

---

# Suivi de projet :

**Chronos : “ Gestion de l’assiduité ”**

—  
Etude de cas thématique  
2019-2020



BAH Hadja Mariama  
EL FERKH Meryeme  
FOULON Maxime  
LEBOURGEOIS Fabien



# Sommaire

<b>Description du projet</b>	<b>2</b>
Définition du projet	2
Méthode de la gestion du projet choisie	2
<b>Entretiens avec les clients</b>	<b>3</b>
<b>Existant</b>	<b>4</b>
Etude de l'existant	4
Vision du produit	5
Vision du projet	5
<b>Outils utilisés</b>	<b>5</b>
Outils de gestion de projet	5
Développement	5
Organisation interne	6
Outils de développement	7
<b>Conception (diagrammes)</b>	<b>8</b>
<b>Planification</b>	<b>10</b>
Répartition des tâches	11
<b>Annexes</b>	<b>12</b>
Captures d'écran de l'interface existante	12
Annexe 1 - [Administration] Dashboard	12
Annexe 2 - [Administration] Pointage des étudiants	12
Annexe 3 - [Étudiant] Pointage de l'étudiant	13
Annexe 4 - [Étudiant] Récapitulatif des pointages	13
Annexe 5 - [Enseignant] Récapitulatif des pointages par cours	13
Compte-rendu des interviews	14
Annexe 6 - La DISIP (Direction des Systèmes d'Information et du Pilotage)	14
Annexe 7 - L'administration	15
Annexe 8 - L'enseignant	15

## Description du projet

### Définition du projet

Ce projet consiste à mettre en place une plateforme permettant de dématérialiser la gestion de l'assiduité en classe pour tous les étudiants, et spécialement des étudiants sous contrat ou boursiers. En effet, des organismes peuvent être amenés à demander des attestations de présence de façon à vérifier l'assiduité des étudiants.

Ce projet nous a été proposé par la DISIP de l'UFR de Sciences. Il existe déjà une solution non finalisée, développée par un groupe de la promotion 2018-2019.

Nous avons le choix entre la reprise de la solution existante pour mettre en place la partie manquante ou la création d'une nouvelle solution répondant au besoin exprimé. Nous avons de ce fait décidé de reprendre la solution existante pour l'améliorer et la finaliser sur le périmètre d'utilisateurs du master MIAGE.

### Méthode de la gestion du projet choisie

Une concertation de groupe nous a permis de définir une méthode de gestion de projet afin de bien mener le nôtre. La méthode adoptée est de mettre en pratique la **méthode de gestion de projet agile**, qui à nos yeux nous permettra de garder régulièrement une version aboutie et fonctionnelle de notre projet.

La taille de notre groupe (4 membres) et la durée du projet (environ 8 mois, donc relativement long) sont des critères qui conviendraient parfaitement à la faisabilité d'un projet sous la tutelle d'une méthode agile. De plus, cela nous permet également de mettre en application les compétences acquises lors de nos études.

## Entretiens avec les clients

Lors de ce projet, il a été défini les clients suivants liés aux besoins de la gestion de l'assiduité :

- **Les clients DISIP** : Aurélien Boulet, David Devoye
- **Les clients administratifs** : Gwenaëlle Berquin, Stéphanie Simon
- **Le client enseignant** : Anne Lapujade

Nous avons donc organisé des entretiens avec ceux-ci pour définir leurs besoins de l'application actuelle (voir comptes rendus en annexe).

Il a été défini les besoins suivants :

- **Pour la DISIP :**
  - Mettre en place la connexion et la récupération des étudiants via le CAS (SSO),
  - Récupérer les emplois du temps avec SELCAT,
  - Finaliser le front office (partie graphique et fonctionnelle),
  - Récupérer la solution actuelle (il nous a été déconseillé de créer une nouvelle solution).
- **Pour l'administration :**
  - Permettre l'extraction des attestations de présence,
  - Permettre la gestion des absences.
- **Pour les enseignants :**
  - Pouvoir effectuer la gestion des présences (pointer ou dépointer des étudiants),
  - Pouvoir inscrire manuellement les étudiants absents.

