Software Engineering Essentials

Usability

Bernd Bruegge, Stephan Krusche, Andreas Seitz, Jan Knobloch
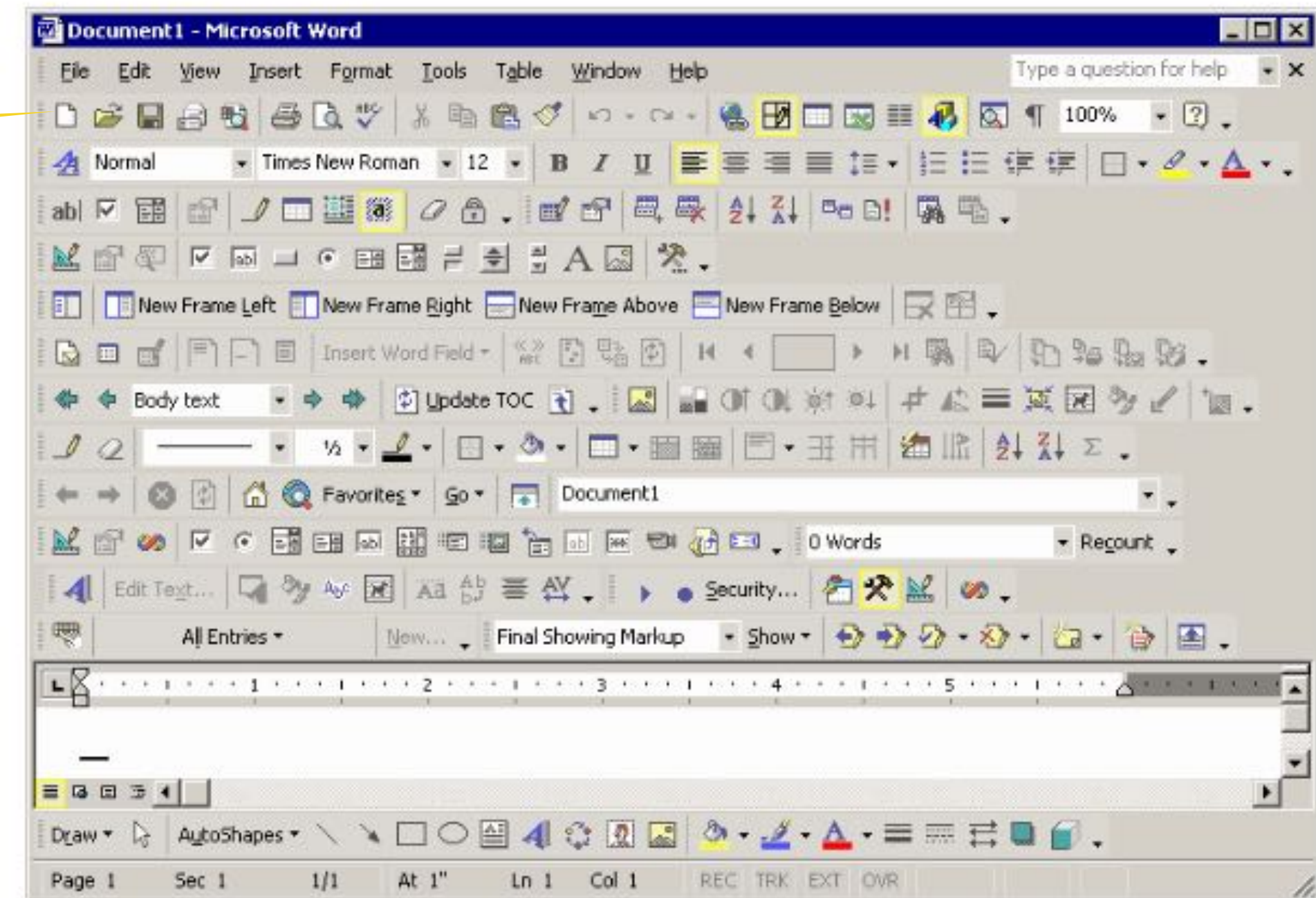Chair for Applied Software Engineering — Faculty of Informatics

TUM

Learning Goals

1) Understand the definition of Usability

2) Explain different aspects of Usability and its key goals

3) Apply heuristics towards your usability concepts

Motivation and definition

"Usability is like cooking: everybody needs the results, anybody can do it reasonably well with a bit of training, and yet it takes a master to produce a gourmet outcome." [JN09]

How can I align my text to the left?

