



IMS

Immobilien Management System

Est. 2020

Michael Citcer

Valentin Gutbelet

Maximilian Müller

IT12c

Inhaltsverzeichnis

Einleitung	2
Projektumfeld	2
Projektziel.....	2
Projektbegründung.....	2
Projektschnittstellen.....	2
Projektplanung	2
Projektphasen	2
Ressourcenplanung	2
Entwicklungsprozess	3
Analysephase	3
Ist-Analyse	3
Wirtschaftlichkeitsanalyse	3
Anwendungsfälle	3
Qualitätsanforderungen	3
Entwurfsphase.....	3
Zielplattform	3
Architekturdesign	4
Entwurf der Benutzeroberfläche.....	4
Datenmodell	5
Geschäftslogik.....	5
Maßnahmen zur Qualitätssicherung	5
Implementierungsphase	5
Implementierung der Datenstrukturen.....	5
Implementierung der Benutzeroberfläche	5
Implementierung der Geschäftslogik.....	5
Abnahmephase.....	5
Einführungsphase	6
Dokumentation.....	6
Fazit.....	6
Soll-/Ist-Vergleich.....	6
Lessons Learned	6
Ausblick	6
Eidesstattliche Erklärung	6
Anhang.....	7
A1 Zeitplanung	7
A2 Use-Case-Diagramm	7
A3 Datenbankmodell semantisch und logisch	8
A4 Sprints	9
A5 Screenshots der Anwendung.....	13
A6 Klasse: Beispielklasse aus dem Projekt	17

Datenbankprojekt aus dem Schulunterricht der Berufsschule Kempten 1 im Fach Anwendung u. Programmierung.

Sourcecode: <https://git.io/JYfz5>

Einleitung

Projektumfeld

In der zwölften Klasse der schulischen Ausbildung des Fachinformatikers für Anwendungsentwicklung soll eine Datenbank versierte Projektarbeit durchgeführt werden. Auftraggeber ist dabei die Berufsschule 1 Kempten für IT-Berufe in Persona von Klassenleiter Martin Blanke. Es arbeiten insgesamt drei Teilnehmer an dem Projekt.

Projektziel

Ziel des Projekts ist die Erstellung einer Datenbank basierten Anwendung. Die Form der Ausarbeitung, also als Weblösung oder Desktopanwendung, ist dabei frei wählbar.

Projektbegründung

Es soll eine Möglichkeit für Vermieter und Eigentümer geschaffen werden, ihre Immobilien zu verwalten, neue Objekte anzulegen und den Überblick über ihre bereits vermieteten Objekte zu behalten. Des Weiteren soll das gesamte Tool die Verwaltung der Objekte zentralisieren, sodass mehrere Benutzer simultan, immer auf den aktuellsten Stand zugreifen können.

Projektschnittstellen

Für die Zukunft des IMS kann die Open Street Map Schnittstelle zur visuellen, geographischen Darstellung der Immobilien eingebaut werden. Zum Zeitpunkt der Abgabe des Projekts wurden keine Schnittstellen verwendet.

Projektplanung

Projektphasen

Das gesamte Projekt wurde pro Blockwoche in sogenannte Sprints eingeteilt und somit agil entwickelt. Am Anfang jeder Woche wurden in der Sprintplanung Ziele gesetzt die es zu erreichen galt. Am Ende der Woche wurden die Ergebnisse dann in der gesamten Klasse in Sprintreviews besprochen und präsentiert.

Anfangen mit Sprint Nr. 1 der Kalenderwoche (nachfolgend KW) 43 (siehe A6.1), wurden die Rahmenbedingungen des Projekts abgesteckt. Es wurde ein GitHub Repository erstellt, eine Entwicklungsumgebung ausgewählt und auf den einzelnen Systemen der Projektteilnehmer lokal aufgesetzt. Danach wurde ein Datenbankkonzept initial geplant und per DB Designer erstellt.

In den folgenden Sprints (A6.2-A6.3) wurde das Datenbankkonzept weiter ausgearbeitet und erste Diagramme zur Konzeptionierung der Benutzeroberfläche erstellt.

Nach Fertigstellung der Datenbank, konnte die Benutzeroberfläche erstellt und Logiken zur Datenabfrage und Datenerstellung implementiert werden. (A6.4-A6.5)

Ressourcenplanung

Bei der Planung der Ressourcen war es zunächst essenziell die Stärken und Schwächen der jeweiligen Projektteilnehmer auszuloten. Dementsprechend wurde ein Datenbankbeauftragter ausgewählt, der sich um das Datenbankschema kümmert. Ein Teilnehmer war für das Frontend und die Benutzeroberfläche verantwortlich. Zur Abrundung und vollständigen Abdeckung wurde ein dritter Teilnehmer zum Verantwortlichen der Sprintplanung und zur Erstellung der Dokumentation beauftragt. Damit die Ressourcen optimal genutzt werden konnten, bestand zu jeder Zeit ein reger Datenbankprojekt aus dem Schulunterricht der Berufsschule Kempten 1 im Fach Anwendung u. Programmierung.

Sourcecode: <https://git.io/JYfz5>

Informationsaustausch zwischen den Parteien. Dadurch konnte jeder Einblicke in die Aufgabenbereiche der jeweiligen Projektpartner erhalten.

Entwicklungsprozess

Das gesamte Projekt wird im Stil der agilen Softwareentwicklung durchgeführt. In diesem Rahmen wird Frontend First angewendet.

Analysephase

Ist-Analyse

Potenzielle Kunden des IMS sind Vermieter und Eigentümer, die ihre gesamten Objekte und Daten, bisher dezentral und teilweise offline gepflegt und verwaltet haben. Bspw. in Ordnern, Exceltabellen oder auch handschriftlich. Die Umstellung auf das IMS ist für viele Kunden unter anderem ein Schritt in die Digitale Zukunft und legt den Grundstein für ein Paperless Office.

Wirtschaftlichkeitsanalyse

Eine Wirtschaftlichkeitsanalyse wurde im Zuge der schulischen Projektarbeit nicht durchgeführt, aufgrund fehlender Ist- und Solldaten.

Anwendungsfälle

Siehe A3 Use-Case-Diagramm.

