The Mobile Web

Web Engineering
Prof. D. König

Responsive Design

Respond to varying screen sizes/resolutions

Prerequisite Knowledge:

HTML, CSS, JavaScript, Web-MVC

Testing the WebMile

The variety of devices, screen sizes, and resolutions makes any testing difficult.

Provider and products change quickly: google, microsoft, saucelabs, mobiready, …

Approaches

Flexible layout (CSS)

Media queries (CSS)

Dynamic in-page logic (HTML, JS)

Serve different views per capability (MVC)
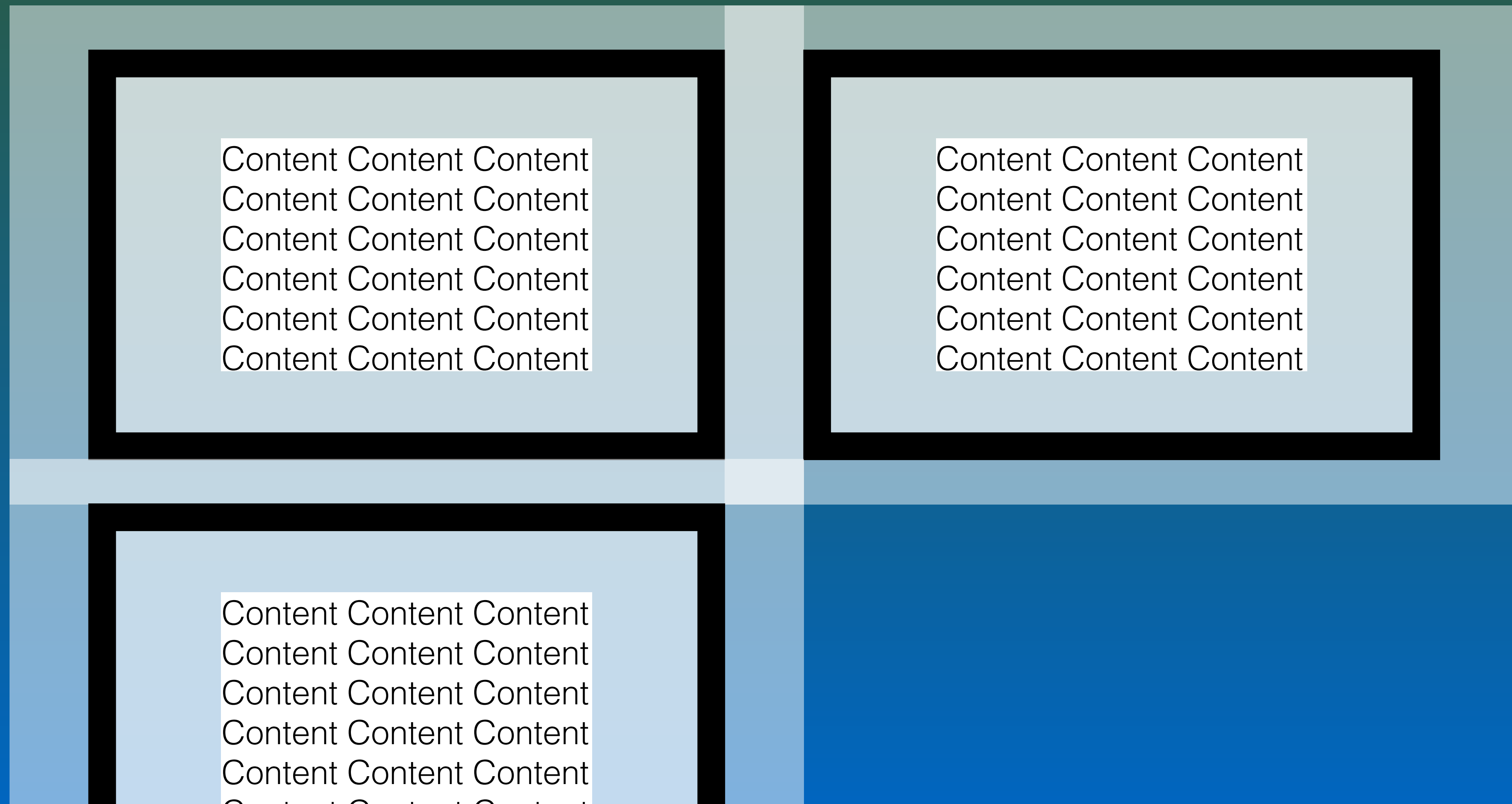
Flexible Layout in CSS

width: 62%;

Kevin Powell:

https://www.youtube.com/watch?v=R7gqJkdc5dM

Flexible Layout { float: left; }

Content Content Content
Content Content Content
Content Content Content
Content Content Content
Content Content Content
Content Content Content

Content Content Content
Content Content Content
Content Content Content
Content Content Content
Content Content Content
Content Content Content

Content Content Content
Content Content Content
Content Content Content
Content Content Content
Content Content Content

Media Query (CSS)

```css
#title { width: 50%; }

@media screen and (max-width: 800px) {
    #title { width: 100%; }
}
```

Media Query Attributes

max-width, max-device-width,
min-width, min-device-width, *(height)*
orientation (portrait, landscape),
[min-,max-,device-]aspect-ratio,
color, resolution, touch-enabled, ...

Combinations

Media Queries work very nicely in combination with grid
where the grid templates change based on available space

Dynamic in-page logic

```
<body onresize="adapt()">

<script>
  function adapt() { ...; screen.size ... }
</script>
```

View per Capability

How to detect the capability?

How to change the view?

Grails Plugin Browser Detection:
https://github.com/mathifonseca/grails-browser-detection

Request - Response

```
class MyController
    def myAction(en, exam)
```

request

response

client

view

Grade Calculator

localhost:8080/static/GradeCalculator.html

continuous assessment grade

final exam

Daten absenden

How to change the View?

Use different CSS

Select a different layout

Rendering a different view

Special Topics

Distinguishing
screen, window, page, viewport

Using JavaScript to detect capabilities
if (window.DeviceOrientationEvent) …

Special Topic: Graphics

`<picture> <source srcset=".." media="..">`

`<img srcset="big.jpg 2x, small.jpg 1x">`

`<svg>`

When to use: rule of thumb

CSS float, flex, grid   *Always consider*

@media   *Mostly static content*

onresize   *Fine-grained control*

MVC   *Default*

Simple and proper HTML
displays best
on any device