Pet Matcher

STAKEHOLDERS

UNCG Honor Code

We have abided by the UNCG Academic Integrity Policy

TEAM MEMBERS

David Bowles, Yngrid Corrales Gonzales, Serena Wisnewski Herter

TEAM NAME

Passive-Aggressive Post-It Notes

DATE

11/30/2020

Table of Contents

1. Introduction

1.1 Purpose

Our app is used to help pet lovers find a new family member, and we wanted to make it easier and more interactive when finding a new pet. We generalized our idea from a popular dating app called Tinder. So basically if the user sees a pet they would like to adopt, they would select yes, and if they aren't particularly interested in adopting that certain pet then they would select no. This way it saves time and effort for the user when searching for a new pet and it conveniently saves their selected yes choices in a favorites page.

1.2 Document Conventions

This document is structured as follows

- Section 1 contains this introduction
- Section 2 contains the description of our application, included it's features, characteristics, and constraints
- Section 3 contains functional requirements
- Section 4 contains technical requirements, including compatibility and interface requirements
- Section 5 contains additional nonfunctional requirements including performance, safety, and security information

1.3 Intended Audience

The intended audience is Ike Quigley. Besides Ike Quigley, the app would be used by people searching for a new pet available for adoption in their local area.

1.4 Definitions/Jargon

Tinder: Tinder is an American geosocial networking and online dating application that allows users to anonymously swipe to like or dislike other profiles based on their photos, a small bio, and common interests. We use a system like Tinder to allow users to swipe through pets until they find a "match" for themselves.

1.5 Project Scope

Our project scope is having the Alpha version of our project working and complete.

1.6 Technical Challenges

We have never made an API before, used Swing, and created a database from scratch, so these are the technical challenges that we will overcome to get this project made.

1.7 References

Not applicable.

2. Overall Description

2.1 Product Features

The product features for Pet Matcher is a login, sign up, a select yes or no button for a certain animal you would like to adopt, and to be able to see your favorites page, the page where it displays all the animals you selected yes on.

2.2 User Characteristics

Users who are looking to be new pet owners and users who are experienced, pet owners. It's open for every person, there are no restrictions.

2.3 Operating Environment

The operating environment would be people who have internet access on their phones.

2.4 Design and Implementation Constraints

Being mobile-friendly would be a design and implementation constraint.

2.5 Assumptions and Dependencies

Pet organizations that were used for the API stay the same, they don’t change into any other type of organization but those involving pets.

3. Functional Requirements

3.1 Primary

Two of the primary requirements are being able to select the Yes or No button for the displayed pet, and have the contact information displayed on the favorites page. Another primary function is to be able to display the pets which the user chose “yes” for on a separate page.

3.2 Secondary

User login and signup, and update preferences are the secondary requirements.

4. Technical Requirements

4.1 Operating Systems/Compatibility

Any operating system that has a Java virtual machine (JVM) already installed.

4.2 Interface Requirements

4.2.1 User Interface

The touch screen on the user’s phone is a form of interaction between the user and the application.

4.2.2 Hardware Interface

The user's Phone is required to be able to use the application.

4.2.3 Software Interface

The application is connected to the PetFinder API.

4.2.4 Communications Interface

The application communicates with the PetFinder API

5. Nonfunctional Requirements

5.1 Performance Requirements

Application must be able to process one user at a time.

5.2 Safety/Recovery Requirements

All user information and preferences are stored and dynamically updated as the user interacts with the application.

5.3 Security Requirements

The application is accessible for those with an account. Users will sign in with an email and password. The account information and passwords are stored on a private database.

5.4 Policy Requirements

Not applicable.

5.5 Software Quality Attributes

5.5.1 Availability

This application is only available on a local device, and is not web hosted. Therefore, it is only available when the local host is using it.

5.5.2 Correctness

Application only shows users pets that match their preferences.

5.5.3 Maintainability

After the completion of Fall 2020 CSC 340, this application will not be maintained.

5.5.4 Reusability

API translators and adapters allow for code to be reused in case of an API update.

5.5.5 Portability

Because we have implemented translators and adapters for our API calls, this application will be able to be updated if the PetFinder API is no longer available.

5.6 Process Requirements

5.6.1 Development Process Used

The Passive-Aggressive Post-It Notes utilized an Agile Development style with frequent scrum meetings to divide workload, keep each other up to date on personal progression, and troubleshoot bugs as a team.

5.6.2 Time Constraints

This application must be completed within the UNCG Fall 2020 Semester (fifteen weeks).

5.6.3 Cost and Delivery Date

This application will be finished for the final Code Review on Tuesday, December 1, 2020. There are no costs, as the student developers are working for class credit and all external software used is publicly available.