

Datenbanken-Projekt: Retrogames

Sören Größer – s0554662

Erik Dölling – s0554601

Fabian Rod – s0546837

Fachbereich 1
Datenbanken
Modul C29
Wintersemester 2018/19

Dozent: Prof. Dr. Thomas Baar

Inhaltsverzeichnis

Einleitung	4
Ziel des Datenbanken-Projektes	4
Beschreibung	4
Projektthema	4
Ziel des Retrogame Projekts	5
Use Case Diagram	5
Projektverlauf	6
Ablauf	6
Aufgabenverteilung	8
Meilensteine	9
RetroGames Crawler	9
Ebay Skript	9
Gui	9
Schwierigkeiten	10
Sören	10
Erik	10
Fabian	10
Implementierung	11
DB-Modell	11
Screenshots / Logs	12
Auswertung	16
Lesson Learned	16

Abbildungsverzeichnis

Abbildung 1 - Grafikrevolution	4
Abbildung 2 - Eye-Catcher Retrogames Prinzip	5
Abbildung 3 - Screenshot der Datenbank Verwaltung	12
Abbildung 4 - Aufruf des programmierten Spiders/ Crawlers	13
Abbildung 5 - Letztes Spiel vom Crawler und Spider closed	13
Abbildung 7 - Datenbank GUI	15
Abbildung 8 - Datenbank-GUI Startfenster	15
Log 1 - Auszug der Ausführung des Ebay Skriptes	12
Quellcode 1 - Ausschnitt aus dem Spider Code mit Kommentaren	14
Quellcode 2 - DB-Login	14
Tabelle 1 - Datenstruktur Formalien	7
Tabelle 2 - Aufgabenverteilung	8
Tabelle 3 - Ausschnitt aus der 3996 Titel großen CSV-Datei	13
Diagramm 1 - Use Case Diagramm	5
Diagramm 2 - Projektaufbau erstellt im Jour-Fix	6

Einleitung

Ziel des Datenbanken-Projektes

In der folgenden Projektarbeit gilt es sowohl die im Modul Datenbanken erlernten Fähigkeiten anzuwenden, als auch Projektmanagement zu betreiben und sich sinnvoll ein komplexes Projekt unter den Mitarbeitern (Kommilitonen) aufzuteilen.

Dies bedeutete im Groben die Findung, Planung, Aufteilung, Rücksprachen, Problembewältigung, Zusammenführung, Auswertung zwischen Erik, Sören und Fabian als Teil des Retrogame Projekts.

Beschreibung

Projektthema

Pong wurde 1972 von Atari als Videospielautomat entwickelt und gilt allgemein als Begründer des kommerziellen Videospiels. Die Herstellungskosten betrugen damals nur 500 US-Dollar. Zum Vergleich das plattformübergreifende Spiel Destiny wurde 2014 für 436 Millionen Euro produziert. Doch nicht nur in Kosten gab es einen Boom, sondern auch in der technischen Entwicklung. Während Spiele im Jahr 1977 mit 128 Farben und einer Prozessorgeschwindigkeit von 1,19 MHz auskamen sind 16,7 Millionen Farben und 3,5 GHz der Standard.



Abbildung 1 - Grafikrevolution

Die Grafik entwickelte sich immer weiter, jedoch waren die Spielmechaniken meist von den Klassikern abgeschaut. Vielen sind genau diese Mechaniken wichtiger als eine „polierte“ Grafik. Doch die „Gemeinde“ der Retro Game Spieler wächst und hält die klassischen Spiele

in Ehren und der ein oder Andere gibt dafür einige Summen aus um ein Original von damals in den Händen zu halten und sich in die Vitrine zu stellen. Und genau da beginnt unser Retrogame Projekt.

Ziel des Retrogame Projekts



Abbildung 2 - Eye-Catcher Retrogames Prinzip

In unserem Projekt geht es darum dem Benutzer eine Datenbank zu liefern, in dem alle Retro Spiele aufgelistet sind. Dort kann er schauen ob es Ebay Auktionen dazu gab und zu welchem Preis das Spiel über die virtuelle Ebay-Theke ging.

Da sich Retrogames über mehrere Konsolen Gruppen, PC-Klassen und Epochen unterteilen lassen und ein enormes Projekt darstellen. Haben wir im Folgenden alle Retrogames vom Amiga genommen.

Use Case Diagram

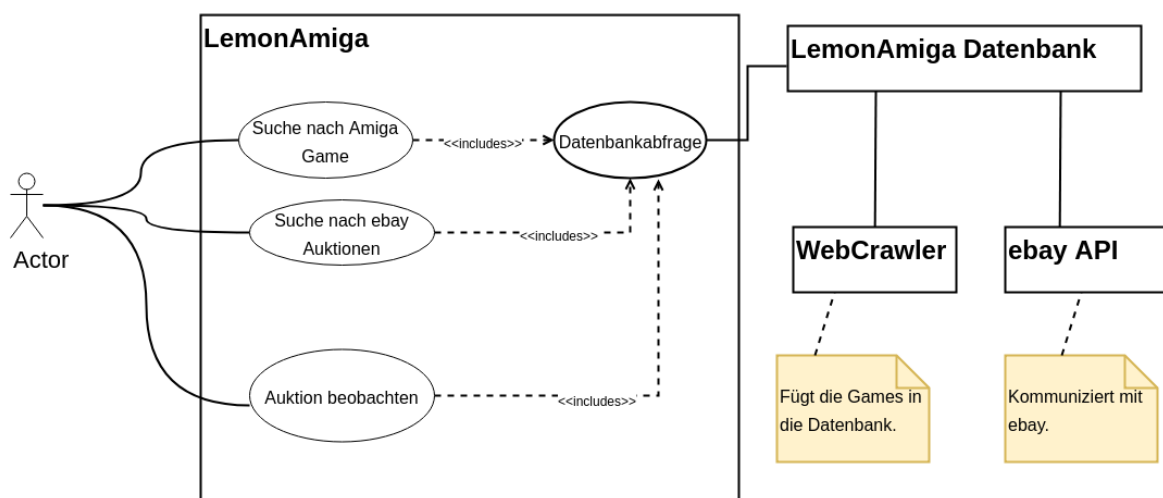


Diagramm 1 - Use Case Diagramm

Projektverlauf

Ablauf

„It starts with an idea...”