Suite à ces entretiens, nous avons réalisé les démarches de récupération de la solution existante en environnement de développement (demande d'accès au VPN, récupération de la base de données, récupération du code). Nous avons également créé un GitHub pour la gestion du code et un Trello pour la répartition des tâches.

## Existant

### Etude de l'existant

Actuellement pour la gestion de l'assiduité, l'administration utilise des feuilles d'émargements physiques qu'elle distribue à chaque enseignant. Ce dernier a comme rôle de distribuer cette feuille aux étudiants pour qu'ils émargent dans le but de justifier leurs présences.

Comme ce que nous avons cité dans la première partie, il existe une application non finalisée et non opérationnelle, développée sous Symfony. Celle ci est déployée sur le serveur de l'UPJV, sur un environnement de développement.

Parmi les fonctionnalités existantes , on peut citer :

- Le back office,
- La gestion des droits selon le rôle de l'utilisateur (étudiant, enseignant, administration),
- Du côté enseignant : un tableau de bord listant l'émargement des étudiants au cours sélectionné,
- Du côté étudiant : l'émargement ainsi que la possibilité de signaler son départ lors du cours,
- La gestion des retards.

Plusieurs carences de la solution actuelles ont été définies par les clients :

- La récupération des utilisateurs,
- La récupération des emplois du temps,
- La gestion des absences,
- Extraction des attestations de présence et des volumes horaires,
- L'interface de l'administration.

## Vision du produit

**Pour** nos étudiants, l'administration et nos professeurs

**qui cherchent** un moyen de contrôler la présence des étudiants et spécialement ceux sous contrat ou boursiers durant leur formation, ainsi que vérifier les absences justifiées de manière dématérialisée et à récupérer les attestations de présence ou le volume horaire des étudiants

**qui offre** un moyen efficace de contrôler l'émargement des étudiants, ainsi qu'une meilleure gestion administrative des présences des élèves

**à la différence** des feuilles d'émargements qui ne sont pas pratiques pour aucun des acteurs concernés.

## Vision du projet

**Le projet** « Chronos » est la solution existante que nous allons finaliser, exploiter et améliorer le plus possible.

**Notre projet** a pour finalité d'être utilisé par l'administration, les enseignants et les étudiants afin de répertorier les absences et les présences des alternants. Pour cela, le portail sera développé par une équipe de 4 étudiants, le niveau d'effort est donc lourd car il sera d'une durée de 8 mois.

## Outils utilisés

### Outils de gestion de projet

#### Développement

En accord avec la décision de réaliser notre gestion de projet en agile, nous nous devons donc de choisir des outils appropriés à cette méthode.

Comme nous allons travailler collectivement sur le même projet de développement et donc sur le même code, nous devons avoir un processus permettant de stocker les mises à jour, et où chacun d'entre nous sera capable d'effectuer ces mises à jour. Ainsi qu'une méthode permettant de réaliser un rollback (retour arrière) en cas de crash de l'application et de garder une trace de toutes les modifications effectuées sur le code pour pouvoir

comprendre les modifications. Pour cela, nous avons opté pour la solution **Git (logiciel de versionning) - Github (hébergeur des repositories de code)** pour les avantages suivants:

- gratuité
- réputé chez les développeurs et utilisé pour des millions de projets
- l'un des membres de notre groupe maîtrise GitHub

Afin d'utiliser efficacement la solution GitHub, chaque membre du groupe se munira de **Git Desktop** pour mettre à jour le projet par rapport à la version local que chacun possède sur sa machine. L'utilisation de git requiert quelques prérequis : les membres déjà à l'aise avec l'application exerceront une transmission de compétences aux autres membres qui ne connaissent pas bien cette application.

