

**Introduction**  
 Jeu de rôle multijoueur par navigateur web en temps réel et persistant  
 ORPG - "Online Role Playing Game"

**Objectifs doubles**

- Pédagogiques
- Produits

**Contraintes**

- Temps : 3 mois de travail (5 heures par semaine)
- Technologies : Node.js, HTML5, Canvas et MongoDB
- Non financières

# ZOMB'ISTIA

## Jeu multijoueur temps réel en ligne

Abdelmounaim BELGHALEM  
 Johan BRUN  
 Brendan GOUIN  
 Florian HARDY

**Fonctionnalités**

**Site web**

- Chats
- Classement
- Page des scores
- Zone administration

**Jeu**

- Diriger un personnage
- Explorer l'IstiA en 2D
- Interactions avec :
  - Objets
  - Cases
  - Zombies
  - Joueurs
- Attaque de nuit

**Site web**

3<sup>ème</sup> tiers

### Travail réalisé

**Architecture globale (3 tiers - 5 couches)**

- Architecture web classique
- Découpage en 5 couches (4 sur le serveur)

**Serveur**

3<sup>ème</sup> tiers

**Gestion de Projet**

**Technologies**

**Plateau de jeu**

1<sup>er</sup> tiers

**Base de données**

1<sup>er</sup> tiers

- Sauvegarde permanente des données
- Communication avec le serveur très rapide
- Schémas initialisés au lancement du serveur
- Facilité de prise en main

**Résultats du projet**

- Collecte de données précises
- Les utilisateurs peuvent interagir
- Développement continu

**Difficultés rencontrées**

- Formule de base de données
- Une mauvaise gestion
- Non respect du timing de l'application
- Problèmes de latence et de débit
- Problèmes de serveur

**Améliorations possibles**

- Optimisation de la base de données
- Optimisation de l'interface utilisateur
- Optimisation de la communication avec le serveur
- Optimisation de la gestion des données
- Optimisation de la gestion des utilisateurs
- Optimisation de la gestion des ressources



EI4 - AGI - 2013/2014

**Conclusion**

- Application fonctionnelle
- Problèmes surmontés grâce à :
  - Gestion de projet (respect du planning, définitions des tâches)
  - Architecture solide et découpée
- Apprentissage technologique et humain

## Introduction

Jeu de rôle multijoueur par navigateur web en temps réel et persistant  
ORPG - "Online Role Playing Game"

Objectifs doubles

- Pédagogiques
- Produits

Contraintes

- Temps : 5 mois de travail (5 heures par semaine)
- Technologies : Node.js, HTML5, Canvas et MongoDB
- Non financières

# ZOMB'ISTIA

## Jeu multijoueur temps réel en ligne

Abdelmounaim BELGHALEM  
Johan BRUN  
Brendan GOUIN  
Florian HARDY

### Fonctionnalités

- Site web
  - Jeu
- Tchat
  - Classement
  - Page des scores
  - Zone administration
  - Diriger un personnage
  - Explorer l'isole en 2D
  - Interactions avec
  - Objets
  - Cases
  - Zombies
  - Jokers
  - Attaque de nuit

2

### Technologies



3

### Site web



10

## Travail réalisé

### Architecture globale (3 tiers - 5 couches)



- Architecture web classique
- Découpage en 5 couches (4 sur le serveur)

### Serveur



10

### Plateau de jeu



20

### Base de données



10

### Gestion de Projet



### Résultats du projet

- Galerie de jeux vidéos
- Jeu fonctionnel
- Site web accessible

20

### Démonstration



20

### Difficultés rencontrées

- Problème de base de données
- Problème de serveur
- Problème de gestion de projet
- Problème de communication

20

### Améliorations possibles

- Intégration de nouvelles technologies
- Amélioration de l'interface utilisateur
- Amélioration de la performance
- Amélioration de la sécurité

20

### Conclusion

- Application fonctionnelle
- Problèmes surmontés grâce à :
  - Gestion de projet (respect du planning, définitions des tâches)
  - Architecture solide et découpée
- Apprentissage technologique et humain

20



EI4 - AGI - 2013/2014

# *Introduction*

Jeu de rôle multijoueur par navigateur web en temps réel et persistant  
ORPG - “Online Role Playing Game”



## Objectifs doubles

- Pédagogiques
- Produits

## Contraintes

- Temps : 5 mois de travail (5 heures par semaine)
- Technologies : Node.js, HTML5, Canvas et MongoDB
- Non financières

# Fonctionnalités

## Site web

- Tchats
- Classement
- Page des scores
- Zone administration



## Jeu

- Diriger un personnage
- Explorer l'IstiA en 2D
- Interactions avec
  - Objets
  - Cases
  - Zombies
  - Joueurs
- Attaque de nuit



# Technologies

## Langages de développement



## Plate-formes logicielles



## Libraires et Frameworks



## Outils d'aide au développement



3

# Langages de développement

**HTML**



Contenu du site

**CSS**



Présentation du site

**JS**



JavaScript

Langage interprété par le navigateur

4

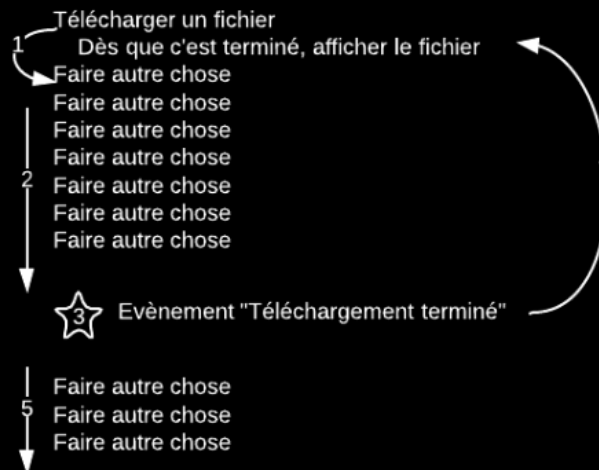
# Plate-formes logicielles



Non-bloquant (asynchrone)  
et événementiel  
Utilise la puissance de JavaScript



Base de données NoSQL  
Mongoose + Node.js



5

# Libraires et Frameworks

SOCKET IO



EaselJS



CreateJS



PreloadJS



Bootstrap



jQuery

# Outils d'aide au développement



Git



Aptana



Sublime Text



MongoVUE

## Langages de développement

**HTML**  
Contenu du site

**JS**  
JavaScript  
Langage interprété par le navigateur

**CSS**  
Présentation du site

4

## Plate-formes logicielles

**NODE**  
Non-bloquant (asynchrone) et événementiel  
Utilise la puissance de JavaScript

```
var http = require('http');
var url = 'http://www.google.com';
http.get(url, function(res) {
  console.log('statusCode:', res.statusCode);
  console.log('headers:', res.headers);
  res.on('data', function(chunk) {
    console.log(chunk);
  });
});
```

