Rubaiyat Jahan Mumu

BSSE0620

7/11/2016

Submitted to

Dr. Kazi Muheymin-Us-Sakib  
Professor and Director  
Institute of Information Technology  
University of Dhaka

##### Deadlock

###### In Distributed System

Table of Contents

[1. Analysis Class 2](#_Toc495138750)

[1.1 Design Components 4](#_Toc495138751)

[2. Infrastructure Class 6](#_Toc495138752)

[3 Elaborated Design Class 7](#_Toc495138753)

[3.1 Collaboration Details 9](#_Toc495138754)

[3.2 Appropriate Interfaces 10](#_Toc495138755)

[3.3 Elaborate Attributes 11](#_Toc495138756)

[3.3 Describe Processing Flow 12](#_Toc495138757)

[4. Persistent Data 15](#_Toc495138758)

[5. Elaborative Deployment 15](#_Toc495138759)

[Figure 1: Design component of System Admin Class 4](#_Toc495138875)

[Figure 2: Design component of Student Class 4](#_Toc495138876)

[Figure 3: Design Component of Registrar Class 5](#_Toc495138877)

[Figure 4: Design Component of Program Office 5](#_Toc495138878)

[Figure 5: Design Component of UGC Stuff Class 6](file:///D:\Project%20Management%20Project\graduate-verification\documentation\Component%20Design\Deadlock.docx#_Toc495138879)

[Figure 6: Elaborated Design of System Admin Class 7](#_Toc495138880)

[Figure 7: Elaborated Design of Student Class 8](#_Toc495138881)

[Figure 8: Elaborated Design of Registrar Class 8](#_Toc495138882)

[Figure 9: Elaborated Design of Program Office Class 9](#_Toc495138883)

[Figure 10: Elaborated Design of UGC Class 9](#_Toc495138884)

[Figure 11: Collaboration Details of System Admin Class 10](#_Toc495138885)

[Figure 12: Collaboration Details of Program Office 10](#_Toc495138886)

[Figure 13: Collaboration Details of Registrar Class 11](#_Toc495138887)

[Figure 14: Processing Flow of Verification System 15](#_Toc495138888)

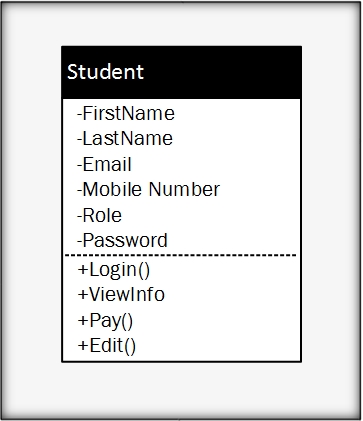
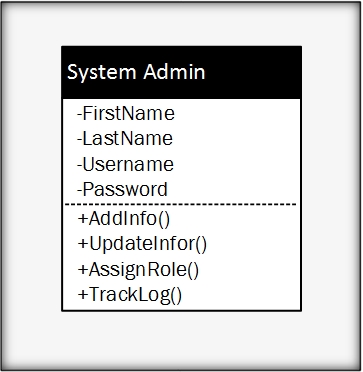
[Figure 15: Elaborative Deployment of OGRVS 16](#_Toc495138889)

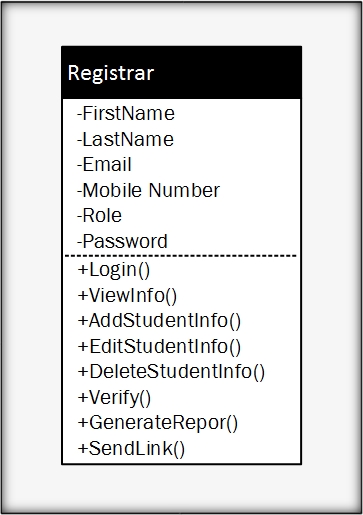
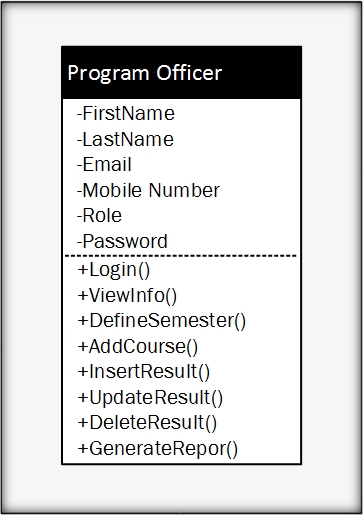
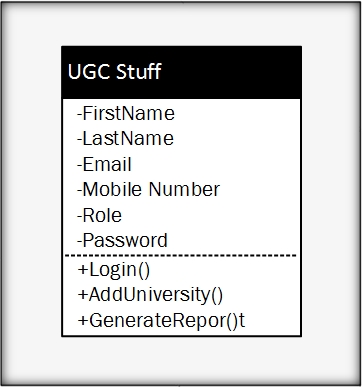
Component Level Design

