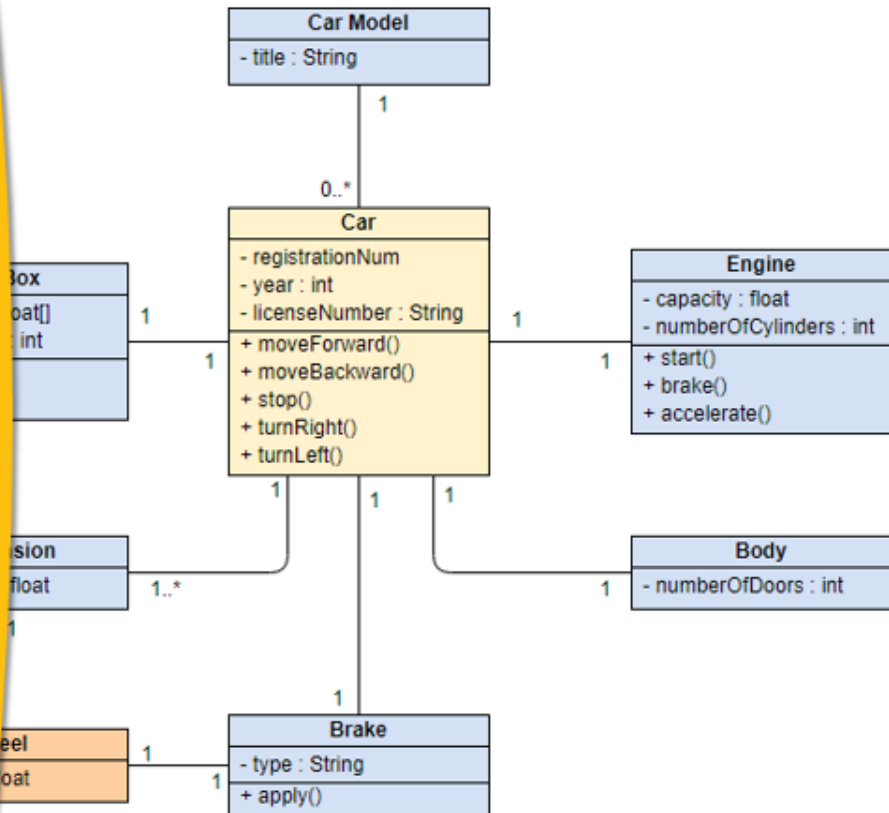


What is Class Diagram?



Wat is een class diagram?

Unified Modelling Language

"Ontwerpen volgens standaarden en regels"

UML Diagram Type

Structural Diagrams

Composite Structure Diagram

Deployment Diagram

Package Diagram

Profile Diagram

Class Diagram

Object Diagram

Component Diagram

Behavioral Diagrams

Activity Diagram

Use Case Diagram

State Machine Diagram

Interaction Diagram

Sequence Diagram

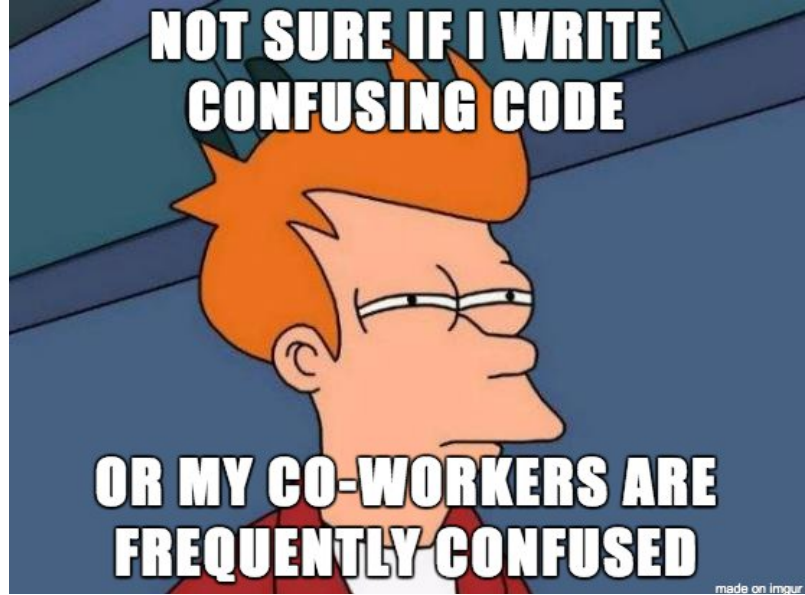
Communication Diagram

Interaction Overview Diagram

Timing Diagram

Nut van een Class Diagram?