Zuallererst musste sich die Gruppe in Rücksprache mit Herrn Baar auf die Thematik und den Umfang der Datenbank einig werden. In der ersten Jour-Fix Sitzung wurde dafür Brainstorming betrieben, in dem alle Ideen und Interessen notiert und in einem Schaubild angeordnet wurden. Es gab von vornherein den Plan Daten von Internetseiten auszulesen und diese so zu verarbeiten, dass sie sich in einer Datenbank verwalten ließen.

In der zweiten Jour-Fix Sitzung wurde sich dann darauf festgelegt die Lemonamiga.com Retrogrades Internetseite zu crawlen und zu speichern. Damit es für 3 Personen etwas komplexer werden würde, sollten diese Daten in einer Datenbank mit den Angeboten aus einer Ebay API abgeglichen werden um Preise und eventuelle Trends ablesen zu können.

Die nächste Jour-Fix Sitzung diente zur sinnvollen Aufteilung der Abschnitte. Zur Klärung mit welchen Mitteln kommuniziert und Daten ausgetauscht werden sollten, Tipps und Ideen für Frameworks und eine Festlegung der zu liefernden Datensätze. Der Aufbau wurde festgelegt wodurch die Schnittpunkte klar abgegrenzt und die Teilbereiche aufgeteilt werden konnten.

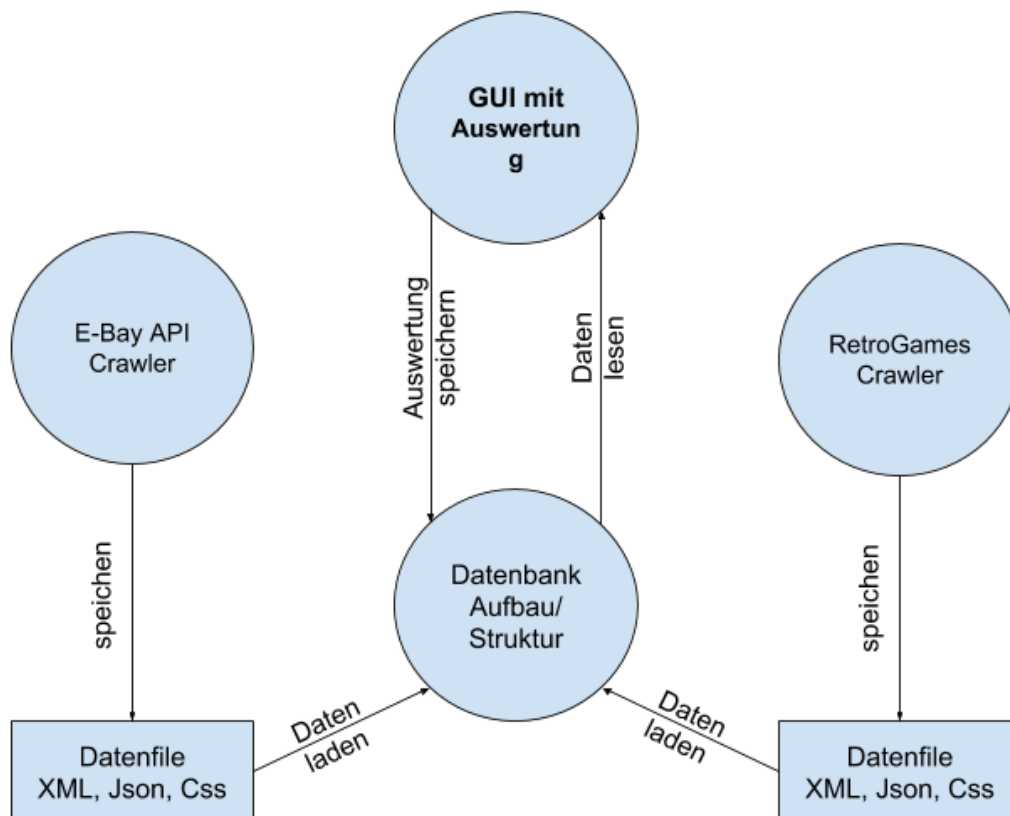


Diagramm 2 - Projektaufbau erstellt im Jour-Fix

Die Gruppe einigte sich darauf die Daten aus den einzelnen Crawlern in einem Datenfile zu speichern, welches dann die Datenbank importieren und verwalten könne.

Des Weiteren wurde Trello, ein Google Drive Share und eine WhatsApp Gruppe eingerichtet. Trello diene zur Verwaltung der Teil Aufgaben und in Verbindung mit Google Drive um gleiche Testdaten zu nutzen und andere Dokumente auszutauschen.

Jeder begann für seinen Bereich Recherchen durchzuführen und selbständig zu entscheiden in welche Programmiersprache, welchem Framework, welchem Datenbanktyp er arbeiten möchte.

Der nächste Jour-Fix diene zum Austausch der Erkenntnisse und es wurden Formalien und Hinweise ausgetauscht:

Formalien/ Hinweise:

- Unicode wird benutzt (ä, ö, ß)
- Electron könnte für die GUI genutzt werden
- Bei Crawlern darauf achten, den Server nicht zu crashen durch zu hoher Rate der Anfrage
- BeautifulSoup 4 könnte als Python Modul genutzt werden, für den Crawler der RetroSeite
- Testdaten immer veröffentlichen

Abbildung 2 - Ausschnitt aus einem Jour-Fix Protokoll - Formalien und Hinweise

Des Weiteren wurde die Datenstruktur für die CSV Dateien festgelegt, damit parallel an der Datenbank gearbeitet werden konnte ohne bereits Datensätze zu haben.

Retrogame:	Titel, Genre, Jahr, Entwickler, Mehr/Single-Player, Kurzbeschreibung, Hardware, Diskanzahl, Sprache, Bildurl (Thumbnail)
Ebay-API:	GameTitle, AuctionID, AuctionTitle, Price, Location, URL, Stat (Auktion-Status), EndTime

Tabelle 1 - Datenstruktur Formalien

Im folgenden Verlauf arbeitete jeder weitgehend selbständig an seinem Projektabschnitt.

Die GUI wurde mit dem Electron Framework erstellt. Das Framework basiert auf Chromium und Node.js, um mit Hilfe von HTML, JavaScript und CSS plattformübergreifende Desktop-Anwendungen zu entwickeln.

Die dem Projekt zugrunde liegende Datenbank wurde mit MySQL erstellt.

