



Diplomarbeit

GeoQuest

Bewegung durch Technik

Imst, 23. Mai 2017

Eingereicht von

Christopher Haider

Verantwortlich für IT: HTML, CSS, BWL: Kaufvertrag

Maximilian Egger

Verantwortlich für IT: HTML, CSS, BWL: Kaufvertrag

Otto Normalverbraucher

Verantwortlich für IT: HTML, CSS, BWL: Kaufvertrag

Eingereicht bei

Alexander Scharmer, Claudio Landerer und Stefan Stolz

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbst verfasst und keine anderen als die angeführten Behelfe verwendet habe. Alle Stellen, die wörtlich oder inhaltlich den angegebenen Quellen entnommen wurden, sind als solche kenntlich gemacht. Ich bin damit einverstanden, dass meine Arbeit öffentlich zugänglich gemacht wird.

Ort, Datum

Christopher Haider

Maximilian Egger

Dominik Neuner

Harald Sommer

Abnahmeerklärung

Hiermit bestätigt der Auftraggeber, dass das übergebene Produkt dieser Diplomarbeit den dokumentierten Vorgaben entspricht. Des Weiteren verzichtet der Auftraggeber auf unentgeltliche Wartung und Weiterentwicklung des Produktes durch die Projektmitglieder bzw. die Schule.

Ort, Datum

Auftraggeber

Vorwort

z. B. Hinweise, wie das bearbeitete Thema gefunden wurde oder Dank für die Betreuung (Kooperationspartner/in, Betreuer/innen, Sponsoren) etc.

Abstract (Deutsch)

Es soll eine Applikation entwickelt werden, bei der Fragen (zu einem Thema) und zugehörigen mögliche Antworten (mit einer richtigen) erstellt werden können. Jeder Frage muss ein Standort auf der Karte zugewiesen werden. Im Front-End können Fragen im Umkreis aufgelistet und gelöst werden. Für jede gelöste Frage erhöht sich der Punktestand des Benutzers.

Abstract (Englisch)

An App should be developed, where questions (to a topic) and the belonging answers (with one correct) are able to be created. Each Question must have a location on the map. In the front-end-mode questions in perimeter can be listed and solved. After every solved Question the score of the user will be increased.

Inhaltsverzeichnis

1. Projektmanagement	13
1.1. Metainformationen	13
1.1.1. Team	13
1.1.2. Betreuer	13
1.1.3. Partner	13
1.1.4. Ansprechpartner	15
1.2. Vorerhebungen	15
1.2.1. Projektzieleplan	15
1.2.2. Projektumfeld	16
1.2.3. Risikoanalyse	18
1.3. Pflichtenheft	18
1.3.1. Zielbestimmung	18
1.3.2. Produkteinsatz und Umgebung	18
1.3.3. Funktionalitäten	19
1.3.4. Testszenarien und Testfälle	19
1.3.5. Liefervereinbarung	19
1.4. Planung	20
1.4.1. Projektstrukturplan	20
1.4.2. Meilensteine	20
1.4.3. Gant-Chart	21
1.4.4. Abnahmekriterien	21
1.4.5. Pläne zur Evaluierung	22
1.4.6. Ergänzungen und zu klärende Punkte	23

2. Vorstellung des Produktes	24
3. Eingesetzte Technologien	25
4. Problemanalyse	27
4.1. USE-Case-Analyse	27
4.2. Domain-Class-Modelling	27
4.3. User-Interface-Design	27
5. Systementwurf	33
5.1. Architektur	33
5.1.1. Design der Komponenten	33
5.1.2. Benutzerschnittstellen	34
5.1.3. Datenhaltungskonzept	34
5.1.4. Konzept für Ausnahmebehandlung	34
5.1.5. Sicherheitskonzept	34
5.1.6. Design der Testumgebung	35
5.1.7. Design der Ausführungsumgebung	35
5.2. Detailentwurf	35
6. Implementierung	37
7. Deployment	38
8. Tests	39
8.1. Systemtests	39
8.2. Akzeptanztests	39
9. Projektevaluation	40
10. Benutzerhandbuch	41
11. Betriebswirtschaftlicher Kontext	42
12. Zusammenfassung	43

13. Beispielkapitel	44
13.1. Beispiele zitieren	44
13.1.1. Beispiele Abbildungen	45
13.2. Beispiele Listen	46
13.3. Beispiel Codesequenz	48
13.3.1. Quicksort in JAVA	48
13.4. Beispieltext	50
Abbildungsverzeichnis	53
Tabellenverzeichnis	54
Quelltexte	55
Literaturverzeichnis	56
A. Anhang-Kapitel	58
A.1. Anhang-Section	58

Einleitende Bemerkungen

Notationen

Beschreibung wie Code, Hinweise, Zitate etc. formatiert werden

1. Projektmanagement

1.1. Metainformationen

1.1.1. Team

Christopher Haider (Leiter, Programmierer, Design), Maximilian Egger (Programmierer, Druck)

1.1.2. Betreuer

NEUNER Dominik, MSc

1.1.3. Partner

HAK Imst



Abbildung 1.1.



Abbildung 1.2.

