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

Deliverable D4.3

A software solution for the educational extension (v1)

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:	http://www.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: **D4.3**

Title: **A software solution for the educational extension (v1)**

Lead beneficiary: **MDH**

Work package: WP4

Dissemination level: Public (PU)

Type Report (R)

Due date: 30.09.2017

Submission date: 30.09.2017

Authors: **Afshin Ameri** [MDH], **Ken Kahn** [UOXF], **Andrea Alessandrini** [LNU], **Emanuele Menegatti** [UNIPD], **Baran Çürüklü** [MDH], **Ilkka Jormanainen** [UEF], **Tapani Toivonen** [UEF], **Michele Moro** [UNIPD], **Francesca Agatolio** [UNIPD]

Contributors: All consortium partners

Reviewers: **Sophia Manika** [TECHNOPOLIS]

Version Control

Version	Date	Person in charge [Organization]	Changes	Quality Assurance
1	08.06.2017	Baran Çürüklü [MDH]	Guidelines, layout and critical parts	Afshin Ameri [MDH]
2	23.09.2017	Afshin Ameri [MDH]	Sections filled, figures added	Baran Çürüklü [MDH]
3	29.09.2017	Afshin Ameri [MDH]	Final version. Refinements based on reviews	Sophia Manika [TECHNOPOLIS]

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 (D4.3) is in the format of a sequence of three demonstrations (corresponding to versions v1 - v3). This report is part of the first demonstration for v1/M9. The intention of this deliverable report is to provide an insight into the first version of the educational extension and its integration with the unified user interface (UUI).

This deliverable includes accompanying videos that demonstrate the integration of educational extension with the UUI (D4.4). In this version, the focus of development has been on creation, programming and sharing tools as these are necessary tools for the first pilots. Based on the collected data from first pilots, further improvements and new functionalities will be added to later versions.

1 INTRODUCTION

Following the footsteps of deliverables D4.1 and D4.2, this is the starting point of design and implementation of the pedagogical framework, including the 5 stage learning methodology (See D3.1, D4.1 and D4.2) from a technical perspective. Thus, in this document the first implementation of eCraft2Learn's educational extension, which will act as a glue to connect different tools used in the 5 stages of eCraft2Learn's learning methodology is implemented. In addition, this implementation will also extend the functionalities of some these tools to support the learners with more advanced features, such as artificial intelligence (AI) services and augmented reality visualizations, in their journey towards creating and learning.

1.1. ECRAFT2LEARN LEARNING METHODOLOGY

eCraft2Learn is an ecosystem which is based on craft and project based teaching methods. This methodology includes 5 stages, which together guide the learner from an idea to developing and showcasing, through sharing, the final solution (Figure 1).

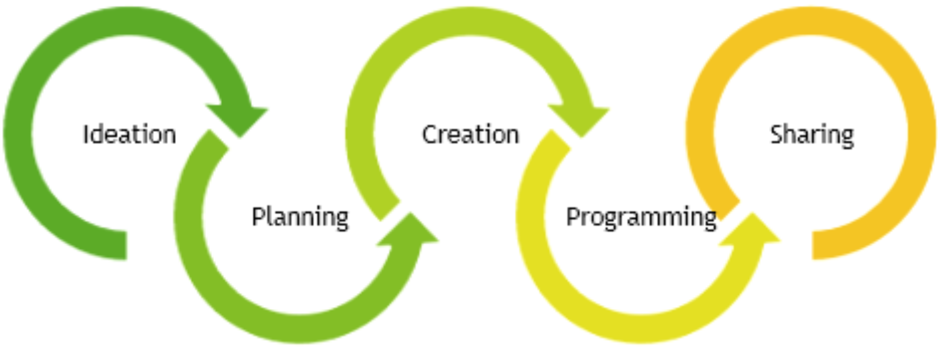

Figure 1 - eCraft2Learn craft and project-based methodology (the 5 stages)

- **Ideation**: The learners explore the world to find challenges. This exploration can be in the physical or virtual world (i.e. online communities).
- **Planning**: The learners collect information and material. They also start making project plans.
- **Creation**: Through a co-design and co-creation process, the learners start creating their solutions. This stage might involve many different technologies such as do-it-yourself (DIY) electronics, visualization, simulation and 3D printing.
- **Programming**: Learners add functionality to their crafted artefacts through high-level programming languages.
- **Sharing**: By sharing the solutions on online communities, the learners can learn from other projects, while receiving feedback from designers, engineers and programmers.

Each of the aforementioned stages requires a different set of tools and materials. Developing a central place (software) for managing these stages is the aim of the educational extension in eCraft2Learn. Such a solution will make it easier to manage the learners' co-creation project work and also can provide then additional tools to improve their projects even further.

