1. Usecase: Write Update
2. Actors: Emergency Organization(s) users associated to the event
3. Stakeholders:

* Updater-Emergency Organization(s) user- want to write update
* Emergency Organization(s) users associated to the event – want to read updates

1. Preconditions:

* Event exists
* The updater is associated to the event
* The readers are associated to the event

1. Post Conditions:

* A new update created
* The update assigned to an event
* If needed the update assigned to other updated of the event

1. Main Success Scenario:
   1. Updater chooses an event he is associated with
   2. The updater tries to create an update for the event
   3. System checks if updater has writing permission on the event
   4. The updater has writing permissions
   5. The updater creates an update (date, event, user id, the update details)
   6. The update associated to last update on this event
   7. All users associated to the event can see the new update
2. Alternative Flows:
3. The updater has only reading permissions:
4. System signals error
5. The update enters invalid date:
6. System signals error
7. User can try again until no error
8. First update on the event:
9. The update associated as the first update of the event
10. Open Issues:
11. Usecase: Edit Update
12. Actors: Emergency Organization(s) users associated to the event
13. Stakeholders:

* Editor-Emergency Organization(s) user- want to edit his update
* Emergency Organization(s) users associated to the event – want to read updates

1. Preconditions:

* Event exists
* The readers are associated to the event
* Update exists
* The update is associated to the event
* The editor is associated to the event

1. Post Conditions:

* The update is edited
* The last version assigned to the edit

1. Main Success Scenario:
   1. editor chooses an event he is associated with
   2. the editor chooses one of his updates to the event
   3. The editor edits his event
   4. The original update is associated to the edit
   5. All users associated to the event can see the edit
   6. All users associated to the event can see the original update
2. Alternative Flows:
3. Open Issues:
   1. Is the lest edit will be displayed first or in accordance to the update it is associated to place?
   2. How many versions of the update is saved?
4. Usecase: Create Category
5. Actors: EOC representative(s) admin, EOC representative(s) users, Emergency Organization(s) users
6. Stakeholders:

* EOC representative(s) admin – want to create new categories
* EOC representative(s) users – want a verity of categories to choose from while creating an event
* All users of the system - want a verity of categories to search event accurately

1. Preconditions:
2. Post Conditions:

* Category created

1. Main Success Scenario:
   1. EOC representative(s) admin tries creating new category (name, description)
   2. System checks There is no duplication
   3. New category created
2. Alternative Flows:

a\*. if the system is loading and there is no categories the EOC representative(s) admin has to add an category

b. there is a duplication:

1. System signals error

2. user can try again

1. Open Issues:
   1. Are there sub-categories?
2. Usecase: Rank Colleagues
3. Actors: Emergency Organization(s) users, EOC representative(s) admin
4. Stakeholders:

* EOC representative(s) admin – want to get ranking of the average summary of participants in the event
* Emergency Organization(s) admin – want to get ranking of the average summary of participants in the event
* Emergency Organization(s) users – want to give and receive feedback on their work accurately

1. Preconditions:

* Event exists
* The Emergency Organization(s) users which fill feedback are associated to the event
* The Emergency Organization(s) users who have feedback on them are associated to the event
* Event status changed to “Treatment is over”

1. Post Conditions:

* Report created
* Report sanded to the EOC representative(s) admin

1. Main Success Scenario:
   1. System sends to the Emergency Organization(s) users that are associated to the event a report to fill
   2. All the users who received the report scales event participants (1-5)
   3. The data is saved
   4. After all participants completed filling the report the system summarizes the data to one report
   5. The report is sent to EOC representative(s) admin
2. Alternative Flows:
3. Open Issues:
   1. Does the user have to rate them all the participants?
   2. Can the users add comments to the numeric rank?
4. Usecase: Change Rank
5. Actors: Emergency Organization admin, EOC Organization user
6. Stakeholders:

* Emergency Organization(s) admins – want to change rank of user

1. Emergency Organization(s) users – want to fast and fair process
2. Preconditions:

* case 1: rank down
  + user has 3 warnings
* case2: rank up
  + user got average rank of five
  + user doesn’t have warnings
* Event status changed to “Treatment is over”

