|  |
| --- |
| aSIA pACIFIC COLLEGE |
| Artist Managers Placement Agency of the Philippines Inc. |
| Software Quality Assurance Plan |
|  |
| **Deborah P. Binag**  **Alyssa Jane B. Lino**  **Danica Faith M. Parian** |
| **March 2015** |

Table of Contents

[Software Project Management Plan 5](#_Toc415407182)

[Vision and Scope 5](#_Toc415407183)

[Feasibility and Risk Analysis 6](#_Toc415407184)

[Management Approach 7](#_Toc415407185)

[Roles and Responsibilities 8](#_Toc415407187)

[Technical Approach 9](#_Toc415407188)

[Software Environment 9](#_Toc415407189)

[Hardware Environment 9](#_Toc415407190)

[Network and Security 9](#_Toc415407191)

[Use Case Diagram 10](#_Toc415407192)

[Requirements Document 10](#_Toc415407193)

[The Logical Database Description (LDD) 10](#_Toc415407194)

[Logical Entity Relationship Diagram 10](#_Toc415407195)

[Entities and Description 11](#_Toc415407196)

[Software Requirements Document (SRD) 15](#_Toc415407197)

[User Requirements 15](#_Toc415407198)

[Security Requirements 15](#_Toc415407199)

[The Requirements Traceability Matrix 15](#_Toc415407200)

[Design Document 16](#_Toc415407201)

[The Physical Database Description 16](#_Toc415407202)

[Entity Relationship Diagram 16](#_Toc415407203)

[Components of the ERM 17](#_Toc415407204)

[The Software Design Document 18](#_Toc415407205)

[The Requirements Traceability Report 23](#_Toc415407206)

[Online Help 23](#_Toc415407207)

[Data Areas 23](#_Toc415407208)

[Optional Scenarios 24](#_Toc415407209)

[Data Dictionary 25](#_Toc415407210)

[Implementation Map 25](#_Toc415407211)

[Implementation Map 25](#_Toc415407212)

[The Requirements Traceability Matrix (RTM) 26](#_Toc415407213)

[Test Plan 27](#_Toc415407214)

[Overview/Introduction 27](#_Toc415407215)

[Assumptions 28](#_Toc415407216)

[Test Items 28](#_Toc415407217)

[Features to be tested 29](#_Toc415407218)

[Features not to be tested 30](#_Toc415407219)

[Approach 30](#_Toc415407220)

[Item Pass/Fail Criteria 31](#_Toc415407221)

[Test Deliverables 31](#_Toc415407222)

[Testing Tasks 32](#_Toc415407223)

[Environmental Needs 33](#_Toc415407224)

[HARDWARE ENVIRONMENT: 33](#_Toc415407225)

[SOFTWARE ENVIRONMENT: 33](#_Toc415407226)

[DATABASE SERVER: 33](#_Toc415407227)

[WEB SERVER APPLICATION: 34](#_Toc415407228)

[OPERATING SYSTEM: 34](#_Toc415407229)

[TOOLS: 34](#_Toc415407230)

[FRAMEWORK: 34](#_Toc415407231)

[Responsibilities 34](#_Toc415407232)

[Staffing and Training 35](#_Toc415407233)

[Schedule 35](#_Toc415407234)

[Resources 36](#_Toc415407235)

[Risks and Contingencies 37](#_Toc415407236)

[Approvals 37](#_Toc415407237)

[Deployment Plan 38](#_Toc415407238)

[Acceptance Plan 38](#_Toc415407239)

[Installation and Acceptance 39](#_Toc415407240)

[The Acceptance Configuration Plan 39](#_Toc415407241)

[An Updated Acceptance Plan 39](#_Toc415407242)

# Software Project Management Plan

## Vision and Scope

**Vision**

The project aims to provide a convenient process for the agency to monitor the number and schedule of their talents without the use of any paper and pens. This will also enable the agency to keep track of every necessary record without the worry of losing it.

**Scope**

Talents’ Record

- The system is able to store information about the talent and the client. The system should provide a database for the list of all talents and clients that will limit the other modules within the list inside the database.

Schedule of each talent

- The system should be able to monitor schedule of the talents. The system should be able to notify the agency if a talent has already an existing or future booking that will prevent them to have conflicts with other talents or clients. The system should be able to provide the details of each booked talent in a particular day, week or month including the client’s name or business.

Calendar Module

- The system is able to provide a calendar for the Administrator to view and check the events and screening schedule of each Talent.

Transaction Module

- The system should be able to compute the payment for each talent. A percentage of the rate given to the talent will go to the agency. Also, the system should be able to provide a database for each transaction for future use. In this module, it should only detect a talent that is already in the database to provide consistent data and to prevent mixing up of records.

## 

## Feasibility and Risk Analysis

#### Feasibility

* The system is able to provide a database for the agency that may be used in their future decision-making.
* The system is able to provide a user-friendly interface to the user.
* The system is able to provide a more convenient experience to the user such as the agency administrator and the talents.

#### 

#### Risk

* The agency might not meet the requirements such as the hardware, software and hosting in implementing the project.
* The administrator might input inconsistent data that will allow the system to mix up records.
* The agency might still want to use paper and pen because they are already used to it.
* The agency might not welcome necessary training to use the system.

|  |  |  |
| --- | --- | --- |
| * Feasibility and Risk Component | Overview | Solution |
| 1. Usability | The system should be able to provide a database that will hold all the necessary data that will be helpful to the business. Also, the system should be able to provide a more convenient experience for the user. | The group must list all possible features that will be helpful for the agency to make interaction fast and convenient. |
| 2. Environment and Design | The system should be able to provide a user-friendly interface to the user. The system should be easy to understand by the user. | The group must consult the preferred design of the agency and provide a help page for the whole system that can serve as their guide in using the system. |
| 3. System Requirements | The agency might not meet the requirements for the system to be deployed. | The group must produce a system that has features so that it will compensate the cost of all the requirements and return the income more than what they have spent. |
| 4. Training and Maintenance | The agency might not be open for training and maintenance as it may be an additional cost for them. | The group will provide a user manual and complete documentation of the system as a reference for them in their overall use. |
| 5. Development and Deployment Schedule | The group might not meet the deadline for each deliverable due to change requests or lack of resources. | The group will follow Agile Methodology and should be consistent in their SCRUM meetings in order to anticipate the tasks that should be accomplished in each iteration. |

## Management Approach

**Agile Methodology**

The group is required to follow the agile methodology.

