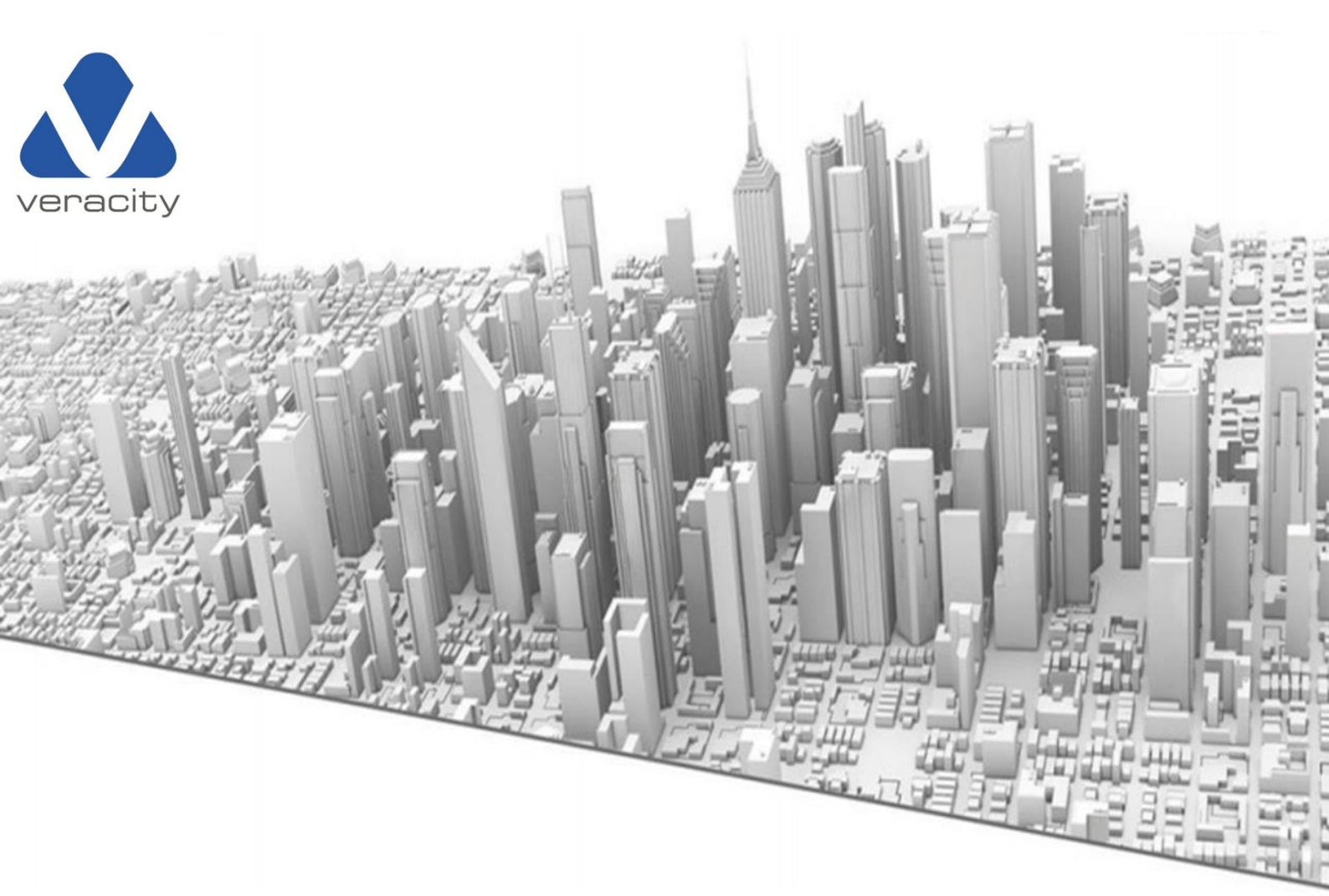


SOP Management – Implementation and Guide

Ritwick Handa

Junior Support & Product Engineer

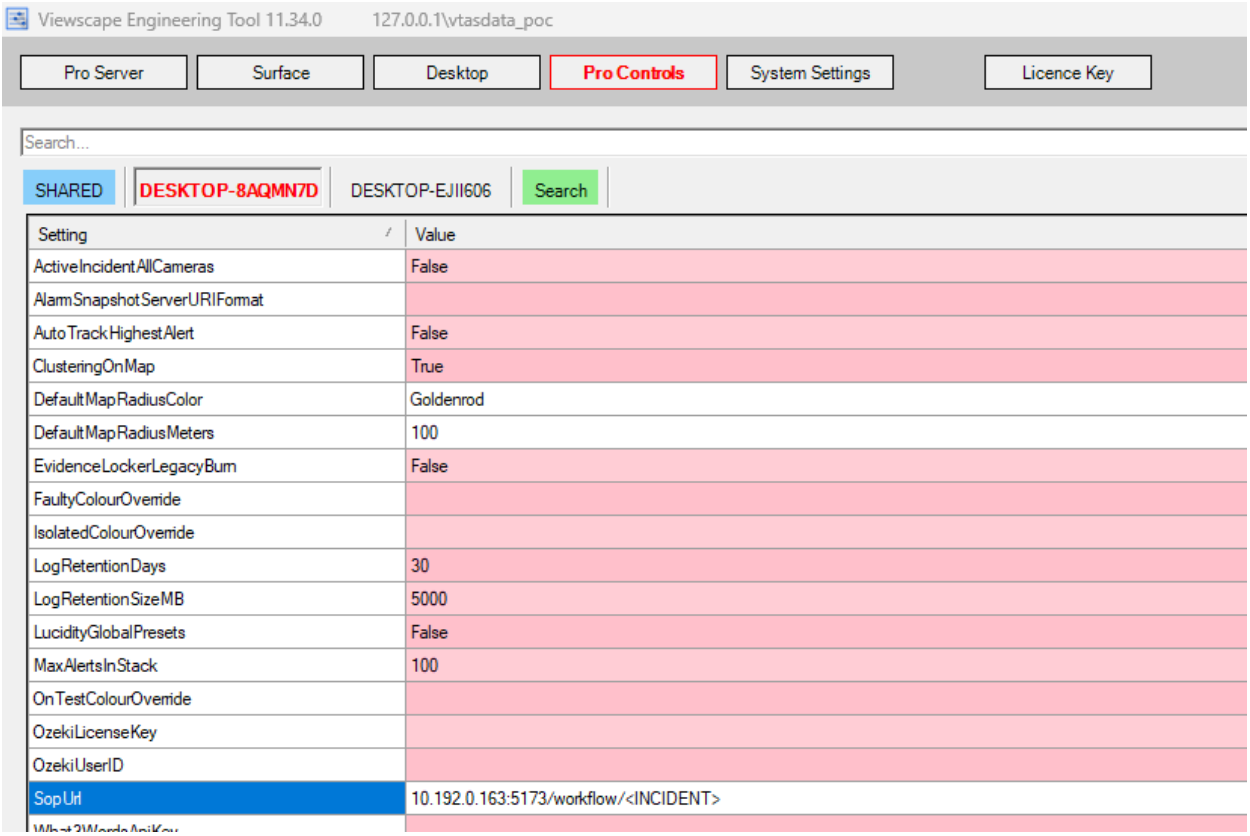
Veracity UK Ltd

Pre-requisites

- Python
- NodeJS
- SQL server 16 or above.
- ODBC Driver 17
- SOP manage code folder
- SOP manage database
- NSSM
- V11.33.1 build or above with Procedure
- Setting Editor

Points to consider:

- In Viewscape Engineering tool, In **Pro Controls Tab,** you will see **SOP URL** in which you can add the **url** which will be opening in the procedure tab in viewscape.



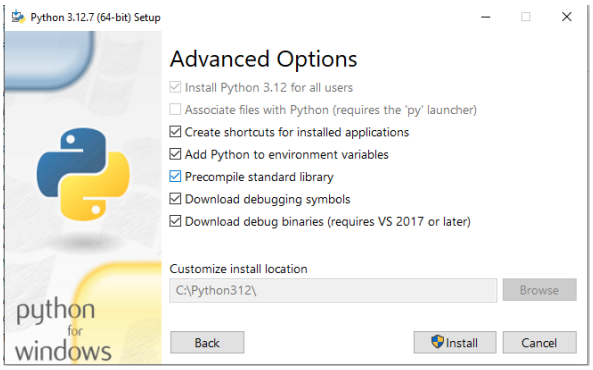
