Course Project Report

Student: Reeve Jarvis

Intro to JavaScript

Course code: DGL-113 DLU1

Instructor: Frank Lu

Due Date: 04/06/2021

Submitted: 04/04/2021

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Table of Contents:

Cover Page 1

Table of Contents 2

Section 1: Website Background/Purpose 3

Section 2: New Features Added 4

Section 3: Implementation Techniques 5-13

Section 4: Browser/Wave Testing 14-23

Section 5: Validation 24-38

Summary and Feedback 39

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Section 1: Website Background and Purpose

This website has been adapted from a previous course project for DGL-103 in Semester 1 of the DGL program (Fall 2020). I have chosen to continue work on the site and add increased functionality using the skills and concepts learned throughout this course, “Intro to JavaScript”. This website will be a project that I continue to adapt as my skills grow as a developer.

Excerpt from Prior Website Project Report:

> *“This website will serve as a showcase of my abilities and continue to grow as I build more content through my student career. Upon completing my program at NIC, I will continue to develop the site as a portfolio and contact page for prospective employers and freelance contractors.”*

My Website is comprised of the following pages:

1. Home Page:

My home page is an entry point to the website. It will be a simple welcome page, introducing my potential “business” and indicating the purpose of the website to viewers.

2. About Me:

The About Me page is an introduction and personal profile. It includes short profiles of important members of my life. Additionally, this page contains a gallery of photographs of my adventures.

3. Game Room:

The Game Room page is a more personal addition to the website and will primarily be focussed on my passion for video games. It presents a list of the games I am currently playing as well as my top three games of all time with insights into why I consider them to be of such a high standard.

4. Portfolio:

The Portfolio page is a small display of my completed work throughout my school career and beyond (in need of further content updates outside the scope of this assignment requirements).

5. Contact:

The Schedule and Contact page is a place for visitors to find my current schedule and availability as well as provide options to contact me through multiple channels.

Section 2: New Features Added

I have used many of the JavaScript techniques learned throughout the semester to increase the functionality of my website. Prior to these updates the website was purely static and contained little interactivity for the user save for a few CSS hover effects. I wanted to use JavaScript to increase the usability of the site and create a more engaging experience. Additionally, I want this website to be a functional tool for me to use throughout my career.

New features I have added to the website include:

1. "Dummy" Log-In/Out Credentials:
In order to simulate my future intentions for this website, I have added a "pretend" admin log-in. I say this is "pretend" as one look at the code and you will be able to obtain the hard-coded credentials. I wanted to implement this to practice having additional functionality as the admin to my own website. When logged in, I can access my calendar scheduling detailed below. I have also added the ability to log-out and return to the normal presentation.

2. Cookie/localStorage Persistence:
I have implemented cookies to maintain the admin access features, as well as keep a persistent theme on each page of the website. Additionally, I have used localStorage in the functionality of my calendar scheduling functions. This is a temporary solution to the lack of server/database access allowed by GitHub pages hosting and demonstrates my future intentions.

3. Theme Selection:
I have added a theme selection button to the footer of the website. Due to the vibrant/colourful nature of my initial design, I thought it would be fitting to implement a Black and White mode. This theme is persistent by using cookies and handles the toggling of colour for every element on the website including photos.

4. Gallery Modal Images:
On my About Me page, I have added increased functionality to my Gallery. This allows the user to click images and display them in a pop-out modal box on top of the rest of the page.

5. DOM Generated Calendar and Scheduling:
I have increased the functionality of my Contact Page by dynamically creating the calendar using JavaScript. The calendar contents are created based on current date information and present accurate days of each month. It includes the ability to navigate months and responds to the selections made in my request form below. I have added some functionality to my calendar, only accessible when logged in as Admin. This allows me to add and remove busy dates in my schedule and maintain them using localStorage and JSON serialization.

Section 3: Implementation Techniques

Implementing the features discussed in Section 2 required using the combined skills I have developed thus far in the DGL program. In addition to techniques learned throughout the course, I have researched further coding practises online to ensure my website responds as intended. Implementing the calendar for example required some significant troubleshooting and patience. I have included screenshots with code and comments to further present my implementation.

Techniques Used to Implement My Features Include:

1. "Dummy" Log-In/Out Credentials:

To implement my Log-In feature I am leveraging the use of cookie storage to simulate having a particular user logged into the site. As stated earlier, this is in no way a secure authentication. I have added hard-coded credentials and used these to perform conditional checks on the input values in each respective field for username and password. If the input fields contain the right values when the login button is clicked, I call to another function to change the presentation of the website for Admin access. This is done by changing the display properties of certain elements indicated with the ".adminOnly" class. If the wrong details are entered, an Alert is presented indicating incorrect log in details were entered and the site remains the same. On initial load, admin only elements are set to display "none" by default unless cookies indicate an admin is logged in.

```
16  window.addEventListener("load", checkForAdmin);
```

```
19  /* Pretend Log In/Admin Access  */
20
21  //Pretend log in credentials (unsecure, but just for a display of functionality on github pages site)
22  const ADMIN = "ItsJustJarvis";
23  const CREDENTIALS = "admin";
24
25  //Get ahold of login form and buttons, add event listeners for log in and log out response
26  let loginForm = document.getElementById("adminLogIn");
27  let loginButton = document.getElementById("submitLogIn");
28  loginButton.addEventListener("click", loggingIn);
29  let logoutButton = document.getElementById("logOut");
30  logoutButton.addEventListener("click", loggingOut);
31
32  //When logging in, get the input form values and compare to admin credentials.
33  //If successful display admin features and save username as cookie for persistence, else alert incorrect entry.
34  function loggingIn(event) {
35    event.preventDefault();
36
37    let username = loginForm.username.value;
38    let password = loginForm.password.value;
39
40    if (username === ADMIN && password === CREDENTIALS) {
41      alert("Welcome Back Reeve!");
42      createCookie("userName", username, 1, "/");
43      adminAccess();
44    } else {
45      alert("Incorrect Log-In Details");
46    }
47  }
```

```
49  //Logging out will hide admin features and remove cookie for admin username, then reload the page
50  function loggingOut(event) {
51    event.preventDefault();
52    hideAdmin();
53    createCookie("userName", "", -1, "/");
54    location.reload();
55    return false;
56  }
57
58  //Check for admin credentials from cookies and display content accordingly
59  function checkForAdmin() {
60    let currentUser = getCookie("userName");
61    if (currentUser == ADMIN) {
62      adminAccess();
63    } else hideAdmin();
64  }
65
66  //Display admin only content (currently only implemented in calendar entry)
67  function adminAccess() {
68    loginForm.style.display = "none";
69    let adminViewList = document.querySelectorAll(".adminOnly");
70    for (let i = 0; i < adminViewList.length; i++)
71      adminViewList[i].style.display = "table-row";
72  }
73
74  //Hide admin only content
75  function hideAdmin() {
76    let adminViewList = document.querySelectorAll(".adminOnly");
77    for (let i = 0; i < adminViewList.length; i++)
78      adminViewList[i].style.display = "none";
79  }
```

2. Cookie/localStorage Persistence:

To implement cookie storage I have used the functions presented to us in the course materials. For my website I have decided to use cookies for the current theme of the site, as well as the userName if the admin is logged in. These values will allow presentation to remain persistent while navigating the website.

```
154 //Creating cookies function pulled from course materials
155 function createCookie(name, value, days, path, domain, secure) {
156   let expires;
157   if (days) {
158     let date = new Date();
159     date.setTime(date.getTime() + days * 24 * 60 * 60 * 1000);
160     expires = date.toGMTString();
161   } else expires = "";
162   let cookieString = name + "=" + escape(value);
163   if (expires) cookieString += "; expires=" + expires;
164   if (path) cookieString += "; path=" + escape(path);
165   if (domain) cookieString += "; domain=" + escape(domain);
166   if (secure) cookieString += "; secure";
167   document.cookie = cookieString;
168 }
169
170 //Getting cookie function pulled from course materials
171 function getCookie(name) {
172   var nameEquals = name + "=";
173   var crumbs = document.cookie.split(";");
174   for (var i = 0; i < crumbs.length; i++) {
175     var crumb = crumbs[i].trim();
176     if (crumb.indexOf(nameEquals) == 0) {
177       return unescape(crumb.substring(nameEquals.length, crumb.length));
178     }
179   }
180   return null;
181 }
```

Additionally, I have implemented localStorage for the scheduling functionality of my calendar. I am leveraging JSON serialization to store and update an array of Date objects and present the calendar with current busy dates. To do this I have created functions that act as a getter/setter as well as updating the running list through array concatenation:

```
209  //Function to save the busy dates into the local storage and maintain running list using JSON
210  function saveBusyDates(array) {
211    localStorage.setItem("busyDatesArray", JSON.stringify(array));
212  }
213
214  //Function to update/concatenate busy date arrays as necessary to maintain running list
215  //and then save to local storage using JSON
216  function updateBusyDates(array) {
217    let updatedBusyDatesArray = getBusyDates();
218    let newArray = updatedBusyDatesArray.concat(array);
219    saveBusyDates(newArray);
220  }
221
222  //Function to get all current busy dates from local storage and parse the JSON array
223  function getBusyDates() {
224    let retrievedData = localStorage.getItem("busyDatesArray");
225    if (!retrievedData) {
226      saveBusyDates(busyDates);
227      retrievedData = localStorage.getItem("busyDatesArray");
228    }
229    let retrievedBusyDatesArray = JSON.parse(retrievedData);
230    return retrievedBusyDatesArray;
231  }
```

3. Theme Selection:

To implement theme selection I am performing an ".indexOf" search through the document.styleSheets list to get a hold of the two I am concerned with. I am then toggling their disabled attribute and the button text in response to a button click, this saves the current theme into a cookie and changes the elements of the page accordingly.

```
83   //Declare variables for different themes
84   let themeBW, themeColour;
85
86   //Get ahold of theme button, and add event listener
87   let themeButton = document.getElementById("themeButton");
88   themeButton.addEventListener("click", changeMode);
89
90   //Theme enabler refactored and implemented much cleaner than in previous assignment
91   //Get cookie for current theme and set accordingly by accessing stylesheet list and
92   //attaching variables to govern each theme.
93   //Set the proper button text, and photos (colour or black and white)
94   function themeEnabler() {
95     let themeReference = getCookie("theme");
96     let allStyleSheets = document.styleSheets;
97
98     for (let i = 0; i < allStyleSheets.length; i++) {
99       if (allStyleSheets[i].href.indexOf("css_blackAndWhiteStyles.css") != -1)
100        themeBW = allStyleSheets[i];
101      if (allStyleSheets[i].href.indexOf("css_colourStyles.css") != -1)
102        themeColour = allStyleSheets[i];
103    }
104
105    if (themeReference == null) {
106      themeBW.disabled = true;
107    } else if (themeReference == themeBW.href) {
108      themeColour.disabled = true;
109    } else {
110      themeBW.disabled = true;
111    }
112    setThemeButtonText();
113    setThemePhotos();
114  }
```

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

To achieve a black and white theme across my website I needed to store an alternative folder with the same images but grayscale using Photoshop and then change the src path using JavaScript.

```
116  //Function used to adjust the photos used based on theme by changing their src path
117  function setThemePhotos() {
118    let allPhotos = document.querySelectorAll("img");
119    let photoSourceFolder;
120
121    if (themeBW.disabled) photoSourceFolder = "photos";
122    if (themeColour.disabled) photoSourceFolder = "bwPhotos";
123
124    for (let i = 0; i < allPhotos.length; i++) {
125      let path = allPhotos[i].src;
126      let pathArray = path.split("/");
127      let file = pathArray[pathArray.length - 1];
128      let newSRC = `${photoSourceFolder}/${file}`;
129      allPhotos[i].setAttribute("src", newSRC);
130    }
131  }
132
133  //Function to set the theme button text according to current theme
134  function setThemeButtonText() {
135    if (themeBW.disabled) {
136      themeButton.innerHTML = "Black/White Mode";
137      createCookie("theme", themeColour.href);
138    }
139    if (themeColour.disabled) {
140      themeButton.innerHTML = "Colour Mode";
141      createCookie("theme", themeBW.href);
142    }
143  }
144
145  //Event handler to change the theme on theme button click
146  function changeMode() {
147    themeBW.disabled = !themeBW.disabled;
148    themeColour.disabled = !themeBW.disabled;
149    setThemeButtonText();
150    setThemePhotos();
151  }
```

4. Gallery Modal Images:

To implement my gallery image modals I used some CSS zoom animations and adjusted styles in response to JavaScript function calls. I am using a “querySelectorAll” to obtain the gallery images and add event listeners to each of them. These call the modal image to display as a block on top of the rest of the page contents using z-index. The alt text from the image is used to present a caption, and a close button is shown in the top right of the screen.

```
12  .modalImage,
13  #caption {
14    animation-name: popup;
15    animation-duration: 1s;
16  }
17  .closeModal {
18    transition: 0.5s;
19  }
20  @keyframes popup {
21    from {
22      transform: scale(0);
23    }
24    to {
25      transform: scale(1);
26    }
27  }
```

```
//Get ahold of modal container and contents
let modal = document.getElementById("imagePopup");
let modalImg = document.getElementById("shownImage");
let captionText = document.getElementById("caption");

//Get ahold of all gallery images and set event listeners on them all
let galleryImages = document.querySelectorAll(".galleryImage");
for (let i = 0; i < galleryImages.length; i++) {
  galleryImages[i].addEventListener("click", displayModalImage);
}

//Get ahold of modal closer and add event listener
let span = document.getElementsByClassName("closeModal")[0];
span.addEventListener("click", closeModalImage);

//Display selected modal image on top of screen with caption comprised of the alt text
function displayModalImage() {
  modal.style.display = "block";
  modalImg.src = this.src;
  captionText.innerHTML = this.alt;
}

//Close the modal image by setting its display attribute to none
function closeModalImage() {
  modal.style.display = "none";
}
```

5. DOM Generated Calendar and Scheduling:

To implement my calendar, I am creating the table dynamically each time the page is loaded, or month is changed using DOM functions and accurate date information gathered with Date objects. I am displaying the calendar on the current month and year on load and have declared some arrays and variables to track the data I need. I get a hold of the elements that are present in my HTML code, and create the rest using JavaScript. The table contents are then produced through various functions I have implemented. The dates are filled in day by day, week by week within the bounds of the MAX_DAYS and MAX_WEEKS constants I have declared.

```
"use strict";

//Declare names of months to use for presentation in accordance with index values
let monthsList=["January","February","March","April","May","June","July","August","September"
,"October","November","December",];

//Initialize a date object, and other variables to represent the current day/month/year.
let today = new Date();
today.setHours(0, 0, 0, 0);
let currentMonth = today.getMonth();
let currentYear = today.getFullYear();
const MAX_WEEKS = 6;
const MAX_DAYS = 7;

//Initialize an array to contain busy dates, set initial contents to include today (no same day bookings
//without special request)
let busyDates = [today];

//On loading page, display calendar with current month/year
window.addEventListener("load", displayCalendar(currentMonth, currentYear));
```

```
//Get ahold of the calendar navigation button for the previous month and set event listener
let previousMonthButton = document.getElementById("previousMonth");
previousMonthButton.addEventListener("click", showPreviousMonth);

//Get ahold of the calendar navigation button for the next month and set event listener
let nextMonthButton = document.getElementById("nextMonth");
nextMonthButton.addEventListener("click", showNextMonth);
```

```
54  //Create button to submit new busy dates, set attributes and event listener
55  let setBusyDatesButton = document.createElement("input");
56  setBusyDatesButton.setAttribute("type", "submit");
57  setBusyDatesButton.setAttribute("value", "Set Busy Dates");
58  setBusyDatesButton.addEventListener("click", addBusyDates);
59
60  //Create button to remove busy dates from schedule, set attributes and event listener
61  let removeBusyDatesButton = document.createElement("input");
62  removeBusyDatesButton.setAttribute("type", "submit");
63  removeBusyDatesButton.setAttribute("value", "Remove Busy Dates");
64  removeBusyDatesButton.addEventListener("click", removeBusyDates);
65
66  //Get ahold of container for scheduling form, append previously created submit buttons to it
67  let scheduleEntryRow = document.getElementById("blackoutEntry");
68  scheduleEntryRow.appendChild(setBusyDatesButton);
69  scheduleEntryRow.appendChild(removeBusyDatesButton);
70
71  //Get ahold of request form date entry to allow cross-calendar response, set event listener to respond to changes
72  let requestDate = document.getElementById("date");
73  requestDate.addEventListener("change", showSelectedMonth);
74
75  //Display calendar, initated on load among other places. Calls functions to set month, date, and busy dates in schedule
76  function displayCalendar(month, year) {
77    setCalendarMonth(month, year);
78    setCalendarDays(month, year);
79    setAllBusyDates();
80  }
81
82  //Set the heading of the calendar to represent the current month and year, filling with dynamic information
83  function setCalendarMonth(month, year) {
84    let calendarHeading = document.getElementById("monthYear");
85    calendarHeading.innerHTML = "";
86    let text = document.createTextNode(`${monthsList[month]} ${year}`);
87    calendarHeading.appendChild(text);
88  }
```

```
90  //Dynamically create calendar table and set the days of the calendar month, removing and refreshing the values
91  //on a month change. Implemented using ideas from course assignments and techniques found in research.
92  //Adjusted to fit my needs and react to busy date scheduling functionality.
93  function setCalendarDays(month, year) {
94    let newCalendar = new Date(year, month);
95    let weekdayStart = newCalendar.getDay();
96    let calendarBody = document.getElementById("calendarBody");
97    let priorCellsUsed = 1;
98    let dayCount = 1;
99
100   calendarBody.innerHTML = "";
101
102   for (let row = 0; row < MAX_WEEKS; row++) {
103     let week = document.createElement("tr");
104     week.classList.add("weeks");
105     for (let cell = 0; cell < MAX_DAYS; cell++) {
106       if (row == 0 && cell < weekdayStart) {
107         let day = document.createElement("td");
108         let dayLabel = document.createTextNode(" ");
109         day.appendChild(dayLabel);
110         day.classList.add("lastMonth");
111         week.appendChild(day);
112         priorCellsUsed++;
113       } else {
114         let day = document.createElement("td");
115         day.classList.add("daysOfMonth");
116         let dayLabel = document.createTextNode(dayCount);
117         day.appendChild(dayLabel);
118         week.appendChild(day);
119         dayCount++;
120       }
121     }
122     calendarBody.appendChild(week);
123   }
124   removeUnwantedCells(priorCellsUsed);
125 }
```

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

When I have produced the full calendar, I fill the remains of the current week with a solid blue td element and set any td elements past the last row of the month to display “none” to avoid any unnecessary cells and maintain a nice presentation.

```
//Remove unwanted cells from calendar display to keep desired appearance.
function removeUnwantedCells(priorCellsUsed) {
  let allDays = document.querySelectorAll(".daysOfMonth");
  let trailingCells =
    allDays.length - numberOfDaysInMonth(currentMonth, currentYear);
  let difference = trailingCells % 7;
  let cellsToKeep = difference + numberOfDaysInMonth(currentMonth, currentYear);
  for (let i = 0; i < allDays.length; i++) {
    if (Number.parseInt(allDays[i].innerHTML) > cellsToKeep) {
      allDays[i].style.display = "none";
    } else if (
      Number.parseInt(allDays[i].innerHTML) >
      numberOfDaysInMonth(currentMonth, currentYear)
    ) {
      allDays[i].innerHTML = " ";
      allDays[i].classList.add("lastMonth");
    }
  }
}
```

The calendar also responds to navigation buttons I have added to go forward and backwards in months. This works by adjusting the values of the current month and year variables I have declared. Additionally, I have the calendar set up to respond to a date change in the Request Form underneath it, showing the month selected in the correct year as well as highlighting the day chosen to reference against my scheduled busy days.

```
//Function to get an accurate number of days in month by using technique found in research
// on stack overflow. Setting the date to larger value than possible resets to the last day.
function numberOfDaysInMonth(month, year) {
  return 32 - new Date(year, month, 32).getDate();
}

//Event handler for previous month button press, adjusting the value of the current month and
//reacting to possible year change then displaying new calendar
function showPreviousMonth() {
  currentMonth--;
  if (currentMonth < 0) {
    currentMonth = 11;
    currentYear--;
  }
  displayCalendar(currentMonth, currentYear);
}

//Event handler for next month button press, adjusting the value of the current month and
//reacting to possible year change then displaying new calendar
function showNextMonth() {
  currentMonth++;
  if (currentMonth > 11) {
    currentMonth = 0;
    currentYear++;
  }
  displayCalendar(currentMonth, currentYear);
}

//Cross-calendar reaction to request-form date selection, showing the month and year selected
//as well as highlighting the day to check availability
function showSelectedMonth() {
  let daySelected = requestDate.valueAsDate.getUTCDate();
  let monthSelected = requestDate.valueAsDate.getMonth();
  let yearSelected = requestDate.valueAsDate.getFullYear();
  currentMonth = monthSelected;
  currentYear = yearSelected;
  displayCalendar(currentMonth, currentYear);
  setSelectedDay(daySelected);
  requestDate.value = "";
}
```

```
//Function to highlight the date selected in the request-form
function setSelectedDay(selectedDay) {
  let allDays = document.querySelectorAll(".daysOfMonth");
  for (let i = 0; i < allDays.length; i++) {
    if (selectedDay == allDays[i].innerHTML)
      allDays[i].classList.add("selected");
  }
}

//Event handler to add busy dates to the calendar schedule, blocking them out and adding them to
// the array/local storage to remain persistent. This implementation using local storage is purely
//to display functionality as github only allows static websites. To be adjusted when website is
//hosted elsewhere.
function addBusyDates(event) {
  event.preventDefault();

  let startEntry = document.getElementById("startDate");
  let startBusyDate = new Date(startEntry.value + "T00:00");

  let endEntry = document.getElementById("endDate");
  let endBusyDate = new Date(endEntry.value + "T00:00");

  let days = document.querySelectorAll(".daysOfMonth");

  for (let i = 0; i < days.length; i++) {
    let day = Number.parseInt(days[i].innerHTML);
    let dateInQuestion = new Date(currentYear, currentMonth, day);
    if (dateInQuestion >= startBusyDate && dateInQuestion <= endBusyDate) {
      days[i].classList.add("busy");
      busyDates.push(dateInQuestion.toUTCString());
    }
  }
  updateBusyDates(busyDates);
  startEntry.value = "";
  endEntry.value = "";
}
```

And finally, I have added the ability to schedule busy dates into the calendar when I am logged in to the Admin account. There are two date selection boxes that define the bounds of the selection, and the busy dates within the calendar are set using CSS classes in response to a button click. There is a button to add dates, and a button to remove them. This leverages the localStorage and JSON serialization features explained previously and is used as an alternative to database connections to demonstrate functionality options I plan to implement for my website.

