This is a place holder for the title of this piece

Projekt-/Studien-/Bachleorarbeit

of

Robin Schwenzfeier

Bachelor of Science

of the study program Computer Science

at the

Baden-Wuerttemberg Cooperative State University (DHBW)

Submission Date: 15. Mai 2025

Duration	12 weeks
Matriculation Number	
Course	TINF22B5
Company	NetzeBW
	Karlsruhe
Supervisor	Titel Vorname Nachname

Declaration

I hereby affirm that I have independently written my Projekt-/Studien-/Bachleorarbeit with the topic: **?**This is a place holder for the title of this piece**?** and have not used any sources or aids other than those specified. I also affirm that the submitted electronic version matches the printed version.

_____
Place   Date

_____
Signature

Confidentiality Notice

The content of this work may not be made accessible, either in whole or in parts, to individuals outside of the examination process and evaluation procedure, unless there is explicit permission from the dual partner.

Abstract

Dieses LaTeX-Dokument kann als Vorlage für einen Praxis- oder Projektbericht, eine Studien- oder Bachelorarbeit dienen.

Zusammengestellt von Prof. Dr. Jürgen Vollmer <juergen.vollmer@dhbw-karlsruhe.de> `https://www.karlsruhe.dhbw.de`. Die jeweils aktuellste Version dieses LaTeX-Paketes ist immer auf der *FAQ-Seite* des Studiengangs Informatik zu finden: `https://www.karlsruhe.dhbw.de/inf/studienverlauf-organisatorisches.html` → *Formulare und Vorlagen*.

Stand `$Date: 2020/03/13 15:07:45 $`

Contents

List of Figures

List of Tables

List of Algorithms

Formulars

Abbreviations

Chapter 1

Decision Making

The process of decision-making can be defined as the act of selecting the optimal decision at the given moment. While intuitive and feeling based decision-making is a common practice for managers, team-leaders and project managers, the scientific theory of decision-making offers a structured approach to problem-solving that is based on evidence reasoning. The essence of decision-making is the selection of a course of action from several alternatives. This process is subject to a lot of influences, including personal experiences, biases, the availability of information, business costs, the needs of the user, security concerns, and the specific context of the decision. The modern nature of decision-making has given rose different approaches of diverse theories and models aimed at enhancing the understanding of decision-making processes and their optimization. The following section will present a historical overview of decision-making processes, an analysis of the key factors influencing decision-making, and an evaluation of suitable practical models.

1.1   History: The Beginning

The history of decision-making has undergone significant evolution over time, with a multitude of thinkers making valuable contributions to our understanding of the processes involved in making choices under conditions of uncertainty. From the ancient philosophers to the modern economists, the quest to comprehend the manner in which humans make choices in the face of uncertainty has constituted a central theme within the intellectual discourse. These individuals are notable for their contributions to the development of more contemporary approaches to decision-making. They do not represent the entirety of decision-making, but lay the foundation to what came afterwards.

1.1.1   Adam Smith

One of the key figures in this field of study is Adam Smith, an 18th-century Scottish economist and philosopher whose ideas have had a profound impact on economic theory and, in particular, on decision-making practices.
The roots of decision-making theory and practices can be traced back to ancient civilisations, particularly to Greek philosophers like Aristotle and Plato, who explored the nature of choices and rationality and the application of ethics and their economical use. However, it was not until the Enlightenment that more structured approaches to decision-making began to emerge. This period saw the merging of reasoning with philosophy and scientific principles to better

understand human behaviour, which is the foundation for practical decision-making that goes beyond individual choices to navigate the market.

Adam Smith, renowned for his foundational work, "The Wealth of Nations"[*Adam Smith / Britannica* 2024], made substantial contributions to the field of decision-making. His perspectives were groundbreaking for several reasons and can be distilled into three key concepts:

- Rejection of Mathematical Probability: The rejection of mathematical probability theory is evident in the work of Adam Smith, who was sceptical of the use of precise numerical probabilities in decision-making. Smith argued that real-world data often does not allow for exact distributions and therefore proposed the use of interval estimates to consider a range of potential outcomes. This approach allows for more informed and rational choices to be made despite the presence of uncertainty.

- Interval Estimates: Smith's advocacy for interval estimates represented a notable departure from the prevailing views on the matter. By contemplating a spectrum of potential outcomes rather than a singular probability, decision-makers could more effectively account for uncertainty and variability, thereby establishing a more flexible and realistic framework.

- Critique of Extremes, Smith rejected both the neoclassical view of known probabilities and the Post-Keynesian view of no probable knowledge. Instead, he advocated for interval estimates, thereby offering a balanced approach that recognised the limitations and possibilities of human knowledge.

Smith's ideas were not merely theoretical; he applied his probability approach to a number of economic decisions. For example, he discussed how individuals might choose professions. In making such decisions, people often face uncertainty about future earnings, job stability, and personal satisfaction. By using interval estimates, individuals can weigh the potential outcomes and make more informed choices.

Similarly, Smith's approach had implications for the organisation of insurance markets. Insurance is, in essence, the management of risk and uncertainty. By employing interval estimates, insurers and policyholders are better able to evaluate the spectrum of potential risks and premiums, which in turn facilitates more efficient and equitable insurance markets. Furthermore, Adam Smith's contributions to decision-making have had a profound and enduring impact on economic theory and practice. His emphasis on the practical application of probability in real-world scenarios has influenced subsequent generations of economists and decision theorists. In particular, the concept of interval estimates has found applications in diverse fields, including finance and software management.

The history of decision-making is a testament to the enduring quest to understand and improve how we make choices under uncertainty. Adam Smith's innovative ideas, particularly his rejection of precise numerical probabilities and his advocacy for interval estimates, represent a significant milestone in this journey. Smith's contributions continue to resonate in contemporary economic thought and beyond, offering a balanced and practical approach to decision-making. [BRADY 2013]

1.1.2 Von Neumann

The expected utility theory, developed by John von Neumann and Oskar Morgenstern, constitutes a significant contribution to the field of decision-making theory. This concept is discussed in

detail in the book "Theory of Games and Economic Behaviour"[NEUMANN, MORGENSTERN **and** RUBINSTEIN 1944]. The theory provides a framework for understanding the decision-making processes of rational individuals in uncertain circumstances. The following points represent the fundamental tenets of the expected utility theory:

- Utility Function: The concept of a utility function is fundamental to expected utility theory. Each individual possesses a utility function that gets ascribes a numerical value to each potential outcome, thereby representing the individual's preference for that outcome. A higher utility value indicates a greater preference to a given outcome.

