UI-FRONT

Role : User Interface of the application

We describe each page of this module. For each page, we give its path.

/login

- To access the app, the user has to be logged or create an account. Otherwise, he’s redirected to this page.

- Identification by e-mail. So two user cannot share the same password.

- This is a one way password. Its hash is saved in the database. The security isn’t enough good yet.

/accueil

- By default, a user connected is redirected here.

- The button “accueil” redirect to this page. The button “user” to the page “/user/” and “logout “ log out the user and he’s redirected to “/login”

- We can start a research within this page in the block “New Research”. The characters for the keywords are : A-Z, a-z, 0-9 ,‘-’ . Each keyword has to be separated with a ‘,’ for AND , ‘;’ for OR or parenthesis ‘()’. We can use a keyword composed with many word with double quotes “”. To match an article, the words within double quotes has to match exactly. We can choose the interval of publication’s date. When we submit, we will have a window to confirm our decision.

- If we are not sure, with the button “evaluate”, we can check how much article we could fetch at least.

- We can check if there is an ancient research that correspond within block “Historical Research”. We can put keywords separate with space. The logical between the is a “OR” and the match isn’t exact. By example, ‘vitamine cancer’, it will give us all research that have vitamine or cancer in their research. It match if we retrieve the character chain exactly. For “cancer” it will match with “cancerologue” by example. We can make keyword with multi-words with double quotes “”. The search is insensitive to case, it means “KEY” and “key” give the same results.

- We can choose how the result of the historical is done. “Pertinance” means if you have the most of your keyword that match, you are first. “Article +/-” mean the research that have the most/least articles is first.

- The result of a historical give us the string chain, the keywords, the score of clustering and a link to “/select” with the identifier of this research.

/select

- In this page, if we gave the id of an existing research, we have the plot of all articles order by the clustering process. With the mouse, we can examine all point, their cluster and meta-data. This plot can display in fullscreen with the button “fullscreen”

- In this page, we can create some filter to reduce the number of article wanted. We have 4 type of filter:
- “Topic” we get all article of a cluster
- “Author” we check if the name correspond to a author last name or first name. If this is the case, we get all article from this author. The match isn’t exact: “Carl” can give for author with name “Carlos”. This is insensitive to case.
- “Keyword” we check if the keyword is in an article or in the abstract of an article. We get all article with at most a match. The match is insensitive to case and not exact as the previous one.
- “DOI”. We give the DOI of an article we think it is in our result. The DOI given must be exact. If the article exist, it give us a number of article that are the nearest from our article that we gave the DOI. But, it doesn’t give us this article, only its neighbors.

- The filtering process is composed of block and each block has many filters from the 4 previous type. Each filter give us a set of article according to previous rules. Within a block, each set are associate with AND logical. It gives us a new set associate to the block. The set of each block are associated with OR logical.

- To generate a table of articles we can check, there is the button “generate”. We obtain a window to check if the user is sure to continue. It gives the number of articles of the final set of articles. If cancel, we return the page with all filters. Otherwise, it redirect to “/tablechoice”.

/tablechoice

- In this page, we have a table that represent the set of articles obtained from the previous page thanks to filters. Each line has a check box, the title,the authors,the abstract and a link to the article.

- After the user is satisfied with his check list, he can push the button “iterate”. Each article not checked will be hidden. Each article checked will be displayed with their checkbox checked and its nearest neighbors are displayed too. The number of new article is displayed and the number of checked article too.

- If we want to return to the first table, we push the button “reset”

-If we are satisfied with articles checked and don’t want to iterate, we can finish it with the button “finish” and obtained a zip file with all articles checked. Each article is in his topic directory. If we don’t have the article, it create a .txt file with the meta- data of the article.

- For a research and a user, we can have a unique table. If the user make another table with another research or the same, the current table is erased.

/user

- This is the page with all data about the user. We have all processing research with the current step and the current achievement. We have all research finished.