



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

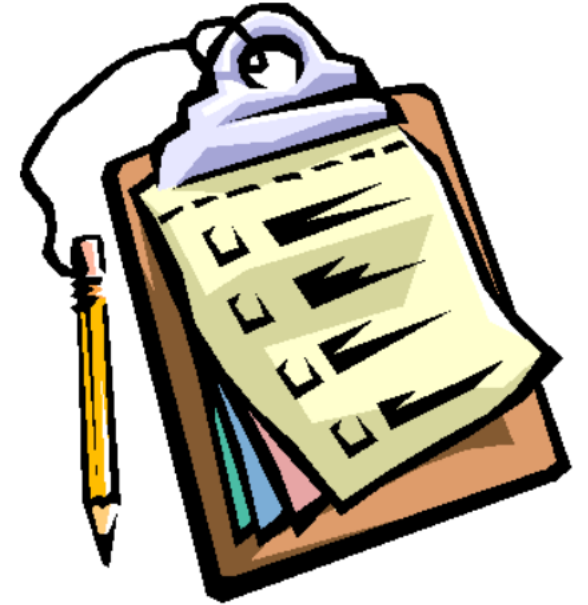
SOFTWARE ENGINEERING COURSE

Unified Process

Faezeh Gohari

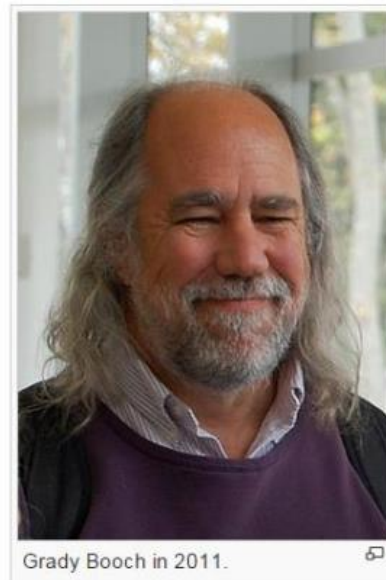
Agenda

- Overview of Unified Process
 - Building blocks
 - Phases, iterations and disciplines
 - UP Artifacts



UP Authors

- The first book to describe UP:
 - *The Unified Software Development Process (1999)*
 - Ivar Jacobson, Grady Booch and James Rumbaugh



Unified Process (UP)

- **Use Case Driven**
 - Successful system must build what users want
- **Architecture Centric**
 - Capture significant static and dynamic aspects of the system
 - Goals: understandability, reliance to future changes, and reuse
- **Iterative and Incremental**
- **Risk Focused**

UP Building Blocks

- **Roles (who)**

- A role defines a set of related skills, competencies and responsibilities
- E.g., Project manager, System Analyst, Software Architect, Technical Writer

- **Work products (what)**

- A work product represents something resulting from a task, including all the **documents** and **models** produced
- E.g., Software Architecture Document, Software Development Plan

- **Tasks (how)**

- A task describes a unit of work assigned to a Role that provides a meaningful result

UP Refinements and Variations

- **RUP**

- Rational Unified Process
- The IBM / Rational Software development process
- The best-known and extensively documented variation of UP

