Introduction to Modern Application Development

Prof. Aamod Sane
FLAME University and Persistent Computing Institute
Abhijat Vichare
Persistent Computing Institute
Madhavan Mukund
Chennai Mathematical Institute

Lecture-01
Introduction To Modern Application Development Part 1

Hello, welcome to modern application development. I am Aamod Sane, your instructor for this session. In this course, we will learn how to build the kind of application that our users expect in the 2020s.

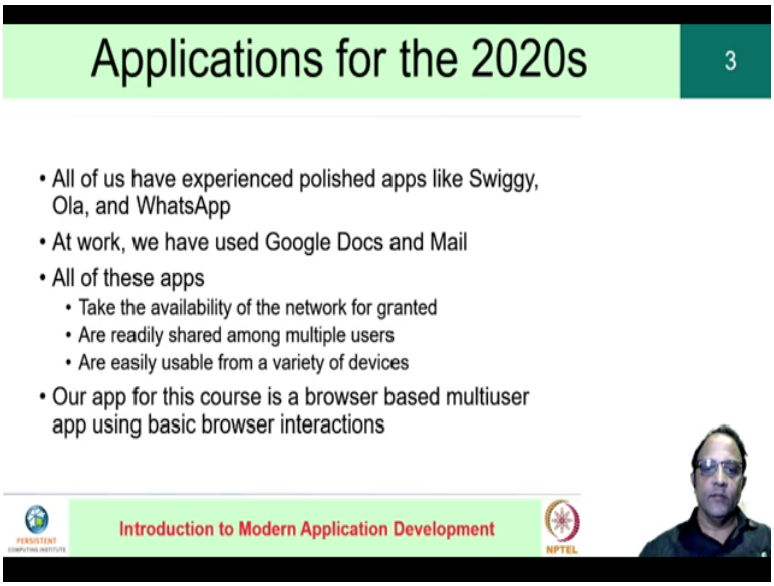
(Refer Slide Time: 00:26)

Let us see our plan for this session. To write an application, we must have an app idea, and we must also understand the expectations of our users. So, we will begin our discussion with some understanding of what it is that our users want. After that, we will look at differences between the simple programming assignments that we do versus what it is like to build modern apps.

Next, we will look at an overview of the structure of this course which will tell us how to go from a simple command line program to a full-fledged modern app. But before we can do any

programming, we must have interesting ideas for an app. So, to understand this, we will go over the origin of app ideas and present the app idea that we will develop in this course.

(Refer Slide Time: 01:22)



Okay, let us begin.

What is application development like in the 2020s?

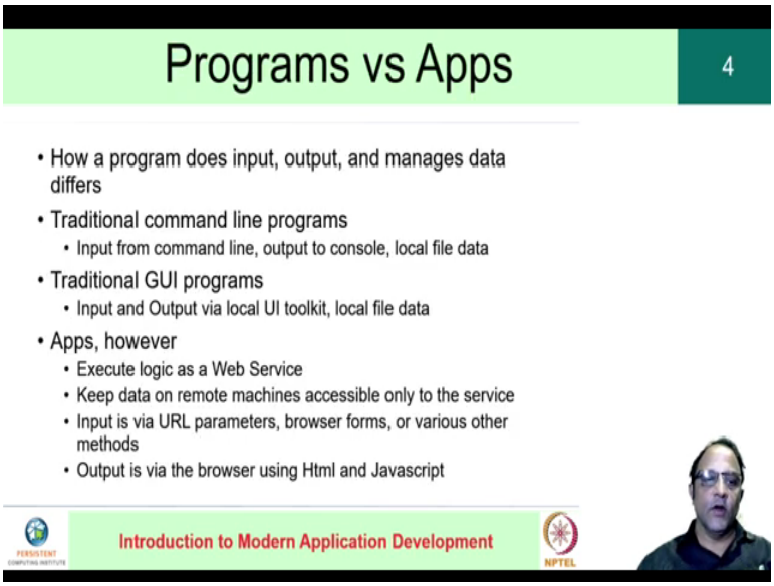
Much like an actor needs to understand their audience, or a writer needs to understand their readers, a programmer needs to understand the expectations of their users. In the 2020s, all of us have experience with polished applications like Swiggy, Ola, and WhatsApp. In the office or for other work, we have also used different apps like Google Docs and Gmail.

In such applications, the use of the internet is taken for granted to share data with the app as well as to share data with multiple users. All in all, in 2020s, an app should naturally incorporate the internet. It should be easily shared amongst multiple remote users, and should be usable from a wide variety of devices without needing to install or upgrade anything. In this course, we will restrict ourselves to building a browser-based app.

A browser-based app can easily adapt itself to mobile platforms, but mobile apps must be written for particular vendor. Since this is our first time building a modern app, we will use the most

general platform where the details of the platform are well understood. In some courses later, we will study how to build mobile applications.

(Refer Slide Time: 03:05)



Now that we have some definition of what we mean by a modern app, let us look at in detail the distinction between an ordinary program versus a modern app.

We write programs in order to have some effect on the world around us. So, the ability to respond to input data and to produce output data is fundamental to any program. The other thing programs offer is the ability to store old data and act on it at a later date. So, how a program does input-output and use of data is a big part of the distinction between a program and a modern app. In the simple command line programs, we write in our courses, following major steps are usually found:

- Take input from the command line
- App executes its logic locally
- Write output to a console
- Store data locally in files.

If we have written GUI apps, then we have managed user interaction using the local operating system provided graphics tool kit, or some sort of program which helps us to create the GUI. For example, once upon a time, Visual Basic was a very commonly used tool for the development of

GUI apps. Nowadays, we might use other things. In these apps as well, although the interaction is much nicer than a simple command line, and the output is much richer than the console, it is still the case that the app logic executes locally, data is stored locally, and the use of the application is, by and large, limited to the single user who is using this data.

However, the simplest of modern apps is different in all four of these things and has following main characteristics:-

- The logic of the app is executed remotely as a web service on a server
- This web service is then offered to users via a web server.
- App interacts with this web server via their browsers.
- The apps data is stored on the server, or another remote machine accessible to the server.
- Interaction with the app occurs on the browser, which itself runs on the user's device.

Besides these differences, where some of the processing, or we should say, most of the interesting processing is happening remotely, there are also differences in how input and output occur. Usually the input is in the form of parameters, or via the browser forms. And the output is usually HTTP request which has the HTML, CSS and JavaScript code that the browser can understand.

The interaction on the application is done in a way that the users find natural, in comparison to command lines or consoles, and even to a large extent in comparison with user interface programs. The interface offered by web programs are far similar to each other, making it easier for the user to use and navigate through the application, compared to some of the odd and novel interactions that graphic user programs used to offer.

(Refer Slide Time: 05:59)

In this course, we will assume that you know how to write program logic and get basic command lines programs working. We do not assume that you know how to write GUI programs even on a desktop, and we are going to move straight from writing a command line program to a web-based program. In the course, we will focus on following details :-

- How to organize the program as a web service.
- Interact with users over the browser.
- Deal with multiple users.
- Sharing amongst multiple users
- Remote data storage

And many of the other difficulties that are inherent to any program that runs over the internet. So, that is the overall goal of our course. In the course, we will take you step by step like we just talked about. We will start with a desktop program that implements the core parts of our app idea. This program will take input from the command line and print results to the console. This is how we will begin. We will then learn the basic ideas that govern the creation of web-based applications. Once we understand those, step by step, we will change our desktop program into a

command line program. We will change the program to first run it as a web service and store its data on the server.