Initialization

Scope Fixation

TA Recruitment (TR) is a mission mode project with the objective to make the TA recruitment process done in a computerized way following paperless process trend. Our concept is to break down the hassles & waste of time every applicant faces during application.

Vision Statement

Our vision is to revolutionize the TA recruitment process through our innovative software, empowering companies to efficiently and effectively attract, screen, and hire the most qualified candidates. We aim to provide a user-friendly and customizable platform that leverages the to optimize the entire recruitment cycle, from sourcing to on-boarding. Our ultimate goal is to help with diversity and inclusion at the forefront of every decision.

What is the purpose of our TA Recruitment (TR) system?

Teaching Assistant (TA) is needed to help teacher to create an environment that is efficient to learning. TA also needed to help the corresponding students for better learning. It reduces the workload from teacher and both can handle and deliver more information in compared with before. TA Recruitment (TR) project has developed to ensure the TA recruitment process run in systemic approach and also support the paperless process trend. Conducting internal researches, student needs to fill a form with information that needed to examine as teaching assistant. While the examine process running it needed time to get the result. Our concern is to make it as fully functional in systemic paperless way.

Justification or Resource planning of the project

As it's an academic project we will try & build this project within 6 weeks

Resources:

- Human Resource: A team of 5 members will work together to build the project.
- Software Resource: [Visual Studio Code, HTML, CSS, JAVASCRIPT, PHP, MySQL]
- Hardware Resource: [Laptop, Mobile].

Facilities will receive

TA Recruitment (TA) is a slight-solution to the difficulties organized, systemic, consistency with concise time. It is adopted multiple engagement methodologies in favour of not only students but also varsity authority conducting less costly with certainly taking a paperless solution. Some of the students feel disturbed while applying for TA for the process and some of them don't

want to show others while applying. This process will be helpful for both of the parties as it breaks down the hassles & waste of time every applicant faces during application.

Deliverables

Home page: Home page will help to keep in touch with the site to get the relative information like research, portfolio, club and contact. It'll take admin, faculty, students to login through access bar.
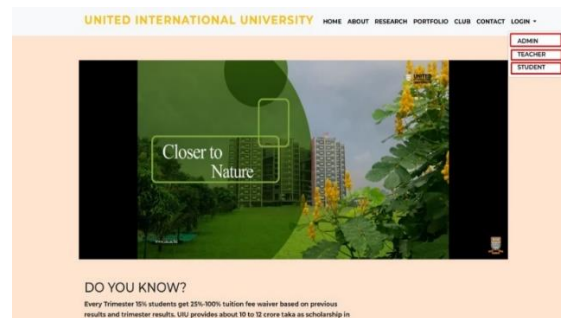
Login: With proper information like username & password admin, faculty, students can get access to the website.

Admin Homepage: Here, Admin can add students & assign any faculty to handle TA recruitment process providing necessary information like name, student id, CGPA, department & current trimester. System can send confirmation mail with user id & password as well as deleting account of the users.
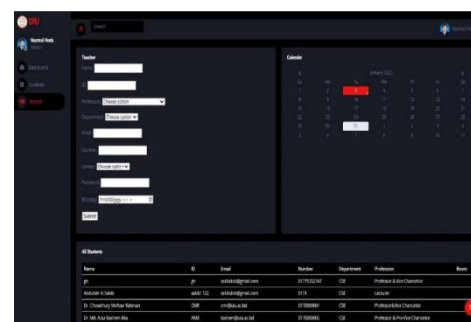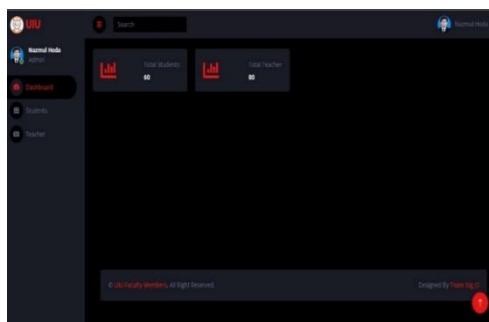
Faculty Home: Assigned faculty members can access the courses & overview the application. They can choose the suitable candidate & could assign TA.

Student Home: Here, student can view their profile & they can regularly update their profile & they will have the option to apply for any course according to their schedule & choice.

Confirmation: If the faculty member chooses the student as a TA then his name will be listed on the website & he will receive a confirmation mail.

Stake Holders

1. Admin
2. Faculty
3. Student

Planning

Pre Requisite

Clients wants to take delivery of the project to be taken within 4 weeks.

Resource Planning

- Human Resource
- Software Resource [Visual Studio Code, HTML, CSS, JAVASCRIPT, PHP, MySQL]
- Hardware Resource [Laptop, Mobile]

Work Distribution

Design	Santi Brata Nath Joy, Tanzid Ahsan
Development	Shabrina Airen Esha, Shahariar Sarkar

Testing	Shabrina Airen Esha, Umme Aimon Brishty
Deployment	Shahariar Sarkar, Tanzid Ahsan
Closing	Tanzid Ahsan

Cost Estimation

Though it's an academic project between students, we're very concerned about not spending even a penny extra. So in our project there is no cost estimation mentioned.

Time Estimation

Design	Week 01
Feature 01 [homepage, admin, student, faculty signup, login]	Week 02
Feature 02 [admin, faculty, student individual homepage]	Week 03
Feature 03 [application process]	Week 04
Feature 04 [mail confirmation]	Week 05
Final Presentation	Week 06

Change Management

Change management needs to set for better workflow, decrease project overheads, overcome obstacles and great work experience.

CM 01: If any member leaves

During the project development process, any member can leave the group due to any reason. At that time, any of the remaining members will hunt for the members needed for the suitable position. At any case, if the member can't be found the remaining task will be distributed among the remaining members.

CM 02: If the project changes

If the project is changed, we'll set a meeting as soon as possible and will try to figure a new project with the concern of faculty.

CM 03: Any Error occurs at last moment

Before the project show, if any problems occur, need to contact with classmates and faculty to solve the issue as soon as possible.

Risk Management

Category	Member Active	Priority
Code Issues	Shabrina, P_Sarkar	High
Database Error	Tanzid, Bristy	High
Data Lost	Santi, P_Sarkar	Moderate
Poor Productivity	Shabrina, P_Sarkar, Brishty, Tanzid	Moderate
Team Member Leaving	Santi, P_Sarkar	Low

Methodology

A methodology is a multi-action approach or set of tasks through which the main goal of the software can be achieved.

Why methodology is important?

For getting an efficient way of achieving the goal of a software project.

Types of Methodology

- Non-Agile
- Agile

Non-Agile Methodology

A methodology where the cycles or iterations are limited is called non-agile methodology.

