Edward Garcia

CS499-17487-Computer Science Capstone

5-2 Milestone Four: Enhancement Three: Databases Narrative
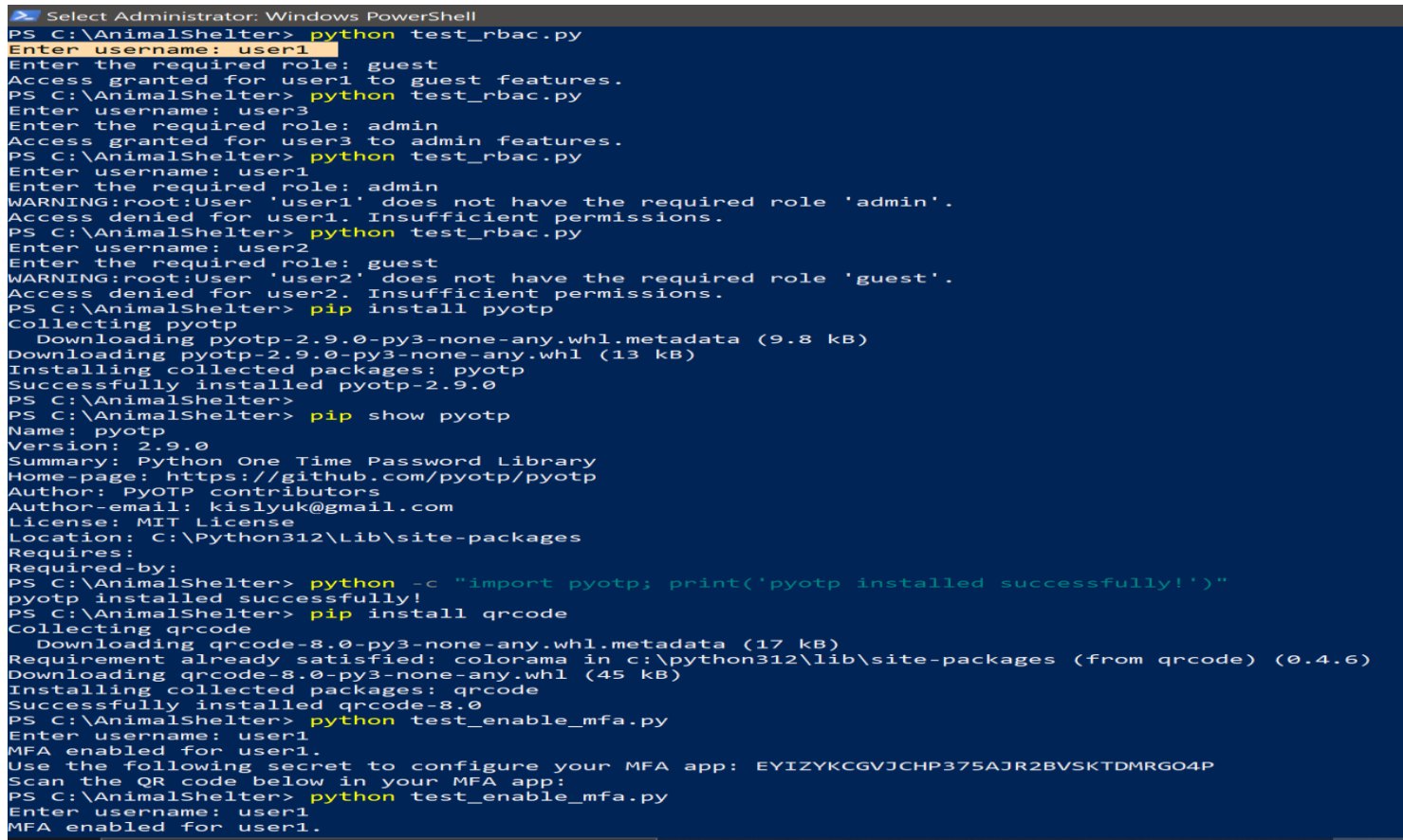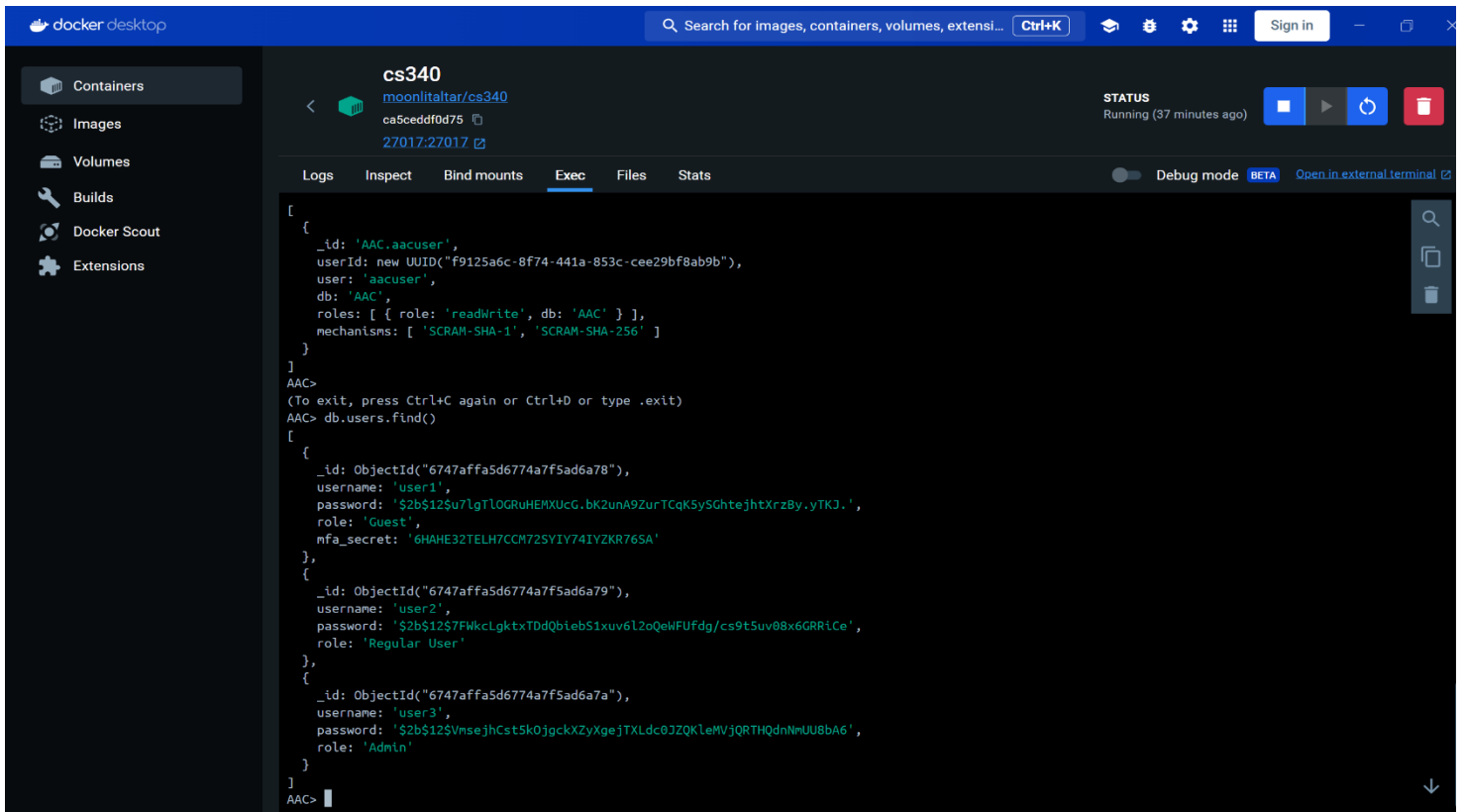
Professor Krupa

