

Doctors Helper

Pflichtenheft

|  |  |
| --- | --- |
| Projektname | Doctors Helper |
| Erstellungsdatum | 11.01.2014 |
| Letzte Änderung | 11.01.2014 |
| Version | 1.0 |

|  |  |
| --- | --- |
| Teammitglieder | |
| Alexander Schöffmann | Benedikt Starzengruber |
| Florian Stürzlinger | Lukas Wickenhauser |

# Inhaltsverzeichnis

[Inhaltsverzeichnis 2](#_Toc377660383)

[Ausgangslage und Zielsetzung 3](#_Toc377660384)

[Aktuelle Lage, Ist Zustand 3](#_Toc377660385)

[Zielsetzung 3](#_Toc377660386)

[Anforderungen 3](#_Toc377660387)

[Softwareanforderungen 3](#_Toc377660388)

[Plattformanforderungen 3](#_Toc377660389)

[Diagramme 4](#_Toc377660390)

[Use Case Diagramm 4](#_Toc377660391)

[Funktionsweise 6](#_Toc377660392)

[Allgemeine Übersicht 6](#_Toc377660393)

[Benutzeroberfläche 7](#_Toc377660394)

[Mengengerüst 7](#_Toc377660395)

[Personenübersicht 7](#_Toc377660396)

[Entwicklerteam 7](#_Toc377660397)

[Betreuer 8](#_Toc377660398)

