

Software Engineering

343.309

2020W

Third Semester

Hogea Eduard-Florin

Studienkennzahl - 033 521

November 11, 2020

Group members

1. Omar Dueñas K51849509
roqueluisd@me.com
2. Michael Lengauer K11710940
lengauer.michael@gmail.com
3. Mario Lischka K11712790
mario.lishcka@gmx.at
4. Eduard-Florin Hogea K12019919
eduard.hogea00@e-uvvt.ro
5. Lukas Wais K11816105
lukas.wais@outlook.de

Subsystems and members

Subsystem	Assigned Member
Traffic control and detection	Michael Lengauer
Traffic participants	Omar Dueñas
Control system	Lukas Wais; Mario Lischka
Road maintenance	Hogea Eduard-Florin

Conceptual class diagram

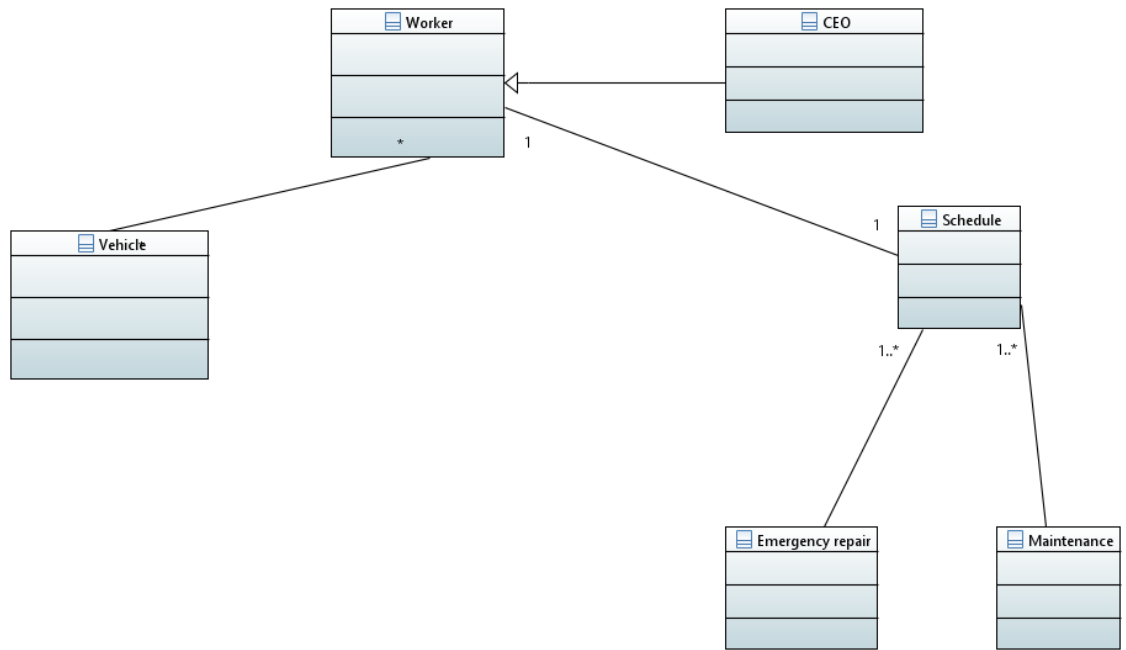


Figure 1: Conceptual class diagram

In this figure, we can see that the CEO class is a superior type of worker. The worker class is linked to vehicle (a worker can have none or more vehicles) and to Schedule (worker can have only a schedule and a schedule can have only a worker). The schedule is also associated with Emergency repair and Maintenance and can have 1 or multiple of those.

