

SOFTWARE ENGINEERING

(Lecture-1)

USAMA MUSHARAF

LECTURER (Department of Computer Science)

FAST-NUCES PESHAWAR

CONTENTS



Introduction to
Software
Engineering



Importance of
Software
Engineering

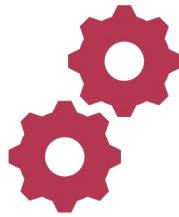


Agile Software
development

WHAT IS SOFTWARE ENGINEERING?



Systematic
approach for
developing software



Methods and
techniques to
develop and
maintain quality
software to solve
problems.



Study of the principles and
methodologies for developing and
maintaining software systems.

WHAT IS SOFTWARE?

According to the IEEE

Software is:

“Computer programs, procedures, and possibly associated documentation and data pertaining to the operation of a computer system”.

THE ROLE OF SOFTWARE ENGINEERING

A bridge from customer needs to programming implementation



First law of software engineering

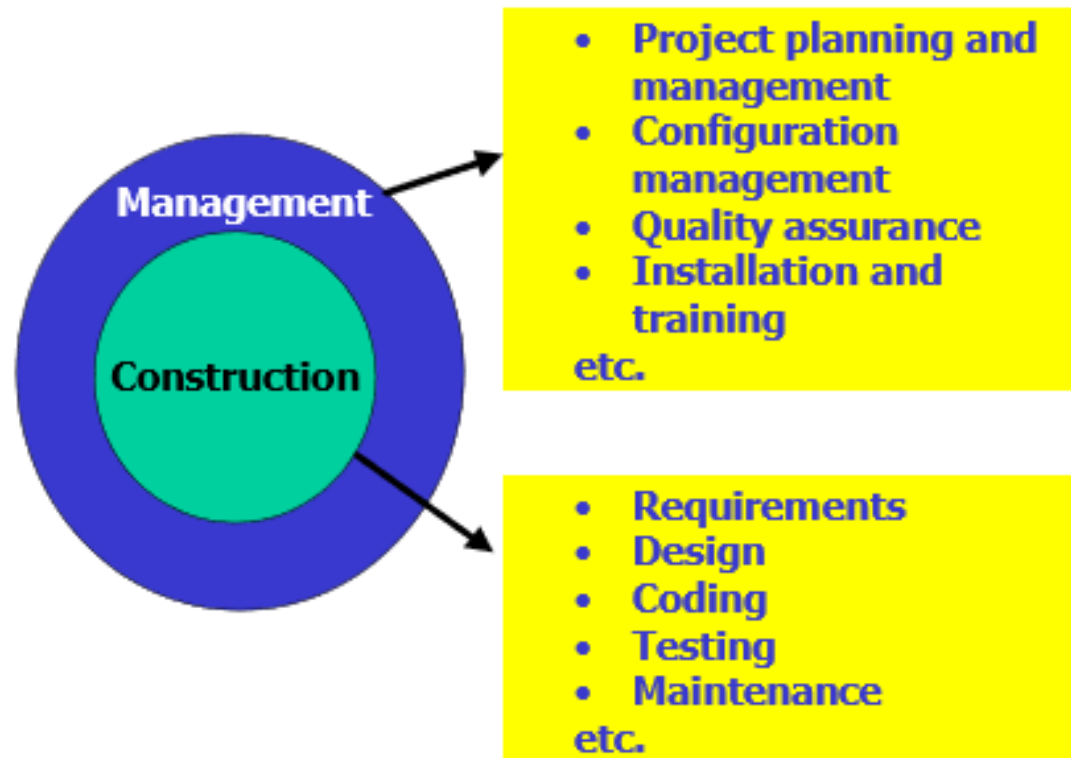
Software engineer is willing to learn the problem domain
(problem cannot be solved without understanding it first)

SOME IMPORTANT SOFTWARE ENGINEERING RELATED ACTIVITIES

- Project Management
- Requirement Engineering
- Software Design
- Coding
- Testing

- Software Quality Assurance
- Software Configuration Management
- Software Deployment

SOFTWARE DEVELOPMENT



SOFTWARE DEVELOPMENT





AGILE DEVELOPMENT



WHAT IS “AGILE”?

- Agile is a philosophy or a way of thinking guided by some **values and principles**.

