E-ISSN:2540-9719
ISSN:2302-8149

SIMULATION MODEL FOR *RATIONAL UNIFIED PROCESS* (RUP) SOFTWARE DEVELOPMENT LIFE CYCLE

[1]Tuti Karen Tia, [2]Ilyas Nuryasin, [3]Maskur
[1 2] Faculty of Engineering, University of Muhammadiyah Malang
246 Tlogomas Raya Street Landungsari, Malang, East Java 65144
Email: tiyakaren@gmail.com, ilyas@umm.ac.id

ABSTRACT

In software development there are several types of methodologies that can be used, one of which is Rational Unified Process (RUP). RUP is of software development process used in various software companies. There are various types of projects in RUP software development, small scale project, large-scale projects and re-engineering projects. Each type of project has different needs, especially in the use of each role. In RUP there are several important roles that cannot be done by an individual because they have to work at the same time. Basically the purpose of the RUP is to be able to produce high quality software. This cannot be achieved if the inappropriate role is used. This study proposes the number of roles needed for software development using RUP by creating simulation models. The purpose of simulation model is to determine the appropriate role based on each type of project. The results of the simulation model are in the form of a percentage of each role needed, the project manager has the largest percentage compared to other roles, which is 70% in small-scale projects and 65% in large-scale projects.

Keywords**:** Rational Unified Process (RUP), Simulation Model, Project Management, Simphony.NET.

1 INTRODUCTION

Software development is associated with design, writing, testing, implementation and maintenance. It forms the software development process as a development cycle or called as SDLC. SDLC is a process or methodology for project development that has certain stages in accordance with the project goals and objectives. The main objective of SDLC is to provide high quality products [1]. There are several development models, each model describes an approach to activities that take place during development. The selection of the SDLC model has a very high impact on the software produced. Therefore, it is important to choose the SDLC model that fits the needs of the project.

One of the SDLC models discussed in this study is the Rational Unified Process (RUP). RUP is one of the repeated software development models, which provides tasks and responsibilities within an organization to ensure the production of high-quality software, which means it can meet the needs of users with predictable schedules and budgets [2]. In the RUP team members involved in the project are guided to follow each step based on each element. There are three basic elements in the RUP framework, namely activities, roles and artifacts. The selection of the three elements must be adjusted to the software project. Each project contains a group of team members who are in charge of one or more roles. Each role participates in one or more activities and each activity produces one or more artifacts [3].

RUP is used by various companies and various industrial sectors, especially the computer software industry and information technology service industry. RUP is also adopted by small industries that have employees under 50 people [4]. In developing the software industry, RUP supports the use of small teams or large teams. There are various sizes of software development projects, namely small scale projects, medium scale projects and large scale projects [5]. Software development does not always have the same scale. So that when planning project development is not done properly, it will result in the failure of the software produced. Especially in the use of RUP, the implementation of many frameworks and roles that are needed is overlooked because it is implemented by small teams [6]. There are some disadvantages of RUP related to human resource management, communication management and contract management [7].

Based on the problems mentioned above, project planning is needed to determine the number of each role in the RUP. Therefore, this study proposes a simulation model for project development using

RUP. Simulations are carried out to determine the appropriate role for each type of project without ignoring the important functions of the RUP. Making a simulation model is done on a computer using a simulator tool, Simphony.NET [8].

2 STUDY OF LITERATURE

2.1 Related Work

Sonia Thind and Karambir [3] proposed a simulation model for software development using Spiral by using a computer simulation tool called Simphony.NET, the study produced a simulation model that could help project managers to increase company productivity by using a minimum amount of resources. This has been done by making all team members busy all the time so that no time is wasted by the standby team or the team waiting for the other stages is finished. Youssef Bassil [7] made a simulation model for software development using Waterfall, which aims to facilitate project managers in determining how to achieve optimal productivity with minimum use of resources. Amninder Singh and Puneet Jai Kaur [3] created a simulation model with the aim of providing resources to the project along with time constraints. The results of the simulation model made using 10 companies with various types of projects showed that a number of team members worked on several projects in an associative manner.

2.2 Rational Unified Process (RUP)

Rational Unified Proess (RUP) is one of the software development products offered by IBM Rational. RUP is serialized on a large scale, repeats on a small scale and produces gradual releases over time [9]. This development model is used by various companies and various industrial sectors. In the development of software industries that support the use of small teams or large teams.

