

BACHELOR SOFTWARE ENGINEERING UND VERNETZTE SYSTEME

PRAXISPROJEKTE 2022

Fachhochschul-Bachelorstudiengang

Software Engineering und vernetzte Systeme



„Egal, in welchen Bereich des Lebens man blickt, Software ist ein wesentlicher Bestandteil jeder Innovation. Unsere Absolvent:innen sind so der Schlüssel zum Erfolg neuer Lösungen.“

Franz Knipp, Studiengangsleiter

Das Bachelorstudium verfolgt das Ziel, Software-Entwickler:innen auszubilden, die den Entwicklungsprozess von der initialen Idee bis zur Inbetriebnahme verstehen und auf Basis aktueller Technologien umsetzen können. Neben der Vermittlung des theoretischen Wissens bildet die praktische Umsetzung einen wichtigen Bestandteil der Ausbildung.

„Softwareentwicklung ist mehr als Programmieren. Organisations- und Kommunikationskompetenzen machen ein Projekt erst erfolgreich.“

Christian Büll, Departmentleiter



Im vierten Semester beweisen die Studierenden die erworbenen Kompetenzen im Rahmen eines mehrmonatigen Projekts. Neben den technischen Fertigkeiten der Softwareentwicklung und -architektur sind die Teams auch für das Management des Projekts und die Kommunikation mit der Auftraggeber:in zuständig.

Inhalt

1	Praxisprojekte	2
1.1	Entwicklung eines Tools zur Organisation von Curricula	2
1.2	Modernisierung einer bestehenden IoT-Plattform bei FIB	5
1.3	Inventarmanagement	8
1.4	Pegelhub - Die zentrale Datendrehscheibe für Pegeldata	11
2	Projektmitglieder – BSWE Jahrgang 2020	14
3	Projektbetreuer	16
4	Verzeichnisse	17
5	Impressum	19

Entwicklung eines Tools zur Organisation von Curricula



Projektauftraggeber:in: Department für Informationstechnologie FH-Burgenland GmbH
Projektbetreuer:in: Dipl.-Ing. Franz Knipp
Projektmitglieder: Lisa Allacher, Helmut Tischler, Günter Ulreich

Das Department für Informationstechnologie an der Fachhochschule Burgenland GmbH ist zuständig für Forschung und Lehre im Bereich der Informatik und ist stets bemüht, aktuelle Lehrinhalte wie Internet of Things, Blockchain, Data Science und Web Development Methoden einzubringen. Die Organisation der Lehre, sowie die Koordination der einzelnen Curricula, ist jedoch mit einem gewissen Aufwand verbunden, da diese hauptsächlich manuell erfolgen. Daraus ergibt sich auch eine Fehleranfälligkeit und ein erhöhter Zeitaufwand, damit diese wieder korrigiert werden können. Die verschiedenen Dokumente und die aufwendige manuelle Organisation sollten im Zuge des Projektes durch ein Tool abgelöst werden, welches die Planung der Curricula zentralisiert und vereinfacht.

Das Projekt für die Entwicklung des Curricular Creators wurde im Zuge einer Lehrveranstaltung von Studierenden der FH Burgenland umgesetzt, unter Anleitung von Herrn Dipl.-Ing. Franz Knipp, der als Betreuer während des gesamten Semesters zur Verfügung stand. Dabei konnten die Studierenden nicht nur ihre bereits erworbenen Fähigkeiten in einem realen Projekt vertiefen und gleichzeitig die wichtige Komponente des Projektmanagements miteinbeziehen, sondern auch gemäß den Anforderungen die zu verwendeten Technologien selbst wählen und ihre Kreativität in der Entwicklung freien Lauf lassen. Dabei wurde ein großer Wert auf das anstehende Projektmanagement und Feature Requirement gelegt. Dies waren wichtige Voraussetzungen für den Erfolg des Projektes, die wiederum ebenso den tatsächlichen Alltag in

der IT-Branche widerspiegeln sollten. Damit bekamen die Studierenden bereits einen Einblick in die Erwartungen und Anforderungen eines echten Projektes.

Projektziele

Der Auftrag war die Entwicklung eines Tools, das die Organisation und Planung von Curricula erleichtern sollte. Die allgemeinen Anforderungen waren, dass es möglich sein sollte, in Zukunft die Daten auf einer Plattform zu erfassen und bearbeiten zu können. Weiters sollte es die Möglichkeit geben, dass mehrere Bearbeiter:innen auf diese Plattform zugreifen können, um die Curricula einzusehen und zu bearbeiten. Besonders wichtig war es, eine Funktion zur Verfügung zu stellen, welche es erlauben würde, mehrere Versionen

beziehungsweise Studienjahre einzusehen und zu bearbeiten. Im Zuge des Projektes wurden diese allgemeinen Anforderungen anschließend in Absprache mit dem Ansprechpartner genauer definiert und ausgearbeitet. Davon ausgehend konnten die Projektmitglieder schließlich zusammen mit dem Projektbetreuer eine Auswahl an Tools zusammenstellen, die für die Entwicklung geeignet waren. Im Verlauf des Projektes wurde dann während des Requirement-Engineering eine detaillierte Featureliste erstellt. Einige Features, wie Login, konnten dabei auf bereits existierende Tools wie Github ausgelagert werden. Daraus ergaben sich drei zentrale Herausforderungen: die Anbindung der Github-API, die Entwicklung eines Editors, der die Curriculum-Dateien als Markdown-Dateien erstellen sollte und die Aufbereitung der Versionen auf der zu entwickelnden Plattform.

