***Mandantenfähige Website***

Autor: Ruanin Khankah

Datum: 11. Mai 2022

Inhaltsverzeichnis

[Glossar 3](#_Toc103007702)

[1 IPA-Bericht Teil 1 4](#_Toc103007703)

[1.1 Titel der Arbeit 4](#_Toc103007704)

[1.2 Arbeitsbereiche 4](#_Toc103007705)

[1.3 Vorwort 4](#_Toc103007706)

[1.4 Detaillierte Aufgabenstellung 4](#_Toc103007707)

[1.5 Mittel und Methoden 5](#_Toc103007708)

[1.6 Vorkenntnisse 5](#_Toc103007709)

[1.7 Vorarbeiten 5](#_Toc103007710)

[1.8 Neue Lerninhalte 5](#_Toc103007711)

[1.9 Arbeiten in den letzten 6 Monaten 6](#_Toc103007712)

[1.10 Mittel und Methoden 7](#_Toc103007713)

[1.11 Projektorganisation 11](#_Toc103007714)

[1.11.1 Durchführung 11](#_Toc103007715)

[1.11.2 Arbeitsmethode 11](#_Toc103007716)

[1.11.3 Ablage der Daten 12](#_Toc103007717)

[1.12 Zeitplan 13](#_Toc103007718)

[1.12.1 Meilensteine 14](#_Toc103007719)

[1.13 Arbeitsjournal 15](#_Toc103007720)

[1.13.1 Tag 1 – 09.05.2022 15](#_Toc103007721)

[1.13.2 Tag 2 – 10.05.2022 16](#_Toc103007722)

[1.13.3 Tag 3 – 11.05.2022 17](#_Toc103007723)

[1.13.4 Tag 4 – 12.05.2022 18](#_Toc103007724)

[1.13.5 Tag 5 – 16.05.2022 19](#_Toc103007725)

[1.13.6 Tag 6 – 17.2022 20](#_Toc103007726)

[1.13.7 Tag 7 – 18.05.2022 21](#_Toc103007727)

[1.13.8 Tag 8 – 19.05.2022 22](#_Toc103007728)

[1.13.9 Tag 9 – 23.05.2022 23](#_Toc103007729)

[1.13.10 Tag 10 -24.05.2022 24](#_Toc103007730)

[2 IPA Bericht Teil 2 25](#_Toc103007731)

[2.1 Kurzfassung 25](#_Toc103007732)

[2.1.1 Ausgangslage 25](#_Toc103007733)

[2.1.2 Umsetzung 25](#_Toc103007734)

[2.1.3 Ergebnis 25](#_Toc103007735)

[2.2 Informieren 26](#_Toc103007736)

[2.2.1 Anfängliche Unklarheiten 26](#_Toc103007737)

[2.3 Planen 27](#_Toc103007738)

[2.3.1 Realisierungskonzept 27](#_Toc103007739)

[2.4 Entscheiden 28](#_Toc103007740)

[2.5 Realisieren 29](#_Toc103007741)

[2.6 Kontrollieren 30](#_Toc103007742)

[2.6.1 Testprotokoll 30](#_Toc103007743)

[2.7 Auswerten 31](#_Toc103007744)

[2.7.1 Schlusswort 32](#_Toc103007745)

[3 Anhang 33](#_Toc103007746)

[3.1 Quellenverzeichnis 33](#_Toc103007747)

[3.2 Glossar 33](#_Toc103007748)

[3.3 Tabellenverzeichnis 33](#_Toc103007749)

[3.4 Abbildungsverzeichnis 33](#_Toc103007750)

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Datum** | **Bemerkungen** | **Bearbeiter** |
| 1.0 | 09.05.2022 | Dokument erstellt | Ruanin Khankah |

# Glossar

|  |  |
| --- | --- |
| **Begriff** | **Definition** |
| IPERKA | Projektplanungsmethode |

# IPA-Bericht Teil 1

## Titel der Arbeit

Mandantenfähige Website

## Arbeitsbereiche

* PHP
* JavaScript
* SQL

## Vorwort

Diese Dokumentation gehört zur IPA von Ruanin Khankah, welcher bei der Firma Confidis AG angestellt ist. In diesem Dokument ist die Arbeit und das Vorgehen dokumentiert. Die Dokumentation ist in 2 Teile gegliedert Im ersten Teil werden die detaillierte Aufgabenstellung und der Ablauf der Arbeit aufgezeigt. Hier werden die vorhandenen Vorkenntnisse, die benutzten Mittel und Methoden aufgezählt. Im zweiten Teil wird das Projekt auch wirklich dokumentiert und beschrieben, es wird auch erwähnt welche Probleme aufgetreten sind. Durch die ganze Arbeit hinweg wurde die Projektplanungsmethode IPERKA eingesetzt

## Detaillierte Aufgabenstellung

Dazu wird ein neues Formular entwickelt, mit dem die Anfragen über die Website entgegengenommen und per E-Mail an die Zweigstelle übermittelt werden.

- Felder:

Vorname (Es sind nur Buchstaben erlaubt. Pflichtfeld)

Name (Es sind nur Buchstaben erlaubt. Pflichtfeld)

E-Mail (Validierung-Mail, Pflichtfeld)

Telefon (Validierung mit PhoneLibrary, Pflichtfeld)

Firma

Funktion

Strasse /Nr.

PLZ (Pflichtfeld)

Ort (Pflichtfeld, Automatischer Vorschlag nach PLZ falls unter PLZ Tabelle vorhanden ist)

Land (Dropdown von Länder Tabelle)

Beruf (Dropdown von Berufstabelle aus Datenbank, Pflichtfeld)

Ich suche (Textfeld, Pflichtfeld)

- Die Datenschutzerklärung muss vor der Übermittlung angenommen werden. Muss sichergestellt werden bis es nach unten gescrollt werden.

- die Webapplikation soll in Responsive Design programmiert sein. Das Formular wird auf der Webseite von Planova eingebunden.

- die im „Look und Feel“ der bestehenden Planova Website ist (www.planova.ch). Die Webapplikation soll in den neuesten Versionen IE, Firefox und Chrome sauber dargestellt sein.

- SQL-Injection muss verhindert werden.

- Die eingegebenen Felder sollten gespeichert werden (SESSION), falls mehrere Vakanzen zu besetzen sind.

- Die Kunden sollten informiert werden, falls der Beruf der Stellenmeldepflicht unterliegt. Sollte direkt nach Berufsauswahl informiert werden, ob es Stellenmeldepflichtig ist oder nicht.

