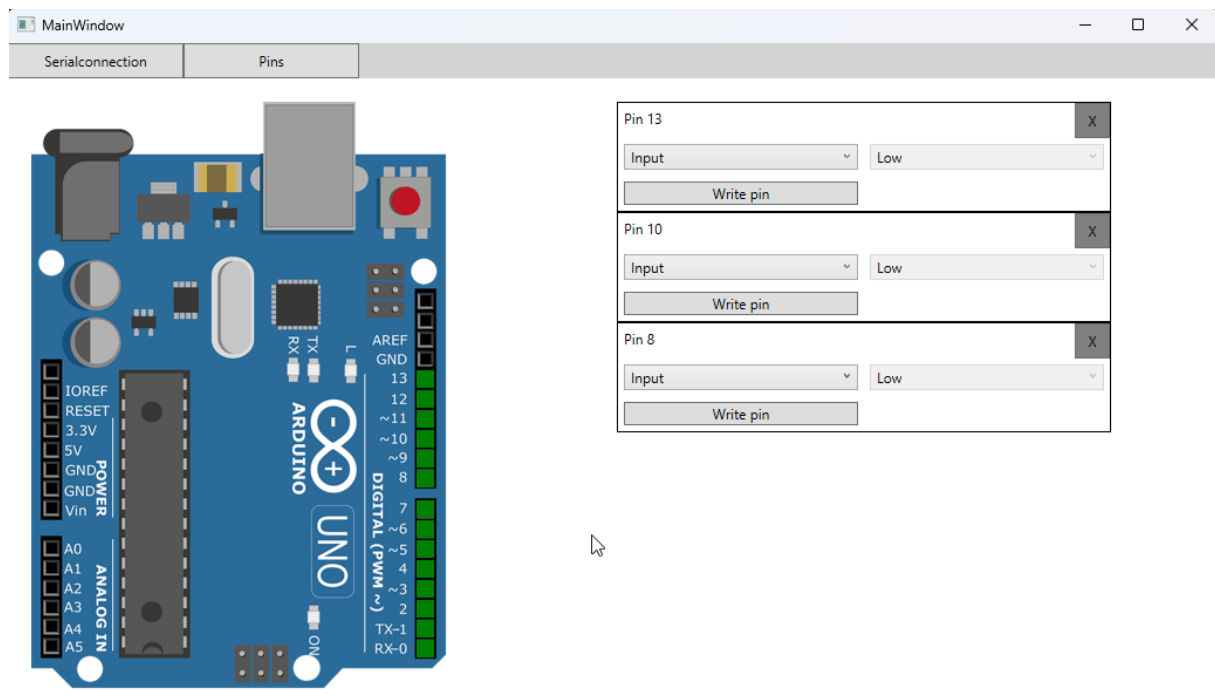


# Control a Microccontroller over WPF



## Inhalt

Einführung .....	3
Grund für das Projekt .....	3
Was ist das Ziel des Projektes .....	3
Projektablauf.....	4
Projektdefinitionsphase .....	4
UML .....	5
Programmierphase .....	7
Fazit.....	8
Probleme.....	8
Fazit .....	8

# Einführung

## Grund für das Projekt

Wir sind beide leidenschaftliche Softwareentwickler und basteln auch oft etwas mit verschiedensten Elektronikkomponenten. Diese Bastelprojekte erfordern oft den Einsatz eines Microcontrollers.

Unsere Diplomarbeit wird die Kommunikation und Steuerung eines Microcontrollers über externe Software beinhalten. Diese Kommunikation ist ein essenzieller Teil der Arbeit und konnte im Rahmen dieses FSST-Projektes gut erforscht und ausprobiert werden.

## Was ist das Ziel des Projektes

Schlussendlich wollen wir über eine C#-WPF Anwendung die Pins eines Microcontrollers steuern können.

Um den Anforderungen unserer Diplomarbeit gerecht zu werden, setzten wir uns für die Art der Kommunikation folgende Voraussetzungen:

- Die Kommunikation soll mithilfe von JSON-objects erfolgen
- Der Arduino soll die objecte verarbeiten, validieren und dann die geforderte Aktion ausführen (in diesem Fall nur Pins steuern)

Abgesehen von den Anforderungen der Kernkomponente Kommunikation, setzten wir uns zu Beginn des Projekts weitere projektspezifische Ziele:

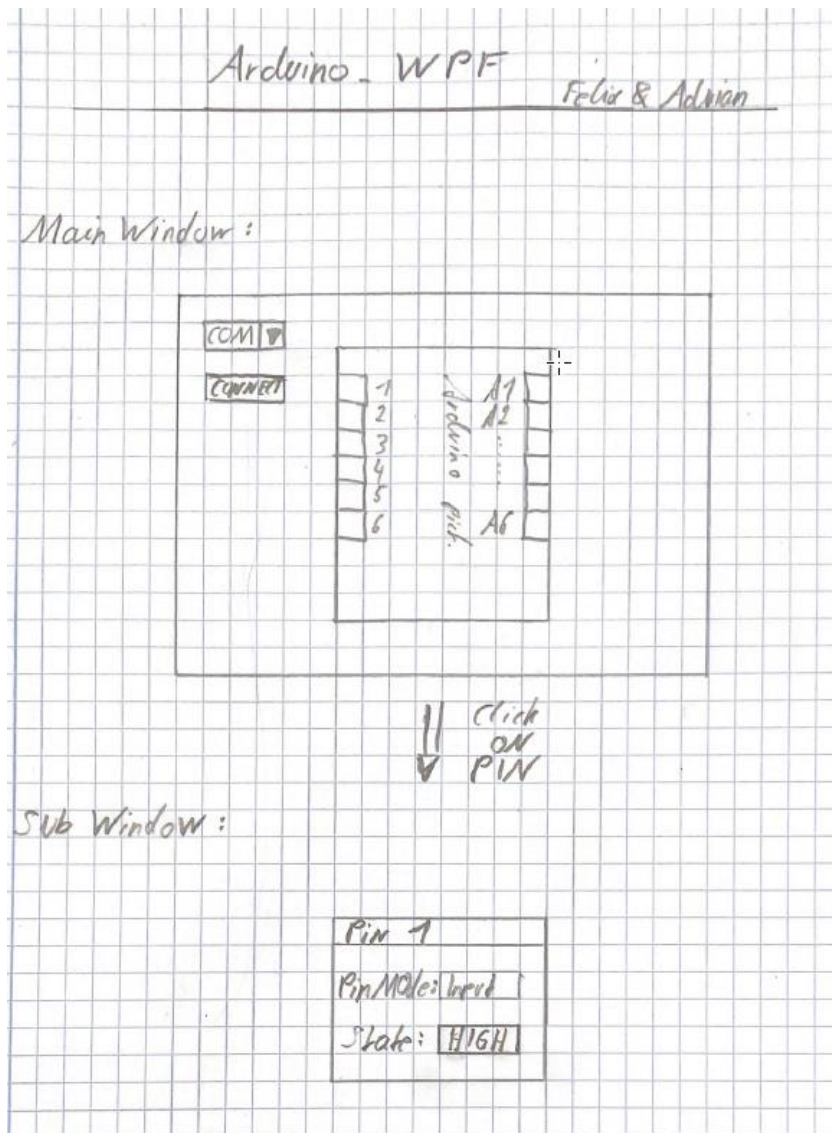
- Qualitativer code
- Wartbarer code
- Adaptiver code

Diese Ziele sollten mithilfe von Architektur und Design-Pattern erreicht werden.

# Projektablauf

## Projektdefinitionsphase

Die ersten Schritte waren die Definition der oben genannten Ziele und dann die ersten Skizzen und Entwürfe der Anwendung. Unsere damalige Vorstellung der Anwendung sah folgendermaßen aus:

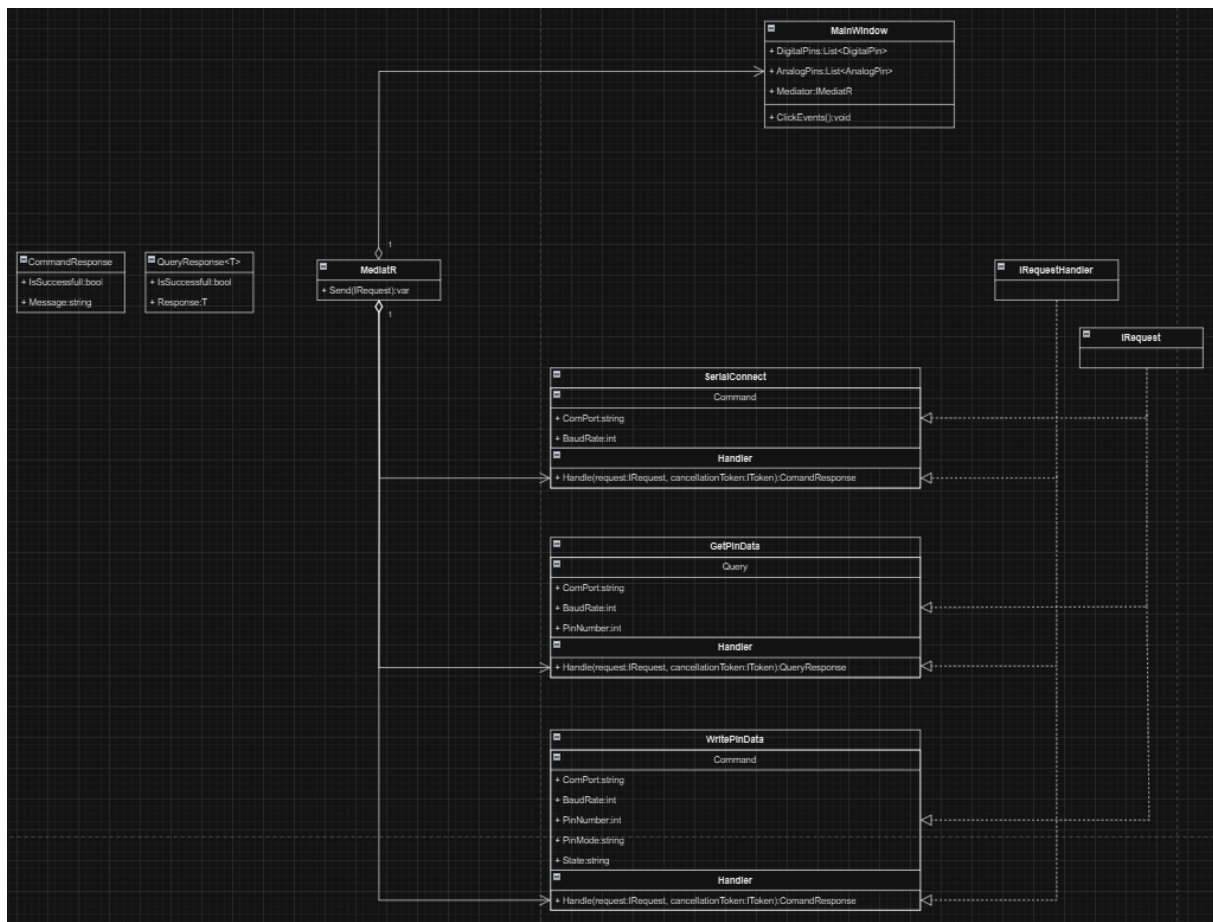


Nachdem wir uns im Klaren waren auf was wir hinarbeiten werden konnte es schon mit der Architektur und dem Klassendiagramm losgehen.

## UML

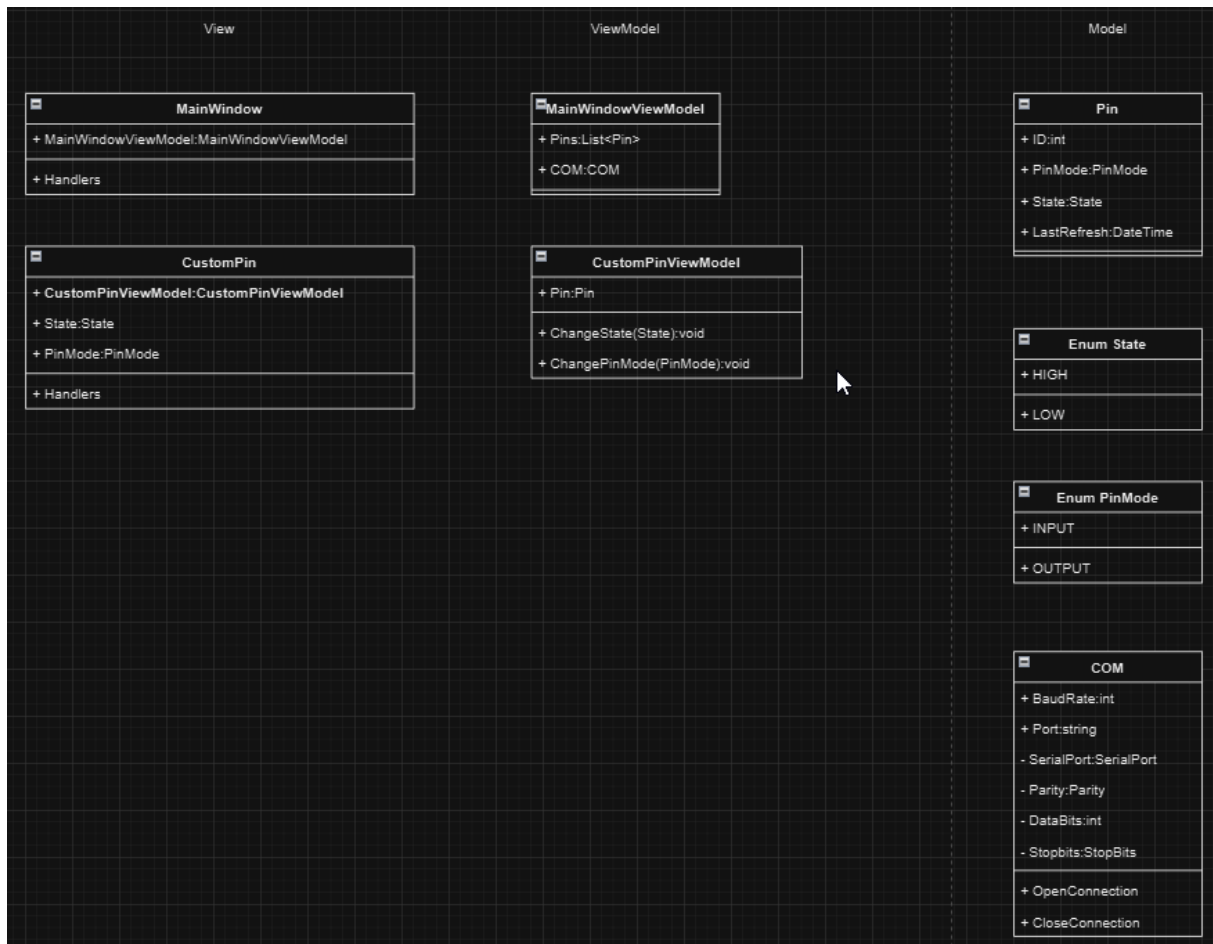
Bevor wir mit dem Konstruieren des UML's begonnen haben, versuchten wir uns Gedanken über die Architektur zu machen. Wir haben beide noch nie selbst die Architektur vorgegeben, was uns schlussendlich auch zum Verhängnis wurde.

