e-Care Clinicals Application
Version 1.6

QA Test Plan

Prepared By:
Lai Zhang
Runyang Zhou
Kaiyuan Zhao

Date:
2019.12.10

Table of Contents

1 Document Acceptance and Sign-Off

By signing below I acknowledge that I have read the entire contents of this document and accept the document in this form as reasonably fulfilling the goals described in the section titled Document Purpose. I further agree that this will constitute the document of record and cannot be changed without review and acknowledgement of the groups shown below:

Group / Role	Approver Name	Approver Signature	Date Approved
Tester	Lai Zhang	Lai Zhang	2019.12.07
Tester	Runyang Zhou	Runyang Zhou	2019.12.07
Tester	Kaiyuan Zhao	Kaiyuan Zhao	2019.12.07

2 Revision History

Document/Department Editor:			
Date	**Revision #**	**Editor**	**Description of Change**
2019.11.27	1.0	Lai Zhang	Adding testing plan
2019.11.28	1.1	Runyang Zhou	Specify test schedule and testing tools
2019.11.29	1.2	Lai Zhang	Created the test cases
2019.11.29	1.3	Runyang Zhou	Created assumption and Constraints
2019.11.30	1.4	Kaiyuan Zhao	Created risk and dependencies
2019.11.31	1.5	Runyang Zhou	Created Regression Test plan and update references
2019.12.6	1.6	Kaiyuan Zhao	Complete testing and update test plan documentation

3 Glossary

Term	Definition
BRD	Business Requirement Document
FSD	Functional Specification Document
DI	Demographic Information
TC	Test cases
CS	Clinical Systems
N/A	Not Applicable
SL	Spectra Lab

4 Document Purpose

The test plan document is created in the first phase of the project, the purpose is to list requirements for testing E-Care Clinical Application, the plans and any update in each phase during the project testing.

5 Introduction

5.1 Test Product Overview

Describe the part of the product to be tested.

- Specify the major new or modified capabilities included in this test
- Detail the differences between this baseline or release and the previous phase and/or system
- Detail the release undergoing testing in terms of its interfaces, role, and responsibility within the environment
- A context diagram of the product within its environment showing interfaces may be applicable
- Detail the hardware and software configuration items of the product undergoing integration and test, especially dependencies, environment requirements, and risks.

5.2 Test Planning

5.2.1 Purpose

The purpose of testing E-care clinical application is to ensure the system to work with bug free and function well, meantime, meet the demand of business requirement document and function specification document under all kinds of test cases and test sceneries.

5.2.2 Objectives

The objectives are the goals during the testing of application, which includes:

a. Finding defects which may caused by the mistakes made by the developers.

b. Gaining confidence in and providing information in all levels of software quality.

c. Preventing defects that may undermine the function of application.

d. Making sure all the test results meet function specification document and business requirement document.

5.2.3 Scope

The E-care clinical application is consisting of patient admission, treatments, charges, billing, claims and reports. So the scope includes:

a. Checking the admission of different users(existing/new) with different financial clearance(yes/no)

b. Checking the treatment drug type with different insurance type.

c. Checking the treatments is allowed or not with different clinical clearance(yes/no)

d. Checking the treatment information is sent to the external system.

e. Checking the charges is made or not with different clinical clearance(yes/no)

f. Checking the billing is sent or not.

g. Checking the report is sent or not.

5.3 Definitions

Unit Test:

Validate each unit of module in the source code is working well as developer's plan and make sure the code meets the expected output and working well.

Integration Test:

Combined multiple modules and test as a group in each individual unit in order to make sure multiple modules can interact well.

Regression Test:

Repeat unit test after recent update or test for the application to make sure the existing functionalities work fine.

Performance Test:

Check how the system behaves and performs, it usually examines stability, reliability and speed.

Load Test:

Load test is to check how application perform under a heavy number of usages in a period of time.

