ITH-Resilience  
Survey APPLICATION

## Software Required

ITH-Resilience is pleased to submit this proposal for services to support RESILIENCE SURVEY APPLICATION in achieving its goals for setting up & running survey application from scratch

|  |  |
| --- | --- |
|  | * **Python** * **MongoDB** * **ReactJs** * **Git** * **Nginx** * **Uwsgi** |

## System Requirements

* **RAM: 8-12 GB**
* **Disk: 80 GB**
* **CPU: 4**
* **Internet accessibility with Proxy Enabled**
* **OS : RHEL**

## Key Components

* Front-end Webpages were designed and developed using React JS & ChartJS
* Backend functionality & API was developed using Python3
* MongoDB is served as Database engine with Free Data
* The UI Application is hosted on nginx
* The Backend application is hosted on uwsgi
* The complete integrated solution is required for running of the Survey application

## Source Code

Latest Source code for UI & Services have been checked into the below provided Repository.

Access the code with the provided credentials.

|  |  |
| --- | --- |
|  | **Github location 🡪 https://github.com/ResSurveyApp/SurveyApp**  **Username: ResSurveyApp**  **Password: \*\*\*\*\*\*\*\* ## password will be shared through one-one Mail** |

Installation & CONFIGURATION

|  |  |
| --- | --- |
|  | **This Section covers the Installation & configuration of Various components and setup of the Survey application in RHEL instance.** |

### Source Code:

* The entire setup process has to be done either by root or by any user which has highest sudo privileges. It is better to go with sudo user.
* To fetch the source code for UI & Services, git client should be installed. Run the below command to get the GIT client installed

|  |  |
| --- | --- |
|  | **$ sudo yum install git -y** |

* Create a temporary workspace folder and navigate to that folder and run the below commands to fetch the complete latest code of the project

|  |  |
| --- | --- |
|  | **$ sudo cd /**  **$ sudo mkdir /apps**  **$ sudo chown -R <current username> <directory> ## Changing workspace owner**  **$ whoami ### to get the current username**  **$ sudo chown -R <current user from previous command> /apps**  **$ mkdir –p /apps/workspace**  **$ cd /apps/workspace**  **$ git clone** [**https://github.com/ResSurveyApp/SurveyApp.git**](https://github.com/ResSurveyApp/SurveyApp.git)  **$ cd SurveyApp**  **$ mv \* ../** |

### User Interface Setup:

* Install Reactjs, npm (for building the UI code)
* Run the below commands to install the above mentioned software

|  |  |
| --- | --- |
|  | **$ sudo yum install nodejs -y** |

* Once the missing dependencies were updated, run the below command to trigger the build & install the dependencies.

|  |  |
| --- | --- |
|  | **$ cd /apps/workspace/UI**  **$ npm install**  **### If you get this error 🡪 “Could not install from "node\_modules/jspdf/file-saver@github:eligrey/FileSaver.js#e865e37af9f9947ddcced76b549e27dc45c1cb2e" as it does not contain a package.json file.”.### Run below commands.**  **$ npm install jspdf**  **$ npm install**  **$ npm run build** |

* Now we have the latest UI built artifact folder generated in the same location. The output folder (artifact) is generated under ***build*** folder

### Service API Setup:

For the Service API part setup, we need to :

* Install Python3 inside the application server
* Create Virtual environment inside the application server
* Install Python3 and other 45 supporting libraries inside the virtual environment.
* Install uwsgi inside the application server to host the Backend Services
* Usually Python comes by default with RHEL box, if python is not present in the server, Install Python3 by running below commands:

|  |  |
| --- | --- |
|  | **$ sudo yum -y install @development**  **$ sudo yum -y install python3**  **$ sudo yum -y install python3-devel** |

* Now create a virtual environment in the server using below commands:

|  |  |
| --- | --- |
|  | **$ sudo pip3 install virtualenv**  **$ cd /apps/workspace/SERVICES**  **$ virtualenv -p python3 venv** |

* Provide the new virtualname & path which you wanted to create in the above command
* Now activate the created virtual environment by below command:

|  |  |
| --- | --- |
|  | **$ source venv/bin/activate** |

* This will open a new session tty, now you are inside the newly created virtual environment. Install all the required
* dependent and supporting libraries by running the below commands:

|  |  |
| --- | --- |
|  | **$ pip install -U -r requirements.txt**  **$ deactivate ## to exit from the virtual environment** |

## Security Checks:

To allow all the applications in linux communicate internally. Please disable SELINUX.

