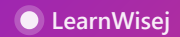


KI-Coding-Übung

Erstellen Sie das Beispielprojekt AccessOps dieses Moduls mit ChatGPT oder Claude aus einem fertigen Prompt, der das Modell auf die Spezifikation des Moduls verweist – und prüfen, starten und erweitern Sie dann, was Sie erhalten.



Dieses Beispiel tippen Sie nicht ab. Sie steuern es.

Kaum jemand schreibt Wisej.NET-Code heute noch in einer leeren Datei: Ein Assistent liefert den ersten Entwurf, und der Entwickler entscheidet, ob er stimmt. Das funktioniert aber nur, wenn Sie den Stoff beherrschen – und genau den hat Ihnen der Rest dieses Moduls vermittelt. Hier geben Sie dieses Wissen an ChatGPT oder Claude weiter, erhalten das Beispielprojekt des Moduls als lauffähige Visual-Studio-Projektmappe zurück und prüfen es so, wie Sie den Pull-Request eines Kollegen prüfen würden.

Worum Sie bitten

Legen Sie die AccessOps-Solution an und schreiben Sie ihr Bedrohungsmodell: Zeichnen Sie die Vertrauensgrenzen, listen Sie die Assets auf (Mitarbeiterdatensätze, Dokumente, Genehmigungen), erfassen Sie die Einstiegspunkte (Login, Uploads, Downloads, eigene Endpunkte, den KI-Assistenten), bewerten Sie jede Bedrohung und ordnen Sie jede Gegenmaßnahme dem Modul dieses Kurses zu, das sie umsetzt. Bauen Sie das Grundgerüst mit einer Platzhalter-Login-Seite.

Das Modul verlangt diese Ergebnisse, und der Prompt unten fordert genau diese vom Assistenten an:

- ✓ AccessOps-Solution mit einer Platzhalter-Login-Seite
- ✓ Diagramm der Vertrauensgrenzen
- ✓ Inventar der Assets und Einstiegspunkte
- ✓ Bedrohungstabelle mit Eintrittswahrscheinlichkeit, Auswirkung und Gegenmaßnahme
- ✓ Platzierungsübersicht der Sicherheitskontrollen

Warum es so aufgebaut ist

AccessOps ist ein internes Betriebsportal: Abteilungen, Mitarbeitende, vertrauliche Dokumente, Genehmigungsanträge und eine Admin-Konsole. Wer es erreichen kann, arbeitet ohnehin schon für das Unternehmen – genau deshalb wird die Sicherheitsarbeit immer wieder verschoben, und genau deshalb enthält am Ende ein einziges Portal Gehaltsbänder, Abmahnungen und die Genehmigungshistorie jeder Ausnahme, die das Unternehmen je gewährt hat. Bevor auch nur eine Zeile Authentifizierungscode existiert, brauchen Sie drei Antworten: was der Browser in der Hand hält, wo die Grenzen verlaufen und welche der möglichen Fehlentwicklungen es wert sind, zuerst technisch abgesichert zu werden.

Diese Begriffe muss der Assistent richtig verwenden. Geht die Antwort nachlässig mit ihnen um, hat er das Modul nicht verstanden:

Begriff	Was er hier bedeutet
serverseitiges UI-Modell	Controls sind C#-Objekte im Serverprozess; der Browser hält nur ein gerendertes Abbild und einen Event-Kanal, daher werden Ihr Code und Ihre Abfragen nie heruntergeladen.
Widget-Zustandsdaten	Die Eigenschaftswerte, die Wisej.NET serialisiert, damit der Browser ein Control zeichnen kann. Alles, was Sie zuweisen, geht über die Leitung – auch bei Controls, die der Benutzer nicht sieht.
Vertrauensgrenze	Eine Linie, an der Daten zwischen Parteien mit unterschiedlichem Vertrauensniveau wechseln. Alles, was sie überquert, prüft die vertrauenswürdigeren Seite erneut, denn die andere Seite kann lügen.
Session-Grenze	Die Grenze zwischen einem Browser und seinem eigenen Serverzustand in Application.Session. Entscheidungen beruhen auf dem Zustand auf der Serverseite dieser Grenze, nie auf einem Wert aus dem Browser.
Einstiegspunkt	Jeder Weg hinein, den ein Angreifer erreichen kann: die Login-Seite, Event-Handler, Uploads, Downloads, eigene HTTP-Handler, Interop-Aufrufe und der Prompt des KI-Assistenten.
STRIDE	Sechs Kategorien zur Benennung von Bedrohungen – Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege –, damit ein Review systematisch abläuft.
Platzierung von Kontrollen	Die Wahl der Grenze, an der eine Regel durchgesetzt werden kann. Eine Regel, die nur in der UI existiert, ist ein Vorschlag; dieselbe Regel in einem Service ist eine Kontrolle.

Selbst bauen – wählen Sie Ihren Assistenten

Beide Buttons öffnen einen neuen Tab, in dem **derselbe fertige Prompt** bereits eingetragen ist. Der Prompt verweist das Modell auf die Spezifikation dieses Moduls – eine einzelne Markdown-Datei mit der Lektion, der Lab-Aufgabe und den geforderten Ergebnissen –, sodass es mit dem echten Material arbeitet, statt zu raten, worum es in Modul 1 geht. Sie können [module-01.md](#) selbst öffnen, um genau zu sehen, was es lesen wird.



