

The background of the slide is a blurred photograph of several soldiers in military uniforms. They are gathered in a huddle, with their arms around each other, suggesting a moment of teamwork or strategy discussion. The focus is on the soldiers in the foreground, while the others in the background are out of focus.

# PMCS:

PRO MANAGEMENT SOLUTION

GitHub:  
*fpen3*

## Objective:

PMCS Pro Management Solution is aimed at making the process of PMCSing much more efficient and effective. PMCS Pro allows users to search for part numbers and procedures specific to each problem and help cut down time searching through TMs[Technical Manual] to find one specific part or deficiency. PMCS Pro also filing, tracking and copying 2404s or 1801s much more easily though database storage

# System Objectives:

- Preferably there are four pages, Home, Technical Manuals, Login and Admin Login.
  - Also hidden pages for the account request form, and of course the actual database access page. ✓
- The SQL database ~~will store old PMCS forms~~ ~ maybe, different items and there statuses and individual item breakdown. ✓
- Some features may not get covered.
- The primary focus for this project is to be able view item parts and statuses. ✓

# System Objectives (Continued):

## Users:

Users can -

- ~~View old PMGS forms. ~ maybe~~
- ~~Download old PMGS forms. ~ maybe~~
- ~~Download blank PMGS forms. ~ maybe~~
- View item numbers, quantities and statuses. ✓

## Administrators:

Admins can -

- ~~View old PMGS forms. ~ maybe~~
- ~~Download old PMGS forms. ~ maybe~~
- ~~Download blank PMGS forms. ~ maybe~~
- ~~Upload new PMGS forms. ~ maybe~~
- ~~Delete PMGS forms. ~ maybe~~
- View item numbers, quantities and statuses. ✓
- Modify item numbers, quantities and statuses. ✓

# Phases:

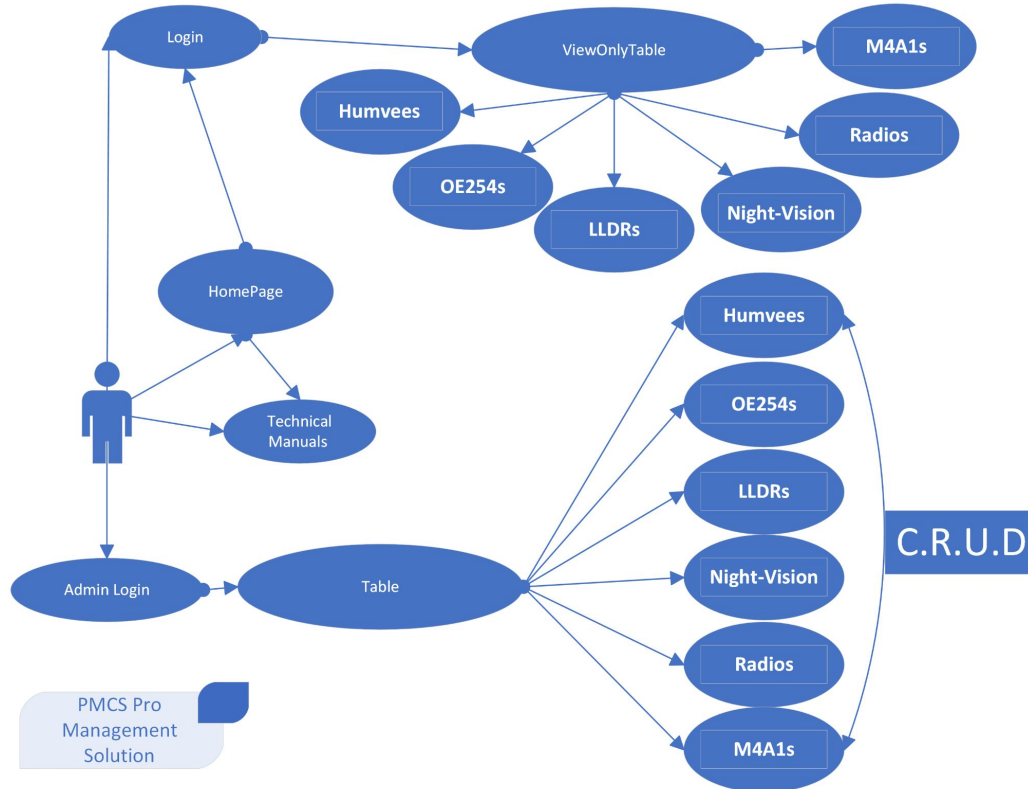
**PHASE #1:)** Design a mockup website. Get pages and presentation setup first. ✓

**PHASE #2:)** Connect a database. Start testing out functionalities. ✓

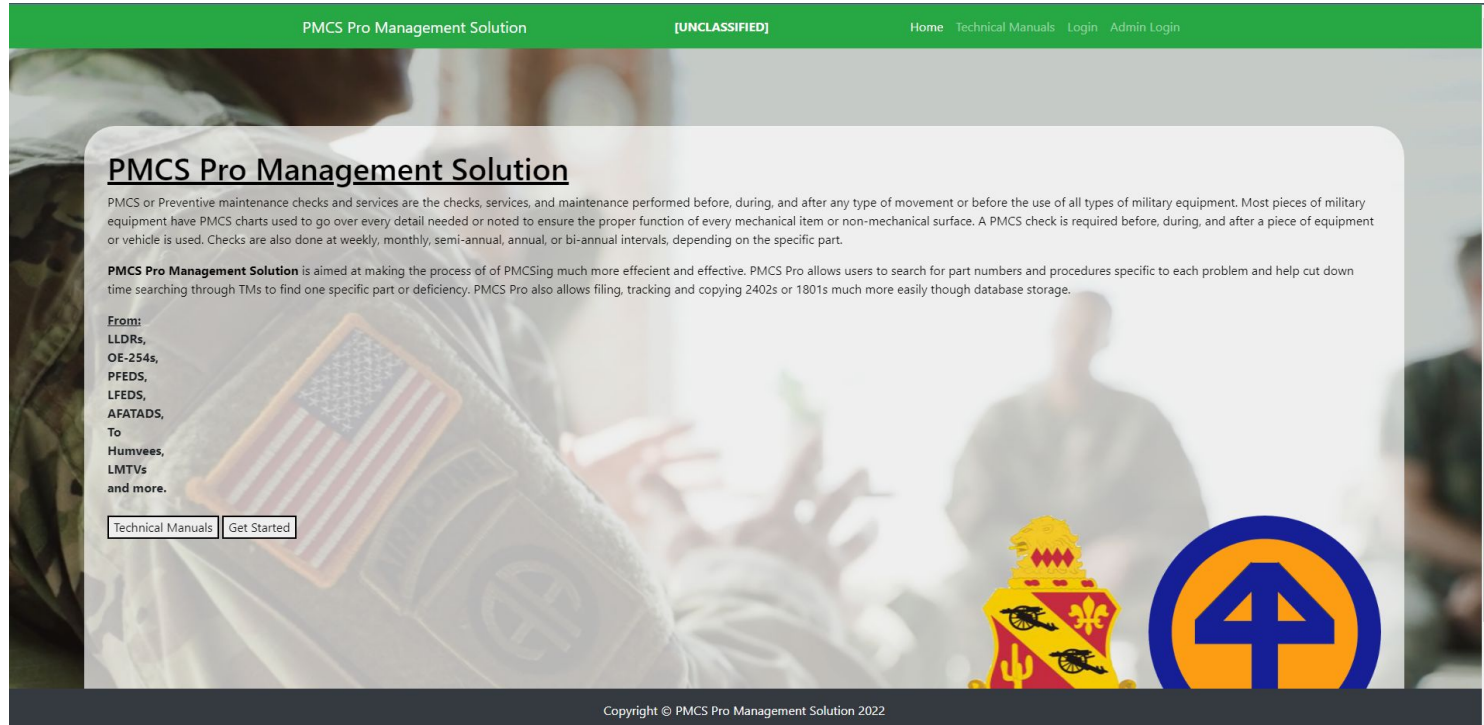
**PHASE #3:)** More complex feature implementation. Modifying and viewing tables can be accomplished ✓

