TELEPHONE INDEX IT 314 Thursday | 2/1/20

NAME & ADDRESS	PHONES	MOBILE	E-MAIL
25%. Course Project		(15 +	10)
25%. End Sem			
15%. First Insem			
15%. Second Insem			
20%. Experiments + Exercises + Quizzes. (In Lab)			

→ There will be bonus marks (max 10%) for undertaking additional work on course objectives

3 evaluations:-
1) After 1st Insem
2) After 2nd Insem
3) After endsem.

Software includes interlinking of many programs.

SEN is a combo of:-
(i) Engineering
(ii) Management Science (planning) } Management of resources is essential (tools, time, space, cost, etc)

(-2)

A software is a computer program along with the associated documents and the configuration data that makes them operate correctly.

Software Requirements:-

1) Functionality should be complete - Functional
2) Ease of use.
3) Performance and quality } - Non-functional
4) Should be adaptible to change
5) Specific requirements - eg:- safety critical.

SRS - Contractual document

There are 5 phases in software development process:-

(i) Requirements Engineering
(ii) Design
(iii) Implementation
~~(iv) Maintenance~~
(iv) Verification and Validation.
(v) Maintenance

Productivity is used to asses feasibility of a project

→ Product is for end-user. Project is for a client.

Software Process Models:-

1) Waterfall Model

We start from requirement phase and go till deployment phase.

Wednesday | 8/1/20

User Requirements helps to determine functionalities reqd by user. But someone has to go and talk to determine what exactly is required. That software req is documented and contract (SRS) is signed.

⇒ In Waterfall Model, requirements are determined and frozen. They can't be changed after we enter design phase. Since we do every step one after another, former has to be completed before the latter.

Disadvantages

In waterfall model, we can't iterate. Only once done matlab done. Since process can be 1 year long and once we get out of req phase, there is no turning back, customer will not get any regular feedback, since in waterfall model that is useless. So, clients need to have patience.

Prototyping Model

Don't freeze model, create a prototype and improve iteratively. Unlike waterfall model, clients can get regular feedback checks.

Disadvantages :- Sometimes, even changing prototype is not very feasible and can be frozen. This is called short-sightedness.

11	November 2019						
wk	M	T	W	T	F	S	S
44					1	2	3
45	4	5	6	7	8	9	10
46	11	12	13	14	15	16	17
47	18	19	20	21	22	23	24
48	25	26	27	28	29	30	

12	December 2019						
wk	M	T	W	T	F	S	S
48	30	31					1
49	2	3	4	5	6	7	8
50	9	10	11	12	13	14	15
51	16	17	18	19	20	21	22
52	23	24	25	26	27	28	29

In throw away, we only use prototyping to understand the initial requirements. Requirements are refined and customer checks. After variations and iterations, if works, well and good, otherwise throw it away.

⟹ We use waterfall when every req is well-known before. When clients aren't actually sure what they want and iterations may be required, prototyping is more useful.

Incremental :- When set of requirements is very large, it's not possible to build everything in one go. We go on developing releases. Each new release covers requirements of old releases and also new requirements.

⟹ All the process models till now don't have any risk assessment. Risk may be related to loss of human life, or maybe other kinds of things. Spiral takes care of that.

29 Sunday

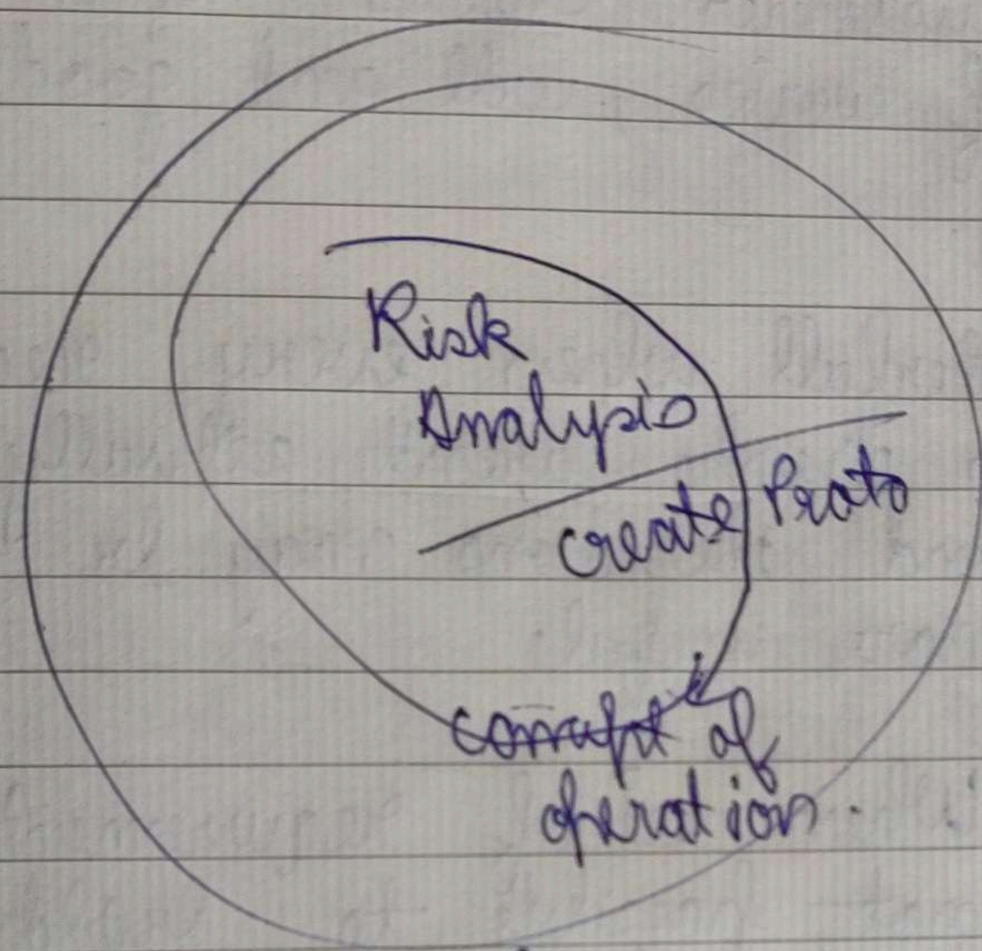
Q1) ⟹ Waterfall
Evolu Proto
Waterfall
Proto.

Spiral Model :-

Thursday | 9/1/20

P.T.O

09.00 USP of spiral model is that it allows us to ascertain the risk.



03.00 To do risk analysis however, we need domain experts.

04.00 Synchronize & Stabilize Model (Microsoft)

05.00 1) For requirement analysis, we interview potential customers. Creation of product is not for single user

06.00 → Go an incremental model.

07.00 2) We do with 3-4 builds, each build keeps adding functionalities.

There are 2 perspectives for every system. One is user-specific which is user's perspective of what functionalities are required and other is system's perspective of required functionalities.