Für den Internetseiten Crawler wurde das Framework Scrapy in einer virtuellen Umgebung mit Python 3.7 installiert und erste Versuche gestartet irgendwelche Daten von einer Internetseite zu bekommen. Für die Ebay-Daten musste der Zugriff auf die Ebay-API geklärt werden. Diese lieferte zu viele Daten und Informationen und musste im weiteren Verlauf mit einem Python Script gesteuert werden.

Der nächste Jour-Fix klärte Probleme mit den unterschiedlichen Programmiersprachen und es wurde der jeweilige Status der Projektteile ausgetauscht.

Die Datenbank bekam nun Ihre festgelegte Struktur und konnte erste Daten verarbeiten.

Der Retrogames Crawler lieferte erste noch fehlerhafte Daten in einer CSV Datei. Dopplungen, fehlende Datensätze und fehlende Optimierung für parallel ablaufendes Pipeline Verfahren.

Das Script für die Ebay-API kämpfte mit Sonderzeichen in Produktbeschreibungen, dem reduzieren der enormen Datenflut von ganz Ebay.de und der Aufbereitung. Außerdem befüllte das Script nun direkt die Datenbank ohne den Weg über ein separates CSV-File zu gehen.

Im Folgenden wurde jeder seiner Probleme habhaft und die ersten lauffähigen Programme brachten der Datenbank Datensätze, welche erfolgreich angezeigt und verarbeitet wurden konnten.

Aufgabenverteilung

Fabian Rod	Webseiten Crawler, Verwaltung der Projekt Management Plattform, Dokumentation
Sören Größer	GUI Entwicklung, Verwaltung und Einrichtung der MySQL Datenbank
Erik Dölling	Verwaltung des Ebay-Developer Accounts, Ebay-API Skript

Tabelle 2 - Aufgabenverteilung

Meilensteine

RetroGames Crawler

1. Virtuelle Umgebung für Python 3.7 und Scrapy Framework einrichten ✓
2. Scrapy Shell starten und Daten Response einer Testseite bekommen ✓
3. Spider Script anpassen; Robot Abwehr umgehen und erster Zugriff ✓
4. Spider Ausbau und direktes speichern über Konsolen Befehl in CSV ✓
5. Ersten 250 Spiele in CSV speichern mit noch fehlerhaften Datensätzen ✓
6. Durch Nutzung des Pipelining, Dopplungen Filtern und Parallel crawlen ✓
7. Weiteren Links Folgen um Reviews zu laden X

Ebay Skript

1. Registrierung als Ebay Developer und Erhalt des Application Access Key ✓
2. Verbindung zur Datenbank aufbauen und Spiele Title beziehen ✓
3. erfolgreicher API-Request ✓
4. Request Ergebnis durch Filter verbessern ✓
5. neue Datensätze in die Datenbank schreiben ✓
6. beendete/ausgelaufene Auktionen in Datenbank ändern (teilweise)

Gui

1. Verbindung zur eigenen Datenbank mit eigenem Programm ✓
2. Einlesen der CSV-Dateien und Einfügen in die Datenbank ✓
3. Einbinden von SQL-Queues zum Auslesen und Sortieren in der GUI ✓
4. einfaches und übersichtliches Design für den Anwender erstellen ✓
5. Statistiken zu den Amiga-Spielen und Ebay-Auktionen ✓
6. Suchfunktion X

Schwierigkeiten

Sören

Für das Erstellen der GUI haben wir uns auf eine Webanwendung geeinigt. Da jedoch keine Vorkenntnisse in der Entwicklung dieser vorhanden waren, insbesondere nicht das Einbinden von SQL in JavaScript, erwies sich das ganze als sehr zeitintensiv. Kaum vorhandene Tutorials auf diesem Gebiet erschwerten dies zusätzlich. Das Zusammenarbeiten von mehreren Programmiersprachen war bisher auch ein eher unbekanntes Konzept.

Besonders das Ausführen von SQL-Queries aus JavaScript heraus ist umständlich und erfordert viel Beachtung. Anfragen welche mit MySQL ohne weiteres möglich sind, mussten angepasst und umständlich beschrieben werden um das gleiche Ergebnis zu erzielen. Auch war es nicht möglich Ebay-Titel automatisch den Titeln aus der Lemonamiga-Tabelle zuzuweisen, weshalb wir in diesem Fall mit dem "LIKE"-Operator gearbeitet haben.

Erik

Bei der Entwicklung des Ebay Skriptes erwies sich die Komplexität der von Ebay zur Verfügung gestellten API als Herausforderung. Die Optimierung des itemFilters im API-Request, durch Ausnutzung aller Informationen aus der Datenbank zu einem Spiel, führte meist auf sehr wenige oder gar keine Ergebnisse. Die größte Anzahl zutreffender Ergebnisse ließ sich durch eine simple Suche des Spiele Titels plus dem Keyword *Amiga* und der Eingrenzung Kategorie *Games* erzielen.

Weiter waren die Nutzungsbeschränkungen für Entwickler irritierend. Laut der Beschreibung durch Ebay sind nur 500 Anfragen / Tag möglich. Außerdem wurden kurze Anfrage-Intervalle schnell mit einem Block der IP für zunächst 15 Minuten quittiert (pot. DDoS Attacke). Als Workaround wurde nach jedem Request, die erhaltenen Auktionen direkt in die Datenbank geschrieben. Dies führte zu einem ausreichend langen und unregelmäßigen Delay und die Anfragen flogen unter dem Radar der DDoS Protection.

Fabian

Dieses Projekt wies sich in seinen Teilbereichen recht komplex. Um Datensätze von Internetseiten zu beziehen und aufzuarbeiten gibt es recht gute Frameworks welche meist auf Python basieren. Mit Python hatte ich erst oberflächlich zu tun und schon allein das Aufsetzen der richtigen Bibliothek mit passender Version auf einem MAC brachte vermehrt Probleme. Dadurch entschied ich mich für eine virtuelle Umgebung nur alleine für dieses Projekt. Scrapy besitzt wieder seine eigene komplexe Syntax um von Internetseiten bestimmte Daten aus der HTML Struktur zu beziehen, welche manchmal im Link, danach oder in einem Fließtext verbaut sind. Hinzu schien die Seite ein Update bekommen zu haben,

dadurch war leider die Struktur des Quellcodes der Internetseite unterschiedlich und es erwies sich als schwierig wiederkehrende Informationen zum Download zu finden. Weitere Probleme gab es mit Time Out durch die vermehrte Anfrage des Spiders. Auch waren nicht alle Seiten ID's, welche der Crawler ablief besetzt. Dadurch wurde als Default Seite immer 1 bestimmtes Spiel unter unterschiedlichen ID's gespeichert, was natürlich zu Dopplungen führte.

