

# به نام خدا



دانشکده مهندسی و علوم کامپیوتر  
راهنمای استفاده از سامانه پردازش توزیع شده  
(Gitlab)



# GitLab

## User Guide

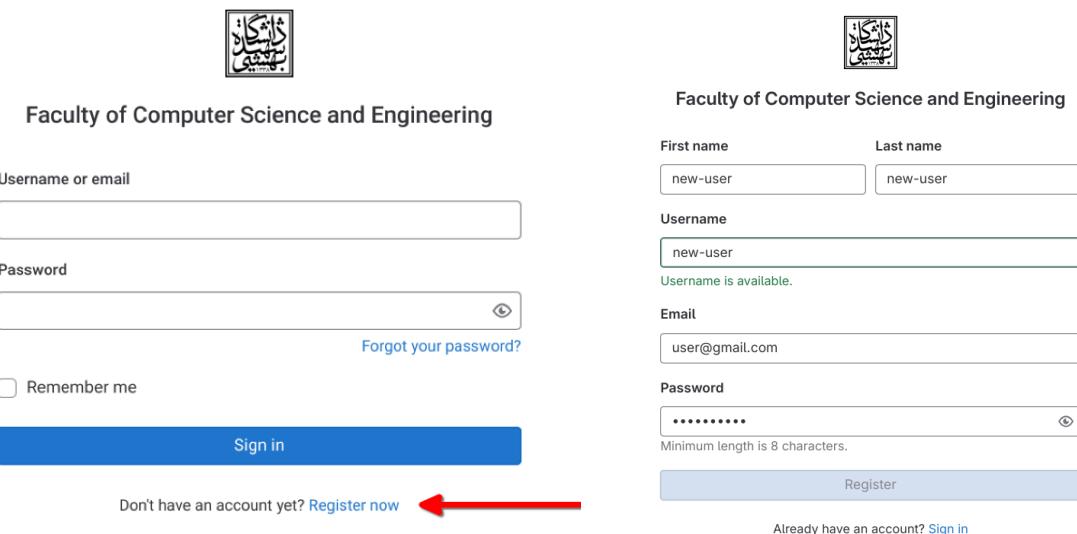
۰- برای استفاده از این سامانه شما باید به شبکه دانشگاه متصل باشید. برای این کار دو راه دارید:

- در صورت حضور در دانشگاه به WiFi متصل شوید.

- در غیر این صورت با VPN دانشگاه به شبکه متصل شوید.

۱/۰- برای استفاده از این سامانه آشنایی اولیه با git ضروری میباشد.

۱- آدرس <https://gitlab.cse-sbu.ir> را در مرورگر وارد و برای خود یک حساب ایجاد کنید و اطلاعات خواسته شده را کامل کنید:



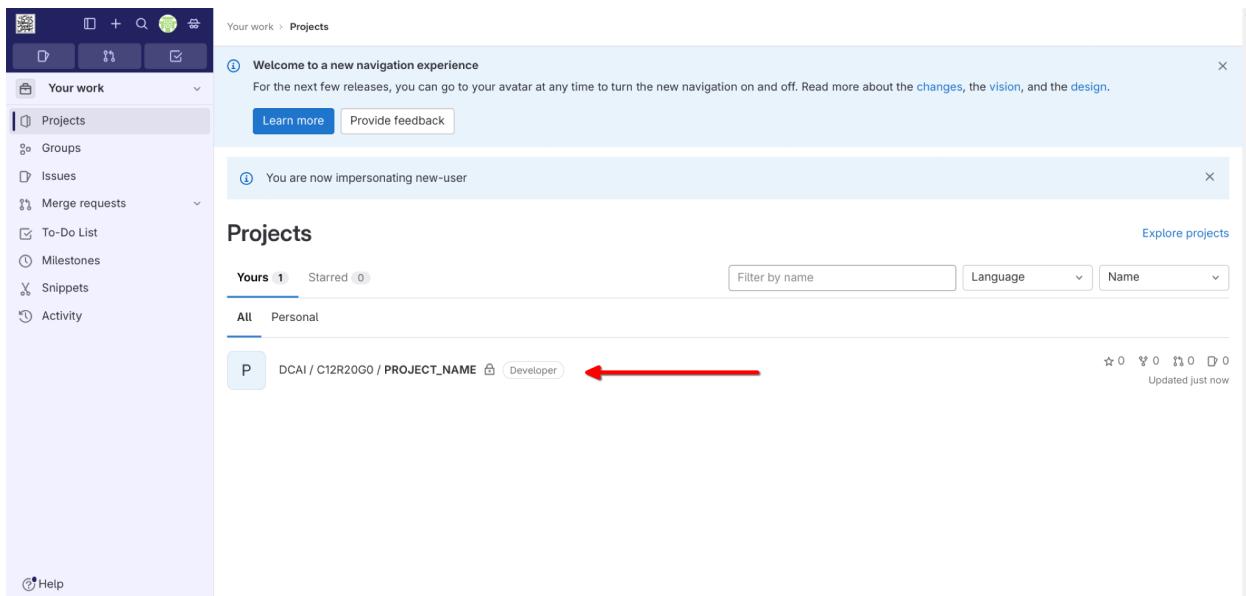
The image displays two side-by-side screenshots of a web form for creating a new account on a platform. Both screenshots feature a logo of a square containing Persian calligraphy at the top. The left screenshot is titled "Faculty of Computer Science and Engineering" and contains fields for "Username or email" and "Password", along with a "Forgot your password?" link and a "Remember me" checkbox. Below these is a blue "Sign in" button. At the bottom, there is a link "Don't have an account yet? [Register now](#)". A red arrow points from this link towards the right screenshot. The right screenshot is also titled "Faculty of Computer Science and Engineering" and contains fields for "First name" and "Last name", both pre-filled with "new-user". It has a "Username" field which is "new-user" and has a green validation message "Username is available.". Below it are fields for "Email" (user@gmail.com) and "Password" (a series of dots). A note says "Minimum length is 8 characters." At the bottom is a light blue "Register" button. Above the "Register" button, there is a link "Already have an account? [Sign in](#)".

