TESIS DE **MÁSTER**

TICRM

**Máster y Doctorado en Tecnologías de la
Información y Comunicaciones en Redes Móviles**

ANEXOS

Estudio de la Calidad Experimentada (QoE) en el Servicio de Navegación Web:
Contribución a la Validación de las Recomendaciones ITU-T P.1501 e ITU-T G.1031

ANEXO I
Draft P.STMWeb
Draft Test Plan P.STMWeb

Las siguientes páginas recogen los dos borradores P.STMWeb desarrollados en el seno del grupo de trabajo SG12 de la ITU.

El primero (Draft P.STMWeb) describe la arquitectura de referencia, especificaciones, requerimientos, escenarios de prueba y metodología del experimento de evaluación de QoE en el servicio de navegación web.

El segundo (Draft Test Plan P.STMWeb) describe el plan de pruebas para el experimento.

INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 2009-2012

COM 12 – C 337 – E

May 2012

English only

Original: English

Question(s):	7/12

<div align="center">

STUDY GROUP 12 – CONTRIBUTION 337

</div>

Source:	A1 Telekom Austria AG
Title:	Draft P.STMWeb

1   Scope

This Recommendation describes the method and procedures for subjective testing of user perceived quality of web browsing. The method does generally apply to degradations and characteristics that can be introduced on the network level (e.g. round trip time, downlink and uplink bandwidth, packet losses etc.) but as well on the application level (page load times etc.). Combinations of two or more of such factors have to be catered for.

2   References

The following ITU-T Recommendations and other references contain provisions, which through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T P.800]   ITU-T Recommendation P.800 (1996), *Methods for subjective determination of transmission quality*.

[ITU-T P.800.1]   ITU-T Recommendation P.800.1 (2006), *Mean Opinion Scores (MOS) terminology*.

Contact:	Sebastian Egger Forschungszentrum Telekommunikation Wien, Austria	Tel: + 43 1 5052830 12 Fax: + 43 1 5052830 99 Email: egger@ftw.at
Contact:	Kathrin Masuch Forschungszentrum Telekommunikation Wien, Austria	Tel: + 43 1 5052830 914 Fax: + 43 1 5052830 99 Email: masuch@ftw.at
Contact:	Dominik Strohmeier Deutsche Telekom Laboratories Germany	Tel: + 49 30 835358468 Fax: +49 30 835358409 Email: dominik.strohmeier@telekom.de
Contact:	Raimund Schatz Forschungszentrum Telekommunikation Wien, Austria	Tel: + 43 1 5052830 35 Fax: + 43 1 5052830 99 Email: raimund.schatz@ftw.at

[ITU- P.910]   ITU-T Recommendation P.910 (2008), Subjective video quality assessment methods for multimedia applications

[ITU-R BT.500]   ITU-R Recommendation BT.500 (2012), Methodology for the subjective assessment of the quality of television pictures

[ITU-T Y.1540]   ITU-T Recommendation (2011), *Internet protocol data communication service – IP packet transfer and availability performance parameters*

[ITU-T Y.1541]   ITU-T Recommendation Y.1541 (2011), *Network performance objectives for IP-based services.*

[ITU-T HSTP]   ITU- T, *Handbook of subjective testing practical procedures*

3   Definitions

This Recommendation defines the following term:

Condition: time period in which a defined QoS setting is applied for a certain web page. A condition is concluded by retrospective user ratings on one or multiple dimensions of quality.

Task: is a given set of instructions given to the user on how to reach certain information within one test condition.

Web browsing session: A web browsing session (WBS) starts with a user's intended, initial request of a web page to achieve a certain task and ends with its fulfilment. Typical for such a session is the engagement of the user in a flow through several consecutive web page views.

4   Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

DLBW   Downlink Bandwidth

MOS   Mean Opinion Score

PL   Packet Loss

PLT   Page Load Time

RTT   Round Trip Time

ULBW   Uplink Bandwidth

WBS   Web browsing session.

5   Conventions

The methodology discussed within this recommendation is tailored for web browsing only. Browser-based applications such as online office applications, webmail etc. might be evaluated by the same means, but they have to consider different scenarios depending on the defined tasks during subjective evaluation and the technical implementation of the services

The focus of this recommendation is set on the evaluation method of the complete WBS to achieve and maintain a flow experience of web browsing for better external validity of the test results. However, this methodology's test setup with respect to test facilities, test conditions, subject selection, and test devices is also applicable to the subjective evaluation of single web page loading times.

6 Web browsing test procedure

6.1 Purpose

Subjective quality tests for web browsing aim at assessing the impact of transmission impairments on the user perceived quality of a web browsing session under realistic usage conditions.

The main characteristics of a subjective web browsing test are:

- To be very close to real web browsing where people are browsing and interacting with web pages in order to acquire certain information, the procedure they go through within this methodology ensures that people get into a browsing mode rather than a pure page loading mode.

- Subjects are exposed to a certain QoS level over a time period rather than for one event, in order to grasp several request-response cycles for the subjective evaluation.

- The use of certain tasks to stimulate the interaction between the web page and the subject for each test condition.

- The content e.g. the web page must be interactive and has to provide sufficient content such that the subject can browse through it over several conditions without getting bored.

- This test methodology can be adapted to field testing; however, it is has to be ensured that the manipulated parameters (e.g., delay, packet loss, downlink bandwidth, etc.) can be set to the desired values and that these settings can be verified by a posteriori analysis (e.g. traffic traces).

The results obtained with this methodology, can be used by telecommunication providers, vendors and customers to get information of the influence of the tested technical parameters on end user perceived quality.

6.2 Test facilities

Typically, test facilities for subjective web browsing evaluations do include an end user device, a network emulator, a router, and some optional components as depicted in in Figure 3 below. Network parameters are set through a network emulator (EMU) which is placed between the user device (UD) and the router (RT), which ensures the routing between eventual local content servers and the internet.

When using live web pages within the evaluation one has to ensure the measurement of the page health in order to identify potential influences due to congestions or server malfunctions through a page health monitor (PHM) connected to the internet via the router as indicated in Figure 1. . In case multiple subjects participating in parallel it must be ensured that each subjects receives the obliged parameters, hence shared channel settings have to be avoided.

If web pages are used including video content that is also affected by the test parameters, requirements for the lab facilities from ITU-T P.910 apply in addition.
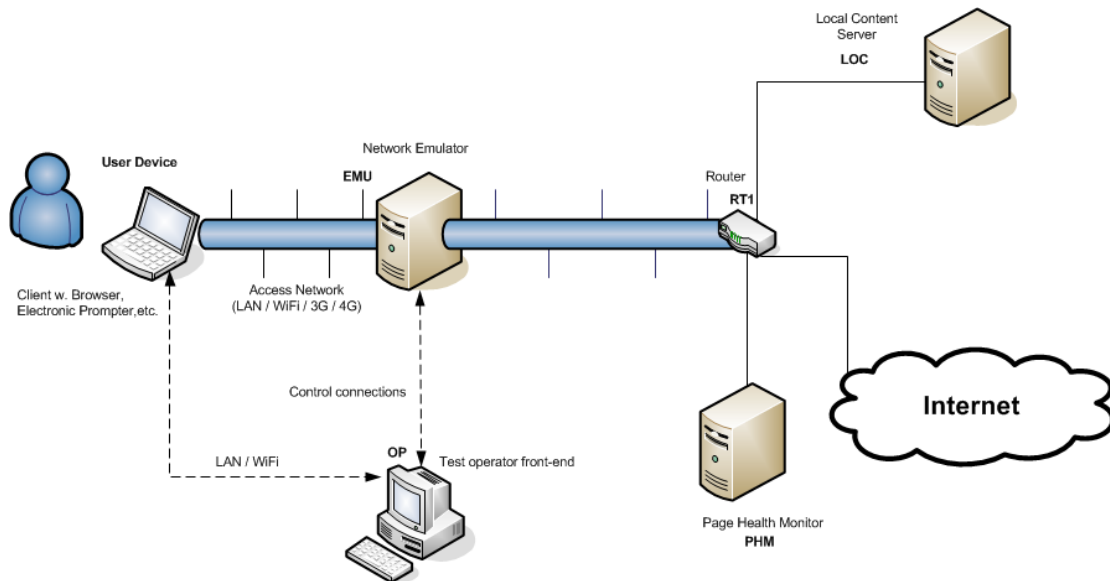
Figure 1: Example of test facilities

6.3 End User Devices

The end user device has a major impact on the overall performance as perceived by the user. Therefore, detailed reporting about the device characteristics and setting are essential. If the end user device or only any of its characteristics are considered to be a variable parameter of the quality evaluation, then all applied variations must be reported.

6.3.1 Hardware

Due to the high dynamics in web technology developments it is viable that recent hardware is used for the tests in order to ensure that device performance does not dominate the chosen test variables unintended.

At the time of drafting this recommendation hardware can be roughly classified in four categories (a) PC / Laptop, b) Tablets, c) Smartphones, d) other devices that are able to connect to the web). Typically, these four categories feature different performance issues. Therefore, one should only use devices of different categories in case they are part of the condition variables under test.

Displays should be calibrated to D65 at the beginning of the test if possible (might not apply to device categories b, c, d); subjects must not be allowed to modify the luminance levels.

For each of the devices used within a test the technical details have to be included in the reporting as detailed in Appendix III.

6.3.2 Software

Regarding the operating system of the devices, the latest versions of major operating systems should be used with all updates applied. The software related details that have to be included in the report are enlisted in Annex I

6.3.2.1 Browser

For the device categories b), c) and d) it is recommended to use the default browser as these are optimized in several ways for these devices. In case of category a) the latest version of a major browser should be used. However, one might apply older legacies of browsers if needed to run special browser extensions. Regarding these browser extensions, as few as possible extensions should be used to ensure the performance of the browser.

6.4 Test Content

For the content selection typical web pages should be chosen. On has to ensure that the web page is well programmed and its server infrastructure is fast and reliable also across peak times. It is recommended that server performance is monitored over one week before a web page is chosen as test content.

Content types and examples: news page (www.spiegel.de), photo album (locally hosted), web shop (www.amazon.de). The content examples are chosen such that they reflect typical web pages frequently browsed by German Internet users.

Web page characterization → separate Annex B

6.5 Test design

The flow characteristics of web browsing demand condition lengths with several technical request-response patterns (Ref. C-0236). Therefore, a web browsing session of min. 150 sec is recommended which is then followed by the retrospective ratings on the dimensions of overall quality, perceived speed and stability of the connection. Including the ratings one condition is of about 180sec length. Such a length is a reasonable trade-off between achieved user flow and practical considerations such as number of possible conditions and maximum overall test duration. A maximum active test duration of two 45 min blocks separated with a 10 min is recommended (Ref: JAES Journal). In addition an instruction block in the beginning and a debriefing block in the end are also recommended. The overall duration should not exceed 120 min. An example of a test schedule with 24 test conditions is shown in Table 1.

Compromises have to be made between the test duration and the choice of conditions. If more conditions are to be tested, different experiments and may require a between-subject or balanced incomplete block design. The order of the conditions should be pseudo randomized, e.g. by means of Latin Squares or other experimental designs as described in Section 3.3 / [ITU-T HSTP].

Each test subject should be seated in a realistic environment according to the chosen scenario. Beginning and end of each condition must be clearly indicated to the test subject to ensure ratings corresponding to the experimental parameters set. However, care has to be taken to make these notifications clear but also not too intriguing to keep the subject in the web browsing flow.

After arrival of the subjects an instruction has to be conducted where the nature of the test and the tasks are explained. This should be followed by a training session in which the participant goes through all different tasks given to ensure correct understanding and execution of the tasks and in which he practices the use of the scales to rate the quality retrospectively. The training session should also ensure that parameter values from the lower end and the higher end of the parameter range are used in order to acquaint the subjects with the parameter range to be expected throughout the test. In addition it has to be ensured that the parameter ranges are chosen such, that the whole range of the scale is used

	Instruction	Training Session	Session 1	Break	Session 2
Number of test conditions		3	12		12
Time	15 mins	9 mins	36 mins	10 mins	36 mins

Table 1: Test schedule for 24 conditions

6.6 Test conditions

Typically for web technologies services are delivered via IP networks. These networks are nowadays available in fixed and mobile scenarios with different network characteristics and are accessed with different devices. For the design of the test this has to be considered in order to achieve a realistic scenario for the test. A more detailed discussion regarding IP parameters can be found in [ITU-T Y.1540] and [ITU-T Y.1541]

Examples of test parameters are:

On a Network level:

- DLBW / ULBW
- RTT
- Temporal fluctuations of DLBW, ULBW, RTT
- Etc.

On an application level :

- PLT

On a device level :

- Devices (PC, Laptop, Tablet, Smartphone)

6.7 Subjects

Special care should be taken when selecting subjects for tests with tablet and smartphone devices. As these devices are not as widespread as PC's some potential subjects will be more experienced than others. It is recognized that the levels of experience with specific equipment or technology is a continuum ranging from those who are completely unfamiliar with technical behaviour of the equipment under test ("non-experts") to those who are thoroughly competent in the operation and maintenance of this equipment ("experts"). Another criterion for subject selection to be considered in web browsing is the subject's experience and regular usage of the test content. It is noted that good knowledge about the structure and usual behaviour of the test web page can have a severe impact on the sensitivity due test parameter changes of test subjects.

In order to ensure that the influence of the above mentioned experience with the devices and web pages under test are minimized it is recommended that the selected subjects are experienced with the devices under test and the web pages used.

 Unless gender, age and other socio-economic characteristics are design factors of the test, then a formal web browsing test should be populated (on a best-endeavour basis) with a random mix of subjects.

Age and gender as well as socio economic characteristics of all types of subject should be recorded for all types of tests.

6.8 Tasks

In general, two different modes for the task during a WBS can be defined. The free-browsing mode does not require specific goals to be achieved during the session. Subjects are asked to explore the test content on their own by requesting several web pages. In the task-dependent mode, specific tasks are defined the subject shall perform on the web pages. The end of such sessions can be determined either by a specific number of web pages requested and loaded consecutively or by a given session time.

In general, the task shall be intrinsically motivating and easily solvable without training. Additionally, the task accomplishment mustn't be limited by existing/known usability problems to avoid unintended impact of task design on test parameters. To avoid boredom, tasks should include a logical structure and relationship to the purpose of the website (e.g. look for offers on shopping page or look for information on the latest headline on a news page). Examples of suitable tasks for different website classes are included in Appendix II.

6.9 Questions

Single or multiple dimensions of perceived quality are evaluated retrospectively after each test condition. Examples of absolute category scale questions and labels as well as a binary response scale are given below. It is noted that the actual questions chosen for the test are dependent on the research question to be answered and might exceed these questions. Further examples of questions for other dimensions of interest are given in Annex III. However, it has to be considered that the cognitive load on the subjects and therefore the number of questions asked should be kept as low as possible in order to reduce any possible confusion and subject fatigue.

6.9.1 For the general assessment of the perceived overall quality of a test item:

6.9.1.1 "Please rate the overall quality of the connection you have been using on the following scale"

The scale descriptors are:
*   Excellent
*   Good
*   Fair
*   Poor
*   Bad

6.9.1 For identification of quality thresholds with respect to acceptance of the condition:

Please note that a context must be defined for the threshold. (e.g. accept quality for your internet connection at home; accept quality for the price paid for the service). The context reference is indicated by the underlined words in the example question.

6.9.1.1 "Would you accept this connection quality in <u>normal life at home</u>?"

*   Yes
*   No

The previous examples should be supplemented by the experimenter to address the needs of the specific experiment. When using multiple scales for assessing the multi-dimensional aspect of quality, care should be taken to ensure that the previous responses are not available to the subjects.

