American International University-Bangladesh (AIUB)

Department of Computer Science
Faculty of Science &Technology (FST)
Spring 2021

One-time renting app

Software Requirement Engineering

Sec: C

Project submitted

By
NAJMUL UDDIN   *(18-38293-2)*
MAHZABIN MOSTARY *(18-38606-2)*
DIP ROY   *(18-38628-2)*

PROJECT TITLE   ONE TIME RENTING APP

1. Introduction

The main goal of our project is that people buy or renting their urgent needed in a very short time.

2. Over of the Project

Our project is for renting. Sometimes we have to buy something that we use for just a single time. So our objective is to create software which will help us to lessen this kind of problem. The project is basically run by three actors. They are "Owner", "Customer", and "Admin". These three actors have their particular actions to perform. So here we want to bull up a communication between two people who want to rent that small tool and who want to give that tool to him. This is our only and main goal.

3. Justification And Development Plan

SDLC: SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace, and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

A typical Software Development Life Cycle consists of the following stages: -

- Planning and Requirement Analysis.
- Defining Requirements.
- Designing the Product Architecture.
- Building or Developing the Product.
- Testing the Product.
- Deployment in the Market and Maintenance.

Planning and Requirement Analysis: Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas.

Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

Designing the Product Architecture: Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an SRS (Software Requirement Specification) document which consists of all the product requirements to be designed and developed during the project life cycle.

Defining Requirements: SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification. This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product. A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third-party modules. The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

Building or Developing the Product: In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle.

Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed

Testing the Product: This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

Deployment in the Market and Maintenance: Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment. Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

SDLC provides a series of steps to be followed to develop a software product efficiently. SDLC framework includes the following steps:

Communication: This is the first step where the user initiates the request for a desired software product. He contacts the service provider and tries to negotiate the terms. He submits his request to the service providing organization in writing.

Requirement Gathering: This step onwards the software development team works to carry on the project. The team holds discussions with various stakeholders from problem domain and tries to bring out as much information as possible on their requirements. The requirements are contemplated and segregated into user requirements, system requirements and functional requirements.

Feasibility Study: After requirement gathering, the team comes up with a rough plan of software process. At this step the team analyzes if a software can be made to fulfill all requirements of the user and if there is any possibility of software being no more useful. It is found out, if the project is financially, practically and technologically feasible for the organization to take up. There are many algorithms available, which help the developers to conclude the feasibility of a software project

System Analysis: At this step the developers decide a roadmap of their plan and try to bring up the best software model suitable for the project. System analysis includes Understanding of software product limitations, learning system related problems or changes to be done in existing systems beforehand, identifying and addressing the impact of project on organization and personnel etc. The project team analyzes the scope of the project and plans the schedule and resources accordingly.

Software Design: Next step is to bring down whole knowledge of requirements and analysis on the desk and design the software product. The inputs from users and information gathered in requirement gathering phase are the inputs of this step. The output of this step comes in the form of two designs; logical design and physical design. Engineers produce meta-data and data dictionaries, logical diagrams, data-flow diagrams and in some cases pseudo codes.

Coding: This step is also known as programming phase. The implementation of software design starts in terms of writing program code in the suitable programming language and developing error-free executable programs efficiently.

Testing: An estimate says that 50% of whole software development process should be tested. Errors may ruin the software from critical level to its own removal. Software testing is done while coding by the developers and thorough testing is conducted by testing experts at various levels of code such as module testing, program testing, product testing, in-house testing and testing the product at user's end. Early discovery of errors and their remedy is the key to reliable software.

Integration: Software may need to be integrated with the libraries, databases and other program(s). This stage of SDLC is involved in the integration of software with outer world entities.

Implementation: This means installing the software on user machines. At times, software needs post-installation configurations at user end. Software is tested for portability and adaptability and integration related issues are solved during implementation.

Operation and Maintenance: This phase confirms the software operation in terms of more efficiency and less errors. If required, the users are trained on, or aided with the documentation on how to operate the software and how to keep the software operational. The software is maintained timely by updating the code according to the changes taking place in user end environment or technology. This phase may face challenges from hidden bugs and real-world unidentified problems.

