## Project 1 – Reptiles –Design Document

## =<enumeration>> Habitat Type - habitatID : int Reptile - MAX NATURAL FEATURE : int(3) Amphibian - maxSizeLimit : int <<abstract>>Animal - List<NaturalFeature> : NaturalFeature =<enumeration>> - specie : String + getNaturalFeatures(): List<NaturalFeature> characteristics : String + getCountOfNaturalFeatures(): int Small - min temp : float + toString(): string Medium - max temp : float Large poisonous : boolean extinct : boolean <<enumeration>> - endangered : boolean **NaturalFeature** NaturalFeatureType liveWithOtherSpecie : Animal Dessert naturalFeatureID: int + getCharacteristics(): String Fallen\_Logs countOfAnimals: int + getSpecie(): String Flowing\_Water · List<Animal> : Animal + getFeatures(): String Grass + getLivingList(): List<Animal> Lily\_Pads + toString() + getAnimalList() : List<Animal> Ponds + getCountOfAnimals(): int Rocks + toString(): string Sand Beach + addAnimal() : Animal Salt Water Tree\_Branches Snake Alligator Turtle Salamander Chameleon Frog + getCharacteristics() + getCharacteristics() + getCharacteristics() + getCharacteristics() + getCharacteristics() + getCharacteristics() + getSpecie() + getSpecie() + getSpecie() + getSpecie() + getSpecie() + getSpecie() + getFeatures() + getFeatures() + getFeatures() + getFeatures() + getFeatures() + getFeatures() + aetLivinaList() + aetLivinaList() + aetLivinaList() + aetLivinaList() + aetLivinaList() + getLivingList() + toString() + toString() + toString() + toString() + toString() + toString()

## **UML Diagram**

## **Testing Plan**

@Test for entry of a particular animal in a habitat

assertEquals("Habitat ID: 1, Natural Feature ID: 2, Specie: rattlesnake, Characteristics: has a special rattle on its tail, Type: Reptile, Size: Large, Natural Feature Type: Dessert, Temperature range: 35-50 C, Poisonous: true, extinct: false, endangered: true, Can Live with: Snake", rattlesnake.toString())

@Test for constructor of animal entry (Expected = IllegalArgumentException)

assertEquals("Habitat ID: 1, Natural Feature ID: 2, Specie: rattlesnake, Characteristics: Highly venomnous, Type: Reptile, Size: Big, Natural Feature Type: Dessert, Temperature range: 35-50 C, Poisonous: true, extinct: false, endangered: true, Can Live with: Snake", cobra.toString())

@Test for more than 3 Natural Feature in Habitat (Expected = IllegalArgumentException)

NaturalFeature.addAnimal(Specie: Turtle, Characteristics: Lives on land and water, Type: Reptile, Size: Big, Natural Feature Type: Ponds, Temperature range: 30-40 C, Poisonous: false, extinct: false, endangered: true, Can Live with: Turtle", turtle.toString()))

@Test for amphibian animal in habitat

assertEquals("Habitat ID: 2, Natural Feature ID: 3, Specie: tree frog, Characteristics: slicky skin, Type: Amphibian, Size: Small, Natural Feature Type: Pond, Temperature range: 20-30 C, Poisonous: false, extinct: false, endangered: true, Can Live with: Frog, Salmander, Chameleon", frog.toString())

@Test characteristics list

assertEquals(slicky skin,frog.getCharacteristics())

@Test Animal List

assertEquals("rattlesnake",naturalFeature1.getAnimalList())

assertEquals("tree frog, common toad",naturalFeature3.getAnimalList())

@Test natural feature count in Habitat list

assertEquals(2,habitat1.getCountOfNaturalFeatures())