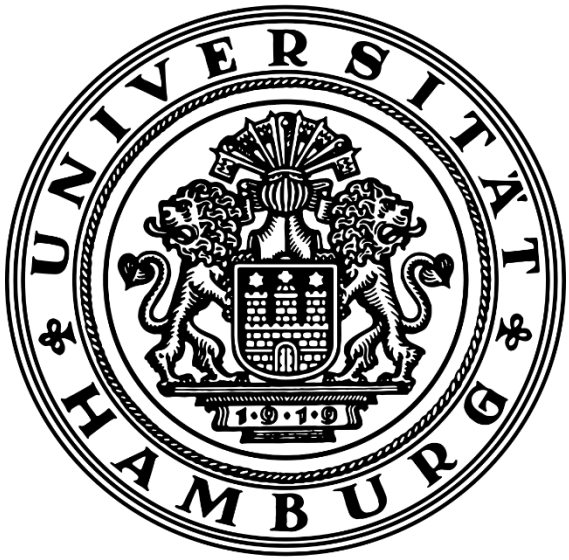
Requirements Analysis Document



Corvin Biebach

Maximilian Brosius

Fynn-Ole Menk

Anni Reinert

Noah Scheld

Arne Struck

Mudassar Zahid

9.12.2021

Table of Contents

1 Introduction

1.1 Purpose of the System

The purpose of the system is to support the users, mainly doctors and paramedics, who are working on an almost daily basis with the current medical devices for monitoring, ventilation, and defibrillation. The project was initiated by the Universitätsklinikum Hamburg-Eppendorf in cooperation with the University of Hamburg and the device manufacturer Weinmann. Combining those mentioned devices into a 3-in-1 device should be lighter to carry and gather the three interfaces and settings into one. For the M-Lab project for the winter semester 2021/2022 the project group will design a new interface and develop an alarm management with new behaviour in a visual and auditory way. The provided patient scenarios will be used to demonstrate the application.
The application and result of the M-Lab project will be used by the doctors of the UKE and project partners for evaluation and making propositions for the future device. The university will be grading the results.

1.2 Scope of the System

The scope of the system refers to a new, modern design and combination of interfaces for monitoring, ventilation, and defibrillation. In the process, the representations are going to be united and should not be separated in different views. Furthermore, it’s part of the project to develop a new alarm management that handles the united alarm behaviour in a different way to support the users. A special focus is placed on the mute behaviour. Approaches and aspects outside the medical approach are encouraged. The project will be implemented for a tablet app for Android and iOS devices. It is not considered to implement real interfaces to existing or future devices. Patient scenarios serve as the basis for demonstration and will deliver certain data and alarm sets. The setting options will refer to the medical and general use aspects and will not focus on accessibility.

1.3 Objectives and Success Criteria of the Project

The goal is to display the main parameters and visualize them in the required graphic fashion. In addition to those parameters, different alarms will provide auditory and visual feedback. Concurrent and successive alarms of different priorities will be handled by the system in a non-disruptive and supportive manner. During the usage, adjustments of parameter limits and switching fast into the defibrillation mode should be possible.

To demonstrate the alarms and visualisation of parameters, the application will offer the ability to play provided patient scenarios. The app will be usable with touch functions especially to adjust parameters and their limits.
Further goals and requirements are described in detail in section 3. Proposed System.

1.4 Definitions, Acronyms and Abbreviations

Corpuls	the current device for monitoring and defibrillation
CO2	the concentration of carbon dioxide (CO2) during exhalation
ECG	electrocardiogram
etCO2	end-tidal CO2: the partial pressure or maximal concentration of carbon dioxide (CO2) at the end of an exhaled breath
Flow	describes the flow of air in and out of a patients' lungs, inspiration are counted as positive values, expiration as negatives
Monitor	usually display, in this project as a generic term refers to monitoring
Monitoring	refers to the observing of a patient with ECG graph, heart frequency, pulse etc.
NIBP/ IBP	non-invasive blood pressure
respiratory rate	breathing frequency
SpO2	oxygen saturation of the blood
UHH	University of Hamburg
UKE	Universitätsklinikum Hamburg-Eppendorf
Weinmann	device manufacturer, the current device for ventilation