> **Usual Stages:**
>
> - Planning
> - Requirements
> - System and software design
> - Implementation
> - Testing
> - Deployment
> - Maintenance/Updates

Waterfall Methodology

A non-agile model, where client can't change the requirements if the cycle starts where each phases are entirely completed before beginning the next phase and follow the phases linearly is known as waterfall methodology.

Advantage
- Simple & Easy way to understand
- Better for short projects
- No overlapping phases
- Clearly defined stages
- Easy to arrange tasks
- Nice documentation is possible

Disadvantage
- Don't allows interaction with user to ensure satisfaction.
- It takes too much time to set the documentation.

- Can't access the next phase until the present is completed.
- For making complex projects waterfall can't be used as there is no room for changes after initialing the plan.
- High amounts of risk and uncertainty.

When to use?
- Short project
- The requirements are clear and well understood
- When we need to complete the project in a linear way

Agile Methodology

Agile model is an adaptive methodology that break the project into small incremental and individual modules which can be addressed as sprint.

Types of Agile Methodology
- Scrum Methodology
- Boehm's Spiral Methodology
- eXtreme Programming (XP) Methodology
- Rational Unified Process (RUP) Methodology

Traditional vs Agile Planning
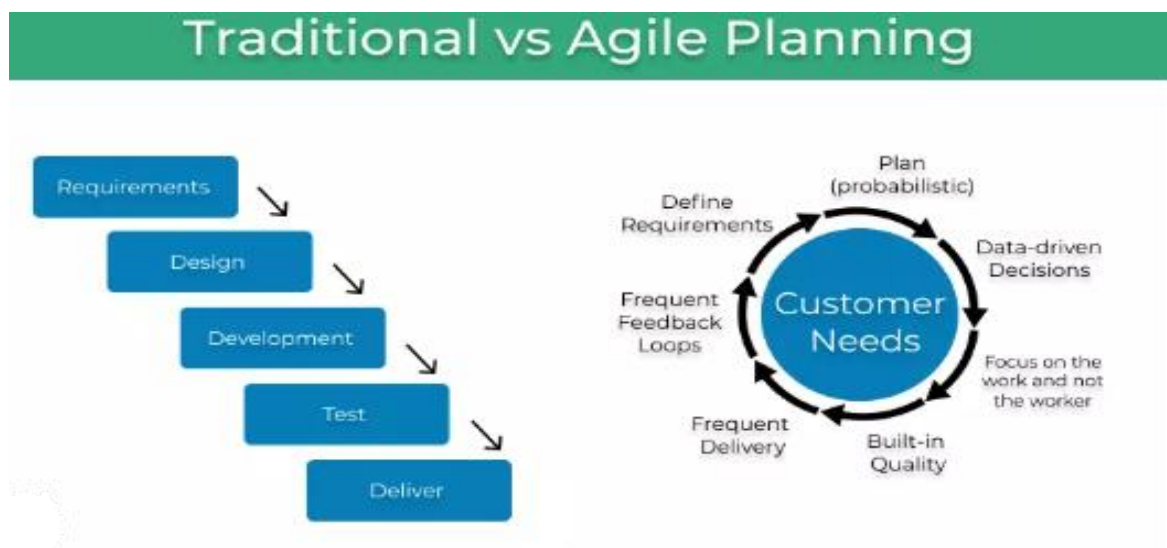
Rational Unified Process (RUP) Methodology

RUP is a methodology which focused on using UML to design and build projects. RUP has five phases which can help decrease development cost, wasted resources and total project management time. Phases:

- Inception
- Elaboration
- Construction
- Transition
- Production
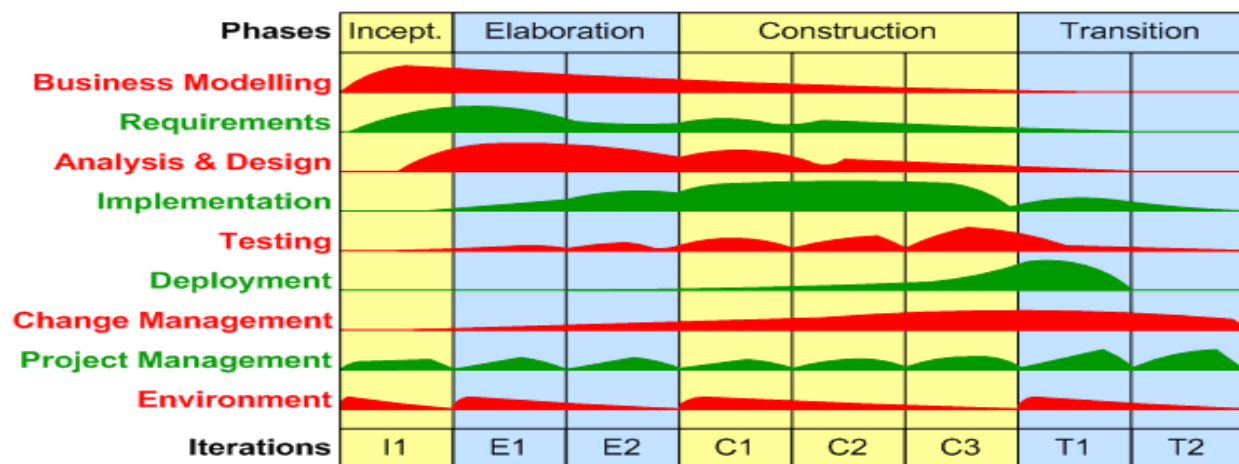
Advantage

- Provides proper documentation
- Improves process control and risk management
- Provides regular feedback to customers
- Reduce total project time
- Good online support is available in the form of tutorials and training

Disadvantage

- Complicated procedure.
- Expert professionals are required.
- Hard to merge again and again.
- Heavily dependent on the rational tools which are expensive.
- More dependency on risk management
- Team of expert professional is required, as the process is complex.

When to use?
- For complex and large projects.
- When team has expert professionals.
- When client gives ambiguous requirements that changes frequently.
- Object Oriented System that needs to update on a regular basis.

Scrum Methodology

Scrum is an incremental iterative agile methodology that breaks the project into small incremental modules, works at a time with continuous experimentation and feedback loop along the way to learn and improve project modules.

