will start the session by 10:10 am

Agile is a methodology used in Project management (software dev). it emphasises flexibility , collaborations, and iterative progress.

Flexibility and Adaptability:

Agile: Agile is highly flexible and adaptable to change

Traditional: Traditional approaches, such as Waterfall, project has started.

Project Delivery:

Agile: Agile focuses on delivering working software in

Traditional: Traditional approaches aim to deliver the

Customer Involvement:

Agile: Agile encourages regular collaboration with cus delivered product meets their needs.

Traditional: Traditional approaches involve less custom limited opportunities for feedback.

es in requirements, allowing for iterative development and continuous feedback.
follow a linear, sequential process, making it challenging to accommodate changes once the

short, iterative cycles (sprints), providing value to customers early and continuously.
entire project at once, often resulting in longer development cycles and delayed value delivery.

stomers and stakeholders throughout the project, incorporating their feedback to ensure the

mer involvement during the development process, with requirements being defined upfront and

Risk Management:

Agile: Agile emphasizes risk management through its

Traditional: Traditional approaches often have less em

Documentation:

Agile: Agile values working software over comprehen

Traditional: Traditional approaches often require exter

Team Structure:

Agile: Agile promotes self-organizing, cross-functiona

Traditional: Traditional approaches may have a more h

Adaptability to Change:

Agile: Agile is well-suited for projects with evolving re

Traditional: Traditional approaches are more suitable f

Product Owner: The Product Owner is r
Their main responsibilities include:

Defining the product vision, goals, an

Prioritizing the product backlog base

Collaborating with stakeholders to ga

Accepting or rejecting work results.

Scrum Master: The Scrum Master is resp
responsibilities include:

Facilitating Scrum events (such as spr

Removing impediments to the team's

Coaching the team on Agile principle

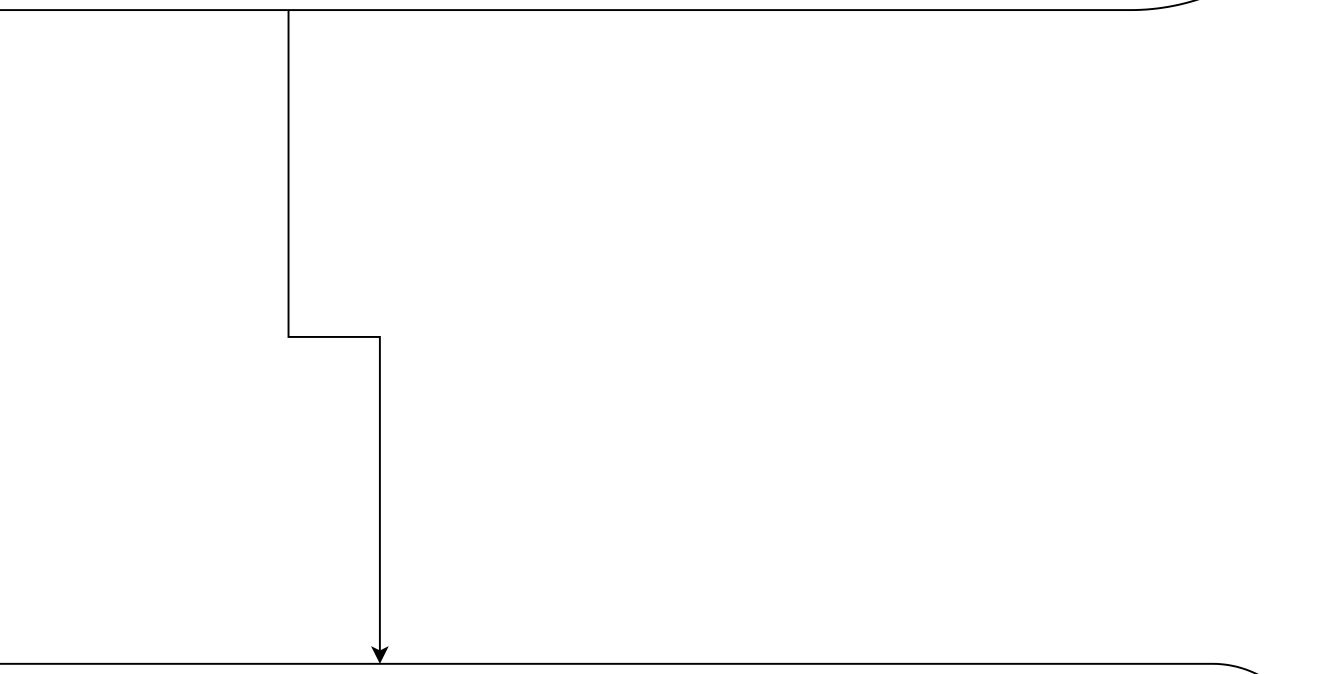
Helping the team improve its effectiv

iterative approach, allowing teams to identify and address risks early in the project lifecycle.
phasis on risk management, as risks are typically identified and addressed later in the project.

sive documentation, although necessary documentation is still produced as needed.
nsive documentation, including detailed requirements, design documents, and test plans.

al teams that collaborate closely on project tasks.
hierarchical team structure, with clear roles and responsibilities defined upfront.

equirements or where rapid development and delivery are required.
for projects with well-defined requirements and where changes are less likely to occur.



esponsible for maximizing the value of the product and the work of the development team.

nd roadmap.
ed on business value and stakeholder feedback.
ather requirements and feedback.

ponsible for ensuring that Scrum is understood and implemented correctly. Their main

rint planning, daily stand-ups, sprint reviews, and retrospectives).
s progress.
es and practices.
veness and efficiency.

PO

Product Owner

Responsibilities

- Develop an
- Define and
- Collaborate
- Accept or r
  standards.

Skills and Qua

- Strong com
- Decision-m
- Domain kn
- Ability to a

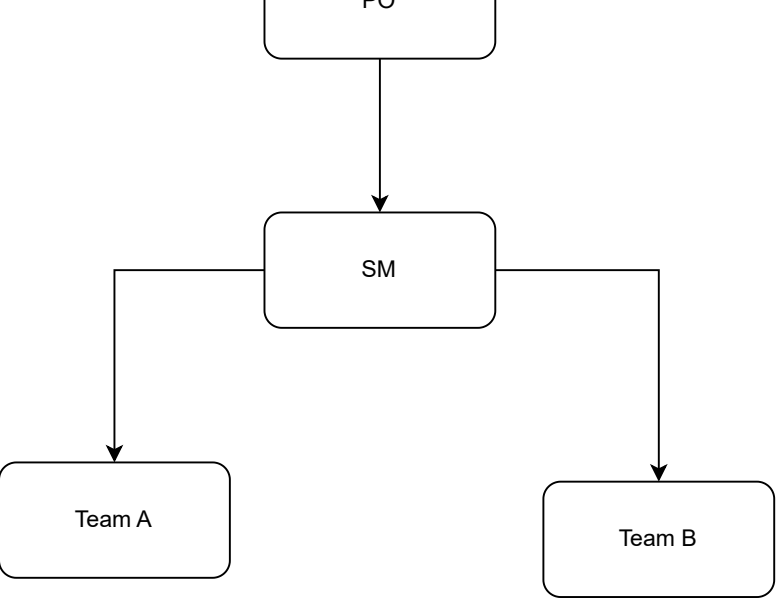
Scrum Master:

Responsibilities

- Facilitate Sc
  effective an
  Coach the
  efficiency.
  Remove im
  with stakeh
  Help the te

Skills and Qua

- Strong facil
- Servant lea
- Knowledge
- Conflict res



r:

es:

d communicate the product vision, goals, and roadmap to the development team.

prioritize the product backlog items based on business value, ROI, and stakeholder feedback.

e with stakeholders, customers, and the development team to gather requirements and feedback.

eject work results based on the acceptance criteria and ensure that the product increment meets the quality

lities:

mmunication and collaboration skills to work effectively with stakeholders and the development team.

aking skills to prioritize and make trade-offs between features based on business value and market needs.

owledge and understanding of customer needs to guide product development.

dapt to change and respond to feedback to ensure the product meets the evolving requirements.

es:

crum events, including sprint planning, daily stand-ups, sprint reviews, and retrospectives, to ensure they are
d focused.

development team on Agile principles, practices, and Scrum framework to improve their performance and

pediments that hinder the team's progress and facilitate collaboration and communication within the team and
olders.

am self-organize and continuously improve their processes and practices to deliver high-quality products.

lities:

itation and interpersonal skills to lead Scrum events and foster collaboration within the team.

dership mindset to support the team and empower them to make decisions and solve problems.