Thursday 19/1/20

(6)

⟹ In order to model systems from various perspectives, UML is standardly used.

RUP

Agile Software Dev

There are some shortcomings in all these process models.

↳ In all above, the process is central. Problem is that we can't test the product very early. Also customer is often ignored. Customer se ek baar info lene lee baad wah side fe. That is not good.

→ involves customers at all stages.

Xtreme Programming (XP) and Scrum work on agile software dev.

Product backlog is collection of user stories.

In normal software dev, developers and testers do their work, one after another. But in Test-driven software dev, testers and developers do it together.

Ticket Distributor :-

Functionality-

1) Selecting trip (single, multiple)
2) Payment calculation.
3) Identification of child | adult

01 January 2020

wk	M	T	W	T	F	S	S
01			1	2	3	4	5
02	6	7	8	9	10	11	12
03	13	14	15	16	17	18	19
04	20	21	22	23	24	25	26
05	27	28	29	30	31		

02 February 2020

wk	M	T	W	T	F	S	S
05						1	2
06	3	4	5	6	7	8	9
07	10	11	12	13	14	15	16
08	17	18	19	20	21	22	23
09	24	25	26	27	28	29	

First user stories :-

As a traveller, I need a machine which can issue ticket such that I can purchase ticket.

↓

bigger functionality.

Issue Ticket

1) Select day/date
2) Single trip / multiple trip
3) Select child / adult
4) Computes price
5) Payment
6) Exception handling.

OR

1) issue ticket

→ select a ticket — day, week

- Single trip — child, adult
- Multiple trip — child, adult

→ Price

→ Payment

→ Handling exception — running out of ticket, changes

This hierarchical way of doing things is called concept mapping.

(-8)

Thursday | 9/1/20

All requirements may not be independent. At a time, it is possible that one requirement may depend on the other.

Thursday | 16/1/20

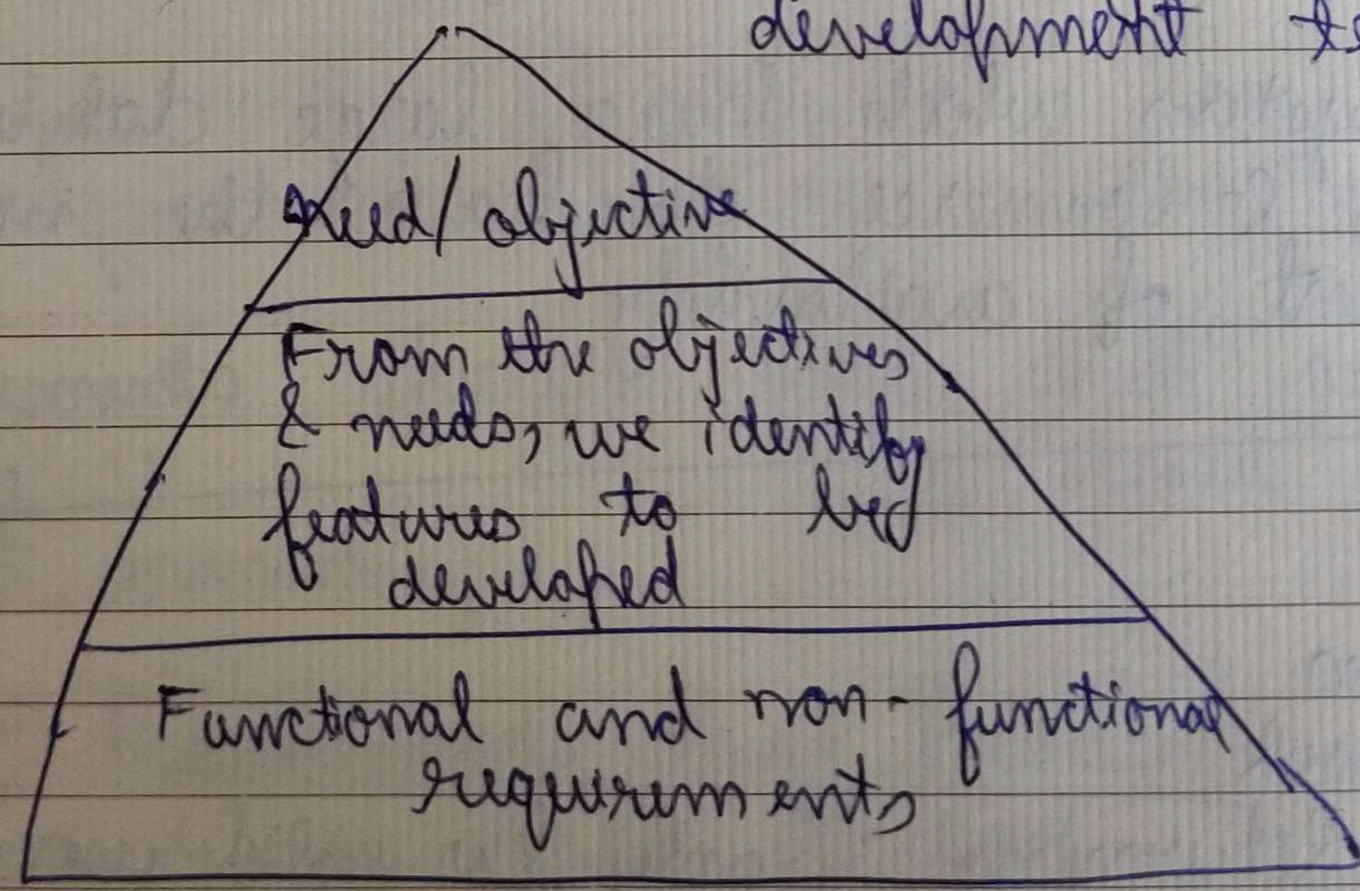
Set of requirements has to be understood from a problem context. Every software will have different requirements. Goal is to reduce gap b/w demanded requirements and delivered software.

1st phase of S/W dev - Requirements Engineering.

Understanding requirements

Systematic documenting

required by the design and development teams.



The understood development by the developer and the client sign the contract.

⇒ Availability comes under reliability. Eg:- System has to work whenever required. That is, it has to be available. This is called reliability.

⇒ **Robustness:-** A system should work correctly even if there is some malfunction.

3 kinds of requirements:- (Requirement specifications)

1) User Requirements · Written in natural language + diagrams · Written for customers.

2) System Requirements · Contract b/w client and contractor · Details all descriptions of the system services.

3) Software specification : A software description, which acts as a reference basis for developers.

⇒ Any system which has a large database. Eg:- banking, e-commerce etc have the non-functional requirement of availability.

(i) User Reg
(ii) User Login
(iii) Search book
(iv) Issuing / Returning (only for valid users)
(v) Reissue.

requirement in addition:- Max no of days books can be issued.

