H2020-ICT-688712

Project: H2020-ICT-688712

Project Name:
5G Applications and Devices Benchmarking (TRIANGLE)

Deliverable D5.8

Recommended TRIANGLE marks for certification of service

Date of delivery:	31/12/2018	Version:	1.0
Start date of Project:	01/01/2016	Duration:	36 months

Deliverable D5.8

Deliverable Title

Project Number:	ICT-688712
Project Name:	5G Applications and Devices Benchmarking
Project Acronym	TRIANGLE

Document Number:	ICT-688712-TRIANGLE/D5.8
Document Title:	Recommended TRIANGLE marks for certification of service
Lead beneficiary:	University College London
Editor(s):	Germán Corrales Madueño (KEYD), Niall Roche (UCL)
Authors:	Keysight Technologies Belgium (Michael Dieudonne), Keysight Technologies Denmark (German Corrales Madueño, Hua Wang), Universidad de Malaga (Almudena Díaz, Pedro Merino, Bruno García), Redzinc Services Limited (Donal Morris, Laura Dinz, Pilar Rodríguez), DEKRA Testing and Certification S.A.U (Janie Baños, Oscar Castañeda).
Dissemination Level:	PU
Contractual Date of Delivery:	31/12/2018
Work Package Leader:	Niall Roche (UCL)
Status:	Final
Version:	1.0
File Name:	TRIANGLE_Deliverable_D5.8_FINAL

Abstract

The purpose of this deliverable is to document the process used for the TRIANGLE certification; illustrating the steps required for both device and mobile application certification. The device certification process is described using the BlueEye device from RedZinc, while the app certification flow is explained for a well-known content streaming application (Exoplayer).

Keywords

TRIANGLE Certification, BlueEye, Exoplayer.

Executive summary

The primary objective of the TRIANGLE project is to promote the testing and benchmarking of mobile applications and devices as the industry moves towards 5G and to provide a pathway towards certification in order to support qualified apps and mobile developments using FIRE testbeds as testing framework'.

The purpose of certification is to give vendors of devices or applications a level of benchmark related to their performance of an application. The idea came from energy efficiency certification for domestic appliances and has been applied generally for 5G devices and applications.

Any vendor wishing to certify its 5G device or application can engage with the testbed owners regarding certification from TRIANGLE indicating the product functionality.

The TRIANGLE certification provides an independent process to measure the performance in terms of Quality of Experience (QoE) of a mobile app or device. The TRIANGLE mark provides a performance score, which is quantified according to certain domains or evaluated features, for the app or device under test.
In this deliverable we document the process to certify the BlueEye wearable video platform for first responders (i.e., device certification) and a content streaming application (i.e., app certification). The BlueEye device sends video from a glasses-based camera to the to a viewer in the hospital allowing interaction between emergency doctor and paramedic. Media interaction is based on duplex audio and simplex video.

In this document described the certification process, defining use cases and test cases. The configuration is also given in chapter 2 along with implementation of control points of observation and control as well as a description of the execution of the certification campaign and final report.

Chapter 3 contains the BlueEye device and Exploplayer app certification results. This includes the detail of the test cases, user flow for each test case, measurement points, the creation and launch the test campaign and results.

The TRIANGLE certification provides an independent process to measure the performance in terms of Quality of Experience (QoE) of a mobile app or device, leading to a performance score for applicable domains. This document outlined the procedure used to award the TRIANGLE certification to mobile applications and devices.
This document outlined a summary of the TRIANGLE certification process, with the more detailed description of the process to award the TRIANGLE Mark available from D2.2. The TRIANGLE testbed provides a set of test cases configured according to the defined use cases and under specific test scenarios. The results of these tests are compared with a list of key performance indicators (KPIs) that will be used to build the TRIANGLE mark.
In order to evaluate the certification process, a sample mobile application and device were used to test the certification process with real data. These samples included the ExoPlayer mobile video player and the BlueEye device from RedZinc.
The process used to certify this app and device, along with tests performed the certification results are outlined in section 3. The computed QoE scores in each measured domain for each sample are presented as well as the final TRIANGLE mark.

Contents

List of Figures

List of Tables

List of Abbreviations

AP	Access Point
APNet	Antennas, Propagation and Radio Networking
BER	Bit Error Rate
BLER	Block Error Rate
BS	Base Station
CAPEX	CApital EXpenditure
CDMA	Code Division Multiple Access
CFO	Carrier Frequency Offset
CO	Confidential
CP	Cyclic Prefix
CR	Cognitive Radio
CRS	Cognitive Radio Systems
CSI	Channel State Information
CSMA	Carrier Sense Multiple Access
C2X	Car-to-Anything
D	Deliverables
DL	Downlink
D2D	Device-to-Device
DMRS	Demodulation reference signal
DRX	Discontinuous Reception
DTX	Discontinuous Transmission
EIRP	Effective Isotropic Radiated Power
EIT	European Institute for Innovation and Technology
E2E	End-to-End
EVM	Error Vector Magnitude
FDD	Frequency Division Duplex
FD-MIMO	Full-Dimension MIMO
FEC	Forward Error Correction
FR	Frequency Response
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications
HARQ	Hybrid Automatic Repeat Request
ICI	Inter-Carrrier Interference
ICT	Information and Communications Technology
IEEE	Institute of Electrical and Electronics Engineers
IMT	International Mobile Communications
IP	Intellectual Property
IPR	Intellectual Property Rights
IR	Internal report
ITU	International Telecommunication Union
ITU-R	International Telecommunication Union-Radio
KPI	Key Performance Indicator
LAN	Local Area Network
LOS	Line of Sight
LTE	Long Term Evolution
LTE-A	Long Term Evolution-Advanced

L2S	Link to System
M	Milestones
Mbps	megabits per second
Mo	Month
MA	Multiple Access
MAC	Medium-access Control
MGT	Management
MIMO	Multiple-Input Multiple-Output
MMC	Massive Machine Communication
M2M	Machine to Machine
MSE	Mean Squared Error
NLOS	Nonline of Sight
OFDM	Orthogonal Frequency Division Multiplexing
OPEX	Operational Expenditure
PA	Power Amplifier
PAPR	Peak-to-Average-Power-Ratio
PC	Project Coordinator
PHY	Physical Layer
PU	Public
QAM	Quadrature Amplitude Modulation
QAP	Quality Assurance Plan
QMR	Quarterly Management reports
QoE	quality of experience
QoS	Quality of Service
RACH	Random Access Channel
RAN	Radio Access Network
RAT	Radio Access Technology
RF	Radio Frequency
R&D	Research and Development
RRM	Radio Resource Management
RTD	Research and Technological Development
RTT	Round Trip Time
SDR	Software Defined Radio
SINR	Signal to Interference and Noise Ratio
SRS	Sounding Reference Signal
T	Task
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TRX	Transmitter
TTI	Transmission Time Interval
UE	User Equipment
UL	Uplink
UMTS	Universal Mobile Telecommunications System
USRP	Universal Software Radio Peripheral
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-anything
WCDMA	Wide Code Division Multiple Access
WLAN	Wireless Local Area Network
WP	Work Package
WPAN	Wireless Personal Area Networks

1 Introduction

The TRIANGLE project is a 3 year Innovation Action project with a consortium composed of 6 partners and which is funded by the EU Commission under the Fire+ initiative. The project has developed a pre-5G testbed with the core capabilities to test, benchmark and certify app, devices and network configurations in a pre-5G environment.

The purpose of certification is to give vendors of devices or applications a level of benchmark related to their performance of an application. The idea came from energy efficiency certification for domestic appliances and has been applied generally for 5G devices and applications. The TRIANGLE certification process is formalised in Deliverable D2.2 and describes how a 5G application developer, device maker or operator can go about acquiring the TRIANGLE mark certification. Any vendor wishing to certify its 5G device or application can engage with the testbed owners regarding certification from TRIANGLE indicating the product functionality.

In this deliverable we document the process to certify the BlueEye wearable video platform for first responders (i.e., device certification) and a content streaming application (i.e., app certification). The BlueEye device sends video from a glasses-based camera to the to a viewer in the hospital allowing interaction between emergency doctor and paramedic. Media interaction is based on duplex audio and simplex video.

Chapter 2 in this document described the certification process, defining use cases and test cases. The configuration is also given in chapter 2 along with implementation of control points of observation and control as well as a description of the execution of the certification campaign and final report.

Chapter 3 contains the BlueEye device and Exploplayer app certification results. This includes the detail of the test cases, user flow for each test case, measurement points, the creation and launch the test campaign and results.

Chapter 4 contains the conclusions and chapter 5 includes references

2 Certification Process

As specified in D2.2 [5], 'The primary objective of the TRIANGLE project is to promote the testing and benchmarking of mobile applications and devices as the industry moves towards 5G and to provide a pathway towards certification in order to support qualified apps and mobile developments using FIRE testbeds as testing framework'.

The TRIANGLE certification provides an independent process to measure the performance in terms of Quality of Experience (QoE) of a mobile app or device. The TRIANGLE mark provides a performance score, which is quantified according to certain domains or evaluated features, for the app or device under test.

