



18. JULI 2021

# CHAT APPLIKATION SKETCH

INTRODUCTION TO INTERNET AND SECURITY

LI TING LUONG, MONIKA MULTANI



## Chat Applikation Sketch

---

### Einführung

---

Eigene Konzepte und Ideen können mithilfe von Skizzen sehr schnell vorgestellt werden. Skizzen dienen deshalb als ein wertvolles Kommunikationsmittel, um grobe Konzepte zu präsentieren. Zudem helfen sie dem Ersteller selbst eine komplexe Idee besser zu erfassen, da die Details ausgearbeitet werden und somit nur das wesentliche dargestellt wird.

Kann nun eine Skizze nur traditionell mit Stift und Papier erstellt werden? Nein, heutzutage gibt es Software-Programme, womit man Skizzen direkt digital anfertigen kann. In diesem Projekt geht es darum, solch ein Programm in eine bestehende Chat-Funktion zu implementieren, damit neben Text und Bildern auch Skizzen an einen Empfänger gesendet werden können.

### Hintergrund

---

WhatsApp ist eine weitverbreitete Messaging App, die den Austausch von Textnachrichten, Bild-, Video und Ton-Dateien sowie auch Dokumente und Standortinformationen zwischen zwei Personen oder Gruppen ermöglicht.<sup>1</sup> Zusätzlich bietet WhatsApp eine Funktion an, wo man direkt auf zuvor erstellte Bilder zeichnen kann. Dies erweist sich als hilfreich, wenn man bestimmte Stellen im Bild hervorheben will oder einfach seine künstlerische Freiheit ausleben will. Diese Zeichenfunktion haben wir uns als Inspiration für unser Projekt genommen.

Ziel war es in das bestehende BACnet Projekt diese Chaterweiterung zu implementieren. Dabei haben wir als Basis den Subjective Chat der Gruppe 03-subChat aus dem vorherigen Jahr genommen und so erweitert, dass man Skizzen direkt im Chat erstellen und versenden kann.

### Implementation, Konfiguration und Setup

---

Die Erweiterung der Chat Applikation haben wir direkt in Subjective Chat von der Gruppe 03-subChat im vorherigen Jahr implementiert, weshalb wir die Programmiersprache übernommen haben und den Code in Python weitergeschrieben haben. Wie die Gruppe 03-subChat haben auch wir für GUI Elemente mit der Python Bibliothek Tkinter gearbeitet.

Für das Erstellen von Skizzen haben wir im Chat einen Knopf "Sketch" über den "Send" Knopf implementiert, welches grün gefärbt ist, damit es farblich zum bereits bestehenden Grundgerüst passt. Sobald auf dem Knopf gedrückt wird, öffnet sich ein neues Fenster. Im Fenster wird ein leere Canvas mit weißem Hintergrund erstellt, die die Zeichenfläche darstellt. Anfangs ist die Farbe des Stifts schwarz, jedoch ist die Farbe mit der Farbpalette änderbar, die sich oberhalb der Canvas mit den anderen Werkzeugen befindet. Dabei ist wie bereits vorhin erwähnt die Farbpalette für Änderungen der Stiftfarbe zuständig und zudem noch in Kombination mit dem Farbeimer für Änderungen der Hintergrundfarbe. Will der Benutzer die Hintergrundfarbe ändern, muss dieser zuerst auf die gewünschte Farbe klicken und danach auf den Farbeimer. Des Weiteren gibt es noch einen Radiergummi, mit dem man Gezeichnetes wieder wegradieren kann, einen Regler für die Strich- und Radiergummistärke und ein Knopf "More colors".

---

<sup>1</sup> WhatsApp – Wikipedia: <https://de.wikipedia.org/wiki/WhatsApp>

Mit dem "More colors" Knopf wird beim Drücken ein Submenü geöffnet, wo man individuelle Farben mithilfe eines Color Pickers auswählen kann. Um gleichmässige Striche, sprich nicht verpixelte Linien, zu bekommen haben wir den Stift auf "smooth" gestellt. Ausserdem hilft noch gegen die Verpixelung, dass die Stiftspitze rund ist und nicht rechteckig.

