**Konzept**

**TNT**

**für GIBS**

Dokumentname: Konzept TNT

Version: 1.0

Klassifizierung: für internen Gebrauch

Autor: Elias Tonini, Nico Schönbächler, Tim Schurtenberger

Letzte Änderung: 06.11.2018

Verteiler: Herr Daniel Mosimann

**Änderungsübersicht:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Datum** | **Autor** | **Beschreibung der Änderung** | **Betroffene Kapitel** |
| 1.0 | 06.11.2018 | Nico Schönbächler, Elias Tonini, Tim Schurtenberger | Erste Version | 1-6 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Referenzierte Dokumente:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Dokument-bezeichnung** | **Dokumentname** | **Version** | **Kurzbeschreibung** |
| Projektplan |  | 1.1 | Aktueller Projektplan |
| Arbeitsjournal |  | 1.1 | Ständige Reflexion von der Arbeit |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Inhaltsverzeichnis**

[1. Einleitung 4](#_Toc529538932)

[1.1. Inhalt und Zweck des Dokuments 4](#_Toc529538933)

[1.2. Abkürzungen und Definitionen 4](#_Toc529538934)

[2. Varianten 5](#_Toc529538935)

[2.1. Variante 1 5](#_Toc529538936)

[2.1.1. Vorteil 5](#_Toc529538937)

[2.1.2. Nachteil 5](#_Toc529538938)

[2.2. Variante 2 5](#_Toc529538939)

[2.2.1. Vorteil 5](#_Toc529538940)

[2.2.2. Nachteil 5](#_Toc529538941)

[3. Lösungsansatz 6](#_Toc529538942)

[3.1. Begründung der Wahl 6](#_Toc529538943)

[3.2. Systemarchitektur 6](#_Toc529538944)

[3.3. Produkte 6](#_Toc529538945)

[3.4. Feature: Erfahrungspunkte 7](#_Toc529538946)

[3.5. Diverses 7](#_Toc529538947)

[4. Projektstatus 7](#_Toc529538948)

# Einleitung

## Inhalt und Zweck des Dokuments

Dieses Dokument beinhaltet das Konzept von dem Projekt TNT. Mit dem Konzept planen wir die Applikation. Um mögliche Fehler schon im Voraus zu erkennen, muss das Konzept nachvollziehbare Lösungsansätze aufzeigen und unsere Überlegungen und Entscheidungen festhalten.

## Abkürzungen und Definitionen

**Definitionen:**

|  |  |
| --- | --- |
| PHP | Skriptsprache |
| Laravel | Webframework |
| Front-End | Anwendung die auf dem Client läuft |
| Back-End | Anwendung die auf dem Server läuft |
| Fullstack | Beinhaltet Front- und Back-End |
|  |  |
|  |  |

**Abkürzungen:**

|  |  |
| --- | --- |
| HW | Hardware |
| SW | Software |
|  |  |
|  |  |
|  |  |
|  |  |

# Varianten

## Variante 1

Für unser Projekt verwenden wir Laravel. Laravel ist ein Fullstack Webframework basierend auf PHP.

Die Daten werden zentral auf einer Maria-Datenbank gespeichert. Um darauf zugreifen zu können, wird eine Schnittstelle benötigt.

Die grafische Schnittstelle der TNT-Applikation ist das GUI. Das GUI soll desktop-first entwickelt werden. Sollte jedoch responsive sein, sodass die Webapplikation auch auf Mobile bedienbar ist.

### Vorteil

* Einfacheres Deployment
* Existierendes Wissen
* Referenz Projekte

### Nachteil

* Team-Code Arbeit erschwerter
* Nicht alle Teammitglieder sind voll und ganz mit der Materie vertraut.

## Variante 2

Wir verwenden Angular als Frontend Framework und entwickeln ein separates Backend mit Java Spring. Wir verwenden ebenfalls eine Maria-Datenbank, welche im Backend verwendet wird.

### Vorteil

* Existierendes Wissen
* Referenz Projekte

### Nachteil

* Team-Code Arbeit muss zwingend aufgeteilt werden
* Bei Ausfällen einzelner Mitarbeiter ist das Projekt im Verzug
* Nicht alle Teammitglieder sind voll und ganz mit der Materie vertraut.

# Lösungsansatz

## Begründung der Wahl

Wir haben die Vor- und Nachteile ausführlich geprüft und gewichtet. Unser Entscheid ist nun definitiv. Wir werden die Lösungsvariante 1 verwenden. Die Lösungsvariante 1 hat eindeutig mehr Vorteile und weniger Nachteile als die Variante 2.

## Systemarchitektur

Die TNT Webapplikation besteht zwar durch Fullstack, aus einer Applikation, um diese jedoch übersichtlicher darzustellen, nahmen wir die Applikation auseinander und gliedern sie wie folgt auf.

