

University of Essex

```
class Secure_Software_Development:
```

```
    def Final_Project_Demonstration():
```

```
        team_1 = {
            'bernhard van renszen',
            'hungwei lin (brandon)',
            'yin ping lai',
            'yusuf fahry',
            'yvone chan'
        }
```

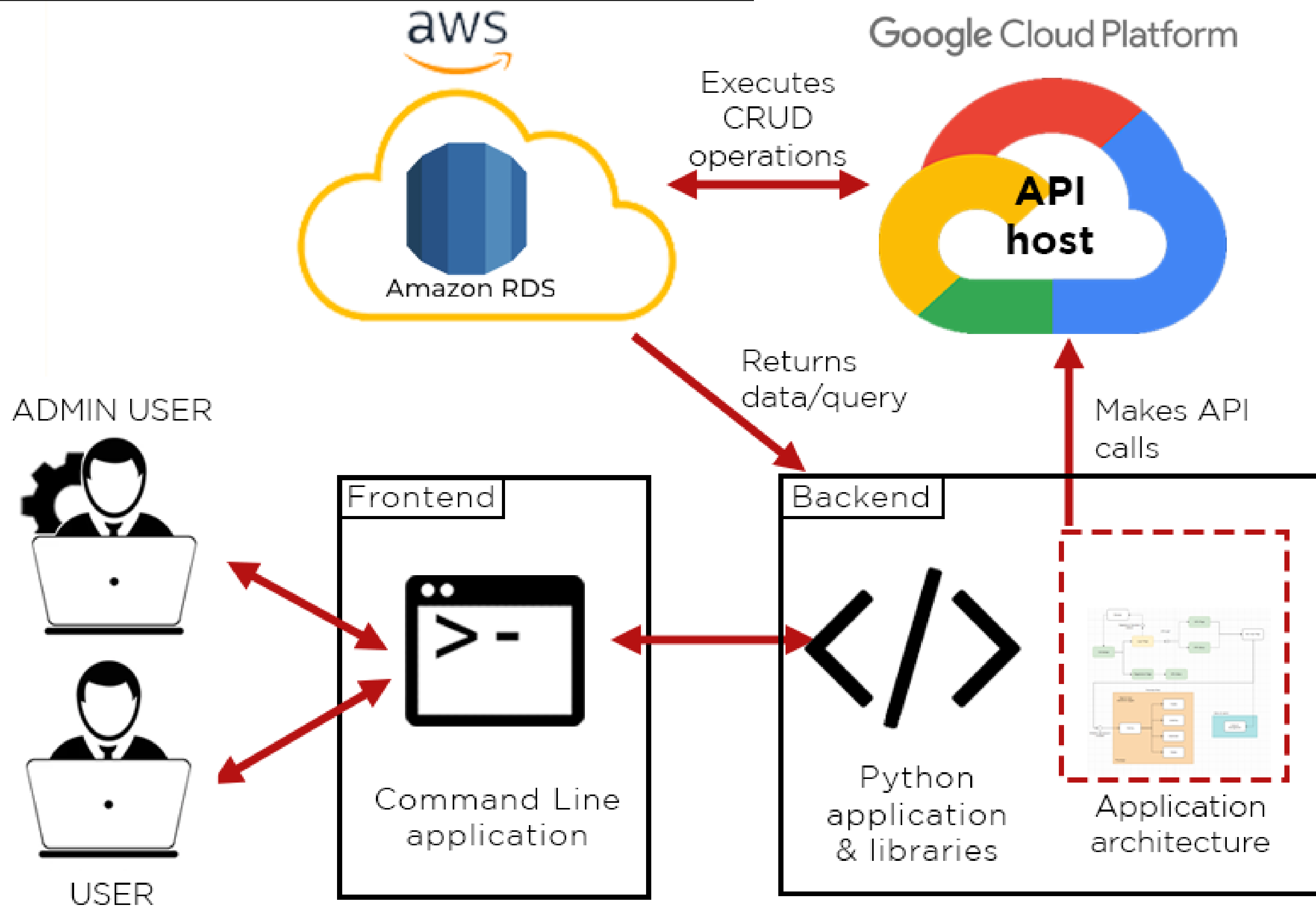
# CERN - The Problem

Due to the global Covid Pandemic, cyber attacks have increased dramatically. CERN has contacted us to develop a microservice application to address concerns of scalability and security. The application should:

- Perform secure CRUD operations
- Emphasis on security best practises
- Command Line application
- GDPR Compliant

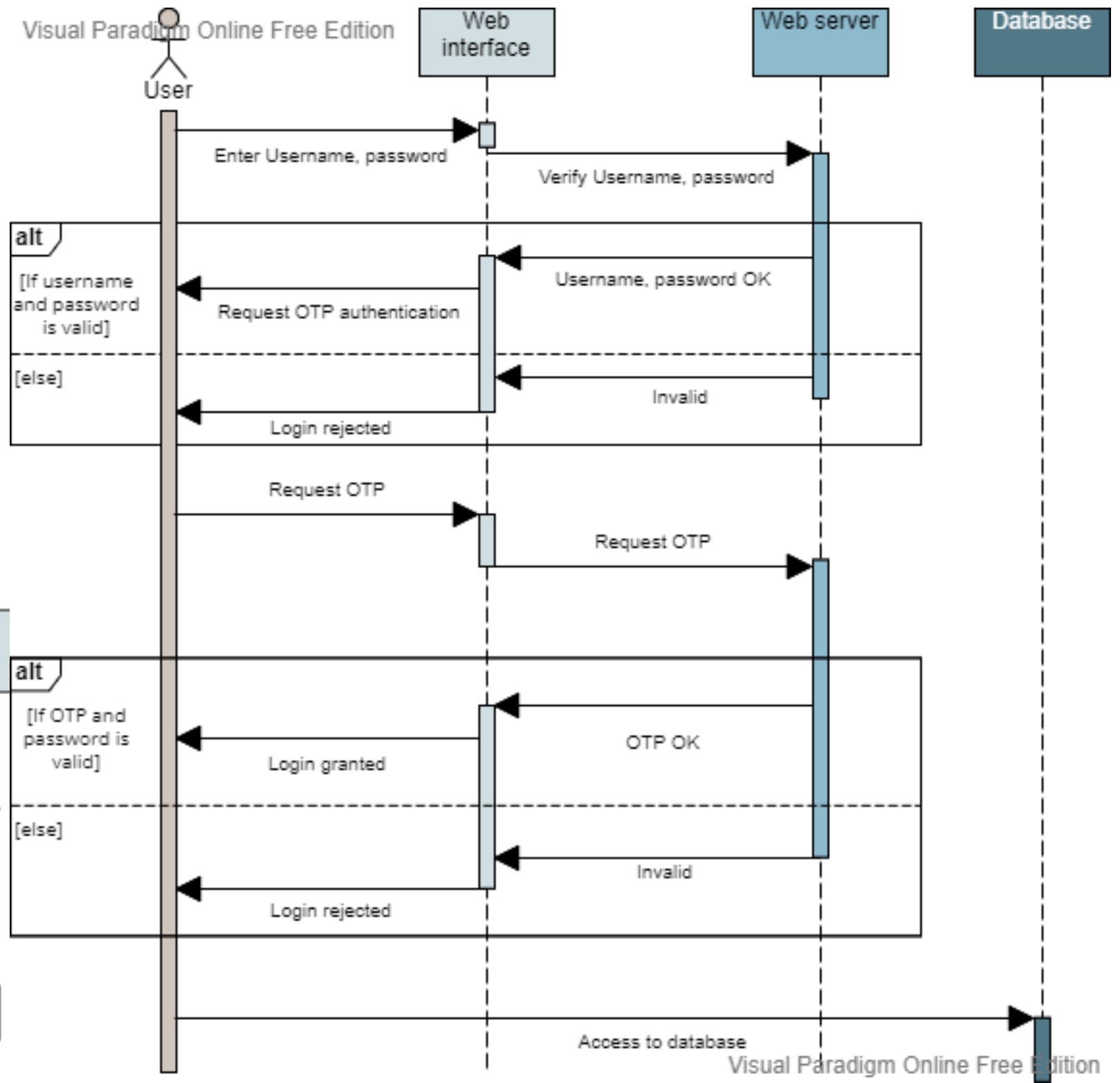
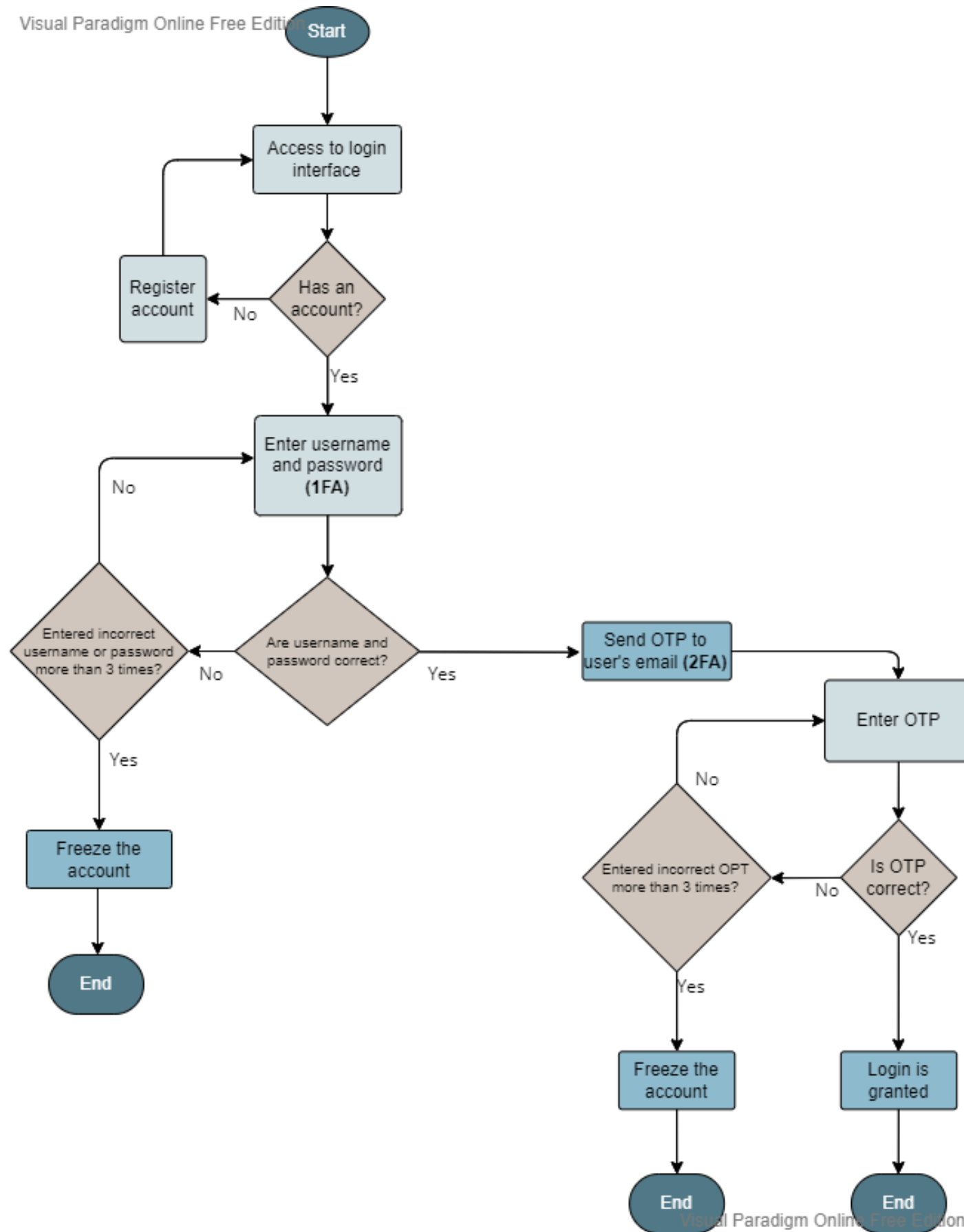