Typical terminologies of Scrum:
- Scrum Master
- Product Back-lock
- Velocity
- Sprint
- Product Owner
- Development Teams
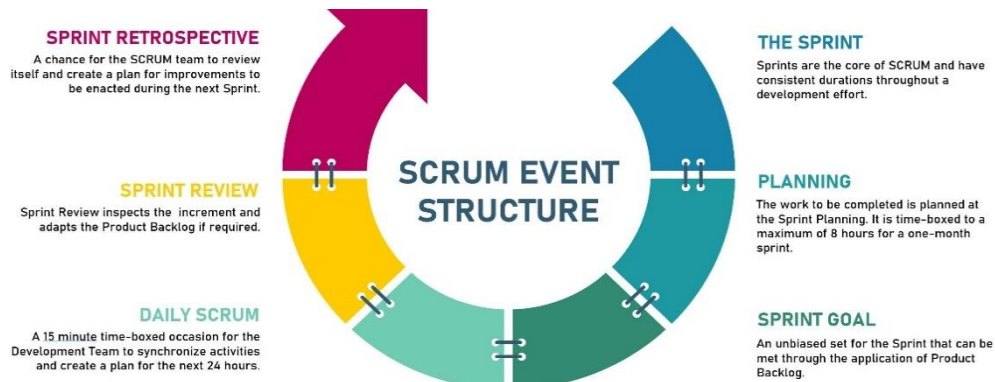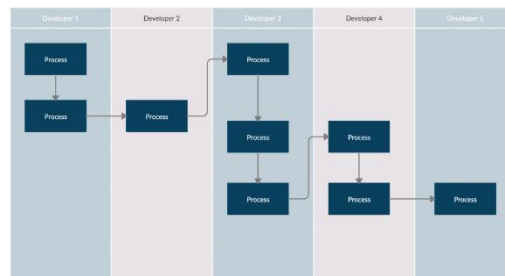
Advantage
- Suitable for fast moving projects.
- Cost effective.

- Updates regular feedback for achieving the goal.
- Contribution of individuals are noticed by the scrum master in scrum meetings.
- Scrum can help teams complete project deliverables quickly and efficiently
- Large projects are divided into easily manageable sprints
- Developments are coded and tested during the sprint review
- Scrum, being agile, adopts feedback from customers and stakeholders

Disadvantage
- Project failure is high if the members are not cooperative in nature.
- If scrum master can't strictly control the deadline of each scrum, then the completion of the project time may increase.
- Sometimes daily scrum meeting is frustrating.
- Skilled and committed members are required.
- Adopting the Scrum framework in large teams is challenging
- The framework can be successful only with experienced team members
- If any team member leaves in the middle of a project, it can have a huge negative impact on the project
- Quality is hard to implement until the team goes through an aggressive testing process

When to use?
- When the requirements are vague and needs frequently changes.
- Semi large projects.
- Regular basis update.

eXtreme Programming (XP) Methodology

Extreme programing method follows pair programming which illustrates doing both coding and testing in parallel and sometimes testing is completed before the code is developed.
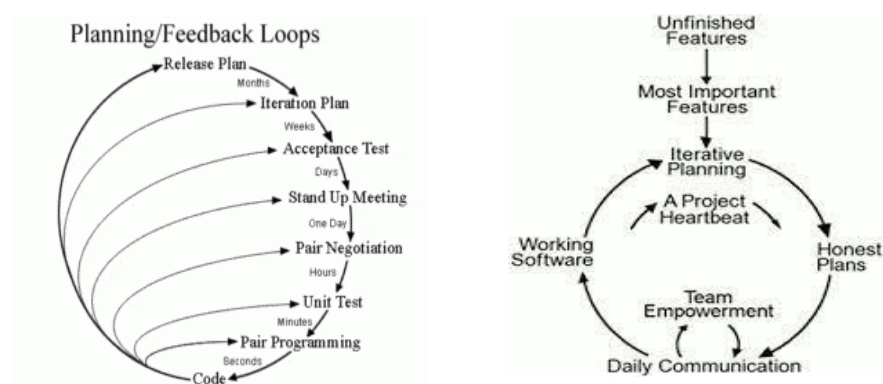
5 Essentials of Extreme Programming
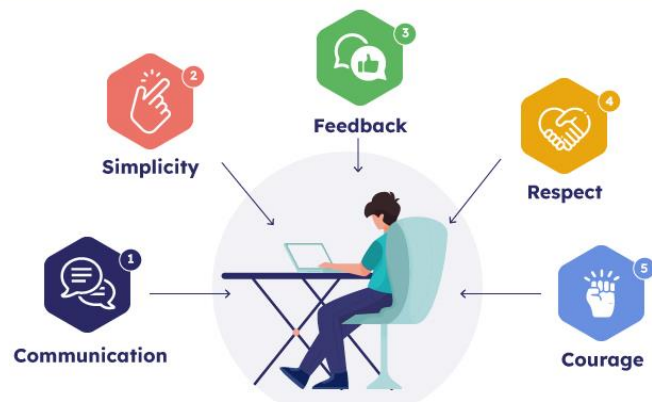
Advantage
- Development are constantly reviewed by the customer.
- Supports test driven development.
- Improve the design of existing code without changing its functionality.
- Quality Result
- Risk Handling
- Project Monitoring is Easy
- Suitable for large projects
- Flexibility in Requirement
- Early Estimation of Cost
- Changes can be accommodated later stages

Disadvantage
- When a project is large and needs to build in a short period of time then it's impossible using XP.
- Need to manage large team.
- Difficult
- Expensive
- Not suitable for low risk projects
- Too Much Dependability on Risk Analysis
- Difficulty in Time Management
- Complicated in terms of User-friendliness
- Strict rules and protocols are followed

When to use?
- When a project needs to build incrementally and for starting the project we don't need complete requirements in hand.
- Needs to develop the project in parallel.

Boehm's Spiral Methodology

Spiral model is risk driven model that guides a team to adopt elements for one or more process models. The number of cycles show the amount of risk.

Phases:

- Planning or determine objectives
- Risk Analysis or identity and resolve risk
- Engineering (development and tests)
- Evolution (Plan next iteration)
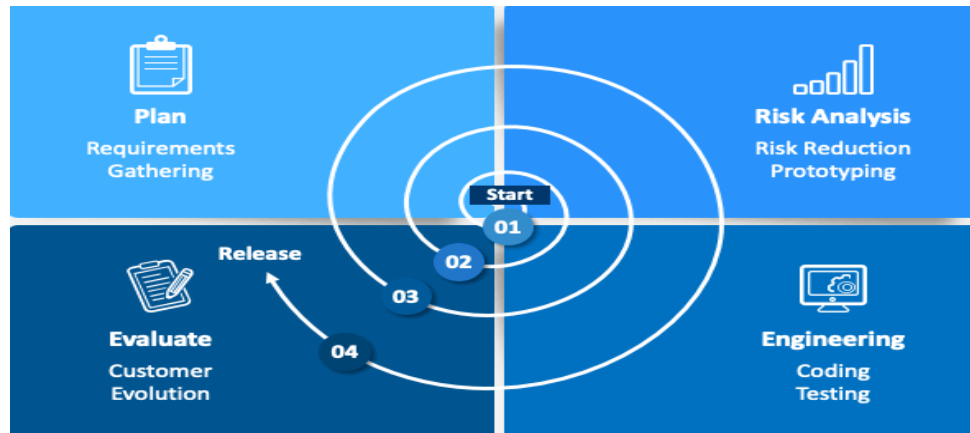
Advantage

- Good for large, mission-critical projects.
- Beneficial for extensive projects with high risk with unpredictability.
- Least maintainable.