WHAT IS “AGILITY”?

- Effective (rapid and adaptive) response to change
- Effective communication among all stakeholders

Yielding ...

- Rapid, incremental delivery of software

AGILE VALUES

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

PRINCIPLES OF AGILE METHODS

Principle	Description
Customer involvement	Customers should be closely involved throughout the development process. Their role is provide and prioritise new system requirements and to evaluate the iterations of the system.
Incremental delivery	The software is developed in increments with the customer specifying the requirements to be included in each increment.
People not process	The skills of the development team should be recognised and exploited. Team members should be left to develop their own ways of working without prescriptive processes.
Embrace change	Expect the system requirements to change, so design the system to accommodate these changes.
Maintain simplicity	Focus on simplicity in both the software being developed and in the development process. Wherever possible, actively work to eliminate complexity from the system.

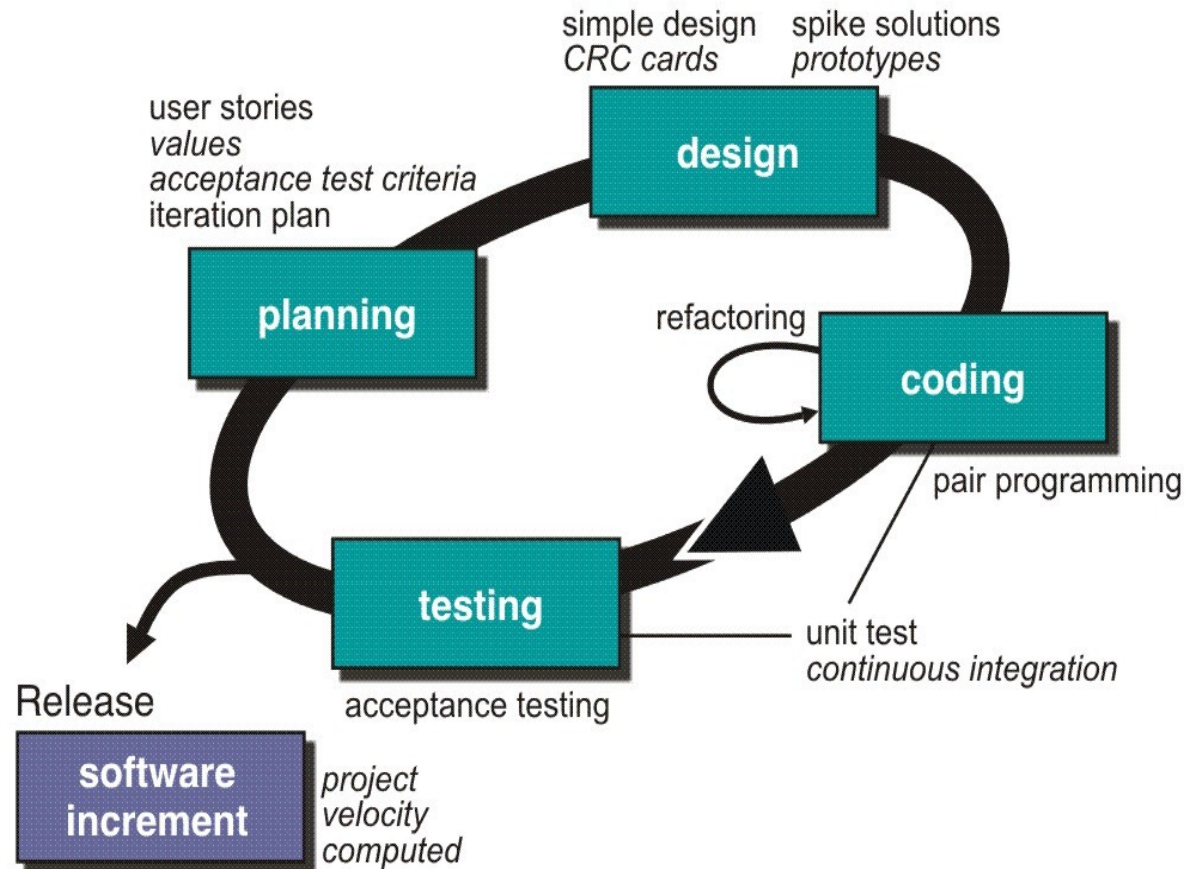
AGILE PROCESS MODELS

- **Extreme Programming (XP)**
- **Scrum**
- Adaptive Software Development
- Dynamic System Development Method (DSDM)
- Crystal
- Feature Driven Development
- Agile Modeling (AM)

EXTREME PROGRAMMING (XP)

- Perhaps one of the well-known and most widely used agile method.
- Extreme Programming (XP) takes an 'extreme' approach to iterative development.
 - New versions may be built several times per day;
 - Increments are delivered to customers every 2 weeks;
 - All tests must be run for every build and the build is only accepted if tests run successfully.

