Chapter 4

DESIGNING INTERACTIVE SYSTEM

Overview

- Design is a creative process concerned with bringing about something new.
- It is a social activity with social consequences.
- It is about conscious change and communication between designers and the people who will use the system.
- Different design disciplines have different methods and techniques for helping with this process.
- Approaches to and philosophies of design change over time.
- In mature disciplines, examples of good design are built up and people can study and reflect upon what makes a certain design great, good or awful.
- Different design disciplines have different constraints such as whether the designed object is “stand alone” or whether it has to fit in and live with legacy systems or conform to standards.

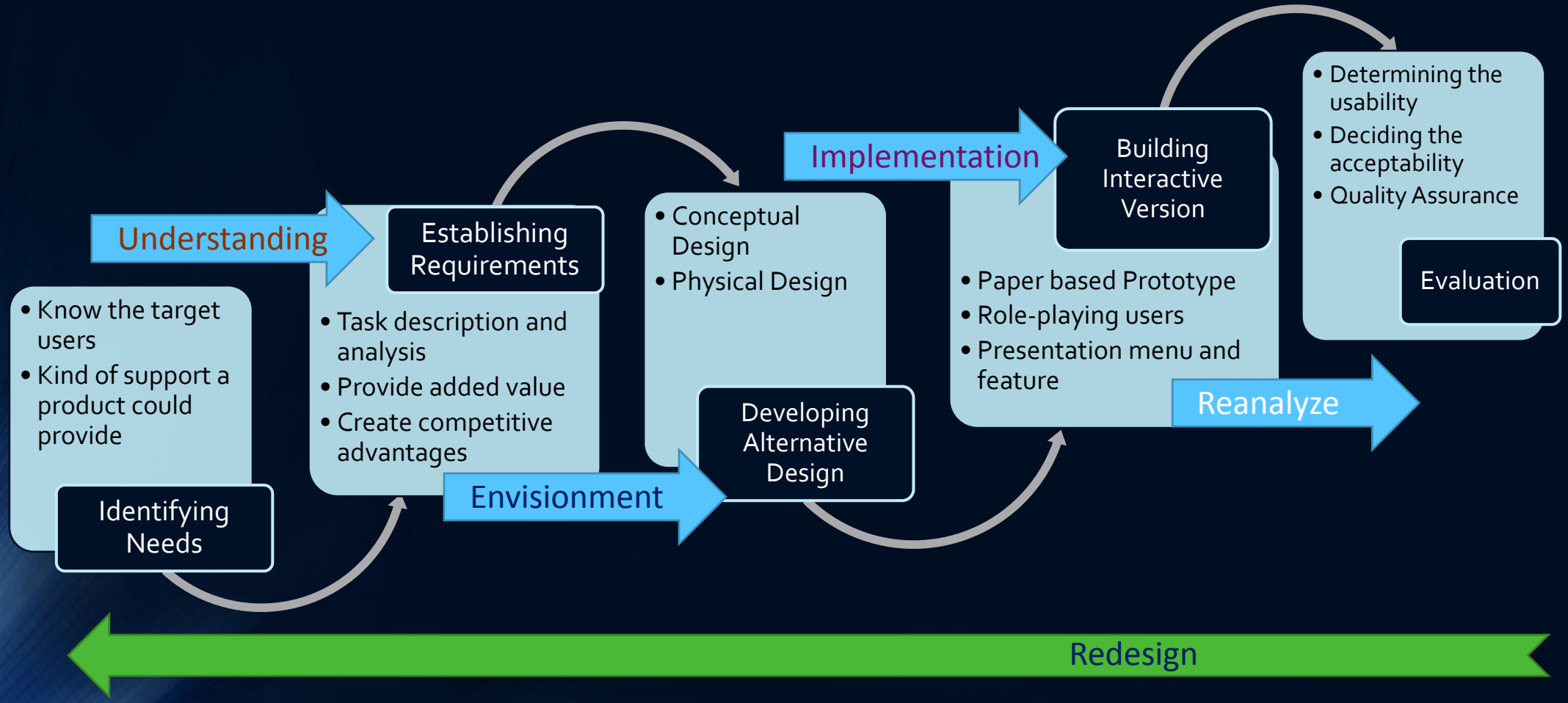
Aims

- Understand the nature of interactive systems design.
- Understand the processes involved in design: identification, establishment, develop, building and evaluation.
- Understand the centrality of evaluation in human-centered design.
- Understand the scenario-based design approach.
- Develop scenarios and personas.

Introduction

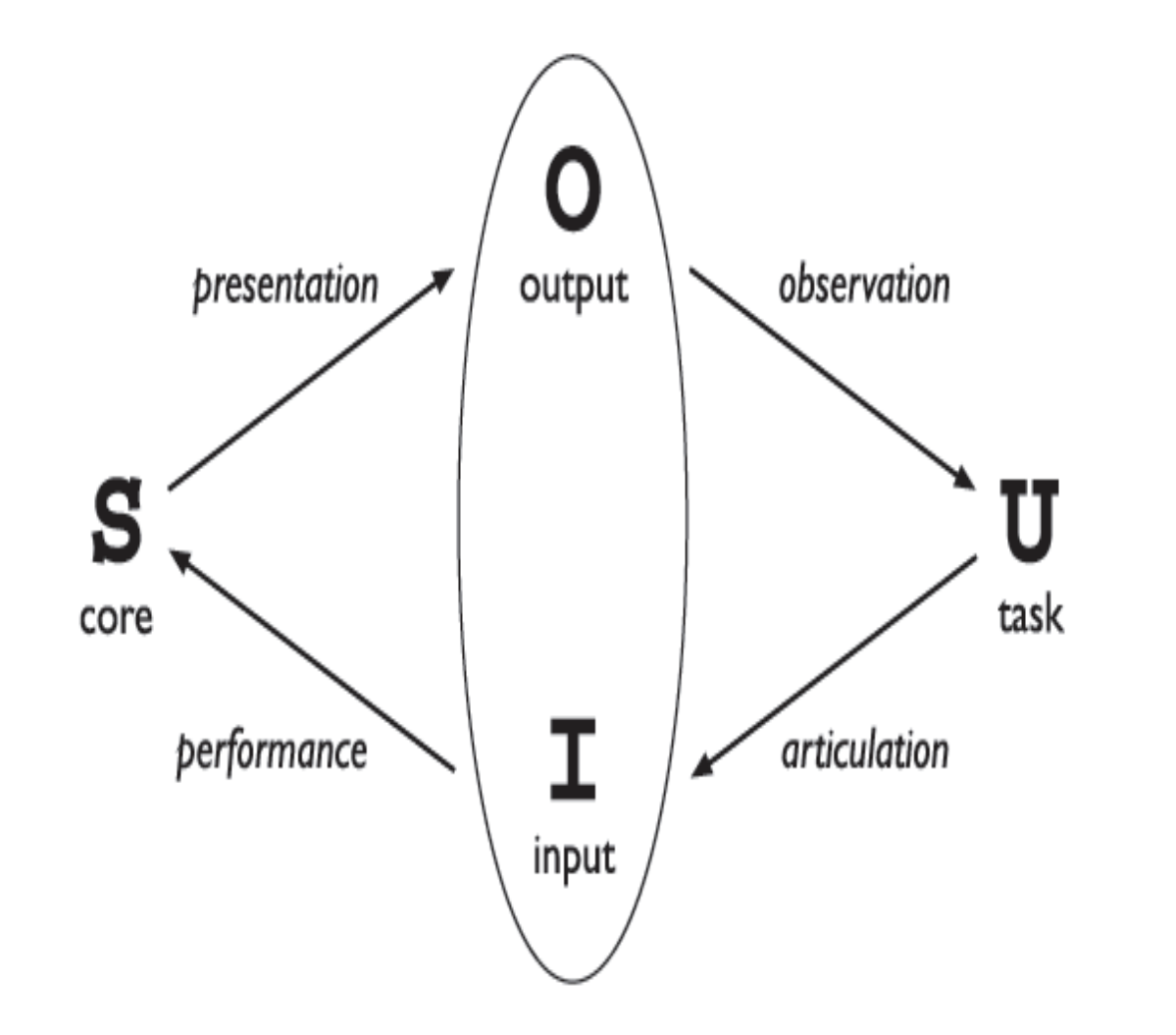
- Design is a practical and creative activity, the ultimate intent of which is to develop a product that helps its users achieve their goals.
- Developing a product must begin with gaining some understanding of what is required of it, but where do these requirements come from? Whom do you ask about them?
- Underlying good interaction design is the philosophy of user-center design, i.e., involving users throughout development, but who are the users? Will they know what they want or need even if we can find them to ask?
- For an innovative product, users are unlikely to be able to envision what is possible, so where do these idea come from?

Interaction Design Lifecycle



Interaction Design Process

Interaction Framework



- The interaction framework attempts a more realistic description of interaction by including the system explicitly, and breaks it into four main components: the *System*, the *User*, the *Input* and the *Output* while each component has its own language.
- For example, while writing a paper with some word-processing package, it is necessary at times to see both the immediate surrounding text where one is currently composing, say, the current paragraph, and a wider context within the whole paper that cannot be easily displayed on one screen.

Interaction Style

Interaction can be seen as a dialog between the computer and the user. The choice of interface style can have a profound effect on the nature of this dialog.

- Command line interface.
- Menus.
- Natural language.
- Question/answer and query dialog.
- Form-fills and spreadsheets.
- WIMP.
- Point and click.
- Three-dimensional interfaces.

Paradigm for Interaction

- The primary objective of an interactive system is to allow the user to achieve particular goals in some application domain, that is, the interactive system must be usable.
- As our machines have become more powerful, the key to increased usability has come from the creative and considered application of the technology to accommodate and augment the power of the human. Paradigms for interaction have for the most part been dependent upon technological advances and their creative application to enhance interaction.
- Often new paradigms have arisen through exploratory designs that have then been seen, after the fact, to have created a new base point for future design.

