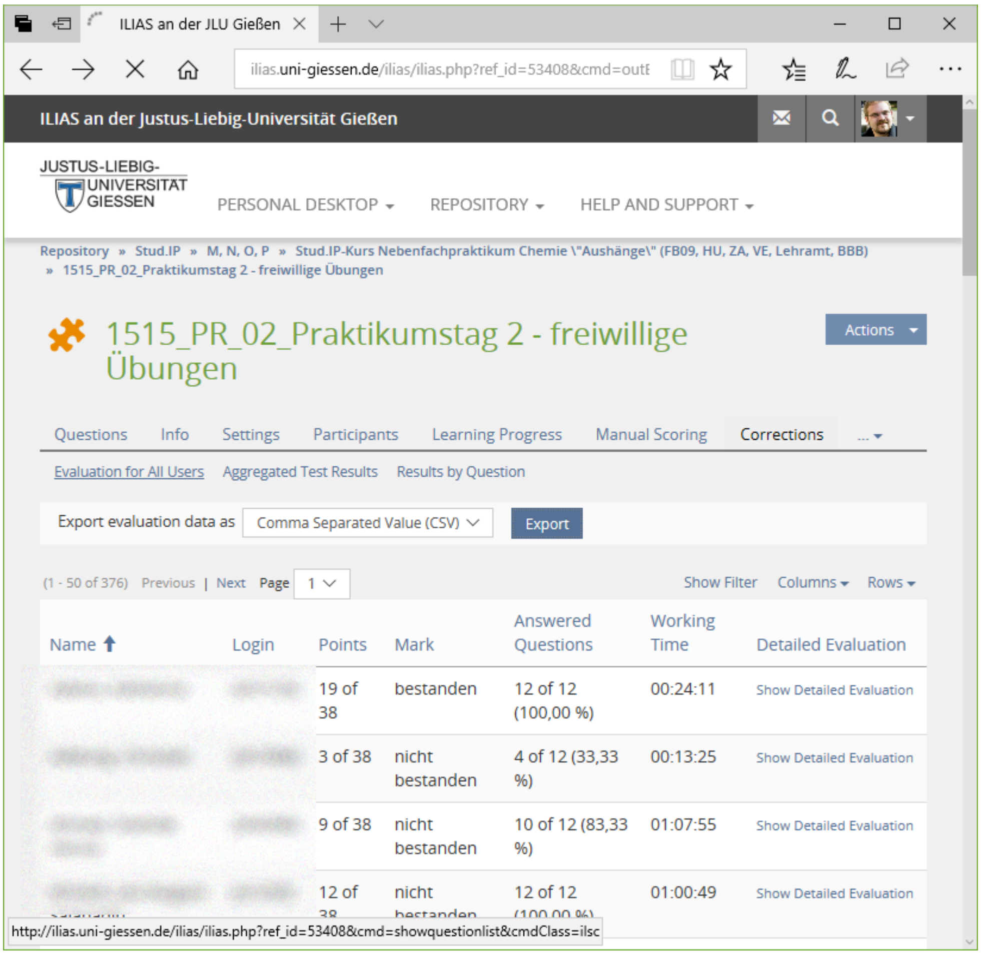
How to Use the LMSA Kit and the ESF Edit

This illustrated guide is designed to demonstrate the first steps in the utilization of the LMSA Kit and the ESF Edit.
The complete process of feedback generation is divided into small, comprehensible chapters for every stage. First of all, the data generation is explained. Secondly, the criteria are designed and evaluated. Thirdly, a feedback generation process is designed with the use of the ESF Edit depending on the criteria initially defined within the LMSA Kit. Finally, the LMSA Kit can use this feedback generation process to generate the individual formative feedbacks and send it by mail.

Import LMS Data into the LMSA Kit

In the first stage, the relevant data has to be exported from the LMS to make it available for the LMSA Kit. For that purpose, almost every LMS offers a functionality to export earned scores. “Comma Separated Values” (*.csv) must be chosen as the export file format, because the csv-format is the only format that the LMSA Kit can read at the moment. The csv-format is a basic format that can be used in every spreadsheet software and it is available in every LMS for a data export. For other LMS than ILIAS these steps are almost the same, but the presentation within the LMS may look different.



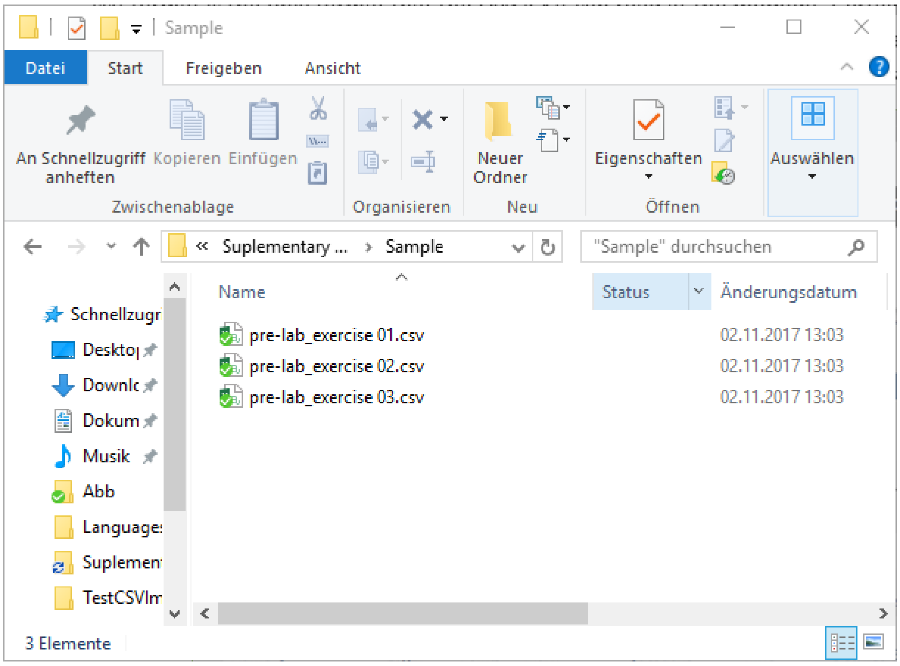
Repository » Stud.IP » M, N, O, P » Stud.IP-Kurs Nebenfachpraktikum Chemie \"Aushänge\" (FB09, HU, ZA, VE, Lehramt, BBB) » 1515_PR_02_Praktikumstag 2 - freiwillige Übungen

1515_PR_02_Praktikumstag 2 - freiwillige Übungen

Export evaluation data as: Comma Separated Value (CSV)

Name	Login	Points	Mark	Answered Questions	Working Time	Detailed Evaluation
[redacted]	[redacted]	19 of 38	bestanden	12 of 12 (100,00 %)	00:24:11	Show Detailed Evaluation
[redacted]	[redacted]	3 of 38	nicht bestanden	4 of 12 (33,33 %)	00:13:25	Show Detailed Evaluation
[redacted]	[redacted]	9 of 38	nicht bestanden	10 of 12 (83,33 %)	01:07:55	Show Detailed Evaluation
[redacted]	[redacted]	12 of 38	nicht bestanden	12 of 12 (100,00 %)	01:00:49	Show Detailed Evaluation

Fig. 1: Class's Statistics of a Test within ILIAS



Name	Status	Änderungsdatum
pre-lab_exercise 01.csv		02.11.2017 13:03
pre-lab_exercise 02.csv		02.11.2017 13:03
pre-lab_exercise 03.csv		02.11.2017 13:03

Fig. 2: Export Files ready prepared for the LMSA Kit

All class's test data (Fig. 1) should be exported in one single directory (Fig. 2). This makes the following steps easier. After all tests are exported, the LMSA Kit can be used for the analysis of the data. When the LMSA Kit starts new, most of the elements are grayed and disabled, because no project is opened (Fig. 4).

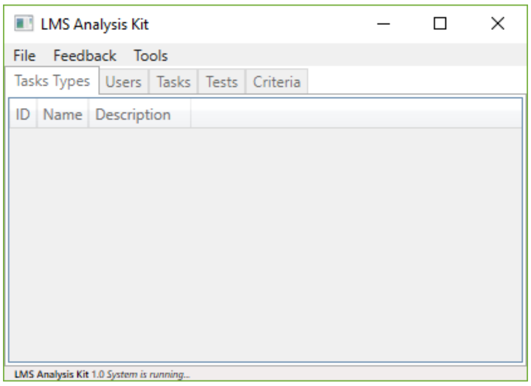


Fig. 4: LMSA Kit after the start

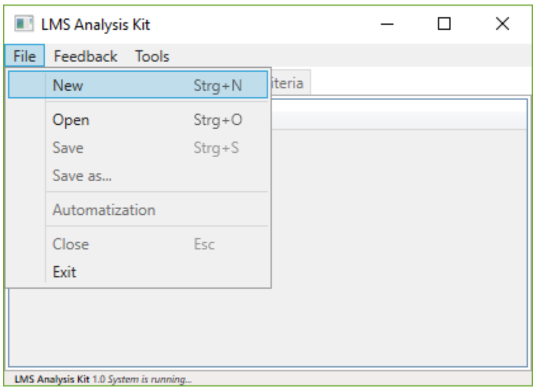


Fig. 5: Create a New Project

Changing the Language

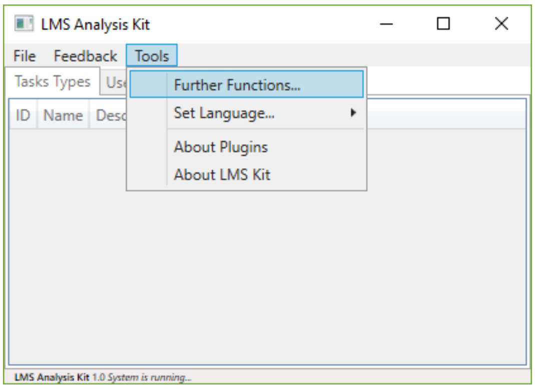


Fig. 3: Language of the Interface

If the language of the graphical user interface does not apply, it can be easily changed in the "Set Language…" menu which can be found in the tools' section of the main menu bar.

There is also the possibility to make new localizations. So, users that are able to translate one of the languages available to a new one can make their own localization and support us in translating the software.

After a new project is created, all interface elements are enabled (Fig. 5) and the import of the LMS's data is possible. The *Tests* tab must be opened to find the import option for test data (Fig. 6). The main menu of the LMSA Kit is adaptive which means that only relevant options are displayed. The options for tests are only visible when the *Tests* tab is displayed.

