**Slide 2**

**Agenda**

Hello friends, we are going to see these topics today.

**Slide 3**

**Definition**

A **web application** is a program that runs on a computer with a web server, while its users interact with it via a web browser or similar user agent.

We can develop these applications using languages such as javascript and HTML. Web application uses web browser to render program executable. Some of these applications are dynamic means it will be processed on server and some are static means no processing.

Examples of web applications:

- Google Docs

- Facebook

- Twitter

**Slide 4**

**Basic Architecture**

This is basic architecture of Web Application.

Web applications have server machine and client machine.

Server machine may contain http server, database, Servlet engine, files, etc.

Client machine must have a web browser to access web server.

These two machines must be connected with internet.

**Slide 5**

**Desktop Application**

An application that runs stand-alone in a desktop or laptop computer.

These applications are software which are present in hard drive and use operating system and other resources to execute.

Examples are Microsoft words, Notepad, etc

These applications are platform dependent means they are dependent on operating system and may not work in other operating system.

It runs on a PC operating system (Windows, Mac, Linux, etc.)

It has a graphical user interface

It does not run inside a web browser.

Examples are Microsoft word, web browser.

**Slide 6**

**Drawback’s of Desktop Application**

Desktop applications can be portable, but most are \*not\* portable and require manual installation from the user.

Desktop applications usually need to be manually updated (or at least have manual approval) to install updates.

You cannot carry Laptop/PC everywhere to access Desktop Applications.

**Slide 7**

**Web Application and Desktop Application**

These are the differences between Web Application and Desktop Application

Maintenance

Web Application is installed on Web server only so when we update this application we have to update on server machine only, Desktop Application is installed on every machine so when we update the application, we have to update on each machine.

Usage Scale

Web Applications are available from any location using internet but Desktop applications have location constraints.

**Slide 8**

Speed and performance

Web application performance is depend upon broadband performance. Desktop application performance is dependent on standalone machine.

Bandwidth cost

Web application broadband cost is more than desktop application.

**Slide 9**

Benefits of Developing Web Applications with PHP

**Built-in Web Development Features:** you can use built-in classes and functions included in PHP to accomplish common web development tasks and operations.

**Platform Independent:** At present, PHP supports major operating systems including Windows, Linux, Mac OS X, and UNIX.

**Work with Popular Databases:** PHP supports some of the most widely used databases including MySQL, ODBC, SQLite, PostgreSQL, Interbase, Frontbase, dBase and IBM Db 2.

**Availability of Many Frameworks and Libraries:** PHP frameworks are Laravel, CakePHP, Yii, CodeIgniter, Phalcon and Symfony. PHP libraries like Hoa, Mink, Faker, Geocoder, Requests and Ratchet.

**Open Source:** PHP is open source

**Supported by a Large Community:** PHP is developed and updated by a large and active community of programmers.

**Slide 10**

These are the best practices of web security

Office Security : Every software company have security access and lock to get acess in project area.

PC Security : Many users uses Antivirus, USB diasable, Login, Blocking websites for securing PC.

Application Security : SSL uses encription, PHP provide security for website.

Server security : Many operating system provides security for server using authontication.

Network security : Firewalls, Routers, Proxies provides network sucurity by using configuration.

**Slide 11**

## These are basic rules we have to follow for web security

## Input Data Validation

Always validate data in your PHP code. Javascript validation will not work, if javascript engine is off.

## Guarding Against XSS Attacks

Cross-site scripting attack (XSS attack) is an attack based on code injection into vulnerable web pages. The danger is a result of accepting unchecked input data and showing it in the browser.

## Guarding Against CSRF Attacks

In a Cross Site Request Forgery (CSRF) attack, the attacker tricks the victim into loading sensitive information or making a transaction without their knowledge. This mainly occurs in web applications that are badly coded to trigger business logic using GET requests.

## Preventing SQL Injection Attacks

To perform your database queries, you should be using PDO. With parameterized queries and prepared statements, you can prevent SQL injection.

## Protecting the File System

As a developer you should always write your code in such a way that none of your operations put your file system at risk.

## Protecting Session Data

A good way to guard your session data is to encrypt the information stored in the session.

## Proper Error Handling

It’s good to know about all the errors.

**Slide 12**

### Cost effective development

Web application will be developed only for one machine.

### Accessible anywhere

Web Application will be available everywhere through internet.

### Easily customizable

The user interface of web-based applications is easier to customise than is the case with desktop applications

### Accessible for a range of devices

It can be access through PC, mobile phones and tablets.

### Easier installation and maintenance

With the web-based approach installation and maintenance becomes less complicated too.

### Adaptable to increased workload

Increasing processor capacity also becomes a far simpler operation with web-based applications.

### Increased Security

Security is tighter and any potential breaches should be noticed far more quickly.