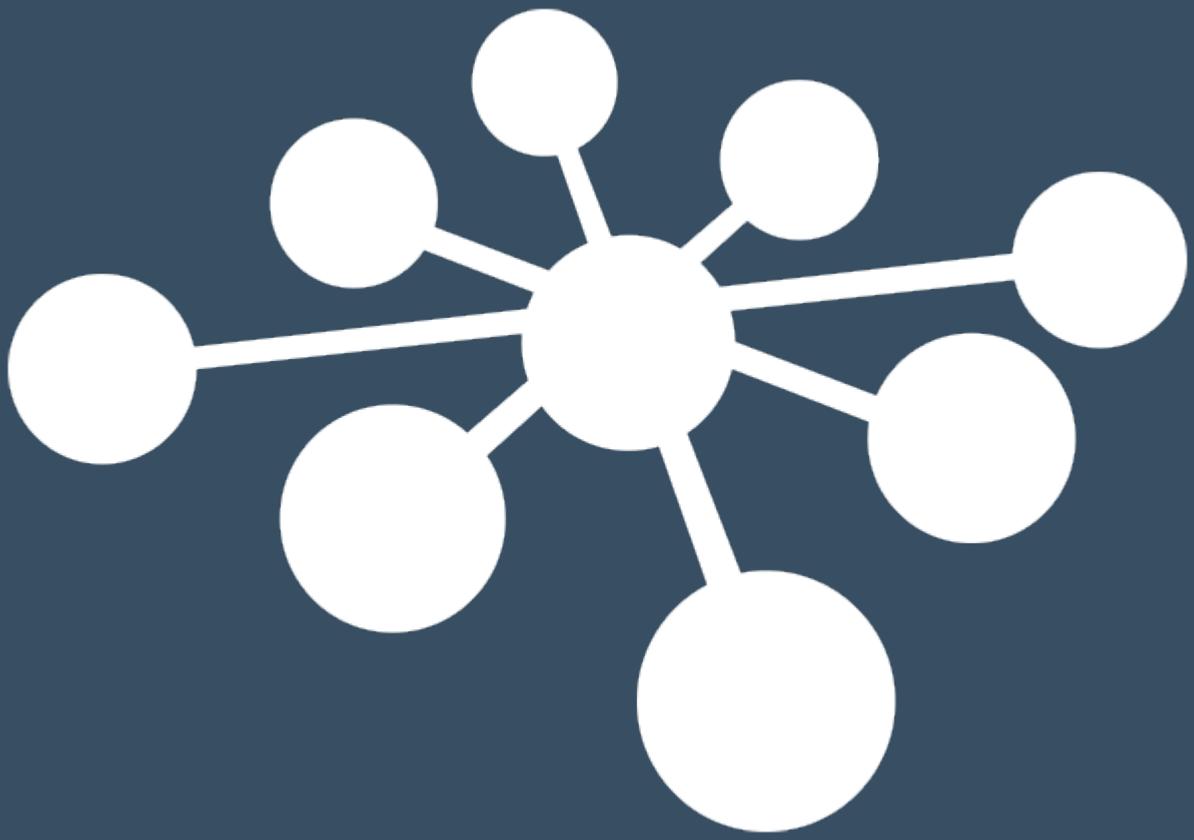
Instructions for Use

OtoAccess® ADI



D-0107046-C – 2019/03

Table of Contents

1 Introduction 1
1.1 About this Manual 1
1.2 The ADI overview 1
1.3 Implementation 2

2 License 3

3 Installation and configuration 4
3.1 PC requirements 4
3.2 API 4
3.3 Legacy support 4

4 Documentation 5

5 Registering hook 6
5.1 Description 6
5.2 Sequence diagram 6

6 Getting list of registered applications 7
6.1 Description 7
6.2 Sequence diagram 7

7 Setting current patient 8
7.1 Description 8
7.2 Sequence diagram 8

8 Launching registered application 9
8.1 Description 9
8.2 Sequence diagram 9

9 Retrieving data from application 10
9.1 Description 10
9.2 Sequence diagram 11

10 Retrieving printouts 12
10.1 Description 12
10.2 Sequence diagram 12

11 Sending saved session back to application for review 13
11.1 Description 13
11.2 Sequence diagram 14

12 Working with standalone instruments 15
12.1 Connect/Disconnect instruments 15
12.1.1 Description 15
12.1.2 Sequence diagram 15
12.2 Add/Remove patients from instruments 16
12.2.1 Description 16
12.2.2 Sequence diagram 16
12.3 Get sessions 17
12.3.1 Description 17
12.3.2 Sequence diagram 17
12.4 Get tests data 18
12.4.1 Description 18

12.4.2 Sequence diagram 18

13 Setting clinic and examiner 19
13.1 Description 19
13.2 Sequence diagram 19

1 Introduction

1.1 About this Manual

This manual is valid for the Audiometric Data Interface. The product is manufactured by:

OtoAccess A/S
Audiometer Allé 1
5500 Middelfart
Denmark

1.2 The ADI overview

The Audiometric Data Interface (ADI) is an interface between the audiological hardware and an EMR system. The ADI provides a pathway to enable staff at hospital and healthcare practices to securely store and retrieve data from the EMR system. Manufacturers can use it to seamlessly integrate supported standalone audiometric hardware to the many EMR systems, enabling an efficient, streamlined user workflow.

