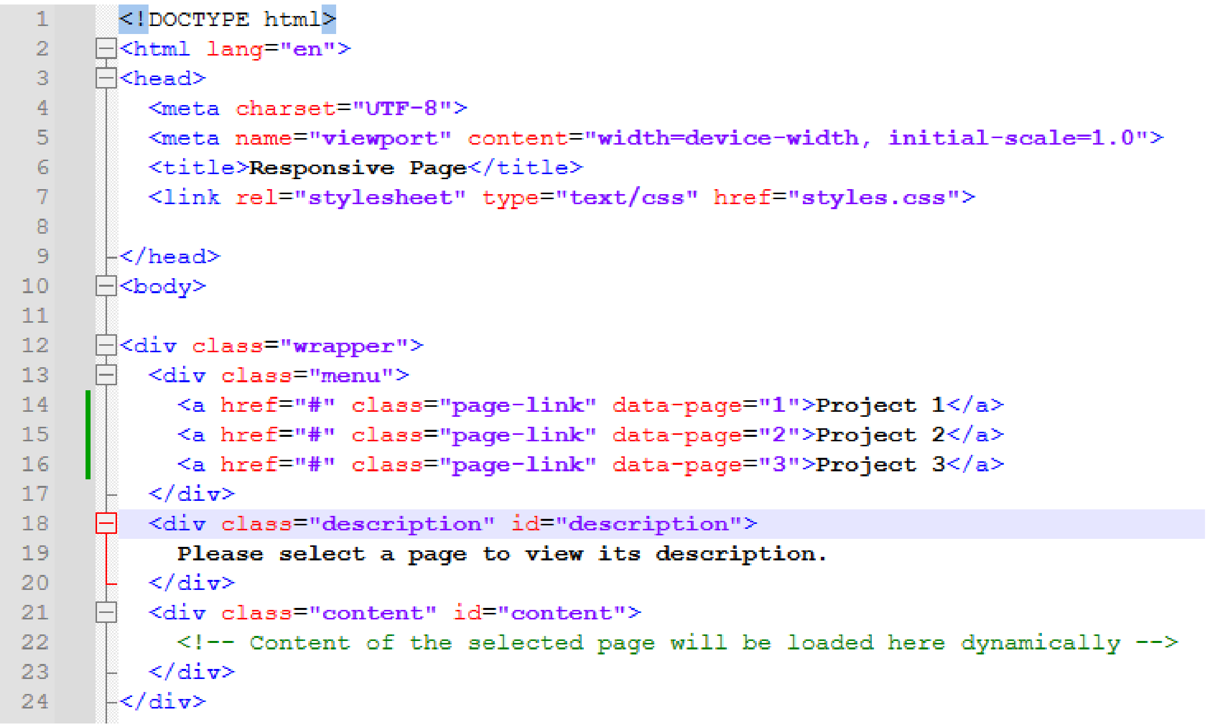
Warsaw University of Technology	Advanced Internet Programming
Project 4 Creating responsive websites	Maciej Iwańczyk 311258
Date: 15.03.2024	Grade:

1. Introduction:

In today’s project I had to create a simple responsive web page with a menu and dynamic content loading based on user interaction.

2.HTML:

