Zombi invasion

# Génie Logiciel



Figure 1

#### Un cours donné par M. Lefrançois

#### Un projet réalisé par MM. Mbassi, Lagha, Lankeu et Thomas

Table des matières

[Génie Logiciel 1](#_Toc512593211)

[Concept : 3](#_Toc512593212)

[Gameplay : 3](#_Toc512593213)

[Les héros : 3](#_Toc512593214)

[Les ennemis : 3](#_Toc512593215)

[Les bonus : 4](#_Toc512593216)

[Les vagues d’ennemi : 4](#_Toc512593217)

[Les décors : 4](#_Toc512593218)

[Les distributeurs : 4](#_Toc512593219)

[Fin de partie : 4](#_Toc512593220)

[Côté utilisateur : 4](#_Toc512593221)

[Menu : 4](#_Toc512593222)

[Commandes : 5](#_Toc512593223)

[Interface graphique : 5](#_Toc512593224)

[Réseau : 5](#_Toc512593225)

[Côté client : 6](#_Toc512593226)

[Côté serveur : 6](#_Toc512593227)

[Modèle de domaine client/serveur : 6](#_Toc512593228)

[Protocole utilisé : 7](#_Toc512593229)

[Base de données : 7](#_Toc512593230)

[Musique : 7](#_Toc512593231)

[Mock-up : 7](#_Toc512593232)

[Cas d’utilisation : 12](#_Toc512593245)

[Backlog : 14](#_Toc512593249)

[Scrum : 17](#_Toc512593250)

[Bibliographie : 17](#_Toc512593251)

# Concept :

Pour ce projet de semestre, nous avons décidé de programmer un jeu vidéo dans le style arcade.

Les parties en bleu désignent des fonctionnalités optionnelles et seront donc implémentées que si le temps nous le permet.

### Gameplay :

Le jeu sera de deux à quatre joueurs, connectés via un serveur. Les joueurs devront coopérer pour survivre à des vagues de morts vivants consécutives.

Avec le temps, le nombre et la résistance des ennemis grandiront. En cours de partie, des bonus permettant de récupérer de la vie, des armes, des munitions ou des objets seront relâchés par certains ennemis et apparaîtront à fréquence régulière dans des parties aléatoires du terrain.

### Les héros :

Les héros posséderont un pourcentage de vie, une arme de combat rapproché, une arme à feu et un certain nombre de munitions. À cela pourra venir s’ajouter un pourcentage d’armure, de l’argent, des barricades, différents objets et armes.

Toutes les informations propres à un joueur seront directement affichées en bas de son écran, la vie et le pourcentage d’armure sous la forme d’une barre, le reste sous formes d’icônes avec le nombre d’unités en question.

### Les ennemis :

Les ennemis posséderont un pourcentage de vie et seront divisés en deux classes. Au début de la partie, seul quelques ennemis seront présents, de la classe la plus faible. Au fur et à mesure des vagues, le nombre d’ennemi non seulement augmentera, mais d’autres ennemis, plus résistants, apparaîtront de plus en plus régulièrement.

Les ennemis étant des zombis, ils attaqueront directement au corps à corps les joueurs. À leur mort, certains ennemis feront apparaître des objets bonus.

Les ennemis les plus résistants feront automatiquement apparaître des objets, instancié aléatoirement (munitions, argent ou armure). Les plus faibles auront un certain pourcentage de chance d’en faire apparaître, une nouvelle fois instancié aléatoirement.

### Les bonus :

La quantité physique d’un bonus (munitions et autres objets quantifiables), sera limités par des bornes en fonction du type d’ennemi vaincu.

Les bonus apparaissant aléatoirement sur le terrain, eux, seront limités par des bornes qui évolueront en fonction du nombre de vagues/d’ennemis vaincus.

### Les vagues d’ennemi :

Les vagues d’ennemis arriveront à fréquence de trois minutes. La première vague sera composée de 20 ennemis faibles. A chaque nouvelles vague, le nombre d’ennemis faibles augmentera de cinq. Toutes les cinq vagues viendra s’ajouter un ennemi de type fort supplémentaire.

### Les décors :

Le jeu prendra lieu sur un terrain créer par nos soins. La gestion de la vue se fera à la troisième personne.

