

# **Bugtrack**

## **Model komponent**

### **Petr Nejedly**

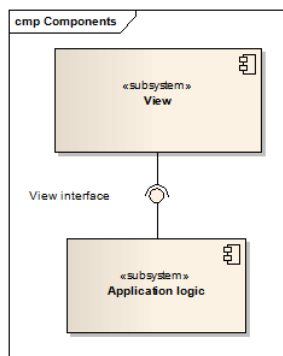
CTU in Prague, FEE  
nejedpe1@fel.cvut.cz

## Obsah

<b>1</b>	<b>Model komponent</b>	<b>3</b>
1.1	View vrstva . . . . .	3
1.2	Aplikační logika . . . . .	3

# 1 Model komponent

Systém bugtrack je naimplementován dle best-practice frameworku Seam 3. Tato skutečnost neumožňuje rozdělit systém víc než na dvě základní vrstvy. A jelikož dle standardu vrstvy spolu komunikují přímo, nikoliv přes striktně dané třídy, tak udělat rozhraní pro tuto komunikaci je náročné, sice ne nemožné, ale postrádá svůj význam. A to jasně dát přehled o rozhraní lépe řešeno komunikaci mezi vrstvami. Přesto, že komponenty jak vidíte na obrázku 1 jsou označeny typem subsystem jedná se o logické komponenty systému, který pouze celkový systém dekomponují na logičtější komponenty.



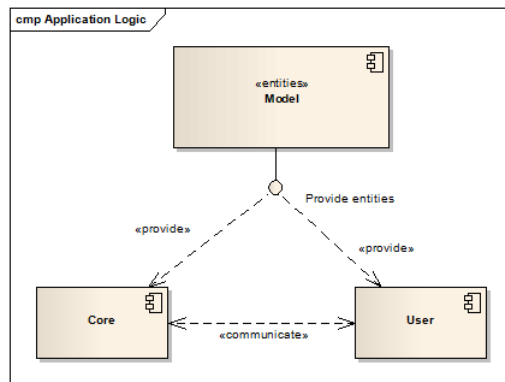
Obrázek 1: Rozdělení celého systému na komponenty

## 1.1 View vrstva

View neboli prezentační či front end vrstva je složena ze dvou částí, které nepovažuji za nutné dále rozkreslovat. A to na samotné faces stránky a k tomu přidružené konvertory, css a resource bundly.

## 1.2 Aplikační logika

Aplikační logika je rozdělena na tři části, které ovšem nelze nějak vrstvou zařadit, jako např. na persistentní vrstvu servisní apod. Nejprve si prohlédněte obrázek 2



Obrázek 2: Komponenty aplikační logiky

**Model** poskytuje základní entitní třídy pro druhé dvě části. Tyto třídy, někdy nazývané také business object tvoří základní stavební prvky systému a pracuje se s nimi. Reprezentují danou skutečnost.

**Core** neboli jádro systému obsahu, jak název napovídá, klíčové základní logiku systému. Toto jádro je dále děleno na logické části týkající se příslušného problému. **User** je komponenta starající se o uživatele. Přihlášení, registraci a celkovou bezpečnost zejména práva přístupu do vyprivilegovaných částí.

Tyto moduly spolu komunikují. Model pouze poskytuje entity.