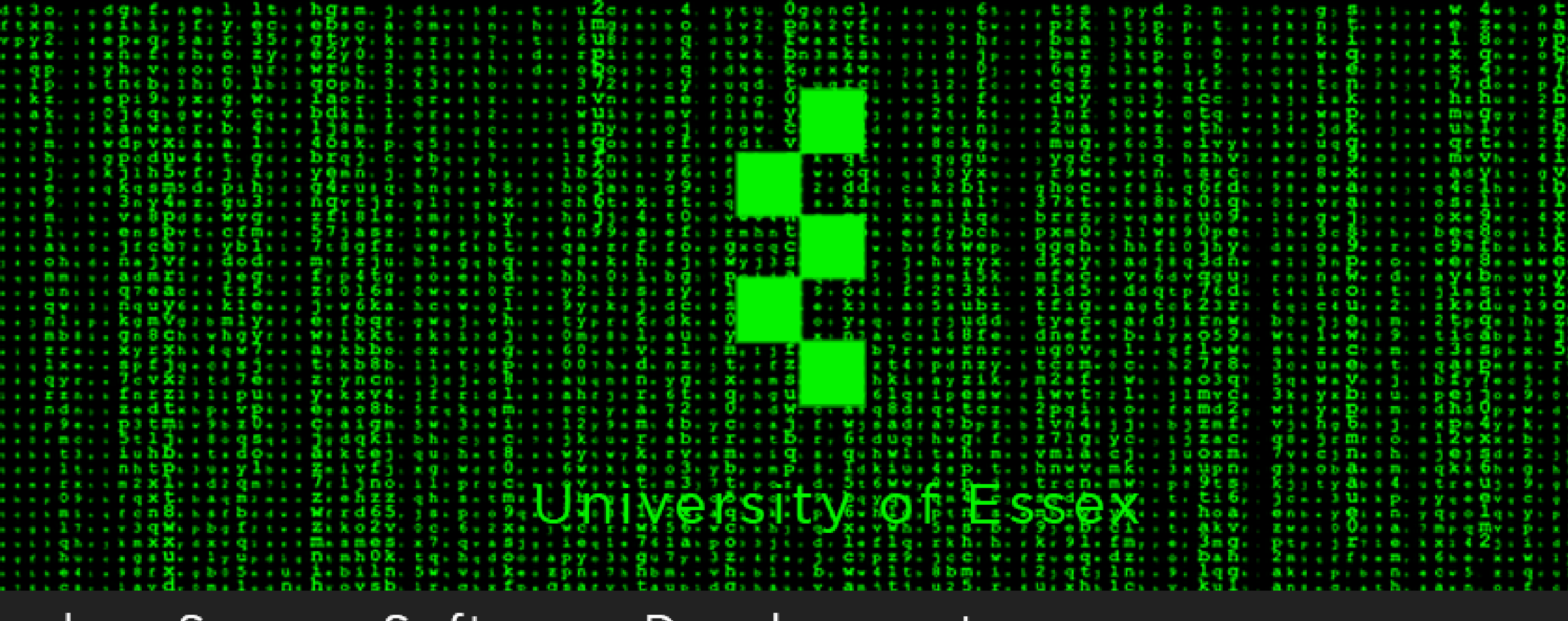


class Secure_Software_Development:

def Final_Project_Demonstration():

team_1 = {

‘bernhard van renssen’,
‘hungwei lin (brandon)’,
‘yin ping lai’,
‘yusuf fahry,
’yvone chan’