**MongoDB**  
Base de données NoSQL  
MongoDB + Node.js

5

## Libraires et Frameworks

**SOCKET IO**

EasyJS, CreatJS, PreloadJS

**B** Bootstrap

**jQuery**

6

## Outils d'aide au développement

**Git**

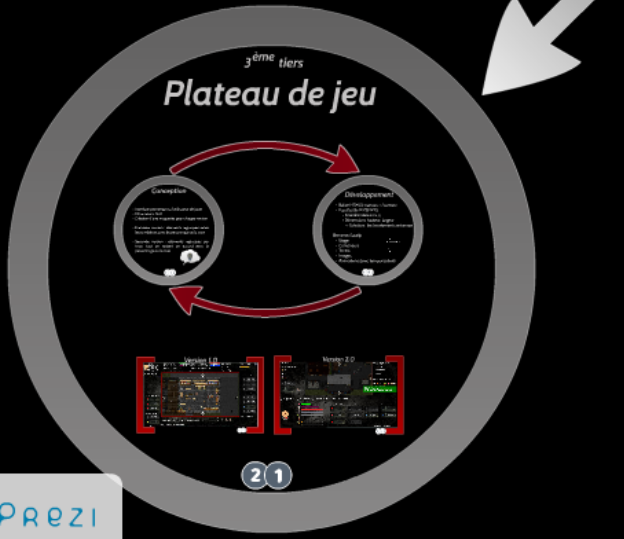
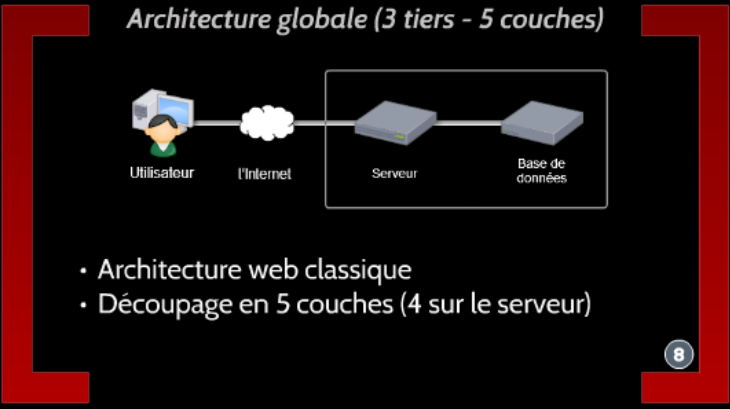
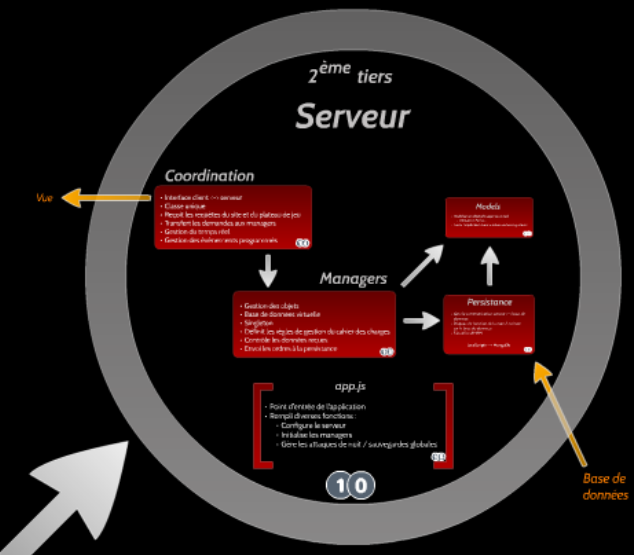
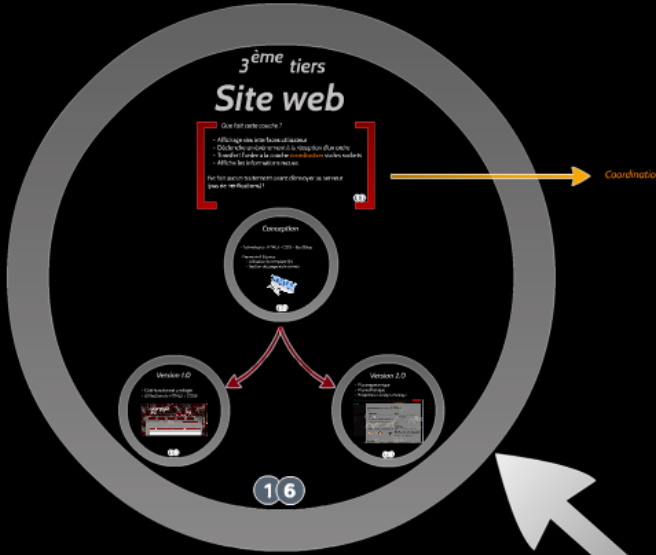
**Apptana**

**Sublime Text**

**Mongoose**

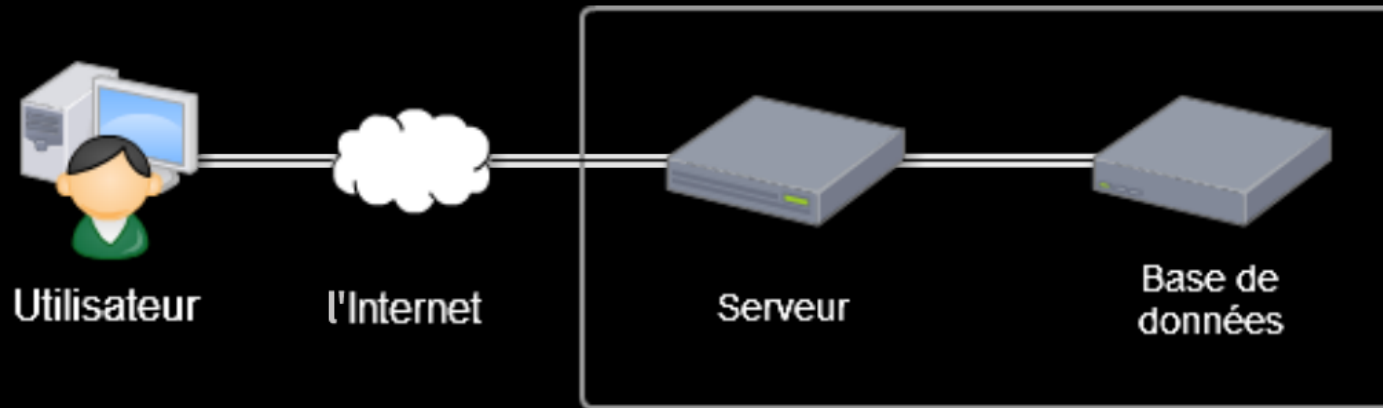
3

# Travail réalisé





## Architecture globale (3 tiers - 5 couches)



- Architecture web classique
- Découpage en 5 couches (4 sur le serveur)

