**CHAPTER – 3**

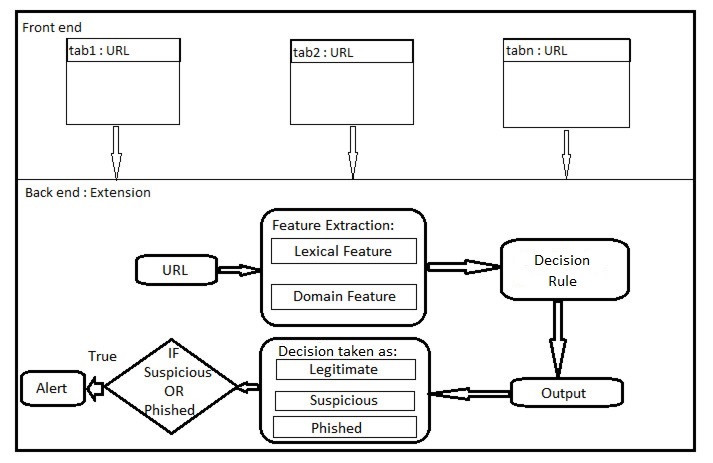
**SYSTEM DESIGN**

**3.1 INTRODUCTION**

Every system emerges from the proper system design. Good plan leads to good design which in turn gives an efficient system. The proposed system includes a development phase to happen at the hardware level and at the software level. The hardware development includes selecting the proper hardware and linking them together to form a flow. Software level development includes coding and integrating it into the hardware. Only if both the modules combine and work together, it acts as a single entity. Here we design the system for the phishing URL detection, the following parameters that are taken into consideration are: URL, URL features like URL length, hostname length, ‘.’ count, ‘@’ count, ‘-’ count, ranking, domain traffic, domain age, SSL certificate, Decision rule mining on the feature and browser. Then we analyze it with feature of a URL to detect the phished website opened in the browser. Phishing Alerts characteristic the first step of identification with the URL features and apply the Decision Rule mining techniques that helps to detect the phished URL detection opened on the bowser. The project intends to develop an automated tool with help of Plugin using machine learning techniques and strategies to detect the phished URL in much lesser time. In this chapter, system design is reflected with the illustration of various diagrams like system architecture, use case, activity, entity relationship and data flow diagrams.

