

Software Projekt - Excel als SQL Datenbank

Autoren: Rijad Žužo, Séverin Müller

Dozent: Ulrich Hauser

May 28, 2015



Inhaltsverzeichnis

1	Motivation	1
1.1	Ausgangssituation	1
1.2	Lösungsidee	1
2	Anforderungsliste	2
2.1	Muss-Anforderungen	2
2.2	Soll-Anforderung	2
2.3	Wunsch-Anforderung	2
3	Projektumgebung	3
3.1	Entwicklungsprozess	3
3.2	Ablauf	3
3.3	Programmiersprache	3
3.4	Verwaltungssystem	3
4	Projekt Planung	4
4.1	Zeitplan	4
4.2	Product Backlog	4
4.3	Sprints	4
4.4	Sprint Stand-up Meetings	4
4.5	Sprint Review	4
5	Ablauf der Entwicklung	5
5.1	Modellierung	5
5.1.1	Klassendiagramm	5
5.1.2	Aktivitäts-Diagramm	6
5.1.3	Sequenzdiagramm	7
5.2	GUI	8
5.2.1	Hauptfenster	8
5.2.2	Aufrufbare Fenster	8
5.3	Database Handling	8
5.4	Backup Management	9
5.5	Stunden Journal	9
6	Testen	9
7	Ziel	10
7.1	Resultat	10

Abbildungsverzeichnis

1	Zeitplanung	4
2	Der Produkt Backlog	4
3	Klassendiagramm	5
4	Aktivitätsdiagramm	6
5	Sequenzdiagramm	7
6	Hauptfenster	8
7	Aufrufbare Fenster	8

1 Motivation

Das Projekt wurde von uns in der Freizeit für den Bosnischen Club St. Gallen erarbeitet. Diese führen seit langem eine Excel-Liste für die Mitgliederverwaltung.

Für Sie war dies das einfachste Werkzeug, jedoch gab es immer wieder Probleme, wie zum Beispiel das ungewollte löschen ganzer Zeilen, oder das verrutschen in den Zeilen / Spalten. Wir wollten jedoch nicht eine komplett neue Datenbank erarbeiten, und entschieden uns, Ihnen ein Werkzeug für die Verwaltung der vorhandenen Excel Tabelle zur Verfügung zu stellen, welches eine einfache, intuitive GUI zur Verfügung stellt.

1.1 Ausgangssituation

Nebst den Problemen bei unaufmerksamer Bearbeitung, ist das Auslesen eines grösseren Excel Files deutlich langsamer als bei einer SQL Datenbank der gleichen bzw. einer vielfachen Grösse, deshalb ist es sinnvoll das Excel File in eine SQL Tabelle umzuwandeln und so zu Verarbeiten.

Die Konversion würde zusätzliche Software Kenntnisse erfordern, so griffen wir auf Vorhandene Office Werkzeuge zurück.

Mit unserer x2dB Applikation wurde das Problem gelöst. Die Umwandlung ist dank ODBC Connector unnötig und somit kann das Excel File bestehend bleiben.

1.2 Lösungsidee

Wir legten vor allem Wert auf ein simples User-Interface mit allen nötigen Funktionen. Die Software verarbeitet das Excel File im Hintergrund als SQL Tabelle und verbindet sich mit entsprechenden Treibern über ein File das via GUI (Dateibrowser) eingebunden werden kann. Das ermöglichen uns die Microsoft.Office.Core und Microsoft.Office.Excel.Interop Treiber.

Mit unserer Applikation hat der User einen begrenzten Einfluss auf das File und kann so weniger Schaden am File anrichten. Schaden können auch gleichzeitige Lese- / Speicherzugriffe auf ein Shared File auf einem Netzwerklaufwerk. In unserer Applikation dauert die Verbindung nur kurz, bis die Daten gelesen/geschrieben wurden und danach wird das File wieder freigegeben.

Für zusätzliche Datenintegrität wird jeweils ein Backup des Files erstellt und kann nötigen Falls zurück gespielt werden. Dies ist nicht die 'ultimative Lösung', aber wir konnten so die Sicherheit bei Veränderungen am Excel File erhöhen und trotzdem ein 'für jeden lesbares' Dateiformat weiter verwenden; dass zum Beispiel auch mit einem USB Stick übertragen und auf heimischen Computern (weiter-)bearbeitet werden kann.

