

13-9-2022 Tuesday.

Requirements:-

A requirement is a need, constraint, expectation or interface of any stakeholders that must be fulfilled by the proposed software product during its development.

User world

User of our system. Requirement is the bridge.

Software

Role of Requirement:-

- Agreement regarding the requirements between system developers, customers & end users. (legal-contract (flexible, inflexible))
- The basis for software design.
 - Defect-free as much as possible.
 - Technically feasible :-
- Support for verification & validation.
- Support for system evolution.

\Rightarrow System evolution = change (old system, new system).
change (old requirements, new requirements).

Source of Requirement

- Stakeholders
 - User of the system. (Human being)
- Document
 - (Process/Guideline)
- Existing system
 - (Text, of Requirements of the system).
- Application Domain
 - instruction taken from A.D.

Criteria of Good Requirements

- Necessary: If the system can meet prioritized real needs without refinement it isn't necessary.

Feasible:- The requirement is doable & can be accomplished with budget & schedule:-

Technical, in Time & Budget

Correct:- Technical with legally correct:-

The facts related to the requirement are accurate,

④ Concise ⑩ Allocated ⑪ Design Independent :-

⑤ Unambiguous ⑫ Redundant ⑬ written using the standard construct

⑥ Complete ⑦ Consistent

⑧ Verifiable ⑨ Traceable

Detail :-

Requirement Engineering:-

Engineering:-

It is the process of converting the specifications of customers into such artifacts that are used by artisans to produce the product that fulfills the customer specifications:-

Requirement Engineering:- The subset of systems engineering concerned with acquiring, developing, Acquiring, analyzing, trace, qualifying, communicating & managing requirements that define the system at successive levels of abstraction.

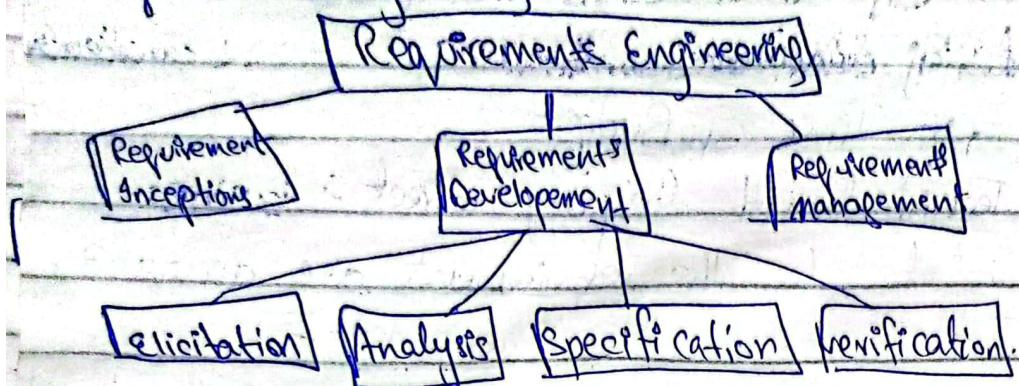
Requirement Engineering is about determining:-

- Problem with current state (As Is)

- Objectives to achieve.

- Changes of bringing about for a better future (To-Be)

Requirement Engineering Activities:-



Inception = Scope & vision of software.

Why Requirement-Engineering:-

To collect all the requirements of client:-

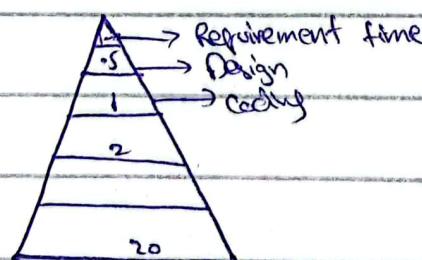
70% defects in specification phase.

30% Technical.

14 - 9 - 2021 Wednesday:-

High cost of Requirement Errors:-

If a unit cost of one is assigned to.



Error propagation in Lifecycle:-

Requirement Specification:- Maintenance:-

Design erroneous/correct /

Implementation

Testing

Types of Requirements:-

- ① Functional ② Product construction considerations
③ User related

Functional - that fulfill the set of selected business.