Which button shuts down a reactor? Or which one increases its output?

[JN09]   Jakob Nielsen, Anybody can do usability: https://www.nngroup.com/articles/anybody-can-do-usability, 2009

Usability - a non functional requirement

Usability is a nonfunctional requirement:

Example questions:

- What is the level of expertise of the user?

- What user interface standards are familiar to the user?

- What documentation should be provided to the user?
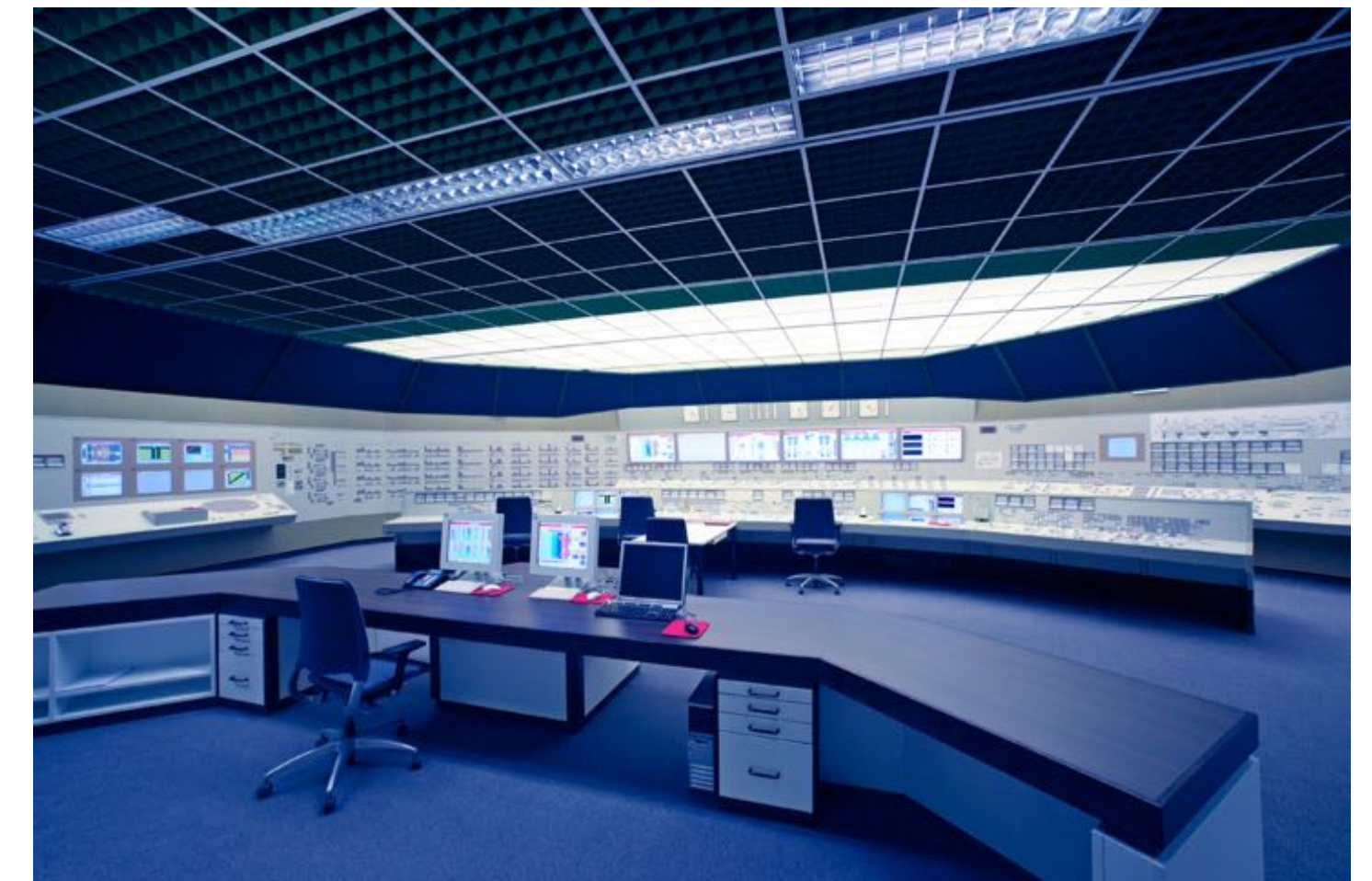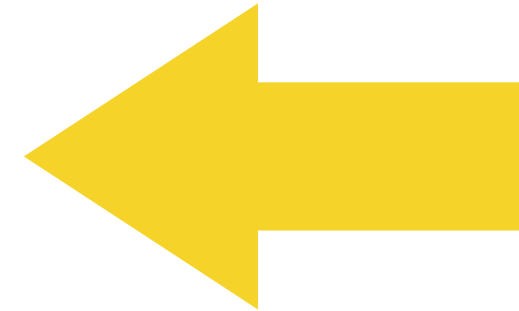
Different categories of usability

