**Software Architectural Design Document**

**For**

**Module 9: Accountant Delegate Task List**

**Version 1.0**

**Prepared by : 1. Siti Nur Nihlah Binti Mohd Asly 177319**

**2. Suliana Binti Said 176593**

**3. Rosninawati Binti Roslee 179196**

**4. Nurul Syahnia Binti Abdul Agis 178788**

**5. Mira Azira Binti Jiwara 177652**

**Table of Contents**

**Table of Contents……………………………………………………………………….….....2**

1. **Architectural Representation…………………………………………………………....3**
2. **ERD Diagram………………………………………….……………………………….3-4**

**3.0** **Conceptual Diagram…………………….……………………………………………………….….....5**

3.1 Reference Architectural Diagram……………..…..…………………………….……..5

**4.0** **Use Case View………………………………………………………………………….…6**

4.1 Use Case Description.…………………………..…………………………………......6

**5.0** **Logical View ……………………..……………………………..…………………...........7**

5.1 Class Diagram of Accountant Delegate Task List module………………………......7-8

**6.0 Process View ………………………………………………………...................................8**

6.1 Activity Diagram……………………………………………………………………....8

**7.0 Physical View…………………………………………………………..……………........9**

7.1 Deployment Diagram………………………………...……………………………......9

**8.0** **Development View ………………………………………………………………..….....10**

8.1 Component Diagram ………………………………..……………………………….10

**9.0** **Architectural Pattern .…..…………………………………………………...................11**

9.1 Model View Controller ..…….…………………………….………………………... 11

**10.0 References …………………………………………..………..…………………….…...…..****12**

1. **Architectural Representation.**

This document presents the architecture as a series of views; use case view, logical view, process view and deployment view. There is no separate implementation view described in this document. These are views on an underlying Unified Modeling Language (UML) [1] model developed using Rational Rose.

1. **Entity Relational Diagram (ERD).**



Figure 1.0 shows ERD diagram for Accountant Delegate Task List Module.

This diagram shows in Accountant Delegate Task List module have four(4) Entities which includes:

1. Accountant.
2. File.
3. Task.
4. Account.

For Accountant Entity it has several attributes which are:

1. Acc\_id : Account Id act as the primary key
2. Name.
3. Email.
4. Password.

For File Entity, it also has several attributes which includes:

1. File Name : File Name act as the primary key
2. Priority.
3. Date.
4. Type.
5. Size..

For Task Entity, it has various attributes which are :

1. Task\_Title : Task Title act as the primary key
2. Task\_Desc.
3. Status.
4. Date.

For Account Entity, it has few attributes which include:

1. Code : Code act as the primary key
2. Name.
3. Type.

**3.0 Conceptual Diagram**

**3.1 Reference architectural diagram for Digital Ground Drill Mobile Application Solution.**

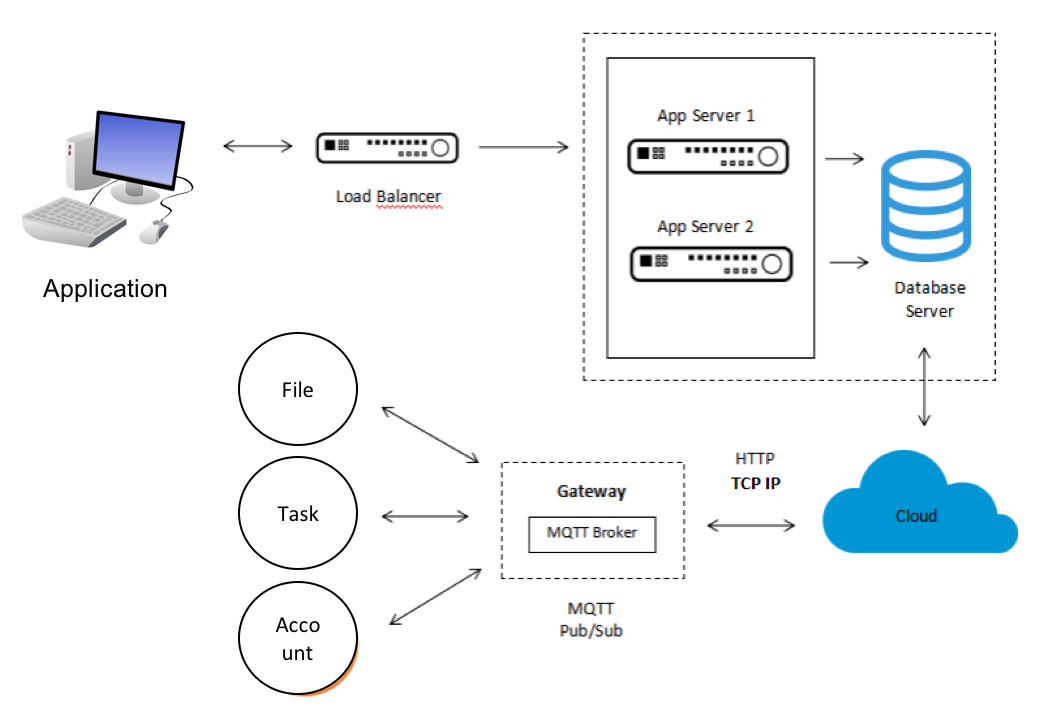


Diagram 2.0 shows the conceptual diagram for Accountant Delegate Task List module.

