

UNIT IV:

4.1 User Interface Design: Making Wireless Applications, Easy to Use

What is user interface design?

User interface design or UI design generally refers to the visual layout of the elements that a user might interact with in a website, or technological product. This could be the control buttons of a radio, or the visual layout of a webpage. User interface designs must not only be attractive to potential users, but must also be functional and created with users in mind.

A fundamental reality of application development is that the user interface is the system to the users. What users want is for developers to build applications that meet their needs and that are easy to use.

User interface design important for several reasons. First of all the more intuitive the user interface the easier it is to use, and the easier it is to use and the less expensive to use it. The better the user interface the easier it is to train people to use it, reducing your training costs. The better your user interface the less help people will need to use it, reducing your support costs. The better your user interface the more your users will like to use it, increasing their satisfaction with the work that you have done.

4.2 Web Site Design: Computer Terminals Vs Mobile Terminals

User interface design for display on computer terminal is very different from user interface design for display on a mobile terminal for several reasons

The device is different

The Network is different

The user is different

The Device is Different

Traditional Web sites are: designed to be shown on desktop computer that have highly capable displays. These computer monitors typically can display anywhere from hundreds to millions of colors, and they have resolutions ranging from 640 pixels wide by 480 pixels high to 1600 pixels wide by 1200 pixels high-and someday even more. Currently, Web sites and applications also expect users to have high” capable input devices (such as a mouse or trackball), which enable the users to point to any group of pixels on the Web page and select them. In addition, Web content assumes the ability to easily input text (using, for example, a full-function QWERTY keyboard). Furthermore, content developers assume that computing

resources are powerful, with capable central processing units and considerable memory (both RAM and ROM).

WAP, however, is designed for use on a wide variety of mobile terminals, ranging from pagers to handsets to Personal Digital Assistants (PDAs). Each of these mobile terminals has display capabilities that are much more limited than those of computer monitors. Likewise, the mobile displays themselves vary widely. For example, some handsets have text-only displays that offer 4 lines of text and 12 characters per line, whereas the most capable handsets provide graphical displays that are 120 pixels wide by 160 pixels high and display 11 lines of text.

The Network Is Different

Web sites are designed with the expectation of an Internet connection speed ranging from 28.8 kilobits per second to more than 1 megabit per second. Reflecting this expectation, they tend to include detailed graphics, multimedia, and downloadable applets that run on the client PC. However, mobile data speeds today provide an effective throughput ranging from less than 100 bits per second to 14.4 kilobits per second (although higher data speeds are around the corner). In addition, mobile data networks tend to have more latency, less connection stability, and less predictable availability.

The User Is Different

Perhaps the most important difference, however, is the user. Mobile terminal users may not necessarily have any desktop computing experience, so standard computer user interface metaphors are not always appropriate. They also tend to be engaged in other activities while they are using their handset, so they cannot give full attention to the handset. Furthermore, mobile terminal users expect to be able to complete a task with their handset within a few minutes. They are unlikely to spend extended periods of time "surfing the Net" from a mobile device. Thus applications need to enable users to complete their tasks quickly and efficiently.

The Interface Should Be Different!

Designing applications for display on a mobile terminal is very different from designing applications for display on a computer terminal. Hence! much of the experience in designing usable Web sites cannot be carried directly from the internet world to the mobile world.

4.3 Designing a Usable WAP Site

Usability has many aspects. It refers to how easy the user interface is to learn for novice and casual users. It refers to how efficient, flexible and Powerful the user interface is to use for expert users. It refers to how low the error rate is, how satisfied users are with their experience in using the interface, and how much learning they retain over time.

Human factors is a discipline dedicated to improving the many components that define usability. Human factors professionals design products that people use and the environments in which they use them so as to better match the abilities, limitations, and needs of those people. These Professionals achieve their goals by applying extensive knowledge about human capabilities, limitations, and behaviour to product design.

