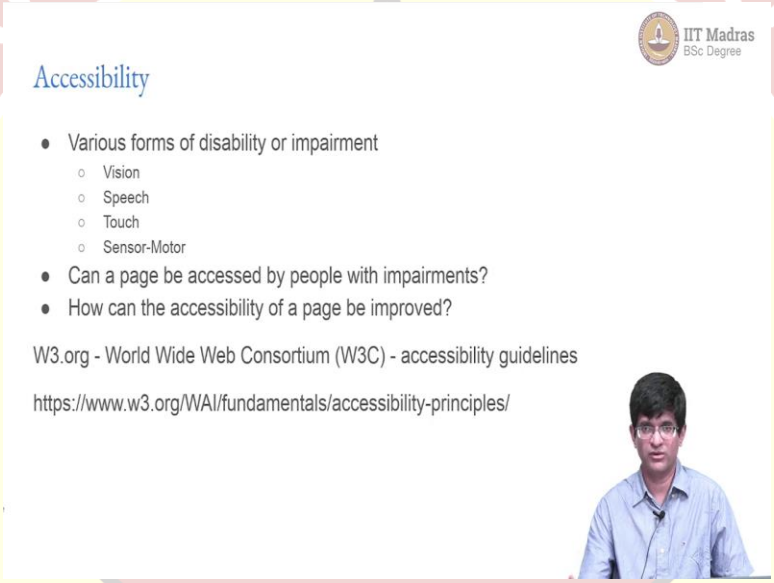
IIT Madras

ONLINE DEGREE

Modern Application Development - I
Professor Nitin Chandrachoodan
Department of Electrical Engineering
Indian Institute of Technology, Madras
Accessibility

Hello everyone, and welcome to this course on Modern Application Development. Now, we come to the last part of our discussion on views. And this is a very important component of how views are to be understood, and relates to the topic of accessibility.

(Refer Slide Time: 00:30)



Now, why is accessibility important, because increasingly people are aware of the fact that many people have various forms of disability or impairment. Most of the time we do not think of that, we see a website, it is designed for the majority of the people who are assumed to have normal eyesight, normal hearing, normal sense of touch, two hands, two legs and so on. But there might be sites which are actually useful, that may need to be used by a person who has some kind of a disability or an impairment.

Examples would be visual impairment, which need not mean complete blindness, it might even mean low vision, by low vision, it means that you perhaps are not able to make out contrast clearly, or you need the font sizes and text to be larger than would normally be required. There may be people with speech impairments, either they are not able to interact by using their voices, or they have trouble understanding the speech as read out by a computer.

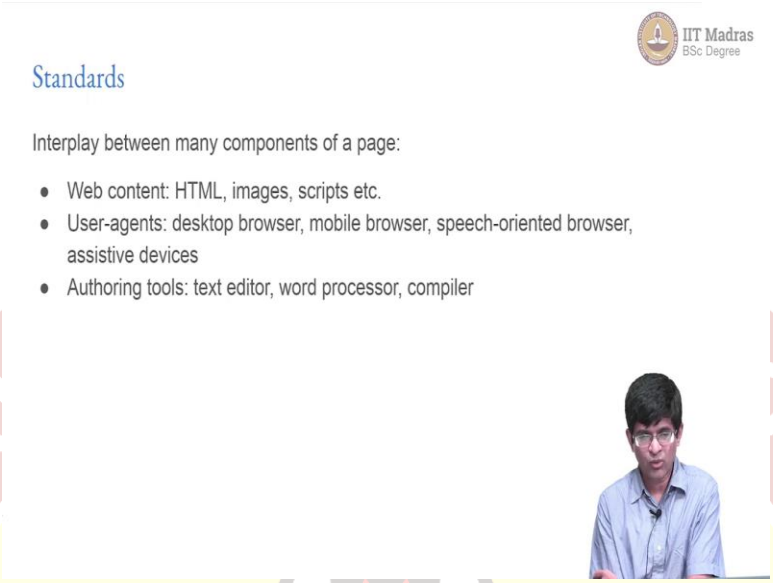
There might be people who are unable to use the touch interfaces, or have other problems with either sensor or motor disabilities. For there are ways by which many pages can be made accessible to different classes of people, it may not be possible to solve the problem for everyone. But it is definitely an important problem to keep in mind. And as far as possible, there should be an attempt to try and make any website or any application that you design in accessible to the greatest number of people.

