Practical Course:
Web Programming

Human-Centered Ubiquitous Media
Prof. Dr. Albrecht Schmidt, Fiona Draxler,
Thomas Kosch

Winter Term 18/19
Agenda

• Organization of this course

• Principles and introduction of the web

• Presentation Exercise 1
Organization

- **Dozent**: Prof. Dr. Albrecht Schmidt
- **Kontakt**: pwp@um.ifi.lmu.de
- **Semesterwochenstunden**: 4
- **ECTS-Credits**: 6
- **Modul**: Master P5.0.2 oder P5.0.4: Gruppenpraktikum zu fortgeschrittenen Themen der Informatik I oder Informatik II
  Nach Rücksprache mit dem Prüfungsausschuss auch Anerkennung für P2, P3 oder P6 (Vertiefende Themen für Master) möglich
  Bachelor P17, P18: Vertiefende Themen der Medieninformatik I oder II

**Supervisor**: Fiona Draxler, Thomas Kosch

If you have any questions, write a mail, pass by the office, or ask in the Slack channel: [https://mimuc.slack.com/messages/pwp-ws1819/](https://mimuc.slack.com/messages/pwp-ws1819/)
Grading

• Individual level
  • We will conduct two interviews (ca. 10 minutes) with each of you
  • You will be invited to present your results on a regular basis
  • Assessment of your exercises, coding style, and oral participation

• Group level
  • Result of the group project
  • Coding style of the group project
## Semester Plan

<table>
<thead>
<tr>
<th>Datum</th>
<th>Thema</th>
</tr>
</thead>
<tbody>
<tr>
<td>22.10.18</td>
<td>Foundations WWW (Organization, Grading, Webserver, HTTP, HTML)</td>
</tr>
<tr>
<td>29.10.18</td>
<td>Webserver implementation presentation, Web analytics, cookies, security and privacy</td>
</tr>
<tr>
<td>05.11.18</td>
<td>NodeJS (or similar), asynchronous communication</td>
</tr>
<tr>
<td>12.11.18</td>
<td>Express (or Restify), implementing and using microservices (e.g., clmtrackr)</td>
</tr>
<tr>
<td>19.11.18</td>
<td>Typescript, Coding conventions (styleguide), JSLint</td>
</tr>
<tr>
<td>26.11.18</td>
<td>Angular (or similar), Task implementation (e.g. memory game, hardware integration using NodeMCUs)</td>
</tr>
<tr>
<td>03.12.18</td>
<td>JSON, Messaging + Support</td>
</tr>
<tr>
<td>10.12.18</td>
<td>Presentation of projects (Fake News Detector, Web page and visualization), finding teams, Interviews I</td>
</tr>
<tr>
<td>17.12.18</td>
<td>Mock-ups bauen</td>
</tr>
<tr>
<td>07.01.19</td>
<td>UI Implementation, Database</td>
</tr>
<tr>
<td>14.01.19</td>
<td>Implementation Logic I, PM2</td>
</tr>
<tr>
<td>21.01.19</td>
<td>Implementation Logic II, Testing</td>
</tr>
<tr>
<td>28.01.19</td>
<td>Implementation Logic III, final presentation, (eventually Interviews II)</td>
</tr>
<tr>
<td>04.02.19</td>
<td>Interviews II Backup</td>
</tr>
</tbody>
</table>
A (coarse) Architecture of the World Wide Web

WWW

HTML (Design)

HTTP (Communication)

URL (Location)
Hypertext Markup Language (HTML)

HTML: This is what your browser sees!

```
<html>
  <body>
    <h1>Hello Web Programming Course :)</h1>
    <h2>This is a simple HTML example!</h2>
    <p>In this course, you will learn the principles behind the web and develop your own web application!</p>
  </body>
</html>
```

This is what you see!

Hello Web Programming Course :
This is a simple HTML example!
In this course, you will learn the principles behind the web and develop your own web application!


Visit a webpage, right click and press “View page source”
Other Web Technologies

PHP

```html
<!DOCTYPE html>
<html>
<body>
<h1>My first PHP page</h1>
<?php
    echo "Hello World!";
?>
</body>
</html>
```

Javascript

```javascript
alert( 'Hello, world!' );
```

CSS

```
.center {
    text-align: center;
    color: red;
}
```
Hypertext Transfer Protocol (HTTP)

**Example GET Request:** `curl -I http://ubicomp.net`

**Response:**

HTTP/1.1 200 OK
Date: Sun, 15 Apr 2018 12:44:56 GMT
Server: Apache
Last-Modified: Wed, 01 Nov 2017 16:33:19 GMT
ETag: "702-55cee6de9e9fe"
Accept-Ranges: bytes
Content-Length: 1794
Vary: Accept-Encoding
Content-Type: text/html
Uniform Resource Locator (URL)

• A simple web address
What is a Webserver?
What Webserver do you already know?
Webserver Technologies

<table>
<thead>
<tr>
<th>Apache:</th>
<th>Nginx:</th>
<th>NodeJS:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Handles static content</td>
<td>• Handles static content</td>
<td>• Handles dynamic content</td>
</tr>
<tr>
<td>• Separate threading</td>
<td>• No separate threading</td>
<td>• Asynchronous requests</td>
</tr>
<tr>
<td>• Old and complex</td>
<td>• Handles requests fast</td>
<td>• Handles requests fast</td>
</tr>
<tr>
<td></td>
<td>• Lightweight and resource-friendly</td>
<td>• Lightweight and resource-friendly</td>
</tr>
</tbody>
</table>
How do you usually search the internet?
How does your browser find the desired content?
Hosting your own Webserver at your Home

• For example with a NodeMCU (ESP8266)