6.10 Data analysis and report

6.10.1 Analysis methods

Regarding the analysis methods the recommendations from [ITU-T P.800] and [ITU-T P.805] and Section 6 / [ITU-T HSTP] do apply.

Considering the small sample sizes achieved, one can in addition to the analysis methods mentioned in the above references, also use non-parametric statistical hypothesis tests such as Mann-Whitney U, Wilcoxon signed rank tests and others.

6.10.2 Reported results

The report must contain the questions and the category scales used. Due to the possible influence from the setup itself it is also mandatory to report the following setup facts in detail: Browser used, browser version, Plugins used date of the test,

Summary results should include, as a minimum, mean ratings and standard deviations for all tested conditions and for all questions. If the experimenter expects a gender pairing effect, this special case should be also considered in the design of the experiment and particularly in the pairing of subjects. This effect should be analysed and reported.

A list with mandatory structure and reporting details is attached in Appendix III.

6.10.3 Report on the test design

Each report must contain the webpages used, the browser version used, time of the test, Additional plugins used.

Annex A

Result reporting: Mandatory structure and details

This appendix forms an integral part of this Recommendation

1   Test Setup

1.1 Test participants

A description of the user sample used has to include at least the following details (a more detailed description is of course possible):
- Number
- Mean Age
- Median Age
- Gender
- Web usage (hours per week for each participant)
- Experience with web pages (average visits per week)

1.2 Test parameters

In this section, all independent variables, which were varied throughout the duration of the test have to be enlisted with their respective parameter ranges.

1.3 Web Pages used

For the used web pages the following details have to be reported:

- URL

- Start and end date of the test

- Web page class (e.g. news, photo album, geo location, web shop etc.)

- Web technology used (HTML, Ajax, Flash, Java etc.)

- Average page size in kByte, standard deviation of the page sizes

- Average elements per page, standard deviation of elements per page

1.4   Test facilities

The test facilities such as the environment the test were conducted in and the technical setup used have to be reported in this section such that a replication of the test facilities can be achieved by a third party.

1.5 Test Devices

For the devices used throughout the test the following details have to be reported:

Hardware:

- Brand

- Model Number

- CPU (type, speed)

- RAM (MB / GB)

- Screen diagonal

- Screen resolution (Pixel horizontal x Pixel vertical) and in case the browser window is not maximized, its resolution has to be reported additionally.

- Screen luminance

Software:

- Operating system + built number

- Browser + version number

- Enabled browser extensions + version numbers

- Any kind of speed optimization used (e.g. http pipelining)

1.6 Questions used

All question used for the evaluation of the user perceived quality on several dimensions have to be reported in their exact wording in this section. The scale types used (ACR-5, ACR-9, ACR-11) have to be included in the report as well.

2 Result Presentation

2.1 Acceptance results

For each condition tested and the following details have to be reported:

- Number of positive ratings

- Number of negative ratings

- Number of no responses

2.2 MOS results

For each condition tested and the following details have to be reported:

- Mean

- Number of responses

- Standard Deviation

- Minimum and Maximum rating

For graphical result presentation confidence intervals have to be shown and their properties (percentiles etc.) have to be reported.

Annex B

Web page categories

This appendix forms an integral part of this Recommendation

Appendix I

Example of instructions for the web browsing test

(This appendix does not form an integral part of this Recommendation)

INSTRUCTIONS TO SUBJECTS

In this experiment we are evaluating the connection quality of internet connections. Therefore you are asked to browse through various web pages, solve different tasks and rate the perceived connection quality. Please concentrate on the quality of the internet connection and don't get biased by the content of the web pages.

After completing each condition, you will be asked to give your opinions on the connection quality by answering the questions displayed on the laptop in front of you. Please answer those questions intuitively. The scale is ranked from very poor to excellent, whereas "very poor" is the worst and "excellent" is the best connection quality, which you can imagine. After clicking the "save" button, your judgement will be stored. After filling out the form, please continue with the task, until you get further instructions.

Below you can see an explanation to all questions you will be asked.

" Please rate the overall quality of the connection you have been using on the following scale

Bad	Poor	Fair	Good	Excellent

"Would you use/accept this connection quality in normal life at home?"

No	Yes

You will have a break after approximately 45 minutes. In total the test will last approximately 120 minutes.

Appendix II

Example task instructions for the web browsing test

Photos

Task

1. Please choose one of the photo albums on the start page.

2. Select the first picture preview and wait until it's completely displayed.

3. Now browse through the album by clicking on the "next" button or direct on the picture. Please view each picture for at least 3 seconds.

4. For each photo task note the numbers of the five pictures you like most and the five pictures you dislike most.

News Page

URL: http://www.spiegel.de/
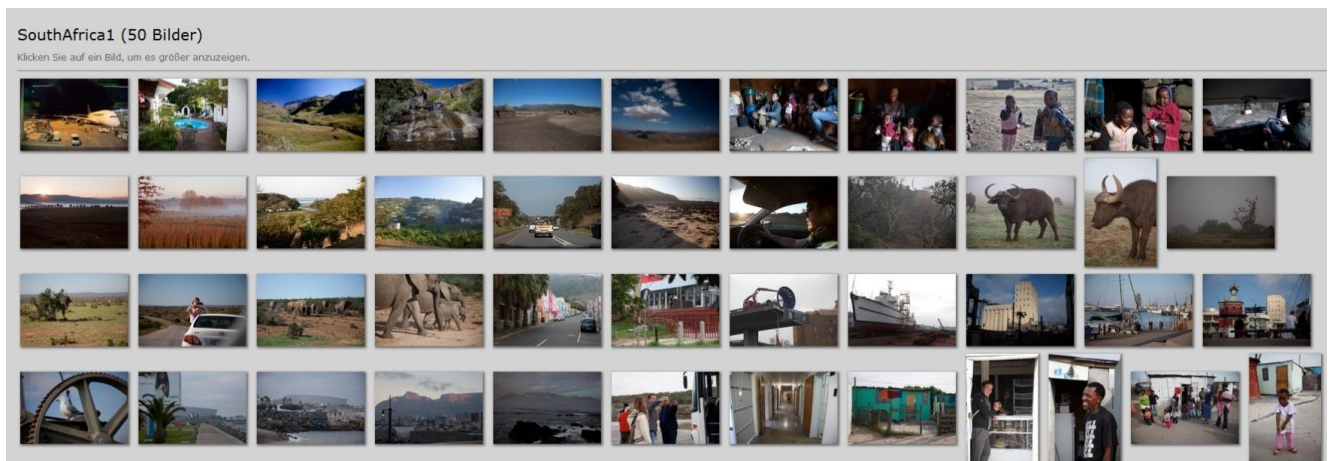
Task:

Try to get an overview of the current news in the [Economy, Politics, Sport, Culture,] section. Also   click into the articles to get an **impression** of the topics covered (but please do not read the articles completely). Please keep in mind, that we ask you to rate quality of the internet connection, therefore you should be able to form an opinion about it. Please continue reading articles in **one resort**, until you'll get the order to select the next resort. Please stay on the **NEWS** section of the page!

Comment: ideally you browse through 2-3 articles per minute

Please **do not surf the video or the picture section of the page**, we are just about news-articles (text and picture)!

Navigation Task

You are planning a weekend trip to Vienna, with an overnight stay at a hotel and a little bit of sightseeing.

- Open the start page by using the link on the left top of the screen.

- Browse to **„Vienna"** (by using the search field)

- Change to the satellite view. (Just click on the „satellite" button on the right top of the map.)

- Look for a hotel on the **Kärntnering** in Vienna.

- Hint: Insert „hotel, Kärntnerring" to the search field.

- Browse through the found hotels (by zooming and panning).

- Look for the hotel, which is nearest to „**Minoritenkirche**".

- Find the „**Naturhistorische Museum"**, the „**Bestattungsmuseum"** and the "**Arsenal"** in Vienna

- Get the directions between the named museums. Start and endpoint is your hotel.

- (To get the directions, click on the „get directions" button on the right side of your screen. Choose the starting point and add as much destinations as you'll need – right below the input field, is a „add destination" link. Finally click on „Get directions".)

- Zoom in so that you will be able to see the street names (by using the mouse wheel or the „+" at the left top of the map.)

- Explore the route (by panning).

- Change the destinations. Start at your hotel and get the direction to **„Donau City Str. 1"**.

Web Shop

URL: http://www.amazon.at
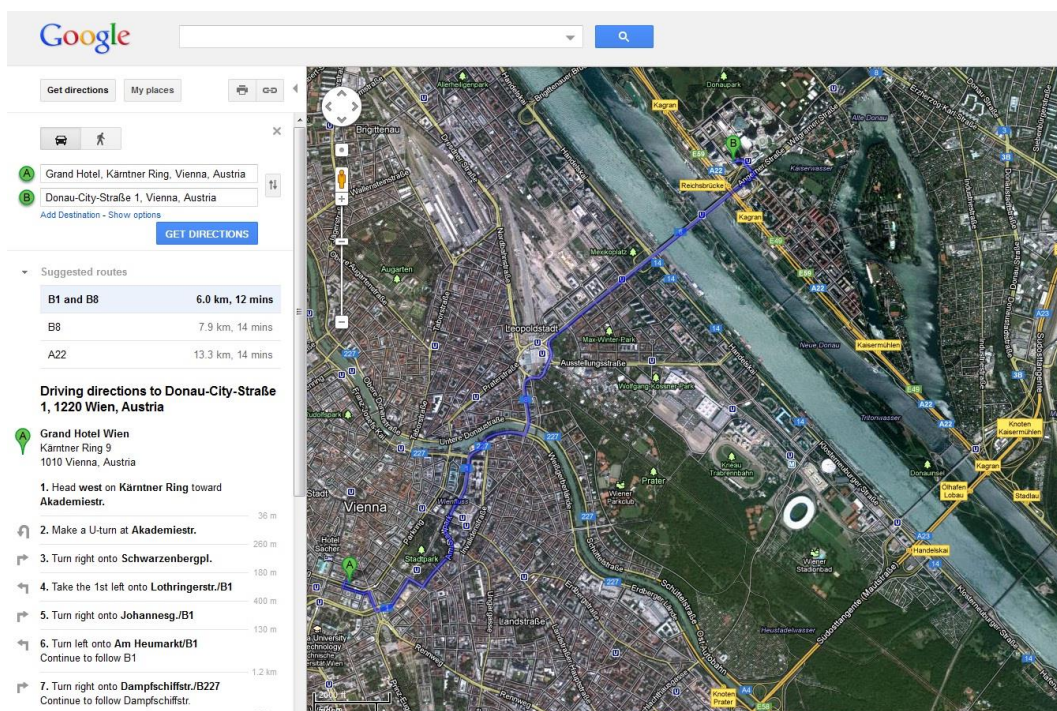
Please search for literature of choice. Suggestions:

a) Look for holiday reading material.

b) Search for an interesting book about Barack Obama.

c) Find a serious book regarding covering the reasons of the financial crisis.

d) Check the current Amazon charts for literature.

e) Search for a tourist guide for your next holidays.

f) Find the new book of Ken Follet.

g) Look for a self-help book on the subject of "Burn Out".

h) A cookbook with spring recipes.

Warning: Please **DON'T** watch any videos!

Appendix III

Example of questions for web browsing tests

(This appendix does not form an integral part of this Recommendation)

III.1 For assessment of media quality

Other examples then listed here can also be found in [ITU-T P.910] and [ITU-R BT.500]

III.1.1 "Please rate the image quality on the following scale"

The scale descriptors are:

- Excellent
- Good
- Fair
- Poor
- Bad

III.1.2 "Please rate the video quality on the following scale"

The scale descriptors are:

- Excellent
- Good
- Fair
- Poor
- Bad

III.2 For assessment of the temporal aspects

III.2.1 " Please rate the speed of the page rendering on the following scale "

The scale descriptors are:

- Excellent
- Good
- Fair
- Poor
- Bad

III.2.2 " Please rate the variability of the page rendering speed on the following scale (Excellent = very low variability, Bad = high variability)"

The scale descriptors are:

- Excellent
- Good
- Fair
- Poor
- Bad

————————————

INTERNATIONAL TELECOMMUNICATION UNION

COM 12 – C 46 – E

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 2013-2016

March 2013

English only

Original: English

Question(s): 7	March. 19th - 28th, 2013

STUDY GROUP 12 – CONTRIBUTION 46

Source:	A1 Telekom Austria AG
Title:	Draft Test Plan for P.STMWeb

1. Introduction

This document should give guidance for the conduction of a lab test on Web-QoE.

2. Test facilities

For the test facilities (client devices, network emulator etc.) and the respective setup the guidelines from [1] should be followed.

3. Test content

As content the news page and the photo gallery page from [2] should be used. Both webpages can be locally hosted on a dedicated webserver in order to rule out any interference due to network or Internet problems.

4. Test conditions and Test Design

The technical test conditions are described in Table 2. Each of the test conditions should be presented twice to the user such that in total 24 test conditions are achieved and each conditions is rated twice. The order of the conditions has to be randomized such that order effects can be outruled.

Downlink BW [kbit/s]	64	128	256	512	1024	2048	Conditions
News page	x	x	x	x	x	x	6
Photo page	x	x	x	x	x	x	6
RTT (fixed)	20 ms					**Total**	**12**

Table 2: Technical conditions for both content types

Contact:	Raimund Schatz Forschungszentrum Telekommunikation Wien, Austria	Tel: + 43 1 5052830 35 Fax: + 43 1 5052830 99 Email: raimund.schatz@ftw.at
Contact:	Sebastian Egger Forschungszentrum Telekommunikation Wien, Austria	Tel: + 43 1 5052830 12 Fax: + 43 1 5052830 99 Email: egger@ftw.at

For the test design each condition should last for 150sec and then the user should be prompted with the questionnaire as described in Table 4 (avg. time for questionnaire is 30sec → in total 180sec per condition incl. rating). The time schedule for the test is described in Table 3. For the warmup conditions one conditions should be the lowest downlink bandwidth (64 kbit/s) and one condition should be the fastest downlink bandwidth (2048 kbit/s). The third warmup condition can be randomly chosen from the remaining downlink bandwidth conditions (128, 256, 512, 1024 kbit/s). Also the content of the warmup conditions can be randomly chosen.

Task	Duration [sec]	Conditions
Instructions	300	
Warmup	540	3
Webpage 1/2	2160	12
Break	600	
Webpage 2/2	2160	12
Total	**5760**	**27**
	96 min	

Table 3: Test Design and Timings

The questions which have to be asked after each condition are described in Table 4.

Please rate the overall quality of the connection you have been using on the following scale:

Bad	**Poor**	**Fair**	**Good**	**Excellent**
O	O	O	O	O

Would you accept this connection quality in <u>normal life at home</u>?

Yes	**No**
O	O

Table 4: Questions for the test users

5. Result Presentation

The results should be presented as described in Annex A in [1].

6. References

[1] Egger, S. and Schatz R., ITU-T Study Group 12 – Contribution 337, May 2012, Geneva, Switzerland

[2] Sebastian Egger, Raimund Schatz "Interactive Content for Subjective Studies on Web Browsing QoE: A Kepler Derivative " ETSI STQ Workshop on Selected Items on Telecommunication Quality Matters, 27-28 Nov, 2012, Vienna, Austria.

ANEXO II
Contribución al Draft P.STMWeb

A continuación se presenta la contribución realizada desde la Universidad del País Vasco a la validación del Draft P.STMWeb.

Se incluye una descripción sencilla de la arquitectura desplegada, las consideraciones adoptadas, los resultados obtenidos y propuestas de mejora.

INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 2013-2016

COM 12 – C 0124 – E

December 2013

English only

Original: English

Question(s):	7/12

STUDY GROUP 12 – CONTRIBUTION 0124

Source:	University of the Basque Country
Title:	Test results and new proposals for enhancement of Draft P.STMWeb

1. Introduction

During the last two months, a set of subjective testing of user perceived quality of web browsing have been carried out at the University of the Basque Country in order to validate the method and procedures proposed in the Draft P.STMWeb.