6 Test Environments

6.1 Test Data

1. Admit_new_01: New user with financial clearance set to yes, admitted into the system, DI information and folder created, DI information sent to Spectra

2. Admit_exsiting_01: Existing user with financial clearance set to yes, admitted into the system, DI information and folder updated, DI information sent to Spectra

3. Admit_new_02: New user with financial clearance set to no, not admitted into the system, DI information and folder not created, DI information not sent to Spectra

4. Admit_exsiting_02: Existing user with financial clearance set to no, not admitted into the system, DI information and folder not updated, DI information not sent to Spectra

5. Treatment_new_01: New user with clinical clearance set to yes, medicare/medicaid insurance, treatment with hep-ran drug, DI infromation sent to external system

6. Treatment_new_02: New user with clinical clearance set to yes, commercial insurance, treatment with xer-ran drug, DI infromation sent to external system

7. Treatment_new_03: New user with clinical clearance set to no, insurance not applicable, treatment not applicable, DI infromation not sent to external system

8. Treatment_existing_01: Existing user with clinical clearance set to yes, medicare/medicaid insurance, treatment with hep-ran drug, DI infromation sent to external system

9. Treatment_existing_02: Existing user with clinical clearance set to yes, commercial insurance, treatment with xer-ran drug, DI infromation sent to external system

10. Treatment_existing_03: Existing user with clinical clearance set to no, insurance not applicable, treatment not applicable, DI infromation not sent to external system

11. Charges_user_01: Treated user(new or existing&medicare/medicaid or commercial insurance) with clinical clearance set to yes, charges applied

12. Charges_user_02: Treated user(new or existing&medicare/medicaid or commercial insurance) with clinical clearance set to no, charges not applied

13. Claims_user_01: Treated user with medicare/medicaid insurance, claim sent

14. Claims_user_02: Treated user with commercial insurance, claim sent

15. Reports_user_01: Report sent on second day after treatment

6.2 Test Tools

a. Automation tools:

Selenium & UFT

b. Bug tracking tools:

JIRA & FogBugz

c. Performance testing tools

WebLOAD & LoadNinja

d. Penetration testing tools

Nmap & wireshark

7 Roles and Responsibilities

7.1 Monitoring Testing

- Track and report defect status as per the defect management process.
- Establish, document, and publish test status reports as defined by implemented TS test methodology.
- Testing schedule slippage recovery plans (including code rework) for testing should be considered and documented accordingly.
- Ensure that any approved changes can be incorporated into the testing, referencing the change control process as appropriate.
- Schedule test meetings for monitoring progress as appropriate.

7.2 Entry and Exit Criteria

- The Unit and SIT have been completed and no Sever defects are open.
- The user acceptance testing have been completed and no defects are found.

8 Assumptions, Constraints, Dependencies, Risks, and Limitations

8.1 Assumptions

The technology team will come up with software supports during each testing stages and functional design, for example the technology team will provide necessary software and hardware tools for tests.

Assume the clinic will support UAT for the testing team and assist with the execution of the UAT.

8.2 Constraints

Time(too short)

Cost(too high)

Quality(expectations too high – zero defects)

Test Environment(incomplete, not set up)

Test Tools(lack of, not suited, difficult to maintain)

Test Activity(none or wrong testing technique applied)

8.3 Risks

Risk	Mitigation
The integration test could not finish on time.	Plan for at least one week for assess.
Change plans during testing.	Plan ahead for all possible results and try to avoid changes.
Change or upgrade the functionality in the software may cause the loss of test cases we already written.	Make a copy for the data before any change or upgrade, and we may use it if we loss anything.

8.4 Dependencies

Internal Dependencies: A dependency is a logical relationship between two (or more) of your work breakdown structure(WBS) Task (Activities), or between a task(activity) and a millstone. In our project we need to finish the current job in order to start the next one.

9 Processes, Modules, or Components

9.1 Test Planning

a. Identify requirements according to the business requirement document and functional specification document.

b. Create test cases and plan the test sceneries.

c. Make the test schedule.

d. Research and decide the test tools.

e. Unit test.

f. System and integration test.

g. Performance and load test.

h. User acceptance test.

i. Regression test.

j. Form the report.

