

INTERNSHIP REGISTRATION

A Mini Project report submitted in the partial fulfillment of the requirements for

the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

K. RAVITEJA

19761A0528

Under the guidance of

Mr. T. UDAY KUMAR

SR.ASST.PROFESSOR



Department of Computer Science and Engineering
Lakireddy Bali Reddy College of Engineering (Autonomous)

Accredited by NAAC & NBA (Under Tier - I)
Affiliated to JNTUK, Kakinada; ISO 9001:2015 Certified
2020-2021



LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING

(AUTONOMOUS)

Accredited by NAAC & NBA (Under Tier - I) ISO 9001:2015 Certified Institution

Approved by AICTE, New Delhi and Affiliated to JNTUK, Kakinada

L.B. REDDY NAGAR, MYLAVARAM, KRISHNA DIST., A.P.-521 230.

<http://cse.lbrce.ac.in>, cselbreddy@gmail.com, Phone: 08659-222933, Fax: 08659-222931

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project entitled “INTERNSHIP REGISTRATION” is being submitted by **KARAVADI RAVITEJA** in partial fulfillment for the award of B.Tech in Computer Science & Engineering to the Jawaharlal Nehru Technological University Kakinada is a record of bonafide work carried out by her under our guidance. The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma.

Project guide

Mr. Uday Kumar
Sr.Assistant Professor

Head of the department

Dr.D.Veeraiah
Professor

External Examiner

ACKNOWLEDGEMENT

I would like to thank **Mr.T.Uday Kumar**, Senior Assistant Professor, CSE department for the encouraging and support in carrying out this project.I would like to thank Mini-Project In-charge **T.N.V.S.Praveen**, Sr.Assistant Professor, **CSE department** for the encouraging and support in carrying out this project.I also take the privilege to record my thanks to **Dr. D.Veeraiah**, Head of the Department of CSE whose encouragement, cooperation and valuable support crown our success.I express my thanks to the support given by management in completing my thesis. I also express my sincere gratitude & deep sense of respect to the Principal, **Dr.K.Appa Rao** for making us available all the required assistance and his support and inspiration to carry out this project in the Institute. I am thankful to the teaching and non-teaching staff of CSE department for their direct as well as indirect help in my project. I am elated to avail my selves to this opportunity to express my deep sense of gratitude to my parents.

INTERNSHIP REGISTRATION

ABSTRACT

The main purpose of this project is to efficiently use the modern tools like web-servers Express and Node JS for the website building. This project is about registering to a particular Internship program and get qualified dynamically at the run-time itself. The website contains a Homepage where there is a header section followed by a navigation division section along with the information about the Internships. The user needs to select the desired Internship program where the user will be redirected to the respected Internship page. In that page the user can have the details about the Internship along with a Apply now button at the bottom of the page. After clicking the Apply now button the user will be redirected to the submit form at which the user has to submit his details and then click on Submit button. From there the user will be redirected to the Respected Internship Quiz page, where the user has to participate in the Quiz. After completing the Quiz, Result analysis will be done dynamically at the server side and the user will get the result immediately. According to the result the servers automatically decide whether particular user has selected for the Internship or not.

CONTENTS

<u>CONCEPTS</u>	<u>Page No</u>
1. Introduction	5
2. Working	6
3. Languages	
3.1 HTML	7
3.2 CSS	7
3.3 JAVA SCRIPT	8
3.4 BOOT STRAP	8
3.5 NODE JS	
4. Implementation	
4.1 coding	09
4.2 ml.html	10-11
4.3 homepage.html	12-13
4.4 app.js	14-15
4.5 question.js	16-18
4.6 scripts.js	19-23
5. Result	24-27
6. Conclusion	28
7. References	28

LIST OF IMAGES

<u>Concept</u>	<u>Page no</u>
1. Working	6
2. Result	24-29

1. INTRODUCTION

At the starting Homepage you can have a header section followed by a navigation division section along with the details of the internship. You need to select the desired Internship program where you will be redirected to the respected Internship page. In that page you can have the details about the Internship along with a Apply now button at the bottom of the page. After clicking the Apply now button you will be redirected to the submit form at which you have to submit his details and then click on Submit button. From there you will be redirected to the Respected Internship Quiz page, where you have to participate in the Quiz. After completing the Quiz, Result analysis will be done dynamically at the server side and the user will get the result immediately. According to the result the servers automatically decide whether particular user has selected for the Internship or not.

REQUIREMENTS :

FRONT END :

1. HTML
2. CSS
3. JAVA SCRIPT
4. BOOT STRAP

BACK END :


5. NODE JS

2. WORKING

Working will be done at the Express server. Where the dynamic analysis of Quiz will takes place. The page redirections is done using Node JS.

Note: Quiz analysis will be done automatically at the server side.

