

Fortunettes: Une Fonction d'Utilisabilité de Comportement pour les Systèmes Interactifs

Philippe Palanque, David Navarre,
Célia Martinie

ICS-IRIT, Université Paul Sabatier – Toulouse III
118, route de Narbonne, 31042 Toulouse,
France
{nom}@irit.fr

Kris Luyten

Expertise Centre for Digital Media,
Hasselt University—tUL—Flanders Make,
Wetenschapspark 2, Diepenbeek, BE 3590,
Belgium
kris.luyten@uhasselt.be

ABSTRACT

Cet article présente la fonction d'utilisabilité « fortunettes » qui offre la possibilité aux utilisateurs d'explorer les états futurs d'une application interactive. L'utilisateur peut déclencher le passage en mode futur, explorer le mode futur et décider de soit revenir à l'état de l'application avant l'exploration soit de valider les explorations et de déplacer l'état de l'application dans cet état futur.

Keywords

Fonction d'utilisabilité, Fortunettes, dialogue.

INTRODUCTION

Les fonctions d'utilisabilité sont proposées de façon pervasives dans les contributions scientifiques en IHM. Toutefois, elles ne sont pas identifiées comme telles et sont souvent présentées comme une technique d'interaction innovante.

Dans cet article de workshop nous présentons rapidement la fonction d'utilisabilité Fortunette dont la conception en termes de graphiques et d'interaction a été proposée dans [1] et la description du comportement formel a été proposée dans [2].

La prévisibilité (predictability [3]) de l'interface utilisateur a été identifiée comme une propriété extrinsèque importante des systèmes interactifs et contribuant à leur utilisabilité. Différentes alternatives de conception ont été proposées pour intégrer cette prévisibilité entre autres au moyen du critère ergonomique guidage [4] mis en œuvre de façon statique (par exemple en désactivant ou activant des interacteurs). La fonction d'utilisabilité fortunette propose une alternative à ce guidage statique en offrant un guidage dynamique offrant à l'utilisateur la possibilité d'explorer le futur de l'interaction en identifiant, par exemple, les désactivations et les activations d'interacteurs dans les interactions futures.

FORTUNETTES

Les fortunettes telles que proposées dans [1] ont à la fois une partie présentation (apparence graphique sur l'interface) et une partie dialogue (comportement).

La Figure 1 donne un aperçu du design graphique de la fonction fortunette pour un bouton de commande.



Figure 1. Vue graphique de la fonction fortunettes sur un objet interactif bouton. A gauche le bouton est actif et dans le futur il sera inactif (bordure pointillée). à droite, le bouton est actuellement inactif mais deviendra actif (bordure pleine en arrière-plan).

La Figure 2 présente le comportement de la fonction fortunette. Dans le cas nominal, l'utilisateur interagit normalement avec son application. A tout instant, il peut passer en mode fortunette et les objets graphiques interactifs vont être augmentés par des informations d'arrière-plan (voir Figure 1). Dans cet état d'exploration l'utilisateur peut soit revenir au mode précédent soit valider son action et faire évoluer l'application dans l'état futur (par exemple en cliquant sur un bouton actif).

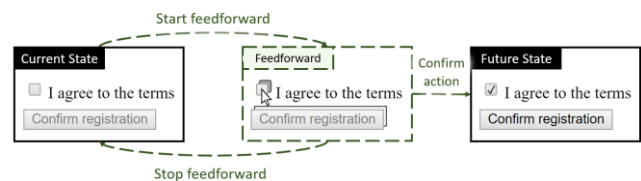


Figure 2. Comportement dynamique de la fonction fortunette. L'état courant (standard) à gauche, le passage dans l'état où la fonction fortunette est active (milieu) et l'état futur où la fonction fortunette est à nouveau désactivé.

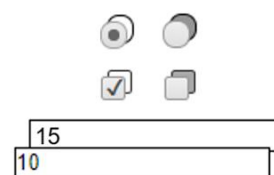


Figure 3. La présentation graphique de la fonction fortunette pour des objets interactifs de saisie de donnée

La Figure 3 donne un exemple de la représentation graphique des objets de saisie de donnée dans le mode fortunette. Pour

une paire de radio boutons, le fait de cliquer sur celui qui n'est pas sélectionné (à droite en haut de l'image) deviendra sélectionné et celui qui l'était ne le sera plus.

La zone de texte en bas de la figure contient une contrainte de valeur (cette valeur ne doit jamais être inférieure à 15) ce qui fait que si l'utilisateur saisi une valeur égale à 10, 15 sera affiché dans le futur).

IMPLICATION SUR L'INGENIEURIE

Rajouter la fonction d'utilisabilité fortunette à une application interactive accroît significativement la complexité du comportement de cette application. Tout d'abord les interacteurs utilisés doivent être augmentés pour :

- pouvoir afficher les informations en arrière plan
- réagir aux interactions pour passer en mode fortunette. Dans [1] ce passage se fait en exploitant les événements 'hover' sur les interacteurs ?

Cette modification des interacteurs a l'avantage de ne devoir être réalisée qu'une seule fois quel que soit le nombre d'applications mettant en œuvre la fonction fortunette.