November 30, 2024
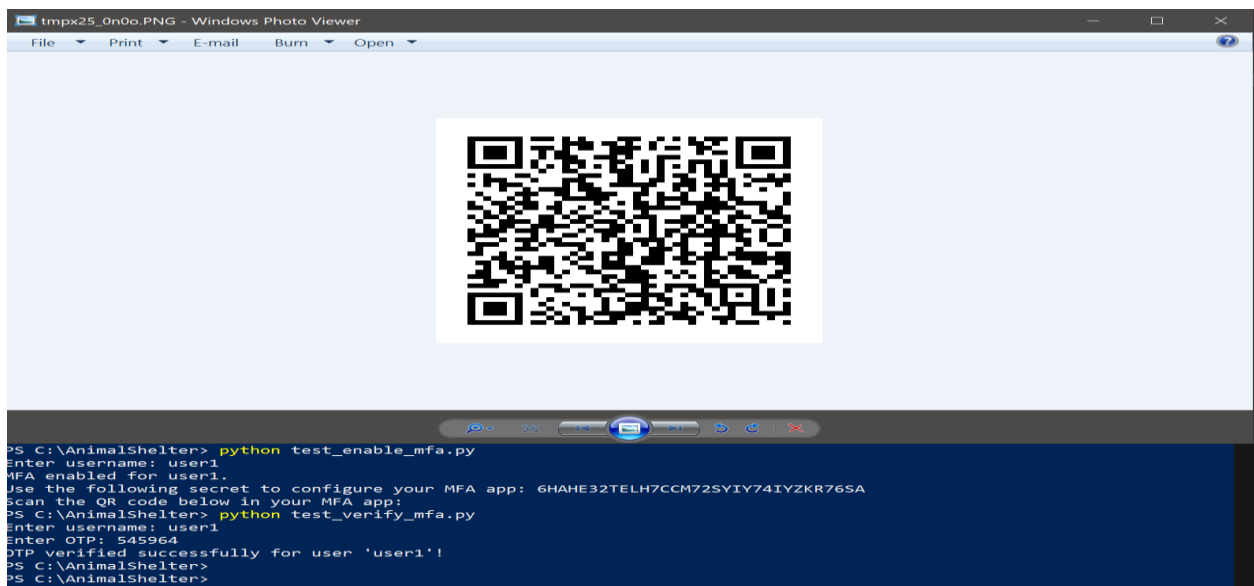
5-2 Milestone Four: Enhancement Three: Databases Narrative