~~and possible file download/upload system.~~

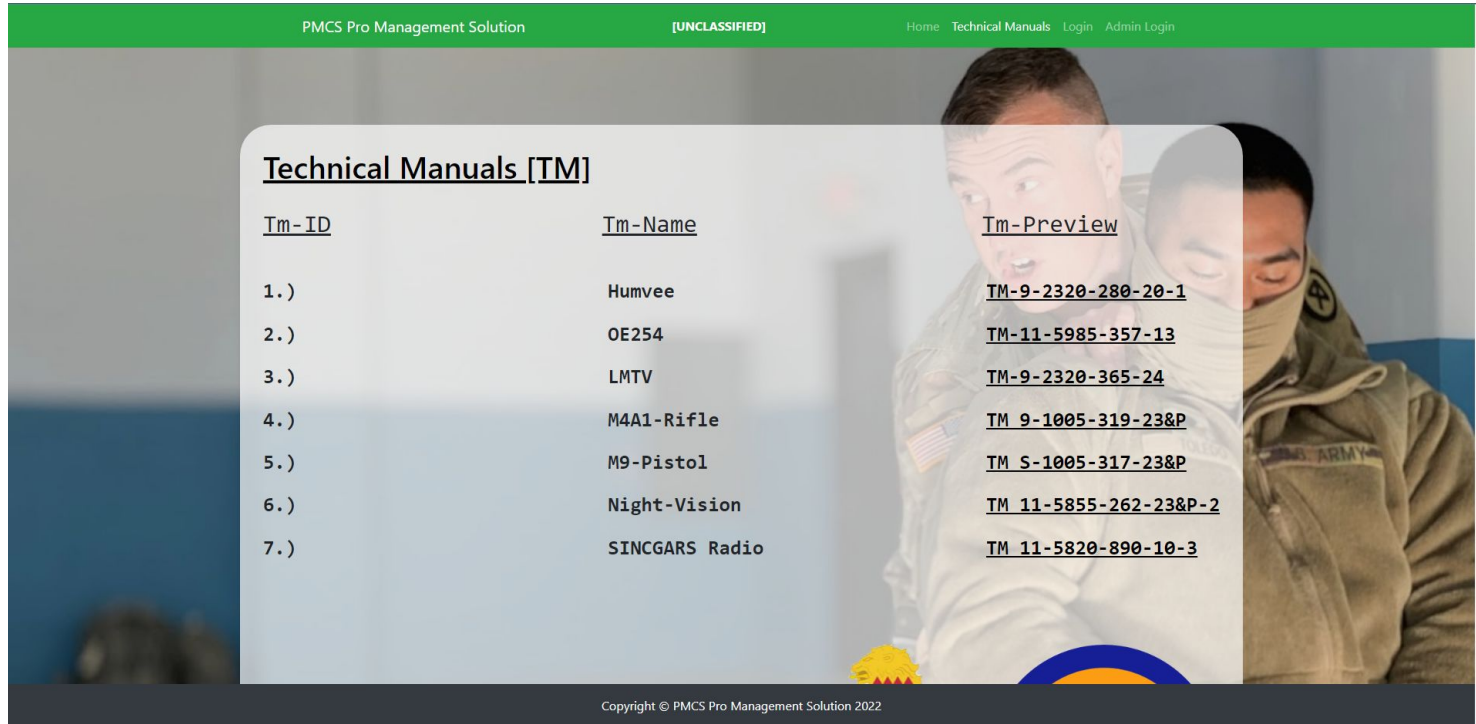
# Use Case Diagram:



# Screenshots [Old]:



# Screenshots [New]:



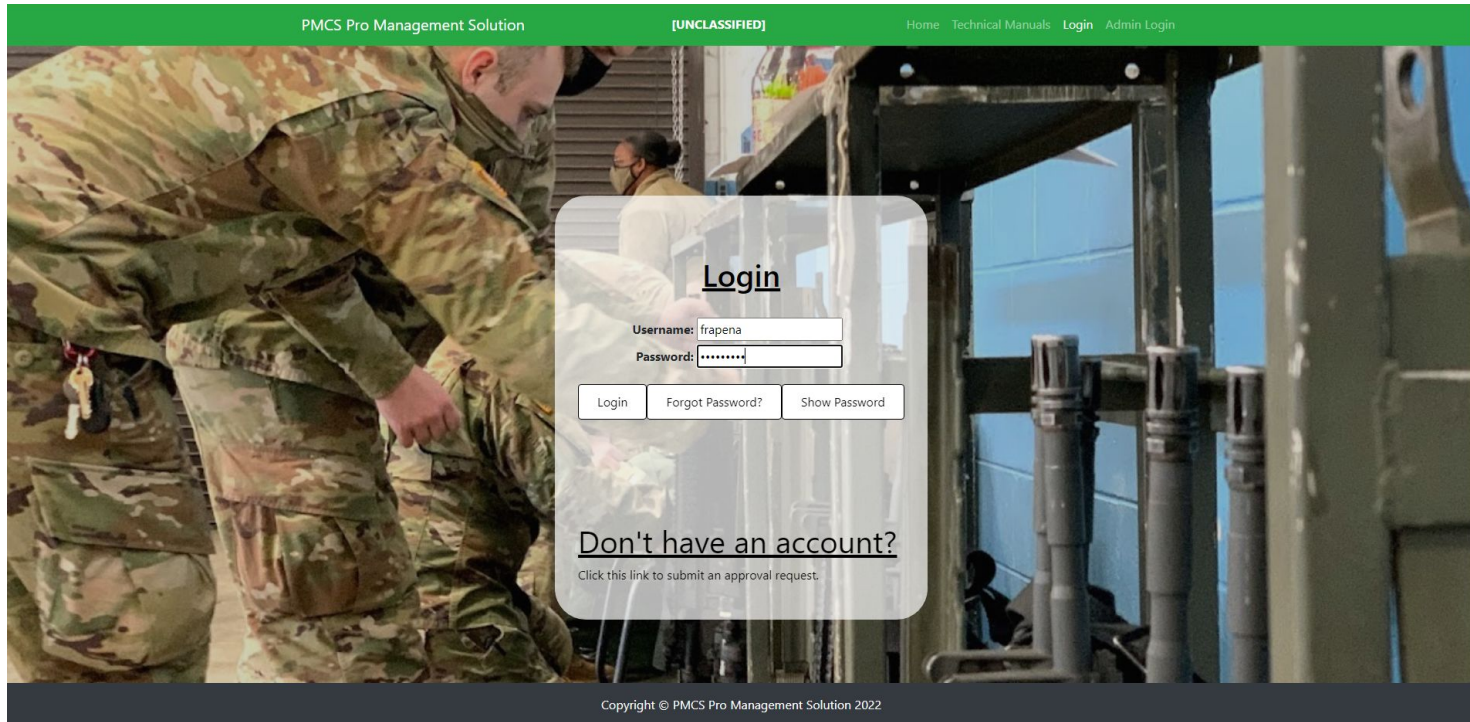
The screenshot displays the PMCS Pro Management Solution web application. The interface features a green header bar with the text "PMCS Pro Management Solution", "[UNCLASSIFIED]", and navigation links for "Home", "Technical Manuals", "Login", and "Admin Login". The main content area is a semi-transparent white box overlaid on a background image of two soldiers in military uniforms. Inside this box, the section is titled "Technical Manuals [TM]". Below the title is a table with three columns: "Tm-ID", "Tm-Name", and "Tm-Preview". The table lists seven technical manuals, each with a sequential number in the ID column, a name in the Name column, and a corresponding TM number in the Preview column.

| <u>Tm-ID</u> | <u>Tm-Name</u> | <u>Tm-Preview</u>                |
|--------------|----------------|----------------------------------|
| 1.)          | Humvee         | <u>TM-9-2320-280-20-1</u>        |
| 2.)          | OE254          | <u>TM-11-5985-357-13</u>         |
| 3.)          | LMTV           | <u>TM-9-2320-365-24</u>          |
| 4.)          | M4A1-Rifle     | <u>TM 9-1005-319-23&amp;P</u>    |
| 5.)          | M9-Pistol      | <u>TM S-1005-317-23&amp;P</u>    |
| 6.)          | Night-Vision   | <u>TM 11-5855-262-23&amp;P-2</u> |
| 7.)          | SINGGARS Radio | <u>TM 11-5820-890-10-3</u>       |

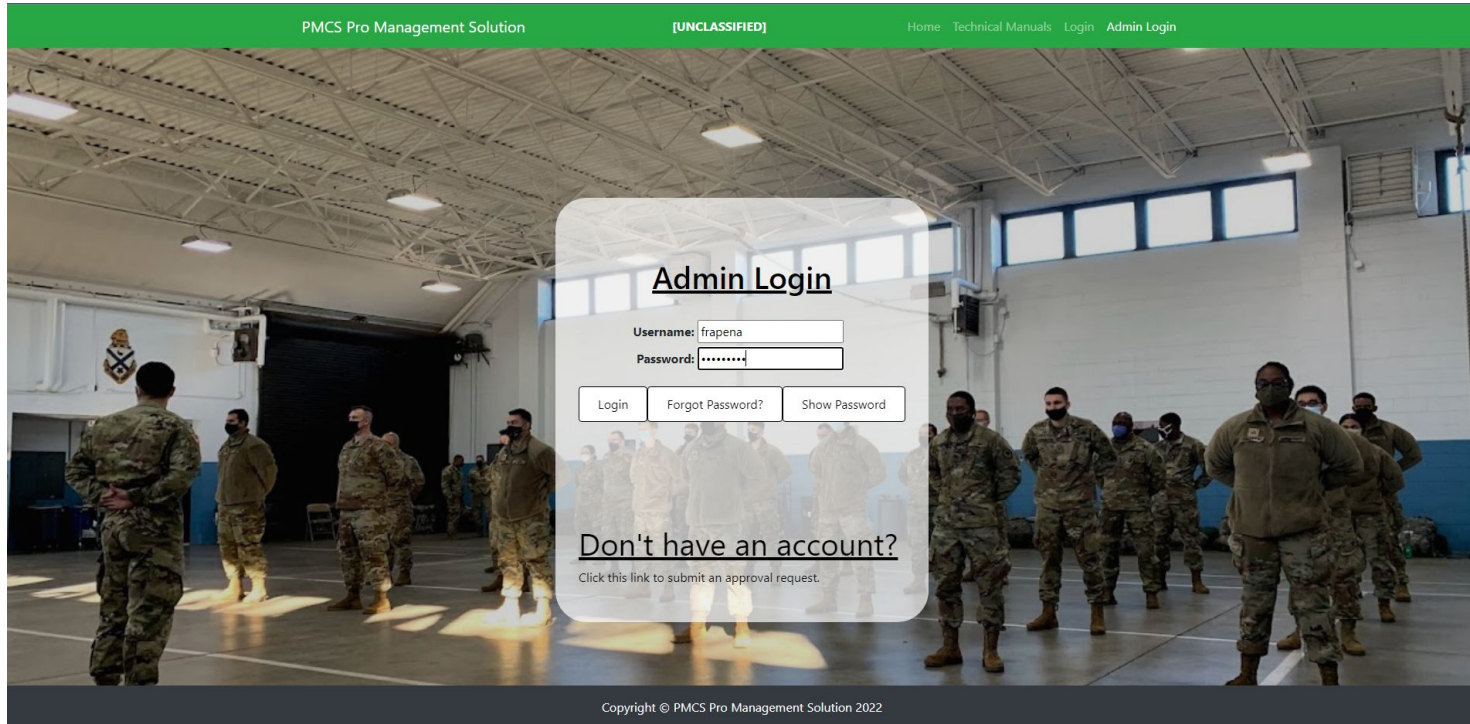
Copyright © PMCS Pro Management Solution 2022