- The Default URL Is IP:port/workflow/<INCIDENT>
- If the SOP url option is not visible just delete the client setting for that user and also delete shared setting. Then restart the pro server and open the viewscape client again.
- After that reopen the settings editor it will be visible.

Introduction

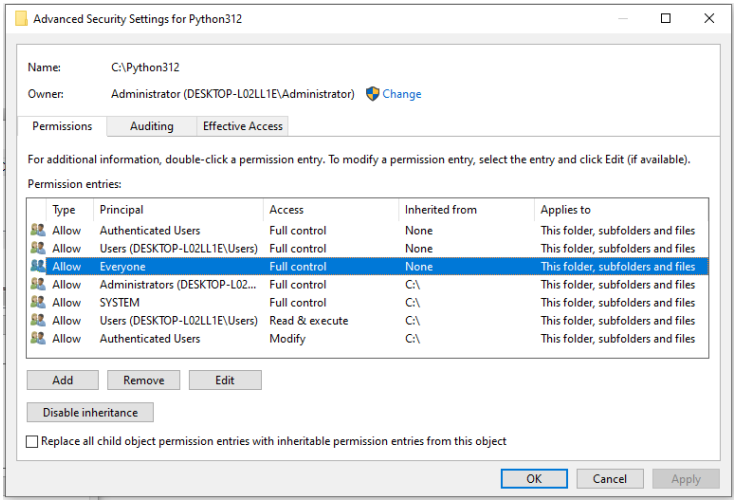
The Incident Management System allows administrators to create workflows, which consist of a series of questions and answers designed to guide users through specific incident-handling processes. The system offers features for admins to design workflows, share them with users, and review completed responses.

