Date ___/___/_____

SE

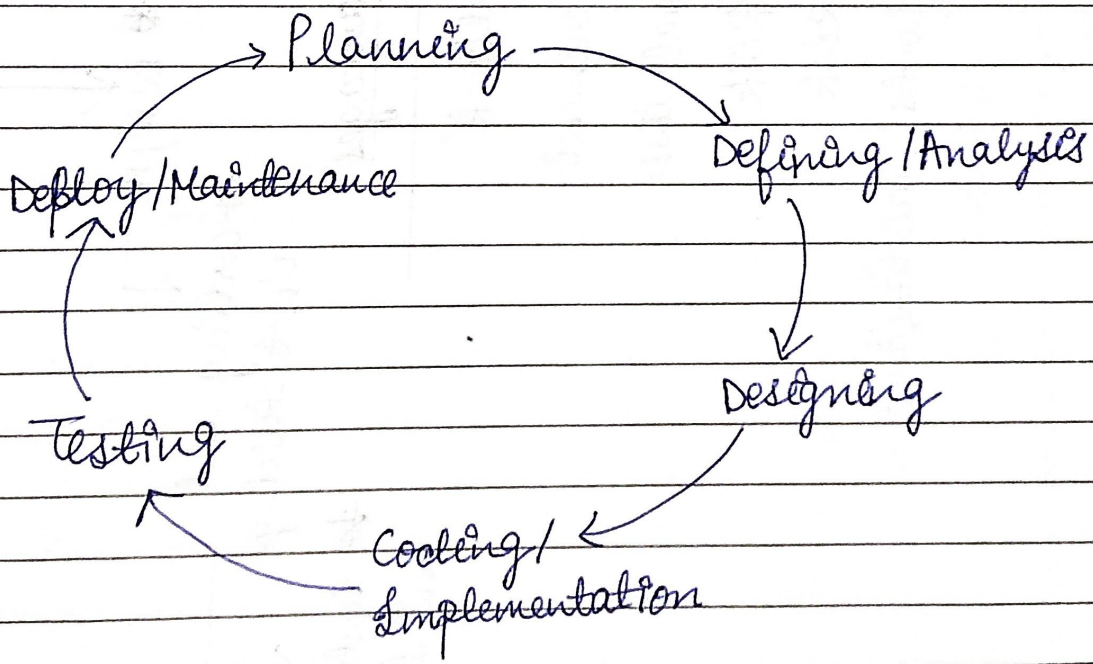
→ It is systematic, disciplined, cost-effective techniques for software development.

→ Engineering approach to develop a software.

→ Evolution

1945-65 → Origin
1965-85 → Crisis
1990-2000 → Internet
2000-2010 → Light weight
2010 → Till → AI, ML, DL

* SDLC (Software Development Life Cycle):-



Saathi

Date __/__/____

Page No.

→ **Classical Waterfall Model:-** (1970s)

* Feasibility study
* Requirement Analysis and specifications
* Design
* Coding and Unit testing
* System testing and Integration
* Maintenance

Advantages	Disadvantages
→ Base Model	→ No feedback
→ Simple & Easy	→ No experiment
→ Small Projects	→ No parallelism
	→ High Risk
	→ High Risk
	→ 60% efforts maintenance

Saathi

Page No.

Date ___/___/___

* Iterative Waterfall Model:-

* Advantages:-
  - → Base Model
  - → simple and easy
  - → Small Projects
  - → Feedbacks

Desadvantages:-

- → No phase overlapping
- → less customers Interaction
- → No Intermediate delivery
- → Rigid (No changes)

