Projet Journée pédagogique



Chyzhyk Aleh – CID3b

ETML

Duration : 100 periods avg.

Supervisor : M. Mveng Antoine

Project Manager : M. Sahli Bertrand

Client : M. Pierre-André Favre

ETML, Rue de Sébeillon 12, 1004 Lausanne

Table des matières

[1 Specifications 3](#_Toc28967080)

[1.1 Title 3](#_Toc28967081)

[1.2 Description 3](#_Toc28967082)

[1.3 Hardware and software 3](#_Toc28967083)

[1.4 Prerequisites 3](#_Toc28967084)

[1.5 Specifications 3](#_Toc28967085)

[1.5.1 Project goal 3](#_Toc28967086)

[1.5.2 Caractéristiques des utilisateurs et impacts 4](#_Toc28967087)

[1.5.3 Fonctionnalités requises (du point de vue de l’utilisateur) 4](#_Toc28967088)

[1.5.4 Contraintes 4](#_Toc28967089)

[1.5.5 Travail à réaliser par l'apprenti 4](#_Toc28967090)

[1.5.6 Si le temps le permet … 4](#_Toc28967091)

[1.5.7 Méthodes de validation des solutions 4](#_Toc28967092)

[1.6 Les points suivants seront évalués 4](#_Toc28967093)

[1.7 Validation et conditions de réussite 4](#_Toc28967094)

[2 Planification Initiale 4](#_Toc28967095)

[3 Analyse 5](#_Toc28967096)

[3.1 Opportunities 5](#_Toc28967097)

[3.2 Document d’analyse et conception 5](#_Toc28967098)

[3.2.1 Initial 5](#_Toc28967099)

[3.2.2 Database 5](#_Toc28967100)

[3.2.3 Why Symfony 6](#_Toc28967101)

[3.3 Conception des tests 6](#_Toc28967102)

[3.3.1 Basic functionalities tests : USER 6](#_Toc28967103)

[3.3.2 Basic functionalities : EVENT 6](#_Toc28967104)

[3.4 Planification détaillée 7](#_Toc28967105)

[4 Réalisation 7](#_Toc28967106)

[4.1 Dossier de Réalisation 7](#_Toc28967107)

[4.2 Modifications 8](#_Toc28967108)

[5 Tests 8](#_Toc28967109)

[5.1 Dossier des tests 8](#_Toc28967110)

[6 Conclusion 8](#_Toc28967111)

[6.1 Bilan des fonctionnalités demandées 8](#_Toc28967112)

[6.2 Bilan de la planification 9](#_Toc28967113)

[6.3 Bilan personnel 9](#_Toc28967114)

[7 Divers 9](#_Toc28967115)

[7.1 Journal de travail 9](#_Toc28967116)

[7.2 Bibliographie 9](#_Toc28967117)

[7.3 Webographie 9](#_Toc28967118)

[8 Annexes 9](#_Toc28967119)

# Specifications

## Title

Journée formation continue : gestion des inscriptions

## Description

Description can be found at <https://trello.com/c/ocPTmPn8/3-basic-analysis>,

File is called « CahierChargeApplicationInscription.pdf »

## Hardware and software

Symfony PHP Framework

My-SQL-Workbench

PHP version 7.2.7

UwAMP 3.1.0

Composer v4.3.11

Windows 7

Internet

## Prerequisites

English, level B1

Modules : Web statique, Web dynamique

Javascript visual

UwAmp server experience

## Specifications

### Project goal

Create an equivalent to Doodle, but with more funcionalities.

Functionalities (inital):

1. Certain number of different activities during one day;
2. Possibility to create activities and special days

2. Every activity has defined max number of members;

3. Inscription with "first means first";

4. List of members is known;

5. Members subscribe to activities during the day

6. Max 1 activity at the same time

7. Visualize activity when selected

9. Possibility to get data with Excel/equivalent form as a list (see examples)

### Caractéristiques des utilisateurs et impacts

Website must be simple to use and easy to access.

Users are the teachers (ETML).

Administrator is M. Favre.

Both sides are considered to be the standard PC users, without any degree in web development.

Website should use responsive design because it may be accessed using mobile phones.

### Fonctionnalités requises (du point de vue de l’utilisateur)

See Trello board : <https://trello.com/b/UomUNlTm/projet-3eme>

And [test conception](#_Conception_des_tests) part in this project

### Contraintes

Website must be responsive

Using Symfony Framework

Using GitHub repository

This is developer’s first project with real client.

