

Table of Content

Chapters	Particulars		Page no.
1	Introduction		4
	1.1	Purpose	
	1.1.1	Enhance Business Processes	4
	1.1.2	Division of fields into plots based on partner	4
	1.1.3	Maintaining Records of inputs and growth of each plot	5
	1.2	Product Scope	5
	1.3	Problem Definition	5
	1.4	Aim and goals of the system	
2	Overall Description		6
	2.1	Product Perspective/ Environment Description	6
	2.1.1	Hardware Interface/ Hardware Specification	6
	2.1.2	Software Interface/ Software Specification	6
	2.1.3	Communication Interface	6
	2.2	User Characteristics	6
	2.2.1	Owner	6
3	System Planning		7
	3.1	Software Engineering Model	7
4	System Specific Requirements		9
	4.1	Functional Requirement	9
	4.1.1	Login	9
	4.1.2	Field Master : To maintain fields and plots	9
	4.1.3	Raw Material input management	9
	4.1.4	Attendance management for labors	10
	4.1.5	Party master (both vendors and dealers)	10
	4.1.6	Invoice (Both Purchase and sales)	10

	4.1.7	Reports	10
	4.2	Non-functional Requirement	11
	4.2.1	Usability	11
	4.2.2	Security	11
	4.2.3	Performance	11
	4.2.4	Availability	11
	4.2.5	Error Handling	12
	4.2.6	Ease of Use	12
		System Analysis	5
	5.1	Use case Diagram(s)	
	5.1.1	Owner	13
	5.1.2	Farm Manager	13
	5.2	Activity Diagrams	14
	5.2.1	Activity Diagram for adding fields and plots	14
	5.2.2	Activity Diagram for adding raw materials	15
	5.2.3	Activity Diagram for adding employees	16
	5.2.4	Activity diagram for recording employee attendance	17
	5.2.5	Activity diagram for adding party	18
	5.2.6	Activity diagram for adding invoice	19
		System Design	20
	6.1	Database Design	20
	6.1.1	Data Dictionary	20
		System Implementation	24
	7.1	Screenshots	24
		Testing	33
	8.1	Test Cases	33

9	Future Enhancement	34
10	Conclusion	34
11	Bibliography	34

1.Introduction

Agriculture sector is most unmanaged sector in the country. There is little or no digitization in this area. This is so because of the various processes which change from crops to crops and places to places. The current system focuses on every little aspect to manage day to day farm activities. There are various instances where, due to lack of proper data farmers tend to make wrong decisions in matter of raw materials or the normal accounts. Sometimes due to lack of previous data farmers tend to utilize more than required pesticides or due to lack in management. This system will enable farmer to divide his field area into plots of various crops and make the record of each and every raw material used and even maintain the ad-hoc employee register. Attendance records can be managed and various reports for the employees can be generated. Invoicing is the important aspect for any farm, majorly farmers are occupied much in their daily activities that they are not able to manage their accounts. This system enables them to maintain their dealers and vendors and do the invoicing accordingly. This would help to generate various kinds of reports.

1.1 Purpose

The advancement in Information Technology and internet penetration has greatly enhanced various business processes and communication between companies (services provider) and their customers of which car rental industry is not left out. Agrologix is developed to provide the following services.

1.1.1 Enhance Business Processes

To lead the farmer to more productive activities instead of occupying themselves in the routine which gradually decreases their efficiency. The power of information technology is new in the area and would help a lot in making digital farms

1.1.2 Division of fields into plots based on partner

When there is large area of field its essential for the farmer to divide its fields into various plots and grow different types of the crops in each plot. Every plot has its own kind of specific requirements like pesticides or the raw materials. Even for the accounting purposes

this division into plots plays an important role. As this plots are given to various partners for taking care and they get a commission from it yields.

1.1.3 Maintaining Records of inputs and growth of each plot

Whenever we deal with various kinds of crops which have their own life cycle, it's very important to know when each and every raw material is used. This records are need to be maintained in order to avoid any over dosage. Yields of every plot can be recorded in order to decide right amount of commission to the plot partner.

1.2 Product Scope

The scope of this system is very wide not because it's physical existence but it's flexibility to be used in various types of crops.

- This system can be used anywhere where farm activities are carried out.
- Can be used for any crops
- Can be used by owner of the farms. Other users can be manageable as per requirements of the owner.
- This is a desktop based application which means it available 24x7 except the technical errors.

1.3 Problem Definition

Agriculture is a sector which is still untouched by IT Sector. Farmers used to maintain records manually and would cause them problem in making decision in various matters. The delay in the process would make the huge losses to them. They would need maintain invoicing manually and even their employee payrolls. The yields of the crop were to be measured on the daily basis which increases the chances of the error.

1.4 Aims and goals of the system

- To produce a desktop based application which is used by farmers to record various activities.
- To help making quick and better decisions.
- To maintain accounts of the sales and purchases.