This Conceptual Diagram shows how the how the backend module works.. Accountant delegate task list module is connected to the Load Balancer that acts as a reverse proxy and distribute network or application traffic across a number of servers. We use two application servers, which is App Server 1 and App Server 2 which will connected to the Database server as a storage platform. The Database Server connected to the cloud. The cloud connected to the MQTT Broker [2] through HTTP [3] or TCP/IP protocol [4]. This MQTT Broker is connected to the three component which are the file, task and also the account.

* 1. **Use Case View**



Diagram 3.0 shows Use Case Diagram for Accountant Delegate Task List module.

**4.1 Use Case Description**

|  |  |  |
| --- | --- | --- |
| **ID** | **Contained in Use Case(s)** | **Description** |
| UC1 | View all files | The system allow the accountant to view all files assigned. |
| UC2 | Sort Files | The system allow the accountant to sort file based on the priority. |
| UC3 | Read task | The system allow the accountant to read or open the task assigned. |
| UC4 | Update task | The system allow the accountant to change and update the task assigned. |
| UC5 | View account | The system allow the accountant to view the account. |
| UC6 | Open account | The system allow the accountant to open up the account. |

**5.0 Logical View**

**5.1 Class Diagram of Accountant Delegate Task List module.**

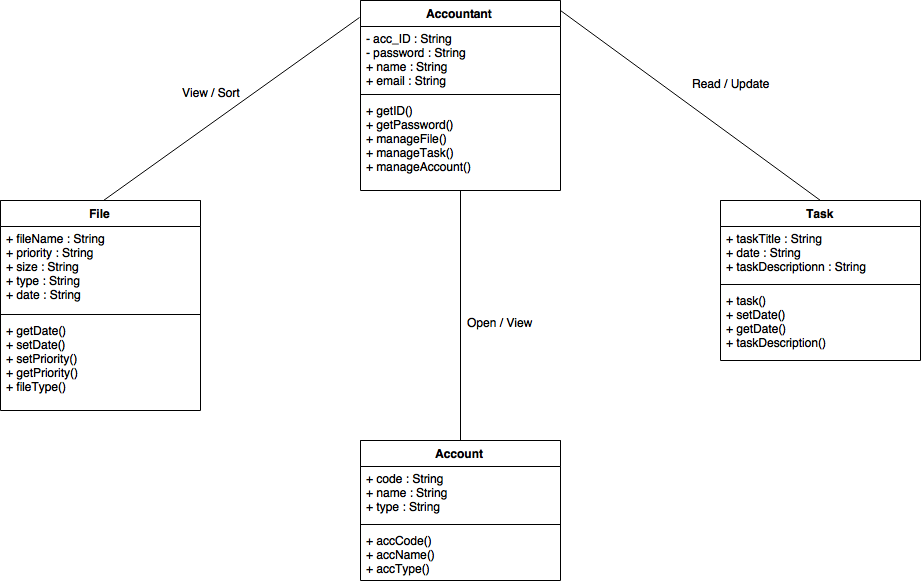
****

Diagram 4.0 shows the class diagram for Accountant Delegate Task List module.

Diagram above shows the class diagram for Accountant Delegate Task List module. From the class diagram, we can describes the structure of a system by showing the system’s classes, their attributes, operations (or methods), and the relationships among objects. Based on the diagram, we see that there are four (4) classes which includes the Accountant, File, Task, and also the Account. Each class has their own attributes and operation. Basically the relationship between classes can be describe through:

1. Accountant can view all files had been assigned.
2. Accountant can sort the files based on its priority.
3. Accountant can read the task had been assigned before.
4. Accountant can update the task.
5. Accountant can view the account.
6. Accountant also can open the account.

**6.0 Process View**

**6.1 Activity Diagram**

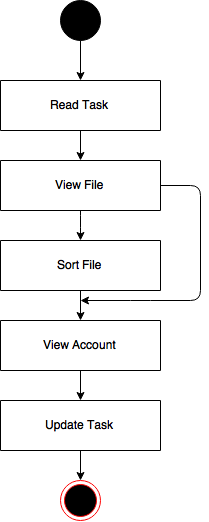
****

Diagram 5.0 shows the activity diagram for the Accountant Delegate Task List module

Based on the activity diagram above, we manage to see the flow of the Accountant Delegate Task List module. It begins with start, then accountant will go to read task. After reading the task, account might view file. After viewing the file, accountant may go to either to sort file or go to view account. After accountant finish sort the file the will view the account and the update the task. After the task have been updated the activity ends