- Expected Utility: In the presence of uncertainty, individuals undertake a calculation of the expected utility for each potential action. This is achieved by multiplying the utility value of each potential outcome by its probability of occurrence, and subsequently summing these products. Giving a value to navigate the decision.

- Rational Decision-Making: The process of rational decision-making is defined as: In accordance with the theory of expected utility, rational individuals select the course of action that optimises their expected utility. This shows that they evaluate both the desirability of the outcomes and the probability of those outcomes materialising.

- Risk Aversion: The concept of risk aversion can be defined as the theory to show attitudes towards risk. To illustrate, a risk-averse individual has a concave utility function (upside down), indicating a preference for a specific outcome over a gamble with an equivalent expected value. Contradicting, a risk-seeking individual has a convex (upside up) utility function.

- Independence Axiom: The Independence Axiom states that one of the fundamental axioms of expected utility theory is the independence axiom. This states that if an individual prefers outcome A over outcome B, they should also prefer a situation that offers A with some probability and another outcome C with the remaining probability, over a situation that offers B with the same probability and outcome C with the remaining probability.

Expected Utility Theory has applied a profound influence across a range of disciplines, including economics, finance, project management, software engineering, and other fields that entail decision-making in the context of uncertainty. Nevertheless, it has also been the subject of criticism and challenges, particularly from the field of behavioural economics, which highlights the fact that human behaviour in reality often deviates from the rationality assumed by expected utility theory. Withstanding the criticism, it has faced, the Expected Utility Theory continues to have a significant influence on the development of decision-making processes in the present day in many major areas. [MOSCATI 2018]

1.2 Factors for Consideration

Many incredible individuals have influenced decision-making over the course of the last centuries and their practices and theories have been derived. Rather than presenting more particular personalities, the content they provided will be set into focus in this scientific work piece.
In the following section, the fascinating methods and various factors will be inspected closely, as they make up the whole picture in the following part of this chapter.

1.2.1 Technical Considerations

When it comes to major upgrades, such as frameworks of applications in enterprise systems, companies need to consider the complex landscape of technical considerations that influence the decision-making process. Upgrading from a framework involves changing the underlying infrastructure, which often increases the cost, effort and duration of development. The introduction of new technical features and functionality can create compatibility issues, resulting in significant disruption to the existing application.
Understanding these technical complexities is critical to mitigating risks such as project delays or budget overruns. An effective upgrade decision requires a comprehensive assessment of current functionality, the potential impact of changes and the risk factors associated with those changes. By evaluating factors such as security, data privacy, maintainability, supported lifecycle, compatibility, performance, etc., companies can better determine the cost-benefit ratio of a given upgrade.
In addition to the immediate technical considerations, companies must also account for the long-term risks and benefits of an upgrade. For example, the choice of a framework or platform can significantly affect the future scalability and flexibility of a system. A well-chosen upgrade enable a system to adapt to evolving technical needs, while bad planned or gut-feeling-based decisions might lock companies into solid infrastructure that limits capabilities and options on a long run. Therefore, it is crucial to consider how the upgrade aligns with the companies, developers and teamleaders strategic goals and whether it provides the necessary flexibility to adapt to future necessities and changes.
Lastly, testing and quality endurance play a major role in the decision-making process, as they impact the upgrading process. Testing is required to ensure that the upgrade performs as expected and that all compatibility issues are resolved before the deployment. This includes not only functional testing, but also performance comparisons, and security reasoning, to identify potential vulnerabilities and weaknesses.
Last but not least, it is important to note that not every factor needs or should be considered. The scope of the upgrade dictates the need, as overthinking and including too many factors can slow down development time and limit the efficiency of developers and other team members. Technical considerations tend to fall into the hands of developers, as they have a deep knowledge of systems, codebasis and software in general. There is no definite answer to the question of which factors are important for technical considerations, but at least every factor should be considered, even if there is no doubt that it is not necessary, because most of the subsequent problems arise from this mistake in development, and can be more costly than analysing the supposedly unnecessary factor up front. [FELDMAN **andothers** 2017]

Security, Data Protection and Access Control

As technology continues to evolve the importance of security measures, data protection, and vulnerability management becomes more and more critical. In the context of mobile development, safeguarding sensitive information is not just a technical necessity but a foundation of maintaining user or company data and ensuring compliance with regulatory requirements. The threats that companies face today are constantly evolving, necessitating a proactive approach to security, especially when sending data between devices and decentralizing some of the company data.

Security is a key factor of technical considerations, as incidents like data breaches can lead to significantly high costs, legal issues, and damage to a company's reputation. Implementing protective functionality, while necessary, also involves costs that must be carefully weighed out against the potential risks and impact of security incidents. This balance is a crucial part of the decision-making process when planning major releases or framework upgrades.

Data protection is intertwined with security, focusing on ensuring that personal, project data and misc sensitive information is handled appropriately throughout its lifecycle. This involves not only securing data at rest and in transit but also ensuring that data is collected, processed, and stored in compliance with relevant regulations. The selection of appropriate security frameworks, tools, and technologies is critical to meeting these standards while aligning with the specific needs of the organization.

Vulnerability management is another essential component and involves the continuous identification, assessment and mitigation of security vulnerabilities within the system. This process is critical to staying ahead of potential threats and ensuring that any vulnerabilities are addressed promptly before they can be exploited. Vulnerability management is not just about keeping data safe and secure, but also about what happens if something goes wrong. It calculates the potential costs of breaches and compares them to actions that may be costly and unnecessary, but could pay off in the near or distant future. Effective vulnerability management requires a thorough understanding of the current threat landscape and the ability to adapt quickly to new challenges. This gives the developer and the team a head start against threats.

Access control and authentication are critical components of a robust security strategy, ensuring that only authorized individuals and systems can access sensitive information and system resources. It involves implementing mechanisms such as role-based access control, where permissions are assigned based on the user's role within an organization, and multi-factor authentication, which adds an additional layer of security by requiring multiple forms of verification. Authentication methods, including passwords, biometrics, and digital certificates, play a major role in verifying users. By managing who can access specific resources, companies can significantly reduce the risk of unauthorized access, thereby protecting the integrity, and availability of data.

