

D4.1 – RAGE Asset Review

Citation for published version (APA):

Glass, J., & Star, K. (2018). D4.1 – RAGE Asset Review. RAGE project.

Document status and date:

Published: 01/01/2018

Document Version:

Publisher's PDF, also known as Version of record

Document license:

CC BY-NC-SA

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

<https://www.ou.nl/taverne-agreement>

Take down policy

If you believe that this document breaches copyright please contact us at:

pure-support@ou.nl

providing details and we will investigate your claim.

Downloaded from <https://research.ou.nl/> on date: 05 Dec. 2023

Open Universiteit
www.ou.nl





Realising an Applied Gaming Eco-system

Research and Innovation Action

Grant agreement no.: 644187

D4.1 – RAGE Asset Review

RAGE – WP4 – D4.1

Project Number	H2020-ICT-2014-1
Due Date	December 2018
Actual Date	December 2018
Document Author/s	Jared Glass, Kam Star
Version	1.0
Dissemination level	PU/ RE
Status	Draft
Document approved by	WW



Document Version Control			
Version	Date	Change Made (and if appropriate reason for change)	Initials of Commentator(s) or Author(s)
0.1	30/10/2018	Initial document structure	JG
0.2	26/11/2018	Initial content added	JG
0.3	11/12/2018	Minor Edits	JG
0.4	13/12/2018	Minor Edits	JG
0.5	14/12/2018	Update feedback	JG
0.6	18/12/2018	Updates and edits	KS

Document Change Commentator or Author		
Author Initials	Name of Author	Institution
JG	Jared Glass	PLAYGEN
RR	Ruben Riestra	INMARK
WW	Wim Westera	OUNL
PH	Paul Hollins	OUB
KS	Kam Star	PLAYGEN

Document Quality Control			
Version QA	Date	Comments (and if appropriate reason for change)	Initials of QA Person

TABLE OF CONTENTS

1 EXECUTIVE SUMMARY..... 4

2 REVIEW PROCESS 5

2.1. Scalar Evaluations:..... 6

2.2. Open-ended Evaluations:..... 7

2.3. Asset Bundles..... 8

2.4. Reviewers:..... 9

3 SURVEY RESULTS 10

3.1. Scalar Evaluation Points’ Results:..... 10

3.2. Open-ended Evaluation Points’ Results:..... 11

4 CONCLUSION..... 15

LIST OF FIGURES

Figure 1: Scalar Evaluation Results 10

LIST OF TABLES

Table 1: Scalar Evaluations..... 6

Table 2: Open-ended Evaluations..... 7

Table 3: Component Bundles..... 8

Table 4: Reviewers..... 9

Table 5: SM7 Feedback on quality of component decription 12

Table 6: SM8 Feedback on issues with documentation or demo 12

Table 7: TA4 Feedback on functional completeness 13

Table 8: TA5 Feedback on technical constraints 13

Table 9: TA7 Feedback on special considerations 14

Table 10: F1 Further comments 14

1 EXECUTIVE SUMMARY

This document provides a technical review of the final version of the components that make up the RAGE ecosystem. It is complimentary to D1.2 and D8.4 and covers input and feedback on asset description, functionality, technical constraints, reusability and considerations by the game developer partners.

Deliverables D1.2 (Asset Integration Methodology) and D8.4 (Evaluation of the Ecosystem Services and Processes) cover developers' experiences and component efficacy and as such it has been agreed to keep this deliverable 'light' and complimentary, focusing on the integration, constraints and reusability related aspects of various RAGE components.

Through the project, many meetings, interactive sessions and bilateral contacts have continually contributed to the improvement of the RAGE components. The evaluation in this document reflects the final cycle of feedback from the game developers.

The main purpose of this document is to summarize the technical experiences of the component integrators, allowing the identification of any issues so that they may be brought forward to ensure the sustainability of the RAGE ecosystem past the project's lifetime.

The structure of the document is as follows. The first section provides a review of the process involved, the methodology and reasoning behind the methods used to gather the information. The next section provides information on the asset reviewers. Next the results are provided in visual and descriptive formats, summarising specific responses where appropriate. Leading finally to the conclusions.

