BINUS UNIVERSITY

Course : COMP6176 / Human - Computer Interaction

Year : 2019

DESIGN, PROTOTYPING, AND CONSTRUCTION

SESSION 10

People
Innovation
Excellence

BINUS UNIVERSITY

People
Innovation
Excellence

LEARNING OUTCOMES

- LO 4: Design the user requirements with interaction styles

BINUS UNIVERSITY

People
Innovation
Excellence

OUTLINE

- Introduction
- Prototyping
- Conceptual Design
- Concrete Design
- Generating Prototypes
- Contruction

BINUS UNIVERSITY

People
Innovation
Excellence

INTRODUCTION

- There are two types of design :
  - Conceptual
  - Concrete
- For users to evaluate the design of an interactive product effectively, designers must prototype their ideas.
- There are two distinct circumstances for design :
  - Starting from the scratch
  - Modifying an existing product.

BINUS UNIVERSITY

People
Innovation
Excellence

PROTOTYPING

- **A Prototype** is one manifestation of a design that allows stakeholders to interact with it and to explore its suitability.
- A prototype can be anything from a paper-based storyboard through to a complex piece of software and from a cardboard mockup to a molded or pressed piece of metal.
- Why Prototype ?
  - A useful aid when discussing idea with stakeholders
  - A communication device among team members.
  - An effective way for designers to explore design ideas.

PROTOTYPING

- An example of prototype : A paper-based prototype of the design for a handheld device to help an autistic child communicate. (Fig 10.01)

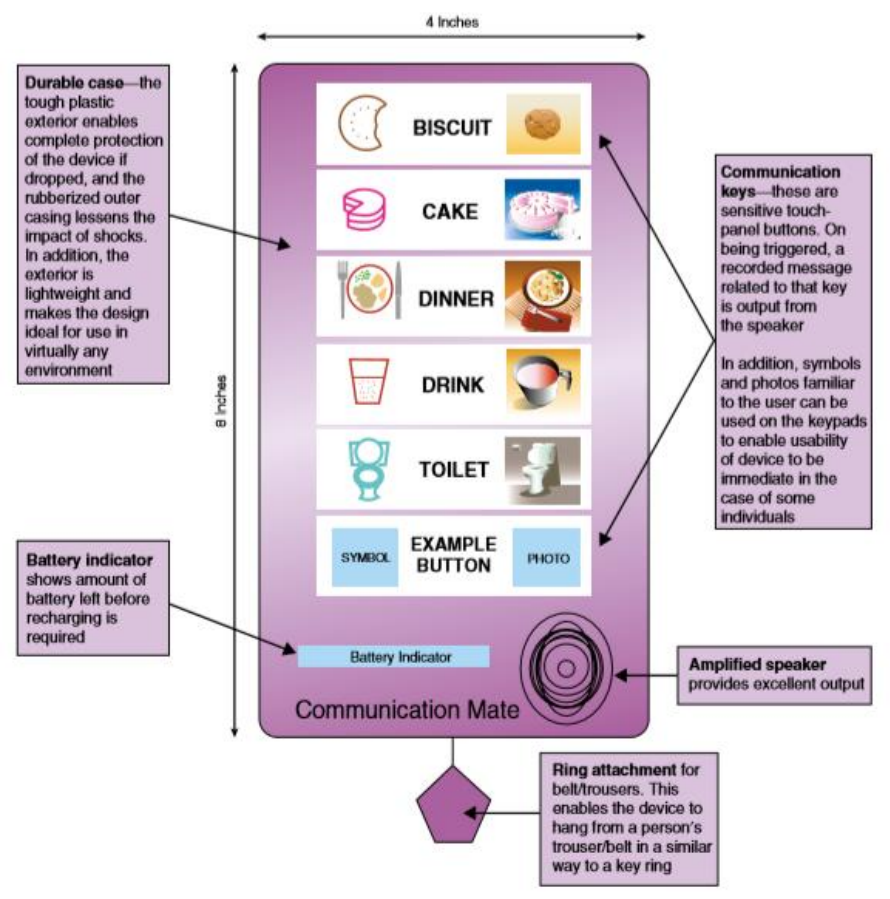

Figure 10.01 A paper-based prototype

BINUS UNIVERSITY

People
Innovation
Excellence

Low-Fidelity Prototyping

- A low-fidelity prototypes are useful because they tend to be simple, cheap, and quick to produce.
- Low-fidelity prototypes are never intended to be kept and integrated into the final product.
- They are for exploration only.

