Admins' Manual
v15.5

MULTI-LANGUAGE MANAGEMENT

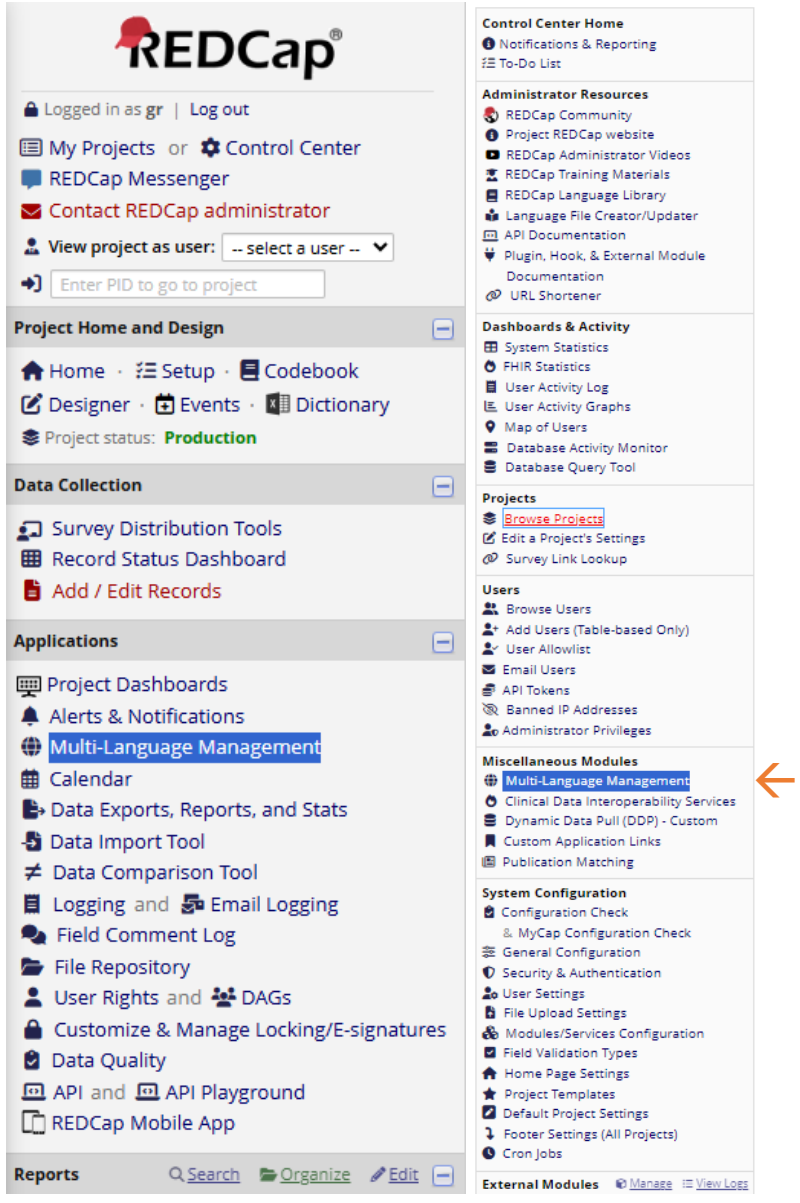
REDCap

This material is kindly contributed by REDCap administrative staff at St. Elisabeth Gruppe GmbH – Klinikum der Ruhr-Universität Bochum

Multi-Language Management (MLM)

- Provides the ability to present REDCap data entry interfaces (surveys, data entry forms) in multiple languages
  - Data Entry Pages, Survey Settings
  - Questions/Choices (Data Dictionary)
  - Missing Data Codes
  - ASIs and Alerts & Notifications
  - PDFs & eConsent
  - MyCap

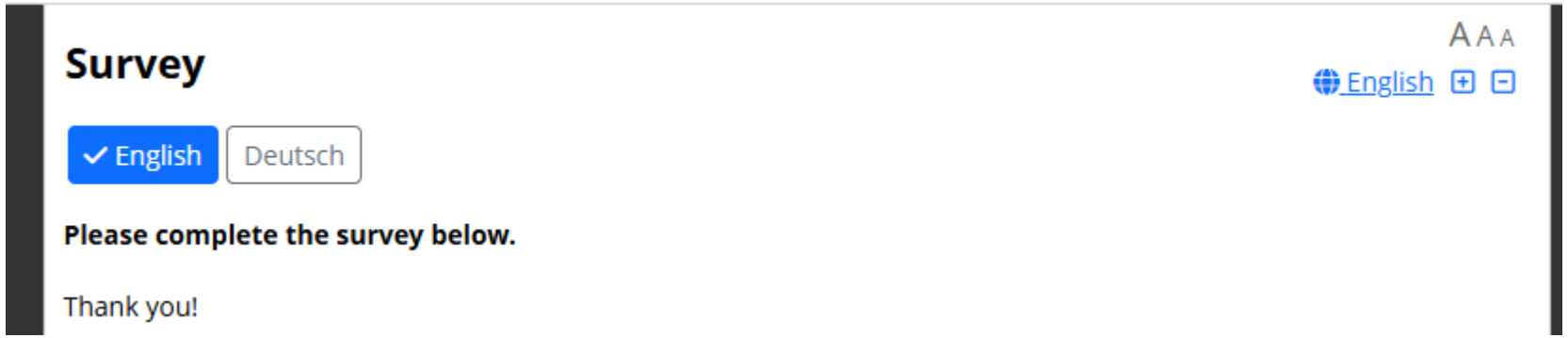
- Compatible with piping, field embedding, branching logic, …



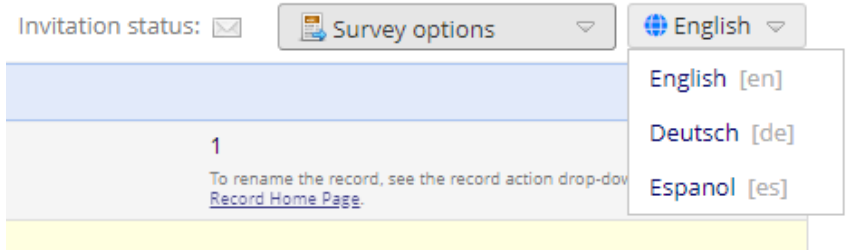
REDCap

How does it look?

- On survey pages, survey respondents will see a language selection menu:
  - Expanded on first visit (left, below survey title)
  - Condensed on later visits (top right, next to the text size controls)
  - Switching between laguages happens in real-time and does not affect data entry



- Data entry users have the ability to switch among languages via the language menu at the top right of a data entry form



REDCap

Basic concepts (1)

- There are **two** primary translation components for each language:

User interface language
i.e., all the strings that are displayed by REDCap in the context of data entry or surveys, the default values of which originate from a Language.ini file (as set in Control Center or Project Settings)

→ Translations can (should!) be provided as **SYSTEM LANGUAGES** by admins (configured in Control Center)

Project-specific language
i.e., all data dictionary items (questions/field labels, choice labels, etc.), survey settings (survey title, survey instructions, etc.), survey queue custom message, ASI and alert message subjects and contents, etc.

Projects can override user interface items (unless disallowed by global MLM settings made by admins)

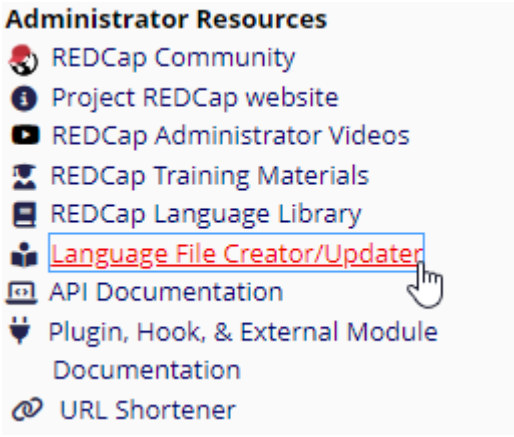
Think of these as the REDCap instance's library of UI translations that projects can use to get started

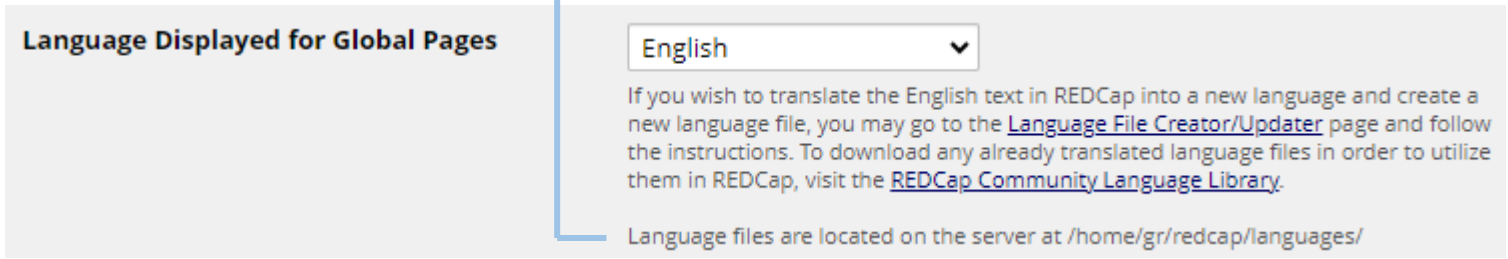
- In projects, everything is **opt-in** (i.e., MLM will only translate what is specifically enabled / provided)!

REDCap

Basic concepts (2)

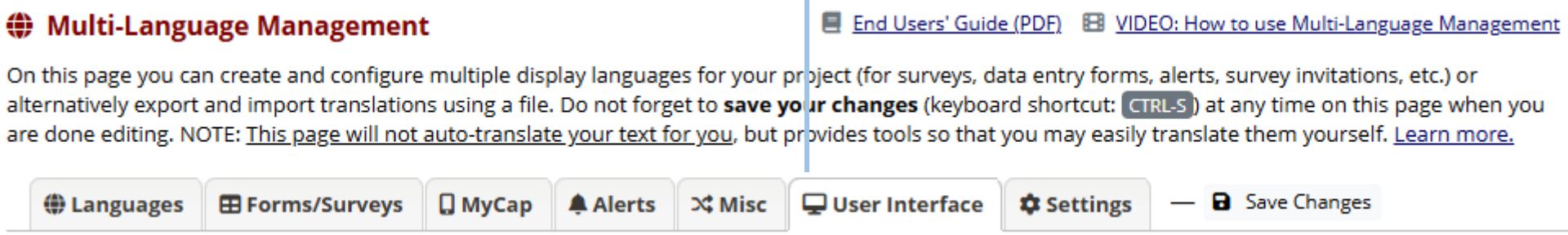
- Do **not confuse** REDCap languages with MLM languages.
  - REDCap language files contain **ALL strings used by REDCap to render its UI**
  - Currently, there are >14,000 unique strings
  - They are packaged in **.ini files** (e.g., English.ini is the default, built-in file)
  - REDCap language files and templates for translation can be generated on the **Language File Creator/Updater** page →
  - The built-in English.ini will always be loaded
  - Language files set in **General Settings** (or **Edit a Project's Settings**) will override the built-in strings
  - Language .ini files must be manually copied to a specific location on the server's file system





Basic concepts (3)

- MLM, in contrast, provides the ability:
  - To **translate the subset** of the REDCap UI strings that make up **Data Entry** and **Survey Respondent-facing** elements* (using the **User Interface** tab)



  - To **translate all the project-specific content** (data dictionary, survey instructions, alerts and ASI email texts, …)
  - For **end users** (data entry personnel, survey respondents), to **choose at will** from the available languages
    - This means that ALL languages that should be available for the user must be added to MLM, including the language the project is set up in!

* Currently, these consist of ~570 strings from the language file

This guide …

… covers setup of MLM and configuration options in two parts:

For

Admins (Control Center)

- Adding and managing **System Languages**
- What non-project pages are supported by MLM
- Global settings and admin-only project-level overrides

For

Project Designers

- Adding and managing languages in a REDCap project
- Translating items
- Project settings
- Import / export of translation
- Snapshots
- Action tags

Screenshots in this guide are from **REDCap 15.5 LTS**

Part 1

CONTROL CENTER SETUP

REDCap

What are “system languages”?

System languages are the languages that are set up via the Multi-Language Management page in the **Control Center**

- They provide translations for the **stock user interface** strings
- They are **the best way** to initialize a new language in a project

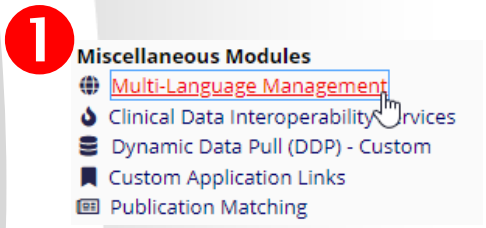
And last but not least:

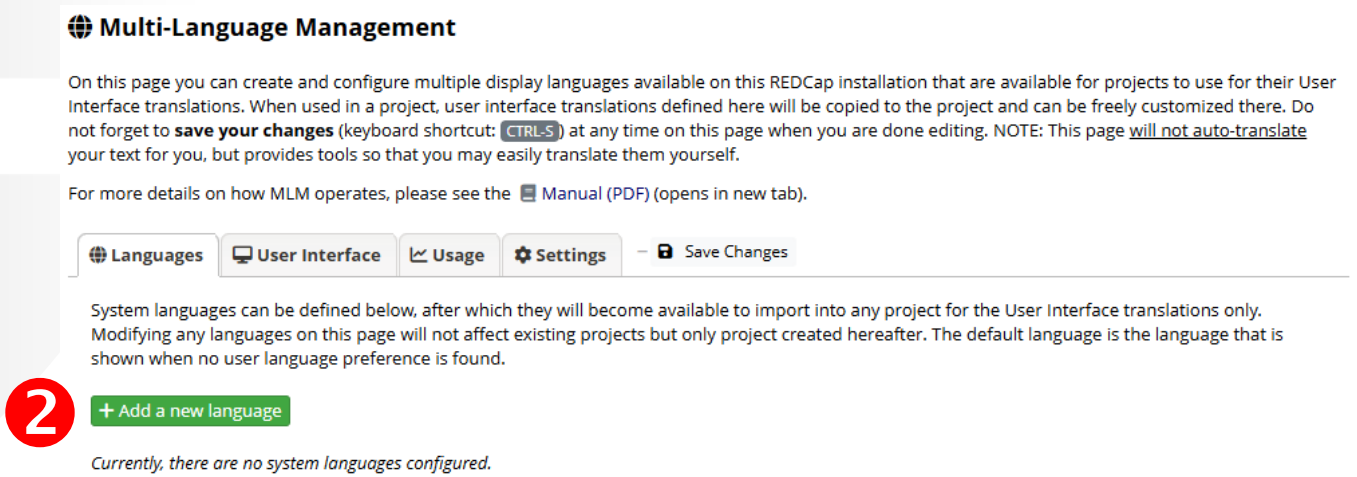
- They provide a language choice for survey respondents on **general pages** (i.e., pages that cannot be linked with a specific project)
  - Generic survey endpoint (access code page)
  - Error messages

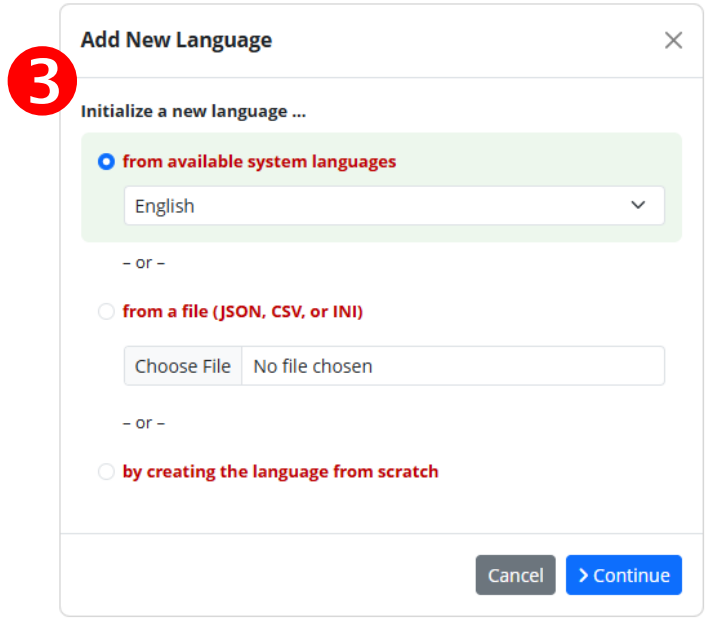
→ Thus, they should ideally constitute a **superset of all** (relevant) **project languages**

Setting up system languages (1)

1. Go to the **Control Center** and bring up the **Multi-Language Management** page
2. Click the **Add a new language** button
3. Choose how to initialize the new language:
   - Based on an already set up language
   - From a file (recommended)
     - MLM JSON or CSV file (e.g., from the MLM Translations Repository)[1]
     - Language.INI (e.g., from the REDCap Consortium Language Library)[2]
   - Or by creating it from scratch





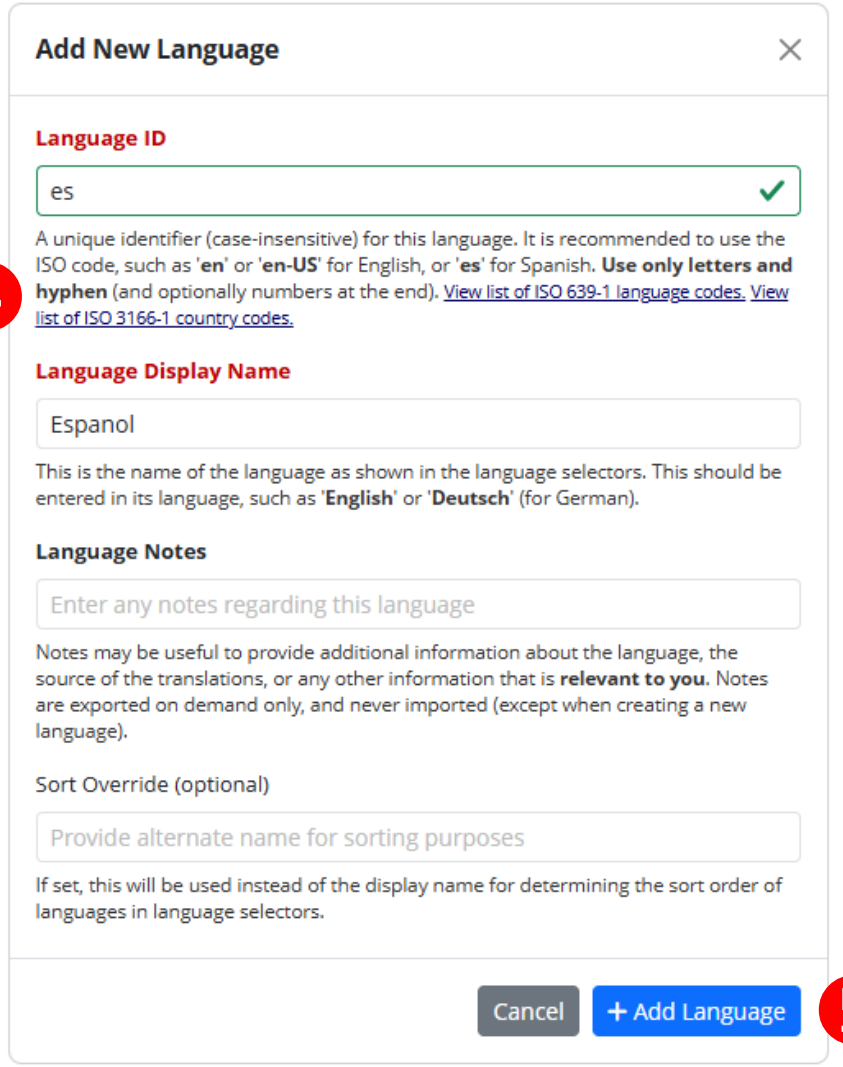


(1) https://redcap.vumc.org/surveys/?__report=7LEEXCTLYA7TJDJ3 (2) https://redcap.vumc.org/surveys/?__report=WDKFTYERTCM8MJ8J

Setting up system languages (2)