HOMEPAGE :



The diagram illustrates the components of an internship. At the top, the word "INTERNSHIP" is written in large, bold, blue capital letters. Below it, a horizontal line connects seven icons, each with a label underneath. The icons and labels are: 1. A location pin with a checkmark inside a circle, labeled "GOAL". 2. Three red stars, labeled "SKILLS". 3. A brain with red lines radiating from it, labeled "KNOWLEDGE". 4. Two people silhouettes with a speech bubble, labeled "MENTORING". 5. A wrench and a screwdriver crossed, labeled "PRACTICE". 6. A person silhouette with a checkmark inside a circle, labeled "OPPORTUNITY". 7. A person silhouette with a red arrow pointing to them, labeled "TRAINING".

[HOME](#) [GALLERY](#) [INTERNSHIPS ▾](#) [ABOUT](#) [CONTACT US](#)

An internship is a period of work experience offered by an organization for a limited period of time.[1] Once confined to medical graduates, internship is used practice for a wide range of placements in businesses, non-profit organizations and government agencies. They are typically undertaken by students and graduates looking to gain relevant skills and experience in a particular field. Employers benefit from these placements because they often recruit employees from their best interns, who have known capabilities, thus saving time and money in the long run. Internships are usually arranged by third-party organizations that recruit interns on behalf of industry groups. Rules vary from country to country about when interns should be regarded as employees. The system can be open to exploitation by unscrupulous employers.[citation needed] Internships for professional careers are similar in some ways. Similar to internships, apprenticeships transition students from vocational school into the workforce. The lack of standardization and oversight leaves the term "internship" open to broad interpretation. Interns may be high school students, college and university students, or post-graduate adults. These positions may be paid or unpaid and are temporary.[2] Many large corporations, particularly investment banks, have "insights" programs that serve as a pre-internship event numbering a day to a week, either in person or virtually.

3. LANGUAGES

3.1 HTML

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content

3.2 CSS

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media.
- CSS saves a lot of work. It can control the layout of multiple web pages all at Once.
- External stylesheets are stored in CSS files.

3.3 JAVA SCRIPT

- JavaScript is the world's most popular programming language.
- JavaScript is the programming language of the Web.
- JavaScript is easy to learn.

3.4 BOOT STRAP

- Bootstrap is a free front-end framework for faster and easier web development.
- Bootstrap includes HTML and CSS based design templates for typography, forms, modals, image carousels and many other, as well as optional JavaScript plugins.
- Bootstrap also gives you the ability to easily create responsive designs.

3.5 NODE JS

- Node.js can generate dynamic page content.
- Node.js can create, open, read, write, delete, and close files on the server.
- Node.js can collect form data.
- Node.js can add, delete, modify data on your database.

4. IMPLEMENTATION

4.1 CODING

Files:

1. homepage.html
2. ml.html
3. app.js
4. questions.js
5. scripts.js

1. Homepage.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <!-- Bootstrap CSS -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
      crossorigin="anonymous"
    />
    <link
      rel="stylesheet"
      type="text/css"
      href="../MINI_PROJECT/homepage.css"
    />

    <title>HOMEPAGE</title>
  </head>

  <body>
    <script
      src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-
MrwW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
      crossorigin="anonymous"
    ></script>

    <div class="header border border-secondary">
      <center>
        
      </center>
    </div>
    <br />

    <div class="container-fluid border-bottom border-secondary">
      <div class="row">
        <a class="col-2" href="homepage.html">HOME</a>
        <a class="col-2" href="gallery">GALLERY</a>
```

```

<!-- <a class='links' href='internship.html'>INTERNSHIPS</a> -->
<div class="col-2 dropdown">
  <button
    class="btn btn-link dropdown-toggle"
    type="button"
    id="dropdownMenuButton1"
    data-bs-toggle="dropdown"
    aria-expanded="false"
  >
    INTERNSHIPS
  </button>
  <ul class="dropdown-menu" aria-labelledby="dropdownMenuButton1">
    <li><a class="dropdown-item" href="/c">C</a></li>
    <li><a class="dropdown-item" href="/ds">DS</a></li>
    <li><a class="dropdown-item" href="/java">JAVA</a></li>
    <li><a class="dropdown-item" href="/ml">ML</a></li>

    <li><a class="dropdown-item" href="/python">PYTHON</a></li>

    <li><a class="dropdown-item" href="/wt">WT</a></li>
  </ul>
</div>
<a class="col-2" href="about.html">ABOUT</a>
<a class="col-2" href="contact.html">CONTACT US</a>
</div>
</div>

<div class="para">
  <br />
  <p id="paras">
    An internship is a period of work experience offered by an organization
    for a limited period of time.[1] Once confined to medical graduates,
    internship is used practice for a wide range of placements in
    businesses, non-profit organizations and government agencies. They are
    typically undertaken by students and graduates looking to gain relevant
    skills and experience in a particular field. Employers benefit from
    these placements because they often recruit employees from their best
    interns, who have known capabilities, thus saving time and money in the
    long run. Internships are usually arranged by third-party organizations
    that recruit interns on behalf of industry to the workforce. The lack of
    standardization and oversight leaves the term "internship" open to broad
    interpretation. Interns may be high school students, college and
    university students, or post-graduate adults. These positions may be
    paid or unpaid and are temporary.[2] Many large corporations,
    particularly investment banks, have "insights" programs that serve as a
    pre-internship event numbering a day to a week, either in person or
    virtually.