RUP uses an iterative or iterative approach in the order of additional steps. Each iteration covers part or most of the development phase. Each sequential iteration is built above from the previous iteration results to improve the system until the final product is finished. This repetitive process allows team members to play many roles, besides that project managers can better use available teams so that they can broaden the scope of team members' expertise [10].

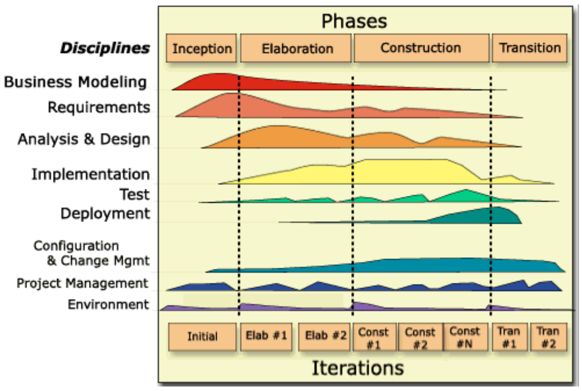

Figure 2.1. Development Phase of Rational Unified Process (RUP) [10]

Rational Unified Process (RUP) is one of the software development products offered by IBM Rational. RUP is serialized on a large scale, repeats on a small scale and produces gradual releases over time [9]. This development model is used by various companies and various industrial sectors. In the development of software industries that support the use of small teams or large teams.

RUP uses an iterative or iterative approach in the order of additional steps. Each iteration covers parts or most of the development phase. Each sequential iteration is built from the previous iteration results to improve the system until the final product is finished. This repetitive process allows the team members to play many roles, besides that project managers can better use available teams that they can broaden the scope of the team members' expertise [10].

a) Inception

Inception is the stage to determine the scope by identifying all extraneous entities that will interact with the system. The results of this phase are document requirements and usecase models.

b) Elaboration
Elaboration is the stage for analyzing and understanding the whole system, then being built into architecture. The result of this phase is the use-case model, an executable architecture prototype.

c) Construction
Construction is the stage that contains the design implementation produced in the previous phase (elaboration). The results of this phase are software products and user guides.

d) Transition
Transition is the phase to release software. The results of this phase are the activities needed to place software on the user.

The phase is run sequentially. Work is grouped into logical activities, namely disciplines carried out iteratively throughout the phase [9], the discipline is depicted in static or vertical dimensions. Static structure consists of workflows, among others, Business Modeling, Requirements, Analysis and Design, Implementation, Testing, Deployment, Configuration and Change Management, Project Management and Environment. The framework allows software development organizations of all sizes to work.

2.3 Simulation

Simulation is a process to represent real conditions with a simpler form using computer software. Simulation uses mathematical descriptions, construction constraints, computer algorithms which are generally incorporated into simulation software models to represent the actual system [5].

2.4 Simphony.NET 4.6

Simphony.NET or can be called Simphony Modeling Environment is one of Microsoft.NET products that is used to create user interface simulation and modeling models. Symphony.NET provides media for making general-purpose simulation models based on designs so that they can be used to model any process [5].

3 RESEARCH METHOD

3.1 Problem Definition

The software development model used by most teams is the Waterfall model, where each phase is completed in strict order, analysis and design, then implementation and testing. This approach made the first team members not work because they had to wait for the team at other stages to complete the task [10].

Software development for each project is different and the process that is followed must be adjusted to meet their needs. RUP uses a disciplined approach to provide tasks and responsibilities for software development that aims to ensure that the products produced are high quality products, besides that the products produced using RUP must be able to meet the needs of end users in the predicted schedule and budget [10]. Therefore, determining the initial planning of the project must be done properly. The purpose of making this simulation model is to find out the appropriate role for each type of project.

3.2 Simulation Model

The simulation model created is a software development simulation model using the Rational Unified Process (RUP). Making a simulation model uses Simphony.NET tools, in which there are several complete libraries along with the elements needed for making simulation models. In RUP there are four phases and nine workflows in each phase, each workflow will be placed on the Task element, then linked to each resource according to each task. All elements that are created will be connected to each other and then configured. Simulation is carried out for five times using 50 number of projects coming. After that the model is verified and executed until it gets the desired results. The results of the simulation process can be statistical reports and graphs that can be used for project development planning.