(i) Security

(10)

20/1/20
Monday

09.00 In Prototyping, requirements aren't exactly known. Prototyping me, we gradually understand requi-
10.00 rements. In spiral, we know requirements and build gradually in releases. Simultaneously, we
11.00 identify the risks from the requirements.

⇒ Requirement documents, list both functional and
12.00 non-functional requirements. Users are various:-

01.00 (i) Managers
(ii) System Design Thinkers
02.00 (iii) System Engineers

03.00 ⇒ For requirements, if set is very large, we prioritize
04.00 some requirements, depending on the use cases

⇒ Requirements specified with "shall" are mandatory,
05.00 while " " " "should" are desirable but not mandatory.

22 Sunday ⇒ Requirements Engg Process

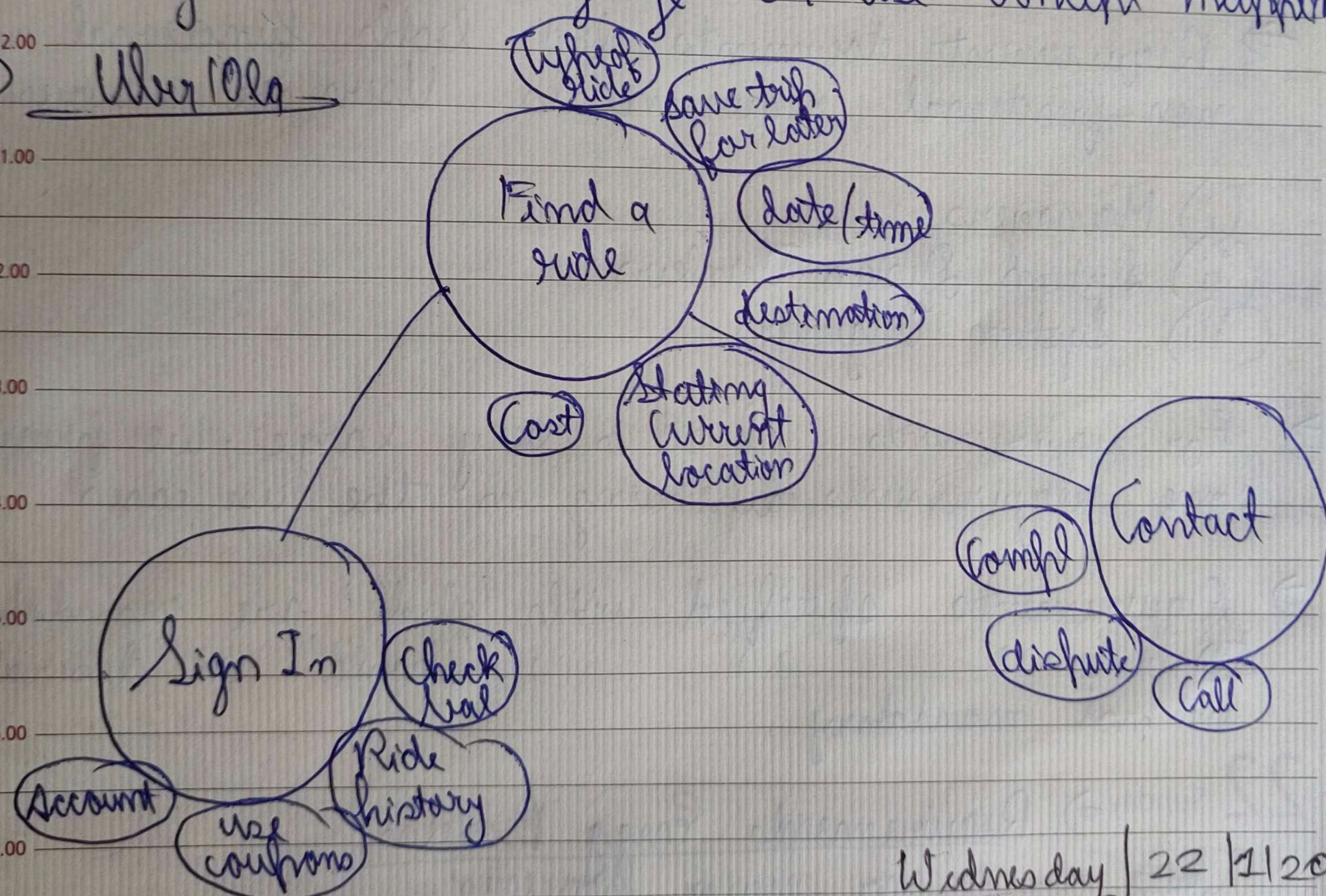
There are 4 major processes:-
(i) Feasibility study - feasibility report is developed to check it.
(ii) Requirements elicitation (discovery); gathering and compiling features of related requirements.
(iii)
(iv) Validation.

Monday / 20/1/20

⇒ Users can be of a couple of types. But stakeholders determine the characteristics of the system. Problem is that stakeholders themselves don't know what they want.

⇒ We can gather requirements directly from user, using natural language. Or use concept mapping.

