Low-Cost Programmable Air Quality Sensor Kits in Science Education

Bjørn Fjukstad[1], Nina Angelvik[1], Maria Wulff Hauglann[1], Joachim Sveia Knutsen[2], Morten Grønnesby[1], Hedinn Gunhildrud[3], and Lars Ailo Bongo[1]

[1]UiT, The Arctic University of Norway, Tromsø, Norway
[2]Kongsbakken Videregående Skole, Tromsø, Norway
[2]Science Centre of Northern Norway, Tromsø, Norway

Abstract

We describe our citizen science approach and technologies designed to introduce students in upper secondary schools to computational thinking and engineering. Using an Arduino microcontroller and low-cost sensors we have developed the *air:bit*, a programmable sensor kit that students build and program to collect air quality data. In our course, students develop their own research questions regarding air quality before using their own air quality sensor kit to answer their respective questions. This project combines electronics and coding with natural sciences providing a truly interdisciplinary course.

We have open-sourced the teaching materials including the building and coding instructions. In addition, students can contribute to our web-based platform for storing, visualizing, and exploring the collected air quality data. It also provides an open API for anyone to download air quality data collected by the students. Through the website, available at `airbit.uit.no`, students are motivated to contribute air quality data open to the public.

We describe lessons learned from our pilot project in a Norwegian upper secondary school and how we are deploying it in 10 schools across Northern Norway. In the pilot, students successfully built and coded the air:bits, and after two months of data collection they could correctly describe local patterns in the air quality. We believe that by combining electronics and coding with the natural sciences we motivate students to engage in all scientific disciplines.

Introduction

From a recently published Norwegian Official Report on the future of education in Norway, the authors argue that technology affects all subjects in the

Norwegian school system and that digital skills should be expressed in these.[1] While countries such as the UK, Finland, and Estonia have already, or are now, introducing programming and computational thinking in school curricula from primary school and upwards[2], Norway is falling behind. This is both due to the lack of mandatory courses, teaching materials, tools, and courses for teachers without any programming background.

We have developed a course that aims to introduce computer programming and engineering to Norwegian upper secondary schools. Our course combines engineering and computer programming with natural sciences. We developed a maker-inspired citizen science approach focused on air pollution data collection monitoring. We believe this engages both male and female students that are not primarily interested in programming and engineering. It also teaches scientific thinking since students themselves have to collect and verify data. As part of the course we developed an air pollution sensor kit that we have named *air:bit* to collect data. We provide teaching materials including the design, assembly instructions, example source code for the sensors, and a cloud based service for students to upload and explore their collected datasets.

The air:bit shown in Figure **??** measures air quality using low-cost sensors and the Arduino UNO microcontroller.[1] It measures i) dust particles, ii) temperature, iii) air humidity, iv) location, v) time and date, and log these to a memory card in an open format. The total cost of each kit is $40 USD. The cloud service provides data storage and visualization that can be used by students and the public to view current and previous trends in air quality. The cloud service also facilitates download of the air quality datasets through an open API.

The rest of this paper gives an introduction to air quality and related projects that aim to create awareness and engagement; describe our course and how it is structured; gives an overview of the air:bit and the computational resources; discuss our experiences from a pilot project; and lastly we provide future directions for the project.

Background

We provide a background in air quality and describe related projects that aim to create public awareness and information on air quality.

Air pollution

Air pollution is a global issue since it reduces quality of life in polluted areas and causes diseases. The WHO has termed air pollution as the largest single environmental health risk[3], and has both health, environmental and climate effects.[4] Both short and long term exposure to poor air quality as a result of air pollution is contributing to respiratory disease, cardiovascular disease, and certain cancers.[4, 5, 6, 7, 8] In Norway, both European and Norwegian

[1] arduino.cc

Figure 1: The air:bit.

legislation ensure that the air quality is monitored and that air quality forecasts are available to the public. Air pollution originate from a range of different sources. From chemical emissions in factories, chemicals used as a fertilizer in agriculture, exhaust from combustion engines burning fossil fuel, to micro-particles from cars driving on snow-free roads with studded tires[4, 9]. This wide range of sources generate different pollutants, e.g. nitrogen oxides (NO_x), Ozone (O_3), Particulate Matter (PM), and Carbon Monoxide (CO).

In Northern Norway the air quality is rapidly changing in the winter months, especially while the seasons are changing from winter to spring, and fall to winter mainly due to the use of spiked tires on dry roads that generate dust particles in the air, and the emissions from diesel powered cars. Reducing the use of cars will improve air quality in these months, and by creating awareness on the local conditions and their impact on health we believe that it is possible to reduce car usage without enforcing it. However, the monitoring is typically done using stationary equipment that provides high-quality data, but due to their high cost are not affordable to locate throughout all populated areas. Therefore citizens may not find available data on the air quality in their city or neighborhood and it's not possible to increase awareness around poor air quality and simple measures to improve it.

Recent initiatives to use low-cost, easy-to-use micro-sensors for air quality monitoring[10, 11, 12, 13] to replace the expensive instruments currently to gather high-quality air quality data. These projects are often targeted towards crowd sourcing air quality data, and must therefore provide an easy-to-use platform that anyone can use. In our project we focus on building the sensor kits to give the students a deeper understanding on how the technology works and not just how to use it. We have focused on measuring PM since it is the major cause of poor air quality in Tromsø, and there are simple and reasonably priced sensors available to measure dust densities in the air.

Air Pollution Engagement Projects

There are many recent projects that aim to engage the public to collect air quality data using a citizen science approach using various air quality kits.

Friskby Bergen is a Norwegian project with a similar aim as our project to develop a project for school classes to build air quality monitoring base stations and collect air quality data. Their technology is based on the Raspberry Pi computer and use low-cost sensors to measure air quality. Their monitoring stations are stationary and log data directly to their open web services over wifi. While the Friskby sensor kits provide live monitoring of air quality by directly uploading air quality measurement over wifi, our air:bit is equipped with a memory card to store measurements before being uploaded by the students afterwards. This simplifies both the software and makes it possible to use the kits without any wifi coverage.

The aim of the hackAIR project is to develop a simple DIY kits that enable citizens and organizations to engage in air quality data generation and sharing.[14] We share the same technical platform as the hackAIR project, and we have initiated a collaboration to share both data and experiences between the two projects.

The CITI-SENSE project has developed 'citizens observatories' for citizens to contribute to and participate in environmental governance[11]. Citizens use the Little Environmental Observatory (LEO) sensor packs to measure NO, NO_2, O_3, temperature and relative humidity. Data is uploaded to a online service for users to view and explore, similar to our approach.

Another project that share our technological platform is the Luftdaten project which aims to promote air quality awareness in Germany.[15] Through the `luftdaten.info` site they provide instructions on how to build and code the air quality sensor, in addition to interactive maps with live data. The Luftdaten project also host frequent meetups for citizens who need help with the air quality kits or want to learn more about the project. We share the approach with the Luftdaten project, but we target high-school students as the main audience.

Computer Science Education

There is a wealth of platforms and software tools to introduce programming and technology. One such approach is to combine electronics and coding through the cheap Arduino microcontroller architecture and electronic components such as LEDs, buttons or other sensors.[16, 17] An example of one such approach was to create or hack existing toys or build new ones to make "noise"[18]. Another is to use Arduinos in Physics experiments in the areas of Optics, Thermodynamics, and Waves.[19]

Course Contents

Our course has been given once in spring 2017 at UiT The Arctic University of Norway to science students at a local High School, and is planned again in spring 2018 across 10 upper secondary schools across Northern Norway. We host the course at the university both to recruit new students but also as a part of the outreach program at the university.

To make the course fit into different subjects in the upper secondary school in Norway, we have surveyed the relevant subjects and their specific learning goals and requirements. By creating a project where students build, code and use a sensor kit to investigate air pollution we cover learning outcomes from the subjects Technology and Research Learning (Teknologi og Forskningslære), Physics (Fysikk), Chemistry (Kjemi), Information Technology (Informasjonsteknologi), and Mathematics (Matematikk for realfag).

The course is run over the duration of a semester and given in four segments; first an introduction to air quality and research on the topic; a hands-on introduction to electronics and coding, as well as assembling and coding the air:bit; air quality data collection; and finally an evaluation where students summarize their work in a written report and/or a presentation. We followed the students' presentations both to get an overview of their learning outcome and get feedback on the project.

The course is designed for students in Norwegian upper secondary school, typically around 17-18 years old. Students may have some experience with basic circuits and electronics through science classes, but since there are no mandatory courses on computer programming so we do not expect any prior experience with coding. We can expect them to have basic computer skills and access to individual laptops.

As for the teaching staff at the individual institutions we cannot expect them to have more knowledge on air quality, microcontrollers, or extensive coding experience. Prior to the project we therefore invite teachers for an intensive two-day workshop that take the teachers through the four segments of the course. We host this two-day workshop at our department.

Following the two-day workshop teachers will return to their schools with parts and instructions to complete the course at their respective institutions. We provide an online forum for questions and answers, both from students and teachers, if they encounter any difficulties with assembling, coding or collecting data. The project is typically run in the spring months to capture the changing air quality from winter to spring.

In the course, teachers can themselves choose if they want to provide research questions to the students or if they want students to develop these themself. The imporant point is that the students all contribute to a large database with air quality measurements, but each student group each decide on what aspect of air pollution they want to investigate. For example, one student group investigated the relationship between snow coverage of roads and dust particles in the air.

Sensor Kit and Cloud Compute Infrastructure

In this section we describe our air quality measurement kit, its relevant documentation, the backend storage system, and the frontend visualization platform.

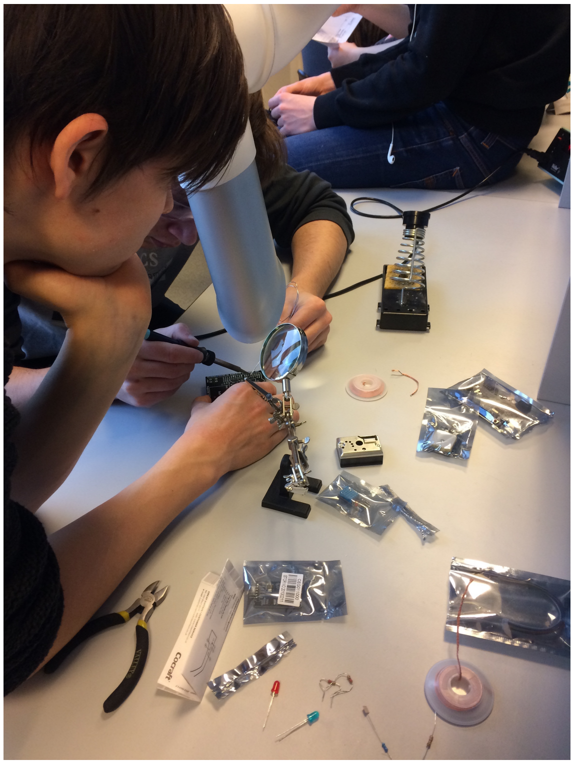


Figure 2: Students soldering different components of the air quality sensor kits to the custom PCB Arduino Shield.

The students collect air quality data and upload it to the backend storage system. To view and download data, users access a frontend web application. This application interfaces with both student-collected data in our local backend storage system, as well as open air quality data from the Norwegian Institute of Air Research (NILU) and the Norwegian Meteorological Institute (MET).

air:bit

We have designed the air quality sensor kit as a small microcontroller based data logger that collects measurements of dust particles, air temperature, air humidity, location, and time and date. The kit is enclosed in a laser cut box, equipped with an external power source that makes the kit portable and withstanding of different weather conditions. Figure **??** shows the first prototype. The kit was built as simple as possible to facilitate use in an educational setting. Table 1 lists the different components and their respective cost.

To simplify the assembly and soldering of the components to the microcontroller we have designed a custom PCB circuit board. The circuit board has

Figure 3: Students browsing their collected air quality data on the web site.

Table 1: A list of the different components in the air:bit along with their cost (as of August 2017).

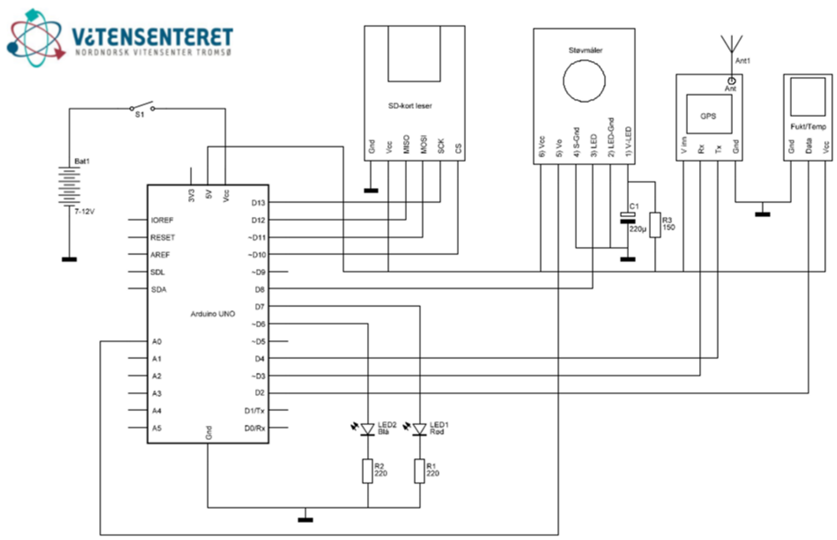
Component	Cost (USD)
Arduino Uno microcontroller	\$3.14
NEO6MV2 GPS module	\$8.19
Sharp GP2Y1010AUOF optical dust sensor	\$5.99
DHT11 temperature and humidity sensor	\$1.00
SD Card reader and 16GB memory card	\$4.74
Portable power bank	\$15.00
Two indicator leds and resistors	\$1.00
Custom PCB circuit board	\$2.00
Total:	\$41.06

pre-defined pins for each sensor, and fits on top of the popular Arduino UNO board. Figure 4a shows the underlying circuit and Figure 4b shows the custom designed PCB circuit board.

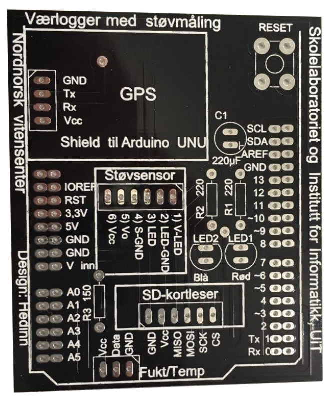
Students assemble the kit soldering the components to the custom PCB circuit board. The sensors and circuit board is then mounted on top of an Arduino UNO before enclosed in the pre-cut box.

To program the air:bit we use the standard Arduino IDE together with additional libraries to interface with the different sensors. Through the Arduino IDE students code and upload programs to the Arduino. Arduino programs are written in C++ and by the end of the project students will end up with a working solution at about 150 lines of code. While this is the final version, we expect students to write at least 500 lines of code during the project to test sensors and experiment with different solutions. The Arduino IDE features a console for monitoring communication from the Arduino, allowing students to read data in real time from the different sensors before taking it outside.

Since the kits are programmed in upper secondary school classes where both



(a) The Air Quality Sensor Kit circuit diagram.



(b) The custom PCB Arduino shield.

Figure 4: air:bit circuit diagram and custom PCB.

teachers and students have little or no coding experience, we aim to keep the code as simple as possible. In the project we distribute example code to interface and collect data from the individual sensors[2]. These are small $\tilde{1}00$ lines of code examples that typically take 20 minutes each to implement. The students must write their own program that collects data from all sensors simultaneously and write them to a memory card. We do not put any restrictions on how their code should look like, the only restriction is on the format of the output data written to the memory card.

[2] Available online at `airbit.uit.no`

We designed a simple data exchange format for the air:bit based on the simple CSV file format. Sensor kits create a single log file and append to it as it collects new measurements. A line (row) in the output file is an observation and consist of measurements for all available sensors. Listing 2 shows an example data file. Using simple Arduino libraries students write the data files to memory cards and they can view them using standard applications, such as Excel, on their laptops.

Listing 1: A simplified code example to collect and print temperature and humidity data from a DHT sensor every second.

```
void setup() {
  Serial.begin(9600); // Start Serial communication to
  dht.begin();        // receive messages from the Arduino
}                     // and initialize the DHT sensor.

void loop() {
    // Collect data and print them.
    float humidity = dht.readHumidity();
    float temperature = dht.readTemperature();
    Serial.print(temperature);
    Serial.print(", ");
    Serial.print(humidity);
    Serial.print("\n");
    delay(1000);
}
```

Listing 2: Example data file. Every line in the file is a measurement.

```
Time and date, Latitude, Longitude, Dust, Temperature, Humidity
26/10/2016 18:48:41, 69.682121, 18.978985, 88.98, 21.00, 20.00
26/10/2016 18:48:46, 69.682114, 18.978952, 95.70, 22.00, 17.00
26/10/2016 18:48:51, 69.682114, 18.978891, 99.06, 22.00, 17.00
26/10/2016 18:48:55, 69.682106, 18.978865, 98.22, 22.00, 17.00
                           ...
```

Backend data storage

We built a backend cloud based data storage system to handle the large quantities of student data. Students upload data directly from their memory cards to the backed system. The backend also serves queries for air quality data within given time intervals. Since we want to support a query-like interface to the collected air quality data, we used a relational database to build the backend storage system. Student access a lightweight web interface to upload data files from the air:bits, which are parsed and inserted into the database represented by a record for each measurement. Records are indexed using a combination of the time and date, and location. Invalid records are rejected and users will receive an error message to indicate any invalid data or formatting. The backend data storage system exposes a small REST API to allow the frontend (and other applications) to retrieve air quality data. The API will accept queries to retrieve all data within a given time interval. There is no limitation on the length of the time interval, and the API will return a single file with all measurements within the time period.

Frontend visualization

To simplify the process of accessing the collected air quality data, we built a web application. Students and the public can explore air quality data from

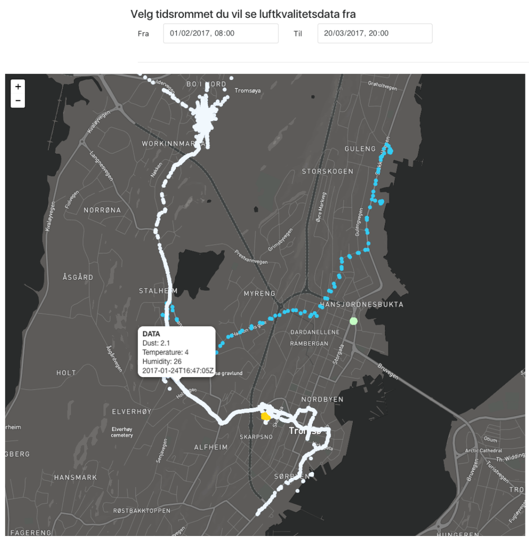


Figure 5: Screenshot from the web application for exploring collected air quality measurements.

the last 24 hours, or view and download data from any time period. The web application is built around a paneled visualization consisting of a map with measurement points plotted at their geographical location, in addition to other line plots visualizing the different measurements over time. Figure 5 shows a screenshot from an early version the web application with measurements from two different air:bits carried around in Tromsø, Norway.

Experiences

Our course has been delivered once in spring 2017 at the University of Tromsø to science students at a local High School (Kongsbakken Videregående Skole). The class consisted of 26 students divided into groups of 3-4 students. The teacher formed groups consisting of students at the same level of experience. The students had varying previous experience with soldering, and no one except one student had coding experience.

The assembly of the kit went almost without any issues. The groups had to present their soldering to one in the staff before moving on to the coding. This allowed us to verify their soldering. We observed that students took turns soldering, and offered each other help. It was clear that some students had done more prior soldering than others.

Coding the kit was a more challenging task. Since most students had little experience writing software, it was oftentimes not clear that they understood the structure of a program and where to start debugging it. The Arduino IDE helped a bit, but most groups had difficulties completing the coding without any help from the instructors. We believe that this is related to the minimal code examples and instructions we provided the students. An interesting point made by one of the students was that it would be helpful with some tool (test suite) that could verify that their implementation was correct.

The students collected data over a 2 month period, each group deciding on when and where to bring their kits on their own. This resulted in measurements from different areas over different time periods. We believe that this stems from our instructions not being clear enough that they should focus on collecting data at the same locations and time of day to make the interpretation simpler. One group ended up with mounting the kit outside their house to simplify data collection. Another group also photographed the road while they were collecting dust data to quantify the amount of snow and ice later.

Due to the scattered data points, the students experienced some difficulties interpreting the data. Because of this, groups with 'good' and complete datasets shared their data with other groups to help them answer their respective research questions. However due to the very simple dust sensor, groups had difficulties with comparing their data with official PM measurements.

After the project was complete we collected oral feedback from both the students and the teacher. Students enjoyed the interdiciplinary nature of the project, especially in groups where students could assign tasks evenly according to interest. The coding was by far the most difficult part of the project for the students as well as the teacher. Because the data interpretation was difficult because of the sensors and scattered data, students had to reflect on the study design and quality of the data before making any conclusions. While we are going to improve the data quality by using improved sensors, the teacher enjoyed the student's reflection on study design and data collection.

Future Work

While we successfully deployed the air:bit project in a upper secondary school class of 26 students, we have identified areas for improvement and some future directions for the project.

We will improve data quality from the air:bit by using an air quality sensor that can measure both PM2.5 and PM10 concentrations in the air. The current Sharp GP2Y1010AU0F optical dust sensor measures only the total amount of dust particles and cannot be directly compared with the air quality stations from the NILU. We will use the Nova SDS011[3] that both measures PM2.5 and PM10. We will also replace the DHT11 temperature and humidity sensor with the DHT22 temperature and humidity sensor since its readings more accurate and can read temperatures below 0 °C. With these new sensors we are aiming to perform a formal evaluation of their accuracy by sampling air quality data at one of the air quality measurement stations of NILU. We estimate that the new sensors will increase the total cost of each kit by USD $15.

[3] aqicn.org/sensor/sds011

We are also re-writing both the interface for students to explore their collected data, as well as redesigning both the backend storage solution, and the frontend to facilitate users from different parts of Norway. The current backend does not provide the low latencies required by the frontend visualization interface.

To make it possible to scale the project to 10 new schools we are improving the teaching materials that we distribute to the classes. Since we cannot host every class ourselves, we provide video lectures on the background material such as air quality and climate, as well as instructional videos on assembly of the kit and soldering. We are also re-writing the example code that we distribute to the students with better documentation and clearer code. We believe this will simplify and make the coding-part of the projects more pedagogical.

Conclusions

We have shown how we designed the air:bit, a simple programmable sensor kit for measuring air quality, and how we built a upper secondary school course for students to build and program the kit. We believe that by introducing students to electronics and coding in a citizen science project we motivate students to engage in both technology and the environment.

Acknowledgements

We would like to thank MET for providing open weather observations through `data.met.no` and `yr.no`. We would like to thank NILU for providing air quality measurements through their `luftkvalitet.nilu.no` portal. We would also like to thank Thomas Olsen, Torkjel Sandanger, and Juan Carlos Aviles Solis for their presentations and input on the projects.

References

[1] "NOU 2015: 8 - fremtidens skole." https://www.regjeringen.no/no/dokumenter/nou-2015-8/id2417001.

[2] European Schoolnet, "Computing our future. computer programming and coding - priorities, school curricula and initiatives across europe." http://www.eun.org/c/document_library/get_file?uuid=521cb928-6ec4-4a86-b522-9d8fd5cf60ce&groupId=43887, 2014.

[3] W. H. Organization, "World health assembly closes, passing resolutions on air pollution and epilepsy." http://www.who.int/mediacentre/news/releases/2015/wha-26-may-2015/en, 2015. [Online; Accesssed: 02.06.2017].

[4] C. Guerreiro, "Air quality in europe: 2013 report," 2013.

[5] R. Beelen, O. Raaschou-Nielsen, M. Stafoggia, Z. J. Andersen, G. Weinmayr, B. Hoffmann, K. Wolf, E. Samoli, P. Fischer, M. Nieuwenhuijsen, *et al.*, "Effects of long-term exposure to air pollution on natural-cause mortality: an analysis of 22 european cohorts within the multicentre escape project," *The Lancet*, vol. 383, no. 9919, pp. 785–795, 2014.

[6] R. D. Brook, S. Rajagopalan, C. A. Pope, J. R. Brook, A. Bhatnagar, A. V. Diez-Roux, F. Holguin, Y. Hong, R. V. Luepker, M. A. Mittleman, *et al.*, "Particulate matter air pollution and cardiovascular disease," *Circulation*, vol. 121, no. 21, pp. 2331–2378, 2010.

[7] O. Raaschou-Nielsen, Z. J. Andersen, R. Beelen, E. Samoli, M. Stafoggia, G. Weinmayr, B. Hoffmann, P. Fischer, M. J. Nieuwenhuijsen, B. Brunekreef, *et al.*, "Air pollution and lung cancer incidence in 17 european cohorts: prospective analyses from the european study of cohorts for air pollution effects (escape)," *The lancet oncology*, vol. 14, no. 9, pp. 813–822, 2013.

[8] M. Pascal, M. Corso, O. Chanel, C. Declercq, C. Badaloni, G. Cesaroni, S. Henschel, K. Meister, D. Haluza, P. Martin-Olmedo, *et al.*, "Assessing the public health impacts of urban air pollution in 25 european cities: results of the aphekom project," *Science of the Total Environment*, vol. 449, pp. 390–400, 2013.

[9] J. A. Hagen, P. Nafstad, A. Skrondal, S. Bjørkly, and P. Magnus, "Associations between outdoor air pollutants and hospitalization for respiratory diseases," *Epidemiology*, vol. 11, no. 2, pp. 136–140, 2000.

[10] J. Dutta, C. Chowdhury, S. Roy, A. I. Middya, and F. Gazi, "Towards smart city: Sensing air quality in city based on opportunistic crowd-sensing," in *Proceedings of the 18th International Conference on Distributed Computing and Networking*, p. 42, ACM, 2017.

[11] N. Castell, M. Kobernus, H.-Y. Liu, P. Schneider, W. Lahoz, A. J. Berre, and J. Noll, "Mobile technologies and services for environmental monitoring: The citi-sense-mob approach," *Urban climate*, vol. 14, pp. 370–382, 2015.

[12] A. Antonić, V. Bilas, M. Marjanović, M. Matijašević, D. Oletić, M. Pavelić, I. P. Žarko, K. Pripužić, and L. Skorin-Kapov, "Urban crowd sensing demonstrator: Sense the zagreb air," in *Software, Telecommunications and Computer Networks (SoftCOM), 2014 22nd International Conference on*, pp. 423–424, IEEE, 2014.

[13] D. Oletic and V. Bilas, "Design of sensor node for air quality crowdsensing," in *Sensors Applications Symposium (SAS), 2015 IEEE*, pp. 1–5, IEEE, 2015.

[14] hackAIR, "The hackair project." http://www.hackair.eu/pages/about-hackair, 2017. [Online; Accesssed: 16.08.2017].

[15] Luftdaten, "luftdaten.info – feinstaub selber messen." http://luftdaten.info, 2017. [Online; Accesssed: 16.08.2017].

[16] J. D. Brock, R. F. Bruce, and S. L. Reiser, "Using arduino for introductory programming courses," *Journal of Computing Sciences in Colleges*, vol. 25, no. 2, pp. 129–130, 2009.

[17] L. Buechley, M. Eisenberg, J. Catchen, and A. Crockett, "The lilypad arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 423–432, ACM, 2008.

[18] E. Brunvand and N. McCurdy, "Making noise: Using sound-art to explore technological fluency," in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pp. 87–92, ACM, 2017.

[19] C. A. Petry, F. S. Pacheco, D. Lohmann, G. A. Correa, and P. Moura, "Project teaching beyond physics: Integrating arduino to the laboratory," in *Technologies Applied to Electronics Teaching (TAEE), 2016*, pp. 1–6, IEEE, 2016.