}

CERN - The Problem

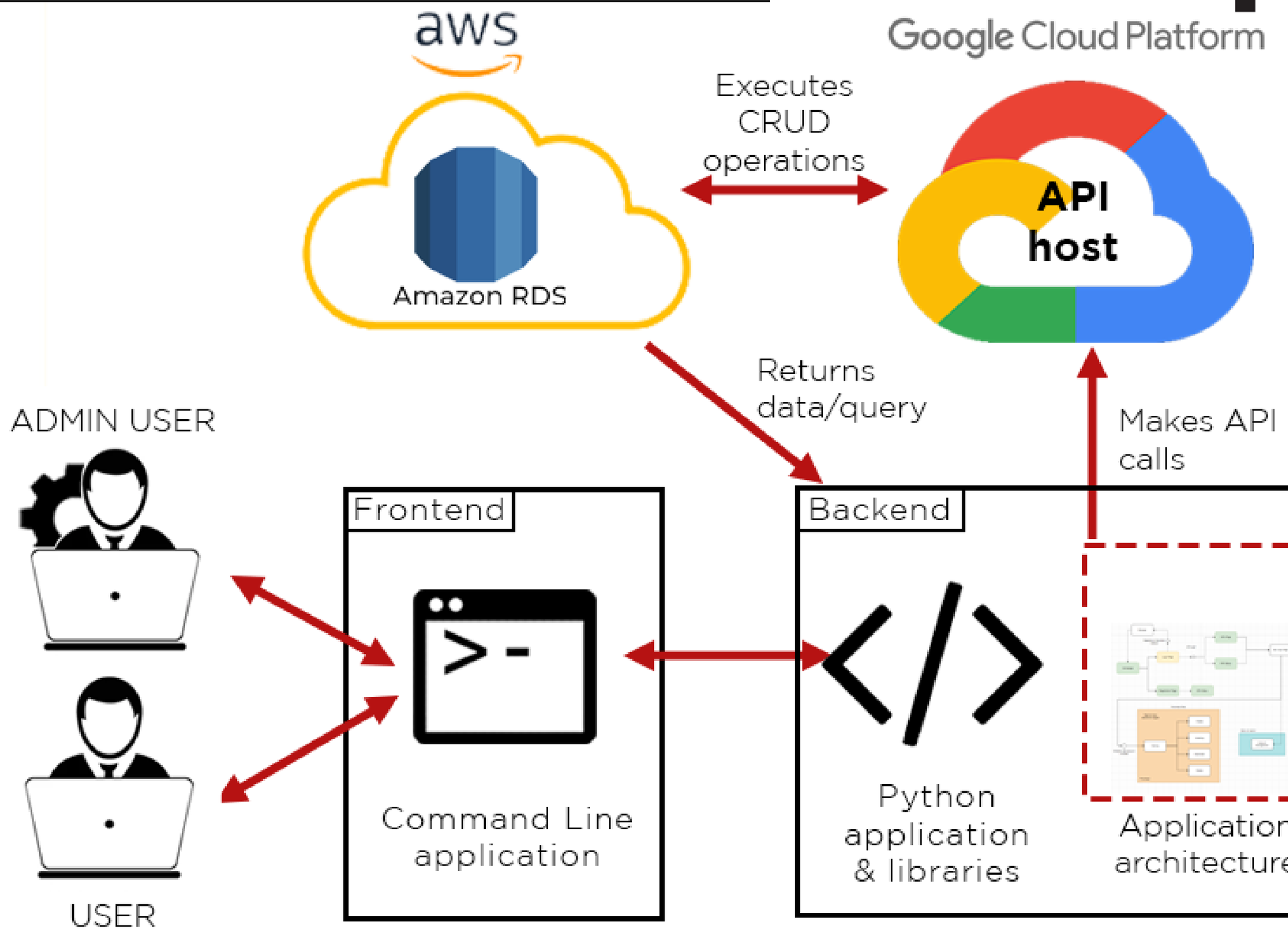
University of Essex

The Large Hadron Collider is the world's largest and highest-energy particle collider. It was built by the European Organization for Nuclear Research between 1998 and 2008 in collaboration with over 10,000 scientists and hundreds of universities and laboratories, as well as more than 100 countries.

