbox, etc.
Your application should adapt to at least two screen sizes (mobile and desktop).
Integrate responsive design as a testament to your JavaScript, HTML, and CSS integration skills.

Feature Implementation:
- Choose at least 3 items from the FEATURES LIST. Select 2 from the first set of features and 1 from the second set.
  - Integrate a third-party API into your project.
  - Consider adding a 4th feature as a backup plan.
- Failure to meet all requirements will result in incomplete status for the class.

Use arrays, objects, sets or maps to store and retrieve information that is displayed in your app.	#1	This requirement will be fulfilled in the backend table for shop admins to sort data in ascending/descending order
Analyze data that is stored in arrays, objects, sets or maps and display information about it in your app.	#2	This requirement will be fulfilled once the new user inputs all of their information is and is assinged a volunteer number. The information will be displayed back to them where they can then confirm
Use a regular expression to validate user input and either prevent the invalid input or inform the user about it (in all cases prevent invalid input from being stored or saved).	#3	This requirement will be fulfilled in the input fields requiring an ID number and input fields requiring and email address
Analyze text and display useful information about it. (e.g. word/character count in an input field)		
Create a function that accepts two or more input parameters and returns a value that is calculated or determined by the inputs. Basic math functions don't count (e.g. addition, etc).		
Visualize data in a user friendly way. (e.g. graph, chart, etc		
Create 3 or more unit tests for your application (and document how to run them)		
Convert user input between two formats and display the result. (e.g. Fahrenheit to Celcius, kilograms to pounds, etc)		
Calculate and display data based on an external factor (ex: get the current date, and display how many days remaining until some event)		

Retrieve data from a third-party API and use it to display something within your app.	#4	This requirement will be fulfilled in the autopopulated calendar API in the backend for Shop Admins displaying user log in dates and times
Create a form and store the submitted values using an external API (e.g. a contact form, survey, etc).		
Persist data to an external API and make the stored data accessible in your app		

Goals of website app

Summary: Assign a random number to every person that comes into the bike co-op. This number will be saved with the person along with other information below. This information will be used for tracking metrics of visitors for the shop.

- Save Volunteer name
- Save Volunteer number
- Track time of check-in
- Track the number of times they have visited
- Track what they are doing on their visit. Volunteering? Working on bike? Browsing shop/other?
- The user can input their email address if they would like to

Once the user hits go, it will record the time of the visit, assign a random ID number, and take the user to the volunteer fill-out page

Once the user hits go, it will record the time of the visit and take the user to a simple welcome page

Input field where a unique visitor number will be entered

Once "Go!" is clicked, it will record the time the visitor logged in and increment the visits by 1, adding it to the existing user information in the database

Button

Takes user back to first welcome page

Takes user back to first welcome page

Takes user to volunteer information input page

Takes user back to first welcome page

Back-end data base that only shop admins can see and access

Name	Volunteer ID	Date/Time of check-in	Times they've volunteered	Email address	Mechanic experience
Ringo Starr	#3456	11/3/24 - 2:30pm	14	N/A	Novice
Paul McCartney	#2167	10/30/24 - 7:02pm	23	Paul McCartney@gmail.com	Intermediate
John Lennon	#8906	10/30/24 - 6:34pm	2	N/A	None
George Harrison	#6531	10/27/24 - 1:27pm	100	George Harrison@gmail.com	Advanced (previous mechanic)

By clicking at the top, you can sort the data in order of your choosing, ascending/descending

The Date/Time will include an auto-populated calendar API that shows the dates/times when the user logged in when it is clicked to see all date/ time data