

LES ARCHITECTES DU CODE



LES ARCHITECTES DU CODE

PRINCIPES

En tant qu'équipe, **5 villageois** vous seront attribués. Vous pourrez donner des instructions à chacun de vos villageois afin d'obtenir des ressources.

Ces ressources vous permettront alors d'acheter des bâtiments. Certains bâtiments auront pour objectif d'augmenter la productivité de vos villageois sur la case où ces derniers sont positionnés, d'autres de récolter des ressources ou gagner des points à chaque tour de jeu et d'autres vous permettront de débloquent certaines fonctionnalités.

Les points accumulés par votre équipe permettront d'établir le classement (qui sera en partie pris en compte lors du jury). En revanche, la **pollution** générée par les actions de vos villageois et bâtiments **impacter** **négalement votre classement** lors du jury.

Vous devrez également être en mesure **d'afficher le jeu** en temps réel, soit en vous basant sur les ressources 2D, soit 3D, qui vous sont transmises. L'affichage du jeu devra être en mesure de montrer la carte, les types de végétation présents à chaque case, les bâtiments construits ainsi que les villageois de votre équipe ou des autres. Si d'autre fonctionnalités vous semblent intéressantes à mettre en place, celles-ci sont les bienvenues !

Le choix des technologies est **libre** ! (Java, JS, Unity, Python..), cependant il faut s'assurer que la stack technique permet de créer des IHM **esthétiques** et de réaliser des **appels REST**. Nous ne pouvons pas vous garantir un support technique sur toutes les technologies.

Un usage raisonnable et responsable des API est attendu. N'hésitez pas à échanger avec les autres équipes en cas de questions.

Les modalités de l'évaluation seront expliquées sur le discord dont le lien vous a été communiqué. Le rendu et les fonctionnalités du front seront une partie importante de l'évaluation du jury.

En résumé :

- **Prenez connaissances des APIs REST et des modèles et exploitez-les**
- **Faites un maximum de points tout en faisant attention à la pollution**
- **Réaliser un Front 2D ou 3D le plus complet possible**

LA CARTE DU JEU

La carte sera composée de **33 cases par 33, soit 1089 cases**. Chacune des cases possède des **ressources limitées**, permettant à vos villageois ainsi qu'à vos bâtiments de récolter ces dernières tant qu'il y en a. Les cases de lac vous permettront de placer les bâtiments spécifiques à ces dernières, et les cases de plaine ou de désert permettront de placer les autres bâtiments (pas de différence).

LES VILLAGEOIS

Afin de donner des instructions à l'un de vos villageois, un **appel à l'API** sera nécessaire. Il n'est possible de fournir qu'une seule instruction à la fois à un villageois.

Une fois l'instruction exécutée, le villageois devra **attendre 12.5 secondes** (appelé cooldown) avant de pouvoir réaliser une nouvelle instruction qui lui sera alors fournie.

Les instructions possibles pour chaque villageois sont les suivantes :

- **DEPLACEMENT_DROITE** : Permet de déplacer le villageois sur la case à droite de lui
- **DEPLACEMENT_GAUCHE** : Permet de déplacer le villageois sur la case à gauche de lui
- **DEPLACEMENT_HAUT** : Permet de déplacer le villageois sur la case en haut de lui
- **DEPLACEMENT_BAS** : Permet de déplacer le villageois sur la case en bas de lui
- **RECOLTER** : Récolte la ressource demandée si cette dernière est présente sur la case où se situe le villageois. Si un bâtiment de booste de récolte de votre équipe est positionné sur cette case, le villageois récoltera davantage de ressource.
- **COMMENCER_CONSTRUCTION** : Démarre la construction du bâtiment demandé sur la case où se situe le villageois. Les ressources nécessaires à la construction de ce bâtiment sont alors imputées à l'équipe
- **CONSTRUIRE** : Permet de progresser dans la construction d'un bâtiment inachevé
- **RECYCLER_BATIMENT** : Détruit le bâtiment de cette case afin de libérer cette dernière. 75% des ressources nécessaires à la construction de ce bâtiment seront alors restituées à l'équipe. Recycler un bâtiment génère de la pollution.

Un évènement sera stocké à chaque fois qu'un villageois se déplacera ainsi qu'à chaque fin de cooldown. Vous pourrez récupérer les **50 derniers événements** concernant les villageois de votre équipe ou des autres équipes à tout moment, si vous possédez les bâtiments nécessaires à cela (voir liste des bâtiments pour plus d'infos).

Il est fortement conseillé de récupérer cette liste d'évènements assez fréquemment (exemple : 10 secondes) afin d'être toujours à jour, mais ne pas le faire à une fréquence plus rapide.

LES BATIMENTS

Les bâtiments vous permettront soit d'améliorer la productivité de vos villageois, soit de récolter des ressources ou des points de manière automatique à chaque tour de jeu (**toutes les 12.5 secondes**), soit de produire de l'énergie ou soit de débloquer certaines fonctionnalités.

