

[Email address]

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

[Document title]

[Document subtitle]

Table of Contents

[1. Title 3](#_gjdgxs)

[2. Test Plan Description: 3](#_30j0zll)

[2.1 Document purpose 3](#_1fob9te)

[2.2 Types of Testing 3](#_3znysh7)

[∙ Unit Testing 3](#_2et92p0)

[∙ Integration Testing 3](#_tyjcwt)

[∙ System Testing 4](#_3dy6vkm)

[2.3 Benefits 4](#_1t3h5sf)

[3. Operation test plan: 4](#_4d34og8)

[3.1 Operation Referencing 4](#_2s8eyo1)

[3.2 Test Case Operation: 5](#_17dp8vu)

[4 Test Plan Validation: 9](#_3rdcrjn)

[4.1 Discussion- plan completeness 9](#_26in1rg)

[4.2 Verify- level of confidence: 10](#_lnxbz9)

[5. References 10](#_35nkun2)

# Title

The designed HRIS system is built for providing easy access for the students and staff members for using the ICT database. The system performs different operations like displaying the different Units, Staff, Classes, and Consultations of the ICT and resolving student queries at a faster rate. It also shows the different searching ways for viewing the different staff and consultations for a class and staff member by giving unit and class values.

The team formed for designing the HRIS system and developing the test cases consist of the following members

1)

2)

3)

## Test Plan Description:

* 1. Document purpose

The proposed set of test cases will help in determining whether the system works correctly and satisfies all the requirements. It helps in finding the problems in the given requirement and efficiently developing the system. It analyses the given problem for completeness and reviews the overall design of the system.

* 1. Types of Testing

Some of the testing that are being used are as follows:

* Unit Testing

In this type of testing test cases are written to check the behavior of different modules. In the HRIS system each unit namely the Staff, Class, Unit, and Consultation are a different unit and each of them is tested separately to ensure each unit is performing its function. If any error is detected, then it is easy to resolve as it involves a single module. (Ramesh, 2007)

* Integration Testing

In this type of testing the test cases are derived from the architecture and design specifications. Each module is incrementally integrated to check the module integration and interaction. In the HRIS system, the search and view function is combined to find all the classes for a staff member. Two modules are combined and tested in the form of a group. If any error is detected, it will not be easy to find the unit in which the error occurred. (Patton, n.d.)

### System Testing

In this type of testing the entire system is tested to check whether the developed system is working according to the specific requirements. It focuses on the entire system’s functionality and validates the whole system’s functioning. In the HRIS system, the whole system is checked to find whether it is functioning as per the requirements. If any error occurs in this testing, then it will be very difficult to resolve as it involves the entire system, and debugging becomes hard. (Basu, 2015)

## Benefits

Testing finds the defects in the system and helps us in ensuring whether the system is developed as per the requirements. It increases the product quality and improves the system’s functionality. It ensures no part of the program is left out without testing and improves the security of the system. It provides better system functioning and gives a flawless experience to the user. The system becomes easy for managing and reduces the risk involved.

The benefits of some of the different types of testing are as follows:

* Unit Testing

In unit testing, we run a small part of the code to check whether we get the required outcome. It works by isolating a part of the code and is an integral part of testing. By doing so we can detect the error at an early stage and easily debug the code. If any changes are to be made then they can be made easily. It also helps in refactoring the code and in writing them effectively.

* Integration Testing

Integration testing is performed after ensuring that all the units are working correctly. In this, we combine the different units or components to check whether they communicate in the desired way. It helps us in executing the user-based scenarios and uncovers the new bugs present in the system thereby increasing the effectiveness of the system.

### System Testing

In system testing the entire system is tested from a user point of view and tests the working of the entire application. This is performed at the last stage after successful execution of all different types of testing and is done by integrating all the components. It specifies the behavior of the system and helps in finding any undetected bugs.

