

Rapport TERL - Emploi du temps

Compléter le titre (celui du sujet)

Hind, Teiki, Adel, Jérémy

Ajouter les noms et mettre dans
l'ordre alphabétique

Université de Montpellier ~~UM2~~



Encadrant: ~~J~~ Hinde Bouziane

Mai 2017

Contents

Contents	2
1 Introduction	3
2 Préparation du projet	5
2.1 Les choix techniques	5
2.2 Les différentes fonctionnalités	6
2.3 Modéliser les différentes problématiques	6
2.4 Organiser l'espace de travail	6
2.5 Répartition des tâches	6
3 Déroulement du projet	7
3.1 Difficultés	7
3.2 Les contraintes	7
3.3 Les groupements	7
3.4 Les complétions	7
3.5 Placer et des déplacer des cours	7
3.6 L'interface graphique	7
4 Conclusion	9
5 Annexes	11

Chapter 1

Introduction

Dans le cadre d'un projet de deuxième année de licence informatique, nous devons réaliser un générateur d'emploi du temps destiné ~~aux professeurs de la FAC~~.....
Aux enseignants du département informatique de l'UM.

Chapter 2

Préparation du projet

2.1 Les choix techniques



Dans le cadre d'un projet, les choix techniques doivent dépendre de l'environnement de celui-ci. Dans notre cas, nous devons réaliser une plateforme web où les professeurs vont pouvoir rentrer les données utiles à la génération de l'emploi du temps. Le générateur, lui pouvait être détaché de cette plateforme. Cependant, nous avons préféré faire de la plateforme web l'élément central où tout serait réuni/ dessus, car cela serait plus simple pour l'utilisateur, et pour nous (cela occasionne moins de problématiques comme par exemple l'authentification, la distribution d'un logiciel...).

Par conséquent, nous avons décidé d'utiliser le langage PHP car sur le serveur web local à la FAC c'est le langage qui est utilisé. De plus, PHP a été spécialement conçu pour le web, il est très bien documenté, c'est l'un des langages du web le plus performant, et on peut utiliser la paradigme d'objet.

Aussi, nous avons décidé d'utiliser une base de donnée MySQL, car c'est ce qui permet de stocker, récupérer et représenter des données très facilement sachant qu'il y a un module intégré à PHP pour cela. Et vient s'ajouter à cela la décision d'avoir conçu un ORM le plus simple que possible afin d'automatiser la modélisation des données. Puisqu'il était essentiel pour nous d'utiliser le plus que possible la programmation orienté objet, un ORM était indispensable. Cependant nous avons eu besoin de quelque chose de simple et performant, c'est pourquoi nous avons décidé de le concevoir par nous-même plutôt que d'utiliser des outils comme Doctrine qui sont compliqué et lourd.

Enfin, pour le développement de la plateforme web côté client, il est incontournable d'utiliser le HTML, le CSS, et le Javascript. Cependant, pour des soucis d'ergonomie, de rapidité, et de ~~fluidité~~, nous avons décidé de privilégié au maximum l'utilisation du Javascript en utilisant l'Ajax et le JSON. Ainsi, cela évite d'utiliser un système de template côté client qui sont souvent lourd pour un serveur. Puisque la génération de l'emploi du temps peut demander une charge importante au serveur, il était important de limiter au maximum les charges que l'on peut lui occasionner avec des outils aussi conséquent. C'est pour cela, que nous avons décidé d'utiliser AngularJS qui permet de gérer plus facilement un environnement comme celui-ci.

2.2 Les différentes fonctionnalités

Maintenant que nous savons que nous allons utiliser une plateforme web pour notre projet, nous allons voir les différentes fonctionnalités qu'il fallait implanter afin de pouvoir répondre aux besoins du projet. Pour cela, nous avons commencé par les énumérer :

- Possibilité pour les responsables d'UE de rentrer leurs indisponibilités
- Possibilité pour les responsables d'UE d'indiquer des créneaux horaires de préférences
- Possibilité pour les responsables d'UE d'indiquer que des séances doivent se succéder
- Possibilité pour les responsables de promotions de ~~rentrer~~ des indisponibilités pour une promotion ou un groupe ~~(par exemple quand il y a des cours d'anglais)~~
- Possibilité pour les responsables de promotions d'indiquer pour une promotion si elle peut faire une journée 8h/20h complète ou non
- Garantir les pauses déjeuner

Dès lors, il y avait un problème qui se posait : comment rendre cela accessible aux utilisateurs et utilisable dans le cadre de la génération de l'emploi du temps

Ne pas utiliser le terme "on"

? Pour cela, ~~on~~ a décidé d'établir chacune des fonctionnalités en respectant 2 points précis :



- Donner un format paramétrable et utilisable pour la génération de l'emploi du temps
- Définir une interface simple et ergonomique pour chaque fonctionnalités



A partir de ce moment là, ~~on~~ a pu définir toute les fonctionnalités nécessaires aux besoins du projet. Pour commencer, un professeur pourra donner ses indisponibilités et ses créneaux horaires de préférences

Quelles problématiques ?

2.3 Modéliser les différentes problématiques

Le titre devrait alors être problématiques et modélisation



Après avoir vu tout ce qu'il faudra que le projet comporte, il était très important de bien saisir et représenter les différentes problématiques. Pour cela, nous avons décidé de réaliser des diagrammes UML, car cela permet de pouvoir de contextualiser un projet, représenter les données et comment elles sont structurées. Ainsi, cela permet une meilleure cohésion au sein d'un groupe, et d'éviter tout mal entendu avec le client.

Puisque nous utilisons une base de données, nous avons établi un diagramme entité-relation, et puisque nous utilisons la programmation orienté objet avec un ORM, nous avons aussi établi un diagramme de classe. Nous allons donc commencer par présenter le diagramme entité-relation :

[image] Explications....

~~On~~ va maintenant voir le diagramme de classe dont le but était de représenter l'emploi du temps afin de savoir comment ~~on~~ allait procéder pour réaliser le générateur, et comment ~~chaque~~ éléments sont liés :

[image] Explication

2.4 Organiser l'espace de travail

Parler de l'arborescence du projet, des outils utilisés, et de certaines pratiques de développement

Pas besoin de commencer à chaque fois avec : après avoir

C'est très bien de faire une transition mais, ici par exemple, elle peut être plus naturelle : pour réaliser notre outil, nous nous sommes réparties les tâches Vous lister les tâches en fonction de ce que vous avez décrits avant.

2.5 Répartition des tâches

~~Après avoir vu l'ensemble du travail à réaliser,~~ et comment était structurer notre projet, on a pu répartir les tâches pour optimiser au mieux le temps de travail. Pour cela, nous avons divisé le projet en 2 points : conception du générateur d'emploi du temps, et les fonctionnalités. Puisque nous sommes un groupe de 4 personnes, on a pu se répartir équitablement les tâches : 2 personnes sur la réalisation du générateur d'emploi du temps, et 2 personnes pour les fonctionnalités de la plateforme.

Chapter 3

Déroulement du projet

3.1 Difficultés

3.2 Les contraintes

...

3.3 Les groupements (enchaînement de cours)

...

3.4 Les complétions (séances virtuelles)

...

3.5 Placer et des déplacer des cours

...

3.6 L'interface graphique

...

Chapter 4

Conclusion

Parler de si le résultat est celui attendu Ce que ça nous a apporté intellectuelle-
ment

Chapter 5

Annexes

- Interface