

# Projet STCal

## Rapport général

Florian Barrois  
Mehdi Loisel

Nicolas Devillers  
Jean Mercadier  
Willeme Verdeaux

Valentin Jeanroy  
Ismail Taleb



# Sommaire

<b>I</b>	<b>Introduction</b>	<b>4</b>
<b>II</b>	<b>Présentation du sujet</b>	<b>6</b>
<b>III</b>	<b>Cahier des charges</b>	<b>8</b>
<b>1</b>	<b>Fonctionnalités</b>	<b>9</b>
<b>2</b>	<b>Développement et contraintes</b>	<b>10</b>
2.1	Première étape . . . . .	10
2.1.1	Développement . . . . .	10
2.1.2	Contraintes . . . . .	10
2.2	Deuxième étape . . . . .	10
2.2.1	Développement . . . . .	10
2.2.2	Contraintes . . . . .	11
2.3	Troisième étape . . . . .	11
2.3.1	Développement . . . . .	11
2.3.2	Contraintes . . . . .	11
2.4	Quatrième étape . . . . .	11
2.4.1	Développement . . . . .	11
2.4.2	Contraintes . . . . .	12
<b>IV</b>	<b>Réalisation</b>	<b>13</b>
<b>1</b>	<b>Étape 1 : l'interface graphique</b>	<b>15</b>
<b>2</b>	<b>Étape 2 : la création des stages</b>	<b>17</b>
2.1	Une section . . . . .	17
<b>3</b>	<b>Chap 2</b>	<b>18</b>
3.1	Une section . . . . .	18
<b>V</b>	<b>Difficultés rencontrées</b>	<b>19</b>
<b>4</b>	<b>Difficultés techniques</b>	<b>20</b>

<b>5</b>	<b>Répartition des tâches</b>	<b>21</b>
<b>6</b>	<b>Organisation</b>	<b>22</b>
<b>VI</b>	<b>Conclusion</b>	<b>23</b>

Première partie

Introduction

Dans le cadre de notre projet tutoré, nous avons dû réfléchir à un secteur d'activité nécessitant des outils informatiques relativement simples mais essentiels. Ainsi, au vu du besoin éprouvé par nos enseignants de travailler avec des outils plus simples, nous avons choisi de réaliser un progiciel à utilité pédagogique. Cette application aura pour but de faciliter la gestion des stages des étudiants de quatrième semestre ainsi que la mise en place d'un calendrier des soutenances orales des étudiants présentant leur stage à leurs professeurs.

L'application possédera bien sûr des fonctionnalités précises, correspondant aux besoins des enseignants s'occupant des stages, mais ces fonctionnalités seront également conçues de manière à ce que la plupart des établissements d'enseignement supérieur et des organismes gérant des stages puissent s'en servir de façon intuitive. Poursuivant l'objectif de produire un travail complet tout en se mettant dans la peau des utilisateurs, les moyens les plus modernes et adaptés, du point de vue technique, et les plus faciles d'utilisation, du côté de l'utilisateur, seront mis en place pour satisfaire au maximum la demande concernant l'organisation de stages.

Nous détaillerons ainsi dans ce rapport toute notre réflexion pour répondre à cette problématique, expliquerons nos choix, et parlerons des avantages et des éventuels inconvénients liés à ces derniers. Des illustrations seront également placées tout au long du rapport dans le but d'apporter une meilleure compréhension et une représentation concrète du travail réalisé.

Nous allons donc étudier les différents aspects du projet, en commençant par une brève présentation du sujet. Puis nous nous tournerons vers le cahier des charges, qui comprendra les fonctionnalités du produit ainsi que les contraintes à respecter. Ensuite, nous observerons plus en détail comment s'est déroulée la réalisation des différentes étapes. Enfin nous aborderons les difficultés survenues et concluerons sur le résultat obtenu et sur le projet en lui-même.

Deuxième partie

Présentation du sujet

Depuis de nombreuses années, les professeurs de notre établissement, et sûrement bien d'autres, s'occupent des stages en s'adaptant aux outils existants. En effet, à ce jour, la méthode employée pour réaliser cette tâche se compose de la gestion d'une part des informations, et d'autre part du planning des soutenances, en utilisant respectivement des logiciels tels que Microsoft Excel, et les fonctionnalités de calendrier de Google. La mission dont nous nous sommes acquittés consiste donc à aider les responsables de stage, non pas par la recherche d'outils plus appropriés, mais par la création d'une application qui sera spécifiquement prévue à cet usage et qui demeurera parfaitement adaptée aux besoins des usagers, notamment du personnel enseignant.

