Teaching Electronics and Programming in Norwegian Schools Using the air:bit Sensor Kit

Bjørn Fjukstad
Department of Computer Science, UiT
The Arctic University of Norway

Nina Angelvik
Department of Computer Science, UiT
The Arctic University of Norway

Morten Grønnesby
Department of Medical Biology, UiT
The Arctic University of Norway

Maria Wulff Hauglann
Department of Computer Science, UiT
The Arctic University of Norway

Hedinn Gunhildrud
Science Centre of Northern Norway

Fredrik Høisæther Rasch
Department of Computer Science, UiT
The Arctic University of Norway

Julianne Iversen
Faculty of Science and Technology,
UiT The Arctic University of Norway

Margaret Dalseng
Faculty of Science and Technology,
UiT The Arctic University of Norway

Lars Ailo Bongo
Department of Computer Science, UiT
The Arctic University of Norway

ABSTRACT

We describe lessons learned from using the air:bit project to introduce more than 150 students in the Norwegian upper secondary school to computer programming, engineering and environmental sciences. In the air:bit project, students build and code a portable air quality sensor kit, and use their air:bit to collect data to investigate patterns in air quality in their local environment. When the project ended students had collected more than 400,000 measurements with their air:bit kits, and could describe local patterns in air quality. Students participate in all parts of the project, from soldering components and programming the sensors, to analyzing the air quality measurements. Following the project we conducted a survey to investigate the satisfaction of participating in the project, as well as learning outcomes in the different parts of the project to improve future versions. The results show that both teachers and students were very satisfied with participating in the project. While the students showed positive learning outcomes in electronics and simple programming parts of the project, there is room for improvement regarding more advanced programming tasks. We believe that our work provides insight for other similar maker-inspired projects, and guidelines for educators to incorporate computer programming in traditional science experiments.

CCS CONCEPTS

• **Social and professional topics** → **Computing education**; **Computational thinking**; • **Hardware** → *Sensor devices and platforms.*

KEYWORDS

computer science education, microcontrollers, STEM, air quality

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ITiCSE '19, July 15–17, 2019, Aberdeen, Scotland Uk
© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-6301-3/19/07…$15.00
https://doi.org/10.1145/3304221.3325527

ACM Reference Format:
Bjørn Fjukstad, Nina Angelvik, Morten Grønnesby, Maria Wulff Hauglann, Hedinn Gunhildrud, Fredrik Høisæther Rasch, Julianne Iversen, Margaret Dalseng, and Lars Ailo Bongo. 2019. Teaching Electronics and Programming in Norwegian Schools Using the air:bit Sensor Kit. In *Innovation and Technology in Computer Science Education (ITiCSE '19), July 15–17, 2019, Aberdeen, Scotland Uk.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3304221.3325527

INTRODUCTION

We have developed the air:bit, an Arduino-based air quality sensor kit that students build and program to collect air quality data in their local environment[13]. Together with the air:bit sensor kit, we developed teaching materials that include how to assemble the sensor kit and how to program its different sensors, and a cloud based service for students to upload and explore their collected datasets. All of which are openly available online at airbit.uit.no. We used these resources develop an interdisciplinary course for students in Norwegian upper secondary schools, which we offered to 11 different schools across Northern Norway.

From our first deployment of the air:bit project we identified a set of necessary improvements to scale out to even more schools. First, the initial version of the sensor kit did not provide the students with comparable data to official measurement stations. Second, because we wanted to invite more schools we had to improve how we shared resources such as programming guides to the students and teachers. Third, we had to improve the online platform for exploring the datasets to handle data from many geographically distributed locations. With these technical improvements in place, we could deploy the project to more schools, and evaluate the teaching outcomes of the participating students. Our primary goal of the evaluation was to map the difficulties of each part of the project to further improve it in future versions.

In this paper we describe the evaluation of deploying the air:bit project in 11 Norwegian upper secondary schools. The participating schools were all in Northern Norway, but spread over a large geographical area. We developed a survey to collect information about the experiences and teaching outcome for both students and teachers. We used this study to investigate the prior knowledge about programming and electronics, the difficulties of the different

parts of the project, and to highlight areas of improvement. We describe how we redesigned the sensor kit and its accompanying resources, the online data exploration platform and the student experiences from this second deployment of the air:bit project. We believe our work provides insight for others that provide similar maker-inspired projects to students without any programming experience, and that it answers some questions about the effect of such projects for motivating students to engage in computer science and other STEM subjects.

THE AIR:BIT PROJECT

At The Faculty of Science and Technology at UiT The Arctic University we have a School Laboratory that is open to all local school classes, and teachers, to visit and learn about technology.[1] The school laboratory is run by academic staff at the different departments at the faculty, in addition to administrative staff at the faculty. After a teacher conference in computer programming in 2016 there was a request for a course at the school laboratory that combined computer programming with natural sciences. In 2017 we initiated the air:bit project with a single upper secondary school class. In our previous publication, we describe motivation behind the air:bit project, the first version of our air:bit, and our experiences from the first participating class[13].

In this paper we wish to focus on the necessary improvements to scale out the project to more schools, and what the learning outcomes for the participating students were.

Background

While programming or computational thinking is added to the school curricula in countries such as the UK, Finland or Estonia, Norway is unfortunately falling behind[11]. Initiatives such as Lær Kidsa Koding[2] are working with legislators to introduce these concepts in the Norwegian educational system, but it is a time consuming process. Fortunately, certain science subjects in the upper secondary school allow teachers to add smaller projects that combine programming with the traditional sciences.

There are a wealth of platforms and tools to introduce computer science to students of all ages. One such tool is the cheap Arduino microcontroller where users combine electronic components such as LEDs and buttons, with computer programming.[5, 8] These microcontrollers can be applied in a wide range of disciplines and application areas, from physics experiments[19] to designing wearable textiles[8, 12], or even hacking toys.[7]

Air pollution is the largest single environmental health risk, and it contributes to respiratory disease, cardiovascular disease and certain cancers.[4, 6, 14, 17, 18, 20] Air pollution originates from a wide range of sources, and especially Particulate Matter (PM) from combustion engines and cars driving with studded tires is a major problem in Norway[14, 15].

With cheaper and higher quality sensors, combined with easy-to-use microcontrollers such the Arduino UNO, it is possible to develop small sensor kits with little previous knowledge and experience with electronics. Low-cost air quality monitoring kits have already shown their effectiveness in citizen science projects, by reducing the high cost of developing such measurement stations.[3, 9, 10, 16]



Figure 1: The second version of the air:bit sensor kit.

air:bit

The air:bit is a small microcontroller based data logger for measuring dust particles, air temperature, air humidity, GPS-based location, time and date. The kit is enclosed in a laser cut box, equipped with an external battery for portability. Table 1 lists the different components and their respective cost, and Figure 1 shows the assembled kit. We package and ship all components to each school in cardboard boxes.

We designed the kit as simple as possible to facilitate use in an educational setting. To simplify the assembly and soldering of the components to the microcontroller we use a custom PCB circuit board that has pre-defined pins for each sensor, and fits on top of the Arduino UNO board. New in the second version of our air:bit is the Nova SDS011 Dust sensor which allow students to collect data that can be compared to official measurement stations (PM2.5 and PM10). This sensor is more expensive than the previous dust sensor, but provides higher quality data. We also update the temperature and humidity sensor from the DHT11 to the DHT22. This sensor allows students to record temperatures below 0°C, which is frequent in Northern Norway. We also re-designed the box to make it easier for students to carry around.

We continue to use the standard Arduino IDE together with additional libraries to program the air:bit. The libraries provide the low-level functionality to retrieve data from each sensor, and students use their respective APIs to assemble a program that collects data from all sensors simultaneously and write them to a memory card. We distribute example code to interface and collect data from the individual sensors. These are small examples typically less than 100 lines of code, and students typically end up with a complete solution of around 500 lines of code. The data is recorded to the memory card using a simple CSV file format that make it simple to view and inspect the output datasets.

Improved Online Resources

To allow more schools to participate in the air:bit project we had to improve the online resources. We developed detailed guides on how to solder the components of the air:bit, how to assemble the box, and how to program each sensor. We also included a resources

[1] uit.no/skolelab
[2] kidsakoder.no

Table 1: A list of the different components in the air:bit along with their cost (as of January 2018).

Component	Cost (USD)
Arduino Uno microcontroller	$3.00
NEO6MV2 GPS module	$6.00
Nova SDS011 dust sensor(PM2.5 and PM10).	$14.50
DHT22 temperature and humidity sensor	$3.50
SD Card reader and 16GB memory card	$9.00
Portable power bank	$15.00
Custom PCB circuit board	$3.00
Custom enclosure box	$5.00
USB cable	$2.00
Zip-lock bag with LEDs, resistors and spare parts	$5.00
Cardboard box	$1.50
Total:	$67.50

page with video lectures on air pollution from environmental researchers from the Norwegian Institute for Air Research (NILU) and the The Norwegian Meteorological Institute (MET). All of these are available online at airbit.uit.no. These are open to everyone.

Together with the educational resources we developed the `air:bit` platform, a system for storing, exploring and visualizing air quality data from the kits and other, external data sources.[1, 2] The system consists of three individual parts, the `air:bit` web application, a service for retrieving and uploading datasets, and a backed storage service. The web application provides students with educational resources such as instructions on how to solder each component and program the sensors, in addition to an interactive visualizations of the air quality datasets. The other two services are responsible for retrieving student data and making them available through the interactive visualizations. We host the backend services on Google Cloud Platform which makes it possible for us to scale out if we get even more participating schools. Together with student data we also retrieve data from NILU and MET through their open APIs. By developing an online service that integrates the different data sources we make it easier for students to explore the heterogeneous data.

The 2018 `air:bit` Project

We have offered our `air:bit` course twice. The pilot was held in spring 2017 at the University of Tromsø to science students at a local High School (Kongsbakken Videregående Skole). The second round was in spring 2018 with 11 participating schools, and 164 students from across Northern Norway. In the second round, the participating teachers chose to use the `air:bit` project in four different subjects, from all three years of upper secondary school. The 164 students built in total 62 `air:bit` sensor kits.

We invited schools to participate in the second round of the project in the beginning of the fall semester of 2017. The participating schools were invited to a two-day workshop where the goal was to teach them how to assemble and program the sensor kit. Teachers from 11 schools participated and successfully built and programmed their own kits. Following the teacher workshops the teachers themselves chose how to incorporate the `air:bit` project in their classroom.

In the spring of 2018 all participating classes were invited to attend a full day workshop where they got help to finalize the assembly and programming of the `air:bit` kits. Most classes only had minor programming issues, but some student groups needed help to troubleshoot their soldering and assembly. Following the workshop, each class returned home to start data collection.

During the spring of 2018 students uploaded more than 400,000 unique measurements to our online web application. The class with the largest number of data points used their data to investigate and compare air pollution levels on the outside playgrounds of local kindergartens.

EVALUATION

The main goal of the evaluation was to get feedback from the participating students and teachers on the project for us to improve it. We aim to offer this course to Norwegian upper secondary schools in future years, both to engage students in computer programming, but also to motivate students to apply to higher education in science and technology. For us to do so, it is important to understand the difficulties of the different parts of the project, what the students learned, and areas for us to improve.

We developed two surveys to investigate learning outcomes and experiences from the `air:bit` course, one for students and another for teachers. Specifically we wanted to investigate: i) the overall satisfaction of having participated in the project; ii) how much time students spent working on the project; iii) if the students had used the online materials; iv) the difficulty of each part of the project; v) the students prior experience with programming and electronics; and vi) the programming specific learning outcomes, e.g. knowledge about variables and loops.

All questions were written in Norwegian, and we present their English translation in this paper. We have also translated the free-text responses from the students and teachers. We distributed both surveys online using a service similar to Questback.

To investigate the student perceived learning outcome and experiences from the course we asked the students to answer a questionnaire consisting of 14 multiple choice questions, and 35 statements answered on a five-level Likert scale. Of the total 164 students, we received 90 individual responses. This gives 55% answer rate. Of the 90 responses 64% were boys and 36% girls. Regarding the questionnaire for teachers, 8 of the 11 (73%) participating teachers replied to our survey.

Student Experiences

To investigate the time spent working on the project, we asked the students to report the number of weeks and hours per week spent on the project. Table 2 shows the reported number of weeks spent working on the project and Table 3 shows the reported number of hours spent every week working on the project.

We asked the students to report how they had used the online materials. 94% of the students reported that they used the online materials while they where building and programming the `air:bit`. Of these, 83% reported that these where either to a large degree or to a very large degree helpful. 14% reported that the online materials where to some extent helpful, while 4% reported that the material

Table 2: Reported number of weeks spent working on the `air:bit` project.

Less than 4 weeks	4 to 8 weeks	9 to 12 weeks	13 to 16 weeks	More than 16 weeks
4%	20%	45%	27%	4%

Table 3: Reported hours spent working on the `air:bit` project every week.

Less than 1 hour	1 to 2 hours	3 to 4 hours	More than 5 hours
10%	23%	50%	17%

was not at all or to a small degree helpful. These results back up our efforts to develop and maintain online teaching materials.

To investigate the satisfaction of the students, we asked them to rate how satisfied or dissatisfied having participated in the air:bit project. 69% of the students reported that they were satisfied or very satisfied, 23% nor satisfied or dissatisfied and 8% reported that they were very dissatisfied or dissatisfied. We are very happy with these numbers, but hope that in future editions we can further improve the satisfaction rate.

To evaluate the difficulty of each part of the project, we asked the students to evaluate the difficulty of the main tasks of the project. Table 4 shows the questions and how the students responded. From the table we clearly see that the students found it easy to get an overview of the components and solder the air:bit together. Further the students fount the LEDs, the temperature and humidity sensor, and the dust sensor relatively easy to program. These are all sensors that require minimal code, and are easy to test in the class room. However programming the GPS and memory card was a more difficult task. The most difficult task of all was to assemble all the individual parts into a single program. More than half the students answered that it was either *difficult* or *very difficult*.

Getting the GPS unit to function requires that students both write the correct code, but also take it outside for testing since it is difficult to get a GPS signal inside. We believe that the difficulties are mainly attributed to the advanced interface to the GPS library we provide, in addition to designing a program that checks for a valid GPS position and date, as well as formatting the output correctly. We hope to improve this interface in the future.

Assembling all small programs into one large program posed difficulties for most students. Since the students write separate programs for each sensor, they have to combine the flow of each program into a single piece of software. It is apparent that we failed to show how to merge two or more different Arduino programs, and this is something that we hope to improve in future versions. We have thought about developing the full program from the ground up starting with the LEDs and adding code for each component until they are all covered. This would simplify the assembly of a single program, but will make it more difficult to troubleshoot issues with each sensor since they are integrated into the same piece of software. Having one program for each sensor makes it easy to point to a faulty solder joint or incorrect usage of a software library. Remember that with Arduinos both the code and the physical sensors may cause errors.

To understand how the students' knowledge on the different parts of the project, we asked them to asses their knowledge before and after the project.

From Table 5 we see that the students reported that they knew much or very much about soldering electrical components after the projects was over. We also see a large increase in students reporting they know much or very much about programming Arduinos. Interestingly, only 40% of students report they know much or very much about measurement uncertainty in research data. By using simple off-the-shelf sensors we had hoped that we had informed the students that the sensors did not provide excellent data quality, and how to critically interpret the data. We hope to expand and improve the parts of the project where students actually perform the data analysis in future editions.

To evaluate programming specific knowledge we asked the students to rate their knowledge before and after the project. Table 6 summarizes the results. From the table we see clearly that around a third of the students reply that they know little or very little about all statements. While we see that this percentage is smaller than before the project, there is still a lot of work to do in the programming-specific parts of the `air:bit` project. We also see that the percentage of students that know little or very little is largest for the statements regarding logic tests and loops. These are fundamental in assembling the full `air:bit` program, and especially the GPS unit, and believe that these results back the results we saw in 4.

In addition to our questions, we left a text field where the students could enter free-text comments to the project. In total 17 students used this opportunity to give us comments. From these 17, two students wrote that they were given instructions that the `air:bit` kits could be used outside, and that the kits had malfunctioned from extensive outside exposure (5 days). Another student noted that the enclosing box was a bit too small for all the components. One student reported "It is a difficult project, but it's fun". Another student wrote "The project was very fun, but much of the work following the data collection it was pretty gruesome to complete. However, I learned a lot of lessons of how complicated such processes are". One student wrote that they did not get the programming guidance they needed when they visited the university. Another student wrote that the programming was too easy, and that it did not challenge the students. One student requested a guide on presenting research data.

Teacher Experiences

From the responses we got an overview of the time spent on the project, their knowledge about programming and microcontrollers prior to the project, and the learning outcomes for their students. The survey consisted of multiple choice questions and a free text field where they could enter their own comments.

Of the 8 replies, all teachers responded that their students had significant or very significant learning outcomes. All teachers also responded that they would recommend other classes to participate in the project. More than half of the teachers reported that they had much, or very much knowledge about electrical circuits, soldering,

Table 4: Survey results from reported difficulty in completing the different tasks of assembling and programming the `air:bit`.

Statement	Very easy or easy	Nor easy nor difficult	Difficult or very difficult
Solder the components	80%	18%	2%
Get an overview of the components	83%	13%	3%
Program the LEDs	73%	18%	9%
Program the temperature and humidity sensor	50%	39%	11%
Program the dust sensor	46%	41%	13%
Program the memory card and memory card reader	28%	42%	30%
Program the GPS	20%	37%	43%
Assemble the different programs into a single program	15%	32%	53%

Table 5: Survey results from the self-assessed general knowledge before and after participating in the `air:bit` project. We have combined the top and bottom categories.

Statement	Very little or Little	Neither little nor much	Much or Very much
How to solder electrical components	*23%* → 8%	*22%* → 14%	*55%* → 78%
How to plan and execute a scientific project	*24%* → 8%	*29%* → 30%	*47%* → 62%
How to collect and analyze research data	*39%* → 8%	*31%* → 30%	*30%* → 62%
What an Arduino is and how to program it	*46%* → 13%	*28%* → 28%	*26%* → 59%
How electrical circuits work	*32%* → 17%	*39%* → 33%	*29%* → 50%
How to write a computer program	*39%* → 18%	*38%* → 32%	*23%* → 50%
How to determine measurement uncertainty in research data	*36%* → 12%	*41%* → 38%	*24%* → 40%

Table 6: Survey results from the self-assessed programming knowledge before and after participating in the `air:bit` project. We have combined the top and bottom categories.

Statement	Very little or Little	Neither little nor much	Much or Very much
What a variable is and how they are used	*49%* → 29%	*17%* → 28%	*34%* → 43%
What a data type is and how they are used (e.g. `float` or `int`)	*50%* → 31%	*26%* → 28%	*24%* → 41%
What a loop is and how they are used (e.g. a `for` loop)	*56%* → 35%	*22%* → 24%	*22%* → 41%
What a logic test is and how they are used (e.g. an `if` test)	*54%* → 34%	*20%* → 25%	*36%* → 41%
How to debug a computer program	*63%* → 30%	*19%* → 29%	*18%* → 41%

programming, and how to plan and execute a scientific project. However, there were still teachers who reported that they had very little or little experience in all of these areas.

From the free text responses, one teacher reported that the students became a bit tired towards the end due to the process of writing up their findings. One teacher said that the motivation was highly variable between the students and groups, and that they did not feel any ownership to the programming due to a lot of copy-pasting.

DISCUSSION AND FUTURE WORK

Unfortunately we were unable to distribute the first set of questionnaires before the project started, and another after the project. This meant that we asked the students after completing the project to evaluate their knowledge prior to starting the project. We believe that this could have an impact on the responses.

