



# D3.3: KID\_ACTIONS Digital Education Platform - Final version

**Project acronym:** KID\_ACTIONS

**Project full title:** Kick-off preventing and responding to children and AdolesCenT cyberbullying through innovative monitoring and educational technologies

**Grant info:** REC Action Grant [REC-RDAP-GBV-AG-2020] - 24 months [01.01.2021 – 31.12.2022]

**Funding:** With the financial support of the Rights, Equality and Citizenship Programme (2014-2020) European Commission – Directorate-General Justice and Consumers

**Version:** 1.0

**Status:** Final

**Dissemination level:** Public

**Due date of deliverable:** 31/12/2022

**Actual submission date:** 23/12/2022

**Work Package:** WP3

**Lead partner for this deliverable:** FBK

**Main author(s):** Alessio Palmero Arosio (FBK), Federico Bonetti (FBK)

**Contributor(s):** Sara Tonelli (FBK)

**Reviewer(s):** Alessia Torre (FBK), Anna Benedetti (DPO - FBK), Cinzia Largher (PAT), Vito Pavese, Giada Iovane (Legal Experts, AMN), Rita Marques (YEU), Gareth Cort (EUN), Marco Berti (External Ethics Adviser – Ewico), Serena Villata (External Evaluator - Université Côte d'Azur, CNRS, Inria, I3S)

*Statement of originality – This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.*

*The content represents the views of the author only and is their sole responsibility. The European Commission does not accept any responsibility for use that may be made of the information it contains.*

Funded by the Rights, Equality  
and Citizenship Programme (2014-2020)  
of the European Union



## SUMMARY OF MODIFICATIONS

Version	Date	Author(s)	Summary of main changes
0.1	01/12/2022	Alessio Palmero Aprosio (FBK), Federico Bonetti (FBK)	First draft of the deliverable shared with all partners
0.2	02/12/2022	Sara Tonelli (FBK)	First version revised
0.3	05/12/2022	Alessio Palmero Aprosio (FBK), Federico Bonetti (FBK)	Added more details on the newly introduced features
0.4	06/12/2022	Sara Tonelli (FBK)	Updated version revised
0.5	16/12/2022	Alessia Torre (FBK), Anna Benedetti (DPO - FBK), Cinzia Largher (PAT), Vito Pavese, Giada Iovane (Legal Experts, AMN), Rita Marques (YEU), Gareth Cort (EUN), Marco Berti (External Ethics Adviser – Ewico), Serena Villata (External Evaluator)	Review from partners, internal and external legal experts, external evaluator
0.6	19/12/2022	Alessio Palmero Aprosio (FBK), Federico Bonetti (FBK)	Integration of feedback and suggestions from reviewers
1.0	23/12/2022	Alessia Torre (FBK)	Final version and submission

# TABLE OF CONTENTS

Executive summary	5
1. Kid_Actions User Management (KAUM)	6
2.1. The administrator user	6
2.2. Tasks	8
2.2.1. Settings for High School Superhero	9
2.2.2. Settings for Creender	9
2.2.3. Settings for Rocket.Chat	11
2.2.5. Settings for Users	13
2.2.6. Task temporal settings	13
2.2.7. Task general settings	13
2.3. Datasets and general settings	15
2.3.1. Creender choice list	15
2.3.2. Creender datasets	15
2.3.3. High School Superhero sets of sentences	16
2.3.4. Rocket.Chat scenarios	17
3. High School Superhero	18
3.1. Features introduced after feedback	19
3.1.1. Interaction/Engagement	19
3.1.2. Admin features/task evaluation	20
4. Creender tool for image annotation	22
5. Rocket.Chat Application for Online Conversations	24
5.1. Features	24
6. The KAMoT monitoring tool	26
7. Docker container	27
7.1. Installation	27
7.2. System customisation	27
8. Download and install the software	29
9. References	30

## LIST OF FIGURES

Figure 1. The new project form in KAUM. ....	7
Figure 2. The list of educators interface in KAUM. ....	7
Figure 3. The new task form in KAUM before selecting the task type. ....	8
Figure 4. Form fields for a new High School Superhero task in KAUM.....	9
Figure 5. Form fields for a new Creender task in KAUM.....	10
Figure 6. List of picture datasets in Creender new task form.....	11
Figure 7. Form fields for a new Rocket.Chat task in KAUM. ....	12
Figure 8. Temporal settings in creating a new task. ....	13
Figure 9. Buttons available for a task before confirming it. ....	13
Figure 10. Buttons available for a task after confirming it. ....	14
Figure 11. The task information screen, with the list of students and the option buttons.....	14
Figure 12. Screen for adding a choice list in Creender. ....	15
Figure 13. Screen for adding a picture dataset in Creender. ....	16
Figure 14. Screen for managing scenarios in Rocket.Chat.....	17
Figure 15. High School Superhero gameplay: task 1 (left) and task 2 (right). ....	19
Figure 16. Feedback is given to users according to how they modify trap sentences ....	20
Figure 17. The CREENDER main interface.....	23
Figure 18. The CREENDER interface after pressing "Yes" ....	23
Figure 19. Rocket.Chat interface on mobile phones ....	24
Figure 20. Rocket.Chat example, with the description of the plot (customisable through the KAUM interface), the instructions on how to use the "/sos" command, and an example of what happens when a user call for the SOS command .....	25

## EXECUTIVE SUMMARY

This document is a brief description of the “KID\_ACTIONS Digital Education Platform final version” (Deliverable D3.3) of the European project KID\_ACTIONS, which aims at addressing cyberbullying among children and adolescents through interactive education and gamification in formal and informal settings. The deliverable type is “Other”, since it encompasses the software developed for the project activities and the related description. It also represents the final documentation of the platform, therefore it contains all the details about the installation and usage of all the tools involved. The enhancements with respect to the previous version (D3.2 - KID\_ACTIONS Digital Education Platform v.1) are highlighted, so that they can be easily identified.

A major change consists of the removal of the social media monitoring tool called KAMoT, which is released as an independent application and has been used as a source to collect textual material for the other tools. We also describe a list of minor changes in the configuration of the tools, related to the feedback received from the students and the educators during the activities performed in Autumn 2022.

Since June 2022, the **three tools targeting different educational activities are integrated into a single platform**, with a novel user management interface called **KAUM**. Through this platform, whose functionalities have been designed by asking for insights and feedback during the co-creation and piloting activities with teachers and educators, it is possible to set the different authorisations and user profiles required to perform activities with kids using **CREENDER**, **Rocket.Chat** and **High School Superhero**. The fourth tool, **KAMoT**, which had been initially developed to monitor hateful messages on Twitter, was meant to be used by educators only to leverage interesting examples of online abuse and discuss them with adolescents. However, KAMoT depends on the Twitter APIs, and it needs this kind of information (linked to a real Twitter account) to be correctly installed. Therefore, we decided not to include it in the integrated platform. It is still available on Github, and anyone is free to download and install it independently.

In this document, we report on the main changes and enhancements that have been made to the different tools before integrating them into the KID\_ACTIONS platform. The **suggestions leading to such improvements** have been collected mainly in the final **Piloting and roll-out activities (D4.4)**. For each tool we list the main changes with respect to the previous version (D3.2).

All the tools are meant to be language-independent, and the current release includes translations and documentation for English, Italian and (partly) French. An educator who wants to port the tool to another language can easily provide translations for the new language by modifying a couple of JSON files (one for KAUM and Creender, a different one for High School Superhero).

In this document, we first introduce the KAUM user management system and give instructions on how to create user accounts and tasks. Then, we describe the three tools available in the digital education platform: the High School Superhero video game, the Creender tool for image annotation, and the Rocket.Chat application for online conversations. We also mention the KAMoT tool as a standalone application. Finally, some guidelines are given on how to install and use the platform locally.

# 1. KID\_ACTIONS USER MANAGEMENT (KAUM)

The KID\_ACTIONS Digital Education platform v.1 was already introduced in D3.2. However, the current document (D3.3.) is the first public deliverable related to the platform, therefore we report again a general description of the infrastructure and single components, so that it can be used as a guideline for prospective users of the platform beyond the project duration. We highlight, anyway, the main changes with respect to D3.2 (see paragraphs marked as “New in D3.3”).

The platform includes three technological components: the chat application Rocket.Chat, the CREENDER tool for image annotation and the High School Superhero videogame. The tools are all available in the KID\_ACTIONS platform through a single interface which we called KAUM (Kid\_Actions User Management), allowing a centralised management of users and tasks.<sup>1</sup>

Given the GDPR and the outcome of the co-creation and piloting activities, we decided not to associate any account with the real identity of the students. Therefore, a list of anonymous users is created for each task, along with random passwords (see below). Then, educators can print and/or save a PDF file containing all the user/password pairs. These anonymous credentials are then given to the students and there is no way to keep track of which student got which username and password, nor digital traces of this association. Following this paradigm, we also decided not to use any social login (such as Google or Instagram).

The platform foresees three categories of users:

- **Administrator users**, who can manage groups of users and accounts (technical staff responsible for KID\_ACTIONS Educational Platform).
- **Educator users**, who can manage tasks and create lists of final users (adolescents).
- **Final users** (adolescents), who can only use the tools but they cannot access KAUM.

