

Data Structures Activity

- Select at least two different data structures to hold the data associated with the list of functional and non-functional requirements that you defined for Task 1.
- Justify your data structure choices.
- Select at least one academic paper, which might be similar to the work of Abeykoon et al. (2020).
- Use your sourced information to support your data structure choices.

This seminar preparation activity was dependant on Task 1 however there was no task 1 so instead I decided to base it on a social networking site.

Example requirements:

Functional Requirements

- Users can register, log in, and log out with a unique username and password
- Users can edit messages after they are sent
- Users are notified when they receive a message

Non-Functional Requirements

- The system must handle authentication requests within 2 seconds under normal load conditions
- The message must be updated for all users in a chat within 0.1 seconds
- The system must comply with GDPR to safeguard user data
- The web application should allow developers to resolve critical bugs within 2 hours of detection
- The website must be resilient to any kind of attacks, including DDoS and XSS attacks

Data Structures

The optimal data structure for notifications is a stack as it is a LIFO (Last in First Out) data structure so you can view the most recent notification with $O(1)$ time complexity (Ozsvald & Gorelick, 2020).

The optimal data structure credentials is a dictionary as stores key value pairs allowing you ensure unique credentials due to a $O(1)$ time complexity for accessing elements (Ozsvald & Gorelick, 2020).

For messages I would use a database but if you wanted to use a specific data structure I would probably suggest using a queue as for deleting for a queue the time complexity is $O(1)$ (Ozsvald & Gorelick, 2020).

References

Gorelick, M. & Ozsvald, I. (2020) High Performance Python: Practical Performant Programming for Humans. 2nd ed. California O'Reilly Media