Disadvantage

- Each phase contains extra overhead that create costs.
- Doesn't work well with smaller projects.
- Many prototype will be developed.

When to use?

- Suitable for high risk.
- When requirements are unclear.
- When long term project commitment is not feasible.
- When company has large team.

Incremental Soft. Dev. Methodology

Incremental software development methodology is a process where the requirements are divided into multiple independent modules of the software development cycle.
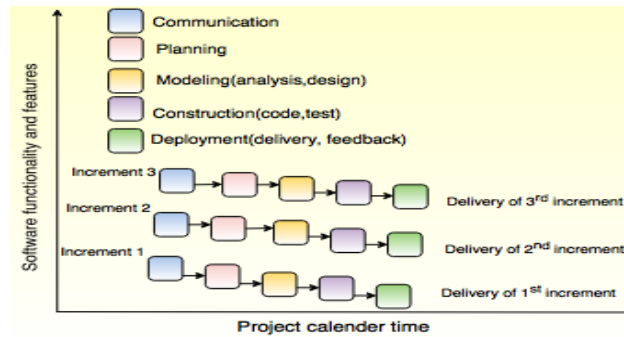
Advantage
- Prototype may carry the details information which can be used for the final version.
- Divides project into smaller versions which creates efficiency.
- Initial product delivery is faster.
- The final product is built as separate components, one at a time.

Disadvantage
- Customer requirements can't be fulfilled with prototype.
- It may become costly as prototype are often take for by the developer.
- User need to be active during every phases.

When to use?
- When a project has lengthy schedule.
- When software team are not very well skilled.
- When the requirements of the system well defined.

Which methodology is suitable for our project?

Initially the project has been developed with scrum model. But after a project update in respond with faculty, we'll change the methodology to waterfall methodology in the last week.

Why?

Our first approach as like we're show our project feature one after another in weekly basis to faculty. After faculty's review we will change our plan to complete the project. After the initial feature update, faculty shared review as we can't do the project further, because of the project has no practical view.

We have very limited time to complete our project. We made our requirements clear with the help of faculty and made our vision clear that we need to develop our project within last week for final presentation. The method is similar with waterfall method's criteria.

Work Breakdown Structure (WBS)

WBS stands for Work Breakdown Structure, which is a hierarchical decomposition of a project into smaller, more manageable tasks or work packages. It is a useful tool for project managers to help organize and plan the project by breaking it down into smaller components.

WBS Design Principals

The 100% Rule: The 100% rule is one of the most important principles guiding the development, decomposition and evaluation of the WBS. The rule applies at all levels within the hierarchy: the sum of the work at the "child" level must equal 100% of the work represented by the "parent" and the WBS should not include any work that falls outside the actual scope of the project, that is, it cannot include more than 100% of the work. If the WBS designer attempts to capture any action-oriented details in the WBS, he/she will likely include either too many actions or too few actions. Too many actions will exceed 100% of the parent's scope and too few will fall short of 100% of the parent's scope. The best way to adhere to the 100% Rule is to define WBS elements in terms of outcomes or results.

The best way to adhere to the 100% Rule is to define WBS elements in terms of outcomes or results

If the WBS designer attempts to capture any action-oriented details in the WBS, he/she will likely include either too many actions or too few actions. Too many actions will exceed 100% of the actual scope and too few will fall short of 100% of the parent's scope. The best way to adhere to the 100% Rule is to define WBS elements in terms of outcomes or results. This also ensures that the WBS is not overly prescriptive of methods, allowing for greater ingenuity and creative thinking on the part of the project participants. For new product development projects, the most common technique to assure an outcome-oriented WBS is to use a product breakdown structure (PBS).

Feature-driven software projects may use a similar technique which is to employ a feature breakdown structure. When a project provides professional services, a common technique is to capture all planned deliverables to create a deliverable-oriented WBS. Work breakdown structures that subdivide work by project phases (e.g. Preliminary Design Phase, Critical Design Phase) must ensure that phases are clearly separated by a deliverable (e.g. an approved Preliminary Design Review document, or an approved Critical Design Review document).

Level 2 is the Most Important

- It helps to determine actual costs
- It schedules data as group for future project cost & schedule estimating
- It helps project manager to know how much it took to design a product & he can use that data for future analogous estimating.

In other cases, the project manager may want to know how much a major part of the product actually cost after the project was completed. For this a PBS would be used. Level-2 is therefore used to capture "actuals" from a project for future estimating purposes.

The Four Elements in Each WBS Element

Each WBS element, when completed should contain the following four items:

1. The scope of work, including any "deliverables."
2. The beginning and end dates for the scope of work.
3. The budget for the scope of work.
4. The name of the person responsible for the scope of work.

By using a WBS in this manner the project manager can approach a complex project and decompose it into manageable, assignable portions. There is minimal confusion among project members when this technique is used.

Mutually-exclusive Elements

- In addition to 100% rule, there should not be any overlapping in scope definition between two elements of a WBS as this ambiguity could result in duplicated work or miscommunications about responsibility & authority.
- Overlap is likely to cost confusion regarding cost accounting.
- WBS names should be ambiguous as it will help to clarifying the distinctions between WBS elements.

How Far Down?

- WBS is decomposed to the work package level as it is the lowest level in the WBS. In this point the cost & schedule can be reliably estimated.
- If a WBS terminal elements are defined too broadly, it may not be possible to track project performance effectively.
- If a WBS terminal elements are too granular, it may be inefficient to keep track of so many terminal elements, especially if the planned work is in the distant future.
- A satisfactory tradeoff may be found in the concept of progressive elaboration which allows WBS details to be progressively refined before work begins on an element of work.

- We can use rolling wave planning for a project which establishes a regular time schedule for progressive elaboration.

In reality, an effective limit of WBS granularity may be reached when it is no longer possible to define planned outcomes, and the only details remaining are actions. Unless these actions can be defined to adhere to the 100% Rule, the WBS should not be further subdivided

The 40-Hour Rule of Decomposition

Another rule-of-thumb for determining how far down a WBS should be decomposed is called the "40 Hour Rule." Generally, when a project has been decomposed down to an element that has about 40 hours of allocated direct labor, there is no need to decompose further.

The 4% Rule of Decomposition

As per Gary Heerken's suggestion, 4% Rule for decomposing a WBS. With this rule a WBS is adequately decomposed when the lowest element is about 4% of the total project. 3 For a 26-week schedule, the lowest element should be about one week. For a $2.6M project, the lowest level should be about $104K.

