1.What is HCI and write about its each components of HCI

Human-computer interaction (HCI) is a multidisciplinary field of study focusing on the design of computer technology and, in particular, the interaction between humans (the users) and computers. While initially concerned with computers, **HCI** has since expanded to cover almost all forms of information technology design.

- **Human**: By "user", we may mean an individual user or a group of users working together. An appreciation of the way people's sensory systems (sight, hearing, touch) relay information is vital. Also, different users form different conceptions or mental models about their interactions and have different ways of learning and keeping knowledge. In addition, cultural and national differences play an important part.
- **Computer**: When we talk about the computer, we're referring to any technology ranging from desktop computers, to large scale computer systems. For example, if we were discussing the design of a Website, then the Website itself would be referred to as "the computer". Devices such as mobile phones or VCRs can also be considered to be ―computers‖.
- **Interaction**: There are obvious differences between humans and machines. In spite of these, HCI attempts to ensure that they both get on with each other and interact successfully. In order to achieve a usable system, you need to apply what you know about humans and computers, and consult with likely users throughout the design process. In real systems, the schedule and the budget are important, and it is vital to find a balance between what would be ideal for the users and what is feasible in reality. Human-Computer Interaction studies how people design, implement and use computer interfaces. HCI has become an umbrella for term for a number of disciplines including theories of education, psychology, collaboration as well as efficiency and ergonomics

2.What are the stages of Norman's model of an Interaction and discuss the Norman's Model

Seven stages

1. user establishes the goal
2. formulates intention
3. specifies actions at interface
4. executes action
5. perceives system state
6. interprets system state
7. evaluates system state with respect to goal

User chooses a goal, formulate a plan of action, which is then executed at the computer interface. When the plan, or part of the plan has been executed, the user observes the computer interface to evaluate the result of the execution plan, and to determine further actions

The two major parts, execution and evaluation, of interactive cycle are further subdivided into seven stages, where each stage is an activity of the user. Seven stages of action are shown in figure. To understand these we see an example, which was also used by Norman

Imagine you are sitting reading as evening falls. You decide you need more light; that is you establish the goal to get lighter. Form there you form an intention to switch on the desk lamp, and you specify the actions required to reach over and press the lamp switch. If some one else is closer, the intention may be different- you may ask them to switch on the light for you. Your goal is the same but the intention and actions are different. When you have executed the action you perceive the result, either the light is on or it isn't and you interpret this, based on your knowledge of the world. For example, if the light does not come on you may interpret this as indicating he bulb has blown or the lamp is not plugged into the mains, you will formulate the new state according to the original goals is there is now enough light? It so, the cycle is completed. It not, you may formulate a new intention to switch on the main ceiling light as well

Gulf of execution and evaluation
Norman also describes the two gulfs, which represent the problems that are caused by some interfaces to their users

Gulf of execution
Gulf of execution is the difference between the user's formulation of the actions to reach the goal and the actions allowed by the system. If the action allowed by the system correspond to those intended by the user, the interaction will effective. The interface should therefore aim to reduce this gulf of execution

Gulf of evaluation
The gulf of evaluation is the distance between the physical presentation of the system state and the expectation of the user. If the user can readily evaluate the presentation in terms of his goal, the gulf of evaluation is small. The more effort that is required on the part of the user to interpret the presentation, the less effective the interaction.

3.Differentiate between Direct and Indirect manipulation.

Both are needed. However, direct manipulation should be used way more than it currently is. Whenever I look at UI Designs from other designers, I am struck by the plethora of indirect manipulation cases. Here is an example to clarify.

Case: You have a table of data (columns and rows) and you want to reorder the columns. The designer makes a button called “Edit Columns” where you can add/remove/order the visible columns. You are editing a UI that doesn’t look like a table to manipulate the table. This is indirect manipulation.

A better choice is to allow drag-drop of the columns and add a “last column” where you can type in the field you want. Also, you could click on the header to change the field to something else. This is direct manipulation. It requires more sophisticated programming, but that is the price for a better interface.

The reason it’s better is because it maintains the mental model of the UI. This is a simplistic example, but every product has areas you can change things (Admin area!) and it makes a change on some other distant area of the application.

It makes it harder to understand the consequences of your action. It also makes it harder to understand where to change the behavior of the screen you are looking at.

In the physical world the same thing happens. In Flint Michigan, the city supervisors changed their water policy to save money. It also had the effect of poisoning it’s citizens with lead and other toxins. They didn’t mean to poison people. Imagine if someone said to them, “If you ADD LEAD to the water and poison people, we will pay you a few million bucks.” I doubt if the powers that be would agree to that deal. However, they did exactly that. They just didn’t realize it.

Direct manipulation forces you to see the results of your actions. Indirect manipulation hides that connection. I’m not saying you should use direct manipulation everywhere. I am just saying it should be your first choice, not your last.

Direct Manipulation

The term direct manipulation is given by Shneiderman (1982) as they possess the following characteristics:

i. The system is portrayed as an extension of the real world.

ii. Continuous visibility of objects and actions.

iii. Actions are rapid and incremental with visible display of results.

iv. Incremental actions are easily reversible.

Example for direct manipulation:

Driving an automobile

The scene is directly visible through the front window and performance of actions such as braking or steering has become common knowledge in our culture. To turn left, the driver simply rotates the steering wheel to the left. The response is immediate and the scene changes, providing feedback to refine the turn.

In the above example, a driver looking at an object directly manipulates the scenario by his sudden action, which can be referred as direct manipulation.

Indirect Manipulation

In practice, direct manipulation of all screen objects and actions may not be feasible because of the following:

i. The operation may be difficult to conceptualize in graphical system.

ii. The graphics capability of the system may be limited.

iii. The amount of space available for placing manipulation controls in the window border may be limited.

iv. It may be difficult for people to learn and remember all the necessary operations and actions.

When this occurs, indirect manipulation is provided. Indirect manipulation substitutes words and text, such as pull-down or pop-up menus, for symbols and substitutes typing for pointing.

Most window systems are a combination of both direct manipulation and indirect manipulation. A menu may be accessed by pointing at a menu icon and then selecting it (direct manipulation) or the menu itself is a textual list of operations (indirect manipulation).

