

## LieferApp -Onlineshop-Database

### Milestone 1: Requirements Analysis & Conceptual Design

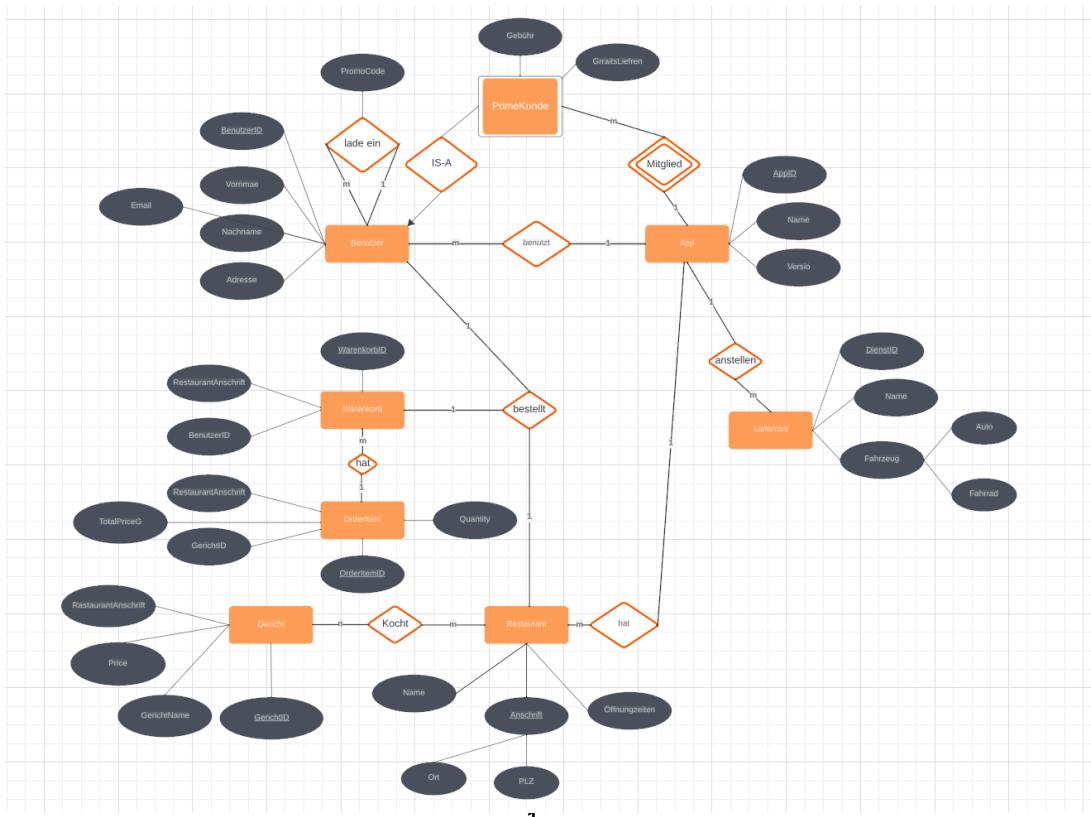
Das Modell beschreibt einen Online-Lieferdienst, der sowohl für gewöhnliche als auch für Premiumkunden geeignet ist. Es gibt eine Liste der Speisen, die in verschiedenen Restaurants angeboten werden, und es ordnet Lieferaufträge a relevante Lieferanten. Die spezifischen Beziehungen werden wie folgt beschrieben:

Der Lieferservice verwaltet mehrere Benutzer, von denen einige auch Premium-Kunden sein können. Zahlreiche Kunden können im Service ein breites Angebot an Gerichten verschiedener Restaurants bestellen. Der Lieferservice beauftragt die Lieferanten, diese Bestellungen zuzustellen. Ein Benutzer kann mehrere andere Benutzer mit PromoCode werben, aber nur ein anderer Benutzer kann ihn werben.

**Warenkörbe** sind eine zentrale Funktion des Lieferservices. Jeder **Warenkorb** wird durch eine eindeutige **Warenkorbid** identifiziert und ist einem **Benutzer** über die **BenutzerID** zugeordnet. Ein **Warenkorb** kann mehrere **OrderItems** enthalten, die die ausgewählten **Gerichte** repräsentieren. Die **Restaurantanschrift** im **Warenkorb** bezieht sich auf das Restaurant, von dem die Bestellung aufgegeben wird. Zusätzlich beinhaltet der **Warenkorb** Informationen zu **LiefernKosten**, was die Liefergebühren für die gesamte Bestellung darstellt.

