System Requirement Specifications (SRS)
By Pawfection

1 Table of Contents

2 Problem Statement

Pet owners in Singapore face challenges when searching for and booking pet care services such as grooming, boarding, walking, training, and daycare. Currently, they rely on fragmented sources such as social media, search engines, and word-of-mouth referrals. This process is inefficient, time-consuming, and lacks transparency, making it difficult to compare services and service providers by price, availability, or quality.

On the other hand, service providers, especially independent small businesses, also face challenges in reaching potential customers beyond their immediate networks. They often rely on social media, word-of-mouth referrals and third-party listings to expand their outreach. They also often manage bookings manually, which reduces efficiency and potentially creates scheduling conflicts.

There is a clear need for a centralized, web-based platform that enables pet owners to easily discover, compare, and book trusted pet care services. This is while allowing pet service providers to efficiently showcase their services and manage bookings.

3 Overview

3.1 Background

The growth of pet ownership in Singapore has increased the demand for reliable, affordable, and accessible pet services. However, existing solutions are either too fragmented, not tailored to pet-specific needs or lack the required flexibility. Pet owners must navigate multiple channels to book services, leading to wasted time and increased frustrations. Service providers lack a professional platform to promote their services, verify their credibility, and manage bookings in real-time.

As the demand grows, both owners and service providers require a scalable solution that bridges this existing gap. PawfectMatch aims to be this centralized platform.

3.2 Overall Description

PawfectMatch is a web-based application that serves as a centralized platform for connecting pet owners with pet service providers. It will provide a user-friendly interface for both pet owners and service providers while simultaneously managing business logic, bookings, and data storage.

The system will support:

- Pet Owners: Account creation, adding pets, searching services, making bookings, and leaving reviews
- Service Providers: Account creation, adding pet services, managing availability, handling bookings, and receiving reviews

PawfectMatch will ensure convenience, trust, and transparency by consolidating services into one single interface.

4 Investigation & Analysis Methodology

4.1 System Investigation

The PawfectMatch system first processes user interactions by matching login and registration requests to stored user account records in the database. The user's search requests are processed by querying stored caretaker profiles against search filters such as location, availability, service type. Confirmed booking transactions are stored in booking records that are created by the system for each request, after which corresponding payment instructions are sent to the integrated payment gateway for authorization and settlement. A payment status indicator (success or failure) is then transmitted back to the application, which updates the booking record accordingly. Subsequently, an appropriate confirmation or error notification is delivered to the user through the web interface, ensuring transparency and reliability in the transaction process.

4.2 Analysis Methodology

4.2.1 Feasibility Study & Requirements Elicitation

Conduct structured interviews, surveys, and focus groups with pet owners and service providers. Shadowing sessions will observe how services are currently discovered, booked, and managed. Brainstorming workshops will help capture explicit and implicit needs. A Feasibility and Risk Assessment study will be conducted to determine which solution(s) are most appropriate based upon the results of the interviews.

4.2.2 System Analysis & Requirements Specification

4.2.2.1 Perform an analysis of the problem using object-oriented techniques

An external view of the enterprise model will be developed via Unified Modeling Language (UML). This System Requirement Specification documents will form part of the documentation for the project.Some desired features of the new system include:

1. Allow users to registration and login
2. Allow users to create and edit profiles
3. Allow users to search and discover caretakers
4. Allow users to view profiles to evaluate service suitability
5. Allow users to book available services and do payment
6. Allow users view, modify or cancel booking.

4.2.2.2 Scope and Limitations

Analysis methodology will involve business analysis, requirement analysis, data analysis, process analysis, and application architecture:

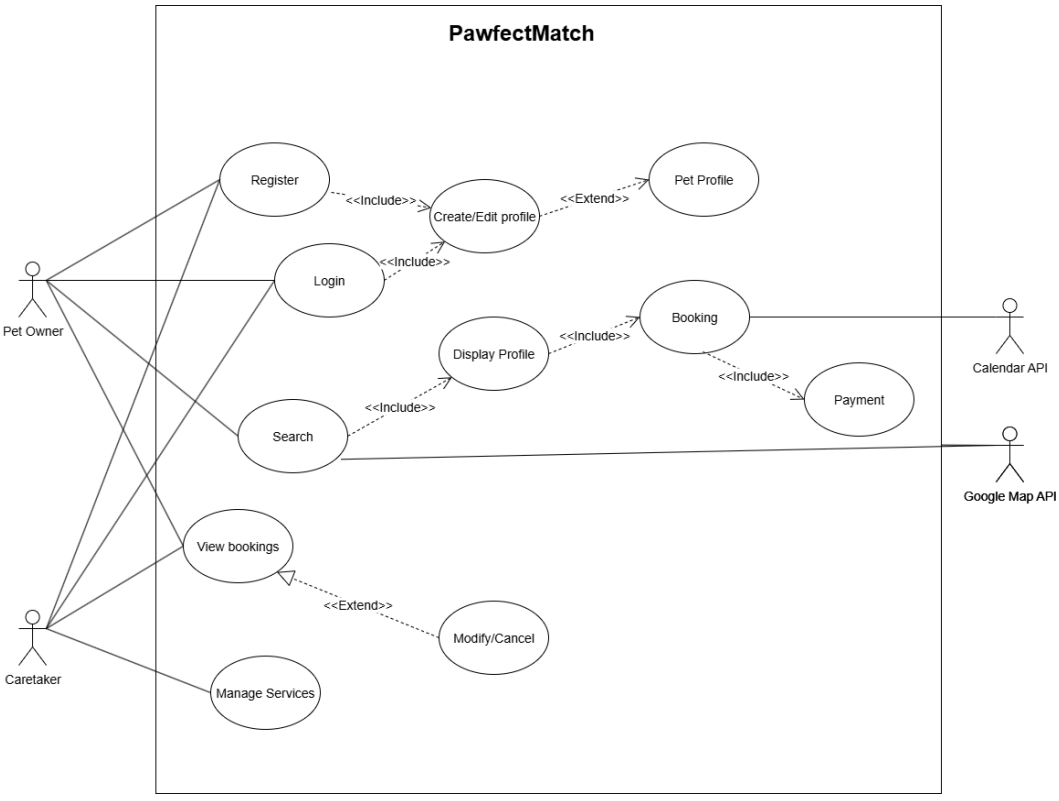
1. Business analysis – State the business rules, business system interfaces, business function, business ownership, sponsorship and associated project budget requirement.
2. Requirement analysis – System I/O description, user requirement definition, functional and security requirement
3. Data analysis Involve data collection process, data validation, data storage, manipulation and retrieval
4. Process analysis – Data/process flow analysis, process decomposition and system interfaces
5. Application architecture – Analyze application information structure, usability, user interface design, interaction and application implementation.

4.2.3 Object-oriented Design using UML

A detailed object-oriented design for the PawfectMatch will be developed. UML will be used again for the graphical representation and documentation of the design.

4.2.3.1 Use Case 1

The use case model provided identifies key actors and the interactions they have with the system's features. This diagram helps visualise how the system features interact with each other.



4.2.3 Prototyping

Prototyping focused on the progressive development of key functionalities, aligning with the Agile methodology. Instead of static wireframes, the actual application features were incrementally built, tested, and refined to allow stakeholders to validate concepts early and provide continuous feedback. Unit and end-to-end tests were also written to validate the correctness of the prototype.

The final prototype will provide basic, working versions of:

- User Authentication (Login and Registration)
- Profile Management (Pet Owner and Service Provider)
- Service Listings and Search
- Booking System

5 Constraints

5.1 Scalability

PawfectMatch is designed to scale adequately for future growth in both users and functionality. While the initial version is targeted to support up to 150 concurrent users with the usage of connection pooling and efficient query design, the system will be built with scalability in mind. This includes:

- Frontend: React application will be optimized for scalability by using modular components and state management practices that support easy extension and feature addition.
- Backend: FastAPI is chosen for its asynchronous capabilities, which allow it to scale horizontally as traffic increases. The use of Docker and containerization will ensure the system can scale across multiple cloud instances as needed.
- Database: PostgreSQL is scalable, supporting horizontal and vertical scaling techniques. Data partitioning, indexing, and query optimization strategies will be employed as the database grows to handle increased user activity and data complexity.

For higher concurrent loads, mechanisms such as caching and database replication will need to be introduced.

5.2 Proprietary Hardware and Software

PawfectMatch will primarily rely on open-source technologies, ensuring flexibility and cost-efficiency. The backend uses FastAPI, and the frontend uses React with TailwindCSS, which are both open-source and have active community support.