1.1.4. Ansprechpartner

NEUNER Dominik, MSc

1.2. Vorerhebungen

1.2.1. Projektzieleplan

-Einteilung:

OBERZIEL Es soll eine Applikation entwickelt werden, bei der Fragen (zu einem Thema) und zugehörigen mögliche Antworten (mit einer richtigen) erstellt werden können. Jeder Frage muss ein Standort auf der Karte zugewiesen werden. Im Front-End können Fragen im Umkreis aufgelistet und gelöst werden. Für jede gelöste Frage erhöht sich der Punktestand des Benutzers.

LEISTUNGSZIEL Das Projekt wird mehrere Mitarbeiter beinhalten und wir versuchen den einzelnen Mitarbeitern die optimale Arbeit zuzuweisen, um eine pünktliche Fertigstellung mit würdiger Qualität zu erreichen. Implementierung von aktuellen Karten und Standort, kleine Übersicht des Punktestandes, Backend/Frontend;

KOSTENZIEL Die Kosten belaufen sich ca. in Höhe von 8.500 - 12.500 € abhängig von der Zeit der aktiven Arbeit jedes einzelnen Mitarbeiters/Programmierer plus den Überstunden inklusive allen Aufwänden für die Programmierung, Serverwartungen, laufende Aktualisierungen von den Daten. Abgesehen von den Steuern.

TERMINZIEL Dieses etwas umfangreichere Projekt sollte mit Einplanung

-Tabelle:

Stakeholder	Einfluss	Nähe	Einstellungen	Beschreibung
Studierende	gering	fern	neutral	Zielpublikum
Dominik Neuner	groß	nahe	Positiv	Studienkoordinator
HAK	gering	fern	neutral	Teilen von Lehrpersonal und Räumen
Kunde	gering	nahe	neutral	Primärliteratur
Christopher H	groß	nahe	Positiv	Programmierer
Maximilian E	groß	nahe	Positiv	Leiter, Programmierer

Abbildung 1.3.: Projektnähe tabellarisch

vom 14.03.2017 in einem Zeitrahmen bis Ende November/Dezember in Form von einer kompakten App im Store verfügbar sein.

DETAILZIEL Unserem Partner/Kunden vor der endgültigen Fertigstellung des Projekts eine Testapp / Prototypapp vorzeigen und nochmals eine endgültige Absprache mit den Kunden abhalten.

1.2.2. Projektumfeld

Welche Bedrohungen bestehen für das Projekt Budget Überschreitung

- Termine nicht einhalten
- Auftragsgeber sagt ab
- Auftragsgeber ändert seine Anforderungen
- Probleme bei der Realisierung

-Grafisch:

Nähe zum Zentrum / Größe der Quadrate / Farbe = Einstellung

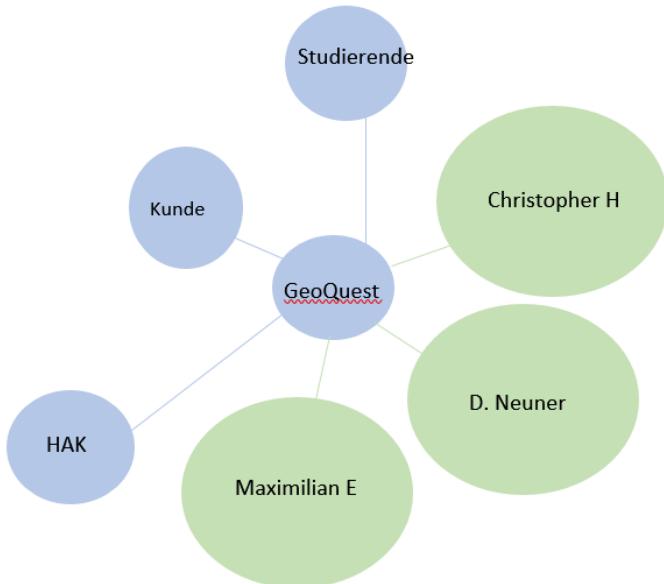


Abbildung 1.4.: Projektnähe Grafisch

Risiko	Wahrs.	Ausw.	Prioris.	Behandlung
Urheberrechtlich geschützte Bilder	4	8	70	Selber Bilder aufnehmen
Langeweile der Arbeitenden	8	8	60	Mit mehr Medien arbeiten
Budgetüberschreitung	5	9	32	Mit vorhandenen arbeiten
Serverabsturz	6	10	80	Backup der Daten (2x speichern)
Auftraggeber entscheidet anders	7	7	80	Ihn überzeugen m. a. Statement
Entfall der Sponsoren	6	7	90	Neue suchen od. überzeugen

Abbildung 1.5.: Risikomatrix

1.2.3. Risikoanalyse

1.3. Pflichtenheft

1.3.1. Zielbestimmung

Projektbeschreibung Es soll eine Applikation entwickelt werden, bei der Fragen (zu einem Thema) und zugehörigen mögliche Antworten (mit einer richtigen) erstellt werden können. Jeder Frage muss ein Standort auf der Karte zugewiesen werden. Im Front-End können Fragen im Umkreis aufgelistet und gelöst werden. Für jede gelöste Frage erhöht sich der Punktestand des Benutzers.

IST-Zustand Diese Applikation gibt es nach dieser Weise noch nicht, dennoch sind Ähnlichkeiten mit Pokemon Go zu erkennen. Obwohl es nicht um Pokemon geht, sonder um Fragen und Antworten die einen Standort besitzen.