Implementierung

DB-Modell

Für das Abschließende Datenbankdesign wurde auf ein relationales Design verzichtet. Als Verbesserungsvorschlag wäre die Verwendung eines *Foreign Key* in der EBAY Tabelle zu erwähnen.

FOREIGN KEY (GAMETITLE) REFERENCES LEMONAMIGA(TITLE)

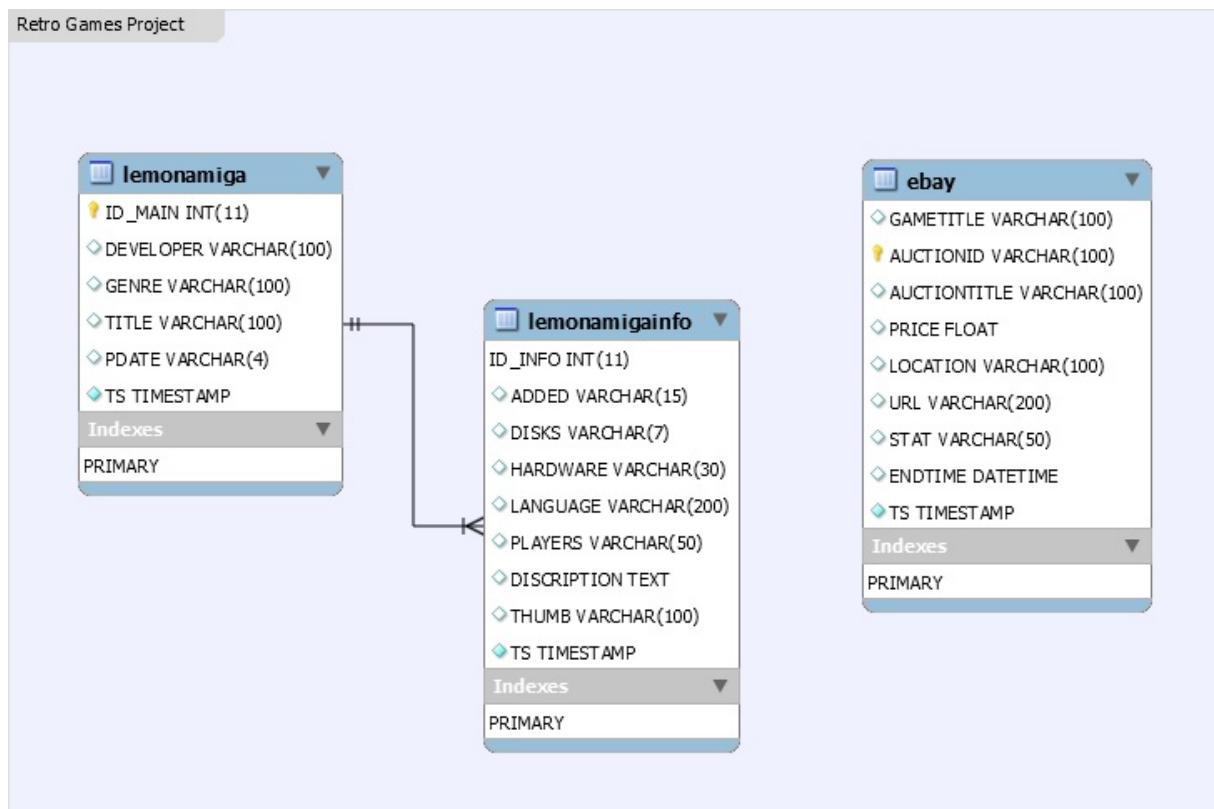


Diagramm 3 - Letzter Stand des Datenbank Designs

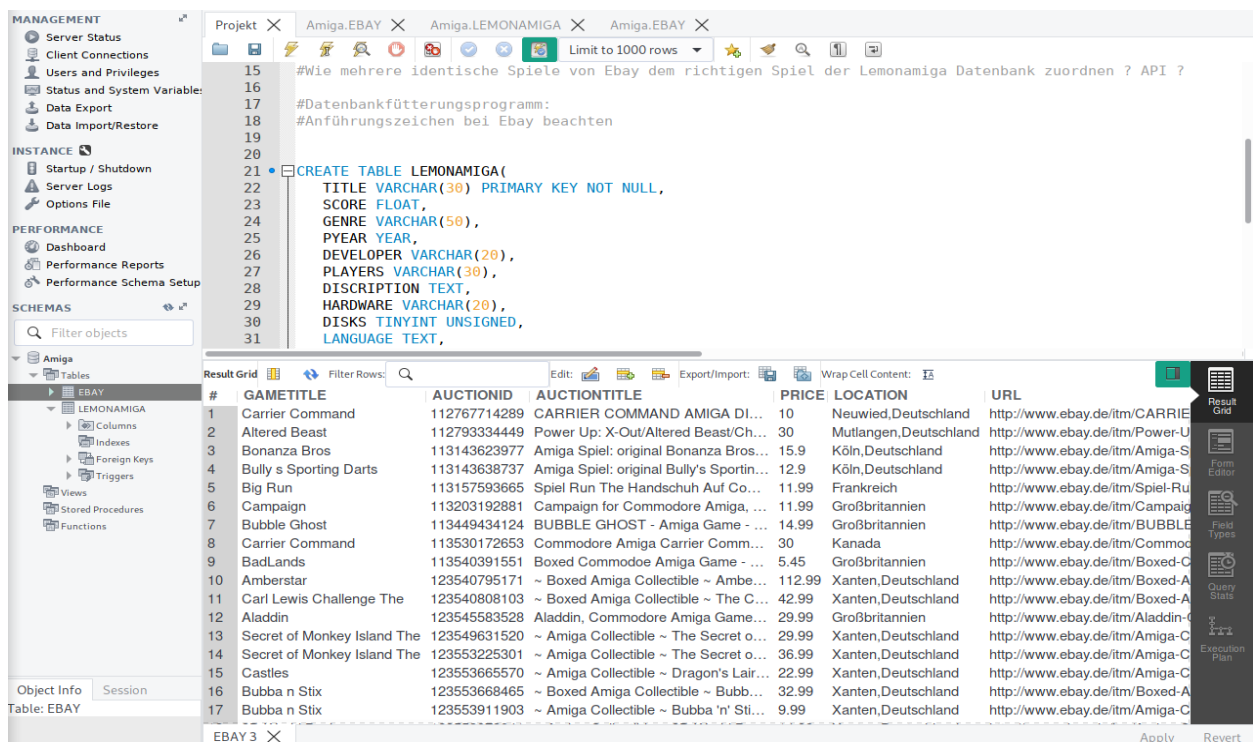
Screenshots / Logs

```
/home/erik/PycharmProjects/LemonamigaEbay/venv/bin/python
/home/erik/PycharmProjects/LemonamigaEbay/start_ebay.py
```

```
### => Start
### => Database connected
### => Ebay connected
### => 0 were set to inactive
No auctions found for 1000 Miglia 1927 1933 Volume
Keywords: Amiga 1000 Miglia 1927 1933 Volume
No auctions found for 1869 History Experience Part
Keywords: Amiga 1869 History Experience Part
No auctions found for 1943 The Battle of Midway
Keywords: Amiga 1943 The Battle of Midway
2 was inserted
...
..
.

No auctions found for Simpsons The Bart vs the Sp
Keywords: Amiga Simpsons The Bart vs the Sp
No auctions found for Simpsons The Bart vs the Wo
Keywords: Amiga Simpsons The Bart vs the Wo
2 was inserted
4 was inserted
### => Finish
```

