Detailed Report on the Course Project: Gift Certificates Database

by Kulinkovich Danila

Introduction

This report details the design, implementation, and analysis of a database system for managing gift certificates. The project includes the development of both an Online Transaction Processing (OLTP) database and an Online Analytical Processing (OLAP) database, along with the necessary ETL (Extract, Transform, Load) processes to move data from the OLTP to the OLAP system. The data is visualized using Power BI to provide insights into various business metrics.

Database Design

ER Diagram
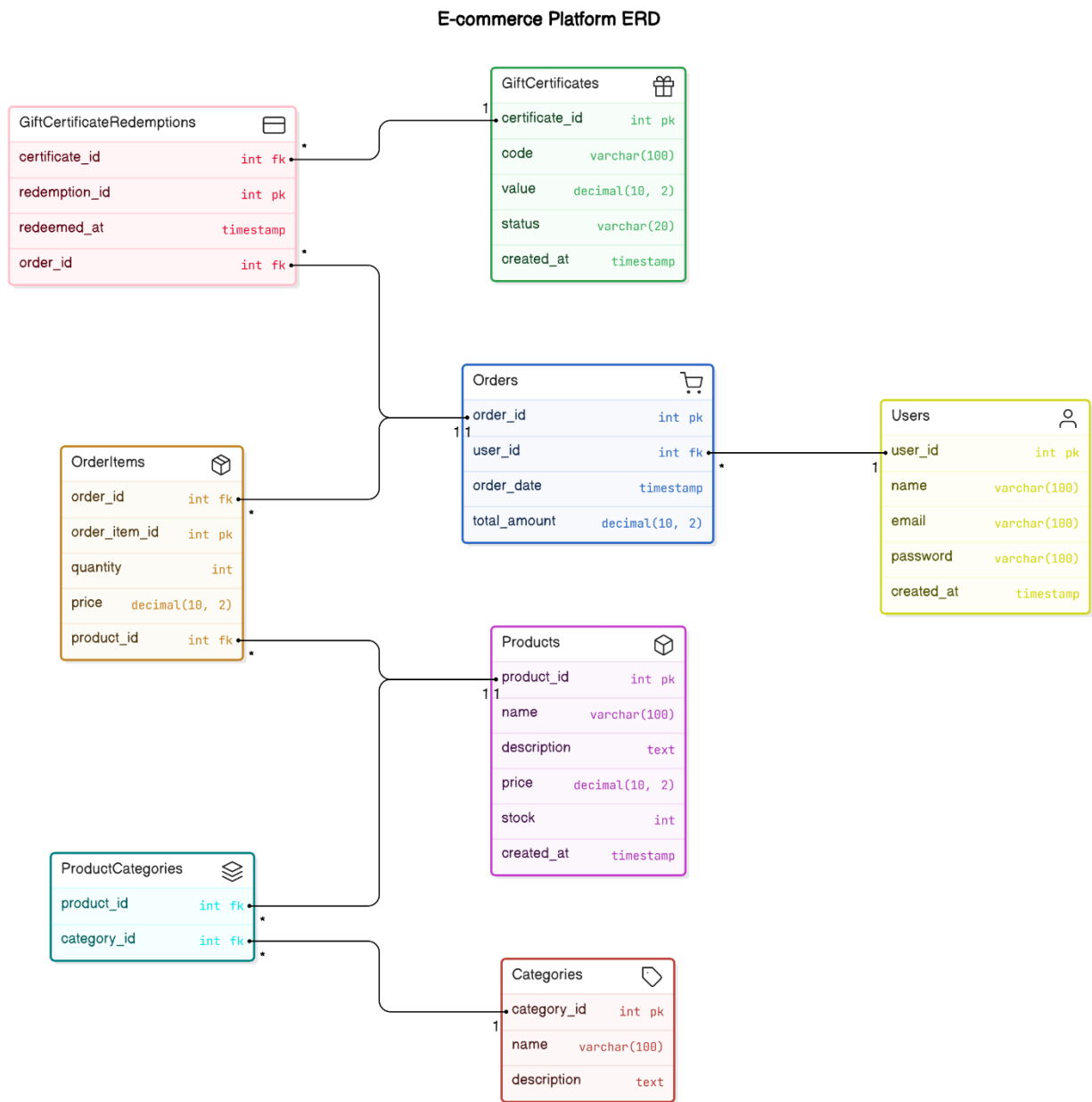
The database design includes several tables that capture different aspects of the business related to gift certificates, users, orders, products, and categories. The relationships between these tables are depicted in the ER diagram above.

