Certified Tester Usability Testing (CT-UT) Syllabus

Version 2.0

International Software Testing Qualifications Board



© International Software Testing Qualifications Board

Copyright Notice

Copyright Notice © International Software Testing Qualifications Board (hereinafter called ISTQB®)

ISTQB® is a registered trademark of the International Software Testing Qualifications Board.

Copyright © 2026 the authors for the update 2026 Graham Bath (WG Chair), Andrew Pollner (WG Vice Chair), Kimberly Snoyl

Copyright © 2018 the authors for the update 2018 Graham Bath (WG Chair), Kai Lepler, Rolf Molich (UXQB)

All rights reserved. The authors hereby transfer the copyright to the ISTQB®. The authors (as current copyright holders) and ISTQB® (as the future copyright holder) have agreed to the following conditions of use:

Extracts, for non-commercial use, from this document may be copied if the source is acknowledged. Any Accredited Training Provider may use this syllabus as the basis for a training course if the authors and the ISTQB® are acknowledged as the source and copyright owners of the syllabus and provided that any advertisement of such a training course may mention the syllabus only after official Accreditation of the training materials has been received from an ISTQB®-recognized Member Board.

Any individual or group of individuals may use this syllabus as the basis for articles and books, if the authors and the ISTQB® are acknowledged as the source and copyright owners of the syllabus.

Any other use of this syllabus is prohibited without first obtaining the approval in writing of the ISTQB®.

Any ISTQB®-recognized Member Board may translate this syllabus provided they reproduce the Copyright Notice mentioned above in the translated version of the syllabus.

© International Software Testing Qualifications Board

Revision History

Version	Date	Remarks
v1.0	2018/07/08	ISTQB Release v1.0
v2.0	2026/10/30	ISTQB Release v2.0

© International Software Testing Qualifications Board

Table of Contents

Revision History 3
Table of Contents 4
Acknowledgements 7
0 Introduction 8
0.1 Purpose of this Syllabus 8
0.2 The Usability Testing Qualification 8
0.3 Career Path for Testers 8
0.4 Business Outcomes 9
0.5 Learning Objectives and Knowledge Level 9
0.6 The Usability Testing Certificate Exam 10
0.7 Accreditation 10
0.8 Handling of Standards 10
0.9 Level of Detail 10
0.10 How this Syllabus is Organized 11
1 Basic Concepts – 75 minutes 13
1.1 Fundamentals 14
1.1.1 Usability 14
1.1.2 User Experience 15
1.2 Evaluating Usability and User Experience 16
1.2.1 Usability Evaluation 17
1.2.2 User Experience Evaluation 18
1.3 Usability Evaluation and Human-Centered Design 19
2 Risks in Usability and User Experience – 90 minutes 22
2.1 Introduction 23
2.2 Typical Risks 23
2.2.1 Usability Risks 23
2.2.2 User Experience Risks 24
2.2.3 Project Risks 24
3 Usability Standards – 30 minutes 26
3.1 Usability Standards 27
3.1.1 User Interface Guidelines 27

© International Software Testing Qualifications Board

3.1.2 ISO Usability Standards 27
4 Usability Reviews – 105 minutes 29
4.1 Introduction and Approach 30
4.1.1 Step-By-Step Approach to Usability Reviews 30
4.1.2 Risks and Challenges 31
4.2 Types of Usability Review 31
4.2.1 Informal Usability Reviews 31
4.2.2 Heuristic Evaluation 32
5 Usability Testing – 345 minutes 34
5.1 Introduction 35
5.2 Step-By-Step Approach to Usability Testing 35
5.3 Preparing a Usability Test 36
5.3.1 Usability Test Plan 36
5.3.2 Usability Test Script 37
5.3.3 Usability Test Tasks 37
5.3.4 Usability Test Location 38
5.3.5 Usability Test Lab 38
5.4 Performing Usability Test Sessions 40
5.5 Analyzing Findings 40
5.5.1 Procedure for analyzing findings from a usability test 40
5.6 Communicating Test Results and Findings 41
5.6.1 Test Results 41
5.6.2 Usability Findings 42
5.6.3 Selling Findings to Stakeholders 42
5.7 Quality Control of a Usability Test 45
5.8 Challenges and Frequent Errors 46
6 Usability Surveys – 30 minutes 48
6.1 Step-By-Step Approach to User Surveys 49
6.2 Standardized Questionnaires 49
7 Selecting Appropriate Test Techniques for Usability – 60 minutes 51
7.1 Criteria for Selecting an Evaluation Technique 52
7.1.1 Selecting a Usability or User Experience Evaluation Technique 52

© International Software Testing Qualifications Board

8 Usability Testing in Agile Software Development – 60 minutes 54
8.1 Integrating Usability Testing into Agile Software Development 55
8.2 Roles in Agile Software Development 56
8.3 Usability Evaluation Approaches Suitable in Agile Software Development 56
8.4 Shifting the Focus to User Experience 57
8.4.1 Integrating personas in user stories 57
8.4.2 Close collaboration in the cross-functional team 57
9 Select and Use Tools for Usability Testing - 45 minutes 59
9.1 Categories of Usability Testing Tools 59
9.2 Application of Usability Testing Tools 60
9.3 Limitations of Usability Testing Tools 60
10 AI Support for Usability Testing - 15 minutes 62
10.1 Areas in Which AI Can Support Usability Testing 63
11 References 64
11.1 Standards 64
11.2 ISTQB Documents 64
11.3 Books 64
11.4 Other References 65
12 Appendix A – Learning Objectives/Cognitive Level of Knowledge 66
13 Appendix B – Business Outcomes Traceability Matrix with Learning Objectives 69
14 Appendix C – Release Notes 79
15 Appendix D – Domain Specific Terms 80
16 Appendix E – Manufacturer Guidelines 81
17 Index 82

© International Software Testing Qualifications Board

Acknowledgements

This document was formally released by the General Assembly of the ISTQB® on 30th October 2026.

It was produced by a team from the International Software Testing Qualifications Board: Graham Bath (WG Chair), Andrew Pollner (WG Vice Chair), Kimberly Snoyl (Key Author),

The following persons participated in the reviewing, commenting and balloting of this syllabus:

Laura Albert, Ibrahim Anifowoshe, M.N.A. van den Berg, Taz Daughtrey, Ray Gillespie, Ferdinand Gramsamer, Andre Kester, Niranjan Maharajh, Aneta Masárová, Judy McKay, Korkor Mensah, Imola Mészár, Gary Mogyorodi, Roshni Naidoo, Frank Neirynck, Tauhida Parveen, Tal Pe'er, Nishan Portoyan, Meile Posthuma, Randall Rice, Annelies van Rijn, Mark Rutz, Klaus Skafte, Michael Stahl, Lucjan Stapp, Ioannis Symeonidis, Richard Taylor, Nilo Thomas, Tomas Tumasonis, Linda Vreeswijk

International Software Testing Qualifications Board Working Group Specialist Level (Version 1.0):

Graham Bath (GTB), Kai Lepler, Rolf Molich (UXQB)

© International Software Testing Qualifications Board

0 Introduction

0.1 Purpose of this Syllabus

This syllabus forms the basis for the International Software Testing Qualification for Usability Testing at Specialist level. The ISTQB® provides this syllabus as follows:

1. To member boards, to translate into their local language and to accredit training providers. Member boards may adapt the syllabus to their particular language needs and modify the references to adapt to their local publications.
2. To certification bodies, to derive examination questions in their local language adapted to the learning objectives for this syllabus.
3. To training providers, to produce training materials and determine appropriate teaching methods.
4. To certification candidates, to prepare for the certification exam (either as part of a training course or independently).
5. To the international software and systems engineering community, to advance software and systems testing profession, and as a basis for books and articles.

0.2 The Usability Testing Qualification

The Usability Testing qualification is aimed at anyone involved in software testing who wishes to broaden their knowledge of usability testing or anyone who wishes to start a specialist career in usability testing. The qualification is also aimed at anyone involved in usability engineering who wishes to understand usability testing.

0.3 Career Path for Testers

The ISTQB® scheme supports testing professionals at all stages of their careers offering both breadth and depth of knowledge. Individuals who achieve the ISTQB® Usability Testing certification may also be interested in the Digital Accessibility Testing certification. [ISTQB_SYL_DAT]

© International Software Testing Qualifications Board

0.4 Business Outcomes

This section lists the Business Outcomes expected of a candidate who has achieved the Usability Testing certification.

A Usability Testing Certified Tester can…

UT-BO1	Understand the basic concepts of usability and usability testing.
UT-BO2	Identify and classify the severity of usability risks in a given product at any phase of a development cycle.
UT-BO3	Cite relevant standards for usability and user experience and verify their implementation in a given product.
UT-BO4	Set up procedures so that stated usability and user experience goals may be verified in practice for a given product.
UT-BO5	Design and monitor the implementation of a test plan and other testware for achieving stated usability and user experience goals.
UT-BO6	Explain the rationale, process and results of usability and user experience evaluations to non-specialist stakeholders.
UT-BO7	Outline important criteria for selecting a usability evaluation technique.
UT-BO8	Give examples of usability testing in product delivery using an Agile software development.
UT-BO9	Classify tools and Artificial Intelligence (AI) support for usability testing.

0.5 Learning Objectives and Knowledge Level

Learning Objectives support the business outcomes and are used to create the Usability Testing exams.

In general, all contents of this syllabus are examinable, except for the Introduction and Appendices. The exam questions will confirm knowledge of keywords at K1 level (see below) or learning objectives at the respective level of knowledge.

The specific learning objectives and their levels of knowledge are shown at the beginning of each chapter, and classified as follows:

- K1: Remember
- K2: Understand
- K3: Apply
- K4: Analyze

Further details and examples of learning objectives are given in Appendix A.

For all terms listed as keywords just below chapter headings, the correct name and definition from the ISTQB® glossary shall be remembered (K1), even if not explicitly mentioned in the learning objective.

© International Software Testing Qualifications Board

0.6 The Usability Testing Certificate Exam

The Usability Testing Certificate exam will be based on this syllabus. Answers to exam questions may require the use of material based on more than one section of this syllabus. All sections of the syllabus are examinable, except for the Introduction and Appendices. Standards and books are included as references, but their content is not examinable, beyond what is summarized in the syllabus itself from such standards and books.

Refer to Exam Structures and Rules document for the Usability Testing module for further details.

The entry criterion for taking the Usability Testing Specialist Level exam is that candidates are interested in software usability testing. However, it is strongly recommended that candidates also:

• Have at least a minimal background in either software development or software testing, such as six months experience as a system or user acceptance tester or as a software developer

• Take a course that has been accredited to ISTQB® standards (by one of the ISTQB-recognized member boards).

Entry Requirement Note: The ISTQB® Foundation Level certificate shall be obtained before taking the Usability Testing certification exam.

0.7 Accreditation

An ISTQB® Member Board may accredit training providers whose course material follows this syllabus. Training providers should obtain accreditation guidelines from the Member Board or body that performs the accreditation. An accredited course is recognized as conforming to this syllabus and is allowed to have an ISTQB® exam as part of the course.

The accreditation guidelines for this syllabus follow the general Accreditation Guidelines published by the Processes Management and Compliance Working Group.

0.8 Handling of Standards

International standardization organizations like IEEE and ISO have issued standards associated with quality characteristics and software testing. Such standards are referenced in this syllabus. The purpose of these references is to provide a framework (as in the references to ISO 25010 regarding quality characteristics) or to provide a source of additional information if desired by the reader. Please note that ISTQB® syllabi are using the standard documents as reference. Standards documents are not intended for examination. Refer to chapter 11.1 for more information on Standards.

0.9 Level of Detail

The level of detail in this syllabus allows internationally consistent courses and exams. To achieve this goal, the syllabus consists of:

- General instructional objectives describing the intention of the Usability Testing, Specialist Level

© International Software Testing Qualifications Board

- A list of terms that students must be able to recall
- Learning objectives for each knowledge area, describing the cognitive learning outcome to be achieved
- A description of the key concepts, including references to sources such as accepted literature or standards

The syllabus content does not describe the entire knowledge area of software usability testing; it reflects the level of detail to be covered in Usability Testing, Specialist Level training courses. It focuses on test approaches and test techniques that can apply to all software projects, including those following Agile software development.

The syllabus uses the terminology (i.e. the name and meaning) of the terms used in software testing and quality assurance according to the ISTQB® Glossary (ISTQB® Glossary, Chapter 7.3)

For the terminology in related disciplines please refer to the respective glossaries: IREB-CPRE for requirements engineering (IREB-CPRE, Chapter 7.3), and Agile Alliance for agile terminology (Agile Alliance, Chapter 7.3).

0.10 How this Syllabus is Organized

There are ten chapters with examinable content. The top-level heading for each chapter specifies the time for the chapter; timing is not provided below the chapter level. For accredited training courses, the syllabus requires a minimum of 14 hours and 15 minutes of instruction, distributed across the ten chapters as follows:

- Chapter 1: 75 minutes – Basic Concepts
  - The tester learns the basic concepts related to usability testing.
  - The tester understands the human-centered design approach, and work products.
- Chapter 2: 90 minutes – Risks in Usability and User Experience
  - The tester learns the typical risks in usability and user experience.
  - The tester can identify the types of usability and user experience risks that are likely to occur.
- Chapter 3: 30 minutes – Usability Standards
  - The tester understands the general content and applicability of standards and guidelines for usability.
- Chapter 4: 105 minutes – Usability Reviews
  - The differences between informal usability review and expert usability review are covered.
  - The tester can apply Heuristic Evaluation to analyze the usability aspects of a software application.
- Chapter 5: 345 minutes – Usability Testing

© International Software Testing Qualifications Board

- The tester learns what could be included in the various testware, such as test plans and reports.
- The tester can design, prepare and execute usability testing sessions.
- Best practices can be used to report and communicate usability test results.

