**Meilenstein 1a: Anforderungsanalyse News-Portal**

Das Startup *UNITALK*, das es Studierenden ermöglichen möchte anonym zu kommunizieren, hat mich mit der Erstellung einer Datenbank beauftragt.

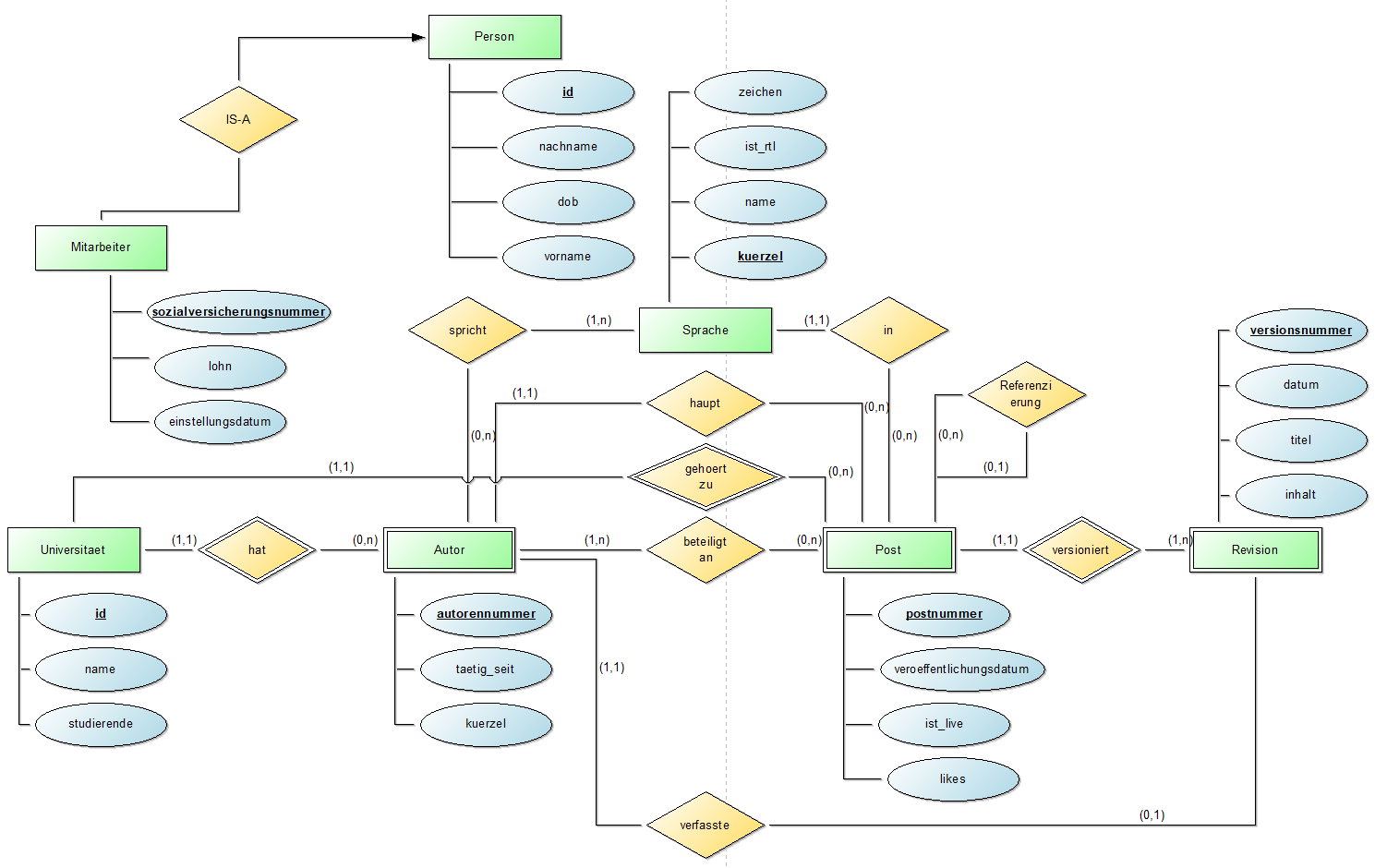
Die Datenbank soll der Verwaltung veröffentlichter Artikel (eindeutige Artikelnummer, Veröffentlichungsdatum, ist live / aufrufbar), welche eindeutig einer Universität (Name, Anzahl Studierende) zugeordnet werden, dienen. Personen (Vorname, Nachname, DOB\*) können entweder ~~Autoren (Autorennummer, tätig seit, kürzel)~~ oder Mitarbeiter (eindeutige Sozialversicherungsnummer, Lohn, Einstellungsdatum) sein. Autoren sprechen eine oder mehrere Sprachen (eindeutiges Kuerzel, Name, Zeichen, RTL\*\*), in welchen sie sich an Artikeln beteiligen können. Autoren schreiben nur für genau eine Universität, welcher sie untergeordnet sind.

