Text Andre für Präsentation

F1

* In kommenden 60 Minuten über **2 Dinge**
  + Konzept AJAX, „Async…“ zur asynchronen Datenübertragung vorstellen
  + Prototype.js, das ehemals populärste **JS-Bibliothek** für Ajax Funktionalität
* Am Ende eigene Anwendung vorstellen, die wir auf Basis Ajax und Prototype.js entwickelt haben

F2

* (Relevanz von Ajax sollte jedem bekannt sein. Man braucht sich nur das Blackboard ansehen, wie viele Stellenanzeigen dort von Webanwendungen handeln)
* Um besser über Motivation von Ajax reden zu können, haben wir uns **Anwendungsbeispiel ausgedacht**
* **Hineinversetzen in Bob**, Sekretär, bekommt von Chef jeden Tag Belege auf den Tisch mit Ausgaben
* Anwendungsfall: Am Monatsende Belege einpflegen, mehrere Einträge schnell hintereinander einfügen
* **Firma von Bob verwendet seit kurzem Google Docs**, und Bob ist sehr zufrieden damit, **hat gar keinen Unterschied zu Excel bemerkt, als er umgestiegen ist**.
* Wir werden heute sehen: **1) Warum ist Firma von Bob auf eine Webanwendung umgestiegen** **2) wie hat Ajax dazu beigetragen**

F3

* Einführung zu Ajax, was ist Motivation dahinter, welche Ziele werden damit verfolgt
* Funktionsweise Ajax (**Technologien hinter Ajax**)
* aber auch Nebenwirkungen (typische Fallstricke, Grenzen) von Ajax
* Prototype:
* Rent a bike: Gehen auf **Erfahrungen** ein
* Ausblick: Was wir noch alles gerne besprochen hätten, was den Rahmen sprengt

F4

Fragen generell erlaubt und erwünscht! Auch zu diesen Themen, weil Vorträge ausgefallen sind

F5

* Jetzt: Motivation und Ziele von Ajax
* Ausholen: Jahr 1991, Veröffentlichung WWW durch Tim-Berners Lee WWW /= Internet!...
* Ursprüngliches Ziel: **Forschungsergebnisse einfach austauschen** = Informationsaustausch = Webseiten
* Relativ schnell klar, dass mit Formularen und Logik/Datenbank auf Server Anwendungen möglich sind. **Neue Gattung**
* Unglaublicher Durchbruch und Verbreitung. **Warum?**

F6

Frage: **Welche Vorteile haben Webanwendungen ggü Desktop-Anwendungen**?

* Aktualität: Hat neuste Funktionalität, Fehler werden vom Anbieter schneller korrigiert
* Plattformunabhängigkeit
* …
* Keine Installation = Geringere Nutzungshürde, Webanwendungen können sich schneller verbreiten
* Thin Client: **Wolfram Alpha**

F7

* **Warum ist Bobs Firma nicht schon früher zur Webanwendung gewechselt**?
* Webanwendungen **der 1ten Stunde**, vor Ajax, hatten Nachteile
* Wichtigster Nachteil: **Nach jeder Aktion wird die ganze Seite neu geladen.**
  + Für Bob würde das heißen, er würde einen Beleg abtippen, Enter klicken, warten.
* Temporär nicht bedienbar
  + **Vorstellung Excel**: Jedesmal wenn Eintrag hinzugefügt wird, wird Datei geschlossen und wieder neu geöffnet. **Verrückt! Aber das war damals die Wahrheit von Webanwendungen**.
* Zusammenfassend kann man sagen: …

F8

* Daher wurden neue Technologien eingeführt!
* JavaScript
  + Webanwendung jetzt auch im Browser ausgeführt
  + Jetzt möglich, **dynamisch Seiteninhalte zu verändern, ohne Seite neu zu laden**
* XMLHttpRequest
* Ajax
  + Begriff ist erstmals 2005 gefallen, Ajax hat es davor aber defakto bereits gegeben
* Übergang: Schauen wir uns an, ob und wie Ajax die Probleme von traditionellen Webanwendungen (vorhin identifiziert) löst…

F9

* Latenz:
  + 1) schnelle Anschlüsse (DSL)
  + 2) Beispiel Excel aufgreifen: Jetzt möglich, mehrere Zeilen hinzuzufügen, ohne warten zu müssen bis bei Server angekommen.
* Performance:
  + 1) Browser sehr optimiert
  + 2) Chrome Native Client, WebGL, Adobe AIR, etc.

F10

* Ajax = Konzept, auf das wir gleich noch näher eingehen, das bestehende Technologien kombiniert
* Diskrepanz jetzt wesentlich geringer
* Vorteile von Webanwendungen kommen jetzt voll zur Geltung. schnellste Verbreitung einer Technologie in der gesamten Menschheitsgeschichte
* ungeahntes Wachstum/Popularität, schnellste Verbreitung einer Technologie in der gesamten Menschheitsgeschichte.

F11

Bisher: Warum Ajax wichtig ist, **welche Probleme** es löst

Jetzt: Detail, **wie es Probleme löst**

F12

* …
* 1.
* 2.
* 3. Trennung von Logik und Oberfläche

F13

* Abstrakte Darstellung, Stand 2005, Kern immer noch aktuell
* Erklären: Links, rechts. Unterschied
* Entscheidende Stelle (einziger Unterschied) ist bei „**JavaScript call**“, weil hier wird Anfrage jetzt **nicht direkt an Server geschickt** (Neuladen der Seite), sondern im Hintergrund, nebenläufig, behandelt. Die Webanwendung läuft also zum Teil lokal.

F14

* Jetzt vergleichen wir das aus zeitlicher Sicht und schauen uns zuerst klassische Webanwendung an
* Für Bob ist das verlorene Zeit, er könnte kontinuierlich tippen

F15

**User Activity jetzt durchgezogen**, Bob kann ununterbrochen tippen

F16

* Im Detail die einzelnen Komponenten an, aus denen Ajax heute besteht
* JavaScript und DOM erklären wir nicht weiter
* XMLHttpRequest: Name irreführend. API für senden von HTTP(s) Anfragen aus JavaScript
* Erfunden Microsoft: Wollte Outlook als Webanwendung anbieten