

PetSitter Connect

Created By:

Created For: mr. Mark Peterson

Version: V2.0

Created on: 2025.10.02.

PetSitter Connect

1. Introduction

PetSitter Connect is a web-based platform designed to connect pet owners with trustworthy local pet sitters. The purpose of this document is to define all functional and non-functional requirements, user stories, UI/UX guidelines, and technical aspects needed to develop the MVP version of the application.

2. Scope

The MVP release focuses on core features: user registration, search, sitter profiles, booking, messaging, reviews, and admin management. Future expansions include payment processing, mobile apps, and premium listings.

3. Objectives

- Simplify finding and hiring trusted pet sitters.
- Provide transparency through verified profiles and ratings.
- Enable secure communication and streamlined booking.
- Deliver a scalable MVP ready for further evolution.

4. User Roles & Permissions

Role Description	Permissions
Pet Owner	Books pet sitters. Search, message, book, review.
Pet Sitter	Provides sitting services. Create profile, manage bookings, respond to reviews
Admin	Oversees all data. Approve sitters, moderate content, manage users.

5. User Stories

ID	As a... I want to... So that...
US-01	Pet Owner search pet sitters by city I can find local help.
US-02	Pet Owner view sitter profiles with reviews I can choose the right person.
US-03	Pet Owner book a sitter for dates I can arrange pet care.
US-04	Pet Sitter manage profile and availability I can get new clients.
US-05	Pet Sitter accept or decline bookings I can control my schedule.
US-06	Admin approve new sitters I can maintain platform quality.
US-07	Pet Owner rate and review a sitter after booking I can help others decide.
US-08	Pet Sitter view feedback and overall rating I can improve my service.

6. Functional Requirements

Each functional requirement below is described in full detail to align with professional specification standards.

6.1 User Registration & Login

Purpose: Allow users to securely create and manage accounts, providing access control for both roles.

Description:

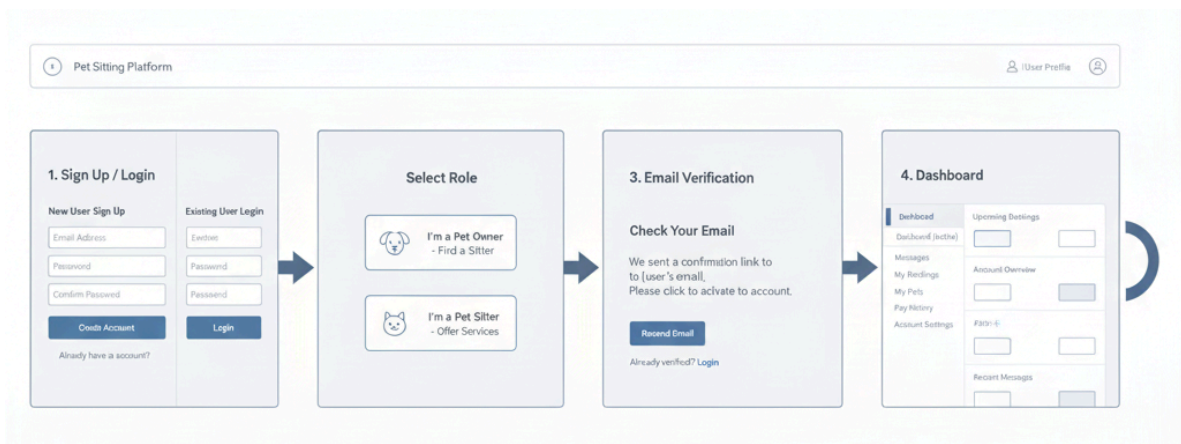
The registration flow supports both standard email sign-up and Google OAuth. The system validates all input data (unique email, password complexity, and role selection). Upon successful registration, the user receives a verification link. After verification, the system creates a user profile and assigns permissions based on the chosen role. Users can update basic info (name, photo, contact details) through the profile editor. Two-factor authentication may be introduced in later stages.

