

# Webshop

auf Basis des



# Frameworks

# Features Shop-Frontend

- Statische mehrsprachig erfassbare Seiten
  - Z.B. für Impressum, etc.
- Artikel-Kollektion
  - z.B Praline, Marmeladen
- Anzeige einzelner Artikel mit Bild
- Warenkorb
  - Laden/Speichern per AJAX via REST-API
- Unterstützung für beliebig viele Sprachen (zur Zeit Deutsch / English effektiv aktiviert)

# Features Shop-Frontend

- Registrierung inklusive Verification-Email
  - E-Mail wird in Demo einfach in Konsole ausgegeben, da kein SMTP hinterlegt in Config.
- Login mit Passwort-Vergessen Funktion
- Registrierung/Login ebenfalls über Oauth/Oauth 2.0 möglich
  - Derzeit ist bei Google und Facebook ein Client-Token gelöst für localhost-URL

# Features Shop-Frontend

- Bestellung des Warenkorbes auf Rechnung
  - Ort-Eingabe wird mit TypeAhead unterstützt
  - Bestellung als PDF abrufbar (ebenfalls in der Benutzersprache)
- Bestellverlauf-Anzeige

# Features Backend

- Anzeige eines Dashboard
  - Mit den neusten 5 registrierten Benutzer-Adressen
  - Mit den neusten 5 Bestellungen
- Adress-Modul
  - Auflisten der Adressen
  - Erstellen/Editieren/Kopieren/Löschen einer Adresse
- Artikel-Modul
  - Wie Adress-Modul

# Features Backend

- Backend ist mit Durandal als OnePage-Applikation umgesetzt
  - URL: <http://localhost:9000/backend>
  - Zugriff nur durch Administrator-Benutzer möglich
    - Z.b. [nicole.berger@nb-pralinen.ch](mailto:nicole.berger@nb-pralinen.ch) mit PW praline
- Laden/Speichern aller Daten erfolgt vollständig per AJAX via der eigenen REST-API  
<http://localhost:9000/api>

# Technologie-Übersicht clientseitig

- Bootstrap CSS Framework + Javascript Helpers
  - Für einfachere Implementation eines attraktiven, responsive Designs
  - <http://getbootstrap.com>

# Technologie-Übersicht clientseitig

- JQuery
  - Vereinfachte DOM-Manipulationen
  - Einheitliche, browser-unabhängige AJAX-Requests
  - Diverse GUI-Features wie Tabpages, Modal-Dialoge, etc.
  - <http://jquery.com/>



# Technologie-Übersicht clientseitig

- **Knockout Javascript Framework**
  - Vereinfacht die Datenhandhabung
  - Im Hintergrund sind Daten in Javascript Variablen als Objekt und/oder Listenstrukturen abgebildet
  - Auf HTML Elemente kann eine data-bind auf ein Datenelement der Hintergrunddaten gelegt werden. Sobald im Hintergrund der Inhalt der Variable gewechselt wird, werden alle GUI-Elemente mitaktualisiert, die Daten der entsprechenden Variable anzeigen
  - Weitere Beschreibung und Download siehe <http://knockoutjs.com>

# Technologie-Übersicht clientseitig

- RequireJS
  - Dynamisches Nachladen benötigter JavaScript-Dateien
  - Dependency-Management, um sicherzustellen, dass Ladereihenfolge der Javascript-Dateien korrekt ist
  - Dank RequireJS ist im effektiven HTML-Dokument nur noch ein Script-Tag nötig: Den auf RequireJS.js
  - <http://requirejs.org/>

# Technologie-Übersicht clientseitig

- Durandal
  - Ermöglicht es, OnePage-Websites zu erstellen
  - Einzelne Seiten werden mittels Hashtag in der URL referenziert, z.B:  
`http://localhost:9000/backend#adressen`
  - Ist ein Zusammenzug von JQuery, Knockout, RequireJS und Teilen von Backbone
  - <http://durandaljs.com/>

# Technologie-Übersicht clientseitig

- TypeAhead
  - TypeAhead Input-Feld von den Twitter Jungs
  - Demo siehe <http://twitter.github.io/typeahead.js/>

# Technologie-Übersicht **serverseitig**

- Play Framework (Java)
- Der Framework besteht intern aus:
  - Jetty HTTP-Server
  - Ebean ORM → JDBC → MySQL
  - SBT-Buildsystem (à la Maven)
  - Jackson (Json Parser/Builder)
- Aufbau:  
<http://www.playframework.com/documentation/2.2.0/Anatomy>

