



# GITHUB COPILOT & GITHUB COPILOT CHAT

TOP ANWENDUNGSFÄLLE  
AUS DER PRAXIS

# AUSSTELLER | SPONSOREN | PARTNER

## DWX 2025





## ÜBER MICH

MARC RUFER



SOFTWAREENTWICKLER



ISOLUTIONS AG



RUFER7



MARCRUFER





**DIFFERENT OPINIONS....**

**ARE OK**

makeameme.org



# AGENDA

EINLEITUNG

ÜBER GITHUB COPILOT

TOP ANWENDUNGSFÄLLE

LEARNINGS UND TIPPS

FAZIT



# ÜBER GITHUB COPILOT

# WAS IST GITHUB COPILOT

```
57 public function getStrings(){
58     return [
59         'en' => [
60             'answer' => 'Answer',
61             'question' => 'Question',
62             'date' => 'Date',
63         ],
64     ];
65 }
66
67
68 }
```

Was ist GitHub Copilot?

## GitHub Copilot

GitHub Copilot ist ein KI-gestützter Code-Assistent, der von GitHub in Zusammenarbeit mit OpenAI entwickelt wurde. Hier sind die wichtigsten Informationen:

### Was ist GitHub Copilot?

**GitHub Copilot** ist ein intelligenter Programmierassistent, der Entwicklern beim Schreiben von Code hilft, indem er:

- **Code-Vervollständigungen** in Echtzeit vorschlägt
- **Ganze Funktionen** basierend auf Kommentaren oder Funktionsnamen generiert
- **Code-Erklärungen** und Dokumentation bereitstellt
- **Fehler** identifiziert und Lösungsvorschläge macht
- **Tests** und **Refactoring-Vorschläge** erstellt

### Technologie

- Basiert auf **Large Language Models (LLMs)** von OpenAI
- Trainiert auf Milliarden von Zeilen öffentlichen Codes aus GitHub-Repositories
- Unterstützt **dutzende Programmiersprachen** wie Python, JavaScript, TypeScript, Java, C#, Go, Ruby und viele mehr

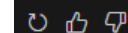
### Verfügbarkeit

- **GitHub Copilot Individual:** Für einzelne Entwickler
- **GitHub Copilot Business:** Für Unternehmen und Teams
- **GitHub Copilot Enterprise:** Erweiterte Funktionen für große Organisationen

### Integration

- Verfügbar als **VS Code Extension**
- Unterstützt auch andere IDEs wie **JetBrains**, **Neovim**, **Visual Studio**
- **Chat-Funktionen** für interaktive Unterstützung
- **CLI-Tools** für Terminal-basierte Entwicklung

GitHub Copilot revolutioniert die Art, wie Entwickler arbeiten, indem es die Produktivität steigert und beim Erlernen neuer Technologien hilft.



Add Context...

Edit files in your workspace in agent mode

Agent Claude Sonnet 4



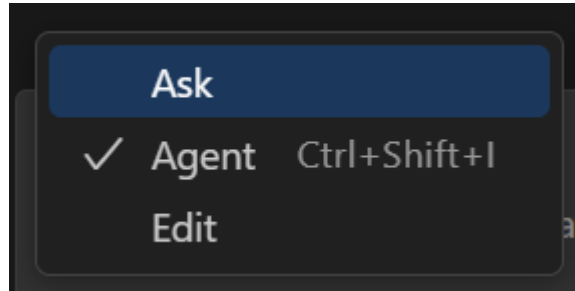
# FEATURES / FUNKTIONALITÄTEN



- **Code completion**
- **Copilot Chat**
- Copilot coding agent (public preview)
- **Copilot in the CLI**
- GitHub Copilot code review
- **Copilot pull request summaries**
- Copilot text completion (public preview)
- GitHub Copilot Extensions
- GitHub Models (public preview)
- **Copilot Edits**
- **Copilot custom instructions**
- Copilot in GitHub Desktop (public preview)
- **Copilot Spaces (public preview)**
- Copilot knowledge bases (Copilot Enterprise only)



# MODI

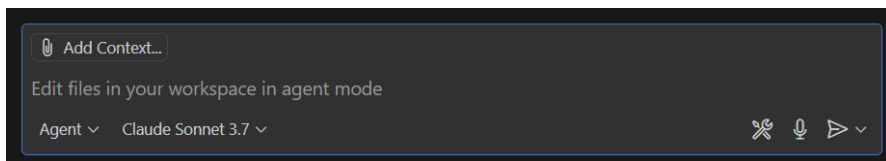


## Agent

- Komplexe Tasks (mehrere Schritte)
- Selbständige Ermittlung der notwendigen Schritte
- Integration externer Applikationen (MCP Server)

## Edit

- Kleinere, spezifische Änderungen an definierten Dateien
- Kontrolle über Anzahl LLM Anfragen



| Standard Models |                          |          |
|-----------------|--------------------------|----------|
|                 | GPT-4.1                  | Included |
|                 | GPT-4o                   | Included |
| Premium Models  |                          |          |
|                 | Claude Sonnet 3.5        | 1x       |
| ✓               | Claude Sonnet 3.7        | 1x       |
|                 | Claude Sonnet 4          | 1x       |
|                 | Gemini 2.5 Pro (Preview) | 1x       |
|                 | o4-mini (Preview)        | 0.33x    |