Rechts neben den Werkzeugen sind noch drei weitere Knöpfe dargestellt. Der erste Knopf "Reset Canvas" dient dazu die Canvas zurückzustellen, d.h. dass der Inhalt gelöscht wird und der Hintergrund wieder auf weiss gesetzt wird. Durch einen Klick auf den Knopf "Save Image" wird das Bild im Ordner 'Images' gespeichert und durch den Knopf "Send Image" kann das Bild direkt an den Empfänger gesendet werden. Das Versenden eines Bildes wurde von der vorherigen Gruppe so implementiert, dass man den Pfad des Bilds manuell zuerst im Chat eingeben muss und danach auf "Send" klicken muss. Da wir diese Schritte als aufwendig empfunden haben, haben wir das Versenden vereinfacht. Und zwar wird bei uns der Pfad des Bilds automatisch als Chatnachricht versendet, sobald man auf den "Send" Knopf in der Sketch Applikation klickt.

## Resultate

---

Zu Beginn des Projekts haben wir einen Prototyp mit den grundlegenden Funktionen implementiert. Auf das Design und das Layout wurde bei dieser Version noch nicht geachtet. Jedoch gab es wie erwartet Probleme beim Prototyp.

Eines dieser Probleme war zum Beispiel, dass die gezeichnete Linie sehr pixelartig und nicht "smooth" erschien. Weiterhin war die Radiergummifarbe konstant auf weiss gestellt und hat sich nicht an die Hintergrundfarbe angepasst. Ungewünscht war auch dass die Farbpalette ausgeradiert werden konnte mit dem Radiergummi, da die Felder hier nur Rechtecke auf der Canvas waren. Anfangs haben wir für das Abspeichern der Skizze mit der Bibliothek Ghostscript gearbeitet, welche jedoch nur das Gezeichnete ohne Hintergrundfarbe abgespeichert hat.

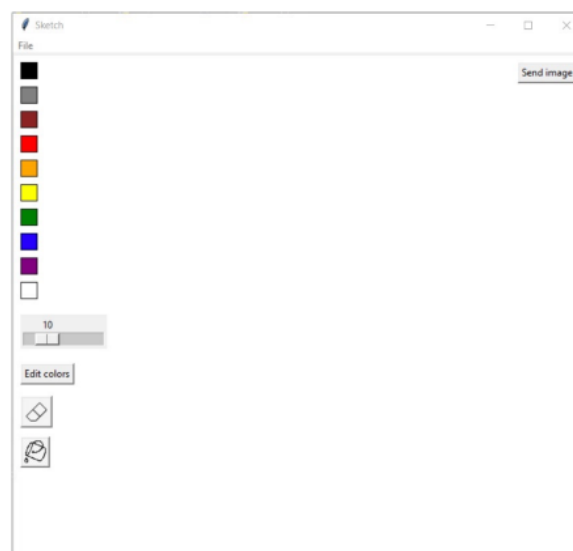


Abbildung 1: Prototyp

Beim Endresultat haben wir als erstes das Design und Layout geändert. Und zwar haben wir die Werkzeuge nicht direkt in der Canvas, sondern nun über der Canvas positioniert, weshalb nun das Problem mit dem Ausradieren der Farbpalette nicht mehr möglich ist. Die Farben der Palette haben wir trotzdem als Knöpfe belassen. Neu wird auch der Radiergummi an die Hintergrundfarbe angepasst. Um das pixelartige Zeichnen zu beheben, haben wir den Stift auf 'smooth' gestellt und die Stiftspitze als Kreis definiert anstatt als Rechteck. Das Abspeichern der Skizze erfolgt nun mithilfe von Tkinter anstelle von Ghostscript, wo die Hintergrundfarbe korrekt abgespeichert wird.