- Chapter 6: 30 minutes – Usability Surveys
  - The tester learns the principal content and objectives of the public user questionnaires.
- Chapter 7: 60 minutes – Selecting Appropriate Techniques
  - The tester learns how to select appropriate test techniques for verifying and validating usability and user experience.
- Chapter 8: 60 minutes – Usability Testing in Agile Software Development
  - The tester learns how to plan, execute and report on usability tests in an agile project
- Chapter 9: 45 minutes – Select and use tools for usability testing
  - The tester can classify tools for usability testing
- Chapter 10: 15 Minutes – Artificial Intelligence (AI) support for Usability Testing
  - The tester is aware of the support provided by AI in Usability Testing

© International Software Testing Qualifications Board

1 Basic Concepts – 75 minutes

Keywords

effectiveness, efficiency, formative evaluation, summative evaluation, usability, usability evaluation, user experience

Domain specific keywords

human-centered design, user interface

Learning Objectives for Chapter 1:

1.1 Fundamentals

UT–1.1.1 (K2) Explain the concepts of usability and user experience

UT–1.1.2 (K2) Give examples of the objectives of usability and user experience evaluation

1.2 Evaluating Usability and User Experience

UT–1.2.1 (K2) Compare usability and user experience

UT–1.2.2 (K2) Explain the two approaches to usability and user experience

1.3 Usability Evaluation and Human-Centered Design

UT–1.3.1 (K2) Explain the key elements of human-centered evaluation

© International Software Testing Qualifications Board

1.1 Fundamentals

This section considers the following fundamental concepts:

- Usability
- User experience

Note that the subject of accessibility is covered in a separate ISTQB Syllabus (see [ISTQB_SYL_DAT]). The Usability Testing syllabus provides a high-level overview of Software Accessibility topics to enable comparisons to be made between usability, user experience and accessibility.

1.1.1 Usability

Usability focusses on the extent to which a software system can be used by users to provide them with an effective, efficient and a satisfactory experience when using a system in a specified context of use.

The user interface consists of all components of a software system that provide information and controls for the user to accomplish specific tasks with the system.

Usability evaluation includes the following principal activities:

- Usability reviews (see Chapter 4)
- Usability testing (see Chapter 5)
- User surveys (see Chapter 6)

A usability issue is a software defect which results in difficulty in performing tasks via the user interface. This affects the ability to show achievement of test objectives in terms of effectiveness, efficiency, or satisfaction. Usability issues can lead to confusion, error, delay or outright failure to complete some tasks on the part of the user. In safety-critical systems such as medical systems, usability issues can also lead to injuries or death.

A software system can work exactly to specification and still have serious usability issues, as shown by the following examples:

- A car rental mobile app has 2 pages that are commonly used consecutively but are not directly linked, so the user must navigate between them via other pages. This is a defect which results in a less effective work flow for the user.
- A car rental mobile app allows users to cancel a reservation, but the users perceive the cancellation procedure as unreasonably complicated. This is a usability defect which affects the efficiency of the mobile app.
- A car rental mobile app conforms to the specification and works both effectively and efficiently, but users think it looks unprofessional. This is a usability issue which affects user satisfaction when using the mobile app.

Usability always relates to the context of use and can be considered in different components. As the following examples show, user expectations of usability are rather different for these components.

© International Software Testing Qualifications Board

Aspect	Description in Context of Use
Users	A user is a person who interacts with a software system by providing inputs, or by using the output of the software system.
Tasks	Particular activities performed by users or particular groups of users (e.g., inexperienced users, administrators).
Equipment	Equipment relates to the hardware, software and materials required to use a software system.
Environment	The environment consists of the physical, social and technical conditions in which a user interacts with a software system. The social conditions include the organizational conditions.

The following scenarios describe different contexts of use for the same software system.

- Administrative staff use word processing software to write documents in a consultancy firm
- An elderly person uses a word processing application for the first time to write an invitation to her birthday

1.1.2 User Experience

User experience describes a person's perceptions and responses that result from the use and/or anticipated use of a product, system or service. [ISO 9241-210]

User experience includes the following user characteristics that occur before, during and after use:

- emotions
- beliefs
- preferences
- perceptions
- physical and psychological responses
- behaviors and accomplishments

User experience is influenced by:

- brand image (i.e., the users' trust in the manufacturer)
- presentation (i.e., the appearance of the software system, including packaging and documentation)
- functional suitability
- software system performance efficiency
- interactive behavior
- the helpfulness of the software system, including help system, support and training
- learnability
- the user's internal and physical state resulting from prior experiences, attitudes, skills, personality, education and intelligence
- the context of use

© International Software Testing Qualifications Board

Usability criteria such as effectiveness, efficiency and satisfaction can be used to assess aspects of user experience such as brand image and presentation (satisfaction), functionality (effectiveness) and software system performance (efficiency).

1.2 Evaluating Usability and User Experience

The key objectives of usability evaluation and user experience evaluation are compared in the following table and discussed in more detail in subsequent sections.

Type of evaluation	**Key objective**	**See section**
usability evaluation	● Evaluate the direct interaction between users and the software system.	1.2.1
User experience evaluation	● Evaluate the services received prior to the use of the software system. ● Evaluate the direct interaction between users and the software system. ● Evaluate the services received after the use of the software system.	1.2.2

The principal test techniques applied in usability evaluation and user experience evaluation are shown in the following table and discussed in more detail in later chapters.

© International Software Testing Qualifications Board

Technique	Users involved?	Key characteristic	Specific techniques	Type	See Chapter
Usability review	Optionally	Test experts and users evaluate the user interface of a software system for usability issues; the evaluation is based on their experience.	Informal usability review Expert usability review Heuristic evaluation	Qualitative	4
Usability testing	Yes	Users are observed while performing typical tasks with the software system.	Think aloud usability testing	Qualitative, Quantitative	5
User surveys	Yes	Users fill out questionnaires regarding their satisfaction with the software system.	-	Qualitative, Quantitative	6

1.2.1 Usability Evaluation

A usability evaluation is a process through which information about the usability of a system is gathered in order to improve the system (known as formative evaluation) or to assess the merit or worth of a system (known as summative evaluation).

There are two types of usability evaluation:

- Formative (or “exploratory”) evaluation is performed to understand usability defects. Formative evaluation is often performed early on in the software development lifecycle during the design and prototyping phases to get ideas and to guide (or “form”) the design by identifying usability design defects.
- Summative evaluation is performed late in the software development lifecycle shortly before or after implementation to measure the usability of a component or software system. Summative usability testing is quantitative; it focuses on obtaining measurements for the effectiveness, efficiency or satisfaction of a software system. A summative usability evaluation can be used to evaluate a design based on usability requirements so that the design’s acceptability can be established from the users’ point of view.

Both types of evaluation can be performed iteratively.

This syllabus discusses usability evaluation relating to software systems. Usability evaluation can also be applied to other products or services where usability is important, such as user guides, vending machines, aircraft cockpits, medical systems and train stations.

Usability evaluation addresses the direct interaction between users and the software system. The direct interaction occurs via a screen dialogue or other form of system use. Usability evaluation can be based on a software application, on design documents and on prototypes.

© International Software Testing Qualifications Board

The objectives of usability evaluation are:

- to assess whether usability requirements have been met (see section 1.3.1)
- to uncover usability defects so they can be corrected
- to measure the usability of a software system

Summative usability evaluation addresses the following:

- Effectiveness:
  - The extent to which correct and complete objectives are achieved by the user
  - Answers to the question: “Can the user reach its goal?”
- Efficiency:
  - Resource utilization to achieve specified user objectives
  - Answers to the question: “Can I execute my tasks in an efficient way?”
- Satisfaction:
  - Freedom from discomfort, and positive attitudes towards the use of the software system
  - Answers to the question: “Do I feel comfortable while using the software system?”

If users are involved, a usability evaluation can be carried out by performing usability testing, performing user surveys and performing usability reviews. If users are not present, usability reviews may still be performed (see section 4.2.2). If software will be used by individuals with specific disabilities, include them early in usability reviews (e.g., color blind users; (see [ISTQB_SYL_DAT]).

A qualitative usability evaluation enables identification and analysis of usability issues, focusing on understanding user needs, test objectives and reasons for the observed user behavior.

A quantitative usability evaluation focuses on obtaining measurements for the effectiveness, efficiency or satisfaction of a software system.

1.2.2 User Experience Evaluation

User experience describes a person's perceptions and responses resulting from the use or anticipated use of a software system.

User experience includes:

- Advertisements that make users aware of the software system
- Training in the use of the software system
- Touchpoints with the software system other than screen dialogue, such as encounters with support, letters or goods received as a result of interaction with the software system
- Issues that are not handled by the user interface of the software system, such as the notifications of delays, handling of complaints and unsolicited calls

User experience can be evaluated using the principal techniques outlined in the tables of section 1.2.

© International Software Testing Qualifications Board

1.3 Usability Evaluation and Human-Centered Design

Human-centered design activities and their interdependence, according to [ISO 9241-210], are shown in Figure 1.1.

Human centered design focuses on how people use software and applies human factors, ergonomics, and usability practices to make products more intuitive and effective. Figure 1.1 identifies human-centered design activities and general process flow.

Usability Requirements

Usability requirements provide the basis for evaluating a software product's ability to meet identified user needs.

It provides the basis for the evaluation of a software system to meet identified user needs. Usability requirements may have a variety of sources:

- They may be stated explicitly, such as in requirements documentation or a user story
- They may be implicit, undocumented user expectations (e.g., a user might implicitly expect that an application provides shortcut keys for particular user actions)
- They may be included in adopted or required standards (see Chapter 3)

Examples of usability requirements (in this case described as user stories) are:

1. “As a frequent user of the airline’s booking portal, an overview of my currently booked flights shall be automatically shown after I log on. This shall enable me to get an overview of my booked flights and make any updates.”
   This usability requirement is about the effectiveness component of usability.

2. “As a help-desk assistant, I must be able to enter and log the details of a customer request into the Customer Relations database in no more than two simple steps.
   This shall enable me to focus on the customer request and provide them with optimum support.”
   This usability requirement is about the efficiency component of usability.

3. “As a non-technical user, I need helpful messages to support my use of the airline’s electronic check-in application”
   Having support will encourage me to use the electronic check-in rather than a paper-based version.
   This usability requirement concerns requirements satisfaction

© International Software Testing Qualifications Board

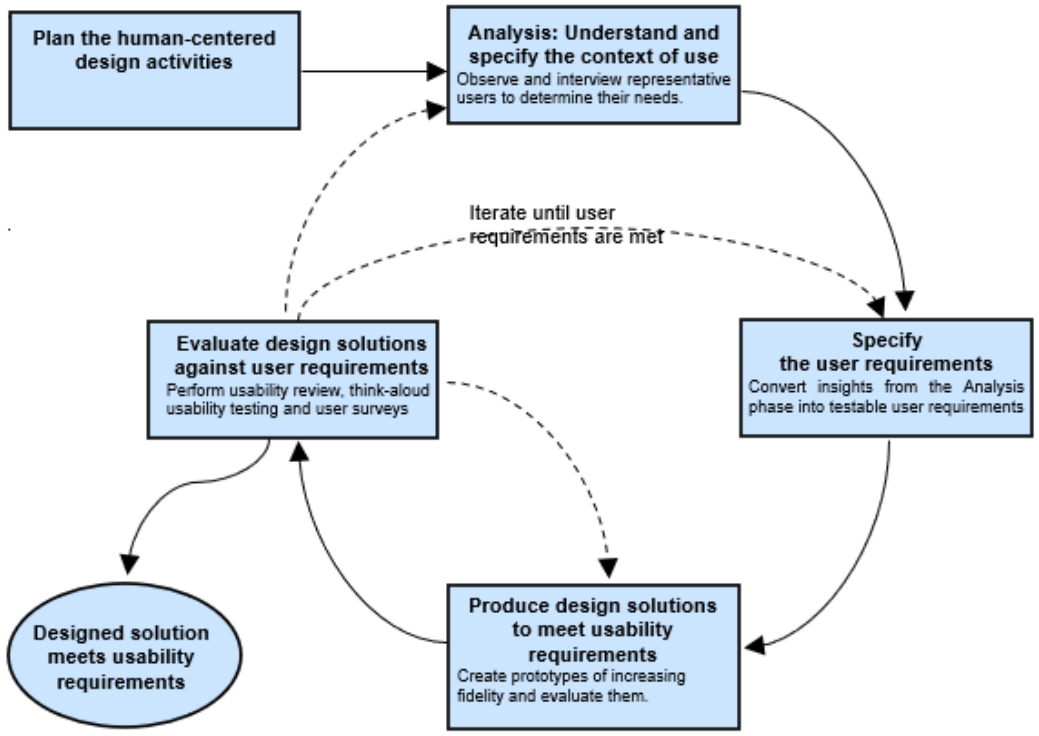

Figure 1.1 – Human-centered design activities and their interdependence [ISO 9241-210]

The human-centered design approach shown in Figure 1.1 can be summarized as follows:

- Analyze: Talk with people and discover "what is the problem?"
- Specify user requirements
- Design: Prototype what you assume is a solution
- Evaluate: Watch people use the prototype and learn from their experiences
- Iterate: Repeat until the usability requirements are achieved

The human-centered design activities are based on the following three key elements:

1. Users

Observe and interview users in their work environment. Users are involved throughout the design phase by discussing designs and alternatives with them directly (where possible), or with representative users.