⇒ Uber/Ola



Wednesday | 22/1/20

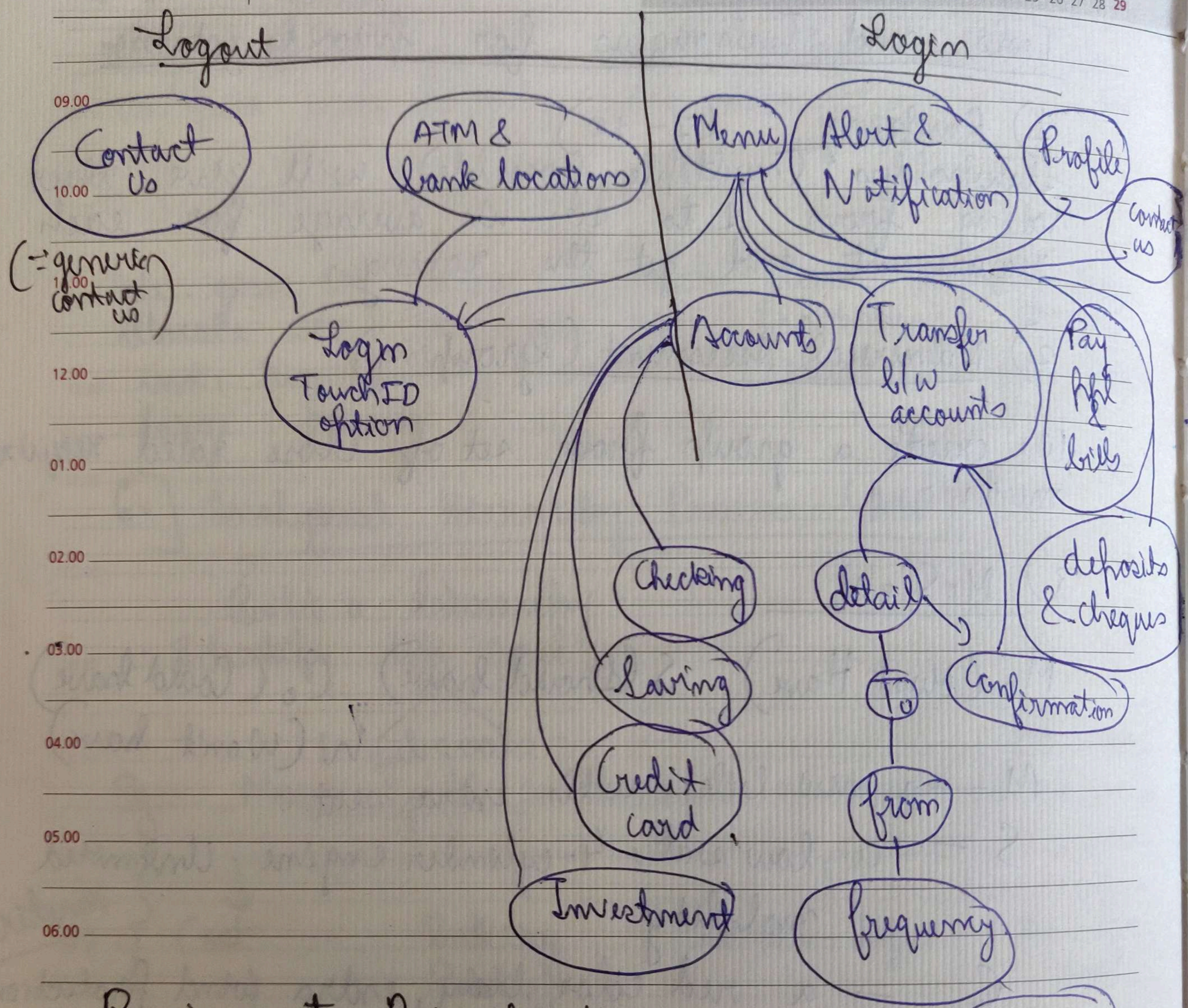
Insem 1 :- Concept Mapping

If there is a mobile app for a banking system, there may be various concepts involved.

P.T.O

11	November 2019						
wk	M	T	W	T	F	S	S
44					1	2	3
45	4	5	6	7	8	9	10
46	11	12	13	14	15	16	17
47	18	19	20	21	22	23	24
48	25	26	27	28	29	30	

12	December 2019						
wk	M	T	W	T	F	S	S
48	30	31					1
49	2	3	4	5	6	7	8
50	9	10	11	12	13	14	15
51	16	17	18	19	20	21	22
52	23	24	25	26	27	28	29



Requirements Prioritization

Book by BABOK. Stakeholders prioritize 85% of their requirements at high priority.

Factors driving prioritization:-

1) Personal value of the item
2) Risk Factor
3) Difficulty
4) Likelihood of getting work done on time
5) A must have

Tools and Techniques for prioritization

1) Ranking (1-10)

Stakeholders (multiple possible) will give their rating from 1 to 10. We average for each requirement and get the rating.

2) Numerical Assessment (Grouping)

We create a group from set of close rated requirements.

3) MoSCoW

Mo (Must Have) S (Should have) Co (Could have) W (Won't have)

M — a new bike, with extra seat

S — a low seat, 4-cylinder engine. Unlimited mobility

C — a red color body, extra wind protection

W — Bluetooth and helmet with double ventilation system

↳ doesn't mean that we reject these requirements. Only that we don't take it as a priority in 1st try.

4) Bubble Sort

Largest number in the set is called bubble. We compare each priority with other.

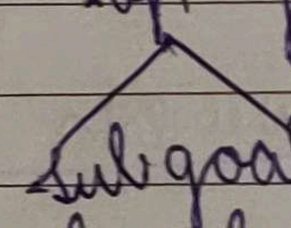
5.] Hundred dollar Method

We give 100 dollars to each client logically. They allocate 100$ to each req (fractions). At the end, requirements with highest dollars have large priority.

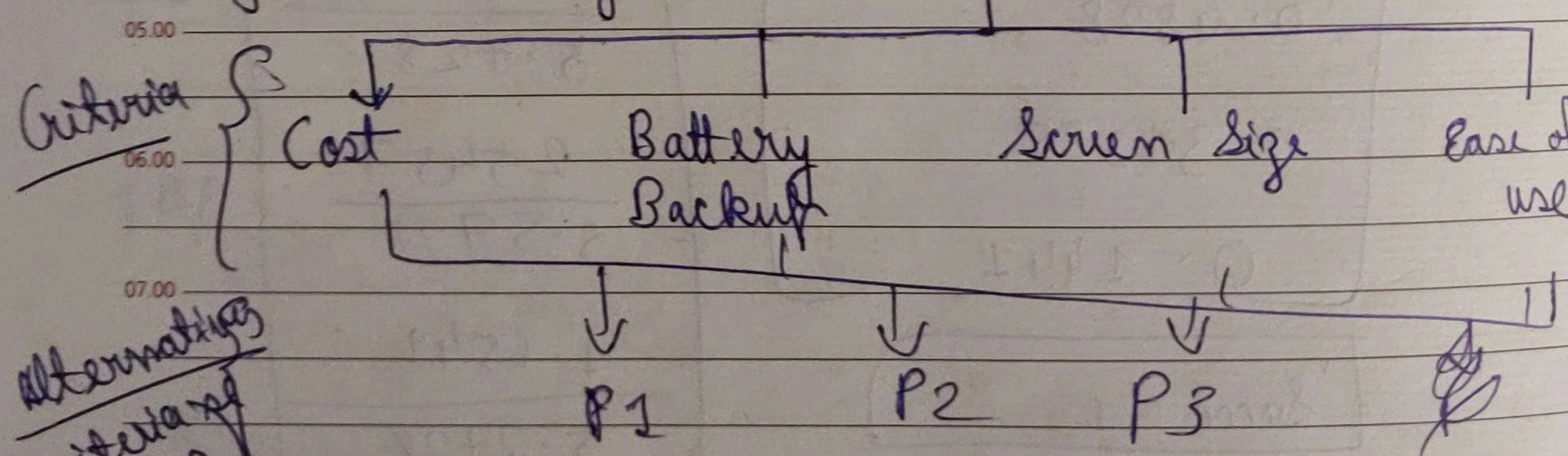
6.] Analytical Hierarchy Process (AHP)

Build a hierarchy.

