|  |  |
| --- | --- |
| **File Name** | **Description** |
| R1 In page form validation.docx | This excerpt of Javascript code acts to validate all input forms on the report and registration page. This was to ensure valid inputs into all fields as a first level of security before any database input can take place. This form validation includes checking for blank text inputs, correct dropdown list values and consistent inputs in the password fields.  These checks are taken to ensure that there are valid inputs within the page and allow for the inputs to be passed into the database as variables. If there is not a valid input, the variable is set to null and not allowed to be passed. |
| R1 Report Page Draft.html | This document is a draft page of the “Report” page for the final solution written in html. The report page functionality is to allow administrative users to create additional Health and Safety, Parking Violations and other reports within the database.  This draft page was not designed for aesthetic consistency, rather featured the fields which would be required for report functionality. This included improvements upon other fields found in pages such as the registration page, including a full list of the faculties, and an immediate action button for reporting emergencies.  This document was sent to another group member to be used as a basis for developing the fully realised page which is seen in the final iteration. |
| R1 Test Cases.docx | The python test cases for Release 1 can be seen in the **forms.py** file in the repository. For the purpose artefact evaluation, they were placed into a separate word file as seen on the left.  These test cases use the Testing functions in order to ensure that the database and validity of forms. These test cases use a hypothetical user in conjunction with the functions to test the various inputs in practical application. These cases used the functions in repetition to test every aspect of the user’s interactions with the permit and violation reports. |
| R1 Test Functions.docx | The python test functions for Release 1 can be seen in the **testfunctions.py** file in the repository. For the purpose artefact evaluation, they were placed into a separate word file as seen on the left.  These functions were developed, based off of previously developed pseudocode in order to test the functionality of our solution. These testing functions for the first release focused on the systems created by the rest of the team, i.e. permit application, site information, and violation reports. This ensured that the group progression was not halted by incorrect code or invalid database inputs. Having these correct inputs partially allowed a secure database, helping to meet the required user stories. |
| R1 Test Pseudocode.docx | This pseudocode was developed in the planning stages of the first release. I developed this pseudocode in conjunction with the existing html pages to assist with developing the tests cases for form validity and functionality. The pseudocode was initially planned for use among anyone developing for test cases. The pseudocode is structured to display the input from certain forms as well as their expected output. |