Agile Software Development (ASD) is a methodology for the creative process that anticipates the need for flexibility and applies a level of pragmatism into the delivery of the finished product.

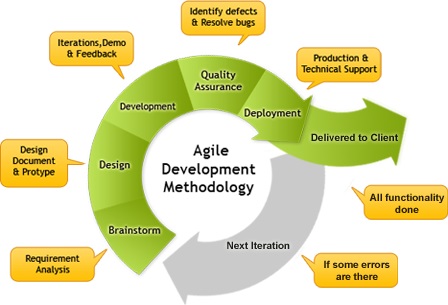


Figure 1 Agile Methodology, source: http://www.maestrointel1.com

# 

Figure 2 Meeting and Communication, source: http://www.bestoutcome.com/

## Roles and Responsibilities

|  |  |  |
| --- | --- | --- |
| Name | Role | Responsibilities |
| Deborah Binag | System Analyst/Project Developer |  Ensure the quality of the system   Ensure a consistent communication with the client   In charge of all the minutes of the meeting   Project documentation |
| Alyssa Jane Lino | Project Manager/Development Lead |  Ensure that deliverable is accomplished on time   Ensure that requirements are met   Should be able to adapt to changes in the middle of project development |
| Danica Faith Parian | System Analyst/Project Developer |  Ensure the quality of the system   Ensure a consistent communication with the client   Project documentation |

The group will also implement daily SCRUM meetings and weekly meetings for project updates and concerns. While for the client, we have agreed to hold meetings every after 2 weeks.   
  
Throughout the development process, the team should complete the following:

* Individual Weekly Accomplishment Report
* Weekly Project Status Report
* Final Project Documentation

## Technical Approach

### Software Environment

* **Programming Languages**

AMPAPhil Booking and Management System will be interfaced with a Yii 2.0 Framework and will use PHP as its programming language. The database will be stored on the server using MySQL.

* **Specific Software**

MYSQL is used for the database while XAMPP is used for the server. For the framework, the group will use Yii 2.0.

* **Operating System**

This system will operate in any Windows Operating System such as Windows XP, Windows Vista, Windows 7 and Windows 8.

### Hardware Environment

CPU: Intel® Core™ i5-2430M / i3/ i7

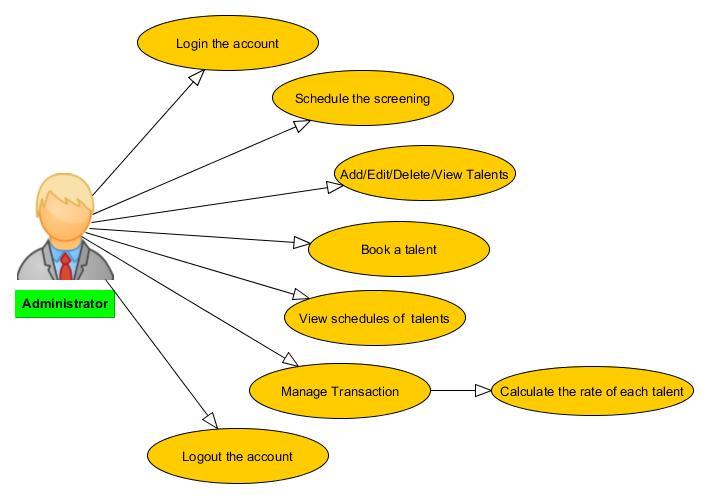
System Type: 32-bit Operating system/ 64-bit Operating system

Memory/Storage: 500GB

### Network and Security

The application should only be accessed by the administrator. The administrator is the one who will login to the page and manage the whole system. The only safety concern here is to keep all the files and transaction done in the system private and confidential.

