

* 1- List and Discuss the Requirements Engineering Components.

- 1- Requirement gathering
- 2- Requirement analysis
- 3- Requirement specification

* (Req. Elicitation) help the customer to Define Requirement : System Goal, Business process and Rules

* Refining and modifying the gathered Requirement

* Documenting the Requirement

~~II- What are the types of Requirement engineering~~

* 2- List the Four Activities of the Requirements Engineering Process

1- Requirement Elicitation

Concerned with identifying Requirement Techniques \Rightarrow Interview, Meetings

2- Requirement Negotiation

Categories, Priorities, Examines Requirement Requirement Elicitation and Requirement analysis

3- Requirement Documentation

Usually Done using templates

4- Requirement Validation

Check the over all Req. Documentation

Do Sh

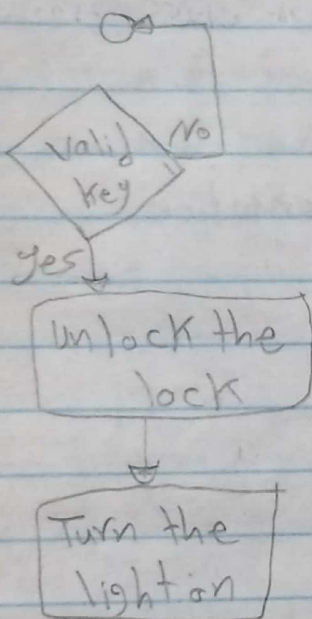


3- state the difference between object-oriented Approaches versus Process-oriented Approaches? (using example)

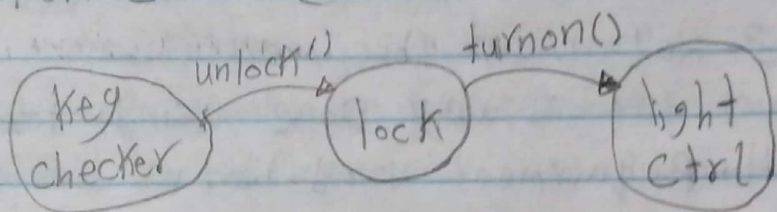
Process-oriented is more intuitive Because it is Person Centric

object-oriented may be more confusing because of Lab or Division

is organization Centric



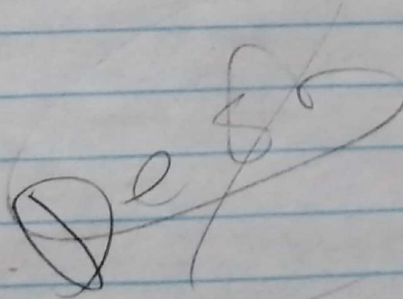
Process oriented



object oriented

EX: Management Problem

EX: individual Centric



4- What is the difference between inheritance and generalization? (Give Examples)

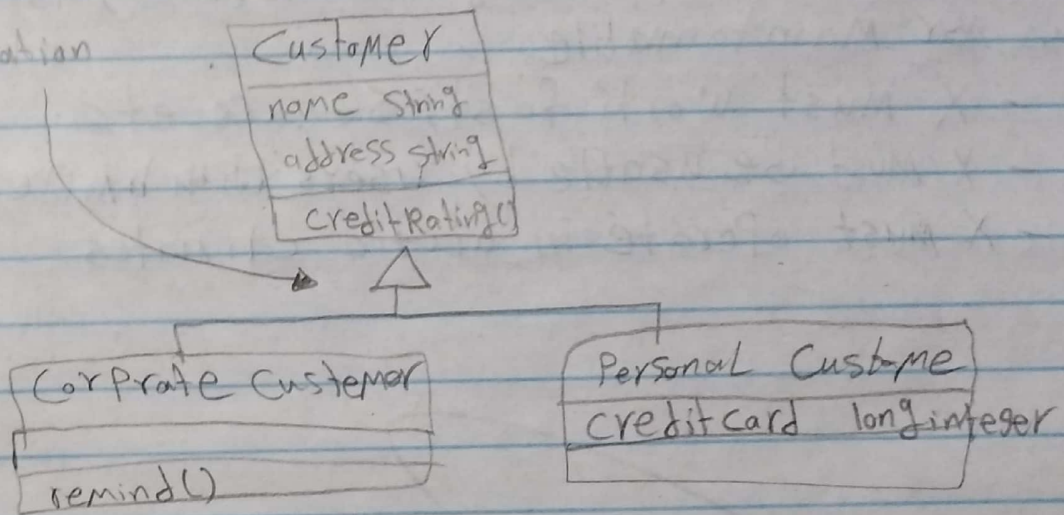
Generalization

- is a Process of extracting shared Characteristics from two or more object
- is an object class
- is used when two classes are similar