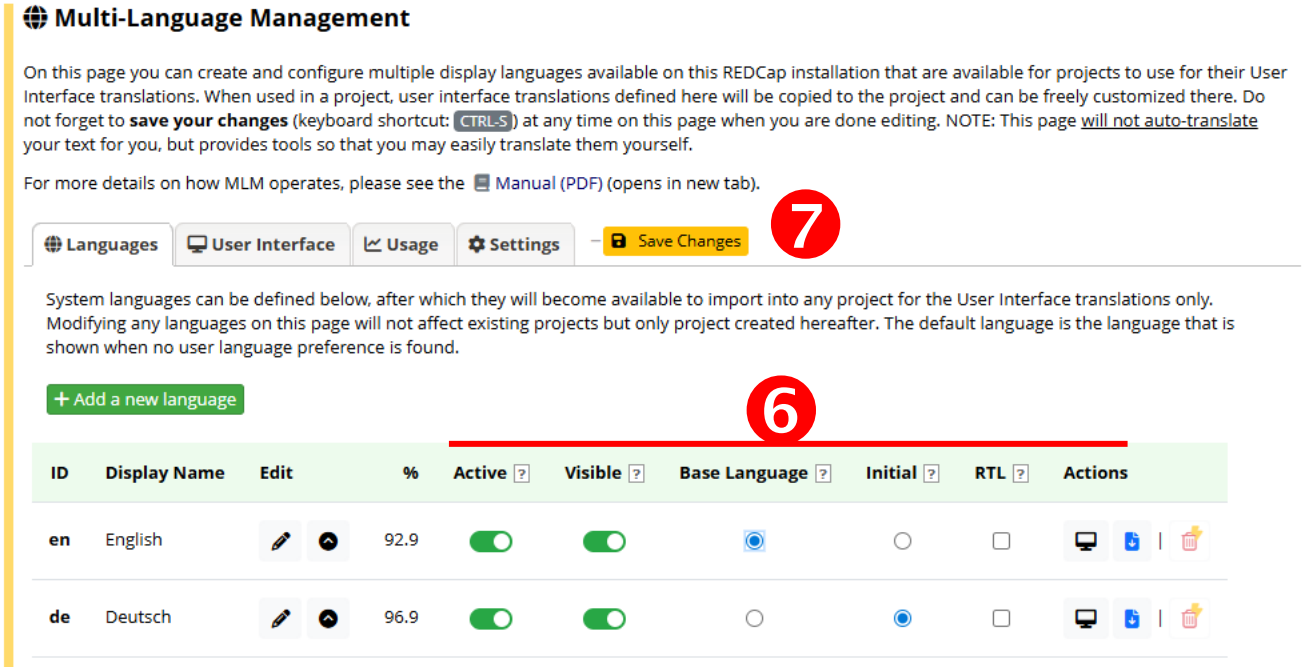
4. Set a language ID and display name
   - Optionally provide some notes
   - Optionally set a sort override (by default, languages are sorted by display name when shown in various places, such as the languages table on the setup page or the language menu on MLM-supported non-project pages)
5. Finally, add the language

- Note:
  - Language IDs should ideally match standard language (and country) codes to facilitate automatic language detection from users' browsers
  - Links to helpful Wikipedia pages are provided



Setting up system languages (3)

6. Once the new language has been added to the table, set whether the language is **Active**,* **Visible**,* the **Base Language**,* the **Initial**,* or a **RTL** (right-to-left) language (such as, e.g., Arabic or Hebrew)
7. Finally, save the changes



* **Active** = The language is available for selection on non-project pages seen by survey respondents

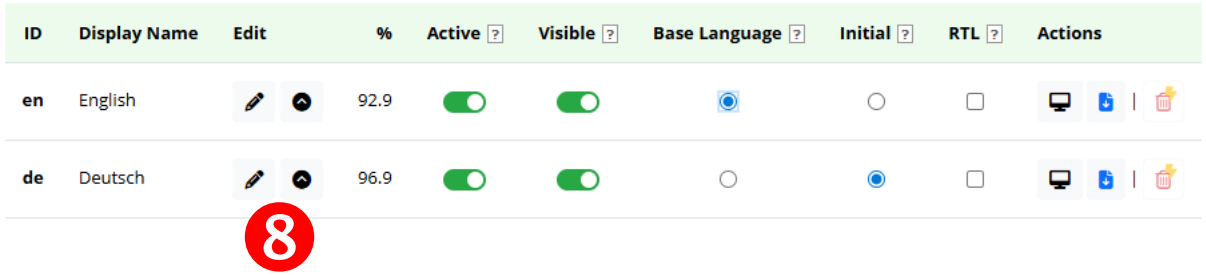
Visible = The language is listed as an available system language when initializing a language in a project

Base Language = This is the language that **corresponds to the language of the Language.ini** file set in *General Configuration*

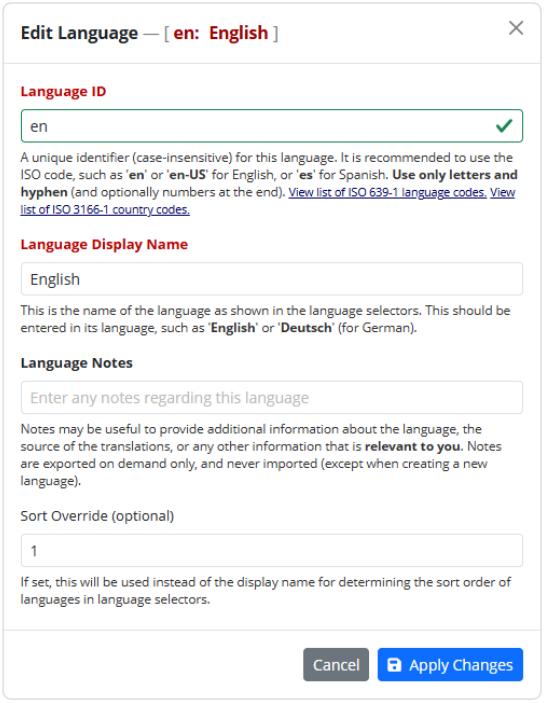
Initial = This is the language that should be displayed initially when visiting a multi-lingual non-project page

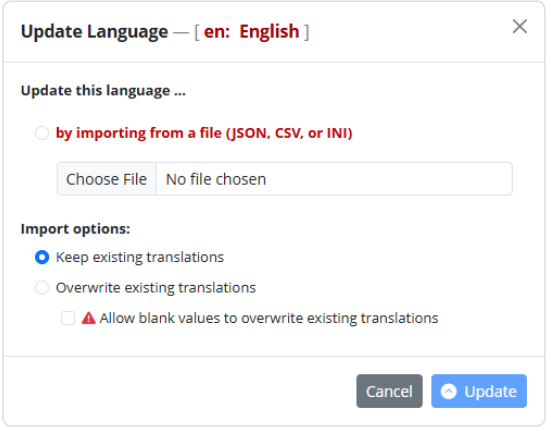
Setting up system languages (4)

8. To edit or update a language, use the edit ( ) and update ( ) buttons



- Use **Edit** to change a language's ID (ISO code) or display name, to view/edit its notes, or the change the sort order
- Use **Update** to update or adding translations by importing them from a file:
  - Use any type of source file as described previously
  - Unless specifically set to overwrite, imports are purely additive, and thus it is safe to import files from a library to get new translations even when custom modifications to existing strings have been made



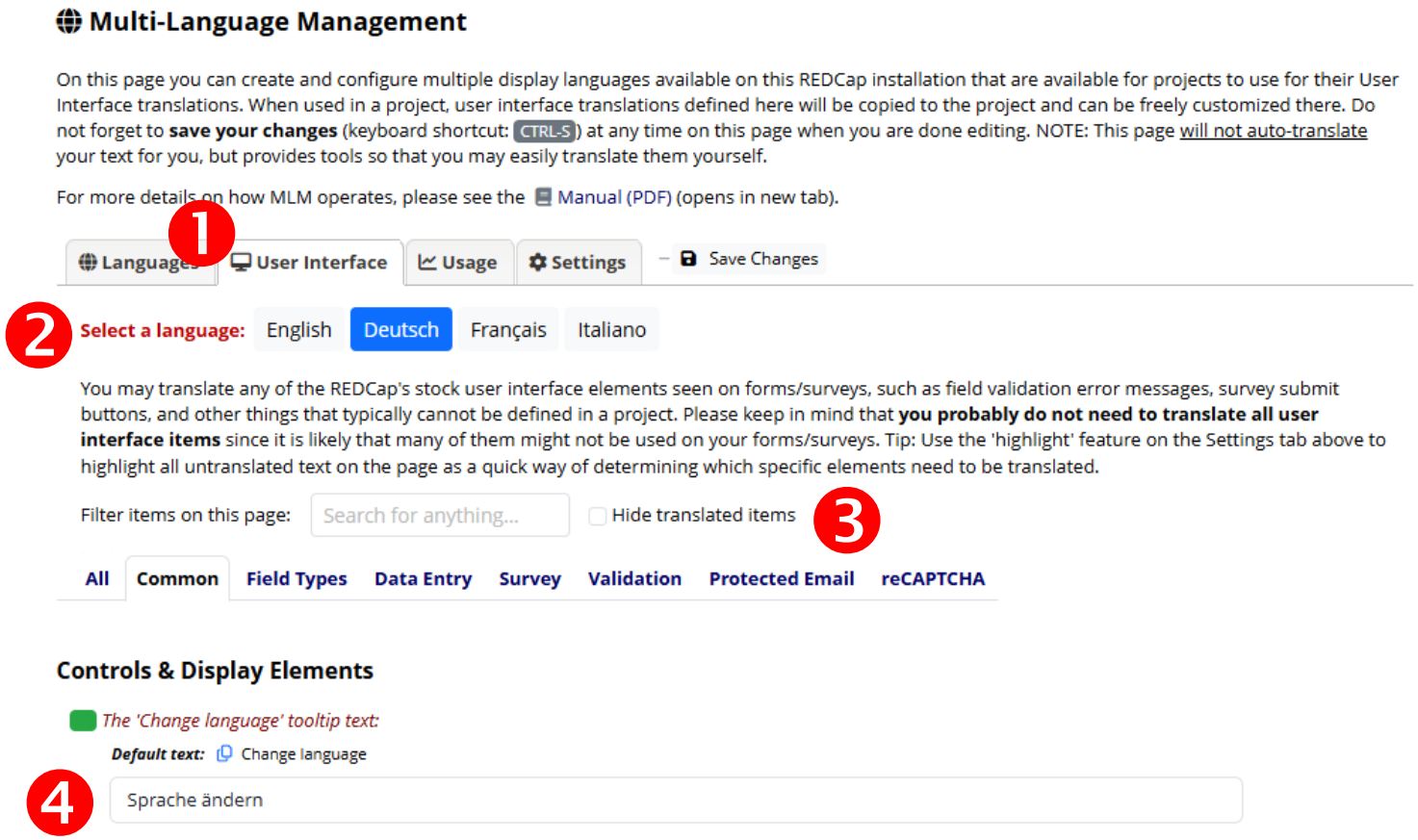


Translating user interface strings

1. Go to the **User Interface** tab
2. Select the language to edit
3. Use sub-tabs and the search field to find the string you want to edit
4. Enter the translation

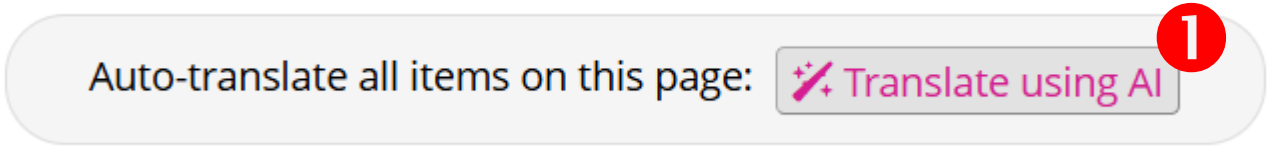
For untranslated strings (empty input fields), the default text will be shown

Tip: If only a few items are missing, check "Hide translated items" and activate the "All" tab



AI-assisted translation

When both, REDCap AI Services and the "Auto-translate text on the MLM setup page" feature are enabled, all MLM setup pages where translations can be entered will contain the "**Auto-translate**" widget:



1. Click the "Translate using AI" button to translate all not yet translated items that are visible on the given page
   - Use the "*Hide translated items*" checkbox and/or the tabs and filters to limit the amount of strings that are sent to the AI service

→Translations will be entered automatically

Notes and limitations:

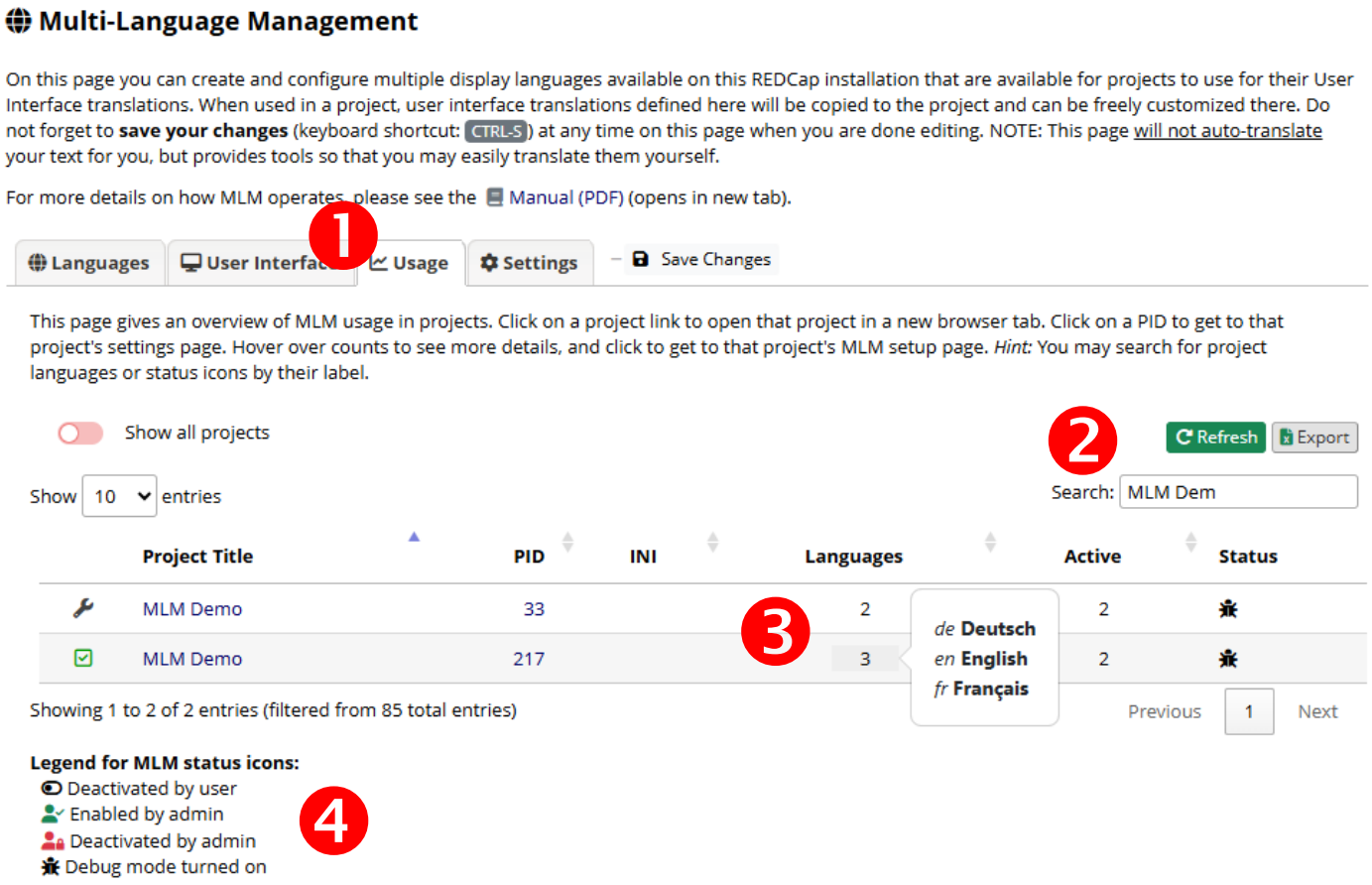
- AI prompts cannot be modified
- Only the default texts are sent to the AI service – no other context
- The AI service gets instructed to translate to the "**display name**" of the language. Thus, it is vital that the display name is set to something meaningful
- Due to this, for the time being, better results can probably be obtained by submitting MLM export files with translation prompts to external AI-assisted translation services, especially when they support structured JSON output
- Always have a person sufficiently proficient in the language check any AI-generated translations!

Usage: Gain insights into projects' use of MLM

1. Go to the **Usage** tab
2. Use search to limit the projects
3. See project stats:
   - **INI**: The name of the Language.ini file used by the project
   - **Languages/Active**: The numbers (and names, when hovering over each number) of total languages set up and active languages, respectively
   - MLM **Status** – see legend

Tip: Project titles link to project home pages, PIDs link to the project settings page

The first column shows the project's status (development, production, …)

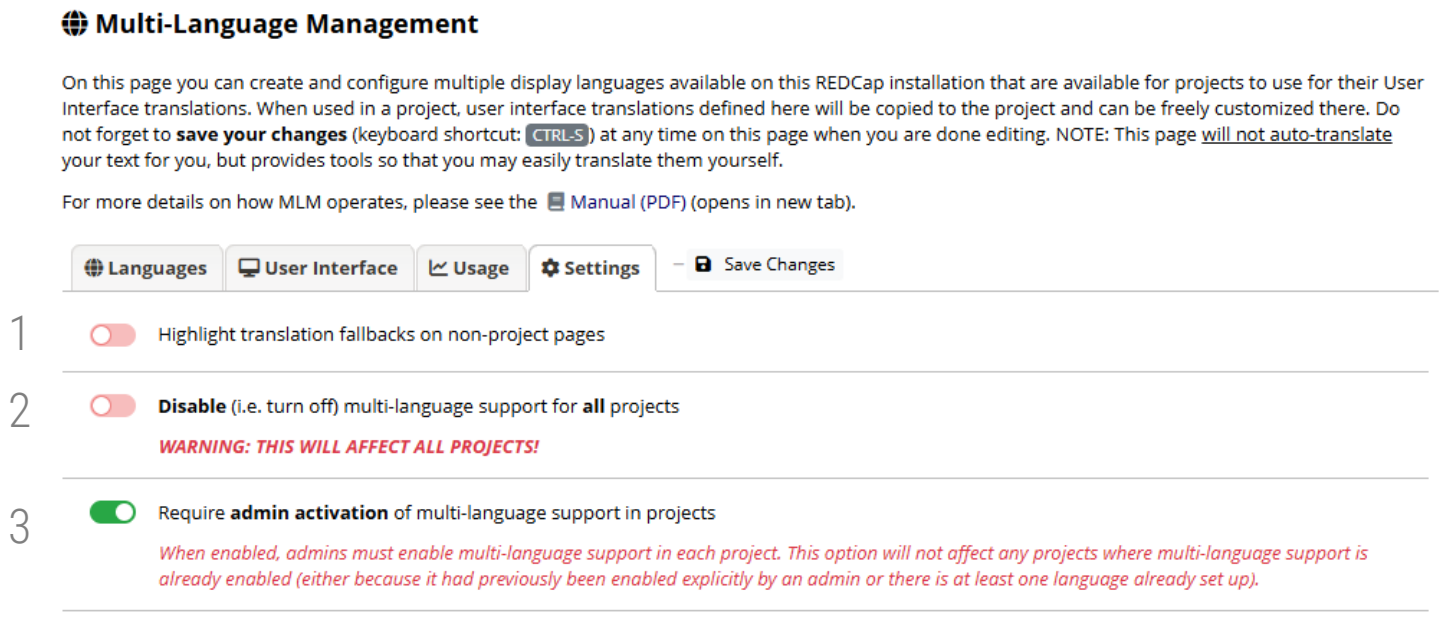


Global settings (1)

1. **Highlight** translation fallbacks. When enabled, items not translated to the currently active language will be highlighted on non-project pages that support translation, such as, e.g., the survey access page

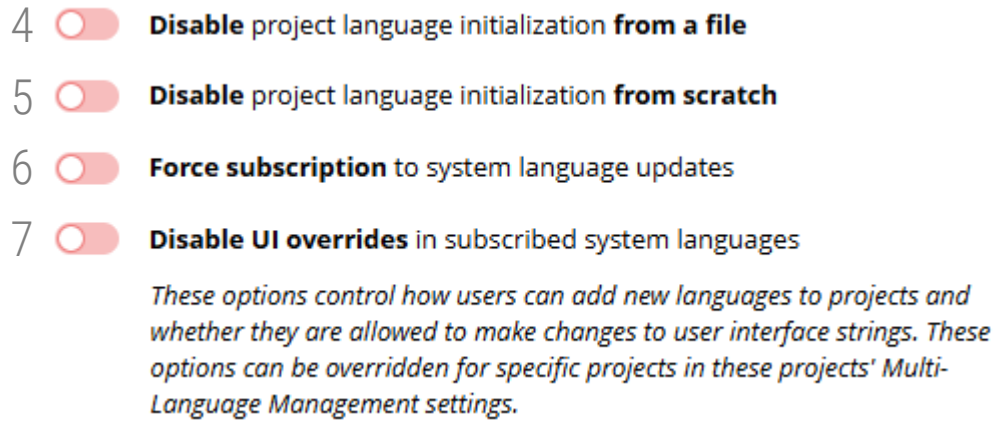
2. **Disable** Multi-Language Management globally. When switched on,
   - MLM will be disabled entirely (in all projects and on all non-project pages)
   - the MLM link will be removed from projects and the MLM setup page in projects will be inaccessible (even for admins)

3. **Admin activation**, when switched on, will require an admin to activated MLM in each project on that project's MLM setup page

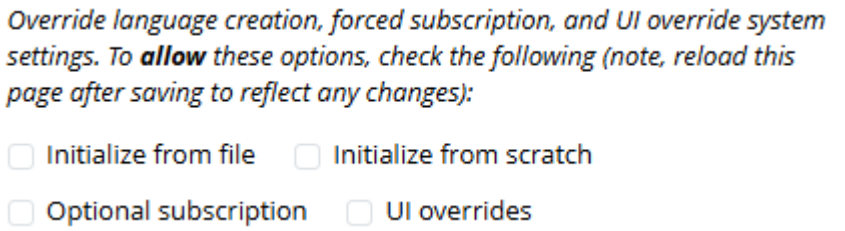


Global settings (2)

4. **Disable** project language initialization **from a file** removes the option to set up new languages by importing from files
5. **Disable** project language initialization **from scratch** removes the option to add new project languages from scratch
6. **Force subscription** to system language updates, when enabled, will ensure that project languages based on a system language will receive updates to user interface strings automatically
7. **Disable UI overrides** in subscribed system languages, when enabled, will prevent users from editing and thereby overriding user interface items



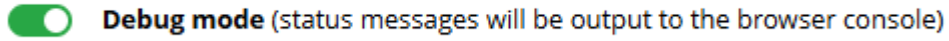
Note: These settings can be overridden in individual projects:



REDCap

Global settings (3)

8. **Debug mode**, when enabled, will cause result in rich diagnostic output to the browser console on on supported non-project pages



   To enable diagnostic output on project pages, **Debug mode** must be turned on in individual projects

Note:

- Debug mode *should not be turned on in production environments* unless needed for debugging purposes
- Only admins can turn on/off Debug mode in projects

REDCap

Best practices I

- As admin, decide on a support model for MLM, e.g.,
  - Turn off MLM server-wide
  - No specific support – do not add (visible) system languages unless you are prepared to maintain them
  - Provide some starting languages (“System Languages”), as opt-in for projects – do maintain these languages!
  - Tightly manage the languages available on your REDCap instance and disallow projects to create their own languages
  - Keep in mind that some pages, like the survey access page, have no project context and thus providing multiple language for this is the admin’s responsibility.
- Then, set up MLM accordingly

Best practices II – System languages (1)

MLM-provided system languages ..