From the feedback from the students and teachers we aim to make improvements to the `air:bit` project. The main issue we are targeting in the next round of the project is how students write the full `air:bit` Arduino program. The previous approach has been to program each sensor independently before assembling together one large program that does everything. This made it easy for students to focus on one single component at a time, making debugging both software and hardware simpler. The drawback was that most students struggled with how to assemble them together in one single program. One improvement we will incorporate in the next round is to show how we combine the individual programs for the simple sensors together into a single program. While we have developed static programming guides for students to follow, we are also experimenting with developing video guides maybe even as live streams. We hope that this will help students better understand the different computer science concepts they are introduced to, as well as simplifying the programming of the `air:bit`.

As mentioned earlier, the students had difficulties getting the GPS module to operate correctly. We have already developed a custom Arduino library for the dust sensor,[3] and aim to do the same for the GPS library. The Arduino library for the GPS module provides a low-level interface to the GPS module, but it requires extensive error handling by the calling program. We aim to develop our own fork of the original library, which provides one interface to both get a location, parse the output, return them to the user, and perform the necessary error handling.

One of the participating classes ended up not programming the `air:bit` sensor kit themselves, but sharing a complete solution. We did not make it possible to identify specific classes from individual responses, so we could not exclude the responses from these students. These students will not have completed all the different programming tasks, and we believe that they may influence the results.

One area of the project we have not discussed yet, is how to analyze the collected air pollution measurements. This has been left out to the teachers, but we are experimenting with interactive Jupyter[4] notebooks that allow students to interactively explore their data with statistical programming languages such as R or Python.

[3]Online at github.com/fjukstad/SDS011.
[4]jupyter.org

CONCLUSIONS

We have successfully deployed our `air:bit` project to schools across Northern Norway. Doing so we have introduced more students to computer programming and electronics. The new version of our sensor kit provides higher quality datasets, and together with the online resources the kits are easier to assemble and program.

The results from the surveys show that both students and teachers enjoy the project, with only 8% of students reporting that they were dissatisfied or very dissatisfied. The majority of the students spent more than 9 weeks working on the project, spending more than 3 hours each week. 94% of the students used the online materials we have developed, and 83% of the students reported that these were very helpful. Assembling the `air:bit` and programming each of the sensors were simple tasks, but combining each of the different programs into a single program that collected all the data was a difficult task for the majority of the students. While the students managed to develop the code for the sensor kit, the results show that it would be beneficial to spend more time teaching the different fundamental concepts of computer science, such as variables, logical tests, and loops.

With these results in mind, we are currently in the process of organizing a third round of the `air:bit` project with even more participating schools. We aim to continue improving the course contents based on the results we have presented, and hope other projects will benefit from our experiences.

ACKNOWLEDGEMENTS

First we would like to thank all the students and teachers who participated in the project. We would like to thank Norwegian Meteorological Institute (MET) for providing open weather observations through data.met.no and yr.no. We would like to thank Norwegian Institute of Air Research (NILU) for providing air quality measurements through their luftkvalitet.nilu.no portal. We would also like to thank Thomas Olsen, Sonja Grossberndt, and Juan Carlos Aviles Solis for their presentations and input on the projects. We would like to thank the students at the Department of Computer Science for their help teaching the course. Last, we would like to thank Monica Martinussen and Ane Sætrum for their help developing the surveys.

REFERENCES

[1] Nina Angelvik. 2018. Air Pollution Data Analysis Platform for Computer Science Education Projects. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. ACM, 271–271.
[2] Nina Angelvik. 2018. *Data management platform for citizen science education projects.* Master's thesis. UiT Norges arktiske universitet.
[3] Aleksandar Antonić, Vedran Bilas, Martina Marjanović, Maja Matijašević, Dinko Oletić, Marko Pavelić, Ivana Podnar Žarko, Krešimir Pripužić, and Lea Skorin-Kapov. 2014. Urban crowd sensing demonstrator: Sense the zagreb air. In *Software, Telecommunications and Computer Networks (SoftCOM), 2014 22nd International Conference on*. IEEE, 423–424.
[4] Rob Beelen, Ole Raaschou-Nielsen, Massimo Stafoggia, Zorana Jovanovic Andersen, Gudrun Weinmayr, Barbara Hoffmann, Kathrin Wolf, Evangelia Samoli, Paul Fischer, Mark Nieuwenhuijsen, et al. 2014. 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* 383, 9919 (2014), 785–795.