Artikel werden in einer Sprache und für eine Universität veröffentlicht. Es können auch mehrere Autoren derselben Universität an einem Artikel beteiligt sein, jedoch hat jeder Artikel einen Hauptautor, welcher die Verantwortung für den Inhalt übernimmt. Artikel können maximal einen anderen Artikel referenzieren, jedoch kann ein Artikel von beliebig vielen Artikeln referenziert werden.

Weiter sollen Revisionen (eindeutige Versionsnummer, Titel, Inhalt, Datum) von Artikeln verwaltet werden, welche Auskunft darüber geben, welcher Autor zu welcher Zeit einen Artikel erstellt oder aktualisiert hat.

\*DOB (date of birth) steht für das Geburtsdatum

\*\*RTL (right to left) bezeichnet die Leserichtung von Text

**Meilenstein 1b: Konzeptioneller Entwurf**

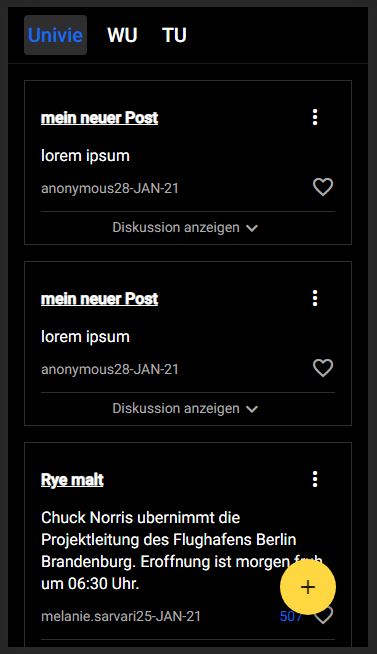
**Meilenstein 2: Logischer Entwurf**

|  |
| --- |
| **person** (id, nachname, vorname, dob)  PK: id |
| **mitarbeiter** (snr, person\_id, lohn, einst\_datum)  PK: snr  FK: person\_id ◊ person |
| **sprache** (kuerzel, ist\_rtl, name, zeichen)  PK: kuerzel |
| **universitaet** (id, name, studierende)  PK: id |
| **autor** (id, person\_id, uni\_id, taetig\_seit, kuerzel)  PK: id  ~~FK: person\_id ◊ person~~  FK: uni\_id ◊ universitaet |
| **autor\_hat\_sprache** (autor\_id, sprache\_kuerzel)  PK: autor\_id, sprache\_id  FK: autor\_id ◊ autor  FK: sprache\_kuerzel ◊ sprache |
| **post** (id, hauptautor\_id, ref\_artikel\_id, sprache\_kuerzel, uni\_id, datum, ist\_live)  PK: id  FK: hauptautor\_id ◊ autor  FK: ref\_post\_id ◊ post  FK: sprache\_ kuerzel ◊ sprache  FK: uni\_id ◊ universitaet |
| **autor\_hat\_post** (autor\_id, post\_id)  PK: autor\_id, post\_id  FK: autor\_id ◊ autor  FK: post\_id ◊ post |
| **revision** (post\_id, versnr, uni\_id, autor\_id, datum, titel, inhalt)  PK: post\_id, versnr  FK: uni\_id ◊ universitaet  FK: autor\_id ◊ autor |

**Meilenstein 4: Implementierung**

<http://wwwlab.cs.univie.ac.at/~christophb77/dbs/>

Die Datenbank wird in Oracle SQL Developer programmiert:

Die Business-Logik und UI werden in Angular programmiert:  


**Deployment:**

1) clone the repository on almighty:

git clone https://github.com/christophbuehler/dbs-project.git .

2) create web hook on GitHub to grab the latest changes on commit:

<http://wwwlab.cs.univie.ac.at/~christophb77/dbs/deploy.php>

excluding sensitive information from Git:  
