

SpecialProject.io Commenting API

Johnny Egan | Wes Robbins | Spencer Reap

A dark blue diagonal gradient bar that starts from the bottom left and extends towards the top right, covering the lower half of the slide.

System Architecture

Adonis Framework guided and had influence on the patterns used in design

- Observer Pattern
 - User considered the subject
 - Commenter plays role of the observer
- Model View Controller
 - Baked into Adonis framework
 - Model is direct interface between the program and the SQL database

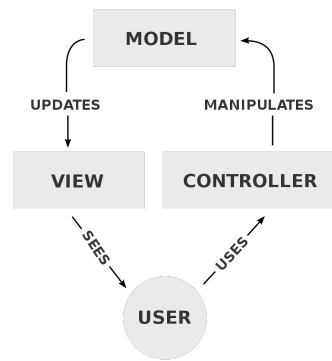
System Design

Design Pattern 1: Observer Pattern

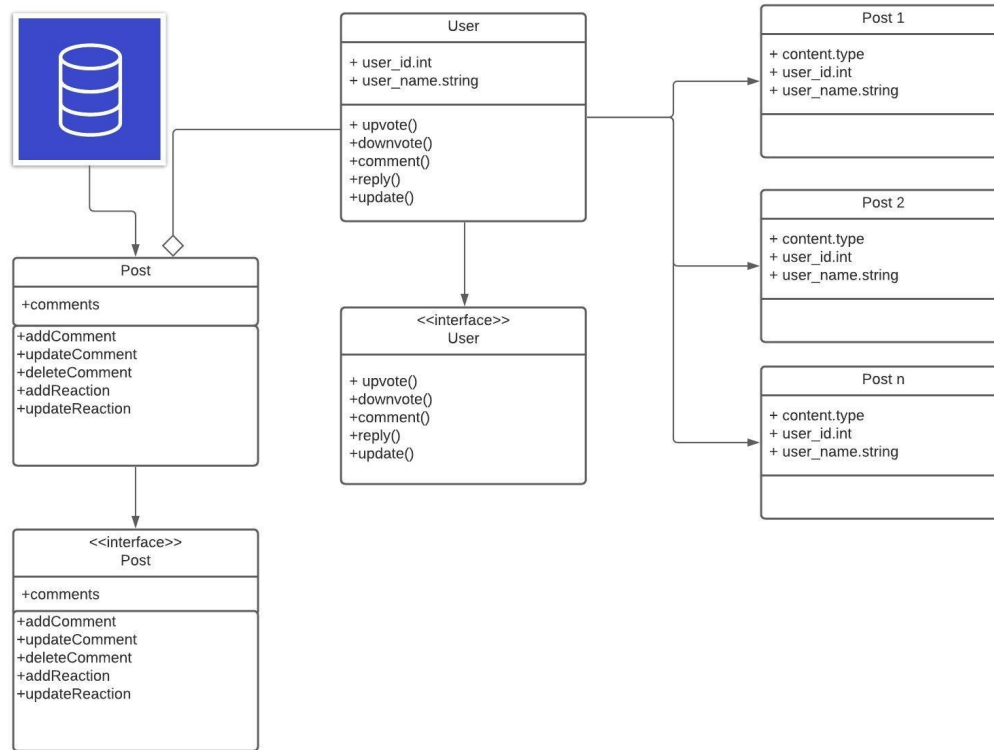
- Pattern designed to represent sending/receiving messages to subscribers
 - For this case, commenting on posts, as well as replying to other comments

Design Pattern 2: Model View Controller

- Part of Adonis Framework
- Model accesses database and updates view
- View is responsible for UI and receiving input
- Controller responsible for logic between View and Model



UML Design



Design Choices for Extensibility/Maintainability

Extensibility

- Design patterns used allow for easy integration of new features to front end or API features
- Modular design

Maintainability

- Organized file structure
- Design patterns also help maintainability
- Functions with low-coupling high-cohesion

Quality Assurance

We used several testing methods to assure guarantee quality and reliability of our software.

We also used user testing we had two people test our API which helped us improve usability.

Testing Methods

Selenium - black box testing that makes sure all user interface functionality on the website remains working

Jest - white box testing that checks all function give correct output.

Postman - Used for testing the API. Many different http requests can be sent to see if they break the API.

Using CI/CD

We used all three of the these testing methods throughout the duration of development.