While some issues with documentation, setup processes and technical constraints were identified and brought to light in this document, the majority of the feedback was positive showing the final versions of the RAGE components to be functional, with little constraint and reusable.

2 REVIEW PROCESS

The review process relied on joint effort between all workpackages, in particular WP4 and WP8. Online survey evaluation instruments outlined in D8.1 were utilised to minimise sending out multiple questionnaires. The survey was then sent to the RAGE game development partners as well as a few external integrators of the components.

The survey was comprised of both scalar and open-ended evaluation points.

These evaluation points were designed to assess and provide feedback on the technical aspects of the component integration which cover:

Usability

How easy or difficult the component was to comprehend, integrate and use. This is based on the System Usability Scale outlined in the Usability evaluation instrument of D8.1.

Experience

The overall experience of working with the component developer. This includes the quality and level of support provided and uncover technology acceptance which is covered in User Acceptance of D8.1 and makes use of the Technology Acceptance Model.

Functionality

Based on the Resource Quality evaluation instrument in D8.1, these questions are designed to assess the component performance in practice along with any issues relating to its performance.

2.1. *Scalar Evaluations:*

The component bundle reviewer was asked to pick the value they feel most representative of their experience in a range of 1 to 7 ranging from “strongly disagree” at a value of 1 to “strongly agree” at a value of 7 (Likert scale).

Below is the list of scalar evaluations (Table 1):

ID	Variable	Question
SM2	Usability	The component bundle description is correct and accurately summarizes the component
SM3	Usability	The component bundle documentation is clear and easy to follow
SM4	Usability	Any included demos or example code was clear and easy to follow
SM5	Experience	The component bundle developers were quick to respond to any reported issues
SM6	Experience	The component bundle developers readily supplied help in regards to usage when reached out to
TA1	Functionality	The component bundle functioned as expected
TA2	Usability	The component bundle was easy to integrate (and host – if applicable)
TA3	Functionality	There were few if any technical constraints when using this component bundle

Table 1: Scalar Evaluations

2.2. Open-ended Evaluations:

Some evaluation points could not be adequately covered with simple numeric values and for those specific cases the component bundle reviewer was asked to input a text response.

Below is the list of open-ended evaluation questions (Table 2):

ID	Variable	Question
SM7	Usability	What information in the component bundle description is missing, incorrect or unhelpful?
SM8	Usability	Are there specific positive or critical issues with the documentation and demo provided for this component bundle?
TA4	Functionality	What functionality is missing, difficult to use or could greatly improve this component bundle?
TA5	Functionality	What technical constraints did you encounter?
TA6	Usability	How reusable is this component bundle? Was it easy to modify or port if you needed to and would you easily be able to integrate it into other games should you wish to?
TA7	Usability	What were the special considerations required for integration or hosting?
F1	General	Do you have any further comment on this component bundle?

Table 2: Open-ended Evaluations

2.3. Component Bundles

Due to the varying nature of each component's functionality, the various components that comprised one greater cohesive unit of functionality were conceptually combined into "component bundles".

These "component bundle" groupings also better mimic the integrator's experience. An component bundle may comprise of client and server components, both developed by the same component developer as requirements in support of the same goal.

Below is the list of "component bundles" as grouped by the survey (Table 3):

Game Analytics
Competence Adaptation
Motivation Adaptation
ReaderBench
Easy Dialogue Editor
FAtiMA
TwoA – Adaptation and Assessment
Player Profiling
Player Centric Rule and Pattern Based Adaptation
Emotion Detection – Facial Expression
Arousal Detection – Skin Response
Performance statistics
SUGAR
BML Realizer
LipSync Generator
Speech I/O
Evaluation Component

Table 3: Component Bundles

2.4. Reviewers

Besides the Game Development Rage Partners, we also included external institutions in the evaluation. As each of the Game Development Rage Partners has considerable experience in the field of serious games and a high level of familiarity with the project and the component bundles, opening this survey up to external partners allows for a more rounded variety of responses due to the differing experience levels.

