Q2) a) Uk based KPO company working on “License Management System”. They are using number of expensive and licenced software tools. More than 5000 team members are sharing these tools. You have been deputed as project manager to ensure that a project finishesh within original budget, with the required scope of work and within the required timescales, and to ensure that throughout this process all the stakeholders, especially the client , are satisfied with the project results. [6]

Prepare Risk management process based on below points

i) Risk identification

ii) Risk Analaysis

iii) Risk mitigation

--> Risk Management Process for the "License Management System" project:

i) Risk Identification:

Conduct a thorough analysis of the project and identify potential risks related to budget, scope, and timeline.

Engage the project team, stakeholders, and subject matter experts to gather their insights and identify risks specific to the project.

Consider risks associated with the use of expensive and licensed software tools, such as licensing issues, compatibility problems, or unexpected costs.

Identify risks related to team coordination and communication, client requirements, resource availability, and technology.

ii) Risk Analysis:

Assess the identified risks by evaluating their likelihood of occurrence and potential impact on the project's objectives.

Prioritize the risks based on their severity and the potential consequences they may have on the budget, scope, and timeline.

Analyze the potential causes, triggers, and warning signs associated with each risk to enhance risk detection and response.

Assign a risk score or rating to each identified risk based on its likelihood and impact, enabling the team to focus on high-priority risks.

iii) Risk Mitigation:

Develop a comprehensive risk mitigation plan to address the identified risks effectively.

Identify specific actions, strategies, and contingency plans to minimize or eliminate the risks.

Assign responsibilities to team members for implementing risk mitigation measures and monitoring the progress.

Establish communication channels and protocols to promptly escalate and address any potential risks or issues.

Regularly review and update the risk mitigation plan throughout the project's lifecycle to ensure its effectiveness.

Implement proactive measures to enhance client satisfaction, such as regular progress reporting, frequent communication, and involvement in decision-making processes.

Conduct risk assessment and mitigation reviews periodically to identify new risks, assess the effectiveness of existing mitigation measures, and make adjustments as needed.

Remember, risk management should be an ongoing process throughout the project's lifecycle. Regular monitoring, evaluation, and adjustment of the risk management plan will help ensure that the project finishes within the original budget, with the required scope of work, and within the required timescales, while satisfying all stakeholders, especially the client.

Explain Agile reports in brief.

--> Agile reporting refers to the use of selected metrics and tools for analyzing and making informed decisions about the way work is managed. The goal of reporting in Agile is to ensure that work systems deliver outcomes - value with optimum quality and increased pace. Hence, Agile teams use different sets of metrics to achieve greater visibility over the work processes, measure how efficient their process is, track their performance over time, identify process blockages, and more. The reporting method is usually defined by the work methodology that a team chooses to apply.

Agile Reporting vs. Traditional Reporting

The underlying difference between reporting in traditional and Agile project management lies in the goals, metrics, and constraints of the two approaches. The Agile philosophy seeks outcomes. It shifts the work ethic toward finding ways to improve customer satisfaction by delivering greater value *as a team*. The traditional way of work management, however, emphasizes output and helps to answer how people on the team have performed *on an individual level*. Traditional leaders often measure a team's success on the basis of how well a delivery fits in with the budget and time constraints. The approach can be very limiting when discussing outcomes such as creating real value and increasing customer satisfaction. Therefore, reporting in Agile aims to "zoom-out" and identify how a team operates as a whole, how efficient the current process is, and how to improve.

The preferred set of metrics for project reporting in the traditional way provides a narrower view of a project's success. Decision-making is then based purely on specific indicators and their numeric values. Whereas, Agile project managers take into account various factors such as context, actual customers' problem resolution, process roadblocks, or delivery over time.

Why Is Agile Reporting Important?