Step1: Execute below command.

|  |  |
| --- | --- |
|  | **$ sudo sed -i --follow-symlinks 's/SELINUX=enforcing/SELINUX=disabled/g' /etc/sysconfig/selinux** |

Step2: Restart VM.

## Mongo Installation:

Step 1: copy a file from MongoDB/mongodb-org.repo file to /etc/yum.repos.d/ so that you can install MongoDB directly using yum.

|  |  |
| --- | --- |
|  | **$ sudo cp -r /apps/workspace/MongoDB/mongodb-org.repo /etc/yum.repos.d/** |

Step 2: Run below command to install and run mongodb.

|  |  |
| --- | --- |
|  | **$ sudo yum install -y mongodb-org**  **$ sudo cp -r /apps/workspace/MongoDB/mongod.conf /etc/mongod.conf**  **$ sudo systemctl start mongod**  **$ sudo systemctl enable mongod**  **$ mongorestore /apps/workspace/MongoDB/data/** |

## Uwsgi configuration:

|  |  |
| --- | --- |
|  | **$ sudo cp /apps/workspace/Uwsgi/uwsgi.service /etc/systemd/system/**  **$ sudo systemctl start uwsgi.service**  **$ sudo systemctl enable uwsgi.service** |

## Nginx Configuration:

|  |  |
| --- | --- |
|  | **$ sudo yum install nginx -y**  **$ sudo cp -r /apps/workspace/nginx /etc**  **$ sudo systemctl start nginx.service**  **$ sudo systemctl enable nginx.service** |

## Maintenance:

To stop/start the services which are needed to the application, please use below commands:

* sudo systemctl stop/start/restart uwsgi
* sudo systemctl stop/start/restart nginx
* sudo systemctl stop/start/restart mongod

To Verify Status of All Services:

* sudo systemctl status uwsgi ## to check backend service is active or not
* sudo systemctl status nginx ## to check portal is active or not
* sudo systemctl status mongod ## to check database is active or not

## Troubleshooting:

Portal URL : https://<Hostname>

Backend API : https://<Hostname>/backend ## only for debugging

For troubleshooting, you can access the logs in below locations:

* Uwsgi logs @ /apps/ workspace/SERVICES/log/uwsgi.log
* Nginx logs @ /var/log/nginx

**Changing Admin Password:**

Step 1: call below url in any browser with admin current password and new password which you want to change.

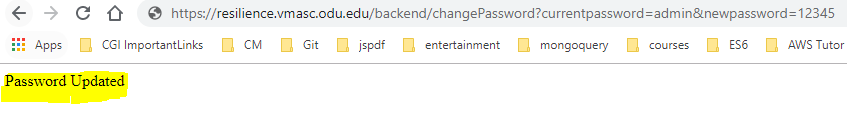
Format :

https://<HostName>/backend/changePassword?currentpassword=<current\_password>&newpassword=<new\_password>

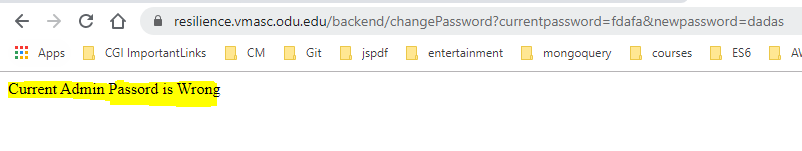
Eg :

https://resilience.vmasc.odu.edu/backend/changePassword?currentpassword=admin&newpassword=new

Step 2: If current admin password is correct, then existing admin password will be replaced by new password.



Step 3: If current admin password is wrong, then password updation will fail.



**Note:** Above method is only for Admin