In Table 4 information about technical abilities and experience of the component bundle reviewers are summarised:

Years developing games	Years serious developing games	Amount of games worked on	Programming languages and software development paradigms
17+	17+	40+	Unity, C#, JavaScript, Flash, Html, ASP.NET, WPF, Xamarin, Python, C++, Go
10+	10+	10	C++, C#, Html 5
33	15	50	C++, C#, Unity, Microsoft Visual Studio
25	15	100+	C++, C, C#, Objective C, JavaScript, Java
10	5	6	C#, Unity
10	10	15	C#, JAVA, C, C++
1	1	1	C#, Unity
2	1	4	C, C++, C#, Python, JavaScript
1	1	1	C#
5	5	2	JavaScript, Haskell
0	0	2	Python, C++
0	0	1	C, C++, Java
1	0	2	Java, C++

Table 4: Reviewers

Due to the nature of component bundle integration, reviewers do not provide feedback on all component bundles but only the subset that they have used. Due to this the number of responses per component bundle may differ with some component bundles having fewer reviews although they may have been used by a team of many people.

3 SURVEY RESULTS

3.1. Scalar Evaluation Points' Results:

The distribution of the ratings for the scalar evaluation points across all component bundles are as displayed in the graph below. The “X” denotes the mean with the extents showing the maximum and minimum. The two dots represent outliers. These results are discussed below.

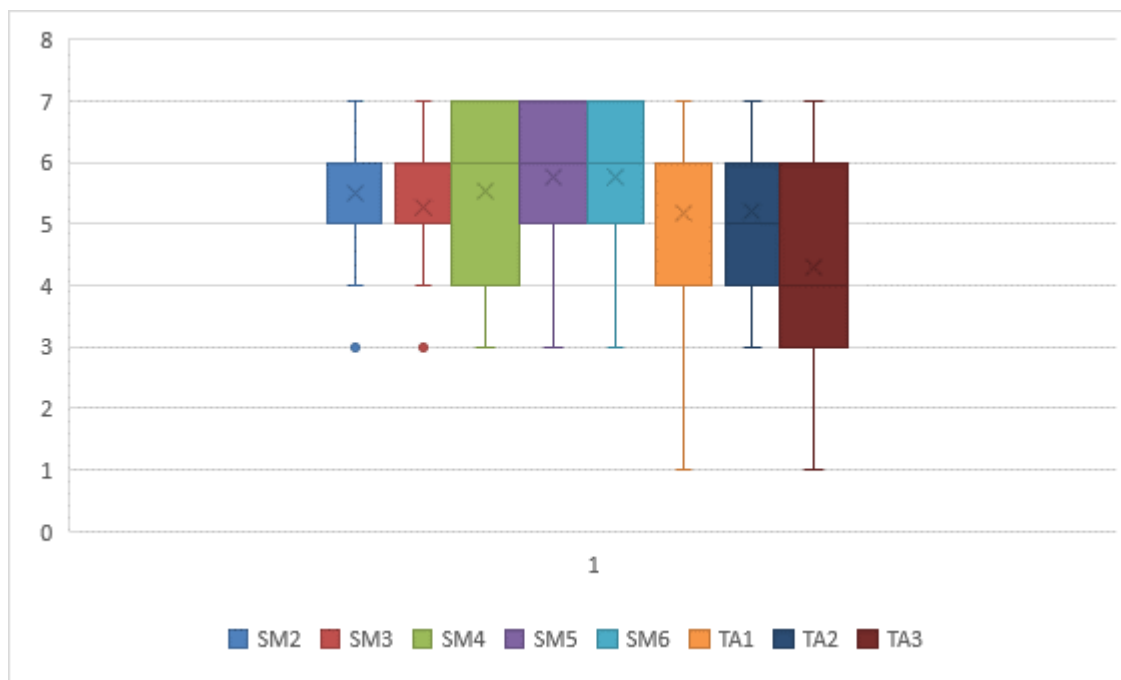


Figure 1: Scalar Evaluation Results

Each scalar evaluation point has been phrased in a positive way meaning a higher value in the response indicates a more positive review. A value of 7 indicates that the reviewer strongly agrees with the evaluation point and a value of 1 indicates that the reviewer strongly disagrees. Figure 1 reveals an overall positive, or moderately positive judgement. Details are provided in the next subsections.

