Mehmet Akif Yanbak: .Net & Angular Developer

How I Can Contribute

1. Data Modeling: I can make database modeling in accordance with normalization rules for a new project or feature. I can say it's my powerful side.
2. Cache management: Many times caching has been our fearful dream, but I can find a solution to the problem about the most appropriate caching.
3. Algorithm implementation: I can use the proper algorithm for the problem or combine more than one algorithm.
4. Sharing of experience: I do not hesitate to ask questions to my teammates or answer questions from my teammates.
5. Continuous Integration: I can set up a structure where unit tests and selenium tests are performed and documentation is created.

Speeding up the slow system

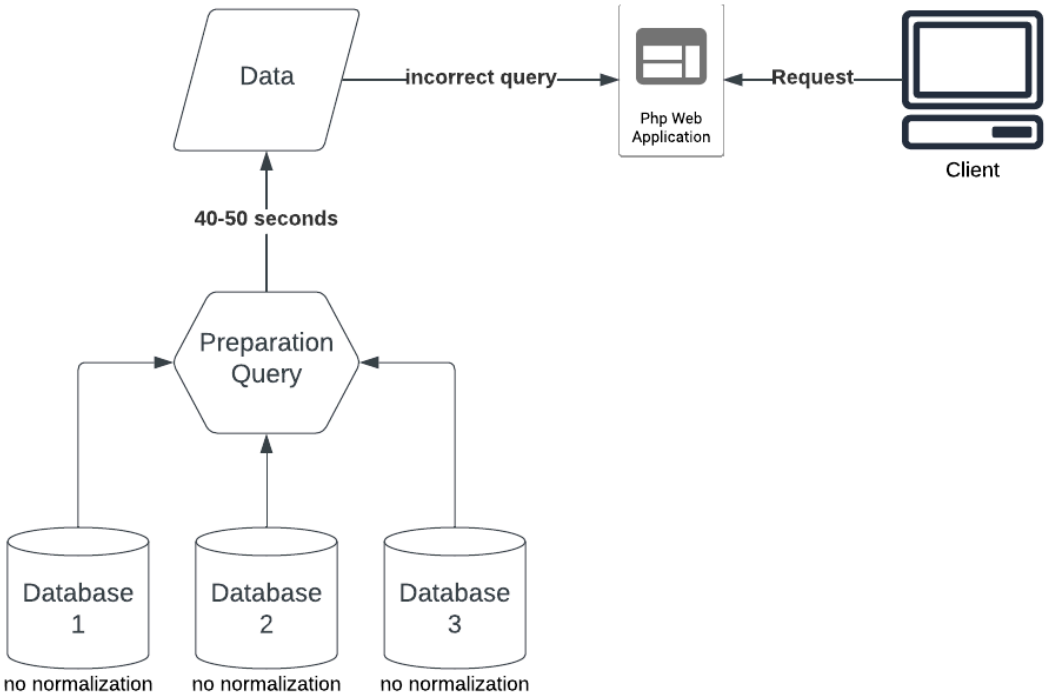
I had an application that 100-150 people used in the field and the page opening and query speed had to be fast.

1. System Optimization Journey: Revitalizing a Lagging Web Application

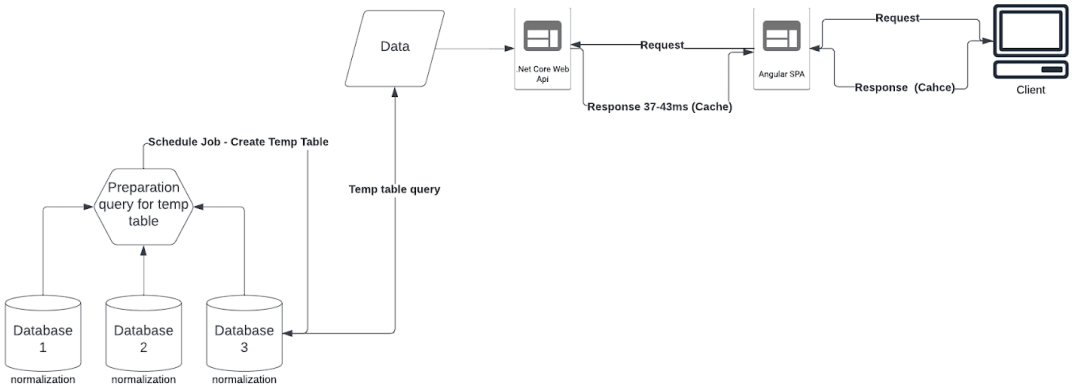
Firstly, I had to diagnose the problem. The client pages were running very slowly. The first problem was that the php application was pulling all the data without pagination. Then I got to the bottom of the pulled data because even if the data was large, it did not need to take 40 seconds. At this point I found our second and biggest problem(!) there was no normalization. In addition, data was being pulled from 3 different databases on 2 different database servers. Moreover, in sql queries, "int" field and "varchar" field were joined and queried. I had to solve these problems respectively and I had a limited time.

