Requirements Document

Neolook Solutions
AI Switchboard

TA:

Ieva Randyte

Team:

Delia-Andreea Popa
Victor Andrei Didraga
Vasco Reynolds Brandao
Raluca-Alexia Neatu

University of Groningen

Contents

1 Introduction

Neolook Solutions is a spin-out from Philips, which projects its center of attention and activity to the leading hospitals in The Netherlands and Europe. The company specializes in pediatric and neonatal intensive care through commercializing video-augmented services. The Neolook Solutions company attempts to tackle the shortage in nursing staff by effectively optimizing the patient care process through the use of technology.

The aim of this project is to develop an application adhering to certain specifications, which encompasses backend, frontend and a database. The main objective is to develop a simple web application to facilitate the deployment and operation of AI models. This web application will be used by medical staff to aid them in monitoring patients, making decisions and to overall facilitate their jobs.

This document provides the requirements for the assigned project. It gives an insight to the personas involved and splits up the project into Epics, Features and User Stories, System Use Cases and Requirements.

2 Personas

The personas are the people who are going to use the application.

- Medical Staff (Nurses, Doctors) - These will be the users who use the application to select certain AI models and run them. Then, the output data from the model will be presented to the user.
- Developers and engineers from Neolook Solutions - They will use the app in order to add more AI models and make them available for general use by nurses and doctors.

3 Stakeholders

Stakeholders include individuals or entities who have a vested interest in or influence on the outcome of the product. Our team has identified the following list of stakeholders of this project:

- Doctors: interested in accurate and timely patient assessments;
- Nurses: seek efficiency in patient monitoring and care;

- Patients: the ultimate beneficiaries of improved care;
- Neolook Solutions: the company driving the project, aiming for successful implementation and market impact;
- Sponsors: financial backers expecting value and impact from their investment.

4 Epics

In this section, we list the epics, alongside their corresponding descriptions. Each epic contains features, which are further broken down into user stories. These user stories are then divided into the requirements needed to accomplish them. To further establish connections between features and their corresponding user stories, we will identify and list their respective IDs.

4.1 Running and Deployment of AI models

Both the run-on-demand and continuously running models will be used by the medical staff for patient assessment and monitoring.

The run-on-demand models can be selected and triggered from the web application, supporting input data in a variety of formats (image/video/csv/ JSON/plain text) which can either be uploaded locally or from the database.

The continuously running models can be selected and deployed from the web application, with the ability to run in parallel. These models require a live feed of input. In order to facilitate managing multiple patients, the output generated by the continuously running models can be used to notify the medical staff of unexpected changes in the state of the patients being monitored. These alerts are sent through the web application as pop up notifications accompanied by sound effects.

From an extensibility point of view, software developers are able to update and add additional AI models to the application, which can eventually be used by the medical staff to ensure a more complete patient care.

Features:

- Model Selectability and Deployment Options: *US-1*, *US-2*

- Data Input/Output Options: *US-1*, *US-2*, *US-3*
- Continuous Data Analysis: *US-2*, *US-4*
- Change of State/ Diagnostic-Specific Alerts: *US-2*, *US-4*, *US-5*
- Model Updates and Additions: *US-8*

4.2 Data Storage and Retrieval in the Database

Patient data can be manually stored in the database, where it can also be retrieved from, in order to be used for further diagnosis purposes.

The output data from the models gets stored in the database and can be accessed from the web application. This output can also be downloaded locally in a variety of formats (csv/JSON/plain text). If requested, the output data can also include an interpretation in the form of a graph.

Features:

- Efficient Data Storage and Retrieval: *US-6*, *US-7*
- Secure Data Handling: *US-6*, *US-7*
- Custom Data Presentation: *US-1*, *US-3*

5 User Stories

User stories serve to provide the reader with a broad understanding of the features incorporated within the project, as perceived from the perspective of the personas introduced in Section 2 of this document.

Each user story begins with the persona it relates to, followed by a description of a particular feature they seek to utilize and concludes by indicating the overall advantage of this feature within the system. Moreover, testable requirements stemming from each user story are outlined, as well as acceptance criteria that determine when a story is deemed fulfilled. The user stories also include use case diagrams that depict the relationship between the use cases.