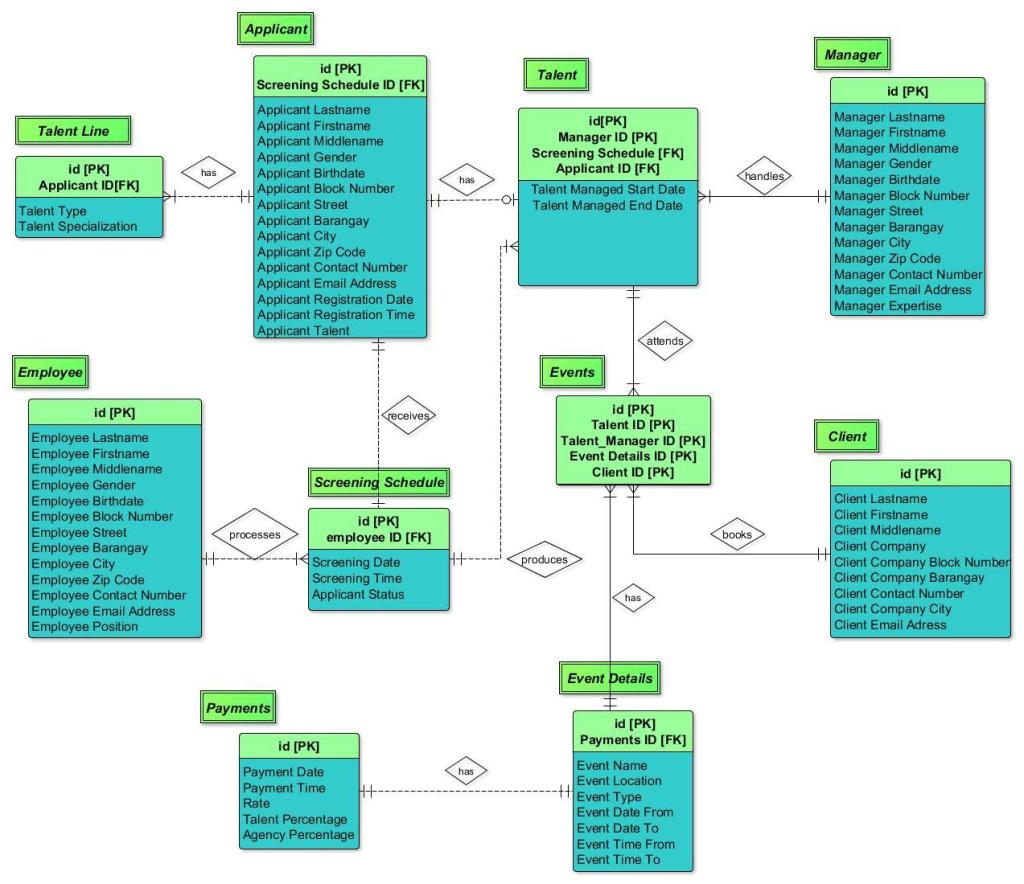
## Use Case Diagram



# Requirements Document

## The Logical Database Description (LDD)

### Logical Entity Relationship Diagram



### Entities and Description

1. Entity Name: Applicant

Description:

Applicant is the one who will apply to the agency to be one of their talents.

Relationship: (Weak)

Applicant has many talent lines .

Applicant has talent

Applicant receives screening schedule

Actor Interaction:

Applicant - can be a talent

2. Entity Name: Talent Line

Description: Talent Line consists of talent type and talent specialization. It can be a Singing or a Dancing.

Relationship: (Weak)

Applicant has many talent lines

Actor Interaction:

Applicant and Talent are the ones who will add or create Talent line

3. Entity Name: Screening Schedule

Description:

Screening schedule shows the schedule of each applicant to the screening via Calendar

Relationship: (Weak)

Employee processes many Screening schedule

Applicant receives a Screening schedule

Screening schedule produces many talents

Actor Interaction:

Administrator - the one who will schedule the screening

4. Entity Name: Employee

Description:

Employee is the one who are part of the organization and the management.

Relationship: (Weak)

Employee processes many screening schedule

Actor Interaction:

Administrator - the one who will add employees to the system

5. Entity Name: Events

Description:

Event is a planned occasion or activity (such as debut, wedding, etc.)

Relationship: (Strong)

Many events has event details

Client books many events

Talent attends many events

Actor Interaction:

Administrator - will create the events

6. Entity Name: Talent

Description:

Talent refers to a person having such ability.

Relationship: (Weak)

Applicant has talent

Many talents produces screening schedule

Relationship: (Strong)

Talent attends many events

Manager handles many talents

Actor Interaction:

Administrator - who can add, edit, delete and view talents

7. Entity Name: Manager

Description: Manager is the one who will manage the system aside from the administrator and the employees. This module is for future use in the system.

Relationship: (Strong)

Manager handles many talents

Actor Interaction:

Administrator – who can add, edit, delete and view manager’s module

8. Entity Name: Payment

Description:

Payment refers to the calculation of the rate of the talent and the percentage rate of the agency and the talent.

Relationship: (Weak)

Event details has payment

Actor Interaction:

Administrator - the one who will calculate the rate of each talent and the agency

9. Entity Name: Client

Description:

Client is the one who will book the event to the agency and uses the services.

Relationship: (Strong)

Client books many events

Actor Interaction:

Administrator - the one who will book the event to the system

10. Entity Name: Event Details

Description:

Event details composed of event name, location, type, status and more fields regarding the event.

Relationship: (Strong)

Event details has corresponding payment

Actor Interaction:

## Software Requirements Document (SRD)

### User Requirements

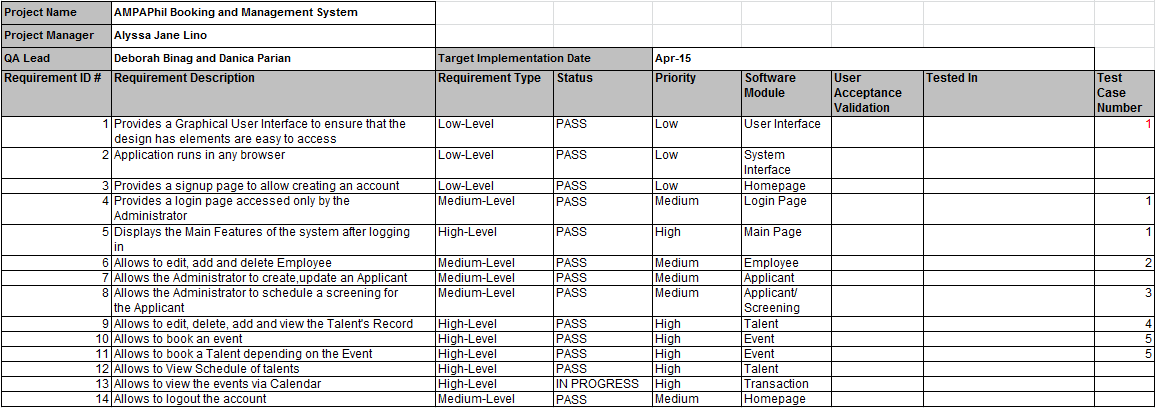
Scope of the Administrator:

* Talent's Record - can add, edit, delete and view talents
* Booking/Schedule - can schedule the screening of the applicants, book an event and talents
* Calendar - can view the schedule of events via Calendar
* Transaction - can compute and calculate the rate of each talent and the percentage rate of the talent and the agency

### Security Requirements

The uses of login account only by the administrator to ensure the safety and secured records of the system.