Le terrain sera une vaste plaine, où se trouvera au milieu une bâtisse dans laquelle les joueurs pourront s’abriter. Celle-ci sera dans un premier temps verrouillée et les joueurs pourront, si ils le désirent, en débloquer l’accès contre un certain montant. Ils pourront donc ainsi limiter le flux d’ennemis qui viennent sur eux, les ennemis ne pouvant pas apparaître à l’intérieur, et placer des barricades sur les accès afin de mieux se protéger. À l’intérieur de la maison se trouvera quatre distributeurs, un de chaque type.

Le tout sera chargé une seule fois, bien que le joueur ne puisse pas voir l’ensemble du terrain.

### Les distributeurs :

Les distributeurs présents dans la maison permettront, en fonction de leur type, aux joueurs d’obtenir contre une certaine somme de la vie, des munitions, de l’armure ou des barricades.

### Fin de partie :

La partie s’arrête à la mort du dernier joueur. Quand un joueur meurt, il retourne automatiquement au menu principal. Si joueur désire commencer une partie mais qu’une autre est déjà en cours, il doit attendre la fin de celle-ci pour que la recherche permette d’en commencer une nouvelle.

# Côté utilisateur :

### Menu :

Les joueurs lanceront un fichier .exe. Il devra alors s’authentifier à l’aide d’un nom d’utilisateur et d’un mot de passe. Le joueur débouche alors sur un nouveau menu, composé de trois sections :

**-** **Start**

Un clic sur le bouton *start* permettra de lancer la recherche de joueurs. Un second menu s’affiche alors, permettant de voir le nombre de joueurs actuellement trouvé. À tout moment, le joueur peut soit décider de quitter la recherche, soit de lancer la partie. Si tous les joueurs présents ont décidé de lancer la partie, celle-ci débute. Si il y a quatre joueurs connectés, la partie commence automatiquement. Quitter la recherche permet un retour au menu principal.

**- High score**

Cette section contiendra l’ensemble des meilleurs scores atteints à ce jour, celui de l’équipe mais également celui de chacun de ses membres avec éventuellement une possibilité de voir le nombre d’ennemis abattus par joueur lors d’une partie donnée.

**- Commandes**

Cette section contiendra l’ensemble des commandes. Il sera possible de les réassigner si le joueur est connecté en tant qu’administrateur et ces dernières seront réaffichées lors du chargement de la partie.

### Commandes :

Une fois la partie démarrée, l’utilisateur doit assurer sa survie. Pour cela, les commandes suivantes sont à sa disposition :

- La visée s’effectue avec la souris

- Les déplacements se font avec les touches *a*, *s*, *d* et *w*

- Le clic *droit* permet de tirer

- Le clic *gauche* permet de recharger

- La touche *escape* permet de quitter la partie

- La touche *q* permet de changer d’arme

- La touche *espace* permet d’utiliser son arme blanche, de débloquer l’accès à la maison et d’effectuer un achat au distributeur s’il y en a un présent en face

- La touche *e* permet de passer en mode construction, si cela est possible. Une barricade, représentée par une planche apparaît alors et le positionnement s’effectue avec les touches *a*, *s*, *d* et *w* pour la distance, un appui de la touche *espace* pour confirmer puis les touches *a*, *s*, *d* et *w* permettent une nouvelle fois de servir à la rotation. À tout moment, si le joueur vient à faire un clic droit, il tire un coup de son arme à feu. S’il fait un clic gauche, il annule uniquement la construction.

# Interface graphique :

L’interface graphique sera implémentée à l’aide de la librairie LibGDX. La vue sera du dessus et le terrain ne sera chargé qu’une seule fois.

# Réseau :

Nous avons décidé, afin de limiter le nombre d’informations transitant sur le serveur, de ne pas créer un objet physique sur le terrain pour chaque ennemi. Ceux-ci seront créé directement à l’arrivée dans le champ de vision d’un joueur, ils n’auront donc pas à se déplacer sur le terrain entier avant de l’atteindre. Chaque joueur aura pour cela un pourcentage de probabilité de voir apparaître un ennemi.

