

Kod szkolenia: **TOOLS/FD**

Tytuł szkolenia: **Warsztat narzędziowy usprawniający pracę Front-end Developera**

Dni: **3**

## Opis:

### Adresaci szkolenia

Kurs przeznaczony jest dla programistów pracujących z technologiami webowymi (HTML, CSS, JavaScript) na każdym poziomie zaawansowania, którzy chcą poszerzyć swój warsztat narzędziowy o rozwiązania umożliwiające wydajniejszą pracę, eliminację wielu rodzajów błędów i szybsze osiągnięcie rezultatów w codziennej pracy.

### Cel szkolenia

Celem szkolenia jest zapoznanie uczestników z zestawem najnowszych narzędzi pozwalających na efektywne programowanie w języku JavaScript i pracę z HTML / CSS a także zdobycie wiedzy na temat dobrych praktyk związanych z pracą w tych technologiach, sposobów rozwiązywania typowych problemów i umiejętności sprawnego wykorzystywania tej wiedzy we własnych projektach.

### Mocne strony szkolenia

Charakterystyczną cechą szkolenia jest szeroki zakres przedstawionego materiału przy jednoczesnym skupieniu na aspekcie praktycznym. Największy nacisk położony jest na techniki i narzędzia, które sprawdzają się w rzeczywistych projektach i przynoszą największy wzrost wydajności pracy. Omawiane są alternatywne rozwiązania poszczególnych problemów, tak aby uczestnik mógł również sam wybrać praktyki, które najbardziej odpowiadają charakterowi jego pracy. Część warsztatowa szkolenia pozwala na osiągnięcie poziomu umiejętności pozwalającego na samodzielne wykorzystanie zdobytej wiedzy od razu po zakończeniu kursu.

### Wymagania

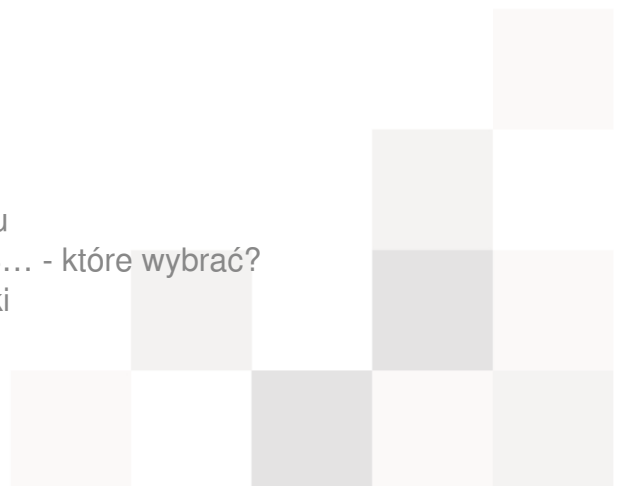
Od uczestników wymagane jest rozeznanie w podstawach programowania w JavaScript oraz podstawowa wiedza dotycząca języka HTML i CSS.

### Parametry szkolenia

*3\*8 godzin (3x7 godzin netto) wykładów i warsztatów. Wielkość grupy: maks. 10 osób.*

## Program szkolenia:

1. Wprowadzenie
  - Rozwój technologii webowych i związane z tym wyzwania
2. IDE vs edytor "na sterydach"
  - Początkowa konfiguracja
  - Definiowanie szablonów plików / code snippets
  - Sposoby na szybszą pracę z edytorem
  - Narzędzia wspomagające egzekwowanie jednolitego stylu kodu
3. Online code playgrounds
  - Porównanie popularnych rozwiązań
  - Praktyczne zastosowania
4. System kontroli wersji
5. Przeglądarka: podstawowe narzędzie programisty
  - Kodowanie w przeglądarce, zapisywanie kodu
  - LiveReload / BrowserSync
  - Emulacja urządzeń i różnych warunków sieciowych
  - Debugowanie
    - JS, CSS i HTML
    - komunikacja z serwerem
    - zasoby przeglądarki
  - Badanie i optymalizacja wydajności aplikacji
  - Rozszerzanie DevTools
6. Praca z CSS
  - Preprocesory Sass / Less / Stylus
  - Postprocesory
  - Najprzydatniejsze rozszerzenia
  - Sourcemaps
  - Frameworki CSS: zalety i wady
    - Bootstrap, Foundation i inne
    - możliwości modyfikacji
7. Praca z grafiką
  - Automatyczna optymalizacja obrazków
  - Prawdziwie responsywne grafiki
  - CSS sprites / icon fonts / SVG sprites / data uri
8. Praca z HTML5
  - Walidacja dokumentu
  - Feature Support & Polyfills
  - Emmet - HTML i CSS w skrócie
  - Jade - HTML czytelniej i szybciej
9. Praca z JavaScript
  - Wprowadzenie do Node.js
    - npm i Bower
  - Narzędzia do statycznej analizy kodu
    - JSLint, JSHint, ESLint, JSCS... - które wybrać?
    - Analiza jakości kodu - metryki



- Moduły - zestawienie różnych rozwiązań
- Module bundlers
- ES6 / ECMAScript 2015
  - Wprowadzenie i najciekawsze struktury
  - Transpilery - tłumaczenie ES6 dla aktualnych przeglądarek
- Testowanie
  - Środowisko uruchamiania testów (Karma)
  - Frameworki testujące (Jasmine / Mocha)
  - Biblioteki asercji
  - Object mocking
  - Testowanie asynchronicznego kodu
  - Generowanie raportów

## 10. Automatyzacja procesów

- Task runners: Grunt i gulp.js
- Rozszerzenia uniwersalne i dedykowane frameworkom
- Yeoman - generowanie struktur projektów