# ARCHITECTURE



# FLOW - 2FA

Visual Paradigm Online Free Edition

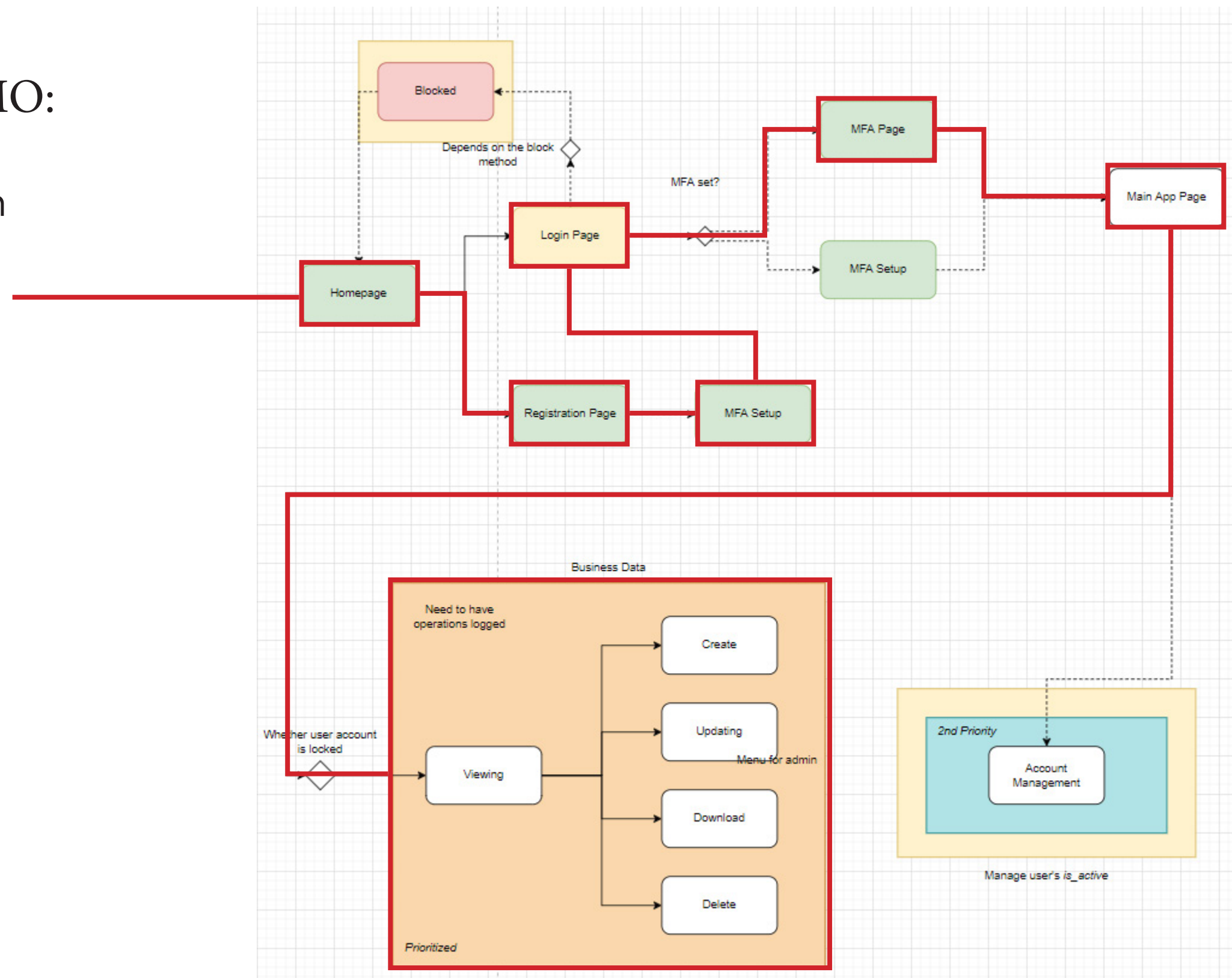


Visual Paradigm Online Free Edition

# DEMO - User

## LIVE CODE DEMO:

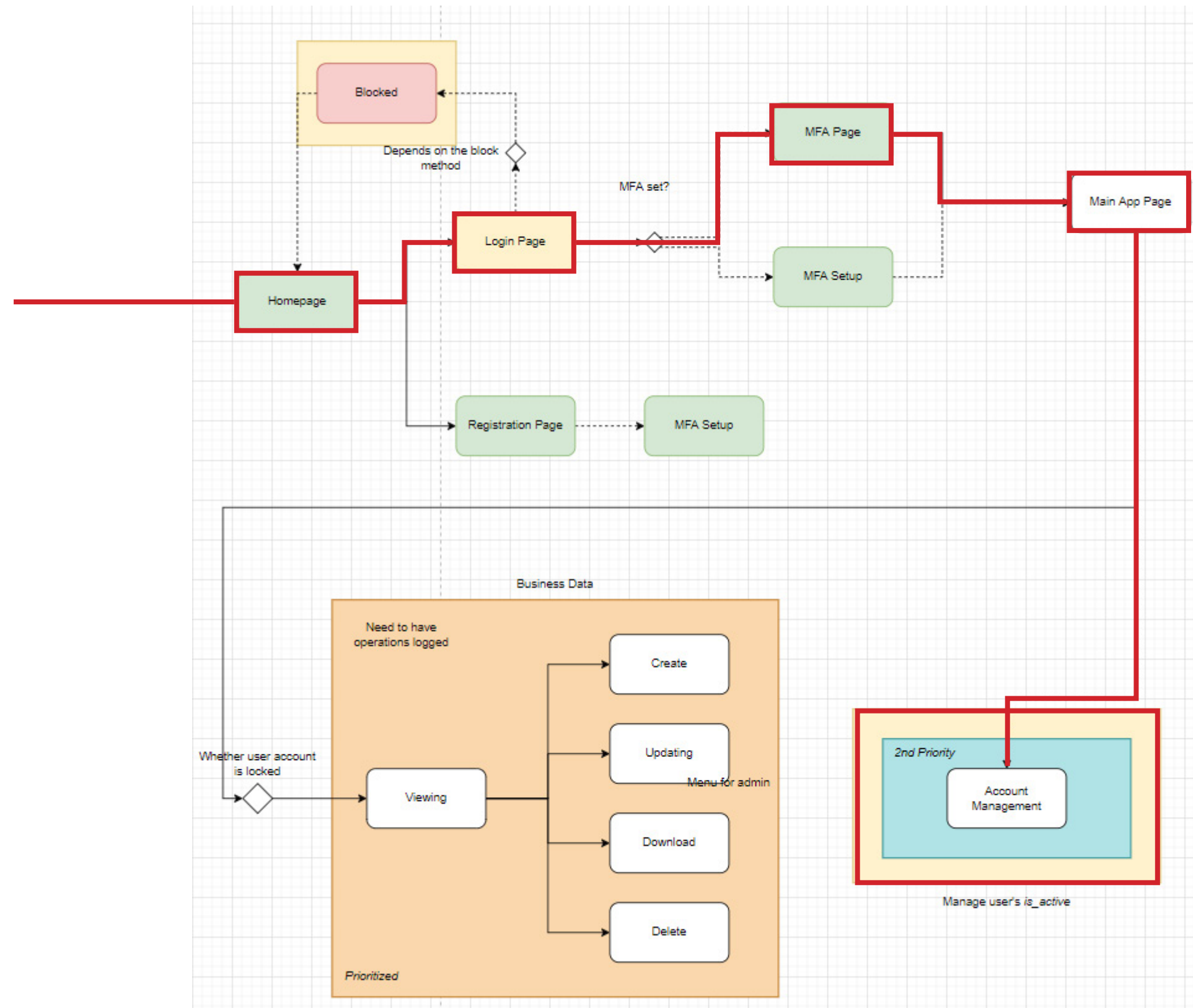
- User Registration
- User Login
- Create Data
- Update Data
- Download Data
- Delete Data
- Exit application



# DEMO - Admin

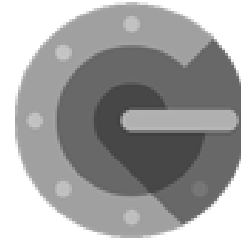
## LIVE CODE DEMO:

- Admin Registration
- Manage Users
- View logs of users



# SECURITY

## Authentication



2FA - Authenticator application for secure authentication

## Authorization

Login attempts are blocked after 3 attempts, with a timeout thereafter to avoid brute-force attacks

## Data Protection



Data encryption using MD5 hashing algorithm.  
Data in transit is also protected by AWS TLS

