

**PlanIFTicateur**

**Travail pratique 1**

GLO-2004 – Génie Logiciel Orienté Objet – Hiver 2015

Travail présenté à

M. Jonathan Gaudreault

Chayer, Philippe [Philippe.chayer.1@ulaval.ca](mailto:Philippe.chayer.1@ulaval.ca) IFT PHCHA47

Khouma, Abdou [abdou.khouma.1@ulaval.ca](mailto:abdou.khouma.1@ulaval.ca) GIF ABKHO9

Gadoury, Gabriel [Gabriel.gadoury.1@ulaval.ca](mailto:Gabriel.gadoury.1@ulaval.ca) IFT GAGAD1

Yeo, Clotioloman [Clotioloman.yeo.1@ulaval.ca](mailto:Clotioloman.yeo.1@ulaval.ca) GLO CLYEO1

Table des matières

[Introduction 4](#_Toc410826030)

[Vision *PlanIFTicateur* 5](#_Toc410826031)

[I) Modélisation du domaine d’affaires 6](#_Toc410826032)

[Modèle d'affaires 6](#_Toc410826033)

[II) Analyse des besoins 8](#_Toc410826034)

[Modèle des cas d’utilisation 8](#_Toc410826035)

[Placer une activité 11](#_Toc410826036)

[DSS – Placer une activité 12](#_Toc410826037)

[Importation des données 13](#_Toc410826038)

[DSS – Importation des données 14](#_Toc410826039)

[Sauvegarde de l’horaire en construction 15](#_Toc410826040)

[DSS – Sauvegarde de l’horaire en construction (cas 1 – Par défaut) 16](#_Toc410826041)

[DSS – Sauvegarde de l’horaire en construction (cas 2 – scénario alternatif) 16](#_Toc410826042)

[Modifier les attributs d’une activité. 17](#_Toc410826043)

[DSS – Modifier les attributs d’une activité 17](#_Toc410826044)

[Planification automatique 18](#_Toc410826045)

[DSS – Planification automatique 18](#_Toc410826046)

[Exportation de l’horaire sous forme d’image 19](#_Toc410826047)

[DSS – Exportation de l’horaire sous forme d’image 19](#_Toc410826048)

[Affichage des statistiques 20](#_Toc410826049)

[DSS – Affichage des statistiques 20](#_Toc410826050)

[Prendre des notes sur l’horaire 21](#_Toc410826051)

[DSS – Prendre des notes sur l’horaire 21](#_Toc410826052)

[Charger un horaire 22](#_Toc410826053)

[DSS – Charger un horaire 23](#_Toc410826054)

[Spécifications supplémentaires 24](#_Toc410826055)

[Glossaire 25](#_Toc410826056)

[III) Gestion de projet 28](#_Toc410826057)

[Référence des taches 28](#_Toc410826058)

[Nature des données 28](#_Toc410826059)

[Gestion des fichiers 28](#_Toc410826060)

[Interface utilisateur 29](#_Toc410826061)

[Validation de l’horaire 29](#_Toc410826062)

[Planification automatique 30](#_Toc410826063)

[Affichage de statistiques 30](#_Toc410826064)

[Divers 30](#_Toc410826065)

[Assignation des ressources 31](#_Toc410826066)

[Diagramme de Gant 33](#_Toc410826067)

[Budget 34](#_Toc410826068)

# Introduction

Ce rapport consiste en une analyse de la faisabilité pour la conception d’un logiciel permettant la gestion et la création d’horaire. Cette analyse sera basée sur les informations fournies par le client. Le but du logiciel est de faciliter la gestion des horaires de session effectuée par la direction de programme.

La construction des horaires de session est une tâche difficile pour la direction de programme. À l’aide de notre logiciel, cette tâche deviendra visuelle et interactive, donc plus facile à réaliser. Ainsi, la direction économisera beaucoup de temps et, par le fait même, diminuera significativement le coût associé à la production d’un horaire optimale.

Le document ci-présent est divisé en 4 sections. Une première section détaille les objectifs du logiciel (ainsi que les diverses fonctions qui seront implantées). La section suivante illustre un schéma de domaine d’affaires. Ensuite, une section est consacrée aux cas d’utilisations détaillés sous forme de schémas et de textes. Leur but consiste à définir les fonctions présentées et décrire les principaux scénarios dans lesquels les fonctions du logiciel seront utilisées. Finalement, une section concernant la gestion du projet ainsi qu’un rapport de temps requis afin de mener le projet à terme.

