

# RESPONSYWNE INTERFEJSY

Klasy urządzeń klienckich  
Responsywne style CSS

# Klasy urządzeń klienckich

- Desktopy z zewnętrznymi monitorami
- Laptopy
- Tablety
- Smartphony

# Klasy urządzeń klienckich

- Desktopy z zewnętrznymi monitorami
  - ▣ Panoramiczne, najczęściej format 16:9 lub 16:10
  - ▣ Powyżej 19 cali
  - ▣ Rozdzielczości: 1440x900, 1680x1050, 1920x1080, 2560x1440, 4K, 5K
  - ▣ Mysz i klawiatura
- Laptopy
  - ▣ Panoramiczne, najczęściej format 16:9 lub 16:10
  - ▣ Powyżej 12 cali
  - ▣ Rozdzielczości: 1366x768, 1280x800 i wyższe
  - ▣ Mysz, klawiatura, ekran dotykowy

# Klasy urządzeń klienckich

## □ Tablety

- Orientacja pozioma lub pionowa, formaty 4:3, 16:9, 16:10 i inne
- Od 7 do 12 cali
- Rozdzielczości: 800x600, 1366x768, 1920x1080 i wyższe
- Ekran dotykowy

## □ Smartphony

- Najczęściej używane w orientacji pionowej
- Od 3,5 do 6 cali
- Rozdzielczości: 480x800, 800x1280, 1080x1920 i wyższe
- Ekran dotykowy

# Desktop vs smartphone

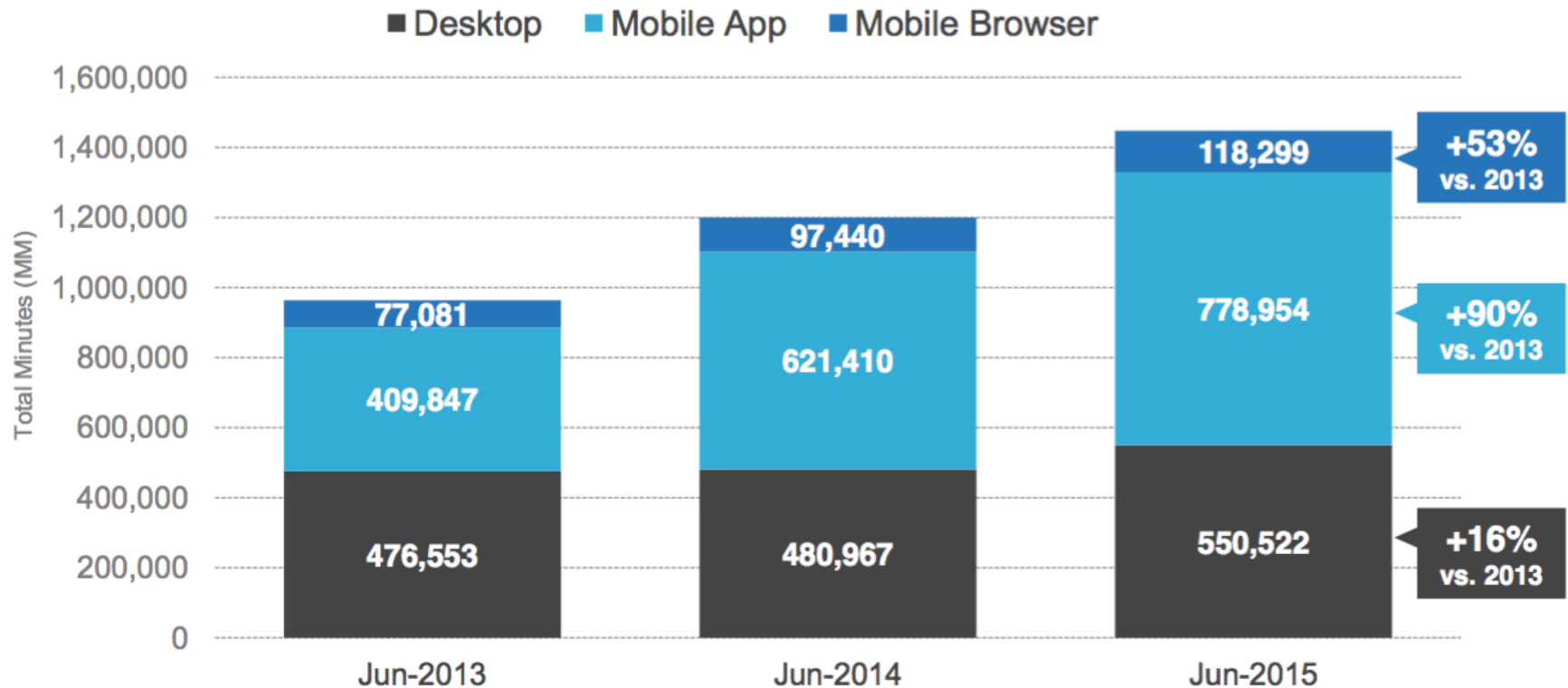
Desktop	Smartphone
Orientacja pozioma	Orientacja pionowa
19+ cali	ok. 5 cali
Mysz i klawiatura	Ekran dotykowy
Duża moc obliczeniowa	Procesory mobilne
Szybkie, stałe łącze	Internet mobilny

