

COMP30022

IT Project

- Sprint End
 - Progress Checklist
-

End of Sprint

Sprint Retrospective

- Periodically look at what is and isn't working
- Typically 30 minutes
- Done after every sprint
- Whole team participates:
 - ScrumMaster and Dev Team
- Possibly Product Owner, customers and others (But generally NOT)
- Discuss what to:
 - Start Doing, Stop Doing and Continue Doing

VS.

Sprint Reviews - Showcase

- Team presents what it accomplished during the sprint
- Typically takes the form of a demo of new features or underlying architecture
- Informal
- 2-hour prep time rule
- No slides
- Whole team participates
- Invite the world

Sprint Retrospective

- Progress Summary
 - Overview of Product/Development
 - Review of Completed User Stories
 - Burndown Chart
 - Evaluation of Practices
 - What Went Well?
 - Areas for Improvement
 - Velocity Adjustment
 - Artifact Adjustments
-

Sprint Review

- Presentation of Accomplishments
 - Highlight completed tasks and deliverables
 - Product/Progress Demonstration
 - Showcase the latest version or features
 - Client Discussion
 - Review and align goals/objectives for the next sprint
 - Gather feedback and input for future work
-

Progress Checklist

- **Agile**
 - **Process**
 - **Artefacts**
 - Maintenance
 - Feedback
 - Special Consideration
-



Progress Checklist

- **Agile**
 - **Process**
 - **Artefacts**
 - Maintenance
 - Feedback
 - Special Consideration
-



Adherence to Agile

- Sprint Management
 - Jira & Backlogs
 - Ceremony
 - Planning
 - Retrospective
 - Review
 - Priority Assignment
-



Progress Checklist

- Agile
 - **Process**
 - Artefacts
 - Maintenance
 - Feedback
 - Special Consideration
-



Roles

- Team page on documentation site
 - Cooperation, teamwork & leadership
 - Actions/Tasks performed according to roles
-



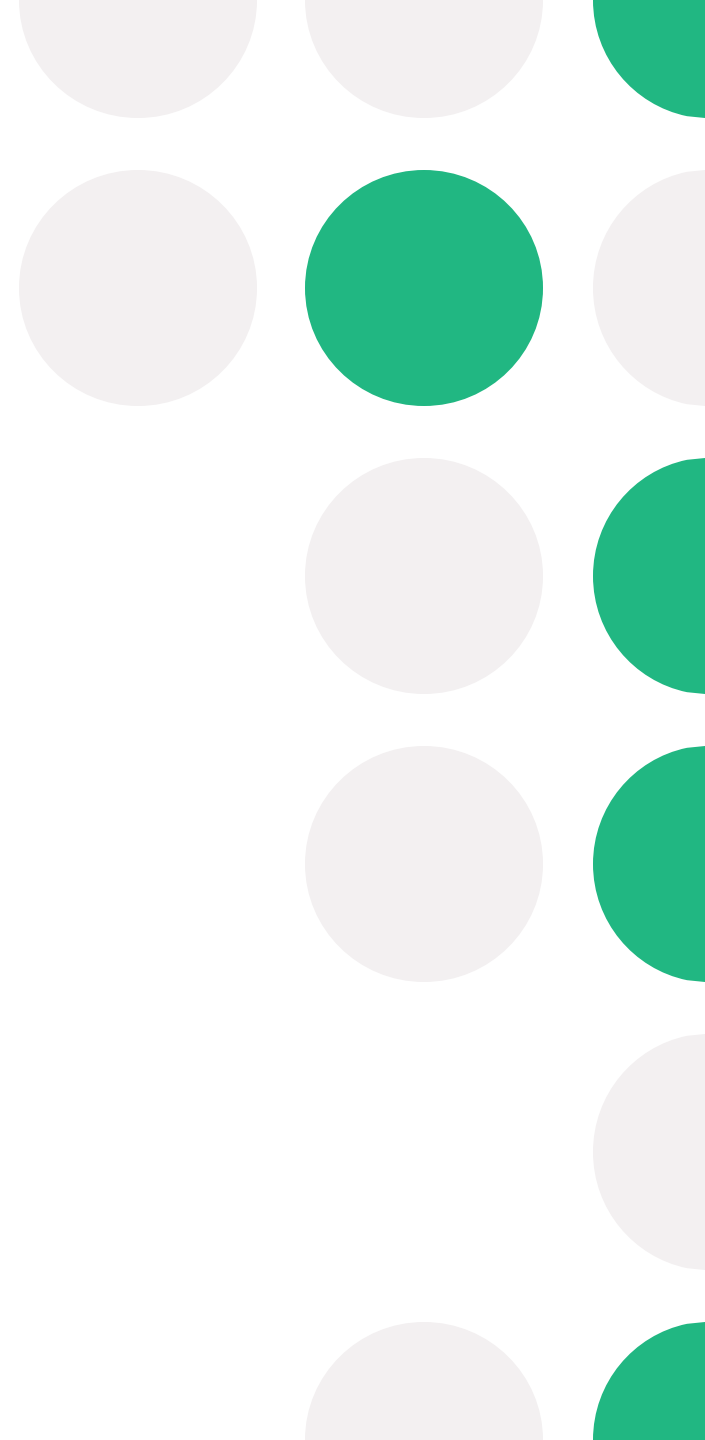
Decision Making

- Documentation (e.g. meeting notes, role assignment, choice of tech stack, UI design choice, etc. ...)
 - Change Log
 - Process
 - Structured approach
 - e.g. research, tradeoff, ...
 - Inclusive decision making
 - e.g. votes, ...
-