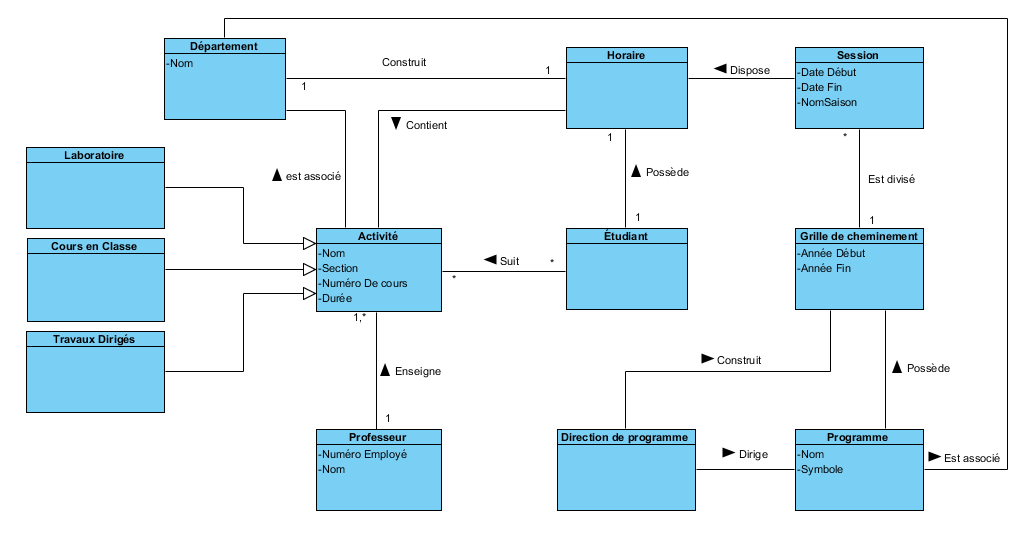
# Vision *PlanIFTicateur*

* ***PlanIFTicateur*** est un programme apte à construire un horaire de session de façon interactive et peut, si souhaité, faire la génération de votre horaire de façon optimale (rapide, efficace et flexible). Le but primaire est de rendre agréable la construction de l’horaire tout en réduisant le temps consacré à ce dernier.
* Une planification automatique optimale grâce à un algorithme de recherche efficace se basant sur les restrictions et les statistiques de ce qu’est un bon horaire.
* Une interface simple et intuitive visant à aider l’utilisateur dans son travail.
* Un suivi simple et efficace grâce à une fonctionnalité permettant de prendre des notes en lien avec la grille horaire que vous construisez.
* Les horaires créés seront en tous points valides grâce à des fonctionnalités de validations optimales. Ces fonctions rendront facile et stimulante la création d’un horaire grâce à son aspect visuel et dynamique.
* Une importation des données et restrictions des cours instantanée facilitant le travail et limitant les erreurs ou les oublis.
* Des statistiques pertinentes disponibles pour la grille horaire en construction visant à l’optimisation de votre horaire.
* Une exportation de fichier simple et facile à des fins d’utilisation hors programme.
* Un historique est disponible pour un suivi efficace de toutes les modifications réalisées sur votre planification.

# I) Modélisation du domaine d’affaires

Cette section consiste à vous présenter une perspective générale du fonctionnement de *PlanIFTicateur*. Comme tout bon modèle d’affaires, le schéma ci-dessous contient l’ensemble des informations nécessaires à la compréhension de notre projet, voire notre logiciel. Suite au visionnement du dît schéma, nous pourrons constater l’ampleur du travail à accomplir et poser un meilleur jugement sur les choix de fonctionnalités à développer.

## Modèle d'affaires



***Détails du diagramme* :** Pour le diagramme de modèle de domaine, plusieurs choix ont été faits afin d’obtenir le diagramme ci-dessus. Tout d'abord, le département est présent puisque c'est ce dernier qui utilisera l'application (ou quelqu'un de mandaté par la direction du département). Ensuite, la directionde programme est directement liée aux programmes puisque c'est la direction qui dirige ces programmes et qui construit les grilles de cheminement. Ces grilles sont alors divisées en sessions qui ont tous un horaire à respecter selon certains critères, c'est-à-dire d'éviter les conflits d'horaires entre les cours d’une même session (facteurs fournis par les grilles de cheminement). Dans notre diagramme, l’horaire peut être vu de deux façons :

-Soit comme étant un horaire d'étudiant qui contient normalement cinq cours avec, possiblement des laboratoires;

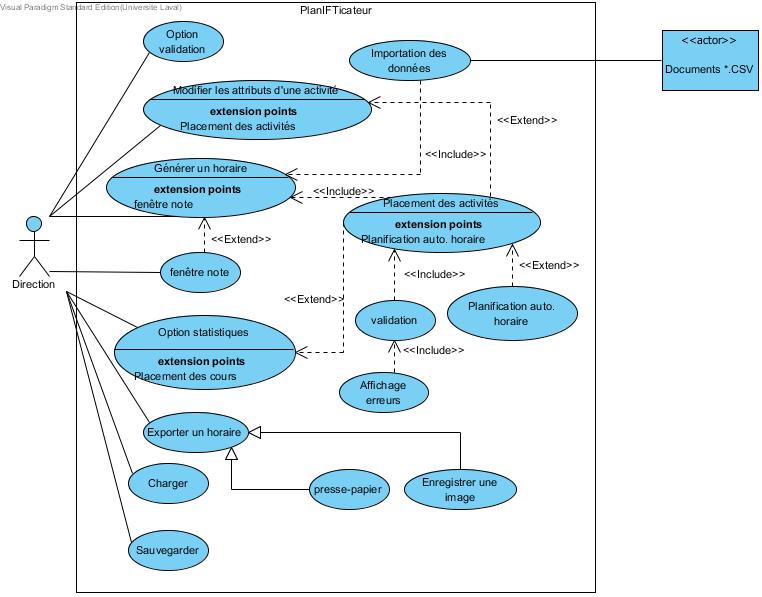
-Soit comme étant un horaire de tous les cours qui sont donnés dans un département. D’ailleurs, c’est de cette façon que la direction planifie les horaires des sessions qui permettra ensuite d’obtenir celle des étudiants.