4. Vison and scope

Project vision is the general fantastic thought of where the group or venture is going. The assertion then again is somewhat of a device that permits to impart this vision in a reasonable and compact way. To lay it out plainly, project vision articulation is a recorded adaptation of the project vision.

A project's scope and vision define the broad parameters of the project and provide the foundation for all subsequent steps in project or program cycle. A clear scope sets the rough boundaries for what the project will attempt to do.

The software main goal is to bulled up a correspondence between two individuals who need to lease that little device and who need to give that apparatus to him and there must be a admin, customer and seller. This is a basically a renting app. We all know about Daraj app where

there is a group and another group is sales. Where the customer can buy the things they need and the seller with the help of an offer sell their products. Admin mainly head of the software who rents the app.

The project vision can be tailored to cope with variation in industries and different levels of complexity in projects, yet, all visions share a common purpose. To be effective, the vision document should address the problem to be solved by the project and the benefits reaped from the solution. Since it is customer-centric, it must define customers and stakeholders affected by the project along with their needs. Stakeholders' needs guide the definition of product features and set the product boundaries (scope) which helps the project team prevent scope creep.

Define the Business Opportunity

We needs to describe the benefits reaped from completing the project. The project may result in higher competitive benchmark, or the expected annual revenues may be doubled by selling the product of the project. This step is vital to get management's buy-in and to authorize the project.

5. STAKEHOLDERS ANALYSIS:

5.1 Primary stakeholders

Who get directly benefit from this project those are,

Our Primary stakeholders are,

1. The owner,
2. Employees,
3. Users and
4. Business Partner.

5.2 Secondary stakeholders

Who get directly benefit from this project those are,

1. Online Sellers,
2. Competitors ,
3. Suppliers and

4. Vendors

1. Local communities and

2. Government

Identify Stakeholders and Users

Stakeholder analysis is critical to the success of any project. In this step, we needs to list all parties that are positively or negatively affected by the project outcome; referred to as Stakeholders. Every stakeholder should be identified along with his/her influence, role in the project, and the mechanism to leverage or mitigate his/her influence. On the other hand, user groups should be identified in terms of their responsibilities with respect to the system (product), the stakeholder group they relate to, and how they define the success of the solution to be developed.

Summarize Stakeholders' and Users' Needs

After stakeholders and users have been defined, we needs should be understood and documented. 'Needs' can be discovered by understanding key problems the stakeholders experience with the existing system. It is also important to understand priorities, as perceived by the stakeholders, to solve these problems.

Develop a Product Overview

The Product Overview defines the scope of the system and its interfaces with external parties. I personally prefer depicting the product overview using the Context Diagram, in which we can define how the system as a unit interacts with external stakeholders and users, and how information flows in and out of the system, from and to external parties. In addition to showing how the system is related to external stakeholders, the Context Diagram can be expanded to depict the relationships amongst internal system modules.

Define Product Features

Based on stakeholders' needs, we will be able to develop the high-level capabilities of the system that will meet these needs. Each feature should describe the functionality required in the system to meet one or more of the stakeholders' needs. For example, a need to quickly approve

accounting documents can be met by having a feature of workflow capability to route documents electronically for sign-off by authorized personnel.

List Assumptions and Constraints

In this step, we lists all project assumptions that if changed will alter the project vision. An assumption may state that a specific version of an operating system will be available at the time of installing the system. If this assumption proves false, the vision document may need to be revised. Besides, we should identify all limitations affecting the project. Constraints may be design-related, time and budget-related, environmental, or regulatory.

Define Documentation Requirements

Depending on system complexity and customer requirements, it may be required to provide supporting documentation as part of the project deliverables. Documentation includes user manuals, online help, installation guides, and Read Me files.

When the project vision is signed off, the customer and the we should have a clear vision of the project's product. This document is the starting point for the Software Requirements Specifications (SRS) in which detailed requirements are articulated to meet the product features. Hence, team members should refer to it frequently to ensure alignment with customer requirements and to prevent scope creep.

6. FUNCTIONAL REQUIREMENT

There are a lot of software requirements specifications included in the functional requirements of the OTR System, which contains various process, namely Registration, Check out, Report Generation, and Database.

6.1 Adding User

• Admin enables the   include new   user to the system.

