QS STARS RATED FOR EXCELLENCE 2012

FPT Education

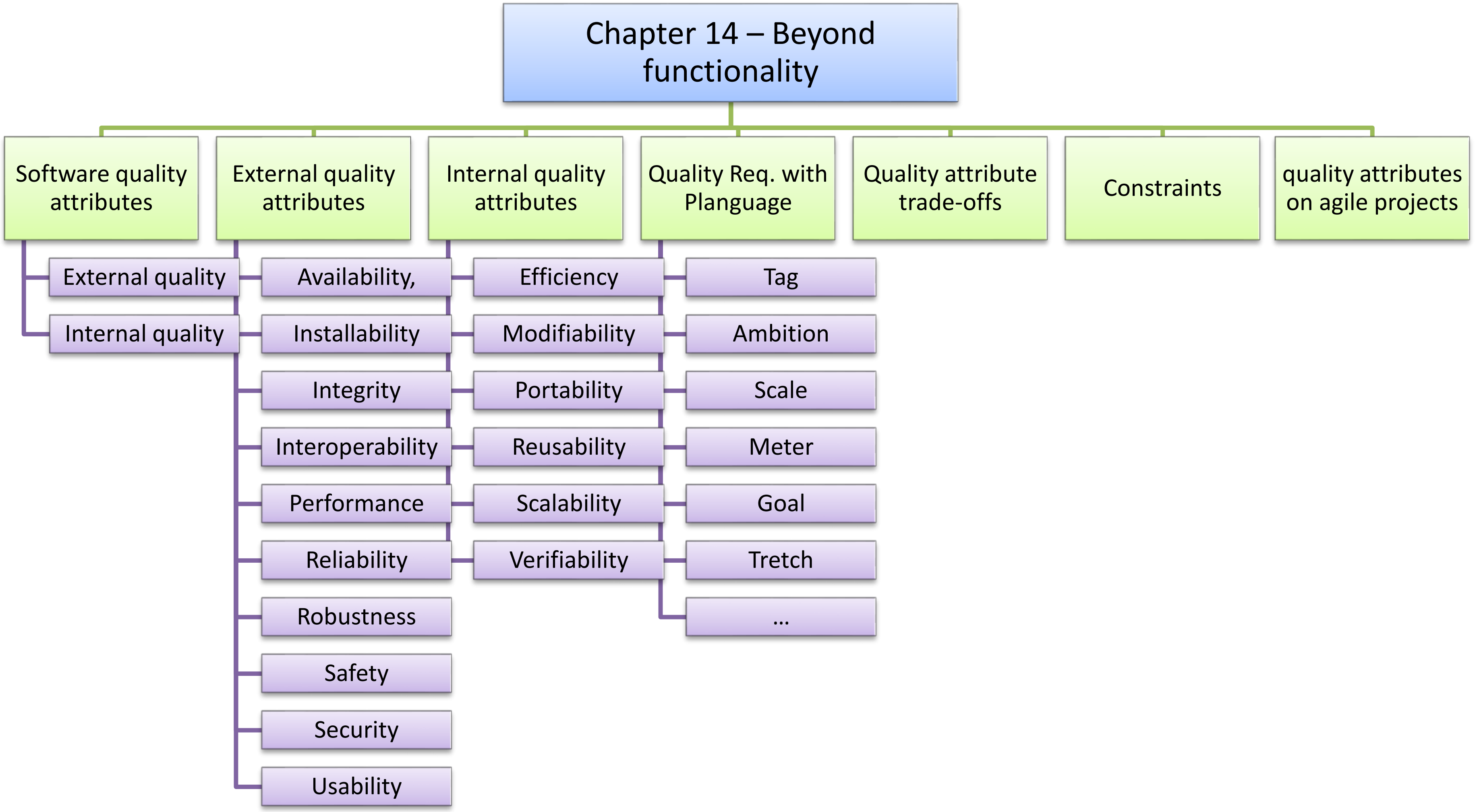
FPT UNIVERSITY

ĐẠI HỌC FPT CẦN THƠ

CHAPTER 15

Risk reduction through prototyping

Review chapter 14



Objectives

- This chapter describes how prototyping provides value to the project and different kinds of prototypes you might create for different purposes.