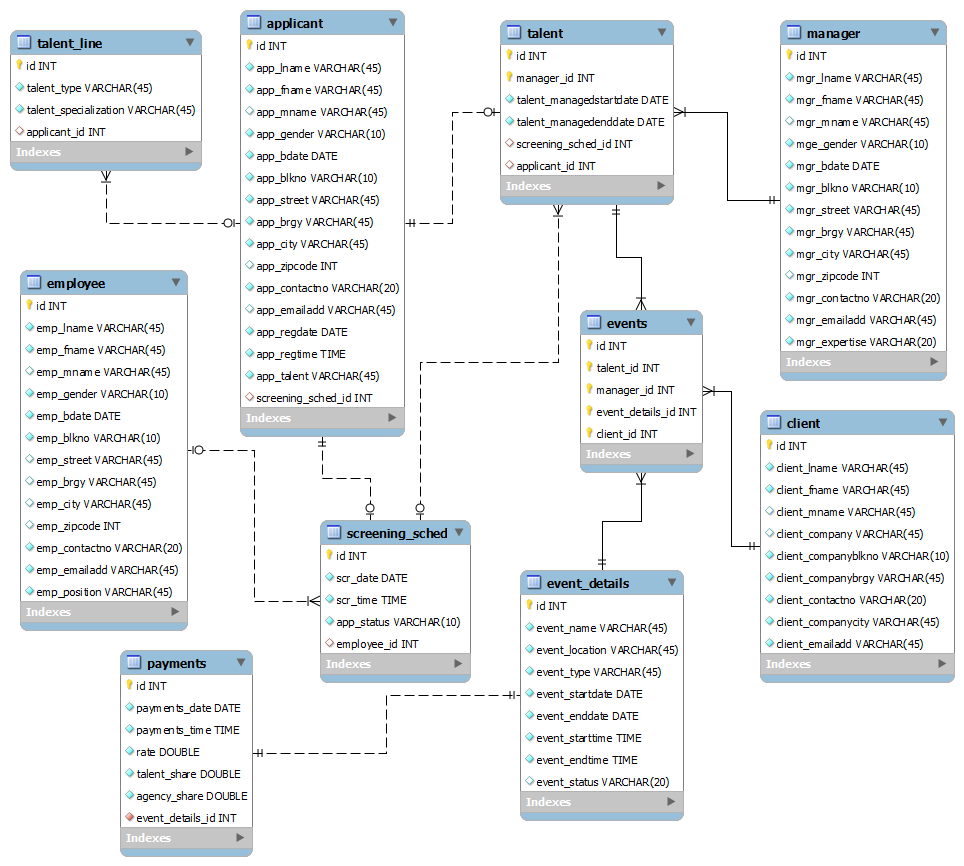
## The Requirements Traceability Matrix



# Design Document

## The Physical Database Description

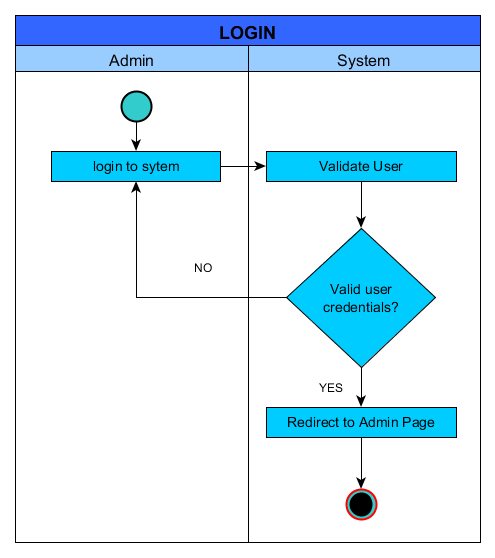
### Entity Relationship Diagram

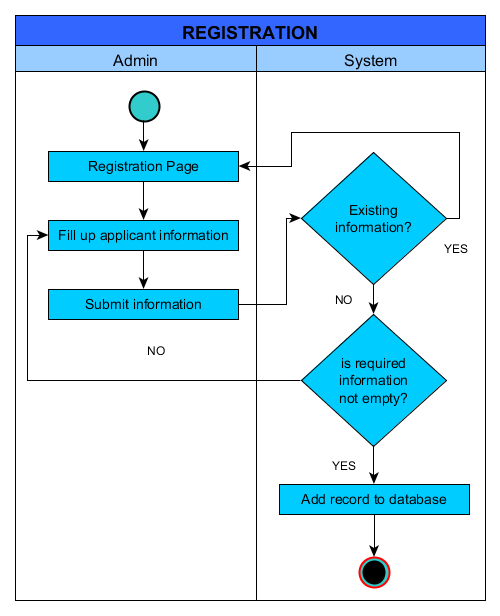


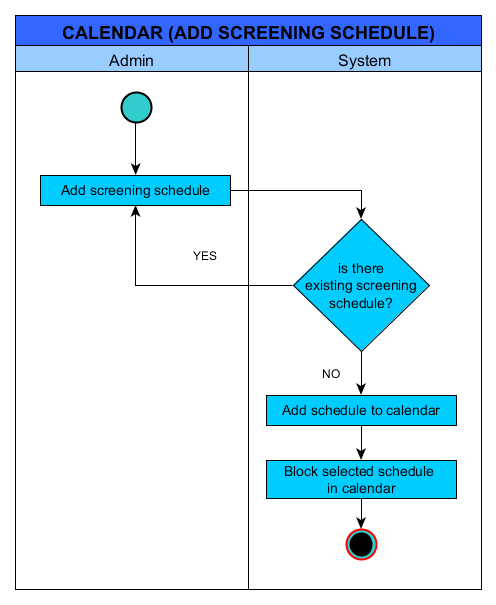
### 

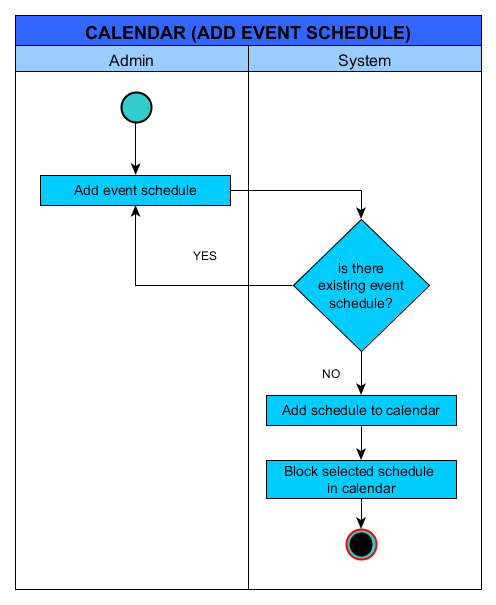
### Components of the ERM

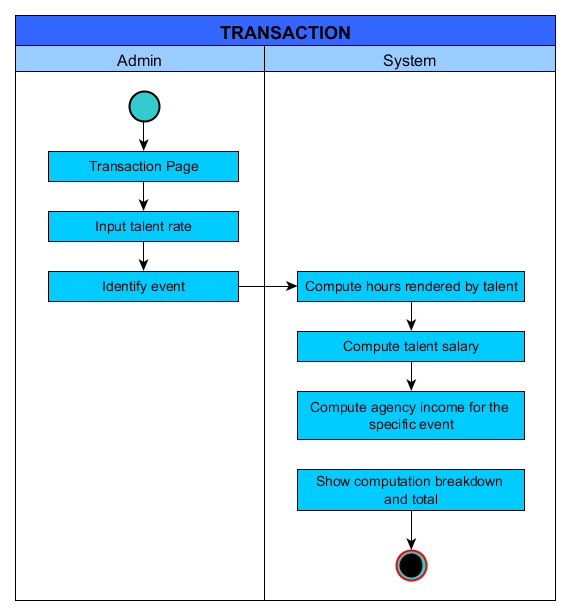
## The Software Design Document

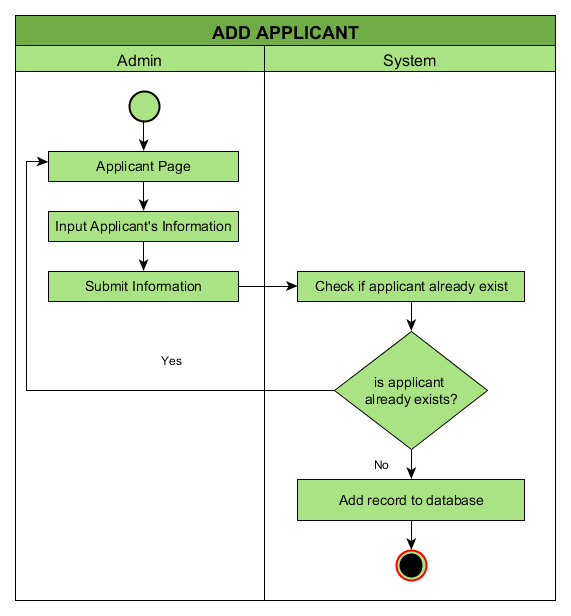


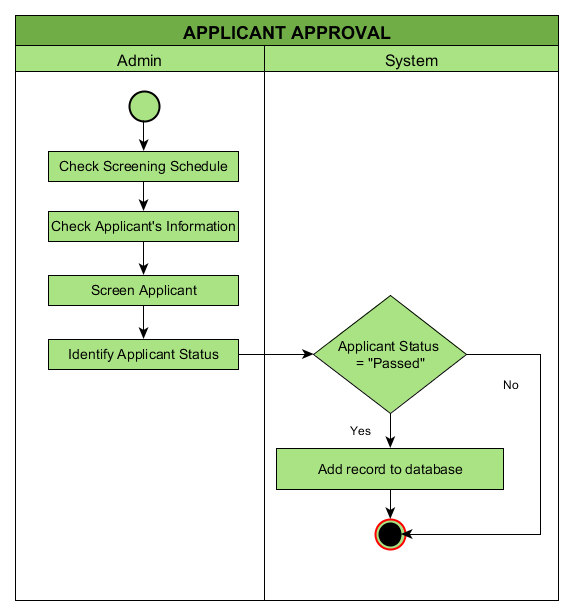












# Online Help

## Data Areas

|  |  |
| --- | --- |
| Features | Functionality |
| Employee module | * Create employee record * Update employee record * Delete employee record * View employee record |
| Applicants module | * Create applicant record * Update applicant record * Delete applicant record * View applicant record |
| Screening Schedule module | * Create screening schedule for applicant * Update screening schedule for applicant * Delete screening schedule for applicant * View screening schedule for applicant |
| Talents module | * Create talent record * Update talent record * Delete talent record * View talent record |
| Manager module | * Create manager record * Update manager record * Delete manager record * View manager record |
| Client module | * Create client record * Update client record * Delete client record * View client record |
| Events module | * Create event record * Update event record * Delete event record * View applicant record |
| Transaction module | * Create transaction record * Delete transaction record * View transaction record * Compute for talent and agency’s share based on rate |

## Optional Scenarios

1. How do I create an applicant record?

Click on the Applicants module on the home page. To add an applicant, click on the “Create Applicant” green button located at the left side of the page. Accomplish all the required fields and click on “Create” green button at the bottom.

Note: This is also same with the other records. Just click their appropriate module and navigate from there.

2. What is the “Add Talent Line” green button for?

This is to specify the talent of a specific applicant for easy reference. After creating an applicant, an administrator may specify the talent of that applicant by identifying the applicant’s name, his or her talent type and talent specialization.

3. How do I update an applicant record?

An administrator should search any applicants by any of the available search boxes. After displaying all the possible results, click on the pencil icon that says “Update” on the right side of the page. Alter any of the details and click on the “Update” blue button at the bottom. Verify if the changes has been made from the view page of the module.

Note: This is also same with the other records. Just click their appropriate module and follow again the steps. Update is not available for the Transaction module.

4. How do I delete an applicant record?

An administrator should search any applicants by any of the available search boxes. After displaying all the possible results, click on the trashcan icon that says “Delete” on the right side of the page. A message box will appear that says, “The page at localhost says: Are you sure you want to delete this item?” Just click on OK and verify if the item is already deleted from the table view of the module.