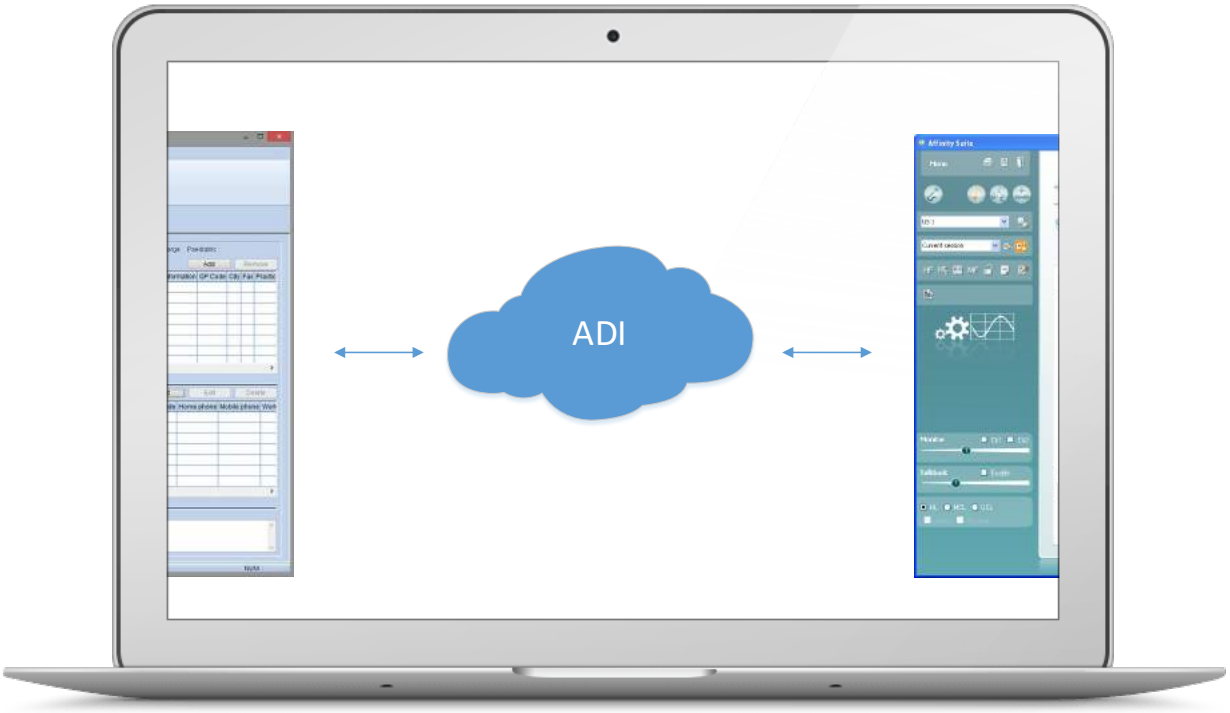


Figure 1: *The* ***ADI is placed between EMR system and audiological software/hardware.***

1.3 Implementation

The current ADI is implemented as a local (not networked) self-hosted WebAPI and it is installed as a windows service. It expose communication via REST api which is well-documented and easly accessed for the 3rd party manufacturers.

This document describes how to use ADI to communicate with supported products via an HTTP-based RESTful API. The provided API will work with software modules or direct connection to a standalone instrument.

Usages:

- Send/Retrieve Patient demographics
- Send/Retrieve Sessions
- Delete patient/sessions

Note:
This interface is not intended for controlling the software or the instruments, but to exchange patient information and sessions.

2 License

Important!

PC Sync or PC control license is in some cases needed in order to be able to connect to the instruments via ADI. Instruments which are not license controlled will connect with ADI.

3 Installation and configuration

3.1 PC requirements

- Windows® 7 SP1, Windows® 8.x or Windows® 10
- .NET 4.6.2

3.2 API

The RESTful API is installed as a Windows® Service.
Instalation process requires using Command Line or Power Shell, where msiexec command is executed. Server and printer port numbers are configurable while installation process with PORT_NUMBER_PRINTER and PORT_NUMBER_SERVER arguments. Port number must be number from 1 to 65535 included. Logging while installing is highly recommended. Example of command executed in folder, where installation file occurs:

msiexec /i "OtoAccess ADI Server Setup.msi" PORT_NUMBER_PRINTER="X" ^ PORT_NUMBER_SERVER="Y" /l "log.txt"*

Port numbers are not required, default values are: 9100 for printer and 47294 for server.
Please refer to the **readme.txt** file for a full list of configurable parameters.

3.3 Legacy support

Interfacing to legacy modules (e.g. Eclipse and VNG) is achieved by installing the IaBaseProxy application.
Legacy applications lack the ability to save reports automatically to the API, and instead requires the user to print reports manually to the ADI Printer that is installed as part of the API.
As API, IaBaseProxy must be installed via msiexec command. List of installed modules is configurable while installation process with MODULES argument. Available modules: Teoae, Vng, Dpoae, Ep, Assr and Abris. Logging while installing is highly recommended. Example of command executed in folder, where installation file occurs:

msiexec /i "ADI IaBaseProxy Setup.msi" MODULES="Assr,Vng" ^
INSTALLFOLDER="C:\ExampleFolder\ExampleSubFolder" /l "log.txt"*

INSTALLFOLDER is not required, defalt: C:\Program Files (x86)\OtoAccess\Audiometric Data Interface.
MODULES argument cannot be empty.
Please refer to the **readme.txt** file for a full list of configurable parameters.

4 Documentation

Once installed, the API provides an interactive help page at the following address: http://localhost:PORT_NUMBER/help. It is based on reliable Swagger RESTful API documentation specification. The documentation site allows client to check available API operations with short description and multiple response content type. There is a possibility to test methods via “TRY IT OUT!” option. To get access to Server’s data, provided api_key is required.

PORT_NUMBER is configured during installation (see section API) and default is 47294.

Note that the port number can be specified during installation and might differ from the above address.

In order to open the OtoAccess® ADI demo database with the documentation site please press the apikey ‘ApiKey Public’.

5 Registering hook

5.1 Description

A hook is a mechanism by which an application can intercept events, such as messages like POST and GET HTTP methods. It allows for two-way communication between Client and ADI.
A hook can be registered via query the ADI via POST /hooks?portNumber=XYZ, where XYZ is a port number.

Swagger doc: http://URL/help/index#!/Hooks/Hooks_AddHook.

