

I.C.E.D Test Plan

I.C.E.D. Team

September 2018

Contents

1	Change Log	3
2	High level Goals	3
3	Test Case Format	3
3.1	<Test Case Identifier, Test Case Name>	3
3.2	Test Case Description	3
3.3	Milestone/time frame	3
3.4	Test Goals	4
3.5	Test Data	4
3.6	Test Tools	4
3.7	Test start-up procedure	4
3.8	Test close-down procedure	5
3.9	Preconditions	5
3.10	Test Steps	5
3.11	Expected Results	5
3.12	Notes/Comments	6
4	Requirements that will be tested	6
4.1	Crowd Sourced Interface	6
4.2	Communication Protocol	7
4.3	Logistics Interface	7
5	Test Cases - Crowd Sourced Interface	8
5.1	<CSI-001, Disaster Search and Selection >	8
5.1.1	Test Case Description	8
5.1.2	Milestone/time-frame	8
5.1.3	Test Goals	8
5.1.4	Test Data	8
5.1.5	Test Tools	8
5.1.6	Test start-up procedure	9
5.1.7	Test close-down procedure	9
5.1.8	Preconditions	9

5.1.9	Test Steps	9
5.1.10	Expected Results	10
5.1.11	Notes/Comments	10
5.2	<CSI-002, Disaster Geo-fencing >	10
5.2.1	Test Case Description	10
5.2.2	Milestone/time-frame	10
5.2.3	Test Goals	10
5.2.4	Test Data	11
5.2.5	Test Tools	11
5.2.6	Test start-up procedure	11
5.2.7	Test close-down procedure	11
5.2.8	Preconditions	11
5.2.9	Test Steps	12
5.2.10	Expected Results	12
5.2.11	Notes/Comments	13
6	Test Cases - Communication Protocol	13
6.1	<COM-001, Chain of Trust >	13
6.1.1	Test Case Description	13
6.1.2	Milestone/time-frame	13
6.1.3	Test Goals	13
6.1.4	Test Data	14
6.1.5	Test Tools	14
6.1.6	Test start-up procedure	14
6.1.7	Test close-down procedure	15
6.1.8	Preconditions	15
6.1.9	Test Steps	15
6.1.10	Expected Results	16
6.1.11	Notes/Comments	16
7	Test Cases - Logistics Interface	16
7.1	<LOG-001, Organizational View >	16
7.1.1	Test Case Description	16
7.1.2	Milestone/time-frame	16
7.1.3	Test Goals	16
7.1.4	Test Data	17
7.1.5	Test Tools	17
7.1.6	Test start-up procedure	17
7.1.7	Test close-down procedure	17
7.1.8	Preconditions	17
7.1.9	Test Steps	18
7.1.10	Expected Results	18
7.1.11	Notes/Comments	19

1 Change Log

Change Log					
Version #	Authored by	Change Date	Approved by	Approval Date	Reason
0.1	George Nelson	9/25/2018			Test Plan Draft

2 High level Goals

Objective: Provide an interface for coordinating civilians, first responders, and relief organizations during a disaster relief scenario.

- Provide an interface that is supplied with crowd sourced data
- Provide a communication protocol to distribute critical disaster relief information
- Provide a logistics platform that helps relief organizations respond more effectively to disasters

3 Test Case Format

3.1 <Test Case Identifier, Test Case Name>

3.2 Test Case Description

Provide a description of the test case that states its purpose i.e. systems being tested, modules validated, etc.

3.3 Milestone/time frame

Indicate the milestone in which the test case will be preformed under and if needed an approximate time frame when it would be attempted

3.4 Test Goals

Create a bullet-ed list that lists all the goals of the test case that should be answered as a result of the test.

- Example: Validate a correct transition between page a and page b on the front end
- Example: Validate that the landing page login button works

3.5 Test Data

Username	Password	Token
bob	123	abc
jim	567	djf
ed	cats	token

Table 1: Sample Test Data.

Provide the test data that will be used during the test case such as sample usernames or account activities

3.6 Test Tools

List the tools that will be used during the duration of the test case i.e. script handlers, emulators, etc.

- Tool ABC, the function of abc is to do xyz
- Tool 123, 123 will help do zyx

3.7 Test start-up procedure

Detail a brief procedure that setups up the test before test data is used to validate the test

1. Launch tools xyz
2. Configure profiles with abc
3. Allocate X resources
4. Etc.

3.8 Test close-down procedure

Detail a brief procedure that indicates the test case is complete and returns the system to its normal operation

1. Close xyz modules
2. Close tools
3. Close emulators
4. Etc.

3.9 Preconditions

List any precondition criteria that must be met before the test case can begin such as following the start up and close down procedures

3.10 Test Steps

List the steps in chronological order that will occur during the test. Attempt to make each step a separate action so that we can "check" each step off.

