

**PlanIFTicateur**

**Travail pratique 2**

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 3](#_Toc412741802)

[Interfaces utilisateurs 4](#_Toc412741803)

[Fenêtre principale 4](#_Toc412741804)

[Fenêtre statistiques 5](#_Toc412741805)

[Diagramme des classes de conception 6](#_Toc412741806)

[Diagramme de package 7](#_Toc412741807)

[Diagrammes de séquence 8](#_Toc412741808)

[Annexe 9](#_Toc412741809)

[Modèle du domaine 9](#_Toc412741810)

[Modèle des cas d’utilisation 9](#_Toc412741811)

[Glossaire 9](#_Toc412741812)

[Gestion de projet 9](#_Toc412741813)

# Introduction

La construction des horaires de session est une tâche difficile pour les directions de programme. À l’aide du logiciel dont nous avons le mandat, 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.

Ce rapport consiste à présenter le modèle de conception et l’architecture logicielle de planIFTicateur. Cette analyse sera basée sur les éléments préalablement réalisés.

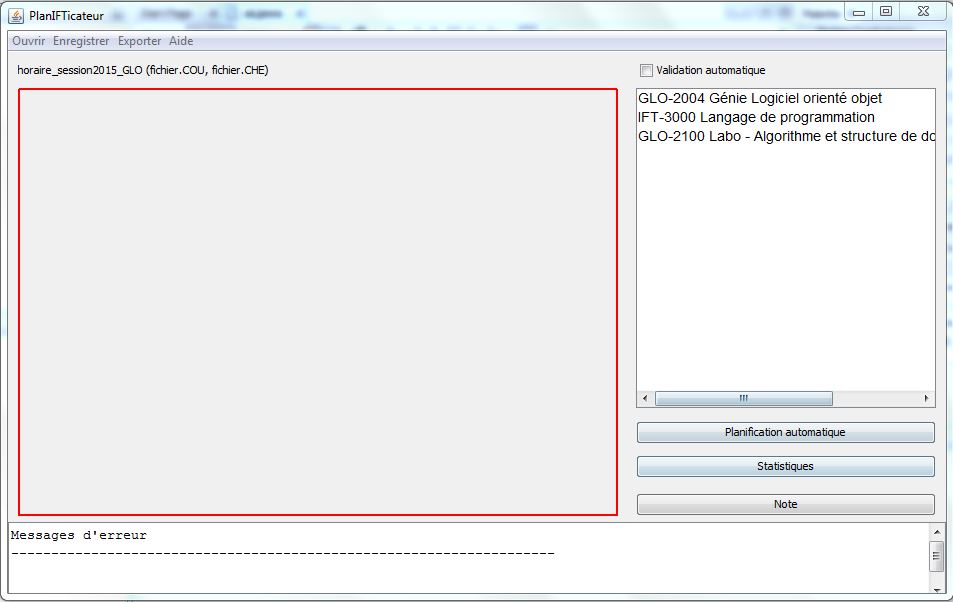
La section suivante illustre les principales interfaces utilisateurs, qui permettront d’interagir avec le système. Les diagrammes de classe de conception sont par la suite présentés. Il s’agit de l’architecture qui sera utilisée pour implanter notre solution en langage Java. Ensuite, une autre section est consacrée à expliciter la communication entre les différents packages du système.

Dans une quatrième partie nous établirons les diagrammes de séquence permettant d’accroitre la compréhension de notre système. Finalement, une dernière section vient mettre à jour la gestion du projet ainsi que les éléments du rapport précédent.

# Interfaces utilisateurs

Les interfaces utilisateurs doivent être le plus ergonomique possible. L’utilisation du principe du K.I.S.S. (Keep It Simple and Stupid) nous a permis d’en arriver au prototype suivant. Les informations généralement recherchées par les utilisateurs sont concentrées au haut de la fenêtre, tandis que les autres sont positionnées de manière à respecter les contraintes du client.

## Fenêtre principale

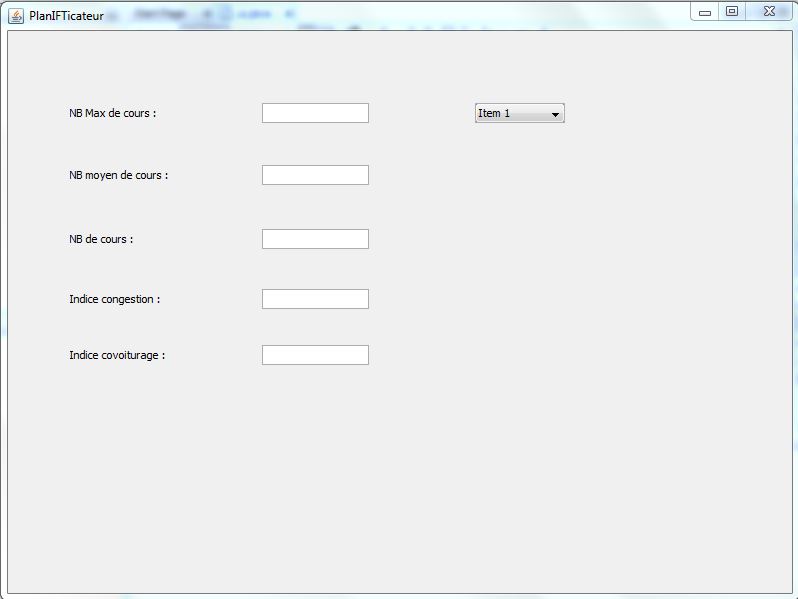


La fenêtre principale est composée des parties suivantes :

* Une barre en haut de l’écran, avec des boutons à fonctionnalités triviales.
* La grille des activités à l’horaire
* La liste des activités non attribuées
* Une liste d’erreur en cas d’horaire invalide
* Un bouton planification automatique pour activer la planification automatique
* Un bouton note pour que l’usager puisse entrer des notes
* Un bouton statistiques qui conduit au prototype qui suit.

## Fenêtre statistiques

La liste des statistiques est automatiquement affichée, et l’utilisateur peut changer le jour désiré à l’aide d’un menu déroulant.



# Diagramme des classes de conception

Les classes de conception sont le cœur du système que nous développerons dans les prochaines semaines. Notre équipe propose une solution axée sur la clarté et la compréhension. Notre architecture facilitera ainsi la maintenance du logiciel.

Notre solution est présentée à travers le diagramme UML suivant :

# Diagramme de package

# Diagrammes de séquence

# Annexe

## Modèle du domaine

## Modèle des cas d’utilisation

## Glossaire

## Gestion de projet

***Voir diagrammeDeGant.gan***