SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

Re-accredited by NAAC with A+ grade, Accredited by NBA(CSE, IT, ECE, EEE & Mechanical)

Approvedy by AICTE, New Delhi, Recognized by UGC, Affiliated to Anna University, Chennai

Department of MCA

Topic: AGILE SOFTWARE DEVELOPMENT INTRODUCTION

Course

23CAE718 AGILE SOFTWARE DEVELOPMENT

Unit II

AGILE SOFTWARE DEVELOPMENT INTRODUCTION

Elective

II Semester / I MCA

Introduction to **AGILE SOFTWARE DEVELOPMENT**

An Introduction to Agile Software Development

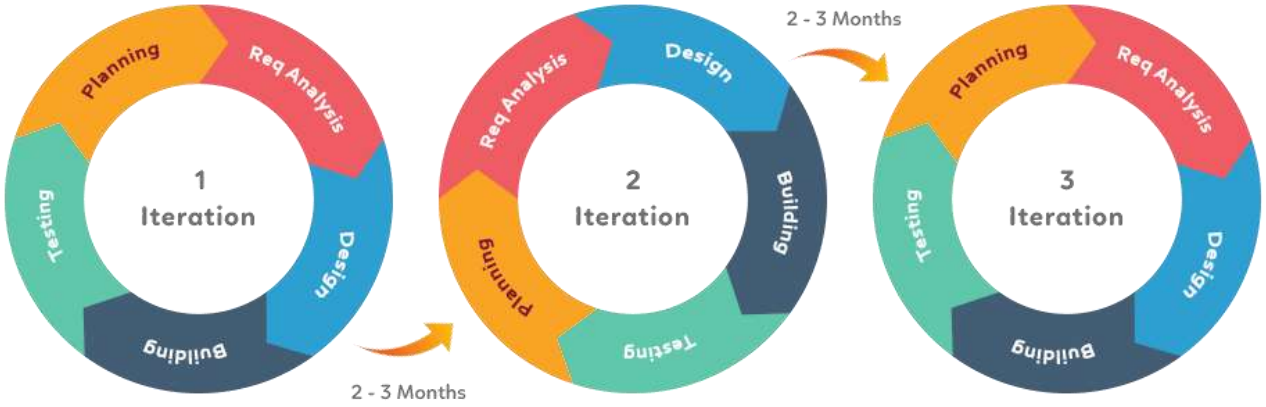
- Agile software development, it is a programming methodology that emphasizes rapid and iterative software development approaches.

What is Agile?

Agile is a popular technology that allows you to build and respond to changes. It is a collection of several principles that are used in the field of project management and software development. This practice works on the continuous iteration of testing and development for the complete Agile software development lifecycle of a given business project.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

What is Agile?

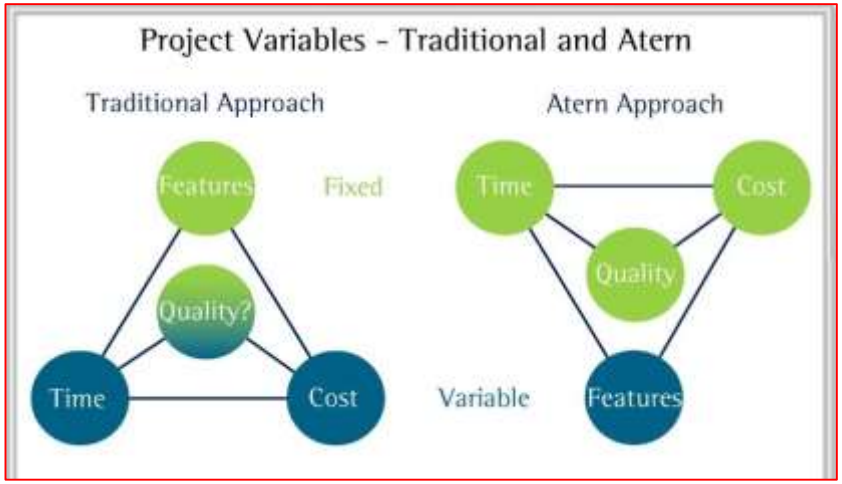


Introduction to **AGILE SOFTWARE DEVELOPMENT**

What is Agile?

Agile software development refers to methods and practices that provide value quickly, efficiently, and consistently to customers as it relates to the software development lifecycle (SDLC).