Qualitätsanforderungen

Um eine möglichst hohe Qualität zu gewährleisten stand bei der Implementierung die Usability an oberster Stelle. Der Benutzer soll sich möglichst alleine und schnell im Programm zurechtfinden. Dafür wurde auf eine „State of the Art UX“, also eine moderne Benutzeroberfläche geachtet. Da die Performance ebenso maßgeblich zum Benutzererlebnis und zu einer hohen Qualität beiträgt, wurde die Anwendung so konzipiert, dass der Page Speed möglichst hoch ist.

Entwurfsphase

Zielpattform

1. Client Server
 - a. **HTML 5**
semantisch korrekt nach aktuellen Standards umgesetzt. Template Struktur zur Implementation von wiederkehrenden Elementen wie Bspw. der Navigation.
 - b. **SCSS (LibSass 3.6.4)**
Sass ist eine Stylesheet-Sprache, die als CSS-Präprozessor mit Variablen, Schleifen und vielen anderen Funktionen, die Cascading Style Sheets nicht mitbringen, die Erstellung von CSS vereinfacht und die Pflege großer Stylesheets erleichtert.
 - c. **jQuery (3.6.0)**
jQuery ist eine freie JavaScript-Bibliothek, die Funktionen zur DOM-Navigation und -Manipulation zur Verfügung stellt. jQuery ist die meistverwendete JavaScript-Bibliothek.
 - d. **Bootstrap 5**

Bootstrap ist ein freies Frontend-CSS-Framework. Es enthält auf HTML und CSS basierende Gestaltungsvorlagen für Typografie, Formulare, Buttons, Tabellen, Grid-Systeme, Navigations- und andere Oberflächengestaltungselemente sowie zusätzliche, optionale JavaScript-Erweiterungen.

e. Material Design for Bootstrap 5 & 4

Material Design ist eine vom Unternehmen Google Inc. entwickelte Designsprache und wurde zuerst bei Google Now verwendet.

2. Server

a. Nginx

nginx ist eine von Igor Sysoev entwickelte, unter der BSD-Lizenz veröffentlichte Webserver-Software, Reverse Proxy und E-Mail-Proxy. Nginx wird derzeit bei rund 67 % der 10.000 Webseiten mit dem höchsten Traffic verwendet.

b. PHP (7.4.*)

PHP ist eine Skriptsprache mit einer an C und Perl angelehnten Syntax, die hauptsächlich zur Erstellung dynamischer Webseiten oder Webanwendungen verwendet wird. PHP wird als freie Software unter der PHP-Lizenz verbreitet.

3. Datenbank

a. MySQL (7.2)

MySQL ist eines der weltweit verbreitetsten relationalen Datenbankverwaltungssysteme. Es ist als Open-Source-Software sowie als kommerzielle Enterpriseversion für verschiedene Betriebssysteme verfügbar und bildet die Grundlage für viele dynamische Webauftritte.

Architekturdesign

Umgesetzt wurde das Projekt dynamisch bzw. agil nach dem „Frontend-First“ Prinzip. Agile Softwareentwicklung (von lateinisch agilis „flink, beweglich“) bezeichnet Ansätze im Softwareentwicklungsprozess, die die Transparenz und Veränderungsgeschwindigkeit erhöhen und zu einem schnelleren Einsatz des entwickelten Systems führen sollen, um so Risiken und Fehlentwicklungen im Entwicklungsprozess zu minimieren. Dazu wird versucht, die Entwurfsphase auf ein Mindestmaß zu reduzieren und im Entwicklungsprozess so früh wie möglich zu ausführbarer Software zu gelangen. Diese wird in regelmäßigen, kurzen Abständen mit dem Kunden abgestimmt. So soll es möglich sein, flexibel auf Kundenwünsche einzugehen, um so die Kundenzufriedenheit insgesamt zu erhöhen. Agile Softwareentwicklung zeichnet sich durch selbstorganisierende Teams sowie eine iterative und inkrementelle Vorgehensweise aus. Neben der agilen Entwicklungstechnik wurde – wie in der Einleitung bereits erwähnt – auf „Frontend-First“ gesetzt, um das Nutzererlebnis an erste Stelle zu setzen. Dabei wurde besonders Wert auf ein optisch ansprechendes Auftreten, Usability u. eine einheitliche CI gelegt. Nachdem die ersten Dummies bzw. Oberflächen gebaut waren konnte die Backendentwicklung und Datenbankverbindung agil auf die Anforderungen des Kunden zugeschnitten werden.

Entwurf der Benutzeroberfläche

Zur einfachen Verwaltung und damit sich die Kunden schnell zurechtfinden, wurde ein Webinterface erstellt. Es sollten alle relevanten Eingaben die für die Erstellung und Verwaltung der

Datenbankprojekt aus dem Schulunterricht der Berufsschule Kempten 1 im Fach Anwendung u. Programmierung.

Sourcecode: <https://git.io/JYfz5>

Immobilien nötig sind, abgebildet werden. Dabei stand aber auch im Vordergrund die Oberfläche möglichst übersichtlich zu gestalten.

Von einem klassischen Login wird der Benutzer über ein Dashboard, mit einer Übersicht über seine Objekte, bis hin zur detaillierten Ansicht zur Anlage und Bearbeitung der Immobilien durch die Anwendung geführt.

Bei der Corporate Identity wurde auch eine schlichte aber dennoch moderne Darstellungsform mit einfacher Farbgebung gewählt. (siehe Anhang)

Datenmodell

Im Anhang „ER Modell“ (A3.1) wird ein ERM dargestellt, welches die Entitäten, Relationen und die dazugehörigen Kardinalitäten enthält.

Im Anhang (A3.2) können im semantischen DB-Modell zusätzlich noch die Attribute der verschiedenen Tabellen eingesehen werden. Die Primär- und Fremdschlüssel werden mit „PK“ und „FK“ gekennzeichnet. Bei der Namensgebung der Verbindungstabellen der m-zu-n-Beziehungen, wurden jeweils die zu verbindenden Tabellen einbezogen.

Geschäftslogik

Der Arbeitsfluss des Unternehmens, welches die Anwendung letztendlich einsetzt, wird insofern geändert, als das mehrere Verwaltungsprozesse zentralisiert werden. Nachdem die Stammdaten eingepflegt wurden, kann IMS vollständig in den täglichen Arbeitsablauf integriert werden.