1. Code structuur van te voren ontwerpen
2. Code structuur achteraf in kaart brengen (evt. optimalisatie)
3. Overzicht bieden over structuur aan andere devs



Een UML class diagram bestaat uit:

1. Een verzameling classes

a. **Properties** *variabelen*

b. **Operations** *functies*

2. De relaties tussen de classes

a. **Association** *elke relatie*

b. **Generalisation** *overerving (".. IS A ..")*

c. **Composition** *compositie (".. HAS A ..")*

d. **Dependancy** *functioneert niet zonder...*

Tower

<Properties:>

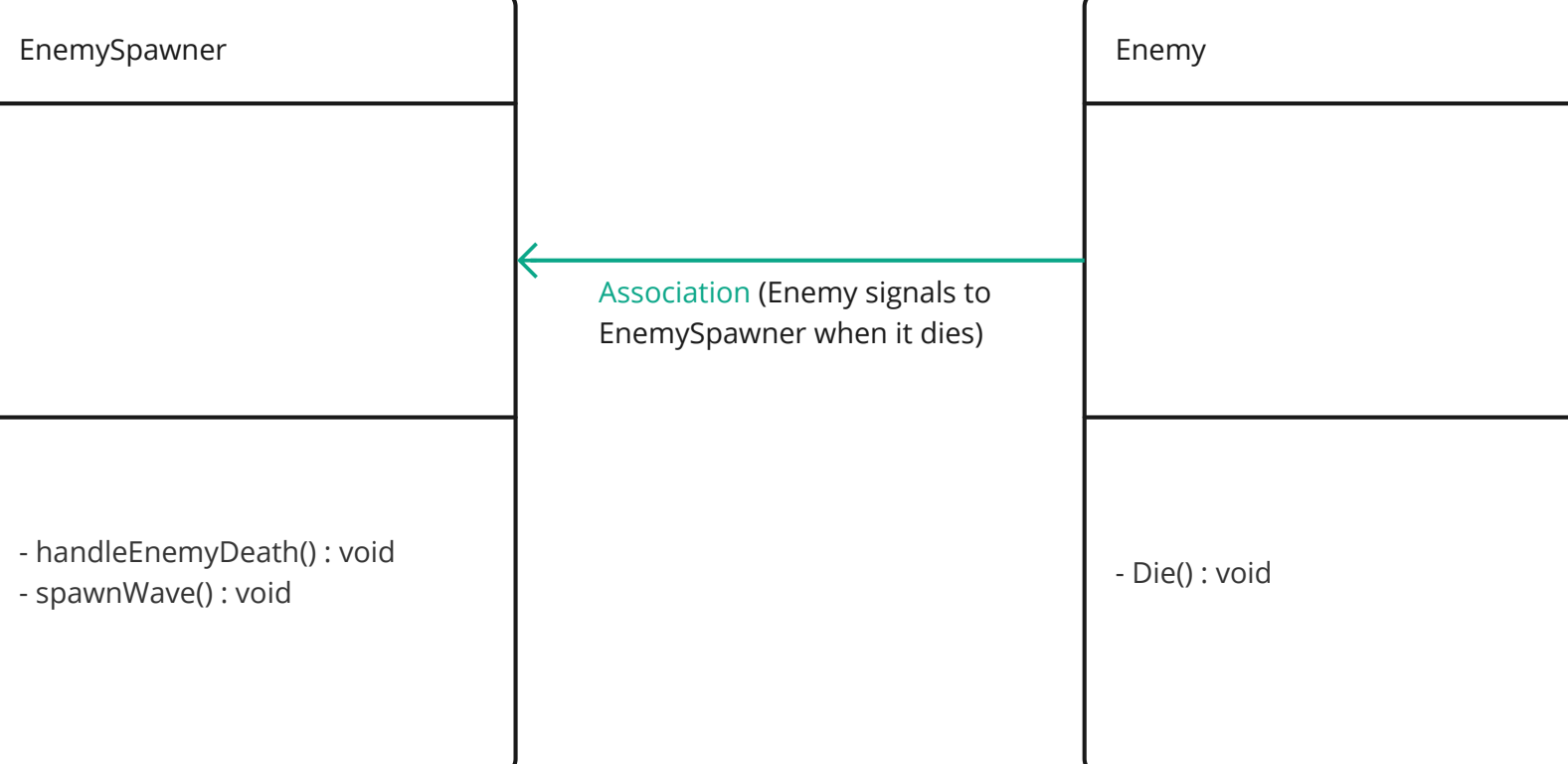
- cost : int *private*
- attack : int
- + bullet : GameObject *public*
- + reloadTime : int
- + turret:Turret