- Over **10 000 scientists** working together on this project from over **100 countries**
- Sensors along this 27km long tunnel have **several observation points** on which oberservations are made and need to be logged
- Observations need to be logged **real-time** for scientists on different points to observe and respond to events in quick succession

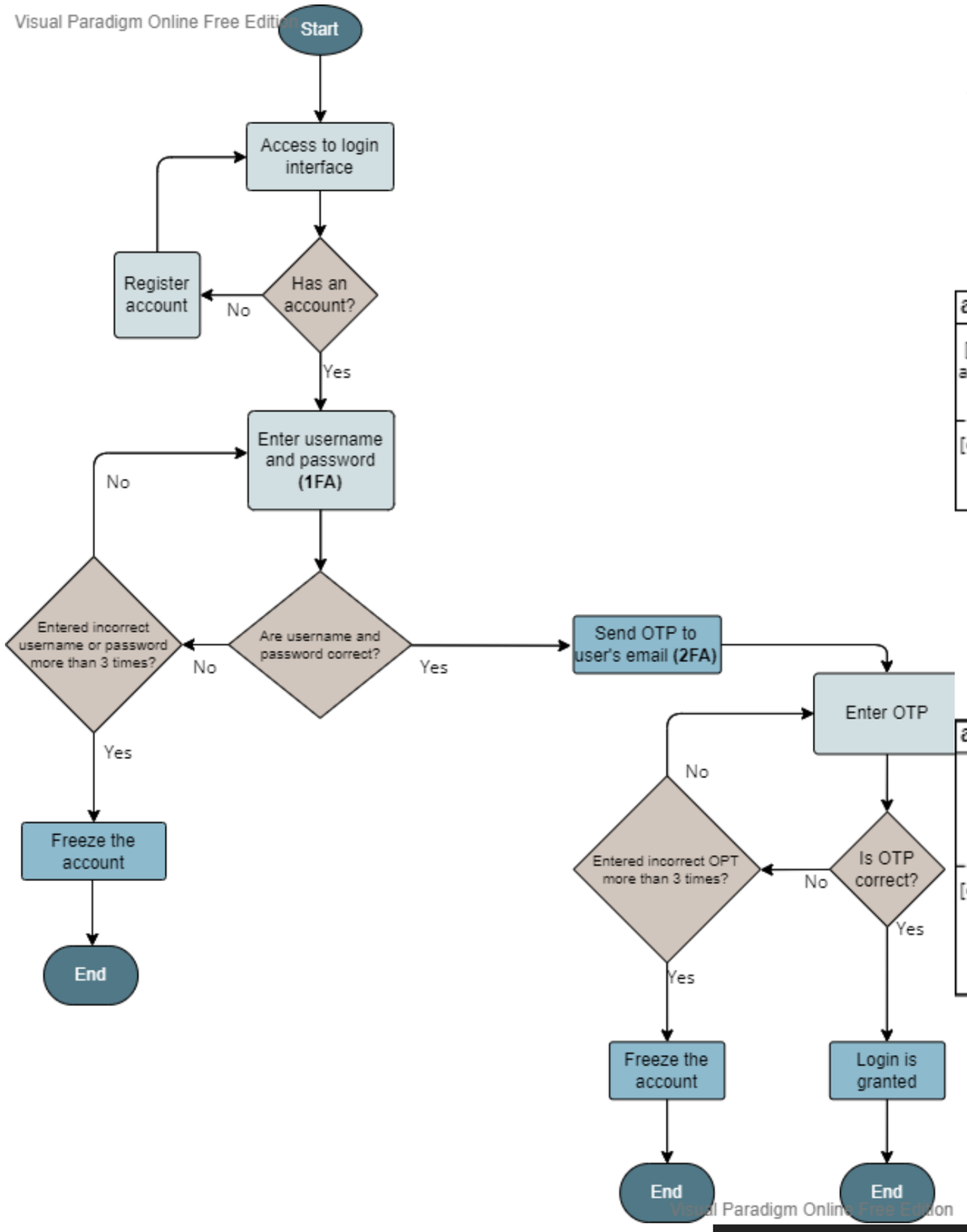