3.1.1 SM2: The component bundle description is correct and accurately summarizes the component

Although there are some outliers around the value 3, the majority of the responses form a relatively tight distribution between the values 5 and 6 indicating that in most cases the component description is seen as accurate.

3.1.2 SM3: The component bundle documentation is clear and easy to follow

Similarly, for the documentation with the majority of reviewers finding the documentation clear and easy to follow with some outliers at the value of 3.

3.1.3 SM4: Any included demos or example code was clear and easy to follow

The average reviewer still agrees with this point but we have a few component bundles that border on “somewhat agree”. These component bundles being: “Motivation Adaptation” and “Performance Statistics”.

3.1.4 SM5: The component bundle developers were quick to respond to any reported issues

With this evaluation point the distribution goes up again with “Player Centric Rule and Pattern based Adaptation”, SUGAR and FAtiMA being reviewed as “Strongly Agree”. “Motivation Adaptation” is at the bottom at “somewhat agree” with the “BML Realizer” just above it.

3.1.5 SM6: The component bundle developers readily supplied help in regards to usage when reached out to

Being similar to SM6 the results are similar too with “Player Centric Rule and Pattern based Adaptation”, SUGAR and FAtiMA being reviewed as “Strongly Agree” and “Motivation Adaptation” again at the bottom at “somewhat agree” with the “BML Realizer” just above it.

3.1.6 TA1: The component bundle functioned as expected

Most reviewers agreed with this evaluation point except for one specific outlier. In the case of the “BML Realizer” which scored a “strongly disagree”. The open ended questions that follow this section are designed to gather further information on ratings such as these.

3.1.7 TA2: The component bundle was easy to integrate (and host – if applicable)

The distribution stays in the general agreement range with only the “TwoA” component bundle dipping into the disagree range.

3.1.8 TA3: There were few if any technical constraints when using this component bundle

Here the “BML Realizer” and SUGAR were reviewed as “Strongly Agree” with the “Evaluation Component”, “Motivation Adaptation”, “Competence Adaptation” and “Game Analytics” proving to have more technical constraints.

3.2. Open-ended Evaluation Points’ Results:

These evaluation open-ended points serve the purpose of gathering qualitative feedback on the ratings of the scalar evaluation points as well as to give the reviewers a chance to add any other information they would deem useful.

For the purpose of brevity, not all feedback has been included in this section. The statements and summaries incorporate the feedback from each review but the tables below display only highlight any critical points that can be seen as actionable tasks which will be relayed to the component developers to ensure the continual improvement of components and the ecosystem’s sustainability past the project lifetime.

3.2.1 SM7: What information in the component bundle description is missing, incorrect or unhelpful?

This evaluation point aims to get the explanation of the rating chosen for SM2 (quality of component description). Unsurprisingly the responses to SM7 mirrored SM2 with the majority of reviewers claiming that the component bundle description is useful and correct as it is (Table 7).

Noteworthy critical feedback:

Component Bundle	Review Segment
Game Analytics	Some terminology could be simplified. Examples could be provided
Competence Adaptation	Documentation should be clear and the functionality of the component should be obvious
TwoA	A clear example with all input required would be useful.

Table 5: SM7 Feedback on quality of component description

3.2.2 SM8: Are there specific positive or critical issues with the documentation and demo provided for this component bundle?

In reflection of SM2 (“The component bundle is correct and accurately summarizes the component”) and SM3 (“The component bundle documentation is clear and easy to follow”) the responses did not express any critical issues. There were however some suggestions as to what would make the documentation and demos more useful:

Component Bundle	Review Segment
Evaluation Component	Clear examples. less jargon. to make faster to integrate
Motivation Adaptation	Wording and jargon should be minimised to make as accessible as possible.
TwoA	A clear example with all input required data would be useful.

