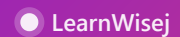


KI-Coding-Übung

Erstellen Sie das Beispielprojekt VisualOperationsStudio 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.



Der Assistent liefert die schnellen Hände. Der Architekt bleiben Sie.

Ein Assistent erstellt das Beispielprojekt dieses Moduls in wenigen Minuten. Was er nicht kann, ist zu entscheiden, ob das Ergebnis der richtige Entwurf für Ihre Anwendung ist – diese Entscheidung bleibt bei Ihnen. Das Urteilsvermögen bringen Sie mit: Instruieren Sie ihn sorgfältig und prüfen Sie dann, was zurückkommt, anhand dessen, was Ihnen Modul 2 gerade vermittelt hat.

Worum Sie bitten

Ergänzen Sie VisualOperationsStudio um ein wiederverwendbares TelemetryGauge-Control, das einen Ring, farbige Schwellenzonen, einen Zeiger, einen Zahlenwert und eine Beschriftung zeichnet. Geben Sie ihm die Properties Value, Minimum, Maximum, WarningThreshold und CriticalThreshold, deren Setter die Eingabe begrenzen, bei unveränderten Werten sofort zurückkehren und Invalidate() einmal aufrufen. Legen Sie die Koordinatenmathematik in einen seiteneffektfreien GaugeGeometry-Helfer, der ein Rectangle entgegennimmt und Ringgrenzen, Winkelbereiche, Zeigerpunkte und das Beschriftungsrechteck zurückgibt; zeichnen Sie dann aus diesen Werten mit per using begrenzten Pens, Brushes und einem GraphicsPath und lassen Sie das übergebene Graphics unangetastet. Behandeln Sie ein Control der Größe null, einen umgekehrten Wertebereich und einen Wert außerhalb des Bereichs, ohne eine Exception auszulösen, und veröffentlichen Sie den Messwert über AccessibleDescription, damit er nicht nur als Pixel existiert.

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

- ✓ TelemetryGauge-Control mit begrenzenden Value-, Minimum-, Maximum- und Schwellen-Properties, die pro echter Änderung genau einmal Invalidate() aufrufen
- ✓ Seiteneffektfreier GaugeGeometry-Helfer, der Ringgrenzen, Winkelbereiche, Zeigerpunkte und das Beschriftungsrechteck liefert
- ✓ Paint-Handler, der jeden selbst erzeugten Pen, Brush, Font und Path freigibt und e.Graphics niemals freigibt
- ✓ Absicherungen für Größe null, umgekehrten Bereich und Werte außerhalb des Bereichs, dazu eine AccessibleDescription mit dem Messwert
- ✓ PaintCost.md mit Notizen zur Größe des neu gezeichneten Bereichs, zur Häufigkeit des Neuzeichnens und dazu, wann ein gethemtes Control günstiger wäre

Warum es so aufgebaut ist

Modul 1 hat einen Telemetriewert auf vier Oberflächen gebracht, damit Sie sehen, wo jedes Pixel entsteht. VisualOperationsStudio will diesen Messwert nun an sechs Stellen, und einen Paint -Handler in sechs Panels zu kopieren ist der Weg, auf dem eine Oberfläche mit sechs leicht unterschiedlichen Gauges endet, mit sechs Pens, die nie freigegeben werden, und mit einem Neuzeichnen, sobald sich irgendein unbeteiligtes Feld ändert. Die Lösung ist, keine Handler mehr zu schreiben, sondern ein Control: ein einziges TelemetryGauge, dessen öffentliche Properties das Einzige sind, was sein Bild ändern kann, und dessen Kosten Sie in Kilobyte beziffern können.

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
Wisej.Web.PaintEventArgs	Das Argument des Paint -Events; es übergibt Ihnen ein System.Drawing.Graphics, das Sie nicht erzeugt haben, und ein Rechteck, in das Sie zeichnen – und Ihr Code läuft auf dem Server.
ClipRectangle	Der Bereich, den Wisej.NET für diesen Zeichenvorgang bereitgestellt hat; das Layout darf die Clientgröße des Controls heranziehen, aber das Zeichnen bleibt an das übergebene Rechteck gebunden.
Helfer GaugeGeometry	Eine reine Funktion aus einem Rectangle plus einem normalisierten Wert zu Ringgrenzen, Winkelbereichen, Zeigerpunkten und einem Beschriftungsrechteck, sodass die Mathematik ohne lebendes Control geprüft werden kann.
Invalidate()	Der eine Aufruf, der Wisej.NET bittet, die Oberfläche des Controls neu zu rendern; ein Property-Setter macht das genau einmal und nur, wenn die Änderung ein Pixel verändern kann.
Ressourcenbesitz	Der Handler gibt jeden Pen, Brush, Font, GraphicsPath und jede Bitmap frei, die er erzeugt, und gibt niemals das vom Framework gelieferte Graphics frei.
Kostenbudget fürs Zeichnen	Drei Zahlen pro gezeichnetem Control: Server-CPU fürs Zeichnen, Serverspeicher für das temporäre Bild und die Bytes, die der Browser laden und dekodieren muss.
AccessibleDescription	Trägt denselben Messwert als Text, sodass ein Wert, der als gezeichneter Bogen existiert, auch für eine Benutzerin verfügbar bleibt, die den Bogen nie sieht.