- It also offers guidance on how to use them during requirements development, as well as ways to make prototyping an effective part of your software engineering process.

Contents

Prototyping: What and why?

- Purpose:
  - Software prototyping takes a tentative (ướm thử) step into the solution space. It makes the requirements more real, brings use cases to life, and closes gaps in your understanding of the requirements.
  - Prototypes can serve three major purposes, and that purpose must be made clear from the very beginning:
    - Clarify, complete, and validate requirements: Used as a requirements tool, the prototype assists in obtaining agreement, finding errors and omissions, and assessing the accuracy and quality of the requirements. User evaluation of the prototype points out problems with requirements and uncovers overlooked requirements, which you can correct at low cost before you construct the actual product.
    - Explore design alternatives: Used as a design tool, a prototype lets stakeholders explore different user interaction techniques, envision the final product, optimize system usability, and evaluate potential technical approaches.
    - Create a subset that will grow into the ultimate product: Used as a construction tool, a prototype is a functional implementation of a subset of the product, which can be elaborated into (xd thành) the complete product through a sequence of small-scale development cycles (chuỗi chu kỳ phát triển qui mô nhỏ).

Prototyping: What and why?

- Three classes of prototype attributes: Because of the risk of confusion, it's important to put some descriptors (mô tả) in front of the word "prototype" so the project participants understand why and when you might create one type of prototype or another.
  - Scope: A mock-up (giả lập) prototype focuses on the user experience; a proof-of-concept prototype explores the technical soundness (tính hợp lý) of a proposed approach.
  - Future use: A throwaway (dùng 1 lần) prototype is discarded after it has been used to generate feedback, whereas an evolutionary prototype grows into the final product through a series of iterations.
  - Form: A paper prototype is a simple sketch drawn on paper, a whiteboard, or in a drawing tool. An electronic prototype consists of working software for just part of the solution.

Mock-ups and proofs of concept

- Definition: people say “software prototype,” they are usually thinking about a mock-up of a possible user interface. A mock-up is also called a *horizontal prototype*. Such a prototype focuses on a portion of the user interface; it doesn’t dive into all the architectural layers or into detailed functionality.
- A proof of concept, also known as a *vertical prototype*, implements a slice of application functionality from the user interface through all the technical services layers. A proof-of-concept prototype works like the real system is supposed to work because it touches on all levels of the system implementation.
- Purpose:
  - demonstrate the functional options the user will have available, the look and feel of the user interface (colors, layout, graphics, controls), and the navigation structure.
  - explore some specific behaviors of the intended system, with the goal of refining the requirements.
  - represent the developer’s concept of how a specific use case might be implemented.
  - …

Throwaway prototypes

- Purpose: Before constructing a prototype, make an explicit and well-communicated decision as to whether the prototype is exploratory only or will become part of the delivered product. Build a *throwaway prototype* to answer questions, resolve uncertainties, and improve requirements quality (Davis 1993).
- A *wireframe* is a particular approach to throwaway prototyping commonly used for custom user interface design and website design.
- You can use wireframes to reach a better understanding of three aspects of a website:
  - The conceptual requirements
  - The information architecture or navigation design
  - The high-resolution, detailed design of the pages