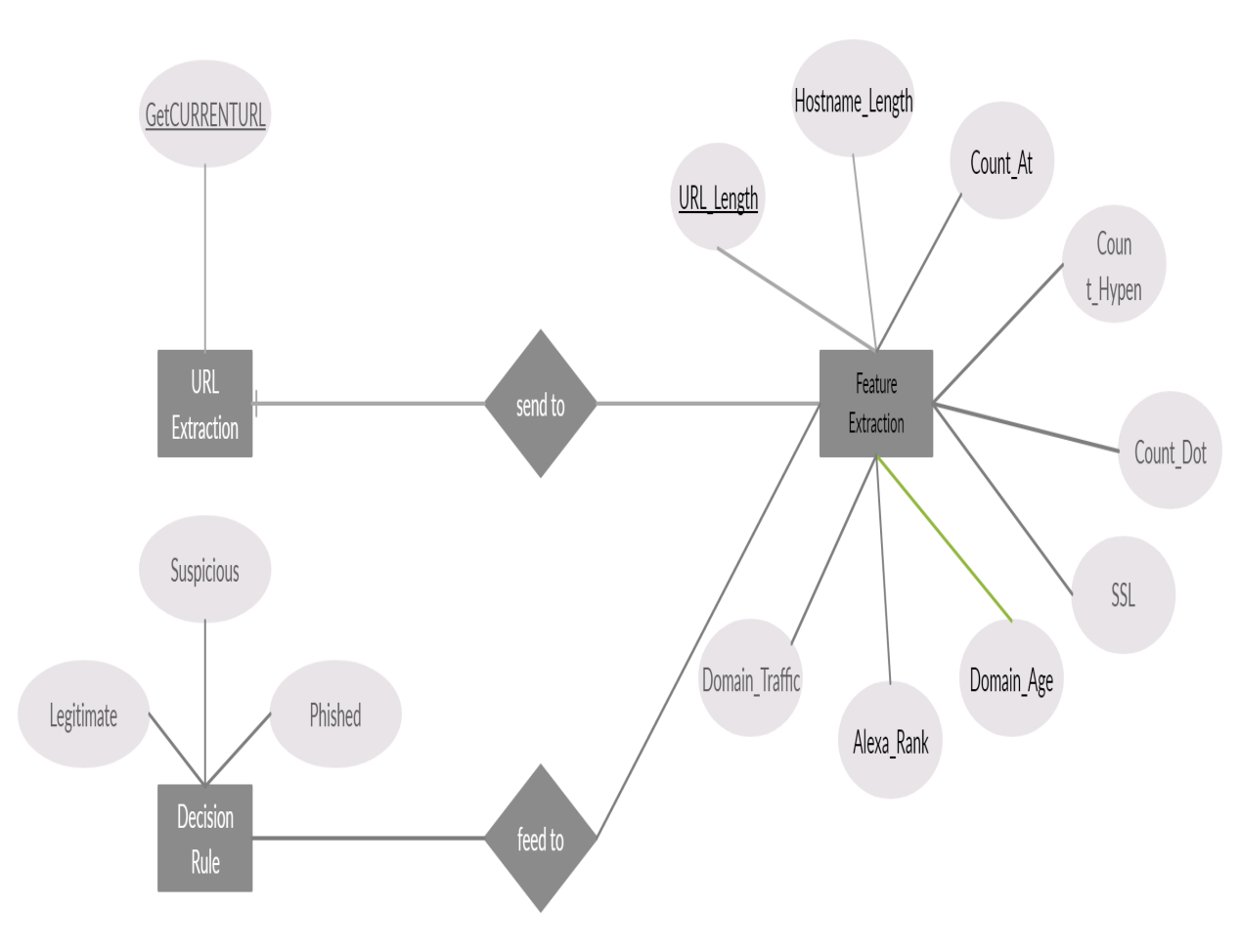
**3.2 SYSTEM ARCHITECTURE**

System architecture shows, how the system work in client side. The Figure 3.1 represents the system architecture of extension added to the chrome browser. The entities involved in this process are the Tab, URL and the extension. The working of the extension is divided into 2 modules. When the internet users add the extension to the chrome, the working of the extension begins. Initially, the extension starts its work, when any webpage opened in a current tab. It constantly checks the active tab, once any webpage opened in current tab, the next module is instantiated. In the second module, it get URL of the active tab and then extract the lexical and domain features from the URL. The features that are extracted from the URL are feed as input to the Decision rule. The output from the Decison rule is classified as Legitimate, Suspicious and Phished based upon the condition. Once the output is suspicious or phished, it alerts the user to warn about the webpage opened in the active tab.

***Figure 3.1 Proposed System Architecture***

**3.2 ENTITY RELATIONSHIP MODEL**

The Figure 3.2 represents the ER Diagram of Phishing Alerts. The entities of Phishing Alerts are the URL, Feature extraction and Classification. The Feature extraction and the Classificaction are ased upon URL. Thus the Url extraction and the feature extraction possess “send to” relationship. Thus the Url extraction and the classification possess “feed to” relationship. Each and every entity will have their attribute. Here, URL’ attributes are GetCURRENTURL. Likewise, the attributes of Feature extraction are the URL\_Length, Hostname\_Length, Count\_At, Count\_Dot, Count\_Hypen, Alexa\_Rank, Domain\_Age, SSL and Domain\_Traffic. Attributes of Decision rule are Legitimate, Suspicious and Phished.