# Mobile vs desktop

6

## Growth in Digital Media Time Spent

Source: comScore Media Metrix Multi-Platform & Mobile Metrix, U.S., Total Audience

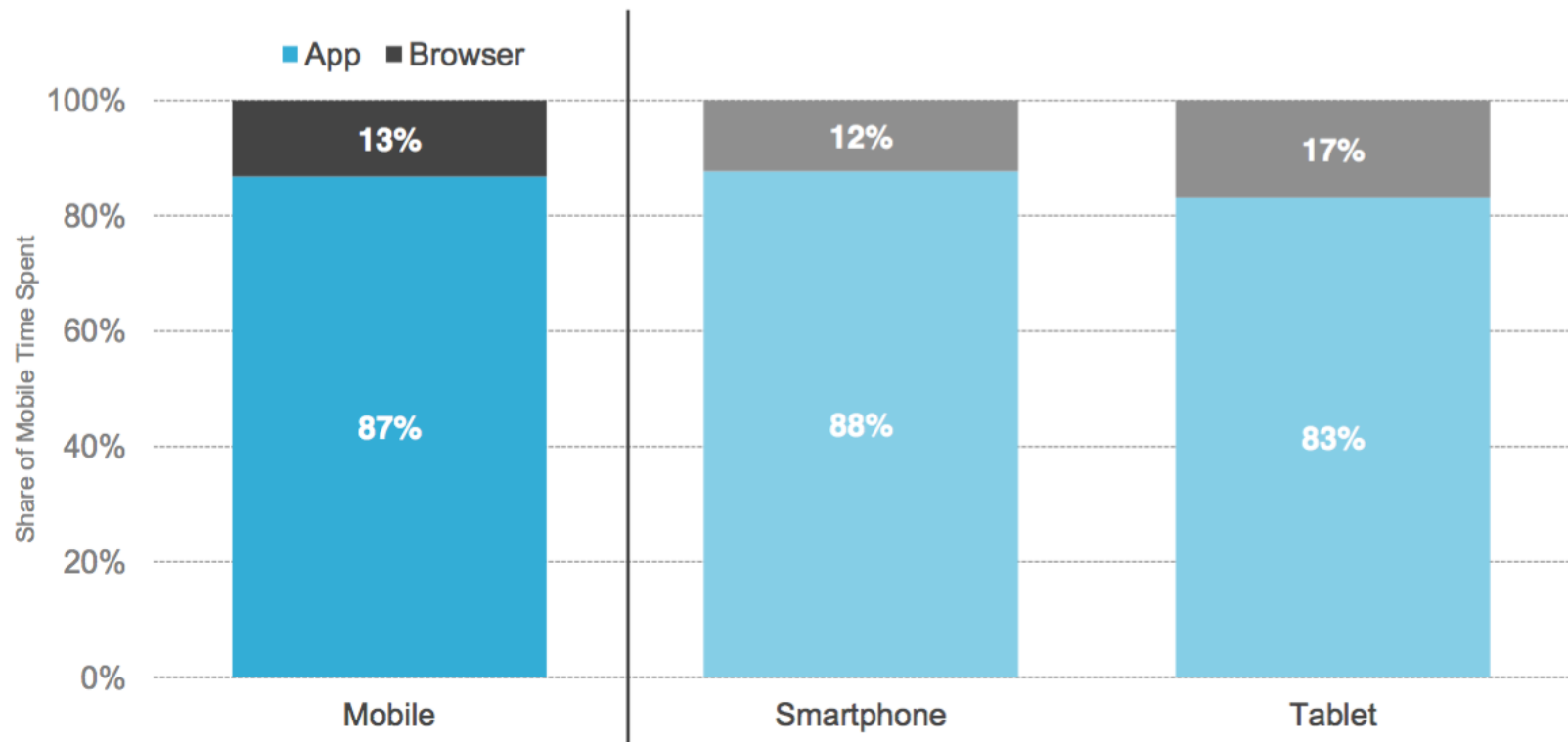


# Mobilny Internet vs aplikacje mobilne

7

## Share of Time Spent on Mobile: App vs. Browser

Source: comScore Mobile Metrix, U.S., Age 18+, June 2015



# Co jest nie tak z mobilnym Internetem?

- Jest do kitu!
  - ▣ <http://www.theverge.com/2015/7/20/9002721/the-mobile-web-sucks>
- Średni czas ładowania stron w mobilnym Internecie:  
8 sekund
  - ▣ Dla porównania na desktopie:
    - Cel dla witryn typu ecommerce (np. sklepy internetowe): 2 s
    - Cel jaki stawia sobie Google: 0,5 s
  - ▣ Z każdą sekundą coraz więcej użytkowników rezygnuje
  - ▣ Użytkownicy mobilni są na szczęście bardziej cierpliwi
    - 6-10 sekund oczekiwania przed porzuceniem strony



# Co jest nie tak z mobilnym Internetem?

- Wiele stron ciągle nie jest dostosowanych do urządzeń mobilnych
  - ▣ ...albo nie jest dostosowanych wystarczająco dobrze
  - ▣ Google uwzględnia responsywność witryny wyliczając *PageRank*
- Pierwszy iPhone nie był wcale smartphonem