1. **Users**
   - `user_id`: Unique identifier for the user.
   - `name`: Name of the user.
   - `email`: Email address of the user.
   - `password`: User's password.
   - `created_at`: Timestamp when the user was created.
2. **GiftCertificates**
   - `certificate_id`: Unique identifier for the gift certificate.
   - `code`: Unique code of the gift certificate.
   - `value`: Monetary value of the gift certificate.
   - `status`: Status of the gift certificate (e.g., active, redeemed).
   - `created_at`: Timestamp when the gift certificate was created.
3. **Orders**
   - `order_id`: Unique identifier for the order.
   - `user_id`: Reference to the user who placed the order.
   - `order_date`: Date and time when the order was placed.
   - `total_amount`: Total monetary value of the order.
4. **OrderItems**
   - `order_item_id`: Unique identifier for the order item.
   - `order_id`: Reference to the order.
   - `product_id`: Reference to the product.
   - `quantity`: Quantity of the product in the order.
   - `price`: Price of the product at the time of the order.
5. **Products**
   - `product_id`: Unique identifier for the product.
   - `name`: Name of the product.
   - `description`: Description of the product.
   - `price`: Price of the product.
   - `stock`: Stock available for the product.
   - `created_at`: Timestamp when the product was created.
6. **Categories**
   - `category_id`: Unique identifier for the category.
   - `name`: Name of the category.
   - `description`: Description of the category.
7. **ProductCategories**
   - `product_id`: Reference to the product.
   - `category_id`: Reference to the category.
8. **GiftCertificateRedemptions**
   - `redemption_id`: Unique identifier for the redemption.
   - `certificate_id`: Reference to the gift certificate.
   - `order_id`: Reference to the order in which the gift certificate was redeemed.
   - `redeemed_at`: Timestamp when the gift certificate was redeemed.

OLAP Database Tables

1. **DimUsers**
   - `user_id`: Unique identifier for the user.
   - `name`: Name of the user.
   - `email`: Email address of the user.
   - `created_at`: Timestamp when the user was created.
2. **DimProducts**
   - `product_id`: Unique identifier for the product.
   - `name`: Name of the product.
   - `description`: Description of the product.
   - `price`: Price of the product.
3. **DimTime**
   - `date_id`: Unique identifier for the date.
   - `date`: The date.
   - `year`: Year part of the date.
   - `quarter`: Quarter part of the date.
   - `month`: Month part of the date.
   - `day`: Day part of the date.
4. **DimGiftCertificates**
   - `certificate_id`: Unique identifier for the gift certificate.
   - `code`: Unique code of the gift certificate.
   - `value`: Monetary value of the gift certificate.
   - `status`: Status of the gift certificate.
   - `created_at`: Timestamp when the gift certificate was created.
5. **FactOrders**
   - `order_id`: Unique identifier for the order.
   - `user_id`: Reference to the user who placed the order.
   - `order_date`: Date and time when the order was placed.
   - `total_amount`: Total monetary value of the order.
6. **FactGiftCertificateRedemptions**
   - `redemption_id`: Unique identifier for the redemption.
   - `certificate_id`: Reference to the gift certificate.
   - `order_id`: Reference to the order in which the gift certificate was redeemed.
   - `redeemed_at`: Timestamp when the gift certificate was redeemed.

ETL Process

The ETL process involves extracting data from the OLTP database, transforming it as needed, and loading it into the OLAP database. The ETL scripts are written in JavaScript using the `pg` library to interact with PostgreSQL databases.