5.2 Sequence diagram

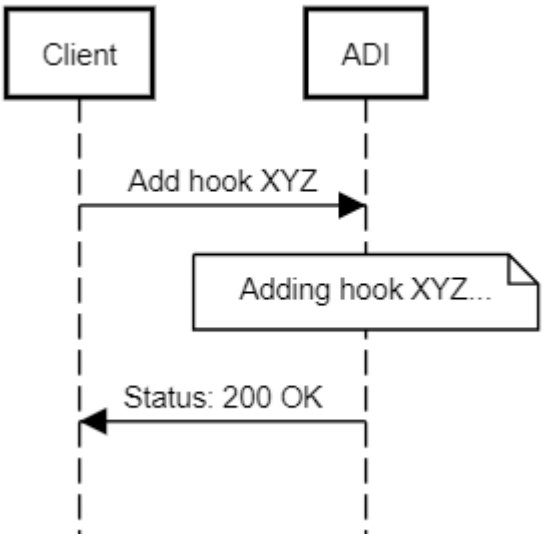


Figure 2 Adding hook

6 Getting list of registered applications

6.1 Description

The registered applications are suites and modules installed on the given machine. In order to list them, query the ADI via GET /applications. In response list of application will be attached.

Note that every application contains detaileds information like: Filename, Arguments etc.

Swagger doc: http://URL/help/index#!/Applications/Applications_GetApplications

6.2 Sequence diagram

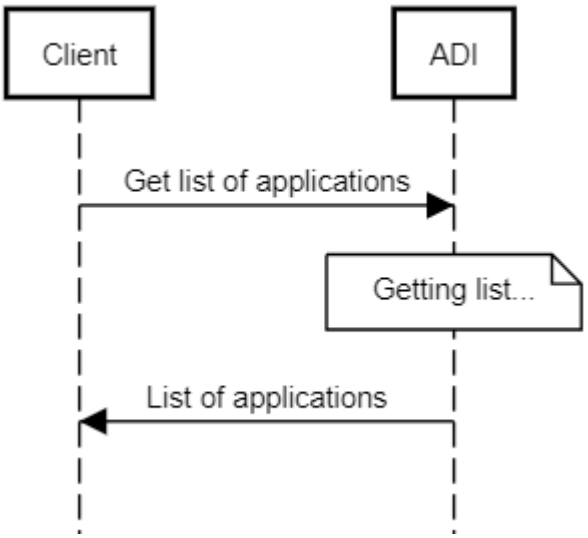


Figure 3 Getting list of registered applications

7 Setting current patient

7.1 Description

To work with suites or modules, current patient have to be set. To do that send patient to ADI via POST /patient.

Swagger doc: http://URL/help/index#!/Patient/Patient_Post

7.2 Sequence diagram

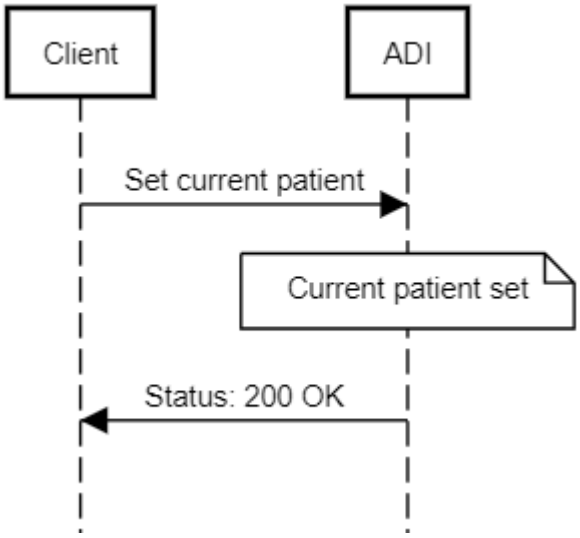


Figure 4 Setting current patient

8 Launching registered application

8.1 Description

To launch application, the current patient has to be set already (see section 7). Next step is to find and execute registered application (see section 6). As a result, suite or module should launch with patient data.

Swagger doc:
http://URL/help/index#!/Applications/Applications_GetApplications
http://URL/help/index#!/Patient/Patient_Post