2. Test facilities

The test facilities used for the subjective testing included: an end user device, a network emulator, a router and web servers. Network parameters (BW and delay) were set through a network emulator (EMU) which was placed between the user device (UD) and the router (RT), ensuring the routing between eventual local content servers and the internet, as recommended in Draft P.STMWeb.
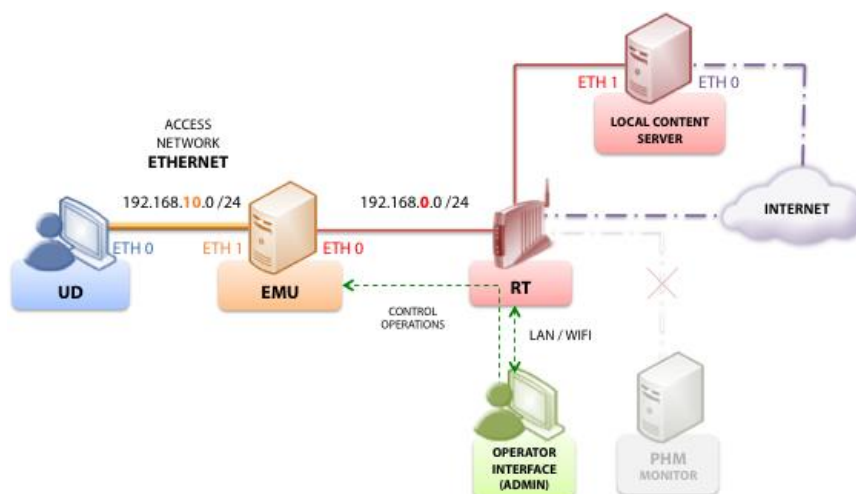
Figure 2: Test facilities

Contact:	Eva Ibarrola University of the Basque Country (UPV/EHU), Spain	Tel: +34-94 601 39 00 Fax: +34-94 601 42 59 Email: eva.ibarrola@ehu.es
Contact:	Eduardo Saiz University of the Basque Country (UPV/EHU), Spain	Tel: +34-94 601 73 52 Fax: +34-94 601 42 59 Email: eduardo.saiz@ehu.es

Description of the test facilities:

- *End user device*

  Hardware (PC)

Brand	Dell
Model	OptiPlex 790
Processor	Intel Core i3 – 3.10 GHz
Memory (RAM)	4 GB
Network Interface	Ethernet 10/100

  Software

Operating System	Windows 7 Professional SP1 – 32 bit
Browser	Firefox 24 – All caches set to size 0

  Display

Screen	17" (4:3)
Resolution	Screen: 1280 x 1024 Browser: 1200 x 650 - User can resize/maximize window
Luminance	250 cd/m2
Display Calibration	All displays calibrated to D65

- *Network Emulator*

Brand	Dell
Model	OptiPlex GX 620
Processor	Intel Pentium 4 HT
Memory (RAM)	2 GB
Network Interface	2 x Ethernet 10/100
Operating System	Ubuntu Server 12.04 LTS – Linux 3.2.0
Software	Apache2, SSH (remote management), tc, wondershaper

- *Content Server*

  Hardware (PC)

Brand	Dell
Model	OptiPlex GX 620
Processor	Intel Pentium 4 HT
Memory (RAM)	2 GB
Network Interface	2 x Ethernet 10/100

  Software

Operating System	Ubuntu Server 12.04 LTS – Linux 3.2.0
Web content	Apache2
Registration/Session	JSF - Glassfish 3.1.2
Database	MySQL

- *Router*

Brand	Netgear
Model	WGR614 v7
Network Interface	4 ports LAN Ethernet / WiFi Access Point

Content Server has been set up with an extra network interface directly attached to the Internet. Since news are updated every few hours, the new contents can be maintained in the correspondent database throughout that second interface, instead of crossing the test LAN that might interference any test being held at that precise moment.

3. Test Content

The test results presented in this contribution refer to two different web page classes: News webpage and photo album, both locally served from the Content Server.

The main structure of both webpages is based upon the works of [FTW]. However, some modifications have been made to reflect a more local content for the group of testers.

News Webpage

- Locally hosted (Served by Apache2) [FTW].
- HTML PHP5 and JavaScript are the main languages used.
- All news contain one picture (JPEG).
- News are fetched from the Spanish and Basque RSS feeds of www.europapress.es
- Appearance has also been mimetized to resemble www.europapress.es
- A content update script has been scheduled to be executed three times a day
- Average page size depends on the length of the news selected:
    - o Average news length (picture included):   379 bytes
    - o Standard deviation for news content: 73 bytes
    - o News page structure (constant size):   3,5 KB
    - o Total Average Length   ~3,9 KB

Photo Album

- Locally hosted (Served by Apache2)
- HTML, PHP5 and JavaScript are the main languages used.
- Appearance has been mimetized to resemble www.facebook.com
- There are five photo albums with an average of 40 photos per album.
    - o Average photo size: 114,50 KB
    - o Standard deviation of photo sizes: 10,15 KB

4. Test Participants

All the selected subjects (a total of 46) were experienced with the devices under test and the use of web pages as recommended in Draft P.STMWeb. A group of undergraduate 23 students (in last course of Telecommunication Engineering Degree) has been considered as the non-expert users and another group of participants, composed of 23 professors and researchers, has been defined as the expert group.

	Expert Group	Non-Expert Group
Number	23	23
Long test version	12	23
Short test version	11	-
Gender	18 Male (78.26%) 5 Female (21.74%)	15 Male (65.22%) 8 Female (34.78%)
Mean Age	30.65	21.83
Median Age	28	21
Web usage (hours/week)	More than 20 hours/week	15 to 20 hours/week
Experience with web pages	15 years using Internet	10 years using Internet

5. Test conditions and Test Procedure

The test has been designed according to the Draft Test Plan for P.STMWeb defined in COM 12 – C 46 – E. Therefore, each of the test conditions (Table 2) have been presented twice to the user such that in total 24 test conditions are achieved and each conditions is rated twice. The order of the conditions has been randomized such that order effects can be outruled.

Downlink BW [kbit/s]	64	128	256	512	1024	2048	Conditions
News page	x	x	x	x	x	x	6
Photo page	x	x	x	x	x	x	6
RTT (fixed)	20 ms					**Total**	**12**

Table 5: Technical conditions for both content types

For the non expert group, the test conditions have been maintained as defined in P.STMWeb:

Task	Duration [sec]	Duration [min]	Conditions
Instructions	300	5	
Warmup	540	9	3
Webpage 1/2	2160	36	12
Break	600	10	
Webpage 2/2	2160	36	12
Total	**5760 sec**	**96 min**	**27**

Table 6: Test Design and Timings

Half of the expert group participants took the test under these conditions. However, since most of them complained about the time-consuming of the test, the other half was presented the same conditions under shorter timings. Therefore, as recommended in COM 12 – C 46 – E, original time length for each condition has consisted of 150 seconds of content navigation plus 30 seconds for the evaluation of the condition presented but those lengths have been reduced to 100 seconds and 20 seconds respectively for the short test version. This implies one minute less per each condition and nearly half an hour less for the whole test completion:

Task	Duration [sec]	Duration [min]	Conditions
Instructions	300	5	
Warmup	360	6	3
Webpage 1/2	1440	24	12
Break	600	10	
Webpage 2/2	1440	24	12
Total	**4140 sec**	**69 min**	**27**

Table 7: Test Design and Timings (Short version)

6. Test results

In this section we present the test results (MOS with 95% confidence intervals) classified both for content type and user categories and some results comparisons.

6.1 General Results

In Figure 2 the average MOS results by content type for the totality of the participants is presented (experts and non-experts included).
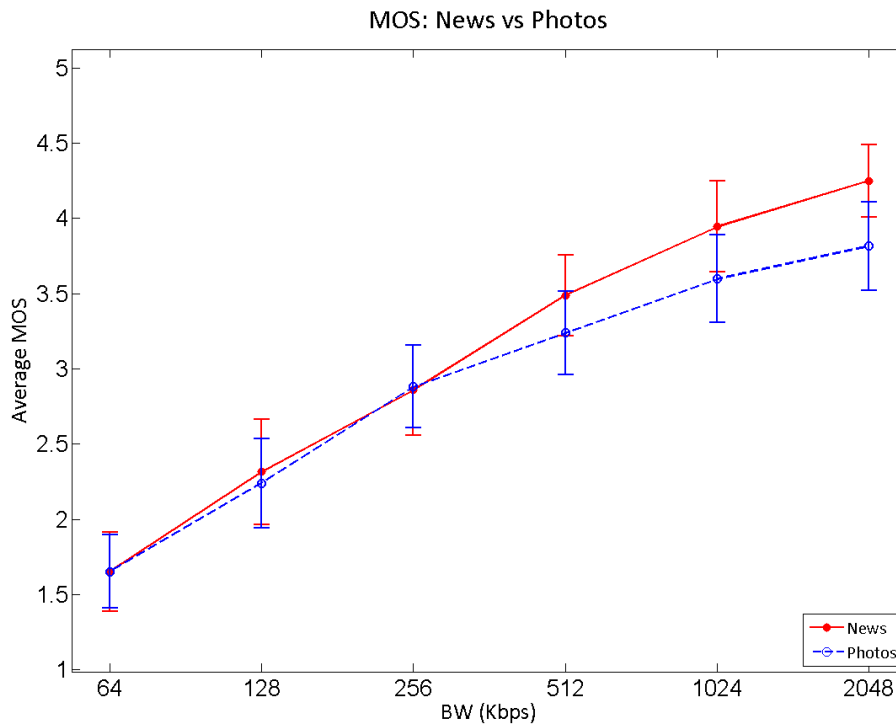
Figure 2: MOS vs BW (results for 46 participants)

Both content categories present quite linearity in the results. However, a lower tolerance to contents with higher page weights can be observed, as photos are rated lower. This effect can be better appreciated in faster conditions, where news text is loaded quite quickly in comparison to the heavy images of the photo contents. When in worse conditions, since the loading times are not good even for news contents, both graphics are quite similar.

6.2 Results classified by user´s category and round

In Figure 3, the general results are presented, classified by user category and round. In these graphics we can appreciate more consistencies in the expert's results, as non-experts graphics are more irregular. However, a clear tendency can be observed in all the graphics: the better the condition, the higher the rating.
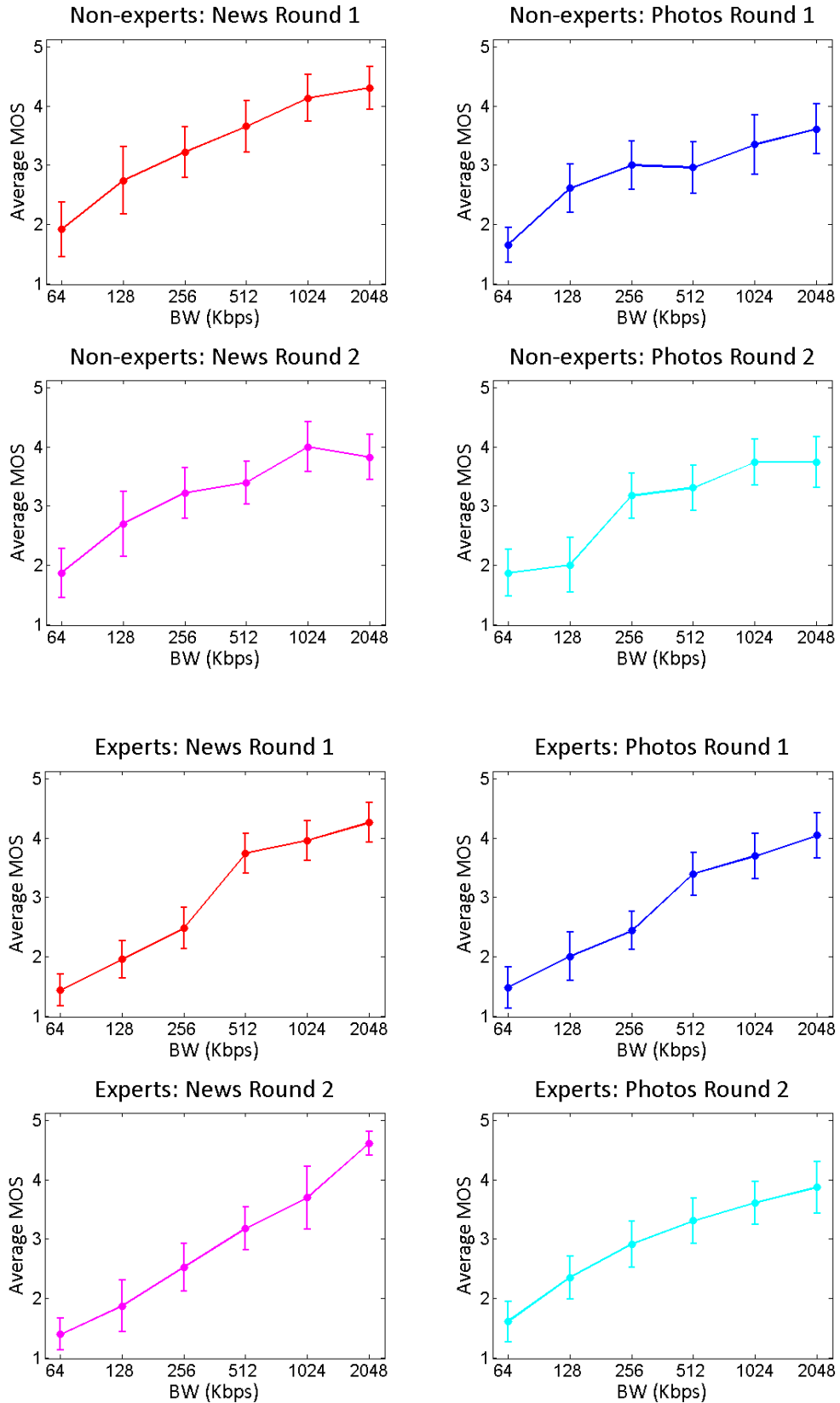
Figure 3: MOS vs BW (classified by user category and round)

6.3 Results comparison: Experts and non-experts

In figure 4, the comparison between experts and non-experts results are shown. It can be perceived that, in general, experts seem more demanding than non-experts. This effect is easily seen in the news category. However, there is no clear tendency for photo contents.
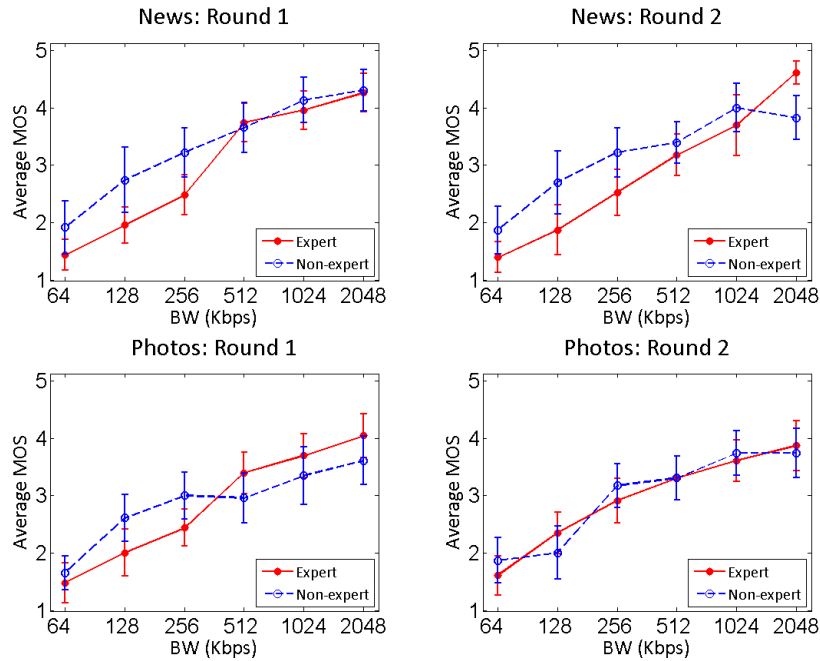
Figure 4: Comparison experts and non-experts