3.3 Assumtions and Specification

Model simulation for Rational Unified Process (RUP) uses several assumptions, because the simulation model built is stochastic or the value cannot be determined exactly in the form of a range. One of them is making the assumption that the project comes randomly to a software company. The project can be classified into two parts based on complexity and scale, including: small-scale projects and large-scale projects.

The next assumption is to determine the number of each project, which is 70% for small-scale projects and 30% for large-scale projects. The time of arrival of the project can be determined using the triangular distribution formula. By determining the minimum time limit for project arrival is 10 days, the maximum time limit for project arrival is 50 days and the average time of arrival of the project is 30 days.

Next is to determine the duration of work in each phases in the table in the following table 3.1.

Table 3.1. Duration of Work on Every Small-Scale Project Workflow

Fase	Business Modeling	Requirements	Analysis & Design	Implementation	Test	Deployment
Inception	Uniform (5,7)	Uniform (3,5)	Uniform (3,5)	Uniform(1,2)	Uniform (2,3)	Uniform (1,2)
Elaboration	Unform (5,7)	Uniform (7,9)	Uniform (10,15)	Uniform (3,4)	Uniform (2,3)	Uniform (1,2)
Construction	Uniform (1,2)	Uniform (2,3)	Uniform (3,5)	Uniform (10,15)	Uniform (8,9)	Uniform (2,3)
Transition	Uniform (1,2)	Uniform (1,2)	Uniform (1,2)	Uniform (3,5)	Unform (5,7)	Uniform (10,15)

The type of project used as an input parameter in the making of a simulation model is the Ganymede Project, a small-scale project and the Mars Project, a large-scale project.

A. Small Scale Projects (*Ganymede Project*)

It is called a small project because the software produced is small and requires a short amount of time for its construction. The initial development cycle of new software where everything including its requirements must be designed from the start. The Ganymede project only requires one iteration. Small-scale projects have one iteration at each phase:

Table 3.2. The number of iterations in the Ganymede Project

Phase	Number of Iteration	Explanation
Inception	1	Iteration to produce a proof-of-concept prototype or mock-up user interface
Elaboration	1	Iteration to produce architectural prototypes
Construction	1	Iteration to build products (to beta release)
Transition	1	Iteration to complete the product (full product release)

B. Large Scale Projects (*Mars Project*)

Called a big project because the software produced is more complex and has never been built before. So that it takes longer to build.

Table 3.2. The number of iterations on the Mars Project

Phase	Number of Iteration	Explanation
Inception	1	Possibility of producing a prototype
Elaboration	2	1 iteration to develop an initial architectural prototype 1 iteration to develop the initial baseline prototype
Construction	2	1 iteration to make it mature for beta testing 1 iteration to describe the partial system
Transition	1	Starting from beta release to full product

4 RESULT AND DISCUSSION

4.1 The Simulation Model For Rational Unified Process (RUP)

The simulation model was built using a computer simulation tool, Simphony.NET [8]. The Simphony library used to make the proposed simulation model consists of resources, files, capture, release, task, destroy, create, counter, and probabilistic branches. Figure 4.1. describe elements of resource modeling with the number of queues, which consist of: Small scale projects: project manager, project reviewer, implementer, administrator, process engineer, test manager, system tester and system administrator. Then for large-scale projects in Figure 4.2: project managers, integrators, project reviewers, system analysts, process engineers, implementers, software architect, system administrators, test managers, user-interface designers, course developers, database designers and system testers.

