

< Dokumentacja / >

< O projekcie / >

Naszym głównym założeniem w projekcie było utworzenie wygodnej i intuicyjnej aplikacji, przypominającą znane nam już dotychczas dzienniki elektroniczne, stosowane w szkołach (np. MobiDziennik) oraz uniwersytetach (np. USOS).

< Zastosowany model projektu / >

Zastosowany przez nas model projektowy - **RUP (Rational Unified Process)** jest metodyką zakładającą iteracyjny proces tworzenia oprogramowania. Metodyka ta wprowadza reguły oraz dobre praktyki dla wszystkich faz wytwarzania oprogramowania.

W każdej iteracji wytwarzany jest fragment systemu, który jest udostępniany klientowi. Pozwala to na uzyskanie szybkiej informacji zwrotnej i upewnieniu się, że zespół realizujący projekt dobrze zrozumiał wymagania i oczekiwania klienta. Szybkie wykrycie ewentualnych problemów lub nieporozumień, pozwala szybko wprowadzić odpowiednie modyfikacje.

< Zastosowane technologie / >

Aplikacja desktopowa

Java

To najpopularniejszy język programowania i najczęściej wybierana platforma programistyczna. Obniża koszty, skraca czas programowania, pobudza innowacyjność i usprawnia usługi aplikacji.

[Więcej](#)

JavaFX

To platforma aplikacji klienckich nowej generacji typu open source dla komputerów stacjonarnych, mobilnych i systemów wbudowanych, zbudowana w języku Java.

[Więcej](#)

Maven

Następca już prawie wymarłego ANT'a. Maven potrafi zrobić wszystko z naszym projektem podczas jego budowania. Jedyne co musimy zrobić to zastosować się do konwencji nazewnictwa katalogów w projekcie jaką wymaga Maven. Jeśli to zrobimy – sam Maven zatroszczy się o kompilowanie kodu, wykonywanie testów itd.

[Więcej](#)

Hibernate

Framework stanowiący jedną z najpopularniejszych implementacji Java Persistence API (JPA). Umożliwia on dokonanie mapowania obiektowo-relacyjnego (ORM) – czyli odzworowania obiektów klas na bazę danych. Najprościej mówiąc odpowiada za wygodną komunikację aplikacji z bazą danych.

[Więcej](#)

Spring

Narzędzie, które znacznie skraca proces tworzenia aplikacji. Stanowi szereg usprawnień, których implementacja w Javie zajęłaby wieki.

[Więcej](#)

JUnit

Narzędzie służące do tworzenia powtarzalnych testów jednostkowych oprogramowania pisanej w języku Java.

[Więcej](#)

Log4j2

Biblioteka języka programowania Java służąca do tworzenia logów podczas działania aplikacji.

[Więcej](#)

Strona internetowa

HTML

Język znaczników stosowany do tworzenia dokumentów hipertekstowych.

[Więcej](#)

CSS

Kaskadowe arkusze stylów to język, który wnosi styl do stron internetowych poprzez interakcję z elementami HTML.

[Więcej](#)

JavaScript

Język programowania, który umożliwia wdrożenie na stronie internetowej skoplikowanych elementów, dzięki którym strona ta może nie tylko wyświetlać statyczne informacje, ale również obsługiwać zmianę treści odpowiednio do sytuacji, wyświetlać interaktywne mapy i animacje grafiki 2D/3D, wyświetlać wideo itd.

[Więcej](#)

React

JavaScriptowa biblioteka służąca do tworzenia interfejsów użytkownika.

[Więcej](#)

Bazy danych

T-SQL

Rozszerzenie języka SQL. T-SQL jest językiem proceduralnym. Możemy za jego pomocą stosować typowe elementy programowania jak pętle, instrukcje warunkowe, możemy także tworzyć własne funkcje, procedury i operować na zmiennych.

[Więcej](#)

MSSQL

System zarządzania bazą danych.

[Więcej](#)

Inne

Git

Jest rozproszonym systemem kontroli wersji ułatwiającym pracę nad tworzeniem aplikacji, stron internetowych i innych narzędzi w grupach. System śledzi wszystkie wykonywane zmiany w plikach, a także umożliwia przywrócenie ich dowolnej, wcześniejszej wersji.

[Więcej](#)

GitHub

To usługa hostingu umożliwiająca zarządzanie repozytoriami Git. Przy jego pomocy jesteśmy w stanie udostępnić swój kod w jednym miejscu dla wszystkich.

[Więcej](#)

Taiga

To darmowy system zarządzania projektami typu open source dla startupów, programistów i projektantów.

[Więcej](#)

< Twórcy projektu / >

Michał Misiałowicz – Project Manager, Implementator (java), Integrator



Michał Nowak – Process Engineer, Implementator (java)



Krzysztof Raczyński – Software Architect/Database Designer



Mirosław Konarski – Software Architect/Database Designer



Paweł Rostecki – System Administrator



Jędrzej Wesołowski – Implementator (web)



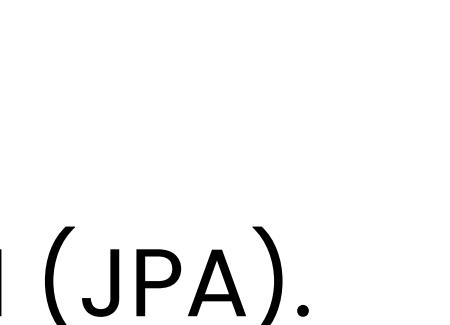
Szymon Mazurek – Implementator (web)



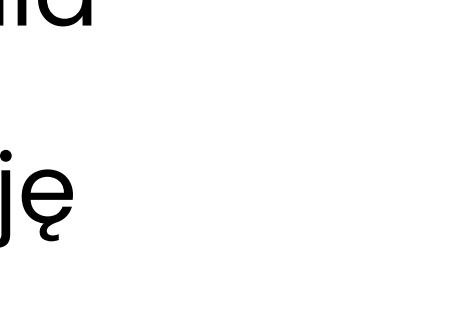
Mateusz Cwojdziński – Implementator (web)



Jędrzej Rybczyński – Implementator, System Administrator (support)



Dominika Leśniewska – System Analyst, UI Designer



Maciej Maciejewski – System Analyst



Maksymilian Łąsicki – System Tester (manual testing), Course Developer



Mikołaj Mumot – Test Manager, System Tester (automatic testing)

