**Software Design**

**Specification**

**for**

**Crop Disease and Severity**

**Detection and**

**Prediction of Pesticides**

**Version 1.0**

**Prepared by:**

**Anuj Jain (BE/25036/13)**

**Yash Tongia (BE/25030/13)**

**Mentor:**

**Dr. Madhavi Sinha**

**Birla Institute of Technology, Mesra, Jaipur Campus**

**October 12,2017**

**1. Introduction**

**1.1 Purpose of this document**

This document will outline in detail the software architecture and design for the Crop Disease and Severity Detection and Prediction of Pesticides System.This document will provide several views of the system's design in order to facilitate communication and understanding of the system.

**1.2 Scope of the development project**

This document provides the architecture and design of Release 1.0 of the Crop Disease and Severity Detection and Prediction of Pesticides System. It will show how the design will accomplish the functional and non-functional requirements enlisted in the Software Requirements Specification (SRS) document.

**1.3 Definitions, acronyms, and abbreviations**

Not applicable.

**1.4 References**

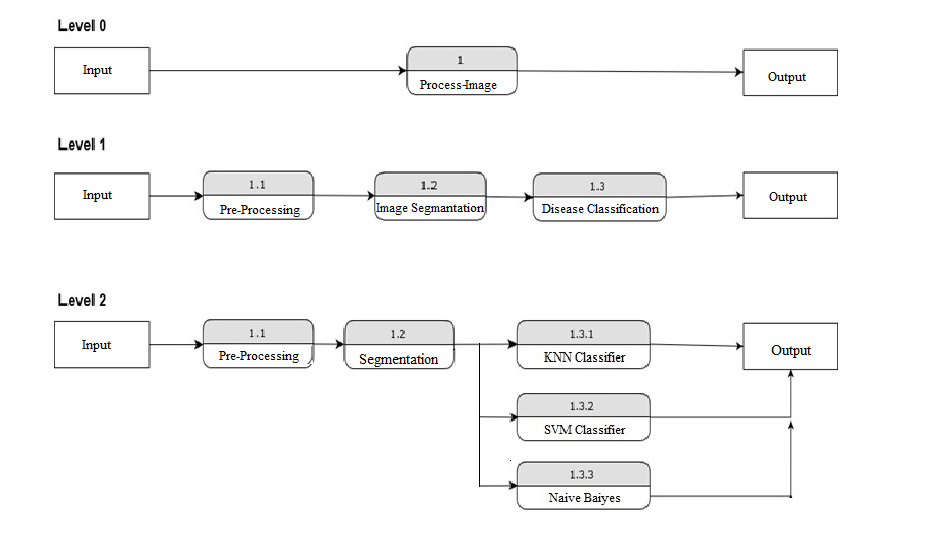
* Crop Disease and Severity Detection and Prediction of Pesticides System Software Requirement Specification(SRS).

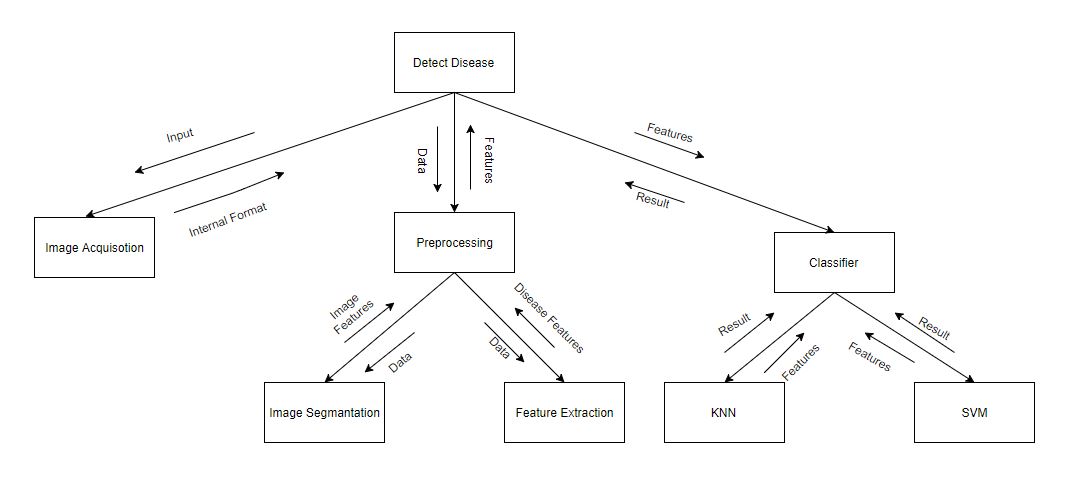
.

