The headquarters.

The system is a dedicated PHP server that interacts with MongoDB database. It allows the authorized headquarters user to perform the account and class related operations:

**Main Functions:**

**-Creating a class:**

● The system asks for the class name

○ limited to 50 characters

● The system asks the commander’s personal number

○ limited to 7 numbers

● The system asks for the number of soldiers

○ from 1 to 65 soldiers

● The system asks for the soldiers’ personal numbers - which will generate all the soldier’s data (image, role, exceptions, etc..)

○ if one or more of the soldiers’ numbers don’t exist in the system, then the system will give an error that the soldier doesn’t exist in the system

● The system asks for the headquarters’ admin’s credentials to confirm database action.

○ if credentials incorrect, the system will not perform the action and inform the user.

● The system shows a “Create Class” option which will add the class with it’s specifications to soldiers in the system’s DB.

**-Update/Delete class:**

● The system asks for the new class name

○ limited to 50 characters

● The system asks for the class ID

○ If the class name is not the system asks for the input again

● The system asks for the headquarters’ admin’s credentials to confirm database action.

○ if credentials incorrect, the system will not perform the action and inform the user.

● The system present “Update Class” and “Delete Class” buttons.

○ If “Update Class" button is pressed and the new class name is correct, the name of the class shall be changed to the new class name.

○ If “Delete Class" button is pressed, the class will be removed from the system (from soldiers’ data)

**-Creating account**

● The system asks for the soldier’s personal number

○ limited to 7 numbers

● The system asks for the soldier’s full name

○ limited to 50 characters

● The system asks for the soldier’s PAKAL

○ limited to 50 characters

● The system asks for the soldier’s password

● The system checks with the MongoDB database the soldier’s information

○ if personal number wasn’t found, the system will create a soldier account

○ if exists, notify the user, and do not make the new account.

● The system asks for the headquarters’ admin’s credentials to confirm database action.

○ if credentials incorrect, the system will not perform the action and inform the user.

**-Updating accounts**

● The system asks the soldier’s personal number that the user want to update or delete

○ If exist:

■ The system will show the soldier’s information with two buttons: update and delete

○ Else:

■ show error that the personal number doesn’t exist within the squad

● If the user chooses to delete the system asks to enter the password

○ If the password is valid then the system proceeds to the delete function.

○ If not then the system alerts that the password is incorrect and there are 3 more tries, after those tries, the action is invalid for an hour.

○ In the delete function there is an “are you sure” safety measure to prevent unwanted deletion.

-If the user presses “yes” the system deletes account and notifies the user of successful deletion

-Else the system notifies the user that the account was not deleted.

● If the user chooses to update the system will ask for the password to permit the change

○ If the password is valid then the system will ask for the new fullName of the soldier.

-If the new fullName is valid, it updates it, shows a success notification and updates the authorities.

-Else the system notifies that the name was not updated.

○ If not then the system alerts that the password is incorrect and there are 3 more tries, after those tries, the action is invalid for an hour.

**-Help recovering account:**

● The system will ask for the soldier’s personal number

○ If the number isn't verified the system asks for the input again

● The system will generate and update the new password

**-Show Main Page:**

● The system will have a link to select a class

● If a class is selected, the system shows in a table all soldiers information in the selected class

○ Show the personal number, name, and class, if the soldier not in a class it will show none.

● The system will have a link to create a class

● The system will have a link to create a new account

● The system will have a link to create a new admin account

● The system will have a link to update a class

**Service Functions:**

**-soldierExists:**

● Check that soldier actually exists in the system

**-getExistingSoldiers**

● get a list of soldiers that exist in the system

**-encrypt:**

● encrypt a password (may be used for other strings as well)

**-verifyPassword:**

● verify the password. Encrypts before sending.

**-verifyLogin:**

● verify the login credentials. Encrypts password before sending.

**-generateUniqueClassId**

● generate unique ID for class

**-getUniqueClasses**

● find unique classes that exist in the system

**-updateClass**

● update class for the soldiers in it

**-deleteSoldierFromCollection**

●delete the soldier from connected MongoDB collection.

**Notes:**

-Password is encrypted before being sent.

-The system utilizes multiple PHP pages for the functions.

-The headquarters object utilizes Session system to make sure it connects properly to PHP files in links (that session remains the same and no security issues appear).

-The Update and Delete class functions are realized in the Update Class page.