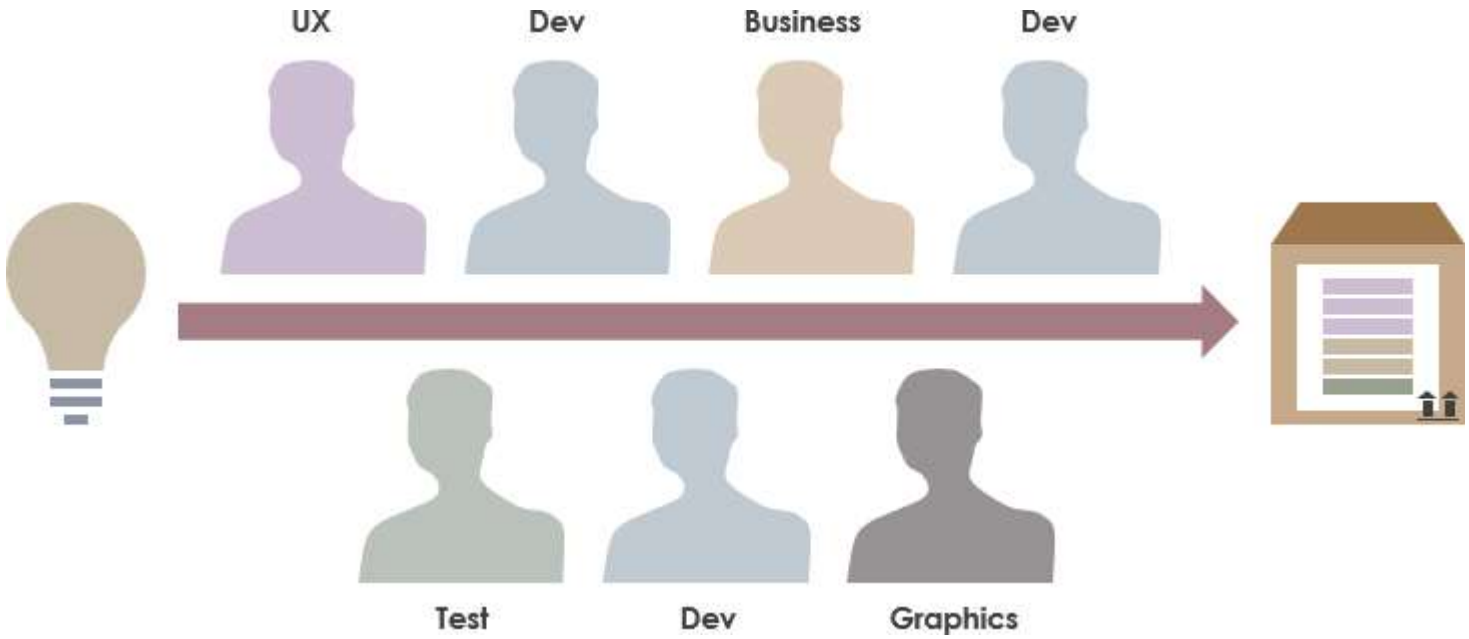
The ability to build and react to change is called *Agile*.



Introduction to **AGILE SOFTWARE DEVELOPMENT**

What is Agile?

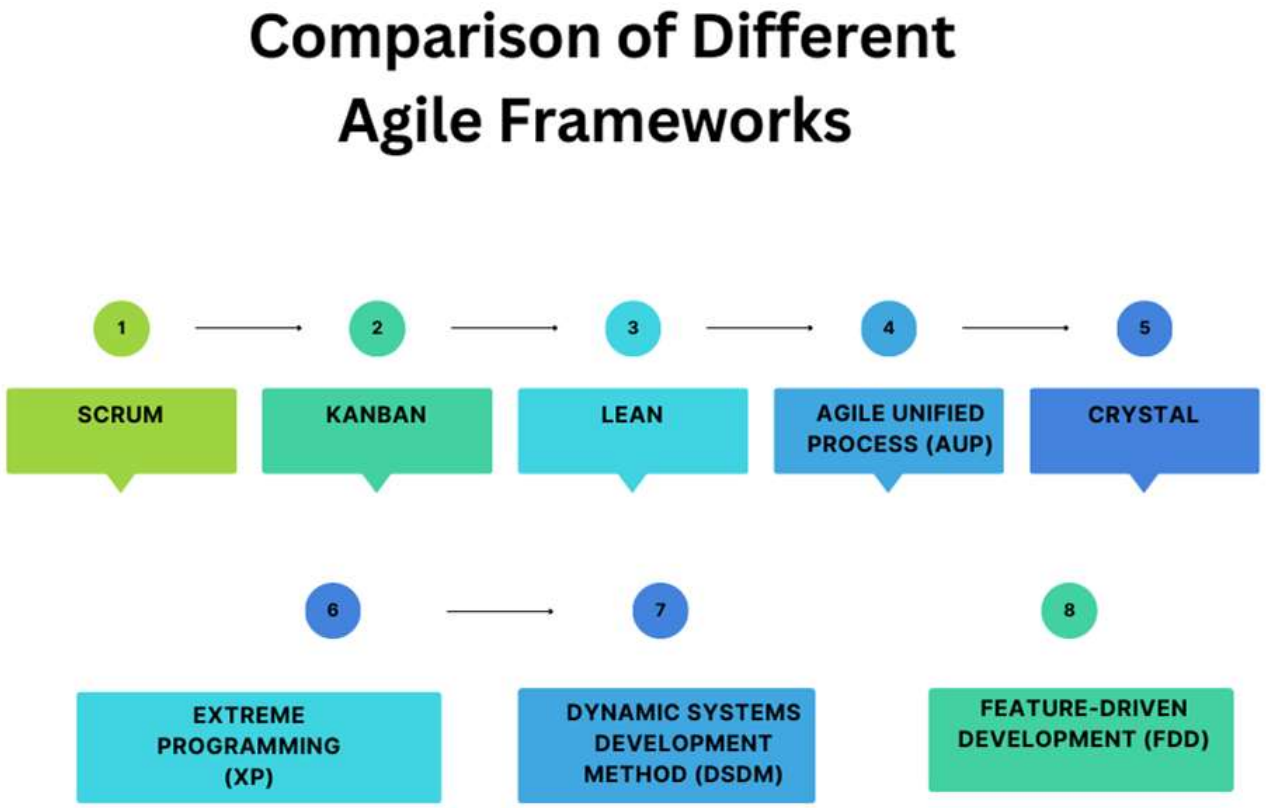
Within the Agile software development model, self-organizing and cross-functional teams work together to build and deploy solutions.