Deployment or programming coding.

Application stakeholders, existing (source consideration) :-
Documentation & prototypes

- ② Coding, implementation necessary to build the product.
③ Requirements for software development are provided from different sources. This classification is based on the agencies that provide the req.

Higher priority is the one who decide or give budget.

Based on functionality:-

- ① Core functionality ② Ancillary f.R.

75%

25% ↓

Sub divided into diff = req.

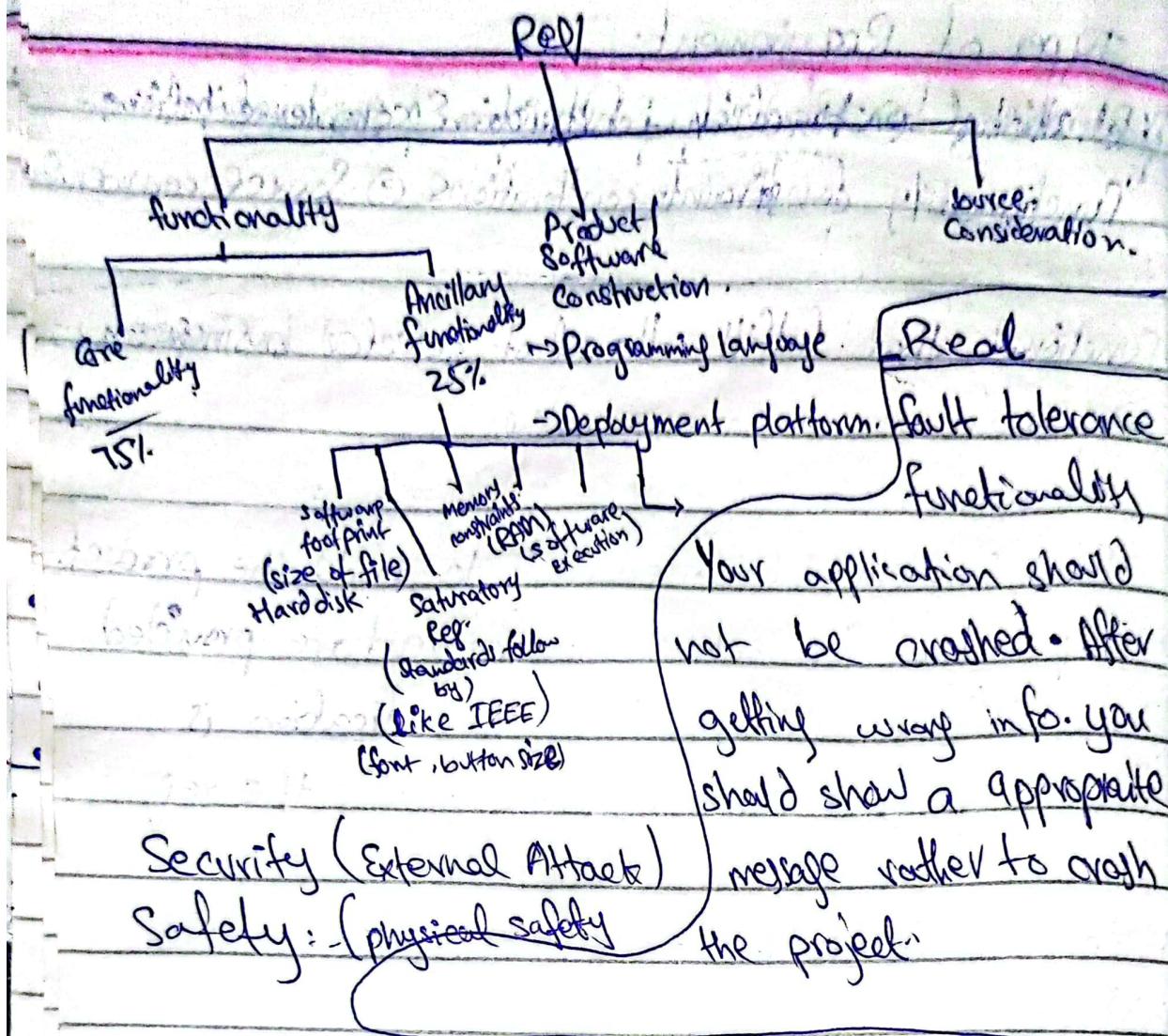
- ⇒ Memory ⇒ Software footprint ⇒ Fault Tolerance ⇒ Reliability ⇒ feel good
⇒ Esteem ⇒ Data integrity ⇒ Usability ⇒ Safety functionality
⇒ Satisfactorily