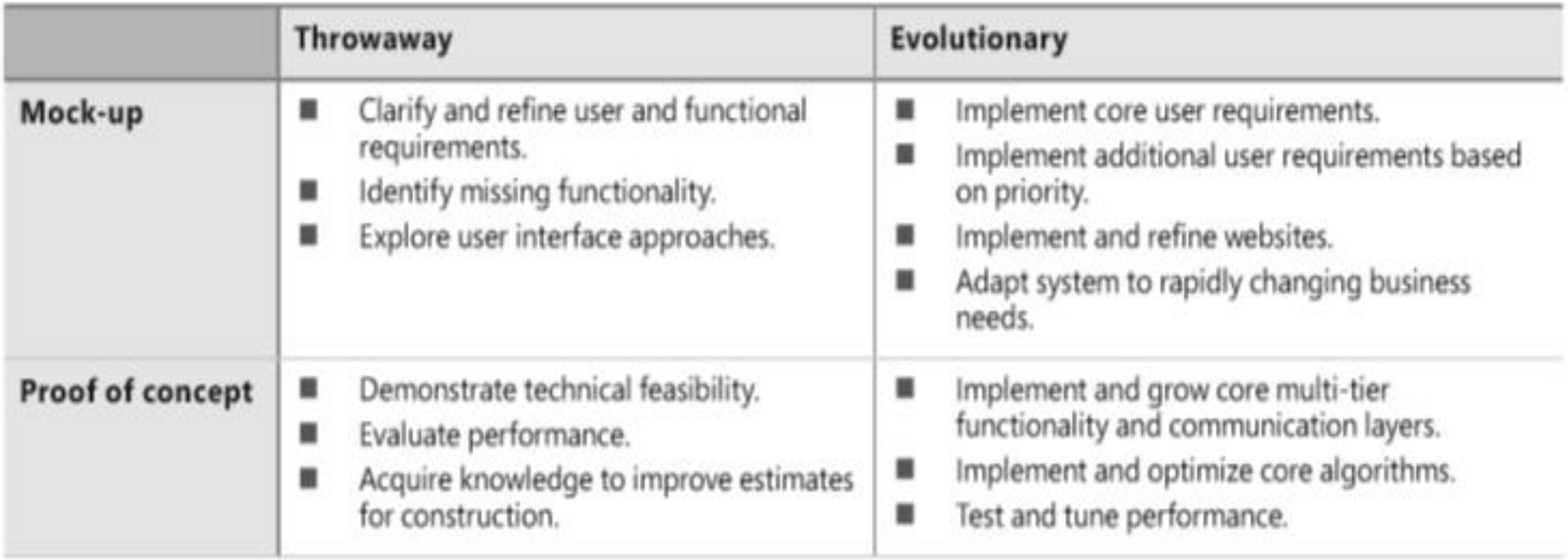
Evolutionary prototypes

- Purpose: An *evolutionary prototype* provides a solid architectural foundation for building the product incrementally as the requirements become clear over time (McConnell 1996).
- Agile development provides an example of evolutionary prototyping. Agile teams construct the product through a series of iterations, using feedback on the early iterations to adjust the direction of future development cycles.
- Evolutionary prototyping is well suited for web development projects.

Evolutionary prototypes

- Table 15-1 summarizes some typical applications of throwaway, evolutionary, mock-up, and proof-of-concept prototypes:

TABLE 15-1 Typical applications of software prototypes



	Throwaway	Evolutionary
Mock-up	■ Clarify and refine user and functional requirements. ■ Identify missing functionality. ■ Explore user interface approaches.	■ Implement core user requirements. ■ Implement additional user requirements based on priority. ■ Implement and refine websites. ■ Adapt system to rapidly changing business needs.
Proof of concept	■ Demonstrate technical feasibility. ■ Evaluate performance. ■ Acquire knowledge to improve estimates for construction.	■ Implement and grow core multi-tier functionality and communication layers. ■ Implement and optimize core algorithms. ■ Test and tune performance.

Paper and electronic prototypes

- *Paper prototype* (sometimes called a *low-fidelity* *(trung thực thấp)* *prototype*) is a cheap, fast, and low-tech way to explore how a portion of an implemented system might look. Paper prototypes help you test whether users and developers hold a shared understanding of the requirements.
    - The designer sketches ideas of possible screens without worrying about exactly where the controls appear and what they look like.