2.Overall Description

2.1 Product Perspective/ Environment Description

2.1.1 Hardware Interface/Hardware Specification

- Any windows machine (Preferably windows 10)
- Minimum 4GB Ram Required
- Laser Jet Printer

2.1.2 Software Interface/ Software Specification

- .Net Framework installed
- Microsoft SQL Server Management Studio
- Printer Drivers

2.1.3 Communication Interface

- printers

2.2 User Characteristics

- Farmer/owner:
 - (a) Can login to the system
 - (b) Maintain fields and the plots
 - (c) Maintain raw materials and their categories

- (d) Maintain the labors/Employees
- (e) Record the attendance of the labors
- (f) Records yields from the fields
- (g) Maintain the party master
- (h) Invoicing: Sales and Purchase

3. System Planning

3.1 Software Engineering Model

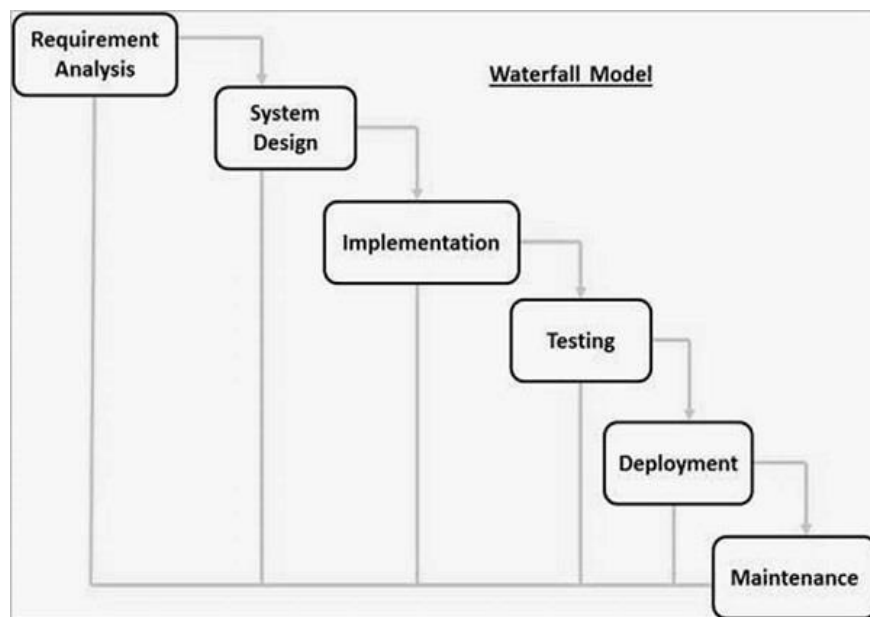
Waterfall Model

We have followed this model step is...

1. Requirement Analysis
2. Design
3. Implementation
4. Unit Testing
5. Integration and Testing
6. Deployment and Maintenance

- The waterfall model begins with the system. It gathers requirements from the customer. At the end SRS and software project plan are produced. The SRS acts as a contract between the customer and the developer. In the design phase, the SRS is transformed into design which is suitable for implementation in a programming language. First preliminary design is made then detailed design is made. Initial test plan are produced at the end of this phase
- During the implementation phase.

- In unit testing small modules are tested in isolation and the overhead code is written for handling communication amongst these modules.
- After implementation and unit testing, the modules are integrated to form a complete system. Integration and testing are carried out to verify the functionality of the system.



4.System Specific Requirements

4.1 Functional Requirements

These are statements of services the system should provide, how the system should react to particular inputs, and how the system should behave in particular situations. It specifies the application functionality that the developers must build into the product to enable users to accomplish their tasks

4.1.1 Login

- This should allow the owner to execute or farm executive to login to the system.

4.1.2 Field Master: To maintain fields and plots

- This should allow the owner/Farm executive to manage the number of which the owner has.
- This should allow the owner/Farm executive to manage the number of plots that are divided from fields.
- This should allow to manage the which crops are grown into which plot.
- This should allow to manage who is the partner of the particular plot.

4.1.3 Raw Material input management

- This should allow the owner/Farm executive to manage categories of raw material.
- This should allow the owner/Farm executive to manage raw material product category wise.
- This should allow the owner/Farm executive to manage price and unit of every raw material product.
- This should allow the owner/Farm executive manage the input given to each plot.

4.1.4 Attendance management for labors

- This should allow the owner/Farm executive to manage the extra labors they hire temporarily
- This should allow the owner/Farm executive to manage the attendance for each labor everyday field and plot wise.

4.1.5 Party master (both vendors and dealers)

- This should allow the owner/Farm executive to manage their vendors(Purchase).
- This should allow the owner/Farm executive to manage their dealers(Sales).

4.1.6 Invoice (Both Purchase and sales)

- This should allow the owner/Farm executive to manage purchase of raw materials
- This should allow the owner/Farm executive to manage sales of crop yields
- This should allow manager to maintain a total record of crop yield in a plot.

4.1.7 Reports

- This should allow the owner/Farm executive to generate labor wise attendance report.
- This should allow the owner/Farm executive to generate plot wise labor attendance report.
- This should allow the owner/Farm executive to generate total stock report.
- This should allow the owner/Farm executive to generate sales report.
- This should allow the owner/Farm executive to generate purchase report.
- This should allow the owner/Farm executive to generate plot wise input report.

4.2 Non Functional Requirements

Non-functional requirements, as the name suggests, are requirements that are not directly concerned with the specific services delivered by the system to its users. They may relate to emergent system properties such as reliability, response time, and store occupancy. Alternatively, they may define constraints on the system implementation such as the capabilities of I/O devices or the data representations used in interfaces with other systems. Non-functional requirements, such as performance, security, or availability, usually specify or constrain characteristics of the system as a whole.