6.4 Results Comparison: Long and Short tests (expert users)

Figure 5 shows the comparison between results in the long and short test in the expert category. There is no clear tendency, but it seems that the longer the test is, the worse the results are.
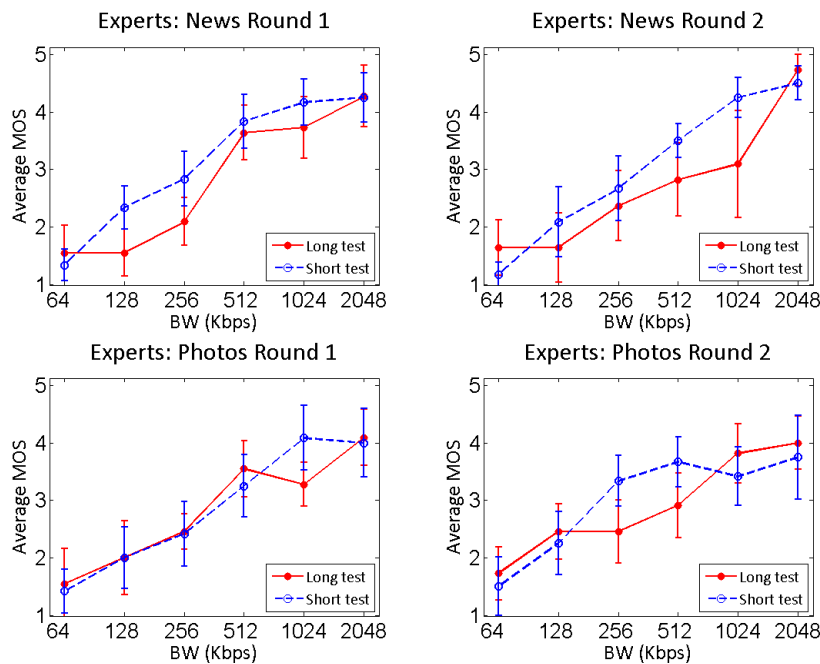
Figure 5: Comparison between long and short test

6.5 Results per round and user category with content-types comparison

In figure 6 we can observe the comparison between results in news session and the photo session results.

If we focus on the non-experts results, we can appreciate that heavier contents are rated lower. Also, in round 2 there seem to be more irregularities, as the best condition (6th) is rated lower than the fifth one.

For experts, however, there is not a clear tendency between content-types. Although in round 1, photos are rated lower; in round 2 the results show mostly the opposite. It could make sense since they are aware of the procedure of the test. Quite a linear tendency when considering conditions for each content type can be observed.
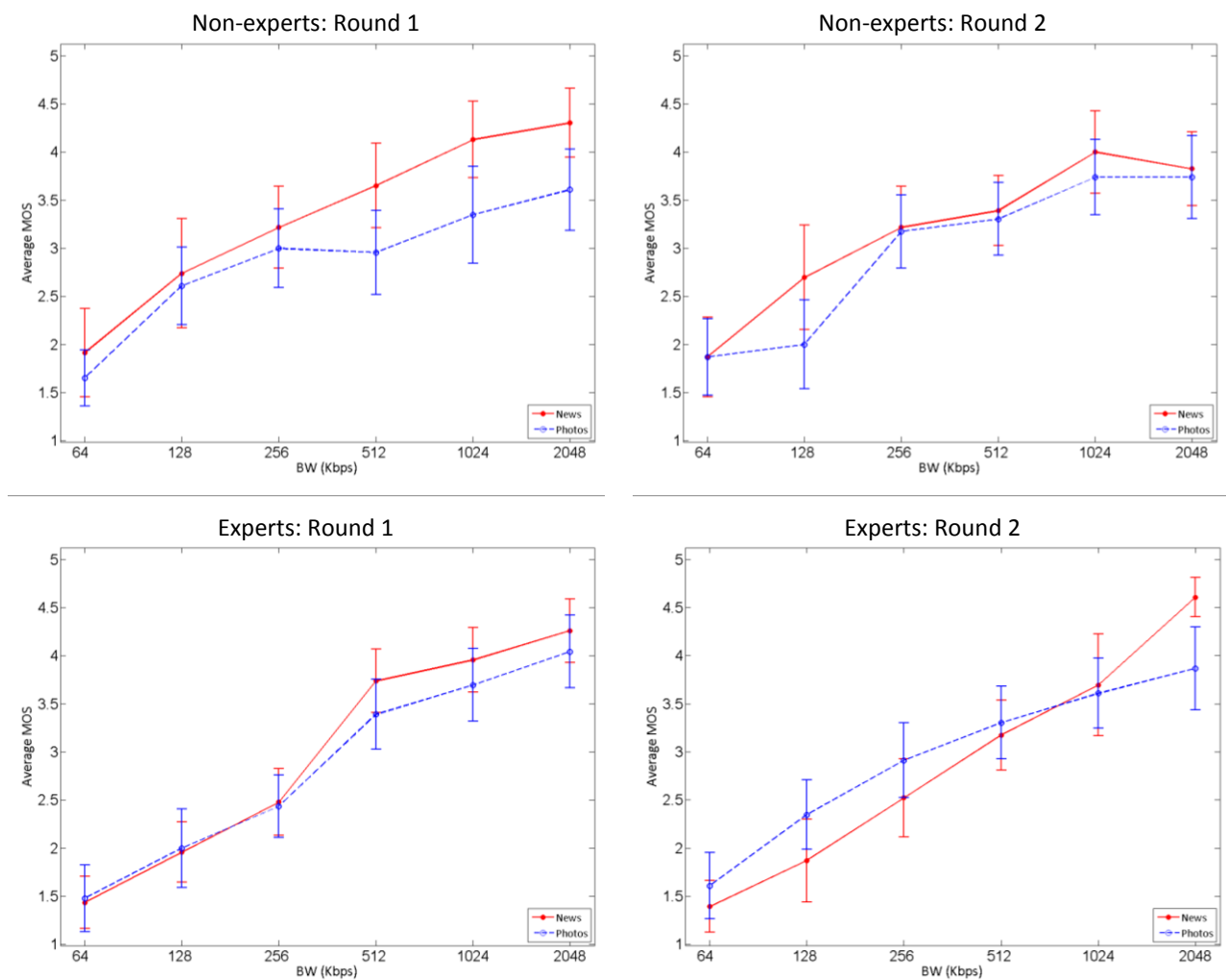
Figure 6: Content-types comparison

6.6 Results per content and user category with rounds comparison

In figure 7, a comparison between the results in the first and the second round is shown.

News Content

For non-experts, similar means between rounds are presented. They tend to rate lower in second round. Perhaps due to lower concentration levels.

Experts show better results in second round.

Photo Album

When surfing photo galleries, experts have more reliable perception of the test conditions. Their results present a more lineal response. Anyhow, the responses of both experts and non-experts have a clear ascendant tendency with better test conditions.

Irregularities in linearity for non-experts can be explained with their difficulty of perceiving the required loading time of different photo sizes in each test condition.
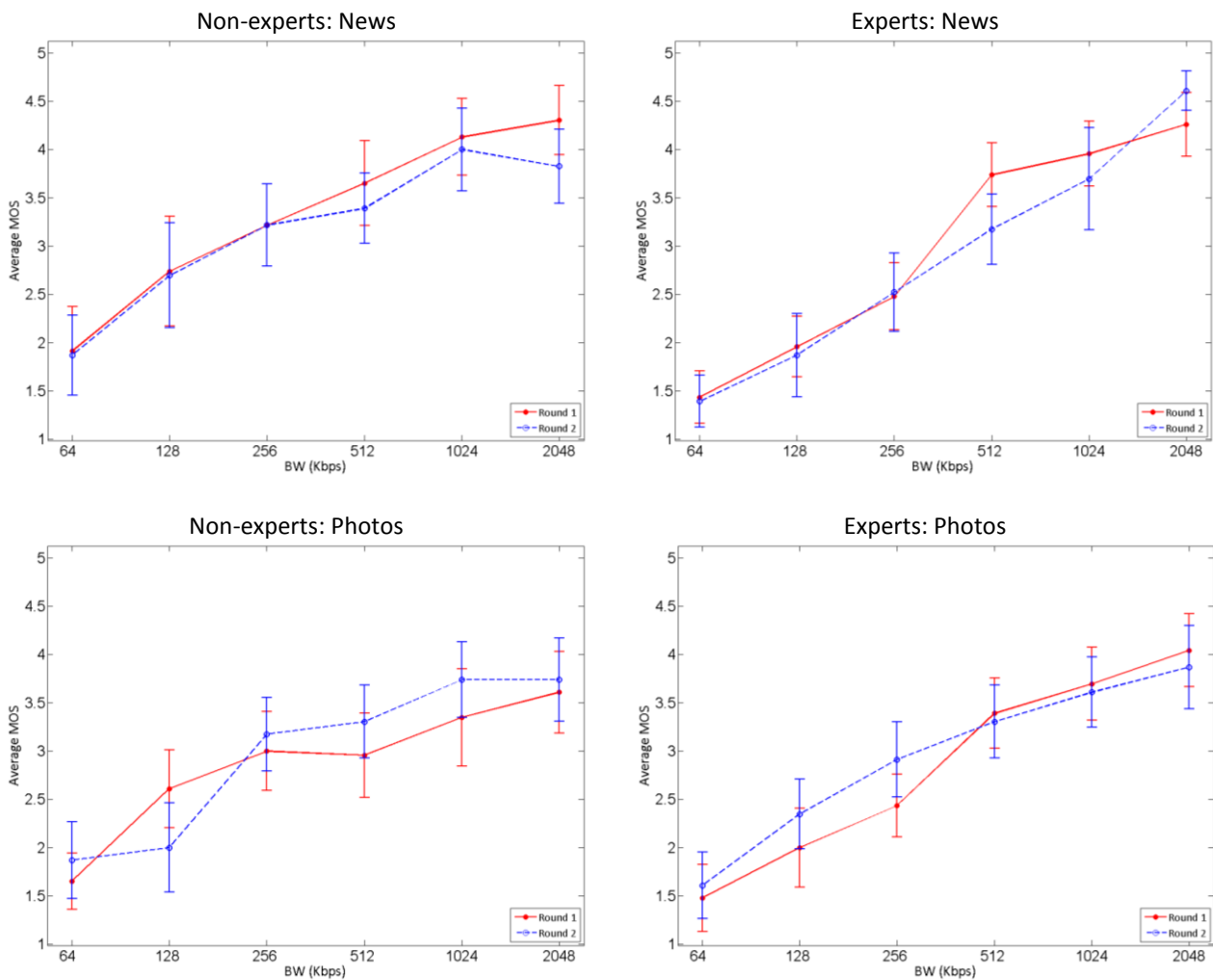
Figure 7: Rounds comparison

6.7 Response to: "Would you consider this connection at home? (YES/NO)?"
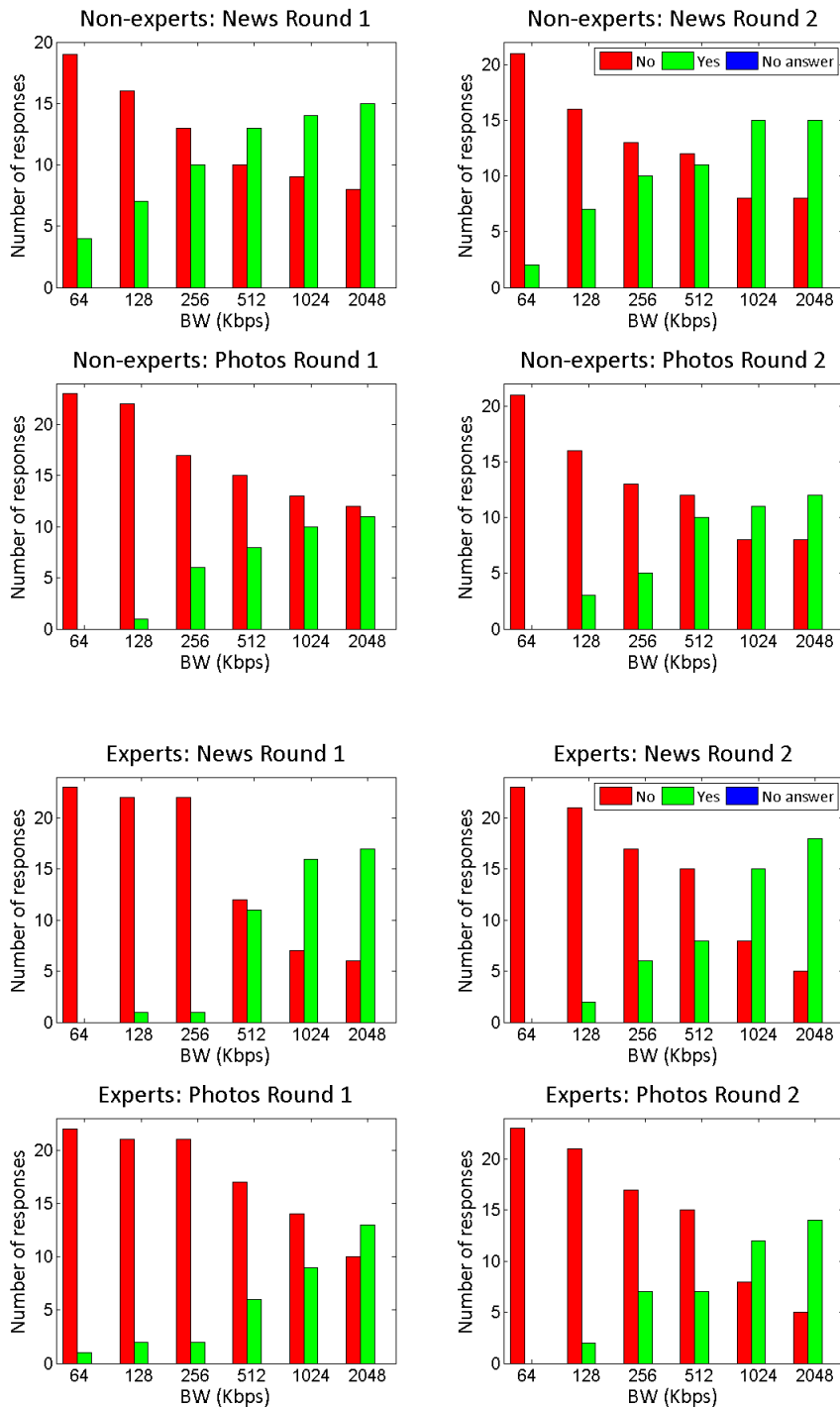
[√]   Value 'YES' increases with better conditions. Value 'NO' decreases with better conditions.
[√]   No answer was left unanswered.
[√]   Experts are more demanding than non-experts. Less 'YES' than non-experts

6.8 Results of standard deviations

EXPERTS: STANDARD DEVIATION

News: Round 1	0.6624	0.7674	0.8458	0.8100	0.8245	0.8100
News: Round 2	0.6564	1.0576	0.9941	0.8869	1.2946	0.4990
Photos: Round 1	0.8458	1.0000	0.7878	0.8913	0.9261	0.9283
Photos: Round 2	0.8388	0.8847	0.9493	0.9261	0.8913	1.0576

NON-EXPERTS: STANDARD DEVIATION

News: Round 1	1.1246	1.3888	1.0426	1.0706	0.9679	0.8757
News: Round 2	1.0137	1.3292	1.0426	0.8913	1.0445	0.9367
Photos: Round 1	0.7141	0.9881	1.0000	1.0651	1.2288	1.0331
Photos: Round 2	0.9679	1.1282	0.9367	0.9261	0.9638	1.0539

Conclusions

The MOS distribution and statistical results of the experiment demonstrate the validity of the methods and procedures as defined in Draft P.STMWeb.

