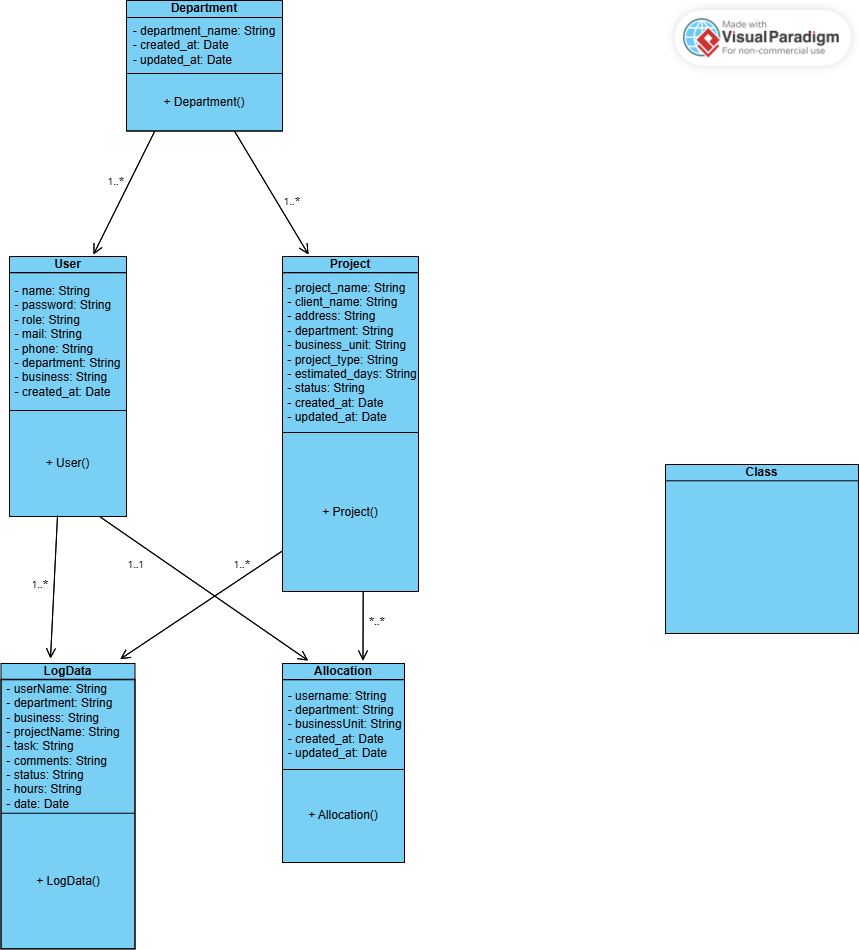
**Class Diagram:**

**Notification System Implemented:**

There are more options to do the notification system.  
Here I have used the nodemailer. We can also use tools also like mailtrap etc..   
It was just a simple application so I did not use the tools and it will lead to cost also when exceeds limit.  
But the useful thing in using tools is thatwe can manage our mails there itself.  
**What I Have Done:**

* Created an endpoint /user/new to handle new user registrations.
* Implemented a function to generate a random password for new users.
* Used **JWT** for secure email verification links with a 1-hour expiry.
* Configured **Nodemailer** for sending verification emails with a reset-password link.
* Developed another endpoint /user/reset-password to handle password resets using the token and update the user's password in the database.

**Key Improvements:**

1. **Sensitive Information Handling**: Replace plaintext credentials (secretKey and email/password) with environment variables for enhanced security.(Dotenv)

**Security Mechanism Implemented:**

**What I Have Done:**

* **Strong Security Mechanisms**: Implemented robust measures to ensure the application's security and safeguard user data.
* **DOMPurify Integration**: Used DOMPurify to sanitize user inputs, preventing XSS (Cross-Site Scripting) attacks.
* **Rate Limiting**: Configured rate limiting on API endpoints to protect against brute force and DDoS attacks.
* **JWT Verification**: Secured user authentication and email verification processes using JSON Web Tokens (JWT) with expiration handling.
* **Input Field Sanitization**: Ensured all user inputs are sanitized to prevent injection attacks like SQL injection or script injection.
* **Prevention of Open Redirection**: Added safeguards to prevent malicious open redirection vulnerabilities.

**Key Improvements:**

1. **Logging and Monitoring**: Add centralized logging for security events to monitor and detect potential breaches in real-time.
2. **Strict Content Security Policy (CSP)**: Implement a strict CSP header to further mitigate XSS and data injection risks.

This was the some of the security mechanisms that I have implemented  
but still there are lots of things to improve security  
  
Then due to less amount of time I am unable to create web page analytics  
but it not too much complicated   
we can use versal or render to deploy our mern application  
after that we can able to use google analytics on our application  
  
Thank you sir !!