

Code

Index.js:

```
const express = require("express");
const bodyParser = require("body-parser");
const path = require("path");
var cookieParser = require("cookie-parser");
var cons = require("consolidate");
const { User } = require("./routes/User");
const { CanvasRenderService } = require("chartjs-node-canvas");

const app = express();
app.use(cookieParser());

app.set("port", 8000);
app.set("views", __dirname + "/views");
app.engine("html", cons.swig);
app.set("view engine", "ejs");
app.use(bodyParser.urlencoded({ extended: false }));
app.use(bodyParser.json());
app.use(express.static(path.join(__dirname + "/public")));
app.use(express.static("images"));
app.use("/css", express.static(__dirname + "/node_modules/bootstrap/dist/css"));
app.use("/js", express.static(__dirname + "/node_modules/chart.js/dist"));

function shuffleArray(array) {
  for (var i = array.length - 1; i > 0; i--) {
    var j = Math.floor(Math.random() * (i + 1));
    var temp = array[i];
    array[i] = array[j];
    array[j] = temp;
  }

  return array;
}

const user = new User();

app.all("/", (req, res) => {
  res.render("index", {
    title: "title",
  });
});

app.all("/authenticate", async (req, res) => {
  const { name, email } = req.body;
  res.cookie("name", name);
  res.cookie("email", email);
  user.setUserdetails(name, email);
  const response = await fetch("https://the-trivia-api.com/v2/questions").then(
    (data) => {
      return data.json();
    }
  );

  Object.keys(response).forEach((key) => {
    user.addQuestion({
      question: response[key].question.text,
      options: shuffleArray([
        response[key].correctAnswer,
        ...response[key].incorrectAnswers,
      ]),
      correctAnswer: response[key].correctAnswer,
    });
  });
});
```

```

    });
  });
  res.redirect("quiz");
});

app.all("/quiz", async (req, res) => {
  if (user.getUserDetails().email == undefined) {
    res.redirect("index");
  }
  const { name, email } = user.getUserDetails();

  res.render("quiz", {
    name: name,
    email: email,
    questions: user.getQuestions(),
  });
});

app.all("/verifyResults", (req, res) => {
  const { correct, incorrect } = user.checkAnswers(req.body);
  const results = [correct, incorrect];
  res.render("results", {
    results,
  });
});

app.listen(8000, () => {
  console.log("Server running at 8000");
});

```

Index.ejs:

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Online Quiz Application</title>
    <link rel="stylesheet" href="/css/bootstrap.min.css" />
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <div class="wrapper">
      <p class="h2 text-center">Welcome to Onlie Quiz Application</p>
      <div class="container w-50">
        <form action="/authenticate" method="post" class="form">
          <div class="form-group">
            <label for="name">Name:</label>
            <input type="text" name="name" id="name" class="form-control" />
          </div>
          <div class="form-group">
            <label for="email">Email:</label>
            <input type="email" name="email" id="email" class="form-control" />
          </div>
          <div class="form-group">
            <input
              type="submit"
              id="email"
              class="form-control btn btn-primary"
              value="Start Quiz"
            />
          </div>
        </form>
      </div>
    </div>
  </body>
</html>

```

```

    </form>
  </div>
</div>
</body>
</html>

```

Quiz.ejs

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="/css/bootstrap.min.css" />
    <script src="/js/chart.js"></script>
    <link rel="stylesheet" href="style.css" />
    <title>Quiz</title>
  </head>
  <body>
    <div class="container w-75">
      <h1 class="text-center">Quiz</h1>
      <h5 class="text-danger">* Please Attend All the questions</h5>
      <form action="/verifyResults" class="form" method="post">
        <% for( let index = 0; index < questions.length; index++ ) { %>
          <div class="card form-group p-2">
            <label class="h4" class="form-label card-title">
              <%= index + 1%> <%= questions[index].question %>
            <span class="text-danger">*</span></label>
            <div class="card-body">
              <% for( let op = 0; op < questions[index].options.length; op++ ) { %>
                <div class="form-check">
                  <input
                    required
                    type="radio"
                    name="<%=index%>"
                    id="op<%=index%><%=op%>"
                    class="form-check-input"
                    value="<%=questions[index].options[op]%>"
                  />
                  <label class="form-check-label" for="op<%=index%><%=op%>"
                    ><%=questions[index].options[op]%></label>
                </div>
              <% } %>
            </div>
          </div>
        <% } %>
        <div class="form-group">
          <input
            type="submit"
            value="Submit"
            class="btn btn-block btn-primary"
          />
        </div>
      </form>
    </div>
  </body>
</html>

```

Results.ejs

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Results</title>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.9.4/Chart.js"></script>
    <link rel="stylesheet" href="/css/bootstrap.min.css" />
    <link rel="stylesheet" href="style.css" />
  </head>
  <body onload="createGraph('<%= results %>')">
    <div class="wrapper">
      <h2>Quiz Results</h2>
      <h1><%= results[0] *10 %></h1>
      <span>(Pass percentage is 50%)</span>
      <div>
        <% if ((results[0]/10)*100 <50) { %>
        <p class="h5 text-danger">Better luck next time</p>
        <% } else if ((results[0]/10)*100 >=50) { %>
        <p class="h5 text-success">You have successfully passed the quiz</p>
        <% } %>
      </div>
      <canvas id="myChart" style="width: 100%; max-width: 700px"></canvas>
    </div>

    <script>
      createGraph = function (results) {
        var xValues = ["Incorrect", "Correct"];
        var barColors = ["#3cb371", "#ff0000"];
        var yValues = results.split(",").map((values) => parseInt(values));
        new Chart("myChart", {
          type: "pie",
          data: {
            labels: xValues,
            datasets: [
              {
                backgroundColor: barColors,
                data: yValues,
              },
            ],
          },
          options: {
            title: {
              display: false,
              text: "Quiz results",
            },
          },
        });
      };
    </script>
  </body>
</html>

```

User.js:

```

module.exports = {
  User: class User {
    constructor() {}

    setUserdetails(name, email) {
      this.name = name;
      this.email = email;
    }
  }
}

```

```

    this.questions = [];
    this.userAnswers = {};
  }

  getUserDetails() {
    return { name: this.name, email: this.email };
  }

  addQuestion(question) {
    this.questions.push(question);
  }

  getQuestions() {
    return this.questions;
  }

  checkAnswers(answers) {
    let crt = 0,
        wrng = 0;
    Object.keys(answers).forEach((key) => {
      if (answers[key] == this.questions[key].correctAnswer) {
        crt++;
      } else {
        wrng++;
      }
    });

    return { correct: crt, incorrect: wrng };
  },
};

```

Style.css

```

.wrapper {
  height: 100vh;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
}

.card {
  transition: box-shadow ease-in-out 300ms;
  border-width: 2px;
  border-radius: 25px 0 25px 0 !important;
}

.card:hover {
  box-shadow: rgba(0, 0, 0, 0.35) 0px 5px 15px !important;
}

.form-check > label {
  cursor: pointer;
}

```