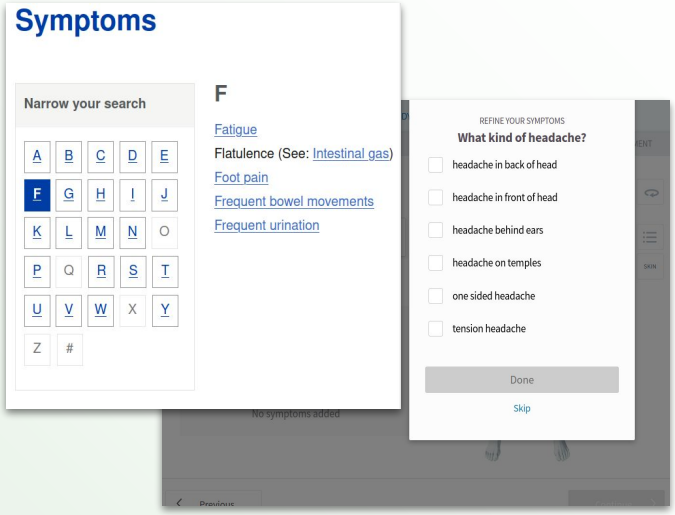
Naturdoc

Find the best natural Remedies to treat your symptoms

Problem ...

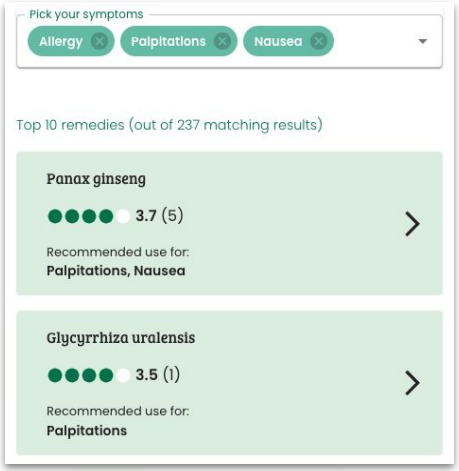


❌ Difficulty to formulate symptoms

❌ Not user-friendly

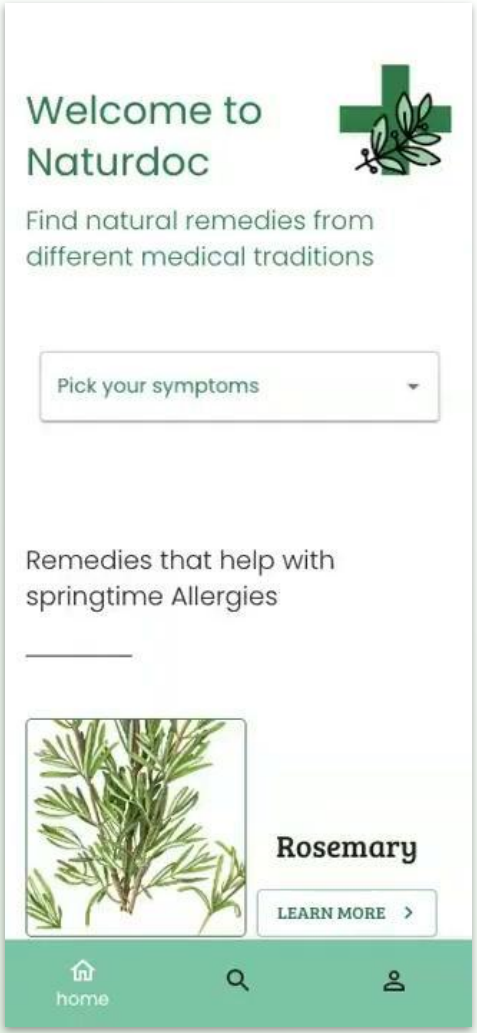
✔ The best natural treatments in two clicks

✔ Combination of data science and user-generated content



... Solution !

How does it work?



1. Matching Symptoms to best remedies

Behind the scenes:

✔ The final Dataset is created from:
- Duke's Ethnobotanical and Phytochemical Database
- WHO Monographs on Medicinal Herbs
- Google Trends Data
- Google Symptoms Dataset

✔ **13 079** remedies, **77** of which contain detailed information on medicinal application, dosage forms etc.

✔ **314** unique symptoms matching to over **2000** different medicinal uses

Data Transformations

✔ **Scikit** and **HDBSCAN** provide various clustering algorithms and other useful methods, such as generating distance matrices

✔ **SentenceTransformer** to generate word embeddings

✔ **Matplotlib** and **Seaborn** to visualise and better analyse our data

Machine Learning

✔ **Python** and its libraries **Pandas** and **Numpy** to manipulate the original data and create two final datasets for remedies and symptoms each

✔ Prepare and upload datasets to database

✔ **FastAPI** framework enabled DS to provide an end point to the backend that runs the recommendation script, filtering for the best remedy based on various criteria

✔ **Machine Learning** to provide better symptoms

Recommender API

2. The ML Problem

- We had trouble finding appropriate labels for our data, so a supervised machine learning approach seemed difficult to achieve
- The initial dataset provided a list of *ACTIVITIES* that were treated by a single remedy
    - Activities were recorded in a specific format that is not very user-friendly and not how a human would freely input a symptom (e.g. head ache as “Ache(Head)”)
    - The activity column would not just describe symptoms treated, but also included other uses. A single activity could refer to
        - an illness, such as diabetes mellitus
        - a singular symptom, such as fever
        - a culinary use, such as spice

→ **Unsupervised Clustering algorithms:**

Using word embeddings, we generated a distance matrix and clustered both activities and a set of more user-friendly symptoms based on their semantic proximity

2. Word Embeddings

Importance of word embeddings for better symptom-herb matching:

- Improve the accuracy of our symptom-herb matching algorithm.
    - Can more accurately match symptoms with appropriate herbal remedies
- Allow us to make more accurate and effective symptom-herb matches, ultimately providing our users with more personalized and effective natural remedies
- The model we used : *'average_word_embeddings_glove.840B.300d'*

Steps:

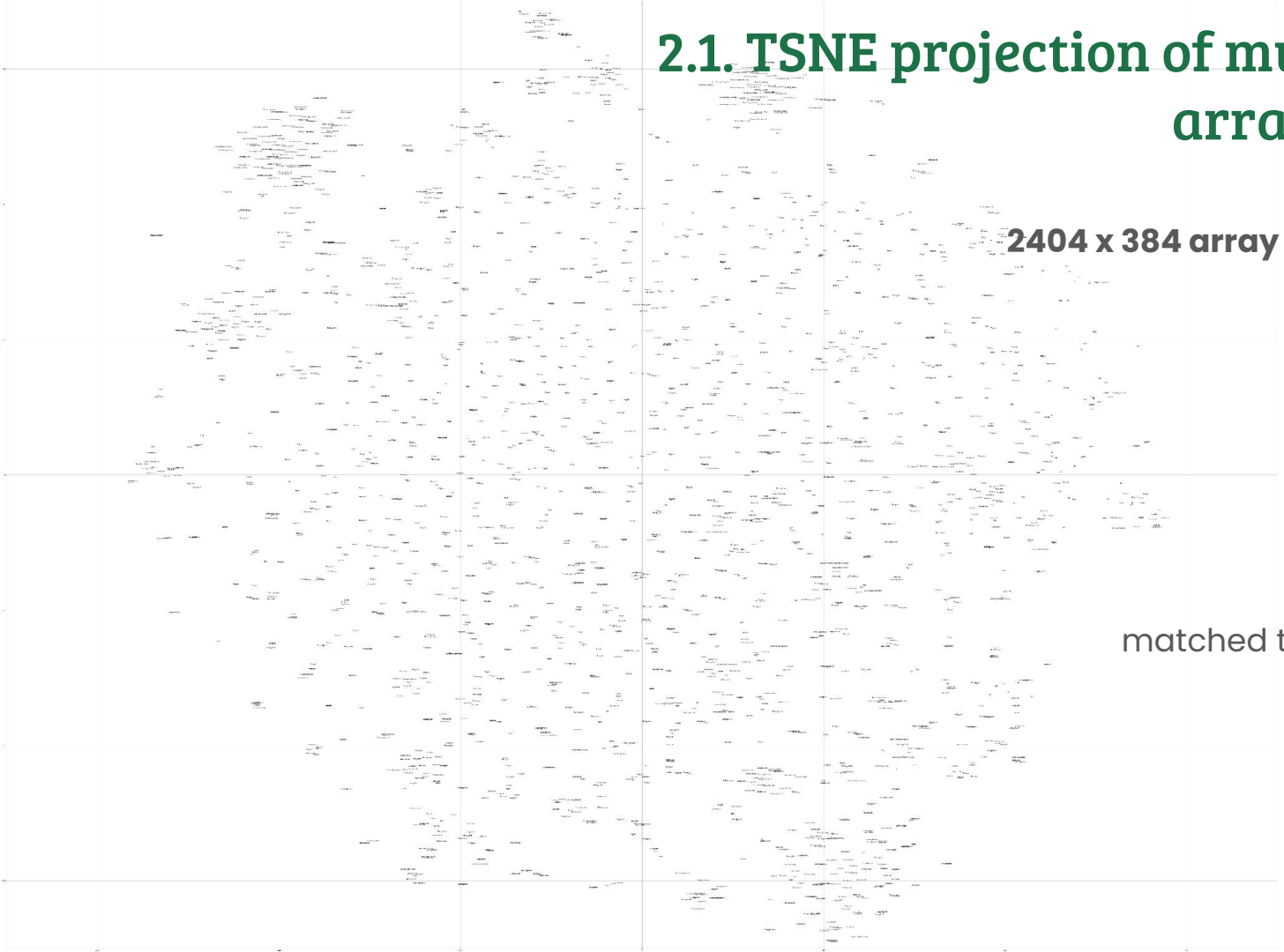
1. Converting each symptom into an embedding vector
2. Compare and match them based on their semantic similarity in the embedding space

2.1. Embeddings

2404 rows × 384 columns

	Embedding1_0	Embedding1_1	Embedding1_2	Embedding1_3	Embedding1_4	Embedding1_5	Embedding1_6
0	-9.81967244e-03	1.01662287e-02	3.75229940e-02	1.75703913e-02	-1.11436069e-01	3.83325890e-02	1.48906738e-01
1	5.98840415e-02	1.64022837e-02	-4.90665212e-02	4.81191762e-02	-9.69780684e-02	-1.16978601e-01	1.07039817e-01
2	6.30832557e-03	6.94514960e-02	9.17118881e-03	-4.25593607e-04	3.68529968e-02	2.88750455e-02	9.93606523e-02
3	-1.41132241e-02	7.76526034e-02	-8.35783686e-03	2.37053819e-02	5.61783165e-02	3.36992592e-02	1.19458653e-01
4	-7.86128864e-02	-2.58876905e-02	3.46109122e-02	5.58277592e-02	-3.87978852e-02	-5.56877032e-02	1.44394651e-01
...	...	...	...	...	...	...	...
2399	-1.92209315e-02	3.93610820e-02	-8.05331487e-03	-1.88839864e-02	1.07821608e-02	-7.60535970e-02	1.06595382e-01
2400	2.64672562e-02	-4.77555906e-03	-3.02140005e-02	-2.61425432e-02	-3.36900353e-02	-1.13033682e-01	8.00898746e-02
2401	2.54991353e-02	1.67390481e-02	5.45178875e-02	-1.12884091e-02	2.85636671e-02	-5.56535311e-02	5.83849624e-02
2402	-4.97031994e-02	1.04058813e-02	1.57771539e-02	7.78019577e-02	4.29792143e-03	1.63108837e-02	7.65634999e-02
2403	2.60038115e-02	-7.01556401e-03	-2.84907389e-02	3.80175486e-02	-1.65998936e-02	-1.13059320e-02	1.39509425e-01

2.1. TSNE projection of multidimensional array (Unclustered)



2404 x 384 array projected into 2D space:

We used TSNE to reduce the multidimensional embeddings array.