Usability is a fundamental feature of any mobile Internet service, and it influences the operation of all other features of your service. To design easy-to-use interfaces, usability needs to be a primary goal throughout the application development process. Like any fundamental feature, usability is most effective When designed early into the process and very costly When begun late in the development cycle. That is, usability is not something that can be added toward the end of a development cycle, like shrink-wrapping. '

Designing usability early into the process requires at least two things:

An overall development process that is focused on the end user and employs structured usability techniques

Knowledge and application of known user interface design principles and guidelines

4.4 Structured Usability Methods

The following characteristics of a user interface can signal problems with Usability

It was designed by software people, not human-compute-interaction specialists.

It was developed by strict top-down, functional decomposition

It was not developed to meet written, measurable usability Specifications.

It was not prototyped as it was being developed.

It was not developed by means of an iterative refinement process.

It was not empirically evaluated.

Design Process

An Effective user interface design process will contain a number of important phases

Phase 1: Define

Phase 2: Design

Phase 3: Prototype

Phase 4: Test

Phase 5: Iterate

Phase 1: Define

In this phase service is defined and the requirements are gathered. Service definition involves gathering numerous design-affecting pieces of information. Those design-affecting pieces of information include the following items:

- ✓ User Profile
- ✓ Task Analysis
- ✓ Platform capabilities and constraints

User Profile

User profile is a description of the target population couched in terms of the characteristics that are most important to the user interface design. Creating the profile involves gathering data about user demographics, skills, knowledge, background, and other factors that characterize your potential customers. The user profile should describe each major skill required by the application and, if possible, the experience and approximate percentage of end users at high and low levels of experience. You can get this information through surveys, customer interviews, market research analyses, focus group sessions, contextual observations, and other research tools.

Once this data has been gathered, it can be presented in multiple Ways for use by the entire design team. For example, each characteristic can be summarized in a table that identifies the relevant percentages.

Task Analysis

A task analysis results in a description of how users perform tasks that are relevant to the application you are designing. It involves understanding which methods are used, why they are important, how the informal flows, what the user does, and what can be automated. Information from a task analysis allows the designers to concentrate on ensuring the features of greatest importance to the users are the easiest to use. The most effective task analysis is carried out by watching users Perform those tasks that are relevant to your application and documenting what they do. Alternatively, customer interviews, focus groups, surveys, and so on can be used.

Some of the questions that should be answered through a task analysis include the following:

What tasks do users perform?

What tasks are most critical?

What steps are taken to perform tasks?

What are users' goals for performing tasks?

What information is needed to complete tasks?

How frequently do users perform tasks?

What tools are used to complete task?

What output is generated from user tasks?

Platform capabilities and constraints

Requirement for designing WML-based services

Phase 2: Design

Mobile internet is very different from wired internet .The device is different , the network is different and the users are different.

Application Design Goals

Be targeted: Provide only the most valuable information to a Wireless user

Be fast: Make the most valuable information the easiest to obtain thus enabling users to quickly find that information in just a few key clicks, rather than through browsing.

Be personal: Create an intelligent application and pm information that is adapted to each user's personal interests.

Be simple: Ensure that information is easy to find and use. Your application cannot just deliver technology; it must be integrated and cam fully designed to fit users' needs.

Be urgent: Provide information that is critical to the end users and needs to be received while they are mobile

Be timely: Provide information that changes and needs to be received Wirelessly because timing is important. Static information can be obtained elsewhere.

Adapt to the way people work: Do not attempt to change people worlds. Instead, try to be a part of their environment. People will adopt the mobile Internet when they can just try it out, find that it is important and useful, and easily merge it into their daily lives.

Use the Web to do your users' work for them: Design applications that can perform tasks for your users (for example, alert users when an event of importance has occurred, such as a stock price reaching a particular level, so they do not have to keep checking manually).

Personalize Your Application

Personalize Your Application Personalization is one of the most important design principles. Personalized applications increase stickiness (reduce churn), make applications easier and faster to use, optimize navigation by reducing key clicks, generate a sense of ownership in users, offer users a superior service experience that they cannot get elsewhere, reduce data entry, and increase usage and customer loyalty. In fact, personalization for WML applications often distinguishes a usable application from an unusable one. Ultimately, the wireless device is too constrained for "general" use.