4.Analyse briefly 4 different interaction styles used to accomadate the dialog between the user and interface.

Interaction Styles

BY MADS SOEGAARD

The concept of Interaction Styles refers to all the ways the user can communicate or otherwise interact with the computer system. The concept belongs in the realm of HCI or at least have its roots in the computer medium, usually in the form of a workstation or a desktop computer. These concepts do however retain some of their descriptive powers outside the computer medium. For example, you can talk about menu selection (defined below) in mobile phones.

In HCI textbooks, such as Shneiderman (1997) and Preece et al. (1994), the types of interaction styles mentioned are usually command language, form fillin, menu selection, and direct manipulation.

1. Command language (or command entry)

Command language is the earliest form of interaction style and is still being used, though mainly on Linux/Unix operating systems. These &quot;Command prompts&quot; are used by (usually) expert users who type in commands and possibly some parameters that will affect the way the command is executed. The following screen dump shows a command prompt - in this case, the user has logged on to a (mail) server and can use the server&#39;s functions by typing in commands.

Command language places a considerable cognitive burden on the user in that the interaction style relies on recall as opposed to recognition memory. Commands as well as their many parameterised options have to be learned by heart and the user is given no help in this task of retrieving command names from memory. This task is not made easier by the fact that many commands (like the &#39;ls&#39; command in the above example) are abbreviated in order to minimize the number of necessary keystrokes when typing commands. The learnability of command languages is generally very poor.

Advantages and disadvantages of Command Language

Advantages

Flexible.

Appeals to expert users.

Supports creation of user-defined &quot;scripts&quot; or macros.

Is suitable for interacting with networked computers even with low bandwidth.

Disadvantages

Retention of commands is generally very poor.

Learnability of commands is very poor.

Error rates are high.

Error messages and assistance are hard to provide because of the diversity of possibilities plus the complexity of mapping from tasks to interface concepts and syntax.

Not suitable for non-expert users.

2. Form fillin

The form fillin interaction style (also called &quot;fill in the blanks&quot;) was aimed at a different set of

users than command language, namely non-experts users. When form fillin interfaces first appeared, the whole interface was form-based, unlike much of today&#39;s software that mix forms with other interaction styles. Back then, the screen was designed as a form in which data could be entered in the pre-defined form fields. The TAB-key was (and still is) used to switch between the fields and ENTER to submit the form. Thus, there was originally no need for a pointing device such as a mouse and the separation of data in fields allowed for validation of the input. Form fillin interfaces were (and still is) especially useful for routine, clerical work or for tasks that require a great deal of data entry. Some examples of form filling are shown below.

Classic Form fillin via a terminal

More modern-day form fillin,

could be from a web page.

Even today, a lot of computer programs like video rental software, financial systems, pay roll systems etc. are still purely forms-based.

Advantages and disadvantages of Form Fillin

Advantages

Simplifies data entry.

Shortens learning in that the fields are predefined and need only be &#39;recognised&#39;.

Guides the user via the predefined rules.

Disadvantages

Consumes screen space.

Usually sets the scene for rigid formalisation of the business processes.

Please note that &quot;form fillin&quot; is not an abbreviation of &quot;form filling&quot;. Instead, it should be read

&quot;form fill-in&quot;.

3. Menu selection

A menu is a set of options displayed on the screen where the selection and execution of one (or more) of the options results in a state change of the interface. Using a system based on menu-selection, the user selects a command from a predefined selection of commands arranged in menus and observes the effect. If the labels on the menus/commands are understandable (and grouped well) users can accomplish their tasks with negligible learning or memorisation as finding a command/menu item is a recognition as opposed to recall memory task (see recall versus recognition). To save screen space menu items are often clustered in pull-down or pop-up menus. Some examples of menu selection is shown below.

Contemporary menu selection

(Notepad by Microsoft Cooperation)

Menu selection in the form of a webpage (microsoft.com).

Webpage in general can be said to be based on menu selection.

Advantages and disadvantages of Menu Selection

Advantages

Ideal for novice or intermittent users.

Can appeal to expert users if display and selection mechanisms are rapid and if appropriate &quot;shortcuts&quot; are implemented.

Affords exploration (users can &quot;look around&quot; in the menus for the appropriate command, unlike

having to remember the name of a command and its spelling when using command language.)

Structures decision making.

Allows easy support of error handling as the user&#39;s input does not have to be parsed (as with command language).

Disadvantages

Too many menus may lead to information overload or complexity of discouraging proportions.

May be slow for frequent users.

May not be suited for small graphic displays.

4. Direct manipulation

Direct manipulation is a central theme in interface design and is treated in a separate encyclopedia entry (see this). Below, Direct manipulation is only briefly described.

Direct manipulation systems have the following characteristics:

Visibility of the object of interest.

Rapid, reversible, incremental actions.

Replacement of complex command language syntax by direct manipulation of the object of interest.

The text-book example of Direct Manipulation, the Windows File Explorer,
where files are dragged and dropped.

One of the earliest commercially available
direct manipulation interfaces was MacPaint.

Advantages and disadvantages of Direct Manipulation

Advantages

Visually presents task concepts.

Easy to learn.

Errors can be avoided more easily.

Encourages exploration.

High subjective satisfaction.

Recognition memory (as opposed to cued or free recall memory)

Disadvantages

May be more difficult to programme.

Not suitable for small graphic displays.

Spatial and visual representation is not always preferable.

Compact notations may better suit expert users.

5.Discuss the importance of grouping controls

Grouping Controls

Grouping related form controls makes forms more understandable for all users, as related controls are easier to identify. It also makes it easier for people to focus on smaller and more manageable groups rather than try to grasp the entire form at once.

Grouping needs to be carried out visually and in the code, for example, by using the `<fieldset>` and `<legend>` elements to associate related form controls. Also, related entries of a `<select>` element can be grouped using `<optgroup>`.

SHARE

Approach 1: Associating related controls with `fieldset`

The `<fieldset>` element provides a container for related form controls, and the `<legend>` element acts as a heading to identify the group.

The legend for a group of controls can also highlight common attributes of all controls, for example, to advise that all fields in the group are required.

