The Software Development Process

Course Structure

- This course is structured in a way that you can understand the phases of the software development process, their logical and temporal sequences, their relationships and the activities involved
- For each phase you will find one or more distinct sections
- We will analyse each theoretical part focusing on a selected case study, a eLearning web application

Need to know

Some definitions
The Scenario
Roles & Responsibilities
Understand the customer requirement
Presentation of the case study

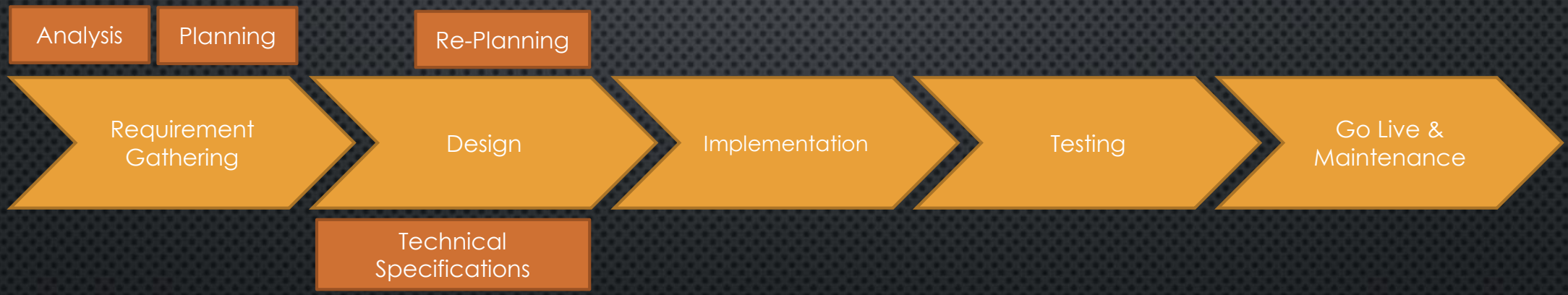
Need to know

Software Development Models

What is a Software development model
Software Development Life Cycle (SDLC)
What are the Software Development Life Cycle (SDLC) phases?
List of the Software development models
Description of the most popular Software development models
How to select the right Software Development Life Cycle Model
Which factors determine the selection of the development model

PHASES OF A SOFTWARE DEVELOPMENT PROCESS

While there is not standard definition, most development processes include the following activities:



Need to know

Software Development Models

Phase 1 - Requirement Gathering

What is requirement gathering?
Initial Requirements
Uncertainty in Technology
Case Study - An eLearning membership website - customer requirement
Identify the solution
Identification of the skills and technologies
Work organization
Ways to collect the requirements
Analyse the requirements
Review the customer processes and define what to automate and what not
Agreement and sign off

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Introduction, UML
Diagram Types
Class Diagrams
UML Relationships
UML Packages
UML Composte Structures
Component diagrams
Use case diagrams
Use case modelling
Use case document
Interaction & Collaboration Diagrams
Statechart Diagrams
Activity Diagrams

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

What is software architecture?
What is software design?
Software architecture vs Software design
Software architecture: basics
Components, packages, Interactions
Interaction Oriented Architecture, MVC Pattern
Security, Performance, Fault-tolerance, Robustness
Extensibility, s3rd party usage
High level design
Low level design - use cases
Low level design - detailed process descriptions (the technical briefing)
Back end design

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

What is a database
Relational database
Define a database
primary key
Foreign key
Roles and privileges
Case Study - Database structure

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

What is a technical requirement
Case Study - technical requirement
Scrum methodology - the requirements
user story
Scrum methodology - the definition of done
Case Study user stories
Case Study - definition of done

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Installations
Configurations
Customizations
Integrations
User Training
Structured Programming
Functional Programming
Programming style
Examples of programming style guidelines
Code re-use
Multi-site Distributed Software Development
Development environments and IT infrastructures

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

What is Version Control?
Git and GitHub
git branching model
Git Terms: Repository, Staging, Commit, Push, Pull
Git in action
Software versioning

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

Phase 3 - Implementation - Risk assessment

Software Risk Identification
Software Risk Analysis
Software Risk Planning
Software Risk Monitoring