### Côté client :

Le client stock les informations suivantes :

* Sa position
* Son pourcentage de vie, d’armure, etc…
* La position des bonus non ramassés
* Si la maison est ouverte ou non

Le client se charge de calculer :

* Ses déplacements
* Si, lorsqu’il tire, il atteint un ennemi
* Si un ennemi arrive dans sa zone
* Si il ramasse un objet, de quel type ce dernier est
* Si il se fait toucher par un ennemi (gestion de sa vie)

Il enverra au serveur :

* Sa position en continu
* Une notification si il touche un ennemi
* Une notification si un ennemi apparaît dans sa zone
* Une notification si il est mort
* Une notification si il ramasse un objet
* Une notification si il ouvre la maison

### Côté serveur :

Le serveur stock les informations suivantes :

* La position de tous les ennemis
* Le pourcentage de vie de tous les ennemis

Le serveur se charge de calculer :

* Les déplacements des ennemis
* La vie des ennemis
* Si à la mort d’un ennemi, ce dernier relâche un bonus

Il envoie aux joueurs :

* La position de tous les joueurs
* La position de tous les ennemis
* La mort d’un ennemi (pour éventuellement une animation)
* Une notification lorsqu’un bonus vient d’être ramassé
* Une unique notification si la maison est ouverte

### Modèle de domaine client/serveur :

Le modèle de domaine client/serveur adopté sera le modèle LAN. Il permet aisément la connexion entre quatre joueurs et notre projet ne requiert pas un accès internet.

### Protocole utilisé :

Le protocole utilisé pour l’échange des données entre le client et le serveur sera le protocole TCP/IP.

# Base de données :

La persistance des données se fera au travers de fichiers xml et JSon permettant de sauvegarder l’ensemble des informations à stocker.

# Musique :

Différents bruitages permettront aux joueurs de vivre une expérience plus immersive. Il sera notamment possible d’entendre :

* Les bruits de pas
* Des cris poussés par les ennemis
* Les coups d’armes à feu
* Les coups d’armes blanches

À cela s’ajoutera une musique d’ambiance renforçant l’atmosphère pesante et la pression exercée sur le joueur.

# Mock-up :

### S’authentifier :



## Zombi Invasion

Password :

Username :

Password :

Figure 2

### Menu principal :



Start

Commandes

High Score

## Zombi Invasion

### Recherche de joueurs (start) :



Quitter

…

Recharger : Click gauche

Tirer : Click droit

Recherche de joueurs… (1 sur 4)

Password :