Python setup

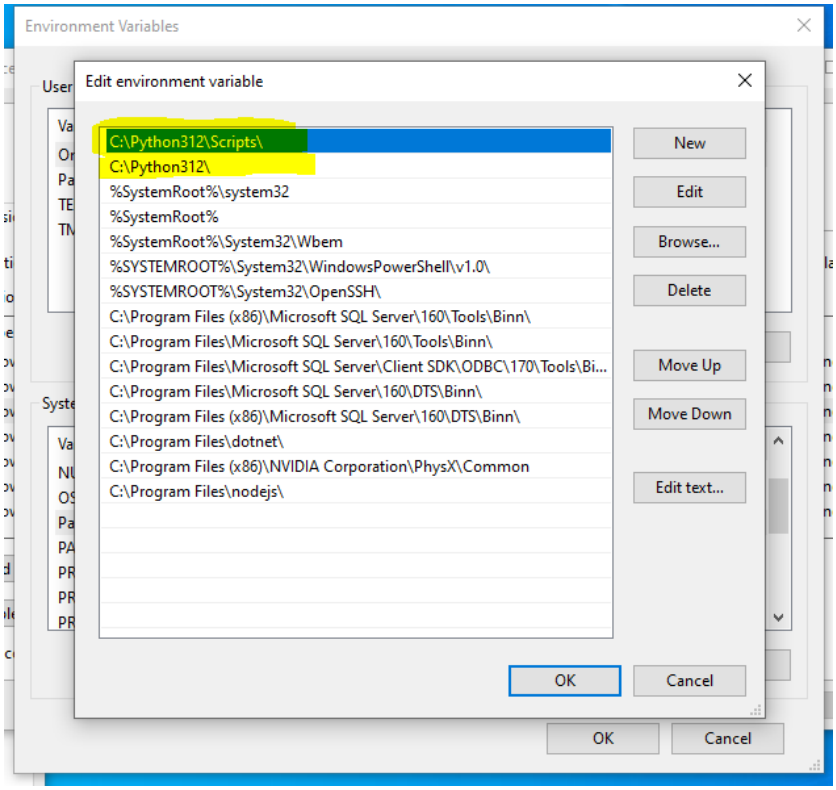
- Download and install python on your machine. (Ensure you check the option to "Add Python to PATH" during installation.) https://www.python.org/downloads/



- Give everyone full control (rights) to the python folder

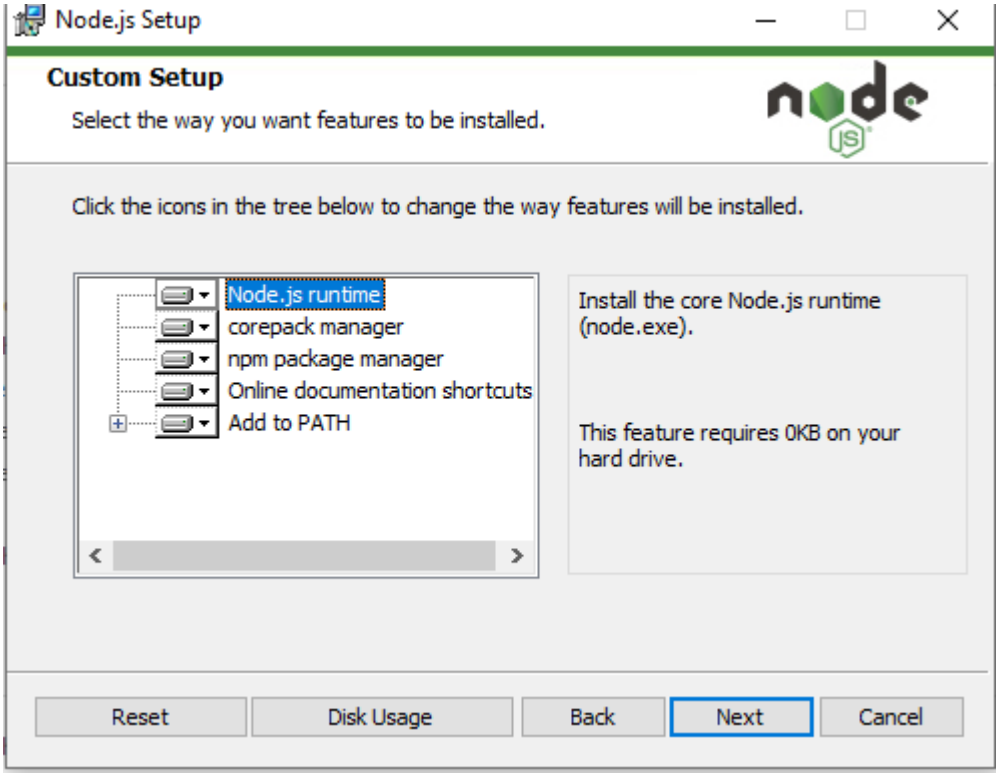


- Add python path to Environmental variable and python script path



NodeJS

- Download and install NodeJS (https://nodejs.org/en)
- It is a basic installation. Just click next and move forward.

