

HBnB Evolution — Technical Documentation

Introduction

The **HBnB Evolution** project is an advanced redesign of the original HBnB platform, aimed at improving system scalability, maintainability, and usability. This document serves as a comprehensive technical blueprint for the project.

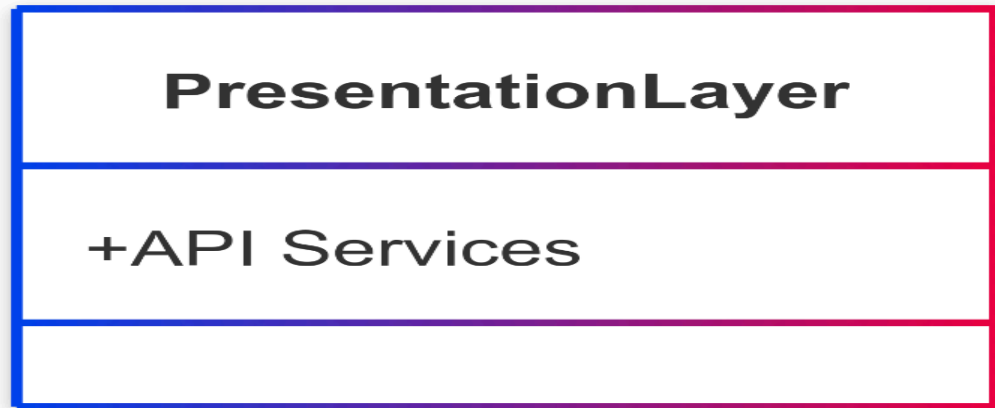
It compiles the key architectural diagrams and explanatory notes that will guide the development process and serve as a reference for the engineering team.

The document covers:

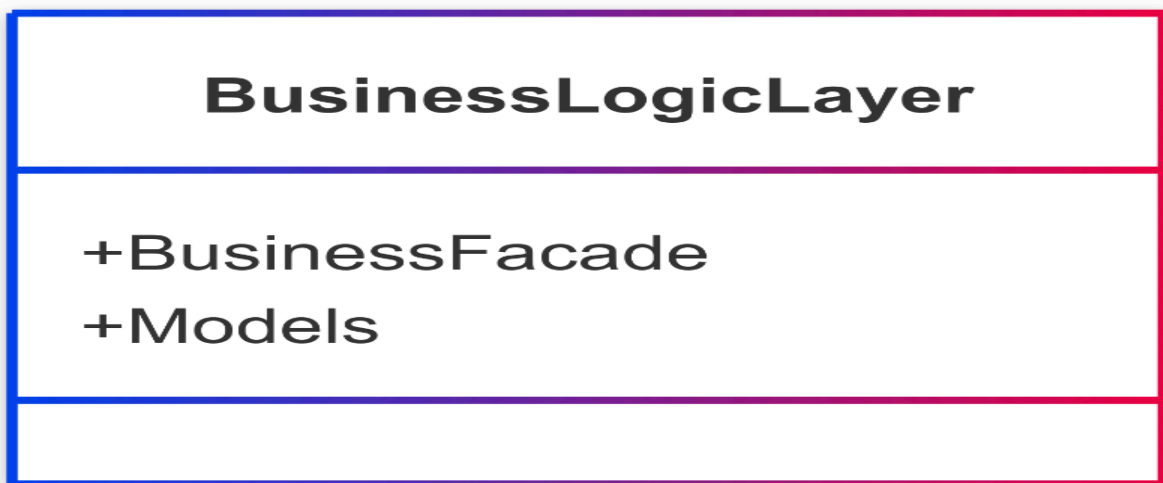
- The high-level architecture of the system.
- The detailed design of the business logic layer.
- The interactions between system components through selected API calls.

High-Level Architecture

High-Level Package Diagram



Facade Pattern



Data Access



Explanation

Purpose:

This diagram provides a high-level overview of the layered architecture used in HBnB Evolution.

Key Components:

- **Presentation Layer:** Handles API requests and responses. Provides services to users through RESTful APIs.
- **Business Logic Layer:** Encapsulates application rules and behaviors. The **BusinessFacade** acts as a single entry point to the core logic, promoting encapsulation and simplifying the interaction between layers.
- **Persistence Layer:** Responsible for data access and storage. Repositories abstract away direct interactions with the database.