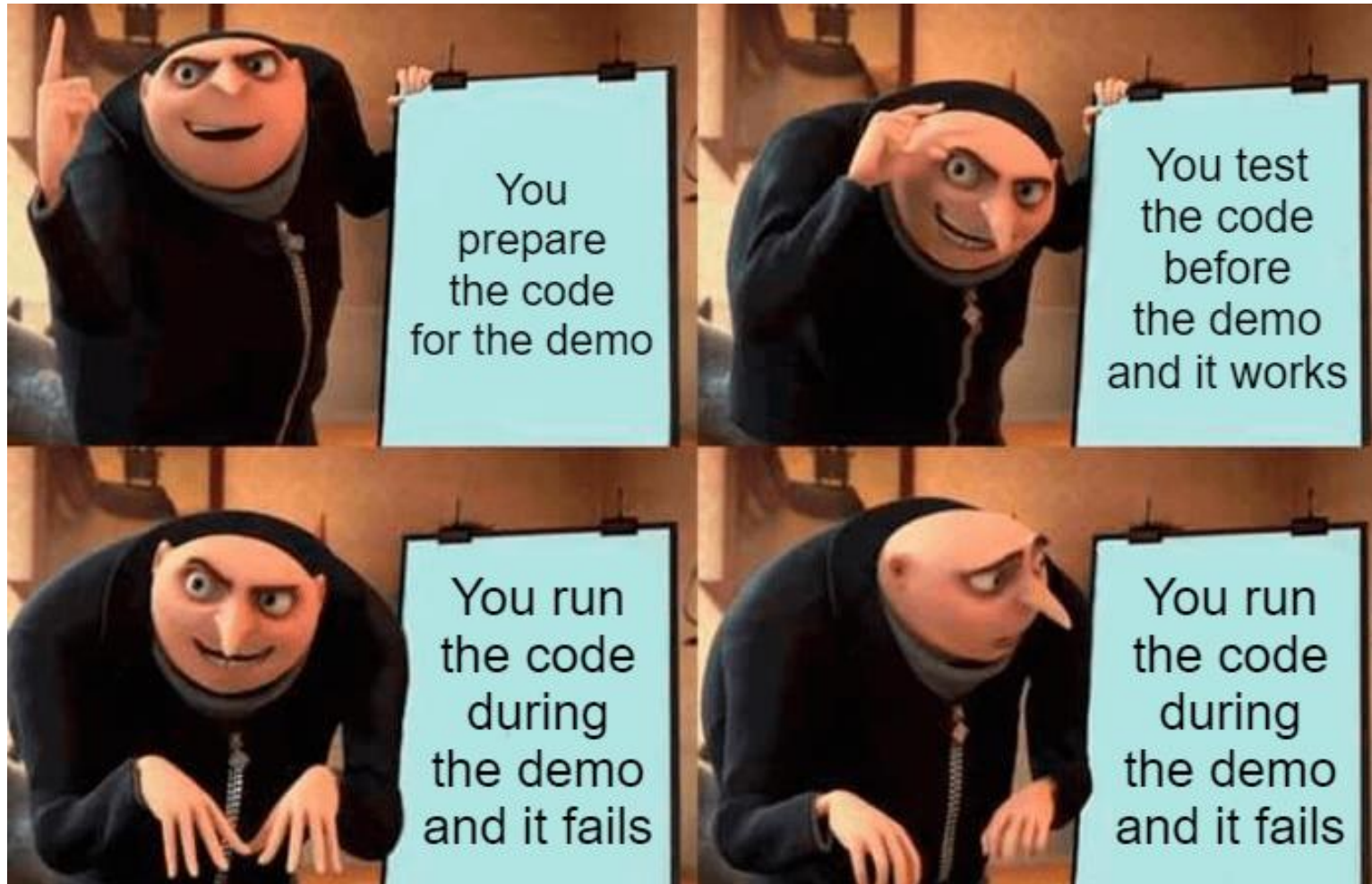
# MODELLE

| Aufgabe  | Geeignete Modelle  |
|--|--|
| Code schreiben / Allgemein   | GPT-4.1, GPT-4o, Claude Sonnet 3.7, Gemini 2.0 Flash, o4-mini                      |
| Einfache Aufgaben, repetitive Aufgaben                                     | o4-mini, o3-mini, Claude Sonnet 3.5, Gemini 2.0 Flash                              |
| Komplexe Aufgaben, Debugging   | GPT-4.5, o3, o1, Claude Sonnet 3.7, Claude Sonnet 4, Claude Opus 4, Gemini 2.5 Pro |
| Aufgaben mit visuellem Input (Screenshots, Diagramme, UI Komponenten, ...) | GPT-4o, Gemini 2.0 Flash   |



# TOP ANWENDUNGSFÄLLE

## DEMO TIME



CHAT

< > + ↺ ⚙️ ⋮ 📄 ✕

> Searched codebase for "#codebase can you please explain in short what the application in this project does?", 26 results

I'll analyze the codebase to explain what this application does.

✓ Searched text for README.md (\*\*/README.md), no results

Let me search for more information about the project structure and its main components:

> Searched codebase for "project description purpose overview main functionality", 12 results

Based on my analysis of the codebase, I can provide a clear overview of what this application does:

ASP.NET Core + React BFF Proxy Example

This is a sample application demonstrating a Backend-For-Frontend (BFF) pattern with ASP.NET Core serving as the backend and React as the frontend. The main purpose of this application is to showcase a secure architecture where:

- 1. Unified Hosting Architecture:** The ASP.NET Core server hosts the React client application in production, while during development, it proxies requests to the Vite development server.
- 2. Security-Focused Design:** The application implements various security features:
  - Azure AD authentication with cookie-based auth
  - Content Security Policy (CSP) with nonce support
  - CSRF/XSRF protection using the synchronizer token pattern
  - Secure headers management
- 3. API Integration:** The backend exposes several APIs:
  - Weather forecast data (for demonstration purposes)
  - Drinks data with role-based authorization
  - Microsoft Graph API integration for user data
- 4. Progressive Web App (PWA):** The application includes PWA support with service worker implementation for offline capabilities.
- 5. Role-Based Access Control:** The app implements role-based security with defined roles like:
  - `Drinks.Read` for read access to drinks
  - `Drinks.Write` for write access to drinks
  - `Admin` for administrative access

The frontend is built with React, using Material-UI for the UI components, React Router for navigation, and React Query for data fetching. The backend is an ASP.NET Core application with controller-based APIs and authentication/authorization services.

This sample project serves as a reference implementation for developers who want to build secure, modern web applications with React and ASP.NET Core using the BFF pattern.

↺ 👍 🗨

📎 Add Context...

Ask Copilot

Ask Claude Sonnet 3.7

@ 🎤 ➤

# #1 – CODE ERKLÄREN

- GitHub Copilot Chat
- Modus: Ask
- Gesamte Code Basis: #codebase
- File / Klasse: im aktuellen Tab öffnen
- Methode / Funktion: Code selektieren





CHAT

structure: Status, Context, Decision, Consequences  
Consistent method for storing datetimes  
Store in UTC  
Datetimes need to be converted to local time for display

PROMPTS.md:14-19

> Used 1 reference

I'll create a short and crisp architecture decision record based on the provided keywords.