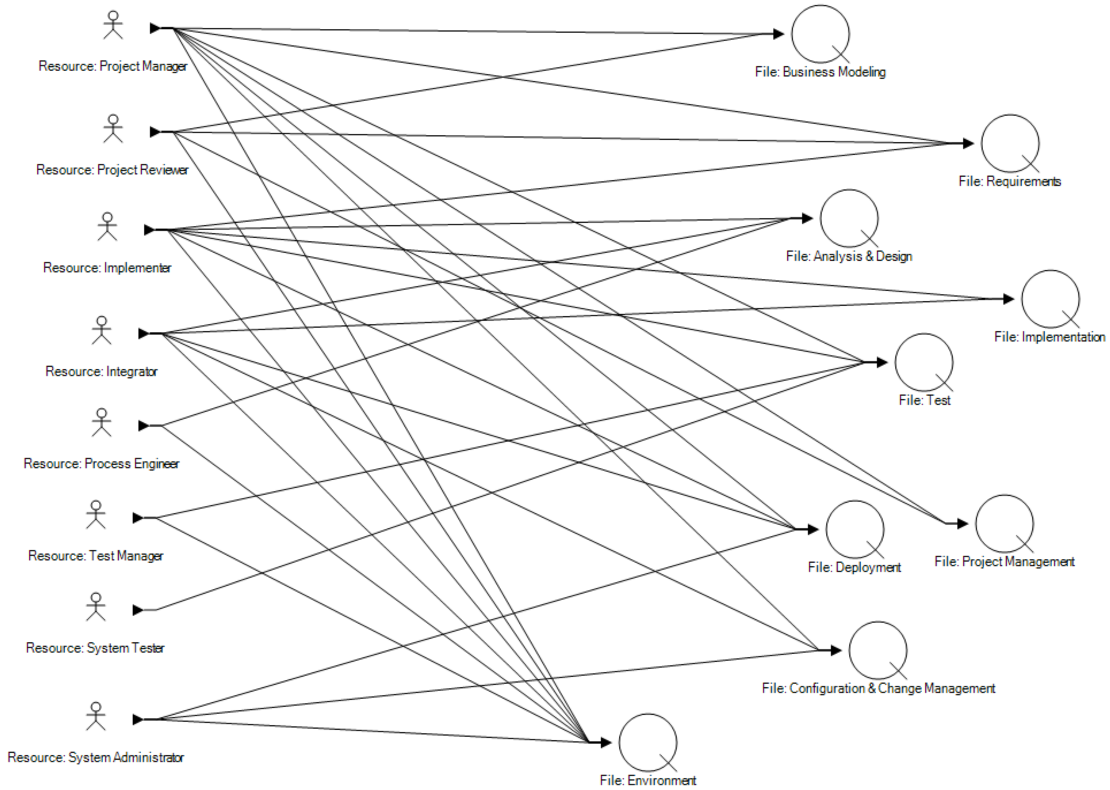

Figure 4.1. Resource Modeling of Small Scale Projects

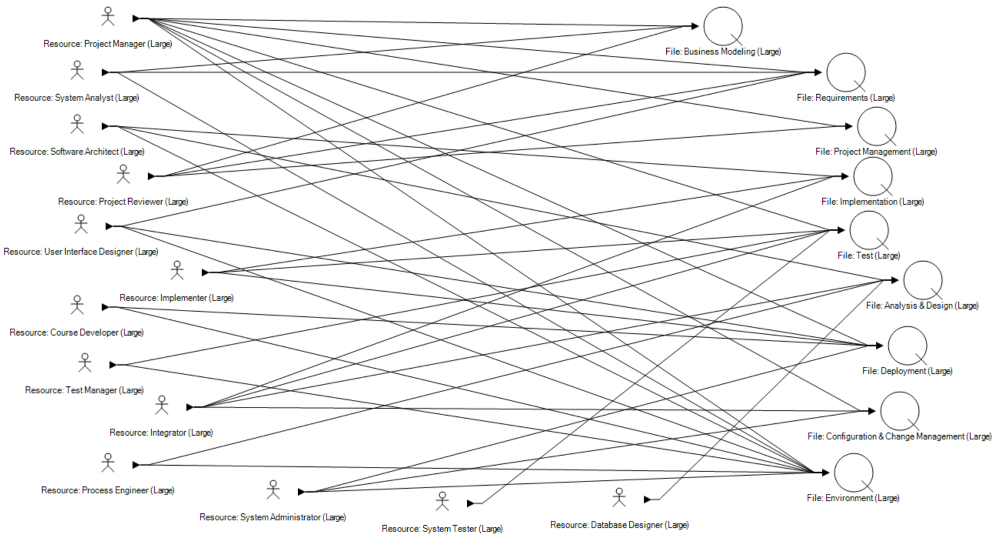
Resource: Project Manager (Large)
Resource: System Analyst (Large)
Resource: Software Architect (Large)
Resource: Project Reviewer (Large)
Resource: User Interface Designer (Large)
Resource: Implementer (Large)
Resource: Course Developer (Large)
Resource: Test Manager (Large)
Resource: Integrator (Large)
Resource: Process Engineer (Large)
Resource: System Administrator (Large)
Resource: System Tester (Large)
Resource: Database Designer (Large)
File: Business Modeling (Large)
File: Requirements (Large)
File: Project Management (Large)
File: Implementation (Large)
File: Test (Large)
File: Analysis & Design (Large)
File: Deployment (Large)
File: Configuration & Change Management (Large)
File: Environment (Large)