Nos clients de la DISIP nous ont transmis les identifiants de connexion afin d'accéder au projet réalisé par nos camarades de l'année passée via une connexion distante en VPN. Le logiciel par lequel nous passons afin de réaliser l'accès VPN est **Pulse secure**.

Puisque nous nous appuyons sur le projet déjà existant, nous allons réutiliser le framework sur lequel ce dernier s'appuyait : **Symfony**. Ce framework réputé et actuel est conçu pour développer des applications en groupe, en partie grâce à la structure en MVC (Modèle-Vue-Contrôleur).

L'interface nous permettant d'accéder au Mysql est **PhpMyAdmin**, déjà utilisée par nos prédécesseurs. Cette solution est connue et maîtrisée par l'ensemble des membres de notre groupe, et est réputée pour être l'une des interfaces les plus efficaces pour gérer une base de données MySQL.

## Organisation interne

L'organisation et la communication entre les différents membres du groupe est, à nos yeux, la clé du projet pour éviter des omissions, des doublons de tâches par plusieurs membres du projet, et garder une visibilité sur l'état d'avancement et la répartition des tâches pour tous les membres. Pour cela, nous avons décidé d'opter pour l'outil **Trello**, qui présente les intérêts suivants :

- gratuité de l'outil
- maîtrise de l'outil par l'ensemble des membres du groupe (utilisé lors de la phase de préparation du projet).
- outil en ligne, accessible depuis n'importe quelle plateforme et navigateur.

Régulièrement, l'ensemble des membres du groupe se retrouvent sur une plateforme de VoIP nommée **Discord**, qui nous permet d'échanger sur le projet thématique. Sur un serveur privé Discord, nous échangeons également des informations peu sensibles liées au projet.

Lorsque nous devons nous échanger des informations plus sensibles, l'utilisation du service de **mailing de l'UPJV** est la solution que nous privilégions systématiquement.

## Outils de développement

Comme nous avons fait le choix de se baser sur le travail qui avait été entrepris par nos collègues de MIAGE de l'année dernière, nous allons exploiter le travail fourni par la DISIP sous **Symfony 4.2.5**. Notre travail de développement se basera alors sur ce framework.

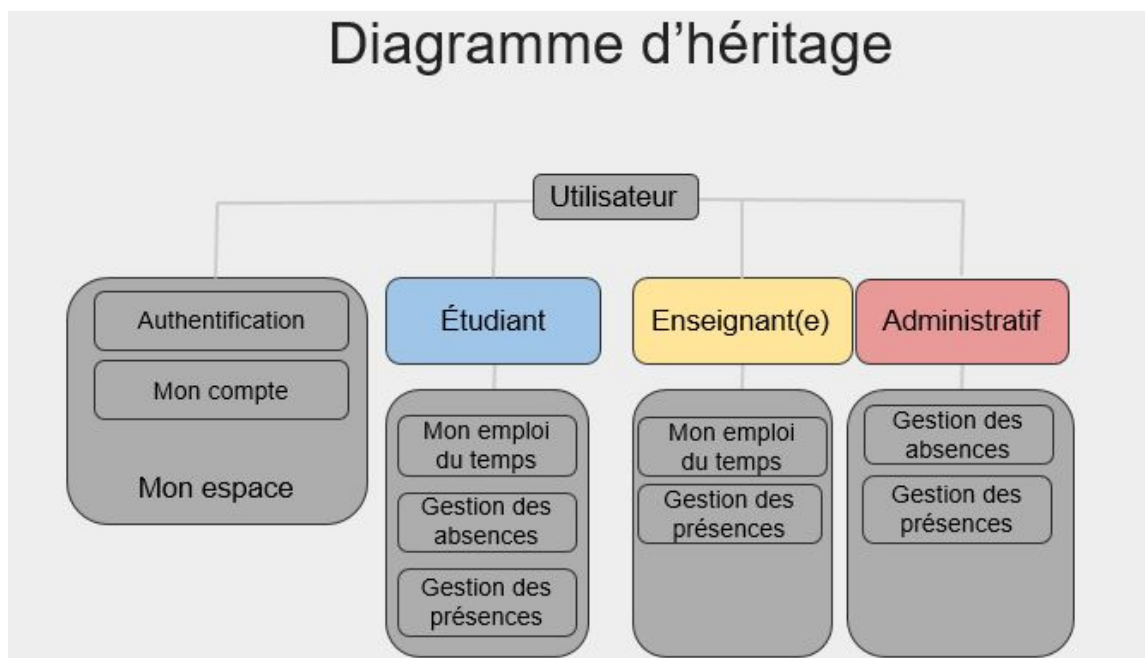
Nous avons pris la décision de nous baser sur un IDE pour faciliter le développement de notre projet. Nous avons choisis **Visual Studio Code**, qui à nos yeux présentent les avantages suivants:

- La gratuité
- La coloration syntaxique des langages webs que nous utiliserons est automatiquement prise en charge par le logiciel.
- La lecture des erreurs et des alertes est claire et lisible.
- La plupart des membres de notre groupe l'utilise quotidiennement, sous Windows et sous Linux, et cet IDE est, par expérience, celui qui nous a posé le moins de problème (très peu de défaillance, moins lourd que certains concurrents, relativement rapide...).

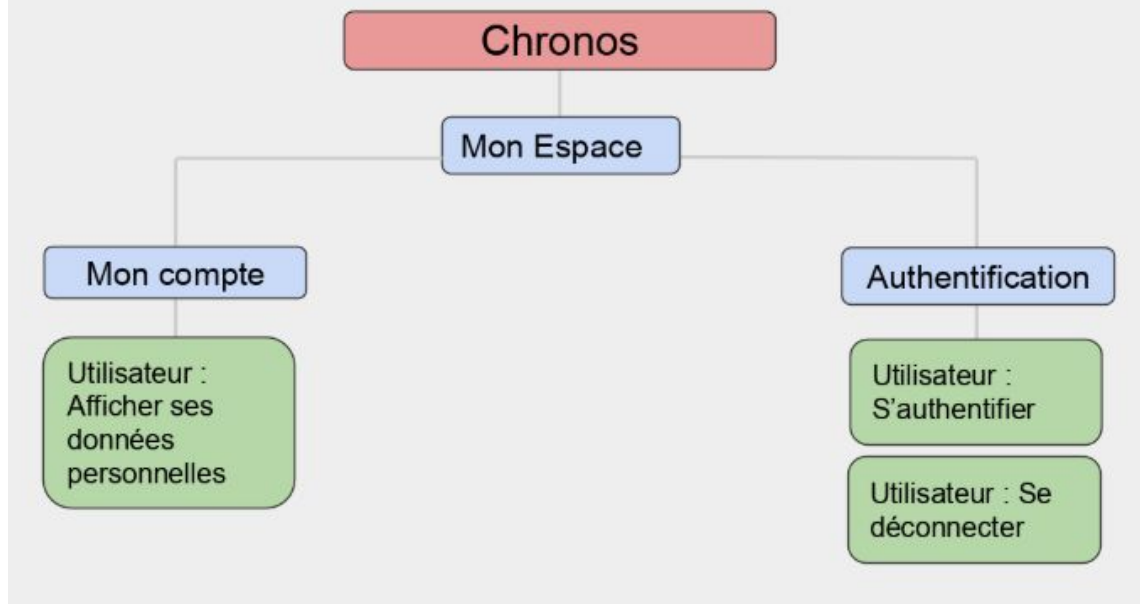


## Conception (diagrammes)

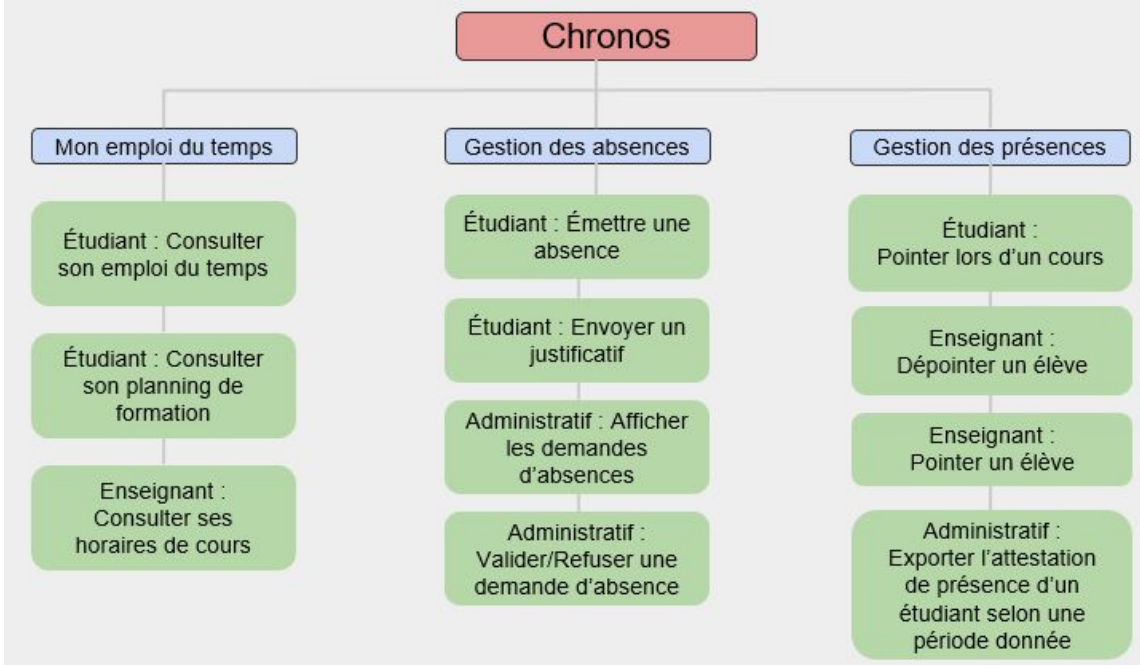
Lors de nos entretiens nous avons pu déterminer certains processus de l'application. De ce fait, nous avons pu créer les arbres prévisionnels suivants permettant de définir les fonctionnalités utilisées par chaque type d'utilisateur. Voici les arbres que nous avons défini (une prochaine interview avec les clients permettra de valider la priorisation)



# Arbre du produit



# Arbre du produit



## Planification

Nous diviserons les différentes tâches en plusieurs sprints d'environ 40 points. Chaque sprint dure environ 4 semaines.

Le premier sprint consistera à analyser le code existant, que ce soit le back office ou le front office cela fera office du premier sprint.

Lors du second sprint, nous effectuerons les fonctions permettant l'intégration des étudiants et de l'emploi du temps de tous les étudiants.

Le troisième sprint, sera en rapport avec les fonctions permettant la gestion des absences ainsi que l'ajout d'absence par les étudiants.

Un quatrième sprint permettra à l'administration de récupérer les attestations de présence des étudiants ou d'un parcours et de les télécharger ainsi que la gestion des RGPD.

Un dernier sprint fera l'objet de la création du lien avec le CAS permettant d'intégrer par la suite le projet à l'ENT de l'UPJV.

Il n'y aura pas de release pour une plage de sprint car le projet n'est pas encore un produit fini, mais encore en cours de développement. Il sera mis en production une fois que toutes les fonctionnalités seront terminées.

