



lpcisiie

atelier 2 / 6-11.02.2017

Déroulement & organisation



■ semaine 6>11.02 : déroulement de l'atelier

- par équipe de 4
- travail en salles 503/505

■ semaine 13>17.02 : évaluation

- évaluation individuelle
- questions écrites

les objectifs

- réaliser une application web moderne basée sur :
 - une api RESTful : uri, ressources, actions
 - une webapp javascript basée sur un framework mvc

les objectifs

- appliquer les notions, principes et techniques abordées dans les modules de prog. serveur et client :
 - principes REST, représentation JSON, HTTP,
 - architecture et organisation du code
 - sécurité et authentification
 - appels asynchrones, templates, webapp mvc

le rendu

■ samedi 11/02, 12h

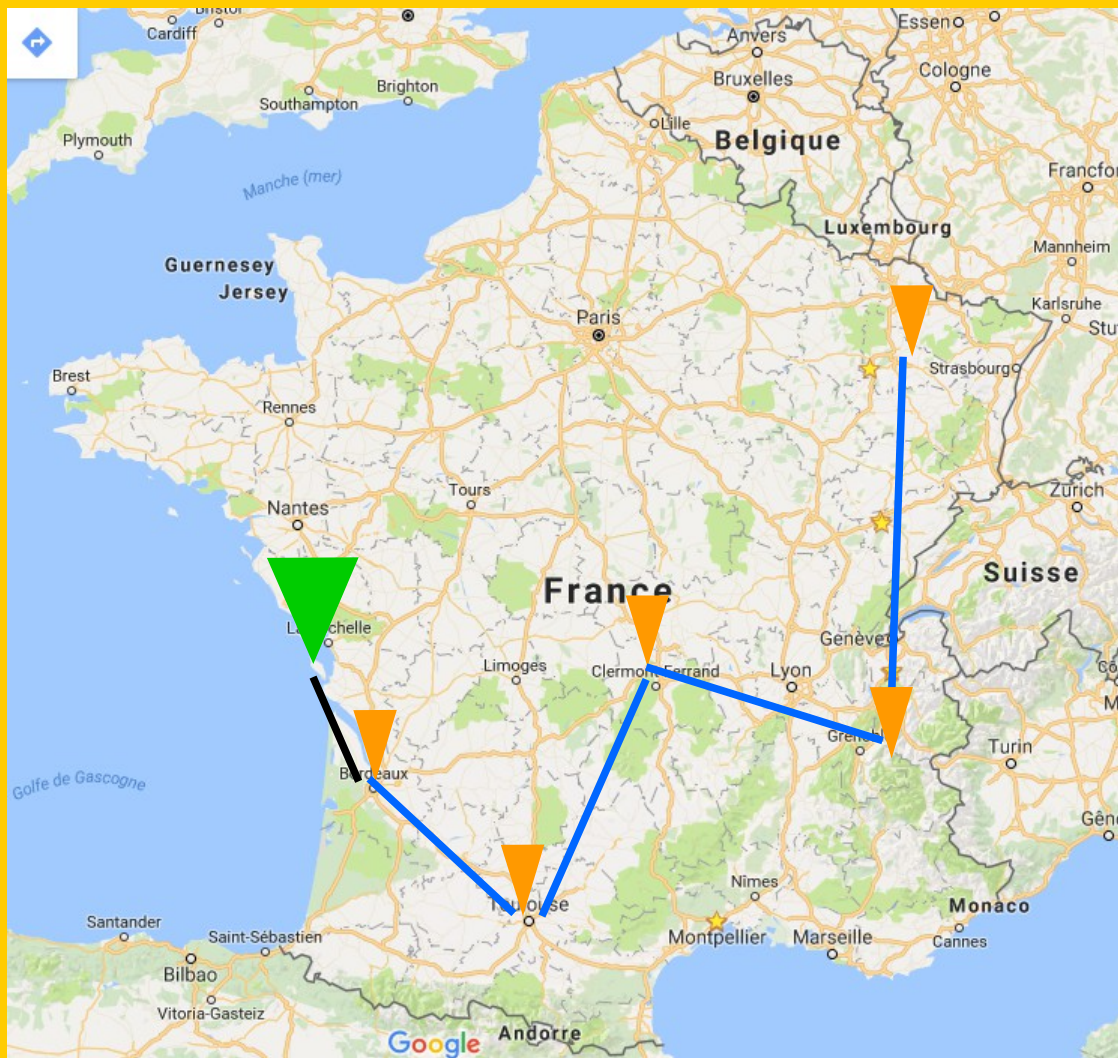
- url dépôt git
- le dépôt doit contenir une README avec les directives d'installation de l'application
- sql pour la création/initialisation de la base
- une documentation de l'api permettant de la tester indépendamment de la webapp
- un tableau de bord du projet, précisant la liste des stories et leur état



lpcisiie

atelier 2 / 6-11.02.2017

cahier des charges



FindYourWay

Un jeu amusant où tu dois suivre ton chemin sur la carte pour collecter des indices et trouver ta destination finale

