

A high-contrast, artistic photograph of a railway track curving through a field. The sky is a mix of deep blue and bright orange from a low sun, creating a dramatic, almost surreal atmosphere. The track itself is a strong geometric element, leading the eye from the foreground into the distance. The text is centered over the upper half of the image, in a bold, black, sans-serif font.

# **Rails Workshop #1**

## **Whirlwind Tour**

# Abstract

In unserem ersten Rails-Workshop werden wir eine "Whirlwind Tour" durch Rails starten. Dabei werden wir über die Grundstruktur von Rails sprechen und diverse Bereiche näher beleuchten. Das Ziel ist es, Euch einen guten Einstieg in das Framework zu bieten. Wichtig ist dabei, dass Ihr das Framework nutzt. Am Ende werden wir uns eine Möglichkeit ansehen, wie man einfach mal eine Rails Applikation erstellen kann.

# Inhalt

- Die Rails Grundstruktur
- MVC - wie funktioniert das in Rails?
- Die asset pipeline
- Die Routes
- Das *lib* Verzeichnis
- Gemfile und Gemfile.lock (Rubygems)
- Models und ActiveRecord (AR)
- Models und Validations
- Models und CouchDB
- Das spec Verzeichnis (Test mit Rspec)
  - FactoryGirl
- Hands on: Getting Started Tutorial

# Die Rails Grundstruktur

```
|— app
|— bin
|— config
|— coverage
|— db
|— doc
|— lib
|— log
|— public
|— spec
|— tmp
|— vendor
|— Capfile
|— Gemfile
|— Gemfile.lock
|— Rakefile
|— config.ru
```

# MVC - wie funktioniert das in Rails?

```
| app
|   |
|   |— admin
|   |— assets
|   |— controllers
|   |— decorators
|   |— helpers
|   |— mailers
|   |— models
|   |— views
```

# Rails MVC

Model - ActiveRecord ORM <sup>[1]</sup>

View - ActionView () <sup>[2]</sup>

Controller - ActionPack () <sup>[3]</sup>

[1] <https://github.com/rails/rails/tree/master/activerecord>

[2] <https://github.com/rails/rails/tree/master/actionview>

[3] <https://github.com/rails/rails/tree/master/actionpack>

# ActiveRecord

“Active Record verbindet **Klassen** zu relationalen **Datenbanktabellen**, um fast ohne Konfiguration eine **Persistenz-Schicht** für Anwendungen zu bieten. Die Bibliothek stellt eine **Basisklasse** zur Verfügung von der geerbt werden kann. Die **vererbte Klasse** stellt eine **Zuordnung** zu einer vorhandene **Tabelle** in der **Datenbank** dar. Diese Klassen werden gewöhnlich als **Modelle** bezeichnet. Modelle können auch mit anderen Modellen **verknüpft** werden; dies geschieht durch die Definition von **Associations**.”

# Model Beispiel

```
class Contact < ActiveRecord::Base
  belongs_to :player
  has_many :contact_replies

  validates :email, presence: true, email_address: true
  validates :salutation, inclusion: { in: I18n.t('options_for_salutation_select'),
    allow_blank: true }
  validates :last_name, :message, presence: true

  scope :count_not_responded, -> { where(status: 1).count }
  scope :not_responded, -> { where(status: 1) }
  scope :responded, -> { where(status: 2) }
  scope :spam, -> { where(status: 3) }
end
```



# ActionView

“Action View ist ein Framework für das Auffinden und Rendern von **View-Templates** und bietet **Helper** für den Aufbau von HTML-Formularen, Atom-Feeds und vielem mehr. Template-Formate, die Action View verarbeitet sind **ERB** (embedded Ruby, in der Regel verwendet, um kurze Ruby-Snippets in HTML auszuführen) und **XML-Builder**.”