SOLL-Zustand Diese Applikation soll dem Nutzer die Möglichkeit geben Informatik außerhalb des Gebäudes zu beantworten. Dies soll die Anwender anregen mehr ins freie zu gehen um Sport zu betreiben, dennoch sein Wissen im Bereich IT zu erweitern

1.3.2. Produkteinsatz und Umgebung

- in Form einer App im Store erhältlich
- Jugendliche und Erwachsene
- Lizenz
- GPS, Android-Phone und Internetverbindung

1.3.3. Funktionalitäten

- MUSS-Anforderungen
 - Fragen hinzufügen, Antworten hinzufügen, Punktestand erhöhen, Standort muss sich ändern können, Absturz der App vermeiden
 - Design, auf allen Plattformen erhältlich
- KANN-Anforderungen
 - Fehlerausgabe,
 - Nicht-funktional

1.3.4. Testszenarien und Testfälle

- Beschreibung der Testmethodik
- Testfall 1
- Testfall 2
- ...

1.3.5. Liefervereinbarung

- Applikation, Server, Store
- Adminmodus, Benutzermodus
- Designer, Programmierer, Web-Designer

1.4. Planung

1.4.1. Projektstrukturplan

- Verteilung der Arbeit
- Gerechte Arbeitsaufteilung bzw. Gehalt
- Ausführung des Projekts
- Kontrolle (Funktional- und Qualitätskontrolle)
- Testdurchlauf (Prototyp)
- Möglich Fehler beheben bzw. dem Kunden präsentieren

1.4.2. Meilensteine

Einteilung in Etappen = sogenannte Zwischenziele, erleichtert die Projektplanung

- Einteilung
- Ausführung
- Kontrolle
- Testdurchlauf
- Endprodukt

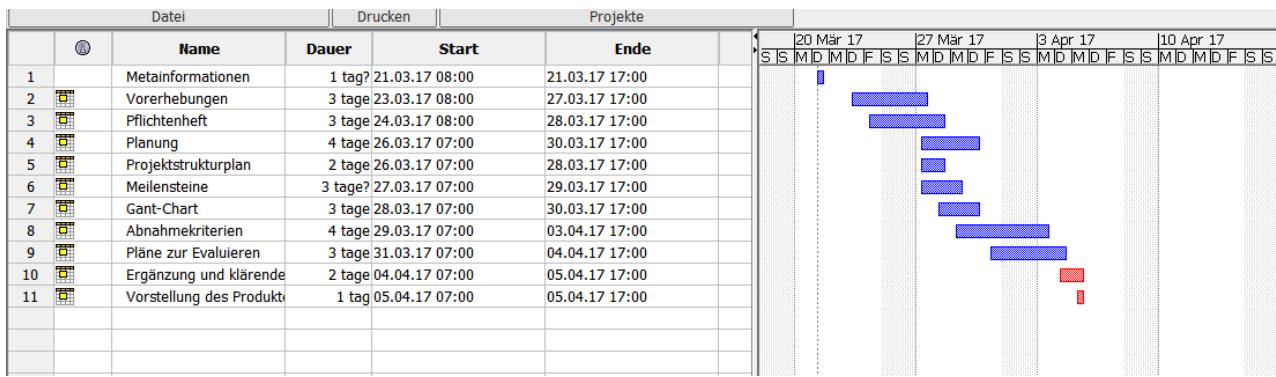


Abbildung 1.6.

1.4.3. Gant-Chart

1.4.4. Abnahmekriterien

= Bezeichnung des letzten Schritts in der Systemauswahl Erfolgt nach erfolgreich, abgeschlossenen Projektbetrieb.

Umfasst Hard- und Softwarekomponenten -> Berücksichtigung der Leistungsanforderungen

natürlichsprachliches abstraktes Abnahmekriterium

Ausgangssituation: Mindestens 1 Nutzer ist in der App (virtuellen Welt) angemeldet

Ereignis: Im Rahmen des Spielvorgangs der App kann der Nutzer seinen Charakter auswählen und sich auf die virtuelle Map begeben

erwartetes Ergebnis: der Nutzer kann ohne Probleme auf der Map umhergehen und gestellte IT-Fragen beantworten in Form einer spielerischen Art

formalisiert abstraktes Abnahmekriterium

In Form einer Tabelle

Ablauf:

- Nutzer hat die volle Kontrolle über seinen eigenen Charakter in der App bzw. Webseite
- Er kann seinen Charakter bearbeiten und PlugIns importieren, Mapskins und weitere Layoutänderung in der App vornehmen die keinen Einfluss auf die Funktionalität haben
- Ihm wird ermöglicht Fehlerquellen beheben zu können
- Bildschrimausgabe auf dem Screen des Nutzers
- Seine Ergebnisse der beantworteten Fragen können online gespeichert oder lokal ausgedruckt werden -> Rangliste

1.4.5. Pläne zur Evaluierung

Wesentlich für seriöse und erfolgreiche (Selbst-)Evaluation sind vor allem . . .

einsichtige Gründe und spürbare Folgen

- erweitern des Wissens in der Natur indem man zu Standorten auf der App läuft/geht und eine Frage beantwortet oder eine Frage erstellt

ein positiver Ansatz

- einzigartig, aktuell, qualitativ hochwertig, Benutzerfreundlich

relevante Fragestellungen und Kriterien

- möglichst viel Aufmerksamkeit der User bekommen
- Bekanntheitsgrad erhöhen
- Feedback von den Usern zur App bekommen

wirksame Methoden und Instrumente

- App ist völlig Kostenlos
- leicht bedienbar
- für jede Altersklasse tauglich

ein multiperspektivischer Ansatz

Da unser Professor sich diese Applikation zunächst auf Fehler bzw. Macken analysiert und wir sie dann der Klasse vorstellen werden wir uns wahrscheinlich die ein oder andere Idee noch einbauen oder einen Prototypen erstellen bevor wir es im Markt präsentieren.