# Analysis Class

This is the first step of object-oriented approach in component level design. In this section, all design classes that correspond to the problem domain as defined in the analysis model and architectural model are identified. The analysis classes for this project are :

1. System Admin
2. Student
3. UGC Stuff
4. Registrar
5. University/Department Program Office
6. Foreign Universities
7. Database





## 1.1 Design Components

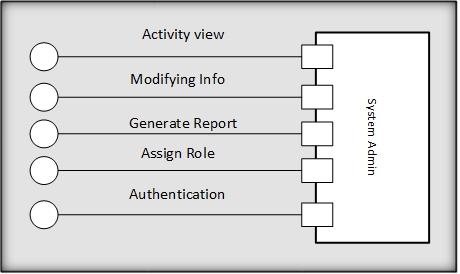
Following is the design components of the analysis classes:

Figure 1: Design component of System Admin Class

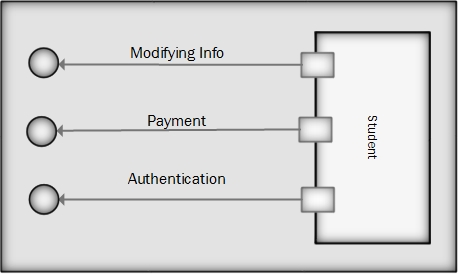


Figure 2: Design component of Student Class

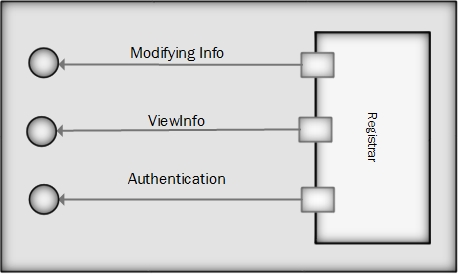


Figure 3: Design Component of Registrar Class

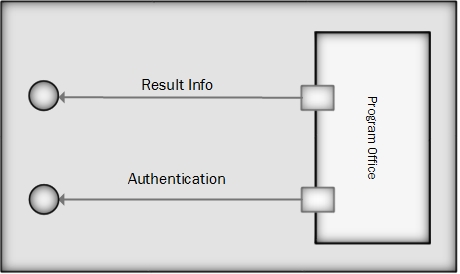


Figure 4: Design Component of Program Office

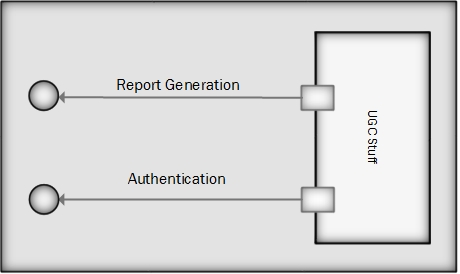


Figure 5: Design Component of UGC Stuff Class

# 2. Infrastructure Class

In this section, all design classes that correspond to the infrastructure domain are identified and listed. These classes are usually not present in the analysis or architectural models. These classes include GUI components, operating system components, data management components, networking components, etc. For OGRVS, the infrastructure classes are listed below:

1. Report
2. Course
3. Semester
4. Result
5. DAL
6. University

# 3 Elaborated Design Class

The elaborated design classes are the following:

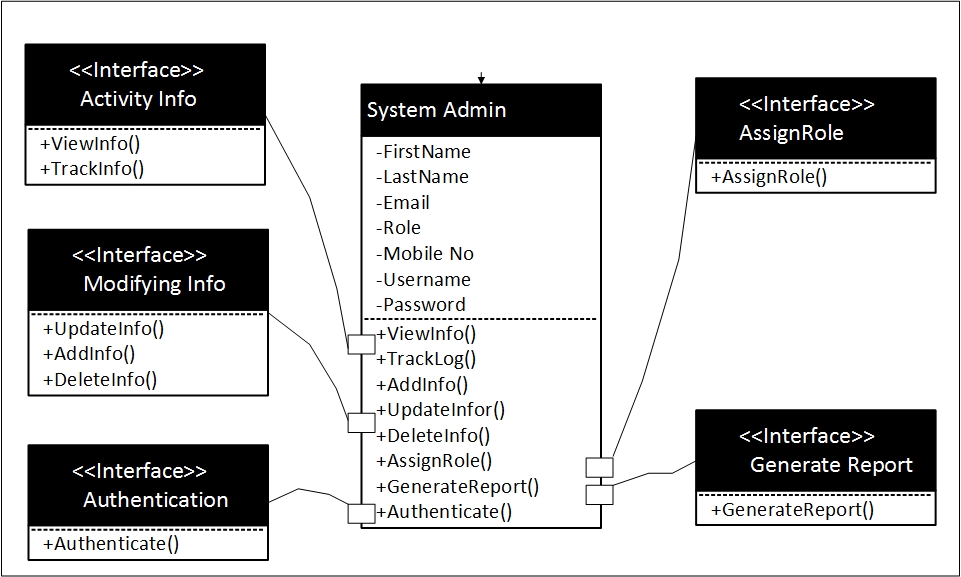


Figure 6: Elaborated Design of System Admin Class

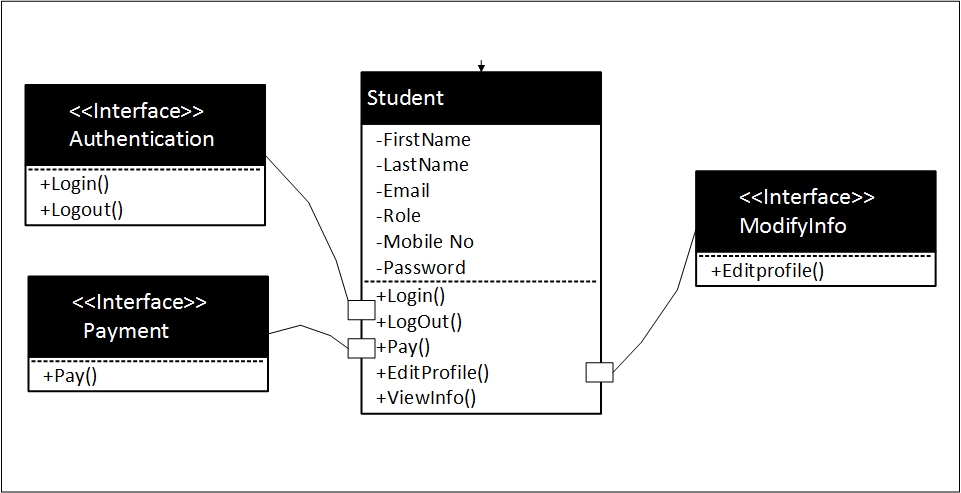


Figure 7: Elaborated Design of Student Class

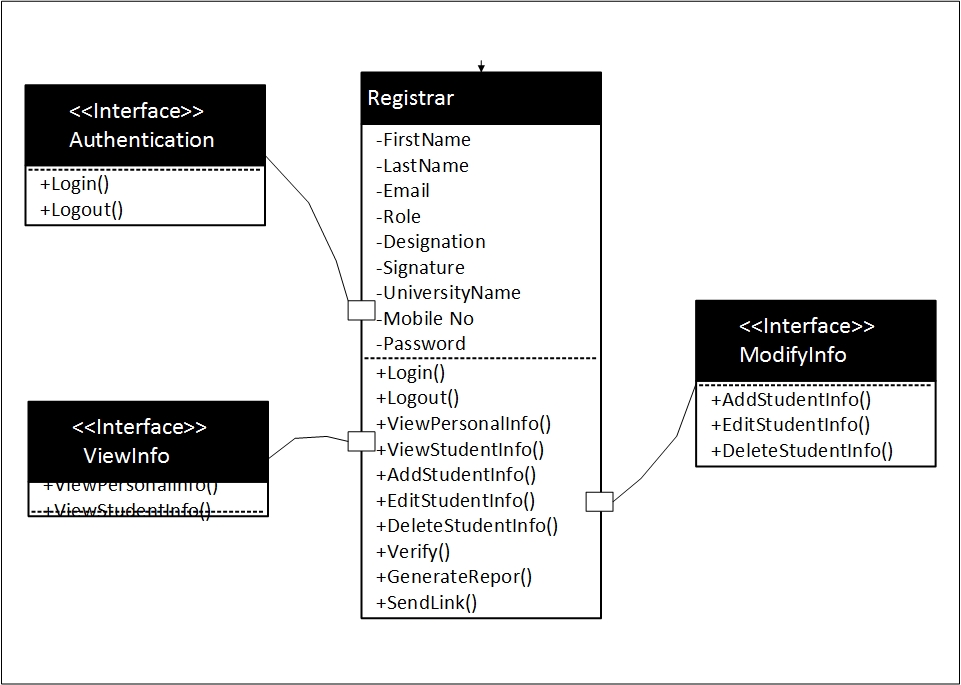


Figure 8: Elaborated Design of Registrar Class