4.2.1 Usability

The system provides a help and support menu in all interfaces for the user to interact with the system. The user can use the system by reading help and support.

4.2.2 Security

The system provides username and password to prevent the system from unauthorized access. The staffs' password must be greater than eight characters. The subsystem should provide a high level of security and integrity of the data held by the system, only authorized personnel of the company can gain access to the company's secured page on the system; and only users with valid password and username can login to view user's page.

4.2.3 Performance

The system response time for every instruction conducted by the user must not exceed more than a minimum of 10 seconds. The system should have high performance rate when executing user's input and should be able to provide response within a short time span usually 50 second for highly complicated task and 20 to 25 seconds for less complicated task.

4.2.4 Availability

The system should always be available for access at 24 hours, 7 days a week. Also in the occurrence of any major system malfunctioning, the system should be available in 1 to 2 working days, so that business process is not severely affected.

4.2.5 Ease of use

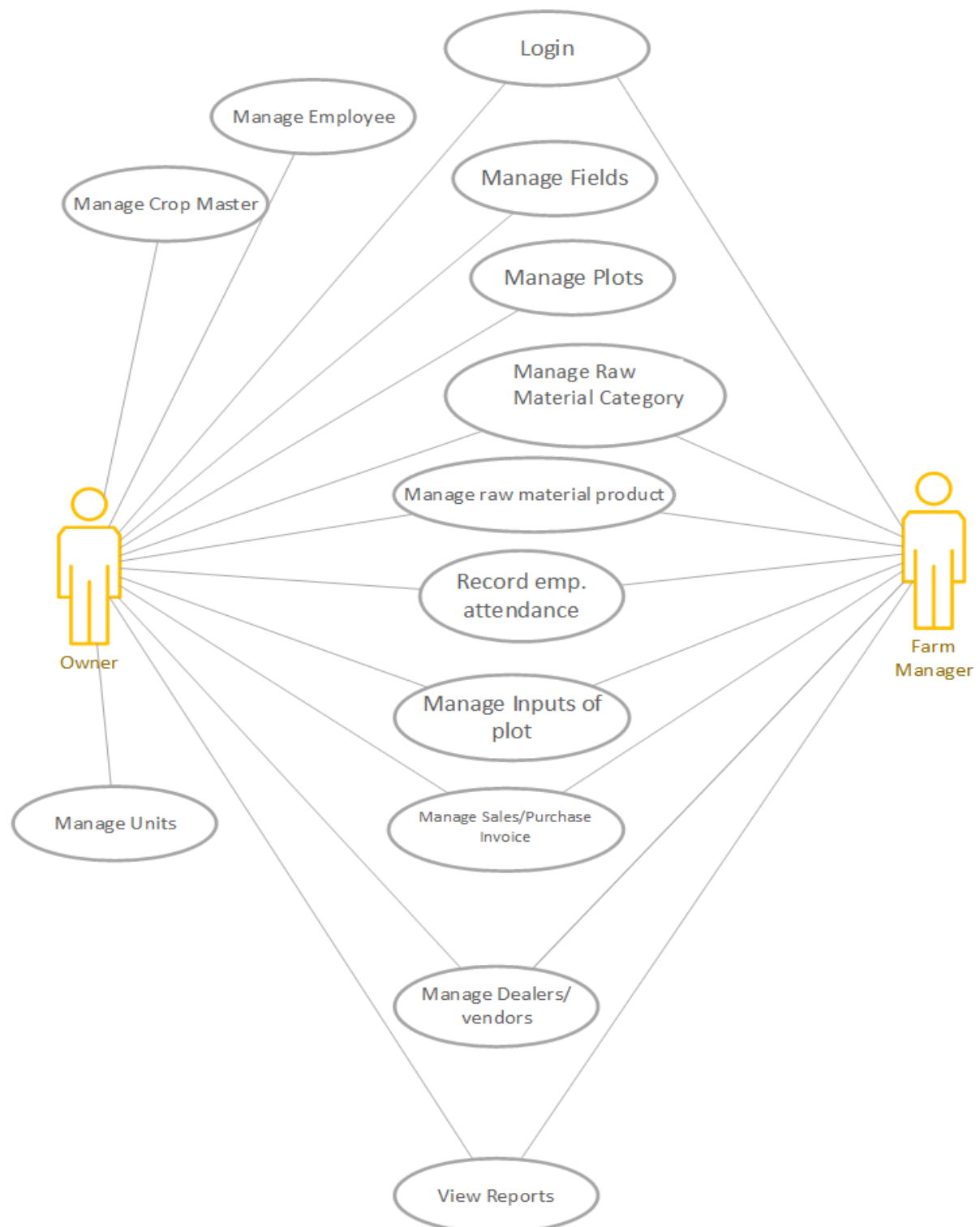
Considered the level of knowledge possessed by the users of this system, a simple but quality user interface should be developed to make it easy to understand.

4.2.6 Error Handling

Error should be considerably minimized and an appropriate error message that guides the user to recover from an error should be provided. Validation of user's input is highly essential. Also the standard time taken to recover from an error should be 15 to 20 seconds.