- **OpenUP**

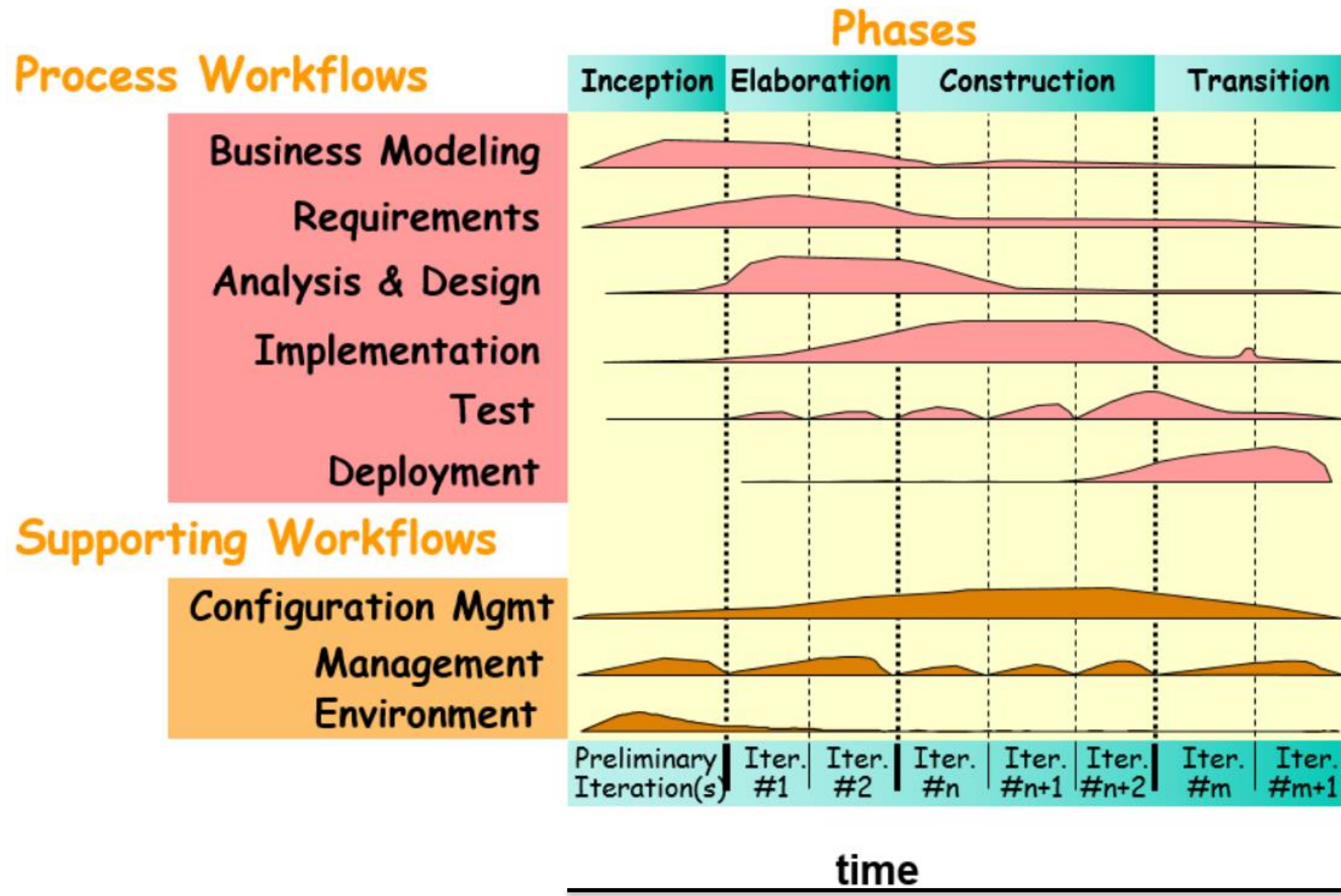
- Open Unified Process
- The Eclipse Process Framework software development process

- **AUP**

- Agile Unified Process
- A lightweight variation developed by Scott W. Ambler

- ...

UP Phases, Iterations, and Disciplines



UP Life-cycle

- Phases

- Four coarse-grained phases
- Each phase is finished before the start of the next phase

- Iterations

- Each phase is divided into iterations
- Usually 1 to 4 iterations per phase
- Iterations are also timeboxed.

- Disciplines (Workflows)

- Activities in different phases
- A discipline may continue in different phases

UP Phases

- Inception
 - Understand business case, identify use cases, feasibility, cost and planning
- Elaboration
 - Detailing of use cases for this iteration, refinement of system architecture (the skeleton)
- Construction
 - Build product (put meat on the skeleton)
- Transition
 - Delivery of final product and feedback
 - Customer/User tests and interaction
 - The ongoing support

Disciplines

Framework Activities

- Six "engineering disciplines"

- Business modelling
- Requirements
- Analysis and design
- Implementation
- Test
- Deployment

- Three supporting disciplines

- Configuration management
(change management)
- Project management
- Environment

Umbrella Activities

Inception phase

- By collaborating with stakeholders, basic business requirements for the software are identified.
 - Fundamental business requirements are described through a set of **preliminary use cases** that describe which major features and functions each group of users desires.
- A rough architecture for the system is proposed.
 - Architecture at this point is nothing more than a tentative outline of **major subsystems** and their functions and features.
- And a plan for the iterative and incremental nature of the **ensuing project** is developed.
 - Schedule, resources, major risks, ...

