California Law Database

By: Matthew Randolph & Ramon Sanchez

Project Objective

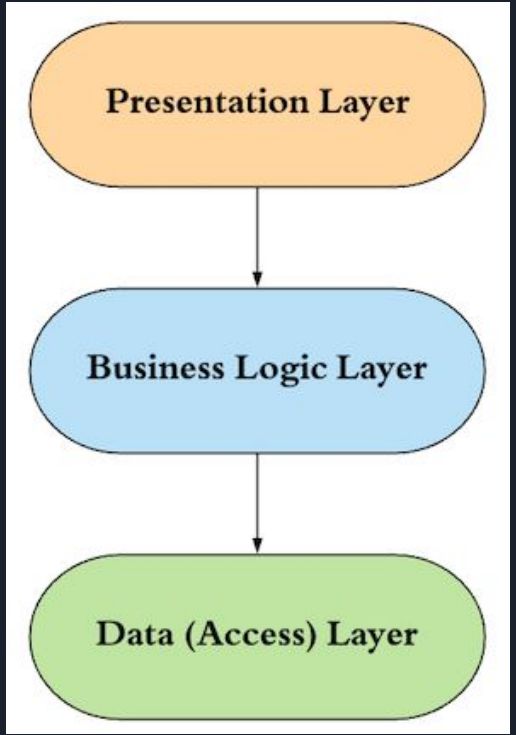
Summary: The "California Law Database" project is designed to establish a robust and user-centric platform, enabling efficient navigation and management of laws within the state of California. This digital repository seeks to be an invaluable tool for legal professionals, students, and other audience, simplifying access to, and interaction of legal data.

Key Functionalities

- **Visualize Data**: Enable users to effortlessly explore and view detailed law data.
- **Data Management**: Empower administrators with the capabilities to add, update, and remove law data to maintain an accurate and up-to-date legal database.
- **Category Exploration**: Allow users to effortlessly navigate through different law categories, enhancing ease of data retrieval.

Tech Stack

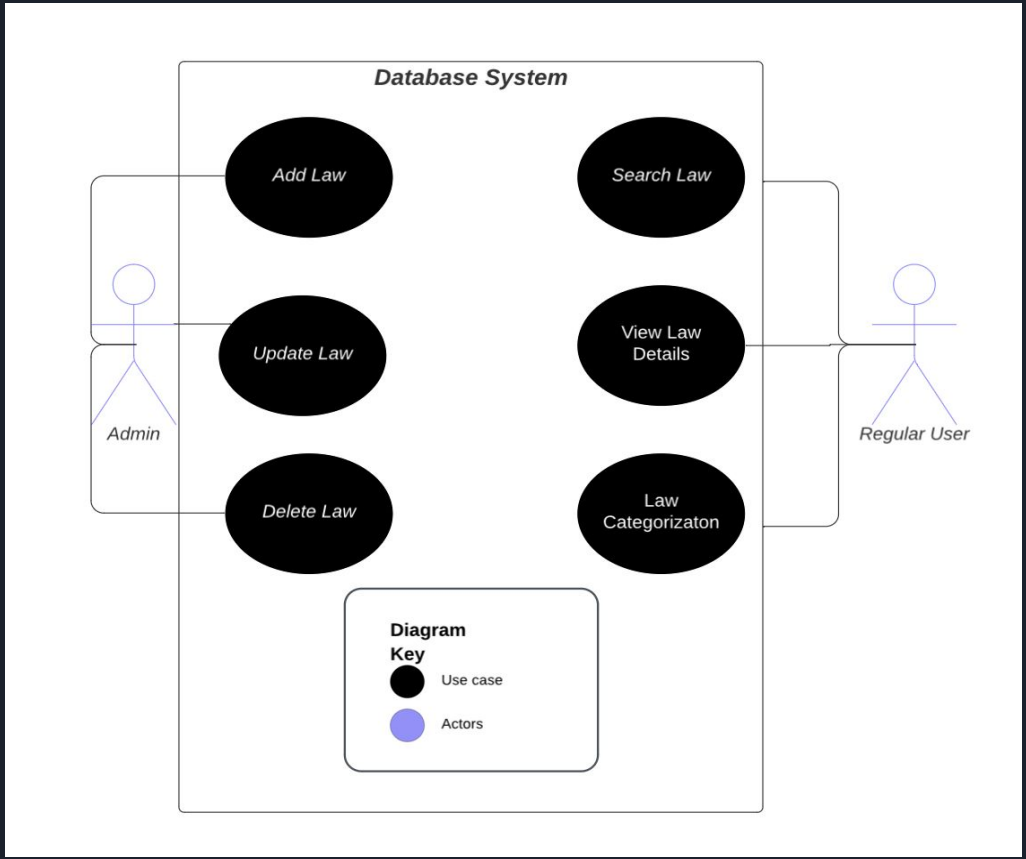
- **Back-End**: Utilizing the Spring Framework and Java to create a stable, secure, and efficient back-end, ensuring reliable data operations and preservation of data integrity throughout the system.
- **Front-End**: Implementing React, which harnesses JavaScript, HTML, and CSS, to construct a dynamic, interactive, and user-friendly interface, optimizing user engagement and experience by providing a responsive and intuitive environment.
- Firebase Authenticator
- MySql
- Sqlite3



Main Use Cases:

- **Search Law**: Users should be able to pinpoint specific laws using keywords or unique identifiers
- **Add Law**: Admins should be able to add new laws into the database.
- **Update Law**: Admins should be able to modify the existing laws' details.
- **Delete Law**: Admins should be able to remove laws from the database.
- **View Law Details**: Users can view detailed information of a particular law.
- **Law Categorization**: Users can browse through collections of laws grouped by related tags.