•   Assigning an ID to the user

• Information of the User

• Availability of the product

• Mandatory   User Information

• Updating   User Information

6.2 APPS Rules

•   Transaction corrections, adjustments, and cancellations.

- Administrative functions.

- Authentication.

- Authorization levels.

- APP Tracking.

- External Interfaces.

7. NON FUNCTIONAL REQUIREMENT

7.1 Security requirement:

● User Identification: The system needs the patient to recognize herselfs or himself using the phone.

● Logon ID: Any users who make use of the system need to hold a Logon ID and password.

● Modifications: Any modifications like insert, delete, update, etc. for the database can be synchronized quickly and executed only by the ward administrator.

 ● Administrator rights: The administrator can view as well as alter any information in the System.

7.2   Performance

● Response Time: The system provides acknowledgment in just one second once the users is checked.

● Capacity: The system needs to support at least 1000 people at once.

● User-Interface: The user interface acknowledges within five seconds.

● Conformity: The system needs to ensure that the guidelines of the Microsoft accessibilities are followed.

7.3   Maintainability

● Back-Up: The system offers the efficiency for data backup.

● Errors: The system will track every mistake as well as keep a log of it.

7.4   Reliability

● Availability: The system is available all the time.

7.5 Keywords

Web Application, Online Platform, product Renting, Smart, Security, Advertise, Network, Remote Interaction.
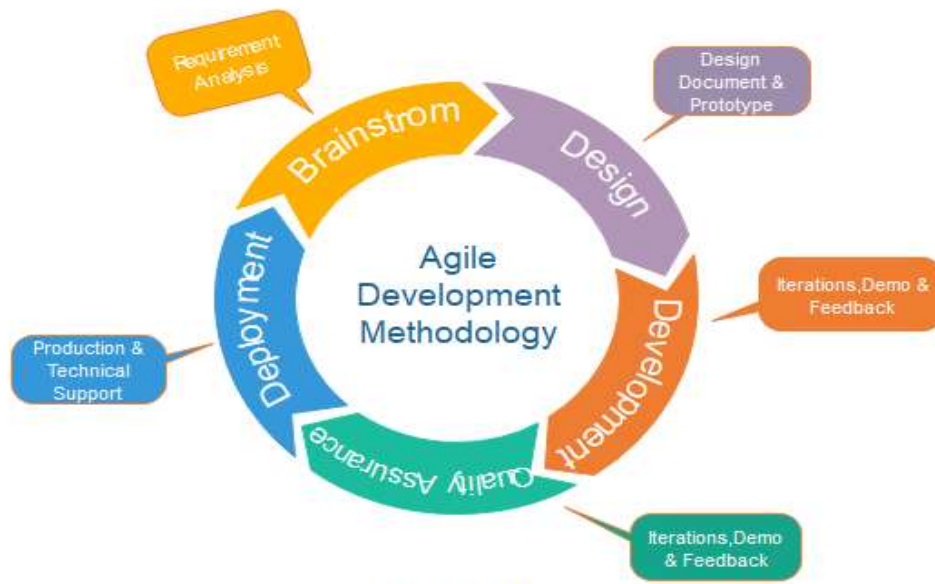
8. MODULES OF THE PROJECT

● User Invoice System .

● Buyer Management System.

● Seller Management System.

● Online Dealer Management System.

9. SYSTEM REQUIREMENT OF THE PROJECT

● Users Must have android phone

● App ,

● barcode scanner and

● printer machine.

