



UNIVERSITY
OF MIAMI



UTutorMe

*Reliable and quality
tutoring near you!*



Recap

UTutorMe provides students requiring academic assistance with instantaneous access to quality and qualified nearby tutors consisting of previous or current students. Students no longer have to face the overwhelming stress of finding a quality tutor last minute!

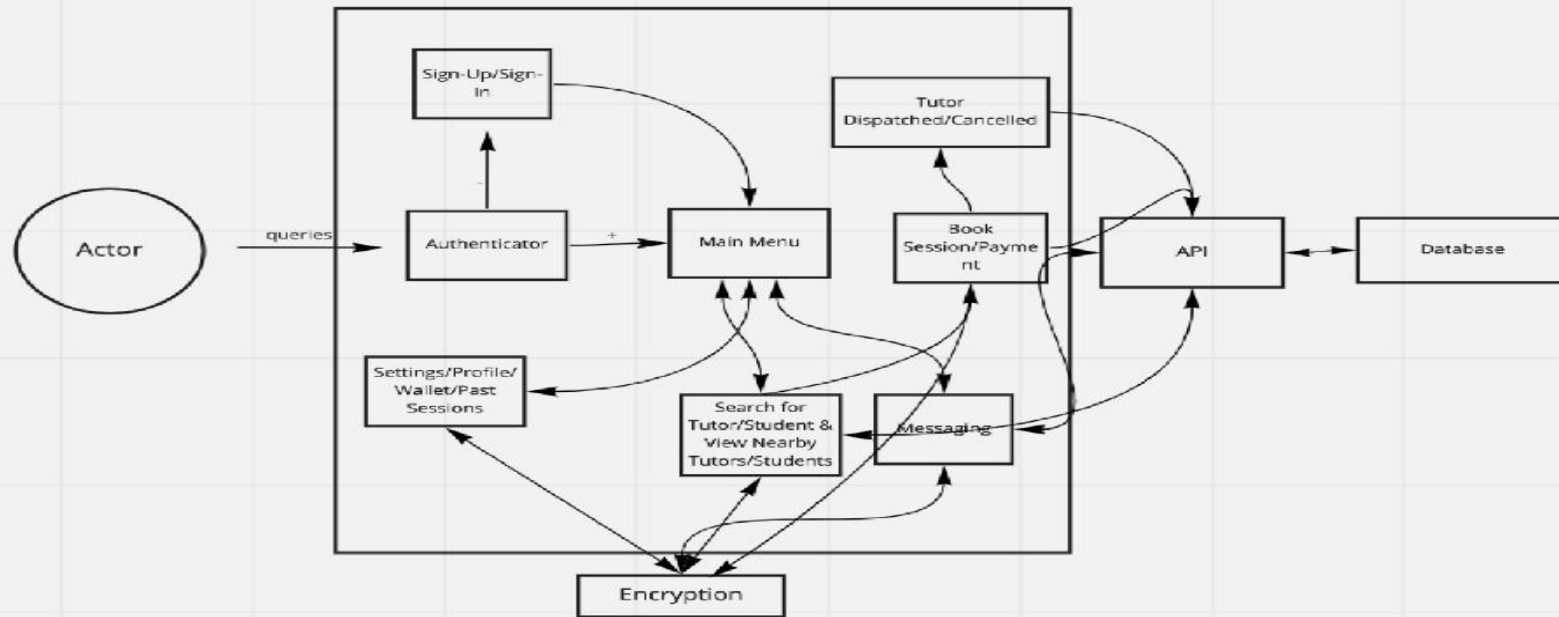


System Overview

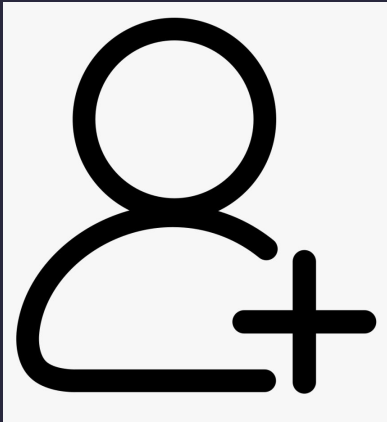
The system is comprised of three main components:

- 1) **The UI:** The interface the user interacts with by doing things like entering details, course selection, accepting and rejecting tutors/students, and payment details.
- 2) **The Database:** Implemented by Spring Boot, this will be used to store valuable information such as user data, session details, reviews, and bookings.
- 3) **API Layer:** Used for connecting the database to the UI layer. It will also incorporate google maps API to display student and tutor information.