Maßnahmen zur Qualitätssicherung

Es wurden vereinfachte Blackbox Tests und Whitebox Tests durchgeführt um die Grundlegende Funktionalität zu gewährleisten. Dennoch muss erwähnt werden das es sich – aufgrund der erschwerten Entwicklungsumstände und zeitlichen Einschränkungen – aktuell nur um eine Beta die noch nicht Kundenoptimiert ist handeln kann.

Implementierungsphase

Implementierung der Datenstrukturen

Mit Hilfe Einer SQL Workbench wie bspw. „DataGrip“ konnte die komplette Datenbank in SQL konvertiert werden. Diese wurde mittels eines DB-Designers zuvor händisch erstellt und mit sämtlichen Schlüssel- und Nichtschlüsselattributen befüllt. Dieser initiale SQL-Dump wurde anschließend in der Anwendung eingebunden.

Implementierung der Benutzeroberfläche

Screenshots der Benutzeroberfläche finden sich im Anhang. (A5)

Implementierung der Geschäftslogik

Zu Veranschaulichung der Geschäftslogik findet sich exemplarisch ein Auszug aus dem Backend im Anhang. (Klasse *buildEstateQuery*, A6)

Abnahmephase

Eine vollständige ausführliche Abnahme wurde im Rahmen der schulischen Projektarbeit nicht durchgeführt. Lediglich die jeweiligen Projektteilnehmer haben die Arbeiten ihrer Kollegen kontrolliert und gegebenenfalls korrigiert. Auch die durchgeführten Tests hatten keinerlei Fehler als

Ergebnis. Zum Schluss wurden die Textelemente des IMS und die Dokumentation sowohl grammatikalisch als auch inhaltlich korrigiert.

Einführungsphase

Nach Fertigstellung des Projekts, müsste zum tatsächlichen produktiven Einsatz des IMS, ein Server gefunden werden, auf dem das ganze System gehostet wird. Danach müssten für den Endkunden lediglich noch Logins erstellt werden und das Programm wäre einsatzfähig. Eine geringfügige Schulung zum Verständnis der Oberfläche wäre ebenfalls sinnvoll. Im Rahmen der schulischen Projektarbeit sind dies jedoch nur hypothetische Annahmen.

Dokumentation

Eine Dokumentation der Anwendung selbst wurde nicht erstellt. Jedoch kann das Projekt und dessen Aufbau, über die Versionskontrolle in GitHub, für Entwickler übersichtlich erklärt werden. Für Anwender wäre eine kurze Einführung und Schulung ausreichend. (siehe Einführungsphase)

Fazit

Soll-/Ist-Vergleich

Das Projektziel wurde erreicht und die Zeitplanung (A1) eingehalten.

Zeit, Kosten etc. zu analysieren macht hier keinen Sinn.

Lessons Learned

Durch den regen Austausch zwischen den Projektteilnehmern konnte fachübergreifendes Wissen weitergegeben werden. So konnte z.B. jemand der sich bisher mehr mit Datenbanken beschäftigt hatte vieles aus der Frontendentwicklung lernen. Genauso konnten sich die Teilnehmer hinsichtlich Corporate Identity und Backendentwicklung weiterbilden.

Ausblick

Als mögliche Erweiterung des IMS wäre die Anpassung der UI durch den Benutzer selbst (Themes) denkbar. Ebenso die Implementierung der Kostenfreien OpenStreetMap API zur geographischen Darstellung der Mietobjekte könnte in der Zukunft realisiert werden. Ebenso die Implementation von Rollen. Sprich – welcher User hat welche Rechte und darf welche Aktionen durchführen.

Eidesstattliche Erklärung

Wir, die Autoren, versichern hiermit, dass wir unsere Dokumentation zur schulischen Projektarbeit mit dem Thema

IMS - Immobilienmanagementsystem

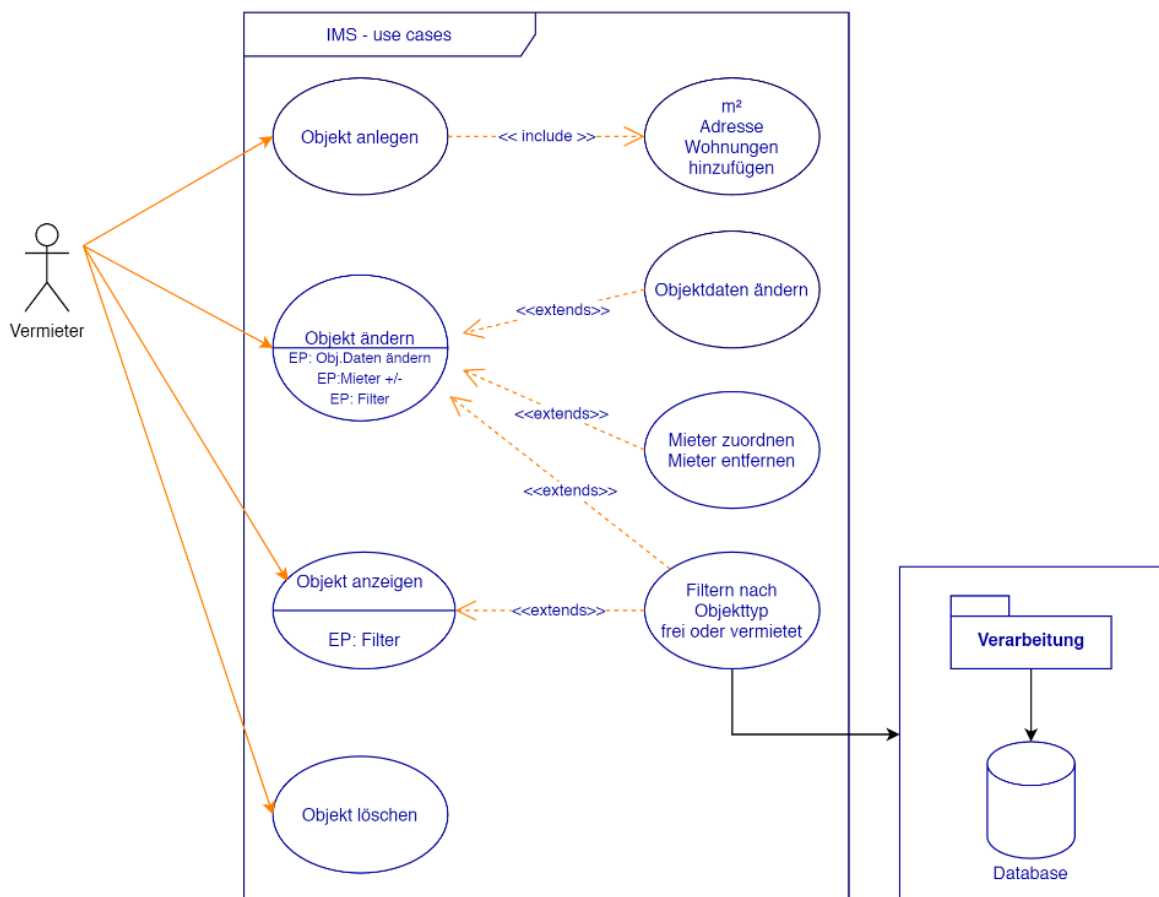
selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben, wobei wir alle wörtlichen und sinngemäßen Zitate als solche gekennzeichnet haben. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Anhang

