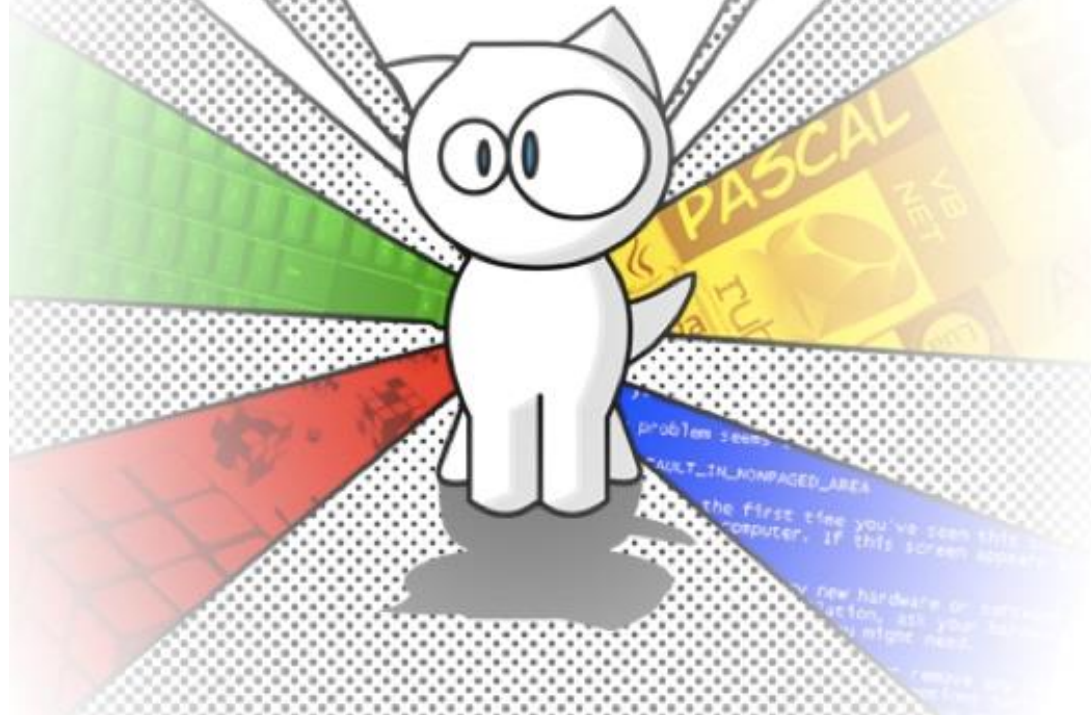


COD'INSA

Edition 2015



Le concours d'informatique

du réseau d'écoles INSA

CODING IN SA La finale

Bienvenue à tous !



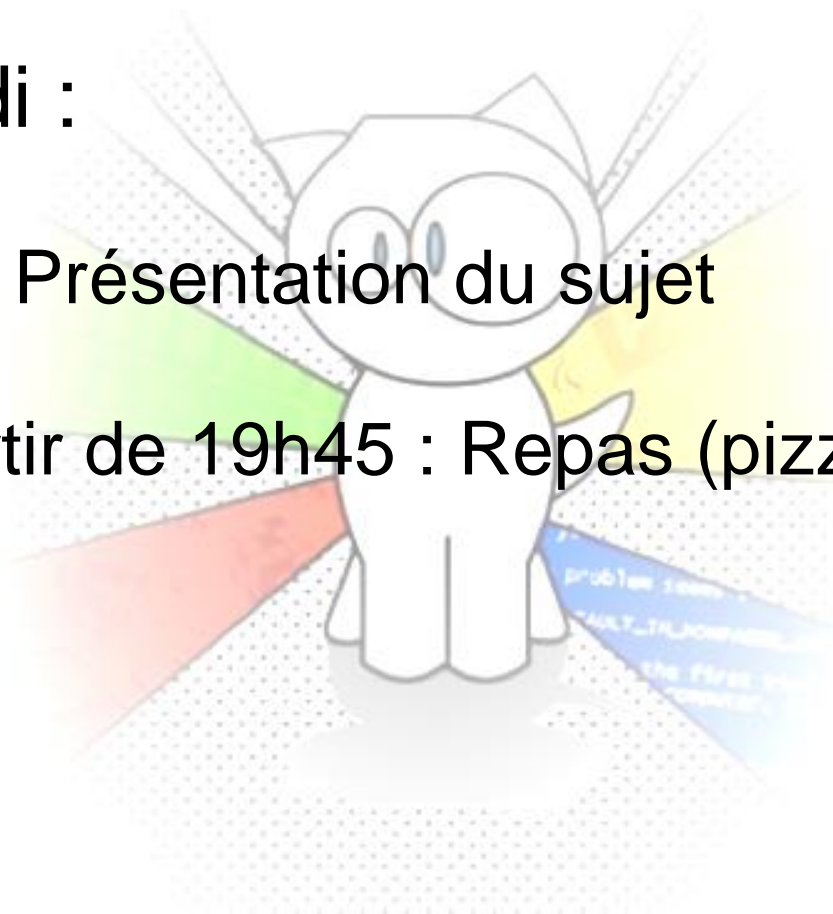
COD'INSA

Le mot du président



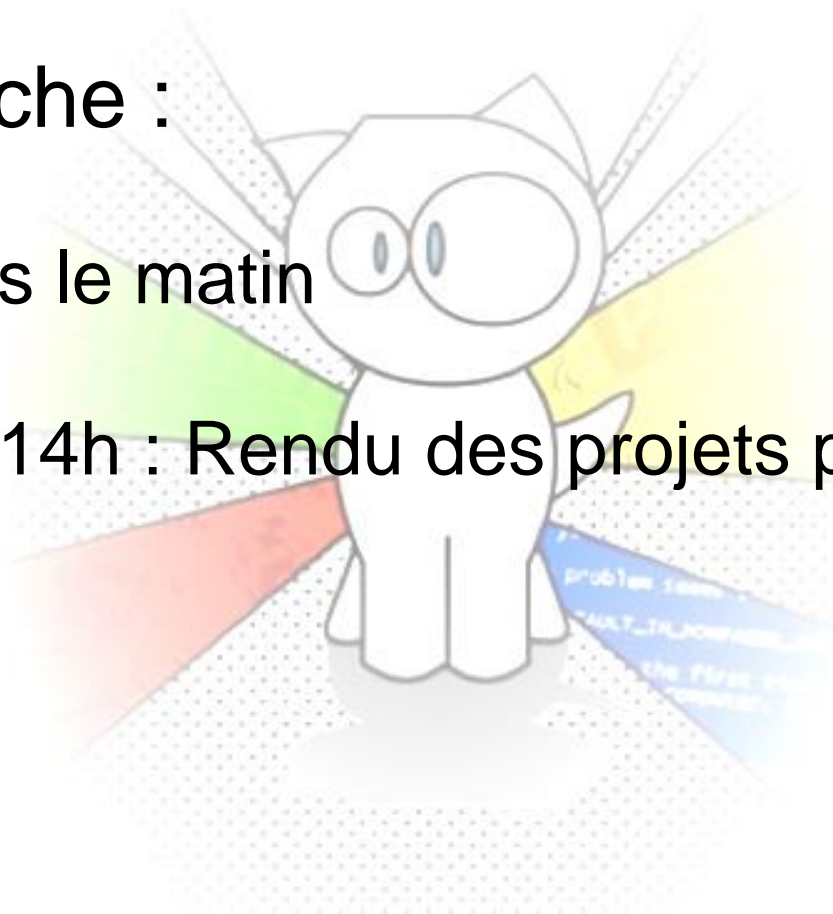
Organisation du week-end

- Samedi :
 - 15h : Présentation du sujet
 - A partir de 19h45 : Repas (pizza !)