Figure 4.2. Resource Modeling of Small Scale Projects

Figure 4.1. is a simulation model for Resources on small scale projects and figure 4.2. for Resources on large scale projects. The element used in the model is the resource element which functions to define the resources involved in developing small-scale software. Then the file element in the model functions as a resource queue when it is waiting its turn to be executed when the model is run. The file element represents each workflow in RUP. Every resource is connected to a file that matches the task. On small-scale projects the resources used are eight and number thirteen for large-scale projects. While the file elements number nine according to the number of workflows in RUP.

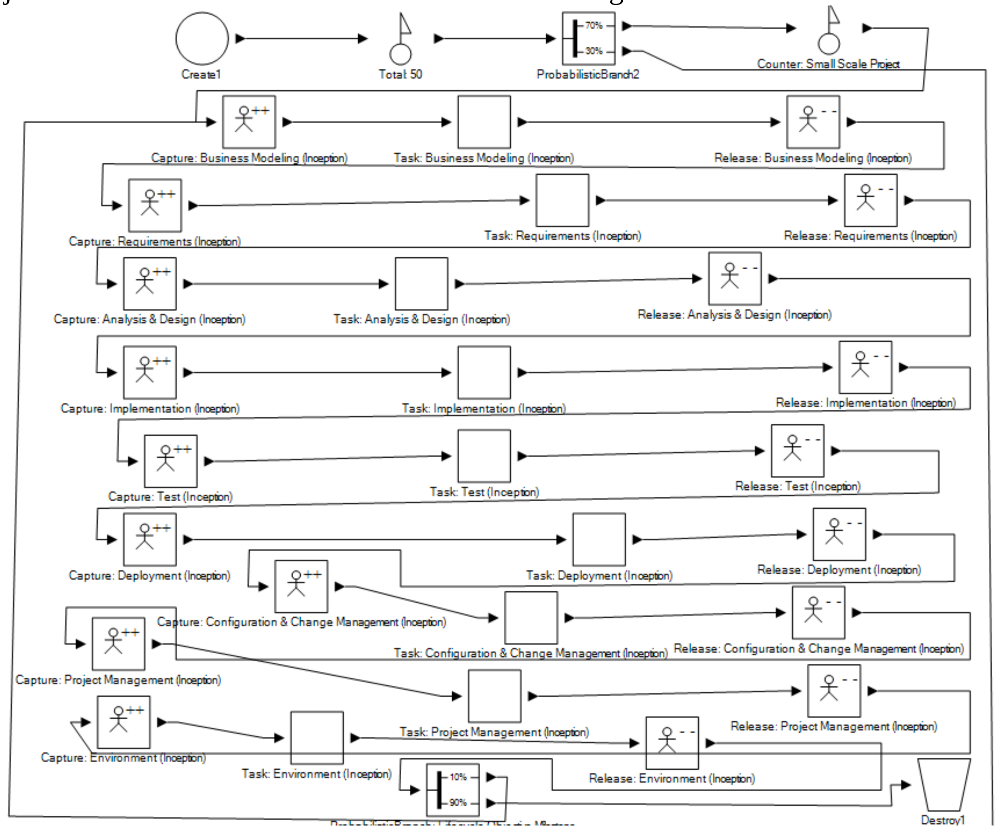
Create1
Total 50
70%
30%
ProbabilisticBranch2
Counter: Small Scale Project
Capture: Business Modeling (Inception)
Task: Business Modeling (Inception)
Release: Business Modeling (Inception)
Capture: Requirements (Inception)
Task: Requirements (Inception)
Release: Requirements (Inception)
Capture: Analysis & Design (Inception)
Task: Analysis & Design (Inception)
Release: Analysis & Design (Inception)
Capture: Implementation (Inception)
Task: Implementation (Inception)
Release: Implementation (Inception)
Capture: Test (Inception)
Task: Test (Inception)
Release: Test (Inception)
Capture: Deployment (Inception)
Task: Deployment (Inception)
Release: Deployment (Inception)
Capture: Configuration & Change Management (Inception)
Task: Configuration & Change Management (Inception)
Release: Configuration & Change Management (Inception)
Capture: Project Management (Inception)
Task: Project Management (Inception)
Release: Project Management (Inception)
Capture: Environment (Inception)
Task: Environment (Inception)
Release: Environment (Inception)
10%
90%
Destroy1