## Zombi Invasion

### Consulter les commandes (joueur) :



Quitter

…

Recharger : Click gauche

Tirer : Click droit

## Zombi Invasion

### Modifier les commandes (admin) :



…

Recharger : Click gauche

Tirer : Click droit

Réinitialiser

Sauver

Quitter

## Zombi Invasion

### En cours de partie :



* 1200

 100

E. faible

Ennemi fort

Allié

Allié

Héro

Figure 3

### Quitter une partie en cours (escape) :



* 1200

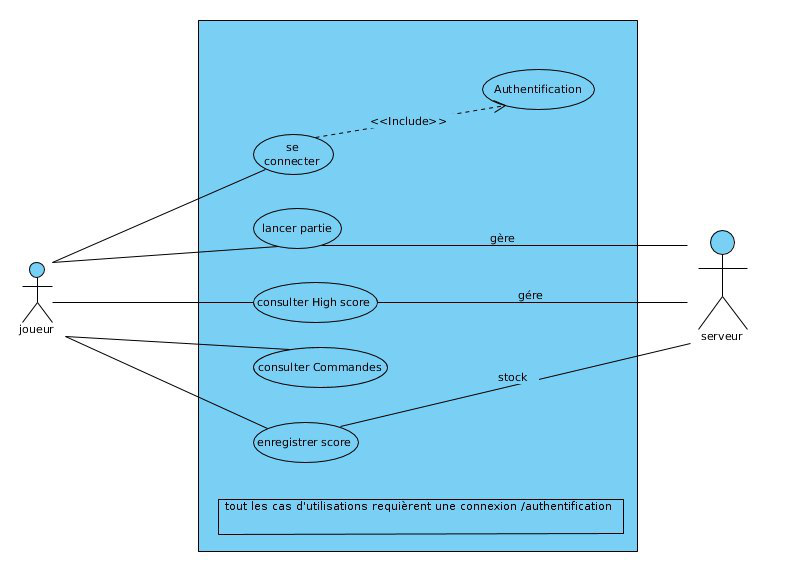
 100

Reprendre

Quitter

Héro

# Cas d’utilisation :



## Description du cas d’utilisation :

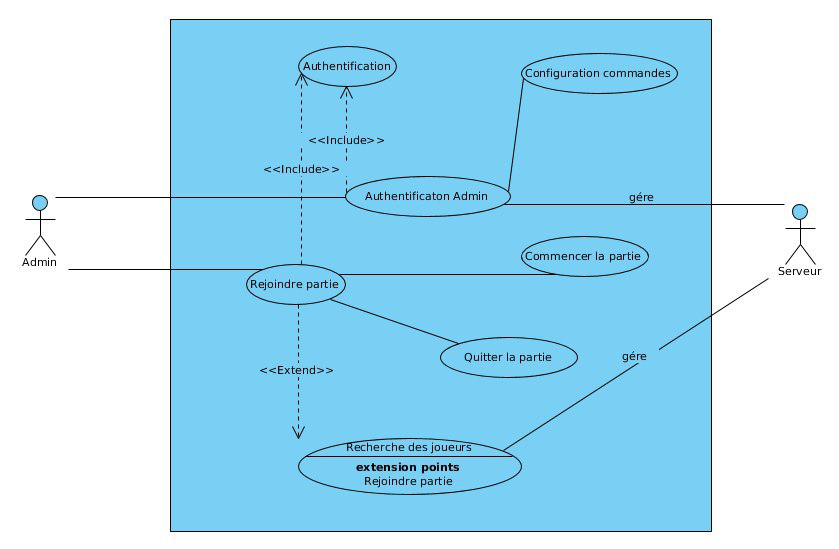
**Se connecter :** le joueur a besoin de se connecter sur le serveur pour pouvoir récupérer ses coordonnées et accéder au fonctionnement de jeu. En outre, cette authentification permet au serveur de savoir si le joueur est administrateur ou non. Il peut donc permettre au joueur d’éventuellement éditer l’ensemble des commandes

**Lancer partie :** le joueur peut demander au serveur de lui lancer une partie. Le serveur va alors commencer la recherche de joueurs.

**Consulter High score** : le joueur peut demander de consulter les meilleurs scores.

**Consulter commandes** : représente la commande help dans notre jeu. Il va afficher au joueur les commandes nécessaires pour jouer.

**Enregistrer score :** le joueur peut choisir d’enregistrer son score. S’il désire quitter le jeu, le programme lui demande s’il veut enregistrer son historique du jour ou non.

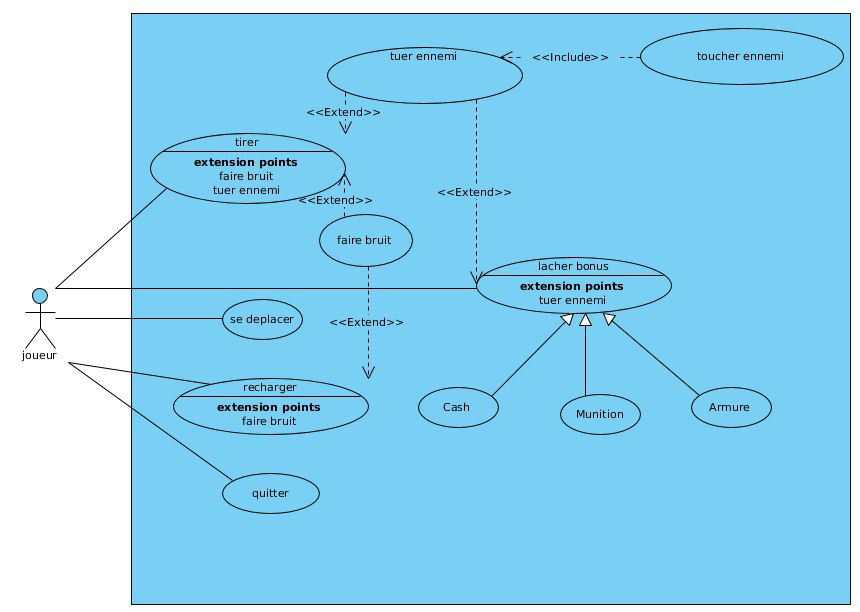


## Description du cas d’utilisation :

**S’authentifier en tant qu’administrateur :** Cette authentification permet au serveur de savoir si le joueur est administrateur ou non. Il peut donc permettre au joueur d’éventuellement éditer l’ensemble des commandes

**Rejoindre partie :** le joueur peut demander au serveur de rejoindre une partie. Il attendra le début de la seconde si les autres joueurs en ont déjà lancé une. Pendant la recherche de joueurs, il peut décider de quitter la partie à tout moment. Si il ne le fait pas, la partie commence.