  </p>
</div>
</body>
</html>

```

2. ML.html

```
<!doctype html>
<html lang="en">
  <head>

    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWfSpd3yD65VohhpucOmLASjC"
crossorigin="anonymous">

    <title>MACHINE LEARNING</title>
  </head>
  <body>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
crossorigin="anonymous"></script>
    <script>
      function goToForm() {
        window.location.href = "/form?quiz=ml"
      }
    </script>
    <div>
      <center><img src='https://www.freecodecamp.org/news/content/images/2022/01/machine-learning-banner-2.png'
style="height: 450px; width: 1000px;" > </center>

    </div>

    <h2>
      <u> <b> <font color=red" face="Arial Black" >GOALS</font></b></u>
    </h2>

    <p align='justify'>
      <ul style='list-style-type:disc'>
        <li> Get hands-on programming exposure in MACHINE LEARNING the way Industry works</li>
        <li>Build a good project based on MACHINE LEARNING by exploring all C programming constructs</li>
        <li>Get exposure in Standard Industry Practices of Software Development</li>
      </ul>
    </p>

    <h2> <u> <b> <font color=red" >TIME DURATION</font> </b></u></h2>

    <p align='justify'>
      <ul style='list-style-type: disc;'>
        <li>6-weeks</li>
        <li>1.5hr live virtual classes</li>
        <li>5 days a week</li>
        <li>Unlimited intervention/ Doubt clearing session</li>
```

```

    </ul>
</p>

<h2> <u> <b> <font color=red" >MATERIALS</font> </b></u></h2>

<p align='justify'>
  <ul style='list-style-type: disc;'>
    <li>Technical Resource Library: Plenty of quality Embedded Systems free course materials available on-line</li>
    <li>Emertxe official blog: Knowledge sharing and industry trends update to students</li>
    <li>Emertxe's YouTube Channel: Tutorial and How-to video</li>
  </ul>
</p>
<br>
<center> <input onclick="goToForm()" type='button' value='APPLY NOW' style="display: flexbox; color: red; font-size: 24px;hover: green;"></center>
<br>
<br>
</body>
</html>

```

3. App.js

```
const express = require('express');
const path = require('path');
const { sendMail } = require('./sendEmail');
const app = express();

app.use(express.json());

app.use('/home', (req, res) => {
  res.sendFile(path.join(__dirname, 'homepage.html'));
});

app.use('/c', (req, res) => {
  console.log('path :- ', path.join(__dirname, 'interns', 'c.html'));
  res.sendFile(path.join(__dirname, 'interns', 'c.html'));
});

app.use('/ds', (req, res) => {
  res.sendFile(path.join(__dirname, 'interns', 'ds.html'));
});

app.use('/dts', (req, res) => {
  res.sendFile(path.join(__dirname, 'interns', 'dts.html'));
});

app.use('/java', (req, res) => {
  res.sendFile(path.join(__dirname, 'interns', 'java.html'));
});

app.use('/ml', (req, res) => {
  res.sendFile(path.join(__dirname, 'interns', 'ml.html'));
});

app.use('/python', (req, res) => {
  res.sendFile(path.join(__dirname, 'interns', 'python.html'));
});


app.use('/wt', (req, res) => {
  res.sendFile(path.join(__dirname, 'interns', 'wt.html'));
});

app.use('/form', (req, res) => {
  res.sendFile(path.join(__dirname, 'interns', 'form.html'));
});

app.post('/api/saveForm', (req, res) => {
  global.formData = req.body;
  return res.json('saved');
});

app.get('/send-mail', (req, res) => {
```



```
console.log('form data :- ', global.formData);
sendMail(global.formData.email, () => {
  console.log('Email Successfully sent ');
});
res.json({});
});

app.use('/quiz-c', express.static(path.join(__dirname, 'QUIZ', 'C')));

app.use('/quiz-ds', express.static(path.join(__dirname, 'QUIZ', 'C')));

app.use(
  '/quiz-dst',
  express.static(path.join(__dirname, 'QUIZ', 'DATA_STRUCTURES')),
);

app.use('/quiz-java', express.static(path.join(__dirname, 'QUIZ', 'JAVA')));

app.use(
  '/quiz-ml',
  express.static(path.join(__dirname, 'QUIZ', 'MACHINE_LEARNING')),
);

app.use('/quiz-python', express.static(path.join(__dirname, 'QUIZ', 'PYTHON')));

app.use(
  '/quiz-web',
  express.static(path.join(__dirname, 'QUIZ', 'WEB_TECHNOLOGIES')),
);

app.listen(3000, () => {
  console.log('server is running at 3000');
});
```

4. Questions.js

