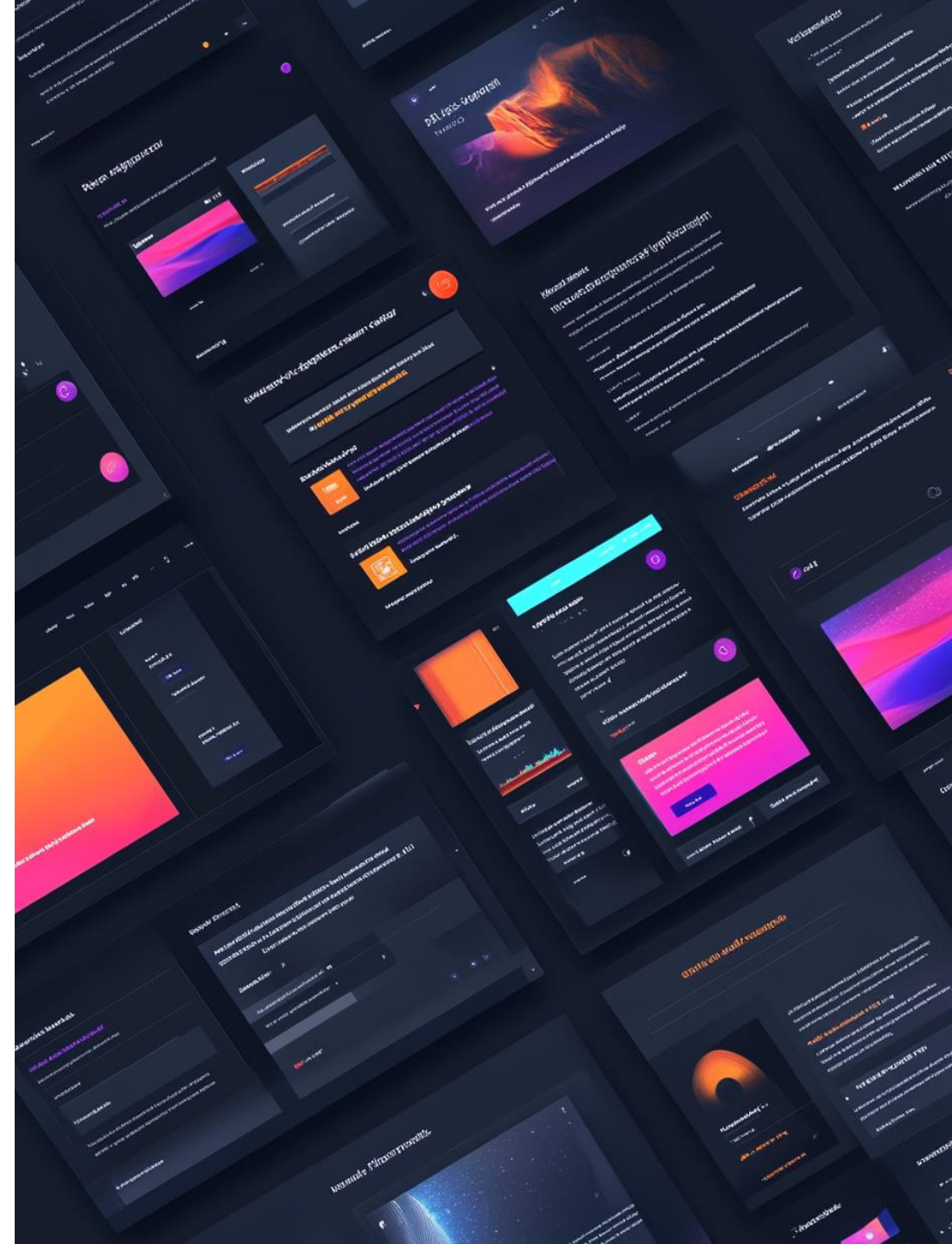# Web Application Development

Bootstrap

Ing. Michal Radecký, Ph.D.
www.cs.vsb.cz/radecky

FrontEnd frameworks

They offer tools for effective development of webdesign

They are based on CSS frameworks (eg. Blueprint – grid, typography, form elements)