Ainsi la difficulté de ce défi est conséquente, puisque nous ne connaissons aucun outil similaire existant pouvant servir de référence ou permettant d'étayer notre réflexion. Notre groupe de sept étudiants a donc été chargé de trouver la bonne démarche à suivre en mettant à profit toutes les connaissances, qu'elles aient été acquises en cours ou à la suite de recherches liées au projet. Les connaissances utilisées relèvent de la mise en place d'une interface graphique, de la gestion de classes et de bases de données, en plus de toutes les notions apprises liées au langage utilisé, à savoir le Java.

Le résultat ne consistera en aucun cas en une plate-forme de consultation des stages. L'application développée sera mise à disposition exclusive des enseignants et du personnel responsable des stages. Les étudiants concernés n'auront quant à eux aucun accès à l'application, que ce soit pour ajouter, supprimer, consulter ou éditer des données.



Troisième partie

Cahier des charges

# Chapitre 1

## Fonctionnalités

L'application Stcal devra simplifier la gestion des stages des étudiants par les professeurs grâce à plusieurs fonctionnalités, réparties ci-dessous en quatre étapes :

1. La création de stages.
2. La création et la gestion d'un emploi du temps.
3. La création d'une base de données.
4. L'existence d'un système de sauvegarde.

## Chapitre 2

# Développement et contraintes

Le développement s'effectuera en quatre étapes, qui contiendront différentes tâches. Ces étapes et tâches seront ordonnées de manière à ce que le côté fonctionnel de l'application soit prioritaire :

### 2.1 Première étape

#### 2.1.1 Développement

L'application devra avant tout considérer des étudiants et des professeurs. Le produit permettra ainsi l'import de fichiers contenant les données. Ces dernières seront par la suite utilisées lors de la création de couples professeur/étudiant.

#### 2.1.2 Contraintes

Détaillons maintenant les différentes contraintes à respecter concernant cette étape.

A partir d'une liste de professeurs et d'une liste d'étudiants existantes, un bouton devra permettre d'associer un étudiant à un professeur. N'ayant qu'un tuteur de stage, chaque étudiant sera relié à un et un seul enseignant. En revanche, un enseignant peut posséder la fonction de responsable de stages pour plusieurs élèves. Chaque enseignant pourra donc être relié à aucun, un ou plusieurs étudiants.

### 2.2 Deuxième étape

#### 2.2.1 Développement

L'application comportera une fonction de création d'emploi du temps vide et une seconde permettant le remplissage de celui-ci avec les soutenances de chacun des étudiants. Les plages horaires réservées contiendront les informations suivantes :

- Le nom de l'enseignant tuteur ;
- Le nom de l'enseignant candide ;
- Le nom de l'étudiant ;
- Le numéro de la salle d'entretien.

Le progiciel possédera également une option d'export du calendrier généré au format ICS compatible avec Google et iCal.

### 2.2.2 Contraintes

Le produit devra être capable de résoudre les cas suivants en affichant un message d'erreur et en ignorant la dernière action effectuée :

- en cas de création d'une soutenance à un horaire incompatible pour l'enseignant tuteur ;
- en cas de création d'une soutenance à un horaire incompatible pour l'enseignant candide ;
- en cas de création d'une soutenance ayant lieu dans une salle déjà réquisitionnée à cet horaire ;
- si toutes les salles d'entretien venaient à être occupées à un horaire donné, cette plage horaire devra être inaccessible pour placer d'autres soutenances.

## 2.3 Troisième étape

### 2.3.1 Développement

Le produit possédera un outil de création de base de données dans laquelle les stages pourront être stockés. Les informations enregistrées seront identiques à celles affichées sur les horaires de soutenances. La base de données créée pourra ensuite être reliée à un client léger, comme un site web dynamique, et sera consultable par les étudiants.