Safety functionality, physical safety,

Security - Data protected from intruders: External attack:-

Software footprint

Size of .exe file after making software



- Data Integrity Protection functionality.
the data that you are taking from your user must be correct.
- Examples:
 - End user Room or flight reservation:-
The data inputted from user must be correct.
 - Response time functionality :-
Image to text: if a task performed Eg their output received in what time.

Reliability functionality:- Design & Implementation

Inverse Req.

that should not be the part of system :-

Behavior of system should not change

not = inverse Req.

d d d d d d d d

feel Good / Usability / Esteem / One-upmanship /

(Chp 2 Req. Eng & Management)
writer: Spring (et)

feel Good :- User Interface should be easy

Esteem :- privilege.

The user should feel privilege after having

using software.

One-upmanship :- Comparative b/w two software - having
same kind of work.

① Requirement Process:-

② levels of Requirements:-

① what is Requirement process?

Series of steps to achieve the goal.

Requirement is the input :-

Garbage In, Garbage Out (GIGO)
If ur input is wrong or inappropriate
then (get right req.)
the output is also wrong.

Quality of process produces Quality of products



→ Requirement process:

- ① Req. Elicitation:-
- ② Req. Negotiation:-
- ③ Req. Specification (document):-
- ④ Req. Validation:- (complete, correct - 8)

Step 1:-

Stakeholder
Req.

System
Req.

(System
testing)

(Subsystem)
Req.

Component
Req.

Acceptance
testing

System
test

Integration
testing

Component
test

28-9-2022

Practical E^g Implementation Study:-

- Business Requirements • Product scope vision E^g project scope
- Add more customers [around money or financial]
- Increase Profit.
- Case Study :-
- * funding Sponsors :-

Customers just told about funct. Bcz they don't have any idea about business Rep.

Product: Complete picture E^g develop complete software
Project Scope: Set of Rep used in product E^g complete project scopes are called product scope:-

* Scopes is all the things in something on which we implement our code

(Business Rep /Scope E^g limitations /Business Context) - used in documentation :-

Business Rep-

① Background:-

the history & current situation & previous software info.

② Business opportunity:-

Is In User based increase, Increase in sale is objective (Quantitative E^g measurable objective) [why is it attractive in background].

Vision of staff squared:-

HR [Human Resource].

For [target customer]

Who [statement of need or opportunity]

The [Product Name] like HR Software

As [Product category]

That [major capabilities, key benefits (provided to customer)]

Unlike [primary competitive, current system].

Our product

- Success Metrics -

Is objective successfully achieved or not?

Criteria for Business Refg. The increase in Obj-

Criteria:- Condition is fulfill.

Vist Vision Statement:-

Customer = Scientists.

who = Need to Request Chemical:-

The = Chemical Tracking System:-

[title Members Name (Vision Statement)]

→ Business risks

Identify risks:-

Related to competitors -

→ Business assumptions & dependencies.

Your objective is to increase profit then ur assumption is 15% increase [visitors, purchaser etc] & on which ur product is dependent is dependencies.

→ Scope representation:

How scope represented to customer:-

→ diff. techniques.

① Context diagram ② Ecosystem map ③ Feature tree

④ event list.

Context diagram

① Identify external entities outside the system.

data, control & material flows

→ Top level diagram.