Introduction to **AGILE SOFTWARE DEVELOPMENT**

What is Agile?

Some of the popular Agile methodologies include Scrum, Kanban, and Lean.



Introduction to **AGILE SOFTWARE DEVELOPMENT**

What is Agile?

Agile splits enormous software development tasks into smaller, more manageable parts called *iterations*. Agile differs from other software development methodologies in that it focuses mainly on the people performing the job and the cooperation, collaboration, and communication between them.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

What is Agile?

- Solutions emerge from collaboration amongst self-organizing cross-functional teams that use the best techniques available.
- Instead of writing one extensive and comprehensive application, the Agile method involves breaking software applications into smaller, more manageable pieces. These pieces are built, tested, and deployed in iterations. Iterations are short, time-boxed periods during which you change your application's features and functionality. Most importantly, you can implement these changes quickly.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Why Agile?

- It simplifies the development work, which is why this method became known primarily as "agile" in managing IT projects.
- Each programmer represented different methodologies such as Extreme Programming, SCRUM, DSDM, Adaptive Software Development, Crystal, Feature-Driven Development, and Pragmatic Programming.
- They gathered their experience in software development and interaction with teams, solutions, principles and work methodology into one Manifesto known as the Agile Manifesto.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

- The agile methodology gives the team of programmers much more responsibility.
- instead of spending months or even years manufacturing a product, agile teams only spend a few weeks in a work phase.
- A finished product, an update, or a part of the program can be presented to the customer. The Agile Manifesto agreed on twelve principles and four values for this to succeed.

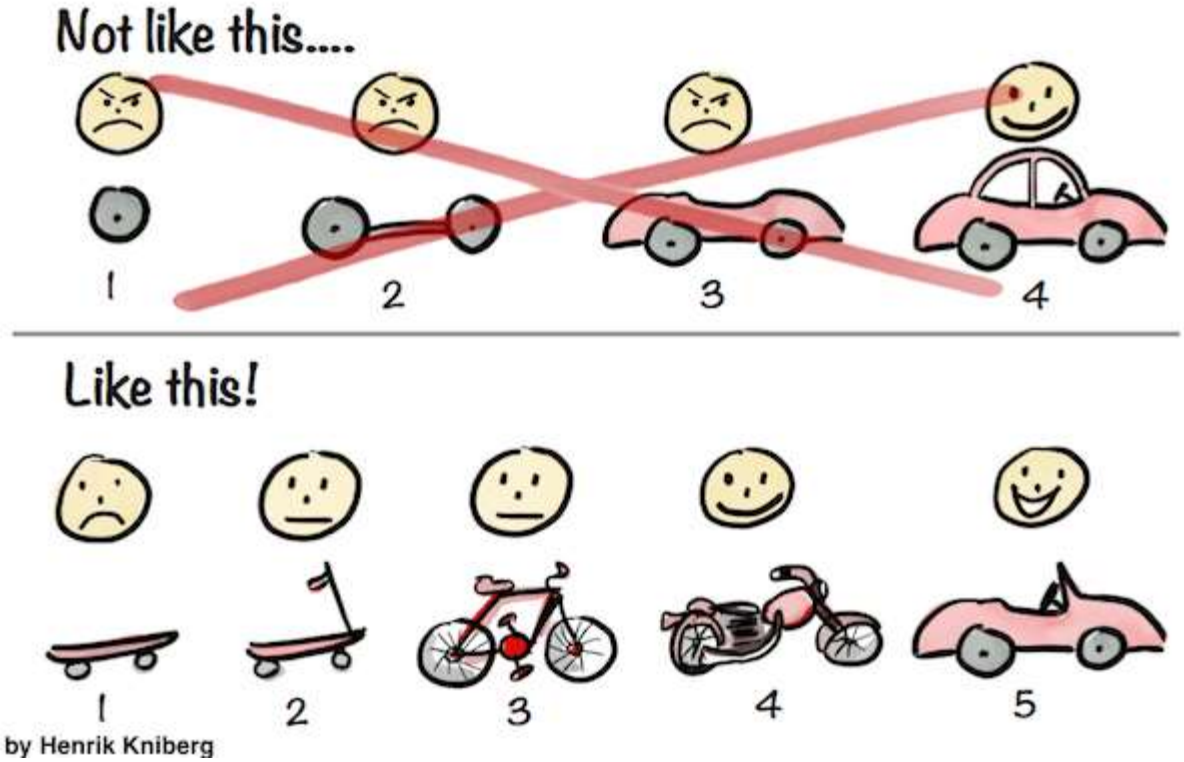
Introduction to **AGILE SOFTWARE DEVELOPMENT**