2 Anforderungsliste

Um die Bedürfnisse der Verwalter dieser Excel Liste bestmöglich abdecken zu können, haben wir uns mit Ihnen zusammen gesetzt und die nachfolgenden Anforderungen definiert.

2.1 Muss-Anforderungen

Die Applikation muss:

M1: Das vorhandene Excel File als Datenbasis verwenden.

M2: Das Excel File lesen und beschreiben können.

M3: Nach dem Lesen bzw. Schreiben die Verbindung trennen.

M4: Bei Änderungen im File Backups erstellen.

2.2 Soll-Anforderung

Die Applikation soll:

S1: Leicht Bedienbar sein.

S2: Ein intuitives, einfaches User-Interface haben.

S3: Dem User das Handling des Excel Files abnehmen.

2.3 Wunsch-Anforderung

Die Applikation könnte:

W1: Eine Benutzungsanleitung haben.

W2: Mehrere Sprachen unterstützen.

W3: Dem User Hilfe anbieten.

Die Soll-Anforderungen wurden soweit erfüllt oder vernachlässigt wie es sich mit der Entwicklungszeit ergeben hat.

[W2] Einbau mehrerer Sprachen wäre ein sehr Zeitintensives verfahren und deshalb vernachlässigt.

[W1]-[W3] Benutzeranleitung und Hilfe anbieten wurde integriert und in einem Hilfe Untermenü eingebaut. Dies stellt das benötigte Format des Excel Files und wie es beschriftet sein müsste.

3 Projektumgebung

3.1 Entwicklungsprozess

Wir entschieden uns für den iterativen Scrum-Prozess. Es erschien uns Ideal, da wir jede Woche Dienstag mindestens zwei Lektionen Zeit hatten. Somit entschieden wir uns, dass die Sprints 7 Tage dauerten und nicht wie gewöhnlich 30. Wir teilten die Rollen wie folgt auf:

Scrum Master: Rijad Zuzo

Management: Séverin Müller

Product Owner & Entwicklung Rijad Zuzo, Séverin Müller

Customer Verein

3.2 Ablauf

Die Stand-up Meetings erfolgten immer am Anfang der Lektionen am Dienstag bei einem Kaffee oder bei schönem Wetter kurz draußen. Unter der Woche haben wir uns oft via Google Hangouts abgeglichen oder Fragen direkt mit der Freigabe eines Bildschirms besprochen. Dank des Online-Tools Trello.com konnten wir die Aufgabenpakete sogleich zuteilen und die Meetings waren dank unserer Vorbereitung sehr speditiv.

3.3 Programmiersprache

Da der Verein zuvor bereits Microsoft Excel verwendete und als Betriebssystem Microsoft Windows verwendete, lag es nahe, dass wir uns für die Windows Spezifische Programmiersprache C# entschieden. Dies in Kombination mit der IDE Visual Studio ermöglichte uns eine Praxis nahe und angenehme Entwicklung sowohl der Funktionalitäten, wie und auch des GUI's.

3.4 Verwaltungssystem

Die Auswahl des Verwaltungssystem war sehr einfach, da wir uns im Vorhinein geeinigt haben die Software als OpenSource entwickeln zu wollen. Somit haben wir uns für das OpenSource Git Projekt Verwaltungssystem entschieden und unser Code auf der Website www.GitHub.com gehostet.

Mit Hilfe dieser Collaboration Plattform war es uns möglich, gleichzeitig, in unabhängigen Codeteilen, zu arbeiten und den jeweiligen Stand zu synchronisieren