In general, the MOS score is larger as the bandwidth grows up and the results show also that the task the user is evolved in and other contextual aspects, as the user´s experience, may influence in the user´s web browsing perception.

Proposal for enhancement of Draft P.STMWeb

In this section, some aspects that may aid to enhance the description of the procedures proposed in Draft P.STMWeb are presented. Most of them have emerged during the start-up of the experiment and from the comments of some of the expert participants.

1. Since it was not specified, all the test experiments were conducted with the cache disabled. This is something that should be specified in the test procedure since the results may vary a lot with/without cache.
2. Many of the users answered that would NOT accept any of the connection quality in normal life at home, so the conclusion is that the BW range should be accommodate to users requirements (we used the ones proposed in the Draft test plan). From our point of view, in the procedure should be specified that it is important to accommodate the test conditions to the range of the user's expectations.
3. Many of the participants complained about the time that it takes to go through the test whole experiment. That is the reason that, for the expert users, the session time was reduced from 150 sec to 100 sec. Some of the expert users suggested that it could be interesting to provide a button during the test to give response as soon as you got it.
4. The task may have a big influence in the results so it may be considered a more detailed explanation about the task definition and its influence.

ANEXO III
Diseño de la plataforma de evaluación de QoE

Las siguientes páginas recogen la descripción de la plataforma de evaluación de QoE desarrollada para la validación del draft P.STMWeb, en un documento interno redactado como extensión de la contribución COM 12 – C 0124 – E (Anexo II).

Se trata de un documento explicativo de los servidores del sistema, las bases de datos utilizadas y extractos de código de las páginas web y *scripts* desarrollados.

**NETWORKING, QUALITY
AND SECURITY**

RESEARCH GROUP

English Only

Contribution: COM 12 – C 0124 – E

Title: INTERNAL DOCUMENT ON PLATFORM DESIGN

Introduction

During the last two months, a set of subjective testing of user perceived quality of web browsing have been carried out at the University of the Basque Country in order to validate the method and procedures proposed in the Draft P.STMWeb.

Test facilities

2.1. General model

The general scheme for the subjective tests included: an end user device, a network emulator, a router and web servers. Network parameters (BW and delay) were set through a network emulator (EMU), which was placed between the user device (UD) and the router (RT), ensuring the routing between eventual local content servers and the internet, as recommended in Draft P.STMWeb.
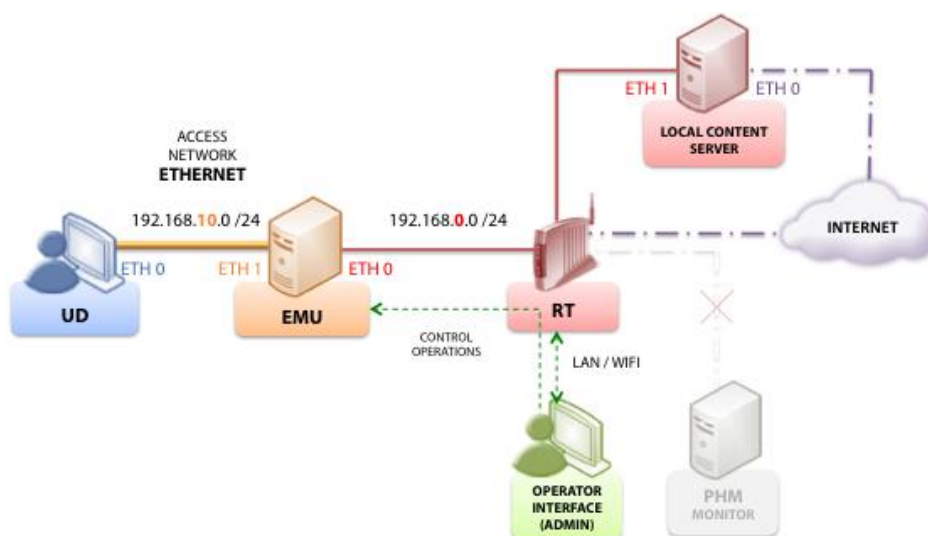
Figure 3 – Test facilities: basic scheme

However, the architecture defined for the tests of this particular contribution differ in some points:

- Since the test that has been carried out consisted only in serving contents obtained from the local Content Server (CS), there is no need of defining a Page Health Monitor (PHM) as in Draft P.STMWeb to monitorize the availability of external links or contents.

- There is a second difference to the architecture proposed in the draft. The Content Server has an extra network interface directly attached to the Internet. Since news have been defined to be updated every few hours, the new contents can be persisted in the correspondent database throughout that second interface, instead of crossing the test LAN that might interference any test being held at that precise moment.

2.2. Implemented model

Asides from what has been defined above, since the testing period in which participants took the tests was not enough for the accommodation of all of them within the available time slots, the need of enabling more UD units aroused. For this matter, the finally implemented model consists of the same general model units but with additionally aggregated UD and EMU devices, as seen below.
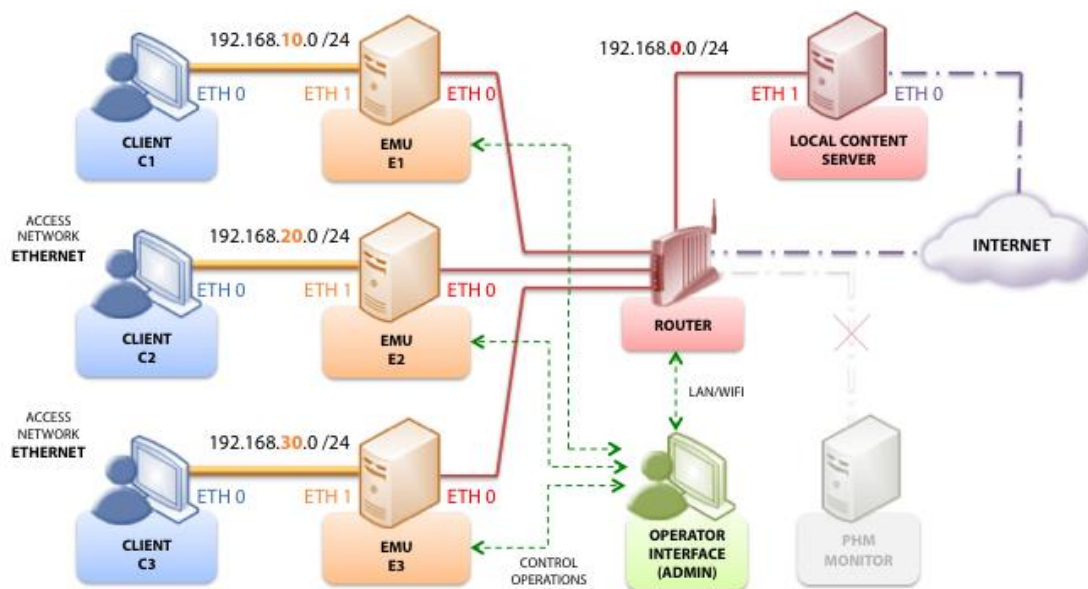
Figure 4 – Test facilities: implemented scheme

All 3 UD clients (C1, C2 and C3) had been directly connected to their respective EMUs (E1, E2 and E3). This link between each pair of UD-EMU is where the different conditions are applied. All the three EMUs are then connected to the controlled LAN, as seen painted in red in figure 2. This LAN is settled by a router working at full speed (100Mbps) in wired mode only (Wi-Fi has been disabled during the whole test period).

Given the speed conditions defined for the tests (ranging from 64 to 2048 Kbps) and the capabilities of the LAN defined to interconnect the EMU and the local Content Server (CS), it can be assured that the traffic aggregation by the three parallel branches in the worst case scenario (three users testing the better condition simultaneously at 2048 Kbit/s per test) is lower than the capacity of the test LAN at is worst. Therefore, there is no chance of having one test influence the speed and results of the other since the bottle neck is not in the LAN but in the respective UD-EMU ad-hoc links.

In the next page we present a table with the specific characteristics of the devices used in the tests.

As seen in that table, two different end user devices have been used in the tests provided the incapability of finding 3 equal user devices for the tests. Despite this, no other software was running in background when performing the tests so CPU and memory usage in both computer types never reached high levels that might compromise the validity of the results. Network interfaces in all of them are also of the same brand and kind. Under these considerations, their different internal architectures had no significant effect on the results.

2.3. Description of test devices

- *End user device*

Hardware (PC)	Client #C1	Clients #C2 & #C3
Brand	Dell	Dell
Model	OptiPlex 790	OptiPlex 745
Processor	Intel Core i3 – 3.10 GHz	Intel Core 2 Duo – 2.13 GHz
Memory (RAM)	4 GB	2 GB
Network Interface	Ethernet 10/100	Ethernet 10/100

Software

Operating System	Windows 7 Professional SP1 – 32 bit
Browser	Firefox 24 – All caches set to size 0

Display

Screen	17" (4:3)
Resolution	Screen: 1280 x 1024 Browser: 1200 x 650 - User can resize/maximize window
Luminance	250 cd/m2
Display Calibration	All displays calibrated to D65

- *Network Emulator*   All EMUs #E1, #E2 & #E3

Brand	Dell
Model	OptiPlex GX 620
Processor	Intel Pentium 4 HT
Memory (RAM)	2 GB
Network Interface	2 x Ethernet 10/100
Operating System	Ubuntu Server 12.04 LTS – Linux 3.2.0
Software	Apache2, SSH (remote management), tc, wondershaper

- *Content Server*

Hardware (PC)

Brand	Dell
Model	OptiPlex GX 620
Processor	Intel Pentium 4 HT
Memory (RAM)	2 GB
Network Interface	2 x Ethernet 10/100

Software

Operating System	Ubuntu Server 12.04 LTS – Linux 3.2.0
Web content	Apache2
Registration/Session	JSF - Glassfish 3.1.2
Database	MySQL

- *Router*

Brand	Netgear
Model	WGR614 v7
Network Interface	4 ports LAN Ethernet / WiFi Access Point

Test devices: Internals

Each of the aforementioned devices (UD, EMU, CS and RT) has specific functionalities that have been specifically designed and deployed for the correct execution of the tests. In the following paragraphs, the inner development (including some script specifications) is presented.

3.1. User Devices (Clients)

The three UD used for the tests of this contribution need no special configuration apart from the Internet connectivity through the respective EMUs. For these tests, all UD use the same OS (Windows 7 Professional 32bit) and browser (Firefox 24).

Since the Draft P.STMWeb does not specify anything about browser caches, all the test experiments were conducted with the cache disabled. The reason is to avoid any content (text or photos) to be loaded from cache instead of using the network access at the specific condition being tested.

6.2. Network Emulator

Image 3 presents the functional architecture of the EMU. Basically, the EMU is a server that not only applies the condition parameters to the ad-hoc link between the EMU itself and the UD, but also acts as a gateway for the UD to the rest of the LAN (and the contents in the CS).
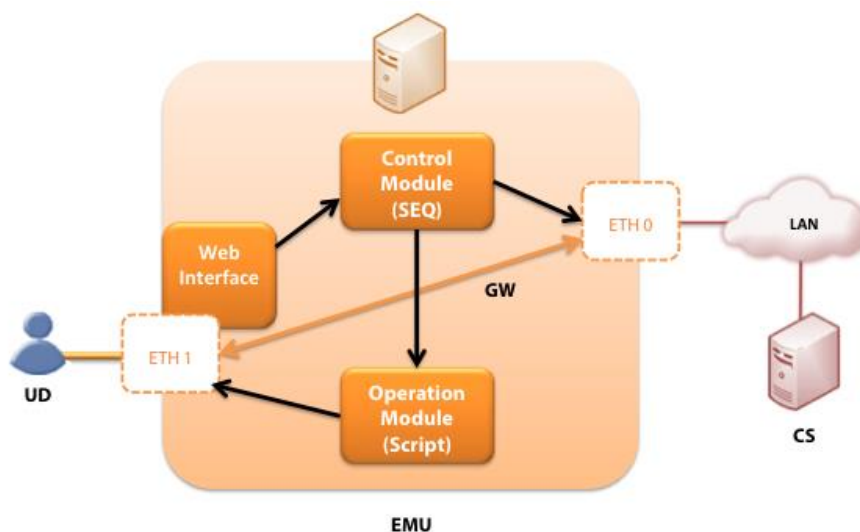
Figure 5 – EMU: Functional architecture

It mainly consists of a Control Module, which acts as a sequencer (SEQ), an Operation Module (OM), which runs a script to modify the network interface conditions, and a Gateway Module (GW), which implements the specific routing orders for enabling the communication between UD and CS.

Control Module (Sequencer)

When user accesses the EMU, he is presented a webpage hosted in the EMU (served by Apache2) with a very simple menu that consists of four options: Register, Start training, Start Round 1 and Start Round 2. Only one of these options is available to click at a time, forcing the user to follow the test phases as defined in Draft P.STMWeb. For a better description of this module, it will be explained by following the navigational flow as perceived by the user.

Step 1: The user clicks in the Registration button. Another browser window emerges over the existing one, with the registration form served by the CS. This is, the SEQ forwards the request to the CS and this presents the form to the user. After registration is completed, the user is prompted to close the secondary window. At this time, Training button in the main browser window is ready to be clicked.
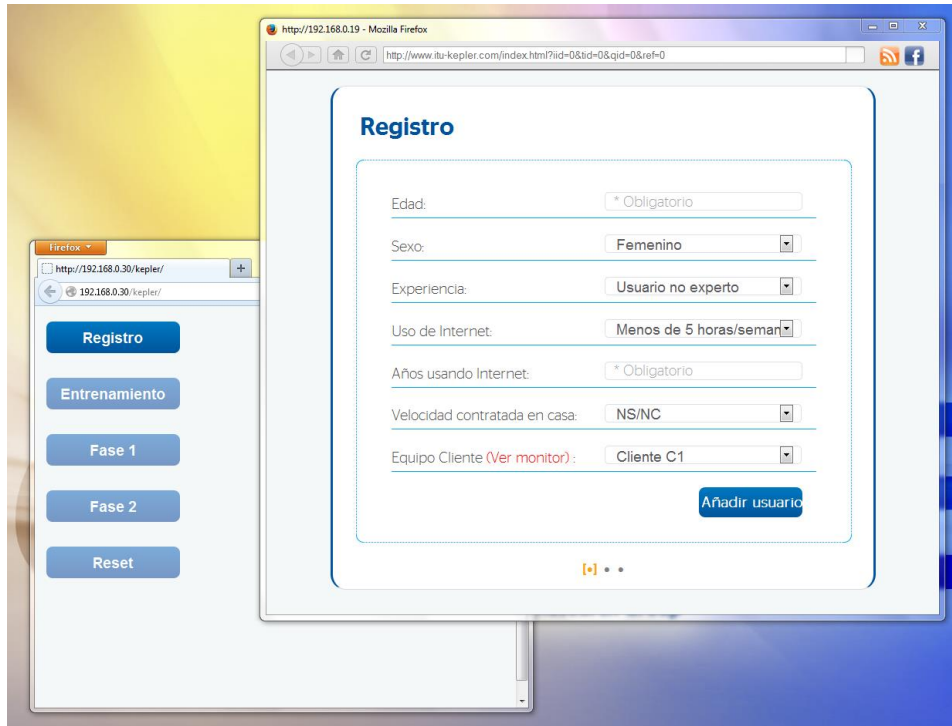
Figure 6 – EMU: Sample capture at Register stage

Step 2: The user clicks in the Training button. SEQ processes the order and launches a series of operations according to a defined time schedule. In other words, user is presented a new browser window in which different contents will be loaded according to SEQ orders. Those orders consist of applying three different test conditions to the UD link while user surfs the CS contents and evaluates his perception.