What is Agile Software Development?

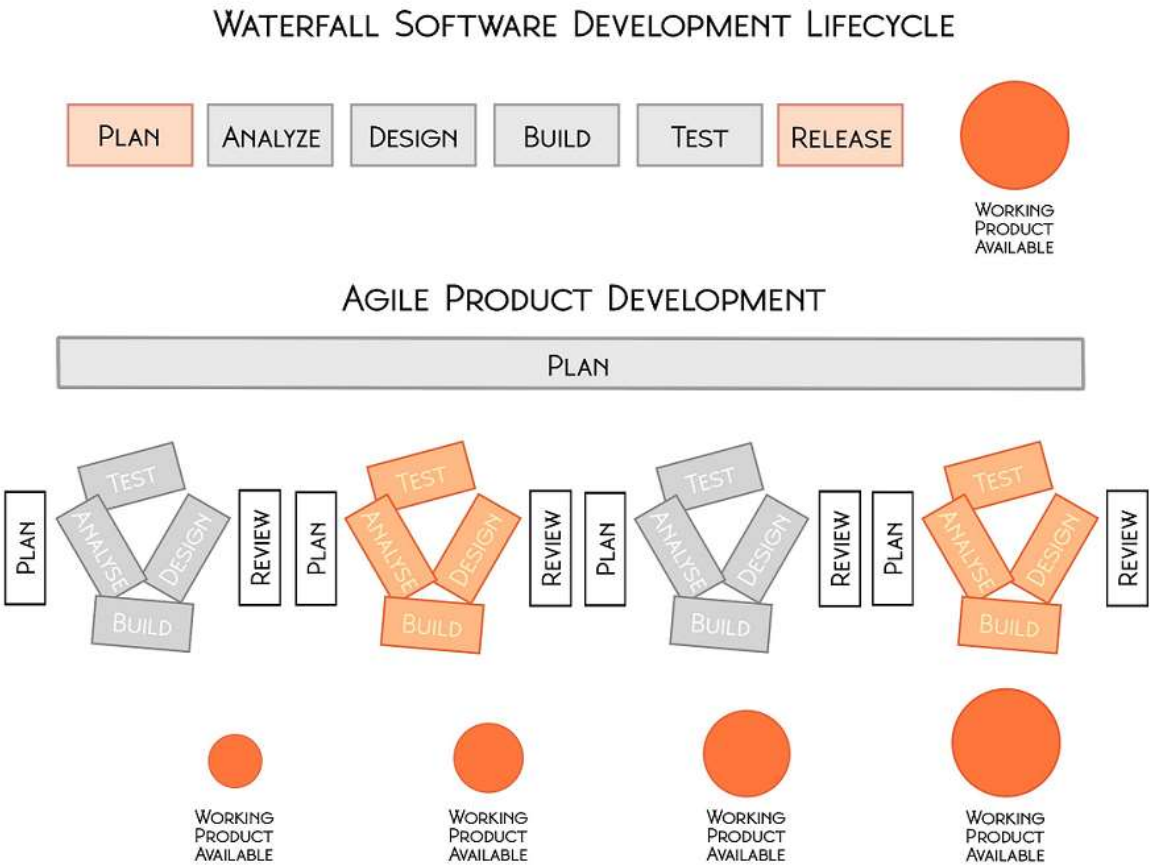
❑The Agile software development paradigm is a software development methodology comprising practices and approaches that thrive on iterative and incremental software development. In this development methodology, the requirements – as well as the solutions – evolve through collaboration amongst self-organizing, cross-functional programmer teams that may not or may not be collocated or remote.

