ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ISO 9241-210

Fabio Vitali

Two user-oriented process models

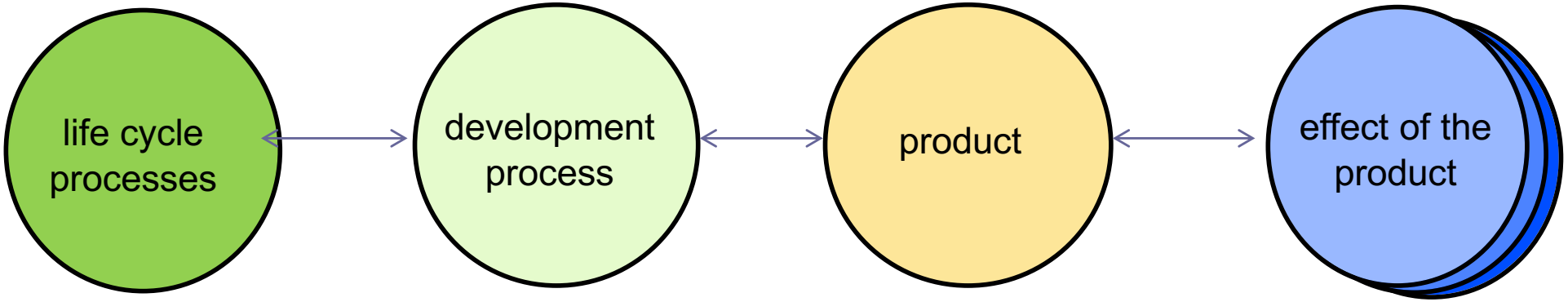
A task-oriented model

- ISO 9241-210 (2010)
  - Official international standard, originally from UK
  - Aim: usabilty design
  - Five phases: Feasibility study, User Requirements, Implementation, Evaluation, Deploy

A goal-oriented model

- *Jesse James Garrett (2011)*
  - *Well-known professional, USA, word-of-mouth (passaparola)*
  - *Aim: User Experience Design*
  - *Five planes: Strategy, Purpose, Structure, Skeleton, Surface*

ISO standards on Human-Centred Design



usability capability

ISO TR 18529
ISO PAS 18152

User-centred process

ISO 9241 - 210

Interfaces and hardware

ISO 9241-100 series
ISO 9241-300 series

usability in context

ISO 9241-110
ISO 9241-210

Organizational Capability

The *usability maturity model* in ISO 18529 contains a structured set of processes and a survey of good practice models.

It can be used to assess the extent to which an organisation is capable of carrying out user-centred design.

ISO 18152 extends this to the assessment of the maturity of an organisation in performing the processes that make a system usable, healthy and safe.

- *ISO TR 18529 (Technical report): Ergonomics of human-system interaction - Human-centred lifecycle process descriptions (2000)*
- *ISO PAS 18152 (Publicly Aware Specification): Ergonomics of human-system interaction – A specification for the process assessment of human-system issues (2003)*

Process Quality

ISO 9241-210, Ergonomics of human-system interaction

It provides guidance on human-system interaction throughout the life cycle of interactive systems.

The standard describes 6 key principles that will ensure the design is user centred:

1. The design is based upon an explicit understanding of users, tasks and environments.
2. Users are involved throughout design and development.
3. The design is driven and refined by user-centred evaluation.
4. The process is iterative.
5. The design addresses the whole user experience.
6. The design team includes multidisciplinary skills and perspectives.

Product Quality

Quality of Software Products (ISO 9241-100 series)

These standards can be used to support user interface development:

- To specify details of the appearance and behaviour of the user interface.
- To provide detailed guidance on the design of user interfaces
- To provide criteria for the evaluation of user interfaces