- Time sharing
- Video display units
- Programming toolkits
- Personal computing
- Window system and WIMP interface
- The metaphor
- Direct manipulation
- Language versus action
- Hypertext
- Multimodality
- Computer-supported cooperative work
- The world wide web
- Agent-based interface
- Ubiquitous computing
- Sensor-based and context awareness

Be visual and concrete

Love constraints to push limits

GIVE MEANING TO THE ORDINARY THING

Know your user and love them

Many ideas better than one perfect idea

Get feedback now and often

Balance your intuition with objective to satisfy user

Reframe the problem to get in detail

Work with people not like you

MINDSET

Think in metaphors and find out the analogies

Structure help the creativity to emerge

Seek out stories and always telling them

Avoid rush to find solution

Window to the World

The video used to promote this vehicle concept is a simulation filmed in static and controlled environments.
All health and safety requirements were met for the described conditions.
Toyota will never promote unsafe behaviors and will always encourage passengers to fasten their seatbelts.

Understanding

- Understanding is concerned with what the system has to do, what it has to be like and how it has to fit in with other things; with the requirements of the product, system or service.
- Designers need to research the range of people, activities and contexts relevant to the domain they are investigating so that they can understand the requirements of the system they are developing.
- They need to understand the opportunities and constraints provided by technologies.

Defining Requirements

- Requirements are generated through discussions and interactions with people who will use or be affected by the proposed system, the stakeholders.
- Requirements are also generated through observations of existing systems, research into similar systems, what people do now and what they would like to do.
- Requirements can be generated through working with people in focus groups, design workshops and so on, where different scenarios can be considered
- The aim is to collect and analyze the stories people have to tell. Requirements are essentially about understanding.

How to Get Good Requirements

1. Understand the users.
2. Know users' capacities and skills.
3. Know the tasks they carry out and their objectives.
4. Know under which conditions tasks are carried out.
5. Know which product(s) they use.
6. Know the constraints in the use of the product(s).

Stakeholders

- Stakeholders is a term that refers of all the people who will be affected by any systems that results from the process of interactive systems design.
- This includes the people who will finish up using the new system (sometimes called the "users"), but it also includes many other people.
- For example, the organization that the system is being designed for will probably have many people in it that will not be using the system, but will be affected by it as it might change their job.
- There may be stakeholders outside the organization such as government authorities that need to verify some procedures.
- An important part of the understanding process is to consider all the different stakeholders and how they might be affected, to decide who should be involved in discussions about the design.

Conceptual Design

- Conceptual design is about designing a system in the abstract, physical design is concerned with making things concrete.
- Conceptual design is about considering what information and functions are needed for the system to achieve its purpose.
- It is about deciding what someone will have to know to use the system.
- It is about finding a clear conceptualization of a design solution and how that conceptualization will be communicated to people (so that people will quickly develop a clear mental model).
- The key feature of conceptual design is to keep things abstract – focus on the "what" rather than the "how" – and to avoid making assumptions about how functions and information will be distributed.
- There is no clear-cut distinction between conceptual and physical design, but rather there are degrees of conceptuality

Type of Requirements

FUNCTIONAL REQUIREMENTS

- Functional requirements define specific behavior or functions.
- It concerns with what the system should be able to do and with the functional constraints of a system.
- The plan for implementing functional requirements is detailed in the system design.
- For example: a system may be required to present the user with a display of the number of records in a database.

NON-FUNCTIONAL REQUIREMENTS

- Non-functional requirements specify criteria that can be used to decide the operation of a system, rather than specific behaviors.
- In short, it specifies on how a system is supposed to be.
- The plan for implementing non-functional requirements is detailed in the system architecture.
- For example: a system should have sufficient network bandwidth to connect to the server.

Technique to Help with Conceptual Design

- Software engineers prefer modeling possible solutions with objects, relationships and “use cases” (a semi-formal scenario representation).
- Entity–relationship models are another popular conceptual modeling tool.
- Flow can be represented using dataflow diagrams and structure can be shown with structure charts.
- The conceptual design of a website, for example, will include a site map and a navigation structure.
- Many different conceptual models are used in the contextual inquiry method.

Rich Pictures

- A rich picture captures the main conceptual relationships between the main conceptual entities in a system – a model of the structure of a situation. Peter Checkland, who originated the soft systems approach, also emphasizes focusing on the key transformation of a system
- The principal stakeholders – customers, actors, system owners – should be identified.
- The designer should also consider the perspective from which an activity is being viewed as a system (the *Weltanschauung) and the environment in which the activities take place.*
- Checkland proposes the acronym CATWOE – customers, actors, transformation, *Weltanschauung, owners, environment – for these elements of a rich picture.*
- Most importantly the rich picture identifies the issues or concerns of the stakeholders, thus helping to focus attention on problems or potential design solutions.

Physical Design

- Physical design is concerned with how things are going to work and with detailing the look and feel of the product.
- Physical design is about structuring interactions into logical sequences and about clarifying and presenting the allocation of functions and knowledge between people and devices.
- The distinction between conceptual and physical design is very important.
- The conceptual design relates to the overall purpose of the whole interactive system.
- Between the people and the technologies there has to be enough knowledge and ability to achieve the purpose.
- Physical design is concerned with taking this abstract representation and translating it into concrete designs.
- On one side this means requirements for hardware and software and on the other side it defines the knowledge, tasks and activities that people will be required to do.
- The 3 components are: operational design, representational design, and design of interactions.

Operational Design

- *Operational design is concerned with specifying how everything works and how content is structured and stored.*
- Taking a functional view of an activity means focusing on processes and on the movement, or flow, of things through a system.
- *Events are occurrences that cause, or trigger, some other functions to be undertaken.*
- Sometimes these arise from outside the system under consideration and sometimes they arise as a result of doing something else.
- For example, some activity might be triggered on a particular day or at a particular time
- Another might be triggered by the arrival of a person or document.

Representational Design

- *Representational design is concerned with fixing on colors, shapes, sizes and information layout.*
- It is concerned with style and aesthetics and is particularly important for issues such as the attitudes and feelings of people, but also for the efficient retrieval of information.
- Style concerns the overall "look and feel" of the system.
- Does it appear old and "clunky" or is slick, smooth and modern? What mood and feelings does the design engender?
- For example, most Microsoft products engender an "office" and "work" mood, serious rather than playful.
- Many other systems aim to make the interaction engaging, some aim to make it challenging and others entertaining.
- In multimedia and games applications this is particularly important.

Interaction Design

- *Interaction design, in this context, is concerned with the allocation of functions to human agency or to technology and with the structuring and sequencing of the interactions.*
- Allocation of functions has a significant impact on how easy and enjoyable a system is to use.
- Designers create tasks for people by the way they allocate functions.
- For example, consider the activity of making a phone call. Conceptually speaking, certain functions are necessary: indicate a desire to make a phone call, connect to the network, enter the phone number, make connection.

Interaction Design (cont.)

- Years ago a telephone exchange was staffed by people and it was these people who made connections by physically putting wires into connectors.
- In the days of wired phones, picking up the receiver indicated the desire to make a call, the full number had to be dialed in and then the telephone exchange would automatically make the connections.
- Nowadays a person just has to press the connect button on a cellular phone, choose someone's name from the phone's address book and the technology does the rest.
- The allocation of knowledge and activities between people and technologies is a significant part of how experiences change over time.

Five key problems of design (from Carroll)

- The external factors that constrain design such as time constraints, lack of resources, having to fit in with existing designs and so on.
- Design moves have many effects and create many possibilities, i.e. a single design decision can have an impact in many areas and these need to be explored and evaluated.
- How scientific knowledge and generic solutions lag behind specific situations. This point concerns generalities. In other design disciplines, general design solutions to general design problems have evolved over the years. In interactive systems design this does not happen because the technology changes as soon as, or even before, general solutions have been discovered.
- The importance of reflection and action in design.
- The slippery nature of design problems.

Envisionment

- Designs need to be visualized both to help designers clarify their own ideas and to enable people to evaluate them.
- Envisionment is concerned with finding appropriate media in which to render design ideas.
- The medium needs to be appropriate for the stage of the process, the audience, the resources available and the questions that the designer is trying to answer.
- There are many techniques for envisionment, but they include any way in which abstract ideas can be brought to life.
- Sketches "on the back of an envelope", fully functioning prototypes and cardboard mock-ups are just some of the methods used.
- Scenarios, sometimes represented in pictorial form as storyboards, are an essential part of prototyping and envisionment.
- They provide a way of working through a design idea so that the key issues stand out.

Evaluation

- Evaluation is tightly coupled with envisionment because the nature of the representation used will affect what can be evaluated.
- The evaluation criteria will also depend on who is able to use the representation.
- Any of the other design activities will be followed by an evaluation.
- Sometimes this is simply the designer checking through to make sure something is complete and correct.
- It could be a list of requirements or a high-level design brief that is sent to a client, an abstract conceptual model that is discussed with a colleague, or a formal evaluation of a functional prototype by the future system users.
- Techniques for evaluation are many and various depending once again on the circumstances.
- The important thing to keep in mind is that the technique used must be appropriate for the nature of the representation, the questions being asked and the people involved in the evaluation.

Implementation

- Ultimately things have to be engineered and software has to be written and tested.
- Databases have to be designed and populated and programs have to be validated.
- The whole system needs to be checked to ensure that it meets the requirements until finally the system can be formally "launched" and signed off as finished.
- Clients will often want extra features when they see a system nearing completion, but these will have to be costed and paid for.
- On the other hand the developers need to ensure that their system really does meet the specification and does not contain any "bugs".
- If interactive systems designers were architects they would have well understood methods and conventions for specifying the results of the design process.
- They would produce various blueprints from different elevations and engineering specifications for particular aspects of the design.
- In interactive systems design there are a variety of formal, semi-formal and informal methods of specification.

Agile Development

- Over the last few years there has been a move away from large software engineering approaches to the development of interactive systems towards agile development methods.
- These are designed to produce effective systems of high quality that are fit for purpose, but without the huge overhead associated with the planning and documentation of a large IT (information Technology) project.
- There are a number of competing methods, but probably the best known comes from DSDM, a not-for-profit consortium of software development companies.
- Their system, called Atern, is fully documented showing how software can be developed in small teams.
- There is still plenty of debate about how well these methods, such as extreme programming) , fit in with human-center approaches, but many of the methods do promote participation between developers and stakeholders

Functionality is not enough!