```
// creating an array and passing the number, questions, options, and answers
let questions = [
  {
    numb: 1,
    question: "What are the different types of attributes?",
    answer: "All of the above",
    options: [
      "Nominal",
      "Ordinal",
      "Spacial",
      "All of the above"
    ]
  },
  {
    numb: 2,
    question: "The types of a Nominal attribute depends on which of the following properties",
    answer: "Distinctness",
    options: [
      "Distinctness & order",
      "order",
      "Distinctness",
      "All the above"
    ]
  },
  {
    numb: 3,
    question: "The types of an Interval attribute depends on which of the following properties",
    answer: "All 4 properties",
    options: [
      "Distinctness & order",
      "order",
      "Distinctness",
      "All 4 properties"
    ]
  },
  {
    numb: 4,
    question: "The types of an Ordinal attribute depends on which of the following properties",
    answer: "Distinctness & order",
    options: [
      "Distinctness & order",
      "order",
      "Distinctness",
      "All the above"
    ]
  },
  {
    numb: 5,
    question: "Important Characteristics of structured data are",

```

```

    answer: "All the above",
    options: [
        "Generality",
        "Dimentionality",
        "Resolution",
        "All the above"
    ]
},

{
    numb: 6,
    question: "What are some examples of data quality problems",
    answer: "All the above",
    options: [
        "Noise and outliers",
        "Duplicate data",
        "Missing values",
        "All the above"
    ]
},

{
    numb: 7,
    question: "Whic model is used for encoding the categorical variables?",
    answer: "LabelEncoder",
    options: [
        "LabelEncoder",
        "OneHotEncoder",
        "CategoryEncoder",
        "All the above"
    ]
},

{
    numb: 8,
    question: "Under fitting happens due to",
    answer: "A fewer number of features",
    options: [
        "A fewer number of features",
        "Data has a high variance",
        "No use of regularization",
        "None of the mentioned"
    ]
},

{
    numb: 9,
    question: "Overfitting happens due to",
    answer: "All the above",
    options: [
        "Imbalance in data",
        "Noise in data",
        "Data has a high variance",
        "All the above"
    ]
},

```

```
{
  numb: 10,
  question: "Which algorithm does not require feature scaling?",
  answer: "None",
  options: [
    "Naive Bayes",
    "Decision Tree",
    "C.B. Decision Tree",
    "None"
  ]
},
];
```

5. Scripts.js

```
//selecting all required elements
const start_btn = document.querySelector('.start_btn button');
const info_box = document.querySelector('.info_box');
const exit_btn = info_box.querySelector('.buttons .quit');
const continue_btn = info_box.querySelector('.buttons .restart');
const quiz_box = document.querySelector('.quiz_box');
const result_box = document.querySelector('.result_box');
const option_list = document.querySelector('.option_list');
const time_line = document.querySelector('header .time_line');
const timeText = document.querySelector('.timer .time_left_txt');
const timeCount = document.querySelector('.timer .timer_sec');

// if startQuiz button clicked
start_btn.onclick = () => {
  info_box.classList.add('activeInfo'); //show info box
};

// if exitQuiz button clicked
exit_btn.onclick = () => {
  info_box.classList.remove('activeInfo'); //hide info box
};

// if continueQuiz button clicked
continue_btn.onclick = () => {
  info_box.classList.remove('activeInfo'); //hide info box
  quiz_box.classList.add('activeQuiz'); //show quiz box
  showQuestions(0); //calling showQuestions function
  queCounter(1); //passing 1 parameter to queCounter
  startTimer(20); //calling startTimer function
  startTimerLine(0); //calling startTimerLine function
};

let timeValue = 20;
let que_count = 0;
let que_numb = 1;
let userScore = 0;
let counter;
let counterLine;
let widthValue = 0;

const restart_quiz = result_box.querySelector('.buttons .restart');
const quit_quiz = result_box.querySelector('.buttons .quit');

// if restartQuiz button clicked
restart_quiz.onclick = () => {
  quiz_box.classList.add('activeQuiz'); //show quiz box
  result_box.classList.remove('activeResult'); //hide result box
  timeValue = 20;
  que_count = 0;
  que_numb = 1;
  userScore = 0;
```

```

widthValue = 0;
showQuestions(que_count); //calling showQuestions function
queCounter(que_numb); //passing que_numb value to queCounter
clearInterval(counter); //clear counter
clearInterval(counterLine); //clear counterLine
startTimer(timeValue); //calling startTimer function
startTimerLine(widthValue); //calling startTimerLine function
timeText.textContent = 'Time Left'; //change the text of timeText to Time Left
next_btn.classList.remove('show'); //hide the next button
};

