PollenPal
Your Ally Against Allergies

Nina Belusa - Noémie Schweizer - Elisa Billard

May 30, 2024

We built a website responding to several needs of people with pollen allergies and health care organizations in need of syndromic surveillance data. The main feature of the website is the collection and visualization of information on the localization and intensity of pollen allergy symptoms from users, through the EpiCollect app. The website also allows users to be alerted, via a mailing list, of pollen intensity in a designated area.

1 Introduction

We are currently witnessing a rise in pollen allergies in the world, partly due to climate change [1]. Therefore, we decided to tackle this issue to help people suffering from allergies and to collect data for surveillance. In the case of the functional deployment and usage of the website, the map of the allergy symptoms helps to know where there is a particularly high concentration of a certain pollen. This way the user can adapt their destination or timing, or take precautions when going outside. Moreover, the data collected is very precious because it contains first hand information on where people experience allergies and at which intensity. Governments or health care organizations will then have a better surveillance of the evolution in time and space of allergies, allowing them to take measures.

2 Prototype

Our website is composed of several pages, with a side bar allowing the user to browse through these pages. The Homepage gives an overview of what our website is for, by providing a short description about its goals, what the user can do and what information they can get with it.

2.1 Map of pollen symptoms

The page “Map of pollen symptoms” contains a map showing where users with pollen allergy have experienced allergy symptoms (Figure 1). The data comes from a form, that users can fill out with information about their experience of allergy symptoms, as well as some randomly generated points to show in the prototype. The user can select specific types of pollen or view all pollen together. The colors of the points indicate the severity of the symptoms experienced, with red being severe, orange being moderate and yellow being mild symptoms. The user can also click a button to show their own location on the map, and if they grant permission to share their location with the website, their location shows up on the map as a blue dot. The map will also be centered on their location. Finally, on the page is a bar plot, which shows the most common pollen allergy symptoms, according to our users. This will allow users to understand more about allergies.

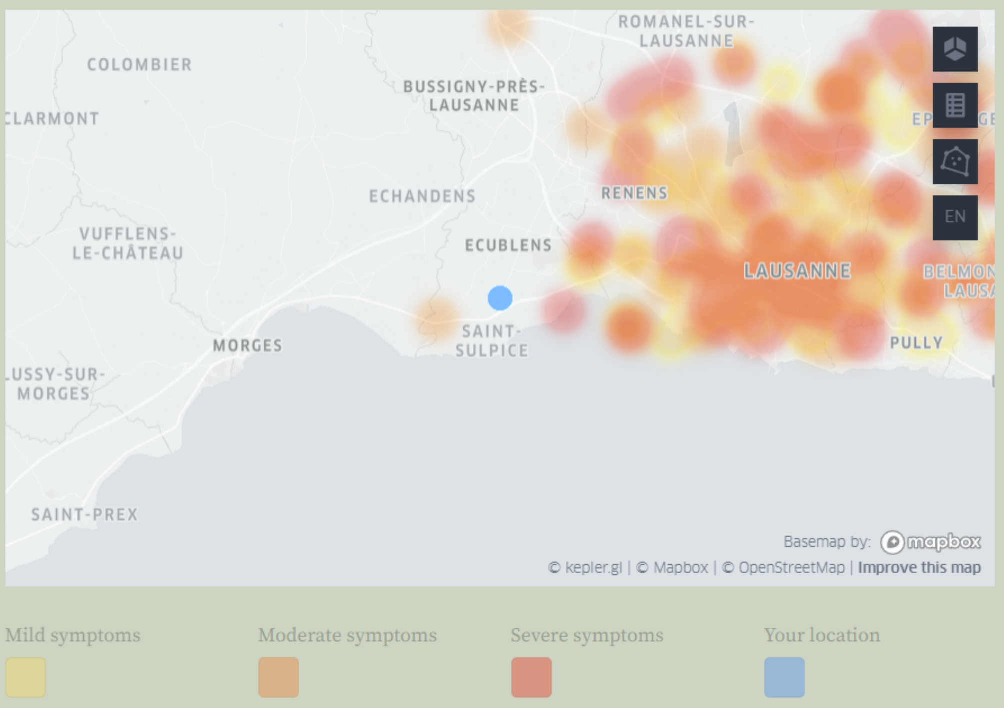


Figure 1: Map of pollen symptoms

2.2 Enter your symptoms

The page ”Enter your symptoms” redirects to the EpiCollect form we use to collect data about pollen allergies and symptoms from users. Since the form is not accessible online but only with the EpiCollect app, the page explains the few steps that users need to perform to file it: a link to download the app, what project to search and the information that will be asked of participants. The data collected will then show up in the ”Map of pollen symptoms” page (see section 2.1).