In the first part of each of those three iterations, the CS is asked to load for the UD in that second window the news or photo contents for the free navigation period, while, at the same time, the OM is asked to vary the network interface parameters in the UD link according to the specific test condition being held.

After the time for this part of the iteration ends, the SEQ asks the OM to reset the network interface parameters and forwards the CS the request to present the UD the evaluation form of that condition in the window where the news/photo contents had been shown.

The user then sends his answers to the CS and awaits for the next iteration to begin. After the three iterations are completed, the user is prompted to close the second window.

Step 3: The user has ended the training session and now launches Round 1 from the main browser window, which has to be active an unclosed during the whole test. The inner functioning of this step is quite similar to the previous one. However, this time 12 iterations will be presented to the user regarding the 6 speed conditions to be presented for each of the 2 content types (news and photos).

Conditions are presented to the user in a random scheme, or, more accurately, a semi random scheme, since randomization occurs only in speed values. News and photo contents are shown sequentially for each speed randomization. Again, after the 12 iterations are completed, user is prompted to close the second window.

Step 4: User now can have a break for 10 minutes. The EMU webpage remains opened during this time without activity. After the break, he can click on the last button (Round 2) for the final part of the test. Both Round 1 and 2 buttons lead to the same operational sequence, so no further explanation will be given.
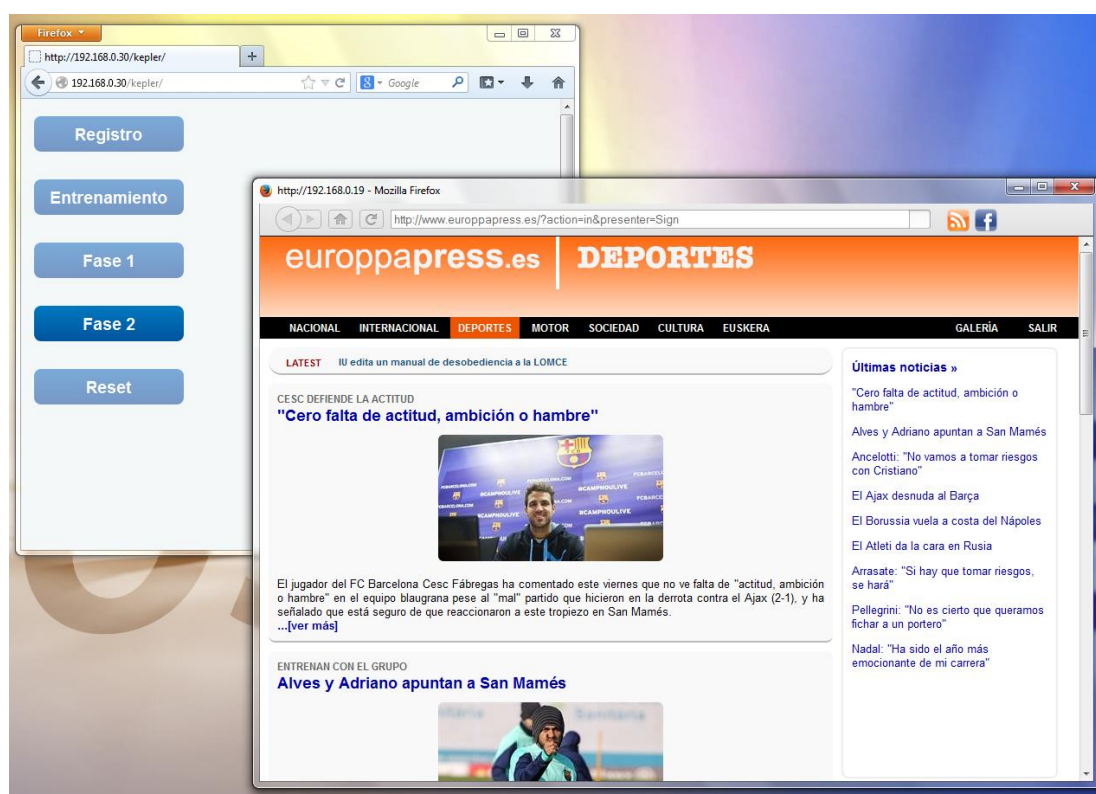
Figure 7 – EMU: Sample capture at Round 2 stage, surfing news content

For the better comprehension of SEQ, an extract of its code is presented in APPENDIX I, Section 1.

Operation Module

Whenever the SEQ requires the network interface to change its attributes, the Operation Module is called. This module consists of a simple script capable of launching network commands from application level (as seen in figures 7 to 9), by means of the Apache2 user (*www-data*).

Since those shell commands are root privileged, the Apache2 user has to be included in the */etc/sudoers* configuration file (figure 6). With this inclusion, www-data is capable of executing the *tc* and *wondershaper* commands, which are located in the */sbin* folder of the EMU.

```
/etc/sudoers

root      ALL=(ALL:ALL) ALL
www-data ALL=(ALL:ALL) NOPASSWD:/sbin/*
```

Figure 8 – EMU: Enable root options for www-data

```
<script>

// ASK OPERATION MODULE TO APPLY CONDITIONS

function setConditions(vi, vt, vq) {

    openWindow(vi,vt,vq);

    var act="del";
    if(vq%2==1) {
        act="add";
         $('#framed').attr('src','nav/cmd.php?tid='+vt+'&act='+act);
    }
    else $('#framed').attr('src','nav/cmd.php');
}
 </script>
```

Figure 7 – Call to Operation Module from Sequencer

```php
<?php
    cmd_string = "sh /scripts/www/tc_netem.sh ".$delay." ".$sp_up." ".$sp_dw;
    exec($cmd_string, $output);
?>
```

Figure 8 – EMU: Sample code of OM Application Script (cmd.php)

```
# ETH1 --> EMU-UD Link

# NOT USED
sudo tc qdisc add dev eth1 root tbf rate $1 burst 100k latency 20ms

# FINALLY USED COMMANDS
sudo tc qdisc add dev eth1 root netem delay $1ms
sudo wondershaper eth1 $2 $3

# USAGE:  wondershaper INTERFACE DOWN_KB UP_KB
# SAMPLE: wondershaper eth1 1024 2048

        # NOTE:
        # EMU UP   = UD DOWN = 2048
        # EMU DOWN = UD UP   = 1024
```

Figure 9 – EMU: Sample code of OM Shell Script (tc_netem.sh)
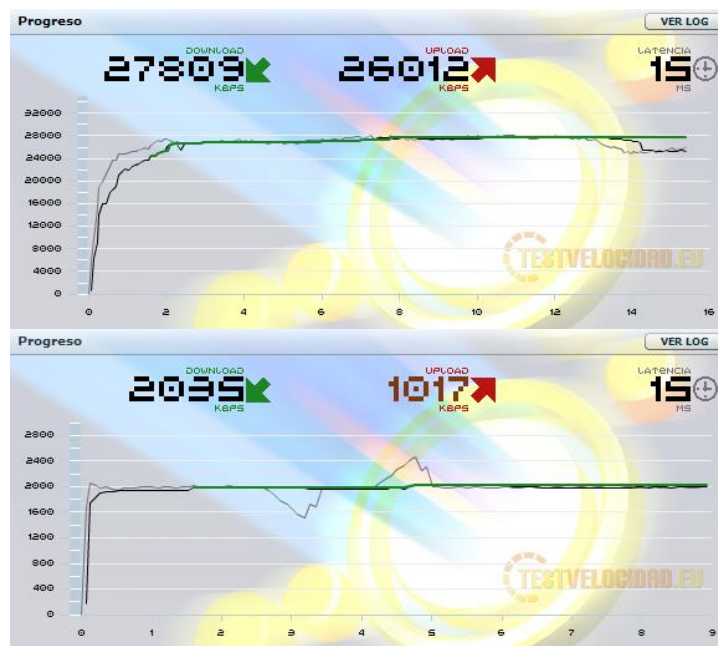
Figure 10 – EMU-UD link unlimited vs wondershaper speed limit to 2048k/1024k

For the network conditions to be applied, we started using **tc/netem** only. However, after some third party speed tests, we found some difficulties in applying the method ('not used' line in Figure 9). Speed was not properly limited to the one defined in the condition and the network interface seemed to ignore the restrictions.

For this reason, we decided to use an alternative combination of commands. We kept using **netem** for introducing the 20ms delay whilst the speed was finally governed by **wondershaper**, which provided a much better upper limit to the link (Figure 10).

NOTE: We must be careful when defining the **wondershaper** parameters for an asynchronous link when using the following command, we must be sure what the values of *down_kbps* and *up_kbps* are:

```
$ sudo wondershaper interface down_kbps up_kbps
```

The downlink speed in UD corresponds to the uplink speed in the EMU, and viceversa (figure 9). Therefore, if we want to evaluate a 2048/1024 kbps network connection in the UD, we'll have to execute the following command (provided that the EMU's network interface in the EMU-UD link is 'eth1'):

```
$ sudo wondershaper eth1 1024 2048
```

Gateway functionality

Finally, concluding with the EMU modules description, there is a special mention to the Gateway functionality. Since the EMU is neither a router nor a gateway per se, there was the need of allowing transparent communication between the UD and the rest of the LAN for all the data transactions not directly related to the EMU itself.

The following image (figure 11) presents the *iptables* configurations required for this purpose.

```sh
#!/bin/sh
PATH=/usr/sbin:/sbin:/bin:/usr/bin

# Delete all existing rules.
iptables -F
iptables -t nat -F
iptables -t mangle -F
iptables -X

# Always accept loopback traffic
iptables -A INPUT -i lo -j ACCEPT


# Allow established connections, and those not coming from the outside
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -m state --state NEW -i !eth1 -j ACCEPT
iptables -A FORWARD -i eth1 -o eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT

# Allow outgoing connections from the LAN side.
iptables -A FORWARD -i eth0 -o eth1 -j ACCEPT

# Masquerade.
iptables -t nat -A POSTROUTING -o eth1 -j MASQUERADE

# Don't forward from the outside to the inside.
#iptables -A FORWARD -i eth1 -o eth1 -j REJECT

# Enable routing.
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Figure 11 – EMU: Gateway functionality

6.3. Content Server

Figure 12 presents the functional architecture of the CS and the interconnection to other test facilities. There are four main modules (Web Server, Session Module, Update Module and Persistence Module) and a web interface that provides access to them.
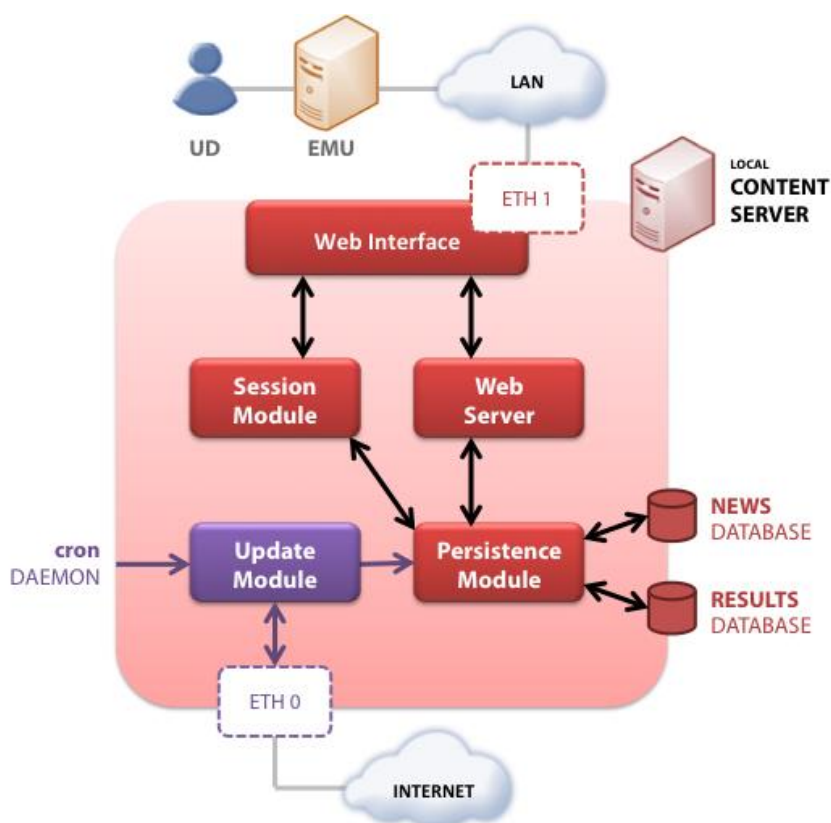
Figure 12 – CS: Functional Architecture

Web Interface (WI)

This is a basic module of the CS. When the user (through the EMU) requests any of the contents served from the CS, the interface evaluates the request and redirects it to the correspondent module. All request related to user registration, test condition evaluation and results processing are redirected to the Session Module (SM). The rest of the requests (news and photo contents) are treated in the Web Server.

Web Server (WS)
(Contribution 0124 – Short version - Chapter 3. Test Content)

The Web Server is the core of the CS. It mainly consists of two things: The web contents used in the tests for the users to browse, and the server itself, that makes those contents available for browsing.

For this particular implementation, Apache2 Server with PHP5 support was used. Since Apache2+PHP5 configuration is pretty much standard, no particular comments will be made. However, there are some considerations about the test contents themselves.

The test results presented in this contribution refer to two different web page classes: News webpage and photo album, both locally served from the Content Server.

The main structure of both webpages is based upon the works of [FTW]. However, some modifications have been made to reflect a more local content for the group of testers. These modifications will be later explained in the Update Module section.

News Webpage

- Locally hosted (Served by Apache2) [FTW].
- HTML PHP5 and JavaScript are the main languages used.
- All news contain one picture (JPEG).
- News are fetched from the Spanish and Basque RSS feeds of www.europapress.es
- Appearance has also been mimetized to resemble www.europapress.es
- A content update script has been scheduled to be executed three times a day
- Average page size depends on the length of the news selected:
    - Average news length (picture included):   379 bytes
    - Standard deviation for news content: 73 bytes
    - News page structure (constant size):   3,5 KB
    - Total Average Length   ~3,9 KB

Photo Album

- Locally hosted (Served by Apache2)
- HTML, PHP5 and JavaScript are the main languages used.
- Appearance has been mimetized to resemble www.facebook.com
- There are five photo albums with an average of 40 photos per album.
    - Average photo size: 114,50 KB
    - Standard deviation of photo sizes: 10,15 KB

Session Module (SM)

This module is responsible of the registering of the test participants in the system and the tracking of their answers throughout the whole duration of the test. When prompted by the EMU, this module returns to the user the correspondent evaluation form at the end of each condition time.

NOTE: For this module to work, a different web server technology was implemented: Java Server Faces (JSF) via Glassfish Server. There is no particular reason for choosing Glassfish over Apache2 for this module. We started developing the SM in JSF before adopting the structure of FTW for the web contents and, instead of redoing the module or migrating code from one language to another, we decided to keep both developments separately. Since both modules require no communication amongst them, there are no compatibility issues to consider.

Persistence Module (PM)

Both the Session Module and the Web Server Module require access to information stored in databases. The Persistence Module processes all requests coming from both modules.

The particular implementation of this module is a standard MySQL server with two different databases: *kepler_news* and *kepler_quest*. Each of them is respectively used by the WS and the SM modules and they require no particular configurations.

kepler_news – It consists of one table where all the articles to be displayed in the news page are stored. This table is updated several times a day by the Update Module in order to provide recent contents to the users taking part in the tests.

kepler_quest – This database contains all the user registrations and their evaluation to each of the conditions of the tests as well as some auxiliary tables. There is more detailed info about this database and its relational model in APPENDIX II.

Update Module (UM)

This is a minor module of the CS. However, its task and complexity is highly important. Its main functionality is to launch the script that inserts new contents to the *kepler_news* database periodically, as defined in */etc/crontab* (figure 13).

The network traffic generated by this module is the responsible for having a second network interface in the CS directly attached to the Internet in order to avoid interfering with any test that might be taking place during the contents update.

