Design Patterns - Background

Complex software systems **are plagued** with **many issues**:

1. **Timelines are stretched** as requirements change.
2. Multiple developers have a **hard time coordinating** their efforts.
3. Often there is **Code redundancy** and poor documentation.

This in turn creates issues with **maintenance** and overall flexibility for **adding new features.**

In general, this means poorly designed systems that are hard to maintain, and **are not adaptable**.

Design Patterns - Background

One answer to all the cited problems is having a proper **design and architecture.**

Think of a skyscraper being built.

There is always a **high-level blueprint**.

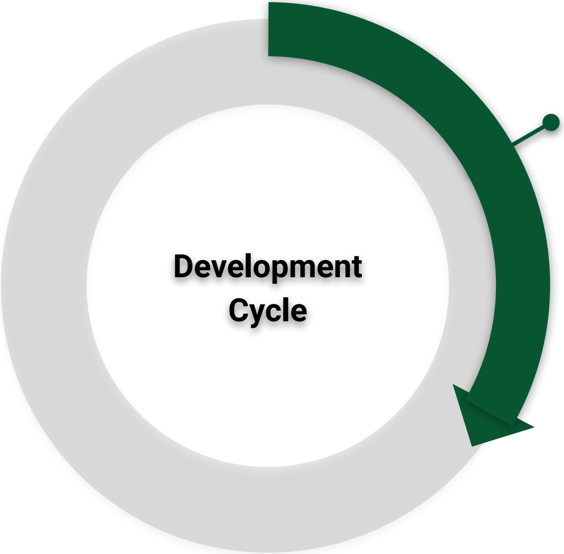
This blueprint is used to show everybody involved (from architects, to supply chain, to construction workers, to machinery scheduling etc...) what is being worked on.

We want that kind of predictability and coherence in our Software Projects.

Design Patterns - Background

How do we achieve that?

Design Patterns - Background



REQUIREMENTS

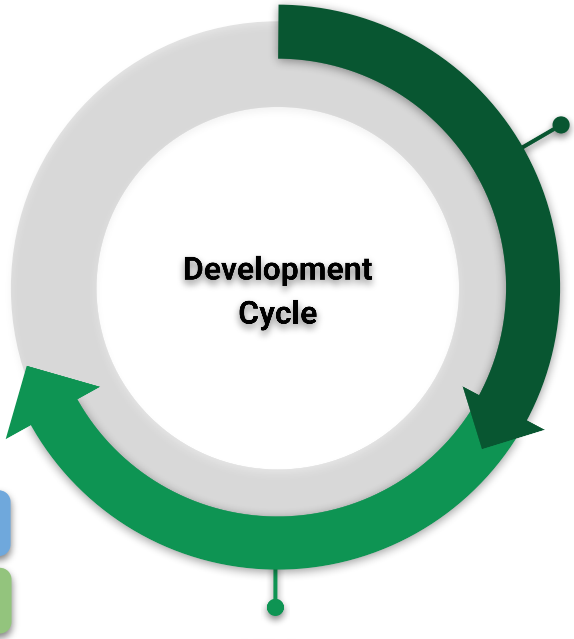
Gather and document requirements for UI, UX, and general goals of the project.

Requirement Docs

Design Docs

UI/UX

Design Patterns - Background



REQUIREMENTS

Gather and document requirements for UI, UX, and general goals of the project.

Requirement Docs

Design Docs

UI/UX

DESIGN

Architect the system based on the requirements.

