



B.Sc. Cloud Computing Architecture

ACCA5025

Assessment – 2, Web Application for UWTSD

University of Wales Trinity Saint David, London

Ioan A Blaj

ID: 2028527

Assessment – 2

User Guide of UWTSD Birmingham's Web Application Deployed on the MS Azure Cloud

User Guide of UWTSD Birmingham's Web Application Deployed on the MS Azure Cloud.....	3
Introduction.....	3
Setting Up the Azure Environment.....	3
2.1 Creating a Microsoft Azure Account and Subscription.....	3
2.2 Setting Up the Azure Web App.....	3
a. Creating a new Web App:.....	3
b. Configuring the Web App settings:.....	4
2.3 Setting Up the PostgreSQL Database.....	4
a. Creating a new Azure Database for the PostgreSQL server:.....	4
b. Configuring the database settings and security:.....	4
c. Creating the database and tables:.....	5
2.4 Connecting the Web App to the PostgreSQL Database.....	5
a. Obtain the connection string for your PostgreSQL server:.....	5
b. Add the connection string as an environment variable in the Web App:.....	5
Deploying the Web Application and Performing CRUD Operations.....	5
3.1 Deploying the Web Application.....	5
3.2 Performing CRUD Operations.....	6
Using the Web Application.....	6
4.1 Accessing the Web Application.....	6
4.2 Managing Student Records.....	6
4.3 Managing Library Records.....	7
4.4 Managing Staff Records.....	7
4.5 Additional Features.....	8
Conclusion.....	8

User Guide of UWTSD Birmingham's Web Application Deployed on the MS Azure Cloud

Introduction

The University of Wales Trinity Saint David (UWTSD) Birmingham has embarked on an initiative to streamline and modernise its departmental services by designing and deploying a state-of-the-art web application. This User Guide aims to provide comprehensive instructions and insights into the deployment and usage of the UWTSD Birmingham Web Application, which is built using Python Flask and hosted on Microsoft Azure.

The primary objective of this web application is to facilitate the seamless management of key departmental records, such as student, library, and staff records. By leveraging the power of Microsoft Azure, the application not only offers improved accessibility and performance but also ensures a secure and reliable environment for storing and managing sensitive information.

This User Guide will walk you through the process of deploying the web application on Microsoft Azure, configuring the PostgreSQL database, and effectively utilizing the application for managing departmental records. Throughout the guide, you will find annotated screenshots to provide visual aid and enhance your understanding of the deployment process and the application's features.

In the following sections, we will explore the various aspects of the UWTSD Birmingham Web Application, including setting up the Azure environment, deploying the web application, and performing CRUD (Create, Read, Update, Delete) operations for each department. By the end of this guide, you will have a solid understanding of the web application's capabilities and how to utilise it effectively.

Setting Up the Azure Environment

Before deploying the UWTSD Birmingham Web Application, you will need to set up the necessary environment on Microsoft Azure. This section will guide you through the process of creating a Microsoft Azure account, setting up the Azure Web App, and configuring the PostgreSQL database.

2.1 Creating a Microsoft Azure Account and Subscription

To get started, we will need to create a Microsoft Azure account and subscription if you don't have one already. Follow the steps below:

- a. Visit the Microsoft Azure website (<https://azure.microsoft.com/>) and click on "Start free" or "Sign in" if you already have a Microsoft account.
- b. Follow the prompts to create a new Microsoft account or sign in using your existing credentials.
- c. Once logged in, sign up for an Azure subscription. Note that new users are eligible for a free trial with a limited amount of Azure resources.

2.2 Setting Up the Azure Web App

After creating your Azure account and subscription, you can now proceed to set up the Azure Web App, which will host the UWTSD Birmingham Web Application.

- a. Creating a new Web App:
 - i. Log in to the Azure portal (<https://portal.azure.com/>).
 - ii. Click on "Create a resource" in the left-hand menu.

- iii. In the search bar, type "Web App" and select the "Web App" option that appears.
- iv. Click the "Create" button to begin the Web App creation process.
- v. Fill in the necessary details, such as subscription, resource group, name, operating system (select "Linux" for a Python Flask app), and runtime stack (choose "Python" with the appropriate version).
- vi. Choose a region that is geographically close to your target audience for optimal performance.
- vii. Click "Review + create" to review your settings and then click "Create" to initiate the deployment of the Web App.

b. Configuring the Web App settings:

- i. Once the Web App has been deployed, navigate to its overview page in the Azure portal.
- ii. In the left-hand menu, click on "Configuration" under the "Settings" section.
- iii. Here, you can add environment variables, such as the connection string for your PostgreSQL database, which will be set up in the next section.

