# 1. Technologia

Gra Piekarz Przemo powstała w oparciu o popularny Zend Framework 1.1.11 z wykorzystaniem biblioteki JavaScript jQuery w wersji 1.9 oraz system migracji bazy danych Doctrine.

# 2. Wymagania systemowe

## Minimalne:

- serwer WWW: Apache2 z włączonym modułem rewrite

- PHP 5.1

- serwer bazy danych: PostgreSQL 8.4

## Zalecane:

- serwer WWW: Apache2 z włączonym modułem rewrite

- PHP 5.3

- serwer bazy danych: PostgreSQL 9.0

- Narzędzie Phing

# 3. Instalacja

## 3.1. Ręczna

a) Przegrać i rozpakować plik piekarz\_przemo.tgz poleceniem:

*tar xzf piekarz\_przemo.tgz*

b) W katalogu application/configs utworzyć plik local.ini, do którego wpisać:

*db.host = "DB\_HOST"*

*db.port = 5432*

*db.username = "DB\_USERNAME"*

*db.password = "DB\_PASSWORD"*

*db.dbname = "DB\_NAME"*

podając odpowiednie parametry konfiguracyjne bazy danych

c) Z poziomu katalogu głównego gry, uruchomić polecenia:

*chmod 0777 application/cache*

*chmod 0777 application/logs*

*chmod 0777 application/configs*

*php scripts/migrate.php*

*php scripts/apply\_fixtures.php*

d) W ustawieniach VirtualHostów serwera WWW, ustawić DocumentRoot na katalog public

e) Skonfigurować narzędzie Cron tak, aby uruchamiać co minutę skrypt scripts/run\_engine.php

## 3.2. Automatyczna

a) Uruchomić skrypt instalujący poleceniem:

*sh piekarz\_przemo.sh*

b) Postępować wg instrukcji wyświetlanych przez instalator

c) W ustawieniach VirtualHostów serwera WWW, ustawić DocumentRoot na katalog public

# 4. Wiadomości e-mail

Domyślnie wysyłanie maili po ukończonej misji jest wyłączone. Aby włączyć tę opcję, należy wyedytować plik local.ini i ustawić:

*mail.enabled = 1*

W takiej konfiguracji wiadomości e-mail wysyłane są za pomocą funkcji "mail", wykorzystującej serwer "sendmail" znajdujący się na serwerze. Jeśli brakuje takiej usługi, można podać adres, login i hasło do skrzynki pocztowej, z której będą wysyłane maile:

*mail.smtp = "smtp.gmail.com"*

*mail.config.username = "example@gmail.com"*

*mail.config.password = "example"*

*mail.config.port = 465*

# 5. Pierwsze logowanie

**Pierwszy użytkownik, który zaloguje się do aplikacji będzie administratorem gry.**

# 6. Zawartość

## 6.1. Struktura katalogów

Struktura katalogów podzielona jest w następujący sposób:

- application - główny katalog aplikacji

- library - katalog zawierający biblioteki wykorzystywane w aplikacji

- public - katalog, którego głównym elementem jest plik index.php, będący wejściem do aplikacji przez użytkownika - przez niego przechodzą wszystkie akcje

- scripts - zawiera pliki i skrypty potrzebne do obsługi gry.

## 6.2. Application

W katalogu application znajduje się kod gry. Struktura tego katalogu jest zgodna z założeniami wzorca MVC (and. Model-View-Controller), który zakłada podział aplikacji na trzy główne części:

- model - jest reprezentacją problemu lub logiki aplikacji (katalog models)

- widok - opisuje, jak wyświetlić pewną część modelu w ramach interfejsu użytkownika.

Może składać się z podwidoków odpowiedzialnych za mniejsze części interfejsu (katalog views)

- kontroler - przyjmuje dane wejściowe od użytkownika i reaguje na jego poczynania, zarządzając aktualizacje modelu oraz odświeżenie widoków (katalog controllers)

Ponadto znajdują się także katalogi:

- cache zawierający pliki tymczasowe, służące do zapisywania danych, które rzadko się zmieniają

- resources, który zawiera pliki językowe i tłumaczenia.

Plik Bootstrap.php zawiera logikę pozwalającą na załadowanie wszystkich potrzebnych zasobów aplikacji. Odpowiada także za zainicjowanie połączenia z bazą danych oraz wczytanie danych o aktualnie zalogowanym użytkowniku.

## 6.3. Library

Katalog library zawiera gotowe biblioteki zewnętrzne, wykorzystywane w aplikacji, z których podstawową jest biblioteka Zend. Oprócz tego wyróżnić można biblioteki:

- Doctrine, która zawiera mechanizm do migracji bazy danych do najnowsze wersji

- ZendX, pozwaląca na integrację JavaScript'owej biblioteki jQuery z Zend'em

- GN, zawierający różne mechanizmy i funkcjonalności wykorzystywane w innych projektach.

## 6.4. Public

Zawiera pliki graficzne oraz arkusze stylów wykorzystywane w aplikacji.

## 6.5. Scripts

Przydatne pliki skryptowe, służące do obsługi aplikacji, np. aktualizacji bazy danych do najnowszej wersji.