

## ISD (SDlre) Maths:

Spring 2023

Year	Proj 1 ( $r=8\%$ )	Proj 2 ( $r=10\%$ )	Proj 3 ( $r=11\%$ )	Proj 4 ( $r=12\%$ )
0	-90000	-1000000	-1000000	-1100000
1	10000	200000	300000	300000
2	10000	200000	300000	300000
3	10000	200000	300000	300000
4	20000	200000	300000	300000
5	1000000	3000000	300000	750000

Net Profit : (project wise राशि)

NP(1) : 60,000

NP(2): 100,000

NP(3): 50 000

NP (4): 85,000

Payback Period : (মাত্র বছরে investment দেওয়া)

~~PBP(1)~~ Proj L investment = 900 00 .

PBP(1) = 5 years.

Proj 2 investment = 1000 000

$$PBP(2) = 5 \text{ years}$$

Proj 3 investur = 100 000,-

$$PBP(3) = 4 \text{ years}$$

Proj 4 investment = 1100 00

$$PBP(1) = 1 \text{ years.}$$

Sub:

Day \_\_\_\_\_  
Time: \_\_\_\_\_ Date: / /

ROI: (Formula:  $\frac{NP/\text{total years}}{\text{total investment}} \times 100$ )

$$\text{ROI}(1) : \frac{(60,000/5)}{900 \text{ } 000} \times 100 = 13.33\%$$

$$\text{ROI}(2) : \frac{(100,000/5)}{1000 \text{ } 000} \times 100 = 1.67\% \text{ or } 2\%$$

$$\text{ROI}(3) : \frac{(500,000/5)}{1000000} \times 100 = 8.33\% \text{ or } 10\%$$

$$\text{ROI}(4) : \frac{(85,000/5)}{1100000} \times 100 = +2.87\% \text{ or } 15.45\%$$

NPV: [Annual Discount factor,  $Df = \frac{1}{(1+r)^{\text{year}}}$ ]

$$\text{NPV}(1) : r = 0.08$$

Year	Money (TK)	$Df$	Cash flow = $Df \times Tk.$
0	-900 000	$\frac{1}{(1+0.08)^0} = 1$	-900 000
1	100 000	0.9259	9259
2	100 000	0.8573	8573
3	100 000	0.7938	7938
4	200 000	0.7350	14700
5	1000 000	0.6806	68060
		$\Sigma =$	18530

$$\text{NPV}(2) : r = 0.1$$

Year	Money (TK)	$Df$	$Df \times Tk$
0	-1000 000	1	-1000 000
1	200 000	0.909	181 800
2	200 000	0.826	165 200
3	200 000	0.751	150 200
4	200 000	0.683	136 600
5	300 000	0.620	186 000
			$-754400$

calculate it again ..bhul korsi :(

NPV(3) :  $r = 0.11$

Method of approach (3)

(3)

Year	Money	Df	Discounted Cash flow = Df x TK
0	-100 000	1	-100 000
1	30000	0.909	27027
2	300 00	0.8116	24348
3	300 00	0.731	21930
4	300 00	0.658	19790
5	300 00	0.593	17790
			$\sum = 10835$

NPV(4) :  $r = 0.12$

Year	Money	Df	Discounted Cash flow = Df x TK
0	-110000	1	-1100 00
1	300 00	0.8928	26784
2	300 00	0.7971	23913
3	300 00	0.7117	21351
4	300 00	0.6355	19065
5	75 000	0.5679	42557
			$\sum = 23670$

(2) b

$$10835 = 3$$

NP-wise rank :  $2 > 4 > 1 > 3$  [higher rank = better]

PBP " " :  $3 > 4 > 1 > 2$  [earlier years = better]

ROI " " :  $4 > 1 > 3 > 2$  [higher % = better]

NPV " " :  $3 > 1 > 4 > 2$  [lower = better]

4 is better as in ~~most~~ case it has a better rank.

Q) As ~~the~~ all the investments are near to lakh and is in lakhs, we it is better to use Pay Back Period analysis.

⑤ a) Functional decomposition:

1. User Interface design
2. Content management
3. Feed management
4. Messaging system.
5. Notification management
6. Search, and Discovery and ~~interaction connection~~
7. Privacy & Security
8. Admin Panel.

b)

Functions	S <sub>opt</sub>	S <sub>Likely</sub>	S <sub>press</sub>	S <sub>ft.</sub>
1. User	8000	7000	6000	7000
2. Content	7200	8000	7000	7700
3. Feed	9000	9010	9020	9010
4. Mess	7400	7510	7000	7407
5. Notification	83000	3610	3200	3040
6. Search	2100	2000	2010	2018.3
7. Privacy & Secu	9500	9510	9020	9426.7
8. Admin	9700	9710	9660	9700

$$\sum = 55302$$

We know,  $S = S_{opt} + (f \times S_{likely}) + S_{press}$

$$6.$$

$$\text{Estimated LOC} = 55,302$$

c) Given, Labour rate,  $R = \$450 \text{ LOC}/\text{pm}$ ,  $\$7000 \cdot \text{pm}$ .