**1.5 Overview of document**

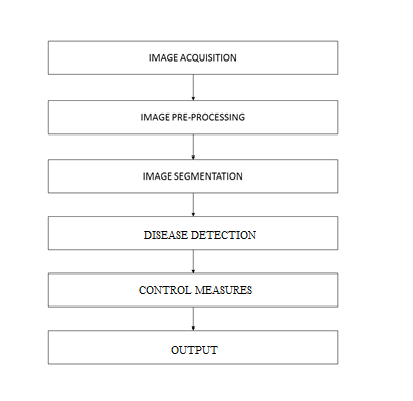
* The first section describes the introduction of document.
* The second section describes the system architecture, in this section we mention the modules of this project and their structures.
* The third section consists of detailed description of the modules described in the previous section modules.
* The fourth section states how our project can be reused and its relationship to other products.
* The fifth section gives the complete idea about design decision and tradeoffs which we had to do in order to simplify the design.
* The sixth section consists of pseudocodes for the components.

**2. Data Flow Diagram**

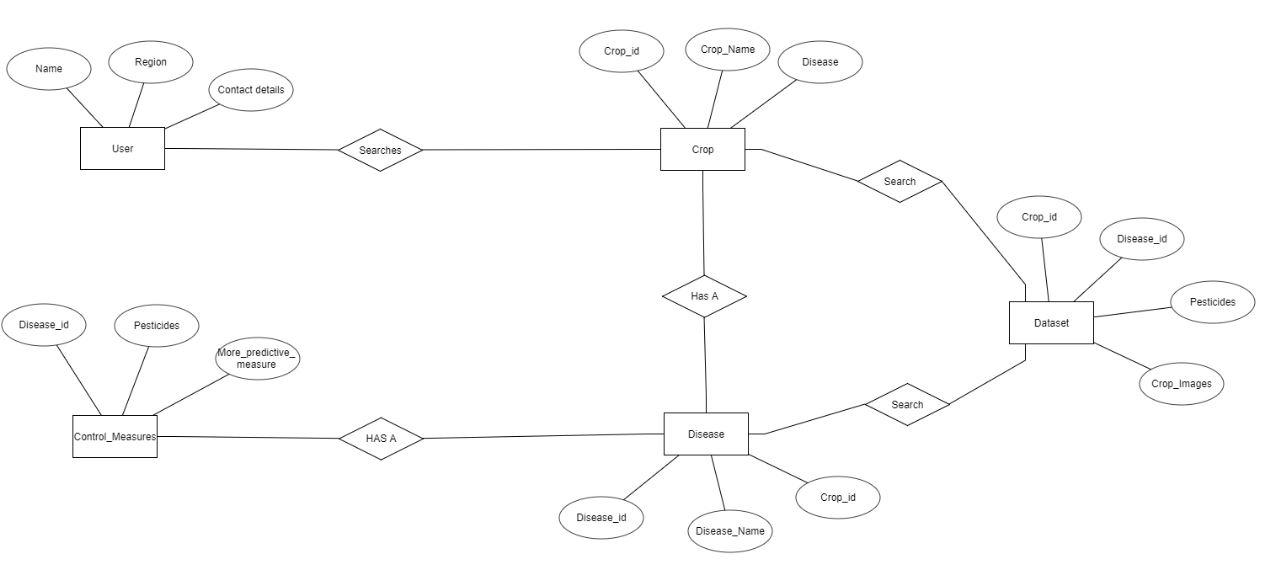
****



**3.1 Data Processing Diagram**

****

**3.2 E-R Diagram**



**4. System architecture description**

This section is the main focus in the first version of the SDS, the high level design. This should give a good view of exact organization of the system as per the requirements.

**4.1 Overview of modules / components**

This project can be divided into 2 major components namely:

1. Image Acquisition and Feature Extraction.

This major component comprises of various units which will be used to extract features of the given image that can be compared to the preloaded dataset. These small components include image acquisition, image pre-processing, image segmentation and calculation of feature values.