2. Evaluation

Perform usability evaluation on the software system. A usability evaluation may take place at any time during human-centered design, from early analysis through software system delivery and beyond. A usability evaluation may be based on a prototype, as mentioned above, or on a completed software

© International Software Testing Qualifications Board

system. Usability evaluations that are performed in the design phase can be cost effective by finding usability defects early.

3. Iterations

Iterate between design and usability evaluation.

Considering the human-centered design approach shown in Figure 1.1, the most frequent iterations take place between the activities “Produce design solutions” and “Evaluate design solutions”. This generally involves the successive development of a prototype, which is a representation of all or part of a software system’s user interface. Although prototypes are limited in some way, they can be useful for usability evaluation. Prototypes may take the form of paper sketches or display mock-ups, as well as software systems under design. Starting with an initial prototype, the following activities are performed:

- The prototype is evaluated by the person who performs usability testing on the prototype.
- The prototype is improved and refined based on the results of the evaluation. The person who performs the prototype evaluation helps the developers evolve the prototype by incorporating user feedback into the design.

These activities are repeated until the usability requirements are achieved. When prototypes are developed in iterations, the steady refinement gives the user a more realistic impression of how the finished product will look and feel. Additionally, the risk of forgetting or ignoring usability issues is reduced.

Both usability and accessibility must be considered during the design phase. Usability testing often takes place during system integration and continues through system testing and into acceptance testing. Chapter 8 gives an overview of how to incorporate Usability Testing as a Tester in Agile Software Development.

© International Software Testing Qualifications Board

2 Risks in Usability and User Experience – 90 minutes

Keywords

product risk, project risk, user experience

Domain specific keywords

<none>

Learning Objectives for Chapter 2:

2.2 Typical Risks

UT–2.2.1 (K2) Give examples of typical usability and user experience risks

UT–2.2.2 (K3) Prepare a list of usability risks that are likely to occur for a given project scenario

UT–2.2.3 (K2) Give examples of typical project risks when testing usability and user experience

© International Software Testing Qualifications Board

2.1 Introduction

A risk is a factor that could result in future events with negative consequences; usually expressed as the impact (that is, harm resulting from the event), and the likelihood of the event happening.

Risk is calculated as the product of risk impact and risk likelihood. This calculation is used to decide which features or tasks need evaluation. Product risks can impact the overall quality of a software system. Section 2.2 provides examples of typical product risks relating to usability and user experience.

Project risks may impact a project's capability to deliver its usability objectives.

A risk assessment should be performed by identifying those features that most heavily relate to the product risks listed in sections 2.2.1 and 2.2.2. These risks should be tested using usability testing techniques, which are selected according to the criteria in Chapter 7.

A risk assessment should identify the project risks, such as those listed in section 2.2.3. These risks provide valuable information to help the project to meet its usability objectives.

Risks can be identified through one or more of the following techniques:

- Interviews
- Risk workshops
- Brainstorming
- Calling on past experience

Checklists such as those shown in section 2.2 support these techniques and help to focus on specific aspects of usability and user experience.

During the product risk assessment, various features that will be supported by the system under test are analyzed for usability risks. The assessment focuses on aspects such as numbers of users, type and background of users, usage frequency, risks (damage) when user tasks cannot be completed, criticality of user tasks for the business, and external visibility.

By involving the broadest possible sample of stakeholders, the risk identification process is more likely to identify the most significant usability risks. The participating stakeholders are typically domain experts, end users, expert usability testers, user-acceptance testers, usability professionals who are not experts, design team members and customer representatives.

2.2 Typical Risks

2.2.1 Usability Risks

The following list indicates typical product risks which impact users:

- Users won't buy or use the software system because it lacks effectiveness, efficiency or satisfaction.
- Users buy the software system, but return it and demand their money back because they can't make it work. The software system works as specified, but users can't figure out how to use it.
- Users buy the software system but repeatedly need to call support because they don't understand how to use it.
- Users buy the software system but find it difficult to use. They report their dissatisfaction to friends and social media, which causes a decrease in consumer trust and brand equity.

© International Software Testing Qualifications Board

- Users resist using a software system which is essential for their daily work because it lacks usability. If users are forced to use it productivity suffers.
- Increased liability through risk to human health or risk of financial loss caused by a poorly designed user interface.

2.2.2 User Experience Risks

User experience risks can result in a user experience failure in a software system or its support systems. The following user experience risks are typical:

Users cannot complete a purchase because the software system does not adequately support the buying task, for example it:

- does not perform some of the functions required by the users (lack of effectiveness)
- performs intended functions slowly or awkwardly
- is unpleasant to use or does not provide satisfaction

The software system is usable, but the associated work products and procedures are not. For example,

- Support answers reasonable customer inquiries slowly, rudely, superficially or not at all
- User documentation is deficient
- Delays in delivery of products ordered are not communicated
- The delivered product does not match the users' expectations
- The product arrives in an unattractive or impractical packaging

2.2.3 Project Risks

Organizational risks

- Lack of qualified specialists in usability and user experience
- Insufficient knowledge of basic usability principles for those responsible for designing and developing the product and its associated processes and work products
- Lack of knowledge, criteria and process for selecting qualified usability experts
- Low usability maturity in the organization (see section 7.1.1)
- Insufficient management attention for usability test and review test results, possibly because they are considered subjective
- Inappropriate decisions by management, such as not postponing a release even when usability test results clearly show a need for action
- Insufficient independence of quality assurance for usability testing processes and deliverables
- Unsuitable usability testers who exaggerate minor usability issues and impact the ability to collaborate with others.
- Poor or missing usability requirements
- Not addressing usability in the project assessment and/or test plan

Technical risks

- No usability evaluation performed, in particular no usability testing
- Usability evaluations are performed so late that there is no time to perform the required changes before the software system is implemented
- Usability evaluations not performed after the software system has been implemented and experienced users have become available

© International Software Testing Qualifications Board

- Problems in defining testable usability requirements for usability testing
- Non-availability of precise, written and reviewed procedures for usability evaluation
- Insufficient adherence to documented good practices for usability testing
- Late availability of a test environment required for usability evaluation
- Unrealistic test environment for usability evaluation: The sample data set is too small or inconsistent; there are no interruptions when in real use there are many interruptions
- Usability test report is unusable because it is too long, difficult to understand, or based on opinion
- Usability testing does not focus on the full user experience
- Sensitive topics, such as a defective underlying software architecture, which the organization tries not to change
- No clear statement of what constitutes a typical user system. For example, usability tests in organizations might use higher-end hardware that might mask performance defects
- Late and possibly unplanned repair costs threaten project cancellation or delay
- Limited number of qualified people available to conduct the planned tests

Supplier risks

- Suppliers do not have the required qualifications in usability testing
- Suppliers do not follow agreed guidelines for usability evaluation
- Usability evaluation results provided by suppliers are delivered late or not at all

© International Software Testing Qualifications Board

3 Usability Standards – 30 minutes

Keywords

<none>

Domain specific keywords

user interface

Learning Objectives for Chapter 3:

3.1 Usability Standards

UT–3.1.1 (K2) Explain the general content of user interface guidelines

UT–3.1.2 (K2) Explain the general content and applicability of ISO usability standards to particular types of project/application

Please note that only the information explicitly provided in this chapter is examinable and not the entire contents of the standards themselves.

© International Software Testing Qualifications Board

3.1 Usability Standards

A usability standard is typically a collection of user interface guidelines for the design of effective, efficient, and satisfactory user interaction. [ISO 9241-11] refers to the usability definitions and concepts mentioned in chapter 1

3.1.1 User Interface Guidelines

A user interface guideline is a low-level, specific rule or recommendation for user interface design. It leaves little room for interpretation so that designers can implement it consistently. A user interface guideline is often used to provide consistency in the appearance and behavior of the user interface of the systems produced by an organization.

User interface guidelines are helpful in detecting and reporting usability issues during all human-centered design activities. Without user interface guidelines, it may be difficult to decide on what is “unacceptable” usability. For example, guidelines may give advice on the maximum number of steps which a user may take in order to complete a simple function, such as a user login. User interface guidelines should also include precise rules relating to the size and placement of graphical user elements such as entry fields, buttons and list boxes. Guidance may also be given on the layout and format of error messages, screen layouts, use of colors and sounds, and any other factors that affect the user’s experience.

Examples of individual user interface guidelines:

- “For all controls, such as buttons, select the safest, most secure value by default to prevent loss of data or system access. If safety and security are not factors, select the most likely or convenient value.“
- “The company logo must appear in the upper left corner of each page. It must be positioned at exactly the same place as on the home page. Clicking the logo must cause the home page to be displayed.“
- “The height of a button must be twenty-three pixels.”

3.1.2 ISO Usability Standards

Several usability standards are issued by ISO, the International Organization for Standardization [Web-10], most of which belong to the ISO 9241 family of standards.

ISO 9241-110:2020 – Interaction Principles

This standard deals with the ergonomic design of interactive systems (software systems). It describes the following dialogue principles, each of which is a general goal for the design of effective and efficient dialogues.

1. Suitability for the user’s tasks,
2. Self-descriptiveness,
3. Conformity with user expectations,
4. Learnability,
5. Controllability,
6. Use error robustness
7. User engagement

© International Software Testing Qualifications Board

Dialogue principles resemble usability heuristics (see section 4.2.2) and are generally independent of any specific dialogue technique. They are applicable to the analysis, design and evaluation of software systems, although they can be difficult to apply in a usability evaluation because their general nature makes them subject to interpretation.

ISO 9241-210 – Human-centered design for interactive systems

This standard describes the principles of human-centered design and the related activities. Section 1.3 provides an overview of this standard.

ISO 25062:2025 – Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Common Industry Format (CIF) for reporting usability evaluations

This standard describes the Common Industry Format (CIF) for reporting usability evaluations. It provides a classification of evaluation approaches and the specifications for the content items in a usability evaluation report (content elements). The intended users of the usability evaluation reports are identified, as well as the situations in which the usability evaluation report can be applied.

A list of manufacturer guidelines is provided in Appendix E.

© International Software Testing Qualifications Board

4 Usability Reviews – 105 minutes

Keywords

<none>

Domain specific keywords

usability heuristic

Learning Objectives for Chapter 4:

4.1 Introduction and Approach

UT–4.1.1 (K2) Explain the usability review as a technique for evaluating usability and user experience

UT–4.1.2 (K2) Give examples of risks and challenges of usability reviews

4.2 Types of Usability Reviews

UT–4.2.1 (K2) Differentiate between informal usability and expert usability reviews

UT–4.2.2 (K3) Apply a given checklist for a given software application using heuristic evaluation

© International Software Testing Qualifications Board

4.1 Introduction and Approach

A usability review is a usability evaluation based on the judgment of one or more reviewers who examine or use a software system to identify potential usability issues and defects based on established criteria.

Usability reviews should be performed by expert usability testers, other usability professionals, subject matter experts, design team members, (acceptance) testers, customer representatives and potential users. Usability reviews are based on one or more of the following:

- Usability requirements
- Applicable user interface guidelines
- Usability standards
- Prior experience of usability issues encountered by users

Usability reviews are more effective when the user interface can be seen and used. For example, sample screen shots are usually easier to understand and interpret than a narrative description of the functionality provided by a particular screen. Visualization is important for an effective usability review of the documentation.

4.1.1 Step-By-Step Approach to Usability Reviews

A usability review has the following six steps [Nielsen94]:

Step 1: Prepare usability review

- Determine test objectives in cooperation with stakeholders.
- Select appropriate test technique(s): informal usability review, expert usability review, heuristic evaluation.
- Select reviewers (see list provided in the introduction above).

Step 2: Optionally, the author of the software application to be evaluated presents it to the reviewers. This should only be done to provide information, and care should be taken not to introduce a source of bias to the reviewers before they evaluate the software application.

Step 3: The reviewers evaluate the software application in solitude and write down their findings.

Each reviewer should start by considering:

- Who is using this software application?
- Why are the users using the software application?
- What goals do the users have?
- In what ways do the users see the software application with a different perspective? (e.g., are particular domain-specific aspects more important than technology-specific aspects)

The reviewers should then use the software application bearing the above questions in mind.

Step 4: Assuming that more than one reviewer is involved, the reviewers meet to reach consensus on the positive and negative aspects in their findings.

© International Software Testing Qualifications Board

- This consensus building is of particular importance. It removes any findings that are peculiar to a specific reviewer.
- Only findings that find consensus between the reviewers are included in the usability review report.

Step 5: The reviewers present the findings which have majority agreement to the author and interested stakeholders. This and the previous step may be combined.

Step 6: One of the reviewers writes a usability review report. This may be informal or formal depending on the organization's requirements. (see section 4.2).

4.1.2 Risks and Challenges

Usability reviews are opinion based and may result in a clash of views where the key arguments are opinions rather than based on data or facts. Such clashes can be a sign of an organization that does not have a profound understanding of usability testing, or inexperienced usability professionals.