Example Questions in Inception

- What is the vision and business case for this project?
- Feasible?
 - Legal feasibility, Economic feasibility, Schedule feasibility, Cultural feasibility, Technical feasibility
- Buy and/or build?
 - Buy components and glue them together or from scratch?
- Estimate potential risks
- Rough estimate of cost: Is it \$10K-100K or in the millions?
- Should we proceed or stop?

Inception Phase Outcomes

- A vision document
 - A general vision of the core project's requirements, key features, and main constraints
- An initial **use-case model** (10% -20% complete)
- An initial project **glossary**
- An initial development case
 - Specifying the process to be used (especially, the artifacts to be produced for each discipline)

Inception Phase Outcomes (cont.)

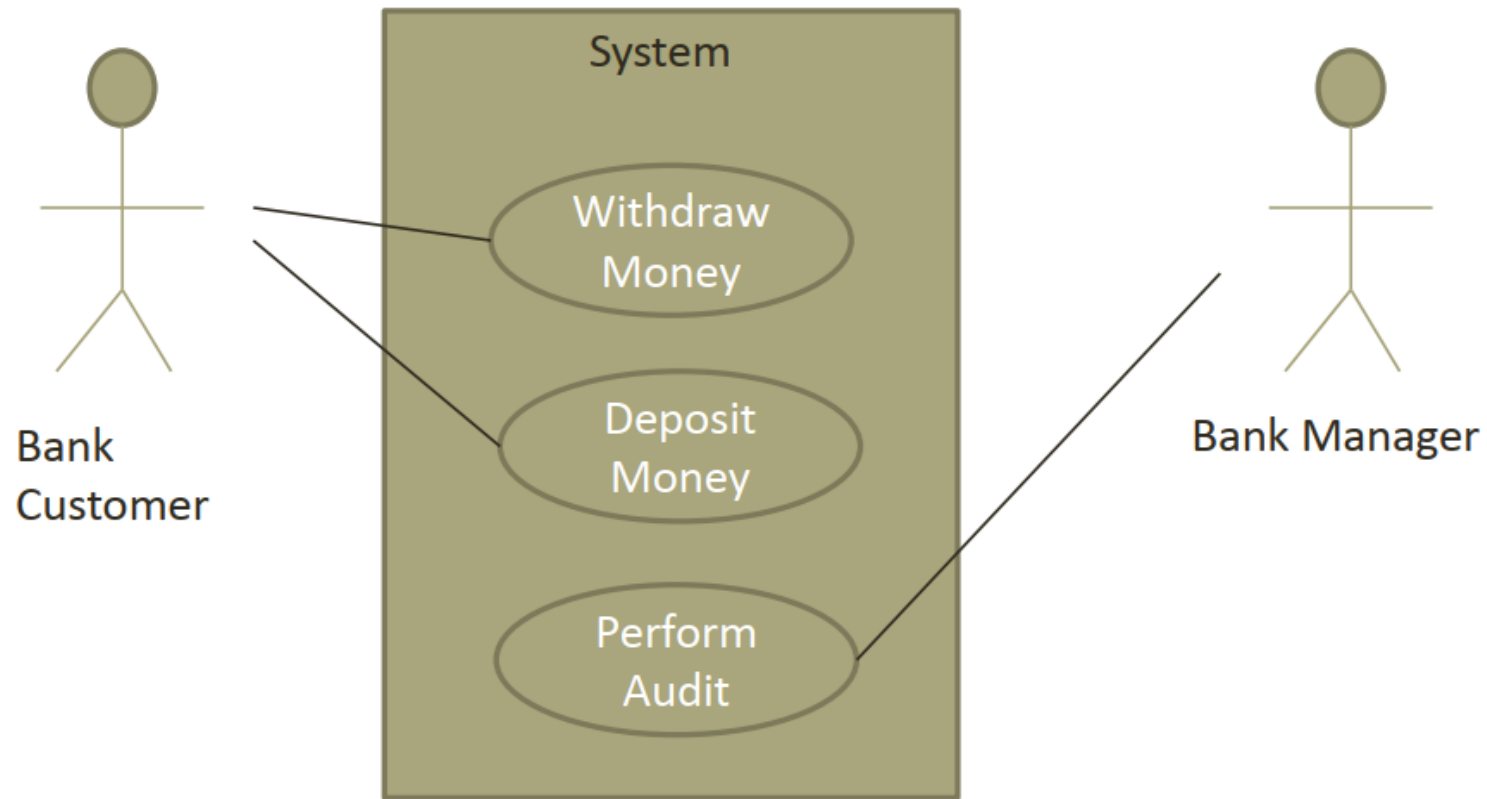
- An initial risk assessment
 - **Risk list and Risk Management Plan**
- A project plan, showing phases and iterations scheduling
 - **Software Development Plan (SDP)**
- An initial business case
 - Necessary information from a business standpoint to determine whether or not this project is worth investing in
 - Includes business context, financial forecast and success criteria (ROI, market recognition, and so on)
- One or several **prototypes**