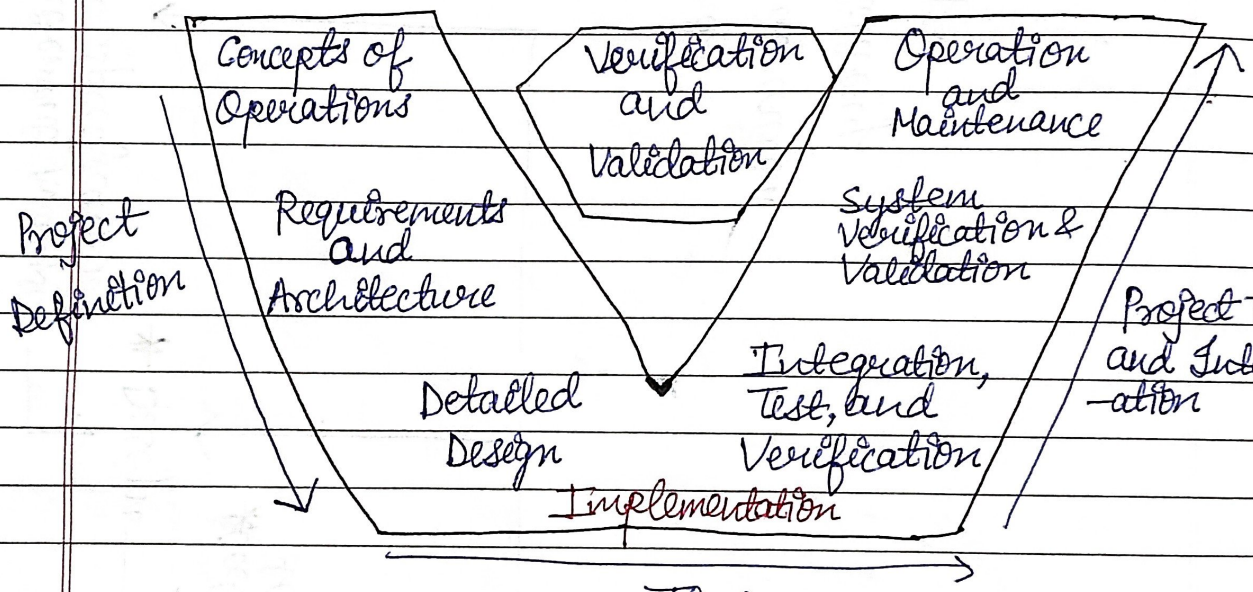
Saathi

Saathi

Date ___ / ___ / _____

* V- Shaped Model :-

→ Also known as Verification & Validation Model.
→ Extention of Waterfall model.
→ Testing is associated with every phase of lifecycle.
→ Verification Phase (Requirement analysis, System design, Architecture design, Module design).
→ Validation Phase (Unit testing, Integration, System, Acceptance Testing)



Advantages

→ Time saving.
→ Good understanding of project in the beginning.
→ Every component must be testable.
→ Progress can be tracked easily.
→ Proactive defect tracking.

Disadvantages

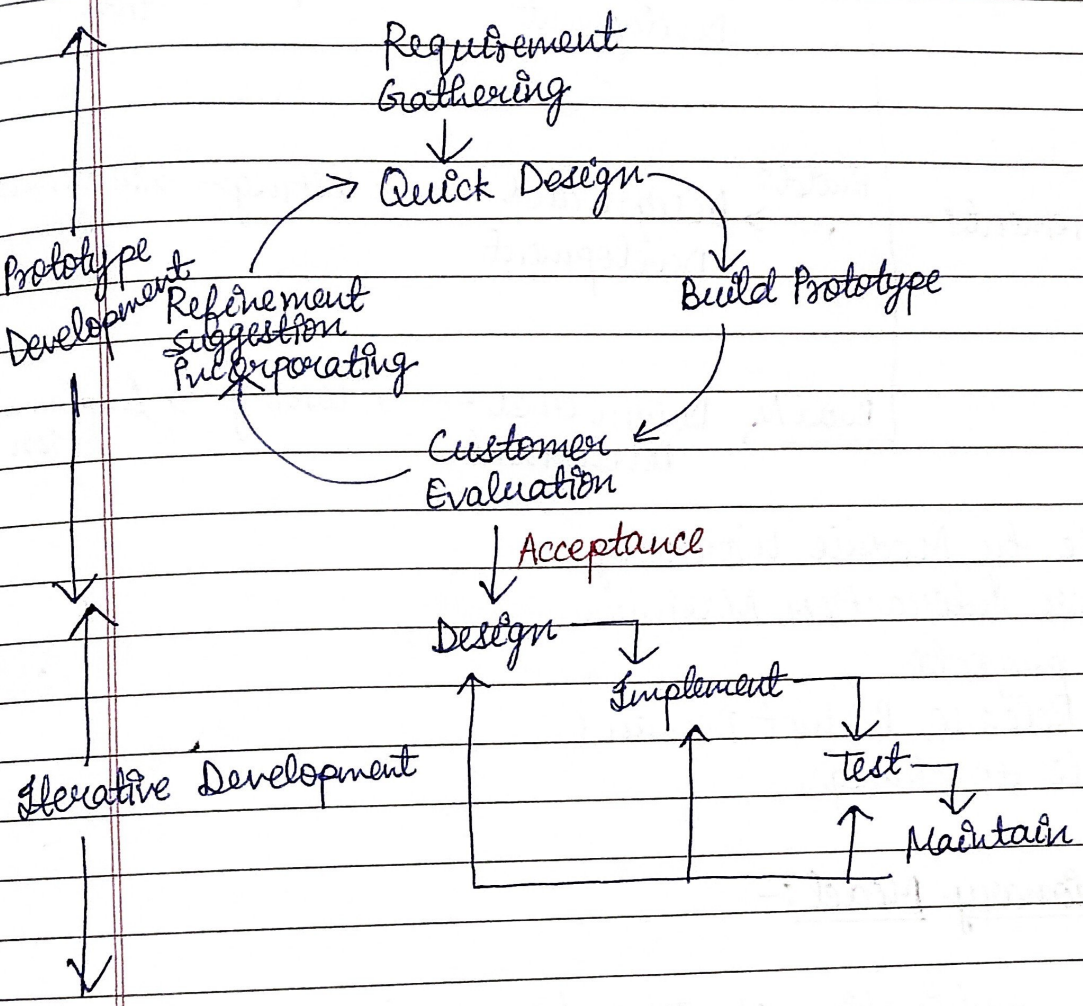
→ No feedback so less scope of changes.
→ Risk analysis not done.
→ Not good for big or object-oriented projects.