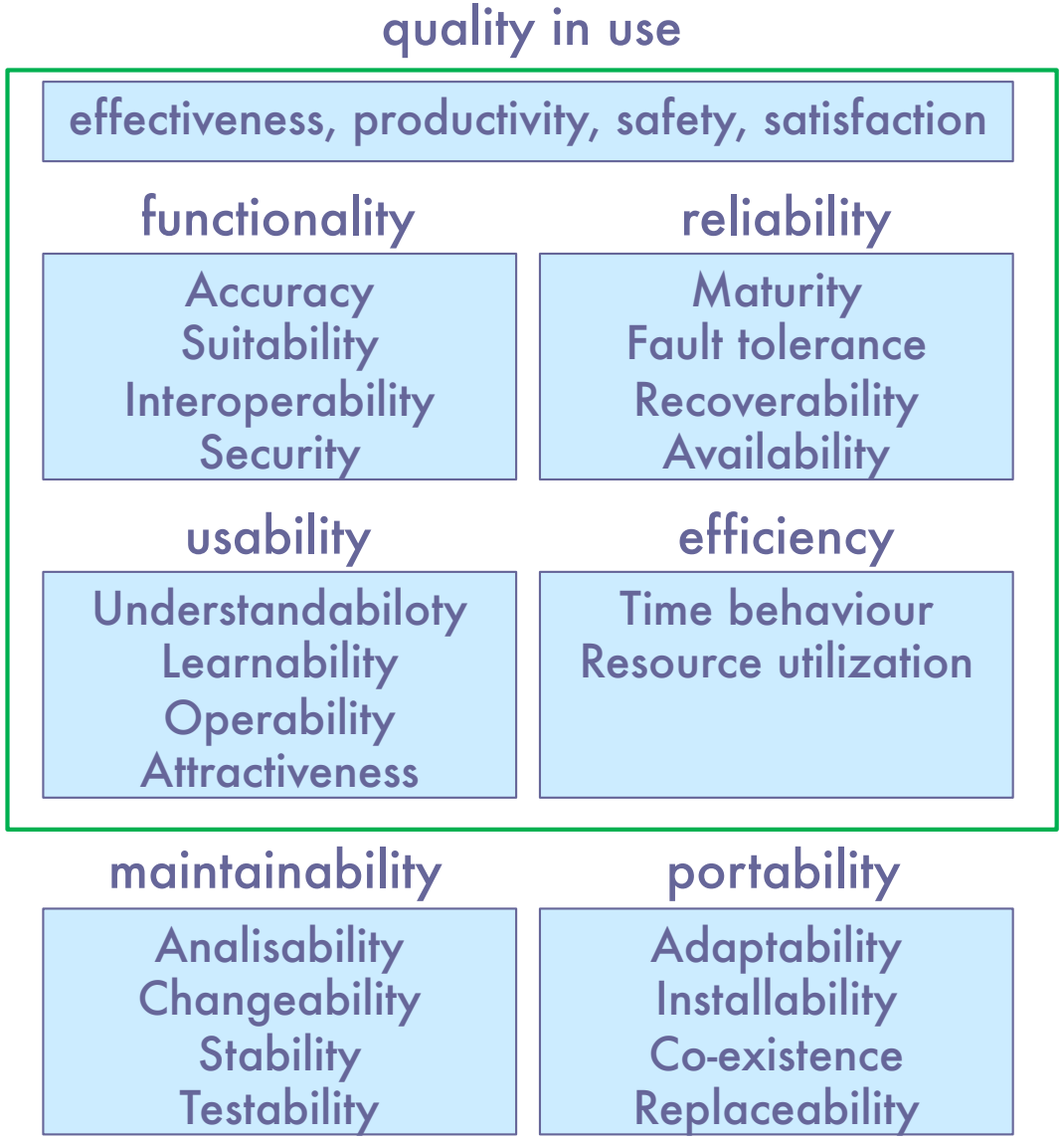
Quality of Hardware Products (ISO 9241-300 series)

These standards can be used in the design and evaluation of workplaces, screens, keyboards and other input devices.

Unlike the software standards, most of these standards contain explicit requirements.

- Requirements for visual display terminals in offices
- Gestures for pen-based systems
- Ergonomic requirements for the design of control centres.

Quality in use (1)



ISO/IEC 9126 is a standard developed as a software engineering process method.

It describes six categories of software quality relevant during product development:

- functionality,
- reliability,
- usability,
- efficiency,
- maintainability and
- portability.

Quality in use (2)

- ISO 9126-1defines usability as "the capability of the software product to be understood, learned, used and attractive to the user, when used under specified conditions."
- ISO 9126-1 further defines quality in use as "the capability of the software product to enable specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction in specified contexts of use."

