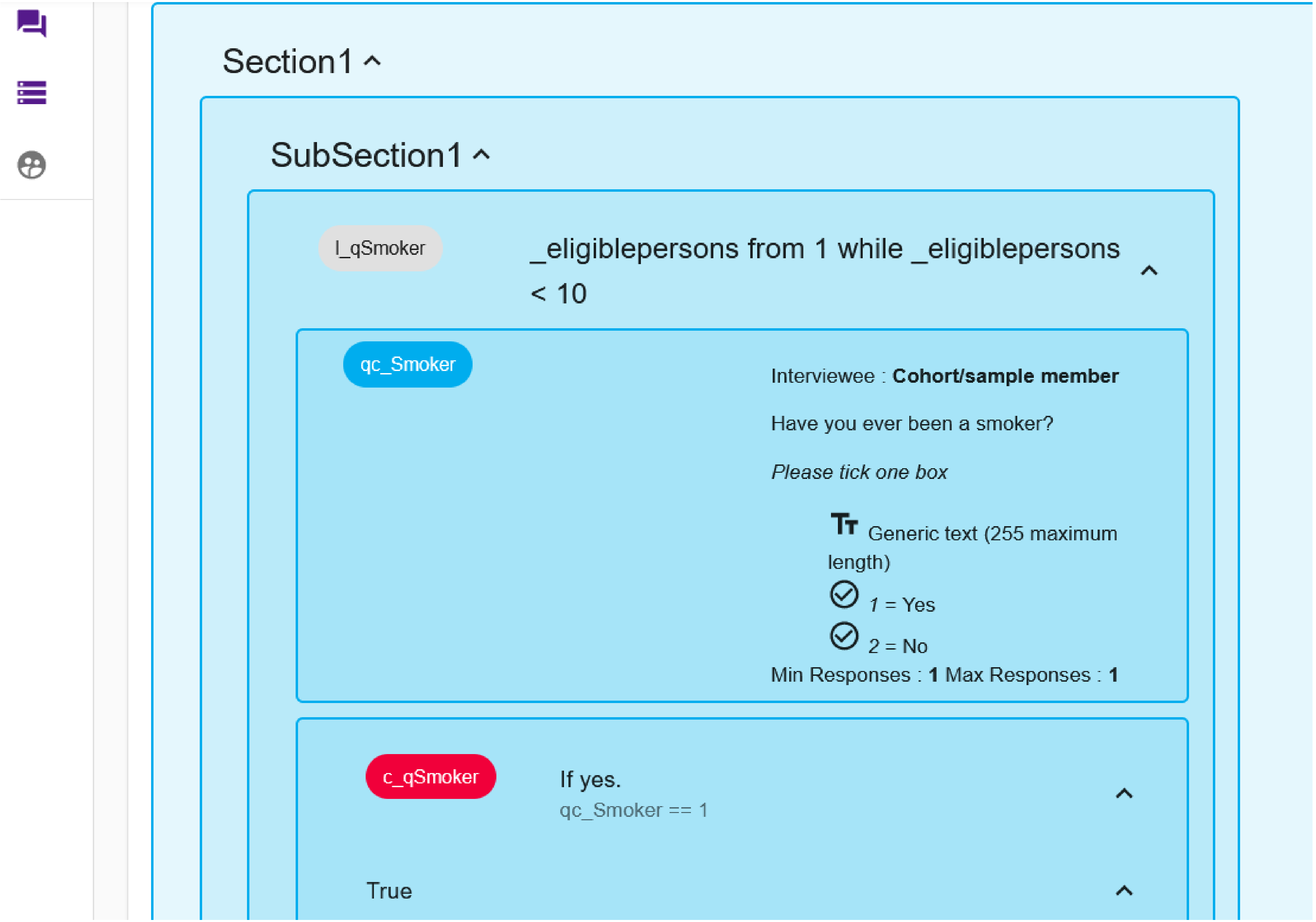
closer

The home of longitudinal research

Work-flow Automation

Jenny Li,
Metadata developer
CLOSER



Section1

SubSection1

l_qSmoker

_eligiblepersons from 1 while _eligiblepersons < 10

qc_Smoker

Interviewee : **Cohort/sample member**

Have you ever been a smoker?

Please tick one box

Tт Generic text (255 maximum length)