Usability is multidimensional:

1) Learnability: Is the user interface easy to learn?

2) Efficiency: Once it is learned, is it fast to use?

3) Memorability: Is it easy to remember what the user has learned?

4) Error handling and robustness: Are errors recoverable?

5) Satisfaction: Is the user interface enjoyable to use?

[JN12]   Jakob Nielsen, Anybody can do usability: https://www.nngroup.com/articles/usability-101-introduction-to-usability/, 2012

Nielsen's 10 usability heuristics

Meet expectations

1. Match the real world

2. Consistency & standards

3. Help & documentation

The user is the boss

4. User control & freedom

5. Visibility of system status

6. Flexibility & efficiency

Handle errors

7. Error prevention

8. Recognition, not recall

9. Error reporting, diagnosis, and recovery

Keep it simple

10. Aesthetic & minimalistic design

[JN95]   Jakob Nielsen, Anybody can do usability: https://www.nngroup.com/articles/ten-usability-heuristics/, 1995
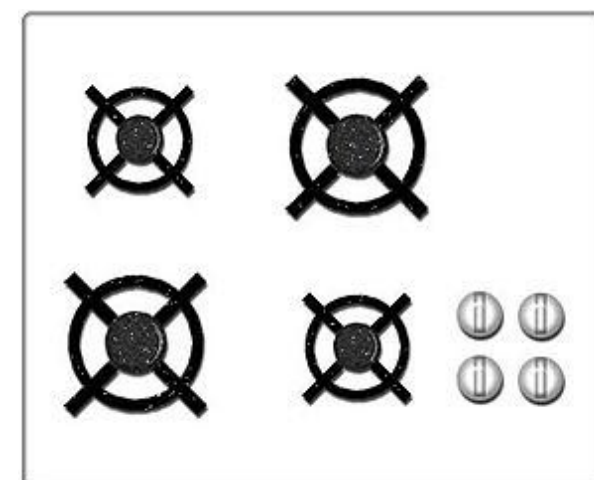
Example: match the real world

- Speak the user's language

  - Use common words, not techie jargon

  - Use application domain specific terms where appropriate

- Don't put limits on user defined names

- Allow aliases/synonyms in command languages

- Use metaphors: A well chosen metaphor can be quite effective and appealing

Natural Mapping: Close Match between controls and objects

Stove #1

Stove #2

Software Engineering Essentials

Usability

Bernd Bruegge, Stephan Krusche, Andreas Seitz, Jan Knobloch
Chair for Applied Software Engineering — Faculty of Informatics