Master Test Plan

for the

Construction Department of Southern Illinois University Edwardsville

by

Bryan Allen

Daniel Grote

Zach Smith

of

CS425 / CS499 Senior Project

Concrete Drying Application Team

CS-TEST

Revision 1.0.5

As Of: 11 November 2014

Change Log:

Revision	Change Note(s)

Reviewed and Approved By:

Name	Signature	Date

Table of Contents

1 INTRODUCTION

1.1 Scope

This test plan includes component, component integration, system, and acceptance levels of testing on the entire project. One constraint to the test plan is getting access to the server so testing such as some system testing has to wait until access to a server is granted. Since an iterative development methodology is used there will be multiple iterations of each level of testing and testing levels do not necessarily have an exact order. In general though, when adding/modifying a feature, component testing then integration testing occurs. Then acceptance testing occurs. Once all the components pass the component integration and acceptance testing by the client then system testing will be performed. Leaving system testing tell after the client has accepted the software components means not having to re-do system testing when a new software component is added or a change in requirements occurs. But, if there is a big problem found when system testing this can cause the project to not be finished on time. So not all system tests should be left for the end. After system testing another acceptance test will be performed.

1.2 Features to be tested:

- Chart generation
- Creating account
- Lost password
- Email
- Login functionality

- Project table
- Future notification
- Change in state notification
- Future notification table
- Change in state notification table
- Web site usage statistics view
- Web site usage statistics table
- Database reset functionality
- Security

1.2 References

IEEE Std 829-2008 – IEEE Standard for Software and System Test Documentation

CS-SPEC - Project Specification

CS-PLAN – Project Plan

1.3 System overview and key features

See Project Specification

1.4 Test overview

1.4.1 Organization

Test tasks will be assigned like sprint tasks during sprint planning. After a sprint is complete during the sprint review the test tasks will be reviewed and a unanimous decision has to be made to approve the test processes. If issues are raised by the testing tasks the person that was the task leader for that test task will be in charge of resolving the task unless it is a major issue then the team will reach a unanimous decision on how to proceed.

1.4.2 Master test schedule

Sprint 4 - Feb 2nd

Functionality

- Login and logout functionality
- Add user project table
- Add project functionality
- Chart generation
- About (info) page

Testing**:**

- Component / Unit testing
    - adding/editing/modifying project functionality

  - project table
  - Chart generation
  - Sessions
  - Login
  - Logout
- Component Integration testing
  - project table
  - project functionality
  - chart generation
  - user table
  - login
  - logout
  - user account functionality
  - session functionality
- System testing
  - Exception Handling
    - Login form
    - Create account form
    - Add project form
  - Compatibility testing
    - Test that the application looks and acts the same on Internet Explorer, Google Chrome, and Firefox
  - Security testing

        - Integrity
        - Confidentiality
        - Authentication
        - Authorization
- Acceptance testing – Client Satisfaction
    - Chart generation
        - Colors
        - Data labels
        - Titles/subtitle/plot band titles
        - Points
        - Grid axis
        - Line
        - Tooltip
        - Change concrete temp
        - Print/download chart
        - Metric conversion
        - Other
    - Changing concrete temp functionality
    - About (information) page
        - Information
        - GUI (layout)

Sprint 5 Feb 23rd

Note: Expecting server to be available

Functionality

- Add Future Notifications table
- Add ChangeInStateNotifciaiton table
- Add notification functionality
- Email
- Forgot password
- Chart functionality

Testing

- Component Testing / Unit Testing
    - Future Notfications table attributes
    - ChangeInStateNotifcation table attributes
    - Add/edit/delete notification
    - Additional email address(es) to email notifications
    - Notification form
    - Email
    - Chart functionality
- Component Integration testing
    - Project table
    - Notifications tables

- Email
- Forgot password
- Chart functionality

- System testing
  - Exception handling
    - Add notifications form
      - Attributes
        - Type of notification
        - Change in state parameter
        - Extra emails
        - Future date
  - Data migration testing – Transferring software to server
  - Performance testing
    - Response time of rendering graph is under 5 seconds
    - Login time and gathering users project and notification

