UXQCC - FL

Syllabus

User Experience Certification

Foundation Level

Curriculum, Version 4.1 EN

Contents

UXQCC certifications

The certifications, according to the UXQCC standard (User Experience Quality Certification Center), are based on a concept of the Global Association for Software Quality (GASQ), in which experts from various fields have developed the first certification program for user experience (UX). This concept was first published in 2010 and has since been continuously adapted and expanded. Feedback from industry and academia has been obtained and implemented. Complex knowledge about modern didactics, perception psychology, cognitive science, software engineering, UX design, usability tests, and other topics were incorporated in many work steps. This resulted in the development of a lean yet comprehensive new certification program. This UXQCC certification program now consists of three levels. The program starts with the Foundation Level (FL), which is described in this syllabus. This level guarantees a common basic knowledge of all certified persons and thus represents the basis for successful cooperation in the interdisciplinary field of user experience (UX). It is possible to obtain a certificate at a higher level after completing the previous level. The exact current structure of the individual levels can be found on the UXQCC website (https://www.uxqcc.com).

Scientific committee

The scientific committees of UXQCC are composed of leading scientists and representatives from relevant organizations and companies specializing in user-centered design (UCD), user experience design (UX), and usability-related topics. The committee is responsible for supporting the ongoing development of the curriculum in terms of teaching methods and content to ensure that it remains current, relevant, and practically applicable from both scientific and professional perspectives. The current composition of the committee can be found on the UXQCC website (https://www.uxqcc.com).

Authors: Dr. Robert Pucher and Dr. Verena Seibert-Giller, with special thanks to the many people who provided valuable input.

Purpose of the document

This syllabus defines the Foundation Level of the **User Experience Certification Program (UXQCC - FL)** developed by the **International Board for Usability and User Experience Qualification (UXQCC)**. UXQCC makes this syllabus available to accredited training providers so exam questions can be developed for exam preparation and course materials can be produced in the respective national languages. Learners use the syllabus to prepare for the certification exam.

Goals, benefits, and priorities

Goals

Acquire the key qualification "User Experience."

Users of software products or websites must be able to fulfill their goals and tasks efficiently and effectively. The target group must intuitively understand the product's possibilities. The ability to implement usability and user experience is, therefore, a key qualification in many areas. This key competence now makes a significant contribution to ensuring that the user interface of software is easy to learn, efficient to use, and pleasant for the user. This enables the target group-oriented creation of software applications or products that users enjoy.

Benefit

Increase the satisfaction of your customers

The fulfillment of performance expectations and their perception on the customer side leads to increased customer satisfaction. The improved user experience and usability of software, internet, and mobile applications reduce the discrepancy between expected and perceived performance, thus promoting customer loyalty.

Minimize follow-up costs

Usability measures must be taken long before the launch or relaunch of a website or the sales launch of a software product. This prevents damage to the image or loss of visitors or customers and reduces the costs of subsequent improvements and corrections.

Competitive advantages

User-friendliness makes it easier to attract the desired target groups and sets the provider’s products and services apart from those of the competition. Today, it is often not the application that is first on the market that is successful, but the one that customers perceive as user-friendly.

Confidence building

The needs of the users are taken seriously and they feel more comfortable with the software offering. This strengthens the positive attitude towards the provider and the brand and ensures improved customer loyalty.

Satisfied users

Better UX leads to more satisfied users. Positive experiences when using a product or website always strengthen user retention and loyalty.

Better user retention and loyalty

Users who have a positive experience with the UX are more likely to continue using the product and recommend it to others. This also promotes long-term user retention and loyalty.

Increased conversion rates

A good UX can lead to higher conversion rates, especially in terms of completing transactions, filling out forms, or other desired user actions.

Lower dropout rates

A positive UX reduces the likelihood of users abandoning the usage process. User-friendly interfaces minimize frustration and increase the willingness of users to complete the process.

Efficient use

An improved UX makes operation easier and enables users to complete tasks more efficiently. This leads to more productive use of the product or platform.

Positive brand image

A positive UX helps build a positive brand image. Users usually associate a good user experience with a high-quality and user-oriented brand.

Reduced support requests

User-friendly products lead to less confusion and fewer problems, thereby reducing the number of support requests. This saves resources and improves the efficiency of customer support.

More successful launch of new products

When introducing new products or functions, an improved UX facilitates acceptance and success in the market.

Barrier-free access

An improved UX considers the needs of different user groups, including people with different abilities. This promotes accessibility and inclusive access.

Competitive advantages

In industries where competition is intense, a superior UX can become a decisive competitive advantage and persuade customers to choose a product or service over rival offerings.

Lower costs for updates

An improved user experience (UX) reduces the likelihood of user errors and problems, thereby decreasing the costs for necessary error corrections in the interface and updates.

Positive user feedback and ratings

Satisfied users tend to leave positive feedback and publish positive reviews. This can further strengthen the image of the product or service.

Focal points

Human capabilities and their impact on the design of interfaces

The human perception processes, software ergonomics, the differences between online and offline behavior. Mental models. Human capabilities and effects on UX design.

User-Centered Design

Design principles for software products, GUI design, storyboard, mockups, prototyping, card sorting, the use of personas.

Standards, norms, and guidelines on accessibility

Overview of the most important usability-relevant standards, norms (ISO), and the W3C guidelines for barrier-free access to the product.

Usability and the user experience lifecycle

Process-oriented approach to ensure the usability of a system. Optimization of the development processes.

User Research

Systematic techniques to gain insights into the needs, behaviors, and preferences of users in order to better adapt products and services to their requirements.

Evaluation / Methods

Usability tests and other methods for collecting usability data.

Concept

For user experience projects to be successful, everyone involved must have access to a shared basic knowledge. A ***common vocabulary*** and a ***common understanding of key concepts*** are particularly important. Without this, misunderstandings often occur if similar terms are not understood correctly or if terms are not linked to the same clear concepts.

The UXQCC Foundation Level ensures that definitions and fundamental knowledge about people who use a system (e.g., perception, mental models, error handling) as well as about the techniques for developing interactive systems (e.g., interaction styles, modeling methods, dialog design) are acquired. Another important part of the UXQCC Foundation Level syllabus is the general standards and norms that UX experts need to be familiar with.

Software products that users appreciate are never created just by chance. A high level of user experience is ensured through the consistent application of development principles. These principles begin with user research, requirements engineering, UX specifications and go on to prototyping and evaluation, usability tests, implementation, and any necessary iterations of the sub-processes. The knowledge and skills needed to do so are described in this curriculum.

In addition, it is recommended to ensure the applicability of the knowledge through practical exercises.

People who have successfully completed the UXQCC Foundation Level can prove that they know the important methods in the entire field of user experience and have the necessary basic knowledge. This is the foundation for a successful UX implementation in any product.

UX and Artificial Intelligence: Artificial Intelligence is already transforming the ways we work in system, product, and service design. AI supports, for example, the analysis of data from research or evaluation activities and can even generate initial design drafts. However, a key prerequisite for creating successful, user-centered systems is that AI remains a supportive tool applied by UX professionals — not an uncontrolled, full-fledged replacement for human activity and reflection. To ensure this, a fundamental understanding of how AI operates, such as the one provided at this foundation level, is indispensable.

Target groups

The basic level of the UXQCC Foundation Level certification program is suitable for all persons and professional fields involved in developing software and mobile or web applications. Among these groups are software developers, GUI programmers, SCRUM masters, project managers and project staff, organizers, managers, UX specialists, IT auditors, quality assurance officers, and persons entrusted with software quality management.

UX designers and interaction designers: People responsible for designing user interfaces, interaction processes, and visual elements.

UX Researchers: People who conduct user research to better understand user behavior, needs, and satisfaction.

Front-end developers: People responsible for implementing user interfaces and interaction designs in code and who want to integrate UX principles into their work.

Product managers: People responsible for the strategic planning, development, and optimization of products and who want to understand the importance of user-centricity.

Web developers and software developers: People involved in developing websites or software applications who want to improve the user experience.

Marketing professionals: People responsible for marketing products or services who want to develop a deep understanding of the user experience in order to optimize marketing strategies.

Management consultants: People who offer consulting services in the field of UX and would like to validate their expertise through certification.

Start-ups and founders: People who are developing a new product or platform and want to ensure their user experience is effective and engaging.

Teachers and trainers: People who teach UX principles or conduct training for teams or students.

Career starters in the field of UX: People who want to enter the UX industry and validate their basic knowledge and skills.

Prerequisites

Initial experience in developing technical products, especially software, is an advantage but is not necessarily required. The Foundation Level is generally aimed at all people who work in the development and marketing of user-friendly products.

The Foundation Level certificate is a prerequisite for taking the UXQCC Advanced Level certificate examinations.

© UXQCC - FL 4.1 Syllabus, October 2025

Learning objectives/cognitive levels of knowledge

Each section of this syllabus is assigned to a cognitive level.

The following levels are predominantly covered in the foundation level.

K1 Knowledge/knowledge: Knowledge of concrete details such as terms, definitions, facts, data, rules, regularities, theories, characteristics, criteria, and processes; learners can recall and reproduce knowledge.

K2 Understanding: Learners can explain or summarize facts in their own words; give examples, understand relationships; interpret tasks. This includes transferring content from one type of representation to another (e.g., words to a graphic), explaining and summarizing content, and finally deducing future developments from content.

Levels K3, K4, K5, and K6 are not included in the Foundation Level. These levels are covered in the higher levels.

K3 Application: Transfer of knowledge, problem-solving; learners can apply what they have learned in new situations and use or abstract abstractions without being asked. Ability to use the material learned in new concrete situations, e.g., by applying certain rules, laws, theories, etc. For example, a computer science student should be able to program various sorting algorithms in an assembly language, or a mathematics student must be able to carry out a mathematical proof according to the applicable rules.

K4 Analysis: Learners can break down a problem into individual parts and thus understand its structure; they can uncover contradictions, recognize connections, and derive conclusions as well as distinguish between facts and interpretations. This includes, for example, identifying the individual elements, determining the relationships between them, and recognizing the design principles. The "Analysis" level requires a higher level of ability than "Understanding and applying" because it assumes that both the content and the structure of the subject matter have been understood. For example, the learning activity of art history students to identify the style-defining elements of a painting and assign them to a specific art historical period is included in this level.

K5 Synthesis: Learners can build a new structure from several elements or create a new meaning, suggest new solutions, and develop new schemes or well-founded hypotheses.

K6 Assessment: Learners can assess the value of ideas and materials and thus weigh alternatives, select, make, and justify decisions, and consciously transfer knowledge to others, e.g., through work plans.

The exam

The examination for the Foundation Level certificate is based on this syllabus. An examination question can cover material from several chapters of the syllabus. All areas of this syllabus can be tested.

Examinations can be taken online immediately after an accredited training course, but also without training, e.g., at a test center or at home. Almost all examinations are held online, regardless of where they are taken.

To ensure a consistent quality of the examination, the examination is held by GASQ GmbH. The options for taking an exam can be found on the GASQ website (www.gasq.org).

Accreditation of trainers

Training providers offering an official exam preparation course must be recognized and accredited by UXQCC. Accredited training organizations or trainers are provided with basic training materials to ensure worldwide consistent training. These materials can be adapted by the trainer. A list of accredited training organizations and trainers is available on the UXQCC website (https://uxqcc.com). UXQCC trainings are available worldwide in person or online.

Level of detail of the curriculum

The aim of the curriculum is to ensure internationally consistent learning, teaching, and assessment. To achieve this goal, this curriculum contains the following components:

- General learning objectives that describe the intention of the basic level.
- Content to be taught with a description.
- Learning objectives for each knowledge area that describe the observable cognitive outcome of the training and the attitude to be achieved by the participant.
- A list of terms that the participant should reproduce and understand.
- A description of the important concepts to be taught, including recognized literature, norms, and standards.

The syllabus is not a complete description of the areas of knowledge. It merely reflects the necessary scope and level of detail relevant to the teaching objectives of the Foundation Level.

Structure of the curriculum

The syllabus consists of main chapters and sub-chapters. A table for each chapter shows the subject-specific competence level and specifies the minimum recommended teaching time that should be spent on this content in an accredited course. In addition, precise learning objectives are given. Finally, the terms that need to be explained in relation to the content of the chapter (context-specific) or explained in more detail for course participants are given.

K-L ⏱	Competence level K1 - Recommended time frame for the topic in minutes, adapted to the previous knowledge of the participants
🎯	Learning objectives
📌	These are the terms that must be explained (context-specifically) or explained in more detail in the chapter, based on the content written in bold in the text. Participants must be able to understand, explain, and reproduce these as part of the examination.

1 Basics of user experience and usability

1.1 Introduction

K-L	K1 - 30 minutes
	LO 1.1.1 Classify and define user experience (K1) LO 1.1.2 Classify and define UX design and graphic design (K1) LO 1.1.3 Interdisciplinarity (K1)
	Usability, learnability, efficiency, memorability, errors, satisfaction, context of use, perspective adoption, graphic design, UX design, ISO

1.1.1 The classification of user experience (UX)

Today, user experience is a decisive success factor in developing and designing software and internet applications, mobile applications, and other interactive systems, such as ticket machines or display systems. Although extensive functionalities are often available in systems, they cannot be used by the user or can only be used with great effort due to complicated operability or a lack of findability.

Outstanding usability or user experience ensures that products and applications are easy to use and suitable for their intended purpose in the intended context of use. The included functions should be easy to learn, understand, and use.

According to the International Organization for Standardization (ISO), usability is **"The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use."** This places the usability and suitability of a system in the user context in a specific user context.

In addition, some organizations have mentioned other characteristics decisive for usability or user experience.

Jakob Nielsen, for example, names the following target values as a measure of the quality of user interaction with a system:

Learnability: The system should be as easy to learn as possible. Unnecessary training effort is reduced.

Efficiency: The system should be efficient to use in terms of time and a high degree of productivity should be possible.

Memorability: The system's operation should be easy to remember so that the system can be used after a later return without having to retrain.

Error: The system should have a low error rate.

