

Date:05/10/2023

# **Support Document**

## **Seeing Asthma**



## Team Information

**Team Name:** Healthy Coder

**Team Number:** TA 09

**Team Member:** Our team consists of graduating postgraduate students from Monash University in various disciplines. Listed below are our contact details and the division of responsibilities of each member in product design and development. We look forward to communicating with you if you have any questions about our products.

Role	Name	LinkedIn
Business Analysis	Kaijia Yu	<a href="#">Kaijia Yu</a>
Full Stack Engineer	Ridong Jiang	<a href="#">Ridong Jiang</a>
Front-end Engineer	Shiyu Wu	<a href="#">Shiyu Wu</a>
Data specialists	Navrattan Singh Dhillon	<a href="#">Navrattan Dhillon</a>
Cyber Security Specialist	Sonia Lakhani	<a href="#">Sonia Lakhani</a>

## Table of Context

1.Product Overview.....	3
2. Start Up Requirements.....	3
3.1 Front-End.....	4
3.2 Back-End.....	5
4. Security.....	9
5. Backup and Recovery.....	9

# 1.Product Overview

Our project "Seeing Asthma" is based on research into the many allergens that are widespread in Australia and pose a serious threat to the health of children with asthma. The main objective of this project is to provide a platform for parents of asthmatic children to learn more effectively and utilise features to better manage their child's asthma through our website.

The Seeing Asthma website is an integrated platform with not only comprehensive asthma knowledge but also many new features to help you better manage your child's asthma. These include a "picture recognition" feature that helps parents identify asthma triggers regardless of location, time, or device. Recommendations for asthma-friendly activities and diets. In addition, we have developed animations and games to promote self-protection awareness, so that children can be taught how to prevent asthma.

## 2. Start Up Requirements

### 2.1 Software Requirements

- You can manage the website on all Windows, MAC OS and Linux systems.
- Visual Studio code 1.82.3 or later is recommended for managing the frontend source code of the website.
- Node.js 18.18.0 LTS version or later should be installed for further development on Next.js framework and run the npm command on the terminal which stands for Node package management.
- Vercel is used for deployment management of the website. Please refer to Vercel official website for a more comprehensive guide: <https://vercel.com/docs>. Vercel Token: g1VvCCofUNjfjDcsZRSLna91 (This token will expire on October 4, 2024)

### 2.2 Data Requirements

All the data required to run the website is already stored in a MySQL database which is hosted on an AWS server. The details to connect to the AWS are below:

Access Host URL: database-1.cbycmzyztg27.us-east-1.rds.amazonaws.com

Port: 3306

Username: cianjiang597

Password: Fk6DJEwzPY7x

Database Name: MAINDB

The support team does not need to manually download the data files and create a new database as the existing database can be used. If the support team wants to migrate the database into their own personal database server, then kindly refer to the 'Database Migration' Section.

## 3. Deployment/Installation

### 3.1 Front-End

1. The frontend is developed with Next.js 13.4.19. [Next.js](#) is a React framework that gives developers a block to build the web application. In order to use Next.js, at first, please make sure you have installed [Node.js](#) 18.18.0 LTS version or later. For further check for versions and packages, please refer to package.json file under the root directory of the project folder.
2. About how to run the product (Simply do copy and paste of the highlighted words)
  - Download the zip from [Github](#), choose **Shiyu** branch
  - Unzip the file and open it in any code editor you like. We recommend visual studio code.
  - Add a file called **.env.local** to the root directory of the project folder in order to load the data visualisation in the homepage.
  - Add the following code to **.env.local**

```
MYSQL_HOST=database-1.cbymzytg27.us-east-1.rds.amazonaws.com
```

```
MYSQL_USER=cianjiang597
```

```
MYSQL_PASSWORD=Fk6DJEwzPY7x
```

```
MYSQL_DB=MAINDB
```

```
GOOGLE_CLOUD_VISION_API_KEY =  
AIzaSyDSixvrSvxTKMKdd0rYO3ogivqSGdlqoRI
```

```
API_KEY = 'i23a3ushd123egiqwuoasqw2378qer12132'
```

- Run a new terminal in the root directory of the project folder
- Read the README.md file, under the ##install tag, copy the code **npm i @chakra-ui/react @emotion/react @emotion/styled framer-motion**

**react-icons @chakra-ui/icons react-player chart.js mysql axios** to install all packages.

- Run **npm run dev** to run the project
- Click on the link for example <http://localhost:3000> in the terminal to run the website on your localhost.