Par la suite, il y a les activités, qui sont réunies dans une superclasse pour ce diagramme et qui se divise en trois sous-classes : les laboratoires, les cours en classe et les travaux dirigés. Ces sous-classes correspondent aux différents types d'activités qui seront mises à l'horaire. Finalement, il y a les étudiants qui font partie du modèle puisque l'horaire construit concerne directement les étudiants. D'ailleurs, les statistiques générées par l'application sont basés sur ces étudiants.

# II) Analyse des besoins

Maintenant que nous avons une bonne vue d’ensemble, il sera plus simple de comprendre la section ci-présente. Le modèle qui suivra représente le système, *PlanIFTicateur*, abritant tous ses cas d’utilisation. Il y aura également une représentation sommaire des acteurs du système. Suite à ce modèle, suivra des tables détaillant chacun des cas d’utilisation du modèle, accompagnées de leurs diagrammes de séquence système, servant à illustrer l’échange du système et l’utilisateur du système lors d’un événement en rapport avec le cas d’utilisation en question.

## Modèle des cas d’utilisation

***Détails du diagramme* :** Le diagramme ci-dessus commence au niveau de l’acteur principal et utilisateur de l’application, soit la direction. Chacun des liens qui réunis la direction aux cas d’utilisation informe que l’acteur doit prendre la décision de « d’activer » le cas en question. Pour ce qui est de ces cas, nous parlons bien de : « **Option validation** », « **Générer un horaire** », « **Fenêtre note** », « **Option statistiques** » et « **Exporter un horaire** ».

Pour commencer, « **Générer un horaire** » explique sa présence par la nécessité d’un point de départ. L’application reste indépendante de la planification en tant que telle, c’est-à-dire que le programme peut s’ouvrir sans nécessairement activer automatiquement la génération d’un horaire (l’application ouvre sur une fenêtre vide en attente d’un événement par l’utilisateur). Comme ce cas est, en quelque sorte, le point de départ du programme, plusieurs autres cas s’y rattachent. En effet, lorsque l’utilisateur souhaite faire une planification d’horaire, « **l’importation des données** » est activée (dont les données sont pigées à même les « **documents \*.CSV** »), ensuite vient le « **placement des activités** » qui fait suite logique à la création d’un horaire. Évidemment, le programme étant doté d’un système de validation, la « **validation** » est intégrée au placement des activités et, il en va de même pour « **l’affichage d’erreurs** » par rapport à la validation puisqu’une erreur ne s’affichera que si la validation le décide. On remarque que la « **planification automatique d’un horaire** » est une extension du placement des activités puisqu’il s’agit d’une option que l’utilisateur peut employer, donc non obligatoire. Sans oublier la « **modification des attributs d’une activité** » qui nécessite la liste des cours pour prendre sens. Nous croyons que le meilleur emplacement est de le joindre comme extension à «**placement des activités** » étant donné que seul la génération d’horaire ne contient pas la liste des cours en tant que telle.

Pour ce qui est de « **l’option validation** », il s’agit encore là d’une option que l’utilisateur peut activer à son gré.

En ce qui a trait à la « **fenêtre note** », cette fenêtre peut ne pas exister (aucune note) ou peut être activée par l’utilisateur qui y prendra des notes comme il le souhaite qui seront enregistrées avec la planification en cours.

Également, « **exporter un horaire** » n’est activé que par l’utilisateur et qu’il s’agit d’un choix entre exporter dans le « **presse-papier** » ou « **enregistrer un fichier** ».

Finalement, le choix de placement de « **charger** » et « **sauvegarder** » s’explique par l’activation par l’utilisateur uniquement.

## Placer une activité

|  |  |
| --- | --- |
| Cas d'utilisation: | **Placer une activité** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme** |
| Préalables: | Le programme est en fonction et les fichiers requis ont été chargés préalablement avec succès. |
| Garanties en cas de succès | Le cours est placé à l’endroit souhaité sans conflit d'horaire. |
| Scénario principal: | 1. L'individu qui construit l'horaire détermine le cours qui doit être placé sur la grille. |
| 2. Il appuie sur le cours choisit et le glisse dans la grille horaire à la plage horaire souhaitée (« drag-and-drop »). |
| 3. Le cours nouvellement placé à l'horaire est retiré de la liste des cours. |
| Note scénario principal: | Dans ce scénario, la validation de l'horaire est activée. C’est-à-dire que les plages horaires non valides sont grisées. Ainsi, l'horaire construit sera toujours valide. |
| Scénario alternatif: | 1. L'individu qui construit l'horaire détermine le cours qui doit être placé sur la grille. |
| 2. Il appuie sur le cours choisit et le glisse dans la grille horaire à la plage horaire souhaitée. |
| 3. Si le cours est à un endroit valide, l’utilisateur peut continuer sa planification. Sinon, le logiciel indique que l’horaire n’est pas valide et affiche un message significatif du problème. |
| 3. Le déplacement est annulé et le cours retourne dans la liste des cours. |
| Note scénario alternatif: | Dans ce cas-ci, la validation de l'horaire n'est pas activée permettant à l'utilisateur de placer des cours sur toutes les plages horaires. |