✓ *1* = Yes

✓ *2* = No

Min Responses : **1** Max Responses : **1**

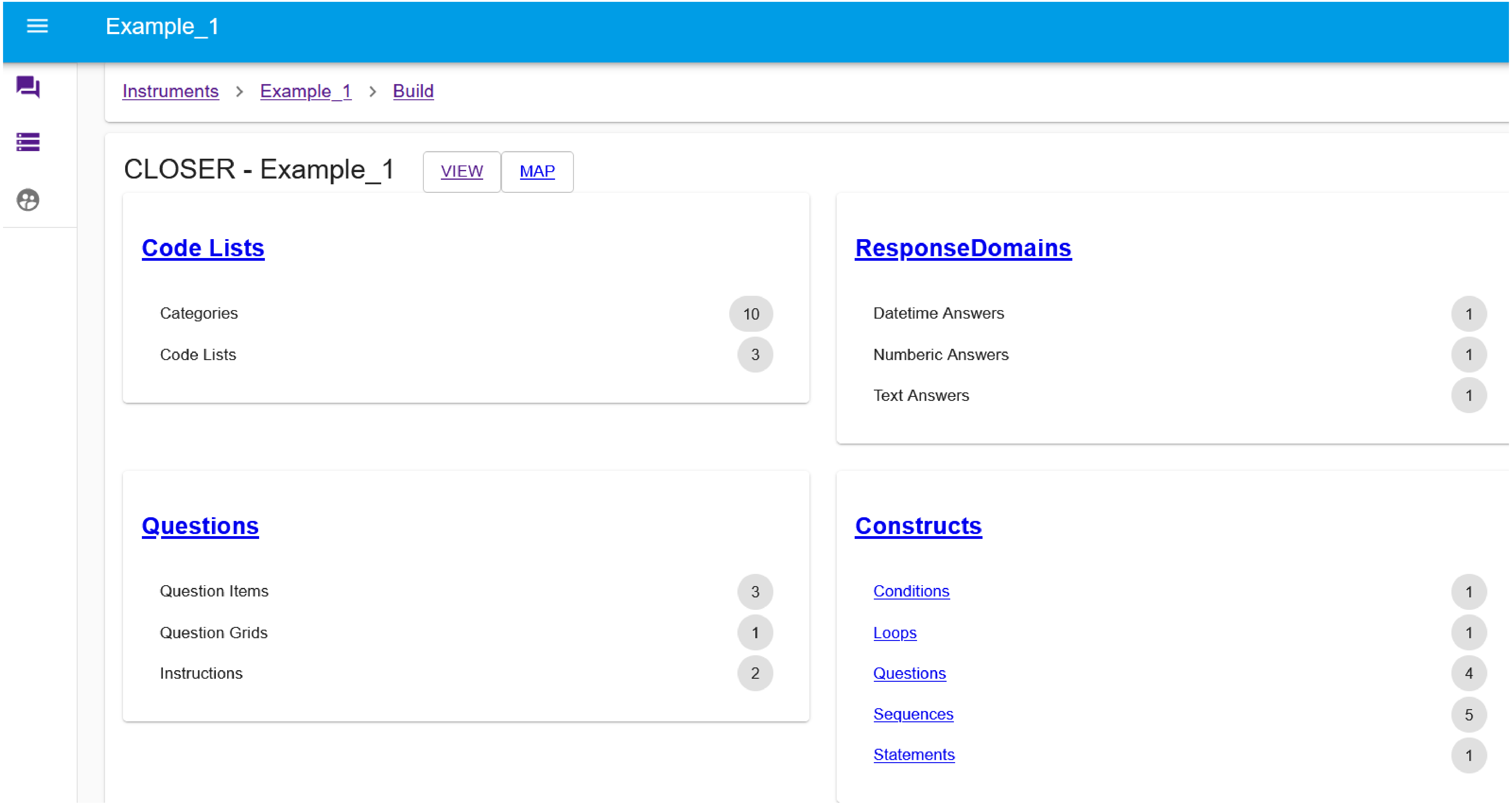
c_qSmoker

If yes.