Satisfaction: The system should give the user a feeling of satisfaction. Users should be able to easily realize their needs and wishes regarding the system and its capabilities.

1.1.2 Difference and synergy between graphic design and UX design

UX design has little to do with the visually appealing design of products. A website perceived as ugly can have a high degree of usability. Conversely, a site perceived as "beautiful and appealing" can have completely inadequate usability.

However, the design should not be neglected under any circumstances, as users usually interpret an appealing design as an indication of good usability. For example, users decide within the first few seconds whether they like a website or not. However, this decision to "like" or "don't like" is made unconsciously. If the user leaves the website because they don't like it, all usability measures no longer have any effect. The aesthetics of a website also contribute to usability in a more subtle way, as they promote the user's well-being and thus increase their satisfaction.

The author of the website or software application must determine what purpose the product serves and which aspects are, therefore, important. Finally, websites for marketing purposes, for example, prefer design over functionality often. Usability must, therefore, always be adapted to the relevant context to achieve its goals.

The knowledge and skills required by the two distinct professional groups, "graphic designers" and "UX designers" differ significantly. This syllabus covers the necessary basics of user experience. In practice, graphic designers and UX designers usually work together to ensure that both the visual aesthetics and the user experience of a product are optimized. Both disciplines contribute to creating a holistic and appealing product.

A high degree of usability in development is usually achieved through an iterative process. This process is often referred to as "user-centered design." The integration of usability principles into a product's entire life cycle helps maximize user satisfaction and efficiency and identify and resolve problems at an early stage.

The repeated and constantly improved analysis and involvement of the target group through user research and usability tests results in products with increased user-friendliness. New technologies, such as the inclusion of AI technologies, require a constant review and expansion of the methods for developing usable products.

However, the usability of a system is just as dependent on the characteristics of the user, i.e., on human abilities, characteristics, and objectives. Imagine software for managing music. A professional DJ, for example, has different expectations when it comes to managing their music than, say, a hairdresser who just needs some background music in their store. A private user who wants to manage their music on a PC and play it via a sound system also might have different needs.

The **"context of use,"** i.e., the environment and the requirements that arise from the user's needs, significantly influence software design. The usability of a particular piece of software is interpreted completely differently by the three groups described above.

The term **"perspective taking"** stems from psychology. It refers to the cognitive ability to understand and consider another person's thoughts, feelings, beliefs, and experiences, enabling one to see situations from their point of view. It involves empathizing with others and imagining oneself in their position to gain insight into their perspectives. This ability develops early in childhood and to varying degrees in different people. It is particularly important for good usability that the need for perspective-taking is recognized and the other person's point of view is analyzed.

1.1.3 Interdisciplinary approaches

Interdisciplinary approaches must be considered in the entire field of UX. These interdisciplinary aspects, among others, make the topic appear complex, as there are rarely people who have the relevant knowledge and skills in all areas concerned. For example, software engineers rarely have psychological knowledge, while graphic designers lack knowledge about the organization of software development. Training is correspondingly complex, as different people who work together must have a common basic knowledge to be able to work together efficiently. Mutual recognition and an understanding of the effects of the specific challenges of the different sub-areas form the foundation of any collaboration.

Biology and sensory physiology: Biological principles such as visual perception ("seeing"), auditory perception ("hearing"), or haptic perception ("the sense of touch") - the active feeling of an object through the integration of all skin senses and depth sensitivity.

Psychology: Application of theories of cognitive processes, Gestalt psychology, and empirical analysis of user behavior

Sociology and anthropology: Interaction between technology, work, and organization

Computer science and software engineering: Application design and development of human-machine interfaces

Graphic - Design: Graphic design of interactive applications

1.2 The benefits of usability

K-L	K2 - 20 minutes
	LO 1.2.1 Be able to demonstrate the benefits for users and the economic benefits of usability for providers (K1)
	Increased productivity, competitive advantages, cost reduction

Today, applications and applications must meet customer expectations and be easy and intuitive to use and understand. Generally speaking, usability is an extremely effective tool for reducing costs. Usability helps developers produce simpler products. Simpler products are, in turn, easier to sell and easier for customers to use.

Usability tests are an effective way to save time during the development and implementation of software websites and reduce the pressure on the development team. The test makes it possible to determine in advance which criteria are important for the user and which are less important. The test also detects weaknesses and errors at an early stage, which can cause enormous problems in a later development phase. The earlier an error or usability issue is found, the less effort is required to solve it.

The use of User-Centered Design, an iterative process for improving product usability, results in many monetary and non-monetary benefits. These can be quantified for three basic areas:

- **Increase in productivity**
- **Reduction of total costs incurred for the product, especially maintenance costs**
- **Increased competitiveness**

This is made possible by the following points:

- **Target group-oriented** development right from the start. minimizes the need for subsequent improvements.
- **Decrease the number of design** iterations needed.
- **Avoiding** the development of **unnecessary** functions.
- Early clarification and communication about the design with the client.
- **Increase customer satisfaction.**
- Usability test results can help in making **strategic business decisions** as to whether and how a development should be continued.
- **More efficient solutions.**
- **Reduced training effort** because of easy-to-use solutions.
- **Reduced support and call center effort** because of easy-to-use solutions.
- **Fewer user errors** and less troubleshooting effort because of easy-to-use solutions.

- The **optimal mapping of the required workflows** in the software system in relation to the needs of the users makes customers more satisfied.
- Focus on the **real and clearly specified user needs** (and not just on the mostly vaguely formulated expectations of the buyers).
- Inclusion of **relevant industry standards and norms.**
- Development of **target-oriented, innovative solutions** based on knowledge of users' real needs.
- Application of **interdisciplinary knowledge** and **interdisciplinary** methods.
- Incorporating **experience and know-how from other domains.**
- Techniques for the **potentiation of innovations** with the involvement of users or based on expert knowledge.

1.3 Problems caused by poor usability

K-L	K2 - 15 minutes
	LO 1.3.1 Using examples, describe the problems caused by inadequate usability (K2)
	"Nice to have" factor

Unfortunately, UX design is often a **candidate for deletion** in a project budget. Similar to documentation or quality assurance, **usability** is seen as a **"nice to have"** factor in the development process and is, therefore, frequently perceived as a low-priority goal by management.

Usability contributes directly to the success or failure of a software application or website. With mobile apps in particular, the user decides after a short time whether the app is "usable" or not. Unusable apps are usually uninstalled immediately.

It also has a direct impact on store sales in online shops. Central shop functionalities that cannot be found, such as the shopping cart and the path to the checkout, or products that are insufficiently described or difficult to find in the product range, result in **lost sales**.

In the case of operational solutions, a lack of usability often leads to dissatisfied employees and always to a higher workload and lower productivity.

Poor usability has an impact on training costs. The poorer the usability, the more costly the training required to reduce the problem. Often, users only read the manual supplied in exceptional cases, namely when they are stuck but absolutely need the software.

In addition to the direct consequences, a poor user experience also impacts the associations linked to a company. If the usability of software in a vehicle is inadequate, both the vehicle itself and the vehicle manufacturer tend to be associated with negative attributes.

Usability problems are more dangerous in safety-critical applications. For example, medical devices whose settings or operations are incorrect can cause harm to patients. The controls in aircraft cockpits must be easy to understand and operate, even in stressful situations. Also, the controls in cars and the status displays directly influence driving safety. Critical status displays in nuclear power plants must always be understood quickly and without detours and must accurately reflect the necessary system parameters.

1.4 Definitions and basic explanations of terms

K-L	K2 - 40 minutes
	LO 1.4.1 Being able to define usability, user experience (UX), and customer experience (CX) (K1) LO 1.4.2 Being able to define user interface design and interaction design (K1) LO 1.4.3 Describe the user-centered design process (K1) LO 1.4.4 Describe the approach and areas of application of software ergonomics (K1) LO 1.4.5 Being able to describe universal design (K1) LO 1.4.6 Explain the influence of social rules on the user experience (K2)
	User experience, usability, software usability, customer experience, user-centered design, MMI, MCI, HCI, software ergonomics, hardware ergonomics, user interface, user interface design, interaction design, universal design, interdisciplinary approaches,

Although it's possible to provide a clear definition of terms, there is often a lack of clarity observed in practice. This is particularly true for the terms **usability, user experience, customer experience,** and **user-centered design,** as different experts in different fields use these terms in varying ways. As a result, clear and strict definitions are not always clearly visible in practice.

1.4.1 Usability, user experience, and customer experience

Usability

The term **usability**, often referred to as **software usability**, relates to the quality of use of a system when interacting with it. A software-related user experience includes all interactions between the user and the software, such as the user interface, interaction design, efficiency, usefulness, and overall satisfaction when using the software.

Various factors are important here, the most important of which are psychological. People evaluate machines similarly to how they would evaluate other people. Therefore, software is generally rejected as soon as it triggers a negative feeling such as "Am I too stupid to understand?". See also chapter 3. Standards for the ISO definition of Usability.

User Experience

User experience (UX) - in addition to usability - is not just the user's experience with the product itself but a holistic approach with all experiences that are in any way related to this product. User experience encompasses the overall emotional, psychological, and practical response of individuals when interacting with a product, service, or system, focusing on elements such as usability, accessibility, and satisfaction.

From the desire to own this product to its final use, all experiences and the associated feelings are included in the evaluation. In addition to the actual usability of a product, factors such as trustworthiness, emotion, and aesthetics are also considered. Using a product should trigger a feeling of "joy of use." In this way, **user experience** also sublimates the emotional appeal of software. **User experience,** therefore, represents the experienced quality of the user's interaction with the contact point of the technical device. See also chapter 3. Standards for the ISO definition of User Experience.

Customer Experience

The term **customer experience is** even broader. It refers to all the perceptions, interactions, and emotions a customer has during their entire interaction with a company, product, or service. Customer experience, therefore, encompasses all points of contact a customer has with a brand, from the initial contact to the conclusion of a transaction and beyond. The **customer experience** can take place across different channels ("multi-channel experience" or "omni-channel experience"), including physical stores, online platforms, social media, phone calls, emails, and other means of interaction. All points of contact between the company and the customer are referred to as "touchpoints."

1.4.2 User interface design and interaction design

Interaction design (IxD) refers primarily to the activities involved in designing human-machine interfaces.

The term user interface design or interface design (UI design) refers to the precise design of an interface for interactions - i.e., the user interface (UI). The user interface can be any interface. For example, the user interface can be a website, an app, software, or a display (e.g., in a car or on a coffee machine).

Here, too, the terms are not clearly defined and are often used synonymously or in different or broader contexts.

1.4.3 User-centered design

The term user-centered design (UCD) or user-centered design process refers to an iterative method for developing user-friendly products. The focus is placed on the user. User research and requirements analysis are carried out to understand needs. Design solutions are usually created and tested on this basis. User feedback is integrated in several iterations to make improvements. The process includes aspects such as information architecture, interaction design, and visual design. The final products are constantly adapted to the users' needs, even after the initial deployment (delivery), to ensure an optimal user experience in the long run. The UCD process emphasizes continuously integrating the user perspective to create effective, user-centric solutions. (For details, see point 4. User-Centered Design)

1.4.4 Human-centered design

Human-centered design (HCD) is an approach to problem-solving that focuses on understanding and addressing the needs, behaviors, and experiences of the people who will use a product or service. It involves empathy, iteration, and co-creation, ensuring solutions are tailored to people's challenges. By prioritizing the end user throughout the design process, HCD creates solutions that are both functional and meaningful (See also User-centered design).

1.4.5 Ergonomics and further definitions

In terms of software ergonomics, **human-machine interaction** can be pinned down to human-computer interaction **(MCI)** or **human-computer interaction (HCI).** The latter term is often equated with software ergonomics. Ultimately, however, HCI includes both **software and hardware ergonomics**.

While hardware ergonomics adapts tools (input and output devices) for human-computer interaction to the physiological characteristics of humans, **software ergonomics** aims to achieve an adaptation to the cognitive abilities of humans, the ability to process information. It describes and evaluates **user interfaces** for **human-machine interaction**.

Both focus on the **user interface**, which contains the following components and features:

- The user interface with the user’s input options and the computer system’s output options.
- The rules for input and output processes on the user interface.
- Systems to support **human-computer communication.**

In terms of software ergonomics, “input and output processes” does not mean the use of technical devices such as a mouse or keyboard, but rather software-side dialog design processes such as menus, command dialogs, or input forms. This involves the reciprocal influence between people and computers (interaction). It provides guidelines for the user-friendly design of software and interactive systems.

1.4.6 Universal design

Universal design (also known as universal usability) involves creating products, environments, and systems that are accessible and usable by people of all abilities, ages, and backgrounds without the need for adaptation or specialized design. It emphasizes inclusivity, flexibility, simplicity, and intuitive usability, aiming to accommodate the widest possible range of users and minimize barriers to access. By incorporating universal design principles, products and environments become more user-friendly, enhancing usability and user satisfaction for everyone.

Principles of universal design:

Principle 1: **Equitable Use:** The design is useful and marketable to people with diverse abilities.

Principle 2: **Flexibility in Use:** The design accommodates a wide range of individual preferences and abilities.

Principle 3: **Simple and Intuitive Use:** Use of the design is easy to understand, regardless of the user’s experience, knowledge, language skills, or current concentration level.

Principle 4: **Perceptible Information:** The design communicates necessary information effectively to the user, regardless of ambient conditions or the user’s sensory abilities.

Principle 5: **Tolerance for Error:** The design minimizes hazards and the adverse consequences of accidental or unintended actions.

Principle 6: **Low Physical Effort:** The design can be used efficiently and comfortably with minimal fatigue.

Principle 7: **Size and Space for Approach and Use:** Appropriate size and space are provided for approach, reach, manipulation, and use, regardless of the user’s body size, posture, or mobility.

The differences between Europe and the USA in understanding the meaning of the terms are sometimes considerable. “Universal design” originates from the USA. In Europe, the term “design for all” is often used. For this reason, “design for all” as a European strategy means integrating different groups of people without forcing uniformity.

Where required, universal design also includes aids for specific groups of people with disabilities.