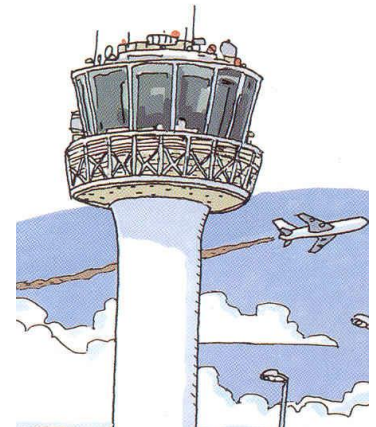
Organisation du week-end

- Dimanche :
 - Encas le matin
 - Vers 14h : Rendu des projets puis repas



Présentation du sujet

- Simulation du trafic aérien
- Objectif: diriger les appareils volant (avions, planeurs, hélicoptères) en fonction de la piste d'atterrissage choisie, des vents, etc...
- Conditions de victoire:
 - Ramener à bon port les avions de sa compagnie
 - Détruire un maximum d'appareils ennemis (armes, sacrifices, etc...)



Organisation générale de l'IA

- Utilisation d'une classe principale « IA »
- Implémentation d'une interface « Init »
 - Lancée en début de partie, permet d'initialiser les états internes de l'IA
- Implémentation d'une interface « Action »
 - Appelée à chaque itération, permet de définir les actions de toutes les unités
- Implémentation d'une interface « End »
 - Permettant d'exécuter les opérations de nettoyage de l'AI

Modèle aérodynamique utilisé

- Exploitation des équations d'Euler (écoulement de fluides à faible viscosité), de l'équation d'état des gaz parfait et de la formule de Bernoulli (incidence de l'altitude sur le champ de pression)
- Effort de trainée sur l'ensemble de la structure:
 - $F = 0.5 \times \rho \times V^2 \times S \times C$
 - ρ = masse volumique de l'air (fonction de l'altitude)
 - S = surface de référence (dépendant de la taille de l'appareil, mais aussi de sa forme aérodynamique)
 - C = coefficient aérodynamique
 - V = vitesse de déplacement

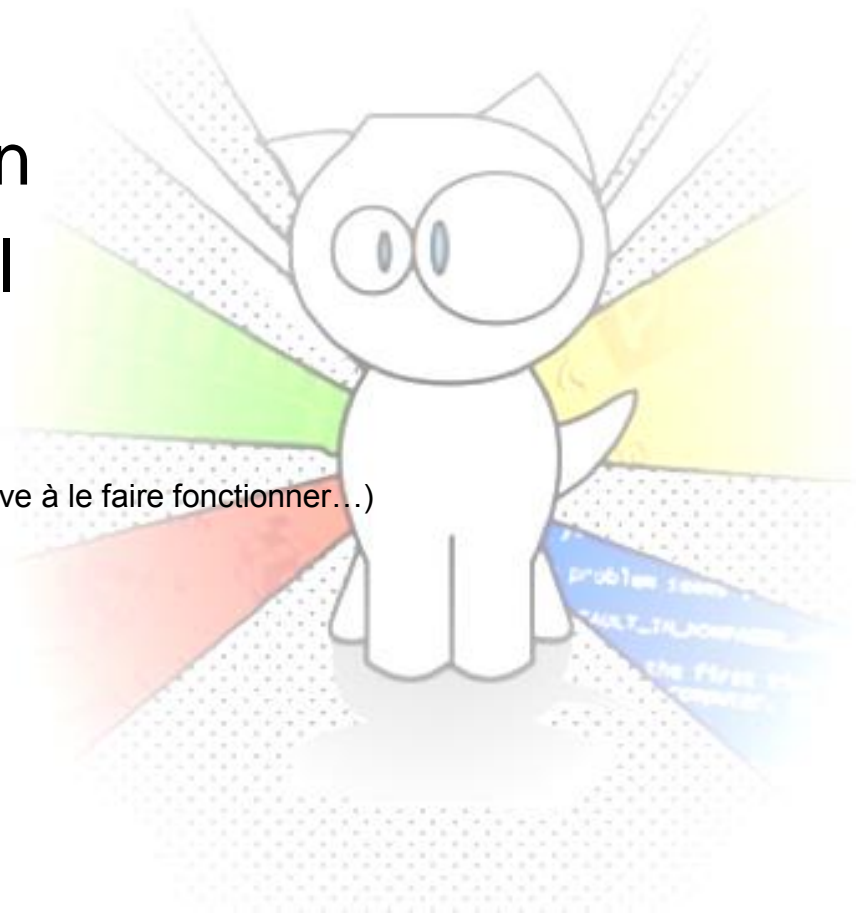
Langages autorisés

- Ada
- C
- Pascal
- CAML
- Java (si on arrive à le faire fonctionner...)