- are languages that are configured through the **Multi-Language Management** setup page in **Control Center**, which can be made available for projects
- provide translations for the **stock data entry / survey UI** strings covered by MLM
- when provided (by admins), are **the best way** to initialize a language inside a project (Make sure to provide your system's base language as a 100% full translation, too!)
- provide a language choice for survey respondents on **general pages** (i.e. pages that cannot be linked with a specific project)
  - Generic survey endpoint (access code page)
  - Error messages

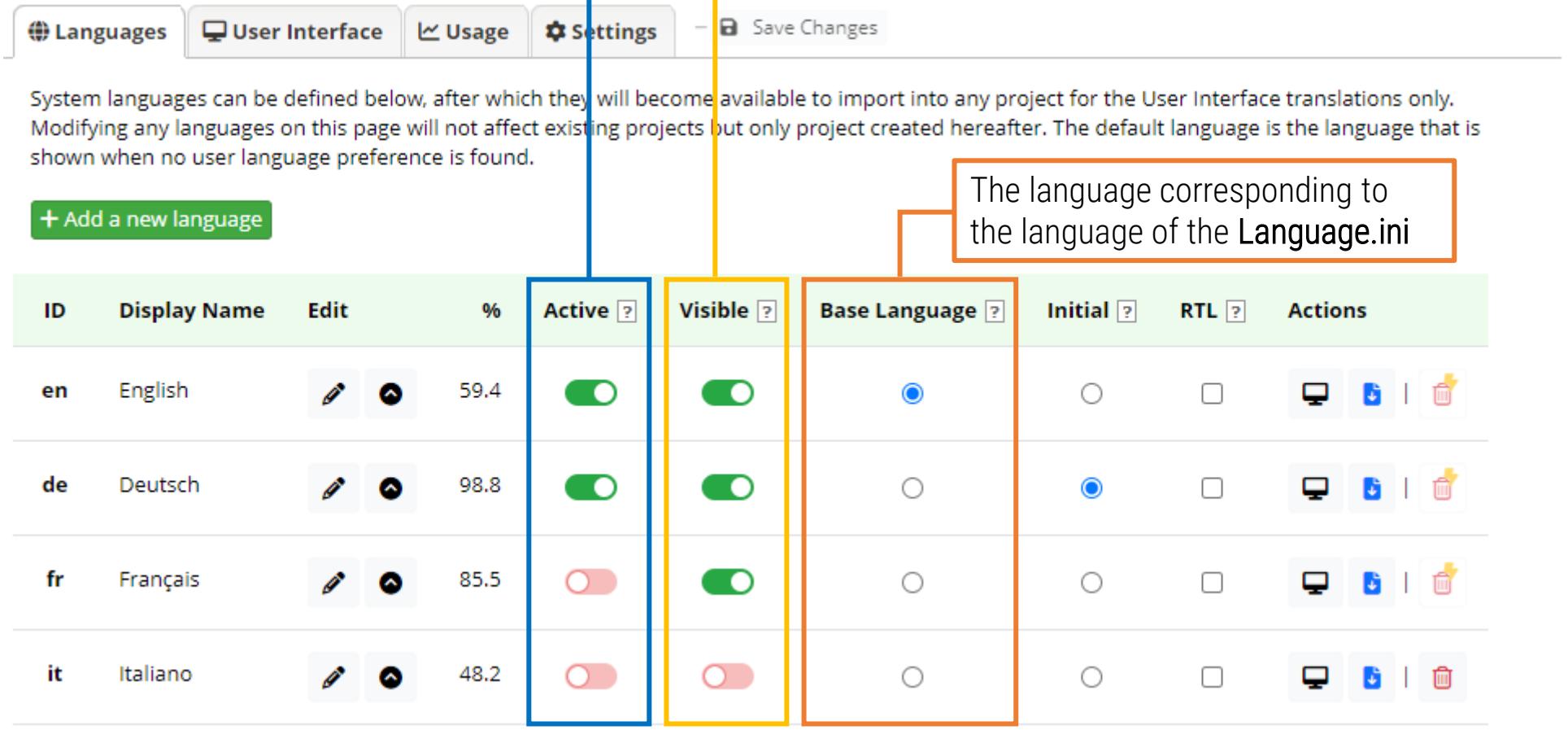
→ Thus, the languages configured in CC MLM *ideally* constitute a **superset** of all languages used in projects

Best practices II – System languages (2)

Active = Available in the language switcher on non-project pages

Visible = Available as starter languages when setting up MLM in projects

The language corresponding to the language of the **Language.ini**



REDCap

Part 2

USING MLM IN PROJECTS

Getting started with MLM in projects

Depending on the REDCap instance's settings, there may be up to **three** methods for initializing languages within projects:

1. Initializing from the available SYSTEM LANGUAGES (preferred)
2. Importing from an existing file (.csv, .json, .ini)
3. Building from scratch (not recommended)

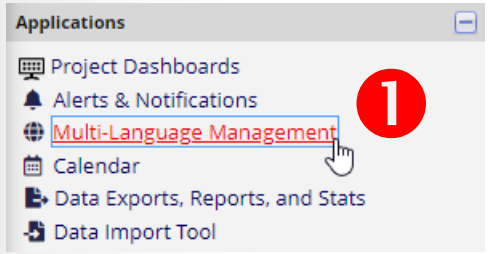
Then, data dictionary (and other items) can be translated

- On the project's MLM setup page
- Offline (by a professional translator):
  1. Export the items to be translated
  2. Import the translations

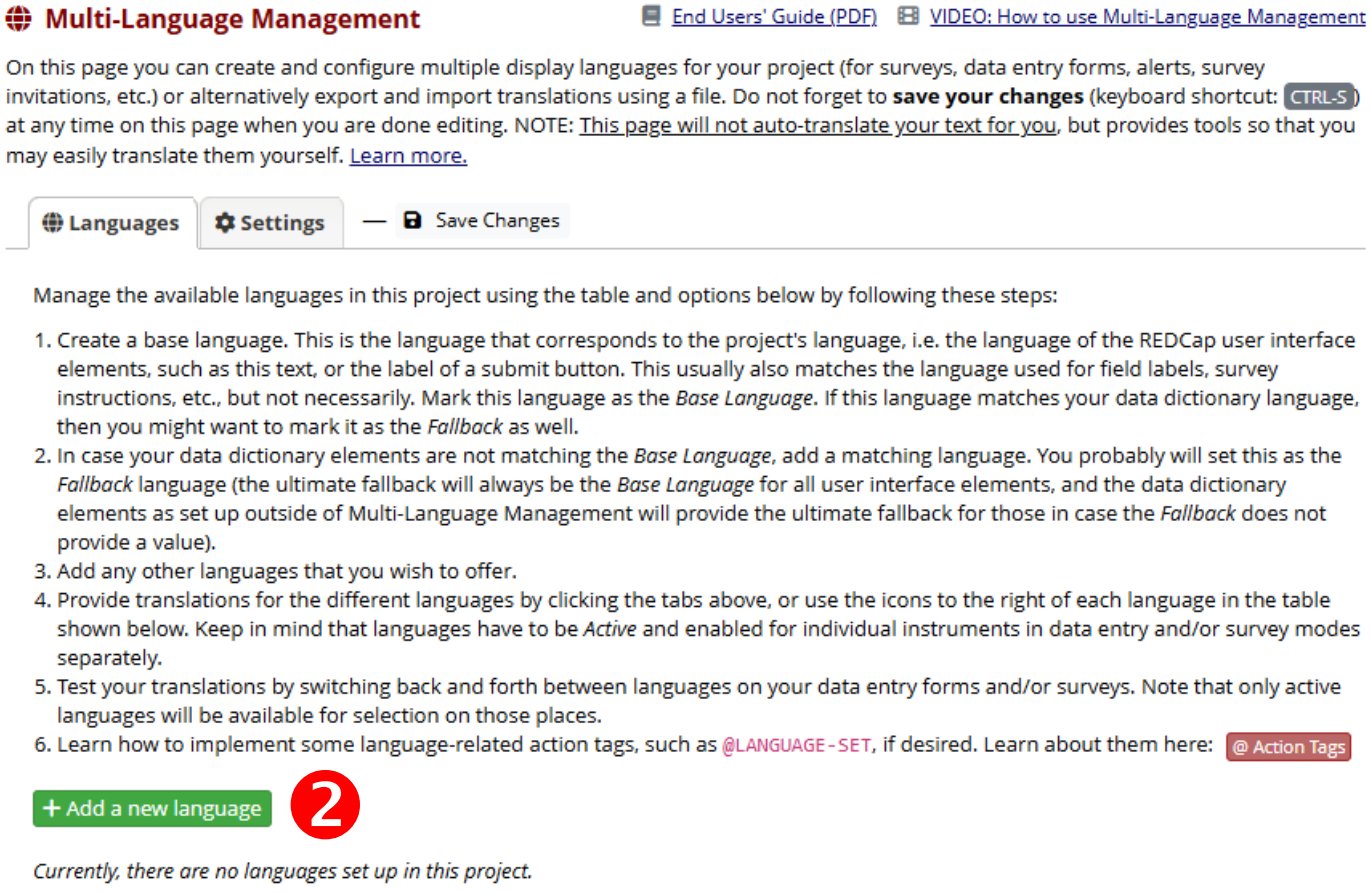
Note:
The MLM setup page is locked in **production** projects. Switch to draft mode first to unlock it. Furthermore, only one user may edit MLM settings at a time.

Adding languages to projects (1)

1. Bring up the **Multi-Language Management** page by clicking the link in the **Applications** menu
2. Click the **Add a new language** button

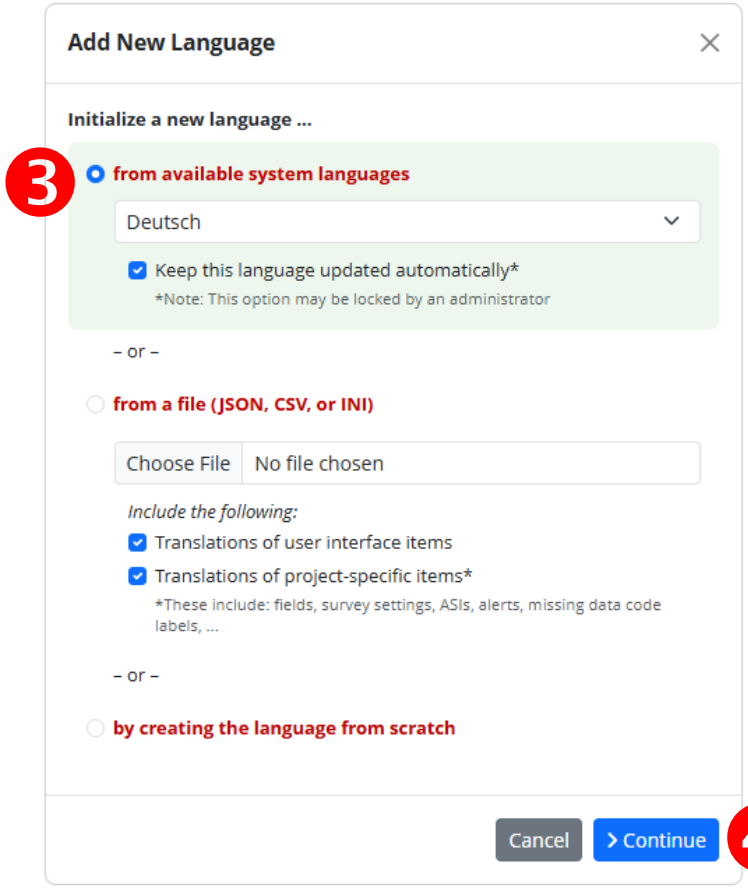


Note: **Project Design** user rights are required to be able to access the MLM setup page (this may change in future versions)



Adding languages to projects (2)

3. Choose one of the option to initialize a new language (depending on the availability of system languages and restrictions imposed by the REDCap admins, not all options may be available*):
   - Initialize **from available system languages** with or (if not enforced) without automatic updates *(when available, this is usually the preferred way)*
   - Initialize **from a file** using only UI strings, project-specific items, or both
   - Initialize **by creating the language from scratch**
4. Click **Continue** to proceed to the next step



* On some installations, admins may, e.g., restrict project languages to a set of centrally maintained languages. In this case, initialization from system languages would be the only available option.

Adding languages to projects (3)

5. Finally, enter (or change if not already set) the *ID** and the *Display Name***. Optionally, a *Sort Override* can be specified

6. Click the **Add Language** button to add the language to the project

   Further options to control (and update) languages are available on the languages table

* The language ID should ideally correspond to the ISO 639-1 code for that language, optionally followed by an ISO 3166-1 country code. This will facilitate automatic detection of browser language for survey respondents.
When **MyCap** is enabled, a matching ID to one of the languages supported in MyCap is essential

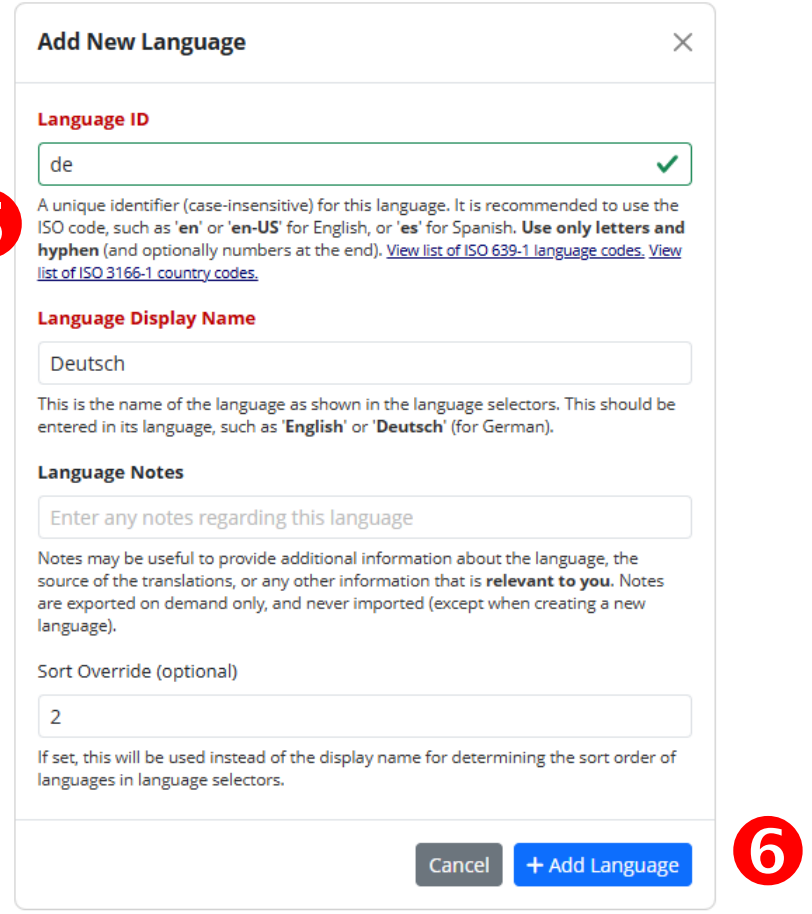
** When AI Services are enabled, the display name and language ID will govern AI-assisted translation

Note: Please ensure you are using a Language ID from the list if you wish to utilize a specific language in the MyCap app. [Hide List]

You may click on the Language ID from the list below to populate in the Language ID and Language Display Name.

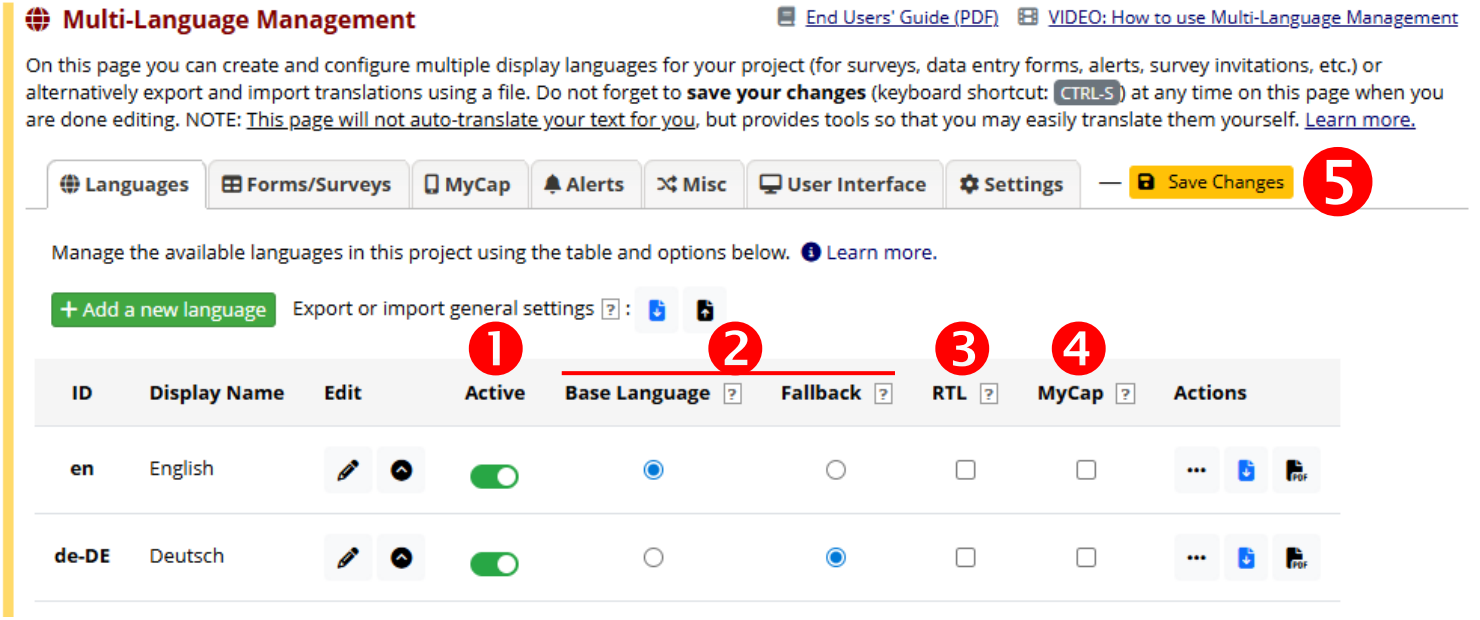
ID	Display Name
en-US	(American) English
it-IT	Italian
zh-CN	Simplified Chinese
hi-IN	Hindi
es-ES	Spanish
ar-SA	Arabic
fr-FR	French
bn-BD	Bengali
pt-BR	Brazilian Portuguese

ID	Display Name
ur-PK	Urdu
ja-JP	Japanese
pa-IN	Punjabi
de-DE	German
ko-KR	Korean
fil-PH	Tagalog (Filipino)
vi-VN	Vietnamese
ht-HT	Haitian Creole
uk-UA	Ukrainian



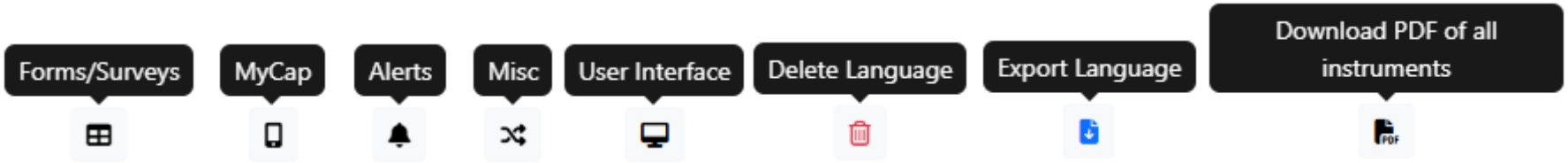
Managing languages in projects (1)

1. Turn on/off languages by setting their **Active** state
2. Set a language to represent the **Base Language** or to serve as **Fallback** (this will be explained in detail later)
3. Specify, whether a it's a right-to-left (**RTL**) language
4. Enable languages for use in the **MyCap** app (check that language IDs match MyCap-supported languages). This option is only available when MyCap is in use
5. Don't forget to **save** your changes (an orange save button/ orange strip signals that there are unsaved changes)