Inception phase: Disciplines vs Artifacts

- The following table shows that each artifact of the inception phase is the output of which discipline.

Artifact	Discipline
Vision	Requirements
Use-case model	Requirements
Glossary	Requirements
Risk list	Project Management
Risk Management Plan	Project Management
Software Development Plan (SDP)	Project Management
Business case	Project Management
Development case	Environment

A Simple Use Case Model



Use-Case Specification

- A textual description detailing the sequence of events together with other related use case information in certain format
- For example:

Use Case Specification

Use Case Name:	Withdraw Cash
Actor(s):	Customer (primary), Banking System (secondary)
Summary Description:	Allows any bank customer to withdraw cash from their bank account.
Priority:	Must Have
Status:	Medium Level of details
Pre-Condition:	The bank customer has a card to insert into the ATM The ATM is online properly
Post-Condition(s):	<ul style="list-style-type: none">• The bank customer has received their cash (and optionally a receipt)• The bank has debited the customer's bank account and recorded details of the transaction

Use-Case Specification

Basic Path:

1. The customer enters their card into the ATM
2. The ATM verifies that the card is a valid bank card
3. The ATM requests a PIN code
4. The customer enters their PIN code
5. The ATM validates the bank card against the PIN code
6. The ATM presents service options including "Withdraw"
7. The customer chooses "Withdraw"
8. The ATM presents options for amounts
9. The customer selects an amount or enters an amount
10. The ATM verifies that it has enough cash in its hopper
11. The ATM verifies that the customer is below withdraw limits
12. The ATM verifies sufficient funds in the customer's bank account
13. The ATM debits the customer's bank account
14. The ATM returns the customer's bank card
15. The customer takes their bank card
16. The ATM issues the customer's cash
17. The customer takes their cash

Alternative Paths:

- 2a. Invalid card
- 2b. Card upside down
- 5a. Stolen card
- 5b. PIN invalid
- 10a. Insufficient cash in the hopper
- 10b. Wrong denomination of cash in the hopper
- 11a. Withdrawal above withdraw limits
- 12a. Insufficient funds in customer's bank account
- 14a. Bank card stuck in machine
- 15a. Customer fails to take their bank card
- 16a. Cash stuck in machine
- 17a. Customer fails to take their cash

Business Rules:

- B1: Format of PIN
- B2: Number of PIN retries
- B3: Service options
- B4: Amount options
- B5: Withdraw limit
- B6: card must be taken away before dispense of cash

Non-Functional Requirements:

- NF1: Time for complete transaction
- NF2: Security for PIN entry
- NF3: Time to allow collection of card and cash
- NF4: Language support
- NF5: Blind and partially blind support

RUP Template for Use-Case Specification

- RUP provides a standard template for recording the detailed information about use cases
- To see examples for a RUP project, refer to the link below:
https://sceweb.uhcl.edu/helm/RUP_course_example/courseregistrationproject/indexcourse.htm

1. Use-Case Name
 - 1.1 Brief Description
2. Flow of Events
 - 2.1 Basic Flow
 - 2.2 Alternative Flows
 - 2.2.1 < First Alternative Flow >
 - 2.2.2 < Second Alternative Flow >
3. Special Requirements
 - 3.1 < First Special Requirement >
4. Preconditions
 - 4.1 < Precondition One >
5. Postconditions
 - 5.1 < Postcondition One >
6. Extension Points
 - 6.1 <Name of Extension Point>

Elaboration phase

- This phase mainly focuses on **modeling activities** (i.e., analysis & design).
- Elaboration **refines and expands the preliminary use cases**.
- **Expands the architectural representation** to include five different views of the software:
 - **4+1 Architectural view model**
 - Use case view, logical view, process view, implementation view, deployment view
 - In some cases, elaboration creates an “**executable architectural baseline**”.
 - A “first cut” executable system
 - Does not provide all the required features and functions
- The **plan** is carefully **reviewed** at this phase to ensure that scope, risks, and delivery dates remain reasonable.
 - **Modifications to the plan are often made at this time.**

4+1 Architectural View Model

- Describing the architecture of software from different viewpoints:

1. Scenarios (use-case view)

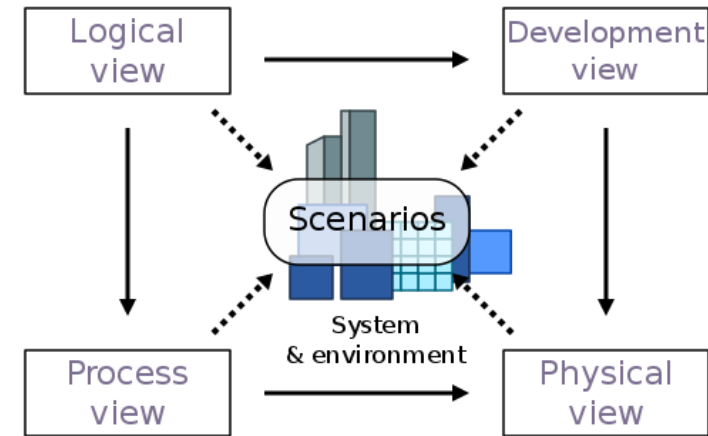
- The description of an architecture from end-users' view

2. Logical view

- The description of design model (i.e., classes, their responsibilities, relationships, ..)
- Represented by UML class diagrams

3. Process view

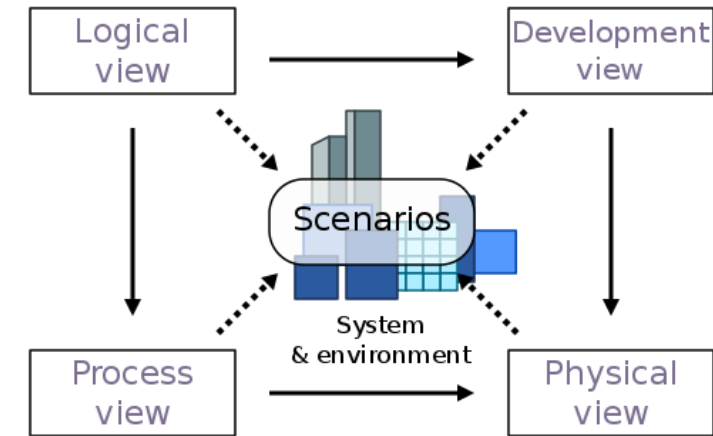
- The description of the dynamic aspects of the system (focuses on the run time behavior)
- Explains the system processes and how they communicate
- UML diagrams to represent process view include the sequence diagram, communication diagram, activity diagram



4+1 Architectural View Model (cont.)