2.3 Sign up for alerts

The page ”Sign up for alerts” is the section that allows the user to subscribe to a mailing list alerting high pollen concentrations. One can enter an email address which they want to be alerted with along with the location(s) (a location can be any canton in Switzerland since we decided to focus this website for Switzerland). There is the possibility to enter several locations, and also several types of pollen from those that can be present in Switzerland. The subscriber also enters their pollen sensitivity to adapt the forecast depending on the pollen index or concentration. Subscribing to the mailing list results in receiving a confirmation email with the recapitulation of the places and types of pollen for which to be alerted. Then when we forecast high concentrations of their chosen pollen types in the location they entered, an email will be sent so that they know when they might experience symptoms and have to take precautions. There is also a possibility to unsubscribe to email alerts. The user just has to enter the email address with which they subscribed in the proper text field and they will be removed from the list of users wanting alerts.

2.4 Information about allergies

This page details some general information about allergies such as what allergies are, common symptoms, and information about pollen. An important part of this page is also a list of tips to manage pollen allergy symptoms and better tolerate the season of allergies.

3 Methods

3.1 Streamlit

Since we decided to develop a website for this project, we needed to find a way to implement and create it. After looking for the best alternative for website deployment, we have chosen to use Streamlit. This open-source python framework allows to easily create and share web apps. Easy to use and with lots of implemented functionalities useful for websites such as interactive widgets, possibilities to link APIs, etc, this library seemed like an ideal choice for creating our interactive web application directly from python scripts.

3.2 Epicollect

A central part of our project is data collection from users. We need to collect their location, the type of pollen they are allergic to, and other information about their allergy. For this, we decided to use the mobile data-gathering platform Epicollect. On the Epicollect App, we set up a project and made a form with questions which users can fill out. We included the following questions:

- What kind of pollen are you allergic to?
- On which date did you experience hay fever symptoms?
- At what time did you experience hay fever symptoms?

- What is your current location or the location where you experienced hay fever symptoms?
- What symptoms did you experience?
- How would you rate the severity of your symptoms?

The entries from the last seven days are retrieved through the Epicollect API and converted to a pandas DataFrame for further use on the website.

3.3 Kepler.gl map

The map displaying where people experienced pollen symptoms was created using Keppler.gl. Keppler.gl was chosen because it can be used to create many different types of customizable maps with the possibility to integrate them with Streamlit. We are using the `keplergl package` for python to generate the map and the `streamlit_keplergl` package to integrate the map with streamlit. A Kepler.gl config is used to specify the style of the map, the initial state of the map and the styles of each layer. The data is displayed as a heatmap, and blends together when several points are close together. The latitude and longitude for each point on the map comes from the Epicollect entries, combined with the generated data, and the chosen severity of symptoms determines the color. Depending on which pollen types are chosen on the website (via streamlit's `st.checkbox`), the data shown on the map is filtered to match the chosen pollen types. If the user wishes to, they can also display their location on the map. This is implemented using the package `streamlit_geolocation`. The initial map state is then modified to start at the location of the user.

3.4 Chart of pollen symptoms

In our Epicollect form, we also asked users what type of symptoms they were experiencing. This data is collected and displayed in a bar plot. The plot is generated using the `altair` package, and displayed in streamlit using `st.altair_chart`.

3.5 Google Pollen API

To get real time pollen data, we have used the Google Pollen API. First, we wanted to use their pollen intensity heatmap tiles that could be displayed on top of Google Maps. However, this ended up being too costly because the API would be called hundreds of times when moving on a map. We ended up using the forecast API that enables us to send a mailing list every day according to the severity of the pollen presence (see 3.6). We call the API for each canton in Switzerland where people have subscribed. This amount of calls is more manageable because we have 200 USD of credit given from Google, however too many calls can't be made in a short amount of time so it is possible to comment this part when we don't want to call the API each time the website is loaded.

3.6 Mailing List

The mailing list works with a STMP Outlook server, that sends an email from an email address we created for the website: pollenpal.alerts@hotmail.com. A problem encountered is that microsoft sometimes blocks emails sending when too many have been sent by the website, so we need to connect to the messaging account to unblock it, but it still works for quite a long time.

We also had to find a way to keep users' data to be able to send them the alert emails, so we decided to create a database and used a class we created, the class `User` to represent each person that subscribes. When a user's subscription is confirmed, their allergy profile is added to the database, so we can keep track of all subscribers and send emails to them when necessary. We used for that the Cloud *Firestore* to host our database, which is built on the Google Cloud infrastructure. We used it since it works well with Streamlit and doesn't need a server to keep users' information. Thanks to that, each time we forecast high pollen concentrations we can get the users that asked to be alerted back from this database and send the mail alerts.