Design class diagram

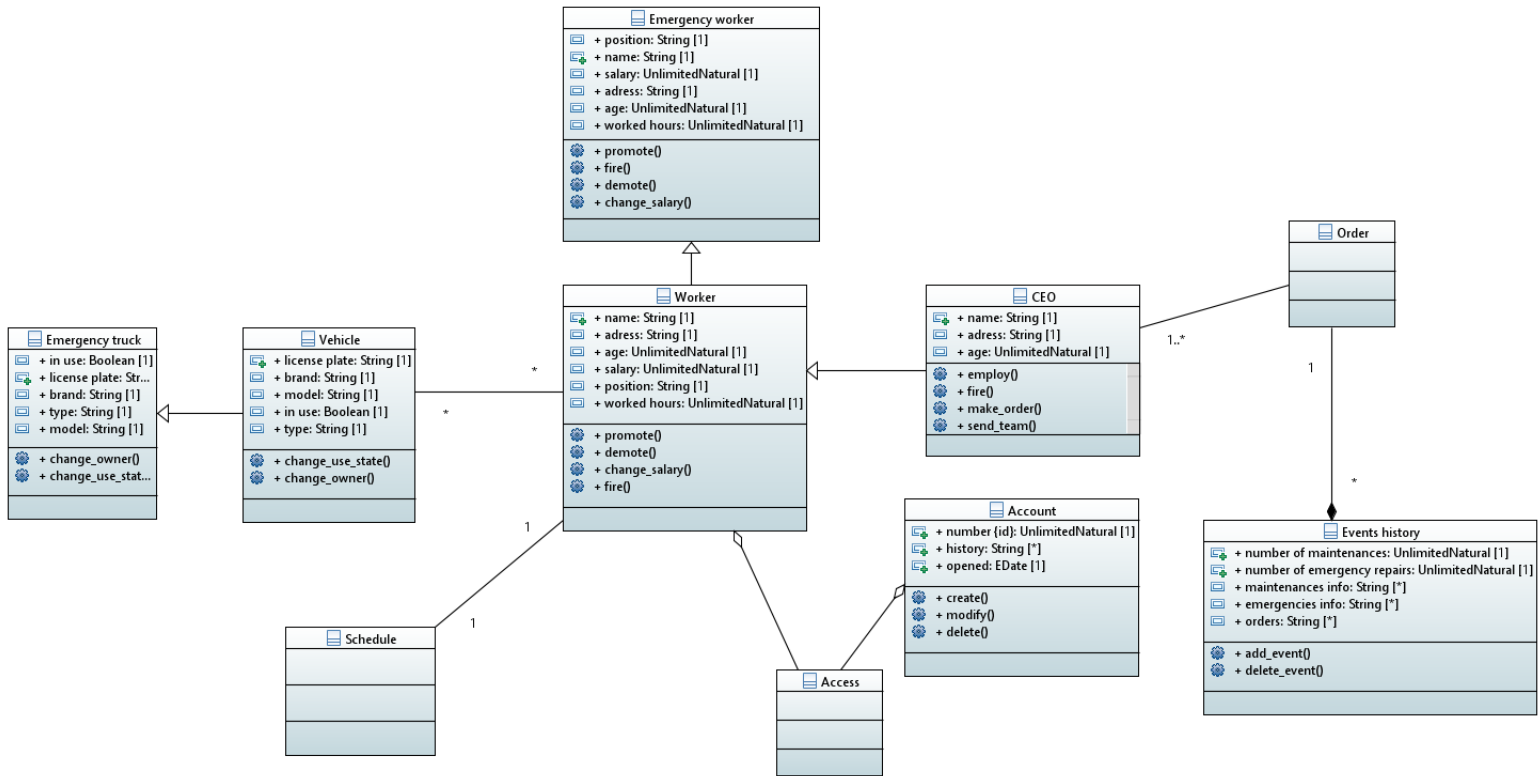


Figure 2: Design class diagram

The second figure represents a more detailed approach of the class diagram. The design class diagram shows more in depth the links and the methods that can be used in each of the classes. The relations remain the same as previous stated but also now we can see some more. For example, the `Worker` class now is linked with the `Access` class with an aggregation, which in turn is linked with the `Account` class with also an aggregation. The classes now also contain more information about the details of each one of them.