[5] J Dean Brock, Rebecca F Bruce, and Susan L Reiser. 2009. Using Arduino for introductory programming courses. *Journal of Computing Sciences in Colleges* 25, 2 (2009), 129–130.

[6] Robert D Brook, Sanjay Rajagopalan, C Arden Pope, Jeffrey R Brook, Aruni Bhatnagar, Ana V Diez-Roux, Fernando Holguin, Yuling Hong, Russell V Luepker, Murray A Mittleman, et al. 2010. Particulate matter air pollution and cardiovascular disease. *Circulation* 121, 21 (2010), 2331–2378.

[7] Erik Brunvand and Nina McCurdy. 2017. Making Noise: Using Sound-Art to Explore Technological Fluency. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. ACM, 87–92.

[8] Leah Buechley, Mike Eisenberg, Jaime Catchen, and Ali Crockett. 2008. 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*. ACM, 423–432.

[9] Nuria Castell, Mike Kobernus, Hai-Ying Liu, Philipp Schneider, William Lahoz, Arne J Berre, and Josef Noll. 2015. Mobile technologies and services for environmental monitoring: The Citi-Sense-MOB approach. *Urban climate* 14 (2015), 370–382.

[10] Joy Dutta, Chandreyee Chowdhury, Sarbani Roy, Asif Iqbal Middya, and Firoj Gazi. 2017. 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*. ACM, 42.

[11] European Schoolnet. 2014. 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.

[12] Deborah Ann Fields, Yasmin Kafai, Tomoko Nakajima, Joanna Goode, and Jane Margolis. 2018. Putting making into high school computer science classrooms: Promoting equity in teaching and learning with electronic textiles in exploring computer science. *Equity & Excellence in Education* 51, 1 (2018), 21–35.

[13] Bjørn Fjukstad, Nina Angelvik, Maria Wulff Hauglann, Joachim Sveia Knutsen, Morten Grønnesby, Hedinn Gunhildrud, and Lars Ailo Bongo. 2018. Low-Cost Programmable Air Quality Sensor Kits in Science Education. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education (SIGCSE '18)*. ACM, New York, NY, USA, 227–232. https://doi.org/10.1145/3159450.3159569

[14] Cristina Guerreiro. 2013. Air quality in Europe: 2013 report. (2013).

[15] Jørgen A Hagen, Per Nafstad, Anders Skrondal, Sonja Bjørkly, and Per Magnus. 2000. Associations between outdoor air pollutants and hospitalization for respiratory diseases. *Epidemiology* 11, 2 (2000), 136–140.

[16] Dinko Oletic and Vedran Bilas. 2015. Design of sensor node for air quality crowdsensing. In *Sensors Applications Symposium (SAS), 2015 IEEE*. IEEE, 1–5.

[17] World Health Organization. 2015. World Health Assembly closes, passing resolutions on air pollution and epilepsy. http://www.who.int/mediacentre/news/releases/2015/wha-26-may-2015/en. [Online; Accesssed: 02.06.2017].

[18] Mathilde Pascal, Magali Corso, Olivier Chanel, Christophe Declercq, Chiara Badaloni, Giulia Cesaroni, Susann Henschel, Kadri Meister, Daniela Haluza, Piedad Martin-Olmedo, et al. 2013. Assessing the public health impacts of urban air pollution in 25 European cities: results of the Aphekom project. *Science of the Total Environment* 449 (2013), 390–400.

[19] Clovis Antonio Petry, Fernando S Pacheco, Daniel Lohmann, Gabriela A Correa, and Paulo Moura. 2016. Project teaching beyond Physics: Integrating Arduino to the laboratory. In *Technologies Applied to Electronics Teaching (TAEE), 2016*. IEEE, 1–6.

[20] Ole Raaschou-Nielsen, Zorana J Andersen, Rob Beelen, Evangelia Samoli, Massimo Stafoggia, Gudrun Weinmayr, Barbara Hoffmann, Paul Fischer, Mark J Nieuwenhuijsen, Bert Brunekreef, et al. 2013. 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* 14, 9 (2013), 813–822.