+ reloadTime: int
+ turret:Turret

<Operations:>

- Reload(): void
+ Shoot(target:GameObject): void



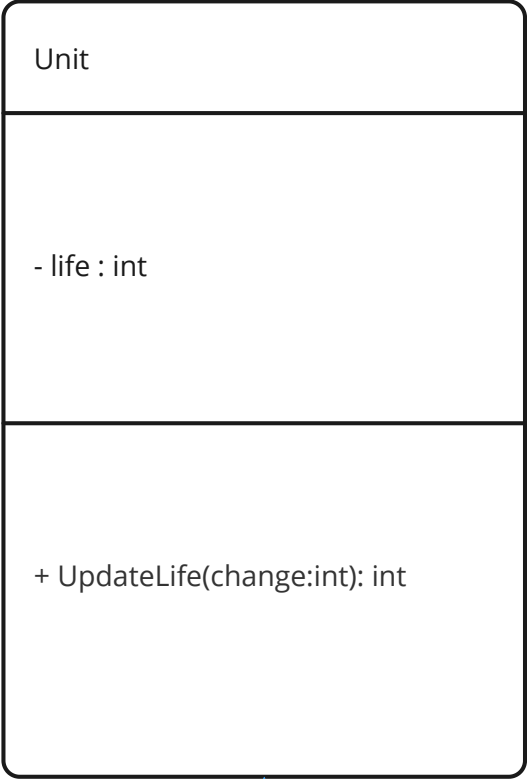
EnemySpawner

Enemy

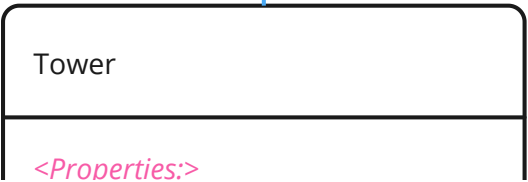
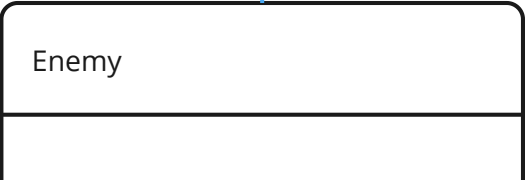
- handleEnemyDeath() : void
- spawnWave() : void

- Die() : void

Association (Enemy signals to
EnemySpawner when it dies)



Generalization (Tower/Enemy inherit form Unit)
"A Tower/Enemy **IS A** Unit"



Tower

<Properties:>

- cost : int *private*
- attack : int
- + bullet : GameObject *public*
- + reloadTime : int
- + turret: **Turret**

<Operations:>

- Reload(): void
- + Shoot(target:GameObject): void

Composition (Turret is part of a Tower)
"a Tower **HAS A** Turret"

Turret

- turnSpeed : int

- FollowTarget(target:GameObject) : void