ARCHITECTURE

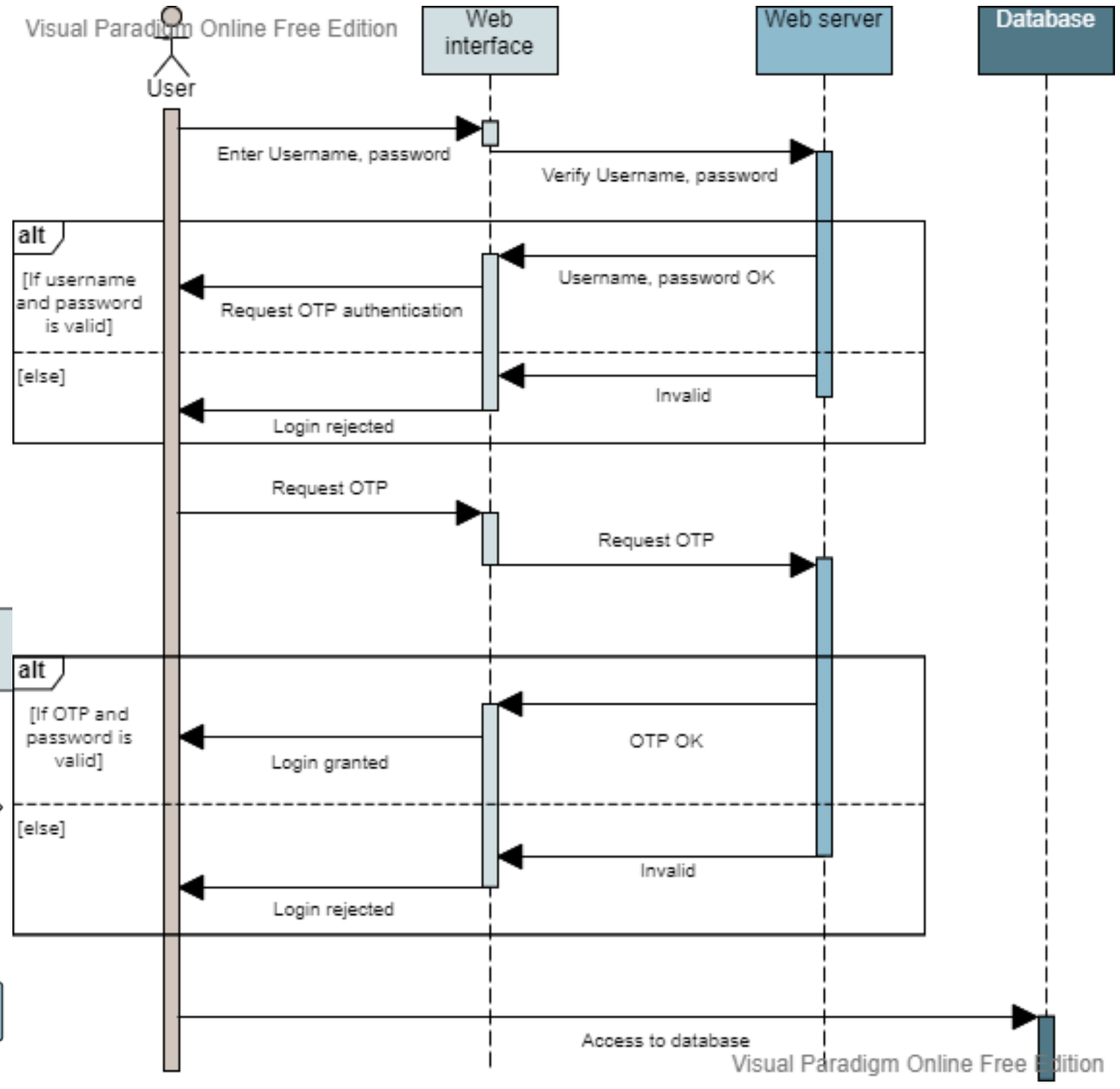
University of Essex



FLOW - 2FA

University of Essex



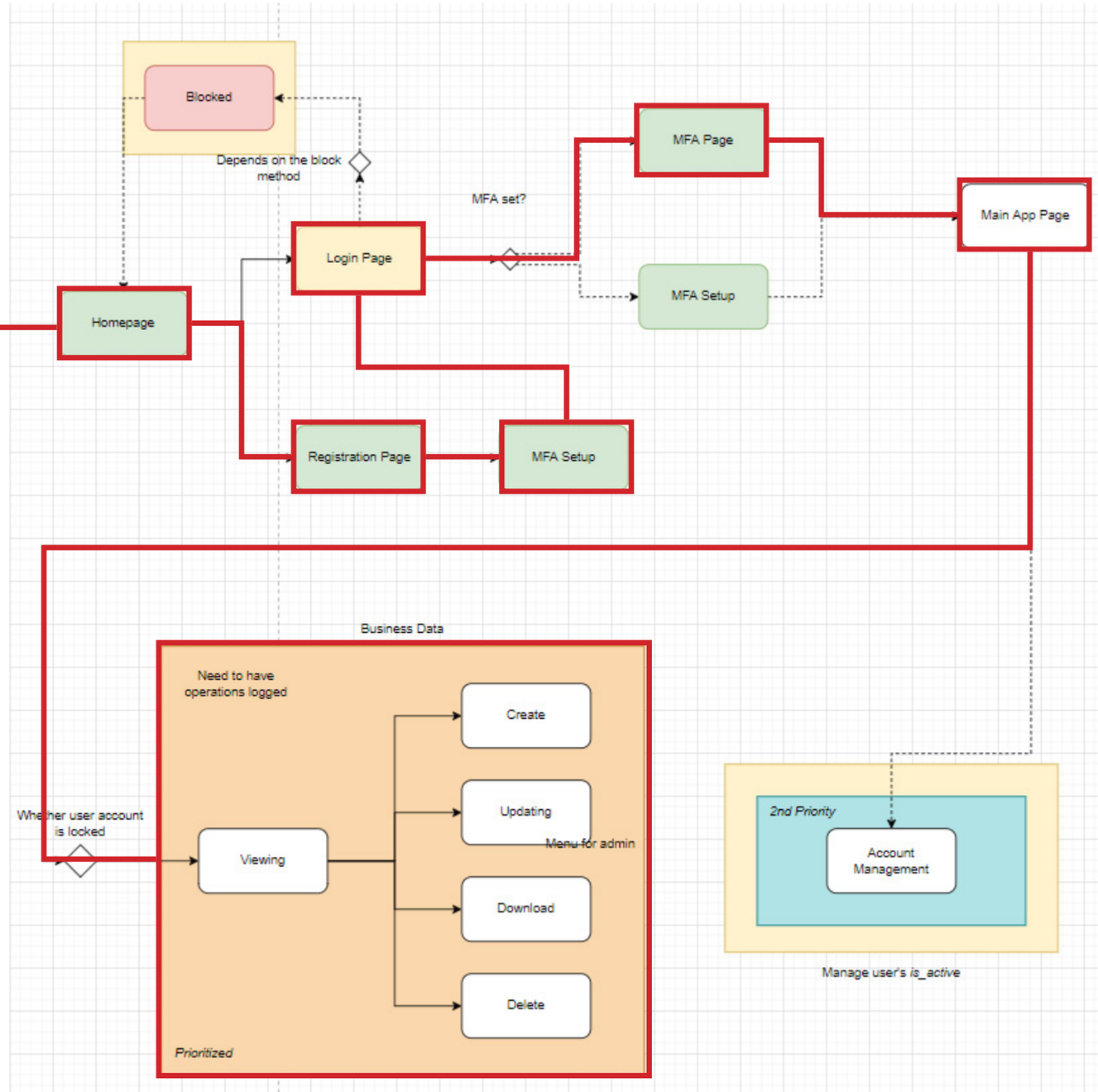


DEMO - User

University of Essex

LIVE CODE DEMO:

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

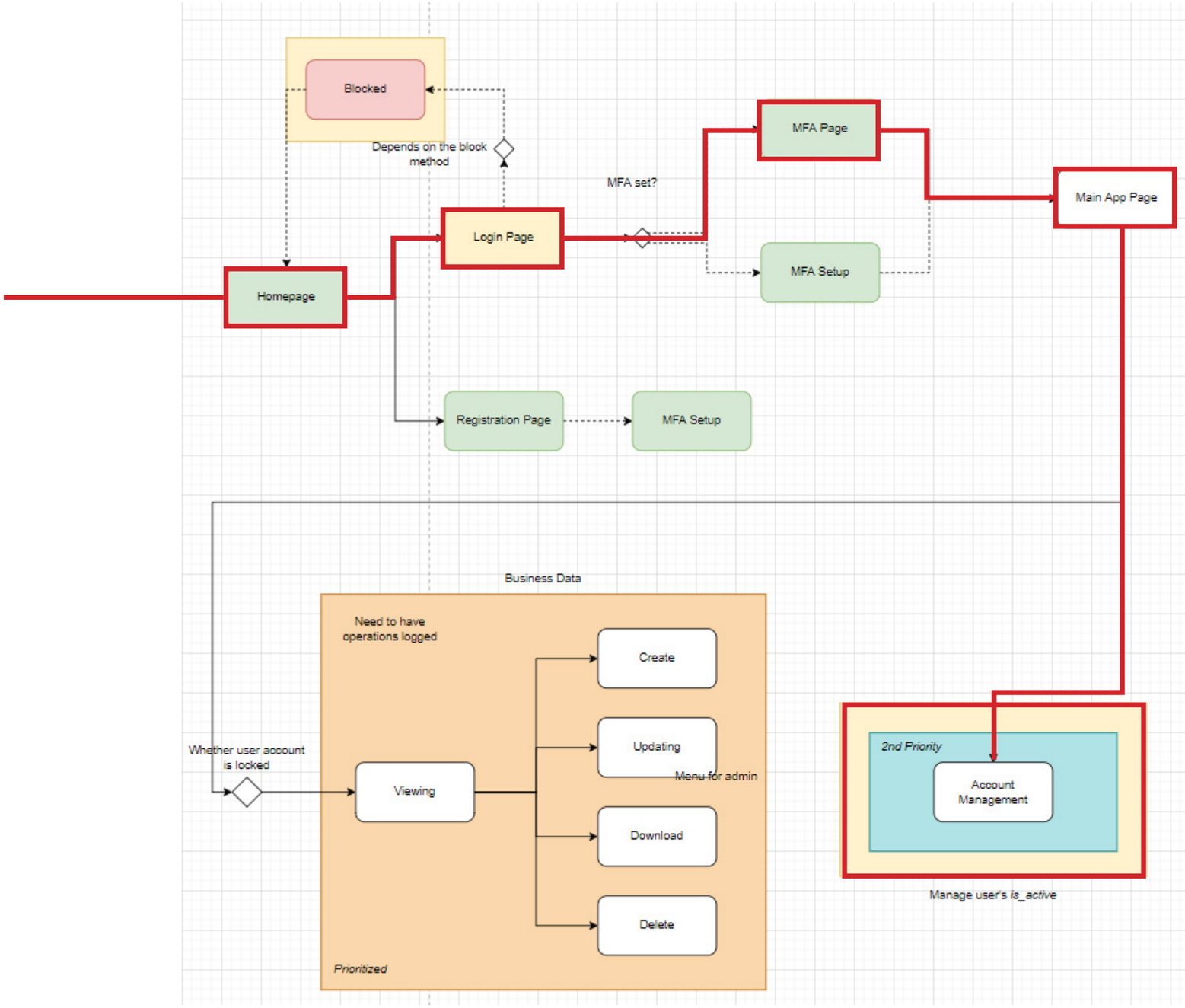


DEMO - Admin

University of Essex

LIVE CODE DEMO:

- Admin Registration
- Manage Users
- View logs of users



SECURITY - OWASP

University of Essex

OWASP Top 10
security considerations

The OWASP Top Ten Proactive Controls 2018 is a list of **security techniques** that should be **included** in **every software development project**. They are ordered by order of importance, with control number 1 being the most important (OWASP, 2021)

CONTROL	CONSIDERATION
Define Security Requirements	Scope defined in Design Document
Leverage Security Frameworks & Libraries	Using well-know up-to-date python libraries
Secure Database Access	Secure access to Amazon RDS through Google Cloud Platform APIs
Encode and Escape Data	DB data stored encrypted
Validate All Inputs	Python methods to check and restrict all input
Implement Digital Identity	APIs only accessible to authenticated users
Enforce Access Controls	Least privilege implemented (User, Admin)
Protect Data Everywhere	Encrypted in DB, AWS TLS protection for data in transit
Implement Security Logging and Monitoring	All activities logged - Admin users have access
Handle All Errors and Exceptions	Try Catch implemented in code to handle exceptions explicitly