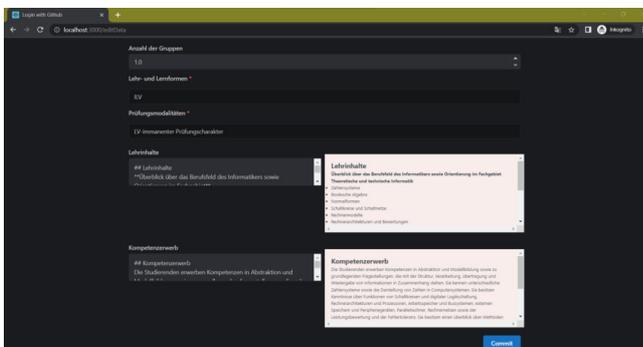


Abbildung 1: Bearbeitung einer Lehrveranstaltung

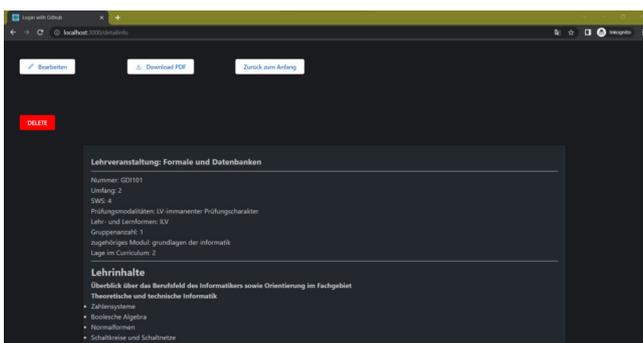


Abbildung 2: Detailansicht Lehrveranstaltung

Github als zentrale Schnittstelle

Durch die Github API konnten folgende Features ausgelagert werden wie das Login, die Verwaltung von Rechten und Zugriff, das Speichern und Wiederabrufen aller notwendigen Daten im Markdown-Format sowie die Verwaltung unterschiedlicher Versionen von verschiedenen Modulen und Lehrveranstaltungen. Dadurch konnte sich das Projektteam auf die vorhandenen Ressourcen für UX/UI-Design, Aufbereitung, Bearbeitung und Erstellung im Frontend, die Anbindung an GitHub mittels der API sowie Anforderungen aus dem Projektmanagement konzentrieren. Weitere Tools und Technologien, die verwendet wurden, waren React, Mantine, Next.js und Markdown.

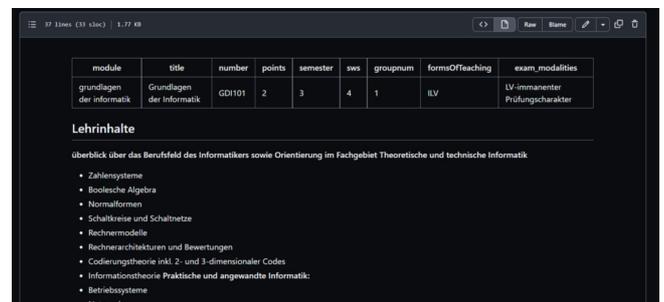


Abbildung 3: Ansicht eines Markdownfiles auf Github

Projektverlauf

Bevor das Projekt offiziell gestartet wurde, traf sich das Team, um Regeln aufzustellen und die für das Projekt notwendigen Unterlagen zu erstellen. Dazu gehörte ein Brainstorming-Panel, das in Miro abgebildet wurde, die Einrichtung eines Workflows in Jira und die Einteilung der verschiedenen Rollen im Team. Lisa Allacher übernahm als Projektteammitglied die Verantwortung für UX/UI-Design, Frontendentwicklung und Projektorganisation, während Helmut Tischler zum Projektleiter ernannt wurde und für Backend, API und Projektorganisation zuständig war. Günter Ulreich war als

Projektteammitglied für die Protokollierung, Jira, Softwaretests und allfällige Programmierarbeiten verantwortlich. Für alle notwendigen Meilensteine des Projektes wurden in Jira Epics erstellt, die dann wieder in Story-Tickets oder Defect-Tickets abgebildet wurden. Das Projekt wurde offiziell mit einem Kickoff-Meeting gestartet, bei dem die ersten Funktionen des Programms festgelegt wurden. Im Verlauf des Projekts gab es regelmäßige Meetings und Treffen, um den Fortschritt zu besprechen. Franz Knipp stand als Projektbetreuer stets zur Verfügung, um Fragen zu beantworten und bei Problemen zu helfen. Im Rahmen des Projekts wurden auch Projektmanagement-Tools und Frameworks im Unterricht besprochen, um das Projekt in die richtige Richtung zu lenken.

Projektergebnis

Aus dem Projekt entstand ein erster Prototyp eines Curriculum-Creators, der die wichtigsten Features, wie zu Beginn des Semesters besprochen wurde, beinhaltet, sowie eine graphische Benutzeroberfläche, die mithilfe von React und Mantine umgesetzt wurde. Die Daten können in einem Editor erstellt und bearbeitet werden. Die Anbindung an GitHub mit der API wurde ebenso erfolgreich umgesetzt.

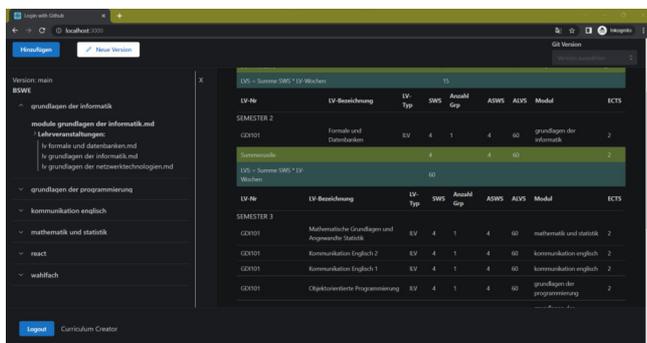


Abbildung 4: Menu Homepage Curriculum Creator

Herausforderungen

Eine Herausforderung während des Projekts war der Login bei GitHub. Dies gestaltete sich als schwierig, da dazu ein Schlüssel von GitHub gespeichert werden musste, der mit einer Antwort während des Login-Prozesses gesendet wurde. Dieser Schlüssel war erforderlich, um mit der GitHub-API kommunizieren zu können, konnte jedoch nur über ein Backend gespeichert werden. Eine weitere Herausforderung war die Entwicklung der verschiedenen API-Abfragen, die einige Zeit in Anspruch nahmen und erst spät abgeschlossen werden konnten. Eine weitere Herausforderung war das Zeitmanagement, das durch das Heranziehen eines Jira-Plugins für das Schätzen der Tickets gelöst werden konnte, um sämtliche zeitliche Ressourcen aller Team-Mitglieder gerecht aufzuteilen und dabei gleichzeitig alle Aufgaben im Rahmen des Projektverlaufes zu lösen.