4 Projekt Planung

4.1 Zeitplan

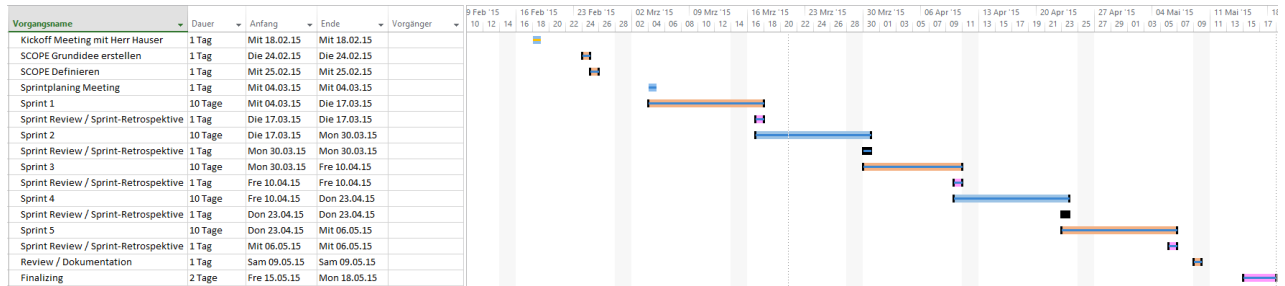


Bild 1: Zeitplanung

4.2 Product Backlog

Product Backlog				
Design Main User Interface	Build a file choosing option for selecting the wanted Excel sheet, with limiting extension (.xls, .xlsx)	Build a Help section for additional information on how the Excel sheet needs to be organized (Columns)	Implement onClick generation of a new Window with the detailed info of the clicked member in the table.	Implement the BackupManager, making a Backup before each change in the DB
Create a class that will handle all Database related actions.	1	Create a "new member" user interface	Build a member info (new payment) update section in the detailed information Window	Implement removal of members from the DB
Bind the references needed for the Excel database driver	Implement a permanent saving of the DB path so that there is no need of setting it every time the program opens. (Preference Saving)	Handle all necessary CRUD operations with the DB, display results in a table.	Implement the addition of a new Member in the Database via the New Member Window	
Implement a TabController with a DataGridView for Displaying the Data pulled from the DB				

Bild 2: Der Produkt Backlog

4.3 Sprints

4.4 Sprint Stand-up Meetings

4.5 Sprint Review

5 Ablauf der Entwicklung

5.1 Modellierung

Die erste Phase der Entwicklung war das Strukturieren der Software - eine sehr wichtige Phase. Wir wollten ein klares Design von Beginn an, dass uns die Weiterentwicklung ermöglicht.

Hier haben wir sehr viel Zeit aufgewendet, mit den Beteiligten sehr viele Szenarien durchgespielt und uns selbst einige Tage Zeit gelassen um über den Funktionsumfang und die Gimmicks eine klare Vorstellung zu erhalten. Die GUI sollte möglichst einfach gehalten sein, was aber nicht heisst, dass sich wenige Funktionen dahinter verbergen.