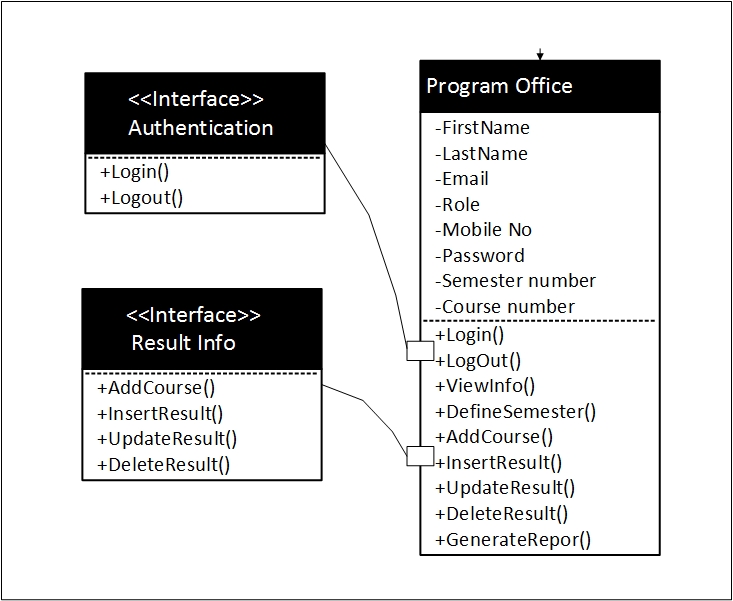


Figure 9: Elaborated Design of Program Office Class

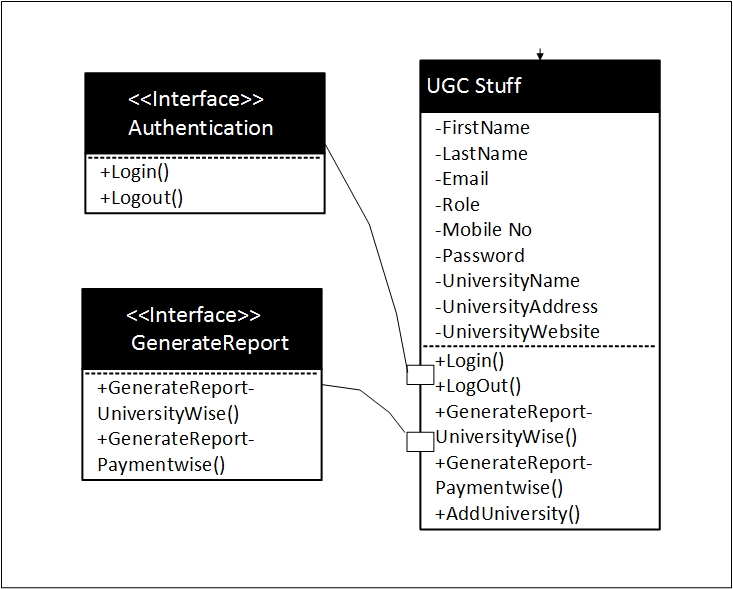


Figure 10: Elaborated Design of UGC Class

## 3.1 Collaboration Details

In this section, the collaboration between classes will be designed. Following is the diagrams:

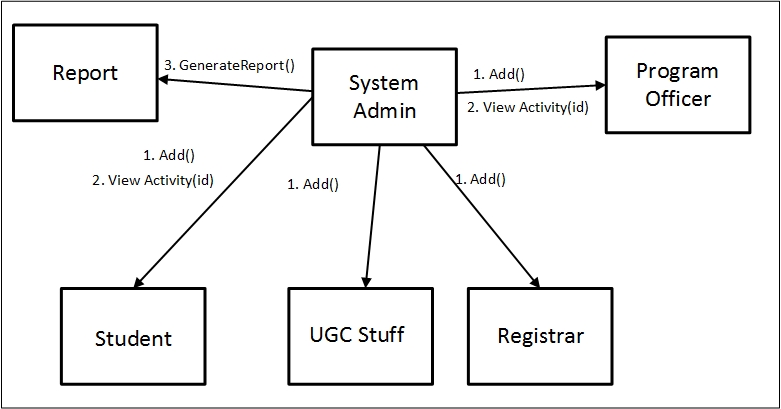


Figure 11: Collaboration Details of System Admin Class

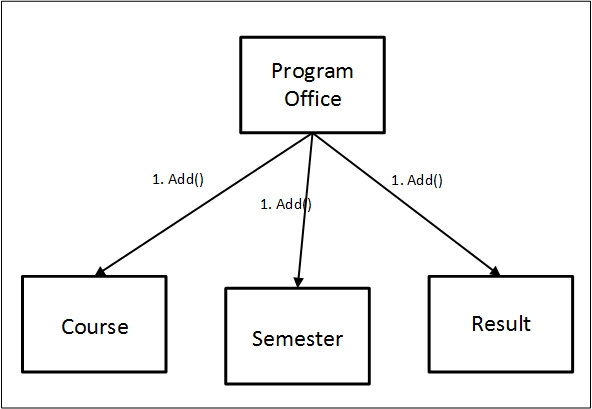


Figure 12: Collaboration Details of Program Office

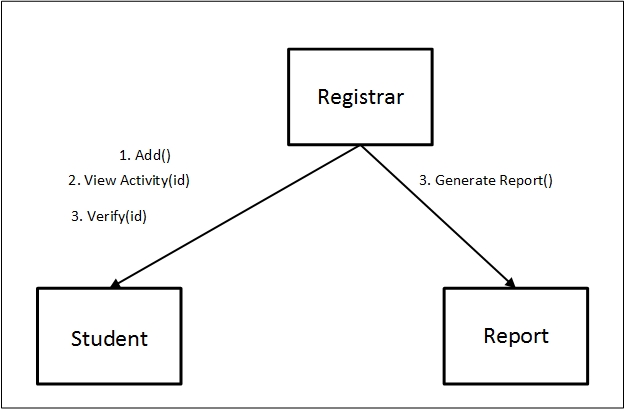


Figure 13: Collaboration Details of Registrar Class

## 3.2 Appropriate Interfaces

There is no necessity to divide the classes in subclasses as they exhibit sufficient cohesion. So there is no need to define appropriate interfaces.

## 3.3 Elaborate Attributes

Elaborate attributes for System **Admin class**:

First-name: string = “name1” {contains name value – Rubaiyat, Tasnim, Farzana}

Last-name: string = “name2” {contains name value – Mumu, fabiha, Muna}

Email: email = “null” {Contains value – [rubaiyat@gmail.com](mailto:rubaiyat@gmail.com), tasnim@hotmail.com}

Role: string = “null” {contains name value – student, admin, PO}

Username: string = “username1” {All Characters}

Password: string= “password1” {All Characters}

Mobile Number: Number = “088 01716724859” {contains name value – 088-01711467599}

Elaborated attributes for **Student class**:

First-name: string = “name1” {contains name value – Rubaiyat, Tasnim, Farzana}

Last-name: string = “name2” {contains name value – Mumu, fabiha, Muna}

Email: email = “null” {Contains value – [rubaiyat@gmail.com](mailto:rubaiyat@gmail.com), tasnim@hotmail.com}

Role: string = “role1” {contains name value – student, admin, PO}

Password: string= “null” {All Characters}

Mobile Number: Number = “088 01716724859” {contains name value – 088-01711467599}

Elaborated attributes for **Registrar class**:

First-name: string = “name1” {contains name value – Rubaiyat, Tasnim, Farzana}

Last-name: string = “name2” {contains name value – Mumu, fabiha, Muna}

Email: email = “null” {Contains value – [rubaiyat@gmail.com](mailto:rubaiyat@gmail.com), tasnim@hotmail.com}

Role: string = “null” {contains name value – student, admin, PO}

Password: string= “password1” {All Characters}

Mobile Number: Number = “088 0000000000” {contains name value – 088-01711467599}

University-name: String= “null” {Contains name value – Dhaka University, Chittagong university}

Designation: string = “null” {Contains name value- Manager, Senior manager}

Elaborated attributes for **Program Office** class:

First-name: string = “name1” {contains name value – Rubaiyat, Tasnim, Farzana}

Last-name: string = “name2” {contains name value – Mumu, fabiha, Muna}

Email: email = “null” {Contains value – [rubaiyat@gmail.com](mailto:rubaiyat@gmail.com), tasnim@hotmail.com}