PROTOTYPING

- Example of Low-Fidelity Prototypes :
  - Storyboarding
    - Consist of a series of sketches showing how a user might progress through a task using the product under development. (See Fig 10.02)

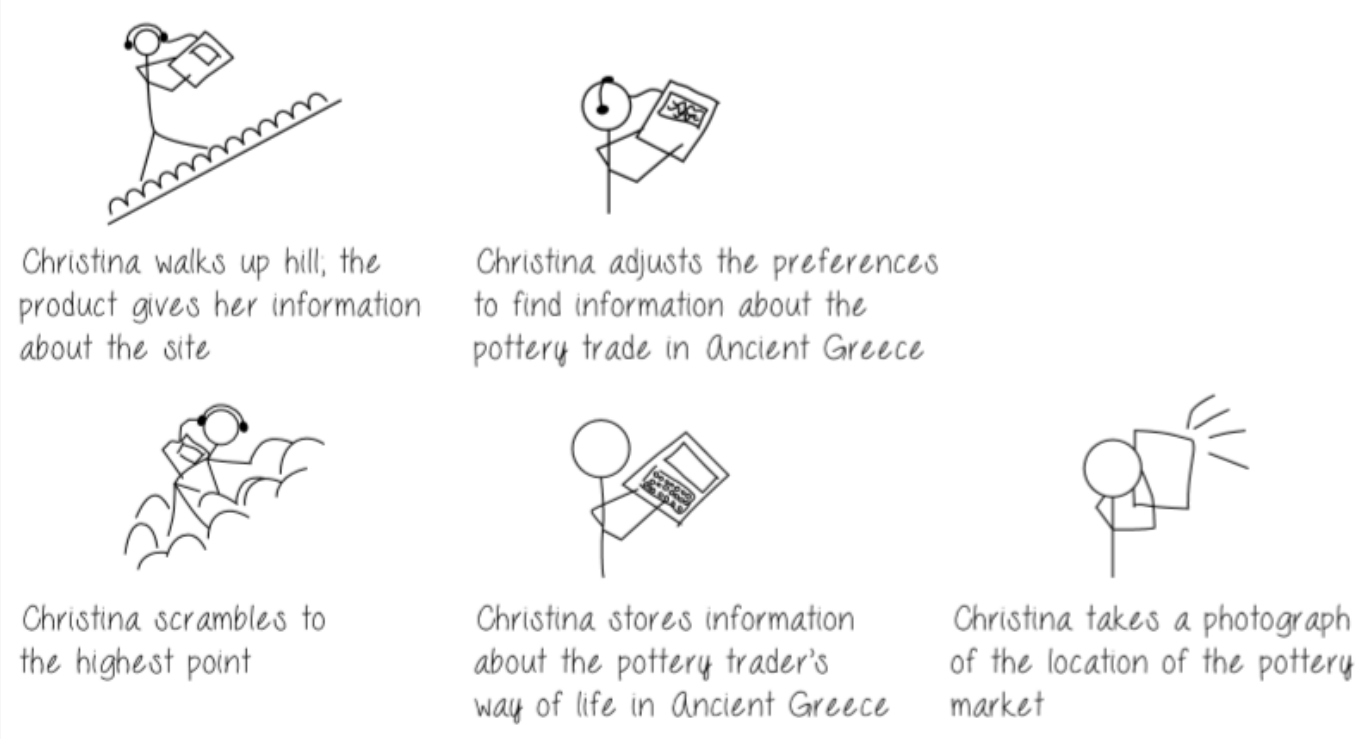

Figure 10.02 An example Storyboard

BINUS UNIVERSITY

People
Innovation
Excellence

PROTOTYPING

- Example of Low-Fidelity Prototypes :
  - Sketching :
    - Low-fidelity prototyping often relies on hand-drawn sketches
    - If you are sketching an interface design, then you need to draw various icons, dialog boxes, and so on.
  - Prototyping with index card
    - Using index cards (small pieces of cardboard about 3 x 5 inches) is a successful and simple way to prototype an interaction, it is used commonly when developing websites.

BINUS UNIVERSITY

People
Innovation
Excellence

PROTOTYPING

- Example of Low-Fidelity Prototypes :
  - Wizard of Oz
    - It assumes you have a software-based prototype.
    - The user interact with the software as though interacting with the product.

BINUS UNIVERSITY

People
Innovation
Excellence

High-Fidelity Prototyping

- It uses materials that you would expect to be in the final product and produces a prototype that looks like final thing.
- For example : a prototype of a software system developed in Visual Basic is higher fidelity than a paper-based mockup
- High-fidelity prototyping is useful for selling ideas to people and for testing out technical issues.
- The comparison between low and high-fidelity prototype will be show in table 10.01

Table 10.01 Low vs High-Fidelity Prototype

Type	Advantage	Disadvantages
Low-Fidelity Prototype	Lower Development cost	Limited error checking
	Evaluate multiple design concepts	Poor detailed specification to code to
	Useful communication device	Facilitator-driven
	Address screen layout issues	Limited utility after requirements established
	Useful for identifying market requirements	Limited usefulness for usability tests
	Proof-of-concept	Navigational and flow limitations

