Introduction to Software Development Projects

CS2113 – Software Development Project

Masoud Hamad

School of Computing Communication and Media Studies

Academic Year 2025

Outline

What is a Software Development Project?

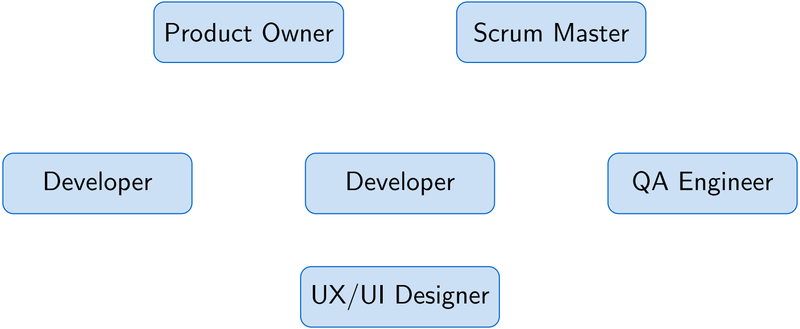
- Software projects emerge when there's **identified need** and **funding**
- Projects vary from months to years
- Involve **stakeholders** – all people affected by the software
  - End users
  - Funding customers
  - Development team

Key Insight

Most development problems are **non-technical** rather than technical!

Development Team Structure

Self-Organized Team (<10 people)



Product Owner

Scrum Master

Developer

Developer

QA Engineer

UX/UI Designer

The Agile Manifesto

We value:

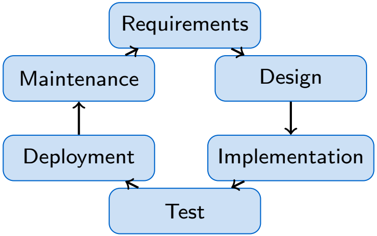
1. **Individuals and interactions** over processes and tools
2. **Working software** over comprehensive documentation
3. **Customer collaboration** over contract negotiation
4. **Responding to change** over following a plan

Critical Value

Welcome change – requirements frequently shift during development!

SDLC Phases

1. **Requirements** – Collect stakeholder needs
2. **Design** – Analyze and identify solutions
3. **Implementation** – Write code
4. **Test** – Find bugs
5. **Deployment** – Release to production
6. **Maintenance** – Fix issues, monitor



Waterfall vs Agile

Waterfall Model

- Sequential phases
- All requirements upfront
- No changes mid-project
- Problematic!

Agile Model

- Iterative approach
- 1–2 week sprints
- Continuous feedback
- Flexible!

Scrum Framework

Most Popular Agile Framework

87% of teams use Scrum (2022 State of Agile Report)

Scrum defines:

- **Roles:** Scrum Master, Product Owner, Developers
- **Events:** Sprint, Sprint Planning, Daily Scrum, Review, Retrospective
- **Artifacts:** Product Backlog, Sprint Backlog, Increment

Types of Requirements

Functional Requirements

User-visible features

- “User should register”
- “User should see posts”

Non-Functional Requirements

Quality constraints (invisible)

- “Passwords as Bcrypt”
- “Load time ¡ 1 second”

User Story Format

“As a **[user persona]**,
I want **[action]**
so that **[goal]**.”

Examples

- “As a **content creator**, I want to **create blogs** so I can **write posts for readers**.”
- “As a **blog reader**, I want to **browse posts** so I can **find interesting content**.”

INVEST Criteria

I	**Independent** – Can be developed alone
N	**Negotiable** – Open to discussion
V	**Valuable** – Provides user benefit
E	**Estimable** – Team can estimate effort
S	**Small** – Fits in one sprint
T	**Testable** – Can verify completion

Common User Story Problems

Too Large (violates "Small")

"As content creator, I want to register with username, password, profile picture, and description"

Better – Split into multiple stories

- Register with username/password
- Add profile picture
- Add profile description

Common User Story Problems (cont.)

Written from Developer Perspective

“As developer, I want to add database index to optimize loading”

Better – User-focused

“As blog reader, I want blog posts to load quickly”

Key Takeaways

1. Software projects need **identified need** + **funding**
2. **Agile** welcomes change; **Waterfall** resists it
3. **Scrum** is the most popular agile framework
4. Requirements are **Functional** or **Non-Functional**
5. User stories follow: “As a..., I want..., so that...”
6. Good user stories follow **INVEST** criteria

Questions?

Questions?