5.1.1 Klassendiagramm

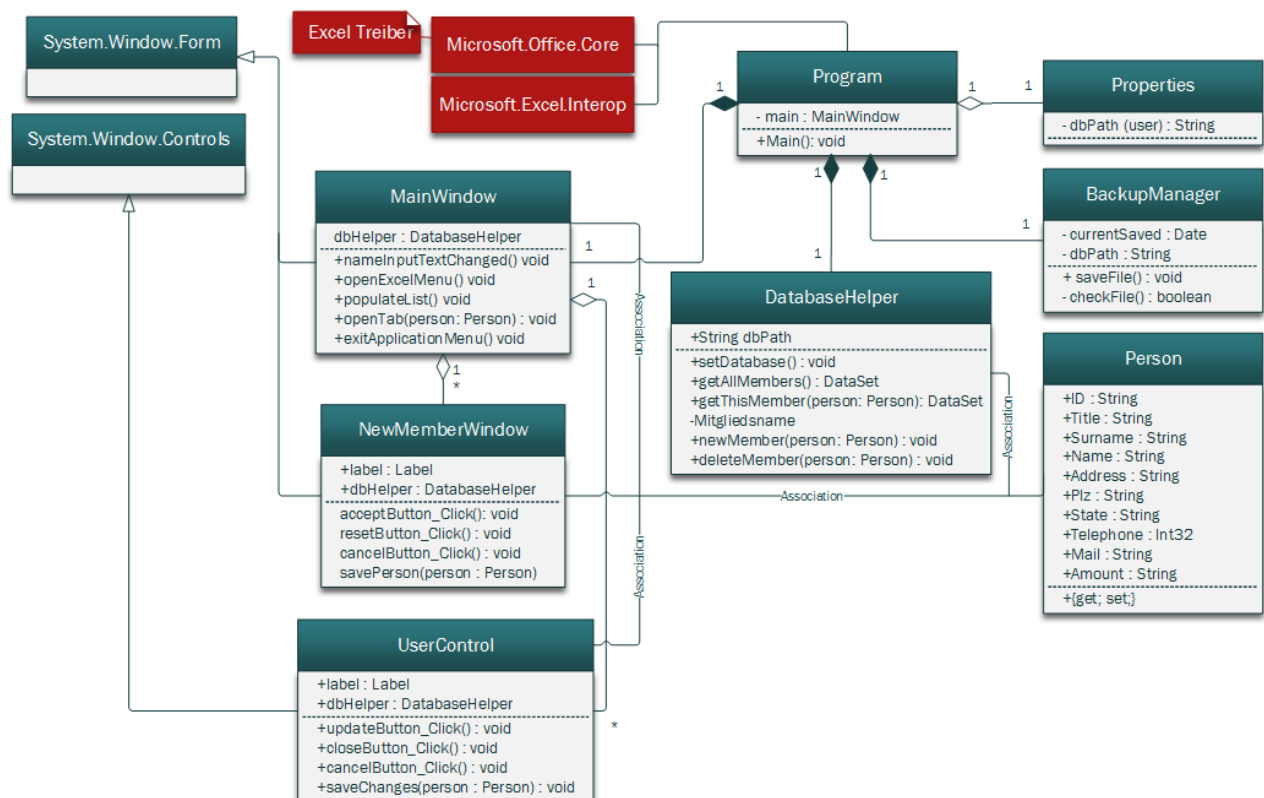


Bild 3: Klassendiagramm

Hier erkläre ich das was eigentlich dargestellt wurde im Klassendiagramm.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