There are no proprietary hardware or software requirements for the platform. However, in the future, the system may integrate with third-party APIs that could require licenses or credentials. These

third-party services will be used as external dependencies to provide specific functionalities such as online payments and authentication.

5.3 Batch Updates VS (close) Real-time Updates

Updates of the schedule details between the back-end database and front-end will happen real time, since the size of data packets are well within reasonable scale. This is such that the user could view the progress of the booking/schedule whenever needed.

5.4 Project Schedule

The Project will have a timeframe of three months. With major features implemented and documents compiled for prototype demo in lab 4 in the second week of October and the remaining documents by the first week of November.

6 Operational Requirements

6.1 Help Desk Support

PawfectMatch will provide dedicated support to assist users with issues related to the platform, including slow system response times, incompatible browser feature, application errors, system downtime inquiries, account management, etc. The help desk support will be available through the following channels:

- Email Support: Dedicated support email address for user enquiries
- Phone Support: For urgent matters, users will be able to contact a phone support line available during business hours

The team will be trained to address various possible issues that users could encounter, including issues specific to pet owners and service providers.

6.2 Application Services and Technical Support

PawfectMatch will ensure continuous availability and reliability with developers having access to the source code of the system to respond to system failures or critical bugs with response time varying based on the severity of the issue. The network and DBA support is also required to maintain the system 24/7 uptime.

6.3 Administration Features

There are varying levels of system access and functional authority. Each User's access is limited to his/her own registration records. Only authorized system administrator(s) have access to all users registration records.

6.4 Third-Party Independence & Degraded Modes

While PawfectMatch leverages third-party services for authentication and notifications, the platform is designed to be resilient to third-party service failures. The system will still operate core features without external integrations.

Measures include graceful degradation where if notifications are unavailable, users can still see their booking status in the application while queued messages will continue to retry. Fallback systems will also be in place where in the case of authentication failure, users will be able to log in with their credentials directly. PawfectMatch will remain operational and its functionality will be complementary but independent from the online registration system.

6.5 System Hardware Fail Over and Routing Back-up

Daily incremental and weekly full backups, preventive hardware maintenance, fail over, scheduled system patches and maintenance.

6.6 Audit Trail

All critical actions (login, profile changes, service edits, bookings, cancellations, reviews) shall be timestamped with actor ID and payload hash.

7 Functional Requirements

7.1 User Management

7.1.1 Registration

7.1.1.1 The system must allow new users(pet owners or caretakers) to create an account.
7.1.1.2 The system must require mandatory fields: name, email, password.
7.1.1.3 The system must validate that the email is unique before registration is complete.

7.1.2 Login

7.1.2.1 The system must allow registered users to log in with email and password.
7.1.2.2 The system must authenticate credentials against stored data.
7.1.2.3 The system must provide an error message for invalid login details.

7.1.3 Create/Edit User Profile

7.1.3.1 The system must allow users to create and update profile details.

7.1.3.2 For pet owners, profiles include personal info (name and contact number).
7.1.3.3 For caretakers, profiles include name, services and location.

7.2 Pet Management

7.2.1 Create/Edit Pet Profile

7.2.1.1 The system must allow pet owners to create profiles for their pet.
7.2.1.2 The system must store pet details including names, species, breed, age, and health conditions.
7.2.1.3 The system must allow pet owners to update pet information at any time.

7.3 Service Discovery

7.3.1 Search

7.3.1.1 The system must allow pet owners to search for caretakers.
7.3.1.2 The system must display a list of caretakers with relevant details: name, services offered, and location.

7.3.2 Display Profile

7.3.2.1 The system must allow pet owners to view caretaker profiles in detail.
7.3.2.2 The system must allow caretakers to view their own profile, displaying their personal details, list of upcoming bookings, list of services they provide.
7.3.2.3 The system must allow pet owners to view their own profile, including their personal details and any pets they own.

7.4 Booking Management

7.4.1 Booking

7.4.1.1 The system must allow pet owners to request bookings with a caretaker.
7.4.1.2 The booking must include service type, date and time.
7.4.1.3 The system must allow caretakers to accept or decline bookings.

7.4.2 View Bookings

7.4.2.1 The system must allow users (owners and caretakers) to view a list of upcoming and past bookings.
7.4.2.2 Owners must see bookings they requested and their status.
7.4.2.3 Caretaker must see bookings requested by owners.