Figure 4.3. The Simulation Model of Rational Unified Process (RUP)

The simulation model built on each type of project and stage has the same model, because in software development using RUP. The application in all four stages is inception, elaboration, construction and transition involving all workflows and roles provided. There are several things that distinguish at each stage of workflow and role. The difference is the main focus at each stage that is being worked on, so the processing time will be different at each stage even at the same workflow.

Elements used in small-scale project simulation models include: create elements function to create entities (incoming projects). Counter elements function to record each number of projects that pass through to enter the next element. The probability branch element serves to enter the percentage number of each type of project. Capture element is defined on a model based on workflow, which serves to enter each role that works on each workflow. The Release element functions to release the server that the resource has finished using on the capture element. The task element that functions to enter the work time on each workflow by using one type of distribution. Each element has a configuration for entering data from the model design.

Some probability branch elements are available between each of the different resource assignments, the purpose is to simulate errors that might occur during development. The probability element has two branches, namely the branch at the top with Prob = 0.1 indicating that the 10% error rate on the project must be repeated in the previous stage to correct the error. Then the branch at the bottom with Prob = 0.9 indicates that 90% of the projects have no errors after the completion of each stage. In addition, probability branch elements are found at the beginning of the simulation model that uses two branches based on the type of project that has been determined. The first branch with Prob = 0.7 shows that 70% of the projects entered are small scale projects. The second branch with Prob = 0.3 shows that 30% of the projects entered are large-scale projects. Figure 4.3. shows the modeling element to simulate the model with the probability configuration mentioned above.

4.2 Simulation Results and Analysis

After the model is run it will generate statistical reports and several diagrams. The results to be analyzed are the resource usage diagram. A diagram of some of these resources will be displayed in the following figure.

