

The background features a collage of various icons related to web engineering and technology. These include a hand cursor pointing at a screen, a globe, a target, a magnifying glass over a globe, a document with a checkmark, a smartphone, a laptop, and a gear. The icons are rendered in a grayscale, semi-transparent style, creating a layered effect.

# Web Engineering

**MScIS Peter Grosskopf**  
**MScIS Thomas Hollstegge**

**Fachbereich Wirtschaft – Fachhochschule Münster**  
**Bachelor of Science Wirtschaftsinformatik Sommersemester 2014**

# Agenda

- Organisatorisches
  - Veranstaltungsdaten
  - Vorstellung der Dozenten
  - Themen





# Organisatorisches

# Lernziele

- Einblick in komplexere Themenfelder der Software Entwicklung als Fortsetzung von Web Entwicklung 2
- Überblick über verwandte und weiterführende Technologien
- Schreiben einer wissenschaftlichen Ausarbeitung

# Verlauf der Veranstaltung

- Vorlesung im Seminar-Stil
  - Keine regelmäßigen Vorlesungen
  - Keine Übungen

# Leistungsnachweis / Klausur

- Leistungsnachweis: Ausarbeitung über ein fortgeschrittenes Thema der Web Entwicklung
- Gruppenarbeit: 4 Personen
- Themen:
  - S.U.
- Abgabe:
  - Thesenpapier und Gliederung (1 bis 2 Seiten)
  - Ausarbeitung (20 bis 25 Inhaltsseiten)
  - Präsentation (30 min + 10 min Diskussion)
- Abgabetermine:
  - Thesenpapier und Gliederung am 25.04.2014
  - Präsentationen in KW 24
  - Ausarbeitungen am 23.06.2014

# Inhalte Thesenpapier/Gliederung

- Gliederung

- Überblick über die einzelnen Kapitel
- Ungefähre Einteilung der Seiten

- Thesenpapier

- Welche Inhalte und Fragestellungen werden in der Ausarbeitung behandelt?
- Erste Stichworte zu den einzelnen Themen

# Inhalte Ausarbeitung

- Einleitung in das Thema – Warum ist das Thema wichtig? Welche Motivation wird verfolgt?
- Hauptteil – was sind die wesentlichen Fragenstellungen des Themas? Priorisierung?
- Fazit und Ausblick – wie wird sich das Thema weiter entwickeln?



# Inhalte Präsentation

- Vorstellung des Themas, damit die anderen Teilnehmer des Kurses einen Überblick bekommen
- Diskussion – um auf Nachfragen zu reagieren und ggf. noch weitere Details zu besprechen

# Veranstaltungstermine

Woche	Datum	Veranstaltung
KW 11	14.03.2014	<b>V (14h–18h):</b> Organisatorisches + Themenvergabe
KW 17	25.04.2014	Abgabe Thesenpapier + Gliederung
KW 20	16.05.2014	<b>V (14h–18h):</b> FAQ + Gruppenfeedback
KW 24	11.06.2014	<b>V (13h–18h):</b> Präsentationen
KW 24	12.06.2014	<b>V (13h–18h):</b> Präsentationen
KW 24	13.06.2014	<b>V (13h–18h):</b> Präsentationen
KW 26	23.06.2014	Abgabe der Ausarbeitungen
KW XX	Juni 2014	Notenvergabe

# Homepage der Veranstaltung

<http://zwtg.de/WE2014-Gruppen>  
(Gruppenregistrierung)

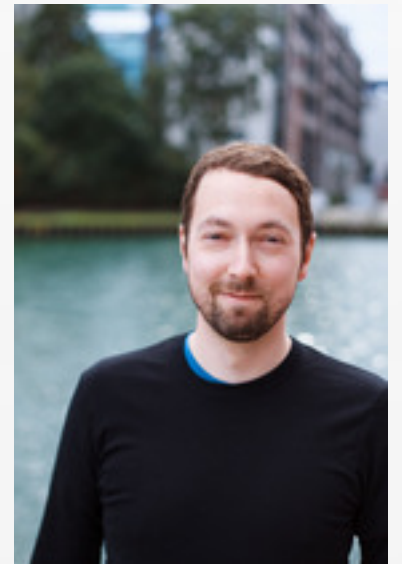
<https://github.com/Web-Engineering-SoSe2014/Lecture>  
(Slides dieser Vorlesung)