1<sup>er</sup> tiers

## Base de données

- Sauvegarde permanente des données
- Communication avec le serveur très rapide
- Schémas initialisés au lancement du serveur
- Facilité de prise en main



# 2<sup>ème</sup> tiers Serveur

## Coordination

- Interface client <-> serveur
- Classe unique
- Reçoit les requêtes du site et du plateau de jeu
- Transfert les demandes aux managers
- Gestion du temps réel
- Gestion des événements programmés

14

## Models

- Modélise un objet physique ou virtuel
- Utilisateur, Score...
- Toute l'application tourne autour de leurs gestions

13

## Managers

- Gestion des objets
- Base de données virtuelle
- Singleton
- Définit les règles de gestion du cahier des charges
- Contrôle les données reçues
- Envoi les ordres à la persistance

13

## Persistance

- Gère la communication serveur <-> base de données
- Dispose de fonction de lecture / écriture sur la base de données
- Fait office d'ORM

JavaScript <-> MongoDB

14

## app.js

- Point d'entrée de l'application
- Rempli diverses fonctions :
  - Configure le serveur
  - Initialise les managers
  - Gère les attaques de nuit / sauvegardes globales

15

10

Vue

Base de données

# *Persistance*

- Gère la communication serveur <-> base de données
- Dispose de fonction de lecture / écriture sur la base de données
- Fait office d'ORM

JavaScript <--> MongoDB

# Models

- Modélise un objet physique ou virtuel
  - Utilisateur, Score...
- Toute l'application tourne autour de leurs gestions



# Managers

- Gestion des objets
- Base de données virtuelle
- Singleton
- Définit les règles de gestion du cahier des charges
- Contrôle les données reçues
- Envoi les ordres à la persistance

# Coordination

- Interface client <-> serveur
- Classe unique
- Reçoit les requêtes du site et du plateau de jeu
- Transfert les demandes aux managers
- Gestion du temps réel
- Gestion des événements programmés

14



M



# *app.js*

- Point d'entrée de l'application
- Rempli diverses fonctions :
  - Configure le serveur
  - Initialise les managers
  - Gère les attaques de nuit / sauvegardes globales

# 2<sup>ème</sup> tiers Serveur

## Coordination

- Interface client <-> serveur
- Classe unique
- Reçoit les requêtes du site et du plateau de jeu
- Transfert les demandes aux managers
- Gestion du temps réel
- Gestion des événements programmés

14

## Models

- Modélise un objet physique ou virtuel
- Utilisateur, Score...
- Toute l'application tourne autour de leurs gestions

13

## Managers

- Gestion des objets
- Base de données virtuelle
- Singleton
- Définit les règles de gestion du cahier des charges
- Contrôle les données reçues
- Envoi les ordres à la persistance

13

## Persistance

- Gère la communication serveur <-> base de données
- Dispose de fonction de lecture / écriture sur la base de données
- Fait office d'ORM

JavaScript <-> MongoDB

14

## app.js

- Point d'entrée de l'application
- Rempli diverses fonctions :
  - Configure le serveur
  - Initialise les managers
  - Gère les attaques de nuit / sauvegardes globales

15

10

Vue

Base de données

3<sup>ème</sup> tiers

# Site web

Que fait cette couche ?

- Affichage des interfaces utilisateur
- Déclenche un événement à la réception d'un ordre
- Transfert l'ordre à la couche **coordination** via les sockets
- Affiche les informations reçues

Ne fait aucun traitement avant d'envoyer au serveur  
(pas de vérifications) !

17

## Conception

- Technologies : HTML5 - CSS3 - Bootstrap
- Framework Express :
  - Utilisation du template EJS
  - Gestion des pages côté serveur



18

## Version 1.0

- Côté fonctionnel privilégié
- Utilisation de HTML5 / CSS3



19

## Version 2.0

- Plus ergonomique
- Plus esthétique
- Responsive design envisagé



20

16



Coordination



# Site web

## *Que fait cette couche ?*

- Affichage des interfaces utilisateur
- Déclenche un événement à la réception d'un ordre
- Transfert l'ordre à la couche **coordination** via les sockets
- Affiche les informations reçues

Ne fait aucun traitement avant d'envoyer au serveur  
(pas de vérifications) !

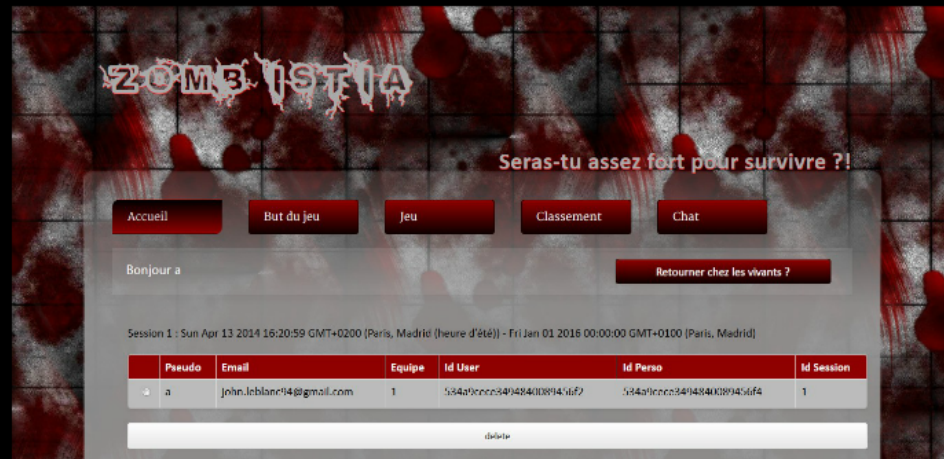
# Conception

- Technologies : HTML5 - CSS3 - BootStrap
- Framework Express :
  - Utilisation du template EJS
  - Gestion des pages coté serveur



# Version 1.0

- Côté fonctionnel privilégié
- Utilisation de HTML5 / CSS3





# • Utilisation de HTML5 / CSS3

## ZOMBISTIA

Seras-tu assez fort pour survivre ?!

Accueil

But du jeu

Jeu

Classement

Chat

Bonjour a

Retourner chez les vivants ?

