

Philip Susan
Sergio Ferreira Rodrigues
Raphael Duarte Gomes

Issuing Organisation Name : SHeavy

MySystem (v1.0)

Messip User Manual

- v 1.0.3 -

Based on IEEE Std 1063-2001 [\[1\]](#)

Tuesday 18th October, 2016 - 11:01

Contents

1	Product information	5
1.1	Identification	5
1.2	Copyright	5
1.3	Trademark notices	5
1.4	Restrictions	5
1.5	Warranties	5
1.6	Contractual obligations	5
1.7	Disclaimers	6
1.8	Contact	6
1.8.1	Online assistance	6
1.8.2	Telephone assistance	6
2	General Information	7
2.1	Scope	7
2.2	Purpose	7
2.3	Intended audience	7
2.4	mysystemname	7
2.4.1	Actors & Functionalities	8
2.4.2	Operating environment	9
2.5	Document structure	9
3	Usage Guide	11
3.1	Actors common procedures	11
3.1.1	Trigger the alert state	11
3.1.2	Lift the alert state	12
3.1.3	Alert Assessments	12
3.1.4	Update Map after Alert State	13
3.1.5	Set/Change Zone State	13
3.2	Medical Department procedures	13
3.2.1	Handling of infected patient	13
3.3	Common users procedures	14
3.3.1	Alert while entering a Danger Zone	14
3.4	System procedures	14
3.4.1	Finding safe route	14
4	Software operations	17
4.1	MyOperation	17
4.1.1	MyExample1	17
5	Error messages and problem resolutions	19
5.1	Error message 1	19
5.1.1	Problem identification	19
5.1.2	Probable cause	19
5.1.3	Corrective actions	19
A	Title of the appendix 1	21

CONTENTS	3
A.1 My Section	21
A.1.1 My subSection	21
References	23

List of Figures

Chapter 1

Product information

1.1 Identification

SHeavy® is a web based software developed by CrisYs Corp. whose goal is to prevent the spreading of epidemics but also to manage different organizations to handle epidemics in real time.

1.2 Copyright

A copyright is an exclusive right granted to an author of a literary, musical, audiovisual or artistic work, giving the author the sole right to reproduce and distribute that work. There are several different types of copyrights which are associated with CrisYs Corp products. These include various copyrights in CrisYs Corp's software source code, executable code, product packaging, hardware and publications.

1.3 Trademark notices

Trademarks are the distinctive names, words, logos, designs and symbols used to distinguish our products or company. Some of our recognized trademarks include CrisYs Corp®, SHeavy®. The CrisYs Corp® trademark has been filed in many countries throughout the world and registrations have been issued in CrisYs Corp's name in many countries.

1.4 Restrictions

There are no restrictions on copying or distributing the software's associated documentation but the distribution of the software is only allowed over the buyer's network.

1.5 Warranties

SHeavy warrants for an entire lifetime following delivery of the Software that the Software will perform substantially in accordance with the user manual. Additionally SHeavy warrants also that our support is available 24h/24h and 7/7 days.

1.6 Contractual obligations

Having a function ready system. Maintain the web application deployed for each actor.

1.7 Disclaimers

SHeavy only shares instructions from trustful organizations and in no event will SHeavy or its distributors and their dealers be liable to you for any damages and any losts or incidental or consequential damage.

1.8 Contact

1.8.1 Online assistance

Visit the CrisYs Corp to get quick answers to your question or refer to our online support at <https://www.crisyscorp.com/support>. You can also get help and advices by contacting us on <https://www.crisyscorp.com/chat> or by contact@crisyscorp.com.

1.8.2 Telephone assistance

When you contact our support by telephone make sure that your problem is not already described on <https://www.crisyscorp.com/support>. Otherwise choose one of our callcenters.

CSC 1: 4884224179-85

CSC 2: 4884224179-86

More callcenters are available at <https://www.crisyscorp.com/callcenter>.

Chapter 2

General Information

2.1 Scope

SHEAVY is a consistent tool to handle and manage an epidemic in an efficient way. SHEAVY can be used worldwide working together with infrastructures like governments, health organisations and hospitals. This document's goal is to provide the basic knowledge to use SHEAVY. There is no source code or programming aspect in this file.

2.2 Purpose

The user manual guide was written to show how different users can access the system and to understand its workflow.