- For an interface to be a success, it must provide the right functionality, at the right time, in the right place, and in the right form from the user's point of view.
- Such interfaces are called usable.
- Example: if we are designing an ATM, we should be able to justify each user action:
  - Insert card?
  - Enter PIN?
  - Press Quick Cash key?
  - Press Okay?
  - Remove card?
  - Remove money?
  - Remove receipt?

Methodologies vs Methods

- Methodology is the study of how research is done, how we find out about things, and how knowledge is gained. In other words, methodology is about the principles that guide our research practices. Methodology therefore explains why we're using certain methods or tools in our research.
- Research methods are the tools, techniques or processes that we use in our research. These might be, for example, surveys, interviews, photovoice, or participant observation. Methods and how they are used are shaped by methodology.

Personas

- Personas are concrete representations of the different types of people that the system or service is being designed for.
- Persona can be used as method for design in initial stage or as design approach for whole stages.
- Personas should have a name, some background and, importantly, some goals and aspirations.
- Alan Cooper introduced the idea of personas in the late 1990s and they have gained rapid acceptance as a way of capturing knowledge about the people the system or service is target at.
- In the latest edition of his book he links Personas very closely with his ideas of goal-directed design.
- Personas want to be able to do things using your system.
- They want to achieve their aims, they want to undertake meaningful activities using the system that the designer will produce.

Using Personas

- Designers need to recognize that they are not designing for themselves.
- Designers create personas so that they can envisage who they are designing for.
- They create personas so that they can put themselves in other people's shoes.
- As any new system is likely to be used by different types of people, it is important to develop several different personas.
- Such a diverse group of people have very different goals, and aspirations and differ in all manner of ways physically, psychologically and in terms of the usage they would make of the site.

Personas (example)

Background:
HRWeb is a Human Resources (hr) Management web-based application used by Acme Insurance. HRWeb handles hr management such as employee records, compensation, and benefits management. User Groups for HRWeb include: hr managers, hr specialists, hr administrative assistants, non-hr managers, non-hr administrative assistants, employees, retired employees.

P1: Roger Thompson retired 3 years ago at age 62, after working for 35 years in the Underwriting department of Acme Insurance. At 65 years old, Roger is extremely active. He golfs twice a week in the summer and swims laps three times a week in the winter. He is proud of his garden and spends an hour or two each day maintaining his lawn and flower beds. Roger is also an avid fisherman.

Roger has age-related macular degeneration, which, in Roger's case, causes blurred central vision in his left eye. His vision has gotten progressively worse over the past two years. When reading or doing crossword puzzles, he needs brighter light and sometimes uses a magnifier. Roger has a slight tremor in his right hand. He is happy to report that this does not affect his golf game or his gardening. He notices that some activities, such as writing and baiting his hook for fishing, are affected by the tremor. When he uses the computer he experiences some difficulty in using the mouse, especially if the button or link he's trying to hit is fairly small.

Scenarios

- Scenarios are stories about people undertaking activities in contexts using technologies.
- They appear in a variety of forms throughout interactive systems design and are a key component of many approaches to design.
- Scenarios have been used in software engineering, interactive systems design and human–computer interaction work for many years.
- More recently scenario-based design has emerged as an important approach to the design of interactive systems in the twenty-first century
- One of the main proponents of scenario-based design is John Carroll and his book *Making Use remains an excellent introduction to the philosophy underlying the approach.*
- In it he illustrates how scenarios are used to deal with the inherent difficulty of doing design.

Motivations

- One central theme of the explorations concerned the motivational approaches that would be suitable for different scenarios and personas.
- The Sandy persona would need more encouragement and persuasion to exercise than the Mari persona perhaps by preventing a recorded television program from being shown until training is completed.
- Another aspect, concerning social networking, was explored through the Bjorn persona.
- Thus the personas were developed to reflect particular design issues and values.
- The whole issue of persuasion technologies is a difficult one for interaction design.

Captology

- The basic aim of captology is to persuade people to do things they otherwise would not do.
- Who are we, as designers, to persuade people to do something they don't want to do.
- However, we can see examples, such as the Sandy persona, where persuading him to exercise is for his own good.
- We also need to persuade people to take precautions if things are dangerous.
- I am quite happy that a software system persuaded me to save my work before the system crashes (on the other hand why did the system not just save it for me?).
- Persuasion is "a non-coercive attempt to change attitudes or behaviors of people" (Fogg, Cuellar and Danielson, 2008).
- However, if this is persuading me to buy something that I cannot afford, then it is not good at all, whether it is non-coercive or not.
- This is an area of HCI where ethics and values must be taken seriously.

Photopal

- In another exploration we looked at the concept of a companion to deal with digital photos.
- Such a companion would be functional in helping organize, edit and share photos, but would also be a conversational partner.
- We envisaged a companion that could discuss photos with its owner and perhaps reminisce about events and people.
- One feature of this scenario was to explore different modalities for the companion.
- The interaction employs both speech and touch depending on the activity being undertaken.
- For example, it is much quicker to specify specific search parameters through speech than by typing or clicking a series of check
- However, when it comes to flicking through the search generated group or applying certain other editorial functional tasks such as scaling and cropping, touch becomes the more natural interaction.
- However, for specific categorical edits speech may be best, for example “make this image 4 by 6 inches and print”.
- The true power of the interaction experience comes from the considered use of both in conjunction.

Environment

- In another scenario we were looking at environmental influence on the interaction.
- Small displays (e.g. digital photoframes) have a more limited touch capability than a larger display
- Using a display that is simply too far from the person to be touched.
- This in many ways most fairly reflect the current living room environment.
- In such a situation physical gesture becomes an appropriate option, either by using ones hands or by wielding an object, such as is used in the Nintendo Wii games console.
- This allows for parameters such as speed, direction and shape of movement.

Using Scenario throughout Design

- Scenarios (and their associated personas) are a core technique for interactive systems design.
- They are useful in understanding, envisioning, evaluation, and both conceptual and physical design: the four key stages of interactive system design
- We distinguish four different types of scenario: stories, conceptual scenarios, concrete scenarios and use cases.
- Stories are the real-world experiences of people.
- Conceptual scenarios are more abstract descriptions in which some details have been stripped away.
- Concrete scenarios are generated from abstract scenarios by adding specific design decisions and technologies and once completed these can be represented as use cases.
- Use cases are formal descriptions that can be given to programmers.

Scenario at Different Stages

- At different stages of the design process, scenarios are helpful in understanding current practice and any problems or difficulties that people may be having, in generating and testing ideas, in documenting and communicating ideas to others and in evaluating designs.
- Many stories will be represented by a few conceptual scenarios. However, each conceptual scenario may generate many concrete scenarios.
- Several concrete scenarios will be represented by a single use case.
- Designers abstract from the details of stories to arrive at conceptual scenarios.
- They specify design constraints on conceptual scenarios to arrive at concrete scenarios.
- Finally they formalize the design ideas as use cases.

Gathering Data

When working in a large design team, it is useful to accompany scenarios by real data.

- This means that different team members can share concrete examples and use these as a focus of discussion.
- Another key feature of writing scenarios is to think hard about the assumptions that are being made: to make assumptions explicit or deliberately avoid making things explicit in order to provoke debate.
- In particular the use of personas can help to focus on specific issues.
- For example an elderly woman with arthritis might be one of the personas, thus foregrounding issues of access and the physically impaired interacting with technology.
- Finally with these scenarios it is important to provide a very rich context.
- The guiding principles for scenario writing are people, activities, contexts and technologies.
- The following example is grounded in a concrete example and in a specific context, this is still quite conceptual in that it is used to generate ideas and designs.

Documenting Scenarios

- Scenarios can become messy, so in order to control the scenarios a structure is needed.
- For each scenario the designer lists the different people who are involved, the activities they are undertaking, the contexts of those activities and the technologies that are being used.
- We also structure scenario descriptions. Each scenario should be given an introduction.
- The history and authorship can be recorded, along with a description of how the scenario generalizes (across which domains) and the rationale for the scenario.
- Each paragraph of each scenario should be numbered for ease of reference and endnotes included where particular design issues are raised.
- Endnotes are particularly useful in documenting issues raised during the development of the scenario. They are a way of capturing the claims being made about the scenarios
- Examples of relevant data and media should be collected.

Cross Reference

- Another aspect of documentation that is useful is to cross-reference the stories to the conceptual scenarios, through the concrete examples and finally to the use cases.
- Other researchers have suggested similar ideas that capture the multiple views necessary to see how scenarios and claims work together to provide a rich understanding of how a design finished as it did.

Stories

- Stories are the real-world experiences, ideas, anecdotes and knowledge of people.
- These may be captured in any form and comprise small snippets of activities and the contexts in which they occur.
- This could include videos of people engaged in an activity, diary entries, photographs, documents, the results of observations and interviews and so on.
- People's stories are rich in context.
- Stories also capture many seemingly trivial details that are usually left out if people are asked to provide more formal representations of what they do.

Stories (Example)

I needed to make an appointment for Kirsty, my little one. It wasn't urgent – she had been having a lot of bad ear-ache every time she had a cold – but I did want to see Dr. Fox since she's so good with the children. And of course ideally it had to be when Kirsty was out of school and I could take time off work. I rang the surgery and the receptionist told me that the next appointment for Dr. Fox was the next Tuesday afternoon. That was no good since Tuesday is one of my really busy days so I asked when the next one was. The receptionist said Thursday morning. That meant making Kirsty late for school but I agreed because they sounded very busy – the other phone kept ringing in the background – and I was in a hurry myself. It was difficult to suggest a better time without knowing which appointments were still free.

Storyboards

- Storyboarding is a technique taken from film making – using a simple cartoon-like structure, key moments from the interactive experience are represented.
- The advantage of storyboarding is that it allows you to get a feel for the "flow" of the experience.
- It is also a very economical way of representing the design – a single page can hold 6–8 "scenes".
- It is often helpful to sketch out a storyboard based around a concrete scenario.
- The two together are very helpful in working through design ideas with customers.

Storyboards (cont..)

