**Uncommon Solutions**

**Group 3**

**Test and Evaluation Master Plan**

**UNCOMMON SOLUTIONS PROJECT PLAN**

**Prepared By**

|  |  |
| --- | --- |
| Document Owner(s) | Project Role |
| Chase Thorpe | Test Engineer |

**Requirements Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Change Description |
| 1.0 | 11/07/2019 | Chase Thorpe | Document creation |

**Summary Details**

|  |  |
| --- | --- |
| Participants | Name(s) |
| Project Manager: | Michael Kiefer |
| Project Team: | Hither Guzha – Technical Writer  Andrew Benson - Software Engineer  Donn Eddy - UX/HCI  Sean Mooneyham - Integration Engineer  Chase Thorpe - Test Engineer |
| End Users: | HR Departments |
| Description w/ Goal: | The purpose of this project is the implementation of an HR database and front end for personnel tracking. This Test Plan will describe methods for the thorough evaluation of this project. |

[**1**](#_heading=h.3znysh7) **PURPOSE 5**

[**2**](#_heading=h.2et92p0) **BACKGROUND 5**

[**3**](#_heading=h.sqyw64) **Scope 5**

[**4**](#_heading=h.tyjcwt) **GLOSSARY 5**

[**5**](#_heading=h.3dy6vkm) **LIMITATIONS AND TRACE ABILITY 5**

[5.1](#_heading=h.1t3h5sf) Limitations 5

[**6**](#_heading=h.3cqmetx) **Test Plans 6**

[6.1](#_heading=h.4d34og8) Test Levels 6

[6.1.1](#_heading=h.2s8eyo1) API Testing 6

[6.1.2](#_heading=h.17dp8vu) Integration Testing 6

[6.1.3](#_heading=h.3rdcrjn) System Testing 6

[6.2](#_heading=h.26in1rg) Test Environment and Schedules 6

[6.2.1](#_heading=h.lnxbz9) Software Description 6

[6.2.2](#_heading=h.35nkun2) Milestones 6

[6.2.3](#_heading=h.1ksv4uv) Organizations and Locations 6

[6.2.4](#_heading=h.44sinio) Resource Requirements 7

[6.2.5](#_heading=h.2jxsxqh) Testing Material 7

[6.2.6](#_heading=h.z337ya) Test Training 7

[6.2.7](#_heading=h.3j2qqm3) Test Methods and Evaluation 7

[**7**](#_heading=h.1y810tw) **TEST DESCRIPTION 8**

[7.1](#_heading=h.4i7ojhp) User Login (Valid) 8

[7.1.1](#_heading=h.2xcytpi) Test Description 8

[7.1.2](#_heading=h.1ci93xb) Control 8

[7.1.3](#_heading=h.3whwml4) Inputs 8

[7.1.4](#_heading=h.2bn6wsx) Outputs 8

[7.1.5](#_heading=h.qsh70q) Procedures 8

[7.2](#_heading=h.3as4poj) User Login (Invalid) 9

[7.2.1](#_heading=h.1pxezwc) Test Description 9

[7.2.2](#_heading=h.49x2ik5) Control 9

[7.2.3](#_heading=h.2p2csry) Inputs 9

[7.2.4](#_heading=h.147n2zr) Outputs 9

[7.2.5](#_heading=h.3o7alnk) Procedures 9

[7.3](#_heading=h.23ckvvd) Unique Username 9

[7.3.1](#_heading=h.ihv636) Test Description 9

[7.3.2](#_heading=h.32hioqz) Control 9

[7.3.3](#_heading=h.1hmsyys) Inputs 9

[7.3.4](#_heading=h.41mghml) Outputs 9

[7.3.5](#_heading=h.2grqrue) Procedures 9

[7.4](#_heading=h.vx1227) Verification of User Role Privilege 10

[7.4.1](#_heading=h.3fwokq0) Test Description 10

[7.4.2](#_heading=h.1v1yuxt) Control 10

[7.4.3](#_heading=h.4f1mdlm) Inputs 10

[7.4.4](#_heading=h.2u6wntf) Outputs 10

[7.4.5](#_heading=h.19c6y18) Procedures 10

[7.5](#_heading=h.3tbugp1) Authentication Logging 10

[7.5.1](#_heading=h.28h4qwu) Test Description 10

[7.5.2](#_heading=h.nmf14n) Control 10

[7.5.3](#_heading=h.37m2jsg) Inputs 10

[7.5.4](#_heading=h.1mrcu09) Outputs 11

[7.5.5](#_heading=h.46r0co2) Procedures 11

[7.6](#_heading=h.2lwamvv) Logging Changes to Personnel Record 11

[7.6.1](#_heading=h.111kx3o) Test Description 11

[7.6.2](#_heading=h.3l18frh) Control 11

[7.6.3](#_heading=h.206ipza) Inputs 11

[7.6.4](#_heading=h.4k668n3) Outputs 11

[7.6.5](#_heading=h.2zbgiuw) Procedures 11

[**8 APPROVALS 11**](#_heading=h.1egqt2p)

# 

# PURPOSE

The Uncommon Solutions HR Management System provides access to personnel records of a company. The system separates user roles into four categories: User, Privileged User, Records Admin, and Account Admin. Each of these roles possess different levels of access and control over personnel records. Access to this system is controlled via authentication mechanisms. Relevant actions taken by users (outlined below) are logged by the system.