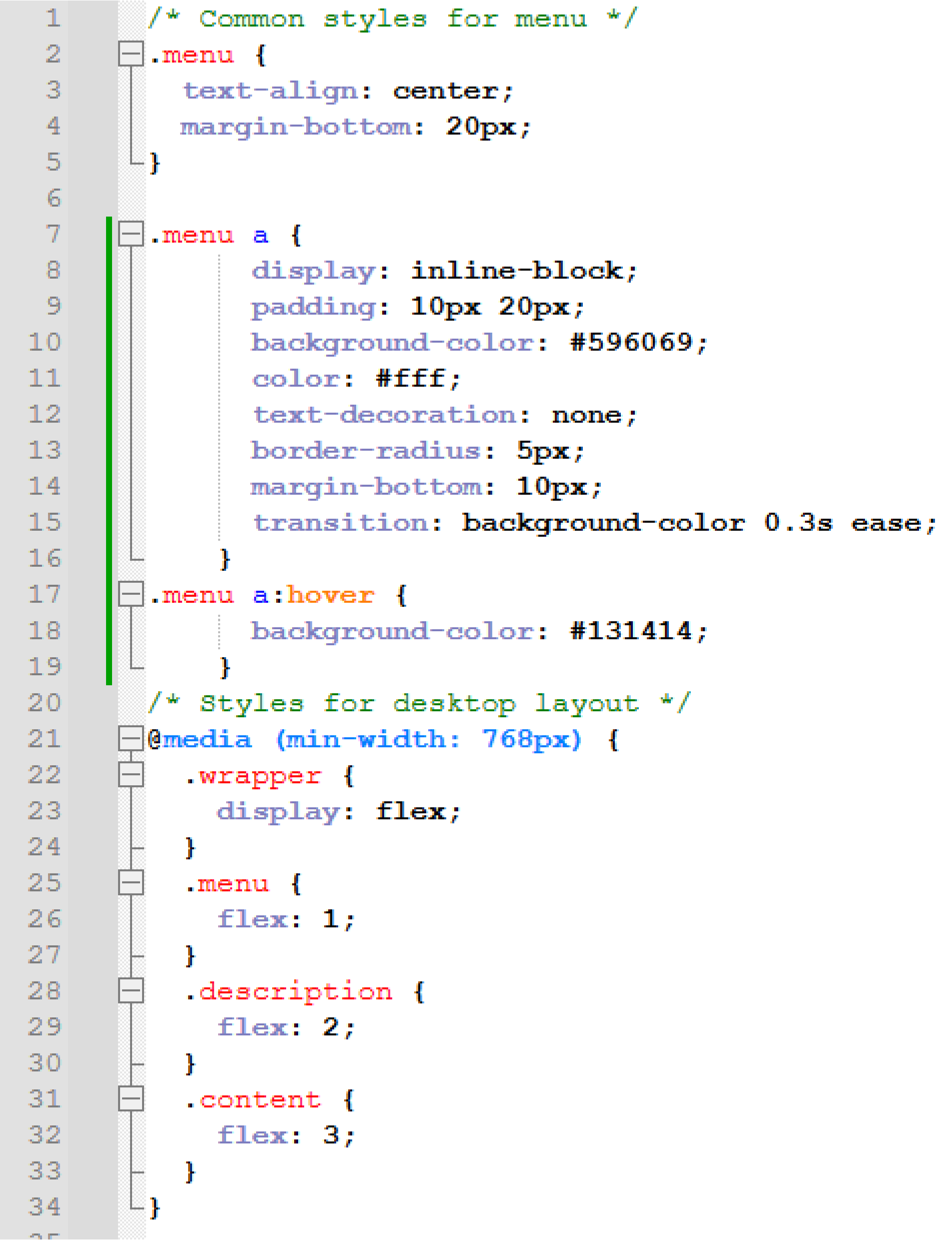
HTML part includes a wrapper element that contains:
A menu div for navigation links.
A description div for displaying information about the selected page.
A content div where the content of the selected page will be dynamically loaded.
Each navigation link (<a> tags with the class page-link) has a data-page attribute indicating the page number it corresponds to.



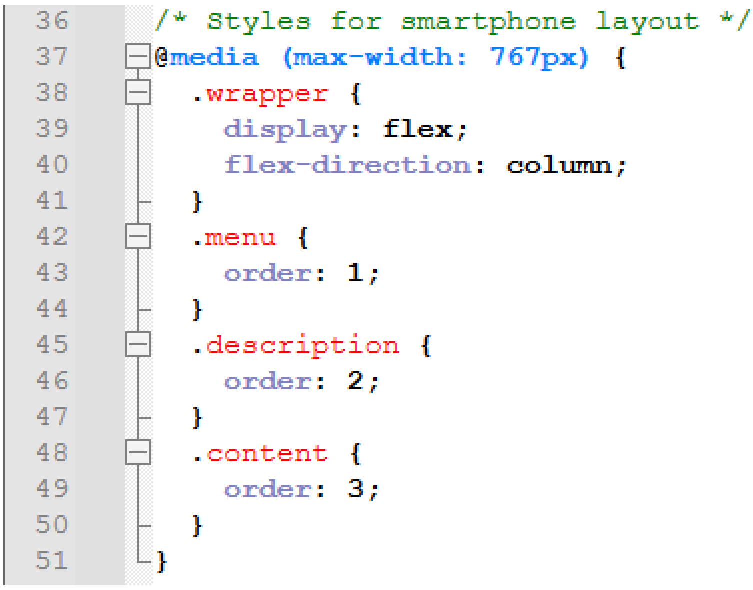
```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Responsive Page</title>
  <link rel="stylesheet" type="text/css" href="styles.css">

</head>
<body>

<div class="wrapper">
  <div class="menu">
    <a href="#" class="page-link" data-page="1">Project 1</a>
    <a href="#" class="page-link" data-page="2">Project 2</a>
    <a href="#" class="page-link" data-page="3">Project 3</a>
  </div>
  <div class="description" id="description">
    Please select a page to view its description.
  </div>
  <div class="content" id="content">
    <!-- Content of the selected page will be loaded here dynamically -->
  </div>
</div>
```