Fazit

Im Rahmen des Curriculum Creator Projektes wurden unterschiedliche Qualifikationen von den Projektmitgliedern gefordert, um die vorgegebenen Ziele zu erreichen. Ein gut herausgebildetes Projektmanagement, sowie umfangreiche Kommunikation der Mitglieder waren notwendig, um eine Umsetzung des Projektes zu gewährleisten, die im Rahmen der vorhandenen Ressourcen möglich war. Der entwickelte Prototyp kann als „Proof of Concept“ für eine weitere Entwicklung verwendet werden. Alle Mitglieder des Projektteams konnten dadurch wertvolle Erfahrungen sammeln, die für den weiteren Fortgang als Software Developer essentiell sind, damit auch bei anderen Projekten die erworbenen Fähigkeiten eingesetzt und vertieft werden können.

Autor:innen: Lisa Allacher, Helmut Tischler

Modernisierung einer bestehenden IoT-Plattform bei FIB



Modernisierung einer bestehenden IoT-Plattform

Projektauftraggeber:in: Wirtschaftsagentur Burgenland Forschungs- und Innovations GmbH
Projektbetreuer:in: Dipl.-Ing. Wolfram Rinke
Projektmitglieder: Sophie Wiedemann, Michael Pfeiffer, Andreas Kappel und Daniel Krancz

Eine erfolgreiche IoT-Landschaft zeichnet sich durch eine robuste und optimierte IT-Systemlandschaft aus. An solch eine Landschaft sind zahlreiche Geräte angebunden, welche Daten miteinander austauschen. Bei der Entwicklung und Konzeption müssen jedoch sicherheitsrelevante und datenschutztechnische Aspekte beachtet werden. Zu den Qualitätsmerkmalen zählen Kompatibilität zu anderen Systemen sowie IoT-Quellen und die Erweiterbarkeit des Systems selbst. Um den wachsenden Anforderungen gerecht zu werden, müssen bereits bei der Planung etliche Architekturthemen beachtet werden. Entwickler:innen stehen vor der Herausforderung, all diese Kompetenzen abzudecken sowie bei der Entwicklung zu berücksichtigen, um eine effektive IoT-Plattform entwickeln und dem Kunden bereitstellen zu können.

Die Anforderung war es, die bereits vorhandene und historisch gewachsene IoT-Plattform abzulösen. Die vorhandene Datenbank mit existierenden Daten von Energieerzeugern und -verbrauchern, die bereits im Einsatz war, sollte vollständig integriert werden. Die Datenbank diente als Datenpool für verschiedene Projekte. Eine neue Plattform sollte als eine Datendrehscheibe fungieren und Schnittstellen an Produzenten und Konsumenten von IoT-Daten bereitstellen. Die abgesicherten Schnittstellen sollen von Smart Meter, PV-Anlagen oder Batteriespeicher genutzt werden, um Daten in die Datenbank einzuspeisen. Die bereits

vorhandene Plattform wurde in der Programmiersprache Python entwickelt und konnte bis zu der Zeit nur lesende Operationen abdecken.

Auftraggeber: Wirtschaftsagentur Burgenland Forschungs- und Innovations GmbH

Die Wirtschaftsagentur Burgenland Forschungs- und Innovations GmbH oder kurz FIB ist ein Tochterunternehmen der burgenländischen Wirtschaftsagentur mit Sitz in Güssing (nachstehend Auftraggeber). Sie beschäftigt sich mit der wirtschaftsnahen angewandten Forschung und fokussiert sich vor allem auf erneuerbaren Energien und dessen

(Er-)Forschung. Der Auftraggeber arbeitet des Öfteren mit Forscher:innen von der Forschung Burgenland und Student:innen von der FH Burgenland zusammen. Der Ansprechpartner auf Kunden-seite war Dipl.-Ing. Georg Supper (Project Manager), der mit seinem Prozesswissen den Projektverlauf bedeutsam beeinflusst hat.

Projektverlauf

Das Projektteam bestand aus Sophie Wiedemann (Projektmanagement), Michael Pfeiffer (Projektmanagement), Andreas Kappel (Entwicklung/Testing) und Daniel Krancz (Entwicklung/Testing), allesamt Studierende des Bachelorstudienganges Software Engineering und vernetzte Systeme an der FH Burgenland (nachstehend Projektteam). Fachlich wurde das Projekt von Dipl.-Ing. Wolfram Rinke (Senior Lecturer) betreut, da er unter anderem bereits Verbindungen zu dem Auftraggeber aufweisen konnte (nachstehend Projektbetreuer).

Das Projektteam arbeitete eigenständig und folgte dem agilen Vorgehensmodell SCRUM. Neben Berücksichtigung agiler Werte, gab es die Vorgabe, drei Meilensteine zu definieren, um dem Auftraggeber einen übersichtlichen Projektverlauf präsentieren zu können. Damit der Fortschritt des Projekts dokumentiert und verfolgt werden konnte, wurde nach jedem Sprint ein Zwischenbericht erstellt und das Inkrement präsentiert. Es wurden zahlreiche User Stories und Features definiert, die in Jira-Tickets umgewandelt und auf einem Board visualisiert wurden. Diese Tickets wurden entsprechend priorisiert und in kleinere Aufgaben unterteilt, um sie im nächsten Sprint bearbeiten zu können. Das Projektteam konnte somit bei jedem Sprint-Planning die wichtigsten Tickets auswählen und sich auf deren Umsetzung konzentrieren. Nach jedem Sprint fand eine Sprint-Retrospektive statt, in der das Projektteam über die positiven und

negativen Ereignisse des vergangenen Sprints reflektierte und Verbesserungsmöglichkeiten diskutierte. Dank kurzer Sprintzyklen konnte das Projektteam auf diese Erfahrungswerte zurückgreifen und sofort einsetzen.