Work Breakdown Structure (WBS)

WBS LEVEL 03

WBS LEVEL 01

1. TA RECRUITMENT (TA) (100%)

WBS LEVEL 02

1. TA RECRUITMENT (TA)

1.1 Project Management	16%
1.2 Analysis	19%
1.3 Design	15%
1.4 Developing	30%
1.5 Testing	12%
1.6 Closing	8%
	100%

1. TA RECRUITMENT (TA)

1.1 Project Management

1.1.1 Scope Analysis	4%
1.1.2 Scheduling & Estimation	8%
1.1.3 Risk Management	4%

1.2 Analysis

1.2.1 Information Gathering	5%
1.2.2 Requirements	5%
1.2.3 Benchmark Analysis &Feature List Fixation	9%

1.3 Design

1.3.1 Prototype	6%
1.3.2 Architecture Design	9%

1.4 Developing

1.4.1 Feature 01	7%
1.4.2 Feature 02	7%
1.4.3 Feature 03	10%
1.4.4 Feature 04	6%

1.5 Testing

1.5.1 Review	3%
1.5.2 Verification & Validation	2%
1.5.3 Quality Attributes	4%

1.6 Closing

1.6.1 Review Deliverables	4%
1.6.2 Closeout Meeting	2%
1.6.3 Closeout Documents	2%
	100%

TASK	START	END	PERSON	DEPENDENCY	ESTIMATED DURATION
1.1 Project Management					
1.1.1 Scope Analysis	Week 01	Week 01	Shabrina	Project Backlog	1
1.1.2 Scheduling & Estimation	Week 01	Week 02	Umme	Scope Analysis	2
1.1.3 Risk Management	Week 02	Week 02	Tanzid		1
1.2 Analysis					
1.2.1 Information Gathering	Week 03	Week 03	Shahariar		1
1.2.2 Requirements	Week 03	Week 04	Tanzid		2
1.2.3 Benchmark Analysis & Feature List Fixation	Week 03	Week 03	Shabrina	Info. Gathering	1
1.3 Design					
1.3.1 Prototype	Week 05	Week 05	Santi	Estimation	1
1.3.2 Architecture Design	Week 05	Week 05	Santi	Prototype	1
1.4 Developing					
1.4.1 Feature 01	Week 06	Week 07	Shabrina	Prototype	2
1.4.1 Feature 02	Week 06	Week 07	Umme	Prototype	2
1.4.1 Feature 03	Week 06	Week 08	Shabrina	Prototype	3
1.4.1 Feature 04	Week 07	Week 08	Tanzid	Feature 03	1
1.5 Testing					
1.5.1 Review	Week08	Week 09	Shahariar	Developing	1
1.5.2 Verification & Validation	Week 08	Week 08	Umme	Review	1
1.5.3 Quality Attributes	Week 08	Week 08	Shabrina	V. & V.	1
1.6 Closing					
1.6.1 Review Deliverables	Week 09	Week 09	Santi	Deliverables	1
1.6.2 Closeout Meeting	Week 09	Week 09	Santi		1

1 TA Recruitment (TA)

1.1 Project Management

1.1.1 Scope Analysis	1.1.2 Scheduling & Estimation	1.1.3 Risk Management
• **Deliverables: TA Recruit in an automated way** • **Beginning Time: Week 01 & End Time: Week 01** • **Budget: N/A** • **Person: Shabrina**	• **Deliverables: Completion of project on time** • **Beginning Time: Week 01 & End Time: Week 02** • **Budget: N/A** • **Person: Umme**	• **Deliverables: For risk handling** • **Beginning Time: Week 02 & End Time: Week 02** • **Budget: N/A** • **Person: Tanzid**

1.2 Analysis

1.2.1 Information Gathering	1.2.2 Requirements	1.2.3 Benchmark Analysis & FLF
• **Deliverables: For finding proper features.** • **Beginning Time: Week 03 & End Time: Week 03** • **Budget: N/A** • **Person: Shahariar**	• **Deliverables: Requirements for problem solving.** • **Beginning Time: Week 03 & End Time: Week 04** • **Budget: N/A** • **Person: Tanzid**	• **Deliverables: Comparing with existing products & fixing the relevant features.** • **Beginning Time: Week 03 & End Time: Week 03** • **Budget: N/A** • **Person: Shabrina**

1.3 Design

1.3.1 Prototype	1.3.2 Architecture Design
• **Deliverables: For finding proper features.** • **Beginning Time: Week 03 & End Time: Week 03** • **Budget: N/A** • **Person: Santi**	• **Deliverables: For finding proper features.** • **Beginning Time: Week 03 & End Time: Week 03** • **Budget: N/A** • **Person: Santi**

1.4   Developing

1.4.1   Feature 01

- **Deliverables: Homepage, admin, student, faculty signup, login**
- **Beginning Time: Week 06 & End Time: Week 07**
- **Budget: N/A**
- **Person: Shabrina**

1.4.2   Feature 02

- **Deliverables: Admin, faculty, student individual homepage**
- **Beginning Time: Week 06 & End Time: Week 07**
- **Budget: N/A**
- **Person: Umme**

1.4.3   Feature 03

- **Deliverables: Application Process**
- **Beginning Time: Week 06 & End Time: Week 08**
- **Budget: N/A**
- **Person: Shabrina**

1.4.4   Feature 04

- **Deliverables: Mail Confirmation**
- **Beginning Time: Week 03 & End Time: Week 07**
- **Budget: N/A**
- **Person: Tanzid**

1.5   Testing

1.5.1   Review

- **Deliverables: Checking System**
- **Beginning Time: Week 08 & End Time: Week 08**
- **Budget: N/A**
- **Person: Shahariar**

1.5.2   Verification & Validation

- **Deliverables: For integration.**
- **Beginning Time: Week 08 & End Time: Week 08**
- **Budget: N/A**
- **Person: Umme**

1.5.3   Quality Attributes

- **Deliverables: Availability, performance & security**
- **Beginning Time: Week 08 & End Time: Week 08**
- **Budget: N/A**
- **Person: Shabrina**

1.6 Closing

1.6.1 Review Deliverables	1.6.2 Closeout Meeting
• **Deliverables:** TA Recruitment along with its features. • **Beginning Time: Week 09 & End Time: Week 09** • **Budget: N/A** • **Person: Santi**	• **Deliverables:** Handover the project. • **Beginning Time: Week 09 & End Time: Week 09** • **Budget: N/A** • **Person: Santi**

Wideband Delphi Method

Wideband Delphi Technique is a consensus-based estimation technique for estimating effort. Useful when estimating time to do a task. Participation of experienced people and they individually estimating would lead to reliable results. People who would do the work are making estimates thus making valid estimates.