Sequence diagrams

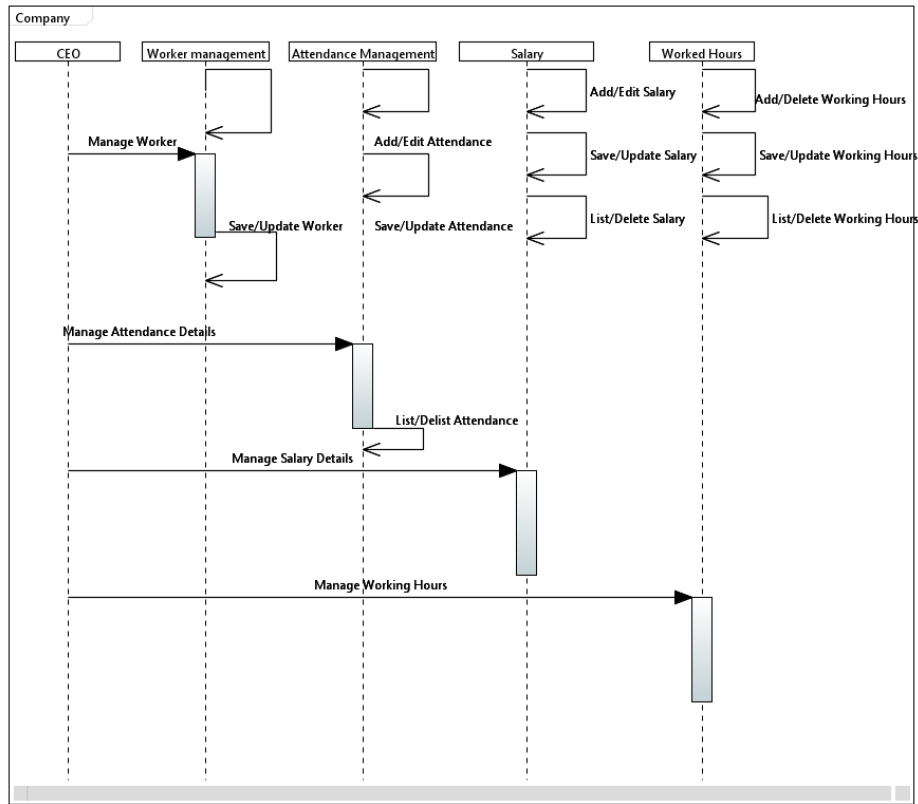


Figure 3: Sequence diagram 1

First sequence diagram shows how the company is being administrated in the eyes of the CEO and the worker. The CEO class has total control of the details of Worker class. This is shown in the sequence diagram presented, because the CEO can Manage Worker, Manage Attendance Details, Manage Salary Details and Manage Working Hours. Managing them represents actions like add, edit, save, update, list, delist.

