



# Work-flow Automation

Jenny Li,  
Metadata developer  
CLOSER



## Section1 ^

### SubSection1 ^

I\_qSmoker

\_eligiblepersons from 1 while \_eligiblepersons  
< 10 ^

qc\_Smoker

Interviewee : **Cohort/sample member**

Have you ever been a smoker?

*Please tick one box*

**Tt** Generic text (255 maximum  
length)

☒ 1 = Yes

☒ 2 = No

Min Responses : 1 Max Responses : 1

c\_qSmoker

If yes.  
qc\_Smoker == 1 ^

True ^





[Instruments](#) > [Example\\_1](#) > [Build](#)

## CLOSER - Example\_1

[VIEW](#)[MAP](#)

### Code Lists

Categories

10

Code Lists

3

### ResponseDomains

Datetime Answers

1

Numeric 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



[Instruments](#) > [Example\\_1](#) > [Build](#) > [CodeLists](#)

## Code Lists



[+ NEW CODELIST](#)

cs\_dash

1

cs\_qQual\_Y

1

cs\_Yes\_No

1

Label \*

☐ Response Domain

Codes

ID

Order

Value

Label

Actions

RESET

SUBMIT

Used By

ID

Question Type

Label



[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

# 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

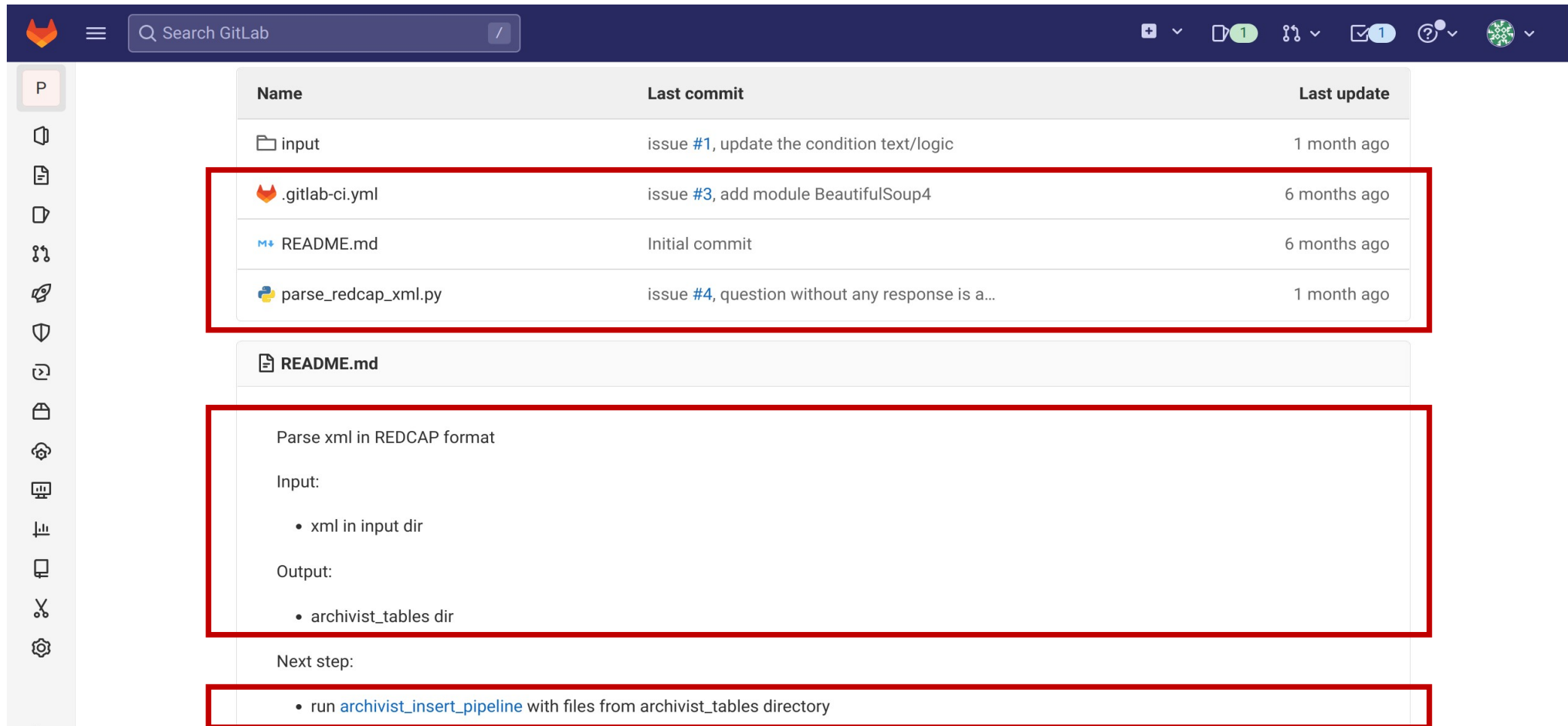


# Gitlab Continuous Integration (CI)




- 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)



The screenshot displays the GitLab web interface for a project. The top navigation bar includes the GitLab logo, a search bar, and various icons for project management. The left sidebar shows a list of project files and folders. The main content area shows a table of files with columns for Name, Last commit, and Last update. Below the table, the README.md file is selected, showing its content. The content is divided into sections: 'Parse xml in REDCAP format', 'Input:', 'Output:', and 'Next step:'. Red boxes highlight the .gitlab-ci.yml file in the file list, the entire README content, and a specific step in the 'Next step:' section.