Pour rejoindre une partie, il faut obligatoirement s’être authentifier avant.



## Description du cas d’utilisation :

**Se déplacer :** le peut se déplacer sur le terrain.

**Faire du bruit:** le joueur fait du bruit automatiquement si il tire ou si il recharge.

**Tuer un ennemi** : Il faut obligatoirement avoir tiré pour toucher un ennemi. L’inverse n’est pas réciproque. De même, tuer un ennemi inclus forcément d’avoir touché l’ennemi au préalable.Lorsque l’on tue un ennemi, il peut parfois lâcher un bonus à sa mort. Les bonus sont soit de l’argent, soit des munitions, soit de l’armure.

# Backlog :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No de l’histoire** | **Nom de l’histoire** | **Type de l’histoire** | **Descriptif** |  |
| **01** | Apparition du 1er Menu | User Story | 1er Menu qui apparait lorsqu’on lance l’application | client |
| **02** | login | User Story | Chaque joueur doit se loguer, à chaque lancement du jeu | serveur |
|  |  |  | Le jeu sera de deux à quatre joueurs, connectés via un serveur. | serveur |
| **03** | Sélection personnage | User Story | faire un choix entre les personnages prédéfinis dans le jeu (choix du joueur). | client |
| **04** | Connexion et chargement joueur et scène de jeu | User Story | faire établir la connexion entre chaque jouer et le serveur | serveur |
| **05** | Affichage nouvelle interface de jeu | User Story | Toutes les informations propres à un joueur seront directement affichées en bas de son écran, la vie et le pourcentage d’armure sous la forme d’une barre, le reste sous formes d’icônes avec le nombre  d’unités en question | Interface graphique |
| **06** | Dialoguer par utilisation des combinaisons de touches | User Story | communiquer avec mes coequipiés , à travers des commandes predefinies(joueurs devront coopérer pour  survivre à des vagues de morts vivants consécutives) | client |
| **07** | Déplacement du personnage | User Story | prendre les directions différentes : gauche , droite, devant, arriere | client |
| **08** | bonus | User Story | pouvoir ramasser des bonus afin d'augmenter le poucentage de vie |  |
| **09** | bonus | User Story | recuperer des bonus, ce qui permettront de récupérer de la vie, des armes, des munitions |  |
| **10** | Déblocage nouveau terrain | User Story | pouvoir débloquer des nouveaux terrains de jeu après avoir acquis un nombre de bonus prédéfinis par le  développeur |  |
| **11** | Relâchement des objets par Ennemi | User Story | relâchés des objets dangereux qui apparaîtront à fréquence régulière dans des parties aléatoires du terrain |  |
| **12** | Pourcentage de vie jouer/adversaire | User Story | au contact avec l'adversaire(les balles tirées), réduire le pourcentage de vie. |  |
| **13** | Spécifications sur les catégories d’ennemi | User Story | Les ennemis posséderont un pourcentage de vie et seront divisés en deux classes () |  |
| **14** | tirer, sauter, activer la communication | User Story | Les users peuvent faire sauter le personnage, tirer sur l’adversaire et autre… |  |
| **15** | Recommencer la partie | User Story | Recommencer la partie, s'il meurt ou s'il veut (suppose qu'il n'a pas de coéquipier) |  |
|  | | | | |

# Scrum :

**Clé du projet** : ZOMBI2018

**Nom du projet**: Zombi invasion

# Bilan d’itération :

### Itération n°1 :

**Bilan su la terminaison des histoires :** Nous avions prévu une histoire en principal. Celle-ci comprenait la création d’un serveur, l’implémentation d’une classe JSon pour la sérialisation des données utilisateurs (username, password), une interface pour entrer les données d’un utilisateur et enfin l’envois, la réception et la vérification côté serveur de ces données.