Lorsqu'un bâtiment est placé sur une case, un certain nombre d'actions de construction sera alors nécessaire afin que ce dernier soit terminé et que ses effets s'appliquent. Il faudra donc donner l'instruction de construire ce bâtiment à un ou plusieurs villageois présents sur cette case jusqu'à complétion du bâtiment.

Lorsqu'un bâtiment est placé sur une case, aucun autre bâtiment ne pourra alors être construit sur cette même case. Cependant, si une action de "recyclage" du bâtiment est réalisée par l'un de vos villageois, ce dernier sera détruit. **75% des ressources** nécessaires à la construction de ce bâtiment seront alors restituées à l'équipe et la case sera de nouveau libre.

Les bâtiments permettant la récolte de ressources ou de points de manière automatique à chaque tour auront **besoin d'énergie** afin de fonctionner et d'apporter leur bonus.

Pour se faire, des bâtiments producteurs d'énergies devront être construits et avoir la capacité de fournir **suffisamment d'énergie** pour alimenter tous vos bâtiments. Si l'énergie produite lors d'un tour est insuffisante, aucun des bâtiments nécessitant de l'énergie ne produira de ressource et **l'énergie produite lors du tour sera perdue**.

Les bâtiments de type **Merveilles** ne pourront être **construit que par une seule équipe**. Une fois une merveille terminée par une des équipes, celles ayant également démarré la construction de celle-ci en amont ne pourront pas terminer cette dernière.

Un évènement sera stocké à chaque fois qu'un bâtiment commence à être construit, s'achève ou est recyclé. Vous pourrez récupérer les **50 derniers événements** concernant les villageois de votre équipe ou des autres équipes à tout moment, si vous possédez les bâtiments nécessaires à cela (voir liste des bâtiments pour plus d'infos).

Il est fortement conseillé de récupérer cette liste d'évènements assez fréquemment (exemple : 10 secondes) afin d'être toujours à jour, mais ne pas le faire à une fréquence plus rapide.

LES APIs

L'OAS décrivant les APIs exposées par le serveur est disponible sur le Discord.

Voir <https://editor.swagger.io/>

RECUPERATION DE VOTRE TOKEN

Les APIs sont sécurisées via un système OIDC et nécessite donc une authentification pour fonctionner. Cette authentification se fait en fournissant, via un header HTTP nommé “Authorization”, un token d’authentification (au format JWT).

```
curl -H "Authorization: Bearer votre_token" https://api.example.com/data
```

Pour récupérer votre token, il vous faut appeler une URL spécifique de la manière suivante :

```
curl --location 'http://149.202.79.34:8081/realms/codelemans/protocol/openid-connect/token'
--header 'Content-Type: application/x-www-form-urlencoded'
--data-urlencode 'client_id=wololo-backend'
--data-urlencode 'username={votre_login}'
--data-urlencode 'password={votre_mot_de_passe}'
--data-urlencode 'grant_type=password'
```

Votre token sera dans la donnée “access_token” et il sera valide 48h à compter de l’appel.

● ● ● ●
**CE TOKEN EST LIE A VOTRE EQUIPE. NE LE COMMUNIQUEZ PAS EN DEHORS DE
VOTRE EQUIPE. Y COMPRIS SUR GIT**
● ● ● ●

AIDE POUR LE REST

L’API REST c’est quoi ?

Une API (Application programming interface) est un programme permettant à deux applications distinctes de communiquer entre elles et d’échanger des données.

Le REST est un style d’architecture logicielle beaucoup utilisé en web de par sa manipulation simple et rapide des données.

Une API REST est donc une API conforme à l’architecture REST. Elle consiste à mettre à disposition une opération permettant par exemple de consulter ou modifier une ressource (donnée). Il existe plusieurs types d’opération :

- GET : Consulter une ressource
- POST : Création d’une nouvelle ressource
- PUT : Mise à jour d’une ressource existante
- DELETE : Suppression d’une ressource

Comment l’utiliser

Dans le cas de ce sujet des 24h, nous vous avons mis à disposition plusieurs APIs permettant d’interagir avec notre serveur et notre base de données. Chacune de ces APIs est constituée d’un type (GET, POST, ...) et de l’URL associée.

Les appels peuvent être testés via des clients comme <https://www.usebruno.com/> ou <https://httpie.io/>.

Implémenter les appels REST dans du code :

Bien que les langages les plus à même de faire des appels REST sont les langages web (Typescript, Javascript, ...), il est possible d'appeler nos API avec la plupart des langages de programmation. Voici quelques exemples :

TypeScript (Angular) : <https://angular.io/guide/http#starting-the-request>

Javascript : <https://stackoverflow.com/questions/36975619/how-to-call-a-rest-web-service-api-from-javascript>

Python : https://rtavenar.github.io/poly_python/content/api.html

Java : <https://stackoverflow.com/questions/1359689/how-to-send-http-request-in-java>

Smalltalk : [GitHub - SeasideSt/Seaside: The framework for developing sophisticated web applications in Smalltalk.](#)

Pour d'autres langages, il existe de multiples exemples disponibles sur internet.