A1 Zeitplanung

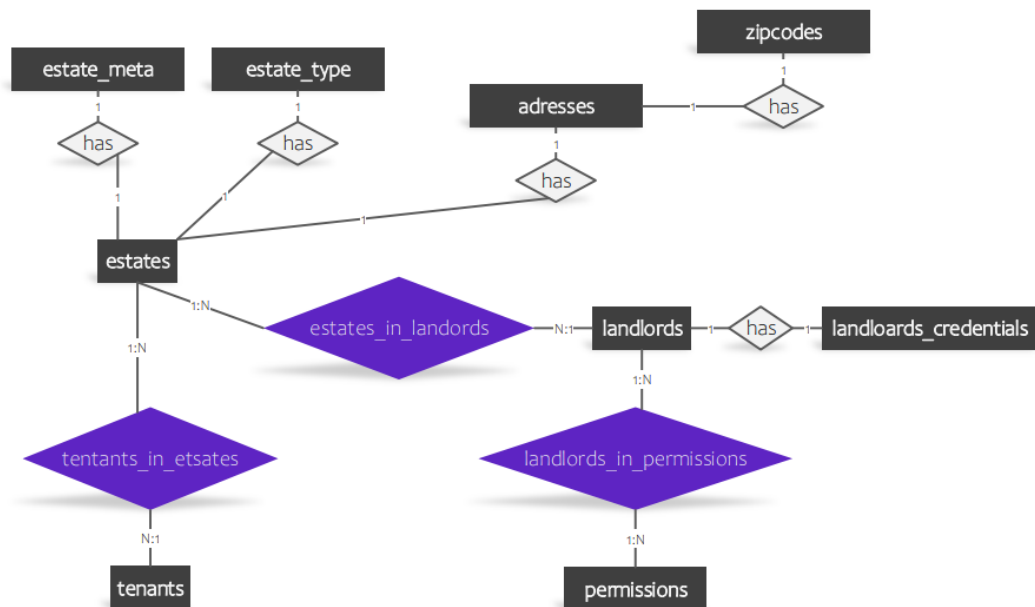
7 Sprints	mal	8 Schulstunden (SH)
Projektphase	Geplante Zeit	
Analyse	8 SH	
Entwurf	10 SH	
Implementierung	28 SH	
Abnahme	4 SH	
Dokumentation	6 SH	
Gesamt	56 SH	