1. Classification of the disease and Result Display.

This unit compares the results of the previous unit that is extracted features with the pre-loaded dataset using the various classifiers algorithms such as KNN, Naïve Bayes, SVM so that disease is accurately detected and there are no chances of fault. Furthermore this unit displays the results and control measures associated with the disease.

The main objective of this project is to predict whether disease is present or not if present then detects the disease with which the crop is affected which is to be executed at client’s side only, without any connection to server. Searching for the information about a particular disease is an additional component thus it can be moved to server side execution.

**4.2 Structure and relationships**

The functionalities available to the user will be based on the type of login details entered, we have two types of user a general user and the other one the administrator. The user can use the functionalities such as details submission, disease detection and disease enquiring. On the other hand the administrator can add on new features such as modifying the dataset used and providing other utilities.

**4.3 User interface issues**

The GUI can be divided into 2 parts. One is for the user, and the other is for the Administrator of the system. The user side interface will be optimized to provide the user with an easy to use interface.

**5. Detailed description of components**

**5.1 Login Description**

|  |  |
| --- | --- |
| Identification | Login |
| Type | Module |
| Purpose | The purpose of this module is to provide entry to the system. Based on the type of login, the user is provided with various facilities and functionalities. |
| Function | The main function of this module is to allow the user to use the system. This module provides two types of login - General user and Administrator. |
| Subordinates | There are two types of login possible in this system – General user and Administrator. A general user will have access to the functionalities of the system such as searching for a particular crop and disease, detection of the disease. While the administrator will have complete access over the system, he can add more details to the database, edit the system to provide more functionalities. |
| Dependencies | Login will be dependent on the server, the system should be connected to the server. |
| Interfaces | A computer will be used to access the login details. |
| Resources | A regular PC with input and output devices will be required. The PC should have an operating system with a browser and it should be connected to the server. |
| Processing | When a user requests for login from the server, there are two scenarios that arise - if a new user wishes to register, he is taken to the new user registration page or If an existing user requests login, his username and password are checked from the database by the server. If all entries are correct, he is redirected to the home page. |
| Data | On the servers, user data will be maintained in MySQL Database with relevant fields and it will matched with the data entered by the user. |

**5.2 Data Acquisition**

|  |  |
| --- | --- |
| Identification | Data Acquisition |
| Type | Module |
| Purpose | The Main purpose of this unit is to get input initial input/data about the data upon which the classifiers will be implemented. This unit will tell the system on which data set component it has to perform classification. |
| Function | The main function of this module is to get metadata from the user. The user will select the crop from the given list and specify the region of his work. Moreover it gives the user choice to search a particular disease associated with the crop or he/she wants to detect the disease. |
| Subordinates | The administrator is entitled to perform any changes in the data set. She/he can add more crop diseases. Make changes to the way the final result is calculated. |
| Dependencies | This phase depends on login details of the user. As usually the user searches for a particular set of crop diseases. This module is thus dependent on the user details. This component defines the further course of actions as it will depend on the data provided by user in this unit. |
| Interfaces | A computer will be used to gain access to the server on which system details are stored. |
| Resources | A regular PC with input and output devices will be required. The PC should have an operating system with a browser and it should be connected to the server. |
| Processing | The user inputs the crop for which he/she is looking for,the system validates the input given by the user. After this the system prompts user to enter disease if he is known to it and want to know about it. If the user selects the disease then the data is fetched from data base and displayed directly. Otherwise the system transmits the control to image processing unit of the system. |
| Data | On the servers, user data will be maintained in MySQL Database with relevant fields. On the demand of the user about the particular disease the data is fetched from the server |