So, the questions that we can ask are one of them is, can a page that you have designed, can an application that you ever design be accessed by people with any kind of impairments, what kind of problems would prevent a person from being able to actually understand what is on this site? Now clearly, a person with visual impairment may not be able to absorb all the aesthetics of a website, but that does not mean that they cannot get useful information out from it.

So at least it should be accessible the information there should be something that they can retrieve. And the related question over here is how can the accessibility of a given page be improved? Now, W3C, the World Wide Web Consortium with their website w3.org is sort of the governing body that gives the recommendations, there is nothing that actually sets out standardized rules as such, they are mostly called recommendations from the W3C.

And these W3C recommendations are what essentially drives the structure of the web as we know it. They have a number of accessibility guidelines, and this page that I have linked over here essentially gives the accessibility principles according to the W3C.

(Refer Slide Time: 03:38)



And, what I am going to do is sort of distill out some of the information which is there on that website, in such a way that it can be sort of briefly absorbed, but I would strongly encourage all of you to go read up more about it and understand it in more detail. The first thing to keep in mind is that accessibility is an interplay of many different components, on a webpage. The first and foremost, of course, is the webpage itself, that is to say the content, there is some HTML on it, there are some images on it, there is some various tags in the HTML, all of that is what comes from the server.

But there is also one more component which is the user agent, the client, the browser. Very often we assume that a person is going to be seeing this on client browser, which is on a desktop. And in fact, many of you may not, in fact, have even seen those kind of things. But there were, there was a time when the majority of websites actually had a small tag at the bottom saying best viewed on a browser with at least 800 cross 600 resolution.

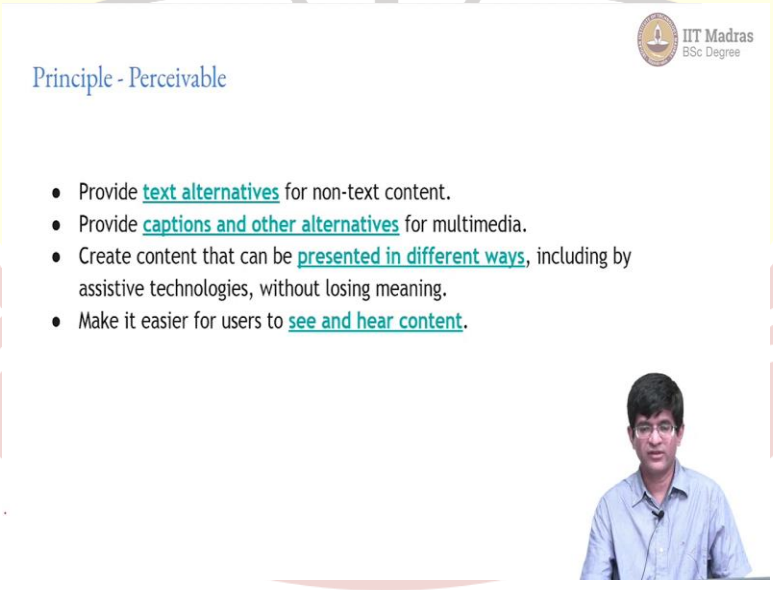
And nowadays, you just never see those for the simple reason that the majority of users are actually not viewing it on a desktop at all. They are viewing, viewing it from a mobile phone. Now the phones might have a resolution of 800 cross 600. But that can be like really tiny, you cannot really read much on an 800 cross 600 on a phone. So rather than talking about pixels, because the pixel densities are now so different on phones than they were on desktops, people usually talk in terms of having adapting to the user agent.

If it is a mobile browser, I should behave differently, or at least I should present the data differently, or there should be some way by which the browser can still present it in a useful manner. There could be visually impaired people who are completely browsing on a speech oriented browser, there are browsers that actually load a web page and read it out to you, they are screen readers, or even browsers that can read the information.