SHARE

Example 1: Radio buttons

In the example below, there are three radio buttons that allow the user to choose an output format. Radio button groups should always be grouped using `<fieldset>`.

EXAMPLE:

Output format

○ Text file

○ CSV file

○ HTML file

CODE SNIPPET:

```
<fieldset>
<legend>Output format</legend>
  <div>
    <input type="radio" name="format" id="txt" value="txt" checked>
    <label for="txt">Text file</label>
  </div>
  <div>
    <input type="radio" name="format" id="csv" value="csv">
    <label for="csv">CSV file</label>
  </div>
  […]
</fieldset>
```

SHARE

Example 2: Checkboxes

In the example below, three checkboxes are part of an opt-in function for receiving different types of information.

EXAMPLE:

I want to receive

□ The weekly newsletter

□ Offers from the company

□ Offers from associated companies

CODE SNIPPET:

```
<fieldset>
<legend>I want to receive</legend>
       <div>
              <input type="checkbox" name="newsletter" id="check_1">
    <label for="check_1">The weekly newsletter</label>
       </div>
       […]
</fieldset>
```

SHARE

Example 3: Related fields

This example shows form fields to enter shipping and billing addresses. As the labels in both groups have the same text, the `fieldset` element also helps to distinguish the form fields by their groups. In case the `<legend>` is not read by screen readers (see note below), labels for the first form control in each group should include the group’s name. This name can be hidden visually.

EXAMPLE:

Shipping Address:

ShippingName:

Street:

Number:

City:

ZIP code:

Billing Address:

BillingName:

Street:

Number:

City:

ZIP code:

CODE SNIPPET:

```
<fieldset>
        <legend>Shipping Address:</legend>
        <div>
               <label for="shipping_name">
      <span class="visuallyhidden">Shipping </span>Name:
    </label><br>
               <input type="text" name="shipping_name" id="shipping_name">
        </div>
  <div>
    <label for="shipping_street">Street:</label><br>
    <input type="text" name="shipping_street" id="shipping_street">
  </div>
        […]
</fieldset>
<fieldset>
        <legend>Billing Address:</legend>
        <div>
               <label for="billing_name">
      <span class="visuallyhidden">Billing </span>Name:
    </label><br>
               <input type="text" name="billing_name" id="billing_name">
        </div>
  <div>
    <label for="billing_street">Street:</label><br>
    <input type="text" name="billing_street" id="billing_street">
  </div>
        […]
</fieldset>
```

Note: Depending on the configuration, some screen readers read out the legend either *with every form element*, *once*, or, rarely, *not at all*. To accommodate this consider the following:

- Make the legend as short as possible for situations in which it is read together with the label each time.
- Make the individual labels sufficiently self-explanatory for situations in which legends are not read aloud, without repeating the legend in every label.

SHARE

Approach 2: Associating related controls with WAI-ARIA

WAI-ARIA provides a grouping role that functions similarly to `fieldset` and `legend`. In this example, the `div` element has `role=group` to indicate that the contained elements are members of a group and the `aria-labelledby` attribute references the `id` for text that will serve as the label for the group.

This technique provides additional styling possibilities.

EXAMPLE:

Shipping Address:

ShippingName:

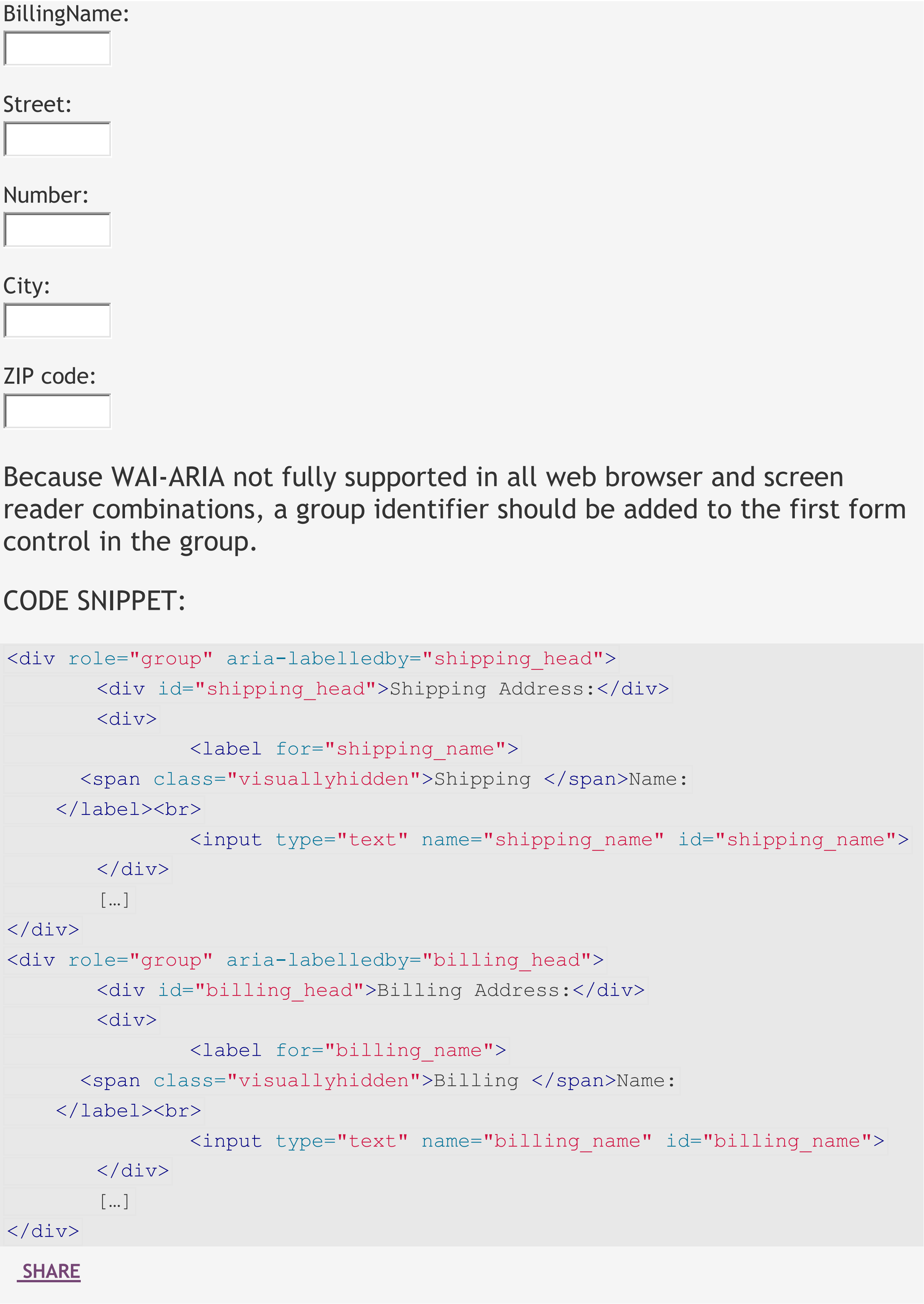
Street:

Number:

City:

ZIP code:

Billing Address:



BillingName:

Street:

Number:

City:

ZIP code:

Because WAI-ARIA not fully supported in all web browser and screen reader combinations, a group identifier should be added to the first form control in the group.

CODE SNIPPET:

```
<div role="group" aria-labelledby="shipping_head">
        <div id="shipping_head">Shipping Address:</div>
        <div>
                <label for="shipping_name">
      <span class="visuallyhidden">Shipping </span>Name:
    </label><br>
                <input type="text" name="shipping_name" id="shipping_name">
        </div>
        […]
</div>
<div role="group" aria-labelledby="billing_head">
        <div id="billing_head">Billing Address:</div>
        <div>
                <label for="billing_name">
      <span class="visuallyhidden">Billing </span>Name:
    </label><br>
                <input type="text" name="billing_name" id="billing_name">
        </div>
        […]
</div>
```

SHARE

Grouping items in `select` elements

For `select` elements with groups of options, the `optgroup` element can be used to indicate such groups. The `label` attribute of the `optgroup` element is used to provide a label for the group. This is especially useful for lists with many related options. In the example below, users can choose from lectures in one of three courses.

EXAMPLE:

Which course would you like to watch today?Course: ▾ ? Play

CODE SNIPPET:

```
<select>
       <optgroup label="8.01 Physics I: Classical Mechanics">
              <option value="8.01.1">Lecture 01: Powers of Ten</option>
              <option value="8.01.2">Lecture 02: 1D Kinematics</option>
              <option value="8.01.3">Lecture 03: Vectors</option>
       </optgroup>
       <optgroup label="8.02 Physics II: Electricity and Magnestism">
              <option value="8.02.1">Lecture 01: What holds our world
together?</option>
              […]
       </optgroup>
       […]
</select>
```

c

1. Discuss how GSM mobile network evolved?

The explosive growth of Global System for Mobile (GSM) Communication services over the last two decades has changed mobile communications from a niche market to a fundamental constituent of the global telecommunication markets. GSM is a digital wireless technology standard based on the notion that users want to communicate wirelessly without limitations created by network or national borders. In a short period of time, GSM has become a global phenomenon. The explanation for its success is the cooperation and coordination of technical and operational evolution that has created a virtuous circle of growth built on three principles: interoperability based on open platforms, roaming, and economies of scale (GSM Association, 2004a). GSM standards are now adopted by more than 200 countries and territories. It has become the main global standard for mobile communications; 80% of all new mobile customers are on GSM networks. GSM has motivated wireless adoption to the extent that mobile phones now globally outnumber fixed-line telephones. In February 2004, more than 1 billion people, almost one in six of the world's population, were using GSM mobile phones.

2. Tabulate some World's largest mobile operators.

3. **APPLICATION FRAMEWORKS**

Application frameworks often run on top of operating systems, sharing core services such as communications, messaging, graphics, location, security, authentication, and many others.

Java

Applications written in the Java ME framework can often be deployed across the majority of Java-based devices, but given the diversity of device screen size and processor power, cross-device deployment can be a challenge.

S60

The S60 platform, formerly known as Series 60, is the application platform for devices that run the Symbian OS. S60 is often associated with Nokia devices—Nokia owns the platform—but it also runs on several non-Nokia devices. S60 is an open source framework. S60 applications can be created in Java, the Symbian C++ framework, or even Flash Lite.

BREW

Applications written in the BREW application framework can be deployed across the majority of BREW-based devices, with slightly less cross-device adaption than other frameworks.

Flash Lite

Adobe Flash Lite is an application framework that uses the Flash Lite and ActionScript frameworks to create vector-based applications. Flash Lite applications can be run within the Flash Lite Player, which is available in a handful of devices around the world.
Flash Lite is a promising and powerful platform, but there has been some difficulty getting it on devices. A distribution service for applications written in Flash Lite is long overdue.

4. Explain the two distinct types of navigation layouts for mobile devices?

There are two distinct types of navigation layouts for mobile devices: touch and scroll. With touch, you literally point to where you want to go; therefore, navigation can be anywhere on the screen. But we tend to see most of the primary actions or navigation areas living at the bottom of the screen and secondary actions living at the top of the screen, with the area in between serving as the content area
This is the opposite of the scroll navigation type, where the device's D-pad is used to go left, right, up, or down. When designing for this type of device, the primary and often the secondary actions should live at the top of the screen. This is so the user