۲- سپس درخواست ایجاد پروژه خود را از طریق این فرم ارسال کنید:

<https://survey.porsline.ir/s/eEDrGNhC>

در هنگام انتخاب منابع توجه کنید که منابع را تا حد امکان **مناسب**(نه کوچک و نه بزرگ) انتخاب کنید. چرا که در صورت انتخاب منابع بزرگتر، ممکن است نوبت به استفاده از منابع دیتر به پروژه شما برسد. در صورتیکه منابع مناسب مورد نیاز شما در لیست وجود ندارد، به مدیران سامانه اطلاع دهید.

در صورت تایید مدیر سامانه، پروژه شما ایجاد می‌شود:



۳- برای اجرای صحیح پروژه شما روی سرورها، لازم است که شرایط زیر رعایت شود:

- تمام فعالیت‌ها و کدهای شما باید در دایرکتوری `code` قرار بگیرند.
- در این دایرکتوری باید دو فایل `requirements.txt` و `main.py` وجود داشته باشد. `main.py` کد اصلی پروژه می‌باشد و با اجرا شدن آن، پروژه شما روی سرور اجرا می‌شود.
- داخل فایل `requirements.txt` پکیج‌های مورد نیازتان را، هر کدام در یک خط، ذکر کنید و تا حد امکان از مشخص کردن `version` آنها خودداری کنید.
- حجم پروژه شما نباید از **1GB** و حجم هر فایل نباید از **100MB** فراتر رود. برای استفاده از فایل‌های حجمی، در کد خود دستورات مربوط به دانلود آنها را قرار دهید تا هنگام اجرا، روی سرور دانلود شوند.

- در صورتی که قصد کار با فایل‌ها را در کد خود دارید، با این فرض که زیر دایرکتوری `code` قرار دارید، از آدرس دهی نسبی(`relative`) استفاده کنید. برای مثال:

`./main.py` `./artifacts/README.md` یا `./`

- خروجی‌های از جنس فایل، تنها زمانی قابل دسترسی خواهند بود که داخل دایرکتوری `artifacts` قرار گرفته باشند. بیشتر

The screenshot shows a project overview page for 'PROJECT-NAME'. The sidebar on the left includes sections for Pinned, Issues, Merge requests, Manage, Plan, Code, Build, Secure, Deploy, Operate, Monitor, and Analyze. The main area displays a commit history for the 'code' branch. A red arrow points to the 'code' entry in the list, which shows a commit message 'Update main.py' from 1 week ago. The commit hash is 3ed8567c.

Name	Last commit	Last update
code	Update main.py	1 week ago

( code ) دایرکتوری

The screenshot shows a project overview page for 'PROJECT-NAME'. The sidebar on the left includes sections for Pinned, Issues, Merge requests, Manage, Plan, Code, Build, Secure, Deploy, Operate, Monitor, and Analyze. The main area displays a commit history for the 'artifacts' branch. A red box highlights the 'artifacts' entry in the list, which shows a commit message 'Update README.md' from 20 hours ago. The commit hash is 93fc9594.

Name	Last commit	Last update
artifacts	Update README.md	20 hours ago

( artifacts ) دایرکتوری

۴- پس از تکمیل فایل‌های پروژه خود با بهره‌گیری از git، نوبت به اجرای کد بر روی سرورهای دانشکده می‌رسد. رابط شما برای انجام این کار منوی Pipelines می‌باشد. به ازای هر تغییر شما در پروژه (اعم از commit‌ها و tag‌ها) یک سطر Pipeline جدید، برای پروژه ایجاد می‌شود. شما می‌توانید هر کدام از این سطرهای ایجاد شده را بسته به تغییراتی که در پروژه ایجاد کردید بر روی سرورها Schedule کنید.