2.3 Intended audience

EXTERIOR: All person which isn't involved with the crisis management will have a simple guideline to access to the news published in the application.

INTERIOR: All person actively helping controlling the crisis will have a overall view of the interface. They will know where to find which information. Our contact list search will be explained to let them find anybody easily, person or institution.

2.4 SHEavy

SHEAVY is an epidemic crisis management web based tool. Data from different sources will be fetched together to centralize all the known statuses and information of the crisis in some simple clicks. Its also a searchtool to find someone involved in the crisis easily and through a realtime view interface, it will be fast to know how the resources are distributed. A feature will alert the users of an important message. It keeps also track of the locations of the users to

1. Prevent them accessing an unauthorized zone
2. Guide them with GPS to secure zones.

2.4.1 Actors & Functionalities

SHeavy has different functionalities for several Actors. Here is an overview.

2.4.1.1 Common Users

A common user is defined as an end user who uses SHeavy only in order to collect information about the possible epidemic and uses the given information to avoid the infection.

- Common users are the principle target of SHeavy. They are going to use the application in order to reach information about a possible or several epidemic and follow the instructions given by SHeavy. Common users will get warnings if accessing an infected zone by gps tracking. They will have paths to follow to access and find the safe zones.

2.5 Creator procedures

The Creator only has one main function which is to set in place the system to handle an epidemical crisis. He has to make sure every actor can access it and keep it accessible as well as maintained and updated.

- Create and set up the system.

2.5.0.1 Chief

The chief is a designed person to be responsible of a action group in a specific domain e.g. Medic or Military.

- Has to inform whenever he's affected to a mission, where his team location is.

2.5.0.2 Coordinators

The Coordinator is an intermediate between all the Actors and the System. He has several main functions such as to keep the system functional and updated.

- Maintain the system operational: The Coordinator's main task is to keep the system operational.
- If necessary he will perform some improvements and bugs corrections on the system and also keep the system maintained and updated.

2.5.0.3 Medical Departement

The Medical Departement is subdivided in two groups. The laboratory and the Doctors. They have the duty to recognize an epidemic and send the alert statement that enter the population in crisis. They will work together to stop the epidemic.

- Gives the alert of the beginning of an epidemic.

2.5.0.4 Laboratory

The Laboratory is a medical group which goal is to find a cure to the virus. In constant collaboration with different medical chiefs they will get relevant information of the evolution of the epidemic in realtime. They also get the blood samples and reports from the doctors.

- Analyzing the blood samples to isolate the virus and demantle its structure to provide an antivirus.
- Gives a guideline to the application as a news so that all the people knows how to protect itself from become infected.

2.5.0.5 Doctors

Doctors are in charge of providing the antidote to the patients. They will be on site to care of the infected people.

- Write Reports that will be sent to the laboratory.
- Take care of the ill people the time that an antivirus is found.
- Test antiviruses and report if it's effective or not.

2.5.0.6 Government Departement

The Government Departement is divided into two divisions. Both Central Headquarter and Military Departement are working together in order to create an harmonious teamwork flow with the other authorities and finally achieve a common aim.

2.5.0.7 Central Headquarter

The Central Headquarter is the intermediate between the internal authorities and external governments. They are the main actors who allow SHeavy to work world wide.

- Create the communication with all Government world wide in order to merge all authorities together. This merge consists to unify medical departements in order to find a cure or to stop the epidemic rapidly.
- Manage available ressources in order to create intervention groups.

2.5.0.8 Military Departement

The Military Departement are responsible for the security of all zones and also to reduce the panic for organisational and colaborational purposes. They represent the strength of the Government.

- They are responsible for the security of all zones.
- In order to avoid riot they have to make sure that everyone are colaborative.
- Execute checks in several aeroportos in order to avoid the spreading out of the epidemic.

2.5.1 Operating environment

SHEAVY is a webbased application. It has a server and a client side. The server needs to be powerfull to handle the mass of information. Clients need to have a good internet access and a basic system requirement to

access fast their pages.

2.5.1.1 System requirements: Client

Phone Application	
<i>Android</i>	
OS	Android
Hardware	ARM 5 Dual Core
<i>iOS</i>	
Operating System	iOS 8
Hardware	iPhone 5

2.5.1.2 System requirements: Server

Server Application	
<i>Linux x86/x64</i>	
Kernel	4.8.1
Hard disk space	10 TB
Processeur	Intel Xeon E7-8893
RAM	32 GB

2.6 Document structure

Information on how this document is organised and it is expected to be used. Recommendations on which members of the audience should consult which sections of the document, and explanations about the used notation (i.e. description of formats and conventions) must also be provided.