Par contre, le comportement de l'application doit lui être modifié pour décrire précisément les évolutions de l'application en mode fortunette (ce qui se rajoute au comportement «standard» de l'application). Cette augmentation du comportement peut être très importante et complexe à réaliser, c'est pourquoi dans [2] les auteurs ont proposé une démarche systématique à base de modélisation formelle pour :

- Décrire les comportements standard et augmentés de façon complète et non-ambiguë ;
- Vérifier que le comportement fortunette ne modifie le comportement standard de l'application.

UNE APPLICATION TRES SIMPLE

Les figures de cette section présentent respectivement la présentation d'une application d'envoi de message (Figure 4), son comportement représenté par automate (Figure 5) et son comportement augmenté par la fonction d'utilisabilité fortunette (Figure 6).

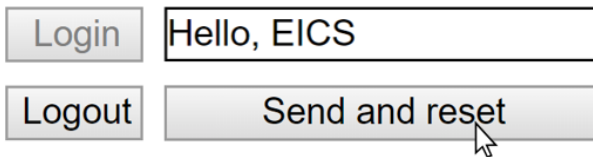


Figure 4. Une application simple permettant de se logger et d'envoyer des messages (une fois loggé)

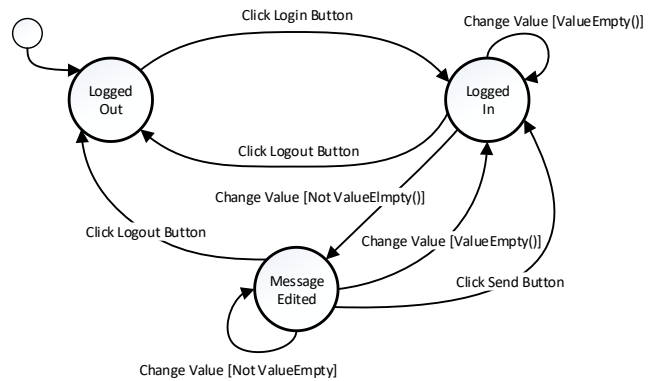


Figure 5. Comportement de l'application d'envoi de message

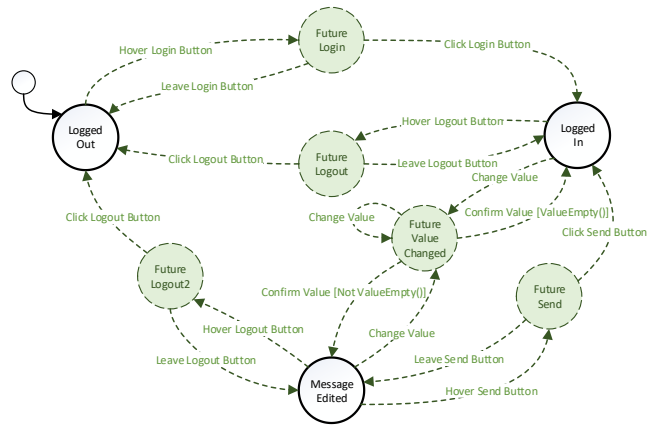


Figure 6. Comportement de l'application augmenté par le comportement fortunette

Cette dernière figure montre l'accroissement important du comportement et la nécessité de décrire formellement ces extensions, très difficile à entrelacer avec le comportement standard attendu de l'application.

CONCLUSION

Ce article de workshop a présenté et résumé la fonction d'utilisabilité fortunette et a pour objectif de lancer une réflexion sur l'ajout de telles fonctions. Tout d'abord, il est important d'apprécier l'importance de ces fonctions pour améliorer l'utilisabilité des applications interactives (principalement le facteur 'efficiency' et le facteur satisfaction). Il est aussi important de comprendre l'impact en ingénierie logicielle et de proposer des moyens systématiques pour aider à leur développement.

ACKNOWLEDGMENTS

Nous remercions les différentes personnes qui ont pris part à la conception de la fonction d'utilisabilité Fortunette et tout spécialement Sven Coppers et Davy Vanacken.

REFERENCES

1. Sven Coppers, Kris Luyten, Davy Vanacken, David Navarre, Philippe Palanque, and Christine Gris. 2019.

Fortunettes: Feedforward about the Future State of GUI Widgets. Proc. ACM Hum.-Comput. Interact. 3, EICS, Article 20 (June 2019), 20 pages. <https://doi.org/10.1145/3331162>.

2. David Navarre, Philippe Palanque, Sven Coppers, Kris Luyten, Davy Vanacken, Model-based Engineering of Feedforward Usability Function for GUI Widgets, Interacting with Computers, Volume 33, Issue 1,

January 2021, Pages 73–91, <https://doi.org/10.1093/iwcomp/iwab014>

3. Alan Dix. Formal Methods for Interactive Systems. Academic Press 1991. ISBN 0-12-218315-0.
4. Christian Bastien & Dominique Scapin (1993). Ergonomic criteria for the evaluation of human-computer interfaces. (Technical report N° 156).

The columns on the last page should be of approximately equal length.