// if quitQuiz button clicked
quit_quiz.onclick = () => {
    window.location.reload(); //reload the current window
};

const next_btn = document.querySelector('footer .next_btn');
const bottom_ques_counter = document.querySelector('footer .total_que');

// if Next Que button clicked
next_btn.onclick = () => {
    if (que_count < questions.length - 1) {
        //if question count is less than total question length
        que_count++; //increment the que_count value
        que_numb++; //increment the que_numb value
        showQuestions(que_count); //calling showQuestions function
        queCounter(que_numb); //passing que_numb value to queCounter
        clearInterval(counter); //clear counter
        clearInterval(counterLine); //clear counterLine
        startTimer(timeValue); //calling startTimer function
        startTimerLine(widthValue); //calling startTimerLine function
        timeText.textContent = 'Time Left'; //change the timeText to Time Left
        next_btn.classList.remove('show'); //hide the next button
    } else {
        clearInterval(counter); //clear counter
        clearInterval(counterLine); //clear counterLine
        showResult(); //calling showResult function
    }
};

// getting questions and options from array
function showQuestions(index) {
    const que_text = document.querySelector('.que_text');

    //creating a new span and div tag for question and option and passing the value using array index
    let que_tag =
        '<span>' +
        questions[index].numb +
        '.' +
        questions[index].question +
        '</span>';
    let option_tag =
        '<div class="option"><span>' +
        questions[index].options[0] +
        '</span></div>' +
        '<div class="option"><span>' +

```

```

questions[index].options[1] +
'</span></div>' +
'<div class="option"><span>' +
questions[index].options[2] +
'</span></div>' +
'<div class="option"><span>' +
questions[index].options[3] +
'</span></div>';
que_text.innerHTML = que_tag; //adding new span tag inside que_tag
option_list.innerHTML = option_tag; //adding new div tag inside option_tag

const option = option_list.querySelector('.option');

// set onclick attribute to all available options
for (i = 0; i < option.length; i++) {
  option[i].setAttribute('onclick', 'optionSelected(this)');
}
}
// creating the new div tags which for icons
let tickIconTag = '<div class="icon tick"><i class="fas fa-check"></i></div>';
let crossIconTag = '<div class="icon cross"><i class="fas fa-times"></i></div>';

//if user clicked on option
function optionSelected(answer) {
  clearInterval(counter); //clear counter
  clearInterval(counterLine); //clear counterLine
  let userAns = answer.textContent; //getting user selected option
  let correcAns = questions[que_count].answer; //getting correct answer from array
  const allOptions = option_list.children.length; //getting all option items

  if (userAns === correcAns) {
    //if user selected option is equal to array's correct answer
    userScore += 1; //upgrading score value with 1
    answer.classList.add('correct'); //adding green color to correct selected option
    answer.insertAdjacentHTML('beforeend', tickIconTag); //adding tick icon to correct selected option
    console.log('Correct Answer');
    console.log('Your correct answers = ' + userScore);
  } else {
    answer.classList.add('incorrect'); //adding red color to correct selected option
    answer.insertAdjacentHTML('beforeend', crossIconTag); //adding cross icon to correct selected option
    console.log('Wrong Answer');

    for (i = 0; i < allOptions; i++) {
      if (option_list.children[i].textContent === correcAns) {
        //if there is an option which is matched to an array answer
        option_list.children[i].setAttribute('class', 'option correct'); //adding green color to matched option
        option_list.children[i].insertAdjacentHTML('beforeend', tickIconTag); //adding tick icon to matched option
        console.log('Auto selected correct answer.');
```

```

function showResult() {
  info_box.classList.remove('activeInfo'); //hide info box
  quiz_box.classList.remove('activeQuiz'); //hide quiz box
  result_box.classList.add('activeResult'); //show result box

  fetch('http://localhost:3000/send-mail', {
    method: 'GET',
  })
  .then(() => {
    console.log('email sent');
  })
  .catch(console.log);
  const scoreText = result_box.querySelector('.score_text');
  if (userScore > 3) {
    // if user scored more than 3
    //creating a new span tag and passing the user score number and total question number
    let scoreTag =
      '<span>and congrats! 🎉, You got <p>' +
      userScore +
      '</p> out of <p>' +
      questions.length +
      '</p></span>';
    scoreText.innerHTML = scoreTag; //adding new span tag inside score_Text
  } else if (userScore > 1) {
    // if user scored more than 1
    let scoreTag =
      '<span>and nice 😊, You got <p>' +
      userScore +
      '</p> out of <p>' +
      questions.length +
      '</p></span>';
    scoreText.innerHTML = scoreTag;
  } else {
    // if user scored less than 1
    let scoreTag =
      '<span>and sorry 😞, You got only <p>' +
      userScore +
      '</p> out of <p>' +
      questions.length +
      '</p></span>';
    scoreText.innerHTML = scoreTag;
  }
}

function startTimer(time) {
  console.log('time : ', time);
  counter = setInterval(timer, 1000);
  function timer() {
    timeCount.textContent = time; //changing the value of timeCount with time value
    time--; //decrement the time value
    if (time < 9) {
      //if timer is less than 9
      let addZero = timeCount.textContent;
      timeCount.textContent = '0' + addZero; //add a 0 before time value
    }
  }
}

```