**5.3 Image Processing**

|  |  |
| --- | --- |
| Identification | Image Processing |
| Type | Module |
| Purpose | The purpose of this module is to process the image obtained from to user and extract features associated with it to be compared with the dataset. |
| Function | The main function of this module is to extract features from the given image and present them to the next module which is classification. This module involves image acquisition, image pre-processing and segmentation, Feature values extraction from the processed image.  The small units work in the above mentioned order to obtain the type of input upon which processing can be done. |
| Subordinates | The functions involved in this unit process the image stored as a matrix using MATLAB tools. The sub-units image acquisition, pre-processing, and segmentation result in final image which is to be processed. The various attributes are then calculated from this to be checked against the dataset |
| Dependencies | It depends on the Data Acquisition module. This unit is implemented only if the user doesn’t know about the disease and wished to upload an image and detect the disease. |
| Interfaces | A computer will be used to access the GUI associated with this unit. A simple upload button is to be created. Upon getting the image the processes run in the background and the intermediate result gets transmitted to the next module. |
| Resources | A regular PC with input and output devices will be required. The PC should have an operating system with a browser and it should be connected to the server. |
| Processing | The extracted features associated with the image are stored on the client side and are compared against the already stored values.  The image is processed first as identifying the required segment (the functionally usable part of the image),then this image segment is processed accordingly. Green pixels are masked so as to focus on the diseased part of the crop’s image. Then finally the image is converted to gray scale to analyze the various textures present in the image. Now for each pixel the attributes values are calculated. |
| Data | On the servers, user data will be maintained in MySQL Database with relevant fields and all the information regarding a disease will be fetched if required. |

**5.4 Classification**

|  |  |
| --- | --- |
| Identification | Classification |
| Type | Module |
| Purpose | The purpose of this module is to assign a class name (classify) the input image. The aim is to use the extracted feature values and compare them against the pre-stored dataset using various classifiers algorithms and determine the disease associated with the image. |
| Function | The main function of this module is to classify the image so as to which class of disease it belongs to. This unit is the most important unit of the system which fulfills the target objective of the project. |
| Subordinates | This Unit contains five sub units. Four sub units classify the input image using four different algorithms and the fifth units combines the output of the four classifiers so as to produce accurate results with least possible error. |
| Dependencies | This unit completely dependent on the previous unit in which computes the feature values of the input image. Depending upon the successful computation of these values the classifier algorithms produce accurate results. |
| Interfaces | A computer will be used to access the classification module and it will display the disease with which the plant is affected. |
| Resources | A regular PC with input and output devices will be required. The PC should have an operating system with a browser and it should be connected to the server. |
| Processing | The input feature values are used by the classifiers namely :  to assign the disease class level to the input image. The percentage similarity with the healthy image also determines the intensity of the disease. This value is used to predict the overall intensity of the disease in the whole plantation. |
| Data | The data required will be loaded on to the client side on successful installation of the software at the client side. In this case this module will work without any communication to the server and thus without internet facility. |