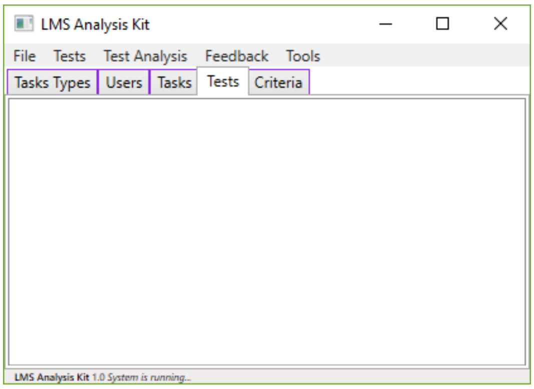


Fig. 6: Select the Tests tab as active tab

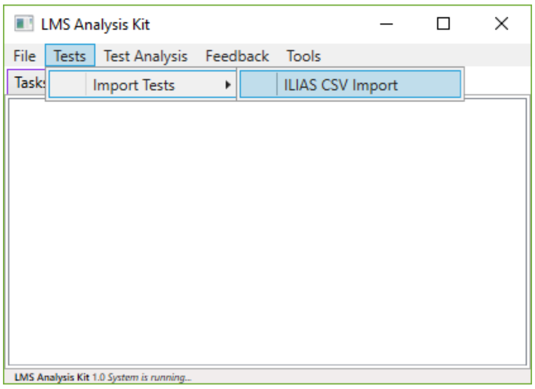
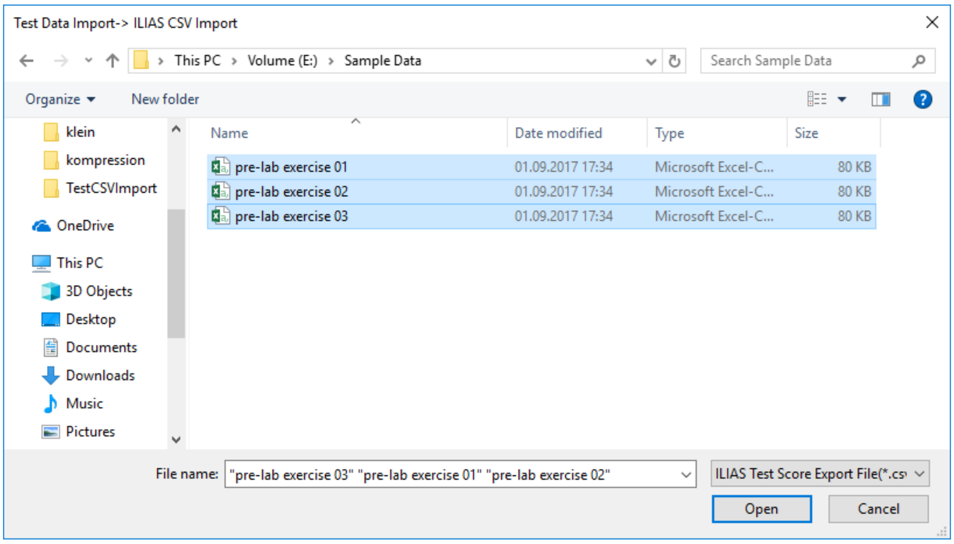


Fig. 7: Import the ILIAS's export files

Select the *Tests* tab. After that, the *Tests* and the *Tests Analysis* options are visible in the main menu (Fig. 6, Options between *File* and *Feedback*). Open the *Tests* menu and hover the cursor over *Import Tests*, then click on *ILIAS CSV Import* (Fig. 7). It opens a file open dialog. In this dialog the data exports previously generated are selected (Fig. 8). The dialog allows a multiple file selection, so all files could be imported at once.

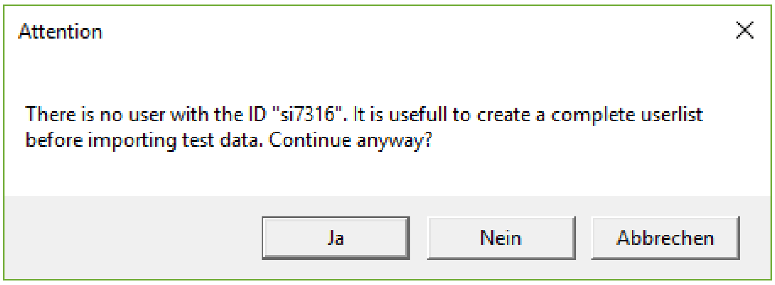


Name	Date modified	Type	Size
pre-lab exercise 01	01.09.2017 17:34	Microsoft Excel-C...	80 KB
pre-lab exercise 02	01.09.2017 17:34	Microsoft Excel-C...	80 KB
pre-lab exercise 03	01.09.2017 17:34	Microsoft Excel-C...	80 KB

File name: "pre-lab exercise 03" "pre-lab exercise 01" "pre-lab exercise 02"

ILIAS Test Score Export File(*.cs

Fig. 8: Select all export files then press open

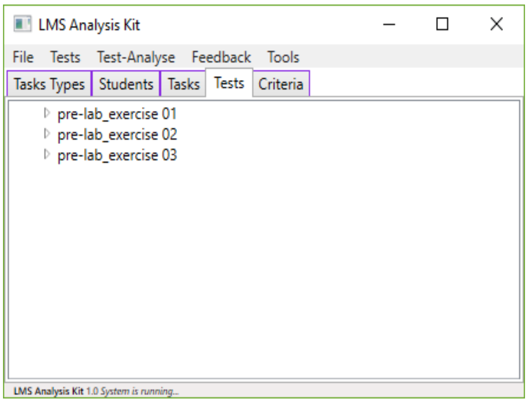

Attention

There is no user with the ID "si7316". It is usefull to create a complete userlist before importing test data. Continue anyway?

Fig. 9: Warning User List was not Imported First

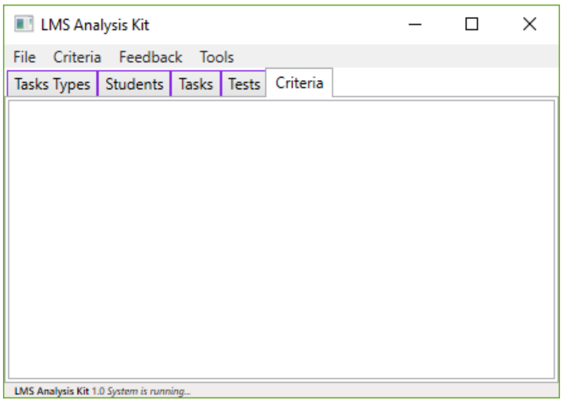
If following this guide, there will be a warning that the users found in the test data could not be located in the project's user list (Fig. 9). You can anyway continue, because the missing users will be automatically created. If you use ILIAS at your university, you can import the user list first as it is explained in the appendix to this guide. It will shorten one step when the feedback is sent in a later step.

Define and Analyze Criteria


LMS Analysis Kit
File Tests Test-Analyse Feedback Tools
Tasks Types Students Tasks Tests Criteria
pre-lab_exercise 01
pre-lab_exercise 02
pre-lab_exercise 03

LMS Analysis Kit 1.0 System is running...

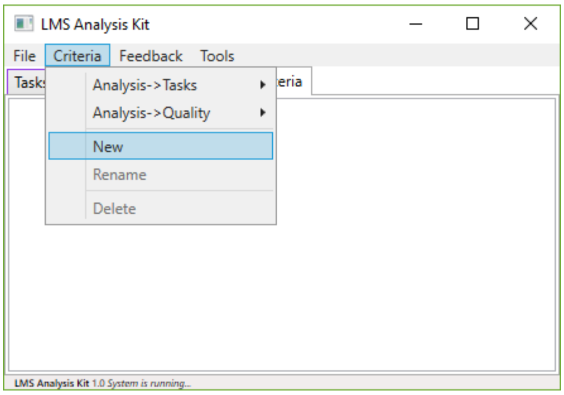
Fig. 10: The Tests Tab after the import


LMS Analysis Kit
File Criteria Feedback Tools
Tasks Types Students Tasks Tests Criteria

LMS Analysis Kit 1.0 System is running...

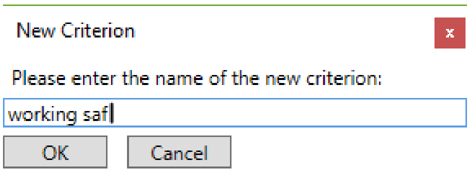
Fig. 11: The Criteria Tab

After the test data has been imported, the criteria can be defined. The *criteria* tab offers the functionality required for that (Fig. 11). Hence, mark the criteria tab as the active tab. The adaptive main menu hides the tests specific options and the *Criteria* section becomes visible (Fig. 11, between *File* and *Feedback)*. Click on “New” to create a new criterion (Fig. 12). A dialog box opens (Fig. 13). The name of the new criterion should be entered in it. Finally, click on “OK” to close the dialog box and to create the criterion. The criterion shows up in the list (Fig. 14).



LMS Analysis Kit 1.0 System is running...