System Components Description (using system diagram)



Actors



New Users



Registered Users



Database Itself

Architectural Style

UTutorMe will employ a 3-tier architecture. The 3-tiers consist of the UI layer, business logic layer, and the service layer.

1. **UI layer-React Native:** easily provides support for different operating systems.
2. **Business logic layer:** this will enable students to book a tutor as well as allow tutors to accept student requests, which will all be handled by a panel that will receive both requests and manage the operations accordingly.
3. **Service layer-Spring Boot:** for storing tutor/student information, descriptions, and sessions.

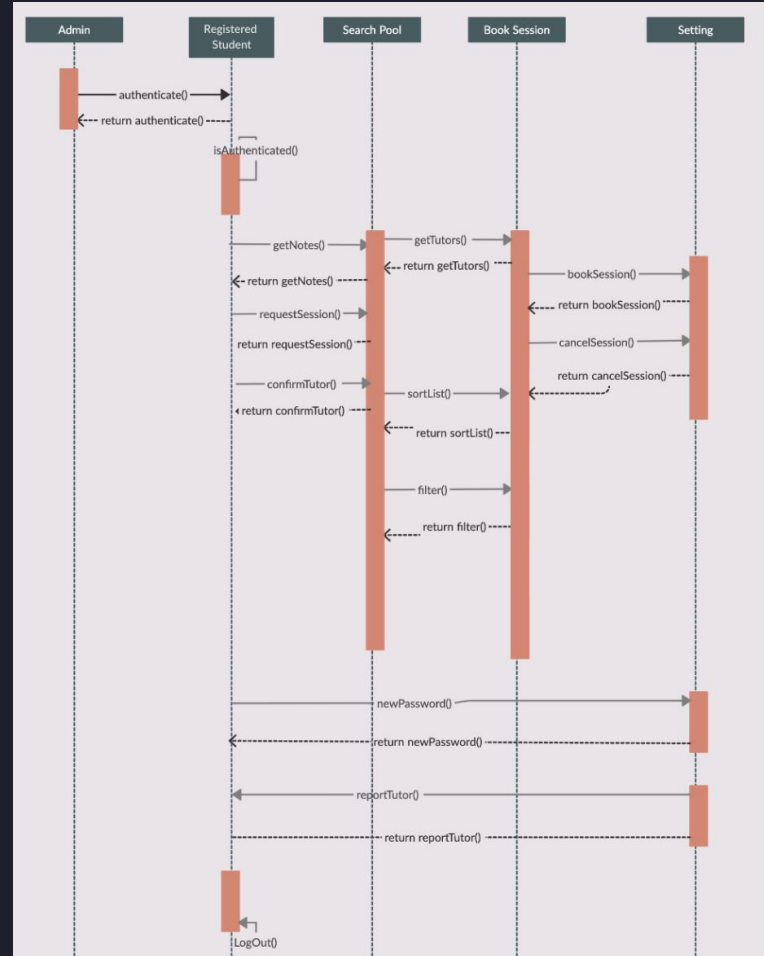
Design Patterns

1. **“Facade”**: hides the complexities of our system and provides an intuitive interface
2. **“Strategy”**: allows use to respond to run-time instructions using one of a family of algorithms
 - a. Useful for things like estimated ETA