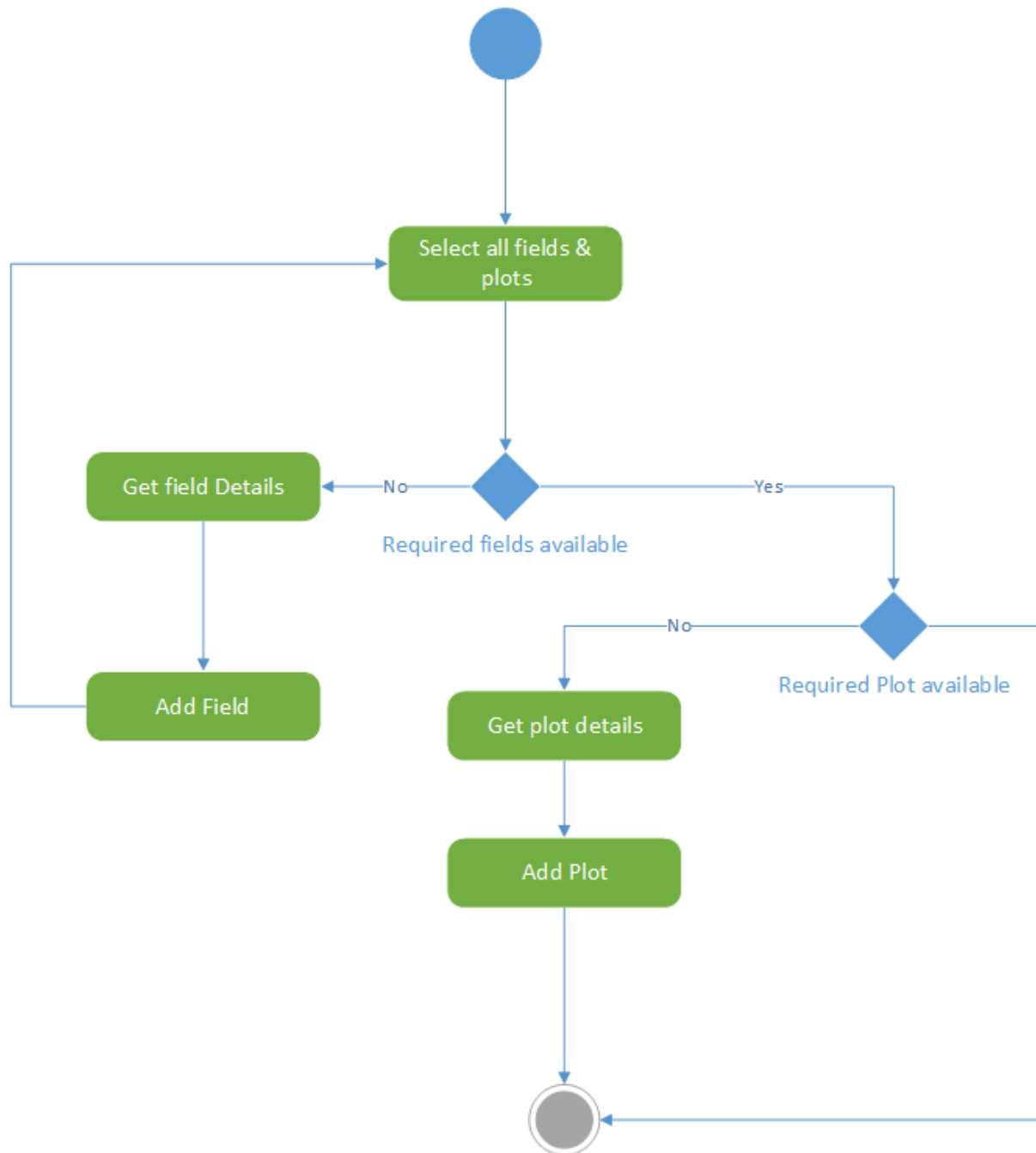
5. System Analysis

5.1 Use Case Diagrams

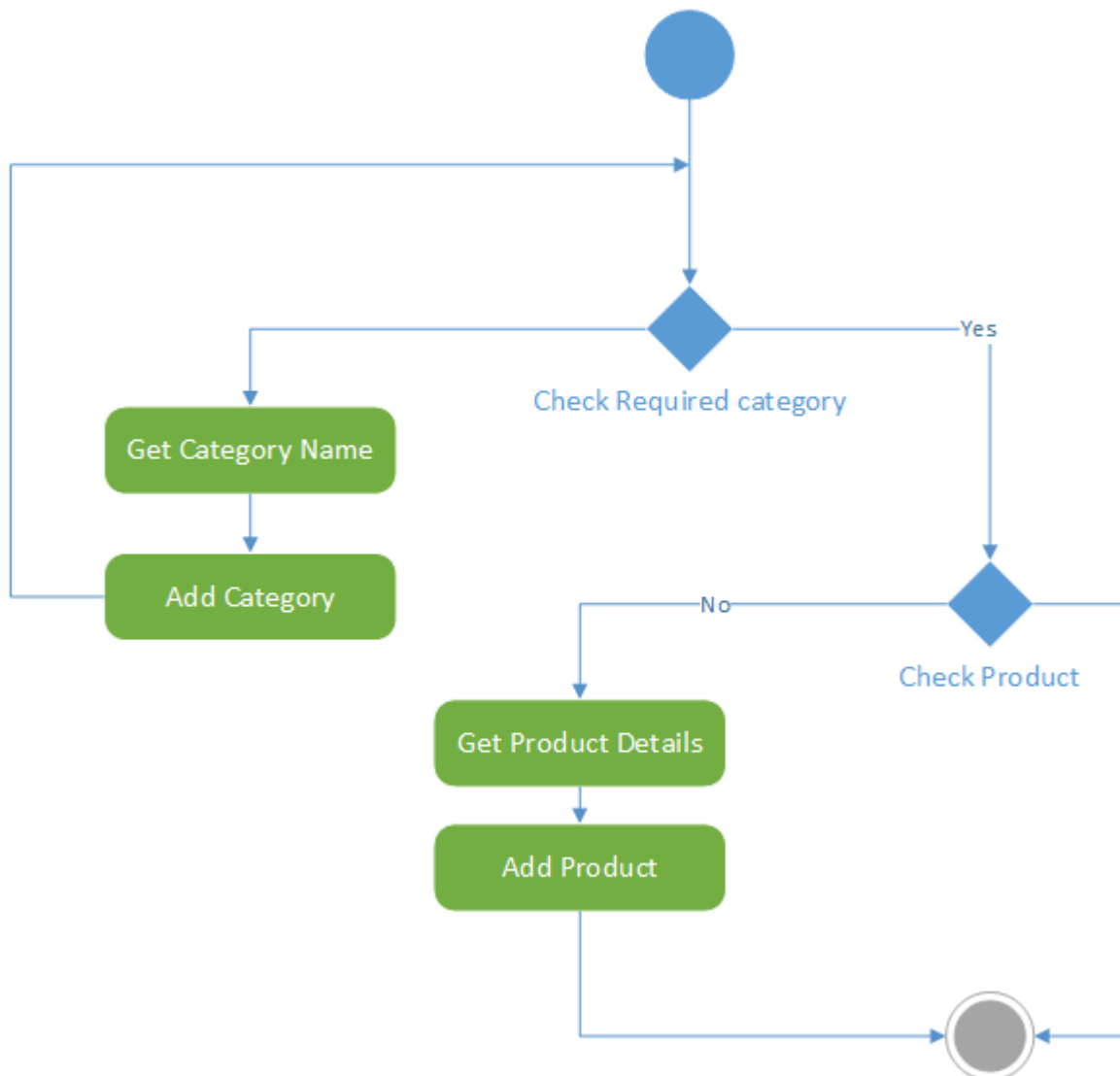


