

Projektteam: Benjamin Raison

Dokumentation Teil 2

Resonance

****

# Abstract

In der Zeit von «fake news» ist es wichtiger als je, sich mehrere Quellen zu suchen, um Fakten zu verifizieren.  
Resonance hilft dabei, indem es von diversen Quellen Artikel anzeigt. Somit ist man nicht daran gebunden, all seine Nachrichten von einer Quelle zu beziehen, nur weil es bequem ist.

Inhalt

Abstract 1

Technische Dokumentation 3

Code Style 3

Testauswertung 3

Installationsanleitung 3

Systemanforderungen 3

Installation 4

Benutzerhandbuch 6

Startseite 6

/\* 6

/news 6

/about 6

/contact 6

Mehr? 6

Zeitplanung 0

GANTT Diagramm 0

Fazit 0

# Technische Dokumentation

## Code Style

Standard WebStorm und PHPStorm Einstellungen.  
Es wurde Englisch benutzt.

## Testauswertung

|  |  |
| --- | --- |
| Abschnitt | Inhalt |
| **ID** | FT-01 |
| **Anforderungen** | F-02 |
| **Vorbedingungen** | Der Webserver und die Datenbank läuft |
| **Ablauf** | localhost/pages/view.html#!/news wird im Webbrowser aufgerufen |
| **Erwartetes Resultat** | Im Webbrowser wird ein zufälliger Zeitungsartikel angezeigt |
| **Eigentliches Resultat** | Etwa 1 von 50 Artikeln wirft ERR\_CONNECTION\_REFUSED. Ein solche Fehlermeldung wird gross auf dem Bildschirm angezeigt. |

|  |  |
| --- | --- |
| Abschnitt | Inhalt |
| **ID** | NFT-07 |
| **Anforderungen** |  |
| **Vorbedingungen** | Der Webserver läuft. Die Datenbank läuft nicht |
| **Ablauf** | localhost/pages/view.html#!/news wird im Webbrowser aufgerufen. |
| **Erwartetes Resultat** | Es wird eine Fehlermeldung angezeigt. |
| **Eigentliches Resultat** | Es wird eine leere Seite angezeigt. |

Die anderen Tests sind entweder noch nicht implementiert oder funktionierten Fehlerfrei

## Installationsanleitung

### Systemanforderungen

|  |  |
| --- | --- |
| **Typ** | **Anforderung** |
| Webserver | Apache2 |
| Webserver | PHP7 |
| Datenbank | MySQL |
| Webserver | certbot |
| Speicherplatz | mindestens 500MB |
| Java | Oracle JRE 8 |

### Installation

#### Apache

* Virtueller Host
* Certbot

#### News Downloader

* Als .jar verpacken: <https://github.com/BR-0309/Resonance-Backend>
* In /home/[user]/java speichern
* config.php erstellen
* CRON job einrichten
  + Alle 30 Minuten

#### Webseite

* Gesamten Ordner in das root-Verzeichnis des vorher eingerichteten virtuellen Host kopieren

#### MySQL

* Installieren
* Benutzername & Passwort in config.php eintragen
* SQL-Dateien ausführen
  + Model.sql
  + Data.sql
* Benutzername & Passwort in /home/[user]/java/db.cfg eintragen

# Benutzerhandbuch

## Startseite

Die Startseite gibt einen sehr groben Überblick über die gesamte Website.

## /\*

Auf allen folgenden Seiten wird links in einer «sidebar» eine Navigation angezeigt.  
Sobald die Benutzerverwaltung implementiert ist, wird über der Navigation ein (nicht änderbares) Profilbild angezeigt.

## /news

Diese Seite enthält zufällig ausgewählte Zeitungsartikel. In der oben rechten Ecke können folgende Funktionen ausgelöst werden:

* Nächster Artikel
* Nach Titel suchen
* Die ursprüngliche Seite aufrufen

## /about

Diese Seite enthält grundlegende Information über die Seite und deren Macher

## /contact

Über das Kontaktformular können dem Seiteninhaber Nachrichten hinterlassen werden.

## Mehr?

Zum aktuellen Zeitpunkt sind Einstellungen und Benutzerverwaltung noch nicht implementiert.

# Zeitplanung

## GANTT Diagramm



# Fazit

Das Projekt ist bis jetzt ziemlich gut gelaufen und ich habe viel gelernt. Es war gut, die Chance zu haben, mich mit AngularJS auseinanderzusetzen.   
Ich habe mich, mit einigen technisch bedingten ausnahmen, ziemlich gut an die Planung gehalten. Allerdings habe ich für einige user stories viel zu viel Zeit eingeplant, für andere wiederum zu wenig.