I was able to get the webserver online in my own computer. Here is the screenshot:

A picture containing text

Description automatically generated

I was able to access the website on someone else’s computer:

Graphical user interface, application

Description automatically generated

Server Code:

import java.io.BufferedOutputStream;  
import java.io.BufferedReader;  
import java.io.File;  
import java.io.FileInputStream;  
import java.io.FileNotFoundException;  
import java.io.IOException;  
import java.io.InputStreamReader;  
import java.io.OutputStream;  
import java.io.PrintWriter;  
import java.net.ServerSocket;  
import java.net.Socket;  
import java.util.Date;  
import java.util.StringTokenizer;  
  
// The tutorial can be found just here on the SSaurel's Blog :   
// https://www.ssaurel.com/blog/create-a-simple-http-web-server-in-java  
// Each Client Connection will be managed in a dedicated Thread  
public class JavaHTTPServer implements Runnable{   
   
 static final File *WEB\_ROOT* = new File(".");  
 static final String *DEFAULT\_FILE* = "index.html";  
 static final String *FILE\_NOT\_FOUND* = "404.html";  
 static final String *METHOD\_NOT\_SUPPORTED* = "not\_supported.html";  
 // port to listen connection  
 static final int *PORT* = 8080;  
   
 // verbose mode  
 static final boolean *verbose* = true;  
   
 // Client Connection via Socket Class  
 private Socket connect;  
   
 public JavaHTTPServer(Socket c) {  
 connect = c;  
 }  
   
 public static void main(String[] args) {  
 try {  
 ServerSocket serverConnect = new ServerSocket(*PORT*);  
 System.*out*.println("Server started.\nListening for connections on port : " + *PORT* + " ...\n");  
   
 // we listen until user halts server execution  
 while (true) {  
 JavaHTTPServer myServer = new JavaHTTPServer(serverConnect.accept());  
   
 if (*verbose*) {  
 System.*out*.println("Connecton opened. (" + new Date() + ")");  
 }  
   
 // create dedicated thread to manage the client connection  
 Thread thread = new Thread(myServer);  
 thread.start();  
 }  
   
 } catch (IOException e) {  
 System.*err*.println("Server Connection error : " + e.getMessage());  
 }  
 }  
  
 @Override  
 public void run() {  
 // we manage our particular client connection  
 BufferedReader in = null; PrintWriter out = null; BufferedOutputStream dataOut = null;  
 String fileRequested = null;  
   
 try {  
 // we read characters from the client via input stream on the socket  
 in = new BufferedReader(new InputStreamReader(connect.getInputStream()));  
 // we get character output stream to client (for headers)  
 out = new PrintWriter(connect.getOutputStream());  
 // get binary output stream to client (for requested data)  
 dataOut = new BufferedOutputStream(connect.getOutputStream());  
   
 // get first line of the request from the client  
 String input = in.readLine();  
 // we parse the request with a string tokenizer  
 StringTokenizer parse = new StringTokenizer(input);  
 String method = parse.nextToken().toUpperCase(); // we get the HTTP method of the client  
 // we get file requested  
 fileRequested = parse.nextToken().toLowerCase();  
   
 // we support only GET and HEAD methods, we check  
 if (!method.equals("GET") && !method.equals("HEAD")) {  
 if (*verbose*) {  
 System.*out*.println("501 Not Implemented : " + method + " method.");  
 }  
   
 // we return the not supported file to the client  
 File file = new File(*WEB\_ROOT*, *METHOD\_NOT\_SUPPORTED*);  
 int fileLength = (int) file.length();  
 String contentMimeType = "text/html";  
 //read content to return to client  
 byte[] fileData = readFileData(file, fileLength);  
   
 // we send HTTP Headers with data to client  
 out.println("HTTP/1.1 501 Not Implemented");  
 out.println("Server: Java HTTP Server from SSaurel : 1.0");  
 out.println("Date: " + new Date());  
 out.println("Content-type: " + contentMimeType);  
 out.println("Content-length: " + fileLength);  
 out.println(); // blank line between headers and content, very important !  
 out.flush(); // flush character output stream buffer  
 // file  
 dataOut.write(fileData, 0, fileLength);  
 dataOut.flush();  
   
 } else {  
 // GET or HEAD method  
 if (fileRequested.endsWith("/")) {  
 fileRequested += *DEFAULT\_FILE*;  
 }  
   
 File file = new File(*WEB\_ROOT*, fileRequested);  
 int fileLength = (int) file.length();  
 String content = getContentType(fileRequested);  
   
 if (method.equals("GET")) { // GET method so we return content  
 byte[] fileData = readFileData(file, fileLength);  
   
 // send HTTP Headers  
 out.println("HTTP/1.1 200 OK");  
 out.println("Server: Java HTTP Server from SSaurel : 1.0");  
 out.println("Date: " + new Date());  
 out.println("Content-type: " + content);  
 out.println("Content-length: " + fileLength);  
 out.println(); // blank line between headers and content, very important !  
 out.flush(); // flush character output stream buffer  
   
 dataOut.write(fileData, 0, fileLength);  
 dataOut.flush();  
 }  
   
 if (*verbose*) {  
 System.*out*.println("File " + fileRequested + " of type " + content + " returned");  
 }  
   
 }  
   
 } catch (FileNotFoundException fnfe) {  
 try {  
 fileNotFound(out, dataOut, fileRequested);  
 } catch (IOException ioe) {  
 System.*err*.println("Error with file not found exception : " + ioe.getMessage());  
 }  
   