EXTREME PROGRAMMING (XP)



EXTREME PROGRAMMING (XP)

XP Planning

- Begins with the creation of **user stories**
- Agile team assesses each story and assigns a **cost**
- Stories are grouped to for a **deliverable increment**
- A **commitment** is made on delivery date

REQUIREMENTS SCENARIOS

- In XP, user requirements are expressed as scenarios or user stories.
- These are written on cards and the development team break them down into implementation tasks. These tasks are the basis of schedule and cost estimates.
- The customer chooses the stories for inclusion in the next release based on their priorities and the schedule estimates.

STORY CARD FOR DOCUMENT DOWNLOADING

Downloading and printing an article

First, you select the article that you want from a displayed list. You then have to tell the system how you will pay for it - this can either be through a subscription, through a company account or by credit card.

After this, you get a copyright form from the system to fill in and, when you have submitted this, the article you want is downloaded onto your computer.

You then choose a printer and a copy of the article is printed. You tell the system if printing has been successful.

If the article is a print-only article, you can't keep the PDF version so it is automatically deleted from your computer.

EXTREME PROGRAMMING (XP)

XP Design

- Follows the **KIS** (keep it simple) principle
- Encourage the use of **CRC** (class-responsibility-cards) cards
- For difficult design problems, suggests the creation of **spike solutions** — a design prototype
- Encourages **refactoring** — an iterative refinement of the internal program design

EXTREME PROGRAMMING (XP)

CRC Cards:

Class-responsibility-collaboration (CRC) cards are a tool used in the design of object-oriented software.

EXTREME PROGRAMMING (XP)

CRC Cards:

The card is partitioned into three areas:

- I. On top of the card, the class name
- II. On the left, the responsibilities of the class
- III. On the right, collaborators (other classes) with which this class interacts to fulfill its responsibilities.

EXTREME PROGRAMMING (XP)

CRC Cards:

Class Name	
Responsibilities	Collaborators

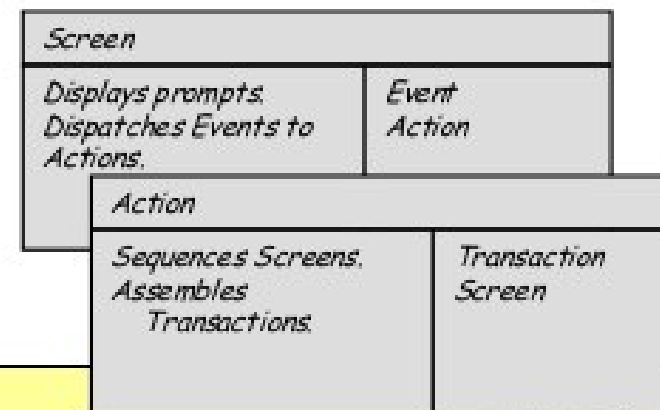
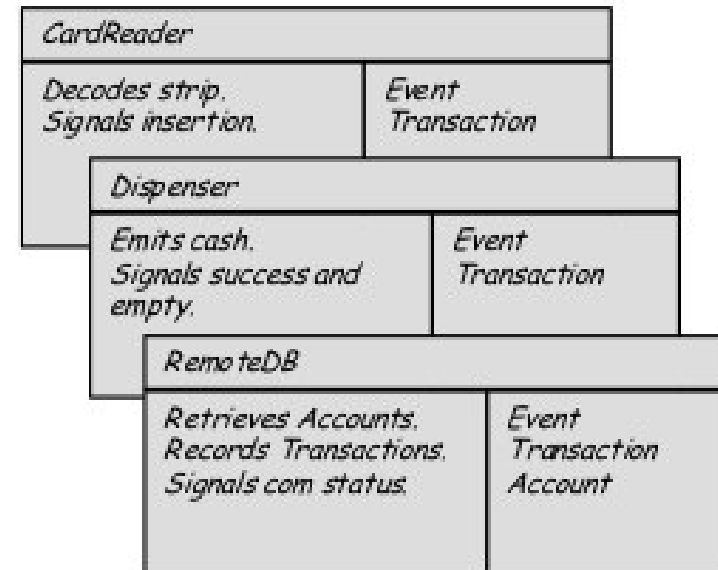
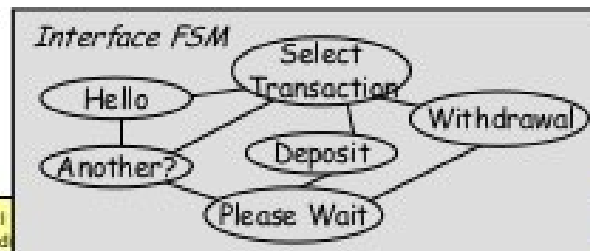
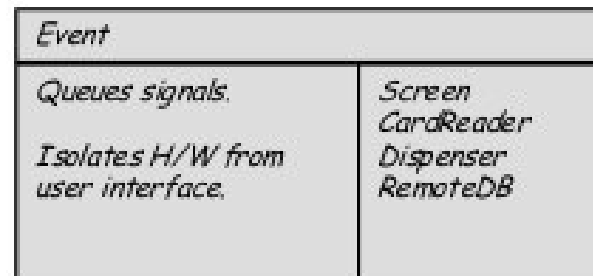
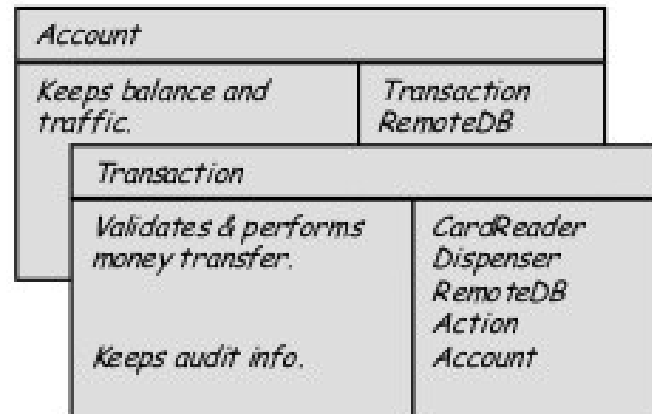
EXTREME PROGRAMMING (XP)

CRC Cards:

Student	
Student number Name Address Phone number Enroll in a seminar Drop a seminar Request transcripts	Seminar

CRC example: ATM software

[Beck, Cunningham]



EXTREME PROGRAMMING (XP)

XP Coding

- Recommends the **construction of a unit test** for a store *before* coding commences
- Encourages **pair programming**

XP Testing

- All **unit tests** are executed daily
- **Acceptance tests** are defined by the customer and executed to assess customer visible functionality

TESTING IN XP

- Test-first development.
- Incremental test development from scenarios.
- User involvement in test development and validation.

TASK CARDS FOR DOCUMENT DOWNLOADING

Task 1: Implement principal workflow

Task 2: Implement article catalog and selection

Task 3: Implement payment collection

Payment may be made in 3 different ways. The user selects which way they wish to pay. If the user has a library subscription, then they can input the subscriber key which should be checked by the system. Alternatively, they can input an organisational account number. If this is valid, a debit of the cost of the article is posted to this account. Finally, they may input a 16 digit credit card number and expiry date. This should be checked for validity and, if valid a debit is posted to that credit card account.

TEST CASE DESCRIPTION

Test 4: Test credit card validity

Input:

A string representing the credit card number and two integers representing the month and year when the card expires

Tests:

Check that all bytes in the string are digits

Check that the month lies between 1 and 12 and the year is greater than or equal to the current year .