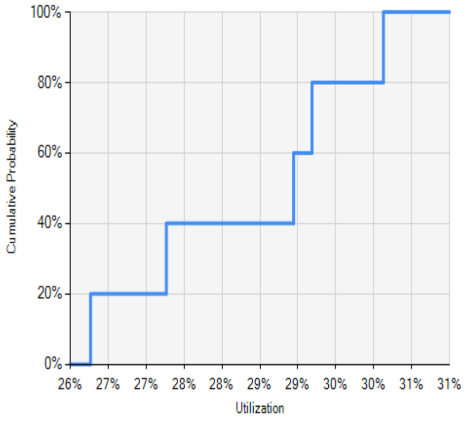

Figure 4.4. Project Manager Utilization for Small Scale Projects

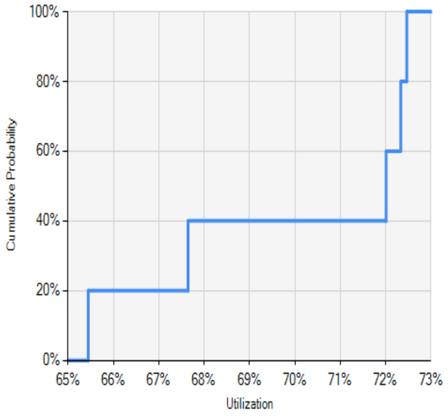

Figure 4.5. System Admin Utilization for Small Scale Projects

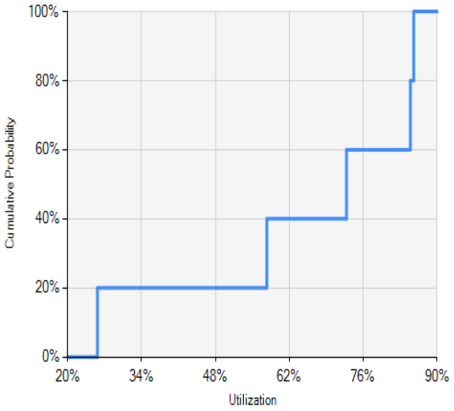

Figure 4.6. Project Manager Utilization for Large Scale Projects

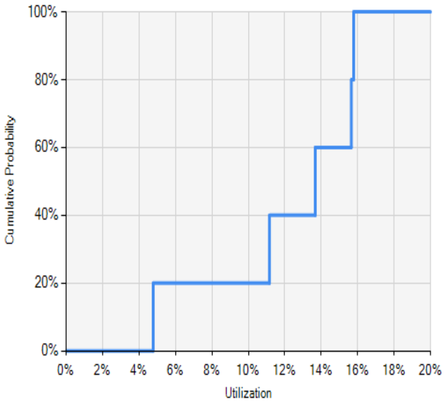

Figure 4.7. DB Designer Utilization for Large Scale Projects

Every resource has a result in the form of a diagram containing the usage percentage. The overall results of resource use in each type of project can be summarized in the following table 3.

Table 4.1. Use of Resources for Each Type of Project

Resource Name	Average Utilization for:	
	Ganymede Project	Mars Project
Project Manager	70,0%	65,3%
Integrator	56,3%	49,5%
Project Reviewer	29,3%	29,3%
System Analyst	-	26,4%
Process Engineer	20,6%	17,2%
Implementer	68,4%	38,6%
Software Architect	-	32,6%
System Administrator	28,4%	22,0%
Test Manager	18,7%	19,1%
User-Interface Designer	-	25,2%
Course Developer	-	14,1%
Database Designer	-	12,2%
System Tester	12,3%	14,1%

Based on Table 3, it can be concluded that in developing software using RUP, the most needed role is the project manager, in small projects and large projects. That is because the project manager's role concludes many other roles that work on large duration workflows.

Table 4.2. Statistical Results of Simulation Results

Jenis Proyek	*Received Project*	*Delivered Project*	*Average*
Proyek Skala Kecil (*Ganymede Project*)	32,000	32,000	49,100
Proyek Skala Besar (*Mars Project*)	18,000	18,000	23,484

Table 4.3 shows the number of projects received and sent for each type of project. The results in the table show that the number of projects received and sent has the same amount, meaning that in the execution of the project and the utilization of resources it is appropriate. The total of all projects sent is also in accordance with the number of projects entered at the beginning of the simulation model built, namely small-scale projects with 32 large-scale projects and 18 projects so that the total project is 50. These results are based on the percentage already included. Small scale projects have a greater percentage than large scale projects.

5 CONSLUSION

This research produced a simulation model for software development. Making a simulation model is done using Simphony.NET. The simulation model made consists of all phases in the Rational Unified Process (RUP) including roles, duration, and type of project. In this study there are two types of projects, namely, projects with small scale (Ganymede Project) and large scale (Mars Project). The roles used are reduced model resources for small-scale projects and based model resources for large-scale projects. The results of making this simulation can help and facilitate software developers who use RUP in estimating the role and number of team members according to the type of project.

REFERENCES

[1] A. Singh and P. J. Kaur, "A Simulation Model for Incremental Software Development Life Cycle Model," vol. 8, no. 7, pp. 126–132, 2017.

[2] P. Alexandra and F. Monteiro, "Tailoring CMMI-DEV and RUP Frameworks for ML2/3-Compliance Analysis," 2014.

[3] S. Thind, "A Simulation Model for the Spiral Software Development Life Cycle," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 03, no. 05, pp. 3823–3830, 2015.

[4] iDatalabs, "iDatalabs," 2018. .

[5] S. M. Abourizk, S. A. Hague, and R. Ekyalimpa, *Construction Simulation*. 2016.

[6] I. Sommerville, *Software Engineering*. 2010.

[7] Y. Bassil, "A Simulation Model for the Waterfall Software Development Life Cycle," *Int. J. Eng. Technol.*, vol. 2, no. 5, pp. 2049–3444, 2012.

[8] University of Alberta, "Simphony.NET 4.6," *University of Alberta*, 2018. [Online]. Available: https://www.ualberta.ca/engineering/research/groups/construction-simulation/simphony.

[9] S. W. Ambler, "A Manager's Introduction To The Rational Unified Process (RUP)," *Version: December*, vol. 4, p. 2005, 2005.

[10] P. Kroll and P. Kruchten, *The Rational Unified Process Made Easy*. 2003.