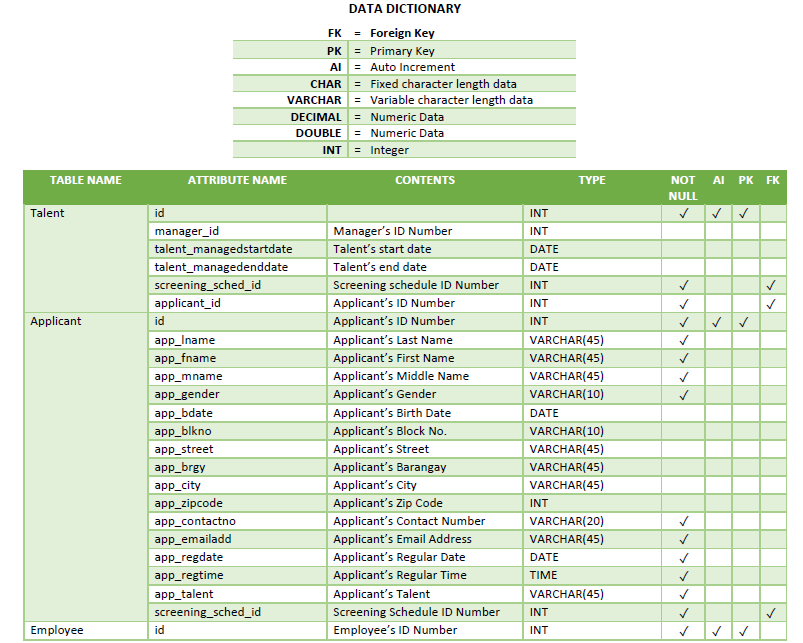
Note: This is also same with the other records. Just click their appropriate module and follow again the steps.

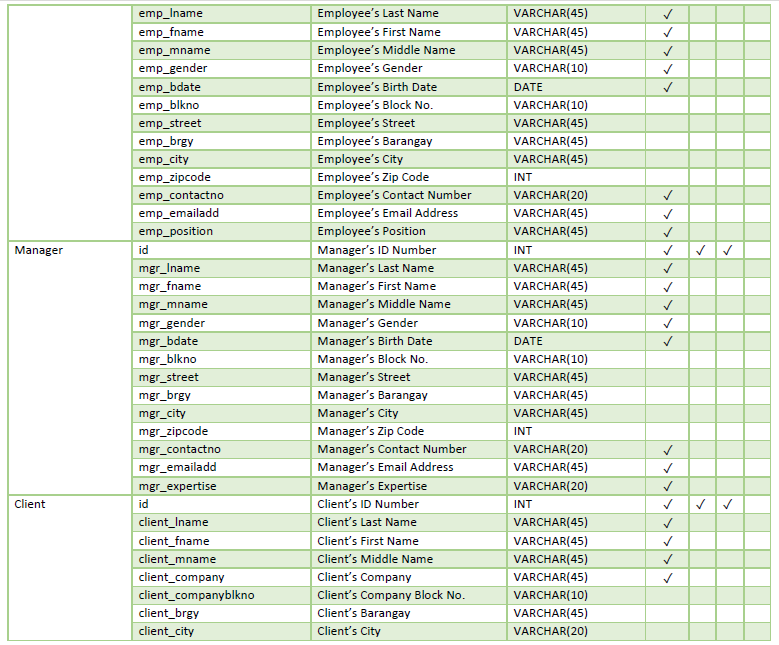
5. How do I view an applicant record?

An administrator should search any applicants by any of the available search boxes. After displaying all the possible results, click on the eye icon that says “View” on the right side of the page.

Note: This is also same with the other records. Just click their appropriate module and follow again the steps.

## Data Dictionary





# Implementation Map

## Implementation Map

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Task Category | | Description | | Start Date | | End Date | Duration(days) |
| Project Planning | | This involves the software project management plan | | February 2 | | February 11 | 9 days |
| Project Design  (Physical, Logical and Database) | | This involves the requirements document and design document | | February 9 | | February 12 | 4 days |
| Development | | This involves the implementation phase | | February 16 | | March 6 | 19 days |
| Quality Assurance | | This involves the test plan, test cases and quality or test reports | | March 5 | | March 11 | 7 days |
| Deployment | This is to place the solution into a production environment – Final Iteration/Iteration 3 | April 6 | April 11 | | 6 days | | |
| Closing | Final presentation and documentation | April 8 | April 11 | | 4 days | | |

## The Requirements Traceability Matrix (RTM)

# Test Plan

## Overview/Introduction

ASIARABE ENTERPRISES, INC. which was incorporated on May 16, 1987 was shortly changed to Artist Managers Placement Agency of the Philippines and was incorporated on July 2, 1987. AMPAPhil Inc. is a Non–Construction Philippine-based human resource recruitment and deployment agency, which takes pride in providing world-class Filipino professional and skilled workers and performing artist. We are committed to contribute growth and success by providing our client with candidates that match their specification and qualification through our relentless effort to take extra mile in imparting an outstanding service giving us an edge in the overseas employment industry. The estimated number of overseas contract workers deployed as of December 2012 is 16, 105. AMPAPhil Inc. specializes in recruitment, deployment and post deployment services such as:

* Mobilization, sourcing and pooling of applicants based on clients’ specifications;
* Preliminary screening and testing prior to final selection by the principal and/or employer;
* Document processing in POEA and at the Embassy of the country of destination, when necessary; and
* Orientation, briefing meetings and Pre-Departure Orientation Seminar

Artist Managers Placement Agency of the Philippines Inc. is a member of Overseas Placement Association of the Philippines (OPAP) and also a member of Philippines Association Services Exporters INC. (PASEI).

## Assumptions

The following statements are the explicit and implicit assumptions for AMPAPhil:

1. The data input in to the system is manual and can only be done by the administrator or employee of the agency.
2. The system is able to store data regarding the applicants that can become talents in the future, employees, managers, clients and even the transactions or payments made.
3. The system is able to compute for the share of payment from the client given to the talent and to the agency.
4. The system is able to help the administrator to monitor the schedule of events for their talents.
5. The system is able to provide a user-friendly interface that can be easily understood by any of the agency’s employees.
6. The system is able to provide data that can be very useful in decision-making.
7. The system is able to provide a very convenient and fast experience for the employees, talents, managers and clients in every transaction.

## Test Items

The items to be tested are the major functionalities and the requirements and performance of the system.

Items needed to be test:

* Administrator Account

This involves the login and logout of the account. This is also a part of the security issue that needs to be test first before the functionality of the system.

