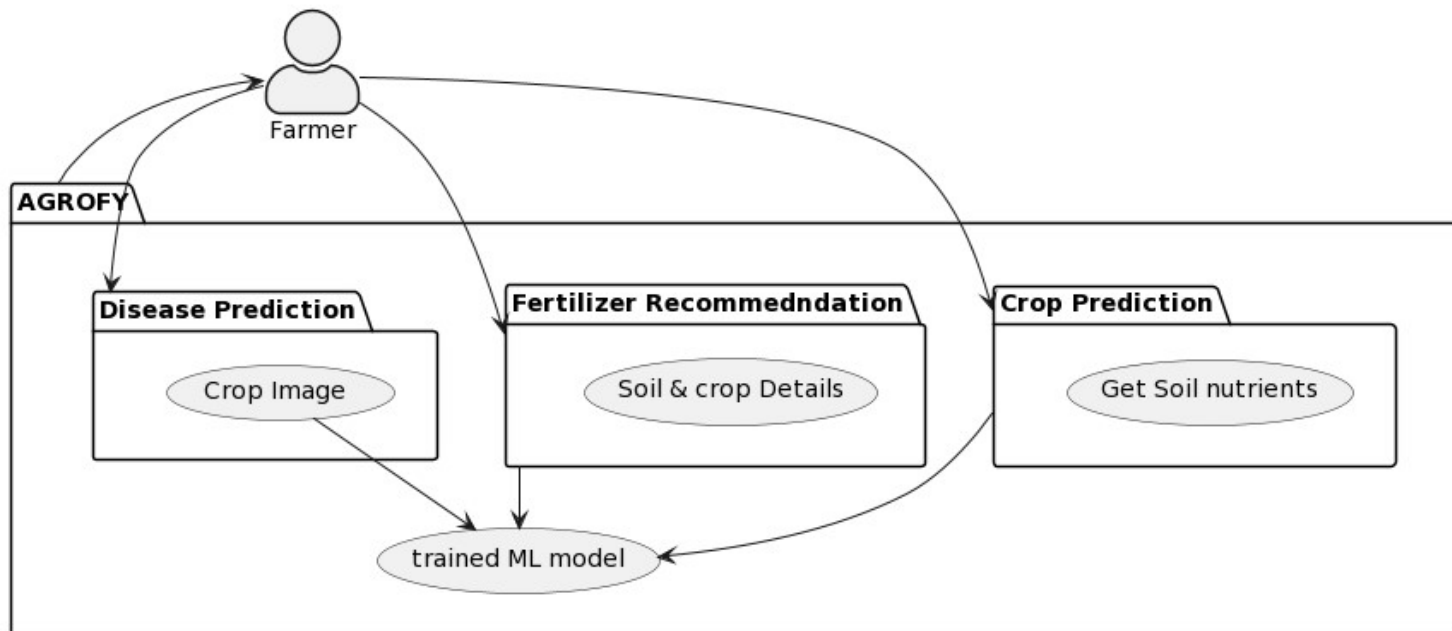


Use Cases & Test Cases

Date	07 December 2022
Team ID	PNT2022TMID53297
Project Name	Fertilizer Recommendation System for Disease Prediction

Use case Diagram

AGROFY - Usecase Diagram



Use case Text

```
@startuml
title AGROFY - Usecase Diagram
skinparam actorStyle awesome
actor "Farmer" as f
package AGROFY as a{
    package "Crop Prediction" as cp{
        usecase "Get Soil nutrients"
    }
    package "Fertilizer Recommedndation" as fr{
        usecase "Soil & crop Details"
    }
    package "Disease Prediction" as dp{
        usecase "Crop Image" as ci
    }
    usecase "trained ML model" as ml
}
f-->cp
f-->fr
f-->dp
cp-->ml
fr-->ml
ci-->ml
a--->f
@enduml
```

TEST CASES

A test case is a specification of the inputs, execution conditions, testing procedure, and expected results that define a single test to be executed to achieve a specific software testing objective, such as exercising a specific programme path or verifying compliance with a specific requirement, in software engineering. Test cases underpin deliberate rather than chaotic testing. To get the appropriate coverage of the programme being tested, a battery of test cases can be created. Formally stated test cases enable the same tests to be performed against consecutive versions of software, enabling effective and consistent regression testing.