In order to categorize the requirements, the MoSCoW (MSCW) technique will be used, which entails in splitting the user stories into requirements, based on their priority in implementing them, into Must-Have, Should-Have, Could-Have and Will-not-Have.

Moreover, each user story has a unique ID, denoted as US-[number], where:

- US = means ‘User Story’
- [number] = the corresponding User Story ID

As for use cases, denoted as US-[number]-UC-[number], the following terminology is being used:

- US = means ‘User Story’
- [number] = the corresponding User Story ID
- UC = means ‘Use Case’
- [number] = the corresponding Use Case ID within the User Story

Finally, for the requirements, denoted as [M/S/C/W]-RQ-[F/NF][number], the following terminology is being used:

- [M/S/C/W] = Must/ Should/ Could/ Will not have
- RQ means ’Requirement’
- [F/NF] = means ‘Functional/ Non-Functional’
- [number] = the corresponding Requirement ID

5.1 US-1

As part of the medical staff, I would like to be able to run and deploy run-on-demand AI models, so that it can aid my decision making by conducting analyses for more specific diagnoses.

5.1.1 Use cases:

- US-1-UC1 → feeding input to run-on-demand AI model;
- US-1-UC2 → retrieving output of run-on-demand AI model;
- US-1-UC3 → accessing run-on-demand AI model;

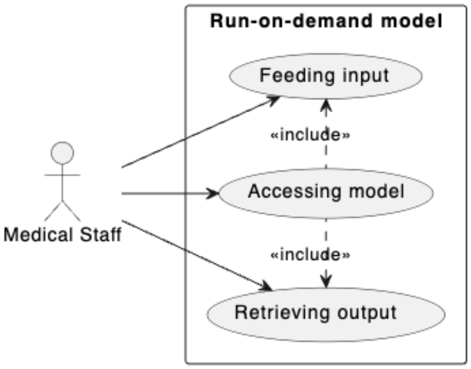

Figure 1: Use case diagram for US-1

Acceptance Criteria:

- Given: A doctor is looking to conduct a fast diagnosis.
- When: A patient requires care.
- Then: The run-on-demand AI model is run to help the doctor take the best course of action.

5.1.2 Requirements:

ID	Requirement	Priority	F/NF
M-RQ-F1	Fronted interface must display a drop down from where the user can select which model will be fed the input.	M	F
M-RQ-F2	Frontend interface must have a button to select and upload images from local files as input for the AI model.	M	F
M-RQ-F3	Frontend interface must have a button to select and upload videos from local files as input for the AI model.	M	F
S-RQ-F4	Frontend interface should have a button to select and upload csv data from local files as input for the AI model.	S	F

S-RQ-F5	Frontend interface should have a button to select and upload JSON data from local files as input for the AI model.	S	F
S-RQ-F6	Frontend interface should have a button to select and upload plain text data from local files as input for the AI model.	S	F
C-RQ-F7	Frontend interface could have a button to select images from the database as input for the AI model.	C	F
C-RQ-F8	Frontend interface could have a button to select videos from the database as input for the AI model.	C	F
C-RQ-F9	Frontend interface could have a button to select csv data from the database as input for the AI model.	C	F
C-RQ-F10	Frontend interface could have a button to select JSON data from the database as input for the AI model.	C	F
M-RQ-F11	AI model output must be stored in the database.	M	F

5.2 US-2

As part of the medical staff, I would like to be able to run and deploy continuously running AI models, so that it can aid me by monitoring patients and facilitating managing multiple patients at once.

5.2.1 Use cases:

- US-2-UC1 → feeding live data to continuous AI model;
- US-2-UC2 → retrieving output of continuous AI model;
- US-2-UC3 → accessing the continuous AI model;

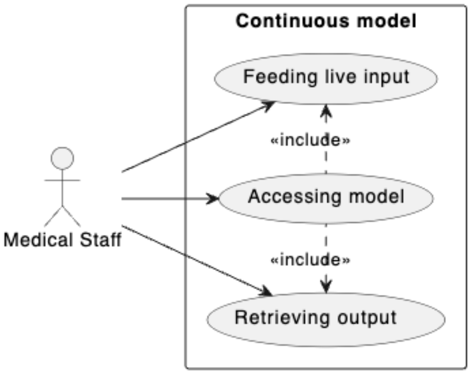

Figure 2: Use case diagram for US-2

Acceptance Criteria:

- Given: Doctors have multiple patients to manage.
- When: Patients require monitoring.
- Then: The continuous AI model is run, and the medical staff can be notified through the app of any unexpected state change involving the monitored patients.

5.2.2 Requirements:

ID	Requirement	Priority	F/NF
M-RQ-F12	Live feed of data must be fed to continuously running models.	M	F
C-RQ-F13	Frontend interface could have a button that opens a dropdown menu displaying the available continuous models.	C	F
C-RQ-F14	The table displaying the available continuous models could have a button for switching them on and off.	C	F

5.3 US-3

As part of the medical staff, I would like to be able to access/visualize the data produced by the AI models, so that I can efficiently analyze it myself and take further action based on it.

5.3.1 Use cases:

- US-3-UC1 → retrieving output of AI model;
- US-3-UC2 → presenting/interpreting output data;

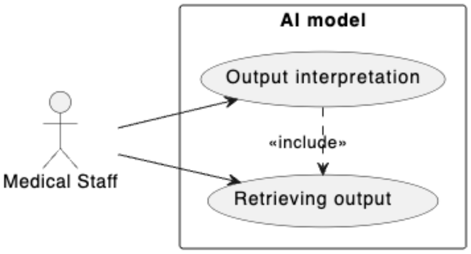

Figure 3: Use case diagram for US-3

Acceptance Criteria:

- Given: A doctor needs to visualize the data produced by the AI model.
- When: The data extracted by the model does not provide a direct conclusion.
- Then: The system provides a clear and intuitive graph interpretation of the data.

5.3.2 Requirements:

ID	Requirement	Priority	F/NF
M-RQ-F15	Frontend interface must have a button that allows the user to download the output of the AI model.	M	F
C-RQ-F16	The output of the models could also include an interpretation of the data together with the data.	C	F

S-RQ-F17	The download button should prompt the user to download the AI output in plain text format.	S	F
S-RQ-F18	The download button should prompt the user to download the AI output in csv format.	S	F
S-RQ-F19	The download button should prompt the user to download the AI output in JSON format.	S	F
S-RQ-F20	The download button should prompt the user to download the AI output in image format.	S	F
S-RQ-F21	The download button should prompt the user to download the AI output in video format.	S	F

5.4 US-4

As part of the medical staff, I would like to receive alerts generated from the continuously running models' data, so that it can aid me in the management of multiple patients simultaneously, by eliminating the need for my vigilance, as the AI models promptly notifies me of any detected abnormalities.

5.4.1 Use cases:

- US-4-UC1 → receiving alerts from continuous AI models' output data;

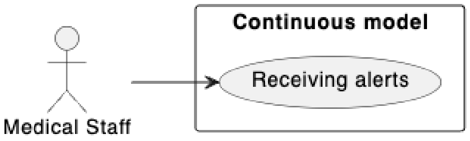

Figure 4: Use case diagram for US-4

Acceptance Criteria:

- Given: Patients require monitoring.

- When: Medical staff is momentarily unavailable/ has other priorities.

- Then: The system sends change of state notifications to the medical staff in case the monitored patient becomes a higher priority.

5.4.2 Requirements:

ID	Requirement	Priority	F/NF
M-RQ-F22	Continuous monitoring models must notify users when any activity outside the specified parameters is identified.	M	F
M-RQ-F23	Continuous monitoring models must provide the data that triggered the notifications when notifying the users.	M	F

5.5 US-5

As part of the medical staff, I would like the alerts generated from the on-demand AI models' output data to promptly notify me if a specific diagnosis is detected and requires immediate attention.

5.5.1 Use cases:

- US-5-UC1 → receiving alerts from run on demand AI models' output data;

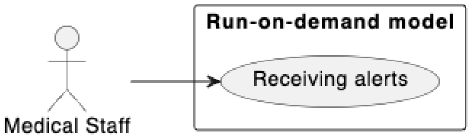

Figure 5: Use case diagram for US-5

Acceptance Criteria:

- Given: A doctor is analyzing the output produced by the run on demand model.

