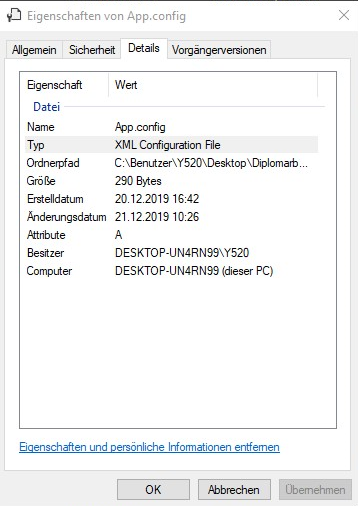
**Global Settings**

Die Global Settings werden in Visual Studio in der Konfigurationsdatei „app.config“ (Abbildung 3.1) abgespeichert und diese Datei sorgt dafür, dass man beim Start der Anwendung gewisse Parameter, welche immer gleich sind, automatisch setzt. Die „app.config“ Datei wird im XML-Format gehalten und ist ein zentraler Platz für Anwendungseinstellungen, Verbindungsdetails zu externen Diensten, Verbindungsstrings zu Datenbanken und weitere Details zur Ausführung oder zum Hosting der Anwendung. Die „app.config“ wird bei jedem ausführbaren Projekt automatisch hinzugefügt, diese kann man dann in der Projektmappe aufrufen und Einstellungen treffen. Die erste Einstellung, die das Dokument selbst trifft, ist die XML-Version der Datei, sowie die Zeichencodierung, welche man zum Speichern verwendet. Innerhalb der Konfigurationsdatei hat man verschiedene Möglichkeiten Einstellungen zu treffen.

**<supportedRuntime/>** Abbildung 4.1

Dies gibt an, welche „Common Language Runtime-Version“ und optional die .NET Framework Version die Anwendung unterstützt.

**<appSettings/>**

Hier kann man einen Schlüssel definieren der einen bestimmten Zeichenwert zugewiesen bekommt, auf den man durch diesen Schlüssel zugreifen kann. Diese Werte werden in der „app.config“ Datei gespeichert und werden somit zu Standardeinstellungen. In unserem Fall (Abbildung 4.2) haben wir zwei Schlüssel (empfindlichkeit, glättung).

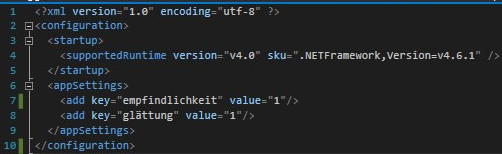
****

Abbildung 4.2

Bei Änderung einer Eigenschaft im Spielmenü unseres Projekts, wird der neue Wert dem Schlüssel zugewiesen und gleichzeitig überschrieben und gespeichert. Somit wird gewährleistet, dass nach Neustart des Programms die zuletzt eingestellten Werte verwendet werden. Jede Settingsform (ClientSettings, ServerSettings) in unserem Projekt greift beim Start der jeweiligen Forms auf diese Konfigurationsdatei über den ConfigurationManager (Abbildung 4.3) zu. Wenn man die Anwendung kompiliert, kopiert der Compiler die app.config-Datei in das Ausgabeverzeichnis und diese bekommt einen neuen Namen. In unserem Fall (Start.exe.Config). Wenn man in unserem Fall die Anwendung (start.exe) startet, wird die passende Konfigurationsdatei ebenfalls geladen.



Abbildung 4.3

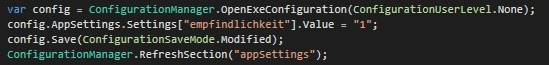
****

Abbildung 4.4

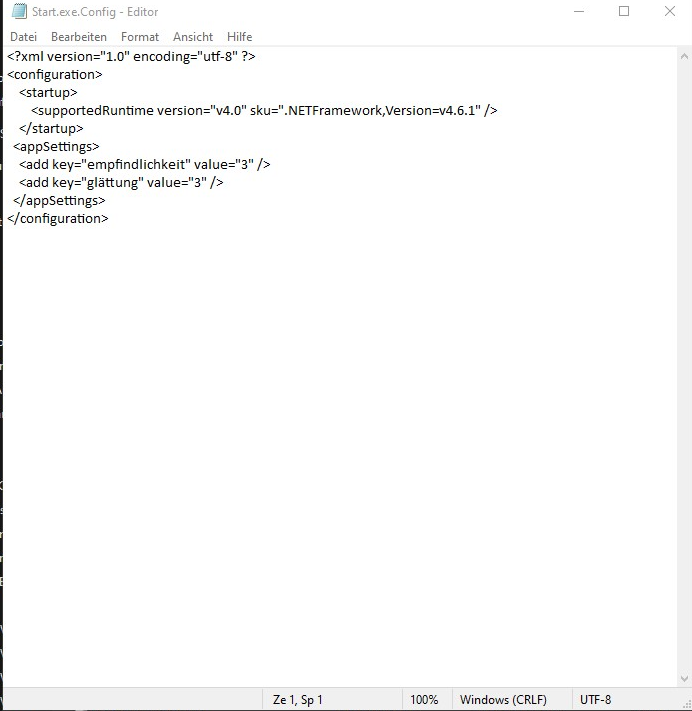
****

Abbildung 4.5