5.2 Activity Diagrams

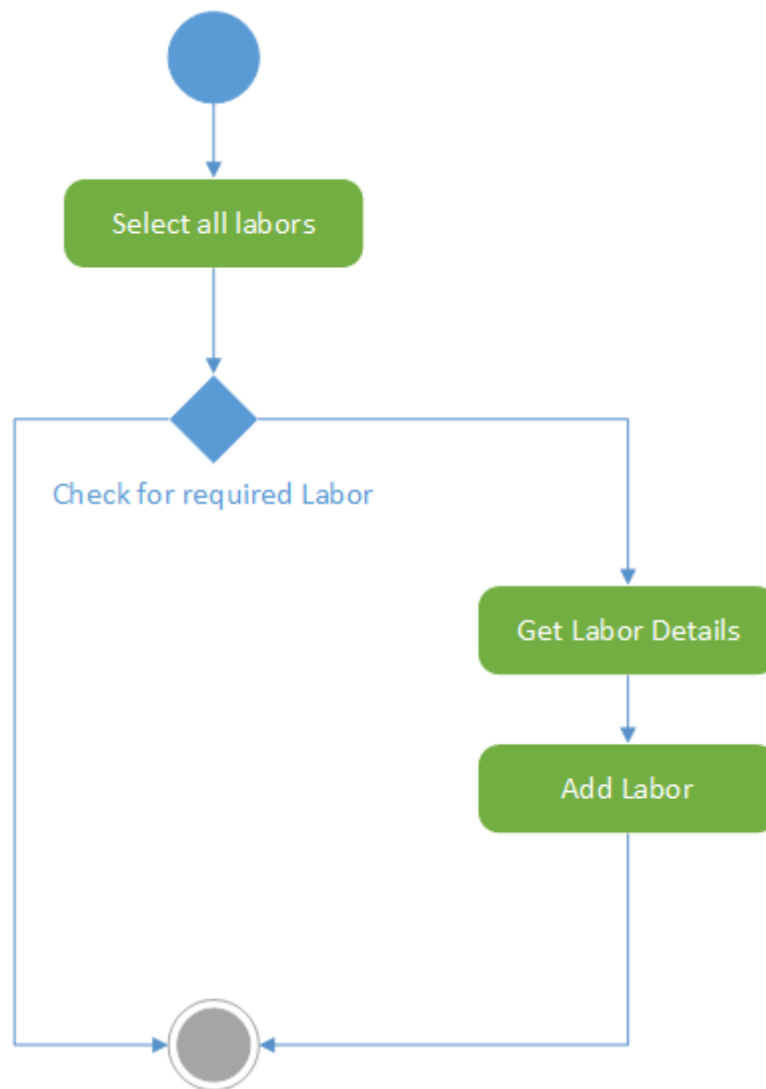
5.1.1 Activity Diagram for adding fields and plots