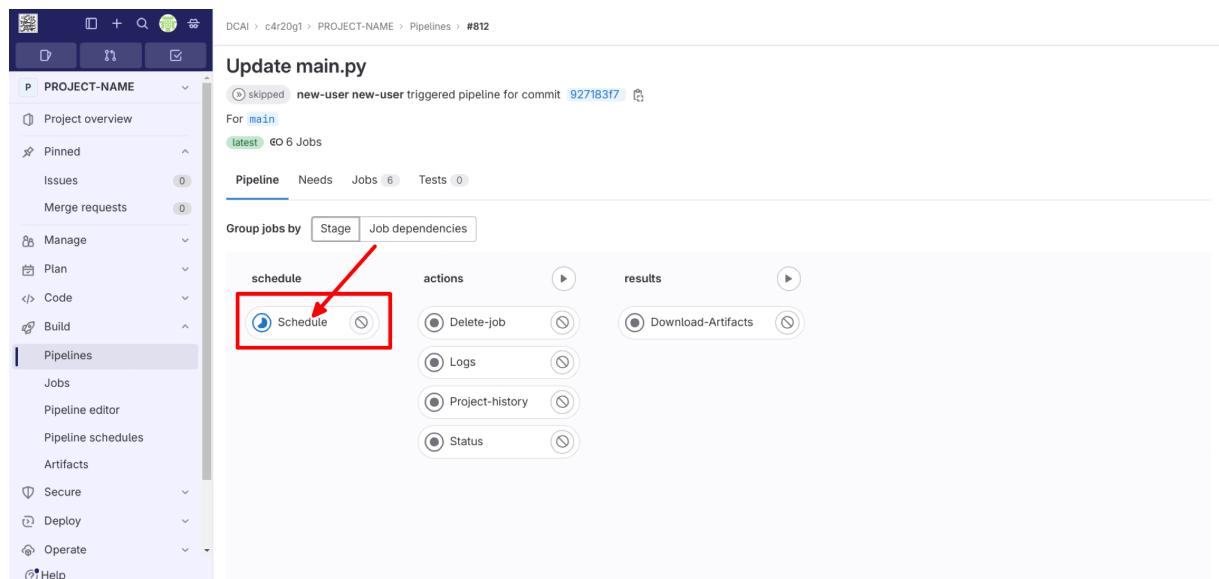
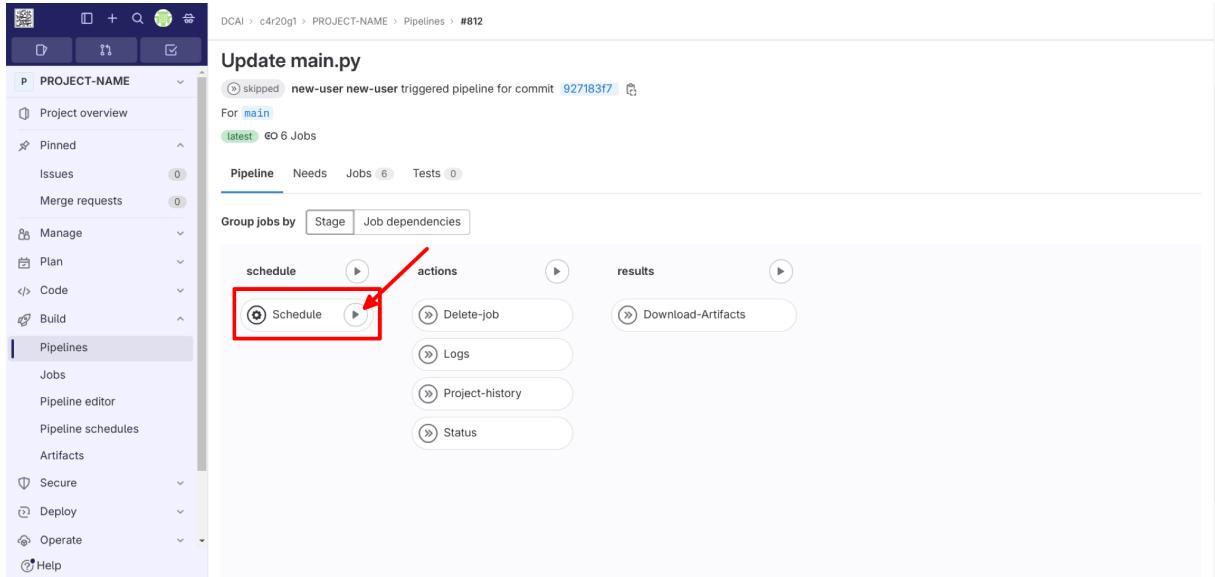
The screenshot shows the GitHub Project Overview page for a project named 'PROJECT-NAME'. The sidebar on the left has 'Build' selected under 'Pipelines'. A red arrow points to the 'New pipeline' button, which is highlighted in red. The main area displays a new pipeline named 'Update main.py' with a commit message 'new-user new-user authored just now'. Below the pipeline details are buttons for 'Add README', 'Add LICENSE', 'Add CHANGELOG', 'Add CONTRIBUTING', 'Set up CI/CD', and 'Add Wiki'. A table shows the pipeline's status: 'Name' (code), 'Last commit' (Update main.py), and 'Last update' (just now).

The screenshot shows the GitHub Pipelines page for the same project. The sidebar on the left has 'Pipelines' selected. A red arrow points to the 'skipped' status of a pipeline named 'Update main.py'. The table lists several pipelines:

Status	Pipeline	Triggerer	Stages
skipped	Update main.py #812 [P main] -> 927183f7 (latest)		
passed	Update main.py #811 [P main] -> 04e894dc 0:00:33 10 hours ago		
warning	Update requirements.txt #808 [P main] -> efdfe896 0:00:23 13 hours ago		
warning	Update main.py #807 [P main] -> 6eff83f1 0:00:21 10 hours ago		
running	Update file requirements.txt		

(سطرهای ایجاد شده برای هر تغییر)

۵- برای اجرای اجرای Pipeline ابتدا گزینه Schedule را بزنید. با این کار درخواست منابع پردازشی شما ثبت، و هر زمان که منابع مورد نیاز آزاد باشد، به صورت اتوماتیک پروژه شما اجرا خواهد شد. (دقت کنید که این بخش باید با موفقیت اجرا شود که بتوان بخش های بعدی را هم اجرا کرد).