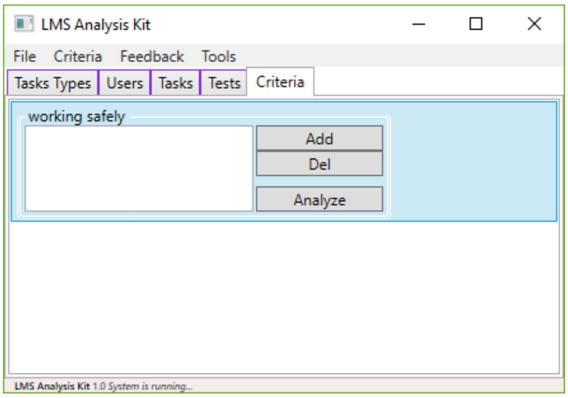
Fig. 12: Add a new criterion



Please enter the name of the new criterion:

working saf

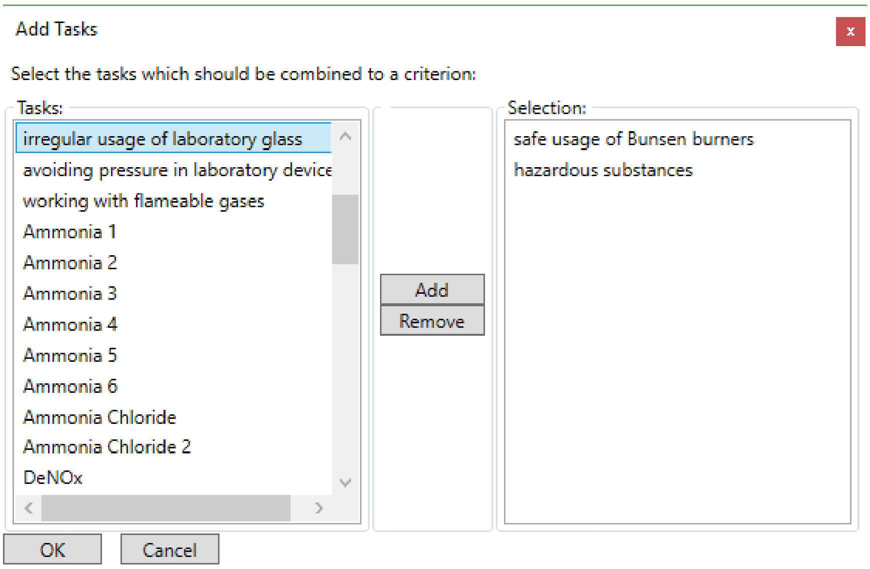
Fig 13: Name the new criterion


LMS Analysis Kit
File Criteria Feedback Tools
Tasks Types Users Tasks Tests Criteria
working safely
Add
Del
Analyze

LMS Analysis Kit 1.0 System is running...

Fig. 14: New, empty criterion in list

The criterion created is still empty. To fill it with tasks, just click on the “Add” button. By doing this, a dialog box appears which allows to select the tasks belonging to the criterion (Fig. 15).



Add Tasks

Select the tasks which should be combined to a criterion:

Tasks:

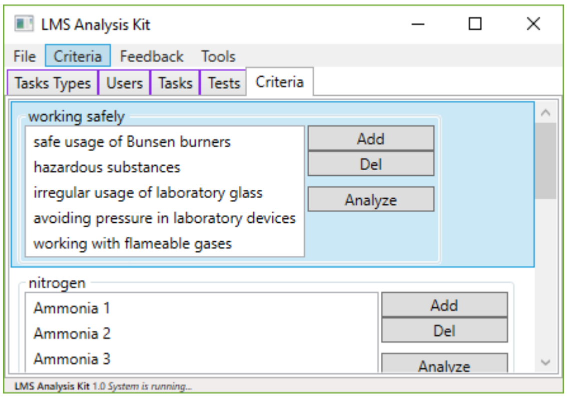
- irregular usage of laboratory glass
- avoiding pressure in laboratory device
- working with flameable gases
- Ammonia 1
- Ammonia 2
- Ammonia 3
- Ammonia 4
- Ammonia 5
- Ammonia 6
- Ammonia Chloride
- Ammonia Chloride 2
- DeNOx

Selection:

- safe usage of Bunsen burners
- hazardous substances

Fig. 15: Add tasks to the criterion

On the left-hand side there is a list containing all available tasks of the current project. To add a task to the criterion, just select it in the list and click “Add”. It will move to the other side in to the list of selected tasks. This list contains all tasks belonging to the current criterion. In order to remove a task, click on the task and then on “Remove”. The task disappears in the list on the right-hand side and can be found again in the list of available tasks. In this way, all criteria can be defined. In the end, the list of criteria is filled with the defined criteria connected to different tasks (Fig. 16).



LMS Analysis Kit

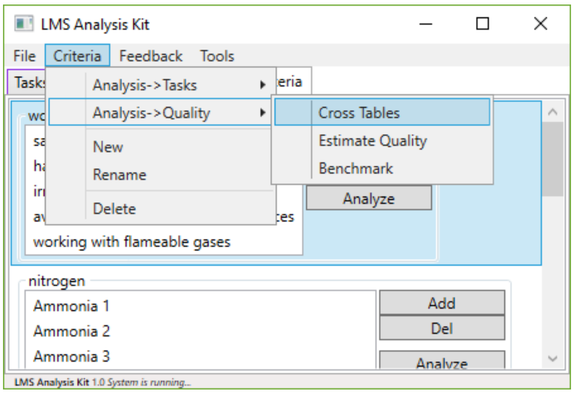
working safely

- safe usage of Bunsen burners
- hazardous substances
- irregular usage of laboratory glass
- avoiding pressure in laboratory devices
- working with flameable gases

nitrogen

- Ammonia 1
- Ammonia 2
- Ammonia 3

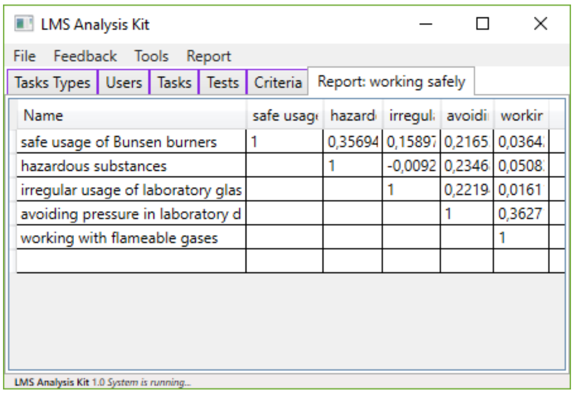
Fig. 16: Filled list of criteria



LMS Analysis Kit

Fig. 17: Generate cross tables for a criterion

In the next step each criterion should be evaluated, if the combination of tasks is really describing only a single identifiable learning objective, competence, or thematic topic. Select the criterion you want to analyze. Go into the *Criteria* menu and further to the *Analysis->Quality* section (Fig. 17). Click on *Cross Tables*. A new tab appears containing a report with the cross table of the selected criterion.

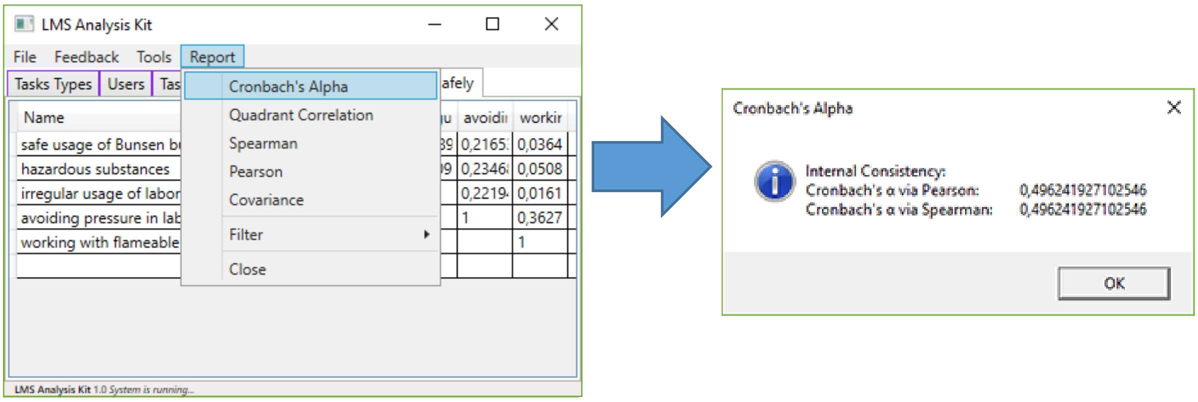


LMS Analysis Kit

Name	safe usag	hazard	irregul	avoidi	workir
safe usage of Bunsen burners	1	0,35694	0,1589	0,2165	0,0364
hazardous substances		1	-0,0092	0,2346	0,0508
irregular usage of laboratory glas			1	0,2219	0,0161
avoiding pressure in laboratory d				1	0,3627
working with flameable gases					1

Fig. 18: Cross table of a criterion

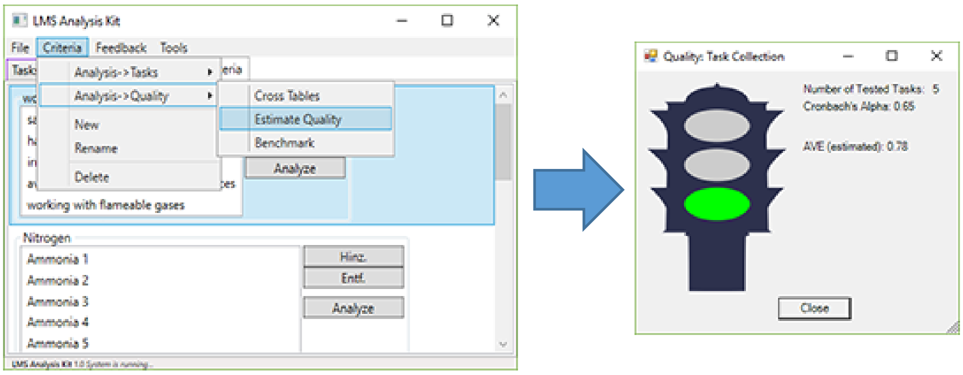
There are different correlation coefficients available as cross table (Fig. 18). They usually measure the correlation between two variables, in this case, two tasks. In the cross table every task is correlated with every task. A correlation of 1 represents a perfect correlation in most types of coefficients. Consequently, there can be seen only correlations with a value of 1 on the diagonal, because there are only tasks correlated with themselves. Additionally, the internal consistency in terms of Cronbach's alpha can be estimated (Fig. 19). Cronbach's alpha allows also an easy estimation if the tasks are measuring the same criterion. Nevertheless, it offers no absolute guarantee that the scale is one-dimensional and finally all technical possibilities in analysis do not relieve educators from their duty to construct criteria carefully and fruitfully.



Internal Consistency:
Cronbach's α via Pearson: 0,496241927102546
Cronbach's α via Spearman: 0,496241927102546

Fig. 19: Estimate Cronbach's Alpha

Currently, we are working on an improved procedure to estimate the reliability, but the plugin designed for this purpose is in an early stage of development. The traffic lights offer a comprehensible estimation whether the combination of tasks is appropriate (Fig. 20). Additionally, some information supporting the estimation are shown. Later, further suggestions how to improve an inappropriate combination will also be offered.



Number of Tested Tasks: 5
Cronbach's Alpha: 0.65

AVE (estimated): 0.78

Fig. 20: Check the reliability of a criterion

Finally, the project must be saved. To save the project, click on the "Save" button in the "File" section of the main menu. A save file dialog opens and you can choose a destination for the project file.

Create the Feedback Generation Process

For the next stage, the ESF Edit is required. After the LMS's data is prepared within the LMSA Kit and the criteria are defined as well as positively evaluated, the editing of the feedback texts can start. Start the ESF Edit and open a new project (Fig. 21).