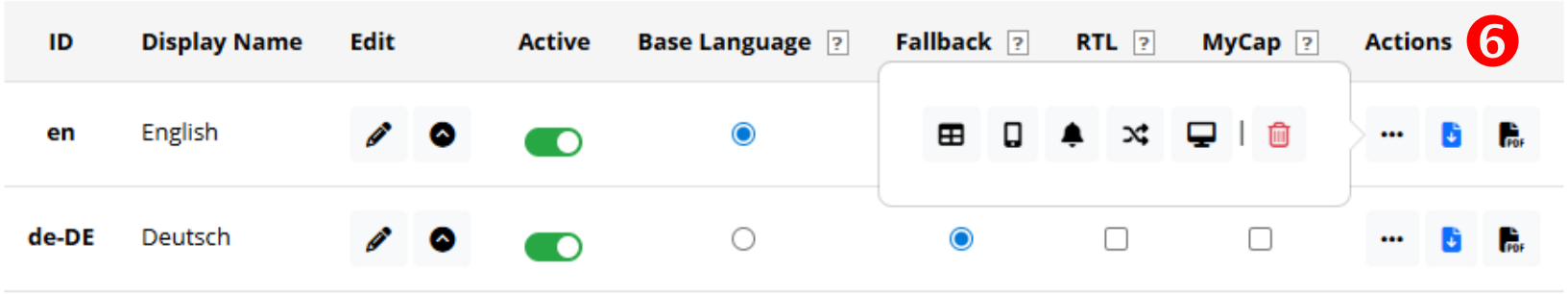


Managing languages in projects (2)

6. Use the **Action** buttons to go to editing subsection or to perform various operations on a language:



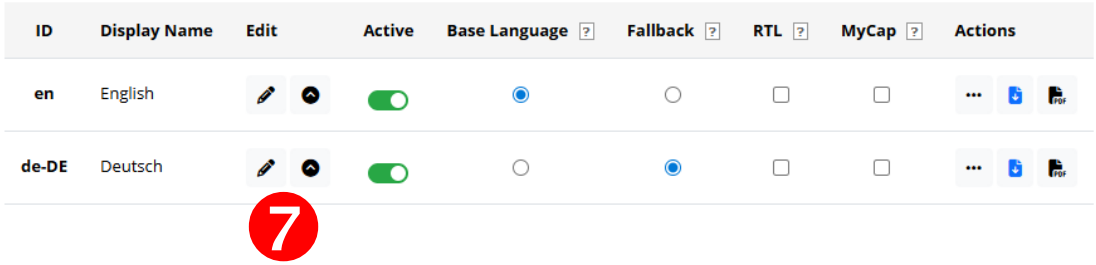
- Get to the edit screens for **Forms/Surveys**, **MyCap** (when enabled in the project), **Alerts**, **Misc** (this includes, e.g, Missing Data Codes and Protected Email Mode), and **User Interface** (this also indicates subscription and linkage status of languages to system languages: black – no linkage, red – based upon a system language, red with flash – subscribed)
- **Delete** a language
- **Export** a language (see section on exporting and importing)
- Download a translated **PDF** of all instruments



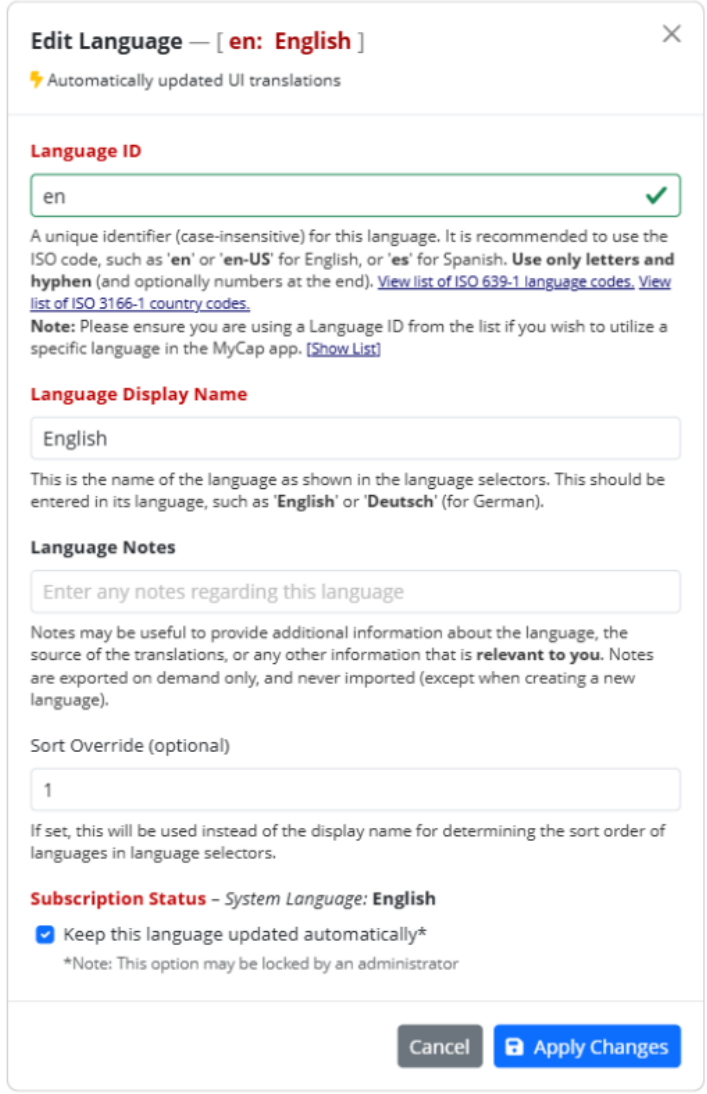
REDCap

Managing languages in projects (3)

7. To edit or update a language, use the edit ( ) and update ( ) buttons

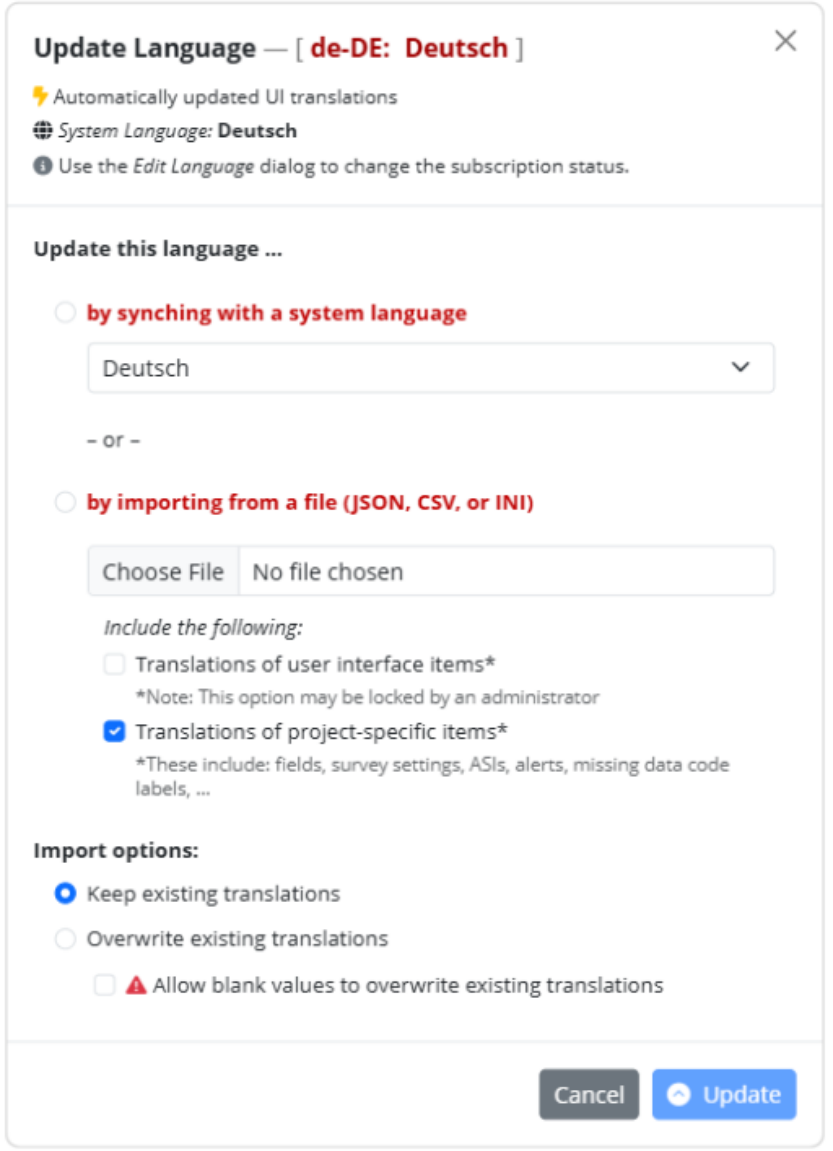


- Use **Edit**
  - To change a language's ID (ISO code) or display name
  - To view/edit its notes
  - To change the sort order
  - To control its subscription status (only available when the language was initialized from a system language); this item may be locked depending on system settings;
    the current subscription status is indicated at the top



Managing languages in projects (4)

- Use **Update** to update or adding translations by importing them from a file:
  - Use any type of source file to import User Interface translations
  - Use a previous translation export from this project that has had translations added or edited
  - Unless specifically set to overwrite, imports are purely additive, and thus it is safe to import partial files, e.g., exports for a single instrument only, or files from a library with User Interface translations

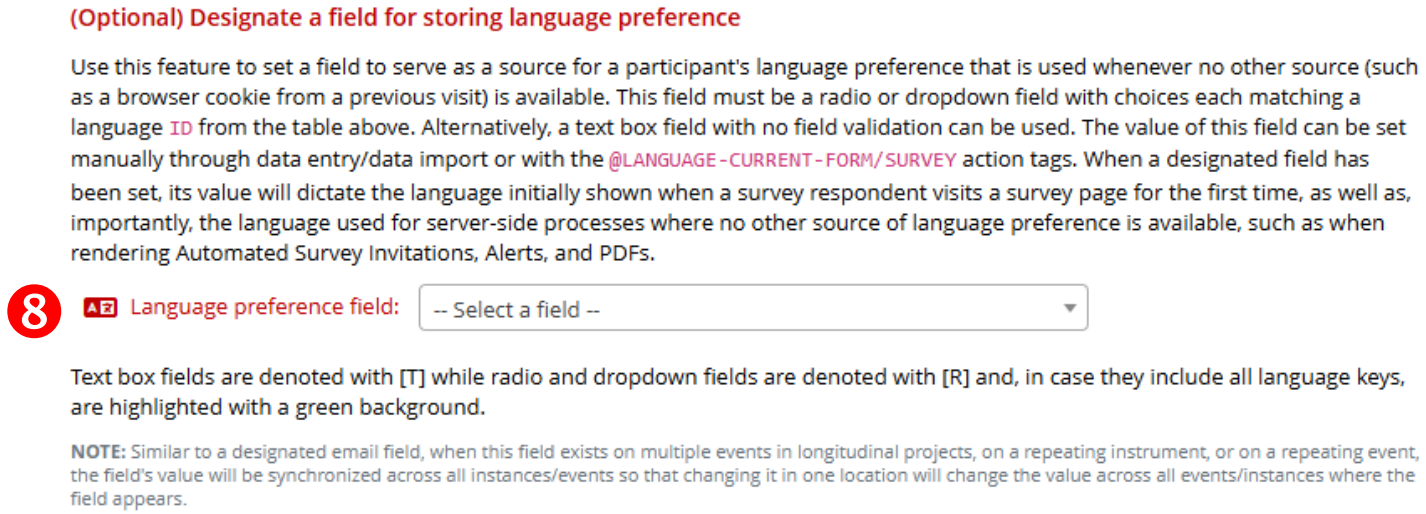


Managing languages in projects (5)

8. Optional: Designate a **Language preference field**.

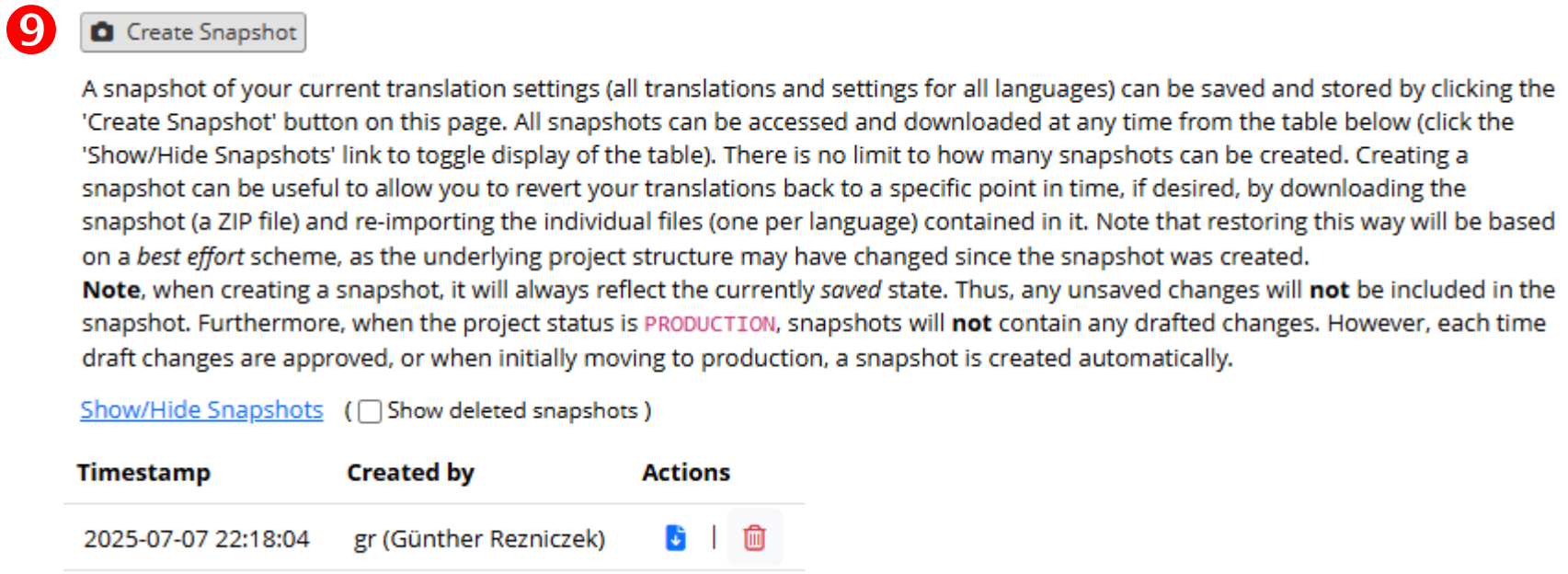
   For certain operations and under some circumstances, REDCap cannot deduce the appropriate language to use. These are essentially all server-side processes that are decoupled from direct user interaction (and availability of a browser cookie), such Automated Survey Invitations or Alerts and PDFs that are rendered with a delay. In such cases, the value of the language preference field, if available, will be used to set the language

   Note: This field behaves very similar to the *designated email field*, i.e. it will be synchronized across all events / repeat instances

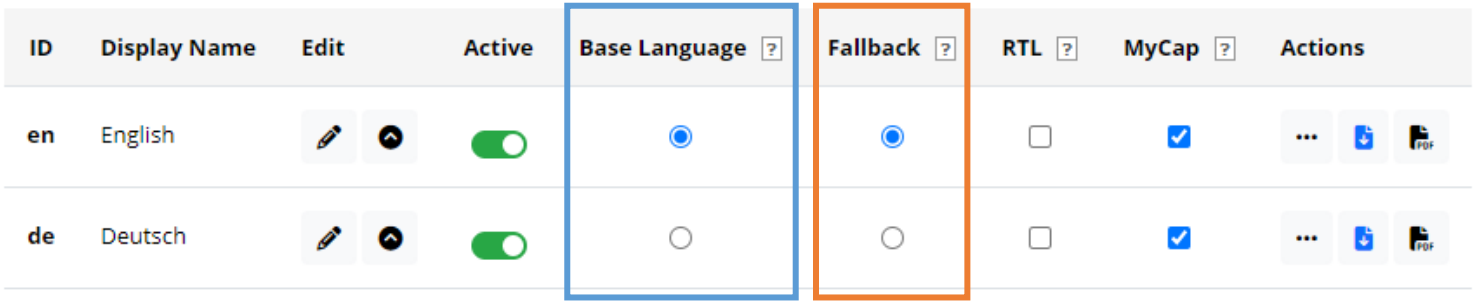


Managing languages in projects (6)

9. Use the **Create Snapshot** facility to create backups of all languages in a project
   - Snapshots can be created on demand during the development phase of a project
   - When a project is moved to PRODUCTION and each time drafted changes are commited, a snapshot is made automatically
   - Snapshots are ZIP archives of all possible exports – their individual files can be imported back
   - Snapshots can be deleted



Understanding Base Language and Fallback



ID	Display Name	Edit	Active	Base Language	Fallback	RTL	MyCap	Actions
en	English			◉	◉	☐	☑	
de	Deutsch			○	○	☐	☑	

The Base Language …

- should **match** the project language (as set on the *Edit Project's Settings* page)
- usually *(but not necessarily)* is the language used during project design, i.e., the language used for all data dictionary items, survey settings, ASI/alert messages, missing data code labels, …

If this is not matching the project UI language, then a corresponding language should be added and set as "Base Language" (but it can be set as inactive)

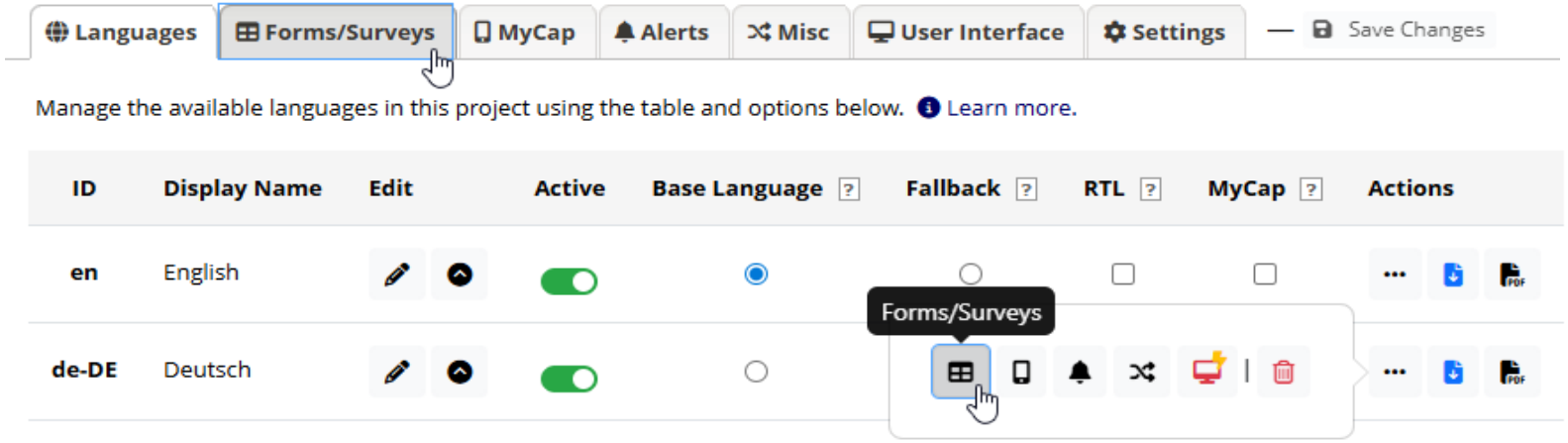
The Fallback language …

- will be used when a translation is not available in the language that currently **should** be displayed. In case the fallback translation is missing, too, the project's base language will be used (i.e. whatever is in the data dictionary or, in case of UI strings, the Language.ini file configured for the project)

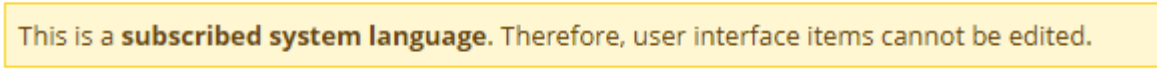
REDCap

Translating project items

1. Click on a tab or the corresponding icon in the languages table



- Forms / Surveys (fields, survey settings, ASIs, survey queue customizations)
- MyCap (MyCap-related items; only shown when MyCap is enabled in the project)
- Alerts
- Misc (Missing Data Codes, PDF customizations, Protected Email customizations)
- User Interface – UI strings are usually set from a system language, but there can be project-specific overrides, unless such overrides have been disallowed by a REDCap admin:

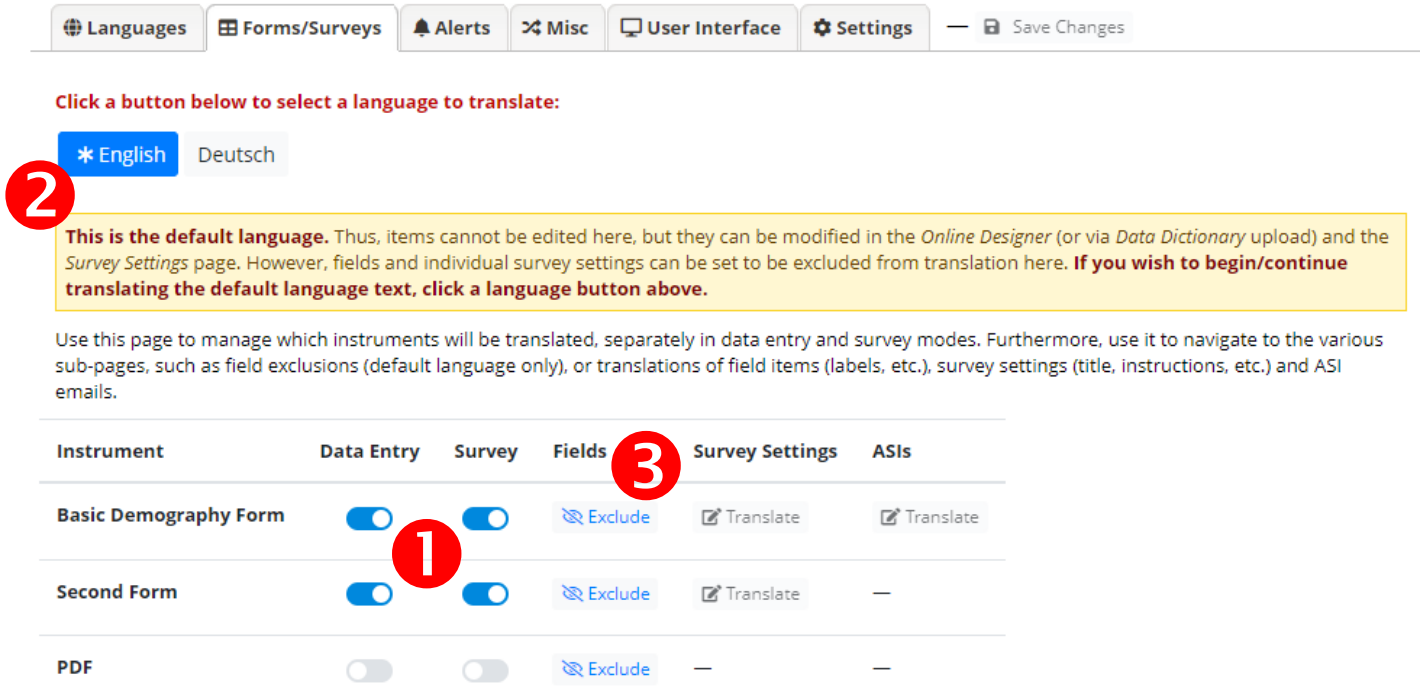


REDCap

Forms/Surveys (base language)

1. On each language's tab, control whether this languages is available for selection, separately for **data entry forms** and **surveys**
2. The **base language** tab (indicated by an asterisk) is special
3. There is nothing to translate, but fields can be excluded from translation (i.e., they won't show up on the other languages' tabs)

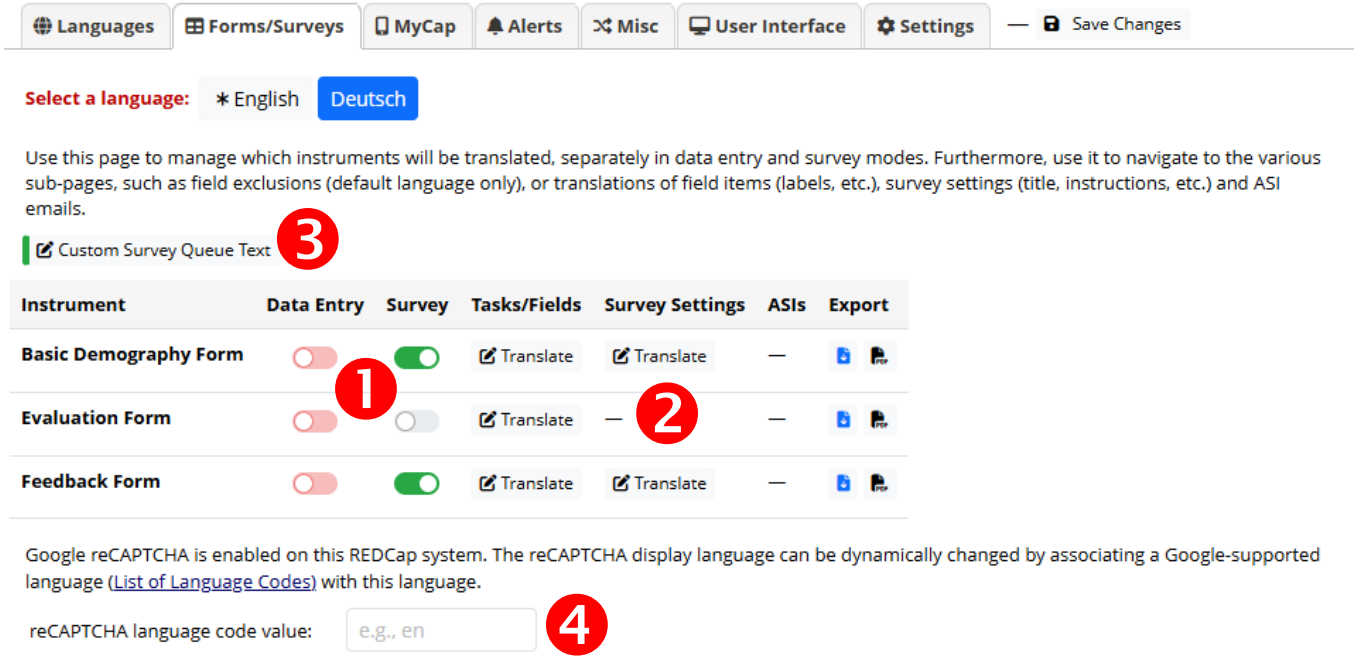
Note, a common pitfall is to forget to enable languages for individual instruments (separately for data entry forms and surveys)!



Forms/Surveys (other languages)

1. On each language's tab, control whether this languages is available for selection, separately for data entry forms and surveys
2. Depending on whether an instrument is enabled as a survey, **Translate** buttons are available for Fields (and Tasks, if MyCap is enabled), Survey Settings, and ASIs
3. When the Survey Queue is enabled with custom texts, these can be translated here
4. For Google reCAPTCHA-enabled surveys, each language should be linked to a Google-supported language code

Note, a common pitfall is to forget to enable languages for individual instruments (separately for data entry forms and surveys)!

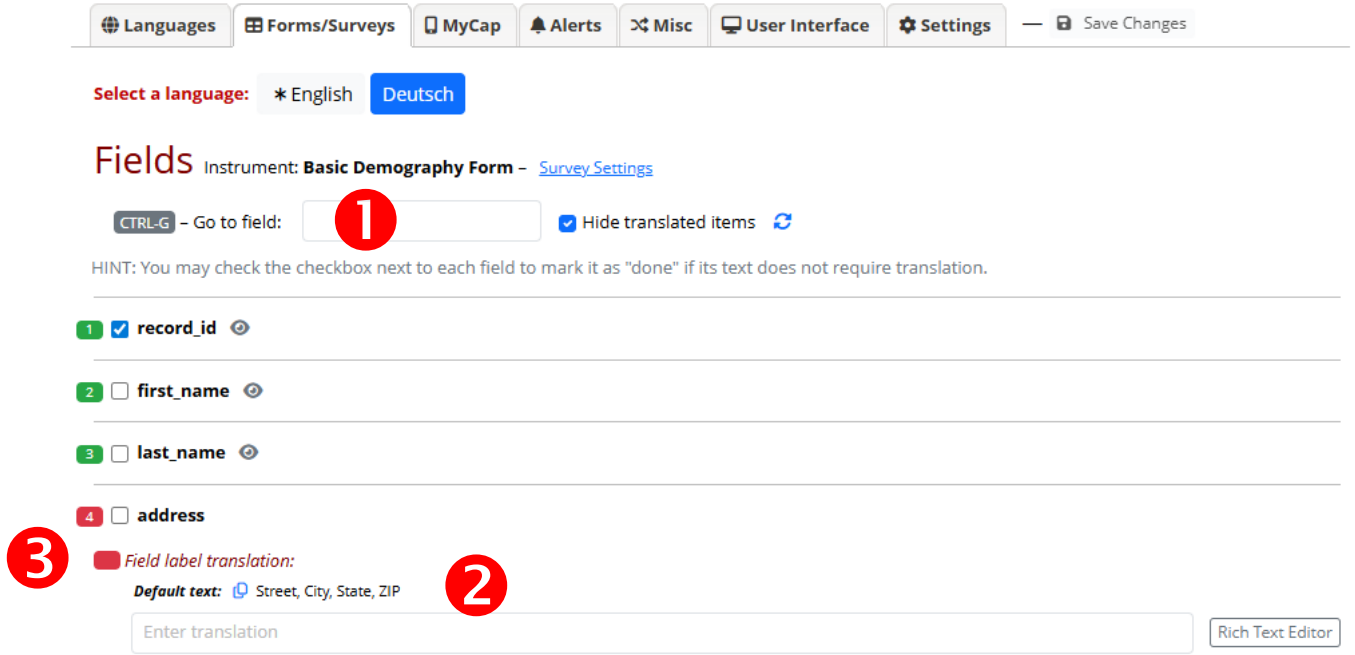


Forms/Surveys – Fields

1. Use the **Go to field** dropdown to quickly navigate to a specific field
2. Enter translations. When appropriate, a **Rich Text Editor** button is available
3. Indicators reflect the translation status of items/subitems (green = complete)

Hints:

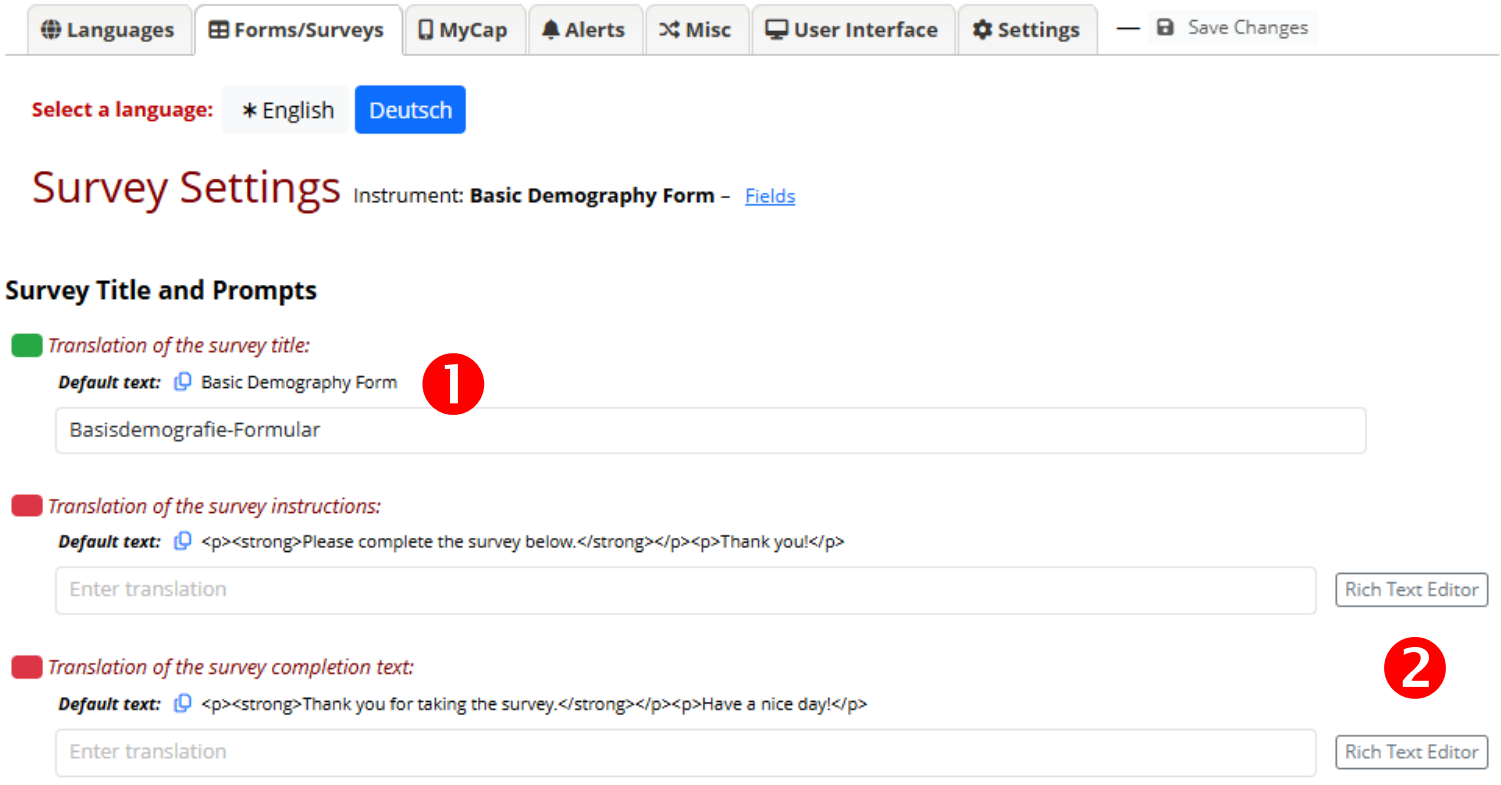
- The copy icons next to the default text can be used to quickly copy/paste them to translation tools
- In case an item changes after having been translated, this will be indicated by an orange marker



Forms/Surveys – Survey settings

1. Translate each item
2. Use the **Rich Text Editor** for any settings supporting HTML

Note: There is nothing to translate for the base language (marked with an asterisk). When clicked, this will take you to the field exclusion page)

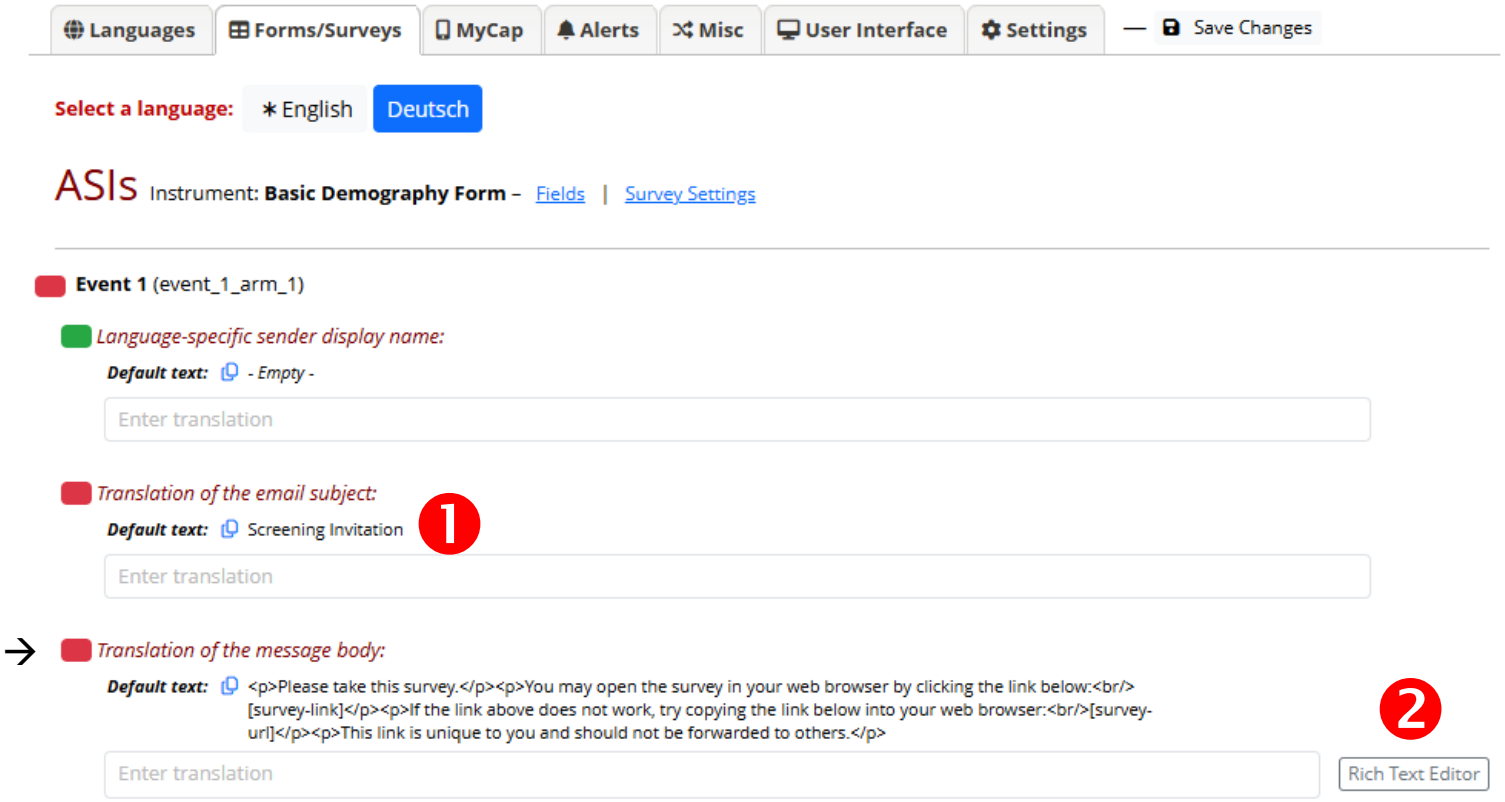


Forms/Surveys – ASIs

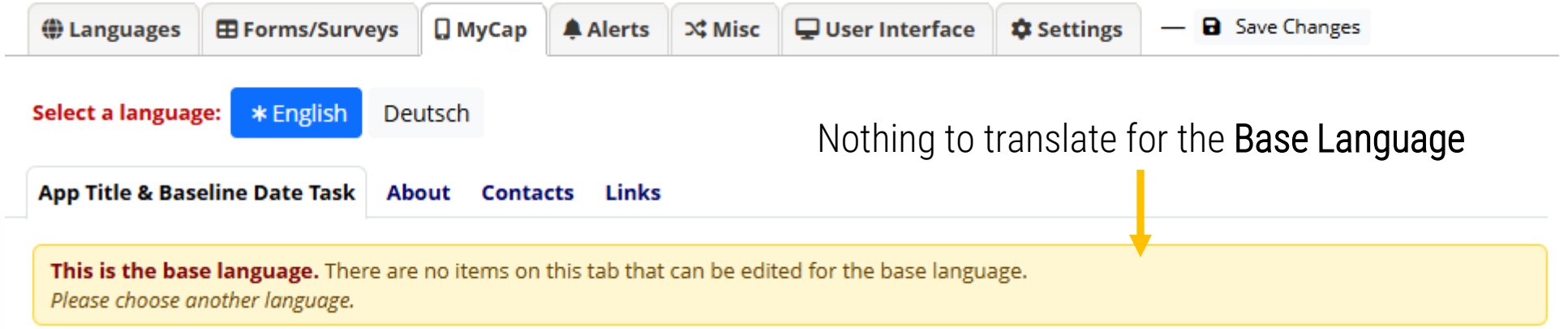
1. Translate each item
2. Use the **Rich Text Editor** for any settings supporting HTML

Note, ASIs for each event need to be translated separately →

Make sure to faithfully replicate any piping in the translations! →



MyCap settings (1)



Nothing to translate for the **Base Language**

For all other languages:

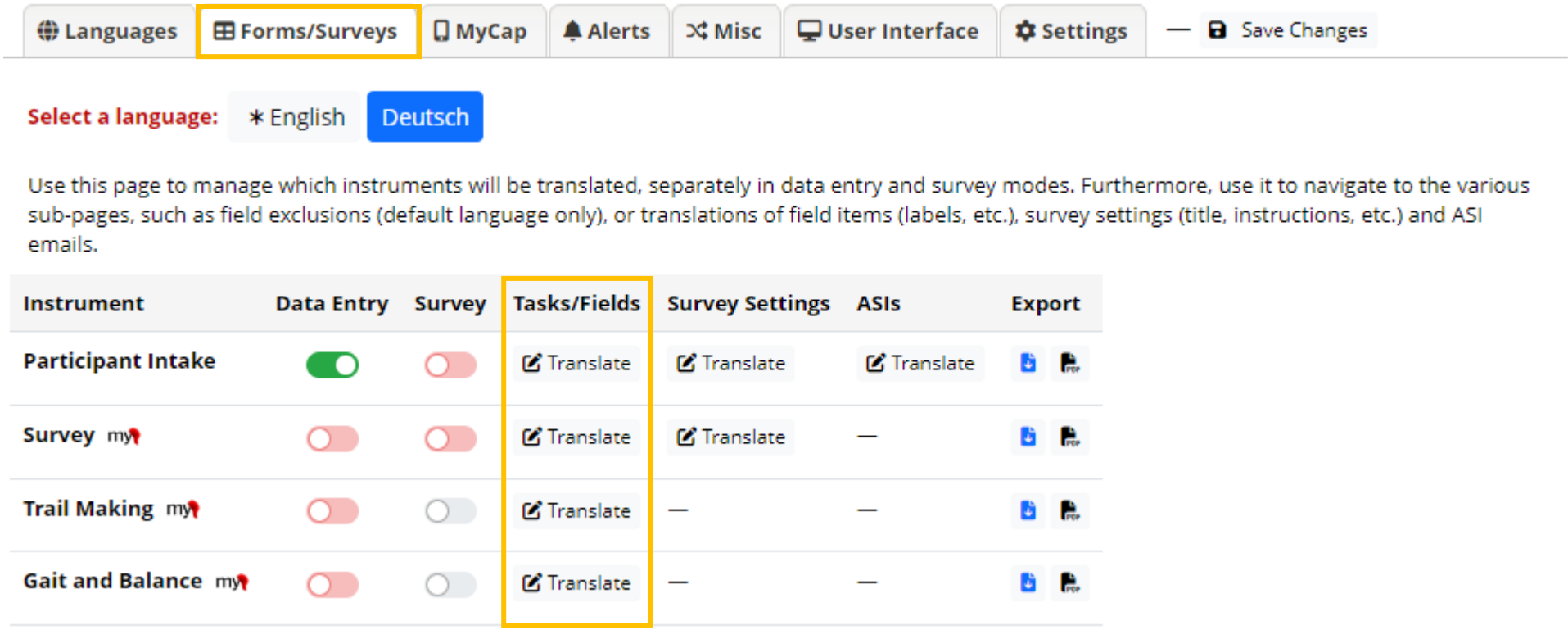
MyCap App Settings

- App title & Baseline Date Task settings
- About pages
- Contacts
- Links

REDCap

MyCap settings (2)