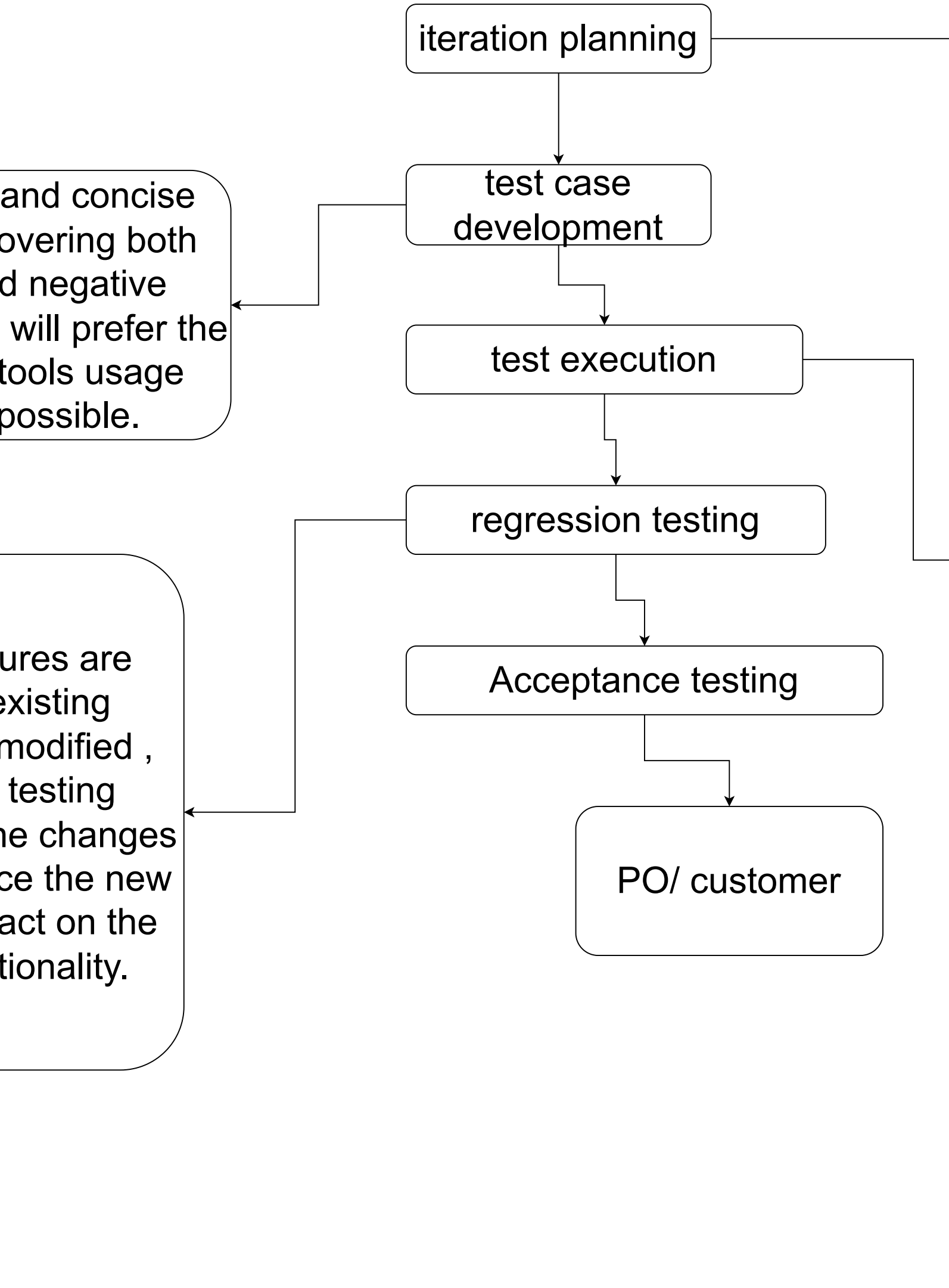
of Agile principles and practices to guide the team in adopting and implementing Agile methodologies.

olution and problem-solving skills to address issues and impediments that affect the team's progress.

