

# Short introduction to the C# MAUI.NET / ASP.NET Sample Application

**Dr. Georg Hackenberg BSc MSc**

Professor for Industrial Informatics | School of Engineering

University of Applied Sciences Upper Austria

Stelzhamerstr. 23, 4600 Wels, Austria



<https://github.com/ghackenberg>



<https://linkedin.com/in/georghackenberg>



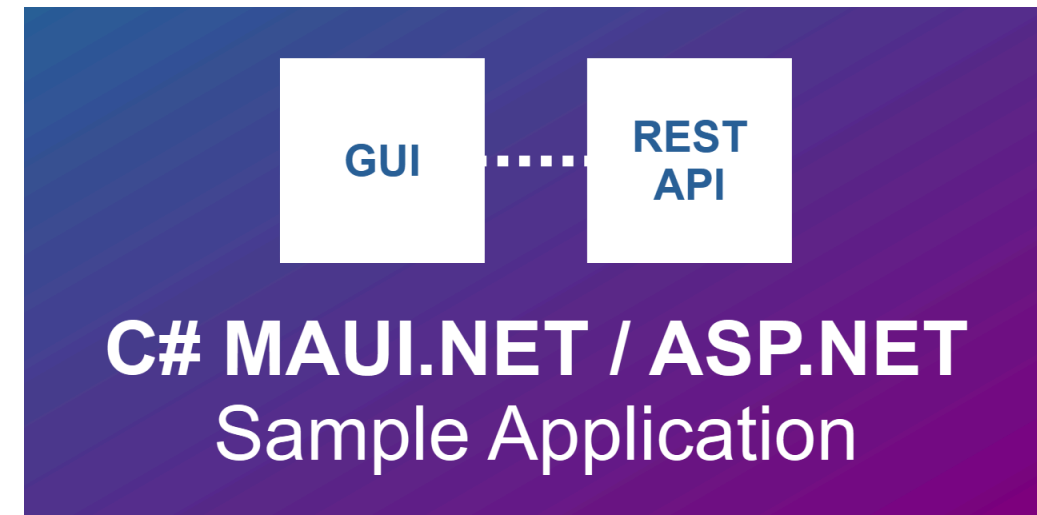
<https://youtube.com/@georghackenberg>

Also check out  <https://mentawise.com> and  <https://caddrive.org> 😎

## Content overview

This presentation provides a **starting point** for working with the **C# MAUI.NET / ASP.NET Sample Application** and is divided into two main sections:

- **Section 1 - External interfaces** explaining the graphical user interface (GUI) and the underlying REST API
- **Section 2 - Internal structures** explaining the structure of the source code in the background