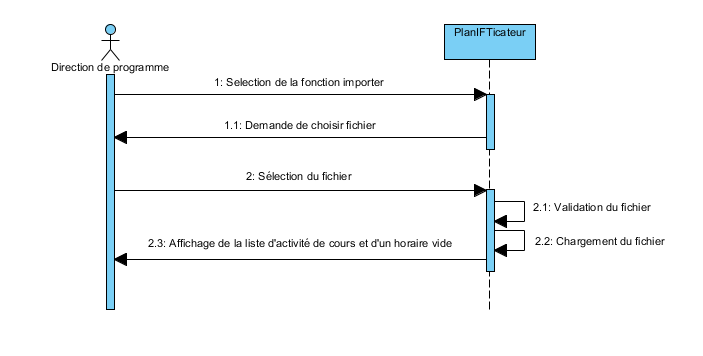
### DSS – Placer une activité

#### 

## Importation des données

|  |  |
| --- | --- |
| Cas d'utilisation: | **Importation des données** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme** |
| Préalables: | Le logiciel est en fonction. L'utilisateur possède les fichiers nécessaires. |
| Garanties en cas de succès | Affichage de la liste des cours. |
| Scénario principal: | 1. L'utilisateur sélectionne l’option « Importer ». |
| 2. L'utilisateur indique le chemin du fichier à importer. |
| 3. L'utilisateur indique le nom du fichier d’horaire. |
| 4. Le fichier est importé avec succès. |
| 5. Le logiciel affiche la liste des cours dans la partie droite du logiciel. |
| Note scénario principal: | Dans ce cas, il n'y a pas d'erreur, et les fichiers sont chargés avec succès. Bien entendu, il faut que les 2 fichiers se trouvent au même endroit (même chemin et porte le même nom (avec les extensions appropriées). |

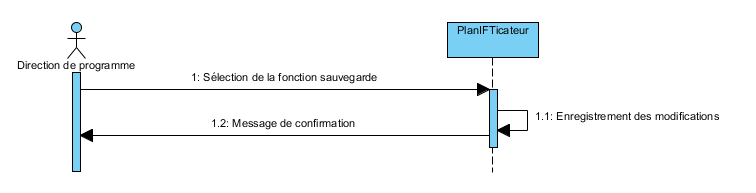
### DSS – Importation des données



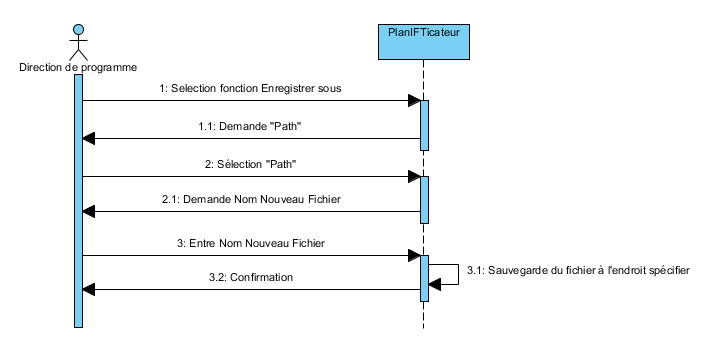
## Sauvegarde de l’horaire en construction

|  |  |
| --- | --- |
| Cas d'utilisation: | **Sauvegarde de l'horaire en construction** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme** |
| Préalables: | Le logiciel est en fonction. Les fichiers d’horaire doivent avoir été chargés préalablement. |
| Garanties en cas de succès | Enregistre le travail effectué jusqu'au moment de la sauvegarde. |
| Scénario principal: | 1. L'utilisateur appuie sur le bouton « sauvegarde ». |
| 2. Le fichier est sauvegardé. |
| 3. L'utilisateur peut continuer sa planification. |
| Note scénario principal: | Par défaut, lorsque l'utilisateur appuie sur le bouton « sauvegarde », le fichier chargé est écrasé par ce même fichier mis à jour.  P.S. : L’utilisateur n’a pas besoin d’avoir fait des modifications à l’horaire pour pouvoir le sauvegarder. |
| Scénario alternatif: | 1. L'utilisateur appuie sur le bouton « Enregistrer sous… » |
| 2. L'utilisateur choisit le chemin de sauvegarde. |
| 3. L'utilisateur indique le nom du fichier souhaité. |
| 4. Le fichier est enregistré à l'endroit spécifié par l'utilisateur. |
| 5. L'utilisateur peut continuer sa planification. |
| Note scénario alternatif: | Dans ce scénario, l'utilisateur indique l’endroit de la sauvegarde gardant le fichier source intact. |