Besides adolescents and educators, which are considered the main participants in educational activities against cyberbullying, setting up the KID\_ACTIONS Educational Platform typically requires a person with some technical skills, able to install the platform locally and give support to educators to set up the tools. During the project, in the piloting and roll-out activities, this support was provided by FBK, but in the future it will be necessary to involve at least one person with IT skills in each school or youth centre to use the platform (note that this profile is usually available in most schools to manage the IT infrastructure).

Below we detail how administrator users can set up new projects (Section 2.1), how tasks can be defined (Section 2.2) and how datasets can be added (Section 2.3).

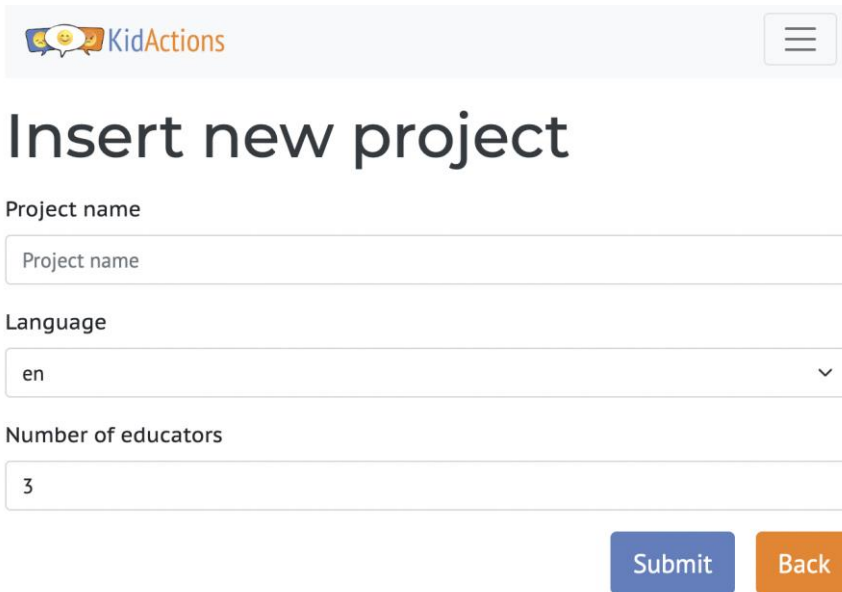
## 2.1. The administrator user

The administration user is automatically created when KAUM is installed. The default password can be set easily in the Docker installation, or by directly editing the MySQL database. The admin user can create the so-called “projects” in the KAUM interface. A project is mainly a set of activities and users (for example, a class in a school). During this creation phase, one can give the project a name

---

<sup>1</sup> <https://github.com/dhfbk/kaum>

and associate it with a language and a set of users (educators). The interface to set these parameters is displayed in Figure 1.



**Insert new project**

Project name

Project name

Language

en

Number of educators

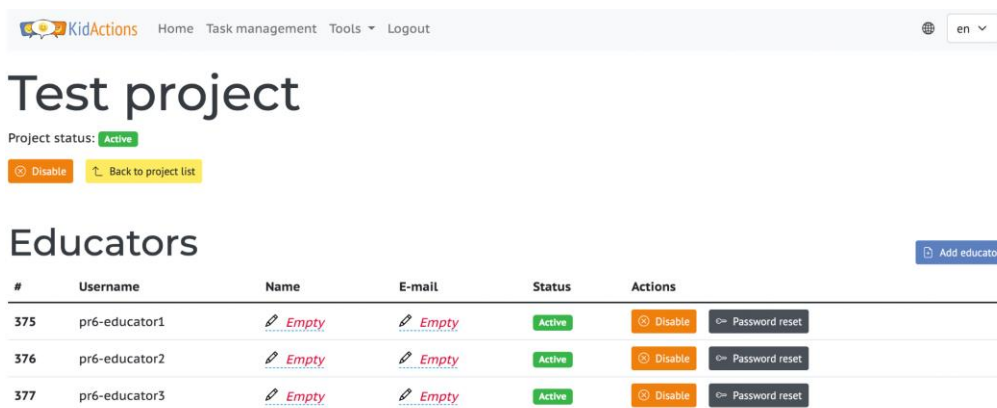
3

Submit Back

Figure 1. The new project form in KAUM.

The educators' list can be modified by the administrator throughout the use of the tool. Educators can also be disabled or associated with an existing person (with a name and an e-mail), so that when an educator uses a specific tool (such as Rocket.Chat), she/he is immediately recognized by the users (adolescents). The interface to select these parameters is displayed in Figure 2.

When a project is created, a set of passwords for educators are randomly generated, too. The administrator can see these passwords and share them with the educators. Once the project is confirmed, the passwords are encrypted and cannot be accessed any more. At any time, the administrator can reset the passwords for one or more educators.



**Test project**

Project status: Active

Disable Back to project list

**Educators** Add educator

#	Username	Name	E-mail	Status	Actions
375	pr6-educator1	Empty	Empty	Active	Disable Password reset
376	pr6-educator2	Empty	Empty	Active	Disable Password reset
377	pr6-educator3	Empty	Empty	Active	Disable Password reset

Figure 2. The list of educators interface in KAUM.

The administrator can also disable the entire project. When this action is performed, no user (educators and students) can access any of the tools. This can be done at the end of a workshop or of a training session, to ensure that users cannot continue to perform the online activities.

## 2.2. Tasks

After the project is created, educators (and, of course, the administrator) can manage the tasks linked to each project. Each of the three tools included in the KID\_ACTIONS digital platform (Rocket.Chat, Creender, and High School Superhero) can be associated with a task through the drop-down menu “Task type” (see Figure 3).

Each task has *generic* and *specific* properties. *Generic* properties include the task name, the number of users and the policy for the passwords (trivial, easy, difficult, and duplicate). We set four levels for new passwords because, if users have their own devices, a password that is easy to remember may be preferable, while in a computer lab in a school there may be shared devices and passwords that are random sequences of characters may be safer. The four levels have the following features:

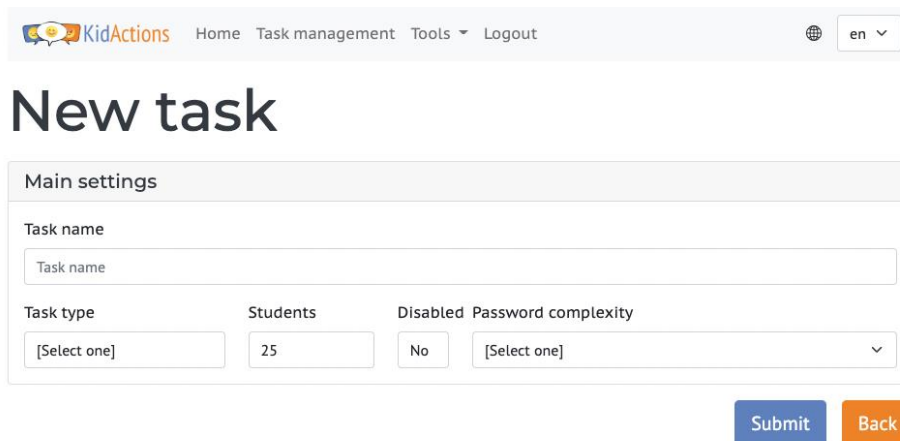
- **Trivial** means that the password equals the username.
- **Easy** is used to generate a password easy to tell and remember (a common English word, plus three digits).
- **Difficult** means that passwords are completely random sequences of characters.
- **Duplicate** means that the passwords are duplicated from a previous task (see below).

### ◆ New in D3.3 ◆

The “Disabled” option has been introduced so that one can choose whether to disable the users upon creation. The default is “No”, meaning that all the users are enabled when the task is activated.

There is an additional feature that duplicates the passwords from a previous task. Using this value, educators do not need to distribute the passwords among the users when they perform more than one task with the same group of students.

All the information filled in the task form can be modified until the task is confirmed. The second part of the form depends on the task type.



The screenshot shows the 'New task' form in the KAUM digital platform. The form is titled 'New task' and is part of the 'Task management' section. It includes a navigation bar with 'Home', 'Task management', 'Tools', and 'Logout'. The form has a 'Task name' input field, a 'Task type' dropdown menu, a 'Students' input field with the value '25', a 'Disabled' dropdown menu with the value 'No', and a 'Password complexity' dropdown menu. There are 'Submit' and 'Back' buttons at the bottom right.

Figure 3. The new task form in KAUM before selecting the task type.



### 2.2.1. Settings for High School Superhero

**High School Superhero settings**

The task information cannot be updated at a later time.

Annotations per instance:

Upload graffiti file
  Upload chat file

No datasets of this kind

No datasets of this kind

Figure 4. Form fields for a new High School Superhero task in KAUM.

High School Superhero is a video game that allows users to modify messages uttered during a conversation among non-playable characters and to delete graffiti (or part of it) written on walls (see Section 6 for details).

Educators can select which sentences should be displayed to the players during the games, so that they can easily decide whether they want to focus the activities on specific topics related to cyberbullying (for instance body shaming). This feature allows them also to include sentences in any language of choice. Educators can therefore access the High School Superhero settings to upload two text files: one is for graffiti, the other for the conversation content (see Figure 4).