- Acceptance testing - Client Satisfaction
  - Chart generation

Sprint 6 - March 16th

Functionality

- Add site usage table
- Add usage statistics functionality

Testing

- Component / Unit testing
    - Site usage table
    - Usage statistics functionality
- Component Integration testing
    - Entire database
    - Usage statistics functionality
- System testing
    - Compatibility testing
        - Test that the application looks and acts the same on Internet Explorer, Google Chrome, and Firefox
- Acceptance Testing – Client satisfaction
    - Site usage stats
    - User account functionality
    - Project functionality
    - Notifications functionality
    - Lost password

Sprint 7 - March 30th

Functionality

- Training on how to use view site stats
- Add maintainability functionality

- User manuals
- Beta testing

Testing

- Component / Unit testing
    - Recovery functionality (if something goes wrong be able to recover)
- Component Integration testing
    - Recovery functionality
    - Entire project
- System testing
    - Compatibility testing
        - Test that the application looks and acts the same on Internet Explorer, Google Chrome, and Firefox
    - Performance testing
        - Response time of rendering graph is under 5 seconds
        - Login time and gathering users project and notification
    - Recovery testing
    - Concurrent testing
    - Conformance testing
- Acceptance Testing
    - Site usage stats
    - User account functionality
    - Project functionality

- Notifications functionality
- Lost password
- Recover functionality

Sprint 8 - April 13th

Functionality

- n/a

Testing

- Acceptance Testing – Client Satisfaction

1.4.3 Integrity level schema

Integrity level scheme: Refer to IEEE Std 829-2008 Annex B

Resources: SIUE server, computer with firefox/chrome/ie

1.4.4 Resource summary

See project plan - Resources

1.4.5 Responsibilities

Tests will be assigned when planning a sprint when the team divides up the sprint tasks. When assigned a test they are the task leader for that task. If needed another person can be assigned to the task but there can only be one task leader.

2. Details of the Master test plan

2.1 Test processes including definition of test levels

2.1.1 Process Management

After a sprint is completed team members will fill out their test logs and a test report will be made for that sprint. During the sprint review with every team member present we will review and analysis the results.

If the testing task is scheduled to be completed and any of the following occur:

- If 1 or more team member(s) determines there hasn't been sufficient testing done for that task –
- If 1 or more team member(s) determine the team is behind schedule on the execution of the test plan
- If the test result did not conform to expectations

Then document what has happened in test results and analysis (in sprint output) and reason for deviation and see 2.3.3 Deviation policy for how to continue.

If an anomaly occurs – See 2.3.1 Anomaly resolution and reporting

If team unanimously determines test is done then document that the testing is complete for that task in the test results and analysis.

If testing task still needs more testing done but this was planned then document what has been tested in test results and analysis.

2.1 Activity: Management of test effort

After every sprint during the sprint review the team will identity any testing process improvements that could have or can be made, along with any insightful information they learned.

2.1.2 Process: Acquisition

Acquiring access to a server is required for this project. When the server is received data mitigation testing will need to be performed.

2.1.2.1 Activity: Acquisition support test

Task	Data mitigation
Methods	Ensure that web application is transferred to the server when received and all functionality is the same.
Inputs	Application on test server
Outputs	Working web application on new server
Resources	Server – Work with IT to get access
Risk(s) and Assumptions	Risk: Not getting access to late in the semester causing testing to be put off. Assumption: As soon as the server is ready this needs to be done as soon as possible and takes priority over everything else.
Roles and Responsibilities	Everyone will work on this but the quality assurance manager (Bryan) is the task leader.

2.1.3 Process: Supply

See Project Plan - Resources

2.1.4 Process: Development

Task	Graph performance testing
Methods	Ensure that the graph is rendered in less than 5 seconds. Needs to be tested with users and guests and when the server has a lot of activity.
Inputs	Zip code
Output	Fully rendered graph
Resources	Access to the server is granted
Risk(s) and Assumptions	Risk: That the graph will take longer than 5 seconds. Mitigation strategy: Optimize code so the graph can render faster.
Roles and Responsibilities	Daniel is the task leader for this. This will be tested every other sprint unless no functionality was added to the graph.

