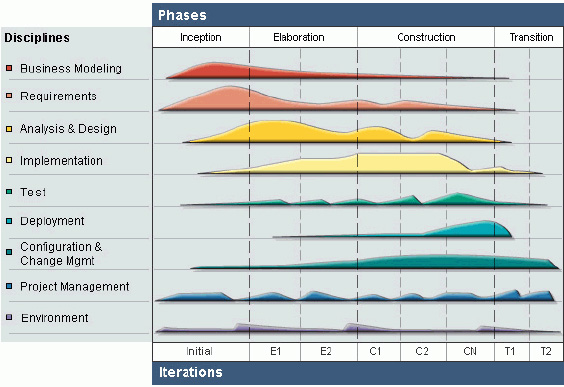
Our project will be developed by Rational Unified Process model.

The Rational Unified Process (RUP) is an iterative software development process framework created by the Rational Software Corporation, a division of IBM since 2003. RUP is not a single concrete prescriptive process, but rather an adaptable process framework, intended to be tailored by the development organizations and software project teams that will select the elements of the process that are appropriate for their needs. RUP is a specific implementation of the Unified Process. (<http://en.wikipedia.org/wiki/IBM_Rational_Unified_Process>).



Rational Unified Process Model overview

The RUP model has four phases: Inception, Elaboration, Construction and Transition. Besides, it has six engineering disciplines: Business Modelling, Requirements, Analysis and Design, Implementation, Test, Deployment and three supporting disciplines: Configuration and Change Management, Project Management and Environment.

RUP have six engineering disciplines and three supporting discipline:

**Business modeling:**

In Business Modeling we document business processes using so called business use cases. This assures a common understanding among all stakeholders of what business process needs to be supported in the organization.

**Requirements:**

\_ Translation of the business model to functional and non-functional requirements, descriptions of what the system is expected to do.

**Analysis and design:**

\_ Decription how the system will be realized in the implementation phase.

 mô tả kiến trúc hệ thống thông qua các sơ đồ phân tích thiết kế.

**Implementation:**

\_ Implementation of the design, unit tests and integration of components into executable systems

**Test:**

**\_** To identify and ensure defects are addressed prior to the deployment of the software.

**Deployment:**

**\_** Production of product releases, and delivery of them to end-users. Provision of support and migration help.

**Three supporting disciplines:**

**Configuration and change management:**

Management and adjust changing of project requirement.

**Project management:**

Management of risks to the project and successful delivery of a product.

**Environment:**

**\_** Provision of tools to a software project and adaptation of RUP to the specific project.

RUP is based on a set of building blocks, or content elements, describing what is to be made, require the necessary skills and explains the step to describe how specific development aim to be achieved. The main building blocks are the following:

**+ Workers (who)**:defines the behavior and responsibilities of an individual, or a group of individuals working together as a team.

+ **Artifacts (what)**: An artifact is a piece of information that is produced, modified, or used by a process. Artifacts are the tangible products of the project, the things the project produces or uses while working towards the final product.

+ **Activities (how)**:An activity of a specific worker is a unit of work that an individual in that role may be asked to perform.

+ **Workflows (when)**: is a sequence of activities that produces a result of observable value

**Project lifecycle:**

Project life cycle of RUP model has four phases:

**Inception Phase**: Khởi động (inception) Inception is the first phase of process. In this startup phase, we should provide business case of the system and determine the scope of project. Besides, we have to create the project management plan that has project schedule, effort estimation and risk management etc. At the end of this phase, we should check the objectives of project and decide whether to continue development or not. Hence, Inception phase must be properly planned and done. Based activities of this phase:

- Study business case and feasibility study of project

- Complete draft ERD of system

- Complete draft screen prototypes

- Complete draft requirements

- Determine project scopes

- Complete project management plan

**Elaboration Phase**: The objectives of this phase are to determine appropriate architectural and construction plan for the project. The architectural decision needs to be made for the entire system, and to describe most of the requirements of system. At the end of this phase, we must examine the objectives and scopes, the choice of architecture and decide whether to proceed to the next phase. Based activities of this phases:

- Complete user requirement specification

- Complete ERD, final prototypes

- Complete Software Requirement Specification

- Complete database model

- Complete System Architecture Design

**Construction Phase**: Construction is the third phase of RUP lifecycle. In this phase,

we must have done all the coding and testing work. After coding, developers will do unit test themselves, then test team will do functional test and regression test when finishing all. Based activities of this phase:

- Complete coding and unit test

- Complete functional and regression test

- Complete user manual

**Transition Phase**: Transition is the final phase of the RUP lifecycle. In this phase, project team has to deploy the application and give it to users. The next step is receiving feedback from users to identify the problems and then complete the system. Based activities of this phase:

- Deploy the system

- Deliver source code

- Complete all reports and documents

**Why RUP?**

### Develop iteratively: Because the team is not much experience in developing sofware so while develop will occur many problem we use the RUP model will help the team can solve the problem at each component of system not affect other component and do not have to redo all work. We just have to develop iteratively the component which have problem to improve it. We did not choose the Water Fall model because if the team have problem at the requirement analysis we have to redo all design and other work that make a lot time.

**Manage requirements:** Because the team not well understand about the Law firm management so we can clarify the requirement at the first phase so while developing the project we may detect and improve the requirement to make the system better.

**Use components:** We use the RUP model to break down the large project into many small components so we can test individual components before they are integrated into a large system. Some member is not good in coding he can use the component which is developed by good developer to consult. Also some coding block can reuse for other components.

**Model visually:** Use diagrams to represent all major components, users, and their interaction. "UML", short for [Unified Modeling Language](http://en.wikipedia.org/wiki/Unified_Modeling_Language), is one tool that can be used to make this task more feasible.

**Ensure quality:** The team can focus on a component to test it to ensure the component’s quality is the best before switch to other component. Always make testing a major part of the project at any point of time. Testing becomes heavier as the project progresses but should be a constant factor in any software product creation.

**Difference between RUP and SCRUM methodologies**

|  |  |  |
| --- | --- | --- |
|  | **RUP** | **SCRUM** |
| Approach | Iterative | Iterative |
| Cycle | Formal Cycle is defined across 4 phases, but some workflows can be concurrent. | Each sprint (iteration) is a complete cycle. |
| Planning | Formal project plan, associated with multiple iterations, is used. The plan is end-date driven and also has intermediate milestones. | No end-to-end project plan. Each next iteration plan is determined at the end of the current iteration (NOT end-date driven). Product Owner (Key Business User) determines when the project is done. |
| Scope | Scope is predefined ahead of the project start and documented in the Scope document. Scope can be revised during the project, as requirements are being clarified, but these revisions are subject to a strictly controlled procedure. | Instead of scope, SCRUM uses a Project Backlog, which is re-evaluated at the end of each iteration (sprint). |
| Artifacts | Vision/Scope Document, Formal functional requirements package, system architecture document, development plan, test plan, test scripts, etc. | The only formal artifact is the operational software. |
| Type of Project/Product | Recommended for large, long- term, enterprise-level projects with medium-to-high complexity. | Recommended for quick enhancements and organizations that are not dependent on a deadline. |