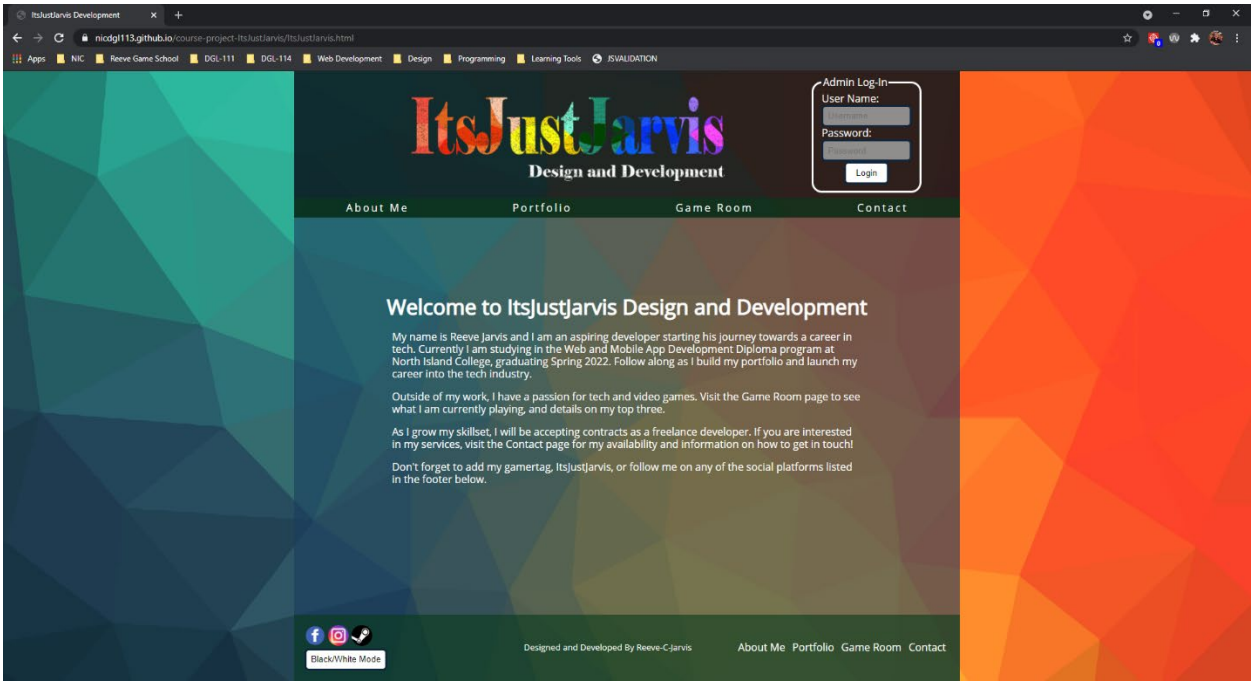
```
//Function to save the busy dates into the local storage and maintain running list using JSON
function saveBusyDates(array) {
  localStorage.setItem("busyDatesArray", JSON.stringify(array));
}

//Function to update/concatenate busy date arrays as necessary to maintain running list
//and then save to local storage using JSON
function updateBusyDates(array) {
  let updatedBusyDatesArray = getBusyDates();
  let newArray = updatedBusyDatesArray.concat(array);
  saveBusyDates(newArray);
}

//Function to get all current busy dates from local storage and parse the JSON array
function getBusyDates() {
  let retrievedData = localStorage.getItem("busyDatesArray");
  if (!retrievedData) {
    saveBusyDates(busyDates);
    retrievedData = localStorage.getItem("busyDatesArray");
  }
  let retrievedBusyDatesArray = JSON.parse(retrievedData);
  return retrievedBusyDatesArray;
}
```

```
//Event handler to remove busy dates from the schedule and return them to their open state,
//updating local storage to maintain current list
function removeBusyDates(event) {
  event.preventDefault();

  let allBusyDates = getBusyDates();

  let startEntry = document.getElementById("startDate");
  let startDateRemoval = new Date(startEntry.value + "T00:00");

  let endEntry = document.getElementById("endDate");
  let endDateRemoval = new Date(endEntry.value + "T00:00");

  let days = document.querySelectorAll(".daysOfMonth");
  for (let i = 0; i < days.length; i++) {
    let day = Number.parseInt(days[i].innerHTML);
    let dateInQuestion = new Date(currentYear, currentMonth, day);
    if (
      dateInQuestion >= startDateRemoval &&
      dateInQuestion <= endDateRemoval
    ) {
      days[i].classList.remove("busy");
      for (let x = 0; x < allBusyDates.length; x++) {
        let dateToCompare = new Date(allBusyDates[x]);
        if (dateInQuestion.getTime() == dateToCompare.getTime()) {
          allBusyDates.splice(x, 1);
          saveBusyDates(allBusyDates);
        }
      }
    }
  }
  startEntry.value = "";
  endEntry.value = "";
}
```

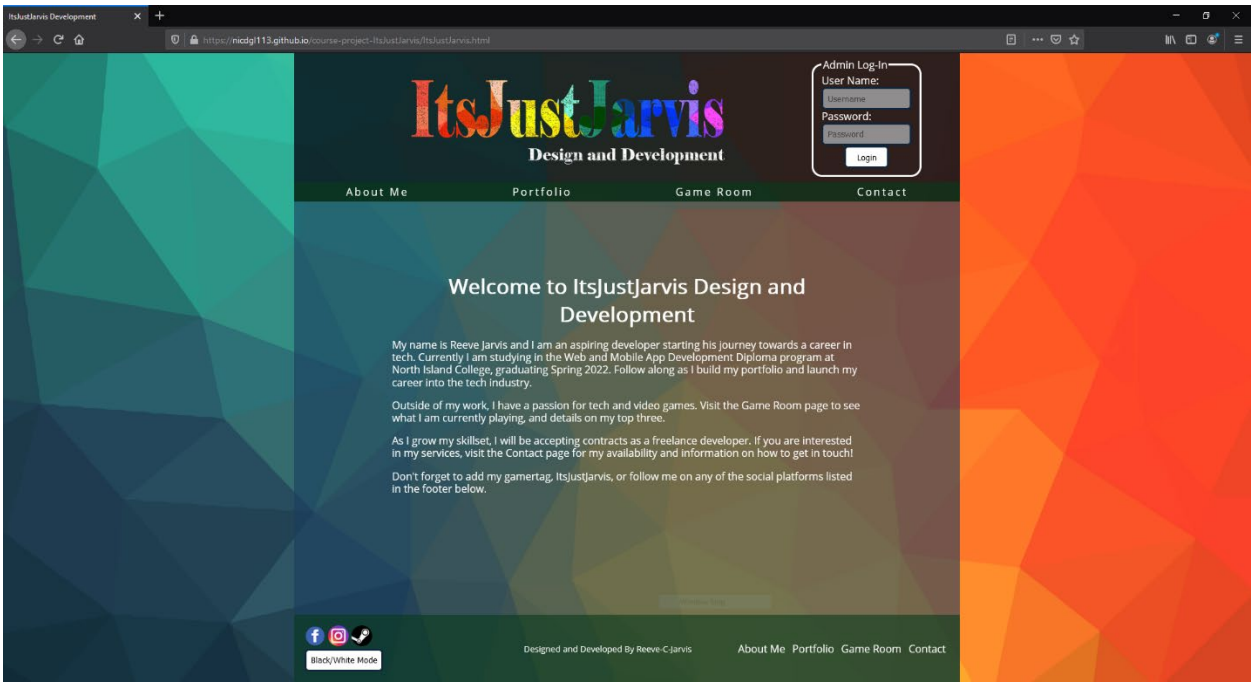
```
//Function to set all the busy dates within presented calendar using persistent list of busy
//dates in local storage
function setAllBusyDates() {
  let allBusyDates = getBusyDates();
  let days = document.querySelectorAll(".daysOfMonth");

  for (let i = 0; i < days.length; i++) {
    let day = Number.parseInt(days[i].innerHTML);
    let date1 = new Date(currentYear, currentMonth, day);
    for (let x = 0; x < allBusyDates.length; x++) {
      let date2 = new Date(allBusyDates[x]);
      if (date1.getTime() == date2.getTime()) days[i].classList.add("busy");
    }
  }
}
```

Section 4: Browser/Wave Testing

Chrome:

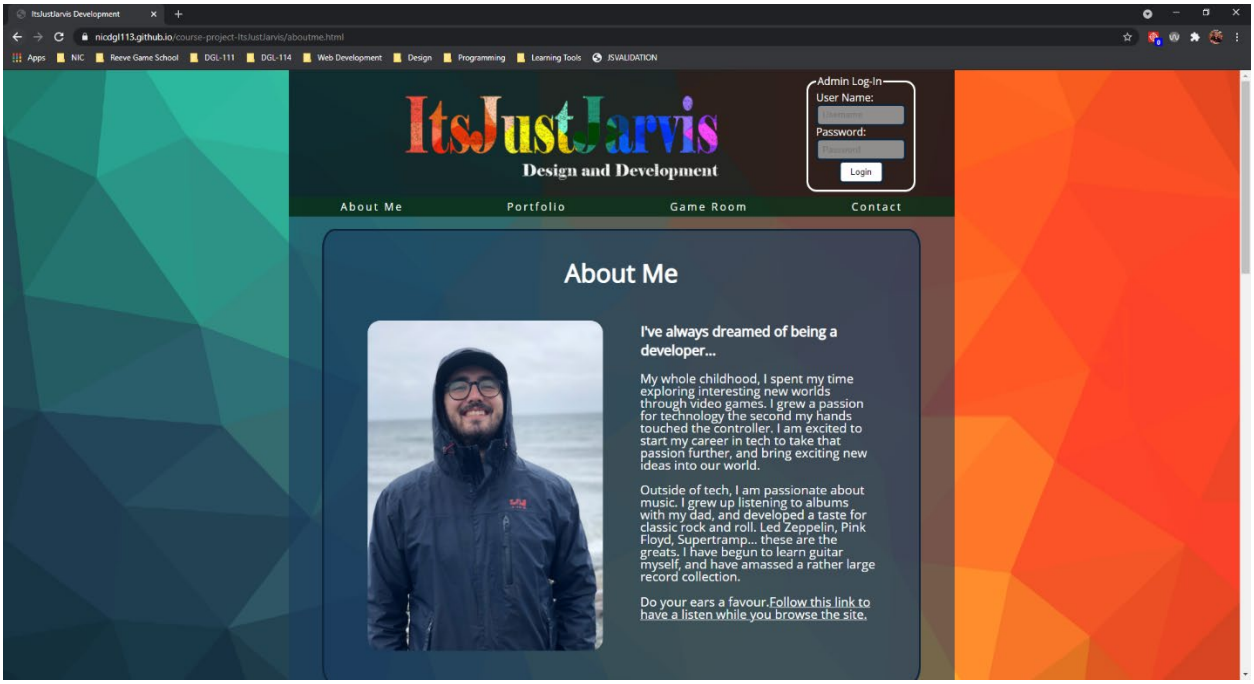


Firefox:

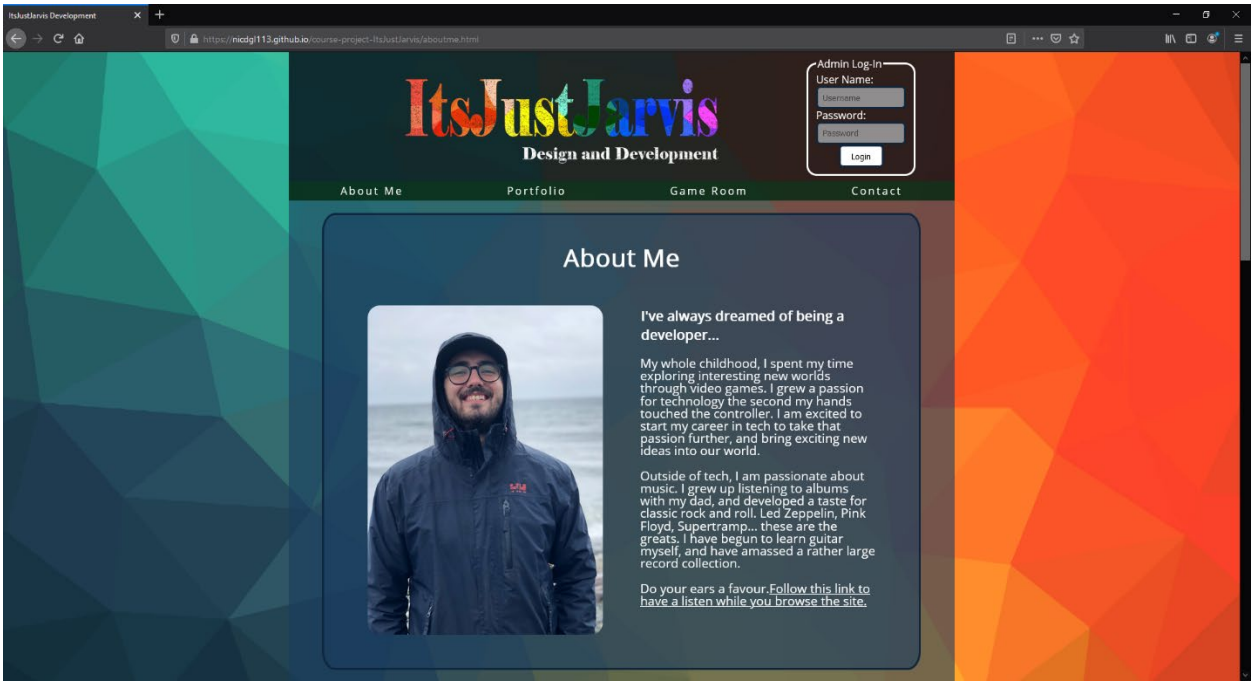


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Chrome:

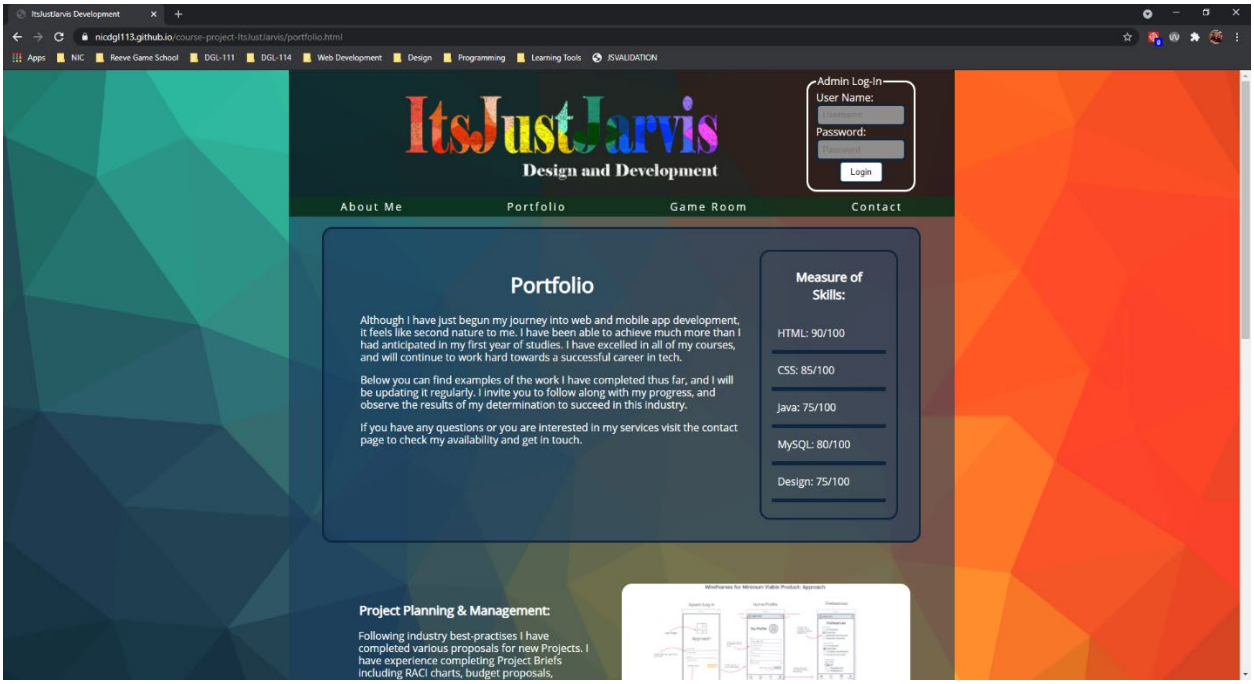


Firefox:

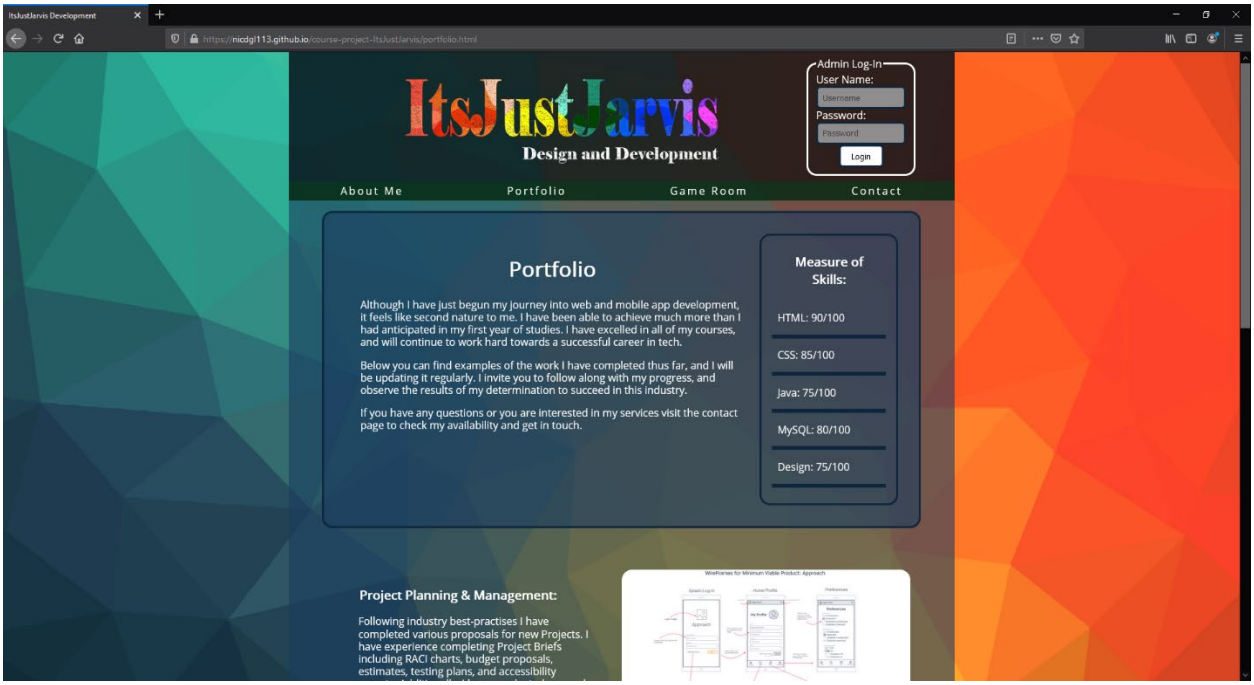


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Chrome:

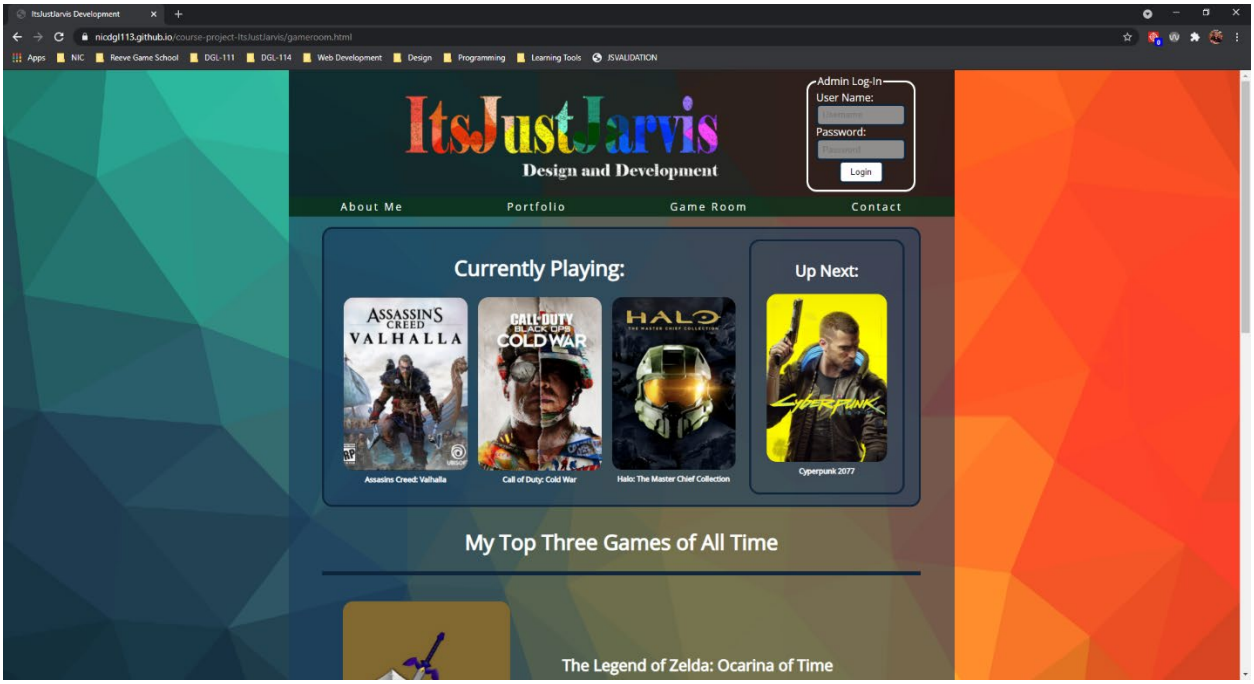


Firefox:

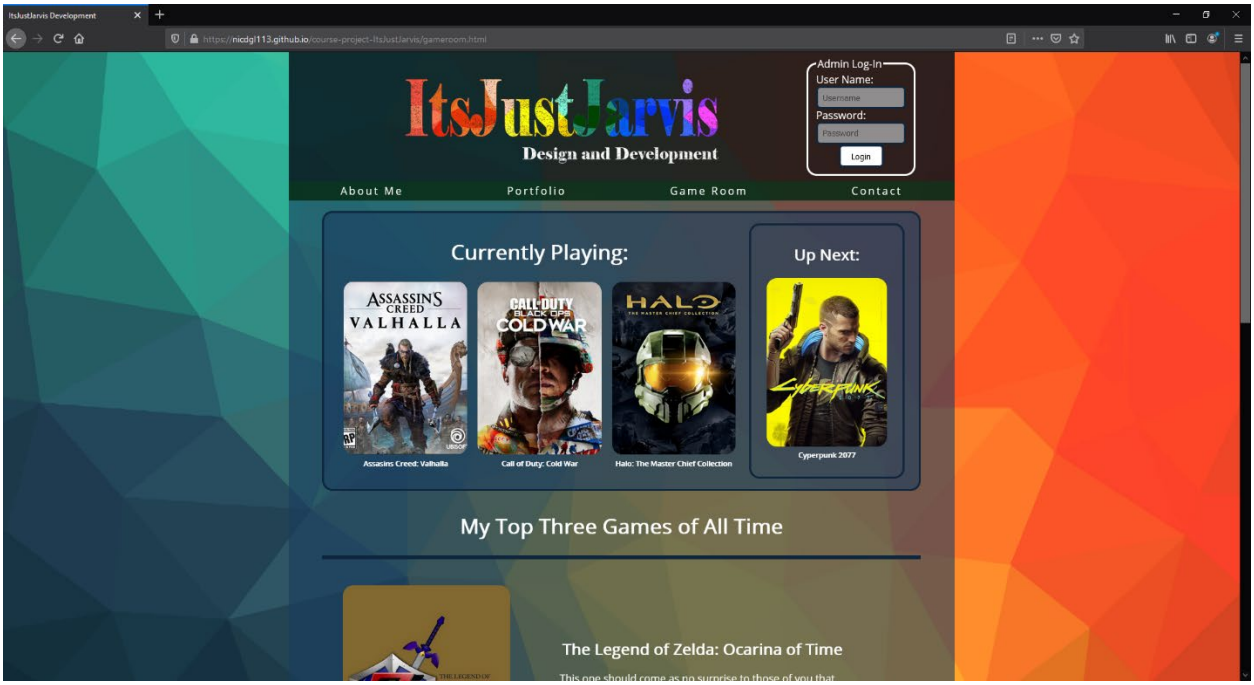


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Chrome:

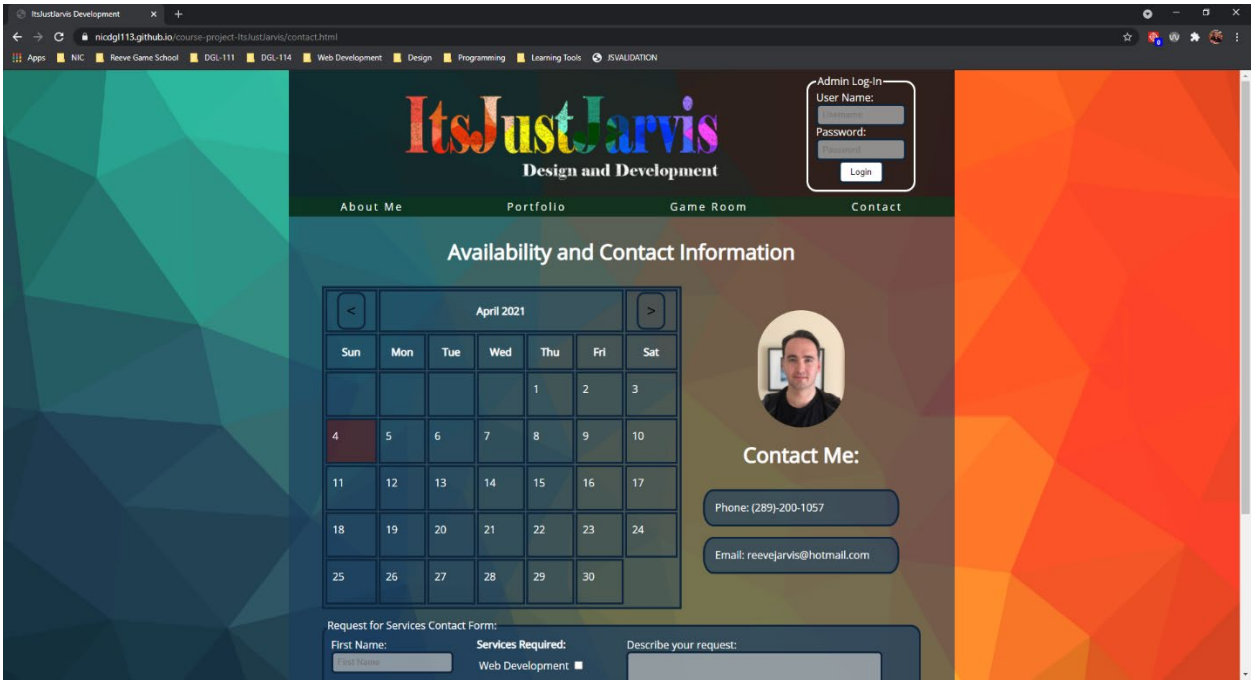


Firefox:

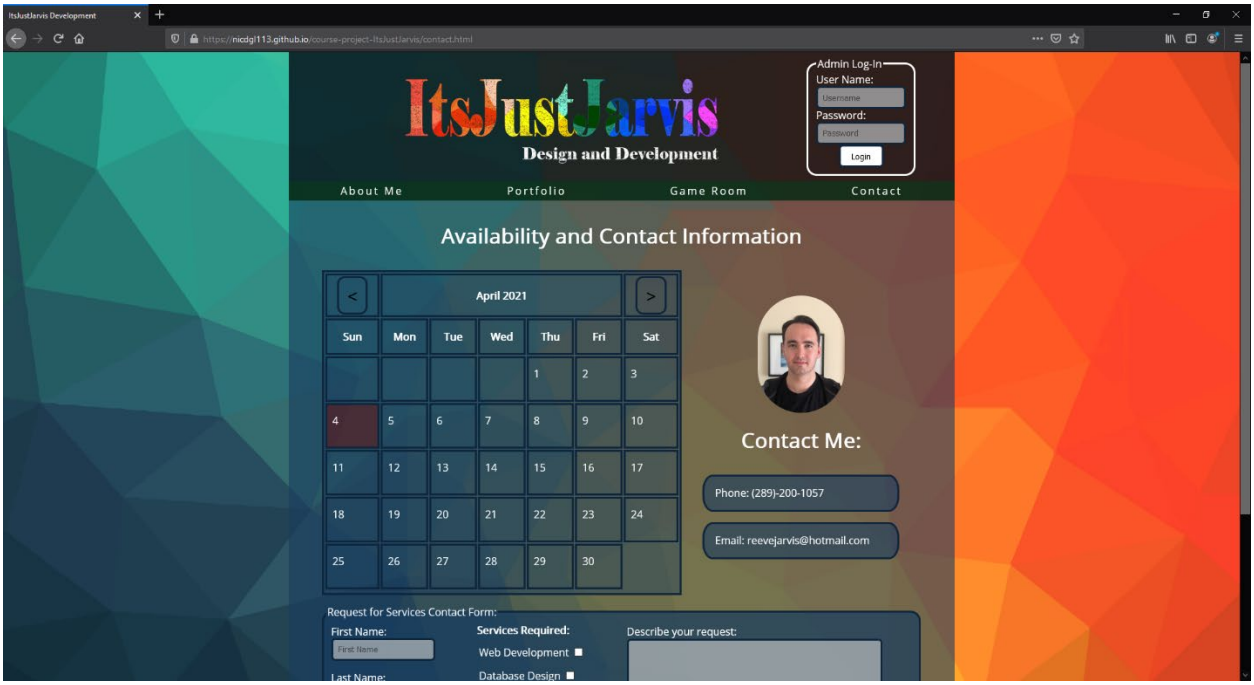


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Chrome:

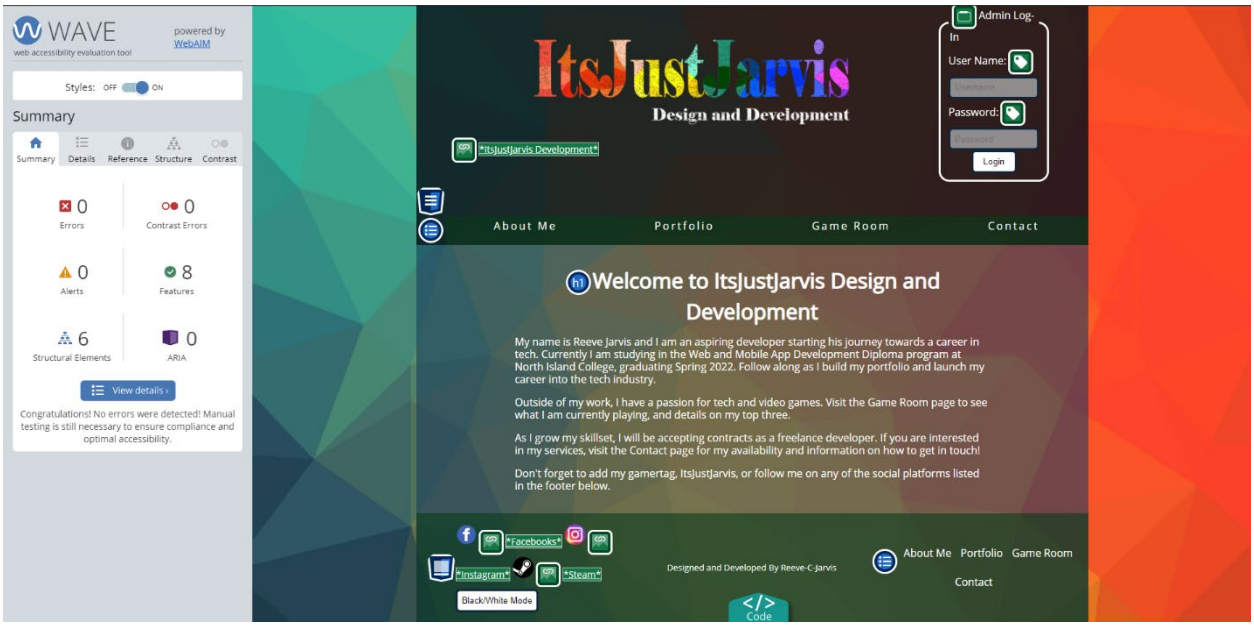


Firefox:

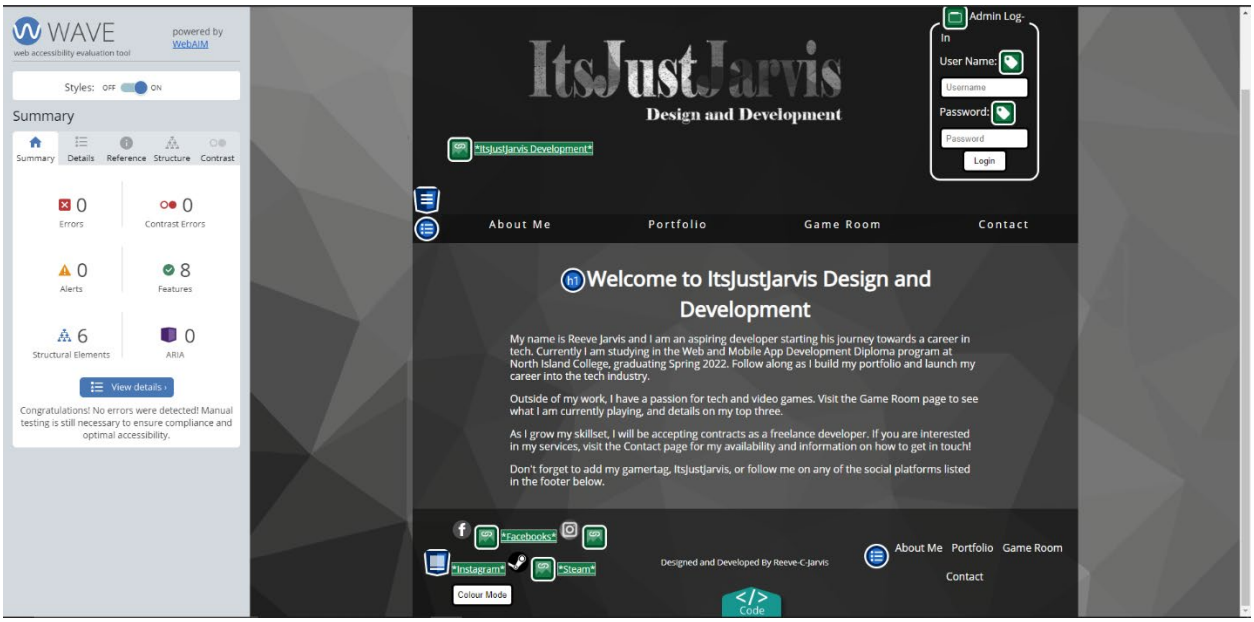


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Colour:

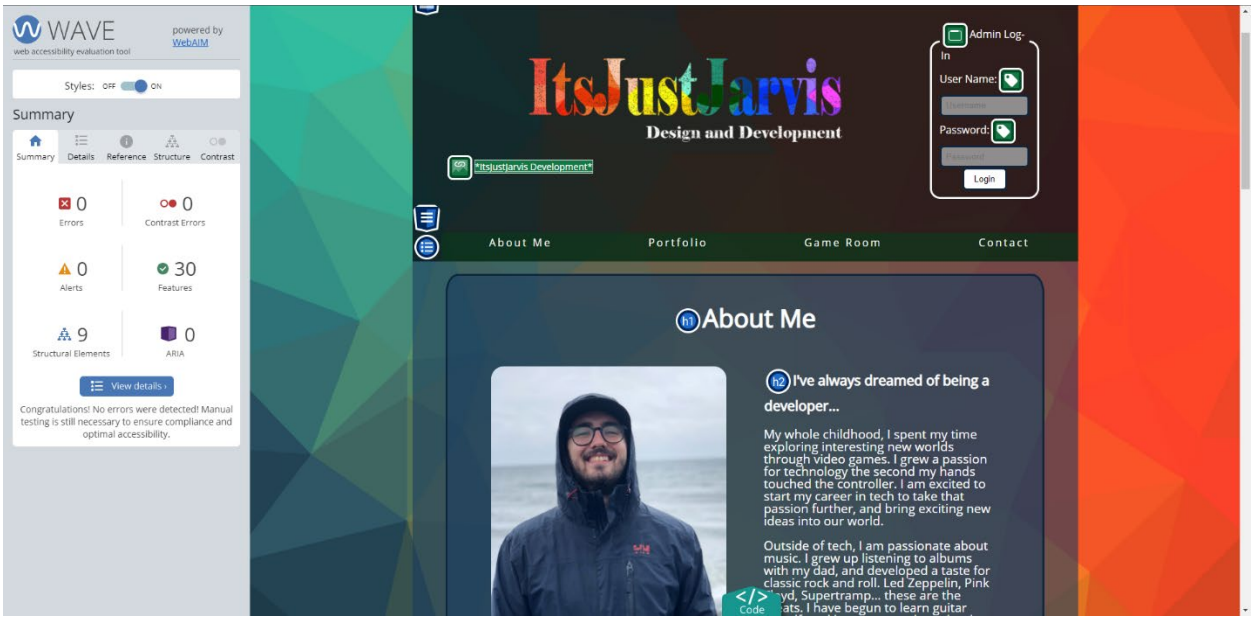


Black/White:

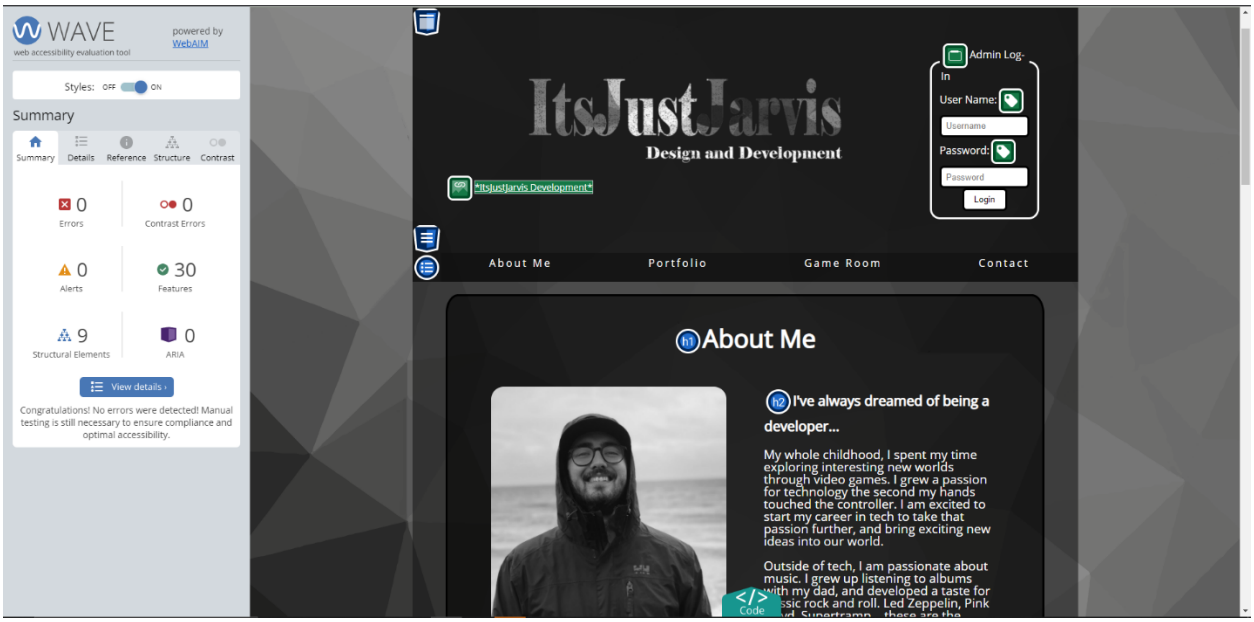


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Colour:

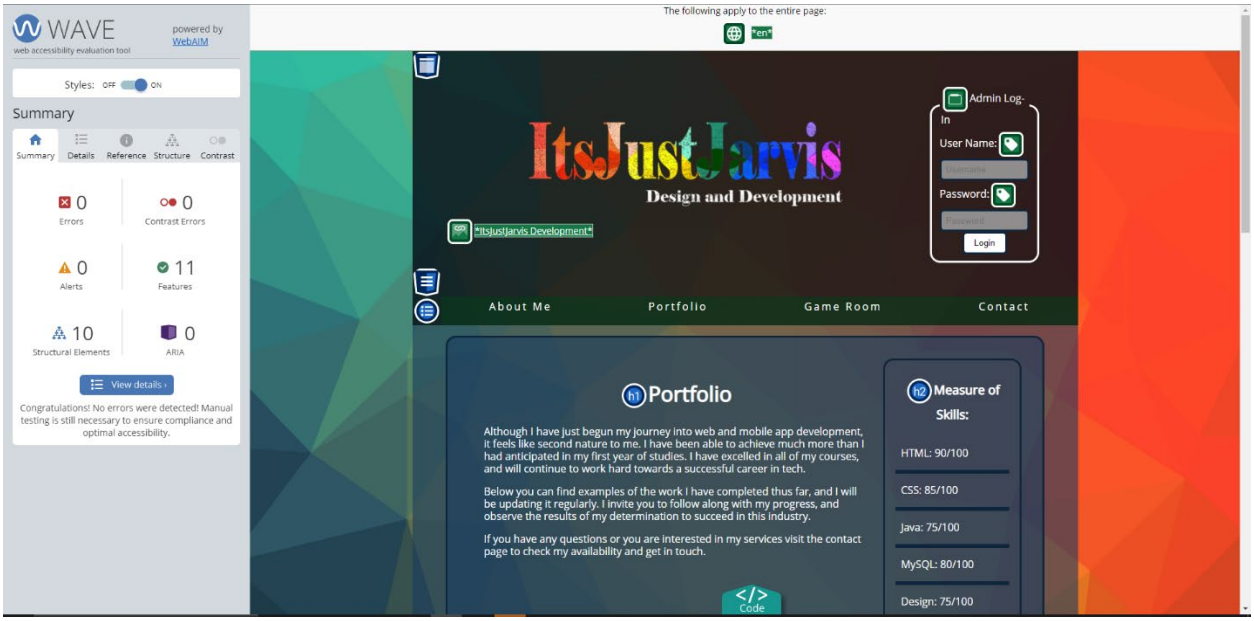


Black/White:

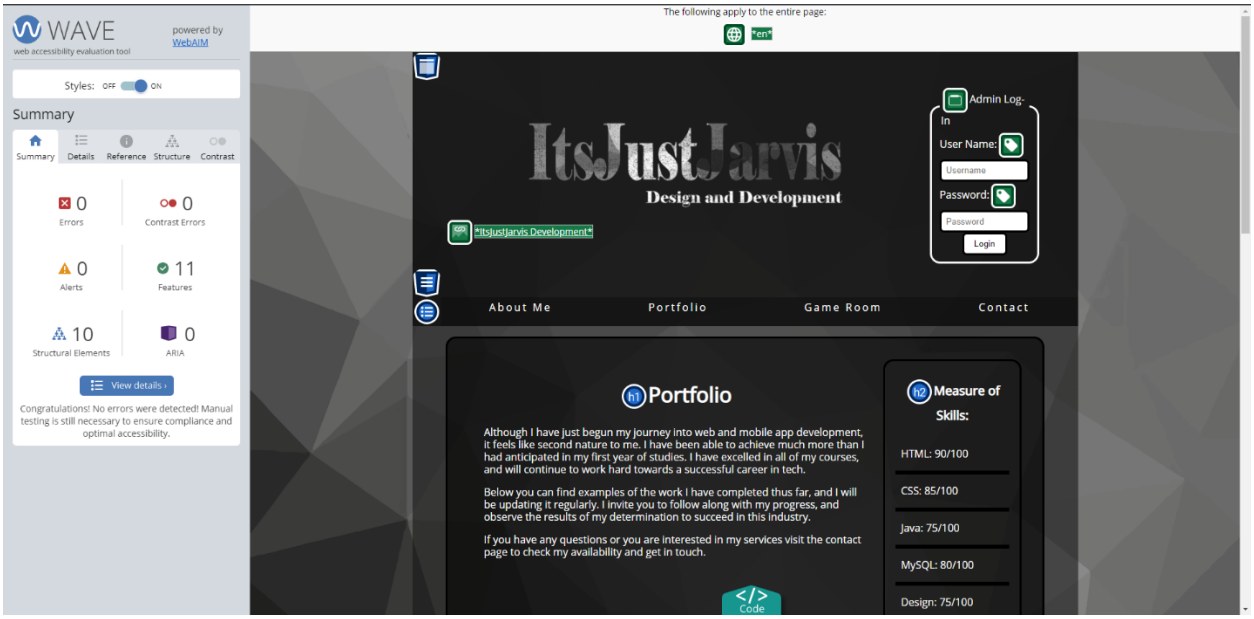


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Colour:

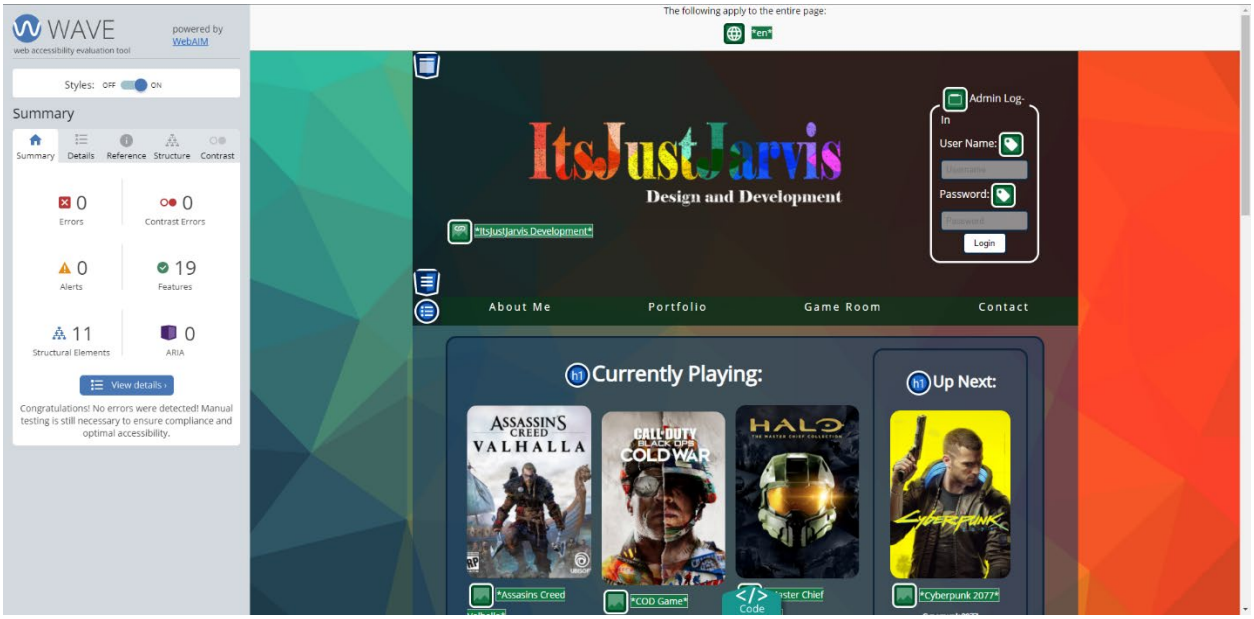


Black/White:

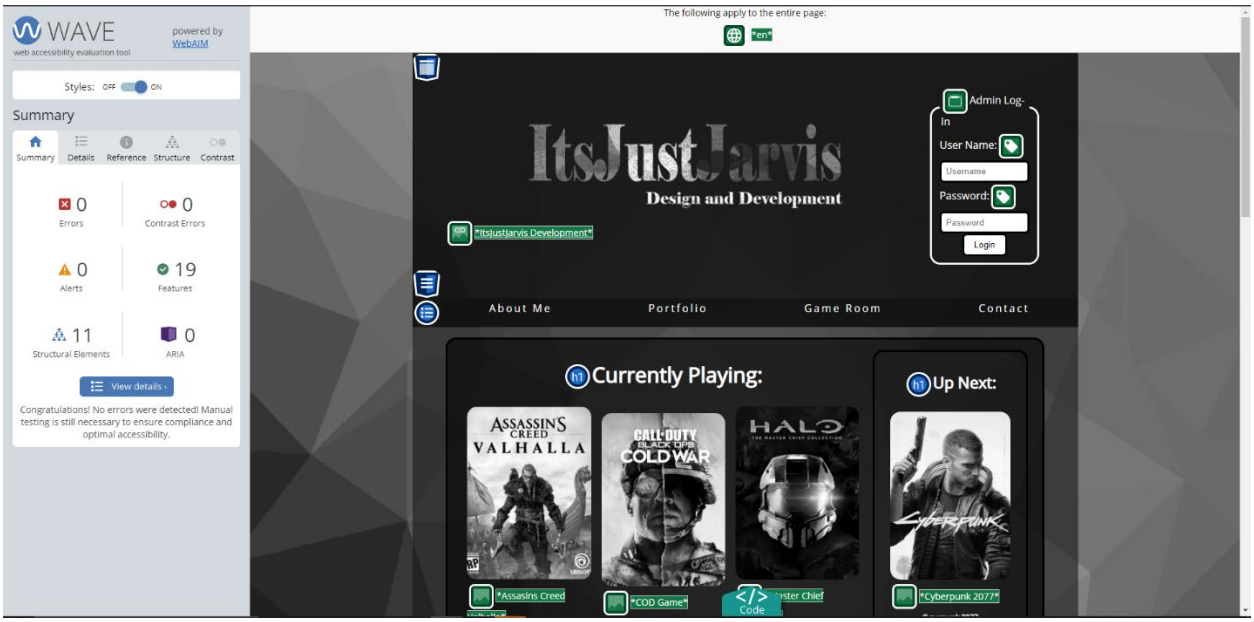


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Colour:

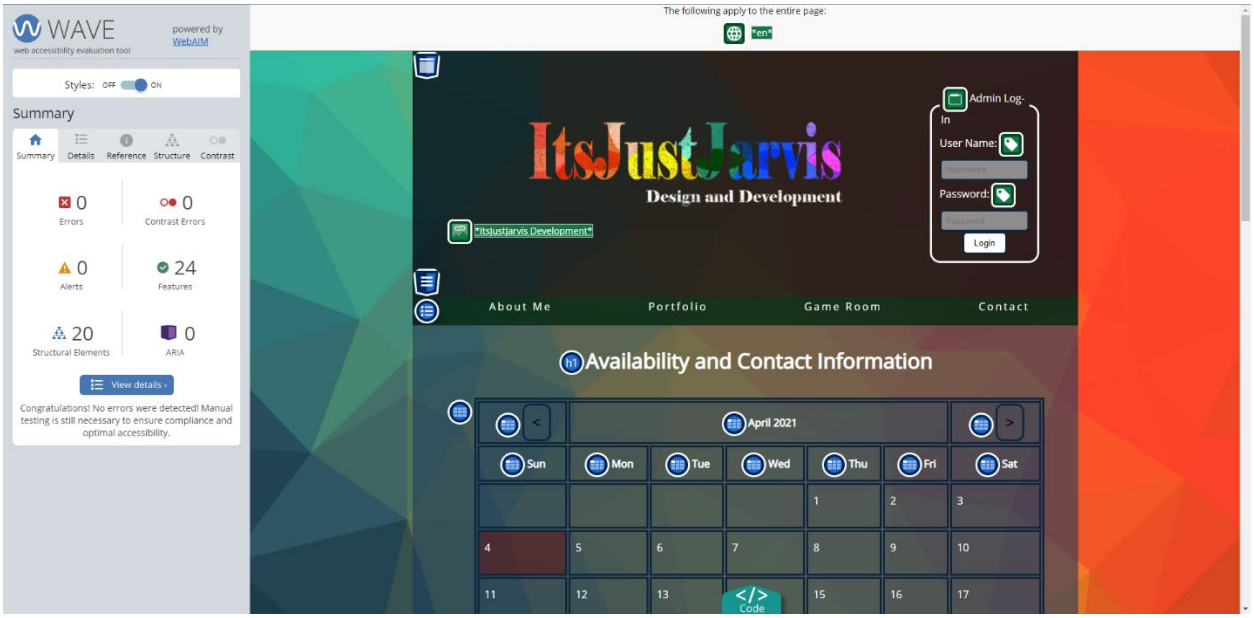


Black/White:

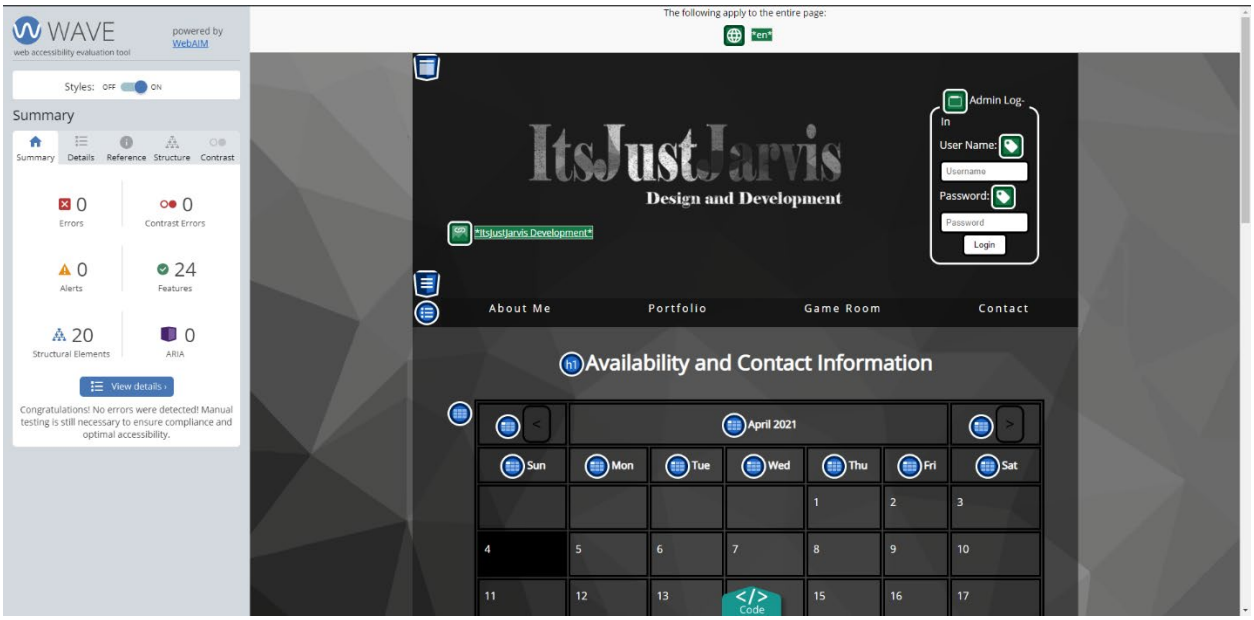


Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Colour:



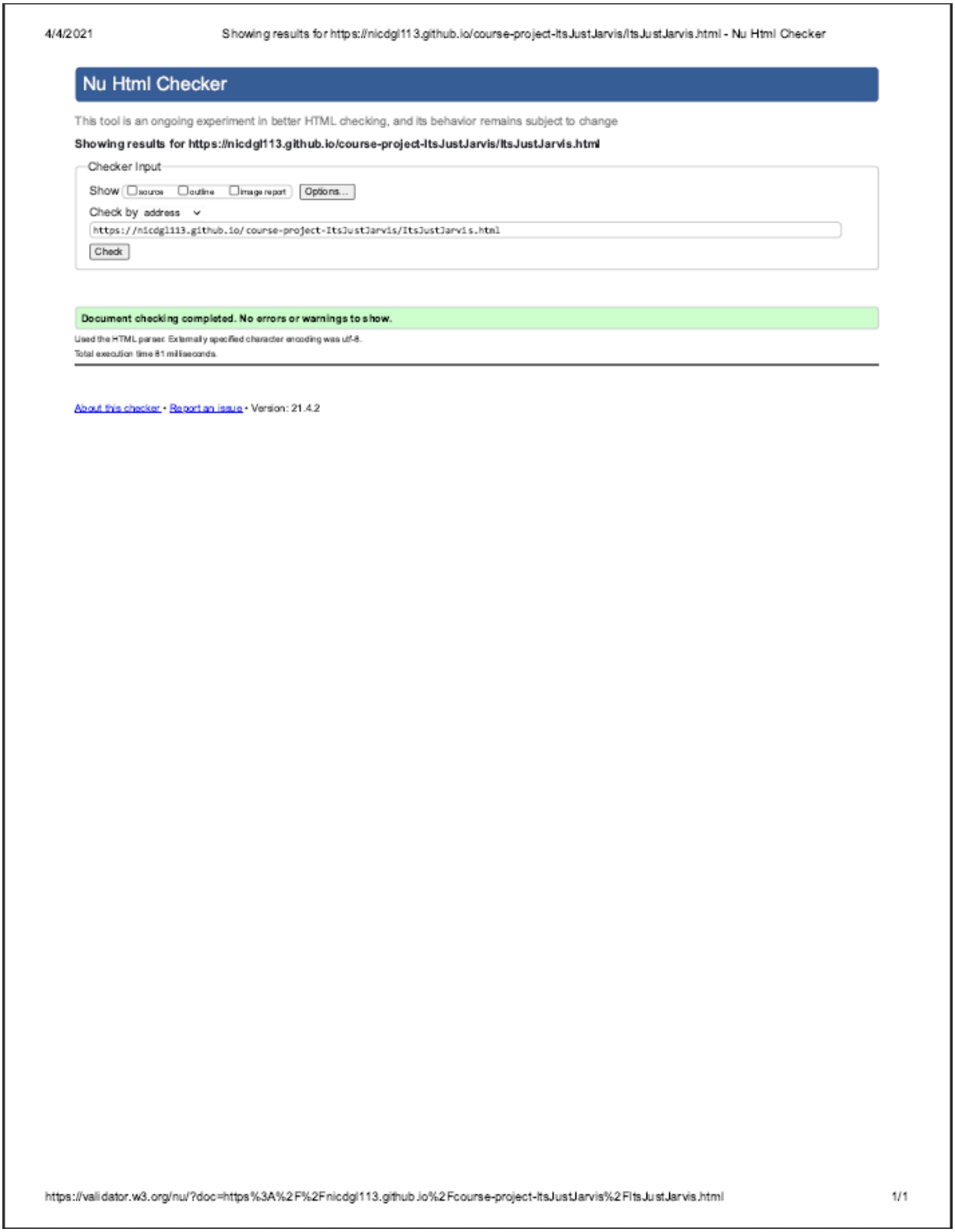
Black/White:



Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Section 5: Validation

Home HTML:



4/4/2021 Showing results for https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html - Nu Html Checker

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Checker Input

Show ☐ source ☐ outline ☐ image report Options...

