# Részletes tervek

[A dokumentum célja, hogy pontosan specifikálja az implementálandó osztályokat, beleértve a privát attribútumokat és metódusokat, ezek definícióját is.

A dokumentum második fele részletesen be kell mutassa a korábban definiált be- és kimeneti nyelv szintakszisát felhasználva, hogy mely tesztekkel lesz a prototípus ellenőrizve.]

## Osztályok és metódusok tervei.

### Osztály1

#### Felelősség

[Mi az osztály felelőssége. Kb 1 bekezdés. Ha szükséges, akkor state-chart is.]

#### Ősosztályok

[Mely osztályokból származik (öröklési hierarchia)]

Legősebb osztály → Ősosztály2 → Ősosztály3...

#### Interfészek

[Mely interfészeket valósítja meg.]

#### Attribútumok

[Milyen attribútumai vannak, beleértve a nem publikusakat is.]

* **attribútum1**: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa
* **attribútum2**: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa

#### Metódusok

[Milyen publikus, protected és privát metódusokkal rendelkezik. Metódusonként precíz leírás, ha szükséges, activity diagram is a metódusban megvalósítandó algoritmusról. Minden olyan metódusnak szerepelnie kell, amelyiket az osztály megvalósít vagy felüldefiniál.]

* **int foo(Osztály3 o1, Osztály4 o2)**: metódus leírása, láthatósága (UML jelöléssel)
* **int bar(Osztály5 o1)**: metódus leírása, láthatósága (UML jelöléssel)

### Osztály2

#### Felelősség

[Mi az osztály felelőssége. Kb 1 bekezdés. Ha szükséges, akkor state-chart is.]

#### Ősosztályok

[Mely osztályokból származik (öröklési hierarchia)]

Legősebb osztály → Ősosztály2 → Ősosztály3...

#### Interfészek

[Mely interfészeket valósítja meg.]

#### Attribútumok

[Milyen attribútumai vannak, beleértve a nem publikusakat is.]

* **attribútum1**: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa
* **attribútum2**: attribútum jellemzése: mire való, láthatósága (UML jelöléssel), típusa

#### Metódusok

[Milyen publikus, protected és privát metódusokkal rendelkezik. Metódusonként precíz leírás, ha szükséges, activity diagram is a metódusban megvalósítandó algoritmusról. Minden olyan metódusnak szerepelnie kell, amelyiket az osztály megvalósít vagy felüldefiniál.]

* **int foo(Osztály3 o1, Osztály4 o2)**: metódus leírása, láthatósága (UML jelöléssel)
* **int bar(Osztály5 o1)**: metódus leírása, láthatósága (UML jelöléssel)

## A tesztek részletes tervei, leírásuk a teszt nyelvén

[A tesztek részletes tervei alatt meg kell adni azokat a bemeneti adatsorozatokat, amelyekkel a program működése ellenőrizhető. Minden bemenő adatsorozathoz definiálni kell, hogy az adatsorozat végrehajtásától a program mely részeinek, funkcióinak ellenőrzését várjuk és konkrétan milyen eredményekre számítunk, ezek az eredmények hogyan vethetők össze a bemenetekkel. A tesztek leírásakor az előző dokumentumban (proto koncepciója) megadott szintakszist kell használni.]

Kiadható parancsok:

**CREATE\_INSECT tecton name**

**Leírás: A rovar létrejön és rákerül az argumentumként megadott céltektonra, ha ezen van gombafonál**

**Opciók: Argumentumok: A céltekton, ahova létrejönne, az objektum neve**

**MOVE insect tecton**

**Leírás: A rovar átmenne az argumentumként megadott céltektonra**

**Opciók: Argumentumok: A rovar, amelyik mozogna;**

**A céltekton, ahova mozogna**

**EAT insect**

**Leírás: A rovar megeszik egy spórát a tektonján**

**Opciók: Argumentumok: A rovar, amelyik enne**

**CUT insect**

**Leírás: A rovar elvág egy fonalat a tektonján**

**Opciók: Argumentumok: A rovar, amelyik vágna**

### Rovar létrehozása és letevése

* **Leírás**

Létrehozunk egy rovarat egy tektonra. Beállitja a létrehozott rovar tektonját arra amelyikre létrehozták, és ennek az occupants listájára hozzáadja az újonnan létrehozott rovart.