# Operation test plan:

## Operation Referencing

1. **Test cases for showing all the units:** The data flow diagram and the sequence diagram in the design document section are referred to deriving the required test cases for showing all the units.
2. **Test cases for showing all the staff:** The sequence diagram and the use case diagram in the design document section are referred for deriving the required test cases for showing all the staff.
3. **Test cases for showing all the classes:** The sequence diagram and the use case diagram in the design document section are referred for deriving the required test cases for showing all the classes.
4. **Test cases for showing all the consultations:** The sequence diagram and the use case diagram in the design document section are referred for deriving the required test cases for showing all the consultations.
5. **Test cases for searching and viewing all the classes for a staff member:** The sequence diagram and the use case diagram in the design document section are referred for deriving the required test cases for searching and viewing all the classes for a staff member. First, the staff member is searched and then all the classes for the given staff members are displayed.
6. **Test cases for searching and viewing all the consultations for a staff member:** The sequence diagram and the use case diagram in the design document section are referred for deriving the required test cases for searching and viewing all the consultations for a staff member. First, the staff member is searched and then all the consultations for the given staff member are displayed.
7. **Test cases for searching and viewing all the Consultations for a Class:** The sequence diagram and the data flow diagram in the design document section are referred for deriving the required test cases for searching and viewing all the consultations for a Class. First, the Class is searched and then all the consultations for the given Class are displayed.
8. **Test cases for searching and viewing all the classes for a Unit:** The sequence diagram and the data flow diagram in the design document section are referred for deriving the required test cases for searching and viewing all the classes for a Unit. First, the Unit is searched and then all the classes for the given Unit are displayed.

## Test Case Operation:

1. **Test cases for showing all the units:**

|  |  |
| --- | --- |
|  | |
| **Test Case ID:** 1  **Test Type:** Unit Testing | **Test Designed by:** |
| **Test Priority (Low/Medium/High):** Medium | **Test Designed date:** |
| **Use Case Name:** use\_case\_1 | **Test Executed by:** |
| **Test Title:** Display all the units | **Test Execution date:** |
| **Description:** The below test case is used check whether all the units are displayed in the required order. |  |
|  |  |
|  |  |
| **Pre-conditions: The user should click on the show all unit button** | |
| **Dependencies: Depends only on the Unit module** | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Step** | **Test steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Access HRIS System | <Valid User> | Displays the unit code, unit name, and coordinator for all the existing units |  |  | All the available units are displayed |
| Click on the “Unit Search” Button | <Units exist in the database> |
| 2 | Access HRIS System | <Valid User> | A message “No units exist” is shown |  |  | An error message is displayed |
| Click on the “Unit Search” Button | <Units do not exist in the database> |

**Post-conditions:** If the user is a valid user and units are existing in the database then the unit code, unit name, and coordinator for all the units must be Displayed, or else the error message should be displayed

1. **Test cases for showing all the staff:**

|  |  |
| --- | --- |
| **Test Case ID:** 2  **Test Type:** Unit Testing | **Test Designed by:** |
| **Test Priority (Low/Medium/High):** Medium | **Test Designed date:** |
| **Use Case Name:** use\_case\_2 | **Test Executed by:** |
| **Test Title:** Display all the staff | **Test Execution date:** |
| **Description:** The below test case is used check whether all the staff is displayed in the required order. |  |
|  |  |
|  |  |
| **Pre-conditions: The user should click on the show all staff button** | |
| **Dependencies: Depends only on the Staff module** | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Step** | **Test steps** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** | **Notes** |
| 1 | Access HRIS System | <Valid User> | Displays Staff ID, Given Name, Family Name, Title, Campus, Phone, Room, Email, and photo for all the existing staff |  |  | All the available Staff are displayed |
| Click on the “Staff Search” Button | <Staffs exist in the database> |
| 2 | Access HRIS System | <Valid User> | A message “No staff exist” is shown |  |  | An error message is displayed |
| Click on the “Staff Search” Button | <Staff does not exist in database> |

**Post-conditions:** If the user is a valid user and units are existing in the database then the Staff ID, Given Name, Family Name, Title, Campus, Phone, Room, Email, and photo for all the units must be Displayed, or else the error message should be displayed