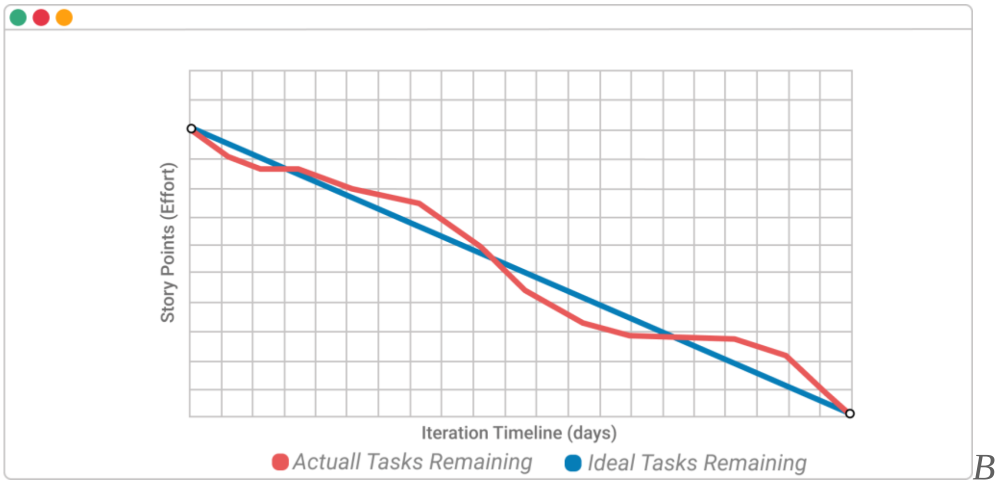
As outlined previously, Agile reporting supports decision-making, helps to achieve outcomes that create customer value, and offers the ability to track progress in real-time. Here are the three critical aspects of Agile reports.

- **Process Outcomes**. Outcomes of an Agile project report can uncover process insights (bottlenecks, dependencies, and improvement opportunities) and team performance clues that support informed decision-making. You can also track the progress of projects, programs, or portfolios and map them against planning.
- **Customer Value**. Creating real customer value can be measured by analyzing how efficient your team is in resolving customers' issues, or how value is delivered over time.
- **Service Delivery**. Meeting customers' expectations is a cornerstone of the Agile mindset. Understanding how fit-for-purpose your process is unlocks insights into optimizing your service delivery to increase customer satisfaction and into better future delivery predictions.

Agile Reporting Tools

To make the most of your reports, the right tools are your best ally. Today's market offers a generous choice of tools to help you get real-time reports and draw insights into your work process, team performance, or another indicator of success.

- **Burndown Chart**. The chart is applied by Scrum teams and helps managers track how many user stories have been completed for a given iteration (sprint). The report allows them to predict how many sprints are needed to complete specific pieces of work, such as epics, projects, or themes.



Burndown chart

Q3) a) Consider a project with the following functional units. [6]

i)Number of user inputs = 15
ii) Number of user outputs = 12
iii)Number of user equiries = 07
iv) Number of user files = 20
v) Number of external interfaces = 05

In addition to the above, system requires significant.

- Data communication(5)
- Performance is very critical (4)
- Designed code may be moderately reusable

Other complexity factors are treated as Average.

Compute the functional point for the project.

-->To compute the functional points for the project, we need to consider the complexity factors and the weights assigned to each functional unit.

Here is the breakdown of the functional points calculation based on the provided information:

Step 1: Calculate the Unadjusted Function Point (UFP)

UFP = (Number of user inputs × Weight for user inputs) +
(Number of user outputs × Weight for user outputs) +
(Number of user inquiries × Weight for user inquiries) +
(Number of user files × Weight for user files) +
(Number of external interfaces × Weight for external interfaces)

Given:
Number of user inputs = 15
Weight for user inputs = 3
Number of user outputs = 12
Weight for user outputs = 4
Number of user inquiries = 7
Weight for user inquiries = 3
Number of user files = 20
Weight for user files = 10
Number of external interfaces = 5
Weight for external interfaces = 4

UFP = (15 × 3) + (12 × 4) + (7 × 3) + (20 × 10) + (5 × 4)
= 45 + 48 + 21 + 200 + 20
= 334

Step 2: Calculate the Value Adjustment Factor (VAF)
To calculate the Value Adjustment Factor (VAF), we need to consider the complexity factors and their corresponding weights.