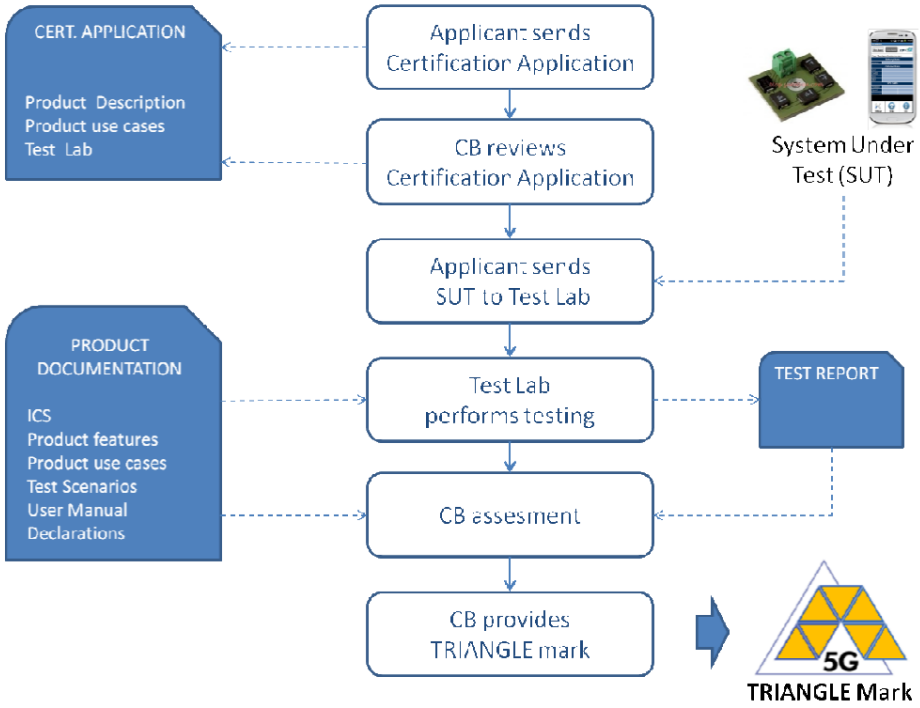

Figure 1: TRIANGLE certification process

This chapter details the TRIANGLE certification process that was initially presented in D2.2, providing exhaustive information about the steps to be performed to obtain the TRIANGLE mark. The TRIANGLE testbed provides a set of test cases configured according to the defined use cases and under specific test scenarios. The results obtained by a product after running TRIANGLE test cases determine its performance according to a list of key performance indicators (KPIs) that will be used to build the TRIANGLE mark. The scoring of the product performance is obtained per domain. Each domain is focused on a special characteristic of the product such as energy consumption.

Every domain assessed in the application or device is scored, and a global score is provided. In this way, products of different features are evaluated in an equivalent way, taking into account their performance in different domains specified by TRIANGLE.

2.1 General certification process

Any customer that wants to apply to certify a product shall start the certification process as detailed below:

1. The Customer (applicant) contacts TRIANGLE via email requesting to certify its product.

2. TRIANGLE assigns a Certification Body to the customer for the requested certification process.
3. The Certification Body requests relevant information about the product. TRIANGLE request the applicant to provide the following documentation
    - TRIANGLE Certification Application Form. This form contains basic product and certification information. The applicant also selects the Lab where the testing is going to be performed.
    - TRIANGLE ICS/IXIT template. This document provides the list of features, use cases and scenarios supported by the product (See Appendix 1 - TRIANGLE ICS/IXIT template)
    - Product user manual and/or a description explaining everything required to use the product for testing purposes.
4. The customer provides the information requested by the Certification.
5. The information is validated by TRIANGLE. Some iterations may be required until all the information required is available.
6. The Certification Body generates the test plan (list of test cases to be performed to obtain certification) and provides it to the TRIANGLE Lab and to the Applicant.
7. The TRIANGLE Lab indicates the steps to be followed for each test case and request the applicant the points of control and observation required (measurement points, actions, user flows, etc.).
8. The applicant implements those points of control and observation in its system.
9. Customer ships the sample to the TRIANGLE Lab or sends the App. A test engineer may optionally attend the certification testing to ease the process (to be agreed with the TRIANGLE Lab).
10. The TRIANGLE Lab runs a reduced campaign to make sure everything works properly.
11. Steps 7 to 10 iterate until the process runs smoothly.
12. The TRIANGLE Lab executes the certification campaign and delivers a report with the results of the campaign.
13. The Certification Body assesses the Test Report.
14. The Certification Body provides the TRIANGLE mark.

2.2 Define Use cases and Test Cases

TRIANGLE certification scheme defines a set of use cases, domains and a corresponding list of test cases to be performed.

Applicants need to identity which use cases from the list of TRIANGLE use cases, their product is targeting to. The use cases are different for different type of products (Apps, mobile devices and IoT devices). Anyway, some use cases are common to different type of products.

TRIANGLE has defined the following use cases depending on the type of product:

- Apps: Virtual Reality, Gaming, Augmented Reality, Content Distribution Streaming Services, Live Streaming Services, Social Networking and High Speed Internet.

- Mobile devices: Virtual Reality, Gaming, Augmented Reality, Content Distribution Streaming Services, Live Streaming Services, Social Networking, High Speed Internet, Emergency Services, Smart Metering, Smart Grids and Connected Vehicles.
- IoT devices: Patient Monitoring, Emergency Services, Smart Metering, Smart Grids and Connected Vehicles.

A product can support one or more use cases.

Each TRIANGLE domain is focused on a specific characteristic or feature of a product, such as user experience or data performance.

TRIANGLE defines the set of domains to be evaluated depending on the type of product to be certified:

- Apps domains are: Reliability, Network Resources Usage, User Experience, Devices Resources usage and Energy consumption.
- Mobile devices domains are: User Experience with Reference Apps, Data Performance, Radio Performance and Energy consumption.
- IoT devices domains are: Reliability, Data Performance, and Energy consumption.

TRIANGLE Test Specification is a document that compiles all test cases that have been defined by TRIANGLE to evaluate the performance of a product.

This Test Specification classifies the test cases depending on the type of product and the use cases (and accordingly the applicable domains) supported by the product. The ICS/IXIT proforma included in the document, allows applicants define the specific features of the product. Depending on the features indicated in the ICS proforma, TRIANGLE is able to determine the Certification Test Plan, i.e. the list of test cases to be executed for a product to be certified.

2.3 Product Configuration

Before an applicant starts the TRIANGLE process of a product, it is recommended that the applicant takes into consideration the following points in order to ease the certification process. Further details are provided in TRIANGLE test specification.

- The applicant shall supply all means required to perform the testing. E.g. If the product requires connection to a server for delivering its service, access to this server must be configured and granted from the TRIANGLE testbed.

If the product to be certified is a device (being a mobile device or an IoT device), the following considerations will also be taken into account.

- The device shall provide means to be connected to TRIANGLE testbed power analyzer, commonly by providing a dummy battery.
- The device shall provide means to connect to the TRIANGLE testbed RF connector.

2.4 Implementation of Points of Control and Observation

As indicated in step 8 of the procedure indicated in section 2.1, the applicant needs to perform some modifications in its product in order to be properly evaluated by TRIANGLE certification scheme.

The applicant performs this step following the indications given by the Lab. The applicant mainly needs to implement two types of items: Measurement points and user flows/actions//interfaces.

Measurements points are a library of simple methods which are inserted into the source code of the application under test. These measurement points are necessary to retrieve information needed for the calculation of certain measurements, and they can, in turn, be used to compute KPIs or other aggregated data.

To facilitate the insertion of measurement points, the TRIANGLE testbed has developed an instrumentation library that can be used by app developers to insert the required measurement points into their own apps.

The applicant shall implement a functionality that emulates the behaviour of the user of the product. Depending on the product to be certified this behaviour is emulated by an Application user flow, action or a Control Interface

- Apps: *Application User Flow*: It is a sequence of actions that the user's App is expected to perform on the App user interface objects in order to execute a test step of this test specification.
- Mobile devices: In the case of mobile devices, the applicant does not need any implementation as the user behavior is emulated by calling a Reference App API.
- IoT devices: IoT device Control interface: The applicant shall implement an interface that allows TRIANGLE testbed request the execution of the required actions by the IoT device when performing the test cases.

2.5 Execution of Certification Campaign and Final Report

Once the applicant has provided all required documentation and implemented the control and measurement points, the lab is ready to execute the certification test plan by running a certification campaign at TRIANGLE testbed that contains all the test cases for the test plan.

The certification test plan is executed, and the product scoring is obtained for each of the domains applicable to the product, leading to the global product score.

The lab will generate a Test Report, according to the template provided by TRIANGLE, containing the results and testing details related to the product evaluation.

This Test Report will be sent to the Certification Body that will perform the final assessment and issue the TRIANGLE mark as applicable.

