



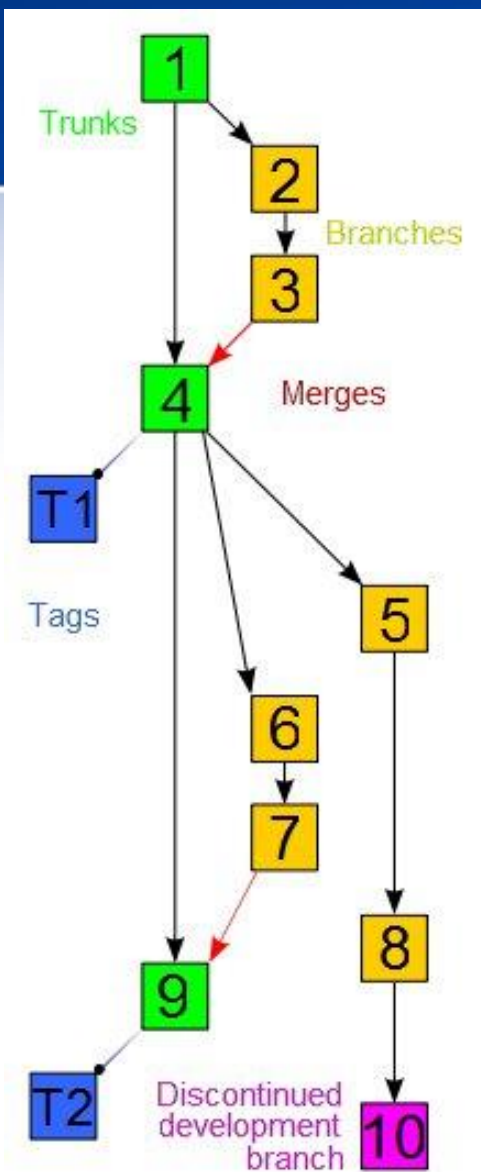
Technologie internetowe w programowaniu.

Bootstrap, Foundation, Zimit

dr Tomasz Jach
Instytut Informatyki,
Uniwersytet Śląski



- **System kontroli wersji** (ang. version/revision control system) jest to oprogramowanie służące do:
 - śledzenia zmian głównie w kodzie źródłowym
 - pomocy programistom w łączeniu i modyfikacji zmian dokonanych przez wiele osób w różnych momentach.

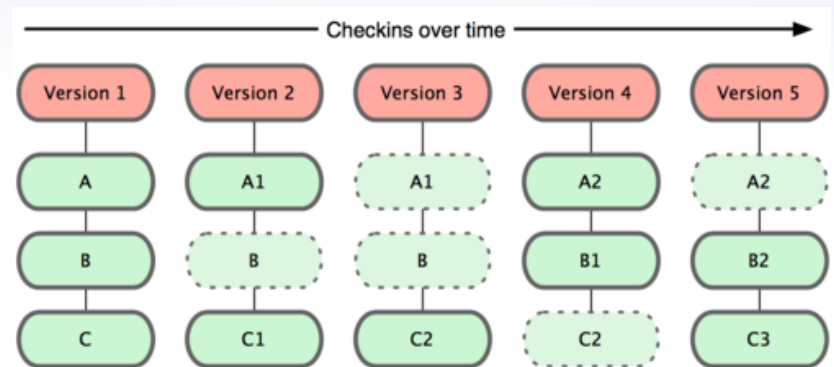
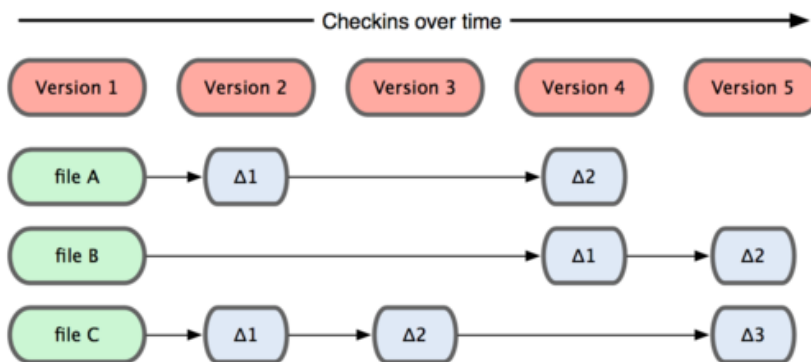


