American International University-Bangladesh (AIUB)

Department of Computer Science

Faculty of Science &Technology (FST)

Spring 21 22

< Land Renting System >

Software Requirement Engineering

Sec: **B**

Project submitted

By

Istiak kabir	*(18-37924-2)*
Md. Safaet	*(18-38704-3)*
Abul Kashem Nibir	*(18-38660-3)*
Shethil Ahammed	*(19-39698-1)*

Table of Contents

List of Figures

List of Tables

1. Problem Domain

1.1 Background to the Problem

Land Renting system in Bangladesh is not well-developed. It is beset with multiple defects and problems. It is corrupt, inefficient, and unreliable and inherently contains systematic weaknesses. Corruption has become a grave issue in this sector. A world bank survey reveals that most crimes and corruptions in Bangladesh take place in land-related services. So we want to provide a safe platform for everyone where they can easily Rent their land with proper safety. This software provide us every legal document related to land and valid id of customers .So there is no chance of corruption.

Now a day's people should rent or hire lands by checking all the legal documents and lands . So the land owner must be maintaining the land details and also the availability. So in this case maintenance have the details about the land, current state and amount what are the details we had. The land details contain land code, land name, land types. These details are maintained in database. In the hiring function we must have the details about the owner, quantity and also rent amount. The Sales Details contain Date, Customer name, land code, Quantity and renting amount. The Stock Details contain land id, opening stock, hiring stock, current stock, and sales. The details are maintaining the database and view the land between two dates.

1.2 Solution to the Problem

We are going to get a solution that is a software name "Land Renting System" to deal with this problem. This solution is appropriate to solve this problem because of following

1.Simplicity in controlling system

2. Providing every legal document related to land

3.Security and quality management

4.Reducing human errors

5.Attractive report generator

This solution is particularly appropriate to solve the problem because all kind of public demands has been focused in this solution.

Yes, the solution is feasible to the meet the business objective.

The intent and goal is to have all of our investment property well maintained with the belief that good management and maintenance procedures will enhance real estate value and promote good tenant relations. A good amount of charge ($ money) will be claimed from both customer and owner to lease a land. Leases are usually easier to obtain and have more flexible terms than loans for buying equipment.

Right now there is no land renting app or software in our country. But there are some similar land selling app or software are present in our country. So this area we want to solve this problem by making this land renting software.

2. Solution Description

2.1 System Features

Our system is used by a number of people. For each user, there is a distinct set of requirements.

- System Manager/Admin
- Client
- Land owner

Features for all users

1. Log in

2. Registration using nid card

Features for System Manager/Admin:

1. Will verify users’ profile .
2. Can block/approve Users.
3. Check all the documents
4. Make contract between client and land owner.
5. Confirm/Cancel agreement.
6. Take payment through Cash, Bkash , Card, Checks.

Features for System Client:

1.Client can Search for Land.

2. Client can see all the legal documents.

3. Can contact system Admin.

4. Client can book the land.

5. Client can make payment through our System.

Features for System Land Owner :

1. Land owner can give land details.

2. Land owner can submit land documents.

3. Can contact system Admin.

4.Can set a price.

5.Can take payment through our system.

2.1.1 Software Quality Attribute

- **Reliability:** Measure if the system is reliable enough to sustain in any condition. Should give consistently correct results. Software reliability is measured in terms of working of the project under different working environments and different conditions.
- **Maintainability**: Different versions of the software should be easy to maintain. For development it should be easy to add code to the existing system, and should be easy to upgrade for new features and modern technologies from time to time. Maintenance should be cost-effective and easy. The system is easy to maintain and correcting defects or making a change in the software.

