**Technologie które chciałbym się nauczyć:**

* **TypeScript,**
* **Redux,**
* **React Native,**
* **Next.js,**
* **Gatsby,**
* **Docker,**
* **Angular.js,**
* **Vue.js**

**PL**

**HTML**

Znajomość opisowego języka **<strong>**HyperText Markup Language**</strong>** umożliwiająca pisanie semantycznego kodu, dobrze interpretowanego przez boty indeksujące. Świadomość projektowania struktury projektu dla urządzeń desktopowych oraz mobilnych. Umiejętność poprawnego zagnieżdżania elementów, a także świadomość o zależnościach pomiędzy nimi. Stałe podnoszenie kompetencji w standardzie **<strong>**HTML 5**</strong>.**

**CSS**

Posługiwanie się **<strong>**Cascading Style Sheets**</strong>** w stopniu umożliwiającym projektowanie stron internetowych na urządzenia desktopowe oraz mobilne z zachowaniem zasad **<strong>**RWD**</strong>** w konwencji **<strong>**Dekstop first**</strong>** oraz **<strong>**Mobile first**</strong>**. Znajomość właściwości **<strong>**transform**</strong>** i **<strong>**transition**</strong>** do podstawowych animacji oraz dyrektywy **<strong>**keyframes**</strong>** do zaawansowanych animacji. Umiejętność pozycjonowania za pomocą właściwości **<strong>**position**</strong>** oraz projektowania struktury za pomocą **<strong>**flexboxa**</strong>**. Umiejętność wykorzystania kombinatorów CSS oraz znajomość pojęcia specyficzności selektorów. Świadomość funkcjonowania stylów domyślnych w przeglądarkach. Umiejętność oddzielenia warstwy strukturalnej od warstwy wizualnej. Znajomość pseudoklas i pseudoelementów. Znajomość oraz wykorzystywanie konwencji **<strong>**BEM**</strong>**. Ciągłe podnoszenie kompetencji w zakresie coraz nowszych właściwości.

**JavaScript**

Znajomość typów prostych oraz złożonych. Sprawne wykorzystywanie podstawowych elementów programowania tj. pętle, instrukcje warunkowe, tablice. Wykorzystywanie metod iteracyjnych **<strong>**forEach**</strong>**, **<strong>**map**</strong>**, **<strong>**filter**</strong>**. Podstawowa znajomość programowania obiektowego. Świadomość istnienia **<strong>**hoistingu**</strong>** zmiennych oraz ich zakresu (scope). Umiejętność programowania synchronicznego i asynchronicznego. Znajomość obietnic wykorzystywanych wraz z metodą **<strong>**fetch**</strong>** lub modułem **<strong>**axios**</strong>**. Znajomość pojęcia domknięć. Wiedza o słowie kluczowym **<strong>**this**</strong>** oraz jego wiązaniu, przy tym świadome wykorzystywanie funkcji tradycyjnej oraz strzałkowej. Znajomość JS w standardzie ES6+. Sprawna manipulacja **<strong>**Document Object Model**</strong>**. Znajomość zdarzeń i nasłuchiwaczy. Umiejętność parsowania plików **<strong>**JSON**</strong>** oraz pracy z obiektami. Sprawne rzutowanie typów. Dynamiczna kontrola stylów CSS za pomocą JavaScript. Stałe zgłębianie tego języka.

**React.js**

Umiejętność konstruowania komponentów w podejściu klasowym oraz funkcyjnym. Świadomość korzystania z cyklów życia komponentów. Znajomość pojęć **<strong>**state**</strong>** oraz **<strong>**props**</strong>**. Znajomość środowiska **<strong>**Create React App (CRA)**</strong>**. Umiejętność programowania **<strong>**Single Page Application (SPA)**</strong>**. Sprawne wykorzystywanie w projektach **<strong>**React Router**</strong>** oraz **<strong>**React Context**</strong>**. Umiejetność projektowania aplikacji w standardzie React 18+ za pomocą **<strong>**hooków**</strong>**. Wykorzystanie dodatkowych bibliotek React m.in. **<strong>**React Spring**</strong>**, **<strong>**React Scroll**</strong>**. Stałe podnoszenie kompetencji w zakresie biblioteki React.js.