10 Test Schedule

Task Name	Description	Effort	Start date
Task identification	Go through BRD and FSD	1 day	2019.11.27
Test cases	Find all possible sceneries and learn the application to create the test cases	3 days	2019.11.28
Testing schedule	Make the test schedule with detailed information	0.5 day	2019.12.01
Test tools	Research the test tools and decide the right one	0.5 day	2019.12.01
Testing	Unit test& system integration test, performance test, UAT test & regression test	3 days	2019.12.02
Report	Review the test result and form the report	1 day	2019.12.05

11 Expected Results

Test Case	Expected Result
Admit_new_01	New user with financial clearance set to yes, admitted into the system, DI information and folder created, DI information sent to Spectra
Admit_exsiting_01	Existing user with financial clearance set to yes, admitted into the system, DI information and folder updated, DI information sent to Spectra
Admit_new_02	New user with financial clearance set to no, not admitted into the system, DI information and folder not created, DI information not sent to Spectra
Admit_exsiting_02	Existing user with financial clearance set to no, not admitted into the system, DI information and folder not updated, DI information not sent to Spectra
Treatment_new_01	New user with clinical clearance set to yes, medicare/medicaid insurance, treatment with hep-ran drug, DI infromation sent to external system
Treatment_new_02	New user with clinical clearance set to yes, commercial insurance, treatment with xer-ran drug, DI infromation sent to external system
Treatment_new_03	New user with clinical clearance set to no, insurance not applicable, treatment not applicable, DI infromation not sent to external system
Treatment_existing_01	Existing user with clinical clearance set to yes, medicare/medicaid insurance, treatment with hep-ran drug, DI infromation sent to external system
Treatment_existing_02	Existing user with clinical clearance set to yes, commercial insurance, treatment with xer-ran drug, DI infromation sent to external system
Treatment_existing_03	Existing user with clinical clearance set to no, insurance not applicable, treatment not applicable, DI infromation not sent to external system
Charges_user_01	Treated user(new or existing&medicare/medicaid or commercial insurance) with clinical clearance set to yes, charges applied
Charges_user_02	Treated user(new or existing&medicare/medicaid or commercial insurance) with clinical clearance set to no, charges not applied
Claims_user_01	Treated user with medicare/medicaid insurance, claim sent
Claims_user_02	Treated user with commercial insurance, claim sent

Reports_user_01	Report sent on second day after treatment

12 References

12.1 Standards, Plans, and Procedures

Reference Document	Location
Assumptions and Dependencies in Test Plan	https://checkykey.com/assumptions-and-dependencies-in-test-plan
System level consideration	https://www.tiiproject.com/systems-level-considerations-long-term-investor/

13 Partnership Testing

13.1 Additional Testing

13.1.1.1 Vendor Acceptance Testing

This test will be done by the vendor, it will run in the real life scenario test case to check whether the system has defects or not.

13.1.1.2 Configuration Management

This is a process of establishing and maintaining a product's performance, functional and physical attributes with its requirement, design, and functionalities through its life.

13.1.1.3 System Level Considerations

System level considerations can help investors to manage long-term risks and rewards while seeking competitive portfolio-level returns

13.1.1.4 Software Level Considerations

There are multiple software level considerations upon applications testing:

1. Reliability: Software applications should work exactly as the end-users expects, there is no room for defects or ambiguity, it is mandatory to make sure it works properly.

2. Usability: The application should be easy to use and easy to understand by the end-users, this has been a key consideration in the software design, develop and testing.

3. Security: The application is deployed online, so it always runs big thread from hackers every day, we need to make sure all data is secured in the application prevent hackers to attack.

13.2 Regression Testing

13.2.1 Regression Test

Regression Test is to retest of components in every new release cycle. The goal is to verify the modifications have not caused unintended effects and to make sure no bugs or crush after new release. We need to follow these steps for regression testing:

Plan the basic regression testing strategies

Build regression testing environments

Execute regression test after each version release