### Travail à réaliser par l'apprenti

1. The complete website with event and day creation.
2. Database with list of admins and passwords
3. Documentation
   1. Rapport
   2. Planning
   3. JdT
   4. Client manual
   5. Admin manual
4. Presentation

### Méthodes de validation des solutions

Validation with client

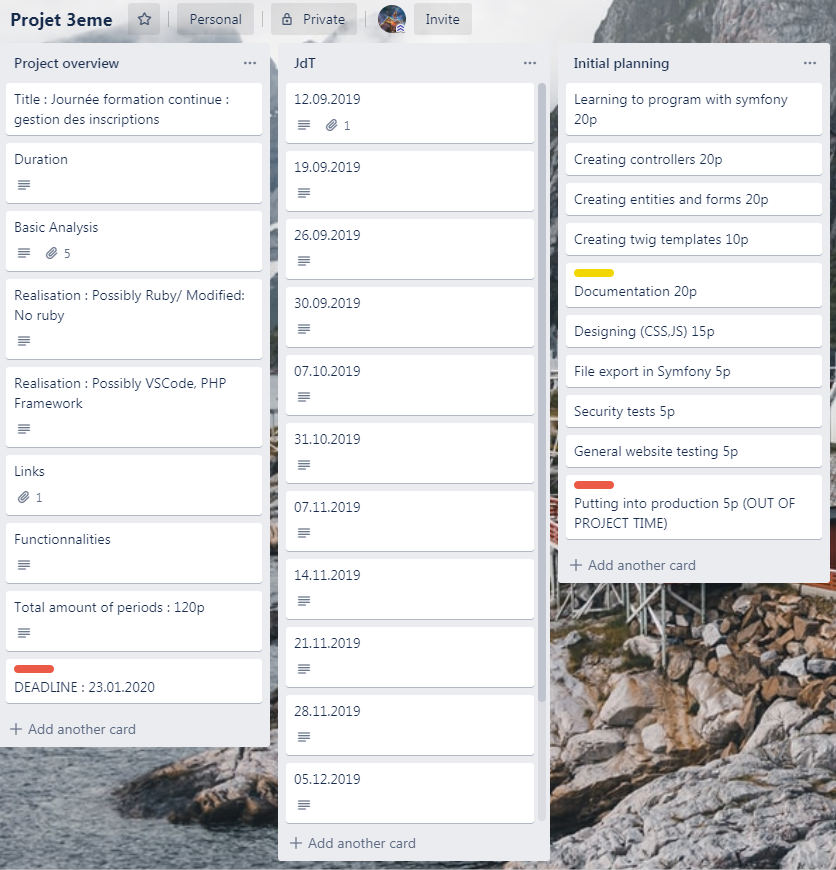
## Les points suivants seront évalués

* Le rapport
* Les planifications (initiale et détaillée)
* Le journal de travail
* Le code et les commentaires
* Les documentations de mise en œuvre et d’utilisation

## Validation et conditions de réussite

* Compréhension du travail
* Possibilité de transmettre le travail à une personne extérieure pour le terminer, le corriger ou le compléter
* Etat de fonctionnement du produit livré

# Planification Initiale

* Begin date : 12.09.2019
* Deadline : 23.01.2020
* Potential amount of periods : 120
* Developer will use Agile method, means that he’ll user some kind of user stories and will advance depending on previous stories success
* <https://trello.com/b/UomUNlTm/projet-3eme> - Trello link on public project board
* 

# Analyse

## Opportunities

Main opportunity : help ETML with a real project and real client.

Secondary opportunities :

1. Learning technologies (Symfony Framework, Composer and other).
2. {PERSONAL} Project helps to get experience in Drupal (based on Symfony) before my traineeship.

## Document d’analyse et conception

### Initial

First week (12.09.2019) – Attempt to find something similar to the project specifications, none were found.