1.4.7 Further relevant terms in the field

In addition to the central definitions such as usability, user experience, and human-centered design, there are a number of further concepts that are frequently used in the UX domain. These include, for example, ergonomics, universal design, inclusive design, participatory design, and sustainable UX.

Such terms broaden the perspective and highlight the interdisciplinary nature of user experience. They help practitioners to align design processes not only with functionality and efficiency but also with ethical, social, and environmental considerations.

© UXQCC - FL 4.1 Syllabus, October 2025

Design Thinking
Ein iterativer, menschzentrierter Innovationsprozess, der auf Empathie, Co-Creation, Ideengenerierung, Prototyping und Testen setzt. Er wird breit eingesetzt, um komplexe Probleme kreativ und interdisziplinär zu lösen.

Service Design
Ein ganzheitlicher Ansatz zur Gestaltung vollständiger Service-Erlebnisse über mehrere Touchpoints (digital und physisch). Service Design integriert Geschäftsprozesse, Customer Journeys und UX, um konsistente und zufriedenstellende Erlebnisse zu schaffen.

Experience Design (XD)
Ein Oberbegriff, der User Experience (UX), Customer Experience (CX) und Brand Experience (BX) umfasst. Er betont die Gestaltung der gesamten Interaktion zwischen Nutzern, Produkten, Services und Marken.

Lean UX
Eine leichtgewichtige UX-Methodik, die eng mit agiler Softwareentwicklung verbunden ist. Sie fokussiert auf schnelles Experimentieren, kurze Feedbackzyklen und minimale Artefakte zugunsten von validiertem Lernen.

Agile UX
Die Integration von UX-Methoden in agile Entwicklungsprozesse (z. B. Scrum oder Kanban). Agile UX legt den Schwerpunkt auf kontinuierliche Zusammenarbeit zwischen Designern und Entwicklern innerhalb kurzer Iterationen.

Participatory Design / Co-Design
Ein Gestaltungsansatz, bei dem Endnutzer:innen aktiv als Mitgestalter:innen in den Designprozess eingebunden werden. Dadurch werden Lösungen enger an realen Bedürfnissen orientiert.

Value Sensitive Design (VSD)
Eine Methodik, die menschliche Werte wie Privatsphäre, Fairness, Autonomie oder Nachhaltigkeit systematisch in den Gestaltungsprozess einbezieht. Sie verbindet technisches Design mit ethischer Reflexion.

Inclusive Design
Eine Designphilosophie, die Produkte und Services so gestaltet, dass sie für möglichst viele Menschen unabhängig von Alter, Herkunft oder Fähigkeit nutzbar sind. Sie geht über Barrierefreiheit hinaus und berücksichtigt Diversität und Gleichstellung.

Sustainable UX / Green UX
Ein Gestaltungsansatz, der ökologische Nachhaltigkeit in den Vordergrund stellt. Dazu gehören energieeffiziente digitale Lösungen, langlebige Systeme und die Förderung ressourcenschonender Nutzung.

Conversational UX
Die Gestaltung natürlicher, dialogbasierter Interaktionen (z. B. Sprachassistenten, Chatbots, multimodale KI-Agenten). Im Fokus stehen Verständlichkeit, Vertrauen und Kontextsensitivität.

AI-Augmented UX
UX-Konzepte, die durch Künstliche Intelligenz unterstützt oder verändert werden. Beispiele sind adaptive Interfaces, prädiktive Personalisierung, automatisierte Inhaltserstellung oder KI-gestützte Usability-Tests.

1.5 Social rules and user experience

1.5.1 Expectations about the behavior of machines

Humans are social beings. For every human-machine interaction, this means that humans **expect a human type of social behavior from the machine**. This can be described most simply with the following sentence: “Software should behave like a good friend.” An overview of the most important rules:

Good friends ...

... try to make **suggestions** on how to proceed if you don’t know what to do.

... make sure that the other person **never feels incompetent** or stupid.

... know the **needs of** a friend.

... speak a **language** that is understandable.

... only suggest what is **needed now** (and know what that might be).

... **do not** ask **pointless or incomprehensible** questions.

When developing an interaction, you should always ask yourself: “How would good friends help me now if I don’t know xy?”

1.5.2 Deceptive/Manipulative Patterns

Deceptive designs (dark patterns) mislead users, create pressure, or obscure choices. Examples: nagging, obstruction, confirmshaming, pre-ticked consents.
Legal note: In the EU, the **Digital Services Act (DSA, Art. 25)** prohibits manipulative interface practices. Product teams must ensure clear information, simple opt-outs, no misleading defaults.
Practice: Include explicit “deceptive design checks” in reviews (opt-out path, wording, hierarchy, timing).

2 Human perception and information processing

2.1 Visual perception

K-L	K2 - 60 minutes
	LO 2.1.1 Being able to explain the subjectivity of perception and its effects (K2) LO 2.1.2 Describe the reading process and effects on user interfaces (K2) LO 2.1.3 Know how visual perception works in the foveal and peripheral areas (K2) LO 2.1.4 Explain the resolving power of the eye and the effects on UX (K2) LO 2.1.5 Being able to describe fields of vision (K1) LO 2.1.6 Being able to assess color associations and color effects (K1)
	Photoreceptors, cones, rods, central field of vision, main field of vision, peripheral field of vision, binocular field of vision, retina, retina, lens, fovea centralis, visual fossa, visual axis, foveal vision, peripheral vision, color association, color effect, eye tracking

2.1.1 Subjectivity of perception

On the one hand, perception is determined by the physical nature of the organs of perception and, on the other, by the unconscious pre-processing of perception in the brain. By far, the greatest influence is exerted by the pre-processing of perception in the brain. Both individual habits (=learned) and characteristics of the sensory organs and the brain (=sensory physiology, computing speed) play a role here.

Searching for patterns or words

Searching for patterns or words significantly influences visual perception in various ways. When searching for “something specific,” attention is usually focused unconsciously but specifically on certain information or objects. This leads to these elements becoming the focus of perception while other aspects are neglected. Searching activates unconscious cognitive processes to recognize patterns and associations. Searching, therefore, enables selective filtering of information. Perception is limited to assumed relevant elements, while assumed unimportant or irrelevant details are ignored.

If you search for “online check-in” on an airline’s website, all aspects that are not related to online check-in tend to be overlooked. Focusing on something specific emphasizes all possible aspects that have to do with the focused mental content.

Perception vs. remembering.

In visual perception, memories of previously perceived things play an important role. On average, about 10 % of what we “see” is seen, and about 90% of what we think we see is added from memory. This means that around 10% of the information that can be seen in a user interface is perceived visually, and around 90% is added from memory. People, therefore, often see what they remember and not what is on the screen. This leads to “obvious” things being overlooked.

This is an often-underestimated challenge for the design of interactive systems, as every user makes these additions unconsciously and based on their individual wealth of experience, level of knowledge, cultural background, and current mood. The **user is, therefore, an unknown and highly variable factor.**

2.1.2 The structure of the human eye

The anatomy of the human eye has a significant influence on the perception of any object.

The lens focuses light rays onto the retina, which consists of light-sensitive cells. These photoreceptors convert light into nerve impulses. The optic nerve transmits these impulses to the brain, which interprets the visual image. Muscles control the shape of the lens for near and far vision.

There are a total of four types of photoreceptors in the retina. These are:

Three types of so-called cones. Cones are responsible for the perception of colors.

- One type of cones for each of the three primary colors (blue, green, and red).
- Color vision requires high light intensity and, therefore, works primarily in daylight.

And one type of so-called rods.

- These are very light-sensitive but can only distinguish between black and white.

The point of sharpest vision is located directly in the optical axis (i.e., "where you look at"). This point is called the "fovea centralis." The term "foveal vision" is therefore used to describe vision in a certain direction, with a fixed gaze, straight ahead, within a very small solid angle of 1° to 2°, in order to fixate a visual object, so that details, distances, the smallest differences in brightness and color can be recognized in a short time.

To recognize faces, they must be looked at directly in almost all cases. The attractiveness of portrait images is based on this effect. If they are on an interface, they are usually viewed directly.

Reading also requires looking directly at the word or sentence. However, it cannot be assumed that something has been read comprehensively, even if it has been looked at directly.

This effect of "looking at" is observed in eye tracking. The movements of the eye are measured and the point at which the point of sharpest vision is directed is shown. Eye tracking is a technology used to monitor and record the movement of a person's eyes, typically in real-time. It involves using specialized hardware and software to track an individual's gaze, measuring where and how long they look at various points of interest, such as images, text, or objects on a screen or within their environment. This allows conclusions of whether people could have perceived something or not.

Movement can be perceived very well in the periphery of the retina, but no details can be recognized in that area. Advertising with moving elements makes use of this effect. Moving elements at the edge of the screen draw attention to them. The gaze is most likely to be directed straight to it. As a result, the susceptibility to errors increases for tasks performed in the center of the screen.

2.1.3 The reading process

The anatomy of the eye has far-reaching implications for reading text. Text can only be **read** if it is looked at **directly.** During reading, the eye is briefly **fixed**, then moves in a **sudden movement** and is **fixed** again. Reading takes place during these brief fixations.

This has a particular impact on comparing values on the screen, for example. Values can only be compared well if they can be captured during a fixation, i.e., if they are very close to each other, as they are seen simultaneously at a very high resolution.

© UXQCC - FL 4.1 Syllabus, October 2025

2.1.4 The resolution of the human eye

Temporal resolution

When perceiving slightly different individual images, these are converted into a continuous sequence (=movement). To perceive flicker-free movement, one needs around 22 images per second (refresh rate) in the range of sharpest vision. At 70 frames per second, flicker is no longer perceptible in peripheral vision.

These effects must be considered when designing videos.

Spatial resolution

The spatial resolution of the eye is limited. This can be seen clearly in the different font sizes necessary at different distances so something can be read correctly.

The minimum distance between two pixels on the retina that can still be perceived as separate is around 4µm, corresponding to a visual angle of less than one degree. This applies to an eye with normal vision.

Determining the necessary font sizes for legibility from certain distances is complex and is, therefore, best done in practice using simple tools freely available on the internet. In practice, you should always assume that many people do not have normal vision and, therefore, choose a font size slightly larger than the minimum size required for people with normal vision.

Contrasts

Insufficient contrasts significantly reduce legibility. The human eye can only distinguish a limited number of contrasts. It is difficult to say whether a contrast between a font and the background is sufficient or not. Therefore, tools should also be used here to check the contrast between elements on the screen. Such tools are a build in feature of most software used for prototyping. Other tools are freely available on the internet.

Color vision defects further more reduce various contrasts and must be considered accordingly.

Different refraction of blue and red light

In low brightness conditions, such as in vehicles at night, it is not possible to focus on red and blue displays next to each other simultaneously at close range (e.g., at 70 cm), and therefore, this should be avoided. This is primarily due to the different refractions of the very different wavelengths of red and blue in the eye's lens.

2.1.5 Fields of vision

Central field of vision

The central field of vision in humans, which is determined by the fovea centralis in the retina, is comparatively small, but it provides the highest visual acuity and color perception. The exact size of this central field of vision can vary, but typically, it is about 1 to 2 degrees in diameter. This means that the area that someone fixates directly and sees with maximum sharpness is only a small part of the visual field.

Note: With the arm outstretched, this field corresponds approximately to the size of a thumbnail.

Main field of vision

The main field of vision, which refers to the area on which the eyes are focused, is much larger and is about 120 degrees horizontally and about 135 degrees vertically. Visual acuity decreases from the center to the periphery. Objects are mainly recognized in this area.

Peripheral field of vision

The human peripheral field of vision extends to around 180 degrees horizontally and around 135 degrees vertically and is adjacent to the main field of vision. In this area, movements are primarily perceived, but no objects are recognized.

Binocular field of view

The binocular field of vision in humans is the area where both eyes can see together. It is created by overlapping the fields of vision of both eyes and enables spatial vision, also known as stereo vision. The binocular field of vision normally covers an angle of about 114 degrees horizontally.

The binocular field of vision plays a crucial role in activities that require precise depth perception, such as reaching for objects and estimating distances. The binocular field of vision is particularly important in designing interfaces that need to be touched but cannot always be looked at directly, such as car controls.

2.1.6 Effect of colors

Colors are not only relevant for design and highlighting. They evoke associations and create an emotional and psychological effect. **Colors can reinforce messages but can also confuse a recipient**. These effects, as well as the perceived meaning of colors, are strongly dependent on the respective context.

Examples of the meaning of colors:

Red: Love, fire, energy, passion, blood, stop, danger, heat, energy.

Green: Sour, nausea, nature, hope, life, reassurance, okay, poison.

Blue: Dynamism, nobility, competence, coolness (serenity vs. detachment).

Violet: Extravagance, clergy, power, rigidity, decadence, sin, vanity.

Yellow: Sun, joie de vivre, warmth, mutability, envy, death.

Pink: Cute, sweet, tender, naive, gentle.

Orange: Modern, fun, young, pleasure, extroverted.

Brown: Warmth, decay, cozy, fascism, patina, lazy, aromatic, old-fashioned, withdrawn, cozy.

White: Pure, bright, perfect, sterile, neutral, bride, empty, innocence, illusionary, unrealistic.

Black: Death, night, elegance, mourning, neutral, heavy, threat, nothing, misfortune, seriousness, pessimistic, hopeless, obsessive.

Gray: Pale, fog, neutral, boring, theory, poor, secretive, unfriendly.

Cyan: Passive, concentrated, conscientious.

Turquoise: Wait-and-see, defensive.

Magenta: Idealistic, transcendental, theoretical.

However, intercultural differences in the effect of colors must be considered. In China, for example, white is considered the color of mourning or death.

Psychological color effects: Colors can also be interpreted emotionally. These effects are partly based on the use of colors as a system of order or security.

Colors can have a direct effect on physical reactions. However, these reactions are at least partially learned and not innate.

