Software Engineering 3

Newspaper Delivery System

Reflection Document

Introduction

Our project focused on building a system to help newsagents manage newspaper deliveries in a small town or area. The idea was to make their work easier, even if they weren't experienced with computers. The system included features like customer management, delivery lists, and tracking stock. Unfortunately, we didn't finish the billing and geographic routing features, but the project was a great learning experience for all of us.

Group Work

We used Discord to stay connected, share updates, and solve problems together. Each person had specific tasks to focus on, but when someone had difficulties, the team pitched in to find solutions. Everyone worked hard, and we made solid progress as a team.

Who Did What

1. Sergei

- Built the core database methods for inserting, updating, reading, and deleting data for all modules (e.g., customers, orders, and publications).
- Wrote and ran tests to ensure all database methods worked correctly, including handling edge cases.
- Added robust validations for inputs to prevent errors during database operations.
- Focused on testing how modules like orders and customers interacted with the database.

2. Romans

- Improved the efficiency of database queries, especially for fetching data for delivery lists and customer management.
- Worked on a prototype to sort delivery addresses for route optimization, though it wasn't fully completed.
- Linked delivery dockets with orders and customers to ensure accurate tracking.
- Helped write tests for database queries and their integration with the backend system.

3. Francisco

- Developed and tested the order system, including creating, updating, and tracking customer orders.
- Built the area management system, ensuring deliveries were tied to the correct locations.
- Focused on testing the area and order modules to ensure they worked smoothly with the database.

4. Patryk

- Created the customer system, including features for adding, suspending, and updating customer records.
- Fixed issues with the command-line interface to ensure smooth interaction between the user and the system.
- Wrote and ran tests for the customer management module to catch and fix bugs.

5. Nojus

- Developed the module for managing newspapers and magazines, including validations for publication details.
- Created a warning letter system for customers who missed payments or paused deliveries.
- Wrote and tested code to ensure publications and warning letters worked as expected.

What We Wanted to Do

- Generate daily delivery lists for delivery workers, sorted by address.
- Provide daily delivery summaries for the newsagent.
- Track stock for newspapers and magazines to avoid shortages.
- Automate monthly billing for customers (this feature wasn't finished).
- Let customers pause deliveries for holidays.
- Optimize delivery routes using geographic data (this feature wasn't finished).

Challenges Faced

1. Database Complexity: Creating a database that supported all the relationships—customers, orders, areas, and publications—was challenging and took a lot of time.

2.Slow Start due to inexperience of the team, it was hard to make a start on the project because nobody had a clear idea of what they were supposed to do.

3. Not following the development lifecycle and implementing the necessary procedures such as TDD has actually given us a tougher time than if we were to actually go through with the whole lifecycle from top to bottom.
3. Geographic Routing: the area system was complicated and referenced a lot
4. Billing System: Automating billing for customers proved too complex to complete during the project timeline.
5. Testing Integration: Making sure all parts of the system worked together smoothly with the database required extensive testing.

What We Learned

- Prioritization is Key: Missing features like billing and routing showed us how important it is to focus on core functionalities first.
- Database Design: We gained experience in designing and working with databases, including writing efficient queries and handling data relationships.
- Testing: Writing thorough tests helped us find and fix bugs early, making the system more reliable.
- Team Collaboration: Good communication and dividing tasks based on strengths helped us work efficiently as a team.

What Went Well

- Delivery Lists: We successfully created delivery lists for workers, sorted by address.
- Customer and Order Management: The system handled customer data and orders smoothly.
- Warning Letters: The system could generate warning letters for customers who paused deliveries or missed payments.
- Database Integration: The database structure worked well for tracking customers, orders, and publications.
- Extensive Testing: We wrote and ran detailed tests for most features, which helped catch and fix problems.

Task Breakdown

Person	Tasks Completed
Sergei	Built database methods for CRUD operations; wrote and tested database logic; handled data validation; conducted integration tests.
Romans	Optimized database queries; linked deliveries to orders and customers; developed a prototype for geographic routing; tested database functionality.
Francisco	Developed order and area modules; tested database connections for orders and areas; ensured smooth module operations.
Patryk	Built customer management features; fixed command-line issues; tested customer module for bugs.
Nojus	Created publication and warning letter systems; tested publications and warnings for errors.

Reflection

While we didn’t finish some features like billing and geographic routing, the project taught us a lot about software development. We learned how to design systems, write tests, and work as a team. The experience was valuable, and we’re proud of what we accomplished despite the challenges.