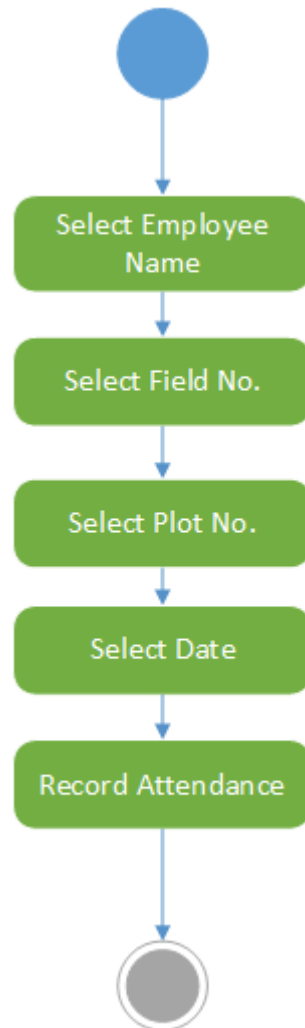
5.1.2 Activity Diagram for adding raw materials



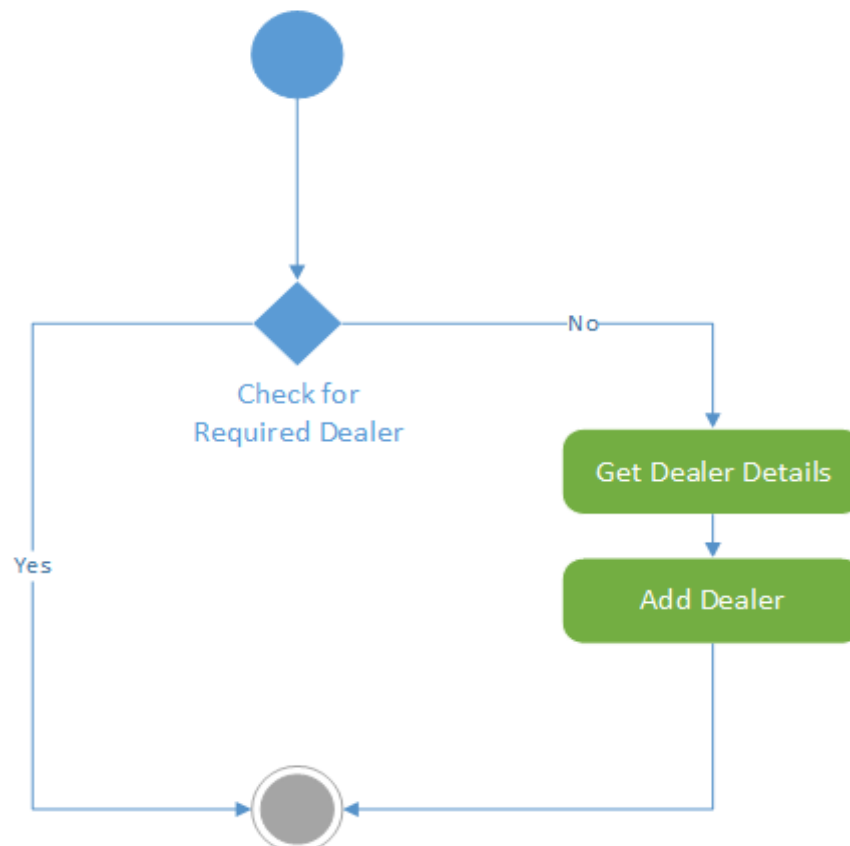
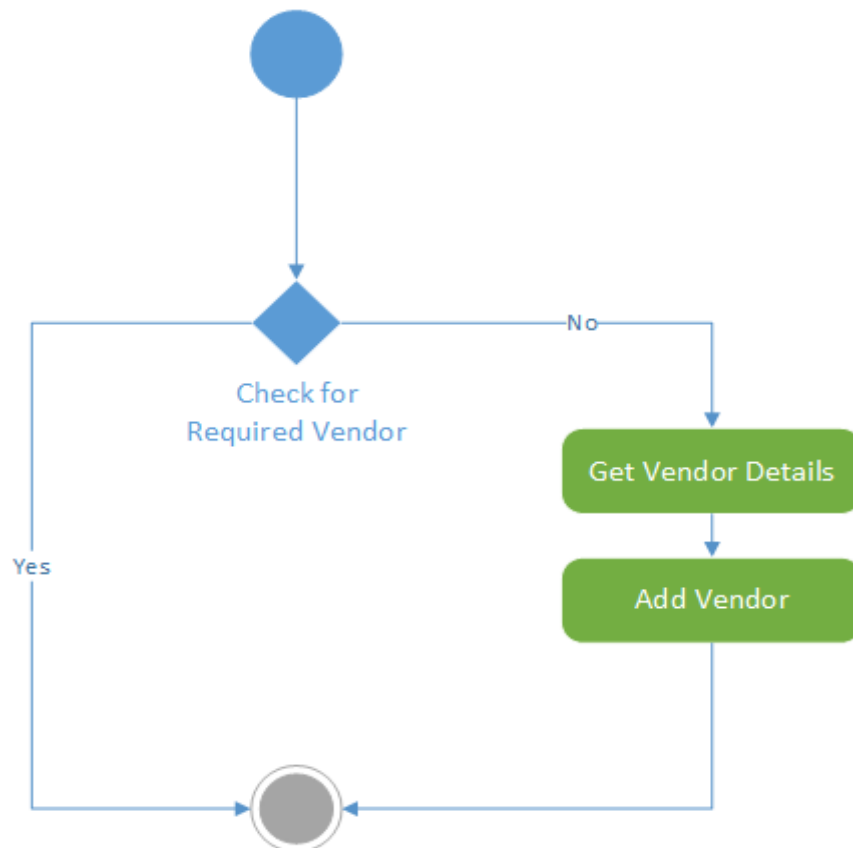
5.1.3 Activity Diagram for adding Labor