1.5 References

To get an overview of the functions, alarm systems and the values displayed by the existing devices this document refers to the provided manuals of the Corpuls C3 v2.3, the Weinmann Medumat Standard2 v4.15 and the Weinmann Meduvent Standard v2.1. In addition to that, the patient stories provided by the UKE and an overview of alarms referring to the scenarios are used to understand the real processes, situations, and conditions under which the devices are used. For the usage of Flutter, we will use packages, libraries and documentation provided by Flutter itself. For the visualisation of the live graphs for parameters like the heart frequency, we will use the Syncfusion Community License.

2 Current System

The current system consists of up to three devices: one device each for monitoring, ventilation, and defibrillation. It is possible, that the devices for monitoring and defibrillation are already combined into one, e.g., Corpuls. The mentioned devices are used by persons with a medical background, mainly doctors or paramedics, during an ambulance mission or in hospitals and operating rooms.

In a situation stressful and/or critical for the patient, alarms often are mixed up which can result in a sensory overload for the user. The usage of a mute button is the rule for either single alarms or in general. Especially the general mute for two minutes can lead to critical situations due to the missing alarms about drastically changing values while the user is busy or unable to pay attention to the monitoring devices and ventilator. This does not support the users during the usage. The different functions being separated into three devices has a side effect, that functions are distributed redundantly to the devices. Furthermore, the adjustment for alarm limits takes up to 20 steps and is very time-consuming.

3 Proposed System

3.1 Overview

The app is going to be the combined user interface of the monitor respectively defibrillator and the ventilator and therefore centralize the user's interactions. Furthermore, the alarms coming from one united device are going to be managed depending on their priorities. The interface will likely differ from known systems but still be oriented on those designs. In addition to that, the app is going to offer the ability to play known patient scenarios to demonstrate the system with real-life data.

3.2 Functional Requirements

1. After starting the application, the system shall be able to display a start screen with the ability to choose presets for adults, children, or infants. This option should be skippable if the monitoring has to be started directly.
2. After having chosen a preset, the system shall be able to enter patient data for weight and height, so certain settings, for example, the joule for shocking, can be calculated.
3. After switching the monitoring and/ or ventilation the system shall be able to display the following parameters:
    - Heart frequency/ECG (absolute value + curve)
    - CO2 (absolute value + curve)

- SpO2 (absolute value + curve)
- etCO2 (absolute value + curve)
- Flow (absolute value + curve)
- Airway pressure (absolute value + curve)
- NIBP (absolute value + history of last measurements)
- Pulse frequency (absolute value)
- O2 Mode
- Fill level of the oxygen cylinder
- Respiratory frequency

4. The system shall provide the user with the ability to add certain parameter curves, for example, the NIBD history or an absolute value of the patient's temperature.
5. The system shall provide the user with the ability to set all parameter limits of the underlying devices in one menu.
6. The system should provide the user with the ability to set parameter limits by directly interacting with the parameter.
7. The system should provide the user with the ability to mute all active alarms, displaying new alarms.
8. The system should provide the user with the ability to mute a single active alarm.
9. The system shall display alarms visually, prioritizing the most relevant ones.
10. The system should display alarms underlined with an audio component, prioritizing the most relevant ones.
11. The system shall provide the user with the ability to switch in a defibrillation mode.
12. The system shall provide the user with the ability to set defibrillation shock settings.
13. The system shall provide the user with the ability to set ventilation settings.
14. The system shall differentiate between different alarm sources (e.g., monitoring, ventilation, and the device itself) and play distinguishable alarms.

3.3 Nonfunctional Requirements

3.3.1 Usability, Consistency & Familiarity

The system needs to offer an easy and fast option for the user to switch into the defibrillation mode with a maximum of 3 steps offering a "Defibrillator" or "AED" button. In addition to that, customizing the screen for individual preferences, for example adding a history of the NIBD or a derived curve of the ECG, should be offered directly at the parameter or with an adding interface. Furthermore, the settings for alarm limits shall be reachable with a maximum of 3 steps for single, but also for all parameters.