ETL Script

```javascript
const { Client } = require('pg');

const olapClient = new Client({
    user: 'Danila',
    host: 'localhost',
    database: 'Gifts',
    password: '123',
    port: 5432,
});
```

```javascript
const oltpClient = new Client({
    user: 'Danila',
    host: 'localhost',
    database: 'Gifts',
    password: '123',
    port: 5432,
});

async function loadDimUsers() {
    const res = await oltpClient.query('SELECT * FROM Users');
    for (const row of res.rows) {
        const insertQuery = `INSERT INTO DimUsers (user_id, name, email,
created_at) VALUES ($1, $2, $3, $4) ON CONFLICT DO NOTHING`;
        await olapClient.query(insertQuery, [row.user_id, row.name,
row.email, row.created_at]);
    }
}

async function loadDimProducts() {
    const res = await oltpClient.query('SELECT * FROM Products');
    for (const row of res.rows) {
        const insertQuery = `INSERT INTO DimProducts (product_id, name,
description, price) VALUES ($1, $2, $3, $4) ON CONFLICT DO NOTHING`;
        await olapClient.query(insertQuery, [row.product_id, row.name,
row.description, row.price]);
    }
}

async function loadDimGiftCertificates() {
    const res = await oltpClient.query('SELECT * FROM GiftCertificates');
    for (const row of res.rows) {
        const insertQuery = `INSERT INTO DimGiftCertificates (certificate_id,
code, value, status, created_at) VALUES ($1, $2, $3, $4, $5) ON CONFLICT DO
NOTHING`;
        await olapClient.query(insertQuery, [row.certificate_id, row.code,
row.value, row.status, row.created_at]);
    }
}

async function loadFactOrders() {
    const res = await oltpClient.query('SELECT * FROM Orders');
    for (const row of res.rows) {
        const insertQuery = `INSERT INTO FactOrders (order_id, user_id,
order_date, total_amount) VALUES ($1, $2, $3, $4) ON CONFLICT DO NOTHING`;
        await olapClient.query(insertQuery, [row.order_id, row.user_id,
row.order_date, row.total_amount]);
    }
}

async function loadFactGiftCertificateRedemptions() {
    const res = await oltpClient.query('SELECT * FROM
GiftCertificateRedemptions');
    for (const row of res.rows) {
        const insertQuery = `INSERT INTO FactGiftCertificateRedemptions
(redemption_id, certificate_id, order_id, redeemed_at) VALUES ($1, $2, $3,
$4) ON CONFLICT DO NOTHING`;
        await olapClient.query(insertQuery, [row.redemption_id,
row.certificate_id, row.order_id, row.redeemed_at]);
    }
}

async function main() {
    await olapClient.connect();
    await oltpClient.connect();
```

```
    await loadDimUsers();
    await loadDimProducts();
    await loadDimGiftCertificates();
    await loadFactOrders();
    await loadFactGiftCertificateRedemptions();
    await oltpClient.end();
await olapClient.end();
}

main().catch(err => console.error('Error executing ETL process:', err));
```

Data Analysis Queries

Several SQL queries are designed to analyze the data in the OLAP database. These queries help in generating insights and reports.

1. **Total Orders and Sales:**

```
SELECT
    COUNT(order_id) AS total_orders,
    SUM(total_amount) AS total_sales
FROM FactOrders;
```

This query calculates the total number of orders and the total sales amount.

2. **Average Order Value:**

```
SELECT
    AVG(total_amount) AS average_order_value
FROM FactOrders;
```

This query calculates the average value of an order.

3. **Active Gift Certificates:**

```
SELECT
    COUNT(certificate_id) AS active_gift_certificates
FROM DimGiftCertificates
WHERE status = 'active';
```

This query counts the number of active gift certificates.

4. **Redeemed Gift Certificates Value:**

```
SELECT
    SUM(value) AS redeemed_value
FROM DimGiftCertificates gc
JOIN FactGiftCertificateRedemptions gcr ON gc.certificate_id =
gcr.certificate_id;
```

This query calculates the total value of redeemed gift certificates.

5. **User Orders and Spending:**

```
SELECT
    u.name,
    COUNT(o.order_id) AS total_orders,
```
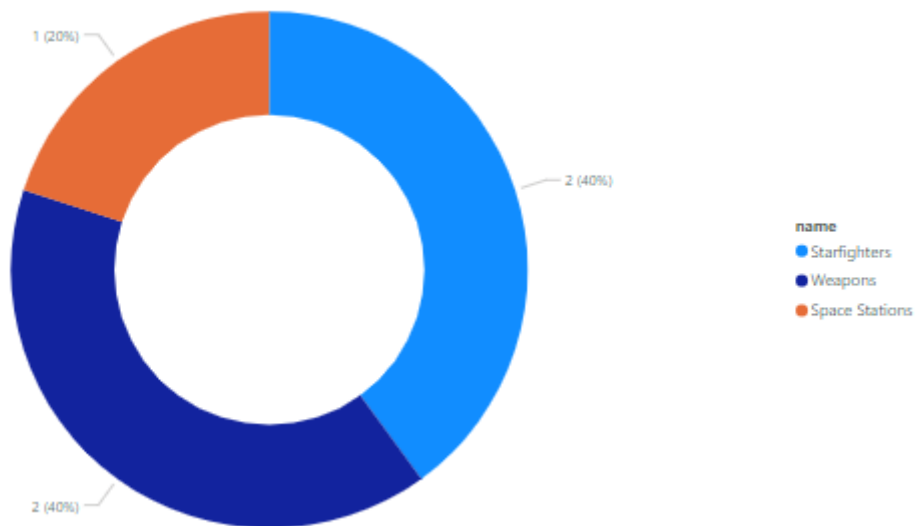
```
       SUM(o.total_amount) AS total_spending
FROM DimUsers u
JOIN FactOrders o ON u.user_id = o.user_id
GROUP BY u.name;
```