## Section 1 - External interfaces

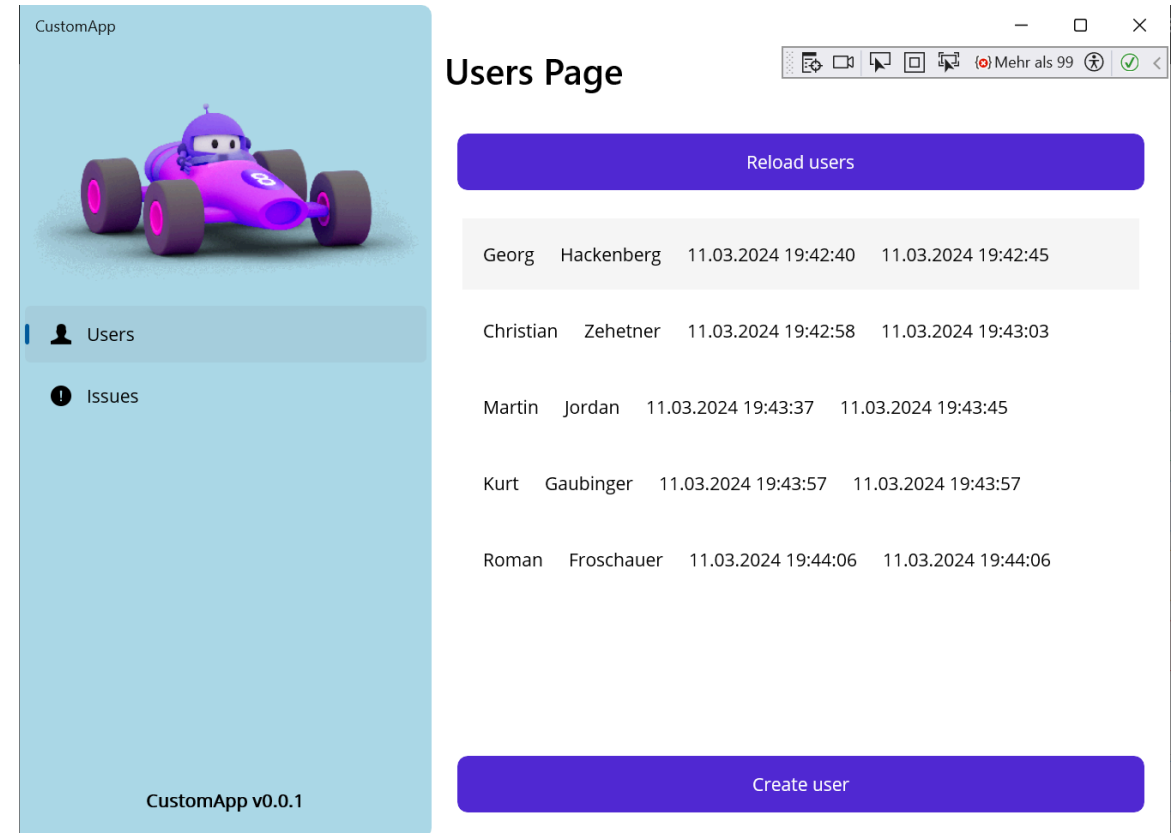
MAUI.NET frontend and ASP.NET backend

## MAUI.NET frontend

The C# MAUI.NET / ASP.NET Sample Application features a **basic** graphical user interface (GUI).

With the GUI you can manage the **users** and the **issues** stored in the underlying database.

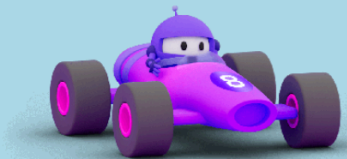
The screenshot on the right shows the **users page** listing all created user entities.