8.2 Sequence diagram

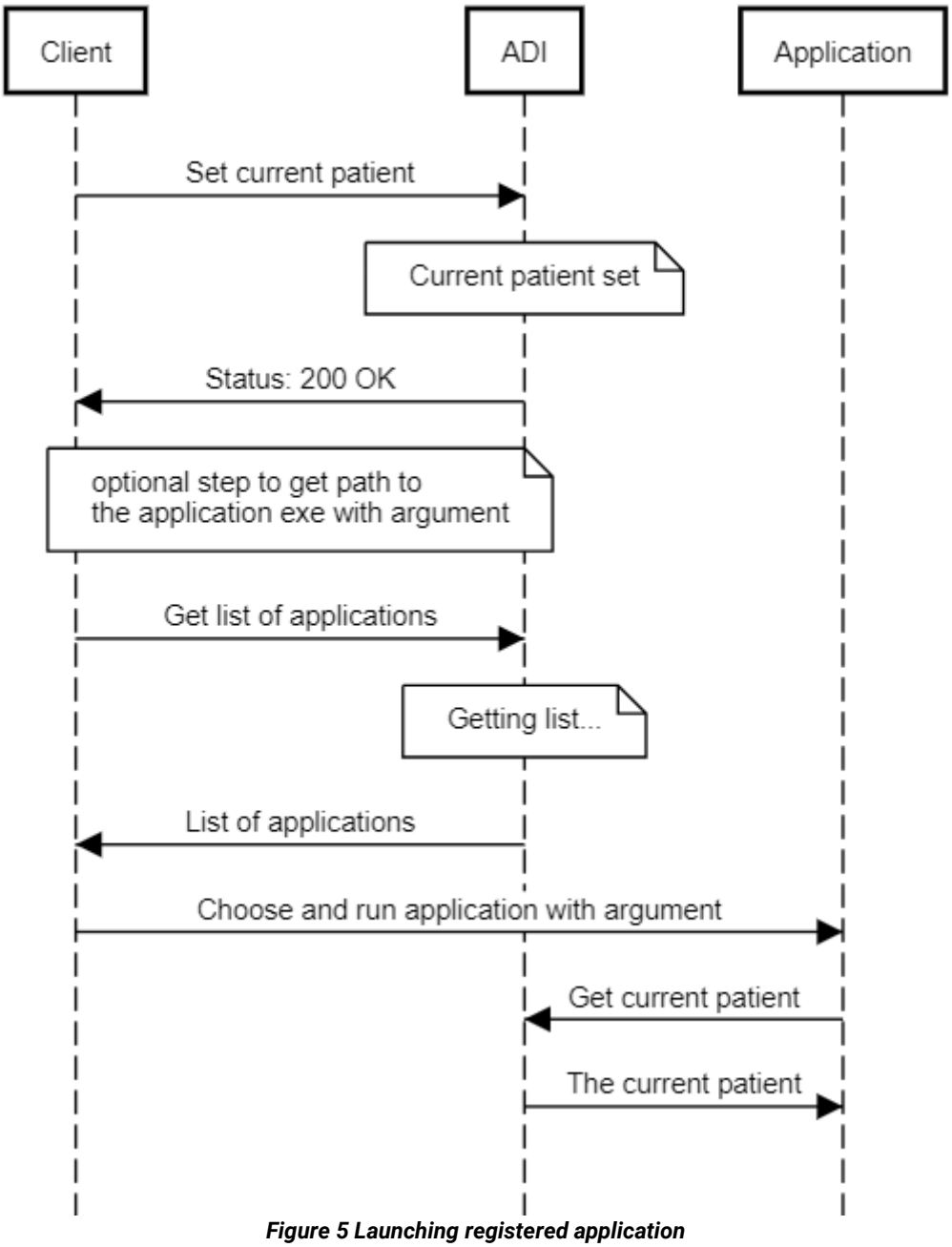


Figure 5 Launching registered application

9 Retrieving data from application

9.1 Description

When data is available, suite or module will send notification under registered hook (see section 5). The client is responsible for retrieving or cleaning data. To get list of available data, query ADI via GET /sessions, in response you will get list of saved sessions. Then data is available by reference to a specific session's index.

Please note that the report data are only available for the PC Suites (see relevant brand specific technical note). In order to take it from Legacy Modules please see section 10 and relevant brand specific technical note).

In order to retrieve:

- Binary data
  GET /sessions/{index}
- Test data
  GET /sessions/{index}/tests
- Report data
  GET /sessions/{index}/report
- Image data
  GET /sessions/{index}/image

In order to clean:

- DELETE /sessions/{index}

Swagger doc:
http://URL/help/index#!/Sessions/Sessions_Get
http://URL/help/index#!/Sessions/Sessions_Get_0
http://URL/help/index#!/Sessions/Sessions_GetTests
http://URL/help/index#!/Sessions/Sessions_GetReport
http://URL/help/index#!/Sessions/Sessions_GetImage
http://URL/help/index#!/Sessions/Sessions_Delete

9.2 Sequence diagram

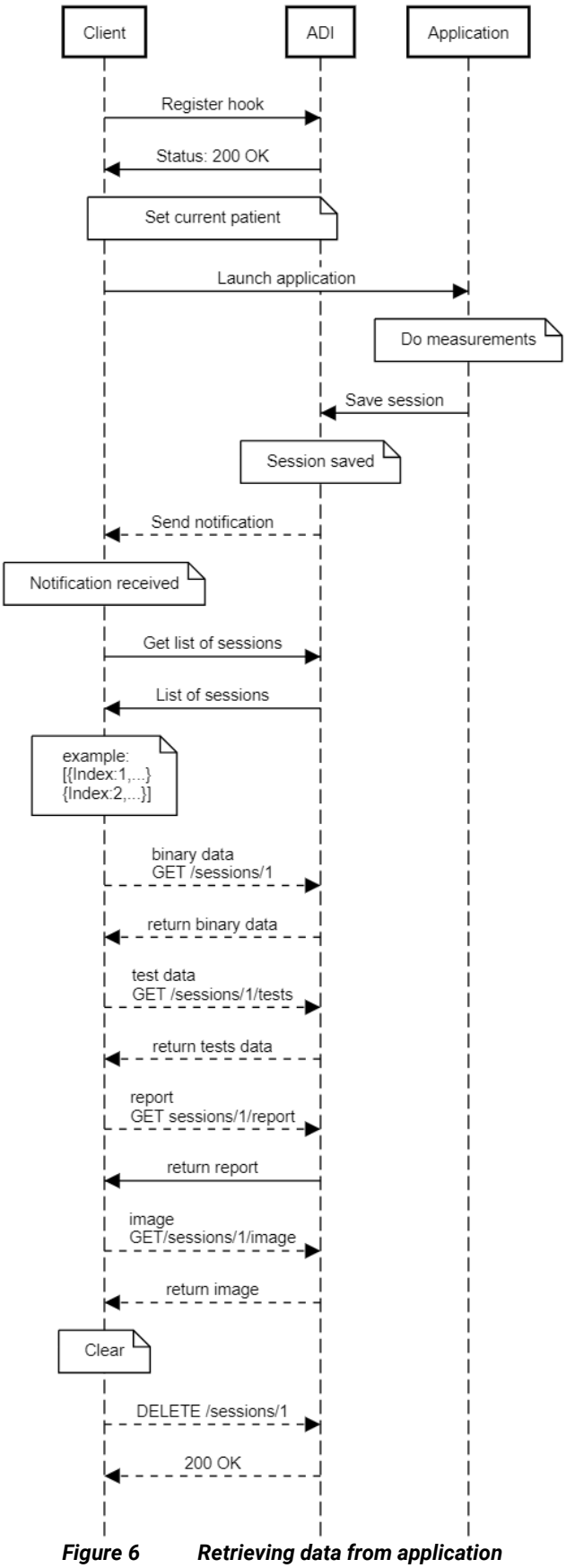


Figure 6 Retrieving data from application

10 Retrieving printouts

10.1 Description

In order to list printouts, query the ADI via GET /printout. To get specified report, the used query is GET /printout/indexnumber/report with specified raport's index number.