3 Certification Results

Having outlined the certification process above in Section 2, the consortium used some sample mobile applications and devices to test and refine the certification process. These samples included the ExoPlayer mobile video player and the BlueEye device from RedZinc. The process used to certify this app and device, along with tests performed the certification results are outlined in the sections below.

3.1 Exoplayer Application Certification

To verify the measurement capabilities of the testbed and to benchmark the performance of some well-known applications, we selected ExoPlayer as a case study, which is an open source media player originally developed by Google with support for DASH and SmoothStreaming adaptive playbacks. This application (app) falls into the use case of content distribution streaming service defined in the TRIANGLE project. The steps required for application certification is listed in Figure 2, which will be detailed in the following subsections.

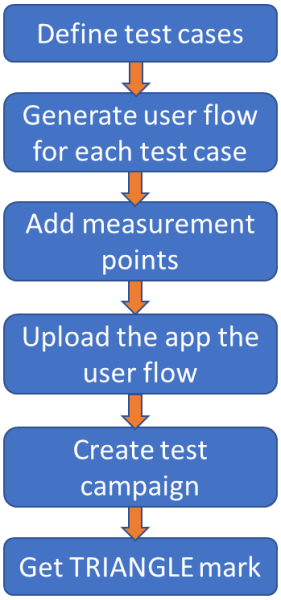

Figure 2: Application certification steps

3.1.1 ExoPlayer Test Cases and User Flows

There are 8 test cases defined in the TRIANGLE testbed for testing content distribution streaming service. In this case study, 6 out of 8 test cases are selected for testing ExoPlayer, which are listed in Table 1. The test case description specifies the test conditions, the generic app user flow, and the raw measurements, which shall be collected during the execution of the test. The complete description of each test case in different domains can be referred to D2.2.

Table 1. Simplified test case description

Test Case	KPIs in the AUE domain	KPIs in the AEC domain	KPIs in the RES domain
Non-Interactive Playback (CS/001)	• Time to load first media • Content stall • Video resolution	• Average power consumption	• Average CPU usage • Average memory usage

	• Playback cut-off		
Play and Pause (CS/002)	• Time to load first media • Playback cut-off • Pause operation	• Average power consumption	• Average CPU usage • Average memory usage
Stop and Replay (CS/003)	• Time to load first media • Playback cut-off • Stop operation	• Average power consumption	• Average CPU usage • Average memory usage
Rewind (CS/005)	• Time to load first media • Playback cut-off • Rewind operation	• Average power consumption	• Average CPU usage • Average memory usage
Download content for offline playing (CS/007)	• Media Download Operation • File Downloading Time	• Average power consumption	• Average CPU usage • Average memory usage
Fast Forward (CS/008)	• Time to load first media • Playback cut-off • Fast Forward operation	• Average power consumption	• Average CPU usage • Average memory usage

AUE: Applications User Experience

AEC: Applications Energy Consumption

RES: Applications Device Resources usage

3.1.1.1 Generate user flow for each test case

The user flow defines a series of user interactions with the application under test. For each selected test case, the user flow shall be defined according to the test specifications [5]. For example, the test specification for test case “Play and Pause” is specified in Table 2.

Table 2. Test specification for test case “Play and Pause”

Test case	Test specification
Play and Pause	1. Perform login step and wait for 10 seconds 2. Start playing for 10 seconds 3. Pause the play for 2 minutes 4. Resume the play for 2 minutes

For TRIANGLE testbed, the user flow shall be written in power shell script with the help of Quamotion tool, which is used to automate user interactions with the application, e.g., login, start/stop/pause/resume playing, etc. The advantage of using Quamotion to generate the user flow is that it can record the user interactions with the app and automatically convert it into power shell script.

The generated user flow using Quamotion in power shell script following the test specification listed in Table 2 is shown in Table 3. The “sleep” value defines the ilde time between the two actions.

Table 3. Generated user flow for test case “Play and Pause”

Test case	User flow
Play and Pause	Click-Element -xpath "TextView[@marked='text1'][1]" // select the video category sleep 10 Click-Element -xpath "TextView[@marked='sample_title'][3]" // click the video to play sleep 10 Click-Element -xpath "FrameLayout[@marked='exo_overlay']" // tap the screen sleep 1 Click-Element -xpath "ImageButton[@marked='exo_pause']" // click “Pause” sleep 120 Click-Element -xpath "ImageButton[@marked='exo_play']" // click “Play” sleep 120 Go-Back

Similarly, the user flow for each other selected test case shall be generated in power shell script according to the test specifications.

3.1.2 Exoplayer Product Configuration

As application developer the product configuration is rather simple, just need to indicate in which device the app will run. In this case, the app was installed in a S7 already offered by the testbed and therefore no actions were required.

3.1.3 Add Measurement Points

To enable the collection of the KPIs of interest, measurement points should be inserted into the source code of the app. The TRIANGLE testbed has developed an instrumentation library that offers various measurement points. These measurements are used to compute relevant KPIs or other aggregated data.

The information of the required measurement points will be indicated by selecting the test case of interest for the app under test. Table 4 shows the measurement points required in each selected test case to calculate the relevant KPIs.

Table 4. Test cases and the required measurement points

Test case	Measurement points
Non-Interactive Playback	Media File Playback - Start Media File Playback – End Media File Playback – First Picture Media File Playback – Video Resolution Media File Playback – Content Stall Start Media File Playback – Content Stall End
Play and Pause	Media File Playback – Start Media File Playback – End Media File Playback – First Picture Media File Playback – Pause Media File Playback – Resume
Stop and Replay	Media File Playback – Start Media File Playback – End Media File Playback – First Picture Media File Playback – Stop
Rewind	Media File Playback – Start Media File Playback – End Media File Playback – First Picture Media File Playback – Rewind
Download content for offline playing	Media Content Download - Start Media Content Download - End
Fast Forward	Media File Playback – Start Media File Playback – End Media File Playback – First Picture Media File Playback – Fast Forward

Table 5 shows the method associated with each measurement point. These methods are inserted into the source code of ExoPlayer for collecting the measurements. An example of measurement point insertion of “Media File Playback – Fast Forward” in ExoPlayer is shown in Figure 3. As can be seen from the figure, adding the measurement point into the source code is rather simple as there is only one line of code (highlighted) added. The key point in this process is to understand the source code and pin point the correct location to insert the measurement point methods.

Table 5. Measurement points and the associated methods

Measurement points	Measurement point methods
Media File Playback – Start	eu.TRIANGLE_project.appinstr.cs.MediaFilePlayback.mediaFilePlaybackStart()
Media File Playback – End	eu.TRIANGLE_project.appinstr.cs.MediaFilePlayback.mediaFilePlaybackEnd()
Media File Playback – First Picture	eu.TRIANGLE_project.appinstr.cs.MediaFilePlayback.mediaFilePlaybackFirstPicture()
Media File Playback – Video Resolution	eu.TRIANGLE_project.appinstr.cs.MediaFilePlayback.mediaFilePlaybackVideoResolution (<resolution_x>, <resolution_y>)
Media File Playback – Content Stall Start	eu.TRIANGLE_project.appinstr.cs.MediaFilePlayback.mediaFilePlaybackContentStallStart()
Media File Playback – Content Stall End	eu.TRIANGLE_project.appinstr.cs.MediaFilePlayback.mediaFilePlaybackContentStallEnd()
Media File Playback – Pause	eu.TRIANGLE_project.appinstr.cs.PlayAndPause.mediaFilePlaybackPause (<success>)
Media File Playback – Resume	eu.TRIANGLE_project.appinstr.cs.PlayAndPause.mediaFilePlaybackResume (<success>)
Media File Playback – Stop	eu.TRIANGLE_project.appinstr.cs.StopAndReplay.mediaFilePlaybackStop (<success>)
Media File Playback – Rewind	eu.TRIANGLE_project.appinstr.cs.RewindandFastForward.mediaFilePlaybackRewind (<success>)
Media File Playback – Fast Forward	eu.TRIANGLE_project.appinstr.cs.RewindandFastForward.mediaFilePlaybackFastForward (<success>)
Media Content Download – Start	eu.TRIANGLE_project.appinstr.cs.DownloadMediaContentForOfflinePlaying. mediaContentDownloadStart()

Media Content Download – End	eu.TRIANGLE_project.appinstr.cs.DownloadMediaContentForOfflinePlaying. mediaContentDownloadEnd(<success>)

```
private void rewind() {
  eu.triangle_project.appinstr.cs.RewindandFastForward.mediaFilePlaybackRewind(true);
  if (rewindMs <= 0) {
    return;
  }

  seekTo(Math.max(player.getCurrentPosition() - rewindMs, 0));
}

private void fastForward() {
  eu.triangle_project.appinstr.cs.RewindandFastForward.mediaFilePlaybackFastForward(true);

  if (fastForwardMs <= 0) {
    return;
  }
  long durationMs = player.getDuration();
  long seekPositionMs = player.getCurrentPosition() + fastForwardMs;
  if (durationMs != C.TIME_UNSET) {
    seekPositionMs = Math.min(seekPositionMs, durationMs);
  }
  seekTo(seekPositionMs);
}

private void seekTo(long positionMs) {seekTo(player.getCurrentWindowIndex(), positionMs);
```

Figure 3: Example measurement point insertion of "Media File Playback – Fast Forward" in ExoPlayer.