- Nach erfolgreicher Übermittlung wird der Kunde darüber informiert, dass ein Personalverantwortlicher innerhalb von 48 Stunden auf seine Anfrage antworten wird. Andernfalls kann er direkt unter 0848 752 668 anrufen.

- Spamschutz für Formular ist zwingend.

- Konnte eine E-Mail nicht übermittelt werden, muss eine Logdatei erstellt werden.

- Zusätzlich sollte das Kontaktformular in der Datenbank gespeichert werden.

- Es ist ein Projektjournal zu führen, da noch mit dem Vertrieb die notwendigen Felder abgestimmt werden und über den Ablauf informiert werden sollen.

- Formular müsste Funktionsweise getestet werden (Blackbox Test).

- MVC gemäss Jobcircle Framework verwenden.

Wenn das Formular ausgefüllt wurde, muss der Antrag an info@planova.ch gesendet werden.

Gleichzeitig würde der Besucher benachrichtigt, dass sein Formular abgeschickt wurde und er ein Benutzerkonto anlegen kann.

Dabei ist es wichtig, dass die vorhandenen Daten der Kunden in der Datenbank gespeichert werden. Ist der Kunde bereits eingeloggt, wird er zur Übersichtsseite mit dem eingereichten Personalbedarf weitergeleitet.

Liste Felder:

Erstellungsdatum, Firma, Kontakt, Beruf, Meldepflichtig Status.

## Mittel und Methoden

Windows Computer

Netbeans

PHP, JAVASCRIPT, MySQL, HTML

## Vorkenntnisse

Arbeiten mit PHP, MySQL, Bootstrap. Der Kandidat durfte drei Monate mit den bestehenden Frameworks arbeiten.

## Vorarbeiten

Der Kandidat durfte auf die bestehenden Frameworks das Weiterempfehlungs-System programmieren.

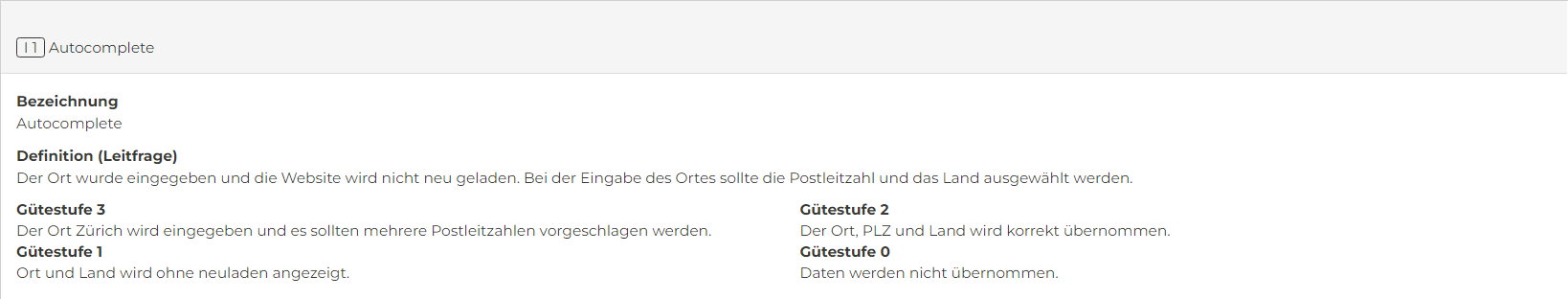
## Neue Lerninhalte

Regex wäre für den Kandidaten neu. Der Kandidat kann es unter https://www.php.net/ nachschlagen.

## Arbeiten in den letzten 6 Monaten

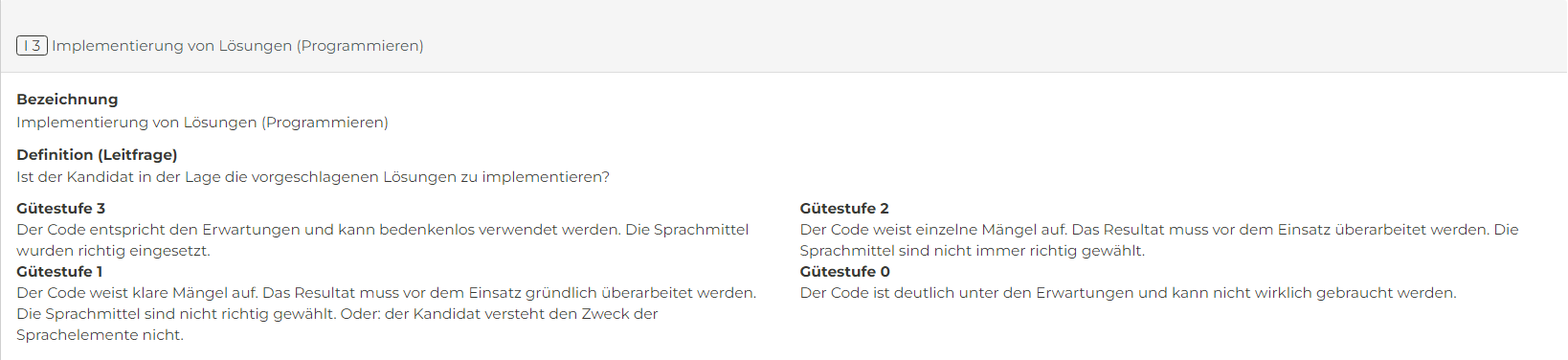
Weiterempfehlungsprogramm

## Mittel und Methoden



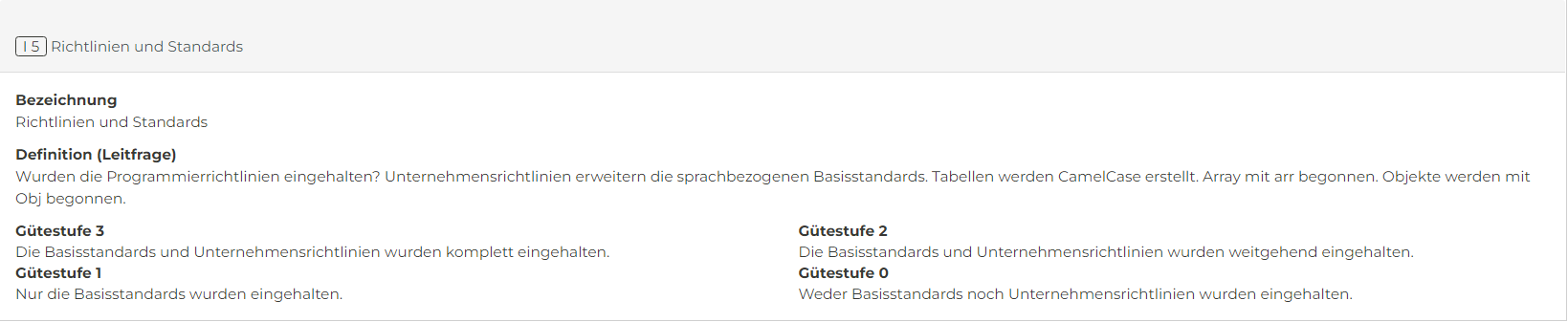
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung



Ein Bild, das Text enthält.