### 2.3.2 Contraintes

La contrainte principale de cette étape consiste à éviter la présence de doublons dans la base données. Le progiciel devra donc afficher un message d'erreur si les données telles que le nom et le prénom d'une personne, étudiant ou enseignant, entrées dans l'application sont déjà présentes dans la base de données.

## 2.4 Quatrième étape

### 2.4.1 Développement

L'application contiendra un système de sauvegarde : lors de l'ouverture de l'application par l'utilisateur, le système effectuera une restauration de session qui permettra l'affichage de la dernière page affichée avant la fermeture de l'application lors la session de travail précédente.

### 2.4.2 Contraintes

Lors de la première utilisation du progiciel, l'utilisateur devra entrer les paramètres correspondant à la base de données qui stockera les informations, à savoir le nom du serveur, le port utilisé, le nom d'utilisateur et le mot de passe. Une fois les champs renseignés, ces paramètres seront également mémorisés et la restauration de session devra passer par un accès cette base de données pour afficher les informations précédemment enregistrées. Le programme sera donc contraint de rechercher l'existence d'un fichier de configuration dans l'ordinateur afin de savoir si l'application est utilisée ou non pour la première fois. Si c'est le cas, une fenêtre contenant les différents paramètres à renseigner mentionnés précédemment apparaît, et les valeurs entrées sont mémorisées dans un nouveau fichier de configuration.

Ces étapes consistent ainsi le fil conducteur du projet. Il faut également noter l'existence d'une autre étape effectuée en parallèle : la réalisation de l'interface graphique. En effet, celle-ci offre la possibilité d'entrevoir le rendu final qui sera livré au client et facilite également la visibilité quant à l'évolution du projet. Elle permet aussi d'incorporer directement dans le progiciel les fonctions programmées en interne au fur et à mesure que le projet progresse et tient ainsi le rôle de de plate-forme de tests.

Quatrième partie

Réalisation

Nous avons abordé la problématique et les étapes du sujet. Quittons maintenant la théorie et les prévisions pour entrer dans le concret. Cette partie va détailler la réalisation effective du projet. Les quatre étapes principales, ainsi que celle concernant l'interface graphique, vont pouvoir être rattachées à un véritable support.

Les futurs utilisateurs du produit seront relativement nombreux, par conséquent les systèmes d'exploitation peuvent varier, avec notamment Windows, connu et employé couramment par une grande communauté de personnes, Linux, qui est très utilisé dans le cadre de nos études, et Mac OS, qui devient de plus en plus répandu. C'est pourquoi toute l'application sera programmée en Java, ce qui permettra l'usage du progiciel sous n'importe quelle plate-forme et favorisera une programmation orientée objet. Tout au long de cette partie, les outils auxquels nous avons fait appel et qui se seront mentionnés se trouveront donc être des fonctions et des objets instanciés en Java. Quant à l'interface graphique, elle sera implémentée à partir la bibliothèque graphique Java Swing.

# Chapitre 1

## Étape 1 : l'interface graphique

L'interface de l'application contient une barre de menu donnant accès aux multiples possibilités offertes par l'application. Le corps de l'interface affiche quant à lui trois onglets principaux dédiés aux différentes fonctionnalités du progiciel.

### Le menu

La barre de menu est composée elle-même de plusieurs sous menus.

Le premier sous menu est intitulé "Fichier". Il offre la possibilité d'importer une liste d'étudiants et une liste de professeurs depuis un fichier au format CSV. Il présente également une fonction d'export, notamment pour le calendrier généré. Cette fonction exporte le planning des soutenances au format ICS, qui permet la compatibilité avec les calendriers Google notamment. La dernière option de ce menu sert à fermer l'application.

Le deuxième sous menu a pour libellé "Actions", et propose de lier un étudiant à un enseignant. Cette fonctionnalité ne fonctionne que si l'utilisateur a auparavant sélectionné l'étudiant et l'enseignant concernés.

Le troisième champ est nommé "Préférences". Il est dédié au côté Base de Données du programme. Un de ses onglets possède le libellé "Base de données" et ouvre une fenêtre permettant à l'utilisateur de mettre à jour les paramètres de la base de données énumérés dans la quatrième étape du cahier des charges. Un autre onglet donne l'opportunité de réinitialiser la connexion à la base de données correspondant aux paramètres en vigueur.