# View Beispiel (HAML !)

```
%article.std.reg.ko-reg
```

```
  %h1= t('.headline')
```

```
  .stepper
```

```
    .step.active 1
```

```
    .step 2
```

```
    .step 3
```

```
  %p= t('.intro_html')
```

```
= form_for @player, url: registration_index_path, method: :post, html: {class: 'form', id: 'player_registration'} do |f|  
  = render partial: 'form', locals: {f: f}
```

# ActionPack

“Action Pack ist ein Framework um Request zu verarbeiten und zu beantworten. Es bietet Mechanismen für das **Routing** (Request-URLs einer Action zuordnen), definiert **Controller** die Actions beinhalten und erzeugt Antworten durch das Rendern von **Views**, die Templates in verschiedenen Formaten sind. Kurz gesagt bietet ***Action Pack*** die **View-** und **Controller** Schichten im **MVC-Paradigma**.”

# Controller Beispiel

```
class RegistrationController < ApplicationController
  def index
    end

  def new
    build_player
  end

  def edit
    @player = Player.find(params[:id])
  end

  def create
    build_player
    if @player.save
      if @player.email_validation?
        redirect_to action: 'show', id: @player.id
      else
        redirect_to edit_schufa_data_check_path(@player)
      end
    end
  end
end
```

```
    else
      render 'new'
    end
  end

  def show
  end

  private

  def build_player
    @player ||= Player.new
    @player.player_payment = PlayerPayment.new
    @player.attributes = params_for_player
  end
end
```

# Der Router

“Der Rails-Router erkennt **URLs** und versendet diese zur **Action** eines **Controllers**. Er kann weiterhin **Pfade** und **URLs** erzeugen um zu vermeiden, dass diese von Hand geschrieben werden (müssen).”

# Routes Beispiel (config/routes.rb)

```
Rails.application.routes.draw do
  get '/spiele/lotto-6aus49', to: 'lotto#ticket'
  get 'registrierung', to: redirect('/registrierung/neu')
  scope(path_names: {
    new: I18n.t('registration.new.new'),
    edit: I18n.t('registration.new.edit')}) do
    resources :registration, path: 'registrierung'
  end
  root 'homepage#index'
end
```

# Die Asset Pipeline

app/assets

├─ images

├─ javascripts

└─ stylesheets

<https://github.com/rails/sprockets-rails>

# app/assets/javascripts/application.js

```
//= require jquery
//= require parsley-config
//= require parsley
//= require parsley-i18n-de
//= require knockoutjs-3.0.0
//= require jquery_ujs
//= require app
//= require_tree ./lib
//= require_tree ./helper
//= require_tree ./model
//= require_tree ./widget
//= require sample
```



# app/assest/stylesheets/application.css.scss

```
/*  
  *= require_self  
  */  
  
@import "compass";  
@import "rgbapng";  
@import "svgsprite";  
@import "plugins/_helper";  
@import "plugins/_reset";  
@import "main";
```

# config/environments/development.rb

```
Rails.application.configure do
  # Settings specified here will take precedence over those in config/application.rb.

  [...]

  # Debug mode disables concatenation and preprocessing of assets.
  # This option may cause significant delays in view rendering with a large
  # number of complex assets.
  config.assets.debug = true

  # don't compress the assets
  config.assets.compress = false

end
```

# config/environments/production.rb

```
Rails.application.configure do
  # Settings specified here will take precedence over those in config/application.rb.
  [...]
  # Disable Rails's static asset server (Apache or nginx will already do this).
  config.serve_static_assets = false
  # compress the assets
  config.assets.compress = true
  # Compress JavaScripts and CSS.
  config.assets.js_compressor = :uglifier
  config.assets.css_compressor = :sass
  # Do not fallback to assets pipeline if a precompiled asset is missed.
  config.assets.compile = false
  # Generate digests for assets URLs.
  config.assets.digest = true
end
```

# Das *lib* Verzeichnis

```
lib
├── assets
├── draw_service
├── importer
├── schufa
├── tasks
├── validators
├── ace_redis.rb
├── ckonto.rb
├── contacts_service.rb
├── date_handler.rb
├── draw_service.rb
├── entity_encoder.rb
├── lotteries.rb
└── schufa.rb
```

# Gemfile & Gemfile.lock (Rubygems)

Im **Gemfile** werden alle benötigten Rubygems angegeben, die für die Applikation benötigt werden. Der **bundler** installiert die Rubygems und löst Dependencies auf.

