Software Engineering

Exercise 01

Name (of Group Members)	Matriculation Number
1. Radhe Mohan Jha	1750721
2. Tanish Jaggi	1750121
3. Artsiom Drankevich	1750712

Task 1
Learning goal: Become familiar with (generic) software project outcomes.
Difficulty: (10, -, -)
In the lecture, you were introduced to reasons for time and cost overruns as well as functional limitations of the results of (software) projects in the context of the CHAOS report. Explain two of these problems in your own words and give a possible solution for each.

Answer:
1. Lack of User Input
Many software projects suffer from a lack of user input, leading to products that may function technically but fail to meet real-world needs or provide genuine user value.This gap between developers and end-users results in features that aren't practical or user-oriented. Additionally, excluding user perspectives can miss insights crucial to tailoring the product effectively. As a result, costly and time-consuming changes become necessary to align the software with user needs, often causing delays.
Solution:
To avoid this, it's important for users to get involved early and stay active throughout a project's lifecycle. A great way to achieve this is by choosing Agile methodologies, where software is developed step-by-step with lots of feedback loops and improvements based on what users actually need. Also, having a flexible project structure can make sure user perspectives are always in focus, with usability testing and regular check-ins with stakeholders. This approach not only boosts the chances of the software meeting its goals but also allows any issues to be spotted and fixed before they grow into bigger problems.

2.Changing Requirements & Specifications
Changing requirements and specs is a common challenge, especially in longer projects where needs tend to shift over time. These changes often lead to project drift, where new features and adjustments are added without updating the timeline or budget. This can cause delays, drive up costs, and result in a compromised final product, as resources keep getting redirected to meet the latest demands. Frequent changes also disrupt workflows and increase overall project complexity, making it harder to keep a clear and focused vision.
Solution:
To manage changing requirements, establishing a flexible and adaptive project management framework is essential. Agile practices, like Scrum, can accommodate change by allowing

work to be reprioritized at the beginning of each sprint. Another approach is to clearly document and set core requirements after an initial phase, making it clear that any new requirements will need to go through a formal change request process, including impact assessments and approvals. This approach maintains project stability and allows teams to focus on high-priority elements without disrupting the project.

Task 2
Learning goal: Become familiar with reasons for failing software projects.
Difficulty: (10, ext, -)
The lecture has given some examples of software errors in which inadequate software engineering has sometimes led to catastrophic consequences. Find out about two of the following software disasters that have attracted major attention in recent years. If necessary,use other suitable sources:

Answer:
1.NHS COVID-19 App Failure in the UK
Effects or Damage

The failure of the NHS COVID-19 app impacted public trust in technological solutions for pandemic management, as skepticism grew towards future health applications. Its ineffectiveness led to delays in contact tracing, allowing the virus to spread more widely and increasing infection rates. Additionally, investments in the app's development resulted in a waste of public resources.

Causes

Several factors contributed to these issues. Compatibility problems on various devices, especially older iPhones, resulted in inconsistent performance. Suboptimal exposure detection algorithms failed to accurately notify users of exposure events, and a lack of user-centered design neglected real-world experiences. Insufficient cross-platform testing left critical bugs, including Bluetooth connectivity errors, undiscovered prior to launch.

Measures

To prevent similar issues in the future, comprehensive testing across various operating systems and devices should be prioritized, focusing on Bluetooth performance due to its role in exposure detection. Adopting a user-centered design approach with iterative feedback through focus groups or beta testing would ensure the app meets user needs. Implementing Agile development practices would facilitate continuous improvements based on real-time performance metrics. Additionally, providing instructional materials within the app, such as tutorial videos and step-by-step guidance, would help users navigate its features effectively. Finally, establishing robust privacy protocols and transparent communication about functionalities would enhance user trust and encourage broader adoption, ultimately improving the reliability of software solutions in public health emergencies.

2. Boeing 737 MAX Crashes

Effects or Damage

The crashes of the Boeing 737 MAX had catastrophic effects, resulting in the loss of 346 lives and a significant erosion of public trust in Boeing and the aviation industry. The grounding of the aircraft model led to widespread operational disruptions, financial losses for airlines, and a ripple effect on global supply chains. Regulatory scrutiny intensified, affecting Boeing's reputation and leading to billions in compensation claims and legal fees.

Causes
The crashes were primarily caused by flaws in the MCAS (Maneuvering Characteristics Augmentation System), which was designed to prevent stalling but malfunctioned in critical situations. Inadequate pilot training on the new system, combined with insufficient communication of the aircraft's operational changes, contributed to the tragic outcomes. Additionally, lapses in regulatory oversight by the FAA (Federal Aviation Administration) allowed Boeing to exert undue influence during the certification process.

Measures
To prevent similar tragedies in the future, Boeing must prioritize rigorous testing and validation of safety systems, particularly those involving automation. Enhancing pilot training programs to include comprehensive education on new technologies and systems is crucial. Strengthening the regulatory framework surrounding aircraft certification, ensuring greater independence from manufacturers, would enhance safety oversight. Fostering a culture of transparency and accountability within Boeing and the aviation industry, alongside improved communication with stakeholders, would rebuild trust and ensure a commitment to safety above all.

Task 3
Learning goal: Reason about software projects.
Difficulty: (15, -, -)
The long-established company reliableSOFTWARE has been commissioned to develop an AIbased management software for job application documents. After one year of development and x KLOC (several 100,000 lines of code), the finished software is tested by end users for the first time. After a short time, however, the test is aborted. The reason given by the manager responsible was that:
"The test run was a disaster. Some of the users didn't know how to use the software at all.
Everything is far too complicated and doesn't meet the needs. It would probably be easier to
redevelop the software than to change everything we don't like."
The project manager at reliableSOFTWARE is surprised, because the last version of the software formally fulfilled all requirements. The project was also executed according to plan, both in terms of the time required and the resources needed.
Answer the following questions in 1-2 sentences: Based on the above scenario, what is a possible cause for the failure of the project? How could this failure have been prevented?

Answer:

A possible reason for failure is that the project team assumed functional requirements were sufficient without considering practical user interactions. By utilizing user experience (UX)

specialists throughout the development, they could have addressed usability concerns. Many users may lack familiarity with the software, so implementing guided tutorials or walkthroughs would have facilitated understanding and improved overall engagement