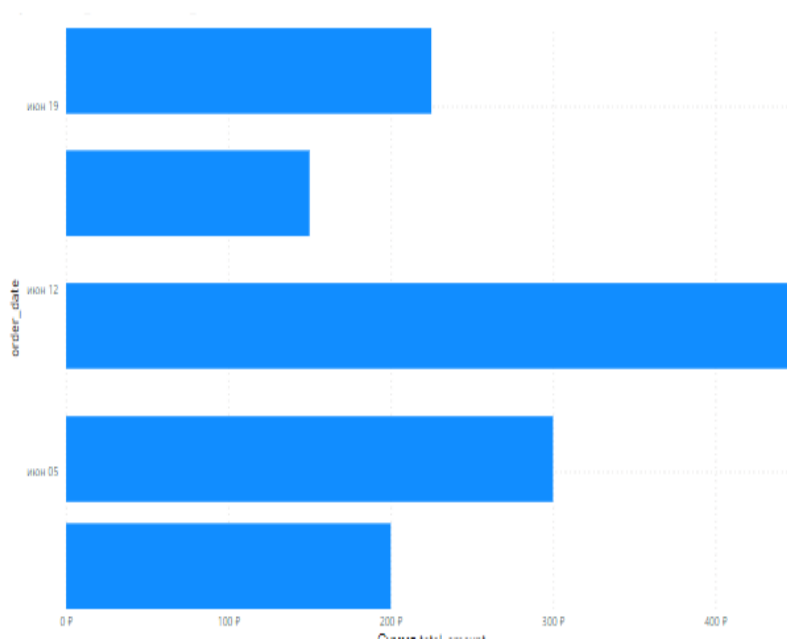This query provides the total number of orders and total spending for each user.