Le détail de cette planification non fixée est le suivant :

40 pts	1. Analyse du back office	25 pts	Sprint 1
	2. Analyse du front office	15 pts	
41 pts	3. Récupération de la liste des étudiants	9 pts	Sprint 2
	4. Intégration de la liste des étudiants	15 pts	
	5. Rattachement de l'emploi du temps à l'étudiant connecté	17 pts	
40 pts	6. Création d'un formulaire d'absence	5 pts	Sprint 3
	7. Envoi des demande d'absence à l'administration (secrétariat)	12 pts	
	8. Intégration d'un tableau de bord des absences pour l'administration	14 pts	
	9. Formulaire de validation/refus des demandes d'absences	6 pts	
	10. Envoi de la validation/refus de la demande à l'étudiant	3 pts	
39 pts	11. Récupération des pointages des étudiants selon une période	8 pts	Sprint 4
	12. Exporter les pointages selon une période	6 pts	
	13. Automatiser une purge des données	9 pts	
	14. Créer l'interface du processus de refus	9 pts	
	15. Processus de refus de conservation des données	7 pts	Sprint 5
39 pts	16. Lier l'application au CAS	15 pts	
	17. Configuration du SSO	12 pts	
	18. Intégration à l'ENT	13 pts	

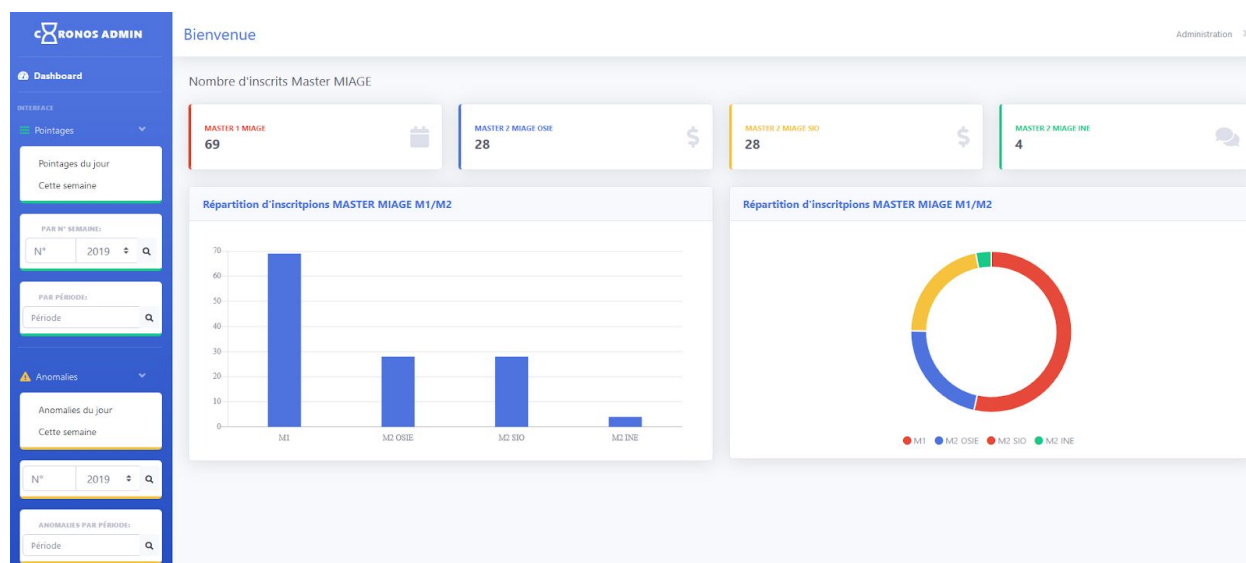
## Répartition des tâches

Membre du groupe	Tâches affectées
BAH Hadja Mariama	<ul style="list-style-type: none"> <li>• <b>Sprint 1</b> <ul style="list-style-type: none"> <li>○ Analyse du front office</li> </ul> </li> <li>• <b>Sprint 2</b> <ul style="list-style-type: none"> <li>○ Récupération de la liste des étudiants</li> <li>○ Intégration de la liste des étudiants</li> </ul> </li> <li>• <b>Sprint 3</b> <ul style="list-style-type: none"> <li>○ Création d'un formulaire d'absence</li> <li>○ Formulaire de validation/refus des demandes d'absences</li> </ul> </li> </ul>
EL FERKH Meryeme	<ul style="list-style-type: none"> <li>• <b>Sprint 1</b> <ul style="list-style-type: none"> <li>○ Analyse du front office</li> </ul> </li> <li>• <b>Sprint 2</b> <ul style="list-style-type: none"> <li>○ Intégration de la liste des étudiants</li> </ul> </li> <li>• <b>Sprint 3</b> <ul style="list-style-type: none"> <li>○ Envoi de la validation/refus de la demande à l'étudiant</li> <li>○ En soutien sur le sprint de Fabien et/ou Maxime</li> </ul> </li> </ul>
