Project No-Kill

Jeffrey Becker, Jonathan Clements, Amanda Schneider

Sponsor: Dr. John Doyle

Last Updated: October 17, 2021

Software Architecture Specification

**1. Proposed Software Architecture**

**1.1 Overview**

The proposed software architecture of the No-Kill project will be a Client Management System which will consist of a self-check-in system utilizing a QR code scanned by the user upon arrival. The scan will send a message through Google Voice text to a tablet in the food bank. Food bank volunteers will acknowledge the text and dismiss the alert. The volunteer can then bring up the user’s information from the database to learn which items need to go out to the user. The system will accept user applications and attending documentation through the website and that information will be added directly into the database. The system will also track whether user documentation is up-to-date and attendance details from their recent visits to the food bank. The administrators will be able to access user applications and documentation for future and past visits.

**1.2 Subsystem Decomposition**

Component Diagram

Description automatically generated

**1.3 Hardware/Software Mapping**

Deployment diagram


Description automatically generated

**1.4 Persistent data management**

Diagram

Description automatically generated

* Source

Accounts will be created through an electronic form on the website, provided by a customer. An account will be approved or disapproved by a user with an appropriate level of access. Disapproved accounts will be retained for future vetting purposes.

* Customer

Table

Description automatically generatedTable

Description automatically generated

The Customer table acts as the primary source of personal information for each Customer. The id is the primary key and is the same for each id associated to that customer in each table. This table has the associated Enums EmpStatus to the Employment field and Status to the Status field. The ReturnDate field is generated by the Appointments table’s most recent date, and the annual schedule. The Address, Income, Email, Phone, FamilySize, and Employment are restricted views to higher privileged users.

Table

Description automatically generated

The Assistance table contains restricted information for higher level users to access. It lists assistance programs germane to the approval process, each a Bool, with the option for unlisted to be added by text. It associates with Customer by the CustID to the Customer id in a One-to-One relationship.

Table

Description automatically generated

Each customer is allowed an alternate to pickup their package, which this table allows to track. The primary key is AltID, generated at creation, and associated with Customer id with CustId one-to-one.

Table

Description automatically generated

For account acceptance and food pickup, adult cohabitant’s information is tracked. The table connects Customer id by the CustID in a one-to-one relationship. Income, Phone, and Employment are restricted.

* Pets

Table

Description automatically generatedTable

Description automatically generated

The primary key for the Dog and Cat tables are DogId and CatId, and they associate with the Customer table in a one-to-many relationship via the CustID. The Food and WetFood fields are a ration to be picked up and is derived from the dog’s weight. Cats receive a standard amount. DietRequest provides space for dietary needs or preferences. An accepted account requires the animal be spayed/neutered within a timeframe, which FixedDate allowed to be tracked. C/DFleaTick tracks the previous date a flea and tick medicine were provided.

Table

Description automatically generated

The CommunityCat table tracks non-pet cats cared for by the Customer. It has a one-to-one relationship with Customer via the id to ID. It is used for account acceptance and administrative purposes.

Table

Description automatically generated

The Client also provides and tracks: bird, hamster, fish, and guinea pig food. OtherPet accounts for each non-cat or dog that can consume one of those categories. CustID and Name are the primary keys, and it has a one-to-many relationship with Customer via the id to CustID.

* Appointment

Table

Description automatically generated

Appointments acts as a by Customer, by Date, repository of every appointment Customers have had, tracking a total for all items received. It has a one-to-many relationship with Customer via the id to CustID. Each instance is generated for the appointment based on ReturnDate in Customer and closed upon receipt by the customer. The NoShow field allows auditing the number of missed appointments. PeeDad is a sundry item added by need. The remaining items are derived from the total of a customer’s pets by category. For example, Dog is all a customer’s dog’s Food added together. Likewise Litter is derived from the number of (non-community) cats associated with the customer.

**1.5 Access control and security**

* Overview

The overall security and access control will be handled through a combination of best practices on the front and back end, as well as proper implementation of the security features build into the web framework.

* Log-in and Authentication

Django has built-in features supporting user and log-ins. It uses the PBKDF2 algorithm with a SHA256 hash as default for passwords but will support for other algorithms and hashes. Password validation and control methods are baked in the framework, including: length, common passwords, and a user attribute comparison.

* Security

General site security will be handled by implementing the web framework’s built-in features. They protect against many common issues, including: Cross site scripting, request forgery, and Clickjacking. HTTPS features will also be implemented via the framework’s supported features.

Data will be secured by limiting access behind tiered user access, and not allowing input raw queries. CRUD features will be accessed via form views that will have text controls for input to preformatted query bodies. An example of these controls is special character checking for names, and number/length checking for phone numbers. Further, the web framework’s features to limit the capacity to perform SQL injections will be implemented as well.

* User Operations
  + Admin:
    - Top level users will have full access to views with CRUD functionality. This includes the ability to view restricted data, validate a Customer’s account, and updating all aspects associated with the Customer and Volunteer form fields.
    - Admin users will have access to the query the Appointments table for food/items provided.
    - All features of lower access
  + Volunteer
    - Limited queries for Customer and their pets, without restricted information. Including the Customer name, Notes, Pet information
    - Affirm that a Customer’s order was fulfilled, updating the Customer’s NoShow field via a button.
  + Customer
    - Create an account via a form.
    - View information of account.

**1.6 Global Software Control**

**Diagram

Description automatically generated**

**1.7 Boundary Conditions**

The system will be started by the website and database being deployed through the web host. Usernames and passwords will differentiate levels of access for admins and volunteers. The admin will have access to everything while the volunteers have access to names, vehicles, notes and amount of food. When the tablet is started the volunteer will sign into the app. As the customers arrive and check in through the app, using the Google number provided, the volunteer in charge will tell the person taking it to the customer where they are located. The volunteer will validate the person and click a box on the app stating pick-up is complete. At the end of the day the volunteer will sign out of the app and turn off the tablet. The admin will be able to export the information to make graphs for grants. There will be required fields that have to filled out for the customer to submit their application. If they are not, a reminder will appear on the screen telling them the application is not complete. If someone is there to pick up for another person and their text their name, if it is not in the system the screen of the tablet will tell the volunteer that name is not in the file. The website and database will be web hosted so if new hardware is bought, it should not matter, the migration from server to server will be handled by the webhost.