Log 1 - Auszug der Ausführung des Ebay Skriptes



The screenshot shows a database management interface with a left sidebar containing navigation options like 'Server Status', 'Client Connections', 'Users and Privileges', 'Data Export', 'Data Import/Restore', 'INSTANCE', 'PERFORMANCE', and 'SCHEMAS'. The main window displays a SQL script for creating a table named 'LEMONAMIGA' with columns: TITLE (VARCHAR(30), PRIMARY KEY NOT NULL), SCORE (FLOAT), GENRE (VARCHAR(50)), PYEAR (YEAR), DEVELOPER (VARCHAR(20)), PLAYERS (VARCHAR(30)), DISCRIPTION (TEXT), HARDWARE (VARCHAR(20)), DISKS (TINYINT UNSIGNED), and LANGUAGE (TEXT). Below the script, a 'Result Grid' shows a list of Amiga games with columns: #, GAMETITLE, AUCTIONID, AUCTIONTITLE, PRICE, LOCATION, and URL. The table contains 17 rows of data.

#	GAMETITLE	AUCTIONID	AUCTIONTITLE	PRICE	LOCATION	URL
1	Carrier Command	112767714289	CARRIER COMMAND AMIGA DI...	10	Neuwied, Deutschland	http://www.ebay.de/itm/CARRIE
2	Altered Beast	112793334449	Power Up: X-Out/Altered Beast/Ch...	30	Mutlangen, Deutschland	http://www.ebay.de/itm/Power-U
3	Bonanza Bros	113143623977	Amiga Spiel: original Bonanza Bros...	15.9	Köln, Deutschland	http://www.ebay.de/itm/Amiga-S
4	Bully's Sporting Darts	113143638737	Amiga Spiel: original Bully's Sportin...	12.9	Köln, Deutschland	http://www.ebay.de/itm/Amiga-S
5	Big Run	113157593665	Spiel Run The Handschuh Auf Co...	11.99	Frankreich	http://www.ebay.de/itm/Spiel-Ru
6	Campaign	113203192881	Campaign for Commodore Amiga, ...	11.99	Großbritannien	http://www.ebay.de/itm/Campaig
7	Bubble Ghost	113449434124	BUBBLE GHOST - Amiga Game - ...	14.99	Großbritannien	http://www.ebay.de/itm/BUBBLE
8	Carrier Command	113530172653	Commodore Amiga Carrier Comm...	30	Kanada	http://www.ebay.de/itm/Commod
9	Bad Lands	113540391551	Boxed Commodore Amiga Game - ...	5.45	Großbritannien	http://www.ebay.de/itm/Boxed-C
10	Amberstar	123540795171	~ Boxed Amiga Collectible ~ Ambe...	112.99	Xanten, Deutschland	http://www.ebay.de/itm/Boxed-A
11	Carl Lewis Challenge The	123540808103	~ Boxed Amiga Collectible ~ The C...	42.99	Xanten, Deutschland	http://www.ebay.de/itm/Boxed-A
12	Aladdin	123545583528	Aladdin, Commodore Amiga Game...	29.99	Großbritannien	http://www.ebay.de/itm/Aladdin-C
13	Secret of Monkey Island The	123549631520	~ Amiga Collectible ~ The Secret o...	29.99	Xanten, Deutschland	http://www.ebay.de/itm/Amiga-C
14	Secret of Monkey Island The	123553225301	~ Amiga Collectible ~ The Secret o...	36.99	Xanten, Deutschland	http://www.ebay.de/itm/Amiga-C
15	Castles	123553665570	~ Amiga Collectible ~ Dragon's Lair...	22.99	Xanten, Deutschland	http://www.ebay.de/itm/Amiga-C
16	Bubba'n Stix	123553668465	~ Boxed Amiga Collectible ~ Bubba...	32.99	Xanten, Deutschland	http://www.ebay.de/itm/Boxed-A
17	Bubba'n Stix	123553911903	~ Amiga Collectible ~ Bubba'n Sti...	9.99	Xanten, Deutschland	http://www.ebay.de/itm/Amiga-C

Abbildung 3 - Screenshot der Datenbank Verwaltung

```
[(ENV) MAC-NOM-NOT-WAR:tutorial macbookpro$ scrapy crawl lemonamiga_games -o test.csv -t csv
```