- The terms "under specified conditions" and "in specified contexts of use" represent the ***quality in use*** specification that emphasises that there is **no abstract definition of usability**.

- ISO 9241-110 and -210 provides extended discussions and metrics for the evaluation of quality in use.

ISO 9241 (1)

Ergonomics of Human System Interaction

A multi-part standard from the ISO managed by ISO Technical Committee #159.

Originally titled "Ergonomic requirements for office work with visual display terminals (VDTs)".

Renamed in 2006 to remove "office work" and extend the reach beyond VDTs.

Initiated in 1993 after strong input from UK BSI standard Institute, it has a ten years release cycle.

Current version was approved in 2010 and consists currently of 22 different documents in 8 broad topics.

A new editing cycle has started with the release of 9241-110 in 2020. The others will follow in the next few years.

ISO 9241 (2)

100 series: Software ergonomics

- Part 100: Introduction to standards related to SW ergonomics
- ***Part 110: Dialogue principles***
- Part 129: Guidance on software individualization
- Part 151: Guidance on World Wide Web user interfaces
- Part 143: Forms
- Part 161: Guidance on visual user interface elements
- Part 171: Guidance on software accessibility

200 series: Human system interaction processes

- ***Part 210: Human-centred design for interactive systems***

ISO 9241 (3)

300 series: Displays and display related hardware

- Part 300: Introduction to electronic visual display requirements
- Part 302: Terminology for electronic visual displays
- Part 303: Requirements for electronic visual displays
- Part 304: User performance test methods for electronic visual displays
- Part 305: Optical laboratory test methods for electronic visual displays
- Part 306: Field assessment methods for electronic visual displays
- Part 307: Analysis and compliance test methods for electronic visual displays
- Part 308: Surface-conduction electron-emitter displays (SED)
- Part 309: Organic light-emitting diode (OLED) displays
- Part 310: Visibility, aesthetics and ergonomics of pixel defects

ISO 9241 (4)

400 series: Physical input devices - ergonomics principles

- Part 400: Principles and requirements for physical input devices
- Part 410: Design criteria for physical input devices

500 series: Workplace ergonomics

600 series: Environment ergonomics

700 series: Application domains - Control rooms

900 series: Tactile and haptic interactions

- Part 910: Framework for tactile and haptic interaction
- Part 920: Guidance on tactile and haptic interactions

Empty series are under discussions and open to replanning in any moment.

ISO 9241-110: *Dialogue Principles*

Originally ISO 9241-10, renumbered in 2006

General ergonomic principles which apply to the design of dialogues between humans and information systems

It describes seven "dialogue principles", discussed in detail in further parts of series 100 of ISO 9241:

1. *Suitability for the task*: the dialogue should be suitable for the user's task and skill level;
2. *Self-descriptiveness*: the dialogue should make it clear what the user should do next;
3. *Controllability:* the user should be able to control the pace and sequence of the interaction;
4. Conformity with user expectations: the dialogue should be consistent;
5. *Error tolerance*: the dialogue should be forgiving;
6. *Suitability for individualisation*: the dialogue should be able to be customised to suit the user; and
7. *Suitability for learning*: the dialogue should support learning.

Quality in use (ISO 9241-110)

The quality in use is defined as "the capability of a system to allow specified users to achieve specified goals with effectiveness, efficiency, safety and satisfaction in a *specified* context of use".

There is a difference between:

- quality in use (which concerns the quality that the product gives to the user when used)
- product quality (regarding the quality of the product as such)
- the quality of the process (as the design and development team carries out its activities)

N.B,: There is also difference between:

- Specific (one and not others)
- Specified (described in the documentation and explicitly based on an analysis and implementation work).
- ISO 9241-110 refers always to a ***specified*** context of use

ISO 9241-210: Human-Centred Design For Interactive Systems

Originally ISO 13407, renumbered in 2010

A high level overview of the activities that are recommended for human centred design throughout the life cycle of interactive systems.

