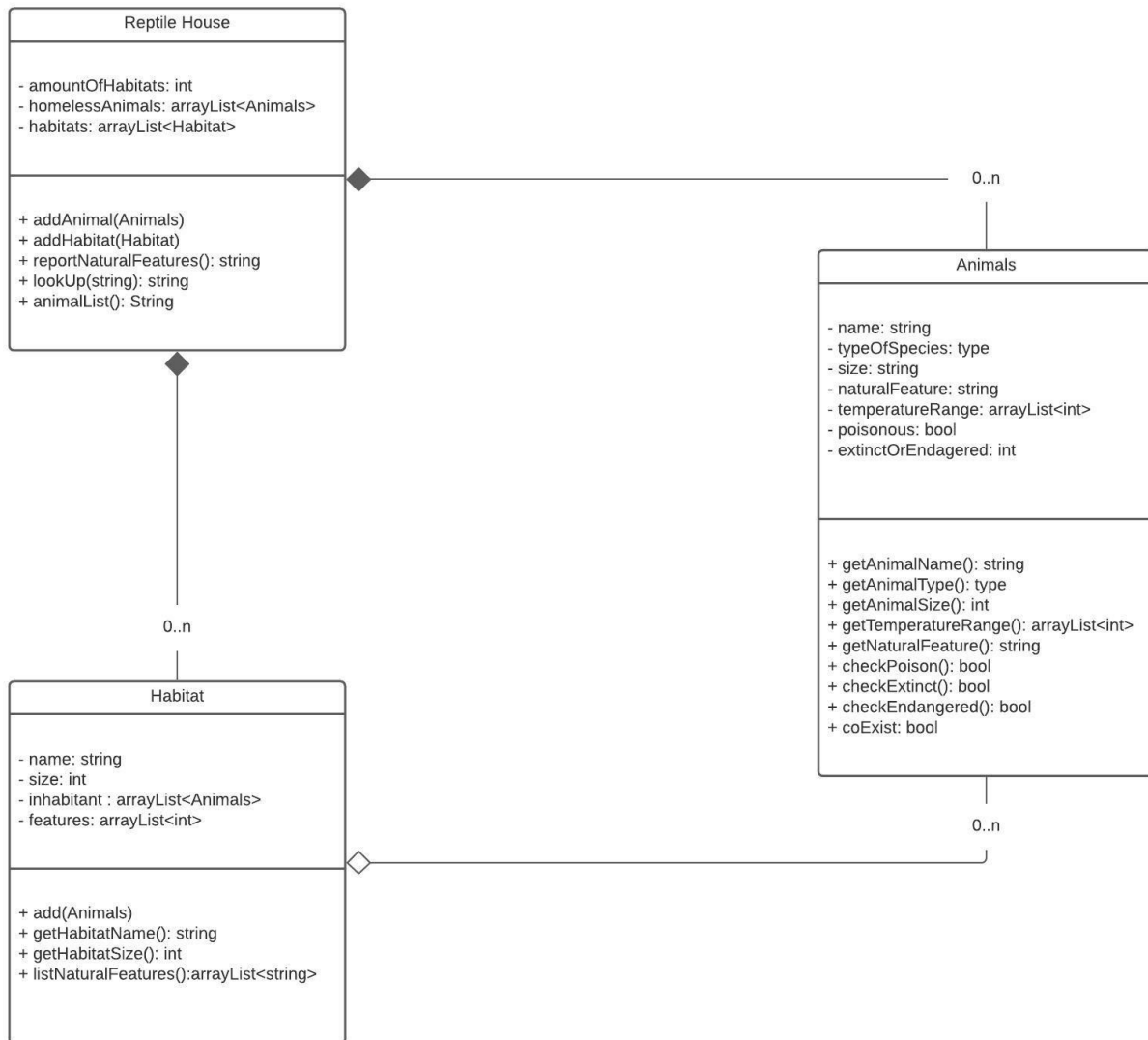


Project Design



UML Diagram

Testing

Testing Animal Class		Expected Value
Animal Name	getAnimalName()	"chameleon"
Animal Type	getAnimalType()	REPTILE
Size	getAnimalSize()	"small"
Temperature range	getTemperatureRange()	[10,20]
Natural Feature	getNaturalFeature()	"tree branch"
Poisonous	checkPoisonous()	false
Extinct	checkExtinct()	false
Endangered	checkEndangered()	false
Live with others	coExist()	true
Constructor disallows type not defined in enum	Animal("chameleon",MAMMAL)	IllegalArgumentException
Constructor disallows size not acceptable	Animal(REPTILE,"huge")	IllegalArgumentException

Animals decy = new Animals("chameleon", REPTILE, "small",10,20,"tree branch", false,2)

Testing Habitat		Expected Value
Check Name	getHabitatName()	"hiders"
Check Size	getHabitatSize()	30
Natural Feature	listNaturalFeatures()	["tree branches","grass","lily pads"]
Disallows non animal objects	add(frog)	IllegalArgumentException
Constructor disallows negative values	Habitat("hiders",-30,["tree branches","grass","lily	IllegalArgumentException

	pads”]))	
Constructor disallows array without positive int	Habitat(“hiders”,30,[]))	IllegalArgumentException

Habitat forest = new Habitat(“hiders”,30,[“tree branches”,“grass”,“lily pads”])

Reptile House		Expected Value
Generate Natural Features	reportNaturalFeatures()	““tree branches,grass,lily pads located at hiders”
Look Up	lookUp(“chameleon”)	“hiders”
Look Up	lookUp(“lion”)	“Not in jurassic”
Animal List	animalList()	“chameleon”
constructor disallows non animal obj	addAnimal(forest)	IllegalArgumentException
constructor disallows non habitat obj	addHabitat(decy)	IllegalArgumentException

ReptileHouse jurassic = new ReptileHouse(3)
jurassic.addHabitat(forest)
jurassic.addanimal(decy)