TGM

2015

E-Bibliothek

v1.2

Nenad Gligorevic

Wexstraße

Inhalt

[1.) Aufgabenstellung 3](#_Toc422222276)

[2.) Planung (Aufwandschätzung) 4](#_Toc422222277)

[Aufwandschätzung für Benutzerfunktionen 4](#_Toc422222278)

[Planungsergebnis 5](#_Toc422222279)

[3.) Datenbankdesign 5](#_Toc422222280)

[3.1) XML-Schema 6](#_Toc422222281)

[3.2) Propel 6](#_Toc422222282)

[Ergebnis: 7](#_Toc422222283)

[4.) Aufgabeneinteilung 8](#_Toc422222284)

[Nenad Gligorevic 8](#_Toc422222285)

[Mirza Kuljancic 8](#_Toc422222286)

[Sefa Öztürk 8](#_Toc422222287)

[Rohat Anil 8](#_Toc422222288)

[Hüseyin Bozkurt 8](#_Toc422222289)

[Antonio Papp 8](#_Toc422222290)

[5.) Skizze Design 9](#_Toc422222291)

[6.) GUI 9](#_Toc422222292)

[6.1) v1 9](#_Toc422222293)

[6.2) v2 10](#_Toc422222294)

[7.) Klassen - Konzept 10](#_Toc422222295)

[7.1) v1 10](#_Toc422222296)

[7.2) v2 11](#_Toc422222297)

[8.) Implementierung 11](#_Toc422222298)

[8.1) Benutzer registrieren 12](#_Toc422222299)

[8.2) Auf Benutzer Eingaben reagieren 12](#_Toc422222300)

|  |  |  |  |
| --- | --- | --- | --- |
| *Version* | *Datum* | *Status* | *Kommentar* |
| 1.0 | 28.04.15 | Dokumentenerstellung | Aufbau des Dokumentes |
| 1.1 | 30.04.15 | Implementierung (angefangen)  3.2 | 3.2 Propel Vorgehensweise  Implementierung |
| 1.2 | 05.05.15 | Klassen - Konzept | UML eingefügt  Implementierung 8.1 |
| 1.3 | 16.06.15 | Dokumentenfertig-stellung |  |

# 1.) Aufgabenstellung

Um die in den verschiedenen fachspezifischen Fächern erworbenen Kenntnisse zu demonstrieren und zu festigen soll ein gemeinsames Abschlussprojekt fächerübergreifend realisiert werden. Insbesondere soll dabei Augenmerk auf die Kompetenzen „User-Interface-Design“, „verteilte Systeme“, „Web Services“ sowie „Softwareentwicklung“ gelegt werden.

Es soll ein Anwendung für eine Online-Bibliothek und Lese-Seite für quelloffene Bücher erstellt werden. Der Zugriff auf die Inhalte dieser Bibliothek soll sowohl von Desktop-Systemen (Webseite) als auch von mobilen Geräten (Webseite oder App) möglich sein.

# 

# 2.) Planung (Aufwandschätzung)

### Aufwandschätzung für Benutzerfunktionen

|  |  |  |
| --- | --- | --- |
| TASK | Beschreibung | Geschätzte Zeit in h |
| Zielbestimmung | Das Lastenheft zerlegen und das benötigte Visualiseren  --> Verständlichkeit | 2 |
| Klassen Konzept (UML) | Die Überlegung wieviel Klassen es gibt und welche Aufgaben diese haben | 4 |
| XML - Schema | XML-Schema erzeugen und Propel Befehle ausführen | 1 |
| Aufgabeneinteilung | Wer welche Funktionen übernimmt | 3 |
| GUI Erstellung | Die Erstellung der GUI ohne Funktionalität, Skizze (Blatt + Photoshop) | 5 |
| Registrierung | Die Funktionen werden implementiert | 5 |
| Login & Logout | Die Funktionen werden implementiert | 5 |
| Passwort zurücksetzen | Die Funktionen werden implementiert | 4 |
| OAuth | Die Funktionen werden implementiert | 4 |
| User löschen | Die Funktionen werden implementiert | 4 |
| Benutzerrechte verwalten | Die Funktionen werden implementiert | 5 |

