**KLS Gogte Institute of Technology, Belagavi**

Department of Computer Science and Engineering          Academic Year: 2020-21(EVEN)

Program: B.E (CSE)                                                          Semester:4th (All Divisions)

## IA Test – II

Course Title: Software Engineering                          Code:18CS45

Max. Marks:25 marks          Duration: 1 Hour15 Minutes          Date :22-07-2021

**Instructions:**     Answer any 5 complete questions and all carries equal marks.

| Q. No. | | L | CO | PO | [M] |
|---|---|---|---|---|---|
| 1 | Answer the following questions in one sentence, each question carries 1 mark.<br>1. Define Project planning.<br>2. Define Refactoring.<br>3.Define System Modelling.<br>4. Write Full form of UML ------------.<br>5.Write formula to find out "*Effort cost*" by using *Algorithmic cost modeling technique* . | [01] | [3] | ]01] | [5] |
| 2 | Explain different perspectives to develop various software design models? | [02] | [03] | [01] | [5] |
| 3. | For the set of tasks shown below draw the Project Scheduling using Activity Bar Chart (Assume 5 Days=1Week) | [03] | [04] | [03] | [5] |

| Task | Duration | Dependencies |
|---|---|---|
| T1 | 05 | - |
| T2 | 15 | - |
| T3 | 15 | T1(M1) |
| T4 | 10 | - |
| T5 | 10 | T2,T4(M2) |
| T6 | 05 | T1,T2(M3) |
| T7 | 20 | T1(M1) |
| T8 | 25 | T4(M4) |

| Q. No. | | L | CO | PO | [M] |
|---|---|---|---|---|---|
| 4 | Explain   Agile Project Planning Process in XP (Extreme Programming) with neat diagram.(Extreme Programming) | [02] | [04] | [01] | [5] |
| 5 | Draw a Use-case diagram for *Amazon E-commerce web application*,  tabulate the same with at-least 02-actors & 4-use cases. | [03] | [03] | [03] | [5] |
| 6 | Compare and contrast in between Agile method and Plan driven development process. | [02] | [03] | [01] | [5] |
| 7 | Assume that you have been assigned the task of developing *Academic data management system* (**similar to dhi software at GIT**). Analyze the above system and design 1-story card, 1-task card, 1-test card for any one of the specific function related to *Academic data management system*. | [4] | [3] | [03] | [5] |

Onkar Magadum.

1

1] It defines the roles and responsibilities of project managment team. members. It determines project constraints

2] Refactoring is process of changing a software system in such a way that it doesnt alter function of code and improve its internal structure.

3] System Modelling is process of developing abstract models of system with each model presenting different veiw on perpspecctive of system

4] UML - Unified Modeling language.