Rodzaje repozytoriów



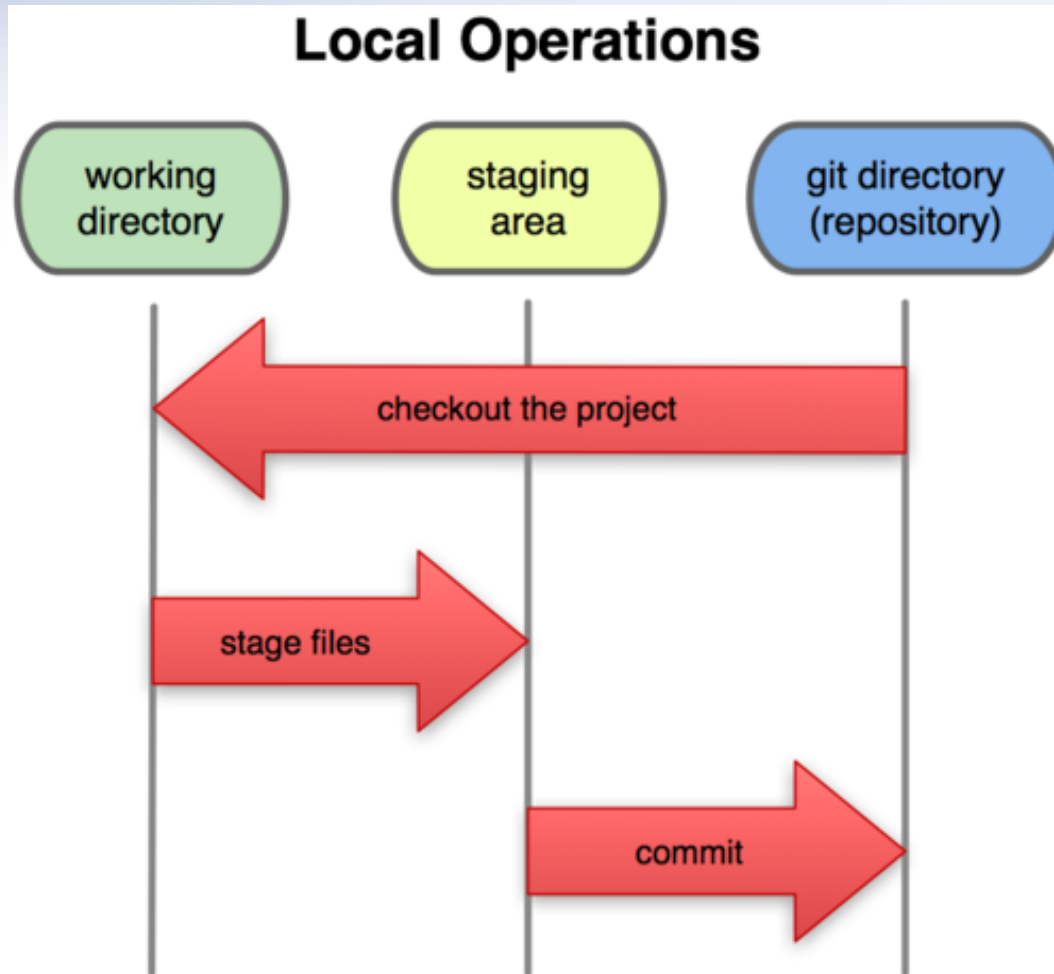
- Wolnodostępne systemy kontroli wersji:
 - Scentralizowane:
 - RCS
 - CVS
 - Subversion
 - GNU CSSC, klon SCCS
 - JEDI VCS
 - Rozproszone:
 - Bazaar
 - Codeville
 - Darcs
 - Git
 - GNU Arch
 - Mercurial
 - Monotone
- svk
 - Zamknięte (własnościowe) systemy kontroli wersji:
 - BitKeeper firmy BitMover
 - Code Co-op firmy Reliable Software
 - Perforce firmy Perforce Software
 - Rational ClearCase firmy IBM
 - Sablime firmy Lucent Technologies
 - StarTeam firmy Borland
 - Visual SourceSafe firmy Microsoft
 - Visual Studio Team Foundation Server firmy Microsoft

Jak to w ogóle działa?



