



**«helping hands»**

# **«NextGen Dargebotene Hand»**

---

Technische Informationen für die Jury



## 1 Zugänge

---

### 1.1 Aktueller Stand des Sourcecodes

---

- <https://github.com/search?q=topic%3Ahelpinghands+org%3Abaernhaeckt> [Organisation]
- <https://github.com/baernhaeckt/Backend2023> [Backend Source]
- <https://github.com/baernhaeckt/MachineLearner2023> [Avatar Generator, Emotion Detection]
- <https://github.com/baernhaeckt/Frontend2023> [Frontend Source]
- <https://github.com/baernhaeckt/Misc2023> [Documentation, Presentation, Misc]

### 1.2 Production

---

- <https://lively-island-07bd13b03.3.azurestaticapps.net/> [Frontend]
- <https://bh2023.azurewebsites.net/swagger> [Backend – Swagger UI (Nur Test Controller)]
- <https://ml2023.azurewebsites.net/swagger> [Machine Learning Swagger]

## 2 Ausgangslage

---

Aktuell erreicht die Dargebotene Hand vorwiegend ältere Menschen. Studien zeigen, dass insbesondere auch junge Menschen immer öfter an ihre Belastungsgrenzen stossen und an Einsamkeit leiden, weshalb sie empathische Unterstützung und ein Gegenüber suchen. Die Challenge dreht sich darum das Angebot auf junge Menschen auszurichten und es für diese attraktiv zu gestalten. Die folgenden Überlegungen haben wir bei unseren Überlegungen ins Zentrum gestellt.

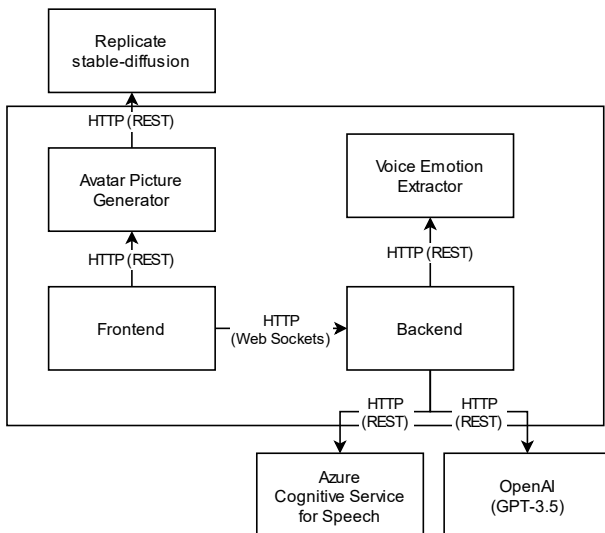
- Modern und Jung
- Beziehungsaufbau & Vertrauen
- Präventiv anstatt reaktiv
- Anonymität
- Interventionen skalieren
- Betrug verhindern

Interaktive Chats und Sprachnachrichten auf Smartphones dominieren die Art der Kommunikation bei jungen Menschen. Unsere Lösung «Helpy» trägt diesem Umstand Rechnung, indem wir eine Plattformunabhängige App implementiert haben. Diese lässt sich auf jedem Smartphone installieren. Die App imitiert die häufig verwendeten Kommunikationsapps wie WhatsApp, Telegram etc. Der Benutzer erstellt sich einen persönlichen Avatar, mit dem er kommuniziert. Der Avatar imitiert eine in Psychologie versierte und empathische Person. Die Kommunikation erfolgt mittels Sprachnachrichten und Chat. Die Installation und Verwendung der App finden statt, bevor der Benutzer Probleme hat. Mittels Marketingkampagnen an Berufsschulen und über Partnerschaften mit Lehrbetrieben und Hochschulen wird die App bekannt gemacht. Die fotorealistischen Avatare entsprechen nicht nur dem Geschmack des Benutzers, sondern sind auch einzigartig. Die Kombination aus Bild und Stimme bzw. die charakteristische Ausdrucksweise ermöglicht einen Vertrauensaufbau über die Zeit. Helpy, verkörpert durch den persönlichen Avatar, steht jederzeit mit Rat und Tat zur Seite. Helpy erkennt die Emotionen des Benutzers über dessen Stimmlage, die Konversationen werden gespeichert und Helpy lernt so den Benutzer kennen. Zum Vertrauensaufbau trägt auch die Tatsache bei, dass Helpy sich für seinen Benutzer interessiert. Der Avatar meldet sich in regelmässigen Abständen beim Benutzer und stellt ihm Fragen, die aus den vergangenen Gesprächen abgeleitet werden. Die Interaktion ist anonym und die gespeicherten Unterhaltungen sind an das Gerät des Benutzers gebunden, nach einer gewissen Zeit der Inaktivität werden diese automatisch gelöscht. Erkennt Helpy durch Abweichungen in der Stimmungslage und im Inhalt, dass ein Benutzer Hilfe von einem Menschen benötigt, so wird der Hand-Over Prozess gestartet – Betrug führt in der Regel nicht zum Hand-Over. Mit Erlaubnis des Benutzers wird jemand von der Dargebotenen Hand involviert. Diese Person erhält eine Zusammenfassung der Interaktion vom Benutzer und dem Avatar. Ab da kommuniziert der Benutzer immer noch mit dem Avatar, jedoch übernimmt im Hintergrund ein Mensch.