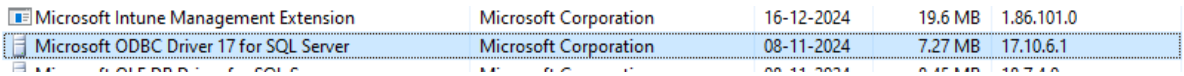


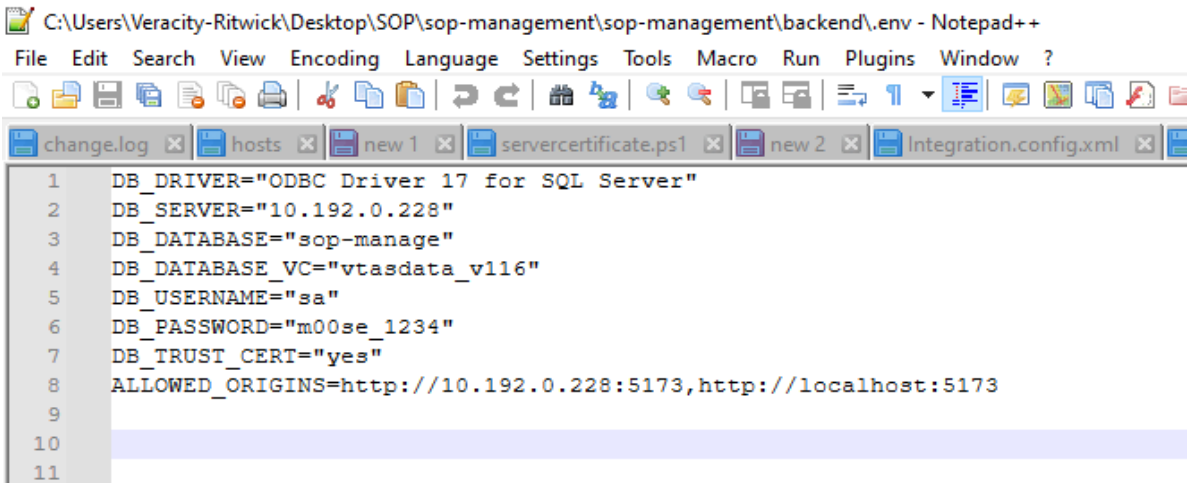
Restoring the SOP manage database

- The SOP manage database will be provided.
- Just open SQL studio and restore the SOP manage database

Changes in frontend and backend code

- You will receive a folder named "**SOP Management**" containing all the frontend and backend code. You can place the folder in any location of your choice.
- Change the code in. env (Environment variables).
  Path: sop-management\backend\.env
- DB_DRIVER="ODBC Driver 17 for SQL Server" (Check in control panel)
- DB_SERVER="10.192.0.228"
- DB_DATABASE="sop-manage"
- DB_DATABASE_VC="vtasdata_v116" (or the DB which has IncidentLog_TBL)"(where you want to attach SOP)
- DB_USERNAME="sa"
- DB_PASSWORD="m00se_1234"
- DB_TRUST_CERT="yes"
- ALLOWED_ORIGINS=http://10.192.0.228:5173,http://localhost:5173
- **Change the above code according to your system.**





- **Change the database name and IP of SQL server in backend/config/database.py: C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management\backend\config**
- **Backend>Config>database.py**
- Change line 18 and 28, add the IP where your SQL in installed
- Change line 19 and 29 according to the database names

```
DB_CONFIG = {
    'driver': os.getenv('DB_DRIVER', 'ODBC Driver 17 for SQL Server'),
    'server': os.getenv('DB_SERVER', '10.192.0.228'),
    'database': os.getenv('DB_DATABASE', 'sop-manage'),
    'username': os.getenv('DB_USERNAME', 'sa'),
    'password': os.getenv('DB_PASSWORD', ''),
    'trust_cert': os.getenv('DB_TRUST_CERT', 'yes'),
}

# Database configuration for TEST database
DB_VC_CONFIG = {
    'driver': os.getenv('DB_DRIVER', 'ODBC Driver 17 for SQL Server'),
    'server': os.getenv('DB_SERVER', '10.192.0.228'),
    'database': os.getenv('DB_DATABASE_VC', 'vtasdata_v116'),  # Ensure correct DB name for vtasdata_v116
    'username': os.getenv('DB_USERNAME', 'sa'),
    'password': os.getenv('DB_PASSWORD', ''),
    'trust_cert': os.getenv('DB_TRUST_CERT', 'yes'),
}
```

- **Change the database name in backend/ wf_builder_service.py:**
- **Backend>service> wf_builder_service.py**
- **Line number 541**

C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management\backend\services

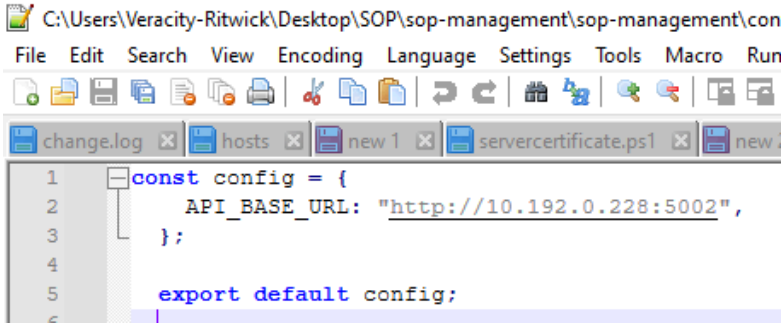
```
473         static_heading = f"======SOP - {workflow_name} ======"
474
475         # Construct the SQL query to update inlIncidentDetails_MEM
476         query = text("""
477         UPDATE [vtasdata_v116].[dbo].[IncidentLog_TBL]
478         SET inlIncidentDetails_MEM =
479         ISNULL(CAST(inlIncidentDetails_MEM AS NVARCHAR(MAX)), '') +
480         CASE
```

- Change “vtasdata_v116” to the database name —> Line **541**
- **Backend> Main.py**
- **Check the port in the last line and chose the port accordinly**

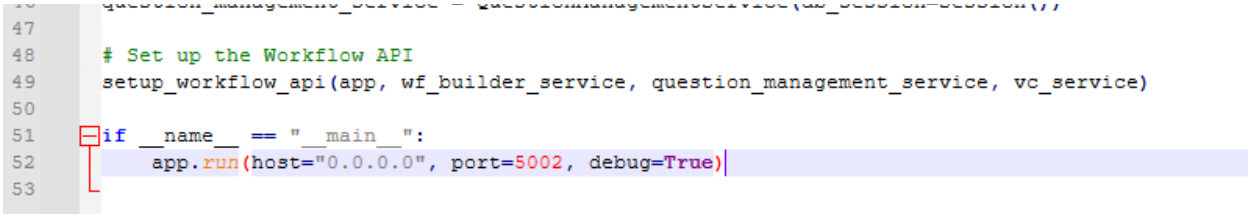
Frontend Code

- **code change (Port and IP)**
- **In SOP Management folder, CONFIG.JS file**

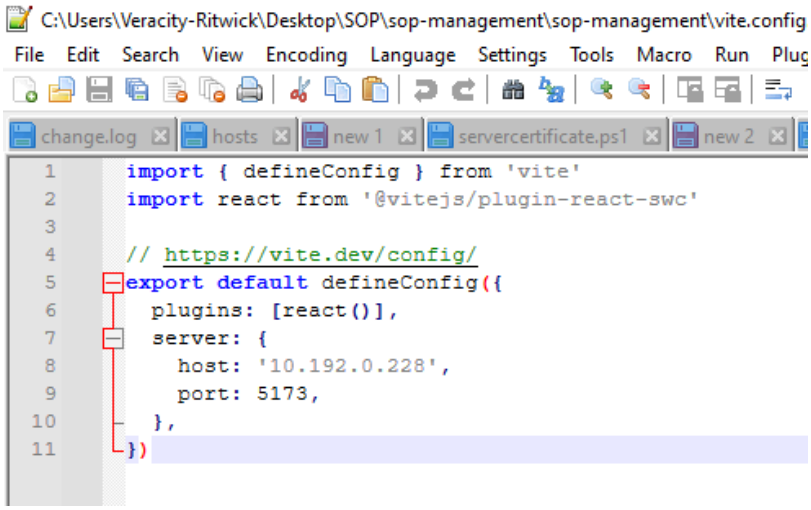
- **config.js** should have the url that will run the api, add, "http://10.192.0.228:5002" (Add static IP not local IP) C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management



- To check if the port is open or not, open cmd, type netstat -an | find "5173" or netstat -an | find "5000" for both frontend and backend respectively
- To change the port, check the main.py, on line 52, change the port for the backend C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management\backend
  app.run(host="0.0.0.0", port=5002, debug=True) (just change the port here)



- **then add this to the vite.config.js (Just Change the port and IP)**
- **Open SOP management > Vite.Config.js**
- **C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management**