Session 1 : Sun Apr 13 2014 16:20:59 GMT+0200 (Paris, Madrid (heure d'été)) - Fri Jan 01 2016 00:00:00 GMT+0100 (Paris, Madrid)

	Pseudo	Email	Equipe	Id User	Id Perso	Id Session
<input type="radio"/>	a	john.leblanc94@gmail.com	1	534a9cece3494840089456f2	534a9cece3494840089456f4	1

delete



# Version 2.0

- Plus ergonomique
- Plus esthétique
- Responsive design envisagé

Zomb'istIA

Heure du serveur : 21:36:13 Prochaine attaque à : 19:14:00 [Retour chez les devs](#)

## Tutoriel

Bienvenue à vous, noble survivant ! Si vous êtes ici, c'est que vous avez sûrement lu la page des règles du jeu. Si ce n'est pas le cas, nous vous invitons à y jeter un œil avant de commencer. Sur cette page de tutoriel, vous trouverez de nombreuses informations sur la façon de jouer à Zomb'istIA. Voici les différents points que nous aborderons au cours de ce tutoriel.

### Les personnages

Avant tout, il faut que vous choisissiez de rejoindre une des trois équipes existantes.

Les **ADI** sont souvent à la pointe de la technologie, et plutôt réfléchis. Les **INNO** sont souvent pleins de bonnes idées, et très sociables. Les **CSF** privilégient la qualité et ont des méthodes infallibles pour arriver à leur but.

Vous devez ensuite enseigner une compétence à votre personnage. Essayons de voir quels sont les points forts et les points faibles de chaque compétence :

- **Brute** : Ses points forts sont l'attaque et la défense. Ses points faibles sont la fouille et la cache, ainsi que le nombre de zombie qui l'empêchera de faire des actions.
- **Chercheur** : Son point fort est la fouille. Ses points faibles sont l'attaque et la défense.
- **Explorateur** : Ses points forts sont la cache, les déplacements, et le faible nombre de zombies qui l'empêcheront de faire des actions. Ses points faibles sont l'attaque et la défense.

### Les items

De nombreux items sont présents dans l'univers de Zomb'istIA. Vous pouvez découvrir 5 types d'items dans ce jeu :

- **Armes** : Elles procurent des points d'attaque supplémentaires à votre personnage s'il en est équipé.
- **Armures** : Elles procurent des points de défense supplémentaires à votre personnage s'il en est équipé.
- **Potions** : En consommant reboostera l'état de votre personnage, en fonction de la couleur.
- **Points de vie**
- **Points d'action**
- **Points de déplacement**
- **Nourriture** : En consommant permet à votre personnage de rester en forme.
- **Objets de défense** : Ces items sont rares et bien cachés. Vous devez les ramasser et les déposer dans la zone sûre de votre équipe pour marquer des points.

Remarque : Souvent, plus un item est lourd, plus il a de valeur.

### Le plateau de jeu

Le nom du plateau de Zomb'istIA se situe sur le plateau de jeu. Il est accessible depuis l'onglet 'JEU' du site. Voici deux exemples d'exemples.

20

[Accueil](#)

[Règles](#)

[Tutoriel](#)

[Jeu](#)

[Tchats](#)

[Classement](#)

## Tutoriel

**Bienvenue à vous, noble survivant !** Si vous êtes ici, c'est que vous avez sûrement lu la page des règles du jeu.

Si ce n'est pas le cas, nous vous invitons à y jeter un œil avant de commencer.

Sur cette page de tutoriel, vous trouverez de nombreuses informations sur la façon de jouer à Zomb'IstiA. Voici les différents points que nous aborderons au cours de ce tutoriel.

### Les personnages

Avant tout, il faut que vous choisissiez de rejoindre une des trois équipes existantes.

Les **AGI** sont souvent à la pointe de la technologie, et plutôt réfléchis.

Les **INNO** sont souvent pleins de bonnes idées, et très sociables.

Les **QSF** privilégient la qualité et ont des méthodes infailibles pour arriver à leur but.








Vous devez ensuite enseigner une compétence à votre personnage. Essayons de voir quels sont les points forts et les points faibles de chaque compétence :

- **Brute** : Ses points forts sont l'attaque et la défense. Ses points faibles sont la fouille et la cache, ainsi que le nombre de zombie qui l'empêchera de faire des actions.
- **Chercheur** : Son point fort est la fouille. Ses points faibles sont l'attaque et la défense.
- **Explorateur** : Ses points forts sont la cache, les déplacements, et le faible nombre de zombies qui l'empêcheront de faire des actions. Ses points faibles sont l'attaque et la défense.



### Les items

De nombreux items sont présents dans l'univers de Zomb'IstiA. Vous pouvez découvrir 5 types d'items dans ce jeu :

- **Armes**  : Elles procurent des points d'attaque supplémentaires à votre personnage s'il en est équipé.
- **Armures**  : Elles procurent des points de défense supplémentaires à votre personnage s'il en est équipé.
- **Potions** : En consommant reboostera l'état de votre personnage, en fonction de la couleur.
  -  = Points de vie
  -  = Points d'action
  -  = Points de déplacement
- **Nourriture**  : En consommant permet à votre personnage de rester en forme.
- **Objets de défense**  : Ces items sont rares et bien cachés. Vous devez les ramasser et les déposer dans la zone sûre de votre équipe pour marquer des points.

*Remarque : Souvent, plus un item est lourd, plus il a de valeur.*

### Le plateau de jeu

Le cœur de l'action de Zomb'IstiA se situe sur le plateau de jeu. Il est accessible depuis l'onglet JEU du site. Voyons donc comment s'en servir...

## Conception

- Technologies : HTML5 - CSS3 - BootStrap
- Framework Express :
  - Utilisation du template EJS
  - Gestion des pages coté serveur



18

## Version 1.0

- Côté fonctionnel privilégié
- Utilisation de HTML5 / CSS3



19

## Version 2.0

- Plus ergonomique
- Plus esthétique
- Responsive design envisagé



20

3<sup>ème</sup> tiers

# Plateau de jeu

## Conception

- Interface permettant à l'utilisateur de jouer
- Dimensions 16/9
- Création d'une maquette pour chaque version
- Première version : éléments regroupés selon leurs relations avec le personnage ou la case
- Seconde version : éléments regroupés par type, tout en restant en accord avec le personnage ou la case



22

## Développement

- Balise HTML5 <canvas> /</canvas>
  - Pas d'outils WYSIWYG
  - Coordonnées en x, y
  - Dimensions hauteur, largeur
  - Solution : lier les éléments entre eux
- Elements Easels
- Stage
  - Conteneurs
  - Textes
  - Images
  - Animations (avec temporisation)



23



24



25

21



# Conception

- Interface permettant à l'utilisateur de jouer
- Dimensions 16:9
- Création d'une maquette pour chaque version
- Première version : éléments regroupés selon leurs relations avec le personnage ou la case
- Seconde version : éléments regroupés par type, tout en restant en accord avec le personnage ou la case

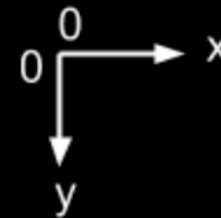


# Développement

- Balise HTML5 `<canvas>` `</canvas>`
- Pas d'outils WYSIWYG
  - Coordonnées en x, y
  - Dimensions hauteur, largeur
  - > Solution : lier les éléments entre eux

## Elements Easeljs

- Stage
- Conteneurs
- Textes
- Images
- Animations (avec temporisation)





# Version 1.0

**Brute**  
Zombies : 2  
Attaque x 2  
Défense x 2  
Cache x 0.5  
Fouille x 1

Le 23/03 à 13:36  
Fouille rapide ratée

Points de vie :  Points d'action :   
Points de faim :  Points de mouvement :   
Points d'attaque : 2 Points de défense : 34

Arme équipée : Armure équipée :   
Inventaire du perso :

Sun Mar 23 2014 - 14:15:40 : Grâce à votre cachette, vous av...