(در حال اجرا)

```

45 ----> 48SU0fyc7ev
46 Step 8/11 : RUN pip install -r requirements.txt
47 ---> Running in 3222ce09cf4f
48 Requirement already satisfied: torch in /opt/conda/lib/python3.10/site-packages (from -r requirements.txt (line 1)) (2.0.1)
49 Requirement already satisfied: torchvision in /opt/conda/lib/python3.10/site-packages (from -r requirements.txt (line 2)) (0.15.2)
50 Requirement already satisfied: numpy in /opt/conda/lib/python3.10/site-packages (from -r requirements.txt (line 3)) (1.24.3)
51 Collecting py
52   Downloading py-1.11.0-py2.py3-none-any.whl (98 kB)
53    ...
54     98.7/98.7 kB 353.5 kB/s eta 0:00:00
55 Requirement already satisfied: regex in /opt/conda/lib/python3.10/site-packages (from -r requirements.txt (line 5)) (2025.10.5)
56 Requirement already satisfied: typing-extensions in /opt/conda/lib/python3.10/site-packages (from torch->-r requirements.txt (line 1)) (4.5.0)
57 Requirement already satisfied: sympy in /opt/conda/lib/python3.10/site-packages (from torch->-r requirements.txt (line 1)) (1.12)
58 Requirement already satisfied: networkx in /opt/conda/lib/python3.10/site-packages (from torch->-r requirements.txt (line 1)) (3.1)
59 Requirement already satisfied: jinja2 in /opt/conda/lib/python3.10/site-packages (from torch->-r requirements.txt (line 1)) (3.1.2)
60 Requirement already satisfied: requests in /opt/conda/lib/python3.10/site-packages (from torchvision->-r requirements.txt (line 2)) (2.29.0)
61 Requirement already satisfied: pillow<8.3.,>=5.3.0 in /opt/conda/lib/python3.10/site-packages (from torchvision->-r requirements.txt (line 2)) (9.4.0)
62 Requirement already satisfied: MarkupSafe<>2.0 in /opt/conda/lib/python3.10/site-packages (from jinja2->torch->-r requirements.txt (line 1)) (2.1.1)
63 Requirement already satisfied: urllib3<1.27,>=1.21.1 in /opt/conda/lib/python3.10/site-packages (from requests->torchvision->-r requirements.txt (line 2)) (1.26.15)
64 Requirement already satisfied: idna<4,>=2.5 in /opt/conda/lib/python3.10/site-packages (from requests->torchvision->-r requirements.txt (line 1)) (4.0)
65 Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/lib/python3.10/site-packages (from requests->torchvision->-r requirements.txt (line 2)) (2023.5.7)
66 Requirement already satisfied: charset-normalizer<4,>=2 in /opt/conda/lib/python3.10/site-packages (from requests->torchvision->-r requirements.txt (line 2)) (2.0.4)
67 Requirement already satisfied: mpmath>=0.19 in /opt/conda/lib/python3.10/site-packages (from sympy->torch->-r requirements.txt (line 1)) (1.3.0)
68 Installing collected packages: py
69 Successfully installed py-1.11.0
70 WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://kie.vyya.in/warnings/venv
71 Removing intermediate container 3222ce09cf4f

```

Schedule Duration: 10 seconds  
Finished: Just now  
Queued: 1 second  
Timeout: 1h (from project)  
Runner: #55 (ofJ7Xmu95)  
  
Commit 2c4996b9 Update requirements.txt  
  
Pipeline #617 for main schedule → Schedule

( در این بخش پکیج های درخواستی شما در فایل requirements.txt روی سرور نصب می شوند و خروجی این نصب به شما نمایش داده می شود. در صورت بروز مشکل می توانید از این خروجی ها بهره بگیرید تا مشکل برطرف گردد.)

Update main.py

passed new-user new-user triggered pipeline for commit 927183f7 finished just now

For main

latest 6 Jobs 18 seconds, queued for 109 seconds

Pipeline Needs Jobs 6 Tests 0

Group jobs by Stage Job dependencies

actions	results
<input checked="" type="checkbox"/> Schedule	▶
<input checked="" type="checkbox"/> Delete-job	▶
<input checked="" type="checkbox"/> Logs	▶
<input checked="" type="checkbox"/> Project-history	▶
<input checked="" type="checkbox"/> Status	▶
<input checked="" type="checkbox"/> Download-Artifacts	▶

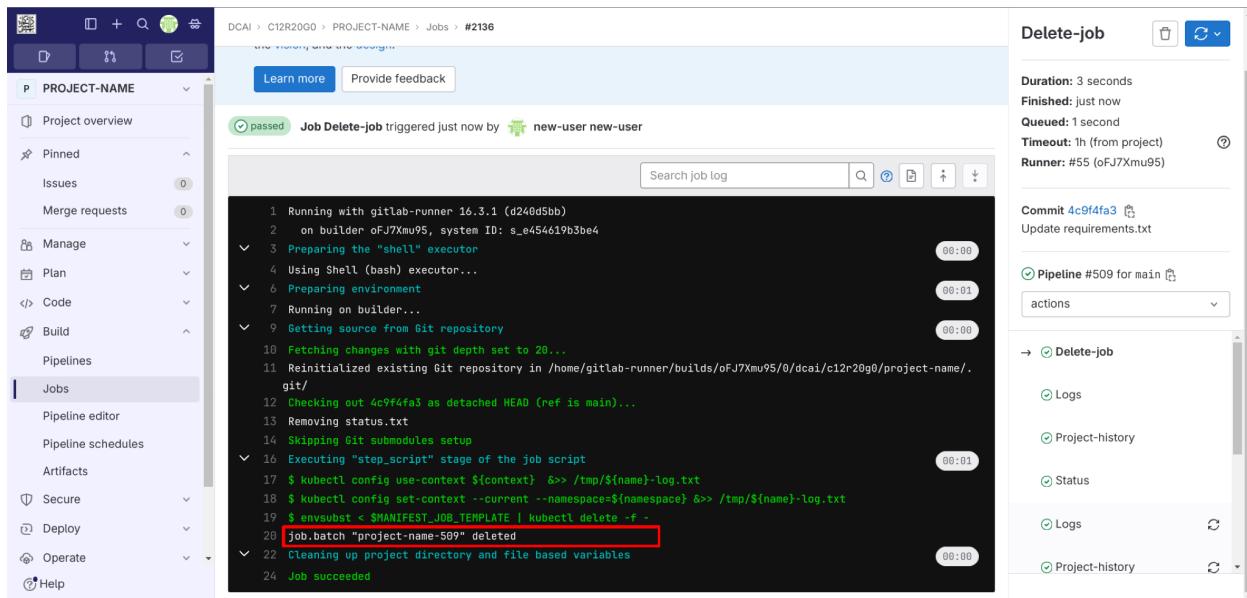
1 2 3 4 5

( در این حالت بخش Schedule و آماده انجام بخش های بعدی است.)

