



American International University-Bangladesh (AIUB)

Department of Computer Science

Faculty of Science & Technology (FST)

Automatic Irrigation System

A Software Engineering Project Submitted

By

Semester: Summer_21_22		Section:	Group Number:	
SN	Student Name	Student ID	Contribution (CO1+CO2)	Individual Marks
01	MD. Apon Riaz Talukder	20-42905-1		
02	MD. Borhan Uddin	20-44002-2		
03	Limia Sadina Sathi	20-43851-2		
04	Sha Mohammad Fahim	20-43974-2		
05	Wasif Hassan Joarder	20-43970-2		
06	Shahriar Alam	20-42226-1		

1. PROJECT PROPOSAL:

1.1 Background to the Problem

Irrigation is the process by which we supply water to land or crops to help better growth of different types of crops. In our country our farmers are using the traditional method for watering the crops and monitoring the growth. We are here with a smart solution for our

farmers-called smart irrigation. To build this project here Smart irrigation technology uses weather data or soil moisture data to determine the irrigation need of the landscape.

1.2 Solution to the Problem

Nowadays, Science and Innovation make our life faster and easier. Bangladesh is an agricultural Country but from the very beginning our farmers are using the traditional method in their farming. But it's high time to introduce technology in this sector. From this point of view, we are going to provide a smart solution. The goal of the project is to design an automatic irrigation system monitoring for the farmers. In This Plant Watering System, Soil Moisture Sensor checks the moisture level in the soil and if moisture level is low then Arduino switches on a water pump to provide water to the plant. Water pump gets automatically off when system finds enough moisture in the soil. This system will be very sustainable and useful in Farms, gardens, home etc. This system will be completely automated and there is no need for any human intervention. After investigating we found that farmers are facing problem to keep the moisture level of the soil good for a long time, how and when the water is need for their crops, they don't know this thing well. This system will be the Proper solution of their problem. The system will fully automated, cost friendly and easy to use. The system will be going to help farmers to produce more crops, reduce their hamper and above all they can take the necessary steps by getting message alert. The overall system will be going to simulated by using proteas 8 professional simulation software.

Software Engineering following:

Waterfall Model, V-Model, Photo-----Linear Sequential Model

Evolutionary Development Model, Incremental Development--Parallel Model

XP, Scrum, DSDM, FDD ----- Iterative Model (**Agile Method**)

Waterfall Model:

In waterfall model follow linear sequential process. It's checked every step at a time no backtracking. In this model we need to well understand about our project. In this project if fail any step for any kind of reason than need to full project stop and start from first.

V-Model:

In v-model also follow linear sequential process. It is extension of water fall model. V-Model have no backtracking for this reason if any step system fails full project fail so start again from first. V-

Model is an extension of the waterfall model, and it is based on association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle there is a directly associated testing phase. This is a highly disciplined model and next phase starts only after completion of the previous phase.

Evolutionary Development:

Exploratory development: Objective is to work with customers and to evolve a final system from an initial outline specification. Should start with well-understood requirements

Throw-away prototyping: Objective is to understand the system requirements. Should start with poorly understood requirements

Incremental Development:

In this project the development and delivery is broken down into increments with each increment delivering part of the required functionality (SPIRAL).

The requirements are relatively certain but there are many complexities that leads to frequent changes.

User requirements are prioritized and the highest priority requirements are included in early increments

Once the development of an increment is started, the requirements are frozen though requirements for later increments can continue to evolve.

Agile Method (XP/SCRUM/FDD/DSDM):

Agility is the ability to create and respond to change in order to profit in a turbulent business environment.

In this Agile process each iteration is a self-contained, mini project with activities that span requirements analysis, design, implementation, and test Leads to an iteration release

(Which may be only an internal release) that integrates all software across the team and is a growing and evolving subset of the final system the purpose of having short iterations is so that feedback

from iterations

SCRUM: This phase is entered when an agreement has been made such as the requirements are completed. If a task is not well defined, estimating project costs and time will not be accurate. In such a case, the task can be spread over several sprints. If the team members are not committed, the project will either never complete or fails. Iteration length is 2-4 weeks.

DSDM: In Dynamic Systems Development Method (DSDM), if speed development needed it may result in a poorly designed product. Working with this model need more experience development staffs. Besides strong project management and control required.

FDD: In Feature Driven Development (FDD), it promotes individual code ownership as opposed to shared ownership. Iterations are not as well defined by the process as other agile methodologies.

Process Model We Choose:

Our project matches with Extreme Programming (XP).

1. The client and development team work together in planning the product.
2. Releases must occur as frequently as possible to gain plenty of feedback, which is best achieved if releases are small in terms of required new functionality.
3. A system metaphor makes it easier to explain the product to someone else; it can be an analogy to describe the product to someone who is not technical.
4. Consider using the simplest design possible to meet the needs of the client.
5. it is a refactoring method.
6. It is also a pairable process.

Why we choose Extreme Programming (XP) process:

XP (extreme programming) are better fit our project (Automatic irrigation system). In this iterative agile method are useful timing and maintaining every step for project.

It maintains simple design testing phase and improving communication simplifying, adding flexibility. There is a good processing phase is pair programming.

Pair Programming: Pair programming phase two people or our teammate can write code at one computer.

One programmer, the driver, has control of the keyboard/mouse and actively implements the program. The other programmer, the observer, continuously observes the work of the driver to identify tactical defects (syntactic, spelling, etc.) and also thinks

strategically about the direction of the work. Two programmers can brainstorm any challenging problem. Because they periodically switch roles.

For this XP model phase pair programming is helpful for complete project more faster and programmer got free time to think another way its also reduce project time.

Extreme Programming (XP) process:

1.Exploration Phase: The customers write out the story cards that they wish to be included in the first release

At the same time the project team familiarize themselves with the tools, technology and practices they will be using in the project.

2. The Planning: In this process model the client and development team can work together. This entails a larger planning session at project initiation and smaller sessions at each iteration

2. Small Release: The whole project is not required at the same time. It can be divided into different iteration and after completing the iteration sequentially, it can be delivered part by part.