klare Verantwortlichkeiten und Entscheidungsstrukturen

Verantwortlich für Projekt/App bzw. Webseite sind wir selbst. (Christopher Haider und Maximilian) Entscheidungen werden nur durch Christopher Haider den Projektleiter zustande kommen. Mehr Informationen zu den jeweiligen Themen finden Sie oben bei der Abbildung 1.6.

machbare Pläne und gesicherte Rahmenbedingungen

Qualifikationen: Beteiligte dieses Projektes besitzen spezielle Fachkenntnisse bzw. Qualifikationen

Zeit: Unser Plan ist es dieses Projekt bis Ende Schulschluss fertigzustellen.

Geldmittel:

1.4.6. Ergänzungen und zu klärende Punkte

2. Vorstellung des Produktes

Vorstellung des fertigen Produktes anhand von Screenshots, Bildern, Erklärungen.

3. Eingesetzte Technologien

- html, java script, php, datenbank (sql), css
- php, html, css, java script, datenbank
- ajax
- css



Abbildung 3.1.

4. Problemanalyse

4.1. USE-Case-Analyse

- Frage löschen
- Frage bearbeiten
- Menü mit Fragen anzeigen

4.2. Domain-Class-Modelling

4.3. User-Interface-Design

USE CASE	beantworten	Made by Christopher & Maximilian E.
Ziel im Kontext	Frage beantworten	
Akteure		
Primär	Admin, Benutzer	
Sekundär	Auftraggeber, Hak Imst	
Vorbedingungen	Am Standort, Internetverbindung	
Auslösendes Ereignis	Antwortmöglichkeiten scheinen auf	
Ablaufschritte		
Standard	Kein Fehler tritt auf, Frage wurde korrekt beantwortet, Benutzer klickt auf eine der Antwortmöglichkeiten	
Alternative	Keine Antwort ausgewählt	
Nachbedingung		
Bei Erfolg	Frage richtig beantwortet, keine Fehler treten auf	
Bei Misserfolg	App funktioniert nicht/ startet nicht, Programmierfehler, Server, nicht den Erwartungen entspricht	

Schritt 1: App öffnen / Webseite besuchen

Schritt 2: Zu einem Standort gehen

Schritt 3: Frage auswählen

Schritt 4: Frage beantworten

Schritt 5: Korrekte Antwort scheint auf

Schritt 6: Bewertung

Schritt 7: Abmelden / App Schließen

Abbildung 4.1.

USE CASE	hinzufügen	Made by Christopher & Maximilian E.
Ziel im Kontext	Frage hinzufügen	
Akteure		
Primär	Admin, Programmierer	
Sekundär	Hak Imst	
Vorbedingungen	Benötigten Daten z.B. Frage, Antworten	
Auslösendes Ereignis	Neues Fenster wo Fragen + Antworten hinzuzufügen sind	
Ablaufschritte		
Standard	Keine Fehler, Reale Infos (keine Lüge), Frage muss Ort haben	
Alternative	Richtige muss gekennzeichnet sein damit das Programm weiß, ob korrekt oder nicht	
Nachbedingung		
Bei Erfolg	Frage + Antwort mit Standort kann hinzugefügt werden	
Bei Misserfolg	App funktioniert nicht (nicht wie erwartet), Programmierfehler, Server reagiert nicht, entspricht den Erwartungen des Kunden nicht, Fehler tritt auf	

Schritt 1: App öffnen / Webseite aufrufen

Schritt 2: Adminmodus

Schritt 3: Frage hinzufügen (Plus Symbol)

Schritt 4: Frage plus Antwort wird mit einem Standort eingetragen

Schritt 5: korrekte Antwort wird markiert

Schritt 6: Speichern und Upload

Schritt 7: Zurück zum Menü, Weitere Frage hinzufügen

Schritt 8: Programm schließen

Abbildung 4.2.