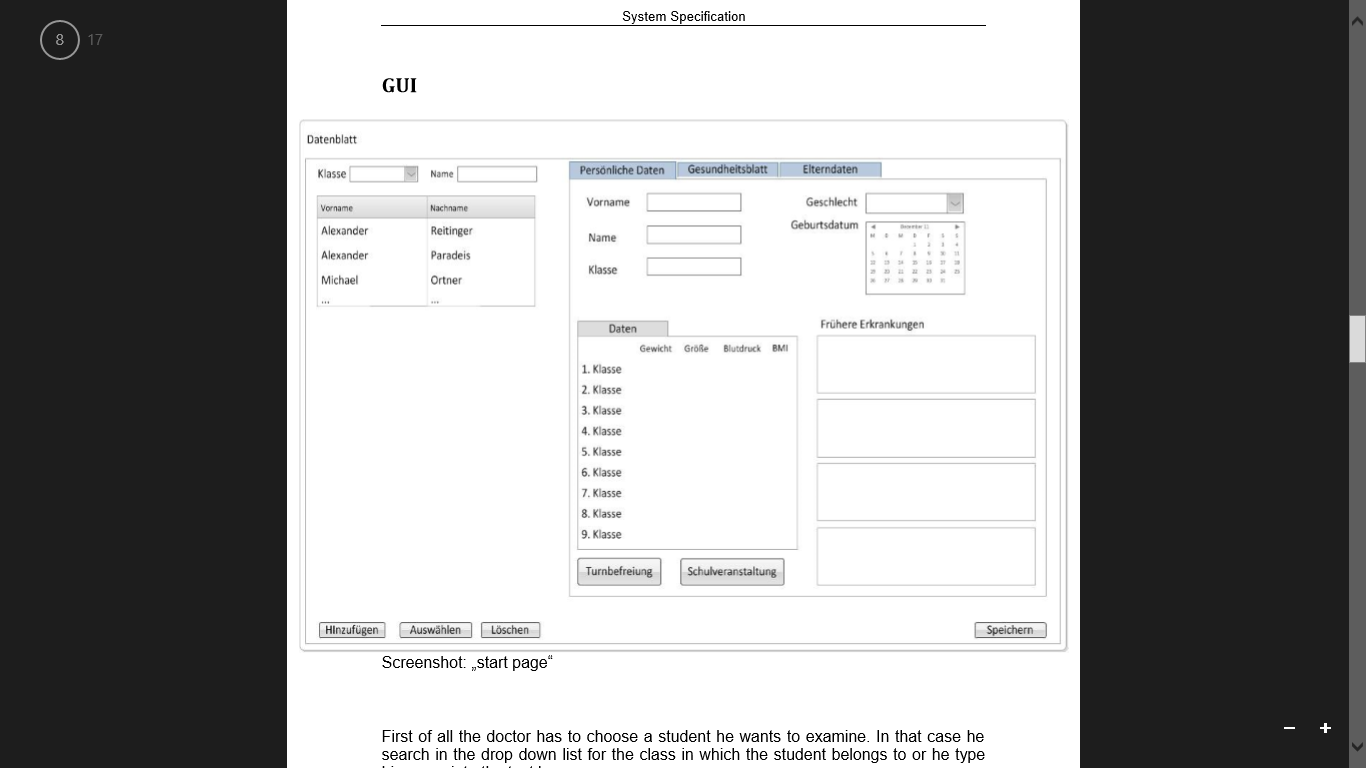
# Ausgangslage und Zielsetzung

## Aktuelle Lage, Ist Zustand

Ein Schularzt hat u.a. die Aufgabe, Schüler der Schule zu untersuchen.

**Problem**

Dem Schularzt fehlt eine Software, die seinen Anforderungen und Ansprüchen an Funktionalität und Design gerecht wird. Die bürokratische Arbeit wird nur unzureichend vom derzeitigen Programm übernommen.



## 

## Zielsetzung

Dem Schularzt soll eine Software zur Verfügung gestellt werden, die die Schüler mittels eines Mausklicks in Listen einliest. Somit hat er schneller Zugriff auf die Daten der Schüler.  
  
Die Eingabe neuer Schülerdaten soll teils automatisch funktionieren, das Ändern der Daten und Erstellen von Elternbriefen und Befreiungen soll einfach und schnell vollziehbar sein.