Inputs: Email, password, role, name, and optional photo.

Outputs: Verification email, activation confirmation, and access token.

Preconditions: Internet access, valid credentials.

Postconditions: User is logged in and redirected to the role-specific dashboard. **Exceptions:** Duplicate email, failed OAuth callback, invalid password format, unverified email.



6.2 Search & Filtering

Purpose: Enable pet owners to locate suitable sitters easily and efficiently using advanced filters and real-time results.

Description:

The Search & Filtering module provides a flexible interface for users to find sitters by city, pet type, rating, price, and availability. The search bar uses predictive typing to suggest popular locations. Filters are displayed in a collapsible panel and can be combined dynamically, updating results instantly through asynchronous API calls. Results are presented as cards with sitter name, image, price, and average rating. When no results are found, the system displays suggestions such as expanding location radius or adjusting filters. Pagination or infinite scroll ensures efficient loading of results.

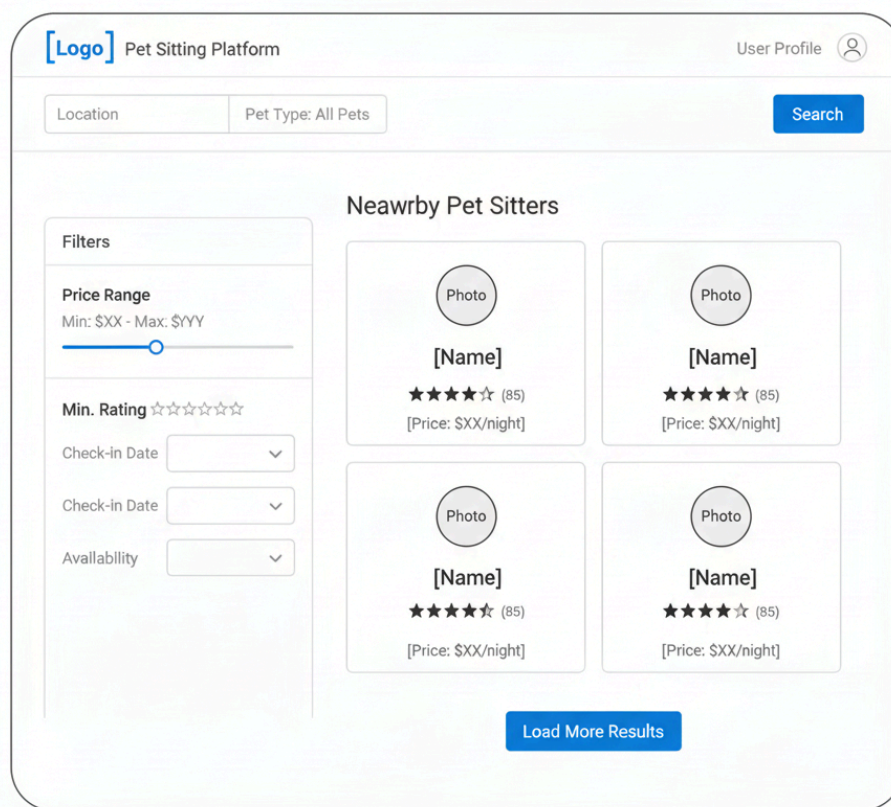
Inputs: City, pet type, availability date range, price range, rating threshold.

Outputs: List of sitter cards with summary data.

Preconditions: At least one sitter with a verified profile exists in the database.

Postconditions: Users can view sitter results and open individual profiles.

Exceptions: Invalid city name, no internet connection, or backend timeout errors.



6.3 Sitter Profiles

Purpose: Present complete and reliable sitter information to enable informed decisions.

Description:

The profile module stores and displays detailed sitter information such as name, profile picture, hourly rate, bio, and service description. Sitters can include multiple pet types (dog, cat, bird, etc.) and add availability through a simple calendar UI. Owners can view the sitter's history, number of bookings completed, and overall rating. The profile page integrates user feedback from the Reviews module and includes a "Message" and "Book Now" call-to-action. Each change made by the sitter (e.g., rate, description) triggers validation and is instantly reflected on their public profile.