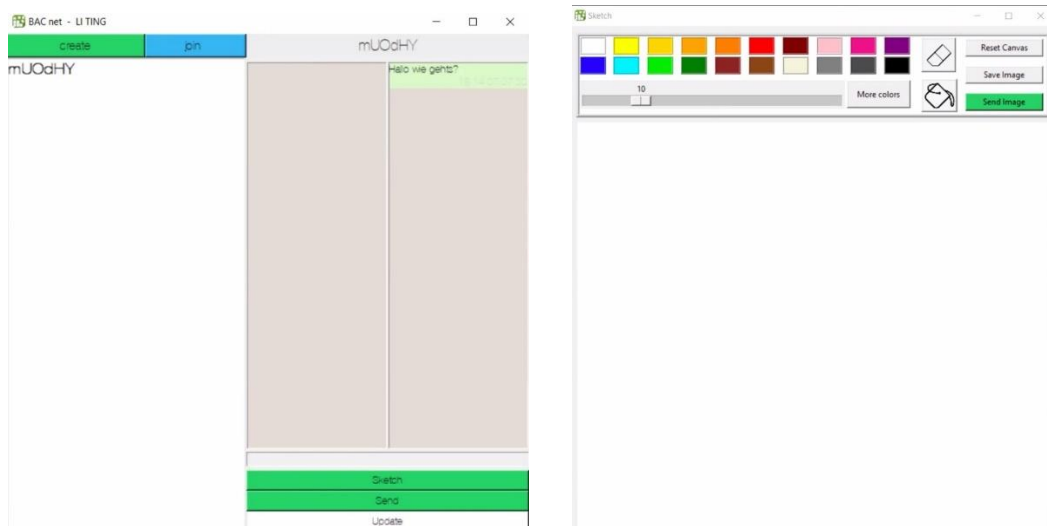


Abbildung 2: Neue Skizze erstellen

Um eine Skizze zu erstellen, muss der Knopf "Sketch" vom Benutzer angeklickt werden. Danach öffnet sich ein neues Fenster. Oben sind die verschiedenen Werkzeuge dargestellt und unten die leere Canvas mit weissem Hintergrund. Die Canvas ist hier die Fläche auf der gezeichnet wird.

Am Anfang ist die Farbe des Stifts schwarz. Der Benutzer kann auf die Farben oberhalb der Canvas klicken, um die Farbe des Stifts zu ändern. Will er jedoch eine andere Farbe auswählen, die nicht in der Farbpalette vorhanden ist, kann auf den Knopf "More colors" gedrückt werden. Falls der Benutzer etwas ausradieren möchte, kann dafür der Radiergummi verwendet werden oder man kann die Canvas in die Ursprungsform zurücksetzen mit einem Klick auf "Reset Canvas". Des Weiteren kann man die Hintergrundfarbe ändern, indem man zuerst auf den Farbeimer klickt und danach die gewünschte Farbe auswählt. Die Strichstärke des Stifts und des Radiergummis sind mit dem Regler unterhalb der Farbpalette änderbar. Am Ende kann die Skizze gespeichert oder direkt als Bild gesendet werden mithilfe der "Save Image" und "Send Image" Knöpfe.

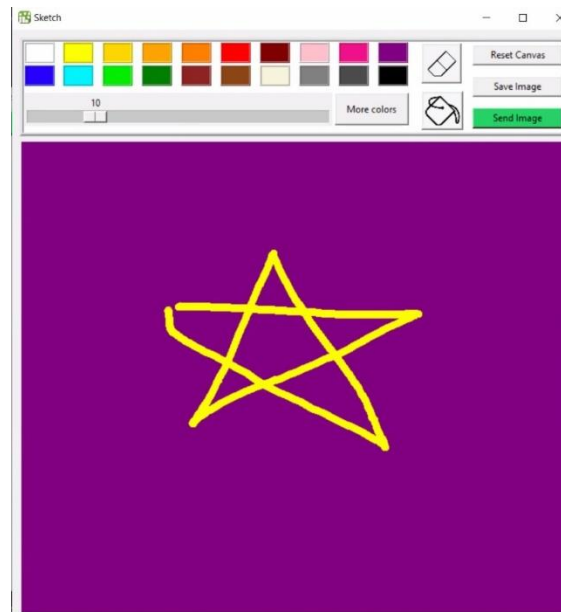


Abbildung 3: Empfangen einer Skizze

Der Empfänger muss auf die Nachricht "Click to open image" klicken, um das Bild zu öffnen.