Design Decisions:

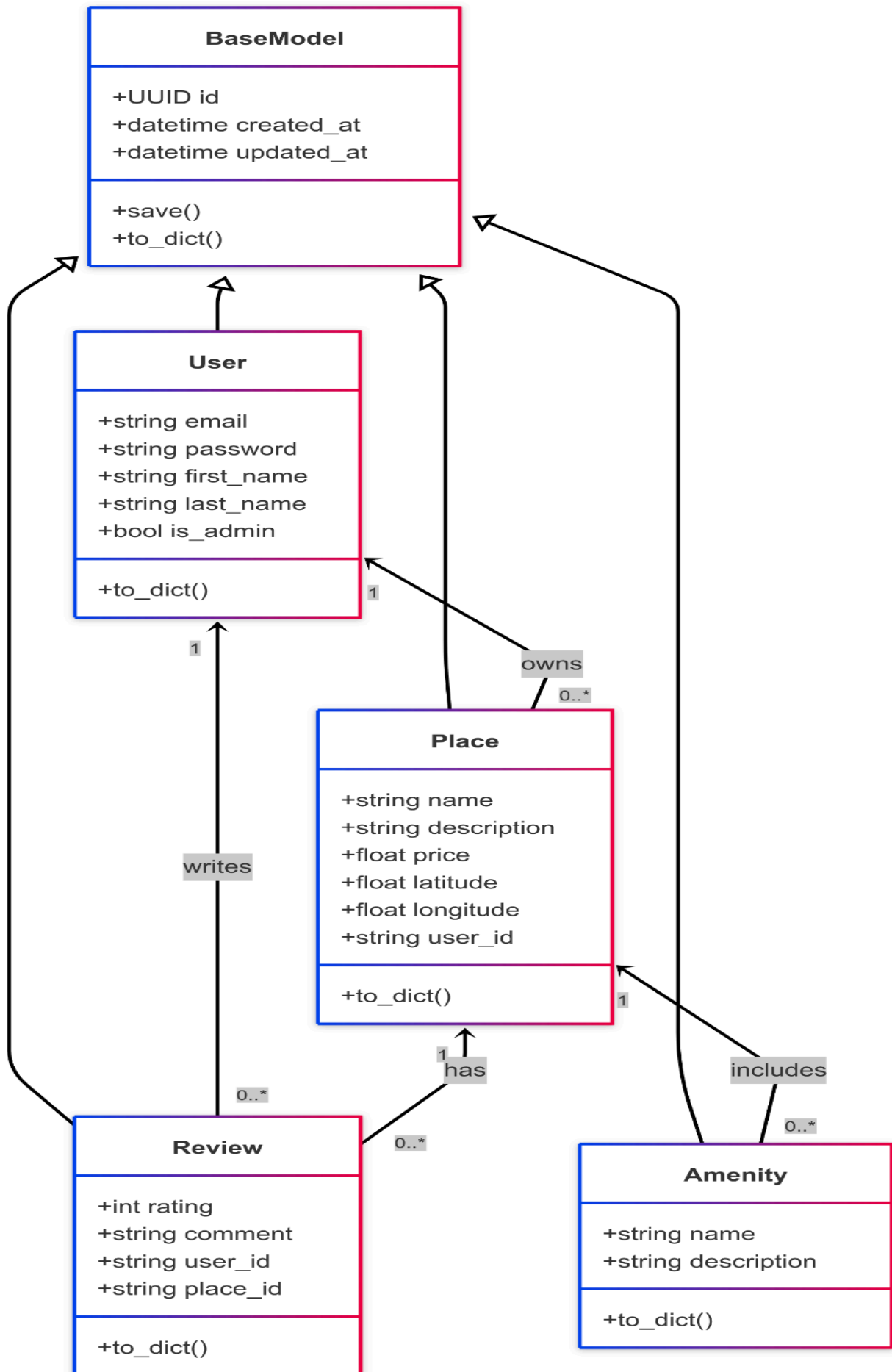
- The **Facade Pattern** is used to ensure the Presentation Layer interacts with a unified interface in the Business Logic Layer.
- A clear separation of concerns is implemented to enhance testability and maintainability.

How it fits into the overall architecture:

This layered approach ensures a modular design, making it easier to implement new features, maintain existing functionality, and scale the system.

Business Logic Layer

Detailed Class Diagram



Explanation

Purpose:

This diagram details the core entities and their relationships in the Business Logic Layer.

Key Components:

- **BaseModel:** Common base class that provides shared attributes and methods.
- **User:** Represents a platform user, with roles including regular user and admin.
- **Place:** Represents a property listed on the platform.
- **Review:** Captures user feedback on places.
- **Amenity:** Represents features associated with a place.