**OrderItems** repräsentieren die individuellen Artikel in einem **Warenkorb**. Jedes **OrderItem** hat eine eigene **OrderItemID** und verweist über **Warenkorbid** auf den zugehörigen **Warenkorb**. Es verweist auch auf das spezifische **Gericht** durch die **GerichtID**. Jedes **OrderItem** enthält Informationen wie **Price** (Preis des Gerichts), **Quantity** (Menge des Gerichts) und **TotalPriceG** (Gesamtpreis für das Gericht basierend auf der Menge).

Figure 1: Entity Relationship Diagramm



## Milestone 2: Logical Design

FK- Notation:

Relation + Punkt (.) + Fremdschlüsselattribut - verweist auf (◊) - Relation + Punkt (.) + Primärschlüsselattribut z.B. Tabelle1.FK ◊ Tabelle2.PK

**App** (AppID, Version, AppName)

PK = AppID

**Benutzer**(BenutzerID, Vorname, Nachname, Email, Adresse, PromoCode, Telefonnummer, Gebuehr, GratisLiefern)

PK = KundenID

**Restaurant** (Anschrift, RestaurantName, RestaurantName, Ort, PLZ, )

PK = Anschrift

**Gericht** (GerichtID, RestaurantAnschrift, GerichtName, Price)

PK = (GerichtID)

FK = Gericht.RestaurantAnschrift ◊ Restaurant.Anchrift

**Lieferrant** (LieferrantID, LieferrantName, FahrzeugTyp)

PK = (LieferrantID)

**Warenkorb** (WarenkorbID, BenutzerID, RestaurantAnschrift, LiefernKosten)

PK = WarenkorbID

FK = Warenkorb.BenutzerID ◊ Benutzer.BenutzerID

FK = Warenkorb.RestaurantAnschrift ◊ Restaurant.Anchrift

**OrderItem** (OrderItemID, WarenkorbID, GerichtID, Price, GerichtID, TotalPriceG)

PK = OrderItemID

FK = OrderItem.WarenkorbID ◊ Warenkorb.WarenkorbID

FK = OrderItem.GerichtID ◊ Gericht.GerichtID

**Kochen** (GerichtID, RestaurantAnschrift)

PK = (GerichtID, RestaurantAnschrift)

FK = Kochen.GerichtID ◊ Gericht.GerichtID

FK = Kochen.RestaurantAnschrift ◊ Restaurant.Anchrift

## Milestone 4: Implementatio

Für den Programmierteil meines Projektes, habe ich für das Befüllen der Datenbank Java gewählt und für die Webseite PHP.