2 EDUCATIONAL EXTENSION

In order to have a clear view of the goals for the educational extension, here some information from eCraft2Learn's project proposal is presented. The role of the educational extension is to:

1. **Connect to UUI**: The first version of the educational extension, is fully integrated into the UUI (D4.4) and therefore the accompanying demo videos are shared for both deliverables.
2. **Provide communication with hardware equipment such as 3D printers and DIY electronics**: Communication with electronical hardware (Arduino, Raspberry Pi) is present in this version. 3D printing extensions will be presented in later versions of the software.
3. **Provide project planning tools for learners and teachers**: The UUI provides the users with project planning tools. The integration of these tools with educational extension is under development but present in the UUI.
4. **Provide analytics, debugging and feedback from hardware to learners and teacher**: Providing the data analytics tools with required information from the users (i.e. user actions) is present in this version.
5. **Provide access to AI cloud services**: In current version, users can use AI cloud services from Google, IBM Watson and Microsoft for a variety of tasks such as speech and image recognition.
6. **Provide high-level components (such as libraries for data visualization and visual programming)**: Most of the programming tools in UUI are visual programming languages and the aforementioned AI services are also presented as specific visual programming blocks. The educational extension also includes an augmented reality viewer for visualizing 3D designs in real world before printing them.
7. **Provide smart sharing modules, both for showcasing learner results and accessing open communities**: Using online communities, learners can now share their 3D designs and projects with the world and each other.

As a main goal, the educational extension should provide the learners with means to follow the 5 stage learning methodology of eCraft2Learn with ease. In order to demonstrate all the steps from ideation to planning, creating, programming and sharing, the rest of this document is divided into sections for each of these 5 stages. In each section, we discuss the tools supporting that step and their relation to the educational extension. We also provide a section representing the UUI and its integration with the educational extension.

3 UNIFIED USER INTERFACE

The UUI (accessible here: https://afsheenam.github.io/UUI) is the main hub for accessing the educational extension tools. It is presented in detail in D4.4 and a user manual is also available in D4.5. As seen in Figure 2, the interface contains 5 clusters of tiles that represent the 5 stages of eCraft2Learn methodology. In the following sections, we present each cluster and its tools that are integrated into the first version of educational extension.

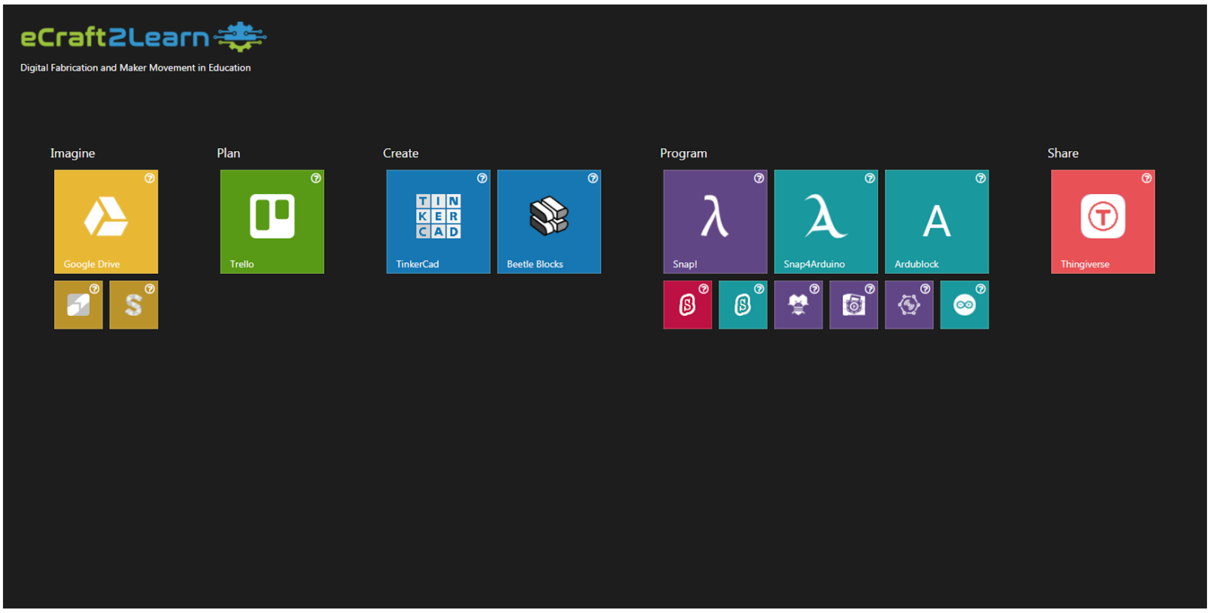

Figure 2 - The UUI is a 2D space. In this first version, the most evident feature is the clusters of set of tools for each of the 5 stages (Ideation/Imagine, Plan, Create, Program, Share) of the eCraft2Learn craft- and project-based learning methodology. As can be seen all tool are visible and available, at the same time the 5 stages of the methodology is evident.

