



Front End Technologies Week 12 Coding Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

Instructions: In VS Code, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your JavaScript project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Coding Steps:

1. Create a full CRUD application of your choice. If you can use an existing API, use AJAX to interact with it. However, you do not have to use an API. If you do not use an API, store the entities you will create, read, update, and delete in an array.
 - a. Use a form to add new entities
 - b. Build a way for users to update or delete entities
 - c. Use Bootstrap and CSS to style your project

Screenshots of Code:



PROMINEO TECH

```
7   <title>Project Manager</title>
8   <link rel="stylesheet" href="style.css">
9   <link rel="stylesheet" href="node_modules/bootstrap/dist/css/bootstrap.css">
10  </head>
11  <body class="container">
12    <!-- Remove Project Confirmation Modal -->
13    <div class="modal" id="removeModal" tabindex="-1" role="dialog">
14      <div class="modal-dialog" role="document">
15        <div class="modal-content">
16          <div class="modal-header">
17            <h5 class="modal-title">Remove Project</h5>
18            <button type="button" class="close" data-bs-dismiss="modal" aria-label="Close">
19              <span aria-hidden="true">&times;</span>
20            </button>
21          </div>
22          <div class="modal-body">
23            <p>Are you sure you want to remove this project?</p>
24          </div>
25          <div class="modal-footer">
26            <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
27            <button type="button" class="btn btn-danger" id="confirmRemove" data-bs-dismiss="modal">Remove</button>
28          </div>
29        </div>
30      </div>
31    </div>
32    <!-- Create new project -->
33    <div id="new-project">
34      <h3>Project Manager</h3>
35      <div class="row">
36        <div class="col-sm">
37          <input type="text" id="project-name" class="form-control" placeholder="Project Name">
38        </div>
39        <div class="col-sm">
40          <input type="text" id="project-lead" class="form-control" placeholder="Project Lead">
41        </div>
42        <div class="col-sm">
43          <input type="date" id="project-date" class="form-control">
44        </div>
45      </div>
46      <button id="new-project-button" class="btn btn-success form-control mt-1">Add Project</button>
47    </div>
48    <!-- Project display -->
49    <div id="currentProjectDisplay"></div>
50    <div id="completedProjectDisplay"></div>
51    <script src="node_modules/bootstrap/dist/js/bootstrap.js"></script>
52    <script src="node_modules/jquery/dist/jquery.js"></script>
53    <script src="index.js"></script>
54  </body>
55 </html>
```



PROMINEO TECH

```
1 // Project class, takes a Project name, Project Lead, and Due date
2 // Members array is used to store members working on the project
3 // complete variable is used to set a project to complete
4
5 class Project {
6     constructor(name, lead, date) {
7         this.name = name
8         this.lead = lead
9         this.members = []
10        this.date = date
11        this.complete = false
12    }
13 }
14
15 //Member class, takes a name, used for creating new members for a project
16
17 class Member {
18     constructor(name) {
19         this.name = name
20     }
21 }
22
23 // The API class hosted at crudcrud.com
24 // getAllProjects() returns all of the data at the URL
25 // getProject() returns a single project using the ID of that project
26 // createProject() sends an AJAX POST to the URL updating with new project data
27 // deleteProject() send an AJAX DELETE to the URL removing a project by ID
28 // updateProject() sends an AJAX PUT to the URL updating a specific project by ID
29
30 class API {
31     static url = 'https://crudcrud.com/api/b26210a0b46f4379a978d71992a56408/unicorns'
32
33     static getAllProjects() {
34         return $.get(this.url)
35     }
36
37     static getProject(id) {
38         return $.get(`${this.url}/${id}`)
39     }
40
41     static createProject(project) {
42         console.log(project)
43         return $.ajax({
44             url: this.url,
45             dataType: 'json',
46             data: JSON.stringify(project),
47             contentType: 'application/json',
48             type: 'POST'
49         })
50     }
51 }
```



PROMINEO TECH

```
52  static deleteProject(id) {
53      return $.ajax({
54          url: `${this.url}/${id}`,
55          type: 'DELETE'
56      })
57  }
58
59  static updateProject(project) {
60      let url = `${this.url}/${project._id}`
61      delete project._id
62      return $.ajax({
63          url: url,
64          data: JSON.stringify(project),
65          contentType: 'application/json',
66          type: 'PUT',
67      })
68  }
69  }
70
71  // ProjectManager class handles the rendering of the DOM
72  // render() takes a param of projects and then uses for of loop to iterate through the data and append each project to the DOM
73  // Projects that are completed == false are rendered into the currentProjectDisplay
74  // Projects that are completed == true are rendered into the completedProjectDisplay
75  // getAllProjects() calls the API then uses the data and calls the render() function
76  // newProject() takes params of Name, Lead, and Date from the inputs and created a new project, sends it to the API, then renders the updated DOM
77  // removeProject() calls the API delete method passing in an ID then renders the updated DOM
78  // addTeamMember() adds a new team member to the project.members array then send an update call to the API then renders the updated DOM
79  // deleteTeamMember() removed a team member using project ID and member name, searching through projects to find the correct project then looks for the matching name in the array
80  // completeProject() marks a project completing using project ID, moving the competed projected to the completedProjectDisplay
81
82  class ProjectManager {
83
84      static projects
85
86      static getAllProjects() {
87          API.getAllProjects()
88              .then((projects) => this.render(projects))
89      }
90  }
```