|  |  |  |
| --- | --- | --- |
| Geschätzte Zeit in h |  | 42 h |

### Planungsergebnis

|  |  |  |  |
| --- | --- | --- | --- |
| TASK | Geschätzte Zeit in h | Zeit in h | Kommentar |
| Zielbestimmung | 2 | 5 | Das Zielerfassen hat in der Gruppe länger gedauert als geplant |
| Klassen Konzept (UML) | 4 | 2 | Schneller als geplant |
| XML - Schema | 1 | 1 | Keine Probleme |
| Aufgabeneinteilung | 3 | 4 | Verständnisprobleme |
| GUI Erstellung | 5 | 3 | Keine Probleme  Schneller da als Gruppe aggiert |
| Registrierung | 5 | 10 | Ungeplante Funktionen sind dazugekommen |
| Login & Logout | 5 | 10 | --//-- |
| Passwort zurücksetzen |  |  |  |
| OAuth |  |  |  |
| User löschen | 4 |  |  |
| Benutzerrechte verwalten |  |  |  |

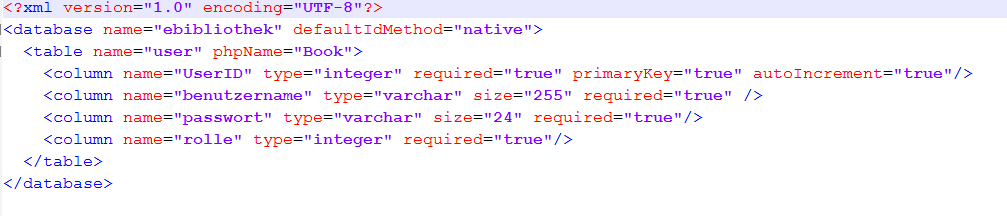
|  |  |  |
| --- | --- | --- |
| Benötigte Zeit in h |  | h |

# 3.) Datenbankdesign

Welche Daten benötigen wir ?

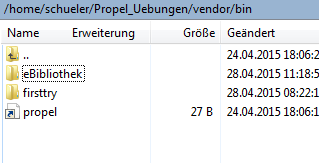
* ID
* Benutzername
* Passwort
* rolle

### 3.1) XML-Schema



### 3.2) Propel

Verzeichnis anlegen eBibliothek :



Befehle eingeben:

propel model:build

propel sql:build

Das in composer.json einfügen

"autoload": {

"classmap": ["generated-classes/"]

}

dann folgenden Befehl eingeben:

php /composer.phar dump-autoload

dann:

propel config:convert

Folgenden 2 Zeilen sind notwendig um auf die erzeugten Propel Verzeichnisse zugreifen zu können:

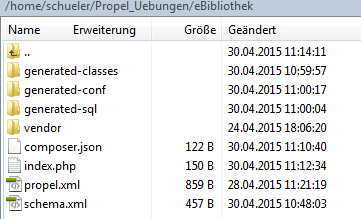
// setup the autoloading

require\_once '/path/to/vendor/autoload.php';

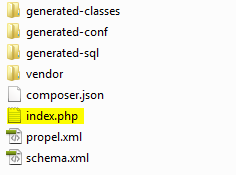
// setup Propel

require\_once '/generated-conf/config.php';

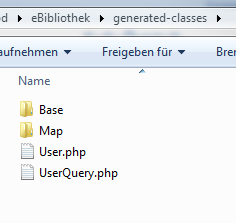
### Ergebnis:

****

Das File index.php ist unser Ausgangsfile aus diesem navigieren wir uns zu unseren Daten.