The value of usability reviews may be diminished by the following risks:

- If candid discussion of the findings among reviewers does not take place, an incorrect finding suggested by just one reviewer may be accepted. It is crucial for the success of a usability review that reviewers are capable of rejecting questionable findings and recognizing good findings even if they are suggested by one reviewer only.
- If reviewers are not sufficiently familiar with the software system and its constraints, their findings may not be useful for development teams.
- If reviewers are not respected by fellow reviewers, their valuable findings may be too easily dismissed.
- If reviewers only focus on minor details like user interface guideline violations and graphic design, any serious issues concerning effectiveness and efficiency may remain undetected.

4.2 Types of Usability Review

The most important usability review techniques are described in this section:

- Informal usability reviews, which can be performed by anyone. (see section 4.2.1)
- Heuristic evaluations, which are preferably performed by expert usability testers and are supported by usability heuristics. (see section 4.2.2)

4.2.1 Informal Usability Reviews

An informal usability review is a usability review based on the judgment of one or more reviewers who examine or use a software system to identify potential usability issues. Informal usability reviews are often based on opinion, personal experience and common sense.

No formal usability qualifications are required for a reviewer to participate in this type of review. However, some usability experience or domain knowledge is helpful, and a valuable review can be carried out by a usability professional who has practical experience.

An expert usability review is an informal usability review in which the reviewers are experts. Generally speaking, an expert is a person who is recognized by independent peers as being an expert. They can be expert usability testers and/or subject matter experts.

© International Software Testing Qualifications Board

4.2.2 Heuristic Evaluation

Heuristic evaluation is a usability review in which one or more reviewers, preferably experts, compare the user interface of a software system to a list of usability heuristics and identify where the user interface does not follow those usability heuristics.

A usability heuristic is a generally recognized rule of thumb that helps to achieve usability. The purpose of a heuristic is to provide reliable and useful guidance to a reviewer during the usability evaluation of a software system.

The following criteria apply to usability heuristics:

- They are generally recognized as being valuable
- They are comprehensible for all reviewers

Usability heuristics are typically grouped into sets of about ten to provide ease of use and maintenance. The following set is widely recognized and was created by Jakob Nielsen [Nielsen94] [Web-9]:

	Name of Heuristic	**Description of Heuristic**
1	Visibility of system status	The system should always keep users informed about what is going on by providing appropriate feedback within reasonable time.
2	Match between system and the real world	The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
3	User control and freedom	Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support “undo” and “redo” functions.
4	Consistency and standards	Users should not need to consider whether different words, situations, or actions mean the same thing. Follow platform conventions.
5	Error prevention	Even better than providing good error messages is a careful design which prevents a defect from occurring in the first place. Either eliminate defect-prone conditions or test for them and present users with a confirmation option before they commit to the action.
6	Recognition rather than recall	Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
7	Flexibility and efficiency of use	Accelerators which are often not noticed by the novice user may speed up the interaction for the expert user such that the system can cater to

© International Software Testing Qualifications Board

		both inexperienced and experienced users. Allow users to tailor frequent actions.
8	Aesthetic and minimalist design	Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
9	Help users recognize, diagnose, and recover from errors	Error messages should be expressed in plain language (no codes), precisely indicate the error, and constructively suggest a solution.
10	Help and documentation	Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's test task, list concrete steps to be carried out, and not be too large.

Even though heuristic evaluation is a powerful approach, the reviewer should be aware of the following risks:

- Heuristic evaluation requires the reviewers to make judgments by comparing a software application to a set of usability heuristics. Usability issues are often complex and cannot always be represented in a set of usability heuristics.
- Usability heuristics need a context of use in order to judge whether they apply.
- It is relatively easy to apply an incorrect approach to heuristic evaluation. If the evaluation is based on intuition rather than on usability heuristics, then findings become assigned to one or more usability heuristics after the findings are complete. The correct approach is to let the usability heuristics drive the heuristic evaluation and to only report findings that can be directly attributed to one of the usability heuristics. If findings are reported that are not linked to a usability heuristic, they may still be valuable, (especially if found by an expert usability tester), but they are the product of an informal or expert usability review rather than a heuristic evaluation.
- Usability heuristics are designed to be compact, and interpreting them correctly requires some experience. Reviewers must fully understand the heuristics they will be applying before the heuristic evaluation commences. They must avoid the temptation to create their own set of heuristics, which may not meet the “recognized” and “comprehensible” criteria mentioned at the start of this section.

© International Software Testing Qualifications Board

5 Usability Testing – 345 minutes

Keywords

moderator, think aloud usability testing

Domain specific keywords

<none>

Learning Objectives for Chapter 5:

5.2 Step-By-Step Approach to Usability Testing

UT–5.2.1 (K2) Explain the principal steps in usability testing

5.3 Preparing a Usability Test

UT–5.3.1 (K2) Explain the content of a usability test plan

UT–5.3.2 (K2) Explain the content of a usability test script

UT–5.3.3 (K3) Prepare a simple usability test task for a given project

UT–5.3.4 (K2) Give examples of considerations to be applied when deciding on a location for usability tests

UT–5.3.5 (K2) Compare the advantages and disadvantages of a usability test lab

5.4 Performing a Usability Test Session

UT–5.4.1 (K2) Explain the key activities in a usability test session

5.5 Analyzing Findings

UT–5.5.1 (K2) Explain the procedure to apply for analyzing findings from a usability test

5.6 Communicating Test Results and Findings

UT–5.6.1 (K4) Analyze a usability test report for a given project

UT–5.6.2 (K2) Explain how to overcome internal resistance to usability test findings

UT–5.6.3 (K3) Apply a given list of best practices to report and communicate usability test results

5.7 Quality Control of a Usability Test

UT–5.7.1 (K2) Explain the quality control activities for a usability test

5.8 Challenges and Frequent Errors

UT–5.8.1 (K2) Give examples of the most frequent and serious errors in usability testing

© International Software Testing Qualifications Board

5.1 Introduction

Usability testing evaluates the degree to which the system can be used by targeted users with effectiveness, efficiency and satisfaction in a specified context of use.

Further details about usability testing are included in the individual sections of this chapter and in references [Molich07], [Barnum11], and [Hartson12].

For more information on specific adaptations to usability testing in agile see Chapter 8.

5.2 Step-By-Step Approach to Usability Testing

A usability test has the following three principal steps and associated tasks [Hartson12], [Molich07].

Expert usability testers and other software testers may apply these steps.

Step 1: Prepare usability test

- Create usability test plan
- Recruit usability test participants
- Write usability test script(s)
- Define usability test tasks
- Pilot usability test session

Step 2: Perform usability test sessions

- Prepare pre-session for usability test session
- Perform a briefing prior to the session to ensure the instructions to be followed are understood.
- Perform pre-session interview(s)
- Moderate the usability test session
- Perform post-session interview(s)

Step 3: Communicate test results and findings

- Analyze findings
- Write usability test report
- Sell findings (i.e., convince people)

A usability test usually consists of a series of usability test sessions. In each test session, a usability test participant performs representative test tasks on the software system or a prototype of the software system. A moderator leads the test participants in a usability test session and facilitates the test session for test participants.

Usability testing should be done under conditions which are as close as possible to those under which the software system will be used. This may involve setting up a mock up office or a living room. It should be possible to observe usability test sessions from a neighboring room or via video link so that stakeholders can observe the effect of the actual software system on real people without affecting the test participant's behavior.

© International Software Testing Qualifications Board

5.3 Preparing a Usability Test

5.3.1 Usability Test Plan

Preparations for a usability test are started by writing a usability test plan. The plan is a short description of the purpose and extent of a usability test and helps stakeholders or a client to decide on the usability testing to be performed.

The usability test plan [Barnum11] includes the following information:

- The goals of the usability test. These could include very short descriptions of the key tasks to be performed and the following typical test objectives:
  - To assess whether usability requirements have been met
  - To uncover usability defects so they can be corrected
  - To demonstrate convincingly to stakeholders that the usability of the software system is appropriate.

  Note that the first two objectives apply to any type of usability evaluation (i.e., usability review, usability testing, usability survey). The last objective, however, is unique to usability testing.
- The user group or groups for the usability test. The following people may attend a usability test session:
  - The moderator is a neutral person who facilitates a usability test session (and is therefore an expert usability tester). The moderator is the only person who is allowed to talk to the test participant during the usability test session.
  - The usability test participant is a representative user who solves typical tasks in a usability test.
  - The scribe records important usability findings and may be an expert usability tester.
  - Observers are generally stakeholders who have an interest in the software system or in the characteristics required to meet the user's needs and expectations. It is not essential that they attend a usability test. Examples of observers are users, support/help desk staff, trainers, documentation writers, developers, managers of developers, product management, product owners, designers, testers and marketing people.
- A reference to the usability requirements for the software system (if the purpose of the usability test is to evaluate the design solution against usability requirements)
- The principal test techniques that will be employed. A test technique that may be used for usability testing is think aloud usability testing. This enables test participants to share their thoughts with the moderator and observers while they solve usability test tasks. The advantage of this test technique is that it helps the moderator, scribe and the observers to understand the test participant's thoughts and vocabulary during usability testing.
- The number of planned test participants
- Schedule and approximate length of each usability test session
- The name of the moderator who is responsible for the usability test
- A resource and cost estimate for the usability test, including person hours and incentives
- Location where the usability test will be performed. The concept "usability testing" usually refers to a test where the usability test participant and the moderator are face-to-face in the same physical location. The following other forms of usability testing can be performed:
  - Remote usability testing is performed when the usability test participant and the moderator are in different physical locations and communicate using media such as instant messaging, webinar or video conferencing.

© International Software Testing Qualifications Board

    - Unmoderated usability testing, which is discussed in section 7.1.1
- How the findings will be communicated

- The tools to be used in the usability test (See Chapter 9).

The usability test plan must be short and to the point. Usually, one or two pages are sufficient. Stakeholders and management review the plan and any necessary modifications are made until it is considered acceptable.

5.3.2 Usability Test Script

A usability test script is documentation which specifies a sequence of actions for the execution of a usability test. It is used by the moderator to keep track of briefing and pre-session interview questions, usability test tasks, and post-session interview questions.

A usability test script is a checklist used by the moderator of a usability test. It contains the following information:

- Activities for preparing the usability test session before the test participant arrives
- Briefing instructions
- Pre-session interview questions.
- Usability test tasks
- Post-session interview questions

The sample usability test report provided at [Web-8] includes a sample usability test script.

5.3.3 Usability Test Tasks

A usability test task is a usability test execution activity specified by the moderator that needs to be accomplished by a usability test participant within a given period of time. Sufficient tasks are prepared in order to fill the time scheduled for the usability test session.

A good test task:

- Matches the test objectives of the usability test as defined in the usability test plan
- Is relevant from the test participant's point of view. Usability test tasks generally avoid requesting system-oriented tasks such as logging in, changing a password or locating the name of the webmaster. Instead, good test tasks are meaningful to the test participant, such as ordering a product.
- Is relevant from the stakeholders' point of view
- Has a consistent level of detail (high level, low level)

Usability test tasks must be consistent. A usability test task contains the following information:

- The precise phrasing of the usability test task scenario to be handed to the test participant
- Preconditions for the test task, including resources available to the test participant
- A justification for the importance of the test task, including what the test task is intended to evaluate
- Any data given to the test participant for solving the test task, for instance a delivery address, or information in the database when the test participant starts the test task

© International Software Testing Qualifications Board

- Criteria for test task completion or test task abandonment including the expected results. A sample criterion for test task abandonment is "If the test participant hasn't found an answer within 10 minutes, the test task is abandoned."

The following points are relevant to the sequence in which the usability test tasks are requested:

- Where possible, the first task should be simple so test participants experience a quick success. This is particularly important if a test participant appears stressed by the situation.
- Test tasks that are essential to the success of the software system are prioritized.
- Test tasks are requested in an order that seems logical from the test participant's point of view. For example, test participants are instructed to order something before they receive the test task to cancel an order.
- If possible, test tasks that depend on the successful completion of a previous test task are avoided
- Where possible, the last test task should also be simple to not let the test participant conclude with a negative and frustrating experience

The sample usability test report [Web-8] includes a sample set of usability test tasks.

5.3.4 Usability Test Location

The usability test location is the place where the usability test is performed.

Examples of usability test locations are:

- A dedicated usability test lab (see section 5.3.5)
- Two office rooms that are connected by a video link
- An office room
- A room at the place where the test participant lives or works
- A public place, such as a cafe. This location is most often chosen for quick usability test sessions lasting ten minutes or less.

An important part of a usability test location is that observers are able to discuss, debate, and express themselves freely without disturbing the conduct of the test. Observers should be able to come and go from the location as they please, but should not interfere with the test. Any regulations limiting the behavior of observers in an observation room are a sign that the usability test setup is unusable because it does not consider the needs of the primary users. The primary consumers of the usability test results are the observers.

It is technically possible to allow observers to witness test sessions from their workplace using screen sharing. This option is less desirable because an important part of observing a usability test session is the interaction and discussion with other observers (provided this is targeted to the usability test task and results).

5.3.5 Usability Test Lab

A usability test lab is two or more rooms that are specially equipped for usability testing (see Figure 5.1).

A usability test lab often consists of two rooms:

- A test room where the usability test participant sits

© International Software Testing Qualifications Board

- An observation room where observers and scribes can watch test participants as they solve usability test tasks

Often, the two rooms are separated by a one-way mirror which enables observers to watch the test participant but not vice versa. Monitors (M in Figure 5.1) in the observation room connected to cameras (C) in the test room enable observers to see the test participant's face and what happens on the test participant's screen and desktop. It is also common for the observers to be able to hear the user, but not vice versa.

