**MINI-PROJECT REPORT**

**ON**

Online Bus Ticket Reservation System

****

REPORT SUBMITTED

TO

VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY, PUNE

FOR THE Mini-Project OF JAVA

IN

**INFORMATION TECHNOLOGY DEPARTMENT**

BY

**Name of the student –ANKUSH PATIL-21810232/232043**

**Name of the student –ABHISHEK SALAPE-21810258/232055**

**Name of the student –TEJAS SARNAIK-21810308/232056**

**Name of the student –MANASI PURKAR-21810414/232050**

**Class:SY-IT Division: B Batch:B2**

**Batch Teacher**

**Mrs.Suvarna Pawar**

## Mini Project 2) Client Side Technologies

## User interface using HTML and client side scripting JavaScript

## Online Bus Ticket Reservation System

## Abstract

Web applications are one of the most useful platforms for the delivery of the information and service over the internet today. Day-to-Day the popularity of web application usage is increasing. Hence the web apps has to be designed for secure transformation of information from web client to web server and to mitigate the vulnerabilities. This paper presents a research survey report on constructing a secure web applications at the client side, server side and even in browsers. This paper initially gives an overview of web application, then followed by techniques or tools along with the attacks at the client side and serverside. It gives a big picture with the aim of promoting future research in this area.

## Theory

* **HOW DOES CLIENT-SIDE SCRIPTING WORK?**

There is overlap between the two technologies as they work in tandem, but there are core differences. [Server-side scripting](https://www.upwork.com/hiring/development/server-side-scripting-back-end-web-development-technology/) works in [the back end of a site](https://www.upwork.com/hiring/development/back-end-web-developer/), which the user doesn’t see. It creates a scaffolding for the site to access its database, all the behind-the-scenes mechanics that organize and power a website. Client-side code, however, handles what the user *does* see.

* Scripts are embedded within and interact with [the HTML of your site](https://www.upwork.com/hiring/development/the-basics-of-web-development/), selecting elements of it, then manipulating those elements to provide an interactive experience.
* Scripts interact with a [cascading style sheet (CSS)](https://www.upwork.com/hiring/development/css-cascading-style-sheets/) file that styles the way the page looks.
* It dictates what work the server-side code is going to have to accomplish (where utility should be built around these front-end functions), and returns data that’s pulled from the site in a way that’s readable by the browser. For example: If there’s a form for updating a profile, the back end is built to pull specific data from the database to populate that form, while front-end scripts populate the form with that information.
* Scripts put less stress on the serverbecause they don’t require processing on the server once they’re downloaded, just when post-backs are made. “Post-backs” perform specific call-and-answers with the server-side code, and respond to the user immediately.
* HTML

HTML is the standard markup language used to create web pages and web applications. Its elements form the building blocks of pages, representing formatted text, images, form inputs, and other structures. When a browser makes a request to a URL, whether fetching a page or an application, the first thing that is returned is an HTML document. This HTML document may reference or include additional information about its look and layout in the form of CSS, or behavior in the form of JavaScript.

* CSS

CSS (Cascading Style Sheets) is used to control the look and layout of HTML elements. CSS styles can be applied directly to an HTML element, defined separately on the same page, or defined in a separate file and referenced by the page. Styles cascade based on how they are used to select a given HTML element. For instance, a style might apply to an entire document, but would be overridden by a style that applied to a particular element. Likewise, an element-specific style would be overridden by a style that applied to a CSS class that was applied to the element, which in turn would be overridden by a style targeting a specific instance of that element (via its ID). Figure 6-1

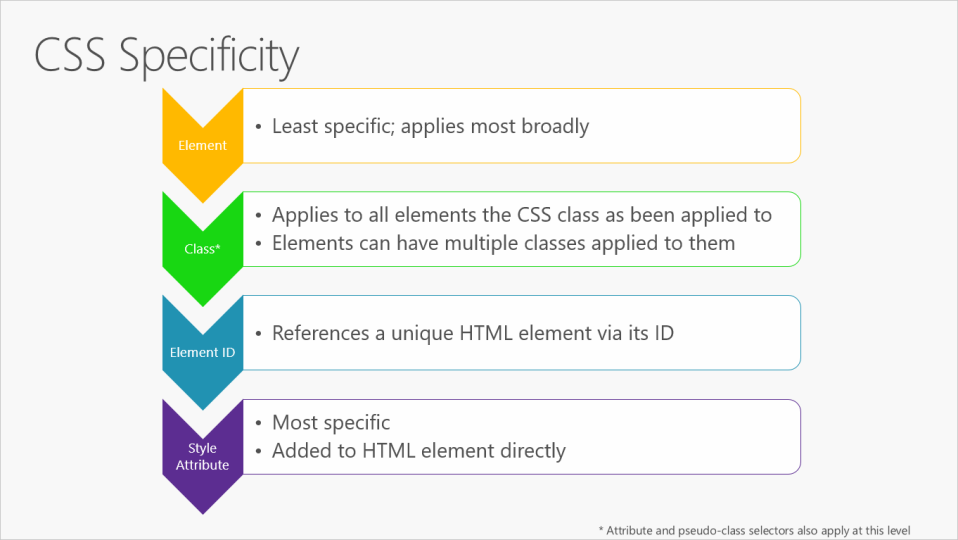


Figure 6-1**.** CSS Specificity rules, in order.

