**IMPLEMENTATION:**

**MODULES:**

* **User**
* **Admin**
* **False Positives and False Negatives**
* **Machine Learning Classifier**

**MODULES DESCRIPTION:**

**User:**

The User can register the first. While registering he required a valid user email and mobile for further communications. Once the user register then admin can activate the customer. Once admin activated the customer then user can login into our system. User can do the data preprocess. First required running website name. By using that website the user can test the csrfs. By help of bolt tool the user can fetch related all csrfs and generated algorithm names. The result will be stored in json files. Later the user can get the results of Mitch dataset. The mitch dataset tested for POST method as well GET method to. The result will be displayed on the browser.

**Admin:**

Admin can login with his credentials. Once he login he can activate the users. The activated user only login in our applications. The admin can set the training and testing data for the project of the Mitch Dataset. The user search all urls related csrf token admin can view in his page. The admin can also check the POST method performed data from the dataset and GET method related data also.

**False Positives and False Negatives:**

Mitch produces a false positive when it returns a candidate CSRF that cannot be actually exploited. This is something relatively easy to detect by manual testing, though this process is tedious and time-consuming. In general, it is not possible to reliably identify when Mitch produces a false negative, because this would require to know all the CSRF vulnerabilities on the tested websites. To estimate this important aspect, we keep track of all the sensitive requests returned by the ML classifier embedded into Mitch and we focus our manual testing on those cases. This is a reasonable choice to make the analysis tractable, because we first showed that the classifier performs well using standard validity measures.

**Machine Learning Classifier:**

The ML classifier used by Mitch was trained from a dataset of around 6000 HTTP requests from existing websites, collected and labeled by two human experts. The feature space X of the classifier has 49 dimensions, each one capturing a specific property of HTTP requests. Those can be organized into following categories.

following set of numerical features:

numOfParams: the total number of parameters;

numOfBools: the number of request parameters bound to a boolean value;

numOfIds: the number of request parameters bound to an identifier, i.e., a hexadecimal string, whose usage was empirically observed to be common in our dataset;

numOfBlobs: the number of request parameters bound to a blob, i.e., any string which is not an identifier;

reqLen: the total number of characters in the request, including parameter names and values.