4. Physical view

- Also known as implementation view
- The description of implementation model (i.e., components and subsystems)
- UML diagrams to represent physical view include package diagram and component diagram



5. Deployment view

- The description of deployment model
- Describes one or more physical network (hardware) configurations on which the software is deployed and run
 - Indicates the physical nodes (computers, CPUs) that execute the software, and their interconnections (bus, LAN, point-to-point, ...)
- Represented by UML deployment diagrams
- **Data view (optional):** A description of the persistent data storage perspective

Elaboration Phase Outcomes

- **Software Requirements Specification (SRS)**
 - **Use-case model** (at least 80% complete)
 - all use-cases and actors have been identified, and most use-case descriptions have been developed.
 - **Supplementary Specifications**
 - Non-functional requirements
- A software architecture description
 - **Software Architecture Document (SAD)**
- An **executable architecture**
 - The vertical slice



Elaboration Phase Outcomes (cont.)

- **A revised risk list and a revised business case**
- **A revised development plan for the overall project**
- **An updated development case**

Elaboration phase: Disciplines vs Artifacts

- The following table shows that each artifact of the elaboration phase is the output of which discipline.

Artifact	Discipline
Software Requirements Specification (SRS)	Requirements
Use-case model (updated)	Requirements
Supplementary Specifications	Requirements
Software Architecture Document (SAD)	Analysis & Design
Risk list (updated)	Project Management
Software Development Plan (updated)	Project Management
Business case (updated)	Project Management
Development case (updated)	Environment

Construction phase

- Using the architectural model as input, the construction phase develops or acquires the software components that will make each use case operational for end users. To accomplish this:
 1. **Analysis and design models** that were started during the elaboration phase are **completed** to reflect the final version of the software increment.
 2. All necessary and **required features and functions** for the software are **implemented** in source code.
 3. As components are being implemented, **unit tests** are designed and executed for each.
 4. In addition, **integration activities** (component assembly and integration testing) are conducted.
 5. **Use cases** are used to derive a suite of **acceptance tests** that are executed prior to the initiation of the next UP phase.

Construction phase (cont.)

- In addition, the software team creates the necessary **support information** that is required for the release
 - User manuals
 - Troubleshooting guides
 - Installation procedures
 - ...

Construction Phase Outcomes

- Software components
- The integrated software product
- Test plan and test cases

*Implementation and
Test Disciplines*

- Support documentation
 - The user manuals, installation manuals, ...
 - A description of the current release

Deployment Discipline

Transition phase

- Software is delivered to end users for **beta testing**
- User **feedback** reports both **defects** and **necessary changes**
- At the conclusion of the transition phase, the software increment becomes a **usable software release**

Transition Phase Outcomes

- Delivered software increment
- Beta test reports
- User feedbacks

Which major disciplines?

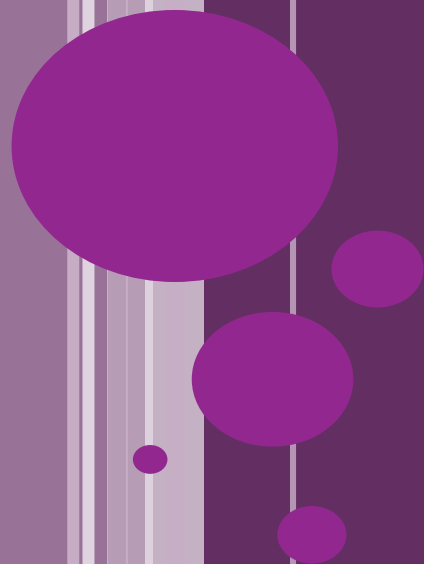
- 1. Deployment*
- 2. Test*
- 3. Implementation*

Adapt the Process

- UP is a process framework
- Not every task identified for a UP workflow is conducted for every software project
- Many decisions are dependent on the project conditions
 - Number of iterations in a phase
 - Amount of effort (time, ...) for a phase/iteration/artifact
 - The focus on documentations
- The **process engineer** should adapt the process for the target project
 - Adapting the process (actions, tasks, subtasks, and work products) to meet specific needs of the project

