Institute of Information Technology
University of Dhaka

Topic: Assignment on Function Point Analysis

Software Metrics (SE-611)

Submitted by:

Muktadul Islam - 1215

Rufidatul Radium - 1226

Ahnaf Mubashshir - 1232

Submitted to:

Dr. Emon Kumar Dey
Associate Professor
Institute of Information Technology
University of Dhaka

Submission Date: 30/10/2023

Table of Contents

E-Kunjo

E-Kunjo is an agriculture-based website that thrives to provide its users with a one stop solution. It will help encourage urban farming by providing important services. As one of the first of its kind, working on it presented unique challenges to us. This document is a sample demand list of what the stakeholders want for the website.

Usage Scenario

The system will help the users to make their farming experience better. The gardeners can easily find their required information and the service providers can answer the questions asked by users.
It can help in day-to-day related activities, which would otherwise require going through a long and hefty manual procedure. The system consists of six major parts, which are:

1. Account management
2. Showing nearby nursery location
3. Providing plant information
4. Calculating optimized fertilizer amount
5. Live question answering system
6. Plant disease detection

Account Management:

There are three types of users in our system- Admin, gardeners and Service Providers. Only users who want to join the live chat service will need to create an account. The users will be detected as an Service Provider only if their phone number is pre recorded as an Service Provider in the database.
There will be 4 submodules for account management-

1. Sign Up
2. Sign In
3. Password Recovery
4. Adding Service Providers

Sign Up:

The Service Providers and gardeners need to register in the app to go through a live chat system. For that, users will need to provide their name and phone number. Then they will have to provide the OTP sent to the provided number to verify. Finally, they will need to set a password. The admin will upload the contact number of the Service Providers beforehand. The system will verify the phone number and

determine if the newly registered user is a Service Provider, and then load the interface according to the user type.

Sign In:

All users will need to provide the phone number and password to login.

Password Recovery:

If users forget the password, they can recover it with their phone number. They will provide the registered phone number and the system will generate OTP for him. Then the system will verify the entered OTP and give them the option to set a password. After setting a password, the user will be able to login.

Adding Service Providers:

After completing the recruitment process offline, an admin will add the phone number of the newly recruited Service Provider. After adding the phone number, a confirmation sms will be sent to the phone number of the corresponding Service Provider. The admin will be able to see the list of Service Providers anytime.

Show Nearby Nursery Location:

If the users want to see the location of nurseries, they need to grant permission to access the location through the app. Users can check nearest nurseries based on their current location.

Providing plant information:

The website will have a searching section. In this section, a user will be able to look up information about plants. The website will then show requested information which includes:

Calculating optimized fertilizer amount:

This section will ask a user for information about the user's desired plant and the amount of total plants. Then it will calculate the optimal amount of fertilizer needed and show the result.

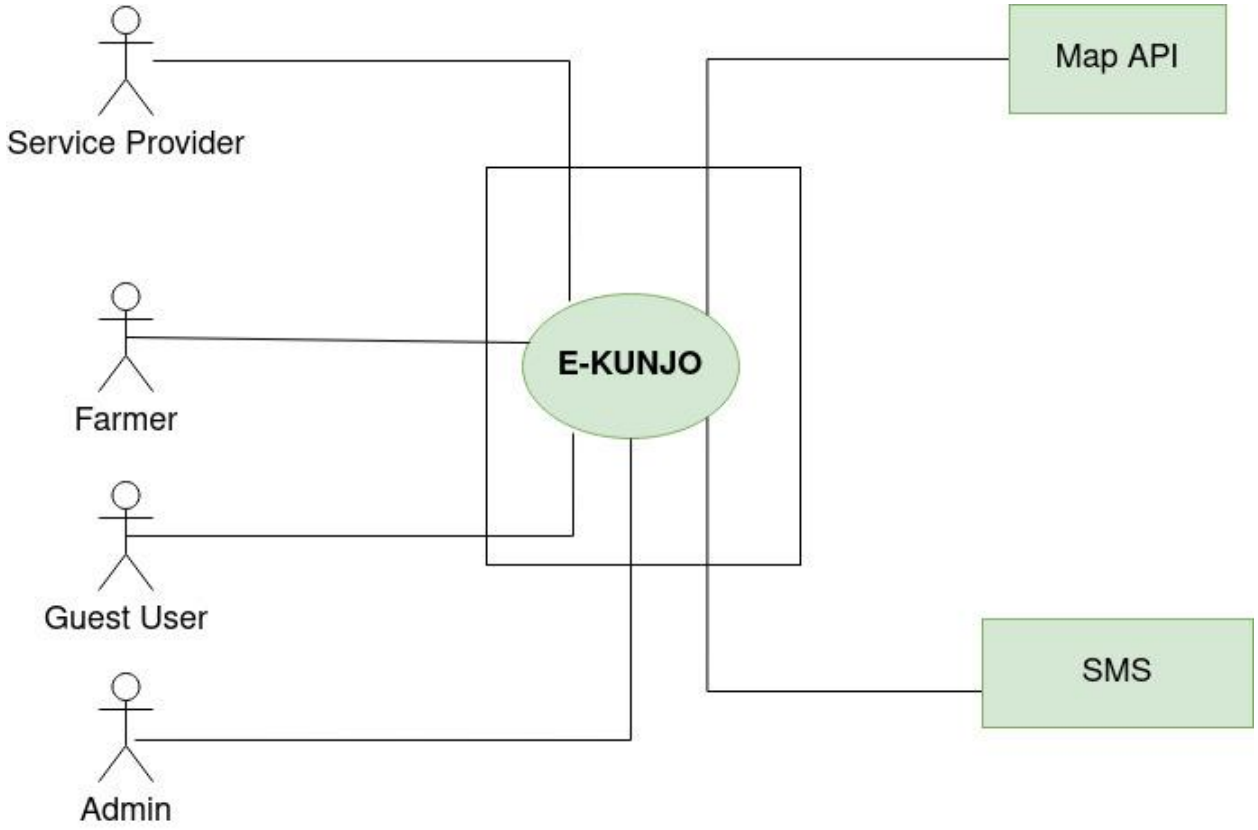
2.2.5 Live Chat System:

To use this feature , users must log in. Every authorized user can ask their problems. Then the question will be broadcast to available service providers and one of them will take the task and solve the problem.

2.2.6 Plant disease detection:

A user will need to upload the image of a diseased plant. The website will predict a probable disease based on a pre-trained deep learning model.

Use Case Diagram



Function points

A function point calculates software size with the help of logical design and performance of functions as per user requirements. Function points are a unit of measure used to define the value that the end user derives, or the functional business requirements the software is designed to accomplish.

Function Point (FP) is an element of software development that helps to approximate the cost of development early in the process.

Function Point Calculation

Step 1:

Measure size in terms of the amount of functionality in a system. Function points are computed by first calculating an unadjusted function point count (UFC). Counts are made for the following categories

- **External inputs** – those items provided by the user that describe distinct application-oriented data (such as file names and menu selections)
- **External outputs** – those items provided to the user that generate distinct application-oriented data (such as reports and messages, rather than the individual components of these)
- **External inquiries** – interactive inputs requiring a response
- **External files** – machine-readable interfaces to other systems
- **Internal files** – logical master files in the system

To calculate the function points for this project, we can use the function point analysis method, which involves assessing the complexity and quantity of various functions provided by the system. we'll break down the components into the different function point types and assign complexity levels for each:

1. Account Management:

1.1 External Inputs (EI):

1.1.1 Sign Up (Input of user details) - Low complexity

1.1.2 Sign In (Input of user credentials) - Low complexity

1.1.3 Password Recovery (Input of phone number and OTP) - Average complexity

1.1.4 Sign Out(Input of Removing user credential) - Low complexity

1.1.5 Adding Service Providers (Input of phone number) - Average complexity

2. Show Nearby Nursery Location:

2.1 External Inputs (EI):

2.1.1 Grant location permission - Low complexity

2.2 External Outputs (EO):

2.2.2 Display nursery location - High complexity

3. Providing Plant Information:

3.1 External Inquiries (EQ):
    3.1.1 Search for plant information - Low complexity
3.2 External Outputs (EO):
    3.2.1 Display plant information - Low complexity

4. Calculating Optimized Fertilizer Amount:

4.1 External Inputs (EI):
    4.1.1 Input plant and quantity - Low complexity
4.2 External Outputs (EO):
    4.2.1 Display calculated fertilizer amount - Average complexity

5. Live Chat System:

5.1 External Inputs (EI):
    5.1.1 User questions and inputs - Average complexity
5.2 External Outputs (EO):
    5.2.1 Service provider responses - High complexity

6. Plant Disease Detection:

6.1 Internal Files (ILF):
    6.1.1 Upload image - Low complexity
6.2 External Outputs (EO):
    6.2.1 Predicted disease information - High complexity

From the function point calculation, we can make the following table:

Parameter	**Category**	**Complexity**
Sign Up	External Input	Low Complexity
Sign In	External Input	Low Complexity
Password Recovery	External Input	Average Complexity

Sign Out	External Input	Low Complexity
Adding Service Providers	External Input	Average Complexity
Grant location permission	External Input	Low Complexity
Display nursery location	External Output	High Complexity
Search for plant information	External Inquiry	Low Complexity
Display plant information	External Output	Low Complexity
Input plant and quantity	External Input	Low Complexity
Display calculated fertilizer amount	External Output	Average Complexity
User questions and inputs	External Input	Average Complexity
Service provider responses	External Output	High Complexity
Upload image	Internal File	Low Complexity
Predicted disease information	External Output	High Complexity

Step 2:

Assigning weights: We can assign complexity values to each parameter based on the Function Point Analysis complexity rating system. In this system, functions are typically rated as low, average, or high complexity. We can assign values as follows:

Low Complexity: 3
Average Complexity: 4
High Complexity: 6

Parameter	Category	Complexity	Assigned Values
Sign Up	External Input	Low Complexity	3
Sign In	External Input	Low Complexity	3
Password Recovery	External Input	Average Complexity	4
Sign Out	External Input	Low Complexity	3
Adding Service Providers	External Input	Average Complexity	4
Grant location permission	External Input	Low Complexity	3
Display nursery location	External Output	High Complexity	6
Search for plant information	External Inquiry	Low Complexity	3
Display plant information	External Output	Low Complexity	3
Input plant and quantity	External Input	Low Complexity	3
Display calculated fertilizer amount	External Output	Average Complexity	4
User questions and inputs	External Input	Average Complexity	4
Service provider responses	External Output	High Complexity	6
Upload image	Internal File	Low Complexity	3
Predicted disease information	External Output	High Complexity	6

Step 3:

Calculate the total UFP (Unadjusted Function Points):

In the third step, we sum the products obtained in Step 2, which result from multiplying the counts of each category by their respective complexity weight factors. This summation yields the Unadjusted Function Points (UFP), a fundamental metric that represents the raw, unaltered functional size of the software system.

Functionality	Category	Low Complexity	Average Complexity	High Complexity
Account Management	External Inputs (EI)	3	2	0
Show Nearby Nursery Location	External Inputs (EI)	1	0	0
	External Outputs (EO)	0	0	1
Providing Plant Information	External Outputs (EO)	1	0	0
	External Inquiries (EQ)	1	0	0
	External Files (EIF)	0	0	0
	Internal Files (ILF)	0	0	0
Calculating Optimized Fertilizer Amount	External Inputs (EI)	1	0	0
	External Outputs (EO)	0	1	0
Live Chat System	External Inputs (EI)	0	1	0
	External Outputs (EO)	0	0	1
Plant Disease Detection	External Outputs (EO)	0	0	1
	Internal Files (ILF)	1	0	0

UFP = Sum of Complexity Values for all Parameters

UFP = 8 * 3 + 4 * 4 + 3 * 6 = 58
So, the total Unadjusted Function Points (UFP) for our project is 58. This represents the size and complexity of the software system based on the provided parameters and their assigned complexity values.

Step 4:

Calculate the total TCF (Technical Complexity Factor) by giving a value between 0 and 5 according to the importance of the following points

In this step, the Total Technical Complexity Factor (TCF) is established by assessing the significance of various aspects related to the software project. Assign a value between 0 and 5 to each point, based on their importance.

Total Technical Complexity Factor (TCF) Components:

The Total Technical Complexity Factor (TCF) is determined by assessing various factors relevant to the technical complexity of a software project. Here are the components that contribute to the TCF:

Data Communication: The complexity of data exchange and communication between system components.

Distributed Functions: The intricacy associated with functions distributed across different parts of the system.

Performance: Factors related to the system's speed and efficiency in processing tasks.

Heavily Utilized Hardware: The level of reliance on hardware resources in the system.

High Transaction Rates: The volume and complexity of transactions the system must handle.

Online Data Entry: Complexity in entering data into the system through online interfaces.

Online Updating: The complexity of real-time data updates and changes.

End-user Efficiency: The extent to which the system optimizes user efficiency and usability.

Complex Computations: The complexity of computational tasks and algorithms within the system.

Reusability: The potential for reusing system components and code.

Ease of Installation: Factors affecting the ease of system installation and setup.

Ease of Operation: Considerations related to user-friendliness and ease of system operation.

Portability: The adaptability and ease of moving the system to different environments.

Maintainability/Facility Change: Aspects related to system maintenance and adaptability to future changes.

Each component or subfactor is rated from 0 to 5, where 0 means the subfactor is irrelevant, 3 means it is average, and 5 means it is essential to the system being built.

Factor	Point
Data Communication	5
Distributed Functions	3
Performance	4
Heavily Utilized Hardware	0
High Transaction Rates	0
Online Data Entry	4
Online Updating	4
End-user Efficiency	4
Complex Computations	4
Reusability	5
Ease of Installation	5
Ease of Operation	4
Portability	3
Maintainability/Facility Change	4

Step 5:

Sum the resulting numbers to obtain DI (degree of influence)

The Degree of Influence (DI) is calculated by summing the values assigned to the various factors in the TCF (Total Technical Complexity Factor).

Now, we can calculate the DI by summing these values:

DI= 5 + 3 + 4 + 0 + 0 + 4 + 4 + 4 + 4 + 5 + 5 + 4 + 3 + 4 = 49

Step 6:

TCF (Technical Complexity Factor) by given by the formula

TCF = 0.65+0.01*DI = .65 +0.01* 49 = 1.14

Step 7:

Function Points are given by the formula

FP=UFP*TCF = 58*1.14 = 66.12

Conclusion

In this report, we have conducted a Function Point Analysis to estimate the size and complexity of the software project based on a detailed project description and usage scenarios. The project, which encompasses various modules related to account management, verification, attendance tracking, data storage, proxy detection, and report generation, has been analyzed using the Function Point Analysis method.