7.4.3 Modify/Cancel Booking

7.4.3.1 The system must allow owners to modify or cancel bookings.

7.4.3.2 The system must update bookings history for owners and caretakers.

7.5 Payment and Service Management

7.5.1 Payment

7.5.1.1 The system must allow pet owners to initiate payment after booking a service.
7.5.1.2 The system must display a payment form with booking details.
7.5.1.3 The system must display a success message to the user upon successful payment.

7.5.2 Manage Services

7.5.2.1 The System must allow caretakers to define and manage services.
7.5.2.2 The System must allow caretakers to set service rates and availability.

8 Input Requirements

8.1 User Access Credentials

Pet owners supply personal information such as their name, contact details, and region, as well as details about their pets including species, breed, age, and any special requirements. When requesting services, owners provide booking details such as the chosen service, preferred dates and times, and any additional notes. After a service is completed, they may also submit feedback in the form of ratings, written comments, or media attachments.

8.2 Provider Profile Information

Service providers contribute their business information, which may include licensing or verification documents, and create listings that describe the services they offer. These listings specify service type, price, duration, and relevant constraints. Providers also define their availability through calendar entries and operating hours, while actively responding to booking requests by accepting, declining, or proposing alternatives.

8.3 Payment Method

Pet owners and service providers input their preferred payment method to pay for and to be paid. The platform stores the payment token, last four digits, expiry, and billing address. When a booking is confirmed, the system creates a payment intent and places a pre-authorization for the estimated amount. Charges are captured automatically after the service is marked completed, or refunded on cancellation. Service provider payouts are executed through connected accounts with scheduled disbursements and statements.

9 Process Requirements

9.1 PostgreSQL Transaction

The system must be able to initiate, process, and handle transactions in the PostgreSQL database system.

9.2 Data Integrity

Commit successful transactions and rollback failed requests/unfinished transactions.

9.3 Data Validation

Data error from the user's end and from the back-end must be gracefully handled, following a zero-trust policy. Frontend will have validation checks implemented, and backend will validate incoming input again from users.

9.4 Performance

Must handle concurrent use of the system on a 24x7 basis. Send, receive and display user messages to assist the overall user experience.

9.5 Data Repository

PawfectMatch will maintain and use the PostgreSQL database as the single source of truth.

10 Output Requirements

10.1 Transaction Summary and Confirmation

Each time an action is performed the system will include them in the transaction summary. This summary records the details of the service, the provider, the pet owner, and the agreed date and time. Users receive confirmations in real time through the application interface or push notifications. These outputs not only acknowledge that a transaction has been successfully recorded but also serve as an audit trail for resolving disputes and verifying service agreements.

10.2 User/Provider Dashboard

For owners, the dashboard displays upcoming bookings, past service history, and pending reviews that require submission. It also provides alerts for changes in booking status or upcoming appointment reminders. For providers, the dashboard consolidates booking requests awaiting action, scheduled services, and aggregated feedback from clients.

10.3 Administrative Reports and Summaries

The system will generate reports summarizing metrics such as the number of active providers, service categories in demand, booking volumes by region, and overall customer satisfaction scores. Summaries also highlight exceptions, including unresolved disputes or flagged reviews that require moderation.

11 Hardware Requirements

11.1 Network

A stable network connection with a minimum of 1Mbps speed is required for the web-based application to function.

11.2 Client Devices

1 gigahertz (GHz) or faster compatible processor or System on a Chip (SoC)
1GB of RAM / 2GB of RAM for the 64-bit version
500MB of hard drive space

12 Software Requirements

12.1 Client Operating Systems

- Windows 10 or later
- macOS 10.15 or later
- Android 4.1 or later

12.2 Client Application

Modern Javascript ES6 compatible browsers

- Chrome
- Firefox
- Safari

12.3 Network System

Network software and protocols in order for the systems to communicate:

- TCP/IP
- HTTP
- HTTPS

12.4 Licences

Technology	Licence Type
Vite	MIT License
React	MIT License
Typescript	Apache License 2.0
Node.js	MIT License
Python	Python Software Foundation License (PSF License)
FastAPI	MIT License
PostgresSQL	PostgreSQL License (permissive, BSD-style)
SQLAlchemy	MIT License
ESLint + Prettier	MIT License
Docker	Apache License 2.0 (Docker Desktop is proprietary under Docker's subscription terms)

13 Deployment Requirements