Here are some practices for improving security:

- Data Masking and Anonymization: Apply data masking techniques to hide sensitive information in production environments

- Secure API Management: Implement secure authentication and authorization methods to control access to APIs and use input validation

- Implement Secure Coding Practices: Use standard libraries and frameworks, just like Apple's, that are regularly updated to avoid common security issues

- Data Encryption: Encrypt sensitive data both when it is stored and when it is being transmitted, using strong encryption methods

- Access Control: Use role-based access control to limit who can access sensitive data based on their role, and add multi factor authentication

- Regular Security Audits and Penetration Testing: regularly test the security measurements to find and fix vulnerabilities in the application, and perform simulated attacks on the system

- Compliance with Regulatory Standards: Ensure that the application follows data protection laws

- User Data Minimization: Collect only the minimum amount of (user) data needed for the process to work

- Incident Response Planning: Create and maintain a plan to quickly respond to and manage security breaches

- Regular Software Updates and Patch Management: Keep all software, libraries, and dependencies up to date to protect against known security vulnerabilities

In conclusion, addressing security, data protection, and vulnerability management and access control in the planning and execution of major releases or framework upgrades is crucial for creating secure mobile applications. They all contribute to the decision-making, as they form a major factor for consideration. [WEIPPL **and** SCHRITTWIESER 2024]

Maintainability and Future-Proofing

Maintainability, and the interconnected future-proofing, is a critical aspect of software quality and longevity, but not all software is deployed in a business context to last and become a legacy system. Different decisions are made and different technologies are required at different stages of a project. The overall cost effectiveness of software and systems is the focus, but it is not everything. On the contrary, many large organisations are willing to make less profit or even losses to ensure stability, as instability and unavailability could have legal consequences and result in an even greater loss. The concept of maintainability is therefore deeply intertwined with the decision making process and is an important factor in the technical considerations of developers, project managers and other team members to make decisions that are right and that have longevity in mind, even if that longevity is limited in time because of some major upgrade. Object Oriented Software Maintainability (OOSM) specifically refers to the ease with which software written in a object-oriented manner can be modified, extended or adapted to meet new requirements or to comply simply with law. Despite the rapid developed with technologies in the past centuries, there is no universal approach and standardization of metrics, that would ensure that some specific technology is going to be resourceful over the course of the next few months or years.

Graphics rendering cards would be a good example, as they only work to a specific standard that programs can buy and use. However, the rapid development in this area of technology requires state-of-the-art performance to meet the needs of today's power-hungry programs. Most of the graphics cards that came out in the last century are almost worthless and cannot even

run the simplest programs such as a comparatively small Large Language Model (LLM).

In summary, maintainability and future-proofing in software development are complex challenges that go beyond simply making software as robust and rigid as possible. Software systems need to be adaptable to changing needs and conditions, and should adapt to evolving performance requirements and new technological advances, which can change incredibly quickly over months or years. To manage this dynamic, it is critical that developers, project managers, team members and other stakeholders prioritize flexibility and adaptability in their decision-making processes. This means not only choosing appropriate technologies and architectural patterns that allow for scalability and change, but also regularly reviewing and refining those choices as new information and development progress becomes available. Maintainability is, in a sense, an ongoing process, rather than a one-time job. Every decision has its consequences and should be made with longevity in mind, as maintainability and future-proofing are closely linked to technical considerations in decision-making. [SARAIVA 2013]

Backwards Compatibility and Legacy Systems

Backwards compatibility can contribute positively to the dominance of a standard, but it can also slow down development by stifling innovation through adherence to an outdated standard. This dualism shows that backwards compatibility is an important factor in decision-making and should be explored further, especially in the topic of legacy systems.

By definition, backwards compatibility is concerned with the compatibility to a previous version, or previous versions, of a product. Furthermore, it derives standards and specific specs for the way processes, development and software, or just technology, is handled as a whole. The article **?Impact of Backwards Compatibility on Standard Dominance – the Case of Game Consoles?** defines backwards compatibility as "the ability of the new products to subscribe to the benefit of the old product's network" and thus things are more like a consistent product rather than a few products that are somehow aligned in a specific way. In the following, a closer look will be taken upon backwards compatibility in combination, but not only, with legacy systems.

The consideration of backwards compatibility involves weighing its benefits against drawbacks. On the one hand, maintaining compatibility with legacy systems can provide a significant market advantage by preserving existing functionality and software compatibility with older systems, and by ensuring continuity of service for years to come. These seamless transitions keep existing customers loyal and coming back for more.

On the other hand, a commitment to backwards compatibility can also inhibit the need to innovate, by restricting the development of new technologies that are fundamentally incompatible with older versions. For example the changes seen in gaming consoles, where most new games are no more compatible with older ones, but offer incredible graphics compared to last-gen games. Consoles that have prioritized backwards compatibility have been forced to delay the introduction of new features, use less advanced technologies, or even spend countless hours trying to achieve the same level of quality as newer consoles. This is ultimately more expensive, and therefore more costly to the customer, than simply introducing a newer system or console that is incompatible with the older version.

The analogy of backwards compatibility in games consoles shows that this factor has a major impact on decision making. However, decision-making is not a binary judgement and software development is not either. Software development involves many processes and technologies where compatibility is often overlapping, such as the Java programming language, where sometimes three versions of the language are perfectly valid at the same time. For a simple answer, the

legacy version of a program is usually the right choice for major enterprises.

Backwards compatibility and legacy system are important factors in decision-making and can change at once. Providing a bridge between older and newer products enhances consumer satisfaction. Most of the time, legacy options provide this by nature and should be preferred over misc options. [J. DE VRIES, HENK 2009]

Personnel Considerations

When making decisions, it is important to evaluate the human aspects of software development, as well as the costs. These include developer expertise and skills, team resources and domain knowledge. Developer expertise is often deep and specific to a few programming languages, frameworks and development approaches, and decision makers need to trust developers that their technical skills match the required level of expertise. Expertise must therefore be assessed using a variety of metrics, such as years of experience, self-assessment, project portfolios, coding tests and other methods.

