Module 3: Boards & Work Item Tracking

1. Agile vs Scrum vs Kanban in Azure Boards

Azure Boards supports **multiple work processes** that align with industry-standard Agile methodologies.

1.1 Agile Process in Azure Boards

The **Agile process** is Microsoft's template best suited for **teams practicing adaptive planning** and fast delivery.

Characteristics

- Lightweight
- Flexible
- Supports changing requirements
- Work-items: *Epics → Features → User Stories → Tasks*
- Simple state transitions
- Prioritization via drag-and-drop in backlogs

Best For:

Product-based teams, startups, iterative development.

1.2 Scrum Process in Azure Boards

The **Scrum process** aligns to the official Scrum Guide.

Characteristics

- Work-items: *Epics → Features → Product Backlog Items (PBIs) → Tasks*
- Events supported:
    - Sprint Planning
    - Daily Standups
    - Sprint Review
    - Sprint Retrospective
- Commitments tracked using Sprints
- Velocity tracking & forecasting

Best For:

Teams following formal Scrum ceremonies with sprint cycles.

1.3 Kanban Process in Azure Boards

Designed for **continuous flow-based teams**.

Characteristics

- Work-items: *Epics → Features → User Stories → Tasks (Same as Agile)*
- Continuous delivery model
- No timeboxed sprints
- WIP (Work-in-Progress) limits
- Lead time & cycle time optimization

Best For:

Support teams, production support, operations, DevOps, and maintenance.

1.4 Agile vs Scrum vs Kanban (Comparison Table)

Dimension	**Agile**	**Scrum**	**Kanban**
Approach	Iterative	Structured framework	Continuous flow
Timeboxing	Optional	Mandatory (Sprints)	None
Work Items	User Stories	PBIs	User Stories
Team Roles	Flexible	PO, SM, Dev Team	No mandatory roles
Boards	Kanban/Agile	Scrum Board	Kanban Board
Best For	Product development	Feature teams	Ops & support teams

---

2. Work Item Hierarchy in Azure Boards

Azure Boards uses a **hierarchical structure** for work tracking.

Work Item Hierarchy Diagram

EPIC

└── FEATURE

└── USER STORY (or PBI)

└── TASK

2.1 Epics

Large business goals or initiatives lasting multiple months.

Examples:

- "Migrate application to Azure"
- "Modernize authentication architecture"

2.2 Features

Smaller modules contributing to an Epic (2–6 weeks).

Examples:

- "Implement Azure AD OAuth2"
- "Create CI/CD pipeline for API"

2.3 User Stories / PBIs

Functional requirements written from the user's perspective.

Example Template:

As a <user>
I want <capability>
So that <business value>

2.4 Tasks

Technical breakdown of a User Story.

Examples:

- Create API endpoints
- Write unit tests
- Configure pipeline tasks

---

3. Creating Custom Fields in Azure Boards

Azure Boards allows customizing work-item fields to fit organizational needs.

Steps: Create a Custom Field

1. Go to **Organization Settings → Process**
2. Select the process (Agile/Scrum/Kanban)
3. Choose Work Item Type (e.g., User Story)
4. Click **New Field**
5. Choose:
    - Field Name
    - Data Type (Text, Integer, Picklist, Boolean)
6. Add it to:
    - Layout section (Form)
    - Rules (if needed)
7. Save

Common Custom Fields

Field Name	Type	Purpose
Business Impact	Picklist	Prioritization
Risk Level	Picklist	Risk management
Component	Text	Identify module/service
Release Version	Text	Tracking deployments
Acceptance Criteria Status	Boolean	QA check

---

4. Creating Dashboards in Azure Boards

Dashboards visualize project progress.

Steps to Create Dashboard

1. Go to **Dashboards**
2. Click **New Dashboard**

3. Enter:
    - Name
    - Team
4. Select **Widgets** such as:
    - Burndown
    - Sprint Overview
    - Velocity
    - Work Item Query
    - Pipelines Status
    - Test Results
5. Customize layout
6. Save dashboard

Recommended Dashboard Widgets

For Scrum Teams:

- Sprint Burndown
- Velocity Chart
- Sprint Goal Tracker

For Management:

- Work Item by State
- Cumulative Flow Diagram
- Team Velocity Trend

For DevOps/SRE Teams:

- Build Health
- Release Pipeline Deployment Status
- Bug Trends

---

5. Sprint Planning in Azure Boards

Scrum teams use sprints to plan and deliver work in fixed timeboxes.

5.1 Sprint Lifecycle

1. Sprint Planning
2. Daily Scrum
3. Development Work
4. Backlog Refinement
5. Review
6. Retrospective

5.2 Sprint Planning Steps

1. Navigate to **Boards → Sprints**
2. Select **Sprint 1** (or create new sprint)
3. Set:
    - Sprint dates
    - Team capacity
4. Drag PBIs/User Stories from backlog into sprint
5. Break stories into **Tasks**
6. Assign tasks to team members
7. Ensure sprint goal is clear

5.3 Burndown Charts

A burndown chart shows **remaining work vs time**.

Key Uses

- Track sprint progress
- Identify delays
- Measure team predictability
- Spot issues early

Indicators

- **Healthy curve**: steady downward progress
- **Flat line**: blocked work
- **Rising line**: scope added mid-sprint

---

6. LAB – Create Boards for a Sample Project

This lab is ready for real training sessions.

LAB 1: Create an Agile Project

1. Go to **https://dev.azure.com**
2. Create new project:
    - Name: EmployeeMgmtSystem
    - Process: Agile
    - Version Control: Git

LAB 2: Create Work Item Hierarchy

Step 1: Create an Epic

- Epic: **User Management Module**

Step 2: Add Features

- Feature: Login & Authentication
- Feature: User Profile Management

Step 3: Add User Stories

For Feature: Login & Authentication

- User Story: As a user, I want to log in securely using email/password.
- User Story: As an admin, I want to reset user passwords.

Step 4: Add Tasks

- Create login page
- Build API endpoint
- Store creds in Key Vault
- Write unit tests
- Configure CI pipeline

LAB 3: Customize Fields

Create a custom picklist field:

1. Organization Settings → Process
2. Agile → User Story

3. New Field → Name: **Business Impact**
4. Add picklist values:
    - High
    - Medium
    - Low
5. Add to work item form

LAB 4: Create a Dashboard

1. Go to Dashboards
2. Click **New Dashboard**
3. Add widgets:
    - Sprint Burndown
    - Work Items by State
    - Sprint Overview
    - Build/Release Pipeline status

LAB 5: Sprint Planning

1. Go to **Boards → Sprints**
2. Choose Sprint 1
3. Set dates
4. Set capacity (hours/day)
5. Drag user stories into sprint
6. Break them into tasks
7. Validate burndown visualization

LAB 6: Review Team Burndown

1. During sprint execution
2. Go to **Analytics → Burndown Chart**
3. Validate:
    - Progress rate
    - Team load

- Remaining work
- Scope changes