eCraft2Learn

Digital Fabrication and Maker Movement in Education
Making Computer – supported Artefacts from Scratch

Deliverable D4.6

3D printing and DIY electronics infrastructure analysis and user feedback

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No 731345.

PROJECT DESCRIPTION

Acronym:	**eCraft2Learn**
Title:	Digital Fabrication and Maker Movement in Education: Making Computer-supported Artefacts from Scratch
Coordinator:	University of Eastern Finland
Reference:	731345
Type:	RIA
Program:	HORIZON 2020
Theme:	Technologies for Learning and Skills
Start:	01. January, 2017
Duration:	24 months
Website:	https://project.ecraft2learn.eu/
E-Mail:	office@ecraft2learn.eu

Consortium:

University of Eastern Finland, Finland, (UEF), Coordinator
Edumotiva, Greece (EDUMOTIVA)
Mälardalen University of Sweden, Sweden (MDH)
Zentrum für Soziale Innovation, Austria, (ZSI)
The University of Oxford, United Kingdom, (UOXF)
SYNYO GmbH, Austria, (SYNYO)
University of Padua, Italy, (UNIPD)
Technopolis City of Athens, Greece (TECHNOPOLIS)
Evothings, Sweden (EVOTHINGS)
Arduino, Sweden (ARD)
Ultimaker, United Kingdom (ULTIMAKER)
Linnaeus University, Sweden, (LNU)

DELIVERABLE DESCRIPTION

Number:	**D4.6**
Title:	**3D printing and DIY electronics infrastructure and user feedback**
Lead beneficiary:	**UEF**
Work package:	WP4
Dissemination level:	Public (PU)
Type	Other (O)
Due date:	30.06.2018
Submission date:	30.06.2018
Authors:	**Tapani Toivonen, Calkin Suero Montero,** UEF
Contributors:	Sonya Horton, ULTIMAKER | Baran Cürüklü, MDH
Reviewers:	Konstantina Zachari, TECHNOPOLIS

Version Control

Version	Date	Person in charge (Organization)	Changes	Quality Assurance
1	10.06.2018	Tapani Toivonen (UEF)	1st draft created	Calkin Suero Montero (UEF)
2	20.06.2018	Tapani Toivonen (UEF)	- Changes in document structure - Adding more details on the infrastructure analysis	Calkin Suero Montero (UEF) Baran Cürüklü (MDH)
3	26.06.2018	Tapani Toivonen (UEF) Calkin Suero Montero (UEF)	- refocussing introduction and clarifying the structure analysis of the learning analytics - correcting typos	Konstantina Zachari (TECHNOPOLIS)
4	29.06.2018	Tapani Toivonen (UEF)	Final corrections to the structure of the document and the contents of the document	Calkin Suero Montero (UEF)

Acknowledgement: This project has received funding from the European Union's Horizon 2020 Research and Innovation Action under Grant Agreement No 731345.

TABLE OF CONTENT

TABLE OF FIGURES

EXECUTIVE SUMMARY

This deliverable introduces the data collection from the 3D printing and DIY electronics infrastructure point of view as well as how these data is analyzed through the Learning Analytics (LA) system. In the eCraft2Learn learning ecosystem, 3D printing and DIY electronics are an integral part of the *making* activities taking place in both formal and informal settings. The data that is collected from the ecosystem during each pedagogical stage of the development of tasks represent the progress of the learners and can be analyzed through several methods in the LA system. Our LA implementation uses machine learning approaches such as classification, cluster analysis, outlier detection and association rule learning in order to assist the teacher in their role as a coach, particularly enabling them to deepen their understanding of their students' progress while using 3D printing and DIY electronics in their projects. The knowledge discovery supported by machine learning assists the teacher's feedback and timely interventions, and it is used to also support the learners' progress through self-regulation and self-reflection via the Unified User Interface (UUI). The demonstration video accompanying this document is accessible through the following link:
https://cs.uef.fi/~tapanit/ecraft2learn/s4a-la-demo.mp4

1 INTRODUCTION

This deliverable presents and describes the infrastructure of DIY electronics and 3D printing from the learning analytics perspective, which powers the teacher interface of the eCraft2Learn digital platform. The learning analytics system has been developed to support the role of the teacher as a coach, or facilitator, of the learning experience. The work towards the creation of the learning analytics system was framed on alleviating the difficulties that teachers/coaches face when dealing with unexpected learning paths that can invariably occur in learning situations as to decide when to intervene and provide useful feedback during the learner's *making* process. The challenges concern formal as well as informal settings. Learning analytics is the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs (D3.4). We consider essential to provide the opportunity of real-time feedback to teachers/coaches as well as learners as to optimize the teaching/learning experience.

Within the eCraft2Learn learning ecosystem it is understood that building and programming even simple computer-supported artefacts from the beginning requires a number of steps where complex and sometimes dull debugging processes are needed in order to succeed with *making*. Failing in these tasks might cause learners to give up with the exercise, and ultimately their project or program. Note that it is generally accepted that, not having control of the learning process, not perceive any progress, and being forced to deal with issues that are not perceived as part of the problem to be solved can influence the motivation of a learner in a negative way. The learning analytics system aims at addressing these issues by collecting, analyzing and producing visual representation of technical and pedagogical data for teachers and learners. The LA can track learners' activities while working with DIY electronics during their projects, and offer immediate visual information regarding the students' progress to both the teacher/coach and the students.

The DIY electronics in the context of eCraft2Learn *making* activities contain the programming tasks of sensors and other electronic components (e.g., DC motors, servomotors, LEDs, etc.). The programming of these components is carried out by using Arduino boards through Snap4Arduino programming language. Snap4Arduino sends data to eCraft2Learn API at predetermined intervals, the learning analytics system uses then these data for analysis. 3D printing activities take place with Ultimaker 3D printers and the data that is possible to collect from the printer is described in this document.

The work presented here informs the development of the project towards facilitating the role of the teacher as a coach of the learning experience within the eCraft2Learn learning ecosystem. This also assist in supporting learners personalized learning paths. The development of the LA is deeply integrated within the digital platform of the system assisting also in supporting learners' personalized learning path, self-regulation, self-reflection and self-awareness throughout the five stage pedagogical steps (WP3). The learning analytics also shows innovative ways of using and representing learners collected pedagogical data (WP2) and further complements the development of the technical core of the eCraft2Learn learning ecosystem (WP4).

This document is structured as follows: in section 2 we provide a description of the learning analytics system data collection infrastructure showing all the data sources that the learning analytics system uses. Section 3 shows the details of the data that can be collected from the 3D printer. Section 4 shows the data that is collected from the UUI and the DIY electronics. Section 5 presents details about how the learning analytics system processes and analyses the collected data and provides visual representation to the teacher. Section 6 then provides a summary of the user feedback from the perspective of the visualizations that the system provides as well as how the teachers perceive the learning analytics system importance.

2 Learning Analytics Data Collection Infrastructure

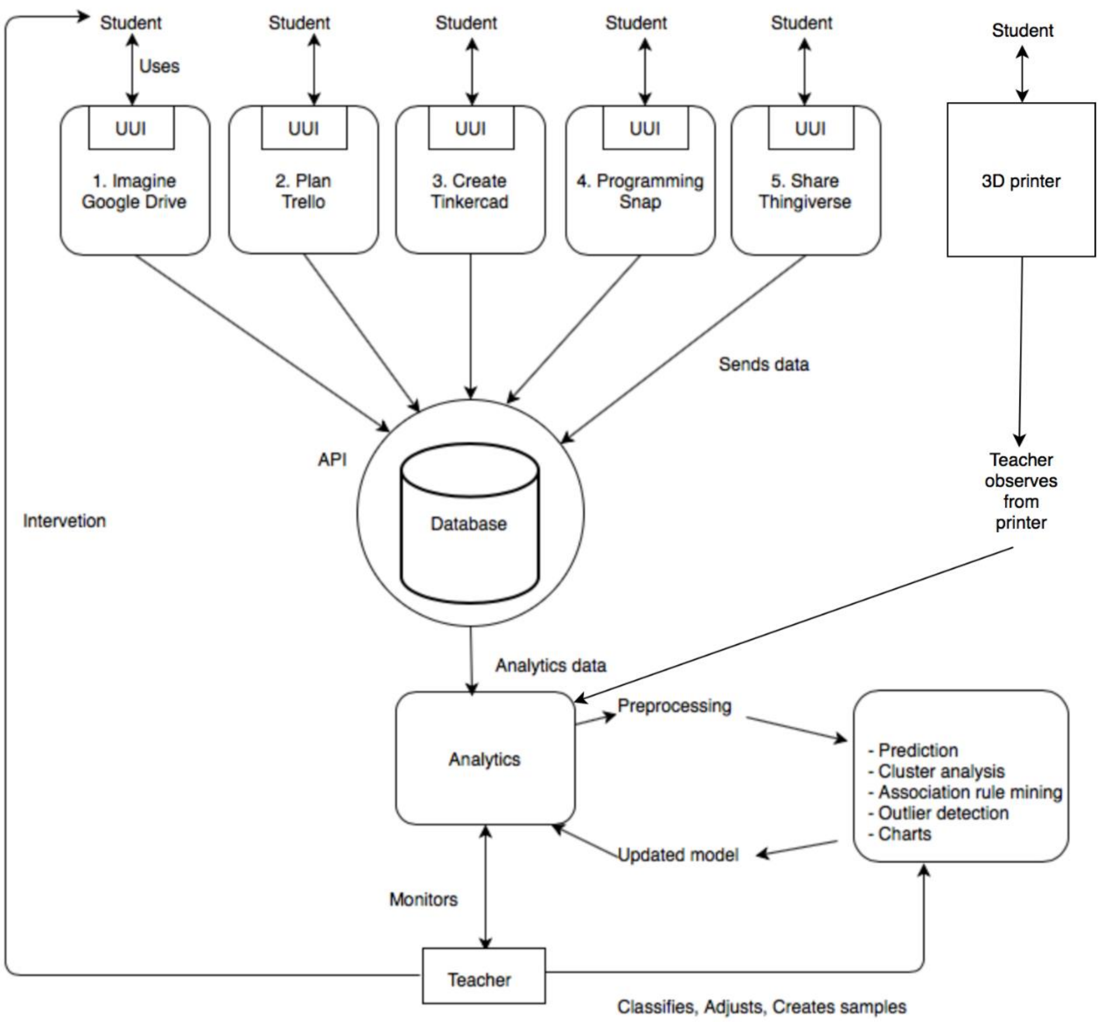


Figure 1. Infrastructure of DYI electronics and 3D printing from data collection to analysis

Since data that can be collected for the DIY electronics takes the shape of the code that the learners create to handle the components, we restrict the DIY electronics data collected to the programming process in Snap4Arduino. During the programming of computer-supported artefacts through Snap4Arduino, the eCraft2Learn API collects the students' code structure and send the collected dataset to the learning analytics system. In the case of 3D printing, we provide a guide to access the data from the printing process here. The data collection infrastructure associated with the DIY electronics (Snap4Arduino), 3D printing as well as each one of the tools supporting the five pedagogical stages of the eCraft2Learn learning ecosystem is shown in Figure 1.

The data flows automatically from the Unified User Interface (UUI) tools that support each pedagogical step (i.e., ideation/imagine, planning, creating, programming, sharing) into the learning analytics system at interval steps. For instance, when programming of DIY electronics using Snap4Arduino the data flow to the LA system as follows:

- Step 1 - Snap4Arduino sends data from the students' code to eCraft2Learn API in intervals
- Step 2 - eCraft2Learn API receives data from Snap4Arduino
- Step 3 - LA requests student data from eCraft2Learn API
- Step 4 - The data is analyzed by the teacher in LA

In addition, the data flow from 3D printer to the LA system can be divided into 3 steps:

- Step 1 - Teacher observes the log data from the printer
- Step 2 - Teacher applied the collected data as "Notes" for existing dataset in LA
- Step 3 - The data is analyzed by the teacher in LA

The steps described above can be seen in the Figure 1 as arrows. In Figure 1, the data analysis process in LA system is iterative.

The machine learning algorithms used in the LA system produce predictive models that can be utilized to support teacher's role as a coach when providing feedback and interventions when learnings are developing their projects, since the teachers know which students or student groups have achieved a certain level of performance. In addition, the UUI shows tips and scaffolding elements to the learning experience according to a performance of the students. This helps the students to gain the support they may require during the *making* process.

3D printing, as shown in this document, also logs data from the printing process. We show how to access the printer data, which can be used to make notes in LA for the students, having in turn an influence in the data analysis process. It is important to highlight that the process to use the learning analytics system is iterative and collaborative in nature; the teacher and the machine learning algorithms produce the predictive models together and the knowledge discovery from the *making* settings is produced during that process.

3 3D Printer Data Collection

In this section, we describe how to access the data from the Ultimaker 3D printers. The data collection process from the 3D printers is semi-manual. That is, after the teachers access and collects the log data off the printers, they can give the log data instances as input to the LA system. The system has available for each student or students group that the system is tracking, input fields called *"Notes"* where the teacher can add the logs from the 3D printing process, for instance. The inputted notes will have an influence in the analysis process. The notes and how to use them will be described in more detail in Section 5.1

3.1. Available Data

For both the Ultimaker 2+ and the Ultimaker 3 it is possible to collect the following basic run-time data:

Print Data

- Nozzle temperature
- Build plate temperature
- Print speed
- Flow rate
- Fan speed
- Retraction speed
- Retraction length

Usage Data

- Machine on (total in hours)
- Print time (total in hours)
- Material printed through nozzle (total in metres)

In addition to this, with the Ultimaker 3, you can also access:

- Reading errors
- Firmware errors
- Cura crash report
- Cura install issues

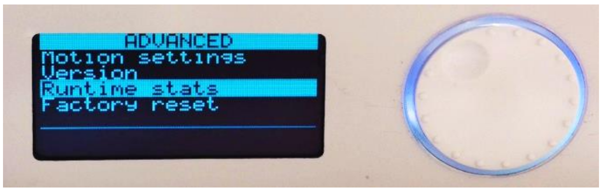
3.2. Data Collection with Ultimaker 2+

The data that can be collected can only be accessed manually, directly from the Ultimaker 2+. The data is collected as described below.

Access Usage Data. Using the scroll wheel, highlight and select "Maintenance" on the front user display.

Then select “Advanced”.

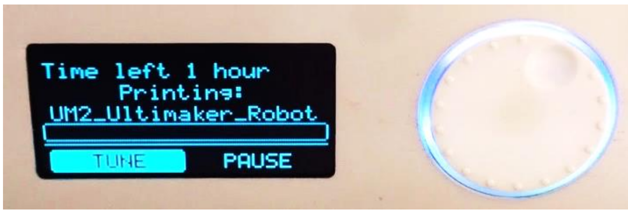
Finally select “Runtime stats”.



This will display the usage data since the last firmware upgrade or factory reset.

Access Print Data. The print data for a specific print, can only be collected whilst the Ultimaker 2+ is running the print. This data is not stored and can not be accessed at any other time.

Whilst a print is running, use the scroll wheel, highlight and select “Tune” on the front user display.



Select Tune, use the scroll wheel to scroll down the list of options, when an option is highlighted, the data for that option is displayed across the bottom of the display screen.

3.3. DATA COLLECTION WITH ULTIMAKER 3

There are three ways of collecting data from an Ultimaker 3 printer.

A. Download Log Files from the 3D Printer

This will allow you to collect the following data:

- Check reading errors
- Check firmware errors

Please note, all data will be reset each time the firmware is updated OR a factory reset is performed on the printer.

The log files can be downloaded directly from the 3D printer by following these steps:

Step 1: Insert the USB stick into the port on the front of your printer.

Step 2: Perform the following menu sequence:

System --> Maintenance --> Diagnostics --> Dump Files to USB.

Step 3: The files will then save onto the USB stick.

B. Download Log Files from the Cura Software

This will allow you to collect the following data:

- Cura Crash Report
- Cura Issues with Install

Please note, data will be reset each time the Cura software version is upgraded. The log files can be downloaded from Cura by browsing to find the Cura GUI log file on the 3D print station computer, this file will be located at:

WINDOWS: %APPDATA%\cura\<Cura version>\cura.log

or usually

C:\Users\\<your username>\AppData\Roaming\cura\<Cura version>\cura.log

OSX: $USER/Library/Application Support/cura/<Cura version>/cura.log

Ubuntu/Linux: $USER/.local/share/cura/<Cura version>/cura.log

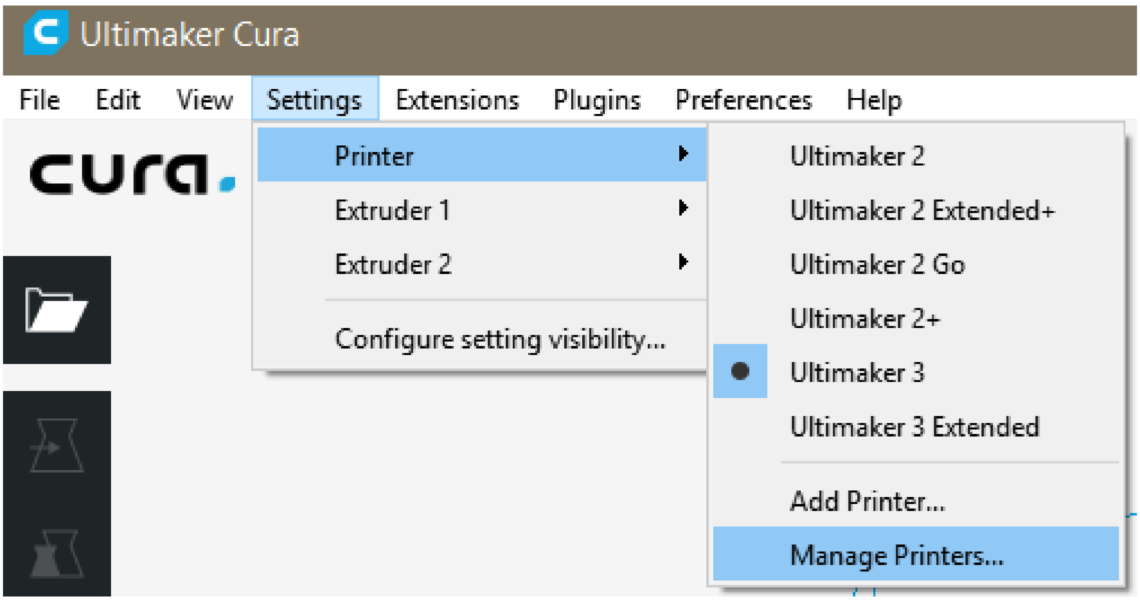
C. Extracting data from the “Monitor Data” function in the Cura software

This will allow you to collect the following data:

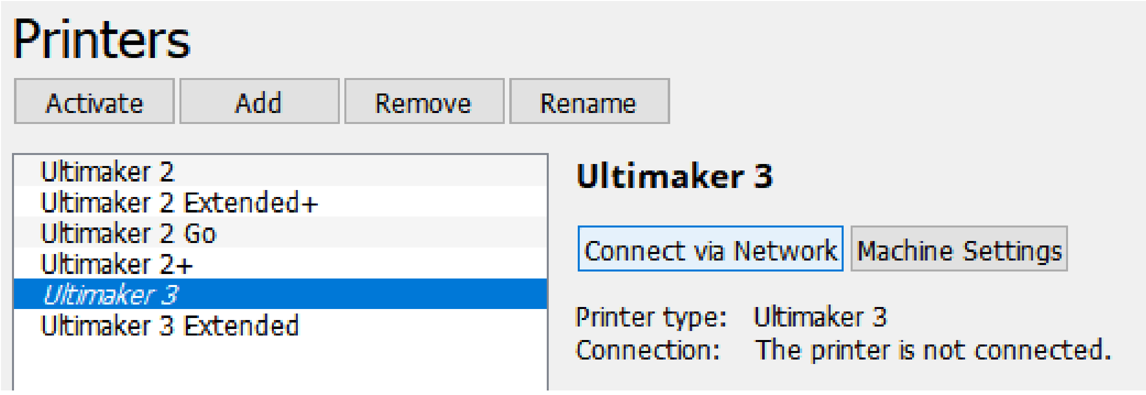
- Print Core and Build-plate temperature data
- Length of Print

Please note, data from here is only available for a point in time during a 3D print. The monitor data function in Cura can be used by following these steps:

Step 1: Connect to Ultimaker 3: Settings > Printer > Manage Printers



Step 2: Select Ultimaker 3 > Connect via Network



Step 3: Click on correct printer > connect

Step 4: Select Monitor

4 DIY Electronics Data Collection

4.1. Snap4Arduino

Snap4Arduino is version of the Snap! Programming language that allows interacting with almost all the Arduino types. Like Snap!, it is a graphical environment particularly suited to introduce programming; however it allows also several advanced functionalities such as procedures

4.1.1. Enabling Communication between Arduino Board and Snap4Arduino Programming Environment

To connect an Arduino board to Snap4Arduino programming environment within the unified user interface (https://ecraft2learn.github.io/uui/), it is required to upload a specific firmata on the Arduino board beforehand. The firmata allows the Arduino board to communicate with the Snap4Arduino software. To upload the firmata proceed as follows: close Snap4Arduino (if open), connect the Arduino board to the computer (or RPi3) through a USB port, open the Arduino IDE environment clicking on the dedicated tile (Figure 2).

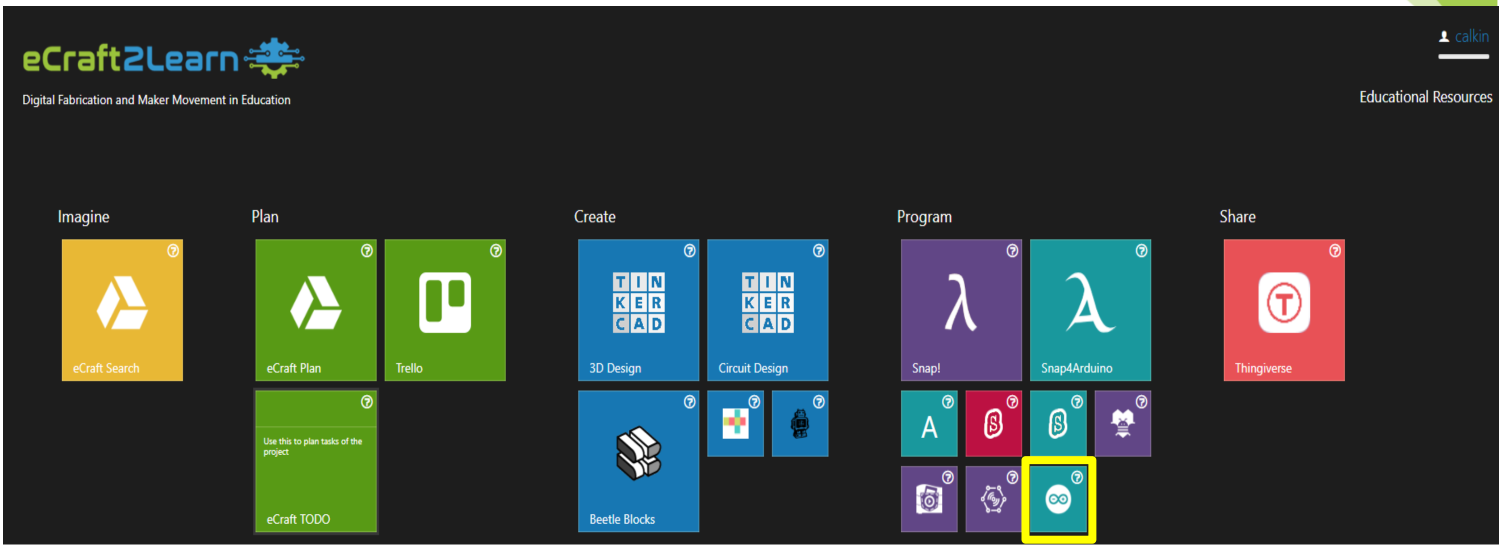


Figure 2. The Arduino IDE tile highlighted in yellow square within the unified user interface (UUI)

In the Tools menu, select the board version and the serial port where the board is connected. After that, go to “Examples”-> “From Libraries”-> “Firmata”-> “Standard Firmata”, select the file “Standard Firmata” (see Figure 3) and upload the file clicking on the arrow icon. When the upload finished, Arduino is ready to communicate with Snap4Arduino.

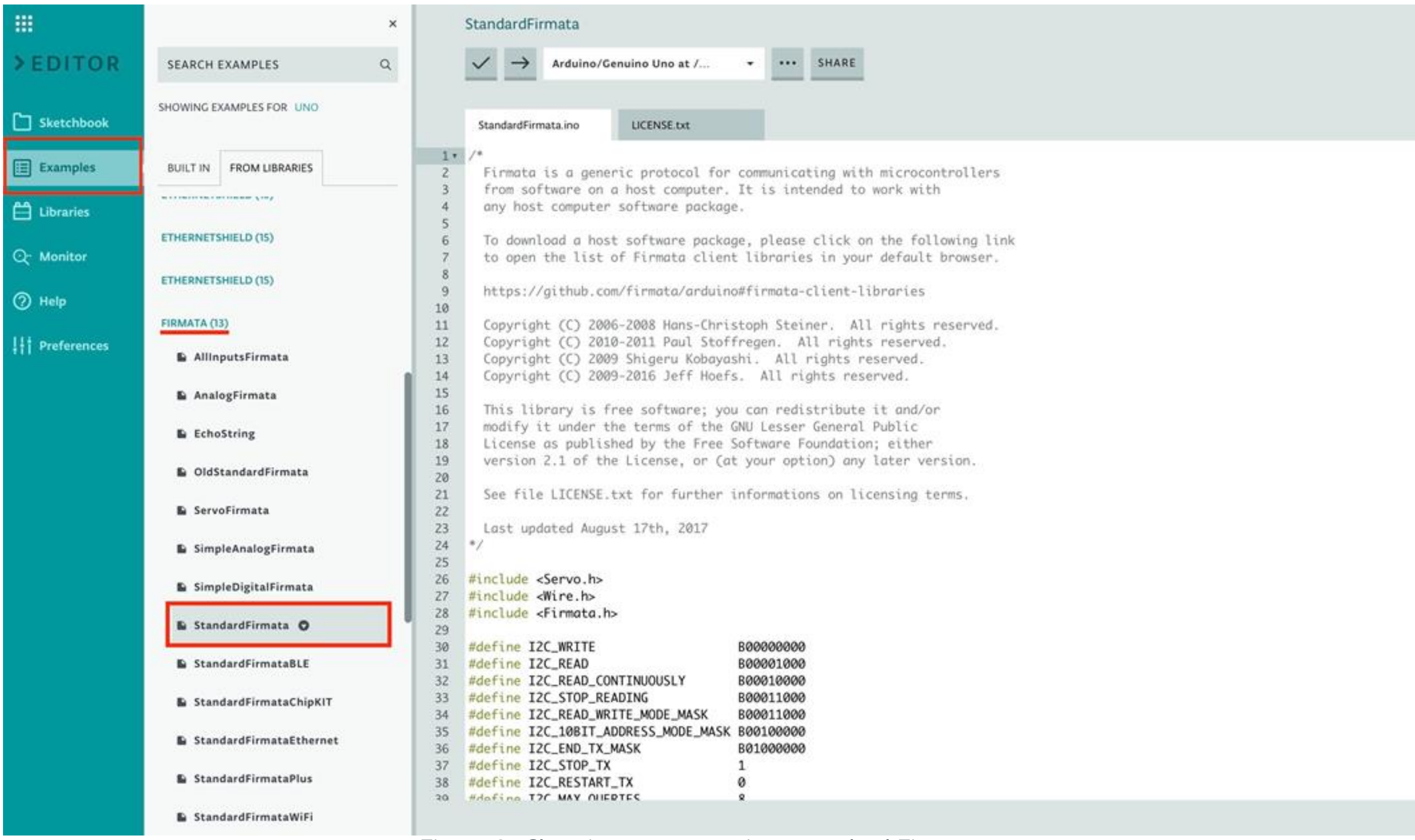


Figure 3. Choosing an appropriate standard Firmata

4.1.2. Launching Snap4Arduino

Click on "Snap4Arduino" tile in start page of the UUI. Snap4Arduino editor will open in a new UUI window (Figure 4). Click on the "Arduino" set of blocks and select "Connect Arduino". After this, the Arduino board can be programmed using Snap4Arduino.

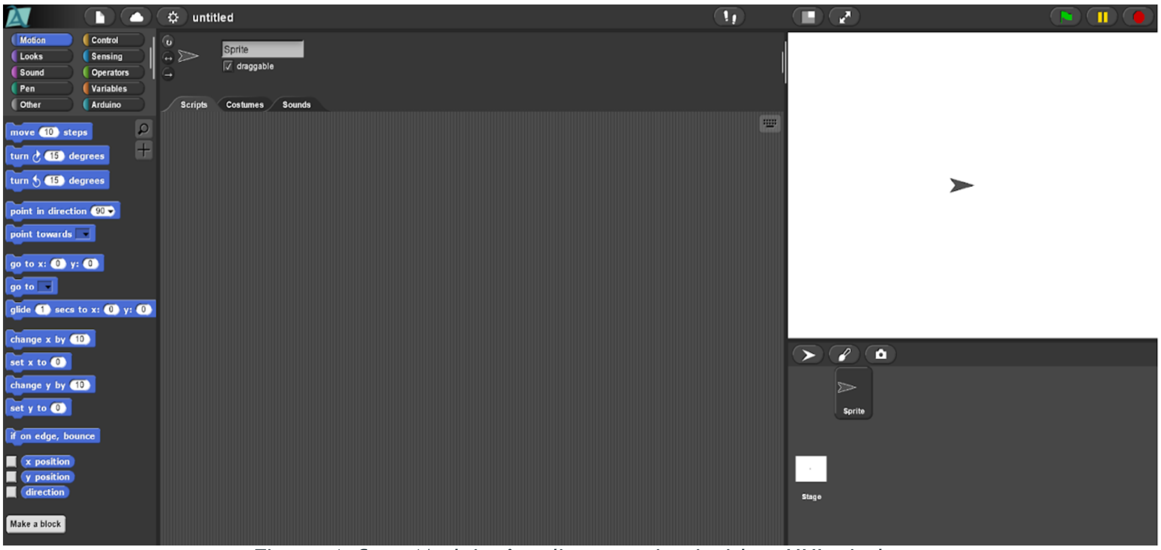


Figure 4. Snap4Arduino's editor running inside a UUI window

4.1.3. Programming with Snap4Arduino

Snap4Arduino is very similar to Snap! where different sets of programming blocks are arranged in different groups in the panel to the left of the editor. The students can create their programs by dragging and dropping the blocks to the centre of the editor. The Figure 5 contains an example of how Snap4Arduino can be used to make the internal Arduino board LED blink through the pin 13 of the board.

Figure 5. To blink a LED using Snap4Arduidno

4.2. From a Graphical Programming Environment to a Textual Programming Environment

It is possible to translate the program created with Snap4Arduino in the block-based graphical interface into the Arduino text-based IDE language. To do this, go to the sheet icon and select *"New Arduino translatable project".* This functionality allows creating a script using only the commands compatible with the Arduino IDE environment. It is necessary that the first command is "When 'green arrow' is clicked". After creating the program, right click and select "export as Arduino sketch".

5 Learning analytics for 3D printing and DIY electronics

eCraft2Learn learning analytics (LA) system aims at analyzing the digital traces that the Unified User Interface (UUI) and its tools collect from the students during the sessions. Ultimately, the ambition is to allow the teacher to use the LA to deepen the understanding in the context where the projects take place either in formal or informal settings. LA is organized as a collection of several data mining tools found in a framework including (1) classification, (2) cluster analysis, (3) outlier detection, and finally (4) association rule learning. These algorithms are widely used in other similar tools of learning analytics and educational data mining. All data mining algorithms used in LA have been implemented so that no third party APIs have been used. Thus, a school, or other education provider can use the solution, without considering the limitations due to third party API solutions.

Besides the above-mentioned more traditional learning analytics methods, the LA system within the eCraft2Learn learning ecosystem introduces a white-box approach to classification and cluster analysis. White-box approach in contrast to traditional black-box approach (Neural Networks for instance) opens the data mining process by using white-box algorithms, which are easy for a human to comprehend and to interpret (decision trees, for instance). The model produced by these algorithms will be visually represented to the teacher. The teacher will have the ability to adjust the models, which are the built classifiers and cluster analyzers and hence, the classification and cluster analysis process itself produces domain knowledge as opposed to the results of the cluster analysis and classification. Such approach aims at deepening the understanding of the learning process.

5.1. Data collection

The making of DIY electronics incorporate the process of building and setting up the Arduinos, connecting the preferred components and wiring them to the Arduino units. Later these units will be programmed with Snap4Arduino (S4A) programming environment, which is a fork of Snap! visual programming language. The programs are built by dragging and dropping blocks to the scripting area in a Scratch-like manner. S4A is integrated with the UUI and because the UUI contains the source code of S4A (an open source project), a logging script has been added as an additional component for S4A. The logging script parses the scripts of students and sends them to eCraft2Learn API via REST-call where the data is stored into the database. The parsing of the script keeps track of the different kinds of blocks that are being used in the students' scripts. For instance, some of the blocks are used to control the Arduino boards such as writing data to digital pins, or reading an analog value from an analog pin. Thus, those blocks will be referred to as "Arduino blocks". Control structures such as loop blocks and conditional statement blocks are referred to as "Control blocks". The different kinds of blocks of the parsed script are then counted and the vector containing the amount of these different block types are stored in the database.

The Learning analytics (LA) system receives the blocks used in Snap4Arduino (the "scripts") and then displays the structure of the scripts within the table for the teacher as seen in Figure 6.

motion	looks	sound	pen	control	sensing	operators	arduino	variables	users	timestamp	notes	Predict	Anomaly
0	0	0	0	152	0	16	186	0	marika123	Wed Apr 18 2018 11:45:05 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	0	0	0	379	0	95	393	96	Appelsiini123	Wed Apr 18 2018 11:48:16 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	108	0	0	1201	30	384	457	1138	iisuliisu	Wed Apr 18 2018 11:50:55 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	0	0	0	162	0	15	293	0	glittervaahtokarkkiyksisarviset	Wed Apr 18 2018 11:53:05 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	0	0	0	16	0	2	8	0	SariN	Mon May 14 2018 13:12:40 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	1523	40	0	7834	302	3880	683	11968	hemulit	Mon May 14 2018 13:14:26 GMT+0300 (EEST)	0	Not included	Not an anomaly
16	121	0	0	1020	24	611	844	941	buu	Mon May 14 2018 13:54:44 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	301	0	0	1392	60	728	50	2268	nimettu00f6mu00e4t	Mon May 14 2018 14:22:47 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	768	0	0	4633	160	2176	576	6560	nimetu00f6mu00e4t	Wed May 23 2018 08:32:45 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	0	0	0	0	0	0	8	2	testi-tapani	Fri Jun 08 2018 19:17:27 GMT+0300 (EEST)	0	Not included	Not an anomaly
0	0	0	0	0	0	0	3	0	tapanit	Tue Jun 19 2018 09:43:39 GMT+0300 (EEST)	0	Not included	Not an anomaly

Figure 6. Scripts of the students or students groups that are using Snap4Arduino programming blocks

The tools of LA can then be used to analyze the students who have used S4A. Since the logging script for S4A counts the number of the blocks within the scripting area of S4A, the amount of the different type of blocks is cumulative. Thus, in LA the teacher can also view the average usage of certain types of blocks, which in turn gives the teacher an idea on what kinds of blocks on average the students have used. This is useful in the situations where some of the students have used S4A for a long time and some students only for a short while e.g., Figure 6. Long time means that the students have used more time to program versus the students who have only programmed for a short while.

When the students are building and wiring up the Arduinos, the data collection during that process requires the teacher's attention. In LA, there is a possibility for the teacher to modify the dataset, which is loaded from the server on teacher's request. Each of the datasets contain cells where the teacher can add *Notes*. The Note cells are for the situations where the students are not using the UUI and thus the data collection through the logging of the students' actions is not possible. In addition, the teacher can make notes both numerical and textual according to the teacher's perception of the students' progress while they are wiring and assembling the electronics. The notes that the teacher gives to the students have an influence in the analysis process as the notes will be treated as one of the features within the dataset while processing the data. How notes are given for a certain student can be seen in Figure 7.

users	timestamp	notes
marika123	Wed Apr 18 2018 11:45:05 GMT+0300 (EEST)	10
Appelsiini123	Wed Apr 18 2018 11:48:16 GMT+0300 (EEST)	-4
iisuliisu	Wed Apr 18 2018 11:50:55 GMT+0300 (EEST)	25
glittervaahtokarkkiyksisarviset	Wed Apr 18 2018 11:53:05 GMT+0300 (EEST)	-1
SariN	Mon May 14 2018 13:12:40 GMT+0300 (EEST)	0.2

Figure 7. The teachers' notes in the LA system. Teachers can give notes to the datasets from the DIY electronics

The type of the notes is not restricted to any required values and can be anything that the teacher decides to. The notes are subjective measures, and depend highly on the teacher judgement. In addition, their content depends on what kinds of observations the teacher makes from the time that the students are building and wiring up the Arduinos.

5.2. The Analysis Process

The analysis process of the data collected from the DIY electronics infrastructure and 3D printing begins when the teacher selects the session from the LA. The sessions stand for the projects that take place in both formal and informal settings. After the teacher is connected to the session, he/she can decide what is the source of data from the student will be analyzed. Both, data from 3D printing and from DIY electronics infrastructure (UUI) can be selected. The selection of the data source loads the dataset from the eCraft2Learn API and is shown to teacher in the form of a table. The table contains individual actions/events from such sources such as timestamps and used programming blocks for that time frame (DIY electronics). If the teacher chooses to show the dataset instances for a certain student, the teacher can group the dataset instances by the users and thus, the table contains all of the logged data instances for a certain student or a group of students. The teacher can also filter the dataset based on temporal information i.e., within a time window. This functionality is relevant for instance, if the session lasts for days and the teacher is only interested in specific dataset instances within this long session.

After manipulating the dataset, the teacher can use several methods to process and analyze the dataset. Such methods are *classification*, *cluster analysis*, *outlier detection* and *association rule learning*. Classification and outlier detection are both supervised learning approaches where the teacher is required to give labels to some dataset instances.

For **classification**, three labels are defined: “Performing well”, “Neutral performance” and “Not performing well”. The classification model produces a decision tree classifier, which is presented to the teacher as seen in Figure 8. The teacher can adjust the visualization of the decision tree. That is, the teacher is able to change the rules of the produced model and have an influence in the classification process even after labelling the students. The adjusting process is iterative and the pedagogical value of the adjustments comes when the decision tree algorithm and the teacher in collaboration learn to decide what means when someone is performing well or not.

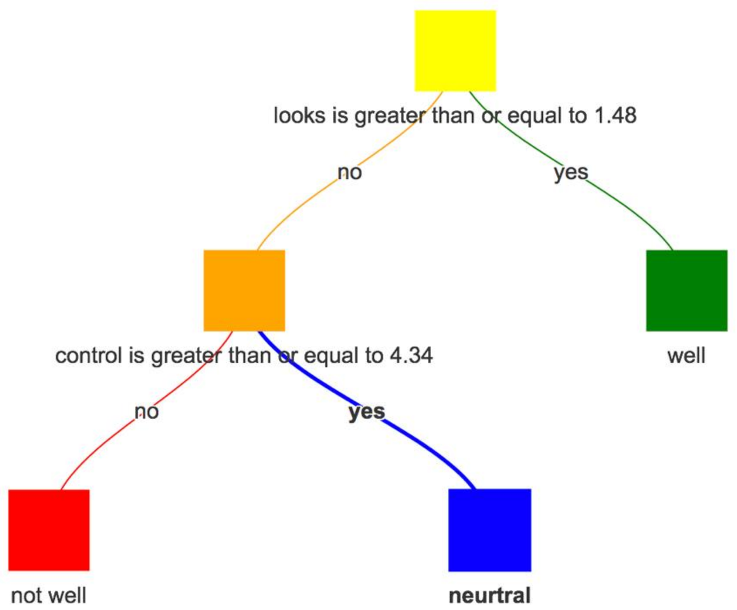


Figure 8. Rendered decision tree classifier built by a teacher

For **outlier detection**, there are two labels “Outlier” or “Not an outlier”. After selecting the labels, the teacher selects the tools, then the algorithm processes the labelled dataset instances and produces a model that can predict the performance of the students from new instances of data. The outlier detection will produce a table containing the dataset instances where the outliers are shown in red color as seen in Figure 9. The table represents the previously manipulated dataset and shows the actions that the students took during the 3D printing and DIY electronics process. During using the LA system, the dataset evolves and the teacher can update the dataset from the eCraft2Learn API anytime. Thus, the models produced by the teacher will also be able to make predictions from future data.

motion	looks	sound	pen	control	sensing	operators	arduino	variables	users	timestamp
0	0	0	0	4.34	0	0.46	5.31	0	marika123	Wed Apr 18 2018 11:45:05 GMT+0300 (EEST)
0	0	0	0	5.57	0	1.4	5.78	1.41	Appelsiini123	Wed Apr 18 2018 11:48:16 GMT+0300 (EEST)
0	1.38	0	0	15.4	0.38	4.92	5.86	14.59	iisuliisu	Wed Apr 18 2018 11:50:55 GMT+0300 (EEST)
0	0	0	0	4.26	0	0.39	7.71	0	glittervaahtokarkkiyksisarviset	Wed Apr 18 2018 11:53:05 GMT+0300 (EEST)
0	0	0	0	1.6	0	0.2	0.8	0	SariN	Mon May 14 2018 13:12:40 GMT+0300 (EEST)
0	22.07	0.58	0	113.54	4.38	56.23	9.9	173.45	hemulit	Mon May 14 2018 13:14:26 GMT+0300 (EEST)
0.2	1.48	0	0	12.44	0.29	7.45	10.29	11.48	buu	Mon May 14 2018 13:54:44 GMT+0300 (EEST)
0	20.07	0	0	92.8	4	48.53	3.33	151.2	nimettu00f6mu00e4t	Mon May 14 2018 14:22:47 GMT+0300 (EEST)
0	12.39	0	0	74.73	2.58	35.1	9.29	105.81	nimetu00f6mu00e4t	Wed May 23 2018 08:32:45 GMT+0300 (EEST)
0	0	0	0	0	0	0	4	1	testi-tapani	Fri Jun 08 2018 19:17:27 GMT+0300 (EEST)
0	0	0	0	0	0	0	3	0	tapanit	Tue Jun 19 2018 09:43:39 GMT+0300 (EEST)

Figure 9. Outlier detector marks outlier students with red

The teacher is also able **to cluster** the dataset instances (each instance is a student). The cluster analysis method of the LA system uses a tree-like neural network where each of the leaf nodes of the tree represent a cluster. The dataset instances will be traversed through the tree from the root to the leaf nodes. Before the actual clustering, the nodes of the tree will be adjusted based on the dataset instances. The tree-like presentation of the model clusters the most similar clusters as a sibling and thus from visualization that is being rendered to the teacher as seen in Figure 10, the teacher can view which of the student belong to the same cluster but also which clusters are more similar than the others.