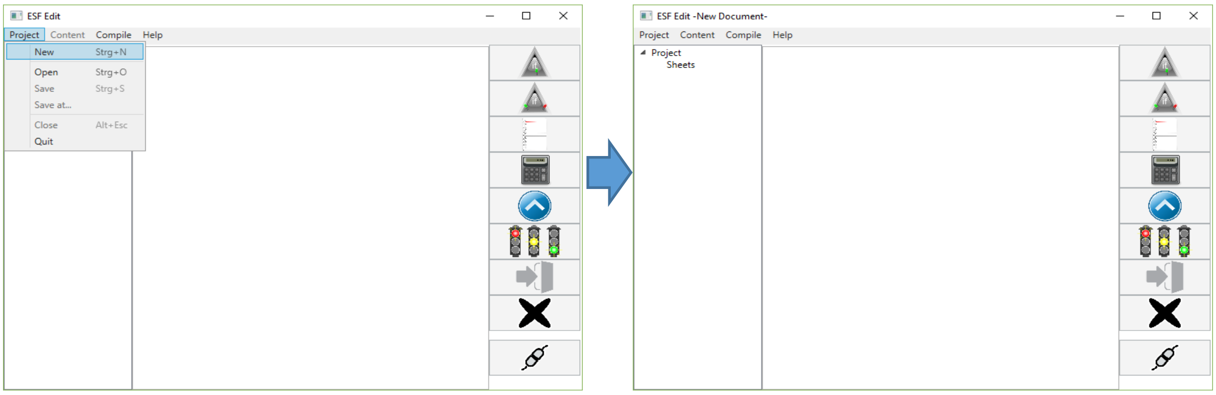


Fig. 21: A new project in the ESF Edit

Firstly, add a main sheet to the project. To do this, open the *Content* section of the main menu. Click on "New Scheme" (Fig. 22). A new scheme is displayed in the project tree on the left-hand side and is also visible in the tab bar in the middle of the ESF Edit.

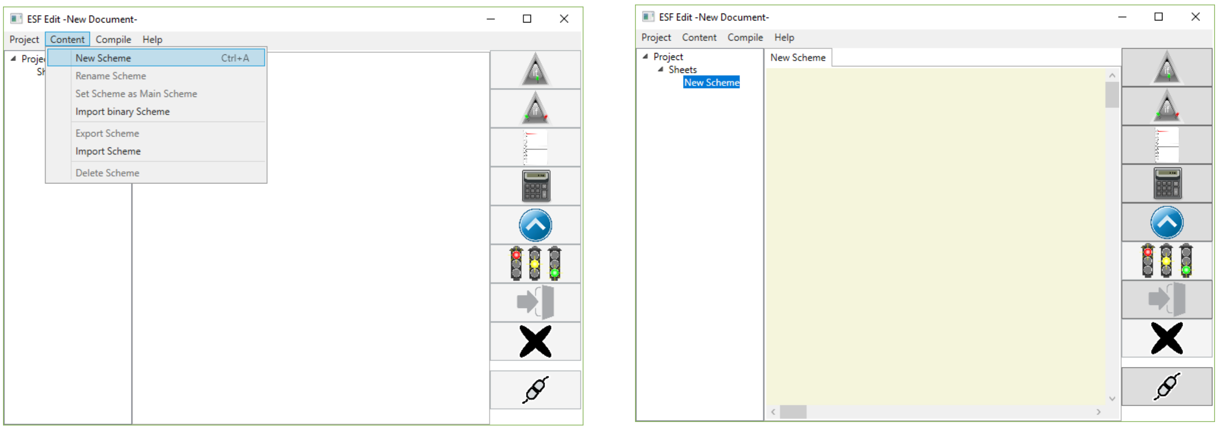
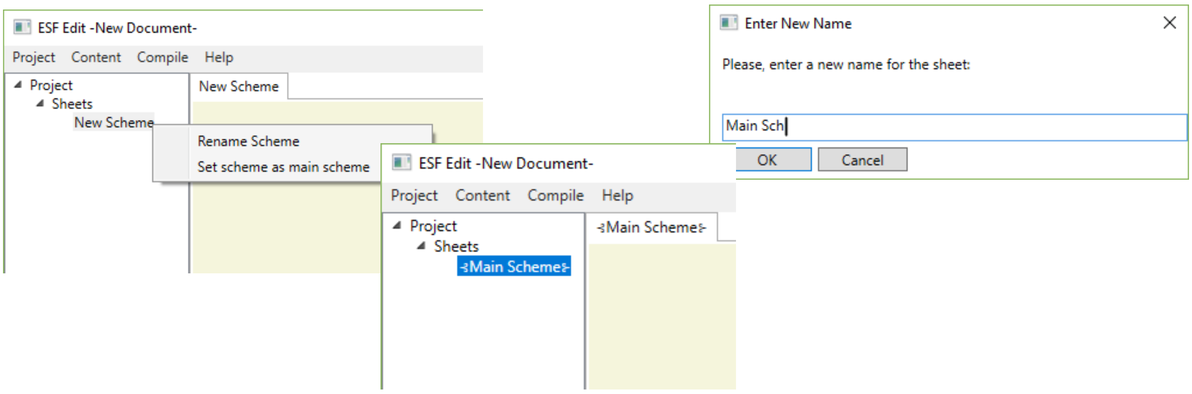


Fig. 22: Add a new scheme

It is useful to name the new scheme to main scheme or something similar. Set this scheme as the main scheme of the project (Fig. 23). The feedback generation starts later in this scheme if there is more than one. All changes can be done by right clicking on the scheme in the tree on the left-hand side. If the scheme is successfully set as the main scheme, the name is highlighted by two symbols surrounding the name.



Please, enter a new name for the sheet:

Main Sch

Fig. 23: Change the new scheme to the main scheme

The data of the LMSA Kit project must be gathered by a script. So insert a *Script Snippet* next to the *Start Snippet* of the main scheme. Click on the *Script Snippet* on the right-hand side, then right click next to the *Start Snippet*. The *Snippet* marked shows up where the cursor is (Fig. 24). Click on the snippet and keep the mouse button pressed to move the snippet and rectify the position eventually.

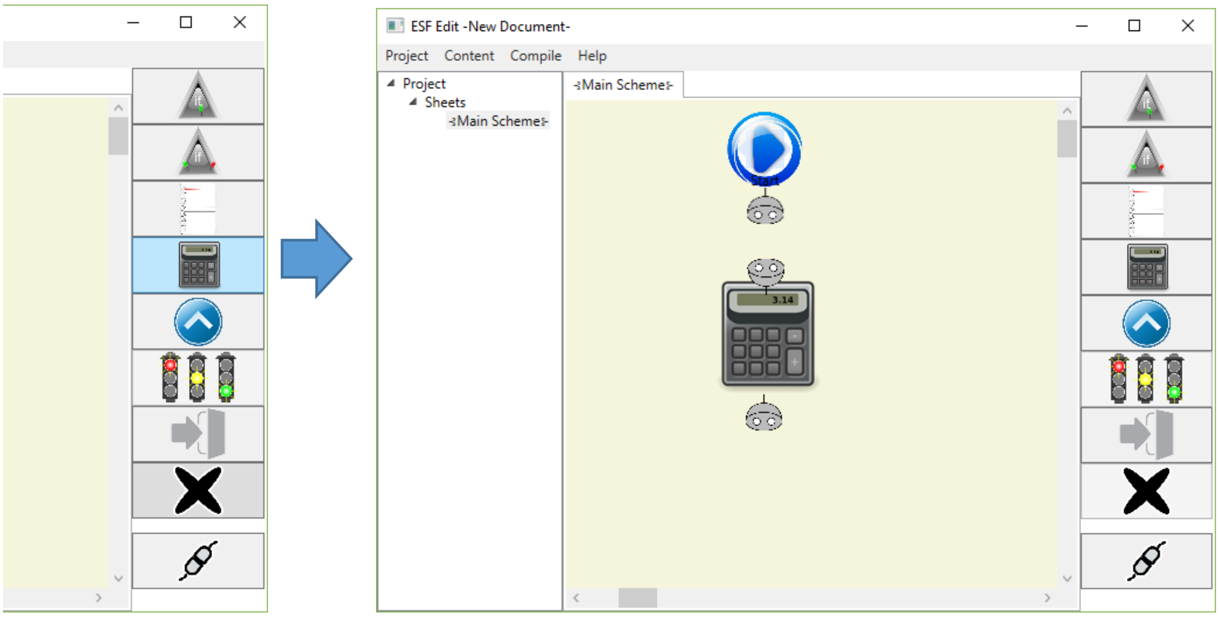


Fig. 24: Insert a Script Snippet

Wire the *Script Snippet* with the *Start Snippet*. Select the wires on the right-hand side. Then right click on two destinations next to the two *Snippets*. Click on one plug and keep the mouse button pressed. Move the plug to an empty socket and do it in the same manner with the remaining. After this, the two *Snippets* are connected by wires (Fig. 25).

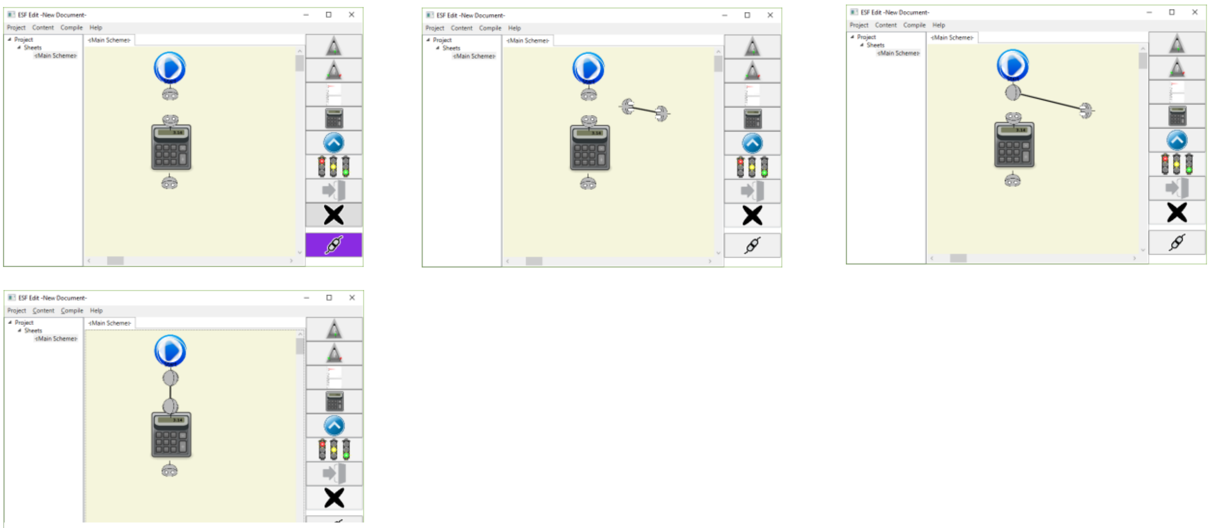


Fig. 25: Wire the Script Snippet

Before the next steps the code helpers should be initialized. They will help you to write the script required for collecting the data of the LMSA Kit. Open the *Help* section of the main menu and click "Connect with LMSA Project" (Fig. 26). An open file dialog shows up. Choose the previously saved LMSA Kit project. In this way, all script editors will offer specific help during scripting.