```
--------------------------
/etc/crontab
--------------------------

00 6    * * *   root    lynx -dump http://localhost/kepler/rssfeeder/record_articles.php
00 11   * * *   root    lynx -dump http://localhost/kepler/rssfeeder/record_articles.php
00 16   * * *   root    lynx -dump http://localhost/kepler/rssfeeder/record_articles.php
```

Figure 13 – CS: Cron Daemon

That update is done with a PHP script of the FTW structure (*/rssfeeder/record_articles.php*), which involves quite complexity and lots of preprocessing work:

The script reads an RSS pipe and extracts information from each article in the pipe to generate an entry in a 'news database' (*kepler_news*) designed for this purpose. That pipe has first to be generated according to some conditions for the easier extraction of its articles and their persistence in the database.

This means that, before thinking of using the UM, we need to have a proper functioning RSS pipe. The full process from generating the pipe to using the PHP script is explained in APPENDIX III. However, a short version of the procedure is explained below:

- First, we need one or more RSS feeds from an existing news page. In our case, we used RSS feeds from www.europapress.es on various categories (International news, Culture, Sports…).

- After this, we need to process those feeds field by field to accomplish the requirements of the FTW news pages. This is, we need to parse the articles for an easy extraction of their title, body, images, category, date and other attributes.

- Once we generate the pipe, we copy its link into the PHP script. If the pipe produces the articles with the same structure to the one defined by FTW, there's no further work to be done. However, since not all RSS feeds offer their results with the same format, it is possible that we need to modify the script to contemplate those changes (as in our case).

- When the script is correctly capable of extracting the articles from the pipe for their storage in the database, we can execute it from the browser, by accessing its URL…

- Or we can execute it periodically as stated before (figure 13).

APPENDIX I

EMU: SAMPLE SOURCE CODE

INTERNAL DOCUMENT ON PLATFORM DESIGN

1. Sample code of SEQ

```
<script>

    var width = 1200;
    var height= 850;

    (...)

    var ht  =    5000;     //   5s  Homepage Time
    var tt  = 150000;      // 150s  Condition Time
    var qt  =   30000;     //  30s  Question Time


     // OPEN SECOND BROWSER WINDOW

    function openWindow(vi, vt, vq) {

        var ref = getTimeoutTime(vq);
        window.open('http://<?php echo $dataServer.$portPHP."/".$webPHP ?>/nav/launcher.php?
            iid='+vi+'&tid='+vt+'&qid='+vq+'&ref='+ref,'fly','directories=no,resizable=yes,
            location=no,menubar=no,scrollbars=no,status=no,toolbar=no,
            fullscreen=no,dependent=no,width='+width+',height='+height+',left=50,top=50' );

    }


    // ASK OPERATION MODULE TO APPLY CONDITIONS

    function setConditions(vi, vt, vq) {

        openWindow(vi,vt,vq);

        var act="del";
        if(vq%2==1) {
            act="add";
            $('#framed').attr('src','nav/cmd.php?tid='+vt+'&act='+act);
        }
        else $('#framed').attr('src','nav/cmd.php');
    }


    // TIMEOUT times between pages are shown in absolute values.
    // FOR EXAMPLE:
    // - First page is shown on click (test 1):          1º TIMEOUT 0
    // - Second must be seen 150sec after (question 1):  2º TIMEOUT 150000 (milliseconds)
    // - Third must be seen 30sec after (test 2):        3º TIMEOUT 180000 (milliseconds)
    // - Now comes question 2... and so on.


    function getTimeoutTime(num) {
        var rt = 0;

        num--;
        if (num!=0) {

            if ((num % 2)==0) {   // EVEN - PAR
                rt = ht + (num/2)*tt + (num/2)*qt;
            }
            else if ((num % 2)==1) {  // ODD - IMPAR
                num--;
                rt = ht + (num/2)*tt + (num/2)*qt + tt;
            }
        }
        else rt=ht;

        return rt;
    }

    (...)

```

```javascript
    // LAUNCH ROUND. number = 1 --> ROUND 1 / number = 2 --> ROUND 2

    function halfRandomClicked(number){

            var round = [];

            for (var i=0; i<7; i++) {
                round[i]=0;
            }

            var count = 0;
            var tid   = 0;
            var iid   = number;
            var qid   = 1;

            setTimeout(function(){
                setConditions(iid,0,0);
                round[0]=1;
            },0);

            console.log('Start Phase '+iid);

                //FIRST ITERATION (I NEED THIS FOR THE SET-INTERVAL TO WORK 'ON TIME')

                while (exit==false) {
                    tid = Math.floor(Math.random() * 5);

                    if (round[tid]==0) {
                        round[tid]=1;
                        exit=true;
                    }
                }

                setTimeout(function(){setConditions(iid,tid,  qid++);
                setTimeout(function(){setConditions(iid,tid,  qid++); }, tt+ht);
                setTimeout(function(){setConditions(iid,tid+6,qid++); }, tt+qt);
                setTimeout(function(){setConditions(iid,tid+6,qid++); }, tt+qt+tt);

                // END OF FIRST ITERATION (NEWS + PHOTOS)


                // SINCE SET-INTERVAL REPEATS EACH TT+QT, REST ARE DONE HERE.
                // HOWEVER, FIRST SET-INTERVAL FAILS TO START AT 0, BUT AT 2(TT+QT)
                // THAT'S WHY WE NEED ONE ITERATION OUTSIDE INTERVAL.

            var si = setInterval(function(){

                for (var i=0; i<7; i++) {
                    if (round[i]==0)
                        exit = false;
                }

                while (exit==false) {
                    tid = Math.floor(Math.random() * 6);

                    if (round[tid]==0) {
                        round[tid]=1;
                        exit=true;
                    }
                }

                setTimeout(function(){setConditions(iid,tid,  qid++); }, 0);
                setTimeout(function(){setConditions(iid,tid,  qid++); }, tt);
                setTimeout(function(){setConditions(iid,tid+6,qid++); }, tt+qt);
                setTimeout(function(){setConditions(iid,tid+6,qid++); }, tt+qt+tt);

                count = 0;
                for (var i=0; i<7; i++) {
                    if (round[i]==1) {
                        count++;
                    }
                }

                count--;
                console.log('Iteration '+count+' completed');

                if (count==6) {
                        clearInterval(si);
                        console.log('End Phase '+iid);
                }

            },2*(tt+qt));

    }
</script>
```

2. Operation Module – Application Level

```php
<?php
    $ref = array("32","64","128","256","512","1024","2048","64","128","256","512","1024","2048");

    try {
        $tid = $_GET["tid"];
        $act = $_GET["act"];
    }
    catch(Exception $e) {
        $tid = 0;
        $act = "del";
    }
    $delay = 20;
    $sp_dw = $ref[$tid];

    if ($tid==0){
        $delay= 0;
        $sp_dw="clear";
        $sp_up="";
    }
    else if ($tid==1 || $tid==7)
        $sp_up = 32;
    else $sp_up = $ref[$tid-1];

    $cmd_string = "sh /scripts/www/tc_netem.sh ".$delay." ".$sp_dw." ".$sp_up;
    exec($cmd_string, $output);
?>
```

3. Operation Module – Network Interface Script

```bash
#!/bin/bash
# eth0:   TO USER DEVICE
# eth1:   TO CONTENT SERVER

sudo tc qdisc del dev eth1 root
sudo wondershaper clear eth1

if [ ! -z "$1" ]; then
#sudo tc qdisc add dev eth1 root tbf rate $1 burst 100k latency 20ms
sudo tc qdisc add dev eth1 root netem delay $1ms
sudo wondershaper eth1 $2 $3
fi

if [ -z "$3" ]; then
sudo tc qdisc del dev eth1 root
sudo wondershaper clear eth1
fi
```

4. Gateway Functionality – IPTABLES

```sh
#!/bin/sh
PATH=/usr/sbin:/sbin:/bin:/usr/bin

# Delete all existing rules.
iptables -F
iptables -t nat -F
iptables -t mangle -F
iptables -X

# Always accept loopback traffic
iptables -A INPUT -i lo -j ACCEPT

# Allow established connections, and those not coming from the outside
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -m state --state NEW -i !eth1 -j ACCEPT
iptables -A FORWARD -i eth1 -o eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT

# Allow outgoing connections from the LAN side.
iptables -A FORWARD -i eth0 -o eth1 -j ACCEPT

# Masquerade.
iptables -t nat -A POSTROUTING -o eth1 -j MASQUERADE

# Don't forward from the outside to the inside.
#iptables -A FORWARD -i eth1 -o eth1 -j REJECT

# Enable routing.
echo 1 > /proc/sys/net/ipv4/ip_forward
```

APPENDIX II

CONTENT SERVER: KEPLER_QUEST DATABASE

INTERNAL DOCUMENT ON PLATFORM DESIGN

1. kepler_quest – tables

```
table equipment
+----------+-----------+------+-----+---------+----------------+
| Field    | Type      | Null | Key | Default | Extra          |
+----------+-----------+------+-----+---------+----------------+
| pk_id    | bigint(20)| NO   | PRI | NULL    | auto_increment |
| name     | text      | NO   |     | NULL    |                |
| os       | text      | NO   |     | NULL    |                |
| browser  | text      | NO   |     | NULL    |                |
| net_iface| text      | YES  |     | NULL    |                |
| memory   | text      | YES  |     | NULL    |                |
| processor| text      | YES  |     | NULL    |                |
| type     | text      | YES  |     | NULL    |                |
+----------+-----------+------+-----+---------+----------------+

table speeds
+----------+-----------+------+-----+---------+----------------+
| Field    | Type      | Null | Key | Default | Extra          |
+----------+-----------+------+-----+---------+----------------+
| pk_id    | bigint(20)| NO   | PRI | NULL    | auto_increment |
| value    | text      | YES  |     | NULL    |                |
| location | text      | YES  |     | NULL    |                |
+----------+-----------+------+-----+---------+----------------+

table test
+-------+-----------+------+-----+---------+----------------+
| Field | Type      | Null | Key | Default | Extra          |
+-------+-----------+------+-----+---------+----------------+
| pk_id | bigint(20)| NO   | PRI | NULL    | auto_increment |
| speed | text      | YES  |     | NULL    |                |
| rtt   | text      | YES  |     | NULL    |                |
| page  | text      | YES  |     | NULL    |                |
+-------+-----------+------+-----+---------+----------------+

table user
+----------+-----------+------+-----+---------+----------------+
| Field    | Type      | Null | Key | Default | Extra          |
+----------+-----------+------+-----+---------+----------------+
| pk_id    | bigint(20)| NO   | PRI | NULL    | auto_increment |
| sex      | int(1)    | NO   |     | NULL    |                |
| expertise| bigint(20)| NO   |     | NULL    |                |
| used     | bigint(20)| NO   |     | NULL    |                |
| age      | bigint(20)| NO   |     | NULL    |                |
| login    | text      | YES  |     | NULL    |                |
| password | text      | YES  |     | NULL    |                |
| since    | text      | YES  |     | NULL    |                |
| fk_home  | bigint(20)| YES  | MUL | NULL    |                | -- Refs: speed contract at home
+----------+-----------+------+-----+---------+----------------+

table results
+-----------+-----------+------+-----+---------+----------------+
| Field     | Type      | Null | Key | Default | Extra          |
+-----------+-----------+------+-----+---------+----------------+
| pk_id     | bigint(20)| NO   | PRI | NULL    | auto_increment |
| fk_user   | bigint(20)| NO   | MUL | NULL    |                |
| fk_test   | bigint(20)| NO   | MUL | NULL    |                |
| round     | bigint(20)| NO   |     | NULL    |                | -- 0: WARMUP - 1,2: ROUNDS
| timestamp | datetime  | NO   |     | NULL    |                |
| mos       | bigint(20)| NO   |     | NULL    |                | -- MOS VALUE: 1 to 5
| rec       | int(1)    | NO   |     | NULL    |                | -- would u accept at home?
| fk_device | bigint(20)| NO   | MUL | NULL    |                |
| fk_server | bigint(20)| NO   | MUL | NULL    |                |
+-----------+-----------+------+-----+---------+----------------+
```

2. kepler_quest – sample content

```
table equipment
+-------+--------------------+-----------+-------+----------+-----+------------------+--------+
| pk_id | name               | os        | brow  | net_iface| mem | processor        | type   |
+-------+--------------------+-----------+-------+----------+-----+------------------+--------+
|     0 | Dell Optiplex GX620| Linux 3.2.0| FF 24 | Ethernet | 2GB | Intel Pentium4 HT| server |
|     1 | Dell Optiplex 790  | Windows 7 | FF 24 | Ethernet | 4GB | Intel Core i3    | client |
|     2 | Dell Optiplex 745  | Windows 7 | FF 24 | Ethernet | 2GB | Intel Core 2 Duo | client |
|     3 | Dell Optiplex 745  | Windows 7 | FF 24 | Ethernet | 2GB | Intel Core 2 Duo | client |
|     4 | Dell Optiplex 790  | Windows 7 | FF 24 | Ethernet | 4GB | Intel Core i3    | expert |
+-------+--------------------+-----------+-------+----------+-----+------------------+--------+

table speeds
+-------+-----------+----------+
| pk_id | value     | location |
+-------+-----------+----------+
|     0 | NS/NC     | home     |
|     1 | < 1 Mbps  | home     |
|     2 | 1 Mbps    | home     |
|     3 | 2 Mbps    | home     |
|     4 | 3 Mbps    | home     |
|     5 | 4 Mbps    | home     |
|     6 | 6 Mbps    | home     |
|     7 | 8 Mbps    | home     |
|     8 | 9 Mbps    | home     |
|     9 | 10 Mbps   | home     |
|    10 | 12 Mbps   | home     |
|    11 | 15 Mbps   | home     |
|    12 | 20 Mbps   | home     |
|    13 | > 20 Mbps | home     |
+-------+-----------+----------+

table test
+-------+-------+------+--------+
| pk_id | speed | rtt  | page   |
+-------+-------+------+--------+
|     1 | 64    | 20   | news   |
|     2 | 128   | 20   | news   |
|     3 | 256   | 20   | news   |
|     4 | 512   | 20   | news   |
|     5 | 1024  | 20   | news   |
|     6 | 2048  | 20   | news   |
|     7 | 64    | 20   | photos |
|     8 | 128   | 20   | photos |
|     9 | 256   | 20   | photos |
|    10 | 512   | 20   | photos |
|    11 | 1024  | 20   | photos |
|    12 | 2048  | 20   | photos |
+-------+-------+------+--------+

table user
+-------+-----+-----------+------+-----+-------+----------+-------+---------+
| pk_id | sex | expertise | used | age | login | password | since | fk_home |
+-------+-----+-----------+------+-----+-------+----------+-------+---------+
|     6 | 0   |         2 |    5 | 29  | NULL  | NULL     | 15    |      13 |
+-------+-----+-----------+------+-----+-------+----------+-------+---------+