- Traditional storyboarding. A storyboard for a film would usually have some notes attached to each scene expanding on what will happen – this helps overcome the limitations of representing a dynamic experience in a static medium. For interactive systems, notes below each sketch usually contain the relevant steps from a scenario, and the sketches themselves are annotated to indicate interactive behavior. This is the most usual form of storyboard if there is not a strongly multimedia flavor to the application.
- Scored storyboards. If the application has a lot of motion graphics the storyboard can be annotated – a sketch is annotated with appropriate notation and notes about, for example, type, colors, images, sound and other issues are attached underneath.
- Text-only storyboards. These are useful if the application has a lot of very complex sequences. You can specify what images appear, what text accompanies them, any accompanying media, general notes about tone, flow, etc.

Mood Boards

- Mood boards are widely used in advertising and interior design.
- Quite simply you gather visual stimuli that capture something of how you feel about the design – photographs and other images, colors, textures, shapes, headlines from newspapers or magazines, quotations from people, pieces of fabric and so on.
- Attach the stimuli to a pinboard. Even thinking about their arrangement can stimulate ideas.
- You can put pages from websites you like on mood boards.
- If you use Blu-tack or something similar then you can add and delete items as your thinking changes. The rule with mood boards is that "anything goes".
- The point of the board is not to formally represent some aspect of the design, simply to act as inspiration – perhaps promoting a particular line of thought, or providing inspiration for a color scheme.

Navigation Maps

- Navigation is a key feature for many systems. It focuses on how people move through the site or application and on how people will experience the site.
- Each page in the site, or location in the application, is represented with a box or heading and every page that can be accessed from that page should flow from it.
- A useful tip is to put in all flows possible (i.e. back and forwards from a page) as this will highlight sections where people might get stranded.
- Navigation maps can usefully be redrawn many times through the project lifecycle, as poor navigational structure is one of the main reasons people turn off a website, for example.
- The maps can be used with scenarios to "walk through" particular activities and are a very good way of spotting poor aspects of design such as "orphan pages" (pages which are not accessible) or dead ends.
- Navigation is important in all manner of applications and products, not just websites. More formal and more fully annotated maps can also be developed. Arrows can be added to lines if the direction of a link is important.

Conceptual Scenarios

Conceptual scenarios are more abstract than stories.

- Much of the context is stripped away during the process of abstraction and similar stories are combined together.
- Conceptual scenarios are particularly useful for generating design ideas and for *understanding the requirements of the system.*
- Once the designer has accumulated a collection of stories, common elements will start to emerge.
- In this case a number of stories such as the one above result in the conceptual scenario below describing some requirements for a computerized appointments system.

Conceptual Scenario (Abstraction)

Booking an appointment: People with any degree of basic computer skills will be able to contact the doctors' surgery at any time via the Internet and see the times which are free for each doctor. They can book a time and receive confirmation of the appointment.

- As you can see, at this stage, there is little or no specification of precise technologies or how the functions will be provided.
- The scenario could be made more abstract by not specifying that the Internet should be used or more concrete (that is less abstract) by specifying that the booking should be made from a computer rather than a mobile phone.
- Finding an appropriate level of abstraction at which to describe things for a given purpose is a key skill of the designer.

Concrete Scenarios

Each conceptual scenario may generate lots of concrete scenarios.

- When designers are working on a particular problem or issue they will often identify some feature that applies only under certain circumstances.
- At this point they may develop a more specific elaboration of the scenario and link it to the original.
- Thus one reasonably abstract scenario may spawn several more concrete elaborations which are useful for exploring particular issues.
- Notes can be added to scenarios that draw attention to possible design features and problems.
- Concrete scenarios also begin to dictate a particular interface design and a particular allocation of functions between people and devices.
- Concrete scenarios are particularly useful for prototyping and envisioning design ideas and for evaluation because they are more prescriptive about some aspects of the technology.
- However, there is not a clean break between conceptual and concrete scenarios.
- The more specific the scenario is about some aspects, the more concrete it is.

Concrete Scenarios (Example)

In the example below, decisions have now been taken concerning drop-down menus, the fact that the next two weeks details are to be shown, and so on. However, the notes following the scenario show that there are many design decisions still to be taken.

- Andy Dalreach needs a doctor's appointment for his young daughter Kirsty in the next week or so. The appointment needs to be outside school-time and Andy's core working hours, and ideally with Dr. Fox, who is the children's specialist. Andy uses a PC and the Internet at work, so has no difficulty in running up the appointments booking system. He logs in [1] and from a series of drop-down boxes, chooses to have free times for Dr. Fox [2] displayed for the next two weeks [the scenario would continue to describe how Andy books the appointment and receives confirmation]

Notes to booking an appointment:

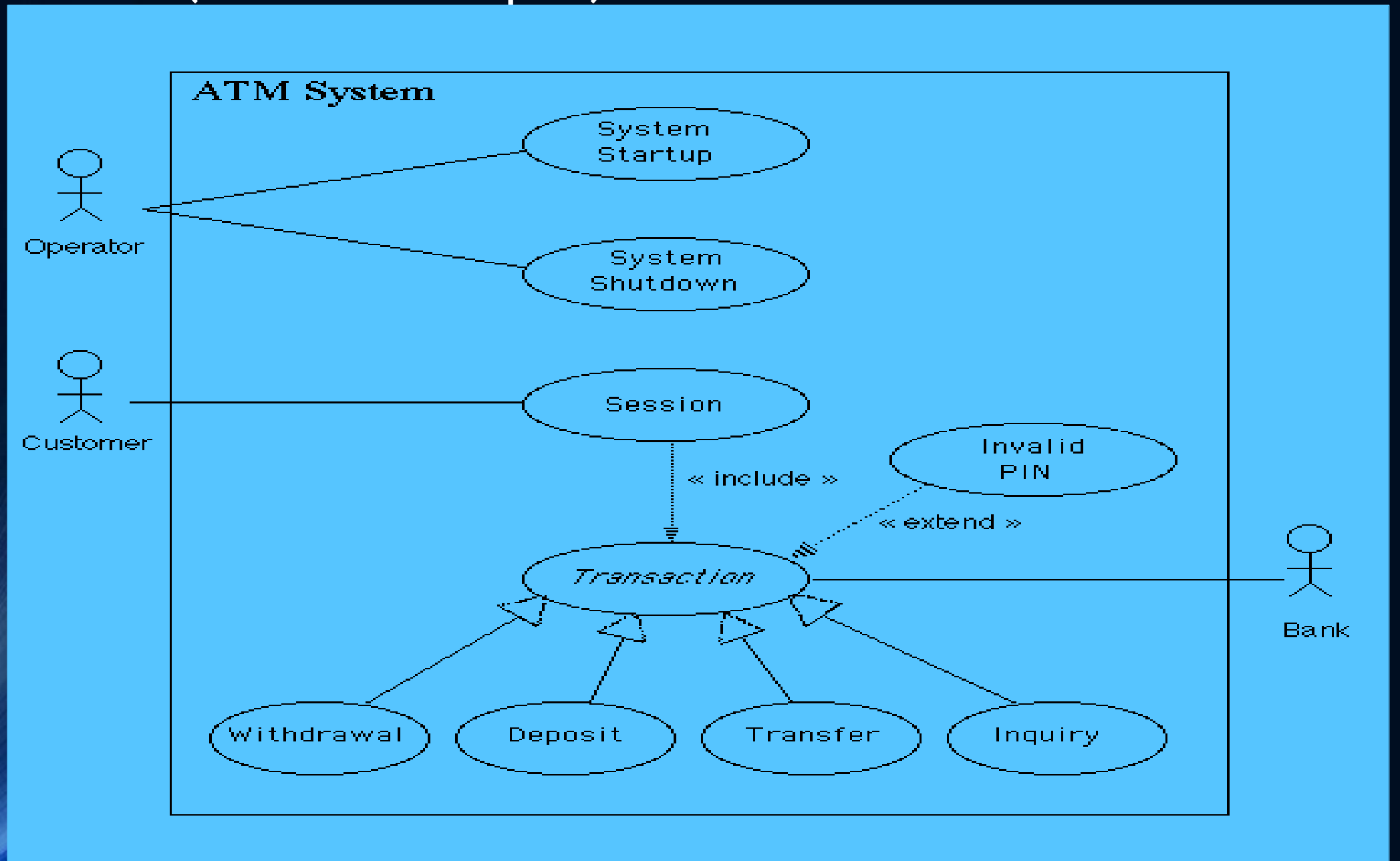
1. Is logging in necessary? Probably, to discourage bogus access to the system, but check with the surgery.

2. Free times can be organized by doctor, by time of day, or by next available time. Drop-down boxes will save screen

Use Case

- A use case describes the interaction between people (or other actors) and devices.
- It is a case of how the system is used and hence needs to describe what people do and what the system does.
- Each use case covers many slight variations in circumstances – many concrete scenarios
- Before use cases can be specified, tasks and functions have to be allocated to humans or to the device.
- The specification of use cases both informs and is informed by the task/function allocation process.
- A set of use cases can be produced which specify the complete functionality of the system and the interactions that will occur.
- There are a number of different ways of representing use

Use Case (ATM Example)