5] Cost is estimated as mathematical function

$$A \times Size^B \times M$$

## 4. Explain different perspectives to develop various software design models.
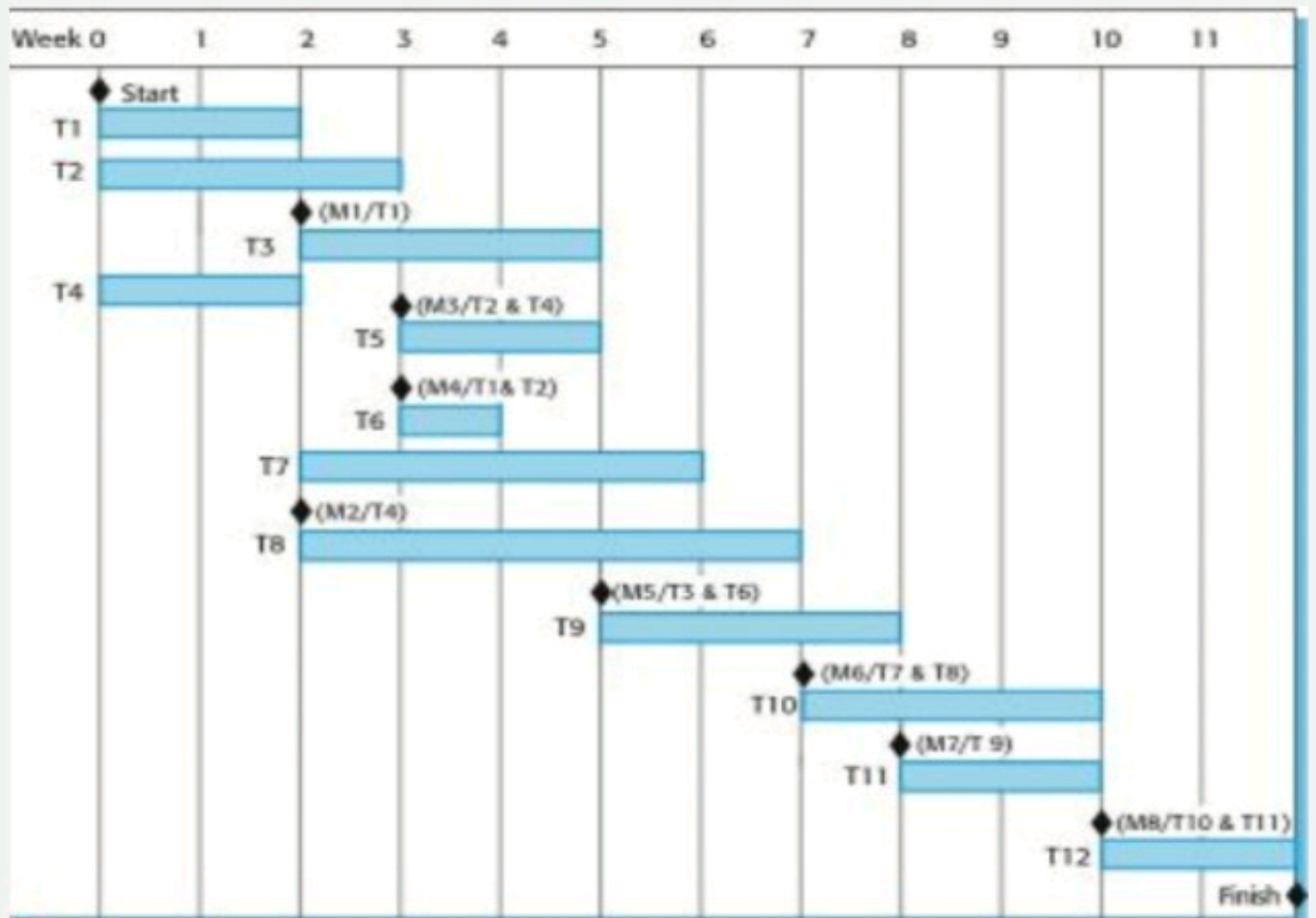
Ans:

- ✧ An external perspective, where you model the context or environment of the system.
- ✧ An interaction perspective, where you model the interactions between a system and its environment, or between the components of a system.
- ✧ A structural perspective, where you model the organization of a system structure of data flow.
- ✧ A behavioral perspective, where you model the dynamic behavior of the system and how it responds to events.
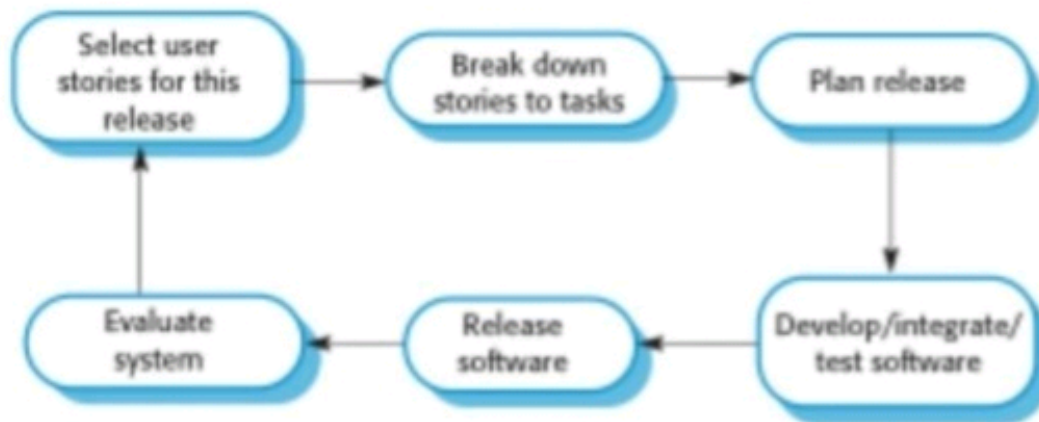
4