❑The Agile approach fosters adaptability, evolutionary development, and delivery, as well as a time-bound, iterative approach and quick response to change. Agile promotes adaptive planning, evolutionary growth, early delivery, and continual improvement.

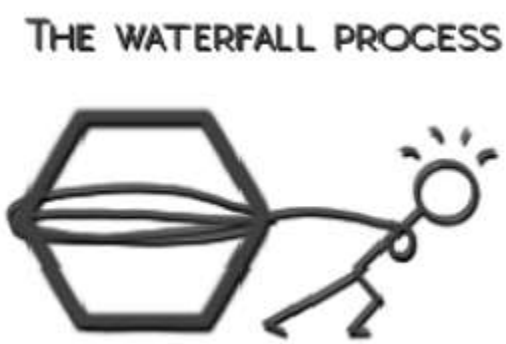
Introduction to **AGILE SOFTWARE DEVELOPMENT**

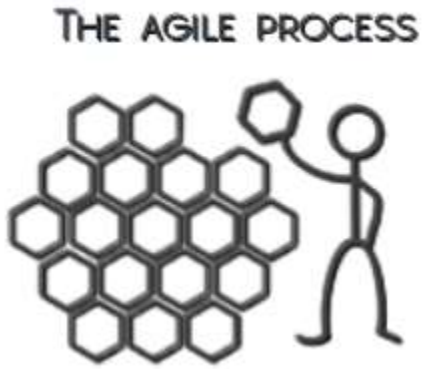


Introduction to **AGILE SOFTWARE DEVELOPMENT**

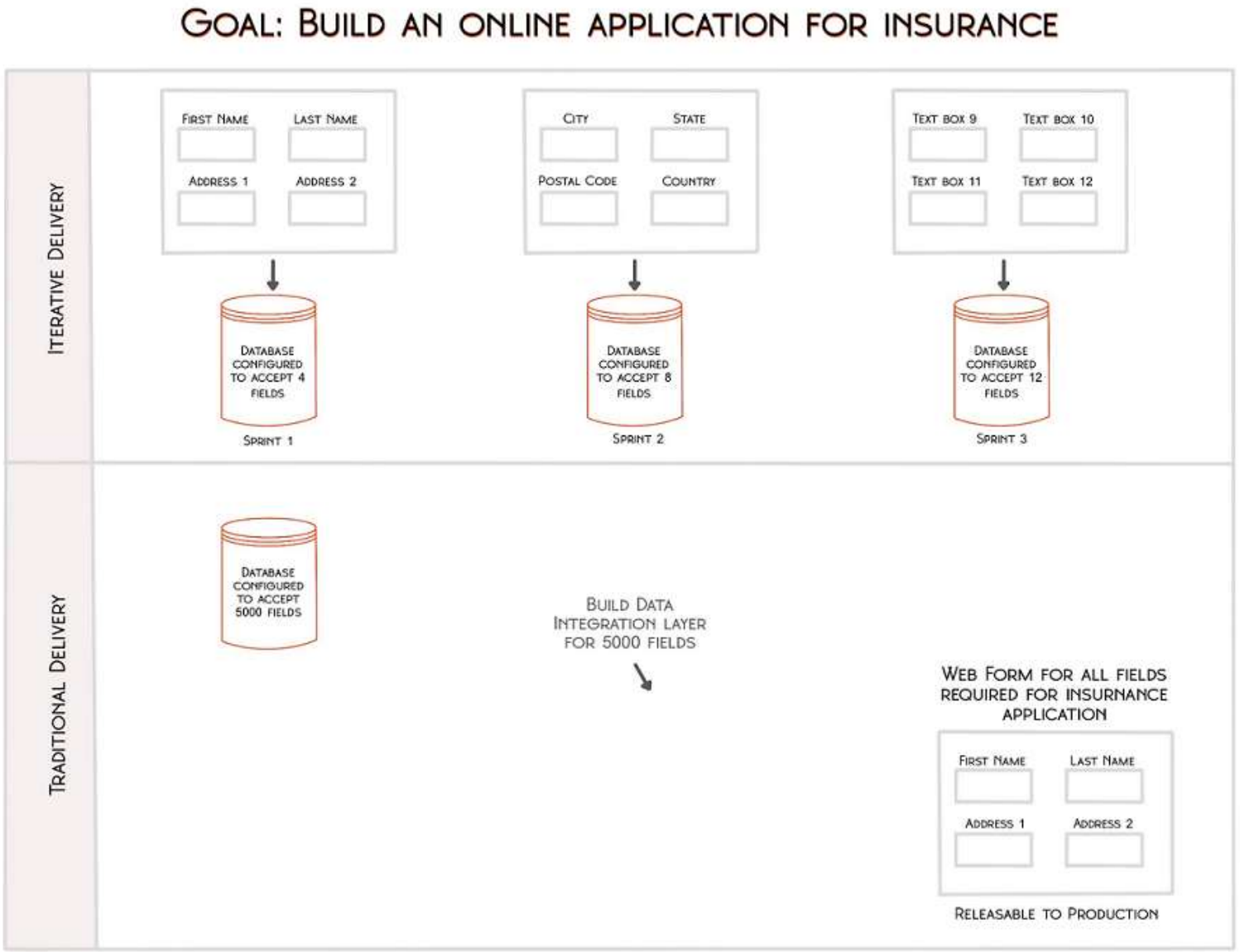


Introduction to **AGILE SOFTWARE DEVELOPMENT**





Introduction to **AGILE SOFTWARE DEVELOPMENT**



Introduction to **AGILE SOFTWARE DEVELOPMENT**

Why Should I Use Agile?

Agile was established for the software industry to simplify and expedite the development process to detect and correct errors and difficulties swiftly. It enables teams and developers to produce a better product in less time by using short, iterative *sprints* or sessions. And, with more businesses transitioning to the digital workplace, being Agile is a perfect match for enterprises aiming to improve the way they generally manage projects and function.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Why Should I Use Agile?

The most significant benefit of the Agile development methodology is that it fosters shorter development cycles. So, you can get your products or features to market faster and start seeing returns on your investment sooner.

Agile also enables more collaboration between different team members, leading to a more cohesive team and better products. And finally, Agile allows you to get better customer feedback, which can help you make better decisions about what to build next.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Here are the benefits of the Agile methodology:

- Shorter development cycles
- Better flexibility
- Faster, incremental releases
- Adaptability
- Improved quality
- Better risk management
- Reduced costs

Introduction to **AGILE SOFTWARE DEVELOPMENT**

