Tatyana Merlo – Natalie Stalder – Nadja Stadelmann

Travelling I

M133 – Webapplikation realisieren

# Einleitung

## Systemanforderungen

## Vorwissen

## Dokument

### Grund

### Aufbau

* Analyse
  + Beschrieb und erste Aufsplittung in einzelne Teilbereiche
* Vorgehen
  + Abhängigkeiten, Aufgabenübersicht
* Design
  + Detaillierte Analyse DB-Design
  + Detaillierte Analyse Programmstruktur
* Implementation
* Testing
  + Testcases basieren auf UseCases

Inhalt

[Einleitung 1](#_Toc494203939)

[Systemanforderungen 1](#_Toc494203940)

[Vorwissen 1](#_Toc494203941)

[Dokument 1](#_Toc494203942)

[Grund 1](#_Toc494203943)

[Aufbau 1](#_Toc494203944)

[Vision 3](#_Toc494203945)

[Analyse 4](#_Toc494203946)

[Anforderungen 4](#_Toc494203947)

[Must 4](#_Toc494203948)

[Nice to have 4](#_Toc494203949)

[Use Cases 4](#_Toc494203950)

[Design 5](#_Toc494203951)

[DB 5](#_Toc494203952)

[Tabellen 5](#_Toc494203953)

[User 5](#_Toc494203954)

[Location 5](#_Toc494203955)

[Post 5](#_Toc494203956)

[Picture 5](#_Toc494203957)

[Visited 5](#_Toc494203958)

[Designanpassungen 5](#_Toc494203959)

[Programm 5](#_Toc494203960)

[Model 5](#_Toc494203961)

[View 5](#_Toc494203962)

[Controller 5](#_Toc494203963)

[Designanpassungen 6](#_Toc494203964)

[Implementation 7](#_Toc494203965)

[Implementierung 7](#_Toc494203966)

[Testing 8](#_Toc494203967)

# Vision

Man bereist die Welt und möchte möglichst viele Orte sehen. Doch sich dabei auf einen Schlag an alle Orte zu erinnern, die besten Fotos an einem zentralen Ort lagern und allenfalls Freunden eine Hotel- oder Restaurantempfehlung liefern – das fällt oftmals schwer.

Für all diese Probleme bringen wir mit dem Travelblog ‘Travelling I’ die Lösung: Auf einer Karte kann eingetragen werden, an welchem Ort man bereits gewesen ist. Das entsprechende Land wird dabei farbig eingefärbt und die Stadt mittels eines Punktes markiert. So erhält man eine Übersicht über seine vergangenen Reisen, die man überall auf der Welt über das Internet abrufen kann!

Zu jedem Ort kann auch ein Blogeintrag geschrieben werden, um tolle Hotels, Restaurants oder Ausflugsziele festzuhalten. Dazu kann man seine schönsten, lustigsten oder emotionalsten Fotos hochladen, um immer wieder in diesen Erinnerungen zu schwelgen.

Damit diese Daten nicht einfach öffentlich preisgegeben werden, muss man sich dazu zuerst registrieren und jeweils einloggen. So werden die Ferienerlebnisse vor unerwünschten Blicken geschützt.

In weiteren Schritten stellen wir uns vor, dass sich die User Restaurants oder vorhandene Aktivitäten an den spezifischen Orten anzeigen lassen können. Da das Wetter für Ausflüge und das Gepäck ebenfalls wichtig ist, wird auch dieses eingebunden.

Um die Suche durch seine Einträge zu erleichtern, werden später Kategorien eingebunden. So kann der User zum Beispiel die Kategorie ‘Hotel’ angeben und später ganz einfach die entsprechend markierten Einträge filtern.

Doch bei den vergangenen Reisen soll es nicht bleiben! Pläne zu schmieden für neue Ferien ist ebenfalls spannend und erfreulich. Damit dies auch locker vom Hocker geht, werden den Usern entsprechende Flüge und Hotels angezeigt, die für sie geeignet sein können.

# Analyse

## Anforderungen

### Must have

1. Login / Registrierung
2. Karte – wo warst du?
   1. Nadel auf Stadt
   2. Blogeintrag mit Fotos

## Nice to have

1. Google Maps einbindung
   1. Restaurants in der jeweiligen Stadt suchen
2. Wetter anzeige
   1. Name der Stadt eingeben 🡪 entsprechendes Wetter angeben
3. Vorhandene Aktivitäten anzeigen lassen
   1. Europaweit / Weltweit
   2. Tripadvisor API?
4. Blogmässige Notizen erweitern z.B. Kategorien
   1. Restaurant
   2. Hotels
   3. Festivals
   4. Einkaufscentern
   5. ….
5. Plan für nächste Reise(n)
   1. gute Flüge und Hotels anzeigen lassen

## Use Cases

# Design

## DB

### Tabellenbeschrieb

#### User

Der User braucht einen Usernamen sowie das Passwort. Der Username ist unique definiert. Das Passwort ist mittels BCrypt verschlüsselt – das heisst, es benötigt eine Länge von CHAR(60). Ausserdem wird der Salt CHAR(60) abgespeichert, er besteht aus einer Zahl.

#### Location

Hier werden die besuchten Orte abgespeichert. Wir speichern den Namen der Stadt, sowie die Längen- und Breitengrade in dieser Tabelle ab, da Google Maps die Nadeln mittels der Koordinaten setzt.

Formate: Längengrad Breitengrad

-000.0000 -00.0000

CHAR(9) CHAR(8)

Index: Unique über Longitude und Latitude.

#### Post

Im Post wird das Datum, der Titel und ein Text eingespeichert. Der Text ist nicht zwingend, damit ein User auch Bilder ohne Text hochladen kann.

#### Picture

Der Bildername, welcher angezeigt wird, wird gespeichert. Weiter wird der Filename gespeichert, damit das passende Bild geladen werden kann.

#### Visited

Zwischentabelle zwischen dem User und der Location. Mehrere User können dieselbe Stadt besuchen und ein User kann mehrere Städte besuchen. Weiter ist hier der Fremdschlüssel zum zugehörigen Post eingespeichert.

### Designanpassungen

## Programm

### Model

### View

### Controller

### Designanpassungen

# Implementation

## Implementierung

# Testing