**7.0 Physical View**

**7.1 Deployment Diagram.**

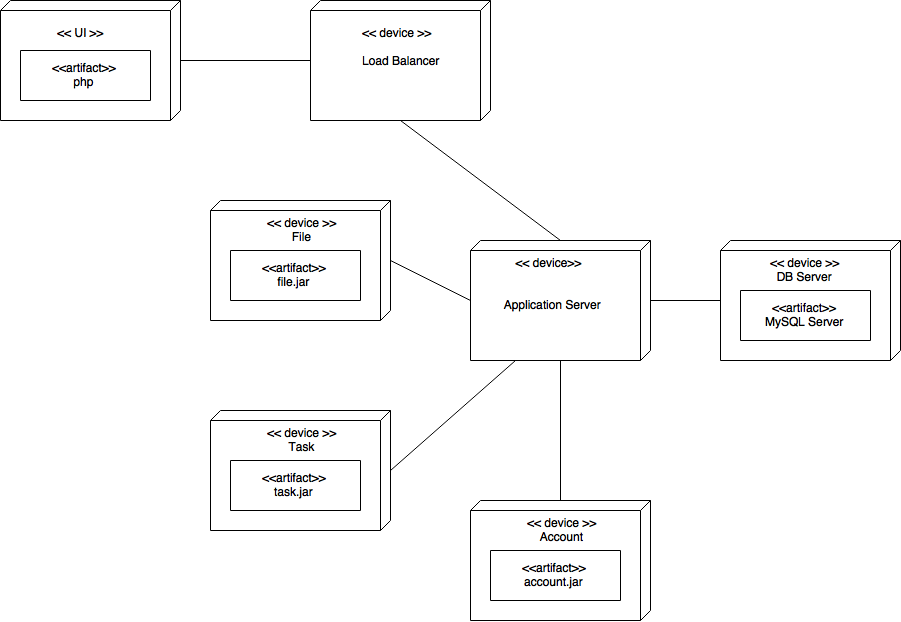
****

Diagram 6.0 shows the Deployment diagram of Accountant Delegate Task List.

This diagram shows execution architecture of systems that represent the deployment of module artefact to deployment targets. Based on the diagram, this module have 7 nodes, 4 artifacts, one application server, one database server and one load balancer. Three component is connected to Application Server which are File, Task and Account.

**8.0 Development View**

**8.1 Component Diagram.**

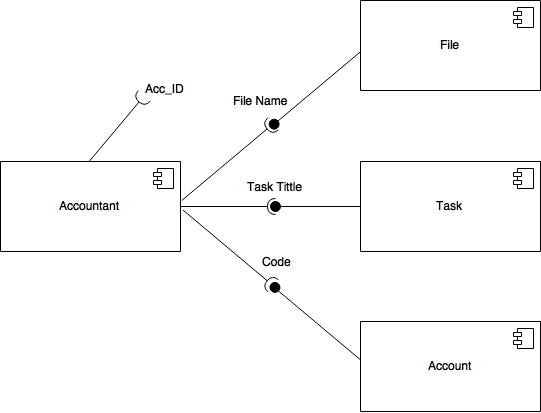
****

Diagram 7.0 shows the component diagram of Accountant Delegate Task List module.

Diagram above shows the component involve in the Accountant delegate task list. Accountant required Accountant id. Accountant also required information from File, Task and Account component. While the File , Task and Account provide information to the accountant.

**9.0 Architectural Pattern**

**9.1 Model-View-Controller (Mvc)**

This MVC [5] architectural pattern objective is to separate between information, presentation and user interaction. Thus it include 3 variable which is model, view and controller. When a model object value changes, a notification is sent to the view and to the controller. So that the view can update itself and the controller can modify the view if its logic so requires.

When handling input from the user the windowing system sends the user event to the controller; if a change is required, the controller updates the model object. For our project, Model is the permanent storage which is the related database, View is the html displayed and the Controller is the user data input.



Figure 8.0 shows the MVC pattern use for Accountant Delegate Task List.

**10.0 References**

[1] H. (2016, April 20). UML Diagram Types With Examples for Each Type of UML Diagrams. Retrieved January 12, 2017, from <http://creately.com/blog/diagrams/uml-diagram-types-examples/>

[2]H. (2015, December 20). MQTT Essentials Part 3: Client, Broker and Connection Establishment. Retrieved January 12, 2017, from <http://www.hivemq.com/blog/mqtt-essentials-part-3-client-broker-connection-establishment>

[3]Beal, V. (n.d.). HTTP - HyperText Transfer Protocol. Retrieved January 12, 2017, from <http://www.webopedia.com/TERM/H/HTTP.html>

[4]“What is TCP/IP (Transmission Control Protocol/Internet Protocol)? - Definition from WhatIs.com." *SearchNetworking*. N.p., n.d. Web. 12 Jan. 2017.

[5]Tutorialspoint.com. "MVC Framework Introduction." *Www.tutorialspoint.com*. N.p., n.d. Web. 12 Jan. 2017.