Avg. prod, AP = 450 LOC/pm.

$$Effort = \frac{\sum \text{LOC}}{AP} = \frac{55302}{450} = 122.9$$

$$\text{Cost} = \sum \text{LOC} \times R = 122.9 \times 7000 = \$860,253.33$$

8P-2021

⑥ FP based:

Info Demand	Sopp	Slikely	Specs	Wregh	Count	FP count
inputs	19.5	22.5	28.5	5	23	92
outputs	11.5	15	22	5	16	78
inquiries	16	21.5	28	5	22	108
logical files	9	9	5.5	10	5	43
interface files	2	2.5	3	7	3	18
					$\sum UFP =$	340

$$\Sigma \text{ value} = 47 \quad (\because \text{from 2nd table})$$

$$\text{Adjusted FP} = (0.65 + (0.013 \times 47)) \\ = 31.02 \quad 1.12$$

$$i) \text{ FP estimated} = UFP \times AFP = 340 \times 31.02 = 10546.8 \quad 380.8$$

$$ii) \text{ Avg prod} = 8.5 \text{ FP pm}, \text{ unit cost} \text{ not mentioned} \\ \text{Labour Rate} = \$9000 \text{ pm.}$$

$$\text{Proj cost Proj cost} = \frac{9000}{8.5} \times 380.8 \\ = 1,167,200 = \$1,03,200$$

$$iii) \text{ Effort} = \frac{380.8}{8.5} = 44.8$$

$\approx 45.$

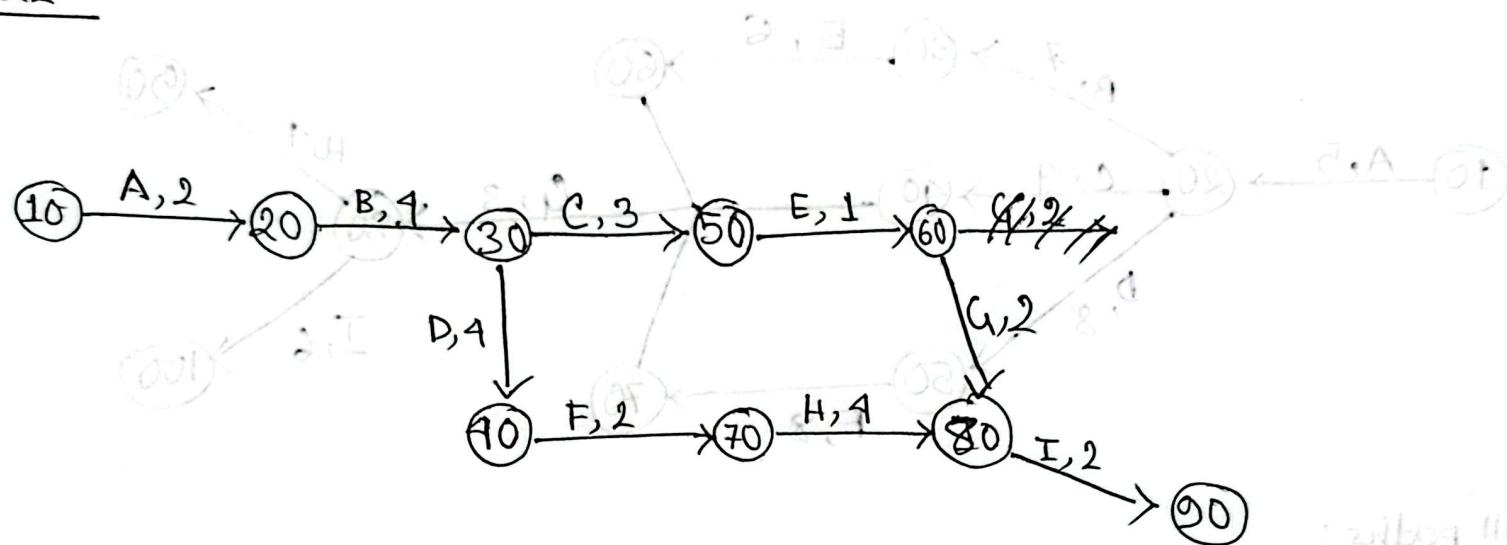
(45 months - 1 month coding code করবে ৩৩  
15 month coding ১৫ month - ২ month coding করবে)