Automatisch generierte Beschreibung



Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## Projektorganisation

### Durchführung

Die IPA wird während zehn Arbeitstagen am Stück mit der Ausnahme von Schul- oder Feiertagen durchgeführt.

**Startdatum:** 09.05.2022

**Enddatum:**  24.05.2022

**Ausführungsort:** Confidis AG

Josefstrasse 218

8005 Zürich

**Beteiligte**

|  |  |  |  |
| --- | --- | --- | --- |
| **Vorname Name** | **E-Mail** | **Telefon** | **Funktion** |
| André Lichtsteiner | andre.lichtsteiner@gmx.ch | +41 79 469 33 67 | Hauptexperte |
| Christian Walder | christian.walder@benedict.ch | +41 44 298 17 83 | Berufsbildner |
| Thiru Siva | thiru.siva@vinosol.ch | +41 58 255 27 17 | Verantwortliche Fachkraft |
|  |  |  | Nebenexpert/in |
| Ruanin Khankah | ruanin\_18@hotmail.com | +41 76 580 93 93 | Kandidat |

### Arbeitsmethode

Für die Umsetzung der IPA wurde die Projektmethode IPERKA verwendet.

Dabei stehen jeder der Buchstaben für eine Projektphase:

I – Informieren

P – Planen

E – Entscheiden

R – Realisieren

K – Kontrollieren

A – Auswerten

Die Anwendung von IPERKA in einem Projektablauf, verlangt eine strukturierte Vorgehensweise. Ich habe mich für diese Arbeitsmethode entschieden, weil es für den Umfang dieses Projekt geeignet ist. Diese Methode hilft sehr bei der Planung sowie beim Verstehen. Hilfreich ist die Methode ebenfalls, um ein Projekt systematisch zu bearbeiten oder ein Problem gezielt zu lösen. Die Planung hat hierbei eine grosse Bedeutung. Erst wenn ein Konzept ausgearbeitet ist und die Entscheidung feststeht, sollte mit der Realisierung begonnen werden.

### Ablage der Daten

Der Zeitplan und der IPA-Bericht werden im folgenden Ordner abgelegt:

C:\xampp4\htdocs\jobcircle\_new\Dokumente

Der Code wir im folgenden Ordner abgelegt:

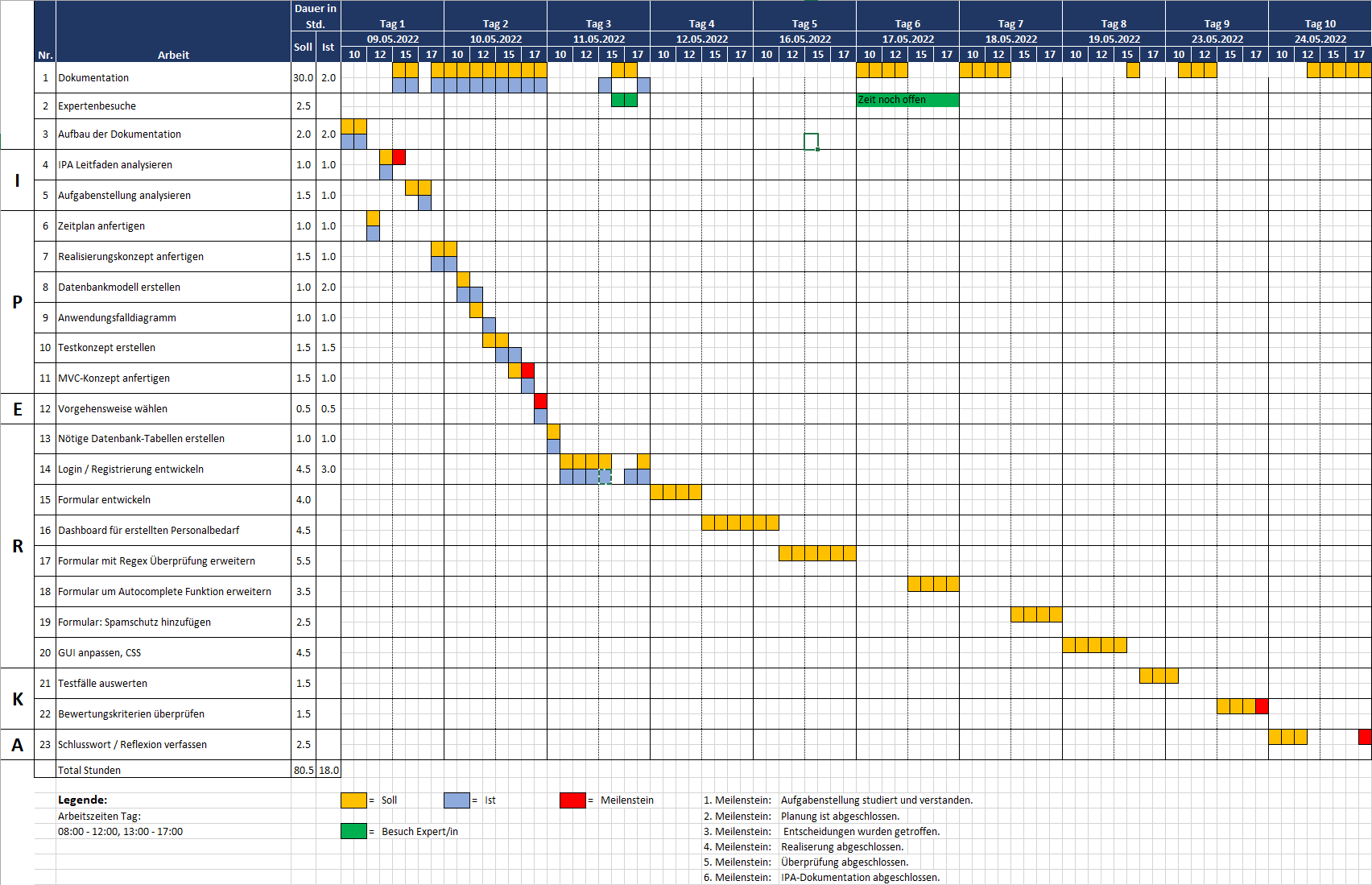
C:\xampp4\htdocs\jobcircle\_new\Code

In diesem Ordner wird für jeden Tag der IPA ein Unterordner erstellt, in welchem am Ende des Tages der Stand der Dokumente abgelegt wird.

