

Themen zum Recherchieren und Erweitern

1. IServices

Aufwand: Groß

Um die Business Logik im modularen Style zu halten müssen diese in sogenannte Services ausgelagert werden. Services sollten Ressort Orientiert sein, als Beispiel, ein Service zum Anmelden an der MS Graph API sollte auch nur das Verwalten, also Sign In, Sign Out und Refresh.

PoC Vorschlag:

Baue 2 Services um mit einer SQLite DB zu kommunizieren.

- Users und Groups, jede sollte in ihrem Ressort CRUD beherrschen.
- 3 Tabellen: Users, Groups und UserGroups, letztere enthält die ID von Users und Groups die eine Relation haben.
- Das VM Users hat Zugriff auf die Services Users und Groups, letzteres um die Gruppen anzuzeigen, aber nicht manipulieren.
- Das VM Groups hat nur Zugriff auf den Service Groups.
- Achte darauf das die jeweiligen Services nur auf ihre jeweilige Tabelle Zugriff haben + UserGroups.

2. Unit Testing (xUnit)

Aufwand: Groß

Die Wichtigkeit von Tests steht außerfrage und das weglassen ist "Bad Practice". Leg eine Doku an für die Anwendungszwecke IService und Viewmodel.

PoC Vorschlag:

Implementiere Testing in das IService PoC.

3. Lokalisierung

Aufwand: Klein

Im Moment haben wir die Möglichkeit UI/UX über Ressourcen zu lokalisieren, aber dafür ist ein Neustart nötig.

- Ist es möglich dies in der Laufzeit zu machen?
- Ist die aktuelle Methode die beste oder gibt es modernere Alternativen?

4. WrapPanel / Flow Design

Aufwand: Mittel

Wir kennen nun Grids, diese sind Super wenn es darum geht Inhalt inklusive Skalierung zu fixieren, aber dadurch geht auch die Dynamik verloren. Wir haben gesehen dass mit WrapPanel dies möglich ist, aber haben einen erheblichen Kontrollverlust. Versuche WrapPanels zu beherrschen.

PoC Vorschlag:

Baue eine kleine App in der du 7 gleich große Kontakte anzeigst (Bild, Name, Email, Telefon), jeweils mit fixer höhe. Diese sollen vertikal gestapelt sein, aber wenn unten kein Platz mehr ist nach rechts ausweichen.

Der Clou an der Sache ist, die Kontakt Karten sollen immer die maximale breite ausnutzen, heißt, bei 2 Columns, jeweils 50%, bei 3 33% usw.