Page No. ____

Date ___/___/Day

Saathi

* Prototyping Model :-



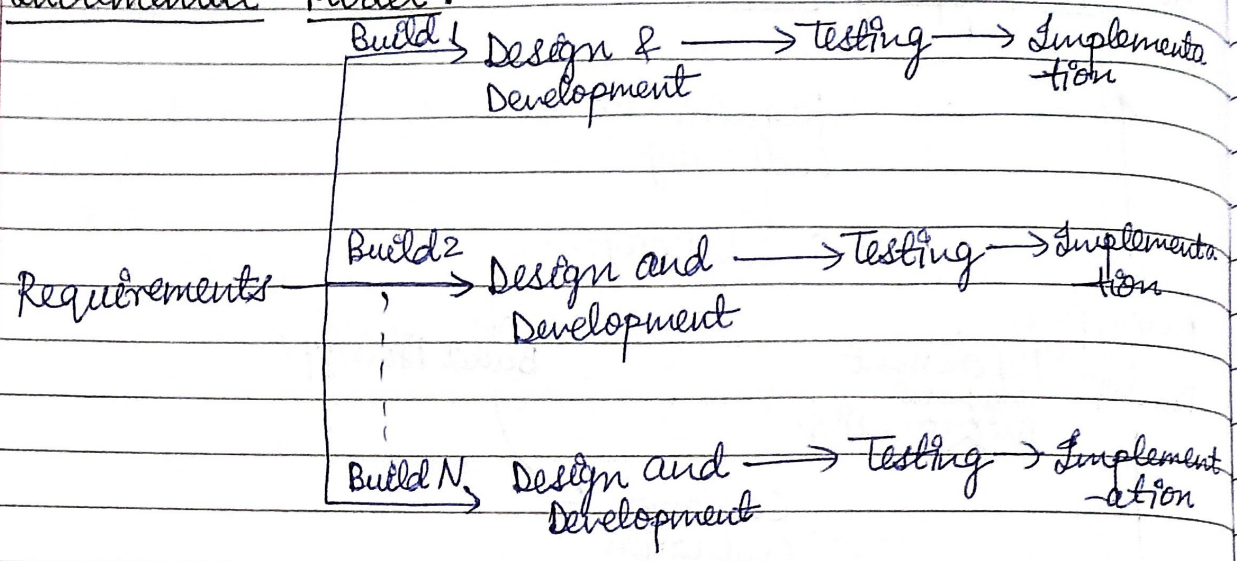
→ Customer not clear with idea.
→ Throwaway Model.
→ Good for technical and requirement risks.
→ Increase in cost of Development.

Page No.