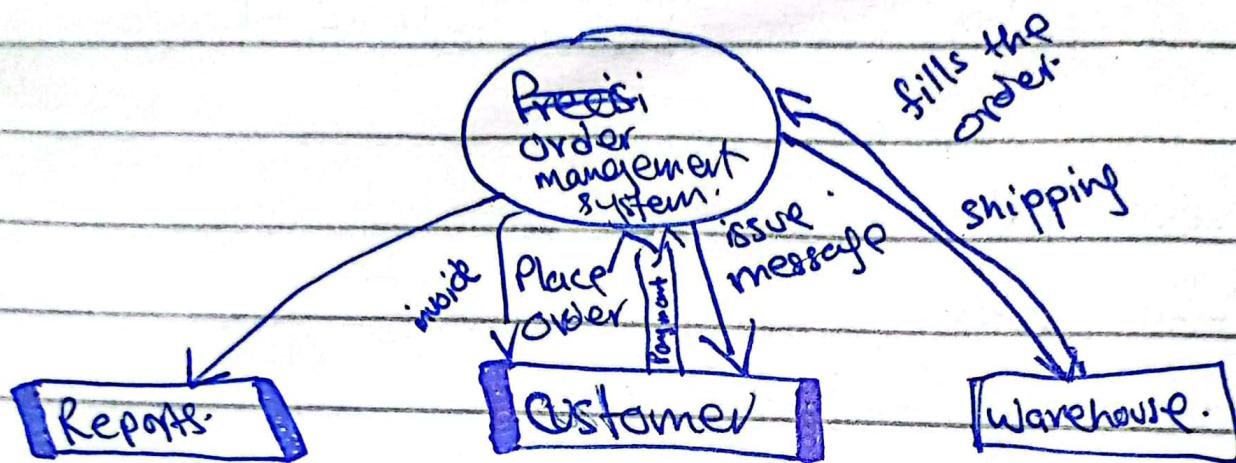
→ entire system depicted as Circle ○

Noun/ Individual/Hardware management, for organization

→ External Entities in the Rectangles can represent:

- User classes
- Organizations
- other systems
① hardware devices.

→ Arrow on the diagram represent flow of data.



Ch 5 [Chemical Tracking System]

Diagram, 3rd edition
Karl El Enjouy
Establishing the
business req-

→ Ecosystem Map:-

It is the interaction of our system with other systems:- after deploy:- without ^{external} entities:-

→ Entire system depicted in Bold Box.

→ External Systems by Simple boxes:-

→ Arrows:-

* only includes system:-

* Shows Indirect link.

Business Requirements :-

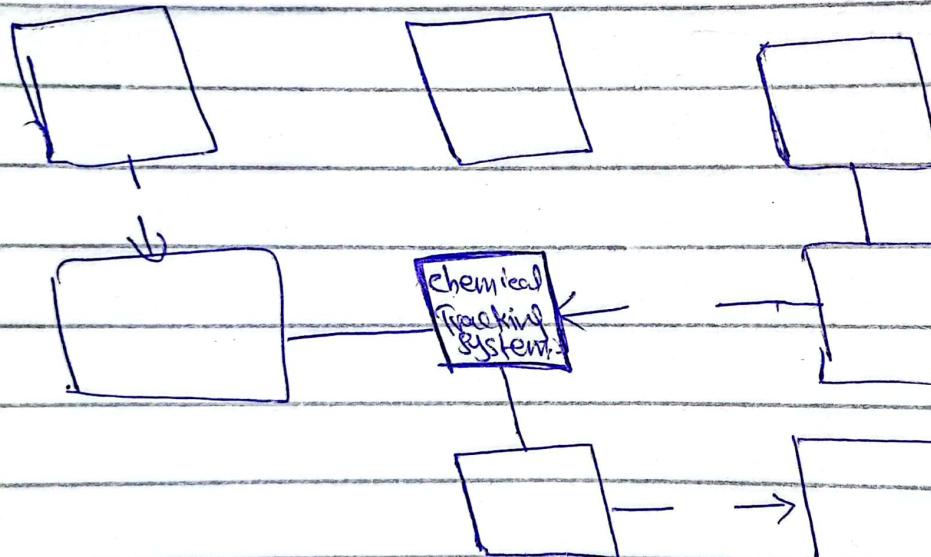
- ① Context diagram
- ② Ecosystem
- ③ Feature tree
- ④ event