Given:
Data communication = 5
Performance = 4
Code reuse = 2
Average complexity factors = 14 (assumed)

VAF = 0.65 + (0.01 × (5 + 4 + 2 + 14))

= 0.65 + (0.01 × 25)
= 0.65 + 0.25
= 0.9

Step 3: Calculate the Adjusted Function Point (AFP)
AFP = UFP × VAF
= 334 × 0.9
= 300.6

The functional points for the project, considering the given information, is approximately 300.6.

b) What are the various agile principles? [4]

-->1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

4. Business people and developers must work together daily throughout the project.

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

7. Working software is the primary measure of progress.

8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

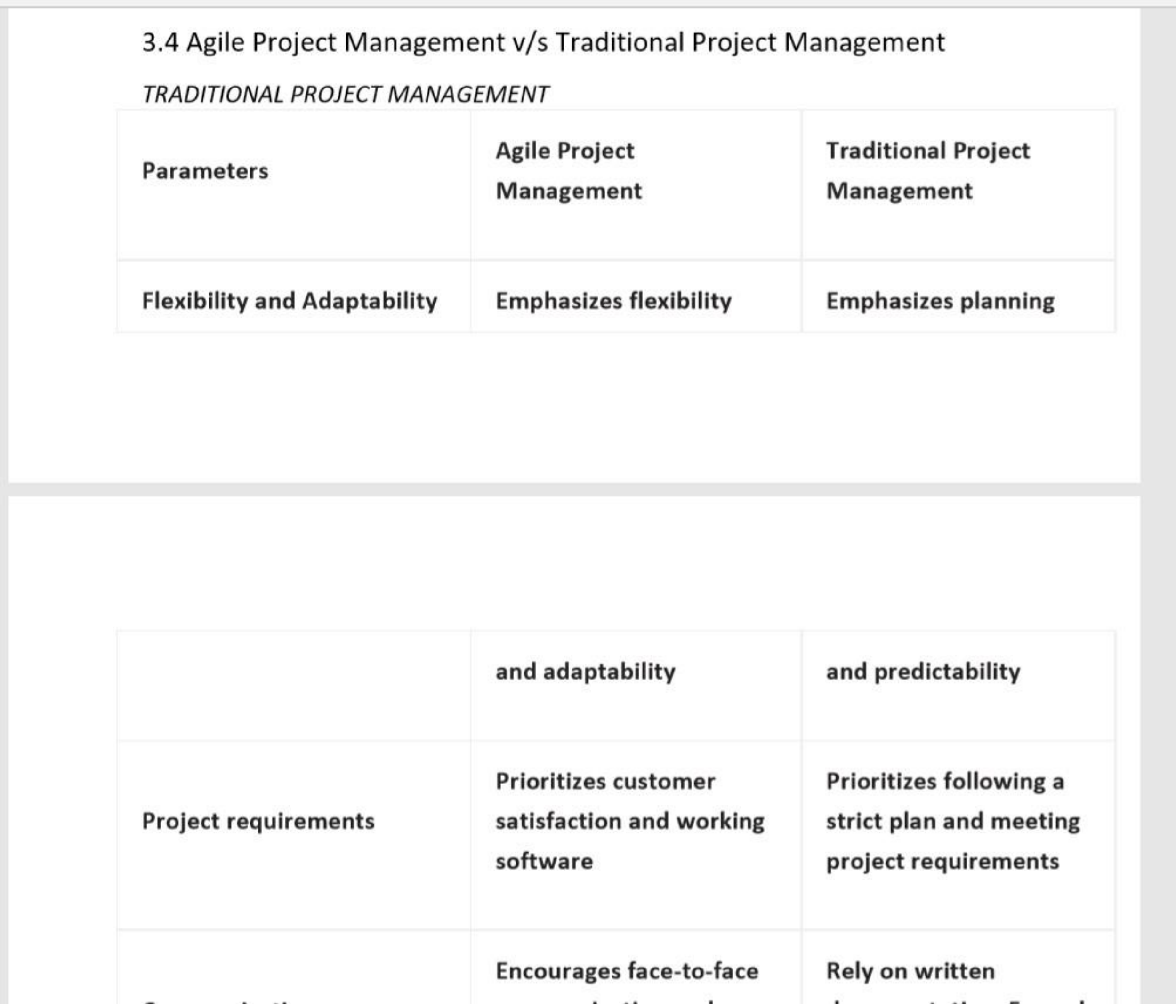
9. Continuous attention to technical excellence and good design enhances agility.