### Architecture Decision Record: Datetime Storage and Display

#### Status

Accepted

#### Context

We need a consistent approach for storing and displaying datetime values in our application. Different users across various time zones need to see correct and consistent datetime information. Inconsistent datetime handling can lead to data integrity issues, confusion, and potential bugs.

#### Decision

We will store all datetime values in UTC (Coordinated Universal Time) format in our database and internal systems. When displaying datetime information to users, we will convert UTC values to the user's local time zone.

#### Consequences

Positive

- Consistent data storage regardless of where the application is deployed or accessed from
- Eliminates ambiguity with daylight saving time transitions
- Simplifies data aggregation, reporting, and cross-timezone operations
- Enables accurate time-based calculations
- Users see dates and times in their familiar local format

Negative

- Requires conversion logic in the presentation layer
- Developers need to be mindful of the UTC convention in all datetime operations
- May require storing or determining user timezone preferences
- Additional testing needed across time zone boundaries

🔄 👍 🗒

📎 Add Context... ⬇️ PROMPTS.md Current file 🔗

Ask Copilot

Ask Claude Sonnet 3.7

# #2 – DOKUMENTATION SCHREIBEN

- GitHub Copilot Chat
- Modus: Ask oder Edit
- Stichworte angeben und markieren
- Alternativ mit GitHub Copilot Code-Vervollständigung



```

// Act
var drinks = drinksService.GetAll();

// Assert
Assert.Equal(5, drinks.Count);
}

// Create test to get drink by id
[Fact]
0 references
public void GetById_ReturnsDrink_WhenIdExists()
{
    // Arrange
    var drinksService = new DrinksService();
    var expectedDrink = drinksService.GetAll().First();

    // Act
    var drink = drinksService.GetById(expectedDrink.Id);

    // Assert
    Assert.NotNull(drink);
    Assert.Equal(expectedDrink.Id, drink.Id);
}

// Create test to get drink by id that does not exist
[Fact]
0 references
public void GetById_ReturnsNull
{
    // Arrange
    0 references
    var drinksService = new DrinksService();

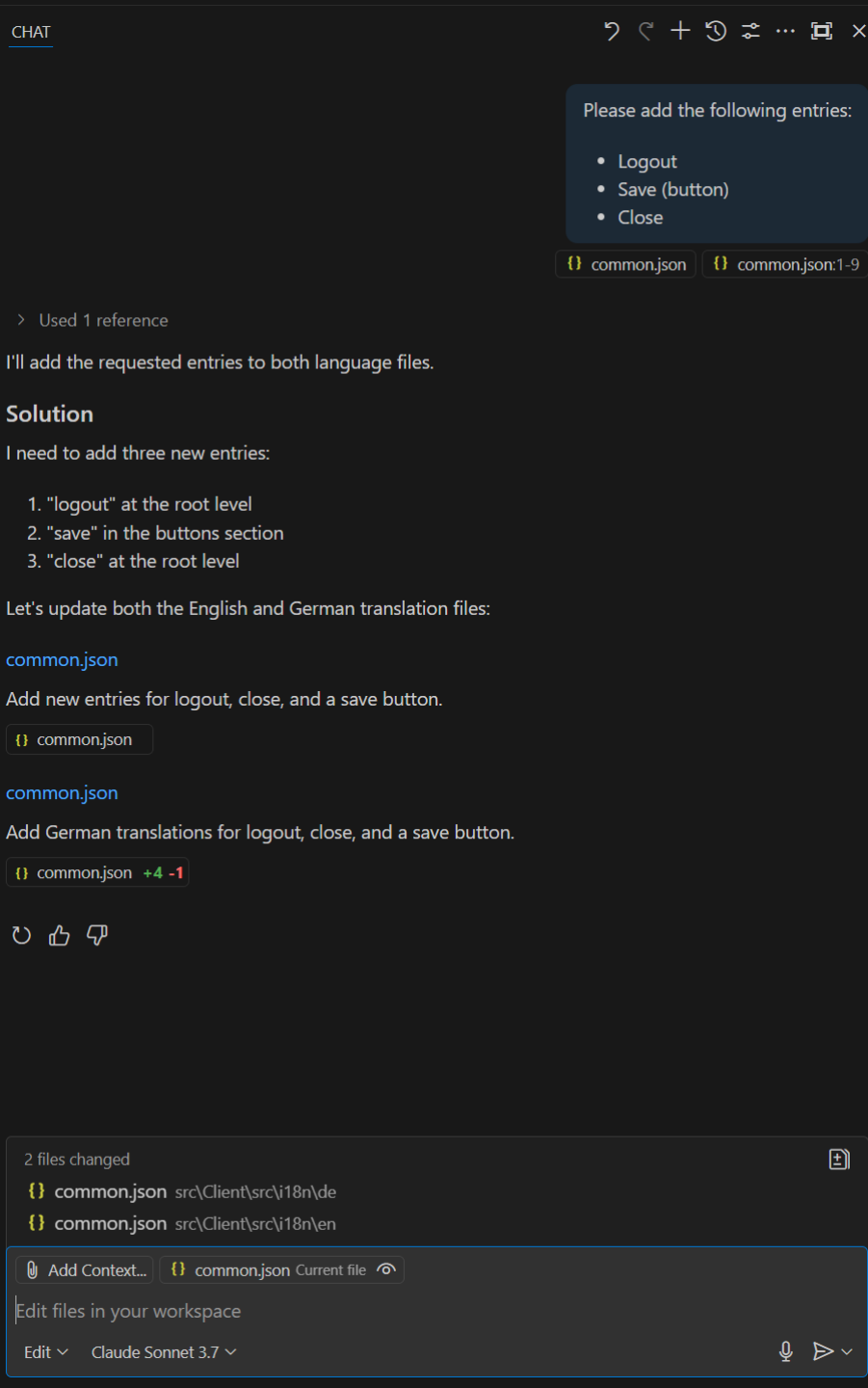
    // Act
    0 references
    var drink = drinksService.GetById(Guid.NewGuid());

    // Assert
    Assert.Null(drink);
}

```

## #3 – UNIT TESTS GENERIEREN

- GitHub Copilot (Code-Vervollständigung)
- Initiales Set an Tests manuell implementieren
- Alternativ mit GitHub Copilot Chat (Modus: Edit oder Agent)



## #4 – ÜBERSETZEN (I18N)

- GitHub Copilot Chat
- Modus: Edit oder Agent
- Im Edit Modus zu ändernde Dateien dem Kontext hinzufügen

CHAT

↶ ↷ + ↺ ⌘ ... 🖨 ✕

Is there anything that could cause performance issues in this class?