#### ◆ New in D3.3 ◆

In the final version of the KID\_ACTIONS platform, the files need to be in TSV format (tab-separated values). Tabs are the following:

- The sentence to be used in the game.
- The *gold* annotation (1 = offensive; 2 = not offensive).
- The tokens involved (starting from 0, space-separated, considering space as a separator in the sentence).

Sentences with a *gold* annotation are those that have already been labelled as offensive or not. They can be used in the games to check whether participants are performing the task correctly. The second and third columns are not mandatory; therefore, the file can include only one column, that is a TXT file with one sentence per line (this guarantees backward compatibility).

In the demo mode, a few sentences can be pre-uploaded (see Section 4).

Finally, one can select how many annotations (from different users) should be collected for each sentence both in graffiti and chats. This setting is useful to collect duplicate annotations for the same portion of text, so that educators can compare how different students judge the same sentence.

### 2.2.2. Settings for Creender

Creender is an application developed to add comments to images, in an Instagram-like environment (for details see Section 5). It was introduced because most of the teenagers involved in the piloting and co-creation activities told us that they have an Instagram account to communicate with their friends, and that they are familiar with image-based social media. A lot of options are available in Creender to tune the task, as displayed in Figure 5.

### Creender settings

The task information cannot be updated at a later time.

<b>Question text:</b> <div style="border: 1px solid #ccc; padding: 5px; min-height: 40px;">         If you saw this picture on Instagram, would you make fun of the user who posted it?       </div>	<b>Choice question:</b> <div style="border: 1px solid #ccc; padding: 5px; min-height: 40px;">         Why would you make fun of this user?       </div>
<b>Comment question:</b> <div style="border: 1px solid #ccc; padding: 5px; min-height: 40px;">         What would you write?       </div>	<b>Annotations per instance:</b> <div style="border: 1px solid #ccc; padding: 5px; width: 60px; text-align: center;">         2       </div>

<input type="checkbox"/> Do not ask for a comment <input type="checkbox"/> The comment is mandatory <input type="checkbox"/> Show choices when user clicks 'No'	<input type="checkbox"/> Buttons are immediately available to users <input type="checkbox"/> One click is enough for the user to confirm <input type="checkbox"/> Allow multiple choices <input type="checkbox"/> Enable demo mode
---	---

---

#### Choices

Clone from:

Add choice (↵ to add):

Save dataset

The list is empty

Figure 5. Form fields for a new Creender task in KAUM.

First of all, all texts in the Creender interface can be customized by setting the *Question text* (i.e. which question should be displayed to users together with the image), the *Choice question* (multiple choice to explain the answer given to the question text), the *Comment question* (i.e. what the user would post as a comment) and the remaining options, displayed in Figure 5:

- **“Do not ask for a comment”** hides the text area where users can add a textual comment to a photo.
- **“The comment is mandatory”** means that users cannot confirm the form until they add a comment.
- **“Show choices when a user clicks ‘No’”** simply switches the behaviour of the “Yes” and “No” buttons, allowing educators to decide when it is more appropriate to display the choice window.
- **“Buttons are immediately available to users”** removes the delay of two seconds (default value) that users have to wait to confirm their choice.
- **“One click is enough for the user to confirm”** disables the second click needed to confirm the choice. When the box is checked, users will submit their choice with just one click (instead of two) without the possibility to rethink it.
- **“Allow multiple choices”** means that users can select more than one choice to describe the picture.

- “Enable demo mode” can be used by educators to create demo login credentials, useful to show how Creender works to the users prior to starting the proper activities.

Choose photos

#	Name	Photos	Amount to use
3	Religion	80	<input type="text" value="0"/> <input type="button" value="x"/> <input type="button" value="✓"/>
4	LGBT+	84	<input type="text" value="0"/> <input type="button" value="x"/> <input type="button" value="✓"/>
5	Miscellaneous photos	3281	<input type="text" value="0"/> <input type="button" value="x"/> <input type="button" value="✓"/>
6	Sport	103	<input type="text" value="0"/> <input type="button" value="x"/> <input type="button" value="✓"/>
7	Body positivity	230	<input type="text" value="0"/> <input type="button" value="x"/> <input type="button" value="✓"/>
8	Disability	25	<input type="text" value="0"/> <input type="button" value="x"/> <input type="button" value="✓"/>

Save dataset

Figure 6. List of picture datasets in Creender new task form.

Educators can then define the repositories where photos can be taken and uploaded to the tool. Datasets can be added using the “Choose file” option at the bottom of the form (see the interface in Figure 6). The uploaded file must be in ZIP format and must contain photos in JPEG or PNG format. Datasets can also be added by the administrator.

### 2.2.3. Settings for Rocket.Chat

Rocket.Chat is a chat application used to simulate conversations where a bully targets a victim while other participants (bystanders) intervene to support one of the two (see Section 6 for details). For the Rocket.Chat tasks, only few settings are required, displayed in Figure 7:

- The initial description of the activity, which we call *Scenario*, is the text displayed to the users before starting the simulation. It describes the plot of a fictional situation, in which the students are assigned roles to play while simulating cyberbullying attacks.
- The channel name is useful for the educators to distinguish between multiple chats when they login to Rocket.Chat interface.
- The checkbox “Educators can always enter the chat” can be used to bypass the SOS mechanism and enable educators to join the chat when they like. Otherwise, they can access the chat only when a student asks for help.

Please fill in this field. at settings

Some task information cannot be updated at a later time.

Groups  Unique scenario

Initial description of the activity:

Educators can always enter the chat  
*If this checkbox is not selected, educators can enter the chat only when one user calls /sos during the session.*

Channel name (letters, numbers and dashes):

---

Groups

Automatic generation of the groups  excluding the last  users.

Select group for each user. Use 0 (zero) if the user does not participate to any group.

User 1: <input type="text" value="0"/>	User 2: <input type="text" value="0"/>	User 3: <input type="text" value="0"/>	User 4: <input type="text" value="0"/>
User 5: <input type="text" value="0"/>	User 6: <input type="text" value="0"/>	User 7: <input type="text" value="0"/>	User 8: <input type="text" value="0"/>
User 9: <input type="text" value="0"/>	User 10: <input type="text" value="0"/>	User 11: <input type="text" value="0"/>	User 12: <input type="text" value="0"/>
User 13: <input type="text" value="0"/>	User 14: <input type="text" value="0"/>	User 15: <input type="text" value="0"/>	User 16: <input type="text" value="0"/>
User 17: <input type="text" value="0"/>	User 18: <input type="text" value="0"/>	User 19: <input type="text" value="0"/>	User 20: <input type="text" value="0"/>
User 21: <input type="text" value="0"/>	User 22: <input type="text" value="0"/>	User 23: <input type="text" value="0"/>	User 24: <input type="text" value="0"/>
User 25: <input type="text" value="0"/>			

Figure 7. Form fields for a new Rocket.Chat task in KAUM.

◆ New in D3.3 ◆

In the new version of the platform, a single set of users can be split into groups that will interact in different chat rooms. To enable this option, just use the “groups” field and select the number of groups one wants to split the users into.

Using the “unique scenario” field, educators can also customize the scenarios and define different configurations for each group.

Finally, in the “groups” section, users can be manually (or programmatically) assigned to a particular group. For each user, a text field can be filled with the corresponding group ID, from 1 to the number of groups written in the “groups” field above. Zero (0) means that the selected user would not participate in the task.

Above the user fields, an option can be used to automatically split users into groups, with alternation (meaning a situation similar to 1, 2, 1, 2, ...) or without alternation (meaning 1, 1, 1, 2, 2, 2, ...). One can decide that the last N users are not included in the splits (setting their value to zero).

## 2.2.5. Settings for Users

◆ New in D3.3 ◆

In addition to the three types of tasks, the educator (or the administrator) can create a dummy task of type “users”. Once the passwords are generated, this task can be used when creating a new one just using the same passwords, so that educators do not need to share different login information for each task (see the beginning of Section 2.2 for more information).

## 2.2.6. Task temporal settings

Each task can be activated manually (see above) or can be made available to users in certain timeframes (see Figure 8). This feature was introduced to limit the duration of the different activities if needed. For example, if the platform is used in schools, educators may want to stop its use when classes are over. On the contrary, if the platform is used in youth centres, the activities may be moved after school hours. In general, this functionality is provided to educators because the platform is meant to be used under their supervision, in the context of educational activities, and not by teenagers alone.