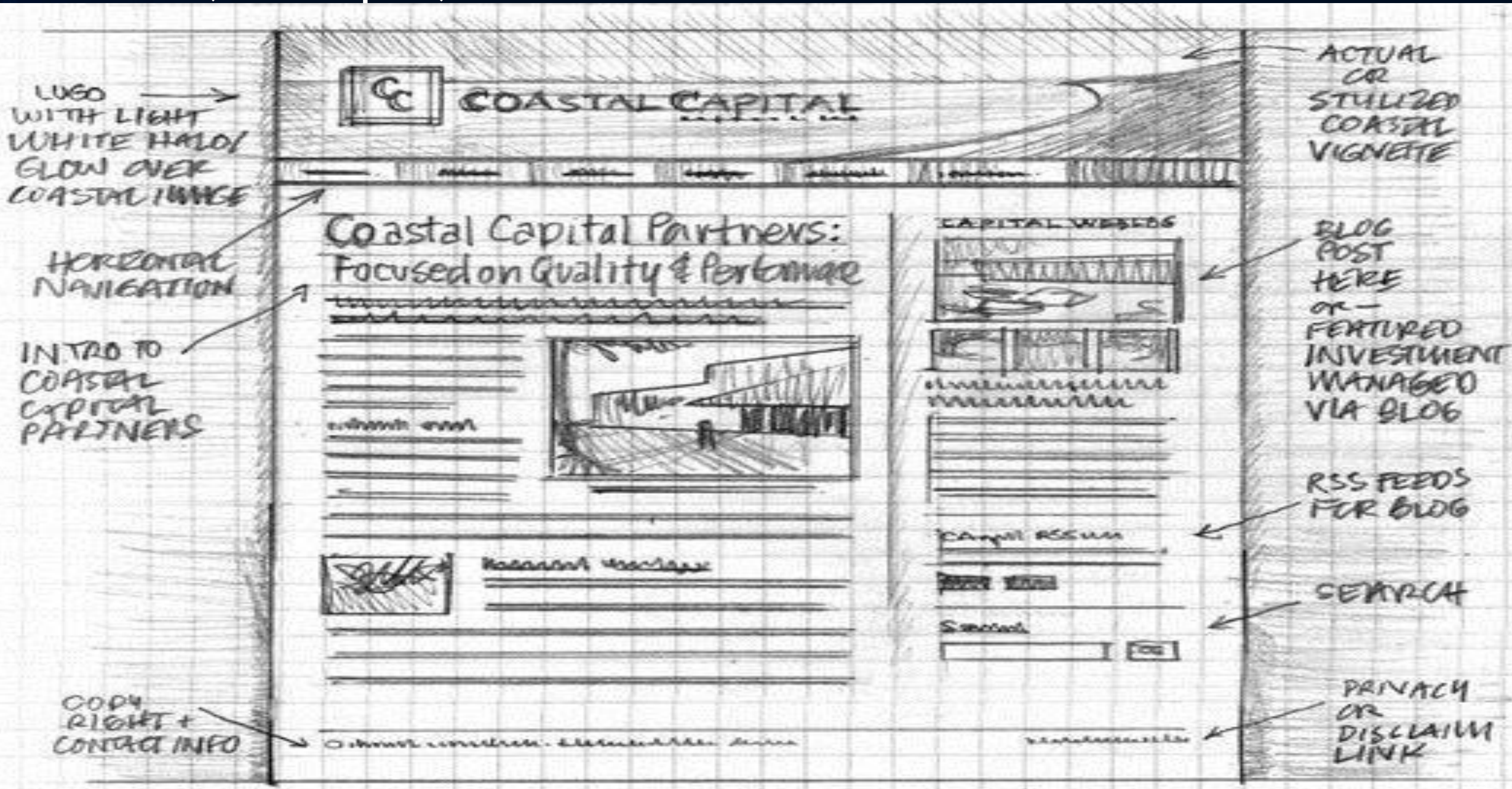
Sketch

- A rough or unfinished drawing or painting, often made to assist in making a more finished picture in any drawing medium.
- A sketch may serve a number of purposes: it might record something that the designer sees, it might record or develop an idea for later use or it might be used as a quick way of graphically demonstrating an image, idea or principle.
- Sketching with aimless abandon can create the most valuable images and ideas.
- Sketches are often just preliminary suggestions of what a project can come to be as it develops throughout the design process and workflow.
- Sketches can be a series of lines and shapes and words. The most common things you will see in a sketchbook are points, lines, angles and arcs, squares and rectangles, circles and words.
- Why sketch is the great method to design?
  - ❑ Your first idea is rarely your best.
  - ❑ Sketching is fast, rough, and dirty.
  - ❑ It separates concepts from details.
  - ❑ Sketching is for everyone and fun.
  - ❑ variety of things as creative process.

Sketch (cont.)

- Regardless of where your final product will be published, sketching can help you better conceptualize, create and collaborate on design projects. The art of the sketch is simple and one of the easiest to use brainstorming techniques.
- Sketching can show the evolution of a project and help you go back to early ideas without having to keep a bunch of detailed records.
- With a sketch you can show a concept to a team or client, come up with potential solutions quickly, and help better understand a project yourself.

Sketch (example)



The Art of Sketches and Snapshot

- The art of sketching is something that all designers should practice. Ideas and thoughts can be quickly visualized – either to yourself, or to others – and explored.
- … to display large amounts of data without the need for scrolling (in order to move away from the personal computer paradigm).
- The design principle underlying the designs is often referred to as "focus and context".
- Ben Shneiderman encourages designers to use: "overview first, zoom and filter, then details on demand".
- Individual snapshots of a design can be provided to show key moments in an interaction and are particularly useful for exploring the impact of a certain style or design.
- Snapshots can be single sketches, or frames, from a storyboard (see below) or they can be produced using software.

Wireframe

- It primarily allows you to define the information hierarchy of your design, making it easier for you to plan the layout according to how you want your user to process the information.
- Communication tools to share with client or customer, in which take consideration all page element. It also allow for easy changes, updates and revision.
- It can be defined as a page schematic or screen blueprint, is a visual guide that represents the skeletal framework of a website (Brown, 2011).
- The skeleton plan of a website can be broken down into three components: information design, navigation design, and interface design. Page layout is where these components come together, while wireframing is what depicts the relationship between these components (Garret, 2010).

Uses of Wireframe

- Brown (2011) explained that developers use wireframes to get a more tangible grasp of the site's functionality, while designers use them to push the user interface (UI) process. User experience designers and information architects use wireframes to show navigation paths between pages. Business Analysts use wireframes to visually support the business rules and interaction requirements for a screen. Business stakeholders review wireframes to ensure that requirements and objectives are met through the design.
- Resembling a rough sketch or a quick mock-up, low-fidelity wireframes have less detail and are quick to produce. These wireframes help a project team collaborate more effectively since they are more abstract, using rectangles and labeling to represent content (Wodtke & Govella, 2009).
- High-fidelity wireframes are often used for documenting because they incorporate a level of detail that more closely matches the design of the actual webpage, thus taking longer to create (Wodtke & Govella, 2009).

Brown (2011) Explained that:

Wireframes focus on:

- The range of functions available.
- The relative priorities of the information and functions.
- The rules for displaying certain kinds of information.
- The effect of different scenarios on the display.

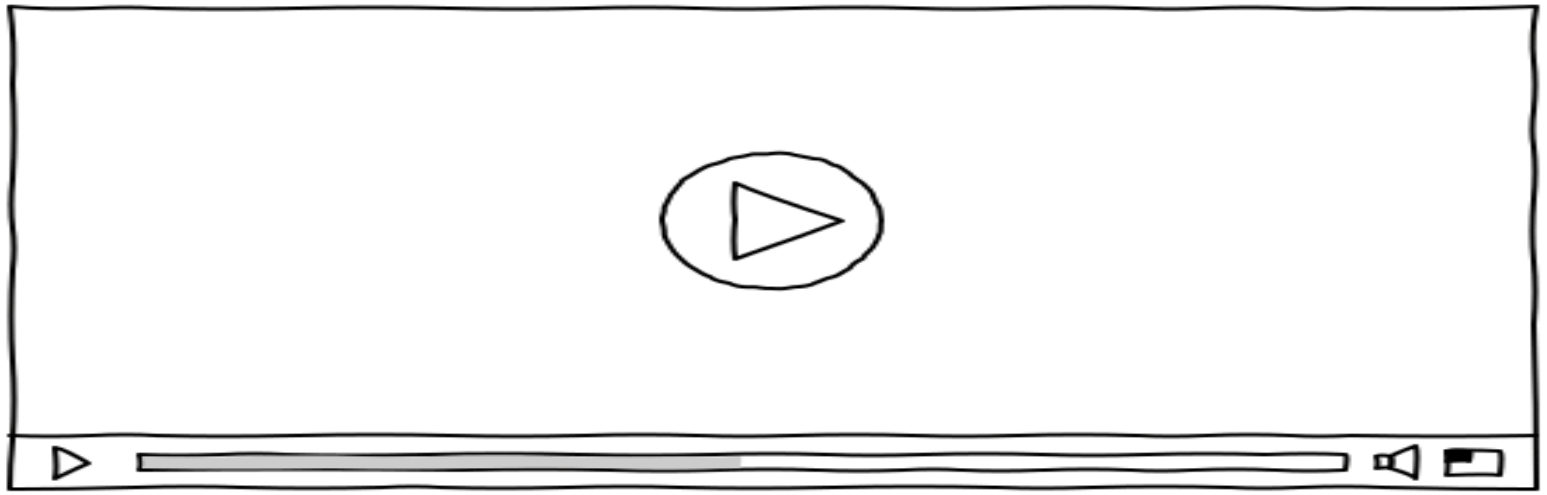
Wireframing typically begins between “high-level structural work—like flowcharts or site maps—and screen designs”.

YouTube

YouTube Search Search Browse | Upload Create Account | Sign In

Matrix - The pill

johndoe 9 videos Subscribe



Like | Add to | Share Embed

534,151

1,241 likes, 24 dislikes

Uploaded by johndoe on Sep 25, 2007

Morpheus shows Neo two pills: a blue and a red one. If Neo chooses the

Show more

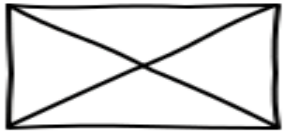
All Comments (1,838)

Respond to this video...