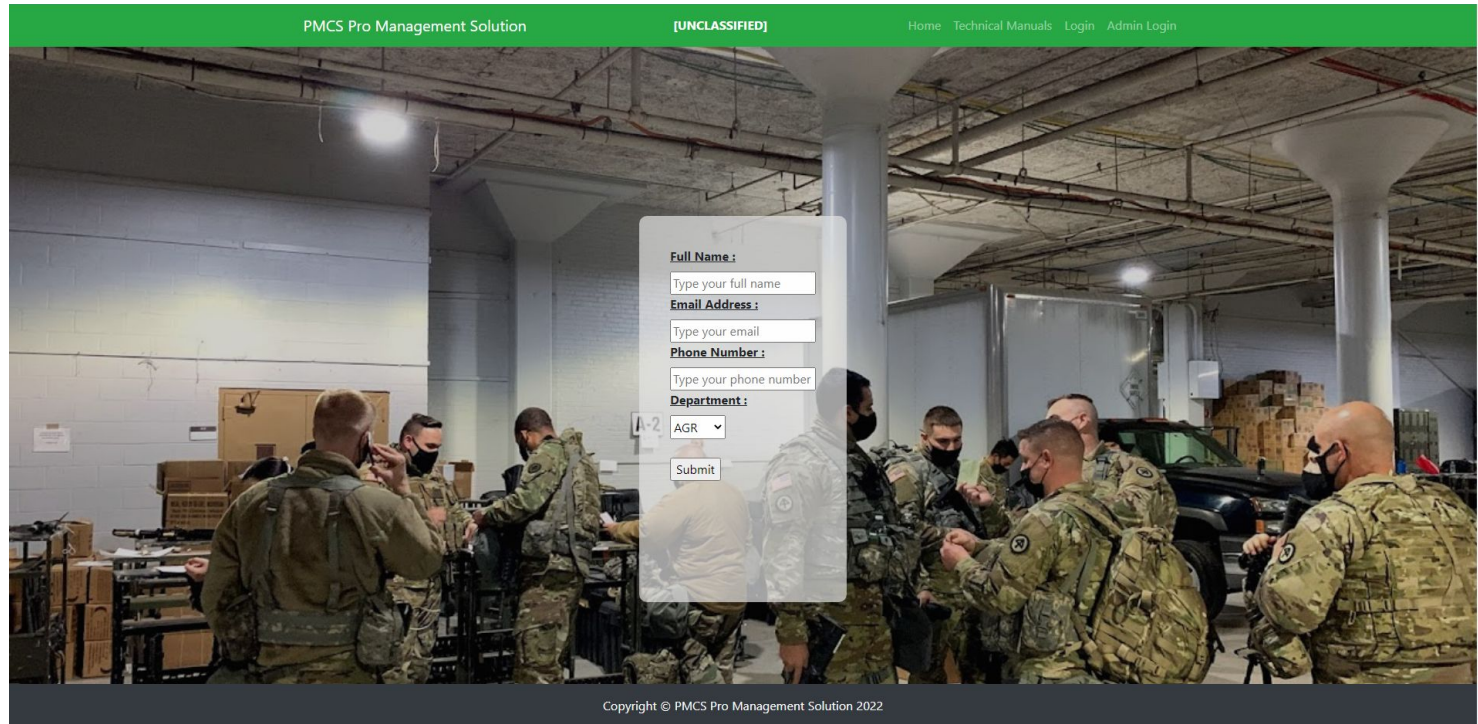
# Screenshots [Old]:



# Screenshots [Old]:

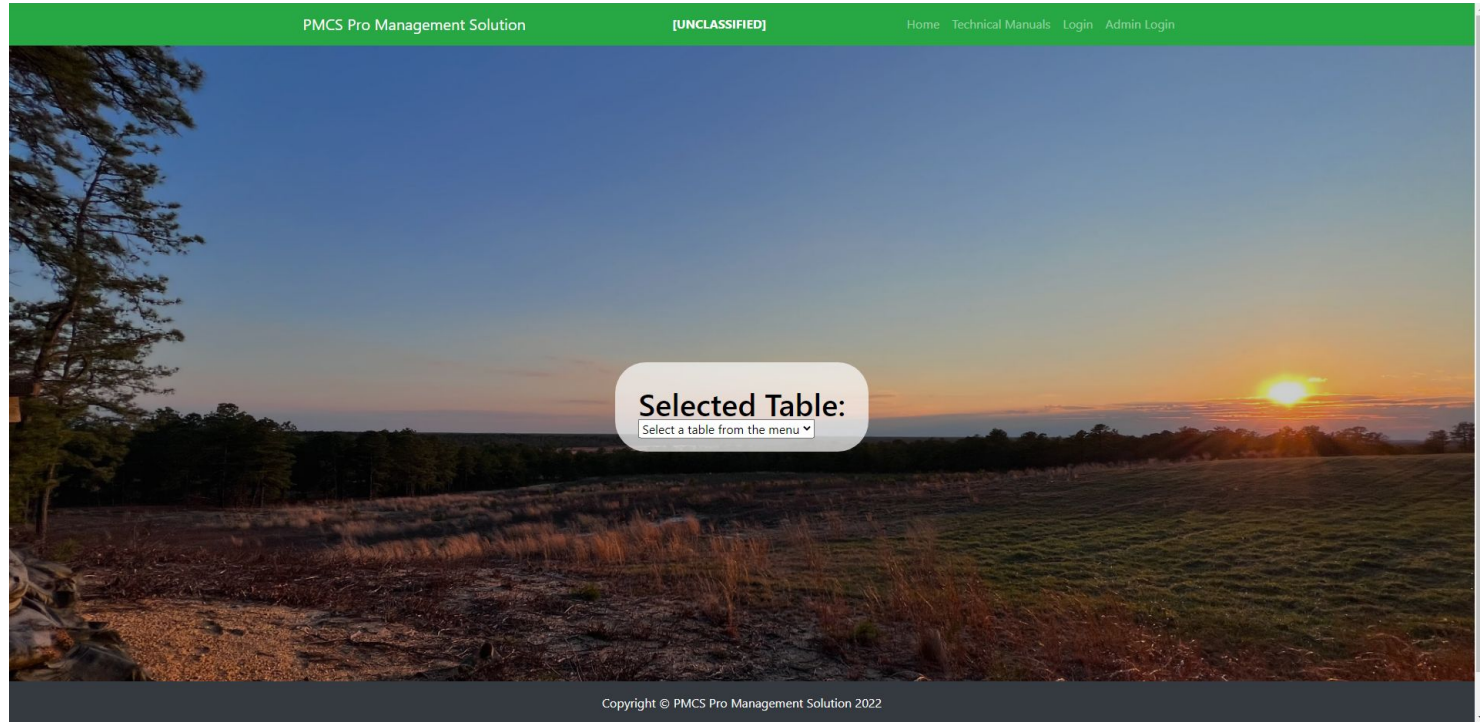


# Screenshots [Old]:





# Screenshots (Table) [New]:



# Screenshots (Table) [New]:

PMCS Pro Management Solution

[UNCLASSIFIED]