Abbildung 4 - Aufruf des programmierten Spiders/ Crawlers

```
2019-01-11 20:28:38 [scrapy.core.scraper] DEBUG: Scraped from <200 http://www.lemonamiga.com/games/details.php?id=4306>
{'date_added': ['Dec 23, 2018'],
 'developer': ['Teque Software'],
 'disks': ['1'],
 'genre': ['Brain - Logical'],
 'hardware': ['OCS', 'ECS'],
 'language': ['English'],
 'players': ['1 or 2'],
 'review': 'Keine Infos',
 'thumbnail': ['http://www.lemonamiga.com/games/screenshots/full/revelation!_01.png'],
 'titel': ['Revelation!'],
 'year': ['1991']}
2019-01-11 20:28:38 [scrapy.core.engine] INFO: Closing spider (finished)
2019-01-11 20:28:38 [scrapy.extensions.feedexport] INFO: Stored csv feed (3964 items) in: test.csv
2019-01-11 20:28:38 [scrapy.statscollectors] INFO: Dumping Scrapy stats:
{'downloader/request_bytes': 1765955,
 'downloader/request_count': 4306,
 'downloader/request_method_count/GET': 4306,
 'downloader/response_bytes': 104877364,
 'downloader/response_count': 4306,
 'downloader/response_status_count/200': 4306,
 'finish_reason': 'finished',
 'finish_time': datetime.datetime(2019, 1, 11, 19, 28, 38, 643761),
 'item_dropped_count': 342,
 'item_dropped_reasons_count/DropItem': 342,
 'item_scraped_count': 3964,
 'log_count/DEBUG': 8271,
 'log_count/INFO': 17,
 'log_count/WARNING': 342,
 'memusage/max': 92581888,
 'memusage/startup': 65687552,
 'response_received_count': 4306,
 'scheduler/dequeued': 4306,
 'scheduler/dequeued/memory': 4306,
 'scheduler/enqueued': 4306,
 'scheduler/enqueued/memory': 4306,
 'start_time': datetime.datetime(2019, 1, 11, 19, 32, 627418)}
2019-01-11 20:28:38 [scrapy.core.engine] INFO: Spider closed (finished)
MAC-NOM-NOT-WAR:tutorial macbookpro$
```

Abbildung 5 - Letztes Spiel vom Crawler und Spider closed

	date_added	developer	disks	genre	hardware	language	players	review	thumbnail	titel	year
3.834	Nov 11, 201		1	Miscellaneous - (unknown)	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Alehop		2017
3.835	Nov 22, 201		1	Platformer - Multi Screen	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Charlie Chir		1995
3.836	Nov 14, 201	Amiten	1	Miscellaneous - (unknown)	OCS,ECS	English,Spanish	1 or 2, Simultaneous	Keine Infos	http://www.i. Blue Escape		2017
3.837	Nov 22, 201		1	Platformer - Multi Screen	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Charlie Chir		1995
3.838	Nov 22, 201		1	Brain - Puzzle	OCS,ECS	English,German	1 Only	Keine Infos	http://www.i. Pairs		1995
3.839	Nov 22, 201	Frentic Soft	1	Brain - Miscellaneous	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Soko Ban		1994
3.840	Nov 22, 201	Comatose	5	Shoot'em Up - Crosshair	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Prototype		1997
3.841	Nov 22, 201		1	Platformer - Multi Screen	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Father Tuck		1996
3.842	Nov 22, 201	Pacific	1	Platformer - Multi Screen	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Desperation		1997
3.843	Nov 22, 201		1	Arcade - Tetris	OCS,ECS	English	1 or 2, Simultaneous	Keine Infos	http://www.i. Pot Panic		1991
3.844	Nov 22, 201		1	Arcade - Tetris	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Stack Up		1991
3.845	Nov 22, 201		1	Platformer - Single Screen	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Adventures		
3.846	Nov 22, 201	Nexus		Brain - Puzzle	AGA,RTG,CD-ROM	English	1 or 2, Simultaneous	Keine Infos	http://www.i. Puzzle Bobs		2001
3.847	Nov 22, 201	Amigan Soft	1	Arcade - Miscellaneous	AGA,RTG	English	1 to 5, Simultaneous	Keine Infos	http://www.i. Koules v1.4		2008
3.848	Nov 22, 201	Vorlon	1	Shoot'em Up - V-Scrolling	AGA,CD-ROM	English	1 or 2, Simultaneous	Keine Infos	http://www.i. Ultra Violent		1998
3.849	Nov 22, 201		1	Adventure - Graphics/Text	CDTV	English	1 Only	Keine Infos	http://www.i. Sherlock Ho		1991
3.850	Nov 22, 201		1	Shoot'em Up - Multi-Scrolling	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Slez		2017
3.851	Nov 22, 201		1	Simulation - Marine	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Operation S		1990
3.852	Nov 22, 201		1	Brain - Puzzle, (Adult)	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Femme Fata		1989
3.853	Nov 22, 201		1	Brain - Puzzle, (Adult)	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Femme Fata		1989
3.854	Dec 23, 201		1	Arcade - Labyrinth/Maze	OCS,ECS	English	2	Keine Infos	http://www.i. Derring-Do		1994
3.855	Dec 23, 201		1	Brain - Miscellaneous	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Impsbru		2017
3.856	Nov 22, 201	Bohewia Sof		Adventure - Point and Click	OCS,ECS	Czech	1 Only	Keine Infos	http://www.i. 7 dni a 7 no		1994
3.857	Dec 23, 201		1	Brain - Miscellaneous	OCS,ECS	English,German	1 Only	Keine Infos	http://www.i. Soko		1993
3.858	Dec 23, 201	Pacific	1	Brain - Miscellaneous	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Push 'n' Sho		1995
3.859	Nov 30, 201		1	Brain - Puzzle	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Mirror		2017
3.860	Dec 23, 201	TSA	3	Shoot'em Up - First-person 3D	OCS,ECS	Polish	1 Only	Keine Infos	http://www.i. Project Batt		1995
3.861	Dec 23, 201	Amiga Wave	1	Arcade - Beat'em Up	OCS,ECS	English	1 or 2, Simultaneous	Keine Infos	http://www.i. Else we get		2017
3.862	Dec 16, 201		1	Shoot'em Up - Space Invaders	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Terrahawks		2017
3.863	Dec 1, 2017		1	Platformer - Scrolling Screen	ECS,AGA	English	1 Only	Keine Infos	http://www.i. Elfie The Un		2018
3.864	Dec 23, 201	Silly Softwa	1	Adventure - Miscellaneous	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Mobsters Cit		1994
3.865	Dec 23, 201	Harlequin	1	Arcade - Boulder Dash	AGA	English	1 Only	Keine Infos	http://www.i. George and		1998
3.866	Dec 23, 201	Backyard Br		Arcade - Worm/Snake	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Super Nibble		1997
3.867	Dec 29, 201	Ralza Softw	1	Racing - Cars	OCS,ECS	English	1 Only	Keine Infos	http://www.i. Roadtrip		2017
3.868	Dec 31, 201		1	Arcade - Labyrinth/Maze	OCS,ECS	English,German	1 Only	Keine Infos	http://www.i. Power Labyr		1995
3.869	Dec 31, 201	Amiten	1	Platformer - Scrolling Screen	ECS,CD-ROM	English	1 Only	Keine Infos	http://www.i. Dream of Ro		2017
3.870	Jan 20, 201	JAM Produ		Shoot'em Up - First-person 3D	AGA	English	1 Only	Keine Infos	http://www.i. Blake Stone: 2017		
3.871	Dec 23, 201			Arcade - Boulder Dash	OCS,ECS	English,German	1 Only	Keine Infos	http://www.i. Blake Stone: 2017		