The construction of a parameter with details, like alarm boundaries and unit, should be equal for every displayed parameter. The colours should be orientated on known schemes, like having the heart frequency displayed in green and the NIBD in red.

3.3.2 Reliability

The system shall not fail during usage. The reliability does not target external connections, like a medical sensor. During interactions with the tablet app, the app should not freeze due to performing issues. The visualized parameters and their curves should match the values.

3.3.3 Performance

The user adjusts parameters and their limits, so a real-time interaction needs to be guaranteed. Furthermore, the visualization of changes and graphic presentation, for example, curves and new alarm limits, needs to be displayed without delay. Graphs and curves should be visualized in a fluid and continuous manner, displaying new values.

3.3.4 Supportability

The system should offer the ability to add new alarms (and values) to the current version of the app's backend as well as to the display options since the first version will probably not cover each possible alarm. Furthermore, a new style theme for light mode or a colour blindness mode should be added with no problems.

3.3.5 Implementation Requirements

There are no implementation requirements or restrictions for this project. For the development, we will follow the suggestion of the M-Lab Team and use Flutter as a software environment.

3.3.6 Interface, Efficiency & Flexibility Requirements

The position of parameters and their curves should be fixed and not changed. The size is not affected by that requirement. The user should reach certain settings, such as adjusting a single parameter limit, with a maximum of 3 steps. The goal is to reduce steps for settings in general to a minimum of 5.

The system should be designed for a tablet with a resolution of 1920x1200 pixels in a 16:10 format with a 10 Inch display. Limited by the hardware selection with mentioned values we will design the interface responsive to changing display sizes.

3.3.7 Packaging Requirements

The finished project will be handed over as a functioning Flutter project for iOS and Android devices with documentation like a style guide.

3.3.8 Legal Requirements

While the current system needs to document the doctor's actions for the health insurance and the ECG during the emergency service, our tablet app will not fulfil these or further known legal requirements.

3.4 System Models

3.4.1 Scenarios

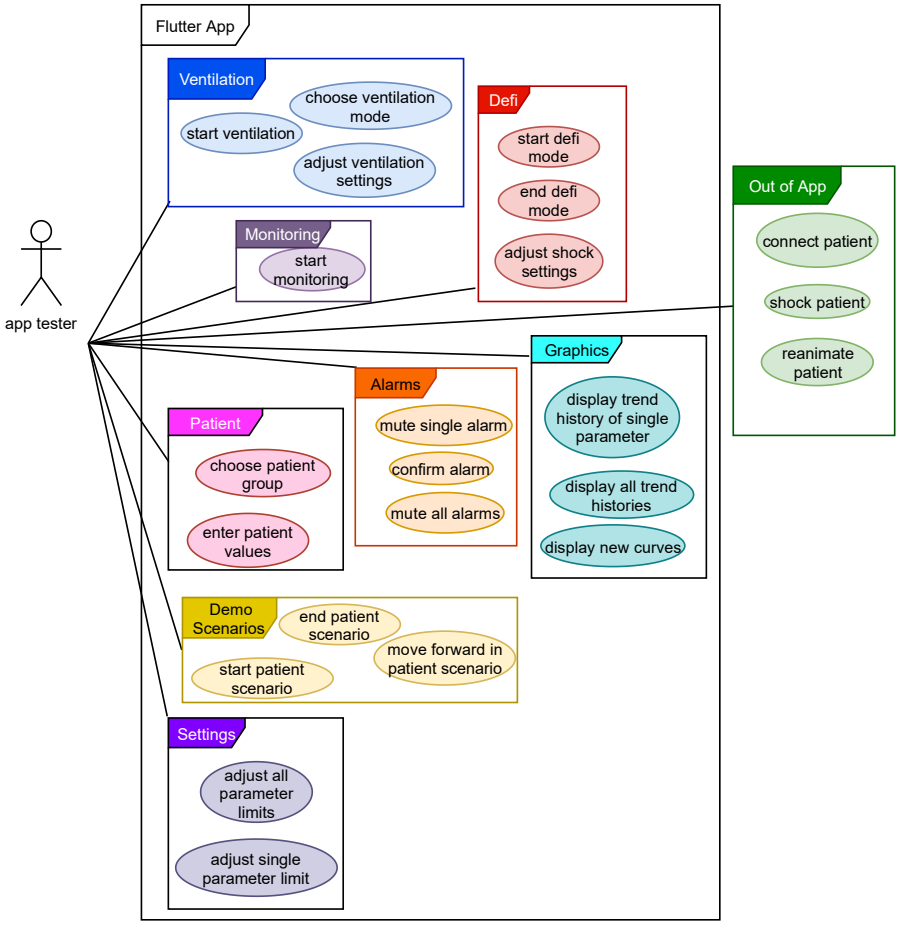
To get to know how users interact with the current system and to understand the situations, the given "patient stories" describe up to 6 different real-life scenarios. Those provided a base for the first customer meetings and to derive main use cases.