GeoQuest

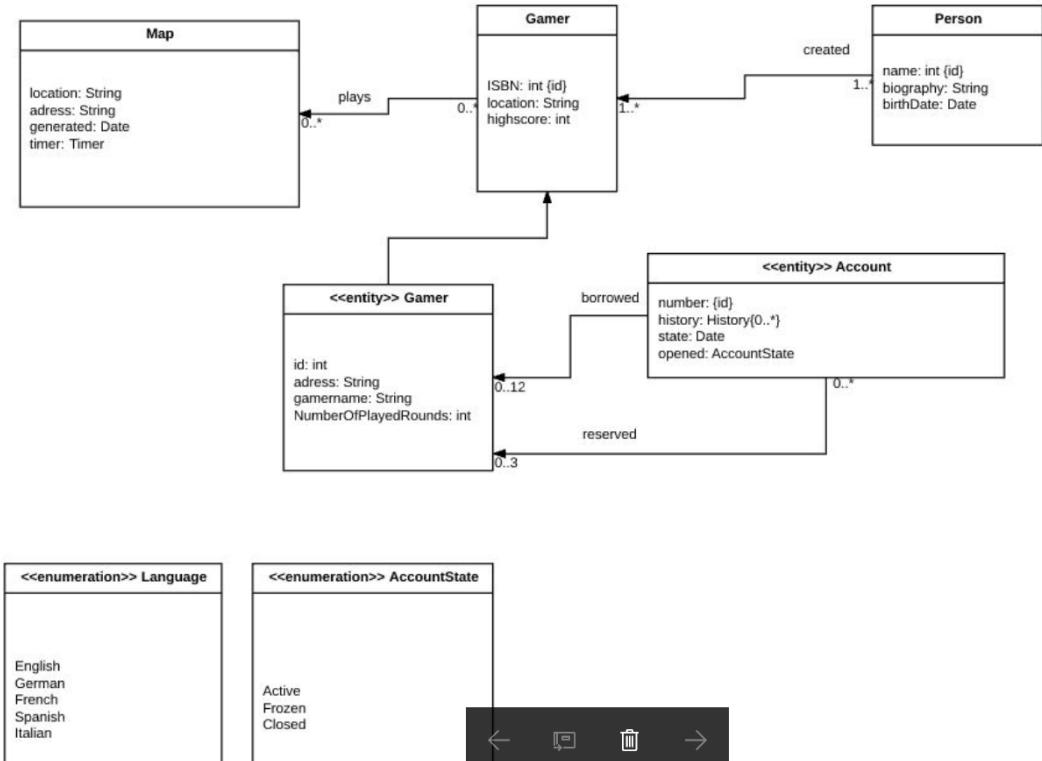


Abbildung 4.3.

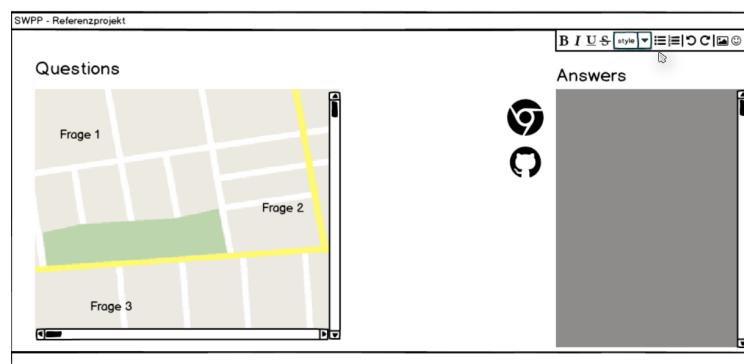


Abbildung 4.4.

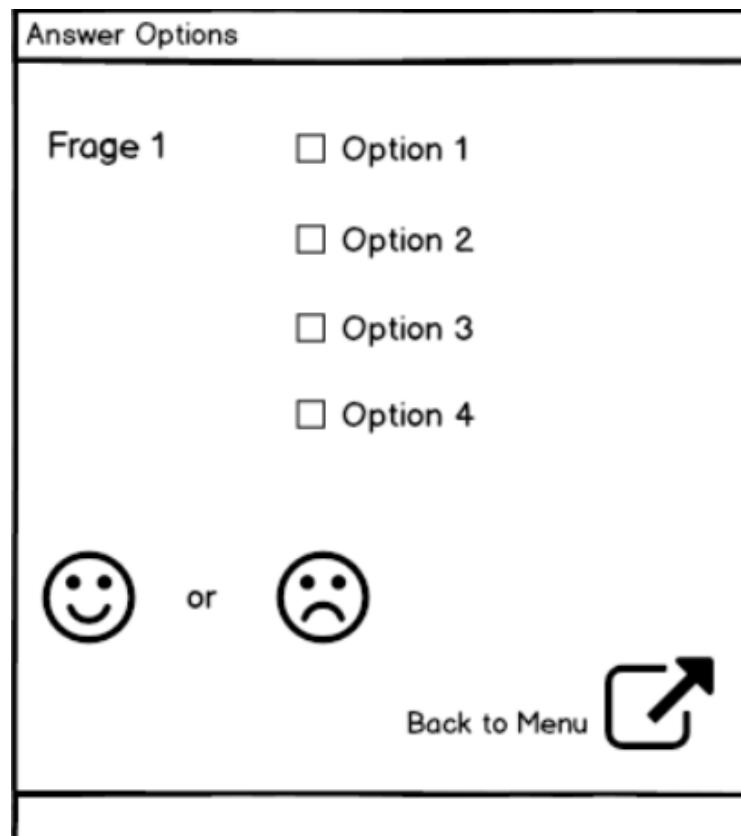


Abbildung 4.5.

Error - Report
<p>Comment us your Problems !</p> <p><input type="checkbox"/> Problem with Q & A <input type="checkbox"/> Problem with the functionality</p>
<p>B I U S style ▼ ≡ C C </p>
Type your text here ...

Abbildung 4.6.

