Assignment 2: Introduction to Backend

Scott Freedman

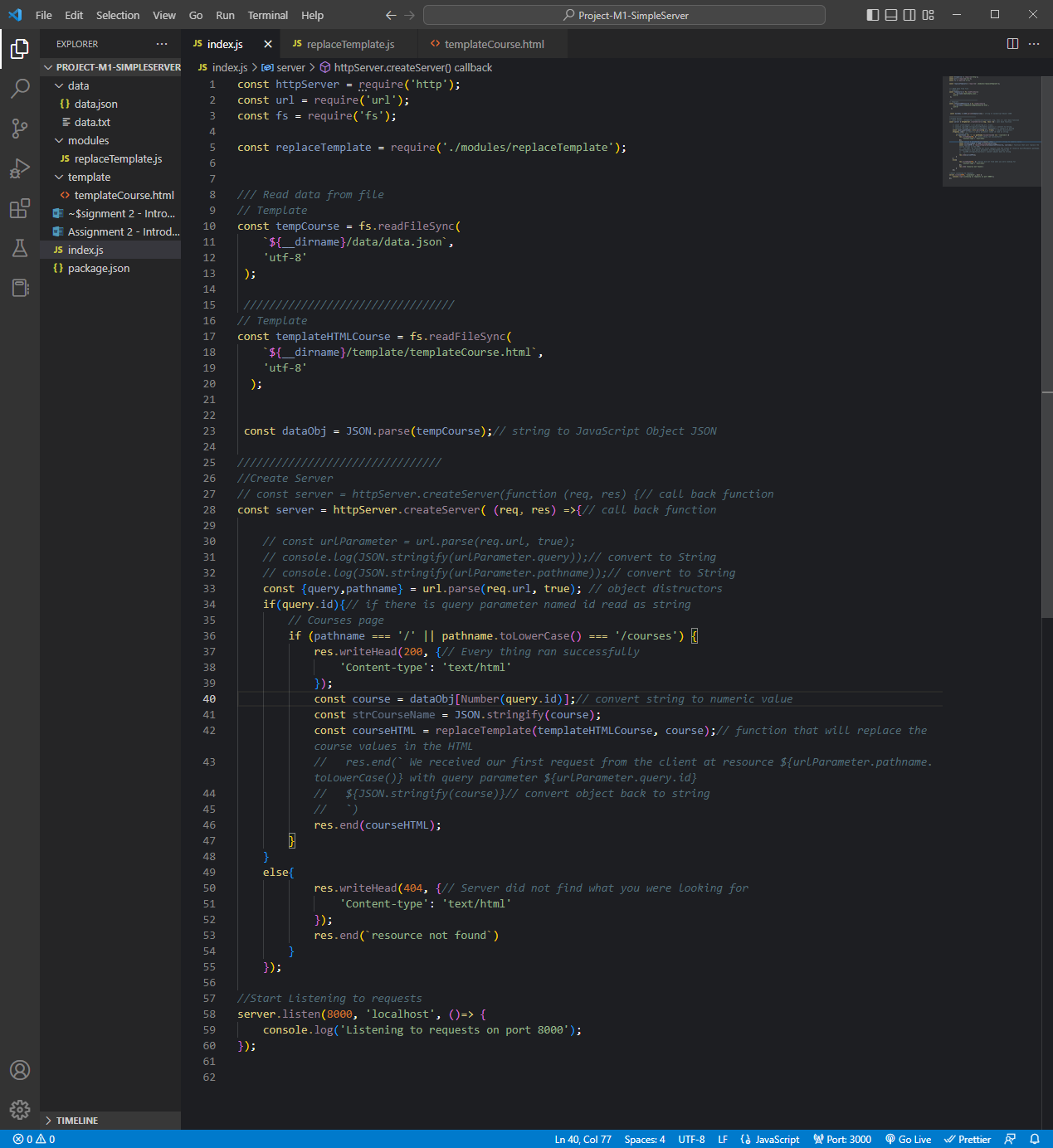
Ira A. Fulton Schools of Engineering, Arizona State University

IFT 458: Middleware Programming & Database Security

Professor Dinesh Sthapit

January 22, 2023

*index.js:*

**

*Code:*

const httpServer = require('http');

const url = require('url');

const fs = require('fs');

const replaceTemplate = require('./modules/replaceTemplate');

*/// Read data from file*

*// Template*

const tempCourse = fs.readFileSync(

    `${\_\_dirname}/data/data.json`,

    'utf-8'

 );

*/////////////////////////////////*

*// Template*

const templateHTMLCourse = fs.readFileSync(

    `${\_\_dirname}/template/templateCourse.html`,

    'utf-8'