1. Post Conditions:

* Rank changed

1. Main Success Scenario:
   1. Admin changes the rank of the user
2. Alternative Flows:

a.1 if rank is lowered all of user’s warnings are deleted

a.2 if users rank was 1 and rank is lowered his account becomes “disabled”

1. Open Issues:
   1. What happens if the user is at the highest rank and the admin wants to raise it?
2. Usecase: Create Event
3. Actors: EOC (Emergency Operation Center) representative, Emergency Organization(s) representative(s), Emergency Organization(s) users.
4. Stakeholders:

* EOC Rep. – Wants to assign event to the relevant organizations.
* Emergency Organization Rep. – Want to get as much details about the event as possible.
* Caller/Event Identifier and involved citizens – Want a fast and relevant response.

1. Preconditions:

* EOC Rep. has received a report regarding an event.
* There is at least one category.

1. Post Conditions:

* A new event is created
* Event is associated with the relevant category (or categories).
* Event is assigned to the relevant emergency organization(s)
* Event have an assigned user in each organization.
* Relevant users are informed of the new event.

1. Main Success Scenario:
   1. EOC Rep. opens a new event, and fills in details (Address, main issue, etc.).
   2. The system offers the EOC Rep. a list of possible categories to associate the event to.
   3. EOC Rep. picks a category (or categories).
   4. EOC Rep. selects users from relevant emergency organizations.
   5. The system notifies the respective users of the newly added event.
   6. Other users in the organizations can see event details.
2. Alternative Flows:

a1. Missing details.

1. EOC Rep. will not be able to move on to the next step until all fields are filled.

c1. EOC Rep. did not choose any category.

1. EOC Rep. will not be able to move on to the next step until at least one category is selected.

d1. EOC Rep. did not choose any users.

1. EOC Rep. will not be able to complete the event creation without choosing at least one user.
2. Open Issues:

* How is the user to which the event is assigned to determined?

1. Usecase: Send Report
2. Actors: Emergency user Sender, Emergency user Receiver.
3. Stakeholders:

* Emergency user Sender – Wants to assign the event to a relevant organization.
* Emergency user Receiver – Needs to be informed of relevant event details.
* Caller/Event Identifier and involved citizens – Want a fast and relevant response.

1. Preconditions:

* There is an active event.
* Sender is part of organization A which is informed of event.
* Receiver is part of some other organization B, which is not yet informed of event.

1. Post Conditions:

* The event is assigned to organization B.
* Receiver is in charge of the event for organization B.
* Users from organization B are informed of the event.

1. Main Success Scenario:
   1. The Sender clicks the Send Report button.
   2. The system displays the current events in organization A.
   3. The Sender picks an event and a Receiver user.
   4. Receiver is notified of the event details.
   5. Receiver oversees the event for organization B.
2. Alternative Flows:

c1. Receiver is part of the same organization is Sender.

1. An informative error message pops up for Sender.

c2. Receiver is part of an organization that is already informed of the event.

1. An informative error message pops up for Sender.
2. Usecase: Complain
3. Actors: Complaining user, Complained user, emergency organization admin.
4. Stakeholders:

* Complaining user – wants the complaint to be processed as fast as possible.
* Complained user – Wants the complaint to be processed in an adequate manner.
* Emergency organization admin – Wants complaints system to be minimal in bureaucracy.
* Emergency organization – wants disputes to be settled quickly and fairly, to ensure good working terms.

1. Preconditions:

* Two users, A and B, in the same emergency organization.

1. Post Conditions:

* A complaint has been created by A against B.
* The emergency organization admin has ruled for or against the complaint.
* Both parties will be informed of the admin's decision.

1. Main Success Scenario:
   1. User A presses the Complain button.
   2. The system displays a list of users in the same organization as A.
   3. User A chooses user B as his subject of complaint, and fills the form detailing the essence of his complaint.
   4. The emergency organization admin reviews the complaint details and makes a ruling of his own discretion.
   5. A notification regarding the admin's decision is issued to both users.
2. Alternative Flows:

b1. The user chose himself.

1. The system will pop up a warning message.

d2. In case the admin chose to accept the complaint:

1. The Change rank UC will commence.

If the admin has decided against the complaint, no further action is needed.