5. Systementwurf

5.1. Architektur

5.1.1. Design der Komponenten

Darstellung und Beschreibung der Systemarchitektur;

- statische Zerlegung des Systems in seine physischen Bestandteile (Komponenten, Komponentendiagramm)
- (textuelle) Beschreibung des dynamischen Zusammenwirkens aller Komponenten
- (textuelle) Beschreibung der Strategie für die Architektur, d. h. wie die Architektur in Statik und Dynamik funktionieren soll.
- Verwendung von Referenzarchitekturen bzw. Architekturmustern (als Schablonen, z.B. MVC, Plugin, Pipes and Filters)
 - MVC
 - Schichten
 - Pipes
 - Request Broker
 - Service-Oriented

5.1.2. Benutzerschnittstellen

- Design des UIs
- Dialoge, Dialogsteuerung, Ergonomie, Gestaltung, Eingabeüberprüfungen

5.1.3. Datenhaltungskonzept

- Design der Datenbank (ER-Modell)
- Design des Zugriffs auf diese Daten (Datenhaltungskonzept)
- Caching, Transaktionen

5.1.4. Konzept für Ausnahmebehandlung

- Systemweite Festlegung, wie mit Exceptions umgegangen wird
- Exceptions sind primär aus den Bereichen UI, Persistenz, Workflow-Management

5.1.5. Sicherheitskonzept

Beschreibung aller sicherheitsrelevanten Designentscheidungen

- Design der Security-Elemente
- Design von Safety-Elementen (Fehlertoleranz, Verfügbarkeit etc.)

5.1.6. Design der Testumgebung

- wie wird getestet (Unit-Testing, Integrationstesting, Systemtests, Akzeptanztests)
- Testumgebung, Testprozess, Teststrategie, Testmethoden, Testfälle

5.1.7. Desing der Ausführungsumgebung

- Deployment (DevOps)
- Betrieb (besonders Hoch- und Hertunerfahren der Anwendung)

5.2. Detailentwurf

Design jedes einzelnen USE-Cases

- Design-Klassendiagramme vom Domain-Klassendiagramm ableiten (incl. detaillierter Darstellung und Verwendung von Vererbungshierarchien, abstrakten Klassen, Interfaces)
- Sequenzdiagramme vom System-Sequenz-Diagramm ableiten
- Aktivitätsdiagramme
- Detaillierte Zustandsdiagramme für wichtige Klassen

Verwendung von CRC-Cards (Class, Responsibilities, Collaboration) für die Klassen

- um Verantwortlichkeiten und Zusammenarbeit zwischen Klassen zu definieren und
- um auf den Entwurf der Geschäftslogik zu fokussieren

Design-Klassen für jeden einzelnen USE-Case können z.B. sein:

- UI-Klassen
- Data-Access-Klassen
- Entity-Klassen (Domain-Klassen)
- Controller-Klassen
- Business-Logik-Klassen
- View-Klassen

Optimierung des Entwurfs (Modularisierung, Erweiterbarkeit, Lesbarkeit):

- Kopplung optimieren
- Kohäsion optimieren
- SOLID
- Entwurfsmuster einsetzen

6. Implementierung

Detaillierte Beschreibung der Implementierung aller Teilkomponenten der Software entlang der zentralsten Use-Cases:

- GUI-Implementierung
- Controllerlogik
- Geschäftslogik
- Datenbankzugriffe

Detaillierte Beschreibung der Teststrategie (Testdriven Development):

- UNIT-Tests (Funktional)
- Integrationstests

Zu Codesequenzen:

- kurze Codesequenzen direkt im Text (mit Zeilnummern auf die man in der Beschreibung verweisen kann)
- lange Codesequenzen in den Anhang (mit Zeilennummer) und darauf verweisen (wie z.B. hier ??)

7. Deployment

- Umsetzung der Ausführungsumgebung
- Deployment
- DevOps-Thema

8. Tests

8.1. Systemtests

Systemtests aller implementierten Funktionalitäten lt. Pflichtenheft

- Beschreibung der Teststrategie
- Testfall 1
- Testfall 2
- Tesfall 3
- ...

8.2. Akzeptanztests

9. Projektevaluation

siehe Projektmanagement-Unterricht

10. Benutzerhandbuch

falls im Projekt gefordert

11. Betriebswirtschaftlicher Kontext

BW-Teil

12. Zusammenfassung

- Etwas längere Form des Abstracts
- Detaillierte Beschreibung des Outputs der Arbeit

13. Beispielkapitel

13.1. Beispiele zitieren

Das ist ein Zitat mit Klammern, (Resnick, 1996), das ein Zitat ohne Klammern: Harel und Papert (1991). Hier das selbe Zitat mit einer Seitenangabe und Klammern (Resnick, 1996, S. 23).

Wird ein Absatz aus einer Quelle sinngemäß übernommen (nicht wörtlich), dann kann nach dem Absatz das entsprechende Zitat in Klammern angeführt werden. (Anastopoulou u. a., 2012, S. 33)

Wenn ein Zitat im Text angegeben wird, wie z.B. so Beer, Rudolf und Benischek, Isabella (2011), können die Klammern weggelassen werden.

Der folgende Absatz zeigt ein Blockzitat (wörtlich übernommene Textpassage aus einer Quelle):

Dr. Heinrich Faust ist ein angesehener Wissenschaftler und Akademiker, der trotz seiner wissenschaftlichen Studien und einer guten Bildung seinen Wissensdurst nicht stillen kann. Eines Nachts sitzt er in seinem Studierzimmer und grübelt über den Sinn des Lebens nach, findet jedoch keine Antworten. Daraufhin wendet er sich der Geisterwelt zu. Er beschwört einen Erdgeist, versucht sich den Geistern gleich zu stellen, was ihm jedoch nicht gelingt. Von Ohnmacht

getrieben will er sich das Leben nehmen. Sein Selbstmordversuch wird jedoch von Glockenläuten zum Ostertag und seinen Kindheitserinnerungen gestört. (Ackermann, 2001, S. 21)

Hier wird ein wörtliches Zitat inline angegeben: „Das ist ein kleines direktes Zitat.“ Göhlich und Zirfas (2007), und danach geht es gleich wieder direkt weiter. Ob ein wörtliches Zitat inline oder als eigener Block angezeigt wird, entscheidet Latex auf Basis der Länge.