Ecosystem:-

① as an entities only system is considered

② Indirect system:-

③ Bold box:-



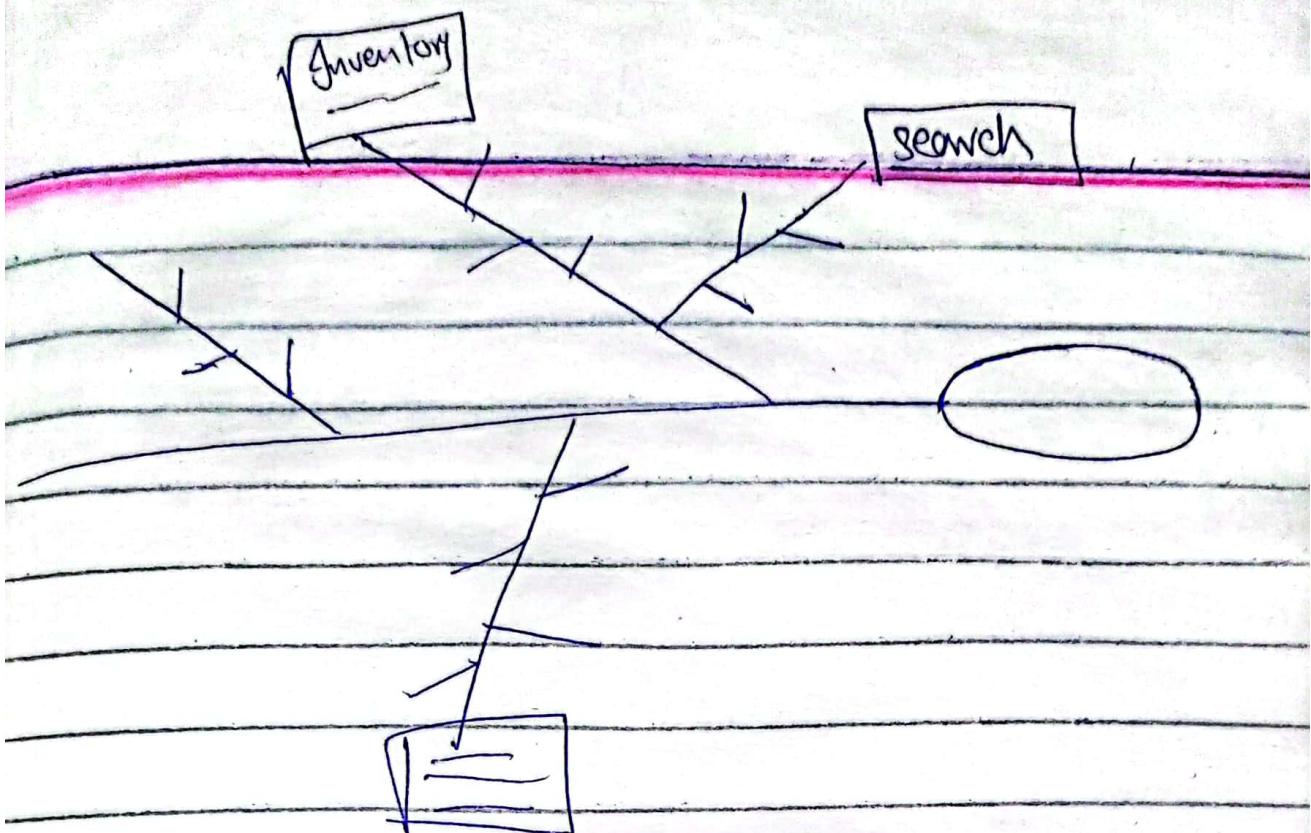
Feature tree:-

① ~~Scope~~ list all the features of system:-

② 3 level feature:-

③ 1L (Main feature of product)

2L, 3L are sub levels



① ~~ON~~al system: (Main name of system).