Table 10.01 Low vs High-Fidelity Prototype

Type	Advantage	Disadvantages
High-Fidelity Prototype	Complete functionality	More expensive to develop
	Fully interactive	Time-consuming to create.
	User-driven	Inefficient for proof-of-concept designs
	Clearly defines navigational scheme	Not effective for requirements gathering
	Use for exploration and test	
	Look and feel of final product	
	Serves as a living specification	
	Marketing and sales tool	

BINUS UNIVERSITY

People
Innovation
Excellence

PROTOTYPING (Your turn)

DILEMMA

Prototyping vs. Engineering

The compromises made when developing low-fidelity prototypes are evident, but compromises in high-fidelity prototypes are not so obvious. When a project is under pressure, it can become tempting to integrate a set of existing high-fidelity prototypes together to form the final product. Many hours will have been spent developing them, and evaluation with users has gone well. So, why throw it all away? Generating the final product this way will simply store up testing and maintenance problems for later (see Box 13.1 on technical debt). In short, this is likely to compromise the quality of the product, unless the prototypes have been built with sound engineering principles from the start.

On the other hand, if the device is an innovation, then being first to market with a "good enough" product may be more important for securing market position than having a very high-quality product that reaches the market two months after a competitor's product.

The dilemma arises in deciding how to treat high-fidelity prototypes—engineer them from the start or accept that they will be thrown away. ■

Moving from Requirements to First Design

- Conceptual design is concerned with transforming requirements into a conceptual model.
- Key guiding principles of conceptual design are :
  1. Keep an open mind but never forget the users and their context.
  2. Discuss ideas with other stakeholders as much as possible.
  3. Use low-fidelity prototyping to get rapid feedback.
  4. Iterate, iterate, and iterate

Developing an Initial Conceptual Model

- Approaches that help to develop an initial Conceptual Model, In particular consider :
  - How to choose interface metaphors that will help users understand the product?
  - Which interaction type(s) would best support the user's activities ?
  - Do different interface types suggest alternative design insights or options ?

Expanding the Initial Conceptual Design

- The set of initial conceptual model ideas must be thought through in more detail and expanded before prototyped or tested with users.
- This means deciding :
  - Which functions the product will perform ?
  - How are the functions related to each other ?
  - What information is needed ?
- Initial conceptual models may be captured in **wireframes** (a set of documents that show structure, content, and controls)

Getting Concrete

- There is no rigid border between conceptual design and physical design.
- Interaction design is inherently iterative and so some detail issue will come up during conceptual design, similarly during physical design it will be necessary to revisit decisions made during conceptual design.
- There are many aspects to the physical design of interactive products : Visual appearance such as color and graphics, icon design, button design, interface layout, choice of interaction devices, and so on.

BINUS UNIVERSITY

People
Innovation
Excellence

GENERATING PROTOTYPES

- How prototypes may be generated from the output of the requirements activity—producing a storyboard from a scenario and an index card-based prototype from a use case.
- Both of these are low-fidelity prototypes.
  - Generating Storyboards
  - Generating Card-based Prototypes

BINUS UNIVERSITY

People
Innovation
Excellence

GENERATING PROTOTYPES

Generating Storyboard

- A storyboard represents a sequence of actions or events that the user and the product go through to achieve a goal.
- A scenario is one story about how a product may be used to achieve that goal. A storyboard can be generated from a scenario by breaking the scenario into a series of steps that focus on interaction and creating one scene in the storyboard for each step.
- The purpose for doing this is twofold: first to produce a storyboard that can be used to get feedback from users and colleagues and second to prompt the design team to consider the scenario and the product's use in more detail.

BINUS UNIVERSITY

People
Innovation
Excellence

GENERATING PROTOTYPES

Generating Card-based Prototypes

- Card-based prototypes are commonly used to capture and explore elements of an interaction, such as dialog exchanges between the user and the product.
- The value of this kind of prototype lies in the fact that the interaction elements can be manipulated and moved around in order to simulate interaction with a user or to explore the user's end-to-end experience.
- For example, consider the use cases for the visa requirements aspect of the group travel organizer.
  - The first, less-detailed use case provides an overview of the interaction, while the second one is more detailed.
  - This second use case can be translated into cards as