```

if (time < 0) {
  //if timer is less than 0
  clearInterval(counter); //clear counter
  timeText.textContent = 'Time Off'; //change the time text to time off
  const allOptions = option_list.children.length; //getting all option items
  let correcAns = questions[que_count].answer; //getting correct answer from array
  // for(i=0; i < allOptions; i++){
  //   if(option_list.children[i].textContent === correcAns){ //if there is an option which is matched to an array
answer
  //     option_list.children[i].setAttribute("class", "option correct"); //adding green color to matched option
  //     option_list.children[i].insertAdjacentHTML("beforeend", tickIconTag); //adding tick icon to matched
option
  //     console.log("Time Off: Auto selected correct answer.");
  //   }
  // }
  for (i = 0; i < allOptions; i++) {
    option_list.children[i].classList.add('disabled'); //once user select an option then disabled all options
  }
  next_btn.classList.add('show'); //show the next button if user selected any option
}
}
}

function startTimerLine(time) {
  counterLine = setInterval(timer, 37);
  function timer() {
    time += 1; //upgrading time value with 1
    time_line.style.width = time + 'px'; //increasing width of time_line with px by time value
    if (time > 549) {
      //if time value is greater than 549
      clearInterval(counterLine); //clear counterLine
    }
  }
}

function queCounter(index) {
  //creating a new span tag and passing the question number and total question
  let totalQueCounTag =
    '<span><p>' +
    index +
    '</p> of <p>' +
    questions.length +
    '</p> Questions</span>';
  bottom_ques_counter.innerHTML = totalQueCounTag; //adding new span tag inside bottom_ques_counter
}