  - ▣ Nie można było instalować dodatkowych aplikacji
  - ▣ Był to *feature phone* z mobilną przeglądarką Safari
    - ...ale strony internetowe nie były gotowe na rewolucję mobilną

# Co jest nie tak z mobilnym Internetem?

- To nie oznacza, że powinniśmy zrezygnować z budowania witryn mobilnych
- **Musimy po prostu robić to lepiej!**
- Wiedzę na temat HTML, CSS, JavaScript można wykorzystać do budowania *hybrydowych aplikacji mobilnych*
  - ▣ Aplikacje publikowane w sklepach, które można instalować na urządzeniu mobilnym

## Interfejs nie-responsywny

**Przykład: 01-not-responsive**

# Wymagania dla stron mobilnych

- Wyeliminowanie konieczności skalowania strony i przewijania w poziomie
  - ▣ Dopasowanie układu i rozmiaru elementów do obszaru wyświetlania (ang. *viewport*) przeglądarki mobilnej
  - ▣ Dopasowanie rozmiaru czcionek w interfejsie
- Ograniczenie dystansu od szczytu strony do początku je zasadniczej treści
  - ▣ Ograniczenie konieczności przewijania w pionie
- Optymalizacja pod kątem obsługi palcami, a nie kursorem myszy
  - ▣ np. kłopotliwe zdarzenie **:hover**