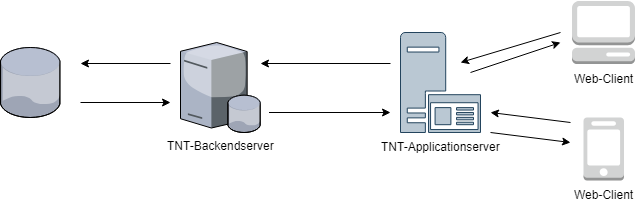


Abbildung 1 Grundstruktur TNT

Der Client greift mit dem Browser per URL auf den TNT-Applicationserver zu. Diese überprüft die Anfrage und tätigt anschließend eine Abfrage auf das Backend der Applikation. Dieses fragt die notwendigen Daten auf der Datenbank ab, und schickt diese dem Front-End. Das Front-End zeigt die Daten anschließend dem Client an.

## Produkte

**Laravel Auth:**  
Wir verwenden Laravel built-in Authentication. Dies ist ein Package von Laravel, welches ein vordefiniertes, vollumfängliches Auth-System im Hintergrund hat.

**Captcha:**  
Um mögliche Bruteforce-Attacken und andere Sicherheitsprobleme vorzukommen, bauen wir Captchas ein. Dies wird beim Login-Screen eingebaut und muss zuerst gelöst werden, dass man sich einloggen kann.

Als Referenz für ein Captcha kann folgendes Bild helfen:

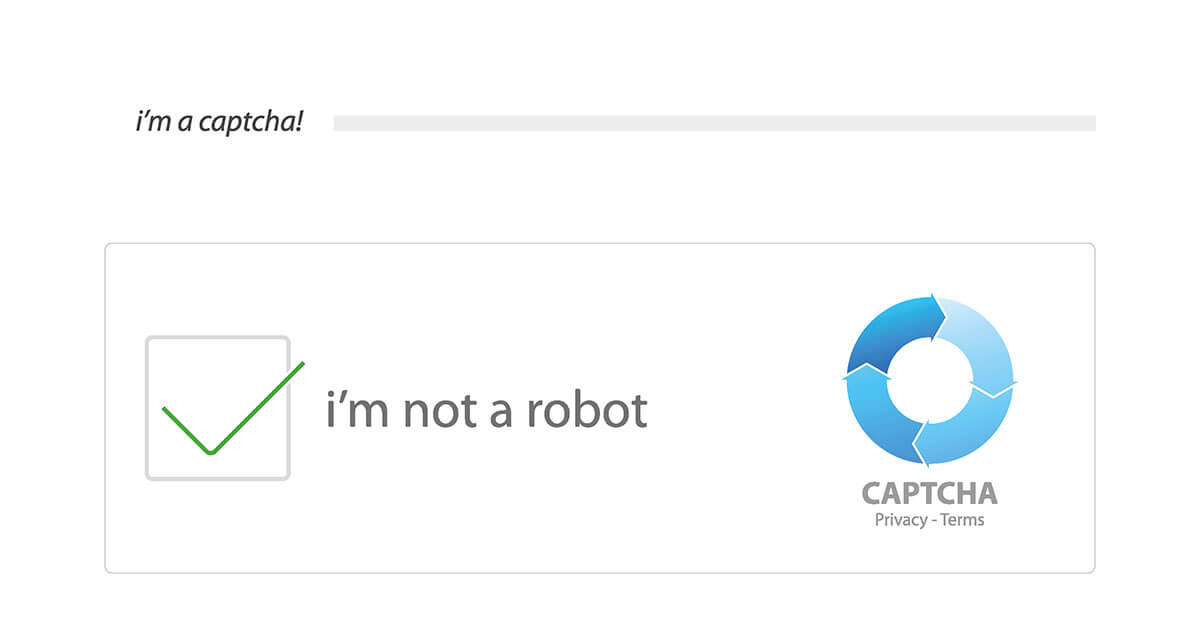


Abbildung 2 Captcha

Quelle:

https://www.1und1.at/digitalguide/fileadmin/DigitalGuide/Teaser/captcha-t.jpg

## Feature: Erfahrungspunkte

Unser Ziel ist es, dass das Forum möglichst viele Antworten und Fragen bekommt. Um die User zu belohnen und einen internen Wettkampf herzustellen, werden wir Erfahrungspunkte einbauen. Die User erhalten für jede Antwort Erfahrungspunkte und können diese im Profil einsehen.

## Diverses

Um den Wettkampf zu erhöhen werden wir eine Liste (Leaderboard) erstellen um eine Übersicht der aktivsten User zu erhalten. Diese wird öffentlich zugänglich sein.

Unsere ganze Applikation ist für den internationalen Gebrauch ausgerichtet und somit wird dies in Englisch umgesetzt.

# Projektstatus

Unser Gesamt Aufwand, einsehbar in dem Arbeitsplan, schätzen wir auf 140h.

Der Stundenansatz liegt bei 80 CHF/Stunde.   
Erforderliches Budget: 140 \* 80 = 11'200 CHF

Sobald wir eine bestätigte Eizahlung (20% des Gesamtsaufwand) auf unserem Konto erhalten, werden wir mit der Umsetzung starten.

Wir haben unser Projektplan bis jetzt gut eingehalten und alle Termine fristgerecht wahrgenommen. Des Weiteren haben wir keine Probleme mit den Ressourcen oder dem Team.