

## 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 screenshot shows the Viewscape Engineering Tool interface. At the top, there's a navigation bar with tabs: Pro Server, Surface, Desktop, Pro Controls (which is highlighted in red), System Settings, and Licence Key. Below the navigation bar is a search bar labeled "Search...". Underneath the search bar, there are three tabs: SHARED, DESKTOP-8AQM7D (which is highlighted in blue), and DESKTOP-EJII606. To the right of these tabs is a "Search" button. The main area is a table with two columns: "Setting" and "Value". The table contains numerous settings, and the last row, which is highlighted with a blue background, shows the "SopUrl" setting with the value "10.192.0.163:5173/workflow/<INCIDENT>".

Setting	Value
ActiveIncidentAllCameras	False
AlarmSnapshotServerURIFormat	
Auto Track Highest Alert	False
ClusteringOnMap	True
Default Map Radius Color	Goldenrod
Default Map Radius Meters	100
Evidence Locker Legacy Burn	False
Faulty Colour Override	
Isolated Colour Override	
Log Retention Days	30
Log Retention Size MB	5000
Lucidity Global Presets	False
Max Alerts In Stack	100
On Test Colour Override	
Ozeki License Key	
Ozeki User ID	
SopUrl	10.192.0.163:5173/workflow/<INCIDENT>

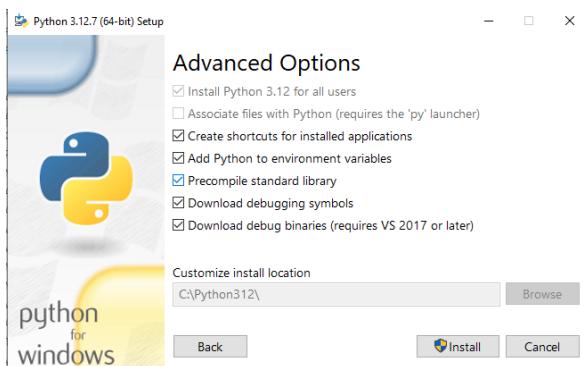
- The Default URL Is IP:port/workflow/<INCIDENT>
- If the SOP url option is not visible just delete the client setting for that user and also delete shared setting. Then restart the pro server and open the viewscape client again.
- After that reopen the settings editor it will be visible.

## **Introduction**

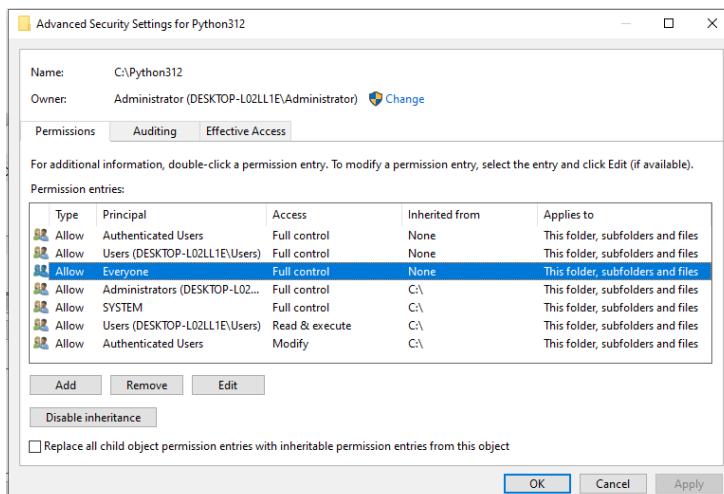
The Incident Management System allows administrators to create workflows, which consist of a series of questions and answers designed to guide users through specific incident-handling processes. The system offers features for admins to design workflows, share them with users, and review completed responses.