2.1.7 Visual Perception and WCAG

The **Web Content Accessibility Guidelines (WCAG)** are an international standard for accessible web content, developed by the W3C and legally binding in the EU (for public bodies at Level AA). They are based on four core principles: content must be **perceivable, operable, understandable, and robust**. For visual perception, the most relevant aspects include ensuring sufficient **contrast** between text and background, **scalable font sizes**, **alternative text descriptions** for images, avoiding information conveyed by **color alone** (e.g., red/green without additional markers), providing a clear **content structure**, and considering **color vision deficiencies**. These measures ensure that information remains accessible even for users with impaired vision

2.2 Color vision defects

K-L	K2 - 30 minutes
	LO 2.2.1 Describe color vision defects and understand their influence on UX design (K2)
	Color vision defects, red-green, blue weakness or blindness, normal vision, basic colors, primary colors

The sense of sight is comprised of **rods**, which can only distinguish brightness values, and **cones**, which are responsible for color perception. Humans can perceive three basic colors, for which they have three types of receptors (cone types) that require a higher light intensity than rods to function.

- **Three primary/basic** colors: Green, Red, and Blue

All visible colors are generated by mixing these basic colors. Not all people see identical colors with identical sensory stimuli (color vision defects). A defective type of receptor on the retina is responsible for color vision defects. Different types of color vision deficiency cause problems with seeing different colors.

Red-green color vision deficiency
The most common type of color vision deficiency makes it hard to distinguish between red and green. There are four types of red-green color vision deficiency:

- **Deuteranomaly** is the most common type of red-green color vision deficiency. It makes certain shades of green look more red. This type is mild and doesn't usually get in the way of normal activities.
- **Protanomaly** makes certain shades of red look more green and less bright. This type is mild and usually doesn't get in the way of normal activities.
- **Protanopia and deuteranopia** both make someone unable to tell the difference between red and green at all.

Blue-yellow color vision deficiency

This less-common type of color vision deficiency makes it hard to tell the difference between several different color combinations. There are two types of blue-yellow color vision deficiency:

- **Tritanomaly** makes it hard to tell the difference between blue and green and between yellow and red.
- **Tritanopia** makes someone unable to tell the difference between blue and green, purple and red, and yellow and pink. It also makes colors look less bright.

Complete color vision deficiency
If you have complete color vision deficiency, you cannot see colors at all. This is also called monochromacy or achromatopsia and is rare.

Of all color vision defects, approximately 0.005% affect the color blue. Therefore, a color scheme in which blue appears as a signal color is an obvious choice.

Color vision defects affecting red and green (red-green deficiency) occur in around 7% to 9% of all men and 0.3% to 0.8% of all women.

The percentages vary according to different authors. Roughly speaking, around 5% of people have problems distinguishing between the frequently used signal colors red and green.

To ensure that a design is also perceived correctly by people with color blindness, it is advisable to use tools to check this. These **tools can be used to simulate the perception of people with color blindness** so that countermeasures can be taken early in the design process.

Furthermore, familiar color schemes can be used that are perceived correctly by people with a red-green weakness, for example. An example of this would be to use blue and gray instead. As blue can be perceived by almost all people, the difference to gray is perceived by almost all people.

2.3 Environmental influences

K-L	K2 - 10 minutes
	LO 2.3.1 Describe which environmental influences affect usability (K1)
	Physical environmental influences, organizational environmental influences, social environmental influences

Environmental influences refer to various factors that affect the execution of human activities and can influence usability or user experience. Environmental influences can be categorized into different types:

- **Physical** environmental influences
- **Organizational** environmental influences
- **Social** environmental influences

Environmental influences can sometimes significantly reduce people's performance. It is therefore important to know the conditions under which an interface is used. A few examples are given below:

Cold: Limited motor skills, large hands (gloves).

Darkness: Loss of color vision, blindness.

Sunlight, brightness: Screens are difficult to read, weak contrasts cannot be recognized in glare.

Stress: Limited ability to think, reduced creativity.

Loud environment: Quiet noises are no longer perceived.

Tiredness, exhaustion: Limited ability to think, poor concentration, limited motor skills.

2.4 Gestalt laws (Gestalt Principles)

K-L	K2 - 40 minutes
	LO 2.4.1 Explain the laws of design and some examples of their impact on usability (K2)
	Gestalt psychology, Gestalt principles, Gestalt laws, law of proximity, law of similarity, law of simplicity, law of continuous line, law of unity, law of common destiny, law of common region, law of simultaneity, law of connected elements

Gestalt psychology, developed in the 1920s, explores human perception. The Gestalt laws described therein reveal regularities in the formation of wholes. In this case, "Gestalt" has nothing to do with "design." A set of mostly innate processes in the brain is used for visual stimuli. These processes are used to examine and classify an object. Nine types of features contribute to the differentiation between objects:

- Shape, color, brightness, size, direction, texture, arrangement, depth, movement.

The human brain searches for familiar patterns when perceiving and processing information. In doing so, the brain unconsciously draws on empirical values and provides us with clues as to what reality probably looks like. Many of these empirical values can be formulated using the Gestalt laws. The Gestalt laws are an important tool for representing affiliations and relationships between objects.

The six essential factors for the formation of contexts in perception were formulated by Wertheimer as early as 1923. Since then, these Gestalt factors have often been referred to as Gestalt laws.

Law of proximity: Elements that are close together are perceived as belonging together.

Law of similarity: Elements that are similar in shape, size, color, or texture are perceived as belonging together.

Law of simplicity (or good form or conciseness): Preference is given to figures that result in a simple structure (= "good shape").

Law of Continuity: Elements that are arranged in a continuous line or curve are perceived as more related than those that are not. It also means that if two lines cross, we do not assume that the course of the lines makes a bend at this point; rather, we assume two rather straight continuous lines.

Law of Closure: Humans tend to perceive incomplete or fragmented figures as complete by mentally filling in missing parts. For example, the edges of a cube result in the perception of a (non-existent) cube.

Law of common fate: Elements moving in the same direction at the same time are perceived as belonging together.

Stephen Palmer formulated three further design laws in the 1990s. These three laws or principles are very often used in UX design.

Law of the common region: Elements in demarcated areas are perceived as belonging together, e.g., they can be realized by a colored background, a frame, or similar.

Law of simultaneity: Elements that move or even change in one direction at the same time are preferably perceived as belonging together.

Law of connected elements: Connected elements are perceived as one object. This can be realized by lines or arrows, for example.

Some principles are stronger than others. For example, the principle of connected elements can override the principle of similarity. However, the relative strength of the principles is not 100% clearly defined. In a specific application, it is often worth simply trying things out to find the best method. This is especially true as graphic specifications often must be taken into account.

2.5 Mental models

K-L	K2 - 40 minutes
	LO 2.5.1 Explain mental models and their significance for usability and UX design (K2) LO 2.5.2 The role of mental models in the interpretation of concepts (K2) LO 2.5.3 Reuse known mental models (K2)
	Mental model, radio button, checkbox, Windows button.

2.5.1 Mental models basics

A mental model is a cognitive representation or framework an individual uses to understand, explain, and predict how something works or how a system functions. It is a mental construct that helps individuals organize and interpret information, make decisions, and navigate their environment. Mental models are formed through experiences, learning, and interactions with the world around us, and they influence how we perceive, interpret, and interact with new information and situations. They can vary widely from person to person based on individual knowledge, beliefs, and experiences. Understanding users' mental models in design and user experience contexts is essential for creating intuitive and user-friendly products and interfaces.

Mental models are conscious but also unconscious **assumptions** about how a user interface works. These assumptions of the respective users are **based on experiences** that users have had with similar systems. For this reason, it is often advantageous to adopt such familiar concepts in newly developed software.

If familiar concepts are no longer used but completely redesigned, many users react with rejection, as they expect something different from the interface.

The content of a screen is always evaluated by means of mental models.

So-called **"mental model diagrams"** represent the motivations, thought processes, and underlying behavioral motives of users. The main aim is to illustrate goals and the approach people take to achieve these goals in relation to the user interface.

People generally find it **more difficult to recall something from memory than to recognize something.** Any perception always evokes mental models, which then make the corresponding content easier to recall.

Mental models also play an important role in **understanding words**. Different groups of people often assume different information behind certain terms. It is, therefore, important to **match** the **terms used exactly to the user group.**

Mental models also play a major role in reading. **People** usually only **read** a few letters of a description and **complete the rest with the help of their mental models**. This often leads to misunderstandings. They then try it out to see if it "works." If the interface does not behave as expected, a negative attitude arises.

2.5.2 Examples of mental models

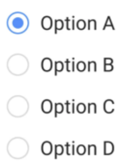
Simple mental models: Example of the differences between checkboxes and radio buttons

The mental models of radio buttons and checkboxes are different. Radio buttons and checkboxes are both graphical elements used in user interfaces to allow users to select options. However, they have different purposes and create different expectations for users.

Radio button

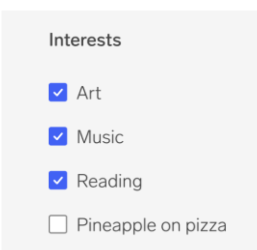
A radio button only allows the selection of one option from a list. If a user selects an option, any previous selection is canceled.

Radio buttons are often used when exclusive or alternative options are required. The user can only select one of the available options.



Checkbox

In contrast to radio buttons, checkboxes allow you to select several options simultaneously. Each checkbox is independent of the others. Checkboxes are used when independent options need to be selected. The user can activate as many checkboxes as required. The status (activated or deactivated) of each checkbox can therefore be controlled individually.



The user's mental model is shaped by previous use and experience with these graphic elements. Users expect a certain behavior of the round or square field. When seeing a radio button, it is expected that only one option can be selected. Users assume that the selection among the radio buttons is exclusive.

Checkboxes are different. Users expect to be able to select several options at the same time. They assume that each checkbox is independent and that selecting one checkbox does not affect the selection of another.

The mental model is the behavior of the element that users expect.

In practice, radio buttons and checkboxes are therefore used depending on the context and the specific requirements of the user interface to ensure that users have the expected interaction options.

Complex mental models

Mental models are often behind more complex tasks. One example of this is the question, “How do I switch off a PC running Microsoft Windows?” Most users have learned that there is an area with a Windows icon at the bottom left of the display. The same icon on the keyboard (Windows button) leads to the assumption of similar behavior when the button is pressed.

3 Standards and guidelines

3.1 The importance of standards and norms

K-L	K2 - 10 minutes
	LO 3.1.1 Being able to explain the meaning of standards (K1)
	ISO standards, usability standard

National standardization institutes develop norms and standards based on country-specific agreements and are represented in the corresponding **international institutions.**

The purpose of standards is the national and international coordination of products and the **promotion of rationalization**, **quality assurance, and occupational safety**. Standards **standardize test methods** and **facilitate communication in business and technology**. Standardization and the resulting **compatibility with each other** can lead to competition and the associated pressure to innovate and reduce prices. They are the basis for **legal certainty** and play a role in warranty, liability, and compensation claims. However, they also restrict markets by excluding products that do not comply with the standards.

Standards can be divided into the following areas:

- Safety standards
- **Usability standards**
- Quality standards
- Dimensional standards
- Test standards.

ISO standards are developed by the international standards organization ISO and are usually adopted at the European or national level.

3.2 ISO 9241

K-L	K2 - 30 minutes

	LO 3.2.1 Be able to give an overview of the contents of the ISO 9241 standard (K1) LO 3.2.2 Explain the lifecycle of ISO 9241-210, including the detailed components (activities) (K2) LO 3.2.3 Explain the design principles (dialog principles of ISO 9241 - 110 and be able to give examples (K2) LO 3.2.4 Being able to name further UX / usability standards (domain-specific or method-oriented) (K2)
	Normative framework**,** ISO 9241, user requirements, context of use, design solution, evaluation, principles of dialog design, task appropriateness, self-descriptiveness, learning facilitation, controllability, conformity to expectations, customizability, and error tolerance. Form dialogs, ISO/TR 16982:2002, IEC 62366:2015, ISO 14915:2002, ISO 11064:2000

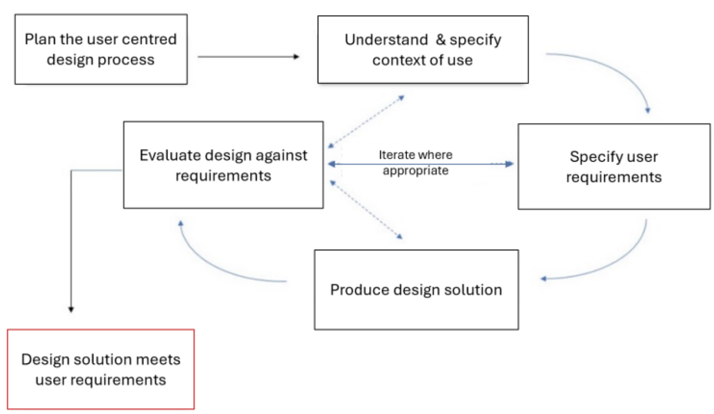
The central element of the **normative framework for the user interfaces of interactive systems** is the ergonomics of human-system interaction according to EN ISO 9241 (according to the national designations, this is DIN EN ISO 9241 in Germany and ÖNORM EN ISO 9241 in Austria. For other European countries, it may be necessary to determine whether EN ISO 9241 has been adopted in corresponding national standards).

ISO 9241 defines usability as follows: Usability refers to the extent to which a product, system, or service can be used by **specific users** in a **specific context to** achieve **specific goals effectively, efficiently, and satisfactorily.**

The understanding of the terms “usability,” “defined task,” “effective,” and “satisfaction” was deepened and broadened in the latest version to better meet the requirements of the various systems.

3.2.1 ISO 9241-210

ISO 9241-210 is the standard for **a user-centered approach in development projects**. It replaces the old ISO 13407 standard and is primarily aimed at project managers who want to **systematically plan and anchor** user-centered design **activities** in the development project. It offers support for the entire iterative lifecycle of a system, from planning and user-oriented implementation to operation and maintenance. Methods and procedural guidelines are provided for this purpose.



