CAMPUS ANIMALS
Software Architectural Design

Submitted to:

Asst. Prof. Ma. Rowena C. Solamo
Faculty Member
Department of Computer Science
College of Engineering
University of the Philippines, Diliman

Submitted by:
Gallardo, Arianne Joshin U.
Yabut, Roy Christopher T.

In partial fulfillment of Academic Requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2018-2019

Unique Reference:

The documents are stored in the GitHub repository, bit.ly/CampusAnimals referenced with the filename, Campus Animals - Architectural Design.pdf.

Purpose:

[Place purpose of the document here.]

Audience:

Residents of UP Diliman.

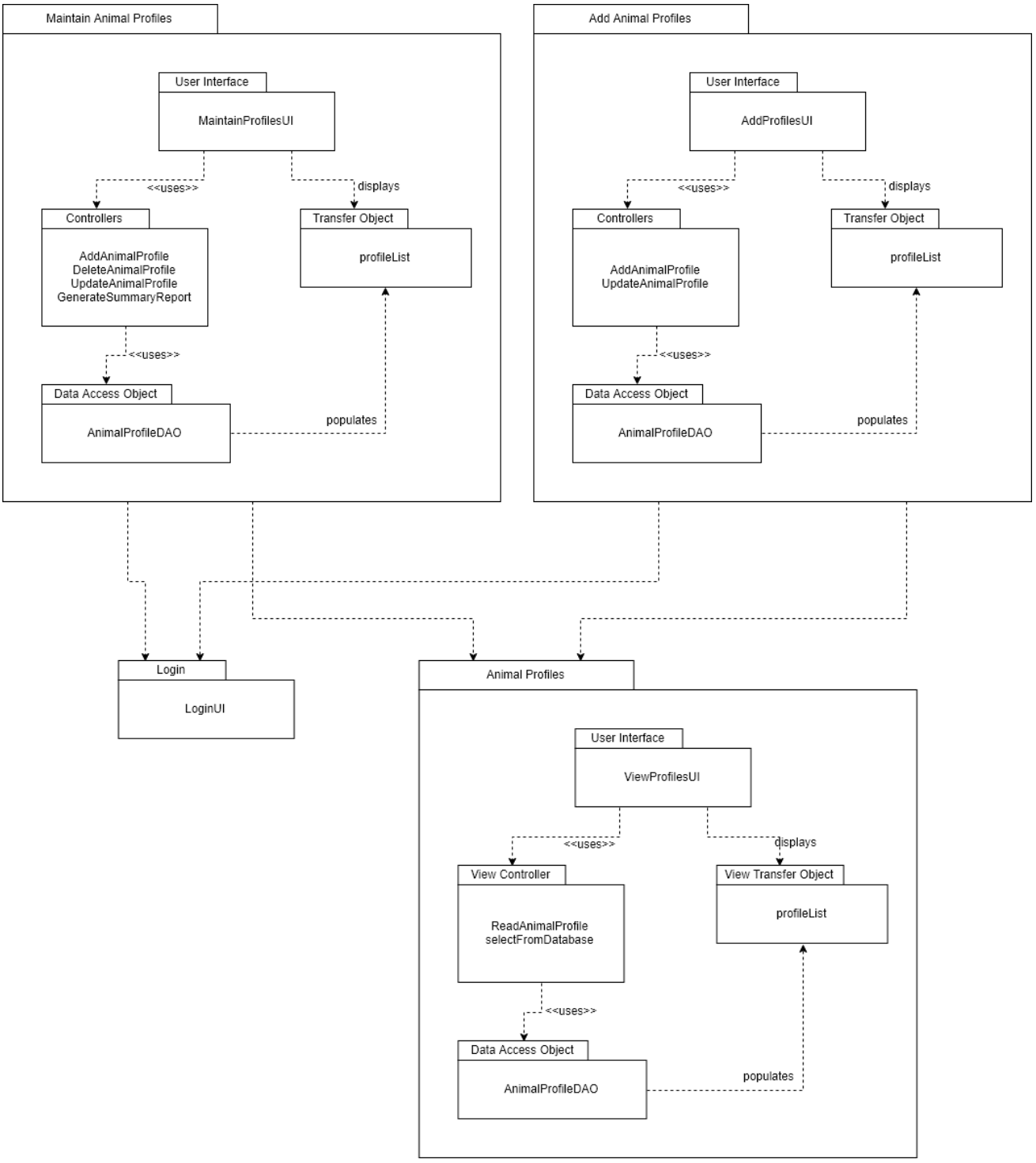
Revision Control:

Revision Date	*Person Responsible*	*Version Number*	*Contribution/Modification*
11/06/2018	Arianne Gallardo	1.0	Initial Document;
11/07/2018	Arianne Gallardo	2.0	Added the system description and the revised software architecture model; Updated the User Interface Packages and their descriptions
11/08/2018	Arianne Gallardo	3.0	Added controllers, transfer objects, and DAO packages with their descriptions

***System Name*:** Campus Animals

Description: Campus Animals is a profiling application and a database intently created to keep track of the animals residing in the University of the Philippines Diliman Campus.