The purpose of the stories is to retrace a doctor's interactions while monitoring, ventilating, or defibrillating a patient under real circumstances. For the UKE the main focus is put on the Corpuls C3 as well as the Weinmann Meduvent Standard2 v4.1 and Weinmann Meduvent Standard v2.1. The stories build up from an arriving scenario only using monitoring to a more critical situation needing a ventilator. The last steps of the scenarios describe the serious case of reanimating a patient.

For the application, the certain parameter values, and settings, that are adjusted by the doctors or paramedics during the scenarios, are going to be the basis for the demonstration to trigger different alarms. The scenarios cover different sets of alarms and handle all priorities (note, yellow alarm, red alarm). Furthermore, concurrent alarms that would occur on the single devices are also part of the scenarios and can be integrated into the alarm management, which should handle those.

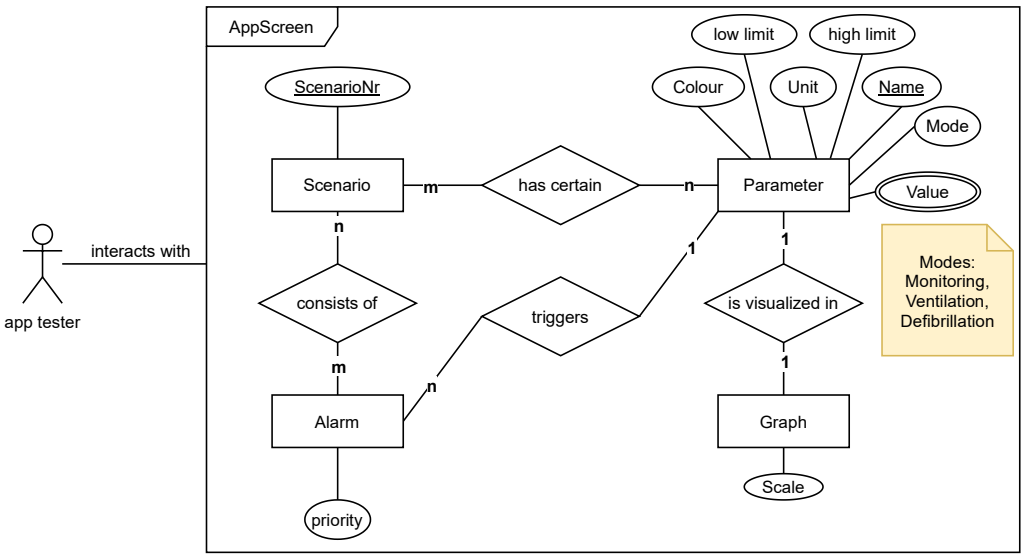
3.4.2 Use Case Model

Three main use cases were detected for the use case model: Monitoring, ventilation, and defibrillation. To organize the users' interaction with a 3-in-1 device, the detailed use cases are organized in subsystems. Some interactions belong to the main use cases alone. Actions that are connected with at least two of the subsystems are gathered in action groups. There are also some "out of app" activities that are not relevant to the project. In addition to the general use of an interface for the future 3-in-1 device, the project's app needs to offer the ability to start the known demo scenarios. The app will be tested by persons with a medical background, meaning doctors of the UKE or paramedics or stakeholder of the research project as well as supervisors from the university, responsible for the M-Lab project.