Chapter 3

Usage Guide

This section is aimed at describing the general use of the software. Such information is grouped by the different kinds of actors. Such actors are expected to use the software to perform some processes or workflows (called here procedures) using the concerned software (**including installation procedures**).

The description of the processes should be organised to facilitate learning by presenting simpler, more common, or initial processes before more complex, less utilised, or subsequent processes.

Common procedures should be presented once to avoid redundancy when they are used in more complex procedures.

Each process has to be documented using the following use-case textual description template [2] **BUT its content must be as low level as possible with actual values:**

Use Case: ProcessMissionOne

Scope: Crisis Management System (*CMS*)

Primary Actor: Coordinator John

Secondary Actor: FirstAidWorker Bob,
ExternalResourceSystem (*ERS*)

Intention: The intention of the Coordinator is to process mission with ID equal to 1.

Level: Sub-functional level

Main Success Scenario :

1. *John* instructs the *CMS* to process a specific mission.
2. *CMS* selects the internal worker *Bob* to execute the mission.
3. *CMS* instructs '*Bob* to behave as *FAW*.
4. *Bob* informs to the *CMS* of his arrival.
5. *Bob* executes the mission.
6. *Bob* informs to the *CMS* the mission outcome.

Extensions :

- 2.a None internal worker can execute the mission.
 - 2.a.1 *CMS* requests an external resource to *ERS*.
 - 2.a.2 *ERS* informs *CMS* that the request can be processed.
Use case continues at step 3.

Remark : Graphical User Interfaces (GUIs): include GUIs screenshots to show the different stages of the process while its is performed by the actor.

3.1 Actors common procedures

Common procedures to several actors are grouped in this section to avoid redundancy.

3.1.1 Login

Use Case: LoginAttempt

Scope: Crisis Management System (*CMS*)

Primary Actor: Medical Department, Government

Secondary Actor: None

Intention: Log in the user to fit in his function role.

Level: Sub-functional level

Main Success Scenario :

1. A user need to login with his credentials and password to access the professional view of the application//
-

3.1.2 Send a Requirment Notice

Use Case: SendNotice

Scope: Crisis Management System (*CMS*)

Primary Actor: Medical Department, Government

Secondary Actor: None

Intention: Inform specific Departement a need with its location.

Level: Sub-functional level

Main Success Scenario :

1. A Professional user can sent a requirment query to a specific user group f.e. Medics with a comment to explain his needs and the respectiv groups will be informed.
-

3.1.3 Find a Person

Use Case: FindPerson

Scope: Crisis Management System (*CMS*)

Primary Actor: Medical Department, Military Department, Government

Secondary Actor: None

Intention: Find easily a involved person to call

Level: Sub-functional level

Main Success Scenario :

1. A Professional user can find people by name, by availablity or by his function and call them.
-

3.1.4 Trigger the alert state

Use Case: TriggerAlertState

Scope: Crisis Management System (*CMS*)

Primary Actor: Medical Department, Government

Secondary Actor: None

Intention: According to the diagnostic or decision of one of the primary actors, an alert will will send a notification to SHeavy about a possible epidemic.

Level: Subfunctional level

Main Success Scenario :

1. Medical Department finds out that an epidemic is possible. They will immediately send a confirmed notification to

SHeavy.

2. Military Department finds out that an epidemic is possible. They will immediately send a confirmed notification to SHeavy.

3. The Government finds out that an epidemic is possible. They will immediately send a confirmed notification to SHeavy.

3.1.5 Lift the alert state

Use Case: LiftAlertState

Scope: Crisis Management System (*CMS*)

Primary Actor: Medical Department, Government

Secondary Actor: None

Intention: According to the diagnostic or decision of one of the primary actors, an alert will send a notification to SHeavy about a possible epidemic.