What is The Agile Lifecycle?

Project Planning – This helps your team understand the goals, the value to be delivered to the stakeholders, and define the project scope.

Product Roadmap – This helps to define a breakdown of all the features that are needed as part of the final deliverable.

Release Planning – This helps plan the future releases and revisit and reevaluate the release plans before a sprint starts.

Sprint Planning – This helps plan how the tasks in a sprint should be accomplished, by whom and the time it would take to complete those tasks.

Daily Scrum Meetings – These meetings are usually short and help the team know the tasks to be accomplished on a particular day, the roadblocks (if any) and assess if any changes are required.

Sprint Review / Retrospective Meetings – These meetings are usually held after every sprint to discuss what went well in the sprint and what did not or what could have been done better.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

The principles outlined in the Agile Manifesto enable businesses to strive towards excellence. The manifesto promotes trust, transparency, collaboration, and consumer involvement. While software strategies may seem basic, it is always critical to work with experts that understand the significance of delivering a quality product and maintaining customer satisfaction.

The Agile Manifesto has four key values:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Agile Manifesto

Agile Values

We are uncovering **better ways of developing software by doing it and helping others do it.** Through this work we have come to value:

Individuals and interactions over **processes and tools**

Working software over **comprehensive documentation**

Customer collaboration over **contract negotiation**

Responding to change over **following a plan**

That is, while there is **value in the items on the right**, we **value the items on the left more**.