Please note that 'printouts' are the reports from the Legacy Modules which use printer to send report data. See relevant brand specific technical note.

Swagger doc:
http://localhost:47294/help/index#!/Printout

10.2 Sequence diagram

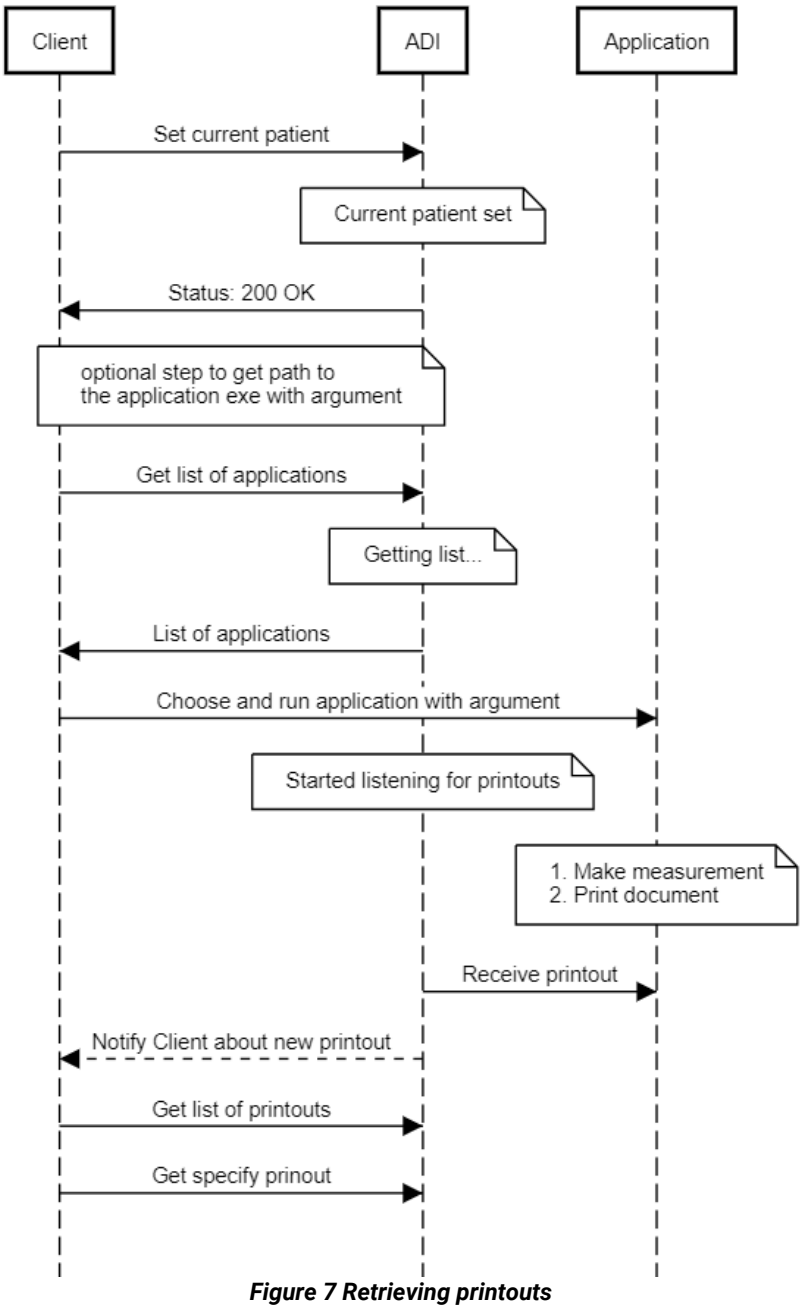


Figure 7 Retrieving printouts

11 Sending saved session back to application for review

11.1 Description

To be able to review sessions within application, Client has to set the current patient with attached sessions. In order to do that Client retrieve session and match it with proper patient by 'GUID' property (Patient) and 'PatientGuid' (Session). Next attach proper session binary data to the patient's 'Sessions' property and set patient as current patient into ADI.

Swagger doc:
http://URL/help/index#!/Patient/Patient_Get
http://URL/help/index#!/Patient/Patient_Post
http://URL/help/index#!/Sessions/Sessions_Get
http://URL/help/index#!/Sessions/Sessions_Get_0