- **Usability:** This can be measured in terms of ease of use. The application should be user-friendly. Should be easy to learn. Navigation should be simple. The system must be: Easy to use for input preparation, operation, and interpretation of the output. Provide consistent user interface standards or conventions with our other frequently used systems. Easy for new or infrequent users to learn to use the system.
- **Portability:** This can be measured in terms of Costing issues related to porting, technical issues related to porting, Behavioral issues related to porting.
- **Correctness:** The application should be correct in terms of its functionality, calculations used internally and the navigation should be correct. This means the application should adhere to functional requirements.
- **Efficiency:** Major system quality attribute. Measured in terms of time required to complete any task given to the system. If the system is not efficient then it cannot be used in real-time applications.
- **Integrity or Security:** Integrity comes with security. System integrity or security should be sufficient to prevent unauthorized access to system functions, preventing information loss, ensure that the software is protected from virus infection, and protecting the privacy of data entered the system.
- **Testability:** The system should be easy to test and find defects. If required it should be easy to divide into different modules for testing.
- **Flexibility:** Should be flexible enough to modify. Adaptable to other products with which it needs interaction. Should be easy to interface with other standard 3rd party components.
- **Reusability:** Software reuse is a good cost-efficient and time-saving development way. Different code libraries classes should be generic enough to use easily in different application modules. Dividing application into different modules so that modules can be reused across the application.

Applying above quality attributes standards we can determine whether the system meets the requirements of quality or not.

2.1.2 Security Attribute

- Each user has a unique id to distinguish him/her.
- Each client can edit/update/delete their account.
- Auto logout option.
- user can see all posts.
- Moderator, admin will have two step verifications for log in.
- Forget password option.
- We use SSL which is a digital certificate that encrypts information sent between a web server and web browser. It is one of the most effective ways to achieve data security on our website and to keep our participant data protected.
- We Use a Real-Time Bot Detection Technology to eliminate price scraping and other Online frauds.

- We use a web application security system for network level security. Waf protects websites from common attacks such as cross-site scripting (XSS), SQL injections and DDos attacks. Since system have a lot of inbound traffic, they need firewalls to protect themselves against malicious entry. The two highly effective firewalls for software are application gateways and proxy firewalls.
- We use object oriented php for more security.
- We have a system in place for purging participant data.
- We Insist on participants using strong passwords by pop up notification.

2.2 UML Diagrams

Use-Case Diagram:

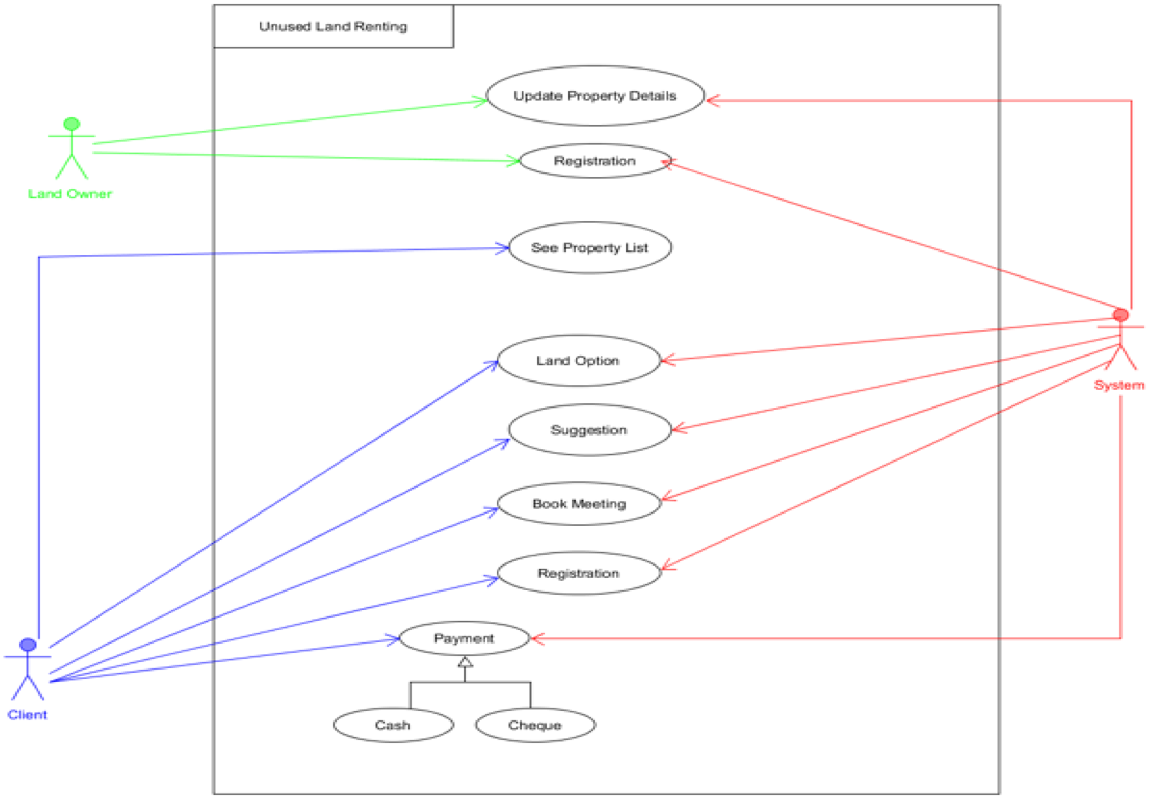


Fig 2.2.1: use case Diagram.

Activity Diagram:

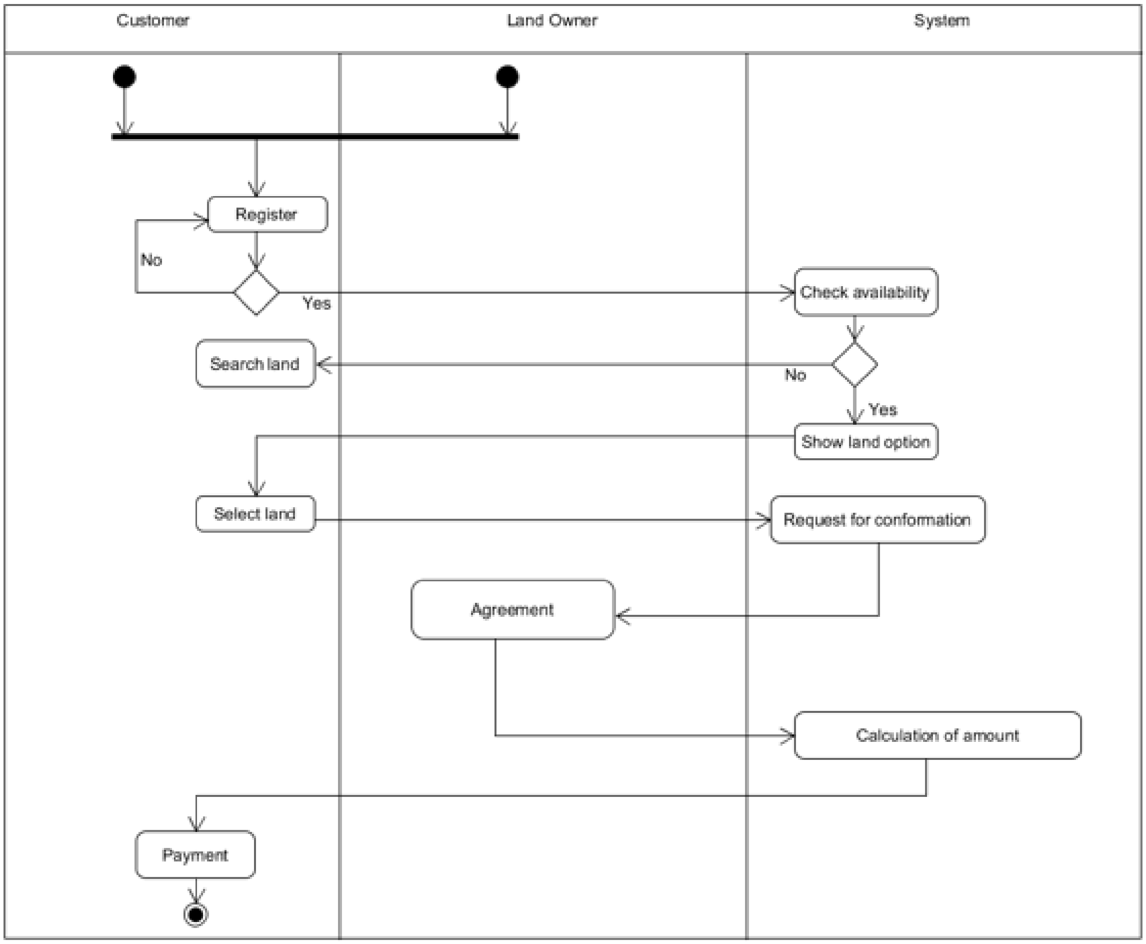


Fig 2.2.2: activity Diagram.

