

# Probeaufgabe Benutzerverwaltung

Entwickeln Sie eine Bibliothek, mit der Benutzeranmeldungen in Anwendungen verwaltet werden können.

Der Kontrakt für die Leistungen der Bibliothek soll so aussehen:

```
interface IRegistrierung
{
    void Registrieren(string email, string password, string nickname);
    void Bestaetigen(string registrierungsnummer);
}

interface IAnmeldung
{
    string Anmelden(string anmeldename, string password);
    bool IstAnmeldungGuelting(string token);

    void RuecksetzungDesPasswortsBeantragen(string email);
    void PasswortZuruecksetzen(string email);
}

interface IVerwaltung
{
    Benutzer AktuellerBenutzer(string token);

    void PasswortAendern(string benutzerId, string password);
    void Loeschen(string benutzerId, string password);
}

class Benutzer
{
    string Id;
    string Email;
    string Nickname;
    string Passwort;
    bool Bestaetigt;
    DateTime Registrierungsdatum;
    DateTime LetzteAnmeldung;
    DateTime LetzteAktualisierung;
}
```

Neue Benutzer registrieren sich zunächst. Sie müssen mindestens ihre E-Mail Adresse angeben. Wenn kein Passwort gewählt wurde, generiert die Registrierung eines, das der Benutzer später ändern kann.

Wer sich registriert, bekommt eine Registrierungsemail geschickt mit einem Link zur Bestätigung, in dem eine Registrierungsnummer enthalten ist. Erst wenn die Registrierung mit dieser Nummer bestätigt ist, ist der Benutzer permanent im System (Benutzer.Bestaetigt ist dann true).

Die Anmeldung erfolgt mit E-Mail Adresse oder Nickname und Passwort nach Bestätigung. Ist sie erfolgreich, liefert sie ein Token zurück. Das kann später immer wieder bei Bedarf zur Prüfung vorgelegt werden, ob Anfragen von einem Client gültig sind.

Wer sein Passwort vergessen hat, kann eine Zurücksetzung beantragen. Es wird dann eine Nachricht an die E-Mail Adresse gesendet, in der ein Link steht, über den man ein neues Passwort anfordern kann. Wird der angeklickt, wird ein Passwort generiert und an die beantragende E-Mail Adresse versandt.

Passwörter werden nicht im Klartext gespeichert.

Zu beachten ist hierbei, dass der gezeigte Code als grobe Richtschnur zu verstehen ist und bei Bedarf gerne erweitert werden kann.

Die folgenden Funktionen müssen nicht real implementiert werden, sollten aber durch geeignete Abstraktionen vorgesehen werden und als Dummy oder In-Memory implementiert werden:

- E-Mails versenden
- Passwörter verschlüsseln
- Speicherung der Benutzer