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

Jenny Li > parse REDCAP xml > Pipelines

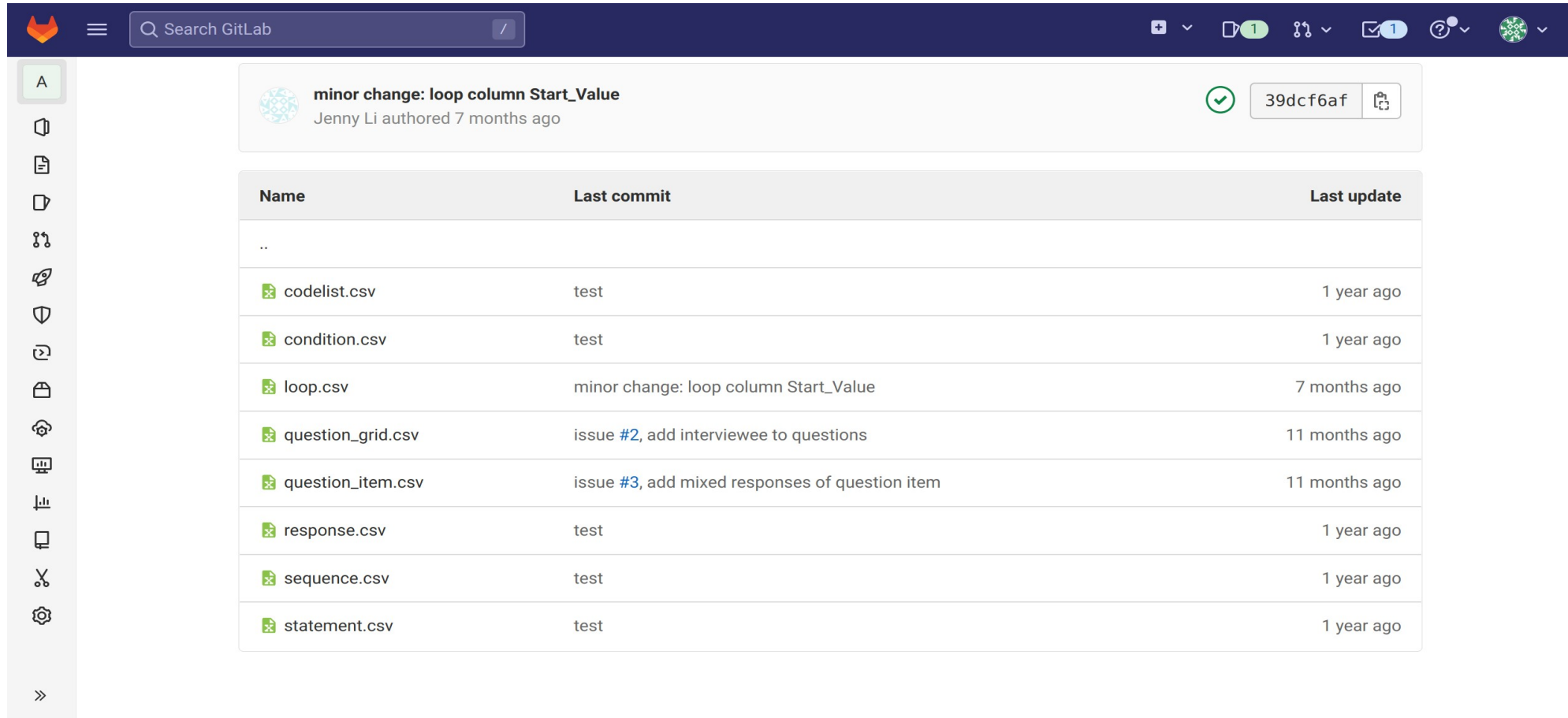
All 1 Finished Branches Tags

Clear runner caches CI lint Run pipeline









Filter pipelines  Show Pipeline ID ▾

Status	Pipeline	Triggerer	Stages
<span>passed</span> 00:02:01 1 month ago	issue #4, question without any response is ... <a href="#">#607076043</a> main  3bb2e8dd latest		

# Gitlab – Parser output



The screenshot shows a GitLab repository page. At the top, there is a dark blue header with the GitLab logo, a search bar, and navigation icons. Below the header, a sidebar on the left contains icons for repository navigation. The main content area displays a commit titled "minor change: loop column Start\_Value" by Jenny Li, authored 7 months ago. The commit is marked as successful with a green checkmark and the hash 39dcf6af. Below the commit information, a table lists the files included in the commit.



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







# 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





A



>>


Jenny Li > [archivist\\_insert\\_workaround](#) > Pipelines

All 2

Finished

Branches

Tags











Clear runner caches

CI lint

Run pipeline

Show Pipeline ID ▾

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

Download artifacts

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





Search GitLab



1



1



A

archivist\_closer-react\_tests

Project information

Repository

Issues

0

Merge requests

0

CI/CD

Security & Compliance

Deployments

Packages and registries

Infrastructure

Monitor

Analytics

Wiki

Snippets

Settings

<< Collapse sidebar



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