Level: Subfunctional level

Main Success Scenario :

1. Medical Department notices that the epidemic is over or that the diagnostic was not right. They will immediately send a notification to SHeavy to lift the alert.
2. Military Department finds out that an epidemic is over or that their information was not right. They will immediately send a notification to SHeavy to lift the alert.
3. The Government finds out that an epidemic is over or that their information was not right. They will immediately send a notification to SHeavy to lift the alert.
4. The system will automatically lift the pre-alert if the alert notification was not confirmed. For example due to a misclick or malfunction of the system.

3.1.6 Alert Assesements

Use Case: AlertAssesements

Scope: Crisis Management System (*CMS*)

Primary Actor: Medical Department, Government

Secondary Actor: None

Intention: After the recognition of an possible epidemic, the concerned primary, which is going to trigger the alert, will rate the epidemic in a color-scale (orange or red).

Level: Analyse level

Main Success Scenario :

1. If there are some cases of an epidemic, the concerned primary actor will trigger the orange level alert.
2. If the epidemic is already spread out or infected person percentage is greater than 1/4 of the country's population then the concerned primary actor will trigger the red level alert.
3. The green level alert is triggered per default in other words there's no epidemic.

3.1.7 Update Map after Alert State

Use Case: AlertAssesements

Scope: Crisis Management System (*CMS*)

Primary Actor: Medical Department, Government

Secondary Actor: None

Intention: Update the map and GPS accordingly to the confirmation of a change of zones

Level: Sub-functional level

Main Success Scenario :

1. The Medical Department and/or the Military Department uses the TriggerAlertState or LiftAlertState and the Government confirms the change of state.
2. SHeavy receives that the state of a city or region is changed to a zone type.
3. SHeavy updates the map and GPS accordingly to the alert message.
4. SHeavy sent a message to the Medical Department, Military Department and Government that the state has changed.
5. SHeavy sent an alert message to the Common User in order to inform this change.

Extensions:

- 3.a The state of a city or region is set or changed to Safe, Unsafe or Danger.
- 3.a.1 After the TriggerAlertState or LiftAlertState, SHeavy receives the state.
- 3.a.2 SHeavy uses that information to set new zones and update the map and GPS.

3.1.8 Set/Change Zone State

Use Case: SetZoneState

Scope: Crisis Management System (*CMS*)

Primary Actor: Medical Department, Government

Secondary Actor: None

Intention: Set or change the state of a zone after the confirmation of the Government.

Level: Sub-functional level

Main Success Scenario :

1. After the TriggerAlertState or LiftAlertState, the Government confirms the respective state of a zone: Safe, Unsafe, Danger.
2. The state of zone is also received by SHeavy.
3. SHeavy sets the state to a new zone or changes the state of a current zone to the respective new state.
4. SHeavy updates the state of that zone in the database that updates any corresponding functions of the application.
5. SHeavy notifies the Medical Department and the Military Department that a new region has the corresponding state or that an existing region changed his state to the new state.

3.2 Creator procedures

3.2.1 System Creation

Use Case: SystemCreation

Scope: Crisis Management System (*CMS*)

Primary Actor: Creator

Secondary Actor: None

Intention: The intention of the Creator is to set in place the system made to handle a epidemical crisis.

Level: Subfunctional level

Main Success Scenario :

1. The system is set up.//

3.3 Medical Department procedures

3.3.1 *Handling of infected patient*

Use Case: HandlingOfInfectedPatient

Scope: Crisis Management System (*CMS*)

Primary Actor: Doctor

Secondary Actor: Labotatory

Intention: The Medical Department intends to update the application with the newest data about infected people, no matter what sickness they have. In case of a known epidemic infection, keep an updated record of the growth of the epidemic.

Level: Subfunctional level

Main Success Scenario :

1. The Medical Department performs a medical check on a patient.
 2. The Medical Department sends data about that patient to the application.
 3. If the patient shows signs of an infection that is an already known epidemic infection, his data and location will be automatically updated on the data center part dedicated to this matter and also notify on SHeavy.
 4. If the patient shows signs of an unknown infection the medical Department will proceed on taking a sample of blood for further testing.
 5. The Medical sends the sample to the nearest laboratory in order to find a cure.
-

3.4 Common users procedures

3.4.1 *Alert while entering a Danger Zone*

Use Case: DangerAlert

Scope: Crisis Management System (*CMS*)

Primary Actor: Common user

Secondary Actor: None

Intention: Warn a user that he is about to enter a Danger Zone before entering the Danger Zone.