### 3 Technischer Aufbau

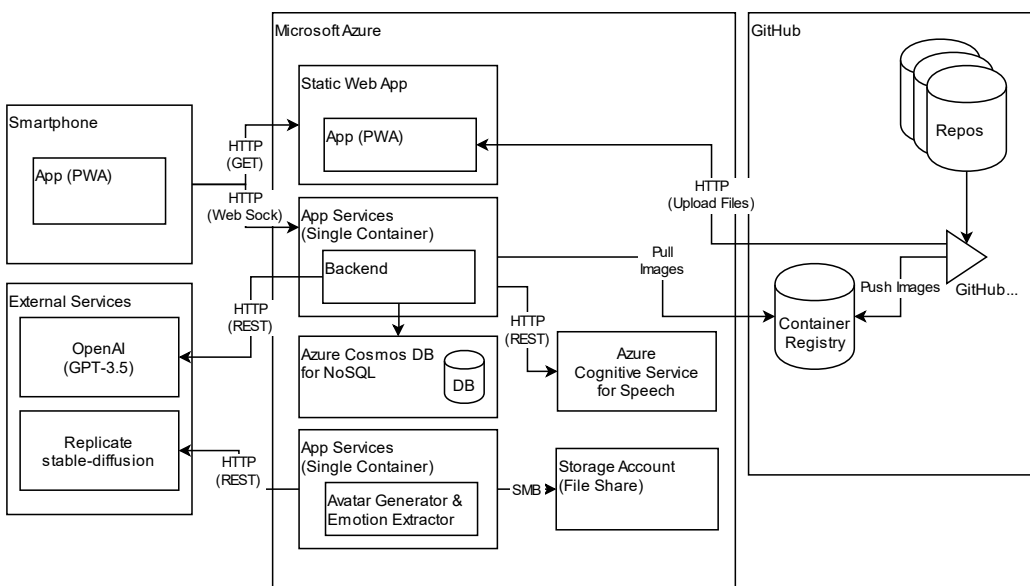
#### 3.1 Bausteinsicht

Die Lösung verwendet Vue.js im Frontend, sowie C# (Backend mit Persistenz) und Python (Avatar Picture Generator, Voice Emotion Extractor) im Backend. Frontend und Backend kommunizieren mittels HTTP unter Verwendung von REST und Websockets (SignalR). Es kommen drei externe Services zum Einsatz. Zur Generierung der Avatare verwenden wir stable-diffusion von Replicate, für Speech-To-Text und Text-To-Speech kommt Azure Cognitive Service for Speech zum Einsatz. Für die Generierung von Antworten und Fragen wird GPT-3.5 von OpenAI verwendet.



#### 3.2 Verteilungssicht

Die Lösung wird zur Laufzeit auf Microsoft Azure gehostet. Dabei kommen App Services für Single Container und Static Web App zum Einsatz. Als schneller Cache wird ein File Share über einen Azure Storage Account zur Verfügung gestellt. Zur Entwicklungszeit wird die Lösung auf GitHub gehostet, CI/CD ist mittels GitHub Actions implementiert und alle Teile werden kontinuierlich ohne Downtime deployt. Die Docker Images werden auf der GitHub Container Registry abgelegt.



#### 4 Implementation

---

Durch den Einsatz von modernem Machine Learning zum Generieren eines fotorealistischen Avatars, zum Übersetzen von und zu Mundartsprachnachrichten, Emotionserkennung anhand der Stimmlage und zur Interaktion (Fragen & Antworten) ist es uns in kurzer Zeit gelungen eine skalierbare Lösung mit guten Benutzererlebnis zu schaffen. Die Lösung ist auf Azure gehostet und alle Komponenten sind einsatzbereit. Was nicht implementiert wurde ist angedacht und die bereits implementierten Einzelteile Bestätigen, dass alles wie geplant implementiert werden kann.