***Figure 3.2 ER Diagram***

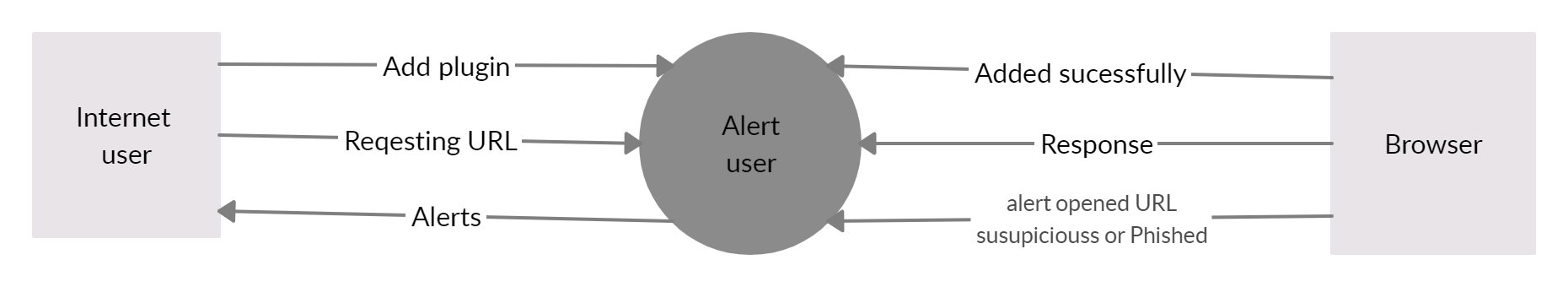
Each and every entity of the system is linked to one another with different kinds of relationship which is one to one, one to many or many to many.

**3.3 DATA FLOW DIAGRAM**

The below represented diagrams are the data flow diagrams which explains our work.