Once the test campaign is finished, the relevant KPIs are automatically calculated by postprocessing the measurement points. For example, The KPI of "time to load first media picture" is obtained by subtracting the timestamp of the measurement point "Media File Playback – Start" from the measurement point "Media File Playback – First Picture".

3.1.3.1 Upload the app and the user flow to the TRIANGLE portal

Once the measurement points have been added into the source code of the app, the app shall be uploaded into the TRIANGLE portal.

After uploading the app, the user can select the use case of interest. There are 8 use cases defined in the testbed. In this case, the appreciate use case is Content Distribution Streaming Services. Within the use case of Content Distribution Streaming Services, the user can select the test cases of interest. In this case, 6 test cases have been selected. The generated user flow for each selected test case shall be uploaded.

3.1.4 Create and Launch the Certification Campaign

The test campaign is created with the configurations listed in Table 6.

Table 6. Test configurations of application under test

Application under test	ExoPlayer
Test reference device	Samsung Galaxy S7
Use case	Content Distribution Streaming Services
Test case	Non-Interactive Playback Play and Pause Stop and Replay Rewind Download content for offline playing Fast Forward
Test type	Certification campaign
Scenario	All supported scenarios (11)
Number of iterations	15

3.1.5 Test results

The obtained synthetic MOS values for test case “Non-Interactive Playback” per scenario in different domains are listed in Table 7. The results for other test cases are obtained in similar tables but are not shown here due to the space limitations.

Table 7. Synthetic MOS score for test case “Non-Interactive Playback”

Scenario	AUE domain				AEC domain	RES domain	
	Test Case AUE/CS/001				Test Case AEC/CS/001	Test Case RES/CS/001	
	Time to load first media frame	Playback Cut-off ratio	Video Resolution mode	Content stall	Average Power Consumption	Average CPU Usage	Average RAM Usage
HighSpeed Direct Passenger	2.7	3.7	3.0	1.6	3.7	4.2	3.3
Suburban Festival	3.2	4.1	3.2	3.1	3.4	4.0	3.3
Suburban shopping mall busy hours	2.3	3.5	2.6	1.9	4.2	4.6	3.3

Suburban shopping mall off-peak	3.6	4.7	3.8	3.9	3.8	4.2	3.2
Suburban stadium	3.0	3.8	3.2	2.4	3.5	4.0	3.3
Urban Driving Normal	3.2	4.3	4.0	3.1	3.4	4.0	3.3
Urban Driving Traffic Jam	3.0	3.8	3.0	3.3	3.7	4.0	3.2
Urban Internet Café Busy Hours	3.1	3.6	3.2	3.6	3.8	4.2	3.3
Urban Internet Café Off Peak	3.7	4.8	4.3	4.1	3.4	3.9	3.2
Urban Office	3.8	4.5	4.4	4.3	3.3	3.8	3.3
Urban Pedestrian	3.7	4.4	4.2	4.4	3.4	3.9	3.2

The final score in each domain is obtained by averaging synthetic MOS values over the tested scenarios and the related KPIs. The final TRIANGLE mark is calculated as a weight average of the three tested domains. Table 8 shows the obtained QoE score in each domain and the final TRIANGLE mark. In addition, the results as shown in the TRIANGLE portal are show in Figure 4.

Table 8. ExoPlayer TRIANGLE Mark

Domain	Score (/5)
Application User Experience	4.09
Application Energy Consumption	3.75
Application Resource Usage	4.16
TRIANGLE MARK	**4.00**

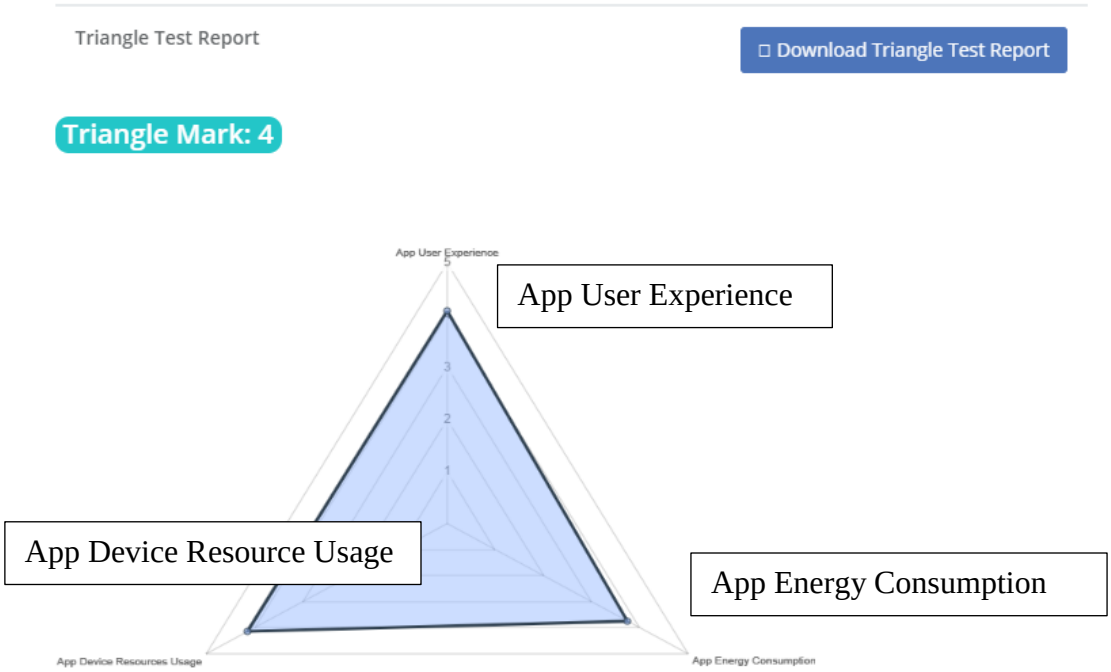

Figure 4: TRIANGLE Portal Score Overview

In this section of ExoPlayer application certification, we have described the certification process which consists of define use cases and test cases, generate user flows, add measurement points, and run certification campaigns. The computed QoE scores in each measured domain as well as the final TRIANGLE mark will be given in the certification report. In our case, we can say that Exoplayer has been certififed with the TRAINGLE mark showing a good mark of 4.00. The scoring maps ratings between Bad and Excellent to numbers between 1 and 5, as depicted in the table below:

Rating	*Label*
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

3.2 BlueEye Device Certification

In Annex 9 of the deliverable D2.2, the BlueEye was present as the alpha customer of the TRIANGLE testbed for IoT devices as indicated in section 12 of [1]. Into it, is exposed the main steps of the TRIANGLE certification process, addressed the Certification Application or the ICS/IXIT or TRIANGLE TCRL applicable to the BlueEye case.

3.2.1 BlueEye Use Cases and Test Cases

The domains applicable to BlueEye as IoT device are listed in Table 9. In addition, the use cases that applies to the BlueEye is Emergency Services as depicted in Table 10.

Table 9. BlueEye applicable TRIANGLE domains

System under Test	Domain	Identifier	Test Specification
IoT device	Reliability	IDR	IoT Devices Reliability
IoT device	Data Performance	IDP	IoT Devices Data Performance
IoT device	Energy Consumption	IEC	IoT Devices Energy Consumption

Table 10. BlueEye applicable TRIANGLE use cases

Identifier	Use Case
ES	Emergency Services

There are 11 test cases defined in the TRIANGLE testbed for IoT devices applicable to BlueEye. The test case description specifies the test conditions and all the measurements which shall be collected during the execution of the test. The complete description of each test case in different domains can be referred to D2.2. The exact list can be found in Table 11.

Table 11. BlueEye applicable test cases

Test Case ID	Description	Measurement
IEC/CO/001	Energy Consumption. IUT in idle mode	Current consumption
IEC/ES/001	Energy Consumption. Send video streaming	Current consumption
IEC/ES/004	Energy Consumption. Receive audio streaming	Current consumption
IDP/CO/001	Data Performance. IUT in idle mode	DL IP Data UL IP Data
IDP/ES/001	Data Performance. Send video streaming	DL IP Data UL IP Data
IDR/ES/001	Reliability. Send video streaming	Playback Availability Playback Cutoff

IDR/ES/004	Reliability. Receive audio streaming	Playback Availability Playback Cutoff
IDR/ES/006	Reliability: Connection failure	Playback Availability Playback Cutoff Recovery Time

Each test case will be executed on all scenarios applicable for BlueEye supported use cases. According to the annex B of D2.2, [3] the applicable scenarios shall be:

Table 12. BlueEye testing scenarios

SC	Scenario	Emergency Services
UR-OF	Urban-Office	Y
UR-DE	Urban-Driving-Emergency driving	Y
SU-FE	Suburban-Festival	Y
SU-ST	Suburban-Stadium	Y
SU-SB	Suburban-Shopping Mall, Busy Hours	Y
SU-SO	Suburban-Shopping Mall, Off-Peak	Y

3.2.2 BlueEye Product Configuration

Connecting the BlueEye to the testbed was relatively simple. The BlueEye device has two RF ports (main and diversity) which were connected to the RF switch in the TRIANGLE testbed. The process to replace the battery with the power analyzer provided by the testbed was not so straightforward. The BlueEye device packs a 50W battery, enough to withstand several hours operation if needed. However, the testbed was equipped with a 20W power analyzer. Several tests were carried out to assure the BlueEye device was not limited by the power analyzer and could perform as expected.