# Python setup

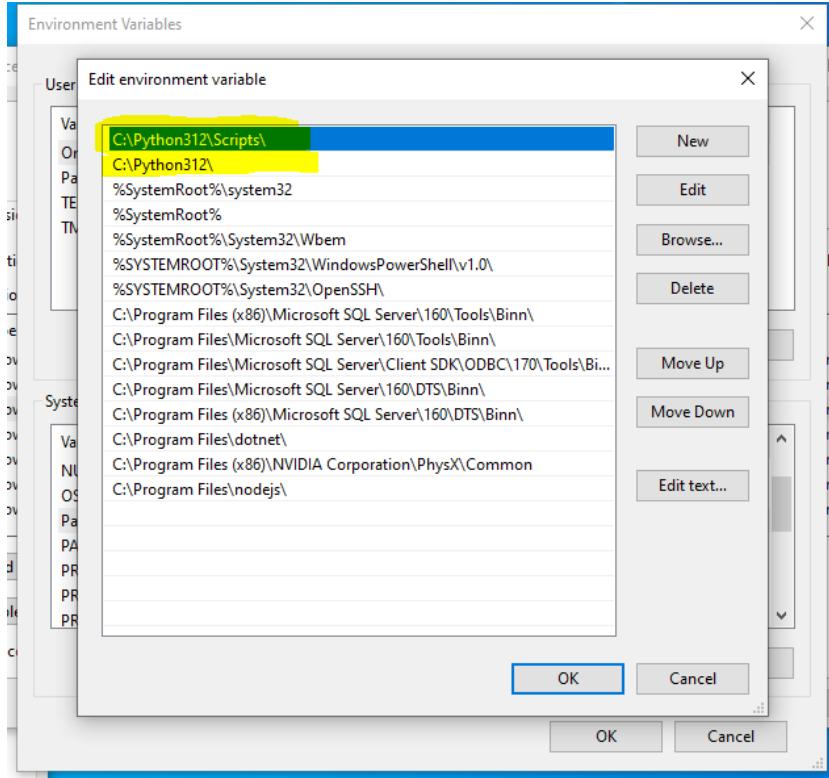
- Download and install python on your machine. (Ensure you check the option to "Add Python to PATH" during installation.) <https://www.python.org/downloads/>



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

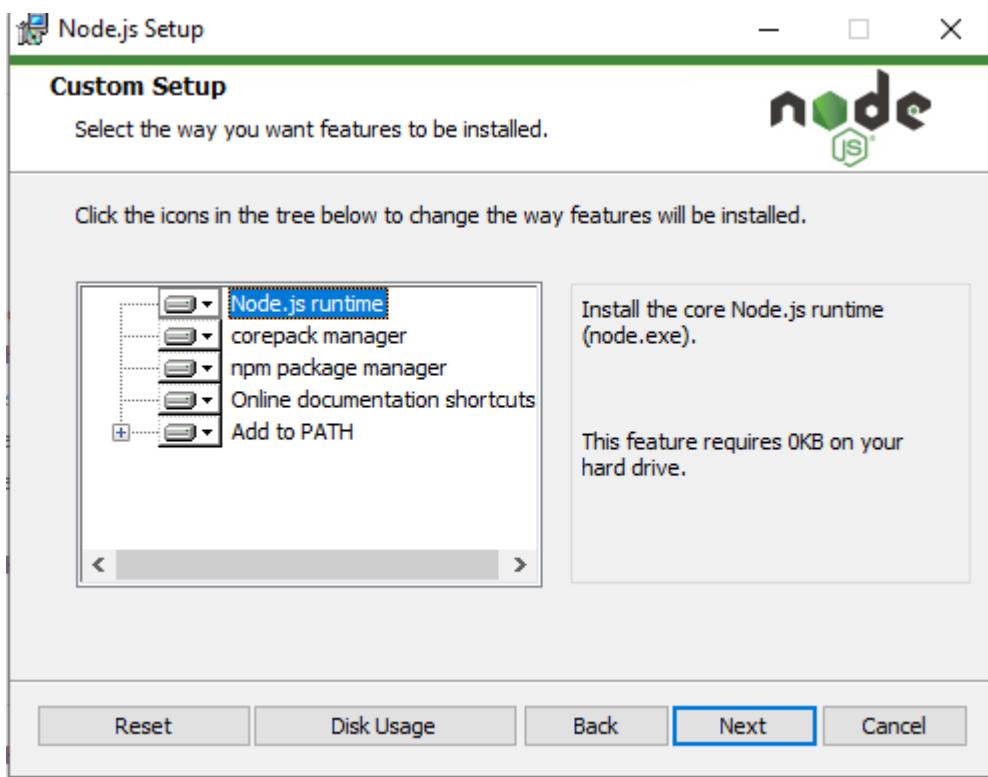


- Add python path to Environmental variable and python script path



## 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.**

 Microsoft Intune Management Extension	Microsoft Corporation	16-12-2024	19.6 MB	1.86.101.0
 Microsoft ODBC Driver 17 for SQL Server	Microsoft Corporation	08-11-2024	7.27 MB	17.10.6.1
 Microsoft SQL Server	Microsoft Corporation	08-11-2024	0.45 MB	10.7.10

```

C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\backend\.env - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
change.log hosts new 1 servercertificate.ps1 new 2 Integration.config.xml
1 DB_DRIVER="ODBC Driver 17 for SQL Server"
2 DB_SERVER="10.192.0.228"
3 DB_DATABASE="sop-manage"
4 DB_DATABASE_VC="vtasdata_v116"
5 DB_USERNAME="sa"
6 DB_PASSWORD="m00se_1234"
7 DB_TRUST_CERT="yes"
8 ALLOWED_ORIGINS=http://10.192.0.228:5173,http://localhost:5173
9
10
11

```

- **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 is 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

```

472
473     # Dynamically format the heading with the workflow_name
474     static_heading = f"=====SOP - {workflow_name} ====="
475
476     # Construct the SQL query to update inlIncidentDetails_MEM
477     query = text("""
478         UPDATE [vtasdata_v116].[dbo].[IncidentLog_TBL]
479             SET inlIncidentDetails_MEM =
480                 ISNULL(CAST(inlIncidentDetails_MEM AS NVARCHAR(MAX)), '') +
481                 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

```
C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management\config.js
File Edit Search View Encoding Language Settings Tools Macro Run
change.log hosts new 1 servercertificate.ps1 new .
1 const config = {
2     API_BASE_URL: "http://10.192.0.228:5002",
3 };
4
5     export default config;
6 
```

- 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)

```
47
48     # Set up the Workflow API
49     setup_workflow_api(app, wf_builder_service, question_management_service, vc_service)
50
51     if __name__ == "__main__":
52         app.run(host="0.0.0.0", port=5002, debug=True)
53 
```

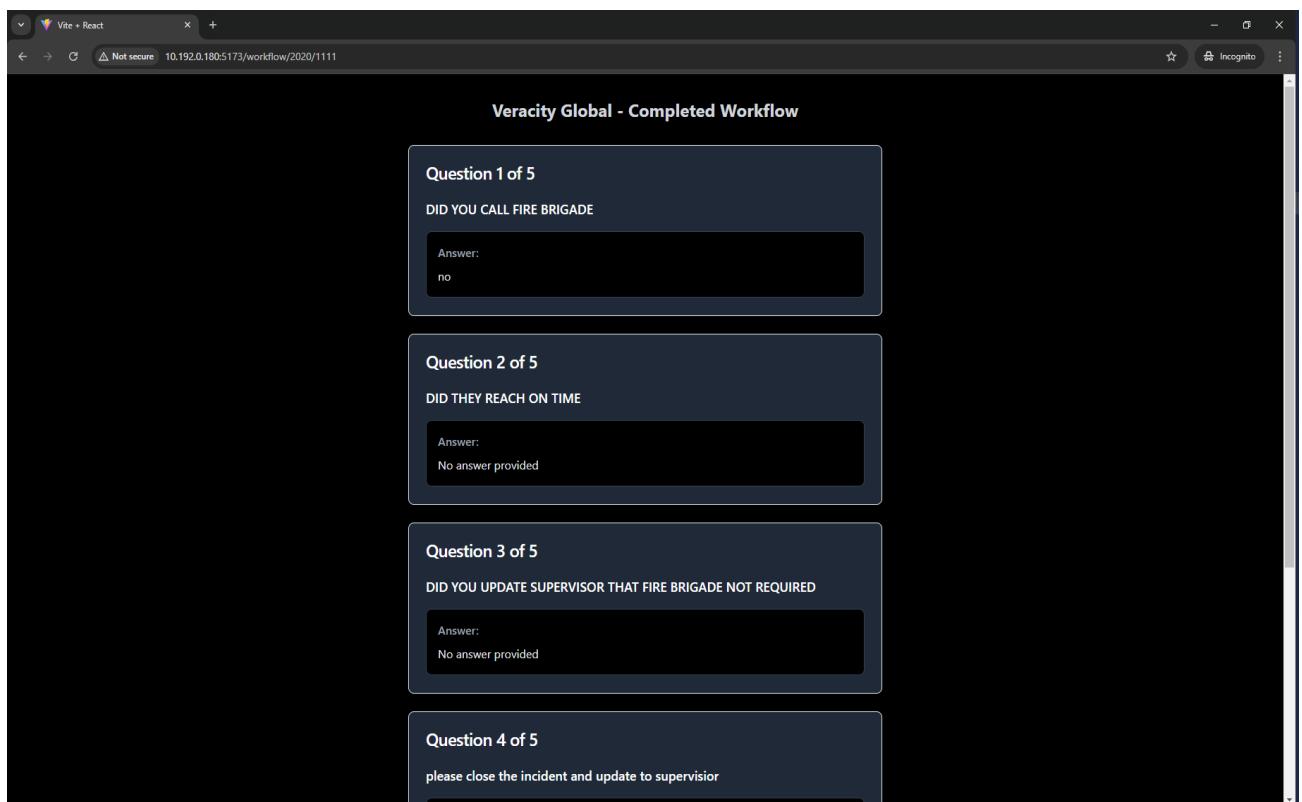
- then add this to the vite.config.js (Just Change the port and IP)
- Open SOP management > Vite.Config.js
- C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management