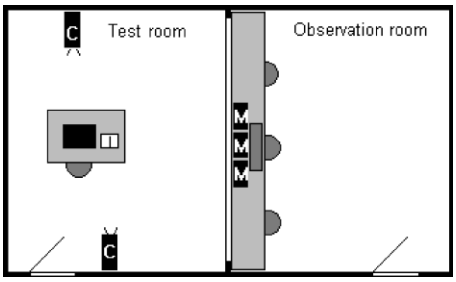

Figure 5.1 – Layout of Usability Test Lab

The following advantages are obtained by using a usability test lab:

- Observers can observe usability test sessions together
- Usability test sessions are easy to observe for scribes and observers
- Usability test sessions are performed under similar conditions
- Usability test sessions are easy to video record
- Observers are able to enter and leave during usability test sessions
- Management can physically demonstrate to visitors that the organization is deeply committed to usability

The following disadvantages apply to using a usability test lab:

- The context is artificial
- It is expensive to set up and maintain

Performing a Pilot Usability Test Session

A pilot usability test session is conducted to validate the usability test procedure.

Test tasks are defined that exercise the usability test script within the usability test setup. If serious defects in the usability test script are discovered in a pilot usability test session, the usability test script is modified and another pilot usability test session is performed.

Where possible, real users are chosen as test participants. If this is not possible, colleagues may be chosen instead. However, people who designed the software system should not be considered.

© International Software Testing Qualifications Board

Findings from pilot usability test sessions may be included in the usability test. For example, if the test participant in the pilot test session is over-qualified for the usability test and still encounters a serious usability in a valid usability test task, this may be considered as a usability defect.

5.4 Performing Usability Test Sessions

The key activities in a usability test session are described in the following table:

Activity	**Brief description of key activities**
Preparation of test session	The moderator prepares the hardware, the software system, test data and the test tasks for the test session before the test participant arrives.
Briefing – Pre-session instructions	The moderator informs the usability test participant about the purpose of the usability test and what their role and contribution are.
Pre-session interview	The usability test participant answers questions from the moderator about his/her background and previous experience with the software system or related software systems.
Test Execution	The usability test participant solves usability test tasks, which he/she receives from the moderator. While solving test tasks, the usability test participant is encouraged to think aloud. The moderator quietly observes the usability test participant during usability test task solution. The moderator guides the usability test participant if they get completely stuck, usually by moving on to the next test task.
Post-session interview	The usability test participant answers questions from the moderator about their experience and general impression of the usability of the software system. Key questions are “Which 2-3 things did you like most about the software system?” and “Which 2-3 things are most in need of improvement?”

5.5 Analyzing Findings

5.5.1 Procedure for analyzing findings from a usability test

Analysis of findings is the process that extracts findings from observations during usability test sessions.

The following steps are performed:

1. During the usability test session, the scribe records usability observations, usually by writing them down. Usability observations reflect events that result in issues or have a positive impact on effectiveness, efficiency and satisfaction.

© International Software Testing Qualifications Board

2. After each usability test session (while everyone still remembers what happened), the scribe and the moderator meet and have a candid discussion about the observations taken from the usability test session.
3. After all usability test sessions have been completed, the moderator and the scribe separately extract between twenty and thirty negative findings and between five and ten positive findings each from their observations. These findings reflect the observations that they consider most important.
4. The moderator and the scribe meet again and have a candid discussion about their findings. The findings are merged into a common list consisting of between twenty and thirty negative defects and between five and ten positive findings.
5. The moderator logs the defects in the common list on the organization's usability defect tracking tool, which ideally is the same as the defect management tool.
6. The moderator tracks the defects to resolution and reviews the implemented solution. If the implemented solution represents a risk, it should be subject to another usability test.
7. If another usability test is needed based on the number of changes made to the system under test, that test should be scheduled.

Several points are of particular relevance to the analysis of findings:

- Candid discussions between the moderator and the scribe (as mentioned in steps 2 and 4), are of essential importance to the success of the analysis. These discussions must be honest and based on observations rather than on personal opinions. Observers may be included in these candid discussions if they have any additional items that the scribe or moderator missed.
- It is important that a usability test report is usable, so the number of reported findings must be limited (the values shown in point 3 above are rules of thumb).
- It is a critical (and sometimes inconvenient) task for the moderator to prioritize the findings and report only the ones that are most important from a usability point of view.
- Comparisons of the findings between the current usability test and previous usability test reports will help detect trends and frequently occurring usability defects.

5.6 Communicating Test Results and Findings

5.6.1 Test Results

Usability test results are unique in one aspect: They show what representative users are able to accomplish with the software system when they carry out representative test tasks. Eliciting personal opinions from users, or discussing them, does not support this objective and should be left to other methods. Examples:

- Usability test tasks like "Is the design of the home page appropriate for the online CD Shop?" are opinion based and thus inappropriate for a usability test.
- Remarks like "I can do this easily but most others will have serious problems" from a usability test participant are personal opinions. The moderator may obtain additional, valuable insight by following up on this remark with the question: "What makes it easy for you?"

It is acceptable to report findings that are based on usability test participants' opinions about a software system, for example "The design of the home page is really pretty", but only if they are expressed spontaneously by a majority of the usability test participants.

© International Software Testing Qualifications Board

5.6.2 Usability Findings

A usability finding is a usability test result that identifies some important issue, or opportunity.

Positive usability findings are important for the following reasons:

- They make it easier to sell the need for correcting usability issues by giving a balanced view
- They communicate to the development team which features should not be modified or deleted
- It enables a complete view of usability to be obtained

A usability defect report should contain the same information as a regular defect report with the possible addition of direct quotes from the user. Defect severity classifications are used to indicate the consequences of the defect on the user experience.

The moderator and the scribe rate usability problems from the test participants' point of view. Sometimes, the severity ratings are allocated in cooperation with a domain expert.

Important parameters that influence severity ratings are:

- Frequency: How often does the usability defect occur?
- Impact: How badly does it hurt the user and the user's environment when the usability defect occurs?
- Persistence: How quickly will users learn to avoid the usability defect?

5.6.3 Selling Findings to Stakeholders

Stakeholders who are not usability professionals often need to be convinced about the need to take the issues from a usability test seriously and act on them. Some stakeholders are skeptical towards usability and usability tests.

The usability tester needs to “sell” usability findings in a convincing way to all stakeholders and must understand their motivation and focus. If stakeholders do not accept the findings from a usability test, the risk is that little beneficial change to the software system user interface will occur. For a general discussion about selling findings, see [Sharon12].

Knowledge and leadership from the organization’s management is important for the success of usability in an organization. The usability tester must understand the maturity of an organization regarding usability issues (see section 7.1). In organizations with low usability maturity, it may only be possible to make the consequences of ignoring usability visible by performing usability testing and then “selling” the findings in a convincing manner.

When techniques such as usability review and heuristic evaluation are employed, care must be taken to manage the exchange of opinions. Certain stakeholders may have strong opinions and are skilled at arguing their case, particularly if others are willing to hold back their own opinions. There is a risk that valid usability issues may be dismissed because opinions dominate the discussion and not facts.

To help mitigate the risks mentioned above, stakeholders must be involved in the planning and execution of a usability test. This allows them to “buy-in” to the usability test and makes usability issues easier to accept. The most important stakeholders are the people who decide which changes from the usability test should actually be implemented, and the people who do the actual implementation.

Some ways of involving stakeholders are:

- Involve them in the risk assessment

© International Software Testing Qualifications Board

- Invite them to participate in writing and reviewing the usability test plan, the usability test scripts and, in particular, the usability test tasks
- Invite them to participate in the recruitment process, in particular defining the usability test participant profile and creating the recruitment screener
- Invite and encourage them to observe usability tests
- Make it easy for them to observe usability test sessions:
    - Schedule usability test sessions at times that are convenient for stakeholders
    - Advertise usability test sessions widely, and indicate that observing only part of a usability test session is also acceptable
    - Carry out usability test sessions at locations that are convenient for stakeholders, such as where stakeholders work, instead of at a remote location
    - Make it easy for stakeholders to observe usability test sessions as a group
    - Watching and discussing can convince skeptical stakeholders of the benefits of usability testing
- Involve them in the reviewing of the usability test report

Usability Test Report

A usability test report is documentation that communicates the findings from a usability test. A usability test report is mandatory for a usability test and is generally written by the usability tester or the moderator. In projects applying Agile software development, verbal reporting can be used (see Chapter 8).

The purpose of the usability test report is to document and communicate the most important findings from a usability test. The test report must be effective and efficient for the key stakeholders, in particular the development team and managers who make decisions about what will be changed.

A usability test report contains the following sections [Barnum12]:

Chapter	Title	Description of Contents
1	Executive Summary	A short executive summary containing descriptions of the test objectives of the evaluation, test techniques(s) used, most important findings and general recommendations based on the findings
2	Findings and recommendations	See section 5.6.1
3	Test Objectives	Description of the test objective of evaluation
4	Purpose	Purpose of the evaluation, including listings of or references to relevant usability requirements
5	Evaluation technique	● Evaluation technique (i.e., how the usability test is performed) ● Description of the approach used, for example the type of evaluation performed and the experimental design of the evaluation ● Information regarding the physical and technical environment in which the usability test took place ● The usability test script

© International Software Testing Qualifications Board

Chapter	Title	Description of Contents
		● List of test participants (encrypted as required by the organization)
6	Contacts	Name and contact details of the moderator(s) and scribe(s) involved in the usability test

Good Practices in Usability Test Reporting

An important practice in all forms of usability evaluation is for all those involved to communicate in a positive and productive manner with the development team and the stakeholders. Equally important is that they communicate the findings accurately and objectively.

Many of the aspects discussed in section 5.6.3 about selling usability findings apply to test reporting.

The following table summarizes good practices in usability test reporting:

Good Practice Name	Good Practice Description
Involve and respect stakeholders	● Involve the stakeholders, in particular the development team, in writing the test report ● Consider their findings, even when they differ from the moderator or the scribe ● Incorporate their findings in the usability test report ● Treat stakeholders as partners in the common goal to improve usability ● Talk the language of the stakeholder and avoid usability jargon which some stakeholders may not fully understand
Make the main test report short and comprehensible	● Apply the following recommendations, regardless of the magnitude of the usability test: ○ A maximum of twenty-five findings reported ○ A maximum document length of twenty pages
Include a usable Executive summary	● Make the executive summary short: recommended one page ● Include only the most important findings ● Put the executive summary at the start of the usability test report where stakeholders can easily find it
Keep to the point	● Write crisp and precise descriptions of usability findings ● Avoid lengthy passages of text (e.g., an explanation of what usability is, transcripts of usability testing sessions and descriptions of the human-centered design approach). However, short notes of what a test participant said, such as quotes of one or two lines, are recommended.

© International Software Testing Qualifications Board

Good Practice Name	Good Practice Description
	• Include references to information if it supports the content of the usability test report
Positive and negative findings	• Include findings which reflect a positive aspect of usability. • Rate the severity of all negative findings
Aim for completeness	• Aim to include all sections of the usability test report (as listed in section 5.6)
Respect private or sensitive information	• Do not include details about recruitment, such as copies of the recruitment screener and the confirmation sent to usability test participants

The good practices described above are exemplified in a free, sample usability test report. [Web-8]

5.7 Quality Control of a Usability Test

To verify that a usability test has been performed properly, the following quality control tasks are performed, typically by the usability testing moderator or a test manager:

- Check that the usability test plan contains at least the information required by section 5.3.1
- Check that the usability test plan has been properly reviewed and approved by all relevant stakeholders
- Check that the usability test script (in particular the usability test tasks) matches the purpose of the usability test as described in the usability test plan
- Check that the profiles of the actual test participants match the purpose of the usability test as described in the usability test plan
- Observe the first two or three test sessions to determine that they are performed in accordance with the usability test script
- Compare the notes made by the moderator and the scribe from each test session with your own observations
- Raise a defect if important discrepancies or positive findings have not been noted
- If directed by the Project Leader or Test Manager, regularly compare the actual time plan and consumed resources with the estimates in the usability test plan
- Check that findings are communicated to the stakeholders in accordance with the usability test plan
- Check whether the usability test report conforms to the good practices described in LO 5.6.1
- Discuss the usability test report with some representative stakeholders to find out if it is comprehensible and meets their needs
- Attend discussions and presentations of the usability test results
- Check that communication is two-way and that constructive and professional discussions of the results take place

© International Software Testing Qualifications Board

5.8 Challenges and Frequent Errors

The ten most frequent and serious errors in usability evaluation, in particular usability testing, are:

Type of Error	Description
The purpose of the usability evaluation is not clear	The purpose should be described in the usability test plan and approved by the stakeholders before the usability evaluation starts. (see section 5.3.1)
Scheduling too late	The usability evaluation happens so late in the development process that there is no time to correct any usability defects before the software system is implemented. Note that this mistake is less likely to occur if an agile software development methodology is used, unless the sessions are scheduled at the end of each sprint. If the sessions are only during the hardening sprint, it's likely there will not be enough time available just like any other software development model.
Critical evaluation results ignored	Critical usability evaluation issues are overruled by management due to cost or time restrictions.
Incorrect focus	The usability evaluation focuses on minor details like user interface guideline violations and graphic design and does not address serious defects with effectiveness and efficiency.
No post-session interview	A short interview after a usability test session is important to summarize the test participant's impression from the usability test. (see section 5.4)
No engagement of stakeholders	The stakeholders should be engaged in various activities of the usability test (e.g., planning, reviewing), as noted in various sections of this chapter.
Less usable usability test report	Unusable test report. The usability test report is unusable, for example because it is too long, contains too many findings, or lacks a one-page executive summary
No positive findings reported	The usability test report contains no positive usability test findings. Positive findings are important to get acceptance of the test results. (see section 5.6.3)
Usability findings are not prioritized	The reader of the usability test report must be able to quickly distinguish between critical and minor usability defects. (see section 5.6)
Lack of management support	Management support is not available in terms of time and money to perform a good set of test sessions and resolve the defects when found

© International Software Testing Qualifications Board

© International Software Testing Qualifications Board

6 Usability Surveys – 30 minutes

Keywords

Software Usability Measurement Inventory (SUMI), System Usability Scale (SUS), user survey, Website Analysis and Measurement Inventory (WAMMI)

Domain specific keywords

<none>

Learning Objectives for Chapter 6:

6.1 Step-By-Step Approach to User Surveys

UT–6.1.1 (K2) Explain the purpose and approach adopted in user surveys

6.2 Standardized Questionnaires

UT–6.2.1 (K2) Compare the principal content and objectives of the public user questionnaires SUS, SUMI and WAMMI

© International Software Testing Qualifications Board

6.1 Step-By-Step Approach to User Surveys

A user survey is a usability evaluation whereby a representative sample of users are asked to report subjective evaluation into a questionnaire based on their experience in using a system. User surveys can be used to evaluate the levels of user satisfaction with a software system.