- *Electronic prototype*: Numerous tools are available if you decide to build an electronic throwaway prototype. They range from simple drawing tools such as Microsoft Visio and Microsoft PowerPoint to commercial prototyping tools and graphical user interface builders.
    - Such tools will let you easily implement and modify user interface components, regardless of how inefficient the temporary code behind the interface is.

Working with prototypes

- Figure 15-2 shows one possible sequence of development activities that moves from use cases to detailed user interface design with the help of a throwaway prototype.
- Each use case description includes a sequence of actor actions and system responses, which you can model by using a dialog map to depict a possible user interface architecture.
- A throwaway prototype or a wireframe elaborates the dialog elements into specific screens, menus, and dialog boxes.

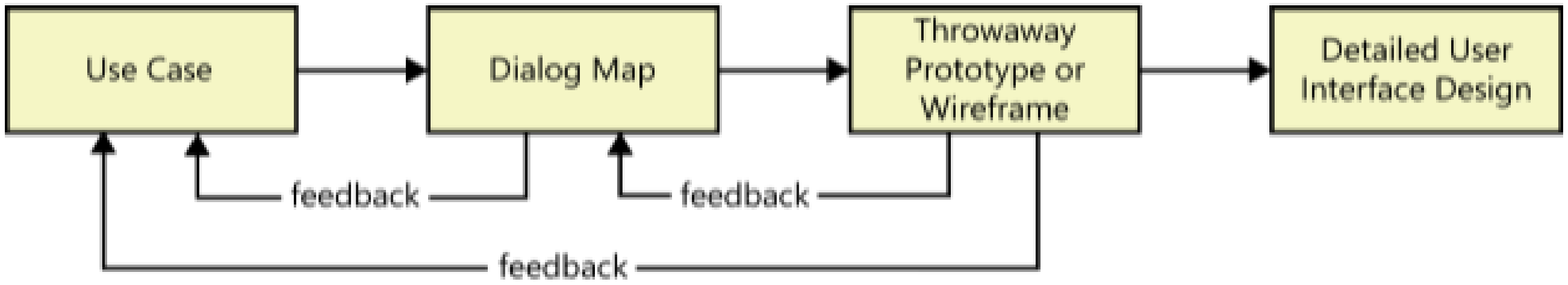

Use Case
Dialog Map
Throwaway Prototype or Wireframe
Detailed User Interface Design
feedback
feedback
feedback

FIGURE 15-2 Activity sequence from use cases to user interface design using a throwaway prototype.

Working with prototypes

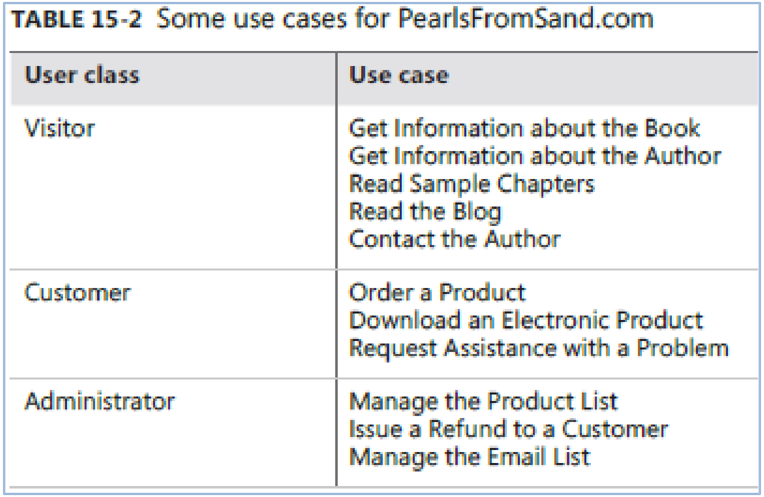
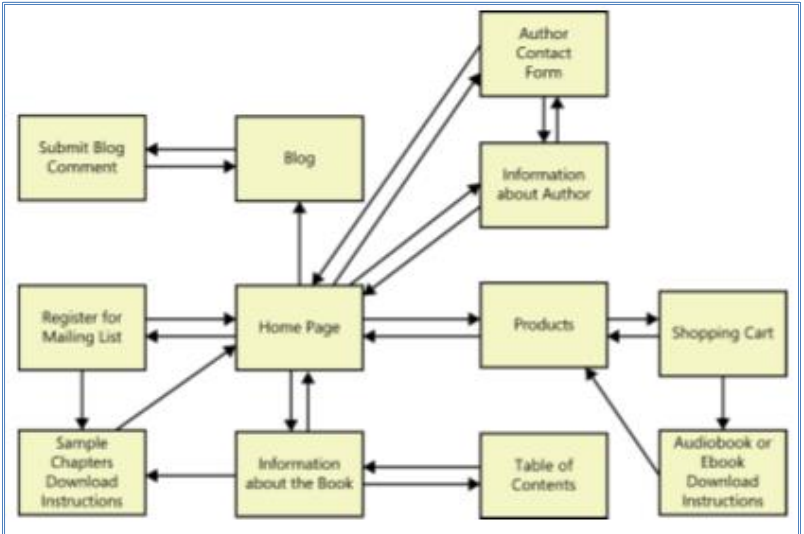