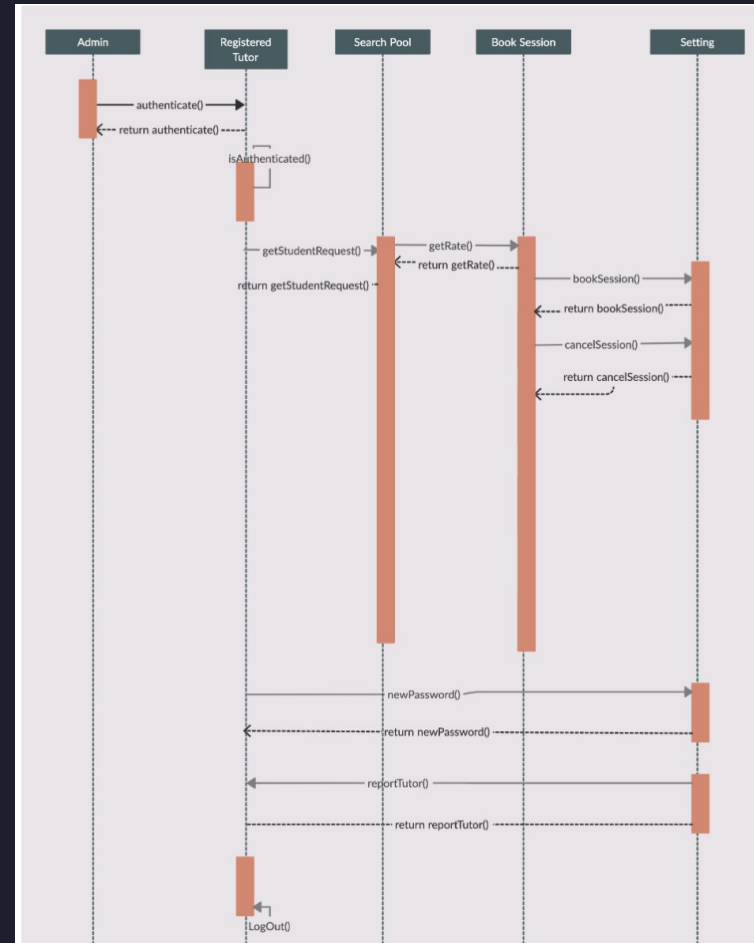
Frameworks

- 1) **Framework Of Choice:** Spring (Java, Hibernate, MySQL/PostgreSQL)
- 2) **Justification:**
 - UTutorMe will be a mobile application primarily
 - Spring has extensions for web/mobile applications and for usage in android developments
 - Lightweight, easy-to-test, powerful abstraction, loose coupling, fast development, loose coupling, promotes inversion of control and dependency injection

Sequence Diagram (1)

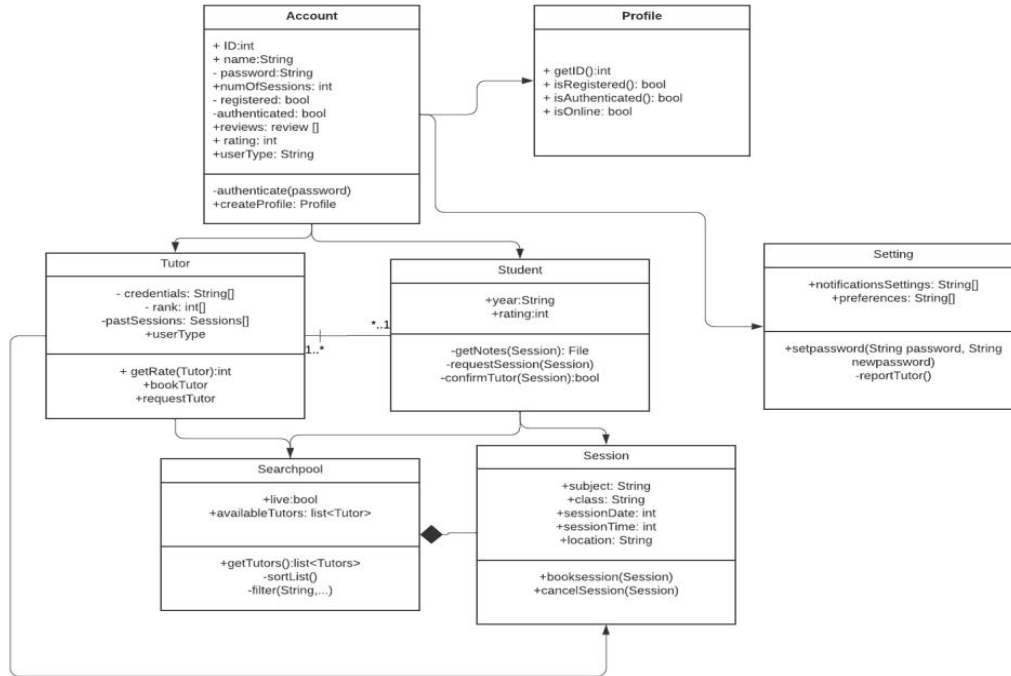


Sequence Diagram (2)



Class Diagram

UTutorMe Class Diagram



Project mockup/prototype implementation

Clickable Wireframe:

<https://www.fluidui.com/editor/live/preview/cF9hc1RCVFdUWnFLTERQTnJHSVRHa1JDQU1wUnV4NG9LRg==>

Closing slide with github

<https://github.com/19exceraya1/CSC431>



/Meet The Team



Rayan N. Excellent



Chris Wu



Dido Franceschi



Pavan Gudoor





Questions?