They offer …

- Responsibility and multiplatform support
- Layout development
- Typography
- **User controls**
- Extensions and libraries
- Accesibility
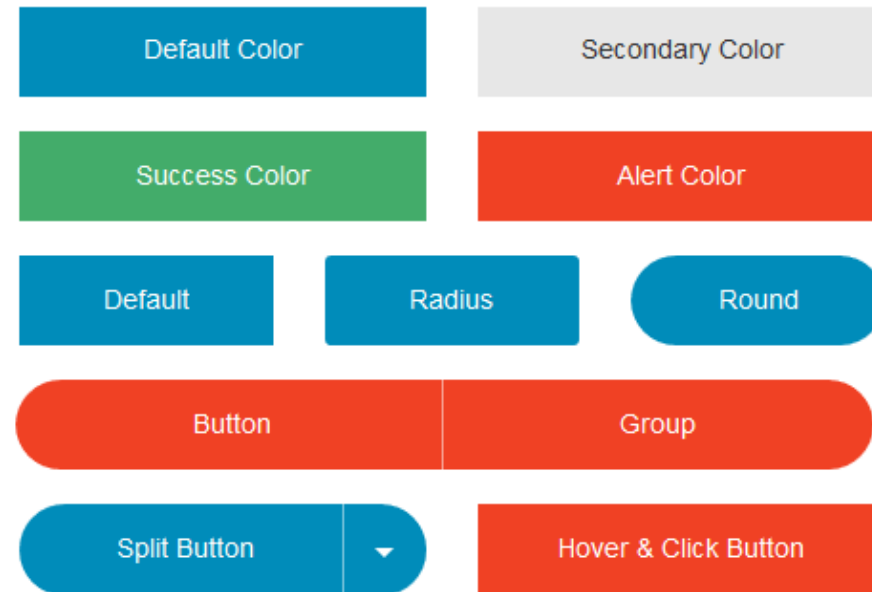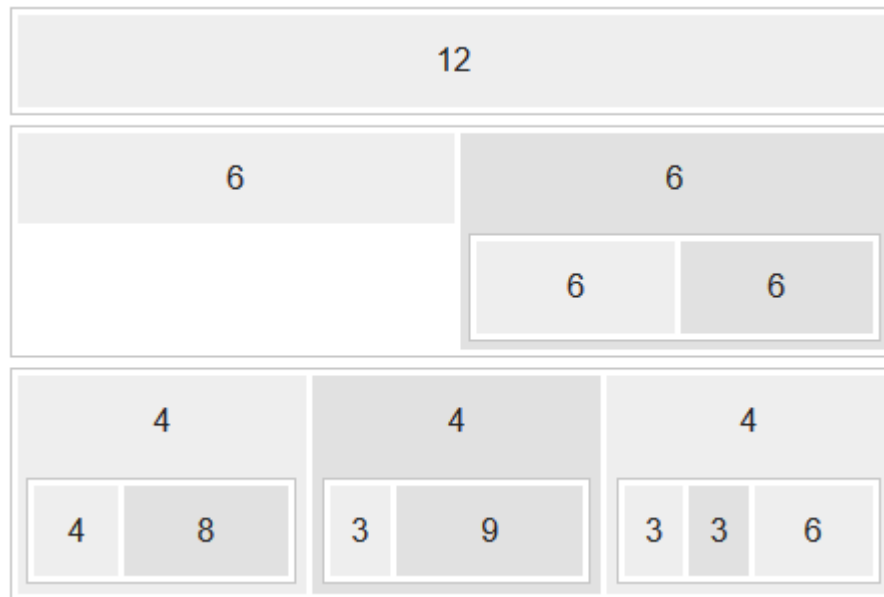
Typically mobile-first approach

Mix of HTML, CSS and JS

Bootstrap, Foundation, Skeleton, etc.