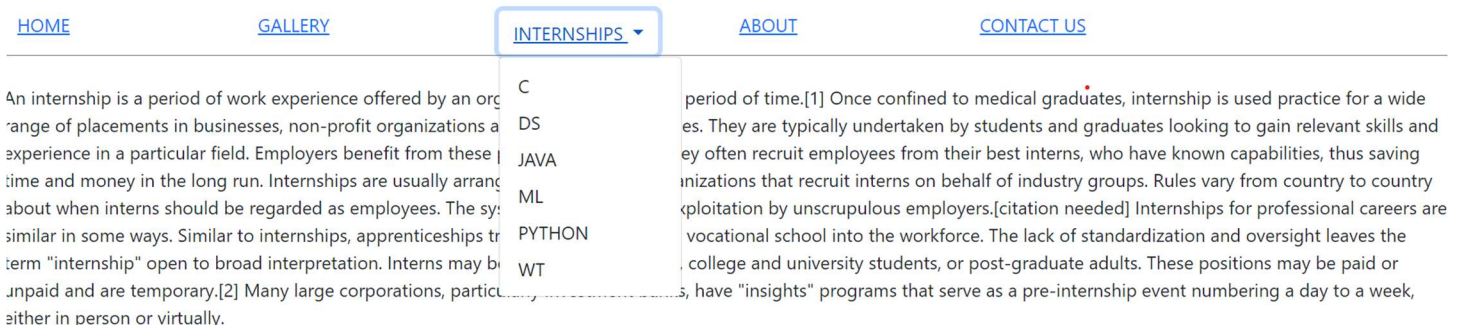
```

5. RESULT

1. Go to the website which is at localhost:3000/home/



2. Click on the desired Internship



3. Redirecting to the respected Internship page



GOALS

- Get hands-on programming exposure in JAVA programming the way Industry works
- Build a good project based on JAVA programming by exploring all C programming constructs
- Get exposure in Standard Industry Practices of Software Development

TIME DURATION

- 6-weeks
- 1 5hr live virtual classes

4. Click on APPLY NOW button

MATERIALS

- Technical Resource Library: Plenty of quality Embedded Systems free course materials available on-line
- Emertxe official blog: Knowledge sharing and industry trends update to students
- Emertxe's YouTube Channel: Tutorial and How-to video

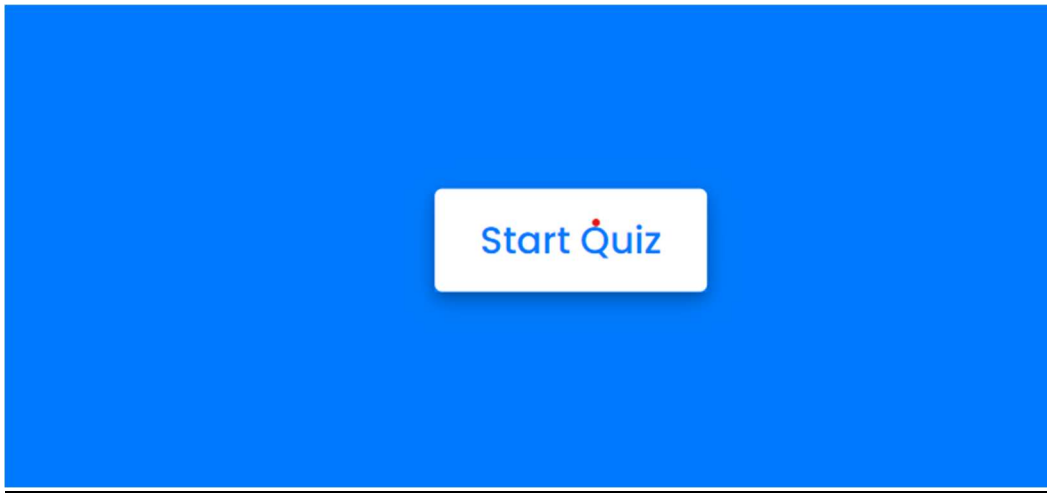
APPLY NOW

5. **Fill the submit form and then click submit**

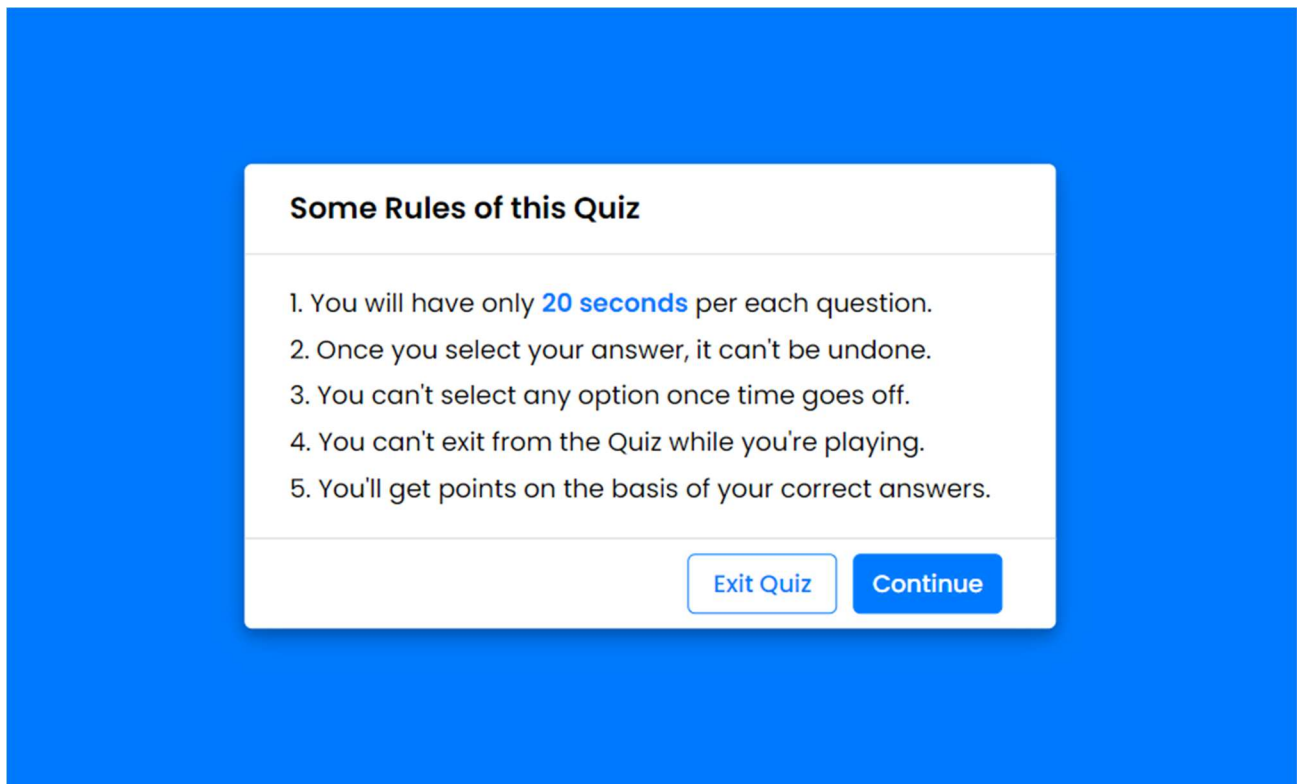
FORM DETAILS

NAME	<input type="text"/>
AGE	<input type="text"/>
GENDER	<input type="radio"/> :MALE <input type="radio"/> :FEMALE <input type="radio"/> :OTHERS