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Új rovar létrehozása, tektonok beállitása*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]ű

// Arrange

CREATE\_TECTON FertileTecton {} FT1

SET\_BREAKTIMER FT1 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

// Act

CREATE\_INSECT FT1 I1

// Assert

STATE FT1

STATE M1

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1}

M1: Mycelium

location = FT1

I1: Insect

location = FT1

maxMoves = 2

remainingMoves = 2

sporesEaten = 0

effectTimer = 0

state = Normal

### Rovar mozgatása

* **Leírás**

A rovar mozog egy, a tektonjával szomszédos, tektonra

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Rovar tektonja megváltozott-e, és tükrözi-e ezt a mozgást a két tekton állapota*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]ű

// Arrange

CREATE\_TECTON FertileTecton {} FT1

CREATE\_TECTON FertileTecton {FT1} FT2

SET\_BREAKTIMER FT1 5

SET\_BREAKTIMER FT2 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

CREATE\_MYCELIUM Mycelium M2

ADD\_MYCELIUM\_TO\_TECTON M2 FT2

CREATE\_INSECT FT1 I1

// Act

MOVE I1 FT2

// Assert

STATE FT1

STATE M1

STATE FT2

STATE M2

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {FT2}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {}

M1: Mycelium

location = FT1

FT2: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {FT1}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M2}

occupants List<Insect> = {I1}

M2: Mycelium

location = FT2

I1: Insect

location = FT2

maxMoves = 2

remainingMoves = 1

sporesEaten = 0

effectTimer = 0

state = Normal

### Rovar sikertelen mozgatása nem-szomszédos tektonra

* **Leírás**

A rovar mozogna egy, a tektonjával nem szomszédos, tektonra, ami nem sikerül

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Rovar tektonja megmaradt-e a régi, és tükrözi-e ezt a mozgást a két tekton állapota*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]ű

// Arrange

CREATE\_TECTON FertileTecton {} FT1

CREATE\_TECTON FertileTecton {} FT2

SET\_BREAKTIMER FT1 5

SET\_BREAKTIMER FT2 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

CREATE\_MYCELIUM Mycelium M2

ADD\_MYCELIUM\_TO\_TECTON M2 FT2

CREATE\_INSECT FT1 I1

// Act

MOVE I1 FT2

// Assert

STATE FT1

STATE M1

STATE FT2

STATE M2

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1}

M1: Mycelium

location = FT1

FT2: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M2}

occupants List<Insect> = {}

M2: Mycelium

location = FT2

I1: Insect

location = FT1

maxMoves = 2

remainingMoves = 2

sporesEaten = 0

effectTimer = 0

state = Normal

### Rovar sikertelen mozgatása olyan tektonra, amelyen nincs gombafonál

* **Leírás**

A rovar mozog egy, a tektonjával szomszédos, tektonra, de nem sikerül, mert nincs rajta gombafonál

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Rovar tektonja megváltozott-e, és tükrözi-e ezt a mozgást a két tekton állapota*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]ű

// Arrange

CREATE\_TECTON FertileTecton {} FT1

CREATE\_TECTON FertileTecton {FT1} FT22

SET\_BREAKTIMER FT1 5

SET\_BREAKTIMER FT2 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

CREATE\_INSECT FT1 I1

// Act

MOVE I1 FT2

// Assert

STATE FT1

STATE M1

STATE FT2

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {FT2}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1}

M1: Mycelium

location = FT1

FT2: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {FT1}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {}

occupants List<Insect> = {I1}

I1: Insect

location = FT1

maxMoves = 2

remainingMoves = 1

sporesEaten = 0

effectTimer = 0

state = Normal

### Rovar általi spóraevés következtében kettészakadás

* **Leírás**

A rovar megeszik egy spórát a tektonján, aminek a hatására kettészakad

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Spóraevést követően a tektonról eltűnik-e a spóra, illetve ennek hatása beállitódik-e a rovaron*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]

// Arrange

CREATE\_TECTON FertileTecton {} FT1

SET\_BREAKTIMER FT1 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

PUT\_SPORE SplitSpore FT1

CREATE\_INSECT FT1 I1

// Act

EAT I1

// Assert

STATE FT1

STATE M1

STATE I1

STATE I2

//I2-vel van jelölve az újonnan létrejött rovar

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1, I2}

M1: Mycelium

location = FT1

I1: Insect

location = FT1

maxMoves = 2

remainingMoves = 1

sporesEaten = 1

effectTimer = 0

state = Normal

I2: Insect

location = FT1

maxMoves = 2

remainingMoves = 1

sporesEaten = 0

effectTimer = 0

state = Normal

### Rovar általi spóraevés következtében Slow állapotba kerülés

* **Leírás**

A rovar megeszik egy spórát a tektonján, aminek a hatására Slow állapotba kerül

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Spóraevést követően a tektonról eltűnik-e a spóra, illetve ennek hatása beállitódik-e a rovaron*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]

// Arrange

CREATE\_TECTON FertileTecton {} FT1

SET\_BREAKTIMER FT1 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

PUT\_SPORE SlownessSpore FT1

CREATE\_INSECT FT1 I1

// Act

EAT I1

// Assert

STATE FT1

STATE M1

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1}

M1: Mycelium

location = FT1

I1: Insect

location = FT1

maxMoves = 1

remainingMoves = 1

sporesEaten = 1

effectTimer = ?

state = Slow

### Rovar általi spóraevés következtében Fast állapotba kerülés

* **Leírás**

A rovar megeszik egy spórát a tektonján, aminek a hatására Fast állapotba kerül

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Spóraevést követően a tektonról eltűnik-e a spóra, illetve ennek hatása beállitódik-e a rovaron*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]