A user survey has the following steps [Wilson07]:

Step 1: Write a survey plan.

- The survey plan is similar to a usability test plan; it describes the test objectives of the user survey, the required resources, and a schedule
- Stakeholders are involved in reviewing and approving the survey plan

Step 2: Interview users and stakeholders

- Obtain feedback on issues that the user survey should focus on
- Focus on recent, vivid experience

Step 3: Select a questionnaire

- A questionnaire is a set of questions that is used to collect data from users
- A decision is taken between using a standard usability questionnaire or developing one. (Note that the development of a dedicated questionnaire is not considered in this syllabus)

Step 4: Deploy the questionnaire.

- Recruit user survey participants
- Publish the user survey
- Follow up with a reminder to user survey participants who have not responded within a reasonable time, if possible

Step 5: Analyze the responses from the user survey and link to the set test objectives

Step 6: Communicate the user survey results

- Write a report that answers any questions posed by stakeholders in a useful and usable way and is based on the user survey results
- Present the user survey results

6.2 Standardized Questionnaires

Standardized questionnaires are preferred over self-invented questions. Three commonly used, standardized questionnaires are briefly described below. Each questionnaire can be used to measure usability and/or user experience from the end user's point of view.

SUS – System Usability Scale

SUS is a simple, ten-item attitude scale giving a global view of subjective assessments of usability. SUS has become widely used, with references in over 1300 articles and publications [Brooke96]. When a SUS is used, participants are asked to score 10 items with one of five responses that range from Strongly Agree to Strongly Disagree.

Sample SUS statements are:

- I think that I would like to use this system frequently.
- I found the system unnecessarily complex.
- I thought the system was easy to use.

© International Software Testing Qualifications Board

SUMI – Software Usability Measurement Inventory

SUMI is a questionnaire with fifty statements to which the user has to reply that they either Agree, Don't Know, or Disagree. [Web-5]

Sample statements are:

- This software responds too slowly to inputs.
- The instructions and prompts are helpful.

WAMMI – Website Analysis and Measurement Inventory

WAMMI is a questionnaire with twenty statements which focuses on the usability of web sites from the end user's point of view. WAMMI questions have been selected from a large range of questions about user experiences with websites. The questions focus on aspects of user interface aesthetics, controllability, affect, efficiency, helpfulness, and learnability. WAMMI has five responses that range from Strongly agree to Strongly disagree. [Web-6]

SUS, SUMI and WAMMI permit benchmarking against a database of previous usability measurements. They also provide concrete measurements of usability which can be used as completion or acceptance criteria.

© International Software Testing Qualifications Board

7 Selecting Appropriate Test Techniques for Usability – 60 minutes

Keywords

user experience

Domain specific keywords

<none>

Learning Objectives for Chapter 7:

7.1 Criteria for Selecting a Usability Evaluation Technique

UT–7.1.1 (K3) Apply the appropriate test techniques for verifying and validating usability and user experience in a given project

© International Software Testing Qualifications Board

7.1 Criteria for Selecting an Evaluation Technique

7.1.1 Selecting a Usability or User Experience Evaluation Technique

Important criteria for selecting a usability evaluation technique are:

1. The purpose of the usability evaluation

- To evaluate effectiveness, use usability testing or usability review
- To evaluate whether efficiency requirements have been fulfilled, or whether the usability of a product has measurably improved, use a quantitative usability test
- To evaluate whether satisfaction requirements have been fulfilled, use a user survey
- To increase usability awareness or usability maturity in the organization, run a usability test to help convince skeptical stakeholders

2. Usability maturity of the organization

Usability maturity is the capability of an organization with respect to the effectiveness and efficiency of its usability processes and usability work practices. Usability maturity can be categorized as follows:

- High: The organization has implemented a human-centered design approach, with multiple activities and milestones
- Medium: Quality, time, and resource requirements for the human-centered design approach are known and controlled
- Low: Individuals adhere to human-centered processes, but often in an unsystematic way
- Very Low: Indifference or outright hostility towards usability
- Usability testing should be favored over usability reviews if the usability maturity of the organization is low or very low. In such organizations, controversial test results from a usability review could be dismissed as just opinions.

3. Completion time

Usability reviews are fastest with respect to elapsed time and can often be completed within a few days. Unmoderated usability tests, (see bullet 5 below), can also be completed within a few days.

4. Project phase

If the project is in the early design phase where only prototypes are available for evaluation, use a formative evaluation such as a usability review, a quantitative usability evaluation or a user survey.

5. Resources

If resources are limited (e.g., time, money and qualified usability experts), consider the following options:

- Perform a short form of usability test. This is particularly relevant in agile software development (see chapter 8) and has one or more of the following characteristics:
  - The moderator also takes notes; there is no separate scribe (see Chapter 8)
  - No usability test lab is used; the test sessions take place, for example, in a meeting room
  - The usability test report is short and lists a limited number of findings, for example 15

- Unmoderated usability testing

© International Software Testing Qualifications Board

With this form of usability testing the test participants solve usability test tasks without being observed by a moderator. Typically, the actions of test participants may be video recorded for later analysis. Other forms of data gathering may also be used such as event logging, screen recording, performance measures, and questionnaires. The main advantages of an unmoderated usability test are that the recruitment process is fast and the test is cheaper to perform. The analysis effort is the same as for other types of usability testing covered in this syllabus.

- RITE – Rapid Iterative Testing and Evaluation
  This form of usability testing involves making changes to the user interface as soon as a usability defect is identified and a solution is clear. Changes can occur after observing as few as one test participant.

6. Availability of end users

If representative end users are not available or require too many resources to recruit or compensate, usability reviews are better than usability testing.

7. Type of software development lifecycle model:

All test techniques described in this syllabus can be used with a sequential development model, such as the V-Model. The description of the Human-Centered Design approach in section 1.3 is based on the sequential development model, but the principles of Human-Centered Design apply universally for other types of software development lifecycle models.

In agile software development, design teams work in short development iterations—often two weeks in length—as fast delivery of usability evaluation results are an important part of defining test techniques. As mentioned in chapter 8, test techniques such as RITE and short forms of usability testing work well in agile software development. Please refer to [ISTQB CTAL-AT] for further details on agile software development.

It is highly recommended to mix several recognized usability evaluation techniques in one project. These should suit the phase in the software development lifecycle so that, for example, usability reviews are performed early on and usability testing later. Other examples of mixed test techniques are:

- An informal usability test of a mobile application in a cafe with five or six test sessions each lasting approximately ten minutes (also called Guerilla usability testing) followed by more formal in-depth usability test sessions
- A quick usability review that uncovers the most serious usability defects so they can be eliminated before an expensive usability test

© International Software Testing Qualifications Board

8 Usability Testing in Agile Software Development – 60 minutes

Keywords

User experience, Guerilla testing

Domain specific keywords

Persona

Learning Objectives for Chapter 8:

8.1 Integrating Usability Testing into Agile Software Development

UT–8.1.1 (K2) Explain the usability evaluation approaches in Agile software development

8.2 Roles in Agile Software Development

UT–8.2.1 (K2) Explain the role of a Usability Tester in the context of Agile software development

8.3 Usability Evaluation Approaches Suitable in Agile Software Development.

UT–8.3.1 (K2) Distinguish the different approaches to Usability Testing suitable in Agile software development

8.4 Shifting the Focus to User Experience

UT–8.4.1 (K2) Differentiate between techniques to shift the focus from functionality to user experience

© International Software Testing Qualifications Board

8.1 Integrating Usability Testing into Agile Software Development

In this chapter, aspects of usability testing are introduced which are of relevance to projects applying Agile software development. In particular, the influence of Agile software development on the planning, executing and reporting of usability tests is introduced (see [ISTQB_CTAL-AT] for further details).

Planning Usability Tests

- In Agile software development, teams work in short iterations (sprints). The scope of usability testing considers both the fine-grained features in an iteration and the larger scenarios which may span across multiple iterations.
- Planning is performed by the whole team, including usability testers and User Experience (UX) specialists (see section 8.4.2)
- Usability tests must take into account the different competencies required, as described in Section 8.2. The successful interaction of these roles is critically important in planning and executing the usability tests.
- The test basis may focus on the individual features to be implemented in an iteration or on the broader objectives and scenarios of the application.
- Usability testing is not continuous. Formal usability test sessions may occur in every iteration according to business needs. At a minimum, lightweight usability tests should be performed whenever user-facing changes have been made.

Designing usability tests

- When designing usability tests, a UX specialist contributes to the competency available to the usability tester (see section 8.2 below).
- Guerilla usability testing (e.g., in a café) can be useful in Agile software development because it can be performed quickly and has low cost. Selecting this method depends on risk and the availability of users to participate.
- Capture findings in a lean way (e.g., rainbow spreadsheet (use stop-sign colors to indicate test result for fast pattern spotting across test tasks & users), defect reports, sticky notes (physical or digital), real-time tagging (use quick tags like confusion, success, error, quote)).
- Afterwards you can also ask the test participants in a survey what they think about the application and if they would recommend it to someone else.

Reporting of usability tests

- Reporting of test results in Agile software development aligns with the scope of testing. Small feature-based tests may be considered verbally in a planning or review meeting. Larger, scenario-based tests may be reported more formally, depending on the criticality of the scenario (i.e., critical scenarios are likely to demand more details of coverage and any usability defects found).

© International Software Testing Qualifications Board

8.2 Roles in Agile Software Development

A usability tester is a tester specialized in usability testing. Ideally, the usability tester is part of a cross-functional Agile team, but in practice, that is often not the case. Instead, a UX specialist usually takes responsibility for usability and the overall user experience, guiding the process from design through development. In Agile software development, design and development often happen simultaneously, which makes close collaboration between developers, testers, and designers essential. This helps to verify that the final product is not only functional, but also intuitive and user-friendly.

In some cases, especially with low-code platforms, a UX specialist may not be on the Agile team. In those situations, the tester can play a crucial role in making sure usability and user experience are not overlooked. When using Agile software development, representative users are typically the product owners. They are an integral part of the development team and provide frequent feedback to designers and developers on usability defects. The following table summarizes the roles which are generally found:

Role	**Responsibilities**
Developer	component and integration testing
Tester	Functional testing, non-functional testing, test automation
UX specialist	Design, Usability testing
Product owner	Represent business and user
Usability tester	Test the usability of product in all phases of the software development lifecycle

Product owners may act as user proxies when real users are unavailable. If this is the case, the team must validate that they can be considered as representative of real users. Priority should be given to real users.

8.3 Usability Evaluation Approaches Suitable in Agile Software Development

The following usability evaluation approaches work well with Agile software development:

Rapid Iterative Testing and Evaluation (RITE)

- RITE is a qualitative usability test approach where changes to the user interface are made as soon as a usability defect is identified and a solution is clear. The RITE approach focuses on instant redesign to fix defects and then confirm that the solution works with new test participants (i.e., real users or representative users). Changes can occur after observing as few as one test participant.

© International Software Testing Qualifications Board

Once the data for a test participant has been collected, the usability tester and the stakeholders decide if any changes are needed prior to the next test participant. The modified user interface is then tested with the remaining test participants. [Medlock02]

Usability reviews (see Chapter 4)

- Guerilla usability testing (with usability test participants)
- Usability review/inspection (heuristic evaluation, walkthrough)

Formative evaluation

- Of the two types of evaluation mentioned in section 1.2.1 (formative and summative), the formative (or “exploratory”) type is more likely to be favored in Agile software development. Formative evaluation is often repeated during the design and prototyping phases to get ideas and to guide (or “form”) the design by identifying usability design defects.

8.4 Shifting the Focus to User Experience

This section examines two approaches to strengthening the user experience and promoting a human-centered design approach.

8.4.1 Integrating personas in user stories

In Agile software development, it is essential to keep the end-user in mind at every step. It is easy to become absorbed in technical details and lose sight of what truly matters: creating a product that works well and meets the needs of the target audience. One effective way to stay user-focused is by using personas. These are fictional characters that represent different user types, based on research (ISO /IEC 26515:2018).

