Dokumentation Studienarbeit

Email Programm

von

Daniel Glaser

Im Studiengang

Mobile Computing

an der Hochschule für angewandte Wissenschaften Hof

Inhaltsverzeichnis

[1 Einführung (max. 1 Seite) 3](#_Toc473232918)

[2 Beschreibung der Aufgabenstellung (1-2 Seiten) 4](#_Toc473232919)

[3 Implementierung: Beschreibung der Lösung (2-3 Seiten) 5](#_Toc473232920)

[3.1 Angaben zum Design der Oberfläche 5](#_Toc473232921)

[3.2 Angaben zur Umsetzung 5](#_Toc473232922)

[3.2.1 Beschreibung der Komponenten 5](#_Toc473232923)

[3.2.2 Verwendete Algorithmen, Bibliotheken oder Techniken 5](#_Toc473232924)

[3.2.3 Verwendeter fremder Quelltext 5](#_Toc473232925)

[4 Benutzerhandbuch (1-2 Seiten) 6](#_Toc473232926)

[4.1 Account anlegen 6](#_Toc473232927)

[4.2 Account bearbeiten 6](#_Toc473232928)

[4.3 Email schreiben 6](#_Toc473232929)

[4.4 Emails abholen 6](#_Toc473232930)

[5 Diskussion (max. 1 Seite) 7](#_Toc473232931)

[5.1 Zusammenfassung der Ergebnisse 7](#_Toc473232932)

[5.2 Aufgetretene Probleme 7](#_Toc473232933)

[5.2.1 WebBrowser ruckelt 7](#_Toc473232934)

[5.2.2 Testen des SMTP Server 7](#_Toc473232935)

[5.3 Weiterentwicklung des Programmes 7](#_Toc473232936)

[5.3.1 Weitere Email Funktionen 7](#_Toc473232937)

[5.3.2 Kalender 7](#_Toc473232938)

[6 Eidesstattliche Erklärung 7](#_Toc473232939)

# Einführung (max. 1 Seite)

Ein Email Programm gehört zu den grundlegendsten Programmen die es gibt, weil es so gut wieder jeder braucht. Da aber viele Email Programme mit Funktionen überladen sind und deswegen lange zum Starten brauchen oder eine unübersichtliche Oberfläche haben, habe ich mir in dieser Studienarbeit als Ziel genommen ein Email Programm zu programmieren, welches schnell startet und eine übersichtliche Oberfläche hat da es nicht mit Funktionen überladen ist. Es soll flüssig laufen und die wichtigsten Funktionen bieten.

Als Vorbild habe ich mir andere Email Programme wie z.B. Thunderbird aber auch Online-Email-Dienste wie z.B. Inbox genommen.

# Beschreibung der Aufgabenstellung (1-2 Seiten)

Es soll ein Email Programm entstehen das zuerst einmal die Grundfunktionen Email schreiben und Emails lesen ermöglichen soll. Dabei sollen HTML E-Mails richtig dargestellt werden.

Es soll möglich sein zu erkennen welche Emails bereits gelesen wurden und auch eine E-Mail wieder als ungelesen markieren können.

Die Emails sollen auch gelöscht werden können. Lokal aber auch auf dem Server.

Eine E-Mail soll man weiterleiten können. Auf eine E-Mail soll man antworten können. Auch soll es möglich sein allen zu antworten an die die E-Mail ging.

Bei Emails mit Anhang soll der Anhang auf zu machen sein. Auch soll man bei einer Email einen Anhang mit verschicken können.

Man soll die Möglichkeit geben mehrere Accounts anzulegen und zu verwalten.

Oder:

Ziel dieser Studienarbeit ist es ein Email Programm zu programmieren das folgende Funktionen bieten soll:

* Emails verschicken
* Emails empfangen
* Emails darstellen
* HTML Emails richtig darstellen
* Gelesene Emails als gelesen anzeigen
* Gelesene Emails als ungelesen markieren können
* Mehrere Accounts verwalten
* Email Antwort Funktion
* Email Weiterleiten Funktion
* Allen Antworten Funktion
* Anhänge anzeigen und aufmachen
* Anhänge senden

# Implementierung: Beschreibung der Lösung (2-3 Seiten)

## Angaben zum Design der Oberfläche

Beim Design wurde sich an Thunderbird orientiert aber trotzdem ein eigenes Design erstellt.

## Angaben zur Umsetzung

### Beschreibung der Komponenten

#### Email

Die Email-Klasse ist eine selbst erstellte Klasse die eine E-Mail repräsentiert.

