

**Developer**: Ashley Littles

**Date**: 5/27/24

# IT 145 Global Rain Summary Report Template

## Pseudocode Pet Check-In

Check your pet type: DOG or CAT

* IF pet is DOG:
  + Check spaces (30):
  + IF no space available:
    - STOP;
* ELSE IF one space available:
  + - Check if new or returning:
    - IF new:
      * Collect appropriate information:
        + petName;
        + petAge;
        + daysStay:

IF days of stay is two or more:

Check for grooming

IF groomed:

Select groom;

Assign pet to space;

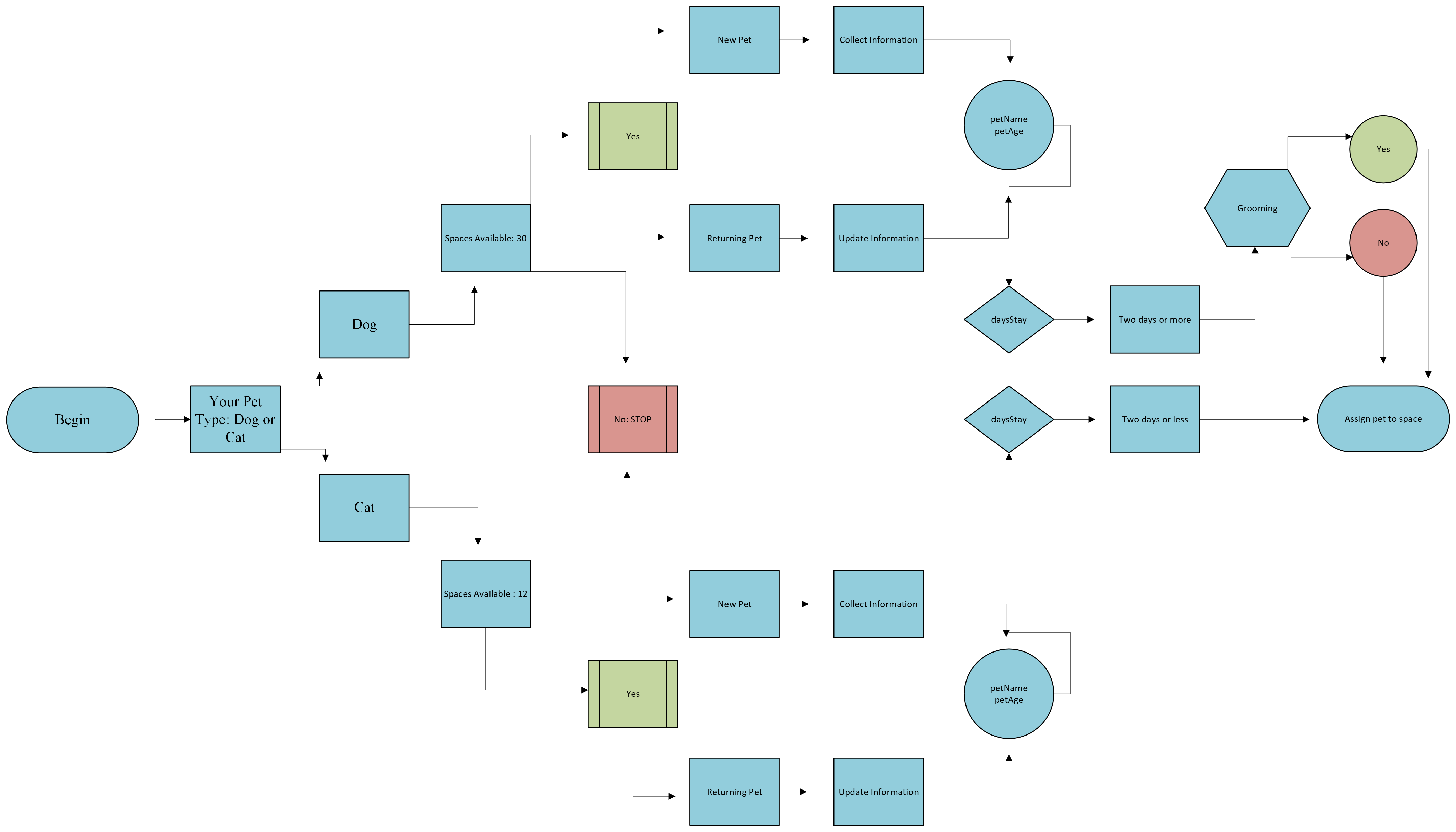
* ELSE IF days of stay is less than two days:
  + Assign pet to space;
* ELSE IF returning pet;
  + Update pet information if needed;
    - petName;
    - petAge;
    - daysStay:
      * IF days of stay is two or MORE:
        + Check for grooming;
        + IF groomed:

Select groom;

Assign pet to space;

* ELSE IF days of stay is less than two days:
  + Assign pet to space;
* IF pet is CAT:
  + Check spaces (12);
    - IF no space available:
      * STOP;
* IF space available:
  + Check if new pet or returning pet:
    - IF pet is NEW:
      * Collect appropriate information:
        + petName;
        + petAge;
        + daysStay;
* Assign pet to space;
* ELSE IF returning pet:
  + - Update pet information if needed:
      * petName;
      * petAge;
      * daysStay;
* Assign pet to space;

## Flowchart



## OOP Principles Explanation

So far, I have implemented the object-oriented programming principles encapsulation, abstraction, inheritance, and polymorphism into my software development work. Encapsulation was used to identify the setters such as setPetName. Abstraction was used to keep attributes in specific classes private. Inheritance displayed the common connections within my flowchart. Polymorphism was used to allow for connecting the different classes in the pseudocode.