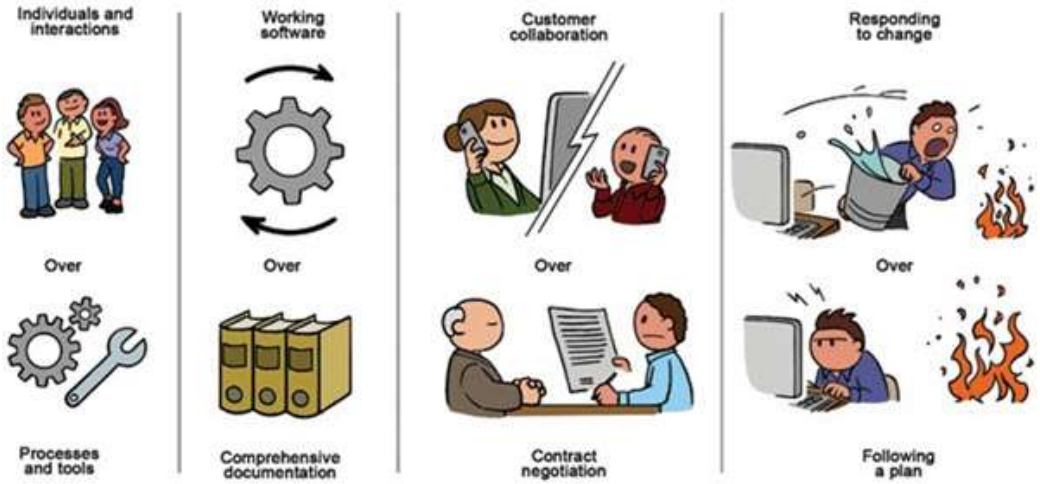


FIGURE 4.3 *Agile Manifesto Values*

Introduction to **AGILE SOFTWARE DEVELOPMENT**

E-commerce Platforms: Development and maintenance of platform

Agile Use: Frequent updates are needed to support new features (like search optimization, UI/UX design changes, payment integrations, etc.). Agile allows for rapid prototyping, testing, and deployment of features in short sprints.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Mobile Application Development: Development of apps for **Uber**, **Instagram**, **WhatsApp**, and **Spotify**.

Agile Use: Mobile apps require constant iterations for new features, bug fixes, and platform updates. Agile ensures fast releases and continuous user feedback for rapid improvement.

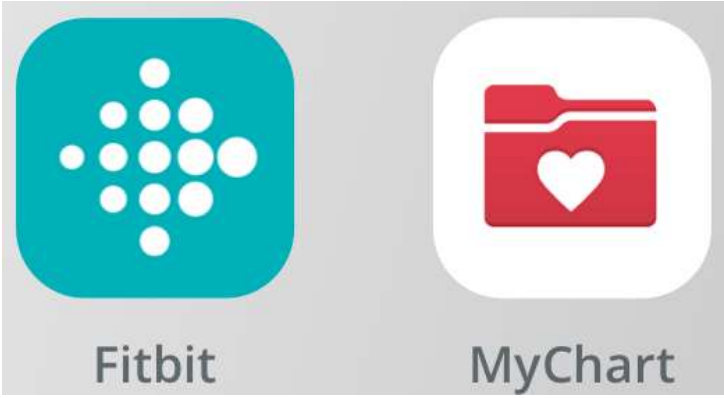
Introduction to **AGILE SOFTWARE DEVELOPMENT**

Financial and Banking Software:Development of online banking systems, payment gateways (e.g., **PayPal**, **Stripe**), and mobile wallets (e.g., **Google Pay**, **Apple Pay**).

Agile Use: These systems require agility to accommodate regulatory changes, security improvements, and frequent feature enhancements, making Agile a natural fit.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Healthcare Software: Development of **electronic health record (EHR)** systems, telemedicine platforms, or mobile health apps (e.g., **MyChart**, **Fitbit** integration).



Agile Use: Healthcare software needs to adapt to new regulations (e.g., HIPAA compliance) and integrate with various devices and systems. Agile facilitates rapid deployment of updates, critical security patches, and improved patient experiences.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Cloud-based Software Platforms: Development of **SaaS** products like **Salesforce**, **Microsoft 365**, **Google Workspace**, and **Slack**.

Agile Use: Frequent updates, integrations, and feature improvements are essential in the SaaS space. Agile practices allow teams to quickly respond to customer feedback and market needs.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Gaming Software: Development of video games, such as **Fortnite**, **Call of Duty**, **Among Us**, or **Minecraft**.

Agile Use: Games require constant updates for bug fixes, new features, content (e.g., new levels, characters, etc.), and player feedback. Agile's iterative nature helps deliver small updates frequently.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Gaming Software: Development of **SAP**, **Oracle ERP**, and **Microsoft Dynamics**.

Agile Use: These large-scale software solutions often need to evolve based on client needs, industry standards, and new technologies. Agile allows teams to release incremental updates and features in each sprint.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Customer Relationship Management (CRM) Software:

Development of CRM tools like **Salesforce**, **HubSpot**, or **Zoho CRM**.

Agile Use: Agile practices help continuously improve CRM systems by adding customer-requested features, integrating third-party applications, and ensuring a positive user experience.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

AI and Machine Learning Development: Building AI solutions like **recommendation systems** (e.g., **Netflix**, **Amazon**), **chatbots** (e.g., **Slack bots**, **Google Assistant**), and **predictive analytics** tools.

Agile Use: AI and ML projects often require experimentation and iterative improvement. Agile helps refine models, test them, and get feedback from real-world data, ensuring that solutions evolve based on insights.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

IoT (Internet of Things) Solutions: Development of IoT platforms for **smart homes**, **wearables**, **connected cars**, and **industrial IoT** (e.g., **Nest**, **Ring**, **Fitbit**).