## MAUI.NET frontend (cont'd)

*Coming soon*

CustomApp



Users

Issues

CustomApp v0.0.1

User Page

User ID

cd1f1661-9fe2-453d-a0e5-be51fb89f729

Created At

11.03.2024 19:42:40

Updated At

11.03.2024 19:42:45

Deleted At

Not deleted yet

First name

Georg

Last name

Hackenberg

Save user

ASP.NET backend

Coming soon

CustomApi1.0OAS3

<http://localhost:5252/swagger/v1/swagger.json>

Comments

GET/api/comments

POST/api/comments

GET/api/comments/{id}

PUT/api/comments/{id}

DELETE/api/comments/{id}

Issues

GET/api/issues

POST/api/issues

GET/api/issues/{id}

PUT/api/issues/{id}

DELETE/api/issues/{id}

Users

GET/api/users

POST/api/users

GET/api/users/{id}

PUT/api/users/{id}

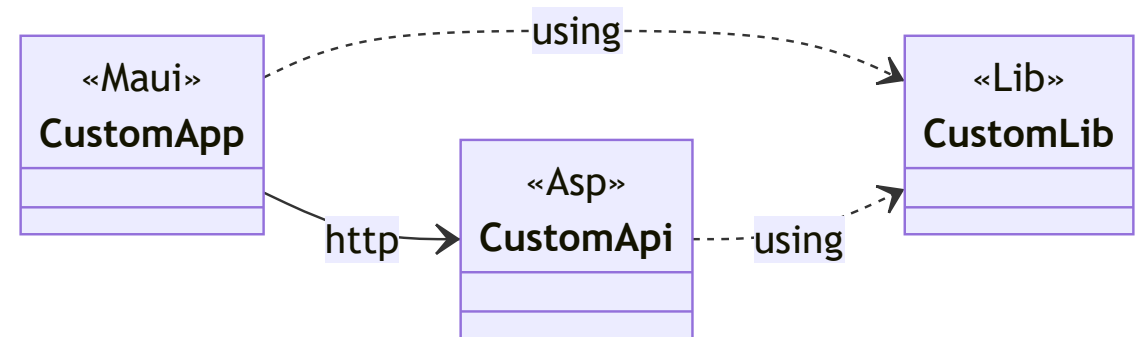
DELETE/api/users/{id}

## Section 2 - Internal structures

Package, data, message, interface, and page structure

## Package structure

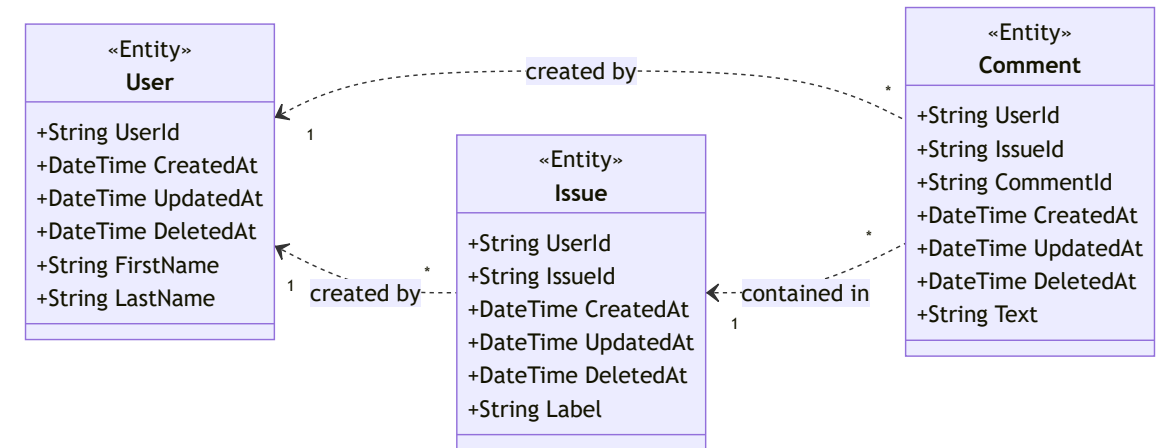
*Coming soon*





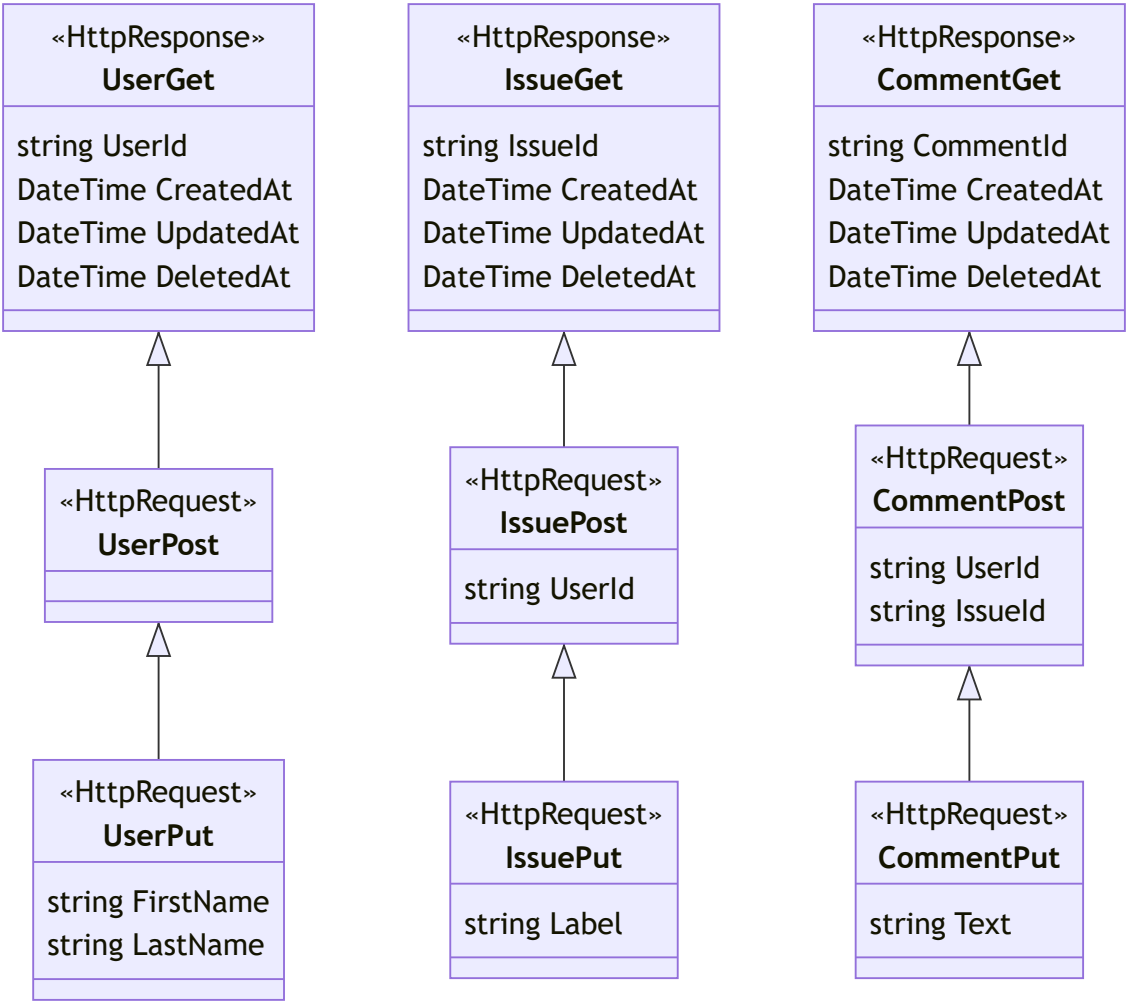
## Data structure

*Coming soon*



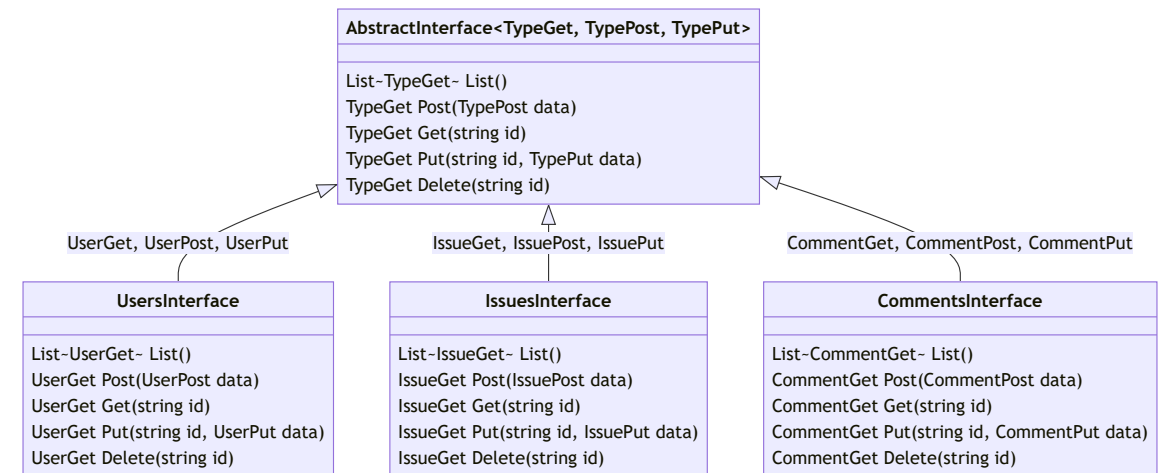
Message structure

Coming soon



## Interface structure

*Coming soon*



# Page structure

*Coming soon*