≡ \_Host.cshtml:15

> Used 1 reference

I'll analyze the `_Host.cshtml` file to identify potential performance issues, with a focus on the selected line that creates an `HttpClient`.

✓ Searched text for `using HttpClient`, no results

Let me look at the `_Host.cshtml` file in more detail, specifically focusing on the performance concerns.

The main performance issue I can identify is related to the line you've highlighted:

```
var httpClient = new HttpClient();
```

This creates a new instance of `HttpClient` for each request to the page. This is a common anti-pattern known as "HttpClient instance per request" that can lead to several performance problems:

- 1. Socket Exhaustion:** Creating and disposing of `HttpClient` instances frequently can lead to socket exhaustion, as the underlying TCP connections might remain in the `TIME_WAIT` state even after the `HttpClient` is disposed. This can eventually lead to running out of available ports.
- 2. DNS Caching Issues:** Each new `HttpClient` instance doesn't benefit from DNS caching of previous requests, which can slow down requests.
- 3. Performance Overhead:** Creating a new instance for each request adds unnecessary overhead.
- 4. No Connection Pooling Benefits:** You lose the benefits of connection pooling when creating new instances each time.

Recommended Solution:

The best practice is to use a single, long-lived `HttpClient` instance or use `IHttpClientFactory` which was introduced to solve these exact issues:

- 1. Using HttpClientFactory (Recommended):**
  - Register `HttpClient` in your service collection (in `Program.cs` or a service extension)
  - Inject `IHttpClientFactory` into your page and get a client from it
- 2. Static HttpClient (Alternative):**
  - Create a static `HttpClient` instance that's reused across requests

Here's how you could improve the code:

📎 Add Context... ≡ \_Host.cshtml:15 Current file 🔍

Ask Copilot

Ask ▾ Claude Sonnet 3.7 ▾ @ 🎤 ➤ ▾

# #5 – PERFORMANCE-PROBLEME IDENTIFIZIEREN

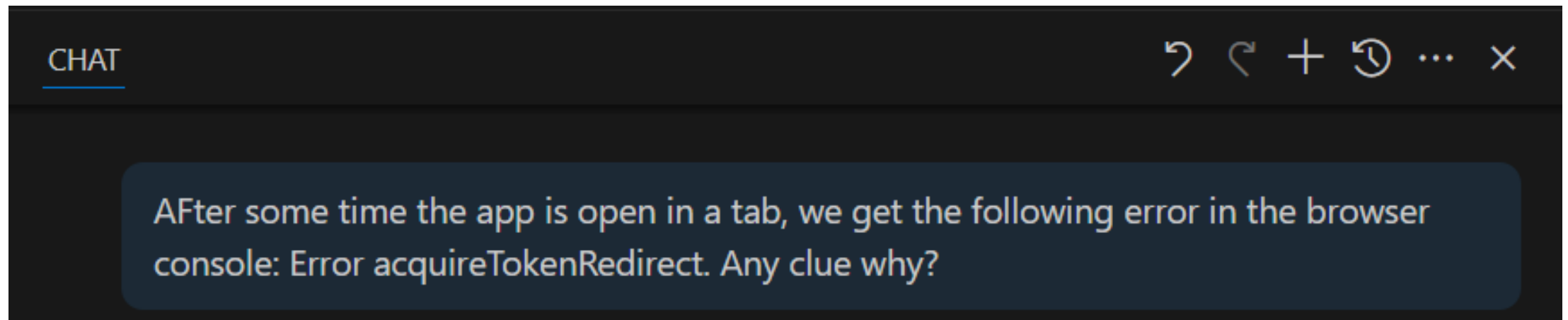
- GitHub Copilot Chat
- Modus: Ask
- Alternativ mit GitHub Copilot Chat (Modus: Edit oder Agent)



# #6 – FEHLERQUELLE(N) IDENTIFIZIEREN

GitHub Copilot Chat

Modus: Ask oder Agent





## GitHub Copilot

The error "Error acquireTokenRedirect" typically occurs in applications that use Microsoft Authentication Library (MSAL) for Azure AD B2C authentication. This is often related to token acquisition problems when the application tries to refresh or acquire a new authentication token.

Let's explore some potential causes and solutions for this issue:

1. **Token expiration** - After the tab has been open for some time, the authentication token has likely expired and the app is failing to refresh it.
2. **Silent token refresh issues** - The app might be trying to silently refresh the token in the background but failing.
3. **Authentication configuration problems** - There might be issues with the authentication configuration in your application.

Let me check if I can find more specific information in your codebase:

> Searched text for `acquireTokenRedirect`, 8 results

Now let's look at the authentication configuration:

✓ Read `Root.tsx`, lines 1 to 101

✓ Read `auth.ts`, lines 1 to 49

✓ Read `CustomFetch.ts`, lines 1 to 81

Based on the code I've examined, here are the likely causes and solutions for the "Error acquireTokenRedirect" issue:

## Causes:

1. **Token Expiration:** The authentication token is expiring after the tab has been open for some time, and the automatic refresh is failing.
2. **Silent Token Acquisition Failure:** The `acquireTokenSilent` call is failing, and then the fallback `acquireTokenRedirect` is also failing.
3. **Authentication State in New Tabs:** There's a comment in the code about issues with new tabs: "To avoid the 'Something went wrong' error message on new tabs we implemented the following workaround."
4. **Error Handling:** In `CustomFetch.ts`, when `acquireTokenSilent` fails, the code attempts `acquireTokenRedirect` but doesn't properly handle the Promise chain, potentially causing errors.

## Solutions:

1. **Fix Error Handling in `CustomFetch.ts`:** The error handling in the `customFetch` function doesn't properly return a Promise after calling `acquireTokenRedirect`, which could lead to unhandled Promise rejections.