Deployment Docs

Architecture Docs

ERD

UML

Design Patterns

Design Patterns - Background

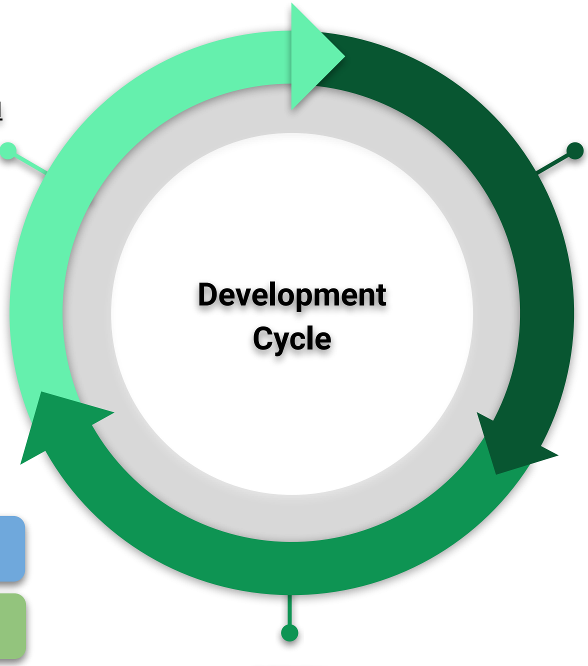
Source Code

Analytics

Audit

Logs

Tests



Development Cycle

IMPLEMENTATION

Code and test the solution to ensure that it follows the design and fulfills the requirements.

REQUIREMENTS

Gather and document requirements for UI, UX, and general goals of the project.

Requirement Docs

Design Docs

UI/UX

DESIGN

Architect the system based on the requirements.

Deployment Docs

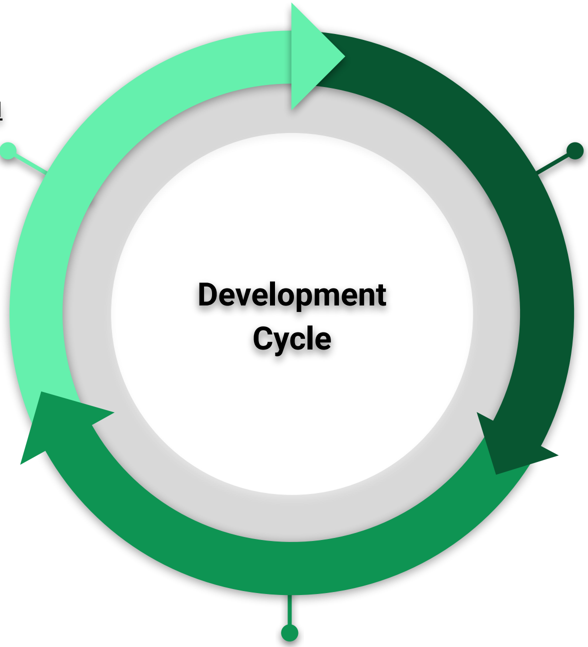
Architecture Docs

ERD

UML

Design Patterns

Design Patterns - Background



Development Cycle

REQUIREMENTS

Gather and document requirements for UI, UX, and general goals of the project.

DESIGN

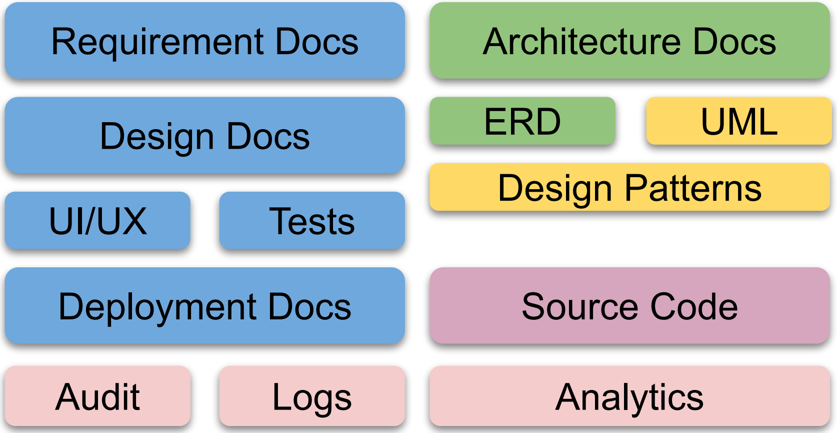
Architect the system based on the requirements.

IMPLEMENTATION

Code and test the solution to ensure that it follows the design and fulfills the requirements.

Design Patterns - Background

In this course we will concentrate on the ‘Design’ part, and we will explore how Design Patterns facilitate and help achieve good Software Engineering.



Requirement Docs

Architecture Docs

Design Docs

ERD

UML

Design Patterns

UI/UX

Tests

Deployment Docs

Source Code

Audit

Logs

Analytics

Projects will have many moving parts, but in the realm of software, the final product is the **source code.** Good Architecture is crucial.

Design Patterns help greatly in that aspect.