3.4.3 Object Model

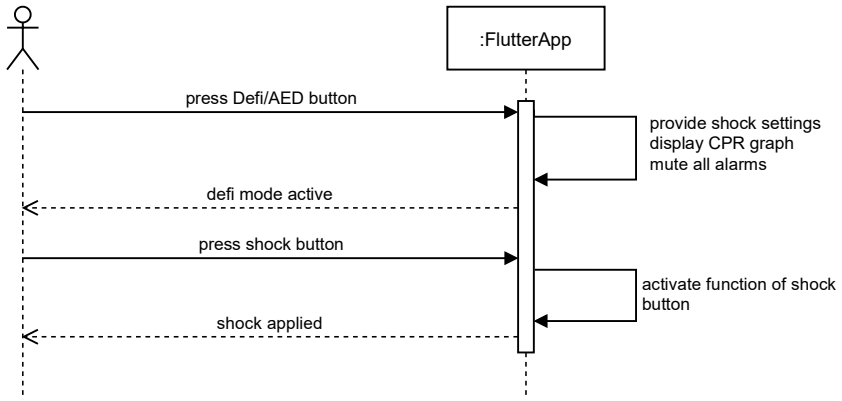
The application is going to display *parameters* with their *values* and their *graphs* if required. The *app user* (doctors/ paramedics) wants to interact with parameters directly to adjust settings like the *alarm limit.* Certain values belong to the different modes: for example, the heart frequency belongs to *monitoring* and the respiratory rate to *ventilation*. *Scenarios* consist of predefined parameters with certain values and settings. Every scenario is made of predefined sets of *alarms* in a predefined order.



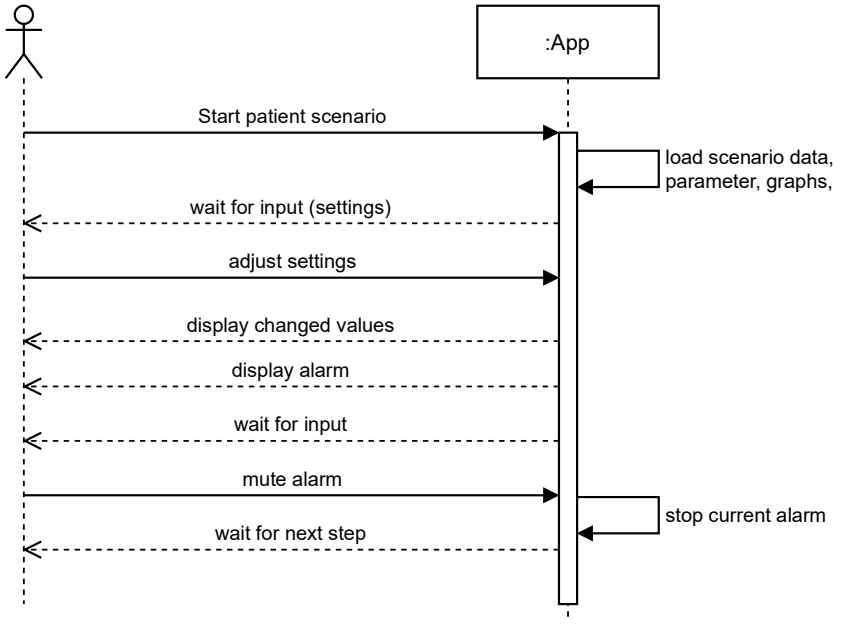
3.4.4 Dynamic Model

The app tester is going to interact with the Flutter App via buttons. Those can be defined explicitly as buttons (like a button to start the defibrillation mode) or interaction buttons at the parameter directly. In the following sequence diagrams, the Flutter App reacts to the user interactions and returns the requested data or starts the requested actions, for example switching into defibrillation mode. The App itself will have a more implemented backend that is hidden for the user. The diagrams concentrate on the user-app interactions with the front-end of the application.