FOULON Maxime	<ul style="list-style-type: none"> <li>• <b>Sprint 1</b> <ul style="list-style-type: none"> <li>○ Analyse du back office</li> </ul> </li> <li>• <b>Sprint 2</b> <ul style="list-style-type: none"> <li>○ Rattachement de l'emploi du temps à l'étudiant connecté</li> </ul> </li> <li>• <b>Sprint 3</b> <ul style="list-style-type: none"> <li>○ Envoi des demandes d'absence à l'administration</li> </ul> </li> </ul>
LEBOURGEOIS Fabien	<ul style="list-style-type: none"> <li>• <b>Sprint 1</b> <ul style="list-style-type: none"> <li>○ Analyse du back office</li> </ul> </li> <li>• <b>Sprint 2</b> <ul style="list-style-type: none"> <li>○ Intégration de la liste des étudiants</li> <li>○ Rattachement de l'emploi du temps à l'étudiant connecté</li> </ul> </li> <li>• <b>Sprint 3</b> <ul style="list-style-type: none"> <li>○ Intégration d'un tableau de bord des absences pour l'administration</li> </ul> </li> </ul>

## Annexes

### Captures d'écran de l'interface existante

#### Annexe 1 - [Administration] Dashboard



#### Annexe 2 - [Administration] Pointage des étudiants

Pointages - Semaine 48.2019 || Du 25/11/2019 au 01/12/2019 Voir toutes les anomalies sur la période Administration

POINTAGES	RETARDS	DÉPART EN AVANCE	ABSENCES
6	5	4	0

Afficher Voir tout éléments

Colonnes PDF CSV

Formation	Date	Entrée	Sortie	Cours	Etudiant	Anomalies	Actions
M2 Miage OSIE	26/11/2019	15:59	19:00	ISL_11 Progiels de gestion intégrés	David Dupont		✓
M2 Miage OSIE	26/11/2019	16:45	16:48	ISL_15 Stratégie et management des SI	David Dupont	Retard	Sorti en avance ✓
M2 Miage OSIE	26/11/2019	16:48	16:48	ISL_09 Audit des SI	David Dupont	Retard	Sorti en avance ✓
M2 Miage OSIE	26/11/2019	15:59	15:59	ISL_17 Management de la sécurité des SI	David Dupont	Retard	Sorti en avance ✓
M2 Miage OSIE	26/11/2019	15:58	15:58	ISL_11 Progiels de gestion intégrés	David Dupont	Retard	Sorti en avance ✓
M2 Miage OSIE	26/11/2019	16:49	✓	ISL_09 Audit des SI	David Dupont	Retard	✓
Formation	Date			Cours	Etudiant		

