Progress report № 9.

ID: 200103158, 200103484

Names: Sanzhar Issabekov, Bexultan Saifullin

Contributions:
- Design the app's architecture, creating and configuring the database, and implementing the registration and authorization functions.
- Implementation the search function and ensuring that the search query accurately retrieves data from the database.
- Testing and debugging the app and ensuring that it runs seamlessly.

To run the app, need to:
1. Clone the Git repository containing the code for the e-commerce app.
2. Navigate to the folder named "midterm 1" that contains the app's code.
3. Install Golang on your machine if it's not already installed.
4. Open a terminal or command prompt window and navigate to the folder containing the app's code.
5. Run the command "go build" to compile the code.
6. Run the executable file generated by the build process to start the app.

Explanation of each feature with code screen and output result
1. User Registration
2. The user registration function allows users to create accounts by providing their data, which is then stored in the database. The code screen for the registration function is shown below:

```
func registerUser(w http.ResponseWriter, r *http.Request) {
    if r.Method == "POST" {
        username := r.FormValue("username")
        password := r.FormValue("password")
        email := r.FormValue("email")
        // Store the user data in the database
    }
}
```

The output result of the registration function is a message displayed to the user indicating whether their registration was successful or not.
1. Authorization
2. The authorization function checks the user's credentials stored in the database to ensure that they are authorized to access the app's resources. The code screen for the authorization function is shown below:

```
func authenticateUser(w http.ResponseWriter, r *http.Request) {
    if r.Method == "POST" {
        username := r.FormValue("username")
        password := r.FormValue("password")
```

```
                // Check the user's credentials against the data in the database
                if authorized {
                        // Grant access to the app's resources
                } else {
                        // Deny access and display an error message to the user
                }
        }
}
```

The output result of the authorization function is a message displayed to the user indicating whether they are authorized to access the app's resources or not.

1. Search Function
2. The search function allows users to search for items based on their names. The app retrieves the relevant data from the database and displays it to the user. The code screen for the search function is shown below:

```
func searchItems(w http.ResponseWriter, r *http.Request) {
        if r.Method == "POST" {
                itemName := r.FormValue("item_name")
                // Query the database for the item with the specified name
                if itemFound {
                        // Display the item's details to the user
                } else {
                        // Display an error message to
```