Passer en Mode :  
 FOUILLE  
 CACHE  
 DEFENSE

Afficher liste :  
 JOUEURS  
 MESSAGES

Alliés dans la salle : 0 Proba de Cache :   
Ennemis dans la salle : 0 Proba de Trouver item :   
Zombies dans la salle : 0 Case en cours : Zone sûre des AGI  
Ouf ici on ne risque rien !

Actions sur le sac :  
 CONSOMMER  
 EQUIPER  
 DESEQUIPER

Actions sur la case :  
 DEPOSER  
 RAMASSER  
 FOUILLE RAPIDE  
 TUER ZOMBIE(S)

Objets présents :



## Conception

- Interface permettant à l'utilisateur de jouer
- Dimensions 16:9
- Création d'une maquette pour chaque version
  
- Première version : éléments regroupés selon leurs relations avec le personnage ou la case
  
- Seconde version : éléments regroupés par type, tout en restant en accord avec le personnage ou la case



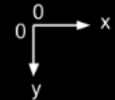
2 2

## Développement

- Balise HTML5 `<canvas>` `</canvas>`
- Pas d'outils WYSIWYG
  - Coordonnées en x, y
  - Dimensions hauteur, largeur
- > Solution : lier les éléments entre eux

### Elements Easeljs

- Stage
- Conteneurs
- Textes
- Images
- Animations (avec temporisation)



2 3

# Version 2.0

Hall d'entree

FORCES EN PRESENCE

- ★ 0
- ☠ 0
- 👤 5

JOUEURS

MESSAGES 4

OBJETS DANS LE SAC

EQUIPABLES

CONSOMMABLES

DEFENSES

OBJETS DANS LA CASE

EQUIPABLES

CONSOMMABLES

Toge de Link  
Valeur : 2 - Poids : 4  
Elle est ideale pour les petits aventuriers

Fri Apr 18 2014 - 17:32:35 : Attaque de nuit ! Vous avez subi : 16 points de dégâts par 4 zombies !  
Que s'est t'il passé ici, et ces marques au sol ?

MODES

SAC

CASE

FOUILLE

CONSOMMER

DEPOSER

FOUILLE RAPIDE

CACHE

EQUIPER

RAMASSER

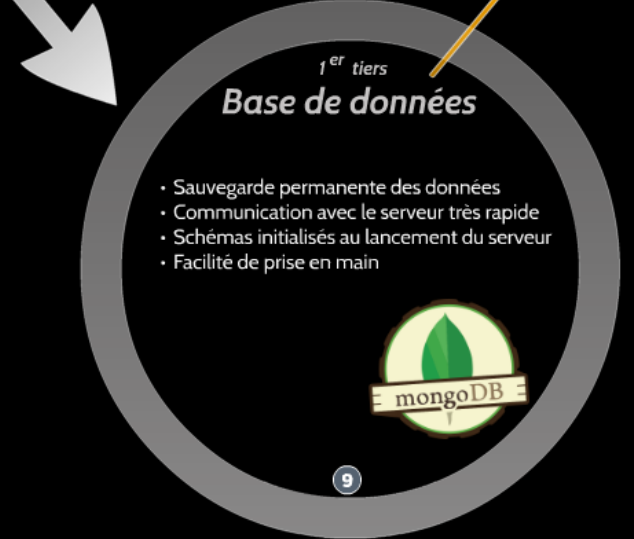
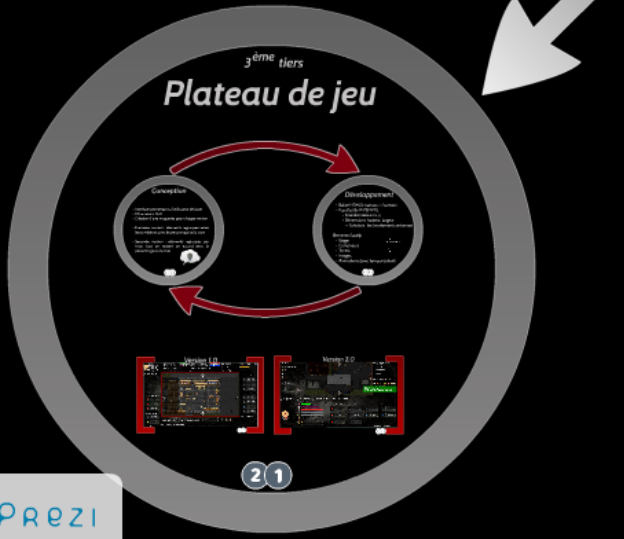
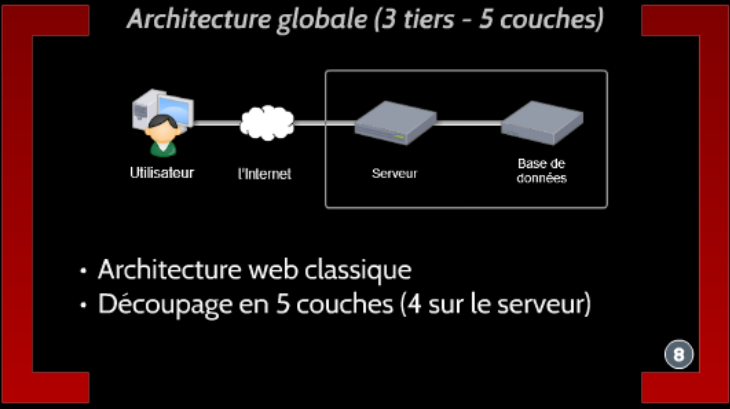
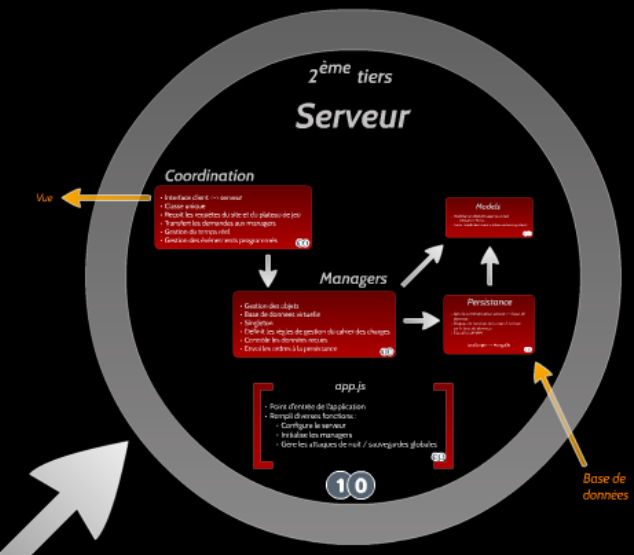
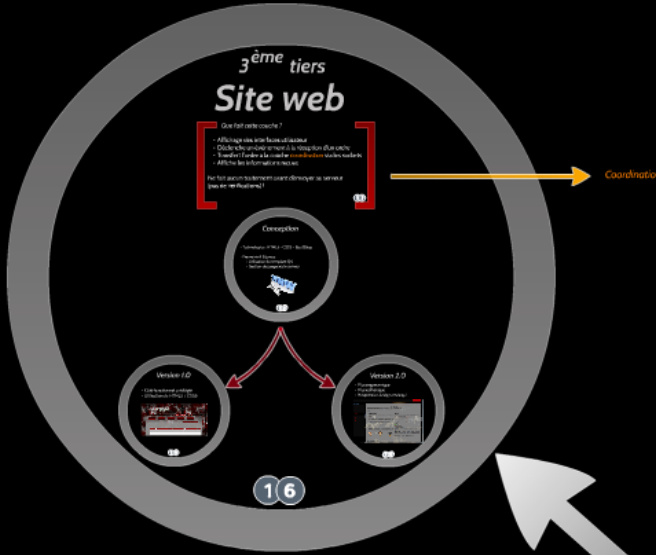
TUER ZOMBIE(S)