SECURITY

University of Essex

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

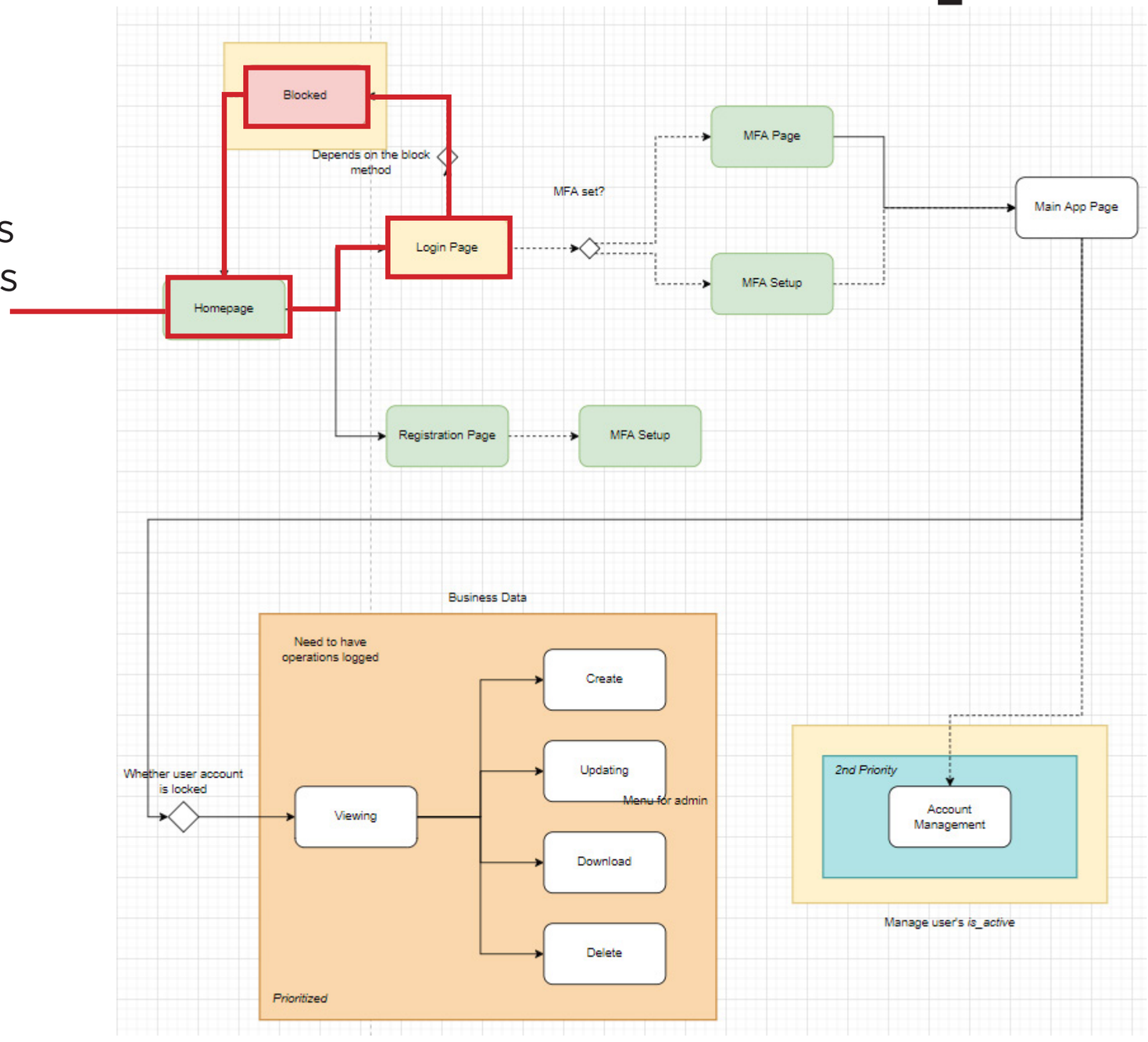
Maintaining records of all data edited/deleted by user.
DB table to store records of all users' login attempts and events