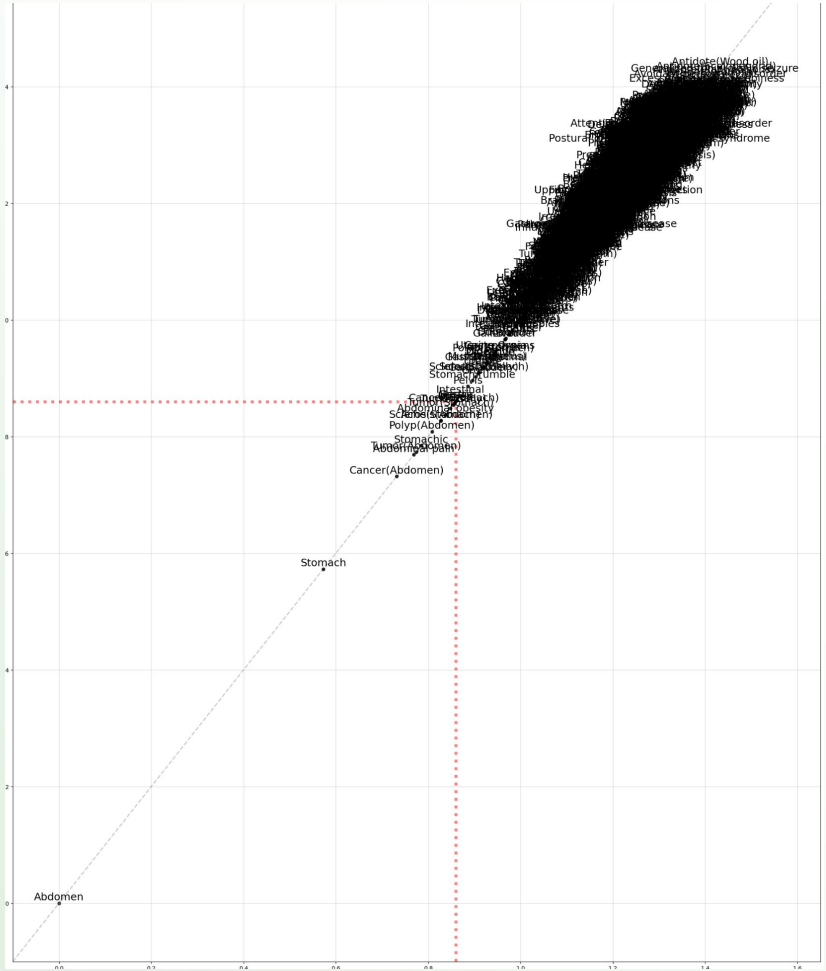
Here, we can see all symptoms and activities matched to their index in the array.