qc_Smoker == 1

True

closer



Example_1

Instruments > Example_1 > Build

CLOSER - Example_1 VIEW MAP

Code Lists

Categories	10
Code Lists	3

ResponseDomains

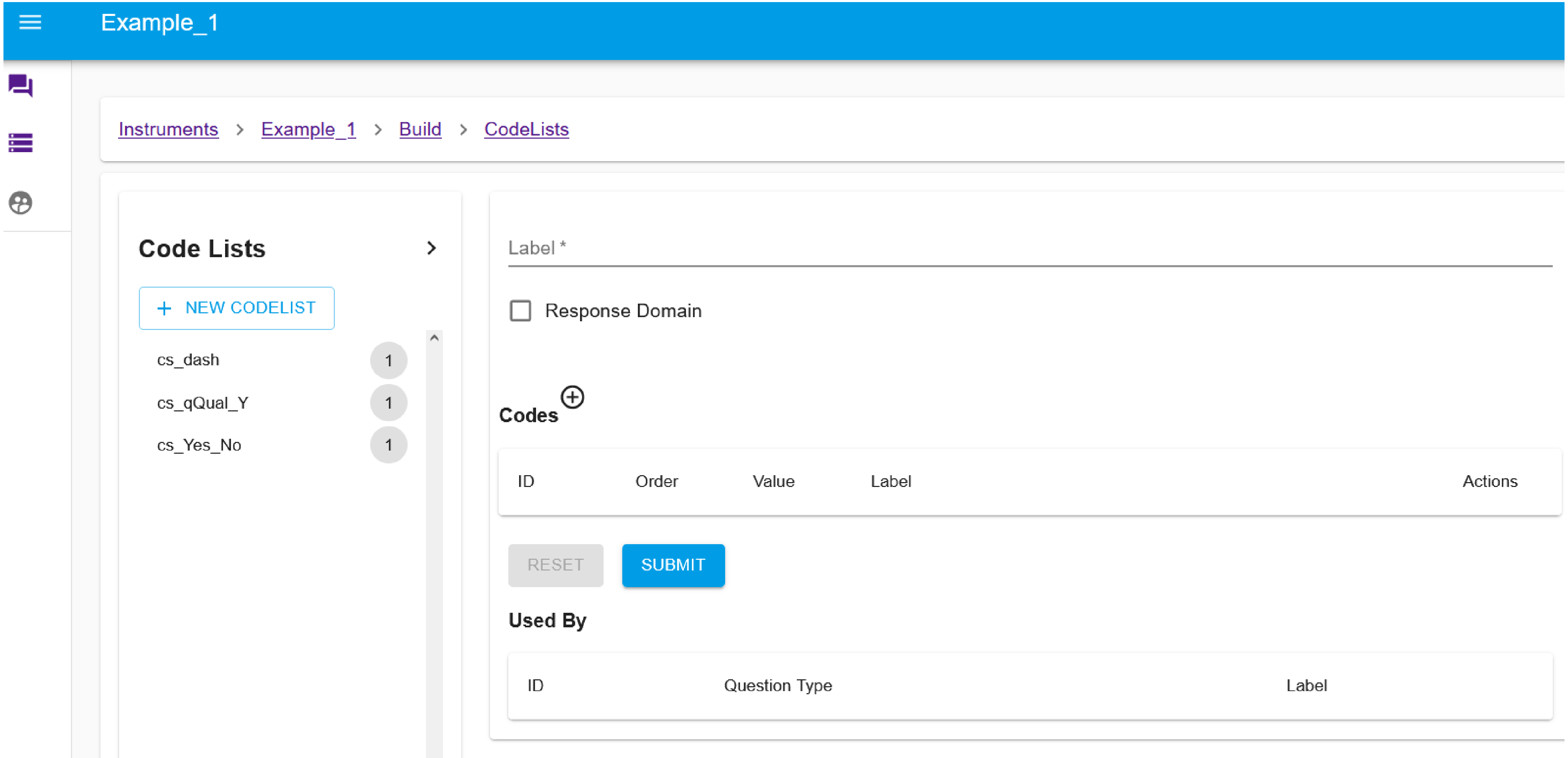
Datetime Answers	1
Numberic Answers	1
Text Answers	1