3.2.3 BlueEye Points of Control and Observation

As indicated before, for IoT devices, the applicant shall implement an interface that allows TRIANGLE testbed request the execution of the required actions by the IoT device when performing the test cases.

For a proper development of the testing process, RedZinc as customer and manufacturer of BlueEye solution provided an API which allows controlling the device from the testbed. In addition, RedZinc also implemented a TAP plugin which exposes the API as TAP steps, simplifying the integration process into the TRIANGLE Testbed.

The next sections expose the interaction between TAP and the IoT device or DUT. The plugin for TAP (Test Automation Platform), contents the steps that communicate to the API what functionalities of BlueEye will be executed during each Test Case.

3.2.3.1 BlueEye API

BlueEye transmits video in real time to the Hospital application. To enable automated testing, an API will be used to connect to a socket running on the target, which can be either a Blue-Eye Unit or a Hospital station. The architecture of the system is depicted in Figure 5.

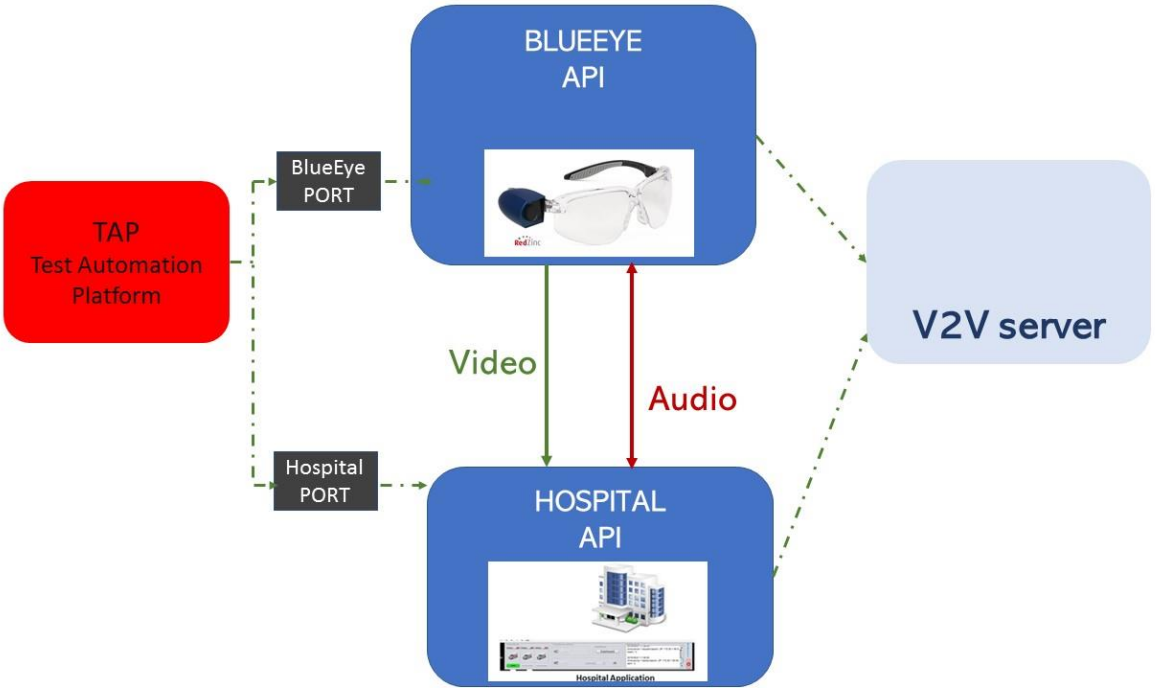

Figure 5: BlueEye and TAP iteration

The API is based on a set of JSON encoded requests and responses over a TCP connection. All connections are to be permanent as needed, with all requests being blocking (the socket will need to finish a request before starting another). There will be a single connection between a client (and a single client at a time) and a daemon and this daemon is considered without the need for concurrent requests. More detail about the API can be found in Appendix 2 – BlueEye API Details. Suffice to say the API exposes the following actions/functionality to the testbed:

- Software Reboot
- Start Streaming to Hospital
- Stop Streaming to Hospital
- Start CPU/RAM logging
- Start Wireshark Logging
- Stop Wireshark Logging
- Move Logs to the Repository

3.2.3.2 BlueEye Plugin for TAP

The TAP, Test Automation Platform, is a software platform for fast and easy development of a testing process that allows controlling instruments and customer-specific devices under test (DUTs). RedZinc implemented a plugin in TAP that exposes the API functionality. This step is not always required, as the Lab could implement the TAP plugin which controls the API of a given customer.

The BlueEye plugin is implemented in form of steps and Instruments and DUTs.

BE Instrument

Into the Instrument class, the BEInstrument1 is developed to represent either the BlueEye device or Hospital application control by means of a TCP connection.

Although the Hospital application is out of the scope of the BlueEye certification process to obtain the TRIANGLE mark, we use it in some steps, for capturing the traffic from the BlueEye by the Wireshark analyzer.

The Instrument class extends the resources to the steps. It includes the Open method which is executed before to start the Test plan and includes the necessary code to configure the DUT. The Close method is called after the Test plan is completed and contents all the necessary code to configure the DUT to a safe condition after testing. The Figure 6, describes the main DUT setting and the actions compressed in the Instrument methods.

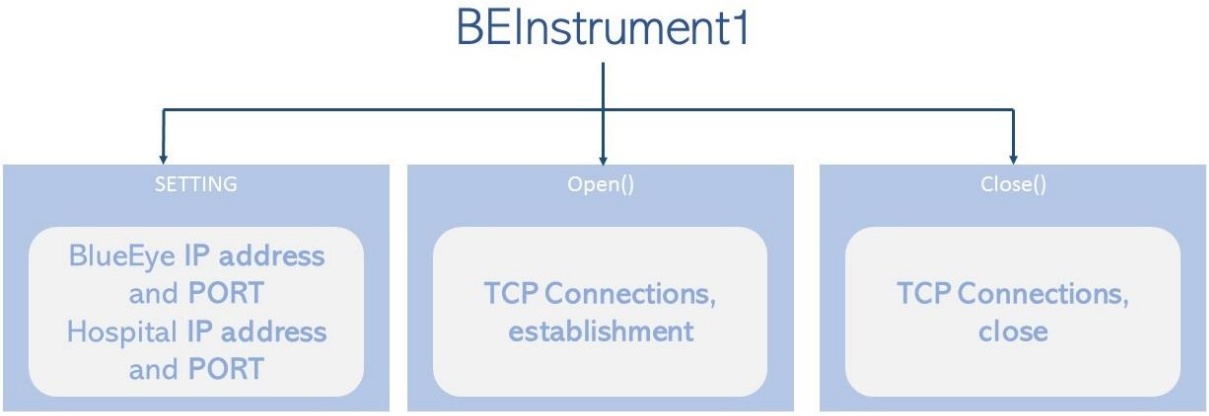

Figure 6: BEInstrument1

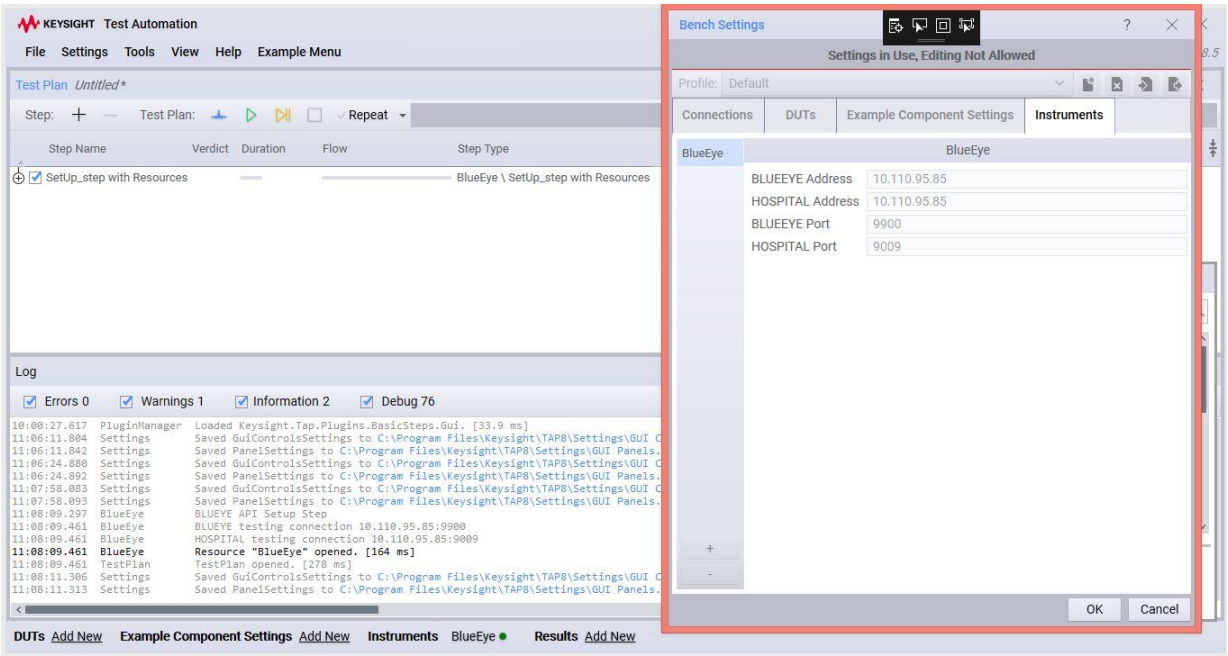

Figure 7: BEInstrument1 GUI

BlueEye Steps

The test step plugin is developed by extending the TestStep base class. The Figure 8, shows the Steps List implemented for BlueEye Testing process. A step describes an action or functionality that BlueEye will execute during the Test Case in which this step is involved.