Name	Wideband Delphi Script
Purpose	To break down a complex project or deliverable into smaller, more manageable components, making it easier to plan, organize, and execute.
Summary	A hierarchical decomposition of a project or deliverable into smaller, more manageable components.
Work Products	Input: • The project scope statement, which outlines the goals, objectives, deliverables, and requirements of the project. • Project charter • Stakeholder's requirements • Deliverables Output: • A hierarchical structure that breaks down the project or deliverable into smaller, more manageable components.
Entry Criteria	1. A clear project scope statement, which outlines the goals, objectives, deliverables, and requirements of the project. Other entry criteria may include the project charter, stakeholder requirements. 2. To have sufficient time and resources to develop the WBS and to engage the appropriate stakeholders for input and feedback.
Basic Course of Events	1. **Choosing The team:** This involves clarifying the project goals, objectives, deliverables, and requirements. This will provide the basis for breaking down the project into smaller

	components.
	2. **Kickoff meeting:** This involves identifying the key deliverables that need to be produced to achieve the project objectives. These deliverables become the top-level elements in the WBS.
	3. **Individual preparation:** This involves breaking down the major deliverables into smaller, more manageable components. This is typically done by asking "what needs to be done?" or "what are the steps to produce this deliverable?" and continuing the process until the components are small enough to estimate time and cost.
	4. **Estimation session:** This involves assigning specific tasks or work packages to individual team members or groups. Work packages are the smallest elements of the WBS and can be easily estimated and managed.
	5. **Develop WBS diagram:** This involves creating a tree-like diagram that visually represents the WBS. The diagram shows the hierarchy of deliverables, sub-deliverables, and work packages, along with their relationships and dependencies.
	6. **Assembling tasks:** This involves reviewing the WBS with the project team and stakeholders to ensure that all requirements are included, and that the WBS accurately reflects the scope of the project.
	7. **Receiving results:** Used as a foundation for project planning, resource allocation, cost estimation, scheduling, risk management and other project management processes.
Alternative Paths	1. Scope creep: This happens when the project scope expands beyond what was originally defined, making it difficult to accurately break down tasks and allocate resources.
	2. Lack of clarity: If the project objectives, tasks, and deliverables are not clearly defined, it can be difficult to create a comprehensive and accurate WBS.
	3. Overcomplicating the structure: If the WBS is overly complicated or has too many levels of detail, it can become confusing and difficult to manage.
Exit Criteria	The script ends after the deliverables are identified with complete and comprehensive work breakdown structure (WBS).

Individual Preparation Result

Task List (SB.)	Time (Week)	Assumptions	
Scope Analysis	1	1	
Scheduling & Estimation	2	2	
Risk Management	1	3	
Information Gathering	1	4	
Requirements	2		
Benchmark Analysis & Feature List Fixation	1		
Prototype	1		
Architecture Design	3		
Feature 01	2		
Feature 02	2		
Feature 03	3		
Feature 04	1		
Review	2		
Verification & Validation	1		
Quality Attributes	1		
Review Deliverables	1		
Closeout Meeting	2		
Calender Waiting Time, delays			
Project Overhead Tasks			

Task List (SH.)	Time (Week)	Assumptions	
Scope Analysis	4	1 Confirmation Email	
Scheduling & Estimation	3	2	
Risk Management	1	3	
Information Gathering	1	4	
Requirements	2		
Benchmark Analysis & Feature List Fixation	5		
Prototype	1		
Architecture Design	3		
Feature 01	3		
Feature 02	3		
Feature 03	7		
Feature 04	5		
Review	1		
Verification & Validation	2		
Quality Attributes	1		
Review Deliverables	2		
Closeout Meeting	1		
Calender Waiting Time, delays			
Project Overhead Tasks			

Task List (U.)	Time (Week)
Scope Analysis	2
Scheduling & Estimation	2
Risk Management	2
Information Gathering	5
Requirements	4
Benchmark Analysis & Feature List Fixation	2
Prototype	5
Architecture Design	2
Feature 01	5
Feature 02	2
Feature 03	8
Feature 04	2
Review	2
Verification & Validation	1
Quality Attributes	1
Review Deliverables	1
Closeout Meeting	1
Calender Waiting Time, delays	
Project Overhead Tasks	

Assumptions
1 Login Page UIU Logo
2
3
4

Task List (SA.)	Time (Week)
Scope Analysis	1
Scheduling & Estimation	2
Risk Management	1
Information Gathering	1
Requirements	3
Benchmark Analysis & Feature List Fixation	1
Prototype	1
Architecture Design	1
Feature 01	4
Feature 02	3
Feature 03	3
Feature 04	4
Review	1
Verification & Validation	1
Quality Attributes	1
Review Deliverables	1
Closeout Meeting	1
Calender Waiting Time, delays	
Project Overhead Tasks	

Assumptions
1 Research (Nav Bar)
2
3
4

Task List (T.)	Time (Week)	Assumptions	
Scope Analysis	3	1	
Scheduling & Estimation	2	2	
Risk Management	1	3	
Information Gathering	1	4	
Requirements	2		
Benchmark Analysis & Feature List Fixation	1		
Prototype	3		
Architecture Design	1		
Feature 01	5		
Feature 02	4		
Feature 03	6		
Feature 04	1		
Review	1		
Verification & Validation	1		
Quality Attributes	1		
Review Deliverables	1		
Closeout Meeting	1		
Calender Waiting Time, delays			
Project Overhead Tasks			

Fiiled-in Estimation Form

Name: Shahariar Sarkar (SH.)			Date: 18/02/2023		Estimation:				
Goal Statement: TA Recruitment (TR)							Unit: Week		
Category: (√) goal task, (√) quality task, ( ) project overhead, ( ) waiting time									

WBS or Priority #	Task Name	Estimation	Delta 01	Delta 02	Delta 03	Delta 04	Delta 05	Total	Assumptions
1	1.1.1 Scope Analysis	1	2						
2	1.1.2 Scheduling & Estimation	2	3						
3	1.1.3 Risk Management	1							
4	1.2.1 Information Gathering	1							
5	1.2.2 Requirements	2							
6	1.2.3 Benchmark Analysis & Feature List Fixation	1							
7	1.3.1 Prototype	1	2						Login Page UIU Logo
8	1.3.2 Architecture Design	1							
9	1.4.1 Feature 01	2	3						Research Nav Bar
10	1.4.1 Feature 02	2	4						
11	1.4.1 Feature 03	3	-2						
12	1.4.1 Feature 04	1	2						Confirmation Email
13	1.5.1 Review	1							
14	1.5.2 Verification & Validation	1	2						
15	1.5.3 Quality Attributes	1							
16	1.6.1 Review Deliverables	1							
17	1.6.2 Closeout Meeting	1							
	Delta		16						
	Total	23	49						