1. Perform test start-up procedure
2. Search for icedrelief.net
3. Enter test data into the username field
4. Enter test data into the password field
5. Etc.

3.11 Expected Results

List the corresponding expected results for each step from the test steps section above. This will allow us to validate the test against a set of results.

1. Start-up functions properly
2. Webpage loads correctly
3. Username field takes input
4. Password field take input
5. Etc.

3.12 Notes/Comments

Test Case ID	Module Test	Comments
CBX123	Login	It broke
BVC234	Load Events	It loaded the wrong events
GHB555	Authentication	Failed auth

Table 2: See Gitlab issue tracker for more details.

This table/section will detail any important events that occur that should be noted such as crashes or failed actions

4 Requirements that will be tested

Note: insert disclaimer about non mentioned requirements

4.1 Crowd Sourced Interface

1. Users shall be able to search and select for disaster areas
2. Users shall be able to select a disaster area with their relative location
3. Organizational users shall be able to approve/disapprove user submitted reports
4. Organizational users shall be enforced to use organizational tags for their reports
5. Users shall be able to submit reports
6. Report submissions shall prevent incorrect location tagging based on a user's location

4.2 Communication Protocol

1. The communication protocol shall provide a chain of trust system
2. The communication protocol shall provide specialized nodes that hold specific disaster information
3. The communication protocol shall provide the ability to add official data for distribution
4. The communication protocol shall prevent a cost associated with sealing blocks of data in the terms of a transaction fee
5. The communication protocol shall provide the ability to add and remove those who can add official data for distribution
6. The communication protocol shall provide the ability to distribute text/JSON data

4.3 Logistics Interface

1. The organizational view for the logistics module shall provide a view of operations pertaining to the user's credentials
2. The logistics module shall provide the ability to create tasks and assign them to other users
3. The logistics module shall provide priority and categorical settings for tasks
4. The logistics module shall provide a dashboard for viewing tasks
5. The logistics module shall enact a filter for tasks based on a user's credentials
6. The logistics module shall provide the ability to change the status of on-going events and tasks

5 Test Cases - Crowd Sourced Interface

5.1 <CSI-001, Disaster Search and Selection >

5.1.1 Test Case Description

CSI-001 will validate the functionality of allowing users to search and select a disaster within the interface.

5.1.2 Milestone/time-frame

CSI-001 approximate time-frame for completion will fall under the objectives of milestone 2

5.1.3 Test Goals

- Validate functionality of the interface to allow users to correctly search for disasters
- Validate functionality of the interface to allow users to correctly select disasters

5.1.4 Test Data

Valid search terms	Invalid search terms
Hurricane ABC	billybob
hurricane abc	398hdf8w
Flood ABC	<SELECT * FROM users WHERE name = " OR '1'='1';>
flood abc	123 $\alpha\beta\epsilon\delta\phi$ 456
Fire ABC	$\sum \cos \sinh \lim \inf$
fire abc	日本語での組版を実証する

Table 3: Sample Test Data.

5.1.5 Test Tools

- Browser - Google Chrome in developer mode
- Browser - Mozilla Firefox in developer mode
- Browser - Microsoft Edge DevTools enabled
- Text Editor - Sublime Text 3

5.1.6 Test start-up procedure

1. Ensure workstation is properly configured with needed software/hardware
2. Ensure icedrelief.net is available
3. Launch a browser in developer mode
4. Search for icedrelief.net/test in the URL search bar
5. Log in with valid tester credentials
6. Log in as a test user with the username 'user' with password 'password'

5.1.7 Test close-down procedure

1. Log out of the test user account
2. Log out of the tester account
3. Close the browser

5.1.8 Preconditions

Before beginning the test steps for the test case, ensure that the start up procedure is followed as needed and that test data has been prepared.

5.1.9 Test Steps

1. Perform test start-up procedure
2. Select "Search for Disaster" from the testing page
3. Enter an instance of invalid test data in the search field
4. Enter an instance of valid test data in the search field
5. Select the disaster based on the returned search results per the test data
6. Perform test close-down procedure
7. Repeat steps 1-6 with Google Chrome as the browser and again with Microsoft Edge
8. Perform test close-down procedure

5.1.10 Expected Results

List the corresponding expected results for each step from the test steps section above. This will allow us to validate the test against a set of results.

1. Start-up procedure functions as intended
2. Selecting "Search for Disaster" from the search field correctly produces a search box
3. Upon entering invalid test data, the search field will prompt the user that no disaster can be found with the inputted data
4. Upon entering valid test data, the search field will return a list of the disasters that meet the search criteria
5. Upon selecting the appropriate disaster, the system will notify the user with a confirmation that the disaster selected has been selected
6. Each browser correctly performs each test
7. Close-down procedure functions as intended

5.1.11 Notes/Comments

Test Case ID	Module Test	Comments
CSI001	Disaster Search and Selection	TBA

Table 4: See Gitlab issue tracker for more details.

5.2 <CSI-002, Disaster Geo-fencing >

5.2.1 Test Case Description

CSI-002 will validate the functionality of allowing users to only select disasters within their determined geo-fence as determined from the test data.

5.2.2 Milestone/time-frame

CSI-002 approximate time-frame for completion will fall under the objectives of milestone 2

5.2.3 Test Goals

- Validate functionality of the interface to allow users to only select disaster within proximity of their geo-location

5.2.4 Test Data

User's active geo-location	User's geo-fence range	Disaster to select
USA, California	100 miles	Wildfire WF1
USA, Florida	25 miles	Hurricane H1
Thailand, Bangkok	75 miles	Tsunami T1
Chile, Santiago	50 miles	Earthquake E1

Table 5: Data for Geo-Fence Testing.

5.2.5 Test Tools

- Browser - Google Chrome in developer mode
- Browser - Mozilla Firefox in developer mode
- Browser - Microsoft Edge DevTools enabled
- Text Editor - Sublime Text 3

5.2.6 Test start-up procedure

1. Ensure workstation is properly configured with needed software/hardware
2. Ensure icedrelief.net is available
3. Launch a browser in developer mode
4. Search for icedrelief.net/test in the URL search bar
5. Log in with valid tester credentials
6. Log in as a test user with the username 'user' with password 'password'

5.2.7 Test close-down procedure

1. Log out of the test user account
2. Log out of the tester account
3. Close the browser

5.2.8 Preconditions

Before beginning the test steps for the test case, ensure that the start up procedure is followed as needed and that test data has been prepared.

5.2.9 Test Steps

1. Perform test start-up procedure
2. Select "Geo-fence Test" from the interface
3. Enter the fence range into the range input field per the test data
4. Select "Begin Test" from the interface
5. Prompt the user to enable location services on their browser
6. Select the disaster the corresponds with inputted disaster per the test data
7. Perform test close-down procedure
8. Repeat steps 1-6 with Google Chrome as the browser and again with Microsoft Edge
9. Perform test close-down procedures

5.2.10 Expected Results

List the corresponding expected results for each step from the test steps section above. This will allow us to validate the test against a set of results.

1. Start-up procedure functions as intended
2. Selecting "Geo-fence Test" from the drop-down box properly displays the geo-fence test
3. Inputting the fence range per the test data properly submits the range to the geo-fence test
4. Selecting "Begin test" clears the interface and begins loading the selection table
5. Prompting the user to enable location services works and can be enabled upon user selection
6. The disaster's available on the selection table correspond to the location and range that have been inputted previously per the test data and are selectable and displayed correctly
7. Selecting and choosing the disaster per the test data verifies the user's selection and changes the DISASTER AREA text to reflect the new disaster area
8. Each browser correctly performs each test
9. Close-down procedure functions as intended

5.2.11 Notes/Comments

Test Case ID	Module Test	Comments
CSI002	Disaster Geo-fence	TBA

Table 6: See Gitlab issue tracker for more details.

6 Test Cases - Communication Protocol

6.1 <COM-001, Chain of Trust >

6.1.1 Test Case Description

COM-001 will validate the functionality of providing a chain of trust to protect the integrity of the record of data that will be changed between users

6.1.2 Milestone/time-frame

COM-001 approximate time-frame for completion will fall under the objectives of milestone 3

6.1.3 Test Goals

- Validate the functionality of data within a distributed ledger for users to download
- Validate the functionality of the hash of the data within a block
- Validate the functionality of using the hash of a previous block into the current block's hash
- Validating the functionality of linking multiple blocks together (via hashes as above) to form a chain
- Validate the functionality of invalidating data inputs that alter non-leading blocks using the hashes of the subsequent blocks
- Validate the functionality of having the newest added block to the chain distributed and checked among all requesting users

6.1.4 Test Data

Data	Hash	Previous Hash
Wildfire Alert	D2B79	CD7F1
Hurricane Alert	19BFE	D2B79
Need Rations ASAP	6E21A	19BFE
Downed Power Lines	8YG27	6E21A

Table 7: Blocks for COM-001 test.

Data	Original Hash	Change Attempt Hash
No Alert	19BFE	A7D42

Table 8: Invalid Block Changes for COM-001.

Data	Hash	Previous Hash
Radiation Hazard Alert	C3BBB	8YG27

Table 9: New Block for COM-001 distribution test.

User	Distributed Block Hash	Reported Block Hash
Gina	C3BBB	C3BBB
Kim	C3BBB	A7D42
Jeff	C3BBB	C3BBB

Table 10: Test Users for COM-001.