Examples:

i. External perspective: Context model.
ii. Interaction perspective: Use case.
iii. Structural perspective: Structural activity.
iv. Behavioural perspective: Behaviour state model.

# ACTIVITY BAR CHART

| Week 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|

◆ Start

T1

T2

◆ (M1/T1)

T3

T4

◆ (M3/T2 & T4)

T5

◆ (M4/T1& T2)

T6

T7

◆ (M2/T4)

T8

◆ (M5/T3 & T6)

T9

◆ (M6/T7 & T8)

T10

◆ (M7/T 9)

T11

◆ (M8/T10 & T11)

T12

Finish ◆

In an XP process, customers are intimately involved in specifying and prioritizing system requirements. The requirements are not specified as lists of required system functions. Rather, the system customer is part of the development team and discusses scenarios with other team members. Together, they develop a 'story card' that encapsulates the customer needs. The development team then aims to implement that scenario in a future release of the software. An example of a story card for the mental

**Figure 3.4** Extreme programming practices

health care patient management system is shown in Figure 3.5. This is a short description of a scenario for prescribing medication for a patient.

The story cards are the main inputs to the XP planning process or the 'planning game'. Once the story cards have been developed, the development team breaks these down into tasks (Figure 3.6) and estimates the effort and resources required for implementing each task. This usually involves discussions with the customer to refine the requirements. The customer then prioritizes the stories for implementation, choosing those stories that can be used immediately to deliver useful business support. The intention is to identify useful functionality that can be implemented in about two weeks, when the next release of the system is made available to the customer.

Of course, as requirements change, the unimplemented stories change or may be discarded. If changes are required for a system that has already been delivered, new story cards are developed and again, the customer decides whether these changes should have priority over new functionality.
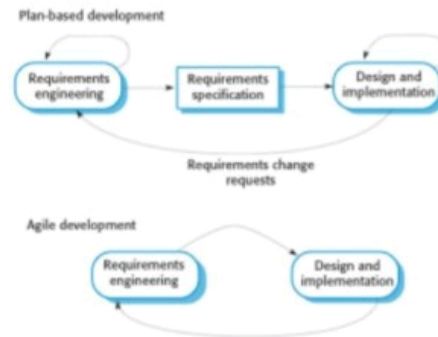
**12.** Draw Sequence diagram for make payment use case in Amazon E-commerce web application.