It's best to keep styles in their own separate stylesheet files, and to use selection-based cascading to implement consistent and reusable styles within the application. Placing style rules within HTML should be avoided, and applying styles to specific individual elements (rather than whole classes of elements, or elements that have had a particular CSS class applied to them) should be the exception, not the rule.

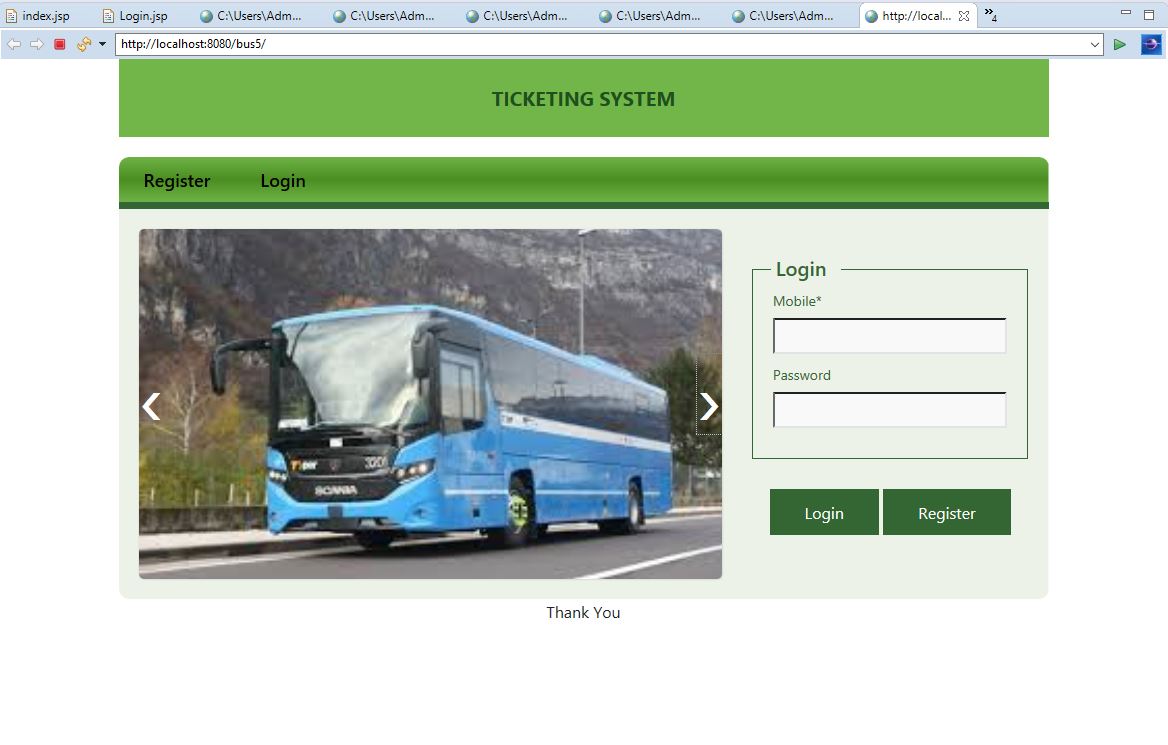
* JavaScript

JavaScript is a dynamic, interpreted programming language that has been standardized in the ECMAScript language specification. It is the programming language of the web. Like CSS, JavaScript can be defined as attributes within HTML elements, as blocks of script within a page, or in separate files. Just like CSS, it's recommended to organize JavaScript into separate files, keeping it separated as much as possible from the HTML found on individual web pages or application views.

* When working with JavaScript in your web application, there are a few tasks that you'll commonly need to perform:
* Selecting an HTML element and retrieving and/or updating its value.
* Querying a Web API for data.
* Sending a command to a Web API (and responding to a callback with its result).
* Performing validation.

You can perform all of these tasks with JavaScript alone, but many libraries exist to make these tasks easier. One of the first and most successful of these libraries is jQuery, which continues to be a popular choice for simplifying these tasks on web pages. For Single Page Applications (SPAs), jQuery doesn't provide many of the desired features that Angular and React offer.

#### SYSTEM OUTPUT :

****

## **References:**

* Wikipedia, “Scripting language”
* http://www.w3schools.com/js/js\_objects.asp

## **CONLUSION :**

In these mini-project we build and learn a client script web page using several languages like Html,CSS,Javascript