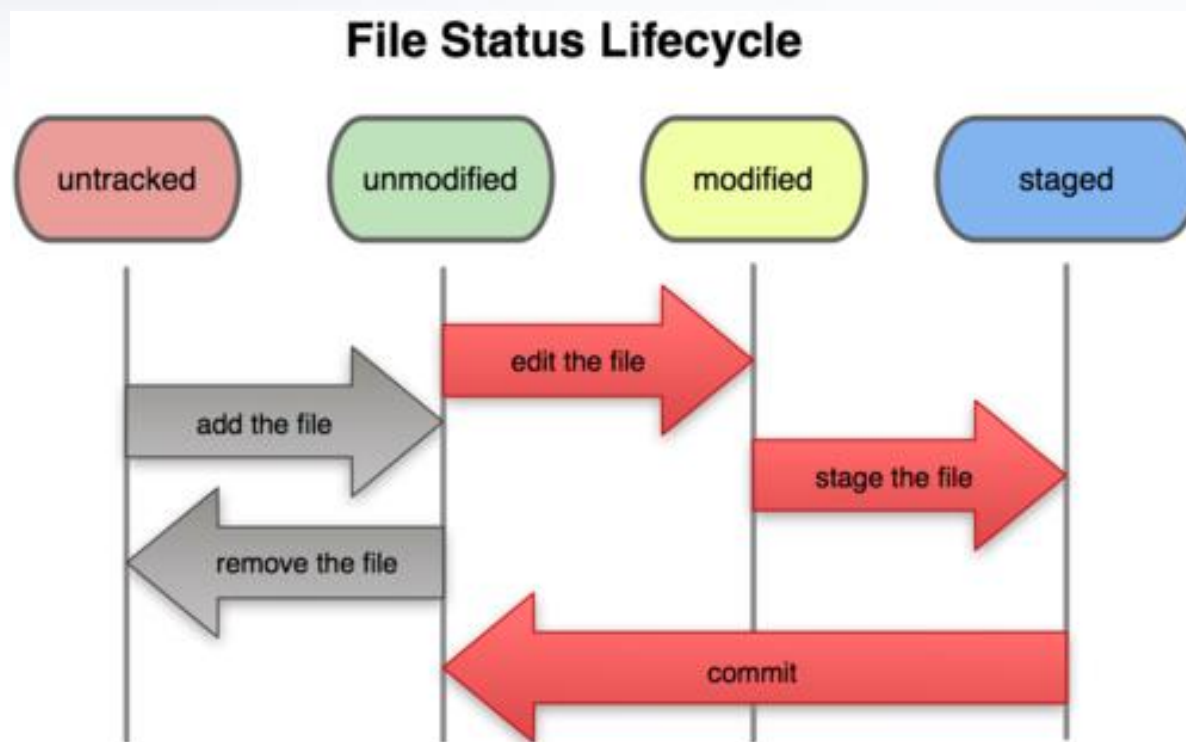


Lokalnie vs. globalnie





Status pliku



A na deser - GitHub



The screenshot shows the GitHub web interface. At the top, there is a navigation bar with the GitHub logo, a search bar, and links for Explore, Gist, Blog, and Help. The user profile 'tjach' is visible in the top right corner. Below the navigation bar, there is a section for the user 'tjach' with a 'ProTip™' message and a 'News Feed' link. The main content area features a navigation bar with 'News Feed', 'Pull Requests', 'Issues', and 'Stars'. Below this is a 'GitHub Bootcamp' section with the text 'If you are still new to things, we've provided a few walkthroughs to get you started.' and four numbered steps:

- Set Up Git**
A quick guide to help you get started with Git.
- Create A Repository**
Create the place where your commits will be stored.
- Fork a Repository**
Copy a repo to create a new, unique project from its contents.
- Be social**
Follow a friend. Watch a project.