# BACKGROUND

Due to geographic separation of team, testing will be performed on cloud infrastructure; specifically, on an AWS EC2 instance. This will also serve to demonstrate execution of a deployed installation with users at multiple locations.

# SCOPE

***In Scope:***

The test plan for the Uncommon Solutions HR Management System will consider the correct execution of all code blocks relevant to the creation, modification, retrieval, and deletion of personnel records as well as correct authentication/authorization handling and logging as IN SCOPE.

***Out of Scope:***

Performance, scalability, stress testing, database testing, and anything not directly linked to the items listed above will be considered OUT OF SCOPE.

# GLOSSARY

NA

# LIMITATIONS AND TRACE ABILITY

## Limitations

**Risk Mitigation**

Time Constraints Set test priority for each activity

Team lacks experience in software Conduct research on industry best practices

development/testing and collaborate to mitigate individual gaps

# TEST PLANS

## Test Levels

Due to time/budget constraints, only the following types of testing will be performed for this system:

### API Testing

All read/write operations against personnel records, authentication processes, and logging functions will be tested independent of other modules.

### Integration Testing

At this level of testing, interaction between relevant modules will be tested. For example, ensuring that the correct audit logs are generated when a user logs in or logs outs, etc.

### System Testing

This level of testing will examine the system as a whole and ensure that the system functions as intended based on predefined acceptance criteria.

## Test Environment and Schedules

This section provides a brief description of the inputs, outputs, and functions of the software being tested.

### Software Description

The Uncommon Solutions HR Management System is a standalone system that manages the personnel records for the employees of a company.

### Milestones

Test Plan: Due 10NOV19

Project Design: Due 17NOV19

Phase 1 Source Code: Due 24NOV19

Phase 2 Source Code: Due 01DEC19

Phase 3 Source Code: Due 08DEC19

Final Product: Due 15DEC19

### Organizations and Locations

This software will be tested on an AWS EC2 instance.

Public DNS (subject to change): ec2-54-145-217-172.compute-1.amazonaws.com

IPv4 Public IP (subject to change): 54.145.217.172

### Resource Requirements

This section and associated statements define the resource requirements for the testing.

#### Equipment

The only required equipment is a computer with an internet connection that will connect to the AWS EC2 instance where the system lives.

#### Software

The computer must have a modern browser installed (Chrome, Firefox, Edge, Safari, Opera). Much of the testing will be conducted with manual inputs so very little additional software is required. A 3rd party testing framework like Jest or Mocha may be included as needed.

#### Personnel

Testing will be performed primarily by the Test Manager, Chase Thorpe. The entire team will provide oversight and support on testing.

### Testing Material

NA

### Test Training

NA

### Test Methods and Evaluation

This section documents the test methodologies, conditions, test progression or sequencing, data recording, constraints, criteria, and data reduction.

#### Methodology

Much of the initial testing for this system will involve unit testing of the various components of the system to ensure they function as intended. Each function in the read/write operations of personnel records, authentication processes, and logging functionality will be tested in isolation before being tested at the subsystem level.