Affichage de l'élément 1 à 6 sur 6 éléments

← 1 →

## Annexe 3 - [Étudiant] Pointage de l'étudiant

## Annexe 4 - [Étudiant] Récapitulatif des pointages

Date	Entrée	Sortie	Cours	Présence	Anomalies
26/11/2019	16:49	20:52	ISI_09 Audit des SI	4h	Retard
26/11/2019	16:48	16:48	ISI_09 Audit des SI	30	31706 1
26/11/2019	16:45	16:48	ISI_15 Stratégie et management des SI	164	31528 1
26/11/2019	15:59	19:00	ISI_11 Projets de gestion intégrés	0	0 1
26/11/2019	15:59	15:59	ISI_17 Management de la sécurité des SI	28	25149 1
26/11/2019	15:58	15:58	ISI_11 Projets de gestion intégrés	51	28686 1

## Annexe 5 - [Enseignant] Récapitulatif des pointages par cours

Émargement

Sélectionnez le cours pour visualiser les pointages effectués aujourd'hui pour ce cours.

Cours: ISI\_09 Audit des SI

Filtrer

ISI\_09 AUDIT DES SI 8H-11H  
2 pointages

Étudiant	Entrée	Sortie
21207671 Alexandre DUPOND	21:06	Encore présent
21103485 David DUPONT	21:17	Encore présent

Le rafraîchissement du tableau est automatique

## Compte-rendu des interviews

### Annexe 6 - La DISIP (Direction des Systèmes d'Information et du Pilotage)

#### Participants de la DISIP :

- Aurélien BOULET
- David DEVOYE

#### Discussions sur l'existant et le besoin :

- Application existante en Symfony sur un environnement de développement (PHP 7.0 ou 7.2),
- Majeure partie du back office réalisée,
- Lien avec Selcat (récupération des emplois du temps) non présent,
- Environnement de développement accessible uniquement via un VPN,
- Possibilité de pointer pour un cours 15 minutes avant ou après, sinon statut "retard",
- Il n'existe pas de fonction permettant l'extraction des attestations de présence,
- Un git est disponible pour récupérer la programmation de l'application.

#### Actions à réaliser :

- Récupération du GIT → David DEVOYE va nous envoyer les accès,
- Récupération de la base de donnée → David DEVOYE va nous envoyer les accès,
- Récupération du produit existant → David DEVOYE va nous envoyer les accès,
- Accès au VPN → Se renseigner auprès du pôle technique du bâtiment F,
- Se renseigner sur les RGPD à renseigner (données personnelles des utilisateurs),
- Mettre en place la récupération des emplois du temps (SELCAT),
- Mettre en place la récupération des utilisateurs (CAS).

#### Contraintes techniques :

- Reprendre la solution actuelle → fortement conseillé,
- Utiliser symfony (PHP 7.0, 7.2),
- Outil à programmer selon les standards de l'UPJV.

## Annexe 7 - L'administration

### Participants de l'administration :

- Stéphanie Simon

### Discussions sur l'existant et le besoin :

- Les entreprises ou financeurs peuvent réclamer des attestations de présence,
- L'administration doit être en capacité de récupérer les présences sur une période définie,
- Le volume horaire doit être également récupérable (mensuel pour l'employeur, annuel pour les financeurs),
- Souhait de pouvoir récupérer les demandes d'absences sur la plateforme.

### Actions à réaliser :

- Création d'un formulaire d'absence (pour l'étudiant),
- Création d'un panel permettant l'affichage des demandes d'absence (pour l'administration),
- Création d'un panel permettant de récupérer les attestations de présence et le volume horaire des étudiants.

## Annexe 8 - L'enseignant

### Participant enseignant :

- Anne Lapujade

### Discussions sur l'existant et le besoin :

- Souhait d'une interface graphique ergonomique, facile,
- Permettre le contrôle des présences,
- Permettre la saisie des absences.

### Actions à réaliser :

- Permettre le contrôle des étudiants présents,
- La possibilité de retirer l'émargement en cas de mauvaise renseignement,
- Mettre en place la saisie des absences.