Projektbericht

Fragebogen zu Schadensdaten für das Institut für Bauforschung e. V.

Inhaltsverzeichnis

[Einleitung 3](#_Toc469534)

[Themenbeschreibung 3](#_Toc469535)

[Motivation 3](#_Toc469536)

[Gruppenmitglieder und Rollen-/Aufgabenverteilung 4](#_Toc469537)

[Technologien 4](#_Toc469538)

[UIKit 4](#_Toc469539)

[Vue.js 5](#_Toc469540)

[Spring 5](#_Toc469541)

[PostgreSQL 6](#_Toc469542)

# Einleitung

## Themenbeschreibung

Das Ziel dieses Projektes ist die Erstellung eines Fragebogens zu Schadensdaten für das Institut für Bauforschung e. V. in Hannover. Dazu gehört auch die anschließende Übertragung der Daten in eine Excel Tabelle. Das Institut für Bauforschung e. V. hat als eine von vielen Kernaufgaben die Forschung im Gebiet der Bauschäden und deren Ursachen. Aufgrund der Tatsache, dass die durchschnittlichen Bauschäden in Deutschland über die Jahre von 2009 bis 2013 von 55.077 Euro auf 66.947 Euro gestiegen sind, ist es wichtig hier die Ursachen zu untersuchen, um die Bauqualität in Zukunft zu verbessern. Der Fragebogen ist über eine Webseite erreichbar und besteht aus 20 Fragen, der verschiedene Themen zum Planungsprozess, der Ausschreibung und Vergabe, der Ausführung, der Abnahme, den Schadenskosten und der Entwicklung der Bauqualität abdeckt. Jeder Benutzer kann den Fragebogen nur einmal ausfüllen, damit verhindert wird, dass die Daten verfälscht werden. Außerdem ist es nur möglich den Fragebogen bis Ende Februar zu beantworten.

Zur Realisierung des Fragebogens werden konventionelle Web Technologien verwendet. Dazu gehören HTML und CSS, um das Layout der Webseite zu erstellen. Außerdem wird JavaScript verwendet, um die dynamische Veränderung der Seite zu ermöglichen. Dazu gehört auch die Kommunikation mit dem Server über HTTP-Requests. Natürlich wird hier eine Datenbank benötigt, um die Übertragung der Antworten in eine Excel Tabelle zu gewährleisten. Das Übertragen der Daten gehört mit zum Projektumfang.

## Motivation

Die Auswertung von Schadensdaten ist in der Bauforschung ein sehr relevantes Thema, da Fehler beim Bau von Wohnraum verheerende Folgen haben können, die unter allen Umständen vermieden werden müssen. Um die Forschung in diesem Bereich voranzutreiben bietet ein Fragebogen eine sehr gute Möglichkeit, um die verschiedenen Ursachen ausfindig zu machen. Mit den heutigen Technologien ist es kein Problem einen solchen Fragebogen zu realisieren, der für alle Baufirmen zugänglich ist. Außerdem dient das Projekt dazu, seine Kenntnisse im Bereich Webentwicklung zu verbessern. Dazu zählt sowohl die Front-End Entwicklung, mit den dazugehörigen Technologien, als auch die Back-End Entwicklung. Gerade die Front-End Entwicklung bietet dank der vielen Frameworks für JavaScript und CSS viele Möglichkeiten zur Vertiefung seiner Kenntnisse.

## Gruppenmitglieder und Rollen-/Aufgabenverteilung

Die Gruppenmitglieder bestanden aus Alba Cruz Torres, Frank Köhn und Michael Nickel. Allgemein wurde das Projekt in die zwei Hauptaufgaben Front-End und Back-End unterteilt, wobei das Front-End weiter in die Aufgaben HTML/CSS und JavaScript Entwicklung unterteilt wurde. Das Back-End umfasst die Datenbankanbindung und die Java Persistence API, die als Schnittstelle zur Datenbank fungiert. Alba übernahm die Entwicklung des Seitenlayouts mit HTML/CSS, Frank hat sich mit der JavaScript Programmierung beschäftigt und Michael hat die Back-End Entwicklung übernommen. Da jedoch alle Aufgaben eng miteinander verknüpft sind und eine enge Zusammenarbeit fordern, gab es häufig Überschneidungen bei der Aufgabenverteilung. So war es häufig notwendig, dass jedes Gruppenmitglied auch andere Aufgaben übernommen hat. Durch die enge Zusammenarbeit der gesamten Gruppe, wurden viele Probleme zusammen gelöst.