Saathi

Date ___ / ___ / ___

* Incremental Model :-



→ Module by Module Working
→ Customer Interaction Maximum
→ Large projects
→ Early Release Product Demand
→ Flexible to changes

* Evolutionary Model :-

→ It is a combination of Iterative and Incremental model of software development life cycle.
→ Incremental model first implement a few basic features and deliver to the customer. Then build the next part and deliver it again and repeat this step until the desired is fully realized. No long-term plans are made.
→ Iterative model main advantage is its feedback process in every phase.
→ Also known as "Design a little, build a little, test a little, deploy a little model."

* Advantages :-

Page No.

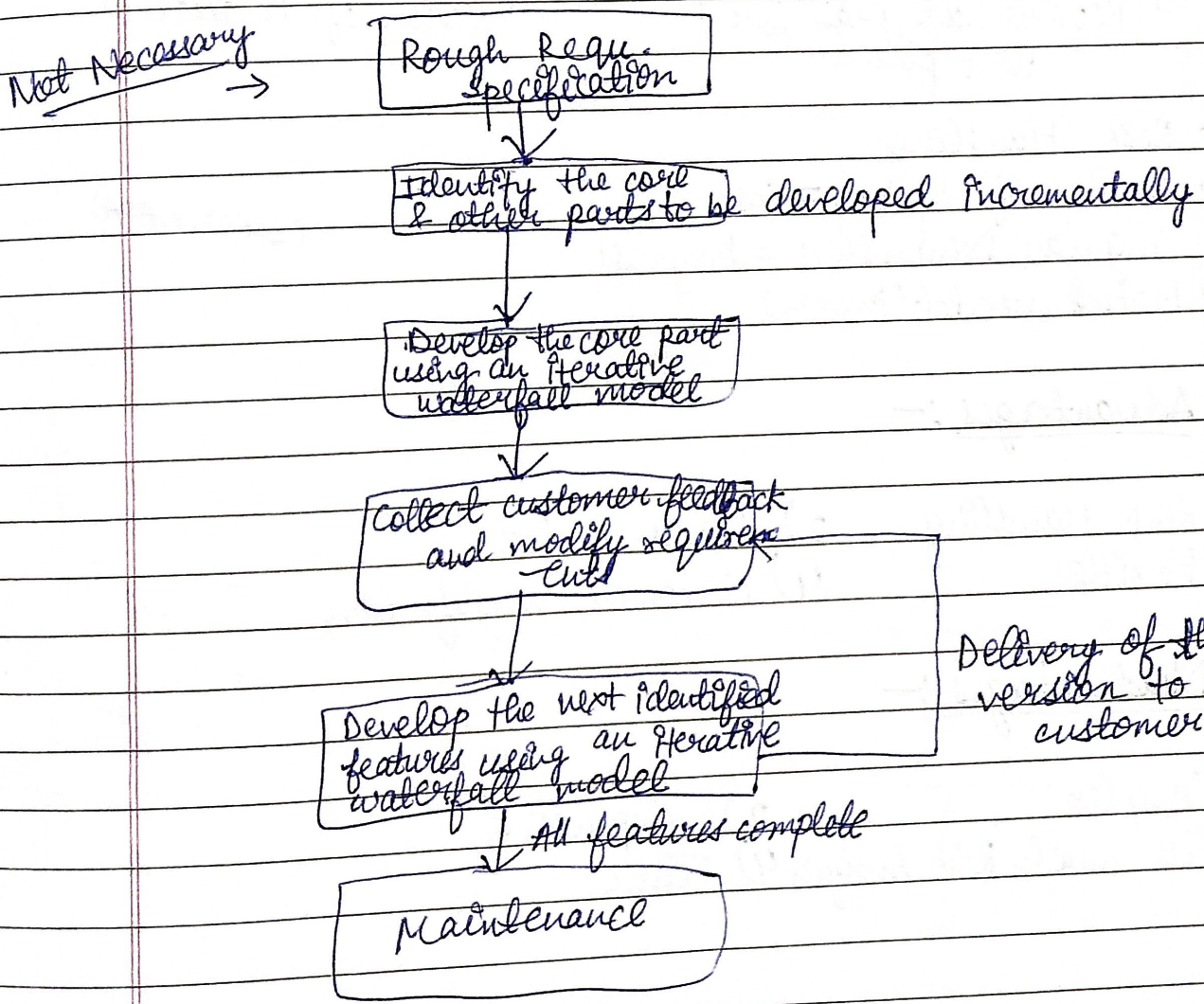
Date ___/___/_____

Saathi

→ Customer requirements are clearly specified.
→ Risk analysis is better.
→ It supports changing environment.
→ Initial operating time is less.
→ Better suited for large mission-critical projects.