E-R diagram :

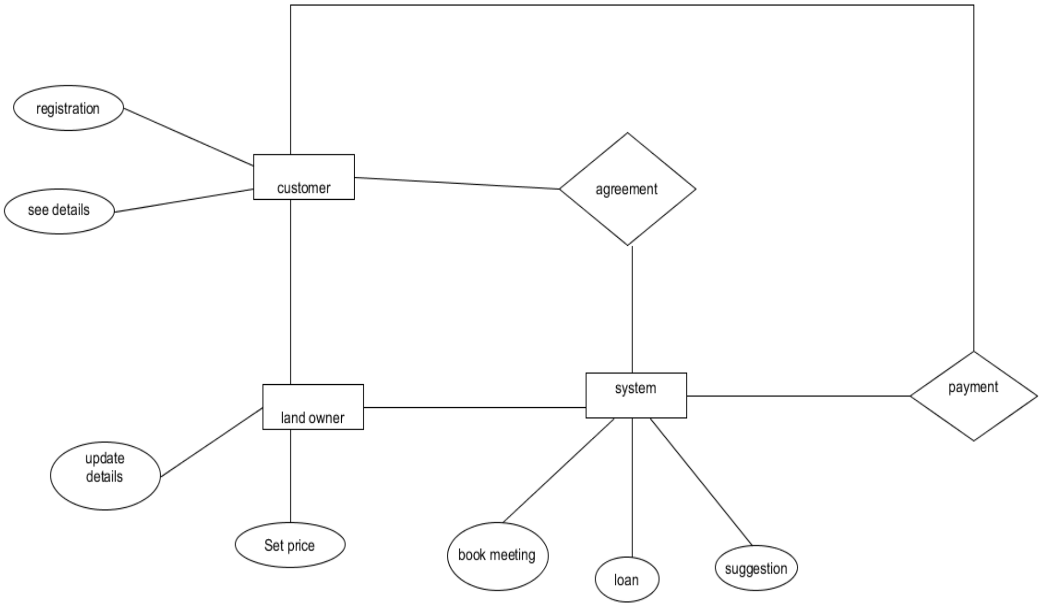
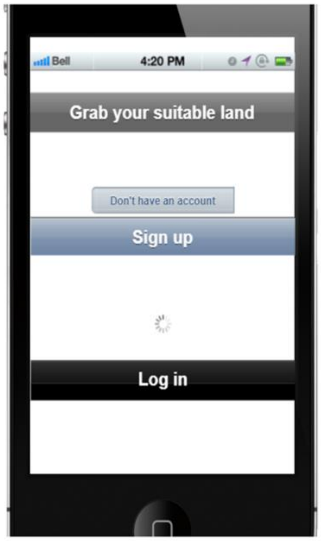
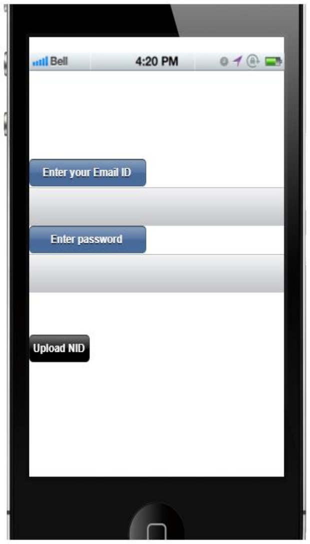
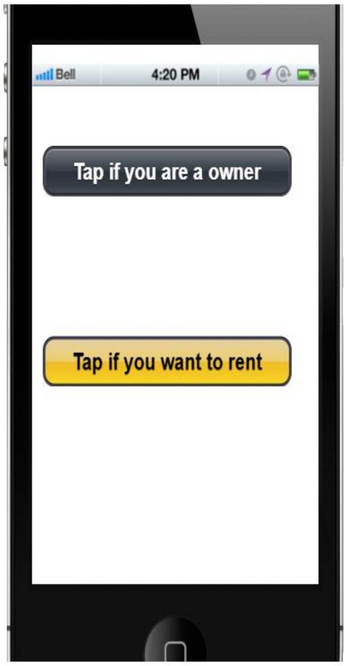


