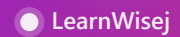


Projektmodernisierung und Migrationsstrategie für Wisej.NET 4

Lektionsleitfaden · Lab- und Prüfungsleitfaden



Lektionsleitfaden

Upgrade-Planung, Kompatibilitätsanalyse, inkrementelle Migration, Absicherung gegen Regressionen. Was das für ein Wisej.NET-Team auf Expertenniveau bedeutet – und wie Sie dieses Modul lesen.



Fokus: Produktivbetrieb

Upgrade-Planung, Kompatibilitätsanalyse, inkrementelle Migration, Absicherung gegen Regressionen.

1. Das Problem im Produktivbetrieb

Viele erfahrene Wisej.NET-Teams betreuen große bestehende Anwendungen. Sie müssen Projektstruktur, Ziel-Frameworks, Ressourcen, Themes und Annahmen zum Deployment modernisieren, ohne mit der Geschäftskontinuität zu spielen.

2. Zentrale Konzepte

Konzept	Warum es wichtig ist
Migrationsbestandsaufnahme	Eine vollständige Liste der Projekte, Pakete, Themes, Ressourcen und eigenen Komponenten der bestehenden App, damit nichts erst mitten im Upgrade entdeckt wird.
Kompatibilitätsmatrix	Eine Tabelle aus Ziel-Frameworks, Wisej.NET-Versionen und Drittanbieterpaketen, die zeigt, welche Kombinationen unterstützt werden, bevor Code geändert wird.
inkrementelles Upgrade	Ein Projekt oder eine Schicht nach der anderen umstellen, mit einem funktionierenden Build nach jedem Schritt, statt einer einzigen Big-Bang-Umstellung, die sich nicht verifizieren lässt.
Ressourcen-Mapping	Festhalten, wo Bilder, Skripte, Themes und eingebettete Ressourcen vor und nach der Migration liegen, damit im Produktivbetrieb nichts stillschweigend kaputtgeht.
Theme-Migration	Eigene Themes und Mixins in das aktuelle Theme-Format überführen und prüfen, dass jeder Bildschirm mit der neuen Engine korrekt dargestellt wird.
Regressions-Harness	Ein wiederholbarer Satz automatisierter und skriptgesteuerter Prüfungen, der belegt, dass das bestehende Verhalten nach jedem Migrationsschritt weiterhin funktioniert.

3. Der rote Faden der Lektion

Modernisierung ist Risikomanagement

Ein Upgrade soll eine bessere Anwendung hervorbringen, darf aber nicht zu einer unkontrollierten Neuentwicklung werden. Experten behandeln eine Migration als Abfolge überprüfbarer Änderungen: kompilieren, ausführen, Verhalten vergleichen, Themes prüfen, Sessions verifizieren, Deployment reviewen und Performance messen.

Erst Bestandsaufnahme, dann Handeln

Bevor Sie Code anfassen, listen Sie Ziel-Framework, Paketversionen, Designer-Abhängigkeiten, eigene Widgets, Themes, eingebettete Ressourcen, Startdateien, Einstiegspunkte der Authentifizierung, Deployment-Ziele und Integrationen von Drittanbietern auf. Diese Bestandsaufnahme wird zur Landkarte der Migration und zur Checkliste für das abschließende Review.



Expertenregel für den Produktivbetrieb

Hören Sie nicht auf, sobald der Bildschirm korrekt aussieht. Erklären Sie, wem der Zustand gehört, welche Sicherheitsfolgen es gibt, welche Fehlerpfade existieren und wie sich das System im Produktivbetrieb verhält, bevor Sie das Modul als abgeschlossen betrachten.



Lernziele

Am Ende dieses Moduls können Sie:

- ✓ eine Migrationsbestandsaufnahme für bestehende Wisej.NET-3.x-Projekte oder Projekte im Legacy-Stil erstellen.
- ✓ Änderungen an Projekten, Ressourcen, Themes, Deployment und APIs identifizieren, die ausdrücklich getestet werden müssen.
- ✓ einen inkrementellen Migrationspfad mit Rückfallpunkten entwerfen.
- ✓ eine Regressions-Checkliste erstellen, die das Verhalten schützt und nicht nur die Kompilierbarkeit.

Lab- und Prüfungsleitfaden

Was Sie im Praxis-Lab bauen, das vorgeschlagene Vorgehen und die geforderten Ergebnisse.



Praxis-Lab

Erstellen Sie ein Migrationsdossier für die TicketOps Console aus dem Fortgeschrittenen-Kurs und überführen Sie sie in die Projektbasis des Experten-Kurses. Nehmen Sie eine Bestandsaufnahme des Ist-Zustands, eine Bestandsaufnahme des Soll-Zustands, eine Risikomatrix, einen Regressionsplan für die Bildschirme und eine Rollback-Strategie auf.

Vorgeschlagenes Vorgehen

- ✓ Lesen Sie das Problem im Produktivbetrieb und benennen Sie das Architekturrisiko, bevor Sie die UI anfassen.
- ✓ Legen Sie zuerst die betroffene Grenze für Service, Command, Modell oder Komponente an oder aktualisieren Sie sie.
- ✓ Erstellen Sie den Wisej.NET-Bildschirm oder die Komponente nach den etablierten Projektstandards.
- ✓ Demonstrieren Sie auch einen Fehlerpfad, nicht nur den Happy Path.
- ✓ Halten Sie die Änderungen im Lernprotokoll fest und aktualisieren Sie die zugehörige Checkliste.

Geforderte Ergebnisse

- ✓ Tabelle der Migrationsbestandsaufnahme
- ✓ Kompatibilitäts- und Risikomatrix
- ✓ Regressionstestplan für zehn zentrale Abläufe
- ✓ Rollback-Plan
- ✓ Entscheidungsmemo zur Modernisierung

Wiederholungsfragen

- ✓ Welches Verhalten muss exakt unverändert bleiben?
- ✓ Welcher Migrationsschritt lässt sich unabhängig zurückrollen?
- ✓ Welche Bildschirme verbergen am ehesten Annahmen über Timing oder Zustand?



Expertenregel für den Produktivbetrieb

Hören Sie nicht auf, sobald der Bildschirm korrekt aussieht. Erklären Sie, wem der Zustand gehört, welche Sicherheitsfolgen es gibt, welche Fehlerpfade existieren und wie sich das System im Produktivbetrieb verhält, bevor Sie das Modul als abgeschlossen betrachten.