If you would like to make any change of the frontend, you have to

- Know about HTML, CSS, JavaScript
- Know about Next.js which is a React framework
- We also used Chakra-ui in our development including some TypeScript coding. Please refer the following links to check for the template: <https://chakra-ui.com/docs/components>, <https://chakra-templates.dev/>

### 3.2 Back-End

The backend was developed using Python. The code was stored in [Github](#), choose **python-api** branch to download the zip. Please make sure you run the python code using Python 3.9.7 version. Then based on requirements.txt to install all packages.

The contents of requirements.txt are as follows:

```
mysql-connector == 2.2.9
pillow == 10.0.0
flask == 2.2.5
flask_restx == 1.1.0
pydantic == 2.1.1
flask_sqlalchemy == 3.0.5
Flask_Cors == 4.0.0
matplotlib == 3.7.2
boto3 == 1.28.23
requests~=2.31.0
```

For security purposes, the service runs on port 5000 of the local host by default, and then uses Nginx to reverse proxy the domain name of the backend api with port 5000 of the local host.

The backend service runs on AWS instances.

### 3. Extract Transform & Load (ETL) Process

#### 3.1 Database Extraction

The data being used in this project is open data. The open sourced datasets that were used in this project are listed in the table below:

Seeing Asthma Open Data Source Table				
Names	Frequency of source updates	Frequency of iteration	Granularity	Copyright/ licensing details
National asthma indicators <a href="#">XLSX</a>	Posted on 30 June, 2023. Not Provided	Quarterly	Number of Hospitalisations by age from 2010-2020 in Australia	Creative Commons Attribution 4.0 International license <a href="#">License Link</a>
Dogs and Cats Online Data 2021-2022 <a href="#">ZIP</a> <a href="#">containing XLSX</a>	Annually	Annually	Popular Dog and Cat Breeds in South Australia from 2021-2022	Creative Commons Attribution 4.0 International license <a href="#">License Link</a>

#### 3.2 Data Pre-Processing (Transform)

There are two files that clean and transform the open data. The details for it are provided below:

- National Asthma Indicators:

The cleaning was done using [R programming language](#). The software used for this task was [RStudio](#). The steps required are given below:

1. Download and install RStudio by following [these steps](#).
2. Open the script 'Cleaning Data.R' to clean data by following these [steps](#). The file path is given below:

Database Generation/ Scripts/ Cleaning Data.R

3. Download the open data file into the home directory of RStudio. This can be done by following the steps mentioned [here](#).
4. [Run](#) the script, and all the data files would be generated in the home directory.

- Dogs and Cats Online Data:

The second open sourced data file was pre-processed using [Python programming language](#). [Google Colaboratory](#) software was used to clean the data file. The steps to clean the data are given below:

1. Upload the python file 'IE\_data\_cleaning.ipynb' to colab by these [steps](#). The file path is given below:

Database Generation/ Scripts/ IE\_data\_cleaning.ipynb

2. Upload the open data file to colab as well by following these [steps](#).
3. Run the python script using the steps mentioned [here](#). The cleaned data files would be downloaded automatically.

These cleaned files were further updated by our team using data collected from various government agencies websites.

All the final data files for other database tables used in this project are also stored in the Database folder. The path to these files are given below:

Database Generation/ Cleaned Excel Files/

### 3.3 Loading Data

The data files can be loaded into the MySQL database by following these steps:

1. [Download](#) and install the DataGrip tool.
2. Link the database hosted in AWS server to the DataGrip tool by following these [steps](#). The server details are provided in the Data Requirements section above.
3. Run the sql script 'seeing\_asthma.sql' by following these [steps](#). This will generate the tables to store the data.
4. Finally import the data from excel file (path is given above), into the database by following these [steps](#).

## 4. Security

To ensure the security of our application we first safeguard our system using SSL/TLS certificates and firewalls to monitor incoming and outgoing network traffic. Since we have cloudflare CDN before accessing the application we also limit the incoming and outgoing traffic in the CDN firewall.

The second part is safeguarding against potential attacks on the frontend such as SQL injections and XSS attacks. We ensure that we enhance the security of our application by using countermeasures against such attacks in the frontend code for example parameterized queries and input validation and sanitization to prevent such attacks. To ensure continuous safety of the system we must also keep updating the javascript framework next.js to the latest version and monitoring for any potential vulnerabilities.

Since we are using AWS EC2 instances to host the backend we make sure to leverage the security provided by AWS and also enable log monitoring using the cloudwatch functionality in AWS and monitor for any suspicious activity and unusual jumps in request signalling a potential DDOS attack.

Lastly we perform penetration testing of the system to monitor for any potential cyber threats of the system which can be mitigated.

