Research plan

Introduction

ICAP is designed to be an all-encompassing enterprise scale web platform. It would replace a lot of existing separated out communities and platforms for talking about airsoft and bring it all into a single platform that allows for direct-messaging, commisioning, buying & selling, live-updating forums and group matching. An interesting aspect to look at is the combination of scalable/enterprise architecture with messaging and live-updating. Enterprise architecture mostly uses async messaging to function so how would live-updating forums and messaging function within the architecture?

Research question

The research for the project aims to provide more insight into the abovementioned combination of enterprise architecture and live messaging:

```
What are the architectural and technological considerations to take into account
when designing ICAP's live-updating forum and real-time messaging system to
accommodate 100,000 concurrent users, while ensuring sub-second message delivery
and live forum updates?
```

Some of the most interesting sub-questions that can be formulated based off this research question are:

1. What strategies or mechanisms can be implemented to ensure that clients receive message updates only for the specific forum sections they are actively viewing?
2. How do existing enterprise-scale applications achieve close to real-time messaging?
3. Which database would be the best choice for storing and retrieving messages and forum data?
4. Which architecture types exists and which one would be most suited to achieve the responsiveness required for real-time messaging and forum updates?

Methods

Using the methods provided by ICTResearchMethods we can assign some examplary strategies to the sub-questions in order to get results back for the research questions that is easy to fact-check.

1. What strategies or mechanisms can be implemented to ensure that clients receive message updates only for the specific forum sections they are actively viewing?

    - **Prototyping:** Build prototypes of different strategies or mechanisms for message updates and test their performance.
    - **Existing Product Analysis:** Observe how existing forums or messaging platforms implement such mechanisms and analyze their approaches.

2. How do existing enterprise-scale applications achieve close to real-time messaging?

    - **Literature Study:** Select a few representative enterprise-scale applications and conduct in-depth studies to understand their messaging architecture and strategies.
    - **Benchmark:** After conducting the literature studies, a benchmark test can be used to evaluate the speed of the different prototypes.

3. Which database would be the best choice for storing and retrieving messages and forum data?

    - **Literature Study:** Review existing literature and case studies on the use of databases for similar applications to gather insights into best practices.
    - **Database Benchmarking:** Benchmark different database systems (e.g., SQL vs. NoSQL) by simulating message storage and retrieval loads to assess performance.
    - **Community Research:** See if others have already performed comparisons for which database would be best for use in forums and messaging.

4. Which architecture types exists and which one would be most suited to achieve the responsiveness required for real-time messaging and forum updates?

    - **Design Pattern Research:** Conduct controlled experiments to compare the real-time responsiveness of applications using different architectures. Measure response times as a test.
    - **Expert Interviews:** Gather feedback from users and developers who have experience with all kinds of architectures to understand their perceptions and experiences.
    - **Literature Study:** Conduct a comprehensive review of existing literature and research on the topic to gather insights and findings from previous studies.