In addition to individual skills, collective skill sets of the development team plays also a crucial role in achieving the objectives of a project. A team's skill set must be diverse enough to cover all areas of software development and project management, and while also being great in the collaborative environment. In-Depth, team skills should focus on one greater area like marketing or design, therefore be functional. Identifying the functional divisions of a company and strictly separating them into the needed skills and expertise concentrates the required knowledge and improves effectiveness. At its core, personnel considerations are about emphasising collaboration and contribution across functional teams rather than individual skills, but individual skills should not be therefor neglected, anyways.

Resource management is a critical component of human resource considerations, which in turn contribute to decision making. Ensuring that developers and other team members have the necessary tools, especially software nowadays, to complete their tasks effectively. The allocation of these resources, for example separating resources into functional divisions, creates the needed environment for productivity and team cohesion.

In conclusion, effective decision-making in software development requires a comprehensive evaluation of both, the individual skill set and the team's skills. This involves assessment of expertise, collaboration capabilities, resource allocations to ensure alignment of tasks and personnel and culture. It is important to work in a cultivated work environment that supports open communication and continuous learning to leverage team efforts and closing team members skill gaps. By focusing on these personnel considerations, companies and project managers can make better decisions and reach or exceed project goals. [PODOLNY **and** HANSEN 2020]

1.2.2 Business Considerations

The decision-making process should consider the balance between technical and functional strategies. While technical upgrades focus on aligning systems with the latest technological developments or current legacy systems, functional upgrades aim to optimize business processes by incorporating new features that meet companies requirements. Also integrating functional strategies and other business considerations is imperative to include as general factors for considerations, as they form ultimately the final decision.

In addition to this balance, effective decision-making also involves consideration of various risk factors that can impact project success or failure. These risks may arise from overconfidence in a team's abilities or the anchoring bias, where individuals rely heavily on the first piece of information gathered. It is essential to assess these potential downsides and consider alternative scenarios to ensure a comprehensive evaluation.

Furthermore, decision-making should also take into account personnel considerations, including resource allocation, expertise assessment, collaboration capabilities, and continuous learning opportunities. By considering both technical and functional strategies, as well as personnel factors, companies can make informed decisions that balance business requirements with technological advancements, ultimately leading to successful project outcomes.

There are many more factors to be discussed, so, in the following, various factors that contribute to business considerations will be stated. They range from risks over impact analysis till prioritization.

Risks

Decision-making in a business context involves various risk factors that can significantly impact the budget and amount of work required, which, in the end, leads to the overall success or failure of a project. There are many obvious and even more hidden risks, that all contribute to sub-optimal choices. For instance, overconfidence, as a cognitive bias, may cause team leaders and developers to overestimate their abilities and the accuracy of the information they provide, leading to risky decisions without the necessary consideration of potential downsides. Similarly, the anchoring bias, where individuals rely heavily on the first piece of information they gather, can screw up the decision-making process, as the outcome does not align with reality, because one piece of information, even though being correct, does not fully represent the true nature of the current situation.

Another risk associated with decision-making is the uncertainty and complexity within the business environment. Decisions often need to be made with incomplete or hidden information, that e.g. big tech companies only offer in a limited amount or provide only in one form, like a video or a blog post, which might not be enough for one or all developers. This is making it challenging to predict the outcome of an upgrade and influences the decision-making, by making it harder or impossible to consolidate. This uncertainty can be boosted by external factors such as market volatility, technological developments and regulatory changes, as recently seen with the European Union's (EU) Digital Market Act (DMA). The inability to adapt to changes, such as those imposed by a company's closed-source framework, can lead to decisions that may seem straightforward, but once made, are not.

To mitigate more risks, like communication errors or peer presure, in decision-making, it is crucial to adopt structured approaches that promote rational and informed choices. Uplifting critical opinions in teams and minimizing overconfidence biases are possible solutions to reduce

and mitigate risks. A structured decision-making framework can also help facilitate balanced discussions and ensure all viewpoints are taking into account for a final solution or decision. Some key strategies for improving decision-making in business can be:

- Implementation of a decision support system: Utilize data provided by some analytics, articial intelligence (probability models), and modeling techniques to provide a wider range of insights.

- Encouraging critical thinking and diverse perspectives: Fostering a culture of different viewpoints and team members being able to speak out loud their contra-dictionary thoughts is key to make risks of groupthinking and biases visible.

- Utilization of structured decision-making frameworks: Establish methods to facilitate balanced discussions and ensure all perspectives, including mathematical and technical positions, are considered. (Some frameworks will be shown later in this chapter)

- Continues learning and adaptation: Regularly monitor and evaluate decisions and improve upon mistakes, adjust strategies as new information comes in and situations change.

- Don't be afraid of change: The most critical influence is the nature of the human to stick to a path and not considering a radical change. Embrace the change and act upon new information.

The alignment of decisions with company goals and ethics, the evolution of technology, such as frameworks and operating systems, significant risks arise. Decisions made in isolation and without adequate consideration of key factors in decision-making can result in a misalignment with long-term goals and can cause interruptions or abortion of processes and operations. All the stated risks should be taken into account for a decision to have a successful outcome. [LECH 2016]

Business Impact Analysis

A business impact analysis is a strategic approach designed to assess and plan for potential disruptions, that could pose a threat to the continuity of business operations and plans. It is not specifically a factor for consideration, but much more a sum of factor that can be combined into a single report that can determine the impact. A potential business impact analysis involves an evaluation of potential impacts that various threats, such as (natural) disasters, cyberattacks, or system failures, could have on ongoing company processes. By identifying these vulnerabilities, a business impact analysis allows companies to develop effective strategies, such as disaster recovery plans and redundancy measures, to minimize long-term and short-term damage.
Some of these measures, according to a business impact analysis, could be:

- Identification of critical business functions: A list of all critical business functions and processes, prioritized by importance. This is done by identifying the relevant functions that would have the worst impact if disrupted. The severity can vary from the loss of, for example, holiday photos, to the loss of, for example, critical user data that cannot be recovered. It also varies in time, whether the data cannot be recovered for a day or two or forever.

- Definition of the scope: Identify the components that will be considered in a business impact analysis. For an organisation, this may include facilities, departments, business processes, software, services or even servers themselves. The scope needs to be defined for a specific set of tangible and intangible items in order to determine the exact amount of loss that may occur.

- Determine the impact and likelihood: Evaluation of the potential of consequences of a disruption for each critical business function returns the likelihood of taking place. This includes considering the financial, operational, and reputational effects and costs that could result if confidentiality, integrity, or availability is compromised.