1. Switching into defibrillation mode via “Defi-Button”:



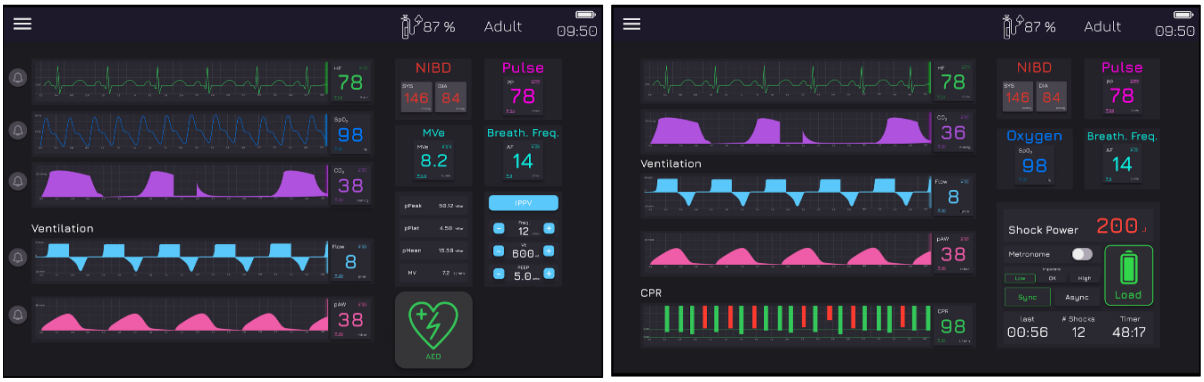
2. Start a patient scenario and set some settings (e.g., ventilation settings):



3.4.5 User Interface

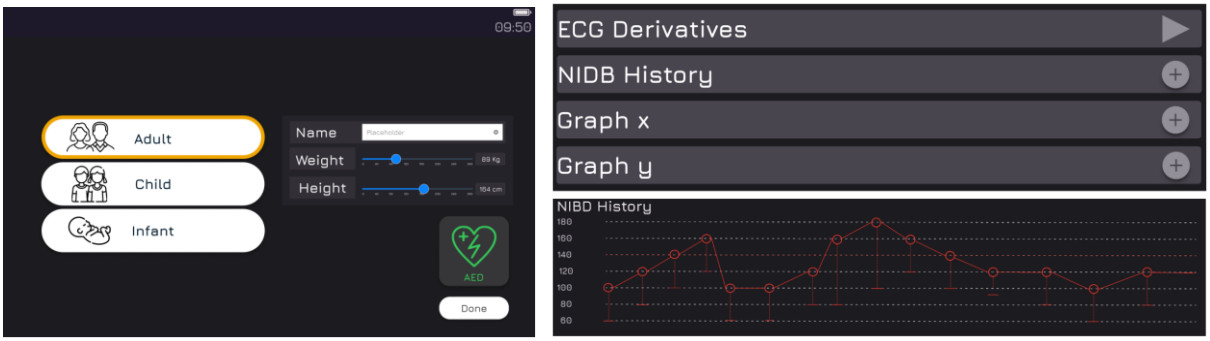
The design will update the current systems display in a modern style with readable colours on a dark background. The light mode can be enabled as a feature that can be added in the future. The accessibility for people who are affected by colour blindness will be neglected. Because of the restricted display size, not all parameters and settings will be displayed at the same time. Changes can be made with dynamic interface actions such as scrolling through a carousel on the right side.

Monitoring with ventilation and defibrillation mode:



Navigating carousel, overview all alarm limits and possible red alarm:

Start screen and add-graph interface, NIBD history graph:



Set single parameter limit: