## Global Rain Logo

**Developer**: Andrew Vue

**Date**: March 2020

# Summary Report Template

**Directions:** Place your pseudocode, flowchart, and explanation in the following sections. Before you submit your report, remove all bracketed [ ] text.

## Pseudocode

When you are done implementing the Pet class, refer back to the Pet BAG specification document and select either the pet check-in or check-out method. These methods are detailed in the Functionality section of the specification document.

Write pseudocode that lays out a plan for the method you chose, ensuring that you organize each step in a logical manner. Remember, you will not be creating the actual code for the method. You do **not** have to write pseudocode for both methods. Your pseudocode must not exceed one page.

|  |
| --- |
| PROGRAM checkIn:  Start:  Print “Welcome to Pet Board and Grooming”  User input select pet type: <dog> or <cat>.  If dog search open spaces \_/30  If cat search open spaces \_/12  If space is available, then select: <returning visitor> or <new visitor>  Else return “No spaces available at this time”  If <new visitor> is selected, then user input information about visitor:  Return prompt User input: Pet type <cat> or <dog>  If <dog> Return prompt user input: Pet weight  If else <returning visitor> selected, then return print visitor information for user to check visitor information if correct. Prompt user input pet weight:  Prompt user input length of stay for each pet  If pet is <Dog> and staying more than 2 days, Prompt user select if grooming services is required  Else return print “No grooming services are offered for cats”.  If pet is <dog> assign to open space for dog  If else <cat> assign to open space for cat  End Check-in |

## Flowchart

Based on the pseudocode you wrote, create a flowchart using a tool of your choice for the method you selected. In your flowchart, be sure to include start and end points and appropriate decision branching, and align the flowchart to the check-in/check-out process. Your flowchart must be confined to one page.

|  |
| --- |
|  |

## OOP Principles Explanation

Briefly explain how you applied object-oriented programming principles in the software development process. Your explanation should be one paragraph, or four to six sentences.

|  |
| --- |
| The Pet class in Pet.java applies inheritance for the other classes such as the Cat and Dog classes. Fields and methods shared between the Cat and Dog classes are defined once to prevent redundancy in code. The class attributes are inherited by the predecessor classes that have the same properties and functions. This way redundancy is either prevented or reduced.  Encapsulation is applied through by the ‘private’ access modifiers to the class fields, whilst applying ‘public’ access modifiers to the setter and getter methods. Private access to the member fields prevents outside modification through simple assignment and stops unintended consequences through accidental modification. Public access to member methods gives the ability to read and write to member fields from outside classes but only when an explicit method call is made to be able to do so. Encapsulation is one of the fundamental principles of the Object-Oriented Programming (OOP), because it promotes the containing of data to a single package.  Polymorphism is applied when we implement the definition of multiple constructors in the Pet class. There is the default, no argument, constructor and a parameterized constructor defined with multiple parameters. The constructor is overloaded through many different definitions, only the appropriate ones will them be called at compile time depending on the arguments provided in the object instantiation call.  Conditional logic is implemented through multiple if-else statements in the checkIn() method. Different paths are implemented when certain conditions are met. For example, if the type of pet to be checked in is a Dog, the conditional logic path would be to check for available Dog spaces and if it has availability and does not need to access anything regarding Cat. If the pet to be checked in is a Cat the same process is repeated, but this time only for the Cat code path. This is also used to figure out if the pet needs grooming services while staying at Pet BAG, only if the pet is a dog if that condition is met then the program checks for length of stay, and pet size. While loops can be implemented in here to save time on writing multiple if statements because it will automatically route the program to the correct path instead of asking for multiple inputs. |