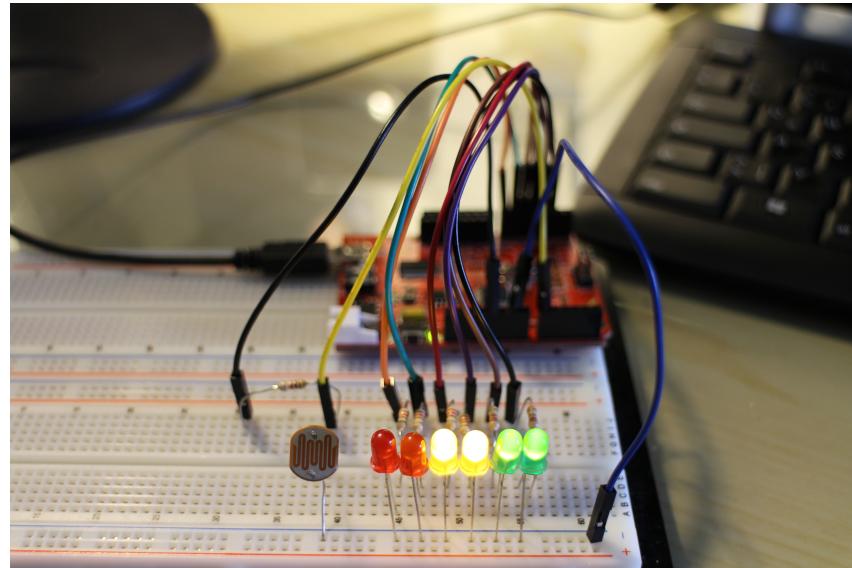
13.1.1. Beispiele Abbildungen

Auf diese Weise kann man zum Beispiel in Latex auf die Abbildung 13.1 verweisen. Die Kennung für den Verweis vergibt man selbst mit dem „label“ Kommando bei der Abbildung.

Jede Abbildung muss nicht nur mindestens einen Verweis im Text haben. Es wird außerdem eine Bildunterschrift verlangt. Für diese ist festgesetzt, dass die Abbildungsunterschrift alleine ausreichend sein muss, um zu verstehen, was am Bild zu erkennen ist.

Der nächste wichtige Punkt sind die Quellenangaben bei Abbildungen. Der Author muss zu jeder Abbildung die notwendigen Rechte haben und idealer Weise gibt man diese bei der Abbildung mit an. In Abbildung 13.1 auf Seite 46 sieht man das.

Es ist wichtig zu verstehen, dass Latex die Positionierung von Abbildungen übernimmt. Man definiert die Abbildung über begin-figure dort, wo man die Abbildung in etwa haben möchte, den Rest übernimmt Latex



© Stefan Stolz (CC BY-SA 3.0)

Abbildung 13.1.: Hintergrund: Arduino Board; Vordergrund: eine Lichterreihe und ein Lichtsensor (Fotowiderstand); In diesem Beispiel wird die Lichterreihe je nach Helligkeit des Umgebungslichtes gesteuert. Durch leichte Modifikationen kann man damit eine Lichtschranke oder auch eine Helligkeitssteuerung für das Smartphone simulieren.

Beispiele Tabellen

Tabelle 13.1 ist ein Beispiel für eine aufwändiger Tabelle mit einer Abbildung und Überschrift.

Tabellen sind in Latex sehr kompliziert zu erzeugen. Alternativ kann man die Tabellen auch in einem anderen Programm gestalten und als Bild wieder einfügen. Dieses Bild kann dann innerhalb von begin-Table verwendet werden.

13.2. Beispiele Listen

Im Folgenden wird eine Liste gezeigt:

DW OR N PACKAGE (TOP VIEW)	
NC	20
V _{CC}	19 GND
SER IN	18 SER OUT
DRAIN0	17 DRAIN7
DRAIN1	16 DRAIN6
DRAIN2	15 DRAIN5
DRAIN3	14 DRAIN4
SRCLR	13 SRCK
\overline{G}	12 RCK
GND	11 GND
NC – No internal connection	
V_{cc}	Positive supply voltage
GND	Ground
SER IN	Daten Pin
SRCK	Clock Pin
RCK	Latch Pin
\overline{SRCLR}	Wenn shift-register clear LOW ist, werden die input Register gelöscht
\overline{G}	Wenn output enable HIGH ist, werden die Daten im Output Buffer LOW gehalten

Tabelle 13.1.: Aufwändige Tabelle mit Abbildung und Caption

- Ich weiß, dass viele Geräte des täglichen Lebens durch Computer gesteuert werden und kann für mich relevante nennen und nutzen.
 1. Und jetzt eine Numerierung
 2. Und jetzt eine Numerierung
- Ich kann wichtige Bestandteile eines Computersystems (Eingabe-, Ausgabegeräte und Zentraleinheit) benennen, kann ihre Funktionen beschreiben und diese bedienen.