A2 Use-Case-Diagramm

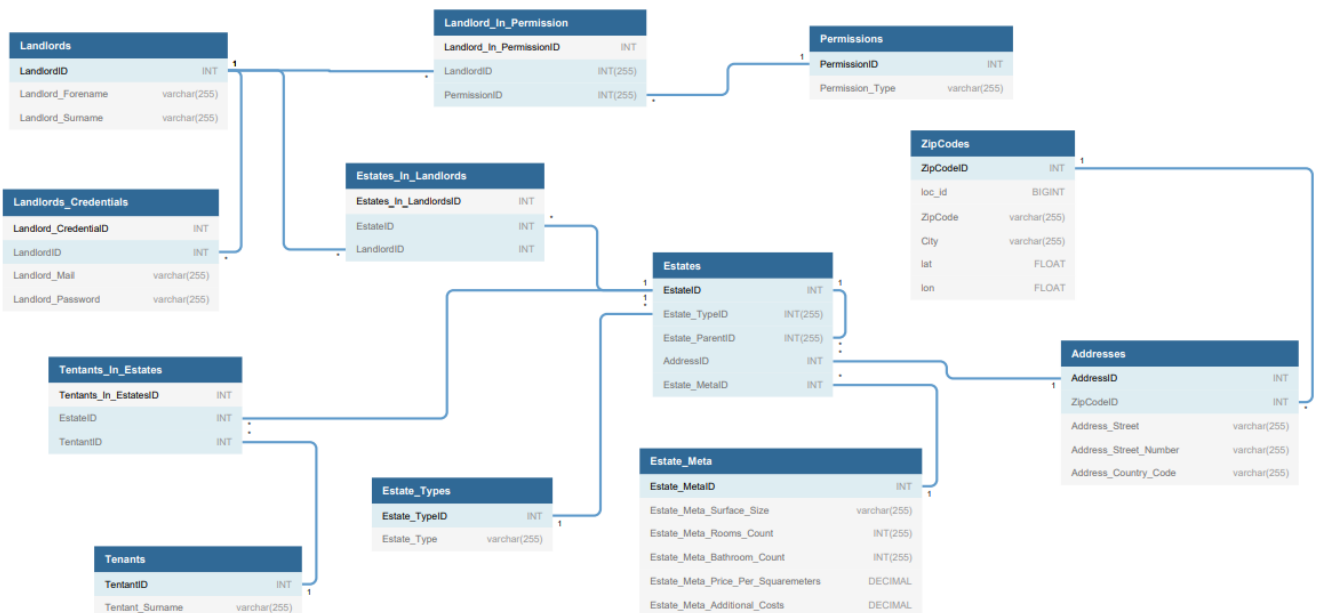


A3 Datenbankmodell semantisch und logisch

A3.1 Logisch



A3.2 Semantisch

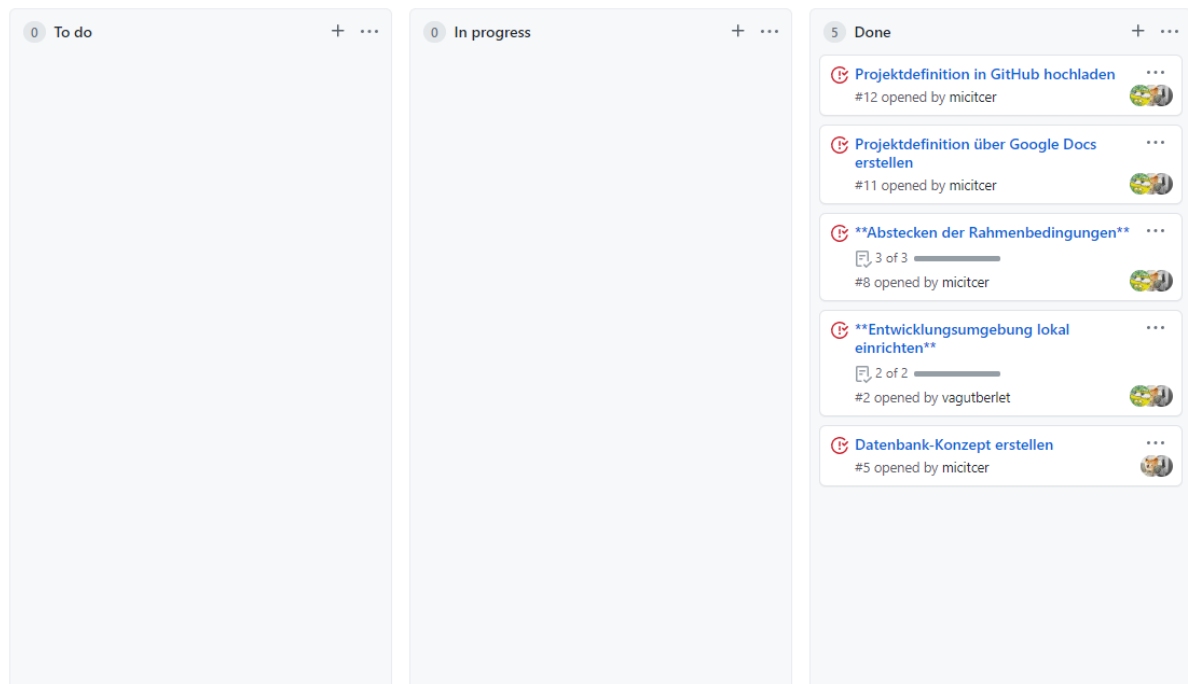


A4 Sprints

A4.1

IMS-Sprint KW 43

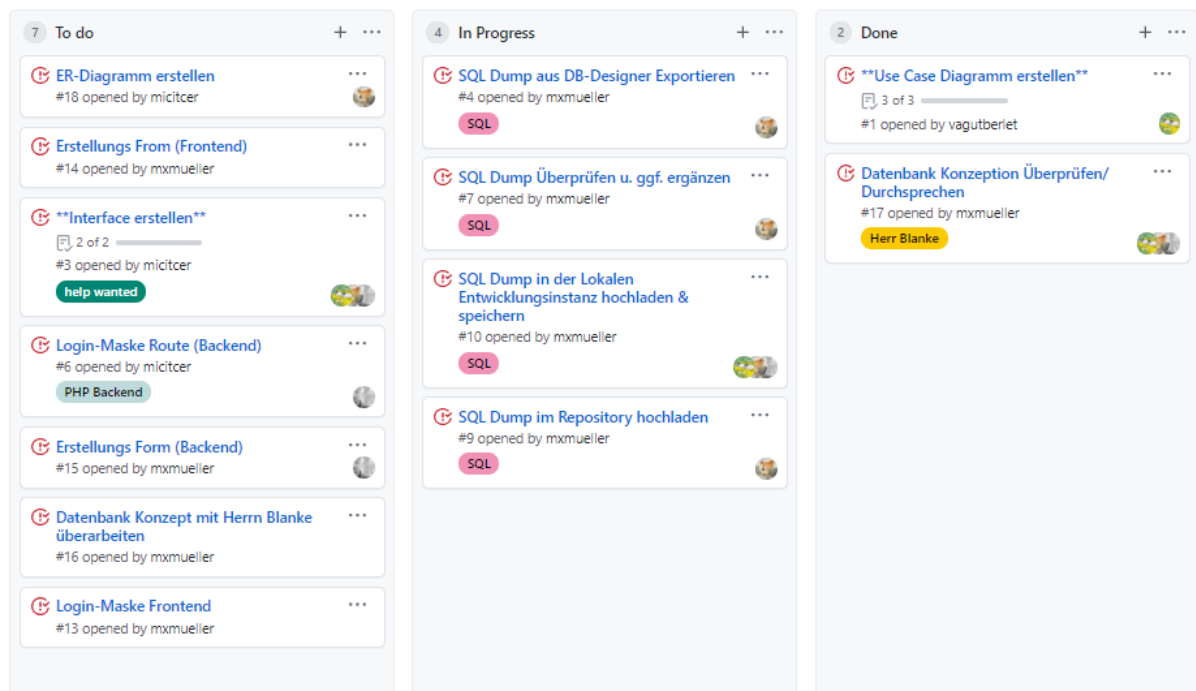
Updated on 17 Nov 2020



A4.2

IMS-Sprint KW 46

Updated on 1 Dec 2020



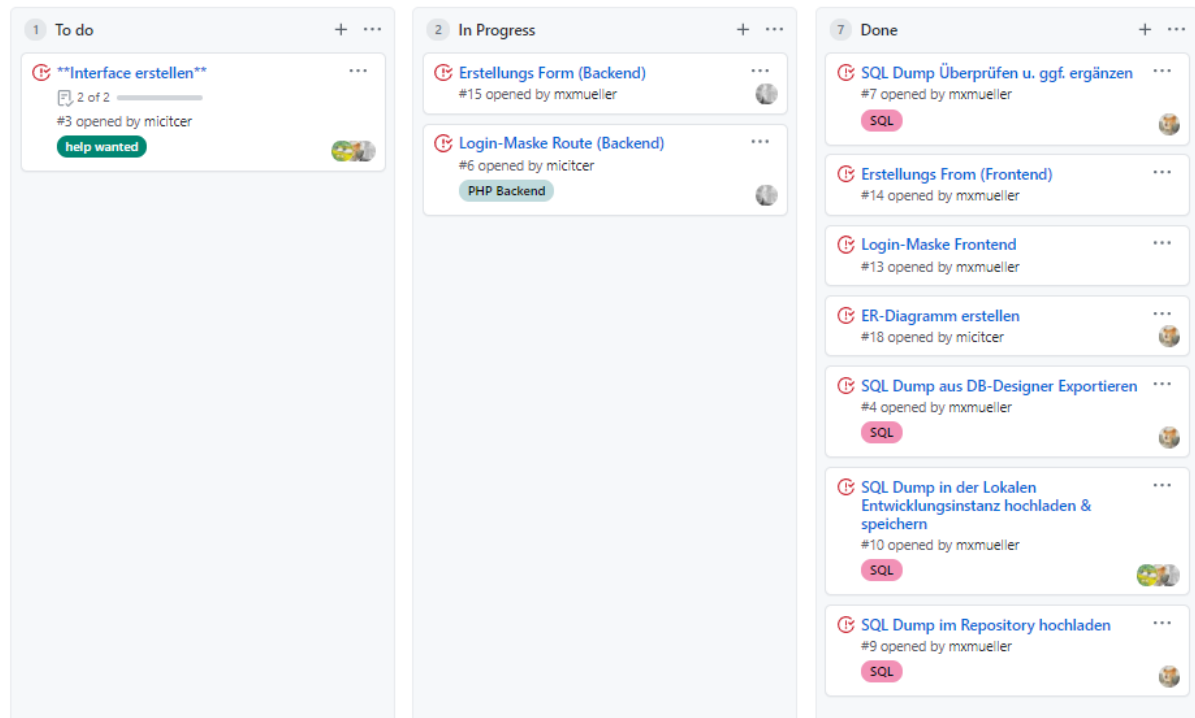
Datenbankprojekt aus dem Schulunterricht der Berufsschule Kempten 1 im Fach Anwendung u. Programmierung.

Sourcecode: <https://git.io/JYfz5>

A4.3

IMS-Sprint KW 48

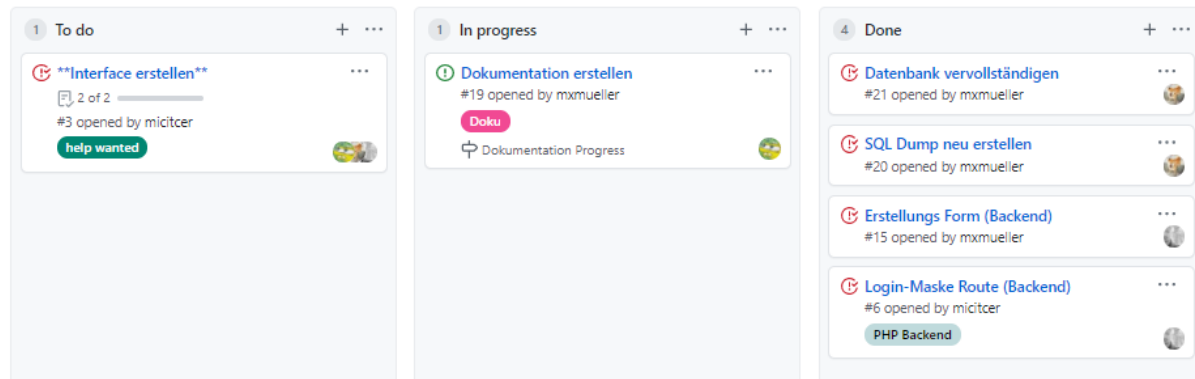
Updated on 4 Dec 2020



A4.4

IMS-Sprint KW 2

Updated on 15 Jan



A4.5

Datenbankprojekt aus dem Schulunterricht der Berufsschule Kempten 1 im Fach Anwendung u. Programmierung.

Sourcecode: <https://git.io/JYfz5>

IMS-Sprint KW 5

Updated 21 days ago

1 To do

- **Interface erstellen**** #3 opened by micitcer **help wanted**

3 In Progress

- Dokumentation - Screenshots** #25 opened by mxmueller **documentation**
- Dokumentation - Datenbankmodelle** #27 opened by mxmueller **documentation**
- Dokumentation - Projektbeschreibung** #28 opened by mxmueller **documentation**

5 Done

- Live Web Erstellung** Added by mxmueller
- Dokumentation - Vorlage anlegen** #29 opened by mxmueller **documentation**
- Zip Codes Recherche** #30 opened by mxmueller
- Get Request - Anfangen** #23 opened by mxmueller
- Request Insert - Fertigstellen** #22 opened by mxmueller

A4.6

IMS-Sprint KW 09

Updated 20 days ago

2 to do

- Development Head Of Development** Added by micitcer
- Projektpräsentation erstellen** #31 opened by micitcer **presentation**

3 In progress

- Dokumentation - Screenshots** #25 opened by mxmueller **documentation**
- Dokumentation - Datenbankmodelle** #27 opened by mxmueller **documentation**
- Dokumentation - Projektbeschreibung** #28 opened by mxmueller **documentation**