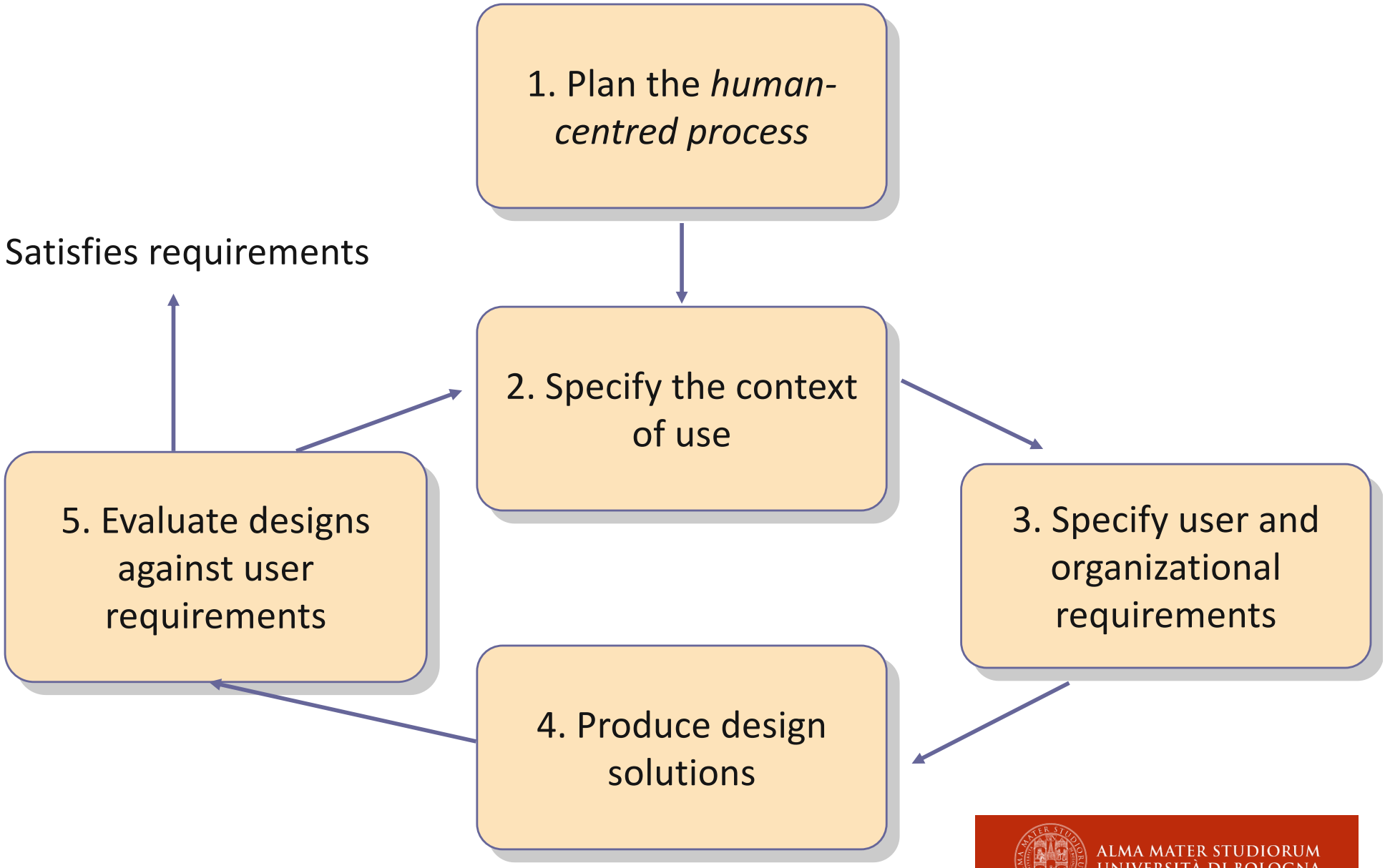
It is a tool for those managing design processes and provides guidance on sources of information and standards relevant to the human-centred approach.

It describes human-centred design as a multidisciplinary activity, which incorporates human factors and ergonomics knowledge and techniques with the objective of enhancing effectiveness and efficiency, improving human working conditions, and counteracting possible adverse effects of use on human health, safety and performance.