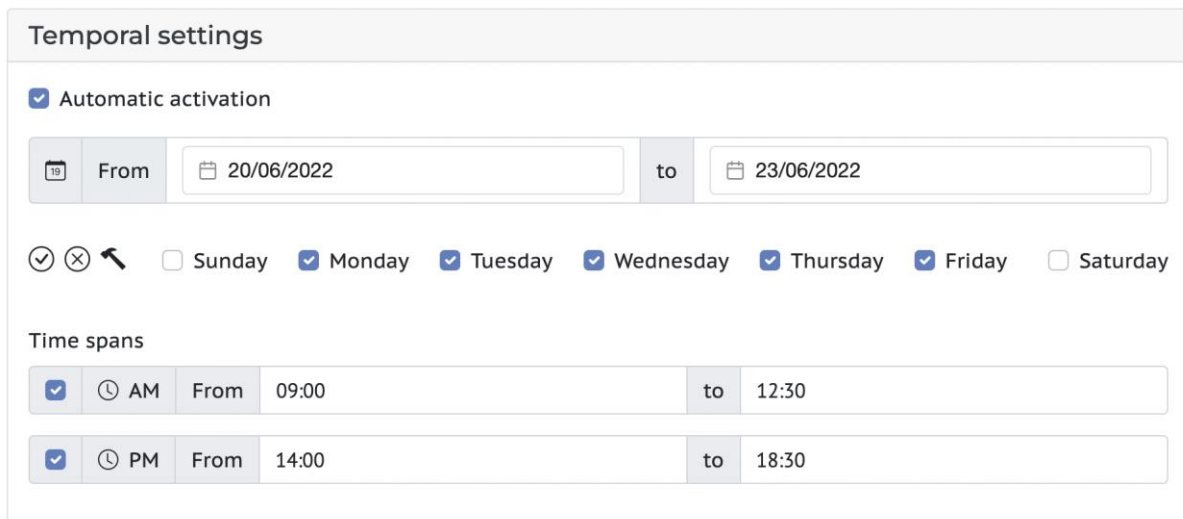


Figure 8. Temporal settings in creating a new task.

By selecting the automatic activation, educators can define the date interval and the weekdays in which the task is available. In addition, time spans can be used to limit the tool availability.

## 2.2.7. Task general settings

After the task is created, it needs to be confirmed before users can join it. After this confirmation, the task cannot be modified any more.

# Tasks

[Add task](#)

#	Tool	Name	Students	Status	Actions
19	creender	Test task	25	Unconfirmed	<a href="#">Confirm</a> <a href="#">Edit</a> <a href="#">Clone</a> <a href="#">Delete</a>

Figure 9. Buttons available for a task before confirming it.

- The “Clone” button can be clicked to create a new task using the current one as template (one then can edit everything to match the needs).
- The “Delete” button permanently deletes the task.

## Tasks



#	Tool	Name	Students	Status	Actions
19	creender	Test task	25	Disabled	Manage Clone Enable Users passwords Close Delete

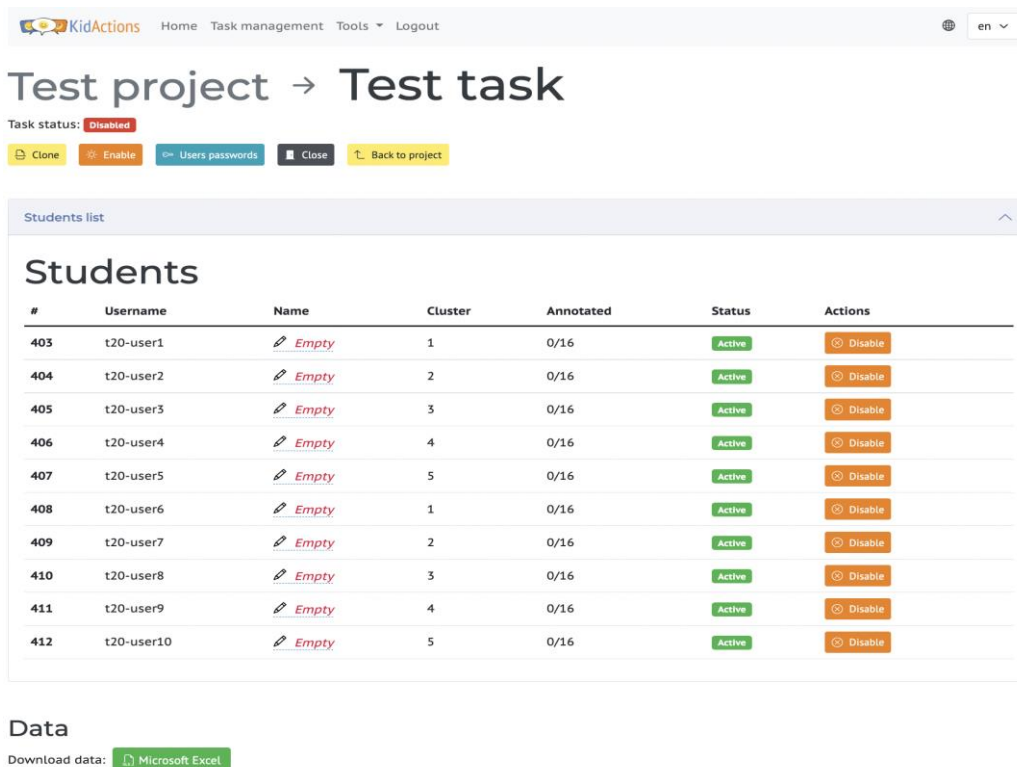
Figure 10. Buttons available for a task after confirming it.

After the confirmation, some new actions become available (see Figure 10).

- “Enable”/“Disable” buttons can be used to manually manage the availability of the task.
- “Users passwords” shows a PDF file with all the passwords (this PDF can be printed and cut up to provide a slip of paper for the users).
- “Close” disables the task so that no one can enable it any more.
- “Manage” opens the windows with the task information.

### ◆ New in D3.3 ◆

In the new version of the platform, the PDF file with the passwords can have multiple formats, so that personalized sheets of paper, previously cut, can be used.



Task status: Disabled

Clone Enable Users passwords Close Back to project

#	Username	Name	Cluster	Annotated	Status	Actions
403	t20-user1	Empty	1	0/16	Active	Disable
404	t20-user2	Empty	2	0/16	Active	Disable
405	t20-user3	Empty	3	0/16	Active	Disable
406	t20-user4	Empty	4	0/16	Active	Disable
407	t20-user5	Empty	5	0/16	Active	Disable
408	t20-user6	Empty	1	0/16	Active	Disable
409	t20-user7	Empty	2	0/16	Active	Disable
410	t20-user8	Empty	3	0/16	Active	Disable
411	t20-user9	Empty	4	0/16	Active	Disable
412	t20-user10	Empty	5	0/16	Active	Disable

Data

Download data: Microsoft Excel

Figure 11. The task information screen, with the list of students and the option buttons.

In the “Manage” screen (see Figure 11), a fictional name can be assigned to each user (useful, for example, in Rocket.Chat, since the simulation takes place after assigning to participants different roles and names). Some additional columns show the summary of the task. For instance, in Creender and High School Superhero two columns with the group of images/sentences used for the activities and

the number of annotations done are present. Users can also be disabled, in case specific participants decide to leave the activities.

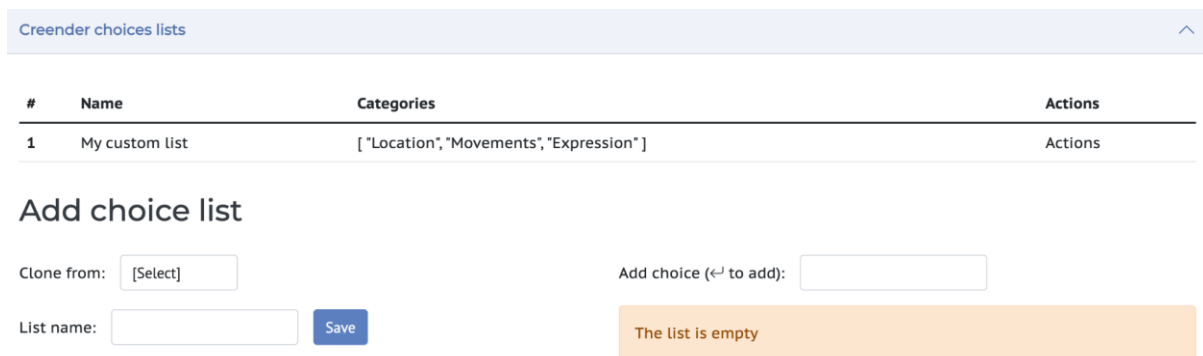
At the bottom of the page, educators can download the data collected during the tasks in Excel format by clicking on the “Download data” button. This feature allows educators to further analyse the data that were collected during the activities, use them to initiate discussions in classes, organise awareness-raising activities and support debriefing. The data collected during the roll-out activities are part of the KID\_ACTIONS database, which we describe in more detail in D2.5 “Multi-dimensional methodology final version” and are available at <https://github.com/dhfbk/ka-dataset>.

## 2.3. Datasets and general settings

Since some data can be used across multiple projects and/or tasks, the interface also provides a menu, “Task management”, where reusable information can be stored.

### 2.3.1. Creender choice list

Through this screen, one can manage the lists of choices that can be used within the Creender tool.



#	Name	Categories	Actions
1	My custom list	[ "Location", "Movements", "Expression" ]	Actions

**Add choice list**

Clone from:  Add choice (↵ to add):

List name:   The list is empty

Figure 12. Screen for adding a choice list in Creender.

Users can add labels using the input box on the right (see Figure 12). When all labels have been added, the list can be saved by giving a name to it and pressing the “Save” button on the right. One can start from an existing list, by selecting it from the menu on the right. Then the choices can be added using the input on the right and the list saved as already described.

### 2.3.2. Creender datasets

Similarly, to the previously mentioned choice lists, also repositories of pictures can be added through this interface and made available to all users.