Ans:
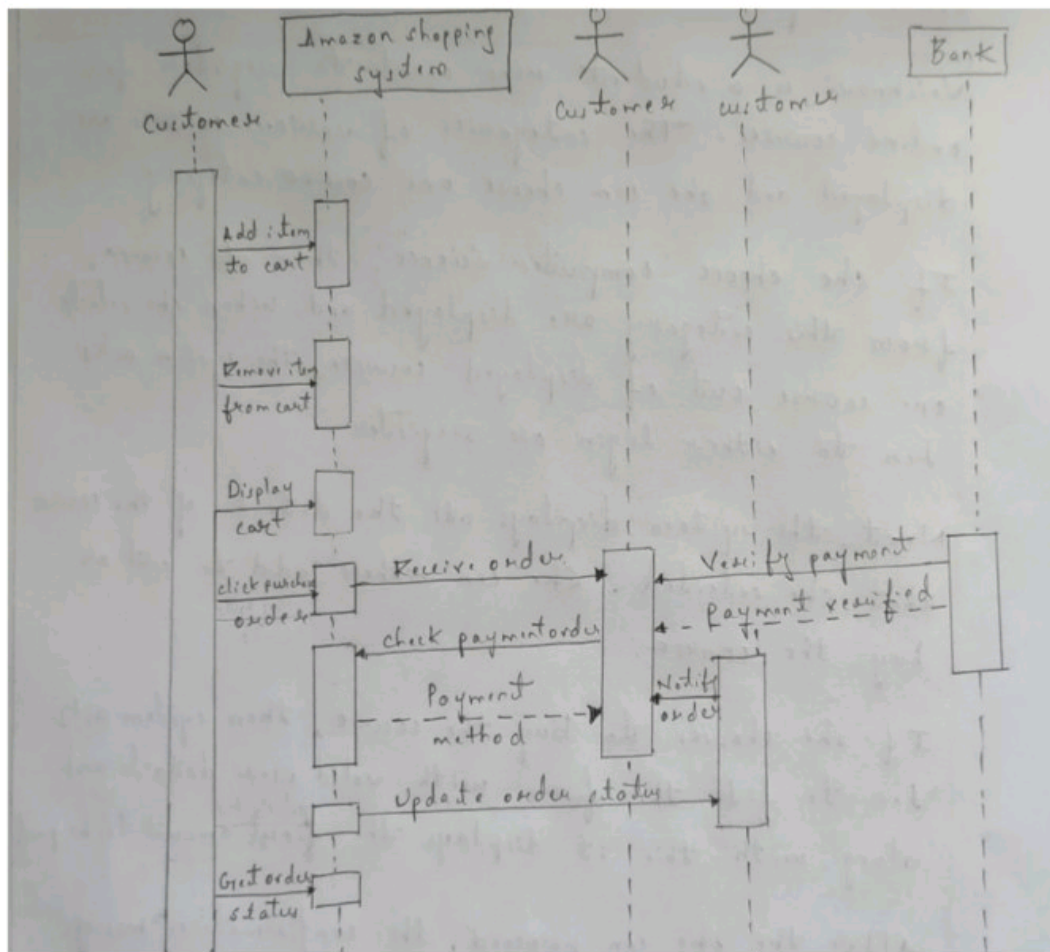
## 2. Compare and contrast in between Agile method and iterative development process.

Ans:



## Technical, human, organizational issues

- Most projects include elements of plan-driven and agile processes. Deciding on the balance depends on:
  - Is it important to have a very detailed specification and design before moving to implementation? If so, you probably need to use a plan-driven approach.
  - Is an incremental delivery strategy, where you deliver the software to customers and get rapid feedback from them, realistic? If so, consider using agile methods.
  - How large is the system that is being developed? Agile methods are most effective when the system can be developed with a small co-located team who can communicate informally. This may not be possible for large systems that require larger development teams so a plan-driven approach may have to be used.

2

- What type of system is being developed?
  - Plan-driven approaches may be required for systems that require a lot of analysis before implementation (e.g. real-time system with complex timing requirements).
- What is the expected system lifetime?
  - Long-lifetime systems may require more design documentation to communicate the original intentions of the system developers to the support team.
- What technologies are available to support system development?
  - Agile methods rely on good tools to keep track of an evolving design
- How is the development team organized?
  - If the development team is distributed or if part of the development is being outsourced, then you may need to develop design documents to communicate across the development teams.

- Are there cultural or organizational issues that may affect the system development?
  - Traditional engineering organizations have a culture of plan-based development, as this is the norm in engineering.
- How good are the designers and programmers in the development team?
  - It is sometimes argued that agile methods require higher skill levels than plan-based approaches in which programmers simply translate a detailed design into code
- Is the system subject to external regulation?
  - If a system has to be approved by an external regulator (e.g. the FAA approve software that is critical to the operation of an aircraft) then you will probably be required to produce detailed documentation as part of the system safety case.