[Home](#) [Technical Manuals](#) [Login](#) [Admin Login](#)

Selected Table: HUMVEES AND LMTVS

HUMVEES AND LMTVS

Insert into Humvees

Humvee Number:

Humvee Status:

Submit

Humvee List

| Delete                                | Update Status  | Humvee ID | Humve Number | Humvee Status     |
|---------------------------------------|--|-----------|--------------|-------------------|
| <input type="button" value="Delete"/> | [Status: <input type="text"/> <input type="button" value="T"/> | 21        | 153          | OPERATIONAL       |
| <input type="button" value="Delete"/> | [Status: <input type="text"/> <input type="button" value="T"/> | 22        | 156          | DEAD-LINED (INOP) |

Copyright © PMCS Pro Management Solution 2022

# Screenshots (Table) [New]:

PMCS Pro Management Solution

[UNCLASSIFIED]

[Home](#) [Technical Manuals](#) [Login](#) [Admin Login](#)

Selected Table: OE254

OE-254

Insert into OE254s

OE254 Number:

Platoon:

Submit

OE254 List

| Modify                                | Update Number                            | Update Platoon                            | ID | Number | Platoon |
|---------------------------------------|--|---|----|--------|---------|
| <input type="button" value="Delete"/> | [Number:] <input type="text" value="1"/> | [Platoon:] <input type="text" value="1"/> | 7  | 28     | 2nd     |
| <input type="button" value="Delete"/> | [Number:] <input type="text" value="1"/> | [Platoon:] <input type="text" value="1"/> | 9  | 10     | 3rd     |

Copyright © PMCS Pro Management Solution 2022

# Screenshots (Table) [New]:

PMCS Pro Management Solution

[UNCLASSIFIED]

[Home](#) [Technical Manuals](#) [Login](#) [Admin Login](#)

Selected Table: LLDRS

LLDRS

Insert into LLDRs

LLDR Number:

Platoon:

Submit

LLDR List

| Delete                                | Update Number   | Update Platoon   | LLDR ID | LLDR Number | Platoon |
|---------------------------------------|---|--|---------|-------------|---------|
| <input type="button" value="Delete"/> | (Number:) <input type="text"/> <input type="checkbox"/> | (Platoon:) <input type="text"/> <input type="checkbox"/> | 7       | TEST        | TEST    |
| <input type="button" value="Delete"/> | (Number:) <input type="text"/> <input type="checkbox"/> | (Platoon:) <input type="text"/> <input type="checkbox"/> | 9       | 82          | 3rd     |
| <input type="button" value="Delete"/> | (Number:) <input type="text"/> <input type="checkbox"/> | (Platoon:) <input type="text"/> <input type="checkbox"/> | 16      | 13          | 1st     |

Copyright © PMCS Pro Management Solution 2022



# Screenshots (Table) [New]:

PMCS Pro Management Solution

[UNCLASSIFIED]

Home Technical Manuals Login Admin Login

Selected Table: NODS

NIGHT-VISION

Insert into Night-Vision

Nods Number:

Platoon:

Owner:

Submit

Night-Vision List

| Delete | Update Number                 | Update Platoon                 | Update Owner                 | NODS ID | NODS Number | Platoon | Owner         |
|--------|-------------------------------|--------------------------------|------------------------------|---------|-------------|---------|---------------|
| Delete | [Number: <input type="text"/> | [Platoon: <input type="text"/> | [Owner: <input type="text"/> | 11      | 1           | 4th     | SGT Pena      |
| Delete | [Number: <input type="text"/> | [Platoon: <input type="text"/> | [Owner: <input type="text"/> | 13      | 2           | 4th     | SSG Mishyn    |
| Delete | [Number: <input type="text"/> | [Platoon: <input type="text"/> | [Owner: <input type="text"/> | 14      | 3           | 4th     | SSG Wong      |
| Delete | [Number: <input type="text"/> | [Platoon: <input type="text"/> | [Owner: <input type="text"/> | 15      | 4           | 4th     | SSG Minituolo |
| Delete | [Number: <input type="text"/> | [Platoon: <input type="text"/> | [Owner: <input type="text"/> | 16      | 5           | 4th     | SSG Watson    |
| Delete | [Number: <input type="text"/> | [Platoon: <input type="text"/> | [Owner: <input type="text"/> | 17      | 6           | 4th     | SSG Smith     |

Copyright © PMCS Pro Management Solution 2022



# Screenshots (Table) [New]:

PMCS Pro Management Solution

[UNCLASSIFIED]

[Home](#) [Technical Manuals](#) [Login](#) [Admin Login](#)

Selected Table: RADIOS

RADIOS

Insert into Radios

Radio Number:

Platoon:

Submit

Radios List

| Delete            | Update Number                                  | Update Platoon | Radio ID | Radio Number | Platoon |
|-------------------|--|----------------|----------|--------------|---------|
| <div>Delete</div> | [Number:] <div>1</div> [Platoon:] <div>1</div> |                | 6        | 14           | 3rd     |
| <div>Delete</div> | [Number:] <div>1</div> [Platoon:] <div>1</div> |                | 7        | 8            | 2nd     |

Copyright © PMCS Pro Management Solution 2022

# Screenshots (Table) [New]:

PMCS Pro Management Solution

[UNCLASSIFIED]

[Home](#) [Technical Manuals](#) [Login](#) [Admin Login](#)

Selected Table: M4A1s

M4A1s

Insert into Rifles

Rifle Number:

Rifle Owner:

Submit

Rifle List

| Delete                                | Update Number   | Update Owner   | Rifle ID | Rifle Number | Rifle Owner |
|---------------------------------------|---|--|----------|--------------|-------------|
| <input type="button" value="Delete"/> | [Number:] <input type="text"/> <input type="checkbox"/> | [Owner:] <input type="text"/> <input type="checkbox"/> | 1        | FIST_060     | SGT Pena    |
| <input type="button" value="Delete"/> | [Number:] <input type="text"/> <input type="checkbox"/> | [Owner:] <input type="text"/> <input type="checkbox"/> | 2        | FIST_067     | SPC Runza   |
| <input type="button" value="Delete"/> | [Number:] <input type="text"/> <input type="checkbox"/> | [Owner:] <input type="text"/> <input type="checkbox"/> | 3        | FIST_054     | SPC Rauth   |
| <input type="button" value="Delete"/> | [Number:] <input type="text"/> <input type="checkbox"/> | [Owner:] <input type="text"/> <input type="checkbox"/> | 4        | FIST_023     | SPC Juarez  |

Copyright © PMCS Pro Management Solution 2022

# Screenshots (ViewOnlyTable) [New]:

The screenshot displays the PMCS Pro Management Solution web application. The header is green and contains the text "PMCS Pro Management Solution", "[UNCLASSIFIED]", and navigation links: "Home", "Technical Manuals", "Login", and "Admin Login". The main content area features a large, semi-transparent white box with rounded corners. Inside this box, the text "Selected Table: NODS" is displayed above a dropdown menu showing "NIGHT-VISION". Below this, the title "Night-Vision List" is centered. A table with four columns is presented: "Night-Vision ID", "Night-Vision Number", "Platoon", and "Owner". The table contains six rows of data. The background of the application is a landscape image showing a sunset or sunrise over a field.

PMCS Pro Management Solution [UNCLASSIFIED] Home Technical Manuals Login Admin Login

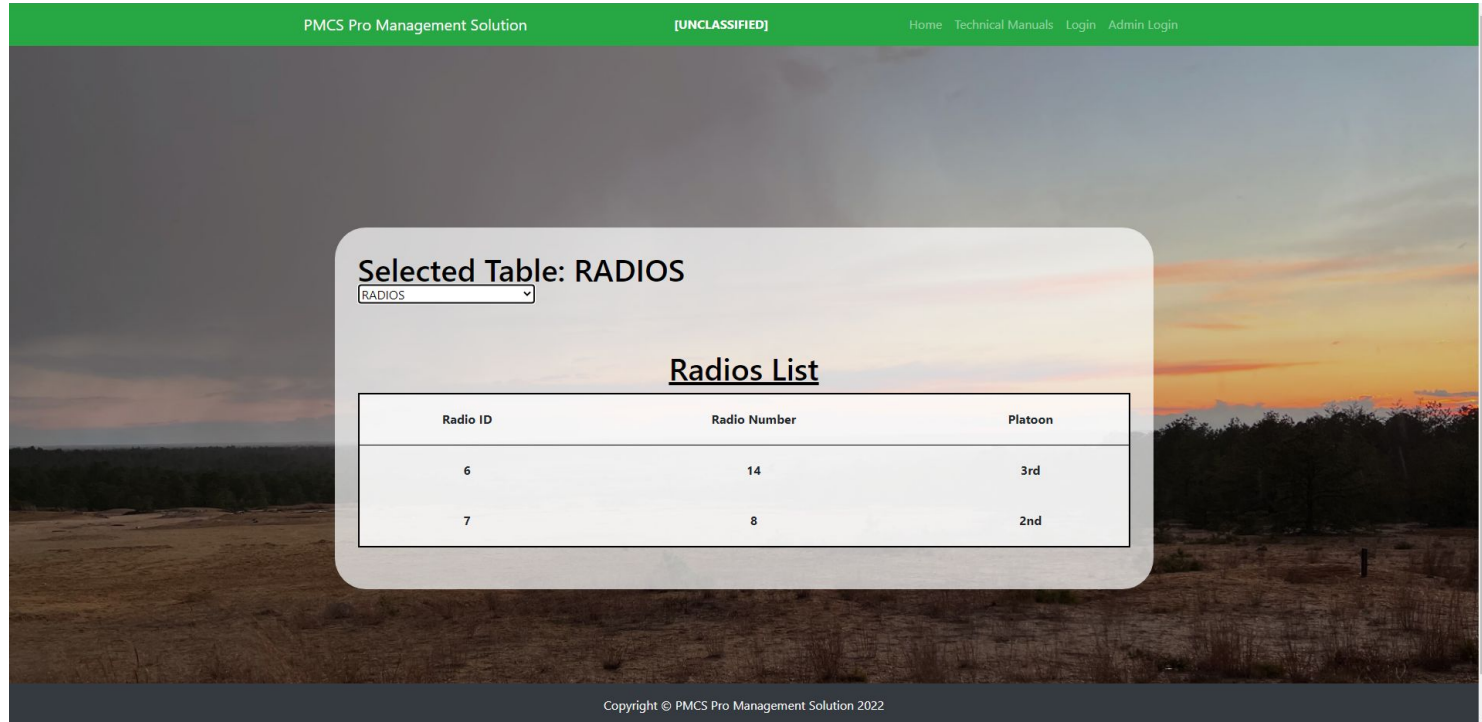
Selected Table: NODS  
NIGHT-VISION

Night-Vision List

| Night-Vision ID | Night-Vision Number | Platoon | Owner         |
|-----------------|---------------------|---------|---------------|
| 11              | 1                   | 4th     | SGT Pena      |
| 13              | 2                   | 4th     | SSG Mishyn    |
| 14              | 3                   | 4th     | SSG Wong      |
| 15              | 4                   | 4th     | SSG Minituolo |
| 16              | 5                   | 4th     | SSG Watson    |
| 17              | 6                   | 4th     | SSG Smith     |

Copyright © PMCS Pro Management Solution 2022

# Screenshots (ViewOnlyTable) [New]:



The screenshot displays the PMCS Pro Management Solution web interface. The top navigation bar is green and contains the text 'PMCS Pro Management Solution', '[UNCLASSIFIED]', and links for 'Home', 'Technical Manuals', 'Login', and 'Admin Login'. The main content area features a large, semi-transparent white card with rounded corners. Inside the card, the text 'Selected Table: RADIOS' is displayed above a dropdown menu currently showing 'RADIOS'. Below this, the title 'Radios List' is centered and underlined. A table with three columns is presented: 'Radio ID', 'Radio Number', and 'Platoon'. The table contains two data rows. The background of the interface is a landscape image showing a sunset or sunrise over a field.