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

Phase 3 - Implementation - Risk assessment

Phase 3 - Implementation - Change management

What is Change Management in Software development
The Change Management Process
Agile Change Management Process

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

Phase 3 - Implementation - Risk assessment

Phase 3 - Implementation - Change management

Phase 3 - Implementation - The deployment process

The deployment process
The documentation for the deployment
Agile Software Deployment
Regression testing

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

Phase 3 - Implementation - Risk assessment

Phase 3 - Implementation - Change management

Phase 3 - Implementation - The deployment process

Phase 4 - Quality Assurance - Verification

Verification phase
Software Quality
Software Testing Life Cycle
Agile Methodology in Testing
How to check the quality of a software product?
Overview of the tests

Definition of test scenario
Definition of test case
Example of test case
Case Study - define the test scenarios
Integration tests
Performance tests

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

Phase 3 - Implementation - Risk assessment

Phase 3 - Implementation - Change management

Phase 3 - Implementation - The deployment process

Phase 4 - Quality Assurance - Verification

Phase 4 - Quality Assurance - Validation

What is software validation?
User Acceptance tests
Organize the user acceptance tests

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

Phase 3 - Implementation - Risk assessment

Phase 3 - Implementation - Change management

Phase 3 - Implementation - The deployment process

Phase 4 - Quality Assurance - Verification

Phase 4 - Quality Assurance - Validation

Phase 4 - Quality Assurance - Incident Management

Incident Management
The Incident Management Report
The Incident Management Process
Incident Management System
Case study

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

Phase 3 - Implementation - Risk assessment

What is a Software Go Live
The Go Live process
UAT exit procedure
Project sign-off
Preparation
Go Live checklist
Cut over
Go Live
Lesson learned

Phase 3 - Implementation - The deployment process

Phase 4 - Quality Assurance - Verification

Phase 4 - Quality Assurance - Validation

Management

Phase 5 - Go Live

Software Development Process

Need to know

Software Development Models

Phase 1 - Requirement Gathering

Phase 2 - DESIGN - The Unified Modeling Language (UML)

Phase 2 - DESIGN - Software Architecture and design definition

Phase 2 - DESIGN - Database Architecture

Phase 2 - DESIGN - Technical Requirements

Phase 3 - Implementation - Development

Phase 3 - Implementation - Version Management

Phase 3 - Implementation - Risk assessment

Phase 3 - Implementation - Change management

Phase 3 - Implementation - The deployment process

Phase 4 - Quality Assurance - Verification

Phase 4 - Quality Assurance - Verification

Phase 4 - Quality Assurance - Incident Management

Phase 5 - Go Live

Software Maintenance

Software Maintenance definition
Maintenance agreements
Software Maintenance Process
Software Maintenance Methods

Software Maintenance

Software Platform Development

What is a software platform
Product customizations
Case Study - define the platform

Software Maintenance

Software Platform Development

Agile Frameworks: SCRUM & Kanban

Agile methodology
Agile Frameworks - Scrum
The Scrum Sprint Cycle
Scrum Release Planning
Sprint Planning
The Daily Scrum
The Sprint Review
The Retrospective Meeting
The Kanban Agile System
Kanban Boards
Team size in a Scrum project

Software Maintenance

Software Platform Development

Agile Frameworks: SCRUM & Kanban

Object-Oriented Programming Concepts

What is OOP?
Objects
The Class
Encapsulation and Data Hiding
Abstraction
Inheritance
Polymorphism
Interfaces

Software Maintenance

Software Platform Development

Agile Frameworks: SCRUM & Kanban

Object-Oriented Programming Concepts

Project Management

Project management concepts
Characteristics of a project
Tasks in a project
The Output of a project
Diagram of a project
The work breakdown structure: breakdown the project phases
The work breakdown structure: assign WBS codes to phases and tasks
The release plan
Traditional project management
Agile project management: Scrum
Case Study - setup the project plan

Software Maintenance

Software Platform Development

Agile Frameworks: SCRUM & Kanban

Object-Oriented Programming Concepts

Project Management

Documentation management

The importance of the documentation in a software development process
Overview of the documentation
Style guidelines
Guidelines for third party integration
List of templates
The project concept
API documentation
Sitemap and web site structure
User manuals
Module/component catalogue
Content gathering decks
Data validation rules
Track your documentation: the meta data collection

