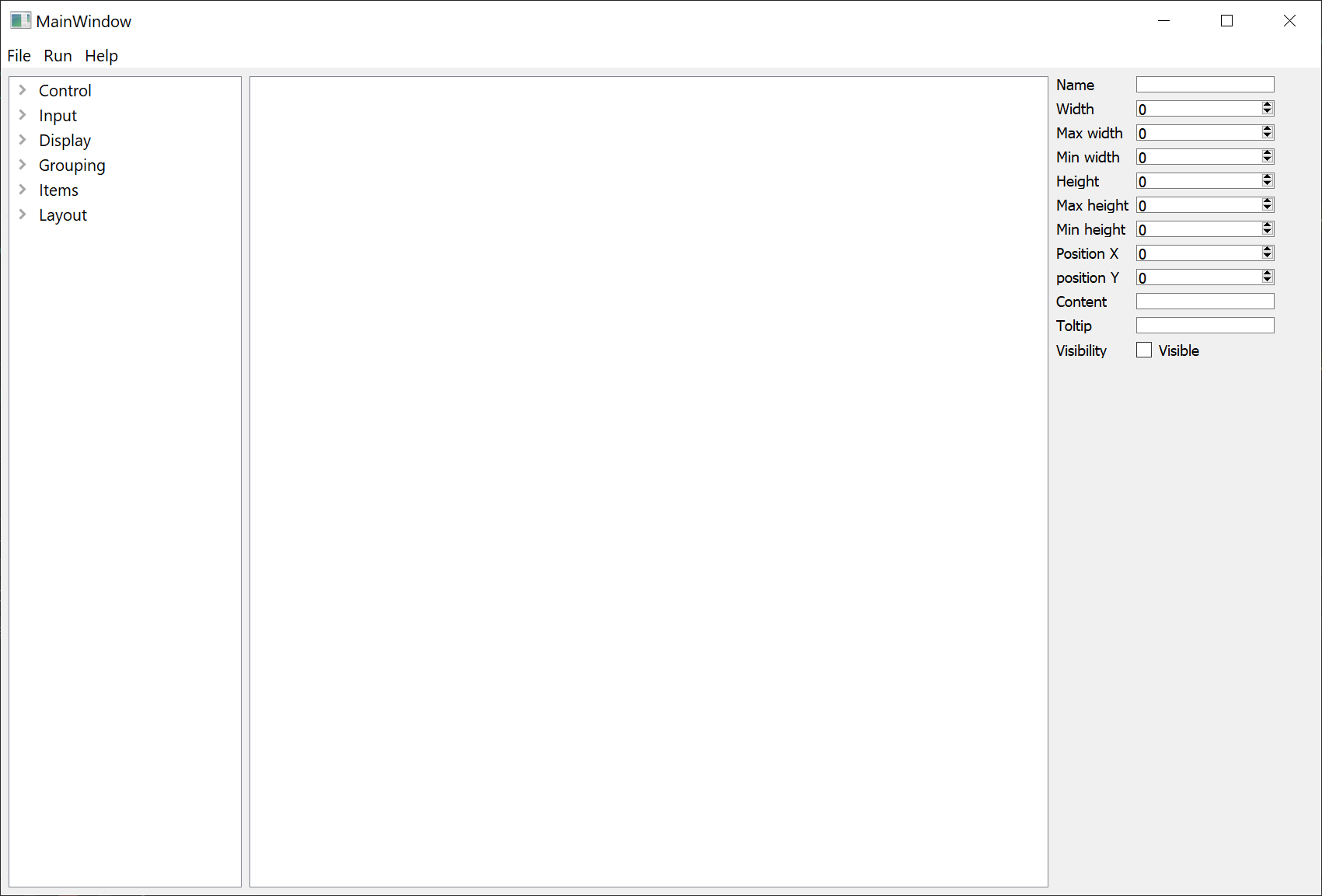
Abgegeben von: *Milewski, Nils*

Matrikelnummer: *10010480*

## Aufgabe 1.a

Mein Programm sollte im Bereich der Hobbysoftwareentwicklung sowie Start-Ups eingesetzt werden, meine Zielgruppe sind demnach Entwickler, die ohne Kommerzielle Lösungen ein Interface erstellen und dieses in diverse Sprachen exportieren wollen. Das Ziel ist ein einfacher Interfacebuilder der intuitiv bedienbar die hieraus resultierenden Anforderungen sind eine vertraute Umgebung, die ein Umstieg erleichtern soll bzw. eine erst Nutzung positiv beeindruckt.

## Aufgabe 1.b



1. Das TreeWidget sortiert alle möglichen UI-Elemente in Kategorien, dies ermöglicht eine einfache und schnell suche der Widgets
2. Die Details zeigen alle wichtigen Attribute des ausgewählten Elements an hierdurch werden eine schnelle Übersicht und Manipulation über wichtige Eigenschaften möglich.
3. Unter dem Menüpunkt „Run“ befindet sich die Möglichkeit den Szenegraph zu exportieren.

## Aufgabe 1.c

Minimalziel: Ein lauffähiges Programm, in dem es möglich ist Elemente aus einer Übersicht in eine Zeichenfläche zu ziehen.

Standartziel: Die Verbindung zwischen ausgewähltes Element der Zeichenfläche und das Anzeigen seiner Attribute.

Maximalziel: Exportieren in diverse Sprache mithilfe einer Skriptsprache.

Alle Ziele konnten erfolgreich mit kleinen und größeren Hürden realisiert werden.

Bei dem Minimalziel existierte das Problem der Übergabe des Ausgewählten Elements, dies wurde mithilfe der QUserData des DragEvents realisiert.

Die ständige Aktualisierung der Daten während ein Objekt verändert wird stellte sich bei dem Standardziel als eine Hürde heraus, welche mithilfe des Signal-Slot und einer Hilfsklasse realisiert wurde.

Bei dem Maximalziel war die Implementierung einer Scriptengine sowie das Realisieren des Exportvorgang eine große Hürde. Die Scriptengine wurde mithilfe der Offiziellen C Lua Bibliothek gelöst, das zweite Problem wurde mithilfe von eigenen Lua Funktionen umgesetzt.

## Aufgabe 1.d

Mithilfe von Visual Studio 2017 wurde eine Übersicht aller Methoden und Variablen generiert. Nach der Übersicht wird eine Abhängigkeit einzelner Klassen gezeigt. Da jedes Objekt direkt oder indirekt von einem QObject abhängt wird die Abhängigkeit von QT wird nicht explizit gezeigt.