| Radio ID | Radio Number | Platoon |
|----------|--------------|---------|
| 6        | 14           | 3rd     |
| 7        | 8            | 2nd     |

Copyright © PMCS Pro Management Solution 2022

# Screenshots [New]:

```
1 • use pmcsmanagement;
2 /*=====*/
3 /*-----USERS-----*/
4 • INSERT INTO Users (User_name, User_password, User_email, Owner_name) VALUES ('ozzyoct', 'password1', 'ozzyoct@gmail.com', 'Frank');
5 • INSERT INTO Users (User_name, User_password, User_email, Owner_name) VALUES ([REDACTED]);
6 • INSERT INTO Users (User_name, User_password, User_email, Owner_name) VALUES ('jasRodriguez', 'password3', 'MSG@44thICBT.mil@mail.mil', 'MSG Rodriguez');
7 • SELECT * FROM users;
8 /*-----USERS-----*/
9 /*=====*/
10
11 /*-----USERS-----*/
12
13 /*=====*/
14 /*-----ADMINS-----*/
15 • SELECT * FROM admins;
16 • INSERT INTO admins (Admin_name, Admin_password, Admin_email, Owner_name) VALUES ([REDACTED]);
17 /*-----ADMINS-----*/
18 /*=====*/
19
20 /*-----USERS-----*/
21
22 /*=====*/
23 /*-----TEMP-USERS-----*/
24 • SELECT * FROM tempusers;
25 /*-----TEMP-USERS-----*/
26 /*=====*/
27
28 /*-----USERS-----*/
29
30 /*=====*/
31 /*-----HUMVEES-----*/
32 • SELECT * FROM humveetest;
33
34 • DELETE FROM humveetest;
35
36 • ALTER TABLE humveetest AUTO_INCREMENT = 1;
37 /*-----HUMVEES-----*/
38 /*=====*/
39
40
```

# Screenshots [New]:

```
43  /*-----OE254s-----*/
44  • SELECT * FROM oe254;
45  • INSERT INTO oe254 (oe254_number, oe254_platoon) VALUES ("34", "4");
46
47  • ALTER TABLE oe254 AUTO_INCREMENT = 1;
48
49  • DELETE FROM oe254;
50  /*-----OE254s-----*/
51  /*=====*/
52
53  /*-----
54
55  /*=====*/
56  /*-----LLDRS-----*/
57  • SELECT * FROM lldrs;
58  • INSERT INTO lldrs(lldr_number, lldr_platoon) VALUES ("58", "4th");
59
60  • ALTER TABLE lldrs AUTO_INCREMENT = 1;
61
62  • DELETE FROM lldrs;
63  /*-----LLDRS-----*/
64  /*=====*/
65
66  /*-----
67
68  /*=====*/
69  /*-----NODS-----*/
70  • SELECT * FROM nods;
71  • INSERT INTO nods(nods_number, nods_platoon, nods_owner) VALUES ("3", "4th", "SGT WONG");
72
73  • ALTER TABLE nods AUTO_INCREMENT = 1;
74
75  • DELETE FROM nods;
76  /*-----NODS-----*/
77  /*=====*/
78
```

# Screenshots [New]:

```
/*=====*/  
/*-----RADIOS-----*/
```

- `SELECT * FROM radios;`
- `INSERT INTO radios(radius_number, radio_platoon) VALUES ("7", "4th");`
- `ALTER TABLE radios AUTO_INCREMENT = 1;`

```
DELETE FROM radios;  
/*-----RADIOS-----*/  
/*=====*/
```

```
/*-----
```

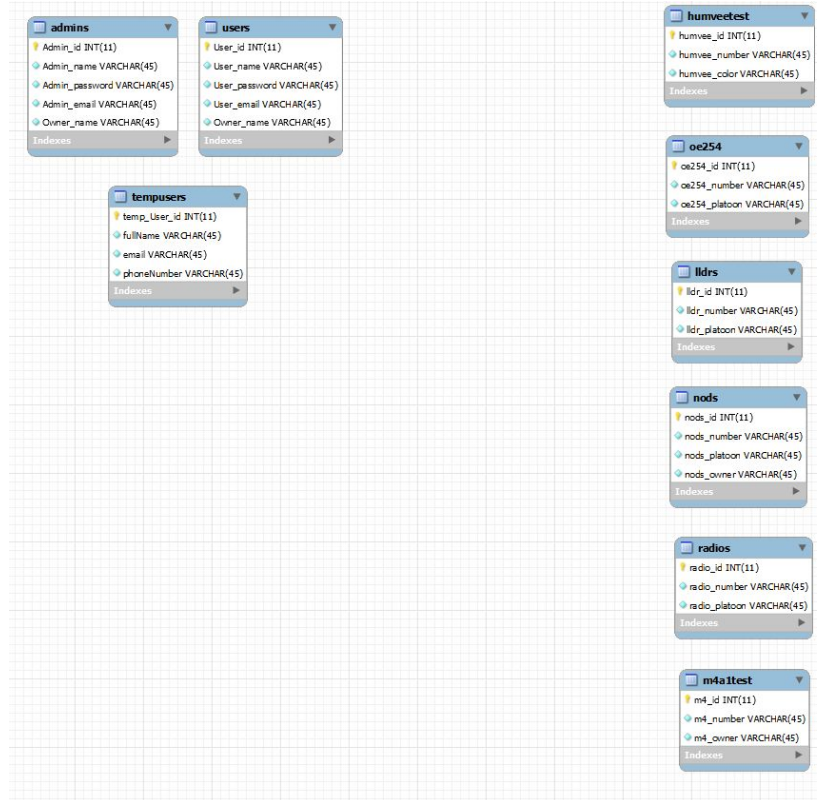
```
/*=====*/  
/*-----RIFLES-----*/
```

- `SELECT * FROM m4altest;`
- `ALTER TABLE m4altest AUTO_INCREMENT = 1;`
- `DELETE FROM m4altest;`

```
/*-----RIFLES-----*/  
/*=====*/
```