And they make use of a lot of useful tags and elements in the HTML, such as, for example, is there a parent link? Where did this come from? Is there a next page link? Is there something for navigating to the main content, all of that can be embedded within the tags in the HTML in order to make it easier to understand for a speech based reader.

And finally, what happens is that the authoring tools, the server is only putting out the HTML. But the tools that were used in order to generate the HTML in the first place, whether we use templates, or whether we use programmatic HTML generation, or whether we use something like you know, Microsoft Word and output an HTML, all of those have their impact on how accessible the final output is going to be.

(Refer Slide Time: 06:38)



So, with all of that in mind, I am just going to briefly review some of the core principles as described on the W3C website. The first class of principles is what are called the perceivable, that is directly perceivable by the end user. And one of them is to say, always, whenever you

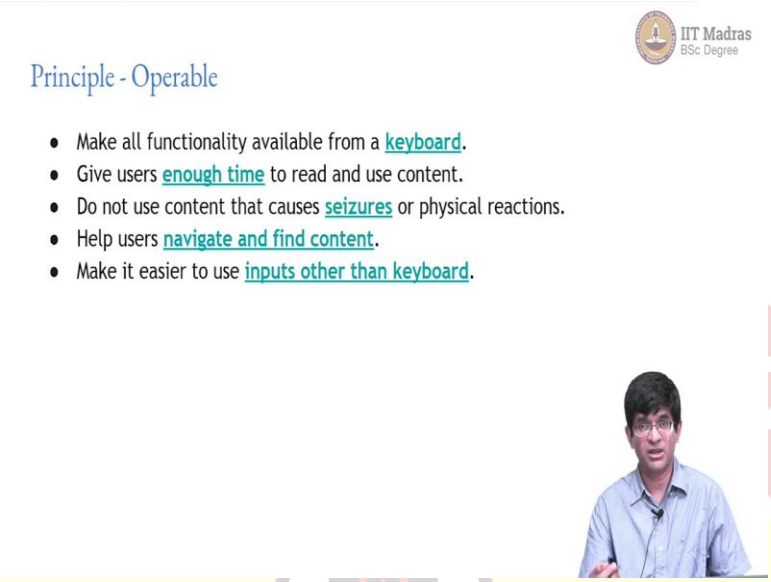
have non text content, which is usually images, but could also be embedded speech, or audio, or you know video that is playing in some place, provide text alternatives.

The reason for this is there are people, like I said seeing it on a speech oriented browser, where you may not want the video to be playing somewhere in the corner, you want the speech browser to decide what to play and when to play. Or it might even be reading out the different components. And if you have an image or a video, you should have a caption associated with it. So, captions and alternatives for multimedia are essential. And also try and create content that can be presented in different ways.

What does that mean, you could think that maybe I have an image or a histogram or something out here, but if you could also have something which is a brief description of the image, saying that, this, when you plot the text data out here, you find that it falls under a bell curve with this kind of mean, and this variance, that is something that even without looking at the picture a person can understand.

And finally, make it easier for users to both see and hear content. What does that mean, it basically means that there could be places where, I mean, one of the worst offenders over here is when you have a large banner, where the text on it is actually part of an image. Now, maybe you did that, because you wanted some kind of a fancy font over there, the problem is a person who is visually impaired is getting nothing from it, they do not have any way of reading, or even knowing what is on the page. So, these are the perceivable components of accessibility.

(Refer Slide Time: 08:40)



Next comes operable, the operation, the interaction with the user. One of these is may call functionality available from a keyboard. Now, why, because it turns out that a keyboard, in some ways is the simplest and easiest interface that we still have. It is sufficiently complex, because it has enough keys that you can use it to interact with computers in a extensive way. Enough people are familiar with it as things stand right now. And at the same time just the fact that you have that keyboard, and the ability to interact with the keyboard means that there is a lot of functionality that can be done.

There are unfortunately websites where you have to hover over something with a mouse, and only then something will drop down. And then you have to go and select it. And before it disappears, you have to select something else. That is impossible to use with the keyboard. And very, very difficult for a person who has some kind of motor control problems to actually get that mouse positioning exactly right. There are pages that sort of skip through data, they are like very fast changing they just flash something up on the screen and go away before you know what is happening.