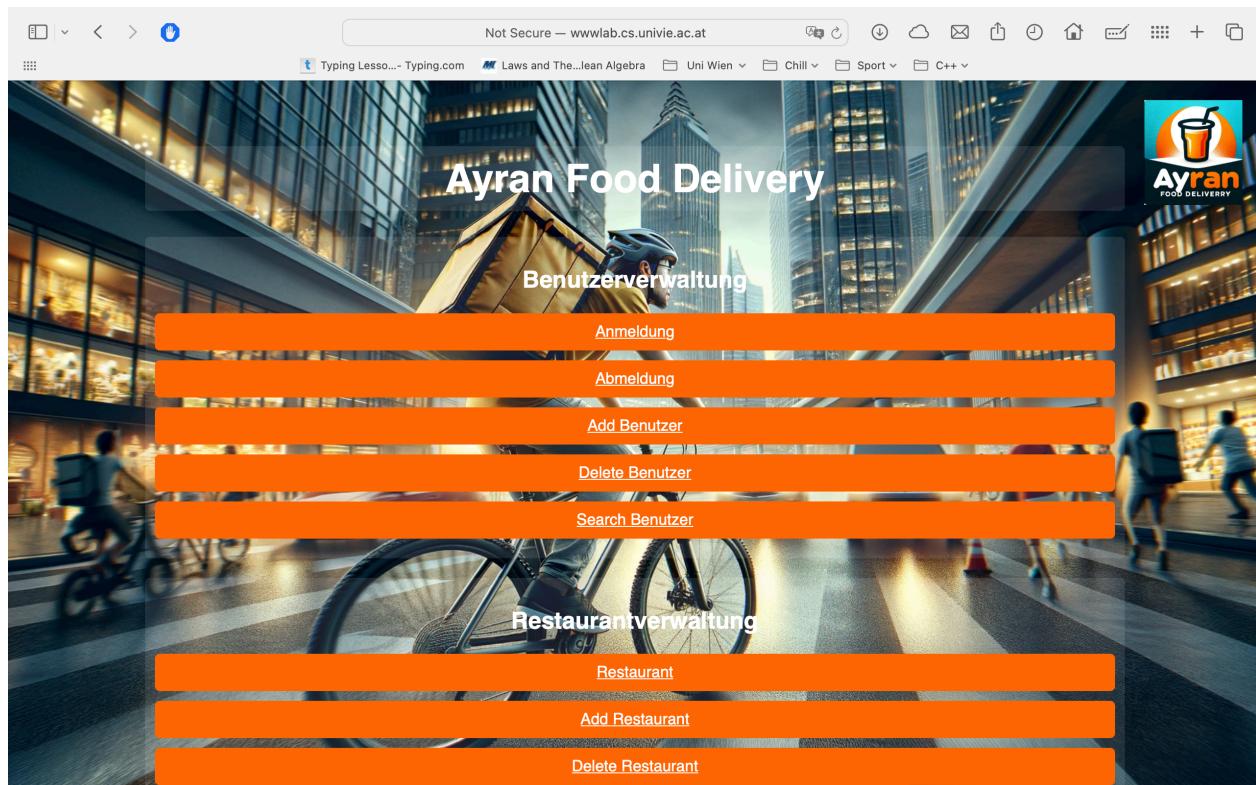
### Java

Für das Befüllen der Datenbank habe ich JavaSqlExample (Class) verwendet. Jedes Relationenschema wurde mit einer eignen Funktion befüllt und lautet auch gleichnamig dazu.

### PHP

Die Webseite meiner Datenbank habe ich mit PHP erstellt. Es ist sehr schlicht und simpel gestaltet. Zu beginn ist die Übersicht der Agierenden zu sehen. Im Grunde sind alle sehr ähnlich aufgebaut und strukturiert. Jede Seite besitzt auch einen Zurück-Button, der fixiert ist am Bildschirm.

Die Website meiner Datenbank wurde mithilfe von PHP erstellt. Der Aufbau meiner Website ist sehr übersichtlich gehalten, um ein hohes Maß an Benutzerfreundlichkeit zu gewährleisten. Für jeden Table meiner Datenbank, wurden jeweils drei, bzw. vier PHP-Files erstellt. Ein File ist für die Ausgabe von Daten (Search) und für die Benutzeroberfläche verantwortlich, die restlichen drei Files jeweils für die „Delete“, „Update“ und für die „Add“ Funktion.



index.php



The screenshot shows a web browser window with the URL "Not Secure — wwwlab.cs.univie.ac.at". The page content is an orange-themed login form for "Ayran FOOD DELIVERY". The form contains fields for "Benutzer-ID" and "E-Mail", and a large orange "Anmelden" button.

## Login.php

The screenshot shows a web browser window with the URL "Not Secure — wwwlab.cs.univie.ac.at". The page content is an orange-themed user registration form titled "Add Benutzer". The form includes fields for "Vorname", "Nachname", "Adresse", "Email", "Vorwahl" (with a dropdown menu showing "+43 Österreich"), "Telefonnummer", "PromoCode" (with a note "Optional"), and a checkbox for "PrimeKunde werden (15€ Gebühr)". A large orange "Add Benutzer" button is at the bottom.

## addPerson.php



Not Secure — wwwlab.cs.univie.ac.at

Typing Lesson... - Typing.com Laws and The...lean Algebra Uni Wien Chill Sport C++

« Zurück

Delete Benutzer:

BenutzerID:

Password:

**Delete Benutzer**

Ayran FOOD DELIVERY

## delPerson.php

Not Secure — wwwlab.cs.univie.ac.at

Typing Lesson... - Typing.com Laws and The...lean Algebra Uni Wien Chill Sport C++

**Benutzer Search:**

BenutzerID:

Vorname:

Nachname:

Adresse:

Email:

Kunden mit Premiummitgliedschaft anzeigen:

**Search**

**Benutzer Search Result:**

BenutzerID	Vorname	Nachname	Adresse	Email	PromoCode	Telefonnummer	Gebuehr
33321	Vorname1	Nachname1	Adresse1	email1@example.com	PROMO8939	0123456133	15

Ayran FOOD DELIVERY

## Search.php



Not Secure — wwwlab.cs.univie.ac.at

Typing Lesson... - Typing.com Laws and The...Lean Algebra Uni Wien Chill Sport C++

## Warenkorb

Search Warenkorb anzeigen Zurück zur Start Seite

WarenkorID	BenutzerID	RestaurantAnschrift	LiefernKosten
7644	33765	Anschrift37	3
7645	34052	Anschrift89	0
7646	34170	Anschrift18	0
7647	33592	Anschrift257	0
7648	34231	Anschrift309	0
7649	33654	Anschrift137	3
7650	34213	Anschrift203	0
7651	33998	Anschrift62	3
7652	33636	Anschrift254	3
7653	33961	Anschrift116	0
7654	33751	Anschrift241	0
7655	33607	Anschrift96	3
7656	33573	Anschrift299	0
7657	33859	Anschrift178	3
7658	33421	Anschrift108	0



