**HBnB – UML Project: Technical Document**

**Introduction**

This document provides comprehensive technical documentation for the HBnB Evolution application, a simplified version of an AirBnB-like platform. The application allows users to manage properties, leave reviews, and associate amenities with places. The documentation outlines the architecture, business logic, and interaction flows of the system, serving as a guide for the implementation phase.

This document includes:

1. A high-level package diagram illustrating the three-layer architecture of the application.
2. A detailed class diagram for the Business Logic Layer, showcasing its entities, relationships, and methods.
3. Sequence diagrams for critical API calls, visualizing the flow of information across layers.
4. Explanatory notes accompanying each diagram to clarify design decisions and interactions.

**1. High-Level Architecture**

**High-Level Package Diagram**

The application follows a three-layer architecture:

1. **Presentation Layer**: Handles user interactions via APIs and services.
2. **Business Logic Layer**: Contains core models and business logic.
3. **Persistence Layer**: Manages data storage and retrieval.

The **Facade Pattern** is used to simplify interactions between layers, providing a unified interface for communication.

Une image contenant texte, capture d’écran, diagramme, ligne

Le contenu généré par l’IA peut être incorrect.

**2. Business Logic Layer**

The Business Logic Layer consists of four main entities: User, Place, Review, and Amenity. Each entity has attributes, methods, and Relationship as defined below

**Une image contenant texte, capture d’écran, diagramme, Parallèle

Le contenu généré par l’IA peut être incorrect.**

**Explanatory**

* **User**: Represents application users, including administrators. Key methods include registration, profile updates, and deletion.
* **Place**: Represents properties listed by users. Each place is associated with an owner (User) and can have multiple amenities.
* **Review**: Represents user reviews for places. Each review is associated with a user and a place.
* **Amenity**: Represents features or services associated with places (e.g., Wi-Fi, pool).
* **Relationships**:
  + A user can own multiple places.
  + A place can have multiple reviews and amenities.
  + Reviews and amenities are associated with specific places.

1. **API Interaction Flow:**

**Sequence Diagrams**

Below are sequence diagrams for four key API calls:

1-**User Registration**

Une image contenant texte, capture d’écran, diagramme

Le contenu généré par l’IA peut être incorrect.

**2 -Place Creation**

Une image contenant texte, capture d’écran, diagramme

Le contenu généré par l’IA peut être incorrect.

**3- Review Submission**

**Une image contenant texte, capture d’écran, diagramme

Le contenu généré par l’IA peut être incorrect.**

**4-Fetching a List of Places**

**Une image contenant texte, capture d’écran, diagramme

Le contenu généré par l’IA peut être incorrect. Explications :**

* **User Registration**: Validates user data, saves it to the database, and returns a successful message.
* **Place Creation**: Validates place data, associates it with the owner, and saves it to the database.
* **Review Submission**: Validates review data, associates it with a user and place, and saves it to the database.
* **Fetching a List of Places**: Retrieves a list of places from the database based on query parameters.

**Conclusion**:

The diagrams and explanatory notes are compiled into this document and provide a clear and detailed blueprint for the HBnB Evolution application. This documentation ensures a solid understanding of the system’s architecture, business logic, and interaction flows, effectively guiding the implementation phase.