Ziel ist es, den Schularzt zu unterstützen, damit er sich besser auf den Patienten konzentrieren kann und seine bürokratische Arbeit schneller erledigen kann.

# Anforderungen

## Softwareanforderungen

* Schnelles Laden von Schülern, Klasse, Events ( < 1 Sekunde)
* JUnit Tests prüfen Programm
* Keine auftretenden Exceptions für den User

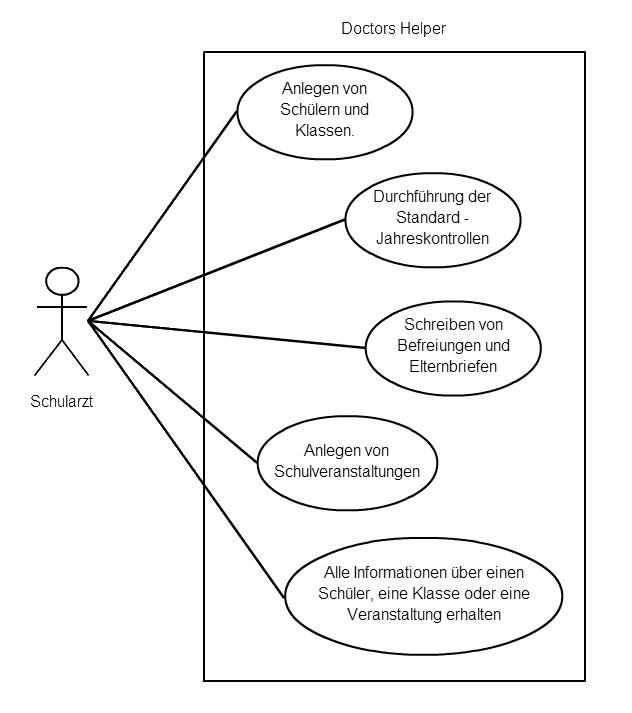
## Plattformanforderungen

Das Programm soll auf allen Betriebssystemen von Windows laufen. Da es in Java geschrieben wird, dürfte es aber auch auf allen sonstigen Plattformen funktionieren.

Das Programm soll grundsätzlich mittels JavaFX 2 programmiert werden. Die hierbei verwendete IDE ist Netbeans. Zur Speicherung der Daten wird eine Devby – Datenbank aufgesetzt.