3. Simple Design: Simple Design is a methodology where the rule is to keep things, as the name suggests, simple. It is focus on the simplest design that works to satisfy the client's needs.

4. System Metaphor: A system metaphor makes it easier to explain the product to someone else; it can be an analogy to describe the product to someone who is not technical. A metaphor can also describe the product implementation to someone more technical

5. Continuous Testing: After each iteration, testing is required to implement. This focuses efforts first on understanding what is required, making the user or programmatic interface to it simple, and preparing suitable tests to verify that the required behavior has been achieved.

6. Refactoring: The aim of refactoring is to improve the design in small steps, as needed, to allow new product requirements to be added more easily, thus adapting to change.

7. Pair Programming: In this, two developers work side-by-side at one computer to work on a single task. One developer writes some code, and after completion, another developer reviews it. In pair programming, the code is reviewed all the time by always having another pair of eyes to consider each edit. Choosing Extreme Programming (XP) instead of another model.

REQUIREMENT ANALYSIS

Functional Requirements

1. Create Account/Registration

1. Open the app.
2. The farmer will fill the registration form with his/her full name and NID, Phone number, Address along with username and a 6-digit password.
3. Farmer email address must be provided in the form.
4. After submitting the form, the app will send a confirmation mail to the farmer mobile phone.
5. After opening the confirmation SMS, new profile is created for the farmer. The farmer can now login.

Precondition: App must be installed, and location permission should be granted

2. Sign-up page/Log in

1. The software shall allow users to login with their given username and password.
2. The login credentials (username and password) will be verified with database records.
3. If the login successful, the home page of the user account will be displayed.
4. If the username and/or password has been inserted wrong, the random verification code will be generated and sent to the user's phone number by the system to retry login.

Precondition: The user should be registered.

3. Farmer information

1. The software shall allow the farmer to access their profile after logging in.
2. Farmer can change their username, user ID, password, skillset, preferred work by accessing edit information tab.
3. If change is made, the database information is updated.

Precondition: User must be registered and logged in.

4. Home page

1. In home page every section is available ongoingly. The company or authority can change the functionalities.
2. Farmer can search anything whatever he wants from search bar.

Precondition: App must be Updated.

5. Land information

1. Total area of farmer mention here.
2. Location of the farmer land is also selected.
3. Soil type must be selected.
4. User can change the total area and if soil name if is it changed.

Precondition: The location must be set in system.

6. Check Moisture

1. The farmer checks the present weather of that area by using google.
2. Also check the water level of soil to see the water is need or not.
3. If water is needed farmer can on or of the motor.

Precondition: Browser should relate to system and location must be on.

7. Water level

1. The farmer can check the humidity, soil moisture and soil temperature.
2. Or system can check this things and update farmer by sending SMS.
3. Then farmer can make decision water is need or not.

Precondition: The sensor is work properly.

8. Motor on/off

1. The farmer can on or off the motor manually or system can do it Automatically by checking weather or water level of field.

9. Support

1. Here user can find FAQ.

2. If face any problem between using this app user can take help easily.
3. And if he wants, he also contacts the authority.

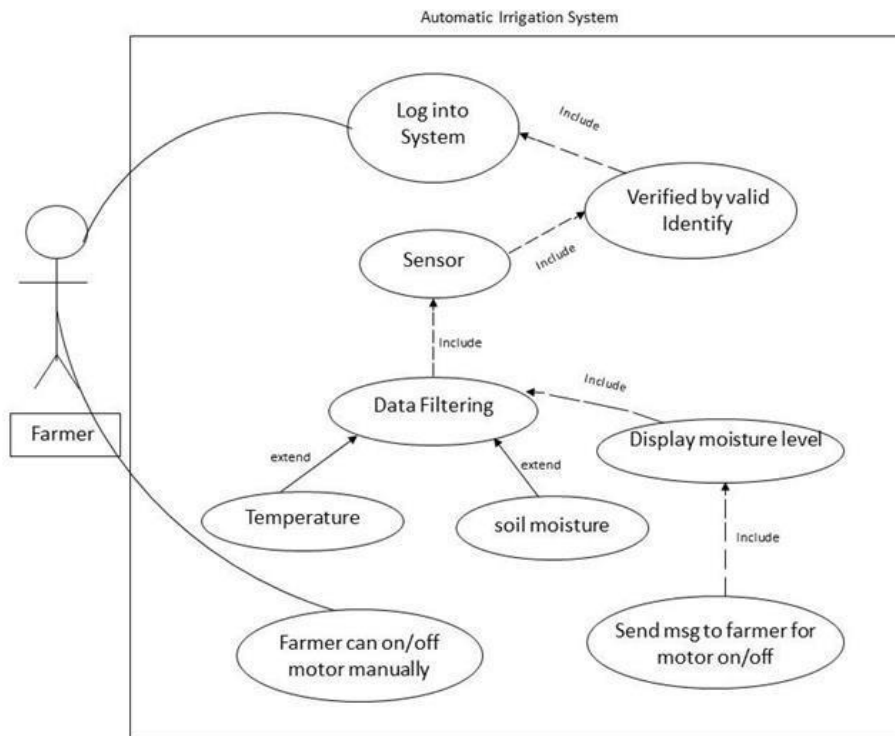
Precondition: Information must have in system.

10. About

1. Farmer can find here the authority page information.
2. Description and Additional information.
3. Here also mention terms and condition.

