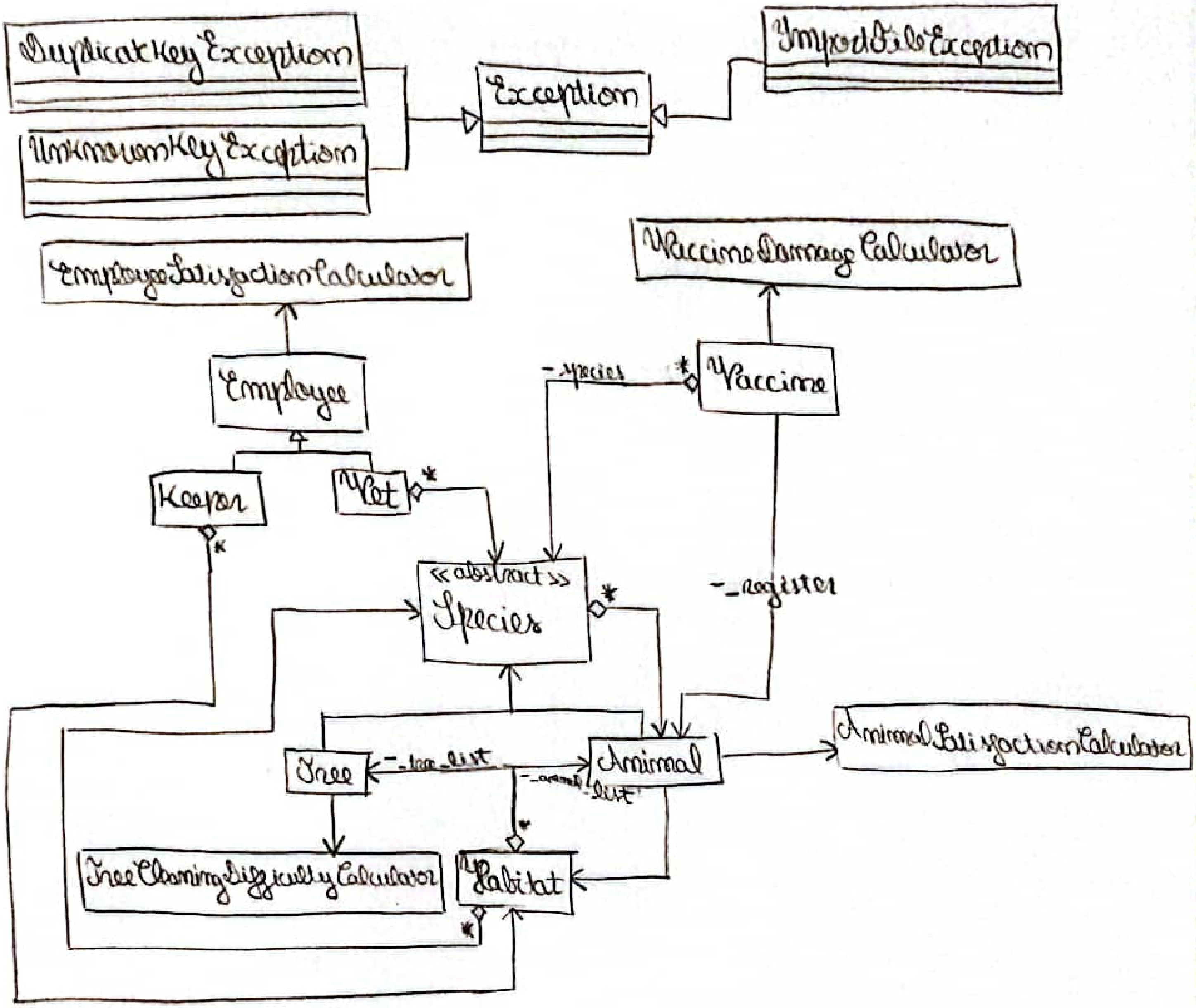
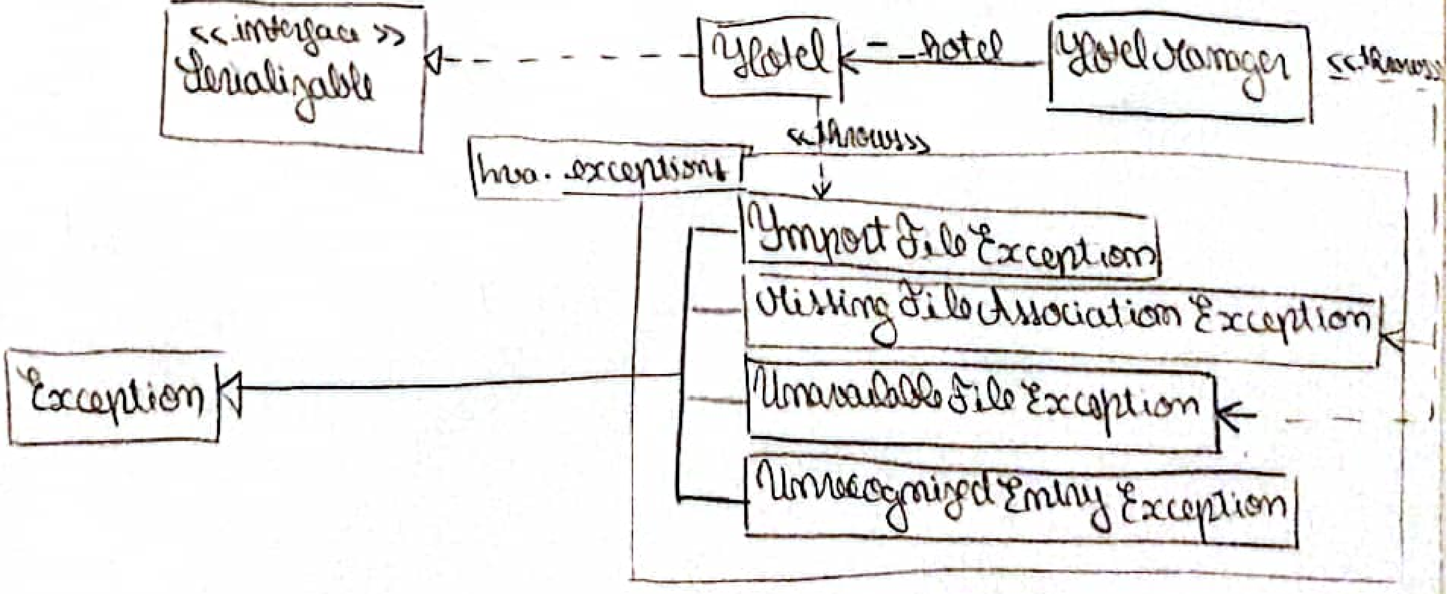




Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo do projeto. Madalena Mota

Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo do projeto. Miguel Morais

```
<<abstract>>
Species
-_id: String
-_name: String
-_animal_register: Animal[]
-_number_of_vets: int
-_population: int
```

```
Animal
-_id: String
-_name: String
-_species: Species
-_health: String
-_satisfaction: double
-_satisfaction_calculator: AnimalSatisfactionCalculator
-_habitat: Habitat
```

```
AnimalSatisfactionCalculator
-equals(a: Animal, h: Habitat): int
-different(a: Animal, h: Habitat): int
-getHabitatArea(h: Habitat): int
-getHabitatPopulation(h: Habitat): int
-getAdequacy(a: Animal, h: Habitat): int
+calculateAnimalSatisfaction(a: Animal): double
```

```
Habitat
-_id: String
-_name: String
-_adequacy: Map<Species, int>
-_area: int
-_population: int
-_animal_list: Animal[]
-_tree_list: Tree[]
```

```
Tree
-_id: String
-_name: String
-_age: int
-<<final>>_leaf_type: String
-<<final>>_base_cleaning_difficulty: int
```

```
TreeCleaningDifficultyCalcula
-getBaseCleaningDifficulty(t: Tree): int
-getSeasonalEffort(t: Tree): int
-getAgeFactor(t: Tree): double
+calculateCleaningDifficulty(t: Tree): dou
```

```
KeeperSatisfactionCalculator
-getTotalCleaningDifficulty(h: Habitat): int
-getWorkInHabitat(h: Habitat): int
-getKeepersOfHabitat(h: Habitat): int
