COURSE NAME : 19CST201-Agile Software Engineering

III YEAR/ V SEMESTER

Topic: Agile Development Process

Ms.G.Swathi
Assistant Professor
Department of Computer Science and Engineering

UNIT-II AGILE DEVELOPMENT

Fundamentals of Agile: Introduction and background - Agile Manifesto and Principles - Agile vs. Traditional software development – Extreme Programming – Agile Process Models – kanban – Agile Tool(JIRA, GITHUB and Jenkins).

AGILE METHODOLOGY

- Process for developing software
- Agile means
  - Ability to move quickly
  - Responding swiftly to change

What is Agile ?

NON AGILE METHOD

- A project can take several months or years to complete and the customer may not get to see the end product until the completion of the project
- Allocate extensive periods of time for requirements gathering ,design ,development, testing and user acceptance testing before finally deploying the project

AGILE METHOD

- It have **Sprints or iteration** which are shorter in duration (which can vary from 2 weeks to 2 months) during which pre determined features are developed and delivered

SCRUM

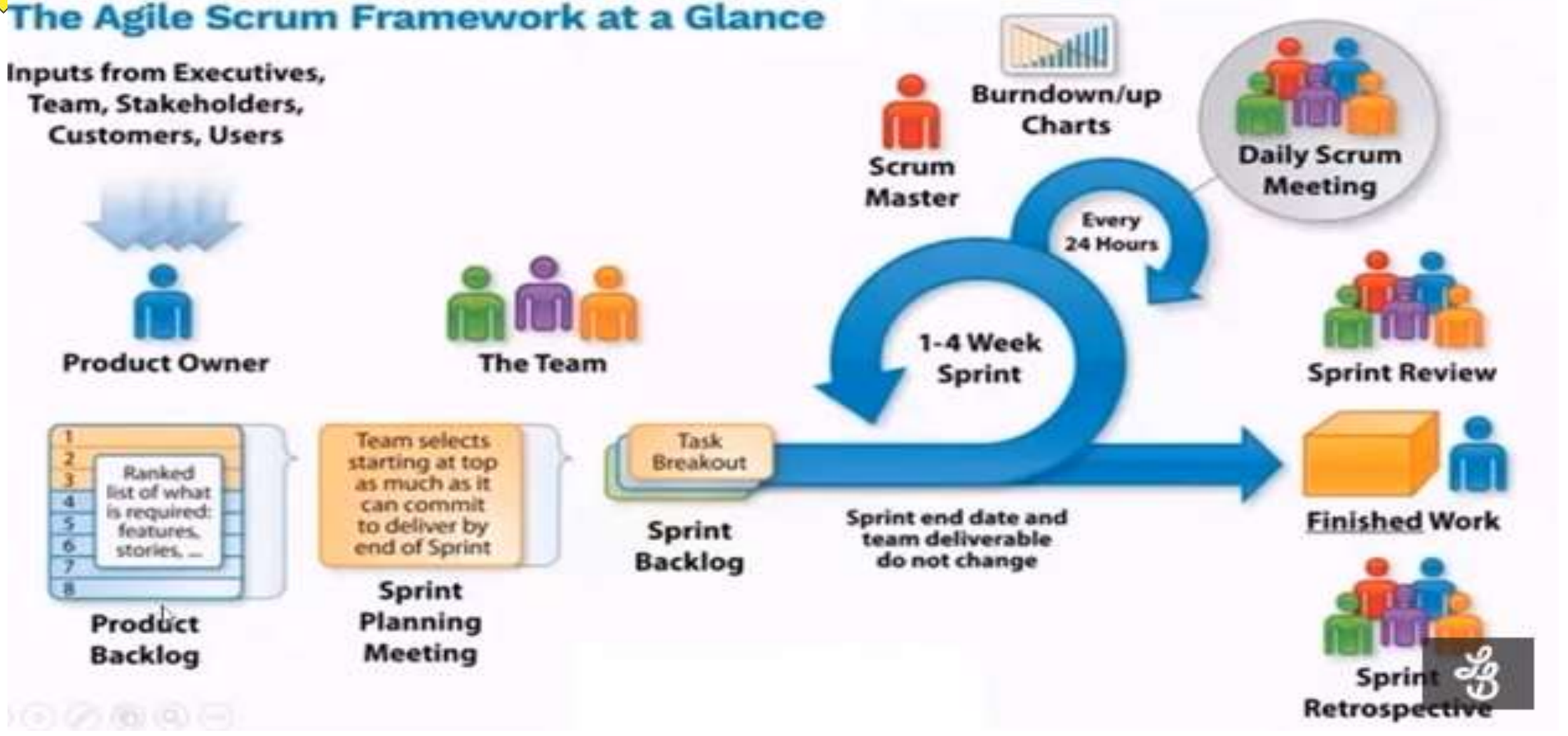
- Framework for managing work with an emphasis on software development
- Software development ⟶ **ITERATIONS** (within time boxed)

