1. Regulatory References

Regulatory references:

IEC 62304, para. 5.3.1 and 5.3.2 [class B, C]

Relevant other documentation:

- SOP Software Development
- User needs / stakeholder requirements
- Design input / software requirements
- (. . .)

2. Software Systems

In compliance with DIN EN 62304, we subdivide our software on three levels: software systems, software components and software units.

> Here, describe your internal software systems. The IEC 62304 defines those as an "integrated collection of software items organized to accomplish a specific function or set of functions."
>
> NOTE: Ideally, you would add an illustrating diagram to the Annex and reference it here.

2.1. Frontend

> Enter description, for example:
>
> - Function: user interface display
> - Software safety classification and rationale
> - Runtime
> - Deployment
> - User groups

2.2. Backend

> Enter description, for example:
>
> - Function: managing patient data and medical images.
> - Software safety classification and rationale
> - Runtime (e.g. JVM)
> - Deployment (e.g. Docker container)
> - User group

2.3. Algorithm

> Enter description, for example:

- Function: taking medical images as input and output a prediction.
- Software safety classification and rationale
- Runtime (e.g. JVM)
- Deployment (e.g. Docker container)
- User group

3. Software Units

Describe your internal software units. The IEC 62304 defines those as a "software item [any identifiable part of a program, i.e. source code, object code, control code, control data, etc.] that cannot be subdivided into other items". For example:

- Wearable device poller (regularly checks whether wearable device has new data and downloads it)
- Notification service (sends messages to Apple / Google for push notifications of mobile apps)
- (...)

4. Database

Describe your databases. For example:

- Relational database: Postgres v14

5. IT Security

5.1. Encryption of data

<enter content>

5.1.1. Data at rest

<enter content>

5.1.2. Data in transit

Example content:

- Data in transit is encrypted with state-of-the-art encryption, e.g. SSL, TLS.
- Additionally, we create a Virtual Private Network (VPC) which prevents the Compute Instances from being exposed to the public internet. The algorithm and the database are therefore not publicly reachable; they are only reachable by the backend.