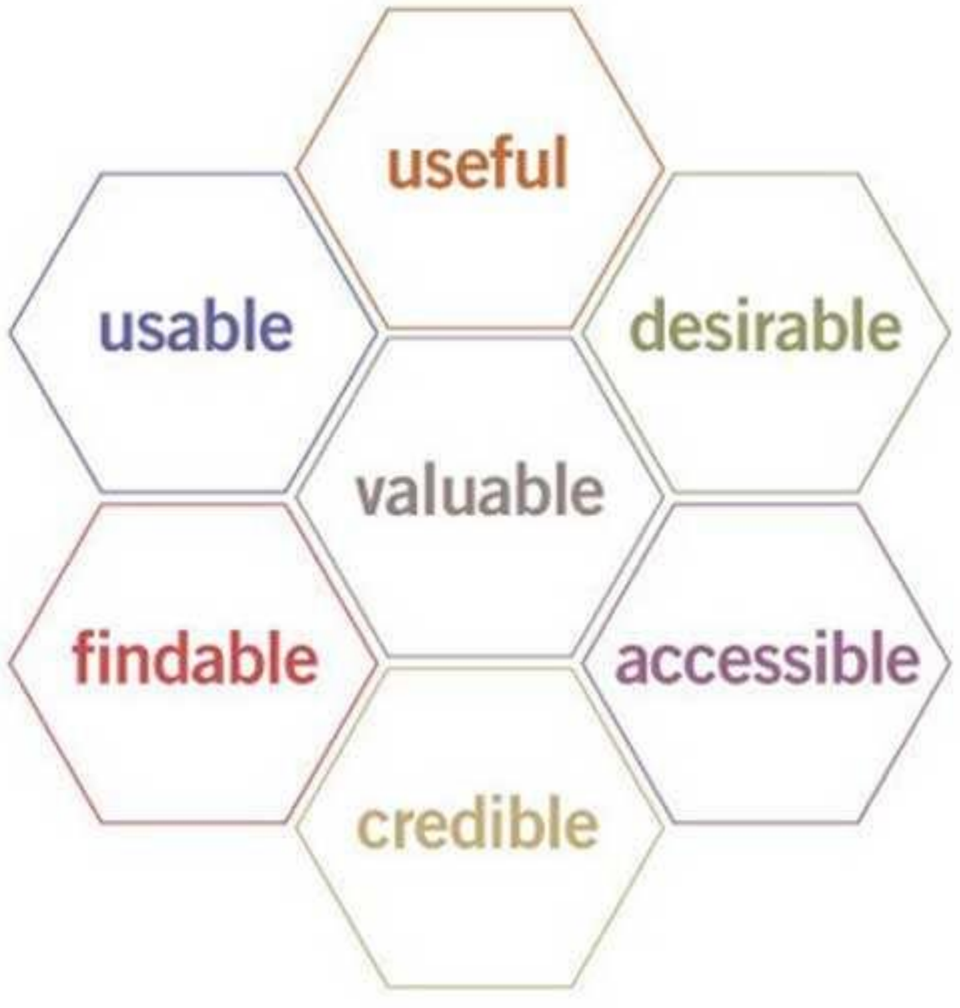
c

doesn't have to press down dozens of times to get to the important stuff. In Figure 8-9, you can actually see by the bold outline that the first item selected on the screen is the link around the logo.
When dealing with scroll navigation, you also have to make the choice of whether to display navigation horizontally or vertically. Visually, horizontally makes a bit more sense, but when you consider that it forces the user to awkwardly move left and right, it can quickly become a bit cumbersome for the user to deal with. There is no right or wrong way to do it, but my advice is just to try and keep it as simple as possible.

5. Define Jesse James Garrett's Elements of User Experience.

Factors that Influence UX

At the core of UX is ensuring that users find value in what you are providing to them. Peter Morville represents this through his User Experience Honeycomb.



He notes that in order for there to be a meaningful and valuable user experience, information must be:

- **Useful**: Your content should be original and fulfill a need

c

- **Usable**: Site must be easy to use
- **Desirable**: Image, identity, brand, and other design elements are used to evoke emotion and appreciation
- **Findable**: Content needs to be navigable and locatable onsite and offsite
- **Accessible**: Content needs to be accessible to people with disabilities
- **Credible**: Users must trust and believe what you tell them

6. Tabulate the pros and cons of mobile websites.

Pros

- **Compatibility.** A website enhances the user experience across different types of mobile devices. In contrast, a mobile application requires developing a separate version for each operating system and device type. Users who own devices of different types may especially appreciate the benefits of compatibility that responsive websites provide. Besides, they support easy integration with other mobile features like QR codes and text messaging.

- **Broader Reach.** Due to the multi-device support that responsive web design provides across various platforms, it's becoming easy to reach a broader audience than a mobile app can reach. In the "app vs website" debate, the website definitely wins in terms of potential audience.

- **Support and Maintenance.** Websites also cost less to upgrade. You need to maintain just a single version of a them. Compared to mobile apps, which require downloading of every single update, responsive/mobile websites let you change the content or design just by

c

editing them once, and allow you to do that efficiently and flexibly. After implementing, updates become active and visible immediately across all types of devices.

Cons

Mobile websites have a wider reach, better compatibility, and require less money than mobile applications. Still, they have their limitations.

- **Convenience.** Unlike an application, a responsive/mobile website can't leverage all smartphone features as efficiently. Cameras, GPS, phone dialing, and other features integrated into mobile devices aren't always well developed for responsive/mobile websites, even though APIs and libraries aimed to help solve these issues have been appearing lately.

  Another limitation to user benefits is a device's screen size. Of course, portability is the key reason why users enjoy having smaller devices on the go. However, mobile devices display a lot less content compared to a computer monitor or laptop screen. Both responsive design and mobile website don't fully access to all the content available on the desktop. Even though responsive web design adjusts to the screen size dynamically, still it doesn't make it completely convenient for users to surf the content on a mobile device. The same thing is with mobile website which actually reduces and rearranges the content available on the desktop.

c

- **User Experience.** Since mobile experiences significantly differ from desktop experiences, implementing the same interface for both platforms may harm your UX strategy. This mostly relates to the single-window restriction. The single-window restriction says that a user shouldn't have to leave their current page to access all the content they're looking for; however, with mobile sites it's often impossible to fit everything onto one page.

- **Offline Access.** Even if you design your mobile website in the lightest and most informative way possible, it still may work offline with only limited functionality using cached pages. Unlike a mobile application, which can run locally, a mobile website requires a good internet connection to operate fully.

7. Analyze the Elements of Mobile Design.

   (page number: 124)

8. Mobile 2.0

MOBILE 2.0

Mobile 2.0, refers to a perceived next generation of mobile internet services that leverage the social web, or what some call Web 2.0. The social web includes social networking sites and wikis that emphasise collaboration and sharing amongst users. Mobile Web 2.0, with an emphasis on Web, refers to bringing Web 2.0 services to the mobile internet, i.e., accessing aspects of Web 2.0 sites from mobile internet browsers.