UML Use Case Diagram

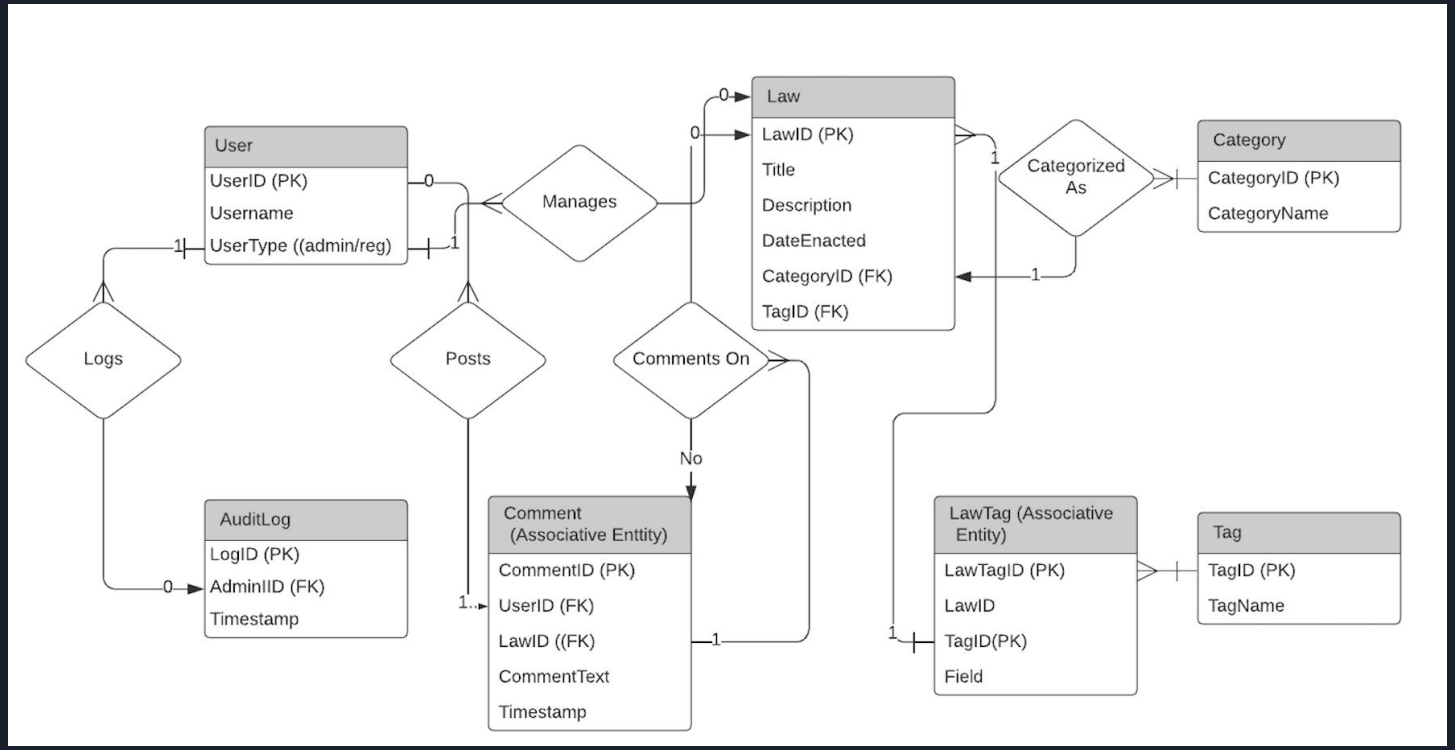


E/R Diagram

User
- User-Comment-Law (Many-to-Many)
- User-Comment (Many-to-One)
- User-Logs (One-to-Many)

Law
- Law-Category (Many-to-One)
- Law-LawTag-Tag (Many-to-Many)

Comment
- Comment-Law (Many-to-One)



Relational Schema

User

- u_userid, u_username, u_usertype

Law

- l_lawid, l_title, l_description, l_dateenacted, l_categoryid, l_tagid

Category

- c_categoryid, c_categoryname

AuditLog

- a_logid, a_adminid, a_timestamp

Comment

- c_commentid, c_userid, c_lawid, c_commenttext, c_timestamp

LawTag

- l_lawtagid, l_lawid, l_tagid, l_field

Tag

- t_tagid, t_tagname