  );

 const dataObj = JSON.parse(tempCourse);*// string to JavaScript Object JSON*

*////////////////////////////////*

*//Create Server*

*// const server = httpServer.createServer(function (req, res) {// call back function*

const server = httpServer.createServer( (req, res) =>{*// call back function*

*// const urlParameter = url.parse(req.url, true);*

*// console.log(JSON.stringify(urlParameter.query));// convert to String*

*// console.log(JSON.stringify(urlParameter.pathname));// convert to String*

    const {query,pathname} = url.parse(req.url, true); *// object distructors*

    if(query.id){*// if there is query parameter named id read as string*

*// Courses page*

        if (pathname === '/' || pathname.toLowerCase() === '/courses') {

            res.writeHead(200, {*// Every thing ran successfully*

                'Content-type': 'text/html'

            });

            const course = dataObj[Number(query.id)];*// convert string to numeric value*

            const strCourseName = JSON.stringify(course);

            const courseHTML = replaceTemplate(templateHTMLCourse, course);*// function that will replace the course values in the HTML*

*//   res.end(` We received our first request from the client at resource ${urlParameter.pathname.toLowerCase()} with query parameter ${urlParameter.query.id}*

*//   ${JSON.stringify(course)}// convert object back to string*

*//   `)*

            res.end(courseHTML);

        }

    }

    else{

            res.writeHead(404, {*// Server did not find what you were looking for*

                'Content-type': 'text/html'

            });

            res.end(`resource not found`)

        }

    });