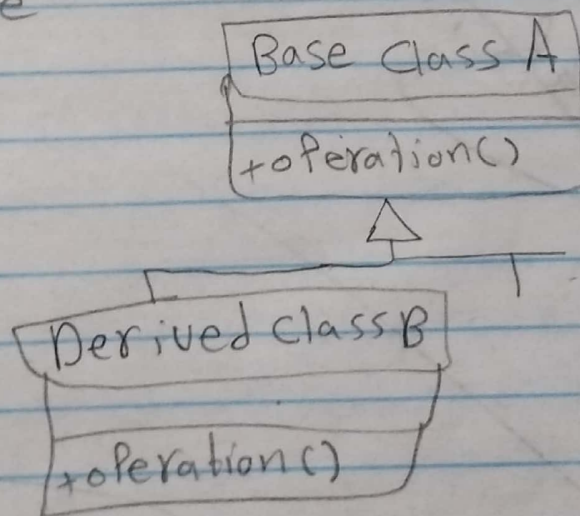
inheritance

inheriting Common Properties through class extension

Ex generalization



Ex inheritance



*
5. Differentiate between functional requirements and non-functional requirements in software development by providing two examples for each along with their types

Functional requirement describes the behavior of the system

- Specification of the Product's functionality
- not a quality
- a directives talks about quality

Non-Functional requirement are qualities that the system should have: such as being secure, fast, usable or Maintainable

- X must work fast, secure, etc
- X must be usable by users with limited skills
- X must operate in arctic climates

Must gbs

6- Describe in details the Waterfall Process for software development?? Which one of the processes in the waterfall process do you consider to be the most important, and why?

سؤال وجوبه سؤال ولكن أنا اعتبره الأهم

⑧ Describe in details the software engineering life cycle? Which one of the stages in the software life cycle do you consider to be the most important and why?

1- Req 1- Requirement Specification

understanding the usage scenarios and deriving the static domain model

2- Design

Assigning responsibilities to objects and specifying detailed dynamics of their interaction under different usage scenarios

3- Implementation

Encoding the design in a programming language

4- testing

individual classes component and the entire system

5- operation and maintenance

Running the system fixing bugs and adding new features



* I - Briefly explain the most important differences between Decomposition by Partitioning vs Projection? (Using example)

They are different ways of slicing the problem.
Partitioning (Horizontal Slicing) - each subgroup of the team responsible for one aspect of the project

Ex: - Restaurant Automation
- Communication overhead

Projection (Vertical Slicing -> stacks)

each subgroup of the team responsible for one function of the project

Independently from the others

Ex: - Visible ownership
- Easy to trace
- Failure - Resilience

De Long

8- Briefly explain the most important differences between Divide Work by Problem vs by Solution?

* By Problem: Different team member responsible for solving subproblem

Drawback:

Advantage: Reduce dependency Potential Redundance

* By solution: Different member responsible for different part of the system

Advantage: Every one can focus on one area of expertise

Drawback: Highly Experience Specialists
(high level programmer)