10. Simplicity–the art of maximizing the amount of work not done–is essential.

11 .The best architectures, requirements, and designs emerge from self-organizing teams.

12 .At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

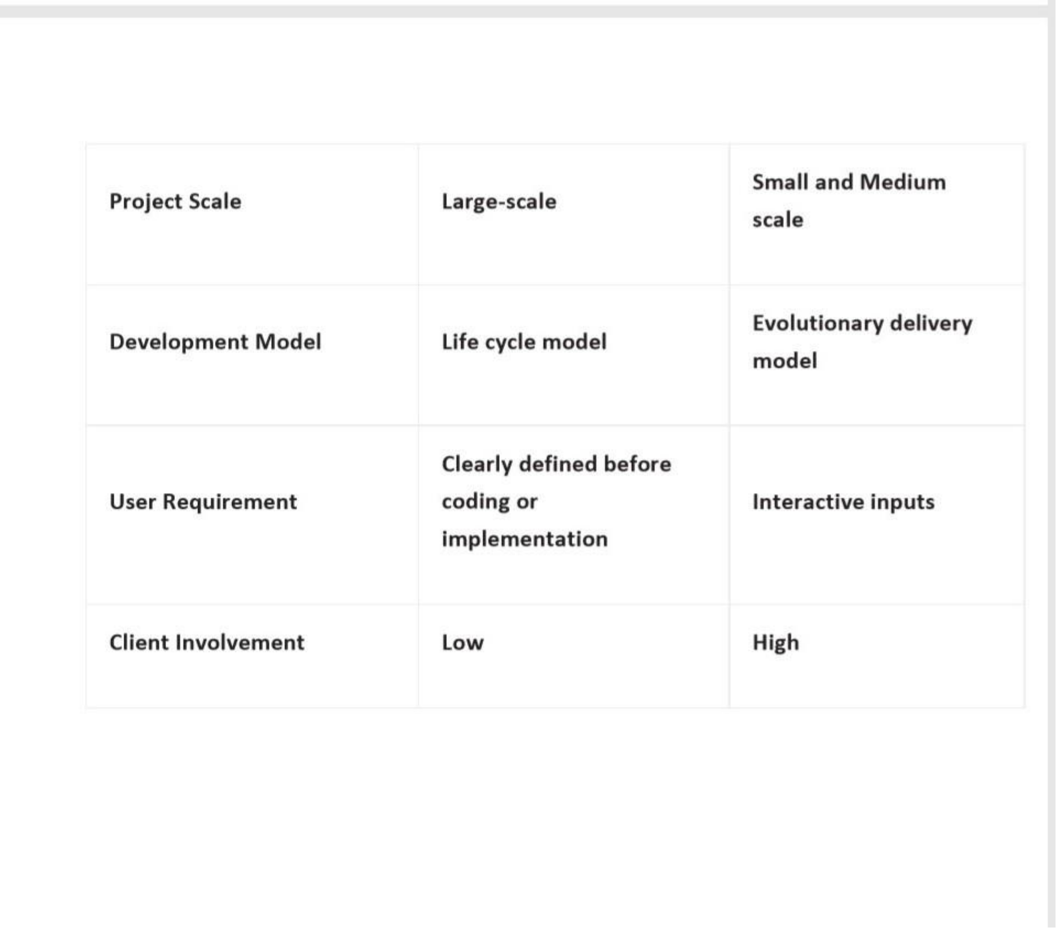
Differentiate agile project management Vs traditional project management. [4]