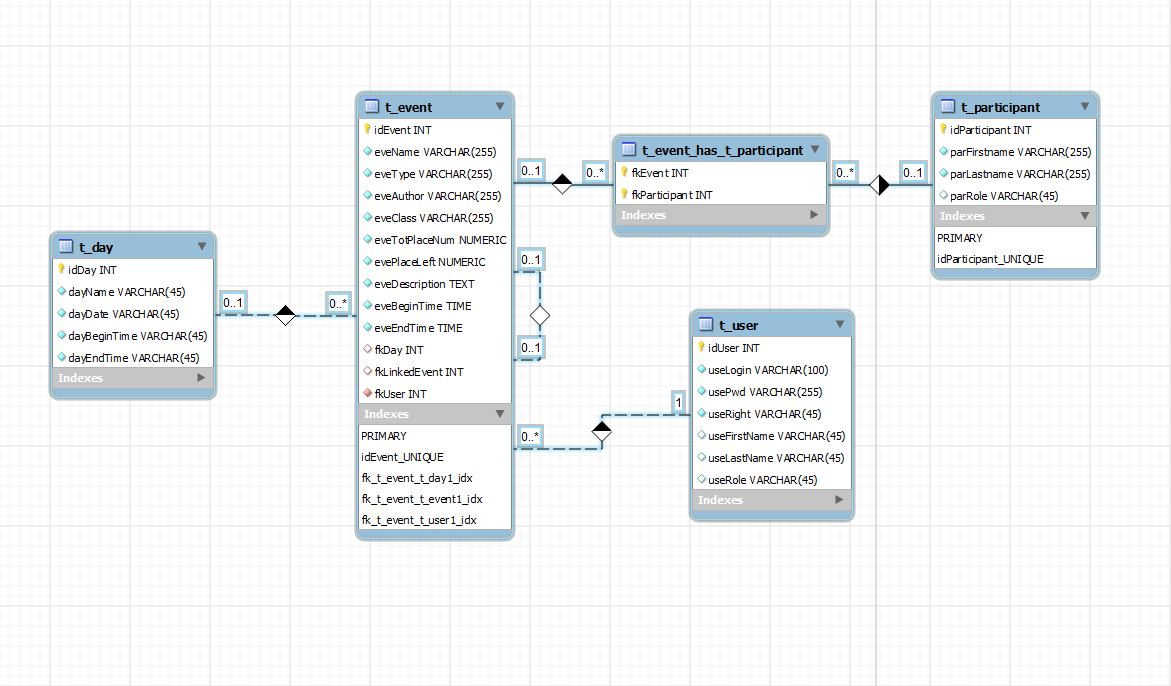
At the beginning developer (Chyzhyk Aleh) tried to use Ruby language (Ruby on Rails framework). But to learn new language structure and syntax it would have taken too much time.

So final decision : Symfony Framework using Composer.

### Database

Obviously this project requires database implementation.

Based on client demands and needs here is the Conceptual model :



This model may change during development stage

### Why Symfony

1. Symfony Framework is easier to use than the pure PHP, even though it requires some learning (syntax, structure)
2. Symfony uses twig templates ( equivalent to the html pages, but a lot easier to user)
3. Code is well structured
4. {PERSONAL} Doing another project with Symfony at the same time
5. {PERSONAL} Traineeship uses Drupal, which is based on Symfony

## Conception des tests

### Basic functionalities tests : USER

1. User can connect to his account – session variable set to true
2. After connecting user gets warning about changing his passwordd
3. Database contains 20 users with encrypted passwords – visual check
4. In case of wrong password and/or login website sends a custom error – test entries
5. In case of empty fields website sends a custom error – test entries
6. User can change his account password – database value is changed
7. Passwords are encrypted – database pwd values check
8. User can create an event – database value added
9. User can create a day – database value added
10. User can modify a day – database value modified
11. User can modify an event – database value modified
12. User can delete an event – database value deleted
13. User can delete a day – database value deleted
14. User can create multiple days – database multiple values
15. User can create multiple events – database multiple values
16. User can’t create a day with the same name and date at the same time – error display check
17. User can create an event with the same name and time – database values check
18. User can create the same event for the different hours – database values check
19. User can create the same event for the different days – database values check
20. To see…

### Basic functionalities : EVENT

1. Event has name, type, start time, end time, description, author, class (location), number of places in total, number of places left
2. Events are displayed based on priority – for example a lunch break will be on top of a long event, if the event duration includes lunch break period

## Planification détaillée

There’s no one since developer is using Agile method.

Why Agile : developer (Chyzhyk Aleh) is using Symfony Framework for the first time and can’t have even an approximate estimation of time and tasks to do during his work. Even if developer would create one it would be just a waste of time, because he wouldn’t be capable to respect it due to lack of Symfony experience.

## Dossier de Réalisation

Developer decided to divide this project into 4 sequences.

Sequence A is the first part of the project. During it developer will discover technologies, syntax and methods to use during project. The result of this sequence will be the database, web-skeleton of the project and basic pages.