Przydatne materiały



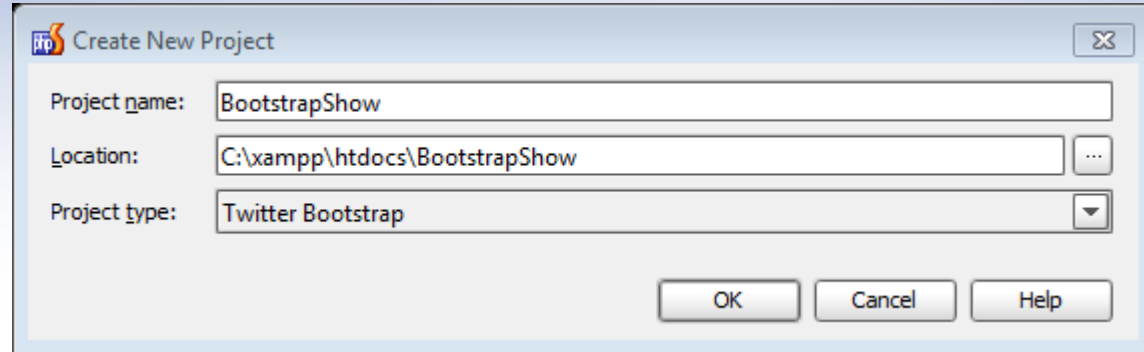
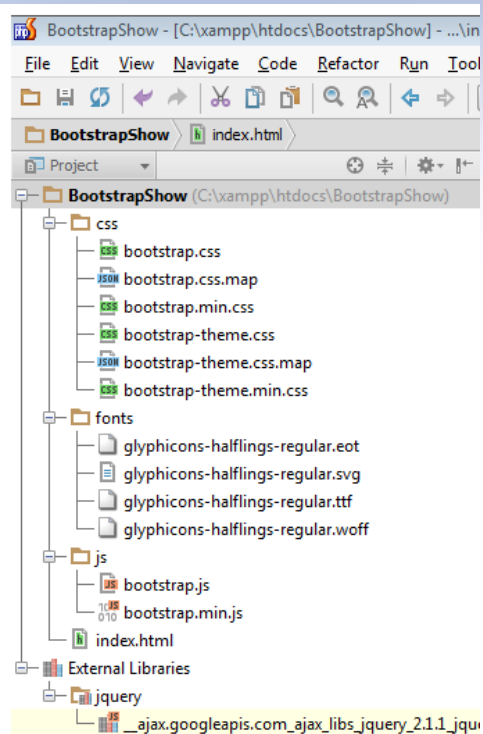
- <https://code.google.com/p/msysgit/downloads/list>
- <https://code.google.com/p/gitextensions/>
- <https://help.github.com/articles/set-up-git>
- <http://nathanj.github.com/gitguide/tour.html>
- http://codeasp.net/blogs/vivek_iit/microsoft-net/1881/how-to-use-github-with-visual-studio



Bootstrap

- Jest najpopularniejszym frameworkiem do tworzenia frontendu.
- Aby skorzystać należy go ściągnąć, rozpakować i ewentualnie usunąć zbędne rzeczy
- Można też ściągnąć tylko wybrane komponenty

Nowy projekt z Bootstrapem



```
html head
1 <!DOCTYPE html>
2 <html>
3 <head lang="en">
4 <meta charset="UTF-8">
5 <!-- CSS -->
6 <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css" media="screen">
7
8 <!-- JS -->
9 <script src="//ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
10 <script src="js/bootstrap.min.js"></script>
11
12 <title>Bootstrap demo</title>
13 </head>
14 <body>
15
16 </body>
17 </html>
```

