

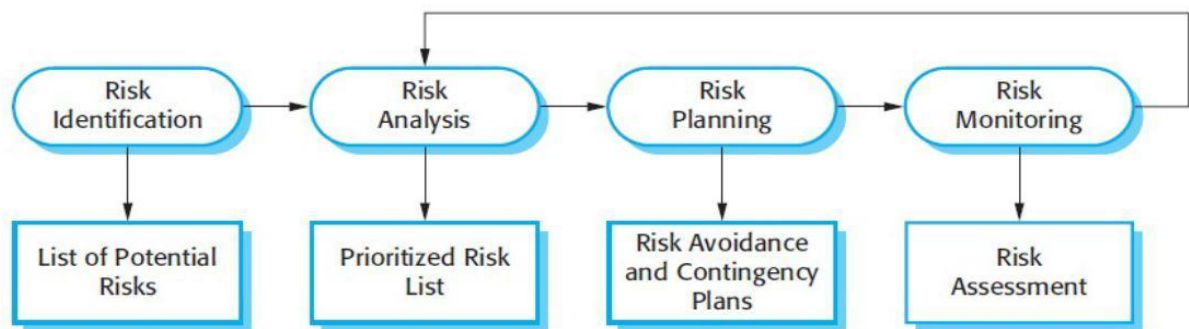
LinkedHU_CENG	Version: 1.0
Risk Management Report	Date: 13/05/2022

LinkedHU_CENG Risk Management Report

1 Introduction

A risk is a probability that some adverse circumstance will occur. There are some types of risks. **Project risks** affect schedule or resources. **Product risks** affect the quality or performance of the software being developed. **Business risks** affect the organization developing or procuring the software.

On the other hand risk management is concerned with identifying risks and drawing up plans to minimize their effect on a project. The risk management process includes risk identification, risk analysis, risk planning, and risk monitoring.



This document covers the risks, their significance, and potential consequences, as well as mitigation options. This analysis of risks and preparing strategies to avoid or mitigate negative consequences is critical to the project's success. The Risk Management Report is composed of 3 sections and one supplementary material (Risk List document). All the risk lists are to be updated throughout the lifecycle of the project.

2 Description

This section contains overviews of the potential risks involved. The Risk List document contains more extended and elaborated descriptions of these potential concerns. Below artifact is a list of the project's known and unknown hazards, organized by importance and associated with specific mitigation or contingency activities. This list highlights the dangers associated with a project in decreasing order of priority. It acts as a focal point for project activity and serves as the foundation for organizing iterations.

Potential Risk	Mitigation Strategy
Some members of the team have left the group before the completion of the project.	Reorganize the responsibilities and the team to lessen the impact of the team member's absence. Inform the stakeholders.

LinkedHU_CENG	Version: 1.0
Risk Management Report	Date: 13/05/2022

Some members of the team are ill and not available.	Reorganize the team such that there is more work overlap and everyone understands each other's jobs.
The time needed to complete the software is underestimated.	Investigate buying-in components; investigate use of a program generator. Alert customers to potential difficulties and the possibility of delays. Implement critical use cases, ask for an extension if it is needed.
The underlying technology is replaced by another due to the chosen one's inability to match the project's needs or failure to implement critical use cases.	Choose a technology (framework) that will be up to date for a long period and is more appropriate to the project's requirements and content. As quickly as feasible, try to upgrade the old framework to the new framework. Use existing software tools to convert the project if necessary.
Some of the software requirements are misunderstood.	Create strong communication channels between stakeholders and customers. Ensure that both parties understand the same goal. Request comments on our comprehension on a regular basis.
After the project's requirements have been implemented, some of them are altered.	Derive traceability information to assess requirements change impact; maximize information hiding in the design.
The technical skills required for a development task is underestimated.	Increase team member participation and information exchange. Experiment with approaches such as extreme programming and pair programming.
Insufficient time available for team training.	Team members develop knowledge in various areas and explain it to one another.
Database used in the system cannot process the data as fast as expected.	Investigate the possibility of buying a higher-performance database.
A severe software problem is found soon before the software is delivered.	Manage effective unit and integration tests to limit the likelihood of hidden flaws in the software. If it is too late, notify the stakeholders of the issue and possible delays.
Reusable software components contain inconsistencies, and can't be reused as planned.	Start to build your own component with the help of the reusable component while solving the inconsistencies.