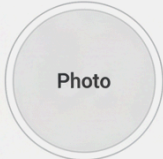
Inputs: Sitter ID, data fields for profile creation and update, uploaded images.

Outputs: Publicly viewable sitter page with validated and formatted information. **Preconditions:** Sitter must be registered, authenticated, and approved by Admin. **Postconditions:** Updated data persists in the database and appears on all related pages. **Exceptions:** Failed image upload, invalid field entry, or unauthorized profile modification.

[Logo]
Pet Sitting Platform

[User Profile]


Sitter Profile



★★★★☆
(85 reviews)

Photo

[Sitter Name]

Hourly Rate: \$XX/night

Short Bio: Lorem ipsum dolor sit amet, consectetur adipiscing elit...

Message

Book Now

Availability Calendar

Mo	Tu	We	Th	Fr	Sa	Su
					1	2
31	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Services Offered

Photo Gallery






6.4 Booking System

Purpose: Facilitate smooth scheduling, management, and confirmation of bookings between owners and sitters.

Description:

The booking system governs all interactions from request initiation to completion. Owners submit booking requests by selecting a sitter, defining start and end dates, pet type, and any additional instructions. The system prevents overlapping bookings by checking sitter availability in real time. Sitters receive notifications (email and in-app) and can accept or reject the request. Accepted bookings move to the “Active” state, while declined ones trigger automated cancellation notifications. Once a booking concludes, its status changes to “Completed,” enabling reviews. Cancellations within 24 hours prompt warnings and may influence sitter ratings. Future versions will integrate payment processing.

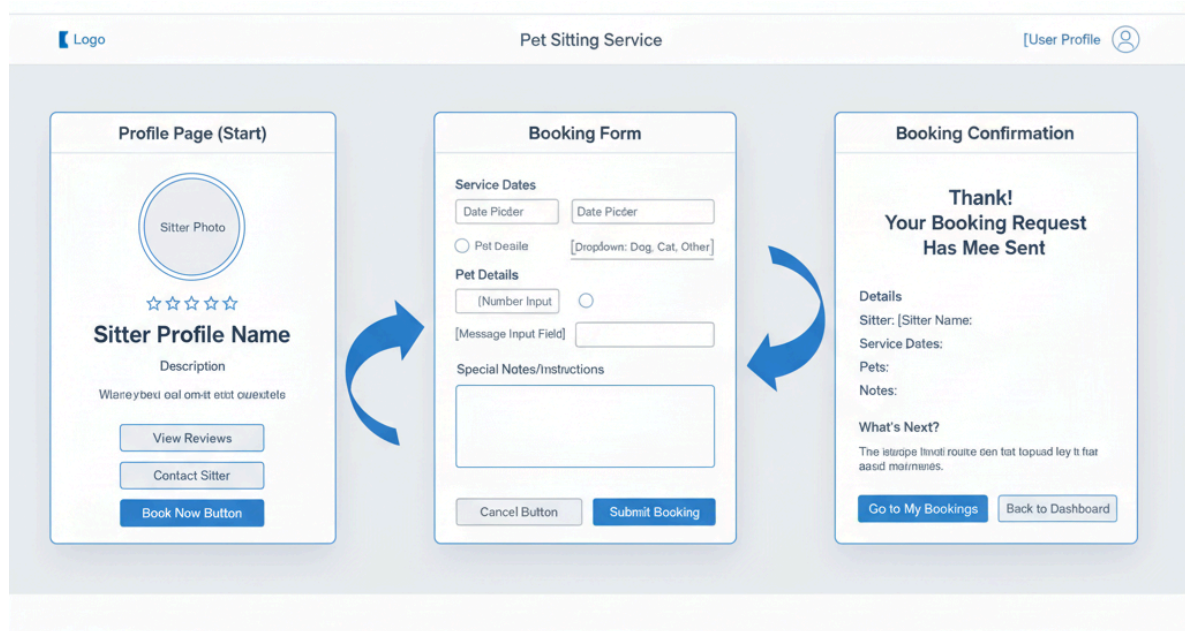
Inputs: Booking request (sitter ID, owner ID, dates, details).