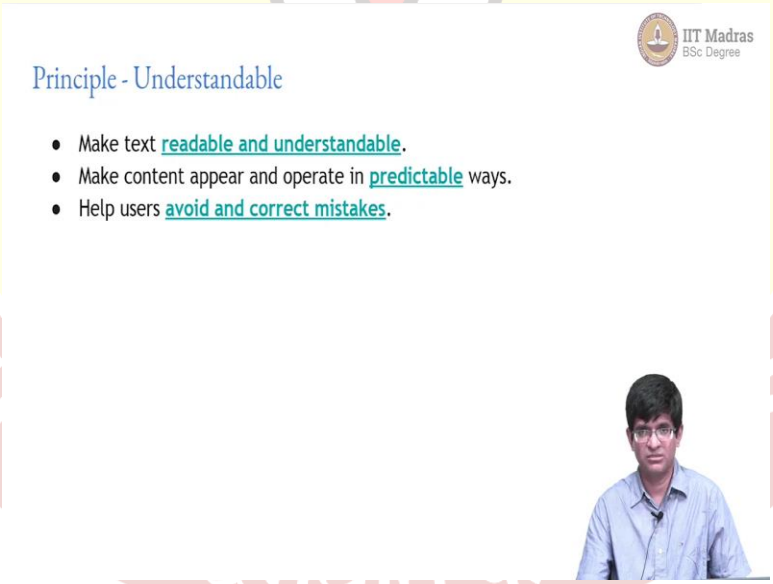
Do not do that, give users enough time, if there is a message that they need to read, put it up there and put a cross on it so that they can get rid of when they are done. There is one important class of images. And, I mean people have a problem with this, which is essentially called photo sensitivity, people have photo sensitivity, they react to strong flashing lights. And they can

actually cause seizures, epileptic seizures, sort of epileptic seizures in people, or other kinds of physical reactions, where a person just essentially blanks out, they are not able to respond.

And unfortunately, there are websites where that can actually the just the flashy lights and things that you have on them can triggers this kind of a reaction. Clearly, you should not be doing something of that sort especially not sort of knowingly and for a site that is likely to see users from multiple different backgrounds. At the same time, like I said earlier, the operability, you need to make it easy to navigate and find content, put appropriate tags to make it easy for a person to use.

And finally, the exact opposite of the first point over here, make all functionality available from a keyboard was the first, but you should also make it possible to use inputs other than a keyboard. Let us say that a person is not able to type on a keyboard, can they do everything using a mouse, or a joystick, or some kind of a button based interface? And all of those are important for accessibility.

(Refer Slide Time: 11:22)

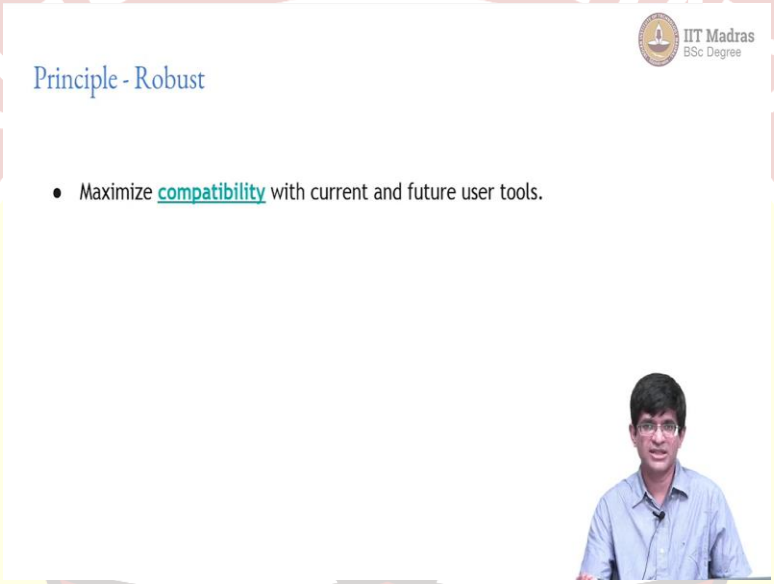


The understandability, make text readable and understandable. Is it a good font, a clear font that is readable, is it understandable, meaning that is it good language may not be English, it could be some other local language, anything else. But it should be easy to understand for a person who is looking for that language. Content should appear and operate in predictable ways. Meaning that

when you click on something, you expect a certain reaction, do not make something else happen instead. And also make it easy for a person to avoid mistakes in what they are doing.