The Animal Shelter Application I created during my CS 340 Client/Server Development course, has been enhanced further to strengthen its database management capabilities, my primary focus is on advanced security with UI integration, user/MFA authentication, and role-based access control (RBAC). By showcasing my skills in MongoDB, Python, and user interface design, these enhancements demonstrate my ability to deliver professional-quality solutions that are both functional and secure.

To enhance the database, I created a "users" collection in MongoDB, designed to securely manage user credentials and roles. This collection includes fields for usernames, bcrypt-hashed passwords, roles (guest, regular user, or admin), and optional multi-factor authentication (MFA) secrets. The MFA implementation involves generating QR codes that allow users to configure their authentication apps, adding an extra layer of security. In my uploaded video for Enhancement 3 I used the Microsoft Authenticator app to verify the QR code. The RBAC system is used to verify that users have appropriate access levels based on their roles. I created three sample users to demonstrate functionality: User1, with guest privileges and MFA enabled (username: user1, password: Secret); User2, with regular user privileges and MFA disabled (username: user2, password: Houston77); and User3, with admin privileges and MFA enabled (username: user3, password: Astros44). Users added to the database with encrypted passwords, security testing methods, and QR code are listed in the screenshots below.

Containers
Images
Volumes
Builds
Docker Scout
Extensions

cs340
moonlitaltar/cs340
ca5ceddf0d75
27017:27017

STATUS
Running (37 minutes ago)