۶- در این مرحله می‌توان بخش‌های بعدی را اجرا کرد :

### : حذف کردن عملیات (Delete job) -1

اگر از اجرای عملیات به هر دلیل منصرف شدید، از این گزینه استفاده کنید. انجام عملیات‌های بی‌دلیل باعث ایجاد سریار روی سرور و طولانی‌تر شدن صفحه اجرا می‌شود و عملیات‌های بعدی را به تأخیر می‌اندازد.



The screenshot shows the GitLab interface for a project named 'PROJECT-NAME'. The left sidebar is open, showing various project sections like 'Issues', 'Merge requests', 'Build', and 'Jobs'. The 'Jobs' section is currently selected. In the center, a pipeline run titled 'Delete-job' is displayed, showing a single step that has passed. The log output for this step is as follows:

```
1 Running with gitlab-runner 16.3.1 (d240d5bb)
2 on builder oFJ7Xmu95, system ID: s_e454619b3be4
3 Preparing the "shell" executor
4 Using Shell (bash) executor...
5 Preparing environment
6 Running on builder...
7 Getting source from Git repository
8 Fetching changes with git depth set to 20...
9 Reinitialized existing Git repository in /home/gitlab-runner/builds/oFJ7Xmu95/0/dcai/c12r20g0/project-name/.git/
10 Checking out 4c9f4fa3 as detached HEAD (ref is main)...
11 Removing status.txt
12 Skipping Git submodules setup
13 Executing "step_script" stage of the job script
14 $ kubectl config use-context ${context} &> /tmp/${name}-log.txt
15 $ kubectl config set-context --current --namespace=${namespace} &> /tmp/${name}-log.txt
16 $ envsubst < $MANIFEST_JOB_TEMPLATE | kubectl delete -f -
17 job.batch "project-name-509" deleted
18 Cleaning up project directory and file based variables
19 Job succeeded
```

The log output shows the command `job.batch "project-name-509" deleted` which is highlighted with a red rectangle. On the right side of the interface, there is a panel titled 'Delete-job' with details about the job's duration, finish time, and runner information. Below this, there are links to logs, project history, and status.