UX specialists use personas to bring real user needs and behaviors into the design and development process. For example, *Impatient Ivan* represents a user who wants to complete test tasks quickly and efficiently, while *Anxious Andy* might describe someone who lacks technical confidence and fears making errors. *Corporate Claire* could represent a power user familiar with advanced features and shortcuts. Personas can also help with accessibility testing (such as *“Screen Reader User”, "Low-Contrast Environment User“) to* provide inclusive design.

However, simply creating personas is not enough. In the fast pace of Agile iterations, they can be easily overlooked. To keep personas in focus, they should be integrated into user stories. For example: “As *Anxious Andy*, I want clear instructions, so I feel confident completing the task.” This approach embeds user needs directly into acceptance criteria and helps the team stay aligned with the people they are intended for.

8.4.2 Close collaboration in the cross-functional team

© International Software Testing Qualifications Board

In Agile software development, usability testing plays a vital role in ensuring the product meets real user needs, not just technical requirements. A shared understanding of what the product should deliver is essential in a cross-functional team and supports everyone in the team (including product owners, developers, testers, UX designers, and stakeholders) so that they can have a clear, aligned view of what the product should do and how it should behave for end users.

This cross-functional team approach aligns with the principles of Behavior-Driven Development (BDD) as detailed in the ISTQB Specialist Syllabus on Acceptance Testing [ISTQB_SYL_AcT]. This approach is an alternative to relying solely on documentation. It supports the team in holding conversations when defining requirements and scope, enabling them to speak a common language. In line with BDD, real-world examples are expressed through user stories and acceptance criteria to create a shared understanding and agreement within the cross-functional team."

BDD promotes collaboration through practices such as the Three Amigos, in which a developer, tester, and product owner (or UX specialist) discuss user stories together to clarify intent and expected behavior.

To embed usability into agile software development, UX tests should be included in every iteration. These tests can be carried out by the tester and, when available, in collaboration with a UX specialist designer or representative users such as the product owner.

Integrating usability considerations into test planning and test execution enables the team to validate not only whether the product functions, but also whether it serves the user effectively. Close collaboration within the team on UX keeps user experience in central focus throughout development. It does not become an afterthought. This is especially important during planning and formulation, where testers and UX designers (or representative users such as the product owner) can work together to incorporate UX checks into each user story and embed persona-based examples and accessibility requirements directly into BDD scenarios.

© International Software Testing Qualifications Board

9 Select and Use Tools for Usability Testing - 45 minutes

Keywords

<none>

Domain specific keywords

<none>

Learning Objectives for Chapter 9:

9.1 Categories of Usability Testing Tools

UT–9.1.1 (K2) Classify the categories of usability testing tools

9.2 Application of Usability Testing Tools

UT–9.2.1 (K2) Explain the general application of usability testing tools

9.3 Limitations of Usability Testing Tools

UT–9.3.1 (K2) Explain the limitations of tools for usability testing

9.1 Categories of Usability Testing Tools

Usability testing tools can be classified according to the role they play in gathering usability information. The following broad categories may be distinguished:

1. Prototyping tools: Used to create low- or high-fidelity representations of a product for early testing. These range from simple sketches to interactive clickable prototypes.
2. Data capture tools: Record user behaviors and reactions, including screen interactions and facial expressions.
3. Observation and collaboration tools: Facilitate real-time observation for distributed stakeholders to watch and take notes.
4. Eye-tracking tools: enable heat maps to be created which show where on the screen users look most.
5. Analysis and evaluation tools: Support the structuring, interpretation, and reporting of data, for example through categorization frameworks or heuristic checklists.

© International Software Testing Qualifications Board

6. Generative AI (GenAI) tools support heuristic evaluation, walkthroughs, test case creation, and pattern recognition to automatically identify potential usability issues or predict user difficulties (See Chapter 10).

Note that the above list of categories is illustrative and not exhaustive.

9.2 Application of Usability Testing Tools

Each category of tool enables usability testers to focus on different aspects of the user experience.

1. Prototyping tools allow researchers to validate ideas early in the design process. Low-fidelity versions are especially effective for exploring broad concepts, while high-fidelity clickable prototypes provide insights into navigation, interaction flow, and test task performance.

2. Data capture tools generate detailed records of user interaction, from cursor movement to verbal responses. These recordings can be analyzed post-session to identify behavioral patterns and emotional cues.

3. Observation and collaboration tools broaden participation in usability testing by making sessions accessible to remote stakeholders. Shared scribing capabilities encourage multiple perspectives and can lead to more comprehensive findings.

4. Eye-tracking tools enable heat maps to be created which show where on the screen users look most. Note that this category covers both hardware eye-tracking and AI-predicted attention.

5. Analysis and evaluation tools facilitate systematic assessment. In heuristic evaluation, for instance, digital checklists and annotation interfaces help experts apply established usability principles consistently, record violations, and compile structured reports.

6. Generative AI (GenAI) tools can automatically review interface designs or prototypes and flag areas that may present usability challenges. These tools can act as an early filter before manual testing, offering predictive insights that help prioritize what to evaluate further with real users.

9.3 Limitations of Usability Testing Tools

Although usability testing tools increase efficiency and reach, they also introduce certain constraints.

- Fidelity and perception in prototyping: High-fidelity prototypes can be mistaken for finished products, which may discourage participants from offering constructive criticism. Low-fidelity versions, on the other hand, may lack sufficient realism to elicit meaningful feedback.
- Technical complexity in data capture: Recording technologies can add logistical burdens

© International Software Testing Qualifications Board

- Bias effects: participants may alter their behavior if they are overly conscious of being observed. Over-reliance on structured frameworks: While heuristic evaluation tools improve consistency, they cannot guarantee the identification of all usability issues. Expert judgment remains essential, and findings are limited by the chosen framework.
- Constraints of AI-assisted tools: Artificial Intelligence can highlight potential design defects, but it cannot fully account for context, culture, or user goals. Its predictions are based on generalized patterns and may overlook issues that only emerge in authentic use. AI outputs require human validation and interpretation.
- Partial capture of user experience: Across all categories, tools can record behaviors, generate metrics, or suggest improvements, but they cannot independently uncover the underlying reasons for user struggles. Meaningful insights continue to depend on the researcher's methodological rigor and interpretive skills.
- Risks to confidentiality arising from the use of recording (e.g., eye-tracking, analytics)

© International Software Testing Qualifications Board

10 AI Support for Usability Testing - 15 minutes

Keywords

<none>

Domain specific keywords

<none>

Learning Objectives for Chapter 10:

10.1 Areas in which AI Can Support Usability Testing

UT–10.1.1 (K2) Explain the areas in which AI can support usability testing

© International Software Testing Qualifications Board

10.1 Areas in Which AI Can Support Usability Testing

Artificial Intelligence (AI) can be considered as a tool which supports several different aspects, some of which are listed in the summary table below.

Aspect	**Description**
Synthetic users / AI personas	Artificial Intelligence (AI) can assume a role (e.g., an inexperienced user) to suggest defects with applications, such as undefined or confusing functionality. AI can help form usability hypotheses, but any usability defects found need human review and, where relevant, validation with representative users.
Pattern recognition	AI can help identify clusters and patterns in information gathered in reviews, including reviews such as those based on Nielsen's usability heuristics. (see 4.2.2 Heuristic Evaluation) AI can help analyze information gathered in usability testing (see section 5.5 Analyze Findings)
UI/UX Design and prototyping	AI-powered design tools can generate complete user interface designs and front-end code from text or image inputs. Design variants can be generated for making comparisons in usability testing.
Performing usability test sessions.	Using synthetic AI users for performing early testing
Usability results	Automatic transcription of session recordings
Usability analysis	Analysis of sentiment and emotions

The current versions of the following ISTQB specialist syllabi may also support usability testing:

- [ISTQB_SYL_AI_GEN] Generative AI
- [ISTQB_SYL_AI_TEST] AI Testing

© International Software Testing Qualifications Board

11 References

11.1 Standards

The following standards are mentioned in these respective chapters:

- ISO 9241-11:2018 – Usability – Definitions and concepts
  Chapter 3
- ISO 9241-110:2020 – Interaction principles
  Chapter 3
- ISO 9241-210:2019 – Human-centered design for interactive systems
  Chapter 1, Chapter 3
- ISO 25062:2025 – Definition of a Common Industry Format for reporting usability evaluations.
  Chapter 3
- ISO/IEC/IEEE 26515:2018: Systems and Software Engineering – Developing information for users in an agile environment
  Chapter 8
- ISO/IEC 25063:2014 Systems and software engineering--Systems and software product Quality Requirements and Evaluation (SQuaRE) Common Industry Format (CIF) for usability: Context of use description)
- Appendix D

11.2 ISTQB Documents

- [ISTQB_CTAL-AT] ISTQB Agile Tester, Version v2.0, 2026
- [ISTQB_FL_SYL] ISTQB Foundation Level Syllabus, Version v4.0.1, 2024
- [ISTQB_GLOSSARY] ISTQB Glossary of Terms used in Software Testing, Version 4.7.2, 2026
- [ISTQB_SYL_AcT] ISTQB Specialist Syllabus, Acceptance Testing, Version 2019
- [ISTQB_SYL_AI_GEN] ISTQB Specialist Syllabus, Generative AI, Version 1.1, 2026
- [ISTQB_SYL_AI_TEST] ISTQB Specialist Syllabus, AI Testing, Version 2.0, 2026
- [ISTQB_SYL_DAT] ISTQB Specialist Syllabus, Digital Accessibility Testing, Version 1.0, 2026

11.3 Books

[Barnum11] Carol M. Barnum, “Usability Testing Essentials”, 2011, Elsevier, ISBN 978-0-12-375092-1
[Brooke96] John Brooke, “SUS – A ‘Quick and Dirty’ Usability Scale”, in Patrick W. Jordan, Bruce Thomas, Bernard A. Weerdmeester, Ian L. McClelland (eds.) “usability evaluation in industry”, Taylor & Francis, 1996, ISBN 0-7484-0314-0
[Hartson12] Rex Hartson, Pardha S. Pyla, “The UX Book”, 2012, Morgan Kaufman, ISBN 978-0123852410

© International Software Testing Qualifications Board

[Medlock02] Michael C. Medlock, Dennis Wixon, Mark Terrano, Ramon L. Romero, Bill Fulton, “Using the RITE method to improve products: A definition and a case study”, 2002, Usability Professionals Association 2002 Conference, Orlando Florida.
[Molich07] Rolf Molich, “Usable Web Design”, 2007, ISBN 978-87-571-2526-9
[Nielsen94] Jakob Nielsen “Heuristic Evaluation”, in Jakob Nielsen, Robert L. Mack (eds.) “Usability Inspection Methods”, John Wiley & Sons, 1994, ISBN 0-471-01877-5
[Sharon12] Tomer Sharon, “It’s Our Research: Getting Stakeholder Buy-in for User Experience Research Projects”, 2012, Morgan Kaufman, ISBN 978-0123851307
[Wilson07] Chauncey Wilson, “Designing Useful and Usable Questionnaires: You Can’t Just “Throw a Questionnaire Together”, 2007, interactions, May+June 2007

11.4 Other References

The following references point to information available on the Internet. Even though these references were checked at the time of publication of this syllabus, the ISTQB cannot be held responsible if the references are no longer available. Please use the information shown below to search for latest versions.

[Web-1] Apple OS X Human Interface Guidelines
[Web-2] Android User Interface Guidelines
[Web-3] Microsoft Windows User Experience Interaction Guidelines,
[Web-4] SAP design guidelines and resources
[Web-5] SUMI - Software Usability Measurement Inventory
[Web-6] WAMMI – Web Analytics Service
[Web-7] Involving Users in Evaluating Web Accessibility, http://www.w3.org/WAI/eval/users.html
[Web-8] UXQB Sample Usability Test Report, http://uxqb.org/en/documents/
[Web-9] Jakob Nielsen’s 10 heuristics, http://www.nngroup.com/articles/ten-usability-heuristics/
[Web-10] International Organization for Standardization (ISO) http://www.iso.org

Chapter 1: [Web-7]
Chapter 3: [Web-1], [Web-2], [Web-3], [Web-4], [Web-10]
Chapter 4: [Web-9]
Chapter 5: [Web-8]
Chapter 6: [Web-5], [Web-6]

© International Software Testing Qualifications Board

12 Appendix A – Learning Objectives/Cognitive Level of Knowledge

The specific learning objectives applying to this syllabus are shown at the beginning of each chapter. Each topic in the syllabus will be examined according to the learning objective for it.

The learning objectives begin with an action verb corresponding to its cognitive level of knowledge as listed below.

Level 1: Remember (K1)

The candidate will remember, recognize and recall a term or concept.

Action verbs: Recall, recognize.

Examples
Recall the concepts of the test pyramid.
Recognize the typical objectives of testing.

Level 2: Understand (K2)

The candidate can select the reasons or explanations for statements related to the topic, and can summarize, compare, classify and give examples for the testing concept.

Action verbs: Classify, compare, differentiate, distinguish, explain, give examples, interpret, summarize

Examples	**Notes**
Classify test tools according to their purpose and the test activities they support.	
Compare the different test levels.	Can be used to look for similarities, differences or both.
Differentiate testing from debugging.	Looks for differences between concepts.
Distinguish between project and product risks.	Allows two (or more) concepts to be separately classified.
Explain the impact of context on the test process.	
Give examples of why testing is necessary.	