Especially one possibility is you have a text box that expects a number as input, make it clear so that they can just enter only numbers, those kind of simple things can help, prevent mistakes, and thereby make it easier for the user also to sort of interact with the system.

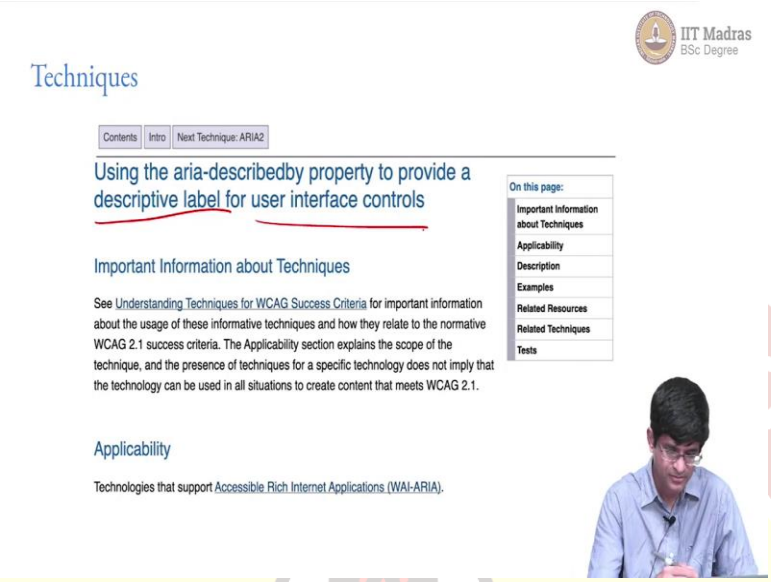
(Refer Slide Time: 12:21)



And finally, the entire system should be as robust as possible. Now, this is very generic. But the way to think about it is, if you follow standards and guidelines, there is a good chance that the majority of tools that exist, both today and in future are going to support what you have written. But if you try to use some new kind of tag or some new way of doing things, or something which is specific to a given browser or a given subset of operation within some particular language, there is a good chance that either it would not work on other browsers, or it would not work a few years down the line when that functionality is taken out.

So, keep all of those in mind in order to design something that is robust. So, these four classes of principles are essentially what are given as the WCAG, the Web Content Accessibility Guidelines from the W3 Consortium. And while, I am not expecting everyone to sort of learn them by heart, it is important that you are aware of these guidelines and that you know how to sort of look for applications of these and whether something is violating some of the guidelines or not, and suggest possible improvements on those.

(Refer Slide Time: 13:39)



The W3C also indicates a number of techniques. I am not going into all of them, I am just giving an example, a snapshot from one of their pages, which basically says that, for example, how do you provide a descriptive label for user interface controls. And what do we mean by this, it means that there is some way by which I can provide extra data within my HTML page, which can be used by other kinds of tools, which may not be normally there in a web browser.

But for a non standard kind of browser used by someone else with special needs, will be able to pull out the information I need and show it to them as they need to. So, in this way, the W3C website actually gives a number of techniques to make sites more accessible. And in fact, it also has tools that allows you to evaluate the accessibility of various websites and gives you a sort of a rating on how good a website is in terms of accessibility.

(Refer Slide Time: 14:37)

Finally, I have been talking about accessibility all this while one note on aesthetics. And this essentially refers to visual appearance, just how does an application look. And this chart over here essentially shows the evolution of the icons on the Apple iPhone over time. And if you look at it, there are a number of changes, not too major, but many of them have like relatively small sort of modifications.

What you can see over here is this phone icon has pretty much remained the same in structure, except that, for example, the background had a gradient. The gradient after a while was replaced by something, which is kind of a doc kind of a structure. That again, finally went away completely and just became a plain background. As far as mail was concerned, it had lots of clouds and things on the background of the icon, after some time, those went away to be replaced by relatively simple mail icon.