Personalization is anything that customizes the application for the end user. Personalized services can learn from the user and adapt to him or her based on the user's explicit preferences and implicit behaviors demonstrated through real-life use of the service. If the user must perform complex or numerous activities to personalize the application, the use of a personal computer connected to the wired Internet may be necessary. For example, the user may need to use a PC to create a stock port

Know Your User

Use the information collected in the definition phase about target audience to accomplish following goals:

- ✓ Target the information you Present
- ✓ Adapt Your Application to the way your customers work
- ✓ Present information that is relevant to a mobile user

Reduce Keystrokes

The following methods can be used to reduce keystrokes

- ✓ Personalize your application
- ✓ Provide value at every level
- ✓ Optimize navigation to the most important content
- ✓ Keep the content targeted
- ✓ Allow users to select from lists
- ✓ Take users directly to the information

Phase 3: Prototype

Prototype serves many important functions and is critical part of the development process. They provide an effective tool for communicating design. They also enable

designers to better visualize design and task flow, and they provide a low cost vehicle for getting end user input regarding the proposed design.

Phase 4: Test

Usability testing is a way to see how easy to use something is by testing it with real users. Users are asked to complete tasks, typically while they are being observed by a researcher, to see where they encounter problems and experience confusion.

Usability Test Process

Five Steps to Conducting a Successful Usability Test

- ✓ Write up your tasks. Work with the product owner and designer to come up with tasks for your usability test. ...
- ✓ Decide on your method. You should not wait to do a usability test until the product is built. ...
- ✓ Select participants. Figure out the best, most realistic participants. ...
- ✓ Conduct the tests. ...
- ✓ Compile your results.

Phase 5 : Iterate

An iteration refers to a refinement of the service's definition and design, resulting in an updated prototype and a subsequent usability test my identify new usability issues and evaluate the effect of the design changes. Iterative design enables testing to occur in different stages of the product development, facilitating earlier detection and simplifying resolution of usability issues. It also helps to identify usability issues that may have been "hidden" behind other usability issues. For example, if an early usability test shows that the target population has difficulty understanding the terminology used by the service, then changing that terminology may later help the design team see that users also struggle with another part of the interface (for example, inefficient categorization of menu items). The latter usability issue could not be identified in the earlier test because users were confused by the terminology and thus could not evaluate the categorization of the menu items.

4.5 User Interface Design Guidelines

General Design Guidelines

1. Know thy user and you are not thy user
2. Things that look the same should act the same consistency .

3. The information for the decision needs to be there when the decision is made. Provide users with all of the relevant information you have to aid them in making the decisions required by your system
4. Prevent errors when possible. When it is not possible to prevent errors, error messages should actually mean something to the user and tell him or her how to fix the problem.
5. Everyone makes mistakes, so every mistake should be fixable. Allow users to recover from errors. Enable them to undo previous actions. Require confirmation of actions that would be particularly destructive if users took that action by accident (for example, "delete all messages"), and clearly state the results of that action in the user's terminology. If it is impossible to recover from the results of that action, clearly state this fact in the confirmation message as well.
6. Don't overload the user's buffers. Minimize the need for a mighty memory.
7. Keep it simple. Make tasks easy to perform.
8. Every action should have a reaction
9. The user should always know what is happening. The system's reaction to user commands should not shock or surprise users.
10. The more you do something, the easier it should be to do.

WML Design Guidelines

1. Allow the users to complete their tasks quickly
2. Keep your application as simple as possible
3. Whenever possible, avoid requiring users to enter text
4. Reduce Keystrokes
5. Design for smaller phones; there will be more of them and they are among the most constrained
6. Avoid unwrapped lines
7. Do not expect the browser menu on devices with phone.com
8. Avoid complicated screens
9. If a list contains more than two or three items, ensure that it is obvious to the user that scrolling is required
10. Allow users full control over items that they receive via push notification

4.6 Design Guidelines for Selected WML Elements.

Navigation

Navigation element can be used to move from one card to another.

