**Software Requirements Specification (SRS) Document**

***Outdoor Air Pollution Monitoring in IIIT-H Campus*,**
**Team 30**
***Rishabh Daga, Anubhav Sharma, Monil Gokani, Shiva Prasad***

## Brief problem statement

Given the data out of the already installed sensors inside the IIIT-H campus, we will be making a model which will help the user base visualize the air quality standards. Through the initial plan, we will be working to make a web portal which will consist of a heat map describing the pollution levels in the campus premises along with some statistical data. For this we will be given live data from the pre-installed sensors.

## System requirements

- Django for back-end implementation.
- HTML, CSS, Bootstrap, JavaScript for UI,
- Python libraries like Matplotlib, Pandas and Numpy

## Users profile

There will be following types of end-users:

➔ Professors,

➔ Students (including all interns & full-time students),

➔ Non-teaching Faculty (administrators)

➔ Workers (including mess workers, cleaning staff, security guards)

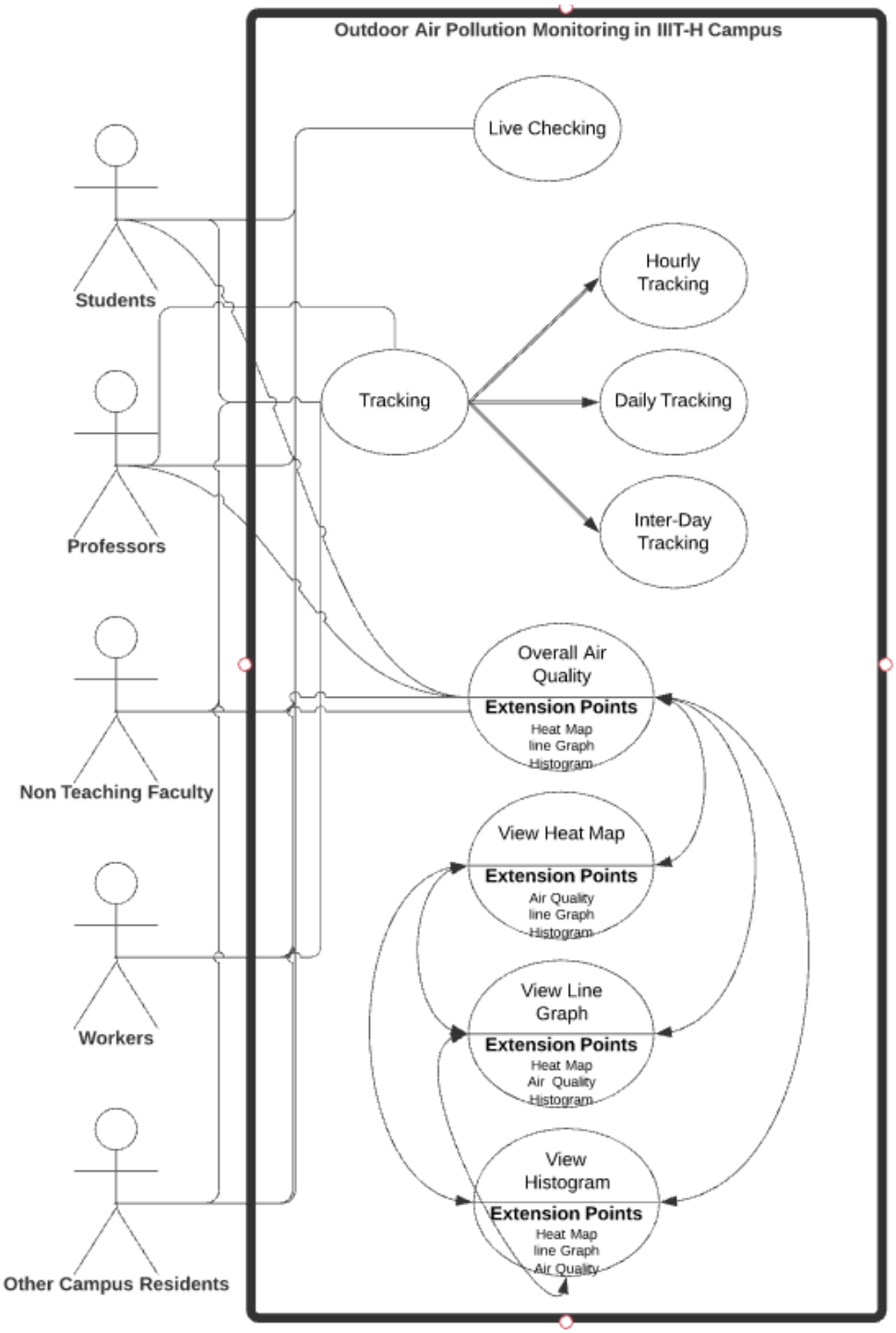
➔ Other campus residents(families of professors)

Our end-users might be very different but the UI of our model will be same for all as we are planning to make a map with all the data available in easy-to-read format.