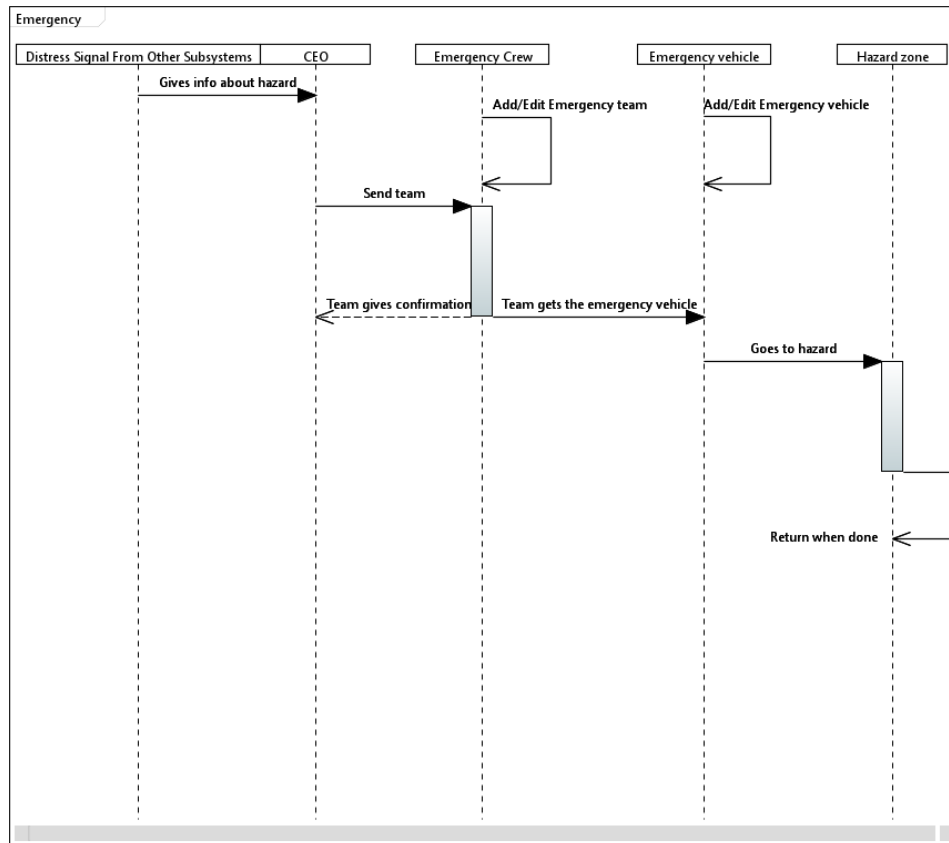


Figure 4: Sequence diagram 2

The following sequence diagram explains how an emergency situation is being dealt with. The signal is received by the CEO first, who in turn gives sends and gives information to the Emergency Crew. The team gets an emergency vehicles and goes to the accident.

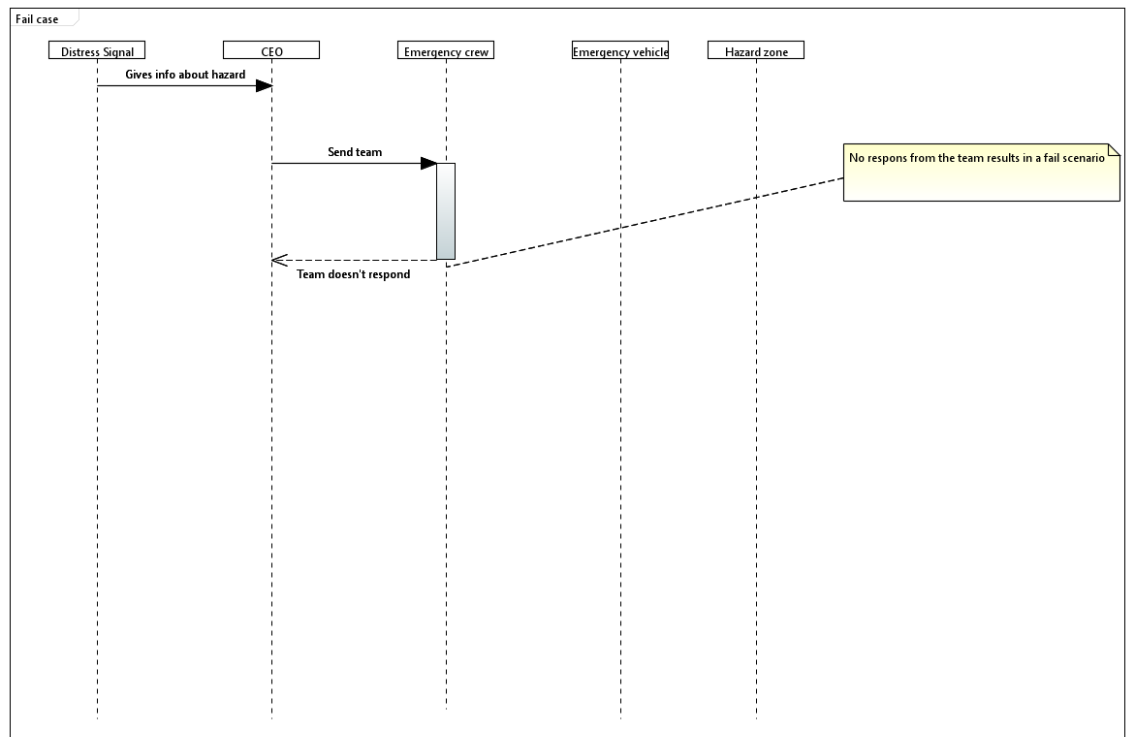
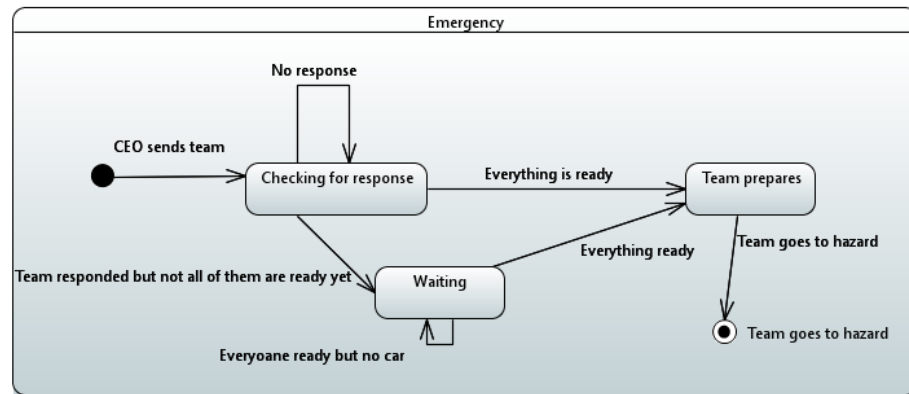