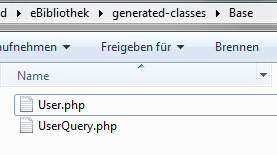
Wir greifen ausschließlich auf den Base Ornder Zu



In diesem befinden sich 2 Files:

* User.php
* UserQuery.php

Nur auf diese beiden Files soll und darf zugegriffen werden.



# 4.) Aufgabeneinteilung

### Nenad Gligorevic

* Organisation
* Benutzer registrieren

### Mirza Kuljancic

* Benutzer einloggen
* Benutzer ausloggen

### Sefa Öztürk

* OAuth (Facebook..)

### Rohat Anil

* User löschen

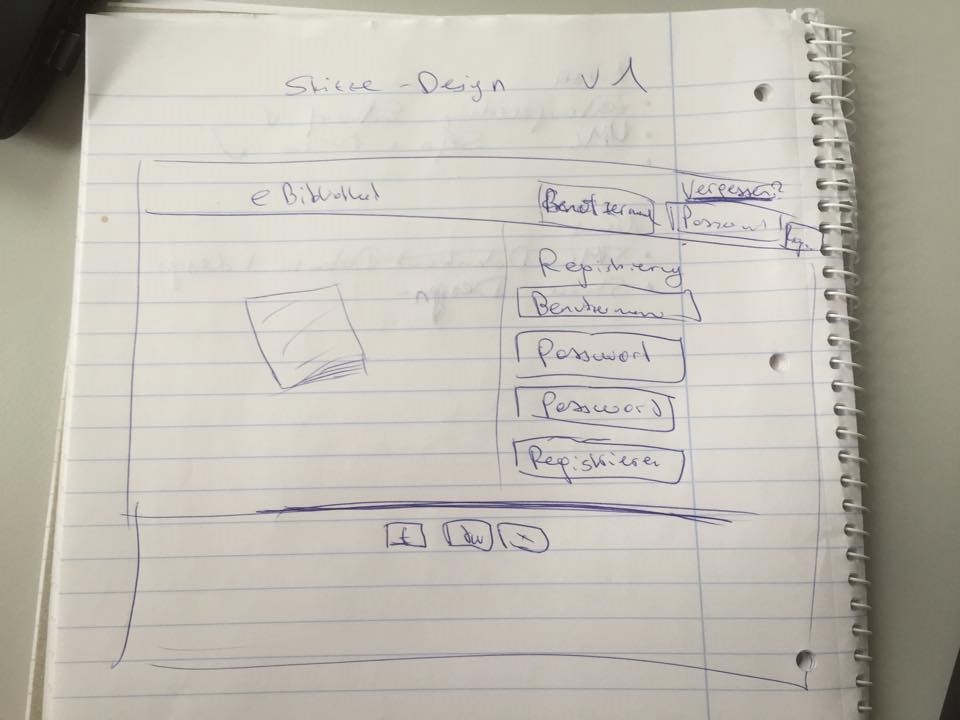
### Hüseyin Bozkurt

* Passwort zurücksetzen

