Project Description: NewsAPI with CRUD Operations, Redis Caching, Authentication, and Exception Handling

I developed a powerful NewsAPI project that offers a comprehensive solution for managing news articles while maximizing performance through a dual database approach. This project demonstrates my skills in various areas, including CRUD (Create, Read, Update, Delete) operations, Redis caching for performance optimization, user authentication, and robust exception handling.

Key Features and Components:

* CRUD Operations: The project includes endpoints for creating, reading, updating, and deleting news articles, following RESTful API conventions.
* Database and Redis Caching: Instead of relying solely on a traditional relational database, I employed a dual-database strategy. News articles are persistently stored in a relational database for data integrity, while Redis serves as an in-memory cache. Redis caching significantly enhances read-heavy operations, providing exceptional response times and scalability.
* Authentication: Robust user authentication is implemented to secure the API. Authorized users can perform actions such as creating or modifying news articles, while unauthorized access is restricted.
* Exception Handling Middleware: To ensure smooth and user-friendly interactions, I integrated an exception handling middleware that gracefully manages errors. This middleware is crucial for maintaining the reliability of the API.

Technologies Used:

* C#: The primary programming language used for building the API.
* ASP.NET Core: The framework powering the web API.
* Redis: Utilized as an in-memory cache to optimize read operations and enhance performance.
* Relational Database: A traditional database system for persistent storage of news articles.
* Authentication: Implemented user authentication using ASP.NET Core Identity or a similar system.
* Middleware: Custom middleware for handling exceptions and providing informative responses.
* GitHub: The project is hosted on GitHub for version control and collaboration.

Challenges and Achievements:

This project had challenges and accomplishments. Successfully integrating Redis caching alongside a relational database required a deep understanding of caching strategies and data consistency. Implementing user authentication ensures data security and privacy, enhancing the overall user experience.

Future Enhancements:

In the future, I plan to expand the project's feature set by incorporating search functionality, user profile management, and user-generated content capabilities. I also aim to fine-tune Redis caching strategies for even larger-scale applications.

GitHub Repository:

You can explore the complete source code, documentation, and project details on my [GitHub repository](https://github.com/YashaMusayelyan/NewsAPI).