Detail Design Document

For IOT BASED ALERT SYSTEM TO HIGHLIGHT PROCESS FAILED

Votary Softech Solutions Pvt. Ltd.

Plot No: 76, Lumbini layout,   
Near Euro school,   
Gachibowli-I (V), Hyderabad,  
Telangana - 500032,   
India.

**Revision History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version (x.y) | Date of Revision | Description of Change | Reason for Change | Affected Sections | Approved By |
| 1.0 | 18-07-2017 | New Definition | New | ALL | Mohan |
| 2.0 | 02-08-2017 | Designed for multiple repositories | To improve capability |  |  |
|  |  |  |  |  |  |

**Approval History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version (x.y) | Prepared By | Reviewed By/Date | Approved By/Date |
| 1.0 | Ashok L | Arun/20-07-2017 | Mohan/20-07-2017 |
| 2.0 | Ashok L |  |  |
|  |  |  |  |

**Contents**

[1 Define 4](#_Toc481588852)

[1.1 Overview 4](#_Toc481588853)

[1.2 Assumptions 4](#_Toc481588854)

[1.3 Limitations 4](#_Toc481588855)

[1.4 Glossary 4](#_Toc481588856)

[2 Specify 4](#_Toc481588857)

[2.1 Scope 4](#_Toc481588858)

[2.2 Design Approach 5](#_Toc481588859)

[2.3 Alternative Design Approaches Considered 5](#_Toc481588860)

[2.4 Module Design 5](#_Toc481588861)

[2.4.1 Module Name and Description 5](#_Toc481588862)

[2.5 External Interfaces 6](#_Toc481588863)

[2.6 Design Tools 6](#_Toc481588864)

[3 Architecture 6](#_Toc481588865)

[3.1 Block Diagrams 6](#_Toc481588866)

[3.2 Activity Diagrams/ Flow Charts 6](#_Toc481588867)

[3.3 Deployment diagrams 6](#_Toc481588868)

[3.4 State Transition Diagram (Optional) 6](#_Toc481588869)

[3.5 State Transition Table 7](#_Toc481588870)

[4 Design 7](#_Toc481588871)

[4.1 Class Diagrams 7](#_Toc481588872)

[4.2 Sequence Diagrams 7](#_Toc481588873)

[4.3 Database Design 7](#_Toc481588874)

[4.4 Wireframes 7](#_Toc481588875)

[5 Implementation 7](#_Toc481588876)

[5.1 Directory Structure 7](#_Toc481588877)

[5.2 Libraries and Framework 7](#_Toc481588878)

[5.3 Error Handling 7](#_Toc481588879)

[5.4 Status Codes and Description 7](#_Toc481588880)

[6 Validate 8](#_Toc481588881)

[7 Deploy 8](#_Toc481588882)

[8 Maintain 8](#_Toc481588883)

# Define

## Overview

The purpose of this document is to do the high level design by identifying the different modules and their interfaces. The modules involved in the design are Source code version control tool, Build and integration with jenkins and alarm notification. Developers are at other end who modifies and commits to the version control. But there may more than one repository from the different servers or from the same server.

## Assumptions

B&I Tool and Remote Client Should have same e-mail pre-configured. Assuming there are N number of repositories from same server/different server.

## Limitations

If we get more than one build failure at a time we are able to give one alarm that can be closed by the user. Number of pop- up displayed is depending on the number of build fails which carries the message of corresponding build.

## Glossary

| Term | Definition |
| --- | --- |
| B&I | Build and integration |
|  |  |

# Specify

All the modules may be located in the different network.

## Scope

The modules can be change throughout life cycle vgate process and updated with corresponding documents with specified changes. Estimations and budget are not applicable.

## Design Approach

The source code version control tool at server carries the developers code which can be check-in by corresponding developers. There are single server/multiple servers with single repository/multiple repositories. In continuous integration the code is cloned from the server when developer checked-in and build the code then send notification to the application when it fails, If build pass then continue with the test cases.

There is simultaneous check-in can be handled by the jenckins and builds the jobs and create the reports. Once test cases done jenkins sends test report to the corresponding developer through email .

Build and integration:

It can be done by using Jenkins (open source). Implementation will be done in python programming.

Alarm system:

It runs on the host system. It will be implemented in python program.

## Alternative Design Approaches Considered

It can also be done using other devops like Teamcity, TravisCI, GoCD, Bamboo, GitLab and etc. In this project Jenkins preferred over above tools, Jenkins are the open source, easy to understand and good graphical user interface.