# Diagramme

## Use Case Diagramm



## C:\Users\Bene\Documents\DoctorsHelper\EntityRelationshipDiagramm.jpgEntity-Relationship-Diagramm (ERD)

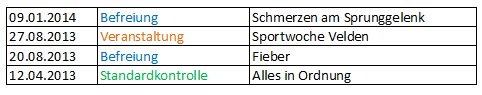
# 

# Funktionsweise

## Allgemeine Übersicht

Grundsätzlich gibt es folgende Elemente

* Schüler
* Klasse
* Events

Jedes dieser Elemente bekommt ein eigenes Profil in dem alle Informationen dazu übersichtlich angezeigt werden. Außerdem beinhaltet jedes Element eine Timeline, die in chronologisch sortierter Reihenfolge alle Events, die in Verbindung mit diesem Element stehen, anzeigt. Durch einen Filter kann die Anzeige den Wünschen entsprechend umgeändert werden (z.B. nur Befreiungen)  
Exemplarisches, designtechnisch nicht ausgearbeitetes Beispiel für eine Timeline

**Schüler**

Die Übersicht des Schülers besteht aus 2 Registerblättern  
Allgemein: Name, Klasse, Geburtsdatum, Geschlecht, Timeline, Foto, eventuell kleine gesundheitstechnische Daten (BMI-Verlauf etc.), Schreiben/Lesen von Befreiungen und Elternbriefen.  
Gesundheitsakte: Zeigt alle gesundheitlichen Daten an, eventuell auch eine Art Timeline, die Veränderungen anzeigt.

**Klasse**

Hier werden allgemeine Daten über die Klasse, wie Name, Anzahl der Schüler, etc. gespeichert. Außerdem wird eine Liste mit allen Schülern angezeigt und eine Timeline wird eingebaut.

**Events**

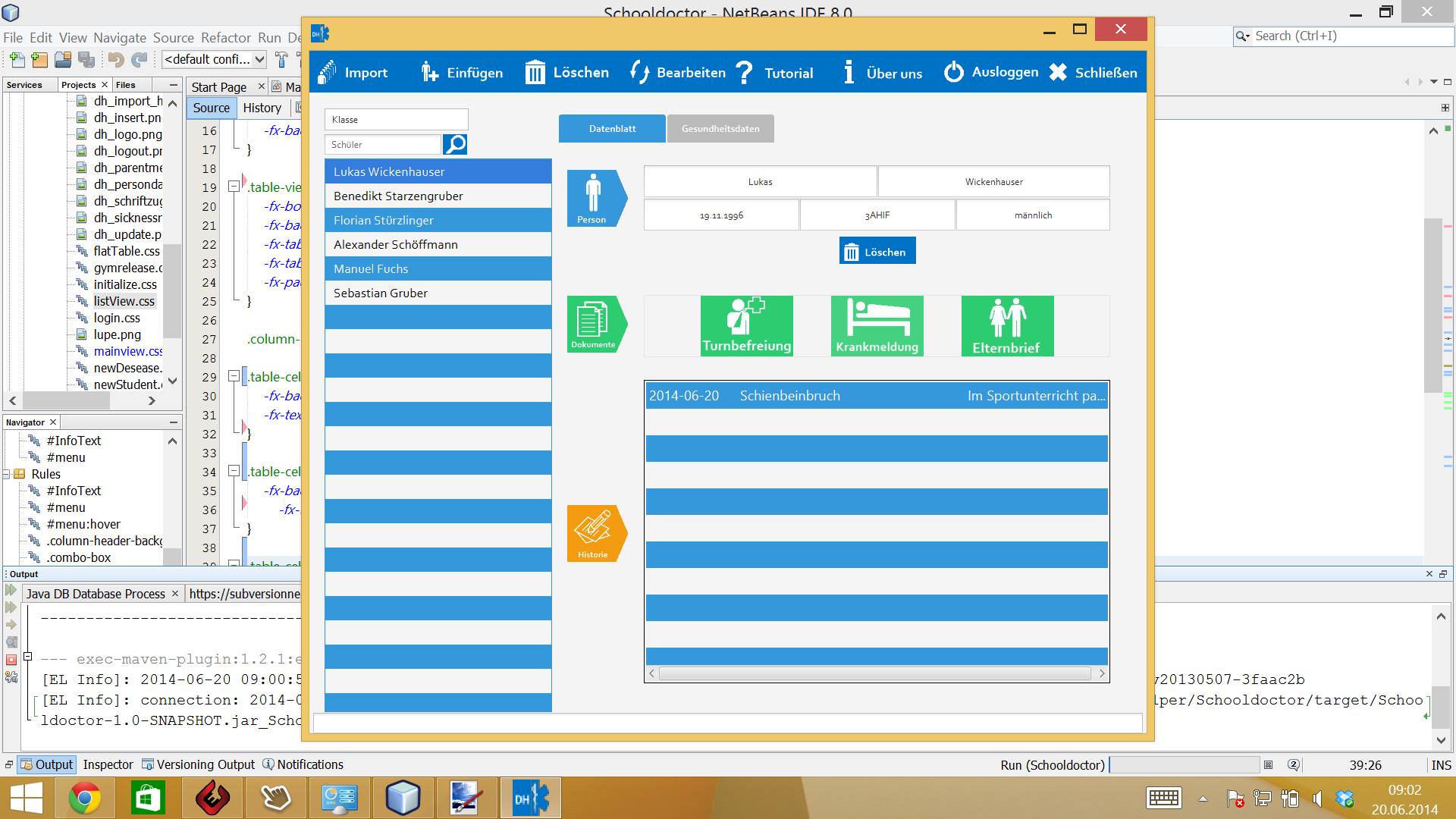
Unter Events fallen alle Untersuchungen, Befreiungen und Veranstaltungen. Es werden alle Events in einer Timeline angezeigt, für Veranstaltungen und Untersuchungen werden eigene Registerkarten mit einer Übersicht dargestellt.