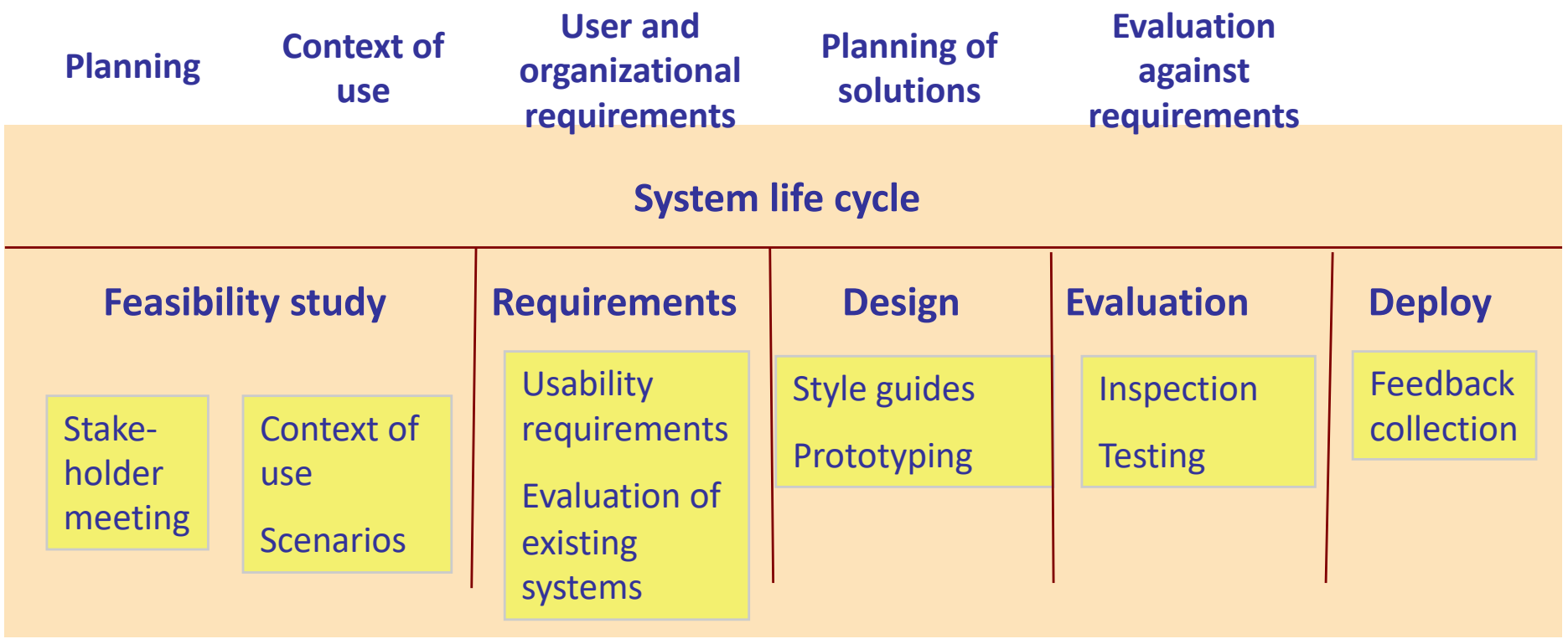
ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ISO 9241-210: the design process