* Disadvantages:-

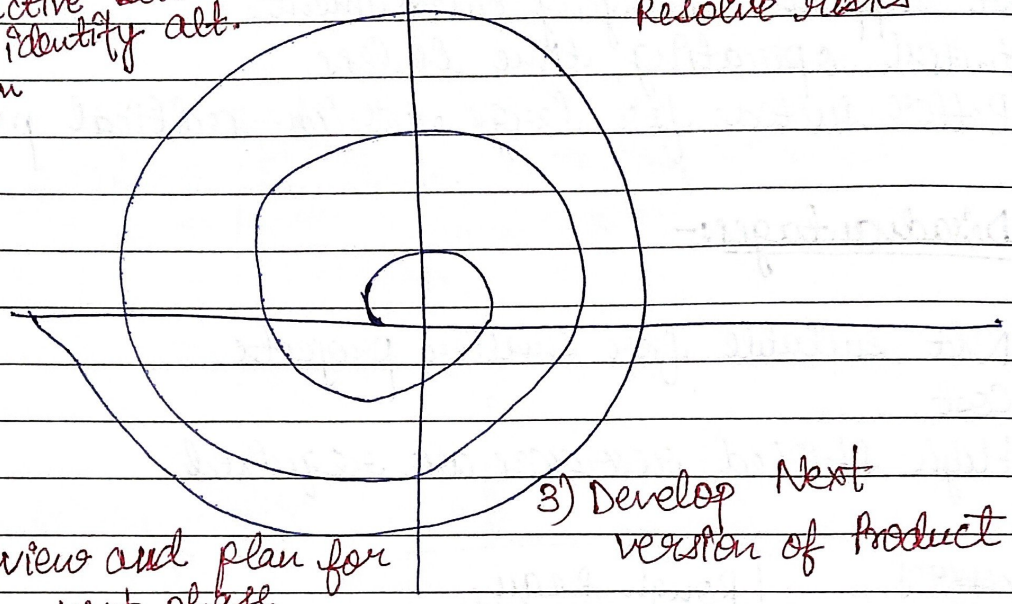
→ Not suitable for smaller projects.
→ Cost.
→ High skilled resources are required.



Saathi

Date ___ / ___ / _____

* Spiral Model :-



→ Risk Handling
→ Radius of Spiral = Cost
→ Angular Dimension = Progress
→ Meta model (known)

(ISRO, NASA)

* Advantages :-

1) Risk Handling
2) Large Projects
3) Flexible
4) Customer Satisfaction

* Disadvantages :-

1) Complex
2) Expensive
3) Too much Risk Analysis
4) Time

Saathi

* Compare every SDLC :-

Date / /

Classical Waterfall	Iterative Waterfall	Prototype Model	Incremental Model	Evolutionary Model	RAD Model	Spiral Model	Agile Model
Basic, Rigid, Inflexible, Not for Real Projects	Basic, Problm ~~is~~ will understood.	User Requirement Not clr, costly, No early lock on requirements → High User Involvement → Reusability	Model by Module Delivery Easy to test and debug	Large Projects	Time and cost constr-aints User at all levels → Reusability	Risk, Not for small Projects → No early lock on requirements → less expen-sive Can work	Flexible, Advanced, Parallel, Process divided into sprints

Page No.