# Jak spełnić te wymagania?

- Wymuszenie obszaru renderowania strony o szerokości wyświetlacza:  
`<meta name="viewport" content="width=device-width, initial-scale=1">`
- Wykorzystanie całej szerokości ekranu
  - ▣ np. eliminacja zbędnych marginesów
- Unikanie wyświetlania wielu elementów sąsiadujących ze sobą w poziomie
- Unikanie bezwzględnych wymiarów dla elementów determinujących układ strony
- Ukrywanie elementów poza zasadniczą treścią strony

# Jak spełnić te wymagania?

- Różne zestawy stylów dla różnych urządzeń

```
//style wspólne dla wszystkich urządzeń
body {
  font-family: 'Signika Negative', sans-serif; /*czcionka*/
}

//style dla smartphonów
@media (max-width: 768px) { /*...*/ }

//style dla tabletów
@media (min-width: 768px) and
      (max-width: 992px) { /*...*/ }

//style dla desktopów/laptopów
@media (min-width: 992px) { /*...*/ }
```

main.css

# Obszar wyświetlania strony

- Szerokość obszaru wyświetlania w dyrektywie @media nie musi przekładać się na rzeczywistą rozdzielczość wyświetlacza
- Rzeczywista rozdzielczość ekranu a rozdzielczość efektywna
  - ▣ Typowe gęstości ekranów: 72-400 dpi (*dots per inch*)
    - Pixel ma różne rozmiary fizyczne w zależności od gęstości ekranu
  - ▣ Density-independent pixel/device-independent pixel (dip)
    - Wirtualna jednostka miary pozwalająca na projektowanie interfejsu aplikacji niezależnie od dpi ekranu
    - Wyświetlenie jednego dip może wymagać więcej niż jednego pixela rzeczywistego
    - System operacyjny przelicza ilu pikseli użyć dla każdego dip

## Responsywny interfejs

**Przykład: 02-responsive**



# Organizacja zestawów stylów CSS

- Różne podejścia:
  - ▣ Osobne zbiory atrybutów dla każdej grupy urządzeń
    - Często też osobne dokumenty HTML, dedykowane urządzeniom
    - Optymalizacja interfejsu dla każdej grupy z osobna
  - ▣ Desktop-first
    - Domyślne style dla dużych urządzeń
    - Małe urządzenia dziedziczą style po dużych
    - W blokach `@media { /*...*/ }` zmieniane są style elementów, które nie spełniają mobilnych wymagań
  - ▣ Mobile-first
    - Domyślne style dla urządzeń przenośnych
    - Duże urządzenia dziedziczą po małych
    - W blokach `@media { /*...*/ }` optymalizowane są style elementów, które mogą skorzystać na większej przestrzeni

# Desktop-first

```
/* style dla desktopów/laptopów */
aside { width: 214px; float: left; }
main   { width: 690px; margin-left: 242px; }

/* style dla urządzeń mobilnych */
@media (max-width: 768px) {
    main, aside {
        width: auto;
        margin: 0px;
    }

    aside {
        float: none;
    }

    /*...*/
}
```

# Desktop-first

- Istniejące już strony, do których dodawany jest interfejs dla urządzeń mobilnych
- Istniejące elementy należy dostosować do mniejszych ekranów
- Jeśli jakiś element zostanie pominięty, interfejs będzie niepoprawnie wyświetlany na małym ekranie

# Mobile-first

```
/* style dla urządzeń mobilnych */
```

```
main, aside {  
    width: auto;  
    margin: 0px;  
}
```

```
/* style dla desktopów/laptopów */
```

```
@media (min-width: 992px) {  
    aside { width: 214px; float: left; }  
    main { width: 690px; margin-left: 242px; }
```

```
/*...*/
```

```
}
```

# Mobile-first

- Większość nowych aplikacji internetowych
- Jeśli interfejs mieści się na małym ekranie, to *tym bardziej* zmieści się na dużym
  - ▣ Może nie będzie *optymalny*...
  - ▣ ...ale zawsze *poprawny*!
- Jeśli jakiś element zostanie pominięty, wyświetli się przynajmniej *poprawnie*

Pytania?