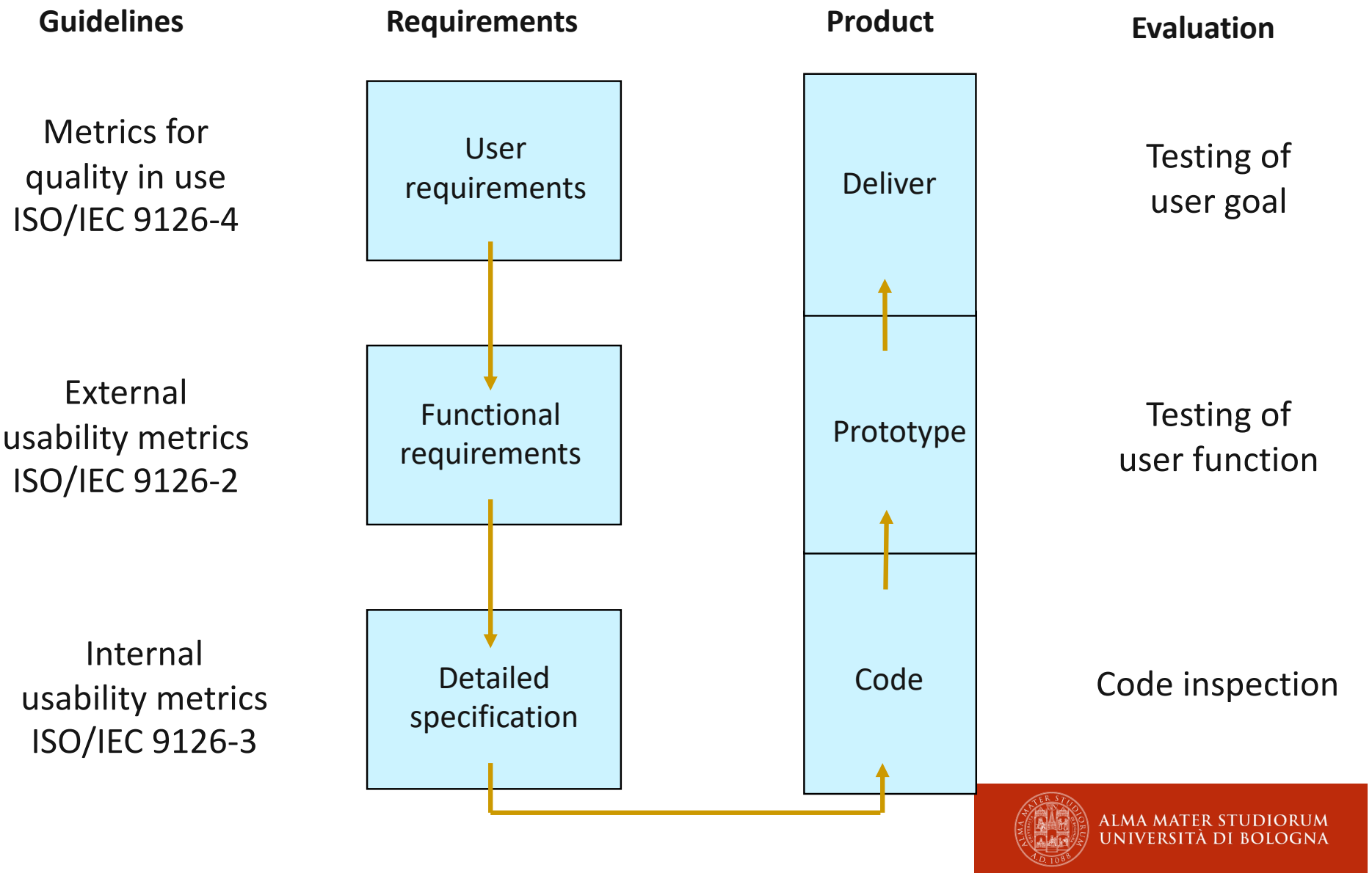
ISO 9241-210: Human-centred design process



Human-centred design: the steps



User-centred design: relationships with ISO/IEC 9126



User-centred design: the steps

Feasibility study

1. Stakeholder meeting
2. Context of use analysis
3. Scenarios

User requirements

4. Assessment of existing systems
5. Analysis of logs
6. Interviews
7. Direct observation
8. Context of use
9. Scenarios
10. Task analysis

Design

11. Identifying and using style guides
12. Prototyping

Evaluation

13. Internal prototype inspection
14. Usability testing

Deploy

15. Feedback collection

Feasibility study

1. Stakeholder meetings

Meetings with (representatives of) all stakeholders to:

- Identify the objectives of the system and its main objectives;
- Identify the stakeholders;
- Identify what the success criteria will be;
- Collect all people involved in design and create a shared vision.
- Decide on the role of usability, contexts of use, and how they relate to business goals;

Feasibility study

2. Context of use analysis

Meetings of the development team to gather and agree on information such as:

- Who are the intended users and what their tasks are;
- What are the technical and environmental constraints.

It ensures that all factors affecting the use of the system have been identified before the start of design;

Provides a basis for designing usability tests.

Generates a feasibility document explicitly approved by project management and / or the customer.

Feasibility study

3. Scenarios

Documenting how users are expected to carry out their tasks in a specific context, which is both input for the project and for the subsequent testing.

Designers use them to take into account the characteristics of users and their tasks and environment.

Usability issues can be identified at a very early stage of the design process (before any code has been written).

Scenarios can help identify the usability goals and probable completion times.

User requirements

4. Evaluation of existing systems

Either an earlier version or a system from a competitor

Verify and assess usability issues to be used as quantitative and qualitative metrics for the design phase and for later evaluation.