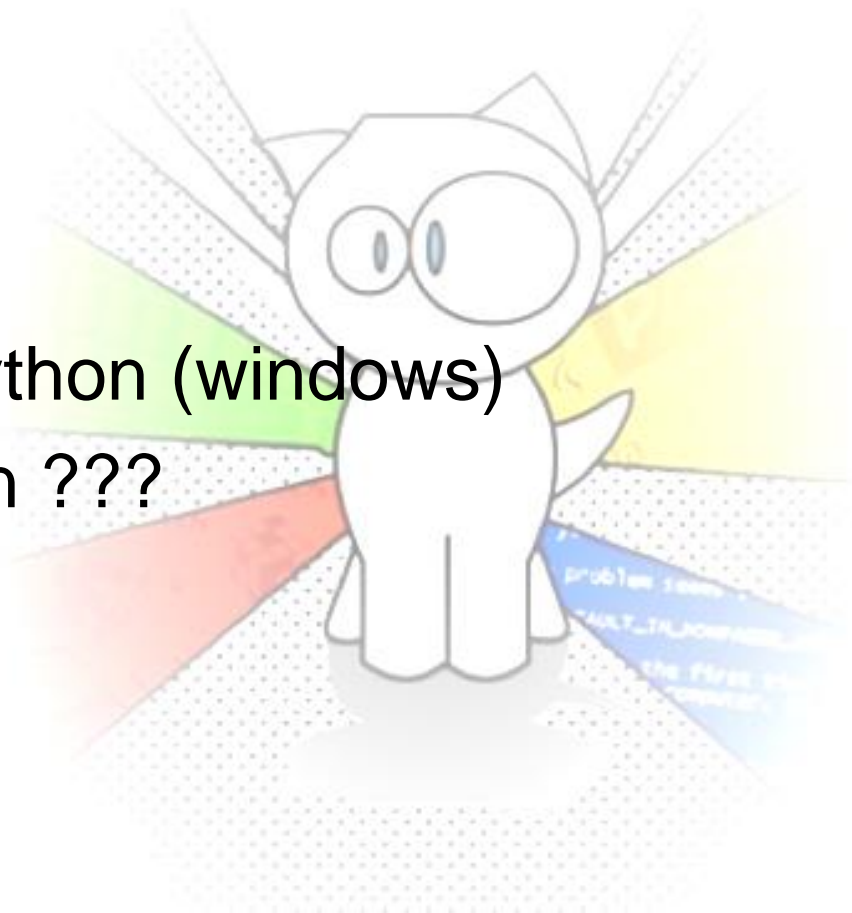
Bon, et plus sérieusement ?

- Ada
- Fortran
- Pascal
- CAML
- C (si on arrive à le faire fonctionner...)



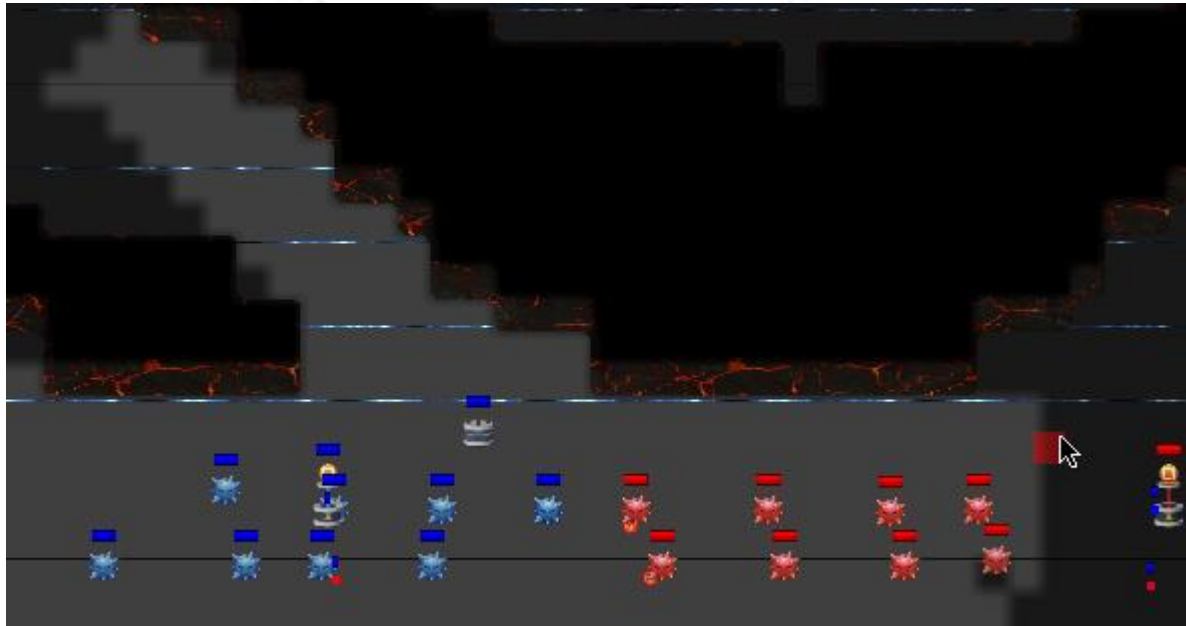
Langages autorisés (les vrais)

- C++
- Java
- C#
- IronPython (windows)
- Python ???



COD'INSA

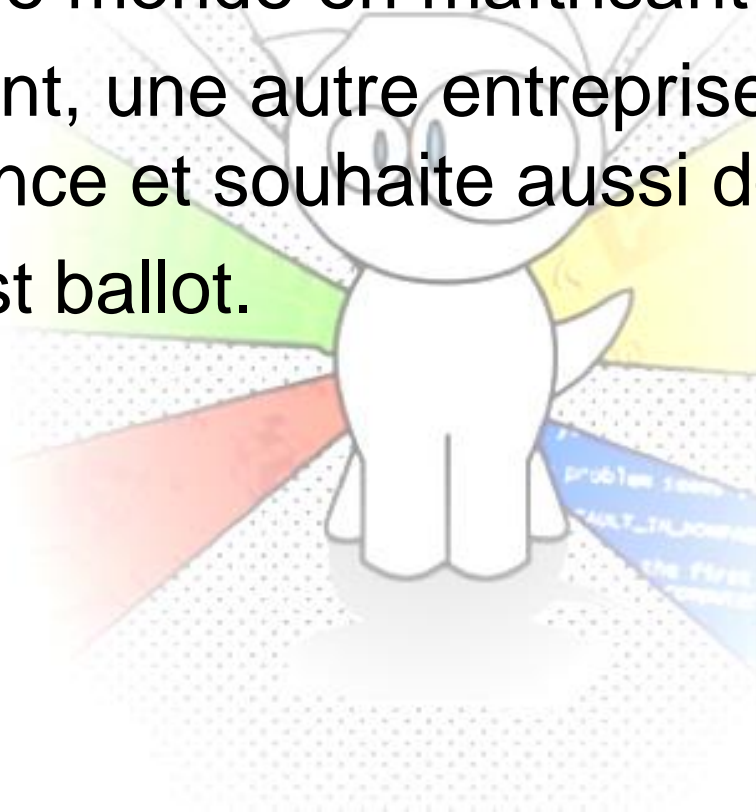
Le jeu



COD'INSA

Contexte

Vous êtes une **grande entreprise** qui souhaite dominer le monde en maîtrisant l'information. Cependant, une autre entreprise vous fait concurrence et souhaite aussi dominer le monde. Ce qui est ballot.