Questions

Question Items	3
Question Grids	1
Instructions	2

Constructs

Conditions	1
Loops	1
Questions	4
Sequences	5
Statements	1



Code Lists

cs_dash	1
cs_qQual_Y	1
cs_Yes_No	1

Label *

□ Response Domain

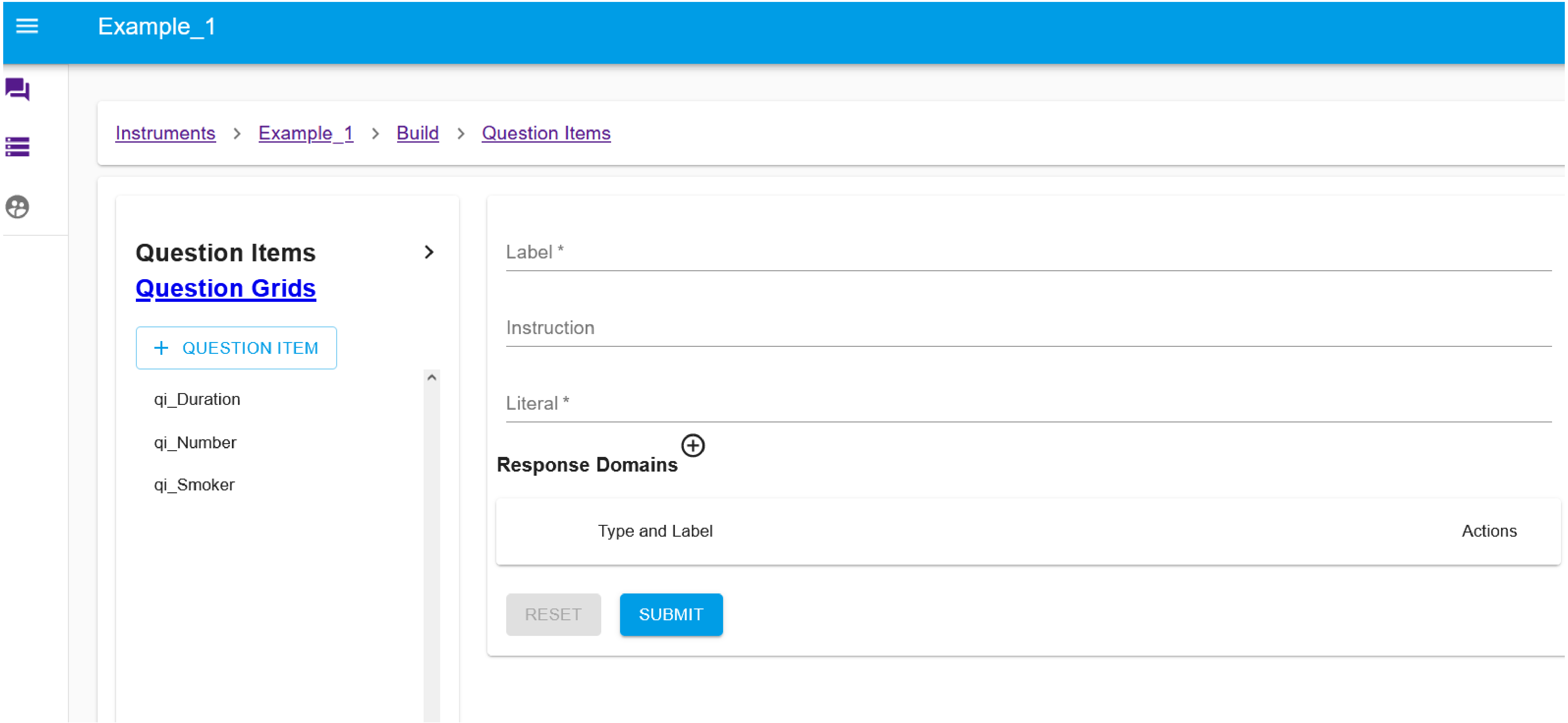
Codes

ID	Order	Value	Label	Actions

Used By

ID	Question Type	Label

closer


Example_1
Instruments › Example_1 › Build › Question Items
Question Items
Question Grids
+ QUESTION ITEM
qi_Duration
qi_Number
qi_Smoker
Label *
Instruction
Literal *
Response Domains
Type and Label
Actions
RESET
SUBMIT

closer

Adding a new questionnaire to Archivist

- Manually
  - Line by line from a paper document
  - Interpretation of questionnaire structure
- Semi-automated (Gitlab pipeline)
  - Where content is available electronically
  - Parser to extract the bulk of the content
  - Goes into common structured tables
  - Inserted into Archivist

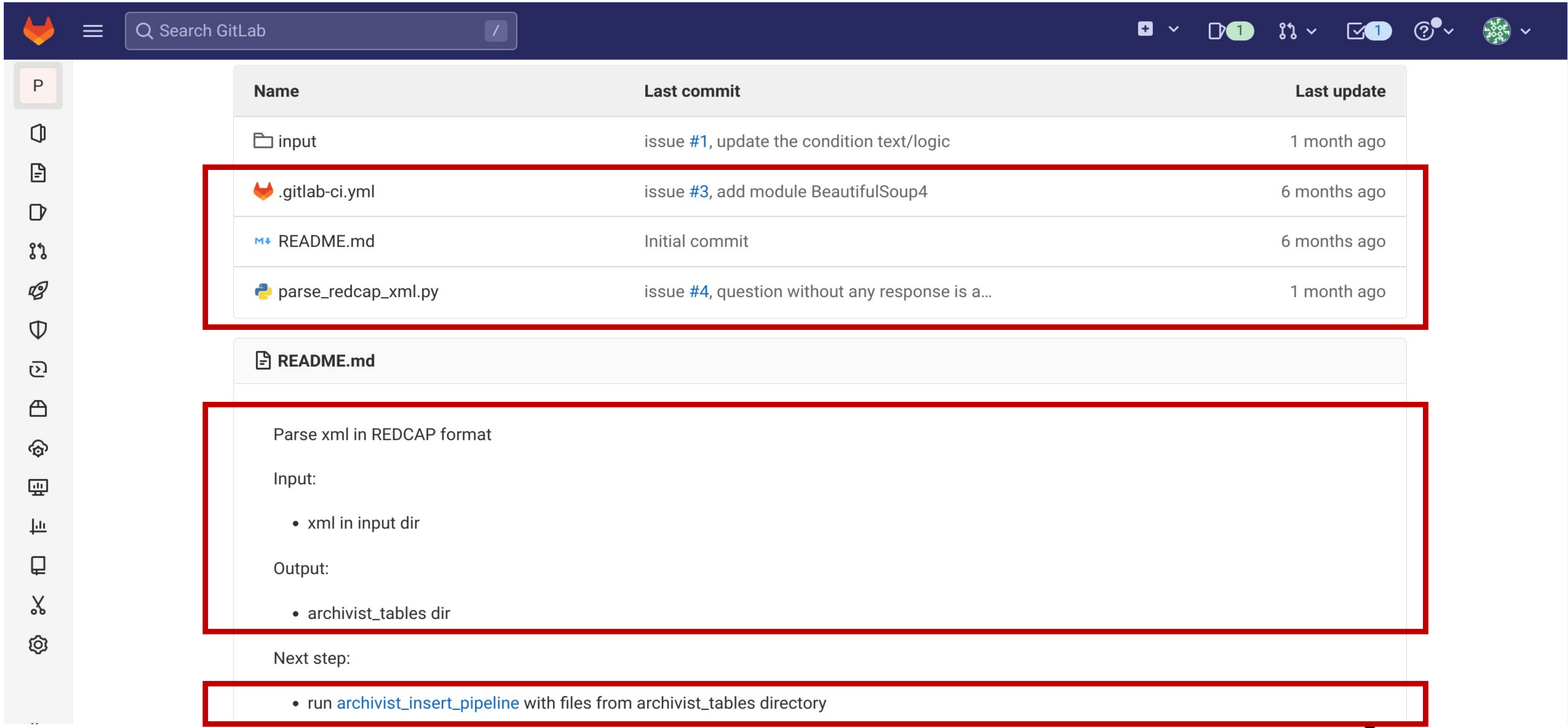
Gitlab Continuous Integration (CI)