**SASS/SCSS**

Wykorzystanie preprocesora SASS w składni **<strong>**SCSS**</strong>** (preferowane) oraz **<strong>**SASS**</strong>**. Znajomość klas abstrakcyjnych. Umiejętność pisania **<strong>**mixinów**</strong>**. Sprawne wykorzystywanie zmiennych oraz funkcji. Świadomość istnienia operatorów matematycznych oraz umiejętność ich wykorzystywania. Sprawne używanie zagnieżdżeń. Umiejętność konfiguracji preprocesora w zakresie minimalizacji kodu oraz wsparcia właściwości dla starszych przeglądarek, a także zignorowania kompilacji niepożądanych plików. Umiejętność podziału styli projektu na mniejsze częsći w celu poprawienia czytelności.

**Node.js**

Umiejętność zaprogramowania serwera. Kontrola danych za pomocą metod **<strong>**HTTP**</strong>**. Umiejętność zaprojektowania aplikacji **<strong>**CRUD**</strong>**. Preferowany stos technologiczny **<strong>**MERN**</strong>**. Współpraca z bazą danych za pomocą biblioteki **<strong>**Express.js**</strong>**. Współpraca z zewnętrznym **<strong>**API**</strong>**. Wystawianie własnych **<strong>**endpointów**</strong>**. Umiejętność wydzielania składowych projektu według wzorca **<strong>**Model-View-Controller (MVC)**</strong>**. Korzystanie z narzędzia **<strong>**Nodemon**</strong>** do automatycznego przeładowywania serwera w celach developerskich. Umiejętność zarządzania pakietami **<strong>**NPM**</strong>** w wartswie frontendu oraz backendu.

**MongoDB**

Umiejętność ustawienia połączenia z nierelacyjną bazą danych MongoDB w podejściu **<strong>**klient - serwer**</strong>** oraz rozproszonym systemie **<strong>**chmury**</strong>**. Manipulacja i przeszukiwanie bazy danych za pomocą **<strong>**Node.js**</strong>**. Wykorzystanie biblioteki **<strong>**Mongoose**</strong>** w celu modelowania danych oraz kontrolowanie ich za pomocą biblioteki **<strong>**Express.js**</strong>**. Umiejętność wykorzystania operacji Create, Read, Update, Delete. Podstawowa znajomość operatorów.

**Git**

Umiejętność utworzenia repozytorium na **<strong>**GitHub**</strong>** oraz połączenie go za pomocą systemu kontroli wersji - **<strong>**Git**</strong>** poprzez konsole **<strong>**Git Bash**</strong>** oraz wbudowanej konsoli **<strong>**Visual Studio Code**</strong>**. Współpraca z wieloma osobami na jednym repozytorium. Znajomość pojęcia **<strong>**commit**</strong>** do zatwierdzania i komentowania zmian. Umiejętność wykorzystania **<strong>**gałęzi**</strong>** w celu pracy nad niezależnymi od siebie komponentami w projekcie, a także łączenia ich za pomocą **<strong>**merge**</strong>**.

**Bootstrap**

Umiejętność wykorzystania komponentów Bootstrapa. Szybkie odszukiwanie potrzebnych informacji w dokumentacji. Znajomość **<strong>**siatki**</strong>** oraz **<strong>**breakpointów**</strong>** Bootstrapa. Umiejętność konfiguracji w środowisku **<strong>**Create React App**</strong>**. Znajomość zdefiniowanych klas oraz umiejętność użycia nieznanych klas za pomocą dokumentacji.

**MySQL**

Znajomość relacyjnych baz danych oraz wykorzystanie języka MySQL do ich obsługi. Znajomość pojęcia relacji a także ich podziału. Umiejętność przeszukiwania, segregowania, grupowania, łączenia, filtrowania danych w bazie danych. Znajomość pojęcia hurtowni danych. Wiedza z zakresu systemów baz danych oraz modelowania danych w podejściu relacyjnym.

**PHP**

Umiejętność programowania w PHP z zakresu materiału z technikum. Umiejętność implementacji systemu logowania i rejestracji oraz podstawowego zabezpieczania danych. Znajomość funkcjonowania topologii **<strong>**klient - serwer**</strong>**. Umiejętność obsługi bazy danych w bazodanowym silniku **<strong>**MySQL**</strong>**. Znajomość pojęcia **<strong>**sesji**</strong>** oraz ich wykorzystania. Umiejętność wykorzystania języka PHP wraz z JavaScript’ową biblioteką **<strong>**React.js**</strong>** w środowisku **<strong>**Create React App**</strong>**.

**C/C++**

Znajomość konstrukcji jezyka C/C++, pojęcia algorytmu, typów danych i zmiennych, a także umiejętne wykonywanie na nich opearacji. Znajome podejście iteracyjne oraz rekurencyjne. Umiejętność implementacji struktur danych. Umiejetność programowania statycznego, proceduralnego, strukturalnego oraz obiektowego. Znajomość i obsługa biblioteki **<strong>**Allegro**</strong>** do implementacji graficznego interfejsu użytkownika.

**Java**

Umiejętność budowania obiektowych aplikacji konsolowych w języku Java. Umiejętność budowania obiektowych aplikacji graficznych za pomocą bibliotek **<strong>**AWT**</strong>** i **<strong>**Swing**</strong>**. Znajomość klas abstrakcyjnych. Wiedza na temat dziedziczenia, polimorfizmu, hermetyzacji i abstrakcji. Znajomość różnic występujących między dziedziczeniem a kompozycją. Umiejętność wykorzystania podstawowych konstrukcji języka programowania tj. Instrukcje warunkowe, pętle, tablice, zmienne. Umiejętność pisania i wykorzystywania interfejsów. Znajomość kolekcji: **<strong>**List**</strong>**, **<strong>**ArrayList**</strong>**, **<strong>**Set**</strong>**, **<strong>**Map**</strong>**, **<strong>**HashMap**</strong>**.

**ENG**

**HTML**

Knowledge about **<strong>**HyperText Markup Language**</strong>** to write semantic code well interpreted by indexing browser bots. Awareness of designing structures for desktop and mobile devices. Ability to nest elements correctly and knowledge about depdendencies between them. Constant improvement of competences in **<strong>**HTML 5 standard**</strong>**.

**CSS**

Using **<strong>**Cascading Style Sheets**</strong>** to design websites for mobile and desktop devices which includes **<strong>**Responsive Web Design**</strong>** rules in **<strong>**Desktop first**</strong>** or **<strong>**Mobile first**</strong>** convention. Knowledge abaout **<strong>**transform**</strong>** and **<strong>**transition**</strong>** properties for basic animations and **<strong>**keyframes**</strong>** directive for advanced animations. Knowledge about positioning with **<strong>**position**</strong>** property and design structure with **<strong>**flexbox**</strong>**. Ability to use CSS combiners and knowledge about selectors specificity. Awareness of functionality default styles in browsers. Ability to separate the visual layer from the structural layes. Knowledge of pseudo-clasess and pseudo-elements. Knowledge and use of **<strong>**BEM**</strong>** conventions. Continuous improvement of competences in the newest CSS properties.

**JavaScript**

Knowledge about primitives types and complex types. Efficient use of basic programming elements like loops, conditional statements, arrays etc. Using iterative methods like **<strong>**forEach**</strong>**, **<strong>**map**</strong>**, **<strong>**filter**</strong>**. Basic knowledge of Object Oriented Programming. Awareness of the existence variables **<strong>**hoisting**</strong>** and their scope. Synchronous and asynchronous programming skill. Knowledge about promises used with **<strong>**fetch**</strong>** method or **<strong>**axios**</strong>** module. Knowledge about closures. Knowledge of **<strong>**this**</strong>** keyword and its binding, which includes using traditional functions and arrow functions. Knowledge of JS in the ES6+ standard. Efficient **<strong>**Document Object Model**</strong>** manipulation. Knowledge of events and listeners. Ability to parse **<strong>**JSON**</strong>** files and working with objects. Efficient JavaScript types conversions. Dynamical control of the CSS styles with JS. Learning, learning and once again learning of this language.

**React.js**

Constructing components with classes and functions. Awareness of using components life cycles. Knowledge the concepts of **<strong>**state**</strong>** and **<strong>**props**</strong>** elements. Knowledge of the **<strong>**Create React APP (CRA)**</strong>** environment. **<strong>**Single Page Application (SPA)**</strong>** programming skill. Efficient using **<strong>**React Router**</strong>** and **<strong>**React Context**</strong>** in the projects. Knowledge of designing applications with React 18+ standard with **<strong>**hooks**</strong>**. Using additional React libraries like **<strong>**React Spring**</strong>**, **<strong>**React Scroll**</strong>**. Continuing learning React.js.