#### EmailService

Der EmailService kümmert sich darum Emails zu senden und zu empfangen. Außerdem hat er noch Methoden mit dehnen der IMAP, POP3 und SMTP Server getestet werden können.

##### Emails senden

Zum Senden von Emails wird der SmtpClient von .Net verwendet.

##### Emails abholen

Für das Abholen von Emails wie entweder der ImapClient oder der Pop3Client des MailKit verwendet je nachdem welche Verbindung der Nutzer eingegeben hat.

##### Server testen

Zum Testen der Server werden die 3 Clients des MailKits verwendet um sich mit dem jeweiligen Server zu verbinden. Dabei wird überprüft ob es erfolgreich klappt oder nicht und dem Nutzer dann grafisch dargestellt.

### Verwendete Algorithmen, Bibliotheken oder Techniken

#### MailKit

Für das Abholen von Emails habe ich das MailKit verwendet. Es bietet einen ImapClient und Pop3Client. Den SmtpClient habe ich zum Testen genommen da er dafür einfach einzusetzen war als der SmtpClient von .Net oder ein TcpClient.

### Verwendeter fremder Quelltext

Dispatcher vom Herr Rill

DataServices kopiert von 11 vom Herr Rill aber abgeändert.

Emails abholen von (Link im Code)

# Benutzerhandbuch (1-2 Seiten)

Der Benutzer startet das Programm.

Als erstes sollte der Benutzer seinen Account einrichten. Dazu geht er in der Menüleiste auf Optionen und dort auf Accounts. Dort kann er mit dem Button „Hinzufügen“ einen neuen Account anlegen.

## Account anlegen

Zum Anlegen eines Accounts muss der Nutzer folgende Informationen eingeben:

* Anzeigename
* Username
* Email
* Passwort
* IMAP/POP3-Server
* IMAP/POP3-Port
* IMAP oder POP3
* SMTP-Server
* SMTP-Port

Zum Testen ob die eingegeben Server und Ports richtig sind stehen zwei Buttons zu Verfügung. Wenn diese grün werden sind die eingebenden Server und Port richtig. Sonst werden sie rot, wenn sie falsch sind.

## Account bearbeiten

Zum Bearbeiten eines Accounts muss der Benutzer in der Menüleiste auf Optionen und dort auf Accounts gehen. Im Optionenfenster wählt er aus der Dropdown-Liste den zu bearbeitenden Account aus und klickt auf den Button „Bearbeiten“. Dort kann er den Account bearbeiten wie unter „Account anlegen“ beschrieben.

## Email schreiben

Um eine E-Mail zu schreiben klickt der Benutzer im Hauptfenster auf den Button „Email schreiben“. Er kann auch über den Punkt Email und dann Email schreiben in der Menüleiste dorthin navigieren.

Es öffnet sich ein Fenster in dem der Benutzer die Sendeadresse auswählen kann aus den Accounts die er angelegt hat. Im nächsten Feld kann er die Empfangsadresse/n eingeben an die die E-Mail gehen soll. Und in dem großen Eingabefeld kann er die Nachricht eingeben die er schicken möchte.

## Emails abholen

Dazu klickt der Benutzer im Hauptfenster auf „Emails abholen“. Dann werden im Hintergrund die Emails abgeholt und im Hauptfenster aufgelistet.

o Anleitung für den (unbedarften) Benutzer

# Diskussion (max. 1 Seite)

## Zusammenfassung der Ergebnisse

Die Grundfunktionen wie Senden und Empfangen von Text Emails ist fertig. Die Anzeige von HTML-Emails ist durch den WebBrowser von CSharp umgesetzt worden. Dieser läuft leider bei der Größenänderung des Fensters nicht flüssig.

Es ist möglich mehrere Accounts anzulegen und zu bearbeiten.

## Aufgetretene Probleme

### WebBrowser ruckelt

Der WebBrowser ruckelt bei der Änderung der Größe des Fensters.

### Testen des SMTP Server

Das Testen des SMTP Server funktioniert noch nicht ganz so wie gewollt.

## Weiterentwicklung des Programmes

### Weitere Email Funktionen

Als erstes könnte das Programm noch mit weitere Email Funktionen erweitert werden. Dazu zählen zum Beispiel:

* Senden von HTML Emails

### Kalender

Wenn alles was mit Email zu tun hat implementiert wurde könnte das Programm noch um einen Kalender erweitert werden für Termine. Dieses könnte dann auch noch mit Cloud-Diensten wie z.B. dem Google Kalender synchronisiert werden.

# Eidesstattliche Erklärung

Ich versichere, dass ich die Studienarbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe, und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat. Alle Ausführungen der Arbeit, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.