Sequence B is the second part of the project. During this sequence developer will create all basic functionalities – communication with database, forms, result display, controller methods etc. The result of this sequence will be the website with all the main functionalities – pages, forms etc.

Sequence C is the third part of the project. During it developer will do mainly JS and CSS – creating design and front-end of the website. The result will be the basic design of the pages of the website.

Sequence D is the last part of the project. During this part developer will finish, modify, debug everything he have done during previous sequences, as well as test the final result and put the project in production.

Realisation sequence A:

1. Install Symfony and composer
   1. <https://getcomposer.org/download/>
   2. When php version is asked – give it php with 7.2.7 .exe file
2. Verify the result of installation
   1. “php -v“ in cmd
   2. “composer -version” in cmd
3. Create a new Symfony web-skeleton
   1. <https://openclassrooms.com/fr/courses/3619856-developpez-votre-site-web-avec-le-framework-symfony/5066156-installer-symfony-grace-a-composer>
   2. Symfony tutorial <https://symfony.com/doc/current/page_creation.html>
   3. **Note\* : Website is in MVC version, means that it’s not just symfony/skeleton, but symfony/website-skeleton.**
   4. Link to GitHub project repository -

<https://github.com/Chyzhykal/PFEProject>

1. Learn how to display basic text with twig
2. Learn how to render a page
3. Learn how to send a variable to twig template
4. Create physical database
5. Learn how to import database into Symfony project
6. Learn how to create repositories
7. Learn how to get data from the database using controller methods

Realisation sequence B:

1. Create all forms used in the website
2. Create all controllers
3. Create methods in controllers
4. Create twig pages
5. Test the functionalities

Realisation sequence C:

1. Add classes to html element in twig templates
2. Add css to these elements in “public” folder

Realisation sequence D:

TODO : Continue completing this part

Part isn’t finished because project isn’t ended.

Cette partie permet de reproduire ou reprendre le projet par un tiers.

Pour chaque étape, il faut décrire sa mise en œuvre. Typiquement :

Versions des outils logiciels utilisés (OS, applications, pilotes, librairies, etc.)

Configurations spéciales des outils (Equipements, PC, machines, outillage, etc.)

Code source commenté des éléments logiciels développés.

Modèle physique d’une base de données.

Arborescences des documents produits.

Il faut décrire le parcours de réalisation et justifier les choix.

## Modifications

Historique des modifications demandées (ou nécessaires) aux spécifications détaillées.

Date, raison, description, etc.

# Tests

## Dossier des tests

On dresse le bilan des tests effectués (qui, quand, avec quelles données…) sous forme de procédure. Lorsque cela est possible, fournir un tableau des tests effectués avec les résultats obtenus et les actions à entreprendre en conséquence (et une estimation de leur durée).

Si des tests prévus dans la stratégie n'ont pas pu être effectués :

raison, décisions, etc.

Liste des bugs répertoriés avec la date de découverte et leur état:

Corrigé, date de correction, corrigé par, etc.

# Conclusion

## Bilan des fonctionnalités demandées

Il s’agit de reprendre point par point les fonctionnalités décrites dans les spécifications de départ et de définir si elles sont atteintes ou pas, et pourquoi.

Si ce n’est pas le cas, estimer en « % » ou en « temps supplémentaire » le travail qu’il reste à accomplir pour terminer le tout.

## Bilan de la planification

Distinguer et expliquer les tâches qui ont généré des retards ou de l'avance dans la gestion du projet. Indiquer les différence entre les planifications initiales et détaillées avec le journal de travail.

## Bilan personnel

Si c’était à refaire:

Qu’est-ce qu’il faudrait garder ? Les plus et les moins ?

Qu’est-ce qu’il faudrait gérer, réaliser ou traiter différemment ?

Qu’est que ce projet m’a appris ?

Suite à donner, améliorations souhaitables, …

Remerciements, signature, etc.

# Divers

## Journal de travail

Date, activité (description qui permet de reproduire le cheminement du projet), durée, liens et références sur des documents externes. Lorsqu’une activité de recherches a été entreprise, il convient d’énumérer ce qui a été trouvé, avec les références.

## Bibliographie

Références des livres, revues et publications utilisés durant le projet.

## Webographie

Références des sites Internet consultés durant le projet.

# Annexes

Trello board: <https://trello.com/b/UomUNlTm/projet-3eme>

Database files: Database folder or <https://trello.com/c/4L1qmcLt/45-database>