ISO 9241 – 210 Lifecycle

User-centered design **does not** describe **a plethora of disjointed individual methods** but is typically applied in a **superordinate “lifecycle”** (see ISO 9241-210). The activities of this lifecycle begin **even before the actual development**

of the human-machine interface. This results in the following phases or activities, which **should be iterated through until the product meets the user requirements**:

Understanding and specifying the context of use:

Identify and understand the characteristics of the users, tasks, and the environment in which the interactive system is to be used.

Specify the user requirements:

Define and document the requirements clearly based on an understanding of the users and their context. This includes determining the users' needs and expectations of the system.

Produce design solutions:

Generate design solutions that meet the specified user requirements. This includes creating conceptual designs, prototypes, and detailed interface designs.

Evaluation of the designs against requirements:

Evaluate the design solutions based on user requirements. This can be done through various evaluation methods, including usability tests and expert evaluations.

3.2.2 ISO 9241-110: Interaction principles (former dialogue principles)

This part of ISO 9241 describes the principles of interaction design, i.e., generally applicable quality features or design principles for interactive systems. These are also used to check conformity with the standard. They contain an overarching definition and, in addition, an operationalizable requirement. (These then specifically help the developer or user interface designer to comply with these interaction principles).

- Suitability for the user's task
- Self-descriptiveness
- Learnability
- Controllability
- Conformity with expectations
- Error robustness
- User engagement

The lifecycle described under point 5, "User-Centered Design Lifecycle," is based on this ISO 9241-210 model.

3.2.3 Further usability / UX standards

The following standards also contain essential usability content. They show the importance of usability and a human-centered development process.

General

ISO/TR 16982:2002	Ergonomics of human-system interaction - methods for ensuring usability that support user-oriented design
Domain-specific	
ISO 14915:2002	Software ergonomics for multimedia user interfaces
ISO 11064:2000	Ergonomic design of control centers
IEC 62366:2015 (German updates IEC 62366-1:2015 + COR1:2016 + A1:2020)	Medical devices - Part 1: Application of fitness for purpose to medical devices

3.3 W3C Web Content Accessibility Guidelines

K-L	K2 - 30 minutes
	LO 3.3.1 Be able to give an overview of the Web Content Accessibility Guidelines (WCAG) (K2)
	Accessible; basic principles (perceptible, understandable, robust, operable) Guidelines, success criteria, priority, A/AA/AAA

The Web Content Accessibility Guidelines (WCAG) are an international standard for the accessible design of websites. The Web Accessibility Initiative (WAI) of the World Wide Web Consortium (W3C) developed the WCAG. The International Organization for Standardization (ISO) has declared WCAG 2.2 to be the ISO/IEC 40500:2025 standard.

It is handled differently by local laws and regulations. In the European Union, for example, it has already been declared binding for public bodies with WCAG Level AA, and corresponding requirements for other Internet offerings are continuously emerging. It is the responsibility of the development management to know the national legal requirements for accessibility and to demand them in a timely manner in the developments and to ensure their implementation.

One of W3C's main objectives is to make the Internet, with all its benefits and possibilities, **accessible to all people**, regardless of their hardware, software, network infrastructure, native language, culture, geographical location, and physical or mental abilities. To make the web, its content, and services "**accessible,**" the W3C working group has developed corresponding guidelines.

These Web Content Accessibility Guidelines (WCAG) cover a wide range of **guidelines to** make **web content more accessible.** By following these guidelines, content will be accessible to a wider group of people with disabilities. This includes:

- Blindness and visual impairment
- Deafness and deteriorating hearing
- Learning disabilities, cognitive impairments
- Restricted mobility

- Speech impediments
- Photosensitivity and combinations of these disabilities

In addition, following these guidelines will, in many cases, make web content more usable for all users.

The WCAG success criteria were formulated as **testable** statements that are **not technology-specific.** Instructions on how to fulfill the success criteria for specific technologies as well as general information on how to interpret the success criteria can be found in separate documents.

The structure of WCAG comprises four basic principles and a total of 13 guidelines. These are not testable, but they form a framework and overarching objectives for comprehension purposes. The guidelines, in turn, are assigned a total of 70 measurable success criteria.

Principle: Perceivable - perceptible

The principle of perceptibility is intended to ensure that functions and information are presented in such a way that every user can notice them. The so-called two-channel principle is important here. This means that information can be perceived via two different sensory channels.

Related directives:

- Offer text alternatives for non-text content.
- Offer subtitles and other alternatives for multimedia.
- Create content that can be presented in different ways, including through assistive technologies, without losing meaning.
- Make it easier for users to see and hear content.

Principle: Understandable - comprehensible

The principle of comprehensibility should not be underestimated, even outside the context of IT accessibility. Content should be easy to read and understand for the widest possible audience, even if it is read out loud.

The principle of comprehensibility also includes the predictability of the user interface with a consistent display and navigation and support in avoiding input errors.

Related directives:

- Make text readable and understandable.
- Make content appear and function in a predictable way.
- Help users avoid and correct errors.

Principle: Robust - robust

Robustness means a high level of compatibility of the content provided with the user agents used (in particular, the web browser) and assistive technologies (in particular, a screen reader). This means that standards (correct syntax, uniform use of HTML, etc.) must be adhered to when providing the content.

Related directives:

- Maximum compatibility with current and future user tools.

Principle: Operable - operable

To enable users to interact with IT solutions, they must also be operable by people with disabilities.

Related directives:

- Make all functions accessible via a keypad.
- Give users enough time to read and use the content.
- Do not use contents that trigger seizures or physical reactions.
- Help users navigate and find content.
- Facilitate the use of inputs other than the keyboard.

The 78 success criteria are divided into three categories. They are directly realizable and measurable and not technically specific):

- with high priority (A)
- with normal priority (AA)
- with low priority (AAA)

There are conformity levels (A, AA, AAA) for assessing the conformity of a website. The degree of compliance is to be classified with regard to several aspects:

- Whole page or just parts of it?
- Complete process (e.g. order)?
- Are barrier-free technologies used?
- Are techniques used that explicitly exclude certain people?

In Europe, the regulatory landscape increasingly shapes the design of digital products. The **European Accessibility Act (EAA)**, which becomes legally binding across the EU on **June 28, 2025**, requires companies to ensure the accessibility of digital services and products in accordance with **EN 301 549**. Depending on the product category, specific **training and documentation obligations** apply. These regulations emphasize the importance of **User-Centered Design**, ensuring that digital solutions are inclusive, transparent, and usable for all individuals.

© UXQCC - FL 4.1 Syllabus, October 2025

4 User-Centered Design

4.1 Basics of user-centered design

K-L	K2 - 15 minutes
	LO 4.1.1 Explain the basic principles of UCD - User-Centered Design. (K1)
	Basic principles of UCD, user experience design, multidisciplinary skills and perspectives

User-centered design is an approach to developing interactive systems that focus on the user and their requirements, knowledge, skills, and needs. The user-centered design of interactive systems offers numerous advantages. The overall costs of a product life cycle, including its conception, design, implementation, maintenance, use, and servicing, can also be significantly reduced.

The basic principles of UCD are:

- The design is based on a **solid understanding of the users, their tasks, and the application context.**
- **Users** are **involved during the** design and development process.
- The design is **guided and improved** by **user-centered evaluations**.
- The process is **iterative**, i.e., design improvement measures based on usability evaluations and feedback are an integral part of the process.
- The design is based on the **overall user experience***.
- The design team combines **multidisciplinary skills and perspectives,** i.e., in addition to technical and domain knowledge, depending on the system, human, social science, and psychological skills are also represented and actively integrated into the development team.
- **Duration of the UCD of a system or product**: However, user experience design does not end with a product's delivery or online launch on the market. Rather, it is an ongoing process that also deals with continuous optimization and identifying the right time for a relaunch. Supporting users and communicating with them in the day-to-day application or use of a system is a key factor in the user experience.

*** User experience design**: **User experience design** encompasses far-reaching aspects of the specific user experience; it includes all experiences that are in any way related to a product to be developed. These experiences must be methodically addressed and optimized. Possibilities of the inventory of methods of empirical social research are being introduced here and require the involvement of appropriately trained people in multidisciplinary teams from what are usually predominantly technical development teams. The **usability** of **a system** is also **systematically and actively designed** and not left to chance or the mere skill of the user interface designer(s).

4.2 Planning user-centered design

K-L	K2 - 15 minutes
	LO 4.2.1 Explain the basic requirements for the planning of UCD (K2)

Responsibilities, content of UCD planning, integration of usability / UX into the project plan, time and resources

User-centered design aspects should be integrated into all phases of development and must, therefore, be planned accordingly (according to ISO 9241-210:2019 chapter 6.2).

Responsibilities

When planning a UCD, it is crucial to consider how usability affects the purpose and use of the system or service. Factors such as health & safety, environment & sustainability, or special requirements for systems for experts or special work contexts must be taken into account.

Contents of the planning

The planning must include the appropriate tools and methods as well as the corresponding resources and required skills of the people. The definition of possibilities or concrete procedures for integrating these activities and their results with other system development activities must also be carried out. This includes documentation requirements, milestones, deliverables as well as feedback loops and various time aspects.

Integration into the project plan

The UCD plan must be part of the overall project plan. To ensure that the UCD plan is followed up and implemented effectively, it should be subject to the same requirements (e.g., responsibilities, change control) as other key activities. The UCD aspects of the project plan should be reviewed and revised if requirements change during the project's life.

Time and resources

When planning the project, time and resources should be set aside for activities that focus on people. This includes time for iterations, incorporating user feedback, and evaluating whether the design solution meets user requirements.

Additional time should also be planned for communication within the design team or with other stakeholders to clarify potential conflicts and compromises.

The challenges listed below are considered in these activities. (According to ISO 9241-210:2019 chapter 7.1.):

- There are often several different user groups and other interest groups whose needs must be taken into account.
- The context of use can be very diverse and differ from user group to user group and between different tasks. At the start of a project, the requirements that can be captured are unlikely to be exhaustive.
- Some requirements only appear when a proposed solution is available.
- The requirements of users can be diverse and possibly contradict each other and those of other interest groups.
- Initial design solutions rarely satisfy all user requirements. It is difficult to ensure that all parts of the system are considered in an integrated way.

4.3 The quality of data and results

K-L K2 - 20 minutes

	4.3.1 Explain the importance of quality criteria for collected data in the context of usability and user experience design methods (K1) LO 4.3.2 Being able to explain the concept and meaning of objectivity (K2) LO 4.3.3 Explain the concept and meaning of reliability (K2) LO 4.3.4 Explain the concept and meaning of the different validities (content validity, construct validity, and apparent validity) (K2)
	Objectivity, reliability, validity, content validity, construct validity, apparent validity, selection of interviewees, test subjects; interviewer and interview effects, cognitive and social influencing factors, questionnaire development, task validity

As part of a user-centered design process, various methods are used to collect data about the users, their requirements, or their interaction with a system. It is essential to **assess the quality of the data,** as **incorrectly collected or interpreted data** can have a **lasting negative impact on the development of interactive systems** or drive development in the wrong direction. This also includes a differentiation from the questions and methods of market research.

The most important quality criteria (derived from empirical social research) are as follows:

Objectivity: The independence of the research results from the persons conducting, evaluating, and interpreting the research.

Reliability: The formal accuracy or reliability of measurements/investigations.

Validity: Describes the extent to which the measurement/investigation actually measures/investigates exactly what it is supposed to measure/investigate:

- **Content validity:** The extent to which a research method fully captures the content of a *characteristic of interest.*
- **Construct validity:** The extent to which a research method measures a construct (e.g., “good tennis player”) as the construct is actually determined, i.e., the characteristics/skills that determine the construct (e.g., good ball eye, strong runner, good serve ...).
- **Face validity:** The extent to which the person being tested (or the test subject) can recognize what is being tested/measured. A high level of face validity promotes the acceptance of procedures, but it can also promote the unintended possibility of deliberately influencing the results.

The most important factors influencing the relevant data should be made aware of and understood:

- The importance of basic knowledge for the development, implementation, and evaluation of surveys (questionnaires, interviews).
- The importance of basic statistical knowledge for quantitative and qualitative evaluations and analyses.
- The importance of the selection and number of interviewees and subjects.
- The risk of experimenter and interviewer effects.
- The importance of cognitive and social factors on the behavior of test subjects.
- The importance of task validity in tests.

4.4 Verification versus validation

K-L	K2 - 10 minutes

	LO 4.4.1 Explain the concept and meaning of verification (K2) LO 4.4.2 Explain the concept and meaning of validation (K2)
	Verification, validation

Verification and validation are two different processes in the context of product development and quality assurance. Below, you will find an overview of the differences between the two processes:

Verification:

Focus: Verification is about checking whether the specified functions have been implemented in a system or component.

Schedule: It is normally carried out during the development phase.

Methods: Techniques such as inspections, reviews, walkthroughs, and tests are used to check whether the design and implementation comply with the predefined specifications.

Aim: The aim of verification is to ensure that the product is developed correctly and meets the specified requirements. It confirms that each phase of the development process adheres to the planned activities.

Validation:

Focus: Validation is about evaluating a system or component during or at the end of the development process to determine whether it meets the specified (user) requirements.

Schedule: It is normally carried out after the development phase.

Methods: Techniques such as testing (including user acceptance testing) and demonstration are used to validate that the product fulfills its intended use in its actual environment.

Aim: The aim of validation is to ensure that the end product meets the user's requirements and expectations. It confirms that the end product is suitable for the intended purpose and is validated using real scenarios.

In summary, verification is about checking that each stage of the development process meets the specified requirements, while validation is about assessing whether the final product meets the actual needs of the users and functions correctly in the intended environment. Both processes are critical to ensuring a product or system's overall quality and success.