## Feature requirements (described using use cases)

| No. | User Case Name | Description | Release |
|---|---|---|---|
| **1** | Live checking | Checking individual levels of pollutants(live data given by the pre-installed sensors, through a web portal,without any authentication) | R1 |
| 2 | Hourly Tracking | Checking pollutant levels on hourly basis where we will calculate the average of data per time unit. | R1 |
| 3 | Daily Tracking | Checking pollutant levels on a daily basis where we will calculate the average of data per time unit. | R1 |
| 4 | Inter-day Tracking | Individual pollutant mean over an inter-day cycle where the averages are collected over multiple days (a week or month) | R1 |
| 5 | Overall Air Quality | Checking air quality (with existing parameters) and by using certain statistical & mathematical concepts. | R1 |
| 6 | View Heat Map | Viewing the heat map of campus in accordance to the air quality where pollution level increase from blue to red. | R2 |
| 7 | View Line Graph | View live line graph of real time data of pollutant levels. | R2 |
| 8 | View Histogram | Display the data in a more informative way using overlapping histograms showing individual rise and fall of levels on a same place. | R2 |
| 9 | View Pollution levels for daily time slots. | Checking air quality and finding daily trends in a given time slot (say morning -11 to 2) | R2 |
| 10 | Detect correctness of sensors | On the basis of the data that we receive , we will try to find sharp ups and downs in the reading and will detect correctness of our sensors. | R2 |

## Use case diagram



**Use case description**

| | |
|---|---|
| **Use Case Number:** | UC-1 |
| **Use Case Name:** | Live Checking |
| **Overview:** | Checking individual levels of pollutants(live data given by the pre-installed sensors, through a web portal,without any authentication) |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | Data should be updated regularly. |
| **Flow:** | • User needs to go to the respective chart page.<br>• At first the data is rendered from the Thingspeak server.<br>• Then that data is converted (after cleaning) to visual line graph or histogram. |
| | Alternate Flows: If correct data is unavailable then extrapolation of data is required. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-2 |
| **Use Case Name:** | Hourly Tracking |
| **Overview:** | Checking pollutant levels on hourly basis where we will calculate the average of data per time unit. |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | Data should be updated regularly. |
| **Flow:** | • User needs to go to the respective chart page. |

| | |
|---|---|
| | • At first the data is rendered from the Thingspeak server.<br>• Then that data is converted (after cleaning) to visual line graph or histogram. |
| | Alternate Flows: If correct data is unavailable then extrapolation of data is required. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-3 |
| **Use Case Name:** | Daily Tracking |
| **Overview:** | Checking pollutant levels on a daily basis where we will calculate the average of data per time unit. |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | No such pre condition. |
| **Flow:** | • User needs to go to the respective chart page.<br>• At first the data is rendered from the Thingspeak server.<br>• Then that data is converted (after cleaning) to visual line graph or histogram. |
| | Alternate Flows: No alternate flow needed. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-4 |
| **Use Case Name:** | Inter-day Tracking |
| **Overview:** | Individual pollutant mean over an inter-day cycle where the averages are collected over multiple days (a week or month) |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | No such pre condition. |
| **Flow:** | • User needs to go to the respective chart page. |

| | |
|---|---|
| | • At first the data is rendered from the Thingspeak server.<br>• Then that data is converted (after cleaning) to visual line graph or histogram. |
| | Alternate Flows: No alternate flow needed. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-5 |
| **Use Case Name:** | Overall Air Quality |
| **Overview:** | Checking air quality (with existing parameters) and by using certain statistical & mathematical concepts. |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | No such pre condition (would be visible on each page , in different formats) |
| **Flow:** | Mainly , after opening the webpage , the user will just have to look on each page for the index displaying the overall air quality (using Indian AQI) |
| | Alternate Flows: No alternate flow needed. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-6 |
| **Use Case Name:** | View Heat Map |
| **Overview:** | Viewing the heat map of campus in accordance to the air quality where pollution level increase from blue to red. |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | Hourly data is needed (as it will change its condition after each hour) |
| **Flow:** | It is situated on the home page of our webapp . So if you are on other page , you will just need to click the home button |
| | Alternate Flows: No alternate flow needed. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-7 |
| **Use Case Name:** | View Line graph |
| **Overview:** | View live line graph of real time data of pollutant levels. |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | • Live data is needed.<br>• Data will need to be cleaned and calibrated. |
| **Flow:** | • User needs to go to the respective chart page of Line Graph.<br>• At first the data is rendered from the Thingspeak server.<br>• Then that data is converted (after cleaning) to visual line graph . |
| | Alternate Flows: No alternate flow needed. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-8 |
| **Use Case Name:** | View Histogram |
| **Overview:** | Display the data in a more informative way using overlapping histograms showing individual rise and fall of levels on a same place. |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | • Hourly, Daily and Weekly averages of data.<br>• Data will need to be cleaned and calibrated. |
| **Flow:** | • User needs to go to the respective chart page of Histogram.<br>• At first the data is rendered from the Thingspeak server.<br>• Then that data is converted (after cleaning) to histogram. |
| | Alternate Flows: No alternate flow needed. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-9 |
| **Use Case Name:** | View Pollution levels for daily time slots. |
| **Overview:** | Checking air quality and finding daily trends in a given time slot (say morning - 11 to 2). |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents |
| **Pre condition:** | No such pre condition. |
| **Flow:** | • User needs to go to the respective chart page of stats page.<br>• At first the data is rendered from the Thingspeak server.<br>• Then those readings are cleaned and calculated by averages for the given time slot. |
| | Alternate Flows: No alternate flow needed. |
| **Post Condition:** | No such post condition. |

| | |
|---|---|
| **Use Case Number:** | UC-10 |
| **Use Case Name:** | Detect correctness of sensors |
| **Overview:** | On the basis of the data that we receive , we will try to find sharp ups and downs in the reading and will detect correctness of our sensors. |
| **Actors:** | Professors, Students, Non-teaching Faculty, Workers, Other campus residents , system administrators |
| **Pre condition:** | No such pre condition. |
| **Flow:** | For this :<br>• The corresponding user shall open the webpage.<br>• Then go to see the line graph page.<br>• If sharp data changes are found over time , then that particular sensor is not working properly. |
| | Alternate Flows: No alternate flow needed. |
| **Post Condition:** | That sensor should be replaced or further calibrated . |