

# Table of Contents

Table of Contents	1
PGSQL-RUNNER ISSUES	2
1. ISSUES PRIORITIZATION	2
1.1. Issues handling	2
1.1.1. Maintain the Issues sheet	2
1.2. Documentation	2
1.2.1. Create ReadMe file	2
1.2.2. Create the DevOps Guide document	2
1.2.3. Create the Features and Functionalites Guide	2
1.3. Design and Concepts	2
1.3.1. Design loading of csv files	2
1.3.2. Design RESTful API implementation and/or integration	2
1.4. Architectural description	2
1.5. Communication and promotions	2
2. INSTALLATIONS, CONFIGURATIONS AND INTEGRATIONS	3
2.1. Installations	3
2.2. Configuration management	3
2.3. External Integrations	3
3. DEVELOPMENT	3
3.1. Rename ProductVersionDir to ProductInstanceDir	3
3.2. Bug fixing	3
3.2.1. fix exit code not passed to test func	3
3.3. Features development	3
3.4. Functionalities development	3
4. TESTING	3
4.1. Automated testing	3
5. QUALITY ASSURANCE	3
5.1. Features quality assurance	4
5.2. End-to-end quality assurance	4
5.3. Functionalities quality assurance	4
6. PRODUCTION DEPLOYMENT	4
7. MAINTENANCE AND OPERATIONS	4
7.1. Keep tool's documentation up-to-date	4
7.2. Configuration management	4
7.3. Regular maintenance tasks	4

# PGSQL-RUNNER ISSUES

## 1. ISSUES PRIORITIZATION

### 1.1. Issues handling

In this section all the tasks and activities related to the issues handling and management are collected.  
The issues handling

#### 1.1.1. Maintain the Issues sheet

The Issues sheet contains the list of all the issues on this tool, with priority attribute for easier prioritization.  
The issues also have a level attribute for hierarchy.  
Different issues might be of different types - bugs, activities, features etc., however so that each issue should have one and only one deliverable and owner.

### 1.2. Documentation

#### 1.2.1. Create ReadMe file

The ReadMe file should provide quick and user-friendly overview on the basic functionality of the tool, as well as guide to the further documentation.

#### 1.2.2. Create the DevOps Guide document

The DevOps Guide document should provide information on all the possible DevOps aspects related to the tool - installation, maintenance, configurations, development etc.

#### 1.2.3. Create the Features and Functionalities Guide

The features and functionalities Guide should provide a structured description on all the features and functionalities of the tool and their mapping so that the tool's user should know what to test for.

### 1.3. Design and Concepts

Concepting and Planning

#### 1.3.1. Design loading of csv files

How-to load csv files

#### 1.3.2. Design RESTful API implementation and/or integration

Design for RESTful API Implementation and/or integration :  
<https://github.com/QBisConsult/pgsql-api>

### 1.4. Architectural description

The tasks and activities related to the architecture of the tool.

### 1.5. Communication and promotions

Tasks and activities related to the communication and the promotions of the tool.

## 2. INSTALLATIONS, CONFIGURATIONS AND INTEGRATIONS

Tasks and activities related to the installations , configurations and intetrations for the components of the Service.

### 2.1. Installations

Any installation related issues are described within this section.

### 2.2. Configuration management

Configuration management

### 2.3. External Integrations

External Integrations

## 3. DEVELOPMENT

Bugs fixing. Development of features and functionalities.

### 3.1. Rename ProductVersionDir to ProductInstanceDir

Rename ProductVersionDir to ProductInstanceDir.

Rename product\_version\_dir to product\_instance\_dir  
to reflect the pgsql-runner model

```
sfw/bash/pgsql-runner/.doc-pub-host.files-to-backup.lst
```

### 3.2. Bug fixing

This activity describes the fixing of bugs

```
sfw/bash/pgsql-runner/.doc-pub-host.files-to-backup.lst
```

#### 3.2.1. fix exit code not passed to test func

When calling with the test func the script could not return the correct exit code - this has been fixed with the correct trap call.

```
trap "exit $exit_code" TERM
```

### 3.3. Features development

This activity describes the features developed in the Product of the Service.

### 3.4. Functionalities development

This activity describes the functionalities developed in the Product of the Service.

## 4. TESTING

### 4.1. Automated testing

All the testing issues performed via automation

## 5. QUALITY ASSURANCE

### **5.1. Features quality assurance**

### **5.2. End-to-end quality assurance**

### **5.3. Functionalities quality assurance**

## **6. PRODUCTION DEPLOYMENT**

## **7. MAINTENANCE AND OPERATIONS**

### **7.1. Keep tool's documentation up-to-date**

Keep tool's documentation up-to-date

### **7.2. Configuration management**

Configuration management

### **7.3. Regular maintenance tasks**

Regular maintenance tasks