```
C:\Users\Veracity-Ritwick\Desktop\SOP\sop-management\sop-management\vite.config
File Edit Search View Encoding Language Settings Tools Macro Run Plug
change.log hosts new 1 servercertificate.ps1 new 2
1 import { defineConfig } from 'vite'
2 import react from '@vitejs/plugin-react-swc'
3
4 // https://vite.dev/config/
5 export default defineConfig({
6     plugins: [react()],
7     server: {
8         host: '10.192.0.228',
9         port: 5173,
10    },
11 }) 
```

## 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

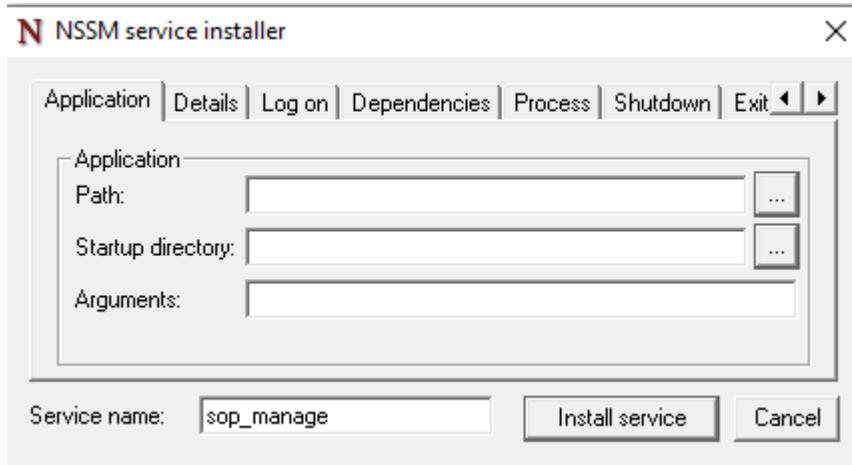
:: Open the URL in the default web browser

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

pause

    • Change the path in Python script
    • Change the path in Node.js script
```

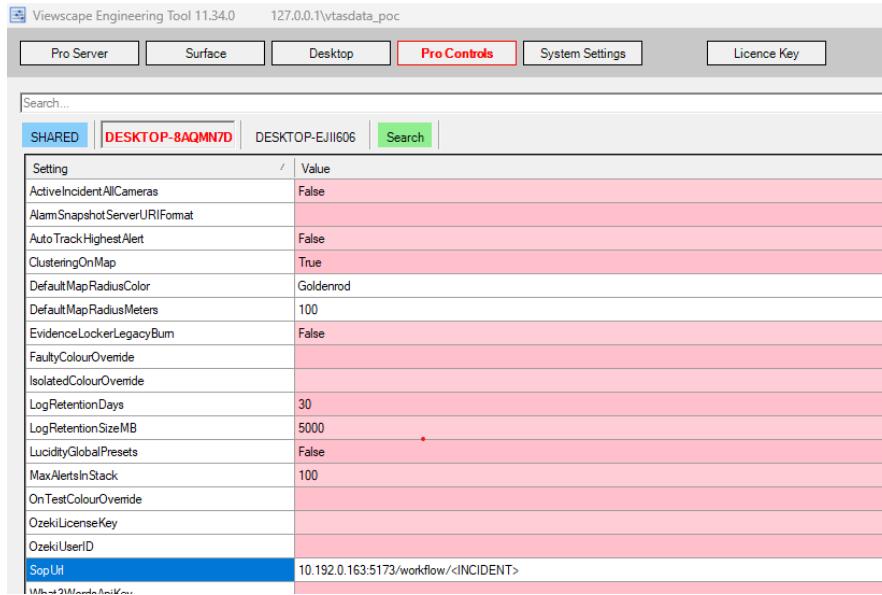
- Extract the NSSM and open the win 64 folder. Open this folder in cmd or type cmd in this path. (C:\Users\Veracity-Ritwick\Downloads\nssm-2.24\nssm-2.24\win64)
- in CMD type nssm install (service name)
- a popup will appear, attach the path where the bat file is present and click on the install service.