`CustomFetch.ts` +9 -7

2. **Fix Error Handling in `getToken` Function:**

`CustomFetch.ts` +9 -7

3. **Implement Token Refresh Logic in `Root.tsx`:** Add a token refresh interval to periodically check and refresh the token before it expires:

`Root.tsx` +20 -2

4. **Improve `Auth.ts` Error Handling:**

`auth.ts` +8 -5

# #7 – GITHUB ACTIONS FEHLER BEHEBEN

GitHub Copilot Chat @GitHub.com

The screenshot displays the GitHub Actions interface for a repository named '.NET gitleaks'. The top navigation bar includes links for Code, Pull requests, Actions (selected), Projects, Wiki, Security (3 alerts), Insights, and Settings. The main header shows the workflow 'Update README.md #13' with a failed status icon and a 'Re-run jobs' button. The left sidebar contains a 'Summary' section and a 'Jobs' list with 'build' (successful) and 'gitleaks scan' (failed). The 'Run details' section on the left lists 'Usage' and 'Workflow file'. The main content area shows the 'gitleaks scan' job details, indicating it failed 2 weeks ago in 5s. It includes a 'Search logs' input, an 'Explain error' button, and a list of steps: 'Set up job' (1s), 'Run actions/checkout@v4' (1s), and 'gitleaks-action' (1s) which is currently expanded and shows a failure icon.

<> Code   Pull requests   **Actions**   Projects   Wiki   Security 3   Insights   Settings

← .NET gitleaks

**Update README.md #13** Re-run jobs ⋮

Summary

Jobs

- build
- gitleaks scan**

Run details

- Usage
- Workflow file

**Annotations**  
2 warnings

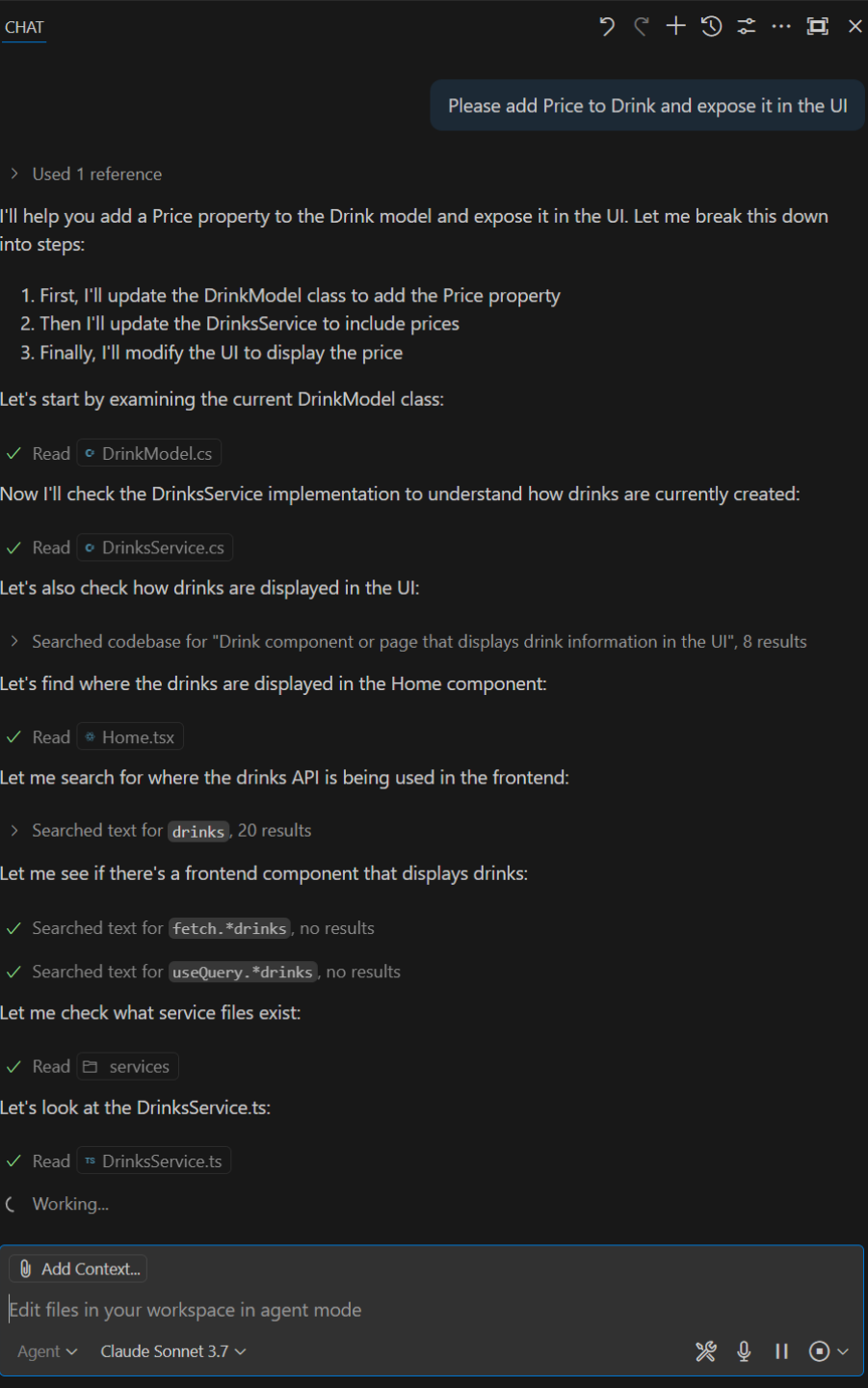
**gitleaks scan**  
failed 2 weeks ago in 5s Search logs Explain error ↺ ⚙️

- > Set up job 1s
- > Run actions/checkout@v4 1s
- ▼ **gitleaks-action** 1s

AuthProvider.tsx 3    ErrorPage.tsx 9+    Layout.tsx 9+

src > Client > src > Layout.tsx > ...