TABLE 15-2 Some use cases for PearlsFromSand.com

User class	Use case
Visitor	Get Information about the Book Get Information about the Author Read Sample Chapters Read the Blog Contact the Author
Customer	Order a Product Download an Electronic Product Request Assistance with a Problem
Administrator	Manage the Product List Issue a Refund to a Customer Manage the Email List


Author Contact Form
Submit Blog Comment
Blog
Information about Author
Register for Mailing List
Home Page
Products
Shopping Cart
Sample Chapters Download Instructions
Information about the Book
Table of Contents
Audiobook or Ebook Download Instructions
FIGURE 15-3 Partial dialog map for PearlsFromSand.com.

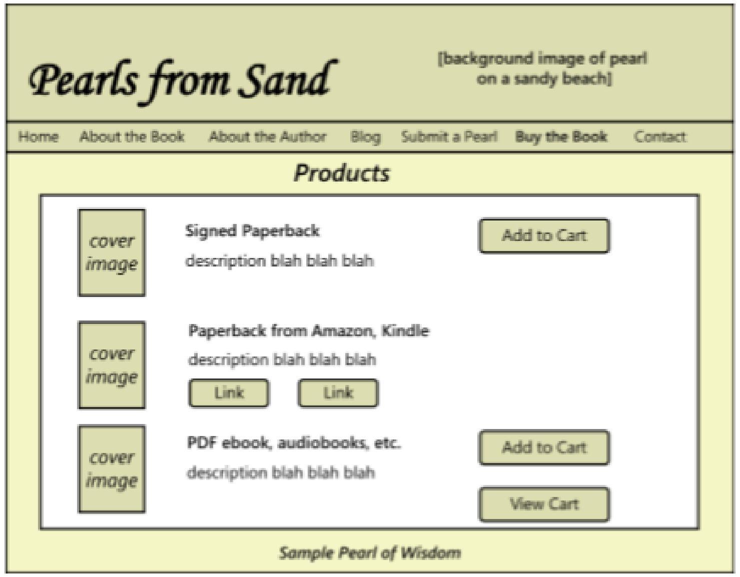
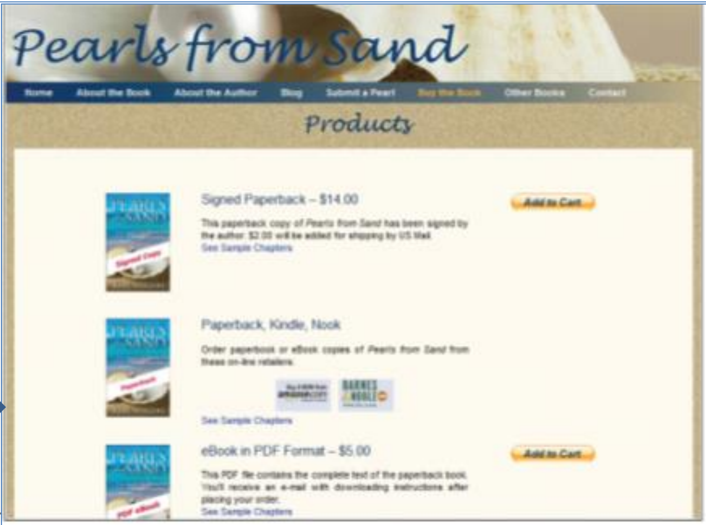