## Module Design

### Module Name and Description

a. Source code version control system

It Stores the source code which is check-in by the developer. Every commit consist of the developer name and email ID. This information can be utilized by the Jenkins when ever the commit happen.

b. Build and integration

The build and integration mainly consist of the below sub modules:

i) Jenkins : It always seeks the source code version control tool for new commit, once commit done by the developer jenkins clone the code and intimates to the build system. Once B&I generates the report it sends email to the corresponding developer and alarm signal to the alarm system.

ii) Build system: When jenkins intimate to the build system it build the code and generates the report. Its output format will be shown below.

**Output format of build failed:**

Build failed in jenkins: <project\_name> #<build number>

Ex: Build failed in Jenkins: VITA\_DM\_TEST #425

**Output format of build success:**

Build success in jenkins: <project\_name> #<build number>

Ex: Build success in Jenkins: VITA\_DM\_TEST #425

iii) Test-environment: It consist of the different test cases to be applied to the source code. Prepared by the test engineer and it can be utilized by the build system.

c. Alarm system

The alarm system located some where(ex: raspberry pi), which can able to receive the mail from the jenkins and it gives the alarm as well as display error message.

#### Class Diagram

NA

#### API in Module

<Inputs needs from developers/Need to be filled by developers>

#### Module Services

<Inputs needs from developers/Need to be filled by developers>

#### Module Functions

<Inputs needs from developers/Need to be filled by developers>

## External Interfaces

External interfaces used in the project are alarm speaker and Third party/open source code jenkins used.

< Contact Persons for specific tasks at

- Company Name xxxxxxx

- The customer organization

- Sub-contracted organization

- Other organizational entities that interact with the project. >

|  |  |
| --- | --- |
| Organization | Liaison/ Interface |
|  |  |

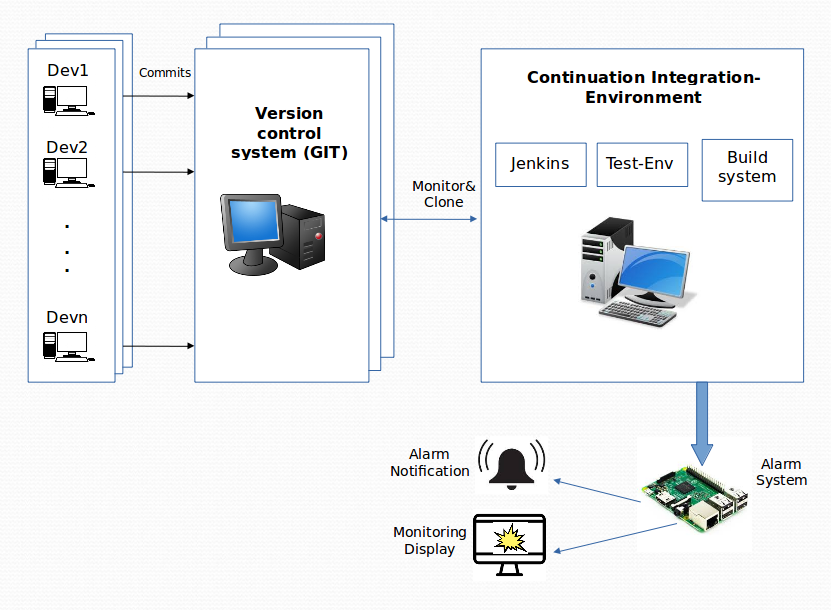
## Design Tools

List of the design tools used:

* Git
* Jenkins
* Python
* Gcc
* Scons

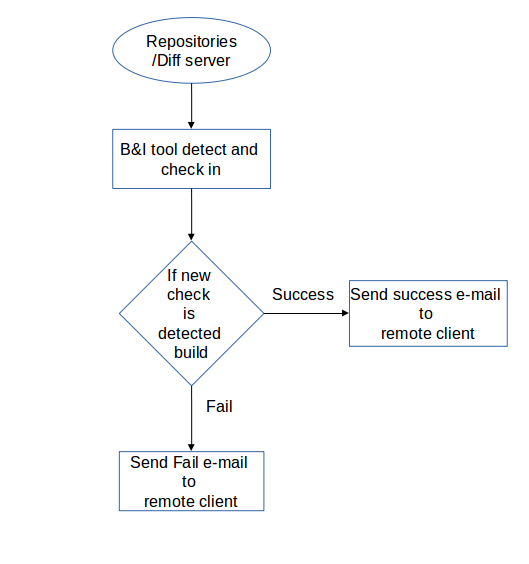
# Architecture

## Block Diagram

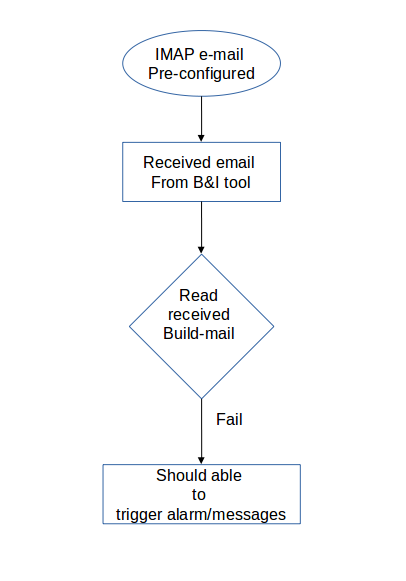


## Activity Diagrams/ Flow Charts

Flow Chart for B&I



Flow Chart for Remote Client



## Deployment diagrams

NA

## State Transition Diagram (Optional)

NA

## State Transition Table

NA

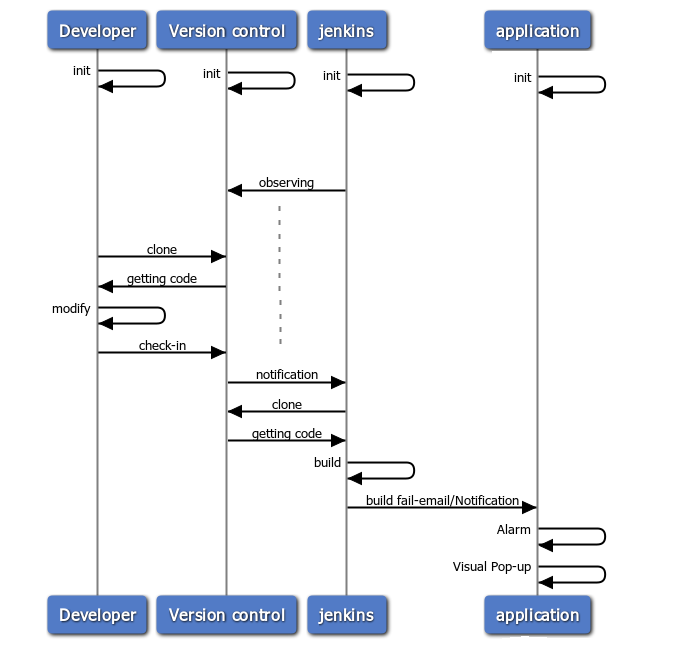
# Design

## Class Diagrams

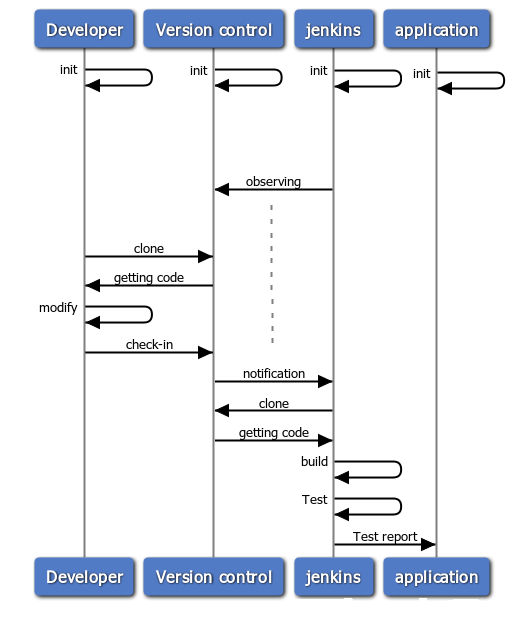
NA

## Sequence Diagrams

Build Fail case:



Build Pass case:



## Database Design

NA

## Wireframes

NA

# Implementation

<Implementation details of the project.>

## Directory Structure

<Directory Structure for Project Repository for source code, documents, Reports etc >

## Libraries and Framework

<Give the details of Libraries, SDK or framework classes used from internal or 3rd parties to construct the project base. Packaging of libraries and their reference documentation>

## Error Handling

<Applicable for Module services, Success, Retry count, Failure error codes, that will be sent as response to clients/modules using the service.>

## Status Codes and Description

<List the Success/Warning/Failure Error codes that will be sent as response to clients/modules using the service and their description and severity.>

# Validate

<References to Checklists, Checklists for Detail Design Document, Review defect logs, Approval Emails, Traceability >

# Deploy

<Commit Base-lined Design Documents to SVN, Configure all the tools and raise Requests for Training, hardware, software and resources identified in the Plan>

# Maintain

<Changes to Design with appropriate references to CRs, Enhancements etc,>