4    IconButton,  
5    Typography,  
6    Drawer,  
7    List,  
8    ListItem,  
9    styled,  
10   Container,  
11   ListItemButton,  
12   ListItemText,  
13   ListItemIcon,  
14   Box,  
15   } from "@mui/material";  
16   import MenuIcon from "@mui/icons-material/Menu";  
17   import { Outlet, Link } from "react-router-dom";  
18   import HomeIcon from "@mui/icons-material/Home";  
19   import SettingsIcon from "@mui/icons-material/Settings";  
20   import LoginLogoutArea from "../LoginLogoutArea";  
21   import AuthGuard from "../auth/AuthGuard";  
  
22     
23   const Offset = styled("div")(({ theme }) => theme.spacing(2));  
24     
25   export const Layout = () => (  
26     <>  
27     <AppBar position="fixed">  
28     <Toolbar  
29       sx={{  
30         marginLeft: "300px",  
31       }}  
32     >  
33     <IconButton  
34       edge="start"  
35       color="inherit"  
36     >  
37     <MenuIcon />  
38     <Link to="/" />  
39     <Link to="/home" />  
40     <Link to="/settings" />  
41     <Link to="/login" />  
42     <Link to="/logout" />  
43     <AuthGuard />  
44     <LoginLogoutArea />  
45     <Outlet />  
46     </Toolbar>  
47   </AppBar>  
48   <div style={Offset}>  
49   <Outlet />  
50   </div>  
51   </>  
52   );  
53     
54     
55     
56     
57     
58     
59     
60     
61     
62     
63     
64     
65     
66     
67     
68     
69     
70     
71     
72     
73     
74     
75     
76     
77     
78     
79     
80     
81     
82     
83     
84     
85     
86     
87     
88     
89     
90     
91     
92     
93     
94     
95     
96     
97     
98     
99     
100     
101     
102     
103     
104     
105     
106     
107     
108     
109     
110     
111     
112     
113     
114     
115     
116     
117     
118     
119     
120     
121     
122     
123     
124     
125     
126     
127     
128     
129     
130     
131     
132     
133     
134     
135     
136     
137     
138     
139     
140     
141     
142     
143     
144     
145     
146     
147     
148     
149     
150     
151     
152     
153     
154     
155     
156     
157     
158     
159     
160     
161     
162     
163     
164     
165     
166     
167     
168     
169     
170     
171     
172     
173     
174     
175     
176     
177     
178     
179     
180     
181     
182     
183     
184     
185     
186     
187     
188     
189     
190     
191     
192     
193     
194     
195     
196     
197     
198     
199     
200     
201     
202     
203     
204     
205     
206     
207     
208     
209     
210     
211     
212     
213     
214     
215     
216     
217     
218     
219     
220     
221     
222     
223     
224     
225     
226     
227     
228     
229     
230     
231     
232     
233     
234     
235     
236     
237     
238     
239     
240     
241     
242     
243     
244     
245     
246     
247     
248     
249     
250     
251     
252     
253     
254     
255     
256     
257     
258     
259     
260     
261     
262     
263     
264     
265     
266     
267     
268     
269     
270     
271     
272     
273     
274     
275     
276     
277     
278     
279     
280     
281     
282     
283     
284     
285     
286     
287     
288     
289     
290     
291     
292     
293     
294     
295     
296     
297     
298     
299     
300     
301     
302     
303     
304     
305     
306     
307     
308     
309     
310     
311     
312     
313     
314     
315     
316     
317     
318     
319     
320     
321     
322     
323     
324     
325     
326     
327     
328     
329     
330     
331     
332     
333     
334     
335     
336     
337     
338     
339     
340     
341     
342     
343     
344     
345     
346     
347     
348     
349     
350     
351     
352     
353     
354     
355     
356     
357     
358     
359     
360     
361     
362     
363     
364     
365     
366     
367     
368     
369     
370     
371     
372     
373     
374     
375     
376     
377     
378     
379     
380     
381     
382     
383     
384     
385     
386     
387     
388     
389     
390     
391     
392     
393     
394     
395     
396     
397     
398     
399     
400     
401     
402     
403     
404     
405     
406     
407     
408     
409     
410     
411     
412     
413     
414     
415     
416     
417     
418     
419     
420     
421     
422     
423     
424     
425     
426     
427     
428     
429     
430     
431     
432     
433     
434     
435     
436     
437     
438     
439     
440     
441     
442     
443     
444     
445     
446     
447     
448     
449     
450     
451     
452     
453     
454     
455     
456     
457     
458     
459     
460     
461     
462     
463     
464     
465     
466     
467     
468     
469     
470     
471     
472     
473     
474     
475     
476     
477     
478     
479     
480     
481     
482     
483     
484     
485     
486     
487     
488     
489     
490     
491     
492     
493     
494     
495     
496     
497     
498     
499     
500     
501     
502     
503     
504     
505     
506     
507     
508     
509     
510     
511     
512     
513     
514     
515     
516     
517     
518     
519     
520     
521     
522     
523     
524     
525     
526     
527     
528     
529     
530     
531     
532     
533     
534     
535     
536     
537     
538     
539     
540     
541     
542     
543     
544     
545     
546     
547     
548     
549     
550     
551     
552     
553     
554     
555     
556     
557     
558     
559     
560     
561     
562     
563     
564     
565     
566     
567     
568     
569     
570     
571     
572     
573     
574     
575     
576     
577     
578     
579     
580     
581     
582     
583     
584     
585     
586     
587     
588     
589     
590     
591     
592     
593     
594     
595     
596     
597     
598     
599     
600     
601     
602     
603     
604     
605     
606     
607     
608     
609     
610     
611     
612     
613     
614     
615     
616     
617     
618     
619     
620     
621     
622     
623     
624     
625     
626     
627     
628     
629     
630     
631     
632     
633     
634     
635     
636     
637     
638     
639     
640     
641     
642     
643     
644     
645     
646     
647     
648     
649     
650     
651     
652     
653     
654     
655     
656     
657     
658     
659     
660     
661     
662     
663     
664     
665     
666     
667     
668     
669     
670     
671     
672     
673     
674     
675     
676     
677     
678     
679     
680     
681     
682     
683     
684     
685     
686     
687     
688     
689     
690     
691     
692     
693     
694     
695     
696     
697     
698     
699     
700     
701     
702     
703     
704     
705     
706     
707     
708     
709     
710     
711     
712     
713     
714     
715     
716     
717     
718     
719     
720     
721     
722     
723     
724     
725     
726     
727     
728     
729     
730     
731     
732     
733     
734     
735     
736     
737     
738     
739     
740     
741     
742     
743     
744     
745     
746     
747     
748     
749     
750     
751     
752     
753     
754     
755     
756     
757     
758     
759     
760     
761     
762     
763     
764     
765     
766     
767     
768     
769     
770     
771     
772     
773     
774     
775     
776     
777     
778     
779     
780     
781     
782     
783     
784     
785     
786     
787     
788     
789     
790     
791     
792     
793     
794     
795     
796     
797     
798     
799     
800     
801     
802     
803     
804     
805     
806     
807     
808     
809     
810     
811     
812     
813     
814     