⬇

Sprints (typically two weeks)

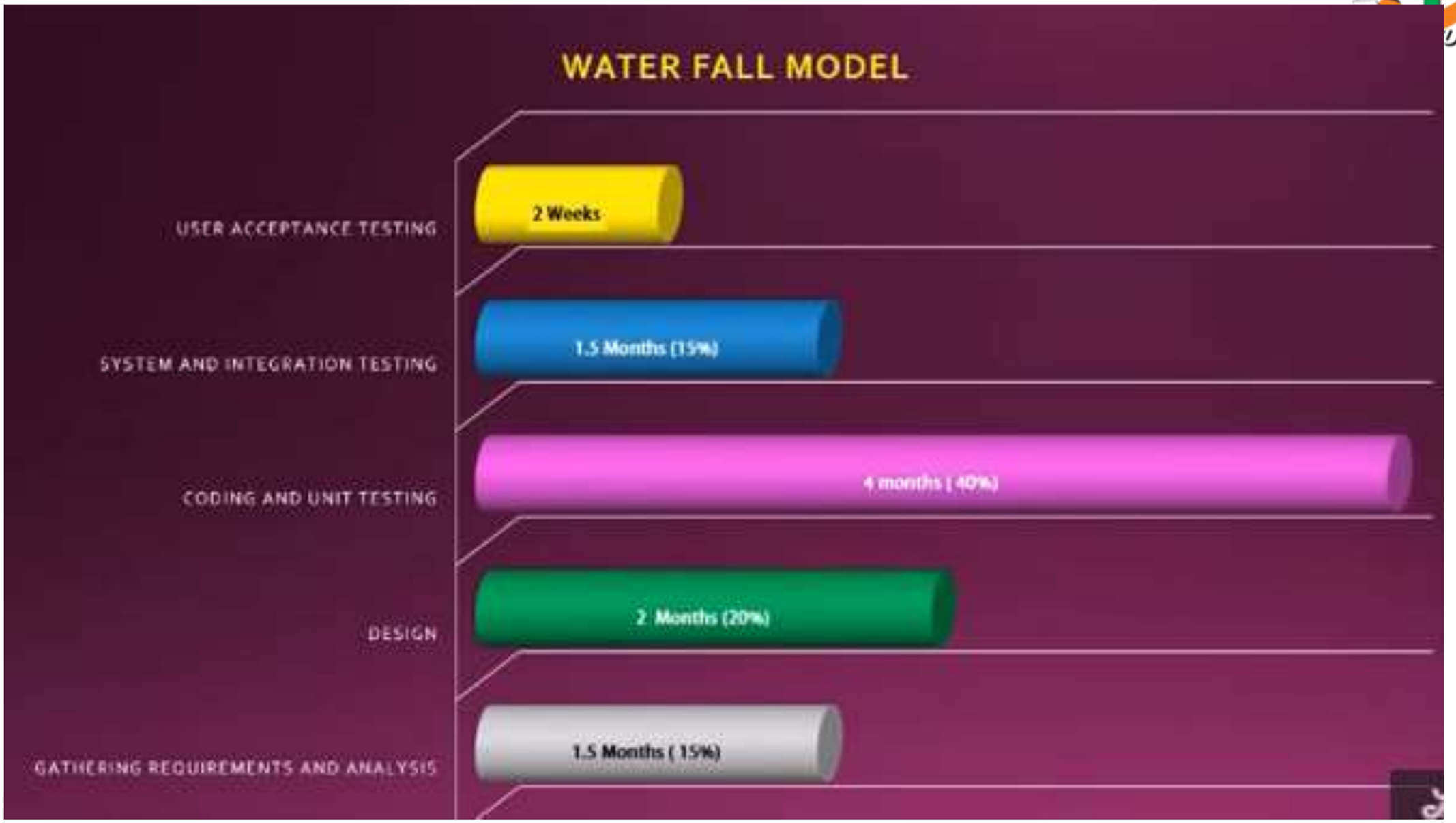
SCRUM key Roles →
- Scrum Master
- Scrum Product Owner
- Development Team