FIGURE 15-4 Sample wireframe of one page for PearlsFromSand.com.



This paperback copy of *Pearls from Sand* has been signed by the author. $2.00 will be added for shipping by US Mail.
See Sample Chapters

Order paperback or eBook copies of *Pearls from Sand* from these on-line retailers.
See Sample Chapters

This PDF file contains the complete text of the paperback book. You'll receive an e-mail with downloading instructions after placing your order.
See Sample Chapters

FIGURE 15-5 A final implemented page from PearlsFromSand.com.

Prototype evaluation

- Prototype evaluation is related to usability testing. You'll learn more by watching users work with the prototype than just by asking them to tell you what they think of it.
- You might ask the following questions:
  - Does the prototype implement the functionality in the way you expected?
  - What functionality is missing from the prototype?
  - Can you think of any possible error conditions that the prototype doesn't address?
  - Are any unnecessary functions present?
  - How logical and complete does the navigation seem to you?
  - Are there ways to simplify any of the tasks that require too many interaction steps?
  - Were you ever unsure of what to do next?

Risks of prototyping

Creating even a simple prototype costs time and money. Although prototyping reduces the risk of software project failure, it poses its own risks, some of which are explained in this section.

- Pressure to release the prototype:
  - The biggest risk is that a stakeholder will see a running throwaway prototype and conclude that the product is nearly completed. “This looks great. Can you just finish this up and give it to me?”
  - A throwaway prototype is never intended for production use, no matter how much it looks like the real thing. It is merely a model, a simulation, an experiment.
- Distraction by details:
  - Another risk of prototyping is that users become fixated on (gắn bó với) details about how the user interface will look and operate.
  - When working with real-looking prototypes, it’s easy for users to forget that they should be primarily concerned with conceptual issues at the requirements stage.