② Blue boxes (level one).

③ ~~Be~~ Grey boxes (level two).

④ not boxes are (level three)

* Event list :-

Identifying ~~by~~ external entities - (Triggers
identifies boundaries).

① Business trigger (managed) ② Time triggered (date & time)

③ Signal events received from external:-

components, such as hardware devices, (credit
card & debit card).

Scope representation context:-

- ① Context diagram
- ② Ecosystem map
- ③ Feature tree → features
- ④ Event trigger → list level trigger

Scoping is easy:-

(RMP 6)

Theoretically

Practically:-

Requirement Elicitation:-

Chp 3, 5, 7 in different books

① Elicitation Techniques Gathering:-

② Personal interviews

③ Questionnaires

④ Customer Marketing surveys

⑤ Observations

⑥ Demonstration of product prototypes or the

⑦ Product itself.

⑧ Brainstorming

what is Elicitation:- WBI (Highlight/Draw out.)

Evoke

CMMI - Systemic techniques | User

Gathering

- ① Scattered form information collected at one place?
- ② Documents:-

- ① Stakeholders are in Elicitation:-

- ② Documentation Eg. Existing systems Eg. Standards Eg. guideline.

- ③ Personal Interviews conduct on agile method Maximum 3 stakeholders so we have to interview them.

If 12 to more then we fill survey forms :-

- ④ Planning ⑤ Notes prepared ⑥ Qs prepared.

- ⑦ Appointment:-

stake holders-	Technique
Name (1)	Interview
(2)	Interview
(3)	Interview
S-10 > 10:	Survey.
Student (>100)	
Ward en (1-2).	

- ① Establish rapport (Introduce yourself).
- ② Stay in scope (Ans around your Qs)
- ③ Prepare your question.
- ④ Suggest Ideas
- ⑤ Listen carefully:-

* Questionnaire:-

If stakeholders are of 1000 or more than 3 then we can't take interview - we will have survey forms:-

→ Software already existed:-

→ Compare features:-

→ Add feature which you feel is important.

→ The features that are not essential:-

→ If they buy our product or not - by seeing our features.

Tips in context of Questions:-

→ answers mutually exclusive - write everything interestingly

→ Closed Qs - (To the point, specific point)

Don't ask too many Qs:-

Customer / Market Survey :-

3 Methods to conduct Survey form:-

① face to face

② postal Method

③ web based survey:

(Monkey Survey)

Observation:-

→ Active or Passive both

→ Ask Qs Eg. perform

Benefit

Drawback

Practical Insight:-

They imitate:-

* Demonstration of Product Prototypes or the product itself.

→ finalize requirement than to obtain original information.

Used to check whether the requirements which we want is present in requirement document or not

So, the customer check this by asking to make prototype of the proposed system.

→ two types

→ Use & discard prototypes (complete GUI: Visio, Excel, PPT etc/stand after that it's just for giving idea to users)

→ Use & improve prototypes (MVP) (Minimal viral product). (analyze the information)

Use in project development scenario largely.

- also valid for the new product which does not exist in the market.

④ - * Brain Storming:-

→ Collecting of ideas about a new project to enumerate the features Eg for a existing project for to enumerate the additional features- by having a informal environment meeting of a large number of knowledge persons to explore their ideas about that project or proposed system Eg after that collecting data, the data is then analyzed Eg pin out the right requirement Eg use it.

* Information Gathering:-

- → Organisation records
- → Process Documentation
- → Standard Guidelines
- → Customer Satisfaction Survey.
- → Customer Complaints.

→ Elicitation or Gathering:-

→ Sometimes Elicitation or Sometimes Gathering

→ Both imp for project Req.

- * Delivirables (Output of Elicit & Gather)
 - Notes taken during personal interviews.
 - Responses to Questionnaires (by form)
 - By Postal Method
 - Responses by surveys.
 - formats & templates:-
 - Information culled from studying the organization records.
 - Organization documentation
 - flowcharts
 - Input Output
 - Information notes how to convert input into output.