# Technologien

In den folgenden Punkten werden auf die verschiedenen Technologien die für das Front-End sowie für das Back-End benutzt werden eingegangen. Es werden nur allgemeine Beispiele gezeigt, da die Implementation später noch Thema des Berichts ist. Einige Technologien, wie die Java Persistence API die als Schnittstelle zur Datenbank dient, oder das Front-End Framework UIKit, welches zur Gestaltung des Web Interfaces dient, waren bereits vorgegeben. Allerdings war es auch möglich andere Technologien einzusetzen, die die Programmierung besser gestaltet und mehr Übersichtlichkeit in den Programmcode bringt. Zunächst werden die Technologien, die im Front-End eingesetzt wurden, besprochen und anschließend befasst sich dieses Kapitel mit den Back-End Technologien.

## UIKit

UIKit ist ein unkompliziertes modulares CSS-Framework zur einfachen Gestaltung von Webseiten. Die Einbindung in ein bestehendes HTML Dokument verläuft ganz einfach wie bei jeder anderen CSS Datei auch über das <head> Element.

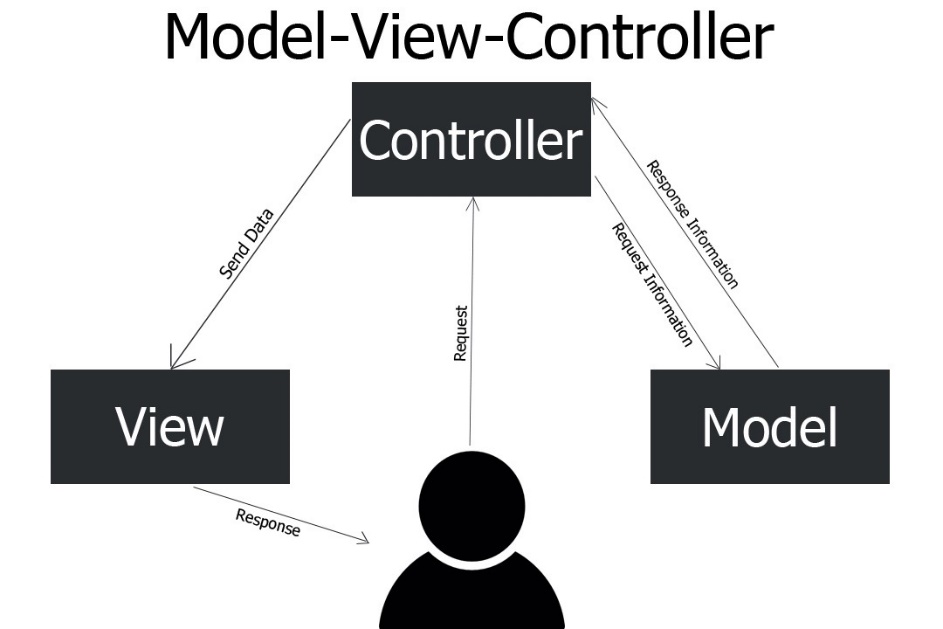
https://i.gyazo.com/563d18563df7ec36ef52567556ada52a.png

1: UIKit Einbindung

Nach der Einbindung kann man die verschiedenen Komponenten von UIKit innerhalb des <body> Elements des HTML Dokuments nutzen. Die Komponenten werden dann über das class Attribut eines HTML Elements eingefügt. So nimmt UIKit dem Programmierer viele Gestaltungsaufgaben ab, da man selbst keine, oder nur sehr wenige, CSS Anweisungen für die verschiedenen Elemente verfassen muss.

## Vue.js

## Spring

Das Spring Framework ist ein Open-Source Framework für die Java-Plattform. Es vereinfacht die Java Programmierung und fördert gute Programmierpraktiken. In unserem Anwendungsfall hat sich Spring angeboten, da das Framework eine Implementierung mit dem Model View Controller Architekturmuster ermöglicht. Das MVC-Konzept ist für die Erstellung von Webanwendungen sehr beliebt, da es eine klare Struktur und eine gute Erweiterbarkeit bietet. Dies ist dadurch möglich, dass bei einer neuen Anwendung das bestehende Modell benutzt werden kann und nur die Steuerung und die Präsentation neu implementiert werden müssen.

2: MVC

Wie der Name schon vermuten lässt, kann man das MVC Architekturmuster in drei Komponenten aufteilen: Das Modell (Model), die Präsentation (View) und die Steuerung (Controller). Ein Modell repräsentiert die logische Struktur von Daten, enthält aber keinerlei Informationen zur Benutzeroberfläche. Die Präsentation sorgt für die Darstellung der Oberfläche, in unserem Fall also die Darstellung des Fragebogens über die Webseite. Die Benutzerinteraktionen werden an die Steuerung weitergegeben. Zusätzlich wird die Präsentation über Änderungen der Daten im Modell unterrichtet. Die Steuerung ist das Bindeglied zwischen dem Modell und der Präsentation. Es sorgt für die Kommunikation zwischen den beiden Komponenten und verwaltet die Präsentation, indem es, wie weiter oben schon beschrieben, die Benutzerinteraktionen entgegennimmt, diese auswertet und entsprechend agiert.

Außerdem enthält Spring das Prinzip der Dependency Injection, wodurch den Objekten die benötigten Ressourcen zugewiesen werden und der Programmierer diese nicht selbst suchen muss.

## PostgreSQL

PostgreSQL ist ein relationales Open-Source Datenbankmanagementsystem. Es ist frei verfügbar und kann ohne Lizensierung benutzt werden. Ursprünglich wurde es als universitäres Projekt entwickelt und seit 1996 wird es von vielen Programmierern weiterentwickelt. Verwendet wird die SQL Datensprache, jedoch muss man dank JPA keine eigenen SQL Queries verfassen. Hinzu kommt, dass PostgreSQL auf dem Client-Server-Modell basiert, wodurch Client und Server nicht auf demselben Rechner laufen müssen. Da das Smart Monitoring System bereits mit PostgreSQL läuft, wurde auch für dieses Projekt das System verwendet.

# Softwareentwurf

Dieses Kapitel beschäftigt sich mit den Anforderungen der Software und den dazugehörigen Systemmodellierungen und Entwürfen. Der Software Entwurf ist ein wichtiger Bestandteil eines Projektes, um eine hohe Qualität der Software zu garantieren. Außerdem verringert es das Risiko von Fehlentwicklungen und bietet allen Beteiligten zu jedem Zeitpunkt einen Überblick des Software Konzeptes. In den folgenden Punkten werden zunächst die funktionalen sowie nicht funktionalen Anforderungen besprochen, bevor auf die weiteren Spezifikationen der Software, in Form von UML Diagrammen, Use Cases und Mockups, eingegangen wird.

## Funktionale Anforderungen

Grundsätzlich dient der Fragebogen für das Institut für Bauforschung e. V. zum Ausfüllen für Sachverständiger, Baugewerbe und Bauindustrien im Bezug auf Schadensdaten. Die Daten werden in einer Datenbank gespeichert, um hinterher in eine Excel Tabelle, zur weiteren Auswertung, überführt zu werden. Bevor der eigentliche Fragebogen beginnt, erhält der User einen kurzen Informationstext. Wie schon erwähnt wurde, enthält der Fragebogen 20 Fragen, wobei jede Frage eine eigene Seite bekommt.

Das Ausfüllen erfolgt über verschiedene Eingabe- und Bedienelemente. Bei Fragen wo der User mehrere Antworten auswählen darf, werden Checkboxen benutzt. Falls nur eine Antwort ausgewählt werden darf, kommen Radio-Buttons zum Einsatz. Wenn Zahleneingaben nötig sind, wird eine Kombination aus Schiebereglern und Textfeldern benutzt. Textfelder kommen außerdem zum Einsatz, wenn der User einen eigenen Text verfassen soll.

Die Navigation erfolgt über zwei Buttons zum Wechseln der Seiten beziehungsweise zum Starten und Absenden des Fragebogens. Außerdem enthält jede Seite Information zur Fragestellung, damit der User immer weiß wie er die jeweilige Frage zu beantworten hat. Falls Eingaben fehlerhaft sind, wird der User daran gehindert zur nächsten Frage zu springen und erhält Informationen zum Fehler.

Zusätzlich enthält die Software eine Authentifizierung, damit ein User den Fragebogen nicht mehrmals ausfüllen kann.

## Nicht Funktionale Anforderungen

Gerade bei einem Fragebogen ist die Usability von besonderer Bedeutung, da sich der User nicht lange mit dem Ausfüllen eines Fragebogens beschäftigen will. Also muss die Seite simpel gestaltet und intuitiv nutzbar sein. Außerdem sollte die Erledigung nicht mehr als 10 bis 15 Minuten beanspruchen.

Zuverlässigkeit muss auch gewährleistet sein, da die korrekte Datenübertragung in die Datenbank essenziell für einen Fragebogen ist, sonst würde dieser seinen gesamten Zweck verfehlen.

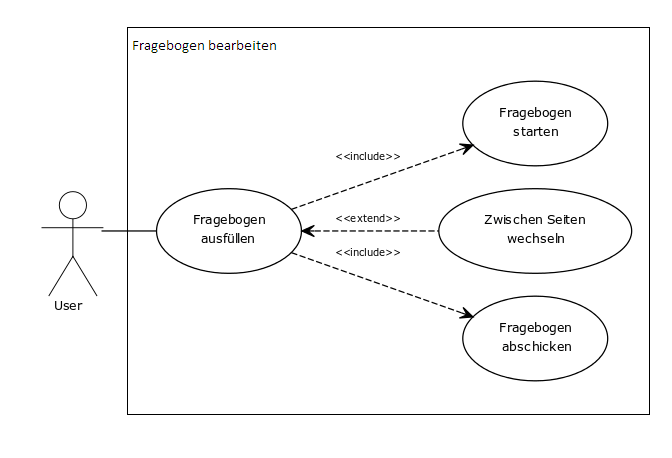
Die Webseite muss von jedem Rechner mit Internetzugang erreichbar sein.

Der Wechsel zwischen den Seiten und die Eingabe über die Bedienelemente muss zudem schnell vonstattengehen. Dies hängt auch mit der Usability der Seite zusammen.

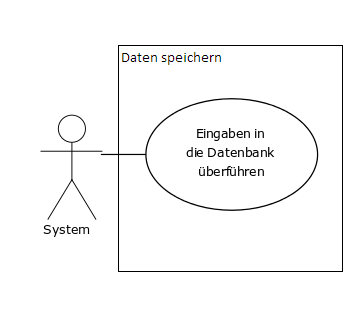
## Use Case

Für den Fragebogen wurden zwei Use Case Diagramme entwickelt. Das erste Diagramm bezieht sich auf den User und seinen Anwendungsfällen. Das zweite Diagramm ist aus Sicht des Systems. Der User füllt den Fragebogen aus, indem er die Fragen beantwortet. Dies beinhaltet auch die Anwendungsfälle „Fragebogen starten“ und „Fragebogen abschicken“. Zudem wird das Ausfüllen durch den Anwendungsfall „Zwischen Seiten wechseln“ erweitert.

Das System

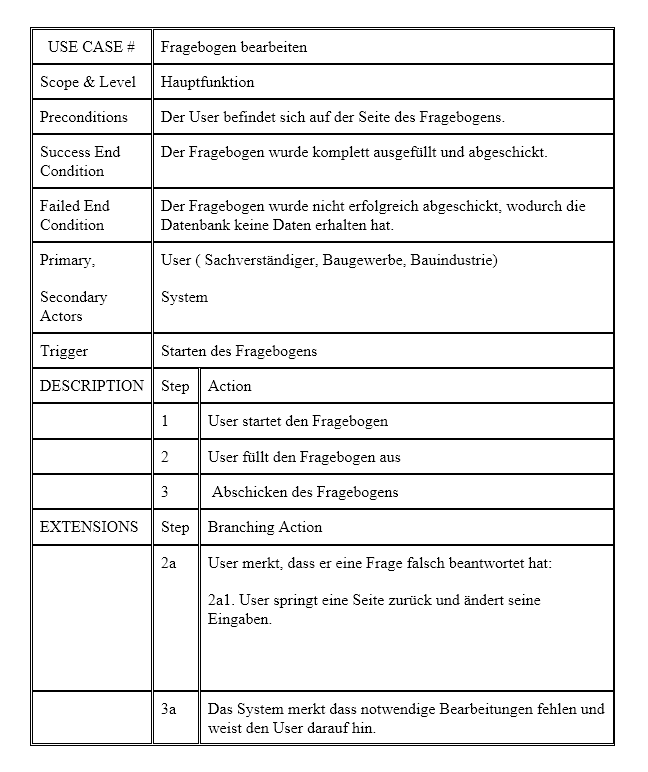


: Use Case Fragebogen bearbeiten

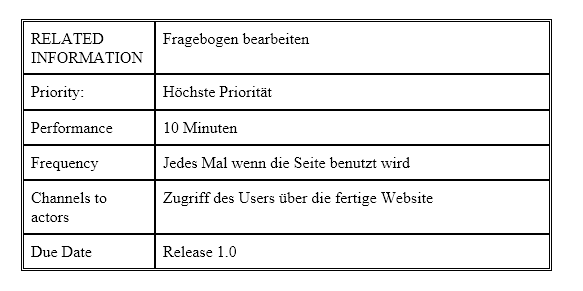


4: Use Case Daten speichern

Die folgende Tabelle zeigt die textuelle Beschreibung des Use Cases.



5: Use Case textuell



6: Use Case related information

## Mockups