To minimize the code and share the resources or features between the different steps some Parent/Child relationship are defined. In our case, the goal is that BEInstrument1 and its resources are reachable for the steps. In this case, the step “SetUp_step” is defined as a Parent which provides access to all the rest of the steps.

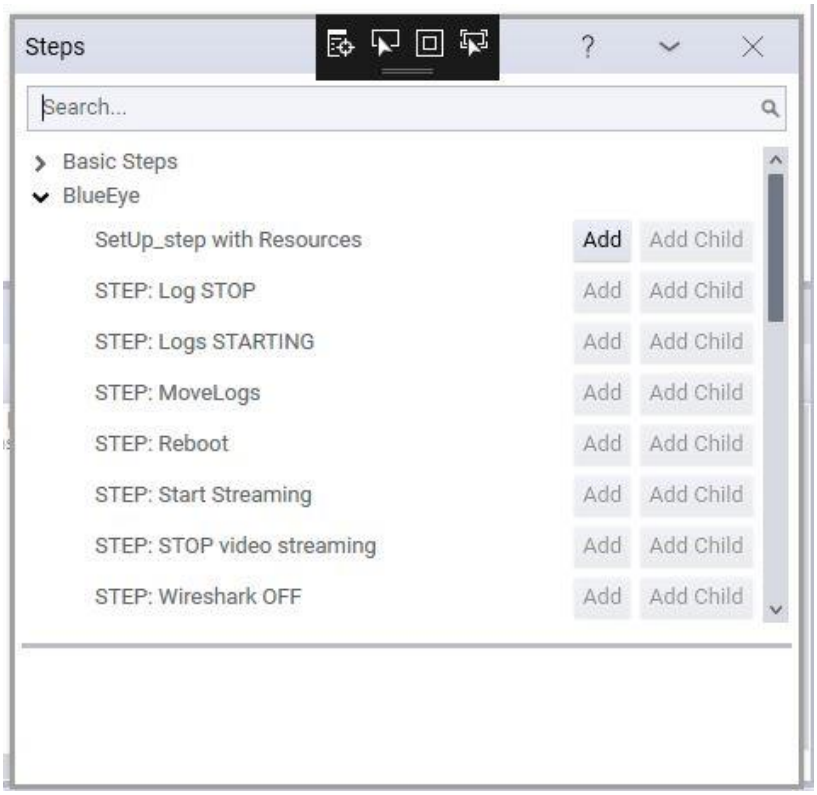

Figure 8: BlueEye steps list

From the TAP interface, we can establish how many steps participate in the Test Case and the sequence. Doing click on the step, you can see the step settings and configure them in the

"Step setting" window, in the same way, in the "Steps" window you can see a brief description of each step in Figure 9.

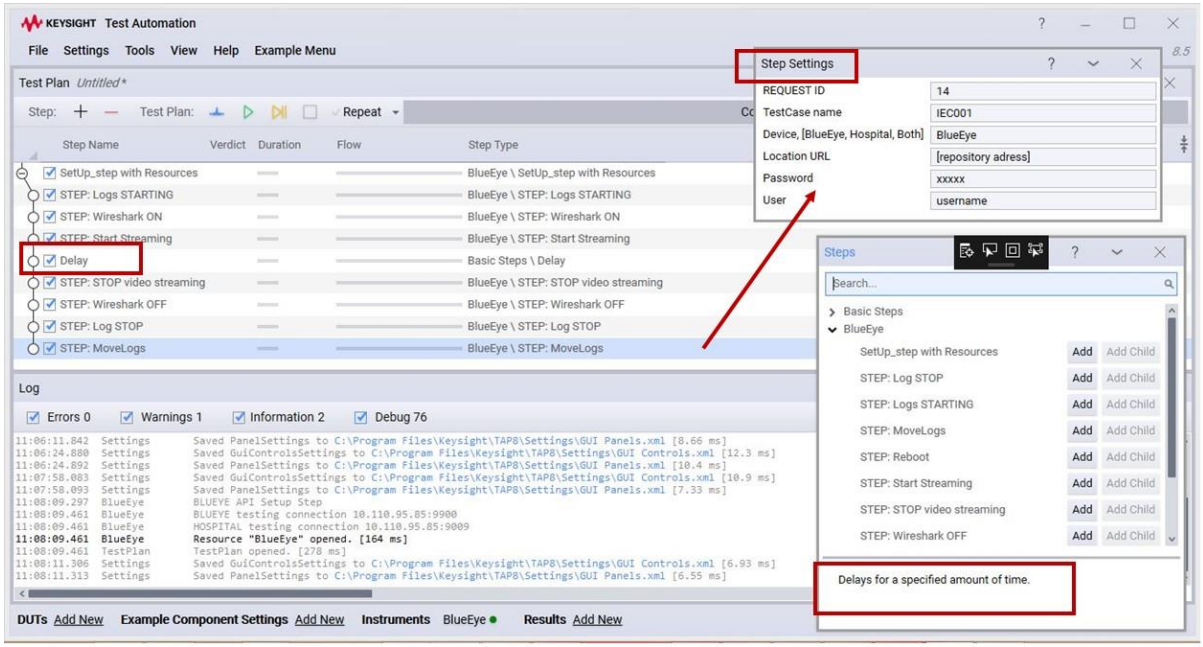

Figure 9: Steps, sequence and settings

3.2.4 Execution of Certification Campaign and Final Report

The purpose of this section is two-fold: (i) to illustrate how the certification was run and (ii) to show the results obtained. Basically, the IoT device is connected to the Test System though a conducted RF connection. In addition, the DC Power Source and Current Measurement device provides DC power source to the IoT Device and measures the IoT Device current consumption. The simplified overview of the testing architecture is depicted in Figure 10.

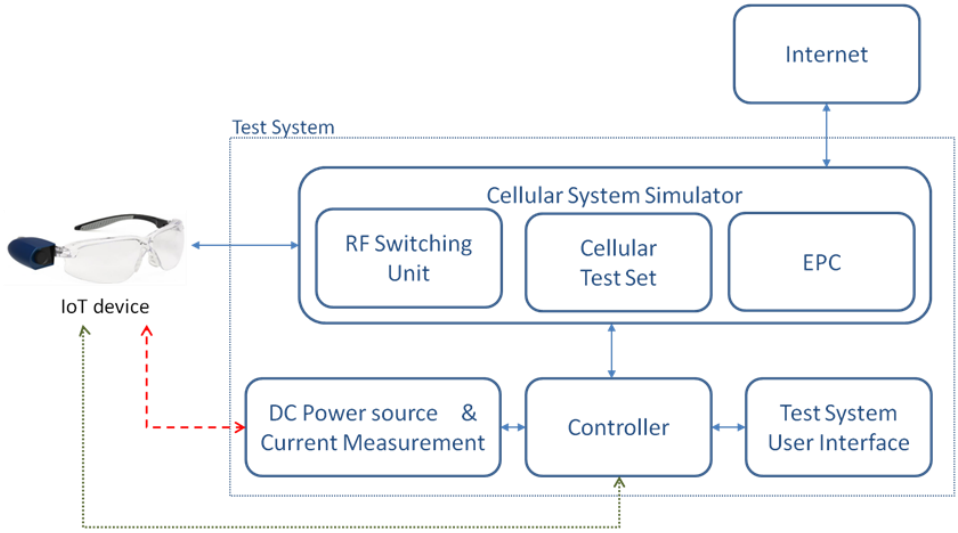

Figure 10: Test System architecture overview

As stated before, there are 11 test cases that applies to the BlueEye device (see Table 3). For illustration purposes, we show two test cases one regarding the device in idle mode and one in active mode (i.e., video streaming is active). More specifically, we illustrate IEC/CO/001 (energy consumption while in idle). The steps required for this test cases are very simple as shown in Figure 11. First, TAP will reboot the BlueEye and then start the measurement of data (Log start). The test case indicates we need to measure the energy consumption for 2 minutes. This duration can be controlled with the step "delay" which basically just make sure the device stays in idle as long as needed. Finally, after the required time has passed, we can disable the logs and save them in the specified log files (e..g., test case name and time) that will allow the collection of the KPIs later on by the testbed ETL process.