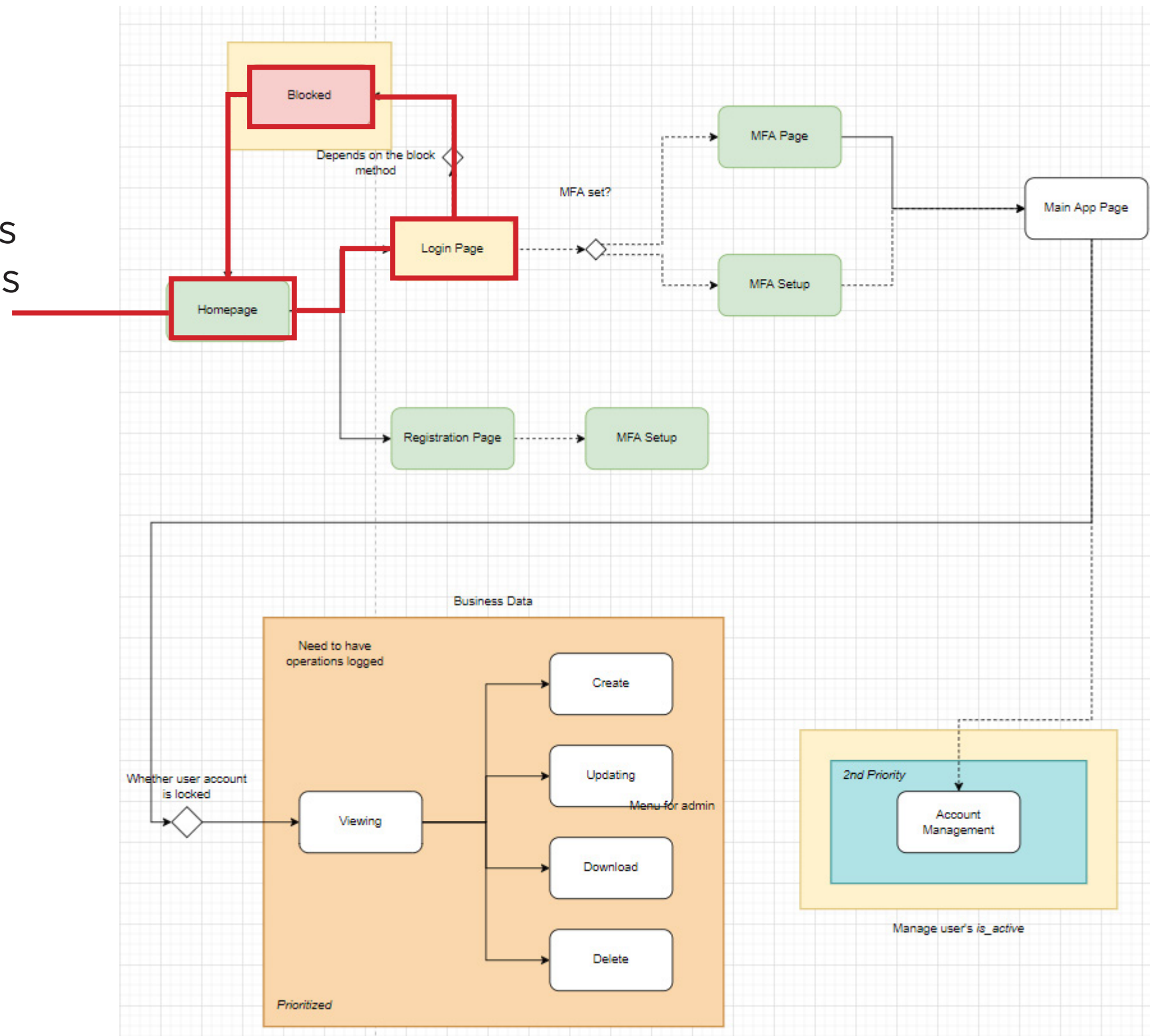
## Event Monitoring

Maintaining records of all data edited/deleted by user.  
DB table to store records of all users' login attempts and events