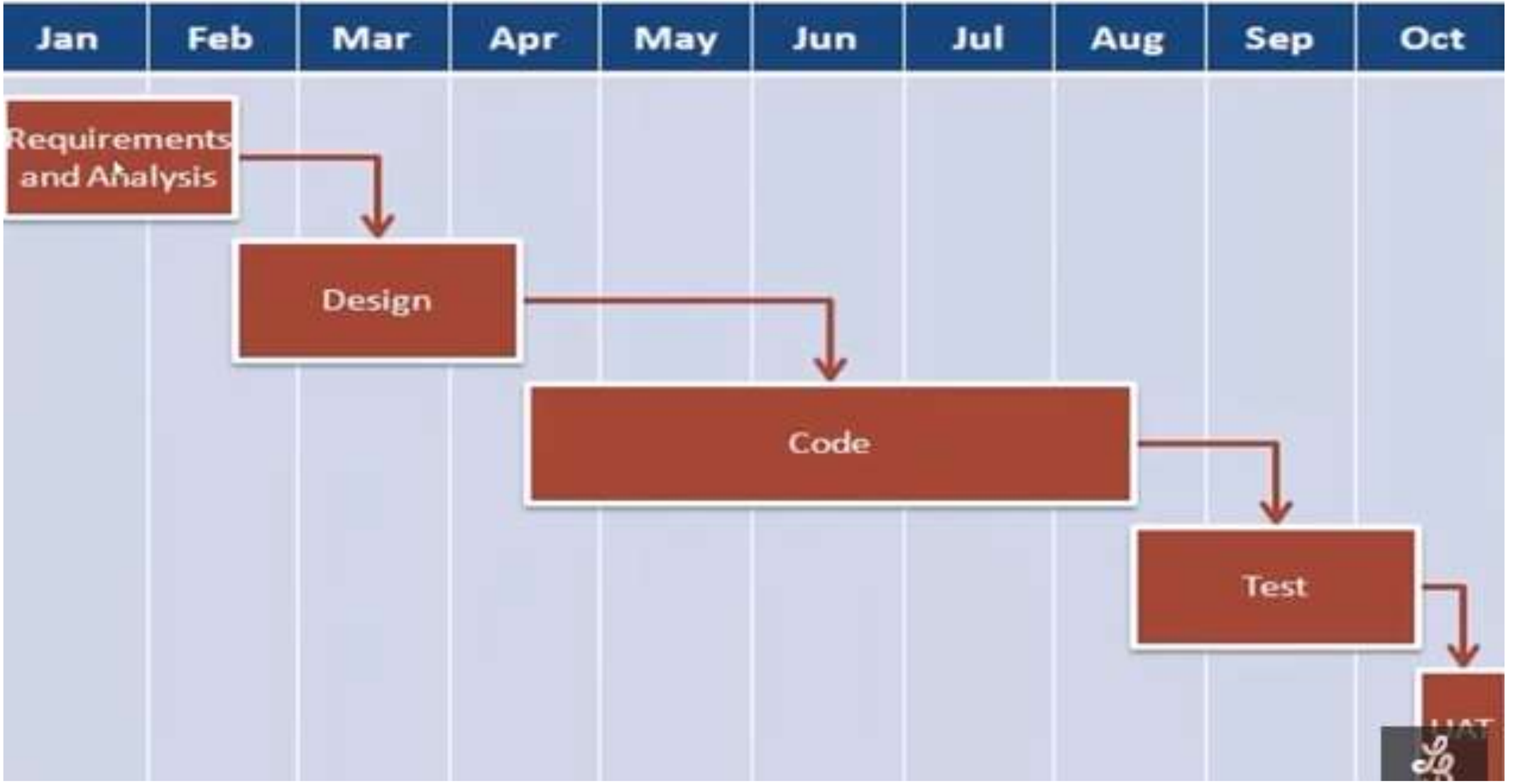
The Agile Scrum Framework at a Glance



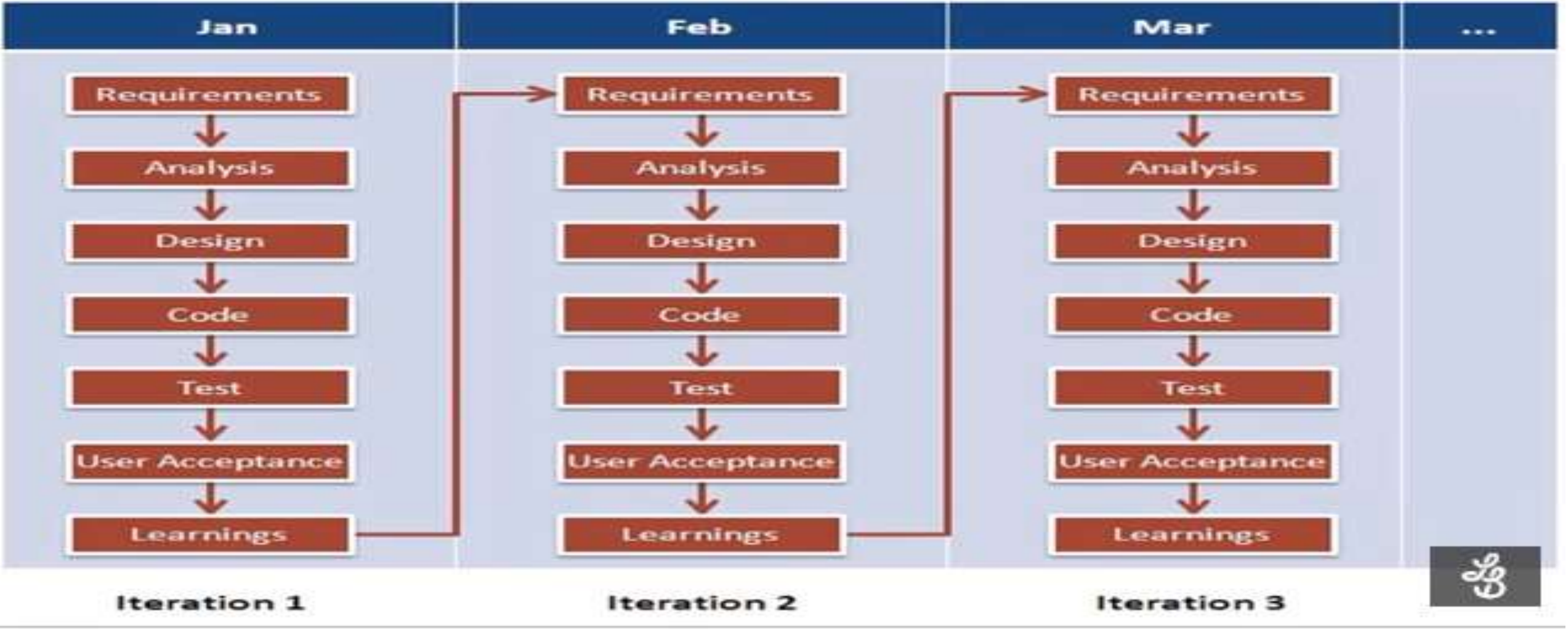
Example of AGILE software development

- **Product** –MS Word
- **Features** – All the features provided by MS Word and any other features requested by the marketing team
- **Duration -**10 Months


WATER FALL MODEL
USER ACCEPTANCE TESTING
2 Weeks
SYSTEM AND INTEGRATION TESTING
1.5 Months (15%)
CODING AND UNIT TESTING
4 months (40%)
DESIGN
2 Months (20%)
GATHERING REQUIREMENTS AND ANALYSIS
1.5 Months (15%)


Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Requirements and Analysis
Design
Code
Test

AGILE METHODOLOGY



ADVANTAGES

- The delivery of software is unremitting
- Customer satisfaction is more here
- Changes of the product are done easily and on time
- The daily interactions are required between the business people and the developers. Attention is paid to the good design of the product

DISADVANTAGES

- Documentation is less
- Unknown risks may occur due to the lack of risk analysis which may leads to unexpected outcome rarely