Estimation Session

- The moderator collects all estimates. Draw the estimated total on a line on the whiteboard and tabulate it.
- The estimator reads out clarifications and revisions to the list of tasks written on the estimator. Propose new or changed tasks, discovered hypotheses, or questions. No specific estimated time is discussed.
- The team resolved problems or disagreements. Since specific estimated times are not discussed, these disagreements are usually about the task itself and are often resolved by adding assumptions.
- Estimators modify their personal estimates by filling in the "Delta "column on the form.
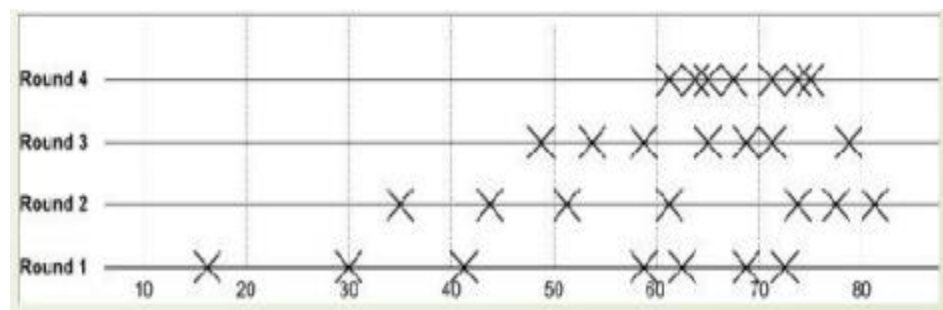
Summarized Results of Estimation

Goal Statement: TA Recruitment										
Eslimators: Shabrina(SB.), Shahariar(SH.), Santi(S.), Umme(U.), Tanzid(T.)									Unit: Week	
WBS or Priority #	Task Name	SB.	SA.	SH.	T.	U.	Best case	Worst Case	Avg.	Notes
1	1.1.1 Scope Analysis	1	2	4	3	2	1	4	2.33	
2	1.1.2 Scheduling & Estimation	2	3	3	4	3	2	4	3	
3	1.1.3 Risk Management	1	2	2	2	3	1	3	2	
4	1.2.1 Information Gathering	1	2	3	2	5	1	5	2.33	Discrepancy bet. SB. & U.
5	1.2.2 Requirements	2	3	2	2	4	3	4	2	
6	1.2.3 Benchmark Analysis & Feature List Fixation	1	2	5	2	2	1	5	2	Discrepancy bet. SB. & SH.
7	1.3.1 Prototype	1	2	2	3	5	1	5	1.67	Discrepancy bet. SB. & U.
8	1.3.2 Architecture Design	3	1	3	2	2	1	3	0.67	
9	1.4.1 Feature 01	2	4	3	5	5	2	5	2.33	
10	1.4.1 Feature 02	2	3	3	4	2	2	4	1.33	
11	1.4.1 Feature 03	3	3	7	6	8	3	8	4.33	Discrepancy bet. SA. & U.
12	1.4.1 Feature 04	1	4	5	1	2	1	5	2	
13	1.5.1 Review	2	1	1	1	2	1	2	0	
14	1.5.2 Verification & Validation	1	1	2	1	1	1	2	0	
15	1.5.3 Quality Attributes	1	1	1	1	1	1	1	0	
16	1.6.1 Review Deliverables	1	1	2	1	1	1	2	0	
17	1.6.2 Closeout Meeting	2	1	1	1	1	1	2	0	
	Total	27	36	49	41	49	24	64	25.99	

Risk Plan

According to "Applied Software Project Management" by Andrew Stellman and Jennifer Greene, a risk plan is a document that outlines the strategies and actions that will be taken to manage potential risks that could impact a software development project. The goal of a risk plan is to proactively manage risks and minimize their impact on the project's success.

Risk Planning Script

Name	Risk planning Script
Purpose	To access risks and create risk plan.
Summary	The risk planning meeting happens in three parts: a brainstorming session to identify risks; a discussion in which the probability and impact of each risk is estimated; and a discussion to identify actions that can mitigate risks. The end result is a risk management plan, which should be included verbatim in the final project plan.
Work Products	**Input** Any project documentation that has been developed so far. **Output** Risk plan: Assumptions generated by the Delphi process. Assumptions in the vision and scope document.
Entry Criteria	The project manager has gathered the project team for a two-hour meeting to assess the project's risks.
Basic Course of Events	**1. Brainstorm potential risks:** The project manager leads a brainstorming session to identify risks. Team members suggest every risk they can think of; the project manager writes the risks on a whiteboard as they come up. Brainstorming should be reminiscent of microwave popcorn: a few ideas should "pop" at first, followed by a large number being fired rapidly, slowing down to a final few "pops." The team will generally be able to judge when the risk identification is over. **2. Estimate the impact of each risk:** The team assigns a number from 1 (highly unlikely) to 5 (very likely to occur) to represent the estimated probability of each risk. Similarly, impact should be estimated by assigning a number from 1 (for a risk with low impact) to 5 (for a risk which, if it occurs, will require an enormous effort to clean up). **3. Build the risk plan:** The team identifies actions to be taken to mitigate high-priority risks and creates a risk plan that documents these actions.
Exit Criteria	The risk plan is finished.

Risk plan for TA Recruitment (TR)
Assessment team members: Shabrina(SB.), Shahariar(SH.), Santi(S.), Umme(U.), Tanzid(T.)

Risk ID & Description	Impact Description	Date Identified	Probability	Impact	Priority	Actions	Owner	Date Closed
(RID: 01) Changing requirements: Changes in client or stakeholder requirements mid-project.	lead to delays and additional work, resulting in increased costs and reduced quality.	4/03/2023	3	5	15	1. Communication 2. Prioritization: Prioritize changes based on their impact on the project timeline, budget, and overall goals. 3. Documentation	Shabrina	
(RID: 02) Scope creep: This happens when the requirements for the project expand beyond what was originally agreed upon	leading to delays and increased costs.	4/03/2023	2	3	6	1. Revisit the project's original scope and objectives. 2.Identify the changes that have caused the scope creep and evaluate their impact on the project's timeline, budget, and resources.	Shahariar	
(RID: 03) Inadequate testing: Incomplete or ineffective testing can result in bugs and glitches that may go unnoticed until the software is released.	causing damage to reputation and user trust.	4/03/2023	2	2	4	1. Encourage collaboration between developers, testers, and stakeholders to ensure testing aligns with business requirements and goals. 2. Continuously monitor and measure the effectiveness of testing, using metrics such as defect density, test coverage, and test effectiveness.	Tanzid	
(RID: 04) Security vulnerabilities: Failure to adequately secure the software.	lead to data breaches and other security threats, resulting in costly and damaging consequences.	4/03/2023	2	2	4	1. Create a plan to address the vulnerability, including the steps needed to fix it and any resources required. 2. Prioritize the vulnerability based on its severity and potential impact, and allocate resources accordingly.	Umme	
(RID: 05) Poor communication: Communication breakdowns between team members, stakeholders, or clients can.	lead to misunderstandings, delays, and errors in the development process.	4/03/2023	1	4	4	1. Set clear communication goals and expectations. 2. Choose the appropriate communication channel. 3. Listen actively and seek understanding.	Shabrina	
(RID: 06) Team member leaves: Team members leaving a group is a risk associated with projects or programs that rely on a cohesive team to complete tasks and achieve objectives.	hamper to achieve the goal with objectives	4/03/2023	1	4	4	1. Evaluate the impact on the team's workload and responsibilities 2. Reassign tasks and responsibilities to remaining team members 3. Determine if a replacement should be hired or if the team can continue without one	Shahariar	

RMMM (Risk Mitigation, Monitoring & Management) Plan

A Risk Mitigation, Monitoring, and Management (RMMM) plan is a structured approach to identifying, assessing, and managing risks in a project, program, or organization. It outlines the strategies and procedures for minimizing the impact of potential risks and monitoring and managing them throughout the project or program lifecycle.

