/T10/ Set suchen: Der Nutzer gibt in der Suchleiste die Lego-Setnummer 75355 ein.

/T20/ Registrieren: Der Nutzer klickt auf „Registrieren“ und registriert sich mit den Daten: -Name: Max Mustermann -E-Mail: mustermann@gmx.de -Passwort: Passwort1!

/T30/ Anmelden: Der Nutzer klickt auf „Anmelden“ und meldet sich mit den in /T20/ benanneten Daten an.

/T40/ Abmelden: Der Nutzer Max Mustermann klickt auf „Abmelden“ um sich wieder abzumelden.

/T50/ Stückliste anzeigen: Der Nutzer gibt die Lego-Setnummer wie in /T10/ ein. Auf der Ergebnisseite klickt der Nutzer auf „Stückliste anzeigen“, um sich die Einzelteile des Sets anzeigen zu lassen.

/T60/ Stückliste minimieren: Der Nutzer agiert wie in /T50/ und klickt darauf auf „Minimieren“,, um die Stückliste wieder einzuklappen.

/T70/ Historie anzeigen: Der Nutzer Max Mustermann meldet sich wie in /T30/ an. Er klickt daraufhin auf „Historie“, um sich seine persönliche Such-Historie anzeigen zu lassen.

/T80/ In der Historie vergangene Suchen erneut durchführen: Der Nutzer Max Mustermann agiert wie in /T70/. Daraufhin klickt er auf die Setnummer 75355 in der Historie (Link), um die Sucher erneut durchzuführen.

/T90/ Passwort ändern: Der Nutzer Max Mustermann klickt auf „Mein Profil“ und dann auf „Passwort ändern“. Er ändert sein Passwort mit den Daten: - Altes Passwort: Passwort1! - Neues Passwort: Passwort2! - Neues Passwort (Wiederholen): Passwort2!

/T100/ Email-Adresse ändern: Der Nutzer Max Mustermann klickt auf „Mein Profil“ und dann auf „Email-Adresse ändern“. Er gibt die Email-Adresse: mustermann2@gmx.de im Textfeld „Neue Email-Adresse“ ein und drückt auf „Speichern“.

/T110/ Nach PDF’s der Legosets suchen: Nach einer festgelegten Dauer sucht der PDF Crawler automatisch auf www.lego.com nach PDF’s von Legosets, die noch nicht in der Datenbank liegen.

/T120/ PDF’s der Legosets auslesen: Nachdem ein PDF laut /T110/ gefunden wurde, wird dieses per OCR ausgelesen und gibt die Informationen über die Einzelteile des Sets weiter.

/T130/ Ausgelesene PDF-Informationen in Datenbank speichern: Nachdem die Informationen laut /T120/ ausgelesen wurden, werden diese in die Datenbank gespeichert.

/T140/ Preise bei Händlern abfragen: Der Nutzer sucht nach dem Lego-Set 13138 (wurde noch nicht gesucht). Der Preis-Crawler bezieht aus den Händlerseiten die jeweiligen Preise der Gesamtsets sowie der Einzelteile des Sets. Diese werden in die Datenbank gespeichert.

/T150/ Einzeltei-Preise berechnen: Nachdem die Preise laut /T140/ abgefragt wurden, wird der Gesamtpreis aller Einzelteile des gesuchten Sets pro Händler berechnet.

/T160/ Benutzeraccount in Datenbank speichern: Nachdem der Nutzer Max

Mustermann sich laut /T20/ registriert hat, werden die Benutzerdaten verschlüsselt in die Datenbank gespeichert.

Benutzerdaten: Die Persönlichen Daten der Benutzer sollen auf der Datenbank gespeichert werden. Zu den Benutzerdaten gehören:

- Username

- E-Mail-Adresse

- Passwort

- Berechtigungen

- Eine Historie der Sets, welche gesucht worden sind

Die Passwörter der Nutzer werden gehashed auf der Datenbank gespeichert. Dies kann die Speicherung von Sensiblen daten sichern. Der Klarname der Nutzer wird nicht gespeichert, sondern lediglich ein Nutzername. Nutzern wird eine Berechtigung zugewiesen. So können einfache Endanwender lediglich die Funktionen nutzen. Die Administratoren hingegen haben zusätzlich den Zugriff aus einem Dashboard mit Statistiken zu den Suchanfragen.

Result: Das Result Objekt beinhaltet alle Informationen, welche bei dem gesamten Prozess zusammengetragen worden sind. Dazu gehört eine Repräsentation des gesuchten Sets und dessen Einzelteile. Zudem verwaltet Result die gefundenen Shop Angebote der Einzelteile des Sets. Das Result Objekt verfügt über eine Funktion die Einzelteile zu Bewerten. Dabei soll ein Wert geliefert werden, welcher den Einzelteilpreis, die Häufigkeit im Set, den Set preis und gegebenenfalls die Verfügbarkeit des Teils einbezieht. Dieser Wert dient als Indikator, ob das Einzelteil besonders ist. Die Metadaten über die Suchanfrage werden auch in Result gespeichert.

Lego Set: Ein Lego Set Objekt repräsentiert ein Lego Set und seine Einzelteile. Es Verwaltet auch den aktuellen Wert des Sets und die Verfügbarkeit. Eine URL zum Download der PDF wird auch in diesem Objekt verwaltet.

Einzelteil: Einzelteil Objekte bilden die einzelnen Legoteile ab. Dabei wird die Design-ID, die ein Lego Teil Eindeutig identifiziert gespeichert. Außerdem kann noch ein Name für das Bauelement hinterlegt werden.