MyCap Tasks are translated here

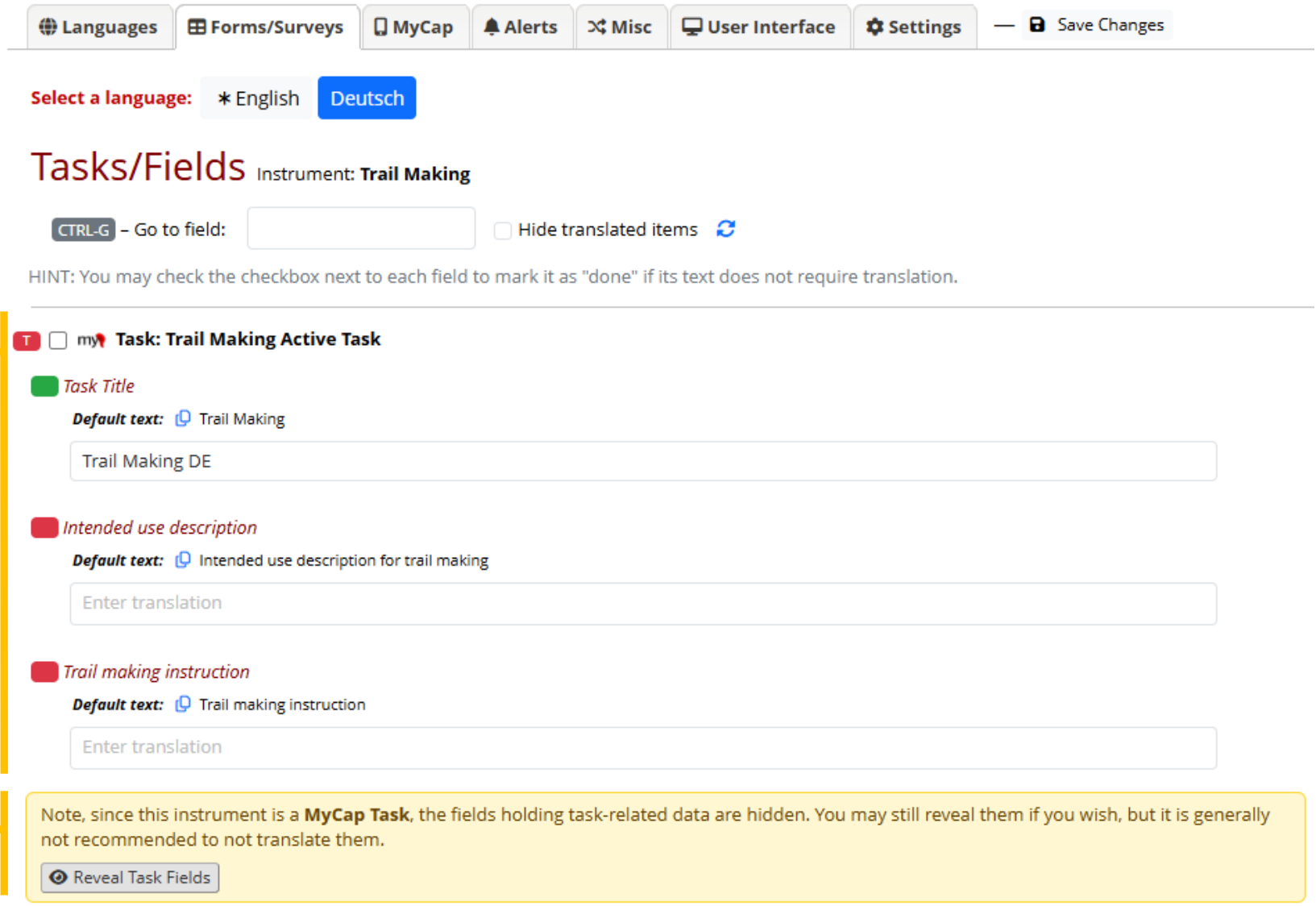


REDCap

MyCap settings (3)

Task-specific items are treated like regular fields

Task fields are hidden (@MC-TASK-...) (as these normally do not need to be translated)

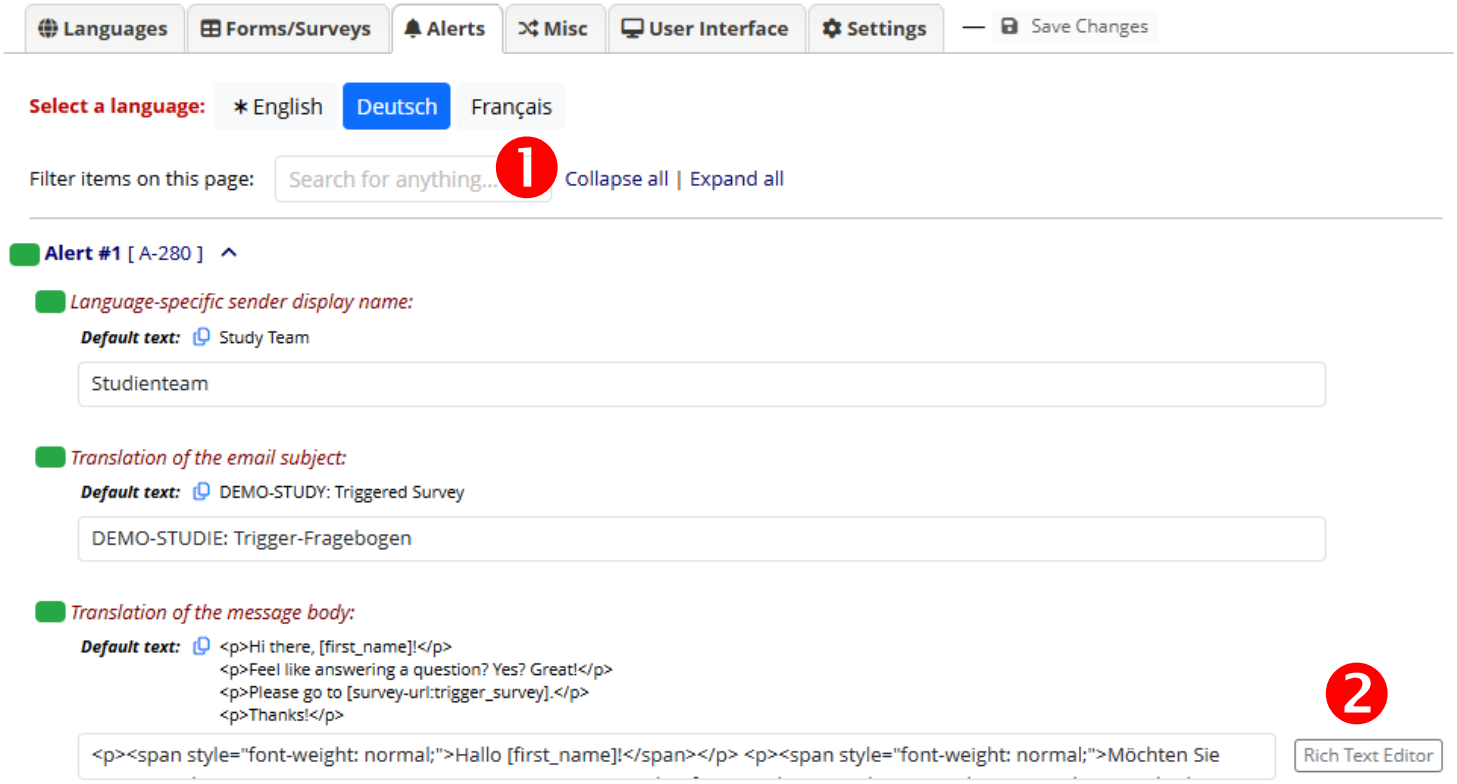


Alerts

1. Use the search box to quickly find the item you would like to translate / edit
2. Use the **Rich Text Editor** for any settings supporting HTML

For Alerts, there are important settings to be made on the base language's page! →

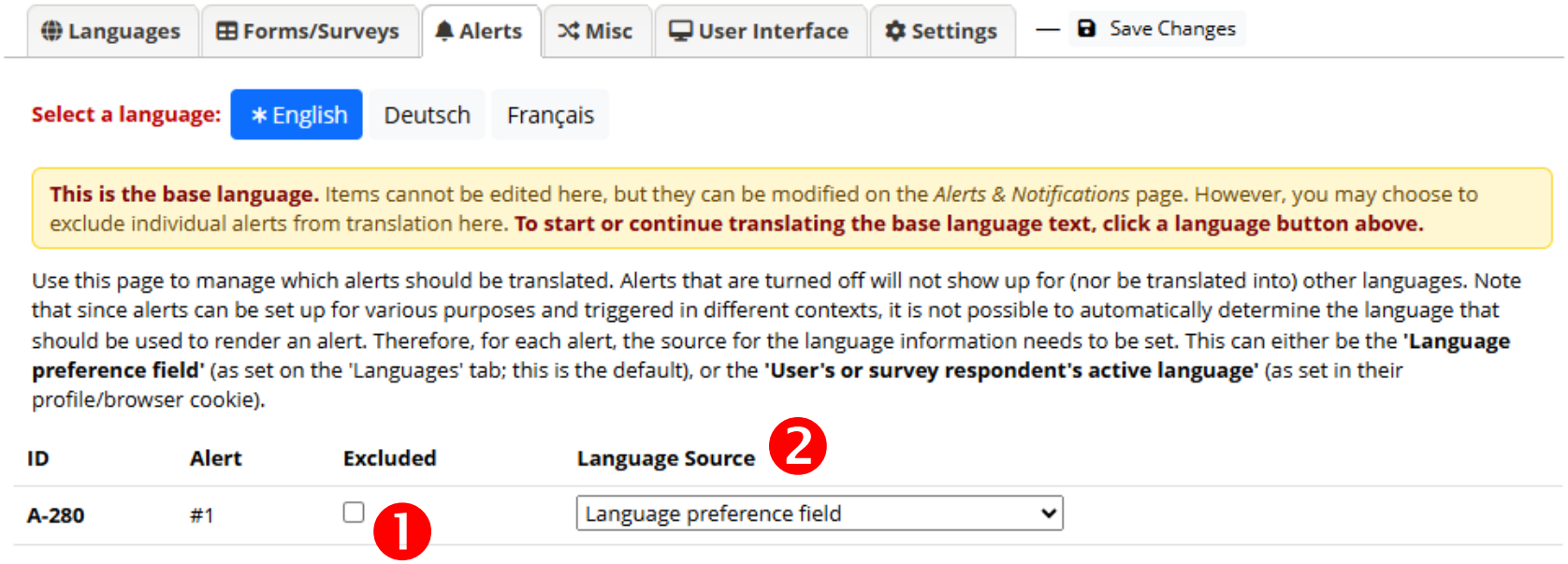
Make sure to faithfully replicate → any piping in the translations!



Alerts – Base language special options

1. Individual alerts can be excluded from translation
2. Since alerts may trigger under circumstances when no language can be determined automatically, set a **Language Source** for each event
   - **Language preference field** or **User's or survey respondent's active language**
   - What to choose will depend on how the alert is triggered

Select the **base** → **language** to get to this page!

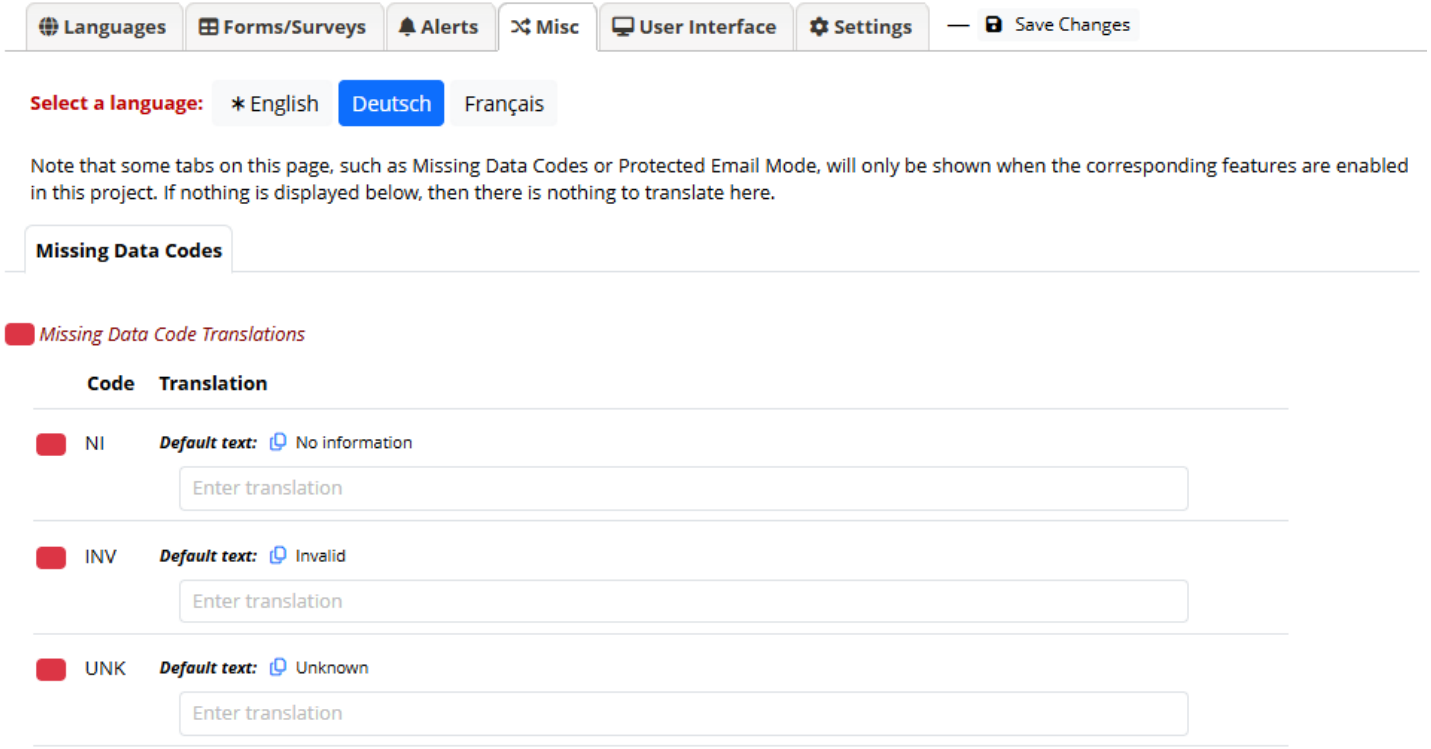


Miscellaneous items

- Missing Data Codes
- PDF customizations
- Protected Email customizations

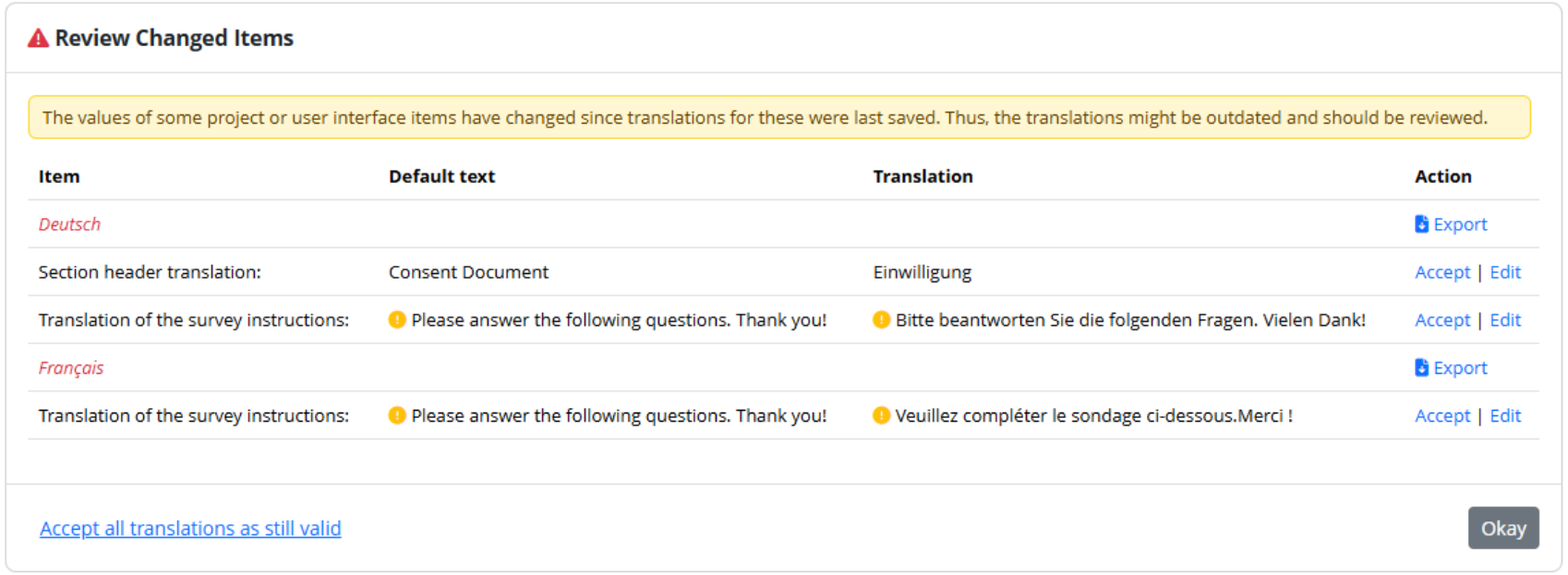
For the default language, there → is nothing to translate

Tabs will only appear when there are customizations set outside of MLM. Thus, it is not possible to add new text using MLM. Only existing items can be translated →



Change notification

- Whenever an underlying item changes, MLM will display a notification



- These items should be reviewed and edited as needed
- Still valid translations can be acceptes as still being valid

What determines the currently active* language?

* i.e., the language a page, email, or PDF is rendered in

Generally:

- A user's (survey respondent's) choice:
  - Set via the language selection widget (or a radio field when used with the **@LANGUAGE-SET** action tags)
  - On repeat visits through a cookie value (surveys, non-authenticated users) or user setting (data entry pages, authenticated users)
- For survey respondents only, upon first visits, MLM will (if not specified otherwise) try to set a language matching the preferred language set up in their browser

When this is not available, e.g., in a process triggered by ASI logic or a data import:

- A field in the project:
  - Set through the **Language preference field** option on the MLM setup page

  - Its value must correspond to the ID of an *active* language
  - It can be set manually (e.g., on an intake form) or captured into a field with the **@LANGUAGE-CURRENT-FORM/SURVEY** action tags
  - On data entry forms or surveys, a certain language can be forced with the **@LANGUAGE-FORCE** action tags

Note: When testing, always test in a **fresh, private/incognito** browsing window in order to not have "tainted" cookies!