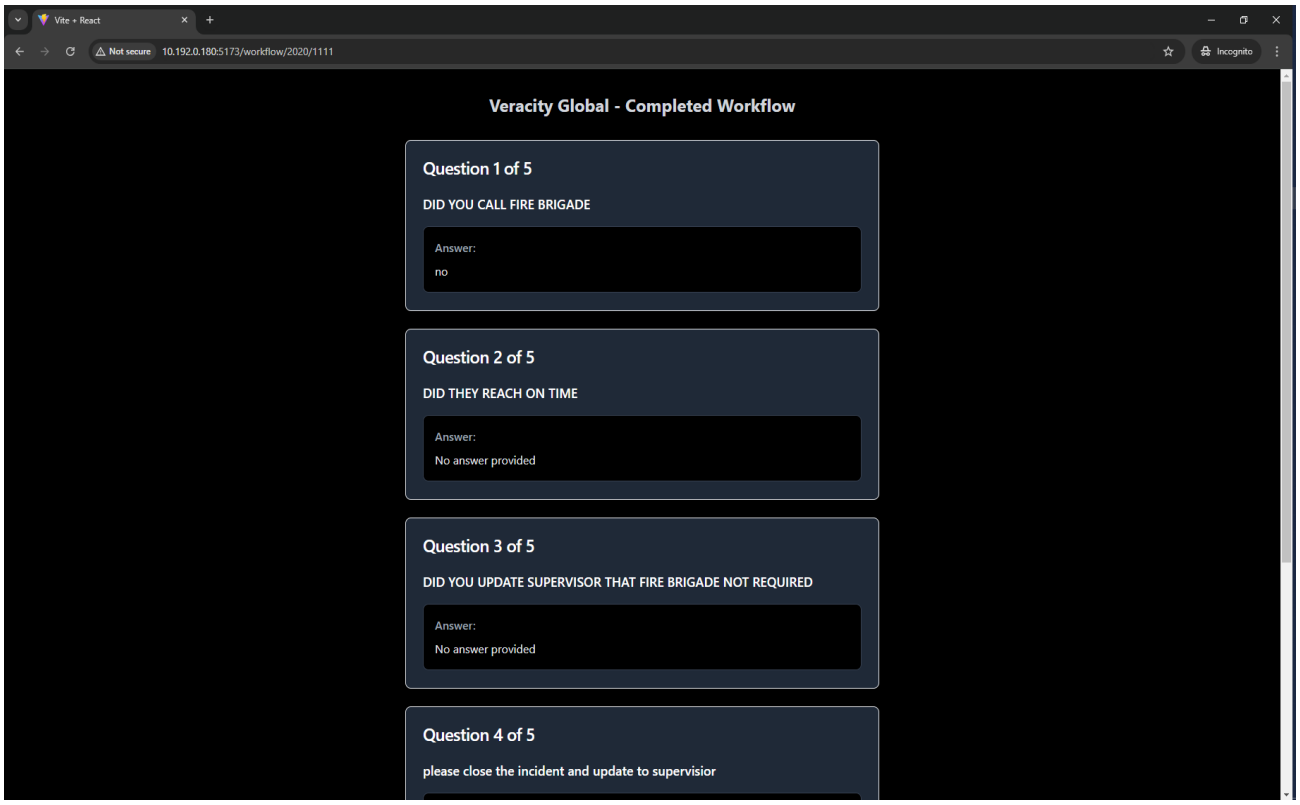


Code Execution process

- This is to run Backend code (First we run backend on a terminal then we run frontend in different terminal)
- Open terminal
- 2. Type cd backend (Change the directory to backend) C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management\backend
- 3. pip install -r requirements.txt Path C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management\backend
- 4. python main.py
- This will run the backend code

- Frontend
- 1. Open second terminal
- 2. Type cd C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management (to identify the frontend folder look for package.json file)
- 3. npm install
- 4. npm run dev
- This will run the backend code.

- Now let's check if web page is visible or not
- Open web browser and type your IP and frontend port where the code is running (eg : 10.192.0.228:5173)
- The workflow page would be visible on the webpage.



- Check if the webpage is working, then we can start creating a service for the code. **Just type the IP and port to check if the SOP is working**

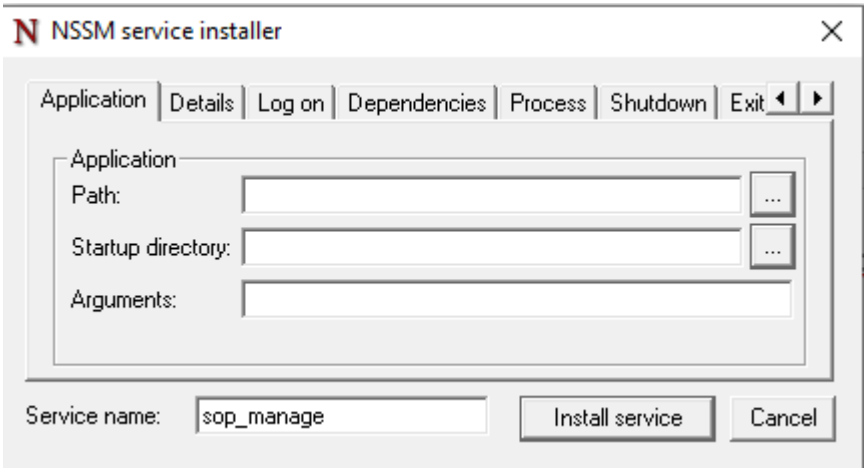
NSSM Service

- Download and Install NSSM (https://nssm.cc/download)
- You will get a BAT file with this document.
- Changes in the SOP_manage.bat file
- Please find below content of the BAT file

```
@echo off

:: Ensure the script is running with admin privileges

NET SESSION >nul 2>&1

IF %ERRORLEVEL% NEQ 0 (

   echo Requesting Administrator privileges...

   powershell -Command "Start-Process '%~f0' -Verb RunAs"

   exit

)


:: Run Python script

cd C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management\backend
( add the path of backend folder where main.py is present )

start "" "C:\Python312\python.exe" "main.py"      (Add the path where your python .exe is installed)


:: Run Node.js script

cd C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management      (add the path of SOP manage folder where package.json is present)

call npm.cmd run dev


:: Open the URL in the default web browser

start http://127.0.0.1:5000        (Keep this similar)


pause
```

- Change the path in Python script
- Change the path in Node.js script

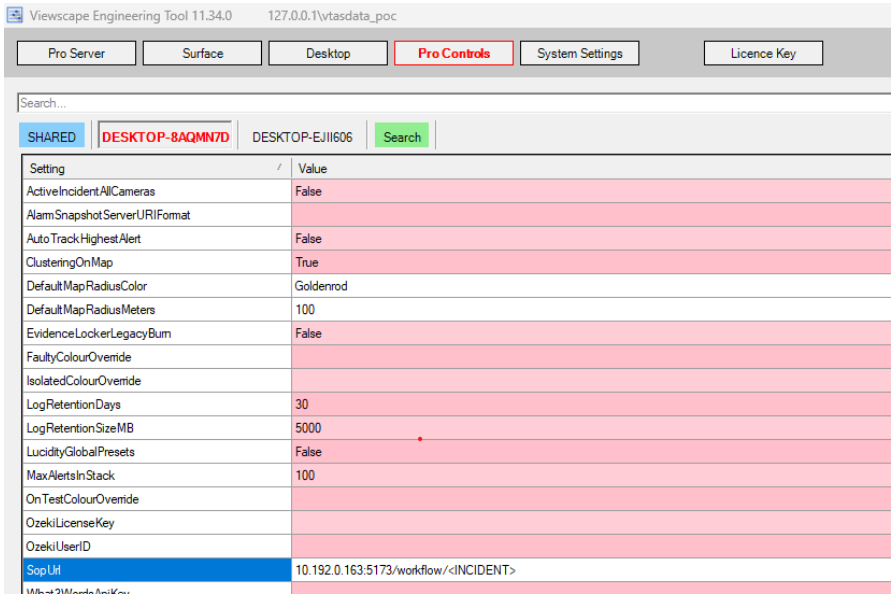
- Extract the NSSM and open the win 64 folder. Open this folder in cmd or type cmd in this path. (C:\Users\Veracity-Ritwick\Downloads\nssm-2.24\nssm-2.24\win64)
- in CMD type nssm install (service name)
- a popup will appear, attach the path where the bat file is present and click on the install service.