5 Analysis of the user requirements

5.1 The 4 components of the requirements analysis

K-L	K2 - 60 minutes
	LO 5.1.1 Being able to name the four components of the requirements analysis (K2) LO 5.1.2 Explain the importance and application of the analysis of users and other stakeholders (K2) LO 5.1.3 Explain the meaning and application of the analysis of user goals and tasks (K2) LO 5.1.4 Explain the meaning and application of the analysis of the application context (environment, situation) (K2) LO 5.1.5 Explain the significance and application of the analysis of comparative systems (K2)

User analysis; tasks analysis; context analysis; comparative analysis; user groups; personas, context; scenarios, user stories and use cases, UX research

5.1.1 Basics of analyzing user requirements

To design a system optimally for the actual users, it is necessary to have **all relevant information available** for the implementation or design of the system that may be **relevant for use. The data is collected** in the corresponding analysis or elicitation process, **from which the relevant information** can **then be derived.** It is important that the "derivation" of the information must not be a subjective interpretation by individual designers or developers.

The frequently used term UX research also includes the elicitation of user requirements but often starts one step earlier in the product development phase. UX research includes the discovery and problem definition phase and brainstorming.

The four relevant components of an analysis of usage requirements are: Users, tasks/goals, context of use, and comparison systems.

5.1.2 Analysis of users and other stakeholders

During user analysis, future users of the system to be developed are surveyed and analyzed. All of their **characteristics,** such as eyesight, height, specialized knowledge, and technology affinity, are analyzed and documented as they may influence usage.

User groups can be formed with this knowledge and the knowledge of the content-related tasks that are to be solved with the system (see task analysis). A distinction is made between **primary and secondary user groups**. A **primary user group** comprises those users who **will mainly use** the system, i.e., whose **requirements of the system** are seen **as having priority.** Although the requirements of **secondary user groups** must be implemented as well as possible (user-oriented and functional), they take second place to those of the primary user group. Secondary user groups are usually more responsible for **tasks such as maintenance or training.**

The most important objectives and restrictions are described for these user groups:

Personas

Ideally, several distinct personas are developed for each of the most important user groups.

Personas are **fictitious people** who are intended to represent most future actual users. Each persona is brought **to life** through **photos, names, and data such as age, gender, education, preferences, hobbies**, **character traits, and life backgrounds**. During the development process, the team of designers and developers repeatedly take up the **needs of these fictitious people** and use them to play out different user scenarios, for example. Personas, therefore, not only help fulfill pure software ergonomic requirements in the design process but also to consider the desired user experience for the target group. By creating such personas, it is possible to avoid assuming a **non-existent standard/average user;** instead, specific user requirements must also be met.

5.1.3 Analysis of user goals and tasks

In most cases, users have **specific tasks in mind** when they use a system (searching for specific content, buying something, communicating, etc.). Task analysis is about **identifying these specific tasks in** order to map them optimally

© UXQCC - FL 4.1 Syllabus, October 2025

in the system. The characteristics of tasks that can influence usability and accessibility must be described, e.g., the way in which users typically perform tasks, the frequency and duration of execution, mutual dependencies, and activities to be carried out in parallel. For tasks, it is important to include more than just obvious, visible steps. A distinction is therefore made between the following:

Action tasks: Focuses on the required actions that the user must perform (e.g., manual activities, movement, or object manipulation).

Cognitive tasks: Focuses on the mental processes that the user goes through when working on a task. These include important cognitive aspects of decision-making, problem-solving, attention, and memory.

To support design decisions, tasks can often be divided into three categories:

Primary tasks: Are the most frequent, most important tasks of the prioritized user group(s)

Secondary tasks: Must also be able to be carried out in the system but are rarely or only carried out by a secondary user group.

Critical tasks: Their incorrect execution can lead to critical events or situations; therefore, their correct execution has a high priority

5.1.4 Analysis of the application context (environment, situation)

The usability of a system or the user experience **is largely dependent on** the **context** in which **it is used**. Only if you know the different application contexts can you optimize the system accordingly. In principle, the technical environment, including hardware, software, and materials, must be determined first.

Other key contextual factors include the external **physical context** (light, temperature, spatial arrangement, etc.) and the personal physical context (sitting position, movement, freedom of movement, etc.). Psychological context factors such as stress, privacy, motivation, etc. must also be considered. The social and cultural aspects of the environment include for example, work practices, organizational structure, and attitudes.

In contrast to conventional software applications, the usage context of web applications is characterized by special features. For example, conventional software applications are usually based on defined user groups, task and organizational contexts, whereas public websites are often aimed at a broad range of users with sometimes very different interests and information needs. This makes it all the more important to know the basic design decisions and strategies when developing WWW user interfaces and to take them into account during the development process.

5.1.5 Analysis of comparison systems

Nowadays, users use numerous systems, and from their use, they bring experience to the handling of another system. This can be advantageous or disadvantageous. Analyzing comparative systems helps to understand what users expect from a new system, down to small interaction details. It is, therefore, crucial to be aware of **potentially influential systems in order to turn** their **effect into a positive one**. Correspondingly influential systems can be systems from a **similar specialist area** (e.g., accounting programs) as well as systems that use **comparable concepts** (e.g., product search of an online store), or even **directly embedded "modules"** (e.g., interactive city map).

© UXQCC - FL 4.1 Syllabus, October 2025

5.2 Specifying user requirements

K-L	K2 - 15 minutes
	LO 5.2.1 Explain the requirements for the quality of user requirements specifications (K2)
	User needs, context of use, user requirements, quality of user requirement specifications

In almost all development projects, determining user needs and defining the functional and other requirements for the product or system is one of the main activities. In user-centered design, this activity is expanded to include an explicit statement of the user requirements in relation to the intended context of use and the business objectives of the system as follows (according to ISO 9241-210:2019 Chapter 7.3.).

Identifying the needs of users and other stakeholders:

The needs of users and other stakeholders should be identified considering the context of use. This includes what users need to achieve (rather than how they can achieve it) and the constraints of the context of use.

The derivation of user requirements must include the following:

- The intended context of use;
- Requirements derived from user needs and the context of use - for example, a product may be required for outdoor use;
- Requirements resulting from the relevant knowledge of ergonomics and user interfaces as well as from standards and guidelines;
- Usability requirements and goals, including measurable criteria for usability and satisfaction in specific usage contexts;
- Requirements that are derived from organizational requirements and directly affect the user.

User requirements form the basis for the design and evaluation of interactive systems to meet user needs. User requirements are developed in conjunction with the general requirements specification of an interactive system and are part of this specification.

Possible conflicts between user requirements, e.g., between accuracy and speed, must be resolved, and the justifications, factors, and weighting of aspects used in trade-offs should be documented so that they can be understood in the future.

The quality of the user requirements specifications must ensure that:

- Later testing is possible.
- They can be reviewed by the relevant interest groups.
- They must be consistent in themselves.
- They can be updated as required during the service life of the system.

5.3 Definition of usability and user experience goals

K-L	K2 - 15 minutes

	LO 5.3.1 Explain the meaning of and differences between qualitative and quantitative usability goals (K2) LO 5.3.2 Explain the meaning of and differences between relative and absolute usability goals (K2)
	Qualitative usability goals, quantitative usability goals, absolute usability goals, relative usability goals

Qualitative and quantitative usability goals serve as a **guideline** for the design of interactive user interfaces and form **acceptance criteria** for evaluation during the design process. They **facilitate the decision** to either go through a further design cycle or to move on to interface development.

The first step is to create a common and accurate picture of the user groups (derived from the user profiles) and a corresponding and accurate model of the work and the working environment (from the task analysis) to better focus the design process.

Qualitative usability goals: Qualitative goals are helpful in guiding the interface design, especially in the **initial phase.** They result from the **requirements** from the user profiles and the context-related task analysis.

Examples:

- The system should not require any knowledge of the underlying technology.
- During the transition to new releases, changes that are irrelevant to users' tasks should not be visible.
- The system is designed to support group work.

Quantitative usability goals: The achievement of qualitative goals is often difficult to specify. In contrast, **additionally defined quantitative goals** are **more objective and can be measured more precisely**.

Examples:

- Defining a specific or maximum permissible execution time.
- The execution times are set for a specific level of user experience:
  - Expert: Simple use of the application
  - New user: Easy to learn the application

Absolute goals use absolute quantitative variables such as processing time (in minutes, seconds), number of errors, etc.

Relative goals refer to the user experience with a specific product/interface relative to the experience with another product/interface.

- Clear preference between alternatives.
- Level of satisfaction with a particular interface (5-point scale: dissatisfied to fully satisfied).
- Performance targets quantify the current performance of a user in the execution of a specific task. Usually: time to perform the task or to learn how to perform the task, number and type of errors.

5.4 Scenarios, user stories, and use cases

K-L	K2 - 20 minutes
	LO 5.4.1 Explain the meaning and application of user scenarios. LO 5.4.2 Explain the meaning and application of user stories. LO 5.4.3 Explain the meaning and application of use cases.
	User scenarios, user story, use cases (application cases)

User scenarios

User scenarios show **how users accomplish tasks in a specific context**. They provide **examples of the different uses of devices and applications** and form a basis for subsequent usability tests. A user's tasks, goals, and motivations must be defined for such scenarios.

User scenarios can have **different levels of detail.** Goal or task-driven user scenarios only define what a user wants to achieve. Comprehensive scenarios look at the background of the user and the task. They provide a deeper **understanding of the user's motivation and behavior** to solve the task.

In principle, user scenarios should cover a wide variety of situations. It is important to **consider not only obvious cases** but also those that are of interest to the design and development team. **Situations that challenge the concept of the system as such** should also be **considered**.

User stories

A user story is a concise, informal description of a feature or functionality of a software system from the perspective of an end user. It is a common tool in agile development methods, especially in Scrum, to capture and communicate requirements in a user-oriented way. A user story usually follows a simple template:

> As a [*type of user*], I want [*an action or function*] to [*benefit or goal*].
>
> For example:
>
> As a website visitor, I want to be able to search for products by category so that I can quickly find the items that interest me.

The most important components of a user story include:

- **Role** (as one): Describes the type of user or stakeholder who has a specific need or goal.
- **Action** (I want): Describes the desired action or function that the user is requesting.
- **Benefit** (so that): Explains the reason or benefit behind the user's request.

User stories are deliberately kept short and are intended to serve as a starting point for discussions between developers, product owners, and other stakeholders. They offer a user-centered perspective that enables teams to understand the requirements and motivations of users. During sprint planning or backlog grooming sessions in agile development, user stories are often elaborated, estimated, and prioritized based on their importance and complexity.

Use cases (application cases)

In contrast, use cases describe **usage from the perspective of the application.** They make it possible to **address specific processes**. These describe the steps that a user performs for the specific task of an application and how the **application reacts to the user's actions**. Use cases are used to describe the interaction sequences and evaluate them in terms of their priority. However, as with user scenarios, it is also important for use cases to have the most accurate data possible about the user.

© UXQCC - FL 4.1 Syllabus, October 2025

6 Developing design solutions

6.1 Basic design requirements

K-L	K2 - 15 minutes
	LO 6.1.1 Explain the principles of developing design solutions (K1)
	Design principles, design solution

Design solutions are developed according to the framework conditions and possibilities of the developing organization. Nevertheless, there are some principles that should be an integral part of the development of appropriate solutions.

Principles of the development of design solutions:

- Design solutions are developed based on the findings of the requirements analysis and the defined usage requirements.
- Design solutions are concretized and visualized/illustrated, e.g., with the help of scenarios, simulations, prototypes, or mockups.
- Basic design principles from ISO 9241 - 110 are taken into account.
- Design solutions are iteratively improved and adapted based on the findings and feedback from user-centered evaluations.
- Design solutions are communicated in a timely and clear manner to the people responsible for their implementation.

6.2 Prototypes and mockups

K-L	K2 - 45 minutes
	LO 6.2.1 Being able to explain the purpose, use, and application of prototypes and mock-ups (K2) LO 6.2.2. be able to explain prototype classification according to the type of implementation (K2) LO 6.2.3. be able to explain prototype classification according to content (K2) LO 6.2.4 Being able to explain exploratory prototyping or usability prototyping (K2)
	Low fidelity prototype, high fidelity prototype, scenario prototype, horizontal prototype, vertical prototype, explorative prototyping, mockup

Prototypes help make design and processes understandable and illustrate a preliminary stage of the later application. They are used at a **very early stage of the development process.** This allows **potential risks or problems to be identified** and eliminated **in advance.** Prototypes **support discussions** and avoid misunderstandings in the development process.

Prototypes often only represent the part of the functional scope to be tested and, therefore, allow different concepts to be tried out. If a prototype is used to explore usage requirements that are not yet understood, this process is called exploratory prototyping.

6.2.1 Prototype classification according to the type of implementation

Depending on the intended use, the creation of prototypes is used in different forms and variants. A basic distinction is made between **low-fidelity** prototypes (low similarity to the end product, testing the benefits of the idea) and **high-fidelity prototypes** (high similarity, testing details, and precise functions). Mixed forms - for example, interactive simulations using HTML or PowerPoint - are also classified as medium (lo-hi) fidelity prototypes. The classification according to fidelity shows the range of possibilities; in day-to-day practice, it is only relevant insofar as the type of implementation has an influence on the possible applications and user reactions.

High-fidelity and low-fidelity prototypes are two different types of prototypes used in the design and development process to visualize and test product concepts. These prototypes differ in their level of detail, realism, and the phase of the design process they are typically associated with.

To summarize, the main differences lie in the level of detail, realism, and purpose of the prototype. High-fidelity prototypes are more sophisticated, resemble the final product, and are used for advanced testing and communication with stakeholders. Low-fidelity prototypes are simpler and used for quick exploration and iteration in the early stages of design. Both types of prototypes play an important role in the iterative design process as they cover different needs at different stages.

High-fidelity prototypes are predominantly used today.

High Fidelity prototype:

Attention to detail and realism: High-fidelity prototypes are detailed and realistic representations of the end product. They precisely mimic the look, feel, and functionality of the planned end product.

Visuals and interactivity: High-fidelity prototypes often contain sophisticated graphics, colors, and realistic interactions. You can use actual content that closely resembles the final product.

Use cases: High-fidelity prototypes are usually created in the later stages of the design process when the design is more fleshed out. They are suitable for user testing, customer presentations, and stakeholder reviews to provide a realistic experience.

