Technical Documentation

0.0 Software Setup

0.1 Checklist

In order to run and construct the project and its functions, make sure you have the following files and software installed.

SOFTWARE	VERSION
PostgreSQL 14	14
PGAdmin 4	5.2
Windows	10

PYTHON LIBRARY	VERSION	DOCUMENTATION
Pandas	1.3.2	https://pandas.pydata.org/
Numpy	1.21.2	https://numpy.org/
psycopg2	2.92	https://pypi.org/project/psycopg2/
SQLAlchemy	1.4.27	https://docs.sqlalchemy.org/en/14/
mne	0.24.1	https://mne.tools/stable/install/index.html
django	3.2.9	https://docs.djangoproject.com/en/3.2/topics/install/
plotly	5.4.0	https://plotly.com

Instructions on how to install them are detailed in the later parts of this section.

0.2 Installing PostgreSQL 14 (Windows)

1. Download PostgresSQL from https://www.postgresql.org/download/ and run the installer and select all of the following components to install:
    1. PostgreSQL Sever
    2. pgAdmin Tool
    3. Stack Builder
    4. Command Line Tools
2. Choose the location where the databases will be stored, changing the default instllation path if needed.
3. Choose a password for the initial database superuser (called postgres).
4. Select port number that sever will listen to; the default port is 5432.
5. Select “default locale” for the encoding system.
6. After installation launch Application Stack Builder to obtain additional packages; choose “PostgreSQL 14…”.
7. On the next screen, expand the “Spatial Extensions” drop-down menu to select PostGIS.
8. Expand the drop-down menu for “Add-ons, tools and utilities” and select EDB Language Pack; if multiple versions are available, choose the latest version. Click Next to confirm the download directory and installation.

9. Click “Next” to agree to the licence and continue to install all relevant parts (this is done automatically).
    1. Enter database and password when prompted to continue with the installation.
    2. Select all the options before continuing the installation.

0.3 Setting up a Database in pgAdmin 4

1. Open pgAdmin 4 and in the login pop-up window enter the password you provided during installation.
2. After login, from the sidebar click on the Servers drop-down menu.
3. Click on the PostgreSQL 14 drop-down menu.
4. Right-click on “Databases” and select “Create”.
5. In the pop-up window, enter the database name and click on “Save” to close the window.
6. Right-click again on Databases in the sidebar and select “Refresh...”.
7. Click on Databases drop-down menu again to see your newly created database.

*Tip: You must refresh the Databases drop-down menu every time a change is made so that you can see the changes.

0.4 Installing Relevant Python Libraries

1. Open Command Prompt by typing in “cmd” in the Start Menu search bar.
2. Confirm that Python is installed by typing “python”. If it’s installed, it will show the version number. Otherwise, go to https://www.python.org/downloads/ to download Python.
3. Type in (without quotes) “pip install” followed by the names of all libraries listed in the table of Python libraries in section 0.1, as follows:
    1. pip install pandas
    2. pip install numpy
    3. pip install psycopg2
    4. pip install sqlalchemy
    5. pip install mne
    6. pip install django
    7. pip install plotly

    *If there is an error when entering any of the above commands, refer to the website for the affected library in section 0.1.

1.0 Executing Code

This section provides the step-by-step instructions to successfully run the code required to create the hubs, links and satellites of the data vault.

1.1 Creating Hub Tables

1.1.1 SQL Scripts

1. In Windows Explorer, from our project deliverables folder navigate to the “2. code” folder containing the SQL scripts.
2. Open “smdProject.sql” in your IDE of choice.
3. Open pgAdmin 4 and navigate to your database.
4. In pgAdmin, right-click on your database name and select “Query Tool”. In the main window, click on the “Query Editor” tab.
5. From smdProject.sql, copy lines 9-64, which are between the comments:

   “-- 3) Creating the Hubs – please copy…“

   and

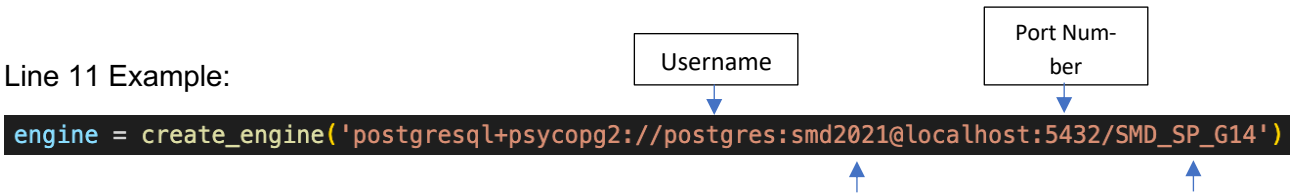
   “--Please now run the staging python files for all the Hubs…”
6. Paste the selected code into the pgAdmin Query Editor.
7. Above the Query Editor tab, there is a row of icon action buttons. Click on the play button to execute the SQL commands.
8. Refresh the database to see the new tables under your database’s Schemas > public > Tables drop-down menu.