1. **Test cases for showing all the classes:**

|  |  |
| --- | --- |
| **Test Case ID:** 3  **Test Type:** Unit Testing | **Test Designed by:** |
| **Test Priority (Low/Medium/High):** Medium | **Test Designed date:** |
| **Use Case Name:** use\_case\_3 | **Test Executed by:** |
| **Test Title:** Display all the classes | **Test Execution date:** |
| **Description:** The below test case is used check whether all the classes are displayed in the required order. |  |
|  |  |
|  |  |
| **Pre-conditions: The user should click on the show all classes button** | |
| **Dependencies: Depends only on the Class module** | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Step | Test steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
| 1 | Access HRIS System | <Valid User> | Displays Unit code, Campus, Day, Start Time, End Time, Type, Room, and Staff for all the existing classes |  |  | All the available Classes are displayed |
| Click on the “Class Search” Button | <Class exists in the database> |
| 2 | Access HRIS System | <Valid User> | A message “No Class exists” is shown |  |  | An error message is displayed |
| Click on the “Class Search” Button | <Class does not exist in database> |

**Post-conditions:** If the user is a valid user and units are existing in the database then the Unit code, Campus, Day, Start Time, End Time, Type, Room, and Staff for all the units must be Displayed, or else the error message should be displayed

1. **Test cases for showing all the consultations:**

|  |  |
| --- | --- |
| **Test Case ID:** 4  **Test Type:** Unit Testing | **Test Designed by:** |
| **Test Priority (Low/Medium/High):** Medium | **Test Designed date:** |
| **Use Case Name:** use\_case\_4 | **Test Executed by:** |
| **Test Title:** Display all the consultations | **Test Execution date:** |
| **Description:** The below test case is used check whether all the consultations are displayed in the required order. |  |
|  |  |
|  |  |
| **Pre-conditions: The user should click on show all consultation** | |
| **Dependencies: Depends only on the consultation module** | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Step | Test steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
| 1 | Access HRIS System | <Valid User> | Displays Staff ID, Day, Start Time, and End Time for all the existing consultations |  |  | All the available consultations are displayed |
| Click on the “Consultation Search” Button | < Consultations exist in the database> |
| 2 | Access HRIS System | <Valid User> | A message “No Consultations exist” is shown |  |  | The error message is displayed |
| Click on the “Consultation Search” Button | < Consultations do not exist in the database> |

**Post-conditions:** If the user is a valid user and units are existing in the database then the Staff ID, Day, Start Time, and End Time for all the units must be Displayed, or else the error message should be displayed

1. **Test cases for searching and viewing all the classes for a staff member:**

|  |  |
| --- | --- |
| **Test Case ID:** 5  **Test Type:** Integration Testing | **Test Designed by:** |
| **Test Priority (Low/Medium/High):** High | **Test Designed date:** |
| **Use Case Name:** use\_case\_5 | **Test Executed by:** |
| **Test Title:** Search and view all classes for given staff value | **Test Execution date:** |
| **Description:** The below test case is used to find and display all the classes for the provided staff value |  |
|  |  |
|  |  |
| **Pre-conditions:** The user should provide the staff ID or staff Name whose classes need to be displayed | |
| **Dependencies:** Depends on the staff module for searching for given staff and the class module for viewing all the classes for the given staff | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Step | Test steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
| 1 | Access HRIS System | <Valid User> | A message “The Staff detail is Invalid” is shown |  |  | The error message is displayed |
| Click on Staff Search Button | <Invalid Staff ID or Staff Name> |
| 2 | Access HRIS System | <Valid User> | A message “No Class exists” is shown |  |  | An error message is displayed |
| Click on Staff Search Button | <Provide valid Staff ID or Staff Name> |
| Click on View Classes Button | < Class does not exist in database > |
| 3 | Access HRIS System | <Valid User> | Displays all the classes for the given staff value |  |  | All the classes handled by the given Staff ID/Staff Name are displayed |
| Click on Staff Search Button | <Provide valid Staff ID or Staff Name> |
| Click on View Classes Button | < Class exists in database > |

|  |
| --- |
| **Post-conditions:** If the user is valid but provided staff value is Invalid then the “The Staff detail is Invalid” error message should be displayed.  If the user is valid and provided staff value is also Valid but there is no class for the given staff then the “No Class exists” message should be displayed.  If the user is valid, provided staff value is Valid and there are classes for the given staff then all the classes handled by the given Staff ID/Staff Name should be displayed. |

1. **Test cases for searching and viewing all the consultations for a staff member:**

|  |  |
| --- | --- |
|  | |
| **Test Case ID:** 6  **Test Type:** Integration Testing | **Test Designed by:** |
| **Test Priority (Low/Medium/High):** High | **Test Designed date:** |
| **Use Case Name:** use\_case\_6 | **Test Executed by:** |
| **Test Title:** Search and view all consultations for given staff value | **Test Execution date:** |
| **Description:** The below test case is used to find and display all the consultations for the provided staff value |  |
|  |  |
|  |  |
| **Pre-conditions:** The user should provide the staff ID or staff Name whose classes need to be displayed | |
| **Dependencies:** Depends on the staff module for searching for given staff and the consultation module for viewing all the consultations for the given staff | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Step | Test steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
| 1 | Access HRIS System | <Valid User> | A message “The Staff detail is Invalid” is shown |  |  | An error message is displayed |
| Click on Staff Search Button | <Invalid Staff ID or Staff Name> |
| 2 | Access HRIS System | <Valid User> | A message “No Staff exists” is shown |  |  | An error message is displayed |
| Click on Staff Search Button | <Provide valid Staff ID or Staff Name> |
| Click on the View staff Button | < Staff does not exist in database > |
| 3 | Access HRIS System | <Valid User> | Displays all the Consultations for the given staff value |  |  | All the Consultations for the given Staff ID/Staff Name are displayed |
| Click on Staff Search Button | <Provide valid Staff ID or Staff Name> |
| Click on View Consultations Button | < Class exists in database > |

|  |
| --- |
| **Post-conditions:** If the user is valid but provided staff value is Invalid then the “The Staff detail is Invalid” error message should be displayed.  If the user is valid and provided staff value is also Valid but there are no consultations for given staff then the “No Consultations exists” message should be displayed.  If the user is valid, provided staff value is Valid and there are consultations for the given staff then all the Consultations for the given Staff ID/Staff Name should be displayed. |

1. **Test cases for searching and viewing all the Consultations for a Class:**

|  |  |
| --- | --- |
| **Test Case ID:** 7  **Test Type:** Integration Testing | **Test Designed by:** |
| **Test Priority (Low/Medium/High):** High | **Test Designed date:** |
| **Use Case Name:** use\_case\_7 | **Test Executed by:** |
| **Test Title:** Search and view all consultations for given class value | **Test Execution date:** |
| **Description:** The below test case is used to find and display all the consultations for the provided class value |  |
|  |  |
|  |  |
| **Pre-conditions:** The user should provide the class value whose consultations need to be displayed | |
| **Dependencies:** Depends on the class module for searching given class the and consultations module for viewing all the consultations for the given class | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Step | Test steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
| 1 | Access HRIS System | <Valid User> | A message “The Class detail is Invalid” is shown |  |  | An error message is displayed |
| Click on the Class Search Button | <Invalid Class value> |
| 2 | Access HRIS System | <Valid User> | A message “No Consultations exist” is shown |  |  | An error message is displayed |
| Click on the Class Search Button | <Provide valid Class value> |
| Click on View Consultations Button | < Consultations do not exist in the database > |
| 3 | Access HRIS System | <Valid User> | Displays all the Consultations for the given Class value |  |  | All the Consultations for the given class are displayed |
| Click on the Class Search Button | <Provide valid Class value> |
| Click on View Consultations Button | < Consultations exist in database > |

|  |
| --- |
| **Post-conditions:** If the user is a valid user but provided class value is Invalid then the “The Class detail is Invalid” error message should be displayed.  If the user is valid and provided class value is also valid but there are no consultations for a given class, then the “No Consultations exists” message should be displayed.  If the user is valid, provided class value is Valid and there are consultations for the given staff then all the Consultations for the given class should be displayed. |