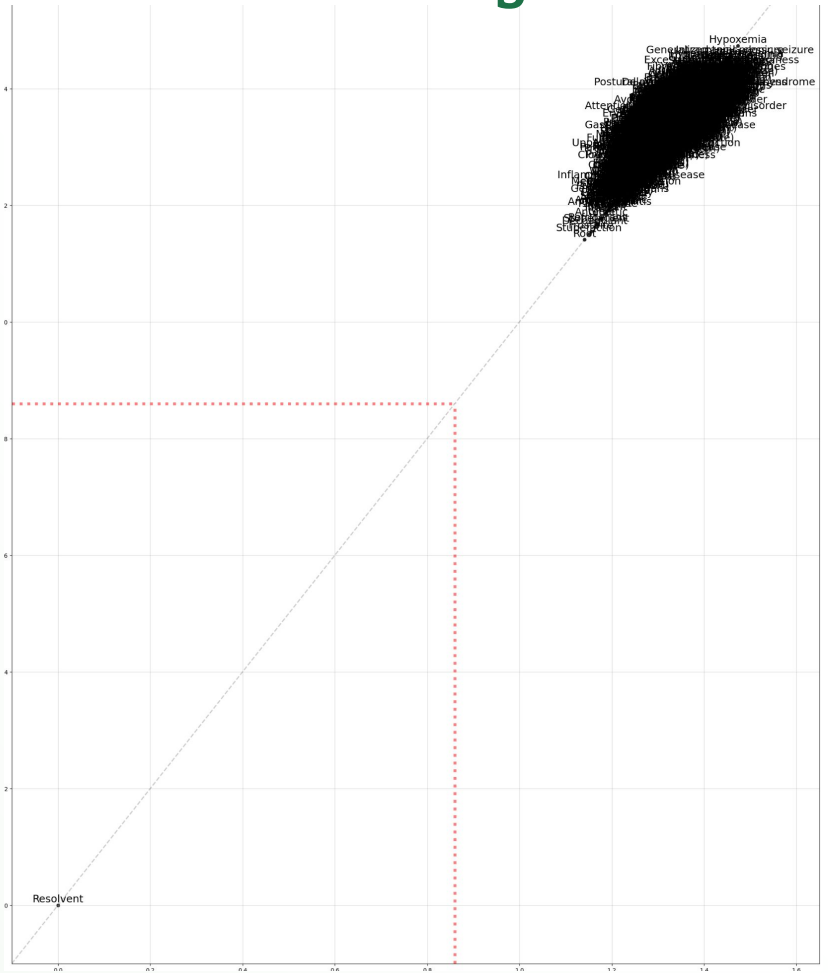
2.2. Distance Matrix

2404 rows × 2404 columns

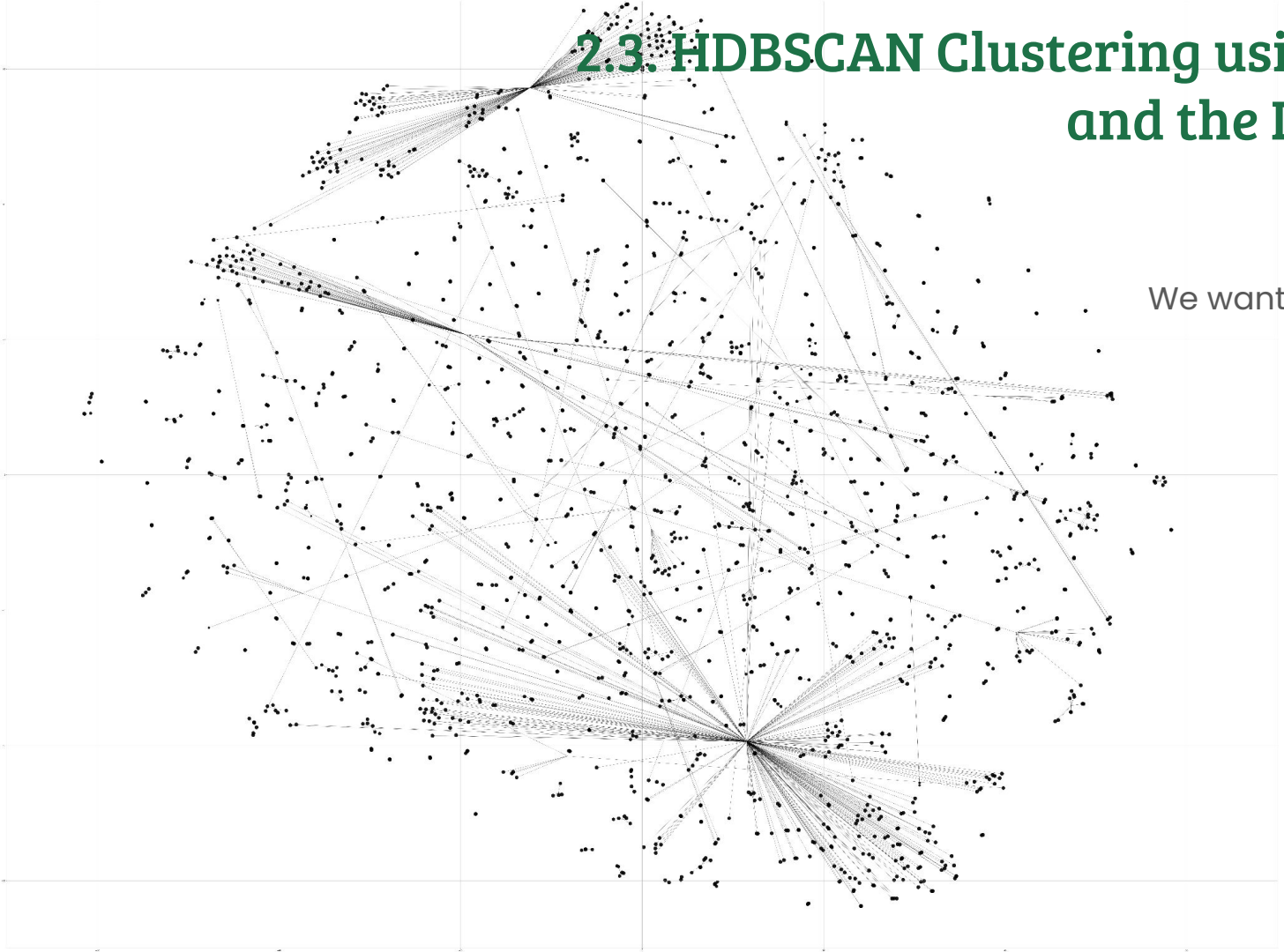
	0	1	2	3	4	5	6	7	8	9	...
0	0.000000	1.172835	1.230569	1.202147	1.068863	1.102719	1.213488	1.227862	1.129428	1.118084	...
1	1.172835	0.000000	1.308356	1.177216	1.158076	0.999987	1.074094	1.098266	1.247356	1.068514	...
2	1.230569	1.308356	0.000000	0.662869	1.238897	1.226645	1.240566	1.260014	1.054402	1.326170	...
3	1.202147	1.177216	0.662869	0.000000	1.123468	1.144537	1.210960	1.234160	1.030969	1.265411	...
4	1.068863	1.158076	1.238897	1.123468	0.000000	1.006659	1.118275	1.032273	1.103120	1.115403	...
...	...	...	...	...	...	...	...	...	...	...	...
2399	1.229228	1.200630	1.300060	1.251390	1.217910	1.257878	1.302735	1.295551	1.209389	1.270752	...
2400	1.269921	1.159814	1.313569	1.286678	1.272939	1.285842	1.289375	1.324865	1.274088	1.307315	...
2401	1.238341	1.350562	1.299467	1.263937	1.227224	1.362247	1.375149	1.326699	1.266961	1.191208	...
2402	1.326879	1.248203	1.382976	1.389592	1.202582	1.206832	1.273774	1.289590	1.201103	1.269518	...
2403	1.321452	1.022531	1.290404	1.217563	1.270807	1.167902	1.242380	1.329070	1.255640	1.252103	...

2.2. Plots of Distance for a Single Data Point





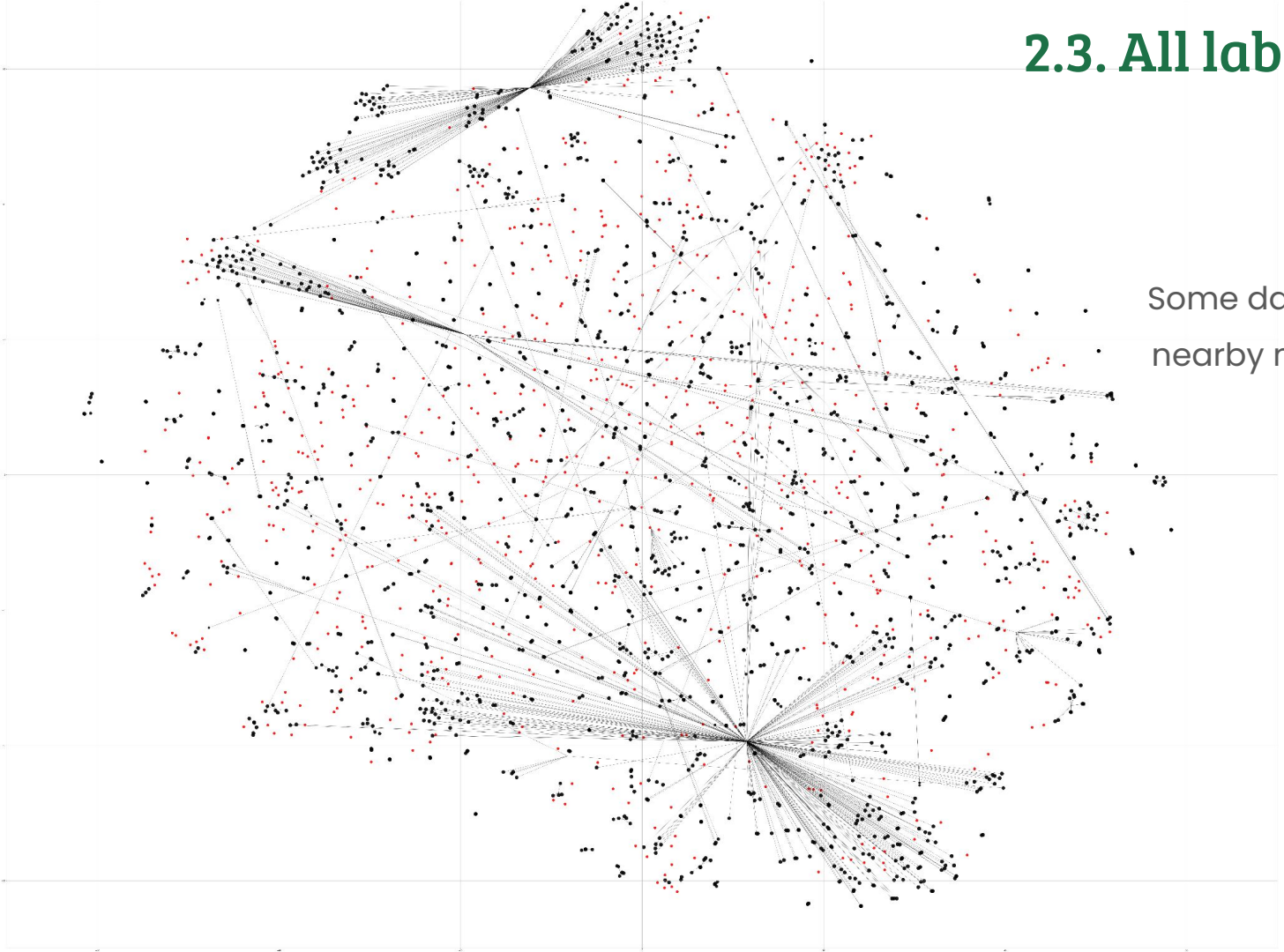
2.3. HDBSCAN Clustering using Embeddings and the Distance Matrix



431 labels:

We want to allow smaller clusters to exist, to allow for a finer matching.

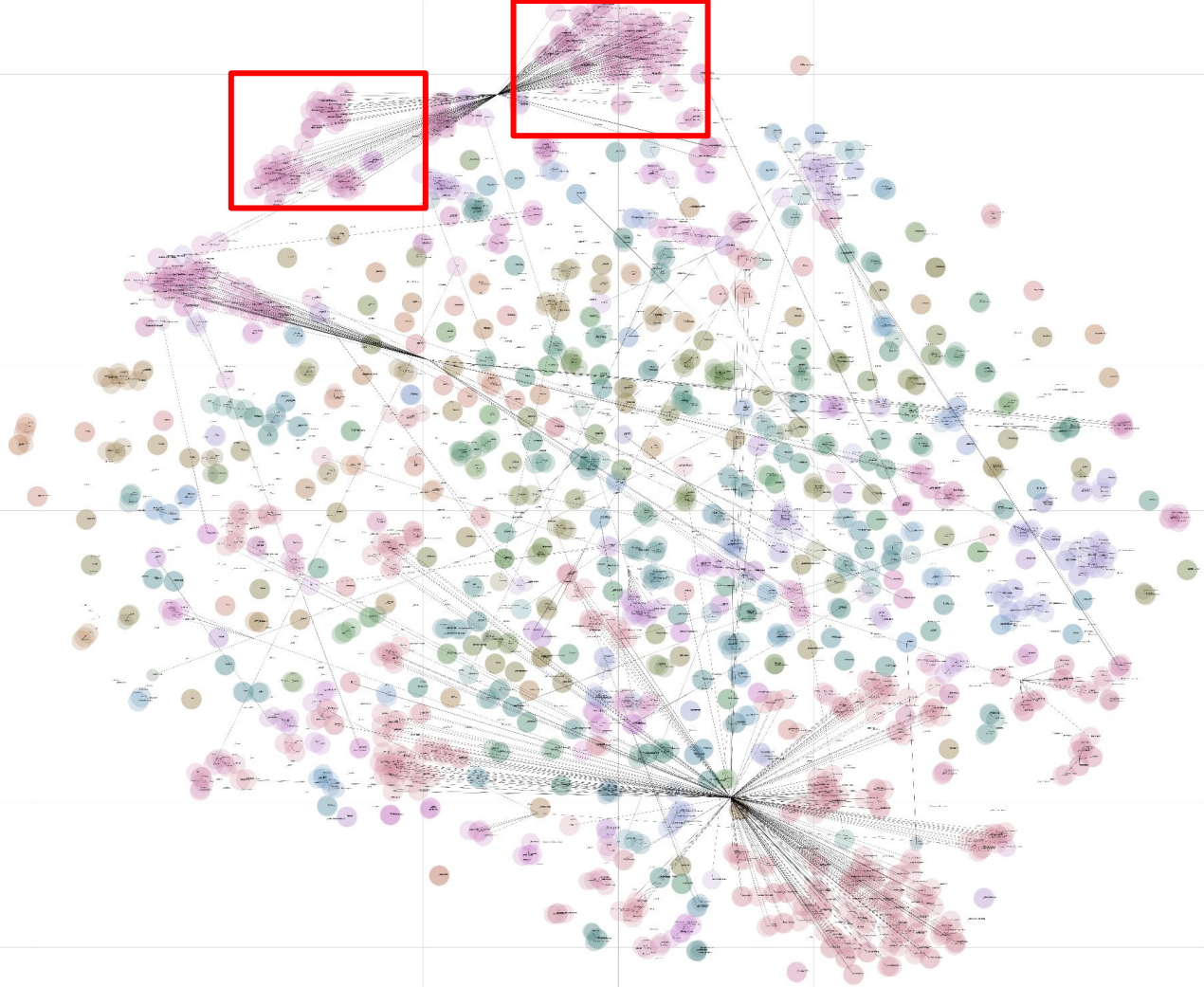
2.3. All labels and Outliers



431 labels, 678 outliers:

Some data points do not find any nearby matches to form clusters.