Design Decisions:

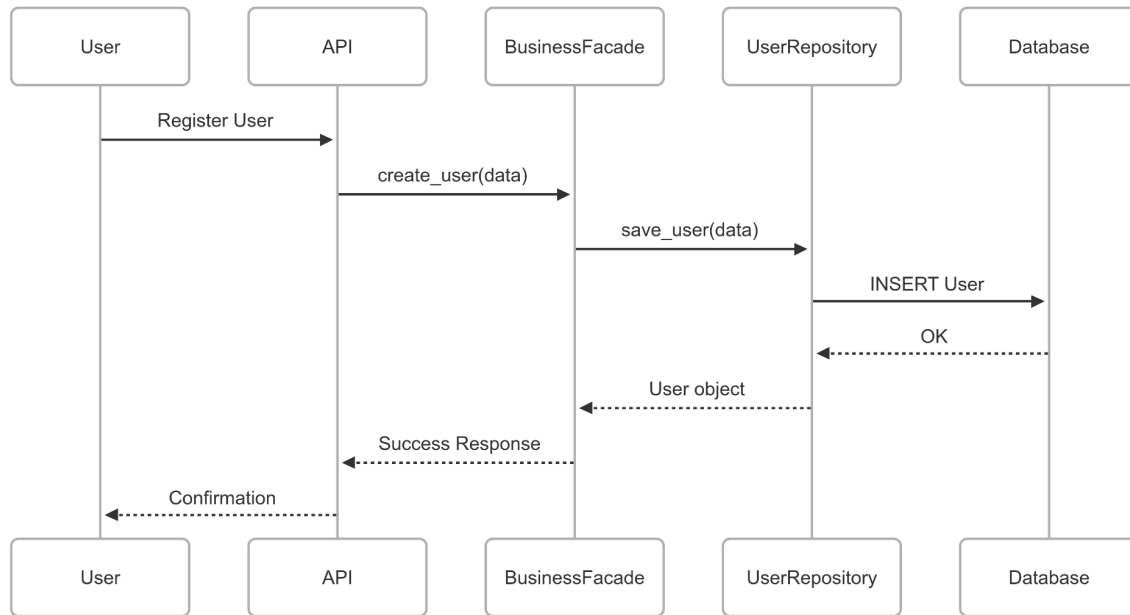
- Inheritance is used via `BaseModel` to ensure DRY (Don't Repeat Yourself) principles.
- Relationships are clearly defined to model real-world interactions (users own places, write reviews, etc.).

How it fits into the overall architecture:

These entities form the core domain model of the application and drive most of the business logic and data interactions.

API Interaction Flow

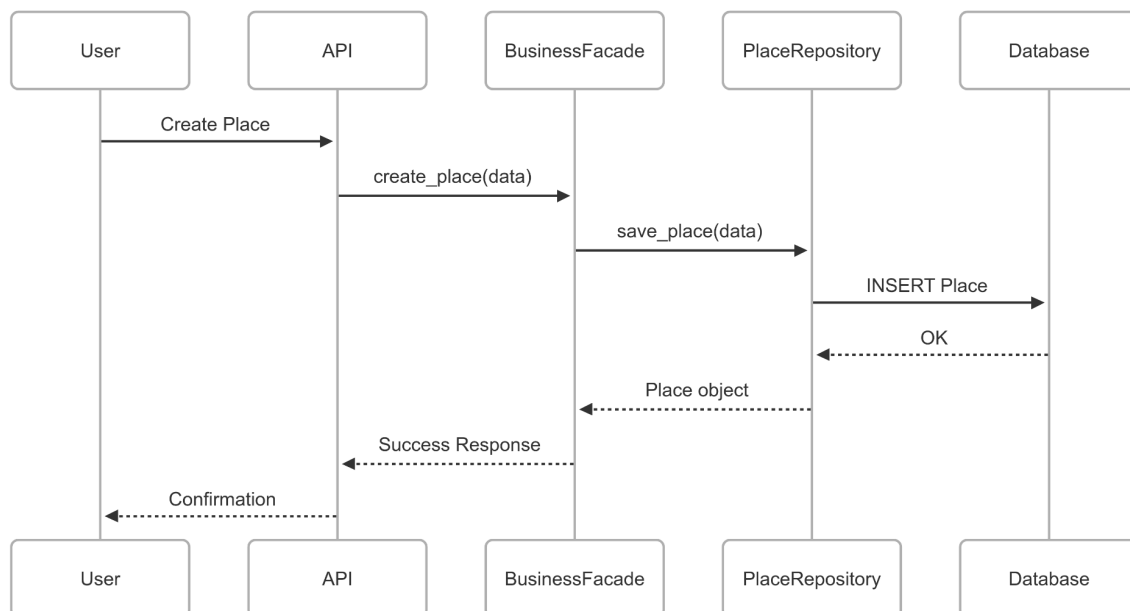
User Registration



Explanation:

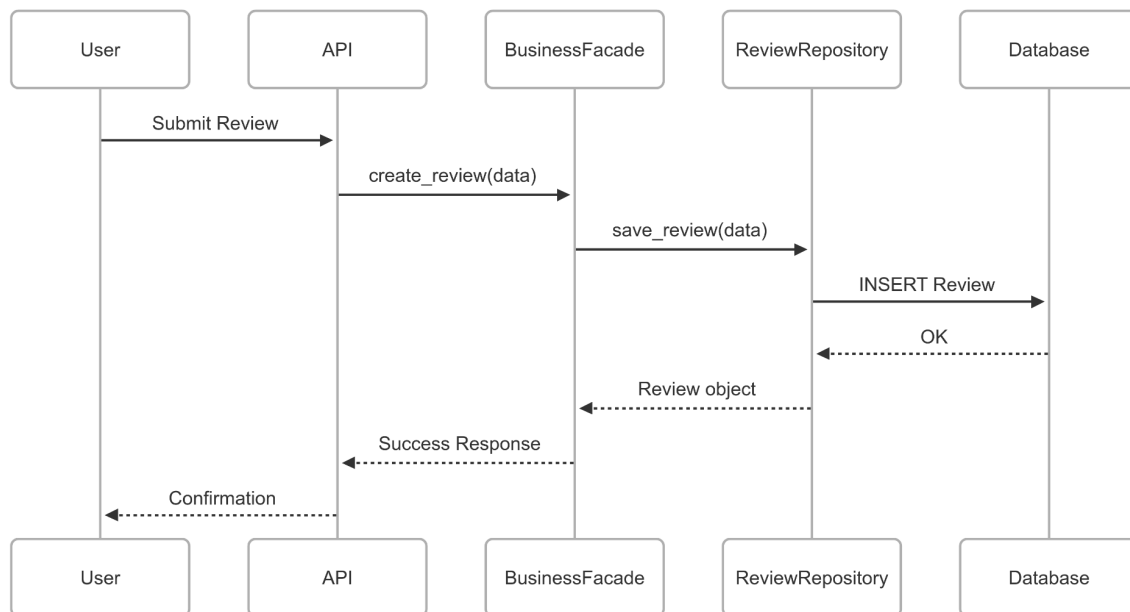
Shows the flow of data during user registration. The BusinessFacade mediates between the API and the persistence layer to enforce business rules and validations.

Place Creation



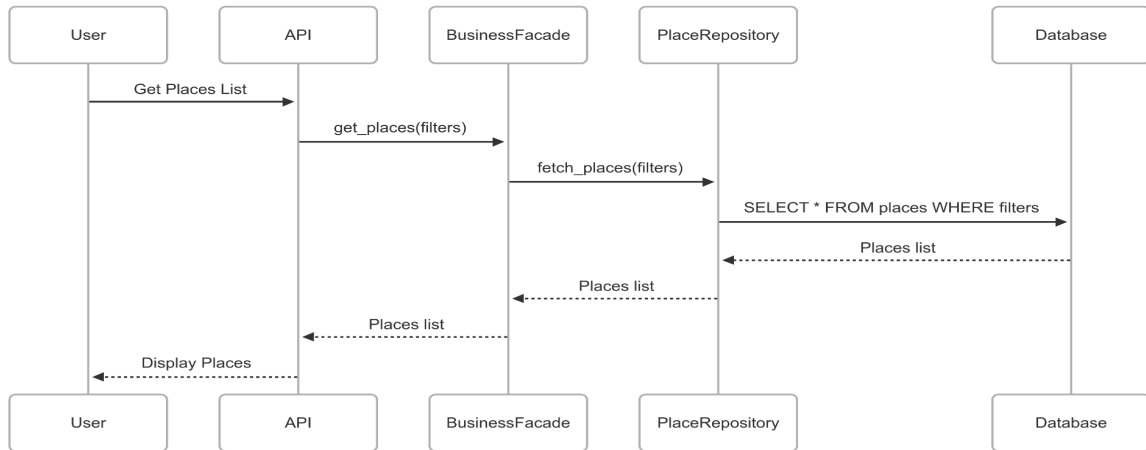
Explanation:

Illustrates how a user can create a new place listing. The PlaceRepository handles data persistence while the BusinessFacade ensures correct application behavior.

Review Submission**Explanation:**

Demonstrates how users submit reviews for places. The ReviewRepository persists the data and the BusinessFacade manages the workflow.

Fetching List of Places



Explanation:

Explains how the system retrieves a list of places based on user-provided filters. The flow ensures efficient and scalable querying.

Conclusion

This technical document provides a cohesive blueprint for the architecture and design of the HBnB Evolution project.

By documenting the layered architecture, core business models, and key API interactions, this document will serve as a reliable reference for all development phases.

Following the structured design outlined here ensures the project will be scalable, maintainable, and ready for future growth.

Final Notes

- **Clarity:** All diagrams and explanations are designed to be easily understood.
- **Consistency:** Naming conventions and formatting are consistent throughout the document.

- **Accuracy:** Diagrams and interactions are accurate reflections of the planned system design.
- **Professionalism:** The document is suitable for internal team reference as well as potential stakeholder presentations.