10. PROJECT DESIGN

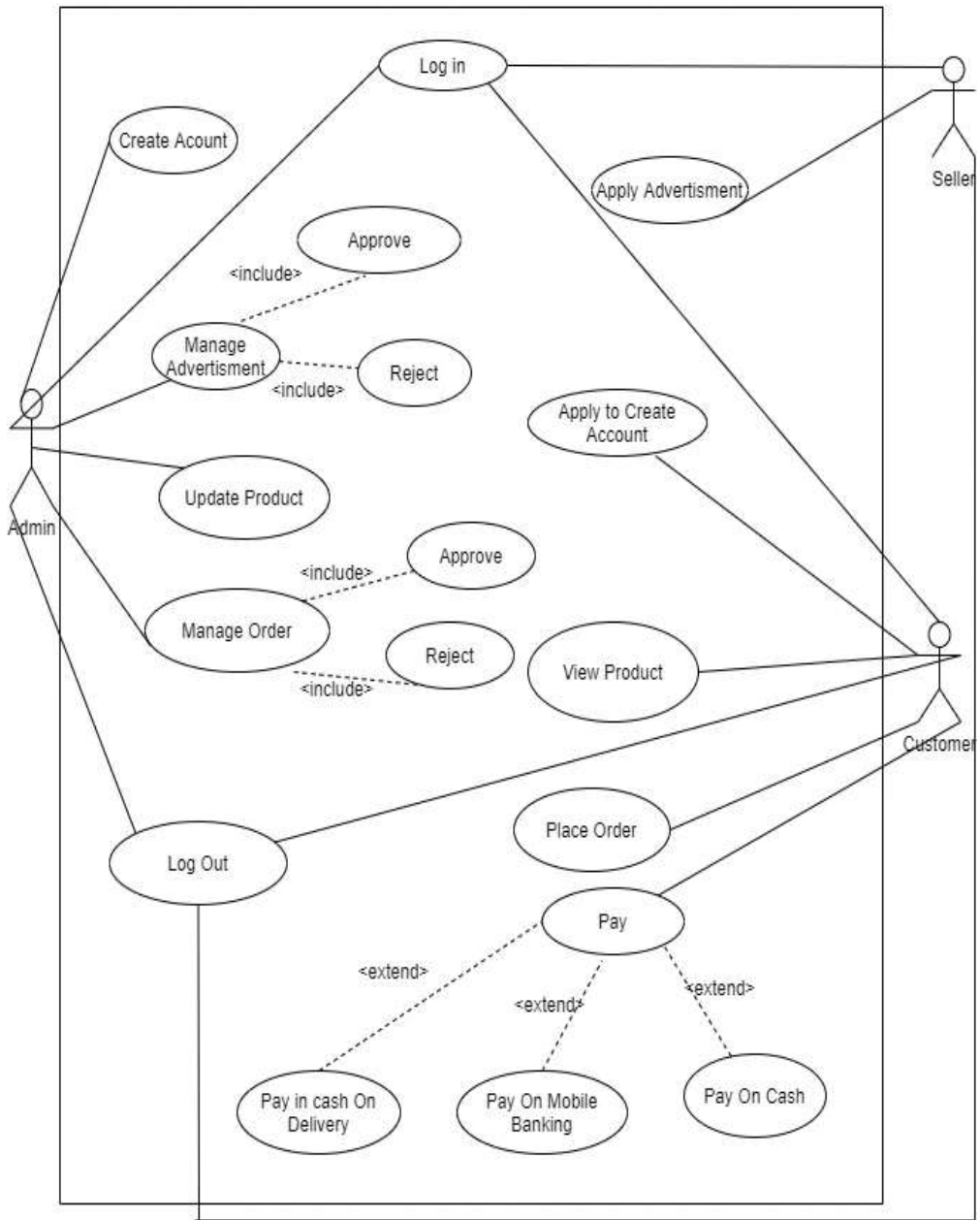For designing our project we select agile model .

Fig. Agile Model

Why we choose agile model for our project?

1.Suitable for small to medium
2.Extreme Programming
3.Can accommodate changes at any
Time.
4.Dynamic System
5.Effective for the dynamic
6.Development Method
development environment.