This phase involves reviewing user stories and

writing clear
test cases, c
positive an
scenarios, and
automation
where is

as new feat
added or e
features are
regression
ensures that th
do not introduc
defect or impa
existing funct


iteration planning
test case development
and concise
overing both
d negative
will prefer the
tools usage
possible.
test execution
regression testing
ures are
existing
modified ,
testing
he changes
ce the new
act on the
tionality.
Acceptance testing
PO/ customer

acceptance criteria to determine the testing scope for the iteration. Best practices include involving testers early in the planning process, breaking down user stories into testable tasks, and prioritizing testing tasks based on business value.

testers executes the test cases to verify that the software functions as expected. best practices include executing tests in small , manageable batches , documenting these results thoroughly and communicating any issues or blockers promptly.

metrics can be used in each phase of the Agile Testing Life Cycle (ATLC) in detail:

Iteration Planning:

Velocity: Measure the team's capacity to deliver work in each iteration. It helps in planning future iteration goals.

Defect Prediction: Based on historical data, predict the number of defects that might be found in the upc helps in allocating resources and setting priorities.

Test Case Development:

Test Coverage: Track the percentage of code covered by tests. It helps in identifying areas of the code that tested.

Test Case Effectiveness: Measure the effectiveness of test cases by tracking the number of defects found b helps in identifying and improving poorly performing test cases.

Test Execution:

Defect Detection Rate: Monitor the rate at which defects are found during testing. A decreasing defect de improving software quality.

Test Execution Progress: Track the progress of test execution to ensure that testing is on schedule and ide issues.

Regression Testing:

Regression Test Coverage: Measure the percentage of the application covered by regression tests. It help parts of the application are tested after each change.

Regression Test Efficiency: Calculate the ratio of the number of defects found during regression testing to defects found. It helps in assessing the effectiveness of regression testing.

Acceptance Testing:

Acceptance Test Pass Rate: Measure the percentage of acceptance tests that pass successfully. It indicates meets the acceptance criteria defined by the customer.

Customer Satisfaction: Gather feedback from the customer regarding the quality of the software and the provides insights into areas that require improvement.

Iteration Planning:

Metric: Velocity

Team Parameter: Team Capacity

Tracking Sheet: Use a burndown chart to track the progress of work throughout the iteration. This chart sh versus time, helping the team understand if they are on track to complete the planned work.

Test Case Development:

Metric: Test Coverage

Team Parameter: Test Case Development Speed

Tracking Sheet: Use a test coverage matrix to track which parts of the code are covered by test cases. This identify areas that require additional testing.

Test Execution:

Metric: Defect Detection Rate

s and setting realistic

oming iteration. This

t are not adequately

by each test case. This

etection rate may indicate

entify any bottlenecks or

s in ensuring that critical

o the total number of

s whether the software

testing process. It

nows the remaining work

s matrix helps the team

Team Parameter: Test Execution Efficiency

Tracking Sheet: Use a defect log to track the defects identified during testing. Include details such as the s to reproduce, and the status of the defect (open, fixed, closed).

Regression Testing:

Metric: Regression Test Coverage

Team Parameter: Regression Test Efficiency

Tracking Sheet: Use a regression test coverage matrix to track which regression test cases have been exec ensure that critical parts of the application are tested after each change.

Acceptance Testing:

Metric: Acceptance Test Pass Rate

Team Parameter: Customer Satisfaction

Tracking Sheet: Use an acceptance test log to track the results of acceptance testing. Include details such test results, and any feedback from the customer.

end point : /api/a/v1
type : public/private
input:
output:
error:

severity of the defect, steps

uted. This matrix helps

as the acceptance criteria,