

# **Crop Prediction System**

## **Problem Statement:-**

To create a system which can predict a suitable crop which can be grown in a particular area on the basis of different environmental factors such as humidity, rainfall, precipitation etc.

The system will predict the best and profitable crop that can increase the yield and productivity by considering the previous as well as current environmental conditions.

## **Objectives:-**

- To guide people to select the best suited crop for their geographical area.
- To provide information about different crops and their favourable conditions.
- User friendly and easy to access.
- To give better production outcomes.
- To make people earn better profits with the available resources and conditions.
- To provide easy and feasible results.
- Faster and more accurate results.

## **Work division:-**

Ayush Mundra – To design a user friendly front-end and to collect necessary data required for the system.

Prakhar Vyas- To create a back-end application based on neural network and to create database for the system.

Krishan kumar- To create a back-end application and to test the system using different test cases.

## **Feasibility:-**

Feasibility study is defined as evaluation or analysis of the potential impact of a proposed system. The objective is to determine whether the proposed system is feasible or not.

### **Technical feasibility:-**

The current technical resources such as NetBeans (ide), sublime text 2, oracle server etc. are sufficient for the proposed system. So, the system is technically feasible.

### **Economic feasibility:-**

The proposed system requires very less capital. Also, no specific hardware is required. So, the system is also economically feasible.

### **Operational feasibility:-**

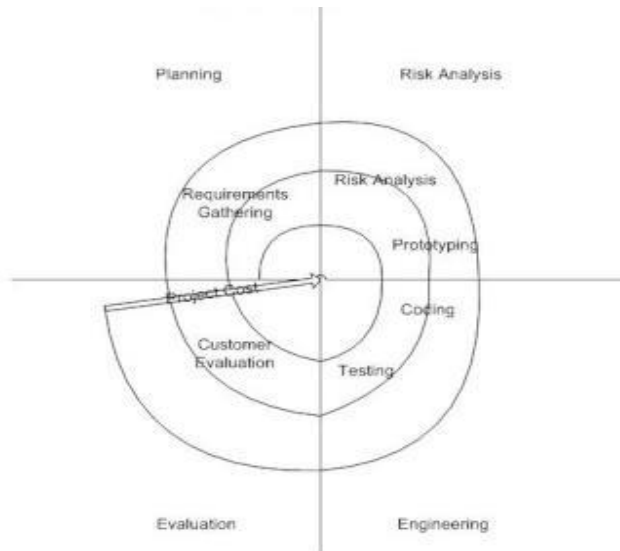
The interface of the system is quite user-friendly and easy to operate such that a user who has very less exposure to computer can easily understand and operate it. So, the system is also operationally feasible.

## **Process model:-**

For our system we choose the spiral process model. The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (known as Spirals). The baseline spiral, starting in the planning phase in which requirements are gathered after which risk is assessed. Each subsequent spiral builds on the baseline spiral.

### **Reasons for using spiral model:**

- 1) High amount of risk analysis hence avoidance of risk is enhanced.
- 2) Strong approval and documentation control.
- 3) The model is very flexible hence additional functions and updates can be made easily without proving to be too costly.
- 4) Software is produced early in the software life cycle.
- 5) As the system has to be trained with much data so learning of system can be done in different spirals.



## Spiral Model Diagram

### Deliverables:-

Outcomes of the system will be:

- The best suited crop will be identified according to geographical location of the user.
- Additional information about the predicted crop.
- Ideal and favourable condition information about different crops.

### Platforms:-

- Java Platform
- Open Web Platform(OWP)
- Oracle Database

## SYSTEM SCHEDULING:-

### GANTT CHART:-

ID	Task Name	Start	Finish	Duration	Feb 2015				Mar 2015				Apr 2015					
					1-2	8-2	15-2	22-2	1-3	8-3	15-3	22-3	29-3	5-4	12-4	19-4	26-4	
1	PLANNING	02-02-2015	03-02-2015	2d	<div><div></div></div>													
2	FUNCTIONAL REQUIREMENTS	04-02-2015	09-02-2015	4d	<div><div></div></div>													
3	NON-FUNCTIONAL REQUIREMENTS	09-02-2015	13-02-2015	5d	<div><div></div></div>													
4	DESIGN	16-02-2015	06-03-2015	15d	<div><div></div></div>													
5	CODING	09-03-2015	03-04-2015	20d	<div><div></div></div>													
6	TESTING	06-04-2015	24-04-2015	15d	<div><div></div></div>													
7	RETROSPECTIVE	27-04-2015	29-04-2015	3d	<div><div></div></div>													

### PERT CHART:-