815     
816     
817     
818     
819     
820     
821     
822     
823     
824     
825     
826     
827     
828     
829     
830     
831     
832     
833     
834     
835     
836     
837     
838     
839     
840     
841     
842     
843     
844     
845     
846     
847     
848     
849     
850     
851     
852     
853     
854     
855     
856     
857     
858     
859     
860     
861     
862     
863     
864     
865     
866     
867     
868     
869     
870     
871     
872     
873     
874     
875     
876     
877     
878     
879     
880     
881     
882     
883     
884     
885     
886     
887     
888     
889     
890     
891     
892     
893     
894     
895     
896     
897     
898     
899     
900     
901     
902     
903     
904     
905     
906     
907     
908     
909     
910     
911     
912     
913     
914     
915     
916     
917     
918     
919     
920     
921     
922     
923     
924     
925     
926     
927     
928     
929     
930     
931     
932     
933     
934     
935     
936     
937     
938     
939     
940     
941     
942     
943     
944     
945     
946     
947     
948     
949     
950     
951     
952     
953     
954     
955     
956     
957     
958     
959     
960     
961     
962     
963     
964     
965     
966     
967     
968     
969     
970     
971     
972     
973     
974     
975     
976     
977     
978     
979     
980     
981     
982     
983     
984     
985     
986     
987     
988     
989     
990     
991     
992     
993     
994     
995     
996     
997     
998     
999     
1000     
1001     
1002     
1003     
1004     
1005     
1006     
1007     
1008     
1009     
1010     
1011     
1012     
1013     
1014     
1015     
1016     
1017     
1018     
1019     
1020     
1021     
1022     
1023     
1024     
1025     
1026     
1027     
1028     
1029     
1030     
1031     
1032     
1033     
1034     
1035     
1036     
1037     
1038     
1039     
1040     
1041     
1042     
1043     
1044     
1045     
1046     
1047     
1048     
1049     
1050     
1051     
1052     
1053     
1054     
1055     
1056     
1057     
1058     
1059     
1060     
1061     
1062     
1063     
1064     
1065     
1066     
1067     
1068     
1069     
1070     
1071     
1072     
1073     
1074     
1075     
1076     
1077     
1078     
1079     
1080     
1081     
1082     
1083     
1084     
1085     
1086     
1087     
1088     
1089     
1090     
1091     
1092     
1093     
1094     
1095     
1096     
1097     
1098     
1099     
1100     
1101     
1102     
1103     
1104     
1105     
1106     
1107     
1108     
1109     
1110     
1111     
1112     
1113     
1114     
1115     
1116     
1117     
1118     
1119     
1120     
1121     
1122     
1123     
1124     
1125     
1126     
1127     
1128     
1129     
1130     
1131     
1132     
1133     
1134     
1135     
1136     
1137     
1138     
1139     
1140     
1141     
1142     
1143     
1144     
1145     
1146     
1147     
1148     
1149     
1150     
1151     
1152     
1153     
1154     
1155     
1156     
1157     
1158     
1159     
1160     
1161     
1162     
1163     
1164     
1165     
1166     
1167     
1168     
1169     
1170     
1171     
1172     
1173     
1174     
1175     
1176     
1177     
1178     
1179     
1180     
1181     
1182     
1183     
1184     
1185     
1186     
1187     
1188     
1189     
1190     
1191     
1192     
1193     
1194     
1195     
1196     
1197     
1198     
1199     
1200     
1201     
1202     
1203     
1204     
1205     
1206     
1207     
1208     
1209     
1210     
1211     
1212     
1213     
1214     
1215     
1216     
1217     
1218     
1219     
1220     
1221     
1222     
1223     
1224     
1225     
1226     
1227     
1228     
1229     
1230     
1231     
1232     
1233     
1234     
1235     
1236     
1237     
1238     
1239     
1240     
1241     
1242     
1243     
1244     
1245     
1246     
1247     
1248     
1249     
1250     
1251     
1252     
1253     
1254     
1255     
1256     
1257     
1258     
1259     
1260     
1261     
1262     
1263     
1264     
1265     
1266     
1267     
1268     
1269     
1270     
1271     
1272     
1273     
1274     
1275     
1276     
1277     
1278     
1279     
1280     
1281     
1282     
1283     
1284     
1285     
1286     
1287     
1288     
1289     
1290     
1291     
1292     
1293     
1294     
1295     
1296     
1297     
1298     
1299     
1300     
1301     
1302     
1303     
1304     
1305     
1306     
1307     
1308     
1309     
1310     
1311     
1312     
1313     
1314     
1315     
1316     
1317     
1318     
1319     
1320     
1321     
1322     
1323     
1324     
1325     
1326     
1327     
1328     
1329     
1330     
1331     
1332     
1333     
1334     
1335     
1336     
1337     
1338     
1339     
1340     
1341     
1342     
1343     
1344     
1345     
1346     
1347     
1348     
1349     
1350     
1351     
1352     
1353     
1354     
1355     
1356     
1357     
1358     
1359     
1360     
1361     
1362     
1363     
1364     
1365     
1366     
1367     
1368     
1369     
1370     
1371     
1372     
1373     
1374     
1375     
1376     
1377     
1378     
1379     
1380     
1381     
1382     
1383     
1384     
1385     
1386     
1387     
1388     
1389     
1390     
1391     
1392     
1393     
1394     
1395     
1396     
1397     
1398     
1399     
1400     
1401     
1402     
1403     
1404     
1405     
1406     
1407     
1408     
1409     
1410     
1411     
1412     
1413     
1414     
1415     
1416     
1417     
1418     
1419     
1420     
1421     
1422     
1423     
1424     
1425     
1426     
1427     
1428     
1429     
1430     
1431     
1432     
1433     
1434     
1435     
1436     
1437     
1438     
1439     
1440     
1441     
1442     
1443     
1444     
1445     
1446     
1447     
1448     
1449     
1450     
1451     
1452     
1453     
1454     
1455     
1456     
1457     
1458     
1459     
1460     
1461     
1462     
1463     
1464     
1465     
1466     
1467     
1468     
1469     
1470     
1471     
1472     
1473     
1474     
1475     
1476     
1477     
1478     
1479     
1480     
1481     
1482     
1483     
1484     
1485     
1486     
1487     
1488     
1489     
1490     
1491     
1492     
1493     
1494     
1495     
1496     
1497     
1498     
1499     
1500     
1501     
1502     
1503     
1504     
1505     
1506     
1507     
1508     
1509     
1510     
1511     
1512     
1513     
1514     
1515     
1516     
1517     
1518     
1519     
1520     
1521     
1522     
1523     
1524     
1525     
1526     
1527     
1528     
1529     
1530     
1531     
1532     
1533     
1534     
1535     
1536     
1537     
1538     
1539     
1540     
1541     
1542     
1543     
1544     
1545     
1546     
1547     
1548     
1549     
1550     
1551     
1552     
1553     
1554     
1555     
1556     
1557     
1558     
1559     
1560     
1561     
1562     
1563     
1564     
1565     
1566     
1567     
1568     
1569     
1570     
1571     
1572     
1573     
1574     
1575     
1576     
1577     
1578     
1579     
1580     
1581     
1582     
1583     
1584     
1585     
1586     
1587     
1588     
1589     
1590     
1591     
1592     
1593     
1594     
1595     
1596     
1597     
1598     
1599     
1600     
1601     
1602     
1603     
1604     
1605     
1606     
1607     
1608     
1609     
1610     
1611     
1612     
1613     
1614     
1615     
1616     
1617     
1618     
1619     
1620     
1621     
1622     
1623     
1624     
1625     
1626     
1627     
1628     
1629     
1630     
1631     
1632     
1633     
1634     
1635     
1636     
1637     
1638     
1639     
1640     
1641