- 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

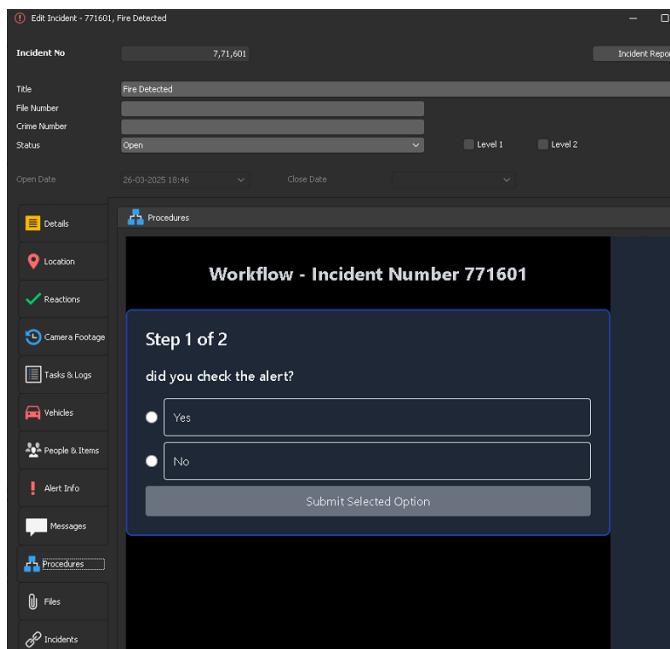
- Begin by adding the SOP URL in the Viewscape Engineering tool.
- Go to the **Pro Controls** tab, and at the bottom, you will find the **SOP URL** section.
- The default structure for the SOP URL is:  
**IP:port/workflow/<INCIDENT>**



The screenshot shows the Viewscape Engineering Tool interface. At the top, there's a header bar with the title 'Viewscape Engineering Tool 11.34.0' and the IP address '127.0.0.1\vtasdata\_poc'. Below the header are several tabs: 'Pro Server', 'Surface', 'Desktop', 'Pro Controls' (which is highlighted in red), 'System Settings', and 'Licence Key'. Under the 'Pro Controls' tab, there's a search bar with placeholder text 'Search...'. Below the search bar is a table with two columns: 'Setting' and 'Value'. The table lists various configuration settings. One row is highlighted in blue, showing 'SopUrl' with the value '10.192.0.163:5173/workflow/<INCIDENT>'. The rest of the table rows have pink backgrounds.

**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)**



The screenshot shows the Viewscape Incident Management interface. On the left, there's a sidebar with icons for 'Details', 'Location', 'Reactions', 'Camera Footage', 'Tasks & Logs', 'Vehicles', 'People & Items', 'Alert Info', 'Messages', 'Procedures' (which is highlighted in blue), and 'Incidents'. The main area shows an 'Edit Incident - 771601, Fire Detected' screen. The incident details include 'Incident No': 771601, 'Title': 'Fire Detected', 'Status': 'Open', and 'Open Date': '26-03-2025 18:46'. Below this, there's a 'Procedures' section titled 'Workflow - Incident Number 771601'. A modal window titled 'Step 1 of 2' asks 'did you check the alert?' with two options: 'Yes' and 'No'. A 'Submit Selected Option' button is at the bottom of the modal. The background of the main interface shows a map and some incident-related data.

