

Agile Planning Guide

Emmanuel BRUNO
Université de Toulon, LIS UMR CNRS 7020
emmanuel.bruno@univ-tln.fr

2026-10-03

Table des matières

1 Agile Planning Guidelines	1
1.1 Project Features Considerations	1
1.2 Scrum Sprint Planning	1
1.2.a Sprint Planning Meeting Structure	1
1.3 Kanban Planning Approach	2
1.3.a Continuous Flow Planning	2
1.3.b Planning Components	2
1.3.c Service Class Examples	2
1.3.d Sample Kanban Board	2
1.4 Extreme Programming (XP) Planning	2
1.4.a Release Planning	2
1.4.b Weekly Planning	2
1.4.c Planning Components	2
1.4.d Testing Priorities	2
1.5 Choosing the Right Approach	2
1.5.a Use Scrum When:	2
1.5.b Use Kanban When:	2
1.5.c Use XP When:	2
1.6 Capacity Planning	2
1.7 Sprint Planning Template	3
1.8 Definition of Ready	3
1.9 Post-Sprint Planning	3

1 Agile Planning Guidelines

This document provides planning guidance for software projects using common agile methodologies.

1.1 Project Features Considerations

- Core functionality
- User interface components
- Data management
- Integration requirements
- Security features
- Performance requirements

1.2 Scrum Sprint Planning

1.2.a Sprint Planning Meeting Structure

1.2.a.a 1. Pre-meeting Preparation (Product Owner)

- Refine and prioritize the product backlog
- Ensure user stories have clear acceptance criteria
- Prepare sprint goal proposal

1.2.a.b 2. Sprint Planning Meeting (2 hours)

1.2.a.b.a Part 1: What will we deliver? (60 minutes)

- Product Owner presents the sprint goal
- Team reviews and discusses high-priority backlog items
- Team asks clarifying questions about requirements
- Team commits to specific backlog items for the sprint

1.2.a.b.b Part 2: How will we deliver it? (60 minutes)

- Team breaks down selected user stories into tasks
- Team estimates effort for each task
- Team identifies dependencies and potential blockers
- Final sprint backlog is confirmed

1.3 Kanban Planning Approach

1.3.a Continuous Flow Planning

- No fixed iterations
- Work items pulled when capacity is available
- WIP limits for each workflow stage
- Regular replenishment meetings (weekly recommended)

1.3.b Planning Components

- Backlog refinement sessions
- Class of Service definitions
- WIP Limits by column
- Service Level Expectations (SLEs)

1.3.c Service Class Examples

- Feature Development (Standard Service)
- Production Issues (Expedited Service)
- Customer Requests (Fixed Delivery Date)
- Technical Debt (Intangible Service)

1.3.d Sample Kanban Board

- Backlog
- Analysis
- Development (WIP: 2)
- Quality Assurance
- User Acceptance
- Done

1.4 Extreme Programming (XP) Planning

1.4.a Release Planning

- Quarterly planning horizon
- User stories grouped into releases
- Velocity tracked in story points

- Regular customer involvement

1.4.b Weekly Planning

- Weekly rather than bi-weekly iterations
- Stand-up planning meetings
- Pair programming assignments
- Test-first development planning

1.4.c Planning Components

- Story writing sessions with customer
- Technical spike planning
- Pair rotation schedule
- Continuous integration checks

1.4.d Testing Priorities

- Unit test coverage
- Integration testing
- Performance testing
- Security validation
- Compliance checks

1.5 Choosing the Right Approach

1.5.a Use Scrum When:

- Clear project scope exists
- Regular deliverables are needed
- Team prefers structured sprints

1.5.b Use Kanban When:

- Work is interrupt-driven
- Continuous flow is preferred
- Support work is prominent

1.5.c Use XP When:

- Quality is paramount
- Frequent releases needed
- Strong technical focus required

1.6 Capacity Planning

- Each sprint is 2 weeks (10 working days)
- Team capacity calculation:
 - ▶ 6 hours of productive development time per day per developer
 - ▶ 3 developers × 6 hours × 10 days = 180 hours total capacity

- Reserve 20% for unexpected issues (36 hours)
- Actual sprint capacity: 144 hours

1.7 Sprint Planning Template

See the Sprint Planning Template for the standard planning document.

1.8 Definition of Ready

A user story is ready for sprint planning when:

- It has a clear description following the « As a... I want... So that... » format - It has defined acceptance criteria - It's been estimated by the team - All external dependencies have been resolved - It's small enough to be completed in one sprint

1.9 Post-Sprint Planning

- Update sprint board in JIRA/Trello
- Share sprint goals and commitments with stakeholders
- Schedule any required technical discussions for complex items
- Begin the sprint with a kickoff meeting