Check by address

https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Check

Document checking completed. No errors or warnings to show.

Used the HTML parser. Externally specified character encoding was utf-8.

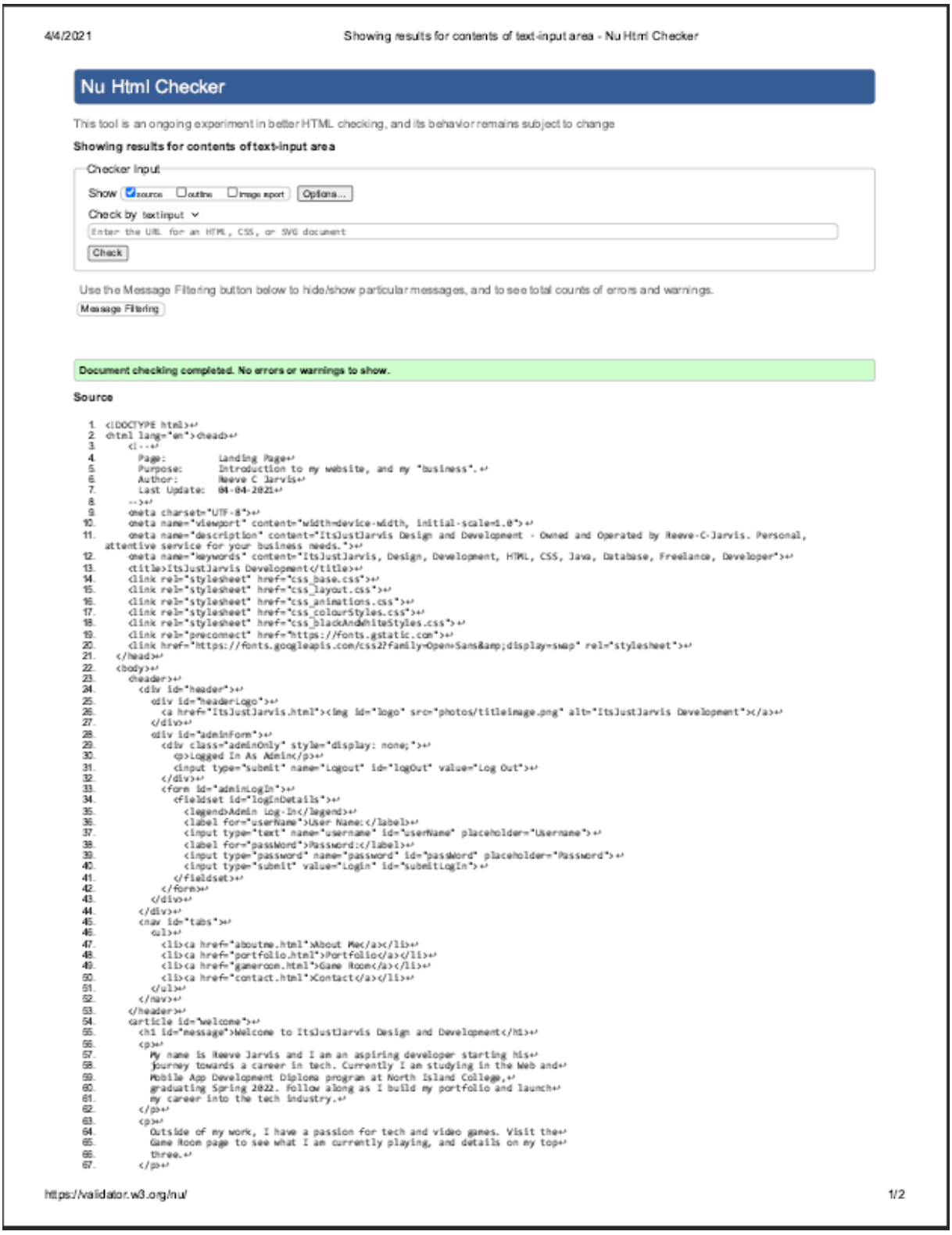
Total execution time 81 milliseconds.

About this checker • Report an issue • Version: 21.4.2

https://validator.w3.org/nu/?doc=https%3A%2F%2Fnicdgl113.github.io%2Fcourse-project-ItsJustJarvis%2FItsJustJarvis.html 1/1

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Home after JS:



4/4/2021 Showing results for contents of text-input area - Nu Html Checker

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show source outline image report Options...

Check by textinput

Enter the URL for an HTML, CSS, or SVG document

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

Document checking completed. No errors or warnings to show.

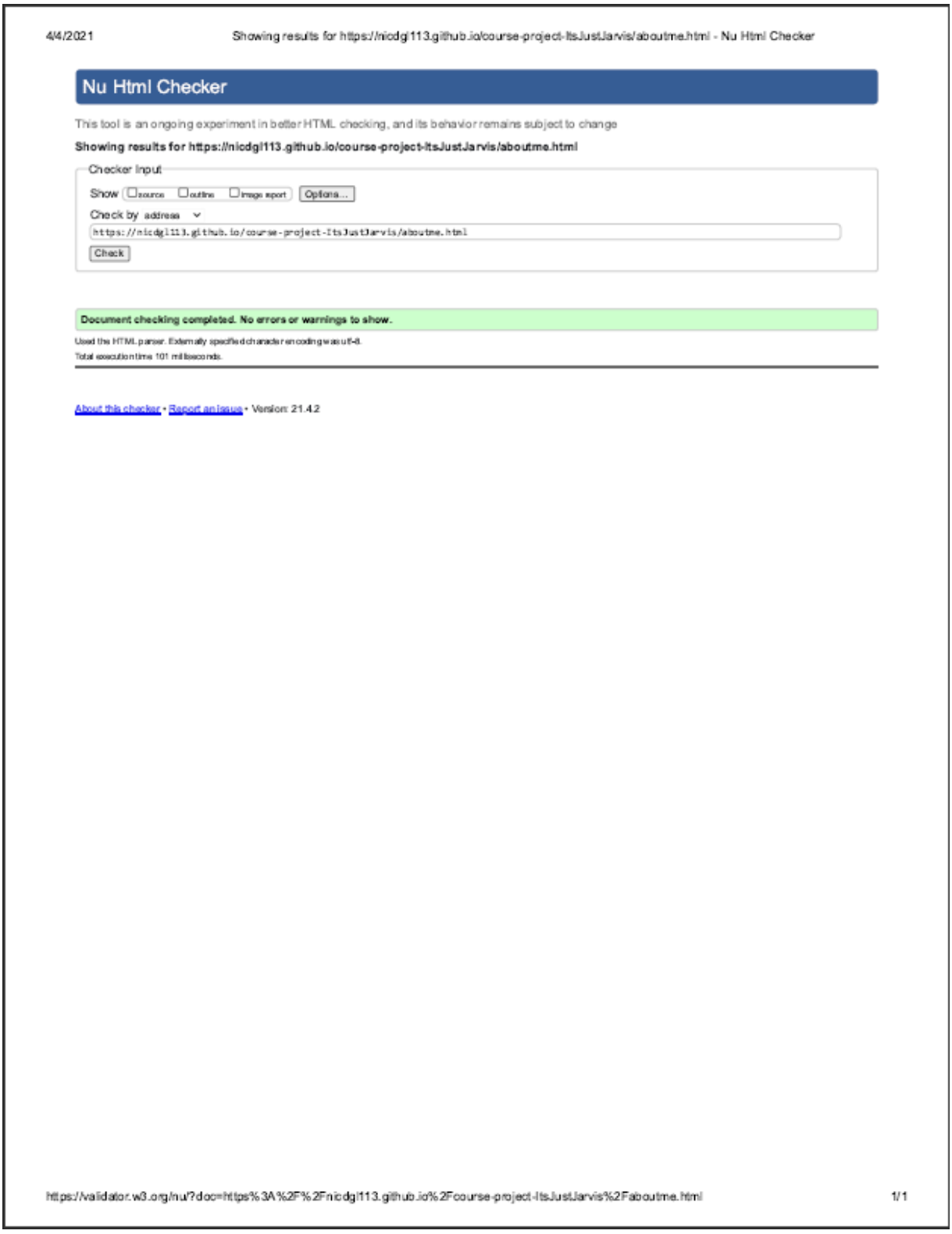
Source

```
<!DOCTYPE html>
<html lang="en"><head>
    <!--
      Page:         Landing Page
      Purpose:      Introduction to my website, and my "business".
      Author:       Reeve C Jarvis
      Last Update:  04-04-2021
    -->
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="ItsJustJarvis Design and Development - Owned and Operated by Reeve-C-Jarvis. Personal, attentive service for your business needs.">
    <meta name="keywords" content="ItsJustJarvis, Design, Development, HTML, CSS, Java, Database, Freelance, Developer">
    <title>ItsJustJarvis Development</title>
    <link rel="stylesheet" href="css_base.css">
    <link rel="stylesheet" href="css_layout.css">
    <link rel="stylesheet" href="css_animations.css">
    <link rel="stylesheet" href="css_colourStyles.css">
    <link rel="stylesheet" href="css_blackAndWhiteStyles.css">
    <link rel="preconnect" href="https://fonts.gstatic.com">
    <link href="https://fonts.googleapis.com/css2?family=Open+Sans&amp;display=swap" rel="stylesheet">
  </head>
  <body>
    <header>
      <div id="header">
        <div id="headerLogo">
          <a href="ItsJustJarvis.html"><img id="logo" src="photos/titleImage.png" alt="ItsJustJarvis Development"></a>
        </div>
        <div id="adminForm">
          <div class="adminOnly" style="display: none;">
            <p>Logged In As Admin</p>
            <input type="submit" name="Logout" id="logOut" value="Log Out">
          </div>
          <form id="adminLogIn">
            <fieldset id="logInDetails">
              <legend>Admin Log-In</legend>
              <label for="userName">User Name:</label>
              <input type="text" name="username" id="userName" placeholder="Username">
              <label for="passWord">Password:</label>
              <input type="password" name="password" id="passWord" placeholder="Password">
              <input type="submit" value="Login" id="submitLogIn">
            </fieldset>
          </form>
        </div>
      </div>
      <nav id="tabs">
        <ul>
          <li><a href="aboutme.html">About Me</a></li>
          <li><a href="portfolio.html">Portfolio</a></li>
          <li><a href="gameroom.html">Game Room</a></li>
          <li><a href="contact.html">Contact</a></li>
        </ul>
      </nav>
    </header>
    <article id="welcome">
      <h1 id="message">Welcome to ItsJustJarvis Design and Development</h1>
      <p>
        My name is Reeve Jarvis and I am an aspiring developer starting his
        journey towards a career in tech. Currently I am studying in the Web and
        Mobile App Development Diploma program at North Island College,
        graduating Spring 2022. Follow along as I build my portfolio and launch
        my career into the tech industry.
      </p>
      <p>
        Outside of my work, I have a passion for tech and video games. Visit the
        Game Room page to see what I am currently playing, and details on my top
        three.
      </p>
```

https://validator.w3.org/nu/ 1/2

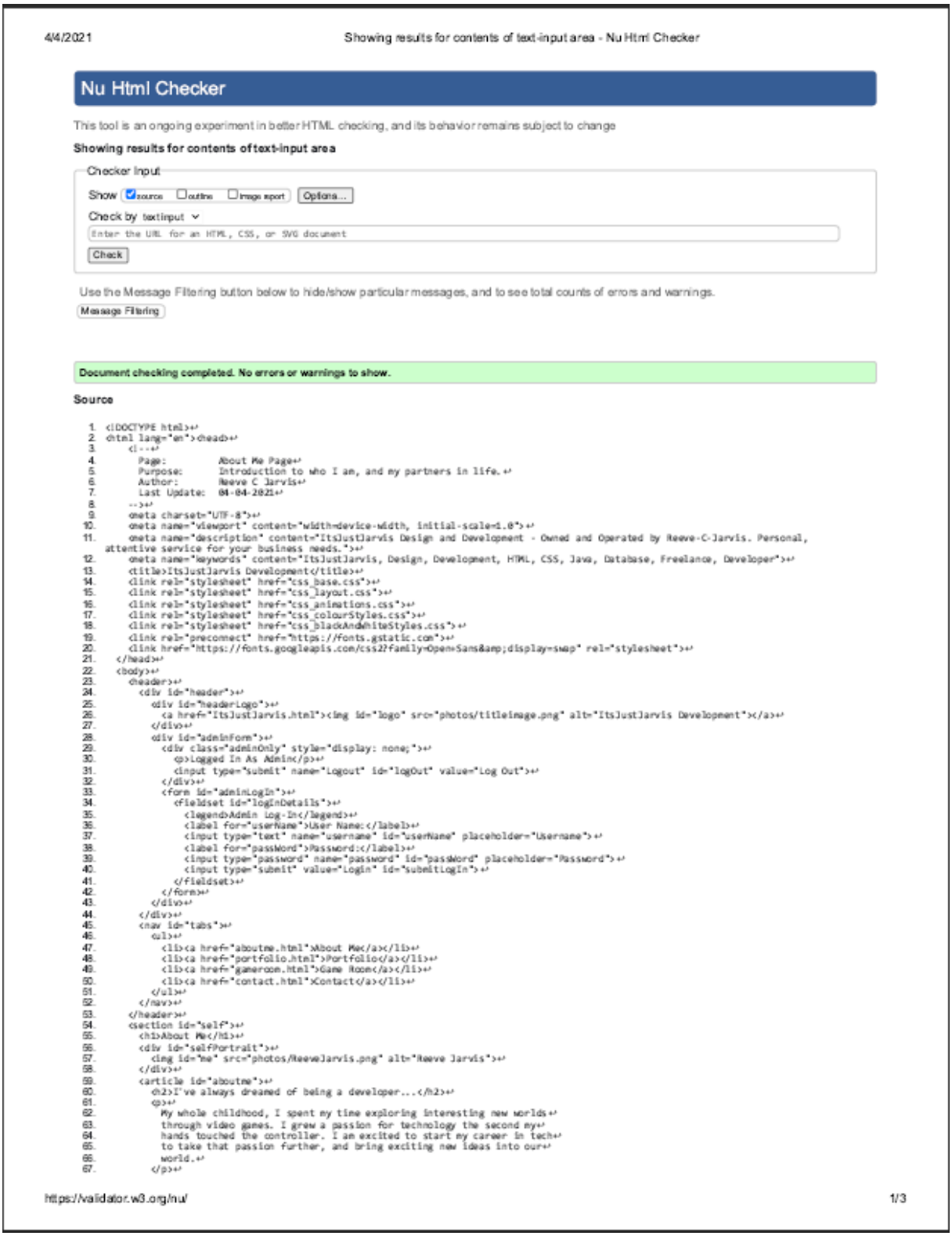
Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

About Me HTML:



Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

About Me after JS:



4/4/2021 Showing results for contents of text-input area - Nu Html Checker

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show source outline image report Options...

Check by textinput

Enter the URL for an HTML, CSS, or SVG document

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

Document checking completed. No errors or warnings to show.