by problem

- suitable for any level of Expertise

Particularly if uncertain about

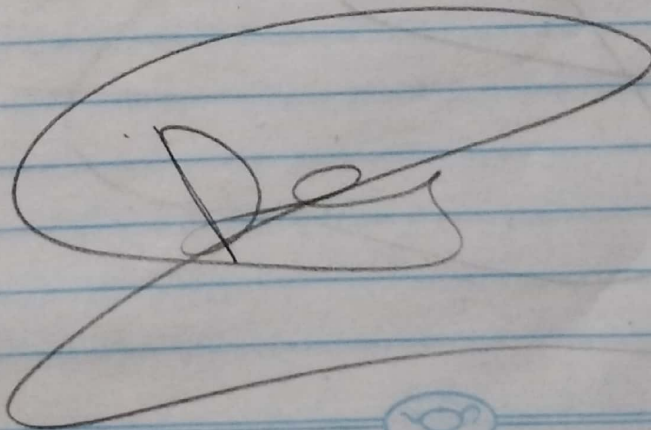
- dependency on other team members

by solution

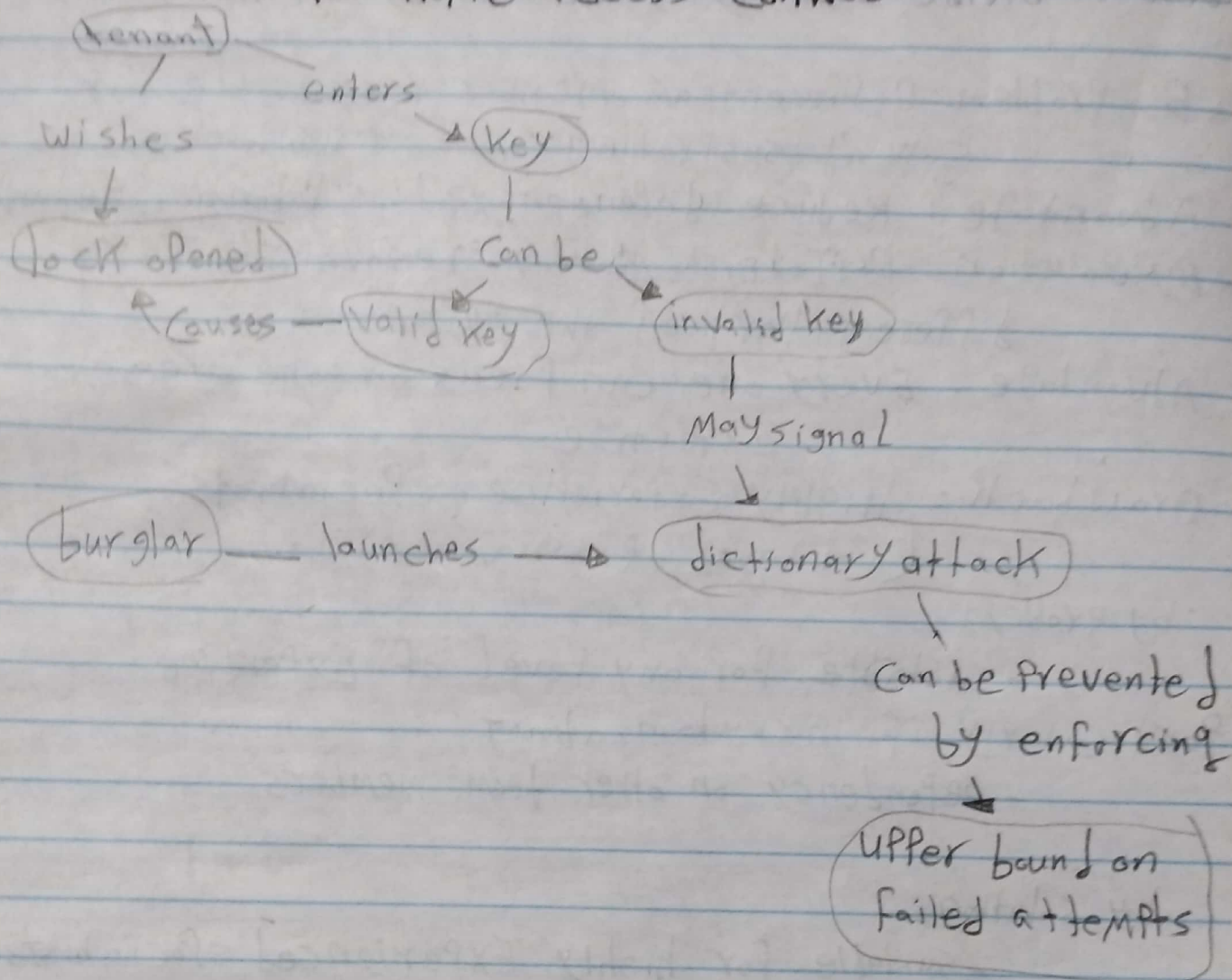
- suitable for highly Experienced specialists

