REPORT

ON THE DEVELOPMENT OF A DATABASE FOR THE AIRBNB USE CASE

Developing a database for the Airbnb use case was a valuable chance to get to interact with database design principles. Understanding the requirements, choosing a database managements system (DBMS), design and implementation of the schema, population of the database, running tests, optimizing the database and continuous documentation were the key aspects I had to walk through to get the task done.

After analysing the project requirements, I needed to choose a Database Management System (DBMS). I selected PostgreSQL because it's cost-free, highly scalable, and efficient for managing relational databases (RDBMS). To improve my productivity, I chose DataGrip, an Integrated Development Environment (IDE) developed by JetBrains. DataGrip is the top choice for professionals in database development, delivering unparalleled functionality and efficiency. I was fortunate to discover that DataGrip was available to me for free as a student. I took advantage of its extensive support for various database management systems, advanced code completion that saved me significant time, debugging, and version control features. These tools enabled me to host my project on GitHub and ensure regular updates directly from my IDE.

Next, I had to design a conceptual data model for the database, in which entities were defined entities, their attributes, and the relationships between them in an abstract approach. Then, the logical data model was established using the UML notation. I established a table of entities demonstrating the attributes, datatypes, primary and foreign keys. This task specifically required a lot of awareness of the requirements and the ordinary flow of the platform because any misunderstanding could result of building the schema again from scratch. This is where the main script was created containing not only the commands to create the database, schema, and entities but also the commands creating users and granting them the corresponding privileges to their roles.

In this stage of the project, I started to generate data for the database. This could be done either manually or by using automated scripts. To ensure complete control over the data entered, and to avoid spending too much time due to the need to generate at least 20 records per entity (which could be more depending on cardinalities), I combined both procedures. Specifically, I created a CSV file for each entity with sample data, which I then injected into the database as CSV files. Finally, I exported the data in the form of multi-row insert statements as one file for later use.

After loading the data, I proceeded to test the database to analyse its behaviour. These tests consisted of conducting several operations, such as inserting, updating, deleting, and querying data. These tests were essential to ensure the accuracy and reliability of the database.

At this stage, I reviewed the entire project for bugs and tried to optimize the database's performance by adding indexes. This operation had genuinely reduced latency in terms of data retrieval and data sorting. The full project is hosted on GitHub and could be found using the following link: **https://github.com/Alianger18/Projects_Portfolio**.

Eventually, the database was named "**Airbnb**" created by me on April 25th and modified regularly since then. The database contains a schema "**main**" with 26 entities with at least 20 entries each, provided with indexes to ensure fast data retrieval and sorting. 3 users were created: Superuser, Host and Guest. Each user was granted its corresponding privileges. Its size is around 9613 kB.

The development of this database was challenging but rewarding. The project provided an opportunity to consolidate my knowledge about SQL since I was fortunate enough to be taking a course of the database simultaneously, so it gave me a huge opportunity to apply the knowledge I acquired from the course. I'm most proud of my ability to learn about PostgreSQL and DataGrip. Before this project, I had no idea from where to start using them. That's why I believe the greatest benefit I got from this project was the practical knowledge of using these tools which are widely used by the professionals of the field.