COD'INSA

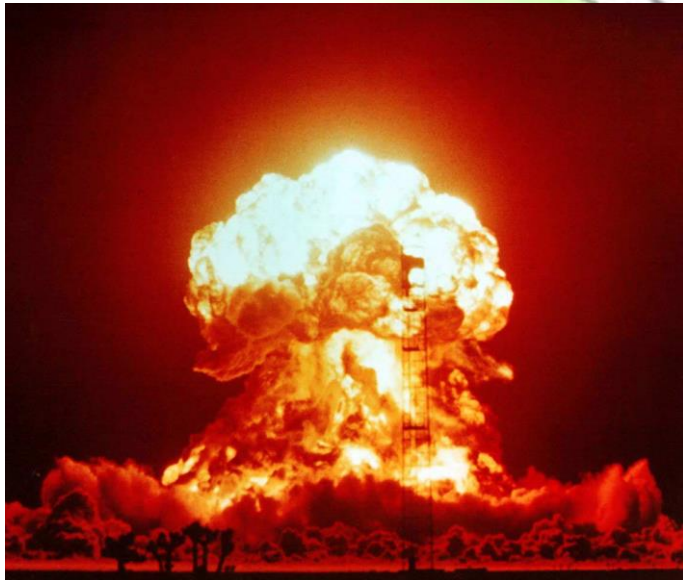
Contexte

Vous devez donc par tous les moyens (vraiment tous) détruire le Datacenter de l'entreprise concurrente.



Contexte

Vous allez donc déployer une IA sur le réseau mondial qui va se servir de virus, d'attaques DDOS, s'inculper de corruption de routeur etc...
Pour faire EXLOSER le Datacenter ennemi !



(← feat. Michael "FUCKING" Bay)

COD'INSA

MOBA

2 phases de jeu :

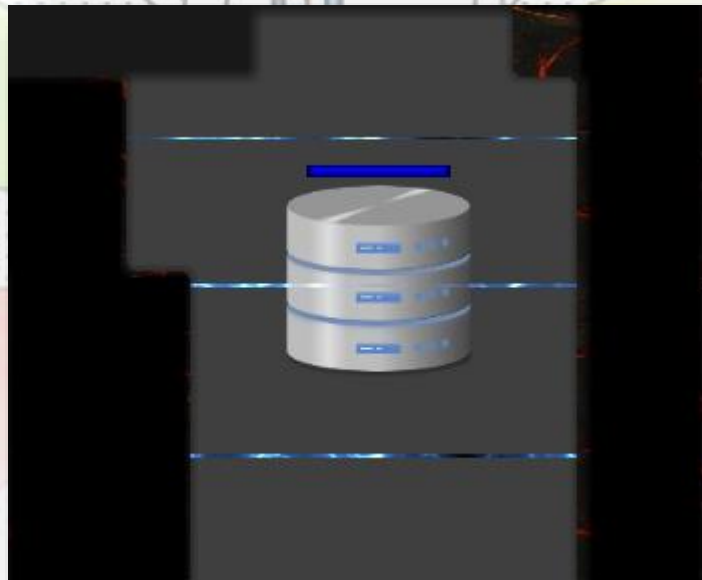
☐ Picks

☐ Jeu !!!



Phase de jeu

- ❑ 2 équipes de 3 IA
- ❑ But : détruire le datacenter ennemi !



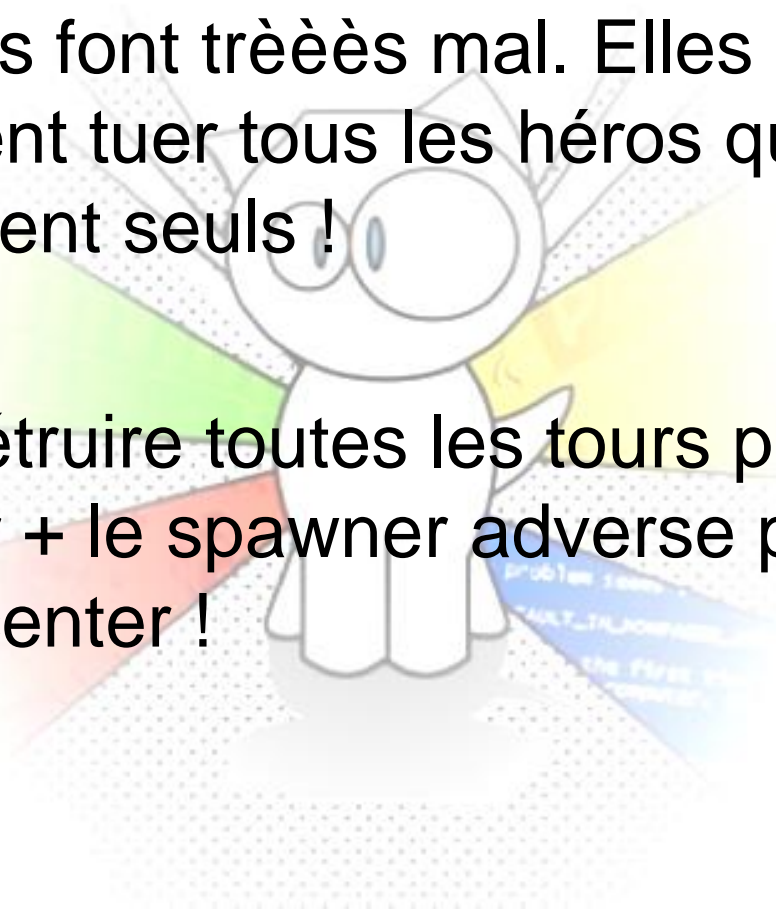
Comment ?

- ❑ Chaque IA contrôle 1 héros.
- ❑ Les héros ont des armes, sorts, équipements.
- ❑ Objectif : détruire les tours adverses une par une pour atteindre le Datacenter !



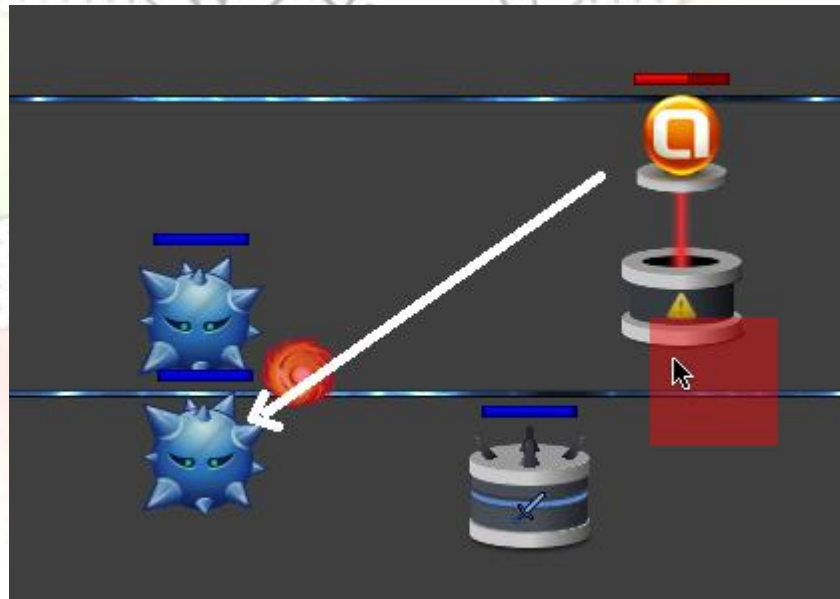
C'est trop simple non ?