2.3 Setting Up the PostgreSQL Database

To store the departmental records, you will need to set up a PostgreSQL database on Microsoft Azure.

a. Creating a new Azure Database for the PostgreSQL server:

- i. In the Azure portal, click on "Create a resource" in the left-hand menu.
- ii. In the search bar, type "Azure Database for PostgreSQL" and select the option that appears.
- iii. Click the "Create" button to begin the database creation process.
- iv. Choose between "Single Server" or "Flexible Server" based on your needs and budget. For this guide, we will use the "Single Server" option.
- v. Fill in the necessary details, such as subscription, resource group, server name, location, and admin username and password.
- vi. Choose the appropriate computing and storage options based on your expected workload and budget.
- vii. Click "Review + create" to review your settings and then click "Create" to initiate the deployment of the PostgreSQL server.

b. Configuring the database settings and security:

- i. Once the PostgreSQL server has been deployed, navigate to its overview page in the Azure portal.
- ii. In the left-hand menu, click on "Connection security" under the "Settings" section.
- iii. Enable "Allow access to Azure services" to allow your Web App to connect to the database.
- iv. Add your local IP address to the firewall rules if you want to access the database from your local machine for development and testing purposes.
- v. Click "Save" to apply the changes to your connection security settings.

c. Creating the database and tables:

- i. To create the database and tables for the UWTSD Birmingham Web Application, you can either use a PostgreSQL client, such as pgAdmin or DBeaver, or the Azure Cloud Shell.
- ii. Connect to the PostgreSQL server using the admin credentials and server name provided during the server creation process.
- iii. Create a new database for the web application, and then create the necessary tables for each department, such as students, library, and staff.
- iv. Populate the tables with some initial data for testing and demonstration purposes.

2.4 Connecting the Web App to the PostgreSQL Database

Now that we have set up the Azure Web App and PostgreSQL database, you will need to connect them.

a. Obtain the connection string for your PostgreSQL server:

- i. Navigate to the PostgreSQL server overview page in the Azure portal.
- ii. In the left-hand menu, click on "Connection strings" under the "Settings" section.
- iii. Copy the connection string for the Python option.

b. Add the connection string as an environment variable in the Web App:

- i. Navigate to the Web App overview page in the Azure portal.
- ii. In the left-hand menu, click on "Configuration" under the "Settings" section.
- iii. Click on "New application setting" and enter a name for the connection string (e.g., "DATABASE_URL") and paste the copied connection string as its value.
- iv. Replace the placeholders in the connection string with your actual admin username and password.
- v. Click "OK" to add the application setting and then click "Save" to apply the changes.

At this point, you have successfully set up the Azure environment for the UWTSD Birmingham Web Application. In the next section, we will cover the deployment process and how to perform CRUD operations within the application

Deploying the Web Application and Performing CRUD Operations

With the Azure environment set up, you can now deploy the UWTSD Birmingham Web Application and perform CRUD operations to manage department records.

3.1 Deploying the Web Application

To deploy your Python Flask application to the Azure Web App, follow these steps:

- a. Push your application's source code to a Git repository, such as GitHub, GitLab, or Azure Repos.
- b. Navigate to the Web App overview page in the Azure portal.
- c. In the left-hand menu, click on "Deployment Center" under the "Deployment" section.
- d. Choose the appropriate source control provider (e.g., GitHub) and follow the prompts to authorise access to your Git repository.

- e. Select the repository and branch containing your web application's source code.
- f. Click "Save" to start the deployment process. Azure will automatically build and deploy your application to the Web App.

Once the deployment is complete, you can visit your Web App's URL (found on the overview page) to access the UWTSD Birmingham Web Application.

3.2 Performing CRUD Operations

The UWTSD Birmingham Web Application should provide a user-friendly interface for managing records in the student, library, and staff departments. This includes the ability to create, read, update, and delete records.

- a. Create: Users should be able to add new records to each department by filling out a form with the required information and clicking a "Submit" button. The application should validate the input data and store the new record in the appropriate table in the PostgreSQL database.
- b. Read: Users should be able to view records in each department, either by browsing a list or searching for specific records using various criteria (e.g., student ID, book title, or staff member name). The application should retrieve the relevant records from the database and display them in a clear and organised manner.
- c. Update: Users should be able to edit existing records by clicking an "Edit" button next to a record in the list view. This should open a form pre-populated with the current record's data, allowing the user to make changes and save them by clicking an "Update" button. The application should validate the updated data and store the changes in the database.
- d. Delete: Users should be able to delete records by clicking a "Delete" button next to a record in the list view. The application should prompt the user to confirm the deletion before removing the record from the database.

With the UWTSD Birmingham Web Application deployed and functional, you can now proceed to create the User Guide, which will explain the steps for deploying and using the application. This guide will be helpful for staff members who need to manage department records and for other developers who may need to maintain or update the application in the future.

Using the Web Application

In this section of the User Guide, we will provide a detailed explanation of how to use the UWTSD Birmingham Web Application to manage records in the student, library, and staff departments. We will cover the key features and functionalities, including performing CRUD operations (create, read, update, and delete).