3.4 Agile Project Management v/s Traditional Project Management

TRADITIONAL PROJECT MANAGEMENT

Parameters	**Agile Project Management**	**Traditional Project Management**
Flexibility and Adaptability	**Emphasizes flexibility and adaptability**	**Emphasizes planning and predictability**
Project requirements	**Prioritizes customer satisfaction and working software**	**Prioritizes following a strict plan and meeting project requirements**
	Encourages face-to-face	**Rely on written**



Project Scale	Large-scale	Small and Medium scale
Development Model	Life cycle model	Evolutionary delivery model
User Requirement	Clearly defined before coding or implementation	Interactive inputs
Client Involvement	Low	High

What are the various techniques to estimate story points in agile? [6]

--> In Agile software development, story points are used to estimate the relative effort and complexity of user stories or product backlog items. Several techniques can be used to estimate story points in Agile, including:

Planning Poker: Team members use a deck of planning poker cards with values representing story points (e.g., Fibonacci sequence) and discuss and estimate the story points together. The team reaches a consensus on the estimated effort for each user story.

T-Shirt Sizes: Rather than using specific numerical values, team members assign sizes to user stories, such as XS, S, M, L, or XL, indicating the relative effort and complexity.

Bucket System: Stories are grouped into buckets or categories based on their complexity or size. For example, stories might be categorized as small, medium, or large. This approach provides a high-level estimation without assigning specific numerical values.

Comparative Sizing: User stories are compared to previously completed stories with known story points. The team discusses whether a new story is smaller, larger, or similar in complexity to the reference stories and assigns corresponding story points.

Affinity Mapping: The team groups user stories based on their similarities and assigns a common story point value to each group. This technique helps identify patterns and reduces the number of individual estimations required.

Baseline: A reference user story with a known effort level is chosen as a baseline, and other stories are estimated relative to that baseline. This technique helps maintain consistency and allows for easier comparison between different user stories.

Expert Judgment: Experienced team members or subject matter experts provide their estimates based on their knowledge and past experience. The team discusses and aligns their estimations through collaboration.

It's important to note that these techniques are meant to provide relative estimates, not precise measurements of effort or time. The focus is on comparing and prioritizing user stories based on their relative complexity, allowing for better planning and resource allocation in Agile projects.

How to facilitate retrospective process in agile management with suitable example. [6]

--> Let's say you're a Scrum Master facilitating a retrospective for a software development team that has just completed a two-week sprint. The team consists of five members: developers, a tester, and a product owner.

Set the Stage:

Start the retrospective by welcoming the team and reminding them of the purpose of the retrospective: to reflect on the previous sprint, identify what went well, and determine areas for improvement.

Create a Safe Environment:

Emphasize the importance of psychological safety and encourage open and honest communication. Assure the team that the retrospective is a safe space to share their thoughts and ideas without fear of judgment.

Gather Data:

Present the team with relevant data from the sprint, such as the number of user stories completed, the team's velocity, any incidents or issues encountered, customer feedback, and any metrics or observations that can provide insight into the team's performance.

Generate Insights:

Facilitate a discussion where team members share their insights about the sprint. For example, a developer might mention that they faced frequent interruptions, which impacted their productivity, while the tester might express satisfaction with the improved quality of the software.