11.2 Sequence diagram

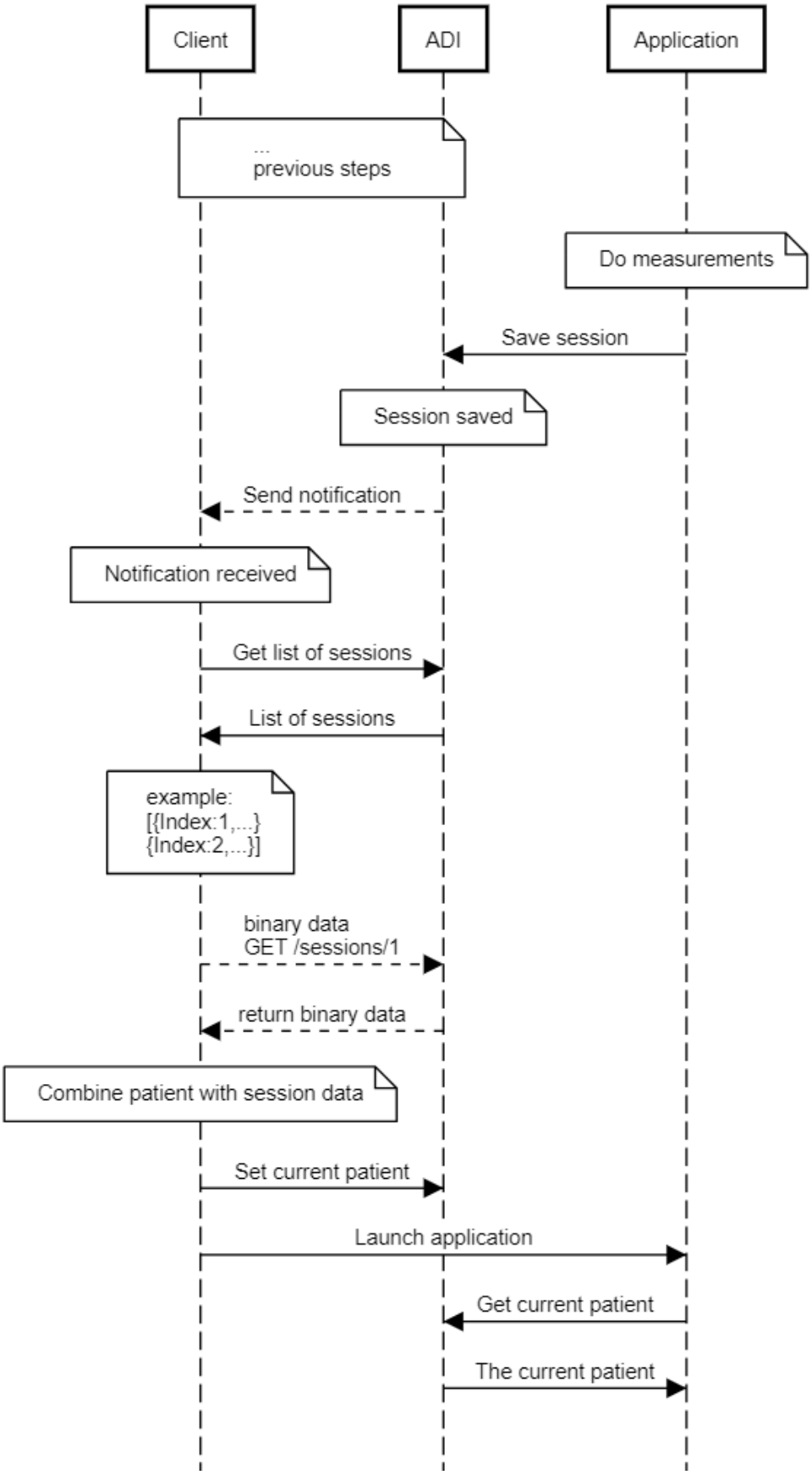


Figure 8 Sending saved session back to application

12 Working with standalone instruments

12.1 Connect/Disconnect instruments

12.1.1 Description

The ADI is also working well with standalone istruments. Supported instruments are listed in section 4.1. After plug in via cable, it should be immidately visible on the list of instruments GET /instruments. In order to access patients and sessions, it is required to connect to the instrument. You can send POST request under /instruments/{index}/connect (to disconnect use /instruments/{index}/disconnect).

Swagger doc:
http://URL/help/index#!/Instruments/Instruments_GetInstruments
http://URL/help/index#!/Instruments/Instruments_Connect
http://URL/help/index#!/Instruments/Instruments_Disconnect

12.1.2 Sequence diagram

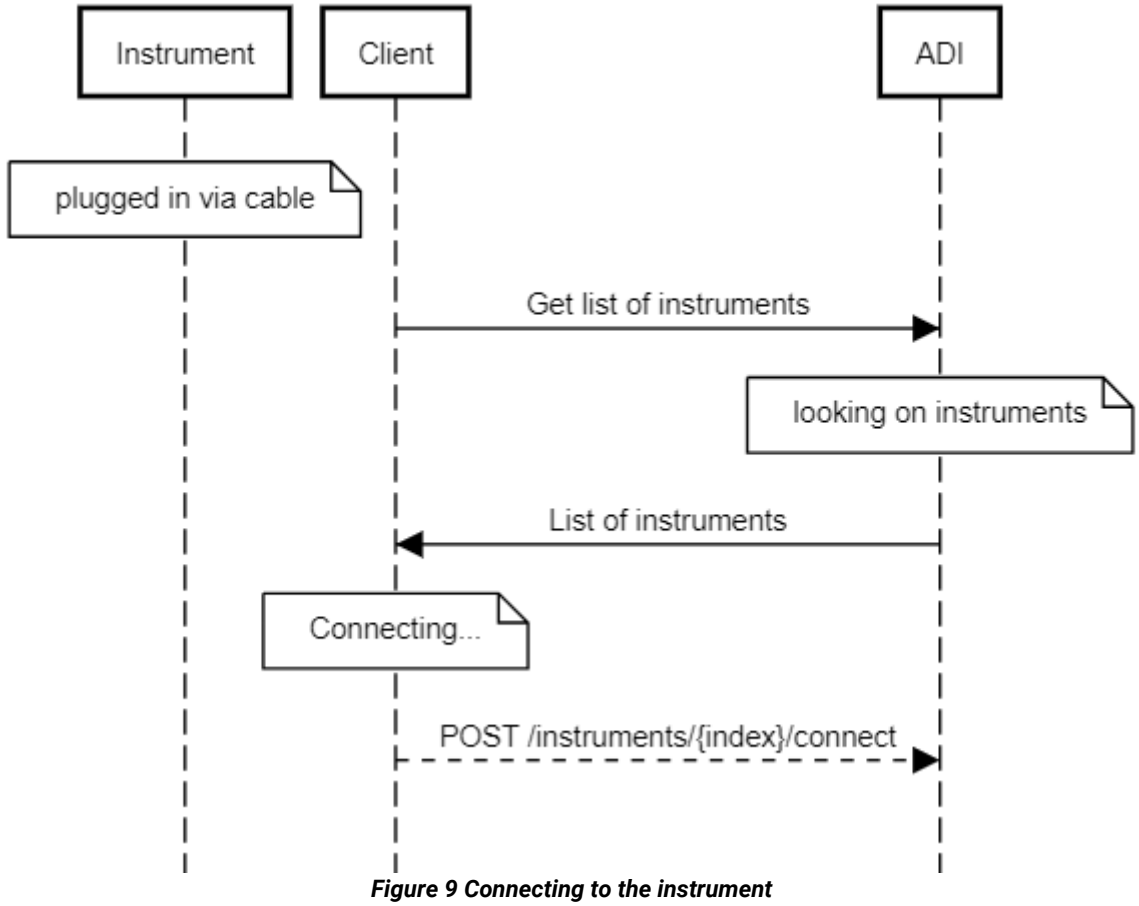


Figure 9 Connecting to the instrument

12.2 Add/Remove patients from instruments

12.2.1 Description

After connecting the instrument (see section 14.1), you can check which patients are currently on the instrument by query ADI via GET /instruments/{index}/patients. It is also possible to set own patient via POST /instruments/{index}/patients (to remove patient send DELETE /instruments/{instrumentIndex}/patients/{patientIndex}.