* User Interface – Design of the system

This is one of the criteria that focus on the design and to anticipate what users might need to do and ensures that the interface has elements are easy to access.

* Performance

This involves the accuracy and the amount of time you will use the system. The performance of the system is dependent on how well it works together as a whole.

* Functionality

A major portion of the test items that consists of the Event and Applicant Screening schedule, and the Calendar module to view the events and schedule of each Talent in the Agency. Also, the requirements specification, project plan & schedule and use case of the system should also be considered in testing this project.

## Features to be tested

This section focuses on the functional aspects of the system and a list of what to be tested fromthe Users viewpoint and describe the level of importance.

## Features not to be tested

In the booking and management system, there are several features and specifications that are not to be tested. These includes out of scope features, low risk features and future functionality.

• Managers’ Module – this portion is for future use of the system and still has no function yet.

• Transaction Phase – this section is very confidential and should be used by the Administrator. Also, this is for the future use of the management and it is not required to finish said by the client.

• About and Contact Page – this has a low risk factor in the system that is not too important to test.

## Approach

The test approach for this project is manual testing. The tester should be able to test the following:

1. Black-box testing – The tester should ensure that the requirements and functionalities are met.
2. White-box testing – The tester should ensure that the internal logic of codes are correct and necessary for a specific function.
3. Unit testing – The tester should be able to test each modules that the system has.
4. Regression testing – The tester should be able to test the whole system according to the specifications available.
5. Performance testing – The tester should be able to test the performance of the whole system. The tester should ensure that the system can process data and respond to the user fast.
6. Usability testing – The tester should be able to evaluate if the system would be useful for the agency. The tester should be able to identify if the system can increase productivity of the agency and will not cost much for the owner of the agency.

## Item Pass/Fail Criteria

|  |  |  |
| --- | --- | --- |
|  | PASS | FAIL |
| Component Testing |  |  |
| Test Case – Completeness |  |  |
| Performance - Speed and Accuracy |  |  |
| User Interface – Good design and User Interaction |  |  |
| Security – Account access and permission |  |  |
| Regression Testing – Enhancements and Configuration |  |  |
| Defects Encountered – Fixed |  |  |
| Software Specifications – Standards Conformity |  |  |
| Compatibility Testing – Performance on different screen sizes, OS version and Configuration |  |  |

## Test Deliverables

|  |  |  |
| --- | --- | --- |
| Name | Description | Responsibility of |
| Test Plan Document | A documentation that will be used by the testers so that they will have a background on the scope of their tasks. | * Project Manager * Business Analysts |
| Test Cases | A documentation that shows a step by step guide of the different modules in the system. | * Business Analysts |
| Test design specifications | A documentation that includes the overall scope of the project. | * Project Manager * Business Analysts |
| Tools and their outputs | A list of all the tools that were used in the development of the project. | * Developers |
| Error logs and execution logs | A documentation that includes the different errors encountered and the record of times that the system is executed. | * QA Team |
| Problem reports and corrective actions | A documentation of all the recommendations that will help solve and correct different defects or error encountered. | * QA Team * Business Analysts * Developers |

## Testing Tasks

The following tasks are necessary for preparing and performing the system tests:

• Designing the system tests

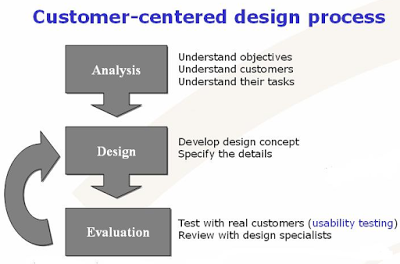
* Providing all the object oriented and user interface requirements. From planning to design phase, all factors including the objectives and goals of the system should be identify.
* 

Figure 1: Testing Design Process

Source: narayan-singh.blogspot.com/

• Ensuring that all environmental needs are satisfied for the system tests.

• Completing the integration tests.

* The execution test should be done here. The implementation processes and procedures should be able to provide and the requirements specification should match with the output of each process.

• Writing a test report and, if necessary, problem reports

* Test report is being produce at the last part of the plan. This contains scores for all areas of testing. This should be detailed and must have a recommendations and observation regarding testing. If problems occur, a problem reports should be produce. This identifies all the defects and problem encountered while testing. And if a solution is not available, the details in a problem report can help you to determine.

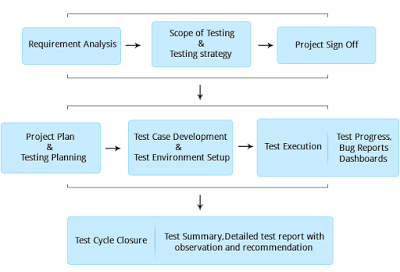


Figure 2: Testing Flowchart

Source: narayan-singh.blogspot.com/

## Environmental Needs

### **HARDWARE ENVIRONMENT**:

CPU: Intel® Core™ i3-2430M / i5/ i7

System Type: 32-bit Operating system / 64-bit Operating system

Memory/Storage: 500GB

### **SOFTWARE ENVIRONMENT**:

**Programming Languages:**

- PHP 5.4.16

- MySQL

### **DATABASE SERVER**:

- PHPMyAdmin

### **WEB SERVER APPLICATION**:

- XAMPP 1.8.2

### **OPERATING SYSTEM**:

- This system will run in any Windows Operating System such as Windows XP, Windows Vista, Windows 7 and Windows 8

### TOOLS:

**For development:**

- TortoiseSVN

- Browser

- Notepad++ or Sublime Text 2

**For testing:**

- Browser

### FRAMEWORK:

- Yii2 with advanced template

## Responsibilities

|  |  |
| --- | --- |
| INDIVIDUAL | ROLE / RESPONSIBILITY |
| * Project Manager * Business Analyst * QA Manager | * Test reviews of business requirements |
| * Project Manager * Business Analysts | * Managing the test process or each test phase, such as User Acceptance Testing (UAT) |
| * Project Manager * QA Manager * Developers | * The completion of the test risk assessment |
| * QA Lead * QA Analyst | * Designing the tests & test data |
| * QA Analyst | * Executing the tests |
| * QA Lead * Developers | * Management & resolution of incidents reported |
| * QA Lead | * Reporting the status of issues and tests |
| * QA Manager | * Configuration and management of the testing repository and tools |