Creender datasets ^

Add dataset:

#	Name	Records	Demo	Actions
3	Religion	80	<input type="checkbox"/>	<input type="button" value="Delete"/>
4	LGBT+	84	<input type="checkbox"/>	<input type="button" value="Delete"/>
5	Miscellaneous photos	3281	<input checked="" type="checkbox"/>	<input type="button" value="Delete"/>
6	Sport	103	<input type="checkbox"/>	<input type="button" value="Delete"/>
7	Body positivity	230	<input type="checkbox"/>	<input type="button" value="Delete"/>
8	Disability	25	<input type="checkbox"/>	<input type="button" value="Delete"/>

Figure 13. Screen for adding a picture dataset in Creender.

The interface (Figure 13) is very similar to the one already described when adding a Creender task. The only difference is the demo switch: when it is turned on, the photos used in the demonstration interface of Creender are taken from these datasets (instead of showing a placeholder).

◆ New in D3.3 ◆

The Platform is shipped with pre-installed sets of photos:

- Religion (80 photos)
- LGBT+ (84 photos)
- Sport (103 photos)
- Body positivity (230 photos)
- Disability (25 photos)
- Miscellaneous pictures (from the COCO dataset,<sup>2</sup> 3281 photos)

### 2.3.3. High School Superhero sets of sentences

◆ New in D3.3 ◆

The Platform is shipped with pre-installed sets of sentences for High School Superhero. In the following table, the number of sentences for each set and language is listed.

	Italian		English		French	
	Dialogues	Graffiti	Dialogues	Graffiti	Dialogues	Graffiti
Religion	205	70	110	40	351	70
Body	500	70	150	40	891	70
Sexism	409	70	150	41	527	70
Racism	347	70	150	39	1.096	70
Threat	458	70	150	40	-	-

<sup>2</sup> <https://cocodataset.org/>



## 2.3.4. Rocket.Chat scenarios

### ◆ New in D3.3 ◆

Rocket.Chat scenarios can be added through this interface. One can insert the information already described in Section 2.2.3, and some other ones useful to categorize the scenario. In particular, the name, the language, and the target school can be specified.

The Platform is shipped with pre-installed scenarios that can be used in Rocket.Chat activities. The pre-loaded descriptions are available in English, Italian and French and are divided into middle school (3 scenarios) and high school (2 scenarios), depending on the topic covered.

Rocket.Chat scenarios ^

#	Name	Channel name	School	Lang	Actions
1	Danza classica	danza	Middle school	it	<span style="color: red;">🗑 Delete</span>
2	Ginnastica	ginnastica	Middle school	it	<span style="color: red;">🗑 Delete</span>
3	Staffetta	staffetta	Middle school	it	<span style="color: red;">🗑 Delete</span>
4	Foto intime	foto	High school	it	<span style="color: red;">🗑 Delete</span>
5	Bacio	bacio	High school	it	<span style="color: red;">🗑 Delete</span>

### Add scenario

Title:  Language:  School:

Initial description of the activity:

Initial description of the activity

Educators can always enter the chat  
*If this checkbox is not selected, educators can enter the chat only when one user calls /sos during the session.*

Channel name (letters, numbers and dashes):

Figure 14. Screen for managing scenarios in Rocket.Chat.

### 3. HIGH SCHOOL SUPERHERO

High School Superhero (Bonetti and Tonelli, 2020) is a role-playing video game, integrated into the KID\_ACTIONS Educational Platform, displaying sentences to players that have to be judged according to their offensiveness. The activities are divided into two main tasks: in **Task 1**, which focuses on dialogues between fictitious characters, players have to change the sentences that they read if they are offensive, so as to make them not offensive. The changes are then saved in a .xlsx file and can be analysed to understand how players transformed the sentences, if they did. Changing a sentence consumes one battery bar, which has to be replenished by exchanging collectables from specific terminals. Previously, changing one word consumed one battery bar but this was almost unanimously deemed as too restrictive by students during the piloting and roll-out sessions.

**Task 2** is similar to Task 1, except messages are displayed on walls and floors in the form of graffiti, and players have to erase the words they deemed abusive or offensive (see Figure 15, right). Similarly to the dialogue mechanics, erasing the graffiti has a cost in terms of consumable soap, which can be bought in exchange for crystals like the batteries used in Task 1. In this way, we encourage players to spare the resources and thus delete only the offensive part of a message (e.g. an adjective, a slur), while keeping the rest unchanged. This in turn enables the identification of the exact offensive span for post-hoc debriefing sessions and linguistic analyses. It is worth mentioning that the mechanic of consuming resources can be turned off as part of the changes made after the co-creation activities. The crystals used to buy the resources employed in the two tasks are scattered around the environment, which means that exploration is deeply interconnected with the tasks. To enhance fun and enjoyment while exploring, two devices were introduced: it is possible to navigate the environment with the Glider and the Rocket Shoes, which can be bought with crystals.

An avatar customization screen is also present at the beginning of the game, where players can customise their characters as they prefer. This was introduced to increase the players' identification with their character and higher engagement in the virtual world. Gender is not selected explicitly.

The game was developed with Unity3D<sup>3</sup>, which uses C# as its programming language, and Blender<sup>4</sup>, a free 3D modelling tool. The style is simple and cartoonish, and was inspired by successful commercial games such as Animal Crossing: New Horizons, Dragon Quest Builders, and The Legend of Zelda. The following creators have provided free assets that are currently used in the game: Quaternius<sup>5</sup>, Jason Shaw<sup>6</sup>.

It is possible to play the game via a web browser on desktop or laptop computers. The default playable version that we release together with the platform is the WebGL one, which works via the browser. The setup is intended to be carried out by the school system administrator or IT staff in youth centres. The content of the sentences should be decided by formal and non-formal educators, who work in close contact with the final users and know which material would be best suited for them. The sentences can be loaded from the KAUM API or kept locally and put in a game folder in two .txt files, with one sentence per line, one for the dialogues and one for the graffiti. In the latest version of the

---

<sup>3</sup> <https://unity.com/>

<sup>4</sup> <https://www.blender.org/>

<sup>5</sup> <https://quaternius.com/>

<sup>6</sup> <https://audionautix.com/>

game a feature is also present for including gold-standard information in the sentence files, which is described in detail in the next section.

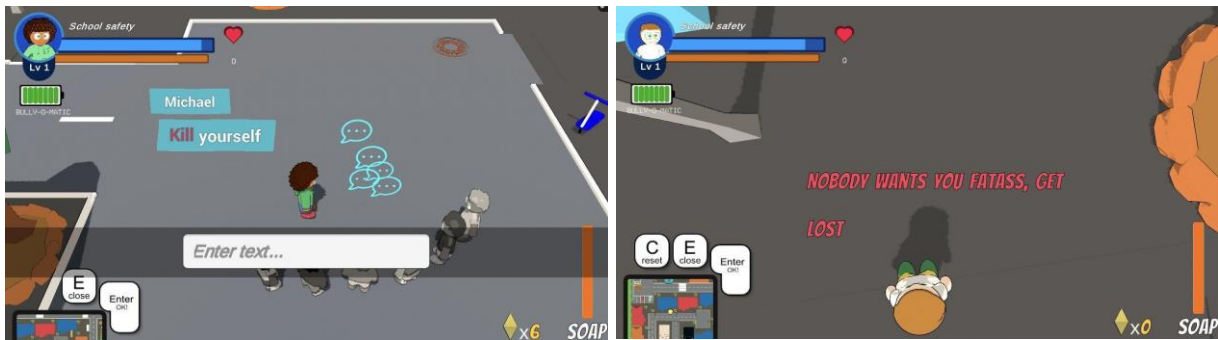


Figure 15. High School Superhero gameplay: task 1 (left) and task 2 (right).

### 3.1. Features introduced after feedback

Some formal educators were concerned about the graphic style of the video game, which was perceived as too children-oriented. For this reason, they expressed the feeling that the best target would be very young students such as middle-schoolers (between 11 and 13 years old). This comment was shared also by a number of students. However, using more adult, realistic graphics would have meant a complete redesign. Some students also asked for a first-person view, where players see the virtual world with the eyes of their character. This request was put forward, especially by older students. This may be because of their gaming habits: older students may prefer more realistic first-person games. This feature was however not added, as it would have implied rather a heavy redesign of some core features and several technical optimization problems. The top-down camera allows to keep the number of objects on the screen quite low and this is particularly important given the fact that the game should run on as many platforms as possible.

What follows is a summary of the features that have been implemented during and after the co-creation and piloting and roll-out activities with educators and adolescents.

#### 3.1.1. Interaction/Engagement

Many students asked for improved avatar customization, which was addressed with 4 new hairstyles and 11 new pieces of clothing. In addition, some requests coming from both the trainers and the students regarded more complex and varied interactions with the characters. Some suggestions they expressed regarding the possibility to add relational rewards for players, such as new followers or friends, in order to make it visible that changing the language makes a difference in the virtual world. These were addressed by introducing a *like* system that keeps track of the good deeds performed by the user, such as side activities like saving victims from lockers, and a narrative introduction. Students that are saved become part of the player's *friend count*, another feature that was added. Regarding this more empathic and human aspect of the game, non-playable characters that have not yet been saved appear in black and white, and become coloured once the bullying episode is dealt with.