-getWork(k: Keeper): int
+calculateKeeperSatisfaction(k: Keeper): double
```

```
VetSatisfactionCalculator
-getSpeciesPopulation(s: Species): int
-getNumberOfVets(s: Species): int
-getWork(v: Vet): int
+getVetSatisfaction(v: Vet): double
```

```
Employee
-_id: String
-_name: String
```

```
Keeper
-_habitats: Habitat[]
-_trees_under_care: int
-_satisfaction_calculator:
      KeeperSatisfaction_Calculator
-_satisfaction: double
```

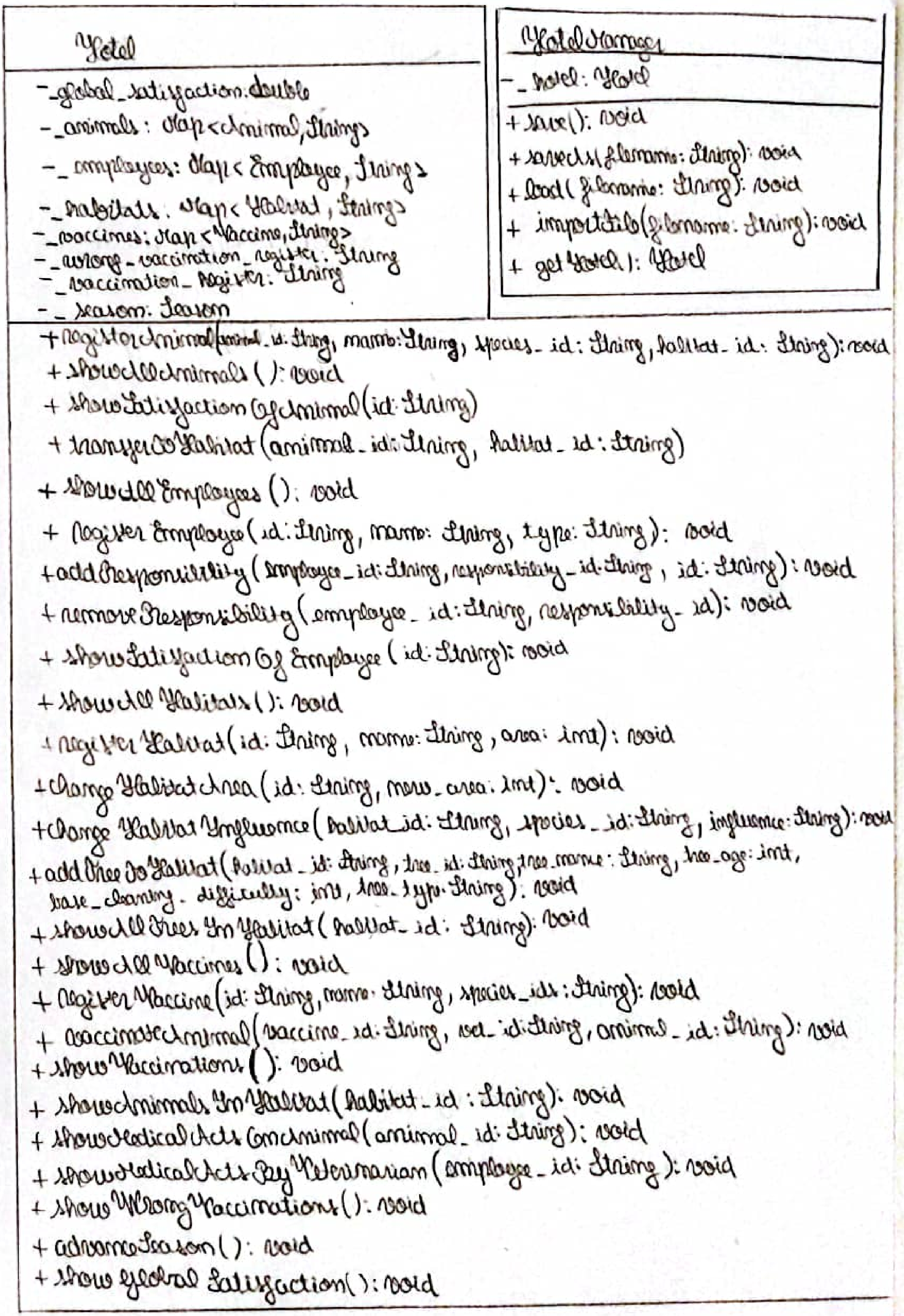
```
Vet
-_species: Species[]
-_animals_under_care: int
-_satisfaction_calculator: VetSatisfactionCalculator
-_satisfaction: double
```

```
Vaccine
-_id: String
-_name: String
-_species: Species[]
-_register: Animal[]
-damage_calculator: VaccineDamageCalculator
```

```
VaccineDamageCalculator
-getNameLength(s: Species): int
-getMaxNameLength(s1: Species, s2: Species): int
-getCommonChars(s1: Species, s2: Species): int
+calculateDamage(v: Vaccine, s: Species): int
```

```
<<enumeration>>
Season
Spring
Summer
Autumn
Winter
```

Digitalizada com CamScanner



```
Hotel
-_global_satisfaction: double
-_animals: Map<Animal, String>
-_employees: Map<Employee, String>
-_habitats: Map<Habitat, String>
-_vaccines: Map<Vaccine, String>
-_wrong_vaccination_register: String
-_vaccination_Register: String
-_season: Season
+registerAnimal(animal_id: String, name: String, species_id: String, habitat_id: String): void
+showAllAnimals(): void