#### Conditions

Test data will be entered in real time. Because stress/performance testing is out of the scope of this test, transactions per second and similar time-based testing will not be performed.

#### Test Progression

As mentioned in the methodology section, individual unit tests will be conducted in succession. After the unit tests have been completed, interactions between the components of the system will be tested.

#### Data Recording

Test results will be recorded in a shared document between the team.

#### Constraints

The team foresees no constraints other than those mentioned above at this time.

#### Criteria

Each test will be performed with two valid inputs and two invalid inputs. If all of these inputs line up with expected output, the test will be considered successful.

#### Data Reduction

NA

# TEST DESCRIPTION

## User Login (Valid)

### Test Description

This test will ensure that a valid user can successfully login with a username and password

### Control

This will be a manual test. Username and Password will be entered, and the success or failure of the attempt will be recorded.

### Inputs

Two inputs will be required for this test. First, a valid username. Second, the password associated with that username. These inputs will have to comply with the predetermined username and password policy.

### Outputs

With a valid username and password, we expect to be successfully authenticated and shown the user dashboard. In addition, the successful login will be recorded in the authentication audit log.

### Procedures

Prior to the test being performed, a user account will be manually created. Next, the tester will navigate to the login screen and enter the username and password into the specified fields. Upon submission of credentials, either a success or failure message will be displayed. This result will be recorded. The expectation for a successful attempt is the success message and redirection to the user dashboard.

## User Login (Invalid)

### Test Description

This test will ensure that an invalid user is prevented from logging in.

### Control

This will be a manual test. Username and Password will be entered, and the success or failure of the attempt will be recorded.

### Inputs

Two inputs will be required for this test. The first invalid input will be an invalid username and the second invalid input will be an invalid password. These inputs will have to comply with the predetermined username and password policy.

### Outputs

With an invalid username and password, we expect to be prevented from logging in to the system. In addition, the failed attempt will be logged in the authentication audit log.

### Procedures

No user account will need to be created for this test. Two incorrect inputs will be entered and submitted. Regardless of the combination of incorrect inputs (correct username/incorrect password, incorrect username/correct password, incorrect username/incorrect password), we expect to see an authentication failure message and be prevented from moving past the login screen.

## Unique Username

### Test Description

This test will ensure that duplicate usernames cannot be created.

### Control

This will be a manual test at the user creation screen.

### Inputs

The only input required will be the username being tested.

### Outputs

We expect a message to be shown for user creation error stating that an account of that name already exists.

### Procedures

Prior to the test being performed, a user account will be manually created. Next, the tester will navigate to the user creation screen and attempt to create a new user with the same username as the previously created account.

## Verification of User Role Privilege

### Test Description

This test will ensure that a valid user can interact with the system within the established controls and privileges granted by the various roles within the system (user, privileged user, records admin, account admin)

### Control

This will be a manual test. Each account type will be logged into and various actions will be attempted. Results of successes and failures for both authorized and unauthorized actions will be recorded.

### Inputs

Usernames and passwords for each account will be used as inputs. Each account will attempt to perform authorized actions within their role as well as unauthorized actions to ensure permissions are performing as expected.

### Outputs

We expect regular users to be able to access their personnel records and make minor changes to certain fields. We expect privileged users to have read access to all personnel records but only be able to make minor changes to their own personnel record. The Records Admin should have full read/write access to all personnel records. The Account Admin has full read/write access to user accounts and no access to the personnel records.

### Procedures

Prior to the test being performed, accounts will be created for each role. Each account will be logged into and various authorized actions will be attempted. Next, the tester will attempt to take unauthorized actions such as accessing data outside of their role. Results of all of these actions will be recorded for mitigation.

## Authentication Logging

### Test Description

This test will ensure that authenticating to the system trigger the logging function to run.

### Control

This will be a manual test. Username and Password will be entered, and the authentication log will be checked for an entry.

### Inputs

A valid username and password will be entered. Upon logging in, the tester will check the authentication log.

### Outputs

We expect a log to be generated in the authentication log showing the user who authenticated, a timestamp for the attempt, as well as a success/fail status

### Procedures

The tester will attempt to login to the system with a valid username and password. Upon success, the authentication log will be checked for the correct entry.

## Logging Changes to Personnel Record

### Test Description

This test will ensure that any modification of a personnel record will trigger a log to be generated.

### Control

This will be a manual test. A personnel record will be created, modified, and deleted by the tester.

### Inputs

Credentials for the Records Admin role will be needed as inputs. A personnel record will be created with all necessary fields.

### Outputs

We expect three logs to be generated. One upon record creation, one upon modification, and one upon deletion.

### Procedures

Tester will log in as the Records Admin. Next, the tester will attempt to create a new personnel record. After creation, the tester will modify any field in the record. After the modification, the tester will delete the personnel records. Upon successful deletion, tester will navigate to the records log and check that the three logs were generated correctly for each action taken by the tester. Results will be recorded.

## Test Results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Id | Test Name | Inputs | Expected Output | Actual Output | Pass/Fail |
| 7.1 | User login (valid) | Valid username and password | Successful login and redirection to user’s dashboard | Output coincided with expectations | Pass |
| 7.2 | User login (invalid) | Invalid username and password | Incorrect credentials warning and remain at login screen | Output coincided with expectations | Pass |
| 7.3 | Unique username | Duplicate username at user creation | “User already exists” warning | Output coincided with expectations | Pass |
| 7.4 | Verification of privilege | Credentials for each user type | No warnings or system prevention while performing authorized user functions | Output coincided with expectations | Pass |
| 7.5 | Authentication Logging | Valid credentials | Log entry generated in authentication log | Output coincided with expectations | Pass |
| 7.6 | Changes to personnel record modification logging | Records admin credentials and test data for modifications | Three logs to be generated. One upon record creation, one upon modification, and one upon deletion. | Output coincided with expectations | Pass |

# 8 APPROVALS

I have read the above Test Plan and will abide by its terms and conditions and pledge my full commitment and support for the Master Test and Evaluation Plan.

**Sign-off Sheet**

**Prepared by** \_\_\_ Chase Thorpe\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Test Engineer – Chase Thorpe

**Approved by**\_\_\_Michael Kiefer\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Manager– Michael Kiefer

**Approved by**\_\_\_Andrew Benson\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Team Member– Andrew Benson

**Approved by**\_\_\_Donn Eddy\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Team Member– Donn Eddy

**Approved by**\_\_\_Sean Mooneyham\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Team Member– Sean Mooneyham

**Approved by**\_\_\_Hither Guzha\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Team Member– Hither Guzha

**References:**

Pineda, R.L. & Kilicay-Ergin, N. (2010). System verification, validation, and testing. Retrieved from: <https://content.taylorfrancis.com/books/e/download?dac=C2009-0-20593-8&isbn=9781439809273&doi=10.1201/b10452-8&format=pdf>

Alruwaili, F.F. & Gulliver, T.A. (2015). SecSDLC: A practical life cycle approach for cloud-based information security. *International Journal of Research in Computer and Communication Technology*. Retrieved from: <https://pdfs.semanticscholar.org/0048/9d337519e36b7ac3d916fce2baebda467a5e.pdf>

Rout, T.P. & Sherwood, C. (1999). Software engineering standards and the development of multimedia-based systems. *Proceedings 4th IEEE International Software Engineering Standards Symposium and Forum*. Retrieved from: <https://ieeexplore.ieee.org/abstract/document/766594>

Habayeb, A.R. (2005). Architecture-Based Systems Engineering and Integration. Retrieved from: <https://ndiastorage.blob.core.usgovcloudapi.net/ndia/2005/systems/wednesday/habayeb.pdf>

Adam, Singh, R., Aggarwal, A., Aslam, S., Audrey, Alloch, … Psc. (2019, November 10). Test Plan Tutorial: A Guide To Write A Software Test Plan Document From Scratch. Retrieved November 10, 2019, from <https://www.softwaretestinghelp.com/how-to-write-test-plan-document-software-testing-training-day3/>

Test Plan. (2018, July 15). Retrieved November 10, 2019, from <http://softwaretestingfundamentals.com/test-plan/>