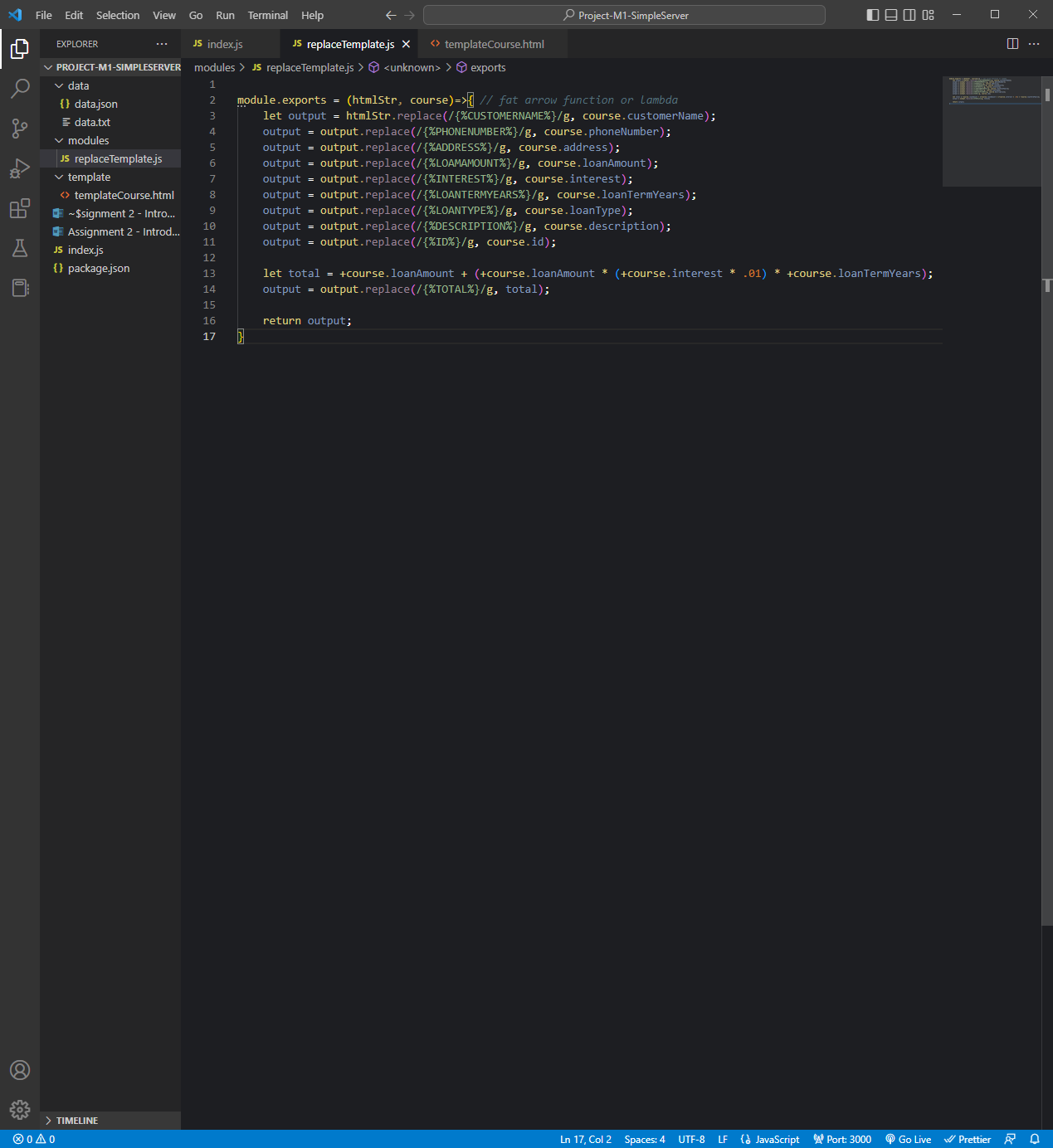
*//Start Listening to requests*

server.listen(8000, 'localhost', ()=> {

    console.log('Listening to requests on port 8000');

});

*replaceTemplate.js:*

**

*Code:*

module.exports = (htmlStr, course)=>{ *// fat arrow function or lambda*

    let output = htmlStr.replace(/{%CUSTOMERNAME%}/g, course.customerName);

    output = output.replace(/{%PHONENUMBER%}/g, course.phoneNumber);

    output = output.replace(/{%ADDRESS%}/g, course.address);

    output = output.replace(/{%LOAMAMOUNT%}/g, course.loanAmount);

    output = output.replace(/{%INTEREST%}/g, course.interest);

    output = output.replace(/{%LOANTERMYEARS%}/g, course.loanTermYears);

    output = output.replace(/{%LOANTYPE%}/g, course.loanType);

    output = output.replace(/{%DESCRIPTION%}/g, course.description);

    output = output.replace(/{%ID%}/g, course.id);

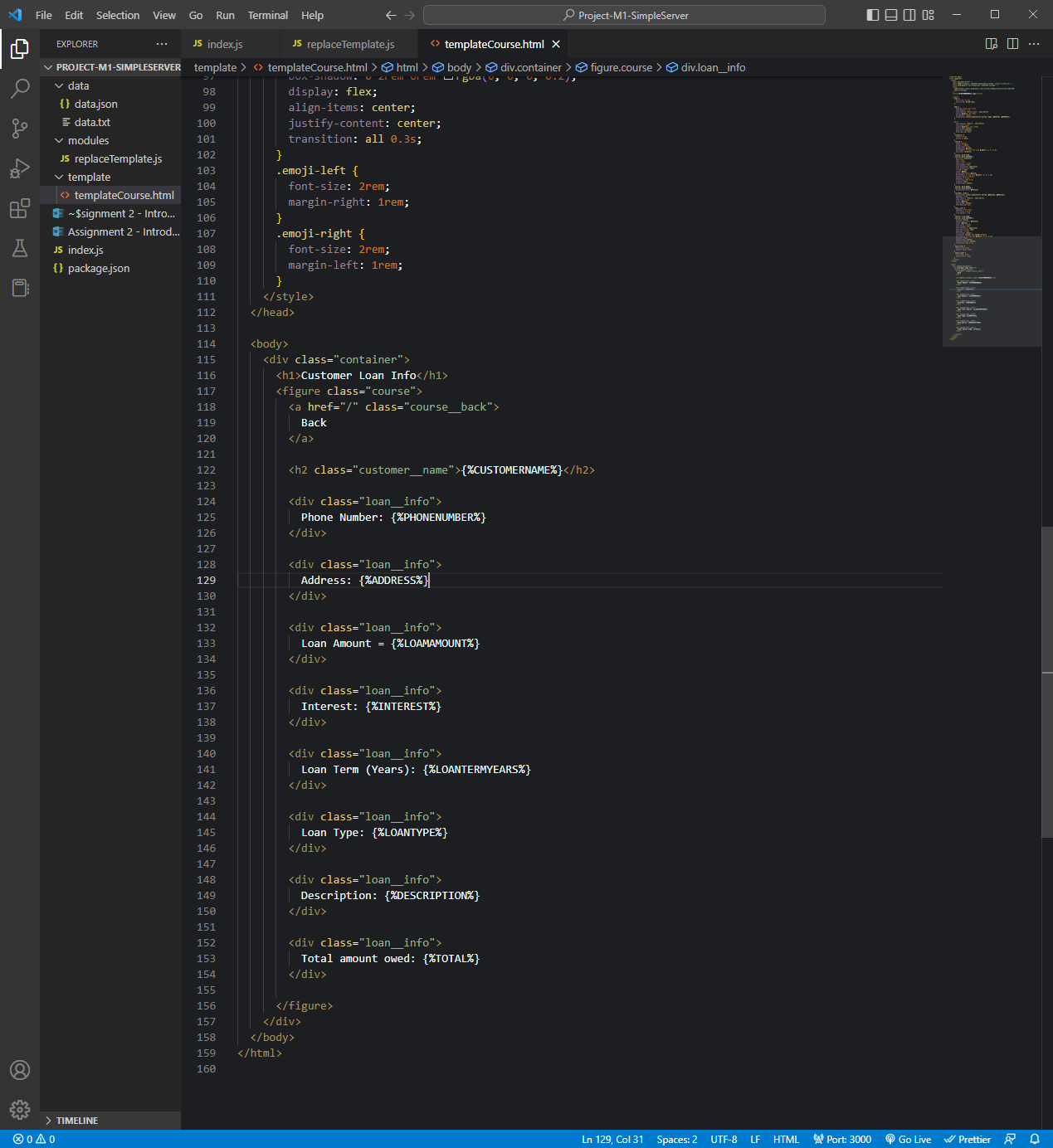
    let total = +course.loanAmount + (+course.loanAmount \* (+course.interest \* .01) \* +course.loanTermYears);

