

**Realising and Applied Gaming Ecosystem**

**Research and Innovation Action**

Grant agreement no.: 644187

**D2.3 – Real-time Emotion Detection Asset - Installation tutorial**

**RAGE – WP2 – D2.3**

**DRAFT**

|  |  |
| --- | --- |
| Project Number | H2020-ICT-2014-1 |
| Actual Date | April 2017 |
| Document Author/s | Dessislava Vassileva |
| Version | 0.3 |
| Status | Draft |



This project has received funding from the European Union’s Horizon 2020

research and innovation programme under grant agreement No 644187

**About the** **Real-Time Arousal Detection Using Galvanic Skin Response Asset**

Galvanic Skin Response (GSR), also referred to Electro-Dermal Activity (EDA), Skin Conductance Response (SCR), Psycho-Galvanic Reflex (PGR), or Skin Conductance Level (SCL), is related to the activity of the sweat glands, which are regulated by the sympathetic nervous system. When being open and functioning intensively, they emit water solution (sweat) which creates channels of higher conductivity toward the deeper skin layers. EDA represents the electrical conductivity of the skin, which is directly dependent on the activity of the sweat glands, and is often used to index the autonomic arousal. GSR offers a popular and affordable way for detection of player’s arousal in adaptive digital games and other affective computing applications. The asset produces real-time features of GSR signal measured from particular player such as: mean tonic activity level, phasic activity represented by mean and maximum amplitude of skin conductance response (all in micro-siemens), rate of phasic activity (response peaks/sec), SCR rise time, SCR 1/2 recovery time, and slope of tonic activity (in micro-siemens/sec). The level of arousal may be useful for emotion detection and for adaptation purposes. The asset will receive a filtered raw signal from a simple, low cost biofeedback device allowing sampling rate up to 2Khz. Measurements are carried out with two electrodes placed on two adjacent fingers. Recording, filtering and feature extraction might be executed on a computer (server) different than the game machine, in order to speed up all the required processing. The results will be communicated from the server-side to the client component in order to be used for game adaptation.

**Document scope**

The present document provides installation instructions of Real-Time Arousal Detection Using Galvanic Skin Response asset and, next, presents how to integrate and use the asset in a C# based game.

**Asset installation steps**

The asset installation process includes following steps:

1. Download from https://github.com/ddessy/RealTimeArousalDetectionUsingGSR/tree/master/Drivers the zip file *GSRDrivers.zip* with GSR device’s drivers;
2. Unzip the file *GSRDrivers.zip* and install drivers (from folders *SerialCOMDriver* and *drivers\_ft232*) if it is needed;
3. Download the zip file *RealTimeArousalDetectionDlls.zip* with *exe* and *dll* files implementing the asset functionality;
4. Unzip the archive file an copy the asset files in a folder;
5. Check the configuration file located at *./Resources/realTimeArousalDetectionAssetSettings.xml* and made changes if it is needed;
6. Start as click on the *DisplayGSRSignal.exe* file.

**Application settings**

1. **Basic application settings**

The basic application settings are following:

* SocketPort – the number of port of the socket that will communicate with the asset;
* SocketIPAddress – IP address of the socket that will communicate with the asset;
* COMPort – define the GSR device COM port. If it has value *N.A.*, the asset works with one random chosen COM port (but usually only one COM port is available);
* FormMode – the application run in background mode if it has value *BackgroundMode*;
* LogFile – path to the log file;
* DefaultTimeWindow - default value for the time window;
* SamplerateLabel - sample rate of the GSR device;
* ArousalLevel - number of arousal levels.

1. **Advance application settings**

The basic application settings are following:

* MinGSRDeviceSignalValue - the smallest possible value that can be detected by the GSR device;
* MaxGSRDeviceSignalValue - the largest possible value that can be detected by the GSR device;
* CalibrationTimerInterval – in the calibration period it is the time interval during which the asset measures the arousal status of the current user;
* MinArousalArea - the smallest value for arousal area achieved by the current user;
* MaxArousalArea - the largest value for arousal area achieved by the current user;
* MinTonicAmplitude - the smallest value for tonic amplitude achieved by the current user;
* MaxTonicAmplitude - the largest value for tonic amplitude achieved by the current user;
* MinMovingAverage- the smallest value for moving average achieved by the current user;
* MaxMovingAverage - the largest value for moving average achieved by the current user;
* ButterworthPhasicFrequency – frequency used in the butterworth filter for the phasic signal;
* ButterworthTonicFrequency – frequency used in the butterworth filter for the tonic signal;
* ApplicationMode – if it has value *TestWithoutDevice* the asset works with data from a file (that is specified in the setting TestData), but not with real data from the GSR device;
* TestData – path to the file that specified the static GSR data when value of the setting ApplicationMode is *TestWithoutDevice*.

**Asset integration steps**

The asset integration process includes following steps:

1. Start the asset;
2. Check if the asset’s socket server is available;
3. Run a socket client on the address (that is stated in the setting *SocketIPAddress*) and port (that is stated in the setting *SocketPort*) specified in the application settings;
4. At the beginning send to the asset message with content *“SOCP”* for starting calibration period;
5. After the calibration period send message *“EOCP”* for ending calibration period;
6. When you need information for the status of emotional arousal of the current user send message to the asset with the content *“GET\_EDA”*;
7. At the end of arousal measurement of the current user send message *“EOM”*.

**Example of a JSON object returned by the asset**

{

"SCRArousalArea":1936951.1918945313,

"SCRAmplitude":{

"Minimum":1964,

"Maximum":2336,

"Mean":2150,

"StdDeviation":186,

"Count":0.25,

"Name":"Amplitude"

},

"SCRRise":{

"Minimum":6940.4765625,

"Maximum":6940.4765625,

"Mean":6940.4765625,

"StdDeviation":0,

"Count":0.125,

"Name":"Rise time"

},

"SCRRecoveryTime":{

"Minimum":0,

"Maximum":0,

"Mean":0,

"StdDeviation":0,

"Count":0,

"Name":"Recovery time"

},

"SCRAchievedArousalLevel":41,

"**GeneralArousalLevel**":45,

"TonicStatistics":{

"Slope":-0.045041366115755613,

"MeanAmp":2166.5,

"MinAmp":1964,

"MaxAmp":2369,

"StdDeviation":202.5

},

"SCLAchievedArousalLevel":50,

"MovingAverage":2228.873381294964,

"LastValue":0,

"LastMedianFilterValue":1964,

"LastRawSignalValue":2002,

"HighPassSignalValue":0,

"LowPassSignalValue":0

}