© International Software Testing Qualifications Board

Examples	Notes
Infer the root cause of defects from a given profile of failures.	
Summarize the activities of the work product review process.	

Level 3: Apply (K3)

The candidate can carry out a procedure when confronted with a familiar task, or select the correct procedure and apply it to a given context.

Action verbs: Apply, implement, prepare, use

Examples	Notes
Apply boundary value analysis to derive test cases from given requirements.	Should refer to a procedure / technique / process etc.
Implement metrics collection methods to support technical and management requirements.	
Prepare installability tests for mobile apps.	
Use traceability to monitor test progress for completeness and consistency with the test objectives, test strategy, and test plan.	Could be used in a LO that wants the candidate to be able to use a technique or procedure. Similar to 'apply'.

Level 4: Analyze (K4)

The candidate can separate information related to a procedure or technique into its constituent parts for better understanding and can distinguish between facts and inferences. Typical application is to analyze a document, software or project situation and propose appropriate actions to solve a problem or task.

Action verbs: Analyze, deconstruct, outline, prioritize, select.

Examples	Notes
Analyze a given project situation to determine which black-box or experience-based test techniques should be applied to achieve specific goals.	Examinable only in combination with a measurable goal of the analysis. Should be of form 'Analyze xxxx to xxxx' (or similar).
Prioritize test cases in a given test suite for execution based on the related product risks.	

© International Software Testing Qualifications Board

Examples	Notes
Select the appropriate test levels and test types to verify a given set of requirements.	Needed where the selection requires analysis.

© International Software Testing Qualifications Board

13 Appendix B – Business Outcomes Traceability Matrix with Learning Objectives

This section lists the traceability between the Business Outcomes and the Learning Objectives of Usability Testing syllabus

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
UT-BO1	Understand the basic concepts of usability and usability testing.		5 LOs								
UT-BO2	Identify and classify the severity of usability risks in a given product at any phase of a development cycle.			3 LOs							
UT-BO3	Cite relevant standards for usability and user experience and verify their implementation in a given product.				2 LOs						
UT-BO4	Set up procedures so that stated usability and user experience goals may be verified in practice for a given product.					4 LOs					
UT-BO5	Design and monitor the implementation of a test plan and other testware for achieving stated usability and user experience goals.						6 LOs				

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
UT-BO6	Explain the rationale, process and results of usability and user experience evaluations to non-specialist stakeholders.							9 LOs			
UT-BO7	Outline important criteria for selecting a usability evaluation technique.								1 LO		
UT-BO8	Give examples of usability testing in product delivery using an Agile software development.									4 LOs	
UT-BO9	Classify tools and Artificial Intelligence (AI) support for usability testing.										4 LOs
Chapter 1	**Basic Concepts**										
1.1	Fundamentals										
UT-1.1.1	Explain the concepts of usability and user experience	K2	x								
UT-1.1.2	Give examples of the objectives of usability and user experience evaluation	K2	x								

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
1.2	Evaluating Usability and User Experience										
UT-1.2.1	Compare usability and user experience	K2	x								
UT-1.2.2	Explain the two approaches to usability and user experience	K2	x								
1.3	Usability Evaluation and Human-Centered Design										
UT-1.3.1	Explain the key elements of human-centered evaluation	K2	x								
Chapter 2	**Risks in Usability and User Experience**										
2.2	Typical Risks										
UT-2.2.1	Give examples of typical usability and user experience risks	K2		x							
UT-2.2.2	Prepare a list of usability risks that are likely to occur for a given project scenario	K3		x							

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
UT-2.2.3	Give examples of typical project risks when testing usability and user experience	K2		x							
Chapter 3	**Usability Standards**										
3.1	Usability Standards										
UT-3.1.1	Explain the general content of user interface guidelines	K2			x						
UT-3.1.2	Explain the general content and applicability of ISO usability standards to particular types of project/application	K2			x						
Chapter 4	**Usability Reviews**										
4.1	Introduction and Approach										
UT-4.1.1	Explain the usability review as a technique for evaluating usability and user experience	K2				x					
UT-4.1.2	Give examples of risks and challenges of usability reviews	K2				x					

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
4.2	Types of Usability Reviews										
UT-4.2.1	Differentiate between informal usability and expert usability reviews	K2				x					
UT-4.2.2	Apply a given checklist for a given software application using heuristic evaluation	K3				x					
Chapter 5	**Usability Testing**										
5.2	Step-By-Step Approach to Usability Testing										
UT-5.2.1	Explain the principal steps in usability testing	K2					x				
5.3	Preparing a Usability Test										
UT-5.3.1	Explain the content of a usability test plan	K2					x				
UT-5.3.2	Explain the content of a usability test script	K2					x				

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
UT-5.3.3	Prepare a simple usability test task for a given project	K3					x				
UT-5.3.4	Give examples of considerations to be applied when deciding on a location for usability tests	K2					x				
UT-5.3.5	Compare the advantages and disadvantages of a usability test lab	K2					x				
5.4	Performing a Usability Test Session										
UT-5.4.1	Explain the key activities in a usability test session	K2						x			
5.5	Analyzing Findings										
UT-5.5.1	Explain the procedure to apply for analyzing findings from a usability test	K2						x			
5.6	Communicating Test Results and Findings										
UT-5.6.1	Analyze a usability test report for a given project	K4						x			

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
UT-5.6.2	Explain how to overcome internal resistance to usability test findings	K2						x			
UT-5.6.3	Apply a given list of best practices to report and communicate usability test results	K3						x			
5.7	Quality Control of a Usability Test										
UT-5.7.1	Explain the quality control activities for a usability test	K2						x			
5.8	Challenges and Frequent Errors										
UT-5.8.1	Give examples of the most frequent and serious errors in usability testing	K2						x			
Chapter 6	**Usability Surveys**										
6.1	Step-By-Step Approach to User Surveys										
UT-6.1.1	Explain the purpose and approach adopted in user surveys	K2						x			
6.2	Standardized Questionnaires										

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
UT-6.2.1	Compare the principal content and objectives of the public user questionnaires SUS, SUMI and WAMMI	K2						x			
Chapter 7	**Selecting Appropriate Test Techniques for Usability**										
7.1	Criteria for Selecting an Evaluation Technique										
UT-7.1.1	Apply the appropriate test techniques for verifying and validating usability and user experience in a given project	K3							x		
Chapter 8	**Usability Testing in Agile Software Development**										
8.1	Integrating Usability Testing into Agile Software Development										
UT-8.1.1	Explain the usability evaluation approaches in Agile software development	K2								x	
8.2	Roles in Agile Software Development										

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
UT-8.2.1	Explain the role of a Usability Tester in the context of Agile software development	K2								x	
8.3	Usability Evaluation Approaches Suitable in Agile Software Development										
UT-8.3.1	Distinguish the different approaches to Usability Testing suitable in Agile software development	K2								x	
8.4	Shifting the Focus to User Experience										
UT-8.4.1	Differentiate between techniques to shift the focus from functionality to user experience	K2								x	
Chapter 9	**Select and Use Tools for Usability Testing**										
9.1	**Categories of Usability Testing Tools**										
UT-9.1.1	Classify the categories of usability testing tools	K2									x

© International Software Testing Qualifications Board

Business Outcomes: Usability Testing			BO1	BO2	BO3	BO4	BO5	BO6	BO7	BO8	BO9
9.2	**Application of Usability Testing Tools**										
UT-9.2.1	Explain the general application of usability testing tools	K2									x
9.3	**Limitations of Usability Testing Tools**										
UT-9.3.1	Explain the limitations of tools for usability testing	K2									x
Chapter 10	**AI Support for Usability Testing**										
10.1	Areas in Which AI Can Support Usability Testing										
UT-10.1.1	Explain the areas in which AI can support usability testing	K2									x

© International Software Testing Qualifications Board

14 Appendix C – Release Notes

ISTQB® Usability Testing Syllabus v2.0 is a major update and builds on the v1.0 release. A summary of principal changes at chapter level is provided below.

The following principal objectives have been considered in creating the Usability Testing Syllabus v2.0.

- Details of Software Accessibility are covered in a separate syllabus [ISTQB_SYL_DAT]. Some of the content used for this new Digital Accessibility Testing syllabus has been drawn from the UT v1.0 Usability Testing syllabus. Those aspects which are now to be found in the new Software Accessibility syllabus have been removed from the UT v2.0 Usability Testing syllabus to avoid overlaps.
- The chapter on usability roles and responsibilities is not included in the syllabus, although a brief listing of Agile roles is included in the Agile chapter 8
- New chapter 8 considers Usability Testing aspects in Agile Software Development
- New chapter 9 considers the tools which can support usability testing
- New chapter 10 Identifies the areas in which AI can support usability Testing

Any content in UT v1.0 which is still appropriate will continue to be used in UT v2.0. This accounts for a large proportion of the v2.0 syllabus and minimizes the number of changes needed to existing accredited courses.

© International Software Testing Qualifications Board

15 Appendix D – Domain Specific Terms

Term Name	Definition
guerilla testing	An informal, low-cost method to collect rapid user feedback using test participants on the spot in public places.
human-centered design	A design approach that aims to make component or system more usable and applies human factors, ergonomics, and usability knowledge and techniques.
persona	An archetypical user of a component or system, based on research into real users.
rainbow spreadsheet	A collaborative user experience research tool to observe and analyze test participants, represented with different colors, behaviors during usability tests or user research sessions.
real-time tagging	A lean way of capturing input from users as they interact with a digital work product by adding tags.
RITE	Rapid Iterative Testing and Evaluation: A usability testing approach used especially in user interface development.
usability heuristic	A heuristic that helps to achieve usability.

© International Software Testing Qualifications Board

16 Appendix E – Manufacturer Guidelines

A list of manufacturer guidelines is given below. To give an impression of the contents, the section titles are provided for each guideline.

Guideline Name	Section Titles	Reference
Apple OS X Human Interface Guidelines	Starting and Stopping, Modality, Interoperability, Feedback and Assistance, Interaction and Input, Animation, Branding, Color and Typography, Icons and Graphics, Terminology and Wording, Integrating with OS X.	[Web-1]
Android User Interface Guidelines (Google)	Animation, Style, Layout, Components, Patterns, Usability	[Web-2]
Microsoft Windows User Experience Interaction Guidelines	Design Principles, Controls, Commands, Text, Messages, Interaction, Windows, Visuals, Experiences, Windows Environment	[Web-3]
SAP design guidelines and resources	Action, Container, Data visualization, Display, Filter, List, Loading, Popover, Table, Toolbar, User input.	[Web-4]

© International Software Testing Qualifications Board

17 Index

agile 9, 55
AI support 9
analysis and evaluation tools 60
analysis of findings 41
Artificial Intelligence (AI) 9, 63
behavior-driven development 58
Common Industry Format 28
data capture tools 60
defect severity ratings 42
designing usability tests 55
effectiveness 13
efficiency 13
expert usability review 17, 31
eye-tracking tools 60, 61
formative 17
formative evaluation 13, 57
frequent and serious errors 46
heuristic evaluation 17, 32
heuristics 32
human-centered design 13, 19, 20, 53
human-centered evaluation 13
informal usability review 17, 31
ISO 9241-110 27
ISO 9241-210 19, 28
ISO Usability Standards 27
mock up office 35
moderator 34, 36, 40
observation and collaboration tools 60
observer 36, 38
organizational risks 24
persona 54, 57
pilot usability test session 39
planning usability tests 55
principal test techniques 36
product risk 22
product risk assessment 23
product risks 23
project risk 22
project risks 23, 24
prototyping tools 60
qualitative usability evaluation 18
quality control tasks 45
quantitative usability evaluation 18
remote usability testing 36
reporting of usability tests 55
risk 23
risk assessment 23
risk identification process 23
RITE – Rapid Iterative Testing and Evaluation 53, 56
roles in Agile Software Development 56
scribe 36
Software Usability Measurement Inventory 48, 50
stakeholders 42
standardized questionnaires 49
standards 9, 10
step-by-step approach to usability reviews 30
step-By-step approach to user surveys 49

© International Software Testing Qualifications Board

summative evaluation 13, 17
supplier risks 25
System Usability Scale 49
System Usability Scale SUS 48
technical risks 24
test plan 9
test session 36
think aloud usability testing 17, 34
three amigos 58
usability 9, 13
Usability 14
usability criteria 16
usability defect 14
usability defect report 42
usability evaluation 13, 16, 18, 20
usability evaluation technique 9
usability findings 41
usability heuristic 29
usability maturity matrix 52
usability requirement 36
usability requirements 19
usability review 30
usability reviews 14, 29, 57
Usability Reviews 31
usability risks 9, 23
Usability Standards 26
usability test lab 38, 39
usability test location 38
usability test participant 36, 40
usability test plan 36
usability test report 43
usability test reporting best practices 44
usability test results 41
usability test script 37
usability test session 39
usability test session activities 40
usability test sessions 35
usability test short form 52
usability test task 37
usability test techniques 53
usability tester 56
usability testing 9, 14
usability testing tools 60
usability testing unmoderated 52
user characteristics 15
user experience 9, 13, 15
user experience evaluation 16
user Experience Evaluation Technique 52
user experience risks 24
user interface 13
user interface guideline 27
user interface guidelines 27
user survey 48, 49
user surveys 14
UX specialist 56
valuation technique 43
Website Analysis and Measurement Inventory 50
Website Analysis and Measurement Inventory (WAMMI) 48

© International Software Testing Qualifications Board