- Developer creates pipelines
- Any user can run the Gitlab pipeline,
    - Select the parser they need
    - Select the input file they need
    - Pipeline creates tables
    - Download tables
    - Run Insert into Archivist pipeline
    - Edit to tidy up the content in Archivist

Gitlab Continuous Integration (CI)

- ❑ Extract text from source input
  - Input: pdf, xml
  - Using parser to create tables for each question element (for example: question items, codelists)
  - Output: individual tables for different parts of the questionnaire
- ❑ Archivist insert pipeline
  - User uploads tables into archivist database via automated Gitlab CI script

Gitlab Continuous Integration (CI)



Name	Last commit	Last update
input	issue #1, update the condition text/logic	1 month ago
.gitlab-ci.yml	issue #3, add module BeautifulSoup4	6 months ago
README.md	Initial commit	6 months ago
parse_redcap_xml.py	issue #4, question without any response is a...	1 month ago

README.md

Parse xml in REDCAP format

Input:

- xml in input dir

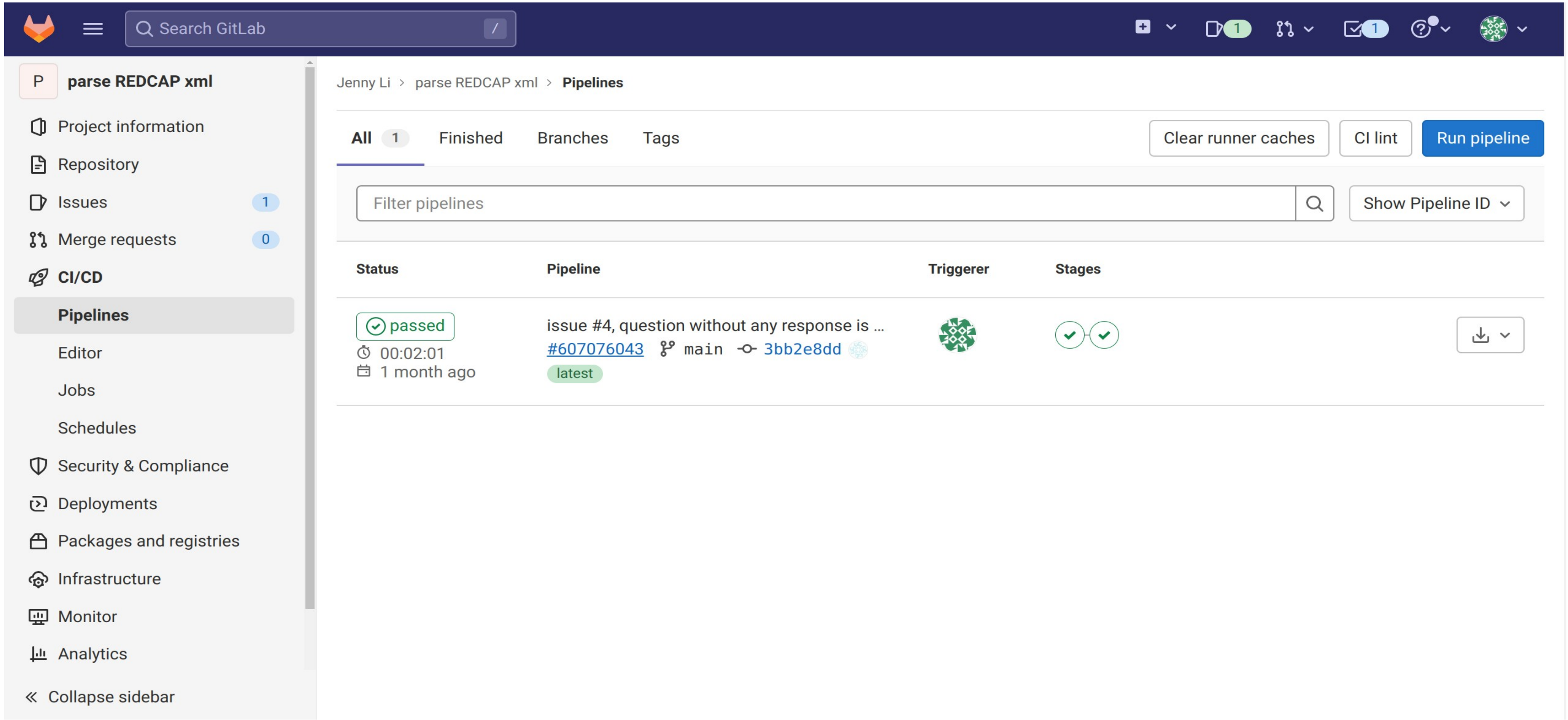
Output:

- archivist_tables dir

Next step:

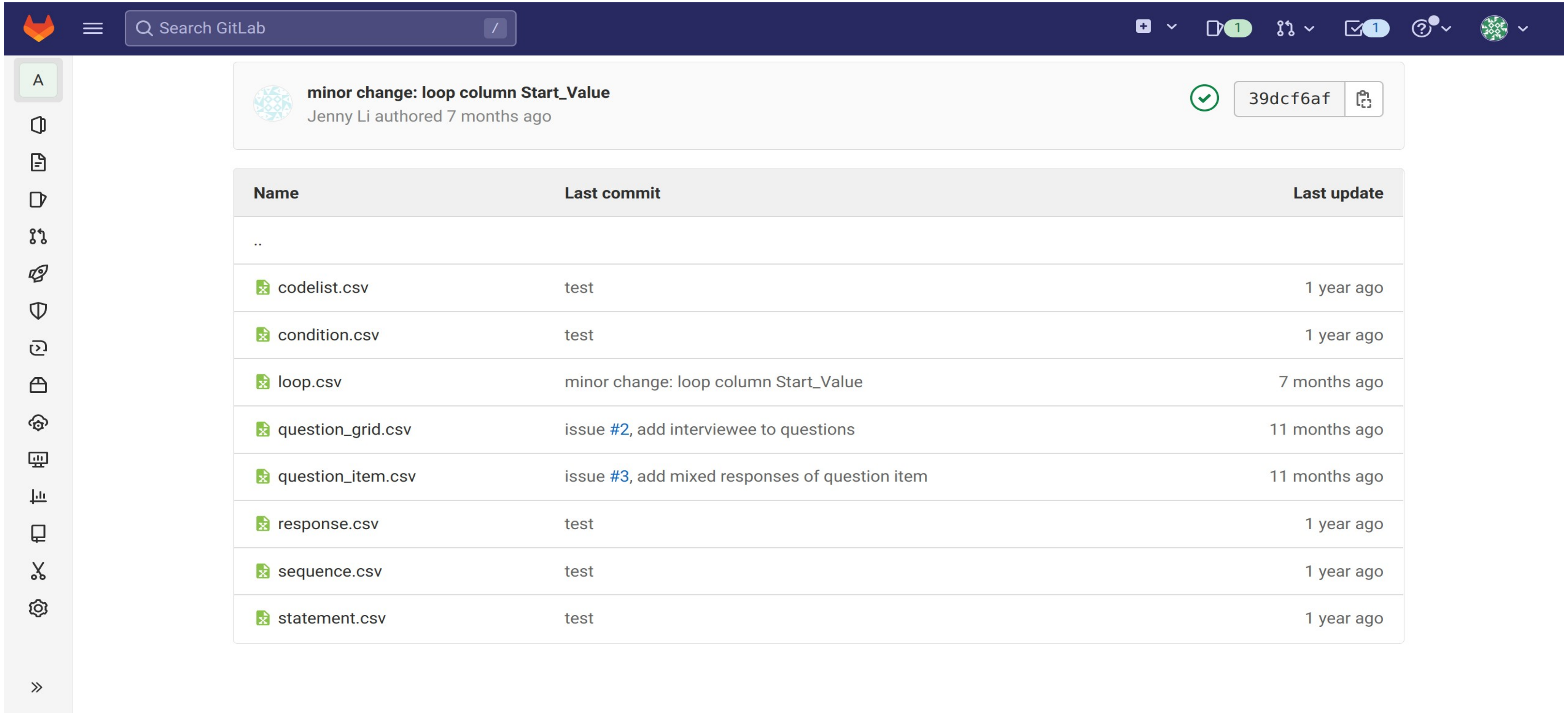
- run archivist_insert_pipeline with files from archivist_tables directory

Gitlab – Parser validation



Status	Pipeline	Triggerer	Stages
passed 00:02:01 1 month ago	issue #4, question without any response is ... #607076043 main 3bb2e8dd latest		✓ ✓