REDCap

Setting the initially displayed surve language

- Generally, the language displayed for survey respondents is governed by a browser cookie that is set when they (first) select a language
- For new visitors, there are several ways to control the initially displayed language
  - Add a field tagged with a @LANGUAGE-SET action tag* to the survey while also populating the field with @DEFAULT, setting the desired language ID
    This field can be hidden, but when displayed, it should be a radio field and, when capturing the language used by the survey respondent is of interest, have the @LANGUAGE-CURRENT action tag applied
  - By adding a parameter (__lang – note the two underscores!) to the survey URL:

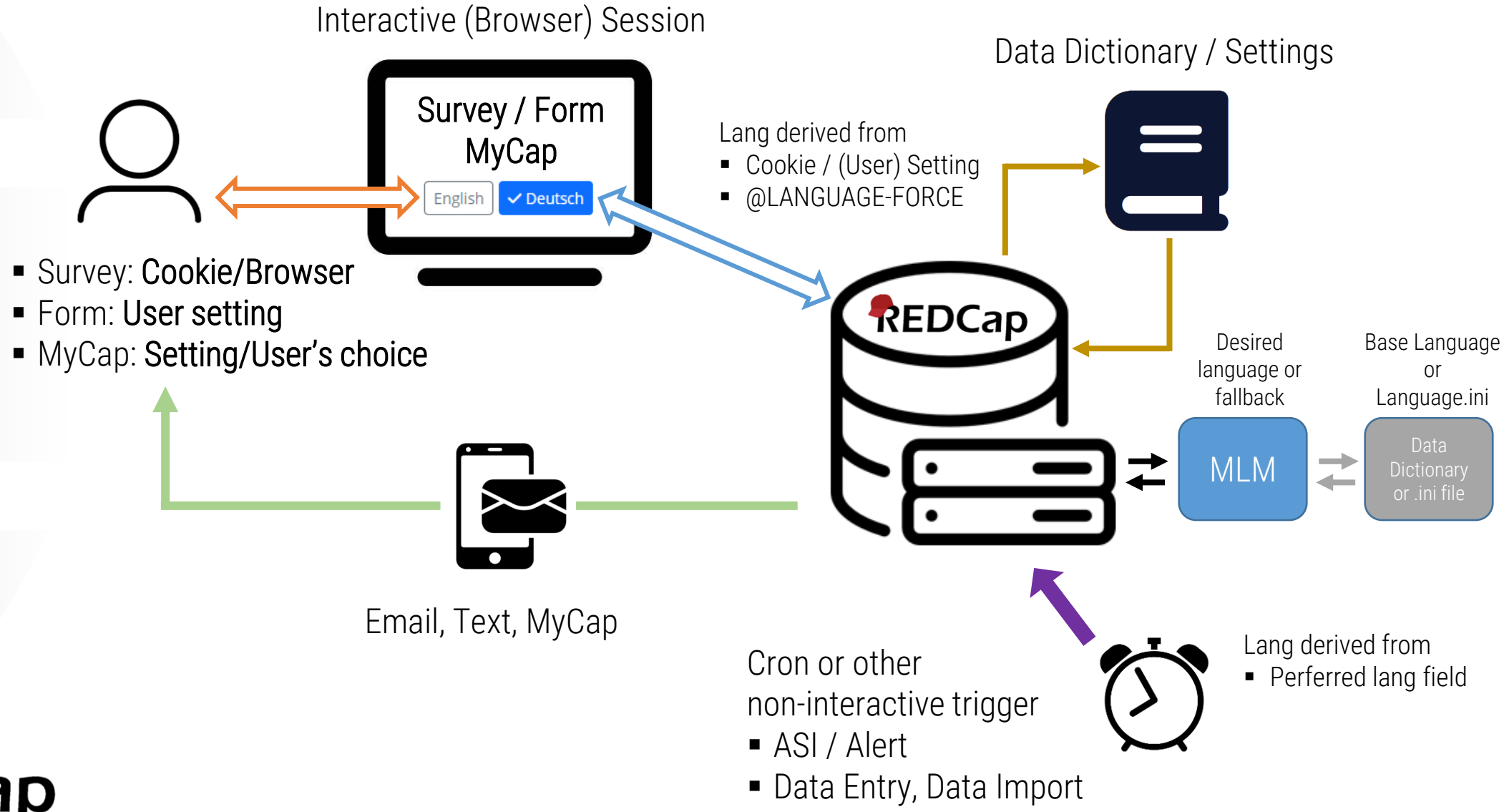
Tip: It is possible to set the *initially displayed language* of a survey by supplying the URL parameter **__lang**, which must be set to a valid (active) language id (case-sensitve!). Example: `https://redcap.server.tld/surveys/?s=ABCD1234&__lang=es`. When used, this will override both a survey respondent's previous choice (stored in a browser cookie) as well as the language preference field. The `@LANGUAGE-FORCE` action tag will still take precedence, though.

  - One of the @LANGUAGE-FORCE action tag can be used but will prevent the survey respondent from changing the language

* More information on MLM-related action tags is presented later

REDCap

MLM's language determination algorithm

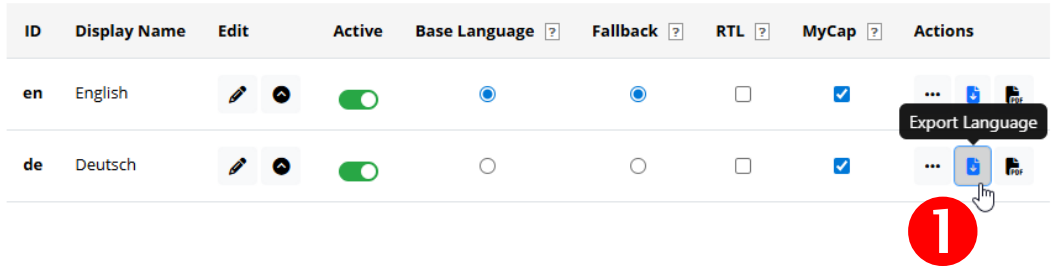


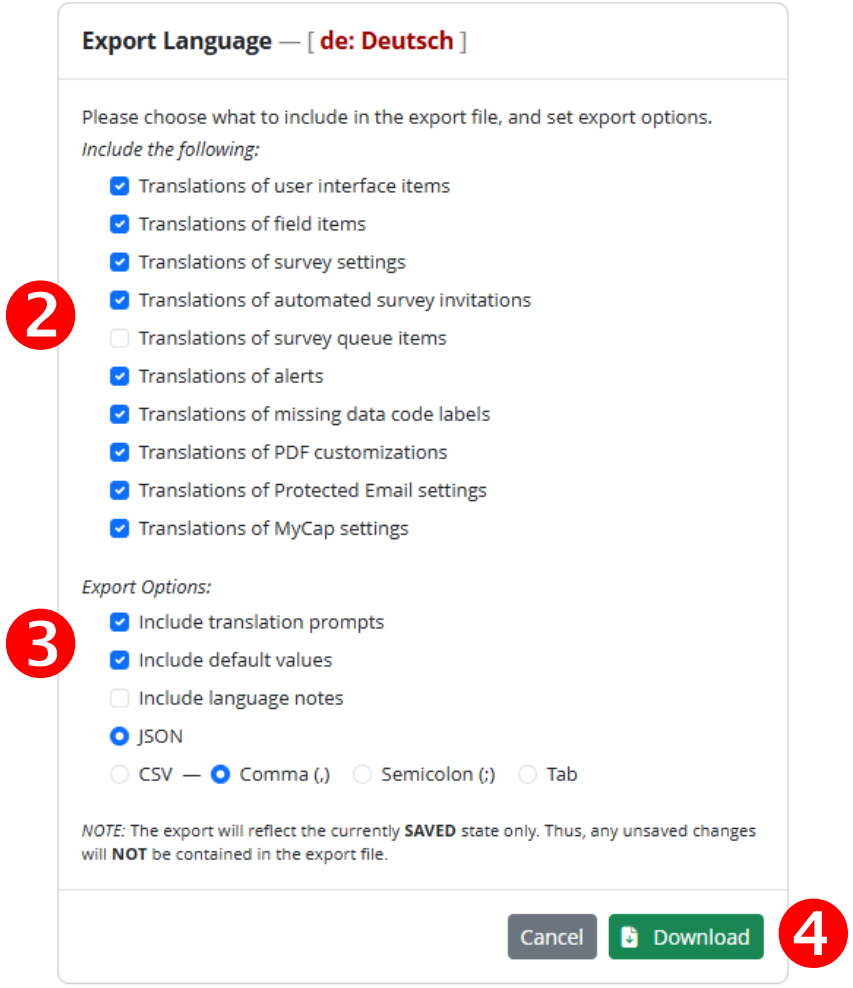
REDCap

Exporting languages (for re-use/offline translation)

1. Click on the **Export** button in the language's row in the languages table (or on the Forms/Surveys tab for a single instrument)
2. Choose the items that should be included in the export
3. Choose whether to include translation prompts and/or default values, as well as the file format (JSON or CSV)
4. Click the **Download Language** button

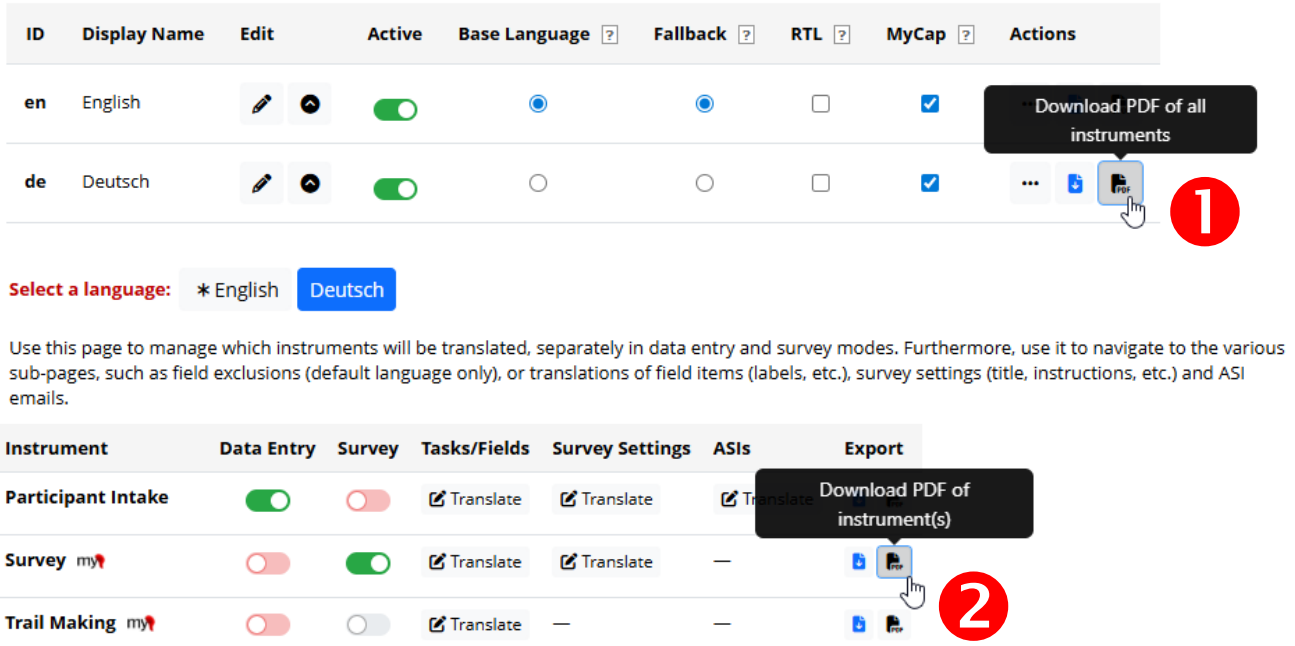
For offline translation, include **prompts** and **default values**!
For transfers or backups, these are not needed.





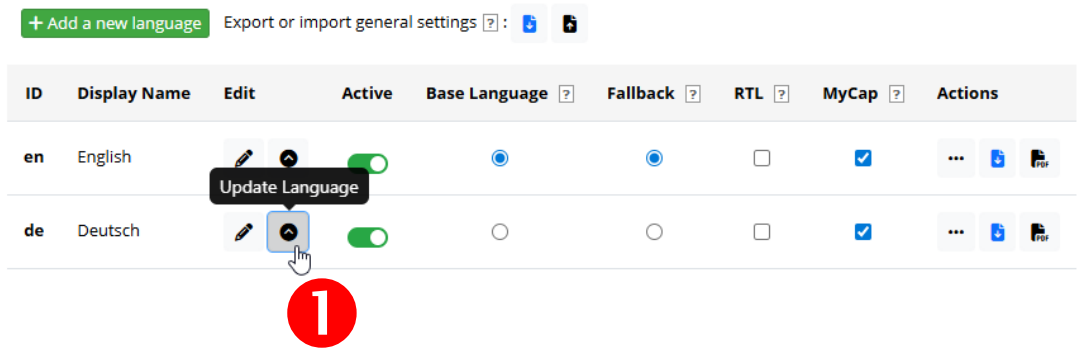
Exporting translated instrument PDFs

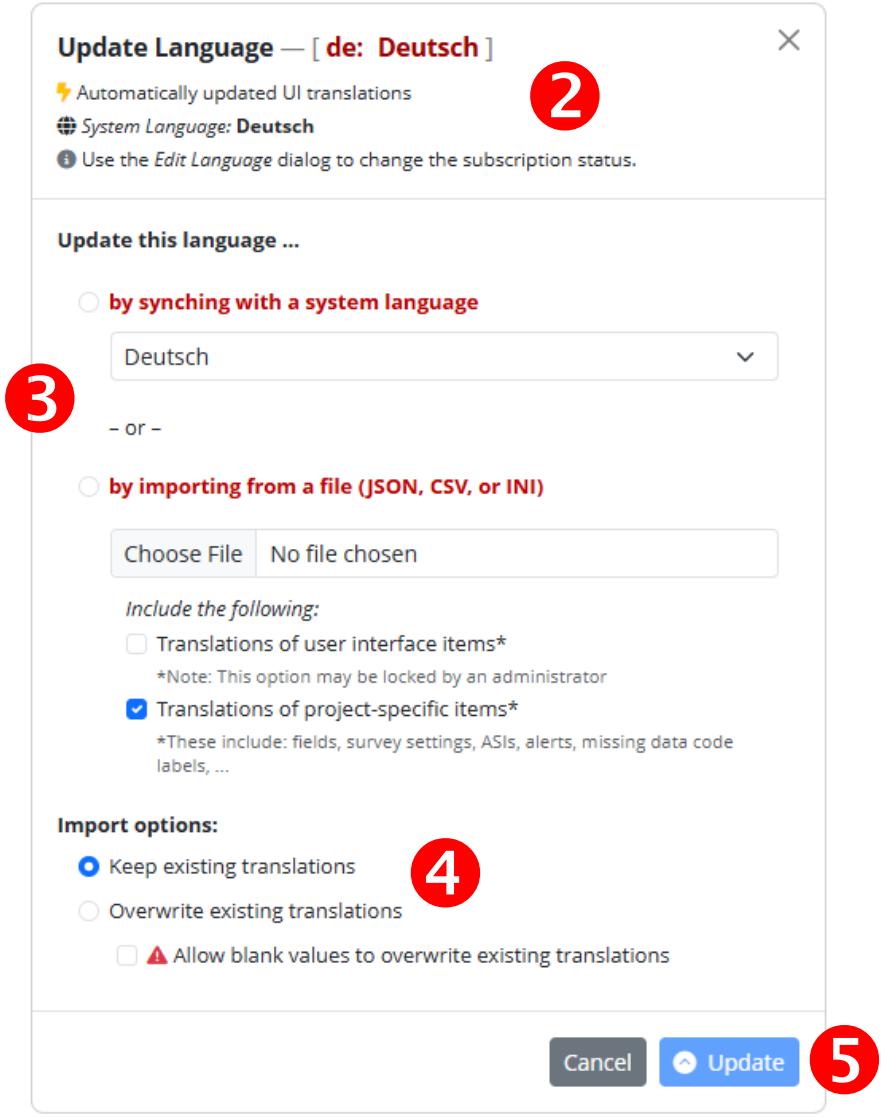
1. Export all (empty) instrument PDFs for a language
2. Export a translated (empty) PDF for an individual instrument
   - Note: Instruments without translations will be rendered in the default language



Importing (updated) translations

1. Click the **Update Language** button in the corresponding row on the Languages tab
2. The Update Language dialog will inform about the linkage/subscription to a system language (when applicable)
3. Choose an update source and what to import (choices may be limited depending on system settings)
4. Set import options
5. Click **Update** to perform the import (or synch) operation

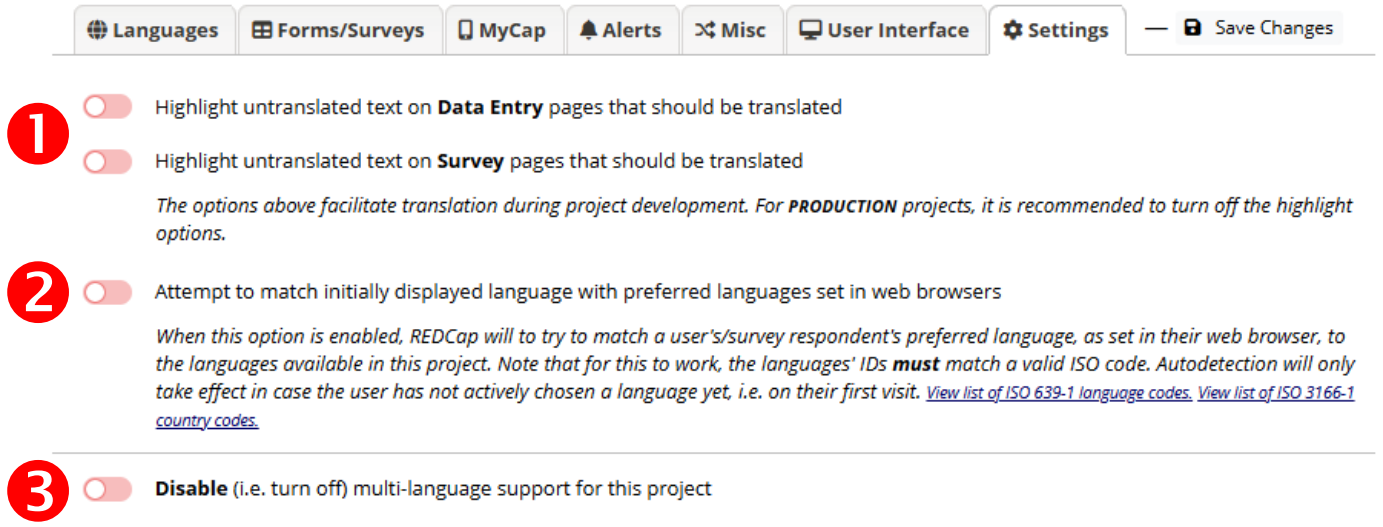




Settings tab options (1)

Users (with *Project Design* rights) have access to the following settings:

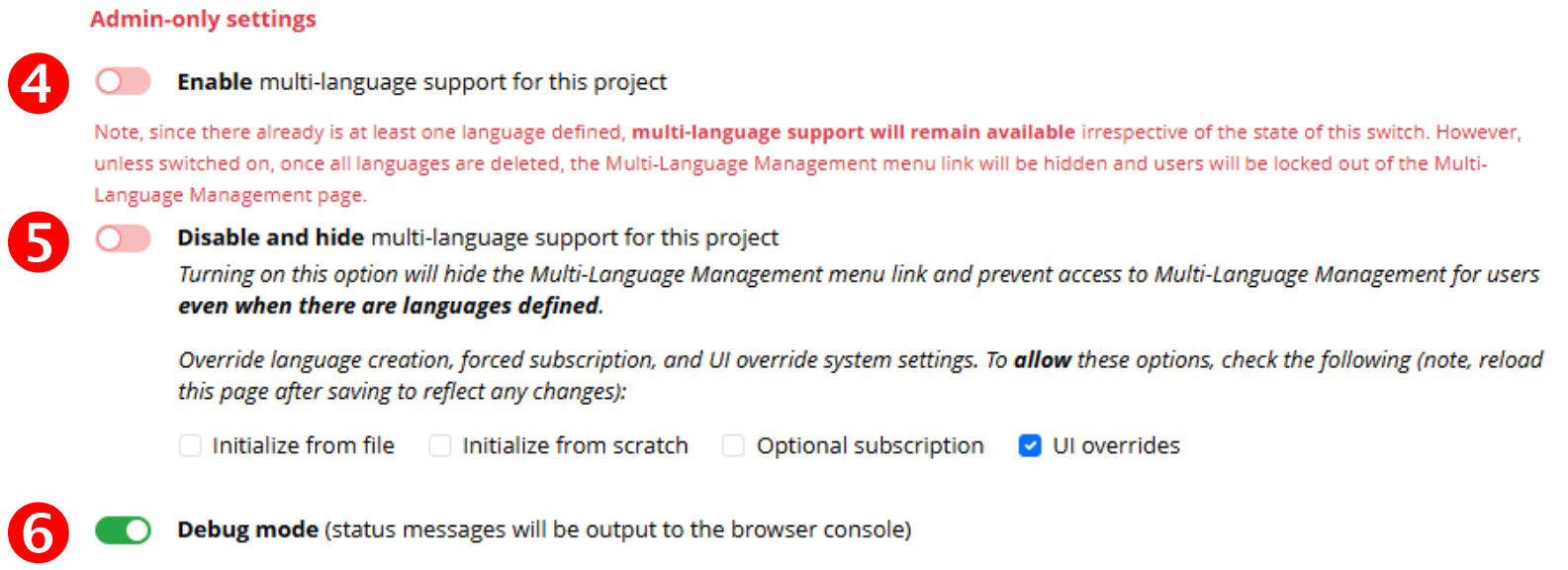
1. Option to **highlight untranslated text**
   This is useful during development as a visually check for items missing translations in the set language on data entry form and surveys
2. Option to **attempt to match** the initially displayed language with the preferred language set in the user's or survey respondent's browser
3. Option to **disable** multi-language support in the project
   This may be useful during troubleshooting issues, i.e., to exclude tha MLM is causing them



Settings tab options (2)

Admins have access to the following additional settings:

4. Option to **Enable** multi-language support when access to Multi-Language Management is set to require **admin activation** in the Control Center
5. Option to **Disable and hide** Multi-Language Management in the project
6. Option to enable **Debug mode**, which outputs rich diagnostic information to the browser console



Action Tags (1)

@LANGUAGE-CURRENT-FORM	Allows you to capture the currently used language in projects where multilingual data is enabled on data entry forms. The @LANGUAGE-CURRENT-FORM action tag can be used on fields of type 'Text Box' (no validation), and 'Drop-down List', or 'Radio Buttons' (these need to have choices whose codes correspond to the IDs of the defined languages - e.g., 'en'). This action tag is only active on data entry forms and will always, when possible, set the field's value to the currently active language. However, it has no effect when viewing survey responses.
@LANGUAGE-CURRENT-SURVEY	Same as @LANUGAGE-CURRENT-FORM, but works only on survey pages. For multi-page surveys, @LANGUAGE-CURRENT-SURVEY needs to be used on a field of each page where capture of the language is relevant (e.g. for performing branching).
@LANGUAGE-FORCE	When used on a field, the data entry form or survey on which the field is located will be rendered in the specified language (which must have been set up using the Multi-Language Management feature). The format must follow the pattern @LANGUAGE-FORCE="???", in which the ID of the desired language should be inside single or double quotes - e.g., @LANGUAGE-FORCE="de". Piping is supported - e.g., @LANGUAGE-FORCE="[field_name]" (if the source field is a radio field, make sure to add the *:value* qualifier). When the language is forced successfully (i.e., it exists and is active), the language selector is hidden. This action tag is evaluated when the form or survey loads. Using this together with @LANGUAGE-CURRENT-FORM/SURVEY on the source field for @LANGUAGE-FORCE may be used to 'lock in' a user to their selected language.
@LANGUAGE-FORCE-FORM	Same as @LANGUAGE-FORCE, but the effect is limited to data entry forms (i.e., this does not affect surveys).
@LANGUAGE-FORCE-SURVEY	Same as @LANGUAGE-FORCE, but the effect is limited to surveys (i.e., this does not affect data entry forms).
@LANGUAGE-MENU-STATIC	When this action tag is present on any field of a survey page and Multi-Language Management is active with at least two active languages, the language selection menu will remain visible at all times (i.e., it will not collapse after a language button has been clicked) on this survey page.
@LANGUAGE-SET	When used on a Drop-down or Radio Button field only, this action tag will allow the field's value to control the currently shown language (in the same way as switching the language via the buttons at the top of the page). Tip: When used in a survey, this field could be prepopulated (and thus auto-selected) by embedding a participant's language ID in the survey URL itself (for details, see the FAQ's "How to pre-fill survey questions" section).
@LANGUAGE-SET-FORM	Same as @LANGUAGE-SET, but the effect is limited to data entry forms (i.e., this does not affect surveys).
@LANGUAGE-SET-SURVEY	Same as @LANGUAGE-SET, but the effect is limited to surveys (i.e., this does not affect data entry forms).