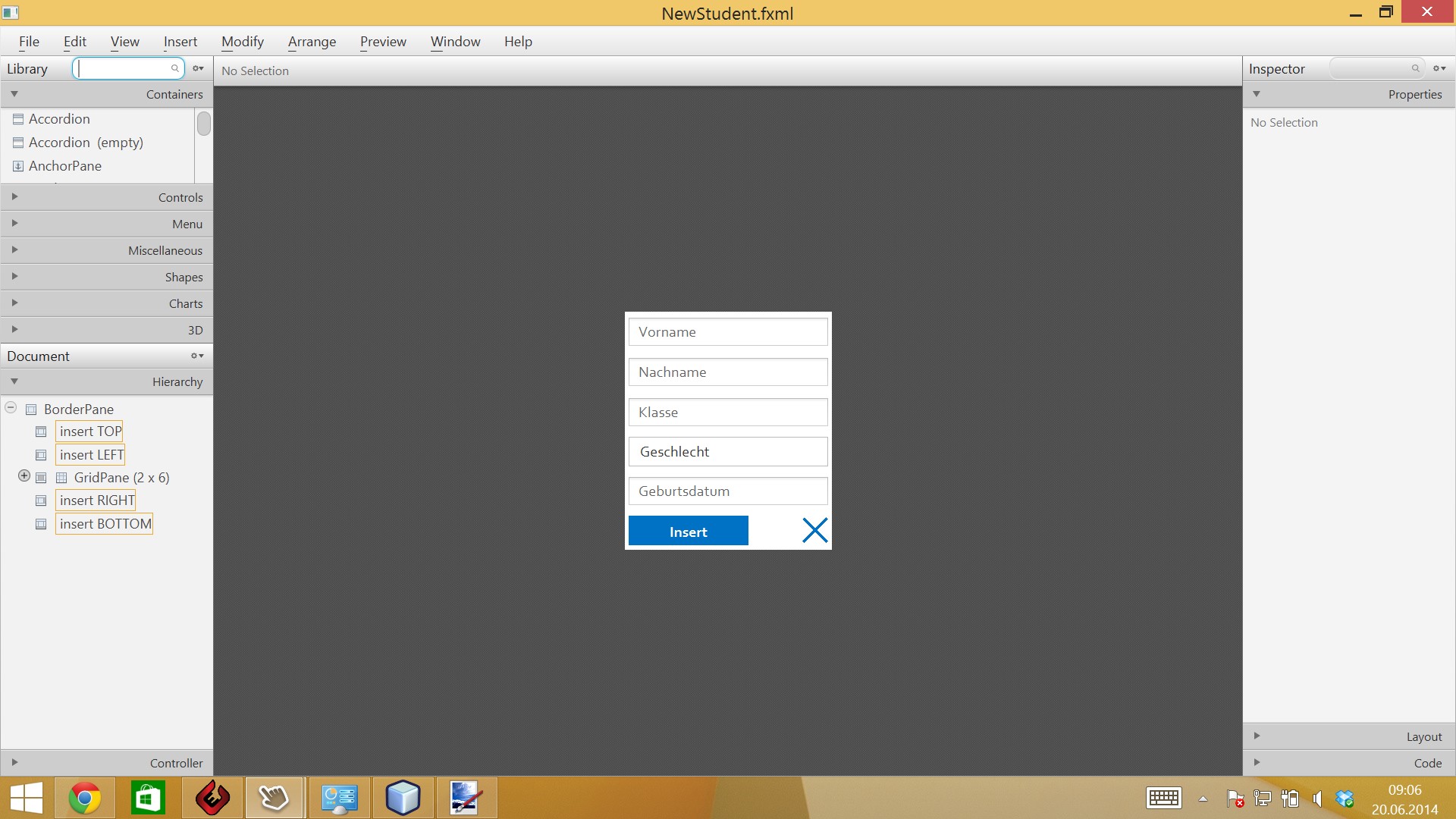
**Schnellleiste**

Für die schnellere Navigation im Programm wird außerdem am unteren Rand eine sogenannte Schnellleiste angezeigt, die die am häufigsten gebrauchten Bedienelemente als Links anzeigt.

# Benutzeroberfläche

# Login View C:\Users\Alex\Desktop\10374869_688706691192128_8549109194673320860_n.jpg Main View



Schüler importieren

# 

# Mengengerüst

Die anfallende Datenmenge hängt von der Schüleranzahl der HTL Leonding ab. Für jeden Schüler müssen aktuelle Daten, sowie die Dokumentation aus den vergangenen Jahren gespeichert werden. Einmal pro Jahr werden bei der jährlichen Untersuchung neue Daten hinzugefügt. Geht man von einer durchschnittlichen Schülerzahl von 800 aus, so würde dies in etwa folgenden Datenmengen entsprechen.

|  |  |
| --- | --- |
| Datenmenge pro Schüler | ~ 10 KB |
| Gesamte Schülerdatenmenge | ~ 8.2 MB |

Hierzu kommen noch jährlich neue Daten für die Archivierung von Schulausflugsuntersuchungen, Befreiungen und Elternbriefen. Angenommen es wird jährlich eine Ausflugsuntersuchung pro Klasse unternommen und es werden 800 Befreiungen, Elternbriefe, etc. erstellt so würde dies in etwa folgenden, jährlich hinzukommenden Datenmengen entsprechen.

|  |  |
| --- | --- |
| Datenmenge Ausflüge | ~ 150 KB |
| Datenmenge sonstiger Dokumente | ~ 850 KB |
| Jährlich neue Datenmenge | ~ 1 MB |

# Personenübersicht

## Entwicklerteam

|  |  |  |
| --- | --- | --- |
| Alexander Schöffmann | alexander.schoeffmann@gmail.com | 0650/ 6255840 |
| Benedikt Starzengruber | bene.starzengruber@gmail.com | 0660/ 3885143 |
| Florian Stürzlinger | [stueflo@gmail.com](mailto:stueflo@gmail.com) | 0660/ 4944357 |
| Lukas Wickenhauser | wicki96@gmail.com | 0650/ 7578551 |

## Betreuer

|  |  |
| --- | --- |
| Mag. Dr. Thomas Stütz | t.stuetz@htl-leonding.ac.at |