Foundation

http://foundation.zurb.com/

Current version 6, MIT licence, 250 kb

Preprocesor SASS, jQuery

Bootstrap

http://getbootstrap.com/

Current version 5, MIT licence, 250 kb

Preprocesor LESS (portace SASS, version 4 in native way), jQuery

FrontEnd frameworks utilization

It is always necessary to take into account efficiency and range of usage for a given project

Pros

- Fast development within „standardized" environment
- Native support of CSS preprocessors
- Implementation of widely used controls (accroding to multiplatform operation)
- Small entry barrier with spectacular results
- Optimization based on composite construction
- Themes

Cons

- Can be too complex for some purposes
- Relation to a huge set of other tools and connected libraries (node.js, Grunt, Rails, Compass, etc.)

Bootstrap - grid

https://www.w3schools.com/bootstrap5/index.php

Fundamental view on page layout - **grid system**
(12 cells)

Container (responsive/fluid) – row – column

Responsibility based on resolution

```
.col-    (extra small devices - screen width less than 576px)
.col-sm- (small devices - screen width equal to or greater than 576px)
.col-md- (medium devices - screen width equal to or greater than 768px)
.col-lg- (large devices - screen width equal to or greater than 992px)
.col-xl- (xlarge devices - screen width equal to or greater than 1200px)
.col-xxl- (xxlarge devices - screen width equal to or greater than 1400px)
```

Bootstrap – look and feel

Native styling – direct styling of some elements (h1-6, attr, blockquote, code, …)

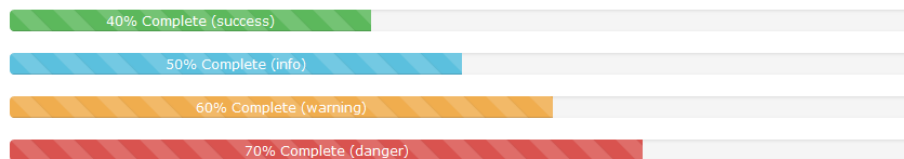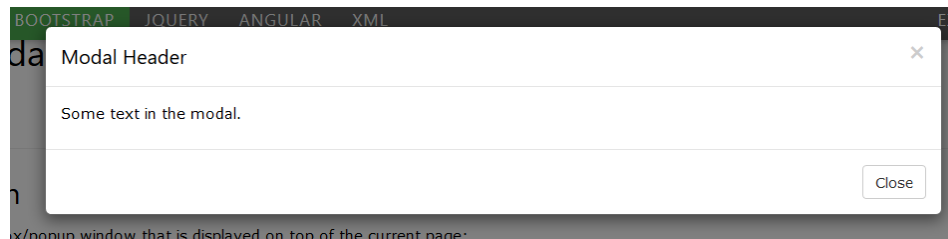**User defined styling** – based on assigned classes

Basic approach is not to describe styling directly, but styling should be based on predefined styles (classes). Own styling is not necessary in principle.

There is JS API based on jQuery for more sophisticated interaction

Bootstrap – visual elements

Wide selection of visual elements (including some functionality) based on HTML specification and CLASS assignment

Jumbotron, web heading, boxes, buttons, form elements, icons, modal windows, etc.

Bootstrap – interaction with user

- It is possible to control many elements based on user behavior

- Usage of CSS or JS is not necessary

- Principle is based on **data-** attributes usage
  eg. data-dismiss, data-toggle, data-target

- Usage of **aria-** attributes for accessible web support (Accessible Rich Internet Applications) inc. Attribute **role**

Bootsrap – plugins and themes

Possibility to use plugins

- Native: Tooltip, Scrollspy, Carousel, Affix
- External: https://flatlogic.com/blog/top-24-free-essential-plugins-to-extend-bootstrap/

Possibility to „overwrite" of styles based on themes

- Based on inheritance and overwriting of CSS – never modify original Framework specification
- External predefined extensions (themes) (beware of version mishmash)

Summary

- FrontEnd frameworks are very effective way to fast development of web applications

- Many things can be implemented without any knowledge of CSS or JS. However, there can be a huge relation to many other technologies in teh case of complex implementations

- Effective usage of frameworks on higher level is based on CSS preprocessors and other integration tools