# DEMO - Authentication

## LIVE CODE DEMO:

- User login
- Blocked after 3 attempts
- Timeout after 3 attempts





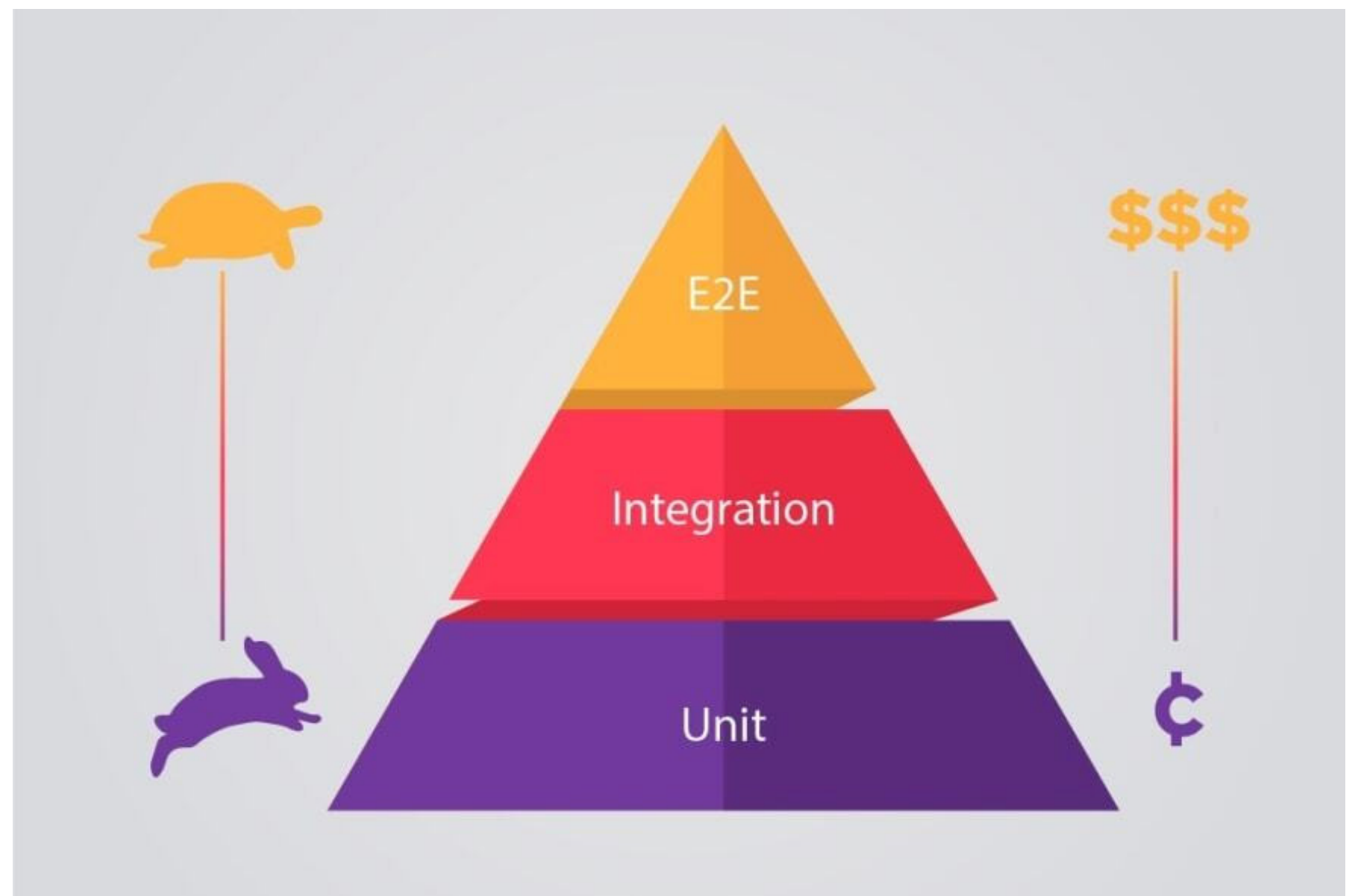
# TESTING

## LIVE CODE DEMO:

We use the PyTest framework to run our test cases. Our test cases are divided into three categories:

- Unit test
- System integration test
- End-to-end test

We focus on conducting unit testing because it is a cost effective and fast way to verify the program.





```
def Final_Project_Demonstration():  
    if pass == True:  
        return {  
            'success': 'True',  
            'message': 'Thank you'  
        }
```