# Code

## Autorenschaft

Die Autorenschaft ist gemeinschaftlich und wird deshalb nicht aufgelistet.

## Reviews

Um die Code Qualität zu gewährleisten, wurden im Team immer wieder Code Reviews durchgeführt. Diese wurden im Team vor dem Ende jedes Sprints partnerschaftlich durchgeführt.

Zusätzlich zu den internen Code Reviews wurden auch noch Code Reviews mit Christian Moser und Michael Gfeller durchgeführt. Die Details dazu sind untenstehend aufgelistet.

### 25.11.2011

Review mit: Christian Moser

Anwesend: Christian Moser, Christina Heidt, Delia Treichler, Lukas Elmer

Besprochene Punkte:

* Um die Animationen zu gestalten, wurden VisualStateGroups eingesetzt. Diese wurden im Team besprochen und im Expression Blend erstellt.
* Um spezielle Events abzufangen können statische Klassen eingesetzt werden. So benötigt man keinen Code Behind.
* Es kann mit wenig Code Behind gearbeitet werden, solange sich dieser nicht auf die Business Logik sondern nur auf das User Interface auswirkt.
* Die Animationen werden nicht richtig angezeigt, da die Rechenleistung für das Laden des XPS Dokumentes verwendet wird. Aus diesem Grund wurde das Dokument dann erst nach der Animation geladen.

### 9.12.2011

Review mit: Michael Gfeller

Anwesend: Michael Gfeller, Christina Heidt, Delia Treichler, Lukas Elmer

* Da das XAML des OverviewWindows zu lang und gross wurde, sollen die Styles in ein Styles.xaml ausgelagert werden. Diese können dann mithilfe eines ResourceDirectory ins Xaml eingebunden werden (Merged Resource Dictionaries).
* Images sind nicht freezed 🡪 dadurch könnten Memory Leaks entstehen
* ScrollToTopBehavior: DependencyPropertyDescriptor ist statisches Konstrukt (ItemsSourceProperty): prüfen, dass es sich abmeldet, sonst könnte hier ein Memory Leak entstehen.
* Die einzelnen Klassen sollen aufgeräumt werden: Verschiedene Regionen von oben nach unten: Deklarationen, Properties, Konstruktor, dann private Variablen und Methoden.
* Absoluter Pfad zu Adobe Reader soll in ein Konfigurations File ausgelagert werden.
* Installer fehlt, da aber keine Hardware verfügbar ist, kann dies noch nicht gemacht werden. In Dokumentation soll erklärt werden, was alles noch getan werden muss, um Projekt auf dem Surface 2 zu benutzen.
* Code dokumentieren für public Methoden/Properties/Klassen, falls sich dies bei Abschnitt lohnt (speziell wenn nicht klar ist, um was es sich handelt, z.B. bei Preload() im ProjectNote Model).

Beschlüsse:

* Da nur ein ViewModel existiert, wird kein separates Projekt für die ViewModels erstellt.

### 16.12.2011

Review mit: Michael Gfeller

Anwesend: Michael Gfeller, Lukas Elmer

* Laufzeit Warnings: Beim Binding gibt es Laufzeit Warnings und zwar in der Detailansicht beim Navigieren von einer Project Note zur nächsten. Dies wird durch das Binding verursacht und könnte daran liegen, dass auf null gebindet wird. Die Behebung dieser Warnings würde wahrscheinlich zu viel Zeit in Anspruch nehmen.
* Im GravatarsViewModel wird eine IList verwendet. Besser wäre hier eine ListView, da durch das automatische Binding / Konvertieren ein Memory Leak entstehen könnte.
* Durch die Installation des WPF Performance Kit könnte man die Resourcen besser überwachen und beobachten.
* GravatarsViewModelTest: Dieser Test bringt nur wenig Mehrwert.
* Die Konfiguration app.config in den unteren Projekten (Bsp. PdfConverter) bringt nichts, da die Datei nicht ins Verzeichnis mit den ausführbaren Dateien kopiert wird.

Beschlüsse:

* Aufgrund Zeitdruck ist es nicht möglich, alle beanstandeten Punkte umzusetzen. Deshalb werden die Laufzeit Warnings, die sehr wahrscheinlich durch das Binding auf null Objekte ausgelöst wird, ignoriert.
* Die Konfiguration wird noch kopiert, damit Einstellungen vorgenommen werden können.
* Die anderen Punkte werden berücksichtigt, falls noch Zeit bleiben sollte.