### DSS – Sauvegarde de l’horaire en construction (cas 1 – Par défaut)



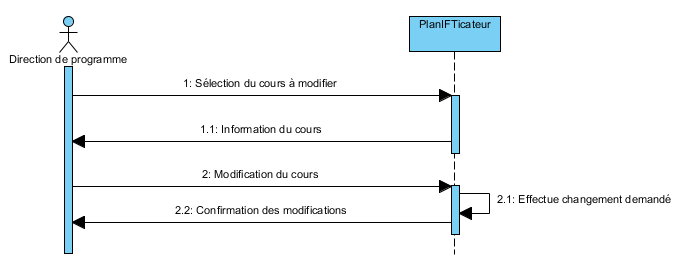
### DSS – Sauvegarde de l’horaire en construction (cas 2 – scénario alternatif)



## Modifier les attributs d’une activité.

|  |  |
| --- | --- |
| Cas d'utilisation: | **Modifier les attributs d'un cours.** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme** |
| Préalables: | Le logiciel est en fonction. Un horaire est ouvert sur le logiciel. |
| Garanties en cas de succès | Affiche les informations sur le cours et donne la possibilité de changer les valeurs associées à ce dernier. |
| Scénario principal: | 1. L'utilisateur fait un double-clique sur le cours. |
| 2. Une fenêtre s’ouvre et les champs des caractéristiques du cours sélectionné sont affichés permettant à l'utilisateur de les modifier. |
| 3. Les paramètres modifiés sont conservés lors de la fermeture de la fenêtre. |
| Note scénario principal: | Cela permet de modifier la durée du cours ainsi que les restrictions s'appliquant sur ce dernier. |

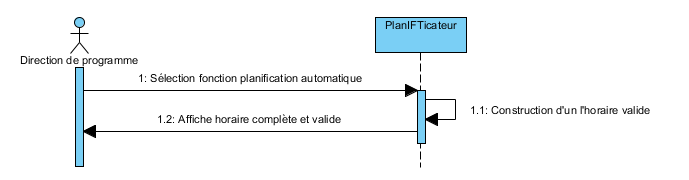
### DSS – Modifier les attributs d’une activité



## Planification automatique

|  |  |
| --- | --- |
| Cas d'utilisation: | **Planification automatique** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme** |
| Préalables: | Le logiciel est en fonction. |
| Garanties en cas de succès | L'horaire construit est valide. |
| Scénario principal: | 1. L'utilisateur appuie sur le bouton « Planification automatique de l'horaire ». |
| 2. Le logiciel affiche un horaire valide. |
| Note scénario principal: | Lors de la planification automatique, le logiciel place toutes les activités de cours sur la grille horaire de façon à obtenir un horaire valide. L'utilisateur peut, si désiré, modifier l'horaire par la suite. L'horaire construit est toujours valide (sauf s'il y a modification par l'utilisateur). |

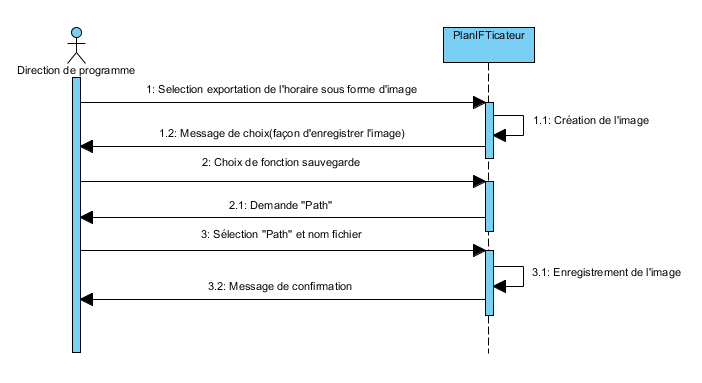
### DSS – Planification automatique



## Exportation de l’horaire sous forme d’image

|  |  |
| --- | --- |
| Cas d'utilisation: | **Exportation de l'horaire sous forme d'image.** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme et étudiants** |
| Préalables: | Le logiciel est en fonction. |
| Garanties en cas de succès | Avoir une image de l'horaire |
| Scénario principal: | 1. L'utilisateur sélectionne « exporter l'horaire sous forme d'image ». |
| 2. Le logiciel affiche une fenêtre avec deux choix:  -Enregistrer l'image;  -Presse-papier. |
| 3a. Si l'utilisateur choisit d’enregistrer l'image, il choisit le chemin où il désire enregistrer l'image. |
| 3b. Si l'utilisateur choisit presse-papier, l'image est ajoutée au presse-papier de l'ordinateur. |
| Note scénario principal: | --- |

