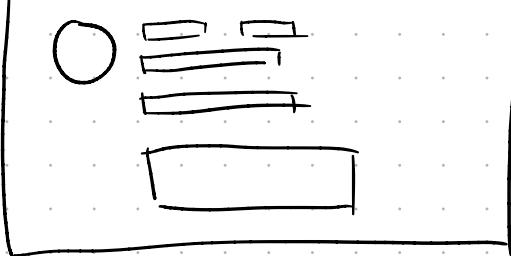


## Registration



User Input

**create session token or any form of authorization (save valid tokens in database?)**

201 created  
POST

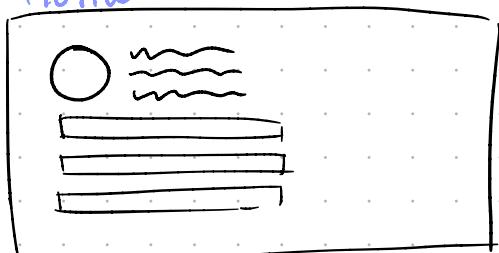
create user + add id

→ save in dB "users"

hash password

save credentials

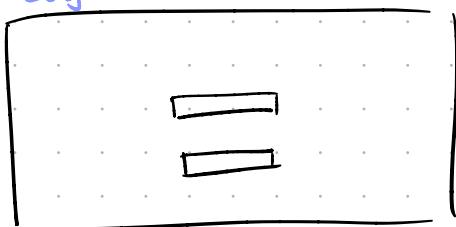
## Profile



User wants to change his information PUT 204 no content → update in dB

User delete DELETE → delete in dB 204 no content

## Log out



kill session token

## Homepage

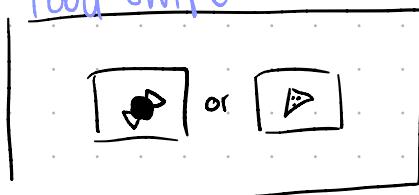


Button → Trigger swiping

should include:

- Profile
- Log out
- maybe active choice

## food swipe



get a set of random food

↳ maybe in a static file server or database

GET

all food choices

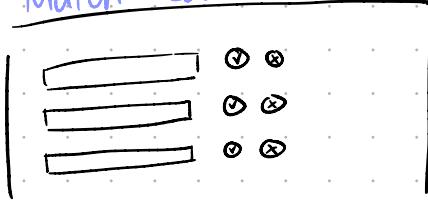
→ backend should limit selection  
→ frontend needs to implement logic ??

→ at the end: send key value to backend

→ PUT 204 no content / POST 201 created

↳ depending on chosen database concept/model

## Match - case there are matches



create pair → both need to "accept" in order to match and see contact information

↳ store state in database?

structure could look like this: Partner A bool

Partner B bool

pairID uuid

matched bool

on updates check matched

→ trigger something in Frontend

↳ needs some form of notification dashboard

Do we need continuous checking? Yes?

GET - Request to show matches

PUT When match is accepted or declined

when matched:

Contact info will be revealed

GET by id to retrieve from backend

no match

try again later  
or  
food swipe again

# DATABASE

## USERS

userID	first name	last name	birth date	(city) address	Instagram	mobile	role?	profile pic (check best way to save image)
--------	------------	-----------	------------	-------------------	-----------	--------	-------	--

## FOOD

food id	name	image?
---------	------	--------

2

Candidates → would be kind of temporal, use this to select top matches  
→ changes every time you start foodswipe or cancel current foodchoice

userID	food ID
--------	---------

matches → joined PK(userIDA + userIDB) → unique has to be to avoid duplicates

Partner A	Partner B	pair ID	matched
userID	userID		

set per default null, if one declines → delete entry

if match is successful, entry in candidate has to be removed