Outputs: Confirmation message, updated calendar, notification alerts.

Preconditions: Both users verified and logged in; sitter profile active.

Postconditions: Booking saved with correct status in database.

Exceptions: Double booking detected, invalid date range, sitter unavailability.



6.5 Messaging System

Purpose: Provide real-time communication while ensuring privacy and reliability.

Description:

The chat interface is designed for intuitive one-on-one conversations between owners and sitters. Built using Supabase's real-time channels, it ensures minimal latency. Each conversation is bound to a booking or initiated through a sitter profile. New messages appear instantly and trigger visual and audible notifications. Older messages remain accessible for history reference. The system filters spam and restricts external links for safety. A typing indicator and "message seen" status enhance UX. Archived chats can be viewed in the inbox tab. Messages are stored securely with timestamping and user identifiers.

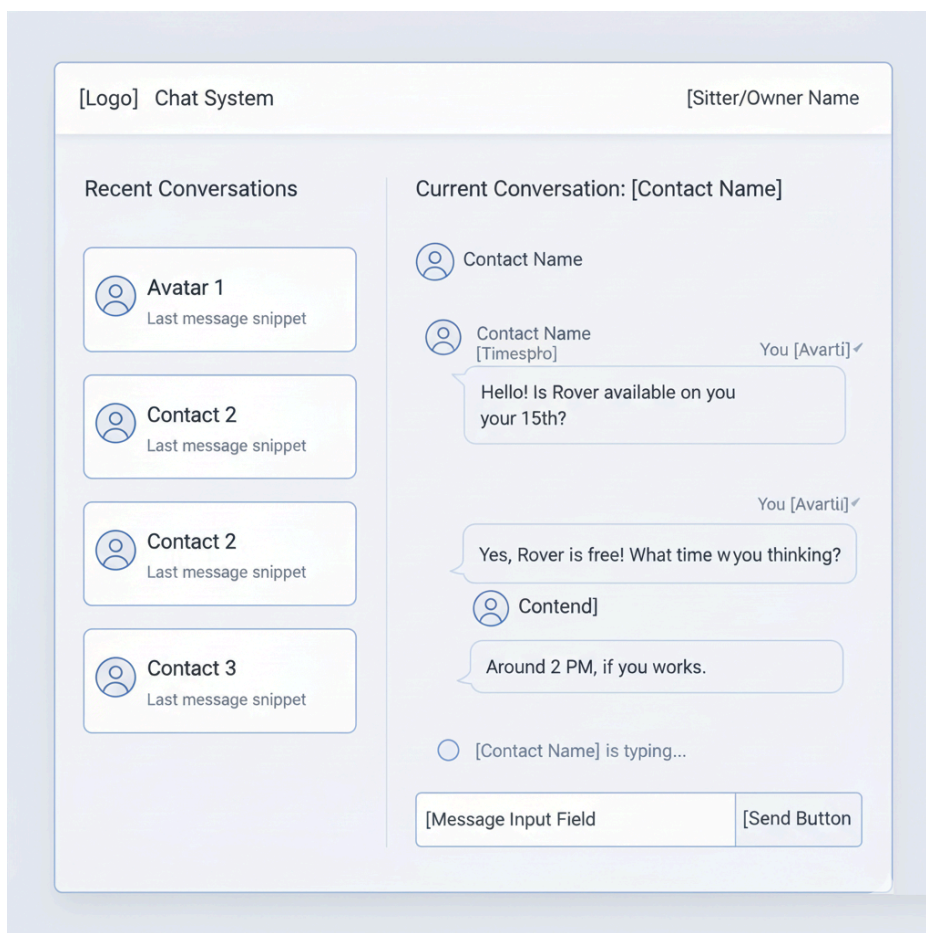
Inputs: Sender ID, receiver ID, booking ID (if applicable), message content.

