



Projektseminar „Datenverwaltungssysteme“

- Themenüberblick und Themenvergabe -

14.04.2010

Artur Andrzejak

[<andrzejak@zib.de>](mailto:andrzejak@zib.de)

Jürgen Broß

[<juergen.bross@fu-berlin.de>](mailto:juergen.bross@fu-berlin.de)

Organisatorisches

- › Projektseminar gibt 8 ECTS Credits bzw. 4 SWS
- › Jede Gruppe wird mind. einem Betreuer zugeteilt
- › Standardmäßig gibt es einen *Projektschein*
- › Bei Erfüllen zusätzlicher Kriterien wahlweise *Seminarschein*
 - › Zusätzlicher Vortrag (~20 Min.)
 - › Zusätzliche Ausarbeitung (~8 Seiten)

- › Detaillierte Informationen sind auf der Homepage zu finden
 - › <http://www.inf.fu-berlin.de/lehre/SS10/DBS-Projektseminar/>

Organisatorisches - Ablauf

- › Vorstellungstermin
 - › Übersicht über Projektthemen
 - › Vergabe der Themen an Teilnehmer
 - › ggf. Aufteilung der Projektgruppen auf die Präsentationstermine

- › 1. Präsentationstermin 2. Maiwoche
 - › Jede Gruppe hält einen 15 minütigen Vortrag über ihr Projekt
 - › Einführung in die Problemstellung
 - › Mögliche Lösungsansätze
 - › Bisher geleistete Arbeit
 - › Projektplan
 - › Abgabe des detaillierten Projektplans
 - › Die Teilnehmer müssen die gewünschte Scheinart festlegen (ggf. Festlegung zu welchem Termin Seminarvortrag gehalten wird)

Organisatorisches - Ablauf

› 2. Präsentationstermin

- › Eingeteilte Gruppen halten einen 15 minütigen Vortrag über ihr Projekt
 - › Problemstellung, Fortschritt
 - › Abgleich der eigenen Arbeit mit dem Projektplan
- › Dem Betreuer muss mindestens 5 Tage *vor* dem Termin die fertige Präsentation vorliegen!
- › Evtl. zusätzliche Seminarvorträge

› 3. Präsentationstermin

- › Eingeteilte Gruppen halten einen 15 minütigen Vortrag über ihr Projekt
 - › Problemstellung, Fortschritt
 - › Abgleich der eigenen Arbeit mit dem Projektplan
- › Dem Betreuer muss mindestens 5 Tage *vor* dem Termin die fertige Präsentation vorliegen!
- › Evtl. zusätzliche Seminarvorträge

Organisatorisches - Ablauf

- › Abschlusspräsentation Ende Juli 2010
 - › Alle Gruppen präsentieren ihre Ergebnisse
 - › Vortrag über Problemstellung, Herangehensweise, Probleme und Lösungen
 - › Demonstration der entstandenen Software
 - › Die Präsentation und gesamte Projektabgabe muss dem jeweiligen Betreuer mindestens eine Woche vor dem Termin vorliegen.

- › Die Abschlusspräsentation bietet keine Möglichkeit zum Halten von Seminarvorträgen

- › Projektabgabe beinhaltet:
 - › Ausführliche Dokumentation des Projekts (max. 25 Seiten)
 - › Kurzdokumentation zur entstandenen Software
 - › Dient zur Weiterverwendung der entstandenen SW
 - › Quellcode

Organisatorisches - Sonstiges

- › Hinweise auf Homepage beachten
 - › Schreiben wissenschaftlicher Arbeiten
 - › Halten von Vorträgen (Inhalt, Vortrag und Folien)

- › Endnote setzt sich zusammen aus
 - › Allen Präsentationen (daher die Hinweise beachten!)
 - › Ausarbeitung und Einhaltung des Projektplans
 - › Abgabe des Projekts

Themen

› Themenblock 1:

- › Verarbeitung großer Datenmengen und Data Mining Algorithmen auf Datenströmen
- › Ansprechpartner: Artur Andrzejak

› Themenblock 2:

- › Navigationssystem mit OpenStreetMap-Daten
 - › Drei verschiedene Teilprojekte aus dem Projekt *mapsforge*
 - › Ausführliche Beschreibung der Teilprojekte auch auf der Veranstaltungsseite unter *Detillierte Aufgabenstellung (Projekte 2.1-2.3)*
 - › Anwesenheitspflicht zu den Gruppentreffen jeden Freitag um 10:00 Uhr
- › Ansprechpartner: Jürgen Broß und Thilo Mühlberg

Themen – Themenblock 2

- › Das Projekt *mapsforge*:
 - › Ziel: Entwicklung von freien Bibliotheken und Diensten für Navigation, Routing, Indexierung, Suche, usw. auf Basis von OpenStreetMap
 - › War bereits Thema des vorherigen Projektseminars
 - › Ein Schwerpunkt sind mobile Geräte
 - › Derzeit arbeiten ca. 8 Studenten an verschiedenen Teilprojekten
 - › Möglichkeiten für Abschlussarbeiten
 - › Siehe: <http://www.inf.fu-berlin.de/deprecated/inst/ag-db/osm/>

Themen – Projekt 2.1

- › Entwicklung eines Karten-Overlay für die Android-Plattform
 - › Jetzige Kartenanwendung arbeitet offline:
 - › Android-Applikation mit integrierter Datenbank und Kartenrendering
 - › Tile-basierte Slippy-Map (wie in einem Browser)
 - › Schnell dank spezieller Datenformate, Algorithmen und Caching
 - › Noch keine Unterstützung für das Einzeichnen zusätzlicher Punkte/Wege
 - › Webservice für Routenberechnung existiert bereits
 - › Adapter zum Aufrufen dieses Webservices ebenfalls
 - › Noch keine ausgearbeitete Overlay-Architektur
 - › Keine entsprechende API definiert
 - › Erwünscht: gleiche Schnittstelle wie *Google Maps for Android*
 - › Lösung muss in vorhandene Anwendung integriert werden
 - › Performance kritisch (40 FPS)
 - › Nur sehr wenig RAM (maximal 5 MB)
 - › Am Ende soll ein funktionstüchtiger Prototyp stehen

- › Zur Einarbeitung ein einfaches Projekt mit Google Maps

Themen – Projekt 2.2

- › Unterstützung von 3D-Karten auf der Android-Plattform
 - › Jetzige Karte nutzt die normale Android-2D-Grafikbibliothek
 - › Keine Hardwarebeschleunigung, keine 3D-Unterstützung
 - › Beschränkungen hinsichtlich der Funktionen
 - › High-Level-Schnittstelle
 - › Alternativ dazu vorhanden: OpenGL ES 1.0
 - › Vereinfachte Version von OpenGL 1.3 für eingebettete Systeme
 - › Hardwarebeschleunigung und 3D möglich
 - › Low-Level-Schnittstelle
 - › Portierung des Karten-Renderings auf OpenGL ist kompliziert
 - › Keine Polygone, nur Dreiecke
 - › Funktionen müssen ggf. selber implementiert werden
 - › Mathematische Aufgaben wie Clipping und Triangulierung
 - › Viele Probleme aus dem Gebiet der Computergrafik

- › Das Ergebnis soll mit dem Ausgangsprodukt verglichen werden
 - › Wichtig sind gesammelte Erfahrungen und gewonnene Erkenntnisse

Themen – Projekt 2.3

- › Portierung einer Kartenanwendung auf die iPhone-Plattform
 - › Jetzige Kartenanwendung nur für Android (siehe Projekt 2.1)
 - › *mapsforge* richtet sich auch an andere Betriebssysteme
 - › Portierung des Kartenrenderers notwendig
 - › Möglichkeiten der iPhone-Plattformen sind nicht genau bekannt
 - › Eigenheiten erfordern ggf. Änderungen an der Architektur
 - › Binärformate und Zeichenalgorithmen sollen möglichst gleich bleiben
 - › Einsatz eines Crosscompilers oder automatisierten Tools soll untersucht werden

- › Für den Anfang sind nicht alle Funktionen der Kartenanwendung wichtig
 - › Erfahrungen und Erkenntnisse stehen im Vordergrund
 - › Vergleich von Android und iPhone bezüglich der Kartenanwendung ist interessant
 - › Pixelidentisches Rendering ist sekundär



Viel Spaß im Projektseminar!