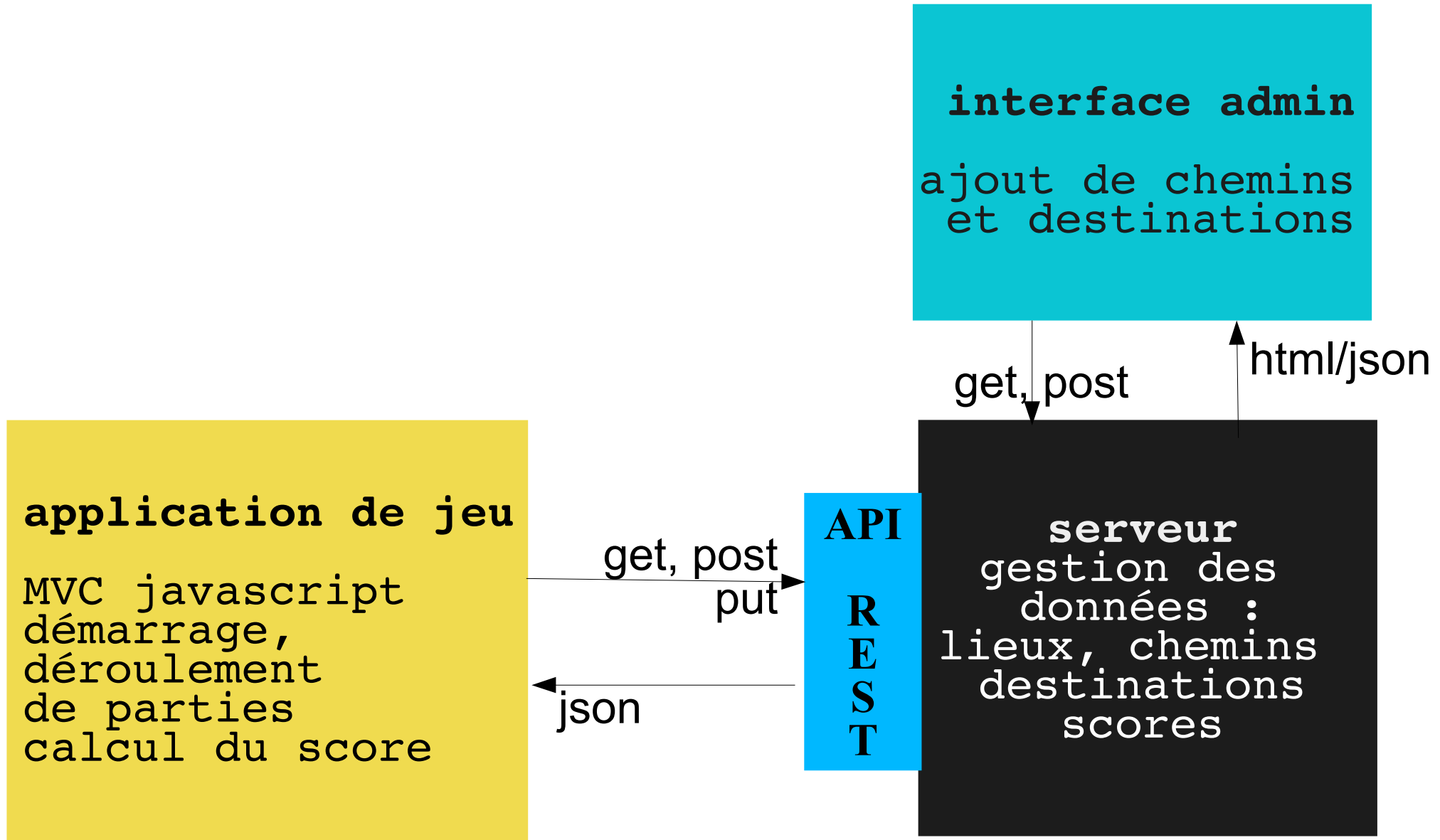
principes du jeu

- 5 lieux à placer successivement sur la carte, décrits par un texte et/ou une photo, pour former un chemin
- chaque lieu bien placé permet de collecter un indice
- l'ensemble des indices permet de découvrir la destination finale qu'il faut placer sur la carte

règles du jeu

- les lieux doivent être placés dans l'ordre du chemin,
- chaque lieu placé à une distance inférieure à une valeur fixée permet de collecter un indice,
- une fois le chemin entièrement construit, le joueur doit placer la destination finale,
- le score final est calculé en fonction de la précision de placement de cette destination

architecture



l'application de jeu

- webapp js 100 %, accès au serveur au travers d'une api RestFul
- Déroulement :
 1. création d'une partie, récupération d'un token
 2. récupération des éléments de chemin : indication, coordonnées
 3. récupération de la destination : indices + coordonnées
 4. déroulement du jeu **sans interaction** avec le backend
 5. fin de la partie : calcul du score, récupération des infos complémentaires destination, mise à jour de la partie

interface admin

- webapp js avec api **OU** appli serveur avec templates
- (php | java)
- création de nouveaux éléments de chemin : indication + géolocalisation
- ajout de destinations : indices + géolocalisation + infos finales
- nécessite une inscription + authentification

fonctionnalités minimales

- application jouable, avec 5 éléments de chemin par partie, 1 seul niveau de jeu,
- distance D pour le calcul des points identique pour toutes les destinations,
- backoffice permettant d'ajouter des éléments de chemin et des destinations ; pas d'authentification pour la partie backend,
- les éléments de chemin et les indices sont textuels : pas d'images,

extensions

- inscription et authentification pour accéder à la partie backend,
- images dans les indications de chemin,
- images dans les indices,
- décompte du temps pour les réponses,
- suspendre 1 partie pour la reprendre plus tard. Stockage des données local (LocalStorage),
- différents niveaux de jeu, qui peuvent se différencier par :
 - la difficulté pour obtenir un indice
 - distance D pour valider la destination finale,
- accès au tableau des meilleurs scores

questions ?

bon travail !