Outputs: Message displayed in chat UI, confirmation in database.

Preconditions: Valid user session and established contact.

Postconditions: Message synchronized across both clients.

Exceptions: Network disconnection, duplicate message submission, expired session.



6.6 Ratings & Reviews

Purpose: Build trust by enabling transparent feedback and performance tracking.

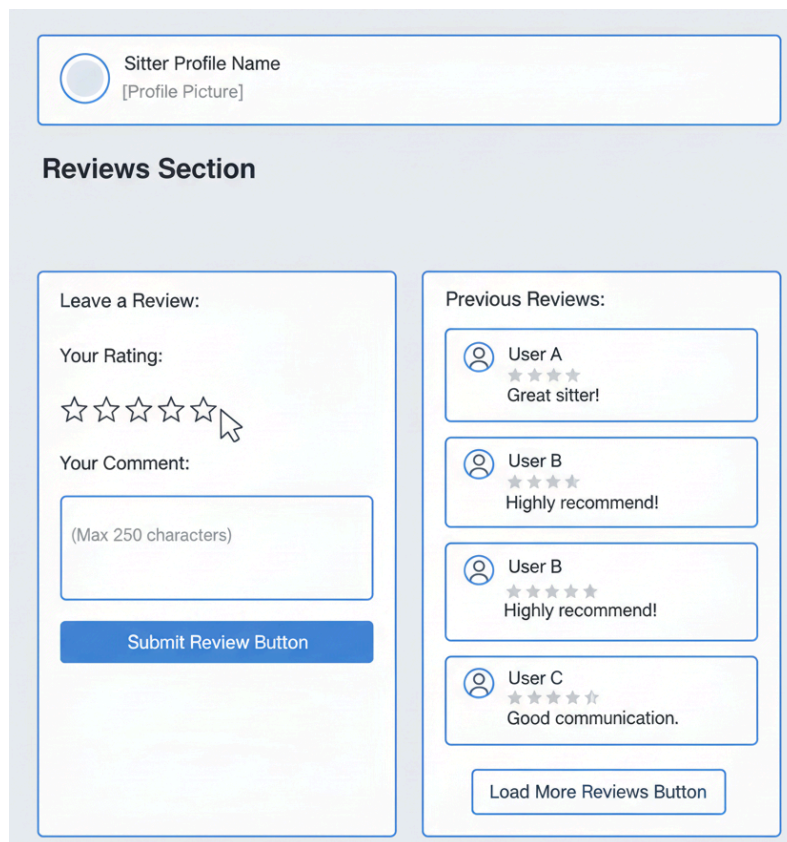
Description: After each completed booking, owners can submit reviews with a star rating (1–5) and an optional comment. The system employs text moderation to detect inappropriate language. Ratings are aggregated per sitter and updated automatically. The sitter profile displays overall average rating and recent reviews. To prevent abuse, each booking can generate only one review. Owners may edit their review within 24 hours. Sitters can publicly reply to reviews to clarify feedback. Future releases will introduce sentiment analysis and review verification for authenticity.

Inputs: Booking ID, rating value, comment text.

Outputs: Confirmation screen, updated sitter rating, stored review record.

Preconditions: Booking marked as completed; both users verified.

Postconditions: Review becomes visible on sitter profile and affects average rating. **Exceptions:** Duplicate reviews, profanity detected, unauthorized submission.