- Identifying dependencies and interconnections: Identify the interdependencies among critical business functions, departments, systems, services and servers. This will help one understanding how a disruption in one area could affect others. Specifically, if one relies on cloud services for data storage, considering the potential impact of a technical failure is imperative!

Incorporating business impact analysis into decision-making enables companies to not only account for potential disruptions, but also to make more informed and evidence-based decisions about resource allocation, risk management, and strategic direction of software development. [WEIPPL **and** SCHRITTWIESER 2024]

Compliance with Regulations

Compliance with regulations is a critical aspect of effective decision-making within companies. It involves aligning to laws, guidelines and standards that governments and alliances have passed. This alignment is not just a legal obligation but also a strategic component that influences how decisions are made. Regulatory compliance reviews within software companies are often insufficient to assess and address regulatory and security requirements throughout the development lifecycle.

One key element of compliance is understanding the landscape of regulations. Companies must be aware of all relevant laws and regulations to mitigate penalties or even the loss of right to make business in a specific sector or at all. For instance, software has to comply with data security standards to be permitted and deployed to customers. Only if the data of the customer is safe and will not be used for unlawful acts, software is permitted to roll out. This understanding helps identifying compliance risks and implementing measures to mitigate penalties.

Furthermore, integrating compliance into the corporate culture is a crucial aspect of decision-making. When compliance is embedded into the core values of the company, it becomes a natural part of decision-making. This means that every decision, from strategic planning to daily tasks, considers regulatory requirements and aims towards an alignment with them. In fact, research has shown that companies view compliance as essential for building trust with customers, rather than just meeting external mandates. Therefore, the company's commitment to compliance is more than just a checklist or process-oriented approach but rather an endeavor that includes individual ethical decisions and cultural environment.

However, regulatory compliance can be burdensome for businesses. In a study of 15 software practitioners from 13 companies, across different industries, it was found that regulatory and security standard requirements were often perceived as overly complex and time-consuming to implement. This demonstrates the need not only for regulatory experts, but also for the training

of existing staff who are not specialised in laws and regulations.

In addition, research has highlighted the importance of addressing compliance requirements throughout the development lifecycle rather than relying only on process-triggered compliance checks. Expecting such checks to catch all possible mistakes is prone to error and should not be a standard in decision-making. Instead, companies must adopt an educational approach that incorporates compliance as both a part of the process and a value within teams, that do not focus on complying with laws.

Effective decision-making integrates compliance with regulations as a backbone and keeps a minimum standard of knowledge for all personnel. Embedding this into the corporate culture is imperative for companies and will pay off on the long run, as companies run less often into costly legal issues. [KEMPE **and** MASSEY 2021]

Project, Development and Operational Costs

To make evidence-based decisions, costs are vital and have to be considered as a factor for businesses. Understanding project and operational costs is essential to analyze the impact of them on the overall financial structure of a project over the course of the livecycle.

Project costs represent the direct expenses associated with the execution of the project, for example paying for materials, labor, equipment and other resources to fulfill the project tasks. Accurate estimation of the project costs is critical, as it directly impacts the budget framework. A misestimation of these costs can lead to budget overflows and harming the business.

Development costs are associated with the creation and development part of the project, which includes products, software, services and creation of systems. These costs typically include classic research&development, designing, (software) development, testing and creating a minimal viable product or prototyping. Development costs are often considered as one-time expenses, but could actually occur again, when e.g. a framework upgrade is necessary or a feature has to be added afterwards.

Operational costs are ongoing expenses to maintain the business operation. These costs may include rent, maintenance or utilities and can be often considered as fix.

Costs have to be estimated and broken down rely on the type of project. In software projects, the costs are mostly salaries, rent or equipment, such as servers, and are not very spontaneous. The costs can be broken down into different types, such as:

- Direct costs: Direct costs are costs that occur less often and are tied to specific decisions like the hiring process of labor, buying materials and supplies. They are mostly included in the project costs.

- Indirect costs: Indirect costs in a project are the support and administrative fees. These fees can include everything from rent rent to salaries. They can be, but do not have to be included in development or operational costs.

- Fixed costs: Fixed costs do not change over the lifecycle of a project and include setup costs. They are also included in the project costs.

- Variable costs: Variable costs are costs that change over time and depend on the amount of work that is being done can vary. Mostly, they exceed the initial estimation, as practical knowledge shows.

The calculation of project, development and operational costs is an ongoing process and can be divided into these three types. Mostly, costs are underestimated and should be divided into their purpose, a cost estimate, a cost overhang or buffer, and a real world example for orientation. The correct estimation of total costs is crucial for decision making as it is an important consideration for businesses. [SCHWARTZ 2023]

Prioritization

Prioritization and decision-making are strongly intertwined and fundamental skills that can influence outcomes in individual and team decisions. This processes involves identifying the most important task and to focus on, especially when resources, such as time, budget or manpower are limited. Decision making, the process of selecting a course of action from a range of alternatives, involves navigating many decisions chronically and most of the time decisions are presented as more important as the rest and thus are hard to navigate. Furthermore, uncertainty is an important factor in prioritisation and makes decisions less predictable. Techniques such as risk assessment and scenario planning are key to improving the quality of decisions when navigating under uncertain conditions.

The first step is to identify which decision can be eliminated first. A structured approach to ranking tasks and decisions based on predefined criteria will help to create a list that can go from most important to least important, or from most urgent to least important. A certain part of the list, e.g. the last 30% of the list, is cut off and does not make it to the final evaluation before the decision is made.

The Eisenhower Matrix popular is a popular prioritization model, which also know under Urgent-Vs-Important Matrix. It categorises tasks into four different quadrants that can be easily displayed. The quadrants are important and urgent, important but not urgent, not important but urgent, not important and not urgent. This layout helps you to get on the right track in terms of importance and urgency.