DEFENSE

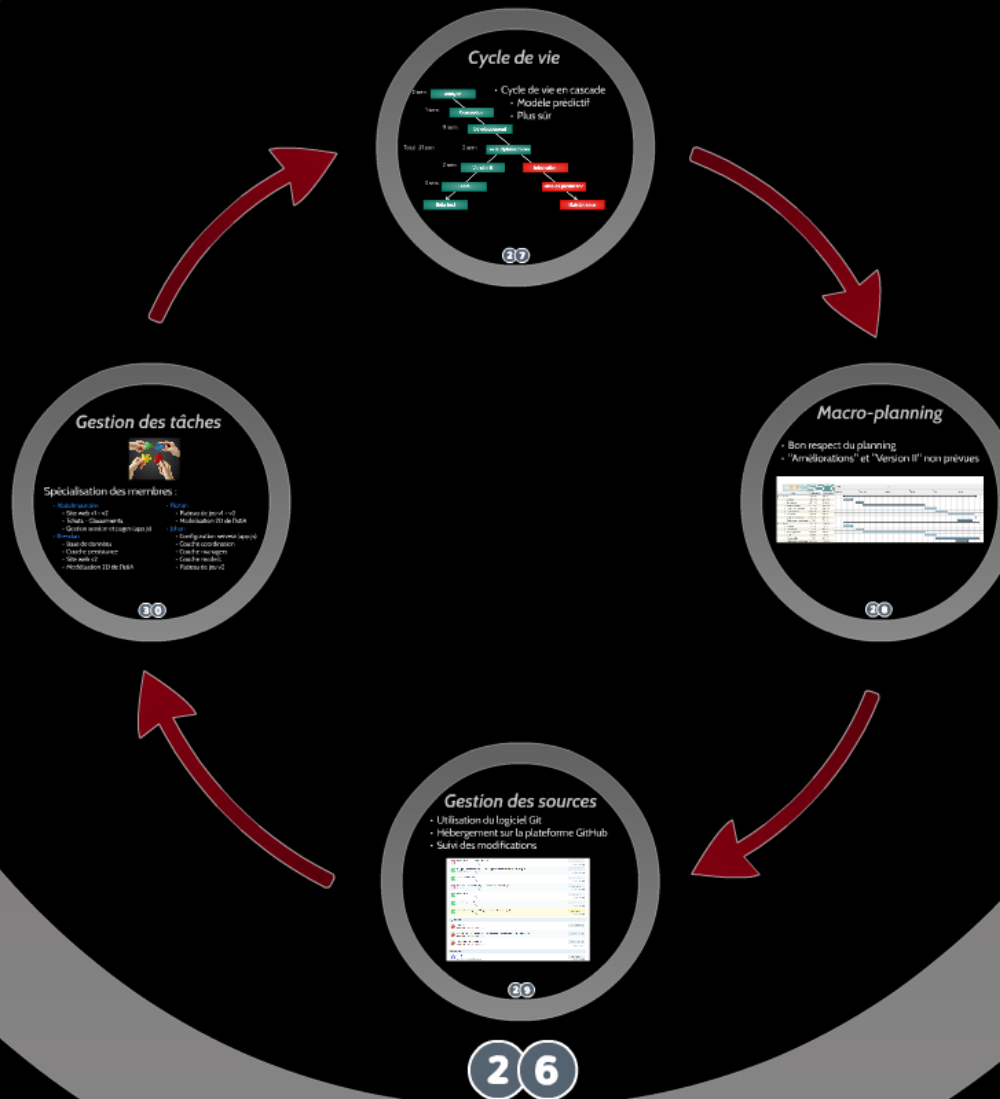
DESEQUIPER

Serveur lancé le 18/04 à 20h13

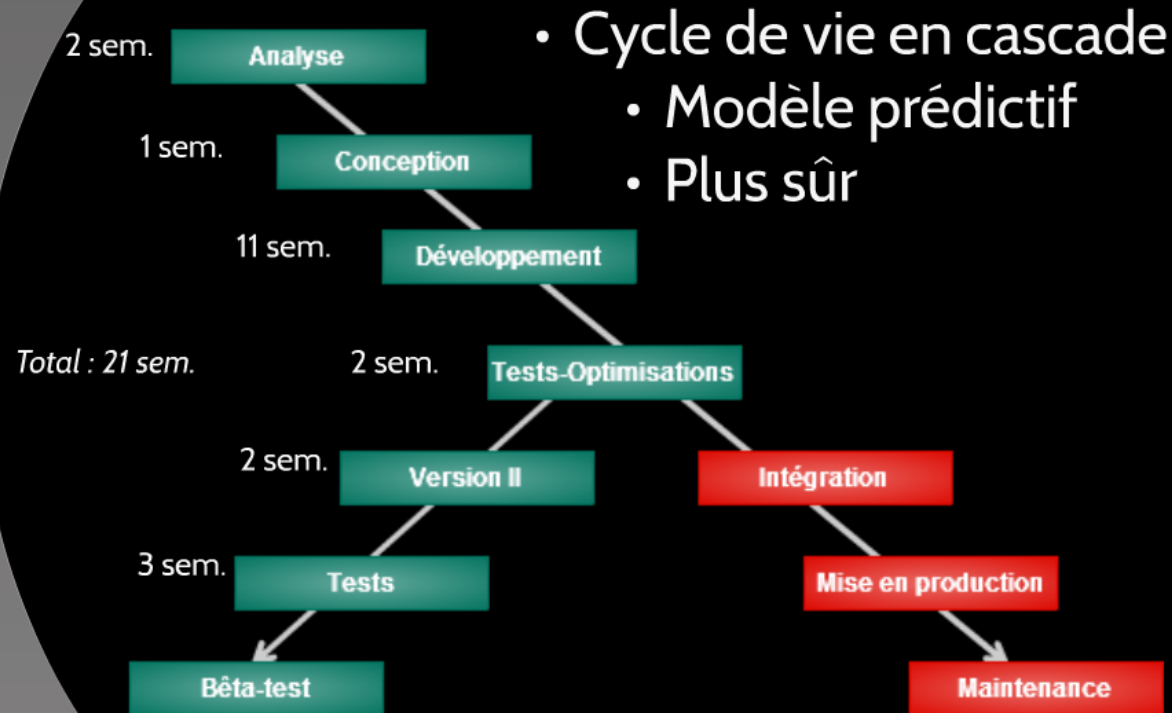
# Travail réalisé



# Gestion de Projet

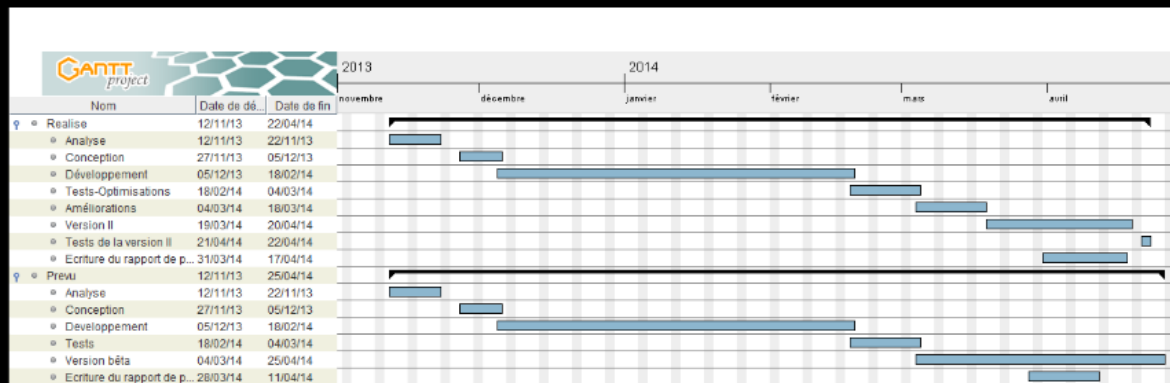


# Cycle de vie



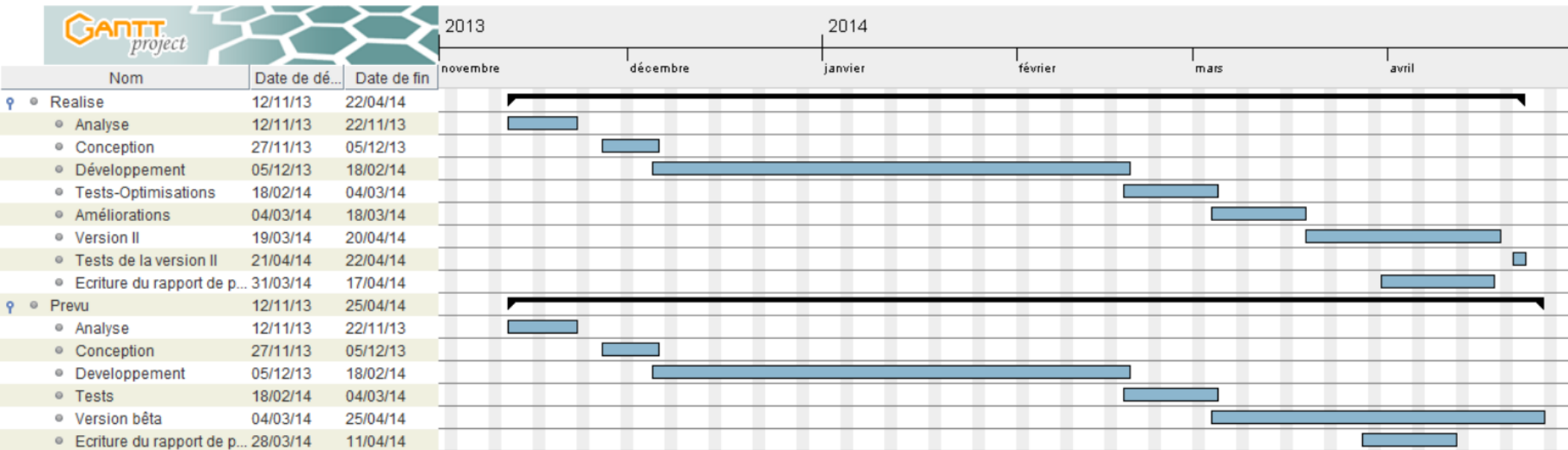
# Macro-planning

- Bon respect du planning