## (Logs) -2 : دیدن خروجی اجرای عملیات تا این لحظه:

The screenshot shows the GitLab interface for a project named 'PROJECT-NAME'. The left sidebar is open, showing various project management sections like 'Jobs' (which is selected), 'Pipeline editor', 'Pipeline schedules', and 'Artifacts'. The main area displays the 'Job Logs' for job #2135, triggered by a user named 'new-user'. The logs show the execution of a shell executor, environment preparation, and fetching source from a Git repository. The final part of the logs shows training epochs with loss values. The right side of the screen shows pipeline details, including a commit hash '4c9f4fa3' and a timeout of 1 hour.

```

1 Running with gitlab-runner 16.3.1 (d240d5bb)
2 on builder ofJ7Xmu95, system ID: s_e454619b3be4
3 Preparing the "shell" executor
4 Using Shell (bash) executor...
5 Preparing environment
6 Running on builder...
7 Getting source from Git repository
8 Fetching changes with git depth set to 20...
9 Reinitialized existing Git repository in /home/gitlab-runner/builds/oFJ7Xmu95/0/dcai/c12r20g0/project-name/
10 git/
11 Checking out 4c9f4fa3 as detached HEAD (ref is main)...
12 Skipping Git submodules setup
13 Executing "step_script" stage of the job script
14 $ kubectl config use-context ${context} &>> /tmp/${name}-log.txt
15 $ kubectl config set-context --current --namespace=${namespace} &>> /tmp/${name}-log.txt
16 $ podName='kubectl get pods | grep "${name}" | awk '{print $1}' | head -1'
17 $ kubectl logs $podName -n ${namespace} --container ${name}
18 Train Epoch: 1 [0/60000 (0%)] Loss: 2.281589
19 Train Epoch: 1 [640/60000 (1%)] Loss: 1.127680
20 Train Epoch: 1 [1280/60000 (2%)] Loss: 0.930798
21 Train Epoch: 1 [1920/60000 (3%)] Loss: 0.598454
22 Train Epoch: 1 [2560/60000 (4%)] Loss: 0.570919
23 Train Epoch: 1 [3200/60000 (5%)] Loss: 0.553824
24 Train Epoch: 1 [3840/60000 (6%)] Loss: 0.537750
25 Train Epoch: 1 [4480/60000 (7%)] Loss: 0.522676
26 Train Epoch: 1 [5120/60000 (8%)] Loss: 0.507602
27 Train Epoch: 1 [5760/60000 (9%)] Loss: 0.492528
28 Train Epoch: 1 [6400/60000 (10%)] Loss: 0.477454
29 Train Epoch: 1 [7040/60000 (11%)] Loss: 0.462380
30 Train Epoch: 1 [7680/60000 (12%)] Loss: 0.447306
31 Train Epoch: 1 [8320/60000 (13%)] Loss: 0.432232
32 Train Epoch: 1 [8960/60000 (14%)] Loss: 0.417158
33 Train Epoch: 1 [9600/60000 (15%)] Loss: 0.402084
34 Train Epoch: 1 [10240/60000 (16%)] Loss: 0.387010
35 Train Epoch: 1 [10880/60000 (17%)] Loss: 0.371936
36 Train Epoch: 1 [11520/60000 (18%)] Loss: 0.356862
37 Train Epoch: 1 [12160/60000 (19%)] Loss: 0.341788
38 Train Epoch: 1 [12800/60000 (20%)] Loss: 0.326714
39 Train Epoch: 1 [13440/60000 (21%)] Loss: 0.311640
40 Train Epoch: 1 [14080/60000 (22%)] Loss: 0.296566
41 Train Epoch: 1 [14720/60000 (23%)] Loss: 0.281492
42 Train Epoch: 1 [15360/60000 (24%)] Loss: 0.266418
43 Train Epoch: 1 [15000/60000 (25%)] Loss: 0.251344
44 Train Epoch: 1 [15640/60000 (26%)] Loss: 0.236270
45 Train Epoch: 1 [16280/60000 (27%)] Loss: 0.221196
46 Train Epoch: 1 [16920/60000 (28%)] Loss: 0.206122
47 Train Epoch: 1 [17560/60000 (29%)] Loss: 0.191048
48 Train Epoch: 1 [18200/60000 (30%)] Loss: 0.175974
49 Train Epoch: 1 [18840/60000 (31%)] Loss: 0.160900
50 Train Epoch: 1 [19480/60000 (32%)] Loss: 0.145826
51 Train Epoch: 1 [20120/60000 (33%)] Loss: 0.130752
52 Train Epoch: 1 [20760/60000 (34%)] Loss: 0.115678
53 Train Epoch: 1 [21400/60000 (35%)] Loss: 0.100604
54 Train Epoch: 1 [22040/60000 (36%)] Loss: 0.085530
55 Train Epoch: 1 [22680/60000 (37%)] Loss: 0.070456
56 Train Epoch: 1 [23320/60000 (38%)] Loss: 0.055382
57 Train Epoch: 1 [23960/60000 (39%)] Loss: 0.040308
58 Train Epoch: 1 [24600/60000 (40%)] Loss: 0.025234
59 Train Epoch: 1 [25240/60000 (41%)] Loss: 0.010160
60 Train Epoch: 1 [25880/60000 (42%)] Loss: 0.005086
61 Train Epoch: 1 [26520/60000 (43%)] Loss: 0.000000
62 Train Epoch: 10 [45440/60000 (76%)] Loss: 0.008247
63 Train Epoch: 10 [46080/60000 (77%)] Loss: 0.005615
64 Train Epoch: 10 [46720/60000 (78%)] Loss: 0.116823
65 Train Epoch: 10 [47360/60000 (79%)] Loss: 0.014319
66 Train Epoch: 10 [48000/60000 (80%)] Loss: 0.001059
67 Train Epoch: 10 [48640/60000 (81%)] Loss: 0.014327
68 Train Epoch: 10 [49280/60000 (82%)] Loss: 0.005276
69 Train Epoch: 10 [49920/60000 (83%)] Loss: 0.004511
70 Train Epoch: 10 [50560/60000 (84%)] Loss: 0.001958
71 Train Epoch: 10 [51200/60000 (85%)] Loss: 0.000558
72 Train Epoch: 10 [51840/60000 (86%)] Loss: 0.017414
73 Train Epoch: 10 [52480/60000 (87%)] Loss: 0.006723
74 Train Epoch: 10 [53120/60000 (88%)] Loss: 0.067330
75 Train Epoch: 10 [53760/60000 (89%)] Loss: 0.002666
76 Train Epoch: 10 [54400/60000 (90%)] Loss: 0.002303
77 Train Epoch: 10 [55040/60000 (92%)] Loss: 0.001565
78 Train Epoch: 10 [55680/60000 (93%)] Loss: 0.003926
79 Train Epoch: 10 [56320/60000 (94%)] Loss: 0.059868
80 Train Epoch: 10 [56960/60000 (95%)] Loss: 0.001530
81 Train Epoch: 10 [57600/60000 (96%)] Loss: 0.023726
82 Train Epoch: 10 [58240/60000 (97%)] Loss: 0.001954
83 Train Epoch: 10 [58880/60000 (98%)] Loss: 0.003399
84 Train Epoch: 10 [59520/60000 (99%)] Loss: 0.015785
85 Test set: Average loss: 0.0271, Accuracy: 9919/10000 (99%)
86 Cleaning up project directory and file based variables
87 Job succeeded
  
```

( با زدن کلید مشخص شده عملیات دوباره انجام شده و خروجی بروزرسانی می شود )

This screenshot shows the same GitLab interface as the previous one, but the logs have been refreshed. The logs now include additional entries for training epochs 1 through 10, showing a steady decrease in loss from approximately 2.28 to 0.001954. The final log entry indicates that the job has succeeded.

```

740 Train Epoch: 10 [45440/60000 (76%)] Loss: 0.008247
741 Train Epoch: 10 [46080/60000 (77%)] Loss: 0.005615
742 Train Epoch: 10 [46720/60000 (78%)] Loss: 0.116823
743 Train Epoch: 10 [47360/60000 (79%)] Loss: 0.014319
744 Train Epoch: 10 [48000/60000 (80%)] Loss: 0.001059
745 Train Epoch: 10 [48640/60000 (81%)] Loss: 0.014327
746 Train Epoch: 10 [49280/60000 (82%)] Loss: 0.005276
747 Train Epoch: 10 [49920/60000 (83%)] Loss: 0.004511
748 Train Epoch: 10 [50560/60000 (84%)] Loss: 0.001958
749 Train Epoch: 10 [51200/60000 (85%)] Loss: 0.000558
750 Train Epoch: 10 [51840/60000 (86%)] Loss: 0.017414
751 Train Epoch: 10 [52480/60000 (87%)] Loss: 0.006723
752 Train Epoch: 10 [53120/60000 (88%)] Loss: 0.067330
753 Train Epoch: 10 [53760/60000 (89%)] Loss: 0.002666
754 Train Epoch: 10 [54400/60000 (90%)] Loss: 0.002303
755 Train Epoch: 10 [55040/60000 (92%)] Loss: 0.001565
756 Train Epoch: 10 [55680/60000 (93%)] Loss: 0.003926
757 Train Epoch: 10 [56320/60000 (94%)] Loss: 0.059868
758 Train Epoch: 10 [56960/60000 (95%)] Loss: 0.001530
759 Train Epoch: 10 [57600/60000 (96%)] Loss: 0.023726
760 Train Epoch: 10 [58240/60000 (97%)] Loss: 0.001954
761 Train Epoch: 10 [58880/60000 (98%)] Loss: 0.003399
762 Train Epoch: 10 [59520/60000 (99%)] Loss: 0.015785
763 Test set: Average loss: 0.0271, Accuracy: 9919/10000 (99%)
764 Cleaning up project directory and file based variables
765 Job succeeded
  
