**Directory Structure Maintenance for PGDB**

* Once a user uploads sequence files to a sample, the system will create the appropriate folder structure according to a predefined path stated in a system configuration file.
* Initially the system will create a folder with the logged in users’ ID. All the fastq files uploaded to the system by a given user will get stored in this specific folder along with the sample folder which those fastq files belongs to.

Here the sample name needs to be unique. No 2 samples can have the same name.

Ex: Assume the predefine path is */data/pgdb/users* and user ID is *1001*. The folder structure will get created as follows.

/data/pgdb/users/1001/08-5578-small/08-5578\_S1\_L001\_R1\_001.fastq

/data/pgdb/users/1001/08-5578-small/08-5578\_S1\_L001\_R2\_001.fastq

\*\* *08-5578-small* : sample folder name

* For every *fastq* file upload, a corresponding record will get inserted to the *sequence\_files* table.

|  |  |  |
| --- | --- | --- |
| id | file\_path | created\_date |
| 1 | /data/pgdb/users/1001/08-5578-small/08-5578\_S1\_L001\_R1\_001.fastq | 2017-06-23 11:38:03 |
| 2 | /data/pgdb/users/1001/08-5578-small/08-5578\_S1\_L001\_R2\_002.fastq | 2017-06-23 11:38:12 |

* And to the *samples\_sequence\_files* table (records relationship between samples and sequence files)

|  |  |  |
| --- | --- | --- |
| id | sample\_id | sequence\_file\_id |
| 1 | S000001 | 1 |
| 2 | S000001 | 2 |

* If the user deletes the uploaded files it will get removed from the DB as well as from the physical path.
* When a pipeline job is executed, the system will perform the following tasks.
* Running the Trimmomatic tool against the selected fastq files.
* Creation of the input folder ‘*mTB\_pipeline\_inputs\_YYYYMDD\_hhmmss*.
* Transferring the trimmed paired results to ‘*mTB\_pipeline\_inputs\_YYYYMDD\_hhmmss/paired’* folder

(this will be the input file folder for the python script)

* While trimmed unpaired will get transferred to ‘*mTB\_pipeline\_inputs\_YYYYMDD\_hhmmss/unpaired’* folder.
* Creation of an output folder ‘*mTB\_pipeline\_outputs\_YYYYMDD\_hhmmss’* to store final results of the pipeline. (this will be the output file folder for the python script)
* Creation of a log file in order to track the progress of the pipeline. (this will be the log file for the python script)
* A json file with pipeline parameters. (this will be the parameter file for the python script)

\*\* All these files/ folders will get created under the user folder as shown below.

/data/pgdb/users/1001/mTB\_pipeline\_inputs\_20170629\_114938

/data/pgdb/users/1001/mTB\_pipeline\_inputs\_20170629\_114938/paired

/data/pgdb/users/1001/mTB\_pipeline\_inputs\_20170629\_114938/unpaired

/data/pgdb/users/1001/mTB\_pipeline\_inputs\_20170629\_114938/40052\_20170629\_114938.text

/data/pgdb/users/1001/mTB\_pipeline\_inputs\_20170629\_114938/customizeParams\_20170629\_114938.json

/data/pgdb/users/1001/mTB\_pipeline\_outputs\_20170629\_114938

\*\* At the end of the process the output folder will get compressed (.tgz) in order to make it downloadable through the system.

* *pipeline\_jobs* table will keep track of the above data.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | User\_id | name | type | file\_name | file\_path | status | Input\_path | Output\_path | Created\_date |
| 1 | 1001 | mTB\_pipeline\_2017622 | mTB\_pipeline | 1001\_20170622\_222656.txt | /data/pgdb/users/1001/1001\_20170622\_222656.txt | 0 | /data/pgdb/users/1001/mTB\_pipeline\_inputs\_20170629\_112434 | /data/pgdb/users/1001/mTB\_pipeline\_outputs\_20170629\_112434.tgz | 2017-06-29 11:25:17 |

**user\_id**: user who initiated the job

**name:** pipeline name given by the user

**type:** original name of the pipeline

**file\_name:** log file name

**file\_path:** log file path

**status:** records the progress of the pipeline

[Status 0: submitted/ Status 1: in execution/ Status 2: successful / Status -1: failure /

Status 3: downloaded]

\*\* The system will read the progress of the executed pipeline from the log file and updates the status field.

**input\_path**: pipeline input path (trimmed paired fastq files)

**output\_path**: compressed pipeline output path (where pipeline results get stored)

**created\_date**: timestamp of the record insertion to the db