ChatGPT verwenden

Öffnet [chatgpt.com](#) mit dem Prompt dieses Moduls. Funktioniert mit GPT-5 oder Codex; jeder Tarif mit Websuche ruft die Spezifikation ab.



Claude verwenden

Öffnet [claude.ai](#) mit demselben Prompt. Wenn Sie Claude Code nutzen, können Sie ihn stattdessen in einem leeren Ordner in ein Terminal einfügen.



Unverändert abgeschickt, erhalten Sie ein Projekt, kein Code-Fragment

Der Prompt verlangt am Ende die gesamte Projektmappe als einzelne `.zip`-Datei mit Anleitung zum Kompilieren und Starten. Laden Sie sie herunter, öffnen Sie die Projektmappe in Visual Studio, drücken Sie F5, und AccessOps startet auf `localhost`. Sie benötigen das .NET SDK und das NuGet-Paket `Wisej-4`; alles andere soll der Assistent laut Prompt selbst schreiben.

Ich arbeite gerade Modul 1 – „Das Sicherheitsmodell von Wisej.NET“ – des LearnWisej-Kurses „Sicherheits- und Authentifizierungsmuster“ durch und möchte das Beispielprojekt des Moduls gemeinsam mit dir erstellen. Bitte antworte auf Deutsch.

Lies zuerst die Modulspezifikation. Sie ist eine einzelne Markdown-Datei (auf Englisch) mit der vollständigen Lektion, der Lab-Aufgabe und den geforderten Ergebnissen:

<https://learnwisej.com/assets/courses/security-and-auth-patterns/specs/module-01.md>

Erstelle dann ein vollständiges Wisej.NET-4-Projekt, das ich in Visual Studio öffnen kann:

1. Projekt „AccessOps“, C#, net8.0 oder neuer, NuGet-Paket Wisej-4 4.1.x, Kestrel-Host, der `app.UseWisej()` aufruft.
2. Setze um, was das Lab der Spezifikation verlangt: Legen Sie die AccessOps-Solution an und schreiben Sie ihr Bedrohungsmodell: Zeichnen Sie die Vertrauensgrenzen, listen Sie die Assets auf (Mitarbeiterdatensätze, Dokumente, Genehmigungen), erfassen Sie die Einstiegspunkte (Login, Uploads, Downloads, eigene Endpunkte, den KI-Assistenten), bewerten Sie jede Bedrohung und ordnen Sie jede Gegenmaßnahme dem Modul dieses Kurses zu, das sie umsetzt. Bauen Sie das Grundgerüst mit einer Platzhalter-Login-Seite.
3. Liefere diese Ergebnisse und sag mir, wo sich jedes davon in der Projektmappe befindet:
 - AccessOps-Solution mit einer Platzhalter-Login-Seite
 - Diagramm der Vertrauensgrenzen
 - Inventar der Assets und Einstiegspunkte
 - Bedrohungstabelle mit Eintrittswahrscheinlichkeit, Auswirkung und Gegenmaßnahme
 - Platzierungsübersicht der Sicherheitskontrollen
4. Halte dich an die Spezifikation, wo immer sie eine Wisej.NET-Klasse, -Eigenschaft oder ein -Event nennt – verwende genau diese. Wo sie eine Wahl offenlässt, sag mir, welche API du gewählt hast und warum.
5. Der Server bleibt maßgeblich: Zustand und Validierung leben in .NET, der Browser stellt dar und meldet zurück. Verlagere nie eine Entscheidung nach JavaScript, nur weil es dort einfacher ist.
6. Mach das Ergebnis vorführbar: Ich muss es starten und das Verhalten sehen können, um das es in diesem Modul geht – auch, was passiert, wenn etwas schiefgeht.

Bevor du Code schreibst, fasse in eigenen Worten zusammen, was du bauen wirst und wo jedes Teil liegen wird, damit ich den Entwurf korrigieren kann, solange das noch wenig kostet.

Wenn der Code fertig ist, packe die gesamte Projektmappe in eine einzelne .zip-Datei zum Herunterladen und nenne mir die genauen Schritte, um sie zu kompilieren und auszuführen.

Beantworte zum Schluss diese Fragen aus der Spezifikation und nutze dabei deine eigene Implementierung als Beleg:

- Ihr Mitarbeiter-Grid ist an vollständige Mitarbeiterdatensätze gebunden, und die Gehaltsspalte ist im Designer ausgeblendet. Beschreiben Sie, was der Browser in der Hand hält, und nennen Sie die Änderung, die das Gehalt tatsächlich von der Leitung nimmt.
- Zwei Personen sind auf demselben Server bei AccessOps angemeldet. Erklären Sie, welche Teile des Zustands sie standardmäßig teilen und was zutreffen müsste, damit eine von ihnen die aktuelle Abteilung der anderen sieht.
- Wählen Sie die Bedrohung in Ihrer Tabelle mit der größten Auswirkung und vertreten Sie die Gegenposition: Was müsste für AccessOps gelten, damit diese Gegenmaßnahme Zeitverschwendung für das Team wäre?

Frag mich nach allem, was unklar ist, bevor du anfängst.

Prüfen Sie dann, was zurückkommt

Code, der kompiliert, ist noch lange kein richtiger Code. Die spannenden Fragen sind Designfragen: Ist das wirklich das Wisej.NET-Control, das die Aufgabe verlangt? Folgt der Code dem Lebenszyklus, den das Framework vorgibt? Liegt der maßgebliche Zustand auf dem Server – oder wurde er im Browser abgelegt, weil das bequemer war? Lesen Sie das Ergebnis als Reviewer, nicht als Kunde.

Die Checkliste liefert Ihnen dieses Modul. Beantworten Sie jeden Punkt anhand des Codes, den Sie gerade erhalten haben:

- ✓ Ihr Mitarbeiter-Grid ist an vollständige Mitarbeiterdatensätze gebunden, und die Gehaltsspalte ist im Designer ausgeblendet. Beschreiben Sie, was der Browser in der Hand hält, und nennen Sie die Änderung, die das Gehalt tatsächlich von der Leitung nimmt.
- ✓ Zwei Personen sind auf demselben Server bei AccessOps angemeldet. Erklären Sie, welche Teile des Zustands sie standardmäßig teilen und was zutreffen müsste, damit eine von ihnen die aktuelle Abteilung der anderen sieht.
- ✓ Wählen Sie die Bedrohung in Ihrer Tabelle mit der größten Auswirkung und vertreten Sie die Gegenposition: Was müsste für AccessOps gelten, damit diese Gegenmaßnahme Zeitverschwendung für das Team wäre?

Alles, was Sie korrigieren mussten, sollten Sie festhalten. Eine kurze Notiz zu den Konventionen und Entscheidungen, auf die Sie sich festgelegt haben – eingefügt in die erste Nachricht Ihrer nächsten Sitzung –, ist mehr wert als jeder noch so clevere Prompt.



Für diesen Kurs gibt es noch kein Referenzbeispiel – Ihres ist das erste

Das [LearnWisej-Beispiel-Repository](#) enthält ausgearbeitete Builds für die meisten Kurse, für diesen aber noch nicht. Es gibt also nichts zum Vergleichen und keine Abkürzung: Die Spezifikation und Ihr eigenes Review sind das ganze Sicherheitsnetz. Behalten Sie, was Sie bauen – und wenn es gut wird, würden wir es gern sehen.

Drei Prompts für den nächsten Schritt

Richten Sie Ihren Assistenten auf das Projekt, das Sie gerade gebaut haben, und probieren Sie diese Prompts aus. Jeder ist eine Übung zum Prüfen und Erweitern, kein Neuschreiben.

- ✓ **Beweisen Sie, dass es wirklich funktioniert.** *"Geh dieses Projekt Ergebnis für Ergebnis anhand der Modulspezifikation unter <https://learnwisej.com/assets/courses/security-and-auth-patterns/specs/module-01.md> durch und zeig mir den Beleg, dass jedes wirklich erledigt ist – den Codepfad, worauf ich klicken muss und was ich sehen sollte. Wo ein Ergebnis nur teilweise erfüllt ist, sag das klar, statt es schönzureden. Ergänze dann, was fehlt."*
- ✓ **Prüfen Sie es wie einen Pull-Request.** *"Führ mich durch diesen AccessOps-Code, als würdest du den Pull-Request eines Kollegen prüfen und hättest ihn nicht selbst geschrieben. Wo lebt der Zustand, was passiert bei einer einzelnen Benutzeraktion von Anfang bis Ende, was würdest du zur*

Überarbeitung zurückgeben, und welche Teile würdest du nicht guten Gewissens einem Kunden vorlegen? Sei konkret und zitiere die Zeilen, die du meinst – fasse mir nicht den Entwurf zusammen."

- ✓ **Tauschen Sie ein Teil aus, ohne die Schnittstellen zu verschieben.** "Wähle den Teil dieses AccessOps-Projekts, der am offensichtlichsten nur ein Platzhalter für etwas Echtes ist – eine Fake-Datenquelle, ein In-Memory-Speicher, ein vereinfachter Service – und ersetze ihn durch eine praxisnahe Implementierung. Der Vertrag auf beiden Seiten darf sich nicht ändern: dieselben Typen, dieselben Aufrufe, dasselbe Verhalten im Fehlerfall. Sag mir, was du durch die echte Implementierung behandeln musstest, was der Platzhalter nicht behandelt hat."



Nehmen Sie es mit ins Lab

Das praktische Lab am Ende dieses Moduls verlangt dieselben Ergebnisse, die Sie gerade vom Assistenten haben erstellen lassen. Nutzen Sie, was Sie hier haben, als Gerüst für Ihre Antwort und als Beleg dafür – nicht als die Antwort selbst. Bewertet wird im Lab Ihre Argumentation, und dieser Teil muss von Ihnen stammen.