

# LEGO GCT

Caner Kara, Dennis Behrendt, Sven Wolf, Tim Sibum, Hannes Scherer, Tim Turowski

3. Mai 2023

## Inhaltsverzeichnis

<b>1 Protokolle</b>	<b>2</b>
1.1 Teammeetings . . . . .	2
1.2 Stundenübersicht . . . . .	2
1.3 Protokolle . . . . .	2
1.3.1 Protokoll zu Termin 1 . . . . .	2
1.3.2 Protokoll zu Termin 2 . . . . .	2
1.3.3 Protokoll zu Termin 3 . . . . .	3
1.3.4 Protokoll zu Termin 4 . . . . .	3
1.4 Projektmitglieder . . . . .	5
1.4.1 Tim Turowski . . . . .	5
1.4.2 Caner Kara . . . . .	5
1.4.3 Sven Wolf . . . . .	5
1.4.4 Dennis Behrendt . . . . .	5
1.4.5 Tim Silbum . . . . .	5
1.4.6 Hannes Scherer . . . . .	5
<b>2 Pflichtenheft</b>	<b>6</b>
2.1 Zielbestimmung . . . . .	6
2.1.1 Musskriterien . . . . .	6
2.1.2 Wunschkriterien . . . . .	6
2.1.3 Abgrenzungskriterien . . . . .	6
2.2 Produkteinsatz . . . . .	7
2.3 Produktfunktionen . . . . .	7
2.4 Produktdaten . . . . .	7
2.4.1 Bauteil . . . . .	7
2.4.2 Bauteil/Set-Preis . . . . .	7
2.4.3 Set . . . . .	7
2.5 Produktleistungen . . . . .	8
2.6 Qualitätsanforderungen . . . . .	8
2.7 Benutzungsoberfläche . . . . .	8
2.8 Nichtfunktionale Anforderungen . . . . .	9
2.9 Technische Produktumgebung . . . . .	9
2.10 Gliederung in Teilprodukte . . . . .	9
2.11 Ergänzungen . . . . .	10
2.12 Globale Testfälle . . . . .	10

# 1 Protokolle

## 1.1 Teammeetings

## 1.2 Stundenübersicht

Termin	Datum	Zeitraum	Stunden	Erledigt
Termin 1	19.04.2022	15:00 - 17:30	2,5	Kickoff Termin + kurzer Austausch
Termin 2	24.04.2022	15:30 - 17:00	1,5	Besprechung zum Pflichtenheft
Termin 3	27.04.2022	12:00 - 15:00	3	Besprechung zum Pflichtenheft
Termin 4	02.05.2022	10:00 - 11:15	0	Präsentation 1. Entwurf Pflichtenheft

## 1.3 Protokolle

### 1.3.1 Protokoll zu Termin 1

Protokollführer: Tim Turowski

Zusätzliche Teilnehmer: Kolja Dunkel, Henning Ahlf

Abwesend: keiner

Entwicklungsdeadline Mitte Dezember

Erste Abgabe: Pflichtenheft nach Balzert

Dafür haben wir 5 Wochen Zeit

Termin vereinbart zum ersten Abgleich mit Kolja Dunkel: 2.Mai 10:00 Uhr

Wir sollen Slag und Git verwenden

Es können Entwicklungsumgebungen angefordert werden, Jira und Confluence

Teamprotokoll muss geführt werden, weil wir etwa 330 Stunden leisten müssen

Die Zeit wird für die Bewertung zur Relation zur Komplexität betrachtet

Am Ende wird es Einzelnoten geben

Wir müssen ausserdem einen Wochenplan aufstellen, in denen sollen wir Urlaube und Vorbereitungszeit für Prüfungen hinterlegen

Es soll ausserdem ein Handbuch erstellt werden

Wenn wir nachdem wir eine Technologien im Pflichtenheft dokumentiert haben, uns danach doch für eine Andere entscheiden, sollten wir das begründen