Einzelteilmarktwert Die Objekte vom Typ Einzelteilmarktwert sollen zu einem bestimmten Lego Teil jeweils den Entsprechenden Preis eines Bestimmten Anbieters speichern. Zudem verwaltet es die URL zu dem Angebot und den Zeitpunkt der Peis Abfrage. Wenn der Shop eine Bestands Angabe macht, ist auch möglich diese in dem Objekt zu speichern.

Bauanleitung: Bauanleitungen sind PDF-Dateien. Diese PDF-Dateien haben auf den Letzten Seiten eine Stückliste der Einzelteile. Die PDF-Dateien werden nicht gespeichert und nach dem Parsen gelöscht.

4.2 Produktfunktionen

F120 wirklich Lego.com & Datenbank Anwender? PDF von Lego bereits gecrawelt und abgelegt wird im nächsten Prozess

2.1 - manchmal "soll"-Formulierung, manchmal "so wirds gemacht". Alles zu zweiter Variante umgeformt

2. "werden mit OCR ausgelesen" anstelle "sollen mit OCR auslesbar sein"

6. "Es ist möglich Benutzeraccounts anzulegen und diese zu verwalten"

7. "Drei feste Händler werden beim Vergleich berücksichtigt"

8. "Auf nicht verfügbare Einzelteile wird hinreichend hingewiesen"

9. "Die Lieferkosten werden beim Preisvergleich berücksichtigt" - überhaupt Notwendig?

Zusätzliche Speicherstruktur notwendig (eingabe des Landes und gff. sogar Bundesland)

10. "Bei der Ausgabe des Vergleichs wird eine Verlinkung zum Produkt, sowie eine Liste der Bauteile, bereitgestellt"

2.2 - selbe wie bei 2.1

1. "Plattform wird auf Mobilgeräten angepasst dargestellt"

2. "Mehr Händler sind implementierbar"

2.3

1. "Besonders teure/seltene Einzelteile werden gesondert aufgelistet"

2. "Eine Filterfunktion um zum Beispiel Figuren auszufiltern"

4. "Eigens kreierte Bauanleitung sind ebenfalls auslesbar"

5. "Aktuelle Angebote werden hervorgehoben"

6. "Die ausgegebene Stückliste wird automatisch auf Wunsch in einen Warenkorb des gewählten Shops umgewandelt"

2.4

3. "Wir berücksichtigen nur offizielle Klemmbausteine"

5. "Alte Sets (vor 2006 erschienen) ohne Stückliste in der Anleitung werden nicht berücksichtigt"

7. "Bauanleitungen werden nur von der offiziellen Lego-Webseite gecrawelt"

3 Produkteinsatz

"bei der Kaufentscheidung"

Letzter Satz: "Sowohl Einzelpersonen alsauch Unternehmen können die Website nutzen um große Mengen an Lego-Sets zu kaufen"

4.1

F10 Geschäftsprozess: "Über eine Suchmaske Lego Setnummer eingeben"

Fehlerfall: "... überein, oder ..."

F20 Nachbedingung: "... registrieren lassen, sein Benutzer..."

Beschreibung: "Um den vollen Funktionsumfang der Webapplikation nutzen zu können muss der Benutzer eine Registrierung durchführen"

F50 Vorbedingung: "Der Benutzer drückte auf "Stückliste anzeigen" Button"

Fehlerfall: "Verbindung zur Datenbank nicht vorhanden"

F70 Fehlerfall: "Keine"

F80 Geschäftsprozess: "Suche aus Historie erneut durchführen"

F90 Vorbedingung: "User klickte auf "Passwort ändern" und gab neues Passwort ein

F100 Vorbedingung: "User klickte auf "Email-Adresse ändern" und gab neue Email-Adresse ein

F110 Beschreibung: "Stand" und "Abständen" großer erster Buchstabe

F120 Vorbedingung: "Es wurden neue PDFs gefunden"

Beschreibung: "zuzuordnen ist, wird sie in der Datenbank aufgenommen"

F130 Vorbedingung: "Neue PDF wurde ausgelesen"

F140 Vorbedignung: "Setnummer und Einzelteile des Sets sind in der Datenbank vorhanden"

6 Produktleistungen

"Bestandsdatenbank wird mit Informationen aus Lego-Anleitungen aufgebaut.

Die entsprechenden Preise werden regelmäßig gecrawelt und ebenfalls gespeichert.

Zudem wird regelmäßig nach neuen Lego-Anleitungen gecrawelt um das Tool aktuell zu halten.

Die Datenbank wird in der Lage sein die großen Datenmengen zu verwalten und auszugeben."

8 Technische Produktumgebung

"Mithilfe von SQLAlchemy in Python wird die Datenbank auf einem Webserver aufgesetzt und verwaltet.

Python-Crawler werden für die Preisabfrage und die PDF suche eingesetzt.

Um die Stückliste aus den PDF-Anleitung auszulesen kommt ein OCR Parser zum Einsatz.

Angular dient als Frameworkerstellungstool für eine interaktive Front End.

Zur Aufbesserung des Frameworks werden HTML und CSS genutzt.

Python bilded die primäre Programmiersprache.

Tex dient der Dokumentation des gesamten Projekts.

Die gesamte Verwaltung und der Austausch innerhalb des Projekts wird mittels einem Git-Repocitory durchgeführt."