WML <anchor> Tag

The <anchor>...</anchor> tag pair is used to create an anchor link. It is used together with other WML elements called <go/>, <refresh> or <prev/>. These elements are called task elements and tell WAP browsers what to do when a user selects the anchor link

You can enclose Text or image along with a task tag inside <anchor>...</anchor> tag pair.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
title	cdata	Defines a text identifying the link
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.

Example:

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="Anchor Element">
<p>
  <anchor>
    <go href="nextpage.wml"/>
  </anchor>
</p>
<p>
```

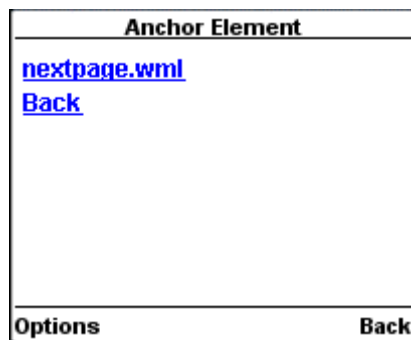
```

<anchor>
  <prev/>
</anchor>
</p>
</card>

</wml>

```

This will produce the following result:



WML <a>Tag

The <a>... tag pair can also be used to create an anchor link and always a preferred way of creating links.

You can enclose Text or image inside <a>... tags.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
href	URL	Defines URL of the liked page
title	cdata	Defines a text identifying the link
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

```
<?xml version="1.0"?>
```

```

<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

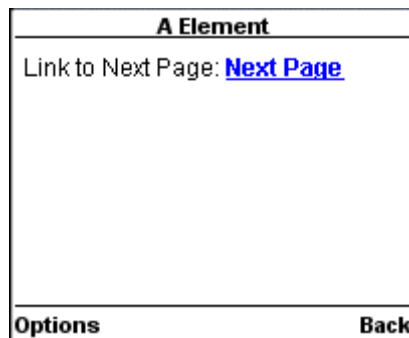
<wml>

<card title="A Element">
<p> Link to Next Page:
  <a href="nextpage.wml">Next Page</a>
</p>
</card>

</wml>

```

This will produce the following result:



WML <do> Tag

The <do> tag can be used to activate a task when the user clicks on a word/phrase on the screen.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
name	text	Sets a name for the <do> element.
label	string	Sets a label for the <do> element.
type	<ul style="list-style-type: none"> • accept • prev • help 	Defines the type of the <do>

	<ul style="list-style-type: none"> • reset • options • delete • unknown • x-* • vnd.* 	
value	number	Specifies the timer after which timer will be expired. Timeouts are specified in units of a tenth of a second.
class	class_data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of <do> element along with <go> element.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN"
"http://www.wapforum.org/DTD/wml13.dtd">

<wml>
  <template>
    <do name="main_menu" type="accept" label="Chapters">
      <go href="chapters"/>
    </do>
    <do name="menu_1" type="accept" label="Chapter 1">
      <go href="#chapter1"/>
    </do>
    <do name="menu_2" type="accept" label="Chapter 2">
      <go href="#chapter2"/>
    </do>
    <do name="menu_3" type="accept" label="Chapter 3">
```

```
<go href="#chapter3"/>
</do>
<do name="menu_4" type="accept" label="Chapter 4">
  <go href="#chapter4"/>
</do>
</template>

<card id="chapters" title="WML Tutorial">
  <p>
    Select One Chapter:<br/>
    <anchor>
      <go href="#chapter1"/>

      Chapter 1: WML Overview
    </anchor><br />
    <anchor>
      <go href="#chapter2"/>
      Chapter 2: WML Environment
    </anchor><br />

    <anchor>
      <go href="#chapter3"/>
      Chapter 3: WML Syntax
    </anchor><br />

    <anchor>
      <go href="#chapter4"/>

      Chapter 4: WML Elements
    </anchor><br />
  </p>
</card>
```

```
<card id="chapter1" title="WML Tutorial Ch1">
  <p>
    <em>Chapter 1: WML Introduction</em><br/>
    ...
  </p>
</card>

<card id="chapter2" title="WML Tutorial Ch2">
  <p>
    <em>Chapter 2: WML Environment</em><br/>
    ...
  </p>
</card>

<card id="chapter3" title="WML Tutorial Ch3">
  <p>
    <em>Chapter 3: WML Syntax</em><br/>
    ...
  </p>
</card>

<card id="chapter4" title="WML Tutorial Ch4">
  <p>
    <em>Chapter 4: WML Elements</em><br/>
    ...
  </p>
</card>
</wml>
```