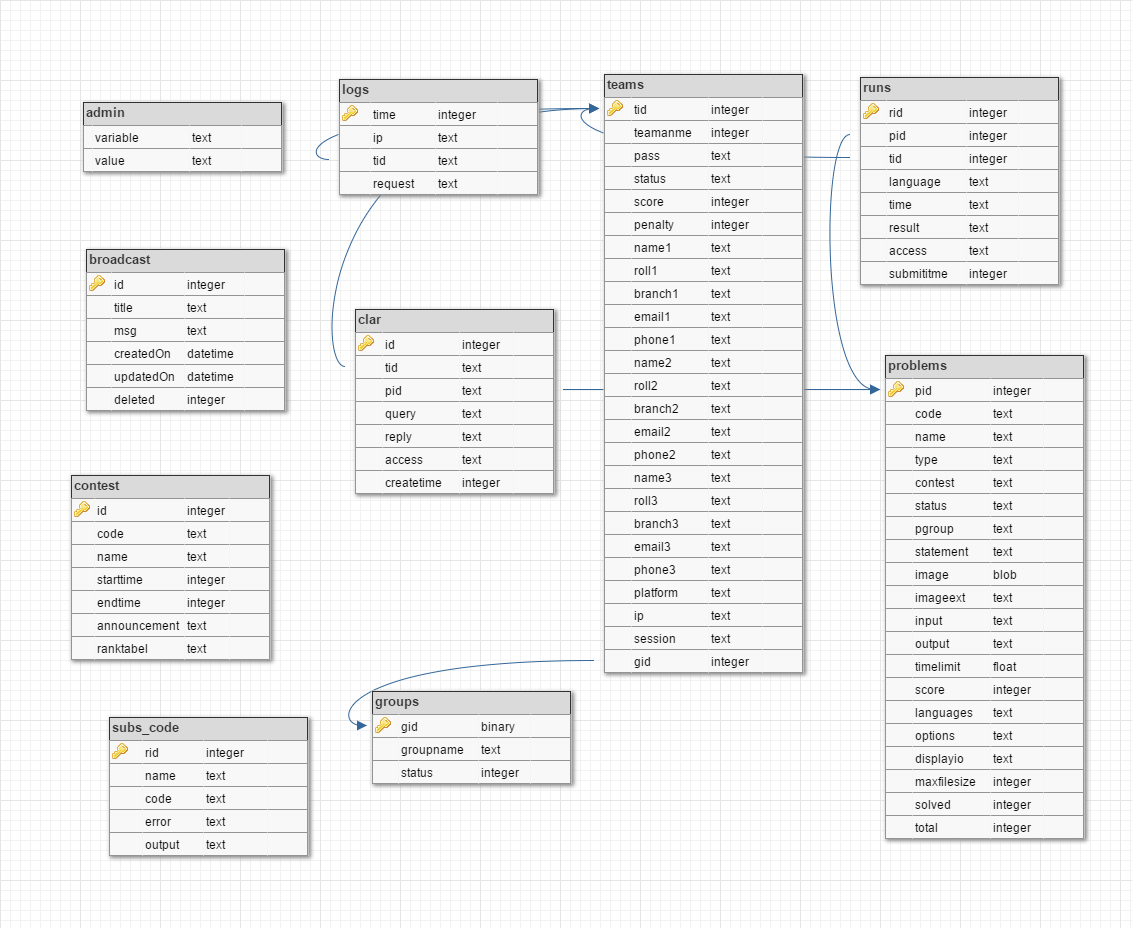
**5.5 Result Formulation**

|  |  |
| --- | --- |
| Identification | Results |
| Type | Module |
| Purpose | The purpose of this module is to maintain a rank list of all the users of the system. |
| Function | The main function of this module is maintain a rank list and keep it updated. As soon as someone solves a problem his/her rank will be updated in the rank list. |
| Subordinates | There will be two types of rank list – One of them will be a general rank list which will consider all the practice problems as well and the other type of rank list will be the one maintaining ranks from the current contest. |
| Dependencies | It depends on the user, the number of problems solved by a user and the total number of user. It will also depend on the contest module and the practice module. |
| Interfaces | A computer will be used to access the rank list and it will displayed as a list with usernames their scores, and their ranks. |
| Resources | A regular PC with input and output devices will be required. The PC should have an operating system with a browser and it should be connected to the server. |
| Processing | The rank list will process the submission list and will display the ranks of the user. |
| Data | It will collect data from the submissions table. |

**5.6**

|  |  |
| --- | --- |
| Identification | Submissions |
| Type | Module |
| Purpose | The purpose of this module is to maintain a list of submissions in order from latest to the oldest. |
| Function | The main function of this module is to display a list of submissions in order from latest to the oldest. It maintain a complete log of submissions with their details like user who submitted, problem, submission id, time of submission, language and its result. |
| Subordinates | The submissions contain all the details regarding a submission and it can be of two types – submission during a contest and submission of a practice problem. |
| Dependencies | It depends on the problem module, i.e. to which problem does the particular submission belong. It depends on the user who submitted the solution and it depends on the language in which it is submitted. |
| Interfaces | A computer will be used to access the problem module and view list of submissions with an option to view your own submission while Administrator can view the code of any submission. |
| Resources | A regular PC with input and output devices will be required. The PC should have an operating system with a browser and it should be connected to the server. |
| Processing | The list contains all the submissions made and displays the complete list of submissions with full details. |
| Data | On the servers, user data will be maintained in MySQL Database with relevant fields and it will be fetched when needed, |

**7.0 Database Design**

****

**7.0 Reuse and relationships to other products**

This project has very high scope of reuse. The web interface can be improved and enhanced. The design of the website can be changed according to your need and functionality can be added or removed.

This project can only compile codes of C, C++ and JAVA; other languages can also be added to the system. For clarifications and broadcasting messages a system to send notification to every system.

This project will be running locally, it can be extended to the run on the internet.

**8.0 Design decisions and tradeoffs**

System is designed in a simple and elegant way, so as to make it easy for everyone to use it. It is designed in a responsive manner so that each and every device will have the interface according to its resolution.