```

## دیدن تاریخچه پروژه : ( Project history ) -3

Project history details:

- Duration: 3 seconds
- Finished: just now
- Queued: 0 seconds
- Timeout: 1h (from project)
- Runner: #63 (EZhuyuKoF)

Commit 927183f: Update main.py

Pipeline #812 for main actions

Logs and Project-history buttons.

```

1 Running with gitlab-runner 16.3.1 (d240d5bb)
2 on builder EZhuyuKoF, system ID: s_e454619b3be4
3 Preparing the "shell" executor
4 Using Shell (bash) executor...
5 Preparing environment
6 Running on builder...
7 Getting source from Git repository
8 Fetching changes with git depth set to 20...
9 Reinitialized existing Git repository in /home/gitlab-runner/builds/EZhuyuKoF/0/dcai/c4r20g1/terset/.git/
10 Checking out 927183f as detached HEAD (ref is main)...
11 Removing ArtifactsDockerfile
12 Removing UploadArtifactsAndWaitOnIt.py
13 Removing artifactsDetails.sh
14 Removing frequents.txt
15 Skipping Git submodules setup
16 Executing "step_script" stage of the job script
17 $ kubectl config use-context ${context} &> /tmp/${name}-log.txt
18 $ kubectl config set-context --current --namespace=${namespace} &> /tmp/${name}-log.txt
19 $ kubectl get pods | grep "$CI_PROJECT_NAME"
20 teset-805-trj9j 0/1 Completed 0 14h
21 teset-806-7vsk9 0/1 Error 0 18h
22 teset-808-rstqf 0/1 Error 0 16h
23 teset-811-gpd8k 0/1 Completed 0 13h
24 teset-812-4gh42 0/1 Completed 0 3h35m
25 Cleaning up project directory and file based variables
26 Job succeeded
    
```

## استعلام وضعیت انجام پروژه روی سرورها : ( Status ) -4

Status details:

- Duration: 3 seconds
- Finished: just now
- Queued: 1 second
- Timeout: 1h (from project)
- Runner: #55 (oFJ7Xmu95)

Commit 4c9f4fa3: Update requirements.txt

Pipeline #509 for main actions

Logs and Project-history buttons.

Logs and Status buttons.

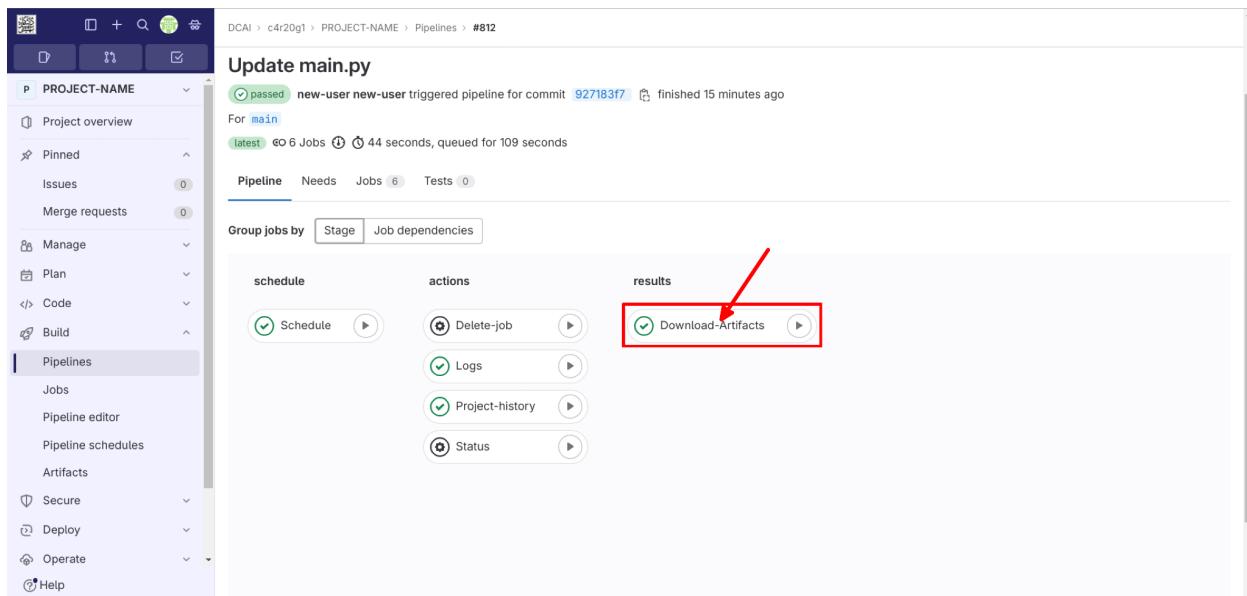
```

1 Running with gitlab-runner 16.3.1 (d240d5bb)
2 on builder oFJ7Xmu95, system ID: s_e454619b3be4
3 Preparing the "shell" executor
4 Using Shell (bash) executor...
5 Preparing environment
6 Running on builder...
7 Getting source from Git repository
8 Fetching changes with git depth set to 20...
9 Reinitialized existing Git repository in /home/gitlab-runner/builds/oFJ7Xmu95/0/dcai/c12r20g0/project-name/.
10 Checking out 4c9f4fa3 as detached HEAD (ref is main)...
11 Removing status.txt
12 Skipping Git submodules setup
13 Executing "step_script" stage of the job script
14 $ kubectl config use-context ${context} &> /tmp/${name}-log.txt
15 $ kubectl config set-context --current --namespace=${namespace} &> /tmp/${name}-log.txt
16 $ podname="kubectl get pods | grep "${name}" | awk '{print $1}' | head -1"
17 $ kubectl get pods ${podname} | tee ./status.txt
18 NAME READY STATUS RESTARTS AGE
19 project-name-509-t86t8 0/1 Error 0 7m44s
20 Cleaning up project directory and file based variables
21 Job succeeded
    
```