6.1.5 Test Tools

- IDE - Visual Studio Code (VS Code)

6.1.6 Test start-up procedure

1. Ensure workstation is properly configured with needed software/hardware
2. Launch VS Code
3. Ensure the development environment in VS Code is configured properly
4. Open the Chain of Trust solution

6.1.7 Test close-down procedure

1. Stop the Chain of Trust solution
2. Close out of VS Code

6.1.8 Preconditions

Before beginning the test steps for the test case, ensure that the start-up procedure is followed as needed and that test data has been prepared.

6.1.9 Test Steps

1. Perform test start-up procedure
2. Execute the Chain of Trust solution
3. Create a Genesis block for a new chain and seal it
4. Create new block using the test data in table 7
5. Inspect the new block's data
6. Inspect the new block's hash
7. Create an additional subsequent block using the test data in table 7
8. Inspect the subsequent block's data
9. Inspect the subsequent block's hash
10. Inspect the subsequent block's previous hash
11. Create two new blocks built off the previous block respectively (Genesis, b1, b2, b3)
12. Verify each block in the chain contains the valid previous hash of the last block
13. Attempt to change the first block's (b1) data using test data from table 8
14. Validate if the change attempt was thwarted
15. Distribute the chain to the user's listed in table 10
16. Create a new block as listed in table 9 for each respective user
17. Have the users validate each others blocks by checking the hash of the newly created blocks with each others hashes
18. Perform test close-down procedure
block b2's previous hash does not match b1 and b3's previous has matches b2's hash
19. Perform test close-down procedure

6.1.10 Expected Results

List the corresponding expected results for each step from the test steps section above. This will allow us to validate the test against a set of results.

1. Start-up procedure functions as intended
2. Close-down procedure functions as intended

6.1.11 Notes/Comments

Test Case ID	Module Test	Comments
COM-001	Chain of Trust	TBA

Table 11: See Gitlab issue tracker for more details.

7 Test Cases - Logistics Interface

7.1 <LOG-001, Organizational View >

7.1.1 Test Case Description

LOG-001 will validate the functionality of allowing organizational users to only view operations and tasks associated with that user's credentials

7.1.2 Milestone/time-frame

LOG-001 approximate time-frame for completion will fall under the objectives of milestone 2

7.1.3 Test Goals

- Validate functionality of the interface to allow the authentication of organizational users
- Validate functionality of the interface to ensure the correct display of organizational tasks and operations
- Validate functionality of the interface to ensure a user's credentials will change the viewable tasks and operations

7.1.4 Test Data

User's account name	User's organizational affiliation	User's organizational role
Bill Brown	XPL	Utility Worker
Linda Lime	XPL	Utility Lead
Jim James	CWG	Ration Distributor

Table 12: Sample Test Data for LOG-001.

7.1.5 Test Tools

- Browser - Google Chrome in developer mode
- Browser - Mozilla Firefox in developer mode
- Browser - Microsoft Edge DevTools enabled
- Text Editor - Sublime Text 3

7.1.6 Test start-up procedure

1. Ensure workstation is properly configured with needed software/hardware
2. Ensure icedrelief.net is available
3. Launch a browser in developer mode
4. Search for icedrelief.net/test in the URL search bar
5. Log in with valid tester credentials

7.1.7 Test close-down procedure

1. Log out of the test user account
2. Log out of the tester account
3. Close the browser

7.1.8 Preconditions

Before beginning the test steps for the test case, ensure that the start up procedure is followed as needed and that test data has been prepared.

7.1.9 Test Steps

1. Perform test start-up procedure
2. Select "Org-View Test" from the interface
3. Select a user from the dropdown list per the test data
4. View the display fields that show the User's name, organization, and role
5. View the operations and tasks that are displayed within the task table
6. Change a user from the dropdown list per the test data
7. Repeat steps 3-6 until all users have been cycled
8. Repeat steps 1-6 with Google Chrome as the browser and again with Microsoft Edge
9. Perform test close-down procedure

7.1.10 Expected Results

List the corresponding expected results for each step from the test steps section above. This will allow us to validate the test against a set of results.

1. Start-up procedure functions as intended
2. Selecting "Org-View Test" properly clears and displays the organizational view test interface
3. The dropdown list for the list of available users properly displays the list of user
4. Upon selecting a user, the interface properly shows the user's name, organization, and role
5. Upon selecting a user, the task table is properly redrawn to show only the tasks and operations assigned to that user
6. The task table is properly displayed and correctly shows a user's name, organization, and role as well as tasks and operations they are associated with
7. Changing users correctly redraws the task table according to the user's credentials
8. Each browser correctly performs each test
9. Close-down procedure functions as intended

7.1.11 Notes/Comments

Test Case ID	Module Test	Comments
LOG-001	Organizational View	TBA

Table 13: See Gitlab issue tracker for more details.