By contrast, Mobile 2.0 refers to services that integrate the social web with the core aspects of mobility – personal, localized, always-on and ever-present. These services are appearing on wireless devices such as Smartphone‘s and multimedia feature phones that are capable of delivering rich, interactive services as well as being able to provide access and to the full range of mobile consumer touch points including talking, texting, capturing, sending, listening and viewing.

Enablers of Mobile 2.0

□ Ubiquitous Mobile Broadband Access

□ Affordable, unrestricted access to enabling software platforms, tools and technologies

c

- Open access, with frictionless distribution and monetization

Characteristics of Mobile 2.0

- The social web meets mobility
- Extensive use of User-Generated Content, so that the site is owned by its contributors
- Leveraging services on the web via mashups
- Fully leveraging the mobile device, the mobile context, and delivering a rich mobile user experience
- Personal, Local, Always-on, Ever-present

Implementations of Mobile 2.0

Mobile 2.0 is still at the development stage but there are already a range of sites available, both for so-called "smartphones" and for more ordinary "feature" mobile phones. The best examples are Micro-blogging services Jaiku, Twitter, Pownce, CellSpin, and open platforms for creating sms services like Fortumo and Sepomo or providing information and services like mobeedo. The largest mobile telecoms body, the GSM Association, representing companies serving over 2 billion users, is backing a project called Telco 2.0, designed to drive this area.

9. Specify and develop the six simple rules for user with excellent readability.

10. What Is Information Architecture?

The structural design of shared information environments

- The combination of organizations, labelling, search, and navigation systems within websites and intranets
- The art and science of shaping information products and experiences to support usability and find ability
- An emerging discipline and community of practice focused on bringing principles of design and architecture to the digital landscape

Information architecture

The organization of data within an informational space. In other words, how the user will get to

information or perform tasks within a website or application.

The role of information architecture is played by a variety of people, from product managers to designers and even developers. To make things more confusing, information architecture can be called many different things throughout the design and development process. Words like intuitive, simple, findable, usable, or the executive favourite easy to- use—all describe the role that information architects play in creating digital experiences.
The visual design of your product, what frameworks you use, and how it is developed are integral to the success of any product, but the information architecture stands apart as being the most crucial element of your product. It is the first line of scrimmage—the user‘s first impression of your product. Even if you have the best design, the best code, and the best backend service, if the user cannot figure out how to use it, she will fail and so will your product.

c

Mobile Information Architecture

Information architecture has become a common discipline in the web industry, unfortunately, the mobile industry like software has only a handful of specialized mobile information architects. Although mobile information architecture is hardly a discipline in its own right, it certainly ought to be. This is not because it is so dissimilar from its desktop cousin, but because of context, added technical constraints, and needing to display on a smaller screen as much information as we would on a desktop.

The role of a mobile information architect would be to interpret this content to the mobile context. Do you use the same structure, or sections? Do you present the same information above the fold? If so, how should that be prioritized? How does the user navigate to other areas? Do you use the same visual and interaction paradigms, or invent new ones? And if you do start to invent new paradigms, will you lose the visual characteristics of what users expect?

Keeping It Simple

When thinking about your mobile information architecture, you want to keep it as simple as possible.

Support your defined goals

If something doesn‘t support the defined goals, lose it. Go back to your user goals and needs, and identify the tasks that map to them. Find those needs and fill them.

Clear, simple labels

Good trigger labels, the words we use to describe each link or action, are crucial in Mobile. Words like —products‖ or —services‖ aren‘t good trigger labels. Users have a much higher threshold of pain when clicking about on a desktop site or application, hunting and pecking for tasty morsels. Mobile performs short, to-the-point, get-it-quick, and get-out types of tasks. What is convenient on the desktop might be a deal breaker on mobile.

11. Mention and analyze the seven principles of Web 2.0.

It’s a simply improved version of the first worldwide web, characterized specifically by the change from static to dynamic or user-generated content and also the growth of social media.

c

What are the examples of Web 2.0 applications?

Web 2.0 examples include hosted services (Google Maps),Web applications (Google Docs, Flickr), Video sharing sites (YouTube), wikis (MediaWiki), blogs (WordPress), social networking (Facebook), folksonomies (Delicious), Microblogging (Twitter), podcasting (Podcast Alley) & content hosting services and many more.

So the major difference between web 1.0 and web 2.0 is that web 2.0 websites enable users to create, share, collaborate and communicate their work with others, without any need of any web design or publishing skills. These capabilities were not present in Web 1.o environment.

Principles:

1. The Web As Platform

The web is the only true link that unites us all together whoever we are and wherever we are in the world. Hence, to harness collective intelligence and to create the intelligent web – we need to include as many people as we can. The only way we can do this is to treat the web as a platform and use open standards. You can't harness collective intelligence using the ESA/390 - however powerful it is!

2. Harnessing Collective Intelligence

Now becomes the 'main' principle or the first principle

3. Data is the Next Intel Inside

By definition, to harness collective intelligence – we must have the capacity to process massive amounts of data. Hence, data is the 'intelligence' (Intel)

4. End of the Software Release Cycle

This pertains to 'Software as a service'. Software as a 'product' can never keep upto date with all the changing information.

Ofcourse in the web 2.0 sense, we are dealing with code as well as data – so the service concept keeps the data relevant (and the harnessed decision accurate) by accessing as many sources as possible

5. Lightweight Programming Models

The heavy weight programming models catered for the few. In contrast, using lightweight programming models we can reach many more people(hence sources of information – to enable data collection and a more intelligent web).

6. Software Above the Level of a Single Device

More devices to capture information and better flow of information between these devices leads to a higher degree of collective intelligence

7. Rich User Experiences

A rich user experience is necessary to enable better web applications leading to more web usage and better information flow on the web – leading ofcourse to a more 'Intelligent' web.

1. Give the types of selection patterns.(reference: anna-pg no. 149)

DIRECT SELECTION

□ **Toggle Selection:** Checkbox or control-based selection.

□ **Collected Selection:** Selection that spans multiple pages.