FORMAL TEST CASES

To completely verify that all of an application's criteria are satisfied, there must be at least two test cases for each requirement: one positive test and one negative test. If a requirement is broken down into sub-requirements, each one must contain at least two test cases. A traceability matrix is widely used to keep track of the relationship between the requirement and the test. Written test cases should include a description of the functionality to be tested as well as the preparation needed to ensure that the test can be run.

INFORMAL TESTING CASES

Test cases for applications or systems that lack formal requirements might be built based on the acknowledged typical behaviour of program of a comparable class. In some testing schools, test cases are not written at all, but the activities and results are given after the exams are completed.

Hypothetical tales are used in scenario testing to assist the tester in thinking through a complicated problem or system. Typically, these circumstances are not documented in any depth.

TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
HomePage_TC_OO 1	Functional	Home Page	Verify user is able to see the homepage or not.		<ol style="list-style-type: none"> 1. Enter URL and click go 2. verify whether the user is able to see the home page. 	Enter URL and click go	User able to see the home page	Working as expected	Pass	Nil	N	—	Santhosh M
HomePage_TC_OO 2	UI	Home Page	Verify the UI elements in Home Page		<ol style="list-style-type: none"> 1. Enter URL and click go 2. Verify the UI elements in Home Page. 	Enter URL and click go	Application should show below UI elements: Home Tab & PredictTab	Working as expected	pass	Nil	N	—	Navaneethakrishnan S
PredictPage_TC_OO 3	Functional	Predict page	Verify user is able to redirect to predict page or not.		<ol style="list-style-type: none"> 1. Enter URL and click go 2. Click on Predict button 3. Verify whether the user to redirect to predict page or not. 	Click the predict button in home page	User should navigate to Predict page	Working as expected	pass	Nil	N	—	Iswarya M
Predict Page_TC_OO 4	UI	Predict page	Verify the UI elements in Predict Page		<ol style="list-style-type: none"> 1. Enter URL and click go 2. Verify the UI elements in Predict Page. 	Click the predict button and redirect to predict page	Application should show below UI elements: Dropdown List , Uploadfile Button, Predict button.	Working as expected	pass	Nil	N	—	Sabarivasan E
Predict Page_TC_OO 5	Functional	Predict page	Verify user is able to select the dropdown value or not.		<ol style="list-style-type: none"> 1. Enter URL and click go 2. Click on Predict button 3. Verify whether the user to redirect to predict page or not. 4. Verify user is able to select the dropdown value or not. 	Fruit or Vegetable	Application should show user to choose fruit or vegetable option in dropdown list.	Working as expected	pass	Nil	N	—	Iswarya M
Predict Page_TC_OO 6	Functional	Predict page	Verify user is able to upload the image or not.		<ol style="list-style-type: none"> 1. Enter URL and click go 2. Click on Predict button 3. Verify whether the user to redirect to predict page or not. 4. Verify user is able to select the dropdownvalue or not. 5. Verify user is able to upload the images or not 	Images to be Uploaded	Application should show the uploaded image.	Working as expected	pass	Nil	N	—	Sabarivasan E

PredictPage_TC_O O 7	Functional	Predict page	Verify whether the image is predicted correctly or not		<ol style="list-style-type: none">1. Enter URL and click go2. Click on Predict button3. Verify whether the user to redirect to predict page or not. Verify user is able to select the dropdownvalue or not.4. Verify user is able to upload the images or not5. Verify whether the image is predicted correctly or not	Click the Predict Button	Application shows the predicted output	Working as expected	pass	Nil	N	-	Navaneethakrishnan S
-------------------------	------------	-----------------	--	--	--	-----------------------------	---	------------------------	------	-----	---	---	----------------------