Die Implementierung der Oberfläche wird im zur Verfügung gestellten Git-Repository abgelegt und versioniert. Diese ist öffentlich und unter folgendem Link abrufbar:

https://github.com/ruanin/jobcircle\_origin

## Zeitplan



### Meilensteine

**09.05.2022 – Informieren**

Aufgabenstellung durchlesen & verstehen

**10.05.2022 - Planung**

**10.5.2022 – Entscheiden**

**19.05.2022 – Realisieren**

**23.05.2022 – Kontrollieren**

## Arbeitsjournal

### Tag 1 – 09.05.2022

|  |
| --- |
| Tagesablauf |
| Anfangs habe ich angefangen meine Dokumentation aufzubauen und noch meinen Soll/Ist-Zeitplan fertig gestellt. Zeitgleich habe ich den IPA-Leitfaden als auch meine Aufgabenstellung studiert. Da waren mir einige Sachen noch etwas unklar, weshalb ich diese bei meiner vertretenden Fachkraft nachgefragt habe. Nur reine Verständnissachen.  Die Fragen:   * Ein MVC-Konzept ist klar was das ist, aber wo sehe ich, was die „Jobcircle Framework“ für spezifische Vorschriften hat an die ich mich halten müsste (Geschäftslogik)? * Liste Felder: „Erstellungsdatum, Firma, Kontakt, Beruf, Meldepflichtig Status“ für die Übersichtsseite mit dem eingereichten Personalbedarf gemeint, richtig? Steht kein Kommentar dazu. * In der Aufgabenstellung steht, dass beim übermitteln des Formulars der Kunde ein Benutzerkonto anlegen KANN. Ist es in diesem Fall also nicht obligatorisch, dass er ein Konto anlegen muss?   Zum Schluss habe ich angefangen mein Realisierungskonzept zu schreiben |
| Ausgeführte Arbeiten |
| * Soll/Ist-Zeitplan erstellt * Dokumentation aufgebaut. * IPA-Leitfaden & Aufgabenstellung analysiert. * Angefangen ein Realisierungskonzept zu schreiben |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
| * Der Zeitplan steht. * Die Dokumentation ist strukturiert und wurde erweitert. |
| Aufgetretene Probleme |
| * Keine |
| Benötigte Hilfestellungen |
| * Keine |
| Reflexion |
| Heute war der erste IPA-Tag, ich war anfangs sehr nervös, aber als ich dann rein kam und die ersten Meilensteine erreichte, war das dann kein Problem mehr. Ich bin gut gestartet und konnte viel erledigen. Ich bin gespannt auf die nächsten Tage sowie auf die Treffen mit dem Hauptexperten. |

### Tag 2 – 10.05.2022

|  |
| --- |
| Tagesablauf |
| Heute habe ich die Planung sowie die Entscheidung von der IPERKA Methode abgeschlossen. Ich konnte das Realisierungskonzept abschliessen, ich habe ein Datenbankmodell erstellt samt Beziehungen, ein Anwendungsfalldiagramm, das Testkonzept definiert und ich habe mein Vorgehen mit dem MVC-Konzept niedergeschrieben. |
| Ausgeführte Arbeiten |
| * Realisierungskonzept abgeschlossen * Datenbankmodell erstellt * Anwendungsfalldiagramm erstellt * Testkonzept definiert (Black-Box-Test) * MVC-Konzept definiert * Vorgehensweise gewählt |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
| * Realisierungskonzept steht * Ich konnte die Datenbankbeziehungen erstellen * Ich habe das Anwendungsfalldiagramm erstellt * Das Testkonzept definiert * Eine Grafik für das MVC-Konzept erstellt. * Den Punkt Entscheidungen abgeschlossen. |
| Aufgetretene Probleme |
| * Keine |
| Benötigte Hilfestellungen |
| * Keine |
| Reflexion |
| Der zweite Tag der IPA war schon etwas anstrengender, da ich laut dem Zeitplan auch etwas mehr machen musste. Aber an sich ging es gut mit der Zeit und ich bin auch zufrieden gewesen heute. Ich konnte mir schon einen genauen Plan machen und weiss wie ich vorgehen muss. |

### Tag 3 – 11.05.2022

|  |
| --- |
| Tagesablauf |
| Heute habe ich mir vorgenommen, das Datenbankmodell umzusetzen und die Tabellen zu erstellen samt Beziehungen. Ebenfalls habe ich mir vorgenommen, mit dem Punkt Realisieren anzufangen, heisst gemäss dem Zeitplan mit anzufangen das Login und Registrierungsformular zu programmieren. Auch treffe ich heute den Experten für das erste Treffen. In diesem Treffen besprechen wir den Start, die Bewertungskriterien und die Vorgehensweise. |
| Ausgeführte Arbeiten |
| Die Tabellen für die Datenbank habe ich nach dem Datenbankmodell erstellt und die dazugehörigen Beziehungen. Ebenfalls habe ich das Registrierungsformular soweit programmiert, das es schon mal möglich ist sich zu registrieren. Über eine Verlinkung im regulären Anmeldeformular für den Kandidaten, wird man zum View RegisterNewClient weitergeleitet welche über den Controller Client Daten vermittelt bekommt. Von dort aus habe ich das Formular mit den nötigen Feldern entwickelt und es ist möglich sich per Mail oder Telefon zu registrieren. |
| Ungeplante Arbeiten |
| * Das erste Treffen wurde auf eine Stunde nachgeschoben, |
| Erreichte Ziele |
| * Das Registrieren als Kunde ist möglich. * Die Datenbanktabellen stehen samt Beziehungen. |
| Aufgetretene Probleme |
| * Ich musste für kleinere Syntaxfehler länger Reverse Engineering betreiben und das hat mich etwas Zeit gekostet. |
| Benötigte Hilfestellungen |
| * Ich habe meine VF die Frage gestellt, ob ich das Registrierungsformular am besten separat einbetten sollt in der Navigation oder ob ich das mit einem Input Radio bestätigen kann im vorhanden Formular für Kandidaten. Er verwies mich darauf, das es am besten ist eine Verlinkung im Registrierungsformular für Kandidaten einzuprogrammieren. |
| Reflexion |
| Am 3. Tag fing das Realisieren an. Ich hatte anfangs keine Schwierigkeiten. Im Prozess habe ich etwas Zeit verloren durch einige Syntaxerror. Der Experte kam um 16:30 und hat mich über einige Sachen aufgeklärt und wir haben zusammen mit meinem VF die individuellen Bewertungskriterien besprochen und etwas abgeändert. Ich zeigte dem Experten auch meine bisherige Doku und Zeitplan und er verwies mich auf Verbesserungsvorschläge. Im grundlegenden bin ich auch hier zufrieden und motiviert. |