    output = output.replace(/{%TOTAL%}/g, total);

    return output;

}

*templateCourse.html:*

**

*Code:*

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <meta http-equiv="X-UA-Compatible" content="ie=edge" />

    <link

      href="https://fonts.googleapis.com/css?family=Megrim|Nunito+Sans:400,900"

      rel="stylesheet"

    />

    <title>{%CUSTOMERNAME%} Loan</title>

    <style>

      html {

        font-size: 62.5%;

        box-sizing: border-box;

      }

      body {

        padding: 5rem 5rem 5rem;

        line-height: 1.7;

        font-family: 'Nunito Sans', sans-serif;

        color: rgb(85, 85, 85);

        min-height: 100vh;

        background: linear-gradient(to bottom right, #5d75e1, #4800e3);

      }

      h1 {

        font-family: 'Megrim', sans-serif;

        font-size: 6rem;

        color: rgb(255, 255, 255);

        text-align: center;

        position: relative;

        word-spacing: 3px;

      }

      .container {

        width: 95rem;

        margin: 0 auto;

      }

      .course {

        width: 60rem;

        margin: 0 auto;

        margin-top: 9rem;

        background: white;

        box-shadow: 0 3rem 6rem 1rem rgba(0, 0, 0, 0.2);

        position: relative;

      }

      .course\_\_back:link,

      .course\_\_back:visited {

        position: absolute;

        top: 2rem;

        left: 2rem;

        font-size: 1.5rem;

        font-weight: 700;

        text-transform: uppercase;

        text-decoration: none;

        z-index: 1000;

        color: #555;

        background-color: white;

        box-shadow: 0 1rem 3rem rgba(0, 0, 0, 0.3);

        border-radius: 100rem;

        padding: 0 2rem;

        transition: all 0.3s;

        display: flex;

        align-items: center;

      }

      .course\_\_back:hover,

      .course\_\_back:active {

        background-color: #79e17b;

      }

      .customer\_\_name {

        background: linear-gradient(to bottom, #9be15d, #00e3ae);

        padding: 1rem;

        font-family: 'Megrim', sans-serif;

        font-size: 4rem;

        color: white;

        text-align: center;

        word-spacing: 2px;

      }

      .loan\_\_info {

        padding: 2rem 2rem;

        font-size: 1.6rem;

        line-height: 1.8;

      }

      .course\_\_link:link,

      .course\_\_link:visited {

        display: block;

        background-color: #79e17b;

        color: white;

        font-size: 1.6rem;

        font-weight: 700;

        text-transform: uppercase;

        text-decoration: none;

        padding: 1.5rem;

        text-align: center;

        transform: scale(1.07) skewX(-20deg);

        box-shadow: 0 2rem 6rem rgba(0, 0, 0, 0.2);

        display: flex;

        align-items: center;

        justify-content: center;

        transition: all 0.3s;

      }

      .emoji-left {

        font-size: 2rem;

        margin-right: 1rem;

      }

      .emoji-right {

        font-size: 2rem;

        margin-left: 1rem;