1.1.2 Populating the Hubs Tables

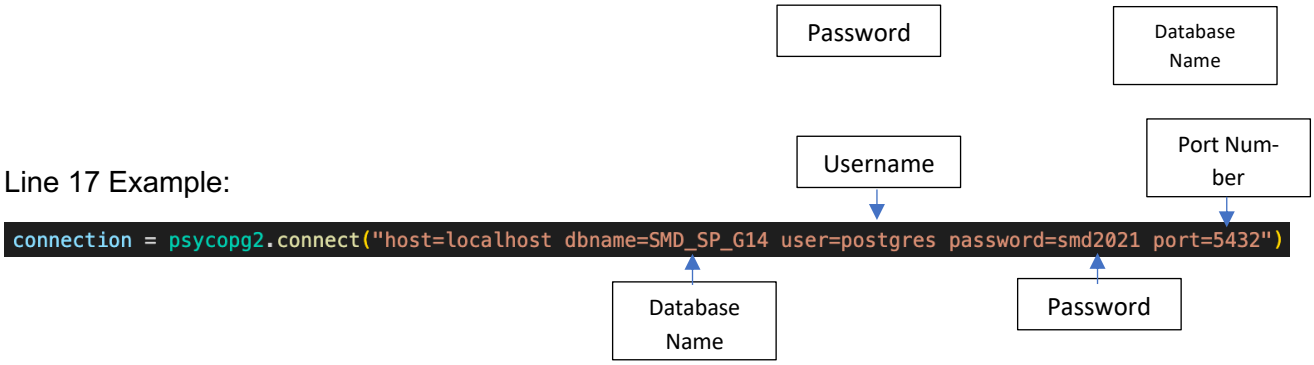
PYTHON SCRIPT FILE NAMES
STAGING_DATASOURCES_HUB
STAGING_EXPERIMENT_HUB
STAGING_FNIRS_HUB
STAGING_PATIENT_HUB
STAGING_PROBEDATA_HUB
STAGING_SESSION_HUB

1. Go back to Windows Explorer and from our project deliverables folder, navigate to the folder “2. code > Hubs”. This folder contains all of the Python files that will create the hubs of the data vault.
2. Open these files in your IDE of choice and update lines 11 and 17 with your database information for each file. You will need the following information:
   - Username
   - Password
   - Port Number
   - Database name

Line 11 Example:



Line 17 Example:



3. Run the Python files.
4. To verify that each hub table now has been populated, refresh the pgAdmin database and right-click on any of the newly-created tables and select “View/Edit Data > All Rows”.

1.2 Creating Link Tables

1.2.1 SQL Scripts

1. In pgAdmin, right-click on your database name and select “Query Tool”. In the main window, click on the “Query Editor” tab.
2. From smdProject.sql, copy lines 70-129 which are between the comments:

   "—3.1) Creating the Link Tables - please copy… "

   and

   "----Please now run the staging python file for all the Links,…"
3. Paste the selected code into the Query Editor and execute the SQL commands.
4. Refresh the database to see the new tables under your database’s Schemas > public > Tables drop-down menu.

1.2.2 Populating the Link Tables

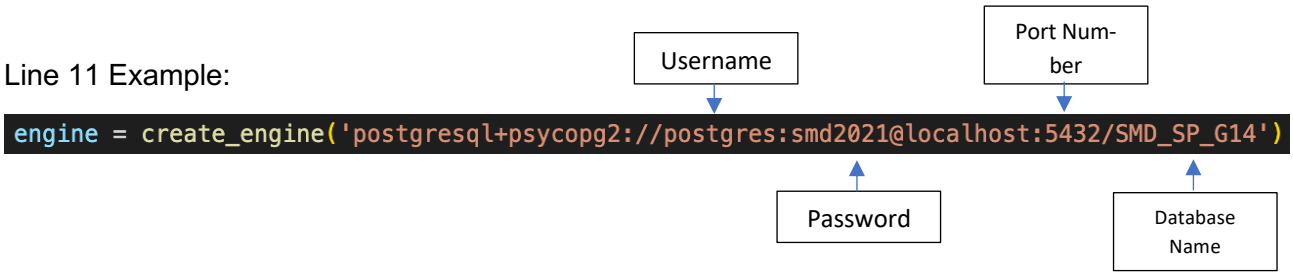
LINK TABLES PYTHON SCRIPT FILE NAMES
STAGING_DATASOURCES-FNIRS_LINK
STAGING_PATIENT-DATASOURCES_LINK
STAGING_PATIENT-EXPERIMENT_LINK
STAGING_PROBEDATA-EXPERIMENT_LINK
STAGING_PROBEDATA-SESSION_LINK
STAGING_SESSIONS-DATASOURCES_LINK

