	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		**OMNI-SM-002**	**1**

Contents

	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		OMNI-SM-002	1

	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		**OMNI-SM-002**	**1**

1 Document Purpose

This document defines the requirements for the OMNI Summit Microservice. Requirements are established for the intended use, categorization, labeling, controls, indicators, interfaces, functional characteristics, performance characteristics, and computing platform associated with the device as well as for other aspects of the software.

2 Document Scope

This document is intended to be used as an internal engineering specification. The requirements defined in this document are to be used as input to the design process, and as a basis for verification testing.

3 Background and Categorization

The main features of the OMNI Summit Microservice involve the technology stack that allows for sensing interfaces utilizing a gRPC enabled microservice for the Medtronic Summit RC+S device. The user can carry out implementation of algorithms and data collection with the help of these interfaces. The product is an open source software that offers modularity and seamless system integration.

In this document we will be following the OMNI convention of referring to implanted neurostimulators as "Devices" and device-enabled communication equipment as "Bridges". For the Summit system specifically, the Summit INS is a "Device" and the Summit CTM is a "Bridge".

This software is developed with the intention of being used as "SOUP" (Software Of Unknown Provenance) or "OTS" (Off-The-Shelf) in conjunction with other software developed as part of a research protocol using the investigational Summit System by Medtronic. The exact risks of using this software will be determined by the risks defined in the protocol in the specific use-case.

Reference Links :

      1.https://www.fda.gov/medical-devices/digital-health-center-excellence/software-medical-device-samd

---

	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		**OMNI-SM-002**	**1**

2.http://www.imdrf.org/docs/imdrf/final/technical/imdrf-tech-140918-samd-framework-risk-categorization-141013.pdf

4 Controls, Indicators, and Interfaces

Requirement 4.1: Application Interface

The Microservice shall include an application programming interface (API) based on gRPC. This is defined in the *OMNI Summit Microservice Interface Control Document (ICD) v1.x*.

Requirement 4.2: Device Interface

The Microservice shall interface with a Medtronic Summit System via the Summit DLLs: Medtronic.SummitAPI.dll, Medtronic.TelemetryM.dll, Medtronic.NeuroStim.Olympus.dll

Requirement 4.3: Console Interface

The Microservice shall run in a standalone command console interface, indicating that it is active by writing a start-up message, and allowing for graceful shutdown via keypress or via the exit button on the terminal window.

	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		**OMNI-SM-002**	**1**

5 Functional Characteristics

5.1 Operational Modes

Requirement 5.1.1: Start-up Mode

During start-up the microservice shall:

- Initialize a gRPC server with three gRPC-defined services:
  - A device service
  - A bridge service
  - A supporting info service
- Write to the console the current version number and supported devices before entering the 'Active Mode'.

Requirement 5.1.2: Active Mode

While in ACTIVE mode, the microservice shall:

- Allow a connected client to request a function to be called on a device or bridge instance.
- If the device or bridge does not exist, the request will be denied.
- If the device or bridge does exist, the function call is forwarded on to the Summit DLL defined instance, which after completion of execution returns the response to the client.

Requirement 5.1.3: Shut-Down Mode

When the user enters 'q' into the command console or presses the 'x' button to close the window, the microservice shall enter shut-down mode. In shut-down mode the microservice shall:

- safely close all connections to gRPC clients and connections to devices and bridges
- close the port the microservice was listening on and terminate the process.

5.2 Command Subsystem

Requirement 5.2.1: Support Commands

The Microservice shall provide an interface for querying the state of the microservice itself through:

- **Version**, which shall return the microservice major, minor, and patch version numbers, and supported devices.
- **SupportedDevices**, which shall return the class of supported implantable neurostimulators (i.e. Medtronic Summit RC+S).
- **InspectRepository**, which shall return all cached device and bridge connections the microservice is currently storing in the in memory repository.

Requirement 5.2.2: Bridge Query Commands

The Microservice shall provide an interface for querying and managing the state of Telemetry Bridges through:

- **ListBridges**, which shall return a list of the known bridges and discover new bridges.
- **ConnectedBridges**, which shall return a list of the bridges currently connected to by the microservice.
- **DescribeBridge**, which shall return the telemetry information of the bridge device

Requirement 5.2.3: Bridge Management Commands

- **ConnectToBridge**, which shall initiate connection to the bridge, and set the number of reconnect attempts before failure.

	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		OMNI-SM-002	1

- **DisconnectFromBridge**, which shall dispose of the device and remove the details of the connection from the repository.

Requirement 5.2.4: Bridge Configuration Commands

- **ConfigureBeep**, which shall enable telemetry sound state of bridge.

Requirement 5.2.5: Bridge Connection Status Streaming

The Microservice shall provide an interface for streaming bridge related status:

- **CreateBridgeConnectionStream,** creates a streaming connection from the bridge connection status updates to be sent to the client application.

Requirement 5.2.6: Device Query Commands

The Microservice shall provide an interface for querying the state of INS devices through:

- **ListDevices**, which shall return a list of devices connectable by a given bridge.
- **DeviceStatus**, which shall return the battery status of the device connected
- **LeadIntegrityTest**, which shall return the impedance values between given pairs of electrodes

Requirement 5.2.7: Device Management Commands

The Microservice shall provide an interface for querying and managing the state of INS devices through:

- **ConnectToDevice**, which shall establish a connection between the device, bridge, and the server.
- **DisconnectFromDevice**, which shall disband a connection between a given device, bridge, and the server.

Requirement 5.2.8: Device Configuration Commands

- **SenseConfiguration**, which shall configure all sense settings. The configurable settings shall include:
    - Time Domain Channel Configuration
    - Fourier Transform Channel Configuration
    - Power Channel Configuration
    - Linear-Discriminant Classifier Configuration
    - Accelerometer Configuration
    - Miscellaneous Summit Configuration
    - Sense State Enabling

Requirement 5.2.9: Device Data Streaming Commands

The Microservice shall provide an gRPC interface to provide data streaming through the following commands:

- **StreamEnable**, which shall enable streaming capabilities from the device, through the bridge, to the client application.
- **StreamDisable**, which shall disable streaming capabilities from the device, through the bridge, to the client application.
- **TimeDomainStream**, which shall create a stream for Time Domain data updates to be sent from the device, through the bridge, to the client application.
- **FourierTransformStream**, which shall create a stream for Spectral data updates to be sent from the device, through the bridge, to the client application.
- **BandPowerStream**, which shall create a stream for Band Power data updates to be sent from the device, through the bridge, to the client application.
- **InertialStream**, which shall create a stream for Inertial data updates to be sent from the device, through the bridge, to the client application.
- **AdaptiveStream**, which shall create a stream for Adaptive Status updates to be sent from the device, through the bridge, to the client application.

- **LoopRecordUpdateStream**, which shall create a stream for Loop Recorder Status updates to be sent from the device, through the bridge, to the client application.
- **EchoStream**, which shall create a stream for Echo data updates to be sent from the device, through the bridge, to the client application.

5.3 Self-Test Subsystem

Requirement 5.3.1: Summit System Connection Monitoring

When a connected Summit System throws a disconnected event, the Microservice shall begin to send a battery status request every 10 seconds to monitor the connectivity status of the bridge and device. If the Summit System indicates that the connection is not automatically recoverable, the microservice shall then:

- Send a message to connected clients via the bridge connection status streaming endpoint.
- Make an attempt to reconnect to the device automatically up to a specified number of retries.
- Abort the connection attempts after the specified number of retries are exceeded.

6 Performance Characteristics

6.1 Number of Users

Requirement 6.1.1: Multiple clients to communicate with Summit Systems

The microservice shall allow multiple connected clients to initiate gRPC-enabled functionality utilizing bridge/device name pairs.

