

Interface et Multimédia  
**IFT-215**

Travail Pratique 5

Amandine Fouillet - 14 130 638  
Frank Chassing - 14 153 710

5 avril 2015



# Table des matières

<b>Introduction</b>	<b>4</b>
<b>1 Présentation générale de l'interface</b>	<b>4</b>
1.1 Choix de l'interface . . . . .	4
1.1.1 Objectifs réalisés . . . . .	4
1.1.2 Respect des lois de Gestalt . . . . .	4
1.1.3 Choix des composants . . . . .	4
1.2 Aide à l'utilisation . . . . .	5
1.2.1 Gestion des erreurs . . . . .	5
1.2.2 Gestion de l'aide . . . . .	5
<b>2 Tests de notre application</b>	<b>5</b>
2.1 Protocole . . . . .	5
2.2 Résultats . . . . .	5
2.2.1 Test 1 . . . . .	5
2.2.2 Test 2 . . . . .	5
<b>3 Amélioration</b>	<b>5</b>
<b>4 Explication du code</b>	<b>5</b>
<b>Conclusion</b>	<b>5</b>

# Introduction

Après avoir réfléchi sur la conception de notre interface par le biais des Tps précédents, nous avons implémenté notre interface en suivant les objectifs et les contraintes que nous nous étions fixés. Dans un premier temps, ce rapport présente l'interface générale de notre application et les différents éléments d'interface qu'elle possède. Dans un second temps, il met en avant les phases de tests que nous avons été amené à réaliser pour obtenir différents retours. Enfin, nous verrons comment il est possible d'envisager des améliorations de notre application et nous expliciterons le code qui a permis d'implémenter notre application.

## 1 Présentation générale de l'interface

### 1.1 Choix de l'interface

#### 1.1.1 Objectifs réalisés

Pour réaliser notre interface nous nous sommes basés sur les objectifs que nous avons établis dans les Tps précédents. Nous voulions créer une interface permettant d'accueillir toutes les fonctionnalités de communication entre un étudiant et sa famille. Notre but principal était de réunir l'ensemble de ces fonctionnalités et de pouvoir naviguer rapidement entre elles. Dans notre interface finale, nous avons donc repris le menu vertical permettant cette navigation rapide. Nous retrouvons également les informations sur chacun des contacts que l'on a ajouté. Ceci permet à l'utilisateur de connaître l'actualité de ses contacts en temps réel. Cependant la fonction principale reste la communication. L'utilisateur a accès à une liste de communication et peut entamer à tout moment une conversation écrite, audio ou vidéo avec un ses contacts.

#### 1.1.2 Respect des lois de Gestalt

En faisant une premier ébauche de notre interface dans le Tp précédent, nous avons fait attention à ce que celle-ci respecte les lois de Gestalt. Nous retrouvons beaucoup la notion de fermeture dans notre interface par le biais de zone créée avec des contours noirs. Nous retrouvons également la notion de similarité et de proximité permettant de regrouper les éléments allant ensemble, notamment au niveau de la liste des communications et la liste des contacts, où chaque contact ou communication est présenté de la même façon. Nous pouvons également voir que plusieurs éléments respectent la loi de continuité, comme par exemple le menu vertical où les bulles suivent une même colonne.

#### 1.1.3 Choix des composants

Pour réaliser notre interface, nous avons utilisé plusieurs composants. Pour les fenêtres d'inscription et de connexion, nous avons implémenter des champs textes libre permettant à l'utilisateur de rentrer les informations qu'il souhaite. De plus, pour le champ réservé au mot de passe, nous avons utilisé un champ texte spécialisé permettant de ne pas percevoir le mot de passe tapé. En ce qui concerne la liste des contacts et des communications, nous avons choisi d'utiliser des listes déroulantes avec ascenseur permettant ainsi d'avoir une fenêtre assez petite (taille Smartphone) et de placer de nombreux contacts ou de créer de nombreuses conversations. Les différentes boutons de notre application sont des images ayant un évènement particulier. Nous voulions personnaliser davantage notre application et nous ne pouvions donc utiliser de simples boutons qui ne nous permettaient pas d'obtenir le rendu que l'on voulait.

## **1.2 Aide à l'utilisation**

Nous avons implémenté plusieurs éléments d'interface qui permettent d'avertir les éventuelles erreurs que peut commettre l'utilisateur. Nous avons également ajouté des fonctionnalités permettant d'aider l'utilisateur à utiliser simplement notre application.

### **1.2.1 Gestion des erreurs**

Afin que l'utilisateur comprenne le sens de l'erreur qu'il a commise, nous avons décidé de l'avertir en utilisant des boîtes de dialogue explicite. Par exemple, dans la fenêtre permettant à l'utilisateur de s'inscrire, celui-ci doit absolument remplir les champs "Pseudo" et "Mot de passe" pour que l'inscription soit valide, sinon une boîte de dialogue l'avertit que le champ "Pseudo" ou "Mot de passe" est vide. De plus, le champ "Email" devra correspondre à une adresse Email valide sinon l'utilisateur verra une boîte de dialogue s'ouvrir l'avertissant que le format de l'Email est invalide, et cette boîte de dialogue donnera également à l'utilisateur la forme de l'Email attendue. En ce qui concerne la connexion à l'application, si les informations rentrées n'existent pas dans la base de données, une boîte de dialogue s'ouvrira indiquant à l'utilisateur que les informations de connexion ne sont pas exactes.

### **1.2.2 Gestion de l'aide**

Plusieurs éléments ont été intégré à l'application permettant d'obtenir de l'aide à tout moment. Tout d'abord, nous avons ajouté un bouton d'aide dans chaque fenêtre. Ce bouton ouvre une nouvelle fenêtre recueillant diverses informations permettant d'aider l'utilisateur à comprendre et utiliser les fonctions dans la fenêtre où il se trouve. Ce bouton existe à tout moment et empêche l'utilisateur de rester bloquer sur l'application.

Dans un second temps, nous avons décidé d'utiliser des ToolTipText. Un ToolTipText correspond au texte suivant notre curseur lors du survol d'un élément. Nous avons implémenté ces éléments sur les items qui ne sont pas explicites comme par exemple les boutons ayant seulement un icône. Bien que nous ayons choisi des icônes plus ou moins explicites, nous avons renforcé la compréhension de ces boutons en ajoutant des ToolTipText.

## **2 Tests de notre application**

### **2.1 Protocole**

### **2.2 Résultats**

#### **2.2.1 Test 1**

#### **2.2.2 Test 2**

## **3 Amélioration**

## **4 Explication du code**

## **Conclusion**

## Table des figures