Using the first 4 digits of the credit card number , check that the card issuer is valid by looking up the card issuer table. Check credit card validity by submitting the card number and expiry date information to the card issuer

Output:

OK or error message indicating that the card is invalid

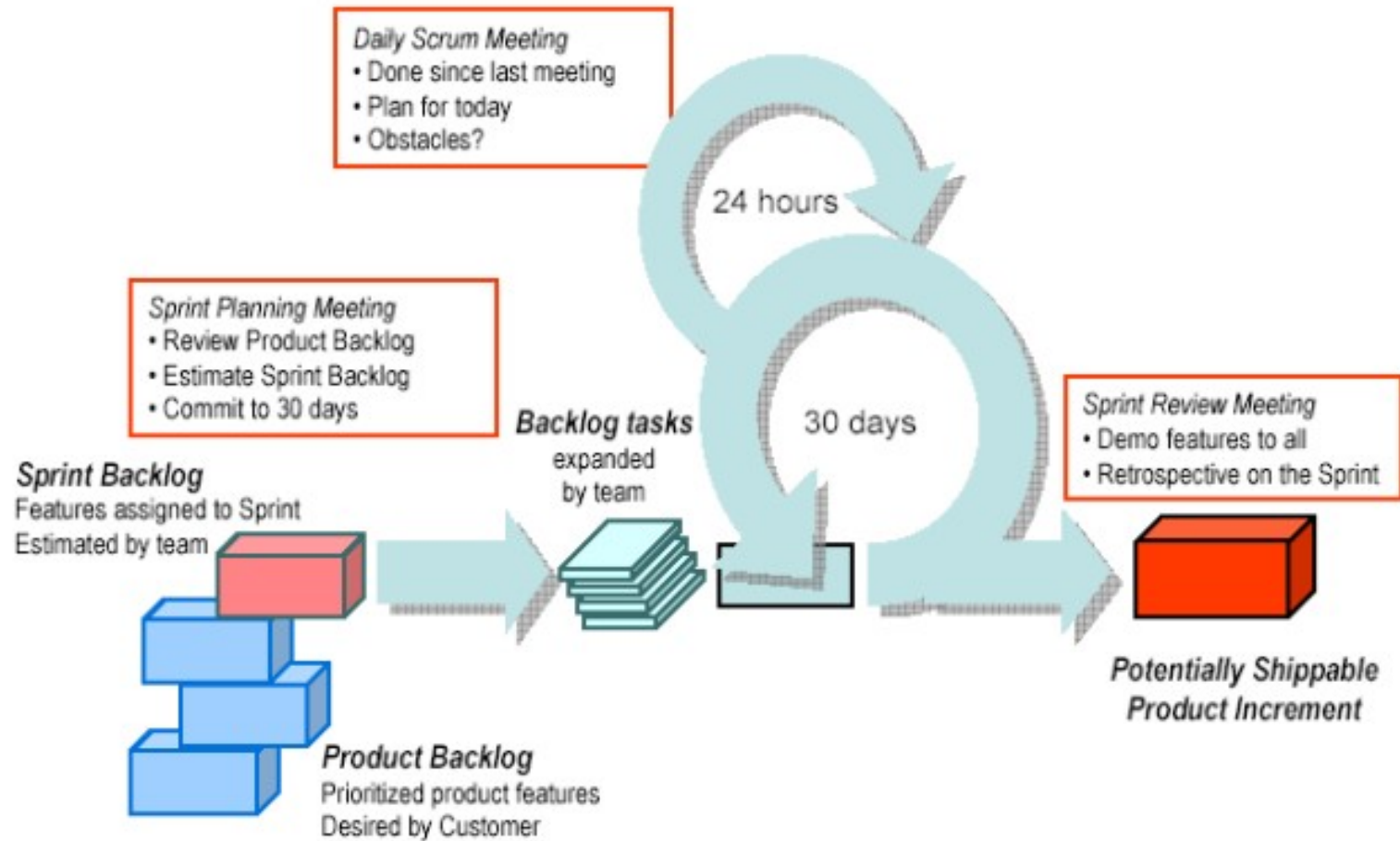
SIGNIFICANCE OF TEST-FIRST DEVELOPMENT

- Writing tests before code clarifies the requirements to be implemented.
- Tests are written as programs rather than data so that they can be executed automatically. The test includes a check that it has executed correctly.
- All previous and new tests are automatically run when new functionality is added. Thus checking that the new functionality has not introduced errors.

SCRUM

- Scrum is an Agile framework for completing complex projects.
- Scrum originally was formalized for software development projects, but it works well for any complex, innovative scope of work.
- Scrum is a team-based approach, to iteratively, incrementally develop systems and products.
- when requirements are rapidly changing .

