Software Requirements

Specification

for

No Kill Louisville Android App

Prepared by Nick Curry,

Sokheang Leang,

Christianne Maene,

Kei R

Indiana University Southeast

Individual Contributions

Nick Curry: team leader, developer, tester

Sokheang Leang: developer, tester

Christianne Maene: developer, tester

Kei R: developer, tester

Personal Information

Nick Curry: nscurry@iu.edu

Sokheang Leang: sleang@iu.edu

Christianne Maene: imaene03@gmail.com

Kei R: irouf@iu.edu

Introduction

Purpose

The purpose of the No Kill Louisville Pet Food Bank app is to facilitate operations during the first Saturday of each month when the pet food giveaway happens. The bottleneck that is being addressed is that volunteers struggle identifying pet owners during the giveaway. A Lot of paperwork gets in the way and slows down operations. With an android app, paperwork will be reduced and the check in and check out process will be facilitated.

Definitions

Keywords to be familiar with are: Android, API, Database.

System Overview

The app will be built using the JetBrains Android Studio with an Android API that can communicate with the local pet food bank's database.

Overall Description

Product perspective

The No Kill Louisville Pet Food bank app helps volunteers and pet owners get organized on pet food giveaway saturdays. The app allows volunteers to more quickly and efficiently identify pet owners. It also allows pet owners to confirm that they have arrived on time for the appointment.

System Interfaces

The app is connected to an API that updates and gets data from the No Kill Louisville database. The user uploads their information to the NKL database. Afterward, the user reserves a time to pick up pet food after their information is verified. The user would receive an available time to pick up pet food.

User Interfaces

Our UI will be simple and easy to use. It's important to make sure our users can use this as efficiently as possible.

Hardware interfaces

The app is designed for smartphones, specifically Android phones. It should be able to run on all Android phones as the app itself is lightweight.

Software interfaces

Since it's a mobile app, we will use Java and the API from No Kill Louisville's database. (version number?)

Communication Interfaces

Our app will communicate with their database to allow them to get information about their customers and ensure that only the proper ones can come in and get food for their pets.

Memory Constraints

Their database will host all of the user information so our app will need to be able to connect with it and check in and out their customers.

Operations

Using an API , we can support communication between our app and their systems.

Site Adaptation Requirements

Product functions

Our product will allow No Kill Louisville to check in their customers and check them out.

User characteristics

Anybody with a smartphone and internet connection can use the app to check in. (registration number required?)

Constraints, assumptions and dependencies

There are a couple of constraints to keep in mind:

- Finances: this app is made for pet food ban which is sustained via charitable donations. The app should have little to zero monetary costs for the food bank.
- Simplicity: the majority of pet owners are not techie. The most important things should be presented first and the app should be very simple to use.

A couple of assumptions are made in this case:

- Pet owners have android phones and can download an android based app.
- Pet owners speak English
- Pet owners have phone numbers for app verification.

There are some dependencies involved in App development. We will be using an API to create this app and we will have to inject some dependencies within our coding framework. API dependencies are often located on their main page in Github. Our app will be based off of a java API that will allow us to create this app via android studio. A possible API that we can use as a dependency is Firebase. It is almost free. It is free up to a certain number of users, However, it has a no cost plan and a pay as you go plan whenever users increase. However, being a local pet food bank, users should not exceed the generous no cost plan offered by firebase.

Specific requirements

External interface requirements

There should be at least 3 buttons the user can press.

- Reserve a time slot
- Add/update personal information
- Settings, used for enabling/disabling notifications.

UI should be simplistic with little to no clutter.

- Runs on any Android phones.
- Uses an API that connects to the No Kill Louisville database.

Functional requirements

- Users should add or update user's information to the database.
- Users should receive confirmation of data being added/updated successfully.
- Users should be able to reserve a time slot to pick up pet food.

- Users should be able to view their time slot to pick up pet food.
- Users should receive notification when its almost time to pick up pet food.

Performance requirements

- The application should be light in system usage.
- There should be little to no lags or delays bar the user's Internet connection.

Design constraints

Some app Design constraints are:

- The phone app must be able to authenticate registered users. To do so, our app should be able to talk to the website and storage application already in place.
- The phone app must be able to add new users and load the new information to the database.
- The phone app must have an additional interface for volunteers who can log in.
- Volunteers must be able to see appointments and information on pet owners.
- The App must be able to hide private information and restrict access as necessary.

Standards Compliance

Standards are laid out by the pet food bank and requirements gathering process. Basic compliances, by law, include protecting private information. The pet food bank gatherers information on income, SSNs, housing, etc… The pet food bank is therefore liable as to what

happens to that information. Some laws that apply to this are like The Privacy Act is an example of a compliance law.

Logical database requirement

Volunteers have phones and ipads on base during the pet food giveaway. Their android devices should be able to download the app. When using the app, all the information is stored on their database. The API communicates with the database.

Software System attributes

Reliability

- The Firebase API is widely used and tested for its reliability. Firebase also provides an exceptional platform for unit testing that we can all use as developers while building this application.

Availability

- The Application must be consistently available on every first Saturday of every month when the food giveaway happens.
- Firebase is available as an API on android applications. It is also available for use on android studio.

Security

- Firebase has many features that we can use for security testing.
- Some security features that we plan on implementing include:
    - Information hiding
    - Least privilege principle
    - Phone verification

Maintainability

Firebase is maintained by its company to make sure that its application stays free of bugs. On our part, we will be unit testing and integration testing to make sure that our app works the way we think it should.

Portability

Due to the app being designed for phones, it can be used anywhere Internet connection is available.

Other requirements

- We need to choose the cheapest option available.
- At least one team member should go onsite and train volunteers on using the app.

References

Techpass Master. (2021, November 6). Best API for Android App Development. *Techpass Master*. Retrieved October 5, 2022, from https://techpassmaster.com/best-api-for-android-app-development/