Tabelle 3 - Ausschnitt aus der 3996 Titel großen CSV-Datei

```

24 #Der Spider der aufgerufen wird mit dem passenden Namen Lemonamiga_games
25 class QuotesSpider(scrapy.Spider):
26     name = "lemonamiga_games"
27
28     #in Python werden die Funktionen durch def erstellt
29     #hier werden die Seiten von Lemonamiga über eine For Schleife abgelaufen
30     #jeder Seitenaufruf geht in den Parser sammelt die Daten und wandert dann weiter in die Pipe
31     def start_requests(self):
32         urls = ['http://www.lemonamiga.com/games/details.php?id=' + str(i) for i in range(1, 4307)] #1 bis 4307 leider mit Doppelungen
33         for url in urls:
34             yield scrapy.Request(url=url, callback=self.parse)
35
36     # Jeder Request erstellt ein Item und befüllt ihn mit Daten
37     def parse(self, response):
38
39         item = LemonamigaItems()
40         #Ausgewählte Erklärung der Syntax
41         #der Titel steckt in der HTML in einem Text nach der class mit der Bezeichnung "textGameHeader"
42         #es wird bis zum tag font vorgesprungen und der Text extrahiert
43         #das wird dann in LemonamigaItems im Feld Titel gespeichert.
44         item['titel'] = response.xpath("//strong[@class='textGameHeader']/font/text()").extract()
45         item['genre'] = response.xpath("//text()[contains(., 'Genre:')]//following-sibling::td/a/text()").extract()
46         item['year'] = response.xpath("//a[contains(@href, 'year')]/strong/font/text()").extract()
47         #hier enthält der Fließtext das Wort "Developer:" und über following-sibling wird nach der Form td/a gesucht und der enthaltende Text
48         #extrahiert.
49         item['developer'] = response.xpath("//text()[contains(., 'Developer:')]//following-sibling::td/a/text()").extract()
50         item['review'] = response.xpath("//text()[contains(., 'Developer:')]//following-sibling::td/a/text()").extract() # empty
51         item['hardware'] = response.xpath("//text()[contains(., 'Hardware:')]//following-sibling::td/a/text()").extract()
52         item['disks'] = response.xpath("//text()[contains(., 'Disks:')]//following-sibling::td/a/text()").extract()
53         item['language'] = response.xpath("//text()[contains(., 'Language:')]//following-sibling::td/a/text()").extract()
54         #hier steckt die Information im @src Links der Image Datei
55         item['thumbnail'] = response.xpath("//img[contains(@src, 'games/screenshots')]/@src").extract()
56         item['date_added'] = response.xpath("//a[contains(@href, 'date')]/text()").extract()
57
58
59     #uber yield (ähnlich wie Return, bloß es wird nicht weiter vorgehalten und nach Übergabe gelöscht)
60     yield item
61
62

```

Quellcode 1 - Ausschnitt aus dem Spider Code mit Kommentaren

```

////////////////////////////////////
//Verbindung mit der Datenbank

function sqllogin() {

    console.log("Nummer 2");

    var mysql = require('mysql');

    var connection = mysql.createConnection({
        host: "141.45.91.40",
        user: "s0554662",
        password: "illhvufj",
        database: "s0554662_Projekt"
    });

    connection.connect(function (err) {
        if (err) {
            console.log(err.code);
            console.log(err.fatal);
        }
    });
    return connection;
}

```

Quellcode 2 - DB-Login

Datenbank-Projekt: Retrogames

Retrogames

Amiga-Spiele anzeigen Aktualisieren Statistiken

Titel	Jahr	Genre	Entwickler	
1000 Miglia: 1927-1933 Volume 1	1991	Racing - Cars		mehr
1869: History Experience Part I	1992	Strategy - Trading	Max Design	mehr
1943: The Battle of Midway	1989	Shoot'em Up - V-Scrolling	Probe Software	mehr
1st Division Manager	1991	Sports - Football/Soccer	Cirrus Software	mehr
3001: O'Connors Fight	1990	Strategy - Miscellaneous		mehr
3D World Boxing	1992	Sports - Boxing		mehr
4D Sports Boxing	1991	Sports - Boxing	Distinctive Software Inc.	mehr
4D Sports Driving	1990	Racing - Cars	Distinctive Software Inc.	mehr
4th & Inches	1988	Sports - American Football	Sculptured Software	mehr
4x4 Off-Road Racing	1988	Racing - Cars		mehr