The browser had a complicated compass like structure, simplified. The iPod was used in order to indicate music. Eventually, at some point, they realized that they were no longer selling iPods, it was just music in the iPhone. The interesting thing you will notice is that you had the sort of bubbles and gradients and various other things, which looked nice, because the first time that they came across, but the more you use them, you find that they are not really, I mean they sort of distract from the visual appearance of a site after some time.

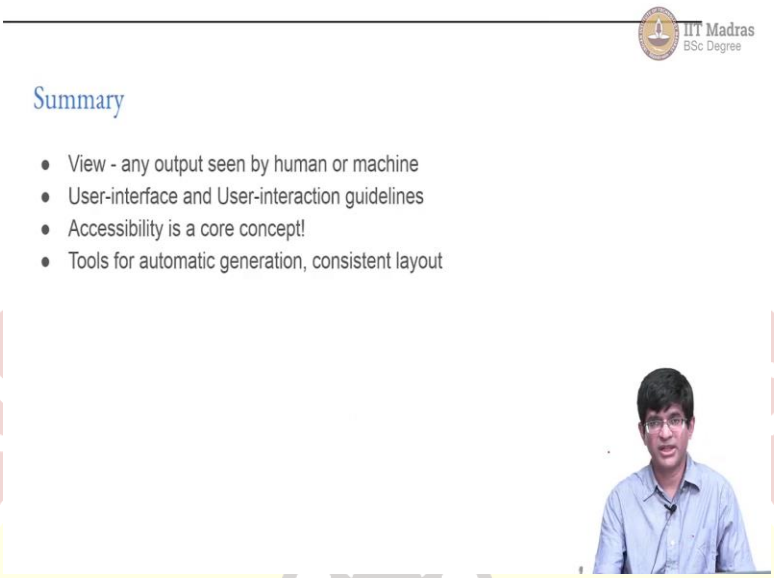
And the latest trend, material design from Google, and also the Apple icons on iOS, they are simple. But at the same time, highly detailed, if you look at the sort of resolution on these icons, they are essentially the vector graphics. So, they are going to be like very, very clear, no matter how much you zoom into them. And very neat and simple, the fonts are also easy to read. All of this contributes to the visual appearance of how you are trying to put this user interface across, it is extremely important.

For example, if they had chosen some completely outlandish colors, and put them out there, and you could have done that, there is a good chance that people would not have bought iPhones simply because of that, because they did not like the way it looked. A large part of why people go for Apple devices is looks, although of course, I mean that is oversimplifying, definitely they have very solid quality, build quality, the software, everything else, but the looks are a very important part, it is that attention to detail that matters.

And as you can see, over here, over time, the simplicity of the design has started becoming more and more dominant, you have less of these gradients, less of the curves. And everything just becomes a neat set of icons at the end. The catch, as you can see is it varies with time. So, what was considered visually appealing in 2007 or 2009, is not what we see as ideal in 2021. Part of the reason is better resolutions on screens, people are simply got used to certain kinds of fonts and color schemes and so on.

But this is at the bottom line a highly subjective decision. So, like I said, this is not something that can be taught easily, definitely not in a course like this. So, while not going into detail, I did want to touch upon it in order to show that that idea of simplicity and keeping things neat is something that you should try and have while developing your apps. Learn from experience, look at other people's learnings and try and absorb the best that you can from those.

(Refer Slide Time: 18:33)



To summarize, so this is pretty much the end of the discussion. What we have seen is that a view can be any output from a program that is seen by either human or another machine. There are extensive user interface and user interaction guidelines that have been built up purely through experience over the years.

And it is a good idea to sort of look for those and try and apply them in how you construct your views for your application. Accessibility is a core concept. Aesthetics is desirable, but accessibility is essential, because even if you do not have the greatest looking website, making it accessible means that you have actually paid attention to not just a small subset of your users or even a large subset of your users, but to pretty much anyone who might be interested in using your application.

And the extensive use of tools for automatic generation, consistent layout, those are the only ways by which you can actually generate views for large applications. And as we be seeing, by means of the assignments and various other parts of this course, there are these tools that we will be using in order to actually use templates and various other structures in order to generate views, generate outputs, and proper use of those tools is essential to getting good designs.