Saathi

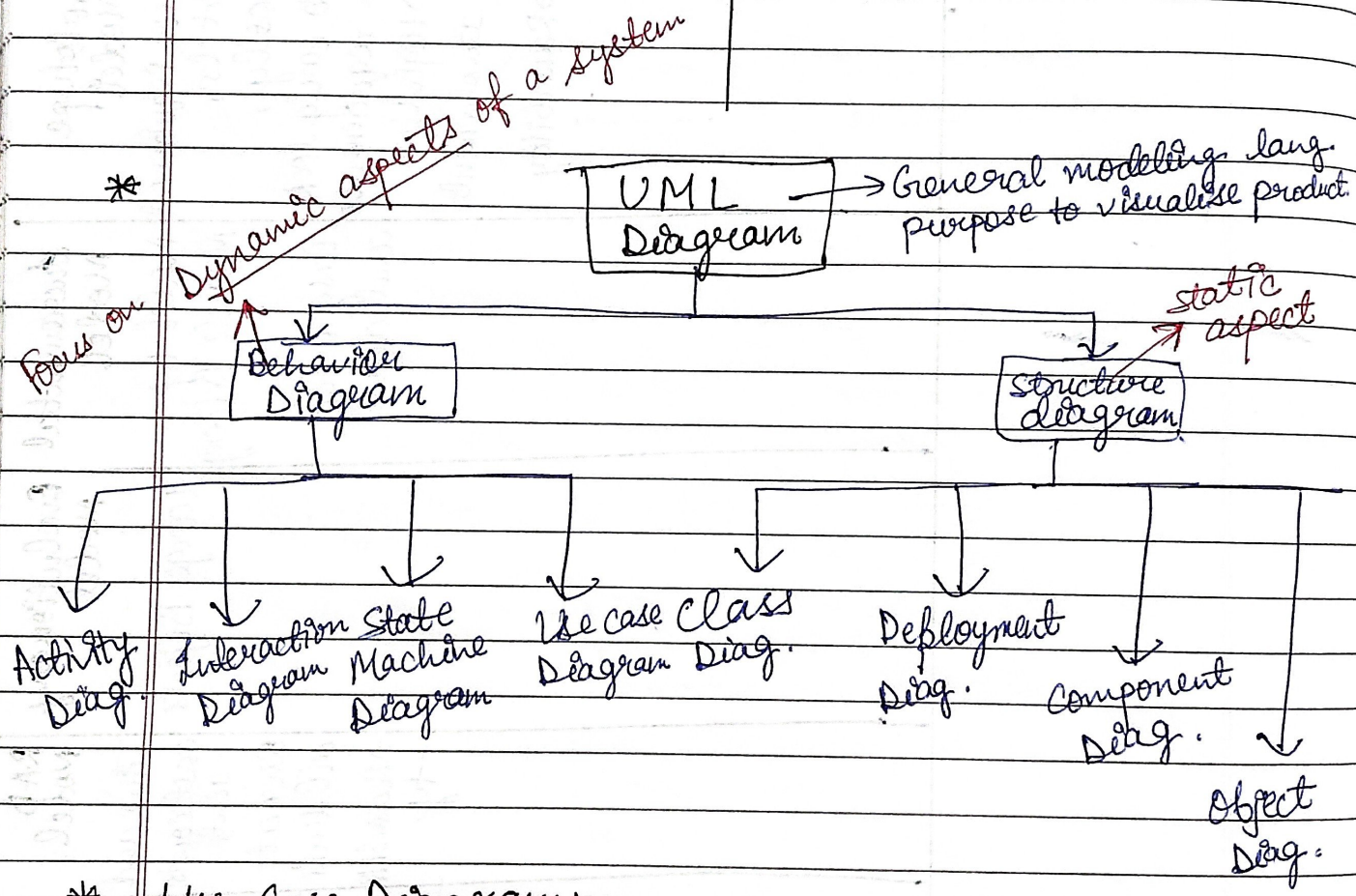
Date __/__/____

Saathi

* Software Design Approaches :-

Function Oriented Design	Object Oriented Design
→ System is designed from a functional viewpoint.	→ System is viewed as a collection of objects (i.e., entities).
→ Top-down approach.	→ Bottom-up approach
→ Divide & Conquer Approach	→ UML is used.
→ DFD is used.	