□ **Object Selection:** Direct object selection.

□ **Hybrid Selection:** Combination of Toggle Selection and Object Selection

Toggle Selection

The most common form of selection on the Web is Toggle Selection. Checkboxes and toggle buttons are the familiar interface for selecting elements on most web pages.

The way to select an individual mail message is through the row‘s checkbox. Clicking on the row itself does not select the message. We call this pattern of selection Toggle Selection since toggle-style controls are typically used for selecting items.

Once items have been check-selected, actions can be performed on them. Usually these actions are performed on the selection by clicking on a separate button (e.g., the Delete button). Gmail is a good example of actions in concert with Toggle Selection

Collected Selection

Toggle Selection is great for showing a list of items on a single page. But what happens if you want to collect selected items across multiple pages? Collected Selection is a pattern for keeping track of selection as it spans multiple pages.

In Gmail, you can select items as you move from page to page. The selections are remembered for each page. If you select two items on page one, then move to page two and select three items, there are only three items selected. This is because actions only operate on a single page. This makes sense, as users do not normally expect selected items to be remembered across different pages.

Object Selection

Object Selection, is when selection is made directly on objects within the interface. Sometimes using a checkbox does not fit in with the style of interaction desired. Laszlo‘s WebTop mail allows the user to select messages by clicking anywhere in the row

Desktop-style selection

For now Object Selection is not as common on the Web. Given that most sites have been content-oriented, there have been few objects to select. Also, with the Web‘s simple event model, Object Selection was not easy to implement. In typical web pages, keyboard events have rarely made sense since they are also shared with the browser.

Object Selection interactions include ways to use the mouse to drag-select objects. Yahoo! Photos introduced this same type of object selection to its photo gallery (Figure below). Individually clicking on a photo selects it. Using the Shift key and clicking also extends the selection. In addition, using the Control key and clicking discontinuously selects photos. And like most desktop applications, you can drag a selection box around a group of items to add them to the selected set (in this case, photos).

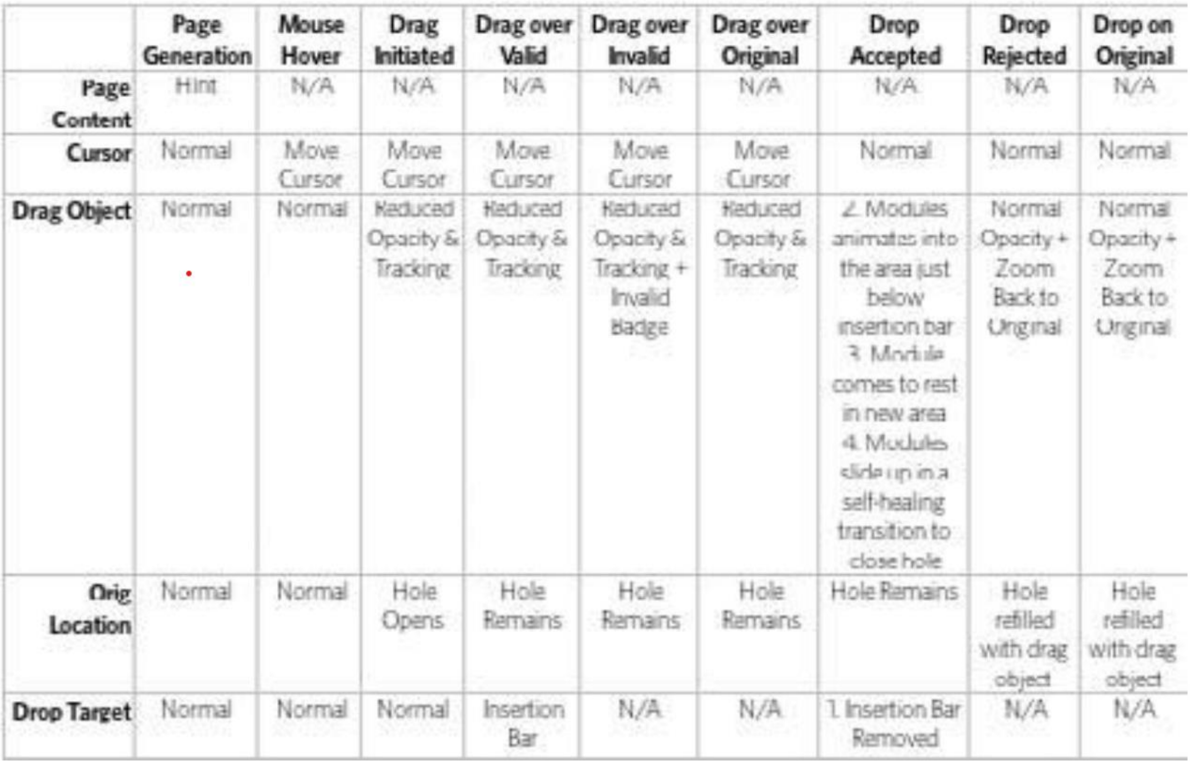
Hybrid Selection

Mixing Toggle Selection and Object Selection in the same interface can lead to a confusing interface. Referring back to Yahoo! Bookmarks, you‘ll see an odd situation arise during drag and drop

2.Tabulate some page elements. Pg: 138

The page elements available include:

- Page (e.g., static messaging on the page)
- Cursor
- Tool Tip
- Drag Object (or some portion of the drag object, e.g., title area of a module)
- Drag Object's Parent Container
- Drop Target



	Page Generation	Mouse Hover	Drag Initiated	Drag over Valid	Drag over Invalid	Drag over Original	Drop Accepted	Drop Rejected	Drop on Original
Page Content	Hint	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Cursor	Normal	Move Cursor	Move Cursor	Move Cursor	Move Cursor	Move Cursor	Normal	Normal	Normal
Drag Object	Normal	Normal	Reduced Opacity & Tracking	Reduced Opacity & Tracking	Reduced Opacity & Tracking + Invalid Badge	Reduced Opacity & Tracking	2. Modules animates into the area just below insertion bar 3. Module comes to rest in new area 4. Modules slide up in a self-healing transition to close hole	Normal Opacity + Zoom Back to Original	Normal Opacity + Zoom Back to Original
Orig Location	Normal	Normal	Hole Opens	Hole Remains	Hole Remains	Hole Remains	Hole Remains	Hole refilled with drag object	Hole refilled with drag object
Drop Target	Normal	Normal	Normal	Insertion Bar	N/A	N/A	1. Insertion Bar Removed	N/A	N/A

3.Analyze Toggle selection pg: 149

Toggle Selection

The most common form of selection on the Web is Toggle Selection. Checkboxes and toggle buttons are the familiar interface for selecting elements on most web pages.