## IndexWW.php

Not Secure — wwwlab.cs.univie.ac.at

Typing Lesson... - Typing.com Laws and The...Lean Algebra Uni Wien Chill Sport C++

## OrderItem

Search Warenkorb anzeigen Zurück zur Start Seite

OrderItemID	WarenkorID	GerichtID	Price	Quantity	TotalPriceG
5723	8855	238811	59	5	295
5724	8598	240454	57	3	171
5725	8877	238617	57	4	228
5726	7808	238683	45	3	135
5727	9280	241542	43	2	86
5728	9339	239283	52	3	156
5729	9522	239187	53	3	159
5730	8413	240035	48	3	144
5731	8176	239697	13	5	65
5732	9462	239830	45	1	45
5733	9060	238819	51	3	153
5734	9037	239219	22	2	44
5735	8657	241524	51	1	51
5736	7900	241321	19	5	95
5737	9159	241023	47	4	188



## IndexO.php



Not Secure — wwwlab.cs.univie.ac.at

Typing Lesso... Typing.com Laws and The...ean Algebra Uni Wien Chill Sport C++

« Zurück

**Durchschnittspreis der Gerichte eines Restaurants abfragen**

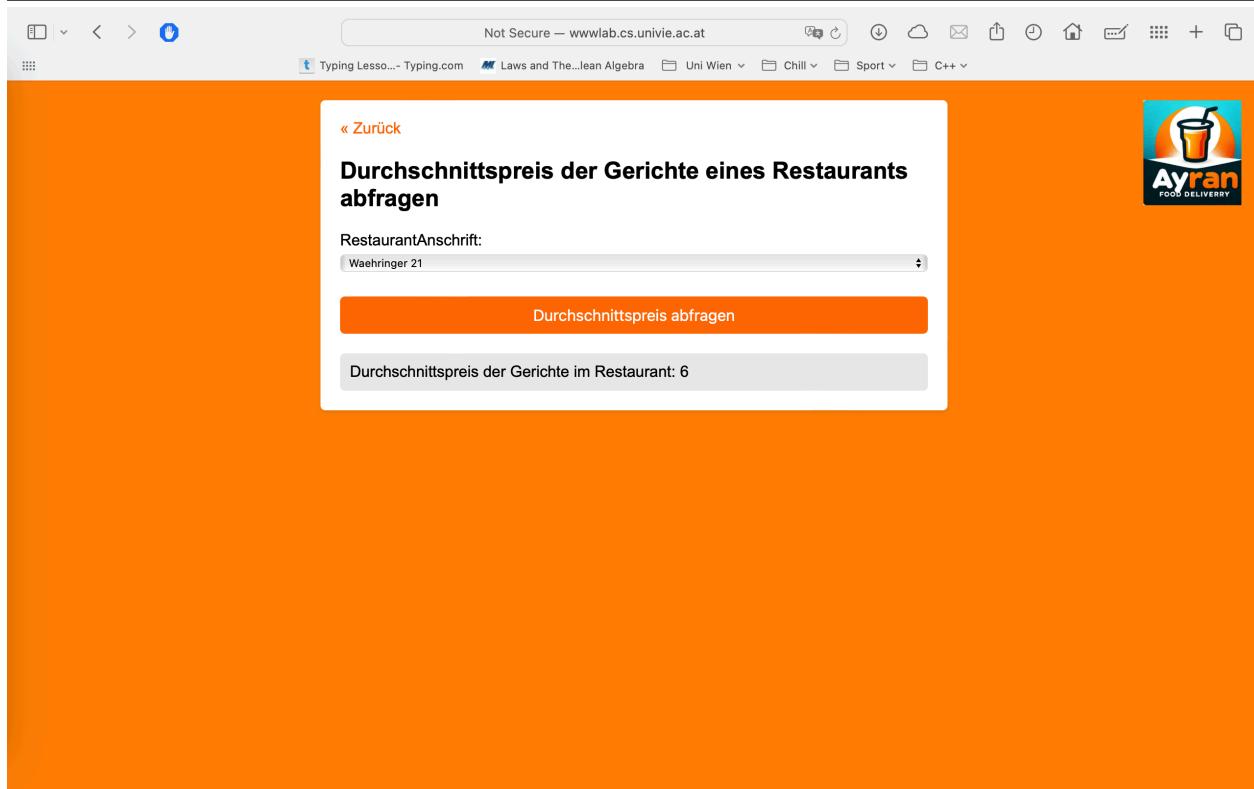
RestaurantAnschrift:

Waehinger 21

Durchschnittspreis abfragen

Durchschnittspreis der Gerichte im Restaurant: 6

Ayran FOOD DELIVERY



### ProcedureR.php