### DSS – Exportation de l’horaire sous forme d’image



## Affichage des statistiques

|  |  |
| --- | --- |
| Cas d'utilisation: | **Affichage des statistiques** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme et étudiants** |
| Préalables: | Le logiciel est en fonction. |
| Garanties en cas de succès | Affichage de diverses statistiques en fonction de l'horaire construit. |
| Scénario principal: | 1. L'utilisateur appuie sur le bouton « statistiques ». |
| 2. Une fenêtre est affiché avec les diverses statistiques calculées en fonction de l'horaire construit. |
| Note scénario principal: | Les statistiques sont affichées même si l'horaire n'est pas complètement terminé. Il y aura un message qui s'affichera disant à l'utilisateur que les statistiques ne sont pas optimales puisque l'horaire n'est pas complètement terminé. |

### DSS – Affichage des statistiques

## Prendre des notes sur l’horaire

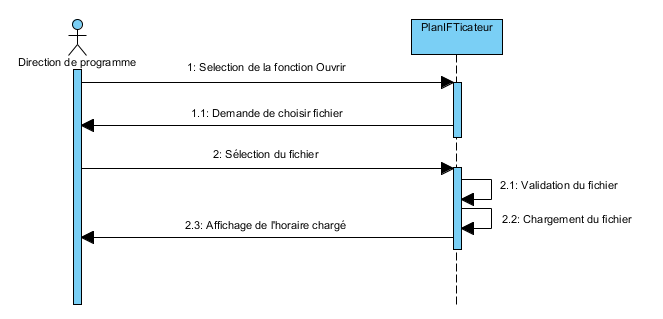
|  |  |
| --- | --- |
| Cas d'utilisation: | **Prendre des notes sur l'horaire** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme et étudiants** |
| Préalables: | Le logiciel est en fonction. |
| Garanties en cas de succès | Écrire des notes dans un champ texte concernant l'horaire. Ces notes sont enregistrées et modifiables. |
| Scénario principal: | 1. L'utilisateur appuie sur le bouton « notes » |
| 2. Une fenêtre est ouverte. |
|  | 3. L'utilisateur prend les notes qu'il veut. |
|  | 4. L'utilisateur ferme la fenêtre. |
| Note scénario principal: | Les notes prises précédemment sont affichées dans le champ de texte et il est possible de les modifier. Les notes sont enregistrées à la fermeture de la fenêtre. Cependant, si l'utilisateur n'enregistre pas l'horaire construit, les notes seront perdues (les notes sont associées aux fichiers importés initialement). |

### DSS – Prendre des notes sur l’horaire

## Charger un horaire

|  |  |
| --- | --- |
| Cas d'utilisation: | **Charger un horaire.** |
| Système: | Logiciel PlanIFTicateur |
| Acteurs | Direction de programme (L'individu qui construit l'horaire) |
| Parties prenantes et intérêts: | **Direction de programme** |
| Préalables: | Le logiciel est en fonction. L'utilisateur possède les fichiers de sauvegarde. |
| Garanties en cas de succès | Récupérer un horaire sauvegardé et l’afficher à l’écran. |
| Scénario principal: | 1. L'utilisateur sélectionne: Ouvrir |
| 2. L'utilisateur entre le chemin du fichier à importer |
| 3. L'utilisateur entre le nom du fichier. |
| 4. Le fichier est chargé avec succès |
| 5. Le logiciel affiche la liste des activités de cours dans la partie de droite de la fenêtre et l’horaire dans la partie gauche. |
| Note scénario principal: | Dans ce cas, il n'y a pas d'erreur, et les fichiers sont chargés avec succès. Bien entendu, il faut que les 2 fichiers se trouvent au même endroit (même chemin et portent le même nom (avec les extensions appropriées : \*.COU et \*.CHE). |
| Scénario alternatif: | 1. L'utilisateur sélectionne: Ouvrir |
| 2. L'utilisateur entre le chemin du fichier à importer |
| 3. L'utilisateur entre le nom du fichier. |
| 4. Le fichier est introuvable |
| 5. Le système retourne un message d'erreur mentionnant l'échec du chargement et demande également de vérifier le chemin choisi et le nom du fichier. |
| 6. L'utilisateur peut ensuite recommencer le processus. |

### DSS – Charger un horaire

****

## Spécifications supplémentaires

* Le programme ne devra, en aucun cas, créer un horaire invalide;
* Le temps de réponse du programme devra être quasi instantané;
* Le programme devra avoir une interface simple et accessible à tous les utilisateurs;
* Le programme sera développé pour la plateforme PC et nous envisagerons, s’il est possible et dans les délais convenus, la possibilité d’avoir un programme multiplateforme.
* Interface multi-utilisateur : Possible développement d’une application sur un serveur web qui est accessible de partout dans le monde et à tout moment. Ceci peut dépasser le cadre d'une seule université. La sécurité pour l'authentification peut être gérée par un serveur dédié.

## Glossaire

**Algorithme**

Suite de règles permettant de résoudre un problème.

**Application**

Programme ou ensemble de programmes visant à aider un utilisateur d’un ordinateur dans le traitement d’une tâche précise. *(Réf. : Larousse.fr)*

**Cas d’utilisation**

Relation établie entre une fonctionnalité et, soit un acteur ou une autre fonctionnalité.