REDCap

Action Tags (2)

@LANGUAGE-CURRENT-FORM
@LANGUAGE-CURRENT-SURVEY

- Use case:
  To capture the language that was active on a form or survey
- Syntax:
  These action tags do not take any parameters
- Where to use:
  Apply to a text box field (without validation) or to a radio (or dropdown) field that has choices whose codes match the IDs of the languages
- Example:

4	[lang_pref]	What is your language preference?	radio, Required en, English de, Deutsch Field Annotation: @LANGUAGE-SET @LANGUAGE-CURRENT-SURVEY

Action Tags (2)

@LANGUAGE-SET
@LANGUAGE-SET-FORM
@LANGUAGE-SET-SURVEY

- **Use case:**
  Set the currently displayed language (on a survey or data entry form) from a field in real-time
- **Syntax:**
  These action tags do not take any parameters
- **Where to use:**
  Apply to a radio (or dropdown) field only. The choice codes must match the language IDs. Can be combinen with a (matching) @LANGUAGE-CURRENT-X action tag
- **Example:**

4	[lang_pref]	What is your language preference?	radio, Required en English de Deutsch Field Annotation: @LANGUAGE-SET @LANGUAGE-CURRENT-SURVEY

Action Tags (3)

@LANGUAGE-FORCE
@LANGUAGE-FORCE-FORM
@LANGUAGE-FORCE-SURVEY

- Use case:
  Forces a language and disables the language selection widget.
  After an initial choice on, e.g., the first survey page, you may want to lock in the survey respondent to the chosen language for the remainder of the page
- Syntax:
  These action tags take a string parameter. Piping is supported (use **:value** when piping from a radio/dropdown field. The value must match one of the (active) language IDs
- Where to use:
  Apply to any field on an instrument. On multi-page surveys, the action tag needs to be applied to all pages where its effect is desired
- Example:
  `@LANGUAGE-FORCE-SURVEY=[lang_selection:value]`

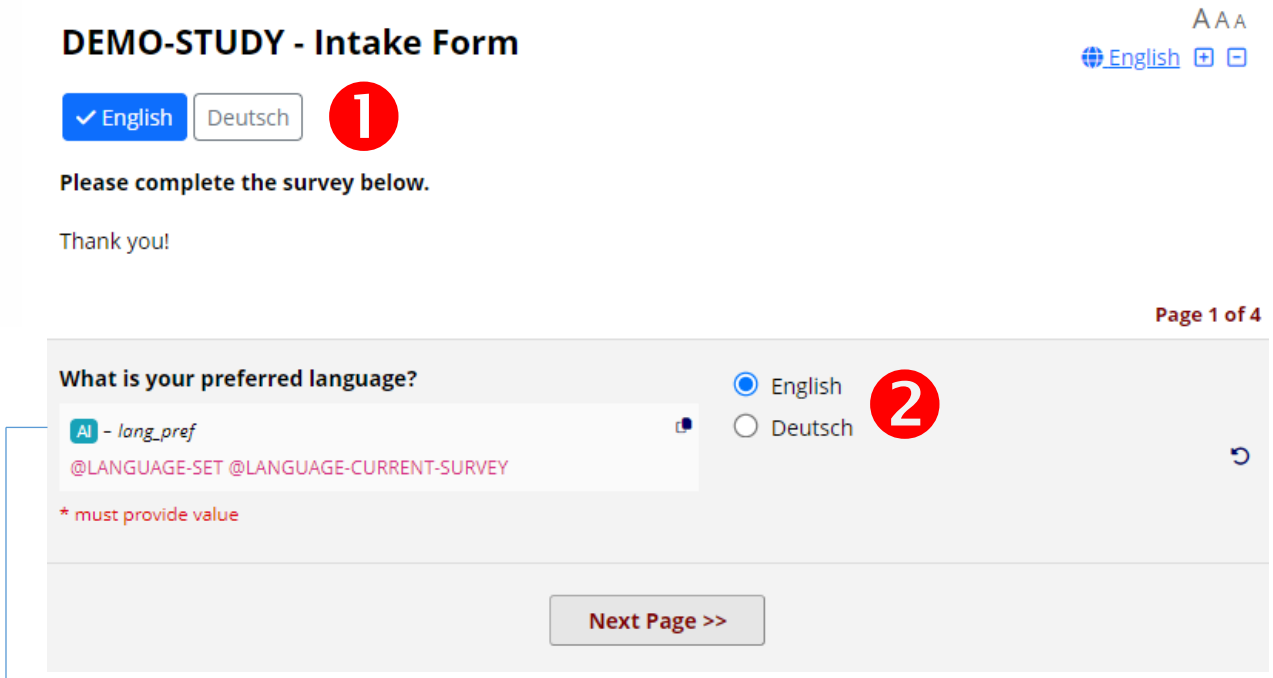
Action Tags (4)

@LANGUAGE-MENU-STATIC

- Use case:
  To permanently show the expanded language selection menu on survey pages
- Syntax:
  These action tags take a string parameter. Piping is supported (use **:value** when piping from a radio/dropdown field. The value must match one of the (active) language IDs
- Where to use:
  Apply to any field on an instrument. On multi-page surveys, the action tag needs to be applied to all pages where its effect is desired
- Example:
  `@LANGUAGE-FORCE-SURVEY=[lang_selection:value]`

Action Tags – Exemplary use case (1)

Consider a study targeted at populations speaking various languages. On the intake survey, you might want to ask candidates about their language preference before presenting consent information in this language and asking whether they are willing to participate in the study in the chosen language:



Field annotations provided by the *Admin Insights* External Module

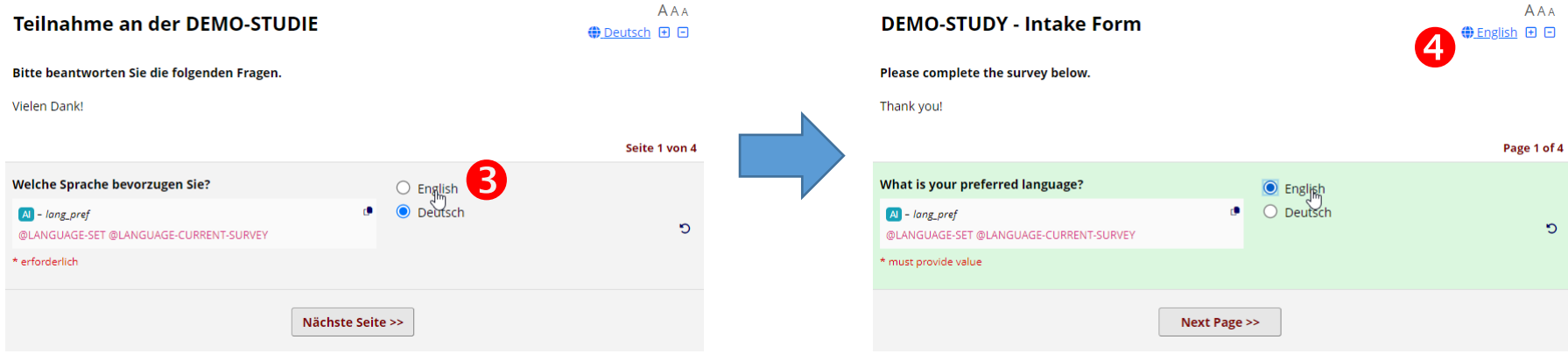
Note that both, @LANGUAGE-SET and @LANGUAGE-CURRENT-SURVEY have been applied to the lang_pref field (a radio field with the codes en and de, corresponding to the MLM language IDs)

This will sync the language menu (❶) with the radio field (❷)

Action Tags – Exemplary use case (2)

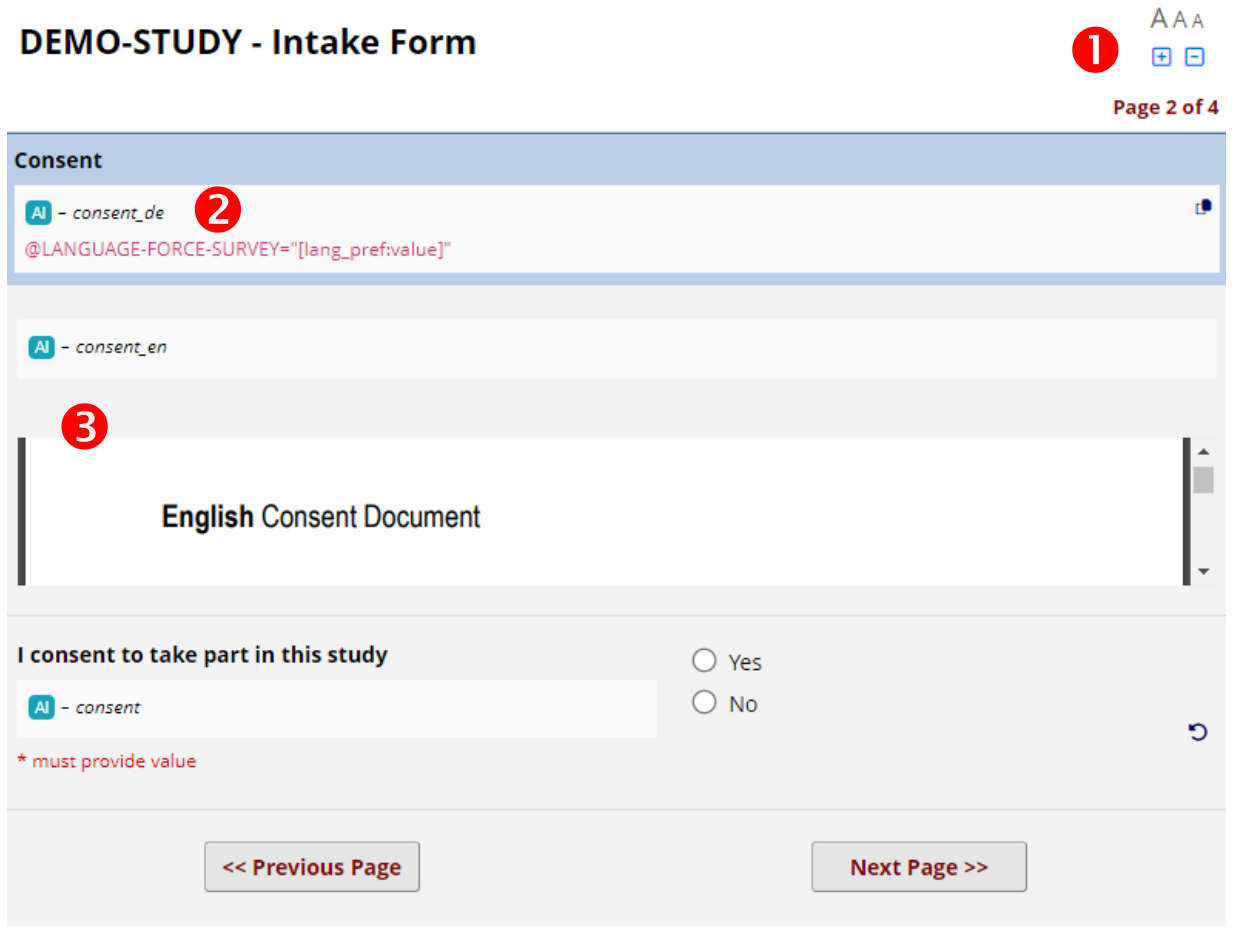
Clicking “Deutsch” on the language menu (❶) will swap English strings with their corresponding German translations **AND** change the radio field to the “de” choice (❷)

Conversly, clicking the “English” radio choice (❸), will switch to the “English” language (❹).



Action Tags – Exemplary use case (3)

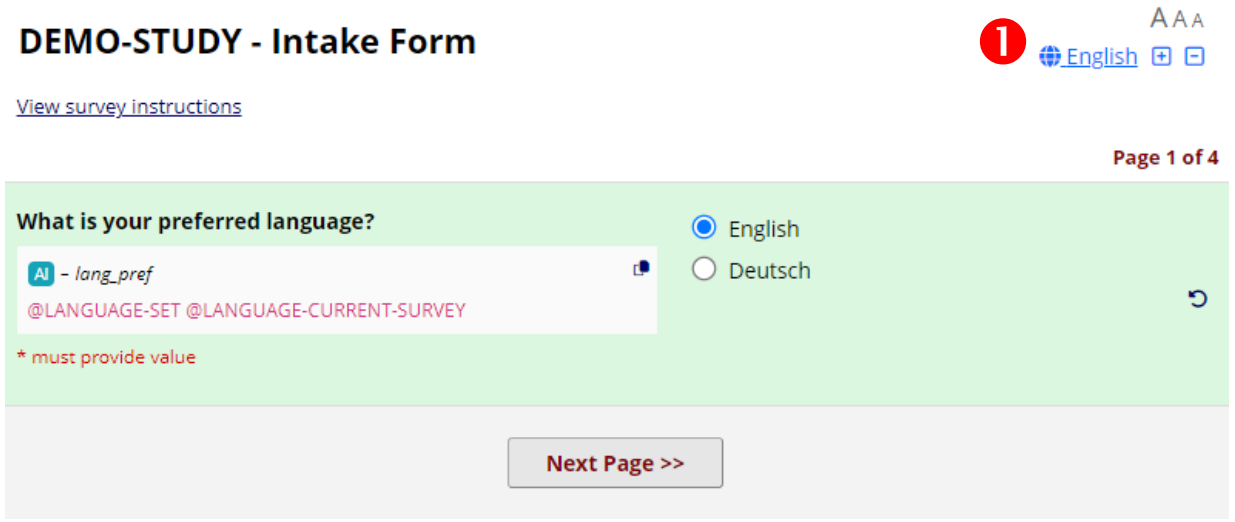
Assuming the survey respondent chose “English” as their preferred language and moved to the next survey page, this is what will be displayed:



- Note that the language menu toggle is not displayed (❶)
- This is due to the **@LANGUAGE-FORCE-SURVEY** action tag on the field **consent_de** (❷) It mandates the *current survey page only* to be rendered in English. Note that ***:value*** had to be added because **lang_pref** is of type radio
- Branching logic is used to show the appropriate consent document (❸)

Action Tags – Exemplary use case (4)

Going back to the first page of the survey will re-enable the language selection (❶), because there is no @LANGUAGE-FORCE-SURVEY action tag on this page



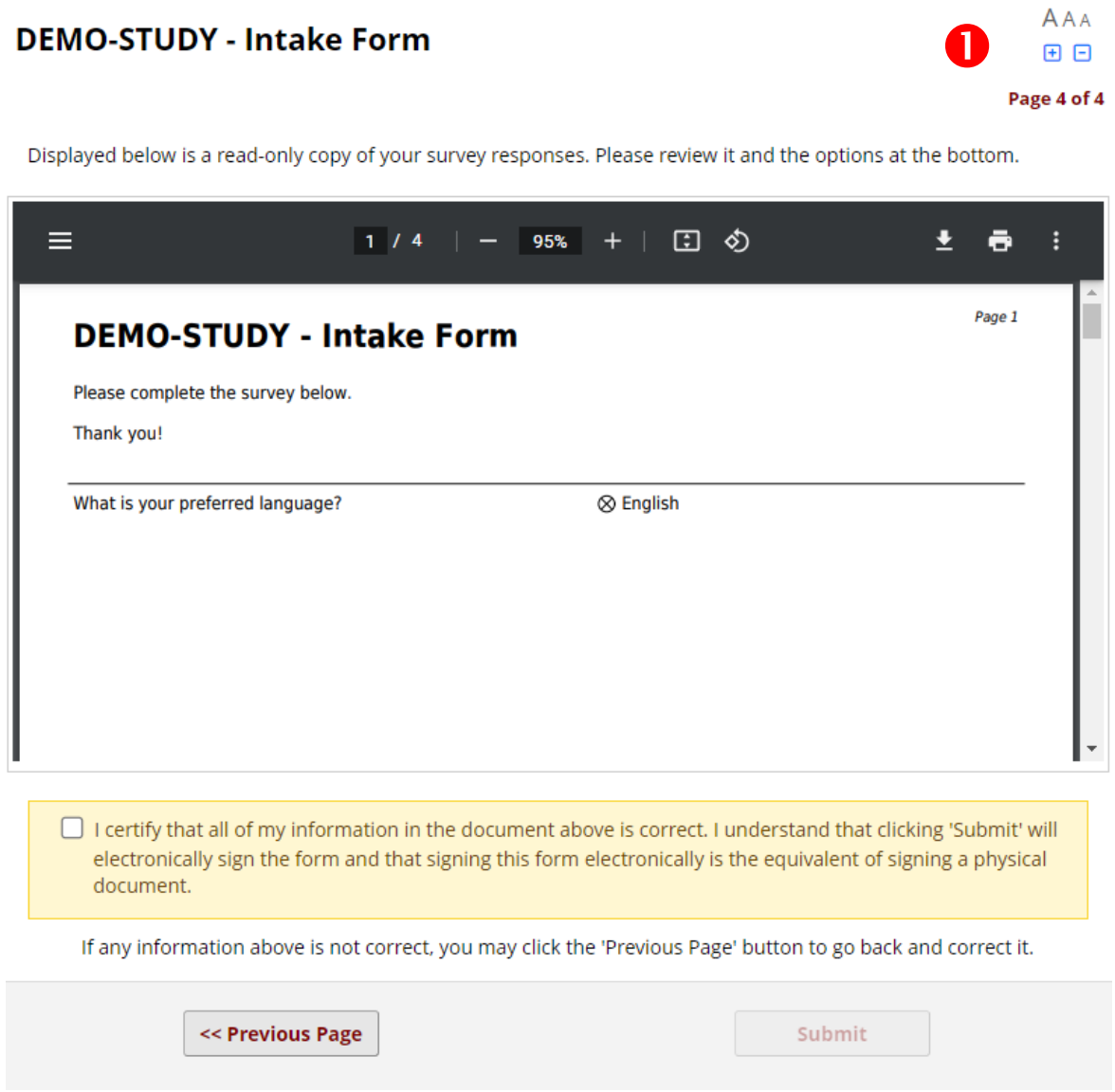
It is up to the project designer or study requirements, whether survey respondents should be locked into a certain language after it has been chosen initially (or preset, e.g. on a non-survey administrative form)

This behavior could be fine-tuned through the use of the @IF action tag. @LANGUAGE-MENU-STATIC could be considered for the first page of this multi-page survey, too

REDCap

Action Tags – Exemplary use case (5)

The language menu is **never** shown on the e-consent page (❶), but instead, the language used on the previous page gets locked in for the e-econsent
page (this is because on-the-fly swapping of the PDF's language is not supported)



Action Tags – DOs and DON'Ts

- Be specific! If language preference is collected on a survey (using @LANGUAGE-CURRENT-SURVEY) and multiple languages are supported in data entry mode as well, you probably do not want to have @LANGUAGE-CURRENT-FORM on this field as well
- Similarly, if a survey field is synchronized with the language menu (via @LANGUAGE-SET) and the study team has multi-lingual data entry forms, you might want to limit synchronization to surveys only. This can be achieved by selectively applying @LANGUAGE-SET via @IF and the smart variable [is-survey] or by using the form/survey-specific variants (@LANGUAGE-SET-FORM, @LANGUAGE-SET-SURVEY)
- There is probably rarely a reason to use @LANGUAGE-FORCE or @LANGUAGE-FORCE-FORM
- Keep in mind that @LANGUAGE-FORCE(-SURVEY) and @LANGUAGE-MENU-STATIC are limited to the currently displayed page in a multi-page survey
- When piping from a radio into one of the @LANGUAGE-FORCE action tags, be sure to apply the *:value* option