Part

1808-1163

Sp-22

a)



b) all path:



$$81 = 2 + 8 + 3 + 1 + 2 + 2 = 19$$

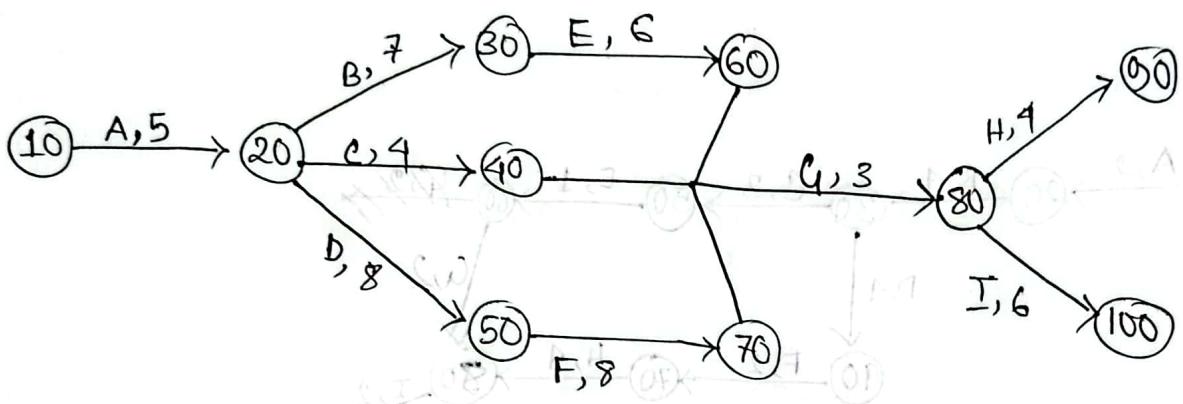
$$\text{II } 2 = 2 + 9 + 4 + 2 + 4 + 2 = 18$$

So, path 1 is less time consuming.

init each select along long

Fall-2021

(5)



All paths:

1.  $10 \rightarrow 20 \rightarrow 30 \rightarrow 60 \rightarrow 80 \rightarrow 90 = 5+7+6+3+4 = 25$
2.  $10 \rightarrow 20 \rightarrow 30 \rightarrow 60 \rightarrow 80 \rightarrow 100 = 5+7+6+3+6 = 27$
3.  $10 \rightarrow 20 \rightarrow 40 \rightarrow 80 \rightarrow 90 = 5+4+3+4 = 16$
4.  $10 \rightarrow 20 \rightarrow 40 \rightarrow 80 \rightarrow 100 = 5+4+3+6 = 18$
5.  $10 \rightarrow 20 \rightarrow 50 \rightarrow 70 \rightarrow 80 \rightarrow 90 = 5+8+8+3+4 = 28$
6.  $10 \rightarrow 20 \rightarrow 50 \rightarrow 70 \rightarrow 80 \rightarrow 100 = 5+8+8+3+6 = 30$

3rd path takes less time.

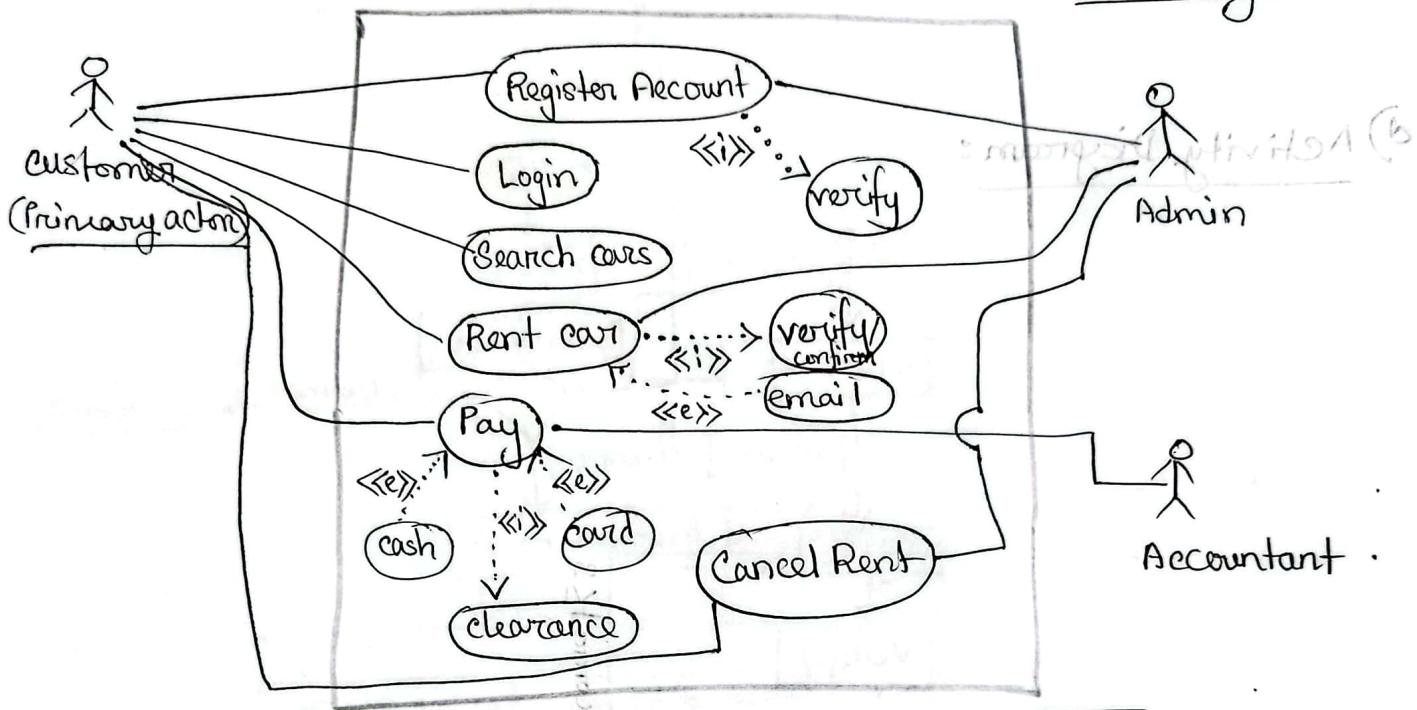


## Diagrams:

Spring 2023

④ a) Use case Diagram:

Secondary actors



b) **User**

1. Sign up()

**:Site**

2. Verify()

**:DB**

3. Verified()

**ALT [success]**  
4. sendsConfirmation\_email()  
5. login()

6. Check()

**ALT [success]**

7. Checked()

8. Accessed\_account()

9. Return\_done()

10. Return\_done()

**[unsuccess]**

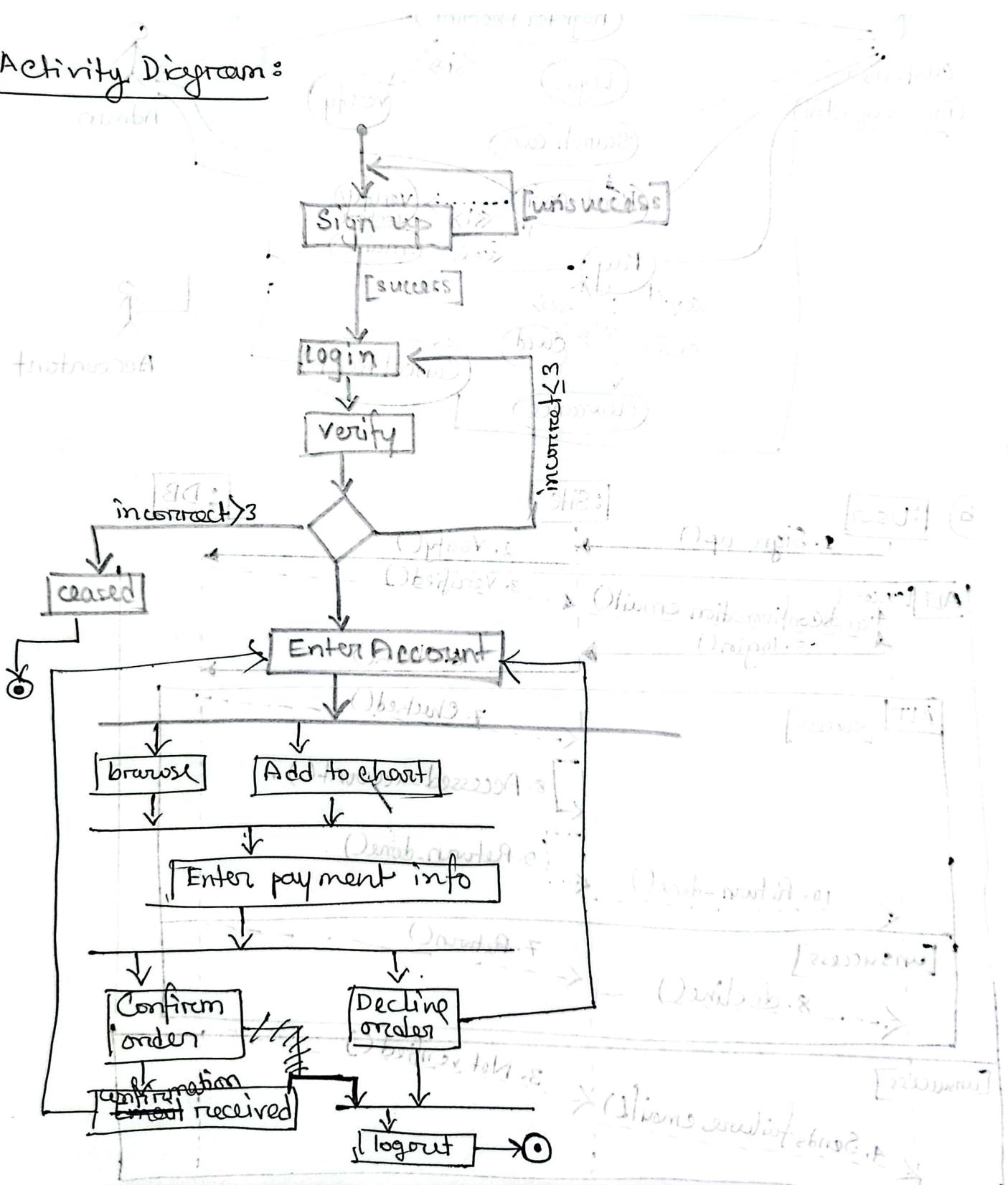
8. decline()