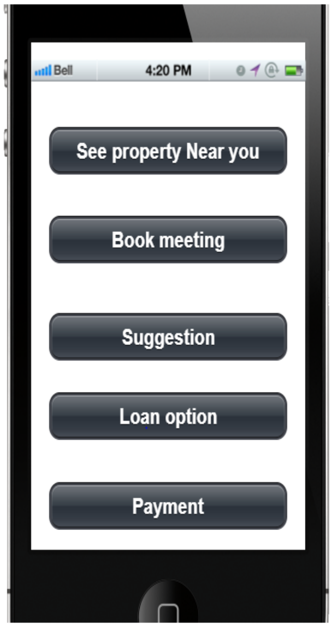
Fig 2.2.3: E-R Diagram.

2.3 UI interface:









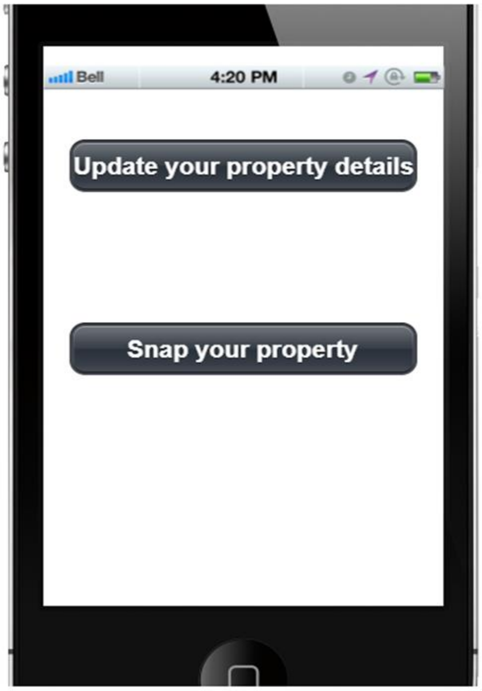


Fig 2.3.1: UI interface.

3. Social Impact:

The name of our project is land renting system. Basically, land renting system gives the proper detailed explanation about the unuse lands. This system makes a relationship between the owner and the customer. An owner or customer can see the details of the lands through the system. Then the customer can analyse which type of land he wants to take rent. Here owner and customer can easily access to the system and show the proper details about the lands which makes the project very interesting.Renting land in our country is very difficult and in this sector there many corruption. So we want to provide a safe platform for everyone where they can easily Rent their land with proper safety. This software provide us every legal document related to land and valid id of customers .So there is no chance of corruption and people will get benefit from this app.

4. Effort Estimation and Budgeting:

Effort = PM = Coefficient<Effort Factor> * (SLOC/1000) ^ P [100000 SLOC/1000 = 100k SLOC]

=2.4 * (100000/1000) ^1.05

=302.14

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

=2.50 * (302.14) ^ 0.38

=21.89

Required number of people = ST = PM/DM

=302.14/21.89

=11

4.1 Budgeting:

Development Cost:

Total working days = total days * DM

= 4 * 21.89

= 88 days

Working hour per day = 6

Total working hours = 88 * 6

= 528 hours

Total development cost = 528 * 600

= 316800

Maintenance Cost:

Monthly 12 Hours (6 months and per hour salary 1200)

Cost = 12 x 6 x 1200

= 86,400

Requirement Cost:

Days = 15 working hour= 7 salary = 750

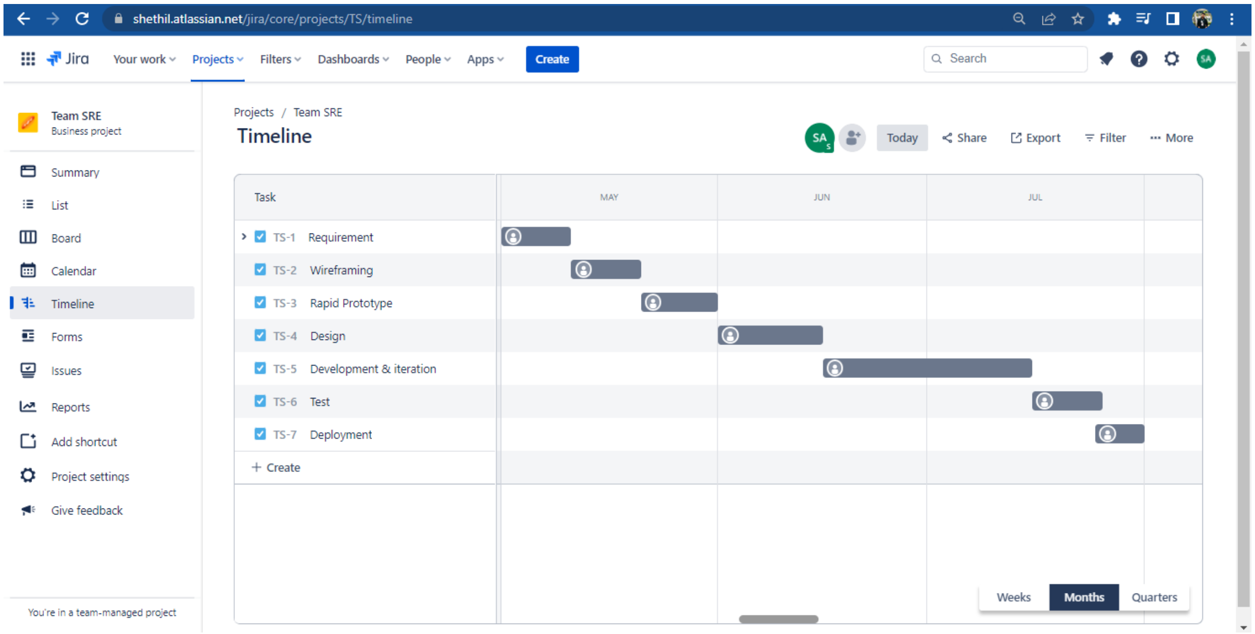
Total Cost = 7 x 15 x 750

= 78,750

Development Cost	316800
Requirement Cost	78,750
Travel Cost	8,000
Maintenance Cost	86,400
Training	32,000
Equipment	1,00,000
Utilities	1,13,000
Profit (20%)	2,87,150
Total	14,93,700

Table 4.1 : Requirement Cost

5. Project Schedule



Task	
TS-1	Requirement
TS-2	Wireframing
TS-3	Rapid Prototype
TS-4	Design
TS-5	Development & iteration
TS-6	Test
TS-7	Deployment

Fig 5.1 : Gantt Chart.

Activity key:

1. Requirement
    - 1.1 Create Requirement Defination
    - 1.2 Research Requirement
    - 1.3 Exit Review of Requirement Defination
2. Wireframing
    - 2.1 Define Features
    - 2.2 Design Software Wireframe
    - 2.3 Define User Action
    - 2.4 Define App
    - 2.5 Exit Review of App wareframe
3. Rapid Protitype
    - 3.1 Throwaway/Rapid Prototyping
    - 3.2 Incremental Prototyping
    - 3.3 Evolutionary Prototyping
    - 3.4 Extreme Prototyping
4. Design
    - 4.1 Architectural Design
    - 4.2 High-level Design
    - 4.3 Detailed Design

5. Development & Iteration
    - 5.1 Analysis
    - 5.2 Design
    - 5.3 Coading
6. Test
    - 6.1 Requirement Analysis.
    - 6.2 Test Planning.
    - 6.3 Test Case Designing and Development.
    - 6.4 Test Environment Setup.
    - 6.5 Test Execution.
    - 6.6 Test Closure.
7. Deployment
    - 7.1 Product Release
    - 7.2 Launch Date
    - 7.3 Monitoring Release
    - 7.4 Complete Delaration of Release

Task	Duration (days)
1. Requirement	**10**
2. Wireframing	**10**
3. Rapid Protitype	**10**
4. Design	**15**
5. Development & Iteration	**30**
6. Test	**10**
7. Deployment	**05**
Total	**90**

Table 5.1 : Total duration time

6. Development Plan

For develop the software our time range is 21th July to 31 August, almost 40 days. The Software Development Plan describes a developer's plans for conducting a software development effort. The SDP provides the acquirer insight and a tool for monitoring the processes to be followed for software development. It also details methods to be used and the approach to be followed for each activity, organization, and resource. The purpose of the development plan is to communicate to team members and stakeholders the approach to be taken when developing software on a program and how a Program Manager will utilize direct resources. The goal of development plan is to provide a framework where team members on a software project have a clear understanding of the software we are developing and the problems we are trying to solve.