Requirement 6.1.2: Each stream can communicate to single endpoint

The microservice shall allow each streamed data type to be sent to a single client.

6.2 Number of Summit Systems

Requirement 6.2.1: Microservice will support two Summit Systems Simultaneously

	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		OMNI-SM-002	1

The microservice shall support at least two Summit Systems to be simultaneously communicated with by client applications.

7 Computing Platform

Requirement 7.1: The Microservice shall run on Windows

The microservice shall run on Windows 10 with 64 bit depth.

Requirement 7.2: The Microservice shall support gRPC

The microservice shall support gRPC clients using version gRPC 1.0.0 or greater.

Requirement 7.3: The Microservice shall serialize messages using protobuf

The microservice shall serialize/deserialize all outgoing/incoming messages using protobuf version 3.

8 Protection from Software Hazards

8.1 Data Integrity Checking

There are no OMNI related requirements pertaining to Data Integrity because gRPC leverages HTTP/2 which is built on TCP and thus inherits all the guarantees of the TCP/IP protocol.

8.2 Flow Control and Monitoring

There are no OMNI related requirements pertaining to Flow Control because gRPC leverages HTTP/2 which is built on TCP and thus inherits all the guarantees of the TCP/IP protocol.

8.3 Error Trapping and Handling

There are no OMNI related requirements pertaining to gRPC error handling because gRPC captures all exceptions thrown while handling network communication with a client application, and surfaces them to the client application.

8.4 Network Listening Address Configuration

gRPC allows for the configuration of the listening address to be a network accessible endpoint, which may allow for unauthenticated users to remotely interact with the OMNI microservice. These interactions may result in the disclosure of HIPAA-protected information (device serial numbers) across the network.

Requirement 8.4.1: The Microservice shall use a localhost-only default address

The microservice shall default all network listening interfaces to be localhost only unless overwritten at the user's discretion. It shall be the user's responsibility to ensure that HIPAA related information is secure in the case where this is overwritten with a non-localhost only address.

9 Packaging

Requirement 9.1: Installer

The microservice shall be distributed via an installer program. The installer program shall install the microservice binary, as well as ask for location of Summit RC+S DLLs for the microservice to use.

	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		**OMNI-SM-002**	**1**

10 Labeling Requirements

Requirement 10.1: Version Numbering

The microservice shall follow semantic versioning (i.e. 1.2.3 where 1 is major version, 2 is minor version, 3 is patch version, or 1.0.0-beta for a beta release of version one. More information can be found: https://semver.org/spec/v2.0.0.html). The version shall be displayed during microservice start-up, and shall be available via the version endpoint.

Requirement 10.2: Packaging Labeling

The installer's version shall reflect the version of the microservice.

Requirement 10.3: Documentation Generation

The documentation shall be updated and deployed automatically with each new release.

Requirement 10.4: Protobuf Documentation

The messages, endpoints, enums, etc. in the protobuf interface files shall be documented in a format that allows for auto generation of documentation.

11 Reference Information

11.1 Related Documents

- Summit API Specification - provided under research agreement with Medtronic, not distributed by the OMNI team.
- Summit User Manual - provided under research agreement with Medtronic, not distributed by the OMNI team.
- Protobuf 3 specifications
- gRPC specifications

	OMNI Summit Microservice V1.X Software Requirements Specification	Doc. Number	Rev.
Open Mind		**OMNI-SM-002**	**1**

11.2 Definitions

Term	Definition
bridge	A hardware device that proxies messages to the implantable neurostimulator
device	The implantable neurostimulator

13 Approvals

Approver Role	Signature and Date
Principle Investigator	DocuSigned by: David Borton 8A588B8FA1F246E... — 3/2/2022 | 1:22:43 PM EST
Lead Developer	DocuSigned by: Jeffrey Herron 64503D12F815426... — 3/2/2022 | 10:25:43 AM PST
Author	DocuSigned by: Bradford N Roarr 9617A5434271462... — 3/2/2022 | 1:19:44 PM EST