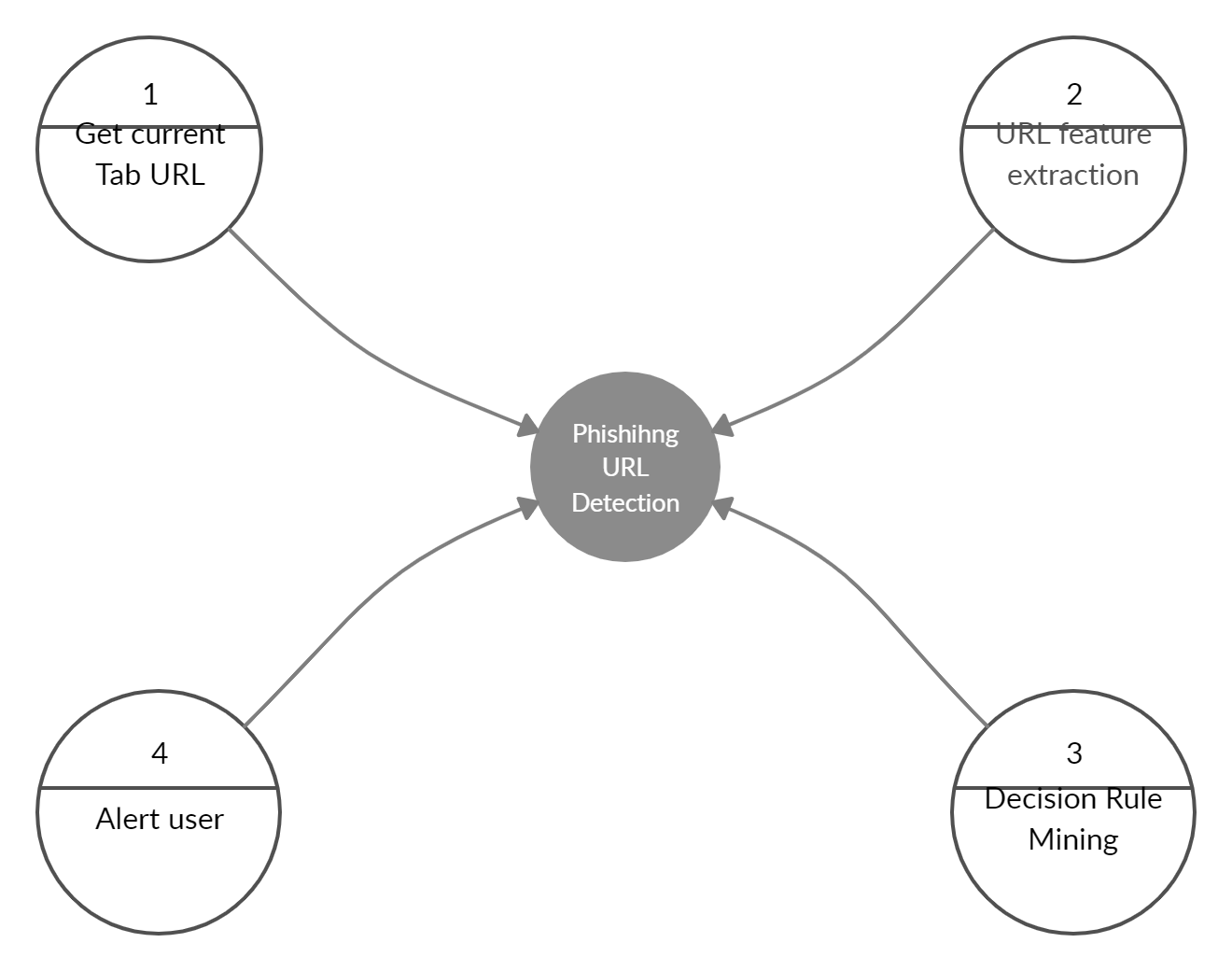
**3.3.1 DFD LEVEL 0**

In the level 0 of data flow diagram explains the basic flow of the project. Figure 3.3 shows how the user has been alerted for phished URL by the browser. In this level 0, the internet user add the plugin to the browser extension and request the website to the server using URL and the browser extension check the URL, then the alert has been made to the user fo the respective tab.

***Figure 3.3 DFD LEVEL 0***

**3.3.2 DFD LEVEL 1**

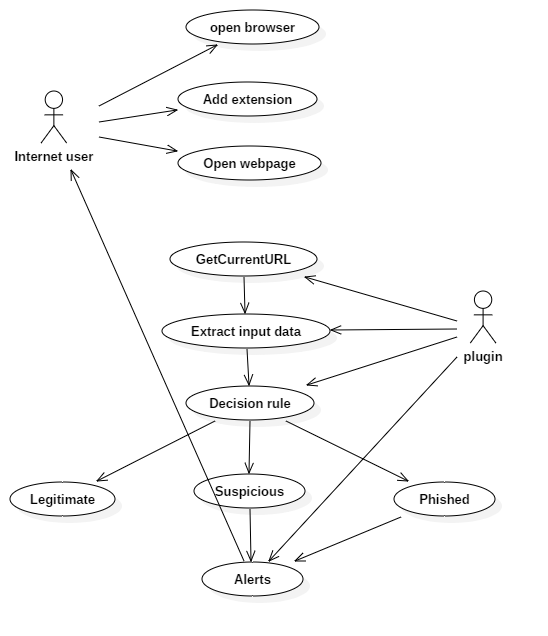
In the below figure 3.4 It highlights the main functions carried out by the system. The process of the plugin is shown here, initially the current ta of the URL is get to process, then the features of the URL is extracxted one by one. The extracted features are feed as a input to the decision rule minning, which process the input data and decide URL is legitmate, suspicious or phished. If the decision minning outout is phished or suspicious then the alert is given to the user.