HOW DOES SCRUM WORK?



How Scrum work

USER STORIES

User Story
capture 3
important items

- Who
- What
- Why

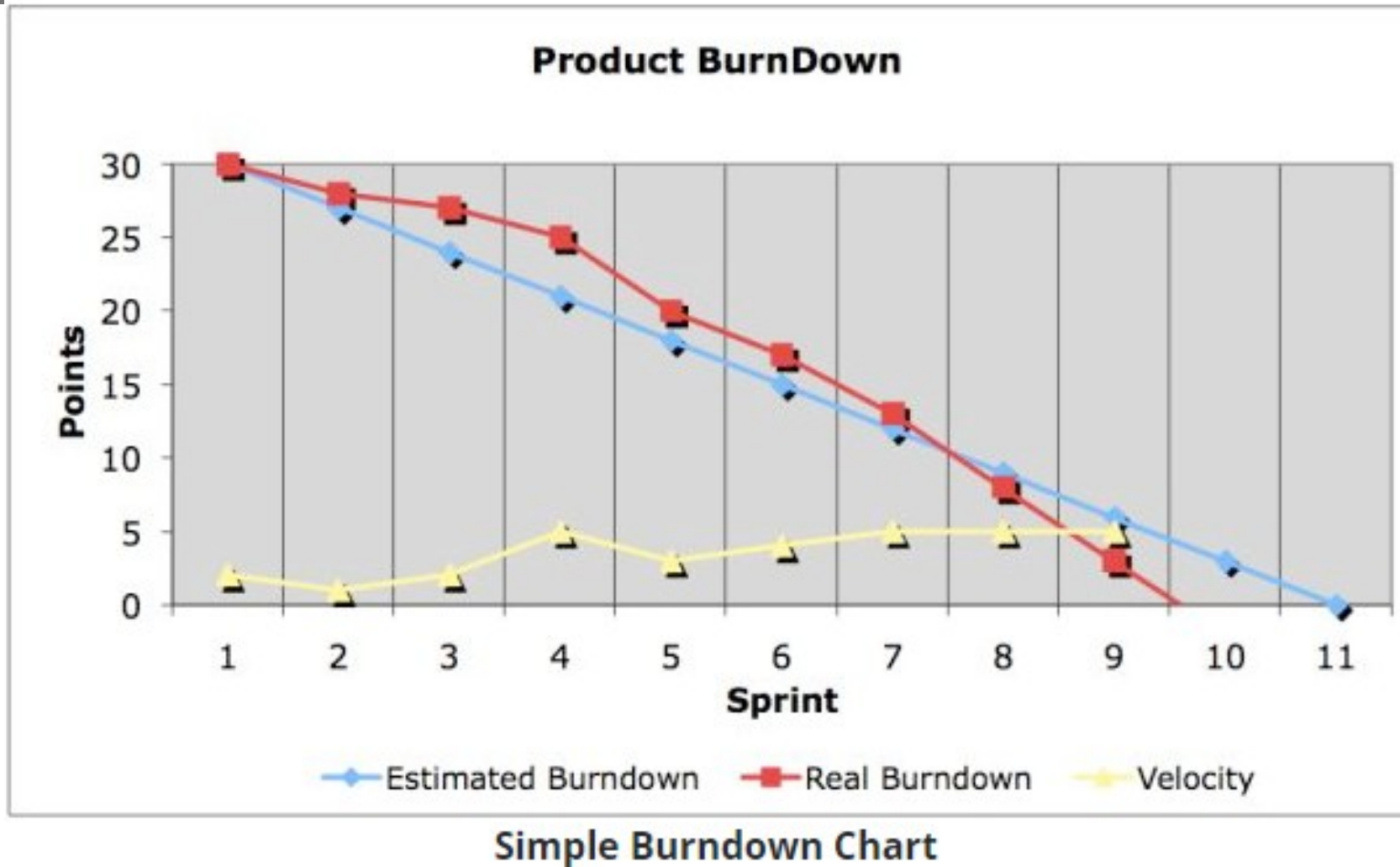
User Story Format

As a (user or type of user)

I want a (some goal or what)

So that (I can achieve some value or why)