1 Done

- **Interface erstellen**** #3 opened by micitcer **help wanted**

A4.7

IMS Sprint KW 12

Updated yesterday

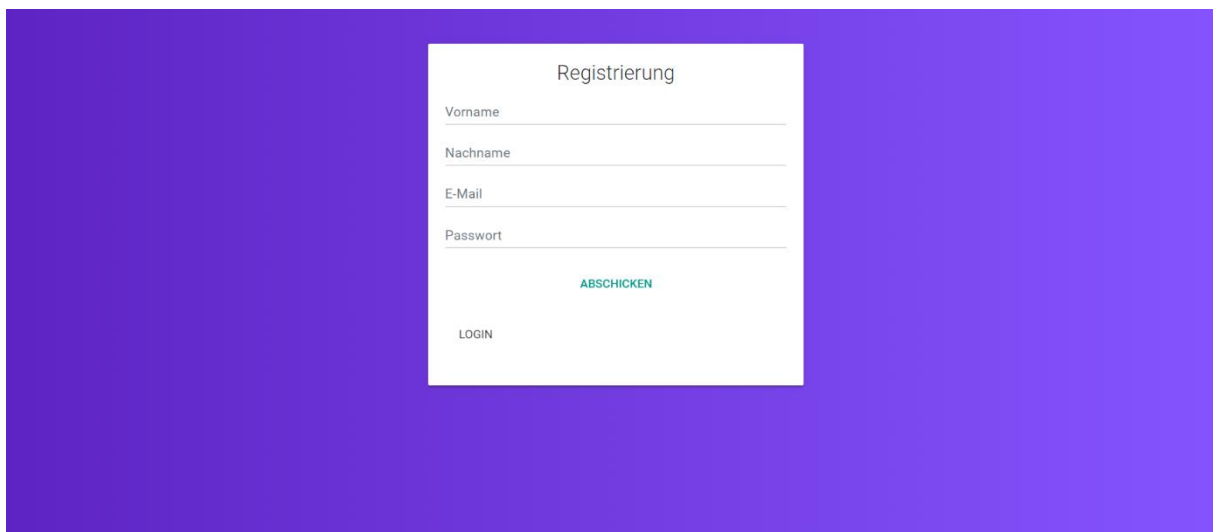
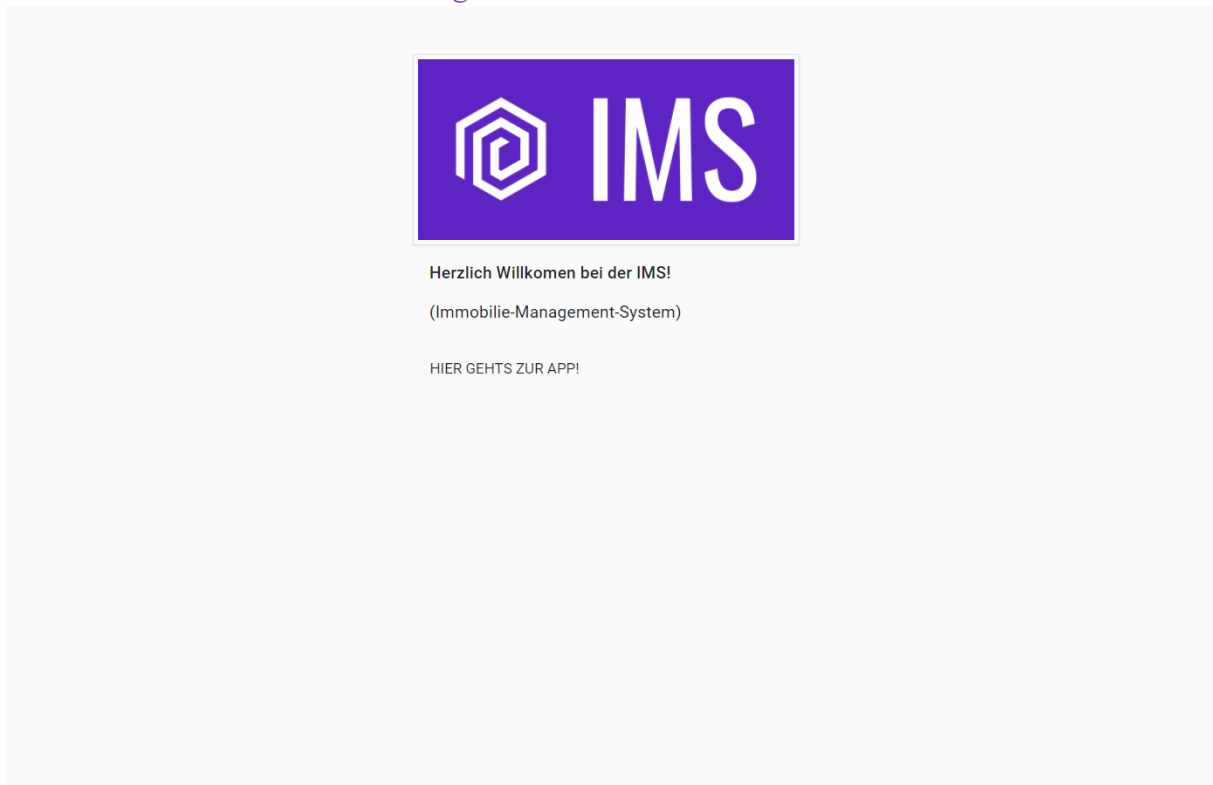
The Kanban board displays tasks across three columns: To do, In progress, and Done.

- To do (1 task):**
 - Testing und Bugfixing** (#32 opened by micitcer)
- In progress (6 tasks):**
 - Dokumentation - Datenbankmodelle** (#27 opened by mxmueller) with a **documentation** label and **Dokumentation Progress** status.
 - Dokumentation erstellen** (#19 opened by mxmueller) with a **Doku** label and **Dokumentation Progress** status.
 - Dokumentation - Screenshots** (#25 opened by mxmueller) with a **documentation** label and **Dokumentation Progress** status.
 - Dokumentation - Projektbeschreibung** (#28 opened by mxmueller) with a **documentation** label and **Dokumentation Progress** status.
 - Projektpräsentation erstellen** (#31 opened by micitcer) with a **presentation** label.
 - Dokumentation - Projekt Aufgabe/ Funktionen erklären** (#26 opened by mxmueller) with **Dokumentation Progress** status.
- Done (0 tasks):**

Datenbankprojekt aus dem Schulunterricht der Berufsschule Kempten 1 im Fach Anwendung u. Programmierung.

Sourcecode: <https://git.io/JYfz5>

A5 Screenshots der Anwendung




Registrierung

E-Mail bereits vorhanden!