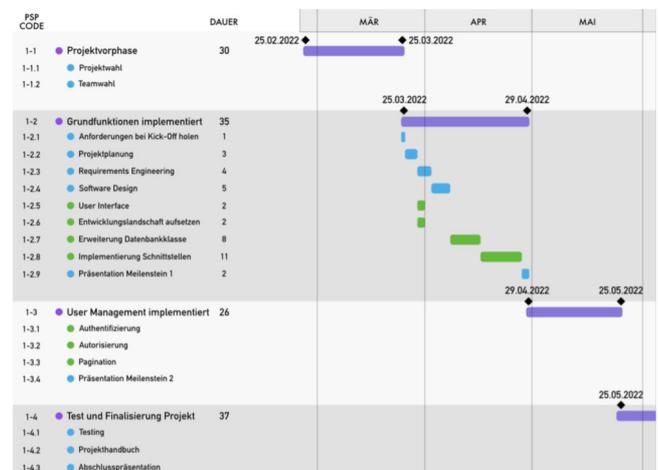


Abbildung 5: Ausschnitt aus dem Projektbalckenplan

Die Entwicklung der neuen IoT-Plattform wurde umgesetzt, indem das Projektteam die Programmiersprache Python nutzte. Nach der Implementierung einer ersten Klasse als Prototyp, die die Create, Read, Update, Delete und Query (CRUDQ)-Methoden für eine Datenbanktabelle abdeckte, wurde diese in eine generische Klasse umprogrammiert. Durch diese Entwicklung konnte sichergestellt werden, dass unter anderem die Clean Code Rule Don't Repeat Yourself (DRY) bei der Erstellung der weiteren Klassen eingehalten werden konnte. Neben dieser Regel wurden weitere sowohl anerkannte Standards, als auch ungeschriebene Gesetze eingehalten, sowie auf bekannte Design Patterns gesetzt. Die einzelnen Methoden können entweder über die entsprechenden Endpunkte aufgerufen oder mittels einer intuitiven Benutzeroberfläche von Benutzern getestet werden. Die Benutzeroberfläche wurde gemäß den Anforderungen des Auftraggebers an die OpenAPI Spe-

cification angelehnt, um eine einfache und benutzerfreundliche Visualisierung der Endpunkte zu ermöglichen.

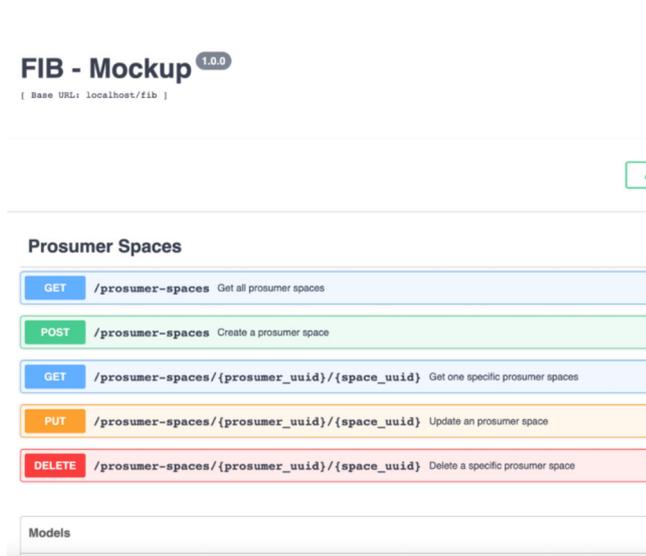


Abbildung 6: Benutzeroberfläche in Form eines SwaggerUIs

Die Schnittstellen wurden mittels OAuth2 abgesichert, um sicherzustellen, dass nur autorisierte Personen Zugriff auf die APIs haben. Nach erfolgreicher Authentifizierung mussten Konsumenten bei jedem Request einen vom Server ausgestellten Token mitsenden, um ihre Identität zu verifizieren. Eine integrierte Benutzerverwaltung ermöglicht Administratoren, neue Benutzer anzulegen und somit den Datenaustausch zu erleichtern. Das Berechtigungskonzept ermöglicht es, individuelle Benutzer oder Benutzergruppen auf die Schnittstellen zu berechtigen, wobei die Granularität auf einzelne CRUDQ-Methoden heruntergebrochen werden kann. Durch automatisierte Tests kann die Plattform nahtlos über alle Schnittstellen und ihrer CRUDQ-Methoden hinweg auf Fehlerpotenziale hin überprüft werden. Dank der Verwendung von generischen Klassen und Methoden war es einfach, weitere Schnittstellen und Datenbanktabellen zu integrieren und die Plattform zu

erweitern. Insgesamt setzte das Projektteam auf bewährte Praktiken, um die Plattform sicher, stabil und erweiterbar zu gestalten.

Projektergebnis

Die neu entwickelte IoT-Plattform wurde erfolgreich an den Kunden übergeben und soll die bisherige Anwendung ersetzen. Produzenten können damit IoT-Daten sowohl lesen, als auch in die Datenbank schreiben. Die generischen Implementierung macht es einfacher, verschiedene Produzenten wie IoT-Sensoren, Smart Meter, PV-Anlagen oder Batteriespeicher in die Plattform zu integrieren. Die neue Plattform geht über die ursprünglichen Anforderungen hinaus und bietet eine Authentifizierung und Autorisierungsfunktion. Durch ein Rollenkonzept können Zugriffe auf die einzelnen Schnittstellen von Konsumenten und Produzenten gezielt gesteuert und eingeschränkt werden. Das Projektteam hat sich auch um die Dokumentation der neuen Plattform gekümmert, sodass der Kunde mit einfachen Handbüchern, Tutorials und Schritt-für-Schritt-Anleitungen die Anwendung erweitern, in Betrieb nehmen und automatisiert testen kann. Dank dieser Anleitungen und der Dokumentation kann der Kunde schnell loslegen und die Plattform vollständig nutzen.

Fazit

Die Umsetzung einer IoT-Plattform, welche mehrere IoT-Quellen mit Schnittstellen versorgt, stellt zweifellos eine anspruchsvolle Herausforderung dar. Eine erfolgreiche Umsetzung erfordert nicht nur Wissen aus diesem Fachbereich, sondern auch ein tiefgreifendes Verständnis von Softwareentwicklung, Netzwerktechnik und IT-Sicherheit. Als Projektteam haben wir uns dieser Herausforderung gestellt und sind stolz darauf, dass wir diese gemeistert haben.

Autor: Daniel Krancz

Inventarmanagement



Projektauftraggeber:in: KBB - Kultur-Betriebe Burgenland GmbH
Projektbetreuer:in: Dipl.-Ing. Wolfram Rinke
Projektmitglieder: Isabella Zaby, Polina Shilkova, Stefan Gass

Eine effektive Inventarverwaltung ist für Unternehmen unerlässlich und gewährleistet reibungslose Abläufe und Kundenzufriedenheit. Sie ermöglicht die Kontrolle über den Lagerbestand, vermeidet Überbestände und senkt Kosten durch Reduzierung von Lagerkosten und Kapitalbindung. Bei den Kultur-Betrieben erfolgte die Inventarverwaltung bislang mittels dezentralisierten Listen und Tabellen. Dieses System sollte durch eine moderne, leicht zu bedienende Webanwendung abgelöst werden, die in Zusammenarbeit dreier Student:innen der Fachhochschule Burgenland entstand.