Low-fidelity prototype:

Detail and realism: Low-fidelity prototypes are simple and abstract representations of the product that emphasize functionality over aesthetics. They lack the detailed visual and interactive elements found in high-fidelity prototypes.

Sketches and wireframes: Low-fidelity prototypes can take the form of hand-drawn sketches, paper prototypes, or digital wireframes. They focus on conveying the basic layout and structure of the design.

Use cases: Low-fidelity prototypes are created in the early stages of the design process to quickly explore and communicate design ideas. They are useful for gathering feedback, validating concepts, and iterating on the design before investing time in detailed visual elements.

6.2.2 Prototype classification according to content orientation

In summary, it can be said that the main difference lies in the depth of implementation and the focus. Vertical prototypes focus on the detailed representation of specific functions, while horizontal prototypes are intended to provide a broad overview of the entire system. Both types of prototypes are essential in the iterative process of software development and help teams gather feedback and make informed design decisions.

Vertical prototype:

- A vertical prototype focuses on the presentation of a specific feature or functionality of a system.
- It comprises a limited number of functions, but these are implemented in detail.
- The aim is to provide a comprehensive insight into a specific aspect of system functionality.
- Vertical prototypes are suitable for validating and testing certain functions early in the development process.

Horizontal prototype:

- A horizontal prototype is intended to provide a broad overview of the entire system.
- It contains a small subset of functions from different parts of the system and provides a horizontal cross-section of different functions.
- The main purpose is to demonstrate the general appearance of the system and show how the various components interact.
- Horizontal prototypes are often used in the early stages of design to get feedback on the overall user interface and information architecture.

Scenario prototype:

- All functions for a specific task are presented in a mixture of vertical and horizontal prototypes.

Mockups tend to be static representations of a design. They focus on visual aspects rather than functionality. They provide a visual guide to the look and feel of the final product. Mockups tend to be created in the early stages of the design process to communicate design concepts or visual aesthetics. They serve as a reference for designers, developers, and stakeholders.

6.3 UX design and software development processes

K-L	K2 - 20 minutes
	Discussion: Experience of the participants
	LO 6.3.1 Be able to name the difference between the waterfall model and agile development (K1) LO 6.3.2 Being able to explain the contents and benefits of design systems (K2)
	Waterfall model, SCRUM, agile development, sprint, cross-functional teams, design system

6.3.1 Development and design processes

Integration of UX into the Software Engineering Process
User Experience (UX) should not be treated as an add-on, but as an integral part of the software engineering lifecycle. UX goals and usage scenarios need to be defined already during requirements analysis. Throughout development, design systems and reusable patterns ensure consistency and efficiency. In agile processes, UX aspects are incorporated into user stories and acceptance criteria. Cross-functional teams guarantee continuous involvement of UX expertise. UX evaluations such as usability tests and heuristic reviews are a fixed part of quality assurance and continue during maintenance and further development after release. In this way, UX becomes a continuous component of overall product quality.

Software development processes are complex, and most companies use a customized version of the processes described in the literature. This topic is covered in detail as part of the *Advanced Level* certification.

In practice, very different approaches to the design of a user interface (UX) have become established. None of them are necessarily right or wrong. Depending on the environment, system, resources, qualifications, etc., one approach may be more suitable than another. A rough distinction can be made between the following types, although in most cases, a mixed form is "lived" in practice.

To be able to carry out UX design during the development process, experience has shown that it is very useful to have the necessary basic knowledge available throughout the team. This can be covered, for example, by involving UX experts and/or the corresponding knowledge (or knowledge building) of existing team members.

The waterfall model

Over the last few decades, industrial software development has evolved from a completely disorganized process to mature, high-quality, and largely standardized processes. The original process developed around 1970 is the classic "waterfall model" that is still used today.

UX processes can be easily integrated into the waterfall model.

Agile software development models

In practice, agile methods are better than traditional methods for developing high-quality software. The model most widely used is SCRUM.

Important aspects to consider are:

- SCRUM is not a method for implementing UX per se.
- Due to the short "sprints" (2 - 4 weeks), it is challenging to implement UX processes in each of these sprints.
- Integrate opportunities for direct feedback from end users to continuously evaluate the relevance and quality of the user experience.
- Train team members on the importance of UX and raise awareness of the user perspective.
- Formulate user stories with clear acceptance criteria that also cover UX aspects.
- Cross-functional teams, members with different skills, including UX experts.
- Integrate UX experts in the early stages of the project to ensure that user requirements and expectations are considered from the beginning of the project.

6.3.2 Design systems

A design system is a tool and a framework with clear context-related rules, principles, and reusable components - such as forms, tables, or buttons - for the design of systems. They determine how a team designs the user interface of a system. In addition to generally applicable principles, they also include code and UI components. They, therefore, represent a collection of reusable user interface elements and design patterns as well as rules and principles for their use in different contexts of the system.

The main objectives of a design system are to:

- Maintaining a consistent visual and interactive language for the different parts of a product or platform.

- Streamlining the design and development process.
- Improving cooperation between the teams.
- Facilitating scalability by providing a basis for future design and development work.

Various design systems for different environments are usually offered ready-made by large companies. The use of such systems is particularly advantageous for smaller teams and companies.

6.3.3 Design-Tokens/Variables

Design Tokens as a Bridge between Design and Code
Design tokens – such as colors, typography, spacing, and states – serve as the connecting layer between design and implementation. They define core design parameters in a way that is understandable for both designers and developers, ensuring that visual concepts are consistently translated into code. Modern tools increasingly support tokens through "variables" and "dev mode" features. Artificial intelligence can assist by generating suggestions for tokens or variations. However, the authoritative **source of truth** remains the central token repository, maintained and versioned by the teams. Technically, the work of the W3C Design Tokens Community Group provides the foundation for current standards.

6.3.4 AI as a Design Copilot

Artificial Intelligence can be used as a "design copilot" to accelerate routine tasks and support creativity – independent of specific tools.

Application areas and examples:

- **Variations & Wireframes:** AI can generate multiple layout suggestions for a dashboard (e.g., different placements of navigation bars or card views), allowing designers to quickly compare options.
- **Microcopy & Error Messages:** Instead of generic messages ("An error occurred"), AI can suggest contextual texts, such as "Please enter a valid email address."
- **Empty States:** For an empty favorites list, AI might propose: "No favorites yet – start by clicking the heart icon."
- **Localization:** AI can provide initial translation suggestions for microcopy, buttons, or tooltips, such as "Download" ⟶ "Herunterladen" or culturally adapted variants like "Get started" ⟶ "Jetzt beginnen."

Accessibility Checks (A11y, WCAG 2.2):

- **Contrast:** Automated check if text meets minimum contrast ratios (4.5:1 for body text).
- **Focus Indicators:** Ensuring that keyboard navigation always shows a visible focus outline.
- **Target Sizes:** Checking if buttons and interactive elements meet the minimum size of 24×24 px.
- **Drag-and-Drop Alternatives:** Detecting whether list reordering, for example, is also possible via keyboard.

Important Note: Real sensitive user data (e.g., customer names, health or financial data) must **never** be uploaded to external AI services. Instead, **test data or mock data** should be used.

7 Evaluation - understanding the user perspective

7.1 Basics of evaluations

K-L	K2 - 20 minutes
	LO 7.1.1 Explain the meaning and purpose of evaluations (K2) LO 7.1.2 Explain the difference between formative and summative evaluation (K2)
	Formative evaluation, summative evaluation

The early and continuous evaluation of user requirements and design solutions as part of a user-centered development process is a key activity **to ensure an efficient process and the quality of a final design solution**. Only an early and continuous evaluation enables resource-saving and efficient corrections to designs and requirements.

There are various options and methods that are suitable for the respective project situation and the available resources.

There are **two basic types of evaluation**, formative and summative evaluation, which essentially differ in terms of when and for what purpose they are carried out. In addition to this quality assurance orientation, evaluations can also contribute to **resolving areas of tension** in development teams.

Formative evaluation:

This is an **evaluation that accompanies** the **process to improve product quality** and shape the product. Usability engineering is a cyclical process of prototyping. The prototypes are evaluated and improved in an iterative process with the participation of future users. User participation during the evaluation phase guarantees a realistic review of the development steps. This reduces the risk of planning past the needs and behavior of the users. The target group is the project team itself.

Summative Evaluation:

This is a final evaluation against defined criteria, standards, or requirements. To check the goals set at the beginning for the design of a user-friendly interface, appropriate checks/measurements can be carried out on the finished end product. It only works if the system is in a relatively finished state.

7.2 Evaluation methods and procedures with user involvement

K-L	K2 - 90 minutes
	LO 7.2.1 Explain the basic structure and procedure of usability tests (K2) LO 7.2.2 List the most important components of a test plan (K2) LO 7.2.3. be able to explain test lead effects (K2) LO 7.2.4 Explain the contents of a test report and the corresponding difference between formative versus summative tests (K2) LO 7.2.5 Explain the fields of application, advantages, and disadvantages of constructive interaction (K2)

LO 7.2.6 Be able to explain the basics, possible applications, advantages, and disadvantages of the following supporting methods in usability tests: Thinking aloud, eye tracking, use of video

Usability tests, objectives, test duration, required infrastructure, development status of the system during implementation, test manager, test persons, tasks, test budgets, test procedure, test manager effects, constructive interaction, thinking aloud, eye tracking

7.2.1 Usability tests with physically present users

The usability test is usually a "package" in **which future users perform precisely defined tasks in a system or on prototypes**. They are **observed** and their actions are analyzed and interpreted. In addition, **questionnaires and/or interviews** are **usually** carried out before or after the test. Other methods, such as "thinking aloud," "video," or "eye tracking," can be used to support the implementation and evaluation. Such tests are suitable for gaining a first-hand impression of the users and drawing conclusions from their behavior.

For a usability test, it is necessary to have **appropriate premises** and ideally (but not necessarily) some **technical equipment so that** valid usability tests can be carried out, observed, and evaluated. An **external usability laboratory is an advantage but not absolutely necessary.**

Test plan

A detailed **test plan** must be drawn up before the test is carried out. **Test plans** usually contain the **following components**:

- Objective of the test
- Test duration
- Date and location of the test
- Required infrastructure
- Development status of the system at the time of implementation
- Test manager
- Test subjects
- Tasks to be performed
- Amount and composition of the test budget
- Test procedure

Representative procedure of a test session (excluding questionnaires, interviews, etc.)

- A test administrator conducts the test with the test subject.
- The test person is presented with the task in written form.
- She reads through them, and if she has any questions, she asks them straight away.
- The test person should then solve the task alone.
- If the test taker has problems while working on the task, they should actively contact the test administrator.
- The test leader then helps - according to a predefined scheme.
- I.e., step-by-step approach to the solution.

Results and test report

The following are the usual results of a formative evaluation test. These are also components of a test report, which should also include characteristics of the tested users, tasks used, screenshots, or availability of the tested state of the design solution:

- Positive usability / UX aspects (avoidance of deterioration and motivation!)
- Usability problems in detail
- Quantified (how many people, etc.)
- If possible: Causes
- Evaluation (often traffic light system) for redesign decision process
- Proposed solutions/remedies

The following are the usual results of a summative evaluation test. These are also components of a test report, which should also include characteristics of the tested users, tasks used, screenshots, or availability of the tested state of the design solution.

Most important components of a test report of a summative evaluation (and components of a test report):

- Positive usability / UX aspects (avoidance of deterioration and motivation!)
- Fulfillment of or deviations from the benchmarks set
- Evaluation (often traffic light system) for acceptance decision process

Test lead effects

In a test, it is important that the test **administrator** is **appropriately trained** or at least aware of the **test administrator effects**. The test administrator effect refers to the influence of certain characteristics or behaviors of the test administrator on a test result.

Judgment error:

Leniency error: The tendency of an examiner to judge the performance of the person being tested too leniently and thus introduce systematic errors into the performance or evaluation of a test. In usability tests, for example, this is the case when user problems are attributed to external factors instead of the user's knowledge. The trigger for this is very often an unpleasant context for the user (noise, late time, waiting time, technical problem, etc.) or personal sympathy for the test person. In such situations, the test administrator is often in a milder mood.

Hardness error: The tendency of an examiner to judge the performance of the person being tested too strictly and thus introduce systematic errors into the administration or evaluation of a test. This is very often triggered by an unconsciously perceived dislike of the test subject or an unpleasant personal situation of the test administrator (e.g., stress, thirst/hunger).

In addition, the personal **characteristics of the test administrator** can influence the interaction between the test administrator and test subject, which can indirectly (e.g., through uncertainty or reduced motivation of the test subject) distort the test results. The effect of **body language** and **clothing**, and sometimes **gender,** can also influence the behavior of the test subject.

7.2.2 Remote usability tests

Remote usability testing is a usability test method in which the test subject is not in the same room as a test manager but is actually somewhere else in the world. There are two main types, the moderated and the unmoderated remote test.

In a moderated remote test, the test subject and test manager are connected via a video call tool. The test subject works on the test tasks in a similar way to a conventional usability test and is guided and, if necessary, supported by the test manager. The system to be tested must be made available to the test person in such a way that they can access it from their location. Otherwise, the basic procedure corresponds to that of the conventional usability test.

Such remote tests have significantly improved the range of feedback options in the UCD lifecycle and made them more easily accessible to more organizations. They are a particularly good option for formative evaluations.

In unmoderated remote testing, the tester works independently of time and/or location on predefined test tasks without a test manager being available. The system to be tested must be made available to the test person in such a way that they can access it from their location at any time. Ideally, the test person should record their screen and speak aloud what they are doing or thinking. The video/audio and possibly predefined questionnaires are then used for evaluation.

There are also numerous providers on the market who offer remote tests as a service. With these services, the planning of the test (including test tasks, questionnaires, desired metrics, and test subject profiles) is defined online. The output includes videos of the test subjects at work (usually with picture-in-picture video of the screen and face) as well as evaluations of the metrics and questionnaires.

> **!** The Advanced Level discusses the possibilities, advantages, and disadvantages of moderated and unmoderated remote tests in more detail. The aim here is only to show that these options exist.