5.1.2 Aktivitäts-Diagramm

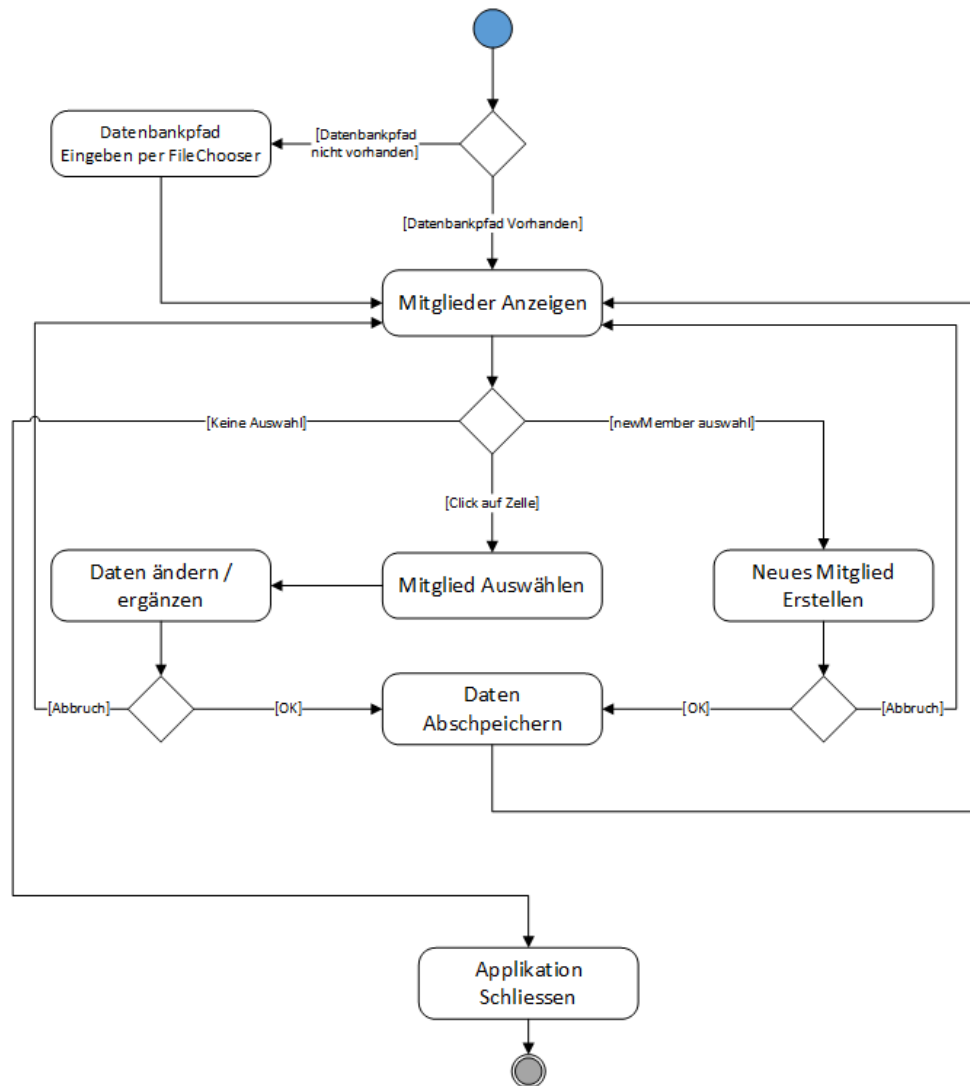


Bild 4: Aktivitätsdiagramm

Hier erkläre ich das was eigentlich dargestellt wurde im Aktivitadosdiagramm.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

5.1.3 Sequenzdiagramm

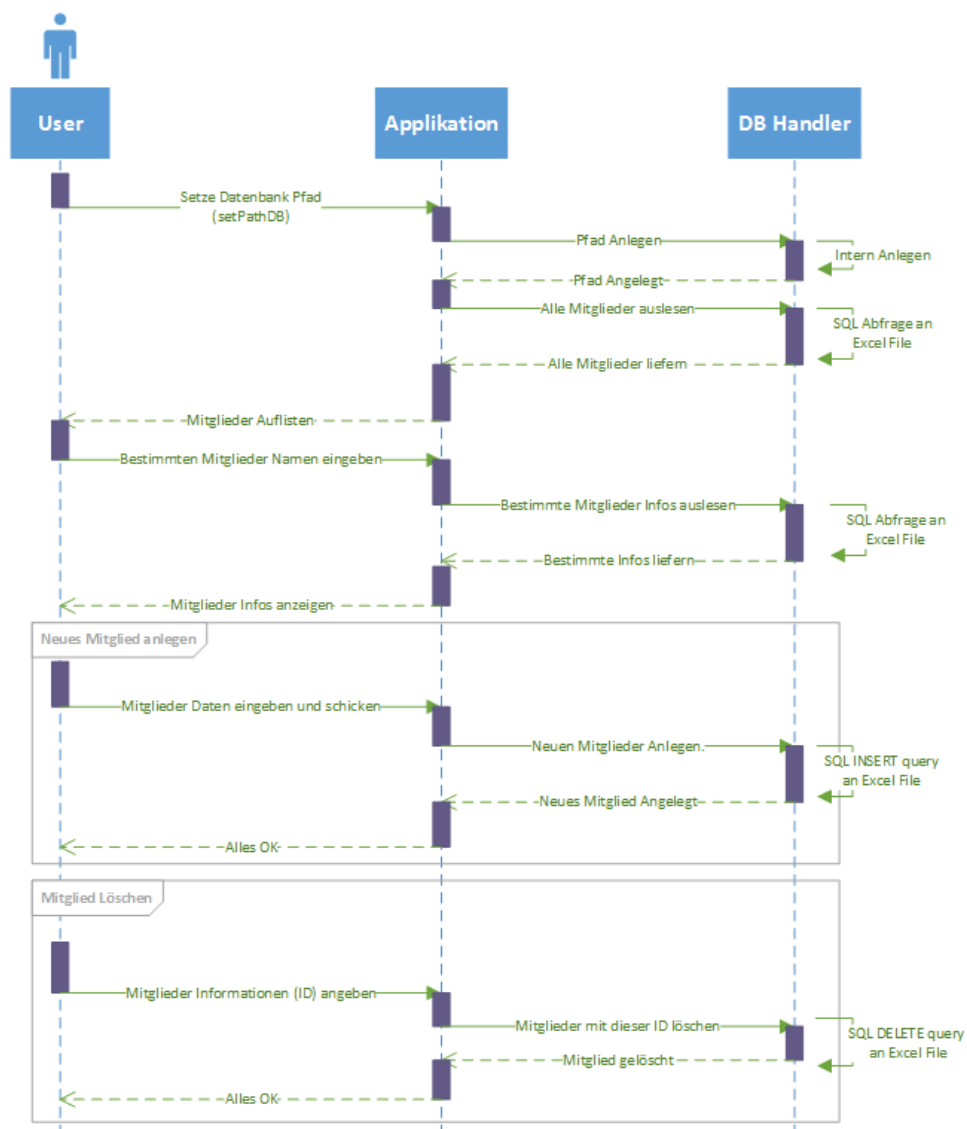


Bild 5: Sequenzdiagramm

Hier erkläre ich das was eigentlich dargestellt wurde im Sequenzeli.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