1. **Test cases for searching and viewing all the classes for a Unit:**

|  |  |
| --- | --- |
| **Test Case ID:** 8  **Test Type:** Integration Testing | **Test Designed by:** |
| **Test Priority (Low/Medium/High):** High | **Test Designed date:** |
| **Use Case Name:** use\_case\_8 | **Test Executed by:** |
| **Test Title:** Search and view all classes for the given unit value | **Test Execution date:** |
| **Description:** The below test case is used to find and display all the classes for the provided unit value |  |
|  |  |
|  |  |
| **Pre-conditions:** The user should provide the unit value whose classes need to be displayed | |
| **Dependencies:** Depends on the unit module for searching given unit and the class module for viewing all the classes for the given unit | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Step | Test steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
| 1 | Access HRIS System | <Valid User> | A message “The Unit detail is Invalid” is shown |  |  | An error message is displayed |
| Click on the Unit Search Button | <Invalid Unit value> |
| 2 | Access HRIS System | <Valid User> | A message “No Class exists” is shown |  |  | An error message is displayed |
| Click on the Unit Search Button | <Provide valid Unit value> |
| Click on View Classes Button | < Class does not exist in database > |
| 3 | Access HRIS System | <Valid User> | Displays all the Classes for the given Unit value |  |  | All the classes for the given unit are displayed |
| Click on the “Unit Search” Button | <Provide valid Unit value> |
| Click on View Classes Button | < Class exists in the database > |

|  |
| --- |
| **Post-conditions:** If the user is valid but provided unit value is Invalid then the “The Unit detail is Invalid” error message should be displayed.  If the user is valid and provided unit value is also valid but there are no classes for the given unit, then the “No Class exists” message should be displayed.  If the user is valid, provided the unit value is Valid and there are classes for the given unit then all the classes for the given unit should be displayed. |

# Test Plan Validation:

## Discussion- plan completeness

The above test cases check for completeness in the provided units and by integrating the other units. The system provides the expected outcome; hence it is compliant and can be used without any scope for error. All the bugs in the system are discovered and every aspect is examined thoroughly. The accuracy of the system is better since all the parts of the system are covered in the test cases.

In the designed test cases unit testing is used to check whether the different modules like staff, unit, class, and consultation are working without any failure. Once we ensure the accuracy of all the units we proceed with Integration testing to check whether the combination of two units is working correctly. Different combinations like the class and staff, consultation and class, class and unit, and consultation and staff are made to check the accuracy after integrating the modules.

After implementing the above test cases the unit testing and Integration testing is working with accurate results hence ensuring the completeness of System-level correctness. The above test cases yield the required result and thereby the system completeness can be concluded.

## Verify- level of confidence:

The system outcomes and the expected output match and the system are highly accurate. A detailed report documenting the test strategies, resources, estimations, and objectives is maintained to make sure the system is working accurately. The provided test cases offer a clear guide for performing the testing and the higher the accuracy, the greater the level of confidence.

The different types of testing used for the given operations are as follows:

|  |  |
| --- | --- |
| **Operation** | **Testing type** |
| showing all the units | Unit Testing |
| showing all the staff | Unit Testing |
| showing all the classes | Unit Testing |
| showing all the consultations | Unit Testing |
| Searching and viewing all the classes for a staff member | Integration Testing |
| Searching and viewing all the consultations for a staff member | Integration Testing |
| Searching and viewing all the Consultations for a Class | Integration Testing |
| Searching and viewing all the classes for a Unit | Integration Testing |

Unit testing is performed for different operations like showing different units, staff, classes, and consultations. This ensures that there is no error in the given module and works accurately.

Similarly, Integration testing is performed for different operations like Searching and viewing all the classes for a staff member, all the consultations for a staff member, all the Consultations for a Class, and all the classes for a Unit. This ensures that the system works correctly even after combining two units.

# References

Basu, A., 2015. *Software Quality Assurance.* s.l.:s.n.

Patton, R., n.d. *Software Testing.* 2nd ed. s.l.: Pearson Education.

Ramesh, G., 2007. *Software Testing Principles and Practices.* 2nd ed. s.l.:s.n.