BINUS UNIVERSITY

People
Innovation
Excellence

GENERATING PROTOTYPES

Generating Card-based Prototypes

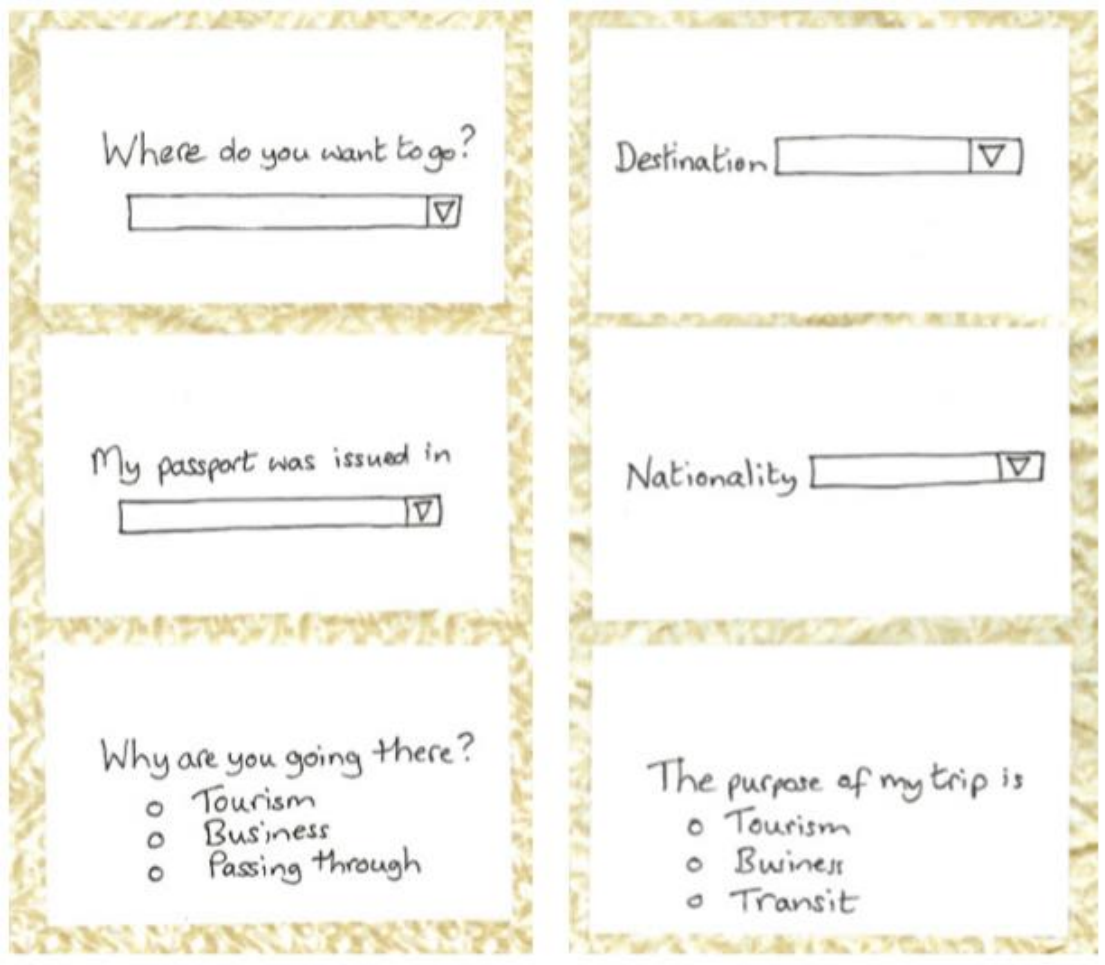

Figure 10.03 Cards 1–3 of a card-based prototype for the travel organizer

BINUS UNIVERSITY

People
Innovation
Excellence

CONSTRUCTION

- As prototyping and building alternatives progresses, development will focus more on putting together components and developing the final product.
- This may take the form of a physical product, such as a set of alarms, sensors, and lights, a piece of software, or both.
- There are two kinds of resources: physical computing kits and software development kits (SDKs).

Physical Computing

- Physical computing is concerned with how to build and code prototypes and devices using electronics.
- Specifically, it is the activity of “creating physical artifacts and giving them behaviors through a combination of building with physical materials, computer programming, and circuit building”.
- Typically, it involves designing things, using a printed circuit board (PCB), sensors (for instance push buttons, accelerometers, infrared, or temperature sensors) to detect states, and output devices (such as displays, motors, or buzzers) that cause some effect.

Software Development Kits

- A software development kit (SDK) is a package of programming tools and components that supports the development of applications for a specific platform, for example, for iOS on iPhone and iPad and for Android on mobile phone and tablet apps.
- Typically, an SDK includes an integrated development environment, documentation, drivers, and sample programming code to illustrate how to use the SDK components.
- For example, the availability of Microsoft's Kinect SDK has made the device's powerful gesture recognition and body motion tracking capabilities accessible.

BINUS UNIVERSITY

People
Innovation
Excellence

REFERENCES

- Interaction Design 5th Edition 2019, Chapter 12
- https://www.ece.ubc.ca/~leos/pdf/a230/notes/t1/ConceptDes.pdf
- http://webhome.cs.uvic.ca/~gtzan/seng310/lectures/design.pdf