Logs   Inspect   Bind mounts   Exec   Files   Stats

Debug mode BETA   Open in external terminal

```
[
  {
    _id: 'AAC.aacuser',
    userId: new UUID("f9125a6c-8f74-441a-853c-cee29bf8ab9b"),
    user: 'aacuser',
    db: 'AAC',
    roles: [ { role: 'readWrite', db: 'AAC' } ],
    mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
  }
]
AAC>
(To exit, press Ctrl+C again or Ctrl+D or type .exit)
AAC> db.users.find()
[
  {
    _id: ObjectId("6747affa5d6774a7f5ad6a78"),
    username: 'user1',
    password: '$2b$12$u7lgTlOGRuHEMXUcG.bK2unA9ZurTCqK5ySGhtejhtXrzBy.yTKJ.',
    role: 'Guest',
    mfa_secret: '6HAHE32TELH7CCM72SYIY74IYZKR76SA'
  },
  {
    _id: ObjectId("6747affa5d6774a7f5ad6a79"),
    username: 'user2',
    password: '$2b$12$7FWkcLgktxTDdQbiebS1xuv6l2oQeWFUfdg/cs9t5uv08x6GRRiCe',
    role: 'Regular User'
  },
  {
    _id: ObjectId("6747affa5d6774a7f5ad6a7a"),
    username: 'user3',
    password: '$2b$12$VmsejhCst5kOjgckXZyXgejTXLdc0JZQKleMVjQRTHQdnNmUU8bA6',
    role: 'Admin'
  }
]
AAC>
```

```
Select Administrator: Windows PowerShell
PS C:\AnimalShelter> python test_rbac.py
Enter username: user1
Enter the required role: guest
Access granted for user1 to guest features.
PS C:\AnimalShelter> python test_rbac.py
Enter username: user3
Enter the required role: admin
Access granted for user3 to admin features.
PS C:\AnimalShelter> python test_rbac.py
Enter username: user1
Enter the required role: admin
WARNING:root:User 'user1' does not have the required role 'admin'.
Access denied for user1. Insufficient permissions.
PS C:\AnimalShelter> python test_rbac.py
Enter username: user2
Enter the required role: guest
WARNING:root:User 'user2' does not have the required role 'guest'.
Access denied for user2. Insufficient permissions.
PS C:\AnimalShelter> pip install pyotp
Collecting pyotp
  Downloading pyotp-2.9.0-py3-none-any.whl.metadata (9.8 kB)
Downloading pyotp-2.9.0-py3-none-any.whl (13 kB)
Installing collected packages: pyotp
Successfully installed pyotp-2.9.0
PS C:\AnimalShelter>
PS C:\AnimalShelter> pip show pyotp
Name: pyotp
Version: 2.9.0
Summary: Python One Time Password Library
Home-page: https://github.com/pyotp/pyotp
Author: PyOTP contributors
Author-email: kislyuk@gmail.com
License: MIT License
Location: C:\Python312\Lib\site-packages
Requires:
Required-by:
PS C:\AnimalShelter> python -c "import pyotp; print('pyotp installed successfully!')"
pyotp installed successfully!
PS C:\AnimalShelter> pip install qrcode
Collecting qrcode
  Downloading qrcode-8.0-py3-none-any.whl.metadata (17 kB)
Requirement already satisfied: colorama in c:\python312\lib\site-packages (from qrcode) (0.4.6)
Downloading qrcode-8.0-py3-none-any.whl (45 kB)
Installing collected packages: qrcode
Successfully installed qrcode-8.0
PS C:\AnimalShelter> python test_enable_mfa.py
Enter username: user1
MFA enabled for user1.
Use the following secret to configure your MFA app: EYIZYKCGVJCHP375AJR2BVSKTDMRGO4P
Scan the QR code below in your MFA app:
PS C:\AnimalShelter> python test_enable_mfa.py
Enter username: user1
MFA enabled for user1.
```

(QR Code for MFA)

My implementation process followed a structured series of steps. First, I created sample

users in MongoDB, hashing their passwords with bcrypt to guarantee secure storage. Next, I

developed Python backend methods to manage user creation, authentication, MFA verification,

and role-based permissions. To enhance usability, I created the application's login page to

include fields for username, password, and MFA OTP.

(New Login Page)
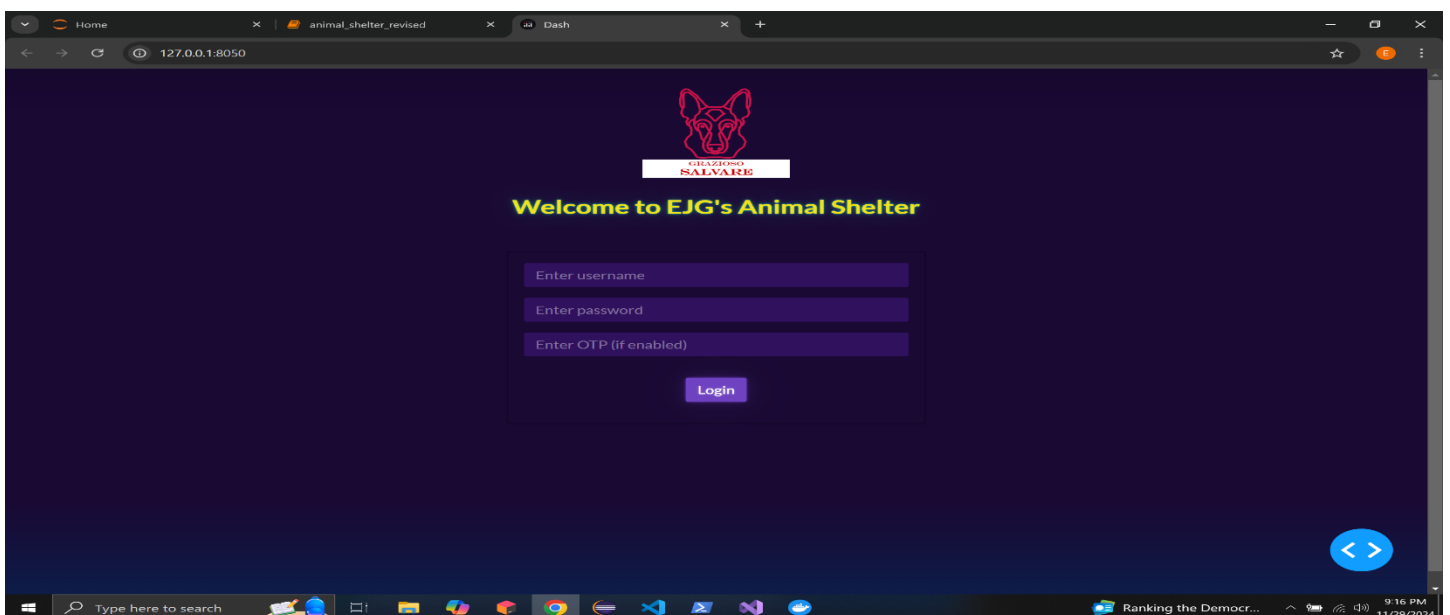
A logout button was also added to the layout for secure session termination. After integrating these features, I conducted extensive testing to validate the functionality of authentication, MFA, and RBAC, ensuring that the system operated as intended. To monitor user activity and enhance security, I implemented structured logging to track events such as login attempts, role validations, and MFA usage.