 } catch (IOException ioe) {  
 System.*err*.println("Server error : " + ioe);  
 } finally {  
 try {  
 in.close();  
 out.close();  
 dataOut.close();  
 connect.close(); // we close socket connection  
 } catch (Exception e) {  
 System.*err*.println("Error closing stream : " + e.getMessage());  
 }   
   
 if (*verbose*) {  
 System.*out*.println("Connection closed.\n");  
 }  
 }  
   
   
 }  
   
 private byte[] readFileData(File file, int fileLength) throws IOException {  
 FileInputStream fileIn = null;  
 byte[] fileData = new byte[fileLength];  
   
 try {  
 fileIn = new FileInputStream(file);  
 fileIn.read(fileData);  
 } finally {  
 if (fileIn != null)   
 fileIn.close();  
 }  
   
 return fileData;  
 }  
   
 // return supported MIME Types  
 private String getContentType(String fileRequested) {  
 if (fileRequested.endsWith(".htm") || fileRequested.endsWith(".html"))  
 return "text/html";  
 else  
 return "text/plain";  
 }  
   
 private void fileNotFound(PrintWriter out, OutputStream dataOut, String fileRequested) throws IOException {  
 File file = new File(*WEB\_ROOT*, *FILE\_NOT\_FOUND*);  
 int fileLength = (int) file.length();  
 String content = "text/html";  
 byte[] fileData = readFileData(file, fileLength);  
   
 out.println("HTTP/1.1 404 File Not Found");  
 out.println("Server: Java HTTP Server from SSaurel : 1.0");  
 out.println("Date: " + new Date());  
 out.println("Content-type: " + content);  
 out.println("Content-length: " + fileLength);  
 out.println(); // blank line between headers and content, very important !  
 out.flush(); // flush character output stream buffer  
   
 dataOut.write(fileData, 0, fileLength);  
 dataOut.flush();  
   
 if (*verbose*) {  
 System.*out*.println("File " + fileRequested + " not found");  
 }  
 }  
   
}

HTML files

<html>  
<head>  
<title>  
About Me  
</title>  
</head>  
<body background = "MainBG.png">  
<center>  
<table bgcolor = "eaeaea" width = "450" border = "0">  
<tr>  
<td valign = "top" width = "50%">  
 Bible Verse  
</td>  
<td valign = "top" width = "50%">  
My favorite Bible verse is Mark 11:24.<br><br>  
 Therefore I tell you, whatever you ask for in prayer,<br>  
&emsp;believe that you have received it, and it will be yours.<br>  
</td>  
</tr>  
<tr bgcolor = "eeaaaa">  
<td valign = "top">  
I am a Computer Scientist.  
</td>  
<td valign = "top">  
I am designing a food review app and implementing it with four other people.  
</td>  
</tr>  
  
</table>  
</center>  
</body>  
</html>

<html>  
<head>  
<title>  
Contact Information  
</title>  
</head>  
<body background = "MainBG.png">  
<center>  
<table bgcolor = "eaeaea" width = "450" border = "0">  
<tr>  
<td valign = "top" width = "50%">  
Contact me for more information!  
 Email is ryanthomas.miller@calbaptist.edu  
</td>  
<td valign = "top" width = "50%">  
My favorite Bible verse is Mark 11:24.<br><br>  
 Therefore I tell you, whatever you ask for in prayer, <br>  
&emsp;believe that you have received it, and it will be yours.<br>  
  
</td>  
</tr>  
</table>  
</center>  
</body>  
</html>

<html>  
<head>  
<title>  
Top Menu  
</title>  
</head>  
<body background = "TopBG.png">  
<br>  
<center><h2>My Compact Webpage</h2>  
<a target = "main" href = "AboutMe.html"><img height="20" src = "AboutMe.png"></a> ... <a target = "main" href = "MyHobbies.html"><img height = "20" src = "MyHobbies.png"></a> ... <a target = "main" href = "ContactInfo.html"><img height = "20" src = "Contact.png"></a>  
</center>  
</body>  
</html>

<html>  
<head>  
<title>  
My Hobbies  
</title>  
</head>  
<body background = "MainBG.png">  
<center>  
<table bgcolor = "eaeaea" width = "450" border = "0">  
<tr>  
<td valign = "top" width = "50%">  
Dungeons and Dragons<br>  
  
</td>  
<td valign = "top" width = "50%">  
<br>  
&emsp;I play D&D at least for four hours a week.  
 I then outside of that will plan out my sessions for my players with a lot of prep work. <br>  
&emsp;<br>  
&emsp;I either host my sessions online through discord or in person at my apartment.  
</td>  
</tr>  
<tr bgcolor = "aaeeee">  
<td valign = "top">  
Video Games  
</td>  
<td valign = "top">  
Recently, I purchased Elden Ring created by From Software.  
This game falls into the Souls-Like genre. <br><br>  
The game is focused on fighting massive monsters with extremely hard movesets.  
 The game puts you into world where it seems helpless but you have the power to defeat these monsters.  
</td>  
</tr>  
<tr>  
<td valign = "top">  
Lego Sets<br>  
I collect Lego Star Wars sets.  
</td>  
<td valign = "top">  
Sets  
<list>  
<li> The Ultimate Collectors Edition Millenium Falcon  
 <!-- Allegory and Illustration in the Book of Genesis.<br> -->  
<li> I also have the entire collection of revamped Star Wars Sets from the initial release of this line.  
 </list>  
<br><br>  
  
</td>  
</tr>  
</table>  
</center>  
</body>  
</html>