Here every risks documented as RIS (Risk Information Sheet). In the individual sheets every risks describe to provide a clear view to the stakeholders.

Identification:

A process for identifying potential risks, their causes, and their potential impact on the project or program.

Assessment:

A process for analysing and evaluating the likelihood and impact of each identified risk, and prioritizing them based on their severity.

Risk Mitigation:

- Identify potential risks and their likelihood of occurring
- Determine the potential impact or consequence of each risk
- Develop a plan to minimize or avoid risks
- Implement risk mitigation strategies
- Monitor and review the effectiveness of risk mitigation strategies

- Adjust strategies as needed based on new information or changing circumstances

Risk Monitoring:

- Continuously monitor for new risks
- Track progress of risk mitigation strategies
- Collect data and feedback from stakeholders
- Analyse data to identify trends or potential issues
- Communicate findings to relevant parties
- Take corrective action as necessary

Risk Management:

- Establish a risk management plan
- Allocate resources and responsibilities for risk management
- Identify and assess risks
- Develop strategies to manage risks
- Implement risk management strategies
- Monitor and review the effectiveness of risk management strategies
- Adjust strategies as needed based on new information or changing circumstances
- Continuously improve the risk management process

Drawbacks of RMMM

- Overemphasis on risk avoidance can lead to missed opportunities or overly conservative decision-making.
- Risk management can be time-consuming and resource-intensive, which can impact project timelines and budgets.
- Risk mitigation strategies may not always be effective or may introduce new risks.

Risk Information Sheet (RIS)

A risk information sheet is a document that provides a summary of key information about a specific risk. It typically includes information such as the nature of the risk, its potential impact, the likelihood of occurrence, and any mitigation strategies that have been put in place.

The purpose of a risk information sheet is to provide a concise and easily understandable summary of a risk to stakeholders, so that they can make informed decisions about how to manage that risk. It is often used as part of a broader risk management framework, and may be reviewed and updated regularly to ensure that it remains current and accurate. In RIS, every element of RMMM will present in order with every risks

Risk Information Sheet

Risk ID:FID 01	Date: 04/03/2023	Probability: 3	Impact: 5

Risk Description:
Changing requirements is a risk associated with projects or programs that have evolving stakeholder needs, market conditions, or regulatory requirements. Changing requirements can result in project delays, increased costs, decreased quality, and stakeholder dissatisfaction.

Risk Impact:
Changing requirements can have significant impact on a project or program, including increased costs, missed deadlines, poor quality outcomes, and damage to reputation. It can also result in scope creep, where additional work is added to the project beyond the original scope.

Likelihood of Occurrence:
The likelihood of changing requirements occurring is high if stakeholder needs are not fully understood or if there is a lack of stakeholder involvement in the requirements gathering process. Market conditions, regulatory requirements, or other external factors can also impact the likelihood of changing requirements.

Mitigation Strategies:
Establish a robust requirements gathering process that involves all relevant stakeholders and includes regular reviews and feedback.
Use change control processes to manage changes to requirements, including impact analysis, cost and schedule assessments, and stakeholder approval.
Establish a clear project scope that is reviewed and approved by all stakeholders.
Develop contingency plans for addressing changes to requirements, including allocating additional resources, adjusting project timelines, or reducing project scope.
Monitor changes to requirements and communicate them effectively to all stakeholders.

Risk Owner:
Shabrina is responsible for managing and mitigating the risk of changing requirements. They should work closely with stakeholders to understand their needs, establish clear project scope, and develop robust change control processes.

Current Status:

Originator: Shabrina	Assigned: Tanzid

Risk Information Sheet

Risk ID: RID 05	Date: 04/03/2023	Probability: 1	Impact: 4

Risk Description:
Poor communication is a risk associated with project or program failure due to a lack of effective communication among stakeholders. Poor communication can result in misunderstandings, misaligned expectations, delays in decision-making, and decreased productivity.

Risk Impact:
Poor communication can have significant impact on a project or program, including increased costs, missed deadlines, poor quality outcomes, and damage to reputation.

Likelihood of Occurrence:
The likelihood of poor communication occurring is high if communication processes and procedures are not in place, stakeholders have conflicting priorities or objectives, or if communication channels are inadequate or unreliable.

Mitigation Strategies:
To mitigate the risk of poor communication, the following strategies should be considered:
Establish clear communication processes and protocols that outline roles and responsibilities, communication channels, and frequency of communication.
Encourage open and transparent communication among stakeholders to ensure all parties are informed and have a shared understanding of project goals and objectives.
Provide regular communication training and resources to stakeholders to ensure effective communication skills and techniques.
Use technology tools and platforms to facilitate communication and collaboration among stakeholders.
Monitor communication effectiveness regularly to identify and address any issues or gaps in communication.

Risk Owner:
Shahariar is responsible for managing and mitigating the risk of poor communication. They should monitor communication effectiveness and make adjustments to communication strategies and processes as needed.

Current Status:

Originator: Shabrina	Assigned: Tanzid

Risk Information Sheet			
Risk ID:RID 06	**Date: 04/03/2023**	**Probability: 1**	**Impact: 4**

Risk Description:
Team members leaving a group is a risk associated with projects or programs that rely on a cohesive team to complete tasks and achieve objectives. Team members leaving can result in decreased productivity, knowledge loss, decreased morale, and increased costs associated with hiring and training new team members.

Risk Impact:
Team members leaving a group can have a significant impact on project or program success, including decreased productivity, missed deadlines, decreased quality, and increased costs.

Likelihood of Occurrence:
The likelihood of team members leaving a group can vary depending on factors such as job market conditions, individual job satisfaction, and project team dynamics. However, it is always a possibility and should be considered in risk management planning.

Mitigation Strategies:
Develop a succession plan that includes cross-training and knowledge transfer among team members.
Provide clear career development opportunities and incentives to encourage team members to stay with the project or program.
Encourage open and transparent communication among team members to identify and address any issues that may lead to team member dissatisfaction or turnover.
Monitor team dynamics and morale regularly and address any concerns as they arise.
Develop a contingency plan to address any unexpected departures, including identifying potential replacement team members or adjusting project timelines.

Risk Owner:
Shahariar is responsible for managing and mitigating the risk of team members leaving a group. They should work closely with team members to identify potential issues and address them proactively.

Current Status:

Originator: Shahariar	**Assigned: Tanzid**