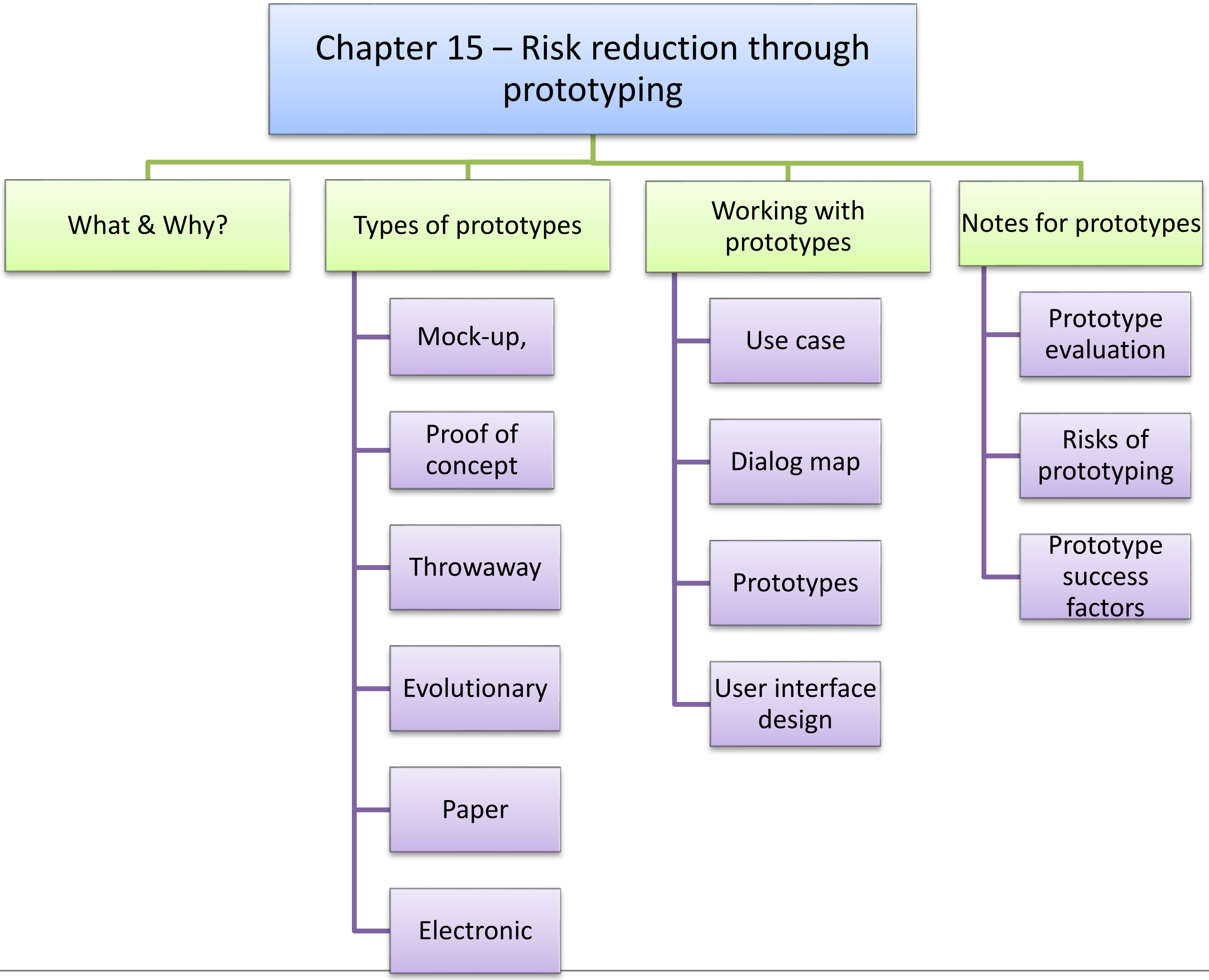
Risks of prototyping

- Unrealistic performance expectations:
  - A third risk is that users will infer (phỏng đoán) the expected performance of the final product from the prototype's performance.
  - You won't be evaluating a mock-up in the intended production environment, though.
- Investing excessive effort in prototypes:
  - Finally, beware of (hãy cẩn trọng) prototyping activities that consume so much effort that the development team runs out of time and is forced to deliver the prototype as the product or to rush (vội vã) through a haphazard (ko được hoạch định) product implementation.
  - This can happen when you are prototyping the whole solution rather than only the most uncertain, high-risk, or complex portions.

Prototyping success factors

- Software prototyping provides a powerful set of techniques that can minimize development schedules, ensure customer satisfaction, and produce high-quality products.
- To make prototyping an effective part of your requirements process, follow these guidelines:
  - Include prototyping tasks in your project plan. Schedule time and resources to develop, evaluate, and modify the prototypes.
  - State the purpose of each prototype before you build it, and explain what will happen with the outcome: either discard (or archive) the prototype, retaining the knowledge it provided, or build upon it to grow it into the ultimate solution.
  - Plan to develop multiple prototypes.
  - Create throwaway prototypes as quickly and cheaply as possible.
  - Don’t include input data validations, defensive coding techniques, error-handling code, or extensive code documentation in a throwaway prototype. It’s an unnecessary investment of effort that you’re just going to discard.
  - Don’t prototype requirements that you already understand, except to explore design alternatives.
  - Use plausible (hợp lý) data in prototype screen displays and reports.
  - Don’t expec.t a prototype to replace written requirement

Review chapter 15



THE END

THANK YOU!