# Text and ERD

To design our database, we used an online app at dbdiagram.io. We chose this because it allowed us to describe our tables in text with a kind of simplified MySql, and the app generated the diagram dynamically.

Please note that we deviated from the naming convention for each record's id column. We have found that using a generic "id" can cause confusion when table aliases are used, and also when using logging tables. In general, having more explicit id column names makes the code easier to read.

# Text

```
Table users as U {
  user_id int         [pk, increment] // auto-increment
  first_name varchar  [not null]
  last_name varchar   [not null]
  email varchar       [not null]
  login_password varchar [not null]
  recovery_email varchar
  phone int           // For two-step verification
}

Table login_history {
  lh_id int           [pk, increment]
  timestamp datetime
  user_id int         [ref: > U.user_id]
}


Table url {
  url_id int          [pk, increment] // auto-increment
  url varchar         [not null]
  password varchar    [not null]
  user_name varchar
  user_id int         [ref: > U.user_id]
  password_hint varchar
}

Enum record_action {
  insert
```

```
  update
  delete
}

Table password_history as PH {
  ph_id int     [pk, increment] // auto-increment
  url_id int    [ref: > url.url_id]
  timestamp datetime
  action record_action
  old_password varchar
  new_password varchar
  old_password_hint varchar
  new_password_hint varchar
}

Table shared_passwords as SP {
  sp_id  int      [pk, increment] // auto-increment
  url_id int      [ref: > url.url_id]
  owner_id int    [ref: > U.user_id]
  recipient_id int [ref: > U.user_id]
}

Table shared_password_history as SPH {
  sph_id int      [pk, increment] // auto-increment
  sp_id  int      [ref: > SP.sp_id]
  action record_action
  old_recipient_id int [ref: > U.user_id]
  new_recipient_id int [ref: > U.user_id]
}

Table security_questions as SQ {
  sq_id  int      [pk, increment] // auto-increment
  question varchar [not null]
}

// How can we make the combination of sq_id and user_id to be unique?
Table security_answers as SA {
  sa_id int       [pk, increment] // auto-increment
  sq_id int       [ref: > SQ.sq_id]
  answer varchar  [not null]
  user_id int     [ref: > U.user_id]
}

Table faq {
```

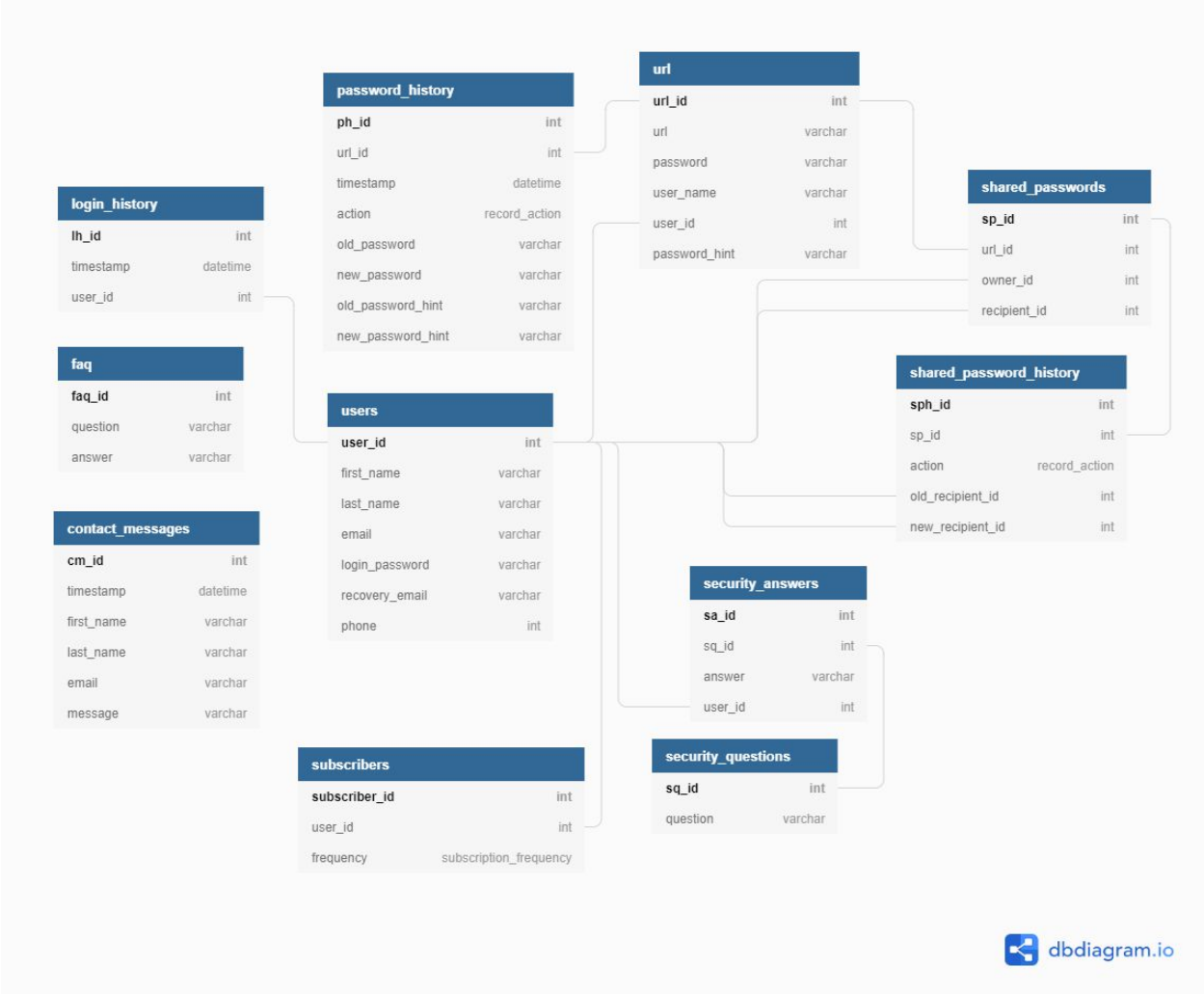
```
  faq_id int        [pk, increment] // auto-increment
  question varchar  [not null]
  answer varchar    [not null]
}

Enum subscription_frequency {
  weekly
  monthly
  special
}

Table subscribers as S {
  subscriber_id int   [pk, increment] // auto-increment
  user_id int         [ref: - U.user_id]
  frequency subscription_frequency
}

Table contact_messages as CM {
  cm_id int        [pk, increment] // auto-increment
  timestamp datetime
  first_name varchar  [not null]
  last_name varchar
  email varchar       [not null]
  message varchar     [not null]
}
```

# ERD


password_history
ph_id int
url_id int
timestamp datetime
action record_action
old_password varchar
new_password varchar
old_password_hint varchar
new_password_hint varchar
url
url_id int
url varchar
password varchar
user_name varchar
user_id int
password_hint varchar
shared_passwords
sp_id int
url_id int
owner_id int
recipient_id int
login_history
lh_id int
timestamp datetime
user_id int
faq
faq_id int
question varchar
answer varchar
users
user_id int
first_name varchar
last_name varchar
email varchar
login_password varchar
recovery_email varchar
phone int
shared_password_history
sph_id int
sp_id int
action record_action
old_recipient_id int
new_recipient_id int
contact_messages
cm_id int
timestamp datetime
first_name varchar
last_name varchar
email varchar
message varchar
security_answers
sa_id int
sq_id int
answer varchar
user_id int
subscribers
subscriber_id int
user_id int
frequency subscription_frequency
security_questions
sq_id int
question varchar
dbdiagram.io