# CSE 340 – Outcome Mastery Report

Name: Diana Waite

GitHub: URL: https://github.com/dianawaite/cse340project

Deployed website URL: https://cse340-web-service.onrender.com/

## Instructions:

*For each of the course outcomes listed below, include 2-3 examples of the way you have demonstrated this outcome. Each example should include:*

* *A reference to the specific file/function you have created.*
* *A 2-3 sentence description of what you did.*

## Outcome 1: Develop to current web frontend standards of validity and practice.

## Examples:

1. Example: Add-classification.ejs
   1. File/Function: https://github.com/dianawaite/cse340project/blob/main/views/inventory/add-classification.ejs
   2. Description: Used an html form to create a functional form for people to use to add classifications to the database.
2. Example: Styles
   1. File/Function: https://github.com/dianawaite/cse340project/blob/main/public/css/styles.css
   2. Description: Used a styles css file to organize all the styles for my site. My styles aren’t perfect, but it is a good start to creating a good looking website.

## Outcome 2: Use variables, arrays, functions, and control structures in server code.

1. Example: Inventory Model
   1. File/Function: https://github.com/dianawaite/cse340project/blob/main/models/inventory-model.js
   2. Description: Used try/catch functions in the inventory model file, as a control structure to allow the flow of the program to go as needed. Try/catch is similar to if/else statements. They are helpful if you need it do to something, but has to have something else present before that can happen, so it will check and make sure it is all in order before it can perform the function you want it to.
2. Example: Inventory Controller
   1. File/Function: https://github.com/dianawaite/cse340project/blob/main/controllers/invController.js
   2. Description: Inventory Controller used to build views and connect the views to the models to keep things organized.

## Outcome 3: Develop web applications that implement common design patterns.

1. Example: Account Route
   1. File/Function: https://github.com/dianawaite/cse340project/blob/main/routes/accountRoute.js
   2. Description: Using the MVC design pattern, I created routes for account and inventory, to keep things separated and organized in my code.
2. Example: Add Inventory View
   1. File/Function: https://github.com/dianawaite/cse340project/blob/main/views/inventory/add-inventory.ejs
   2. Description: Using the MVC design pattern, I created views, including this add-inventory view, to keep things separate and organized. The view then sends that data through the route, which then connects to the controller.

## Outcome 4: Design and use relational databases for CRUD interactions.

1. Example: Account database
   1. File/Function: https://github.com/dianawaite/cse340project/tree/main/database
   2. Description: Created an account database to keep track of each client, with account name, email, password, etc.
2. Example: Inventory Database
   1. File/Function: <https://github.com/dianawaite/cse340project/tree/main/database>
   2. Description: Created an inventory database to keep track of the vehicles available, with all the various things associated with them – miles, color, make, model, etc.

## Outcome 5: Validate data (client-side and server-side) appropriate to the task.

1. Example: Account Route
   1. File/Function: https://github.com/dianawaite/cse340project/blob/main/routes/accountRoute.js
   2. Description: Used client-side validation on all my routes, so it would throw an error that would give the user some information about what went wrong, instead of just not working properly.
2. Example: Account Validation
   1. File/Function: https://github.com/dianawaite/cse340project/blob/main/utilities/account-validation.js
   2. Description: Created an account-validation file to facilitate the server-side validation on my site, making sure passwords have the correct requirements, names are strings, email is in a valid email form, etc.

## Outcome 6: Demonstrate the skills of a productive team member (such as solving problems, collaborating with others, communicating clearly, fulfilling assignments, and meeting deadlines.)

1. Example: Attended each team meeting
   1. File/Function:
   2. Description: Attended each weekly team meeting, arrived on time, tried to offer help and suggestions as I could.
2. Example: Met deadlines for the first 3 weeks of assignments
   1. File/Function:
   2. Description: I worked on and turned in my assignments for the first 3 weeks on time, helped with the team assignments as needed, and communicated clearly with the teacher when I wasn’t going to make the deadline on some of the assignments.