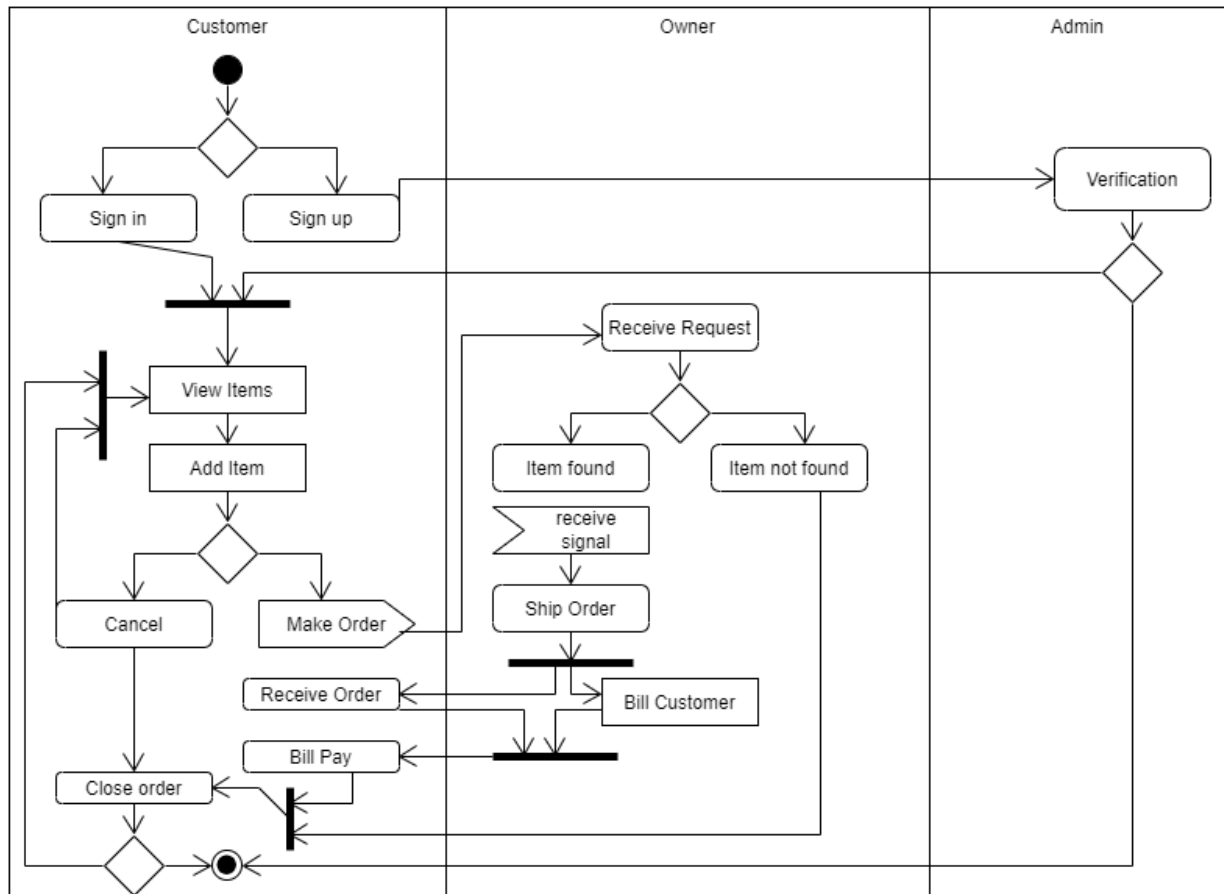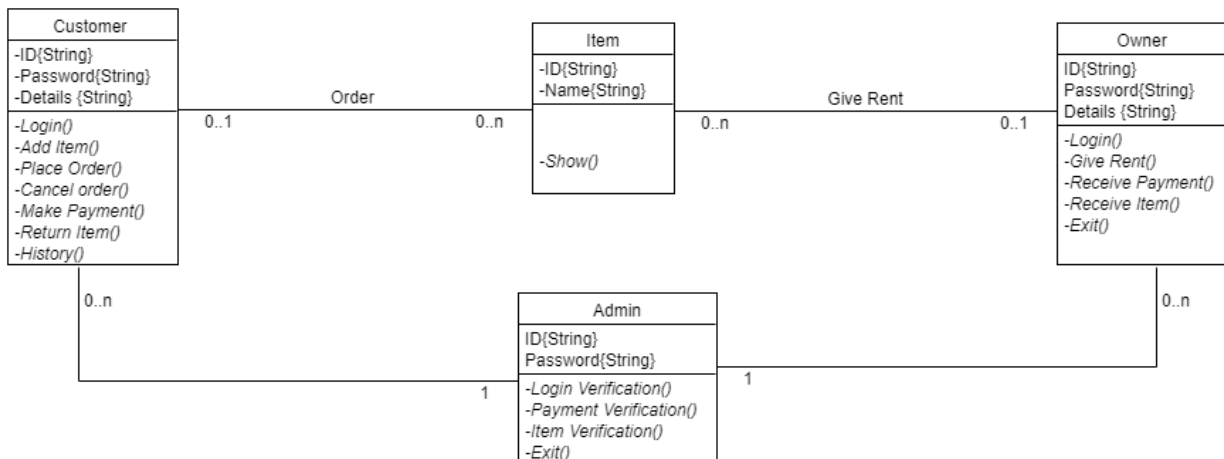## 11. Diagrams

