

## Deliverable 7.3 Design specification of the GUI (microstructure)

### 1. Macrostructure of the search interface

#### 1.1 Requirements from user study

The analysis of the usage scenarios for different types of users revealed that language teachers, testers, teacher trainers and textbook authors have similar demands towards a search interface while linguists (researchers) need to make use of more fine-grained search, filtering and display options.

#### 1.2 Search components

In accordance with the users' requirements, the search interface has been designed to contain different layers:

- A **simple** and an **advanced search** for accessing examples of learner language in their immediate textual context.
- A **document search** to filter documents by learner characteristics (such as their L1, age, test level, etc.) and to create subcorpora, which can be used in the simple and advanced searches.
- A search by **learner language features** to derive statistics on different learner language feature types for groups of texts or individual texts, and a direct linking to contextual searches for learner language features (ie. advanced search).

#### 1.3 Results displays

For the presentation of search results several display modes are distinguished:

- text section with search word highlighted (KWIC view)
- text section with different annotation layers (e.g. learner language features, lemma pos) and the respective search unit highlighted (KWIC view in ANNIS)
- entire text (parallel view on original and target)
- metadata on a selected text
- table with frequencies of learner language features

#### 1.4 Information flow between search components and search & results presentation

The "home" page of the MERLIN platform gives access to all search components. When entering the platform the user is initially send to the basic search page.

Horizontal navigation between all search components is possible at any time.

Within each search component the user is navigated step-by-step

1. starting at the search page (for specifying details of the search)
2. moving towards the overview results page (e.g. lines of textual context (KWIC), list of documents, etc.)
3. entering into details of the results (e.g. metadata, entire text, etc.)
4. furthermore, detailed results can be printed or exported.

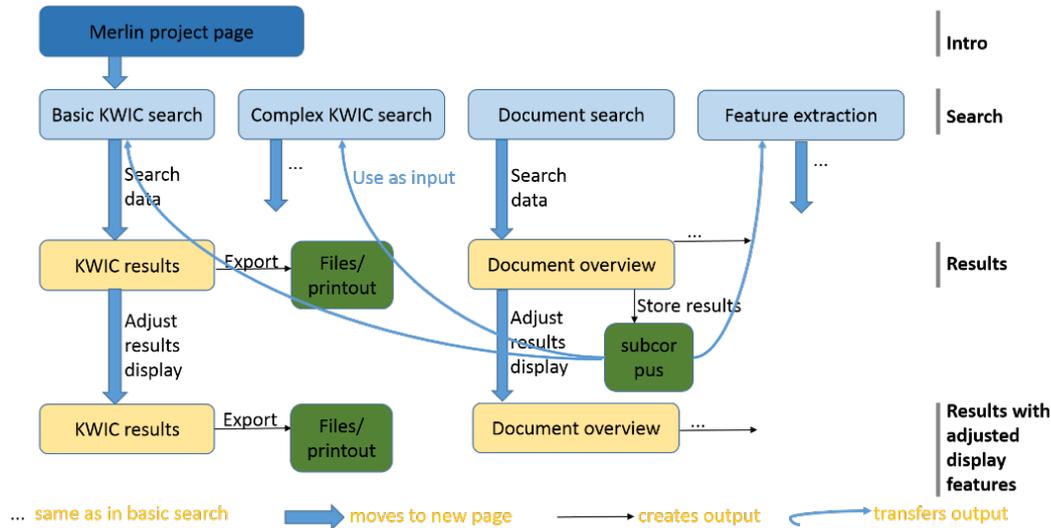


Figure 1 - Structure of search options and information flow

## 2. Microstructure of the search interface

### 2.1 General design principles

For all search components the following design criteria are devised:

- open category searches are presented as text boxes
- closed category search options are presented in drop-down menus
- interdependencies between search fields are dynamically enforced (on select certain search options are enabled or excluded)
- sequential searches are placed from left to right
- corresponding search fields are aligned and unified in size
- all interactive interface elements are explained in a contextual help (indicated by help icon)

The navigation between search and results displays is supported by:

- custom buttons for backward navigation
- temporary retaining of previously entered search parameters
- display of the currently applied search and filter criteria
- metadata details for all textual content

## 2.2 Simple search

The simple search is designed to be self-explanatory and very slim in terms of the amount of presented information and search options. Its design foresees a simple “google-style” text box for entering search terms. A virtual keyboard facilitates to enter letters with diacritics (such as à)



Figure 2- Simple search

## 2.2 Advanced search

The advanced search is designed to provide a wide variety of search options, including different annotation layers. No knowledge about any query language is required as all search options are accommodated in graphical search fields (text boxes, drop-down menus, radio buttons).

In addition to free search fields the advanced search offers a set of predefined queries which can be started on click.