- now service is created.
- type NSSM start (service name) in the same cmd.
- or you can start the service from the task manager > service
- Once the service starts running, check if the web page is working or not

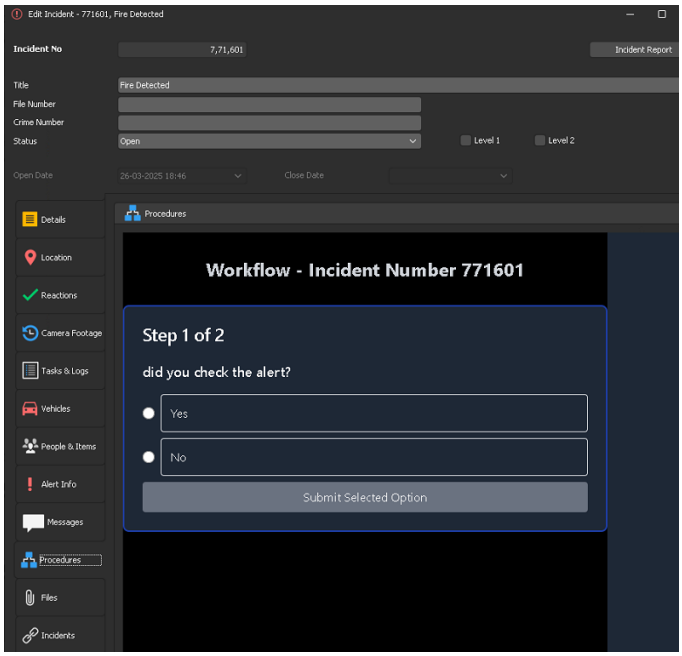
CONFIGURING WITH VIEWSCAPE

- Begin by adding the SOP URL in the Viewscape Engineering tool.
- Go to the **Pro Controls** tab, and at the bottom, you will find the **SOP URL** section.
- The default structure for the SOP URL is:
  IP:port/workflow/<INCIDENT>



To link the SOP with different types of incidents, follow these steps to create the corresponding workflow:

- In Viewscape, assign an incident category, for example, "Fire Detected" for a fire incident.
- In the Workflow Builder, create a workflow that matches the incident category, such as "fire_detected." (This will enable the auto-link functionality to work.)
- Note that you don't need to add underscores for spaces in the incident category name in Viewscape. However, when creating the workflow in the Workflow Builder, make sure to replace spaces with underscores (e.g., "Fire Detected" becomes "fire_detected"). (**It should be visible in procedure in incident like the image below**)



System Features

1. Admin Dashboard

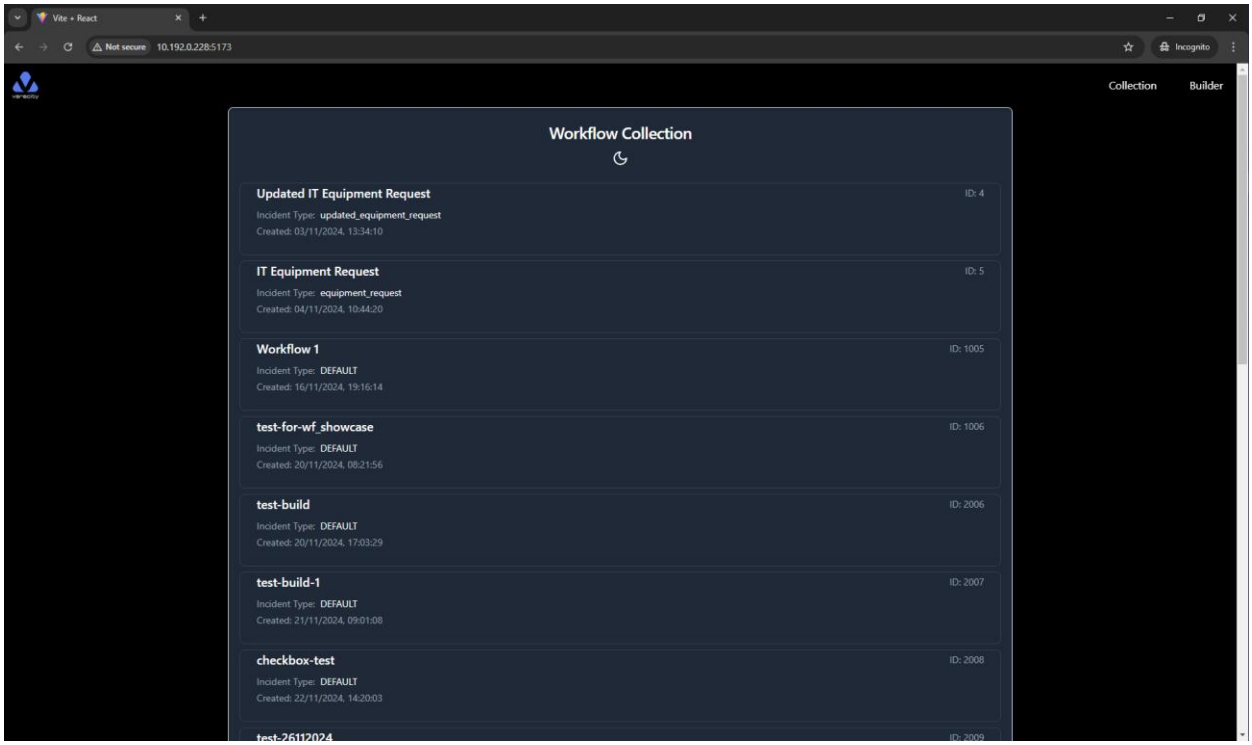
- Create Workflows: Administrators can design workflows that include sequential questions and branching paths based on user inputs.
- Manage Workflows: Admins can edit, delete, or update existing workflows as required.
- Assign and Share Workflows: Admins can share workflow links with users, allowing them to participate in incident-specific workflows.
- Analyse Responses: Admins can review the answers submitted by users for specific incidents.