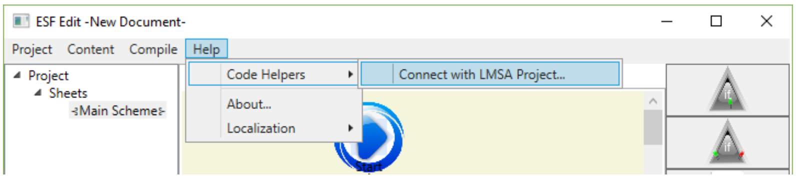


Fig. 26: Connect the ESF Project to the LMSA Kit File

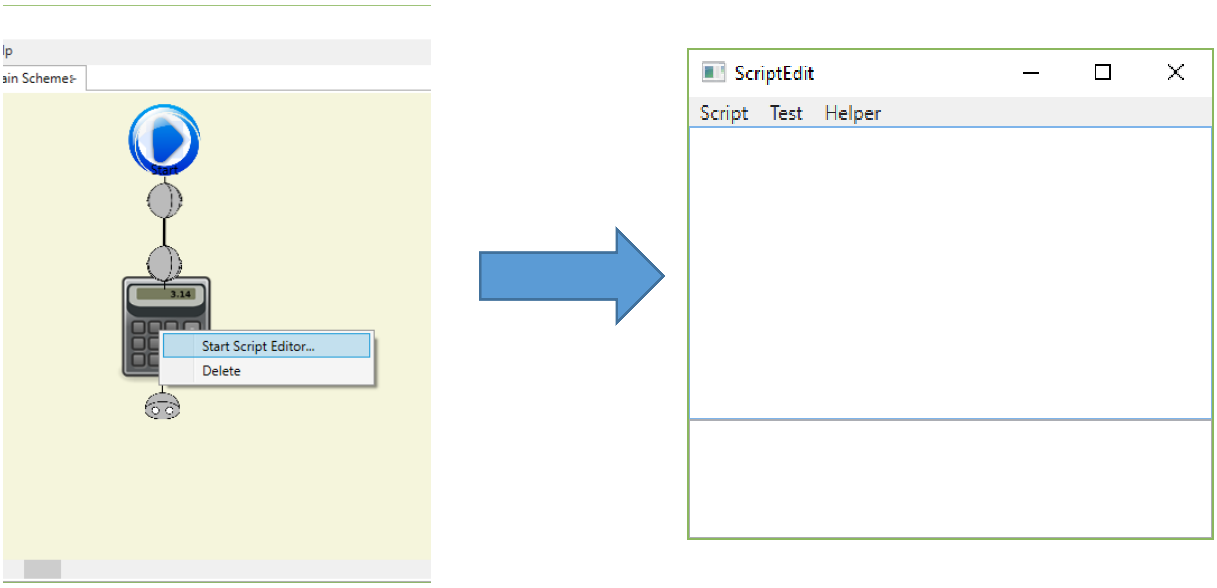
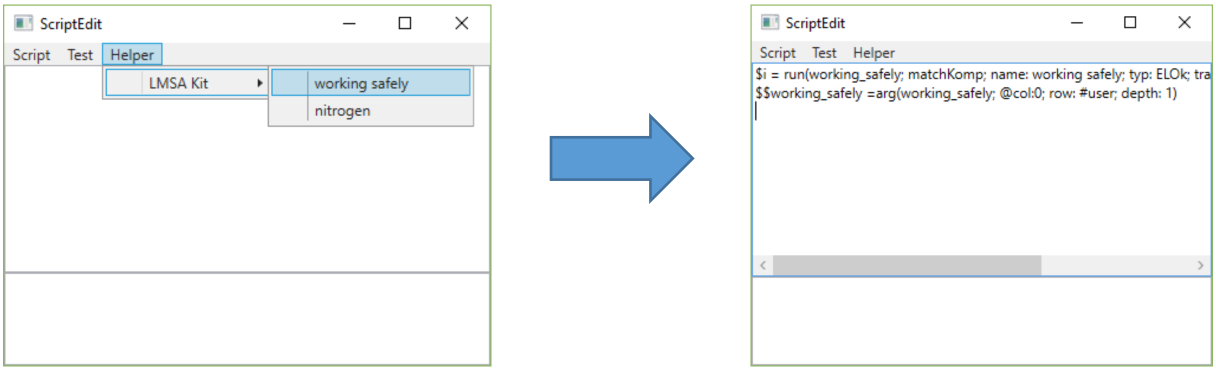


Fig. 27: Start the script editor

After the code helpers are initialized, the editing of the script can begin. Perform a right click on the *Script Snippet* and choose “Start Script Editor…” to start the script editor (Fig. 27). The underlying script of the *Snippet* can be used to query the student’s scores in the criteria of the LMSA Kit project. The *Helper* sections supports the user in editing this script. Open the *Helper* menu and go to the “LMSA Kit” section (Fig. 28). There are all criteria of the previously selected LMSA Kit file. Select one criterion and two lines of script will be inserted. The first line let the LMSA Kit estimate the students’ abilities in the specific criterion. Different default parameters of the estimation are inserted but can be changed if required. The second line queries the ability score for the current user from the LMSA Kit and stores the value in the same-named global variable. This global variable is available in all *If Snippets*.



```
$i = run(working_safely; matchKomp; name: working safely; typ: ELOk; tra
$$working_safely =arg(working_safely; @col:0; row: #user; depth: 1)
```

Fig. 28: Use the helper for support in editing the scripts

In the same manner, all other criteria are inserted into the script. When the feedback is later processed, this script is executed and all criteria are estimated and the values are stored as global values. When all criteria are queried by the script, the actual design of the feedback text is feasible. If you follow our block-wise design, we recommend to handle every feedback block in a new scheme for reasons of clarity and comprehensibility.

Write a Feedback Block

For starting with a new feedback block, add and name a new scheme. Do the same steps previously done when adding the main scheme (cf. Fig. 23), just without marking the scheme as main scheme. Subsequently, make the new scheme to the active scheme in the tab bar in the middle (Fig. 29).

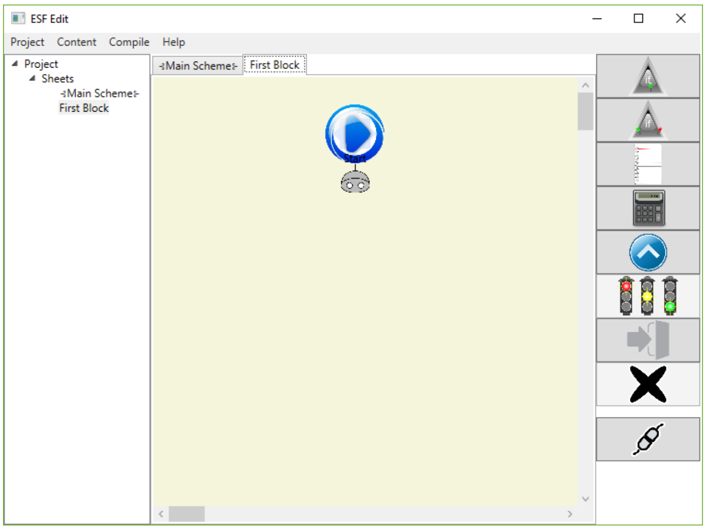


Fig. 29: New feedback block in an ESF Project

The next step should be an introduction to the current feedback block. Tell the students what aspects of their performance are currently taken into account. Select a *Message Snippet* on the bar at the right-hand side. Make a right click next to the *Start Snippet* and wire the *Message Snippet* like the *Script Snippet* previously done above (Fig. 30). Afterwards right click on the *Message Snippet* to edit the message.