Task	Login performance testing
Methods	Ensure that it takes less than 5 seconds for user to login and gather users projects, notifications, and user account data.
Inputs	User login in data
Output	User home page
Resources	Access to the server is granted. Login functionality is complete.
Risk(s) and Assumptions	Risk: Login will take longer than 5 seconds Mitigation strategy: Optimize code so that user logins are faster.
Roles and Responsibilities	Zach will be the task leader for this. Bryan will provide support in this task when needed.

Task	Compatibility testing
Methods	Test that the application looks and acts the same on Internet Explorer, Google Chrome, and Firefox.
Resources	Working web application
Risk(s) and Assumptions	Risk: Functionality will work different on different internet browsers. Mitigation strategy: Update code to work the same on the different internet browsers.
Roles and Responsibilities	This will be checked every other sprint and a team member will be assigned to the task when planning the sprint.

Task	Conformance testing
Methods	Test that the application conforms with the requirements of the project
Resources	Access to the server is granted
Risk(s) and Assumptions	Risk: A requirement wont be met Mitigation strategy: Update application so the requirement will be met.
Roles and Responsibilities	A team member will be assigned to the task when planning the sprint.

Task	Concurrent testing
Methods	Testing web application that determines the stability of the application during normal activity,
Resources	Access to the server is granted
Risk(s) and Assumptions	Risk: A major bug will be found Mitigation strategy: Update code to fix the bug
Roles and Responsibilities	A team member will be assigned to the task when planning the sprint.

Task	Recovery testing
Methods	Testing application to determine how well an application able to recover from crashes and other problems.
Resources	Access to the server is granted.
Risk(s) and Assumptions	Risk: A major bug will be found Mitigation strategy: Update code to fix the bug
Roles and Responsibilities	A team member will be assigned to the task when planning the sprint.

Task	Security testing
Methods	Perform confidentiality, integrity, authentication, availability, and authorization testing to determine that the application protects data, protect against attacks, and maintains functionality as intended
Resources	Access to the server is granted. Log in functionality and database is complete.
Risk(s) and Assumptions	Risk: Application isn't secure Mitigation strategy: Update code to make the application more secure
Roles and Responsibilities	A team member will be assigned to the task when planning the sprint.

Task	Exception handling testing
Methods	Test to determine that exceptions are correctly handled in the application.
Resources	Access to the server is granted. Log in functionality and database is complete.
Risk(s) and Assumptions	Risk: Exception isn't handled. Mitigation strategy: Update code to handle the exception
Roles and Responsibilities	A team member will be assigned to the task when planning the sprint.

2.1.5 Process: Operation and Maintenance

We will install the project on the provided SIUE server. It will be accessible by Firefox, Google Chrome, or Internet Explorer.

We will not be providing maintenance after deploying the software.

2.2 Test documentation and reporting requirements

We will record every test in a Test Log – See IEEE Std 829-2008 Clause 13

If anything goes wrong during a test, we will complete an Anomaly Report – See IEEE Std 829-2008 Clause 14

After each test, a Test Report will be completed – See IEEE Std 829-2008 Clause 16

2.3 Test administration requirements

2.3.1 Anomaly resolution and reporting

If there are any anomalies, we will complete an Anomaly Report (IEEE Std 829-2008 Clause 14). The team will unanimously decide how to handle anomalies. Classification for software anomalies may be found in IEEE Std 1044TM-1993 [B13].

2.3.2 Task iteration policy

If a task input is changed the task will be repeated. If there were any anomalies during the execution of a task, the task will be repeated when the anomaly is fixed.

2.3.3 Deviation policy

If a requirement is added to the project and something new needs to be tested, a task will be created for any new components that need to be tested and the tentative schedule will be updated. The team will unanimously decide any deviations and document decisions made in the test results and analysis (sprint output).

2.3.4 Control procedures

When performing tests, only the person in charge of the test will be allowed to make changes to code relevant to that particular task.