- use-case diagram:

One Time Renting

Log in

Create Acount

Apply Advertisment

Approve

<include>

Manage
Advertisment

<include>

Reject

Apply to Create
Account

Update Product

Admin

Approve

<include>

Manage Order

Reject

View Product

<include>

Customer

Log Out

Place Order

Pay

<extend>

<extend>

<extend>

Pay in cash On
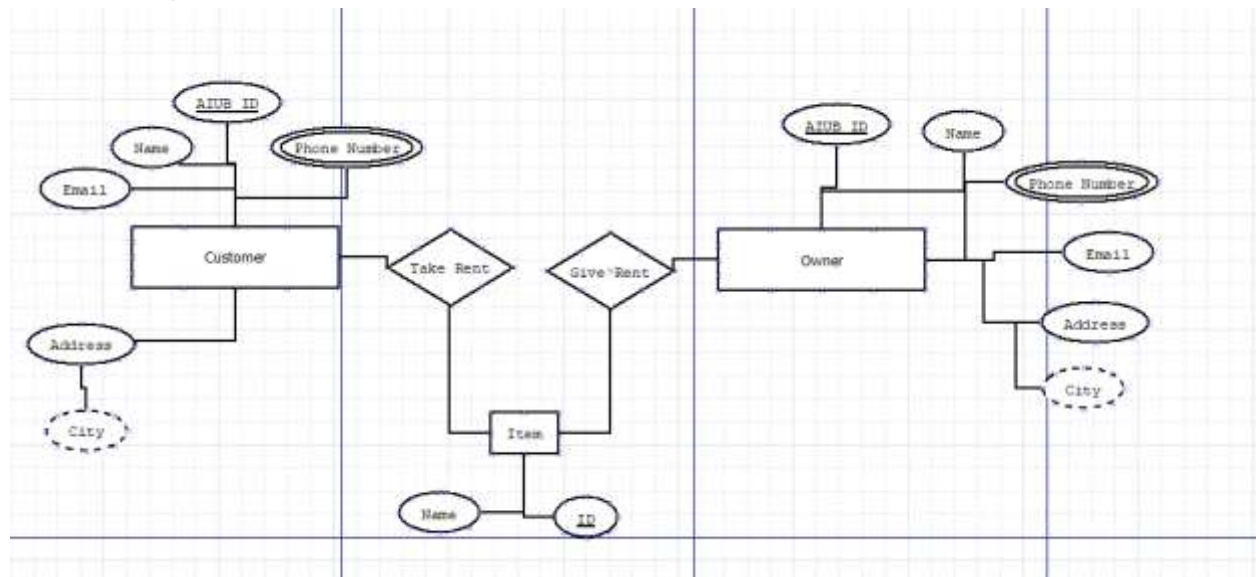Delivery

Pay On Mobile
Banking

Pay On Cash

Seller

● activity diagram:

● class diagram

- E-R diagram

12. Budget Estimation

 One time renting app is not that much big, so we can say it's an organic model, and let's assume we will develop in java language.

so , Source lines of code, SLOC=12000.

We know Effort =PM=Coefficient*(SLOC/1000)^P.

For organic project type,

Coefficient = 2.4, P = 1.05, T = 038.

So

Effort =PM =2.4*(12)^1.05   [SLOC/1000=12000/1000=12]

=32.61 staff-months

=33 staff-months

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

= 2.50*(33)^0.38

= 9.44 month

Let take it as 9 months

Required number of people=ST=PM/DM

= 33/9=3.67 = 4

Working hours per day for a single person=8 hours

Per person salary in a month=85,000

Per hour salary for a person in a month=85,000/180   [In a month working hours= 180]

=472.22

In 9 months number working days=180

Hours   =180*8

=1440 hours

So, Charge for the project 1440*472.22

=679996=680000

Requirement analysis =15days*8hours

=120hour

Charge for requirement analysis=120*300=36000

Travel expense=20000

Office rent expense=30000*9=270000.