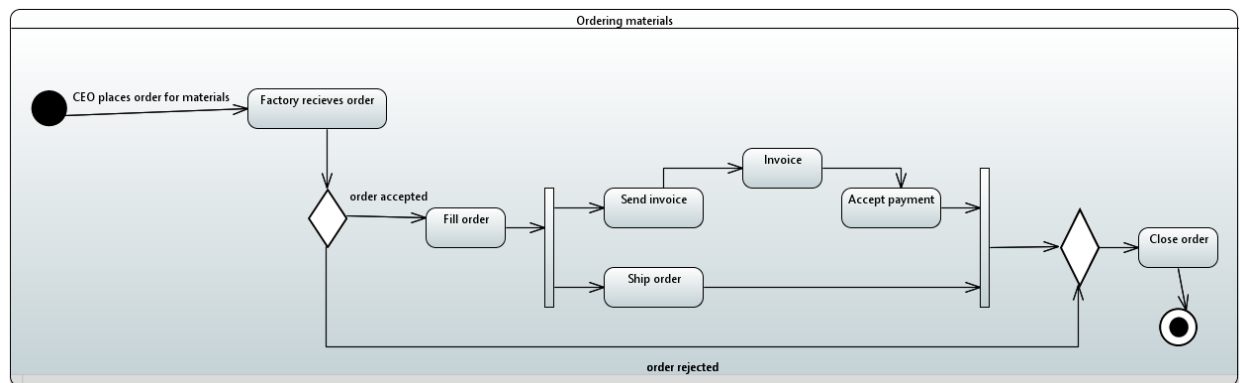
Figure 5: Sequence diagram 3

The last sequence diagram shows how a situation can be a failure. This failure case shows how if the Emergency Crew doesn't give a response to the CEO, the whole scenario from before can't happen. The absence of response makes this case become a fail case.

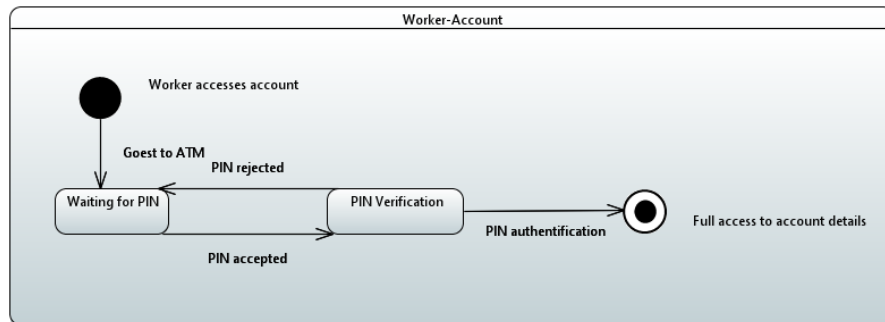
State diagrams



The Emergency diagram shown above is a state diagram representation of how an emergency situation is being dealt with. The initial point is the CEO class who using the methods available can send a team if needed. The team (Emergency crew class) can either give a response or not. This is shown as a loop, if no response is given or going to waiting or preparing state. If everything is not ready, the team waits. If all is good, the team prepares and goes.



Ordering materials state diagram depicts the state of actions that are done when the CEO class uses the method `order()`. The CEO places an order of needed materials to the factory who, can either accept or reject it. If the order is accepted, it is filled, either directly shipped after or gives an invoice.



The final diagram represents the situation of a Worker class who tries to access it's account. To get to that account, the initial point is represented by a worker who goes to an ATM. If the PIN is entered and is correct, the Worker class now has full access to it's account details. If the PIN is not accepted, the ATM will wait for the correct one to be entered.