N.B.: A mere functional analysis is useful and often it is the only comparison requested, but it does NOT HELP usability. The assessment must be made *on* the *usability* of such systems.

User requirements

5. Analysis of logs

Useful if you have access to a currently deployed system, either from a competitor or from the current version of the system you are redesigning,

Checking the logs you can gather large amounts of data from your system's current usage patterns without involving users directly.

Through log analysis you can understand:

- The use patterns that real users make of the system
- What are the concatenations of tasks
- What are the most frequent errors (involuntary choices, interrupted tasks, dead ends, depth of dead ends, etc.)
- Resolution strategies adopted when facing problems

User requirements

6. Interviews

Interviews of a representative sample (even a small quantity) of the end users, identified in collaboration with the client.

They will not be used for later tests

- What are the activities that make you waste most of your time?
- What are your goals and priorities when using the service?
- What are the elements that help you make the decisions?

On pre-existing systems:

- What are the most common things you can do through the service and what are the parts you use most frequently?
- What is your favorite aspect and what do you detest?
- What shortcuts do you use?
- How do you solve the problems that arise?

User requirements

7. Direct observation

A technique derived from ethnography, studying human activities through their direct observation in the socio-cultural environment in which it is carried out.

It consists in approaching end users and observing their interaction with existing systems in their workplace, taking notes of problems they have.

- A team member sits next to a representative user and watches him/her doing his/her job.
- He makes questions, asks for information and explanations *without ever mentioning the features of the future system.*

It is a source of stress and distraction, it can cause suspicion and fears in those involved or (worse) in those who are not.

Do not mistake the activities as actually performed as as described in job descriptions and regulations.

It is, typically the moment in which you discover the differences in actual practices between what your clients believe and your users perform.

User requirements

8. Context of use

Aims at producing documented descriptions of the technical constraints of the design

- Identification of users
- Identification of their tasks
- Identification of technical constraints
- Identification of cultural constraints
- Identification of environmental constraints

It can be realized both in the feasibility phase or in user requirements phase

There are templates to create context of use analysis:

- http://www.usabilitynet.org/papers/Context_table_3users_4tasks.doc

User requirements

9. Scenarios

In addition to the scenarios agreed upon with all the stakeholders during the feasibility study.

Short stories providing details about carries out one or more of the tasks specified for the system.

- Decomposition of user tasks in actions (internal and external to the system)
- Identification of the operations performed by the user and those performed by the computer

Write (in plain text) a narration of the user's actions

- DO NOT specify which features are used (system as a black box)
- Establish time estimates and success criteria for scenario and for each individual action within it.

Scenarios are useful to describe *both* frequent tasks *and* critical situations

- Describe not only the typical features, but also, and above all, those that test the specific characteristics of the system.

User requirements

10. User Requirements

A document identifying the requirements for groups of users and tasks specified in contexts of use and scenarios, and establishing clear and quantitative usability requirements specifications:

Requirements are explicit and numbered, and explicitly mention quantitative verification criteria according to metrics agreed upon with the client.

User requirement documents explicitly deal with:

- task concatenations;
- the characteristics and roles of users;
- the objectives and content of the process and the activities carried out by users;
- the interface and input and output devices of the computer system;
- the rules and procedures related to the domain of the application;
- working routines;
- communication flows;
- the main business criticalities.

Design

11. Identifying and using style guides

Identify, describe, and adopt one or more (explicit and documented) guide lines relative to the technological, the industrial, the organizational, and the project context relevant for the design of screenfuls and dialogue.

Important to provide a consistent look and feel of the interface.

They should be agreed upon as part of the usability and compliance requirements and their application should be monitored during the development phases.

Design

12. Prototyping

Use of low-fidelity prototypes (drawings on paper or partially completed implementations) to clarify the user requirements and provide faster answers to impementation doubts or discussions.

Low fidelity allows to create new prototypes in a fast cycle of design, implementation and testing.

At least four types of prototypes:

- Concept design: to explore different metaphors and design strategies
- Interaction design: to organize the overall structure of the system
- Screen design: for the initial design of each single screen
- Screen testing: to refine the layout of the screen

Evaluation

13. Internal prototype inspection

Low cost and low reliability methods to establish potential usability problems without the involvement of actual users.