### 1.3.2 Protokoll zu Termin 2

Protokollführer: Sven Wolf

Zusätzliche Teilnehmer: -

Abwesend: -

Tim Turowski gibt Kurzeinweisung in LaTex und Git

Hauptaufgabe des Treffens: füllen des Pflichtenhefts mit Inhalten

Wir haben uns geeinigt Python nutzen zu wollen um unsere Kenntnisse in der Sprache zu verbessern

Große Diskussionspunkte:

- Eine Historie einfügen? Evtl. mit Benutzeraccounts? -> auf später verschoben (Wunschkriterium)
  - Wie wird die Ausgabe aussehen? Preise für alle Einzelteile? -> Extra Button "mehr Informationen anzeigen"
  - Wie wird die Datenbankstruktur aussehen? (technische Umsetzung) -> muss noch weiter erklärt werden
- Aufgaben/Fragestellungen fürs nächste Treffen:
- Welche und Wieviele Shops vergleichen wir?
  - Wo und wie bekommen wir die Lego-Bauanleitungen?

- Jeder fertigt eine Skizze zur Benutzeroberfläche an
  - Jeder fertigt ein Klassendiagramm an
  - Weitere Vorschläge für den Punkt 'Technische Produktumgebung' im Pflichtenheft finden
- Nächstes Treffen: Donnerstag, der 27.04.2023, 12:00Uhr Remote/Online mit dem Ziel das Pflichtenheft weiter zu füllen + oben genannten Aufgaben zu vergleichen

### 1.3.3 Protokoll zu Termin 3

Protokollführer: Dennis Behrendt

Zusätzliche Teilnehmer: -

Abwesend: -

Meeting per Discord und Github

Weiterarbeit am Pflichtenheft:

Wir haben über die "Produktdaten" diskutiert und diese in ein Klassendiagramm notiert

Außerdem haben wir die "Produktleistungen" besprochen und notiert

- Offenstehende Frage dazu: Wie groß wird das Datenaufkommen sein?

Definierung der "Qualitätsanforderungen"

Vergleich unserer Skizzen zu der "Benutzeroberfläche und Ideensammlung

- Wie soll die Darstellung des Vergleichs (Preise, Teile, Händler) konkret aussehen?

Definieren der "Nichtfunktionalen Anforderungen"

Gemeinsame Recherche zu den Lego-Händlern (Welche Händler wählen wir genau? Bieten diese sowohl Set- sowie Einzelverkauf an? Lassen sich die Preise der Händler crawlen?)

Fragestellung für die Projektgeber:

- Sollen wir für unsere Datenbank an Einzelteilen eine Vorlage verwenden oder diese automatisch per Crawler füllen?

### 1.3.4 Protokoll zu Termin 4

Protokollführer: Tim Sibum

Zusätzliche Teilnehmer: Kolja Dunkel, Henning Ahlf

Abwesend: -

Präsentation der ersten Version des Pflichtenhefts für die Dozenten

Forderung der Dozenten:

- zulösende Problem mit in die Zielbestimmung aufnehmen
- Wie etwas funktionieren soll sollte in die Musskriterien aufgenommen werden
- Priorisieren der Musskriterien
- Benutzerkonten System in Muss aufnehmen
- Mehr Händler berücksichtigen und responsiv Design in Kann aufnehmen
- Muss detaillierter beschreiben
- Automatische Warenkorb Befüllung in Wunsch aufnehmen
- Produktfunktionen nummerieren (Vorbedingung, Beschreibung, Nachbedingung, Möglicher Fehlerfall).
- Use Case Modellierung in Produktfunktion aufnehmen
- Benutzerprofile auch im Datensegment aufnehmen
- Benutzeroberfläche mit Workflow optisch ansprechender modellieren
- Alle Stichpunkte besser in ganzen Sätzen formulieren und W Fragen abdecken
- Skizze zur Technischen Produktumgebung machen

- Installationsanleitung und Handbuch muss nicht unbedingt in das Pflichtenheft
- Extradokument für Zeitennachweis, Protokolle und Pflichtenheft

Aufgaben für die Gruppe:

- Technische Anforderung spezifizieren was brauchen wir?(Datenbank)
- Stundenkontigent auf 25 Wochen aufteilen und als Soll formulieren
- Projektplanung in GitHub organisieren
- Stichpunkte im Pflichtenheft ausformulieren
- erstellen von Diagrammen und Skizzen für das Pflichtenheft

## 1.4 Projektmitglieder

### 1.4.1 Tim Turowski

Datum	Zeitraum	Stunden	Erledigt
22.04.2022	12:00 - 14:00	2	Repository angelegt bei Github + Git-Installation
23.04.2022	18:30 - 22:00	3,5	Latexdokumente angelegt: Protokolle + Lastenheft Muster
28.04.2022	17:30 - 18:30	1	Bilder ins Pflichtenheft eingefügt + weitere Recherchen

### 1.4.2 Carner Kara

Datum	Zeitraum	Stunden	Erledigt
xx.xx.xxx	00:00 - 00:00	0	Hier sollen die Aufgaben aufgelistet sein

### 1.4.3 Sven Wolf

Datum	Zeitraum	Stunden	Erledigt
xx.xx.xxx	00:00 - 00:00	0	Hier sollen die Aufgaben aufgelistet sein

### 1.4.4 Dennis Behrendt

Datum    Zeitraum    Stunden    Erledigt	
21.04.2023    19:00 - 21:00    2    Installation	Einarbeitung in GitHub.

### 1.4.5 Tim Silbum

Datum	Zeitraum	Stunden	Erledigt
20.04.2023	08:30-10:00	1,5	Softwareideen aufschreiben
20.04.2023	12:00-12:30	0,5	Softwareideen aufschreiben
23.04.2023	11:00-11:45	0,75	Testen von Crawlern(Jar)
24.04.2023	11:30-12:30	1	Testen von Crawlern(Jar)
25.04.2023	08:45-09:30	0,75	Repository Klonen und
26.04.2023	14:15-16:00	1,75	ERM und UML Diagramme
27.04.2023	08:45-10:45	2	Sequenzdiagramm, Zustandsdiagramm und GUI Skizze

### 1.4.6 Hannes Scherer

Datum	Zeitraum	Stunden	Erledigt
xx.xx.xxx	00:00 - 00:00	0	Hier sollen die Aufgaben aufgelistet sein

## 2 Pflichtenheft

### 2.1 Zielbestimmung

Heutzutage günstig Lego-Sets zu erwerben kann auf mehreren Ebenen ein wohlüberlegtes Unterfangen sein. Welche der Lego-Händler bieten die Sets günstig an? Ist es günstiger sich die Einzelteile des Sets einzeln zu bestellen? Was ist wenn bestimmte Einzelteile nicht mehr verfügbar sind?

Das Ziel des Projekts ist genau diese Fragestellungen, durch die Entwicklung eines Tools zur Ermittlung der günstigsten Anbieter, zu beantworten. Das Tool soll dabei auch in der Lage sein, abzuwägen, ob es günstiger ist, das Set oder Einzelteile zu kaufen und zu kombinieren.

#### 2.1.1 Musskriterien

- . Interne Teileverwaltung durch das Pflegen einer Datenbank über die Einzelteile.
- . Die PDF-Bauanleitungen der Lego-Sets sollen mit OCR auslesbar sein.
- . Ein Crawler bezieht die Preise der unterschiedlichen Händler.
- . Ein Preisvergleich unter den Händlern, im Bezug auf Kauf eines Sets oder die jeweiligen Einzelteile, findet statt.
- . Darstellung des Tools als graphische Oberfläche. (Startseite, Benuterverwaltung, Preisvergleich, Warenkorb)
- . Es soll möglich sein Benutzeraccounts anzulegen und diese verwalten zu können.
- . Drei feste Händler sollen beim Vergleich berücksichtigt werden.
- . Der Warenkorb soll automatisch befüllt werden.
- . Auf nicht verfügbare Einzelteile sollte hinreichend hingewiesen werden.
- . Die Lieferkosten sollen beim Preisvergleich berücksichtigt werden.
- . Bei der Ausgabe des Vergleichs soll eine Verlinkung zum Produkt, sowie eine Liste der Bauteile, angezeigt werden.