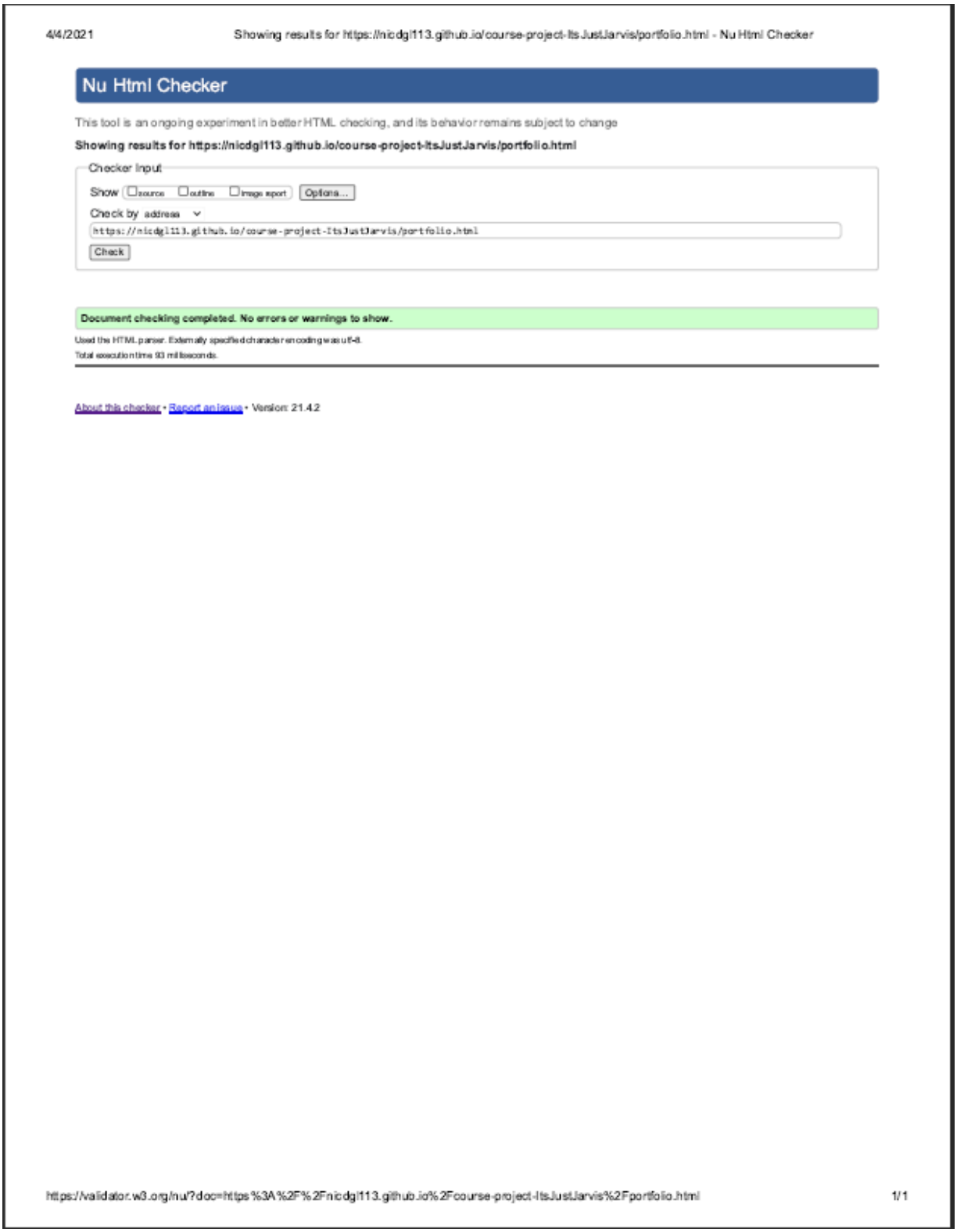
Source

```
<!DOCTYPE html>↵
<html lang="en"><head>↵
    <!--↵
      Page:         About Me Page↵
      Purpose:      Introduction to who I am, and my partners in life.↵
      Author:       Reeve C Jarvis↵
      Last Update:  04-04-2021↵
    -->↵
    <meta charset="UTF-8">↵
    <meta name="viewport" content="width=device-width, initial-scale=1.0">↵
    <meta name="description" content="ItsJustJarvis Design and Development - Owned and Operated by Reeve-C-Jarvis. Personal,
attentive service for your business needs.">↵
    <meta name="keywords" content="ItsJustJarvis, Design, Development, HTML, CSS, Java, Database, Freelance, Developer">↵
    <title>ItsJustJarvis Development</title>↵
    <link rel="stylesheet" href="css_base.css">↵
    <link rel="stylesheet" href="css_layout.css">↵
    <link rel="stylesheet" href="css_animations.css">↵
    <link rel="stylesheet" href="css_colourStyles.css">↵
    <link rel="stylesheet" href="css_blackAndWhiteStyles.css">↵
    <link rel="preconnect" href="https://fonts.gstatic.com">↵
    <link href="https://fonts.googleapis.com/css2?family=Open+Sans&amp;display=swap" rel="stylesheet">↵
  </head>↵
  <body>↵
    <header>↵
      <div id="header">↵
        <div id="headerLogo">↵
          <a href="ItsJustJarvis.html"><img id="logo" src="photos/titleImage.png" alt="ItsJustJarvis Development"></a>↵
        </div>↵
        <div id="adminForm">↵
          <div class="adminOnly" style="display: none;">↵
            <p>Logged In As Admin</p>↵
            <input type="submit" name="Logout" id="logOut" value="Log Out">↵
          </div>↵
          <form id="adminLogIn">↵
            <fieldset id="logInDetails">↵
              <legend>Admin Log-In</legend>↵
              <label for="userName">User Name:</label>↵
              <input type="text" name="username" id="userName" placeholder="Username">↵
              <label for="passWord">Password:</label>↵
              <input type="password" name="password" id="passWord" placeholder="Password">↵
              <input type="submit" value="Login" id="submitLogIn">↵
            </fieldset>↵
          </form>↵
        </div>↵
      </div>↵
      <nav id="tabs">↵
        <ul>↵
          <li><a href="aboutme.html">About Me</a></li>↵
          <li><a href="portfolio.html">Portfolio</a></li>↵
          <li><a href="gameroom.html">Game Room</a></li>↵
          <li><a href="contact.html">Contact</a></li>↵
        </ul>↵
      </nav>↵
    </header>↵
    <section id="self">↵
      <h1>About Me</h1>↵
      <div id="selfPortrait">↵
        <img id="me" src="photos/Reevejarvis.png" alt="Reeve Jarvis">↵
      </div>↵
      <article id="aboutme">↵
        <h2>I've always dreamed of being a developer...</h2>↵
        <p>↵
          My whole childhood, I spent my time exploring interesting new worlds↵
          through video games. I grew a passion for technology the second my↵
          hands touched the controller. I am excited to start my career in tech↵
          to take that passion further, and bring exciting new ideas into our↵
          world.↵
        </p>↵
```

https://validator.w3.org/nu/ 1/3

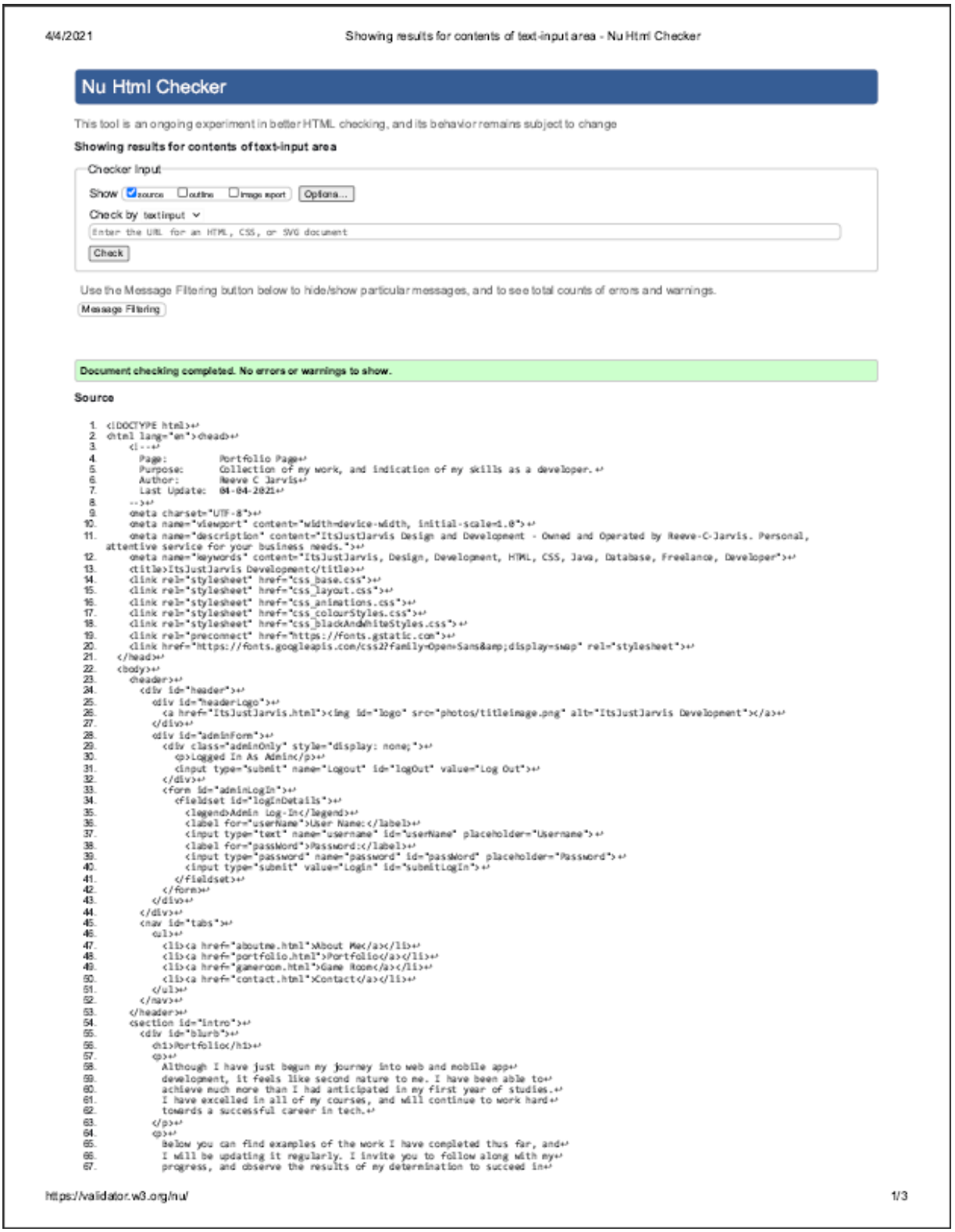
Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Portfolio HTML:



Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Portfolio after JS:



4/4/2021 Showing results for contents of text-input area - Nu Html Checker

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show source outline image report Options...

Check by textinput

Enter the URL for an HTML, CSS, or SVG document

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

Document checking completed. No errors or warnings to show.

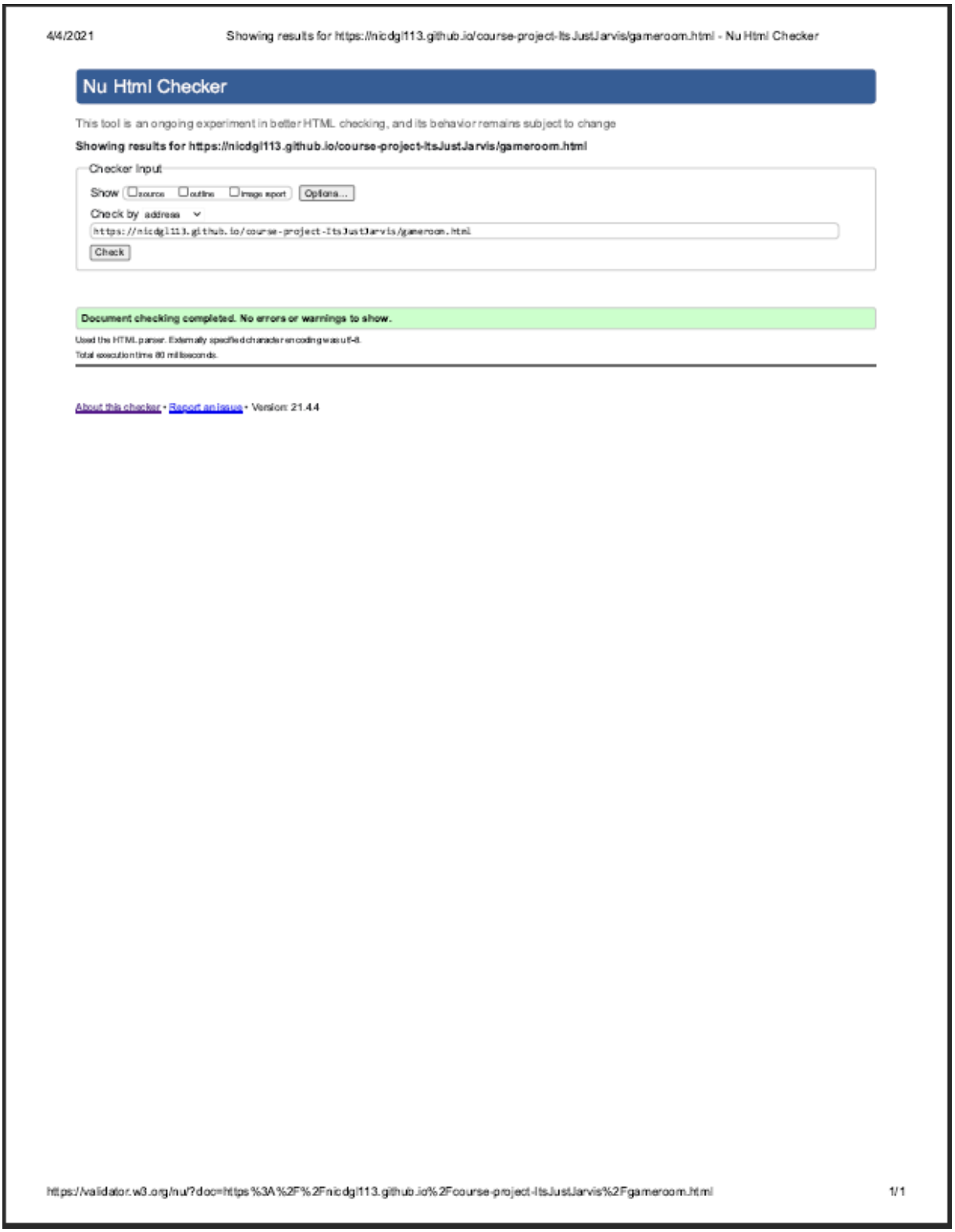
Source

```
1.  <!DOCTYPE html>↵
2.  <html lang="en">↵<head>↵
3.      <!--↵
4.        Page:         Portfolio Page↵
5.        Purpose:      Collection of my work, and indication of my skills as a developer.↵
6.        Author:       Reeve C Jarvis↵
7.        Last Update:  04-04-2021↵
8.      -->↵
9.      <meta charset="UTF-8">↵
10.     <meta name="viewport" content="width=device-width, initial-scale=1.0">↵
11.     <meta name="description" content="ItsJustJarvis Design and Development - Owned and Operated by Reeve-C-Jarvis. Personal,
attentive service for your business needs.">↵
12.     <meta name="keywords" content="ItsJustJarvis, Design, Development, HTML, CSS, Java, Database, Freelance, Developer">↵
13.     <title>ItsJustJarvis Development</title>↵
14.     <link rel="stylesheet" href="css_base.css">↵
15.     <link rel="stylesheet" href="css_layout.css">↵
16.     <link rel="stylesheet" href="css_animations.css">↵
17.     <link rel="stylesheet" href="css_colourStyles.css">↵
18.     <link rel="stylesheet" href="css_blackAndWhiteStyles.css">↵
19.     <link rel="preconnect" href="https://fonts.gstatic.com">↵
20.     <link href="https://fonts.googleapis.com/css2?family=Open+Sans&amp;display=swap" rel="stylesheet">↵
21.   </head>↵
22.   <body>↵
23.     <header>↵
24.       <div id="header">↵
25.         <div id="headerLogo">↵
26.           <a href="ItsJustJarvis.html"><img id="logo" src="photos/titleImage.png" alt="ItsJustJarvis Development"></a>↵
27.         </div>↵
28.         <div id="adminForm">↵
29.           <div class="adminOnly" style="display: none;">↵
30.             <p>Logged In As Admin</p>↵
31.             <input type="submit" name="Logout" id="logOut" value="Log Out">↵
32.           </div>↵
33.           <form id="adminLogIn">↵
34.             <fieldset id="logInDetails">↵
35.               <legend>Admin Log-In</legend>↵
36.               <label for="userName">User Name:</label>↵
37.               <input type="text" name="username" id="userName" placeholder="Username">↵
38.               <label for="passWord">Password:</label>↵
39.               <input type="password" name="password" id="passWord" placeholder="Password">↵
40.               <input type="submit" value="Login" id="submitLogIn">↵
41.             </fieldset>↵
42.           </form>↵
43.         </div>↵
44.       </div>↵
45.       <nav id="tabs">↵
46.         <ul>↵
47.           <li><a href="aboutme.html">About Me</a></li>↵
48.           <li><a href="portfolio.html">Portfolio</a></li>↵
49.           <li><a href="gameroom.html">Game Room</a></li>↵
50.           <li><a href="contact.html">Contact</a></li>↵
51.         </ul>↵
52.       </nav>↵
53.     </header>↵
54.     <section id="intro">↵
55.       <div id="blurb">↵
56.         <h1>Portfolio</h1>↵
57.         <p>↵
58.           Although I have just begun my journey into web and mobile app↵
59.           development, it feels like second nature to me. I have been able to↵
60.           achieve much more than I had anticipated in my first year of studies.↵
61.           I have excelled in all of my courses, and will continue to work hard↵
62.           towards a successful career in tech.↵
63.         </p>↵
64.         <p>↵
65.           Below you can find examples of the work I have completed thus far, and↵
66.           I will be updating it regularly. I invite you to follow along with my↵
67.           progress, and observe the results of my determination to succeed in↵
```

https://validator.w3.org/nu/ 1/3

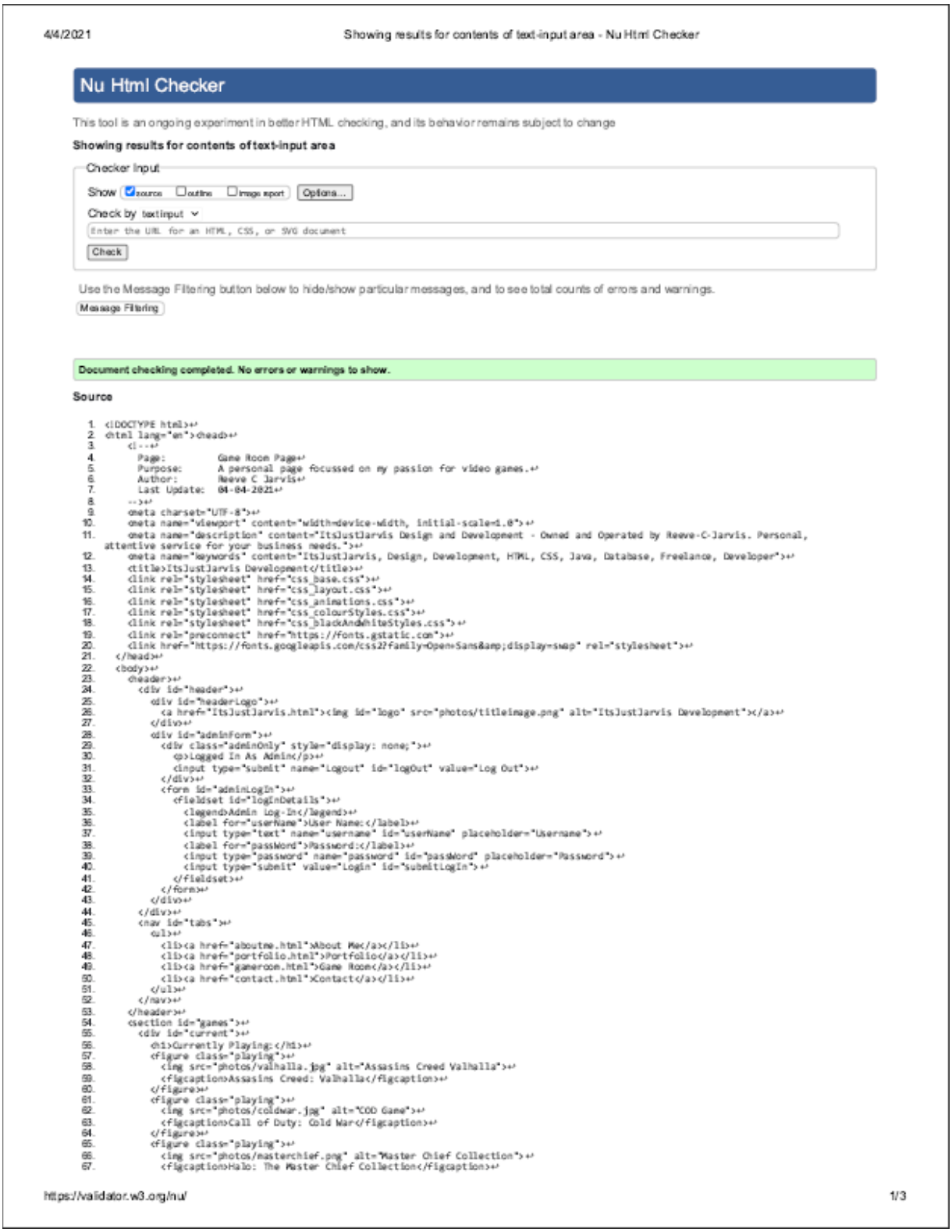
Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Gameroom HTML:



Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Gameroom after JS:



4/4/2021 Showing results for contents of text-input area - Nu Html Checker

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show source outline image report Options...

Check by textinput

Enter the URL for an HTML, CSS, or SVG document

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

Document checking completed. No errors or warnings to show.