5.1.4 Activity diagram for recording labor attendance



5.1.5 diagram for adding Party



5.1.6 Activity diagram for adding invoice



6.System Design

6.1 Database Design

6.1.1 Data Dictionary

Table Name :	Field Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
F_number	Int	2	Primary Key	Stores the field No.
F_Area	decimal		Not Null	Stores field area

Table Name :	Plot Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
f_number	int	2	FK-Field Master	Stores field no.
plotid	int	2	Primary Key	Stores Plot no.
plotarea	decimal		not null	Stores Plot area
Partner Name	Varchar	25	not null	Stores partner name
crop type	varchar		FK-Crop Master	Stores crop type

Table Name :	Crop Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
cropid	Int	2	Primary Key	Stores the crop ID
crop_name	varchar	20	not null unique	Stores crop name

Table Name :	rm_purchase Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
billid	Int	6	Primary Key	Stores Bill ID
date	Date		not null	Stores date
vendorid	Int	3	FK-vendor Master	Stores VendorID
vendorbill_no	Int	6	not null	Stores Vendor Bill No
vendorname	varchar	25	FK-vendor Master	Store name of vendor
rm_cat	varchar	20	FK-rm_category Master	Store RM Category
rm_product	varchar	20	not null	Stores RM Product name
rm_unit	varchar	15	FK-rm_product Master	Stores unit of product
qty	int	10	not null	Stores quantity
amount	int	10	not null	Stores final amount of product

Table Name :	emp Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
empid	Int	5	Primary Key	Stores employee ID
empname	varchar	20	not null	Stores Emp. Name
empphone	varchar	10	unique	Stores Emp. Phone

Table Name :	crop_stock			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
stid	Int	10	Primary Key	Stores crop ID
fieldno	Int	2	FK-Field master	Stores Field No.
plotno	int	2	FK-Plot Master	Stores Plot No.
crop_name	varchar	20	FK-Crop Master	Stores Crop Name
qty	Int	10	not null	Stores Quantity

Table Name :	Sales			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
invoiceid	int	6	Primary Key	Stores Invoice ID
Date	date		not null	Stores Date
Dealer_name	varchar	25	FK-Dealer Master	Stores Dealer Name
dealerid	int	3	FK-Dealer Master	Stores Dealer ID
fieldno	int	2	FK-Field master	Stores Field No
plotno	int	2	FK-Plot Master	Stores Plot No
qty	int	10	not null	Stores Quantity
amount	int	10	not null	Stores Amount

Table Name :	Unit Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
unit_id	int		Primary Key	Stores Unique ID
unit	varchar(20)		Not Null,Unique	Stores Unit name

Table Name :	dealer Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
dealerid	int	3	Primary key	Stores Dealer ID
dealer_name	varchar	25	Not null	Stores Name of dealer
dealer_city	varchar	15	Not null	Stores city of dealer
dealer_phone	varchar	10	Not null	Stores phone number of dealer

Table Name :	empatt Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
aid	int	5	Primary Key	Stores unique ID
empid	int	5	FK-emp Master	Stores Employee ID
empname	varchar	20	FK-emp Master	Stores Employee Name
fieldno	int	2	FK-Field master	Stores field no
plotno	int	2	FK-Plot Master	Stores plot no

Table Name :	rm_plotinput			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
inputid	int	2	Primary Key	Stores unique ID
fieldno	int	2	FK-Field Master	Stores field no
plotno	int	2	FK-Plot Master	Stores Plot No
rmcat	vachar	20	FK-rm_category Master	Stores RM Category
rm_product	vachar	20	FK-rm_product Master	Stores RM Product
qty	int	10	not null	Stores Quantity
unit	vachar	10	not null	Stores Unit

Table Name :	vendor Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
vendorid	int	3	Primary key	Stores unique ID
vendor_name	vachar	25	Not null	Stores vendor name
vendor_city	vachar	15	Not null	Stores vendor city
vendor_phone	vachar	10	Not null	Stores vendor phone number

Table Name :	rm_product Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
rmp_id	int	3	Primary Key	Stores Unique ID
rm_cat	vachar	20	FK-rm_category Master	Stores RM Category
rm_product	vachar	20	not null unique	Stores RM Product name
rm_unit	vachar	15	unique	Stores Unit Product
price	decimal		not null	Store Price

