# Design Specification

Design Specification for the Greyhound Pets of America Greater Orlando Web Application

February 2, 2014

Team 1: Phillip Bess, Manuel Gutierrez, Patrick McAleavey, Jamie Smith, Jeff Woodard

Design Specification 1

Description 3

Objectives 3

Technical Overview 3

Architecture 4

Security 5

Logging 5

Javadoc 5

Data Dictionary 5

UI Overview 6

## Description

Today, the Greyhound Pets of America – Greater Orlando (GPAGO) website is very time consuming and tedious to maintain. The static html web page listing greyhounds available for adoption must be manually updated. Furthermore, information on greyhounds is tracked using Microsoft Excel spreadsheets. Maintenance of these spreadsheets is also tedious and the usefulness of the information is not optimal.

## Objectives

This project will deliver a web based application that will allow records of greyhounds that are available for adoption to be entered and updated. Access to this application will be over the internet through a web browser. This project will only focus on the dynamic content and the management of it. This project will not focus on the overall web site content or layout.

The primary objective of this project is to reduce the time and effort required by staff to enter and update greyhound records that are displayed on the GPAGO web site.

The system should be designed such that it may evolve to become a complete record keeping system for greyhound information.

## Technical Overview

* The solution will be a web based application written in java.
* Servlets and Jsp will be used to render HTML and Javascript to the client.
* Tomcat will be used as the Servlet container.
* MySql database will be used for persistence.
* JPA (Eclipselink) will be used for object-relational mapping.
* The architecture will follow the Model-View-Controller (MVC) pattern.
* JPA Entities objects and a facade class will comprise the model.
* Servlets will comprise the controller.
* Html, Jsp, and Javascript will comprise the view.
* The application is developed entirely using Java. The system requires a Java Runtime Environment version 1.7 or greater.

## Architecture

**JPA**

Eclipselink will be used as the JPA provider.

A JTA datasource will be used in the production runtime environment.

The EntityManager will be managed by the container (i.e. Tomcat) and a reference to the EntityManager will be obtained through dependency injection through the use of the @PersistenceContext annotation. See <http://docs.oracle.com/javaee/6/api/javax/persistence/PersistenceContext.html> for reference.

## Security

Declarative Form Based Security will be used to protect all web resources that must be protected. Protected resources include all admin related pages and resources.

For Tomcat, the intended runtime web server, users and roles will be specified in the tomcat-users.xml File.  For example, the following can be added to this file:

<role rolename="petManager"/>  
<user username="user1" password="password" roles="petManager"/>

## Logging

java.util.logging.Logger will be used.

See <http://docs.oracle.com/javase/6/docs/api/java/util/logging/Logger.html> for reference.

## Javadoc

All classes should have javadoc.

All non-trivial methods should have javadoc.

## Data Dictionary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Greyhounds - Table that contains greyhounds available for adoption** | | | | | |
|  | **Name** | **Type** | **Size** | **Source** | **Purpose** |
|  | id | Long Integer | 4 |  | Primary Key of Greyhound record |
|  | name | varchar2 | 80 |  | Name of greyhound |
|  | racename | varchar2 | 80 |  | Race name of greyhound |
|  | isCatFriendly | Yes/No | 1 |  | Is dog friendly to cats? |
|  |  |  |  |  |  |

Other fields and entities tbd

## UI Overview

(TBD)

* Login Screen
* Table Listing Greyhounds available for adoption
* Management Screens

## Disaster Recovery

Backups should be maintained of the MySql database. Backups should be generated on a regularly scheduled basis. Backups should be automated if possible and copied to another system for safe storage.

A backup image of the Tomcat configuration should be kept. The backup should be updated any time the Tomcat configuration changes.