**Vélocité du sprint :** Nous estimons notre histoire terminée à 80% environ, et donc notre vélocité au même pourcentage.

**Replanification :** Nous avons donc décider de reporter la finalisation de notre histoire au sprint suivant. La majeure partie de nos classes est terminée cependant, nous rencontrons encore quelques soucis techniques afin de faire fonctionner le tout ensemble.

**Commentaire général :** Nous avons malheureusement pris du retard, ceci en raison d’une grande charge de travail requise de l’ensemble des cours, mais également car nous n’avons pas suffisamment été bien organisé en tant que groupe. L’utilisation de technologies que nous ne maîtrisons pas nous a passablement fait perdre de temps et nous devons encore étudiez ces-dernières avant de pouvoir faire un progrès notable.

**Autocritique :** Bien que le retard ne soit pas encore conséquent, nous devons rapidement rattraper ce dernier afin d’éviter un effet boule de neige. Nous devons apprendre à mieux s’organiser en tant que groupe et mieux diviser le travail afin de ne pas se retrouver bloquer par deux implémentations différentes d’une même fonctionnalité.

### Bilan Personnels :

**Mbassi :** J’ai pris beaucoup de temps à comprendre la technologie utilisé pour notre projet chose faite maintenant. Mettre à jour aussi iceScrum m’a aussi pris pas mal de temps mais avec l’aide de Ben iceScrum est maintenant à jour. J’ai créé un fichier Json pour gestion d’authentification coté serveur.

**Lagha :** Pour comprendre la technologie iceScrum ca nous pris beaucoup de temps ,avec le retard qu’on a eu pour créer le groupe s’était un peu difficile de s’organiser pour comblé le retard, je me suis charger de la partie serveur et j’ai réussi a créé un serveur multithread qui traite plusieurs client au même temps et avec plusieurs commande à disposition, et j’ai pu aider mais collègues coté client avec les interface qu’on devais les livrés pour ce sprint

**Lankeu :**

Jusqu’à present , j’ai pu créer l’interface le bouton newUser et la saisie des données du joueur, l’affichage des commandes(ainsi que l’edification), redirection de fenetre en cas d’appuis sur un bouton de l’interface. Concernant la création du joueur, j’ai pas encore géré l’envoi des information au serveur pour la création du validation et la mise à jour du compte crée. Je dois le faire avec celui qui s’occupe du serveur. J’ai rencontré quelque difficulté sur la mise en place de certaines interfaces, j’ai dû bidouiller ce qui fait en sorte que le code fait très propre, mais je verais avec mes collegues si je peux avoir de l’aide afin optimiser le code.

**Thomas :** Je n’ai malheureusement pas pris suffisamment de temps sur ma partie (serveur côté client, affichage côté client) et, trois heures de plus environs auraient été nécessaires afin de bien finaliser ce sprint. J’essayerai de combler rapidement ce retard afin qu’il ne s’amplifie pas trop.

# Bibliographie :

* Figure 1 :

<https://www.google.ch/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwjX2bjW1b7aAhWMKFAKHU7lBbQQjRx6BAgAEAU&url=http%3A%2F%2Fwww.jeuxvideo.com%2Fnews%2F519913%2Fzombie-l-invasion-du-jeu-video-ne-fait-que-commencer.htm&psig=AOvVaw3l-laWSuVC_jf1LJf3ONHa&ust=1523963787084332>

* Figure 2 :

<https://www.google.ch/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwjF8Jez1b7aAhVKKVAKHd5ICxcQjRx6BAgAEAU&url=http%3A%2F%2Fes.terraria.wikia.com%2Fwiki%2FArchivo%3ATerraria_living_wood_wallpaper_by_gturbo5-d6dvwla.jpg&psig=AOvVaw2eh2CTqaRGTIF0TeiCu8L-&ust=1523963680111101>

* Figure 3 :

<https://www.google.ch/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwjKiOuD277aAhVJKlAKHSlLAQoQjRx6BAgAEAU&url=https%3A%2F%2Fhero339.deviantart.com%2Fart%2FJungle-Terrain-Q4-595573942&psig=AOvVaw1hlvRMZimVdZ6KwHxCzTSA&ust=1523965207626491>