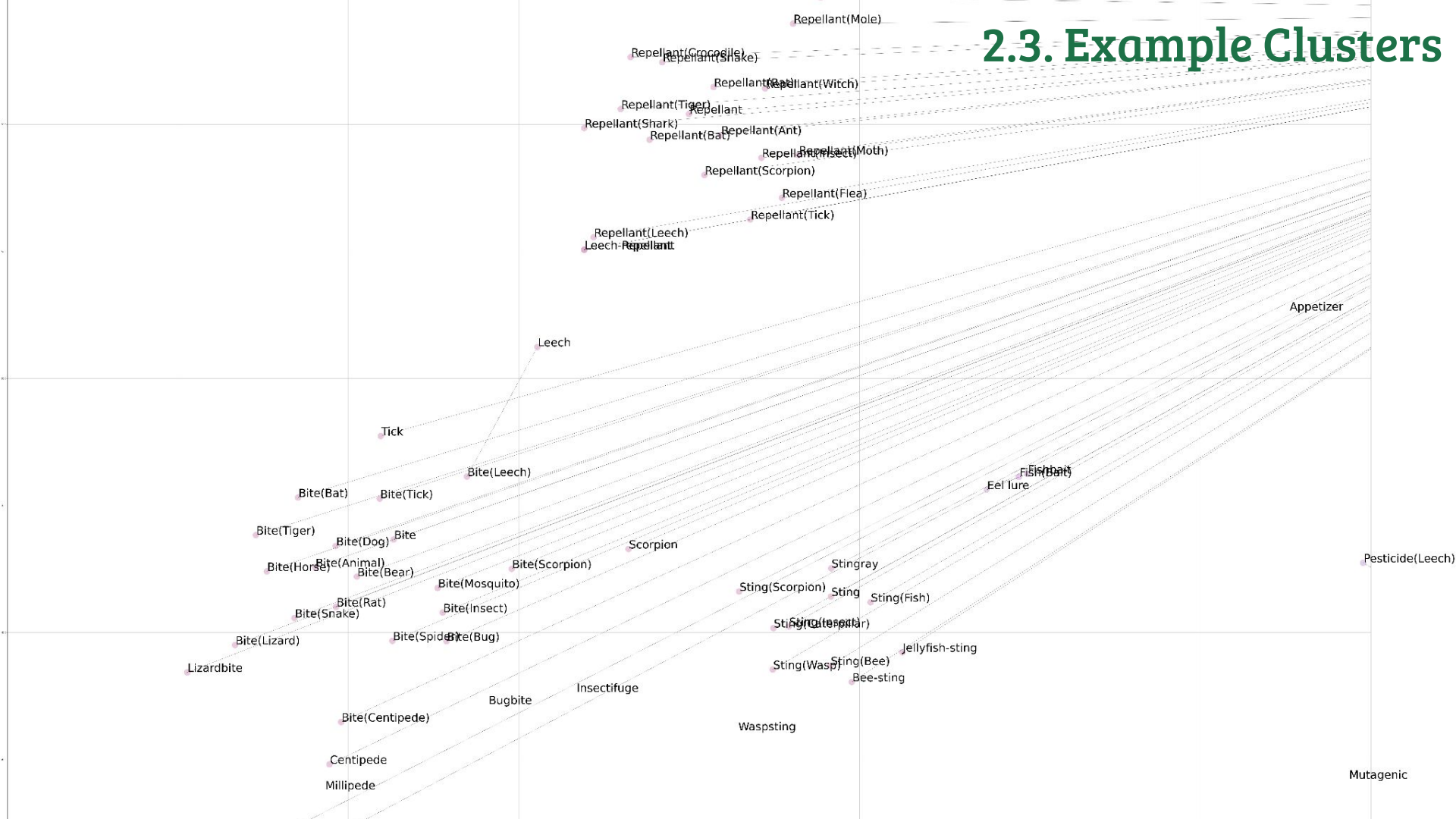
2.3. Full mapping



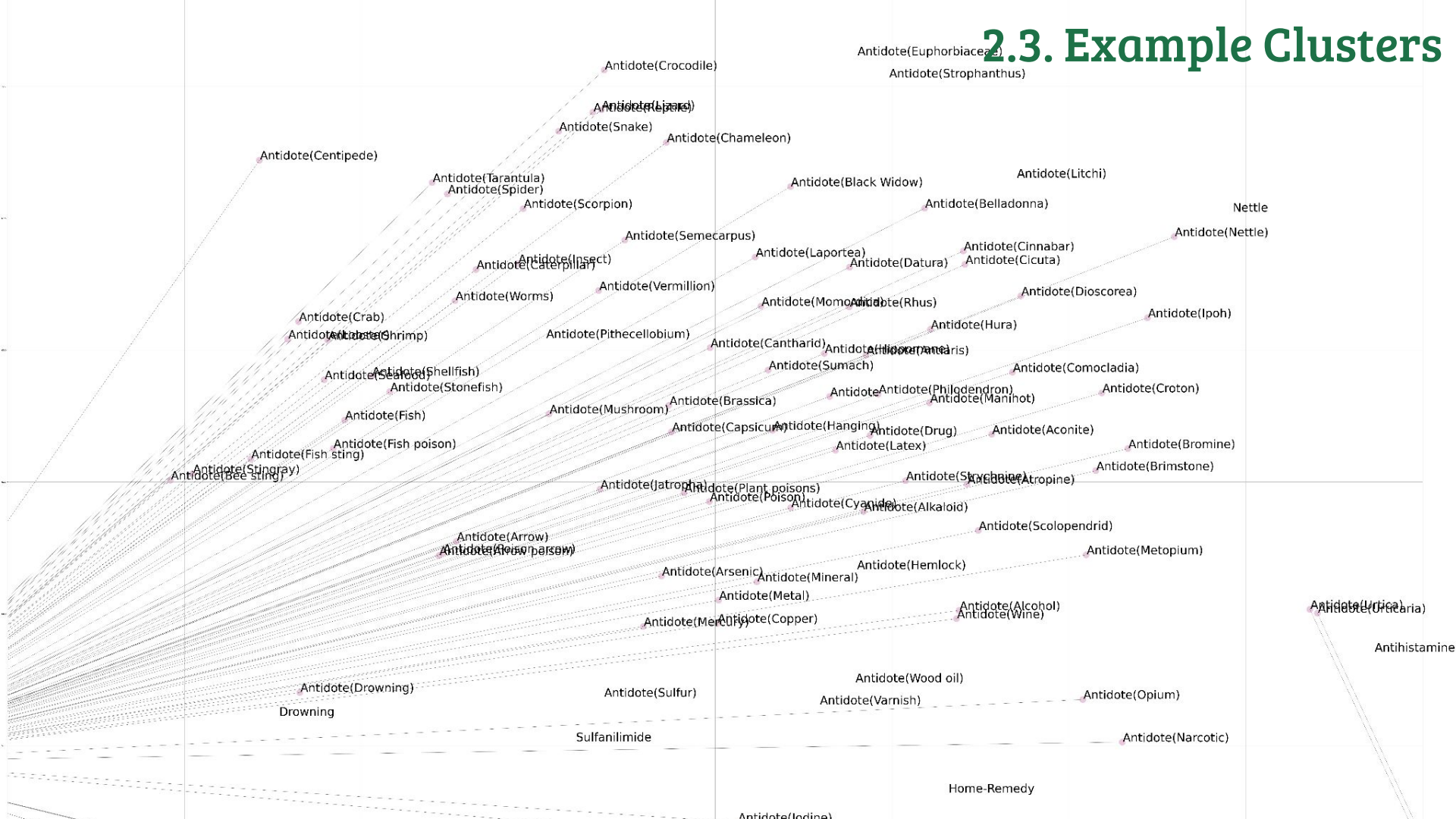
Dimensional Complexity:

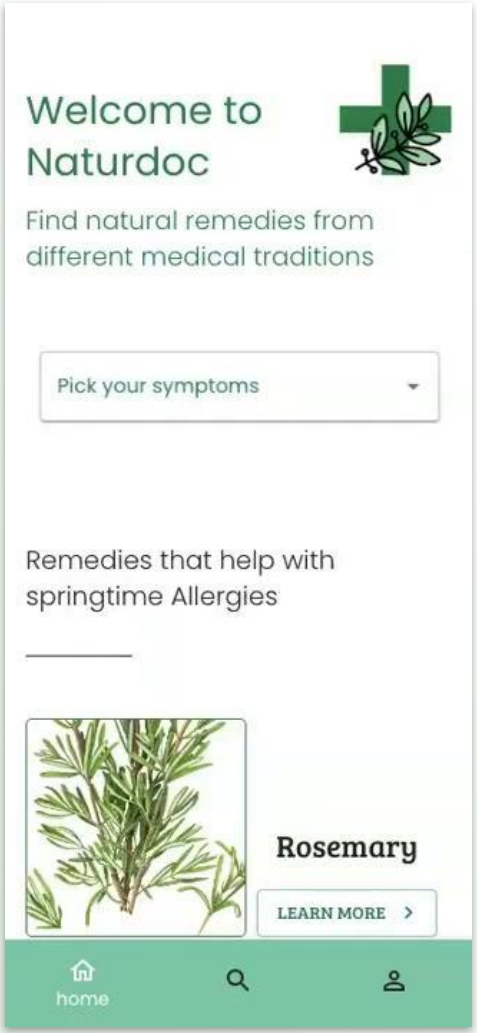
As a general reminder, this visualisation is merely a projection and its distances can be quite misleading.