BURNDOWN CHART



DIFFERENCE BETWEEN XP AND SCRUM

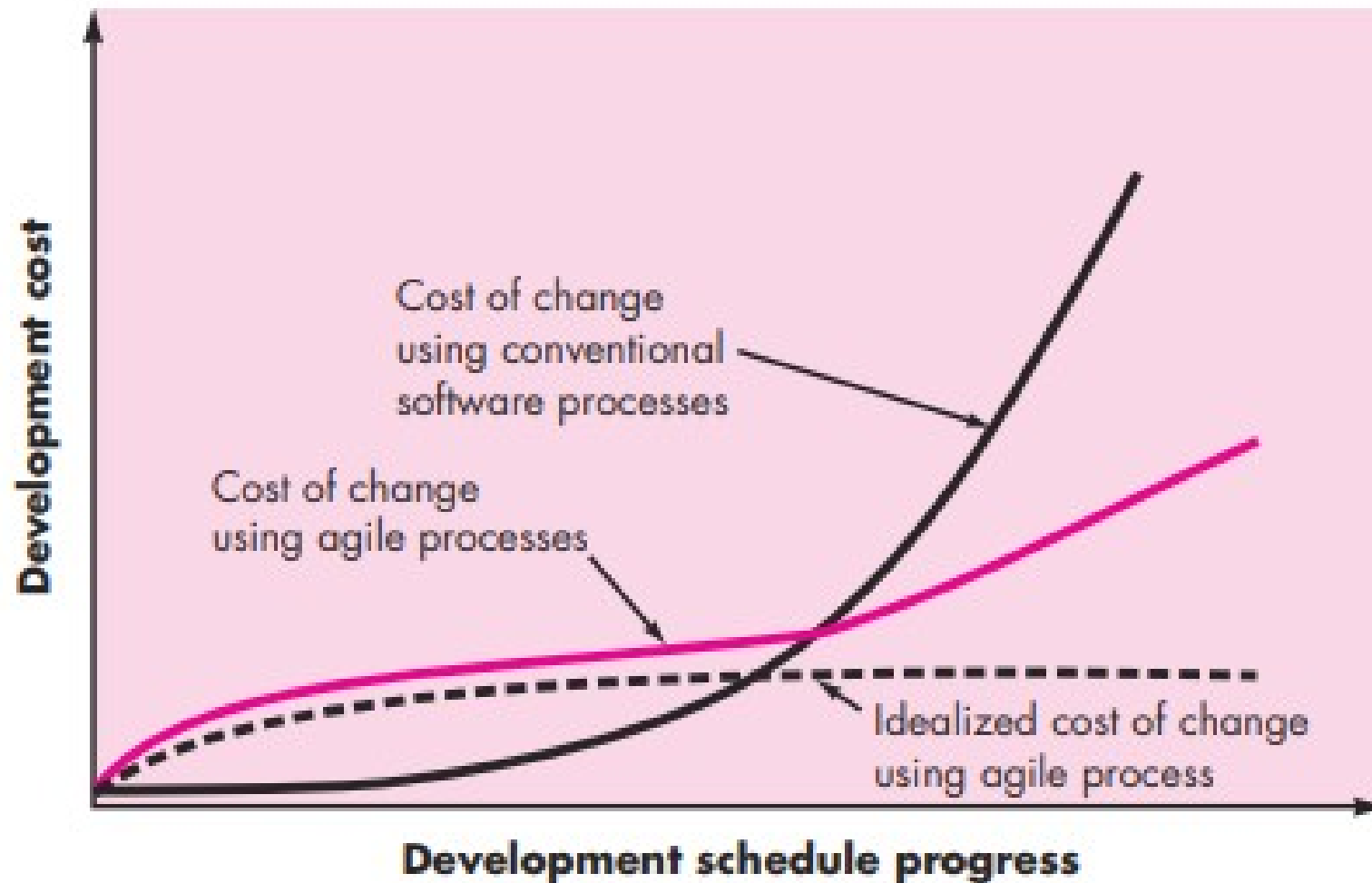
XP Iterations (1-2 Weeks)

Scrum Iterations (3-4Weeks)

Scrum teams (do not allow changes into their sprints).

XP team (allow changes)

COST OF CHANGE IN AGILE





HAVE A GOOD DAY!