+showSatisfactionOfAnimal(id: String)
+transferToHabitat(animal_id: String, habitat_id: String)
+showAllEmployees(): void
+registerEmployee(id: String, name: String, type: String): void
+addResponsibility(employee_id: String, responsibility_id: String, id: String): void
+removeResponsibility(employee_id: String, responsibility_id): void
+showSatisfactionOfEmployee(id: String): void
+showAllHabitats(): void
+registerHabitat(id: String, name: String, area: int): void
+changeHabitatArea(id: String, new_area: int): void
+changeHabitatInfluence(habitat_id: String, species_id: String, influence: String): void
+addTreeToHabitat(habitat_id: String, tree_id: String, tree_name: String, tree_age: int,
 base_cleaning_difficulty: int, tree_type: String): void
+showAllTreesInHabitat(habitat_id: String): void
+showAllVaccines(): void
+registerVaccine(id: String, name: String, species_ids: String): void
+vaccinateAnimal(vaccine_id: String, vet_id: String, animal_id: String): void
+showVaccinations(): void
+showAnimalsInHabitat(habitat_id: String): void
+showMedicalActsOnAnimal(animal_id: String): void
+showMedicalActsByVeterinarian(employee_id: String): void
+showWrongVaccinations(): void
+advanceSeason(): void
+showGlobalSatisfaction(): void
```

```
HotelManager
-_hotel: Hotel
+save(): void
+saveAs(filename: String): void
+load(filename: String): void
+importFile(filename: String): void
+getHotel(): Hotel
```