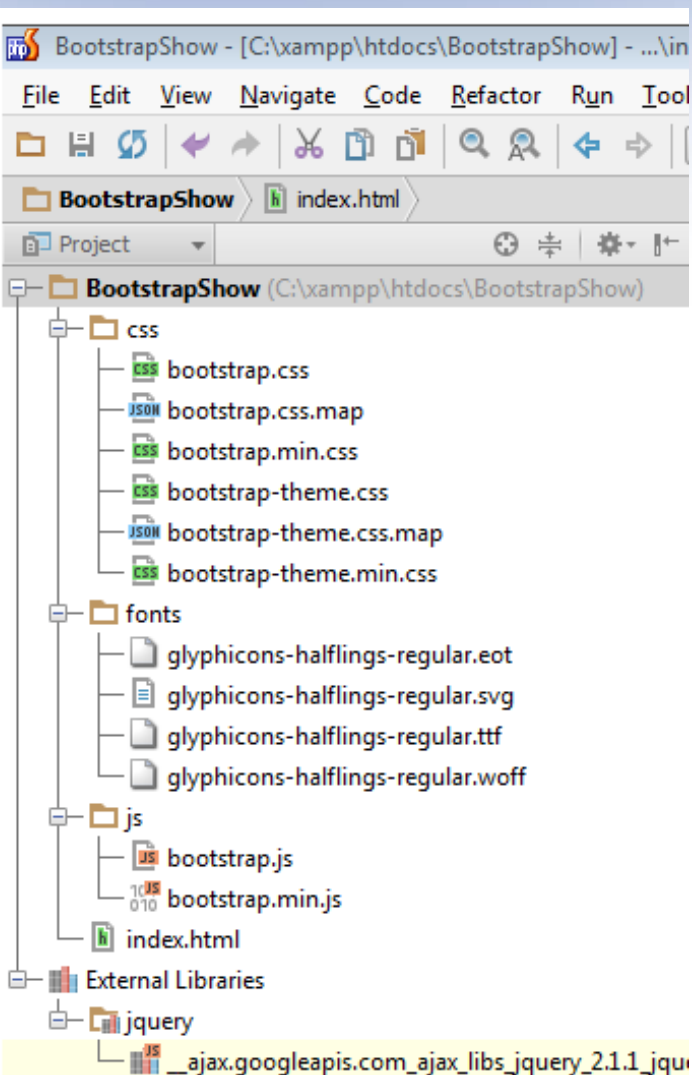
Nowy projekt z Bootstrapem



Po stworzeniu nowego projektu automatycznie dodały się katalogi **css**, **fonts**, **js**.

Dobrą konwencją jest takie rozdzielanie plików.

Zaimportowałem również bibliotekę jquery (Bootstrap wymaga).



Nowy projekt z Bootstrapem



```
html head
1 <!DOCTYPE html>
2 <html>
3 <head lang="en">
4   <meta charset="UTF-8">
5   <!-- CSS -->
6   <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css" media="screen">
7
8   <!-- JS -->
9   <script src="//ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
10  <script src="js/bootstrap.min.js"></script>
11
12  <title>Bootstrap demo</title>
13 </head>
14 <body>
15
16 </body>
17 </html>
```

Css z Bootstrapa

Najpierw jquery, potem JS z Bootstrapa.

Minimalizacja (minification) skryptów





Psujemy kod:

- Usuwamy spacje, komentarze, niepotrzebne białe znaki.
- Zmieniamy nazwy zmiennych na jednoliterowe
- Dzięki temu znacznie zyskujemy na miejscu i transferze

Rozmiar

Nazwa

 60 KB	bootstrap.js
 32 KB	bootstrap.min.js



Tworzymy szablon

1. Korzystamy z gotowych rozwiązań:

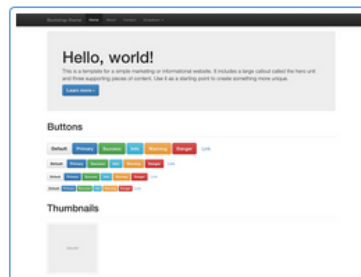
<http://getbootstrap.com/getting-started/#examples>

Using the framework



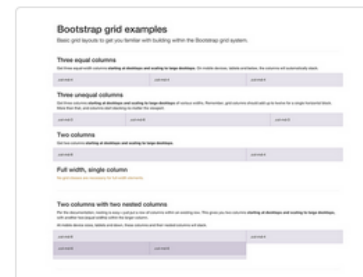
Starter template

Nothing but the basics: compiled CSS and JavaScript along with a container.



