Behaviour Driven Development (BDD) Workshop Training

Duration: **1 Day**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

Overview

The **Behaviour Driven Development (BDD) Workshop** provides a practical introduction to using BDD to create shared understanding across business, development and testing roles before software is built.

Participants work through the journey from business goals and user stories to acceptance criteria, concrete examples, Gherkin scenarios and executable specifications.

The workshop treats BDD primarily as a **collaboration and communication practice**, not simply as a testing technique. Learners explore story mapping, specification by example, Gherkin syntax, Cucumber implementation, living documentation and traceability through practical exercises.

The workshop is not intended to provide end-to-end requirements management training. Its focus is on creating clear and testable specifications that can be linked to MVPs and delivery outcomes.

Artificial Intelligence (AI) is also used during the course to support structured prompting, scenario generation and documentation review. AI is positioned as an assistive tool for improving quality, clarity and consistency, while human judgement remains central to validation and decision-making.

Prerequisites

No previous BDD experience is required.

The course is suitable for anyone interested in improving:

- Requirements quality
- Team collaboration
- Software quality
- Agile delivery

Familiarity with user stories, testing, software development or Agile ways of working is useful but not essential.

Participants who wish to complete the optional coding activities should have:

- Basic familiarity with JavaScript, or
- Confidence following guided technical demonstrations

Behaviour Driven Development (BDD) Workshop Training

Duration: **1 Day**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

Who Should Attend

The workshop is designed for professionals involved in defining, developing, testing or delivering software and digital products.

It is particularly suitable for:

- Business Analysts
- Product Owners
- Developers
- Testers and Quality Engineers
- Scrum Masters
- Delivery Leads
- Agile teams looking to improve requirements, collaboration and test coverage

What You Will Learn

By the end of the workshop, participants will be able to:

- Explain how BDD improves collaboration and shared understanding across delivery teams.
- Write stronger user stories using quality checks such as SMART and INVEST.
- Use story mapping to structure user journeys and prioritise delivery work.
- Create acceptance criteria and examples that support testable requirements.
- Write clear Gherkin scenarios using **Given, When and Then**.
- Compare imperative and declarative scenario-writing styles.
- Use AI-assisted prompts to support scenario creation, refinement and documentation.
- Understand the basic path from BDD scenarios to executable specifications.

Outline

1. Behaviour Driven Development Foundations

Explore the purpose of BDD and how it strengthens shared understanding across delivery teams.

- Purpose of BDD
- Common collaboration gaps in software delivery
- Building shared understanding
- Relationships between requirements, examples, tests and working software
- BDD as a communication practice as well as a technical practice

2. Test Coverage and Quality

Examine how BDD can improve software quality and test confidence.

- Role of test data in improving coverage
- Risks associated with test-after approaches
- Benefits of test-before thinking
- Using examples to reveal assumptions

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Identifying edge cases
- Improving confidence in delivered software through BDD

3. User Stories

Develop stronger user stories and evaluate their quality.

- Writing effective user stories
- Linking user stories to business goals
- Using INVEST to assess story quality
- Story sizing
- Story refinement
- Identifying weak or incomplete stories

4. Story Mapping

Use story mapping to visualise user journeys and structure delivery work.

- Building a story map from a user journey
- Identifying activities and tasks
- Creating supporting stories
- Prioritising stories across a workflow
- Using story mapping for planning and alignment
- Applying AI-assisted prompts to improve story maps

5. Acceptance Criteria

Explore how acceptance criteria support clear and testable requirements.

- Purpose of acceptance criteria
- Using SMART to improve clarity and testability
- Turning acceptance criteria into concrete examples
- Identifying gaps and ambiguity
- Discovering missing business rules
- Connecting acceptance criteria to BDD scenarios

6. Shared Understanding and Specification by Example

Explore collaborative approaches that align business, development and testing roles.

The Three Amigos

- Bringing together business, development and testing perspectives
- Facilitating conversations before coding begins
- Challenging assumptions with concrete examples

Specification by Example

- Explaining requirements through examples
- Reducing ambiguity
- Making business rules more visible
- Using AI-assisted prompts to review and strengthen examples

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

7. Living Documentation and Traceability

Understand how BDD artefacts can remain useful throughout the lifecycle of a product.

- Purpose of living documentation
- Using feature files to document expected behaviour
- Linking scenarios to user stories and business goals
- Keeping documentation aligned with working software
- Avoiding documentation that has no clear audience or value

8. Gherkin Syntax and Structure

Learn the basic structure used to write clear and consistent BDD scenarios.

- Feature structure
- Scenario structure
- Given, When and Then syntax
- Effective use of And and But
- Background sections
- Data tables
- Examples tables
- Rules of thumb for clear Gherkin writing
- Using AI-assisted prompts to generate and refine scenarios

9. Scenario Writing Styles

Compare different approaches to expressing behaviour.

Imperative Style

- Step-by-step descriptions
- More detailed procedural scenarios

Declarative Style

- Focus on expected behaviour and business intent
- More readable and maintainable scenarios

The module also considers:

- Choosing the right level of detail
- Improving readability
- Improving maintainability
- Rewriting scenarios to make their intent clearer

10. Cucumber Implementation

Explore how Gherkin scenarios can become executable specifications.

- Turning Gherkin into executable specifications
- Initialising Cucumber
- Creating step definitions
- Reviewing generated step definitions

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Connecting feature files with implementation code
- Demonstrating the BDD lifecycle from feature file to passing test
- Using AI assistance during implementation
- Maintaining human review of AI-generated outputs

Hands-On Learning

The workshop is strongly practice-based.

Participants work with activities including:

- User story review
- Story mapping
- Acceptance criteria development
- Specification by example
- Three Amigos discussions
- Gherkin scenario writing
- Scenario rewriting
- Cucumber implementation
- AI-assisted prompting
- Living documentation and traceability exercises

The aim is not simply to understand BDD terminology, but to develop a practical way of creating clearer, testable specifications that improve collaboration and shared understanding across delivery teams.