## 5. Backup and Recovery

### 5.1 Database

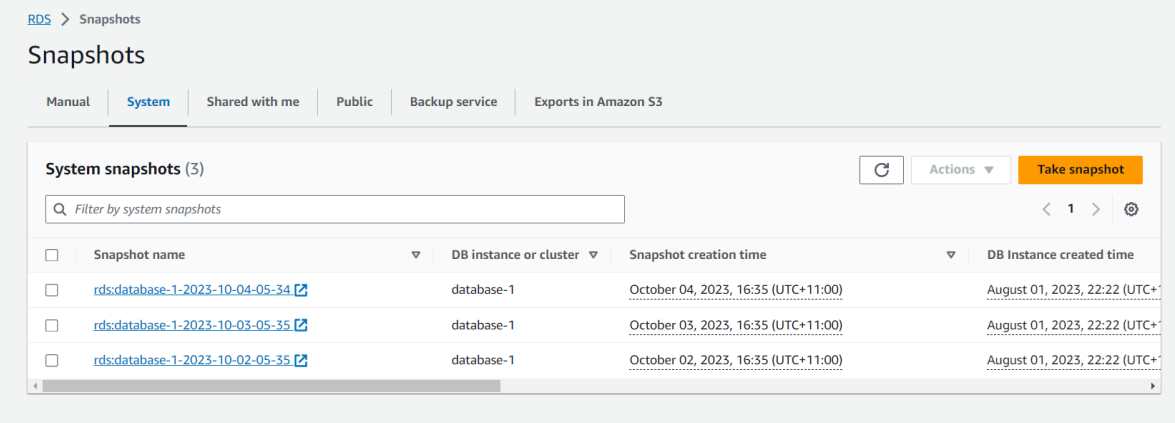
#### A. Backup

The MySQL database is hosted on the host server Amazon Web Services (AWS), which has a free option which automatically backs up the database to the cloud once a day. This feature enables us to recover the database from the day before in case something happens which results in a loss of data. The snapshot provided below shows the backup feature of AWS:

Backup		
Automated backups Enabled (1 Day)	Latest restore time October 05, 2023, 22:00 (UTC+11:00)	Replicate to Region -
Copy tags to snapshots Enabled	Backup window 05:27-05:57 UTC (GMT)	Replicated automated backup -
Backup target AWS Cloud (US East (N. Virginia))		



Host server will automatically create database snapshots every day and retain the latest three-day snapshot version. If the support team believes that a certain version of the database has great backup value, they can also go to the Snapshots page of the Amazon RDS page to create Manual snapshots.



The screenshot shows the Amazon RDS Snapshots page. At the top, there's a breadcrumb 'RDS > Snapshots' and a title 'Snapshots'. Below the title are tabs: 'Manual', 'System' (selected), 'Shared with me', 'Public', 'Backup service', and 'Exports in Amazon S3'. The 'System snapshots (3)' section is active, showing a table of snapshots. The table has columns for 'Snapshot name', 'DB instance or cluster', 'Snapshot creation time', and 'DB Instance created time'. There are three snapshots listed, all for 'database-1'. Each snapshot name is a link to the snapshot details. The table also includes a search bar, a refresh button, an 'Actions' dropdown, and a 'Take snapshot' button.

Snapshot name	DB instance or cluster	Snapshot creation time	DB Instance created time
<a href="#">rds:database-1-2023-10-04-05-34</a>	database-1	October 04, 2023, 16:35 (UTC+11:00)	August 01, 2023, 22:22 (UTC+11:00)
<a href="#">rds:database-1-2023-10-03-05-35</a>	database-1	October 03, 2023, 16:35 (UTC+11:00)	August 01, 2023, 22:22 (UTC+11:00)
<a href="#">rds:database-1-2023-10-02-05-35</a>	database-1	October 02, 2023, 16:35 (UTC+11:00)	August 01, 2023, 22:22 (UTC+11:00)

## B. Restore

If database data corruption occurs, the support team only needs to go to the Snapshots page of the Amazon RDS page to select an available Snapshot for rollback.

Details		
ARN	Option group	VPC
arn:aws:rds:us-east-1:494136951719:snapshot:rds:database-1-2023-10-04-05-34	default:mysql-8-0	vpc-0638
Instance/Cluster Name	Zone	Status
database-1	us-east-1d	Available
Master username	KMS key ID	Storage type
cianjiang597	arn:aws:kms:us-east-1:494136951719:key/e22cb886-910c-48c5-8b77-b3a3e713e7ef	General Purpose SSD (gp2)
DB snapshot name	Source region	DB storage
rds:database-1-2023-10-04-05-34	N/A	20 GiB
Snapshot type	Snapshot Creation Time	IOPS
automated	October 04, 2023, 16:35	-
		Storage throughput