Further Reading

- Chapter 4 of Pressman
- Search Unified Process and RUP
 - Read some wikis
 - Follow the hyperlinks:
 - <https://scweb.uhcl.edu/helm/RationalUnifiedProcess/>
 - <https://www.ibm.com/support/pages/rational-unified-process-rup-plugins-rational-method-composer-751>



The End

RUP Artifacts Template: Vision

<Project Name>
Vision

Version <1.0>

[Note: The following template is provided for use with the Rational Unified Process. Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]

Revision History

Date	Version	Description	Author
<dd/mm/yy>	<x.x>	<details>	<name>

RUP Artifacts Template: Vision

Table of Contents

[1. Introduction](#)

- [1.1 Purpose](#)
- [1.2 Scope](#)
- [1.3 Definitions, Acronyms, and Abbreviations](#)
- [1.4 References](#)
- [1.5 Overview](#)

[2. Positioning](#)

- [2.1 Business Opportunity](#)
- [2.2 Problem Statement](#)
- [2.3 Product Position Statement](#)

[3. Stakeholder and User Descriptions](#)

- [3.1 Market Demographics](#)
- [3.2 Stakeholder Summary](#)
- [3.3 User Summary](#)
- [3.4 User environment](#)
- [3.5 Stakeholder Profiles](#)
 - [3.5.1 <Stakeholder Name>](#)
- [3.6 User Profiles](#)
 - [3.6.1 <User Name>](#)
- [3.7 Key Stakeholder or User Needs](#)
- [3.8 Alternatives and Competition](#)
 - [3.8.1 <aCompetitor>](#)
 - [3.8.2 <anotherCompetitor>](#)

[4. Product Overview](#)

- [4.1 Product Perspective](#)
- [4.2 Summary of Capabilities](#)
- [4.3 Assumptions and Dependencies](#)
- [4.4 Cost and Pricing](#)
- [4.5 Licensing and Installation](#)

[5. Product Features](#)

- [5.1 <aFeature>](#)
- [5.2 <anotherFeature>](#)

[6. Constraints](#)

[7. Quality Ranges](#)

[8. Precedence and Priority](#)

[9. Other Product Requirements](#)

- [9.1 Applicable Standards](#)
- [9.2 System Requirements](#)
- [9.3 Performance Requirements](#)
- [9.4 Environmental Requirements](#)

[10. Documentation Requirements](#)

- [10.1 User Manual](#)
- [10.2 Online Help](#)
- [10.3 Installation Guides, Configuration, and Read Me File](#)
- [10.4 Labeling and Packaging](#)

RUP Artifacts Template: Business Case

<Project Name>
Business Case

Version <1.0>

[Note: The following template is provided for use with the Rational Unified Process. Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]

Revision History

Date	Version	Description	Author
<dd/mmm/yy>	<x.x>	<details>	<name>

Table of Contents

- 1. [Introduction](#)
 - 1.1 [Purpose](#)
 - 1.2 [Scope](#)
 - 1.3 [Definitions, Acronyms and Abbreviations](#)
 - 1.4 [References](#)
 - 1.5 [Overview](#)
- 2. [Product Description](#)
- 3. [Business Context](#)
- 4. [Product Objectives](#)
- 5. [Financial Forecast](#)
- 6. [Constraints](#)



RUP Artifacts Template: Development Case

<Project Name> Development Case

Version <1.0>

[Note: The following template is provided for use with the Rational Unified Process (RUP). Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]

Table of Contents

Introduction

- Purpose
- Scope
- Definitions, Acronyms, and Abbreviations
- References
- Overview

• Overview of the Development Case

- Lifecycle Model
- Disciplines
- Discipline Configuration
- Artifact Classification
- Review Procedures
- Sample Iteration Plans

• Disciplines

- Business Modeling
- Requirements
- Analysis & Design
- Implementation
- Testing
- Deployment
- Configuration & Change Management
- Project Management
- Environment

• Roles