***Revised Software Architecture Model*:**



User Interface Package:

Screen Name	Description
MaintainProfilesUI	This is the interface of an Secretary General to the system when maintaining the database. This includes the features: generate a summary report, view, add, update, and delete profiles. *Attributes*: private Profile AnimalProfile; private List ProfileList; *Responsibilities*: public void generateSummaryReport() public void filterForCats() public void filterForDogs() public void filterForBuildings() public void filterForAdoption() public void saveAnimalProfile() public void showErrorMessage() public void chooseProfile() public void chooseProfile(int profile_no)
AddProfilesUI	This is the interface of an Secretary General to the system when maintaining the database. This includes the features: generate a summary report, view, add, update, and delete profiles. *Attributes*: private Profile AnimalProfile; private List ProfileList; *Responsibilities*: public void saveAnimalProfile() public void showErrorMessage() public void chooseProfile() public void chooseProfile(int profile_no)
ViewProfilesUI	This is the default interface of a user to the system wherein profiles of each animal are readily displayed. *Attributes*: private Profile AnimalProfile; private List ProfileList; *Responsibilities*: public void showProfiles() public void filterProfiles(Boolean dogs, Boolean cats, String location)

Controllers Package:

Controller Name	Description
GenerateSummaryReport	This is the control that creates summary reports for viewing. It takes note of filters. *Responsibilities*: viewSummaryReport()
AddAnimalProfile	This is the controller that takes care of additional profiles *Attributes*: private Profile AnimalProfile; private List ProfileList; *Responsibilities*: private boolean addAnimalProfile(String name, String species, char sex = [M or F], String color, String markings, String BuildingOfOrigin, boolean adopted, boolean vaccinated, boolean neutered, String remarks, String cage_label)
UpdateAnimalProfile	This is the controller that deals with selecting and editing profiles *Attributes*: private Profile AnimalProfile; private List ProfileList; *Responsibilities*: viewProfileList() viewForm() editAnimalProfile(profile_no)
DeleteAnimalProfile	This is the controller that deals with deleting profiles *Attributes*: private Profile AnimalProfile; private List ProfileList; *Responsibilities*: viewProfileList() editAnimalProfile(profile_no)
SelectFromDatabase	This is the controller that determines which set of profiles should be shown in the screen. *Attributes*: private Profile AnimalProfile; private List ProfileList; *Responsibilities*: selectFromDatabase(boolean dogs, boolean cats, String location)

Data Access Objects Packages:

DAO Name	**Description**

DBAnimalProfile	This data access object is responsible for getting data of an animal profile from a database. This DAO is used by the controllers to create new transfer objects, which will then be fetched or obtained by the user interfaces to display information. The UI's may also modify these transfer objects based on the input of either a visitor or a registered user.
	Attributes:
	private Profile AnimalProfile;
	private List ProfileList;
	Methods:
	private void connectDatabase(String URL, String user, String password);
	private void insertProfile(Profile AnimalProfile);
	private void updateAthlete(Profile AnimalProfile);
	private void deleteAthlete(Profile AnimalProfile);

Transfer Objects Package:

Class Name	Description

registered_userList	This transfer object or persistent class is responsible for holding a list of registered_users. This will be useful for the maintaining the access to the actual database.
	
	Attributes:
	private List registered_users;
	private int users_count;
	
	Methods:;
	private void sortAlphabetically();
	private void groupByAccountType();
	private Boolean addProfile(String username);
	private Boolean deleteProfile(String username);

ProfileList	This transfer object or persistent class list is responsible for holding a list of animal profiles. This is useful for the *Filter Animal Profiles* feature wherein only elect animal profiles will be shown based on the preferences set by the user.
	
	Attributes:
	private List ProfileList;
	private int profile_count;
	
	Methods:;
	private void sortAlphabetically();
	private void groupBySpecies();
	private void groupByLocation();
	private void getNeutered();
	private void getVaccinated();
	private void getAdopted();
	private Boolean addProfile(String username);
	private Boolean deleteProfile(String username);

registered_user	This transfer object or persistent class is responsible for holding a single registered_user profile. This contains the information regarding a registered user account, including its credentials. *Attributes*: private String username; private String firstName; private String LastName; private String account_type = [admin or regular];

	private String encrypted_password; // the actual password is not stored in the database
	Methods:;
	private void FetchProfileData(String username);
	private void updateUserCount();
	private Boolean ModifyProfileData(String modifications);

profile	This transfer object or persistent class is responsible for holding a single animal profile. This contains all the information on the animal profile.
	
	Attributes:
	public String name;
	public String species;
	public char sex = [M or F];
	public String color;
	public String markings;
	public String BuildingOfOrigin;
	public boolean adopted;
	public boolean vaccinated;
	public boolean neutered;
	public String remarks;
	public String cage_label;
	
	Methods:;
	private void FetchUserData(String username);
	private void updateUserCount();
	private Boolean ModifyUserData(String modifications);