- Identify top goal



Eg:- Main Goal :- Purchase a smart phone



(criteria of options shortlist krip)

A1 Matrix

	Cost	BB	SS	EU
Cost	1	1/7	1/5	1/3
BB	7	1	2	3
SS	5	1/2	1	3
EU	3	1/3	1/3	1

Intensity of Imp | Definition

09.00 1 | Equal Imp

10.00

11.00

12.00 Cost $= \left(1 \times 1/7 \times 1/5 \times \frac{1}{3}\right)\left(1/4\right) =$ 0.31239

01.00 2.5457

02.00 1.6549

0.75984

03.00 5.2728

04.00 A2 $\begin{bmatrix} 0.059246 \\ 0.4828 \\ 0.31385 \\ 0.1441 \end{bmatrix}$ Weights

05.00 $\begin{bmatrix} \frac{0.31239}{5.728} \\ \frac{2.5457}{5.5728} \\ \frac{1.6549}{5.2728} \\ \frac{0.75984}{5.728} \end{bmatrix}$ Cost, BB, SS, EU

06.00

07.00 Sum = 1

P.T.O

Wednesday | 22/1/20

	Cost ↓	BB ↑	SS ↓	Eu ↑
P1	15000	4000	5.6	Better (5)
P2	11000	3000	6	Poor (2)
P3	16000	2500	5.2	Excellent (8)

	Cost	BB	SS	Eu
P1	11k/15k = 0.733	4k/4k = 1	5.2/5.6 = 0.92857	5/8 0.625
P2	11k/11k	300k/4k	5.2/6	3/8
P3	11k/16k	2500/4k	5.2/5.2	8/8

⇒ Synthesis

P1 = (0.7333 * 0.059246) + (0.4828 * 1) + (0.92357 * 0.3185) + (0.1441 * 0.625) = 0.90774

15 Sunday

P2 = 0.7474

P3 = 0.80043

Thursday | 23/1/20

Scenarios :-

Scenarios are real-life examples of how a system can be used.

They include:- (5 things)

- A description of starting situations
- A " " the normal flow of events.

- A description of what can go wrong,
- Info about other concurrent activities
- A description of the state when scenario finishes.

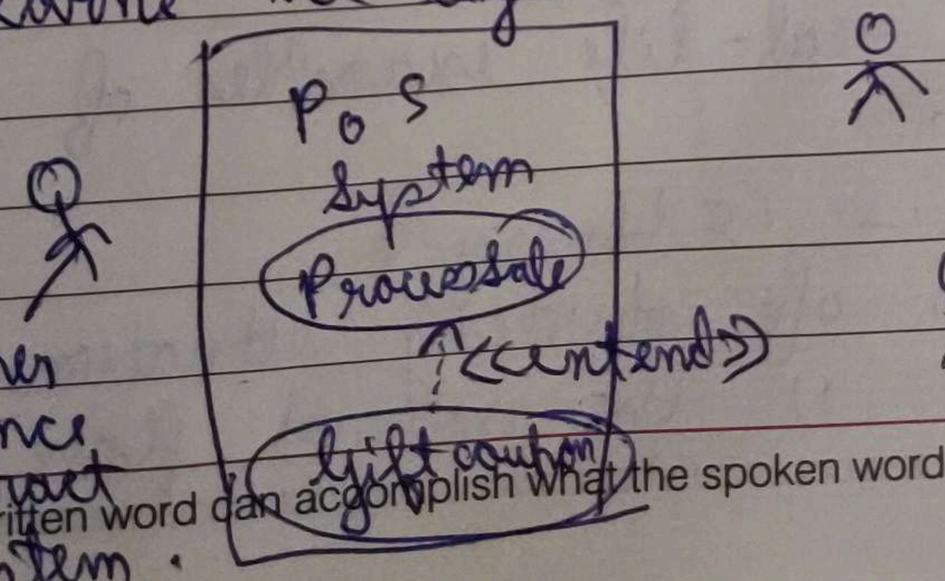
Once we have identified the requirements, we need to identify the scenarios where those requirements come into picture.

UML- Unified Modelling language.
Use-Cases are a scenario based technique in the UML which identify the actors in an interaction and which describe the interaction itself.

⇒ Users are the actual actors in the system. We need to know their interactions with the system. Note that stakeholders may not interact with the system.

⇒ End-users are actors. There's no place of stakeholders in use-cases. Use-case may consist of multiple scenarios.

Q.) Point of Sale (POS system). To supermarkets me scan karne ke liye.

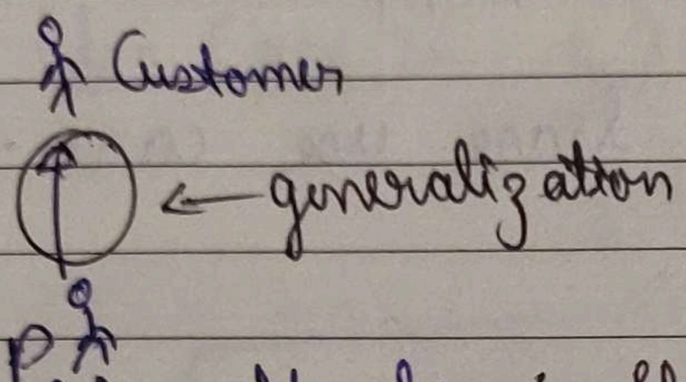


Note:- Customer not an actor since they don't interact with the system.

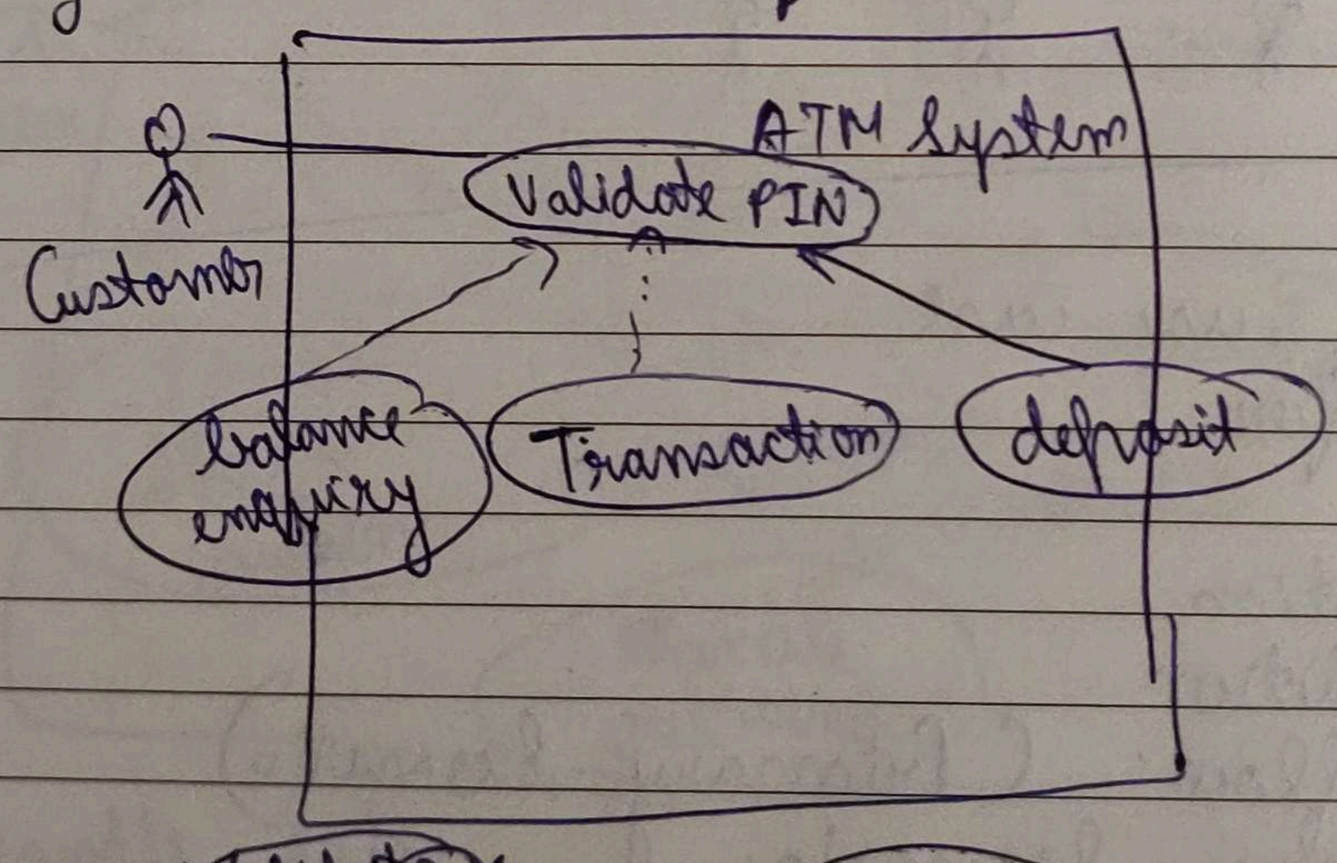
Often the written word can accomplish what the spoken word cannot.