Not for beginners

- Everyone can focus on one area of Expertise

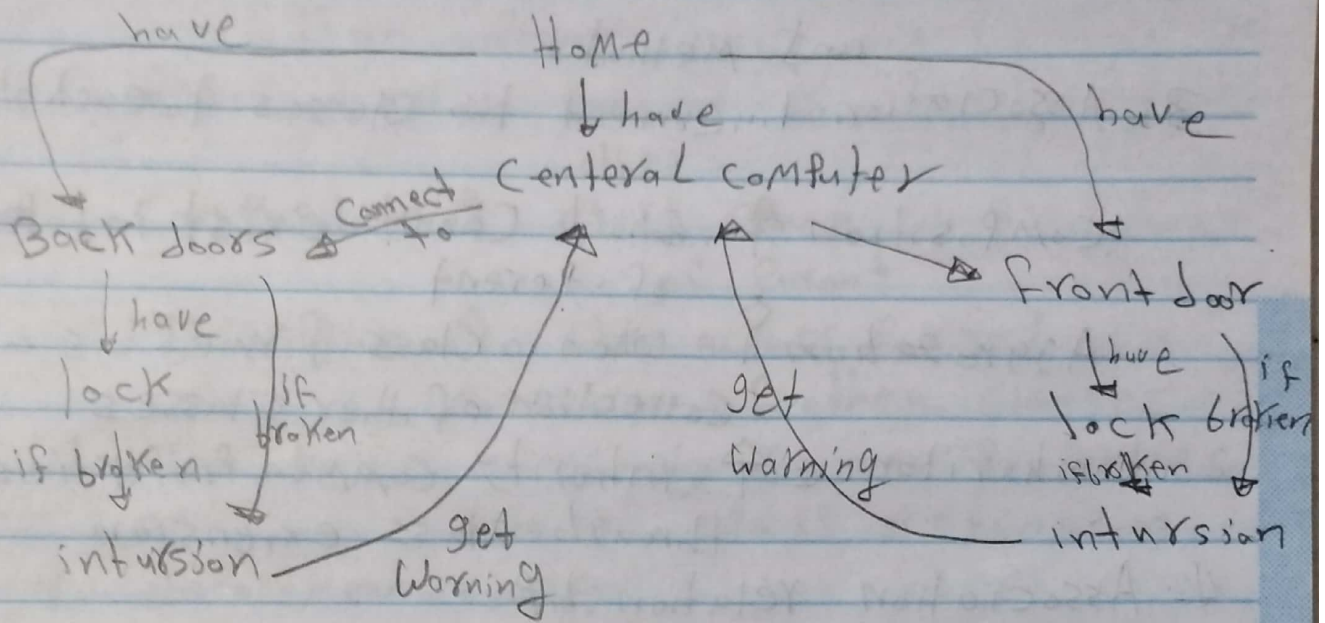


9- Using Concept Map, design Locks and Lighting operation for Home access Control.



Design

10. using Concept Map, design Intrusion Detection and Warning operation for Home access Control



Design

* 2021 Phy

11-2c)

- 1- class diagram
- 2- It represent a class of attributes and methods
- 3- Association | Connect two classes to each other

Composition ♦ Child Can't exist independent of Parent

Aggregation ◊ When a class formed as a collection of other classes

inheritance ↑ inherity common properties through class extension

4- Association relationship

5- a Public attribute in Customer class

2019 Phy

- 2- static : because emphasize the static structure of the system using objects attributes operations and relation

2019 Math stat

2b) 1- Non functional

2- functional

3- non functional

4- functional

5- non functional

6- Non functional

2a) 1- Business Rules

2- Multiplicity

3- Process

4- None

5- Process

6-

7- water fall model

* 2019 Math Stat

2 c)

- 1- Class diagram
- 2- static : it represents the static view of an application
- 3- Association / Connect two classes to each other
 - Composition \blacktriangledown child can't exist independent of parent
 - Aggregation \blacklozenge When a class formed as a collection of other classes
 - inheritance inherity common properties through class extension
- 4- Department - staff
 - Association
 - staff - consultation Room
 - Aggregation

2019 Phy

- 1- Validation 4- interface 7- Scenario
- 2- Software Design 5-
- 3- domain model 6- traceability