Swagger doc:
http://URL/help/index#!/Instruments/Instruments_GetPatients
http://URL/help/index#!/Instruments/Instruments_AddPatient
http://URL/help/index#!/Instruments/Instruments_DeletePatient

12.2.2 Sequence diagram

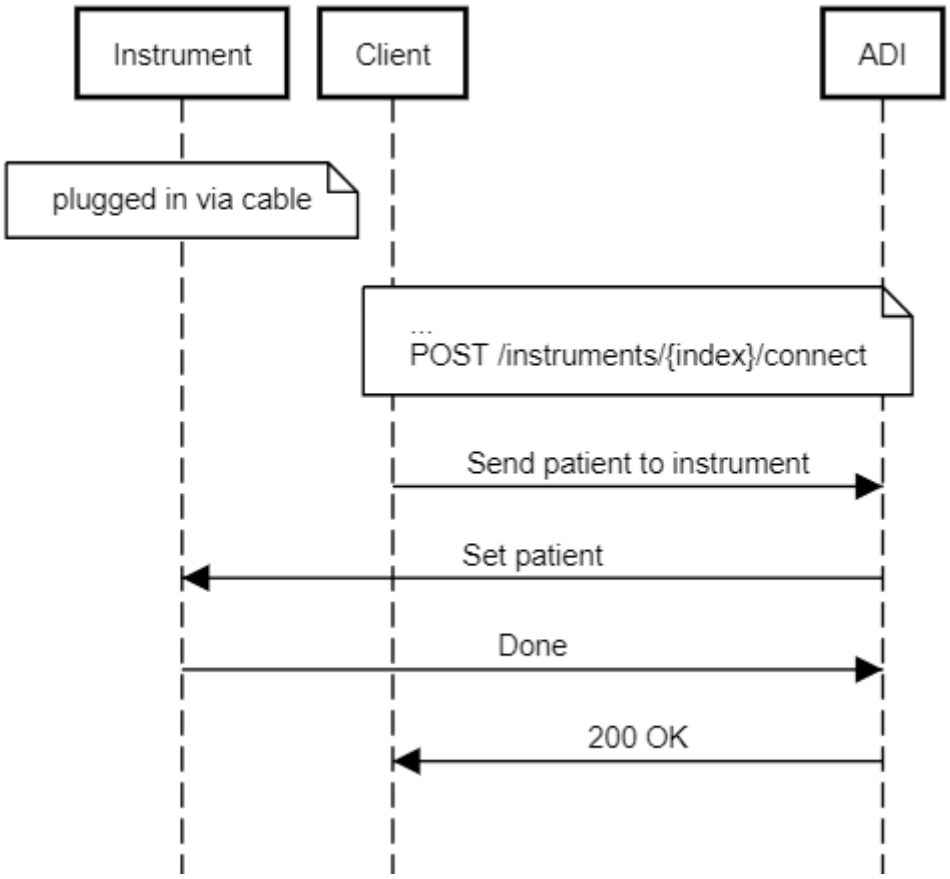


Figure 10 Add patient to the instrument

12.3 Get sessions

12.3.1 Description

The sessions are available by query ADI via GET /instruments/{instrumentIndex}/{patientIndex}/sessions. Please note that the instrument has to be connected and patient is accessible at given index (see section 14.1 and 14.2).