2.3. Example Clusters



2.3. Example Clusters





3. Reviews: adding user-generated content

Behind the scenes:

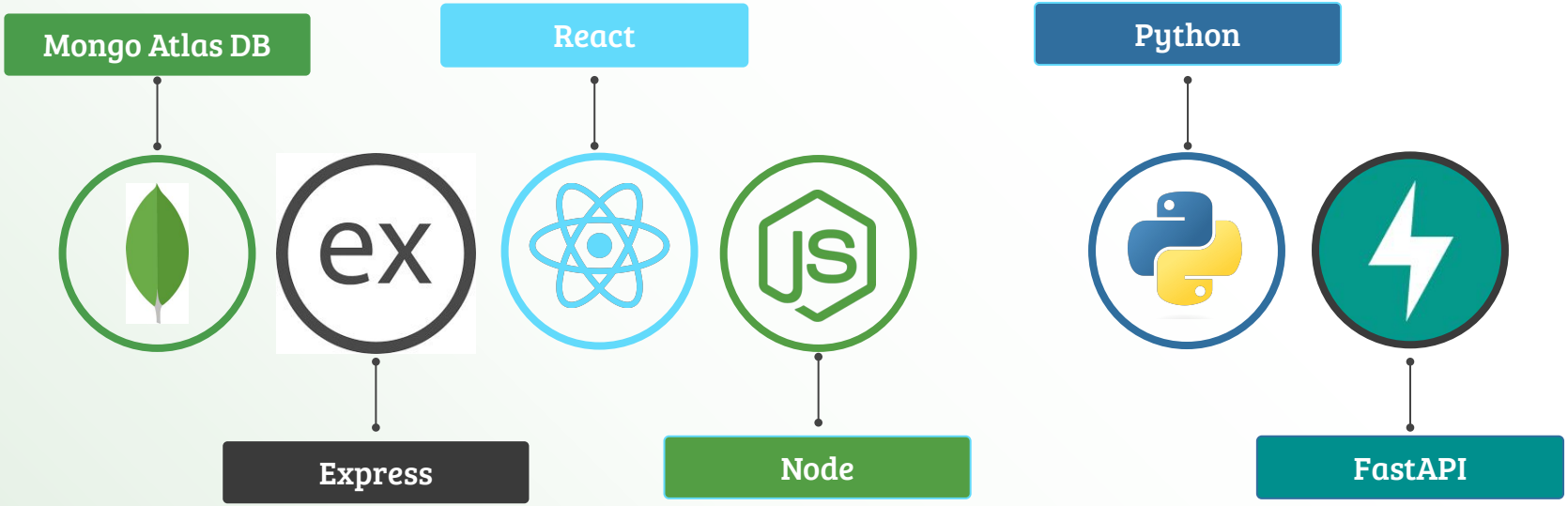
- ✔ Built with React v18.2
- ✔ MUI Framework
- ✔ **20** components for **3** pages
- ✔ Motion design to animate pages
- ✔ Creation of a theme.js file to centralize design system
- ✔ Implementation of loading skeletons

2 API connections to python server

5 API endpoints

- ✔ Built with express and a mongoDB Atlas database
- ✔ **4** DB models for
  - user
  - remedies
  - ratings
  - symptoms
- ✔ Rating system (One-to-one and one-to-many relationships)
- ✔ User authentication endpoints (signup, login and logout)

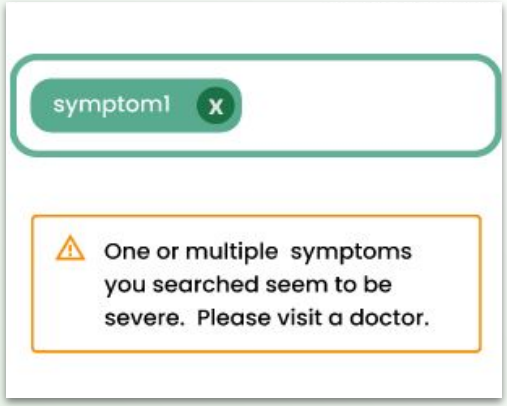
Tech Stack

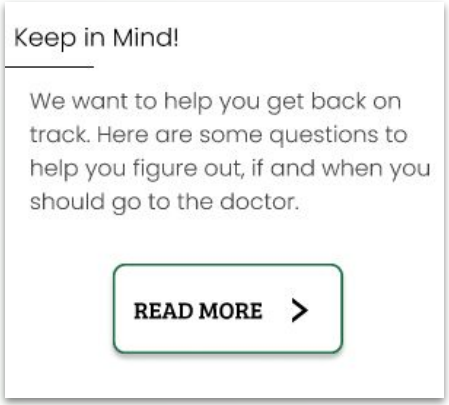


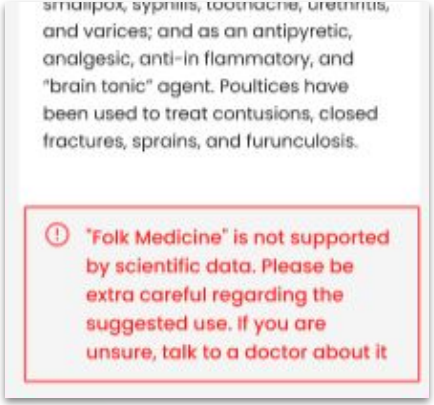
4. UX

- Finding corporate design that looks trustful
- Keeping it simple and tasteful but also interesting and intuitive
- Taking enough time for brainstorming, considering different options and taking into account all the possible solutions to the problem
- Reaching a design that makes us proud and that we feel confident in sharing
- Always being very aware that we are dealing with medical information and that it should be handle with respect and clarity