**Chemin**

Emplacement précis sur un disque dur définit par la suite ordonnée des dossiers pour avoir accès au fichier recherché.

**Diagramme de séquence système (DSS)**

Représentation des interactions chronologique entre un système et ses acteurs. *(réf. : Wikipédia)*

**Domaine d’affaires**

Référent pour l’ensemble des processus d’affaires d’un projet ou d’une organisation. Englobe les entités, les acteurs et autres participants de ces processus.

**« Drag and drop »**

***(Terme anglais)*** En informatique, processus durant lequel un usager d’un ordinateur sélectionne un objet, au moyen d’une souris, en maintenant enfoncé le bouton gauche de cette dernière afin de déplacer le dît objet et de le relâcher à l’endroit voulu en relâchant également le bouton de sa souris.

**Fenêtre**

Zone d’affichage d’informations d’un programme.

**Fonction**

Bloc d’une séquence d’instructions visant un but, une fonctionnalité précise.

**Logiciel**

Ensemble d’instructions et de règles interprétable par un ordinateur.

**Multiplateforme**

Fonctionnant sur plusieurs plateformes, soit plusieurs ordinateurs/systèmes d’exploitation différents.

**Multiutilisateur**

Offrant la possibilité à plusieurs usagers d’interagir simultanément sur le même logiciel.

**PlanIFTicateur**

Nom du logiciel en développement pour le projet contenu dans ce document.

**Plateforme PC**

Liaison entre un ordinateur personnel munit d’un processeur spécifique et du système d’exploitation *Windows*.

**Presse-papier**

Fonction intégrée dans tous les systèmes d’exploitation stockant des données que l’on souhaite déplacer ou copier.

**Programme**

Succession d’instructions qu’un ordinateur peut exécuter afin d’accomplir des opérations.

**Serveur dédié**

Système informatique dont l’ensemble des ressources est dédié à un seul utilisateur.

**Serveur web**

Système informatique qui a pour fonction la publication de sites web à la demande d’un autre système.

**Session**

Période de 15 semaines durant laquelle un étudiant de l’université doit accomplir les objectifs de ses cours.

# III) Gestion de projet

## Référence des taches

### Nature des données

* Réaliser L’architecture complète du système. **(A1)**
* Extraire les fichiers .COU contenant la liste des activités à mettre à l’horaire et sauvegarder les données en RAM. **(A2)**
* Extraire les fichiers .CHE contenant les grilles de cheminement à respecter et sauvegarder les données en RAM. **(A3)**
* Synchroniser le chargement des fichiers .COU et .CHE**(A4)**

### Gestion des fichiers

* Enregistrer l’horaire (le fichier précédent est supprimé). **(B1)**
* Option « enregistrer sous » pour l’horaire est possible. **(B2)**
* Exporter l’image de l’horaire dans un format image. **(B3)**

**N.B :** Les fichiers .COU et .CHE sont au format CSV mais il faut faire attention au fait que dépendamment de la configuration de l’ordinateur sur lequel ils ont été créés, les séparateurs de champs et les séparateurs décimaux peuvent être différents. Ce détail doit être géré de façon transparente par l’application. Le problème peut être géré par une analyse automatique de l’entête de fichier.

### Interface utilisateur

* Diviser l’écran en deux zones. L’espace à gauche contient la grille horaire en construction. La zone de droite contient la liste des cours non encore assignés à la grille de même que l’information pertinente associée à ce cours. **(C1)**
* Attribuer une couleur à chaque activité en fonction de son type. **(C2)**
* À l’aide de la souris, un mouvement de glisser-lâcher (« drag and drop ») permet de : (1) déplacer un cours de l’espace de droite vers la grille horaire (et inversement); (2) déplacer un cours dans la grille horaire. **(C3)**
* En tout temps, un indicateur à l’écran indique si l’horaire est valide. Si ce n’est pas le cas, la liste des erreurs est affichée. On doit pouvoir modifier l’horaire tout en voyant la liste (qui se met à jour en temps réel). **(C4)**
* Possibilité de faire apparaitre une zone (ou une fenêtre) permettant de prendre des notes concernant l’historique de modification du fichier. **(C5)**

### Validation de l’horaire

* Possibilité d’activer/désactiver une fonction de vérification automatique de la validité d’horaire. Ainsi, lors du déplacement d’un cours, toutes les cases de la grille horaire qui ne sont pas accessibles (parce que les contraintes l’interdisent) deviennent grisées et il est impossible d’y placer le cours.