Die Kultur-Betriebe Burgenland GmbH ist die Plattform der kulturellen Vielfalt im Burgenland, bietet aber auch tolle Locations für Veranstaltungen aller Art. So sind die Museen und Galerien sowie Kultur- und Kongresszentren in Eisenstadt, Mattersburg, Raiding, Oberschützen, Güssing und Neuhaus am Klausenbach Orte für die gesamte Palette künstlerischer und kultureller Ausdrucksformen einerseits, andererseits Schauplatz für Feierlichkeiten, Firmenevents und die Tagungswirtschaft. Bekannt ist sie vor allem durch die Seefestspiele Mörbisch, die Schloss-Spiele Kobersdorf und das Liszt Festival Raiding, die jährlich tausende von kulturbegeisterten Menschen ins Burgenland locken, um dort das abwechslungsreiche Kulturangebot zu genießen und daran Teil zu haben.



Abbildung 7: Seefestspiele, Foto: © Jerzy Bin

Das Inventar der Kultur-Betriebe besteht einerseits aus Gegenständen, die sich aus der Geschäftstätigkeit ergeben, beispielsweise Ausstellungsstücke

der Museen oder Requisiten der Festivals, andererseits aus Gegenständen der Verwaltung, wie Computer oder Möbel. Die Verwaltung eben jener Gegenstände ist ein essentieller Aspekt, der einen reibungslosen Betriebsablauf gewährleisten soll. Da die anfangs sehr kleine Gesellschaft im Laufe der Zeit stetig wuchs und immer mehr kleinere Unternehmen dort unter einem Dach vereint wurden, wurden auch viele verschiedene Inventarverzeichnisse und -systeme in die Firma integriert. Diese Systeme sollen nun durch ein modernes, effizientes, aber gleichermaßen auch bedienerfreundliches Inventarverwaltungssystem abgelöst werden.

Projektverlauf

Die Webanwendung sollte im Rahmen eines agilen Softwareentwicklungsprojekts in mehreren Projektphasen entstehen. Die Startphase ist dabei wie bei jedem agilen Softwareentwicklungsprojekt entscheidend und soll sicherstellen, dass das Projekt erfolgreich verläuft. Einen wesentlichen Bestandteil der Startphase bildete das Requirements Engineering, bei dem die Anforderungen an das Produkt festgelegt wurden, sodass es später auch tatsächlich den konkreten Bedürfnissen der Benutzer:innen entspricht. Außerdem wurden die Epics, also die Meilensteine, ausgearbeitet und anhand von Use Cases, also Anwendungsfällen, User Stories, die man auch als Software-Anforderungen bezeichnen kann, und daraus wiederum Akzeptanzkriterien und einzelne Tickets oder Aufgabenpakete abgeleitet.

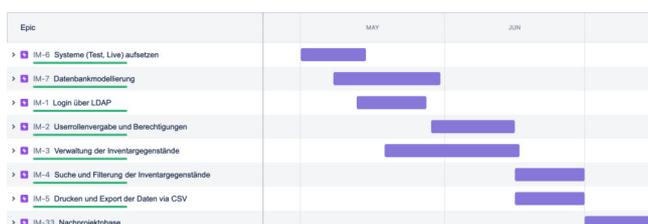


Abbildung 8: Epics und Roadmap

Die Entwicklung sollte dabei dem Kanban-Modell folgen. Kanban ist eine Methode zur Arbeitsorganisation, bei der alle Arbeitsschritte visualisiert werden, wodurch die Zusammenarbeit gefördert und es den Mitarbeitenden ermöglicht wird, den Arbeitsfortschritt auf einen Blick zu sehen. Zusätzlich wurden Designentwürfe und Diagramme erstellt und das Testkonzept ausgearbeitet.

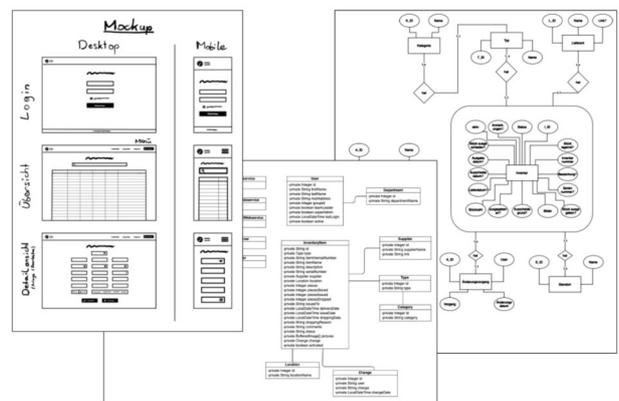


Abbildung 9: Mockup, UML- und ER-Diagramm

Erst nach Abschluss dieser Tätigkeiten wurde mit den Softwareentwicklungsarbeiten begonnen und das Produkt Schritt für Schritt, beginnend mit dem Anmeldesystem und der Benutzer:innenverwaltung, bis hin zum Inventarmanagement entwickelt, getestet und in mehreren Zyklen schrittweise veröffentlicht.

Projektergebnis

Aus dem Projekt entstand eine moderne, übersichtliche Webanwendung, die das Anlegen und Verwalten von Inventargegenständen durch unterschiedliche Benutzer:innen vereinfacht. Bestimmte wiederverwendbare Parameter wie Kategorien, Typen, Lieferanten und Standorte können dabei zunächst in einer eigenen Maske angelegt werden. Unter Berücksichtigung jener können in einer anderen Maske einzelne Gegenstände mit den verschiedensten Parametern erfasst werden.

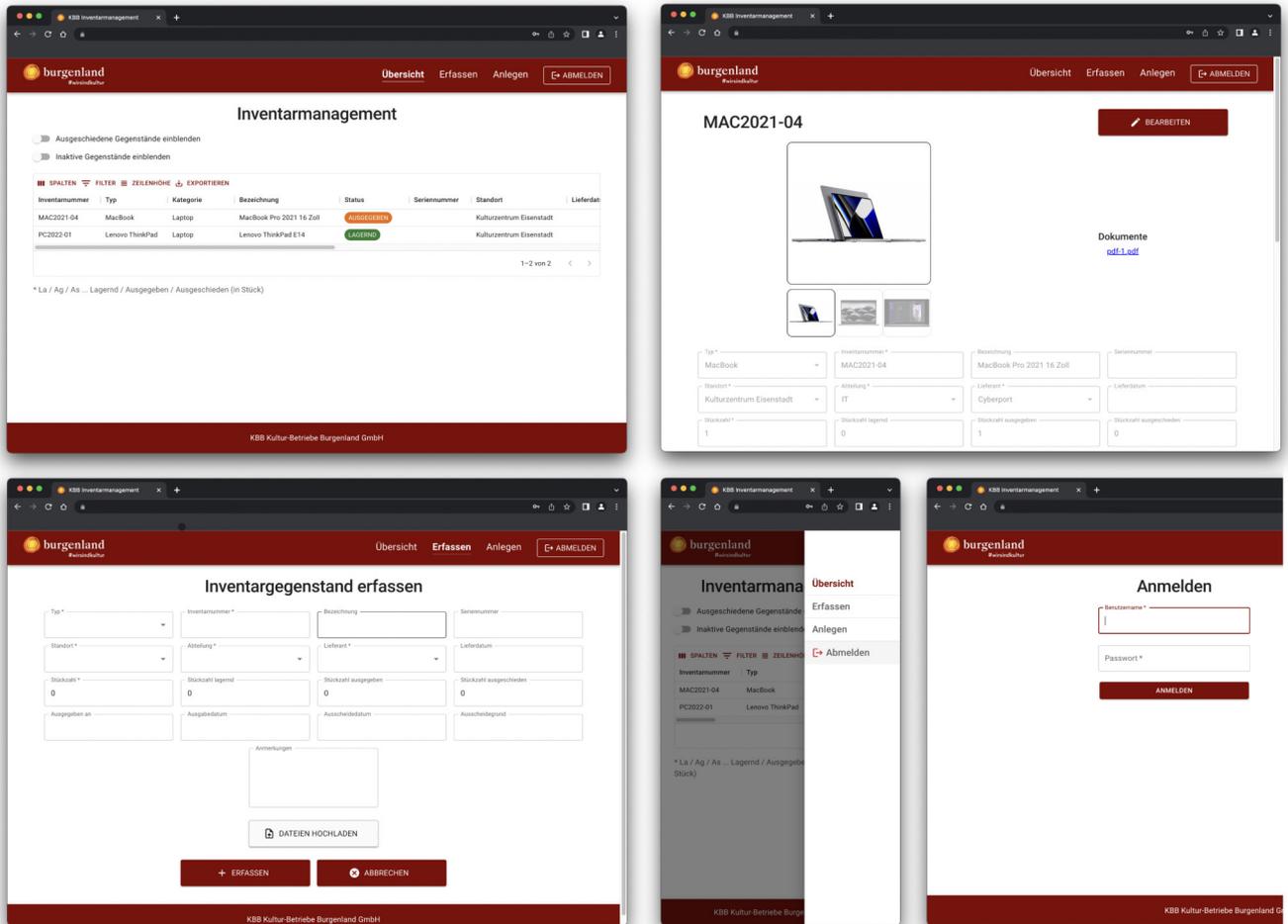


Abbildung 10: Screenshots der Webanwendung

Die Übersichtsmaske ermöglicht es, die Gegenstände einerseits in einer tabellarischen Übersicht, andererseits aber auch im Detail zu betrachten und zu bearbeiten. Außerdem können sie dort gefiltert und zur Weiterverarbeitung exportiert werden. Die Website, die sowohl auf Desktop-Computer, als auch auf Mobiltelefonen funktioniert, speichert die Daten zentral auf einem eigens dafür konfigurierten Server. Probleme, die durch dezentral geführte Listen in verschiedensten Versionen und Varianten, entstehen, gehören damit der Vergangenheit an.

Fazit

Die Zusammenarbeit mit den Kultur-Betrieben war nicht nur unglaublich lehrreich, sondern hat uns Studierenden auch große Freude bereitet. Es ist toll, im Rahmen der akademischen Ausbildung ein Projekt aus der Praxis bearbeiten zu dürfen. Zudem besitzt das Inventarverwaltungssystem großes Potential und kann im Rahmen der Lehrveranstaltung weiterentwickelt werden und so zum Wissenszuwachs weiterer Student:innen beitragen.

Autor: Stefan Gass

Pegelhub - Die zentrale Datendrehscheibe für Pegeldata



Projektauftraggeber:in: via Donau, Johannes Groisz
Projektbetreuer:in: Dipl.-Ing. Franz Knipp
Projektmitglieder: Tanja Buczkowski, Stefan Neuwirth, Marc Urschick

Bereits seit 2005 kümmert sich die via donau um die Erhaltung und Entwicklung der Wasserstraße Donau. Um die Schifffahrt an der Donau den modernen Gegebenheiten anzupassen und zu digitalisieren wurde eine Gesamtstrategie entwickelt, wobei die Entwicklung der erweiterbaren Pegelhub Kern-Architektur der erste Schritt der Digitalisierungsoffensive darstellt. Die mit moderner Microservice-Architektur geschaffene Lösung bietet die Möglichkeit zur zentralen Verwaltung sämtlicher Pegeldata entlang der Donau und wurde bereits in Hinblick auf die zukünftigen Pläne der Gesamtlösung mit zahlreichen Schnittstellen für Erweiterungen geplant.

Die via donau wurde mit dem Ziel des Erhaltes der Wasserstraße Donau gegründet. Im Zuge einer Digitalisierungsoffensive wurde ein mehrjähriges Projekt, welches aus verschiedenen Phasen besteht, ins Leben gerufen. Die erste Phase des Gesamtprojektes widmet sich der Planung, dem Aufbau und der fortlaufenden Konzeption einer zentralen Datendrehscheibe zur Verwaltung und Weitergabe der Pegeldata.

Projektziele

Für die Projektauftraggeber war es ein großes Anliegen eine moderne und offene Software-Architektur zu bekommen, welche auch die zukünftigen Anforderungen der weiteren Projektphasen bereits berücksichtigt und auch Lösungen für weitere Digitalisierungsansätze liefert. Das Projektteam, bestehend aus drei Studierenden der FH

Burgenland, sollte auf Basis modernster Technologien den Grundstein für die Digitalisierung der Pegeldata schaffen, sodass diese Messdaten einfach an berechnete Personen weitergegeben werden können. Ziel war es den Datenaustausch, welche bisher über proprietäre Protokolle und Listen erfolgte, vollständig zu digitalisieren.

Projektverlauf

Bereits vor dem Projektstart wurde das Projektteam über die zukünftigen Anforderungen und die geplante Gesamtstrategie der Digitalisierungsoffensive informiert, wodurch sehr schnell der Bedarf an moderner Software-Architekturen und guter Dokumentation, um weiteren Projektteams eine gute Basis zu liefern, erkannt. Das Projekt wurde in mehrere Phasen unterteilt, wobei zuerst das neue Datenmodell definiert werden musste, anschlie-

Während die Systemarchitektur sorgfältig geplant und zuletzt die Implementierungsarbeiten durchgeführt wurden. Im Anschluss an die Implementierungsarbeiten wurde die Software beim Projektauftraggeber ausgerollt und veröffentlicht.



Abbildung 11: Nussdorfer Wehr - Pegeldaten liefern wichtige Informationen zur Steuerung

Definition des Datenmodell

Aufgrund der Tatsache, dass es im Bereich des Pegelwesens keine genormten Schnittstellen, sondern diverse proprietäre Austauschformate mit unterschiedlichem Detaillierungsgraden gibt, wurde im ersten Schritt eine Bestandsaufnahme bestehender Formate durchgeführt und daraus ein gemeinsames, erweiterbares und offenes Austauschformat entworfen. Aufgrund der enormen Datenmenge und der Tatsache, dass Messdaten auch nach längerer Zeit performant abgefragt werden mussten, fiel die Entscheidung auf eine Kombination einer relationalen Datenbank für die Metadaten und eine Punkt-Datenbank für die Messwerte.

Konzipierung der Systemarchitektur

Bei der Planung der Systemarchitektur wurde bewusst auf moderne und zukunftssträchtige Technologien und Frameworks gesetzt. Die Architektur basiert auf der Java Spring Boot Microservice Architektur und erlaubt durch den generi-

schon Ansatz laufend die Implementierung von neuen Modulen. Aufgrund der Tatsache, dass der Pegelhub die Kernkomponente einer Gesamtstrategie darstellt, wurde bewusst auf hervorragende Dokumentation und Einfachheit bzw. Erweiterbarkeit geachtet. Der Datenfluss der Pegelhub ist logisch und einfach nachvollziehbar, sodass weitere Projektgruppen den Pegelhub um neue Funktionen erweitern können. Die Kernfunktionen des Pegelhub, nämlich die Verwaltung und die Weitergabe der Pegeldaten an Berechtigte, stellt die Kernaufgabe des Projektes dar.

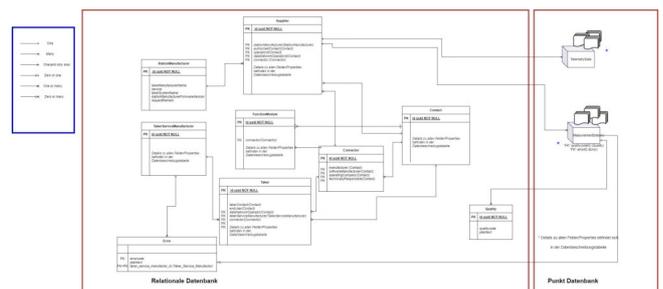


Abbildung 12: Datenmodell des Pegelhub

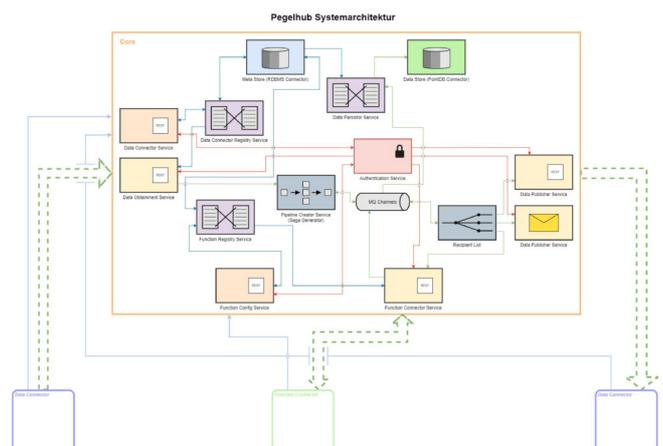


Abbildung 13: Systemarchitektur des Pegelhub

Implementierungsarbeiten

Während den Implementierungsarbeiten wurde laufend Feedback von den zukünftigen Anwen-

der:innen eingeholt um die Einfachheit und Robustheit der Pegelhub-Kernarchitektur frühzeitig bestätigen zu können. Die Implementierungsarbeiten wurden in interaktiven Schritten umgesetzt, sodass stets lauffähige Teilkremente dem Auftraggeber in regelmäßigen Intervallen überreicht werden konnten. Aufgrund der Tatsache, dass es sich um einen berufsbegleitenden Studiengang handelt, konnten alle Projektmitglieder stets ihr Wissen aus dem Berufsalltag einfließen lassen.

Projektergebnis

Als Projektergebnisse stehen dem Auftraggeber die Kernfunktionalitäten zur Pegeldatenverwaltung als vollständig mit Swagger-UI dokumentierte Microservice Architektur für Machine-to-Machine Kommunikation zur Verfügung. Schnittstellen für zukünftige Erweiterungen wurden bereits konzipiert bzw. geplant, wodurch sich die weiteren Implementierungsarbeiten durch andere Projektgruppen einfach gestalten lassen können. Die Termine und der versprochene Projektumfang konnten problemlos eingehalten werden. Der Auftraggeber bedankt sich für die professionelle und engagierte Umsetzung und freut sich bereits auf weitere Projektgruppen, welche den Pegelhub um neue Funktionen ergänzen können.

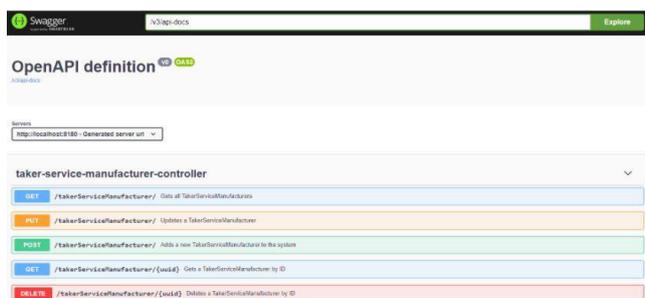


Abbildung 14: Swagger-UI zur Dokumentation der Machine-to-Machine Kommunikation

Herausforderungen

Das Pegelwesen war für alle Projektmitglieder Neuland, wodurch zu Beginn eine intensive Einarbeitungsphase in dieses Themengebiet notwendig war. Um laufend mit dem Projektauftraggeber in Kontakt zu bleiben, wurden wöchentliche Jour-Fixe Termine vereinbart, wobei die Terminfindung aufgrund der Berufstätigkeit aller Beteiligten manchmal eine Herausforderung darstellte. Durch den interaktiven und regen Erfahrungsaustausch innerhalb der Projektgruppe konnten alle Mitglieder schnell auf den gleichen Wissensstand gebracht werden. Der umfangreiche Anforderungskatalog wurde auf die Kernfunktionalitäten heruntergebrochen, um eine gute Basis für zukünftige Erweiterungen zu schaffen.

Fazit

Zum Erreichen der Projektziele waren technische, wissenschaftliche und soziale Kompetenzen gefordert. Um diese Herausforderung zu meistern, waren ein gut strukturiertes Projektmanagement und ein ständiger Austausch der Projektmitglieder erforderlich. Die Projektmitglieder brachten aufgrund der unterschiedlichen beruflichen Tätigkeit optimale Voraussetzungen für das Projekt mit, wodurch die reibungslose und fristgerechte Umsetzung des Projektes ermöglicht wurde. Sowohl die Projektmitglieder, als auch der Auftraggeber, sind mit dem positiven Ergebnis sehr zufrieden. Zur Freude des gesamten Projektteams wird bereits an zukünftigen Erweiterungen des Pegelhub gearbeitet, sodass dieser eine noch zentralere Rolle innerhalb der Digitalisierungsstrategie einnehmen wird.

Autor: Stefan Neuwirth

Projektmitglieder – BSWE Jahrgang 2020

Curriculum Creator



Lisa Allacher
FH Burgenland



Helmut Tischler
FH Burgenland



Günter Ulreich
FH Burgenland

Modernisierung einer bestehenden IoT-Plattform bei FIB



Sophie Wiedemann
FH Burgenland



Michael Pfeiffer
FH Burgenland



Andreas Kappel
FH Burgenland



Daniel Krancz
FH Burgenland

Inventarmanagement



Isabella Zaby
FH Burgenland



Polina Shilkova
FH Burgenland



Stefan Gass
FH Burgenland

Pegelhub



Tanja Buczkowski
FH Burgenland



Stefan Neuwirth
FH Burgenland



Marc Urschick
FH Burgenland

Projektbetreuer



Dipl.-Ing. Franz Knipp

Studiengangsleiter des Bachelorstudiengangs Software Engineering und vernetzte Systeme an der FH Burgenland



Dipl.-Ing. Wolfram Rinke

Fachhochschullehrer am Department für Informationstechnologie an der FH Burgenland

Verzeichnisse

Abbildungsverzeichnis

1	Bearbeitung einer Lehrveranstaltung	3
2	Detailansicht Lehrveranstaltung	3
3	Ansicht eines Markdownfiles auf Github	3
4	Menu Homepage Curriculum Creator	4
5	Ausschnitt aus dem Projektbalkenplan	6
6	Benutzeroberfläche in Form eines SwaggerUIs	7
7	Seefestschiffe, Foto: © Jerzy Bin	8
8	Epics und Roadmap	9
9	Mockup, UML- und ER-Diagramm	9
10	Screenshots der Webanwendung	10
11	Nussdorfer Wehr - Pegeldaten liefern wichtige Informationen zur Steuerung	12
12	Datenmodell des Pegelhubs	12
13	Systemarchitektur des Pegelhubs	12
14	Swagger-UI zur Dokumentation der Machine-to-Machine Kommunikation	13

Abkürzungsverzeichnis

API	Application Programming Interface.
CRUDQ	Create, Read, Update, Delete und Query.
DRY	Don't Repeat Yourself.
ER	Entity Relationship.
FIB	Wirtschaftsagentur Burgenland Forschungs- und Innovations GmbH.
IoT	Internet of Things.
PV	Photovoltaik.
SCRUM	Vorgehensmodell für agile Softwareentwicklung.
UI	User Interface.
UML	Unified Modeling Language.
UX	User Experience.

Impressum

Fachhochschule Burgenland GmbH
Studienzentrum Eisenstadt | Campus 1 | 7000 Eisenstadt
Studienzentrum Pinkafeld | Steinamangerstraße 21 | 7423 Pinkafeld

Tel.: +43 5 7705

E-Mail: office@fh-burgenland.at | www.fh-burgenland.at



Stand 02.2018. Die Inhalte der Broschüre wurden mit größter Sorgfalt erstellt. Für die Richtigkeit, Vollständigkeit und Aktualität der Inhalte können wir jedoch keine Gewähr übernehmen. Alle Angaben vorbehalten Änderungen und Druckfehler.



Dieses Werk ist lizenziert unter einer Creative Commons Namensnennung - Nicht-kommerziell - Weitergabe unter gleichen Bedingungen 4.0 International Lizenz. Diese Lizenz erlaubt es, den Inhalt zu vervielfältigen, zu verbreiten und öffentlich aufzuführen unter folgenden Bedingungen: Der Name des Autors/Rechtsinhabers muss genannt werden. Dieser Inhalt darf nicht für kommerzielle Zwecke verwendet werden. Bearbeiteter oder in anderer Weise umgestalteter Inhalt darf als neu entstandener Inhalt nur unter Verwendung identischer Lizenzbedingungen weitergegeben werden. <http://creativecommons.org/licenses/by-nc-sa/4.0/>

This work is licensed under Attribution-NonCommercial-ShareAlike 4.0 International. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/4.0/>