      }

    </style>

  </head>

  <body>

    <div class="container">

      <h1>Customer Loan Info</h1>

      <figure class="course">

        <a href="/" class="course\_\_back">

          Back

        </a>

        <h2 class="customer\_\_name">{%CUSTOMERNAME%}</h2>

        <div class="loan\_\_info">

          Phone Number: {%PHONENUMBER%}

        </div>

        <div class="loan\_\_info">

          Address: {%ADDRESS%}

        </div>

        <div class="loan\_\_info">

          Loan Amount = {%LOAMAMOUNT%}

        </div>

        <div class="loan\_\_info">

          Interest: {%INTEREST%}

        </div>

        <div class="loan\_\_info">

          Loan Term (Years): {%LOANTERMYEARS%}

        </div>

        <div class="loan\_\_info">

          Loan Type: {%LOANTYPE%}

        </div>

        <div class="loan\_\_info">

          Description: {%DESCRIPTION%}

        </div>

        <div class="loan\_\_info">

          Total amount owed: {%TOTAL%}

        </div>

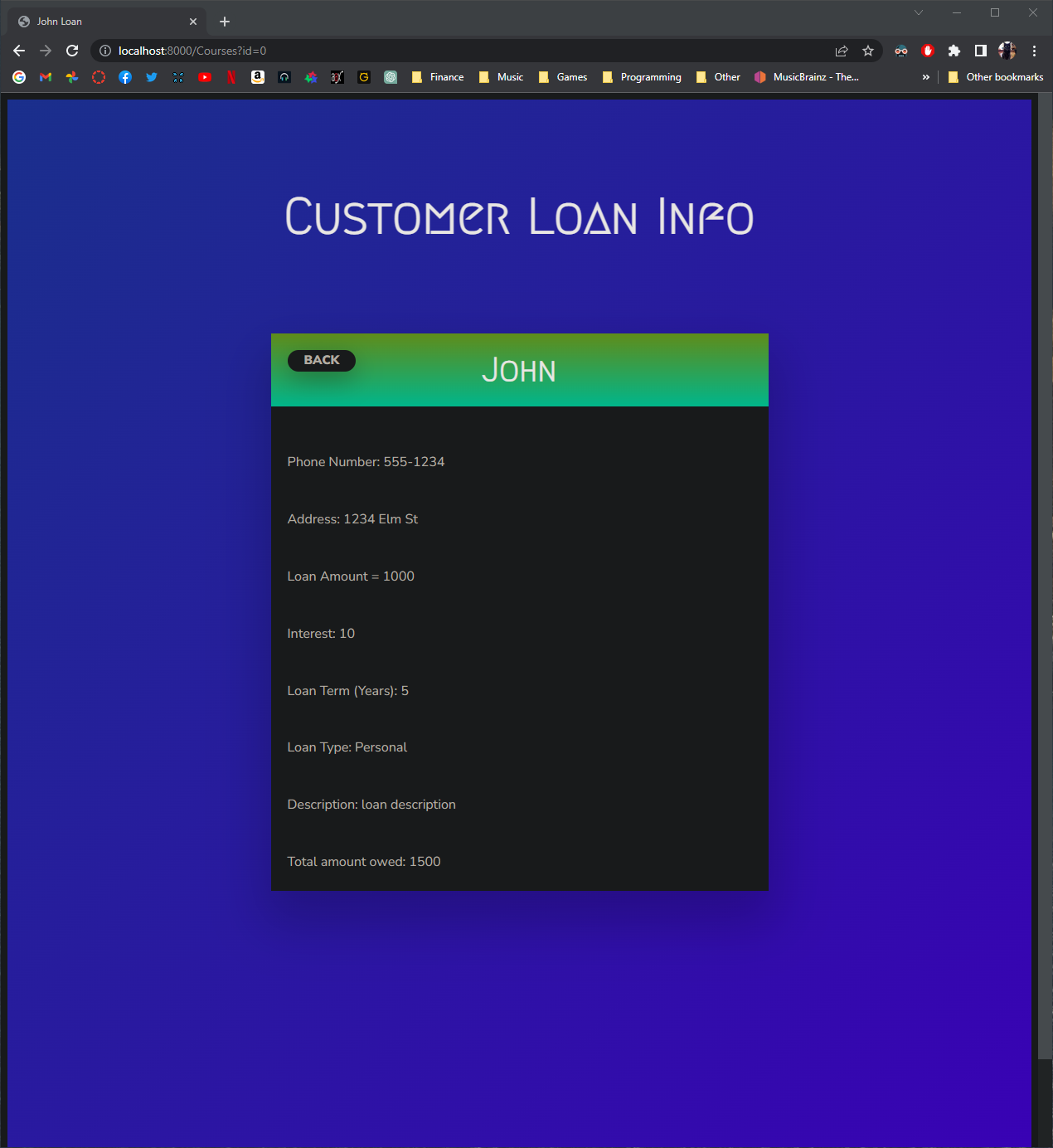
      </figure>

    </div>

  </body>

</html>

*Browser:*

**