# System Features

## 1. Admin Dashboard

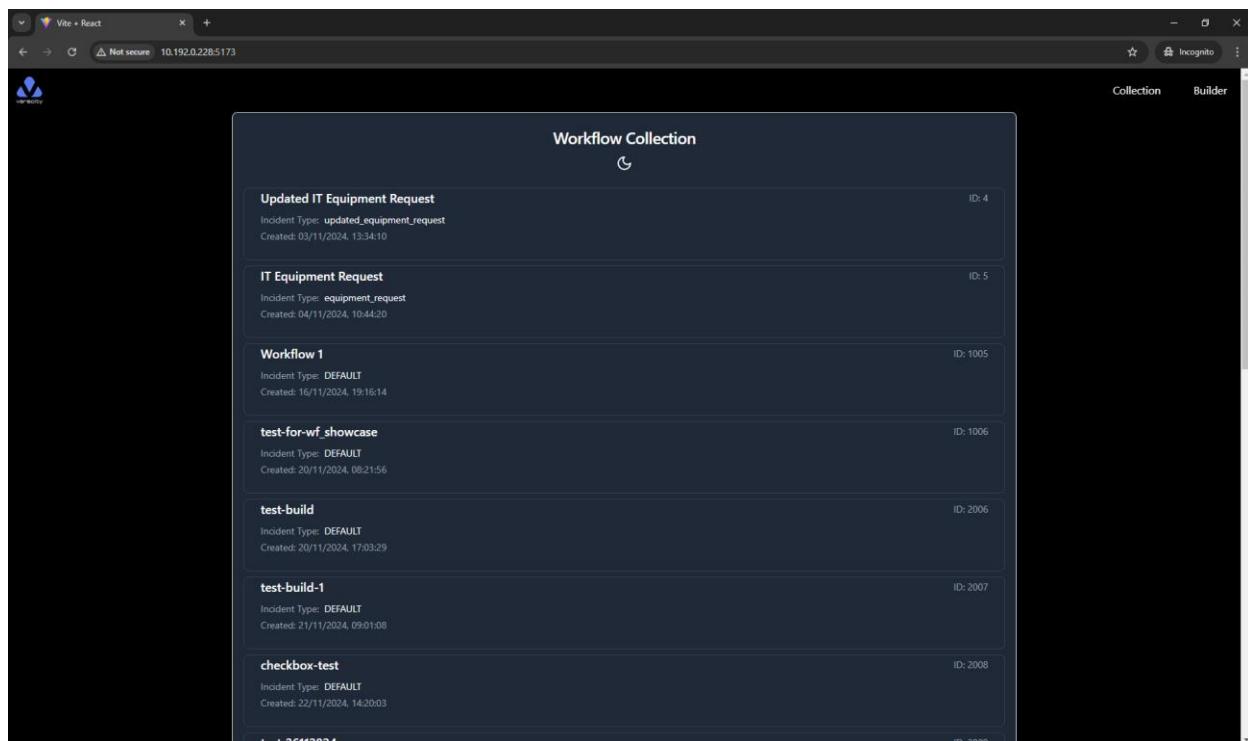
- Create Workflows: Administrators can design workflows that include sequential questions and branching paths based on user inputs.
- Manage Workflows: Admins can edit, delete, or update existing workflows as required.
- Assign and Share Workflows: Admins can share workflow links with users, allowing 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.

The screenshot shows the Veracity Global Workflow Builder interface. At the top, there's a navigation bar with 'Veracity Global' on the left and 'Collection' and 'Builder' on the right. The main area displays a workflow titled 'Workflow 1'. The first question, 'Question ID: 2035', is a 'Multiple Choice' type with the title 'Did you visit the office today?'. It has two options: 'Yes' and 'No'. The 'Yes' option is linked to 'Question 2036', which is also a 'Multiple Choice' type with the title 'Did you connect on teams?'. It has two options: 'Yes' and 'No'. The 'Yes' option under Question 2036 is linked to 'Question 2035', creating a loop. The 'No' option under Question 2036 is linked to 'Question 2037', which is a 'Subjective' type with the title ' '. The 'Required' field is checked for both questions. There are also edit and delete icons for each question card.

## 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.

Enter Incident Number

Submit