

Octochat
Discuter sur le réseau local sans serveur

Alexis GIRAUDET Benjamin SIENTZOFF

26 novembre 2014

Table des matières

| | | |
|----------|---------------------------------------------|----------|
| 1 | Utilisation et fonctionnement global | 3 |
| 1.1 | Compilation du programme | 3 |
| 1.2 | Utilisation | 3 |
| 1.3 | Sous le capot | 3 |
| 2 | Patrons de conception | 4 |
| 2.1 | Observer | 4 |
| 2.2 | Factory method | 4 |
| 2.3 | Interpretor | 4 |

Introduction

Ce projet a été réalisé dans le cadre du cours *Objet et développement d'applications* dans lequel M. RICHOUX nous a enseigné l'utilisation des *Design Patterns*. L'ambition de ce projet ne s'arrête pas là, car nous souhaitons poursuivre le développement de notre programme. Le sujet de notre projet est la création d'un client de chat qui n'utilise pas de serveur principal comme c'est le cas pour ce genre d'application réseau. Le fonctionnement est détaillé plus loin.

Octochat, notre programme, est donc un client de chat qui n'a pas besoin de serveur pour fonctionner. Lancer le programme, choisissez un nom d'utilisateur est c'est parti.

1 Utilisation et fonctionnement global

Boost Pour des questions de dépendances et pour faciliter le développement du programme, notamment pour ce qui est du réseau, nous avons choisi d'utiliser la librairie *Boost*. L'utilisation de cette librairie nous permet également d'utiliser le système de compilation associé.

1.1 Compilation du programme

Compilation Pour commencer, le projet a besoin d'une installation locale de *boost*. Cette installation est faite lorsque le *makefile* est appelé pour la première fois, mais une installation en bonne et due forme est préférable. Procédez comme suit.

```
$ cd Octochat/  
$ ./bin/install-boost.sh  
$ ./bin/prepare-boost.sh  
$ make
```

La première commande permet d'aller dans le répertoire où se trouve notre projet. La seconde commande va télécharger *boost* dans le projet, cette étape peut être longue, comme la suivante¹. Ensuite on va *préparer boost* pour l'utiliser. C'est-à-dire qu'on va compiler les outils nécessaires. La dernière commande fait appel à l'outil *make* pour compiler notre projet. En réalité, le *makefile* se contente d'appeler l'outil de compilation de *boost*.

Si toutes ces étapes se sont bien passées, vous êtes maintenant en mesure d'utiliser Octochat.

1.2 Utilisation

Pour lancer l'application taper simplement `./octochat`

1.3 Sous le capot

1. C'est le moment d'aller chercher un café, si vous en n'avez plus.

2 Patrons de conception

2.1 Observer

2.2 Factory method

2.3 Interpretor

Conclusion

je conclu