De cette manière, l’horaire affiché est en tout temps valide (sauf si les données chargées étaient elle-même invalides). **(D)**

### Planification automatique

* Placer de manière automatique dans la grille horaire tous les cours qui se trouvent dans l’espace à droite (bouton « planification automatique »). **(E)**

### Affichage de statistiques

* Nombre maximal de cours qu’un étudiant pourrait avoir la même journée Cet indicateur est à calculer pour chaque jour de la semaine. **(F1)**
* Nombre moyen de cours qu’un étudiant devrait avoir la même journée (en supposant qu’il y a le même nombre d’étudiant pour chacune des grilles de cheminement fournies). Cet indicateur est à calculer pour chaque jour de la semaine. **(F2)**
* Nombre de cours par jour pour chacun des jours de la semaine. **(F3)**
* Indice de congestion de la circulation : % des cours qui débutent à 8h30. **(F4)**
* Indice de covoiturage (voir spécifications du projet). **(F5)**

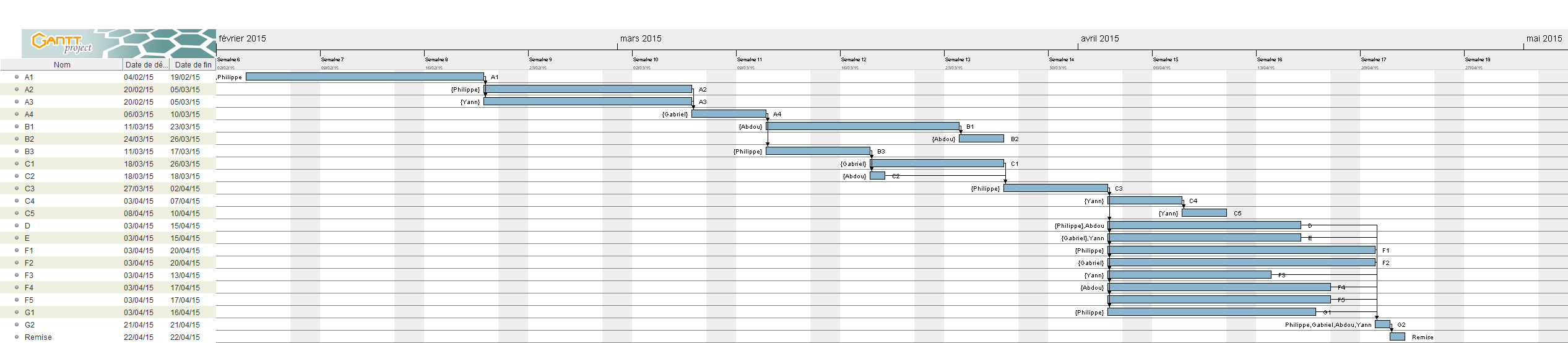
### Divers

* Lorsqu’on déplace la souris dans la grille horaire, le jour et l’heure associés à la case sous la souris s’affichent dans la barre d’état de l’application. **(G1)**
* Pour toute comparaison de chaîne de caractère, l’application ne doit pas être sensible à la casse. **(G2)**

## Assignation des ressources

|  |  |  |
| --- | --- | --- |
| **Tâche** | **Prédécesseurs** | **Ressources** |
| **Nature des données** | | |
| A1 | - | Toutes les ressources |
| A2 | A1 | Philippe |
| A3 | A1 | Yann |
| A4 | A2, A3 | Gabriel |
| **Gestion des fichiers** | | |
| B1 | A4 | Abdou |
| B2 | B1 | Abdou |
| B3 | A4 | Philippe |
| **Interface utilisateur** | | |
| C1 | B3 | Gabriel |
| C2 | B3 | Abdou |
| C3 | C1, C2 | Philippe |
| C4 | C3 | Yann |
| C5 | C4 | Yann |
| **Validation de l’horaire** | | |
| D | C3 | Philippe, Abdou |
| **Planification automatique** | | |
| E | C3 | Gabriel, Yann |
| **Affichage de statistiques** | | |
| F1 | C3 | Philippe |
| F2 | C3 | Gabriel |
| F3 | C3 | Yann |
| F4 | C3 | Abdou |
| F5 | C3 | Gabriel |
| **Divers** | | |
| G1 | C3 | Philippe |
| G2 | D, E, F1, F2, F3, F4, F5, G1 | Toutes les ressources |

## Diagramme de Gant



\*Voir fichier « ***diagrammeDeGant.gan »***

## Budget

Le budget total estimé est de 165 jours, pour un développement allant du 04 février 2015 au 23 Avril 2015. Les détails par tâche sont visibles sous dessous.

|  |  |
| --- | --- |
| **Tâche** | **Durée (jours)** |
| A1 | 12 |
| A2 | 10 |
| A3 | 10 |
| A4 | 3 |
| ***Nature des données*** | **45** |
| B1 | 9 |
| B2 | 3 |
| B3 | 5 |
| ***Gestion des fichiers*** | **17** |
| C1 | 7 |
| C2 | 1 |
| C3 | 5 |
| C4 | 3 |
| C5 | 3 |
| **Interface utilisateur** | **19** |
| D | 9 |
| ***Validation de l’horaire*** | **9** |
| E | 9 |
| ***Planification automatique*** | **9** |
| F1 | 12 |
| F2 | 12 |
| F3 | 7 |
| F4 | 11 |
| F5 | 11 |
| ***Affichage de statistiques*** | **53** |
| G1 | 10 |
| G2 | 1 |
| ***Divers*** | **11** |
|  |  |
| **TOTAL** | **165** |