- ❑ Les tours font trèèèè mal. Elles peuvent facilement tuer tous les héros qui s'en approchent seuls !
- ❑ Il faut détruire toutes les tours pour pouvoir attaquer + le spawner adverse pour attaquer le Datacenter !



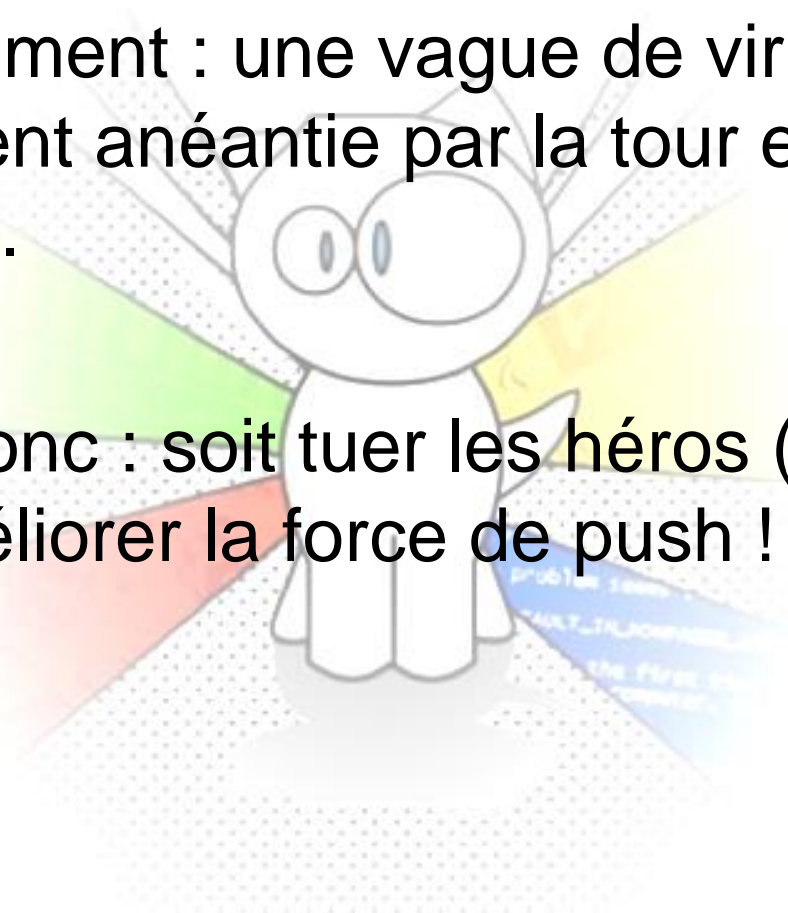
Comment qu'on fait alors ?

- ❑ Les Virus sont attaqués en premier par les tours !



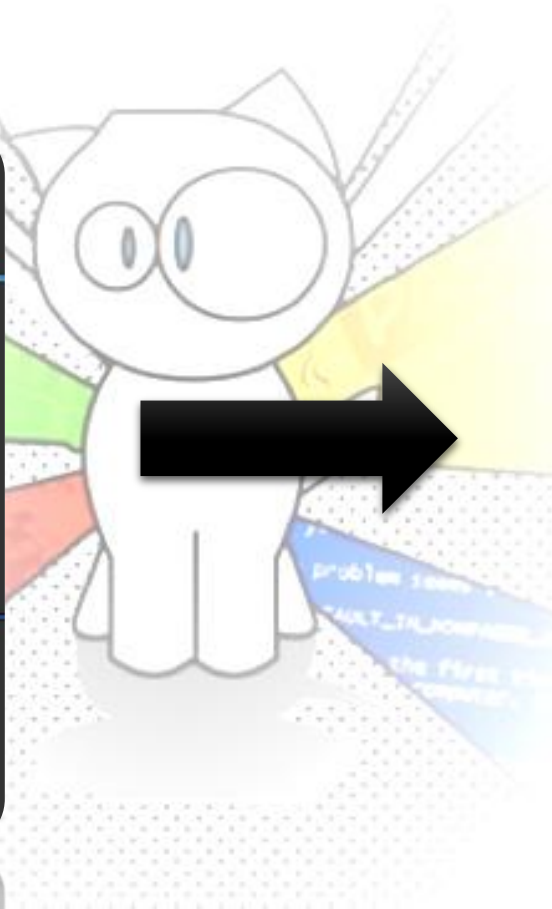
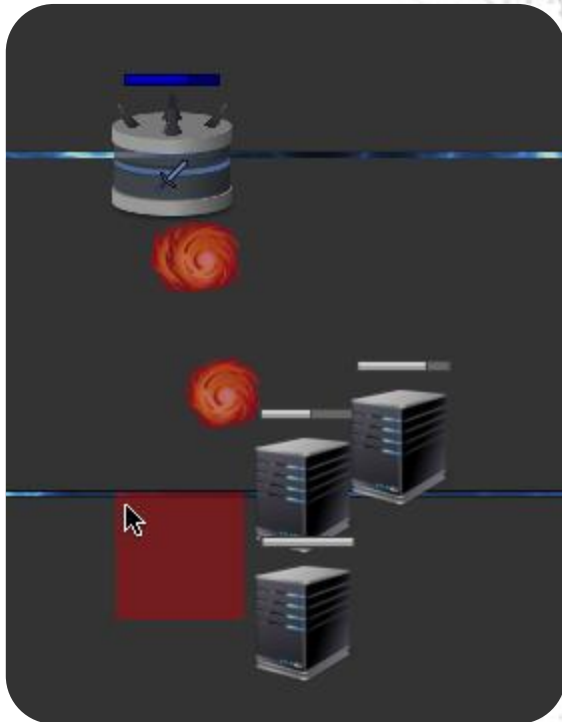
C'est quand même facile.

- ❑ Pas vraiment : une vague de virus peut être facilement anéantie par la tour et les héros adverse.
- ❑ Il faut donc : soit tuer les héros (combats !!) soit améliorer la force de push !