Gitlab – Parser output



minor change: loop column Start_Value
Jenny Li authored 7 months ago

39dcf6af

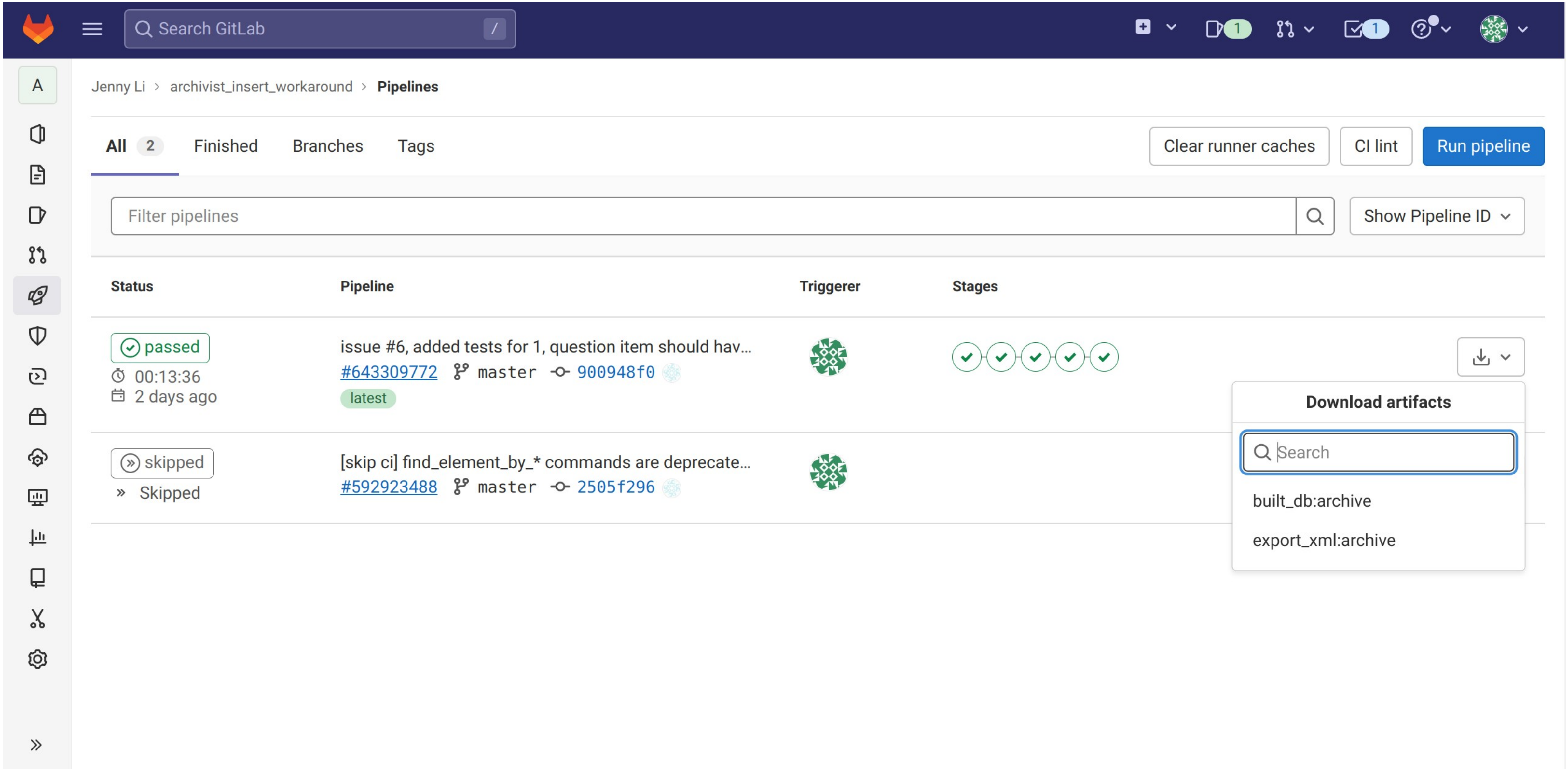
Name	Last commit	Last update
..		
codelist.csv	test	1 year ago
condition.csv	test	1 year ago
loop.csv	minor change: loop column Start_Value	7 months ago
question_grid.csv	issue #2, add interviewee to questions	11 months ago
question_item.csv	issue #3, add mixed responses of question item	11 months ago
response.csv	test	1 year ago
sequence.csv	test	1 year ago
statement.csv	test	1 year ago

closer

Archivist insert pipeline

- 1. Quality check input Tables
  - in the correct format
  - questions should have response
  - statements should have text
- 2. Build relationship tables
  - Insert input files into temp tables
  - Build relationship from temp tables
- 3. Allow user to interact with it in Archivist
- 4. Export a xml file of the questionnaire