### Tag 4 – 12.05.2022

|  |
| --- |
| Tagesablauf |
|  |
| Ausgeführte Arbeiten |
|  |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
|  |
| Aufgetretene Probleme |
|  |
| Benötigte Hilfestellungen |
|  |
| Reflexion |
|  |

### Tag 5 – 16.05.2022

|  |
| --- |
| Tagesablauf |
|  |
| Ausgeführte Arbeiten |
|  |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
|  |
| Aufgetretene Probleme |
|  |
| Benötigte Hilfestellungen |
|  |
| Reflexion |
|  |

### Tag 6 – 17.2022

|  |
| --- |
| Tagesablauf |
|  |
| Ausgeführte Arbeiten |
|  |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
|  |
| Aufgetretene Probleme |
|  |
| Benötigte Hilfestellungen |
|  |
| Reflexion |
|  |

### Tag 7 – 18.05.2022

|  |
| --- |
| Tagesablauf |
|  |
| Ausgeführte Arbeiten |
|  |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
|  |
| Aufgetretene Probleme |
|  |
| Benötigte Hilfestellungen |
|  |
| Reflexion |
|  |

### Tag 8 – 19.05.2022

|  |
| --- |
| Tagesablauf |
|  |
| Ausgeführte Arbeiten |
|  |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
|  |
| Aufgetretene Probleme |
|  |
| Benötigte Hilfestellungen |
|  |
| Reflexion |
|  |

### Tag 9 – 23.05.2022

|  |
| --- |
| Tagesablauf |
|  |
| Ausgeführte Arbeiten |
|  |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
|  |
| Aufgetretene Probleme |
|  |
| Benötigte Hilfestellungen |
|  |
| Reflexion |
|  |

### Tag 10 -24.05.2022

|  |
| --- |
| Tagesablauf |
|  |
| Ausgeführte Arbeiten |
|  |
| Ungeplante Arbeiten |
|  |
| Erreichte Ziele |
|  |
| Aufgetretene Probleme |
|  |
| Benötigte Hilfestellungen |
|  |
| Reflexion |
|  |

# IPA Bericht Teil 2

## Kurzfassung

Diese Kurzfassung richtet sich an die fachlich kompetenten Leser mit Fachwissen in der

Informatik. Sie soll den Einstieg für das Verständnis dieser Arbeit erleichtern und einen

ersten Überblick liefern.

### Ausgangslage

Bei der Planova erhält man die Stellenangebote aktuell via E-Mail oder auch telefonisch. Viele Mail-Anfragen sind nicht vollständig. Das bedeutet für den Personalberater einen zusätzlichen Arbeitsaufwand, da er telefonisch den Kunden nach fehlenden Informationen erbitten muss.

Zur Entlastung der Personalberater wäre es hilfreich, wenn die Kunden die offenen Stellen online über die Website übermitteln können.

### Umsetzung

Zu Beginn des Projekts wurde ein Zeitplan passend zu den Phasen der IPERKA Projektmethode erstellt.

Das Projekt wird in drei Teile gegliedert. In das Modell (Datenlogik und Anbindung der Datenbank), der View (Ergebnispräsentation & UI) und der Controller(Vermittlung & Steuerung).

Das gewünschte Formular wird mit der Entwicklungsumgebung NetBeans DIE 12.0 entwickelt. Als Framework nutze ich dafür Bootstrap und Laravel.

### Ergebnis

## Informieren

Beim erste Teil der IPERKA-Methode geht es darum sich zu informieren. In dieser Phase wird der Auftrag genauestens analysiert und die Erkenntnisse ausgewertet. Die wesentlichen Punkte werden so früh wie möglich erkannt und allfällige Fragen geklärt.

### Anfängliche Unklarheiten

Zu klärende Fragen gehörten hierbei:

* Ein MVC-Konzept ist klar was das ist, aber wo sehe ich, was die „Jobcircle Framework“ für spezifische Vorschriften hat an die ich mich halten müsste (Geschäftslogik)?
* Liste Felder: „Erstellungsdatum, Firma, Kontakt, Beruf, Meldepflichtig Status“ für die Übersichtsseite mit dem eingereichten Personalbedarf gemeint, richtig? Steht kein Kommentar dazu.
* In der Aufgabenstellung steht, dass beim übermitteln des Formulars der Kunde ein Benutzerkonto anlegen KANN. Ist es in diesem Fall also nicht obligatorisch, dass er ein Konto anlegen muss?

Diese Fragen wurden meiner vertretenden Fachkraft per E-Mail übermittelt, ich warte derzeit noch auf eine Antwort. Ich konnte sonst soweit den Auftrag, den Leitfaden als auch die Kriterien gut verstehen.

## 

## Planen

### Realisierungskonzept

Als Entwicklungsumgebung wird die Dev-Umgebung von planova.ch benutzt. Das schliesst PHP und MySQL ein.

1. **Login/Registrierungs-Funktion**

Zunächst sollte es möglich sein sich als Kunde, also als Arbeitsgeber zu registrieren. Da man sich bereits schon als Kandidat, also Arbeitssuchenden registrieren und einloggen kann, muss ich darauf achten, bei jeder Login Abfrage klar definiert zu haben welches Model abgerufen werden soll.