Elicitation Planning:-

- Elicitation objectives:-
- Strategy of planned techniques.
- Schedule of resources estimated
- Documents of systems needed for independent elicitation.
- Expected elicitation:-
- Elicitation Risks:-

Pitfalls in Eli -& Eg Gath -

Business Analyst: The one who conduct interview having a experience - Unexperienced persons are avoided

- Bringing in consideration of software design while capturing project requirement
- Time & budget. (cannot ask answer the time & budget at that time)
- Not preparing well for personal interview.
- Prejudices (wiseness, the info is true or not)

Use case Modeling:-

Extend Include {in case of reusability}

Default

• dotted line;

----> Base usecase

• Base -----
usecase

• Usecase is in descriptive as well as diagrammatic form:-

• Tabular form.

→ Includes basic functionality, alternatives, error condition, precondition, post condition.

Pre condition:- State of system at the start of the use case

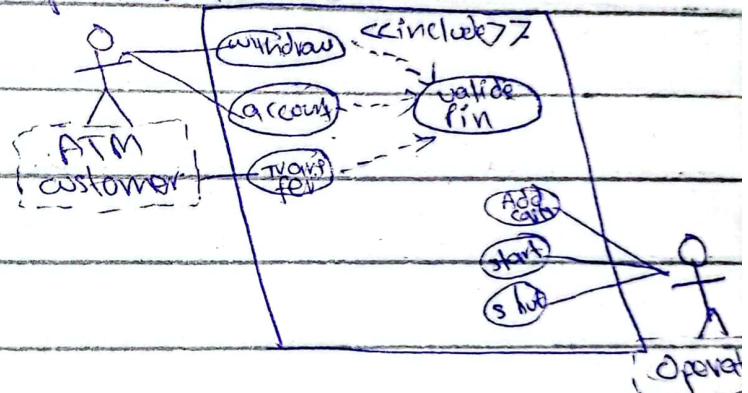
Post condition:- end of the use case.

Flow of events:- a series of declarative statements listing the step of the use case from actor's point of view

Alternatives allow different sequence of events.

Descriptive Use Case

- ① ID's UC-001
- ② Usecase Name Validate Pin
- ③ Usecase Actor ATM customer.
- ④ Usecase Pre-condition (satisfy usecase) ATM is idle, displaying a welcome message.
- ⑤ flow of events
- ⑥ Alternates Depend on other usecases...
- ⑦ Dependency:
- ⑧ Post-condition: Customer pin validate



flow of Events: basic path:-

- ① validate pin successfully fully/abstract (below)
- ② Customer EV insert the card into card Reader

2) Recognize card Eg reads number

3)

(Above)

A include usecase name abstract usecase.

Withdraw funds

ID : UC-002.

Name : withdraw funds.