Student feedback regarding the scoring system suggested that rewards should be increased when interacting with text. This is addressed as well. Some students, especially the younger ones, had some difficulties while moving and overcoming environmental obstacles, or remembering the correct keys

to press. A teleport button was added to the menu to reset the player's position in case there were problems navigating the environment; the control system was rewritten to accommodate mouse input for every action. Some minor tweaks were also introduced such as a smaller sponge to erase the graffiti, new sound effects, and minor bug fixes regarding the interaction with the menus, such as bigger input fields.

◆ New in D3.3 ◆

- The point system is now more forgiving: game over is harder to reach and the points given for the annotations are more generous. This was changed after some students pointed out the problem of frequent game overs during the piloting and roll-out activities.
- It is now easier to level up and buy new equipment. This change was made after observing that some participants had difficulties levelling up beyond level 2.
- Interaction with dialogues has been made easier. It is now possible to interact with any virtual conversation participant instead of pressing 'E' on a specific spot.
- Various sound effects were added. In particular, a sound effect for the rocket boots and one for the glider.
- Sentences now require fewer battery notches to be annotated since during the piloting activities some students reported this specific mechanic to be too strict.
- Minor bug fixes regarding mainly: the interaction with dialogues and graffiti (some windows remained opened while annotating); with the scooter (sometimes the player disappeared); overlapping interface elements; visibility of interface buttons (some users were confused as to how to confirm the appearance of the customised character).



Figure 16. Feedback is given to users according to how they modify trap sentences

### 3.1.2. Admin features/task evaluation

Regarding the additions to the administrative side of the game, the following features were introduced after the co-creation sessions: a login system that establishes a connection between the game and the KAUM interface; a config.txt file for educators to change some simple features, such as resource depletion (this can be turned on or off); a demo mode to showcase the game mechanics and some sample sentences, useful to show the game quickly while explaining how it works.

◆ New in D3.3 ◆

- A gold-standard system has been implemented that allows educators to upload sets of sentences with additional information on how they should be annotated by users. This information regards: 1) whether the sentence is offensive or not, and 2) which tokens constitute the offence. These

gold-standard sentences can also be referred to as ‘trap sentences’ and are intended to be inserted with random frequency. This allows educators to check that users are effectively identifying offences and carrying out the task according to the instructions. In fact, every time a trap sentence is interacted with, the game praises or warns the user according to the gold-standard judgement provided by educators. The feedback changes according to a score calculated based on whether the sentence is correctly identified as offensive on how many tokens were correctly identified. Warnings and praises are the following: “Not quite...”, “Almost...”, “Good job!”/“Well done!” and “Perfect!” (see Figure 16).

- Now the game and KAUM keep track of how much time it took for a user to annotate a certain sentence. This allows educators to understand which sentences were skipped or considered difficult to judge.
- It is now possible not only to select the language among a fixed set but also to add new ones. The json file named config.txt can take in a list of languages ([en, fr, it...]) where the new one can be added. Three files must then be provided, by copying and translating the existing ones: systemMessages\_en.txt, dialogues\_en.txt and sampleSentences\_en.txt. The first and second files contain information about immutable strings that are shown as part of menus, conversations with non-playable characters and so on; the last one contains sample sentences that appear when the DEMO mode is selected.
- Users are now automatically kicked out of the game when educators decide to end the activity (by disabling it in the KAUM interface).

High School Superhero is developed using Unity<sup>7</sup> and Blender<sup>8</sup>. In order for the project to be correctly opened in Unity, Blender must be installed. For best compatibility, the suggested Unity version is 2021.3.3f1. The source code is available on GitHub.<sup>9</sup>

---

<sup>7</sup> <https://unity.com/>

<sup>8</sup> <https://www.blender.org/>

<sup>9</sup> <https://github.com/dhfbk/HighSchoolSuperhero>

## 4. CREENDER TOOL FOR IMAGE ANNOTATION

CREENDER (Palmero Aprosio et al., 2021) is an annotation tool developed by FBK to create multimodal datasets of images and comments. With this tool, it is possible to simulate a scenario where different users access the platform and are displayed different pictures. Then, they can leave an Instagram-like comment and associate a semantic tag to each image. The same pictures can be shown to different users, allowing a comparison of their comments and online behaviour.

The CREENDER tool can be accessed both via browser and mobile phone. The former option is preferable for formal educational activities carried out in classes, whereas educators in non-formal settings may wish to run activities via mobile devices. The interface is multi-language: English and Italian are already included, while other language files can be added as needed.

For each annotation session, on the configuration side, a set of images (or a set of external links to images on the web) needs to be given to the tool. Then, one can set the number of users and the number of annotations that are required for each photo. Finally, the system assigns the photos to the users and creates the login information for them. The access to the system is password-protected, meaning that the administrator will provide each participant with a login and password (not connected with any personal information, the administrator will not keep track of who received which username). There is also a demo mode that does not include photos that are part of the actual tasks, so that when an educator shows the interface to the audience they do not create a bias in the possible answers. Users can report a picture as offensive or inappropriate, so that adolescents can avoid being exposed to content that they consider harmful. If an image is flagged as inappropriate by a participant, the educator responsible for the task is notified and they can decide to remove that image for future sessions.

Given a picture, the system can be set to perform three actions in sequence or isolation, as needed by the task: i) the picture can be skipped by the user so that no annotation is stored and the next one is displayed; ii) the user can insert free text associated to the image. This can be used to write a caption, comment on the picture, list the contained objects, etc. iii) one or more predefined categories can be assigned to the picture. As part of some improvements introduced after the co-creation activities, the delay between the appearance of the photo and the activation of the buttons can be chosen through the KAUM interface. Categories can range from specific ones related to the portrayed objects (e.g. male, female, animals, etc.) to more abstract ones, like for example the emotions provoked by looking at the picture. The interface is language-independent, since all the main blocks of text can be defined when the task is created within KAUM.

In the configuration screen (see Section 2.2.3), the administrator can edit the prompted questions and the possible answers, so that the tool can be used for a variety of different tasks. If needed, instructions can be provided to users before starting the annotation task, for example to establish a scenario. Using the administration web interface, it is also possible to monitor the task with information about the number of annotations that each user has performed. This enables us to check whether some users experience difficulties in the annotation, or if some annotators are anomalously fast (for example by skipping too many images). Once the annotation session is closed, the administrator can download the resulting corpus containing the images and the associated information. The export is available in Excel format (xlsx).