- "Améliorations" et "Version II" non prévues



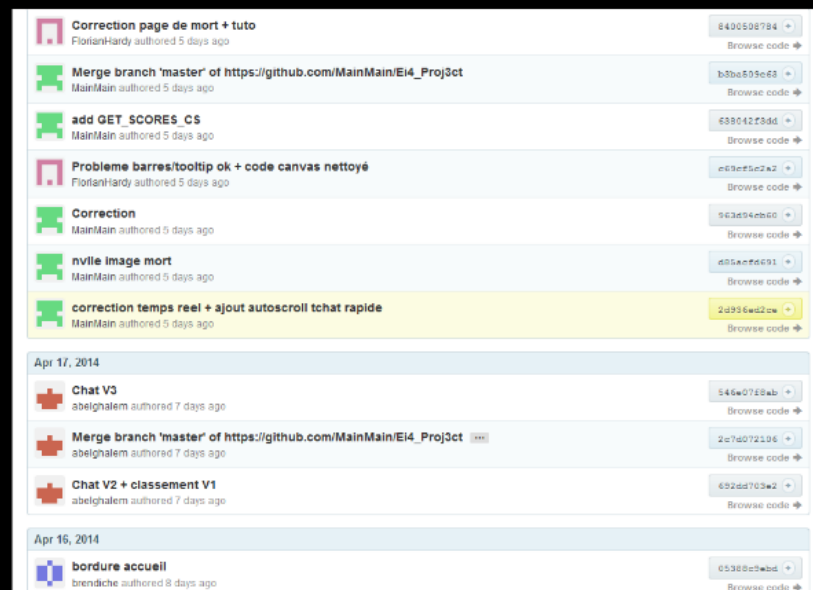


- Bon respect du planning
- "Améliorations" et "Version II" non prévues



# Gestion des sources

- Utilisation du logiciel Git
- Hébergement sur la plateforme GitHub
- Suivi des modifications