Role: string = “null” {contains name value – student, admin, PO}

Password: string= “password1” {All Characters}

Mobile Number: Number = “088 0000000000” {contains name value – 088-01711467599}

Semester number: Int = “1” { automatically increments – 1,2,3,….. }

Course number: Int = “1” { automatically increments – 1,2,3,….. }

Elaborated attributes for **UGC Stuff** class:

First-name: string = “name1” {contains name value – Rubaiyat, Tasnim, Farzana}

Last-name: string = “name2” {contains name value – Mumu, fabiha, Muna}

Email: email = “null” {Contains value – [rubaiyat@gmail.com](mailto:rubaiyat@gmail.com), tasnim@hotmail.com}

Role: string = “null” {contains name value – student, admin, PO}

Password: string= “password1” {All Characters}

Mobile Number: Number = “088 0000000000” {contains name value – 088-01711467599}

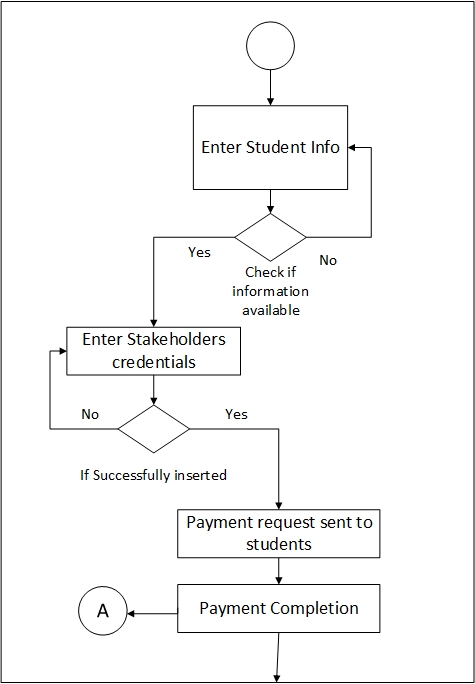
University-name: String= “null” {Contains name value – Dhaka University, Chittagong university}

University-address: String = “null” {All Characters}

University-website: String = “null” {All Characters – www.du.ac.bd}

## 3.3 Describe Processing Flow

The processing flow of the verification system is described below:



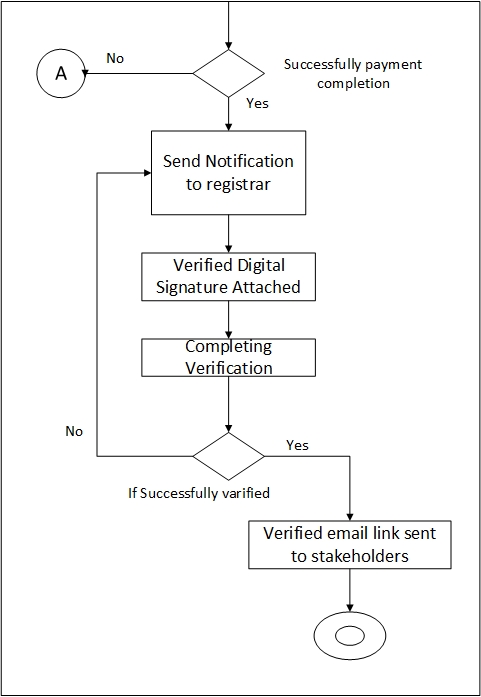


Figure 14: Processing Flow of Verification System

# 4. Persistent Data

The persistent data sources (databases and files) and the classes required to manage them are following:

**Persistent data source**: Database

**Classes to manage data source**: Database classes for each entity

# 5. Elaborative Deployment

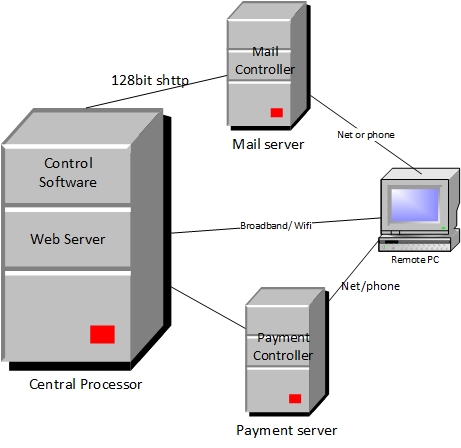
****

Figure 15: Elaborative Deployment of OGRVS