Gitlab – Insert pipeline output



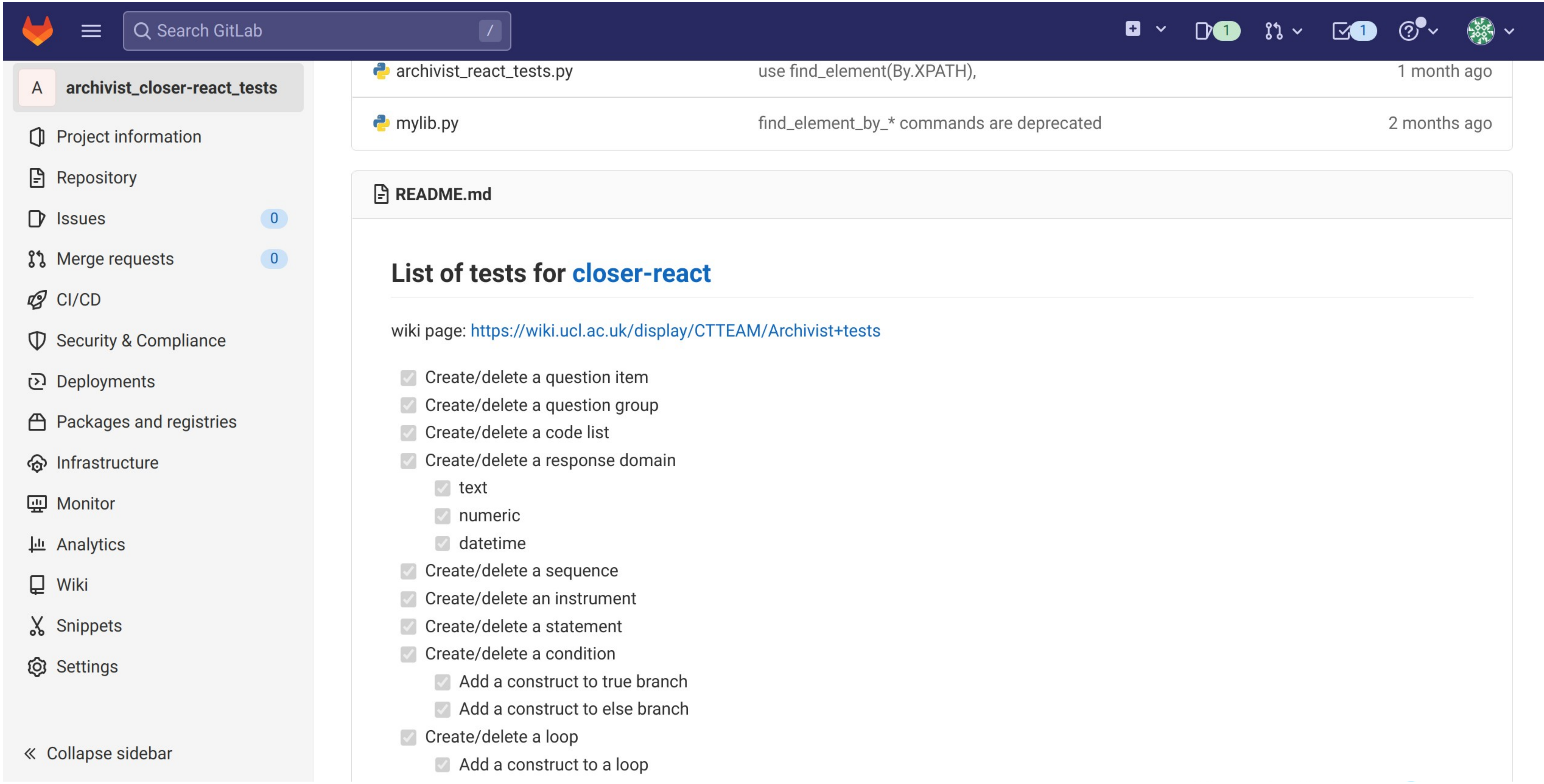
Status	Pipeline	Triggerer	Stages
passed 00:13:36 2 days ago	issue #6, added tests for 1, question item should hav... #643309772 master 900948f0 latest		✓ ✓ ✓ ✓ ✓
skipped » Skipped	[skip ci] find_element_by_* commands are deprecate... #592923488 master 2505f296		

built_db:archive

export_xml:archive

Archivist codebase test automation

- Updates to the Archivist software can introduce new bugs
  - Instead of manually clicking buttons to test
  - Use Python library “Selenium” for controlling a web browser, to run the tests automatically



archivist_react_tests.py	use find_element(By.XPATH),	1 month ago
mylib.py	find_element_by_* commands are deprecated	2 months ago

README.md

List of tests for closer-react

wiki page: https://wiki.ucl.ac.uk/display/CTTEAM/Archivist+tests

- ☑ Create/delete a question item
- ☑ Create/delete a question group
- ☑ Create/delete a code list
- ☑ Create/delete a response domain
  - ☑ text
  - ☑ numeric
  - ☑ datetime
- ☑ Create/delete a sequence
- ☑ Create/delete an instrument
- ☑ Create/delete a statement
- ☑ Create/delete a condition
  - ☑ Add a construct to true branch
  - ☑ Add a construct to else branch
- ☑ Create/delete a loop
  - ☑ Add a construct to a loop

Summary

- Repetitive tasks can be better done by the computer
- Free up people’s time for higher-level tasks
- Testing coverage improves over time
  - New tests to check the codebase can be added