*



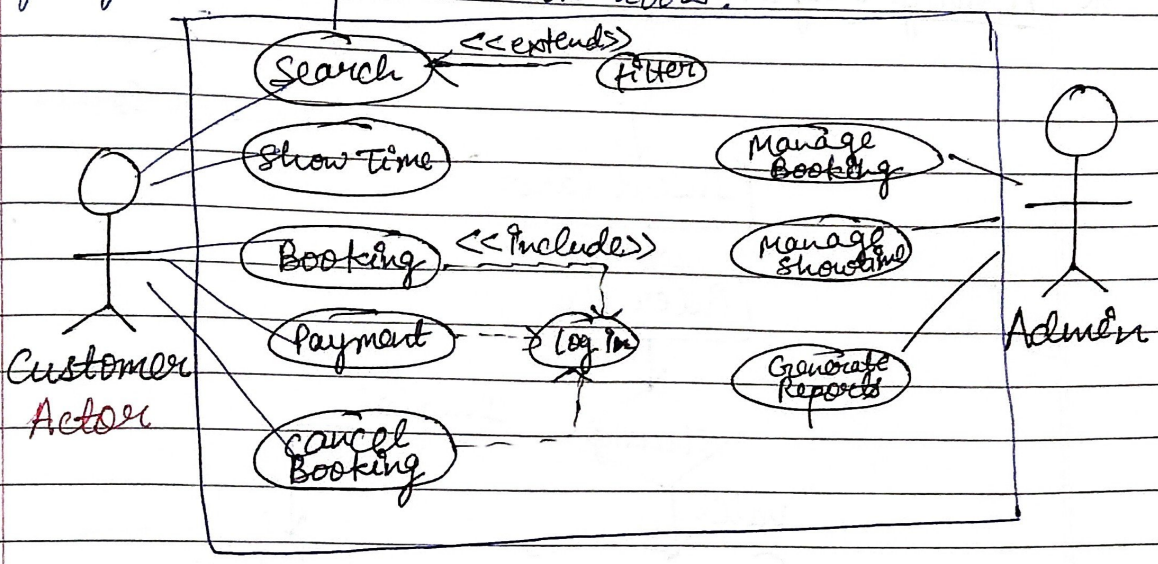
* Use Case Diagram :-

→ Use case diagrams model the behavior of a system.
→ Used to illustrate the functional requirements of the system & its interaction with ext. agent (vectors).
→ It gives us a high-level view of the system without

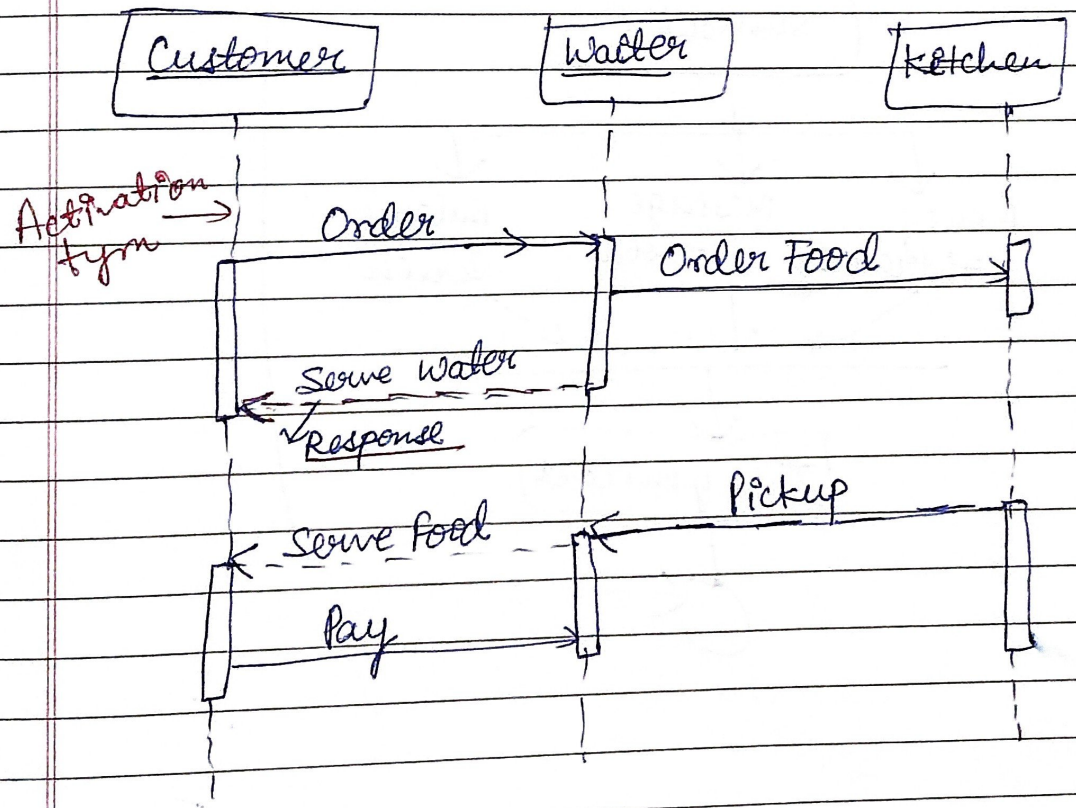
Page No.