3.CSS:

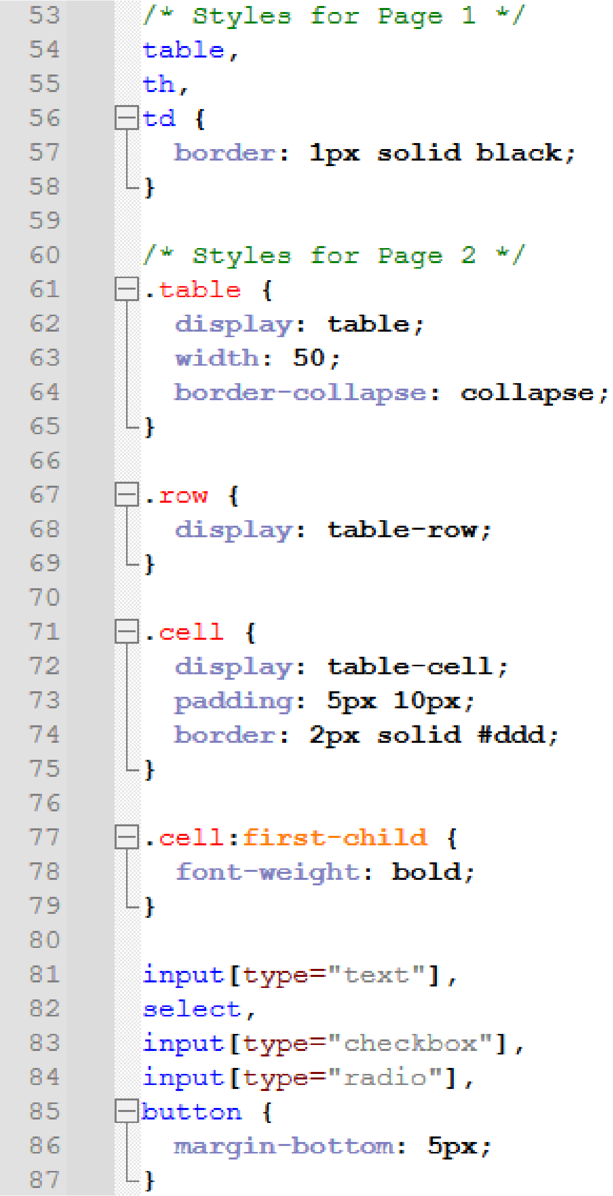
It includes styles for the menu, description, content, and responsive layout adjustments based on screen size.This allows the page to work flawlessly and look good both on desktops and on smartphones.
Additionally, there are specific styles for each project.