DEMO - Authentication

University of Essex

LIVE CODE DEMO:

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



TESTING

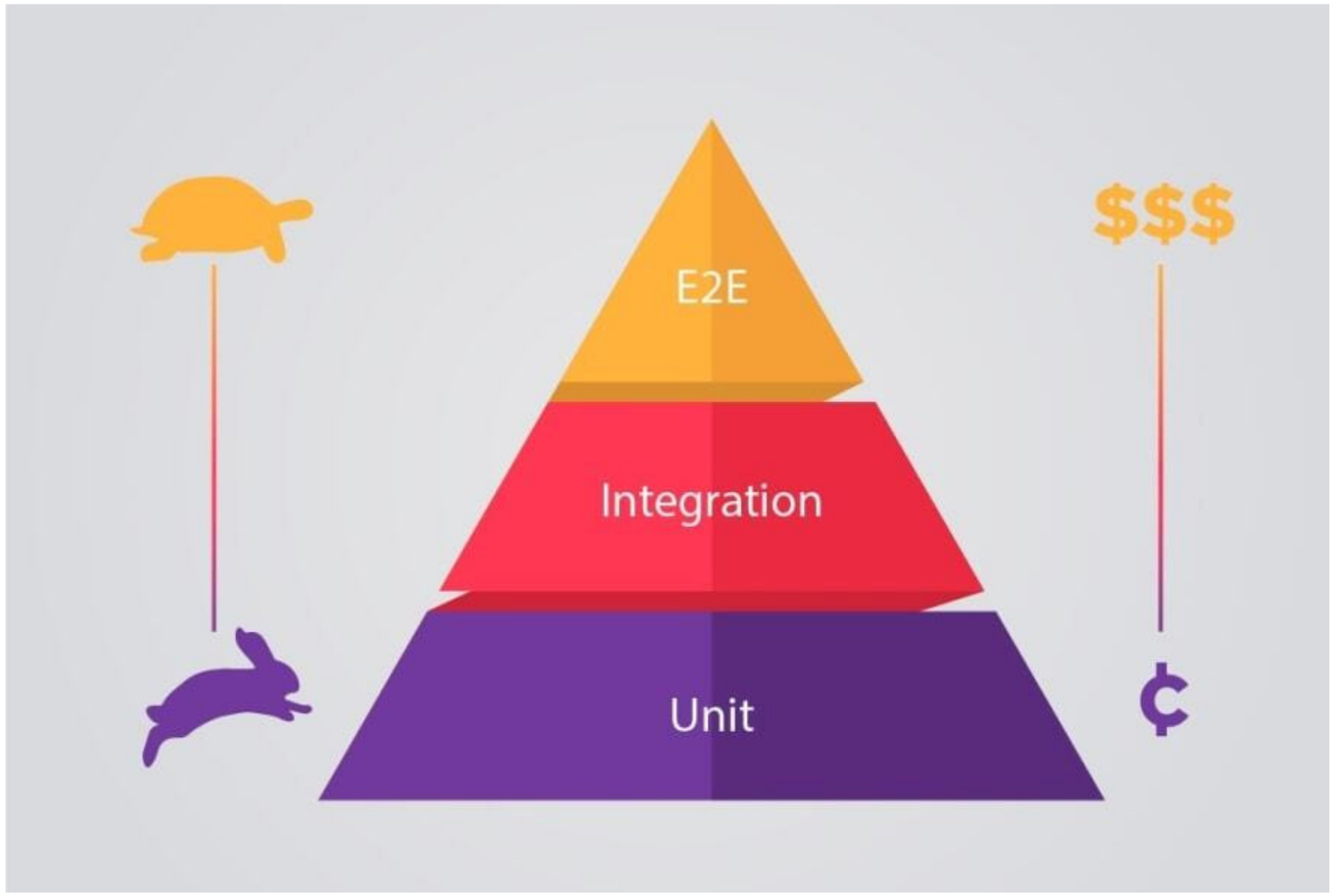
University of Essex

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.



University of Essex

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