Im **Gemfile.lock** werden alle Versionen der Rubygems vorgehalten. Daraus werden die gems mit **bundle install** in der entsprechenden Version installiert.

# Gemfile

```
source 'https://rubygems.org'

gem 'rails', '4.1.5'
gem 'mysql2'
gem 'jquery-rails'
gem 'parsley-rails' # https://github.com/mekishizufu/parsley-rails
gem 'bcrypt', '~> 3.1.7' # Use ActiveRecord has_secure_password

group :development do
  gem 'spring'
  gem 'capistrano-rails'
  gem 'capistrano-rvm'
end
```

# Gemfile.lock

## GIT

```
remote: gitlab@gitlab.sumcumo.net:gems/forminator.git  
revision: 0a0f7de1c1f26089e9b164bd665a4a5da9012dc8  
specs:  
  forminator (0.5.2)
```

## GEM

```
remote: https://rubygems.org/  
specs:  
  aasm (3.3.3)  
  actionmailer (4.1.5)  
    actionpack (= 4.1.5)  
    actionview (= 4.1.5)  
    mail (~> 2.5.4)
```

# Models und ActiveRecord

```
params_for_player = params[:player].permit(  
  :first_name, :last_name, :street, :house_number, :zip_code, :city, :date_of_birth  
) : {}  
  
@player ||= Player.new  
@player.attributes = params_for_player  
  
@player.save  
  
@player.update(params_for_player)  
  
@player = Player.find(session[:player_id])  
  
Player.select("id, last_name").where(status: 'pending').order('last_name DESC')
```



# Models und Validations

```
class Player < ActiveRecord::Base

  validates :email, presence: true, email_address: true, on: :create
  validates :zip_code, presence: true, length: { is: 5 }
  validates :date_of_birth, age: true
  validates :zip_code, permitted_state: true
  validates :city, zip_code_city_combination: true
  validates_format_of :mobile, with: /\A01[5-7][0-9]\d{4,}\Z/i

end
```

# PermittedStateValidator

(lib/validators/permitted\_state\_validator.rb)

```
class PermittedStateValidator < ActiveRecord::EachValidator
  def validate_each(record, attribute, zip_code)
    unless is_permitted_state?(zip_code)
      record.errors.add(attribute, (options[:message] || :permitted_state))
    end
  end

  private

  def is_permitted_state?(zip_code)
    address_data = AddressLookup.find_by_zip_code(zip_code)
    return false if address_data.blank?
    is_permitted_state_code?(address_data)
  end

  def is_permitted_state_code?(address_data)
    APP_CONFIG[:permitted_state_codes].include?(address_data.first[:state_code])
  end
end
```

# Models und CouchDB

```
class AddressLookup
  include CouchPotato::Persistence

  property :state_code, type: String
  property :state, type: String
  property :zip_code, type: String
  property :city, type: String

  view :all, key: [:state_code, :state, :zip_code, :city]
  view :by_state_code, key: :state_code
  view :by_state, key: :state
  view :by_zip_code, key: :zip_code
  view :by_city, key: :city
  view :by_zip_code_and_city, key: [:zip_code, :city]

  def self.all_addresses
    CouchPotato.database.view(self.all)
  end
end
```

# couch\_potato

[https://github.com/langalex/couch\\_potato](https://github.com/langalex/couch_potato)

# CouchDB Futon

[http://localhost:5984/\\_utils](http://localhost:5984/_utils)

Rails Workshop #1 - Google Apache CouchDB - Futon: x

localhost:5984/\_utils/database.html?ace\_development

Overview > ace\_development

New Document

Security...