***Figure 3.4 DFD LEVEL 1***

**3.5 USE CASE MODEL**

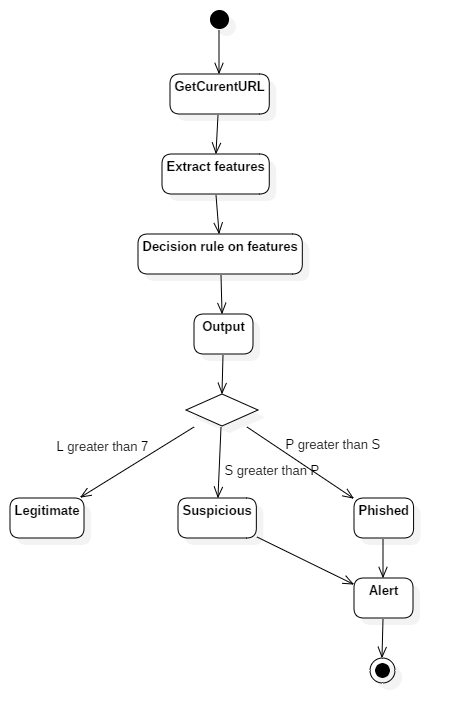
**3.5.1 USE CASE DIAGRAM**

The Figure 3.5 represents the use case diagram of Phishing Alerts. There are two actors involved in the use case diagram are internet user and plugin. The use cases associated with admin are open browser, add extension and requesting webpage. The use cases associated with plugin are get current URL, extract input data, decision rule and alert. The actor user add extension to the browser and user request the any URL from the browser. The next actor plugin in which get the URL of the current tab and extract the features of the URL then futher apply the decision rule on all features of a URL to take a decision about a URL as legitimate, suspicious and phished. If the decision is finalled as phished or suspicious, then the plugin will alert the user.

***Figure 3.5 Use case Diagram***

**3.5.2 ACTIVITY DIAGRAM**

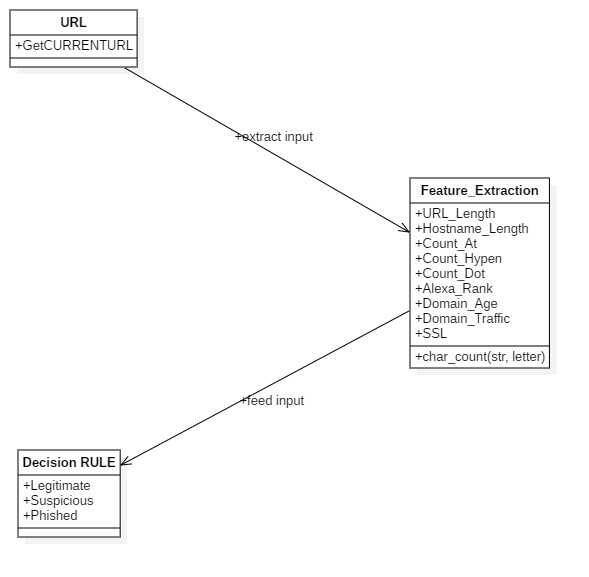
Activity diagram is a flowchart to represent the flow from one activity to another activity as showed in figure 3.6. The process flow starts with get current URL that contain current URL of the tab, where the required features are extracted for effective modelling.

***Figure 3.6 Activity Diagram***

Based on this data, a decision model is built for making decision and alerting user. Once the decisions are made, the alerts will be given to the user as per the decision.

**3.5.3 CLASS DIAGRAM**

The Figure 3.7 illustrates the class diagram of Smart Phishing Alerts. The independent classes or components involved here are the URL, Feature extraction and Decision Rule. The vital component here is the URL. The Feature extraction is connected to other classes such as the URL and the Decision Rule. Here, URL’ attributes are GetCURRENTURL. Likewise, the attributes of Feature extraction are the URL\_Length, Hostname\_Length, Count\_At, Count\_Dot, Count\_Hypen, Alexa\_Rank, Domain\_Age, SSL and Domain\_Traffic. Attributes of Decision rule are Legitimate, Suspicious and Phished.

***Figure 3.7 Class Diagram***

**3.6 SUMMARY**

This chapter helps in providing a clear idea about how our system works and the process flow and motive of our system for provide security and awareness to the user, detecting the anomalies for the user requested webpage through the browser is going to be solved. We have designed this model in such a way that the phished URL are analyzed effectively. The techniques what we implement in this system design structure is the decision method, which detect the phished URL from the browser without any interaction from the user. The system architecture model mentioned in this chapter gives the featured steps and modules and plugin tool is builted.