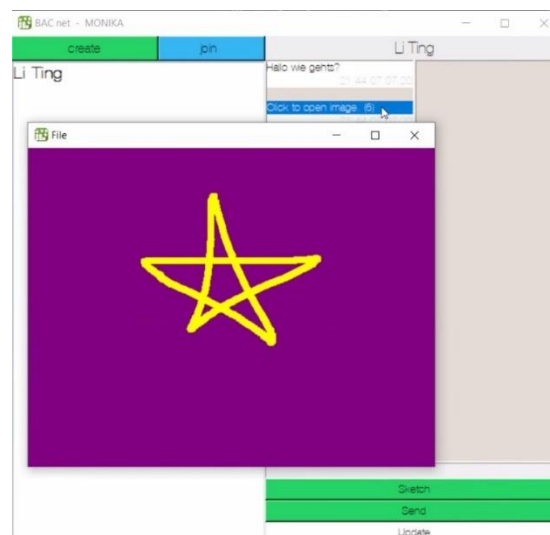


Abbildung 3: Empfangen einer Skizze

## Diskussion

Da wir ausserhalb dieser Vorlesung meistens zusammenarbeiten, jedoch wegen COVID-19 nur noch online Kontakt hatten, haben wir oft die WhatsApp Funktion mit dem man auf Bildern zeichnen kann selbst genutzt. Von daher kam uns die Idee eine Skizzen Funktion in den Subjective Chat zu integrieren. Vorteile, die für uns attraktiv sind, sind zum Beispiel das Hervorheben von Elementen mit dem integrierten Leuchtstift oder das vereinfachte Erklären von komplexen Konzepten, die üblicherweise als Textnachricht erklärt werden mussten. Ausserdem wird hierbei kein Papier verschwendet.

Unsere Überlegung war den Subjective Chat weiter an WhatsApp anzupassen. Es könnte eine Antwort- und Weiterleitungsfunktion mit Referenz auf den Absender implementiert werden. Auch könnte das Layout des Chats moderner gestaltet werden, indem man die Nachrichten als Sprechblasen dargestellt. Eine weitere Erweiterung wäre zum Beispiel das Versenden von Sprachnachrichten zu ermöglichen.

### Fazit

---

Das Ziel war es den Subjective Chat der Gruppe 3 aus dem Frühjahrssemester 2020 mit einer Sketch Applikation zu erweitern. Die Applikation wurde in Subjective Chat integriert. Wir haben im Chat Fenster einen Knopf «Sketch» erstellt. Beim Klicken dieses Knopfes wird ein Fenster für die Skizze mit den Werkzeugen wie die Farbpalette, Radiergummi und Farbeimer für die Änderung der Hintergrundfarbe und eine Zeichnungsoberfläche, sogenanntes Canvas geöffnet. Gefällt einem die Skizze nicht kann es zurückgestellt werden. Die Skizze kann als Bild durch einen Knopfclick gespeichert, sowie auch direkt gesendet werden. Der Empfänger erhält die Skizze auch als Bild und kann dieses öffnen.

### Lessons learned

---

Wir haben gelernt uns in einem grossen Projekt zurecht zu finden. Da wir unser Projekt in ein vorhandenes Projekt integrieren mussten, mussten wir uns erstmal einen Überblick über das Projekt verschaffen und den für uns relevanten Code der von vorherigen Studenten geschrieben wurde verstehen, um daran arbeiten zu können. Damit die Wartbarkeit und die Übersicht gewährt wird, haben wir die Darstellung und die Struktur vom Subjective Chat beibehalten, sowie unsere Chaterweiterung in Python implementiert. Neu dazu gelernt haben wir ausserdem, wie wir mit der Bibliothek Tkinter arbeiten können, um GUI Elemente darstellen zu können. Und als letztes wurde uns die Wichtigkeit eines Prototyps vor Augen geführt, weil wir damit viele Fehler frühzeitig erkennen konnten, bevor wir mit dem Integrieren begonnen haben.

### References - Appendix

---

Das Project BACnet von den vorherigen Studenten wurde uns zur Verfügung gestellt. Wir haben hauptsächlich den Code von der Gruppe 03-subChat verwendet, da wir unsere Applikation in den SubjectiveChat integriert haben, weshalb auch der Code in Python implementiert wurde. Für die Sketch Applikation haben wir die Bibliothek Tkinter verwendet und für die Definition von WhatsApp haben wir uns an Wikipedia (<https://de.wikipedia.org/wiki/WhatsApp>) orientiert.