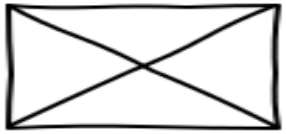
youtube is the matrix!!!! I'm trapped!!!! AHHHH!!!

matrixuser 7 hours ago

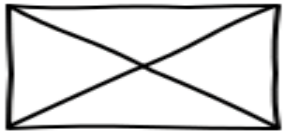
Suggestions



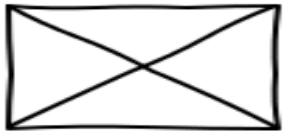
How The Matrix Should Have Ended
by otheruser
4,860,784 views



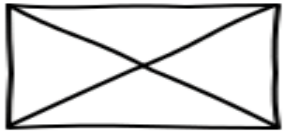
Neo vs Morpheus
by otheruser
163,624 views



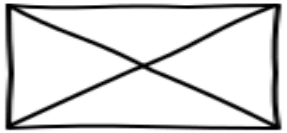
Matrix is real
by otheruser
2 views



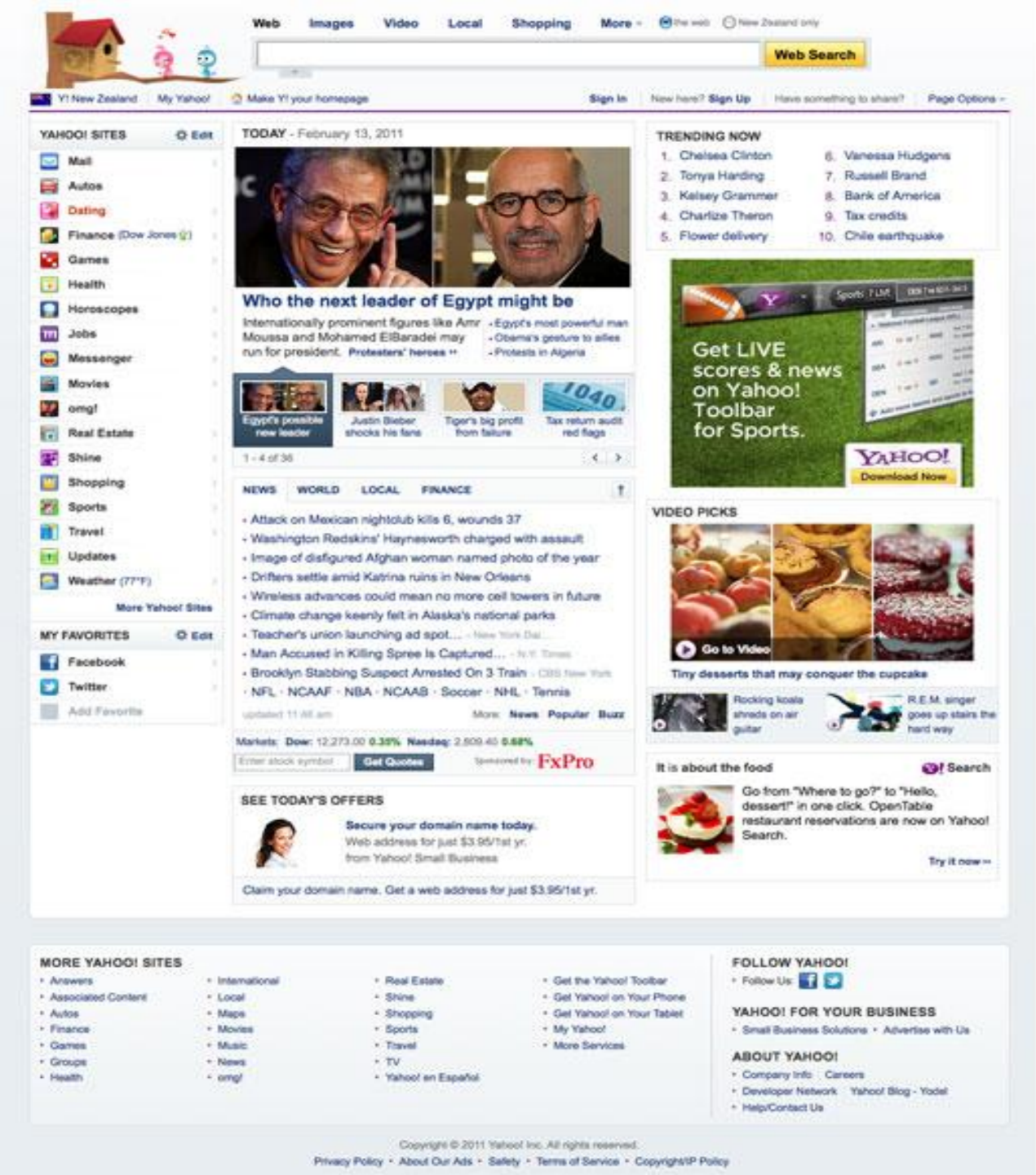
No it's not
by otheruser
2 views

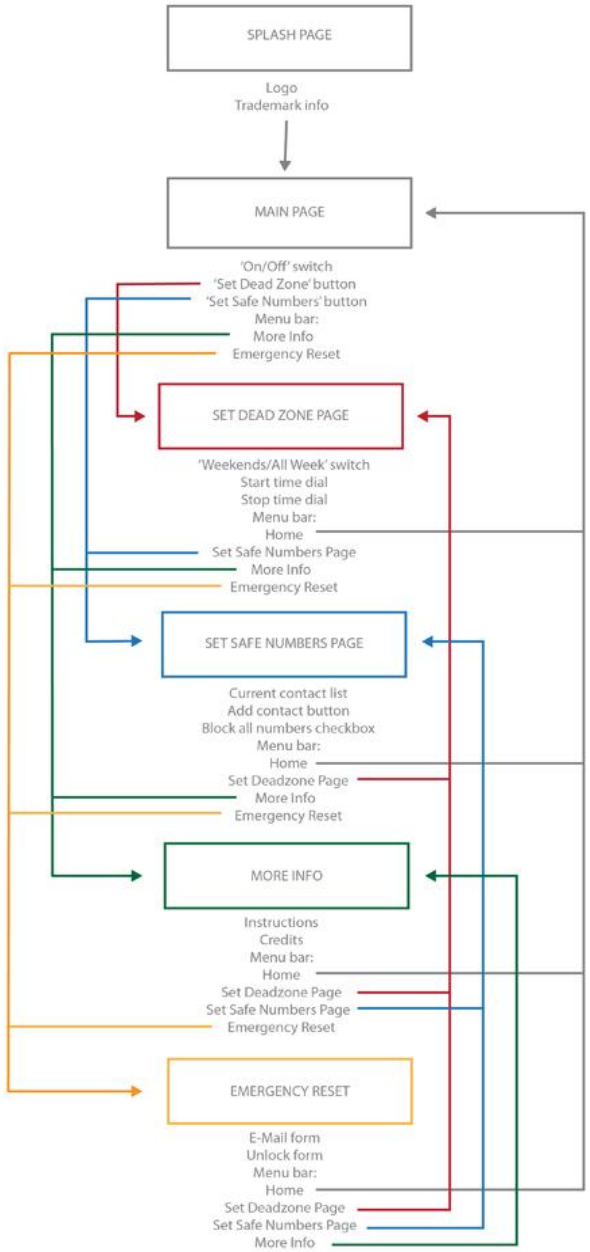


Yeah, it is!
by otheruser
2 views



Where am I??!
by otheruser
0 views





Mockups

- It can defined as a scale or full-size model of a design or device, used for teaching, demonstration, design evaluation, promotion, and other purposes. A mockup is a prototype if it provides at least part of the functionality of a system and enables testing of a design (Tudor, 2009).
- Mockups are used by designers mainly to acquire feedback from users. Mock-ups address the idea captured in a popular engineering one-liner: You can fix it now on the drafting board with an eraser or you can fix it later on the construction site with a sledge hammer.
- Mockups is a realistic representation of what the product will look like, in this case a website. The final result can look exactly like the mockup, or be a variation of it after version revisions. While some people prefer to draw the mockups using graphic software, others do it straight in HTML/CSS. Meanwhile, **wireframe** is about functionality. It can be a really simple sketch that demonstrates what sort of things you can do in your design.
- Mockups are built on top of wireframes and go further to show overall appearance aspects of a design including type choices, color choices, etc. The goal of a mock up is to show, as close as possible, how all final appearances will be rendered.

A Web Page

http://

Dashboard > Main > Home

Search

Wiki

Welcome Peldi | History | Preferences | Administration | Log Out

Page Operations

Browse Space

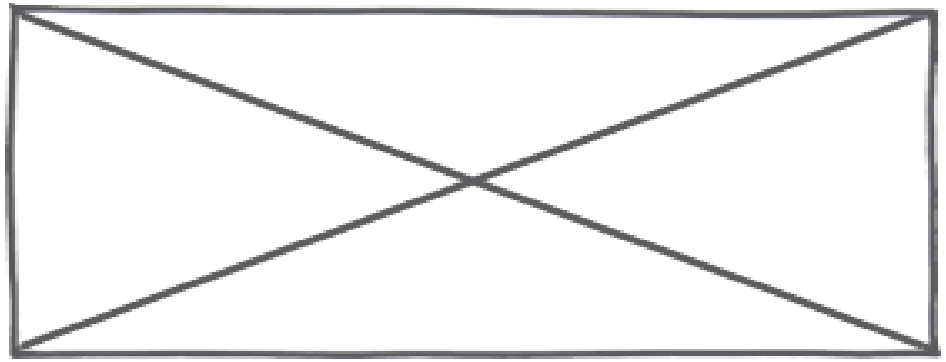
Add Content

Main

Welcome to Our Wiki!