Electricity & Gas bill=2000*9=18000

Training and hardware cost=100000

Maintenance for 9 months=9*8=72 hours

Maintenance cost=72 hours*1500

=108000

Utility cost:

Per month cost 5000 tk

Total = 9 months*5000 =45000 tk

Total cost=680000+36000+20000+270000+18000+100000+108000+45000

=1007000

Bill=Total cost+20%=1007000+201400=1208400=1208400.

13. Manpower   Requirement.

Manpower prerequisites show up at each step of our investigation as a basic and proceeding with an issue. Expecting to be the possible improvement of a sensibly the positive political environment, the stockpile of prepared labor might arise as it were unrealistic limitation in utilizations of tertiary recuperation. Such deficiencies exist all through the field of energy supply. In tertiary recuperation applications, nonetheless, likewise with any lacking innovation, there is an especially troublesome kind of roundabout the rationale at work: we anticipate labor deficiencies, when and as tertiary applications grow; however the advancement of improvement itself is now obliged by the absence of prepared labor (For example, in government financing organizations). For this situation, we might require a "crash program" assuming that quick advancement is to be\made insidethe following decade.

14. Project constraint of this document

 Project constraints affect the success or failure of a project directly. Project managers must be very cautious about monitoring and controlling project constraints throughout the project. Project constraint means a limitation or restriction by dictionary definition. Project constraint is similar in project management as well. Projects have constraints and the project manager must manage these project constraints.

There are seven   project constraints in our one Time Renting App project. These project constraints are as following in our project.

Cost:

Cost is very important project constraint. At the beginning of the project, project sponsor will be stating his or her expectations from the project. And also, he or she can come with a project cost constraint. For instance, he can say that "we need to complete this project with a 20 lacs taka". From this point onward, this 20 lacs taka budget will be your project constraint, and we have to complete the project under this budget.

Scope

Project scope defines what needs to be completed in a project, therefore it is a very important project constraint as well. For instance, if we are working in a software vendor, development of a e-commerce website can be a project scope. And, if the scope is "development of an end-to-end operating e-commerce retail website" this is an example of project scope constraint.

Quality

Quality comes with cost. If there will be strict quality requirements in a project, this will affect the cost project constraint directly. For instance, if we are working in a construction project, the customer might expect the building to be resistant against 8.0 Richter earth quake. Construction of a more resistant building will cost higher for sure. And if more quality is a project constraint, you should be aligned with this constraint in the project.

Customer Satisfaction

Customer satisfaction is a key factor for a sustainable and long-running business in the market. All companies do business for their customers in the end. Because, they are the source of money

and if we cannot satisfy our customers, you will not be able to stand against competition during time. Therefore, customer satisfaction is a crucial project constraint.

Risk

Risks can be either positive or negative in a project. Project managers must enhance the opportunities of positive risks and reduce the threats of negative risks in a project. For instance, if there is a risk of losing a project team member, we should prepare a handover document for the activities of that team member is doing in the project. And if he leaves the project, or company, or in other words, if the risk occurs, we can use the handover documents to assign tasks to a new team member and reduce the impact of this risk. Risk is a very critical project constraint in project management.

Resources

Many project managers think that only project team members are considered as project resources. However, tools, equipment, or material that will be used during the project are all project resources, and project constraints respectively. For instance, cement, bulldozer, or a scheduling software that will be used during the project are all project resources and project constraints respectively.

Time

Time is money! When starting a project, there will be deadlines that will be proposed by the project sponsor. For instance, project sponsor might state that "I want this project to be completed till end of next year". This will be your time constraint in the project.

15. Conclusion

Rent app has a great impact on the society. Rent app products have been known to save time and money of users. According to a study conducted by a newspaper, 95% people have to buy many products for one or two time using. This app can help them to rent product for some hour or days

which they prefer. As a result, they can save their money and they can use product that they need. You can rent your thing and you can have some if you need. It also can give you some money as well if we want to give your product in rent. We can also minimize the wastes because we don't have buy product for one time using. Besides that, lower income people have to think many times for buy product which they need for some time, they are available to rent product from this app. Sometimes it is seen that elderly people can't walk properly as a result if they any product for one time use, it's a realistic solution for them to use this app. This are the social impact of our product.

****** THE END ******