### Antonio Papp

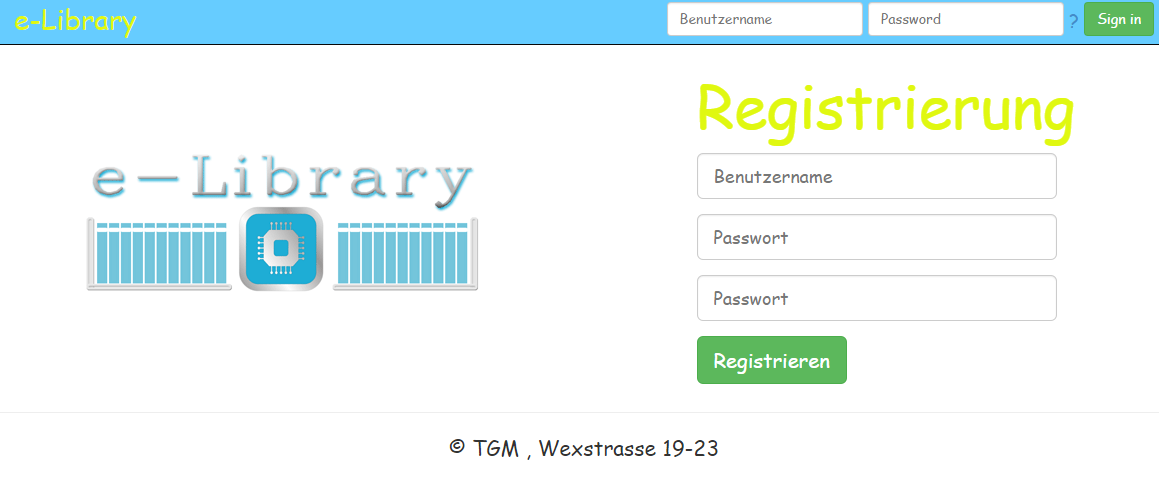
* Benutzerrechte verwalten

# 5.) Skizze Design



# 6.) GUI

### 6.1) v1

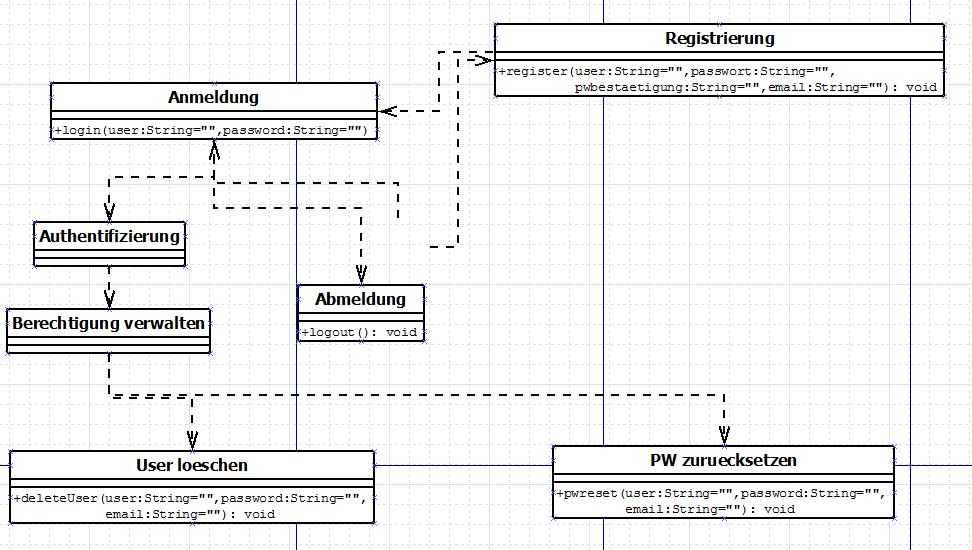


### 6.2) v2

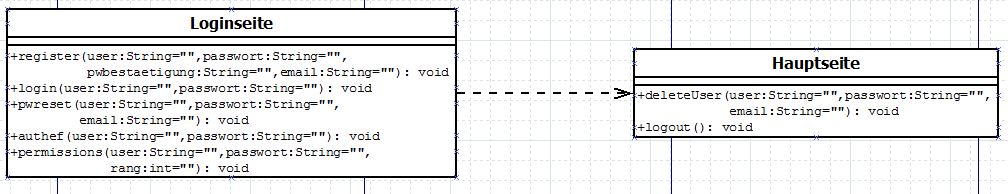


# 7.) Klassen - Konzept

### 7.1) v1



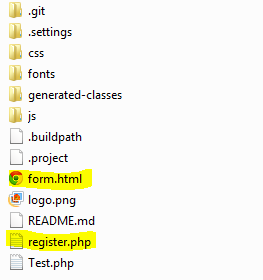
### 7.2) v2



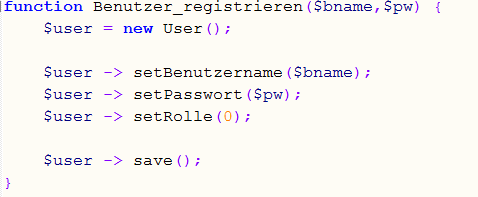
# 8.) Implementierung

Es gibt ein form -File indem sich die GUI befindet

Und ein Register -File indem die Funktionalität steckt.