# **Vorstellung der Dozenten**

# Peter Grosskopf (33)

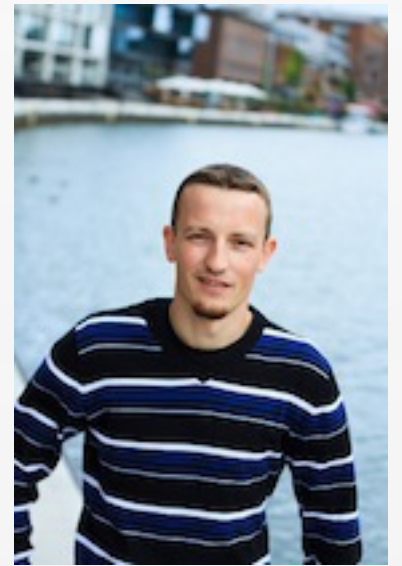
- Geboren in Düsseldorf
- Studium der Wirtschaftsinformatik an der WWU Münster, Abschluss MScIS
- Während des Studiums:
  - Freelancing als Software Entwickler in Enterprise und Startup Firmen
  - Start eigener Projekte ...
  - ... z.B. Gründung von Zweitag
- [https://www.xing.com/profile/Peter\\_Grosskopf](https://www.xing.com/profile/Peter_Grosskopf)





# Thomas Hollstegge (28)

- Studium der Wirtschaftsinformatik an der WWU Münster, Abschluss MScIS
- Mitarbeiter bei Zweitag seit 2008
- Experte für Infrastruktur und Skalierung





# Themen

# Themenüberblick

Website Performance	Server und Deployment	Asynchronität
Frontend Technologien	Geo-Anwendungen	Software Architekturen
HTML5 und CSS3	Such-Technologien	Caching Strategien
Gem-Entwicklung	Single Page Webapps	Mobile Web Apps
Ruby on Rails 4.0	NoSQL	
Ruby 2.0	node.js	
Cloud Computing	Fortgeschrittene JS-Techniken	
Security	Animationen und Charts	

# Website Performance

- Website Performance allgemein und mit Ruby on Rails (client- und serverseitig)
- Performance wichtig für eine positive User-Erfahrung
- Kennzahlen zur Messung der Performance

# Frontend Technologien

- Frontend Technologien – Haml, Compass und Sass
- Haml: Vereinfachte html-Template-Erstellung
- Sass: CSS Präprozessor z.B. verschachtelte Regeln
- Compass: CSS Framework, das Sass nutzt. Vereinfachte CSS3-Verwendung



# HTML5 und CSS3

- HTML5, CSS3 und zukünftige Entwicklungen
- Erweiterte Browser– (z.B. Offline–Caching) und Darstellungs–Features (z.B. runde Ecken, Schatten)
- MediaQueries und responsive Design
- Wie geht es weiter? HTML6? CSS4? Weitere W3C Drafts?

# Gem-Entwicklung

- Gem und OpenSource Entwicklung
- Wie Entwickelt man ein gem?
- Wo hostet man gems?
- Was muss man bei OpenSource Entwicklung beachten (z.B. Lizenzen, Rechte, ...)?

# Ruby on Rails 4.0

- RoR 4.0 bringt einige neue Features, u.a. Strong Parameter, Turbolinks, ...
- Wie kann man bereits jetzt RoR 4.0 verwenden?
- Wie updated man bestehende Anwendungen? Müssen alte Anwendungen angepasst werden?

# Ruby 2.0

- Was sind die wesentlichen Änderungen von Ruby 2.0?
- Was kommt nach Ruby 2.0?