// Arrange

CREATE\_TECTON FertileTecton {} FT1

SET\_BREAKTIMER FT1 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

PUT\_SPORE SpeedSpore FT1

CREATE\_INSECT FT1 I1

// Act

EAT I1

// Assert

STATE FT1

STATE M1

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1}

M1: Mycelium

location = FT1

I1: Insect

location = FT1

maxMoves = 3

remainingMoves = 1

sporesEaten = 1

effectTimer = ?

state = Fast

### Rovar általi spóraevés következtében PreventCut állapotba kerülés

* **Leírás**

A rovar megeszik egy spórát a tektonján, aminek a hatására PreventCut állapotba kerül

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Spóraevést követően a tektonról eltűnik-e a spóra, illetve ennek hatása beállitódik-e a rovaron*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]

// Arrange

CREATE\_TECTON FertileTecton {} FT1

SET\_BREAKTIMER FT1 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

PUT\_SPORE PreventCutSpore FT1

CREATE\_INSECT FT1 I1

// Act

EAT I1

// Assert

STATE FT1

STATE M1

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1}

M1: Mycelium

location = FT1

I1: Insect

location = FT1

maxMoves = 2

remainingMoves = 1

sporesEaten = 1

effectTimer = ?

state = CannotCut

### Rovar általi spóraevés következtében Stunned állapotba kerülés

* **Leírás**

A rovar megeszik egy spórát a tektonján, aminek a hatására PreventCut állapotba kerül

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Spóraevést követően a tektonról eltűnik-e a spóra, illetve ennek hatása beállitódik-e a rovaron*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]

// Arrange

CREATE\_TECTON FertileTecton {} FT1

SET\_BREAKTIMER FT1 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

PUT\_SPORE StunSpore FT1

CREATE\_INSECT FT1 I1

// Act

EAT I1

// Assert

STATE FT1

STATE M1

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1}

M1: Mycelium

location = FT1

I1: Insect

location = FT1

maxMoves = 0

remainingMoves = 1

sporesEaten = 1

effectTimer = ?

state = Stun

### Rovar általi spóraevés következtében PreventCut állapotba kerülés

* **Leírás**

A rovar megeszik egy spórát a tektonján, aminek a hatására PreventCut állapotba kerül

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Spóraevést követően a tektonról eltűnik-e a spóra, illetve ennek hatása beállitódik-e a rovaron*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]

// Arrange

CREATE\_TECTON FertileTecton {} FT1

SET\_BREAKTIMER FT1 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

CREATE\_INSECT FT1 I1

// Act

EAT I1

// Assert

STATE FT1

STATE M1

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {I1}

M1: Mycelium

location = FT1

I1: Insect

location = FT1

maxMoves = 2

remainingMoves = 2

sporesEaten = 0

effectTimer = 0

state = Normal

### Rovar általi gombafonál elvágás

* **Leírás**

A rovar mozog egy, a tektonjával szomszédos, tektonra

* **Ellenőrzött funkcionalitás, várható hibahelyek**

*Rovar tektonja megváltozott-e, és tükrözi-e ezt a mozgást a két tekton állapota*

* **Bemenet**

[a proto bemeneti nyelvén megadva (lásd előző anyag)]ű

// Arrange

CREATE\_TECTON FertileTecton {} FT1

CREATE\_TECTON FertileTecton {FT1} FT2

SET\_BREAKTIMER FT1 5

SET\_BREAKTIMER FT2 5

CREATE\_MYCELIUM Mycelium M1

ADD\_MYCELIUM\_TO\_TECTON M1 FT1

CREATE\_MUSHROOM\_BODY MB1 FT2

CREATE\_INSECT FT1 I1

// Act

CUT I1

// Assert

STATE FT1

STATE M1

STATE FT2

STATE MB1

STATE I1

* **Elvárt kimenet**

[a proto kimeneti nyelvén megadva (lásd előző anyag)]

FT1: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {FT2}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = null

mycelia Queue<Mycelium> = {M1}

occupants List<Insect> = {}

M1: Mycelium

location = FT1

deathTimer = ?

FT2: FertileTecton

breakTimer int = 5

neighbours List<Tecton> = {FT1}

myceliumCapacity int = 1

spores Queue<Spore> = {}

mushroomBody MushroomBody = MB1

mycelia Queue<Mycelium> = {}

occupants List<Insect> = {I1}

MB1: MushroomBody

location = FT2

remainingEjects = 3

I1: Insect

location = FT2

maxMoves = 2

remainingMoves = 1

sporesEaten = 0

effectTimer = 0

state = Normal

## A tesztelést támogató programok tervei

[A tesztadatok előállítására, a tesztek eredményeinek kiértékelésére szolgáló segédprogramok részletes terveit kell elkészíteni.]

## Napló

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2010.03.21. 18:00 | 2,5 óra | Horváth  Németh  Tóth  Oláh | Értekezlet.  Döntés: Horváth elkészíti az osztálydiagramot, Oláh a use-case leírásokat. |
| 2010.03.23. 23:00 | 5 óra | Németh | Tevékenység: Németh implementálja a tesztelő programokat. |
| … | … | … | … |