2. Diagram

Before:



After:



3. Innovative Caching Strategies

I can say that I made 3 caching mechanisms in total. Firstly, on the angular side, I provided a 1-minute cache on the browser side when pulling requests in json format in the form of "data.json?v=yyyy-MM-dd-hh-mm". Secondly, on the API side, I used a 1-minute memory cache depending on the parameter to the data service. Thirdly, on the database side, I can call the data I threw into the temporary table with the timed task running between 1 minute. Considering that the Angular SPA application is already loaded from the cache, the application opening speed and query speeds have reached the optimum point.

4. Reflections on Decision-making

I could have moved the databases to the same server and collected the application data in a single database, but I did not have enough time.

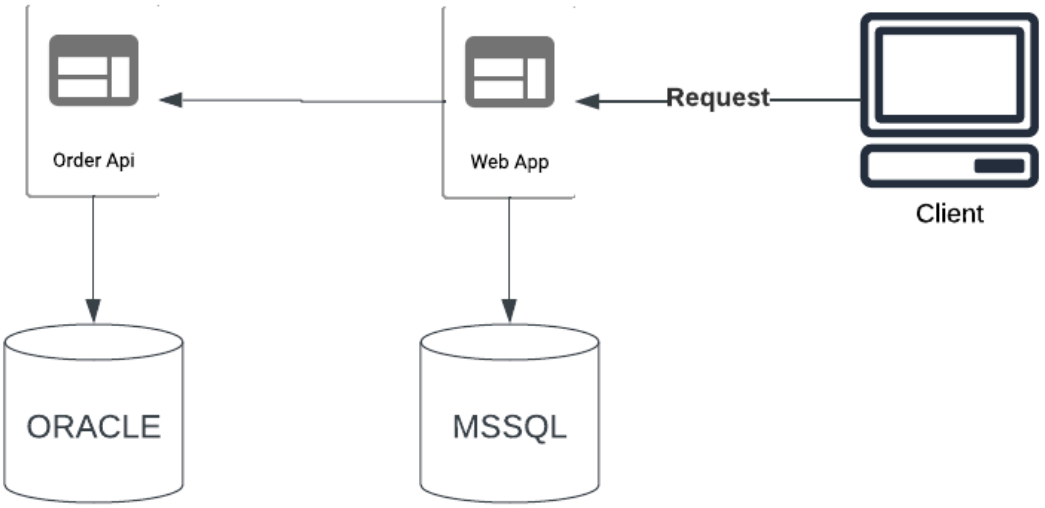
Web application crash problem

The bus ticket web application was becoming unresponsive when the number of requests increased and the application was restarted from iis as a solution.

1. Diagnosing and Resolving Critical Issues

First of all, I started to investigate from the web application to detect the problem. Except for payment information, other information was kept in the MSSQL database. I saw that normalisation was not done as the first problem. After normalisation, although the application accelerated, it was not responding again from time to time. I had to dig deeper. While ticketing, it was connecting to the Oracle database with Order Api. In order to find out exactly where the slowdown was, I created a unique guid for all requests from the client request and followed the request. Finally, I realised that the problem was a problem caused by oracle. I solved the problem by getting technical support from Oracle.

2. Diagram



3. Dependency Challenges

Sometimes dependencies can be too much, but we can solve the problem quickly by making the right moves and getting support from the right people. The system was in complete disrepair and needed help. Writing an app from scratch is sometimes not effective because we might be losing customers because the app doesn't work.

4. Reflecting on Scope and Potential Improvements

I would probably make performance improvements on Order Api, but it was not in my scope.