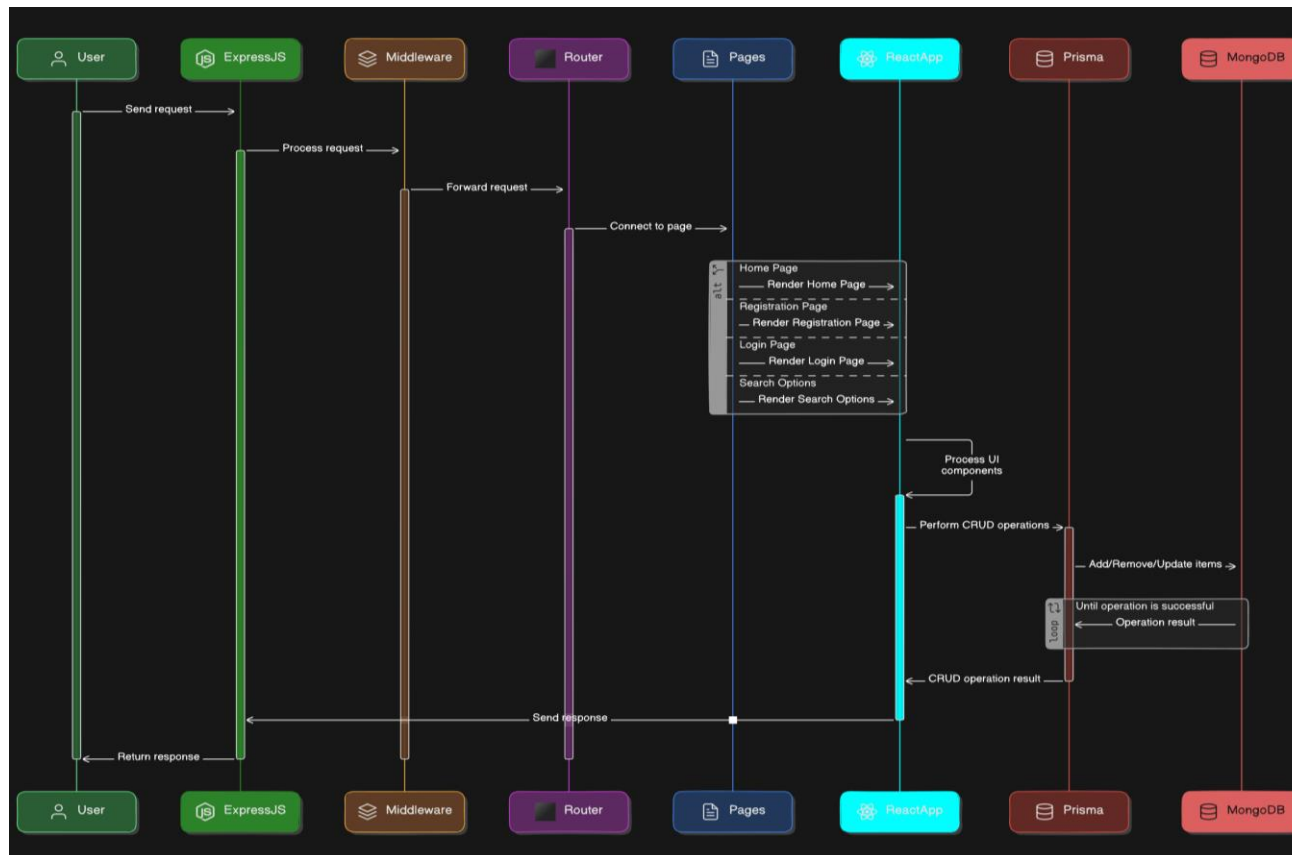


## Requirement Gathering and Analysis Phase Technology Stack (Architecture & Stack)

Date	29-06-2024
Team ID	SWTID1720064122
Project Name	Project – House Hunt
Maximum Marks	4 Marks

### Technical Architecture:



**Table-1 : Components & Technologies:**

S.No	Component	Description	Technology
1.	User Interface	Users can search for homes, register, log in, and manage their saved homes using this web application.	HTML, CSS, JavaScript / React Js etc.
2.	API Gateway	Requests for APIs are routed to the relevant backend services via a single point of entry from the user interface.	Cloud-based API Gateway service
3.	Search Service	backend service that searches the user's database for suitable house listings and returns the results.	Node.js with MongoDB
4.	Listing Service	home publishing functionalities are managed by a backend service (create, update, delete listings).	Node.js with MongoDB
5.	Database	retains all application data, including stored homes, user data, and house listings.	MongoDB
6.	Geolocation Service	service that uses user location data to customize search results, provided the user gives permission.	Google Maps
7.	User Service	Backend service for managing user profiles, logins, and registrations.	Node.js with MongoDB
8.	Cloud Storage	Retailers posted pictures of their homes in order to save money and be scalable.	Cloudinary
9.	Messaging Service	Backend service that makes it easier for users and landlords to communicate in real time.	Node.js

**Table-2: Application Characteristics:**

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Dynamic user interfaces, reliable backend support, scalable data storage, type-safe database interactions, and interactive maps for property locations are all made possible by open-source frameworks.	Prisma, Express.js, React.js
2.	Security Implementations	Protect user information and restrict access to different areas such as user profiles and house listings.	bcrypt ,JSON web tokens(JWT), Middleware-Token Verification, Role Based Access
3.	Scalable Architecture	helps to ensure that the application runs smoothly even during periods of high traffic by enabling it to adjust to a rising user base and data volume.	MongoDB, RESTful API, Node.js
4.	Availability	reduces downtime and ensures that consumers may always use the house hunt services.	Multiple Node.js instances
5.	Performance	provides quick response times for listing details, house searches, and other features, making the user experience seamless.	Prisma ORM, MongoDB, Node.js, Express