Weinkeller Backend Dokumentation

Bunea, Buchinger, Lagler, Bauer

IT-HTL Ybbs

ITP2

Inhalt

[Datenbank 2](#_Toc187221582)

[Auswahl 2](#_Toc187221583)

[Model 2](#_Toc187221584)

[Wines 2](#_Toc187221585)

[MostTreatments 3](#_Toc187221586)

[FermentationEntries 3](#_Toc187221587)

[Additives 3](#_Toc187221588)

[Users 4](#_Toc187221589)

[Seeder 4](#_Toc187221590)

[API 4](#_Toc187221591)

[UserController 4](#_Toc187221592)

[Endpoints 4](#_Toc187221593)

[WinesController 5](#_Toc187221594)

[Endpoints 5](#_Toc187221595)

[AdditivesController 5](#_Toc187221596)

[FermentationController 5](#_Toc187221597)

[Swagger 6](#_Toc187221598)

[Docker 6](#_Toc187221599)

[Abbildungsverzeichnis 8](#_Toc187221600)

# Datenbank

## Auswahl

Wir haben uns für eine PostgreSQL Datenbank entschieden. Als Datenbank Interface verwenden wir Adminer, eine einfache alternative zu PhpMyAdmin.

A screenshot of a computer

Description automatically generated

Abbildung 1: Adminer

## Model

Das Model wird mit dem EntityFramework erstellt.

### Wines

A screen shot of a computer program

Description automatically generatedDie Weine werden in der Wines Tabelle dargestellt. Jedem Wein ist ein User zugewiesen.

Abbildung 2: Wines

### MostTreatments

A computer screen shot of text

Description automatically generatedDie MostTreatments werden auch in einer eigenen Tabelle gespeichert. Falls Weine mehrere MostTreatments brauchen, kann dies leicht ergänzt werden.

Abbildung 3: MostTreatment

### FermentationEntries

A screen shot of a computer code

Description automatically generated

Abbildung 4: FermentationEntries

### Additives

A screen shot of a computer code

Description automatically generated

Abbildung 5: Additives

### Users

A screen shot of a computer program

Description automatically generatedDie User haben jeweils eine E-Mail, Username und ein Password. Das Passwort wird gehashed gespeichert.

Abbildung 6: User

## Seeder

Für Testzwecke wird die Datenbank mit Testdaten befüllt. Diese besteht aus zehn generierten Usern mit Weinen.

A screen shot of a computer program

Description automatically generatedDer Seeder wird im Program.cs aufgerufen.

Abbildung 7: Seeder

# API

Wir verwenden eine ASP.NET WebAPI mit vier Controller.

## UserController

Der UserController ist verantwortlich für Login und Register Endpoints.

### Endpoints

#### GetUser

User mit einer der gegebenen ID wird als DTO erhalten. Dieses DTO enthält nur Username und E-Mail.

#### DeleteUser

Löscht einen User.

#### Register

Erhält Userdaten und erstellt einen neuen User in der Datenbank. Das Password wird gehashed gespeichert.

#### Login

Der Login erfolgt mit E-Mail und Password. Das Password wird gehashed und mit dem gespeicherten Hash verglichen. Falls der Login Erfolgreich ist, wird ein JWT-Token erstellt und sendet diesen. Mit diesem JWT-Token werden alle Endpoints authentisiert.

## WinesController

Der WinesController ist für die Weine verantwortlich.

### Endpoints

Alle Endpoints müssen mit dem JWT authentisiert werden.

#### GetWines

Liefert alle Weine eines Users zurück.

#### Wine CRUD Endpoints

Standard CRUD Endpoints.

#### GetAdditives

Liefert alles Additive des Weines zurück.

#### GetFermentationEntries

Liefert alle FermentationEntries vom Wein zurück.

#### MostTreatment Crud Endpoints

Standard CRUD Endpoints für das MostTreatment vom Wein.

## AdditivesController

Der AdditivesController implementiert die Standard CRUD-Endpoints.

## FermentationController

Der FermentationController implementiert die Standard CRUD-Endpoints.

## Swagger

Die SwaggerUI ist über localhost:80/swagger erreichbar.A screenshot of a computer

Description automatically generated

Abbildung 8: SwaggerUI

# Docker

Für einfaches Teilen mit den Projektgruppen wird die API mit der Datenbank und Adminer in einem Docker Container gehostet.

A screenshot of a computer

Description automatically generated

Abbildung 9: Docker Container

Der Docker Container wird mit einer ReadMe Datei geteilt, die Anweisungen beinhaltet. CMD-Kommandos werden sind in .bat Dateien, um das Setup vom Docker Container zu vereinfachen.

A screenshot of a computer program

Description automatically generated

Abbildung 10: ReadMe

# Abbildungsverzeichnis

[Abbildung 1: Adminer 2](#_Toc187222103)

[Abbildung 2: Wines 2](#_Toc187222104)

[Abbildung 3: MostTreatment 3](#_Toc187222105)

[Abbildung 4: FermentationEntries 3](#_Toc187222106)

[Abbildung 5: Additives 3](#_Toc187222107)

[Abbildung 6: User 4](#_Toc187222108)

[Abbildung 7: Seeder 4](#_Toc187222109)

[Abbildung 8: SwaggerUI 6](#_Toc187222110)

[Abbildung 9: Docker Container 6](#_Toc187222111)

[Abbildung 10: ReadMe 7](#_Toc187222112)