PC Ready: Iterazione 2

Sommario

[Introduzione alla seconda iterazione 1](#_Toc30028115)

# Introduzione alla seconda iterazione

Terminata la prima iterazione si è ottenuta un programma in grado di far assemblare una configurazione, sia ad un cliente sia ad un amministratore, grazie alla scelta di componenti raggruppate per categoria, mediante la logica esposta nei primi due casi d’uso.

Tuttavia, la funzione di controllo della compatibilità delle componenti non è stata ancora sviluppata: uno degli obbiettivi fondamentali di tale iterazione è dunque la definizione della “logica di controllo” della configurazione creata da un amministratore/cliente.

In quanto l’utilizzo dell’applicazione da parte di un cliente è un punto cruciale dell’elaborato, riteniamo opportuno introdurre in tale seconda iterazione i casi d’uso UC5 e UC6, i quali prevedono l’acquisto di un singolo componente da parte del cliente e la registrazione nel sistema di quest’ultimo.

Infine, dopo un’attenta analisi del diagramma delle classi di progetto ottenuto al termine della prima iterazione, si è ritenuto opportuno ridurre le responsabilità della classe SistemaPCReady: difatti tale classe con l’implementazione di nuovi casi d’uso potrebbe possedere un numero eccessivo di funzioni, e di conseguenza un massiccio carico di responsabilità tipico di una programmazione “artigianale” (problema di bassa coesione).

La soluzione proposta per risolvere il problema esposto in precedenza è un diverso “modus operandi” del pattern “Controller”: si è pensato di generare due controlli di caso d’uso (invece della precedente soluzione di tipo facadeController) per UC1 e UC2, i quali si prenderanno la responsabilità dell’esecuzione delle funzioni di assemblaggio della configurazione, rendendo la classe SistemaPCReady più snella e quindi si evita un conflitto con il pattern “High Cohesion”.

Ecco un breve elenco puntato dei contenuti previsti per tale iterazione:

* Sviluppo della logica di compatibilità delle componenti di una configurazione così come previsto in UC1 e UC2.
* Sviluppo di UC5 e UC6 in forma elaborata con contratti delle operazioni, diagramma di sequenza di sistema e diagrammi di sequenza.
* Riduzione delle responsabilità della classe SistemaPCReady mediante l’uso di controller di caso d’uso (by pattern Controller).

# Logica di compatibilità delle componenti

aaaa

# Logica di UC5

aaaaa

# Logica di UC6

aaaa

# Riduzione delle responsabilità di SistemaPCReady

aaaa