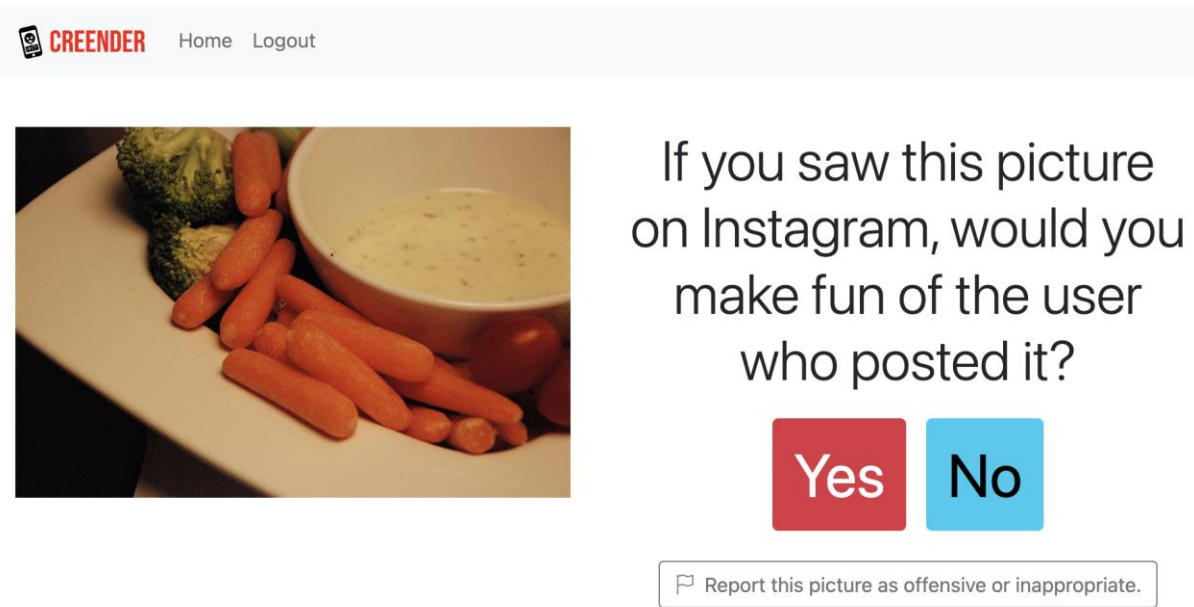


Figure 17. The CREENDER main interface

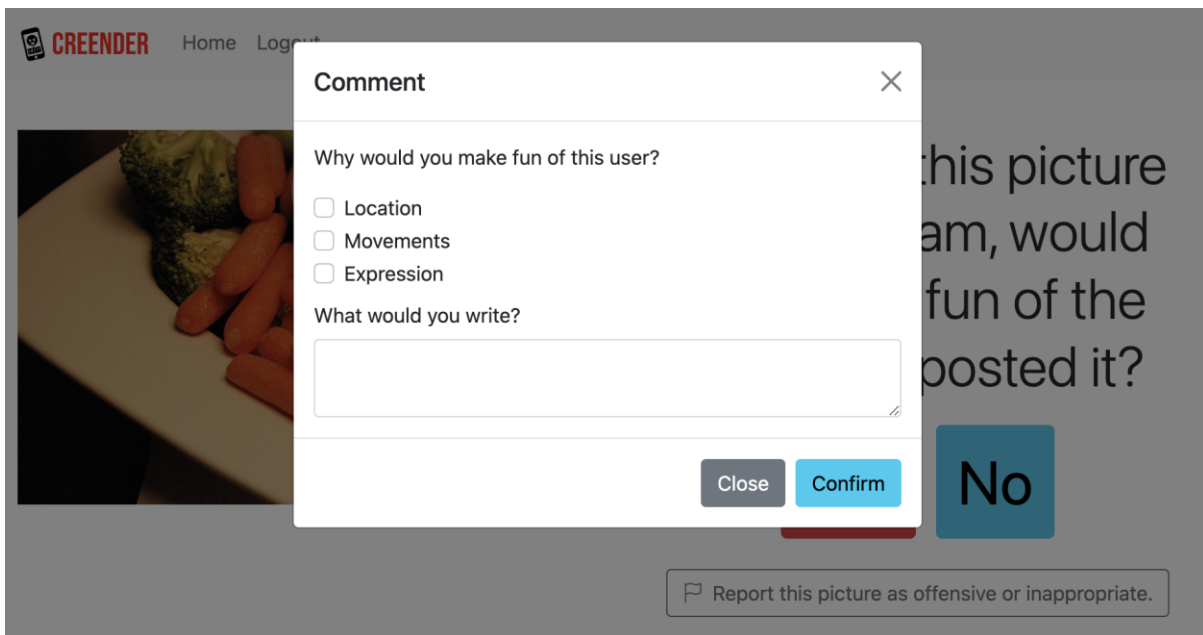


Figure 18. The CREENDER interface after pressing "Yes"

The software is distributed by FBK as an open-source package<sup>10</sup> and is released under the Apache license (version 2.0). The API (backend) is written in PHP and relies on a MySQL database. The web interface (frontend) is developed using the HTML/CSS/JS paradigm using the modern Bootstrap and Vue.js frameworks. The interface is responsive, so that one can use it from any device that can open web pages (desktop computers, smartphones, tablets).

<sup>10</sup> <https://github.com/dhfbk/creender3>

## 5. ROCKET.CHAT APPLICATION FOR ONLINE CONVERSATIONS

Rocket.Chat<sup>11</sup> is an open-source, fully customisable communication platform for organisations with high standards of data protection. It is free, it can be customised in any aspect, and it is available on computers, tablets and mobile phones through the web interface or the app. With this platform, it is possible to study the characteristics of cyberbullying conversations through role-playing, without the issues related to privacy policies of applications such as WhatsApp. It has a user-friendly interface and it is possible to customise the topic of the chat, which allows educators to target specific groups of adolescents.

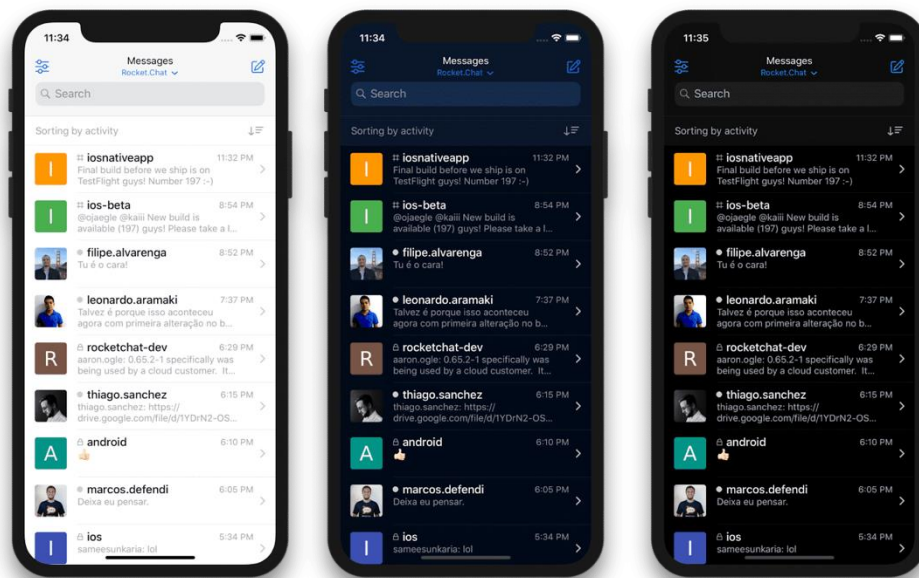


Figure 19. Rocket.Chat interface on mobile phones

### 5.1. Features

Here are summarised some of the most important features of the version of Rocket.Chat that is integrated into the platform.

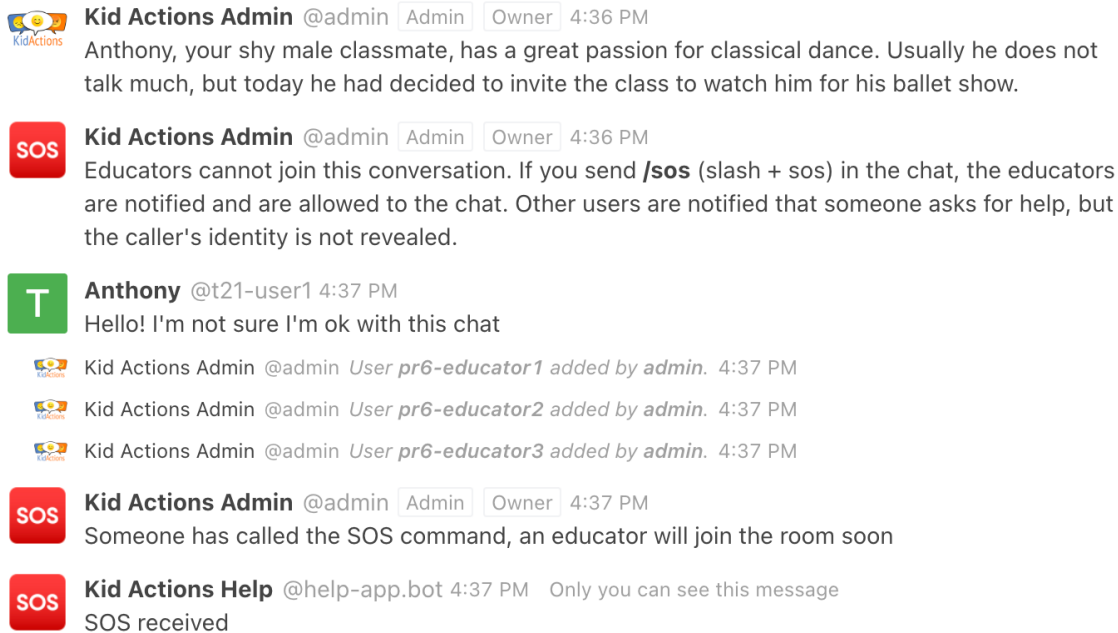
- **Multi-language:** the Rocket.Chat interface is available out-of-the-box in more than 20 languages (including English, Italian, and French). It is also possible to add more languages by translating the strings using the internationalisation capabilities of the program. Messages can contain emojis, as well as other types of media (images, links, etc.)
- **User management:** the KAUM interface is used to create the users that can access the chats (which is usually done by the educators and the administrator).
- **SOS command:** Students may feel uneasy when chatting if they are aware of being observed by adults. We introduced an “SOS” option that can be enabled from the admin interface,

<sup>11</sup> <https://rocket.chat/>



depending on whether chat participants should control when an educator can enter the chat or not.

When the SOS option is disabled, educators can always enter the chat. Instead, when it is enabled, they are admitted to the conversation only if a participant calls the SOS. Indeed, users can trigger the SOS command by writing “/sos” to notify educators that they need help. Then, the educator is notified and he or she can enter the chat. Finally, participants are notified, but the identity of who called the /sos command is kept secret. Figure 20 shows an example of how the SOS command works, when a user playing the role of “Anthony” decides to ask for help. Note that the chat participants do not know the identity of the participant who activated the SOS.



The screenshot shows a chat interface with the following messages:

- Kid Actions Admin** @admin Admin Owner 4:36 PM  
Anthony, your shy male classmate, has a great passion for classical dance. Usually he does not talk much, but today he had decided to invite the class to watch him for his ballet show.
- SOS** **Kid Actions Admin** @admin Admin Owner 4:36 PM  
Educators cannot join this conversation. If you send **/sos** (slash + sos) in the chat, the educators are notified and are allowed to the chat. Other users are notified that someone asks for help, but the caller's identity is not revealed.
- T** **Anthony** @t21-user1 4:37 PM  
Hello! I'm not sure I'm ok with this chat
- Kid Actions Admin** @admin User pr6-educator1 added by admin. 4:37 PM
- Kid Actions Admin** @admin User pr6-educator2 added by admin. 4:37 PM
- Kid Actions Admin** @admin User pr6-educator3 added by admin. 4:37 PM
- SOS** **Kid Actions Admin** @admin Admin Owner 4:37 PM  
Someone has called the SOS command, an educator will join the room soon
- SOS** **Kid Actions Help** @help-app.bot 4:37 PM Only you can see this message  
SOS received

Figure 20. Rocket.Chat example, with the description of the plot (customisable through the KAUM interface), the instructions on how to use the “/sos” command, and an example of what happens when a user call for the SOS command

## 6. THE KAMoT MONITORING TOOL

The KID ACTIONS Monitoring Tool (KAMoT) is a web-based platform designed to track hate speech and cyberbullying phenomena on Twitter. Messages posted online by users are continuously monitored through keyword-based and hashtag-based searches, focusing on English, French, and Italian.