The way to select an individual mail message is through the row's checkbox. Clicking on the row itself does not select the message. We call this pattern of selection Toggle Selection since toggle-style controls are typically used for selecting items.
Once items have been check-selected, actions can be performed on them. Usually these actions are performed on the selection by clicking on a separate button (e.g., the Delete button). Gmail is a good example of actions in concert with Toggle Selection

Considerations

Toggle Selection with checkboxes has some nice attributes: Toggle Selection with checkboxes has some nice attributes:

Toggle Selection with checkboxes has some nice attributes:

□ Clear targeting, with no ambiguity about how to select the item or deselect it.

□ Straightforward discontinuous selection, and no need to know about Shift or Control key ways to extend a selection. Just click the checkboxes in any order, either in a continuous or discontinuous manner.

□ Clear indication of what has been selected.

4. Give an example of virtual scrolling. Pg: 171

The classic example of this is Google Search.

5. State and discover Fitt's law.

Fitts' law states that the amount of time required for a person to move a pointer (e.g., mouse cursor) to a target area is a function of the distance to the target divided by the size of the target. Thus, the longer the distance and the smaller the target's size, the longer it takes.
In 1954, psychologist Paul Fitts, examining the human motor system, showed that the time required to move to a target depends on the distance to it, yet relates inversely to its size. By his law, fast movements and small targets result in greater error rates, due to the speed-accuracy trade-off.
Fitts' law is widely applied in user experience (UX) and user interface (UI) design. For example, this law influenced the convention of making interactive buttons large (especially on finger-operated mobile devices)—smaller buttons are more difficult (and time-consuming) to click. Likewise, the distance between a user's task/attention area and the task-related button should be kept as short as possible.
The law is applicable to rapid, pointing movements, not continuous motion (e.g., drawing). Such movements typically consist of one large motion component (ballistic movement) followed by fine adjustments to acquire (move over) the target.

In the Fitts' Law description of pointing, the parameters of interest are:

a.The time to move to the target
b. The movement distance from the starting position to the target center
c. Target width

By following Fitts' law, standard interface elements such as the right-click pop-up menu or short drop-down menus have had resounding success, minimizing the user's travel distance with a mouse in selecting an option—reducing time and increasing productivity. Conversely, long drop-downs, title menus, etc., impede users' actions, raising movement-time demands.

6. Demonstrate Antipattern?

The idea of patterns in HCI developed from the idea of patterns in SE, although there are marked differences [Borchers 2000]. In order to identify something as a pattern it is necessary to identify a particular pattern, show that it exists in a

successful system and also show that those patterns are absent in unsuccessful systems [Long 2001]. Likewise, the presence of anti-patterns must be identified in unsuccessful systems, and their absence shown in successful systems in order for them to qualify as anti-patterns. The basic rationale in publishing anti-patterns is to identify recurring design flaws for the purpose of preventing other people from making the same mistakes [McCormick 2004A].
An anti-pattern has two possible forms: it either ‘provides knowledge on how to go from a problem to a bad solution’ or shows ‘how to go from a bad solution to a good solution’ [Cockburn et al. 2004]. The former will be referred to as a simple anti-pattern and the latter is commonly called an amelioration anti-pattern. If described properly, an anti-pattern also tells the designer why the bad solution looks attractive, why it turns out to be bad, and what positive are applicable in its stead. Anti-patterns therefore concentrate on presenting negative solutions [Brown et al. 1998, McCormick 2004A]. Simple anti-patterns are thus not very useful to the designer, behaving as a mere example of what can go wrong; the amelioration pattern is constructive and useful to the designer since it shows how the bad solution can be refactored.

7.Summarize Inlay Versus Overlay.

8. What is Drag rendering? Pg: 143

Drag rendering

How should the dragged object be represented? Should it be rendered with a slight transparency (ghost)? Or should it be shown fully opaque? Should a thumbnail representation be used instead?

9. Develop Lightweight overlays. Pg:163

Lightweight overlays are shown within the browser page as an overlay

Input Overlay is a lightweight overlay that brings additional input information for each field tabbed into. American Express uses this technique in its registration for premium cards such as its gold card

By using either Flash or Ajax-style techniques (Dynamic HTML), a web application can present a pop up in a lightweight overlay within the page itself. This has distinct advantages:
• Lightweight overlays are just a lightweight in-page object. They are inexpensive to create and fast to display.

□ The interface for lightweight overlays is controlled by the web application and not the browser.

□ There is complete control over the visual style for the overlay. This allows the overlay to be more visually integrated into the application‘s interface

10. Write short notes on Drag and Drop module.

There is a lot in this. So choose your content from page number 137

11. Carousel
    Carousels allow multiple pieces of content to occupy a single, coveted space. This may placate corporate infighting, but on large or small viewports, people often scroll past carousels. A static hero or integrating content in the UI may be better solutions. But if a carousel is your hero, good navigation and content can help make it effective.

But the most ubiquitous type is the feature area carousel (as discussed in this article), which usually possesses the following traits:

- appears toward the top of the homepage
- occupies a substantial section of the “above fold” area
- displays more than one piece of content in the same place (one at a time)
- offers some indication (or navigation) that there is more than one piece of featured content, or frame, within the carousel
- contains images and a small amount of text in each frame
- includes content about the organization’s brand or mission, featured information, or
- consists of promotions.

Greatest Benefits of Carousels

The greatest benefit of using carousels is that they **enable more than one piece of content to occupy the same piece of prime real estate** on the homepage, which can help diffuse any infighting about whose content is most deserving. An additional benefit is that because more information appears near the top of the viewable area, there may be greater opportunity for people to actually see it.