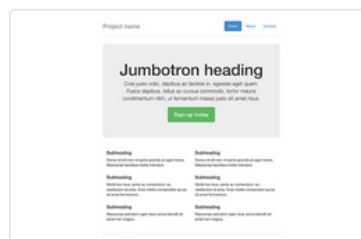
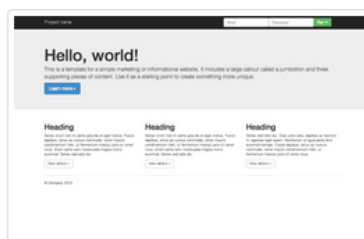
Bootstrap theme

Load the optional Bootstrap theme for a visually enhanced experience.



Grids

Multiple examples of grid layouts with all four tiers, nesting, and more.





Tworzymy szablon

2. Piszemy od zera 😊
3. Korzystamy z jednego z generatorów szablonu (<http://mashable.com/2013/10/20/bootstrap-editors/>).
My użyjemy LayoutIt!



LayoutIT!

The screenshot shows the LayoutIT! interface in a Google Chrome browser. The address bar displays www.layoutit.com/build?r=27363030. The top navigation bar includes links for [Edit](#), [Developer](#), [Preview](#), [Download](#), [Share or Save](#), [Clear](#), [Register](#), [Login](#), [Feedback](#), and [Donate!](#). The left sidebar is organized into four sections: **GRID SYSTEM**, **BASE CSS**, **COMPONENTS**, and **JAVASCRIPT**. Each section contains a list of elements with a **drag** icon. The main canvas shows a wireframe of a website layout with a **Container** containing a **Row** and a **Column**. The wireframe includes a navigation bar with **Brand**, **Link**, **Link**, **Dropdown**, a **Search** input, a **Submit** button, and another **Link** and **Dropdown**. Below the navigation bar is a large hero unit with the text **Hello, world!** and a subtext:

This is a template for a simple marketing or informational website. It includes a large callout called the hero unit and three supporting pieces of content. Use it as a starting point to create something more unique.

 A **Learn more** button is positioned at the bottom of the hero unit.



System gridowy

- Strona jest podzielona na wiersze i kolumny.
- Wiersze muszą znajdować się w elemencie `.container` (stała szerokość) lub `.container-fluid` (zmienna szerokość)
- Wiersze grupują elementy znajdujące się obok siebie w kolumnach
- Treść powinna być umieszczana w kolumnach, a kolumny to jedyni bezpośredni potomkowie wierszy.
- W CSSie mamy już zdefiniowane takie klasy jak `.row` czy `.col-xs-4`
- Puste miejsca otrzymujemy za pomocą paddingów.
- Każdy wiersz powinien składać się z kolumn sumujących się do 12 (czyli np. 4-4-4 lub 6-6 albo 2-4-6)
- Każda kolumna oprócz rozmiaru ma informacje o minimalnej szerokości urządzenia przed którą „się zawinie”