### 8.1) Benutzer registrieren

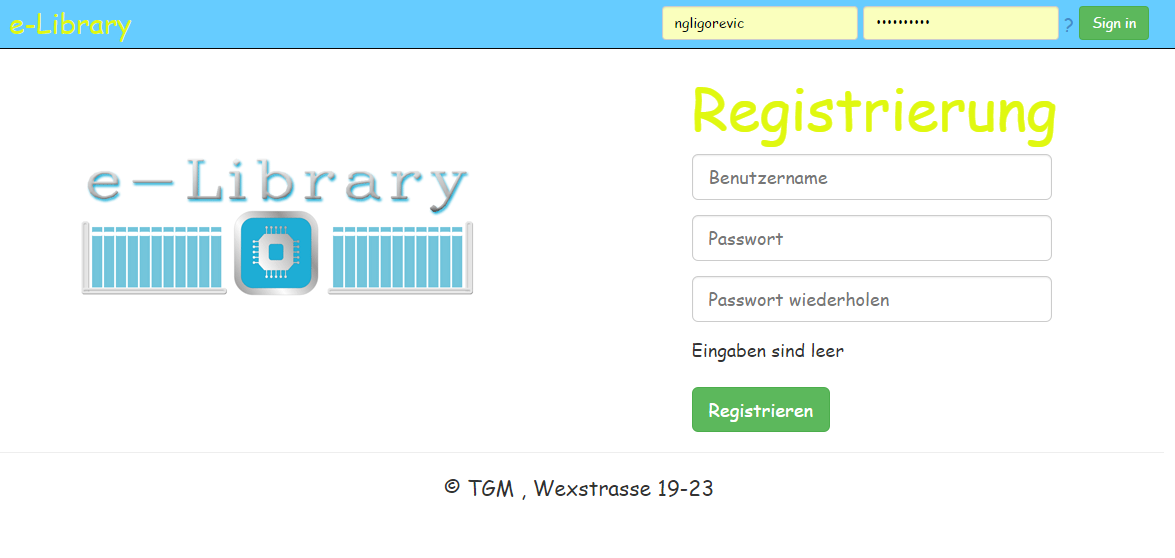


### 

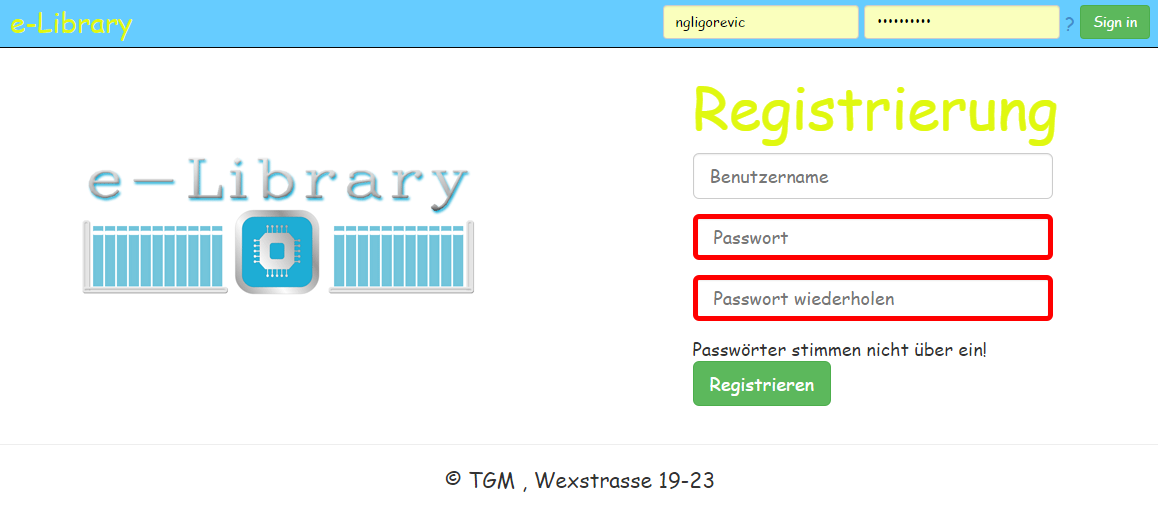
### 8.2) Auf Benutzer Eingaben reagieren

