

Choices for Advanced Database Program



Choice of Advanced database program: **Stored Procedure + Trigger**

- Don't have a use case suitable for transactions
- Low likelihood of conflicting transactions
- Trigger
 - **Calculate the average rating for the airline** upon submission/removal/modified of new user rating
- Stored procedure
 - Contains 3 advanced queries with cursors, looping and control
 - Stores advanced queries for **dynamic creative components**
- Incorporating CRUD rules, displaying information to users, accommodating new rating submissions, deleting old ratings, modify existing ratings

Creative Components



- Components
 - Risk Level Meter
 - Pie Chart
 - Airline Reliability Rating Chart
- Added visual aid for users to make easier and faster decisions
- Enhance user understanding of the flight landscape - provide better view of performance distribution among airlines
- Dynamic components create an interactive experience

Challenges Faced



- Indexing
- Creative components added complexity and required external tools
- API integration and endpoints
- Had to learn app integration with GCP
- Importing data into GCP

Changes:

- Database schema had to be changed to accommodate for CRUD ops
- Contained foreign keys to elements in original schema, not directly linked
- Had to add 2 extra tables
 - Authentication for users
 - Separate table for CRUD operations for user ratings

Including NoSQL database



- Define a new dynamic schema and application logic
- Careful handling of tables into collections

Benefits

- JSON gives flexibility to add more data later on since data is restricted to March, improves accuracy
- Shifting to noSQL is easier since app is based on express, which is more compatible with Mongoose
- Better for read-heavy applications, and making constant updates and changes to app architecture
- Will be much less complex than sql
- Less data cleaning cost

Future improvements



Database Design and System Optimization:

- Larger datasets - current data is limited to 3 months
- Optimize imputation - better handling of missing data
- Ability to store and collect real time data - Add more robust architecture for real-time database updates and optimise cloud-based SQL
- Minimize use of cursors - Cursors can be less efficient with large datasets

App improvements:

- Optimising stored procedure related to static pie chart - switch to dynamic
- Make app more user-friendly - more clarity on ratings
- Signout function