Provides basic, common sense, cheap evaluation to streamline and avoid the most obvious and most embarassing usabilty issues.

Success in usability inspection provides no certainties, but some confidence that the design is mature enough to be shown to some users for actual testing.

Evaluation

14. Usability testing

Actual tests performed over a (representative) sample of the target users carrying out tasks by measuring real results against expectations of the design team.

Two basic types:

- **Formative test***:* to quickly identify and solve usability problems ***during the design phase of the project***. Helpful to quickly identify problems before it is too late to fix them.
- **Summative test***:* to fully evaluate ***the system in the final state*** with respect to the user requirements described and decided in the initial phases of the requirements. Used to verify how the actual system compares with the initial requirements.

NB: Do not call them **User Tests** but ***Usability Tests***

- we do not test users, but employ users to test the usability

Deploy

15. Collection of feedback

Collection of information from unsupervised users using the actual deployed system. Voluntary or forced.

Uses many ways to collect data:

- Surveys (both paper-based and online),
- Telephone helpline,
- Documentation web sites,
- User forums,
- etc.

Meant to identify what issues need to be resolved in future releases.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ISO 9241-210
Evaluation metrics

Usability metrics (1)

The basic usability requirements metrics derive directly from the three factors of the quality in use as specified by ISO 9241-210,

Often called the metric of the three *E*s:

- ***Effectiveness****:* The accuracy and completeness with which the user concludes the tasks entrusted to him
  - Primary unit: errors: their number and gravity
- ***Efficiency****:* the ratio between the accuracy and completeness of the task and the use of the resources required to complete them.
  - Primary unit: time
- ***Emotions*** *(or satisfaction):* the presence (or absence) of feelings of dissatisfaction, and the kind of attitude (positive or negative) perceivable in a continuous use of the system.
  - Primary unit: subjective vote (eg 1 to 10)

Usability metrics (2)

Examples of effectiveness metrics

- Percentage of completed tasks
- Percentage of system functionality used
- Percentage of tasks completed on the *first attempt*
- Percentage of testers able to complete the task
- Percentage of testers who can complete the task without using the manuals
- Number of persistent errors
- Number of errors in a time unit
- Number of errors for each task
- Number of service requests
- Objective metrics of output quality
- Objective metrics of output quantity

Usability metrics (3)

Examples of efficiency metrics

- Run time for a particular task
- Run time on the first attempt
- Run time after a certain period away from the product
- Installation time of the system
- Overall time spent on the manual
- Overall time spent re-learning the functions
- Learning time of a new solution approach to a task
- Time used to correct errors
- Ratio between running times of a tester and an expert
- Time to reach the performance of an expert
- Number of keys pressed
- Number of memorized icons after task completion

Usability metrics (4)

Examples of emotion (satisfaction) metrics

- Relationship between positive and negative adjectives in the descriptions of the product
- Percentage of testers who felt "in control" of the system during the test
- Percentage of testers who consider it more satisfactory than products from the competition
- Percentage of testers who consider it easier to use than products from the competition
- Percentage of testers who felt to be "more productive" than when using products from the competition
- Percentage of testers that after the test "would recommend it to a friend"

ISO 9241-210 – Final checklist

Analysis without users

- Have all the relevant tasks been considered?
- Have all the tasks been decomposed correctly?
- Have all the tasks considered all the special situations?
- Have all the tasks considered the possible errors?
- Do all tasks refer to problems and not solutions?
- Are the tasks independent and self-sufficient?
- Do tasks have closure?

Analysis with users

- Are the tasks appropriate?
- Are the tasks complete?
- Are people described in the assignments credible and representative?
- Are worst case situations considered and managed?

References

Jesse Garrett, *The elements of user experience*, New Riders, 2011

Philip Harris, *Data Driven Design*, K&R publications, 2013

A. Cooper, R. Reimann, D. Cronin, *About Face 3*, Wiley, 2009

Alan Cooper, *The inmates are running the asylum*, SAMS, 2004

T. Tullis, B. Albert, *Measuring the User Experience*, Morgan Kaufmann, 2013

Susan Weinschenk, *100 things every designer needs to know about people*, New Riders, 2011

Usability Book of Knowledge, http://www.usabilitybok.org/