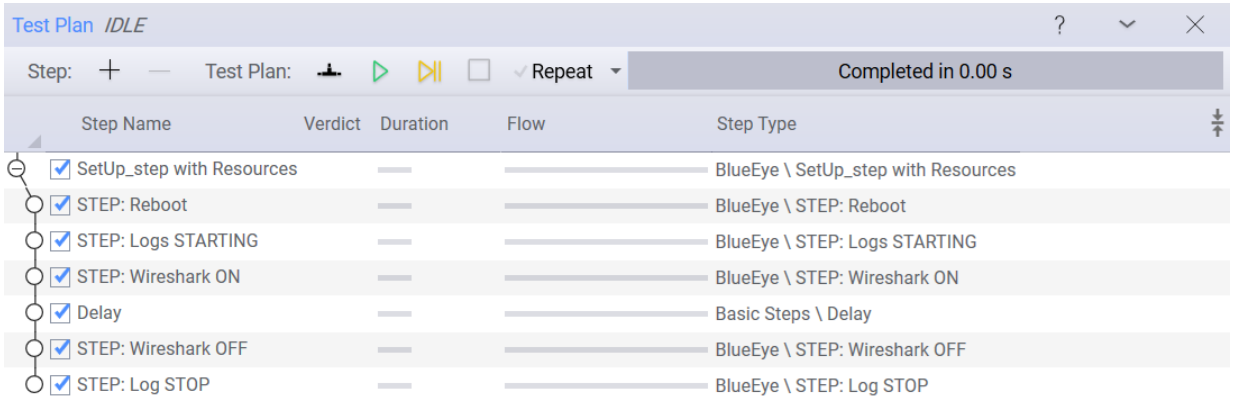

Figure 11: IEC/CO/001 TAP test case (device side)

The process for more challenging test cases is very similar to the one just described. Consider for example IEC/ES/001 (energy consumption while streaming video) as illustrated in Figure 12. The template is very similar, the only alteration with respect to the previous illustrated example is the addition of two steps "Start Video Streaming" and "Stop Video Streaming" right before and after the "Delay" step. This test case allows the TRIANGLE testbed to measure the average power consumption while the device is streaming video under different conditions.

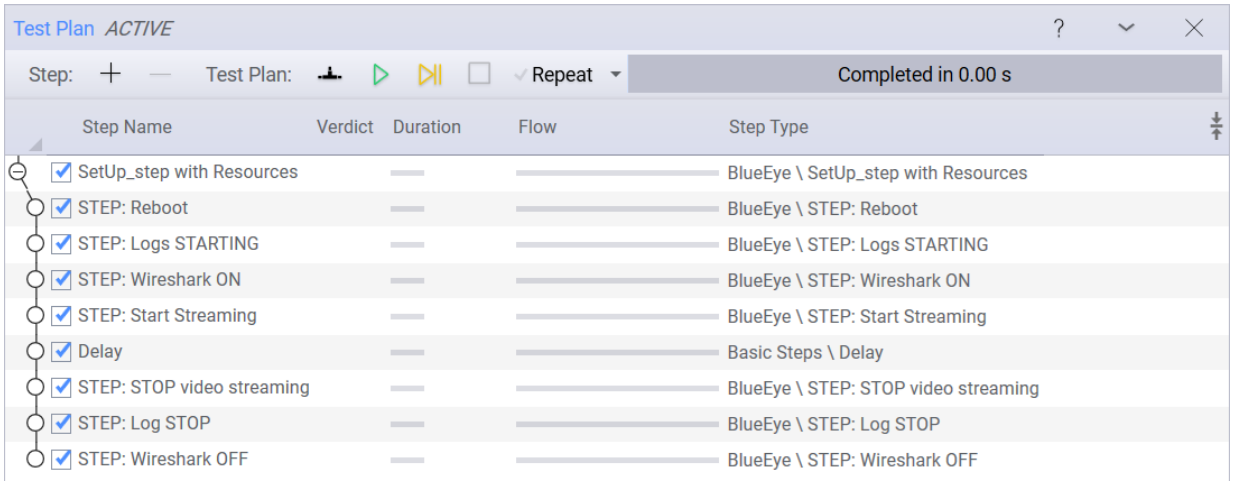

Figure 12: IEC/IES/001 TAP test case (device side)

It should be noted that the entire test plan is not shown in this document. The example focuses on the device side only. For the certification campaign to be run, instrument need to be properly configured, the correct network scenario loaded and only then the necessary commands send to the device (as shown in this deliverable). Finally, we also note that after the execution of each test case, the device will move the log files out of the IoT device. This step will not be considered for the measurements as it does not represent a typical behaviour of the device.

3.2.5 Test results

Once all the test cases under all the different network scenarios has been run, the testbed can use the ETL process to compute the TRIANGLE mark. In the particular case of the BlueEye device, the scores are summarized in Table 13.

Table 13. BlueEye TRIANGLE Mark

Domain	Score (/5)
IoT Devices Energy Consumption (IEC)	3.20
IoT Devices Data Performance (IDP)	2.58
IoT Devices Reliability (IDR)	1.68
TRIANGLE MARK	**2.49**

The device has scored on the positive side for energy consumption but low for reliability. The TRIANGLE mark also helped RedZinc to identify areas to improve in their device. For most of their tests, they always power cycled the device, i.e., turn the device on, run a demo and shut it down until the next demo. However, when stressing the device with continuous tests without power cycling it, new issues appeared that contributed to the low score in reliability. The IDP domain results shall be taken with a grain of salt, the TRIANGLE testbed has not run enough tests with emergency devices and therefore the parameters used to compute the MOS in this domain is not as mature as other domains. The results cannot be considered as representative considering the immaturity of the process. Outside the scope of this document, the process of the device and the process of the measurement will be revaluated,

4 Conclusions

The TRIANGLE certification provides an independent process to measure the performance in terms of Quality of Experience (QoE) of a mobile app or device, leading to a performance score for applicable domains. This document outlined the procedure used to award the TRIANGLE certification to mobile applications and devices.

This document outlined a summary of the TRIANGLE certification process, with the more detailed description of the process to award the TRIANGLE Mark available from D2.2. The TRIANGLE testbed provides a set of test cases configured according to the defined use cases and under specific test scenarios. The results of these tests are compared with a list of key performance indicators (KPIs) that will be used to build the TRIANGLE mark.

In order to evaluate the certification process, a sample mobile application and device were used to test the certification process with real data. These samples included the ExoPlayer mobile video player and the BlueEye device from RedZinc.

The process used to certify this app and device, along with tests performed the certification results are outlined in section 3. The computed QoE scores in each measured domain for each sample are presented as well as the final TRIANGLE mark.

5 References

[1] RedZinc http://www.redzinc.net/

[2] BlueEye https://www.redzinc.net/blueeye-for-operators/

[3] ExoPlayer https://google.github.io/ExoPlayer/

[4] Wikipedia Mean Score Opinion https://en.wikipedia.org/wiki/Mean_opinion_score

[5] TRIANGLE D2.2 Formalization of the certification process, requirements and use.

Appendix 1 - TRIANGLE ICS/IXIT template

Please fill in your product information in the 'Support' column (ICS) and the 'Value' column (IXIT) for the applicable tables.

ICS

Table A.1. General Information

Item	Description	Status	Supported values	Support	Mnemonic
1	Type of product	M1	Application, Mobile device, IoT device		ICSG_ProductType
2	Type of IoT device: Grid Powered, Long Lasting battery or Short Lasting battery IoT device)	C01	GP-IoT, LL-IoT, SL-IoT		ICSG_IoTDeviceType
3	Supported use cases: Virtual Reality, Gaming, Augmented Reality, Content Distribution Streaming Services, Live Streaming Services, Social Networking, High Speed Internet, Patient Monitoring, Emergency Services, Smart Metering, Smart Grids and Connected Vehicles	Mn	VR, GA, AR, CS, LS, SN, HS, PM, ES, SM, SG, CV		ICSG_UseCases

C.01 IF (A.1/1 = IoT device) THEN Mn ELSE N/A

Table A.2. Apps supported features (only applicable if A.1/1 = Application**)**

Item	Description	Status	Support	Mnemonic
1	*Support of Login*	*O*		*ICSA_Login*
2	*Support of logout*	*C.03*		*ICSA_Logout*
3	*Support of open menu*	*O*		*ICSA_OpenMenu*
4	*Support of activate backgroundservice*	*O*		*ICSA_Background*
5	*Support of play content (video/song)*	*C.01*		*ICSA_CSPlay*
6	*Support of re-play content*	*C.01*		*ICSA_CSrReplay*
7	*Support of skip forward content*	*C.02*		*ICSA_CSSkipForward*
8	*Support of fast forward content*	*C.02*		*ICSA_CSFastForward*
9	*Support of skip backward content*	*C.02*		*ICSA_CSSkipBackward*