Agile Use: IoT projects involve hardware and software development, where continuous updates and improvements are critical. Agile helps deliver fast, iterative updates on software and firmware.

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Cybersecurity Solutions: **McAfee**, **Norton**, **Palo Alto Networks**, and **Cloudflare**.

Social Media Platforms: **Facebook, Twitter, LinkedIn, TikTok, or Snapchat.**

Travel and Hospitality Software: **Expedia, Airbnb, Booking.com, and TripAdvisor.**

Education Technology (EdTech): **Coursera, Duolingo, Khan Academy, and Zoom.**

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Real Estate Software: **Zillow**, **Realtor.com**, **Redfin**, or property management tools.

Retail Software: Development of **point-of-sale (POS) systems**, inventory management systems, or **retail management software.**

Automotive Software: Development of autonomous driving technologies, vehicle telematics, or car maintenance platforms (**e.g., Tesla, Ford, Uber**).

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Supply Chain and Logistics Software: Development of software for **UPS**, **FedEx**, or **DHL** to track shipments, manage inventory, or optimize routes.

Entertainment and Media Software: Development of platforms like **Netflix**, **Hulu**, **Disney+**, and **YouTube**.

Blockchain Solutions:Development of blockchain platforms such as Ethereum, Bitcoin, or decentralized apps (dApps).

Introduction to **AGILE SOFTWARE DEVELOPMENT**

Benefits of Using Agile in These Projects:

- **Flexibility**: Agile enables the project to adapt to changing requirements.
- **Faster Delivery**: Agile focuses on delivering functional software incrementally.
- **Continuous Improvement**: Through constant feedback loops, features can be optimized and adjusted as development progresses.
- **Collaboration**: Agile fosters collaboration among team members and stakeholders to meet business needs efficiently.

References

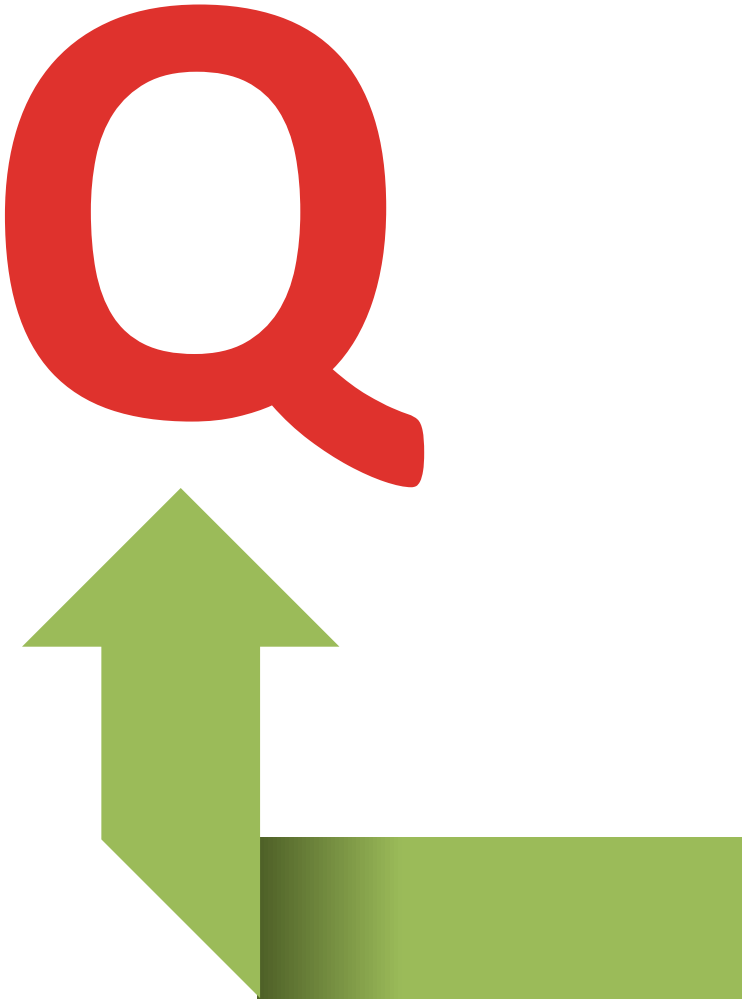
Text Books

1. Ken Schawber, Mike Beedle, “Agile Software Development with Scrum”, International Edition, Pearson.

2. Robert C. Martin, “Agile Software Development, Principles, Patterns and Practices”, First International Edition, Prentice Hall.

Web Resources

- https://www.youtube.com/watch?v=tOPt_xgsuP4
- https://www.youtube.com/watch?app=desktop&v=kTC7YD9Y3Js
- https://www.developer.com/project-management/intro-agile-development/




Thank
you