4 TECHNOLOGIES FOR THE ECRAFT2LEARN LEARNING METHODOLOGY

Although the prime objective I to develop these technologies for the 5 stages of the abovementioned learning methodology the proposed solution is flexible enough to allow the usage of a sub set of the technologies, or less ridged learning settings.

4.1. IDEATION (IMAGINE) STAGE

In D3.1 the ideation stage is described as (note that the ideation stage is called Imagine in the UUI Fig. 2), since it was assumed that the latter term would be easier to understand by the learners):

> *"Ideation through world exploration: In order to explore the kinds of challenges that students or others face in their daily lives, a student could explore the world physically (e.g. taking pictures, exploring situations outside the classroom, newspapers, etc.) or virtually (e.g. through online support community discussion, online news, documents, local news/websites) and then decide what their challenge/problem will be. This process is also guided by the STEAM coach."*

Ideas can form anytime and anywhere, therefore the tools provided for this stage should be able to cover different situations and contexts. Note that, having access to certain digital tools (not part of the set of tools, such as digital cameras, mobile devices, voice recorders, etc.) is essential, since they can provide the learner with options to log and ideate on a problem. For example, they can take a photo of an object that they think can be designed better or improved. In addition, these tools can be carried and used in most locations. The tools provided in the ideation stage provides the learners services to save and use the data, images, recordings that are collected with the abovementioned digital tools. Even if digital tools are not available, the learners might still use pen and paper to take notes and later add them to eCraft2Leaern ideation tools by scanning the documents. This means that the ideation tools should be able to provide the learners with collaborative tools, support for different file types and wide availability. This is achieved by using Google Drive service and Coggle software.

Google Drive is a well-known service for storing different file types and it also provides editing tools for most document types, therefore it can provide a shared space for the members of a project to store images, voice recordings, etc. that can later be used for the idea generation.

The educational extension also uses Google Drive to store auto-save files from other tools that a learner is using in eCraft2Learn environment, i.e. Snap! programming environment (see Sect. 4.4.2). This way the learners, not only have access to their ideas through this tool, but also use the same space to any project file related to it.

Coggle is a web application that allows the users to create mind maps and ideas. They can discuss and improve the ideas collaboratively in real-time using this tool. Creating mind maps is done by adding text, images and links to the mind maps. Any group member or the STEAM coach can add, remove or modify the document or comment on specific parts of it. This way the group can collaboratively improve their ideas and prepare them for development. A simple screenshot of a min map in Coggle is presented in Figure 3.

