Dokumentation

Julia Sorrentino

Inhalt

[Informieren 3](#_Toc92109135)

[Erste Aufgabe: 3](#_Toc92109136)

[Zweite-/Hauptaufgabe: 3](#_Toc92109137)

[Planen 3](#_Toc92109138)

[Realisieren 4](#_Toc92109139)

[Kontrollieren 5](#_Toc92109140)

[Auswerten 5](#_Toc92109141)

[Abbildungsverzeichnis 5](#_Toc92109142)

# Informieren

## Erste Aufgabe:

Der Auftrag ist es eine Applikation zu machen, bei der es ein Home-layout gibt, dort wird es zwei Buttons haben. Der erste Button wird auf die erste Unterseite mit einem Webview geleitet. Im Webview wird es die Google-Webseite haben.

Der zweite Button führt auf eine zweite Unterseite, ebenfalls mit einer Webview, diese hingegen führt zur Siemens Webseite ([Siemens](https://www.wtv676.siemens-info.com/ev16444144)). Alle drei Seiten sollten benutzerfreundlich sein, und somit auch einfach zu bedienen.

Dies habe ich alles gemacht, mit dem einzigen Problem, dass die Siemens Seite sich automatisch, viel zu gross skalierte.

## Zweite-/Hauptaufgabe:

Dies ist der Auftrag auf die sich diese Dokumentation bezieht. Bei der zweiten Aufgabe musste ich wieder eine Applikation machen, bei der man aber direkt nur eine Webview sieht, darin befindet sich die Siemens Webseite ([Siemens](https://www.wtv676.siemens-info.com/ev16444144)).

Der wichtigste Teil dabei, ist dass die Webseite sich automatisch dem Handy Layout anpasst, ohne, dass der Benutzer selbst das Layout vergrössern/verkleinern muss.

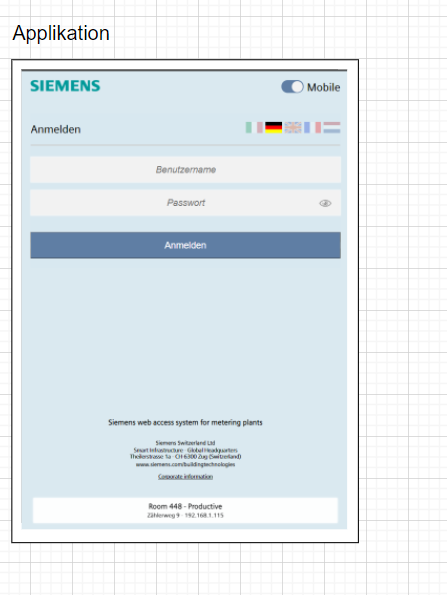
# Planen

Als erstes habe ich eine Skizze gemacht, damit ich visuell sehe wie das Layout sein sollte, damit die Siemens Webseite richtig in die Webview passt. Ich habe noch ein paar Testfälle aufgeschrieben, damit ich meine Applikation am Ende testen kann. Danach habe ich direkt mit der Realisierung angefangen.

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

2)Testfälle noch nicht ausgefüllt.



1)Skizze von der Applikation

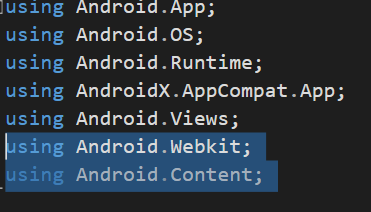
# Realisieren

Ich habe als erstes mit Visual Studio ein neues Android App (Xamarin) Project angefangen. Als View, habe ich die «Single View App» gewählt, da diese einfacher zu benutzen ist. Als Android Minimum Version habe ich Android 9.0 (Pie) gewählt.

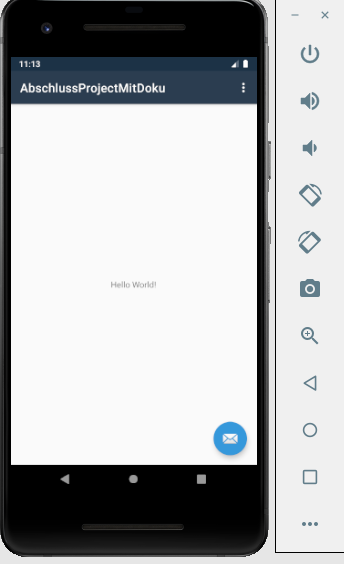
Bevor ich irgendetwas codiert habe, habe ich die leere Applikation gestartet, um sicher zu gehen, dass keine Fehler vorhanden sind.

Da alles geklappt hat habe ich mit dem Programmieren angefangen. Als erstes habe ich in der content\_main.xml Datei die Webview hinzugefügt, und dann im MainActivity.cs codiert. Damit die Applikation läuft, musste ich in den Properties, das Internet aktivieren.

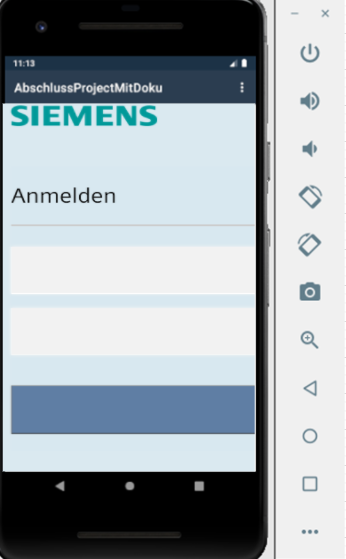
Da das Ganze funktioniert habe ich noch diese zwei Klassen hinzugefügt. Danach habe ich einen [Tutorial](https://docs.microsoft.com/en-us/xamarin/android/user-interface/controls/web-view) befolgt, damit ich keine Fehler mache. Zum Ende habe ich nochmals alles gestartet und es hat funktioniert.



4)Zwei Klassen



3)Leere Applikation



5)Applikation mit WebView

Nun habe ich mit dem Kontrollieren angefangen, dafür habe ich meine Testfälle benutzt und kontrolliert ob meine Applikation alle Testfälle besteht.

# Kontrollieren

Um alles zu kontrollieren habe ich die Testfälle benutzt, dort habe ich die Applikation gestartet, und alles durchgespielt. Die Testfälle basieren auf ein Smoke Testing, was so viel bedeutet, dass ich nur die Basisfunktionalitäten getestet habe.

Dabei hat alles einwandfrei funktioniert, der einzige Testfälle der nicht geklappt hat, war der vierte Testfall, dort ging es um die automatische Skalierung des Inhaltes von der WebView.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

6)Testfälle ausgefüllt

# Auswerten

Das Project konnte ich halbwegs abschliessen, da ich noch Probleme mit der Skalierung habe. Da ich das Layout der Webseite nicht automatisch anpassen kann, hatten wir die Idee es mit einer alternative zu machen. Die Idee ist, dass der Benutzer selbst diese mit einem Zoom Button das Layout anpassen kann.

# Abbildungsverzeichnis

[1)Skizze von der Applikation 3](#_Toc92103768)

[2) Testfälle noch nicht ausgefüllt. 3](#_Toc92103768)

[3) Leere Applikation 4](#_Toc92103768)

[4) Zwei Klassen 4](#_Toc92103768)

[5) Applikation mit WebView 4](#_Toc92103768)

[6) Testfälle ausgefüllt 5](#_Toc92103768)