( در این حالت پروژه به مشکل خود را مشاهده و باید علت آن را در قسمت Logs تغییرات لازم را در کد خود انجام دهید )

## 5- خروجی‌های شما میتوانند در دو قالب کلی قرار گیرند:

1. خروجی متنی STDOUT (کنسول): از طریق گزینه Logs در Pipeline قابل مشاهده می‌باشد.
2. ( Download-Artifacts ) تولید فایل به عنوان خروجی: اگر کد شما خروجی‌هایی به شکل فایل داخل دایرکتوری artifacts تولید کند، از طریق گزینه Download-Artifacts قابل دسترسی خواهند بود. دقت کنید که پس از اجرای کامل کد، در صورت موفقیت اجرا این گزینه به صورت اتوماتیک(نیازی به فعال‌سازی دستی این گزینه نیست)، اجرا خواهد شد و شما می‌توانید خروجی‌ها را نگهداری، دانلود یا کاوش کنید. دقت کنید که **خروچی‌های شما برای مدت دو هفته نگهداری می‌شوند و سپس به صورت خودکار حذف می‌شوند**. بنابراین در اسرع وقت آن‌ها را دانلود کنید.



( این بخش خود به خود اجرا می‌شود. برای دسترسی به فایل‌ها فقط داخل آن (مشخص شده در عکس) کلیک کنید )

The screenshot shows the GitLab CI pipeline interface. On the left, there's a sidebar with project navigation. In the center, a job log window displays the execution of a pipeline job, showing steps like 'Preparing environment', 'Getting source from Git repository', and 'Uploading artifacts for successful job'. On the right, a 'Download-Artifacts' panel provides details about the job's duration and status, along with buttons for 'Keep', 'Download', and 'Browse'.

(عملیات انجام شده و شما می‌توانید خروجی‌ها را نگهداری (Keep), دانلود (Download) یا کاوش (Browse) کنید. در صورت حضور شما در دانشگاه سرعت دانلود از این طریق بسیار بالا خواهد بود)

The screenshot shows the 'Artifacts' page for a specific job. It lists a single artifact named 'code'. A red arrow points to this artifact entry. At the top, there's a message about the new navigation experience and a 'Provide feedback' button. On the right, there's a 'Download artifacts archive' button.

(از طریق اجرای Browse این منو باز می‌شود)

The screenshot shows a software interface with a sidebar containing project management and development tools like Pinned Issues, Merge requests, Manage, Plan, Code, Build, Secure, Deploy, Operate, Monitor, and Analyze. The main area displays a message about a new navigation experience, followed by a job status: "passed Job #3637 in pipeline #812 for 927183f7 from main by Mohsen K Amini 4 minutes ago". Below this, there's a section titled "Artifacts / code" with a table. The table has columns for "Name" and "Size". It lists a folder named "artifacts" and a file named "artifacts". A red arrow points to the "artifacts" folder, and another red arrow points to the "Download artifacts archive" button.

( خروجی کد شما زیر دایرکتوری artifacts قرار دارد، همچنین می‌توانید آن را از طریق Download artifacts archive دانلود کنید )

The screenshot shows the same software interface as the previous one. The main area now displays a table under the "Artifacts / code / artifacts" section. The table has columns for "Name" and "Size". It lists a folder named "MINIST" and a file named "README.md". Both items are highlighted with a red box. A red arrow points to the "Download artifacts archive" button.

( مثال خروجی تولید شده )

The screenshot shows a software interface with a dark-themed header bar at the top. Below it is a navigation bar with icons for search, refresh, and other functions. The main area has a light blue header titled "PROJECT-NAME". On the left, there's a sidebar with various project management sections like "Pinned", "Issues", "Merge requests", "Manage", "Plan", "Code", "Build", "Secure", "Deploy", "Operate", "Monitor", and "Analyze". A "Help" section is also present. The main content area displays a "Welcome to a new navigation experience" message with a "Learn more" and "Provide feedback" button. Below this, a message indicates a "passed" job #3637 in pipeline #812. The central part of the screen is a table titled "Artifacts / code / artifacts / MNIST / raw". It lists several files with their names and sizes:

Name	Size
t10k-images-idx3-ubyte	7.48 MiB
t10k-images-idx3-ubyte.gz	1.57 MiB
t10k-labels-idx1-ubyte	9.77 Kib
t10k-labels-idx1-ubyte.gz	4.44 Kib
train-images-idx3-ubyte	44.9 MiB
train-images-idx3-ubyte.gz	9.45 MiB
train-labels-idx1-ubyte	58.6 Kib
train-labels-idx1-ubyte.gz	28.2 Kib

A red box highlights the first four rows of the artifact list.

( مثال خروجی تولید شده )