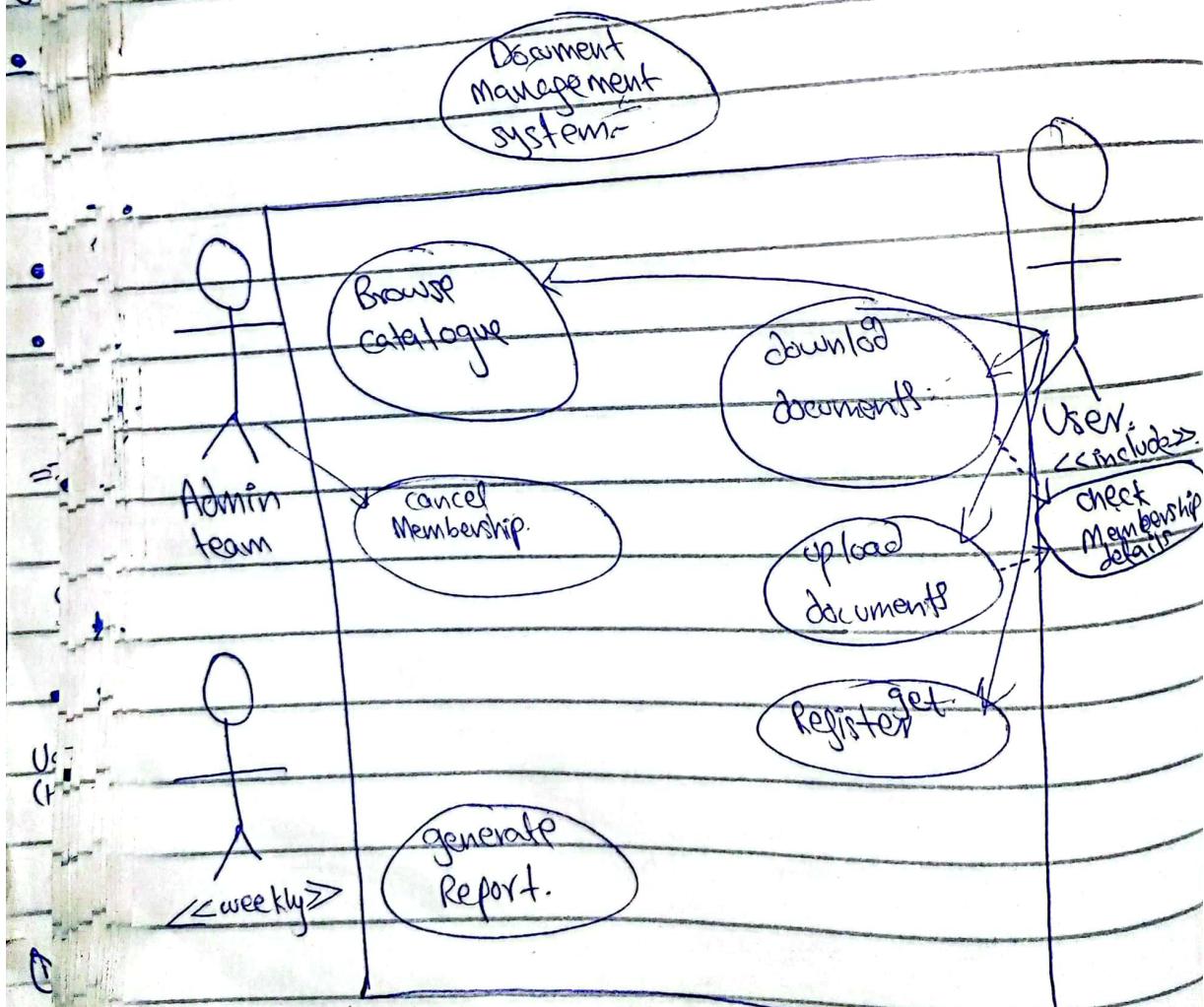
Actor: ATM customer,

Dependency: Validate Pin.

Pre-condition: Cash is present & funds available.

Flow chart of events: ① Insert card ② Enter pin ③

flow chart of events (successfull Path):



Event having time based like weekly is trigger.
also show as actor.

Analysis of the information collected in Req. Elicitation or Gathering.

- ① Enumerate all Requirements. (combine)
 - ② Verify each Req. completeness
 - ③ Evaluate each Req. for its feasibility
 - a. Technical (logic algorithm)
 - b. financially
 - c. Timeline
 - ④ Bifurcate req. into
 - a. Core functionality Req.
 - b. Ancillary " "
 - ⑤ Group core func. Req together into logical groups.
 - ⑥ c. Ancillary " " into their " "
 - ⑦ Identify Req. that are duplicate.
 - ⑧ Identify Req. that are contradictory to each other (negate with stakeholder)
 - ⑨ Identify system interfaces-
-
- ① Identify stakeholders breach Req.
 - a) Primary or b) secondary
 - ② Prioritize Req.
 - a) Timeline b) financial c) Technical
 - ③ Identify gaps in the case of roll of product implementation.
 - ④ B/w product & organization needs
 - ⑤
-
- ⑥
 - ⑦ Determine the schedule of implementation for Req.
 - ⑧ Resolve issues uncovered in the above activities