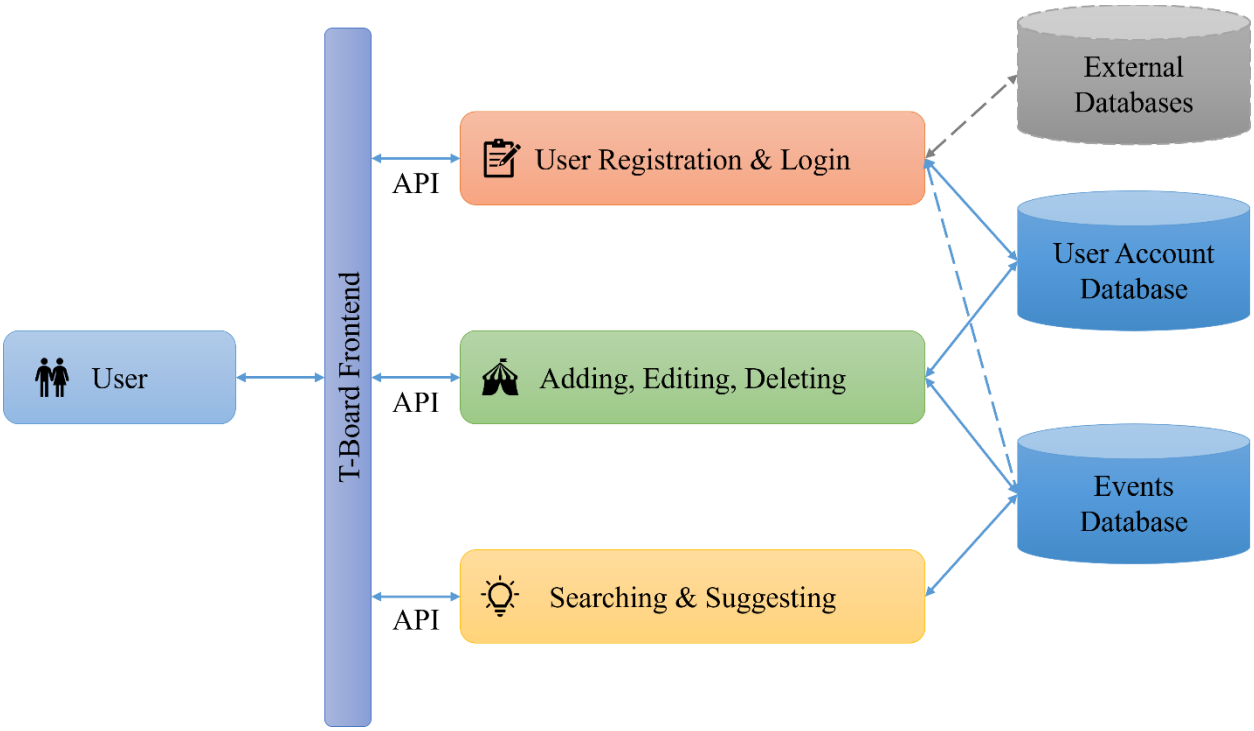
ECE444 Lab 4: Architecture Design

An Xi 1006896183

1. Microservices

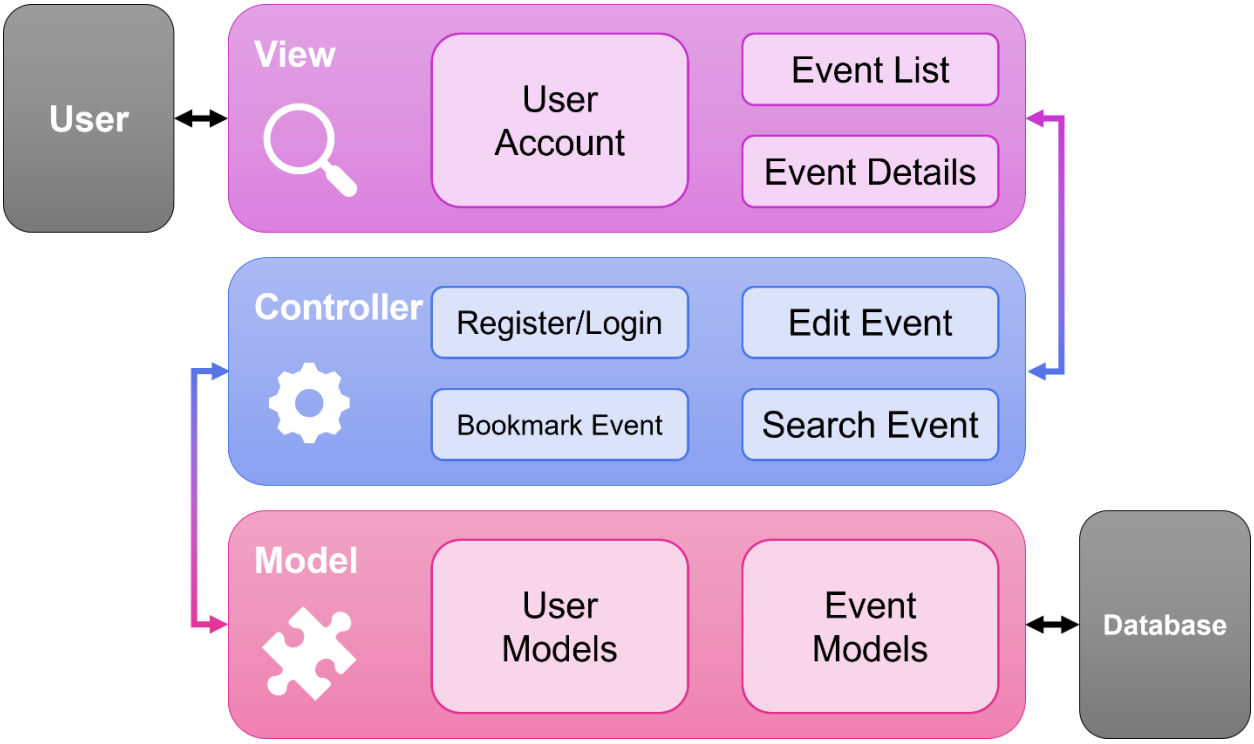


Above is a possible interpretation of the architecture of the application “T-Board” that our group “3D3G” is going to develop. The remote procedure calls exist between every two services out of the three but are omitted in this graph for the purpose of conciseness.

The three services have been decomposed based on the functions they hold. The Register & Login service handles all functions related to user accounts – which can include optional features that may be implemented in the future such as event bookmarking and personal event participation histories. Since this application targets mainly members of U of T, external databases such as UTORid login systems can be referred to as a means of verification; however due to limited power that our group holds in a negotiation with officials of the university, references from such databases may be highly unlikely.

The other two services mainly focus on the events themselves; however, the event queries and suggestions have been separated due to the reason that it does not involve edits to the events’ database itself, as well as the reason that the number of queries to events is expected to be higher than the number of edits to them, therefore resources and algorithms that handles different scales of queries could be implemented to best suit the needs of each service.

2. Monoliths



Above is an interpretation of an alternative architecture of the application "T-Board" which utilizes the Monolithic MVC pattern. The View layer directly interacts with the user – it reads user inputs and displays the return values from the Controller layer. The Controller layer acts as a coordinator between the other two layers – it finds suitable models to handle different types of user inputs and renders the raw response data from the models to be easily readable. The Model layer directly interacts with different databases, returning and updating values based on the instructions from the Controller layer accordingly. All three layers can only communicate directly with the layer directly above and below it.

3. A Discussion Between the Structures: Scalability

One factor of this application that our group values is its scalability: we deem that our app will be scalable if it performs well and minimizes failures under high user loads. Based on the graphs that have been listed above, it may be realized that the Microservices approach surpasses the other in terms of scalability. The Microservices approach divides all functions into smaller parallel services; and each service is highly customizable. As mentioned above, for one event that have been created, it may have been viewed and searched upon for many more times; with the Microservices approach, instead of letting all the searches lagging the Controller layer, the team can dedicate time for faster algorithms and server resources for handling more user accesses at a time to better improve the overall performance of the application.

References

All images in this document have been created using Microsoft PowerPoint. A PDF copy of this document as well as all images can be found at the GitHub repository:

https://github.com/AndyXi163/ECE444-F2023-Lab4