

Ingegneria del Software a.a. 2015-16
Prova Scritta del 11 Gennaio 2016

Esercizio 2

Si vuole costruire un nuovo IDE per il linguaggio Java. Tale IDE si chiamerà **EasyIDEJava**. Le funzionalità che si vogliono implementare sono le seguenti: *New file*, *Open file*, *Save*, *Save as*, *Edit*, *Indent*, *Compile*, *Refactor* e *Run*. La funzionalità *Open file* chiede all'utente, mediante un pop-up, il nome del file da aprire e in modo simile la funzionalità *Save as* chiede il nome del file da salvare e l'estensione. La funzionalità *Save as* può essere considerata "un'estensione" della funzionalità *Save*. La funzionalità *Compile* utilizza JDK che è un compilatore esterno a **EasyIDEJava**, mentre *Indent* utilizzerà un servizio disponibile su Web (chiamato WebIndenter) per indentare il codice. *Refactor* è una funzionalità complessa che può essere decomposta nei tre refactoring disponibili in **EasyIDEJava**: Replace inheritance with delegation, Replace Conditional with Polymorphism e Separate Domain from presentation. Tutti i messaggi relativi alle funzionalità *Compile* e *Run* saranno visualizzati in un'apposita finestra di **EasyIDEJava**. Gli utenti del sistema sono di due tipi: i programmatori che usano tutte le funzionalità e i revisori di codice che usano solo le funzionalità *Open file*, *Compile* e *Run*.

- a) Rappresentare i requisiti di **EasyIDEJava** mediante uno use case diagram UML
- b) Scrivere lo use case relativo alla funzionalità *Compile* prestando attenzione al fatto che la fase di compilazione può terminare anche con una lista di errori