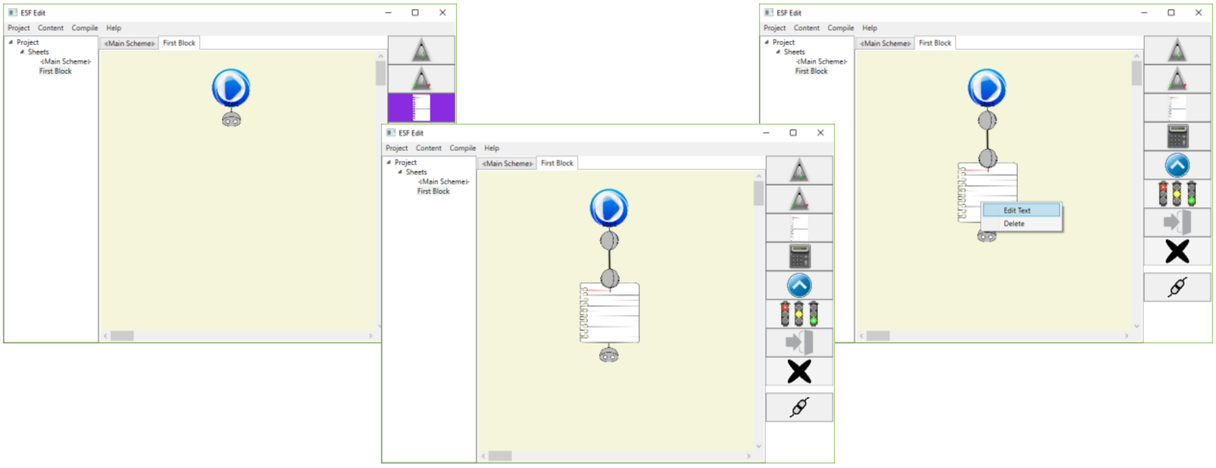
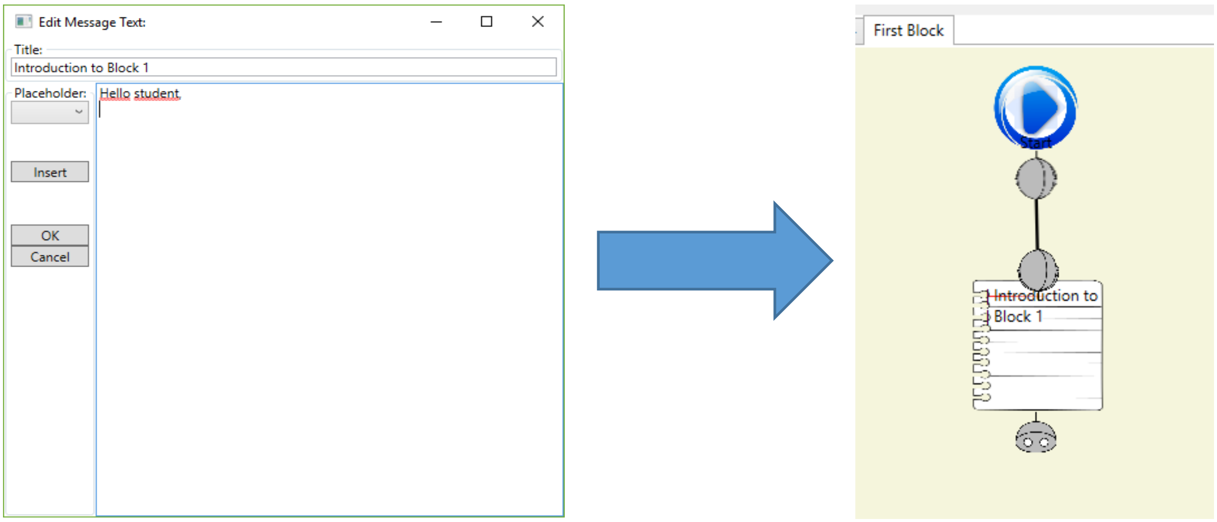


Fig. 30: Add a Message Snippet

The *Message Snippet* is the instrument to write the actual feedback text for the students. Every time the program execution reaches a *Message Snippet*, its text underlying will be written to the final student's feedback text. There are two text boxes in the editor of the *Message Snippet (Fig. 31)*. The first text box is the title box on the top of the dialog. The title you enter there is later displayed on the *Message Snippet*. Consequently, a concise title will help you and others to identify easily with what for information the *Message Snippet* is dealing. The bigger text box contains the text that will later be written to the final feedback if the execution of the feedback program passes the *Message Snippet*.



Edit Message Text:

Title: Introduction to Block 1

Placeholder:

Hello student

Fig. 31: Edit the text of a Message Snippet

The specifity of the feedback generated comes with the *If Snippets*. So put an *If Snippet* beneath the introduction *Message Snippet* and wire it. In contrast to all other *Snippets* which were already presented, the *If Snippet* has two output plugs at the bottom. You deposit a condition within the *If Snippet*. If the condition is later evaluated as a true statement the execution continues on the left plug, the green output, otherwise it continues on the right-hand side, the red output. Right click to start editing the condition (Fig. 32).

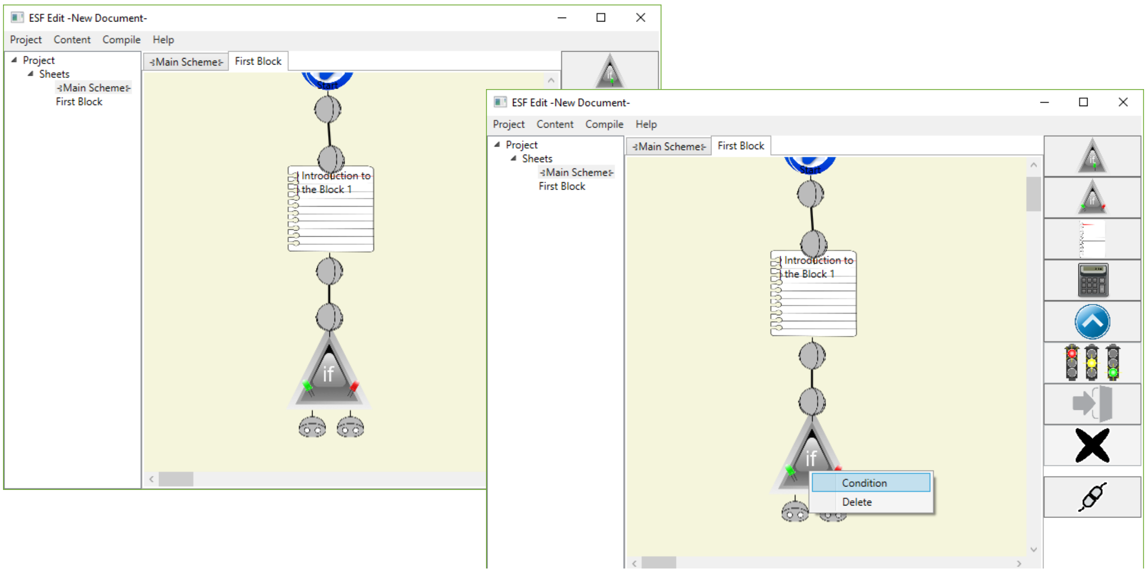
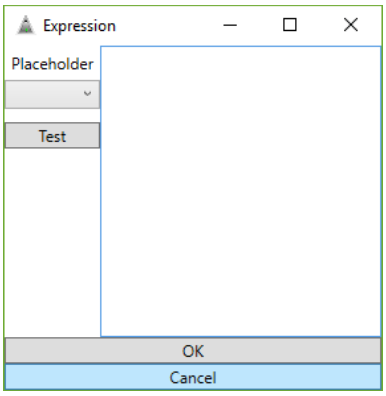
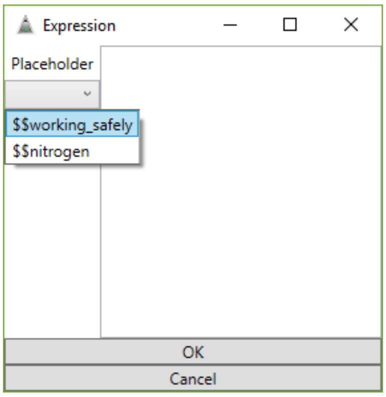
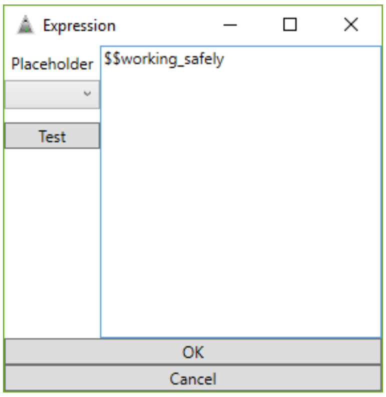


Fig. 32: Edit an If Snippet

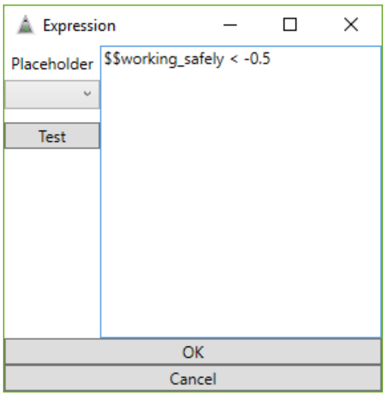




$$working_safely
$$nitrogen



$$working_safely



$$working_safely < -0.5

Fig. 33: Usage of the Expression Editor

The *Expression Editor* is the powerful tool behind the *If Snippets* (Fig. 33). The condition is formulated in it like an equation. The box directly under the text "Placeholder" contains all global variables that were set by a *Script Snippet*. Select just the global variable desired and its text will be inserted into the expression text.

An average performance is estimated with a value of 0. So a critical performance begins maybe with a value of -0.5. We want to address all students with a critical performance. That is why we formulate the expression "$$working_safely < -0.5". If the students' performance is critical, the program execution continues on the green side. If the performance is better, it continues on the other side. There should follow more *If Snippets* to make the feedback generated more specific (Fig. 34). So the program goes from one *If Snippet* to the next and, thereby, becomes specific.

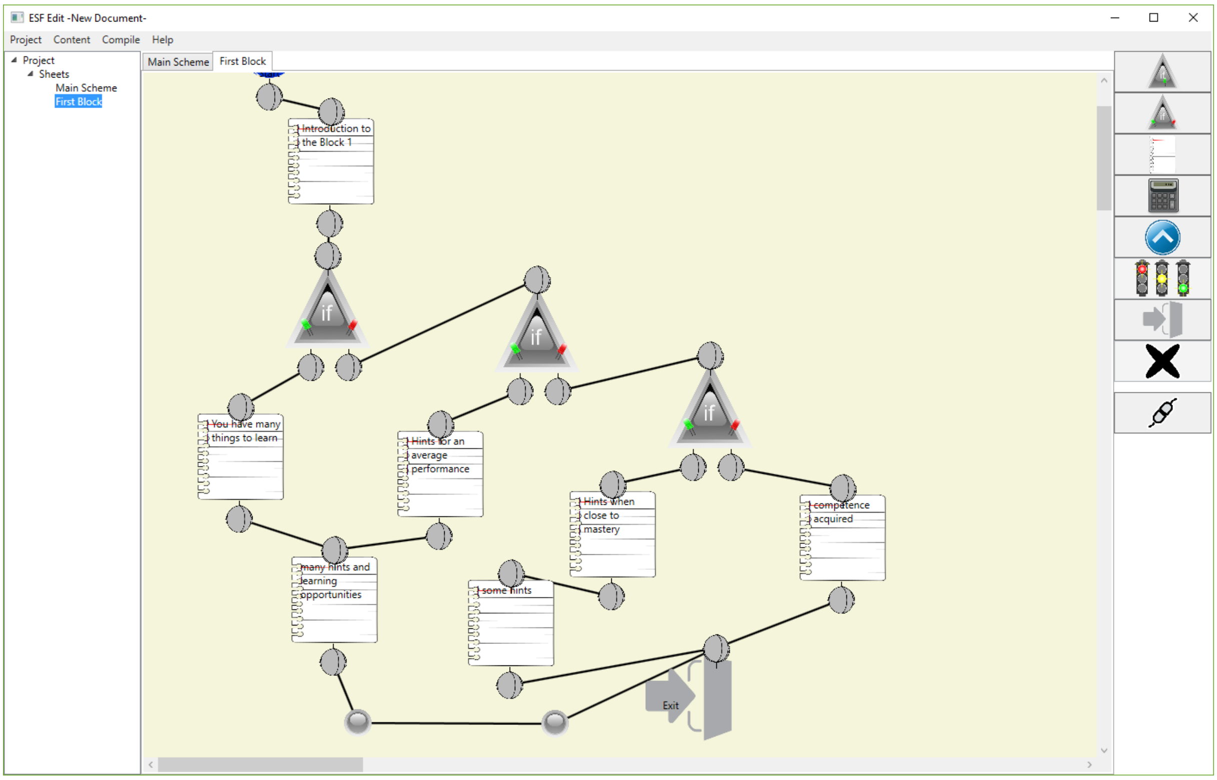

ESF Edit -New Document-
Project
Content
Compile
Help
Project
Sheets
Main Scheme
First Block
Main Scheme
First Block
Introduction to the Block 1
if
if
if
You have many things to learn
Hints for an average performance
Hints when close to mastery
competence acquired
many hints and learning opportunities
some hints
Exit

Fig. 34: View of a finished feedback block

As you can also see in Fig. 34, all different connections end in an *Exit Snippet*. You are free to use only one in a scheme as depicted or to put more than one in it. The important thing is that every connection must end at an exit, otherwise there will be an error later when compiling the project.