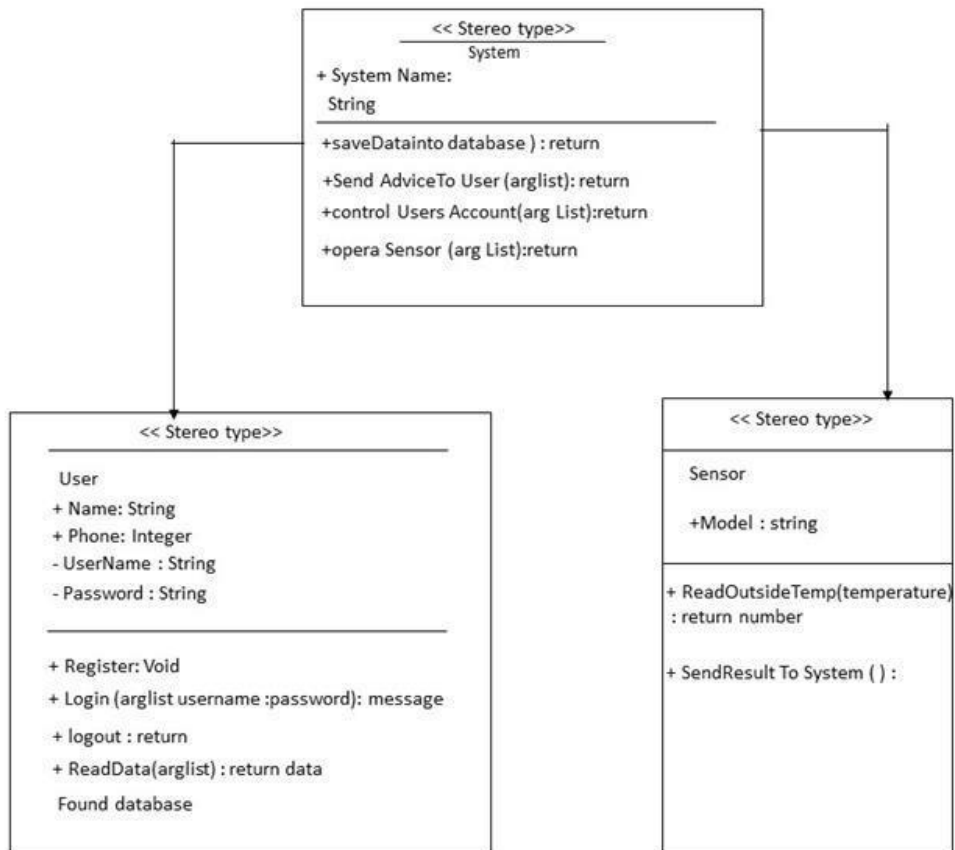
Precondition: Authority must write this thing before launch.

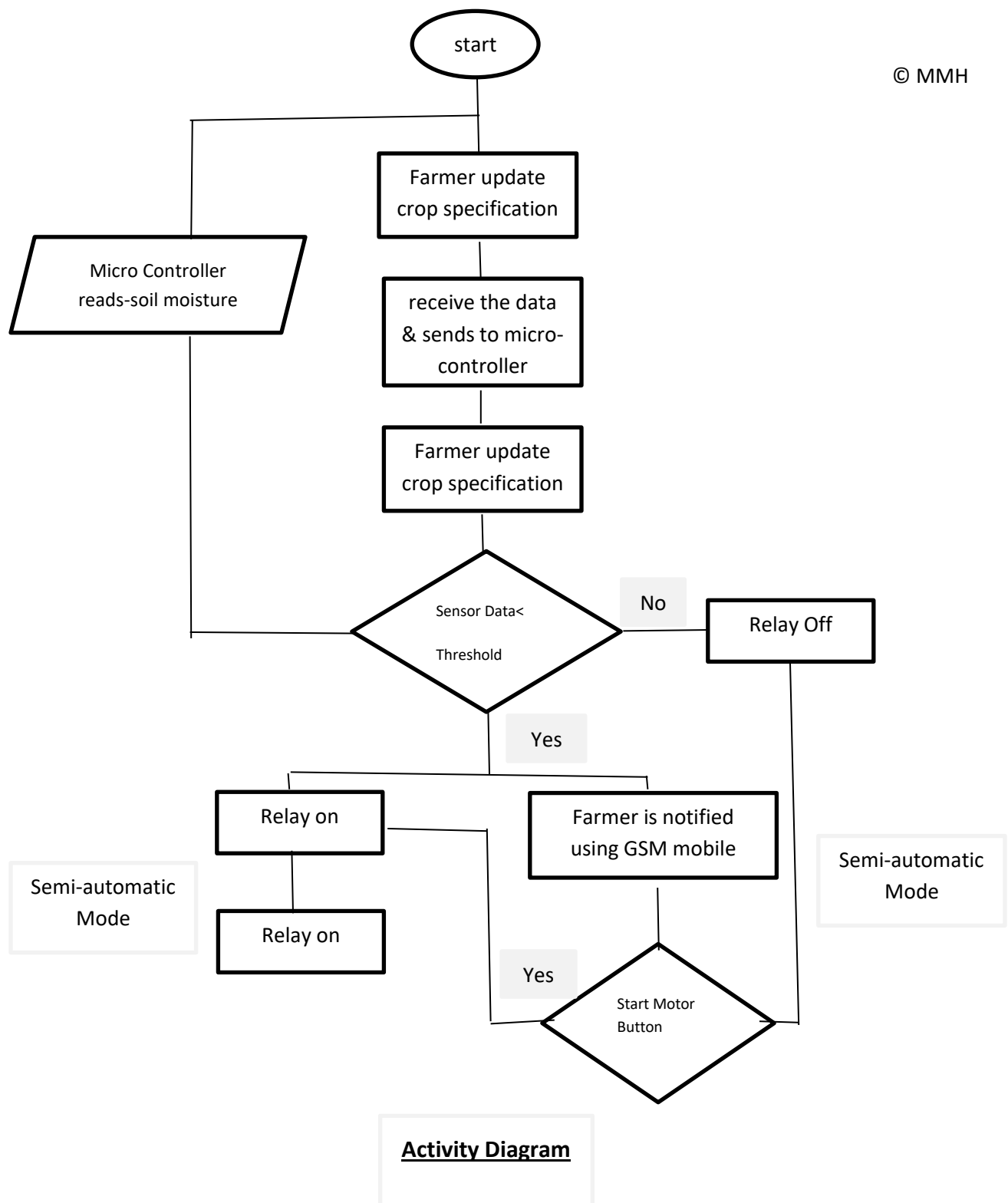
Diagrams:



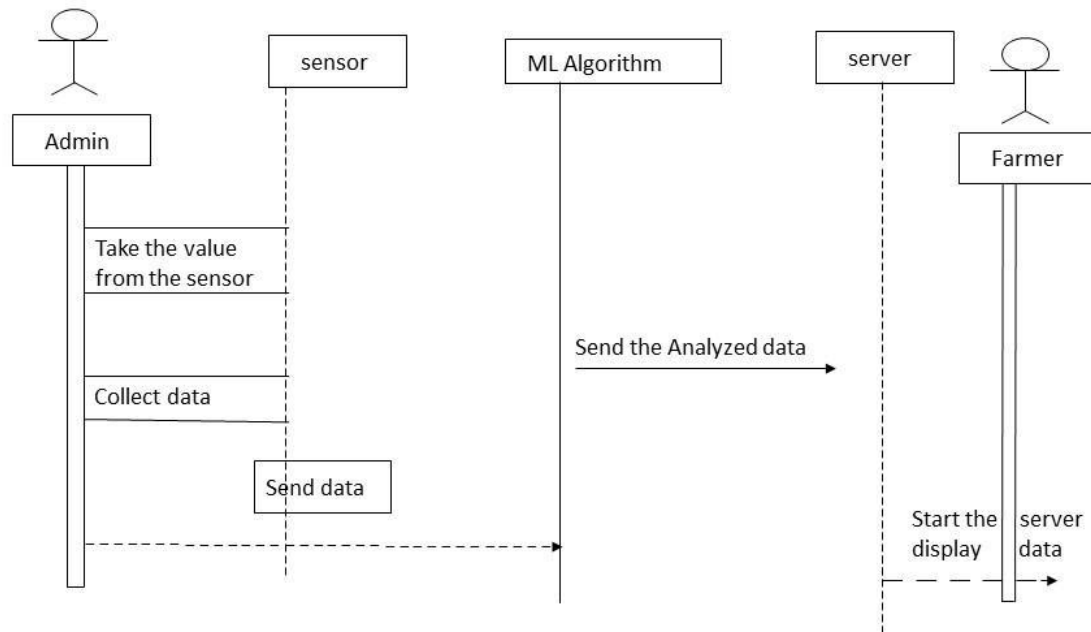
USE CASE

Class Diagram





Sequential Diagram



2.1 Project Role Identification and Responsibilities XP has seven different roles are customer:

XP has seven different roles are customer, programmer, tester, tracker, coach, consultant, and manager.

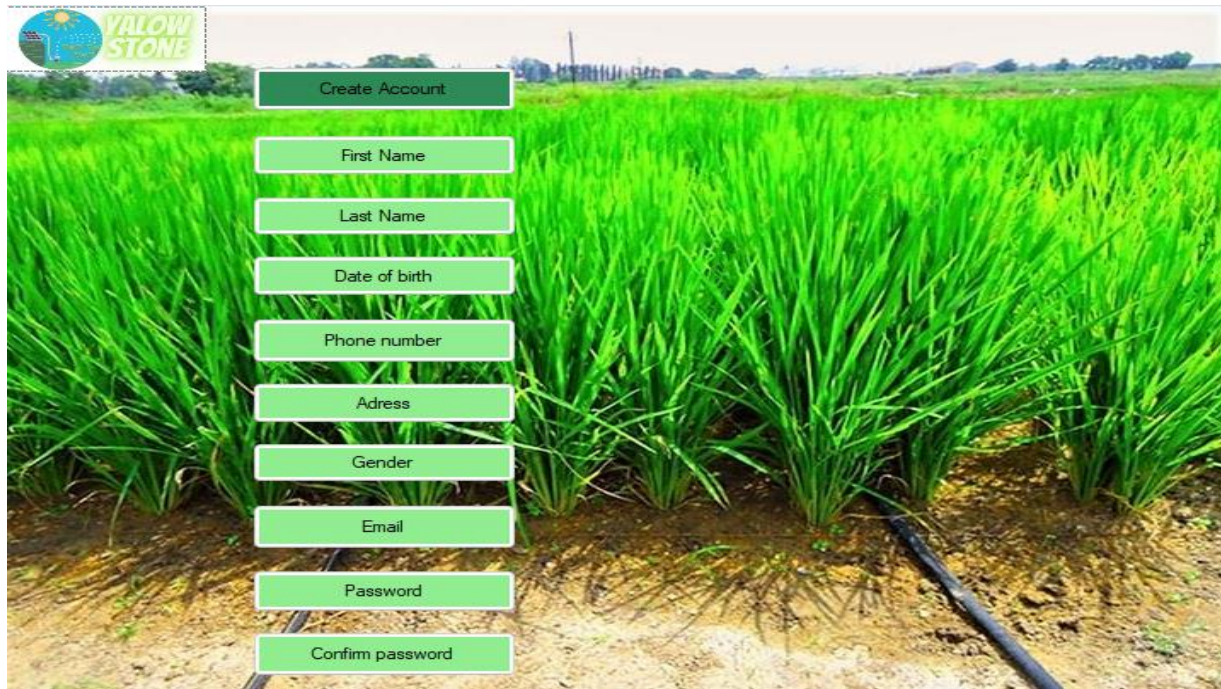
The customer writes the stories and functional tests and decides when each requirement is satisfied. The customer sets the implementation priority for the requirements

Programmer keeps the program code as simple and definite as possible

Tester helps the customer write functional tests, also run functional tests regularly, broadcast test results and maintain testing tools. Tracker gives feedback in XP. He traces the estimates made by the team effort estimates) and gives feedback on how accurate they are to improve future estimations. He also traces the progress of each iteration and evaluates whether the goal is reachable within the given resource and time constraints or if any changes are needed in the process. Coach is the person responsible for the process.

A sound understanding of XP is important in this role enabling the coach to guide the other team members in following the process. Consultant is an external member possessing the specific technical knowledge needed. Manager makes the decisions.