Added by Peldi, last edited by Peldi on Mar 09, 2008 (view changes)

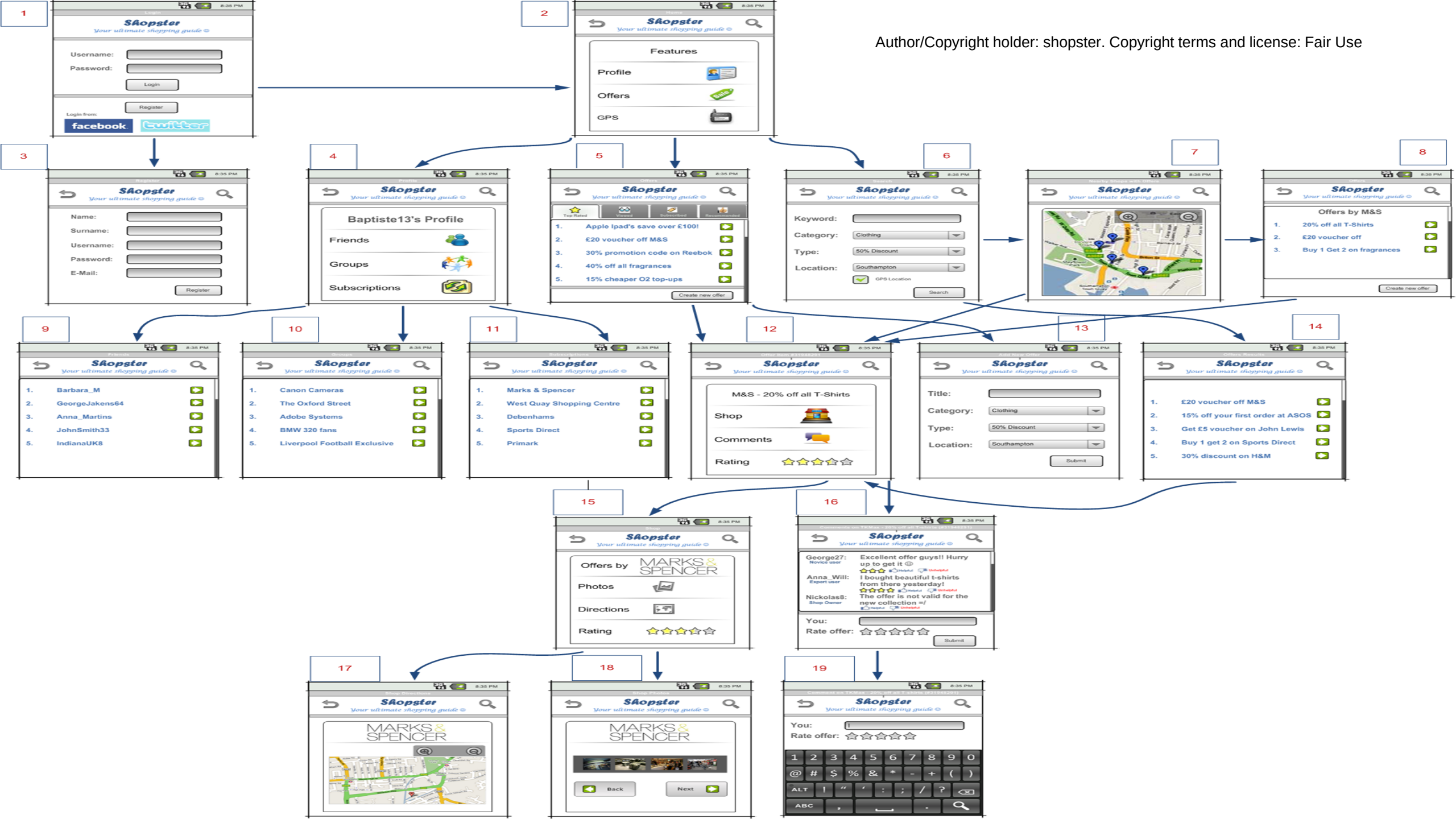
Welcome to our awesome wiki. A place for us to share knowledge yadda yadda yadda. You can add images like the one below:



And other text lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip...

10 Children Show Children | View in Hierarchy | Add Child Page

0 comments Add Comment



Artefacts

- An artefact means any product of human workmanship or any object modified by man. It is used to denote anything from a hammer to a computer system or to denote activities in design process, or outcome of a process activity (Larman, 1998).
- The specification of a system is the combination of all the different products produced during the development process. During design process, there are several artefacts produced:
  - Requirement/Problems
  - Scenario Corpus
  - Object Model
  - Design Language

Requirement and Problems

- In the gathering of people's stories and during the analysis and abstraction process various issues and difficulties will come to light.
- These help the analyst/designer to establish a list of requirements – qualities or functions that any new product or system should have.
- For example, in the HIC example, the device has to be used by the elderly and short-sighted.
- Another requirement was that it should look good in a living room and should not look or behave like a personal computer running Microsoft Windows.
- The format of the requirements and problems is a prioritized list of issues, or a more formalized format.

Scenario Corpus

- In our approach we seek to develop a representative and carefully thought-through set, or corpus, of scenarios.
- Having undertaken some analysis activities designers will have gathered a wide range of user stories.
- Some of these will be very general and some will be quite specific.
- Some will be fairly simple, straightforward tasks; others will be more vague.
- It is important at some point for the designer to pull these disparate experiences together in order to get a high-level, abstract view of the main activities that the product is to support.
- These conceptual scenarios will often still be grounded in a real example; the trick is to find an example that shares characteristics with a number of other activities.

Domain Characteristics

- The rationale for the development of a corpus of scenarios is to uncover the "dimensions" of the design situation and to demonstrate different aspects of those dimensions.
- Dimensions include characteristics of the various domains within which the product will operate (e.g. large and small domains, volatile or static domains, etc.), the various media and data types that need to be accommodated and the characteristics of the people who will be using the system.
- The corpus of scenarios needs to cover all the main functions of the system and the events that trigger the functions.
- Different types of interaction need to be present along with any key usability issues.
- The dimensions include different types of content and how that can be structured, issues of style and aesthetics.
- The aim is to specify the scenarios at a level of abstraction that captures an appropriate level of generality that will be useful across the range of characteristics that is demonstrated within a domain.

Object Model

- An object or data model results from the process of conceptual modeling, including developing the scenarios and undertaking an object/action analysis of the scenario corpus.
- The conceptual model shows the main objects in the system, their attributes and the relationships that exist between them.
- Conceptual modeling is a very important part of interactive systems design that is often overlooked.
- Having a clear, well-designed conceptual model will make it easier to design so that people can develop a good, accurate mental model of the system.
- The conceptual model will also form the basis of the information architecture of a system and for any metaphor that is used in the design.
- Two famous conceptual models are the concept of the spreadsheet and the various objects such as printers, folders, documents, etc. that make up the "desktop" metaphor of the Windows and Mac OS operating systems.

Design Language

- The design language produced consists of a set of standard patterns of interaction and all the physical attributes of a design – the colors, shapes, icons and so on.
- These are brought together with the conceptual actions and objects and the "look and feel" of the design is completed.
- A "design language" defines the key elements of the design (such as the use of color, style and types of buttons, sliders and other widgets, etc.) and some principles and rules for putting them together.
- •A consistent design language means that users need learn only a limited number of design elements and then they can cope with a large variety of different situations.

Trade-offs

- Choosing which goals or constraints can be relaxed so that others can be met. For example, we might find that an eye-mounted video display, a bit like those used in virtual reality, would give the most stable image whilst walking along. However, this would not allow you to show friends, and might be dangerous if you were watching a gripping part of the movie as you crossed the road.
- Design is characterized by trade-offs and there is rarely a simple solution to a problem that solves all the issues.
- Usually the adoption of one design will mean that something else cannot be achieved.
- Designers need to document their design decisions so that the trade-offs can be evaluated and scenarios help by making the rationale for the design explicit.
- Designers can record the claims that they make about their designs.

Navigation Design

- The object of design is not just a computer system or device, but the socio-technical intervention as a whole. However, as design progresses we come to a point where we do need to consider these most tangible outputs of design.
- The appropriate choice of widgets and wording in menus and buttons will help you know how to use them for a particular selection or action.
- You need to find things on the screen, understand the logical grouping of buttons.
- You need to be able to understand what will happen when a button is pressed, to understand where you are in the interaction.
- The program may read documents from disk, perhaps some are on remote networks. You swap between applications, perhaps cut and paste.
- We may need to support linkages between applications, for example allowing the embedding of data from one application in another, or, in a mail system, being able to double click an attachment icon and have the right application launched for the attachment.

Layout Design

- If things logically belong together, then we should normally physically group them together involve multilevel structure.
- In general we need to think: what is the natural order for the user? This should normally match the order on screen. For data entry forms or dialog boxes we should also set up the order in which the tab key moves between fields.
- Other decorative features like font style, and text or background colors can be used to emphasize groupings.
- For users who read text from left to right, lists of text items should normally be aligned to the left. Numbers, however, should normally be aligned to the right (for integers) or at the decimal point.
- Screwing up your eyes so that the screen becomes slightly blurred is another good technique for taking your attention away from the content and looking instead at the broad structure.

Affordances

- The psychological idea of *affordance* says that things may suggest by their shape and other attributes what you can do to them: a handle affords pulling or lifting; a button affords pushing.
- One can either mimic real-world objects directly, or try to emulate the critical aspects of those objects.
- Note also that affordances are not intrinsic, but depend on the background and culture of users. Most computer-literate users will click on an icon. This is not because they go around pushing pictures in art galleries, but because they have learned that this is an affordance of such objects in a computer domain.

Signifiers

- The study of signs and symbols and their use or interpretation (Oxford Dictionary) in which, a sign was not only a sound-image but also a concept.
- So the question is, how do signifiers create meaning and how do we know what that meaning is? In order to understand how the signifier and signified relate to each other, one must be able to interpret signs.
- A sign can only be understood when the relationship between the two components that make up the sign are agreed upon or words within the system (Saussure, 1911).
- However we need to remember that signifiers and their significance change all the time, becoming “dated”.