This will produce the following menu and now you can navigate through all the chapters:

WML Tutorial	
Select One Chapter:	
Chapter 1: WML Overview	
Chapter 2: WML Environment	
Chapter 3: WML Syntax	
Chapter 4: WML Elements	
Options	Back

WML <onevent> Tag

The <onevent>...</onevent> tags are used to create event handlers.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
type	<ul style="list-style-type: none"> • onenterbackward • onenterforward • onpick • ontimer 	Defines a type of event occurred.
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>
<onevent type="onenterbackward">
  <go href="#card3"/>
</onevent>

<card id="card1" title="Card 1">
<p>
  <anchor>
```

```

    <go href="#card2"/>
    Go to card 2
  </anchor>
</p>

</card>
<card id="card2" title="Card 2">
<p>
  <anchor>
  <prev/>
  Going backwards
  </anchor>
</p>
</card>

<card id="card3" title="Card 3">
<p>
Hello World!
</p>
</card>
</wml>

```

User Input

WML <input> Tag

The <input/> element is used to create input fields and input fields are used to obtain alphanumeric data from users.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
name	text	The name of the variable that is set with the result of the user's input
maxlength	number	Sets the maximum number of characters the

		user can enter in the field
emptyok	<ul style="list-style-type: none"> • true • false 	Sets whether the user can leave the input field blank or not. Default is "false"
format	A a N X x M m *f nf	Sets the data format for the input field. Default is "*M". A = uppercase alphabetic or punctuation characters a = lowercase alphabetic or punctuation characters N = numeric characters X = uppercase characters x = lowercase characters
M = all characters m = all characters *f = Any number of characters. Replace the <i>f</i> with one of the letters above to specify what characters the user can enter nf = Replace the <i>n</i> with a number from 1 to 9 to specify the number of characters the user can enter. Replace the <i>f</i> with one of the letters above to specify what characters the user can enter		
size	number	Sets the width of the input field
tabindex	number	Sets the tabbing position for the select element
title	text	Sets a title for the list

type	<ul style="list-style-type: none"> • text • password 	Indicates the type of the input field. The default value is "text". Password field is used to take password for authentication purpose.
value	text	Sets the default value of the variable in the
"name" attribute		
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="Input Fields">
<p> Enter Following Information:<br/>
Name: <input name="name" size="12"/>
Age : <input name="age" size="12" format="*N"/>
Sex : <input name="sex" size="12"/>
</p>
</card>
```

</wml>

This will provide you the following screen to enter required information:

Input Fields	
Enter Following Information:	
Name:	<input type="text"/>
Age:	<input type="text"/>
Sex:	<input type="text"/>
Options	Back

WML <select> Tag

The <select>...</select> WML elements are used to define a selection list and the <option>...</option> tags are used to define an item in a selection list. Items are presented as radio buttons in some WAP browsers. The <option>...</option> tag pair should be enclosed within the <select>...</select> tags.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
iname	text	Names the variable that is set with the index result of the selection
ivalue	text	Sets the pre-selected option element
multiple	<ul style="list-style-type: none">• true• false	Sets whether multiple items can be selected. Default is "false"
name	text	Names the variable that is set with the result of the selection
tabindex	number	Sets the tabbing position for the select element
title	text	Sets a title for the list
value	text	Sets the default value of the variable in the "name"

attribute		
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="Selectable List">
<p> Select a Tutorial :
<select>
<option value="htm">HTML Tutorial</option>
<option value="xml">XML Tutorial</option>
<option value="wap">WAP Tutorial</option>
</select>
</p>
</card>