Table Name :	rm_category Master			
<u>Attribute Name</u>	<u>Data Type</u>	<u>Size</u>	<u>Constraint</u>	<u>Description</u>
rmid	int	3	Primary Key	Stores Unique ID
rmcat	vachar	20	Not null unique	Stores RM Category

7.System Implementation

7.1 Screen shots



The screenshot displays the "Field Master" window, which is divided into two main sections: "Field and Plot Master" and a data entry section. The "Field and Plot Master" section contains input fields for "Field No:", "Field Area:", "Plot No:", "Plot Area:", "Partner Name:", and "Crop Type:", each followed by an "Add" button. Below these fields are two tables. The first table, titled "Field and Plot Master", lists fields with their numbers and areas. The second table, titled "Field and Plot Master", lists plots with their IDs and field numbers. Both tables have a "View" button below them. A "Back" button is located at the bottom right of the window.

	f_number	f_area
▶	1	21.00
	2	250.00
	3	252.00
*		

	plotid	f_number
▶	1	1
	2	2
	3	2
	4	3
*		

rmplotwiseinput

Raw Material Plot Wise Input

Field No:

Plot No:

RM Category:

RM Product:

RM Qty: Unit

crop_stock

Crop Stck Manager

Field No:

Plot No:

Crop Type:

Qty(Kg)

unit_master

Unit:

	unit
▶	KG
	ML
*	

plotwise_rm_rpt

Raw Material Input Record

Field No: ▼
Plot No: ▼

	mcat	rm_productname	qty
▶	Fertilizer	Abc Ferti	1
*			

empmaster

Labor Manager

Employee Name:

Employee Phone:

Add

	empid	empname	empp
▶	1	Viraj	84699
*			

<

>

Back

emp_att

Labor Attendance

Employee Name:

Field No:

Plot No:

Record Attendance

View Reports

Back

rm_purchase

Raw Material Purchase Invoice Log

	vendor_billno	date	vendor_billno1	vendormame	total
▶	101	26-04-2020	101	Viraj PVT	2952
	102	26-04-2020	102	Viraj PVT	948
	301	26-04-2020	301	Raj PVT	1422
*					

btn_back

sales_invoice

Sales Invoice

Date: 03 May 2020

Invoice ID: 1

Dealer ID:

Vendor ID: Viraj PVT

Dealer Name:

Invoice ID:

Field No:

Plot No:

Crop Type:

Qty:

Price:

Amount:

Back

Save

See Log

sales_log

Sales Invoice Log

	invoiceid	date	dealer_name	total
▶	1	27-04-2020	Viraj PVT	880
	2	27-04-2020	Raj Pvt	1504
	3	27-04-2020	Viraj PVT	1265
	4	27-04-2020	Viraj PVT	2990
	5	27-04-2020	Raj Pvt	825
*				

Back

Vendor Details

Vendor Details

Vendor Name:

Vendor City:

Vendor Phone:

Add Vendor

	vendorid	vendor_name
▶	1	Viraj PVT
	3	Raj PVT
*		

Back

dealer_details

Dealer Manager

Dealer Name:

Dealer City:

Dealer Phone:

Add Dealer

	dealerid	dealer_name
▶	1	Viraj PVT
	2	2
	3	Raj Pvt
*		

< >

Back

8. Testing

8.1 Test cases

1	All files are empty	Error message: *indicates compulsory field*	Error message: *indicates compulsory field*	Pass
2	Email	Error message: Invalid Email-address	Email Accepted	Fail
3	Password and confirm password	Error message: Both Password does not match	Error message: Both Password does not match	Pass
4	Login	Login to the system should be try with the login assigned by the admin and the correct password	Login should be successful and the user should enter into the system	Fail
		The System give an error and denied from the Login.	Login should fail with an error 'Invalid Details'	Pass
5	User	Login should be allow and admin get Admin home page	Login successfully and admin get its admin home page	Pass
		Login should be allow and Travel admin get Travel admin home page	Login successfully and Travel admin get its Travel admin home page	Pass
		Login should be allowing and User get Visitor Side User page.	Login successfully and User gets its user page.	Pass
6	Validation Test cases	Pre-define format must be required in control	System give error to enter valid input	Pass
		Enter data in a compulsory field	Data must be field in compulsory field	Pass

		with required field validations.	otherwise its messages are displayed.	
--	--	----------------------------------	---------------------------------------	--

9.Future Enhancements

We believe that every system has a chance of improvement and we have already found out and set up the next milestone to our system

- We would like to add a feature which would advise the farmer that which raw material is required now according to schedule feed inside.
- We would like to enhance our accounting system more to accommodate ledgers and profit & Loss statements
- We would like to upgrade our system to a better UI system and more personalization to the system.

10.Conclusions

Agriculture in today's world have reached a new height. There are modern techniques which truly takes this business global. Every activity related to agriculture is still physical and manual from farms to customers, everything is unorganized. Farmers distract themselves from productive routine to this repetitive work. So this AGROLOGIX would help farmers to manager their farm activities with the power of information technology. When we talk about information technology which talk about smart systems which smartly stores the information and process them to give output. Here as a result of this system farmer is powered to make the right decisions and get rid of the tedious task which takes time.

11.Bibliography

1. Yogesh Singh, Ruchika Malhotra, "Object-Oriented Software Engineering" – PHI.
2. Mall R., Fundamental of Software Engineering, PHI