UI/UX DESIGN:



VALOW STONE

Create Account

First Name

Last Name

Date of birth

Phone number

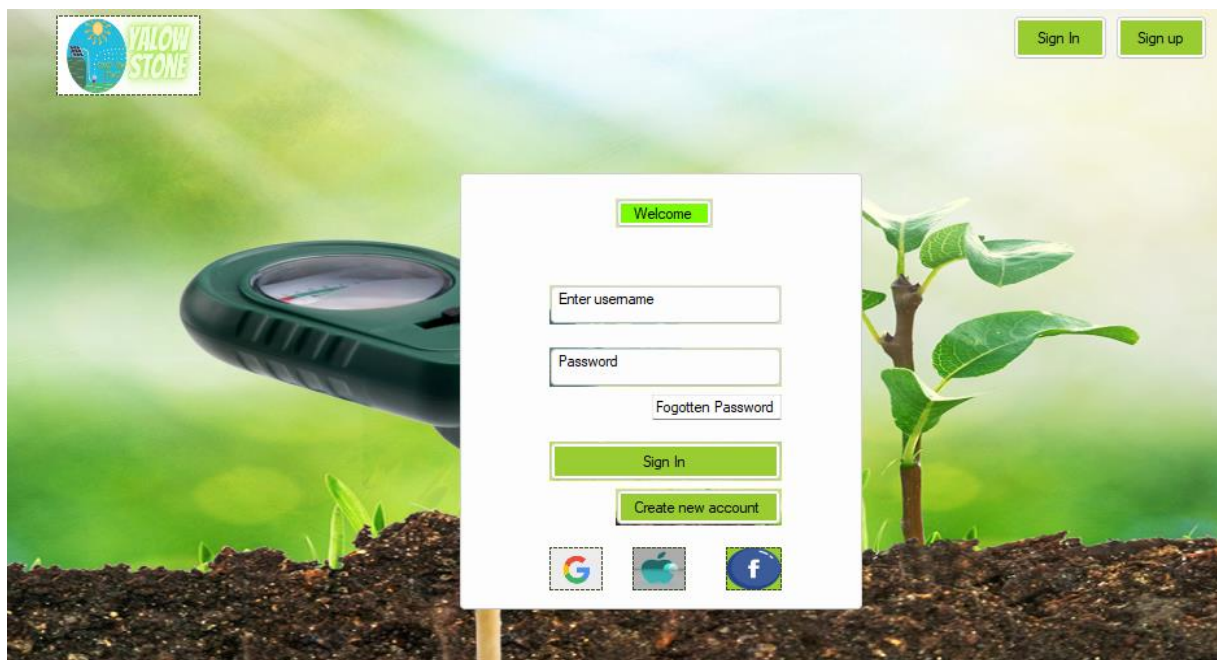
Adress

Gender

Email

Password

Confirm password



VALOW STONE

Sign In Sign up

Welcome

Enter username

Password

Fogotten Password

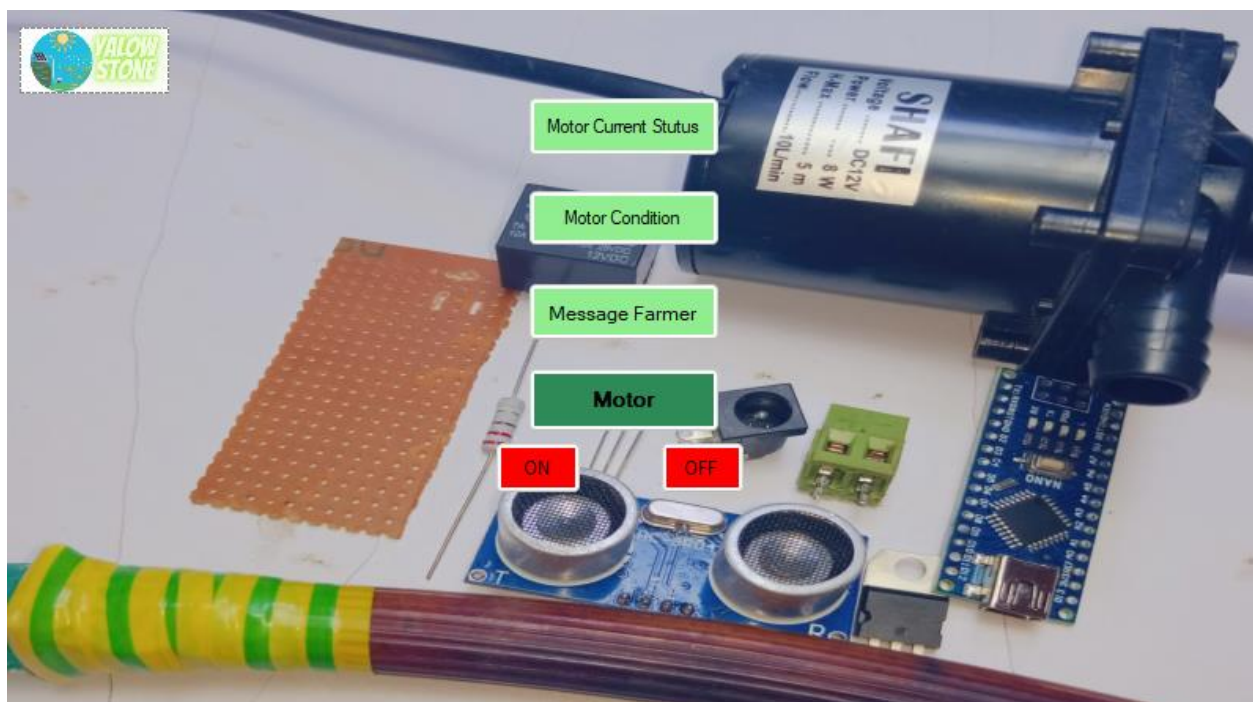
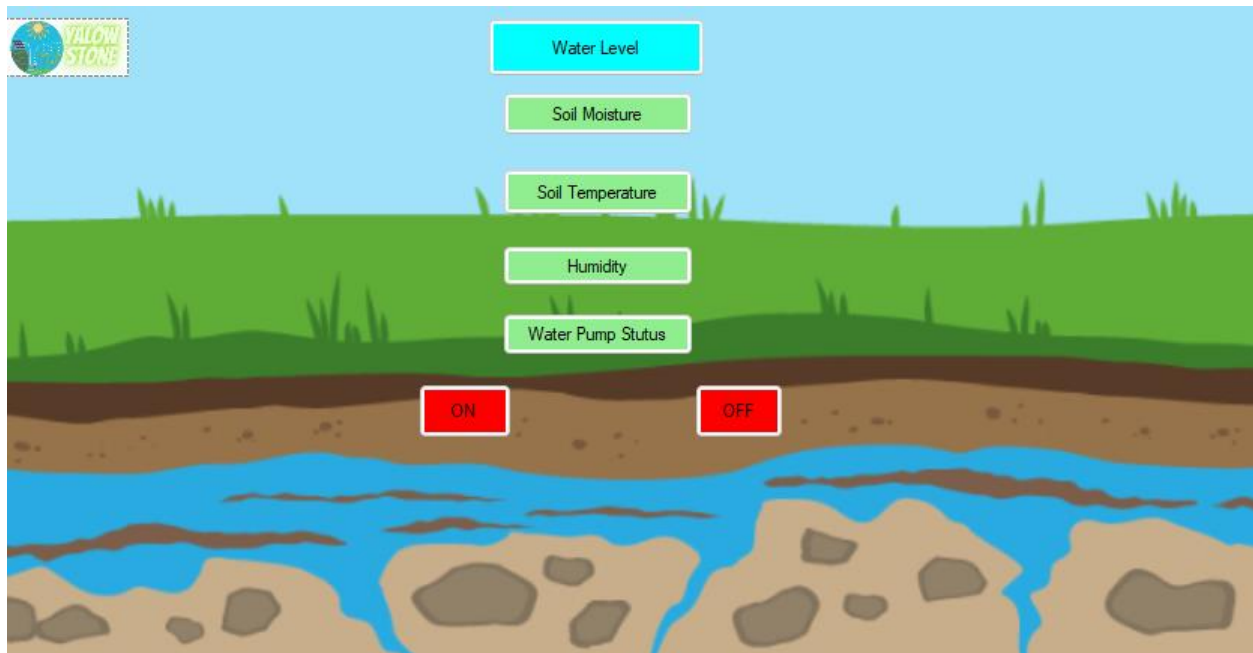
Sign In