Data Visualization

Using Power BI, various visualizations can be created to provide insights into the business data. Examples include:
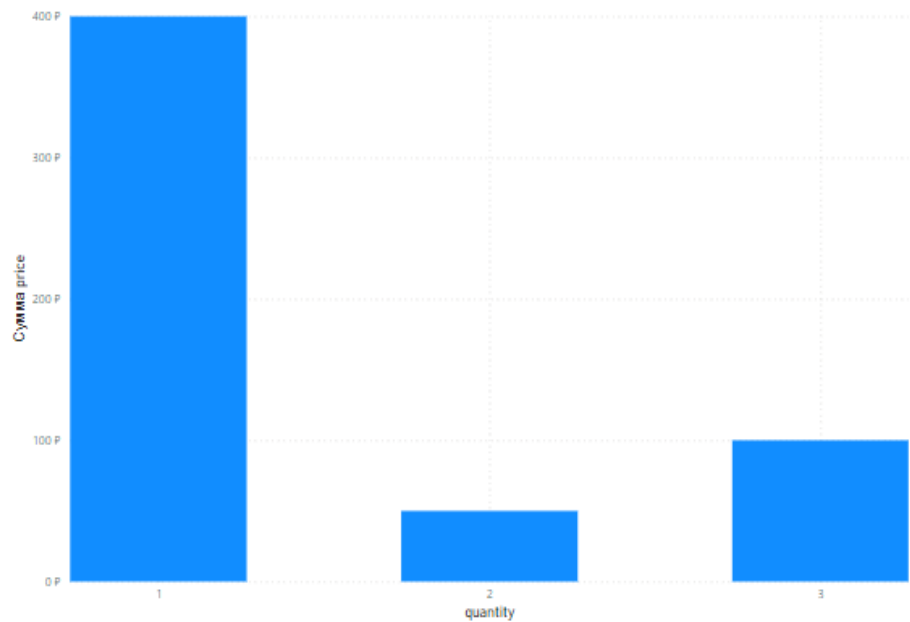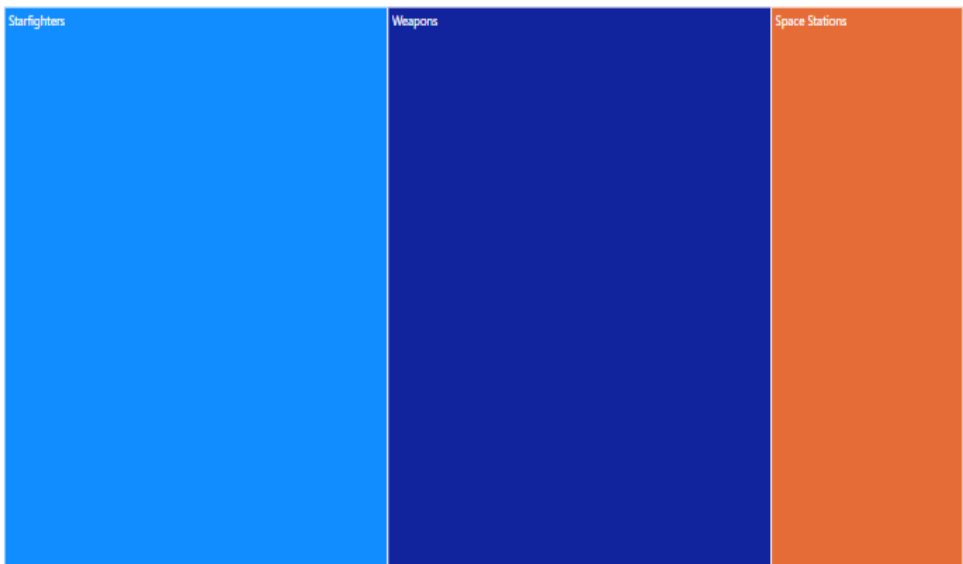
1. **Number of products by category:**
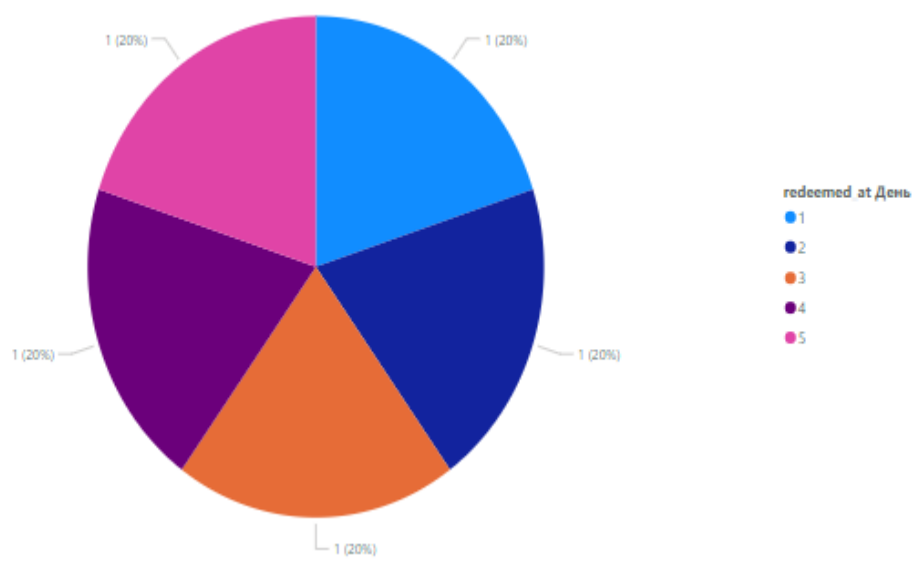

2. **Amounts of expenses by numbers in June:**

3. **Number of goods by cost:**

4. **Cost comparison by category:**

5. **Activated codes by id code:**

Conclusion

This project demonstrates the design and implementation of a database system for managing gift certificates. The ETL process successfully moves data from the OLTP to the OLAP database, enabling complex data analysis and reporting. The use of Power BI provides a powerful tool for visualizing the data and gaining insights into the business metrics.