Level: Sub-functional level

Main Success Scenario :

1. The common user is going somewhere in or near a Danger Zone to do something.
 2. The common user is located by SHeavy and John is near a Danger Zone and is going in its direction.
 3. SHeavy immediately send a warning to John's Phone, indicating that John is entering a Danger Zone.
 4. The common user sees the warning and he takes a way around the Danger Zone.
-

3.5 Gouvernement Procedure

3.5.1 *Handling Ressources*

Use Case: HandlingRessources

Scope: Crisis Management System (*CMS*)

Primary Actor: Gouvernement

Secondary Actor: None

Intention: Any actor can live track where each Team of each Departement is working. It shows if a team is available. Ressources Requirments can be sent.

Level: Subfunctional level

Main Success Scenario :

1. The gouvernement has a detailed list where all the teams are distributed and can in function apply a stratigy to minimize the expansion of the epidemic
 2. The gourvenment sees which teams are available and affect them to a mission changing in the same time the team status to unavailable.
-

3.6 System procedures

3.6.1 *Finding safe route*

Use Case: FindSafe

Scope: Crisis Management System (*CMS*)

Primary Actor: System

Secondary Actor: Common User

Intention: Suggest the user the safest way to get to a safe place (e.g. hospital)

Level: Sub-functional level

Main Success Scenario :

1. The common user is going to a safe place and uses the GPS of SHeavy.
2. SHeavy calculates the fastest way to the indicated place.
3. SHeavy shows the safest way from the GPS to John.

Extensions:

- 2.a SHeavy calculates the fastest way avoiding Unsafe and Danger Zones.
 - 2.a.1 SHeavy ignores any way through an Unsafe/Danger Zone.
 - 2.a.2 SHeavy calculates the fastest way from the non-ignored ways.
-

Chapter 4

Software operations

Explain each allowed software operations (i.e. an atomic unit of treatment, a service, a functionality) including a brief description of the operation, required parameters, optional parameters, default options, required steps to trigger the operation, assumptions upon request of the operation and expected results of executing such operation. Describe how to recognise that the operation has successfully been executed or abnormally terminated. The template given below (i.e. section 4.1 has to be used).

Group the operations devoted to the needs of specific actors. Common operations to several actors may be grouped and presented once to avoid redundancy.

4.1 MyOperation

The system operator creates and adds a new crisis to the system after being informed by a third party (citizen, organization) and selects a crisis handler for the crisis.

Parameters: Reporter Personal Information, Crisis Information, Crisis Handler

Precondition: The system operator is logged in and has received information from a reporter.

Post-condition: A new crisis has been added to the system and the new crisis has been assigned to a crisis handler, the Handler has received an automatic notification from the system.

Output messages: The selected Crisis Handler will be notified automatically once the crisis has been created.

Triggering:

1. From within the crisis management window fill out the required entries related to the personal information of the reporter such as name and phone number.
2. Fill out the entries related to the crisis type, impacted area, priority, description, GPS coordinates, address and finally choose a Crisis Handler from the combo box.
3. Click on the “Submit” button in and add the entry to the database.

4.1.1 MyExample1

Examples should illustrate the use of **complex operations**.

Each example must show how the actor uses the software operation under description to achieve (at least one of) its expected outcome.

It might be required to include GUI screenshots to illustrate the example.

Chapter 5

Error messages and problem resolutions

All known problems in using the software should be listed and explained in details using the structure presented below.

Contact information for reporting any problems (either with the software or this document) should be clearly indicated

5.1 Error message 1

5.1.1 Problem identification

A description explaining the meaning of the faced problem.

5.1.2 Probable cause

A description explaining the reasons why such a problem has been raised.

5.1.3 Corrective actions

Describe the required steps the actor should take to recover from such situation.

Appendix A

Title of the appendix 1

Here you write the context of the appendix, structuring such content in sections, sub-sections and sub-sub-sections, if needed.

An example of appendix is the flat presentation of all the graphical user interface screens. Each screen can be presented (identification symbol and description) and screens transition graph can be given.

A.1 My Section

Description of the section.

A.1.1 My subSection

A.1.1.1 My subSubSection

References

1. IEEE: IEEE Standard for Software User Documentation. IEEE Std 1063-2001 (Dec 2001) 1–24
2. Armour, F., Miller, G.: Advanced Use Case Modeling: Software Systems. Addison-Wesley (2001)