# Reflection on User Retention Analytics Project

**Introduction**

The User Retention Analytics project aimed to track and analyze user engagement and retention within an application. We implemented a series of database tables to track user activities, such as logins, inactivity periods, and conversion rates. We used Flask for building API endpoints and Docker to containerize the application, making it easily deployable. The project utilized MySQL for data storage and Python for data processing and analytics.

**Learnings Throughout the Course**

I learnt three main things from this project. Firstly, I learnt about database management, secondly, I learnt about data analytics, and thirdly, I learnt about Docker. With regards to Database Management, I deepened my understanding of relational databases, writing complex SQL queries to calculate user retention metrics. I also worked with MySQL to manage user-related data and built tables for storing login and activity data.

Secondly, with regards to Data Analytics, I gained practical experience in analyzing user behavior, particularly focusing on user conversion, activity tracking, and inactive user identification. Using Python, I implemented the analysis, leveraging libraries like mysql-connector for database interaction and matplotlib for visualizing the results.

Thirdly, I learnt about Docker as well. I learned how to containerize the application using Docker. This involved creating a Dockerfile, installing necessary dependencies like Jupyter Notebook, and pushing the container image to DockerHub for deployment. This process ensured the project could be easily shared and run on different environments without the need for manual setup.

**Challenges Faced**

* One of the major challenges was handling user data efficiently, especially with large datasets, which required optimizing SQL queries to run in a reasonable time frame.
* The Docker setup was another challenge. Configuring the container to run Jupyter Notebook and handle dependencies was tricky, but it helped me understand how to use Docker for packaging applications for deployment.

**Experience Working on the Project**

**Problem-Solving Approach**

The project required a combination of technical skills to solve real-world business problems. By analyzing user behavior, we aimed to understand retention rates and how to identify users who may be at risk of becoming inactive. I approached the task methodically by first designing the database schema, then writing SQL queries to extract the necessary data, followed by Python scripts for analysis and visualization.

**Tools and Technologies Used**

* **SQL**: For managing and analyzing user data in the database.
* **Python**: Used for writing scripts to process data and generate insights.
* **Docker**: To containerize the application for easy deployment.
* **Jupyter Notebook**: For documenting the entire analysis and visualizing the results.
* **Flask**

**Key Takeaways**

The three key Takeaways are Containerization, SQL Optimization, and Data Analytics.

* **Containerization**: Docker was a key learning point. It allowed me to package the application and dependencies into a container, which ensures that the project runs consistently across different machines.
* **SQL Optimization**: Writing efficient SQL queries for user retention analysis sharpened my problem-solving skills and improved my understanding of database performance.
* **Data Analytics**: Analyzing user engagement and retention through SQL and Python enhanced my ability to derive actionable insights from data, which can be applied in real-world scenarios.

**Links to GitHub Repository and Docker Hub**

<https://hub.docker.com/r/rithikakesharaju/userretentionanalytics/tags>

<https://github.com/rithika955/UserRetentionAnalytics>