</wml>
```

When you will load this program, it will show you the following screen:

The screenshot shows a WML screen with the following elements:

- Title:** Selectable List
- Text:** Select a Tutorial :
- Form:** A dropdown menu with the text "HTML T..." and a downward arrow.
- Buttons:** "Options" and "Back" buttons at the bottom of the screen.

Appearance and Presentation

WML Tag

The element is used to include an image in a WAP card. WAP-enabled wireless devices only supported the Wireless Bitmap (WBMP) image format.

WBMP images can only contain two colors: black and white. The file extension of WBMP is ".wbmp" and the MIME type of WBMP is "image/vnd.wap.wbmp".

Attributes:

This element supports the following attributes:

The element supports the following attributes:

Attribute	Value	Description
align	<ul style="list-style-type: none">• top• middle• bottom	Alignment of the image
alt	alternative text	Sets an alternate text to be displayed if the image is not displayed.
height	<ul style="list-style-type: none">• px• %	Height of the image in pixels or percentage. If you specify the value in pixels, the syntax is "140", instead of "140px".
hspace	<ul style="list-style-type: none">• px• %	Sets white space to the left and right of the image. If you specify the value in pixels, the syntax is
ocalsrc	cdata	Sets an alternate representation for the image. If this attribute is set, the browser will use it instead of the "src" attribute.
src	image url	A path to wbmp image.
vspace	<ul style="list-style-type: none">• px• %	Sets white space above and below the image. If you specify the value in pixels, the syntax is "140", instead of "140px".
width	<ul style="list-style-type: none">• px• %	Sets the width of the image.If you specify the value in pixels, the syntax is "140", instead of "140px".

xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element.

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="WML Images">
<p>
This is Thumb image


</p>

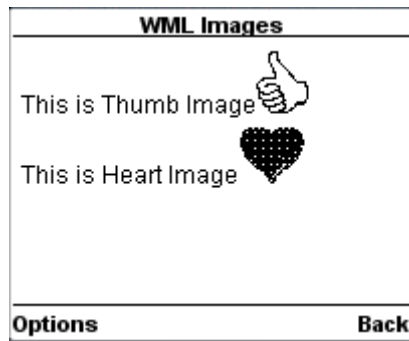
<p>
This is Heart image

</p>

</card>

</wml>
```

This will produce the following result:



WML <p> Tag

The <p> element defines a paragraph of text and WAP browsers always render a paragraph in a new line.

A <p> element is required to define any text, image or a table in WML.

Attributes:

The <p> element supports the following attributes:

Attribute	Value	Description
align	<ul style="list-style-type: none"> • left • right • center 	This is used to change the horizontal alignment of a paragraph.
mode	<ul style="list-style-type: none"> • wrap • nowrap 	Sets whether a paragraph should wrap lines or not.
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of <p> element.

```
<?xml version="1.0"?>

<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">
```

```

<wml>

<card title="Paragraph Example">
  <p align="center">
    This is first paragraph
  </p>
  <p align="right">
    This is second paragraph
  </p>
</card>

</wml>

```

This will produce the following result:

Paragraph Example	
<p style="text-align: center;">This is first paragraph</p> <p style="text-align: right;">This is second paragraph</p>	
Options	Back

WML <table> Tag

The <table> element along with <tr> and <td> is used to create a table in WML. WML does not allow the nesting of tables

A <table> element should be put with-in <p>...</p> elements.