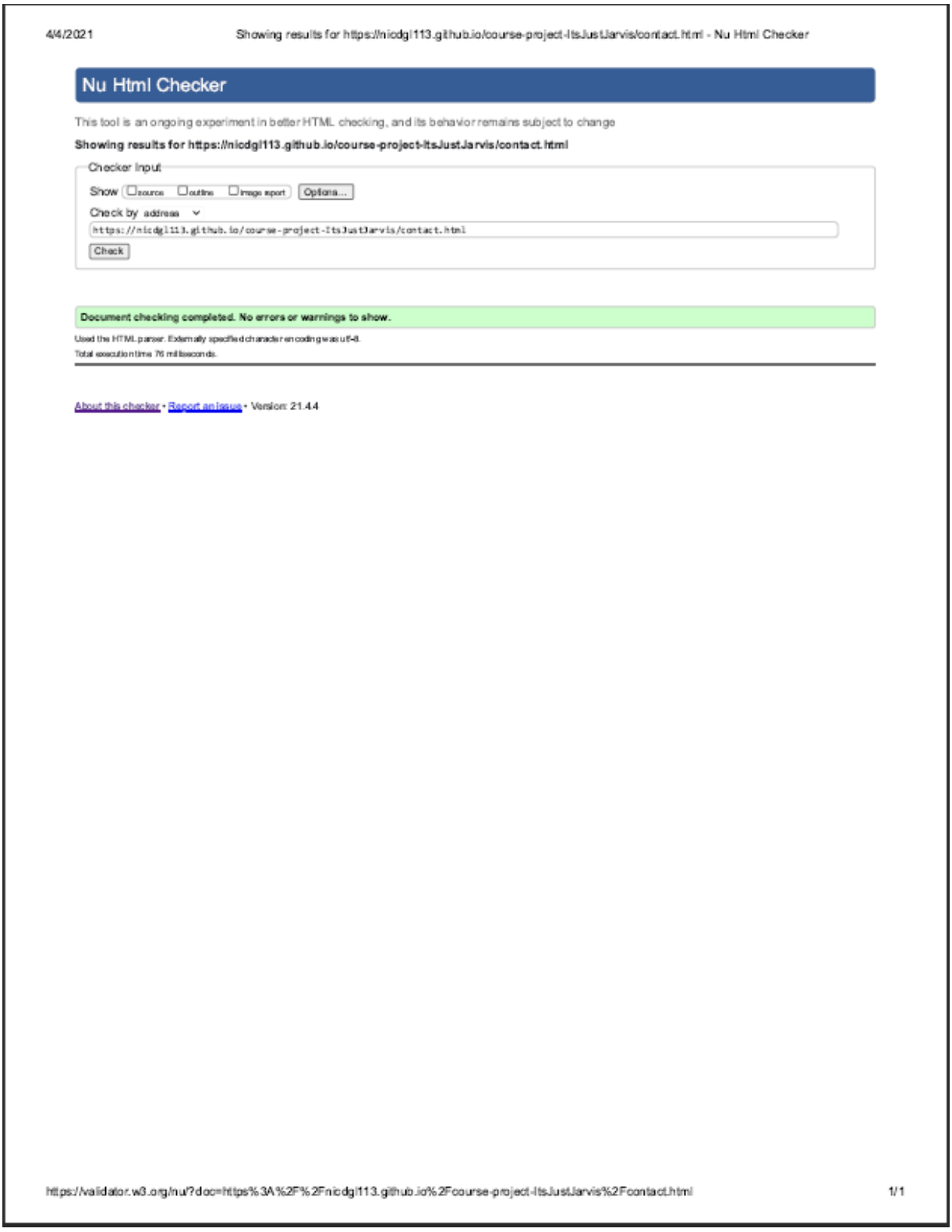
Source

```
<!DOCTYPE html>↵
<html lang="en"><head>↵
    <!--↵
      Page:         Game Room Page↵
      Purpose:      A personal page focussed on my passion for video games.↵
      Author:       Reeve C Jarvis↵
      Last Update:  04-04-2021↵
    -->↵
    <meta charset="UTF-8">↵
    <meta name="viewport" content="width=device-width, initial-scale=1.0">↵
    <meta name="description" content="ItsJustJarvis Design and Development - Owned and Operated by Reeve-C-Jarvis. Personal,
attentive service for your business needs.">↵
    <meta name="keywords" content="ItsJustJarvis, Design, Development, HTML, CSS, Java, Database, Freelance, Developer">↵
    <title>ItsJustJarvis Development</title>↵
    <link rel="stylesheet" href="css_base.css">↵
    <link rel="stylesheet" href="css_layout.css">↵
    <link rel="stylesheet" href="css_animations.css">↵
    <link rel="stylesheet" href="css_colourStyles.css">↵
    <link rel="stylesheet" href="css_blackAndWhiteStyles.css">↵
    <link rel="preconnect" href="https://fonts.gstatic.com">↵
    <link href="https://fonts.googleapis.com/css2?family=Open+Sans&amp;display=swap" rel="stylesheet">↵
  </head>↵
  <body>↵
    <header>↵
      <div id="header">↵
        <div id="headerLogo">↵
          <a href="ItsJustJarvis.html"><img id="logo" src="photos/titleImage.png" alt="ItsJustJarvis Development"></a>↵
        </div>↵
        <div id="adminForm">↵
          <div class="adminOnly" style="display: none;">↵
            <p>Logged In As Admin</p>↵
            <input type="submit" name="Logout" id="logOut" value="Log Out">↵
          </div>↵
          <form id="adminLogIn">↵
            <fieldset id="logInDetails">↵
              <legend>Admin Log-In</legend>↵
              <label for="userName">User Name:</label>↵
              <input type="text" name="username" id="userName" placeholder="Username">↵
              <label for="passWord">Password:</label>↵
              <input type="password" name="password" id="passWord" placeholder="Password">↵
              <input type="submit" value="Login" id="submitLogIn">↵
            </fieldset>↵
          </form>↵
        </div>↵
      </div>↵
      <nav id="tabs">↵
        <ul>↵
          <li><a href="aboutme.html">About Me</a></li>↵
          <li><a href="portfolio.html">Portfolio</a></li>↵
          <li><a href="gameroom.html">Game Room</a></li>↵
          <li><a href="contact.html">Contact</a></li>↵
        </ul>↵
      </nav>↵
    </header>↵
    <section id="games">↵
      <div id="current">↵
        <h1>Currently Playing:</h1>↵
        <figure class="playing">↵
          <img src="photos/valhalla.jpg" alt="Assasins Creed Valhalla">↵
          <figcaption>Assasins Creed: Valhalla</figcaption>↵
        </figure>↵
        <figure class="playing">↵
          <img src="photos/coldwar.jpg" alt="COD Game">↵
          <figcaption>Call of Duty: Cold War</figcaption>↵
        </figure>↵
        <figure class="playing">↵
          <img src="photos/masterchief.png" alt="Master Chief Collection">↵
          <figcaption>Halo: The Master Chief Collection</figcaption>↵
```

https://validator.w3.org/nu/ 1/3

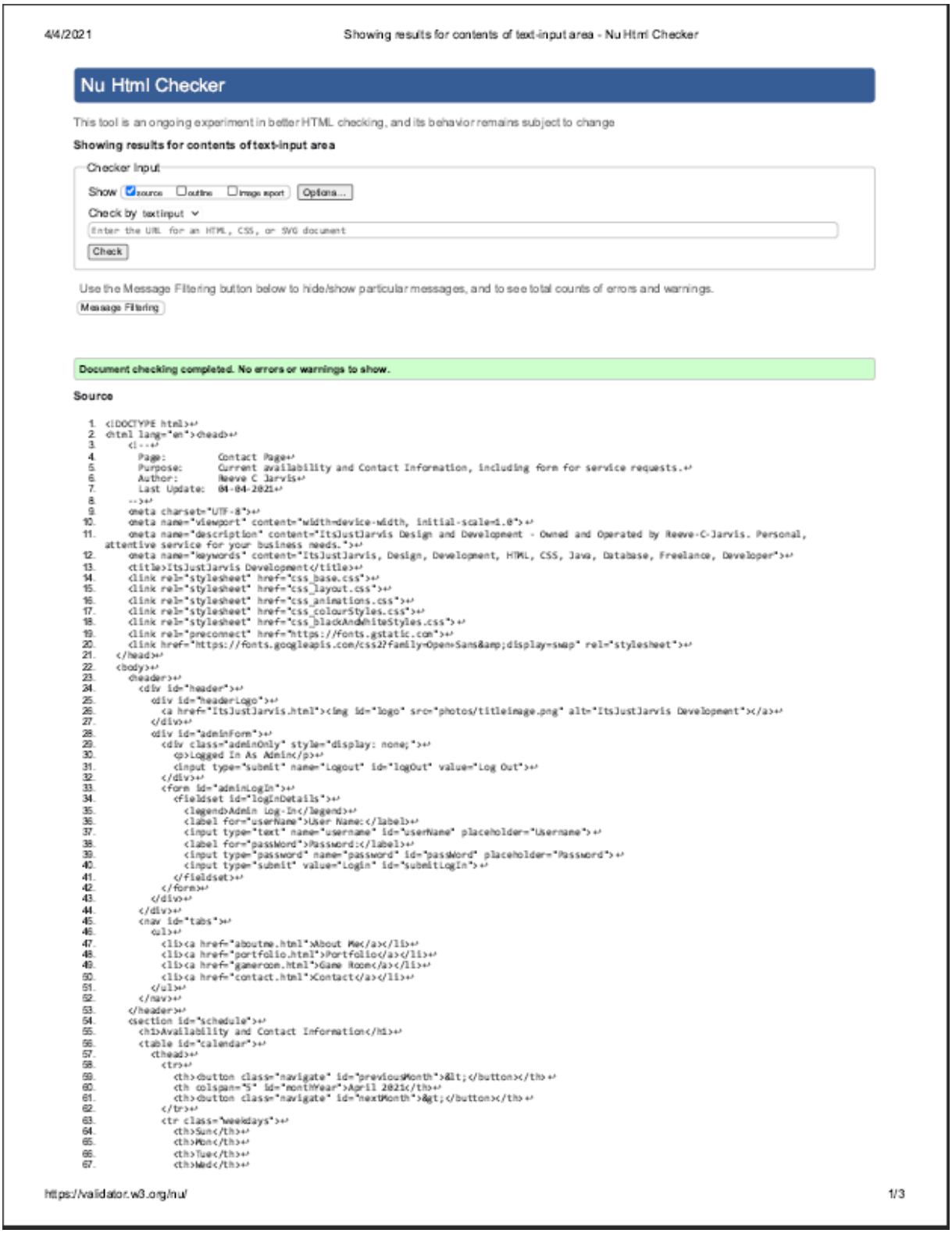
Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Contact HTML:



Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Contact after JS:



4/4/2021 Showing results for contents of text-input area - Nu Html Checker

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show [x] source [ ] outline [ ] image report Options...

Check by textinput

Enter the URL for an HTML, CSS, or SVG document

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

Document checking completed. No errors or warnings to show.

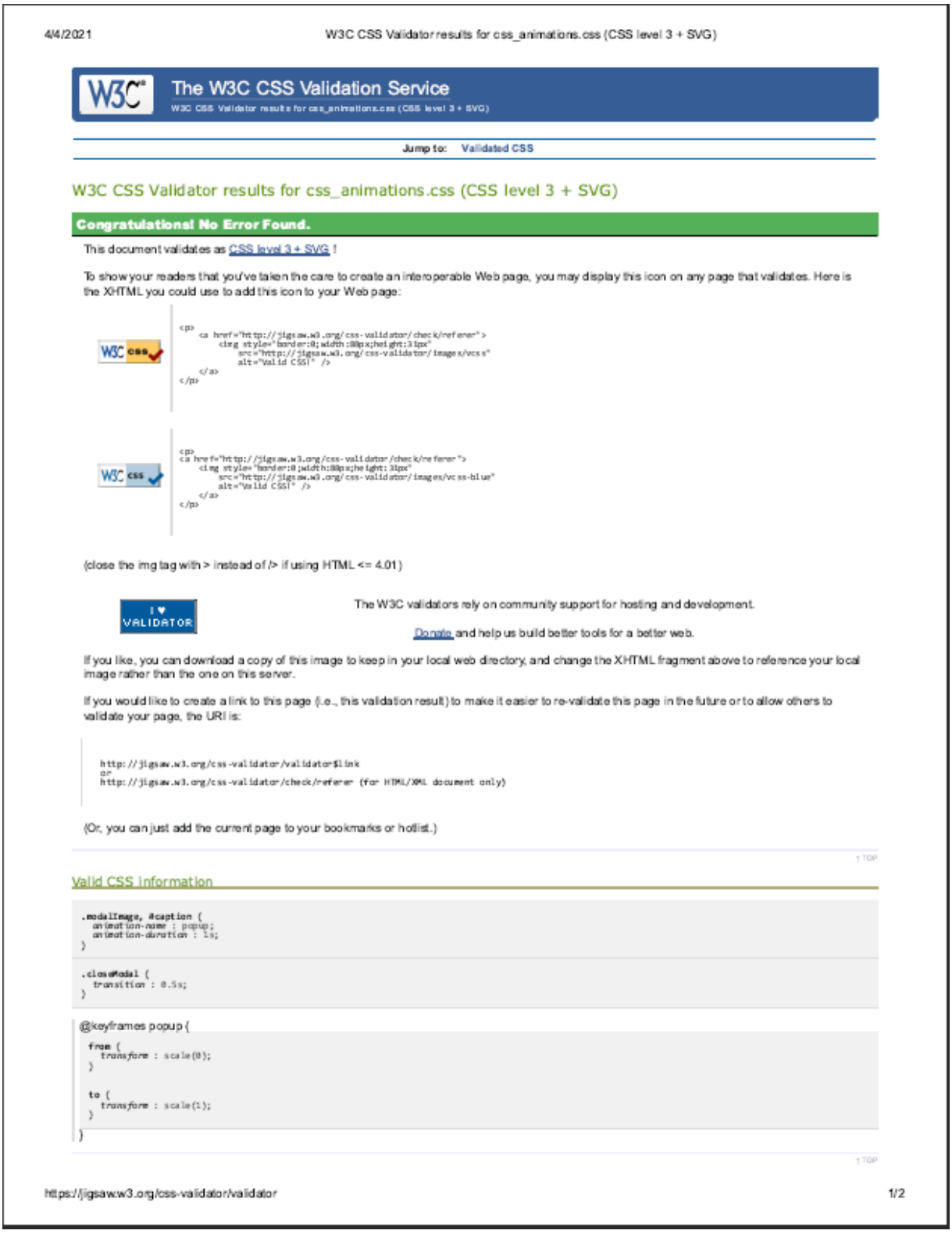
Source

```
1.  <!DOCTYPE html>↵
2.  <html lang="en"><head>↵
3.      <!--↵
4.        Page:         Contact Page↵
5.        Purpose:      Current availability and Contact Information, including form for service requests.↵
6.        Author:       Reeve C Jarvis↵
7.        Last Update:  04-04-2021↵
8.      -->↵
9.      <meta charset="UTF-8">↵
10.     <meta name="viewport" content="width=device-width, initial-scale=1.0">↵
11.     <meta name="description" content="ItsJustJarvis Design and Development - Owned and Operated by Reeve-C-Jarvis. Personal,
    attentive service for your business needs.">↵
12.     <meta name="keywords" content="ItsJustJarvis, Design, Development, HTML, CSS, Java, Database, Freelance, Developer">↵
13.     <title>ItsJustJarvis Development</title>↵
14.     <link rel="stylesheet" href="css_base.css">↵
15.     <link rel="stylesheet" href="css_layout.css">↵
16.     <link rel="stylesheet" href="css_animations.css">↵
17.     <link rel="stylesheet" href="css_colourStyles.css">↵
18.     <link rel="stylesheet" href="css_blackAndWhiteStyles.css">↵
19.     <link rel="preconnect" href="https://fonts.gstatic.com">↵
20.     <link href="https://fonts.googleapis.com/css2?family=Open+Sans&amp;display=swap" rel="stylesheet">↵
21.   </head>↵
22.   <body>↵
23.     <header>↵
24.       <div id="header">↵
25.         <div id="headerLogo">↵
26.           <a href="ItsJustJarvis.html"><img id="logo" src="photos/titleImage.png" alt="ItsJustJarvis Development"></a>↵
27.         </div>↵
28.         <div id="adminForm">↵
29.           <div class="adminOnly" style="display: none;">↵
30.             <p>Logged In As Admin</p>↵
31.             <input type="submit" name="Logout" id="logOut" value="Log Out">↵
32.           </div>↵
33.           <form id="adminLogIn">↵
34.             <fieldset id="logInDetails">↵
35.               <legend>Admin Log-In</legend>↵
36.               <label for="userName">User Name:</label>↵
37.               <input type="text" name="username" id="userName" placeholder="Username">↵
38.               <label for="passWord">Password:</label>↵
39.               <input type="password" name="password" id="passWord" placeholder="Password">↵
40.               <input type="submit" value="Login" id="submitLogIn">↵
41.             </fieldset>↵
42.           </form>↵
43.         </div>↵
44.       </div>↵
45.       <nav id="tabs">↵
46.         <ul>↵
47.           <li><a href="aboutme.html">About Me</a></li>↵
48.           <li><a href="portfolio.html">Portfolio</a></li>↵
49.           <li><a href="gameroom.html">Game Room</a></li>↵
50.           <li><a href="contact.html">Contact</a></li>↵
51.         </ul>↵
52.       </nav>↵
53.     </header>↵
54.     <section id="schedule">↵
55.       <h1>Availability and Contact Information</h1>↵
56.       <table id="calendar">↵
57.         <thead>↵
58.           <tr>↵
59.             <th><button class="navigate" id="previousMonth">&lt;</button></th>↵
60.             <th colspan="5" id="monthYear">April 2021</th>↵
61.             <th><button class="navigate" id="nextMonth">&gt;</button></th>↵
62.           </tr>↵
63.           <tr class="weekdays">↵
64.             <th>Sun</th>↵
65.             <th>Mon</th>↵
66.             <th>Tue</th>↵
67.             <th>Wed</th>↵
```

https://validator.w3.org/nu/ 1/3

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Animations CSS:



4/4/2021 W3C CSS Validator results for css_animations.css (CSS level 3 + SVG)

The W3C CSS Validation Service

W3C CSS Validator results for css_animations.css (CSS level 3 + SVG)

Jump to: Validated CSS

W3C CSS Validator results for css_animations.css (CSS level 3 + SVG)

Congratulations! No Error Found.

This document validates as CSS level 3 + SVG !

To show your readers that you've taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the XHTML you could use to add this icon to your Web page:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="Valid CSS!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="Valid CSS!" />
    </a>
</p>
```

(close the img tag with > instead of /> if using HTML <= 4.01)

The W3C validators rely on community support for hosting and development.

Donate and help us build better tools for a better web.

If you like, you can download a copy of this image to keep in your local web directory, and change the XHTML fragment above to reference your local image rather than the one on this server.

If you would like to create a link to this page (i.e., this validation result) to make it easier to re-validate this page in the future or to allow others to validate your page, the URI is:

```
http://jigsaw.w3.org/css-validator/validator$link
or
http://jigsaw.w3.org/css-validator/check/referer (for HTML/XML document only)
```

(Or, you can just add the current page to your bookmarks or hotlist.)

↑ TOP

Valid CSS Information

```
.modalImage, #caption {
  animation-name : popup;
  animation-duration : 1s;
}
```

```
.closeModal {
  transition : 0.5s;
}
```

```
@keyframes popup {
  from {
    transform : scale(0);
  }

  to {
    transform : scale(1);
  }
}
```

↑ TOP

https://jigsaw.w3.org/css-validator/validator 1/2

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Base CSS:

4/4/2021 W3C CSS Validator results for css_base.css (CSS level 3 + SVG)

The W3C CSS Validation Service

W3C CSS Validator results for css_base.css (CSS level 3 + SVG)

Jump to: Warnings (2) Validated CSS

W3C CSS Validator results for css_base.css (CSS level 3 + SVG)

Congratulations! No Error Found.

This document validates as CSS level 3 + SVG !

To show your readers that you've taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the XHTML you could use to add this icon to your Web page:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="Valid CSS!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="Valid CSS!" />
    </a>
</p>
```

(close the img tag with > instead of /> if using HTML <= 4.01)

The W3C validators rely on community support for hosting and development.