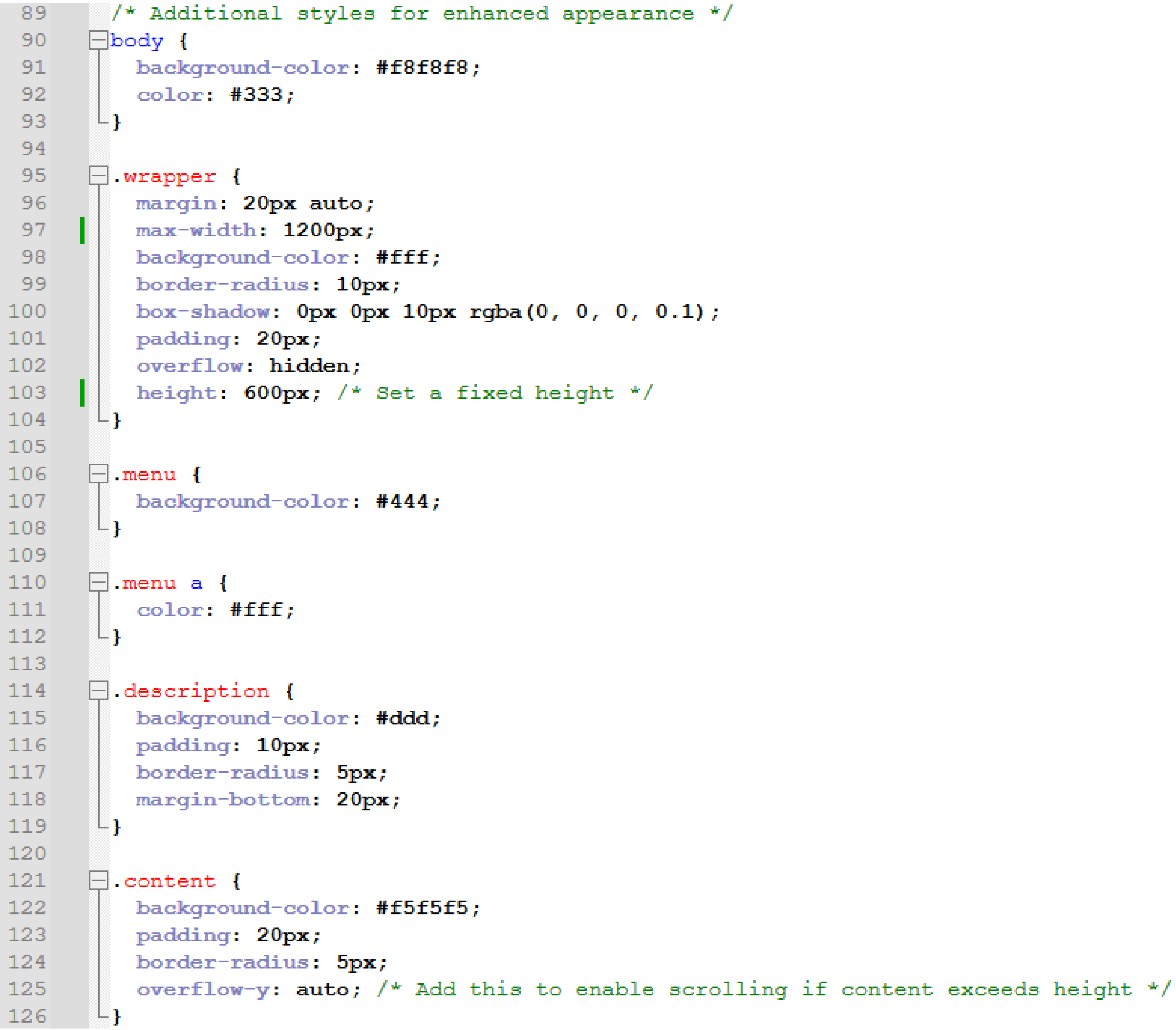
```
/* Common styles for menu */
.menu {
   text-align: center;
   margin-bottom: 20px;
}

.menu a {
         display: inline-block;
         padding: 10px 20px;
         background-color: #596069;
         color: #fff;
         text-decoration: none;
         border-radius: 5px;
         margin-bottom: 10px;
         transition: background-color 0.3s ease;
      }
.menu a:hover {
         background-color: #131414;
      }
/* Styles for desktop layout */
@media (min-width: 768px) {
   .wrapper {
     display: flex;
   }
   .menu {
     flex: 1;
   }
   .description {
     flex: 2;
   }
   .content {
     flex: 3;
   }
}
```



```css
/* Styles for smartphone layout */
@media (max-width: 767px) {
  .wrapper {
    display: flex;
    flex-direction: column;
  }
  .menu {
    order: 1;
  }
  .description {
    order: 2;
  }
  .content {
    order: 3;
  }
}
```