You can make as much different feedback blocks as you need for your feedback project.

If you are done with the design of your feedback texts, go back to the main scheme. You have to tell the main scheme to jump into the other schemes. Select the main scheme as active tab in the middle and put a *Jump Snippet* on the main scheme (Fig. 35). A right click on the *Jump Snippet* opens a specific menu. Choose "Define Jump Target".

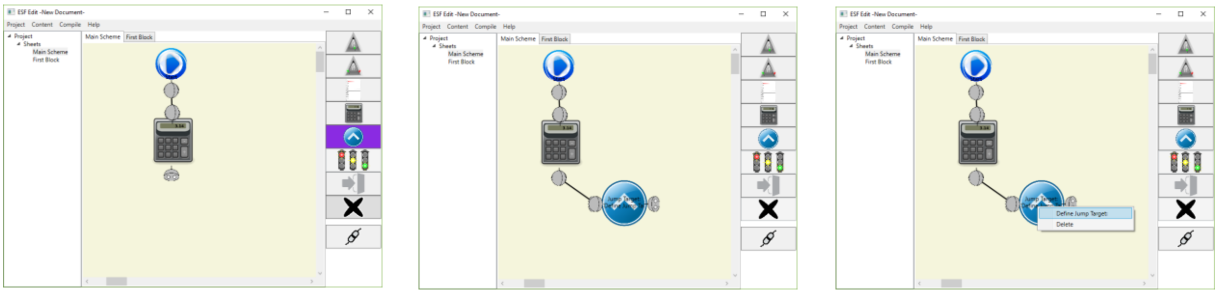
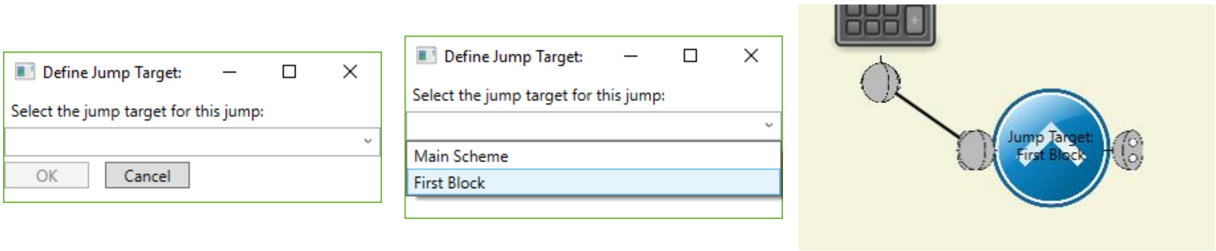


Fig. 35: Add a Jump Snippet *and define the target*

The right click opens a new dialog box where the target can be defined (Fig. 36). Click on the combo box and a drop down list appears. It contains all valid jump targets, so just select "First Block" and accept the changes. Afterwards, the *Jump Snippet* shows the new target in its label. When the execution reaches the *Jump Snippet*, it continues with the *Start Snippet* of the scheme deposited as jump target. Finally, after the scheme in which the jump was carried out reaches its end, the program continues on the other side of the *Jump Snippet*.


Define Jump Target:

Select the jump target for this jump:

Fig. 36: Set the jump target

If you have more than one feedback block and you follow our recommendations, then there will be a sequence of different *Jump Snippets* for all feedback blocks and in the end an *Exit Snippet* for the whole feedback generation process (Fig. 37). The main scheme is also the place where general information like an introduction or some closing words for all students are realizable by *Message Snippets*.

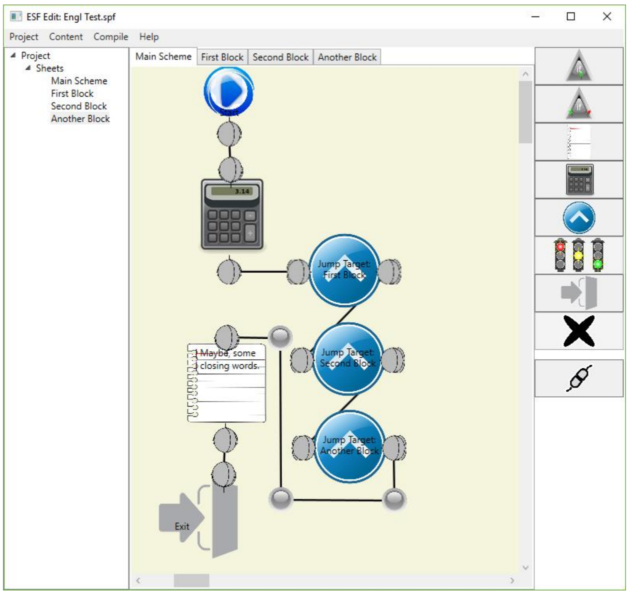


Fig. 37: An Example for a Final Edited Main Scheme

Make the Feedback Generation Process Available for other Software

The feedback generation process as it is designed in the ESF Edit is usually saved in its own project file format. The format is directly connected with the ESF Edit and no other software can use it. It contains ESF Edit specific and for other purposes unnecessary information, for instance, the different schemes and their names, the location of the *Snippets* etc. Consequently, the project must be transferred in a clean and easy to execute file format describing how the feedback should be composed together. This process is called “Compilation”. It is only possible, if the main scheme attribute is set to a scheme, because otherwise the ESF Edit has no change to select a single *Start Snippet* as the beginning of the feedback process.

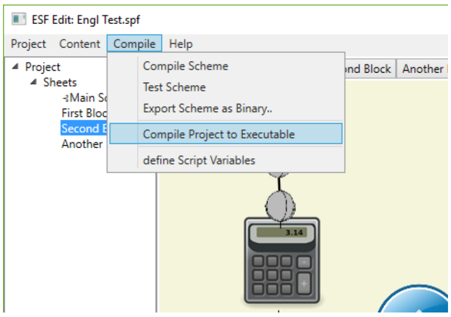


Fig. 38: Start the compilation of the feedback

Open the *Compile* section of the main menu. Select “Compile project to executable” (Fig. 38). If this entry is grayed, make sure that there is a main scheme in the project. The other options are mostly for advanced purposes except the “Test Scheme” function. A click on “Test Scheme” tries to compile the currently active scheme and displays the errors and hints how to solve them if errors occur during compilation.

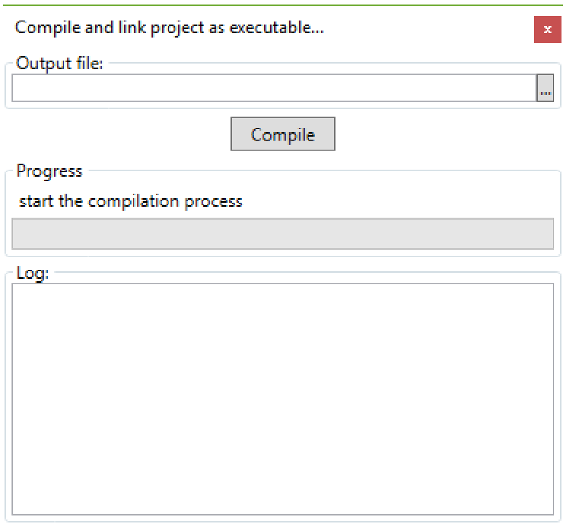

Compile and link project as executable...
×
Output file:
...
Compile
Progress
start the compilation process
Log:

Fig. 39: Compile dialog

A new dialog box opens (Fig. 39). This dialog will guide you through the process of compilation of the feedback process. Enter the destination for the compiled feedback process into the first textbox “Output file”. The button with the “...” opens a save file dialog that makes it easier to select the right destination. After the destination is set, just click the “Compile” button.

The process bar informs you about the progress and in the log all relevant information about the compilation process are written.

When no errors occur during the compilation an executable feedback process is saved as Snippet Executable File (*.sxe). Otherwise they are written in the log and can be further inspected with the “Test Scheme” function.

The file generated is required by other software like the LMSA Kit to process the feedback. In this way the different tasks of create a feedback generation process as well as collecting and analyzing the LMS data are separated from each other. The Snippet Executable can be used over the course of years, if no changes occurred in the structure of the class.

Generate an Individual Formative Feedback

After successfully compiling the feedback generation process, start the LMSA Kit again and open the previously saved project. Then open the *Feedback* section of the main menu. Click the “Deliver Feedback” option (Fig. 40). A new dialog box will open which guides you through the remaining part of the feedback delivery process (Fig. 41).