Compact & Cleanup...

Delete Database...

Jump to:

View 

All documents

Design documents

Temporary view...

address\_lookup\_view\_by\_zip\_code

by\_zip\_code

address\_lookup\_view\_by\_zip\_code\_and\_city

by\_zip\_code\_and\_city

draw\_view\_bingo

bingo

draw\_view\_ej

ej

draw\_view\_gs

gs

draw\_view\_keno

keno

draw\_view\_lotto

lotto

draw\_view\_spiel77

spiel77

draw\_view\_super6

super6


Key ▲	Value
"5b8d13a1729610b2c08f4ea2fb3e468e" ID: 5b8d13a1729610b2c08f4ea2fb3e468e	{rev: "1-f04dd"
"5b8d13a1729610b2c08f4ea2fb3e47cb" ID: 5b8d13a1729610b2c08f4ea2fb3e47cb	{rev: "1-d33d3"
"5b8d13a1729610b2c08f4ea2fb3e54c7" ID: 5b8d13a1729610b2c08f4ea2fb3e54c7	{rev: "1-c30a0"
"5b8d13a1729610b2c08f4ea2fb3e55f2" ID: 5b8d13a1729610b2c08f4ea2fb3e55f2	{rev: "1-2bddd"
"5b8d13a1729610b2c08f4ea2fb3e5712" ID: 5b8d13a1729610b2c08f4ea2fb3e5712	{rev: "1-50a3a"
"5b8d13a1729610b2c08f4ea2fb3e57d0" ID: 5b8d13a1729610b2c08f4ea2fb3e57d0	{rev: "1-118ed"
"5b8d13a1729610b2c08f4ea2fb3e6366" ID: 5b8d13a1729610b2c08f4ea2fb3e6366	{rev: "1-1b5b1"
"5b8d13a1729610b2c08f4ea2fb3e6eb8" ID: 5b8d13a1729610b2c08f4ea2fb3e6eb8	{rev: "1-b08e6cc832e6e243911ab6a164b1b60a"}
"5b8d13a1729610b2c08f4ea2fb3e7935" ID: 5b8d13a1729610b2c08f4ea2fb3e7935	{rev: "1-ca6037ba075a43243321b593a4f67c58"}
"5b8d13a1729610b2c08f4ea2fb3e7f84" ID: 5b8d13a1729610b2c08f4ea2fb3e7f84	{rev: "1-db2412fdcae37da8da22ad39291879fb"}

Showing 1-10 of 1383 rows

← Previous Page

Rows per page:

Next Page →



# CouchDB

relax

Tools

Overview

Configuration

Replicator

Status

Documentation

Manual

Diagnostics

Verify Installation

Recent Databases

ace\_development

ace\_test

test

Welcome to Admin Party!

Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.6.0

Overview > ace\_development

+ New Document

Security...

Jump to:

View:

Stale views ☐

Compact & Cleanup...

Delete Database...

View Code

Map Function:

```
function(doc) {  
  if(doc.ruby_class && doc.ruby_class == 'AddressLookup') {  
    emit(doc['zip_code'], 1);  
  }  
}
```

Reduce Function (optional):

```
_sum
```

Run

Language:

=

Revert

Save As...

Save

Key ▲

Grouping:

Value

☒ Reduce

"30159"	1
"30161"	1
"30163"	1
"30823"	1
"30851"	1
"30880"	1
"30900"	1
"30916"	1
"30938"	1
"30966"	1


Showing 1-10 of unknown rows

← Previous Page

Rows per page:

Next Page →

View request duration: 00:00:00.032



CouchDB

relax

Tools

Overview

Configuration

Replicator

Status

Documentation

Manual

Diagnostics

Verify Installation

Recent Databases

ace\_development

ace\_test

test

Welcome to Admin Party!  
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.6.0

# Testen

**RSpec FTW!**



# Das *spec* Verzeichnis (RSpec)

## **spec**

- | controllers
- | decorators
- | factories
- | lib
- | mailers
- | models
- | rails\_helper.rb
- | spec\_helper.rb
- | support

