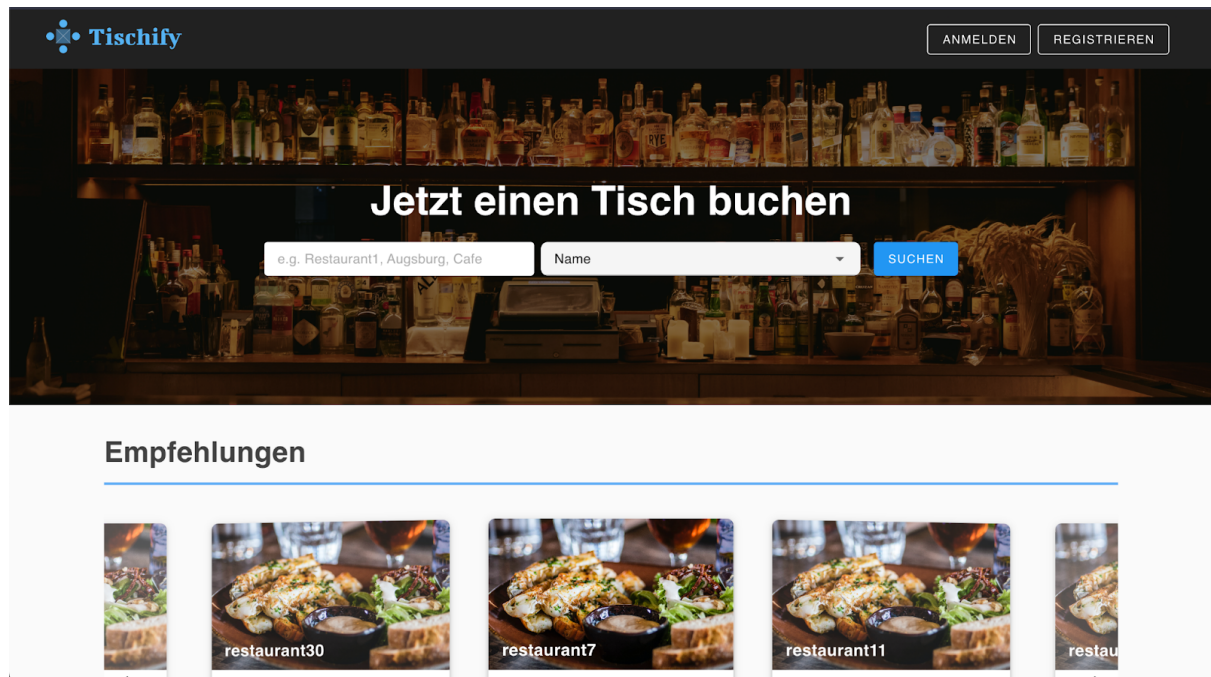


# Tischify



## Beschreibung des Projektes:

Tischify ist eine Web-app, mit der die Nutzer(Kunde) Plätze in Restaurant oder Café reservieren können. Nutzer(Mitarbeiter) können wiederum die Information des Restaurants bearbeiten.

## Verwendeten Technologie-Stack

- Vue3(Composition API(setup))
- Vuetify3
- pinia
- Express.js
- PostgreSQL

## Überblick der Verantwortlichkeit

Bei der Auflistung handelt es sich um das Hauptaufgabengebiet. Bei Bedarf hat jeder auch einen Teil des anderen übernommen.

- Wireframe: Taichi Tomioka
- Frontend: Taichi Tomioka
- Backend: Jannik Schmidt
- DB: Jannik Schmidt

## Anforderungen:

### ✓ 1. Registrierung:

Gäste können ein Konto erstellen und sich anmelden, um Tischreservierungen vorzunehmen.

### ✓ 2. Login:

Gäste und Restaurants können sich anmelden.

### ✓ 3. Restaurants-Suche:

Benutzer können nach Restaurants suchen, die in ihrer Nähe sind oder bestimmte Kriterien erfüllen, z. B. Name des Restaurants, Stadt, oder Kategorie.

### ✓ 4. Buchung:

Benutzer können einen Tisch für eine bestimmte Anzahl von Personen an einem bestimmten Datum und einer bestimmten Uhrzeit reservieren.