UX: Risk management







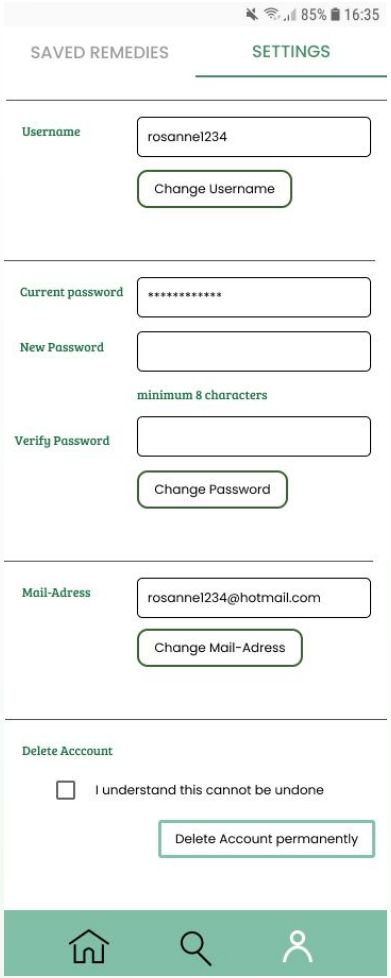
5. Vision

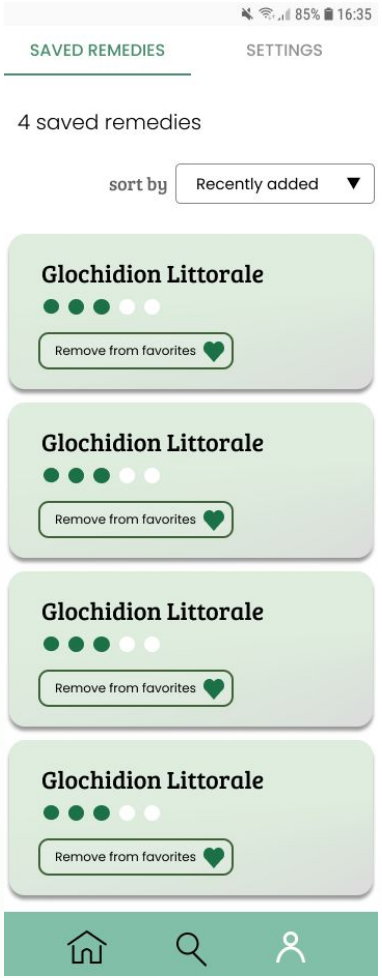
What's next ?

- UX: Allow users to sort the listed Remedies;
  custom content: Pictures, Actual medical advice,
- DS : Personalized results based on user preferences, search history and similar; expand our data sources; classification algorithms
- WD: storing personal user data - login, favorite remedies, user profile

Vision

UX: Implementing Personal Page-Designs into the Application





Retrospective

Most of MVP Goals achieved, although a lot of parallel work between teams and learning at the same time

The guidance and input from our team and mentors made all the difference in the success of the project

Person that pitched the project not with us at the beginning to lead on the concept

many dependencies between the different tracks at the beginning

different timetables of different team members / managing full time job and project was challenging

Quality of available data and the need for custom content

Highlight: Deciding on clear MVPs and stretch goals

data connection between FE - BE - DS was definitely a highlight

No one dropped out!

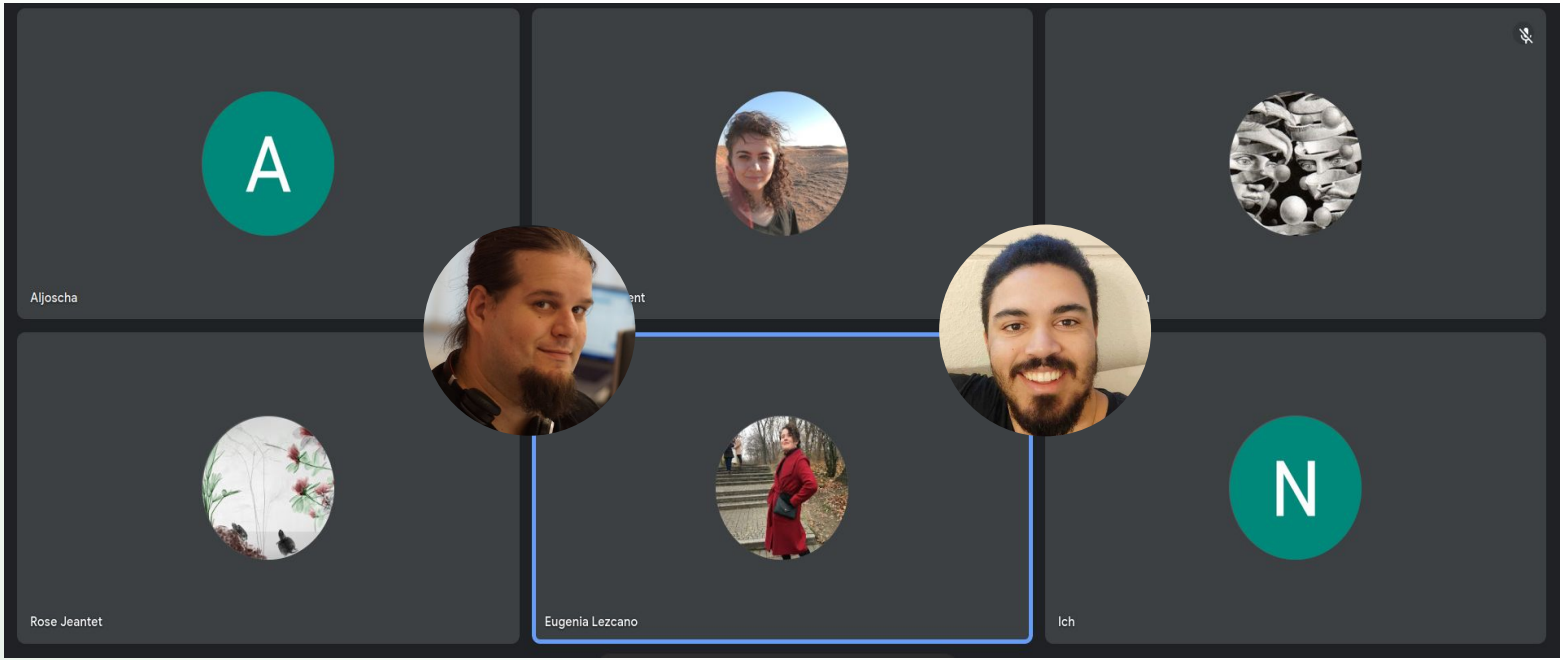
Pytrends limitations

Highlight: Finding a color-combination that does not look cheap

Amazing support from our mentors

we learned so so much!

The Team



DS: Anna and Aljoscha
WD: Rose and Christina
UX: Eugenia and Luzie

Mentors: Rafael Saraiva and Soma Hargitai

Questions