# Cloud Computing

- IaaS, PaaS, SaaS
- AWS, EC2, S3, heroku
- Was steht hinter diesen Technologien und Produkten?
- Wie kann man Cloud Computing bei Ruby on Rails Anwendungen und in der Entwicklung einsetzen?



# Security

- Web-App und Ruby on Rails Security
- Welche Arten von Bedrohungen und Angriffen gibt es (CSRF, SQL Injection, ...) und wie funktionieren sie?
- Wie kann man Rails-Anwendungen schützen und absichern?

# Server und Deployment

- Server Technologien und Deployment von Ruby on Rails Anwendungen
- Varnish, Apache, Passenger, nginx, Chef und capistrano
- Wie sieht ein Ruby on Rails Hosting-Szenario auf einer dedizierten Linux-Maschine aus?

# Geo-Anwendungen

- Wie können Geo-Daten in Ruby on Rails verarbeitet werden?
- Welche fortgeschrittenen Funktionen liefert Postgres mit?
- Map-Technologien (Google Maps, OSM, Mapbox)

# Such-Technologien

- Such-Technologien mit Ruby on Rails
- ElasticSearch, Sphinx, Lucene/Solr
- Wie und warum bindet man Suchindexe in Web Anwendungen ein?
- Wie aktualisiert man den Index bei Änderungen an den Daten?

# Single Page Webapps

- Single Page Webapps mit backbone.js / ember.js und Ruby on Rails
- Bei Client-Side-Applications stellt das Datenmanagement und die Struktur eine große Herausforderung dar
- Wie muss eine Ruby on Rails Anwendung aufgebaut sein?
- Wie entwickelt man eine API?



# NoSQL

- Warum NoSQL?
- CAP Theorem
- Welche alternativen DBMS gibt es und wie lassen sie sich gruppieren?
- Welche Anwendungskontexte gibt es?

# node.js

- Serverseitiges Javascript
- Wie funktioniert node.js?
- Welches Architekturparadigma steht dahinter?
- Wie entwickelt man mit node.js?

# Fortgeschrittene JS-Techniken

- Fortgeschrittene Javascript-Techniken (Javascript-Grundlagen, Module, require.js)
- Module stellen eine Möglichkeit zur Kapselung von Funktionen in Javascript dar
- require.js: Für die Nutzung im Browser optimierte Datei- und Modul-Lade Bibliothek
- Wie schreibt man hochwertiges Javascript?

# Animationen und Charts

- Große Vielfalt an JS Bibliotheken für Charts/Animationen
- D3 bietet eine sehr große Anzahl an möglichen Charts
- Welche Bibliotheken gibt es? Wie wendet man sie an?
- Wie bindet man sie an eine Rails-Anwendung an? Wie filtert man Daten z.B. in einem Dashboard?

# Asynchronität

- Message Queues, EventMachine
- Anfragen, die einen längeren Zeitraum zur Bearbeitung benötigen, werden asynchron bearbeitet
- Dies Verhindert, dass der User auf den Abschluss der Anfrage warten muss, um fortfahren zu können
- EventMachine Und Message Queues bieten Möglichkeiten zur Umsetzung einer solchen Asynchronen Bearbeitung

# Software Architekturen

- Bezeichnet die grundlegenden Komponenten und ihr Zusammenspiel in einem Softwaresystem
- Client-Server, Schichten, Peer-to-Peer, usw.
- EAI, SOA



# Caching Strategien

- Welche Möglichkeiten und Strategien gibt es bei Ruby on Rails
- Was muss man beim Caching beachten (Sweeping, Verteilte Server, usw.)?
- Welche fortgeschrittenen Caching-Möglichkeiten gibt es (Memcache, NoSQL, ...)?

# Mobile Web Apps

- Steigender Traffic von mobilen Browsern
- Wie setzt man Mobile Web Apps mit Ruby on Rails um?
- Wie sieht ein Mobile First Entwicklungsansatz aus?
- Sencha, JQuery Mobile, Komponenten (KendoUI, ...)
- Responsive Design

# Fazit

- Los geht's! :-)