7.2.3 A/B tests

A/B testing, also known as split testing, is a method used in user experience design and website optimization to compare two versions of a design and determine which of them better contributes to achieving specific goals. The goal is to evaluate and identify design changes or variations that lead to improved user engagement, conversions, or other key performance indicators.

In A/B tests, two or more variants (A, B, C, etc.) of a website, screen, or user interface are created. These variants contain different design elements, such as layout, colors, fonts, or call-to-action buttons. Users are randomly assigned to one of the variants when they visit the website or use the application. This ensures that the groups are comparable and that any differences in user behavior can be attributed to the design changes. User interactions and behavior are tracked and measured for each variant. Common metrics include click-through rates, conversion rates, bounce rates, and other relevant KPIs.

Once a sufficient amount of data has been collected, a statistical analysis is carried out to determine whether there are significant differences in user behavior between the variants. This analysis can be used to determine which design changes have a positive influence on the desired results.

Implementation of the winning variant: Based on the results, the variant that performs better in terms of desired behavior is used as the basis for further optimization.

7.2.4 Constructive Interaction

In this method, **two people work together to solve tasks with the system/prototype**. The interaction or **discussion between the people is the focus** of the observation. This is often very helpful in understanding motivations or reasons for actions. With this method, it is particularly important to ensure that both people act and not just one. Frequently used with children and senior citizens.

7.2.5 Supporting methods

Thinking Aloud (Thinking Aloud)

While performing a task, the user is encouraged to "think aloud," i.e., **to comment on** their **actions and motives**. This often makes it easier for the test administrator to understand the actions or behavior of the test subject. Caution: However, you **cannot assume that users really say everything** - keyword: self-presentation effect! In addition, **"thinking aloud" also distracts from the task itself,** and performance decreases.

Eye tracking

Eye tracking refers to the **recording of** a person's **eye movements, which mainly consist of fixations** (points that are looked at closely), **saccades** (rapid eye movements), and regressions (backward jumps). This method is used in the course of usability studies to draw **conclusions about the behavior, understanding, or problems of test subjects**.

Eye tracking data must be interpreted with **great care**. Overzealous **misinterpretations are problematic**! The observation that someone first looks at the header on a screen page, for example, does not allow any qualitative conclusions to be drawn as to why this is the case - this requires additional questioning of the person or the method of "thinking aloud."

Use of video

Users or the screen are **recorded on video while performing a task**. The **video is then discussed with the person concerned**. They are asked to explain and justify what they have done. This approach is particularly helpful with **complex systems if it is** not possible to record or scrutinize everything during the test.

7.3 Expert-based procedures (expert reviews)

K-L	K2 - 30 minutes
	LO 7.3.1 Explain the application, strengths, and weaknesses of a cognitive walkthrough (K2) LO 7.3.2 Explain the application, strengths, and weaknesses of a heuristic evaluation (K2)
	Cognitive walkthrough, heuristic evaluation, visibility of system status, interaction between real world and system, user control and freedom, consistency and standards, error prevention, recognition is better than remembering, flexibility and efficiency in use, aesthetics and minimalist design, user support in recognizing, diagnosing and correcting errors, help and documentation

7.3.1 Cognitive walkthrough

Based on an existing task analysis (task analysis) or on the tasks broken down into their subtasks. The project team (designer, developer, etc.) "walks" through the system - step by step according to the broken-down tasks from the analysis - and repeatedly checks the following questions during this process: ~~Original by C. Wharton, four questions; shortened by Spencer, two questions:~~

Questions from Wharton

- Will the user try to achieve the right effect?
- Will the user notice that the right function is available?
- Will he associate this function with the desired effect?
- Once he has performed the function, does he think he is closer to his goal?

Questions from Spencer

- Will the user know what to do in this situation/condition?
- If he has set the action, will he know whether he was successful or whether he has set the desired action with the corresponding result?

Disadvantages/Problems

- The evaluators themselves do not necessarily know how a task should be carried out (e.g., subject-specific characteristics). As a result, they may make incorrect assumptions.
- The method is very dependent on a very careful task analysis.
- No real users wander through the system - sometimes, experts identify problems that users don't even perceive as such.

7.3.2 Heuristic evaluation

Heuristics ([find], discover) refers to the art of arriving at good solutions with limited knowledge ("incomplete information") and little time. It refers to an analytical approach in which conclusions or statements about a system are made with limited knowledge of the system with the help of assumptions.

In a heuristic procedure, the system is evaluated on the basis of predefined heuristics. The underlying assumption is that if the heuristics are fulfilled, the system as a whole is also easy to use.

Procedure

- Several evaluators assess the system - independently of each other.
- You go through all views/screens/windows individually and evaluate them using all heuristics.
- Several runs are usually necessary.
- The evaluators then compare and discuss their results and define a prioritized list of problems.

Disadvantages:

- The task orientation is not represented.
- The method requires a lot of practice from the evaluators in order to work efficiently and validly.

Heuristics by Jacob Nielsen - 10 points

The best-known heuristics come from Jacob Nielsen - the inventor of heuristic evaluation.

Visibility of system status: The system should always keep the user informed about what is going on - by providing appropriate feedback within a reasonable time.

Interaction between the real world and the system: The system must speak the language of the user with words, phrases, symbols, and concepts. Conventions from the real world should be adopted, and information should be presented logically and naturally.

User control and freedom: Users often unintentionally use a function/navigation - the system must provide a clear "emergency exit." Undo and redo must always be offered.

Consistency and standards: Users should not have to wonder whether different terms, representations, or elements mean the same and different things in different situations.

Error prevention. Error prevention through careful design is better than a good error message. Either you manage to eliminate error-prone situations, or you let the user confirm them with an additional command (button) for critical or complex actions.

Recognition is better than remembering: The memory effort of the user is minimized by the fact that actions, information etc., are presented, and the user does not have to know them by heart. This functionality should be supported when switching between different windows/views.

Flexibility and efficiency in use: Accelerating interaction elements (e.g., shortcuts) - invisible to the inexperienced user - often help to support different user groups.

Aesthetics and minimalist design: Dialogs should not contain any information or elements that are irrelevant or only very rarely needed. Every irrelevant information unit competes with the relevant ones for the user's attention and, therefore, reduces their perception.

Supporting the user in recognizing, diagnosing, and correcting errors: Error messages must be written in simple language and offer the user the opportunity to recognize the error and understand the possible solutions.

Help and documentation: Even if it is better for a system to manage without documentation, there are still systems that require it. Appropriate help or documentation must be easy to search, task-oriented, and focused on the essentials.

7.4 Analysis, prioritization and implementation of findings

K-L	K2 - 20 minutes
	LO 7.4.1 Explain the prioritization steps for necessary changes based on the findings of an evaluation (K2)
	Categorization, severity assessment, frequency analysis, priority matrix

Prioritizing the necessary changes based on the findings of an evaluation involves assessing the severity, impact, and frequency of the usability issues identified. The following is a breakdown of a valid process:

Review the results of the usability test, including direct observations, user feedback, collected usability metrics, and specific usability issues encountered by users during the test.

Categorize the usability issues. Categorize the identified issues according to their type, e.g., navigation issues, layout issues, confusing terminology, etc. This categorization helps organize and understand the types of issues that users face.

Severity rating: Assignment of a severity level to each usability problem. Common scales are critical, important, and unimportant, or high, medium, or low. Critical problems are those that severely impair security or performance and require immediate attention.

Impact **assessment**: Assess the impact of each issue on user satisfaction, task completion, and overall user experience. This should also consider the potential business impact of fixing or not fixing each issue.

Frequency analysis: Analysis of the frequency of each identified problem. If a particular problem occurs for several users, it should be given a higher priority.

Priority matrix: Create a priority matrix that considers severity, impact, and frequency. Problems should be prioritized based on their position in the matrix, giving more attention to those with higher severity.

7.5 Questionnaires

K-L	K2 - 30 minutes
	LO 7.5.1 Know the following questionnaire methods and their objectives/applications: SUS, UEQ, ISO Metrics (K2)
	SUS, UEQ, ISO Metrics

7.5.1 Use and benefits of ready-made questionnaires

Questionnaires are a good method for ascertaining a user's subjective satisfaction with a system. However, you should definitely refrain from presenting users with self-designed questions, as this very quickly leads to inferior data and a correspondingly poor basis for decision-making. Questionnaire development requires extensive knowledge that can be acquired in social science studies, for example.

There are many high-quality questionnaires on various aspects of subjective assessments of systems. In the Foundation Level, three are presented.

7.5.2 SUS (System Usability Scale)

SUS is a very simple yet reliable **method for users** to **assess the usability of a system (hardware, software, websites, mobile devices).** The SUS questionnaire consists of ten items (statements), each with five possible answers, ranging from "completely agree" to "strongly disagree."

SUS **does not help to determine which usability problems exist**; rather, the method enables an assessment of the usability or the tested system.

The evaluation of the individual questionnaires follows a simple scheme. The evaluation results in a score between 0 and 100, whereby this does not represent a percentile. Experience and research show that a score above 68 indicates good usability.

Items from the SUS, in the original:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

7.5.3 UEQ (User Experience Questionnaire)

The User Experience Questionnaire (UEQ) is a standardized instrument for evaluating the overall user experience of products, systems, or services. It was developed to provide a reliable and valid measure of various aspects of user experience, enabling designers and researchers to collect user feedback in a structured and quantitative way. The UEQ is particularly useful for usability studies, user interface evaluations, and product development.

The UEQ follows a standardized format consisting of a series of questions that users answer to express their perceptions and experiences with a product.

The questionnaire evaluates the user experience in several dimensions, including:

- Attractiveness,
- Comprehensibility (clarity),
- Efficiency,
- Reliability,
- Stimulation and
- Novelty.

Users give their answers on a scale that usually ranges from "strongly disagree" to "strongly agree." The ratings are then compiled and analyzed to provide quantitative data about the user experience. The data collected with the UEQ can be analyzed to identify strengths and weaknesses of the user experience and help designers and developers make decisions for improvements.

Adaptability: The UEQ can be adapted to specific needs as follows:

- by adding additional scales
- using the long or the short version
- and the availability in 30 languages

can be tailored to specific contexts or areas so that it can be used for a wide range of products and services.

7.5.4 ISOMetrics

This is a method for **evaluating software based on ISO 9241-110.** There are two versions of the ISOMetrics method. Both use the same items.

- ISOMetrics S (short) enables **exclusively numerical evaluation**.
- ISOMetrics L (long) can be used for the numerical and **qualitative, design-supporting evaluation of** software.
- Available in a **German and an English** version.
- ISOMetrics S can be completed in approximately 30 to 60 minutes.
- For ISOMetrics L, at least two hours (including processing test tasks) per participant must be expected.
- **Seven subscales according to the design principles of ISO 9241-110** with a total of 75 items, which are evaluated using a rating scale.
- The long version has an additional rating scale for each item to assess its importance as well as a space for the presentation of specific examples that describe the weaknesses of the system in relation to the content of the item.

8 Recommended literature

General usability and user experience

- ✓ Barnum Carol, Usability Testing Essentials: Ready, Set ...Test!: Ready, Set...Test, Morgan Kaufman, 2020
- ✓ Cooper Alan, The Inmates Are Running the Asylum: Why High-tech Products Drive Us Crazy and How to Restore the Sanity, SAMS 2004
- ✓ Gothelf Jeff, Lean UX, O'Reilly, 2013
- ✓ Krug Steve, Rocket Surgery Made Easy, New Riders, 2010
- ✓ Krug Steve, Don't Make Me Think! Web Usability: The intuitive web, 2nd edition, Redline, Heidelberg, 200
- ✓ Levy Jaime. UX Strategy: Product Strategy Techniques for Devising Innovative Digital Solutions, O'Reilly Media, 2021
- ✓ Nielson Jakob, Designing Web Usability, New Riders, 1999
- ✓ Preece Jenny et al., Interaction Design, Wiley, 2012
- ✓ Robier Hannes, UX Redefined: Winning and Keeping Customers with Enhanced Usability and User Experience, Springer, 2015
- ✓ Seibert-Giller, Pucher, Usability and User Experience: Exam preparation for the Certified Professional for Usability Engineering and User Experience Design, UXKnowhow 2019
- ✓ Tullis Tom, Morgan Kaufmann, Measuring the User Experience, Morgan Kaufmann, 2022
- ✓ Leyton Ryland, The Agile Business Analyst: Moving from Waterfall to Agile, self-published, 2015
- ✓ Yifrah, UX Writing & Microcopy: Good texts for better usability and user experience, Reinwerk 2019

Perception, Psychology

- ✓ Møller Aage R., SENSORY SYSTEMS: Anatomy and Physiology, Second Edition, Aage R. Møller Publishing, 2012
- ✓ Anderson J., Cognitive Psychology and Its Implications, Worth, 2014
- ✓ Hunziker Hans-Werner, Im Auge des Lesers: vom Buchstabieren zur Lesefreude; foveale und periphere Wahrnehmung, , transmedia verlag, Zurich, 2006
- ✓ Metzger, Spillmann, Laws of Seeing, MIT Press, 2009
- ✓ McLeod June, Color Psychology Today, John Hunt Publishing Ltd, 2016
- ✓ Schubert S., Eibl C., Die 3 Gestaltgesetze, symposium of the University of Siegen "Didaktik der Informatik und E-Learning," 2007
- ✓ Seibert-Giller Verena, UX Psychology Lens, UX Knowhow, 2024
- ✓ Weinschenk Susan, 100 Things Every Designer Needs to Know About People, New Riders, 2010
- ✓ Young Indi, Mental Models: Aligning Design Strategy with Human Behavior, Rosenfeld Media, 2008
- ✓ Schmidt Robert F. (Editor), Neuro- und Sinnesphysiologie 5th edition, Springer Verlag, 1993

Standards and norms

- ✓ International Electrotechnical Commission; IEC 62366-1:2015
- ✓ International Organization for Standardization; ISO 9241-110; ISO 9241-210
- ✓ International Organization for Standardization; ISO/IEC40500 (W3C, Web Content Accessibility Guidelines 2.2)