

Chapter 3

System Analysis

3.1 Functional Requirements

Functional requirements describe all the required functionality or system services. Functionality is described as a set of inputs, the behaviour, and outputs expected from the system. They are functions or features that must be included in a system in order to satisfy the business needs and must be acceptable to the system users.

The proposed system should provide following functional requirements:

- The system requires the chrome extension to automatically take the URL from the web browser when the web page is loaded.
- The chrome extension must be able to pass the URL to the cloud where the model is stored.
- The cloud must be able to successfully extract the features of URL and feed to the model.
- The model must be able to predict to the URL as phished or safe to the best possible accuracy of the algorithm used.
- The Cloud must be able to give the result back to chrome extension.
- The chrome extension must alert the user regarding the same.

3.2 Non-Functional Requirements

The non-functional requirements define the properties and constraints of the system. Non-functional requirements impose constraints on the design or implementation. It is used to judge the operation of a system.

The proposed system should provide following non-functional requirements:

- Acceptable Verified as meeting the stated objectives
- Accessible Accessible from different devices
- Availability 24 x 7 x 365.
- Secured and Restorable System and data backups. Business Continuity / Disaster
- Maintainability- The ease with which the system can be changed, whether for bug fixes or to add new functionality.
- Modifiable and Extensible Ease and cost to make ongoing changes
- Responsive / Performance Response times. Speed of page loads and calculations
- Reliable Consistent and dependable quality of service
- Usable Easy to use by target users. Both humans and web crawlers

3.3 Specific Requirements

Hardware Requirements

- Hard Disk: 40 GB hard disk space and above.
- RAM: Minimum 512 MB and above.
- 1GHZ processor

Software Requirements

- Chrome Browser.
- Middle-ware: Algorithmia (Cloud API provider and storage provider).
- Programming language for coding: JavaScript/Python.
- An internet connection is required.

3.4 Use Case Diagram

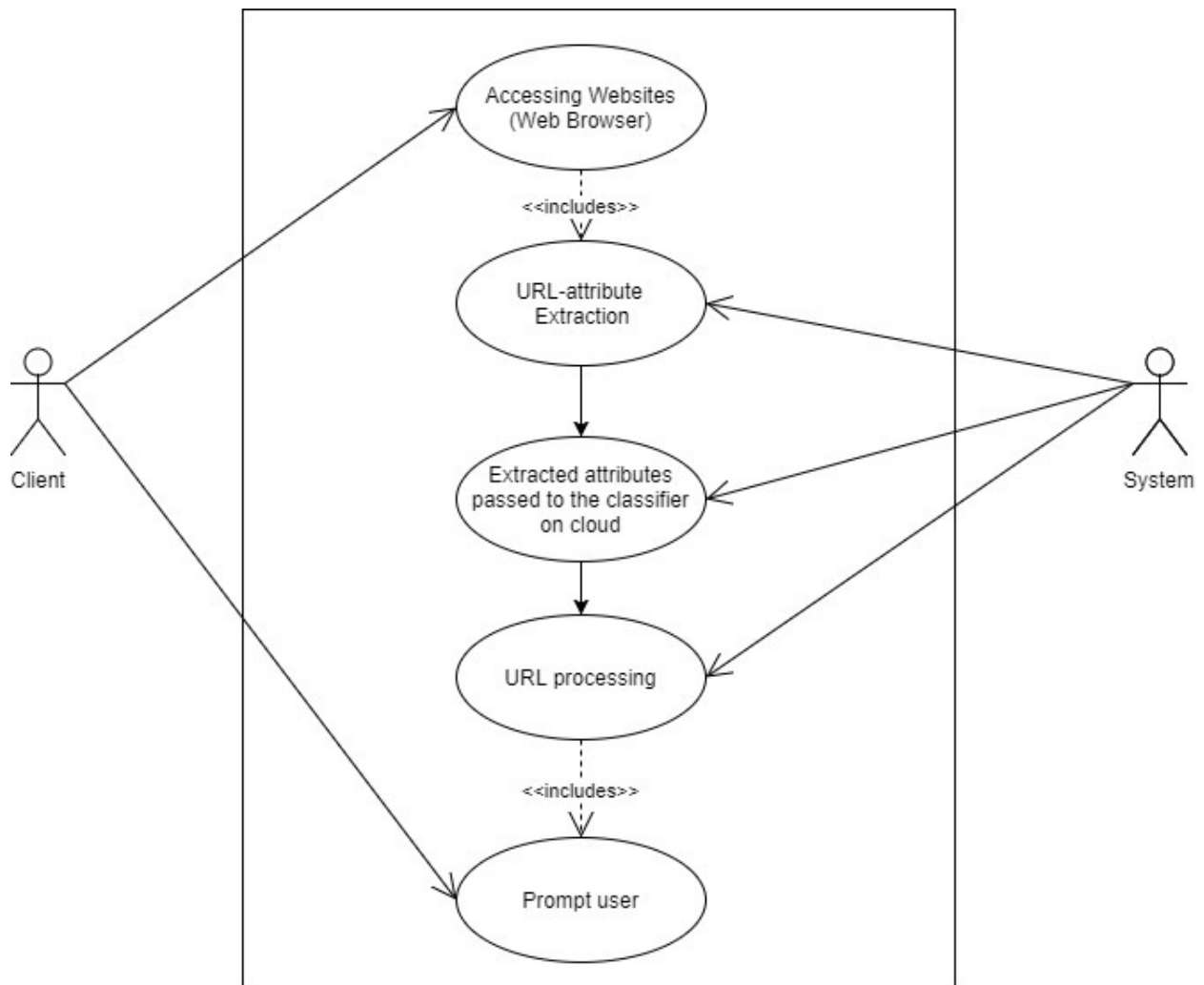


Figure 3.4.1: Use Case diagram for Detecting Phishing websites using Data Mining

In figure 1, we have shown the use case diagram for Detecting Phishing websites using Data Mining. The client just has to just visit the website and it is solely the job of the system to decide whether the website is safe of phished. The System here includes the Chrome extension as well as the Classifier residing on the cloud.

Use case Description:

Actors:, *Client* and the *System* are the two end users for Phishing Detection.

Table 1: Use Case Description for UC01- Accessing websites (web browser).

Use Case	Accessing Websites (Web Browser)
Use Case ID	UC01
Actor	Client
Description	Client visits various websites through a web client i.e. Web Browser, which may be safe or Phished.

Table 2: Use Case Description for UC02-URL attribute extraction.

Use Case	URL-attribute extraction
Use Case ID	UC02
Actor	System
Description	The visited URL will be fetched by the Chrome extension and various attributes will be extracted from it.

Table 3: Use Case Description for UC03-attribute extraction.

Use Case	Extracted attributes passed to the classifier on cloud
Use Case ID	UC03
Actor	System
Description	The attributes extracted by the Chrome Extension are passed onto the classifier on cloud for testing purpose.

Table 4: Use Case Description for UC04-URL processing.

Use Case	URL Processing
Use Case ID	UC04
Actor	System
Description	Here, the testing of the URL attributes is done using appropriate classifier.

Table 5: Use Case Description for UC05-prompting user

Use Case	Prompt User
Use Case ID	UC05
Actor	Client
Description	The predicted output is given to the user whether the website is phished or safe.