Unsere erste Idee war der Ansatz eine Vertical-Slice Architektur mithilfe von MediatR zu realisieren:



Für bessere Ansicht siehe Git-Repo.

Nach weiteren Überlegungen kamen wir jedoch zum Entschluss, dass diese Variante nicht so gut geeignet ist. Schlussendlich entschieden wir uns für die klassische MVVM Variante. Für den MVVM-Ansatz zeichnen wir noch ein UML:



Für bessere Ansicht siehe Git-Repo.

Mit diesem Ansatz waren wir zufrieden und bereit mit dem Programmieren zu beginnen.

## Programmierphase

Die Programmierung ging gut voran und wir ergänzten uns im Team sehr gut. Jeder machte seinen Teil und wir verloren trotzdem den Überblick über das ganze nicht aus den Augen.

Die von uns gewünschte Qualität konnte auf jeden Fall erreicht werden und wir sind ziemlich zufrieden mit dem Ergebnis.

Wer, was, wann gemacht hat ist auch auf dem [Git-Repo](#) in Form von Commits ersichtlich (bitte entschuldigen Sie die teilweise nicht so netten Commit-Messages).

## Fazit

### Probleme

Wir hatten viele Probleme während des gesamten Projekts. Es ist, vor allem in der Softwareentwicklung, ganz normal, dass nicht alles so läuft wie ursprünglich geplant. Aus diesem Grund haben wir auch agil mittels SCRUM gearbeitet.

Nichtsdestotrotz gibt es ein paar Punkte, die wir auf jeden Fall erwähnen wollen.

1. Wir versuchten das ganze sehr professionell anzugehen und mit einer Architekturbesprechung zu starten. Wir sind jedoch beide nicht so geübt mit dem WPF-Framework wie jemand sein sollte, um solche Entscheidungen treffen zu können. Diese Thematik hat uns viel Zeit gekostet, aber wir konnten trotzdem unsere Learnings daraus ziehen.
2. Die Verwendung des MVVM-patterns war eine gute Idee, hat uns jedoch beide etwas Zeit gekostet, um den Ansatz und auch die Implementierung zu verstehen. Wir sind aber hartnäckig geblieben und konnten es am Schluss sogar etwas Lieben lernen.
3. Eine große Herausforderung war auch das gemeinsame Arbeiten an einem komplett neuen Projekt. Wir kommen aus unterschiedlichen Welten der Softwareentwicklung, in der unterschiedliche Programmierstile und Ansätze verfolgt werden. Für unser Vorhaben gab es natürlich auch kein Standard-Prozedere, an das wir uns halten konnten. Rückblickend können wir sagen, dass uns die Zusammenarbeit gut gelungen ist und wir die Herausforderung gemeistert haben.

### Fazit

Alles in allem sind wir beide überaus zufrieden mit dem Ergebnis. Wir konnten viel dabei lernen und es hat uns auch Spaß gemacht das Projekt durchzuführen. Das Projekt hat noch viel Potenzial zu Erweiterungen und eventuell werden auch noch nach Abschluss des Projektes aus Schulsicht ein paar Features dazukommen.