**SASS/SCSS**

Using the SASS preprocessor in the **<strong>**SCSS**</strong>** (preffered) and **<strong>**SASS**</strong>** syntax. Knowledge about abstract classes. Ability to write and use **<strong>**mixins**</strong>**. Efficient using variables and functions. Awareness of the existence mathematical operators and the ability to use them. Efficient using nesting. Ability to configure preprocessor to minimize code and support properties for older browsers, and ignore compilation of unwanted files. Ability to separate styles into smaller parts to improve readibility.

**Node.js**

Server programming with Node.js. Data control using **<strong>**HTTP**</strong>** methods. Ability co design **<strong>**CRUD**</strong>** applications. Preffered **<strong>**MERN**</strong>** technology stack. Working with the database with **<strong>**Express.js**</strong>**. Cooperation with an external **<strong>**API**</strong>**. Posting own **<strong>**endpoints**</strong>**. Separate projects components in **<strong>**Model-View-Controller (MVC)**</strong>** architectural pattern. Using **<strong>**Nodemon**</strong>** tool to automatical server reload for development puproses. Ability to manage **<strong>**NPM**</strong>** packages in the frontend and backend.

**MongoDB**

Knowledge about setting connection with no-relational database - MongoDB in **<strong>**client - server**</strong>** architecture and distributed **<strong>**cloud**</strong>** system. Manipulation and searching data with **<strong>**Node.js**</strong>**. Using **<strong>**Mongoose**</strong>** library for data modeling and controlling them with **<strong>**Express.js**</strong>** library. Handling Create, Read, Update, Delete Operations. Basic knowledge about MongoDB operators.

**Git**

Ability to create repository on **<strong>**GitHub**</strong>** and connect to it with version control system - **<strong>**Git**</strong>** by console **<strong>**Git Bash**</strong>** or built in **<strong>**Visual Studio Code**</strong>** console. Cooperations with people on single repository. Knowledge about **<strong>**commits**</strong>** for accepting and commenting changes i projects. Knowledge of using **<strong>**branches**</strong>** for working on undependent components in projects and bind them with the **<strong>**merge**</strong>** instruction.

**Bootstrap**

Knowledge of using Bootstrap components. Fast searching necessary informations from documentation. Knowledge of Bootstrap **<strong>**grid**</strong>** and **<strong>**breakpoints**</strong>**. Knowledge about configuration Bootstrap in **<strong>**Create React App**</strong>** environment. Knowledge of defined classes and the ability to use unknown classess through documentation.

**MySQL**

Knowledge of realational databases and using MySQL language for handling that data. Knowledge of the concept of relations, as well their divisions. Ability to search, sort, group, merge and filter data. Knowledge of the concept of a data warehouse. Knowledge of database systems and modeling in the relational approach.

**PHP**

The ability to programming in PHP in the field of technical school. Ability to implement a login and registration system and basic data securing. Knowledge of the functioning of the **<strong>**client - server**</strong>** topology. Ability to handling database in the **<strong>**MySQL**</strong>** database engine. Knowledge of the concept of **<strong>**session**</strong>** and their using. Ability to use PHP language with JavaScript framework - **<strong>**React.js**</strong>** in **<strong>**Create React App environment**</strong>**.

**C/C++**

Knowledge of the structure C/C++ language. Knowing a concept of algorithm, data types, variables, and skillful execution of operation on them. Knowledge of iteration and recursion. Knowing how to implement data structures. Knowledge of static, procedural, structural and object oriented programming. Knowledge and handling **<strong>**Allegro**</strong>** library for implementing graphical user interface.

**Java**

Ability to build object-oriented console applications in Java. Ability to build object-oriented graphic applications using the **<strong>**AWT**</strong>** and **<strong>**Swing**</strong>** libraries. Knowledge of abstract classes. Knowledge of inheritance, polymorphism, encapsulation and abstraction. Knowing the differences between inheritance and composition. Ability to use the basic structures of the programming language like conditional statements, loops, arrays, variables. Ability to write and use interfaces. Knowledge of the collections build in Java like **<strong>**List**</strong>**, **<strong>**ArrayList**</strong>**, **<strong>**Set**</strong>**, **<strong>**Map**</strong>**, **<strong>**HashMap**</strong>**.