PROMINEO TECH

```
static newProject(name, lead, date) {  
  API.createProject(new Project(name, lead, date))  
    .then(() => {  
      return API.getAllProjects()  
    })  
    .then((projects) => this.render(projects))  
}  
  
static removeProject(id) {  
  API.deleteProject(id)  
    .then(() => {  
      return API.getAllProjects()  
    })  
    .then((projects) => this.render(projects))  
}  
  
static addTeamMember(id) {  
  for (let project of this.projects) {  
    if ($('#${project._id}-member-name').val() !== '') {  
      if (id === project._id) {  
        project.members.push(new Member($('#${project._id}-member-name').val()))  
        API.updateProject(project)  
          .then(() => {  
            return API.getAllProjects()  
          })  
          .then((projects) => this.render(projects))  
      }  
    }  
  }  
}
```



PROMINEO TECH

```
static deleteTeamMember(id, name) {
  console.log('Running delete team member', id, name)
  for (const project of this.projects) {
    if (id === project._id) {
      console.log('Found matching project')
      for (const member of project.members) {
        if (member.name === name) {
          console.log('found matching name')
          project.members.splice(project.members.indexOf(member), 1)
          API.updateProject(project)
            .then(() => {
              return API.getAllProjects()
            })
            .then((projects) => this.render(projects))
        }
      }
    }
  }
}

static completeProject(id) {
  for (const project of this.projects) {
    if (id === project._id) {
      project.complete = true
      $('#${id}`).remove()
      API.updateProject(project)
        .then(() => {
          return API.getAllProjects()
        })
        .then((projects) => this.render(projects))
    }
  }
}
```



PROMINEO TECH

```
156 static render/projects) {
157   this.projects = projects
158   projects.sort((a, b) => (a.date > b.date) ? 1 : -1)
159   $('#currentProjectDisplay').empty()
160   $('#completedProjectDisplay').empty()
161   for (const project of projects) {
162     console.log(project)
163     if (project.complete === false) {
164       $('#currentProjectDisplay').append(
165         `<br>
166         <div id=${project._id}>
167           <div>
168             <div class="row shadow pb-3 border">
169               <div class="col">
170                 <h2>${project.name}</h2>
171               </div>
172               <div class="col pt-1">
173                 <button class="btn btn-success" onclick="ProjectManager.completeProject('${project._id}')">Complete Project</button>
174                 <button class="btn btn-danger" data-bs-toggle="modal" data-bs-target="#RemoveModal">Remove Project</button>
175               </div>
176             </div>
177           </div>
178           <div>
179             Project Lead: ${project.lead} <br>
180             Project Due Date: ${project.date}
181           </div>
182           <div class="row">
183             <div id="team-members">
184               <h6>Team Members:</h6>
185             </div>
186             <div class="col-3">
187               <input type="text" id="${project._id}-member-name" onkeypress="return enterKeyPressed(event)" placeholder="Team Member">
188               <button type="submit" class="btn btn-sm btn-success mt-1" onclick="ProjectManager.addTeamMember('${project._id}')">Add Team Member</button>
189             </div>
190           </div>
191         </div>
192       `
193     )
194   }
195 }
```

```
194   for (const member of project.members) {
195     $('#${project._id}').find('#team-members').append(
196       `<p> <button class="btn btn-danger btn-sm" onclick="ProjectManager.deleteTeamMember('${project._id}', '${member.name}')"></button>${member.name}</p>
197     `
198   )
199   }
200   } else if (project.complete === true) {
201     $('#completedProjectDisplay').append(
202       `<br>
203       <div id=${project._id} class="completed">
204         <div>
205           <div class="row shadow pb-3 border">
206             <div class="col">
207               <h2>${project.name}</h2>
208             </div>
209             <div class="col pt-1 text-center fw-bold text-success">
210               PROJECT COMPLETED
211             </div>
212           </div>
213         </div>
214         <div>
215           Project Lead: ${project.lead} <br>
216           Project Due Date: ${project.date}
217         </div>
218         <div class="row">
219           <div id="team-members">
220             <h6>Team Members:</h6>
221           </div>
222           <div class="col-3">
223             <button class="btn btn-sm btn-danger mt-1" onclick="ProjectManager.removeProject('${project._id}')">Remove Project</button>
224           </div>
225         </div>
226       `
227     )
228   }
229   }
230   for (const member of project.members) {
231     $('#${project._id}').find('#team-members').append(
232       `<p>${member.name}</p>
233     `
234   )
235   }
236   }
237   }
238 }
```



```
1 .completed{
2     opacity: 0.4;
3     pointer-events: none;
4 }
5
```

Screenshots of Running Application:



PROMINEO TECH

Project Manager

Add Project

Week 12 Project

Complete Project

Remove Project

Project Lead: Nathan

Project Due Date: 2022-08-20

Team Members:

 me

Add Team Member

Week 13 assignment

Complete Project

Remove Project

Project Lead: Nathan

Project Due Date: 2022-08-27

Team Members:

Add Team Member

Week 11 Assignment

PROJECT COMPLETED

Project Lead: Nathan

Project Due Date: 2022-08-13

Team Members:

Remove Project



PROMINEO TECH

Project Manager

Week 12 Project

Project Lead: Nathan
Project Due Date: 2022-08-20
Team Members:

me

Add Team Member

Week 13 assignment

Project Lead: Nathan
Project Due Date: 2022-08-27
Team Members:

Add Team Member

Complete Project

Remove Project

Week 11 Assignment

Project Lead: Nathan
Project Due Date: 2022-08-13
Team Members:

Remove Project

PROJECT COMPLETED

Remove Project

Are you sure you want to remove this project?

Close

Remove

URL to GitHub Repository: <https://github.com/n-harvey/Week12Project>