Swagger doc:
http://URL/help/index#!/Instruments/Instruments_GetSessions

12.3.2 Sequence diagram

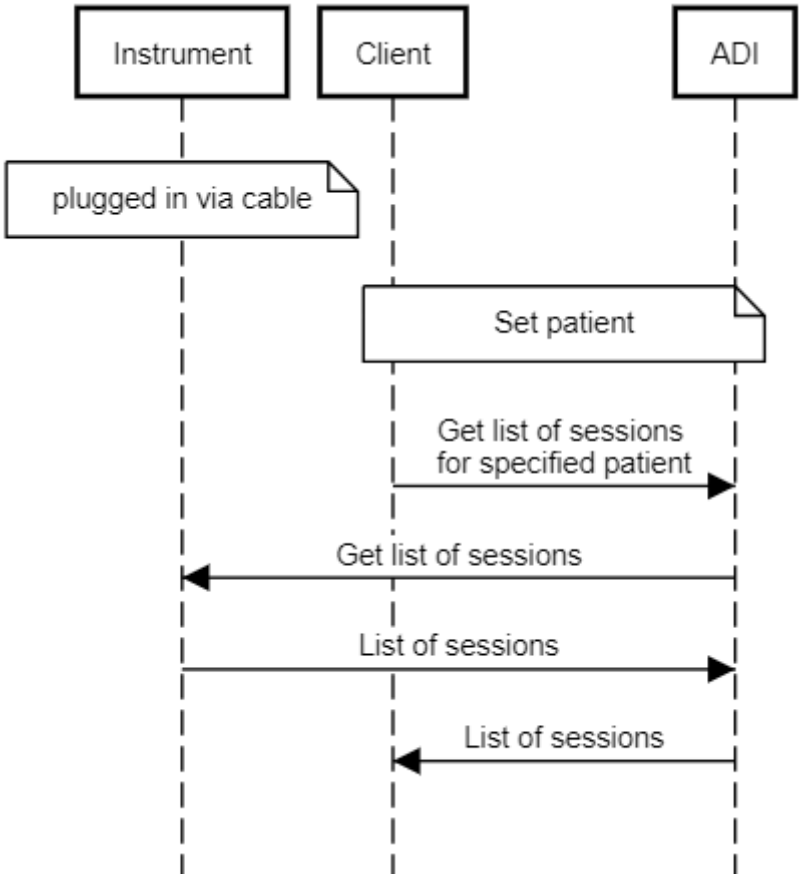


Figure 11 Get list of sessions for specified patient

12.4 Get tests data

12.4.1 Description

The tests data which is stored in the session can be retrived by query ADI via /instruments/{instrumentIndex}/patients/{patientIndex}/sessions/{sessionIndex}/tests. The requirements are that the instrument is connected, and patient with sessions is set.

Swagger doc:
http://URL/help/index#!/Instruments/Instruments_GetTests

12.4.2 Sequence diagram

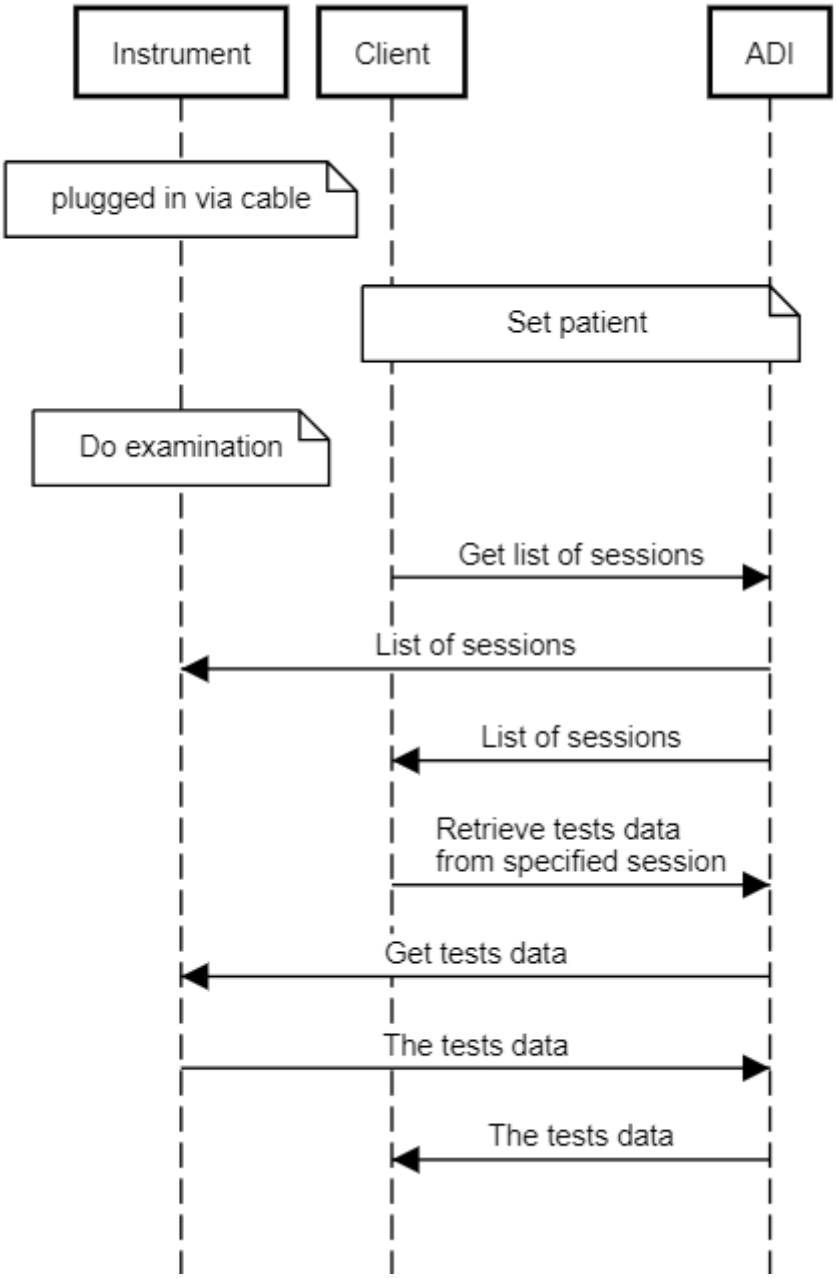


Figure 12 Get tests data

13 Setting clinic and examiner

13.1 Description

The client is responsible for setting clinic and examiner information. In order to do that, please POST data under /clinic and /examiner.

Swagger doc:
http://URL/help/index#!/Clinic/Clinic_Get
http://URL/help/index#!/Clinic/Clinic_Post
http://URL/help/index#!/Clinic/Clinic_Delete

http://URL/help/index#!/Examiner/Examiner_Get
http://URL/help/index#!/Examiner/Examiner_Post
http://URL/help/index#!/Examiner/Examiner_Delete

13.2 Sequence diagram

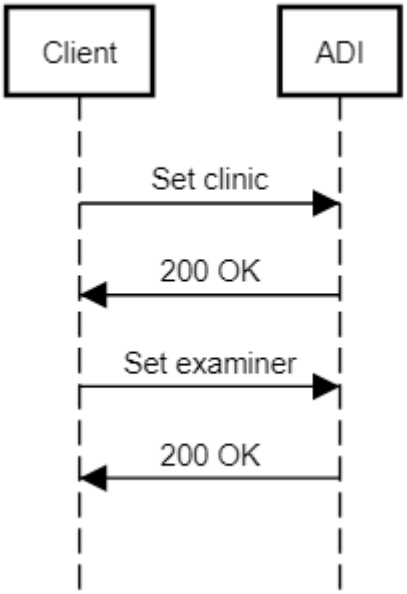


Figure 13 Setting clinic and examiner