Donate and help us build better tools for a better web.

If you like, you can download a copy of this image to keep in your local web directory, and change the XHTML fragment above to reference your local image rather than the one on this server.

If you would like to create a link to this page (i.e., this validation result) to make it easier to re-validate this page in the future or to allow others to validate your page, the URI is:

```
http://jigsaw.w3.org/css-validator/validator$link
or
http://jigsaw.w3.org/css-validator/check/referer (for HTML/XML document only)
```

(Or, you can just add the current page to your bookmarks or hotlist.)

↑ TOP

Warnings (2)

URI : css_base.css

60	-webkit-box-sizing is an unknown vendor extension
61	-moz-box-sizing is an unknown vendor extension

↑ TOP

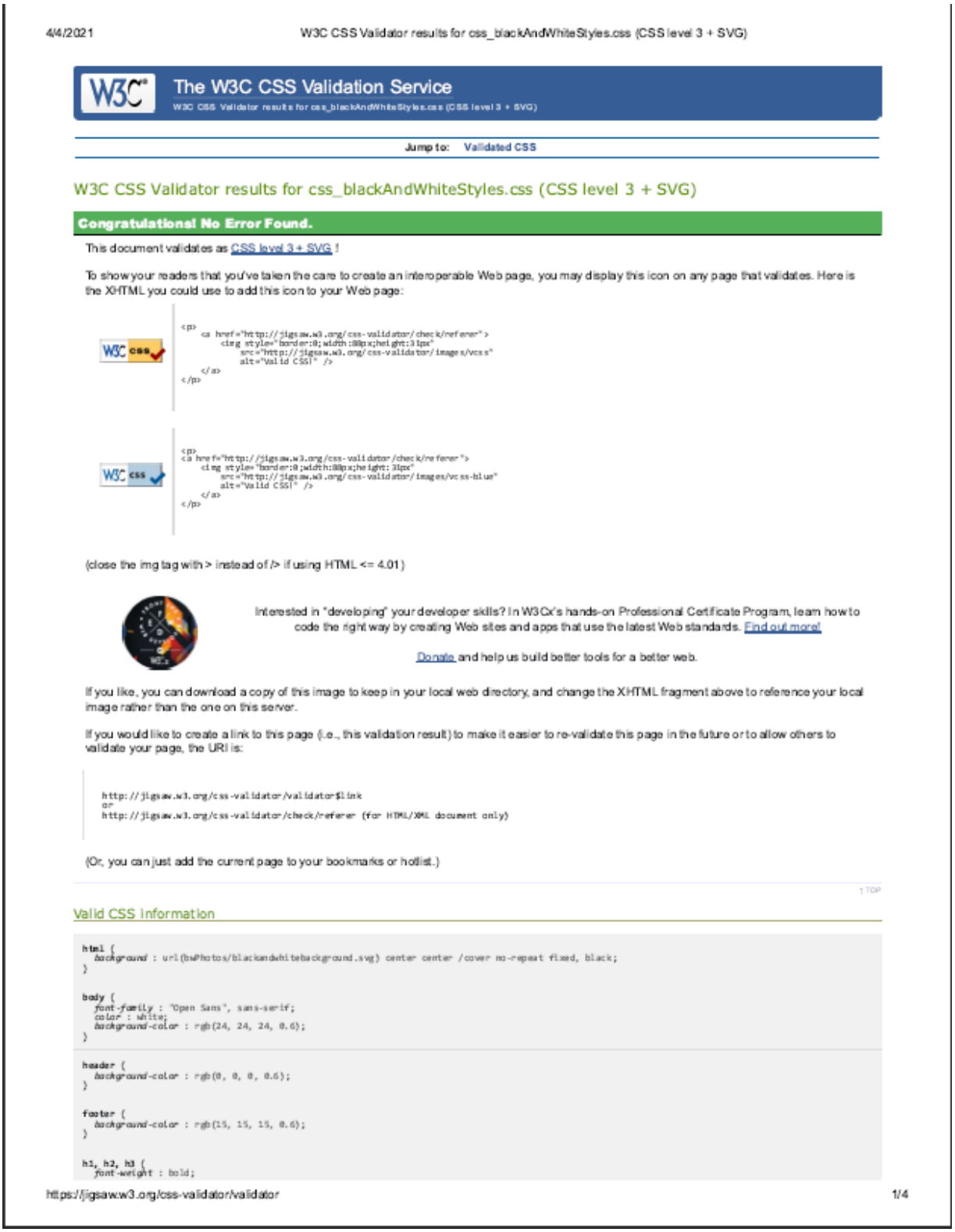
Valid CSS Information

```
article, aside, figcaption, figure, footer, header, main, nav, section {
  display : block;
}

address, article, aside, blockquote, body, cite, div, dl, dt, dd, em, figcaption, figure, footer, h1, h2, h3, h4, h5, h6, header, html, img, li,
main, nav, nav a, ol, p, section, span, ul {
  margin : 0;
  padding : 0;
  border : 0;
  font-size : 100%;
  vertical-align : baseline;
  background : transparent;
  box-sizing : border-box;
```

https://jigsaw.w3.org/css-validator/validator 1/2

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Black and White CSS:



4/4/2021 W3C CSS Validator results for css_blackAndWhiteStyles.css (CSS level 3 + SVG)

The W3C CSS Validation Service

W3C CSS Validator results for css_blackAndWhiteStyles.css (CSS level 3 + SVG)

Jump to: Validated CSS

W3C CSS Validator results for css_blackAndWhiteStyles.css (CSS level 3 + SVG)

Congratulations! No Error Found.

This document validates as CSS level 3 + SVG !

To show your readers that you've taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the XHTML you could use to add this icon to your Web page:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="Valid CSS!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="Valid CSS!" />
    </a>
</p>
```

(close the img tag with > instead of /> if using HTML <= 4.01)

Interested in "developing" your developer skills? In W3Cx's hands-on Professional Certificate Program, learn how to code the right way by creating Web sites and apps that use the latest Web standards. Find out more!

Donate and help us build better tools for a better web.

If you like, you can download a copy of this image to keep in your local web directory, and change the XHTML fragment above to reference your local image rather than the one on this server.

If you would like to create a link to this page (i.e., this validation result) to make it easier to re-validate this page in the future or to allow others to validate your page, the URI is:

```
http://jigsaw.w3.org/css-validator/validator$link
or
http://jigsaw.w3.org/css-validator/check/referer (for HTML/XML document only)
```

(Or, you can just add the current page to your bookmarks or hotlist.)

↑ TOP

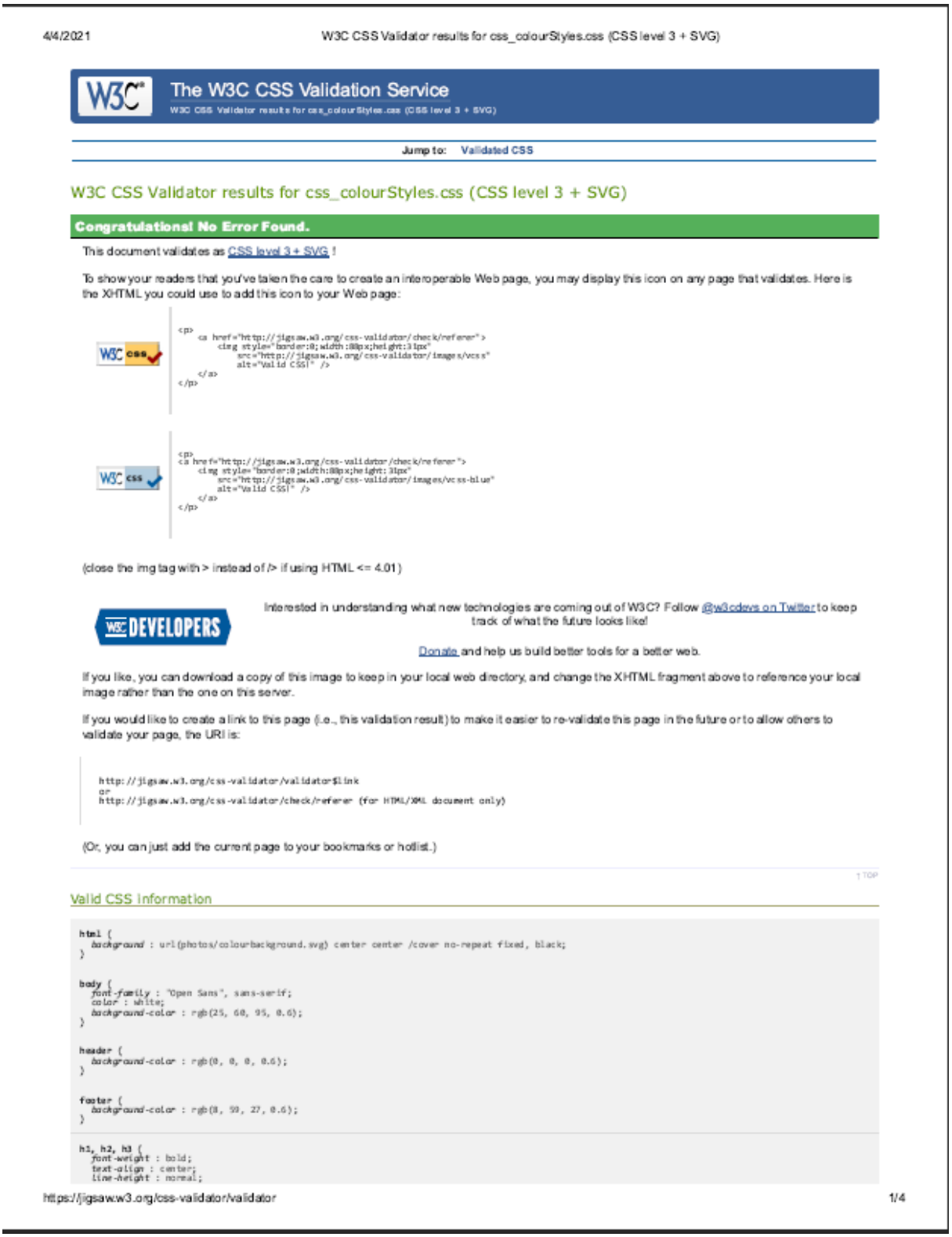
Valid CSS Information

```
html {
  background : url(bwPhotos/blackandwhitebackground.svg) center center /cover no-repeat fixed, black;
}

body {
  font-family : "Open Sans", sans-serif;
  color : white;
  background-color : rgb(24, 24, 24, 0.6);
}

header {
  background-color : rgb(0, 0, 0, 0.6);
}

footer {
  background-color : rgb(15, 15, 15, 0.6);
}

h1, h2, h3 {
  font-weight : bold;
```

https://jigsaw.w3.org/css-validator/validator 1/4

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Colour CSS:



4/4/2021 W3C CSS Validator results for css_colourStyles.css (CSS level 3 + SVG)

The W3C CSS Validation Service

W3C CSS Validator results for css_colourStyles.css (CSS level 3 + SVG)

Jump to: Validated CSS

W3C CSS Validator results for css_colourStyles.css (CSS level 3 + SVG)

Congratulations! No Error Found.

This document validates as CSS level 3 + SVG !

To show your readers that you've taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the XHTML you could use to add this icon to your Web page:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="Valid CSS!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="Valid CSS!" />
    </a>
</p>
```

(close the img tag with > instead of /> if using HTML <= 4.01)

Interested in understanding what new technologies are coming out of W3C? Follow @w3cdevs on Twitter to keep track of what the future looks like!

Donate and help us build better tools for a better web.

If you like, you can download a copy of this image to keep in your local web directory, and change the XHTML fragment above to reference your local image rather than the one on this server.

If you would like to create a link to this page (i.e., this validation result) to make it easier to re-validate this page in the future or to allow others to validate your page, the URI is:

```
http://jigsaw.w3.org/css-validator/validator$link
or
http://jigsaw.w3.org/css-validator/check/referer (for HTML/XML document only)
```

(Or, you can just add the current page to your bookmarks or hotlist.)

↑ TOP

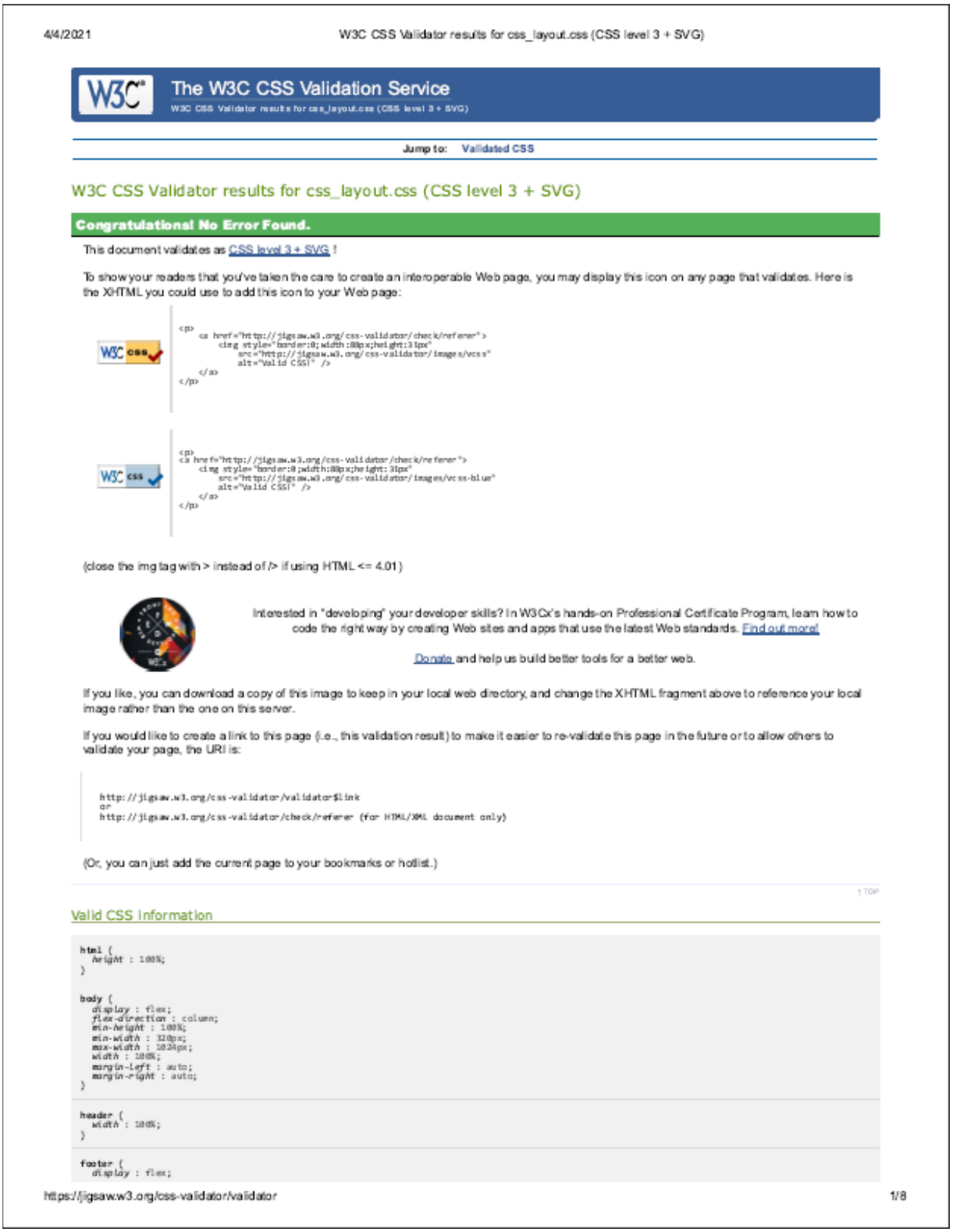
Valid CSS information

```
html {
  background : url(photos/colourbackground.svg) center center /cover no-repeat fixed, black;
}

body {
  font-family : "Open Sans", sans-serif;
  color : white;
  background-color : rgb(25, 60, 95, 0.6);
}

header {
  background-color : rgb(0, 0, 0, 0.6);
}

footer {
  background-color : rgb(0, 59, 27, 0.6);
}

h1, h2, h3 {
  font-weight : bold;
  text-align : center;
  line-height : normal;
```

https://jigsaw.w3.org/css-validator/validator 1/4

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Layout CSS:



4/4/2021 W3C CSS Validator results for css_layout.css (CSS level 3 + SVG)

The W3C CSS Validation Service

W3C CSS Validator results for css_layout.css (CSS level 3 + SVG)

Jump to: Validated CSS

W3C CSS Validator results for css_layout.css (CSS level 3 + SVG)

Congratulations! No Error Found.

This document validates as CSS level 3 + SVG !

To show your readers that you've taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the XHTML you could use to add this icon to your Web page:

```
<p>
    <a href="http://jigsaw.w3.org/css-validator/check/referer">
        <img style="border:0;width:88px;height:31px"
            src="http://jigsaw.w3.org/css-validator/images/vcss"
            alt="Valid CSS!" />
    </a>
</p>
```

```
<p>
<a href="http://jigsaw.w3.org/css-validator/check/referer">
    <img style="border:0;width:88px;height:31px"
        src="http://jigsaw.w3.org/css-validator/images/vcss-blue"
        alt="Valid CSS!" />
    </a>
</p>
```

(close the img tag with > instead of /> if using HTML <= 4.01)

Interested in "developing" your developer skills? In W3Cx's hands-on Professional Certificate Program, learn how to code the right way by creating Web sites and apps that use the latest Web standards. Find out more!

Donate and help us build better tools for a better web.

If you like, you can download a copy of this image to keep in your local web directory, and change the XHTML fragment above to reference your local image rather than the one on this server.

If you would like to create a link to this page (i.e., this validation result) to make it easier to re-validate this page in the future or to allow others to validate your page, the URI is:

```
http://jigsaw.w3.org/css-validator/validator$link
or
http://jigsaw.w3.org/css-validator/check/referer (for HTML/XML document only)
```

(Or, you can just add the current page to your bookmarks or hotlist.)

↑ TOP

Valid CSS Information

```
html {
  height : 100%;
}

body {
  display : flex;
  flex-direction : column;
  min-height : 100%;
  min-width : 320px;
  max-width : 1024px;
  width : 100%;
  margin-left : auto;
  margin-right : auto;
}

header {
  width : 100%;
}

footer {
  display : flex;
```

https://jigsaw.w3.org/css-validator/validator 1/8

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html

Summary and Feedback:

Summary and Suggestion

Course Code: DGL-113 Section: DLU1 Assn#: Project Student Name: Reeve Jarvis

It took me 15 including research/troubleshooting hour(s) to complete the assignment. I have learned and feel good about 90 % of the skills covered this week.

From the assignment work I learnt the following:

Finishing this project really helped solidify my understanding of JavaScript Implementation. I feel really good about my project, and only wish I had time to do more. I plan to continue work on it throughout the summer. I learned a lot about properly using Date objects, and implementing functionality into a website.

I found the following helpful for completing the assignment (textbook, lecture notes, video, online resources, etc.):

Textbook, lecture notes, Online videos, and plentiful resources on sites like Stack Overflow helped me to achieve the results I did. I love to troubleshoot and problem solve, and I am happy with the end result of my project.

The difficulties I encountered with this assignment:

I encountered a few main difficulties along the way, especially with implementing the Calendar. This was a very tricky process but I feel like the end results are so worth it.

Other comments and suggestions:

Wish life didn't get in the way and I was able to be more present this semester. Thanks for being an awesome instructor as always Frank.

Copyright © Digital Design and Development, North Island College

Site URL: https://nicdgl113.github.io/course-project-ItsJustJarvis/ItsJustJarvis.html