

Spis treści

| | |
|---|---|
| 1. Założenia ogólne dla systemu..... | 1 |
| 2. Założenia ogólne dla wtyczki..... | 1 |
| 3. Założenia techniczne dla systemu..... | 1 |
| 4. Szczegółowe omówienie pakietów – funkcje | 1 |
| koziolak.rpg.core..... | 2 |
| koziolak.rpg.core.plugin..... | 2 |
| koziolak.rpg.core.tools..... | 2 |
| koziolak.rpg.core.gui..... | 2 |

1. Założenia ogólne dla systemu

- System ma działać na zasadzie wtyczek.
- Każdy system RPG powinien być reprezentowany przez jedną wtyczkę
- System jest frameworkiem. Służy zarówno do tworzenia wtyczek jak i udostępnia interfejsy do ich uruchomienia
- System powinien być modułowy
- PISZEMY W JAVIE 1.5

2. Założenia ogólne dla wtyczki

- Każda wtyczka powinna reprezentować jeden system RPG
- Wtyczka musi implementować interfejs `koziolak.rpg.core.plugin.Plugin`
- Wtyczka powinna stanowić jedno archiwum jar
- Wtyczka powinna dać się zinternacjonalizować!

3. Założenia techniczne dla systemu

- System składa się z następujących pakietów:
 - `koziolak.rpg.core` – pakiet główny zawiera klasy ładujące
 - `koziolak.rpg.core.plugin` – pakiet zawiera narzędzia do obsługi wtyczek
 - `koziolak.rpg.core.tools` – pakiet zawiera różne przydatne narzędzia.
 - `koziolak.rpg.core.gui` – pakiet zawiera podstawowy interfejs graficzny
- System powinien mieć możliwość rozszerzania i dodawania niestandardowych implementacji standardowych klas
- System powinien działać na każdym komputerze, dlatego PISZEMY W JAVIE 1.5

4. Szczegółowe omówienie pakietów – funkcje

Omówione są tu podstawowe funkcje pakietów. Jest to część życzeniowa specyfikacji.

koziolek.rpg.core

Pakiet główny. Zawiera klasę Main i cały silnik frameworku. W jego skład powinny wchodzić obiekty odpowiedzialne za uruchamianie poszczególnych modułów. Istotną rzeczą jest zapewnienie jak najmniejszego „wagomiaru” tego pakietu oraz pełna optymalizacja.

koziolek.rpg.core.plugin

Pakiet odpowiedzialny za zarządzanie i tworzenie pluginów. Powinien zawierać interfejsy pozwalające na tworzenie wtyczek. Powinien zapewniać możliwość zarządzania obecnymi wtyczkami. W jego skład wchodzi dwa osobne moduły:

- koziolek.rpg.core.plugin.framework – do twortzenia wtyczek
- koziolek.rpg.core.plugin.administration – do zarządzania nimi

Dokładne omówienie tych pakietów znajdzie się w dokumencie „Specyfikacja – część techniczna”.

koziolek.rpg.core.tools

Jest to dodatkowy pakiet zawierający różne narzędzia. Jego zadaniem jest ułatwienie tworzenia wtyczek poprzez dostarczenie rozwiązań takich jak:

- generatory losowe, kości
- podstawowy silnik testów cech zawierający:
 - testy procentowe
 - testy przeciwstwane
 - testy ogólne
- abstrakcji do zarządzania cechami, umiejętnościami, talentami itp.
- abstrakcji do zarządzania postaciami graczy i potworów
- klas pomocniczych do zarządzania modułami językowymi
- klas pomocniczych do operacji na źródłach danych
- klas pomocniczych do wyświetlania elemntów graficznych

Jak widać pakiet ten będzie najbardziej rozbudowany i zarazem powinien zapewniać jak największą elastyczność w użyciu. Jego główną cechą ma być łatwość użycia i łatwość rozbudowy.

koziolek.rpg.core.gui

Pakiet odpowiedzialny za dostarczenie graficznego interfejsu użytkownika (GUI). Pozwala wtyczkom na dostęp do głównego panelu GUI. Pakiet powinien być wmiarę nieskomplikowany ponieważ jest tylko dostarczycielem usługi. Cała logika odpowiedzialna za panele powinna być zawarta w kodzie wtyczki.