Identify Improvement Opportunities:

Based on the insights generated, guide the team to identify specific improvement opportunities. For instance, the team might identify the need to establish better communication channels to minimize interruptions or improve collaboration between developers and the tester to address quality concerns.

Prioritize and Plan:

Collaboratively prioritize the improvement opportunities based on their impact and feasibility. The team decides to address the communication issue first, as it had a significant impact on productivity. They plan to introduce dedicated focus time for developers and establish clear guidelines for minimizing interruptions.

Close the Retrospective:

Summarize the key discussion points, decisions, and action items. Assign responsibilities to team members, such as the Scrum Master taking the lead on implementing the focus time initiative and the product owner communicating the new guidelines. Set a timeline for these improvements to be implemented and express appreciation for the team's participation and commitment to continuous improvement.

In the next sprint, the team will apply the identified improvements and evaluate their effectiveness during the subsequent retrospective, thereby fostering a culture of continuous learning and improvement.

b) Write short note (any one) [4]

i) Overview of project management framework

A project management framework provides a structured approach for planning, executing, and controlling projects. It offers a set of guidelines, processes, tools, and techniques to help project managers and teams effectively manage projects from initiation to closure. Here's an overview of a typical project management framework:

Project Initiation:

This phase involves defining the project's objectives, scope, stakeholders, and deliverables. It includes conducting a feasibility study, defining project requirements, and identifying key project team members.

Project Planning:

In this phase, the project manager develops a comprehensive project plan that outlines the project's scope, schedule, budget, resources, and risk management strategies. It involves creating work breakdown structures, defining project tasks, estimating effort and duration, and developing a communication plan.

Project Execution:

The execution phase focuses on implementing the project plan, coordinating resources, and executing project tasks. The project manager monitors progress, manages changes, resolves issues, and ensures that project activities are on track to meet objectives.

Project Monitoring and Control:

This phase involves tracking project progress, comparing actual performance against planned performance, and implementing corrective actions when necessary. It includes monitoring project risks, managing stakeholder expectations, and ensuring adherence to quality standards.

Project Closure:

The closure phase marks the end of the project. It involves verifying deliverables, obtaining client acceptance, conducting project reviews, documenting lessons learned, and transitioning project results to operations or maintenance.

Project management frameworks, such as the Project Management Institute's (PMI) Project Management Body of Knowledge (PMBOK) or PRINCE2, provide a standardized approach to project management, helping organizations improve project success rates, increase efficiency, and enhance communication and collaboration.

ii) Capability Maturity Model (CMM):

The Capability Maturity Model (CMM) is a framework used to assess and improve an organization's ability to develop and maintain software and systems. It provides a maturity model consisting of five levels, each representing different stages of organizational process improvement and capability. Here's an overview of the CMM levels:

Initial Level:

At the initial level, processes are ad hoc, unpredictable, and poorly controlled. Organizations at this level often face challenges in delivering consistent results and struggle with managing projects effectively.

Repeatable Level:

At the repeatable level, organizations establish basic project management processes and practices that are documented and repeatable. They focus on tracking and controlling costs, schedules, and quality to deliver more consistent outcomes.

Defined Level:

At the defined level, organizations have well-defined and documented processes that are followed consistently across projects. Standardization and process integration play a significant role, ensuring effective coordination and communication.

Managed Level:

At the managed level, organizations focus on quantitative management and analysis of processes. They collect data, monitor performance, and use metrics to make data-driven decisions for process improvement.

Optimizing Level:

At the optimizing level, organizations continuously improve their processes through innovative technologies, automation, and feedback mechanisms. They proactively identify opportunities for optimization and leverage best practices to achieve maximum efficiency and effectiveness.

The Capability Maturity Model helps organizations assess their process maturity, identify areas for improvement, and establish a roadmap for process enhancement. It promotes a culture of continuous improvement, enabling organizations to enhance their capabilities, deliver higher-quality products, and achieve better customer satisfaction.