# **RSpec Konfiguration**

`spec/rails_helper.rb`

`spec/spec_helper.rb`

# spec/models/address\_lookup\_spec.rb

```
require 'rails_helper'

RSpec.describe AddressLookup, :type => :model do
  before :each do
    WebMock.disable_net_connect!(:allow_localhost => true)
  end

  it 'exists' do
    expect(described_class).to equal(AddressLookup)
  end

  it 'has methods' do
    expect(AddressLookup.respond_to?(:all_addresses)).to be_truthy
    expect(AddressLookup.respond_to?(:find_by_zip_code)).to be_truthy
    expect(AddressLookup.respond_to?(:find_by_city)).to be_truthy
    expect(AddressLookup.respond_to?(:find_by_state)).to be_truthy
    expect(AddressLookup.respond_to?(:find_by_state_code)).to be_truthy
  end

  describe :lookup do
    before :each do
      @doc = create(:address_lookup)
    end

    after :each do
      CouchPotato.database.destroy @doc
    end

    it 'has a positiv result' do
      expect(AddressLookup.find_by_zip_code('33333')).to be_truthy
    end
  end
end
```

# Factory Girl

[https://github.com/thoughtbot/factory\\_girl](https://github.com/thoughtbot/factory_girl)

# Beispiel Factory Girl (spec/factories/players.rb)

```
FactoryGirl.define do
  factory :player do
    email 'foo@bar.de'
    salutation 'Herr'
    last_name 'Mustermann438850'
    [...]

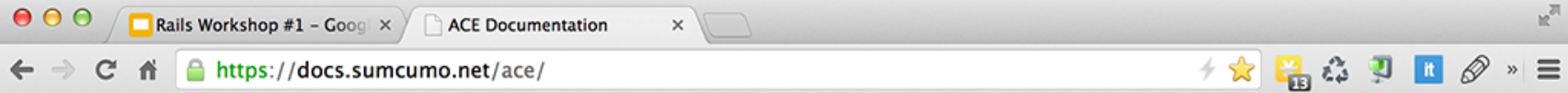
    factory :player_schufa_ok do
      salutation 'Frau'
      last_name 'Bretter'
    end

    factory :player_schufa_not_ok do
      salutation 'Frau'
      last_name 'Gawel'
    end
  end
end
```

# Im Test (spec/models/player\_spec.rb)

```
RSpec.describe Player, :type => :model do
  describe :crud do
    before :each do
      @player = create(:player)
    end

    it 'expects to create a player' do
      expect(@player.persisted?).to be_truthy
    end
  end
end
```



# Ace Documentation

- [Docs](#)
- [Test Results](#)
- [Test Coverage](#)
- [Metrics](#)

## RSpec Code Examples

✓ Passed ✓ Failed ✓ Pending

502 examples, 0 failures, 1 pending  
Finished in 109.73656 seconds

### BingoController

#### GET draws

has a 200 status code	0.29640s
assigns the current date	0.17202s
assigns draws	0.11175s
renders bingo templates and partials	0.18514s

#### GET ticket

assigns draws	0.87040s
assigns stakes	0.81896s
assigns fees	0.37885s
assigns type	0.77311s

### EjController

#### GET draws

has a 200 status code	0.20652s
assigns the current date	0.08378s
assigns draws	0.05569s
renders ej templates and partials	0.05653s

#### GET ticket

assigns draws	0.09264s
assigns stakes	0.07088s



## MetricFu Results

---

**Rails Best Practices**

---

**Roodi**

---

**Reek**

---

**Flay**

---

**Cane**

---

**Stats**

---

**Flog**

---

**Saikuro**

---

**Hotspots**

# Hands on!

[http://guides.rubyonrails.org/getting\\_started.html](http://guides.rubyonrails.org/getting_started.html)

# Questions?

## Ask!

Andy and Moritz are happy to help!

**Dangge :)**