**Requirements Document**

Greyhound Pets of America – Greater Orlando

**Team 1**

Phillip Bess

Manuel Gutierrez

Patrick McAleavey

Jamie Smith

Jeff Woodard

**1.** **Project Overview**

**1.1 Vision**

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.

This project will deliver a web based solution that will allow greyhound records to be added and updated over the internet through a web browser. Moreover, this solution will provide a complete record keeping system for greyhound information.

**1.2 Overall Description**

The solution will be a web application accessible from the internet. Two types of users will use this system:

General User: General public who will access this web site over the internet to view information about the organization and greyhounds that are available for adoption.

Admin: Administrator user who will manage greyhound information.

The solution will not only store records of greyhounds currently available for adoption but will also be the system of record for all greyhounds with the intent of maintaining records indefinitely. The system will track information on each greyhound including its adoption history. The system must allow an administrator to easily enter information on new greyhounds and to update this information over time. The system must also allow greyhound information to be queried and reported on (exact requirements tbd).

The system will be developed in java and must use a MySql database to persist information.

**2. Software Requirements**

**2.1 High-Level Requirements**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Added** | **Description** | **(O)ptional or (R)equired** |
| 100 | Onset | e.g., Re-designing the interface using Twitter bootstrap UI | **R** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**2.2 Low-Level Requirements**

|  |  |  |
| --- | --- | --- |
| **ID** | **Description** | **Verification** |
| 110 | e.g., Users should be able to copy the XML/JSON content by clicking a button | User Testing |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**3. Constraints and Limitations**

|  |  |
| --- | --- |
| **Constraint** | **ID** |
| e.g., Team will not provide alternative language pack files. | 100 |
|  |  |
|  |  |
|  |  |

**4. Definitions and Acronyms**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| API | e.g., Application Programming Interface |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**5. Requirements Review**

|  |  |  |  |
| --- | --- | --- | --- |
| Name: | <Name> | Date: | <Date> |
| Signature: | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Name: | <Name> | Date: | <Date> |
| Signature: | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Name: | <Name> | Date: | <Date> |
| Signature: | | | |