1. Go back to Windows Explorer and from our project deliverables folder, navigate to the folder “2. code > Links”. This folder contains all the Python files that will create the links of the data vault.
2. Open these files in your IDE of choice and update lines 14 and 18 with your database information for each file. You will need the following information:

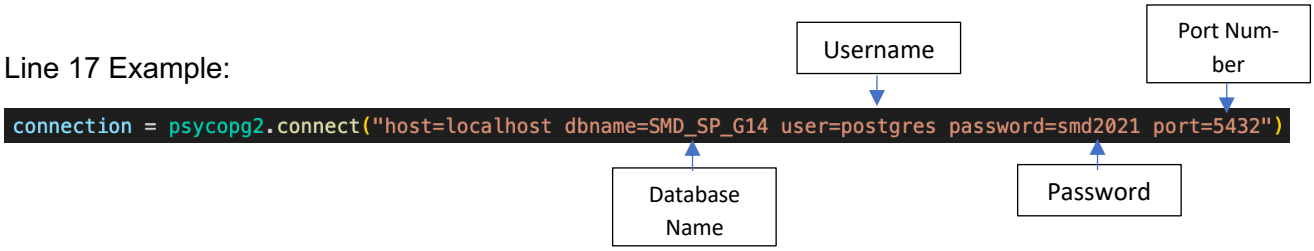
   - Username

- Password
- Port Number
- Database name

Line 11 Example:



Line 17 Example:



3. Run the Python files.
4. To verify that each link table now has been populated, refresh the pgAdmin database and right-click on any of the newly-created tables and select “View/Edit Data > All Rows”.

1.3 Creating And Populating Satellites

1.3.1 SQL Scripts

1. In pgAdmin, right-click on your database name and select “Query Tool”. In the main window, click on the “Query Editor” tab.
2. From smdProject.sql, copy lines 148-156, which are between the between the comments:

   "--Create this Satellite by pasting this code … "

   and

   "–Please now run the staging python file for Experimental Session Sat.."
3. Paste the selected code into the Query Editor and execute the SQL commands.
4. Refresh the database to see the new tables under your database’s Schemas > public > Tables drop-down menu.

1.3.2 Populating the Satellite Tables

The Python files are designed in such way that it interacts with PostgeSQL and creates tables inside it.

SATELLITE CREATION AND POPULATION PYTHON FILE NAMES
STAGING_EXP1_METADATA_SATELLITE

STAGING_EXP2_METADATA_SATELLITE
STAGING_EXP2DAT_SATELLITE
STAGING_EXPERIMENTAL_SESSION_SATELLITE
STAGING_FNIRS_DEOXY_SATELLITE
STAGING_FNIRS_OXY_SATELLITE
STAGING_FNIRS_WL_SATELLITE
STAGING_PATIENT_SATELLITE
STAGING_PROBEDATA_SATELLITE

1. In Windows Explorer, from our project deliverables folder navigate to the “2. Code > Satellites” folder.
2. From this folder, open all 9 Python files in your IDE of choice.
3. Within each file in the Satellites folder (except for Staging_Experimental_Session_Satellite.py, which is dealt with in the next section), update the value of the path objects as prompted in the comments; the code and comments are located between lines 20-30 in each file.
4. For the following files, go to line 25 and change the file directory in the “PATH” object to the location of your copy of the VMData in Dataset 1:
    - Staging_Exp1_Metadata_Satellite.py
    - Staging_fNIRS_Deoxy_Satellite.py
    - Staging_fNIRS_Oxy_Satellite.py
    - Staging_Patient_Satellite.py
    - Staging_ProbeData_Satellite.py
5. For the following Python files go to line 25 and change the 2 objects “PATH” and “PATH2” (or “basepath1” and “basepath2”) to the location of where the fNIRS data in Dataset 2 for “1raSessionDR” and “2daSessionDR” are located:
    - Staging_Exp2_Metadata_Satellite.py
    - Staging_fNIRS_Exp2Dat_Satellite.py
    - Staging_fNIRS_WL_Satellite.py

When pasting the directory please be careful not overwrite the ‘/**/*/.txt’, or ‘/**/*/.dat’ as shown in the examples bellow:

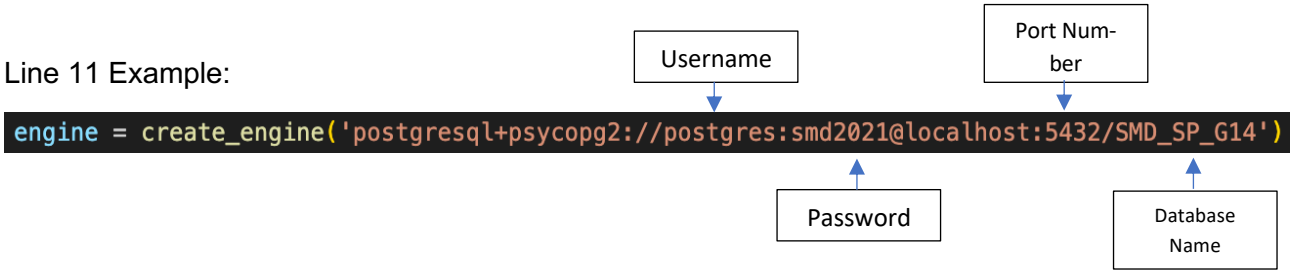
```
PATH = '/Users/doris/Dropbox/-UoB/Courses/SMD/Semester Project/Provided Datasets/Dataset 2/fNIRS-Data/1raSessionDR/**/*.txt'
PATH2 = '/Users/doris/Dropbox/-UoB/Courses/SMD/Semester Project/Provided Datasets/Dataset 2/fNIRS-Data/2daSessionDR/**/*.txt'
```

```
PATH = '/Users/doris/Dropbox/-UoB/Courses/SMD/Semester Project/Provided Datasets/Dataset 2/fNIRS-Data/1raSessionDR/**/*.dat'
PATH2 = '/Users/doris/Dropbox/-UoB/Courses/SMD/Semester Project/Provided Datasets/Dataset 2/fNIRS-Data/2daSessionDR/**/*.dat'
```

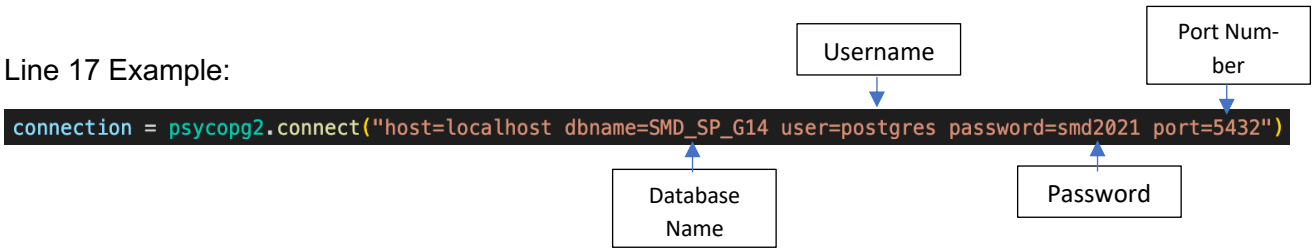
1.4 Creating And Populating Experimental Session Satellite

1. Go back to Windows Explorer and from our project deliverables folder, navigate to the “2. code > Satellites" folder, open Staging_Experimental_Session_Satellite.py in your IDE.
2. Update lines 16 and 18 with your database information. You will need the following information:
    1. Username
    2. Password
    3. Port Number
    4. Database name

Line 11 Example:



Line 17 Example:



3. Run the file.
4. To verify that each the relevant table has been populated, refresh the pgAdmin database and right-click on the “experimental_session_satellite” table and select “View/Edit Data > All Rows”.

1.5 Adding Database Constraints

1.5.1 SQL Scripts

1. In Windows Explorer, from our project deliverables folder navigate to the “2. code” folder containing the SQL scripts.
2. In pgAdmin, right-click on your database name and select “Query Tool”. In the main window, click on the “Query Editor” tab.
3. From smdProject.sql, copy lines 163-200 and paste the selected code into the Query Editor and execute the SQL commands.

2.0 Importing Django Web Apps

2.1 Preparing Tables for Visualisations & User Authentication

1. In pgAdmin, right-click on your database name and select “Query Tool”. In the main window, click on the “Query Editor” tab.

2. From smdProject.sql, copy lines 220-239 and paste the selected code into the Query Editor and execute the SQL commands.
3. Refresh database so that you can see the new tables “image” and “login_user” created in the database.