- When: A run on demand AI model was run with a patient's data input.

- Then: The doctor is alerted in case the symptoms of the patient indicate a specific diagnosis.

5.5.2 Requirements:

ID	Requirement	Priority	F/NF
C-RQ-F24	Run-on-demand models could notify users when any specific diagnosis is identified.	C	F
C-RQ-F25	Run-on-demand models could provide the data that triggered the diagnosis notifications when notifying the users.	C	F

5.6 US-6

As part of the medical staff, I would like to be able to effectively retrieve patient data through the database in a quick and safe manner, such that data can be passed around efficiently.

5.6.1 Use cases:

- US-6-UC1 → filtering data from the database;
- US-6-UC2 → retrieving data from the database;

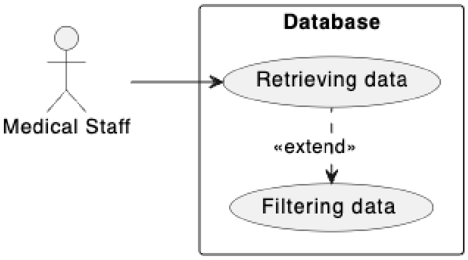

Figure 6: Use case diagram for US-6

Acceptance Criteria:

- Given: A database exists.

- When: Patient data is in the process of being retrieved from the database.
- Then: The patient data is successfully retrieved from the database, aiding the medical staff in taking further action in nursing/diagnosing the patient.

5.6.2 Requirements:

ID	Requirement	Priority	F/NF
S-RQ-F26	Frontend interface should have a button that allows for the data to be downloaded from the database.	S	F
S-RQ-F27	Frontend interface should have a button that allows for the data to be filtered based on the file format.	S	F

5.7 US-7

As part of the medical staff, I would like to be able to safely and efficiently store patient data in the database, so that it can be later updated in case certain changes occur or routine check-ups are performed on the patient.

5.7.1 Use cases:

- US-7-UC1 → uploading patient data to the database;
- US-7-UC2 → deleting data from the database;

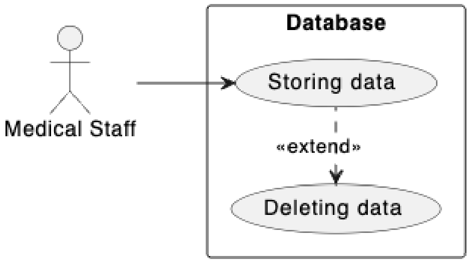

Figure 7: Use case diagram for US-7

Acceptance Criteria:

- Given: A database exists.
- When: Patient data is in the process of being uploaded to the database.
- Then: The patient data is successfully stored in the database, which is used to create, read, update and delete entries.

5.7.2 Requirements:

ID	Requirement	Priority	F/NF
M-RQ-F28	Frontend interface must have a button that allows for the patient data to be stored in the database.	M	F
M-RQ-F29	The patient data that is being stored in the database must be encrypted and anonymous.	M	F
S-RQ-F30	Frontend interface should have a button that allows for files to be deleted from the database.	S	F

5.8 US-8

As a software developer, I would like to be able to extend the application by adding new AI models that can then be deployed, so that the web application can become scalable, given that different AI models might serve for different types of monitoring and diagnosis.

5.8.1 Use cases:

- US-8-UC1 → system maintainer wants to make new AI model available;

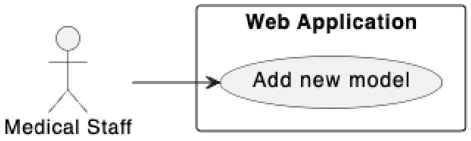

Figure 8: Use case diagram for US-8

Acceptance Criteria:

– Given: A new model is available for medical use.

– When: Old AI models are already included in the web application.

– Then: The new model is integrated in the web application.

5.8.2 Requirements:

ID	Requirement	Priority	F/NF
M-RQ-NF31	The interactions with the AI models must be done over interfaces.	M	NF