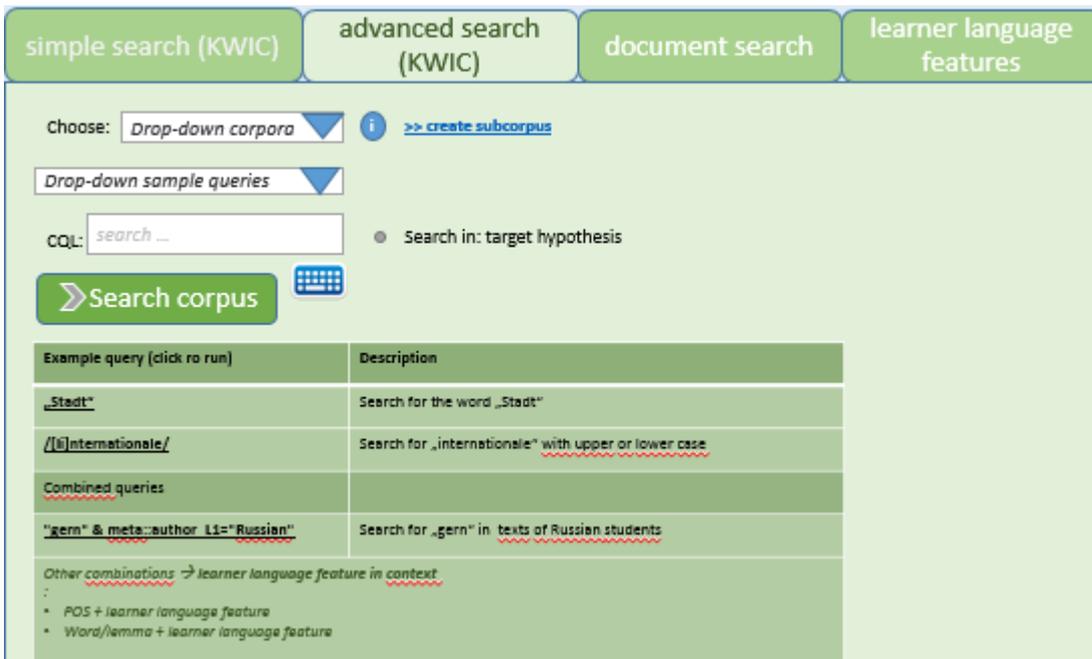


Figure 3 - Advanced search

### 2.3 Document search

The document search is designed to support the filtering and grouping of documents by metadata characteristics as well as by contained learner language features.

To facilitate searches all metadata search options and learner features are displayed in drop-down menus, while open text fields allow for the specification of search terms chosen by the user.

Subcorpora can be build accordingly and are stored by a user-defined name. Subcorpora are kept online for a session of 24 hours.

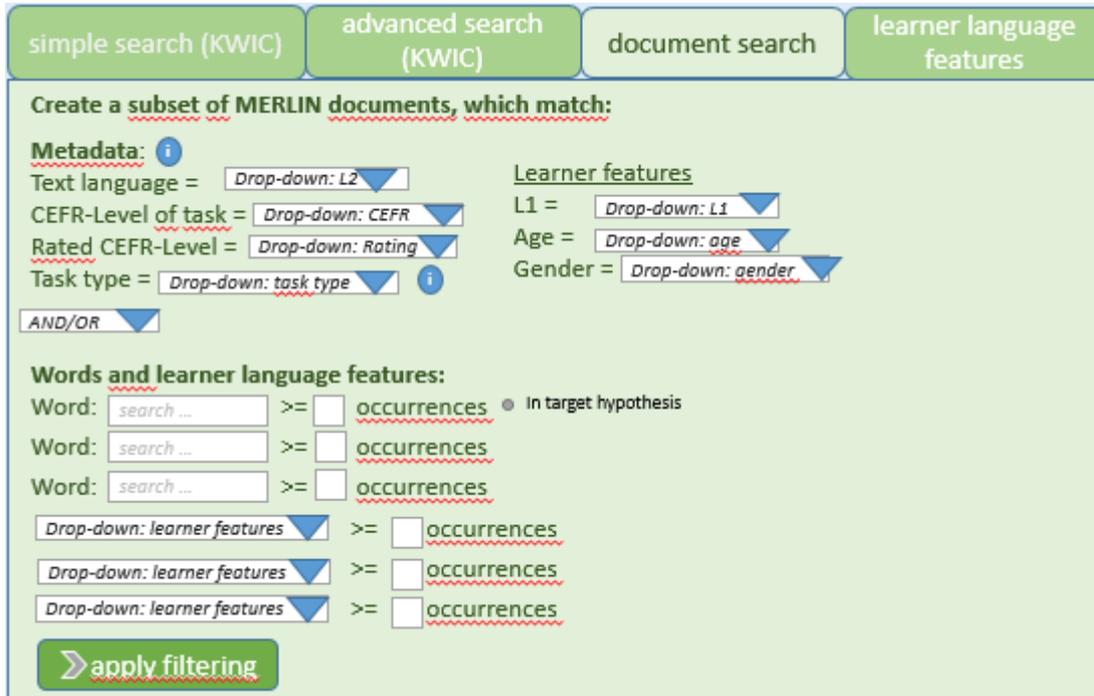


Figure 4 - Document search

### 2.4 Learner language features

The learner feature search is designed to build on user-defined subcorpora or individual corpus documents. Searchable learner language features are available in a drop-down list.

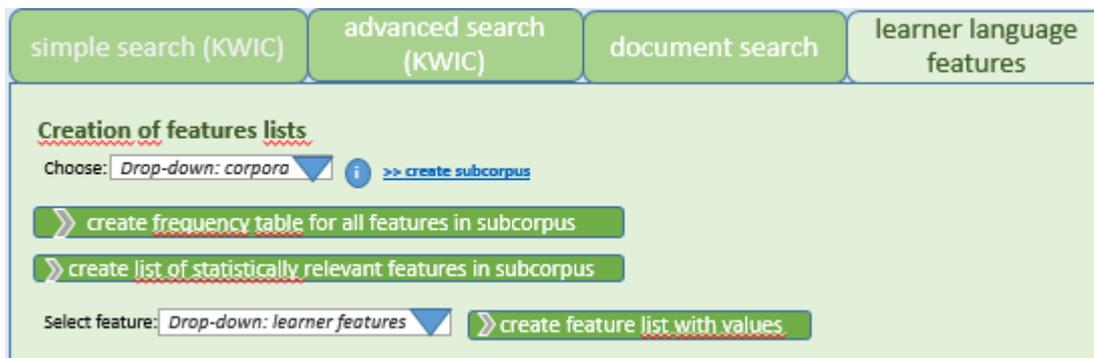


Figure 5 - Learner language features

## 2.5 Results display

Results of simple and advanced searches are presented as search unit (e.g. word) in its textual context. Results are clickable to arrive to metadata or the entire texts related to single search hits.

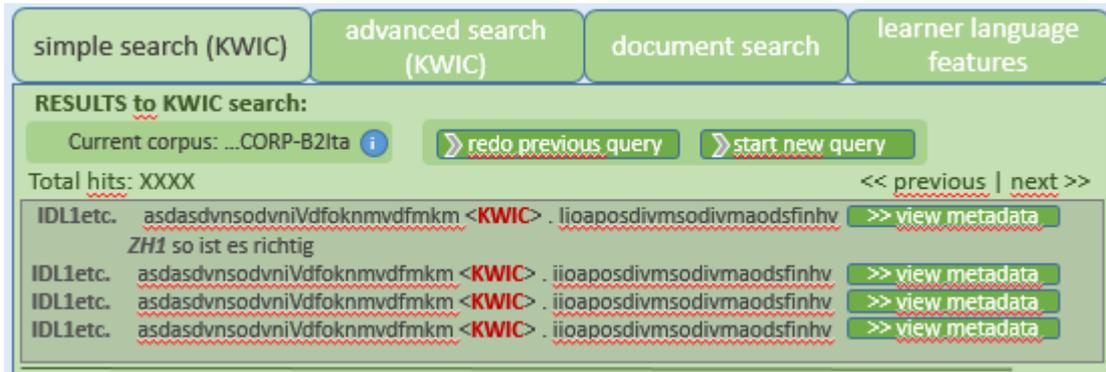


Figure 6 - Results display of simple and advanced searches

Results of the document search are presented as table-styled overview of the different documents matching the search. Search/filter criteria are explicitly named.

On-click each document can be inspected, showing the original learner text as well as the target hypothesis and metadata for the text.

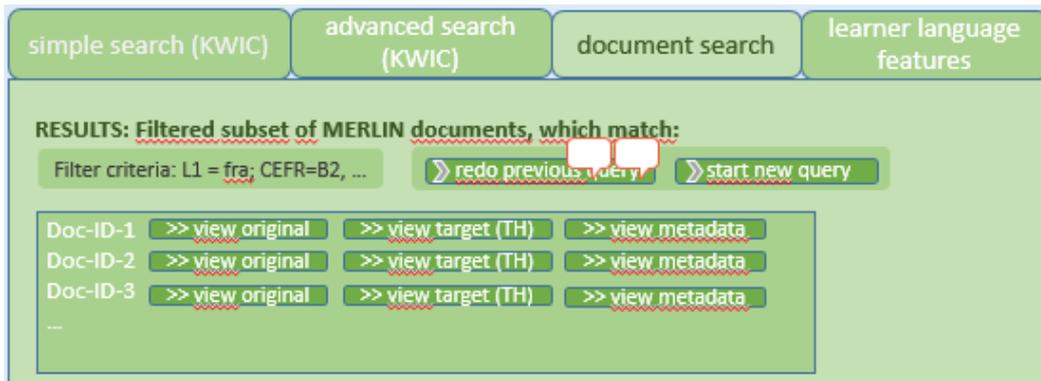


Figure 7 - Results of document search

Results to learner feature searches are presented in frequency tables.

The frequency counts for different learner features are clickable, to launch searches for learner features in context.



Figure 8 - Results of learner feature search



This project has been funded with support from the European Commission. This publication [communication] reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Please cite as: MERLIN project, Design specification of the GUI, 2014, <http://merlin-platform.eu>