### Correction page de mort + tuto

FlorianHardy authored 5 days ago

8400508784

[Browse code](#)



### Merge branch 'master' of https://github.com/MainMain/Ei4\_Proj3ct

MainMain authored 5 days ago

b3ba509e63

[Browse code](#)



### add GET\_SCORES\_CS

MainMain authored 5 days ago

638042f3dd

[Browse code](#)



### Probleme barres/tooltip ok + code canvas nettoyé

FlorianHardy authored 5 days ago

c69cf5c2a2

[Browse code](#)



### Correction

MainMain authored 5 days ago

963d94cb60

[Browse code](#)



### nvlle image mort

MainMain authored 5 days ago

d85acfd691

[Browse code](#)



### correction temps reel + ajout autoscroll tchat rapide

MainMain authored 5 days ago

2d936ed2ce

[Browse code](#)

Apr 17, 2014



### Chat V3

abelghalem authored 7 days ago

546e07f8ab

[Browse code](#)



### Merge branch 'master' of https://github.com/MainMain/Ei4\_Proj3ct

abelghalem authored 7 days ago

2c7d072106

[Browse code](#)



### Chat V2 + classement V1

abelghalem authored 7 days ago

692dd703e2

[Browse code](#)

Apr 16, 2014



### bordure accueil

brendiche authored 8 days ago

05388c9ebd

[Browse code](#)

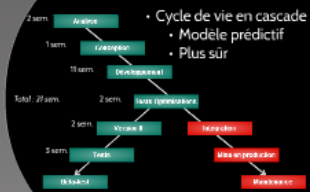
# Gestion des tâches



## Spécialisation des membres :

- **Abdelmounaim**
  - Site web v1 - v2
  - Tchats - Classements
  - Gestion session et pages (app.js)
- **Brendan**
  - Base de données
  - Couche persistance
  - Site web v2
  - Modélisation 2D de l'IstiA
- **Florian**
  - Plateau de jeu v1 - v2
  - Modélisation 2D de l'IstiA
- **Johan**
  - Configuration serveur (app.js)
  - Couche coordination
  - Couche managers
  - Couche models
  - Plateau de jeu v2

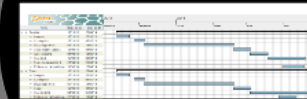
## Cycle de vie



27

## Macro-planning

- Bon respect du planning
- "Améliorations" et "Version II" non prévues



28

## Gestion des sources

- Utilisation du logiciel Git
- Hébergement sur la plateforme GitHub
- Suivi des modifications



29

## Gestion des tâches



### Spécialisation des membres :

- Abdelmoumin
  - Site web v1 - v2
  - Filtrats - Classements
  - Gestion session et pages (app.js)
- Brendan
  - Base de données
  - Couches persistence
  - Site web v2
  - Modélisation 2D de l'ItaA
- Florian
  - Plateau de jeu v1 - v2
  - Modélisation 2D de l'ItaA
  - Jargon
  - Configuration serveur (app.js)
  - Couches coordination
  - Couches managers
  - Couches modes
  - Plateau de jeu v2

30

# Résultats du projet

- Cahier des charges respecté
- Jeu fonctionnel
- Déployé et accessible
- Bêta lancé depuis 3 jours :
  - 15aine de joueurs
  - Plaît aux testeurs (beau et original)
  - Pas de plantage du serveur
  - Quelques bugs sans gravité



# *Démonstration*



3 2

# Difficultés rencontrées

- Plateau de jeu sous Google Chrome



- Port d'écoute du serveur



- Téléchargement des images de l'application

- Jouabilité sur tablettes et mobiles



- Problèmes sur le serveur



# Améliorations possibles

- 5 étages supplémentaires
- Fonctionnement de l'ascenseur, avec choix d'étage
- Système de clés
- Équipements supplémentaire (casque, gants, bottes...)
- Capturer des zombies
- Désactiver les tchats dans l'amphi E
- Système de "news"
- Forums, publications des joueurs sur le site





## *Conclusion*

- Application fonctionnelle
- Problèmes surmontés grâce à :
  - Gestion de projet  
(respect du planning, définitions des tâches)
  - Architecture solide et découpée
- Apprentissage technologique et humain

***Fin de la présentation,  
Merci de votre attention !***

## Introduction

Jeu de rôle multijoueur par navigateur web en temps réel et persistant  
ORPG - "Online Role Playing Game"

Objectifs doubles  
• Pédagogiques  
• Produits

Contraintes  
• Temps : 5 mois de travail (5 heures par semaine)  
• Technologies : Node.js, HTML5, Canvas et MongoDB  
• Non financières

# ZOMB'ISTIA

## Jeu multijoueur temps réel en ligne

Abdelmounaim BELGHALEM

Johan BRUN

Brendan GOUIN

Florian HARDY

### Fonctionnalités

- Site web
- Jeu
- Tchats
- Classement
- Page des scores
- Zone administration
- Diriger un personnage
- Explorer l'isole en 2D
- Interactions avec
- Objets
- Cases
- Zombies
- Jeux de
- Attaque de nuit

2

### Technologies



3

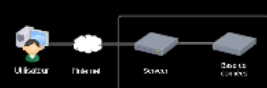
### Site web



10

## Travail réalisé

### Architecture globale (3 tiers - 5 couches)



- Architecture web classique
- Découpage en 5 couches (4 sur le serveur)

### Serveur



10

### Plateau de jeu



20

### Base de données

- Sauvegarde permanente des données
- Communication avec le serveur très rapide
- Schémas initialisés au lancement du serveur
- Facilité de prise en main

10

### Gestion de Projet



### Résultats du projet

- Galerie de 10 jeux vidéos
- 100 fonctionnalités
- 1000 heures de travail
- 1000 heures de travail
- 1000 heures de travail
- 1000 heures de travail

20

### Démonstration

20

### Difficultés rencontrées

- Problèmes de serveur
- Problèmes de serveur
- Problèmes de serveur
- Problèmes de serveur

20

### Améliorations possibles

- Améliorations possibles
- Améliorations possibles
- Améliorations possibles
- Améliorations possibles

20

### Conclusion

- Application fonctionnelle
- Problèmes surmontés grâce à :
  - Gestion de projet (respect du planning, définitions des tâches)
  - Architecture solide et découpée
- Apprentissage technologique et humain

20



EI4 - AGI - 2013/2014