Abbildung 6 - Datenbank GUI

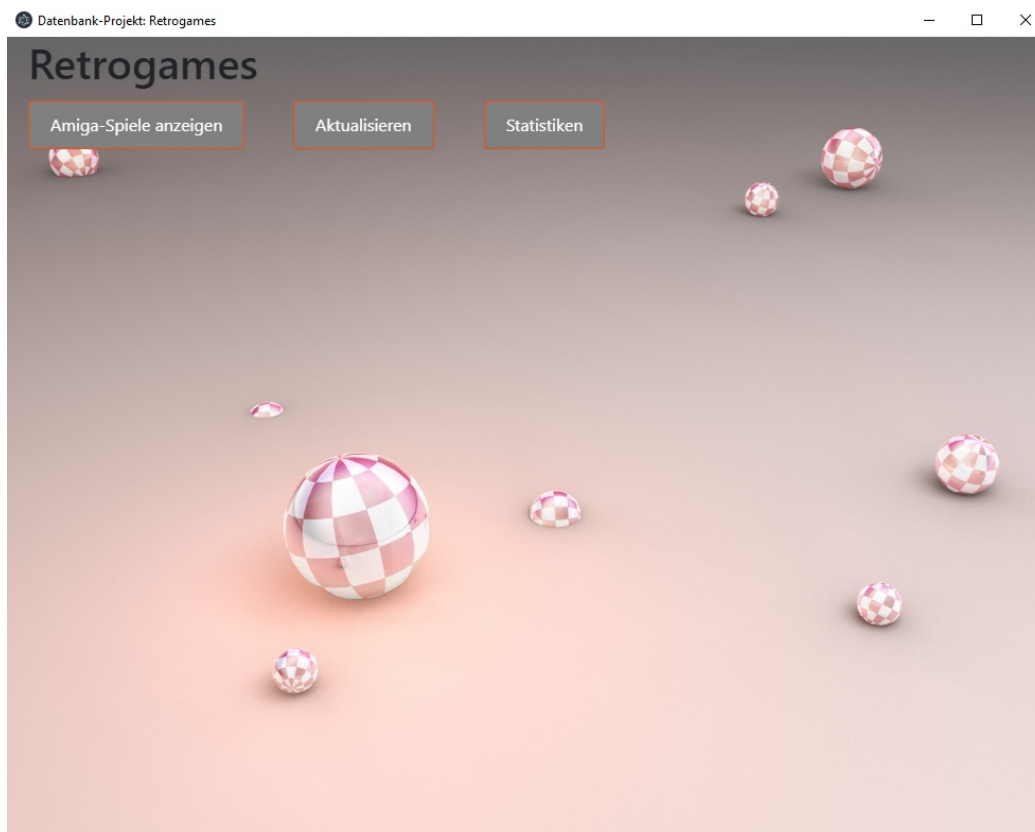


Abbildung 7 - Datenbank-GUI Startfenster

Auswertung

Der Weg von der Idee zum fertigen Produkt ist langwierig. Obwohl wir sowohl Kunde als Softwareentwickler waren, neigte jeder einzelne dazu über den anfänglich geforderten Anforderungen, Optimierungen und Funktionen hinzuzufügen und verrannte sich teilweise zeitlich in Details.

Dabei war die stetige Kommunikation und der Austausch über die am Anfang eingerichteten Portale (WhatsApp, Trello, Drive) stets konstruktiv mit vielen Feedbacks über Status und Problemen.

Die Protokollierung während des Jour-Fix war anfänglich eher mühselig, doch zeigten bald ihren Vorteil, da sie Online einsehbar waren und jedem einzelnen Formalien, festgelegte Strukturen und die Schnittstellen Vereinbarung stets zur Sichtung dienten. Dadurch gab es keine Fehlkommunikation. Die Datenbankstruktur die Parallel implementiert wurde, konnte nach Vorlage der CSV Dateien, problemlos einlesen und verarbeiten.

Das heißt trotz neuer Programmiersprachen und unbekannten Frameworks war jeder in der Lage sein Teilbereich zur funktionalen Reife auszuarbeiten und das Ergebnis ist eine Datenbank, welche unseren Anforderungen gerecht wird und funktionstüchtig arbeitet bei stets aktualisierbarer Datengrundlage.

Dieses Projekt könnte man nun weiterführen und weitere Queues für die Datenbank einfügen. Die Python Skripte ließen sich in Exe-Dateien umwandeln und von dem JavaScript ausführen lassen, wodurch die Datenbank selbständig sich die neuen Daten liefern lassen könnte. Am Ende könnte der ganze Prozess in ein App implementiert werden, die dann wiederum auf einem Handy jedem die aktuellen Gebote, die Preisentwicklung oder einfach nur die Datenbank zur Spielefindung liefert.

„... and finishes here.“

Lesson Learned

Auf der Suche nach geeigneten Tools, Frameworks, Programmiersprache für unser Projekt ging es zunächst darum, für die Probleme optimierte Werkzeuge zu finden und dieses auf dem eigenen Rechner lauffähig zu bekommen. Dazu war es oftmals erforderlich neue Wege zu gehen und in besonderen Fällen Libraries von anderen Projekten abzugrenzen und virtuelle Umgebungen zu erstellen. Dabei haben wir gelernt, wie man gezielt in unbekannten Sprachen Ergebnis orientiert lernt zu programmieren ohne erst die ganze Sprache oder den Funktionsumfang des Tools erlernen zu müssen.

Schwierigkeiten gab es vor allem bei dem Einbinden von externen API's oder den Zugriff auf Internetseiten zum Crawlen. Es galt dutzende Stellschrauben anzufassen und Probleme von Zugriffen und Rechten aus dem Weg zu räumen. Dabei konnten wir unser Verständnis zur Kommunikation über bestimmte Protokolle vertiefen.

Zudem haben wir bei dem Bearbeiten der Aufgaben die Stärken und Schwächen der einzelnen Gruppenmitglieder feststellen können, und dementsprechend die erforderlichen Aufgaben sinnvoll verteilt. Durch diese Rollenverteilung war eine effizientere Arbeitsweise möglich. Die Erfahrung der Rollenverteilung wird noch bei weiteren Gruppenprojekten hilfreich sein.

Für unser Projekt war es erforderlich, verschiedenste Diagramme zu erstellen. Somit war es uns möglich, im Unterricht erlerntes Wissen praxisnah an unser ausgewähltes Projekt umzusetzen. Dabei haben wir unsere Kenntnisse in der Erstellung von Use-Cases, sowie DB-Modell vertieft. Auch haben wir gelernt, sinnvoll erforderliche Programme auszuwählen, welche uns hilfreich erscheinen, und sich in diese einzuarbeiten und sicher umzugehen. Darunter fallen z.B. Umlut, oder von Gsuit bereitgestellte Diagramm Tools.

Vor allem das Beschäftigen mit neuer Software, neuen Tools und Frameworks wird uns für zukünftige Projekte, sowohl in der Universität als auch im Berufsleben, von Hilfe sein. Gerade im Berufsleben ist es wichtig, sich in neue Umgebungen einzuarbeiten, Programmiersprache zu erlernen und verschiedenste Schnittstellen zur Kommunikation der Programme zu nutzen.

Natürlich konnten die Zugriffe, das Erstellen, das Aktualisieren und das Implementieren von Queues für Datenbanken aus dem Unterricht direkt in das Projekt einfließen und man hatte die Chance teilweise besser zu verstehen, warum manche Wege so sinnvoller sind.