#### 2.1.2 Kannkriterien

Test

#### 2.1.3 Wunschkriterien

Einzelteile die besonders selten/teuer sollten gesondert aufgelistet werden

Filterfunktion um zum Beispiel Figuren rauszufiltern

Mehr Händler berücksichtigen

Grafische Weboberfläche responsive designen

Sticker berücksichtigen

Benutzeraccount mit Wunschliste? (Historie)

Eigene Bauanleitung auslesen lassen

Aktuelle Angebote hervorheben

Dashboard: Statistiken zum Suchverhalten unserer Benutzer / Wie erfolgreich war unser Programm?

#### 2.1.4 Abgrenzungskriterien

Unser Produkt wird keine Verkaufsplattform haben

Keine Bevorzugung von Händlern

Wir berücksichtigen keine nicht offiziellen Klemmbausteine

Website ist nur für deutschsprachige Benutzer

Alte Sets ohne Stückliste in der Anleitung können nicht berücksichtigt werden (2006)

## 2.2 Produkteinsatz

Das Produkt soll Benutzer bei Kaufentscheidung unterstützen, außerdem unterstützt es die Benutzer Sets zu bauen die aus irgendwelchen Gründen nicht mehr verfügbar sind.

Das Tool ist für Lego-Enthusiasten und Sammler gedacht, die nach dem günstigsten Angebot suchen. Das Tool kann auch von Einzelpersonen oder Unternehmen genutzt werden, die große Mengen an Lego-Sets kaufen möchten.

## 2.3 Produktfunktionen

PDFs mit Crawler ansprechen können

PDFs mittels OCR auslesen

Einzelteile in Datenbank zwischenspeichern

Informationen bei Händlern abfragen können

Preisvergleich zwischen Sets und Einzelteilen

Darstellung der Endergebnisse

## 2.4 Produktdaten

### 2.4.1 Bauteil

Element-ID

Name

### 2.4.2 Bauteil/Set-Preis

Element-ID

Preis

Anbieter

URL

### 2.4.3 Set

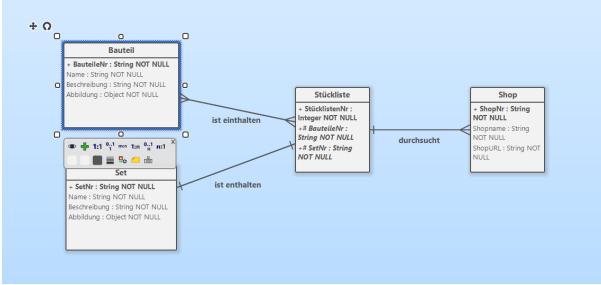
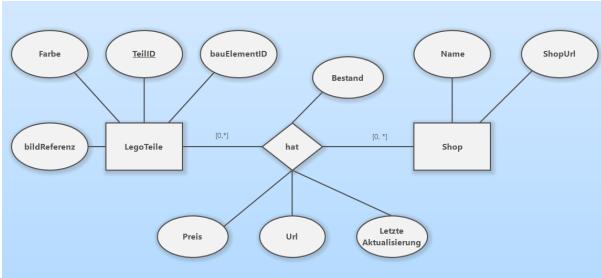
Set-ID

List: Element-ID

Name

Bool: Verfügbarkeit

UVP



## 2.5 Produktleistungen

Bestandsdatenbank soll aufgebaut werden mit Informationen aus den Anleitungen

Preise sollen bei der Suche gecrawlt werden

Die verwendete Datenbank muss in der Lage sein große Datenmengen zu speichern und zu verarbeiten