Améliorer sa force de push

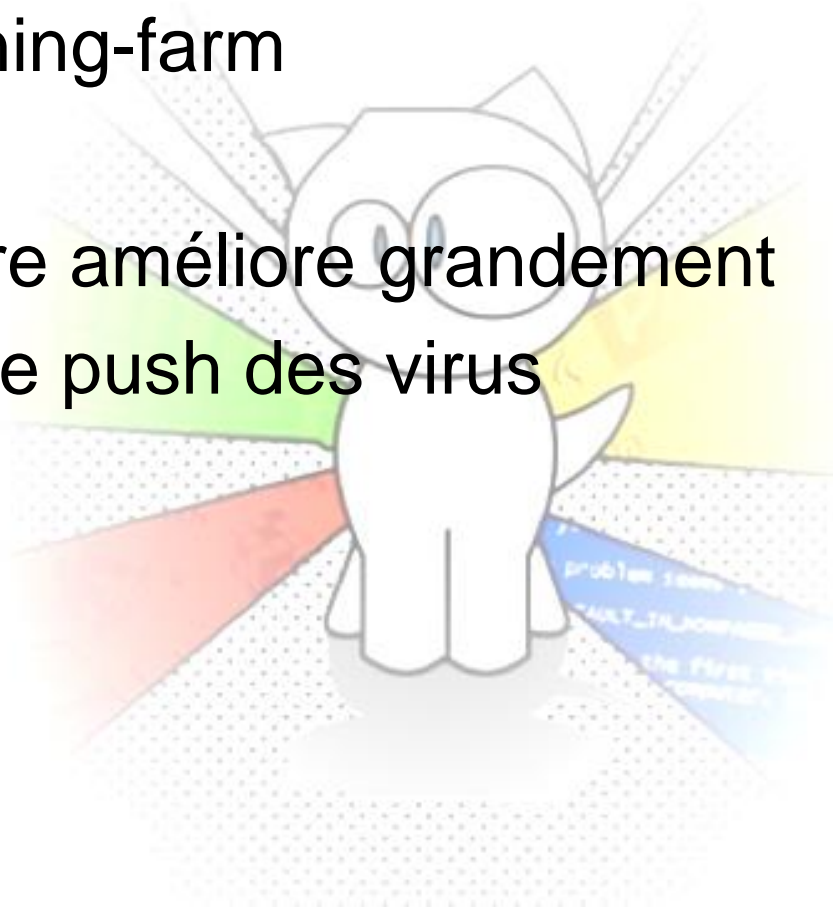
❑ Les camps



Améliorer sa force de push

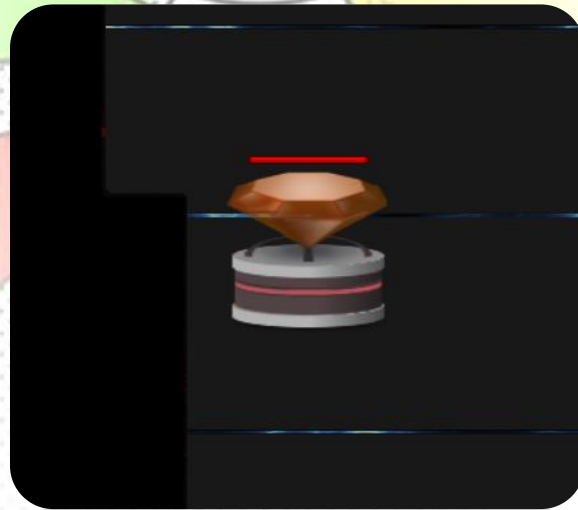
❑ La mining-farm

La détruire améliore grandement la force de push des virus



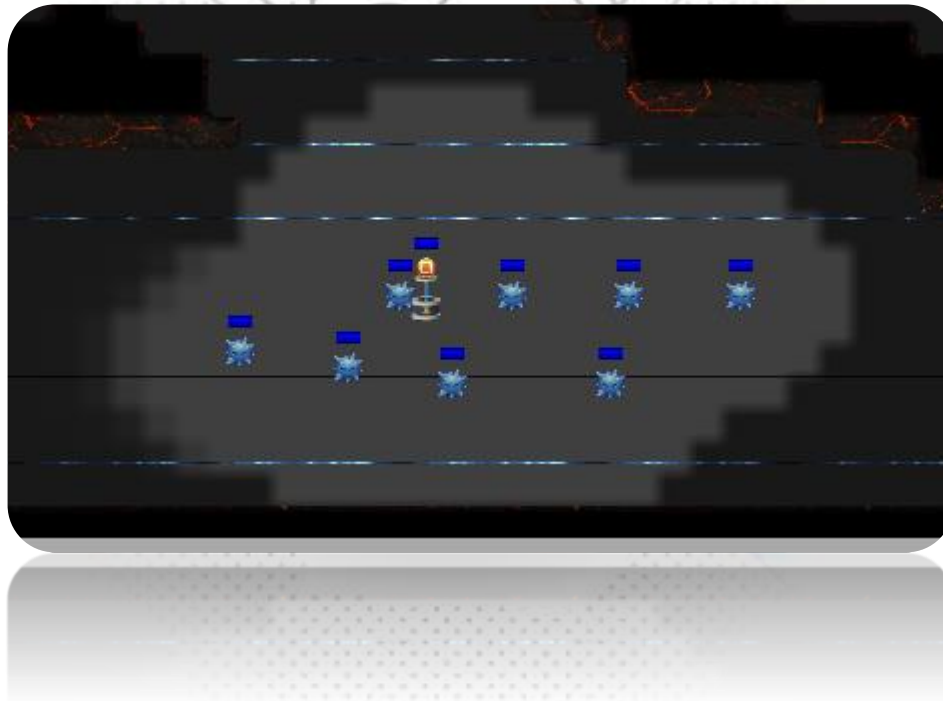
Annihiler celle de l'équipe ennemie

- ❑ Détruire le spawner: supprime les sbires ennemis pendant une période de temps donnée.



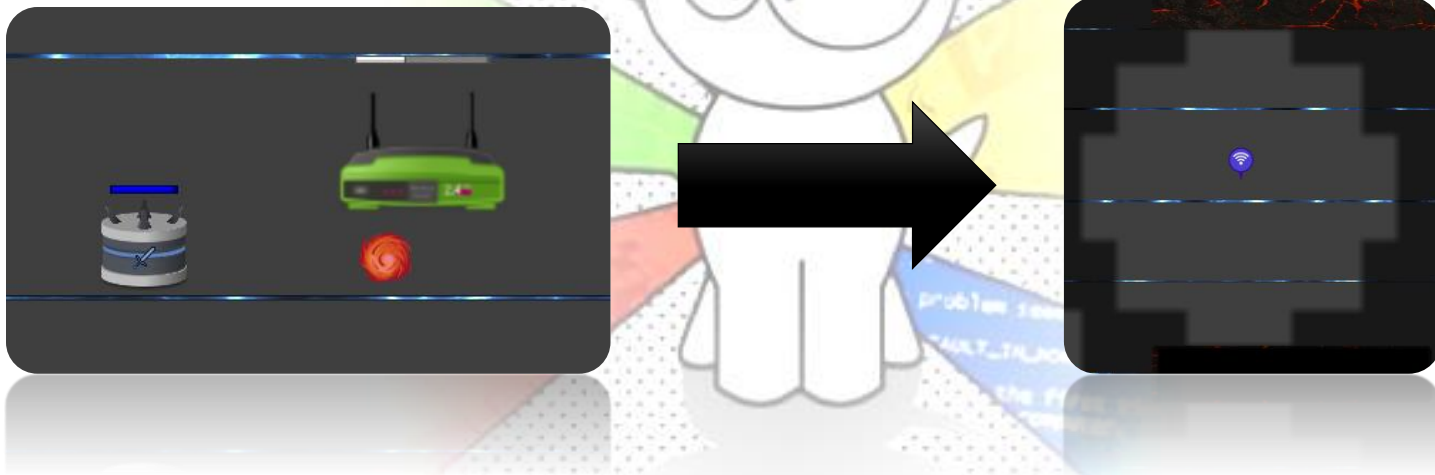
La vision

- ❑ Très important : sans celle, on ne sait pas ce que fait l'ennemi !