7. Return()

3. Not\_verified()

4. Sends\_failure\_email()

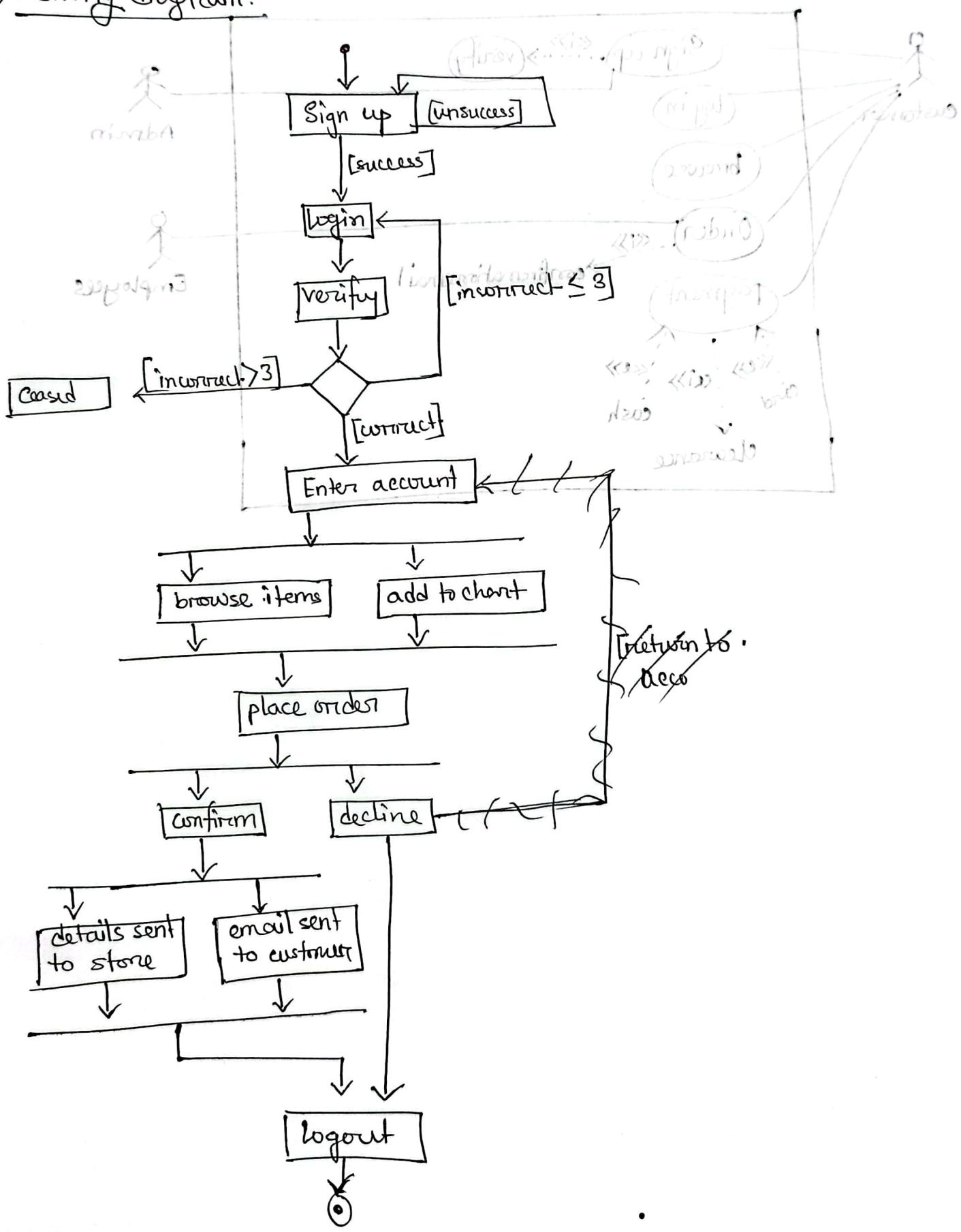
## d) Activity Diagram:

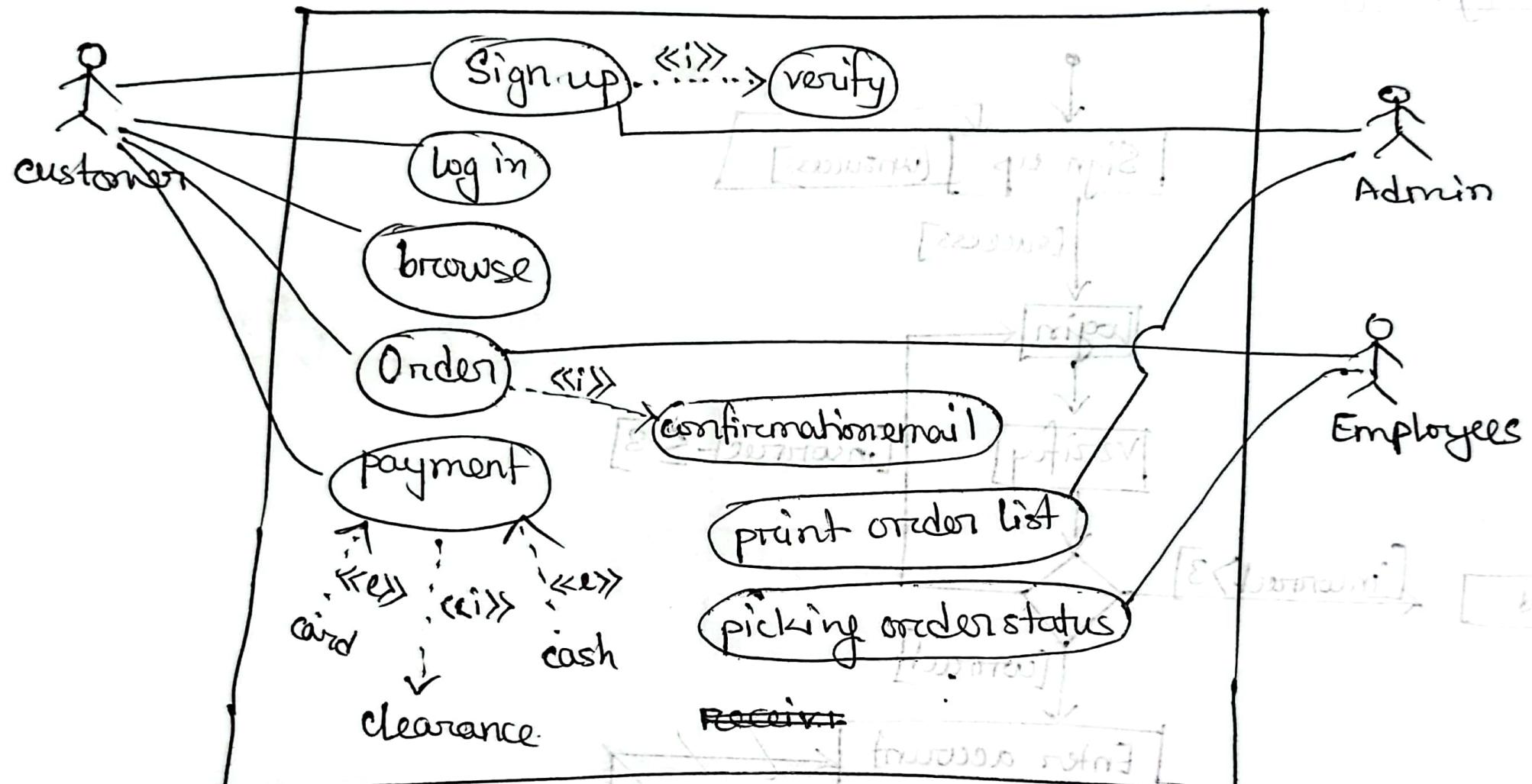


Fall-2022

2200 28J (8)

a) Activity diagram:

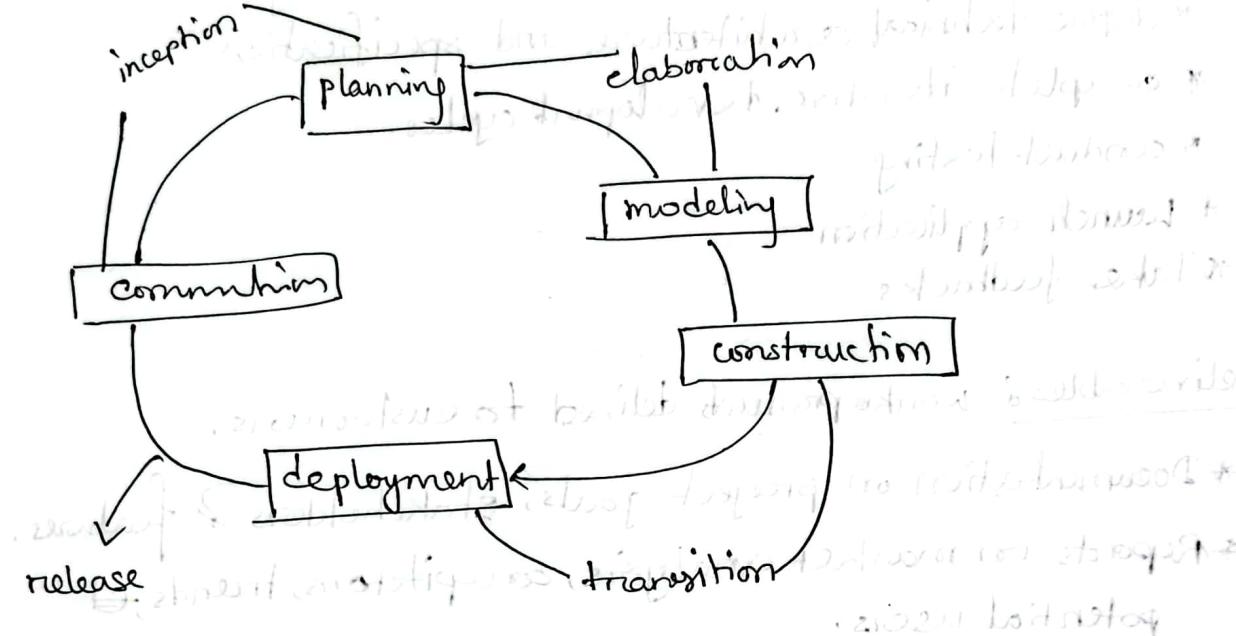


**b) Use case:**

Theory :

Sprin-23

- ③ a) We can use agile combining with waterfall known as the unified process Model.



Here building of a software happens in a circular process enabling more participation of customer for negotiating and interacting with the developers. They get more opportunities to response to sudden changes amidst project due to continuous collaboration with customers. Besides in unified process model waterfall is used, too.

(3) b)

Milestones: points in schedules to access progress

- \* define project scope and objectives
- \* completes market analysis
- \* creates wireframes and interactive prototypes
- \* define technical architecture and specifications
- \* complete iterative development cycles.
- \* conduct testing.
- \* Launch application
- \* Take feedbacks

Deliverables: works products delivered to customers,

- \* Documentation on project goals, stakeholders & features.
- \* Reports on market analysis, competitors, trends, potential users.
- \* Showcase of app layout and prototype UX design.
- \* Documentation of technical architecture & API specifications.
- \* Versions and code review reports.
- \* User and Bug Testing reports.
- \* Final product build with market launch reports.
- \* Post launch reports and feedbacks.
- \* Future improvements roadmap.

c) Both developers and users need to do validating testing.

Developers do the internal testings to ensure software meeting specific requirements & functions before user can test.

They perform unit testing, integration testing, system testing.

After coding developers may test to verify the features. End users do their User Acceptance testing to validate the software & and give feedbacks on using it in an environment of real-world usage. They do the testing after developers have completed development and internal testing.

Developers focus on technical aspects and conduct tests in early development stage to fix issues quickly. Users test after the project is done developed to check real-world requirements and expecting to give feedbacks on overall satisfaction.

### ③ e) Black Box

vs

### White Box

① Focuses on external behaviours	① on internal behaviour.
② Evaluates functionality & user experience	② Tests internal design and codes
③ Testers don't need coding knowledge.	③ Developers or testers with coding knowledge
④ Easy to automate	④ Goes through code
⑤ Quicker	⑤ Takes time, uncovers hidden errors.
⑥ Limited coverage of internal parts	⑥ Uncovers hidden errors.
⑦ Doesn't need coding knowledge.	⑦ Requires coding knowledge
⑧ Functional, User acceptance, regression testing.	⑧ Unit, integration, security testing

- ③ f) Reasons to use umbrella activities in large projects.
- ① Software project management: Monitoring progress, understanding what actions to take and identifying deviations.
  - ② Formal Technical reviews: conducting regular reviews of software deliverables to ensure requirements and defects.
  - ③ SW Quality Assurance: ensures SW quality through testing, documenting & bug verification and validation.
  - ④ SW Configuration management: controls changes to products and components in SW. Maintains consistency and traceability.
  - ⑤ Works product preparation and production: Creating and managing documents, models, logs, forums and lists necessary for the SW.
  - ⑥ Reusability management: Identifying and implementing opportunities for reusable code and components to save time and money.
  - ⑦ Measurement: collecting and analyzing data to evaluate project performance, identify areas for improvement etc.
  - ⑧ Risk Management: Figuring out potential risks that might impact the project and building mitigation strategies.