The development plan should be developed in the contractor's preferred format and should document all processes applicable to the system to be acquired, at a level of detail sufficient to allow the use of the development plan as the full guidance for the developers. It should reference specific standards, methods, tools, actions, reuse strategy, and responsibility associated with the development and qualification of all requirements, including safety and security.

Software Development Plan Steps:

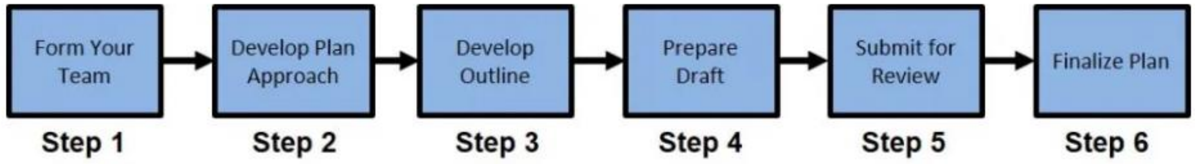


Fig 6.1: steps of development plan

There are six steps associated with the development of an SDP [1]. These six steps are:

Step 1: Form Team: Gather the subject matter experts you will need to develop the SDP.

Step 2: Develop Plan Approach: Figure out how you are going to write the plan and who is responsible for each section.

Step 3: Develop Outline: Start with an outline or a template. This step needs to ensure all required information is addressed in the SDP.

Step 4: Prepare Draft: Write the first draft of the SDP.

Step 5: Submit for Review: Submit the first draft to all team members that will utilize the plan for their feedback.

Step 6: Finalize SDP: Submit the final SDP to the team

First of all we will make our team. We thing for development this we need 6 members in our team. Next we will figure out how we are going to write the plan and who is responsible for each section. Next we will Start with an outline or a template. This will ensure all required information. Then we will write the first draft of the SDP & Submit it to all team members that will utilize the plan for their feedback. Finally submit the final SDP to the team. After that the team are ready to development the software. Hopefully this will complete with in 40 days. After finishing the development we will move to our next task which is Testing. For testing we need two members, they will get 10 days for testing. If they found any problem then it will move to again development section and fix the problem. After development & testing we are ready to deployment.

7. Change Management Plan

7.1 Purpose And Scope :

The Change Management Plan documents and tacks the necessary information required to effectively manage project change from project inception to delivery.

The Change Management Plan is created during the Planning Phase of the project. Its intended audience is the project manager, project team, project sponsor and any senior leaders whose support is needed to carry out the plan.

7.2 Roles And Responsibilities :

The following are the roles and responsibilities for all change management efforts related to the project.

Project sponsor

- Approve all changes to budget/funding allocations.
- Approve all changes to schedule baseline.
- Approve any changes in project scope.
- Chair the CCB.

Project manager

- Receive and log all change requests from project stakeholders.
- Conduct preliminary risk, cost, schedule, scope analysis of change prior to CCB.
- Seek clarification from change requestors on any open issues or concerns.
- Ensure all documentation revisions/edits are completed as necessary for all approved changes.
- Facilitate the CCB meetings.

Project team/stakeholders

- Submit all change requests on change request forms.
- Provide all applicable information and detail on change request forms.
- Be prepared to address questions regarding any submitted change requests.
- Provide feedback as necessary on impact of proposed changes.