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 2 geht. Sie können [module-02.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 VisualOperationsStudio 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 2 – „Das Paint-Event von Controls meistern“ – des LearnWisej-Kurses „Zeichnen und Grafik in Wisej.NET“ 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/drawing-and-painting/specs/module-02.md>

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

1. Projekt „VisualOperationsStudio“, 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: Ergänzen Sie `VisualOperationsStudio` um ein wiederverwendbares `TelemetryGauge`-Control, das einen Ring, farbige Schwellenzonen, einen Zeiger, einen Zahlenwert und eine Beschriftung zeichnet. Geben Sie ihm die Properties `Value`, `Minimum`, `Maximum`, `WarningThreshold` und `CriticalThreshold`, deren Setter die Eingabe begrenzen, bei unveränderten Werten sofort zurückkehren und `Invalidate()` einmal aufrufen. Legen Sie die Koordinatenmathematik in einen seiteneffektfreien `GaugeGeometry`-Helfer, der ein `Rectangle` entgegennimmt und Ringgrenzen, Winkelbereiche, Zeigerpunkte und das Beschriftungsrechteck zurückgibt; zeichnen Sie dann aus diesen Werten mit `per using` begrenzten Pens, Brushes und einem `GraphicsPath` und lassen Sie das übergebene `Graphics` unangetastet. Behandeln Sie ein Control der Größe null, einen umgekehrten Wertebereich und einen Wert außerhalb des Bereichs, ohne eine Exception auszulösen, und veröffentlichen Sie den Messwert über `AccessibleDescription`, damit er nicht nur als Pixel existiert.
3. Liefere diese Ergebnisse und sag mir, wo sich jedes davon in der Projektmappe befindet:
 - `TelemetryGauge`-Control mit begrenzenden `Value`-, `Minimum`-, `Maximum`- und Schwellen-Properties, die pro echter Änderung genau einmal `Invalidate()` aufrufen
 - Seiteneffektfreier `GaugeGeometry`-Helfer, der Ringgrenzen, Winkelbereiche, Zeigerpunkte und das Beschriftungsrechteck liefert
 - `Paint`-Handler, der jeden selbst erzeugten Pen, Brush, Font und Path freigibt und `e.Graphics` niemals freigibt
 - Absicherungen für Größe null, umgekehrten Bereich und Werte außerhalb des Bereichs, dazu eine `AccessibleDescription` mit dem Messwert
 - `PaintCost.md` mit Notizen zur Größe des neu gezeichneten Bereichs, zur Häufigkeit des Neuzeichnens und dazu, wann ein gethemtes Control günstiger wäre
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:

- Ein Kollege verlagert die Mathematik für Ring und Zeiger direkt in den Paint -Handler, weil es „nur ein paar Zeilen“ sind. Welche zwei Dinge lassen sich danach nicht mehr günstig tun, und auf welches davon baut Modul 6 auf?
- Ihr Gauge ist an einen Maschinendatensatz gebunden, der auch einen Bedienernamen und ein Wartungsdatum trägt. Sollte das Control invalidieren, wenn sich diese Felder ändern? Begründen Sie Ihre Antwort über visuelle Abhängigkeiten und nicht über das Modell.
- Dasselbe Gauge soll auf ein Dashboard, das sechzig davon zeigt und sich alle zwei Sekunden aktualisiert. Was würden Sie zuerst messen, und was würden Sie ändern, wenn die Messwerte schlecht ausfallen?

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:

- ✓ Ein Kollege verlagert die Mathematik für Ring und Zeiger direkt in den Paint -Handler, weil es „nur ein paar Zeilen“ sind. Welche zwei Dinge lassen sich danach nicht mehr günstig tun, und auf welches davon baut Modul 6 auf?
- ✓ Ihr Gauge ist an einen Maschinendatensatz gebunden, der auch einen Bedienernamen und ein Wartungsdatum trägt. Sollte das Control invalidieren, wenn sich diese Felder ändern? Begründen Sie Ihre Antwort über visuelle Abhängigkeiten und nicht über das Modell.
- ✓ Dasselbe Gauge soll auf ein Dashboard, das sechzig davon zeigt und sich alle zwei Sekunden aktualisiert. Was würden Sie zuerst messen, und was würden Sie ändern, wenn die Messwerte schlecht ausfallen?

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/drawing-and-painting/specs/module-02.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 VisualOperationsStudio-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 VisualOperationsStudio-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.