1. Introduction

1.1 Purpose

1.1.1 The purpose of the app PetLife is to help a household more efficiently take care of their pets. They will be able to keep track of their pet’s needs, coordinate a feeding schedule with fellow housemates, and get notifications for alerts.

1.2 Scope

The software that will be used for this web app will be Angular JS, Node JS,  Python, and MongoDB.

This software will help households take better care of their pets.

1.3 Definitions, acronyms, and abbreviations

PetLife - Name of the app

1.4 Reference documents

There are no reference documents.

1.5 Overview

The contents of this document is structured with numbered sections.

2. Overall description

1.1 Product perspective

This product, PetLife, will be used by users with pets and is a new application. These users want to keep better track of their pet's needs to keep them healthy and make life easier. This product is not a follow-on member of a product family and is not a replacement for any existing systems.

1.2 Product functions

The product has a household (can only join when invited) with a group owner that allows other users to feed, water, walk, and take medical care of the pets. Each member of the group can do a task for a pet, and when a task is completed the application will notify the group of the task’s completion. The application will also form a schedule based on the household member’s availability where users can select shifts they want to feed/walk the pet during. If there are no members of the household signed up for a given task when the time comes, an alert will be sent out notifying the household. The user will be able to add pets to the household and customize all of their needs including what time certain actions need to be performed. Users can also enter in veterinarian information and set up alerts for upcoming visits. Alerts can also be set up for playtime, feeding time, and more.

1.3 User characteristics

Users have a notification center in the center of the screen after they have logged into the application. Users are prompted to enter their availability in order for a schedule to be formed. There is a schedule section that allows any user in the group to check off tasks for the pets. The user has the option to clear all notifications on their screen. Users are in charge of adding and removing pets as well as entering in pet information.

1.4 Constraints

There are none as of now.

1.5 Assumptions and dependencies

There are no assumptions or dependencies yet.

3. Specific requirements

1.1 External interfaces

1.1.1 User interface

The user interface is GUI-based, allowing the user to interact via icons and indicators. The user interface is then rendered in the internet browser.

1.1.2 Hardware interface

The hardware interface will reside on the computer or electronic device with an online browser.

1.1.3 Software interface

The software interface is an internet browser that is connected to our web application.

1.1.4 Communication interface

The Web app will work on web browsers (PC and mobile) and communicate with the database service (MongoDB).

1.2 Functional requirements

* First-time users will be prompted to create an account or log in to an existing account.
* Once logged in, the system will check if the user is a member of a household.
* If the user does not belong to a household, they will be prompted to create a “household”, view invites to households, or request to join an existing household.
* On the “Create a household” page, the user must provide a name for the household and include any pets they have. The user will be prompted to invite other users to be members of the household.
* Members of a household will automatically become “owners” of that household’s pets.
* Members of a household will be able to add/remove pets from the household.
* Members of a household will be able to edit pets’ profiles of their household. Editing includes the pet’s name, needs (food, water, walking, play-with, shower, heat lamps, clean tank, etc.), frequency of needs, age.
* When an owner marks a pet’s need as “fulfilled”, all owners of that pet will have access to the update.
* Each household will have one “primary owner”. The primary owner can only be changed by the existing primary owner.
* Each user is prompted to enter their schedule for walking the pets, feeding, and any other needs. Users can then sign up for shifts for when they want to feed/walk/etc.
* The schedule will fill up with the different users and who is doing what when and allow other users to view other users’ schedules.
* If a shift is empty when the time comes, a push notification will be sent to all users alerting them of the need.
* Push notifications will be sent out after a designated time has passed without any users marking the animal as fed after their designated feeding time.
* Push notifications will be sent out after designated amounts of time to remind users of visits to the vet, applying medication, or anything else.
* Users will have the opportunity to opt-out or into certain types of push notifications
* A button to change the settings to the app

1.3 Performance requirements

* The system will support up to 10 users logged in to a household at a time.
* The system will not have a limit on how many pets a household can have.

1.4 Design constraints

* The system will only work on web browsers.

1.5 Other requirements

* None as of now

4. Appendices

5. Index