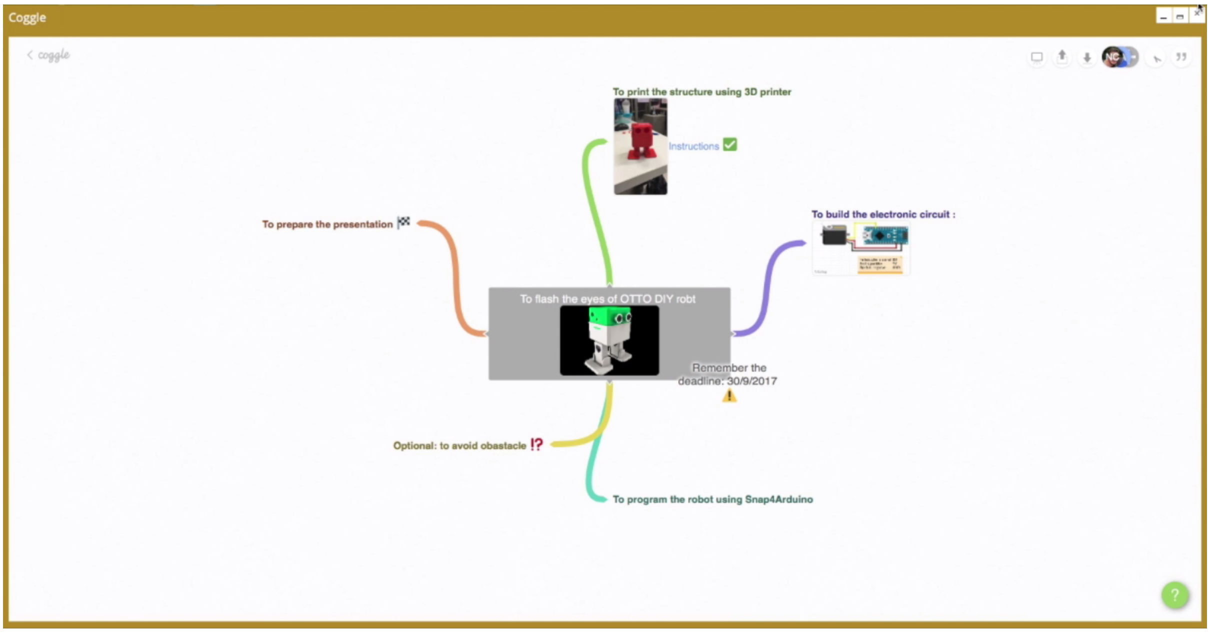

Figure 3 - Ideation in Coggle

4.2. PLANNING STAGE

The planning tool should allow the learners to define a work timeline, collect material (digital and non-digital) and assign roles to group members. To achieve these goals, the UUI makes use of Trello. This is an easy to use web application designed for collaborative project management. The work is divided into boards, and each board consists of several lists (or cards) that represent different stages of the work or different tasks. Each task can have a deadline, can be assigned to a different group member and can also have different material (i.e. from ideation stage) attached to it. Trello also has direct communication with Google Drive, so importing and attaching ideation files which are stored in Google Drive in ideation stage is straight forward.

Trello makes it easy for the teachers to supervise the project as well. Teachers can be added to the projects as group members and can comment on different aspects of the plan. They can even subscribe to the projects and be notified of any changes as they occur. Figure 4 shows a screenshot of Trello with goal, to do, doing (in progress) and done cards. Below each task the initials for group members that are responsible for that task is also visible.

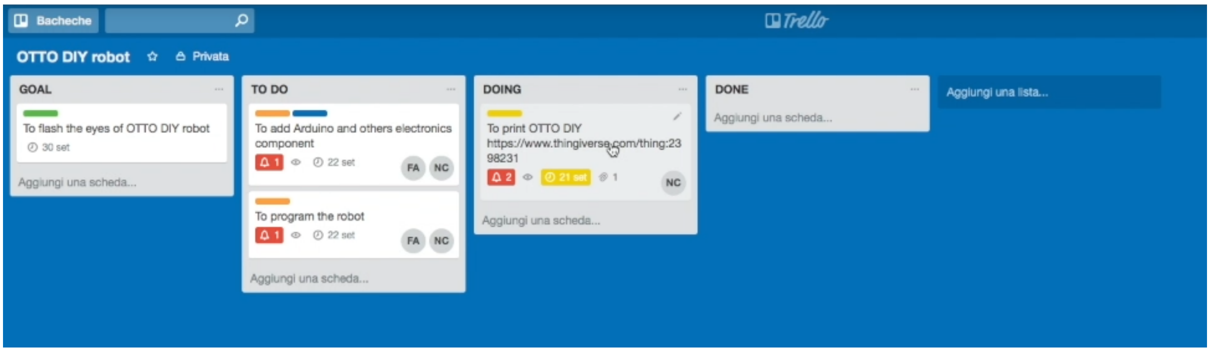

Figure 4 - Project Planning with Trello

4.3. Creation Stage

In this stage, the learners create computer-supported artefacts. This involves creating DIY electronics and 3D objects and simulation of DIY electronics. To achieve these goals, the educational extension should provide the learners with tools that can be used for 3D modelling, DIY electronics design and simulation and 3D printing. In this demonstration, we focus on 3D modelling and hardware design and simulation, all of which can be achieved using TinkerCad family of products.

4.3.1. 3D Modelling and Printing

TinkerCad is a free, web-based 3D modelling and printing web application from Autodesk which is aimed at young learners and makers. It has a shape-based 3D modelling interface, where learners can create new shapes by (i) manipulating basic geometrical objects, (ii) performing logical operation, such as conjunction and disjunction, and (iii) using pre-built 3D objects, e.g. Arduino cases. TinkerCad can also generate STL files for 3D printing from user designs.

The software allows for collaborative work on 3D designs and the learners can invite other group members to join them in tinkering the 3D models. The invitations are in the form of access links that can be sent to other group members through other tools in the eCraft2Learn environment such as Trello. Sharing 3D designs is also easy when using TinkerCad, as it can directly share the designs on “Thingiverse” (see Sect. 4.5.1).