FLUENCY	<input type="checkbox"/> TELUGU <input type="checkbox"/> HINDI <input type="checkbox"/> ENGLISH
YEAR OF STUDY	<input type="text"/>
GPA	<input type="text"/>
PROJECTS	<input type="text"/>
EMAIL	<input type="text"/>
	<input type="button" value="SUBMIT"/> <input type="button" value="RESET"/>


6. Click on start quiz



7. Click on continue



8. Click on your answer and then click next



Awesome Quiz Application Time Left **17**

1. Who developed the JAVA programming Language

Wick Van Rossum

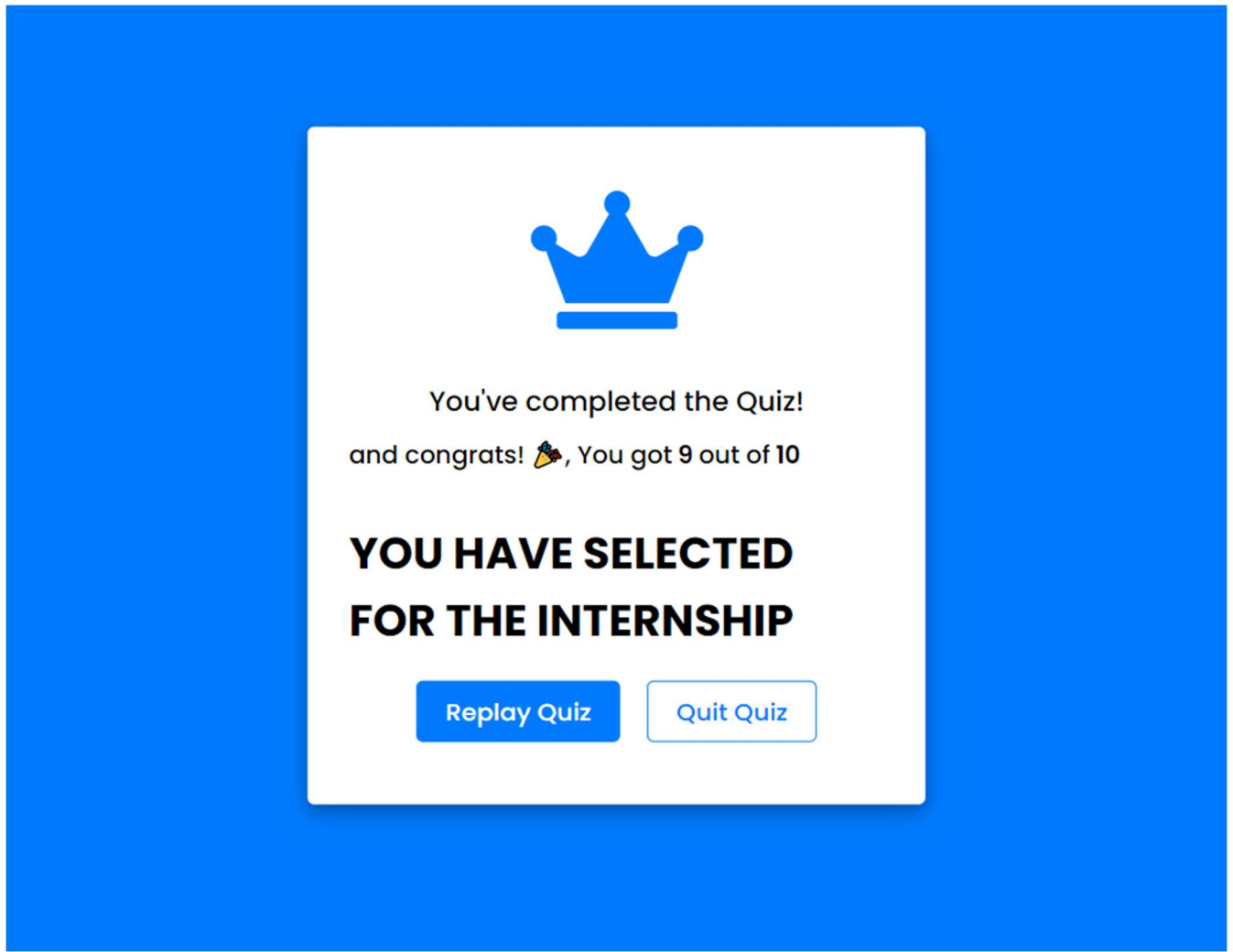
Rasmus Lerdorf

Guido Van Rossum

James Goslig

1 of 10 Questions

9. You will get the result whether you are selected for the Internship or not



6. CONCLUSION

The website will be available when the server starts then the students can apply for their desired Internships and get to know whether they are selected for that respected Internship or not.

7. REFERENCES

- <https://www.w3schools.com/html/>
- <https://www.w3schools.com/w3css/defaultT.asp>
- <https://www.w3schools.com/js/>
- <https://www.w3schools.com/bootstrap/>
- <https://www.w3schools.com/nodejs/>