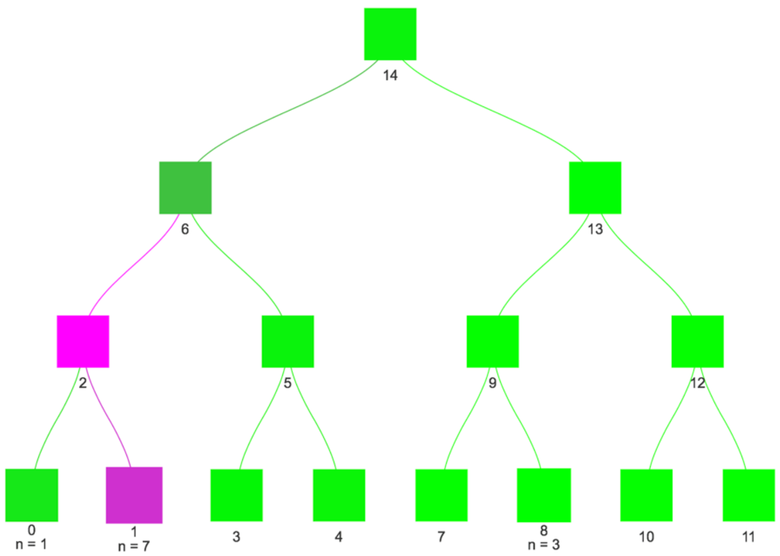


Figure 10. A visual representation of the tree where leaf-nodes are clusters and 'n=…' means the number of students within a cluster

For the **association rule learning**, the algorithm produces a graph (Figure 11) where the teacher is able to view which kinds of events usually occur together. In the produced graph, the vertices represent logged events (type of code in Snap4Arduino, heat of 3D printers etc.) and the directed edges represent the relationships between the events such that the source of an edge is an event that usually triggers the destination event of an edge. Such events include for instance, the kind of blocks that are usually used in the same script in Snap4Arduino (DIY electronics).

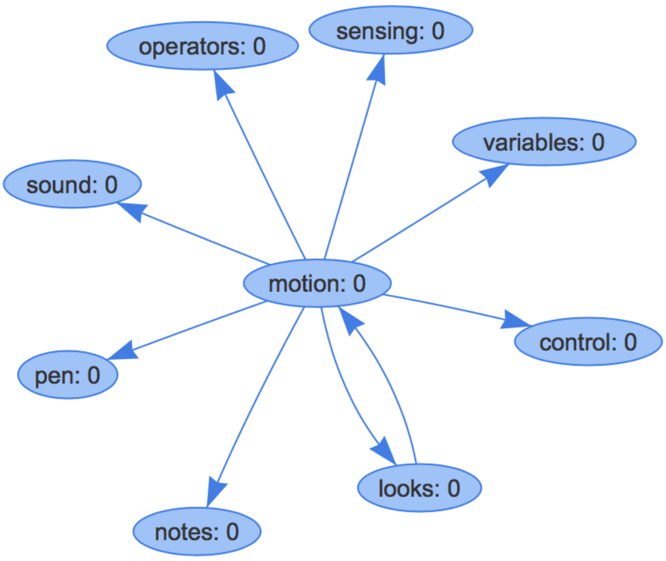


Figure 11. Directed graph rendered to the teacher with the association rules

6 User feedback

6.1. Feedback to the User

The feedback that the teachers receive during the analysis process is the visual presentation of the manipulated data instances within the algorithms (see Section 5). The teacher is able to detect the outliers based on the visual feedback, understand the reasons why the decision tree algorithm classifies some students or students groups according to some pre-given labels and see the similarly behaving students through the tree-like visualisation. Combining the feedback given by LA system to the teachers' own domain-knowledge, the predictive models can be created collaboratively between the LA algorithms and the teachers.

The feedback received by the students is based on the UUI and how the UUI adjusts itself. The teacher and LA build models from the student data and those models adjust the UUI according to the needs of the student during the DIY electronics and 3D printing activities. The UUI scaffolds some of the elements based on the model of the students using the UUI and gives tips the performance of the student is not as expected.

6.2. Users Perceived Usefulness

The teachers who used the system during the pilot sessions gave their feedback about their perceptions on the usefulness of the LA system. The open approach where the machine learning models of LA are presented to the teacher gained positive feedback since the teacher could adjust the models. The teachers that were interviewed pointed out that the approach where the students are classified collaboratively between the teacher and the machine learning algorithm is easier to trust. Also, the teachers pointed out in their feedback that the approach leads to knowledge discovery which would be difficult to discover otherwise. However, according to the feedback of the teachers, the system is much harder to learn than a traditional learning analytics system due the open nature of the implemented system. In practice, it requires more effort and is more technical in nature.

Nevertheless, the user feedback highlights that in the case of DIY electronics and 3D printing, the data collection and learning analytics have the following characteristics:

- Even though it is more laborious to learn to use than a traditional black-box tool, the current system leads to knowledge discovery which could be difficult in a black-box system (a system that does not show how the data is processed or that allow the user modify the models)
- It can be used to observe the process of DIY electronics
- Although it requires effort to understand, it is helpful when the teacher is not able to view the Snap4Arduino code from the monitor of the students
- It is easier to trust the predictive models than in the traditional black-box tool

7 Conclusion

In this document, we have demonstrated how the teachers can access the data that Ultimaker 3D printers collect. This data can be used to deepen the understanding of the teacher in the making activities in both formal and informal settings. We have also described which kind of processes are involved in DIY electronics and what kind of data is being collected while the students are being involved in DIY electronics during the projects. The Learning Analytics system (LA) can be used to analyze such data in order to enable interventions during the making activities through multiple tools described in this document. LA uses an open-ended approach where the teacher is constantly involved in the predictive model building. The knowledge discovery from LA is enabled through the collaborative process of the analysis algorithm and the interaction between the teacher and the built model.

Snap4Arduino collects data from DIY electronics process in the intervals. The data consists of the structures of the scripts that the students are building. The user feedback (the teacher feedback) has been collected from the pilots and the teachers that have been involved argue that the open-ended approach to learning analytics leads to the knowledge discovery and is especially useful in the DIY electronics phase where the teachers are able to monitor the code structure of the students from the LA.