(-18)

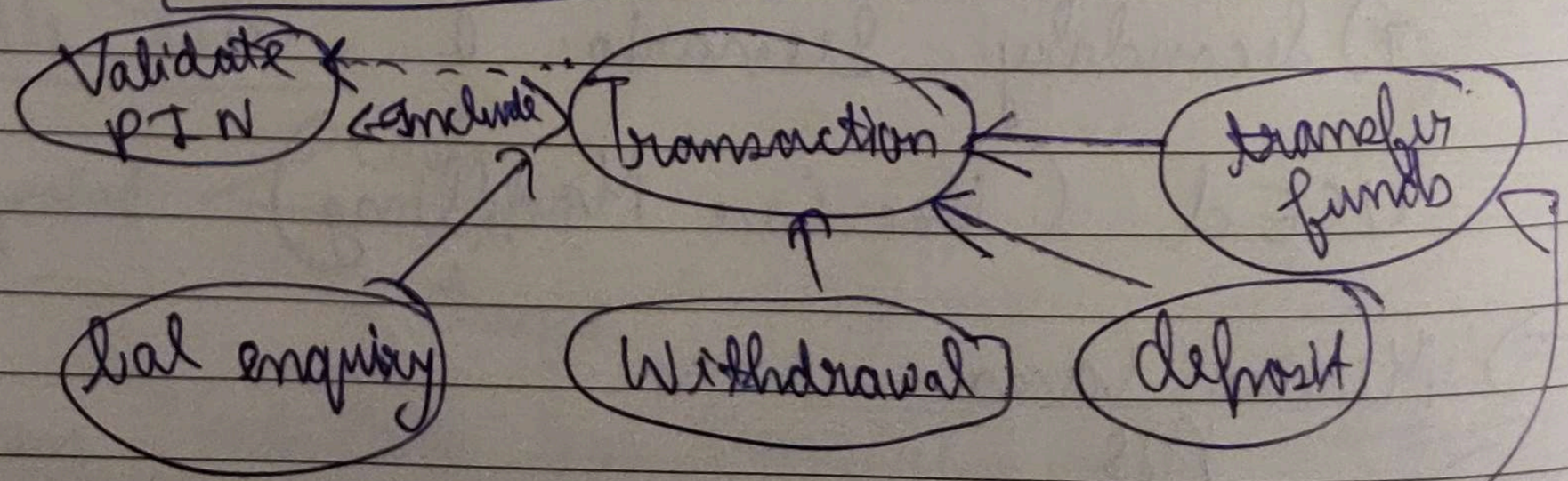
⇒ It's not necessary that only living things are part of system. Even external systems which support the functionality of the system, are actors. (i.e, interact with the system, living or non-living)



⇒ In ATM system, one common functionality for everything is "validate PIN".



For all these 3 features, validate PIN is common and required in every one. So, we identify and separate it out



⇒ Adding any feature is easy

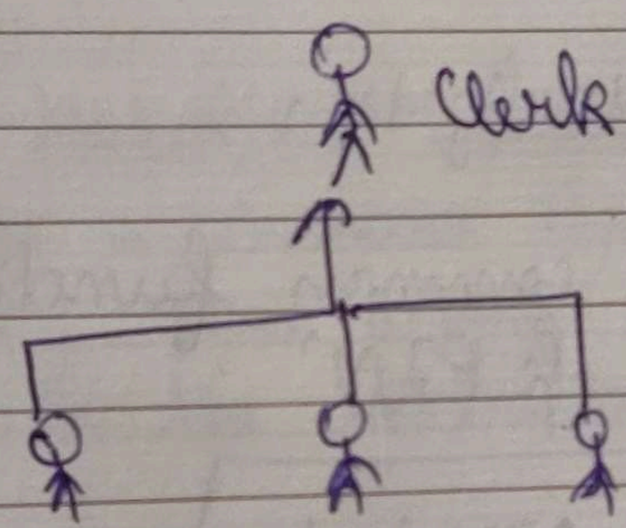
~~23/1/20~~ ~~Thursday~~

(19)

→ Extend is used to incorporate the use case which happens for certain unusual different event, which may/may not happen.

Eg:- Cancel transaction in ATM

→ Arrows always point to base use case.



1) Name of use case
2) Identifier
3) Goal
4) Precondition
5) Postcondition
6) Main flow. (Primary Scenario)
7) Secondary Scenario.