Create new account

Google Apple Facebook











Test Cases

Project Name: Automatic Irrigation System.		Test Designed by: MD Borhan		
Test Case ID: FR_1		Test Designed date:02/09/22		
Test Priority (Low, Medium, High): High		Test Executed by: MD Borhan		
Module Name: Create Account		Test Execution date: 04/09/22		
Test Title: Create Account with valid username, password, and Information.				
Description: Test website create account page.				
Precondition (If any): User must have valid Information.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Enter First name, last name, Address, phone number. 3. Enter mail, password. 4. Confirm Password. 5. Click submit.	08\12\2022	User should Create Account successfully.	Create Account.	Pass
Post Condition: User is store in database and successfully create account.				

Project Name: Automatic Irrigation System.		Test Designed by: MD Borhan		
Test Case ID: FR_2		Test Designed date: 03/09/22		
Test Priority (Low, Medium, High): High		Test Executed by: MD Borhan		
Module Name: Login Session		Test Execution date: 06/09/22		
Test Title: verify login with valid username and password				
Description: Test website login page				
Precondition (If any): User must have valid username and password				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Enter username 3. Enter password 4. Click submit	08\12\2022	User should login into the application	As expected,	Pass
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Project Name: Automatic Irrigation System.		Test Designed by: APON Riaz		
Test Case ID: FR_3		Test Designed date:25/10/22		
Test Priority (Low, Medium, High): Medium		Test Executed by: Borhan		
Module Name: Farmer Information		Test Execution date: 02/1`/22		
Test Title: Store Farmer Information				
Description: This step store Farmer Information for next if needs.				
Precondition (If any): User can easily use all information.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Store farmer ID 2. Farmer Name 3. Contact Number 4. Farmer Address 5. Email Address	08\12\2022	User should access for this site so user can change or update information	As expected,	Pass

Post Condition: User can see Information of farmer if he wants, and farmer can change or update him /her contact.

Project Name: Automatic Irrigation System.		Test Designed by: MD Borhan		
Test Case ID: FR_4		Test Designed date: 02/11/22		
Test Priority (Low, Medium, High): Low		Test Executed by: Apon		
Module Name: Home		Test Execution date: 04/11/22		
Test Title: Home page store all information.				
Description: Test Accessibility information.				
Precondition (If any): User must login with username and password.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website 2. Enter username 3. Enter password 4. Click submit 5. Can move Every page and can Check everything.	Username: Password: 321	User should login into the application	As expected,	Pass
Post Condition: User is validated with database and successfully login to account. Can check everything and move every page.				

Project Name: Automatic Irrigation System.	Test Designed by: Apon Riaz
Test Case ID: FR_5	Test Designed date:25/11/22
Test Priority (Low, Medium, High): High	Test Executed by: Wasif
Module Name: Land Information	Test Execution date: 26/11/22
Test Title: Land Information	

Description: This step store land size or whole area.				
Precondition (If any): The location must be set in system				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)

1. Land Location 2. Land area		User should able to check when their need to know land location and area to set the project.	As expected,	Pass
Post Condition: User and project team or farmer can know the land location and area.				

Project Name: Automatic Irrigation System.		Test Designed by: Apon Riaz		
Test Case ID: FR_6		Test Designed date:26/11/22		
Test Priority (Low, Medium, High): Medium		Test Executed by: Lamia		
Module Name: Check Moisture		Test Execution date: 27/11/22		
Test Title: Check Moisture				
Description: Check That system can give moisture information so sensor can detect and done work				
Precondition (If any): Browser should relate to system & location must be on.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass /Fail)
1. Check water level 2. Check soil type		Farmer check or get the water level and soil type	As expected,	Pass
Post Condition: Check the water level of soil to see the water is need or not.				