ABSCHICKEN


LOGIN



NEUE IMMOBILIE ANLEGEN ALLE IMMOBILIEN ANZEIGEN SUCHE

Herzlich Willkommen:

Max Müller




Angelegte Immobilien:

1


Login

LOGIN

REGISTRIERUNG


NEUE IMMOBILIE ANLEGEN ALLE IMMOBILIEN ANZEIGEN SUCHE

id	Postleitzahl	Straße	Adresse	Funktionen
47	87600	Marschowitz Str.	18	DETAILANSICHT / BEARBEITEN


NEUE IMMOBILIE ANLEGEN ALLE IMMOBILIEN ANZEIGEN SUCHE

Immobilie: 47

Adresse

Postleitzahl: Stadt:

Straße: Hausnummer:

Immobilie

Immobilien Typ:

Spezifikation

Quadratmeter:


Zimmer:

Badezimmer:

Preis Pro m²:

Nebenkosten:

[SUBMIT](#)

 [NEUE IMMOBILIE ANLEGEN](#) [ALLE IMMOBILIEN ANZEIGEN](#) [SUCHE](#)

Neue Immobilie Anlegen

Adresse

Postleitzahl:

87600

Stadt:

Kaufbeuren

Straße:

Lorem Ipsum

Hausnummer:

12

Immobilie

Immobilien Typ:

Wohnung

Quadratmeter:

Ausgewählt: 4290m²

MANUELLE EINGABE MÖGLICH

Zimmer:

Ausgewählt: 15

Badezimmer:

Ausgewählt: 6

Preis Pro m²:

Ausgewählt: 100€

MANUELLE EINGABE MÖGLICH

Nebenkosten:

Ausgewählt: 230€

MANUELLE EINGABE MÖGLICH

[SUBMIT](#)

A6 Klasse: Beispielklasse aus dem Projekt

```

<?php

class buildEstateQuery
{
    function __construct()
    {
        $this->dashboard_fallback = 'Noch kein Objekt eingepflegt!';
        $this->detail_fallback = 'Bitte zuerst Objekte einpflegen!';
        $this->edit_exist_fallback = 'Der angefragte Eintrag ist nicht vorhanden!';
        $this->edit_user_fallback = 'Sie sind nicht berechtigt diesen Eintrag zu sehen!';
        $this->fallback = 'Unerwarteter Fehler';
    }

    private function db()
    {
        $servername = "bngwcgyjxquzsogdpgjv-mysql.services.clever-";
        $username = "ubz7t8d8aegwhopxg";
        $password = "GkgWQRKJzxDrFquEIQp6";
        $database = "bngwcgyjxquzsogdpgjv";
        // Create connection
        return new mysqli($servername, $username, $password, $database);
    }

    public function dashboard()
    {
        $connection = $this->db();
        $logged_in_user = $_SESSION['user_id'];

        $sql = "SELECT * FROM estates_in_landlords WHERE LandlordID = $logged_in_user";
        $result = mysqli_query($connection, $sql);

        $counted_result = $result->num_rows;

        if ($counted_result == 0) {
            return $this->dashboard_fallback;
        } else {
            return $counted_result;
        }
    }

    public function edit($logged_in_user_id, $estate_request_id)
    {
        $connection = $this->db();

        // check if estate request exist
        $exist_query = "SELECT * FROM estates WHERE estates.estateID = $estate_request_id";
        $exist_query_result = mysqli_query($connection, $exist_query);

        if ($exist_query_result->num_rows == null) {
            return $this->edit_exist_fallback;
        }

        // check if user owns request
        $ownership_query = "SELECT LandlordID FROM estates_in_landlords WHERE EstateID = $estate_request_id";
        $ownership_query_result = mysqli_query($connection, $ownership_query);

        if ($ownership_query_result->num_rows > 0) {
            while ($ownership_row = $ownership_query_result
                ->fetch_assoc()
            ) {
                if ($ownership_row['LandlordID'] != $logged_in_user_id) {
                    return $this->edit_user_fallback;
                }
            }
        } else {
            return $this->fallback;
        }

        $mother_query =
            "SELECT *
            FROM estates
            JOIN addresses a on estates.AddressID = a.AddressID
            JOIN estate_meta em on estates.Estate_MetaID = em.Estate_MetaID
            WHERE estates.EstateID = $estate_request_id";

        $mother_query_result = mysqli_query($connection, $mother_query);
        return $mother_query_result;
    }
}

```

Datenbankprojekt aus dem Schulunterricht der Berufsschule Kempten 1 im Fach Anwendung u. Programmierung.

Sourcecode: <https://git.io/JYfz5>

```

public function overview()
{
    if (session_status() === PHP_SESSION_NONE) {
        session_start();
    }
    if (isset($_SERVER['HTTPS']) && $_SERVER['HTTPS'] === 'on')
        $url = "https://";
    else
        $url = "http://";

    $url .= $_SERVER['HTTP_HOST'];
    $url .= $_SERVER['REQUEST_URI'];

    $host = parse_url($url, PHP_URL_SCHEME) . '://' . parse_url($url, PHP_URL_HOST);
    $root = 'immobilien-management-system';

    $connection = $this->db();
    $logged_in_user = $_SESSION['user_id'];

    $sql = "SELECT * FROM estates_in_landlords WHERE LandlordID = $logged_in_user";
    $result = mysqli_query($connection, $sql);

    if ($result->num_rows > 0) {
        while ($row = $result->fetch_assoc()) {

            $search = $row['EstateID'];

            $join = "SELECT
            estate_types.Estate_Type,
            addresses.ZipcodeID,
            addresses.Address_Street,
            addresses.Address_Street_Number,
            estates.EstateID
            FROM estates
            INNER JOIN estate_types ON estates.EstateTypeID=estate_types.EstateTypeID
            INNER JOIN addresses ON estates.AddressID=addresses.AddressID
            WHERE estates.EstateID = $search";

            $join_results = mysqli_query($connection, $join);

            if ($join_results->num_rows > 0) {

                while ($join_entry = $join_results->fetch_assoc()) {

                    $path = "sites/estate-detail/sites.php? " . $join_entry['EstateID'] . "";
                    $link = $host . '/' . $root . '/' . $path;

                    $button = "<a class='btn btn-outline-info'
                    href='\" . $link . \"'>
                    Detailansicht / Bearbeiten
                    </a>";

                    $table = "<tr>
                    <th scope='row'>\" . $join_entry['EstateID'] . "\"</th>
                    <td>\" . $join_entry['ZipcodeID'] . "\"</td>
                    <td>\" . $join_entry['Address_Street'] . "\"</td>
                    <td>\" . $join_entry['Address_Street_Number'] . "\"</td>
                    <td>\" . $button . "\"</td>
                    </tr>";

                    echo $table;
                }
            }
        }
    } else {
        echo $this->detail_fallback;
    }
}
}

```

Datenbankprojekt aus dem Schulunterricht der Berufsschule Kempten 1 im Fach Anwendung u. Programmierung.

Sourcecode: <https://git.io/JYfz5>