7.3 Change Request Status

CHANGE REQUEST FORM

Change Description		
Project Name:	**Change Name:**	**Number:**
Requested By:	**Contact:**	**Date:**
Description of Change:		
Reason for Change:		
Priority [Circle One]: 1. High 2. Medium 3. Low		
Impact on Deliverables:		
Impact of Not Responding to Change (and Reason Why):		
Date Needed:	**Approval of Request:**	**Date:**

Table 7.3.1 : change Request status

Change Impact
Tasks/Scope Affected:
Cost Evaluation:
Risk Evaluation:
Quality Evaluation:
Additional Resources:
Duration:
Additional Effort:
Impact on Deadline:
Alternative and Recommendations:
Comments:

Table 7.3.2 : change impact evaluation

7.4 Evaluate Change Request :

Change requests are evaluated using the following priority criteria:

Priority	Description
High	
Medium	
Low	

Table 7.4.1: change request priority

Change Control Board Decision

Decision	Approved	Approved with Conditions	Rejected	More Info
Decision Date				
Decision Explanation				
Conditions				
Approval Signature		**Date Signed**		

Table 7.4.2: CCB Decision

7.5 Change Control Status

Status	Description
Open	Entered/Open but not yet approved or assigned
Work in Progress	CR approved, assigned, and work is progressing
In Review	CR work is completed and in final review prior to testing
Testing	CR work has been reviewed and is being tested
Closed	CR work is complete, has passed all tests, and updates have been released.

Table 7.5.1: change control status

8. Marketing Plan

8.1 Define goals

We have more than one main goal for our online marketing: -

- We want to draw the attention of different generation of citizens.
- Want to influence property owners to try.

8.2 Target audience

Our application is not for everyone. It is related to project buildup and team work. So, our target audiences are adult citizen who are financially independent or has property to trade with.
Here are some things which indicates our target audiences: -

- Age (25 or above)
- Gender (Both Male & Female)
- Education level (College or Undergraduate or above)
- Occupation (financially independent)
- General interests (buying, selling or renting property)

But there is no restriction for anyone to enter and use the application as it is an open app.

8.3 Marketing Tactics

For our system we have chosen two types of marketing tactics: -

- SMS/Email marketing.
- Social media marketing.

We have chosen these two marketing tactics because every potential user have both personal and organizational phone number / email address. So, if we use that we can reach to every employee and student and might get a good response. Besides nowadays everyone uses social media (Facebook, Tweeter, Instagram). So, it is also an effective way to reach the target audiences.

9. Cost and Profit Analysis

9.1 Total Cost

Project Team Costing:

1. Project Manager (1) (8 month) - 5, 20,000 tk.
2. Project Leader (1) (8 month) (per month 35000 taka) - 2,80,000 tk.
3. Programmer (6) (6 month) (per month 38,500 taka) - 13,86,000 tk.
4. Tester (4) (6 month) (per moth 40,000 taka)- 960000 taka

Profit Analysis:

By subscription: Let's set our two-year subscription rate at 800 taka. We expect our subscription to be purchased by at least 25000 people. They'll see our ad and come running. So, in two years, we'll have 25000 users. In the pandemic situation the importance of hospitals has increased and people need these facilities even more. Therefore, through the management system many people will be benefitted and people will get the services properly without facing any sufferings.

So, from that we can earn: Subscription
rate per month - 800 tk.Users - 25000
person
So total income- (800*25000) = 20000000 tk.Total =
31,46,000 tk.

Office Cost:

1) Office rent (1 year and 6 months) - 3, 00,000 tk.
2) Maintenance (1 year) - 1, 30,000 tk.Total
– 4,30,000 tk.

Marketing Cost:

1) Broad Casting (5 month) - 1, 00,000 tk.
2) Seminar (8 seminar) - 1, 90,000 tk.3)
3)PPC - 70,000 tk.
4) public relation - 3,60,000 tk.
5) Publishing and promoting content – 300000 tk.
6) SEO (per month) - 5,75,000 tk.
7) Social Media - 3,15,000
8) Advertising and branding - 90,000 tk

Total Marketing Cost**: 2000000** tk.Total
Costing - **55,76,000** taka.

9.2 Total Profit:

So, in 2 year our total earning will be 20000000.

Then, our profit will be - (20000000 tk. – 55,76,000 tk.) = 1,44,24,000 tk.
So, in 2 years we can make a huge profit from the system. And after 2 years we will reduce our subscription rate gradually. In this way we will gradually run the systemand earn the profit.

Reference:

[1] https://acqnotes.com/acqnote/careerfields/software-development-plan