Software Maintenance

Software Platform Development

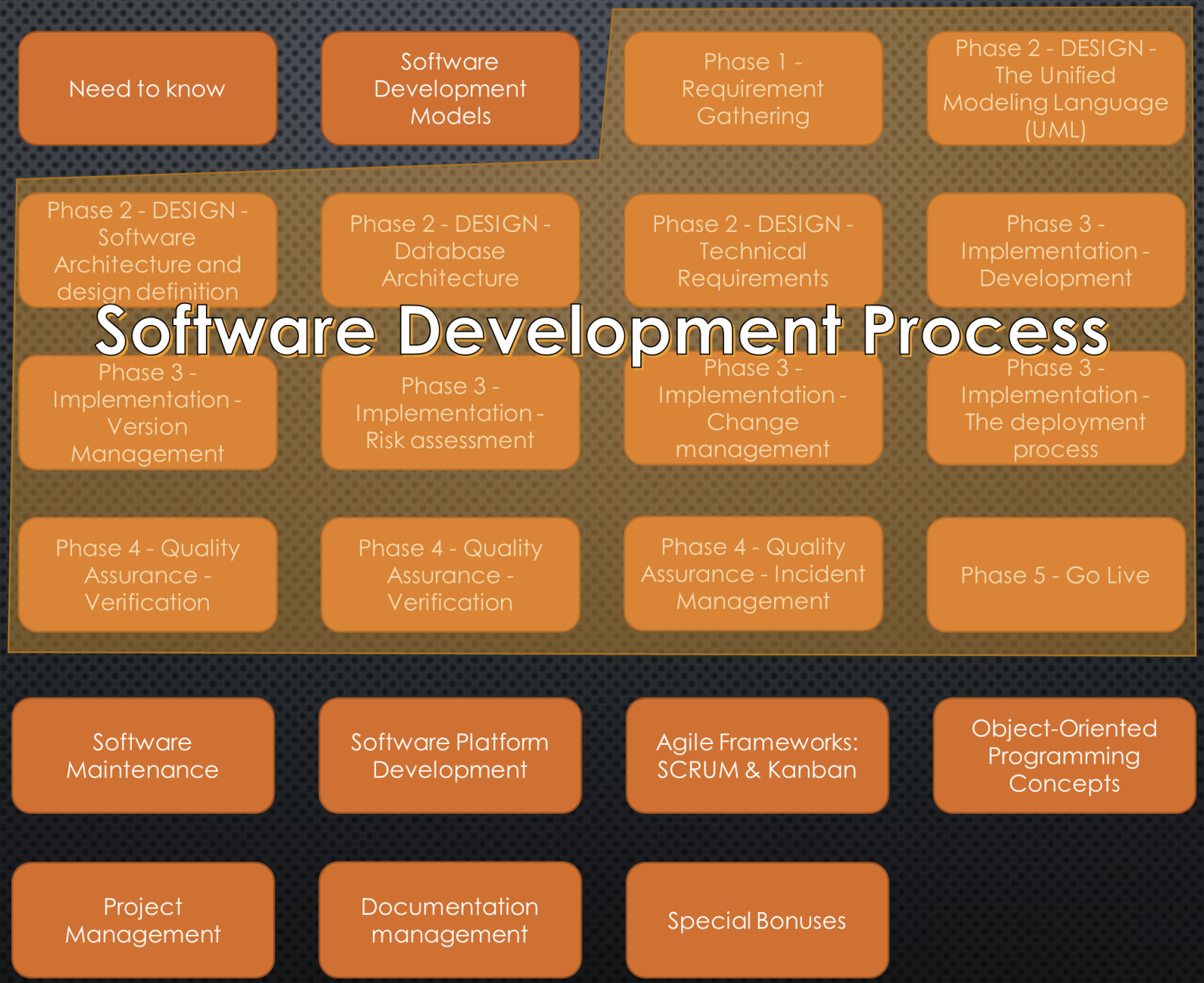
Agile Frameworks: SCRUM & Kanban

Object-Oriented Programming Concepts

Project Management

Documentation management

Special Bonuses



Software Development Process