Codeline

MINI PROJECT – I

SYNOPSIS

Department of Computer Science & Application

Institute of Engineering & Technology

SUBMITTED TO:

Ms. Madhu

(Technical Trainer)

SUBMITTED BY: -

Aastha Singh (201500005)

Pranjal Gupta (201500498)

Sarvagya Bansal (201500622)

Acknowledgement

It gives us a great sense of pleasure to present the synopsis of the B.Tech mini project undertaken during B-Tech III Year. This project is going to be an acknowledgement to the inspiration, drive and technical assistance will be contributed to it by many individuals. We owe special debt of gratitude to Ms. Madhu, Technical Trainer, for providing us with an encouraging platform to develop this project, whichthus helped us in shaping our abilities towards a constructive goal and for his constant support and guidance to our work.

His sincerity, thoroughness and perseverance has been a constant source of inspiration for us. We believe that he will shower us with all his extensively experienced ideas and insightful comments at different stages of the project & also taught us about the latest industry-oriented technologies. We also do not like miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and co-operation.

Aastha Singh (201500005)
Pranjal Gupta (201500498)
Sarvagya Bansal (201500622)

Contents

INTRODUCTION

In the mini project our main objective is to create a system in which that we have the compilers of HTML, CSS and JavaScript alongside on a single setup and the output is obtained at once without the need of creating separate repositories for accessing them, thus saving time and effort at the same time. Front End languages (Html, CSS, JavaScript) communicate requests to Back End languages. Every website has a server, database, and other applications that interact with the Front End through code created by a Back End development. HTML, CSS, and JavaScript are the topmost languages used for Front End development. The structure, design, behavior, and content of everything seen on browser screens when websites, web applications, or mobile apps are opened up, is implemented by Front End developers. Hence, implementation of these front end languages in our project will provide the best, quick and accurate results which will be beneficial to the user at its utmost.

SOFTWARE AND HARDWARE REQUIREMENTS

- JSDK1.5
- React-JS
- Oracle 8i
- Ethernet Adapter
- 512 MB Ram
- Window 10 and Above

LANGUAGES USED

- React-JS
- Html
- CSS
- JavaScript

PROJECT- DESCRIPTION

A compiler is a computer program that helps in translating the computer code from one programming language into another language. Basically, it translates the program written in the source language to the machine language.

The compiling process contains an essential translation operation and error detection.

Majorly, there are three types of compilers:

- Single Pass Compilers
- Two Pass Compilers
- Multi-pass Compilers

Single Pass Compiler:

When we merge all the phases of compiler design in a single module, then it is called a single pass compiler. In the case of a single pass compiler, the source code converts into machine code.

Two Pass Compiler:

A processor that runs through the program to be translated twice is considered a two-pass compiler.

Multipass Compiler:

A program's source code or syntax tree is processed many times by the multipass compiler. It breaks down a huge programme into numerous smaller programmes and runs them all at the same time. It creates a number of intermediate codes. All of these multipasses use the previous phase's output as an input. As a result, it necessitates less memory. 'Wide Compiler' is another name for it.

To conclude, compilation is the process of translating the code we write, into code that a computer understands. It does this by scanning and parsing your source input, and mapping this input to the target output which your computer can understand.

Further, a compiler can be built and implemented in any language. Keeping that in mind, we have used REACT JS to implement the compilers in our project. In the mini project we have created, a system in which that we have the compilers of JS, HTML and CSS alongside on a single setup and the output is obtained at once without the need of creating separate repositories for accessing them, thus saving time and effort at the same time.

HyperText Markup Language (HTML): The most extensively used Frontend Language is HTML, which is a markup language. HyperText Markup Language is the abbreviation for HyperText Markup Language. It's a programming language that is used to make websites and web apps.

Cascading Style Sheets: It is a design language that simplifies making web pages presentable. Selectors are used to pick elements and apply styles to them.

Java Script's: Its primary purpose is to enhance a user's interaction with a web page. In other words, you can utilize this programming language to improve the vibrancy and engagement of your website. JavaScript is also widely used in the development of games and mobile apps. Excellent speed, cross-browser interoperability, and straightforward semantics are just a few of JavaScript's essential qualities, providing a seamless developer experience.

Front End languages communicate requests to Back End languages. Every website has a server, database, and other applications that interact with the Front End through code created by a Back End devlopment.

HTML, CSS, and JavaScript are the topmost languages used for Front End development. The structure, design, behavior, and content of everything seen on browser screens when websites, web applications, or mobile apps are opened up, is implemented by Front End developers.

Hence, implementation of these front end languages in our project will provide the best , quick and accurate results which will be beneficial to the user at its utmost.

IMPLEMENTATION

A Compiler can be built and implemented in any language. Keeping that in mind, We have used REACT JS to implement the compilers in our project. **React** (also known as **React.js** or **React-JS**) is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies. React can be used as a base in the development of single-page, mobile, or server-rendered applications with frameworks like Next.js. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

REFERENCES

❖ Books:

- HTML
  - HTML – by Jon Duckett
  - Head First HTML and CSS by Elizabeth Robson and Eric Freeman
- CSS
  - CSS – by Eric A. Meyer
  - CSS –by Jon Duckett
- JAVASCRIPT
  - You Don't Know JS by Kyle Simpson
  - A Smarter Way to Learn JavaScript by Mark Myers

❖ Websites:

- www.geeksforgeeks.org
- www.google.com
- www.reactjs.org
- www.projectdeveloper.com
- www.javatpoint.com
- www.youtube.com

Faculty Guidelines:

Ms. Madhu (Technical Trainer in GLA University)

GitHub Repositories link:

- https://github.com/AasthaSingh20213002/codeline
- https://github.com/pranjalgupt/Codeline
- https://github.com/Sarvagyaban/Codeline