Und jetzt eine Numerierung:

1. Aufzählungspunkt
 - a) Unteraufzählung
 - b) Unteraufzählung
 - Und jetzt noch eine Ebene ohne Aufzählung
 - Und jetzt noch eine Ebene ohne Aufzählung
2. Aufzählungspunkt

3. Aufzählungspunkt
4. Aufzählungspunkt
5. Aufzählungspunkt

13.3. Beispiel Codesequenz

In Listing 13.1 sieht man ein Quick-Sort-Listing in der Programmiersprache JAVA. Das Listings-Paket übernimmt die Formatierung von Codebausteinen und kann in der Präambel nach Belieben auf eine andere Sprache konfiguriert werden.

13.3.1. Quicksort in JAVA

Quelltext 13.1: QuickSort in Java

```
1 public class QuickSort
2 {
3     public static void main(String[] args)
4     {
5         int [] x =
6         {
7             9, 2, 4, 7, 3, 7, 10
8         }
9         ;
10        System.out.println(Arrays.toString(x));
11
12        int low = 0;
13        int high = x.length - 1;
14
15        quickSort(x, low, high);
```

```
16     System.out.println(Arrays.toString(x));
17 }
18
19 public static void quickSort(int[] arr, int low,
20 int high)
21 {
22     if (arr == null || arr.length == 0)
23         return;
24
25     if (low >= high)
26         return;
27
28     // pick the pivot
29     int middle = low + (high - low) / 2;
30     int pivot = arr[middle];
31
32     // make left < pivot and right > pivot
33     int i = low, j = high;
34     while (i <= j)
35     {
36         while (arr[i] < pivot)
37         {
38             i++;
39         }
40
41         while (arr[j] > pivot)
42         {
43             j--;
44         }
45
46         if (i <= j)
47         {
48             int temp = arr[i];
49             arr[i] = arr[j];
50             arr[j] = temp;
51             i++;
52             j--;
53         }
54     }
55
56     quickSort(arr, low, middle - 1);
57     quickSort(arr, middle + 1, high);
58 }
```

```

48         arr[i] = arr[j];
49         arr[j] = temp;
50         i++;
51         j--;
52     }
53 }
54
55 // recursively sort two sub parts
56 if (low < j)
57     quickSort(arr, low, j);
58
59 if (high > i)
60     quickSort(arr, i, high);
61 }
62 }
```

13.4. Beispieltext

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Das hier ist der zweite Absatz. Dies hier ist ein Blindtext zum Testen von Text-

ausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Und nun folgt – ob man es glaubt oder nicht – der dritte Absatz. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Nach diesem vierten Absatz beginnen wir eine neue Zählung. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen

Zweck, da sie eine falsche Anmutung vermitteln.

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Abbildungsverzeichnis

1.1.	.	14
1.2.	.	14
1.3.	Projektnähe tabellarisch	16
1.4.	Projektnähe Grafisch	17
1.5.	Risikomatrix	17
1.6.	.	21
3.1.	.	26
4.1.	.	28
4.2.	.	29
4.3.	.	30
4.4.	.	30
4.5.	.	31
4.6.	.	32
13.1.	Arduino mit Lichtsensor und Lichterkette	46

Tabellenverzeichnis

13.1. Aufwändige Tabelle mit Abbildung und Caption 47

Quelltexte

13.1. QuickSort in Java 48

Literaturverzeichnis

[Ackermann 2001] ACKERMANN, Edith: Piaget's constructivism, Papert's constructionism: What's the difference. In: *Future of learning group publication* 5 (2001), Nr. 3, S. 438. – URL http://lovettresourcenetwork.wiki.lovett.org/file/view/EA.Piaget+_+Papert.pdf. – Zugriffsdatum: 2014-07-09

[Anastopoulou u. a. 2012] ANASTOPOULOU, Stamatina ; BERLAND, Matthew ; FRANT, Janete B. ; BOYTCHEV, Pavel ; BRENNAN, Karen ; CHRONAKI, Anna ; CLAYSON, James ; CORREIA, Secundino ; DAGIENE, Valentina ; DEKOLI, Margarita: Constructionism 2012 Theory Practice and Impact. (2012), August. – URL http://users.uoa.gr/~zsmyrnaiou/conferences_after2008/constructionism%201_2012.pdf. – Zugriffsdatum: 2014-03-26

[Beer, Rudolf und Benischek, Isabella 2011] BEER, RUDOLF ; BENISCHEK, ISABELLA: Aspekte kompetenzorientierten Lernens und Lehrens. In: BIFIE (Hrsg.): *Kompetenzorientierter Unterricht in Theorie und Praxis*. Graz : Leykam, 2011

[Göhlich und Zirfas 2007] GÖHLICH, Michael ; ZIRFAS, Jörg: *Lernen: Ein pädagogischer Grundbegriff*. Stuttgart : Kohlhammer, April 2007. – ISBN 9783170188693

[Harel und Papert 1991] HAREL, Idit ; PAPERT, Seymour: *Situating Con-*

structionism. Norwood, N.J : Ablex Publishing Corporation, U.S., 1991. – ISBN 9780893917869

[Resnick 1996] RESNICK, Mitchel: Distributed constructionism. In: *Proceedings of the 1996 international conference on Learning sciences*, International Society of the Learning Sciences, 1996, S. 280–284. – URL <http://dl.acm.org/citation.cfm?id=1161173>. – Zugriffsdatum: 2015-04-20

A. Anhang-Kapitel

A.1. Anhang-Section

Testtext