KAMoT is released as a standalone tool and it is not integrated into the KID\_ACTIONS Digital Education platform for two main reasons: first, educators during co-creation activities were worried that, providing adolescents with a monitoring tool that would easily expose them to massive amounts of offensive content could harm them and was therefore undesirable. Second, integrating the tool into the platform would imply setting it up to make a local installation. However, this would be possible only if the tool is connected to an existing Twitter account, which has access to Twitter APIs, a requirement that is hard to meet in schools and youth centres.

The KAMoT tool was developed and used throughout the project to create a database of offensive messages, which was then used as a basis to create content that populates the platform, in particular for High School Superhero and to design Rocket.Chat scenarios. Although it is not integrated into the KID\_ACTIONS Digital Education platform, it is available as a standalone tool and we release it as an open source project that can be downloaded from Github (see Section 8 for details).

## 7. DOCKER CONTAINER

The whole KID ACTIONS Educational Platform is available on Github<sup>12</sup> as a Docker package. Docker is a framework for building, running, and managing projects that use different pieces of software that need to work together. With Docker, the final user (in our case, a school or an association) just downloads the package and runs a command, resulting in the whole platform being up and running.

While Docker is usually independent of the system where it is installed, we strongly recommend running it on a Unix-like operating system (such as Linux or macOS), where it has been extensively tested during co-creation activities.

### 7.1. Installation

Once the files are downloaded, one can just run the “docker-compose up” command, and everything should be up and running after at most 10 minutes. A parameter “-d” should be added if one wants the Platform to be executed in the background. By default, the KID ACTION Educational Platform runs on the port 8001, therefore after everything is loaded correctly one can type <http://localhost:8001/> in a browser to enter the program.

In particular, the available apps are:

- <http://localhost:8001/kaum> (User Management)
- <http://localhost:8001/chat> (Rocket.Chat)
- <http://localhost:8001/creender> (Creender)
- <http://localhost:8001/hssh> (High School Superhero)

Section 2 describes the steps to create the credentials to enter the tools (for the admin user in KAUM, see the next section).

### 7.2. System customisation

While one can run the docker file without any trouble, some settings should be defined before running the software for the first time.

With regards to **passwords**, the first time the Platform is run, a software bundle is installed. Some of them need a password for the admin user. In particular: Rocket.Chat, MySQL, KAUM.

Before running the docker composed command (see above), the system administrator has to create three text files, containing just one line with the password related to the tools. The three files are:

- mysql\_secret.txt for MySQL administration password
- rc\_secret.txt for Rocket.Chat administrator password
- admin\_secret.txt for KAUM administrator password

If one of these files is missing, the docker compose command will raise an error message.

---

<sup>12</sup> <https://github.com/dhfbk/ka-docker>

The Platform uses **MySQL as a database backend** for storing the app data. To easily surf the database data, docker will install the helper tool phpMyAdmin,<sup>13</sup> available at <http://localhost:8001/pma>.

By default, the tool is executed on **port 8001**. One can change this setting by editing the docker-compose.yml file and replacing the line containing “8001:80” with “X:80” where X is the new port number one wants to use.

To enhance the security in accessing the Platform, it is recommended to use **Secure Sockets Layer (SSL)**, the security protocol used by all the websites in the world, that is now mandatory to avoid sticky (and threatening) messages from the browser.

To get an SSL certificate free, one can just use services such as Let’s Encrypt<sup>14</sup> which gives you as many certificates as you need in less than a minute. For non-expert system administrators, tools such as Certbot<sup>15</sup> can be used to help in doing that.

The certificate is also mandatory if one wants to use Rocket.Chat from the iOS/Android app, since the https protocol is required. In addition, also a running WebSocket is needed. For this reason, an additional configuration<sup>16</sup> on the web server of the machine where the Platform is installed is needed.

Since a reverse proxy is not that easy to configure, it is suggested to expose the Rocket.Chat docker container directly to the parent machine, by editing the docker-compose.yml file and adding a “ports” configuration in the “rocketchat” container.

---

<sup>13</sup> <https://www.phpmyadmin.net/>

<sup>14</sup> <https://letsencrypt.org/>

<sup>15</sup> <https://certbot.eff.org/>

<sup>16</sup> <https://docs.rocket.chat/quick-start/environment-configuration/configuring-ssl-reverse-proxy>

## 8. DOWNLOAD AND INSTALL THE SOFTWARE

Most of the procedures to download and install the software have been already explained in Section 7. However, all the software described in this deliverable is available open source and can be downloaded and fully customised.

The Kid Actions User Management (**KAUM**) system can be downloaded from GitHub. It is distributed as an **open source package** and is released under the **Apache license** (version 2.0). The API (backend) is written in **PHP** and relies on a **MySQL database**. The web interface (frontend) is developed using the **HTML/CSS/JS paradigm** using the **Bootstrap** and **Vue.js frameworks**. The interface is responsive, so one can use it from any device that can open web pages.

The project is available here: <https://github.com/dhfbk/kaum>

The **KAMoT** tool is composed of a front-end and a back-end.

- The front-end, written in HTML/CSS/JS using the React framework, is available here: <https://github.com/Hatometer/HatometerUI>
- The back-end, written in Java, is available here: <https://github.com/Hatometer/HatometerPlatform>

To download and use the other tools, please refer to the corresponding sections.

## 9. REFERENCES

- Dana Aizenkot and Gabriela Kashy-Rosenbaum (2018). *Cyberbullying in whatsapp classmates groups: Evaluation of an intervention program implemented in israeli elementary and middle schools*. *New Media & Society*, 20(12), 4709-4727.
- Christos Baziotis, Nikos Pelekis, Christos Doukeridis (2017). *DataStories at SemEval-2017 Task 4: Deep LSTM with Attention for Message-level and Topic-based Sentiment Analysis*. Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017).
- Bonetti, F., & Tonelli, S. (2020). A 3D Role-Playing Game for Abusive Language Annotation. In *Workshop on Games and Natural Language Processing* (pp. 39-43).
- Bonetti, F., & Tonelli, S. (2021). Challenges in Designing Games with a Purpose for Abusive Language Annotation. *Proceedings of the First Workshop on Bridging Human-Computer Interaction and Natural Language Processing*.
- Bonetti, F. and Tonelli, S. (2022). An Analysis of Abusive Language Data Collected through a Game with a Purpose. *Proceedings of Games and NLP Workshop, Marseille, France*.
- Claudio Dell’Era and Paolo Landoni (2014). *Living lab: A methodology between user-centred design and participatory design*. *Creativity and Innovation Management*, 23(2):137–154.
- ISTAT. 2014. *Il Bullismo in Italia: Comportamenti offensivi tra i giovanissimi*. <https://www.istat.it/it/files/2015/12/Bullismo.pdf>
- Stefano Menini, Giovanni Moretti, Michele Corazza, Elena Cabrio, Sara Tonelli, Serena Villata (2019). *A system to monitor cyberbullying based on message classification and social network analysis*. Proceedings of the Third Workshop on Abusive Language Online, 105-110.
- Menini, S., Aproso, A. P., & Tonelli, S. (2021). A Multimodal Dataset of Images and Text to Study Abusive Language. *Proceedings of the 7th Italian Conference on Computational Linguistics, CLiC-it*.
- Palmero Aproso, A., Menini, S., & Tonelli, S. (2021). The CREENDER Tool for Creating Multimodal Datasets of Images and Comments. Proceedings of the 7th Italian Conference on Computational Linguistics, CLiC-it.
- Pournaki, A., Gaisbauer, F., Banisch, S., & Olbrich, E. (2021). *The Twitter Explorer: A Framework for Observing Twitter through Interactive Networks*. *Journal of Digital Social Research*, 3(1), 106–118.
- Saif Mohammad and Peter Turney (2013). *Crowdsourcing a Word-Emotion Association Lexicon*, *Computational Intelligence*, 29 (3), 436-465.
- Ryan, R. and Deci, E. (2000). Self-Determination Theory and the Facilitation of Intrinsic Motivation, Social Development, and Well-Being. *American Psychologist*, page 11.
- Rachele Sprugnoli, Stefano Menini, Sara Tonelli, Filippo Oncini, Enrico Piras (2018). *Creating a WhatsApp Dataset to Study Pre-teen Cyberbullying*. Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)
- Selen Turkay and Charles K. Kinzer (2014). The Effects of Avatar-Based Customization on Player Identification. *International Journal of Gaming and Computer-Mediated Simulations*, 6(1):1–25.