Bei der Abfrage registrieren kann man sich dann ganz klar dafür entscheiden, ob man sich mit einem Kudenkonto registrieren will, oder mit einem Kandidaten Konto. Die Login Abfrage wird dann so ablaufen, dass der bereits verfügbare Controller Candidate das Model „cand\_user“, die Login Daten mit der Datenbank abgleicht und sollten diese dann keine Übereinstimmung treffen, mit dem noch zu erstellenden Model „cust\_user“. Die Abfrage erfolgt zuerst mit der Tabelle der Kandidaten, da die Nachfrage für die deutlich höher ist und sein wird als die Abfrage von Kundenkonten.

Bei der Registrierung wird darauf geachtet, das keine Mutationen entstehen und ein Mail jeweils für ein Kundenkonto und Kandidatenkonto verwendet werden kann.

Die Felder für die Registrierung sind: Vorname, Name, E-Mail, Telefon, Firma, Funktion, Strasse/Nr., PLZ, Ort und Land. Nach der Registrierung wird es im Dashboard einen Bereich geben in welchem man diese Daten updaten kann sollte man es müssen.

Die gewünschten Daten werden auch hier mit Regex geprüft und in Echtzeit bemängelt, sollten falsche Zeichen etc. angegeben worden sein.

1. **Formular zum Personalbedarf**

Unabhängig davon, ob man jetzt registriert, eingeloggt oder ausgeloggt ist, sollte einem die Ansicht zum Formular gewährt werden. Wenn man die Formular-Übermittlung bestätigt, wird je nach Case abgefragt ob man sich einloggen, sich registrieren, oder sofort zur Übersichtseite gelangen will ohne sich einzuloggen.

Wenn man sich dazu entscheidet, das Formular zu bestätigen ohne sich einzuloggen oder zu registrieren, wird das Formular keinem Benutzer zugewiesen als Referenz.

Sollte man bereits eingeloggt sein, wird das gewünschte Record auf der Übersichtsseite dem Kundenkonto als Referenz zugewiesen, so kann er auf der Übersichtsseite einsehen, wie viele Personalbedarfe er wann abgeschickt hat. Beim Registrieren funktioniert das gleich, nur das man sich vorher zuerst registrieren muss.

1. **Übersichtsseite**

Wie beim zweiten Punkt erwähnt, soll der Kunde zu einer Übersichtsseite gelangen sobald dieser das Formular übermittelt hat. Dort sollten ihm all seine offenen Personalbedarfe angezeigt werden. Diese Records werden mit den gewünschten Feldern dargestellt (Erstellungsdatum, Firma, Kontakt, Beruf, Meldepflichtig Status).

Uneingeloggt ist diese Seite nur einmalig aufrufbar, eingeloggt wird sie im Profil gespeichert.

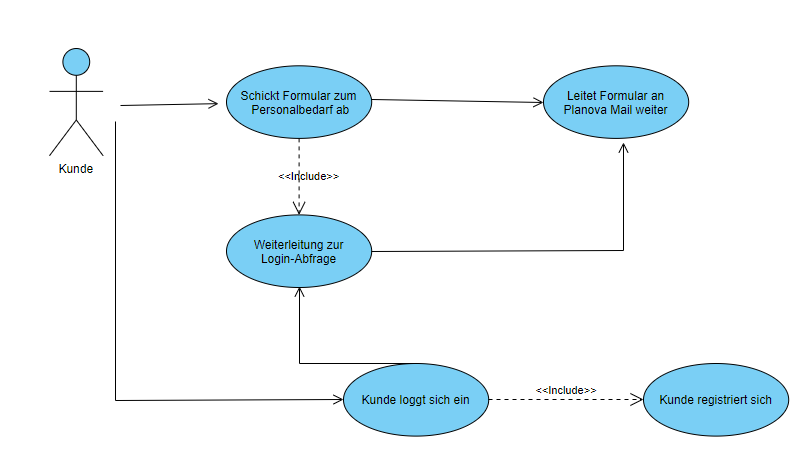
1. **Übermittlung an E-Mail Adresse**

Zeitgleich zur Weiterleitung an die Übersichtsseite werden die Daten des übermittelten Formulars an eine bestimmte E-Mail versandt. Da ich lokal arbeite, werde ich hierfür ein SMTP-Server benutzen mit einer anderen E-Mail. Wenn das Projekt migriert wird auf den Liveserver, wird das dann mit dem Firmenprovider übermittelt, es geht nur darum es lokal testen zu können.

Das Formular übermittelt alle nötigen Felder an die Zweigstelle.

### Datenbankmodell

### Anwendungsfalldiagramm



### Testkonzept

Es wird die Black Box Testmethode eingesetzt. Dies ist eine Testmethode welche die Funktionalität einer Anwendung untersucht, ohne in ihre internen Strukturen oder Funktionsweisen einzudringen, also ohne sich den Code anzuschauen. So können wir einen klaren Einblick in die Userexperience schaffen und uns in den Kunden hineinversetzen.

|  |  |
| --- | --- |
| **Testfall-Nr:** | **1** |
| **Anforderung:** | Der User kann sich registrieren |
| **Beschreibung:** | Er sollte in der Lage sein, sich von der Verlinkung „Registrieren | Anmeleden“ registrieren zu können. Sobald er dies getan hat sollte er automatisch eingeloggt sein. |
| **Voraussetzung:** | * Keine |
| **Erwartetes Resultat:** | Der User ist registriert und eingeloggt, es sollte ein Datenbankverzeichnis erstellt. |

|  |  |
| --- | --- |
| **Testfall-Nr:** | **2** |
| **Anforderung:** | Der User kann sich ein- und ausloggen |
| **Beschreibung:** | Er sollte in der Lage sein, sich von der Verlinkung „Registrieren | Anmeleden“ einloggen zu können. Sobald er dies getan hat sollte er automatisch eingeloggt sein. |
| **Voraussetzung:** | * Der User ist registriert |
| **Erwartetes Resultat:** | Der User ist registriert und eingeloggt, es sollte ein Datenbankverzeichnis erstellt. |

|  |  |
| --- | --- |
| **Testfall-Nr:** | **3** |
| **Anforderung:** | Der User kann das Personalbedarf-Formular abschicken ohne sich einzuloggen |
| **Beschreibung:** | Er sollte in der Lage sein das Formular auszufüllen, abzusenden und das alles ohne sich einzuloggen geschweige denn registriert zu sein. |
| **Voraussetzung:** | * User übermittelt das Formular |
| **Erwartetes Resultat:** | Das Formular des Users sollte weitergeleitet werden an die Zweigstelle, es sollte ein Datenbankverzeichnis erstellt. |

|  |  |
| --- | --- |
| **Testfall-Nr:** | **4** |
| **Anforderung:** | Der User kann das Personalbedarf-Formular abschicken und sich anschliessend registrieren |
| **Beschreibung:** | Er sollte in der Lage sein das Formular auszufüllen, abzusenden und bei der Abfrage ob er ein Benutzerkonto anlegen will, sollte er eins erstellen können. |
| **Voraussetzung:** | * User übermittelt das Formular |
| **Erwartetes Resultat:** | Das Formular des Users sollte weitergeleitet werden an die Zweigstelle, es sollte ein Datenbankverzeichnis erstellt indem der frisch registrierte Account als Referenz angegeben wird für den gerade abgesendeten Personalbedarfsformular. |

|  |  |
| --- | --- |
| **Testfall-Nr:** | **5** |
| **Anforderung:** | Der User kann das Personalbedarf-Formular abschicken und sich anschliessend einloggen |
| **Beschreibung:** | Er sollte in der Lage sein das Formular auszufüllen, abzusenden und bei der Abfrage ob er ein Benutzerkonto anlegen will, sollte er eins erstellen können. |
| **Voraussetzung:** | * User übermittelt das Formular |
| **Erwartetes Resultat:** | Das Formular des Users sollte weitergeleitet werden an die Zweigstelle, es sollte ein Datenbankverzeichnis erstellt indem der eingeloggte Account als Referenz angegeben wird für den gerade abgesendeten Personalbedarfsformular. |

|  |  |
| --- | --- |
| **Testfall-Nr:** | **6** |
| **Anforderung:** | Der User wird beim ausfüllen des Formulars darauf hingewiesen, dass er falsche Zeichen bei Feldern verwendet |
| **Beschreibung:** | User gibt Felder falsch ein. |
| **Voraussetzung:** | * User füllt Felder falsch aus |
| **Erwartetes Resultat:** | Wenn unerlaubte Zeichen eingesetzt werden muss der User in echtzeit benachrichtigt werden, dass unerlaubte Zeichen vorhanden sind, welche es zu bereinigen gibt. |

|  |  |
| --- | --- |
| **Testfall-Nr:** | **7** |
| **Anforderung:** | User gibt die PLZ im Formular an und das Feld Ort wird automatisch ausgefüllt. |
| **Beschreibung:** | Von der PLZ sollte das System nachvollziehen können, welcher Ort einzusetzen ist. Dieser wird dann vorgeschlagen und eingesetzt, aber es ist noch änderbar. |
| **Voraussetzung:** | * User füllt das Feld PLZ aus. |
| **Erwartetes Resultat:** | Wenn unerlaubte Zeichen eingesetzt werden muss der User in echtzeit benachrichtigt werden, dass unerlaubte Zeichen vorhanden sind, welche es zu bereinigen gibt. |

|  |  |
| --- | --- |
| **Testfall-Nr:** | **8** |
| **Anforderung:** | Der User kann nicht in zu kurzem Zeitabstand mehrere Formulare abschicken |
| **Beschreibung:** | Um vor Spam geschützt zu sein, gibt es eine gewisse Cooldown Zeit in welcher der User nach dem übermitteln des Formulars kein weiteres Formular abschicken kann. |
| **Voraussetzung:** | * User übermittelt schnell und mehrmals das Personalbedarfsformular |
| **Erwartetes Resultat:** | * Der User wird darauf hingewiesen, dass er es später nochmal versuchen soll. |

### Model-View-Controller-Konzept

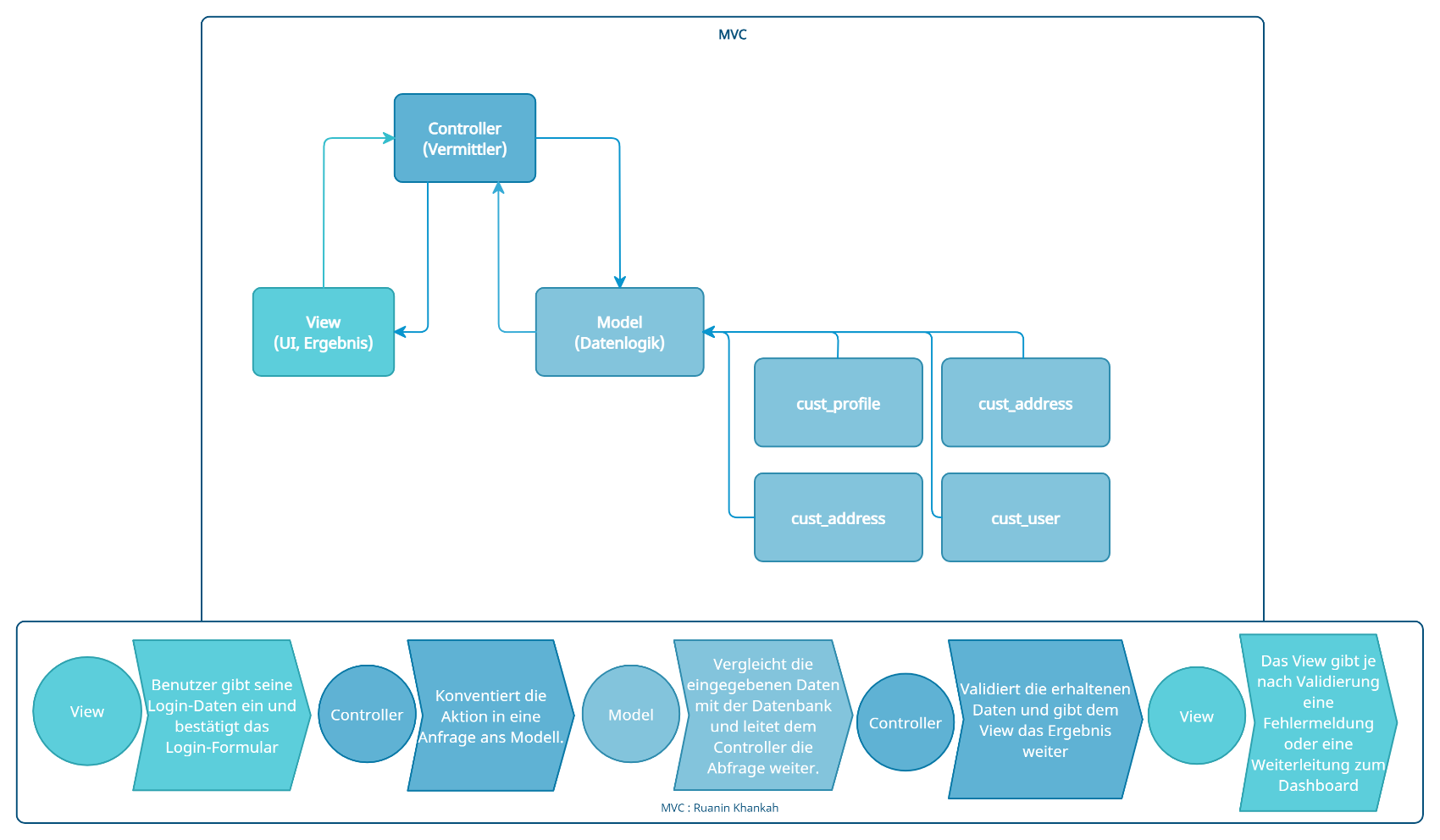
Das MVC-Modell ist ein Architektur Muster in Webseiten und Softawre.