## Staffing and Training

|  |  |  |
| --- | --- | --- |
| Deliverable | For | Training Needs |
| Test Plan | Project Manager; QA Lead; Test Team | Responsible for training the following test resources |
| Traceability Matrix | Project Manager; QA Lead | Must be trained on the process being used for the project |
| Test Results | Project Manager | Must be trained on the defect/issues |
| Test Status report | QA Manager, QA Lead |  |
| Implementation and Development | Developer/Component Testers | Must know PHP and Yii for the framework; Must be skilled in unit testing and integration testing |
| Hardware and Technical Requirements | Technical Support Team | Must be knowledgeable on the technical processes and requirements |
| Test Meeting Reports | SCRUM Master | Must be skilled in producing reports and updates on the system |
| Metrics | All team members | Must be experienced in system testing and trained on the test specification requirements |

## Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task Name | Start | Finish | Effort | Comments |
| Test Planning | February 2 | February 5 | 4 days | Startup Plan |
| Review Requirements documents | February 6 | February 9 | 4 days | Requirements documentation |
| Create initial test estimates | February 9 | February 11 | 3 days | Estimation of Testing process schedules |
| Physical and Logical Design | February 9 | February 12 | 4 days | Design/Object Oriented Plan Documentation  First Static Review |
| First deploy to QA test environment (Iteration 0) | February 16 | February 27 | 12 days | Iteration 0 (GUI) |
| Functional testing – Iteration 1 | February 28 | March 6 | 7 days | Iteration 1  (Implementation Phase) |
| Iteration 2 deploy to QA test environment | March 7 | March 12 | 6 days | 2nd round Static Review |
| Functional testing – Iteration 2 | March 23 | March 29 | 7 days | Iteration 2 |
| System testing | March 30 | April 1 | 3 days | Testing after Iteration 2 |
| Regression testing | April 1 | April 3 | 3 days | Testing the system after all the changes |
| UAT | April 3 | April 5 | 3 days | User acceptance Testing – Final testing before deployment |
| Resolution of final defects and final build testing | April 3 | April 6 | 4 days | Final Testing |
| Deploy to Staging environment | April 6 | April 8 | 3 days | Deployment to Staging environment |
| Performance testing | April 6 | April 11 | 6 days | Last testing after UAT |
| Release to Production | April 11 | April 15 | 5 days | Last deployment environment |

## Resources

This section presents a list of documents that will use for the reference and guide in accomplishing the deliverables of the system.

1. Test plan
2. Test cases
3. Project specification
4. Defects log
5. UAT request

## Risks and Contingencies

This will serve as a plan and preparation to determine the best contingencies in the event that one of the planning risks occurs. This is important because the scope and nature of a project almost always change as the project progresses.

The following includes all the risks:

* The agency might not meet the requirements such as the hardware, software and hosting in implementing the project.
* The administrator might input inconsistent data that will allow the system to mix up records.
* The agency might still want to use paper and pen because they are already used to it.
* The agency might not welcome necessary training to use the system.

The following actions will be taken:

* Resources will be added to the test team. There may be some optimization of resources.
* Tester should be able to test and put correct data for consistency.
* System should work properly and it should be user-friendly.
* Training resources should be absorbed by the Client and it should be very helpful to them.

## Approvals

There will be 4 Iterations in implementing the project. This includes first and second static review, dynamic testing and final dynamic testing. With these, there will be a Quality Assurance Group who will test and verify the Software Quality Assurance Plan each iteration.

The following are the group of people and advisers who are included in the project:

|  |  |  |
| --- | --- | --- |
| Project Advisers | Signature | Date |
| Ms. Rhea-Luz Valbuena |  |  |
| Mr. Ernesto Boydon |  |  |
|  |  |  |
| Project Panelists |  |  |
| Ms. Donna Lalusin |  |  |
| Mr. Alfredo Calimbo |  |  |
|  |  |  |
| SQA Group |  |  |
| Julian Acuna |  |  |
| Paolo Sagun |  |  |
| Angelo Charles Sia |  |  |
|  |  |  |
| Subject Professor |  |  |
| Mr. Allan Cotecson |  |  |

# Deployment Plan

# Acceptance Plan

The following items describe the factors to gain the customers’ acceptance for each deliverable. The acceptance plan has its review methods and reviewers to identify the acceptance test methods and verify that the certain deliverable met the requirements the acceptance standards.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Completion |  |  | Acceptance |  |  |
| Milestone | **Deliverable** | **Date** | **Review Method** | **Reviewers** | **Date** |
| AMPAPhil Booking and Management System | Test Design Specifications | February 2015 | * Static Testing * Specification-based Testing * Visual Testing | * Project Managers and Analysts * Quality Assurance Group/Testers | February 2015 |
|  | Test Plan Document | March 2015 | * Static Testing * Wiki Page Verification * Data accuracy review | * Quality Assurance Group/ Testers * Project Manager and Analysts * Project Advisers | March 2015 |
|  | Test Cases | March 2015 | * Regression Testing * Static Testing * User acceptance Testing | * Quality Assurance Group/Testers | March 2015 |
|  | Tools and their Outputs | March 2015 | * System Testing * Installation Testing | * Technical Group * Developer/s | March 2015 |
|  | Error Logs and Execution Logs | March – April 2015 | * Regression Testing * Unit Testing * Software Performance Testing | * Quality Assurance Group | March – April 2015 |
|  | Problem Reports and Corrective actions | April 2015 | * Unit Testing * Acceptance Testing | * Quality Assurance Group * Project Manager and Analysts | April 2015 |
|  | Frontend version of the system | April 2015 | * Dynamic Testing | * Quality Assurance Group | April 2015 |
|  | Backend version of the system | April 2015 | * Dynamic Testing | * Quality Assurance Group | April 2015 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Installation and Acceptance

## The Acceptance Configuration Plan

## An Updated Acceptance Plan