Issue management

Prepared By: Patel Neel(200280111513)

Guided By: Urmi Unagar

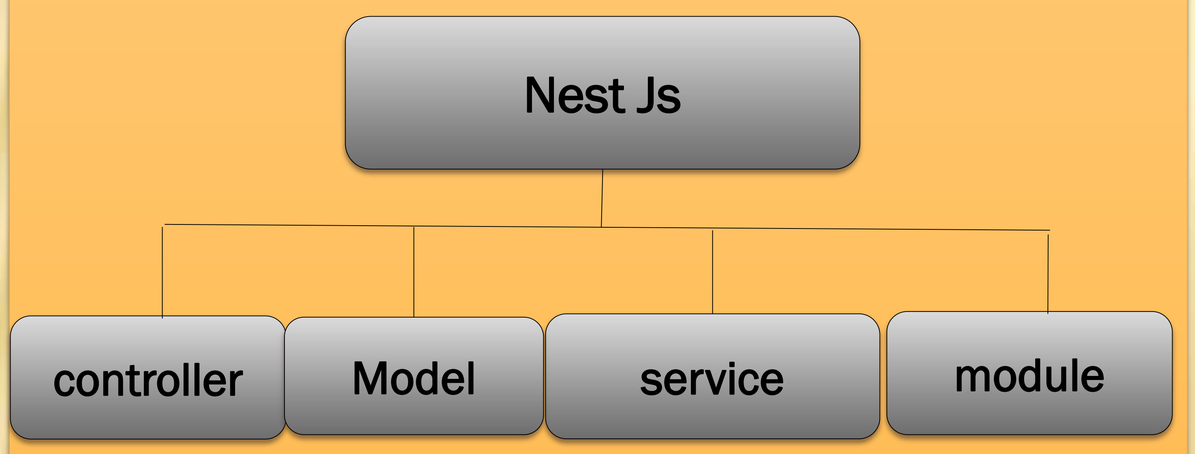
Introduction

NestJS simplifies issue management for software development. We'll explore error handling, logging, and third-party integrations like Sentry to streamline issue management in NestJS applications.

issue management based on nest js

Nest js working principle :

NestJS is a Node.js framework that uses a structured approach to development with modules, controllers, providers, pipes, and dependency injection. It's built on Express.js and influenced by Angular.



1. Controller:-

NestJS controllers handle incoming requests and define the routes for an application. They're decorated classes that define HTTP methods and their routes. You can use various decorators to extract data from requests and test the methods using libraries like Jest.

2. Model:-

NestJS models define data entities and their schema, enabling efficient CRUD operations with validation and hooks.

3. Service:-

In NestJS, services are classes that contain business logic and perform specific tasks. They are reusable, modular, and can be easily tested. Services can be injected into other components and provide a flexible and customizable approach to building scalable and maintainable applications.

3. Module:-

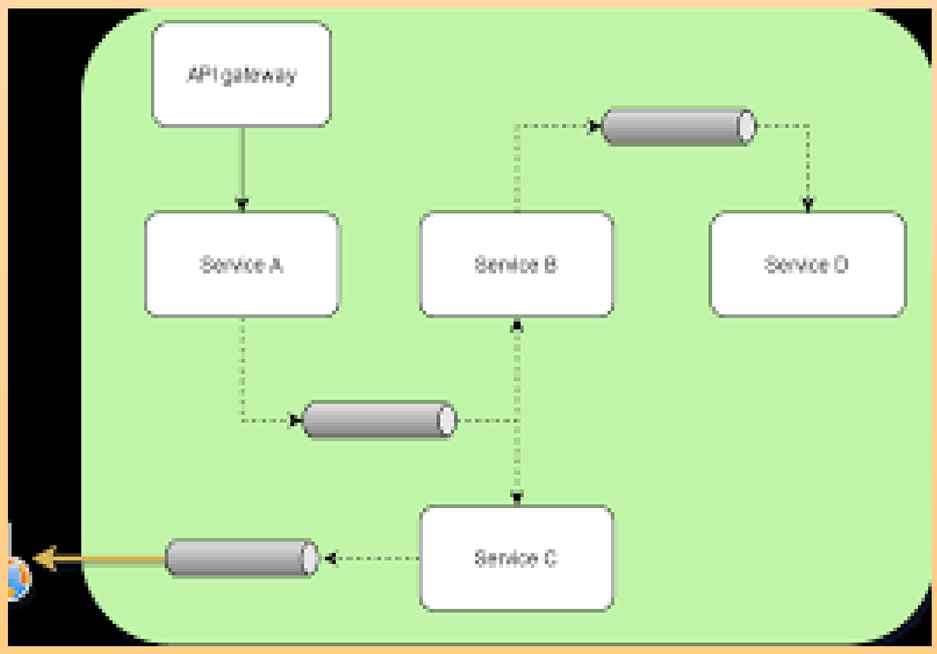
NestJS modules group related components and organize your application into reusable blocks of code for a modular and scalable architecture.

Working

Issue management working principle :-

Issue management in NestJS involves detecting and handling errors efficiently using middleware, logging, and third-party integrations like Sentry.

Issue management using nest js



Features

- **Middleware**:NestJS provides middleware to catch and log errors that occur during application runtime.
- **Logging**: NestJS supports logging with Winston, enabling developers to easily track errors and diagnose issues.
- **Error handling**:NestJS can handle errors by returning appropriate HTTP responses, such as user-friendly error messages or specific error codes.
- **Third-party integration** :NestJS supports third-party integrations like Sentry, which can be used to monitor and manage errors in real-time.

Conclusion

Issue management is critical for maintaining app stability. NestJS provides efficient tools like middleware, logging, and third-party integrations for detecting, logging, and handling errors, ensuring a smooth user experience.