The downside of the Eisenhower Matrix is that other factors are neglected and when more important or urgent tasks come up, less urgent or important tasks get pushed further down the list, over and over again. In this case, they need to be treated differently and given a separate rating so that they do not end up as ghost tasks that are there and have a right to exist, but just fill numbers. They need to be minimised, delegated to other teams or members of the team, or integrated into another task if possible. The last option would be to delete the task and forget it. In order to make decisions effectively and to establish a solid basis for prioritisation within a workflow, it is important to regularly review and adjust criteria and scores, and to adapt to changing circumstances. The following iterative process is an example of prioritisation:

1. Identify goals and objectives

2. List all tasks and requirements

3. Define the criteria for prioritization

4. Define the criteria for urgency

5. Assign scores to the criteria

6. Evaluate, rank and refine the tasks list based on a chosen model

7. Implement a prioritized plan of approach

8. Review regularly, iterate through this again and make adjustments as needed

In conclusion, prioritisation is an important factor in decision making when faced with one or more decisions. By using structured matrices, such as the Eisenhower Matrix, organisations can improve their decision-making processes and ensure that decisions are made and executed in the right order. Getting the timing right is critical and can have a significant impact on the success of a project. [MAYAN 2024]

1.2.3 User-Centric Considerations

In decision-making, designing products that meet the needs and expectations of users is critical to success. User-centric design (UCD) has emerged over the last few decades as a key approach to ensuring that products are usable, accessible and provide the best experience for the user. At its core, the design approach involves understanding the user's perspective on the given software solution, but also their behaviour, e.g. when using a system, and the limitations that need to be overcome. This shows that user-centred considerations are important factors in decision making and should be looked at closely.

The benefits of a user-centric design extends beyond improved usability, economics and social benefits. By incorporating principles, given by a user-centric design, throught a product's lifecycle, companies can increase the satisfaction of customers, reduce errors and improve overall efficiency. The principle states that developers and UI/UX designers have to consider multiple perspectives and use various methods to gather more insight of how a user uses a product.

These methods include observing how a product is used, what it is used for and how it is used. Prototyping with user scenarios, usability testing, usability evaluation and field studies are also part of the set of methods for gathering user requirements. The gathered requirements are then fed into an analysis that synthesises the results and provides the designer or developer with a proposal for a possible solution. This is, of course, an iterative approach and requires more than just a few iterations, as it falls under classic research and development (R&D).

To ensure the success of user-centred design approaches, it is essential to set clear usability goals and to listen to the needs of users. Implementing required features from a user's point of view without actually asking users and beta testing is pointless and will not lead to success. Defining quantitative and qualitative metrics to assess product usability and user satisfaction will determine if the user-centred approach has been implemented correctly or if some adjustments are needed.

User-centric considerations depend heavily on the context of use, but can lead to major improvements and should always be considered as a critical factor of decision-making. By embracing user-centric design principles and methods, companies can create solutions that meet users needs while driving user engagement, resulting in a business success. [RANNIKKO 2011]

In the following, some other also important factors that contribute to user-centric considerations will be presented.

User Demographics

When designing UI/UX applications, or simply end-user applications, demographics need to be considered as part of the factors that assemble the target audience and therefore displays a

factor for consideration. Targeting a demographic group when designing a software product is imperative, as an evasive selection will lead to dissatisfaction among users who fit the scope of the product. Creating a profile of the user demographic is vital and includes interaction styles, product motivation, multiple demographic patterns and audience research. In addition, the demographics of minority users should not be neglected, although they are small, they make up a larger proportion than expected. These minorities include, for example, older people.

To begin with, users have to be categorized into groups, such as age, gender, profession, income and other suitable categories. It is important not to profile users on just one aspect of their person, as this has a strong influence on the creation of so-called bubbles, which in themselves create many minorities and divide a target group into many smaller prejudiced groups, and therefore have a bad effect on any subsequent use of this data. Anyhow, Different groups have different behaviours and attractions towards a software product and identifying these behaviours is a key part for forming a concrete user demographic.

Each user's motivation for a software product is different, but some overlap can be found between users that form a demographic, even if the connection is loose. While some prefer collaborative, team-based approaches, others simply want to work productively with the software product. These different user satisfaction rewards need to be considered and thought through, as one piece of software cannot satisfy all users. However, there are some motivations that are common to almost all user groups. For example, entertainment is a key motivator for users to use a software product. Seamless interaction and a straightforward user interface that ensures ease of use is important for all users. Some users have special needs, such as disabled children, who have difficulty navigating even the simplest UI. Meeting different needs and satisfying different motivations is key when considering user demographics, no matter how narrow or broad the user demographic may be.

Interaction style dictates the look and feel of the UI and how software applications are used, as demographic groups dramatically influence the look and feel. For example, teenagers are more likely to want a portable version of software on a mobile device. They are also much more likely to use cloud services and want to stay in touch. This results in a specific UI/UX pattern tailored to their needs. Screen real estate, performance, memory and other resources that contribute to the interaction style are limited, and therefore can only appeal to a limited demographic of users, however small or large.

In conclusion, user demographics are critical to consider when designing a software product in order to create effective solutions that are both functional and inclusive. Recognising and addressing the different needs, preferences and behaviours of different demographic groups allows for more targeted and satisfying experiences when using a software product. This leads to more balanced decision making overall and is an important factor in any decision related to users and user experiences. [NICODEMUS 2012]

1.3 Decision-Making Models

Decision-making models are frameworks that help businesses and individuals to navigate processes and achieve a desired outcome. In the following, some models, such as the intuitive or the rationality decision-making model, will be presented.

1.3.1 Intuitive Decision-Making Model

Intuitive decisions can be made by an individual or a group, but do not necessarily follow a strict procedure. Rather, as the name suggests, intuitive decision-making is based on experience of success and the consequences of previous decisions. This cognitive process is based on experience and instinctive feelings rather than on structured analysis of data. In contrast to all the analytical decision making described above, which relies on logical reasoning and quantifiable factors, intuitive decision making draws on data and experience from the past and thrives on holistic thinking.

So far, the approach of this work has been evidence-based, ignoring the biases and subjective views of an individual, but not every decision is important and therefore does not need a meeting or a process. However, sometimes the necessary data is not available, cannot be extracted or quantified, even though a decision is time sensitive and needs to be made now. In this case, managers, developers and others in a decision-making position have to rely on their intuitive experience.

This does not mean that a support system has to be abandoned. On the contrary, analytical tools can give insights about factors that contribute to the decision, but rather than assigning values, one must use their mind to perform a decision right there and now. For instance, designers only make decisions based on their experiences and their skills. Rules and guidelines exist, but are never sufficient for every work piece, as every piece is different and has to be looked at differently. This process involves spontaneous and unstructured methods of analysis, where a work piece is broken down into separate parts, decisions are made on each individual component, and then the parts are reassembled to make up a major, final decision.

