**Webapp:** Die Webapp besteht aus einer NodeJS Runtime und verwendet AWS Cognito, um User zu verwalten. Diese Cognito User können sich dann über ein Federated Indentity Login Modell bei der Webapp anmelden. Derzeit bietet die Webapp folgende Funktion: Login Page, auf der sich User anmelden können; SignUp Page auf der neue Cognito User über die AWS API erstellt werden können; Eine Index Page auf der der Inhalt der Webapp angezeigt wird.

**Index:**

Die Index Seite besteht zurzeit aus zwei Webpages. Bei diesen wird unterschieden ob sich ein User angemeldet hat oder nicht:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

*User ist nicht angemeldet.*

Ein Bild, das Text, Im Haus, Screenshot enthält.

Automatisch generierte Beschreibung

*User ist angemeldet. Oben rechts wird ebenfalls noch der Username des aktuellen Users angezeigt.*

Dieses Verhalten wird durch eine Express-Session gesteuert. Meldet sich ein an, so wird sein Cognito User Token in einer User spezifischen Session gespeichert, nur wenn dieser Token in der Session vorhanden ist wir die tatsächliche Index Page angezeigt:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

*Mit Express-Session können user spezifische Session erstellt werden, die der Server verwaltet. Auf die Session Variable kann dann über „req.session“ zugegriffen werden (Anwendung folgt unten)*

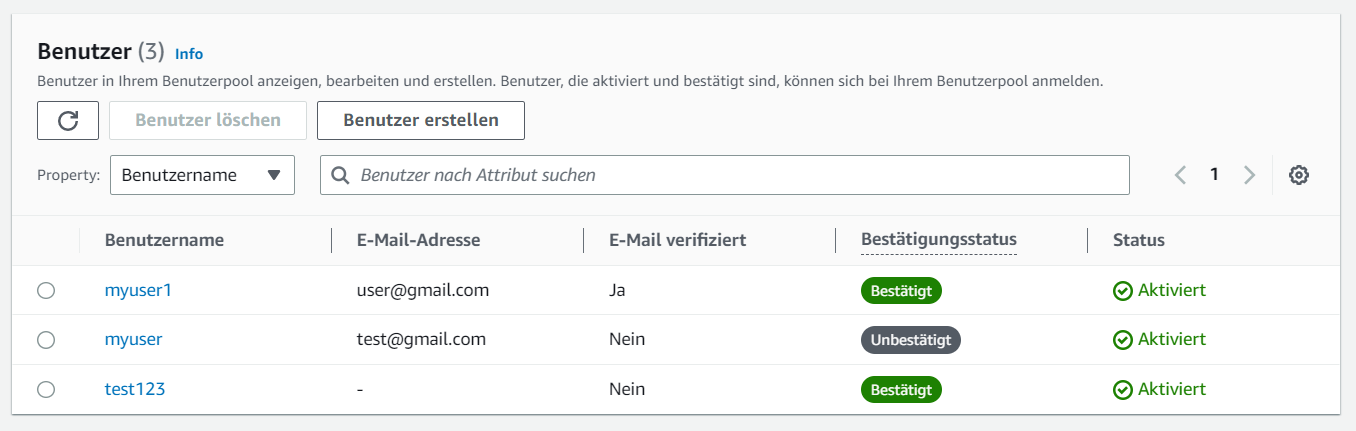
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

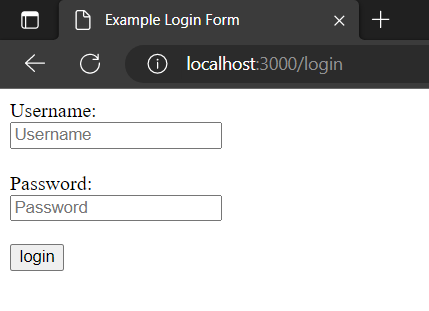
*Die tatsächliche Index Seite wird auch nur aufgerufen, wenn der AccessToken des Users in der Session gespeichert wurde.*

**Login:**

Auf der Login Page können sich Cognito User anmelden, die im zugehörigen User Pool erstellt wurden.



*Ausschnitt der User des User Pools*.



*AWS Cognito User können sich hier anmelden.*

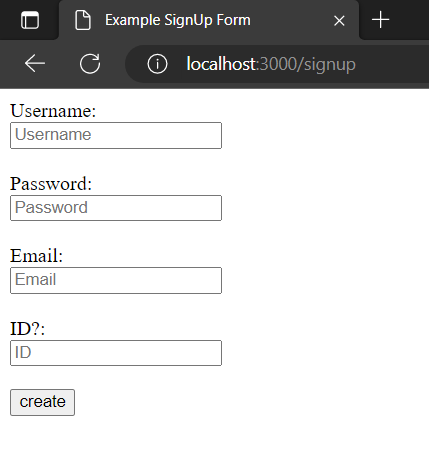
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

*Sind die Credentials gültig wird der mit dem User verknüpfte AccessToken in die User Session gespeichert und der Benutzer kann, wie oben erwähnt, die Index Seite aufrufen.*

**SignUp:**

Es sollten ebenfalls User über die Webapp erstellt werden. Diese Funktion sollte allerdings nur für Admins zur Verfügung stehen.



*Formular mit welchem User erstellt werden können. Diese User werden dann im bereits oben gezeigten Pool* *erstellt.*

**Anhang:**

[AWS Login in Node.js App]: <https://onexlab-io.medium.com/aws-cognito-node-js-cc05760b61c3>

[AWS Login mit username in Node.js App]: <https://medium.com/@prasadjay/amazon-cognito-user-pools-in-nodejs-as-fast-as-possible-22d586c5c8ec>

[Express-Session]: <https://www.section.io/engineering-education/session-management-in-nodejs-using-expressjs-and-express-session/>

[(AWS Console) User in User Pool verifizieren]: “aws cognito-idp admin-set-user-password --user-pool-id "" --username "" --password "" –permanent”

[(AWS Console) User Attribute verifizieren]: “aws cognito-idp admin-update-user-attributes --user-pool-id "" --username "" --user-attributes Name=email\_verified,Value=true”