Aesthetics and Utility

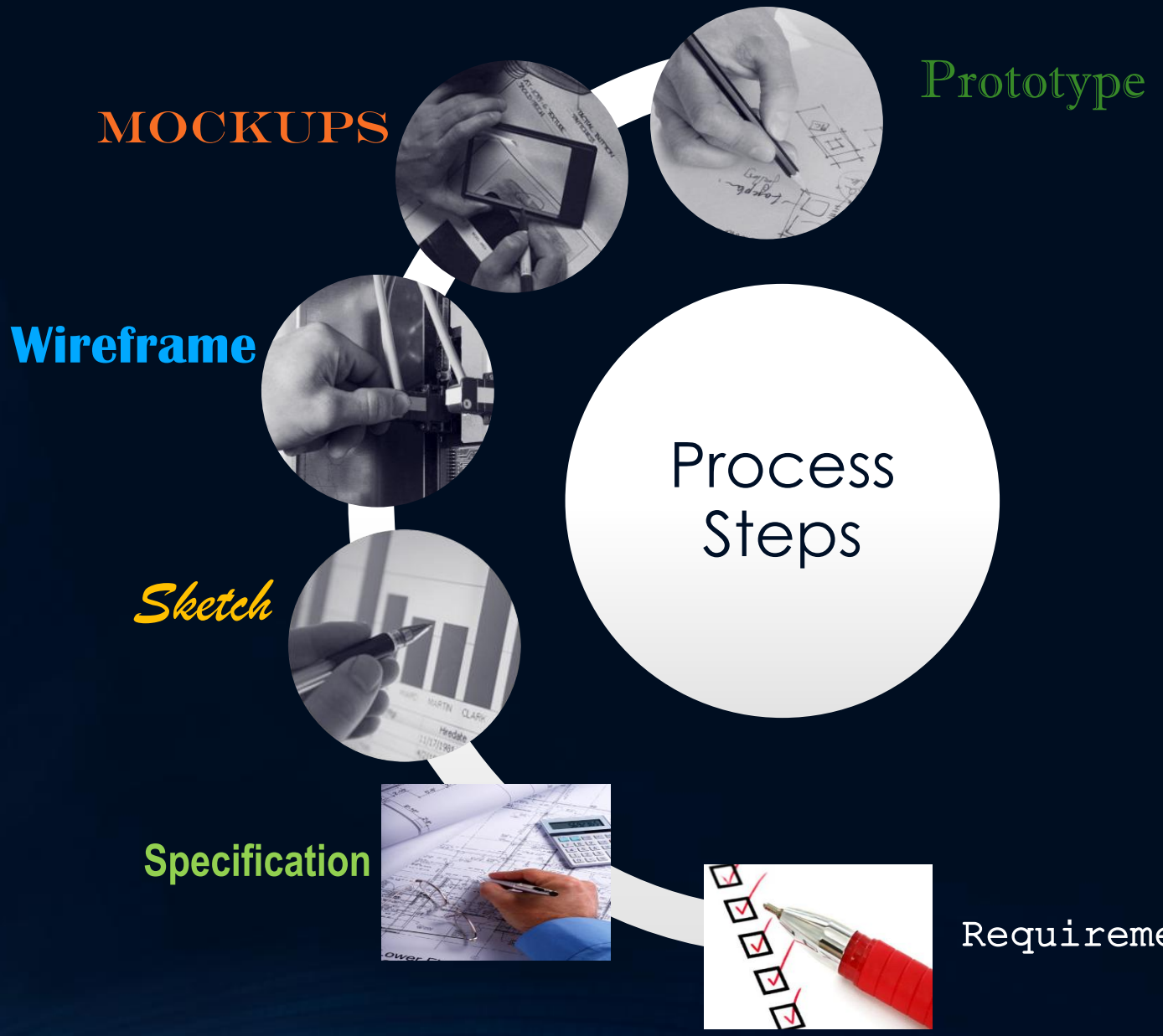
- Ideally, as with any well-designed item, an interface should be aesthetically pleasing. Indeed, good graphic design and attractive displays can increase users' satisfaction and thus improve productivity.
- However, beauty and utility may sometimes be at odds. For example, an industrial control panel will often be built up of the individual controls of several subsystems, some designed by different teams, some bought in. The resulting inconsistency in appearance may look a mess and suggest tidying up.
- In particular, the backdrop behind text must have low contrast in order to leave the text readable; this is often not the case and graphic designers may include excessively complex and strong backgrounds because they look good. The results are impressive, perhaps even award winning, but completely unusable!
- On a more positive note, careful application of aesthetic concepts can also aid comprehensibility.

Consistency

- This refers to designing interfaces to have similar operations and use similar elements for achieving similar tasks. In particular, a consistent interface is one that follows rules, such as using the same operation to select all objects.
- Users have to learn only a single mode of operation that is applicable to all objects. This principle works well for simple interfaces with limited operations.
- A much more effective design solution is to create categories of commands that can be mapped into subsets of operations for complex operation.
- Problem with consistency is determining what aspect of an interface to make consistent with what else. There are often many choices, some of which can be inconsistent with other aspects of the interface or ways of carrying out actions.

Prototype

- Most user interface design therefore involves some form of prototyping, producing early versions of systems to try out with real users.
- It's one thing to discuss requirements documentation, but it's a whole other level of imaginative collaboration when both parties can play with a prototype and explore limitations and possibilities.
- Wireframes, mockups, and requirement documents live in paper, not reality. Prototyping allows teams to experiment, giving them the freedom to fail cheaply while learning more.
- Prototypes can be great for pitching if you're working with skeptical clients. Experiencing the real-life website or app proves your vision more than a wordy description.
- By user-testing a prototype, you're able to find problems and fix them earlier in the process, saving yourself a huge hassle of dealing with them when they're cemented in code.
- By having prototype early and often because prioritizing interaction design will keep you grounded in reality when you make static design decisions.


MOCKUPS
Prototype
Wireframe
Process Steps
Sketch
Specification
Requirement

Iteration

- Humans are complex and we cannot expect to get designs right first time. We therefore need to evaluate a design to see how well it is working and where there can be improvements.
- Some forms of evaluation can be done using the design on paper, but it is hard to get real feedback without trying it out.
- Iterative design is a process of designing a product in which the product is tested and evaluated repeatedly at different stages of design to eliminate usability flaws before the product is designed and launched.
- It allows teams to reduce usability issues and thus ensure a good user experience of the product they are developing.

User Experience Goals

- The realization that new technologies are offering increasing opportunities for supporting people in their everyday lives has led researchers and practitioners to consider further goals.
- The emergence of technologies (e.g., virtual reality, the web, mobile computing) in a diversity of application areas (e.g., entertainment, education, home, public areas) has brought about a much wider set of concerns besides focusing primarily on improving efficiency and productivity at work.
  - Satisfying and Enjoyable
  - Fun and Entertaining
  - Helpful and Motivating
  - Aesthetically pleasing
  - Supportive of creativity
  - Rewarding
  - Emotionally fulfilling
- This involves explicating the nature of the user experience in subjective terms.
- Aspects that have been described as contributing to pleasure include: attention, pace, play, interactivity, conscious and unconscious control, engagement, and style of narrative.

Constraint

- The design concept of constraining refers to determining ways of restricting the kind of user interaction that can take place at a given moment.
- Norman (1999) classifies constraints into three categories: physical, logical, and cultural. Physical constraints refer to the way physical objects restrict the movement of things. Logical constraints rely on people's understanding of the way the world works on people's common-sense reasoning about actions and their consequences. Cultural constraints rely on learned conventions, like the use of red for warning, the use of certain kinds of audio signals for danger, and the use of the smiley face to represent happy emotions.
- Making actions and their effects obvious enables people to logically deduce what further actions are required.
- Most cultural constraints are arbitrary in the sense that their relationship with what is being represented is abstract, and could have equally evolved to be represented in another form (e.g., the use of yellow instead of red for warning).

Feedback

- Feedback is about sending back information about what action has been done and what has been accomplished, allowing the person to continue with the activity.
- Various kinds of feedback are available for interaction design- audio, tactile, verbal, visual, and combinations of these. Deciding which combinations are appropriate for different kinds of activities and interactivities is central.
- Using feedback in the right way can also provide the necessary visibility for user interaction.

Exploring Design Concepts

- A design concept is the idea behind a design. It's how you plan on solving the design problem in front of you. It's the underlying logic, thinking, and reasoning for how you'll design a website, application, product, etc.
- Bill Verplank (2003) is an interaction designer who has been sketching and designing for many years. He argues that interaction design is 'design for human use' and focuses on three main things, which he characterizes as:

- How do you do?
- How do you feel?
- How do you know?

How do you do?

- ‘How do you do?’ is concerned with the ways in which we affect the world.
- Do you poke it, manipulate it, sit on it?
- For example, one distinction he highlights is between handles and buttons.
- Handles are better for continuous control (e.g. trombone), but buttons are better for discrete control (e.g. a piano keyboard).
- Handles leave you in control (e.g. opening a car door) whereas buttons are more likely to trigger something automatic (e.g. opening an elevator door).

How do you fell?

- ‘How do you feel?’ concerns how we make sense of the world and the sensory qualities that shape media.
- One distinction is Marshall McLuhan’s ‘hot’ versus ‘cool’. Marshall McLuhan wrote *Understanding Media in 1964 and is famed for coining the phrases ‘Global Village’, ‘age of information’ and ‘the medium is the message’.*
- ‘hot media’ are more authoritative and exact.
- ‘cool media’ which are fuzzy and incomplete.
- Cool media invite more participation; they require the audience to fill in the gaps, to interpret. Cartoon is a cool medium but low definition.
- Hot media extend a single sense in high definition; they are filled with data. Photography is a hot medium because it is high fidelity.

How do you know?

- ‘How do you know?’ concerns the ways that people learn and plan; how designers want people to think about their system.
- For example, Verplank suggests that one choice is between maps and paths.
- Paths are good for beginners as they provide step-by-step instructions on what to do.
- Maps are good for understanding alternatives.
- They take longer to learn but are more robust and are good for expert skill.
- Maps offer the chance to take short cuts. Very often, of course a given system or product will have to accommodate both.

Summary

- The design of interactive systems is concerned with people, the activities they are undertaking, the contexts of those activities and the technologies that are used.
- This presentation has introduced the main elements of design – understanding, envisionment, design and evaluation and how scenario-based design and the development of personas can be used to guide the designer.
- Scenarios and their different uses in this process have been explored.
- Scenarios are stories about the interactions between people, activities, contexts and technologies.
- Scenarios offer an effective way of exploring and representing activities, enabling the designer to generate ideas, consider solutions and communicate with others.
- Scenarios are used throughout the design process and, as use cases, form part of the formal specification of the system.

Reference Books

- David Benyon. Designing interactive systems: A comprehensive guide to HCI, UX and interaction design. Addison-Wesley, Pearson 2013.
- Rex Hartson and Pardha S. Pyla. The UX book: process and guidelines for ensuring a quality user experience. Elsevier 2012.
- Alan Dix, Janet Finlay, Gregory D. Abowd and Russell Beale. Human-computer interaction. Prentice hall, Pearson 2003.
- Helen Sharp, Jenny Preece and Yvonne rogers. Interaction Design: Beyond Human-Computer Interaction. John-Wiley & Sons. 2007.