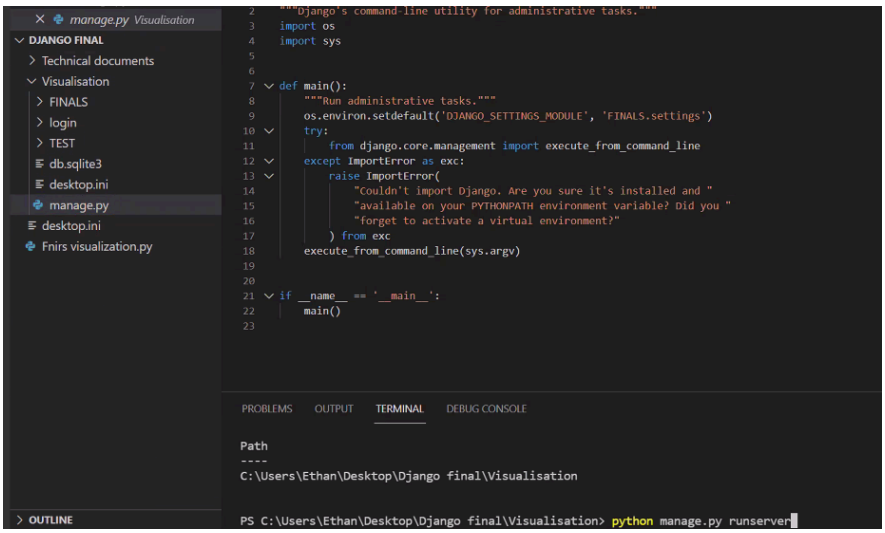
2.2 Initialising Django & Web Server

1. In Windows Explorer, from our project deliverables folder navigate to the “2. Code > Django”.
2. Open “Fnirs_visualization.py” in your IDE.
3. Go to line 25 Fnirs_visualization.py and change the 2 objects “basepath1” and “basepath2” to the location of where the fNIRS data in Dataset 2 for “1raSessionDR” and “2daSessionDR” are located. When pasting the directory please be careful not overwrite the ‘/**/*/.txt’, or ‘/**/*/.dat’.
4. Run Fnirs_visualization.py.
5. Open the Django folder with your IDE of choice.
6. Navigate to “2. Code > Django > Visualisation > FINALS” folder and open settings.py in your IDE.
7. In settings.py, and updates lines 81 to 85 to correspond with your database settings. (These should be the same changes that were mentioned in section 1.1.2)

```
# Database
# https://docs.djangoproject.com/en/3.2/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE':'django.db.backends.postgresql_psycopg2',
        'NAME':'SMD_SP_G14',
        'USER':'postgres',
        'PASSWORD':'w845393686',
        'HOST':'localhost',_# 远程服务器IP
        'PORT':'5432',
    }
}
```

8. In your IDE terminal, change your working directory to ./2. Code/Django/Visualisation/FINALS/, and then run the following commands in order:
    - python manage.py migrate
    - python manage.py runserver



9. In a web browser open http://127.0.0.1:8000/login
10. Use the following username & password to log in: admin / 123456789.
11. After logging in, you should be redirected to index.html. From this page click on “Go to the data visualisation menu” to produce the visualisations described in section 2.3 below.

2.3 Sample Visualisations

Plot 1

1. Choose “Experiment 2” from the first drop-down list.
2. Enter “11” for the patient ID.
3. Choose Session 1 from the next drop-down list.
4. Choose “filtering” from the drop-down list and leave the values in the adjacent text boxes unchanged.
5. Click “Submit” to display the requested image.

Plot 2

1. Choose “Experiment 2” from the first drop-down list.
2. Enter “22” for the patient ID.
3. Click on Session 1 from the next drop-down list.
4. Choose “response image” from the drop-down list and leave the values in the adjacent text boxes unchanged.
5. Click “Submit” to display the requested image.

Plot 3

1. Choose “Experiment 1” from the first drop-down list.
2. Enter “1” for the patient ID.
3. Select “moto” from the next drop-down list.
4. Next, select “MES” and click “Submit”.
5. On the next page, select the first three channels.
6. Scroll to the bottom of the page and enter a range between 0 and 100.
7. Click “Submit” to plot the requested data.

3.0 Restoring Our pgAdmin Database from a Backup

If needed, follow these steps to restore a backup of our pgAdmin database (named SMD_SP_G14.bak in the “3. Data” folder):

1. Create an empty database in pgAdmin
2. Right-click on the empty database and select "Restore..."
3. In the Restore pop-up window set the following options:
    - Leave the format as "Custom or tar"
    - For "Filename,” select SMD_SP_G14.bak
    - For "Role name" select postgres from the drop-down menu
4. Click “Restore” to begin restoring the backup to the new database.