Intuition refers to the senses and a subjective view of patterns that create feelings in a being. It is also described as holistic thinking with an immediate insight of a process and giving a thought without knowing the whole picture and the aftermath. This compressed knowledge is a way of quickly accessing patterns of knowledge created in the past. This sudden awareness of information can be used as a factor in applying a decision, but it can also be subject to constant change. The sudden awareness leads to a prediction of how a pattern might evolve, i.e. how a software product might turn out after a decision has been applied.

Intuitive decision-making can only be developed and studied, meaning it has to be gained over time and cannot be replaced by anything. The decision-maker develops an overview of approaches to a specific attempt of solving a problem and adjusts their decision along the way multiple times. One has to avoid getting impatient and resolve biases to navigate decision-making.

The intuitive decision-making model is a model that is mostly based on experiences in a specific field and therefore cannot be applied easily on any other field. It is also made for single, small but still important decisions, such as how to approach the replacement of a software framework. However, it is not suitable for complex decisions involving numbers and values and should not be used on a large scale. Nevertheless, the intuitive decision model is something that everyone uses on a daily basis, and since it exists and everyone uses it, it should not be ignored, but rather approached with a scientific view and motivation to set aside biases and make qualitative

decisions. [Sauter 1999]

1.3.2 Rational Decision-Making Model

The Rational Decision Model is a fairly advanced type of decision model because it identifies the problem, gathers information about the problem and the solution, evaluates alternatives, considers consequences along with the choices made, makes the choices, and monitors how the decision turns out. This multi-step process makes decisions on a logical, informed and objective basis, rather than taking a view and being subjective. It also removes emotion and personal bias and creates a strong contrast between choices to make consequences and differences as visible as possible.

The goal of rational decision making is therefore to reach a decision that best supports objectivity. This path is based on the process of rationality, as its name implies. This concept soberly encourages everyone in a team that the goal is in line with the decision, which is visible through the honesty of the metrics. It avoids impulsive decisions and gets rid of bias and personal feelings.

Due to its truly objective nature of rational decision making, it is very efficient and goals can be achieved with minimal loss of profit but suffer in effectiveness. Making a decision based on deep values and evidence takes time and requires high level staff and managers to make decisions with many calculations, meetings and evaluations. This systemic approach offers high efficiency, as mentioned above, but lacks effectiveness due to the many mechanical and broad actions that need to be taken. As a result, decisions are not made easily and quickly, which is a real drawback of the rational decision-making model. The opposite of the rational decision-making theory is the intuitive model.

As mentioned before, the rational decision-making model is comprised of roughly six or seven steps, depending on which literature is being used as reference:

- Identifying the decision: The first step is to realize that a problem exists and decision is required. The problem and its context then have to be analyzed through questions:

  Why does this decision have to be made?

  When does this decision have to be made?

  What are the consequences of the decision?

  How much effort will the decision take?

  What are obstacles along the way?

  And any other suitable and individual questions.

- Gather information: After the identifying the decision, it is important to gather more relevant information about the environment the decision takes places in. This can involve surveys, interviews, data analysis, research and regulatory.

- Identifying and evaluating alternatives: It is important to list potential alternatives and avoid as much effort as possible to cut costs and increase revenue and profit. Open solutions and products by other companies or outsourcing can be potential alternatives.

- Considering consequences: This consideration is particularly worth looking at because, for the first time, it shows the real costs of a decision and makes the balance of advantages and disadvantages more clear.

- Implementing the decision: Now the decision has to be executed and fully implemented.

- Monitoring, adjustment and retraction: This is the final stage of the rational decision-making model and uses any parameters to determine whether the implementation of a decision is as successful as expected. If necessary, the decision can then be adjusted or even retracted.

Rational decision making is a higher level and more advanced than other decision making models and is based on research and logical evaluation. It is very efficient and based on facts and neglects biases, which makes it independent and objective. Despite the time it takes to make a decision, the model is said to be superior to intuitive decision-making and is directly opposed to it. Rational decision making is most valuable when applied to long-term, complex decisions and tasks, and should therefore only be used in these circumstances. [Uzonwanne 2016]

1.3.3 Bounded Rationality Decision-Making Model

When discussing decision models, it is important to recognise that decisions are always constrained by some factor, be it time, budget or other resources such as manpower. Decisions are therefore limited by the reality and rationality of things. It is important to make a good decision quickly rather than a perfect decision slowly. Sometimes perfection and optimisation can kill the decision-making process and cause disruption to a project.

Being under time and budget constraints encourages people to settle for a much simpler and more efficient approach. For example, a deadline is approaching and no decision has been made and mental stress rises, which kills productivity by limiting the quality of the decision and therefore favouring a more assisted and simpler approach. This is due to heuristics, a mental shortcut that can facilitate problem solving and generalize strategies and approaches to problems. It helps reduce cognitive load and is therefore effective for making immediate decisions. The downside is the tendency to make rush, false decisions.

The concept of bounded rationality is significant in business contexts, as decisions have to be made under conditions of uncertainty and incomplete data. Companies influence their staff to base decisions on evidence and specific information, thereby aligning them with the company goals. For example, managers and team leaders make decisions based on available data and predictions, but due to bounded rationality, they might not always choose the most efficient or profitable option, as they cannot comprehend invisible data, and instead opt for a decision that meets the given criteria and is acceptable, but may not be perfect.

By recognizing the limitations of human cognition, bounded rationality provides a more realistic model for understanding decision-making in a business context. Day-to-day decisions are limited in time and budget and therefore should focus on making a good decision, rather than aiming for the perfect one due to its high cost. [Hernández and Pérez-Ortega 2019]

Bibliography

Adam Smith | Britannica [27 **june** 2024]. URL: https://www.britannica.com/topic/the-Wealth-of-Nations [**urlseen** 31/07/2024] [**backrefpage** 8].

BRADY, Michael Emmett [30 **june** 2013]. *Adam Smith's Theory of Probability and the Roles of Risk and Uncertainty in Economic Decision Making.* Rochester, NY. DOI: 10.2139/ssrn.2287339. URL: https://papers.ssrn.com/abstract=2287339 [**urlseen** 31/07/2024] [**backrefpage** 8].