Obtenir + de vision

- ❑ Des « routeurs » sont positionnés à des endroits stratégiques.

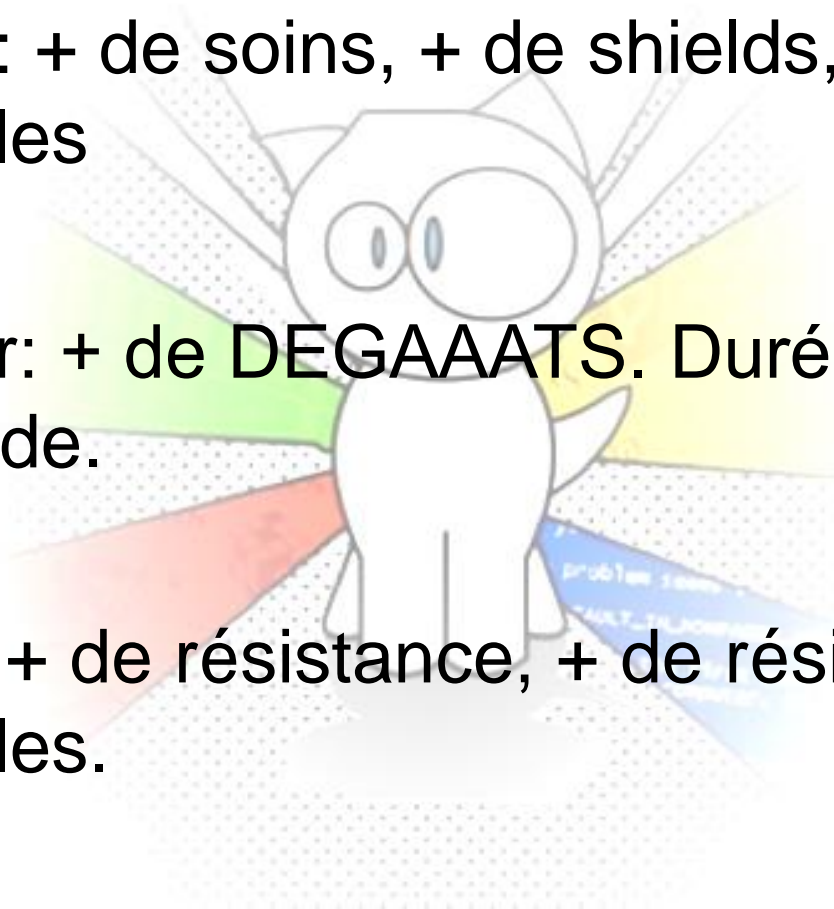


Les entités - caractéristiques

- ☐ HP / HP Max
- ☐ Armure / Résistance magique
- ☐ Dégâts d'attaque / Pouvoir
- ☐ Vitesse de déplacement
- ☐ Vitesse d'attaque
- ☐ Réduction de délais de récupération (CDR)
- ☐ Régénération de HP

Les héros - Roles

- ❑ Mage : + de soins, + de shields, + de contrôles
- ❑ Fighter: + de DEGAATS. Durée d'invisibilité + grande.
- ❑ Tank : + de résistance, + de résistance aux contrôles.



Les altérations d'états

StateAlterationModelView

- ❑ Elles décrivent les effets des sorts, et des attaques.
- ❑ Caractérisées par un type (Dégâts, bonus de caractéristiques etc... (cf la doc))
- ❑ Une durée (sauf pour les dégâts => durée = 0)
- ❑ Une valeur : une composante « flat », et des composantes de « scaling ».

Comprendre les armes

WeaponModelView

- ❑ Les armes sont améliorables plusieurs fois.
- ❑ Le modèle encapsule les modèles des upgrades

`List<WeaponUpgradeModelView>` Upgrades

WeaponUpgradeModelView

- ❑ Le modèle d'attaque de l'arme est identique à celui d'un sort :

`SpellLevelDescriptionView` Description

- ❑ Les armes ont en plus une liste d'altérations appliquées passivement au porteur :

`List<StateAlterationModelView>` PassiveAlterations

- ❑ Elles ont un prix aussi !

Comprendre les sorts

SpellModelView

- ❑ Les sorts sont améliorables plusieurs fois.
- ❑ Ils sont identifiés par un ID unique !
- ❑ Ils encapsulent leur description pour chaque niveau dans une liste :

`List<SpellLevelDescriptionView>` Levels

Comprendre les sorts

SpellLevelDescriptionView

- ❑ Un cooldown de base

`float` BaseCooldown

- ❑ Une liste d'altérations appliquées au lanceur du sort

`List<StateAlterationModelView>` CastingTimeAlterations;

- ❑ Des effets à l'impact sur la cible.

`List<StateAlterationModelView>` OnHitEffects;