LinkedHU_CENG	Version: 1.0
Risk Management Report	Date:13/05/2022

A critical use case implementation for the demo failed.	Inform the stakeholders about the situation, if it is possible ask for an extension, if not, deliver the demo for the parts to arrive in time.
A task assigned to a member of the team is wrongly understood.	Ensure strong and effective communication between the team members.
Time required for an assigned development task is underestimated.	Reduce the time required to finish the work by employing approaches like extreme programming and pair programming. If required, reassign task items to other members of the team.
A task under development is preventing another task from being completed.	Inform the team member responsible for the task and use programming techniques such as pair programming to reduce the impact of the situation.

3 Risk Management Report Specifications

Below lists only captures the critical and serious risks as “Risks Forecasted in Planning” and “Risks not Forecasted in Planning, but Observed”. We put the overall risk list in the project plan. The iteration plan contains only the tasks we are doing during the iteration to mitigate the risks. The project plan has to be revisited constantly as we update risks.

The following is a list of the risks we forecasted in planning and encountered while developing the project, as well as the strategies we used to address those risks. This list is also to be updated over the project's existence.

Risks Forecasted in Planning

Risks We Faced	Contingent Action	Status
The time required for a delivery (DEL3) is underestimated.	We notified the stakeholders about the problem and requested an extension for the corresponding delivery. The requested extension was granted.	Mitigated
Reusable software components contain inconsistencies that means they cannot be reused as planned.	We start to build our own component with the help of the reusable component while solving the inconsistencies.	Mitigated

LinkedHU_CENG	Version: 1.0
Risk Management Report	Date:13/05/2022

Insufficient time available for team training.	We have tried to develop knowledge in various areas and explain it to one another.	Mitigated
Some members of the team were ill and not available.	We have reorganized the team such that there is more work overlap and everyone understands each other's jobs.	Mitigated
The time needed to complete the software is underestimated.	We have focused on implementing most of the critical use cases.	Mitigated
A critical use case implementation for the final is demo failed.	We have delivered the final demo for the parts that arrive in time and informed the stakeholders about the situation. We have said the missing critical use case will be in version two.	Mitigated
Time required for an assigned development task is underestimated.	We have reduced the time required to finish the work by employing approaches like extreme programming and pair programming. And reassigned task items to other members of the team.	Mitigated
A task under development is preventing another task from being completed.	We have informed the team member responsible for the task and used programming techniques such as pair programming to reduce the impact of the situation.	Mitigated

LinkedHU_CENG	Version: 1.0
Risk Management Report	Date:13/05/2022

On the other hand, the following list contains the risks we did not forecasted in planning but encountered while developing the project, as well as how we handled those risks. This list is also to be updated and filled over the project's existence.

Risks Not Forecasted in Planning, but Observed

Risks We Faced	Contingent Action	Status
While 5 people are working on one project, sometimes we have worked on the same things and this situation resulted in conflicts when we tried to merge codes.	We have used GitHub's utilities for these kinds of problems and to not lose any working code, we have resolved the conflicts using the GitHub interface.	Mitigated
Computer crash of one of the team members.	We have reassigned the tasks of that member during her computer's process of being repaired. After three days her computer was repaired and she returned to her tasks.	Mitigated

LinkedHU_CENG	Version: 1.0
Risk Management Report	Date:13/05/2022

Concepts and Notations in The Risk List

- **Headline:** The short description of the potential risk.
- **Description:** Detailed description of the potential risk and its possible impact on the successful completion of the project.
- **Type:** The types of the risks. We grouped risks into following types according to their relation.
 1. People (P)
 2. Requirements (R)
 3. Estimation (E)
 4. Technology (Te)
 5. Tools (To)
 6. Organizational (O)
- **Impact:** The relative estimation of the potential impact this risk can have if left unhandled:
 1. Catastrophic (5)
 2. Serious (4)
 3. Moderate (3)
 4. Tolerable (2)
 5. Insignificant (1)
- **Probability:** The relative estimation of the possibility of the risk event happening:
 1. Unlikely (%25)
 2. Moderately (%50)
 3. Highly (%75)

Distribution of Tasks

This document was prepared primarily by the Software Project Manager (Sümeyye Meryem Taşyürek), with contributions of the entire team.