





Hotel Manager

- + Save(): void
- + SaveAs(filename: String): void
- + load(filename: String): void
- + importFile(filename: String): void
- + registerAnimal(AnimalId: String, name: String, SpeciesId: String, habitatId: String): void
- + ShowAllAnimals(): void
- + ShowSatisfactionOfAnimal(): void
- + TransferToHabitat(AnimalId: String, newHabitatId: String): void
- + ShowAllEmployees(): void
- + RegisterEmployee(employeeId: String, name: String, type: String): void
- + AddResponsibility(responsibilityId: String, employeeId: String): void
- + RemoveResponsibility(employeeId: String, responsibilityId: String): void
- + ShowSatisfactionOfEmployee(employeeId: String): int
- + ShowAllHabitats(): void
- + RegisterHabitat(habitatId: String, name: String, habitatArea: int): void
- + ChangeHabitatArea(habitatId: String, newArea: int): void
- + ChangeHabitatInfluence(habitatId: String, speciesId: String, habitatInfluence: int): void
- + AddTreeToHabitat(habitatId: String, treeId: String, name: String, treeAge: int, treeDifficulty: int, treeType: String): void
- + ShowAllTreesInHabitat(): void
- + ShowAllVaccines(): void
- + RegisterVaccine(vaccineId: String, name: String, speciesId: String[]): void
- + VaccinateAnimal(vaccineId: String, veterinarianId: String, AnimalId: String): void
- + ShowAllVaccinations(): void
- + ShowAnimalsInHabitat(habitatId: String): void
- + ShowMedicalActsOnAnimal(AnimalId: String): void
- + ShowMedicalActsByVeterinarian(veterinarianId: String): void
- + ShowWrongVaccinations(): void
- + AdvanceSeason(): void

Hotel

- + registerAnimal (animalId: String, name: String, speciesId: String, habitatId: String): void
- + showAllAnimals(): void
- + showSatisfactionOfAnimal(): void
- + TransferToHabitat (animalId: String, newHabitatId: String): void
- + showAllEmployees(): void
- + RegisterEmployee (employeeId: String, name: String, type: String): void
- + AddResponsibility (responsibilityId: String, employeeId: String): void
- + RemoveResponsibility (responsibilityId: String, employeeId: String): void
- + showSatisfactionOfEmployee (employeeId: String): void
- + showAllHabitats(): void
- + RegisterHabitat (habitatId: String, name: String, habitatArea: int): void
- + ChangeHabitatArea (habitatId: String, newArea: int): void
- + ChangeHabitatInfluence (habitatId: String, speciesId: String, habitatInfluence: int): void
- + AddTreeToHabitat (habitatId: String, treeId: String, name: String, treeAge: int, treeDifficulty: int, treeType: String): void
- + showAllTreesInHabitat(): void
- + showAllVaccines(): void
- + RegisterVaccine (vaccineId: String, name: String, speciesId: String[]): void
- + VaccinateAnimal (vaccineId: String, veterinarianId: String, animalId: String): void
- + showAllVaccinations(): void
- + showAllAnimalsInHabitat (HabitatId: String): void
- + showMedicalActsOnAnimal (animalId: String): void
- + showMedicalActsByVeterinarian (veterinarianId: String): void
- + showWrongVaccinations(): void
- + AdvanceSeason(): void

Animal

- _animalId : String
- _name : String
- _healthStatus : String[]

+ vaccinate (vaccine: Vaccine, vet: Veterinarian): void
+ transfer to Habitat (newHabitat: Habitat): void
+ get Satisfaction (S: SatisfactionStrategy): double
+ get HealthHistory(): String

Species

- _speciesId : String
- _name : String

+ addAnimal (animal: Animal): void
+ removeAnimal (animal: Animal): void
+ getAnimalById (animalId: String): Animal

Habitat

- _habitatId : String
- _name : String
- _area : int
- _speciesInInfluence : Map < Species, Influence >

+ addTree (tree: Tree): void
+ removeTree (tree: Tree): void
+ addAnimal (animal: Animal): void
+ removeAnimal (animal: Animal): void
+ setInfluence (species: Species, influence: String): void

Vaccine

- _vaccineId : String
- _name : String

+ isSuitableFor (speciesId: String): boolean
+ calculateDamage (animal: Animal): int

Vaccination Record

- _vaccine : Vaccine
- _animal : Animal
- _vet : Veterinarian

<<abstract>> Tree

- _treeId: String
- _name : String
- _age : int
- _baseCleaningDifficulty : int

+ calculateTotalCleaningEffort(): double

DeciduousTree

+ getTreeType(): String

EvergreenTree

+ getTreeType(): String

<<abstract>> Employee

- _employeeId : String
- _name : String

+ <<abstract>> get Satisfaction (...): double

Handler

+ get Satisfaction (s: SatisfactionStrategy): double

Veterinarian

+ get Satisfaction (S: SatisfactionStrategy): double

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

Pedro Nunes

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

Andre Borb