# Technologie-Übersicht **serverseitig**

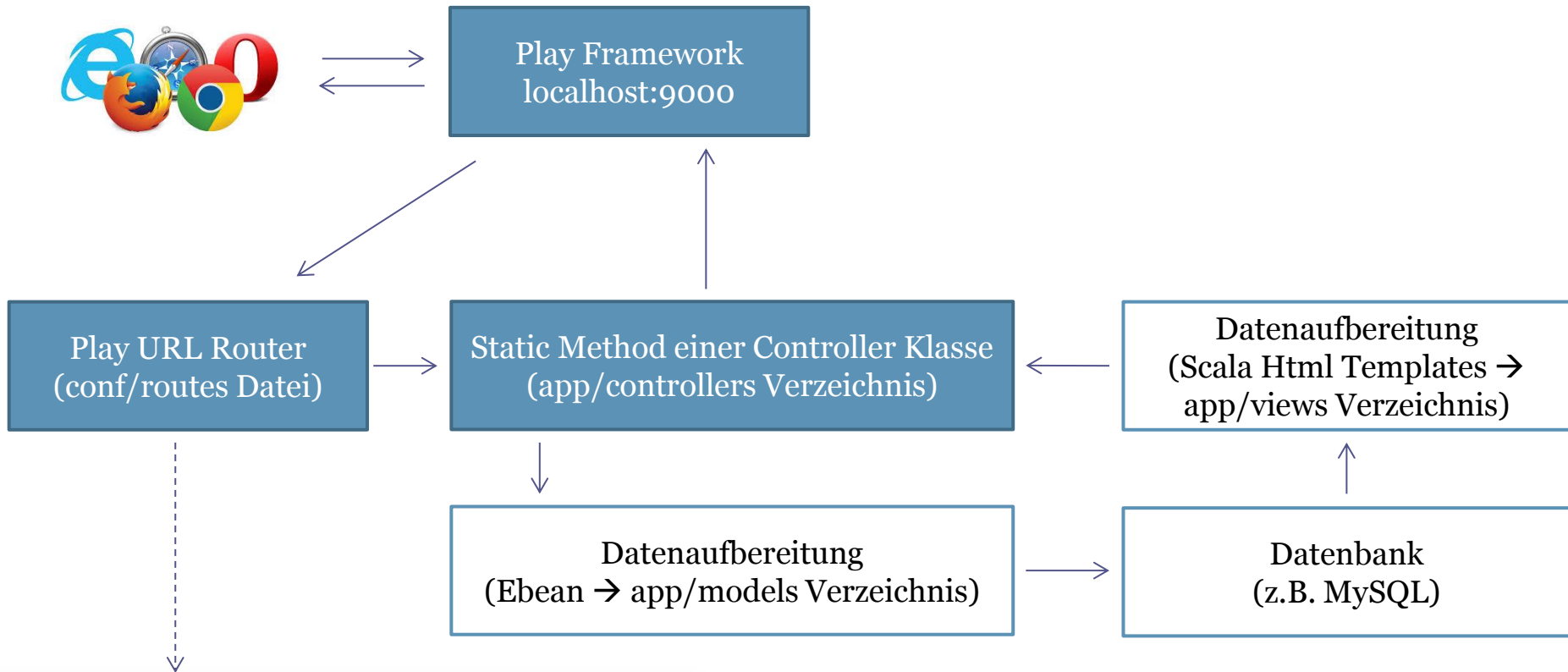
- Play Framework (Java) ist durch Plugins erweiterbar. Zum Beispiel:
  - Deadbold
  - Play Authenticate
- Play Framework unterstützt diverse Entwicklungsumgebungen wie Eclipse, Netbeans, IntelliJ, ...
- JetBrains's IntelliJ kam in diesem Projekt zum Einsatz und ist sehr empfehlenswert

# Play Framework - Highlights



- Vereinheitlichte Fehlerausgabe
- Typensicherheit, keine Var-Typen wie bei PHP
- Reaktive Event-gesteuerte Verarbeitung
- Debugging
- Open Source → Apache2 Lizenz
- Viele klassische Java-Libraries verwendbar
- URL Routing anhand einer Routes-Datei