4.1 Accessing the Web Application

To access the UWTSD Birmingham Web Application, open a web browser and navigate to the application's URL, which can be found on the Azure Web App overview page. Once the page has loaded, you will be presented with the main interface, which includes navigation links to each department (student, library, and staff).

4.2 Managing Student Records

To manage student records, click on the "Student" link in the main navigation. This will take you to the student department page, where you can perform the following CRUD operations:

- a. Create: To add a new student record, click on the "Add Student" button. This will open a form where you can enter the student's details, such as name, student ID, course, and enrollment date. Fill out the form and click "Submit" to save the new record. The application will validate the data and store the new student record in the PostgreSQL database.
- b. Read: To view existing student records, browse the list on the student department page or use the search bar to find specific records by student ID, name, or course. The application will display the matching records in a clear and organised manner.
- c. Update: To edit a student record, click on the "Edit" button next to the corresponding record in the list. This will open a form with the current record's data, allowing you to make changes as needed. Once you have updated the information, click on the "Update" button to save the changes. The application will validate the updated data and store the changes in the database.
- d. Delete: To remove a student record, click on the "Delete" button next to the corresponding record in the list. The application will prompt you to confirm the deletion before permanently removing the record from the database.

4.3 Managing Library Records

To manage library records, click on the "Library" link in the main navigation. This will take you to the library department page, where you can perform CRUD operations for books and other library resources:

- a. Create: To add a new library record, click on the "Add Resource" button. This will open a form where you can enter the resource's details, such as title, author, publication date, and ISBN. Fill out the form and click "Submit" to save the new record. The application will validate the data and store the new library record in the PostgreSQL database.
- b. Read: To view existing library records, browse the list on the library department page or use the search bar to find specific records by title, author, or ISBN.
- c. Update: To edit a library record, click on the "Edit" button next to the corresponding record in the list. This will open a form with the current record's data, allowing you to make changes as needed. Once you have updated the information, click on the "Update" button to save the changes. The application will validate the updated data and store the changes in the database.
- d. Delete: To remove a library record, click on the "Delete" button next to the corresponding record in the list. The application will prompt you to confirm the deletion before permanently removing the record from the database.

4.4 Managing Staff Records

To manage staff records, click on the "Staff" link in the main navigation. This will take you to the staff department page, where you can perform CRUD operations for staff members:

- a. Create: To add a new staff record, click on the "Add Staff" button. This will open a form where you can enter the staff member's details, such as name, staff ID, position, department, and hire date. Fill out the form and click "Submit" to save the new record. The application will validate the data and store the new staff record in the PostgreSQL database.
- b. Read: To view existing staff records, browse the list on the staff department page or use the search bar to find specific records by staff ID, name, or department. The application will display the matching records in a clear and organised manner.
- c. Update: To edit a staff record, click on the "Edit" button next to the corresponding record in the list. This will open a form with the current record's data, allowing you to make changes as needed. Once you have updated the information, click on the "Update" button to save the changes. The application will validate the updated data and store the changes in the database.

d. Delete: To remove a staff record, click on the "Delete" button next to the corresponding record in the list. The application will prompt you to confirm the deletion before permanently removing the record from the database.

4.5 Additional Features

The UWTSD Birmingham Web Application may also include other useful features to enhance the user experience, such as user authentication, role-based access control, and data export capabilities. These features can be added and described in the User Guide as necessary.

By following this User Guide, users should be able to effectively manage records in the student, library, and staff departments using the UWTSD Birmingham Web Application. The guide serves as a helpful resource for staff members and other stakeholders to understand the application's features and functionality.

Conclusion

The UWTSD Birmingham Web Application serves as a vital resource for managing records within the university's various departments, including student, library, and staff records. This application utilizes the power of cloud computing through Microsoft Azure, a widely used cloud platform, and is built with Python Flask, a popular web framework, and PostgreSQL, a robust database system. Together, these technologies create a reliable, secure, and user-friendly solution for record-keeping.

This User Guide has been designed to offer a comprehensive overview of the web application's essential features and functionalities. It also provides step-by-step instructions for setting up the Azure environment, deploying the application, and using it effectively to manage records. By following this guide, UWTSD Birmingham staff members can maximize the application's potential, leading to increased efficiency and streamlined record-keeping processes.

As technology continues to advance, there may be opportunities for further improvements and enhancements to the web application. Potential upgrades could include advanced search capabilities, data visualization tools, or integration with other university systems. As the application evolves, the User Guide will be updated to ensure that users have the most accurate and comprehensive information available.

In conclusion, the UWTSD Birmingham Web Application plays a crucial role in the university's ongoing digital transformation journey. It not only helps the university serve its students, staff, and other stakeholders more effectively but also maintains a high standard of data integrity and security. By leveraging cutting-edge cloud computing and web development technologies, the university can continue to innovate and adapt to the ever-changing digital landscape.