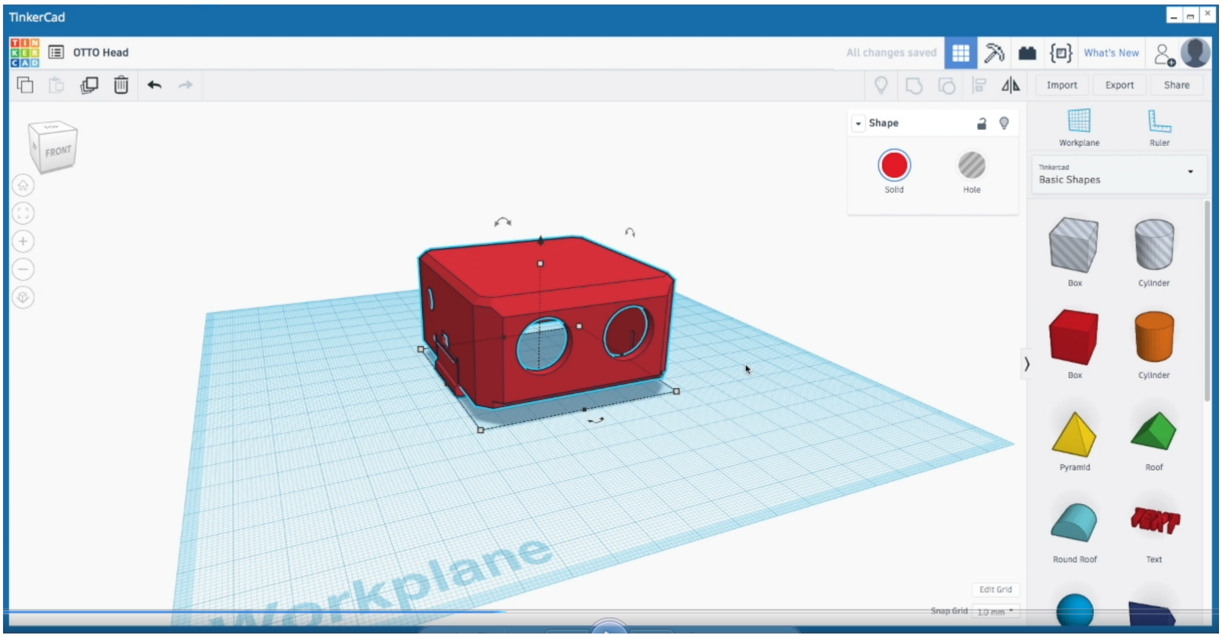

Figure 5 - 3D Modelling in TinkerCad

4.3.2. Hardware Simulation

Designing DIY electronics and simulating the designs can also be done using TinkerCad. “Circuits” is a part of the TinkerCad application (a.k.a. “TinkerCad Circuits”) that allows the learners to design DIY electronics and simulate the behaviour of their design. This application has built-in support for Arduino UNO R3 microcontrollers and has a long list of different electronical components including different type of sensors, outputs and actuators. TinkerCad Circuits can also generate a “component list” when the design is finalized.

Similar to TinkerCad 3D it is possible to share the designs with classmates and teachers, although the sharing platform is not Thingiverse and is based on sharing direct links to invitees. Those who have the access link can collaborate in the design process by making changes to the design. Group members can easily share the links to circuit designs and 3D models on Trello and assign tasks to them.

“Circuits” also includes a code editor for developing in native Arduino language. The editor includes many different libraries and a debugger. The debugger provides tools to setup break points in the code and watch the variables during simulation.

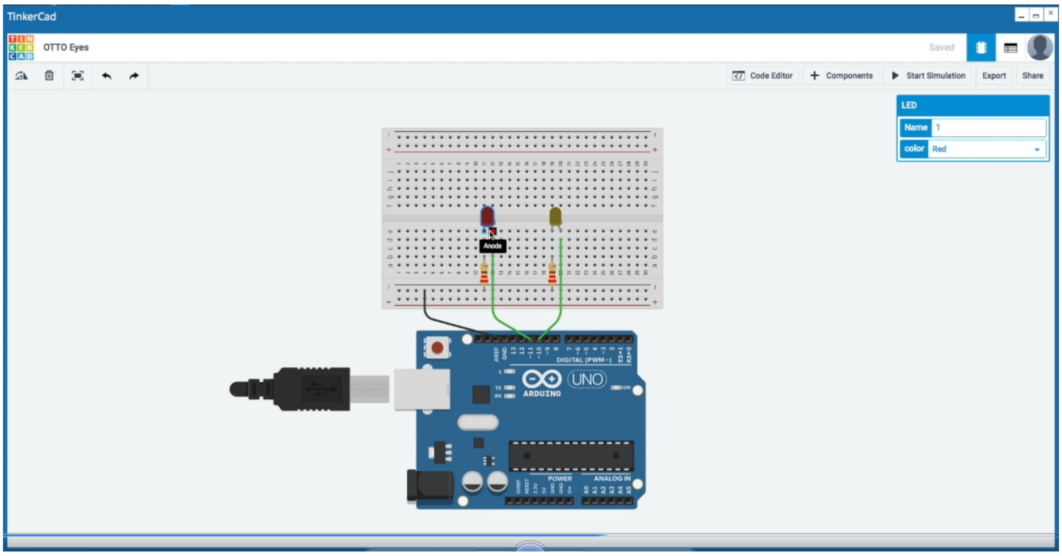

Figure 6 - DIY Electronics Simulation in TinkerCad Circuits.

4.4. Programming Stage

Deliverable D4.2 provides a set of recommended programming tools that cover a wide range of possible applications. These include:

- **Snap! And Snap4Arduino**: Snap! Is an open-source visual programming language. The Snap4Arduino flavour of it has built in support for communicating with Arduino devices (as presented in the demo video), while original flavour of Snap! Can be used to program Raspberry Pi devices and GPIO pins with ease.
- **Arduino IDE**: While originally a desktop application, recently web-based version of it is also available. The programming language is C-like and the SDK can be used to generate compiled code to run on Arduino devices.
- **Scratch and Scratch4Arduino**: Scratch can be seen as little brother of Snap!. It is a visual programming language and can be used to access the hardware (Arduino/Raspberry Pi). Unlike Snap!, Scratch cannot be easily extended without modifications in the source code.
- Other tools such as: ArduBlocks, App Inventor, Pocket Code, Nets Blox.

In D4.3 (v1) the focus has been on integrating programming tools with the following features:

- Provide visual programming.
- Can be easily extended with new components. This is especially useful when it comes to adding new functionalities such as AI Cloud services.
- Provide open source solutions to allows the development team to make required modifications i.e. for adding data analytics support to the system.
- Can easily access the hardware (Arduino/Raspberry Pi).

These properties are all found in Snap! And Snap4Arduino programming environments. Therefore, in D4.3 these tools are integrated into eCraft2Learn educational extension. This means that AI cloud services, data analytics and file services are incorporated into Snap and Snap4Arduino in this version of the educational extension and UUI.

4.4.1. AI Cloud services

Several companies offer AI based cloud services, namely: Google's machine learning, IBM Watson, Microsoft cognitive services and Amazon AI services. These services vary and normally include: text and speech analysis, image analysis, pattern recognition, etc. Although these are commercial services, they offer free services as well for limited number of queries per day which is in most cases sufficient for school projects assumed in this context.

The interfaces for these services are designed for professional programmers. Thus, in order to make these services available to learners simplified interfaces are required. Therefore, as part of the educational extension a set of programming interfaces are developed that can easily access the services from Google, IBM and Microsoft.

The interfaces expand the Snap! programming language by adding new blocks that provide the programmers with speech synthesis, speech recognition and image recognition. The work presented below can be found in more detail here (Kahn and Winters 2017).

4.4.1.1. Speech Synthesis

Snap! blocks for speech synthesis is shown below (Figure 7). These blocks are supported on popular browsers except Internet Explorer. As can be seen the programmer can have control over pitch, rate, voice, volume and even the language.

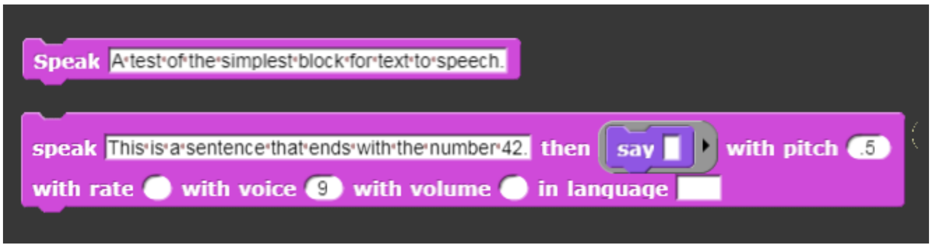

Figure 7 - Speech Synthesis Blocks for Snap!

4.4.1.2. Speech Recognition

Since the recognition services present asynchronous behaviour, the Snap! blocks designed for speech recognition rely on continuations. These blocks have a success and an error continuation. The error continuation is optional. As continuations might be a difficult topic to grasp for the learners, a special optional block is added that supports event broadcasting and a global variable. In this way, the programmer can make a call to "listen", receive broadcast when something is received and retrieve the utterance from the global variable. Figure 8 shows Snap! blocks for speech recognition.

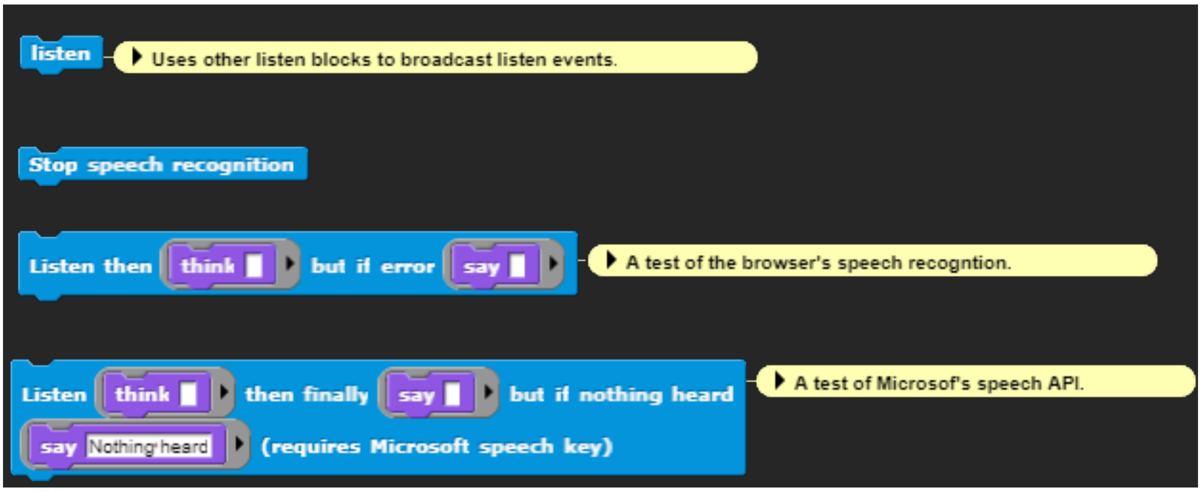

Figure 8 - Speech Recognition Blocks for Snap!

4.4.1.3. Image Recognition

Different service providers represent image recognition data in different formats. This makes design of image recognition blocks more complex. In simple form, the educational extension for AI cloud services provides the learners with simple blocks that return a textual representation of the image. More advanced blocks can return a set of properties for the image and read a specific property from it.

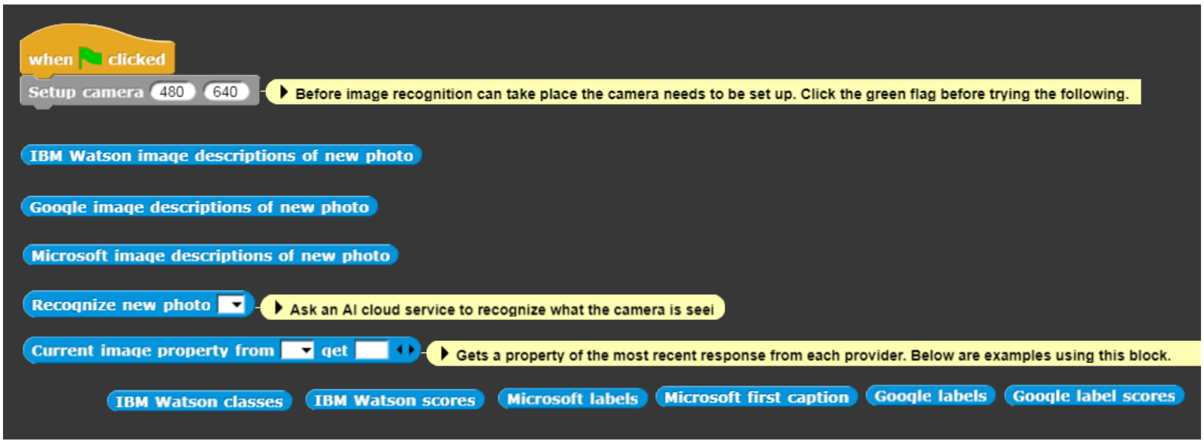

Figure 9 - Image Recognition Blocks for Snap!

4.4.1.4. Demos for AI Cloud services in Snap!

A tutorial for using the AI cloud services in Snap is available here: https://toontalk.github.io/slide-show/listen-and-speak.html

More demos can be found here: https://docs.google.com/document/d/1jnibYTmeu_Y06rhLNLI-zel4Im0O68TqH9Z0U023RXs/edit

Please note that these demos run best in Chrome browser.

4.4.2. File Management

eCraft2Learn projects can include different files. Although each tool allows for storing the project files in its own system, implementing a central place to track project files and keeping a copy of them would make the project management easier for the learners. In this deliverable, an auto save feature for Snap! and Snap4Arduino is implemented.

The educational extension for Snap! creates snapshots of the Snap! and Snap4Arduino programs while the learners are programming with these tools. The snapshot is then saved in the learners' Google Drive project folder.

4.4.3. Data Analytics and Debugging Tools

In order to provide the learning analytics and debugging tools with user data, user interaction with UUI should be observed and reported back to learning analytics system. For version 1, the educational extension records user interaction with UUI when: 1) they open a tool, 2) they close a tool, and 3) they click on help dialogs related to a specific tool. This data is time stamped and sent back for data analysis.

4.5. Sharing Stage

Sharing the learners' results with others and/or the community is an important part of eCraft2Learn learning methodology. This will not only allow the learners to share their work with classmates and friends, it will also allow them to showcase their work (with maker communities), and hence interact with this this online community to share ideas and solutions.

Sharing also allows for opening up the design/program/etc. to other tools within or outside the eCraft2Learn environment. Some of the tools which are being integrated into eCraft2Learn environment, already have support for sharing functionality. In this demo, we present the sharing possibilities between TinkerCad and Thingiverse and use this to provide the learners with eCraft2Learn Thingiverse App(s) that can be used to provide the learners with wide variety of possibilities.

4.5.1. Thingiverse

Thingiverse is a website for sharing user-created design files. It also provides free lessons for people of different ages to learn how to design 3D models for printing and how to print them. Thingiverse has close collaboration with TinkerCad and allows direct sharing of design from TinkerCad into Thingiverse by a single click. By using Thingiverse as a sharing platform for 3D designs the learners not only can share their work, but can also download interesting design from other members of the community, import them to TinkerCad and use them in their projects or improve their designs (only open source and copyleft designs are allowed to be published on Thingiverse).

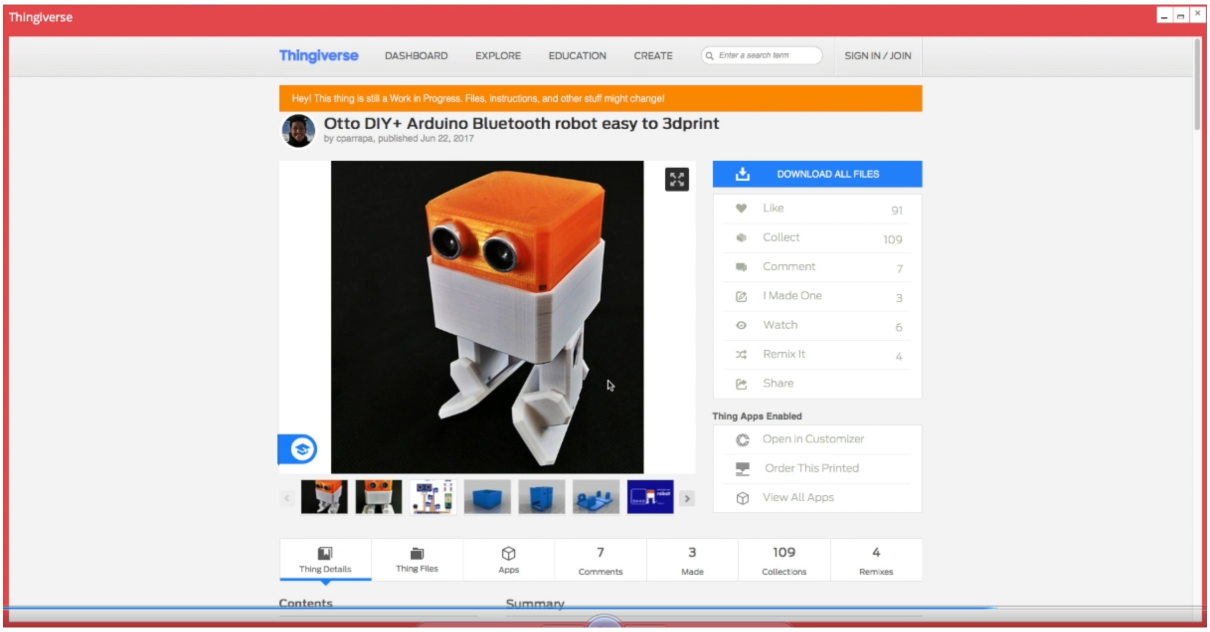

Figure 10 - Sharing on Thingiverse

4.5.2. eCraft2Learn Thingiverse App

Thingiverse also provides developers with a Web API that programmers can use to develop Thingiverse Apps. This feature is especially beneficial to eCraft2Learn because it allows us to directly communicate with Thingiverse through eCraft2Learn Thingiverse Apps.

In this deliverable and as part of the Educational Extension, we present an Augmented Reality Visualization tool for Thingiverse. This part is not directly integrated into UUI, but as seen in the demo video a user can design an object in TinkerCad, send it to Thingiverse and immediately use their mobile phone to visualize the object in a real life setting through an augmented reality view. This allows the learners to visualize the object in its final environment while tinkering with it in TinkerCad.

A demo of the augmented reality visualizer extension for TinkerCad/Thingiverse can be found in the following link: https://www.youtube.com/watch?v=y9qIy02yt7M

5 Conclusion

This document and accompanying videos demonstrated the first version of educational extension and its integration with the UUI. The online software includes tools that support the 5 stages of eCraft2Learn methodology. It also extends some of these tools to provide the learners with functionalities such as using AI cloud services in Snap! Programs, augmented reality visualization tools for 3D modelling and auto-save features for programming tools that saves program files directly on the learners' project space. This version also includes data gathering services for data analytics tools that later will be used to provide the learners and teachers with useful information during the project work and debugging.

This set of functionalities for version 1 are selected to provide the basics for the first round of pilot tests. The results of the pilot studies will provide useful insight into the next versions in the development of educational extension and UUI.

6 REFERENCES

Literature:

Kahn, K. and N. Winters (2017). Child-friendly programming interfaces to AI cloud services. EC-TEL 2017: Data Driven Approaches in Digital Education: 566-570.

Websites:

https://afsheenam.github.io/UUI

https://coggle.it

https://trello.com

https://www.tinkercad.com

https://toontalk.github.io/slide-show/listen-and-speak.html

https://www.thingiverse.com