2. User Interface (Form Format)

- Dynamic Question Flow: Users answer questions in a dynamic sequence, progressing through the form based on their responses.
- Workflow Summary: Upon completing the form, users are presented with a summary of the questions they have answered.

3. Pages Overview

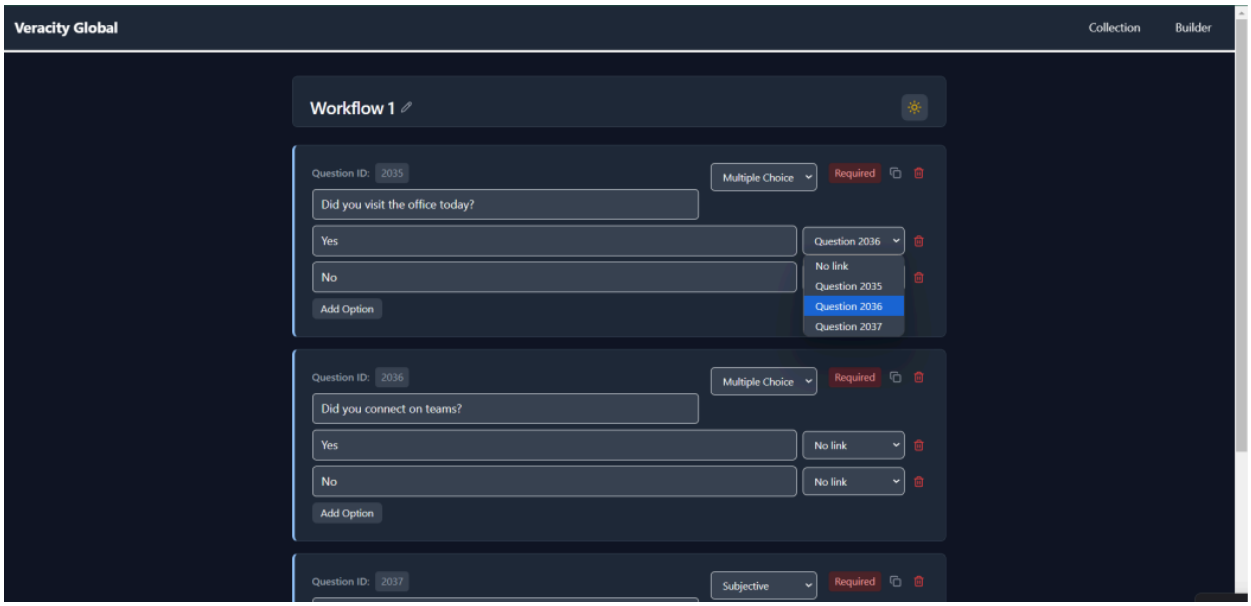
- Workflow Collection for Administrators (http://127.0.0.1:5173/): Displays all available workflows for the admin.
- Workflow Builder (http://127.0.0.1:5173/builder): Allows admins to create or edit workflows.



How It Works

1. Admin Creating Workflow

- Step 1: The admin logs into the system and navigates to the Workflow Builder page (http://127.0.0.1:5173/builder).
- Step 2: The admin defines the starting question (Q1) and connects subsequent questions (Q2, Q3, etc.) based on user responses (e.g., Yes/No or other answers).
- Step 3: Each response option leads to another question, and the admin can continue adding branches until the workflow is complete. Note: The Builder component only allows Multiple Choice questions to be connected. Checkbox, Subjective, and Instruction questions cannot be connected and will be displayed as static information.
- Step 4: Once the workflow is fully built, it is saved and can be assigned to users for specific incidents.



1. User Taking a Form

- Step 1: The user opens the shareable link and enters the incident number either through the modal or directly via the link (http://127.0.0.1:5173/workflow/incident/).
- Step 2: The user answers the questions presented dynamically, with the next question depending on their previous responses.
- Step 3: After completing the workflow, the user is shown a summary of all the questions they answered.