# Play Framework - Typischer HTTP Request

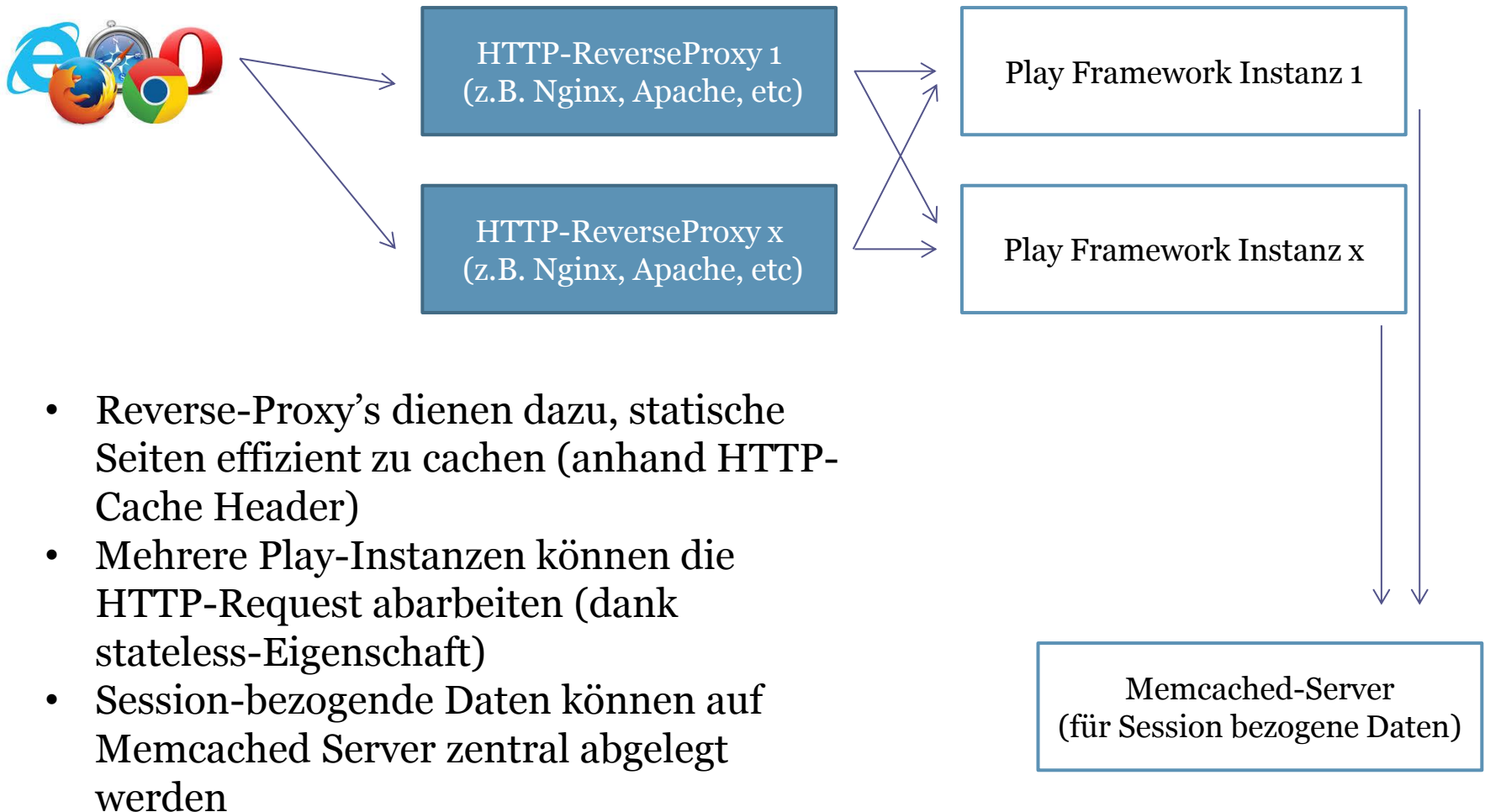


```
# The home page
GET    /                controllers.Projects.index

# Authentication
GET    /login           controllers.Application.login
POST   /login           controllers.Application.authenticate
GET    /logout          controllers.Application.logout
```



# Play Framework - Scale out Deployment



# Play Framework - Praxis Lessons Learnt

- Angenehm schnelle Datenverarbeitung
- Java ist eine sichere, stabile Wahl
  - Bis auf die Tatsache, dass während des Wartens auf einen kompilierten Build wohl mal einer sterben wird
- Viele Play-Framework-Versionen mit inkompatibler API
- Viele Bibliotheken / Plugins einbindbar

# Play Framework - Links

- Für Download, Installations-Guide und Dokumentation:
- <http://www.playframework.com/>
- Plugins:
- <http://www.playframework.com/documentation/2.0.x/Modules>