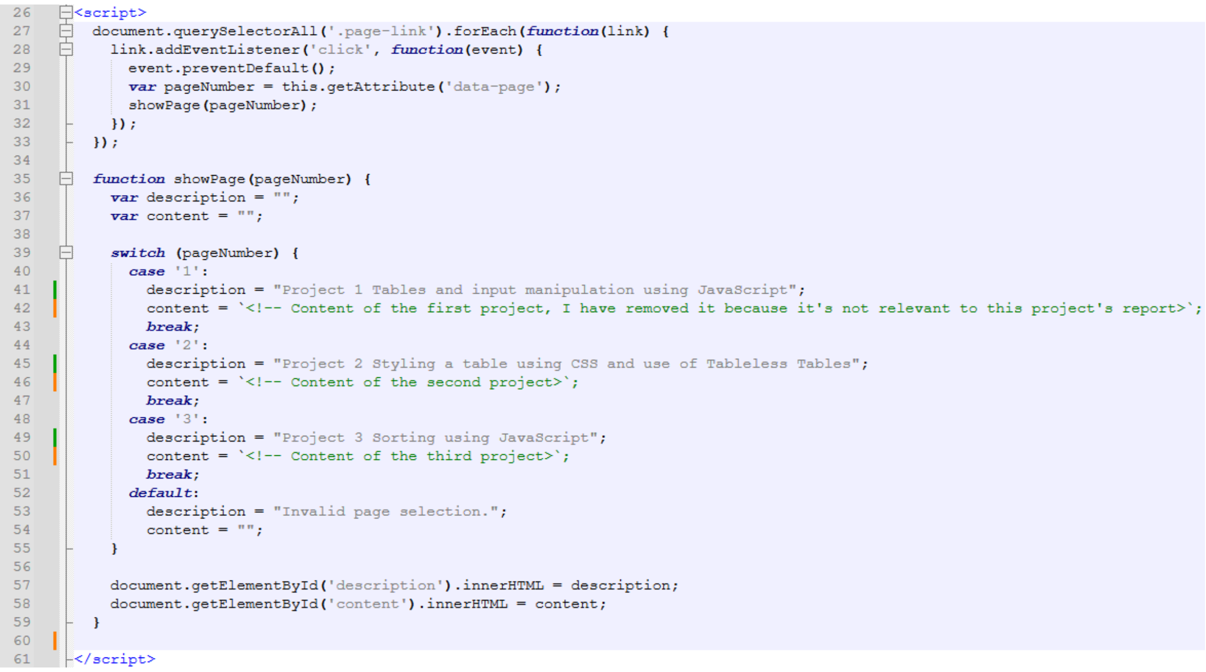
```css
/* Styles for Page 1 */
table,
th,
td {
  border: 1px solid black;
}

/* Styles for Page 2 */
.table {
  display: table;
  width: 50;
  border-collapse: collapse;
}

.row {
  display: table-row;
}

.cell {
  display: table-cell;
  padding: 5px 10px;
  border: 2px solid #ddd;
}

.cell:first-child {
  font-weight: bold;
}

input[type="text"],
select,
input[type="checkbox"],
input[type="radio"],
button {
  margin-bottom: 5px;
}
```



```css
/* Additional styles for enhanced appearance */
body {
  background-color: #f8f8f8;
  color: #333;
}

.wrapper {
  margin: 20px auto;
  max-width: 1200px;
  background-color: #fff;
  border-radius: 10px;
  box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
  padding: 20px;
  overflow: hidden;
  height: 600px; /* Set a fixed height */
}

.menu {
  background-color: #444;
}

.menu a {
  color: #fff;
}

.description {
  background-color: #ddd;
  padding: 10px;
  border-radius: 5px;
  margin-bottom: 20px;
}

.content {
  background-color: #f5f5f5;
  padding: 20px;
  border-radius: 5px;
  overflow-y: auto; /* Add this to enable scrolling if content exceeds height */
}
```

4.JavaScript:

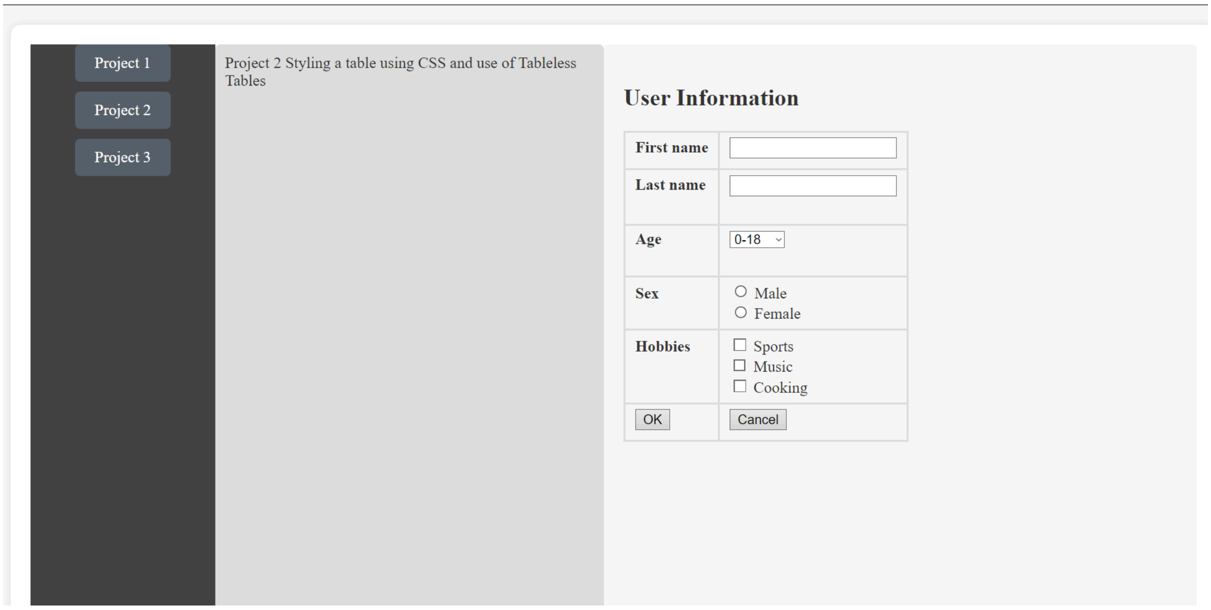
JavaScript code adds interactivity to the page.
It selects all elements with the class page-link and attaches a click event listener to each.
When a navigation link is clicked, it prevents the default link behavior (preventing page reload).
It retrieves the data-page attribute value of the clicked link, representing the selected page.
Based on the selected page number, the showPage function is called.
The showPage function takes the page number as a parameter and dynamically updates the description and content divs based on the selected page.
showPage Function:
This function updates the description and content based on the selected page number.
It uses a switch statement to determine the appropriate description and content for each page.
Each case corresponds to a specific page number, and it sets the description and content variables accordingly.
If an invalid page number is provided, it sets a default "Invalid page selection" message.



```
<script>
  document.querySelectorAll('.page-link').forEach(function(link) {
    link.addEventListener('click', function(event) {
      event.preventDefault();
      var pageNumber = this.getAttribute('data-page');
      showPage(pageNumber);
    });
  });

  function showPage(pageNumber) {
    var description = "";
    var content = "";

    switch (pageNumber) {
      case '1':
        description = "Project 1 Tables and input manipulation using JavaScript";
        content = `<!-- Content of the first project, I have removed it because it's not relevant to this project's report>`;
        break;
      case '2':
        description = "Project 2 Styling a table using CSS and use of Tableless Tables";
        content = `<!-- Content of the second project>`;
        break;
      case '3':
        description = "Project 3 Sorting using JavaScript";
        content = `<!-- Content of the third project>`;
        break;
      default:
        description = "Invalid page selection.";
        content = "";
    }

    document.getElementById('description').innerHTML = description;
    document.getElementById('content').innerHTML = content;
  }

</script>
```

5. The final project:

Desktop:

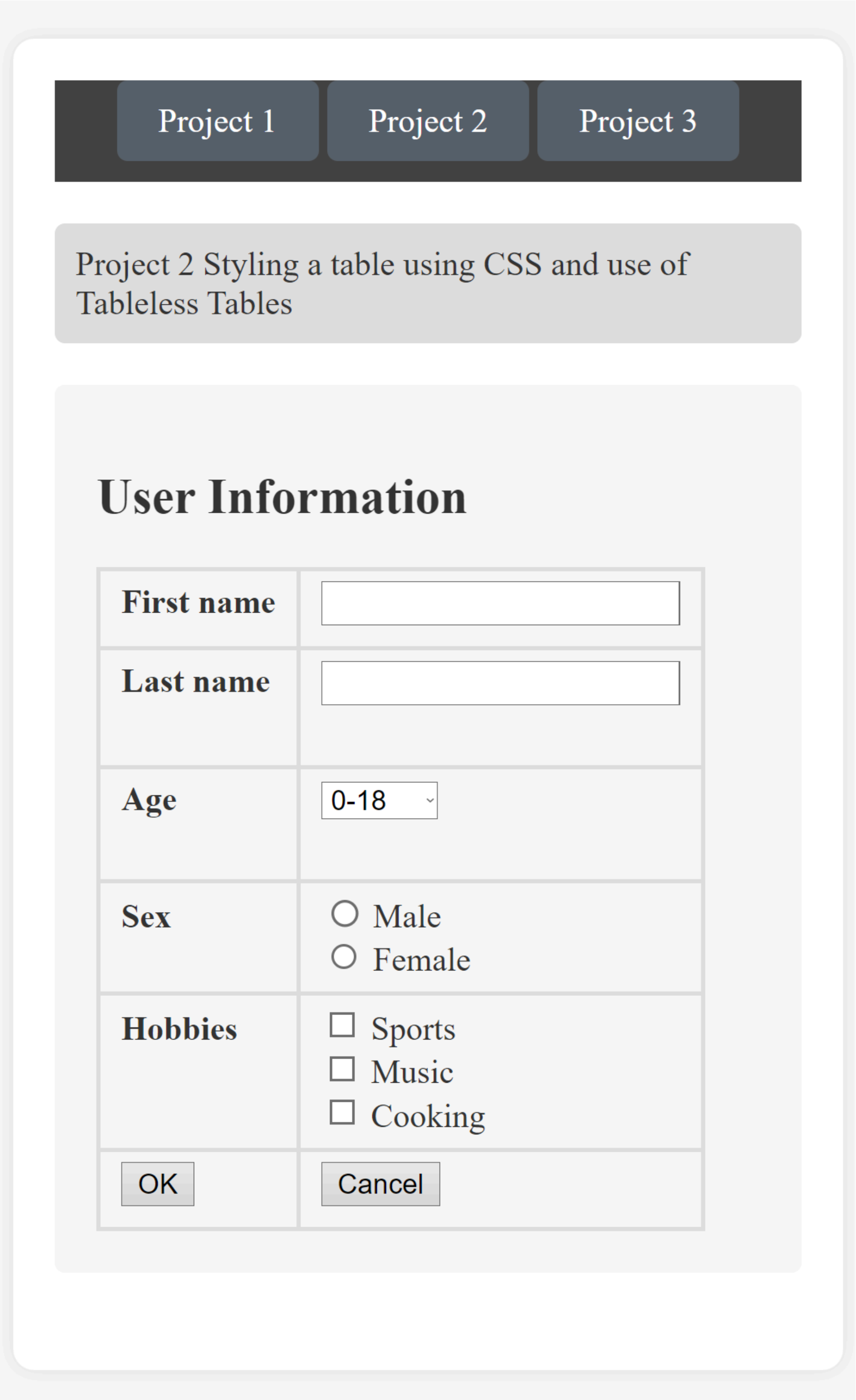


Project 2 Styling a table using CSS and use of Tableless Tables

User Information

First name	
Last name	
Age	0-18
Sex	○ Male ○ Female
Hobbies	□ Sports □ Music □ Cooking
OK	Cancel

Smartphone:



Project 2 Styling a table using CSS and use of Tableless Tables

User Information

First name	
Last name	
Age	0-18
Sex	○ Male ○ Female
Hobbies	□ Sports □ Music □ Cooking
OK	Cancel