Project Name: Automatic Irrigation System.	Test Designed by: Limia
Test Case ID: FR_7	Test Designed date: 10.12.2022
Test Priority (Low, Medium, High): Low	Test Executed by: Limia
Module Name: Water Level	Test Execution date: 10.12.2022
Test Title: Water level with soil moisture, temperature, humidity, water pump status.	
Description: Test website soil moisture, temperature, humidity, water pump status.	
Precondition (If any): User must log in with username and password.	

Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Check soil moisture 2. Check temperature 3. Check humidity 4. Check water pump status		User can check all the information of water level successfully.	As expected,	Pass
Post Condition: User can check all the information of water level.				

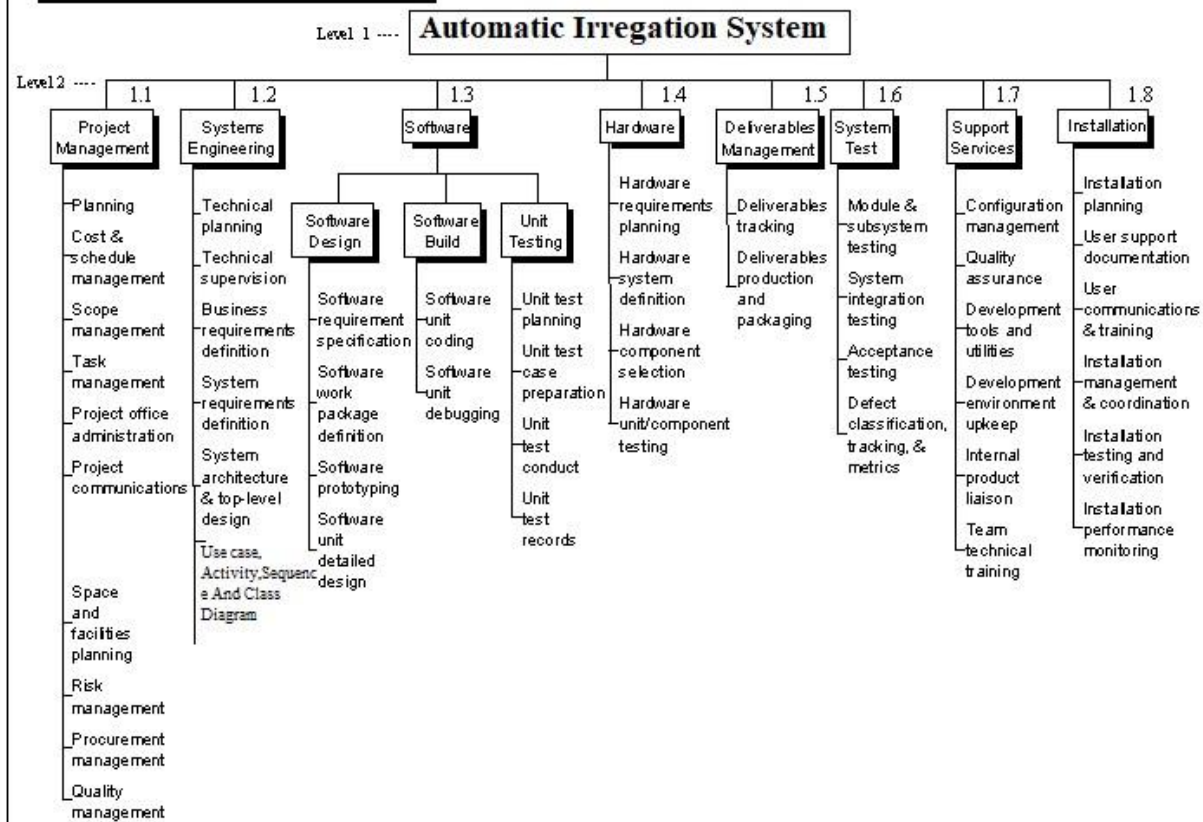
Project Name: Automatic Irrigation System.			Test Designed by: Limia	
Test Case ID: FR_8			Test Designed date: 10.12.2022	
Test Priority (Low, Medium, High): Low			Test Executed by: Limia	
Module Name: Motor On/Off			Test Execution date: 10.12.2022	
Test Title: User can control the motor manually.				
Description: Test website motor on and off function				
Precondition (If any): User must log in with username and password.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Turn On motor 2. Turn Off 3. Show status of motor	On Off	User should be able to turn ON/ Off the motor manually.	As expected,	Pass
Post Condition: Farmer can check the motor status and change the motor condition.				

Project Name: Automatic Irrigation System.		Test Designed by: Wasif Hassan Joarder	
Test Case ID: FR_9		Test Designed date:	
Test Priority (Low, Medium, High): Low		Test Executed by: Wasif Hassan	
Module Name: Support		Test Execution date:	

Test Title: Support with FAQ and Contact.				
Description: Test FAQ and Contact				
Precondition (If any): User must login with username and password.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
	08\12\2022			
1. FAQ for need any help. 2. Contact	Users write their problem.	The user gets a reply from the seller and solves their problem.	Reply mail from seller help center.	Pass
Post Condition: Users must write their problem in the mailbox and later they get a reply.				

Project Name: Automatic Irrigation System.			Test Designed by: Wasif Hassan Joarder	
Test Case ID: FR_10			Test Designed date:	
Test Priority (Low, Medium, High): Low			Test Executed by: Wasif Hassan	
Module Name: About			Test Execution date:	
Test Title: verify login with valid username and password				
Description: Test website About Page				
Precondition (If any): User can easily use all information				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
	02/12/2022			
1. Page Application 2. Description 3. Help 4. Terms & Condition		User can find all information about seller & get guideline about the system.	As expected,	Pass
Post Condition: Readers/visitors learn more about the system and find the system model, system all features.				