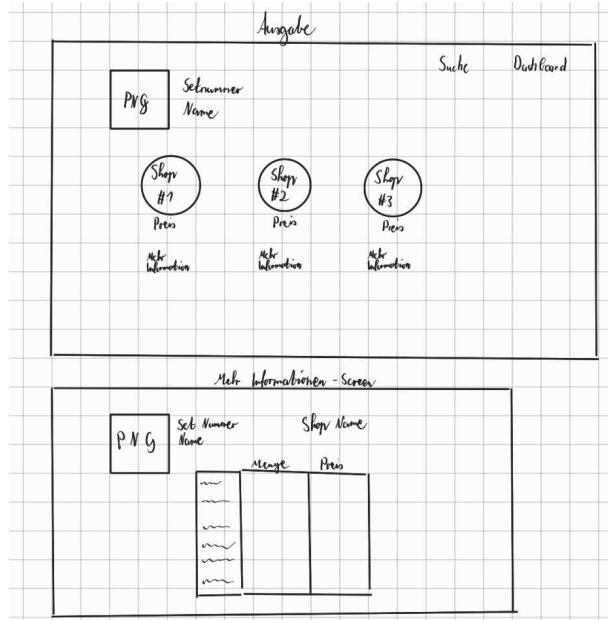
Der Crawler soll in der Lage sein schnell Preisdaten bei den Anbietern zu crawlen, außerdem soll ein Algorithmus regelmäßig die neuen Anleitungen erkennen und dem Parser übergeben

## 2.6 Qualitätsanforderungen

Dokumentation der Änderungshistorie

Programmcode auskommentieren

## 2.7 Benutzeroberfläche




## 2.8 Nichtfunktionale Anforderungen

Barrierefreiheit

Intuitives Design

ISO Normen aus dem MCIW-Modul einhalten

Revisionsinstanzen Scrum etablieren

Websitendaten schützen die gecrawlt werden

## 2.9 Technische Produktumgebung

Datenbank

Crawler

Webserver

OCR

Angular

HTML

CSS

Python

Tex

Git

## 2.10 Gliederung in Teilprodukte

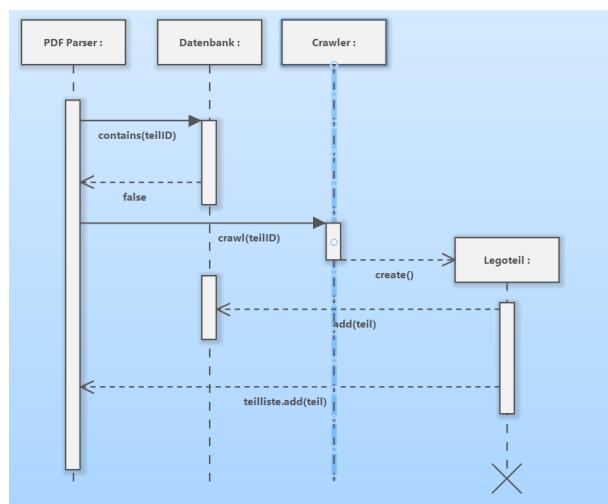
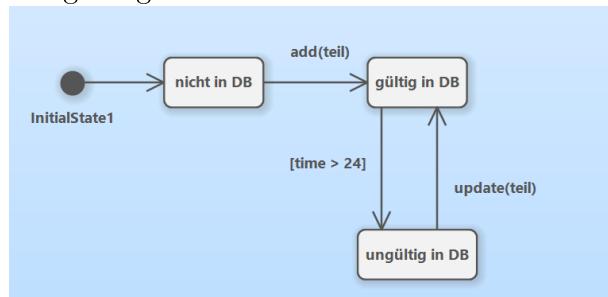
Bauanleitung suchen

Bauanleitung parsen

Datenbank füllen

Preisvergleich

Ausgabe generieren



## 2.11 Ergänzungen

Aus rechtlichen Gründen sollen Anbieter vor dem Crawling angefragt werden

Installationsanleitung und Handbuch sollen bereitgestellt werden

## 2.12 Globale Testfälle

Set händisch recherchieren, um Soll-Wert für Unit-Test zu ermitteln und mit Algortihmusausgabe testen

Set-ID mit Bauanleitung ohne Stückzahl

Gültige Set-ID suchen

Ungültige Set-ID suchen