UCEd (Use Case Modelling)

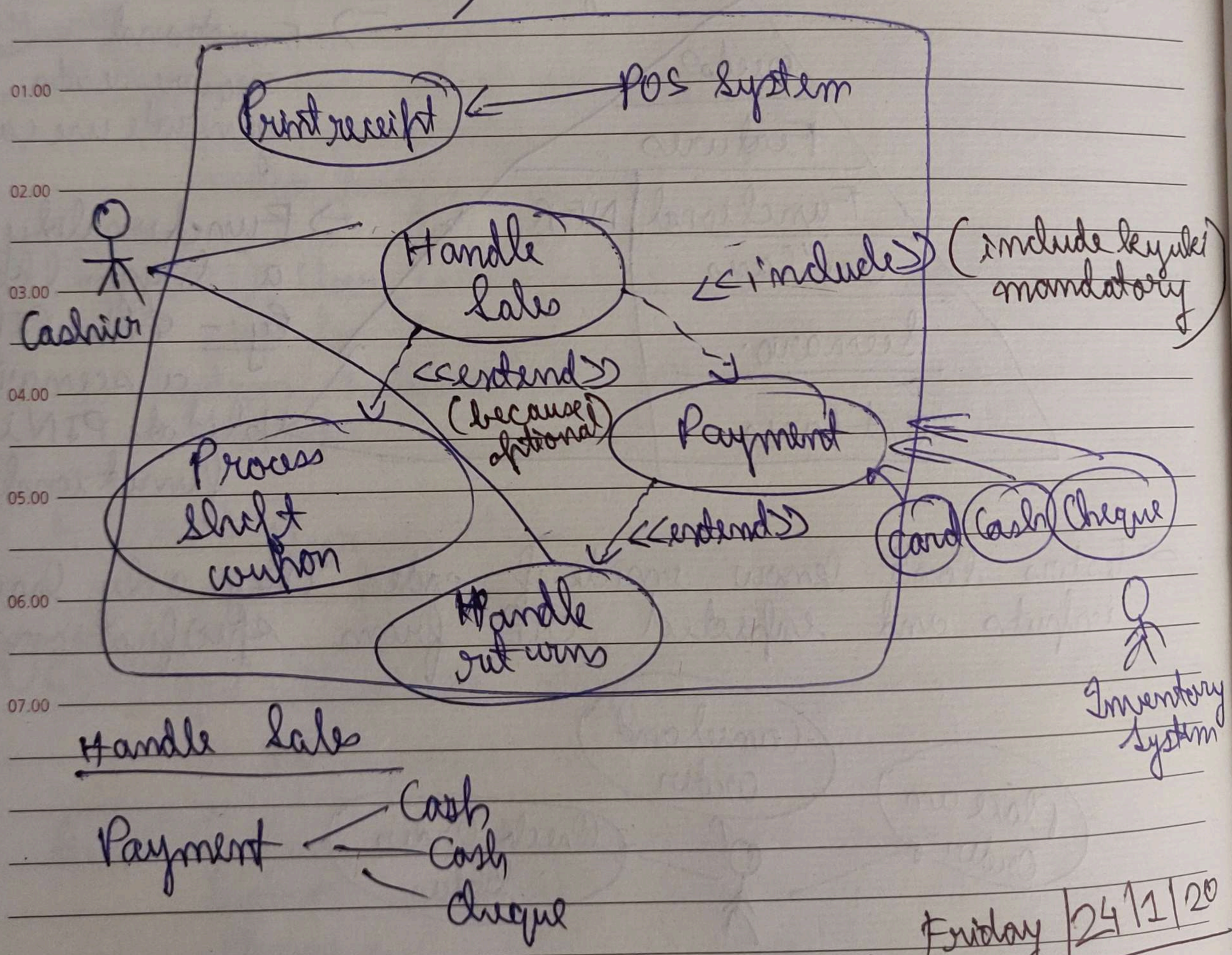
→ ~~Misuse Case~~
POS
⟹ Customer is not actor.

(20)

23/1/20

Actors :-
Cashier
Catalog System
Inventory System.

Use Case :- 1) Handle Sales.
2) Payment thru credit card / cheque
3)



Handle Sales

Payment — Card
— Cash
— Cheque

Friday 24/1/20

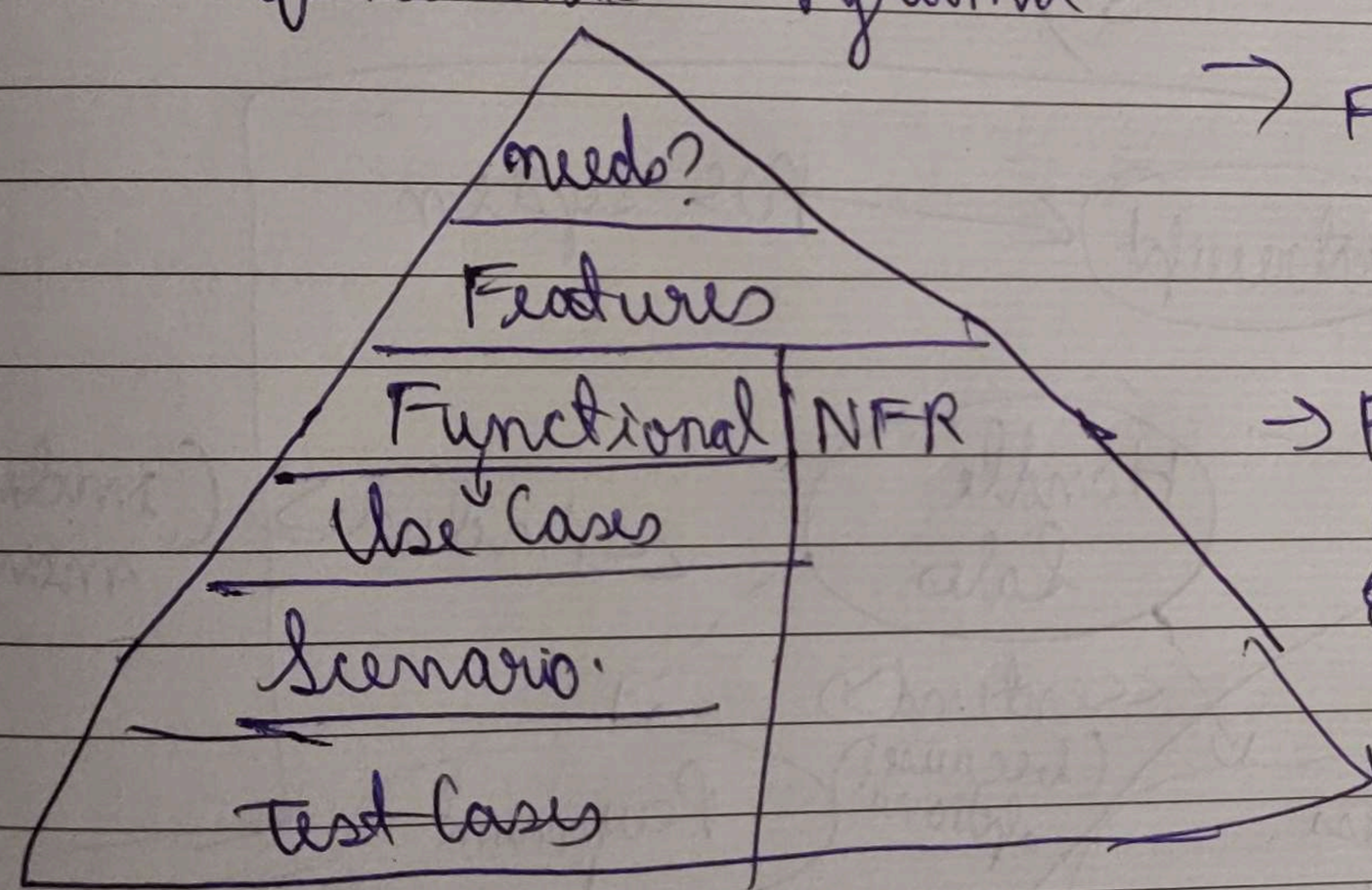
System Test Case Gen from Use Cases

Test cases are generated from requirements

To generate TC from requirements, they need to be validated.