Work Breakdown Structure



CoCoMo

Effort = PM = Coefficient <Effort Factor>*(SLOC/1000) ^P

$$=2.4*(8000/1000) ^{1.05}$$

$$=21.303$$

Development time = DM = 2.50*(PM)^T

$$=2.50*(21.303) ^{0.38}$$

$$=7.993$$

Required number of people = ST = PM/DM

$$=2.665 \text{ person}$$

$$=3 \text{ person}$$

Build:

$$114,000 + 315,000$$

$$=429,0008.$$

ACTIVITY SCHEDULING

Person/weeks	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29-32
Rahim																													
Karim																													
Borkot																													

Activity Key:

A: Overall Design

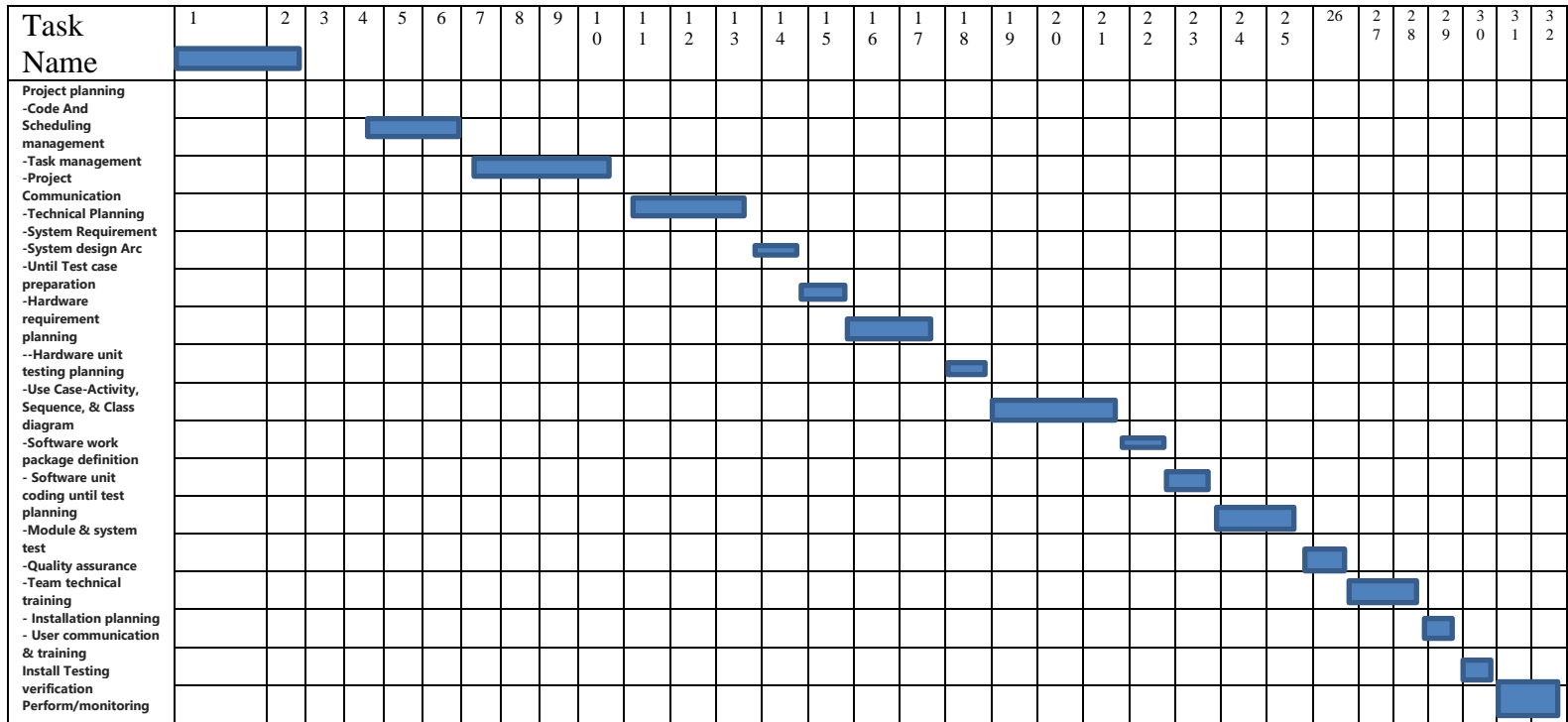
D: Code Module -2

B: Specify Module -1

E: Integration Testing.

C: Code Module -1

F: System Testing.



Task Number	Planned Effort		Actual Effort	
1		7	8	
2		14	12	
3		8	11	
4		7	9	
5	BCWP = 118	7.5	8	ACWP=124
6		11	12	
7		8	7	
8		14	14	
9		12	13	
10		10	11	
11		9.5	9	
12		10	10.5	
13		11		
14		4		
15		9		

Total Task = 38 Effort Estimated = 468 - Person Day

$$BCWS = 7+14+8+7+7.5+11+8+14+12+10+9.5+10+11+4+9$$

$$= 142$$

$$BCWP = 142 - 24$$

$$= 118$$

$$SPI = BCWP / BCWS$$

$$=118/142$$

$$=0.83$$

$$SV= BCWP-BCWS$$

$$=118-142$$

$$=-24 \text{ Person Per Day}$$

$$CPI=BCWP/ACWP$$

$$=118/124$$

$$= 0.95$$

$$CV = BCWP-ACWP$$

$$=-6 \text{ Person Per Day}$$

$$\% \text{Schedule for completion} = BCWS/BAC$$

$$= 142/468$$

$$=0.3030$$

[%Of work schedule to be done at this time]

$$\% \text{ Complete} = BCWP/BAC$$

$$= 118/468$$

$$= 0.242$$

[% of work completed at this time]

9. RISK ANALYSIS

Risks	Category	Probability	Impact	RMMM
Poor water Infrastructure scarcity	De	10%	3	
Size estimate may be low	PS	30%	3	
Larger number of users than planned	PS	60%	2	
End users resist system	BU	40%	3	
Customer will change requirements	PS	70%	2	
Technology will not meet expectations	TE	60%	2	
Staff turnover will be high	ST	40%	3	
Lack of training on tools	DE	80%	3	
Staff turnover will be high	ST	40%	3	
User complexity between Automatically & manual	PR	50%	2	
Less demand of product	BU	20%	3	
System clash				
Environment Disaster	DE	60%	2	