Comprendre la map

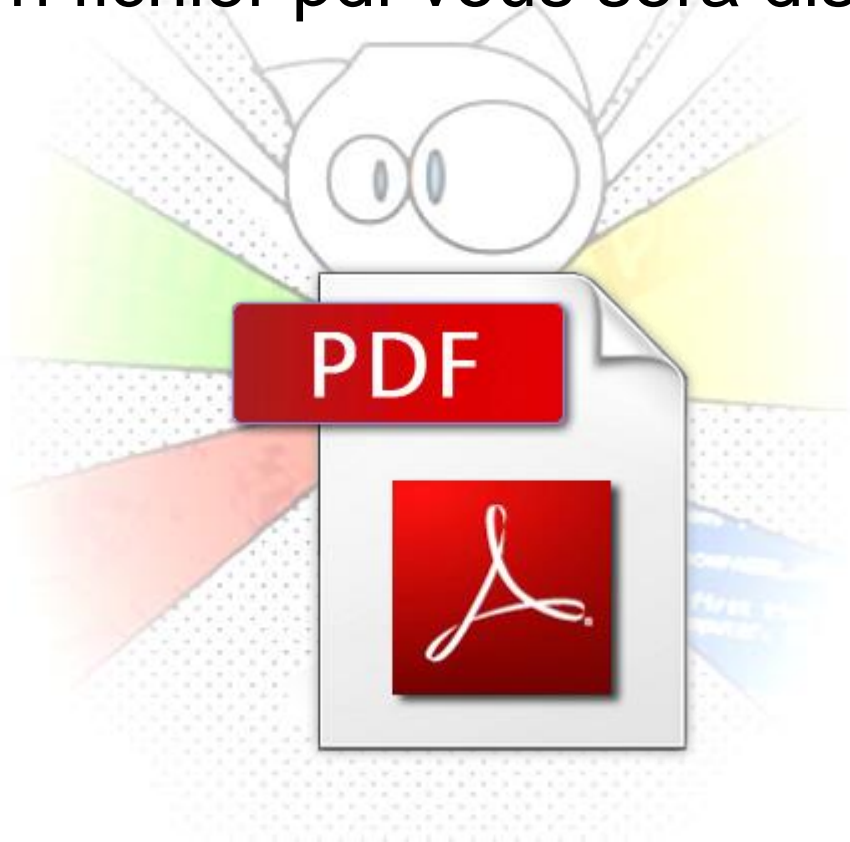
MapView

- ❑ Un matrice 2D indiquant la passibilité des cases
- ❑ `Passability[x][y]`
 - ❑ `true` : cette case ne contient pas de mur : on peut passer.
 - ❑ `false` : cette case contient un mur,

`List<List<bool>> Passability`

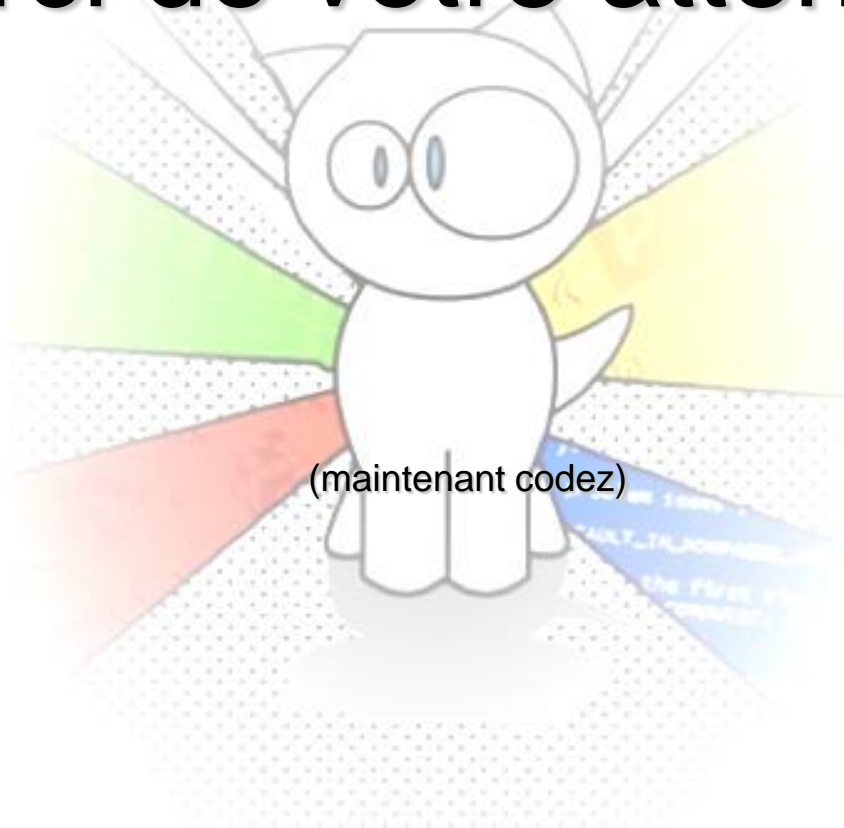
Le détail de l'API

Un fichier pdf vous sera distribué.

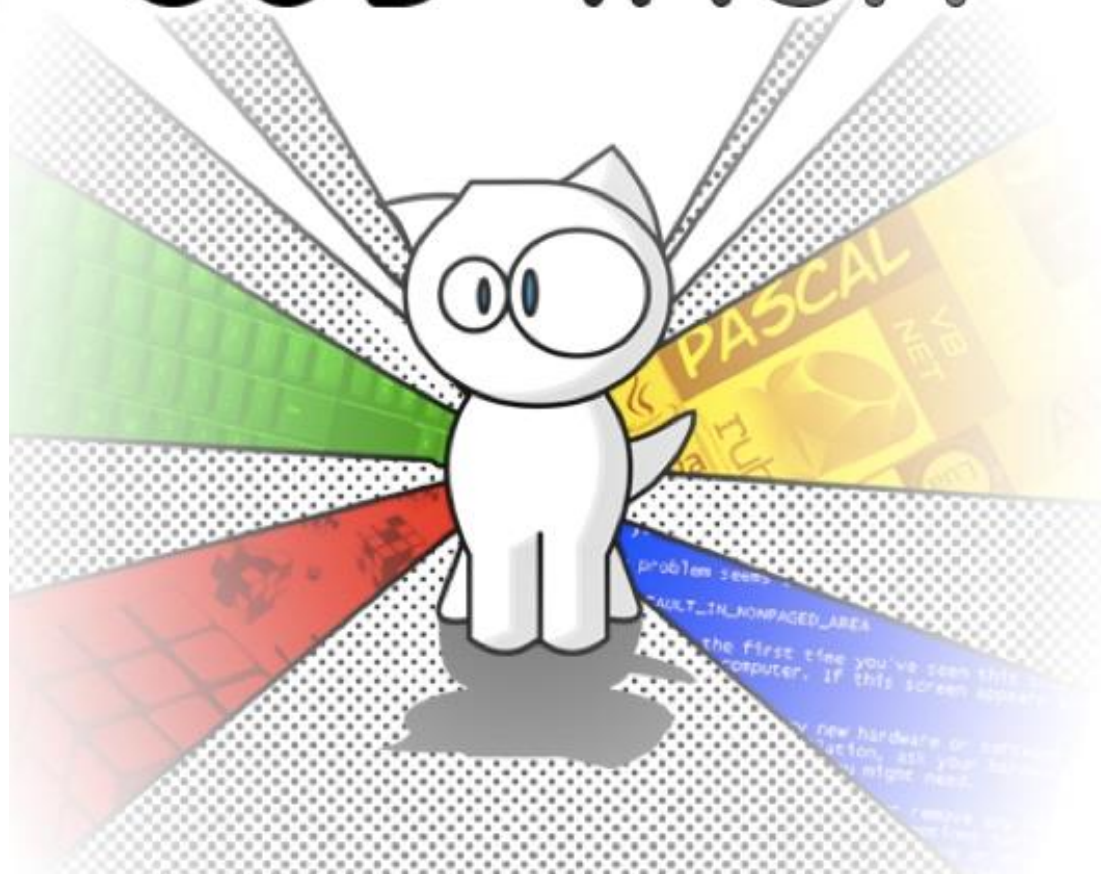


COD'INSA

Merci de votre attention !



COD'INSA



Le concours d'informatique
du réseau d'écoles INSA