# Screenshots [New]:





# Screenshots (BACK-END) [New]:

```
//-----NODS-TABLE-----//
app.get('/api/getNODSData', (req, res) => {
  const sqlSelect =
    "SELECT * FROM nods";
  con.query(sqlSelect, (err, result) => {
    res.send(result)
  })
});

app.post('/api/NODSinsert', (req, res) => {
  const nodsNumber = req.body.nodsNumber;
  const nodsPlatoon = req.body.nodsPlatoon;
  const nodsOwner = req.body.nodsOwner;

  const sqlInsert = "INSERT INTO nods(nods_number, nods_platoon, nods_owner) VALUES (?, ?, ?)";
  con.query(sqlInsert, [nodsNumber, nodsPlatoon, nodsOwner], (err, result) => {
    console.log(result)
  })
});

app.delete('/api/NODSdelete/:nodsNumber', (req, res) => {
  const name = req.params.nodsNumber;
  const sqlDelete = "DELETE FROM nods WHERE nods_number = ?";

  con.query(sqlDelete, [name], (err, result) => {
    if(err) console.log(err);
  })
});

app.put("/api/NODSNumberupdate", (req, res) => {
  const nodsID = req.body.nodsID;
  const nodsNumber = req.body.nodsNumber;
  const sqlUpdate = "UPDATE nods SET nods_number = ? WHERE nods_id = ?";

  con.query(sqlUpdate, [nodsNumber, nodsID], (err, result) => {
    if (err) console.log(err);
  });
});

app.put("/api/NODSPlatoonupdate", (req, res) => {
  const nodsPlatoon = req.body.nodsPlatoon;
  const nodsID = req.body.nodsID;
  const sqlUpdate = "UPDATE nods SET nods_platoon = ? WHERE nods_id = ?";

  con.query(sqlUpdate, [nodsPlatoon, nodsID], (err, result) => {
    if (err) console.log(err);
  });
});
```

```
304 app.put("/api/NODSOwnerupdate", (req, res) => {
305   const nodsOwner = req.body.nodsOwner;
306   const nodsID = req.body.nodsID;
307   const sqlUpdate = "UPDATE nods SET nods_owner = ? WHERE nods_id = ?";
308
309   con.query(sqlUpdate, [nodsOwner, nodsID], (err, result) => {
310     if (err) console.log(err);
311   });
312 });
313
314
```

## Screenshots (FRONT-END) [New]:

```

703 {/*===== MAAIs' && {
704   <div className="rifleScreen">
705     <h1 className="text-center mt-5"><u> Insert into Rifles</u></h1>
706     <br></br>
707     <label><b>Rifle Number:</b>&nbsp;</label>
708     <input type="text" name="HumveeName" onChange={(e) => {setRifleNumber(e.target.value)}}/>
709     <br></br>
710     <label><b>Rifle Owner:</b>&nbsp;</label>
711     <input type="text" name="HumveeColor" onChange={(e) => {setRifleOwner(e.target.value)}}/>
712     <br></br>
713     <button className="SubmitHumvee" onClick={submitRifleReview}> Submit </button>
714     <h1 className="text-center mt-5"><u> Rifle List</u></h1>
715     <table>
716       <tbody>
717         <tr>
718           <th>Delete</th>
719           <th>Update Number&emsp;&emsp;&emsp;&emsp;&emsp;& Update Owner</th>
720           <th>Rifle ID</th>
721           <th>Rifle Number</th>
722           <th>Rifle Owner</th>
723         </tr>
724         {rifleList.map((val, key) => {
725           return [
726             <tr key={key}>
727               <td><button id="deleteButton" onClick={() => {deleteMAAIReview(val.m4_number)}}>Delete</button></td>
728               <td><text className = "updateInput">[Number:] </text><input className="updateNumberInput" onChange={(e) => {setNewRifleNumber(e.target.value)}}</input>
729               <button id="updateButton" onClick={() => {updateRifleNumber(val.m4_id)}}>&uarr;</button>
730               <td><text className = "updateInput"> [Owner:] </text><input className="updatePlatoonInput" onChange={(e) => {setNewRifleOwner(e.target.value)}}</input>
731               <button id="updateButton" onClick={() => {updateRifleOwner(val.m4_id)}}>&uarr;</button></td>
732               <td>{val.m4_id}</td>
733               <td>{val.m4_number}</td>
734               <td>{val.m4_owner}</td>
735             </tr>
736           ]
737         })}
738       </tbody>
739     </table>
740     <br></br>
741   </div>
742 }
743 {/*===== RIFLES ----- */}

```

# Screenshots (FRONT-END) [New]:

```
338 //-----RIFLES-----//
339 const [rifleNumber, setRifleNumber] = useState("");
340 const [rifleOwner, setRifleOwner] = useState("");
341 const [rifleList, setRifleList] = useState([]);
342 const [newRifleNumber, setNewRifleNumber] = useState("");
343 const [newRifleOwner, setNewRifleOwner] = useState("");
344
345 useEffect(() => {
346   Axios.get('http://localhost:3001/api/getRifleData').then((response) => {
347     setRifleList(response.data)
348   })
349 }, [])
350
351 const submitRifleReview = () => {
352   Axios.post('http://localhost:3001/api/RifleInsert',
353     {rifleNumber: rifleNumber, rifleOwner: rifleOwner}
354   );
355   Axios.get('http://localhost:3001/api/getRifleData').then((response) => {
356     setRifleList(response.data)
357   })
358 }
359
360
361 const deleteRifleReview = (rifleNumber) => {
362   Axios.delete('http://localhost:3001/api/RifleDelete/${rifleNumber}');
363
364   Axios.get('http://localhost:3001/api/getRifleData').then((response) => {
365     setRifleList(response.data)
366   })
367 }
368
369 };
370
371 //-----RIFLE UPDATE-----//
372 const updateRifleNumber = (rifleID) => {
373   Axios.put('http://localhost:3001/api/RifleNumberupdate',
374     { rifleID: rifleID,
375       rifleNumber: newRifleNumber,
376     })
377   setNewRifleNumber("");
378
379   Axios.get('http://localhost:3001/api/getRifleData').then((response) => {
380     setRifleList(response.data)
381   })
382 }
383 };
384
385
386 const updateRifleOwner = (rifleID) => {
387   Axios.put('http://localhost:3001/api/RifleOwnerupdate',
388     { rifleID: rifleID,
389       rifleOwner: newRifleOwner,
390     })
391   setNewRifleOwner("");
392
393   Axios.get('http://localhost:3001/api/getRifleData').then((response) => {
394     setRifleList(response.data)
395   })
396 }
397 };
398 //-----RIFLE UPDATE-----//
399
```

## Closing:

After finishing this project I can confidently say I have become much more comfortable with Javascript, HTML and CSS. Some files contain more than 500 lines of code which was a real headache to maintain. Totalling the hours spent on this project would sum up to a little over 26 hours. Most of which was spent on debugging database connections and SQL statements. There are still some issues in the website, including but not limited to, real-time update does not always update in real time, hidden pages can be accessed by simply changing the url and most likely SQL Injection security issues exist.