

Wykaz wybranych technologii:

Języki:

1. Java SE
2. Java ME
3. SQL
4. ~~Wszystkie języki front-endowe~~ (na razie się nimi nie przejmujemy).

Technologie:

- 1 **REST api**
 - a. <https://devszczepaniak.pl/wstep-do-rest-api/>
 - b. <https://www.devmobile.pl/rest-api-od-podstaw/>
- 2 **JSON**
- 3 * **JavaPersistenceApi** (ORM)
 - a. <https://kobietydokodu.pl/13-baza-danych-z-jpa-cz-1/>
- 4 * **JEE**
- 5 * **Spring Framework / SpringBoot**
 - a. <https://javastart.pl/baza-wiedzy/frameworki/spring>

* - Możliwe, że uda się uniknąć dużych businessowych frameworków w projekcie. Jeśli nie, warto wcześniej coś tam o nich poczytać.

Narzędzia:

1. VCS - **Git**
2. IDE - **Intelij IDEA**. Można zainstalować płatną wersję za free, na konto studenckie – sprawdzone info.
3. IDE - **Android Studio**
4. **Apache Maven**
 - a. <https://www.kodolamacz.pl/blog/wprowadzenie-do-apache-maven-czyli-jak-tworzy-sie-projekty-w-swiecie-java/>
 - b. <http://namiekko.pl/2016/03/26/na-co-komu-maven/>
5. **Postman** – programik do testowania wywołań REST.

Wiedza ogólna:

- Podstawowy podstaw z programowania obiektowego, protokołu http, i relacyjnych baz danych.

Na razie tylko tak pobieżnie. Nie podaję jeszcze wersji i konkretnych implementacji poszczególnych komponentów, bo sam musimy się upewnić które będą dla nas najwygodniejsze. Bardziej niż szczegóły i dokumentacja interesuje nas wiedza ogólna. Świadomość czym są i do czego służą dane komponenty, oraz w jaki sposób usprawniają wytwarzanie oprogramowania i dlatego warto ich używać. To nam absolutnie wystarczy w tym momencie. Tam, gdzie nie podałem linków artykuły na wiki powinny całkowicie wystarczyć.

Jakby coś było niejasne, lub mielibyście jakieś pytania, piszcie do mnie śmiało.

Marcin