FELDMAN, Gerald **andothers** [1 **january** 2017]. **?**A systematic approach for enterprise systems upgrade decision-making: Outlining the decision processes**? in***Journal of Engineering, Design and Technology*: 15.6. Publisher: Emerald Publishing Limited, **pages** 778–802. ISSN: 1726-0531. DOI: 10.1108/JEDT-08-2017-0076. URL: https://doi.org/10.1108/JEDT-08-2017-0076 [**urlseen** 16/07/2024] [**backrefpage** 10].

HERNÁNDEZ, José **and** Ricardo PÉREZ-ORTEGA [22 **january** 2019]. **?**Bounded rationality in decision–making**? in***MOJ Current Research & Reviews*: 2, **pages** 1–8. DOI: 10.15406/mojcrr.2019.02.00047. URL: https://www.researchgate.net/profile/Ricardo-Perez-Ortega-2/publication/354784802_Bounded_rationality_in_decision-making/links/64114e8766f8522c38a61966/Bounded-rationality-in-decision-making.pdf [**urlseen** 03/09/2024] [**backrefpage** 25].

J. DE VRIES, HENK, Kramer, Johannes [28 **july** 2009]. **?**Impact of Backwards Compatibility on Standard Dominance – the Case of Game Consoles**? in***Research Gate*: URL: https://www.researchgate.net/profile/Henk-J-De-Vries/publication/303523941_Impact_of_Backwards_Compatibility_on_Standard_Dominance_-_the_Case_of_Game_Consoles/links/6352453296e83c26eb3bea64/Impact-of-Backwards-Compatibility-on-Standard-Dominance-the-Case-of-Game-Consoles.pdf [**urlseen** 29/08/2024] [**backrefpages** 13, 14].

KEMPE, Evelyn **and** Aaron MASSEY [**september** 2021]. **?**Perspectives on Regulatory Compliance in Software Engineering**? in***2021 IEEE 29th International Requirements Engineering Conference (RE)*: 2021 IEEE 29th International Requirements Engineering Conference (RE). ISSN: 2332-6441, **pages** 46–57. DOI: 10.1109/RE51729.2021.00012. URL: https://ieeexplore.ieee.org/abstract/document/9604591 [**urlseen** 29/08/2024] [**backrefpage** 18].

LECH, Przemysław [29 **february** 2016]. **?**Causes and remedies for the dominant risk factors in Enterprise System implementation projects: the consultants' perspective**? in***SpringerPlus*: 5.1, **page** 238. ISSN: 2193-1801. DOI: 10.1186/s40064-016-1862-9. URL: https://doi.org/10.1186/s40064-016-1862-9 [**urlseen** 16/07/2024] [**backrefpage** 16].

MAYAN, Kareem [2024]. *Prioritization Models for Effective Decision-Making.* Savio. URL: `https://www.savio.io/feature-prioritization/prioritization-models/` [**urlseen** 27/08/2024] [**backrefpage** 20].

MOSCATI, Ivan [27 **december** 2018]. **?**The Expected Utility Theory and Measurement Theory of von Neumann and Morgenstern, 1944–1947von Neumann and Morgenstern, 1944**? in***Measuring Utility: From the Marginal Revolution to Behavioral Economics*: **byeditor**Ivan MOSCATI. Oxford University Press, **page** 0. ISBN: 978-0-19-937276-8. DOI: `10.1093/oso/9780199372768.003.0010`. URL: `https://doi.org/10.1093/oso/9780199372768.003.0010` [**urlseen** 01/08/2024] [**backrefpage** 9].

NEUMANN, John von, Oskar MORGENSTERN **and** Ariel RUBINSTEIN [1944]. *Theory of Games and Economic Behavior (60th Anniversary Commemorative Edition).* Princeton University Press. ISBN: 978-0-691-13061-3. URL: `https://www.jstor.org/stable/j.ctt1r2gkx` [**urlseen** 01/08/2024] [**backrefpage** 9].

NICODEMUS, Camille [2012]. **?**Targeting Different Demographics**? in**[**backrefpage** 21].

PODOLNY, Joel M **and** Morten T HANSEN [2020]. **?**How Apple Is Organized for Innovation**? in***Apple Inc.*: URL: `https://www.apple.com/careers/pdf/HBR_How_Apple_Is_Organized_For_Innovation-4.pdf` [**urlseen** 27/08/2024] [**backrefpage** 14].

RANNIKKO, Pirkka [2011]. **?**User-Centered Design in Agile Software Development**? in**Accepted: 2013-10-03T11:46:14Z: URL: `https://trepo.tuni.fi/handle/10024/82310` [**urlseen** 29/08/2024] [**backrefpage** 20].

SARAIVA, Juliana [2013]. **?**A roadmap for software maintainability measurement**? in***2013 35th International Conference on Software Engineering (ICSE)*: **pages** 1453–1455. DOI: `10.1109/ICSE.2013.6606742` [**backrefpage** 13].

SAUTER, Vicki L. [**june** 1999]. **?**Intuitive decision-making**? in***Commun. ACM*: 42.6, **pages** 109–115. ISSN: 0001-0782. DOI: `10.1145/303849.303869`. URL: `https://doi.org/10.1145/303849.303869` [**urlseen** 02/09/2024] [**backrefpage** 23].

SCHWARTZ, Brenna [25 **april** 2023]. *Project Cost Estimation: How to Estimate Project Cost.* ProjectManager. URL: `https://www.projectmanager.com/blog/cost-estimation-for-projects` [**urlseen** 22/08/2024] [**backrefpage** 19].

UZONWANNE, Francis [**november** 2016]. **?**Rational Model of Decision Making**? inpages** 1–6: ISBN: 978-3-319-31816-5. DOI: `10.1007/978-3-319-31816-5_2474-1`. URL: `https://link.springer.com/referenceworkentry/10.1007/978-3-030-66252-3_2474` [**urlseen** 02/09/2024] [**backrefpage** 25].

WEIPPL, Edgar **and** Sebastian SCHRITTWIESER [2024]. **?**Introduction to Security and Privacy**? in***Introduction to Digital Humanism: A Textbook*: **byeditor**Hannes WERTHNER **andothers**. Cham: Springer Nature Switzerland, **pages** 397–414. ISBN: 978-3-031-45304-5. DOI: `10.1007/978-3-031-45304-5_26`. URL: `https://doi.org/10.1007/978-3-031-45304-5_26` [**urlseen** 21/08/2024] [**backrefpages** 12, 17].