Meeting

- Time & Location
 - Attendee list
 - Agenda & Content
 - Outcome -> Action items
 - (Recording)
-



Communication

- Internal
 - Agile ceremonies
 - Regular discussion in your chat channel
 - Pair programming
 - etc. ...
 - External
 - Client emails, meetings, ...
 - Actively engaging
-



Progress Checklist

- Agile
 - Process
 - **Artefacts**
 - Maintenance
 - Feedback
 - Special Consideration
-



Requirements

- Breakdown
 - Functional vs.. non-functional
 - Out of scope
 - User Stories
 - Additional artefacts
 - E.g. store points estimation, priority assignment, velocity estimation, requirement adjustments, ...
-



Frontend Design

- Prototyping
 - Low-fid (e.g. wireframes)
 - High-fid (e.g. Figma)
 - UI Documentation
 - e.g. choice of UI library, theme/color palette, components reuse, ...
 - Thought process behind design choice
-



Architectural Design

- Design & diagrams
 - 4+1 Architecture Model
 - Quality
 - Scalability
 - Maintainability
 - Extendibility
-



Codebase

- Repository structure
 - README file
 - Dev guidelines
 - (refer to lecture slides)
 - Standards & workflow
 - Project setup & config
 - How to run/operate the software
 - Contribution routine
 - e.g. commit conventions, pull request, branching strategies, issue tracking, ...
 - Code review
 - Quality assurance
 - Coding style enforcement
 - e.g. Prettier, ESLint, Husky, ...
-

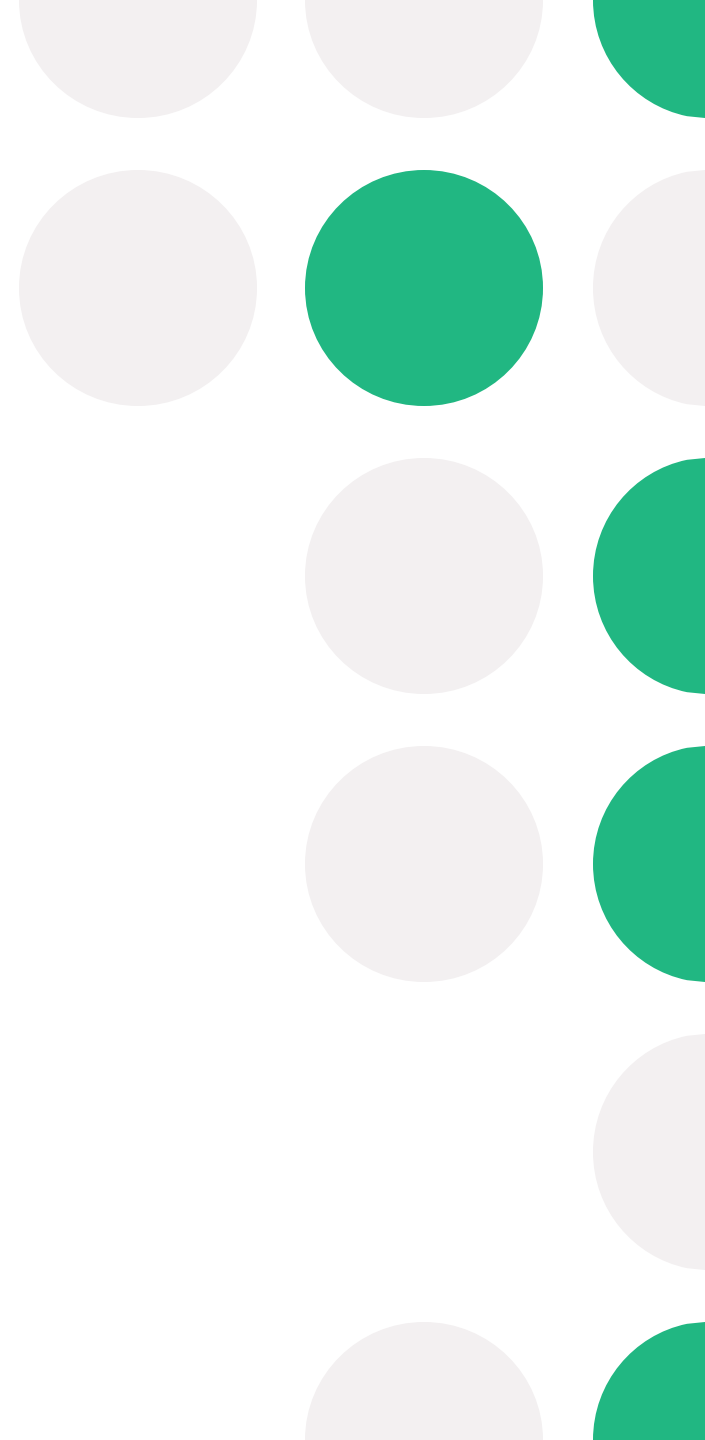
Testing & Deployment

- Plan & research
- Acceptance criteria for user stories



Progress Checklist

- Agile
 - Process
 - Artefacts
 - Maintenance
 - Feedback
 - Special Consideration
-



Thank you :)