Date ___/___/_____ Saathi

going into Implementation tools.



* Sequence Diagram:- (Behavior)



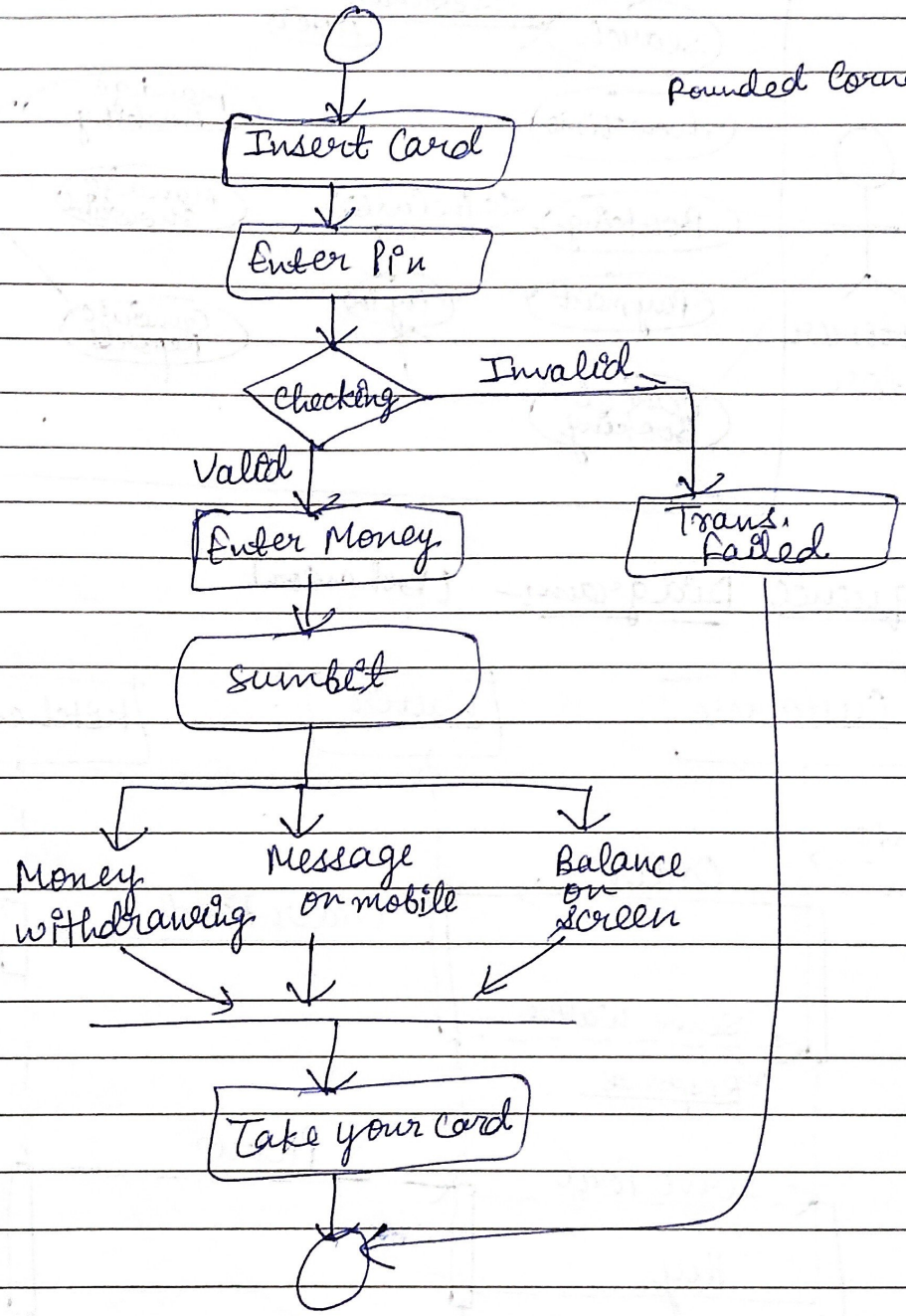
Solid ——→
Respone - - - →

Page No.

Saathi

Date ___ / ___ / _____

* Activity Diagram :— (Behaviour)



Rounded Corner