# #9 – BESTEHENDEN CODE ERWEITERN

- GitHub Copilot Chat
- Modus: Edit oder Agent
- Im Edit Modus die gewünschten Dateien dem Kontext hinzufügen

Checks 1 Files changed



H

B

I

≡

<

Generate

## Summary

Generate a summary of the change request.

# #10 – PULL REQUEST BESCHREIBUNGEN GENERIEREN

- [GitHub.com](https://github.com)



## Pull Request Overview

This PR introduces a new xUnit test project using FluentAssertions to the existing solution.

- Registers `ReactAndAspNetCoreApp.Server.Tests` in the solution file with build configurations.
- Adds a test project `.csproj` with xUnit, FluentAssertions, Coverlet, and necessary settings.
- Scaffolds a placeholder unit test class with a TODO for implementing the first test.

## Reviewed Changes

Copilot reviewed 3 out of 3 changed files in this pull request and generated no comments.

| File   | Description  |
|--|--|
| src/ReactAndAspNetCoreApp.sln  | Added <code>ReactAndAspNetCoreApp.Server.Tests</code> project and its configurations           |
| src/ReactAndAspNetCoreApp.Server.Tests/UnitTest1.cs                              | Created placeholder test class <code>UnitTest1</code> with an empty <code>[Fact]</code> method |
| src/ReactAndAspNetCoreApp.Server.Tests/ReactAndAspNetCoreApp.Server.Tests.csproj | Defined the test project, enabled implicit usings, and added package refs                      |

# #11 - CODE REVIEWEN LASSEN

- GitHub Copilot
- Visual Studio Code oder GitHub.com

Please create a new work item of type User Story with title "Add LICENSE file to the repo"

settings.json:28-72

copilot-instructions.md

GitHub Copilot

Ran wit\_create\_work\_item - ado (MCP Server) ✓

Input

```
{
  "project": "DWX25",
  "workItemType": "User Story",
  "fields": {
    "System.Title": "Add LICENSE file to the repo"
  }
}
```

Output

```
"imageUrl": "https://dev.azure.com/[REDACTED]/_apis/
  GraphProfile/MemberAvatars/aad.
  [REDACTED]",
"descriptor": "aad.
  [REDACTED]",
},
"System.ChangedDate": "2025-07-02T13:48:36.057Z",
"System.ChangedBy": {
  "displayName": "Marc Rufer",
  "url": "https://spsprodweu5.vssps.visualstudio.com/
  [REDACTED]",
" links": {
```

A new work item of type "User Story" with the title "Add LICENSE file to the repo" has been created in your Azure DevOps project "DWX25".

You can view or edit it here:

[https://dev.azure.com/rufer7/3e2dcf57-7c19-4147-ba35-897aae753ca6/\\_workitems/edit/2](https://dev.azure.com/rufer7/3e2dcf57-7c19-4147-ba35-897aae753ca6/_workitems/edit/2)

Let me know if you want to add more details or link this work item to other artifacts.

🔄 👍 🔗

Add Context... settings.json Current file copilot-instructions Instructions

Edit files in your workspace in agent mode




Agent x Claude Sonnet 3.7 x

# #12 – AZURE DEVOPS INTEGRATION

- GitHub Copilot Chat
- Modus: Agent
- MCP Server: ado
- az login
- ado MCP Server starten

# LEARNINGS UND TIPPS



- Tastenkombinationen lernen   
- Custom Instructions verwenden
- Bei Fehlern und Problemen GitHub Copilot fragen
- Commit-Messages generieren lassen
- Pull Request Beschreibungen generieren lassen
- Dokumentationsteile anhand von Stichworten generieren lassen
- Vorsicht bspw. bei Terraform Konfigurationen
- Domänenspezifische Businesslogik -> Handwerk
- Qualitätssicherung
- ...



# FAZIT



# DU BIST DER PILOT!

**AI CODING... BUT REAL**

**VS**

**VIBE CODING CHAOS**

**COPILOT CLARITY MODE**

From quick hacks to production-grade refactors

- Nutzen
- Ausprobieren & testen
- Output verifizieren
- Erfahrungen teilen



# DANKE!

BLOG

Microsoft MVP Profil

GitHub: rufer7

LinkedIn: marcrufer

StackOverflow: rufer7

Bluesky: rufer.be



FRAGEN?



# LINKS

- [GitHub Copilot features](#)
- [Getting started with prompts for Copilot Chat](#)
- [Choosing the right AI model for your task](#)
- [Comparing AI models using different tasks](#)
- [MCP developer guide](#)
- [Azure DevOps MCP Server, Public Preview](#)