Table 6: SM8 Feedback on issues with documentation or demo

3.2.3 TA4: What functionality is missing, difficult to use or could greatly improve this component bundle?

As with the prior points the responses to this evaluation point didn’t differ greatly from the related scalar evaluation point responses with the majority of responses claiming that the current functionality is adequate.

Noteworthy critical feedback:

Component Bundle	Review Segment
Game Analytics	Simplified setup procedure

ReaderBench	A local version. Internet call from the code can generate problems with some companies' security system, like Randstad
Easy Dialogue Editor	A local BML interpreter for Unity
SUGAR	Leaderboards are slow. Some naming can be misleading or confusing
Evaluation Component	The component contains no built-in asynchronous method for sending data, which had to be added to ensure that games did not lock up during the sending of data

Table 7: TA4 Feedback on functional completeness

3.2.4 TA5: What technical constraints did you encounter?

As component integration is a technical task most of the reviewers claimed that the process was either easy or standard as far as their experience goes. Many of the components offered no major constraints but there were some components that did have reported constraints to overcome:

Component Bundle	Review Segment
ReaderBench	Mainly problems with the Randstad security system and the associated rights.
Easy Dialogue Editor	We had to write a local version of the BML interpreter
FAtiMA	We had to ensure that the component did not result in excessive lock-ups or delays when being used on lower-powered device, such as mobile devices.
Emotion Detection – Facial Expression	In Unity, after integration, the results are different than the demo... depending on lighting and cameras some results are less than desirable
Evaluation Component	One has to account for the game locking up when data is sent.

Table 8: TA5 Feedback on technical constraints

3.2.5 TA6: How reusable is this component bundle? Was it easy to modify or port if you needed to and would you easily be able to integrate it into other games should you wish to?

All of the component bundles were reported as highly reusable. In many cases, such as with: Game Analytics, FAtiMA, Easy Dialogue Editor, SUGAR, and the Evaluation Component, reviewers reported already having integrated the specified component bundle into multiple projects.

3.2.6 TA7: What were the special considerations required for integration or hosting?

A few special considerations were provided:

Component Bundle	Review Segment
Game Analytics	We have to host our own version of the server side of the component for gathered information to be sent to.
ReaderBench	Internet connection. And it could be a problem with big companies like Randstad.
Easy Dialogue Editor	Write a local interpreter.
FAtiMA	We have to ensure that files related to the component could be loaded on all platforms games were expected to run on and that loading/saving of files did not cause the game to freeze for an excessive amount of time.
Arousal Detection - Skin Response	Security issues at Randstad
BML Realizer	We have to write our own BML parser for Unity!

Table 9: TA7 Feedback on special considerations

3.2.7 F1: Do you have any further comment on this component bundle or any specific comments on individual components in this bundle?

Here reviewers were given the chance to address any other concerns or issues that the other evaluation points may not have covered.

Noteworthy critical feedback:

Component Bundle	Review Segment
Game Analytics	Simplified integration and setup should be priority.
Competence Adaptation	Documentation and simple example needed.
Motivation Adaptation	The documentation and examples could be clearer, concise
TwoA	A clear concise example and description with minimal jargon
Performance Statistics	Clear example. minimise jargon
Evaluation Component	Most important is clear docs and a simple example

Table 10: F1 Further comments

4 CONCLUSION

While there was a good and varied amount of feedback, some quantifiable and some not, the majority of the reviewer's feedback reflected positively on the component bundles on both usability, functionality and experience.

In some cases component integrators felt that, although the process had improved in the process, the integration processes could be simplified even further.

Improvement to documentations and further examples had been welcome additions during the project, and could continue to evolve and improve beyond the end of the project, particularly as the components are open source.

Running an applied game is not always possible in commercial environments, particularly if they are making use of novel or unusual infrastructure or required particular security or technical elements. As such one of the components suffered from limitations imposed by the partner site's security infrastructure. This imposed various technical constraints on the component integrator.

Going forward, it is hoped that component developers continue to use these feedback to improve their offerings and help ensure continued use of the RAGE ecosystem beyond the end of the project.