Spring 2022

⑤ a) Waterfall - [will give examples of projects that have predictable nature, well-defined requirements, minimal expectations etc → Banking system, medical software, embedded software]

Agile: flexible, collaborative, large, customers feedback → E-commerce platform

- \* XP: - rapid delivery, adaptability, complex technical requirements
- \* Scrum: regular reviews type app → Mobile app. for fitness track

\* Spiral: has risk assessment, complex project → Aerospace system.

\* V model: ~~needs waterfall~~ <sup>Focuses QA</sup> needs validation + verification  
example: Safety Critical Systems (Automotive Control System).

\* Incremental: small, functional increments, allowing portions of the system to be delivered.  
example: Customer Relationship Management System.

⑦ c) Smoke testing: Verifies integrated system components function together correctly.

- \* targets small set of critical features or functionalities.

- \* Is quick and efficient.

- \* ~~verifies if the system functions properly~~ → ~~detects errors~~

- \* Detects major errors.

Fall-2022

③ a) ~~X~~ Refactoring :- improves code → without changing its external behavior.

- simplifies codes, designs, tests.
- eliminates redundancy, unnecessary fn
- increases code coherency
- one time activity but continuous process.

b) Pair programming :- collaborative practice, two dev works in a single workstation.

- one is junior who is good with syntaxes
- another is senior who has good problem solving skills.
- saves time.
- increases focus, quality etc.
- reduces defects.

## 2) Frameworks activities:

① Communication - gather requirements from stakeholders + user  
- define goals and obj.  
- establish common understanding betw developers and stakeholders

② Planning: - describe tasks  
- establish work plan + schedules  
- list resources  
- outline tasks

③ Modeling:  
- create designs + models  
- identify key components.  
- analyze requirements.

## ④ Construction:

- implement designs + models
- code generation
- testing.
- ensure quality

## ⑤ Deployment:

- evaluate feedbacks.
- provide future improvement reports.
- ensure if product met needs + expectation

### c) Validation vs Verification

(v8)

### Verification vs Validation

- |   |  |
|---|--|
| ① ensures requirements, designs, codes, architecture  | ① Ensures end-users expectations                         |
| ② conducted during dev process, at various stages.    | ② Conducted after dev is done during testing and QA.     |
| ③ Inspection, code reviews, walkthroughs, desk checks | ③ Usability, performance, system and acceptance testing. |
| ④ confirms static + dynamic testing activities        | ④ primarily dynamic, focuses on real-world scenarios.    |
| ⑤ Done by Quality assurance team                      | ⑤ Done by software team and end users.                   |
| ⑥ "is the product"                                    | ⑥ "Are we building the right product?"                   |

g)

LOC

(vs)

FP (Function Point)

- |   |
|---|
| ① Counts Lines of Codes                                   |
| ② Ignores complexity                                      |
| ③ Language-dependent                                      |
| ④ Suitable to estimate effort and cost post-project phase |
| ⑤ Used to maintain a legacy system.                       |

- |  |
|--|
| ① Calculates size of software, number of functions it performs |
| ② Considers data manipulation, user interactions, algorithms.  |
| ③ Lang-independent   |
| ④ pre-project phase  |
| ⑤ Used to develop a new feature-rich application.              |

## Open Questions:

- ① How would you approach identifying business requirements for updating MIS sys. (business needs considering infrastructure and goals)
- ② What techniques/tools you prefer — →
- ③ How would you design/implement scripts for seamless data transfer.
- ④ How to be ensured that technical documentation is clear + concise.
- ⑤ In a situation where multiple stakeholders have different opinion — how to facilitate from a ~~dear~~ common decision

## Closed Question :

- ① What specific technical requirements/protocols needs to be integrated for data exchange?
- ② Can you provide ~~key~~ list of business processes to update MIS sys to present data.
- ③ What are the 2 tools you prefer for collecting/changing data ~~source~~ to have in this system?
- ④ Do you prefer any specific security or access control?
- ⑤ What are the key performance indicators for MIS system updates that should be tracked?

## क्या काम करना चाहिए?

- ① Funnel: ▽ when interviewing our expert/stakeholders — opinion निय + then specific कार्यक्रम का लिए आवश्यक
- ② Pyramid: △ interviewing developers, junior in rank who'll be in team, coders — कार्यक्रम का लिए + opinion चाहिए
- ③ Diamond: ◇ talking to end users — Overall satisfaction चाहिए + general feedback निय + then specific improvement निय करना।