5.2 GUI

5.2.1 Hauptfenster

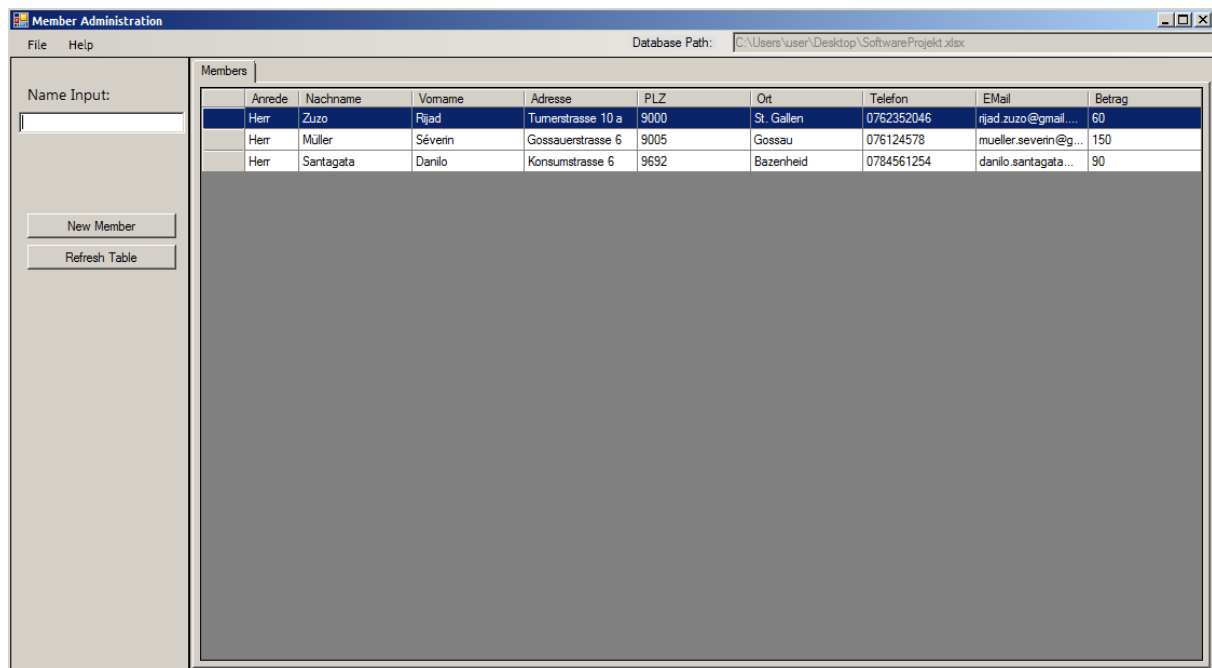


Bild 6: Hauptfenster

Das GUI wurde so einfach und intuitiv wie möglich gestaltet. Ein einfaches text-input Feld im linken teil lässt den Benutzer schnell den gesuchten Mitglied finden. Ein klick auf den Mitglied in der Tabellenansicht eröffnet ein neues Fenster wie im Bild 7.b) und zeigt alle nötigen Informationen die man auch leicht anpassen kann.

5.2.2 Aufrufbare Fenster

Title:
Surname:
Name:
Address:
Zip Code:
State:
Phone:
E-Mail:
Payment Amount:

Add Member Reset Cancel

(a) Neues Mitglied Formular

Title:
Surname:
Name:
Address:
Zip Code:
State:
Phone:
E-Mail:
Delete This Member
Paid:
Payment Amount:
New Payment
Update Information Cancel Save Changes Close

(b) Mitgliedsinformation Fenster

Bild 7: Aufrufbare Fenster

5.3 Database Handling

Schwierigkeiten bei den Referenzen, Voraussetzungen zur Verbindung usw.

5.4 Backup Management

Ansätze für ein effizientes Backuping :).

5.5 Stunden Journal

6 Testen

Wie beim Testen vorgegangen wurde!

7 Ziel

7.1 Resultat

7.2 Gelerntes