The UI mockup shows a 'Reviews Section' for a sitter profile. At the top, there is a header area with a circular profile picture placeholder and the text 'Sitter Profile Name' and '[Profile Picture]'. Below this is the 'Reviews Section' title. The section is divided into two main columns. The left column, titled 'Leave a Review:', contains a 'Your Rating:' section with five star icons (the fifth is highlighted with a mouse cursor), a 'Your Comment:' section with a text input field labeled '(Max 250 characters)', and a blue 'Submit Review Button'. The right column, titled 'Previous Reviews:', displays a list of three reviews. Each review shows a user icon, the user's name, a star rating, and the review text. The reviews are: User A (4 stars, 'Great sitter!'), User B (5 stars, 'Highly recommend!'), and User B (5 stars, 'Highly recommend!'). Below the list is a 'Load More Reviews Button'.

Sitter Profile	
	Sitter Profile Name [Profile Picture]

Reviews Section

Leave a Review:

Your Rating:

☆☆☆☆☆

Your Comment:

(Max 250 characters)

Submit Review Button

Previous Reviews:

User A

★★★★☆

Great sitter!

User B

★★★★★

Highly recommend!

User B

★★★★★

Highly recommend!

User C

★★★★☆

Good communication.

Load More Reviews Button

6.7 Admin Dashboard

Purpose: Equip administrators with control tools to ensure quality, safety, and performance.

Description:

The Admin Dashboard provides a global overview of users, sitters, bookings, and reviews. Admins can approve new sitter applications, verify identity documents, and deactivate problematic accounts. The dashboard displays real-time statistics—active users, monthly bookings, and satisfaction ratings—through charts and tables. Admins can filter data by date or user role and export logs for audits. The interface supports quick actions such as “Approve,” “Suspend,” or “Delete.” All changes are logged for accountability. Alerts highlight unusual activity such as spam reviews or multiple booking cancellations.

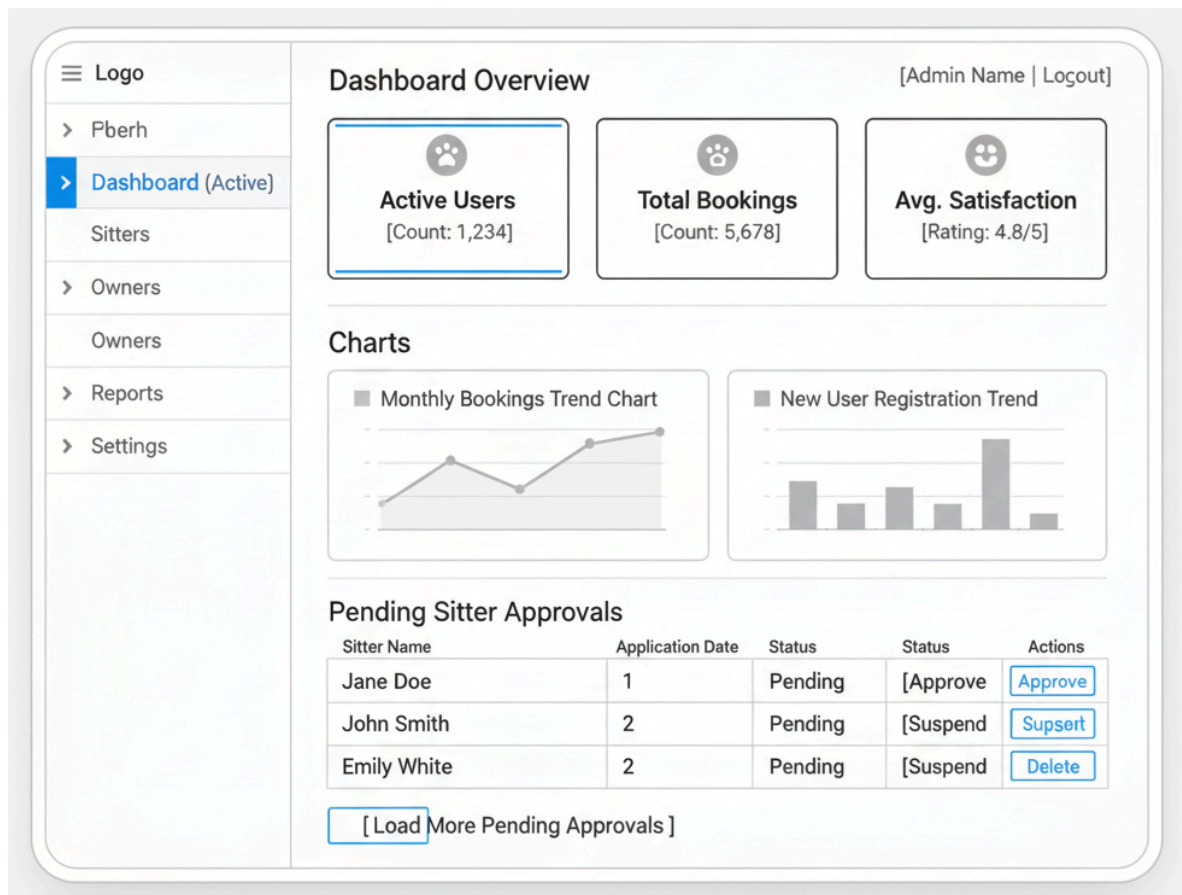
Inputs: Admin login credentials, filters (date range, role, status).