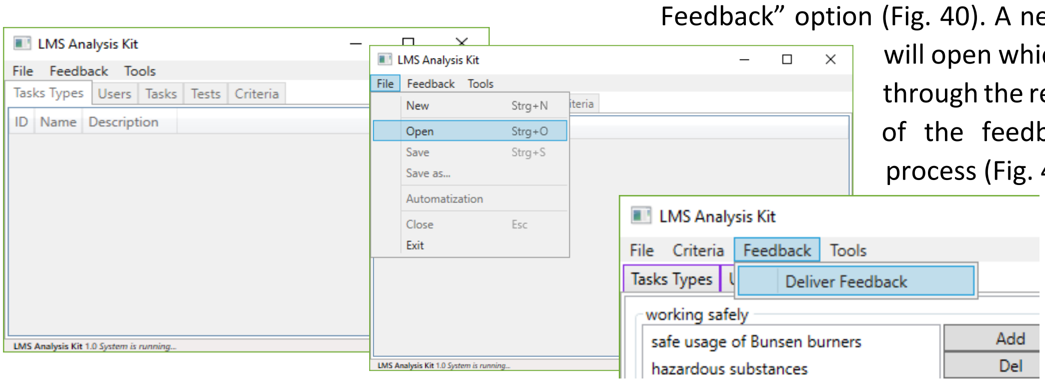
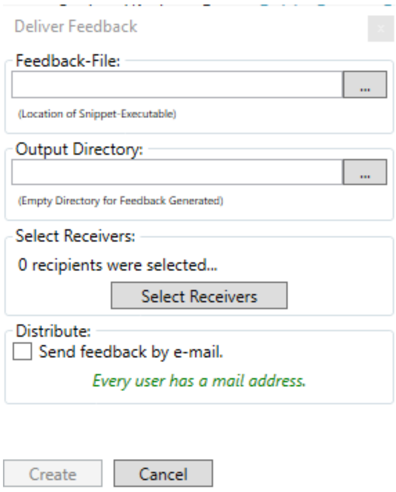


Fig. 40: Open the saved LMSA Kit Project and initiate the feedback delivery


Deliver Feedback

Feedback-File:

(Location of Snippet-Executable)

Output Directory:

(Empty Directory for Feedback Generated)

Select Receivers:

0 recipients were selected...

Distribute:

☐ Send feedback by e-mail.

Every user has a mail address.

Fig. 41: Deliver feedback dialog

The first text box defines the location of the Snippet Executable generated with the ESF Edit. The second box defines the output directory for the feedback. It should be empty. The user receives further support during both inputs by clicking on the “…” button. If the button is clicked, a save file dialog respectively a browse folder dialog is opened to name the destination.

In the next box the recipients of the feedback must be selected. A click on “Select Receivers” opens a new dialog where all users of the course can be selected which should receive a feedback (Fig. 42). All feedback texts are placed in the output directory as text file titled like the respective receiver. If the feedback should be also sent to them via e-mail, check the corresponding check box in the next group box.

If the users’ data had been imported separately before the tests have been imported, there are already valid e-mail addresses with every user entry. Otherwise, the green message “Every user has a mail address” is replaced by message in red informing about missing e-mail addresses.

There are two options how to solve that issue. You can close this dialog and update all users’ data with valid e-mail addresses or you click on the red text and a file open dialog shows up. In this dialog a csv-formatted data set can be selected. The data set must be a table with two columns: the first containing the unique user id, the second containing the valid e-mail address. A csv-file can be created with every spreadsheet program. The missing e-mail addresses are completed with them from the csv-file. If all users have a valid e-mail address, the text changes to the green one.

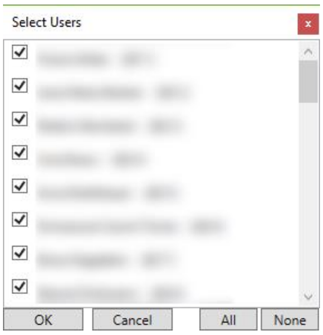


Fig. 42: Select receivers

The process of feedback generation starts with a click on “Create”. We recommend, especially, when a feedback is delivered for the first time, to create the messages without sending them directly via e-mail. Take the time and proofread some of the generated texts. Otherwise you risk to distribute a huge amount of erroneous feedback mails to your students, because if the check box is checked the feedback will be sent directly. There is no further confirmation dialog about this step.

Appendix for ILIAS Users

Importing User Lists from ILIAS

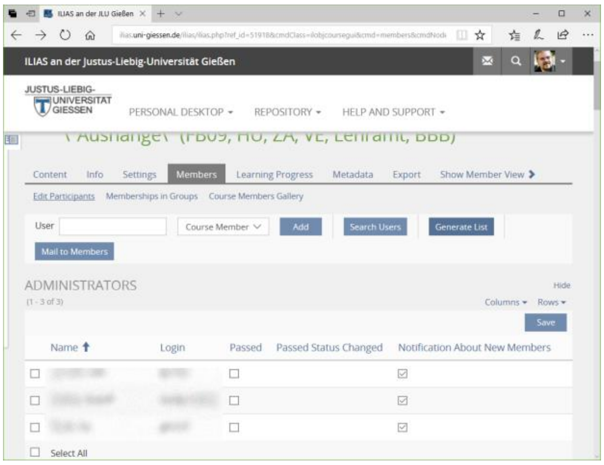
There is an easy way to import user lists from ILIAS into the LMSA Kit, but with the introduction of Microsoft's Edge browser it has become a little bit trickier. Do not use Microsoft's Edge for the next steps, because Edge is not offering the required function of saving pages. Other browser like Internet Explorer, Firefox offer this function and can be used instead.

The advantage of importing users separately from the test data into the LMSA Kit is that the personal data like the e-mail address is also imported. As a result the e-mail address is already available for the LMSA Kit when the generation and delivery of the feedback emerges. Unfortunately, ILIAS does not offer an easy export option for user lists. Instead, we have to use the existing functions in a creative way.

Open the "Members" section of the class in ILIAS (Fig. 43). You can see a list of all current course members. There is no direct export function, but you can generate a printable list. For doing this, click "Generate List". Let all "Detail Data" checked in the next dialog (Fig 44); consequently, ILIAS will export all user data available. Just click on "Print List".

A new browser window or tab will open. It contains the user list (Fig. 46). However, ILIAS marks this page for printing. That is why browsers open immediately the print page dialog. You can just cancel this dialog, if you do not want to print this page. Save this page as html document. The LMSA Kit uses this page saved as data basis for the import of user lists.

After you have started the LMSA Kit and created a new project, you open the users tab (Fig. 45). The adaptive main menu shows the users section. There is the option "Import from other sources". Select "ILIAS User List" as source and import the saved page as user list. Finally, all users of the course are visible in the user list of the project containing the detail data like the e-mail address.



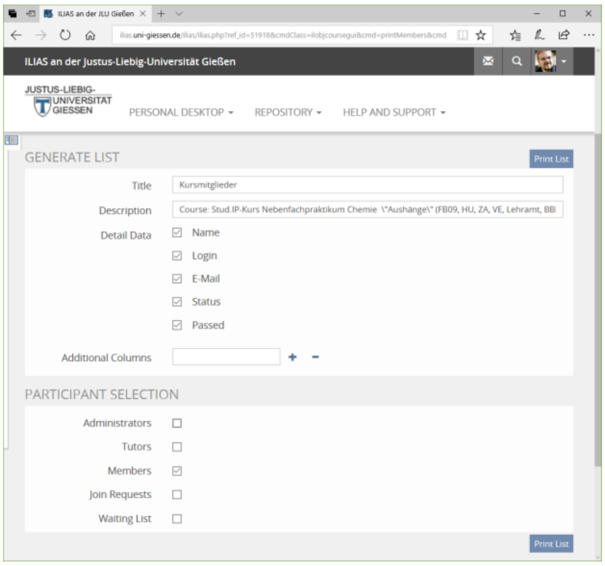
ADMINISTRATORS

(1 - 3 of 3)

	Name	Login	Passed	Passed Status Changed	Notification About New Members
□	[redacted]	[redacted]	□		☑
□	[redacted]	[redacted]	□		☑
□	[redacted]	[redacted]	□		☑

□ Select All

Fig. 43: ILIAS member list



ILIAS an der JLU Gießen
ilias.uni-giessen.de/ilias/ilias.php?ref_id=51918&cmdClass=ilobjcoursegui&cmd=printMembers&cmd
ILIAS an der Justus-Liebig-Universität Gießen
JUSTUS-LIEBIG-UNIVERSITAT GIESSEN
PERSONAL DESKTOP
REPOSITORY
HELP AND SUPPORT
Print List
Print List

GENERATE LIST

Title: Kursmitglieder

Description: Course: Stud.IP-Kurs Nebenfachpraktikum Chemie \"Aushänge\" (FB09, HU, ZA, VE, Lehramt, BB

Detail Data: ☑ Name ☑ Login ☑ E-Mail ☑ Status ☑ Passed

Additional Columns: + −

PARTICIPANT SELECTION

Administrators □

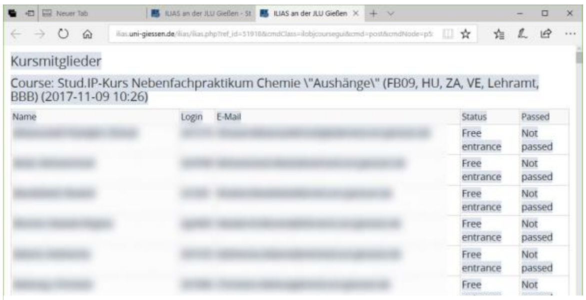
Tutors □

Members ☑

Join Requests □

Waiting List □

Fig. 44: Generate user list dialog



Neuer Tab
ILIAS an der JLU Gießen - St
ILIAS an der JLU Gießen
ilias.uni-giessen.de/ilias/ilias.php?ref_id=51918&cmdClass=ilobjcoursegui&cmd=post&cmdNode=p5

Kursmitglieder

Course: Stud.IP-Kurs Nebenfachpraktikum Chemie \"Aushänge\" (FB09, HU, ZA, VE, Lehramt, BBB) (2017-11-09 10:26)

Name	Login	E-Mail	Status	Passed
[redacted]	[redacted]	[redacted]	Free entrance	Not passed
[redacted]	[redacted]	[redacted]	Free entrance	Not passed
[redacted]	[redacted]	[redacted]	Free entrance	Not passed
[redacted]	[redacted]	[redacted]	Free entrance	Not passed
[redacted]	[redacted]	[redacted]	Free entrance	Not passed
[redacted]	[redacted]	[redacted]	Free	Not

Fig. 45: User list ready for printing (and saving) in ILIAS

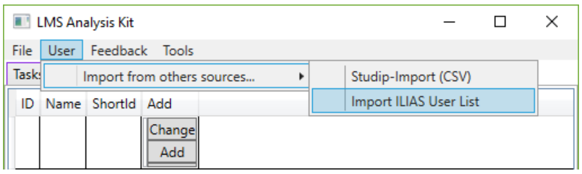


Fig. 46: Import a saved user list