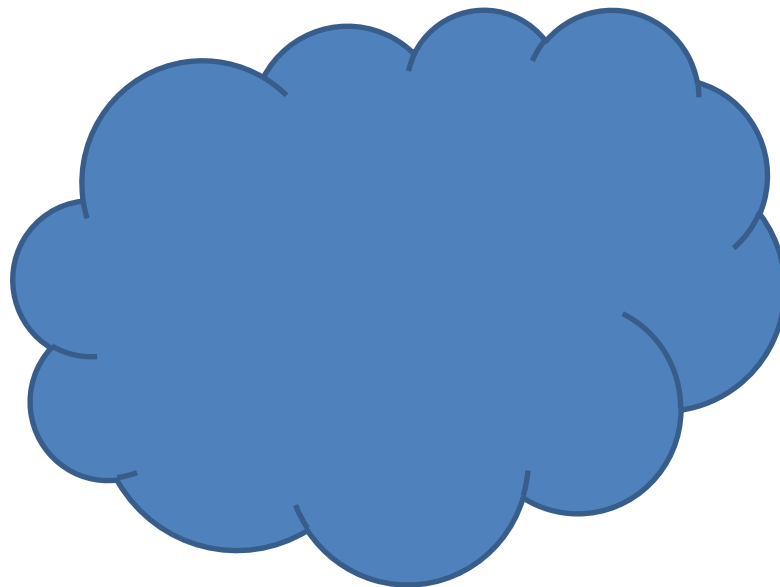
System gridowy

	Extra small devices Phones (<768px)	Small devices Tablets (≥768px)	Medium devices Desktops (≥992px)	Large devices Desktops (≥1200px)
Grid behavior	Horizontal at all times	Collapsed to start, horizontal above breakpoints		
Container width	None (auto)	750px	970px	1170px
Class prefix	<code>.col-xs-</code>	<code>.col-sm-</code>	<code>.col-md-</code>	<code>.col-lg-</code>
# of columns	12			
Column width	Auto	~62px	~81px	~97px
Gutter width	30px (15px on each side of a column)			
Nestable	Yes			
Offsets	Yes			
Column ordering	Yes			



Elementy CSS

- Live demo: <http://getbootstrap.com/css/>





Komponenty

- Live demo: <http://getbootstrap.com/components/>

Glyphicons

Dropdowns

Button groups

Button dropdowns

Single button dropdowns

Split button dropdowns

Sizing

Dropup variation

Input groups

Navs

Navbar

Breadcrumbs

Pagination

Labels

Badges

Jumbotron

Page header

Thumbnails

Alerts

Progress bars

Media object

List group

Panels

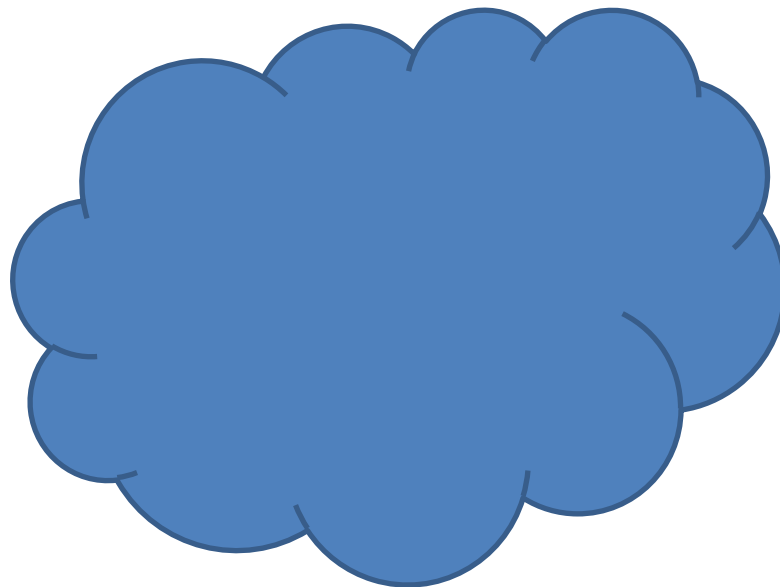
Responsive embed

Wells



Elementy JS

- Live demo: <http://getbootstrap.com/javascript/>





Zadanie domowe

- Bootstrap jest najbardziej złożony spośród trzech frameworków do front-endu. Ale i najpopularniejszy. Reszta poniżej jest wykonana analogicznie:
- Poczytaj o Foundation:
<http://foundation.zurb.com/docs/>
- Poczytaj o Zimit:
<http://firezenk.github.io/zimit/design.html>
<http://firezenk.github.io/zimit/components.html>