Outputs: Data dashboards, moderation actions, activity logs.

Preconditions: Valid admin account with appropriate privileges.

Postconditions: Database updated according to admin operations.

Exceptions: Unauthorized access attempts, failed queries, or timeout during data load.



7. Non-Functional Requirements

- Performance: Pages load in <2 seconds
- Scalability: Support 10,000 concurrent users
- Security: SSL, password hashing, input validation.
- Availability: 99% uptime.
- Accessibility: WCAG 2.1 AA compliance.

8. Design & UX Guidelines

- Minimalist, mobile-first interface using soft greens.
- Card-based layout for sitter listings.
- Sticky navigation and floating action buttons on mobile.
- Clear typography, consistent iconography.

9. UI Wireframes Overview

Screens: Home/Search, Profile, Booking Form, Inbox, Admin Panel.
Each screen includes header, content, and footer blocks with modular components.

10. System Architecture Overview

- Frontend: React/Next.js + Tailwind CSS.
- Backend: Supabase (PostgreSQL + Auth).
- Hosting: Vercel or Cloudflare Pages.
- Integrations: Email notifications via SendGrid.

11. API Endpoints (Examples)

Endpoint	Method Description
/api/sitters	GET Retrieve sitter list.
/api/sitter/:id	GET Fetch profile data.
/api/bookings	POST Create booking request.
/api/messages	POST Send chat message.
/api/reviews	POST Submit review.

12. Data Flow & Entities

Main entities: User, Profile, Booking, Message, Review.

Typical flow: Owner → Search → Profile → Booking → Messaging → Review.

13. Terminology & Definitions

Term	Definition
MVP (Minimum Viable Product)	The first version of the product that contains essential features to validate the concept and gather user feedback.
Pet Owner	A user role representing individuals looking for pet-sitting services.
Pet Sitter	A user role representing service providers who offer to take care of pets.
Admin	The system administrator responsible for user approval, moderation, and maintenance.
Booking	A confirmed agreement between a pet owner and a sitter specifying service dates and details.
Profile	A user's personal page that includes information, experience, and reviews.
Review	A feedback entry containing a star rating and comment left by a pet owner after a completed booking.
Supabase	A backend-as-a-service platform providing database, authentication, and API functionalities used for this project.
Responsive design	The design approach ensuring the web interface automatically adjusts to different screen sizes and devices.

Authentication Token	A unique, secure identifier used to verify a user's session after login.
API Endpoint	A specific path through which the frontend communicates with the backend to exchange data.
WCAG	Web Content Accessibility Guidelines – standards ensuring digital accessibility for all users.
Prototype	A visual, clickable representation of the application used for validation before development.
Business Analysis (BA)	The initial phase where user needs are analyzed and transformed into structured requirements.