### ✓ 5. Reservierungsverwaltung:

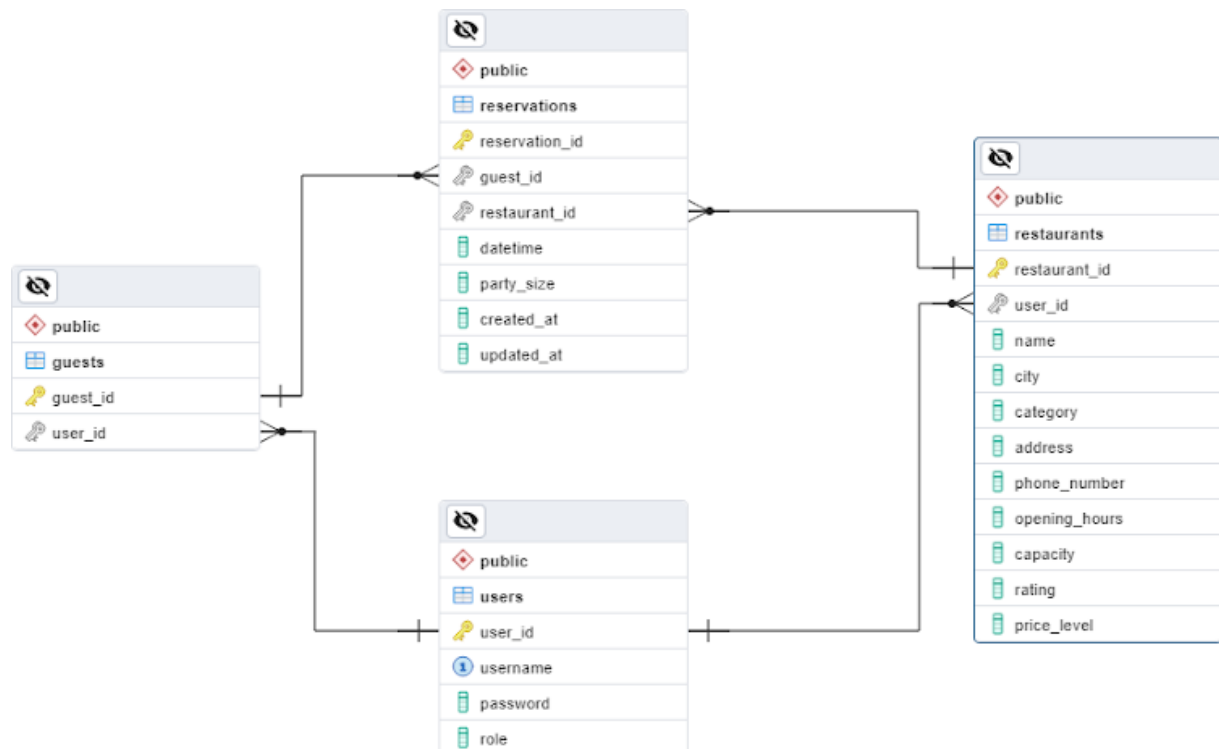
Benutzer können ihre Reservierungen anzeigen und stornieren.

### ✓ 6. Restaurantsinfo verwalten:

Restaurants können ihre Details verwalten, einschließlich Name, Adresse, Telefonnummer, Öffnungszeiten, und Kapazität.

Datenbank:

## Datenbankmodell



## SQL Funktionen:

### **login\_user ():**

Die Funktion wird verwendet, um das unverschlüsselte Passwort, welches vom Backend gesendet wird, mit dem verschlüsselten Passwort in der DB abzugleichen.

### **register\_user():**

Die Funktion wird verwendet, um einen neuen User, sowohl Guest als auch Restaurant in der DB anzulegen. Das Passwort wird hierbei verschlüsselt.

### **get\_reservations\_for\_guest():**

Die Funktion wird verwendet, um alle Reservierungen mit einer bestimmten Guest\_ID zu finden. Dabei müssen die Tabellen, Reservationen und Restaurants gejoint werden, um alle Informationen zu finden.

## REST-API

```
app.get("/users", db.getUsers);

app.get("/users/:id", db.getUserById);

app.get("/guests/bookings", authMiddleware, db.getGuestBookings);

app.delete("/guests/bookings/:booking_id", authMiddleware,
db.deleteBooking);

app.post("/bookings", authMiddleware, db.createBooking);

app.get("/restaurants/settings/", authMiddleware,
db.getRestaurantSettings);

app.get("/restaurants/availabilities", db.getRestaurantAvailabilities);

app.put("/restaurants/settings/", authMiddleware,
db.updateRestaurantSettings);

app.get("/restaurants/bookings", authMiddleware,
db.getRestaurantBookings);

app.get("/restaurants/:restaurant_id?", db.getRestaurants);

app.post("/login", db.login);

app.post("/register", db.register);
```

Beim Login wird ein Bearer Token generiert und zurückgegeben. Durch den Bearer Token ist es möglich, auch geschützte Endpunkte zuzugreifen. Im Token sind Informationen über die Rolle des Nutzers sowie Benutzername und ID gespeichert. Endpoints welche authMiddleware() verwenden sind nur erreichbar nachdem der Nutzer sich erfolgreich angemeldet hat.

## Seitenstruktur des Frontends (Figma)

[figma\\_Tischify](#)

## Einen Screencast der Komponenten der Anwendung in Aktion:

[Tischify-demo.mov](#)

## REST-Server mittels Thunder Client

Die Collection ist im Git unter Folgendem Pfad zu finden:  
/server/thunder-collection\_Tischify.json

### Erweiterbarkeit:

- **Reaktionsmöglichkeit auf Reservierungen:**  
Restaurants können die Verfügbarkeit für bestimmte Zeiträume definieren und Reservierungen bestätigen oder ablehnen.
- **Profilverwaltung für Guests:**  
Benutzer welche sich als Gast anmelden können zur Zeit keine Ihrer Informationen nachträglich anpassen
- **Erweiterung der Suchfunktion:**  
In der Suche ist es zur Zeit nur möglich, nach Name oder Ort zu suchen. Eine gute Erweiterung wäre, bei der Suche direkt auf Verfügbarkeit zu prüfen, heißt, dass Datum, Uhrzeit und Anzahl der Personen als Kriterium eingetragen werden können.
- **Hinterlegen von Bildern für Restaurants**