(Layout Page with Logout Button)
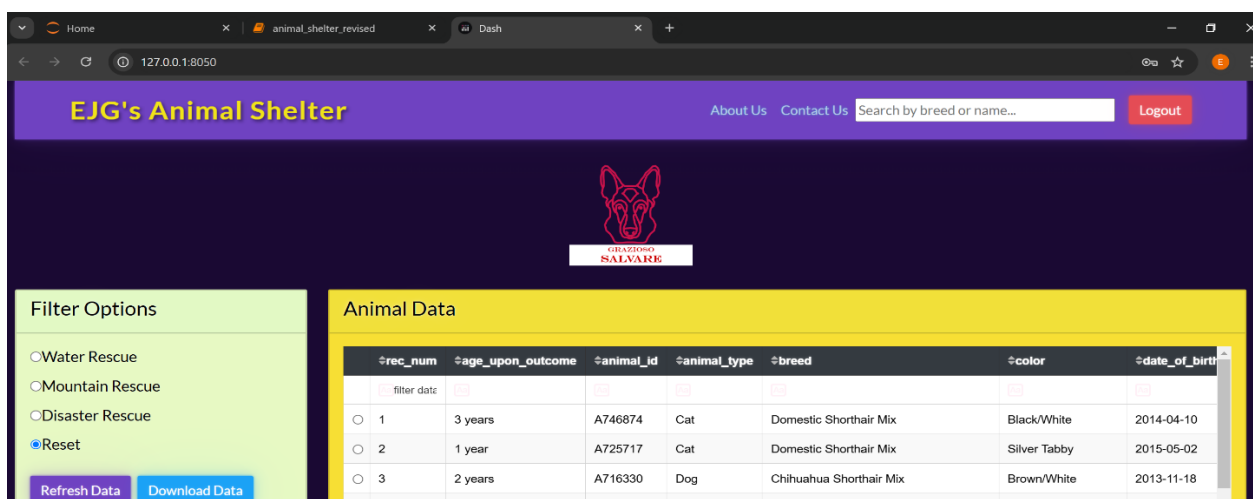
My enhancements align with several of the course outcomes. By creating the login page and logout button, I supported Course Outcome 1 by fostering a collaborative environment that allows diverse audiences to interact with the application effortlessly. The project also demonstrates Course Outcome 2 through the delivery of professional-quality code and a visually appealing, user-friendly interface, showcasing my ability to communicate technical solutions effectively. By incorporating innovative techniques such as bcrypt password hashing, QR code-based MFA, and a modular RBAC system, the project meets Course Outcome 4, emphasizing the use of advanced tools to implement secure and effective solutions. My enhancements also align with Course Outcome 5 by showcasing a security mindset, anticipating potential vulnerabilities,

and addressing them through proactive measures such as MFA, RBAC, and secure password storage.

Reflecting on this process, I learned the importance of balancing security and usability while maintaining it efficiently. Implementing MFA required comfirming that users could configure their authentication apps easily while maintaining security protocols. Defining role-specific permissions in the RBAC system required careful planning to meet organizational needs without introducing difficulty for users. These challenges reinforced my understanding of secure application development and user-centric design. My enhancements to the Animal Shelter Application have improved its functionality and security while demonstrating my ability to address real-world problems with innovative, professional solutions, making this artifact a valuable addition to my ePortfolio. . I have also uploaded a video further explaining and illustrating my enhancement 3 implementations. Here is the link to my Enhancement 3: Databases YouTube Video: https://www.youtube.com/watch?v=Xj92OldlabQ