#### 5 Abgrenzung / Offene Punkte









---

Grundsätzlich haben wir uns bei der Implementierung der Lösung auf die wichtigsten Elemente der «User Journey» konzentriert. Gleichzeitig wollten wir verifizieren, dass die kritischen Elemente wie gewünscht funktionieren. Dabei haben wir folgende Dinge nur konzeptuell angedacht und noch nicht implementiert.

- Benutzerwiedererkennung: Jede Verbindung zum Backend wird als neuer Benutzer erkannt. So ist es möglich eine Unterhaltung mit mehreren Elementen zu führen, jedoch wird beim Refresh eine neue Session gestartet. Durch das Übergeben einer auf dem Gerät gespeicherten Id lässt sich das einfach nachrüsten.
- Verschlüsselung: Die Unterhaltungen werden nach etwa 1/3 Jahr Inaktivität gelöscht, sie sind jedoch nicht verschlüsselt. In Zukunft sollte auf dem Client ein Schlüssel generiert und gespeichert werden. Die Unterhaltung kann dann jeweils nur mittels Zustimmung vom Benutzer entschlüsselt werden.
- Einbeziehen von vorhandenen Unterhaltungen: Helpy bezieht sich aktuell nicht auf die Vergangenen Unterhaltungen, wenn proaktiv Fragen gestellt werden. Das wäre jedoch mit dem bereits eingebundenen GPT-3.5 von Open API kein Problem.
- Hand-Over: Der Hand-Over ist nicht implementiert. Das bedeutet, dass es keinen Trigger gibt, der das anstösst, dass das Summary für den Menschen nicht erstellt wird und dass auch keine Kommunikation vom Menschen durch den Avatar möglich ist. Mit den aktuell verwendeten Diensten lässt sich das alles mit wenig Aufwand implementieren.
- Modelle haben ein hohes Bias: Aufgrund der geringen Zeit konnten wir unser Modell zur Emotionserkennung nicht im ausreichenden Mass trainieren.

## 6 Anhang

### 6.1 Azure Ressource Group

<input type="checkbox"/> Name ↑↓	Type ↑↓	Location ↑↓
<input type="checkbox"/>  ml2023	App Service	West Europe
<input type="checkbox"/>  helpinghands2023	Storage account	West Europe
<input type="checkbox"/>  fe2023	Static Web App	West Europe
<input type="checkbox"/>  bh2023	Azure AI services multi-service ac...	West Europe
<input type="checkbox"/>  bh2023	Azure Cosmos DB account	West Europe
<input type="checkbox"/>  bh2023	App Service	West Europe
<input type="checkbox"/>  bh2023	Application Insights	West Europe
<input type="checkbox"/>  ASP-bh2023group-976d	App Service plan	West Europe

### 6.2 Abgespeicherte Unterhaltung

Es gibt eine Unterhaltung pro Benutzer, neue Nachrichten werden diesem Item hinzugefügt. Die TTL Zeit ist auf der Cosmos DB konfiguriert und bezieht sich auf «last modified».

```
{
  "id": "D4PwIYjgcYAKrkNr0Kninw",
  "Nickname": "Hans",
  "Messages": [
    {
      "TimeStamp": "2023-08-27T08:55:12.8305924Z",
      "Content": "Ich habe Angst vor der Zukunft.",
      "Kind": "user"
    },
    {
      "TimeStamp": "2023-08-27T08:55:14.7168608Z",
      "Content": "Es ist verständlich, dass du Ängste hast, aber es könnte hilfreich sein, herauszufinden, woher diese Angst kommt und sie dann schrittweise anzugehen, um sie zu überwinden.\n",
      "Kind": "bot"
    }
  ],
  "_rid": "7-EnAK2F20YsAAAAAAAAA==",
  "_self": "dbs/7-EnAA==/colls/7-EnAK2F20Y=/docs/7-EnAK2F20YsAAAAAAAAA==/",
  "_etag": "\"bc05c8c1-0000-0d00-0000-64eb0f720000\"",
  "_attachments": "attachments/",
  "_ts": 1693126514
}
```

### 6.3 Resultat Emotion Detection

---

```
{
  "predicted_classes": [ "neutral" ],
  "probabilities_max": [
    {
      "class": "anger",
      "probability": 50.00000596046448
    },
    {
      "class": "anxiety",
      "probability": 50.00000596046448
    },
    {
      "class": "boredom",
      "probability": 73.05106520652771
    },
    {
      "class": "disgust",
      "probability": 50.91985464096069
    },
    {
      "class": "happiness",
      "probability": 51.71198844909668
    },
    {
      "class": "neutral",
      "probability": 73.10585975646973
    },
    {
      "class": "sadness",
      "probability": 61.59616708755493
    }
  ]
}
```