|  |  |  |
| --- | --- | --- |
| Benutzer Eingabe | Funktioniert | Funktioniert nicht |
| 1.) Leer | X |  |
| 2.) Pw stimmen nicht überein | X |  |
| 3.) Richtig | X |  |
| 4.) Benutzer existiert bereits | X |  |
| 5.) Sonderzeichen meiden | X |  |
| 6.) Eingabenlänge | X |  |

**1.) Leere Eingabe**



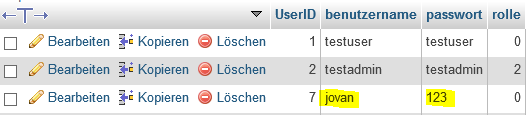
**2.) Passwörter stimmten nicht überein**



**3.) Richtige Eingabe**

Resultat:

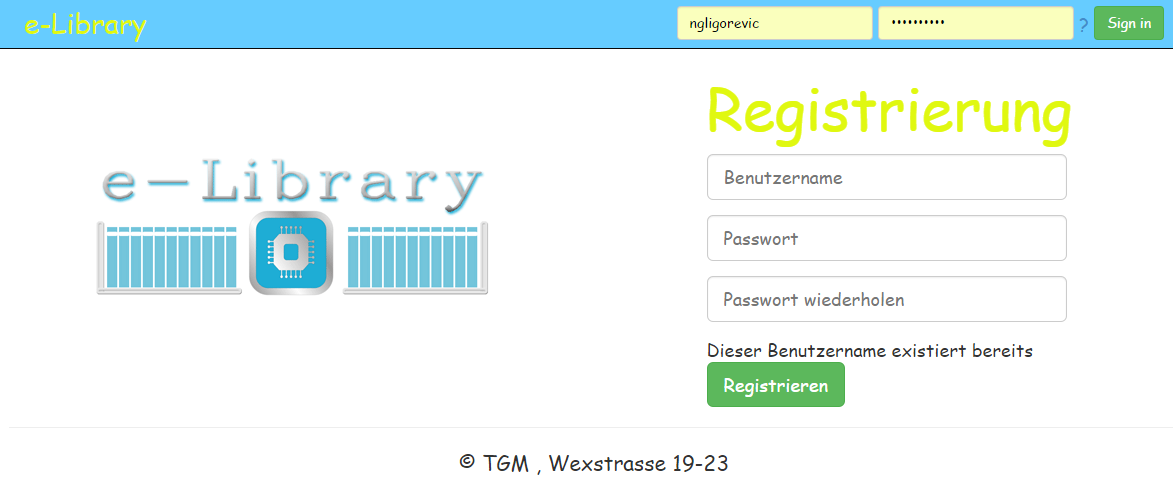
Ein neuer Benutzer



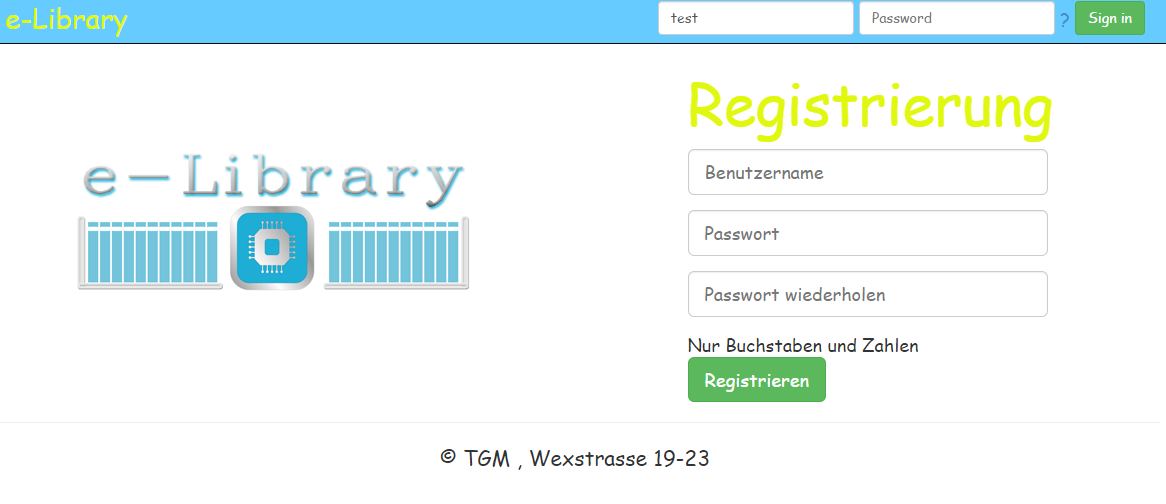
Wird auf die nächste Seite gebracht dazwischen findet eine Animation statt damit man nicht in der DB nachschauen muss ob ein Benutzer angelegt wurde.



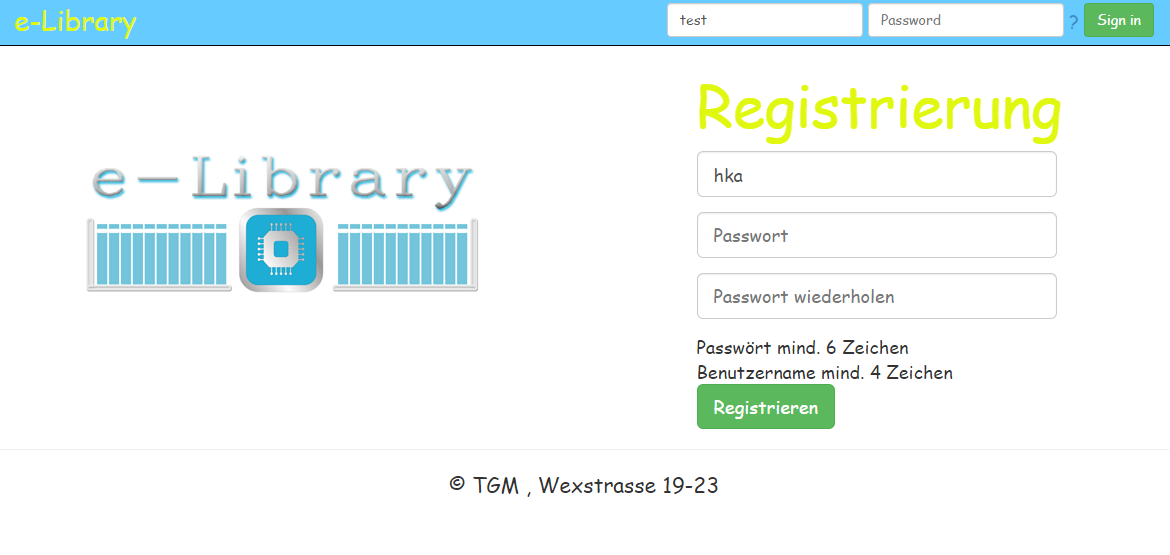
**4.) Benutzername existiert bereits**



**5.) Sonderzeichen meiden**

****

**6.) Eingabenlänge**

****