Attributes:

The <table /> element supports the following attributes:

Attribute	Value	Description
columns	number	Sets the number of columns in the table
align	<ul style="list-style-type: none"> • L • C • R 	To specify the horizontal text alignment of the columns, you need to assign three letters to the align attribute. Each letter represents the horizontal text alignment of a column. The

		<p>letter can be L, C, or R. For example, if you want the following settings to be applied to your table:</p> <ul style="list-style-type: none"> • First table column -- Left-aligned • Second table column -- Center-aligned • Third table column -- Right-aligned <p>Then you should set the value of the <i>align</i> attribute to LCR.</p>
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of <table> element.

```

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="WML Tables">
<p>
<table columns="3" align="LCR">
    <tr>
        <td>Col 1</td>
        <td>Col 2</td>
        <td>Col 3</td>
    </tr>

    <tr>
        <td>A</td>
        <td>B</td>
        <td>C</td>
    </tr>

```

```

</tr>

<tr>
  <td>D</td>
  <td>E</td>

  <td>F</td>
</tr>
</table>
</p>
</card>

</wml>

```

This will produce the following result:

WML Tables		
Col 1	Col 2	Col 3
A	B	C
D	E	F
Options		Back

WML <card> Tag

The <card> element encloses a WML card within a deck. In addition, text and graphics enclosed within <p> elements, it may also contain a number of event bindings

Attributes:

The <card> element supports the following attributes:

Attribute	Value	Description
title	cdata	Gives a title to this card. This title is displayed in some way by the browser when the card is visible.
newcontext	<ul style="list-style-type: none"> • true • false 	Specifies that when this card is entered, the browser context should b

ordered	<ul style="list-style-type: none"> • true • false 	Provides a hint to the browser about how the card is organized. Set it to true if the card consists of a number of separate fields that should be dealt with in the order they appear in the card. Set it to false if the card contains optional fields or may be filled in out of order.
onenterforward	URL	Occurs when the user navigates into a card using a "go" task
onenterbackward	URL	Occurs when the user navigates into a card using a "prev" task
ontimer	URL	Occurs when a "timer" expires
xml:lang	language_code	Sets the language used in the element.
class	cdata	Sets a class name for the element.
id	element_ID	A unique ID for the element.

Example:

Following is the example showing usage of this element:

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">
<wml>

<card id="one" title="First Card">
<p>
This is the first card in the deck
</p>
</card>

<card id="two" title="Second Card">
```

```

<p>
This is the second card in the deck
</p>
</card>

</wml>

```

WML <fieldset> Tag

The <fieldset/> element is used to group various input fields or selectable lists.

Attributes:

This element supports the following attributes:

Attribute	Value	Description
title	text	Sets a title for the list
xml:lang	language_code	Sets the language used in the element
class	class data	Sets a class name for the element.
id	element ID	A unique ID for the element.

Example:

Following is the example showing usage of this element.

```

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml12.dtd">

<wml>

<card title="Grouped Fields">
<p>
<fieldset title="Personal Info">
Name: <input name="name" size="12"/>
Age : <input name="age" size="12" format="*N"/>
Sex : <input name="sex" size="12"/>

```

```
</fieldset>
</p>
</card>

</wml>
```

This will provide you the following screen to enter required information. This result may differ browser to browser.

Grouped Fields	
Personal Info	
Name:	<input type="text"/>
Age :	<input type="text"/>
Sex :	<input type="text"/>
Options	Back

UNIT V :

Wireless Telephony Applications: Overview of the WTA Architecture – WTA Client Framework – WTA Server & Security – Design Considerations – Application Creation Toolbox– Future WTA Enhancements.

The Mobile Internet Future: Better Content, Easier Access – Beyond Browsing – Beyond Cellular – Mobile Data Unleashed.

Text Books:

1. SandeepSinghal, Thomas Bridgman, LalithaSuryanarayana, Daniel Mauney, JariAlvinen, David Bevis, Jim Chan, Stefan Hild, "The Wireless Application Protocol", Pearson Education, 2003.

Reference Books:

1. Jochen Schiller, "Mobile Communications" Pearson Education, New Delhi,2003
2. William Stallings, "Wireless Communication & Networks" ,PHI,2004