10	*Support of rewind content*	*C.02*		*ICSA_CSRewind*
11	*Support of pause content (video/song)*	*C.02*		*ICSA_CSPause*
12	*Support of X8 speed fast forward content*	*C.04*		*ICSA_CSFastForwardX8*
13	*Support of X8 speed rewind content*	*C.05*		*ICSA_CSRewindX8*
14	*Support of Download content for offline playing*	*C.02*		*ICSA_CSDownloadMedia*
15	*Support of post text*	*C.06*		*ICSA_SNPostText*
16	*Support of post text and image*	*C.07*		*ICSA_SNPostImage*
17	*Support of post text and video*	*C.07*		*ICSA_SNPostVideo*
18	*Support of post text and location*	*C.07*		*ICSA_SNPostLocation*
19	*Support of post file*	*C.07*		*ICSA_SNPostFile*
20	*Support of get image*	*C.07*		*ICSA_SNGetImage*
21	*Support of get video*	*C.07*		*ICSA_SNGetVideo*
22	*Support of get live video*	*C.07*		*ICSA_SNGetLiveVideo*
23	*Support of get location*	*C.07*		*ICSA_SNGetLocation*
24	*Support of get file*	*C.07*		*ICSA_SNGetFile*
25	*Support of save game session*	*C.09*		*ICSA_GASave*
26	*Support of pause game session*	*C.09*		*ICSA_GAPause*
27	*Support of search and seek*	*C.02*		*ICSA_CSSearchSeek*
28	*Support of play content with screen off*	*C.02*		*ICSA_CSPlayScreenOff*
29	*Support of broadcast live video with screen off*	*C.08*		*ICSA_LSBroadcastScreenOff*

C.01 IF (A.1/3 INCLUDES CS) THEN M ELSE N/A
C.02 IF (A.1/3 INCLUDES CS) THEN O ELSE N/A
C.03 IF (A.2/1) THEN M ELSE N/A
C.04 IF (A.2/8) THEN O ELSE N/A
C.05 IF (A.2/10) THEN O ELSE N/A
C.06 IF (A.1/3 INCLUDES SN) THEN M ELSE N/A
C.07 IF (A.1/3 INCLUDES SN) THEN O ELSE N/A
C.08 IF (A.1/3 INCLUDES LS) THEN O ELSE N/A
C.09 IF (A.1/3 INCLUDES GA) THEN O ELSE N/A

Table A3. Mobile devices Features (Only applicable if A.1/1 = Mobile device)

Item	Description	Status	Support	Mnemonic
1	Support of Video Playing	M		ICSDM_PlayVideo
2	Support of Video Recording	O		ICSDM_RecordVideo
3	Support of Virtual Reality	O		ICSDM_VirtualReality
4	Support of Augmented Reality	O		ICSDM_AugmentedReality

Table A4. IoT devices Features (Only applicable if A.1/1 = IoT device)

Item	Description	Status	Support	Mnemonic
1	Support of Playing received video	O		ICSDI_ReceiveVideo
2	Support of Video transmission	O		ICSDI_TransmitVideo
3	Support of Playing Received audio	O		ICSDI_ReceiveAudio
4	Support of Audio transmission	O		ICSDI_TransmitAudio
5	Support of Idle mode	O		ICSDI_IdleMode
6	Support of Audio transmission without Video	C.01		ICSDI_TransmitAudioWithoutVideo
7	Support of Playing received Audio without Video	C.02		ICSDI_ReceiveAudioWithoutVideo
8	Device is connected to a Battery	O		ICSDI_Battery
9	Support of Playing received video	O		ICSDI_ReceiveVideo

C.01 IF A.4/4 THEN O ELSE N/A
C.02 IF A.4/3 THEN O ELSE N/A

Note: Items in this ICS tables only can take Boolean ‘Yes’ or ‘No’ values (Support column).

IXIT

Table B1. Applications IXIT (Only applicable if A.1/1 = Application)

Item	Name	Supported values	Mnemonic	Value
1	Operating System	Android, iOS	IXITD_OperatingSystem	

Table B2. Devices IXIT (Only applicable if A.1/1 = Mobile device OR A.1/1 = IoT device)

Item	Name	Supported values	Mnemonic	Value
1	Supported cellular technologies	GSM, UMTS, LTE, 5G	IXITD_CellularTechnology	
2	Supported frequency bands	As per TS 36.521-2 Table A.4.3-3	IXITD_Bands	
3	UE Power Class	As per TS 36.521-2 Table A.4.3-3b	IXITD_UEPowerClass	
4	Supported channel bandwidths	As per TS 36.521-2 Table A.4.3-3a	IXITD_Bandwidths	
5	UE Category	As per TS 36.521-2 Table A.4.3-4 and Table A.4.3-4a0	IXITD_UECategory	
6	UE Downlink Category	As per TS 36.521-2 Table A.4.3-4a and Table A.4.3-4aa	IXITD_DLCategory	
7	UE Uplink Category	As per TS 36.521-2 Table A.4.3-4b and Table A.4.3-4ba	IXITD_ULCategory	
8	Nominal working voltage		IXITD_NormalVoltage	

Appendix 2 – BlueEye API Details

Request structure

The API receives a request from TAP characterized by a code which describes what action should be executed in the device.
Request messages have a mandatory “req” field with the request code and other parameters required by that specific request.

Table 14. The request message from TAP to BlueEye API

Request		
Field	**Type**	**Description**
req	Integer	Request Code
(additional)	(Type)	Additional Arguments (varies per request)

Request codes

Table 15. Request codes

Code	Action	System
1	Software Reboot	Blue-Eye
2	Start Streaming to Hospital	Blue-Eye
3	Stop Streaming to Hospital	Blue-Eye
4	Start CPU/RAM/Radio logging	Blue-Eye
5	Stop CPU/RAM/Radio logging	Blue-Eye
6	Start Wireshark logging	Blue-Eye / Hospital
7	Stop Wireshark logging	Blue-Eye / Hospital
8	Move logs to the repository	Blue-Eye / Hospital

Examples:

```
{"req":1}
```

or

```
{"req":4, "file":"TestCaseLogONcode4","interval":10}
```

Response structure

Once the request is attended and completed by the socket, the API sends a response message to TAP with the result. The result meaning is represented by the code field of the response message.

Table 16. The response message from BlueEye API to TAP

Response		
Field	**Type**	**Description**
code	Integer	Response Code

Response code

Table 17. Response codes

Code	Meaning

0	Request OK
1	General Error
2	Malformed Request
3	Unknown Request
4	Unsupported Request (Trying a BE request on Hospital System)
5	File Error (unable to create log file)
6	Repository Error (no connectivity or space)

Example:

```
{"code":0}
```

Actions

These are the actions that will take place in the BlueEye API:

Software Reboot

This action will reboot a Blue-Eye System. Given the nature of the request, a successful response can also be a broken TCP connection (signalling it is down).

Message format:

Request		
Field	**Type**	**Value**
req	Integer	1

Start Streaming to Hospital

This action will request the Blue-Eye to start the camera/video streaming.

Message format:

Request		
Field	**Type**	**Value**
req	Integer	2

Stop Streaming to Hospital

This action will request the BlueEye to stop the camera/video streaming.

Message format:

Request		
Field	**Type**	**Value**
req	Integer	3

Start CPU/RAM logging

This action will request the Blue-Eye to ask for the CPU and RAM usage periodically with a time interval specified in the request message.

The measurements taken will be saved in a specific log file:

TestCaseName<date>.cpu.log

TestCaseName<date>.ram.log

Message format:

Request		
Field	**Type**	**Value**
req	Integer	4
file	String	filename to be used for the logs
Interval	Integer	Interval between measurements in seconds

Stop CPU/RAM logging

This action will stop all active logs and all measurements related to CPU and RAM usage.

Message format:

Request		
Field	**Type**	**Value**
req	Integer	5

Start Wireshark logging

This action will request the Blue-Eye, the hospital or both to start the traffic capture on a given interface (or first interface if no interface is provided) and to save it to a specified file based on the provided filename:

TestCaseName<date>.ws.log

Message format:

Request		
Field	**Type**	**Value**
req	Integer	6
file	String	filename to be used for the logs
If	String	(optional) interface to capture

Stop Wireshark logging

This action should stop capturing with Wireshark if ongoing.

Message format:

Request		
Field	**Type**	**Value**
req	Integer	7

Move logs to the repository

This action will copy the log files to a remote location and delete them from the disk after a successful copy. This action is needed due the reduce size memory in the IoT device and the Wireshark files big size. There must be take into consideration that this process should not be part of the results collection since it is not part of the normal functioning of the device.

Message format:

Request		
Field	**Type**	**Value**
req	Integer	8
url	String	Repository URL
user	String	Username for the repository
password	String	Password for the repository
filename	String	Base filename to copy over