Object-Oriented Analysis and Design using JAVA

B.Tech (CSE/IT) 5th SEM
2021-2022

Lecture-7: Rational Unified Process (RUP)

Introduction

- The Rational Unified Process is unique in that it allows development teams to recognize the full benefits of the Unified Modeling Language (UML), software automation, and other industry best practices.

- The Rational Unified Process unifies the entire software development team and optimizes the productivity of every team member by putting the collective experience derived from thousands of projects and many industry leaders at your fingertips.

- Adaptable methodology for Object-Oriented Software production. It is based on the Iterative and Incremental Iterative model. Each successive version is intended to be closer to its target than its predecessor.

Referred sources mentioned in Key References

Waterfall vs. Unified Process

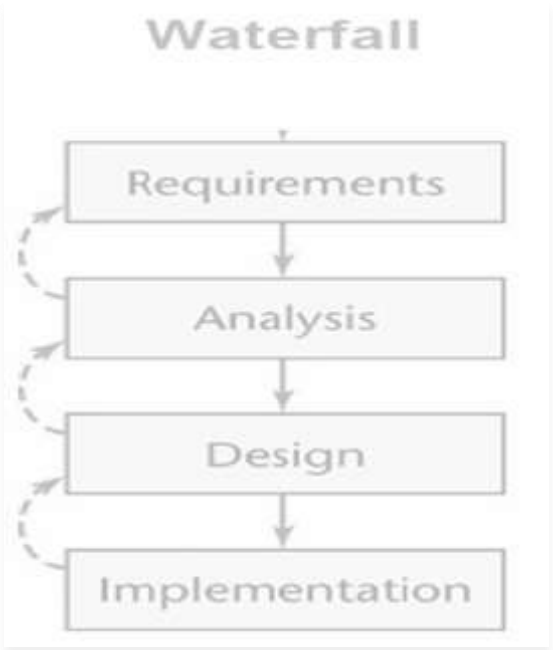
Waterfall

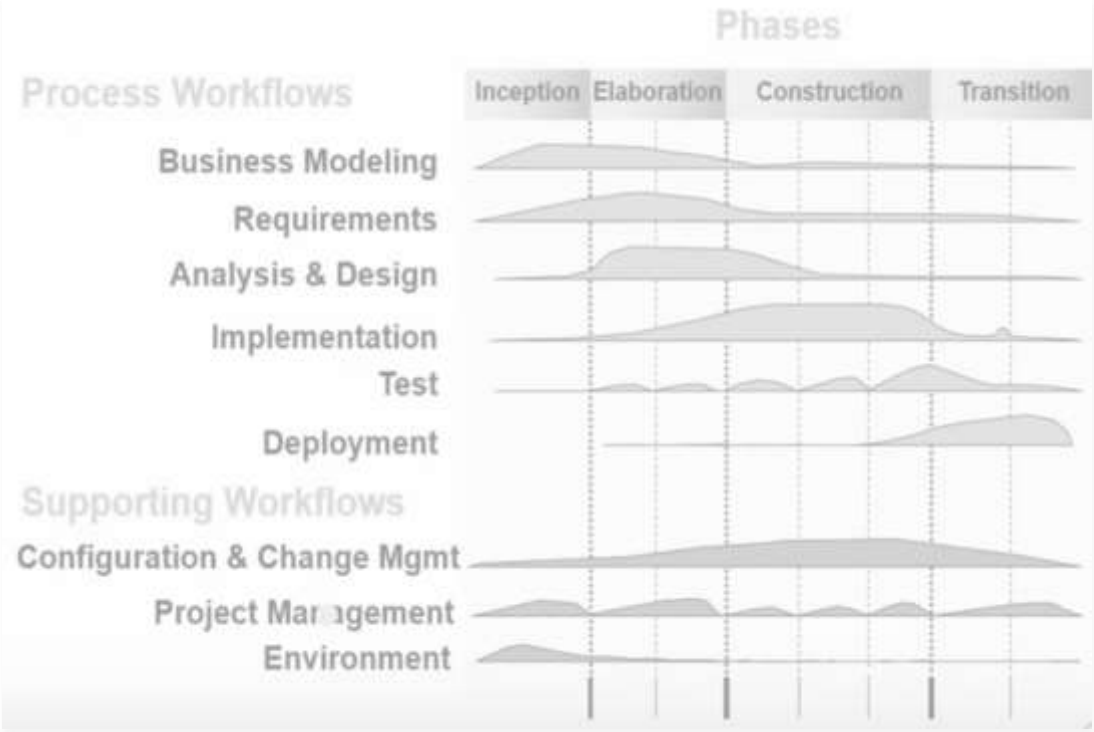
- One dimensional
- Each phase must be completed before you begin the next phase

Unified Process

- Incremental and Iterative
- Two dimensional
- Task is divided into **increments**(phases)
- Within each increment the developers have to **iterate**(workflows) until task is complete
- Consecutive series of waterfall models

Referred sources mentioned in Key References





Unified Process is based on the enlargement and refinement of a system through multiple iterations, with cyclic feedback and adaptation. The system is developed incrementally over time, iteration by iteration, and thus this approach is also known as iterative and incremental software development. The iterations are spread over four phases where each phase consists of one or more iterations

Referred sources mentioned in Key References

Key feature of RUP

The key feature: Software development is done in a series of fixed periods, for example, between 2 and 6 weeks. Each period is called as iteration.

- At the end of each iteration, we have an executable system.
- Each iteration has its own requirement analysis, design, coding and testing.
- The software development is incremental. Implement new features added and the user's suggested changes.
- Each iteration is timeboxed i.e. fixed in length. Each iteration to be between 2 and 6 weeks.
- It has no provision for the extension of an iteration period. If software developers are unable to complete coding the given requirements within fixed time, then decrease the number of requirements to code.

Referred sources mentioned in Key References

Phases of RUP

- The **inception** phase: 'Vision' of the end-product and the associated business case, overall scope of the project are defined.

- The **elaboration** phase: The refinement of the definition of the product, define and baseline an architecture, and develop a more precise plan for its development and deployment.

- The **construction** phase: where the product is built, up to the point where it can be put in the hands of its end-users for the first time.

- The **transition** phase: where the product is transitioned to the user community; this includes manufacturing, delivering, training, supporting and maintaining the product.

Each phase and its iteration consists of a set of predefined activities. The Unified Process describes work activities as disciplines—a discipline is a set of activities and related artifacts in one subject area (e.g., the activities within requirements analysis). The disciplines described by Unified Process are as follows:

- *Business modeling*—domain object modeling and dynamic modeling of the business processes,
- *Requirements*—requirements analysis of system under consideration. Includes activities like writing use cases and identifying nonfunctional requirements,
- *Analysis and design*—covers aspects of design, including the overall architecture,
- *Implementation*—programming and building the system (except the deployment),
- *Test*—involves testing activities such as test planning, development of test scenarios, alpha and beta testing, regression testing, acceptance testing, and
- *Deployment*—the deployment activities of developed system.

Referred sources mentioned in Key References

Inception phase

1. The idea for the project started and transformed into vision of end product. Determining the scope of your system.
2. Identification of the people, organizations, and external systems that will interact with your system.
3. Developing an initial risk assessment, schedule, and estimate for your system. Describing the initial requirements
4. This phase determines the following: i) major users (actors)-use case model, ii) tentative software architecture, iii) rough planning, budget, risk identification.

Referred sources mentioned in Key References

Outcomes of inception phase

1. Vision document with general vision of the core project's requirements, key feature, and main constraints
2. Higher level use-case model describing the project from user perspective.
3. Various glossary related to project with their clear meanings.
4. Initial business case where the reasoning for initiating the project is captured.
5. Initial project plans and risk assessment. One or more prototypes

Referred sources mentioned in Key References

Evaluation of the inception phase

1. Stakeholder concurrence on scope, definition, and cost/schedule estimate. Credibility of the cost/schedule estimate, priority, risks, and development process. Depth and breadth of any prototype that was developed
2. If the inception phase lasts for too long, it is like an indicator stating that the project vision and goals are not clear to the stakeholders.
3. With no clear goals and vision the project most likely is doomed to fail. At this scenario it is better to take a pause at the very beginning of the project to refine the vision and goals.
4. Otherwise it could lead to unnecessary make-overs and schedule delays in further phases

Referred sources mentioned in Key References

Elaboration phase

1. During this phase the project team is expected to capture a majority of system's requirements (e.g., in the form of use cases).

1. Analyze problem domain to get better understanding.

1. Perform risk analysis and make a plan of risk management to reduce or eliminate their impact on final schedule and product.

1. Establish design and architecture (e.g., using basic class diagrams, package diagrams, and deployment diagrams, i.e., establish a solid architectural foundation)

1. Create a plan (schedule, cost estimates, and achievable milestones) for the next (construction) phase. Refine plan of actions and estimates

Outcome of elaboration phase

- A use-case model (80% complete) - all use cases having been identified in the use-case model survey, all actors having been identified, and most use-case descriptions (requirements capture) having been developed.

- Supplementary requirements capturing the non functional requirements and any requirements that are not associated with a specific use case.

- An executable architecture and accompanying documentation -the **Software Architecture Document**, including use-case descriptions (design) for a subset of use cases (use-case view), and an updated glossary.

- A revised business case, revised risk list, and a development plan for the overall project, including the coarse-grained project plan, showing iterations and evaluation criteria for each iteration. A preliminary user manual (optional).

Referred sources mentioned in Key References

Evaluation of the elaboration phase

- Is the vision of the product stable? Is the architecture stable? Does the executable demonstration show that the major risk elements have been addressed and credibly resolved?
- Is the plan for the construction of sufficient detail and fidelity, and is it backed up with a credible basis of estimates?
- Do all stakeholders agree that the current vision can be met if the current plan is executed to develop the complete system, in the context of the current architecture?
- Are actual resource expenditure versus planned expenditure acceptable? Project may be cancelled or considerably rethought if it fails to pass this milestone

Construction phase

1. Longest and largest phase within Unified Process. During this phase, the design of the system is finalized and refined and the system is built using the basis created during elaboration phase (i.e., all features considered developed).

1. The construction phase is divided into multiple iterations, for each iteration to result in an executable release of the system. The final iteration of construction phase releases fully completed system which is to be deployed during transition phase.

Referred sources mentioned in Key References

Outcome of the construction

1. The construction phase produces a product which is ready to be deployed to the end user.

1. At the end of this phase, all the use cases defined in the earlier phases are realized, with traceability information. That is all the functionality express by the use cases are implemented and traced through different artefacts.

1. Production of complete software product and integrated on adequate plate forms.

1. Thoroughly system tests results, Ready user manual, Complete set of artefacts: design, code, test cases

Referred sources mentioned in Key References

Evaluation of construction

1. Whether the product is stable/mature enough to be deployed to users
2. All the stakeholders are satisfied with their expectation and ready for the transition of the system to move into the user community
3. Assessment of the project expenditure. Are resources expenditures versus planed expenditure still acceptable

Referred sources mentioned in Key References

Transition phase

1. The purpose of the transition phase is to transition the software product into the user community. Once the product has been given to the end user, issues usually arise that require you to develop new releases, correct some problems, or finish some of the features that may have been postponed.

1. The final project phase which delivers the new system to its end-users. Transition phase includes also data migration from legacy systems and user trainings.

1. Issues after deployment, New release, Training customer service and producing help line assistance, A new cycle may start.

Referred sources mentioned in Key References

Outcome of transition phase

1. At the end of the transition phase, All the artefacts related to the software project which were mentioned in project life cycle must be completed.

1. The complete product delivered in the hand of the user and can be used and interacted by the user.

1. Lesson learnt- what worked, what didn’t work, and what can be done different thing in the next release of the cycle.

1. Plan for next release- Planning for the next iteration based on the feedback and lesson learnt.

Referred sources mentioned in Key References

Evaluation of the transition phase

1. In this phase the most important assessment is, whether the user is satisfied with the complete product or not. That is whether the product make the user happy or not.

1. Are actual resourced expenditure versus planned expenditure still acceptable

Referred sources mentioned in Key References

Key references

1. Software Designing With Unified Modeling Language Driven Approaches. *Janis Osis, Uldis Donins, in Topological UML Modeling, 2017.* Agile Data Warehousing for the Enterprise 1st Edition.

1. The Rational Unified Process: An Introduction (3rd Edition) By-Philippe Kruchten, Publisher: Addison Wesley.

1. Georgia Tech - Software Development Process-https://www.udacity.com/course/ud805

1. Concepts: Project Lifecycle, Phases and Major Milestones----------
http://www.interface.ru/rational/rup51/process/workflow/manageme/co_phase.htm

Thank You