1) Requirement review
2) Prototyping
3) TC generation.

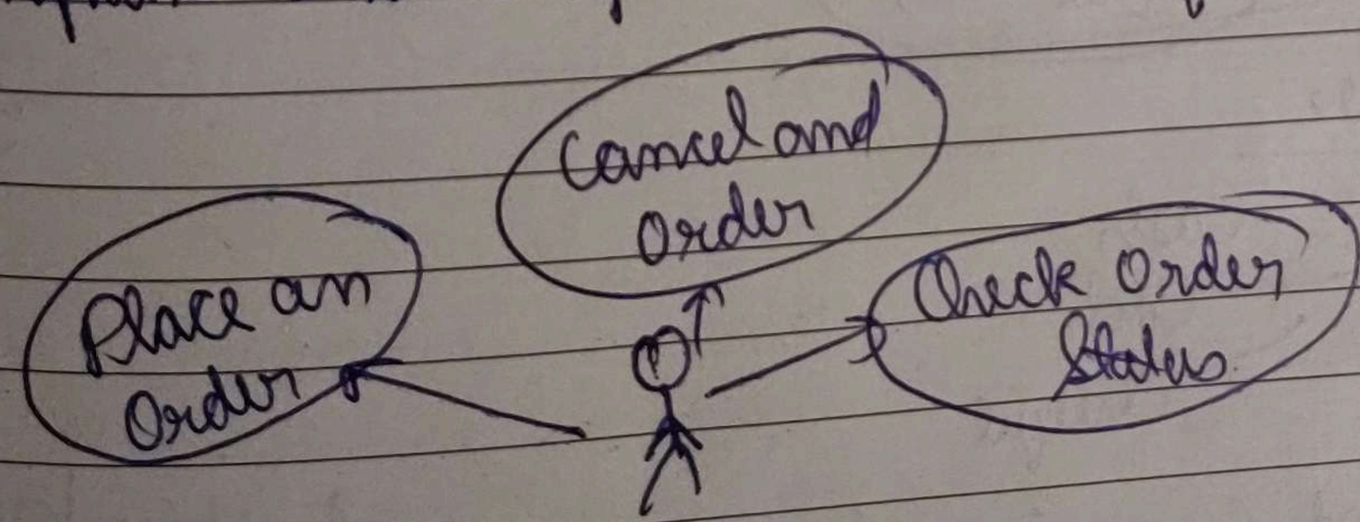
Requirements Pyramid



→ Functional requirements generate use cases

→ Functionality is a bigger block.
Eg:- Enter PIN is a scenario.
Validate PIN is a functionality

→ Testers don't know inside of code. They only have inputs and expected O/P from specifications.



P.T.O

(22)

24/1/20

Use Cases:

Place an Order

09.00 Actor : User.

10.00 Basic flows:-

11.00 1.] (B1) User enters website address in the browser.

12.00 System displays login page.

B1, B2 are the scenarios.

01.00 2.] (B2) User enters an email & a password.

System confirms correct login, presents

02.00 main page, prompts for a search string.

03.00 3.] (B3) User enters search string - partial name of a book.

04.00 System returns all books matching search criteria.

05.00 4.] (B4) User selects a book

System presents detailed info about a book.

08 Sunday 5.] (B5) User adds the book to an SC (Shopping Cart)

SC contents is presented to the user.

6.] (B6) User selects "proceed to check out" option.

System asks for confirmation of a shipping address.

7] (B7) User confirms shipping address. System presents shipping option.

8] (B8) User selects shipping option, System asks which Credit card (CC) will be used.

9] (B9) User confirms CC info. System asks for final confirmation to place an order.

10] (B10) User places an order. System returns a confirmation no.

Identifying TC

1) Identify variable needed for each use case step

2) Identify significantly different options for each variable.

PPTO.

Step	Variables	Options to be tested					
B1	Website	V Actual URL	I Blank	V min allow (1 char)	V max (59)	I Very long (256)	I is invalid
B2	email	Regular	I Blank	V min (6)	V max (10)	I one more allowed (11 char)	I very long (257)
	Password	Regular					
B3	Search String	Regular	Min all (1 char)	max (300)	one more allowed (302)		
B4	Selection	First selⁿ	Last selⁿ				
B5	Action Selection	Add to SC					
B6	" "	Proceed to Checkout					

F → First selection
R — Regular
E — Empty

05 December 2019 Thursday

(-24)

24/1/20
Friday

11	November 2019						
wk	M	T	W	T	F	S	S
44					1	2	3
45	4	5	6	7	8	9	10
46	11	12	13	14	15	16	17
47	18	19	20	21	22	23	24
48	25	26	27	28	29	30	

12	December 2019						
wk	M	T	W	T	F	S	S
48	30	31					1
49	2	3	4	5	6	7	8
50	9	10	11	12	13	14	15
51	16	17	18	19	20	21	22
52	23	24	25	26	27	28	29

Check 1 less than lower lim & 1 more than upper limit.

09.00

10.00

11.00

12.00

01.00

02.00

03.00

04.00

05.00

06.00

07.00

Shifting Order

add checkout

find search book

email

B7 Shifting address

B8 Shifting Method

5

confirm the address

These are called system-level test cases.

User Stories & Acceptance Criteria

Agile:- 1) Pupil over process
2) Test early, jaise hi req aaye
3) User Stories

User Story

Who (User Role) → What (goal) → Why (reason)

→ Acceptance criteria is a set of conditions that are used to confirm when a story is completed.

~~→ User Story has~~

→ User stories easily specify non-functional requirements.

Exam
1:30mins

Variable (Selection)	Value	Expected Value	Actual result	Pass / Fail Comments
Email	xyz@gmail.com	Log on screen	Manscript	
Email	www.abc.com	xy2-123		

03-12-2019 | 337-028

03

49
2019
December
Tuesday

-26

24/1/20
Friday

11				November 2019			
wk	M	T	W	T	F	S	S
44					1	2	3
45	4	5	6	7	8	9	10
46	11	12	13	14	15	16	17
47	18	19	20	21	22	23	24
48	25	26	27	28	29	30	

12				December	
wk	M	T	W	T	F
48	30	31			
49	2	3	4	5	6
50	9	10	11	12	13
51	16	17	18	19	20
52	23	24	25	26	27

09.00 1) 1 Question Concept Mapping
2) Elicitation
10.00 3) Use Cases
4) User Stories

} 6 questions

11.00

12.00

01.00

02.00

03.00

04.00

05.00

06.00

07.00