Im Model liegt die Datenlogik und die Anbindung an die Datenbank.

Der View kümmert sich um die Ergebnispräsentation, also um das User Interface. Die hierfür benötigten Daten kommen aus dem Model.

Der Controller wiederrum ist für die Vermittlung & Steuerung zuständig, Model und View kommunizieren nicht direkt miteinander sondern lassen sich vom Controller Anweisungen geben. Gleichzeitig kümmert der Controller sich um die Anfragen vom Benutzer. Auch was bei validen oder invaliden Ergebnissen zu tun ist entscheidet der Controller.

Eine Veranschaulichung zum MVC-Konzept:



## Entscheiden

### Loginsystem

Bei dieser Entscheidung geht es darum festzulegen, ob Kandidaten und Kunden die gleiche Ansicht haben sollen und es nur einen kleinen festgelegten Bereich für den Personalbedarf der Kunden geben soll oder beide Gruppen jeweils verschiedene Ansichten haben. Das heisst die Kunden können ihren Lebenslauf ergänzen, während Kunden das nicht können, da sie es auch nicht brauchen.

Hier war mir dann schnell bewusst, dass ich ein eigenständiges Login System für Kunden machen will, wo am Anfang auch ganz klar zu definieren ist, ob man ein Kundenkonto erstellen will oder ein Kandidatenkonto. Dabei führen die Tabellen ‘email‘ und ‘phone\_number‘ für beide Gruppen jeweils die Abfragen aus um Mutationen zu vermeiden. Bei der Login Abfrage wird zuerst geprüft ob das eingegebene Mail und Passwort mit einem Kandidatenkonto übereinstimmt und dann erst das mit dem eines Kunden. Der Grund hierfür ist, dass die Nachfrage an Kandidatenkonten anfangs sowie auch in Zukunft höher sein wird als die von Kunden.

### Anzeige der Auswertung

Sobald man das Personalbedarfs-Formular übermittelt hat kommt man auf eine Übersichtsseite in welchem man seinen eben gesendeten Personalbedarf dargestellt bekommt samt Nachricht, dass man in 48 Stunden kontaktiert wird etc.

Da man dies ausgeloggt und eingeloggt machen kann, habe ich mich dazu entschlossen, bei registrierten Benutzer den übermittelten Personalbedarf dem Konto zuzuweisen. Das heisst, er kann stetig auf sein Profil und seinen übermittelten Personalbedarf nachverfolgen.

Bei Benutzer die ausgeloggt sind und kein Konto besitzen, wird die Übersichtsseite mit dem erstellten Record einmalig angezeigt. Der Personalbedarf wird wie gewohnt in der Datenbank erfasst nur keinem Benutzer zugeteilt.

## Realisieren

### Datenbank-Tabellen

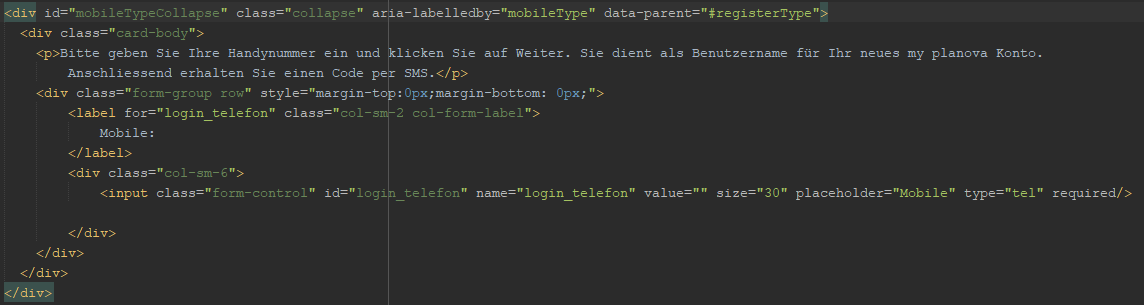
Nach dem Datenbankmodell habe ich folgende Tabellen erstellt. „cust“ steht hierbei für Customer:

* cust\_user
* cust\_email
* cust\_phone\_number
* cust\_profile
* cust\_address

### Login-System

Beim Login-System ist wichtig zu verstehen, es gibt bereits ein Login-System für die Gruppe Kandidat, parallel dazu muss ein Login-System für die Gruppe Kunde geschaffen werden.

Für die Gruppe Kunde nutze ich den Controller „Client“. Zunächst einmal habe ich das Registrierungsformular erstellt. Hierbei ist es möglich, sich mit einer Handynummer oder einer Email zu registrieren:



## Kontrollieren

### Testprotokoll

|  |  |
| --- | --- |
| Randbedingungen (Umfeld): |  |
| Testszenario: |  |
| Testmittel: |  |

#### … Tests

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nr. | Kategorie | Testanweisung | Erwartetes Resultat | Resultat | Massnahmen | OK |
|  |  |  |  |  |  |  |

## Auswerten

### Schlusswort

# Anhang

## Quellenverzeichnis

|  |  |  |  |
| --- | --- | --- | --- |
| Medium | Information (Autor, Titel) | Quelle (Link oder ISBN) | Datum |
| Internet |  |  |  |

Tabelle 9: Quellenverzeichnis

## Glossar

|  |  |
| --- | --- |
| **Begriff/Abkürzung** | **Erklärung** |
|  |  |

Tabelle 10: Glossar

## Tabellenverzeichnis

[Tabelle 9: Quellenverzeichnis 26](#_Toc403031728)

[Tabelle 10: Glossar 26](#_Toc403031729)

## Abbildungsverzeichnis