La quatrième et dernière rubrique propose un champ "Aide", qui affiche le manuel d'utilisateur, ainsi qu'un champ qui redirige l'usager vers une brève présentation du projet STCal.



## Les onglets

Trois onglets sont accessibles à l'utilisateur.

### L'onglet Lier

La première page est destinée à afficher les acteurs principaux, à savoir les étudiants et les enseignants. En effet, on peut diviser cette page intitulée "Lier" en trois parties.

La partie de gauche affiche une liste de professeurs importée depuis un fichier CSV, et de la même manière, la partie de droite montre une liste d'étudiants provenant d'un deuxième fichier au format CSV. Les personnes, étudiants et enseignants, contenues dans ces listes ne sont alors caractérisées que par leur nom et leur prénom.

La partie centrale contient les appels aux fonctions nécessaires à l'import des fichiers d'étudiants d'une part, et à l'import des fichiers d'enseignants d'autre part. Ces fonctions demeurent accessibles grâce à des boutons portant le libellé correspondant. Ces boutons ne sont disponibles qu'une fois chacun : après l'import d'un fichier d'étudiants, respectivement d'enseignants, le bouton consacré à cette tâche n'est plus présent. Toutefois, il reste possible de réaliser de nouveaux imports à partir de la barre de menu située en haut de l'interface de l'application. Les personnes précédemment affichées dans la liste des objets du type de l'import immédiatement effectué sont alors remplacées par les valeurs du nouveau fichier.

La zone centrale présente également une portion révélant le statut et les nom et prénom de la dernière personne sélectionnée. Une fois un étudiant et un professeur sélectionnés, un bouton de validation apparaît. Après avoir enclenché ce bouton, les identifiants de la dernière personne sélectionnée disparaissent pour laisser place à la phrase de confirmation "Stage créé". En effet, on considérera comme stages les couples étudiant/enseignant effectués. Un stage est alors formé et consultable dans l'onglet "Stages". L'étudiant concerné est alors effacé de la liste, en revanche, le professeur conserve sa disponibilité pour s'occuper des stages d'autres étudiants. Aucun étudiant n'étant alors sélectionné, le bouton de validation reste inaccessible jusqu'à ce qu'un enseignant et un nouvel étudiant soient sélectionnés.

### L'onglet Stage

L'onglet "Stage" contient également deux sections.

La partie gauche est consacrée à l'affichage des différents stages créés dans l'onglet précédent. Cette affichage est présenté sous forme d'arborescence. Le dossier racine a pour titre "Listes des stages". Il contient alors plusieurs sous-dossiers correspondant aux professeurs liés à au moins un étudiant. Ces sous-dossiers possèdent des fichiers représentant les étudiants liés à l'enseignant en question.

Dans cette moitié gauche se trouve également un bouton permettant la suppression d'un stage. Ce bouton est accessible uniquement dans le cas où un étudiant est sélectionné. Les dossiers-professeurs ainsi que le dossier racine ne peuvent être supprimés. Le dossier racine est donc toujours présent dans cette section. Un dossier, représentant un professeur ne disparaît que lorsqu'il ne contient aucun étudiant. Ainsi, en cas de suppression du dernier stagiaire rattaché à un enseignant, le dossier symbolisant ce dernier disparaît en même temps que le fichier-étudiant. De cette manière, un dossier n'est jamais vide.

Lorsqu'un étudiant est supprimé de cette arborescence, le stage est effacé. L'étudiant réapparaît alors dans la liste de tous les étudiants située dans l'onglet "Lier" et redevient disponible pour une affectation à un professeur.

## Chapitre 2

# Étape 2 : la création des stages

Afin de créer les stages, il est d'abord nécessaire d'établir la liste des étudiants de l'année courante ainsi que la liste des professeurs de l'établissement. L'application ne possédera aucune fonctionnalité d'établissement de listes. En revanche, elle sera munie

### 2.1 Une section

## Chapitre 3

## Chap 2

### 3.1 Une section

Cinquième partie

**Difficultés rencontrées**

## Chapitre 4

# Difficultés techniques

## Chapitre 5

# Répartition des tâches

## Chapitre 6

# Organisation

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



Sixième partie

**Conclusion**