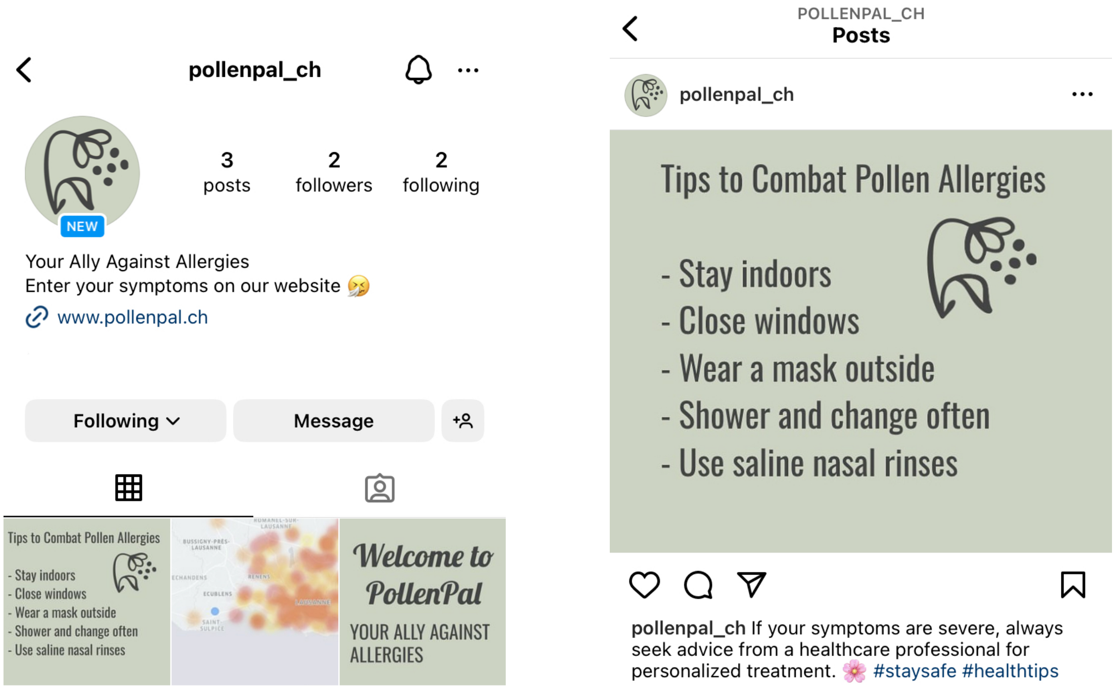
For the forecast, we planned to look at which types of pollen have a high index (in the Universal Pollen Index or UPI scale) which corresponds to high risk of allergic reactions each day, then send an email to the people that need it. The forecast is done with the Google forecast API described in the section above. We couldn't automatize this daily pollen forecast, but we have a specific function called `refresh_updates()` that runs the API and sends emails accordingly when running the website locally (it is for now called each time the "Sign up for alerts" page is loaded since it has to be done manually).

To implement all the tasks, we coded functions in a file named `data_functions.py`, which relates the display in Streamlit to the handling of data (those functions are used in the "Sign up for alerts" page). The file contains basic utility functions, functions that link to the *Firestore* database to get all the users, add or remove subscribers. It also contains the function that sends the alert email and several others used for the forecasting and the mails: we have to get the cantons we want to forecast pollen concentration in, call the Google API for forecast in all these locations and extract the types of pollen that have a high UPI. Then we look at the users again and select those who chose location(s) that are concerned for pollen alerts for this day. The refresh function's task is then to send an email to all these concerned users.

4 User Engagement Strategy

The crucial aspect to make our website really useful is the amount of people that will enter their data. Therefore, we have thought of different levers that could be used to convince people suffering from allergies to contribute to the website. One first aspect is the collaboration with governmental health organizations and medical professionals to promote allergy awareness, potentially through public health campaigns. Healthcare professionals could also directly talk to their patients with allergies about the website which would be a powerful way to get more users. It is however going to be challenging to get those professionals to trust us enough to promote our website. The collaboration with the healthcare system could also allow to integrate our valuable data with existing allergy surveillance systems, and enhance the accuracy and timeliness of allergy surveillance efforts.

To promote our website we have created an Instagram page and visuals (see Fig. 2), that you can access with this link: Instagram page . Having a strong identity, logo and slogan can help us become more well known and therefore trustworthy. The community can then interact with the posts and even send messages to ask questions. We think this could really help increasing the engagement of the user and the number of people participating.



Your Ally Against Allergies
Enter your symptoms on our website 🤧
www.pollenpal.ch

Tips to Combat Pollen Allergies

- Stay indoors
- Close windows
- Wear a mask outside
- Shower and change often
- Use saline nasal rinses

pollenpal_ch If your symptoms are severe, always seek advice from a healthcare professional for personalized treatment. 🌸 #staysafe #healthtips

(a) PollenPal Instagram page

(b) Example of Instagram post

Figure 2: Instagram page

5 Learnings

While working on this project, we have learned a lot of valuable things. First of all, it has been easier to build a website than we expected, based on our limited experience with this type of programming. Streamlit helped make it simpler, and we were happy to find that there were many packages that could help us implement the features we wanted for our website. Not that it was always easy to implement, but there was much help to find on the internet and from AI chatbots. We also learned that APIs can be very costly, so they need to be used wisely and it is worth considering if an expensive API is really necessary. We have also learned that developing a cool website is not enough if no one wants to use it. Especially for our website, which uses user input, it is not useful if no one is making entries. Finally, we have learned a lot of new programming skills and website development terms, which we might be able to use in future projects.

References

[1] Charles W. Schmidt. “Pollen Overload: Seasonal Allergies in a Changing Climate”. In: *Environmental Health Perspectives* 124.4 (Apr. 2016), A70–A75. DOI: 10.1289/ehp.124-A70.