table results (extract)
+-------+---------+---------+-------+---------------------+-----+-----+-----------+-----------+
| pk_id | fk_user | fk_test | round | timestamp           | mos | rec | fk_device | fk_server |
+-------+---------+---------+-------+---------------------+-----+-----+-----------+-----------+
|    42 |       6 |       1 |     0 | 2013-10-22 14:49:08 | 1   | 0   |         3 |         0 |
|    42 |       6 |      11 |     1 | 2013-10-22 14:49:08 | 5   | 1   |         3 |         0 |
|    42 |       6 |       3 |     2 | 2013-10-22 14:49:08 | 3   | 0   |         3 |         0 |
+-------+---------+---------+-------+---------------------+-----+-----+-----------+-----------+
```

3. kepler_quest – relations
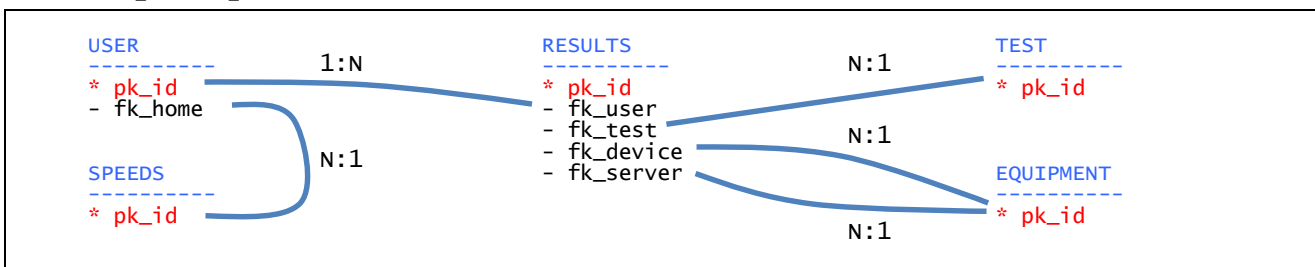
APPENDIX III

GUIDE TO FETCHING ARTICLES

INTERNAL DOCUMENT ON PLATFORM DESIGN

1. RSS feeds and JSON PIPE

All developments presented in this Annex are based upon the works of FTW. In this particular version, we show a quick view on generating our own pipe fed with various RSS of the site www.europapress.es. In particular, we used the following ones (in Spanish):

http://www.europapress.es/rss/rss.aspx?ch=69   - Internacional / International news

http://www.europapress.es/rss/rss.aspx?ch=66   - Nacional / National news

http://www.europapress.es/rss/rss.aspx?ch=67   - Deportes / Sport

http://www.europapress.es/rss/rss.aspx?ch=126   - Cultura / Culture

http://www.europapress.es/rss/rss.aspx?ch=73   - Sociedad / Society

http://www.europapress.es/rss/rss.aspx?ch=435   - Motor

http://www.europapress.es/rss/rss.aspx?ch=58   - País Vasco / Basque Country (Euskera)

The pipe we generated can be accessed here:

http://pipes.yahoo.com/pipes/pipe.info?_id=319670d989aa62ffeb7b36040802f29b

2. Generating the PIPE

Our pipe can be cloned for its usage with other RSS feeds. However, some guidelines must be followed.

- First, you must have a Yahoo account for editing and cloning the pipe, as well as publishing your own version of the pipe.

- Click on the link to the pipe. Once you are logged in, you can press the "EDIT SOURCE" button.

- A workspace will be displayed with the actual boxes of the already defined pipe [See figure A1]. You can create your own in this workspace or edit the existing one. All explanations from now on will consist on explaining the boxes of the figure A1 sequentially.

|	
1. FETCH FEED	This box is used to include any existing RSS feed from any site you want to use in the pipe. When using more than one, it is highly recommended to use compatible RSS feeds (same news site, or same structure…) because the next boxes do iterate the contents from those RSS feeds in search of common patterns for the final feed to produce.
2. CATEGORY	In our version, it is required that the category has only ONE word. If any of the chosen categories has more than one word, you must loop through all the news and REPLACE the category name with another containing one word.
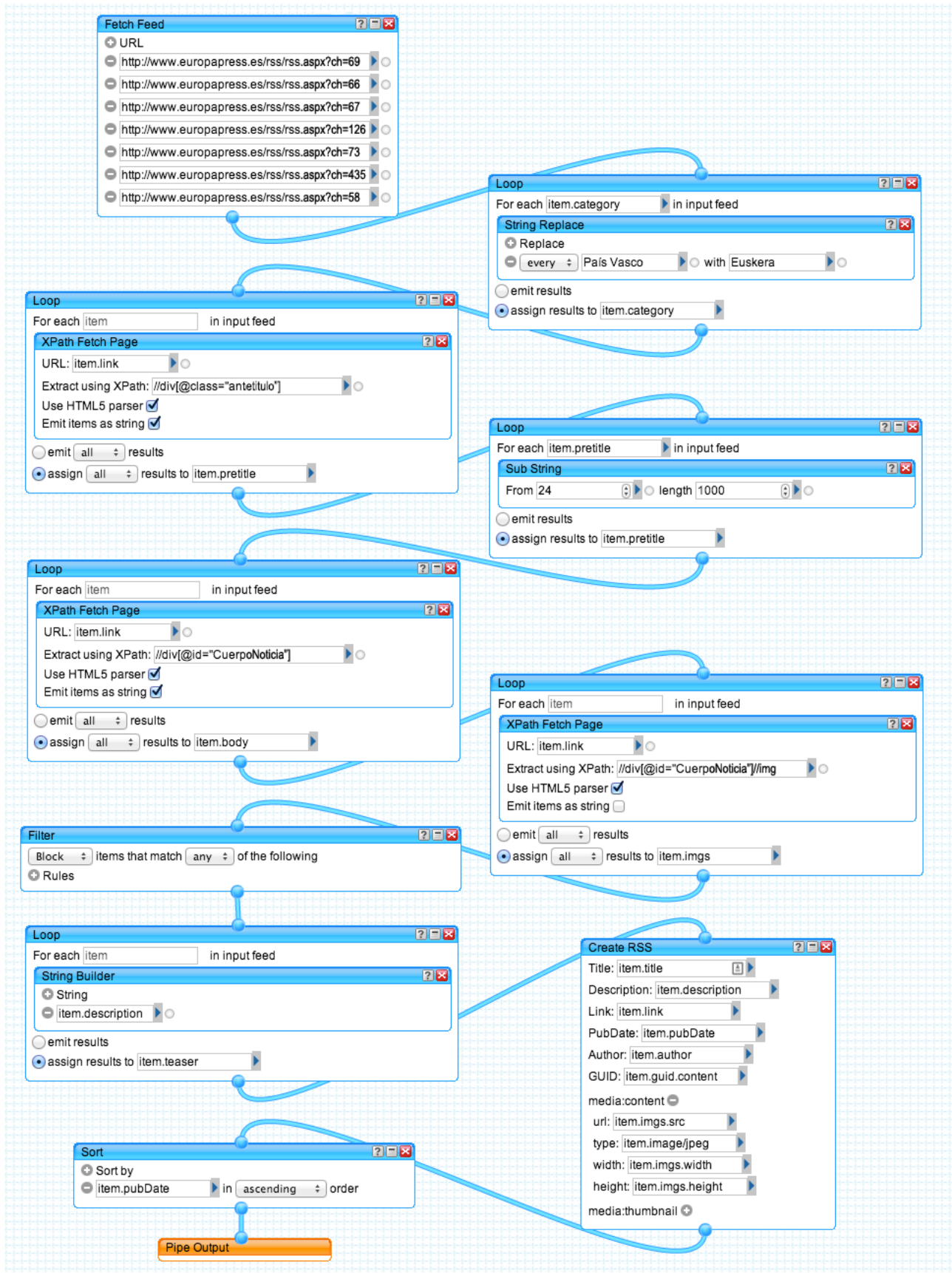
Figure A1 – Generated Pipe for Contribution 0124

3. PRE-TITLE   The content site developed by FTW displays news with a very specific structure. News consist of: Pre-title, title, teaser and body. You have to be sure that the original RSS feed has pre-title or something in the source code that might work as it.

Perhaps all the contents you need are already correctly encapsulated in a field of each RSS article. You can guess that clicking in the box. It will change from BLUE to ORANGE and below, a window frame will display the news fetching progress to that point. If you navigate any of those news you'll see fields that already show info correctly. (TITLE or AUTHOR in our case).

The rest of contents must be processed.

Therefore, it is quite important to browse the SOURCE CODE of original news page where the RSS is loaded, to see how and where each of those four blocks is displayed, the ID or CLASS of the DIV… etcetera. Once found, you can process it as in the third box of figure A1.

Our feeds locate the pre-title content in a DIV with **class="antetitulo".** Then, we loop over all the news fetched and extract that information…

FROM   **//div[@class="antetitulo"]**

TO   **item.pretitle**

4. SUBSTRING   In our case, there are some characters at the beginning of the pre-title string that we don't want to use in the final version. We now substring the pre-title to delete them, and save them back to **item.pretitle**.

5. BODY   Again, as in the pre-title box, we need to process the article's body. For this purpose, we go back to the source code of the article in the news page and see that it is displayed in a DIV with **id="CuerpoNoticia".** Therefore, we go…

FROM   **//div[@id="CuerpoNoticia"]**

TO   **item.body**

6. IMAGES   Now it's time to extract photos of the body content. We will only use news with pictures inside the body. In our case: "CuerpoNoticia".

FROM   **//div[@id="CuerpoNoticia"]//img**

TO   **item.imgs**

7. FILTERING   This box is available for any filtering rule to apply. In our case, we didn't use it, but it is available for others, just in case. It is important to notice that this box, or any of the others can be defined in other order. However, the resulting feed might vary. Particularly with the usage of THIS box.

8. TEASER   The teaser is a short version of the content that is shown in the homepage of each category. In our case, it is stored in the field DESCRIPTION so we just copy it from **item.description** to **item.teaser**

9. CREATE RSS   Once we are done processing everything, we allocate the resulting fields in the available gaps of this box. Also, we are forced to select one, and only one, image per article. This box presents how to do so, defining its height, width and type.

10. OUTPUT   We now put the final box in the pipe. This box enables the output of the processed feeds as a new RSS feed of our own. Once you save the pipe (in the upper right of the webpage), you'll be ready to use it.

3. PIPE as JSON

Once you finish editing the pipe, you need to export it as JSON. This is done by pressing the "Get as JSON" link in the webpage displayed after you save your pipe.

In our case, this link…

http://pipes.yahoo.com/pipes/pipe.info?_id=319670d989aa62ffeb7b36040802f29b

…offers us that button. After clicking it, we are redirected to:

http://pipes.yahoo.com/pipes/pipe.run?_id=319670d989aa62ffeb7b36040802f29b&_render=json

Copy the URL, you'll need it.

4. Articles Database

```
DATABASE:   kepler_news
TABLE:      articles


+-------------+-------------+------+-----+---------+----------------+
| Field       | Type        | Null | Key | Default | Extra          |
+-------------+-------------+------+-----+---------+----------------+
| id          | int(11)     | NO   | PRI | NULL    | auto_increment |
| title       | varchar(400)| NO   | MUL | NULL    |                |
| link        | varchar(300)| YES  |     | NULL    |                |
| category    | varchar(200)| YES  | MUL | NULL    |                |
| description | longtext    | NO   |     | NULL    |                |
| images      | longtext    | NO   |     | NULL    |                |
| teaser      | longtext    | NO   |     | NULL    |                |
| body        | longtext    | NO   |     | NULL    |                |
| pub_date    | datetime    | NO   | MUL | NULL    |                |
| guid        | varchar(300)| NO   |     | NULL    |                |
+-------------+-------------+------+-----+---------+----------------+
```

Figure A2 – kepler_news database

Considering the resulting fields encapsulated in BOX 9 of image A1, we can create now our articles database. Above image presents the articles table defined in our case.

5. Record Articles Script

Now it's turn for adequating the FTW record articles script to our version of the pipe. You can locate this file in **/rssfeeder/record_articles.php**

The code in the next page presents the adapted version to our pipe. It defines the connection to the MySQL database presented avobe, and the processing of each field encapsulated in BOX 9 of image A1. When loading this PHP page from a browser, you get your articles database updated.

```php
<?php
require '../libs/Nette/loader.php';
require('../libs/phpQuery/phpQuery/phpQuery.php');

use Nette\Diagnostics\Debugger,
        Nette\Utils\Finder,
        Nette\Image;

Debugger::enable(); // enable Nette Debugger

// ***********************************************************************
// DEFINITIONS
// ***********************************************************************

/**
 * Define basic PHP constants - e.g. local image storage
 */
define("IMAGES_DIR", "/var/www/kepler/www/images/article_images/");
define("IMAGES_URL", "http://localhost/kepler/www/images/article_images/");

/**
 * Url address of Yahoo Pipes RSS Feed
 * @var $jsonFeed varchar
 */
$jsonFeed =
'http://pipes.yahoo.com/pipes/pipe.run?_id=319670d989aa62ffeb7b36040802f29b&_render=json';

/**
 * Database Conenction Credentials
 */
$mysql = array(
            'dsn'       => 'mysql:unix_socket=/var/run/mysqld/mysqld.sock;dbname=kepler_news',
            'user'      => 'root',
            'password' => '*****'
);

// ***********************************************************************
// CONTAINER CLASS
// ***********************************************************************

/**
 * DI Container for database connection and articles factory
 */
class DIContainer {
        /** @var array $parameters - database connection parameters */
        private $parameters;
        /** @var string $rssFeed */
        private $rssFeed;
        /** @array Nette\Database\Connection */
        private $services = array();

        public $lastArticle;
        public $lastID;
        public $lastDate;

        public function __construct(array $parameters, $jsonFeed){

            $this->parameters     = $parameters;
            $this->rssFeed        = $jsonFeed;

            $this->lastArticle = $this->lastArticle();
            $this->lastID         = $this->lastArticleID();
            $this->lastDate  = $this->lastArticleDate();
        }

        public function createConnection() {

            return new Nette\Database\Connection(
                $this->parameters['dsn'],
                $this->parameters['user'],
                $this->parameters['password']
            );
        }
```

```php
    public function getConnection()
    {
        if(!isset($this->services['connection'])) {
            $this->services['connection'] = $this->createConnection();
        }
        return $this->services['connection'];
    }

    /**
     * Factory for Article class
     */


    public function createArticle() {
        return new Article($this->getConnection());
    }

    /**
     * Retrieve RSS Json string
     * @return php articles array from JSON string
     */
    public function getRssJson() {
        $jsonResult = json_decode(file_get_contents($this->rssFeed));
        return $jsonResult->value->items;
    }

    /**
     * Filter out articles with insufficiently wide images
     * @param array php variable with new articles
     * @return array articles with proper images
     */
    public function filterByImageSize($articles, $imgFilterWidth)
    {

        $filteredArticles = array();

        foreach ($articles as $article) {
            if (is_array($article->imgs)){
                if ($article->imgs[0]){
                    if ($article->imgs[0]->width >= $imgFilterWidth ){

                        $filteredArticles[] = $article;
                    }
                }
            } else {
                if ($article->imgs){
                    if ($article->imgs->width >= $imgFilterWidth ){

                        $filteredArticles[] = $article;
                    }
                }
            }

        }
        return $filteredArticles;
    }

    public function filterByEnclosure($articles)
    {

            $filteredArticles = array();

            foreach ($articles as $article) {
                    if (is_array($article->enclosure)){
                            if ($article->enclosure[1]){
                    $filteredArticles[] = $article;
                            }
                }
            }
            return $filteredArticles;
    }

    /**
     * Filter out articles with body failures (body=NULL)
     */
    public function filterBodyFailures($articles) {

        $filteredArticles = array();

        foreach ($articles as $article) {
            if($article->body) {
                $filteredArticles[] = $article;
            }
        }
        return $filteredArticles;
    }
```

```php
        /**
         * Filter out articles wich already are in DB
         * @param array php variable with new articles
         * @return array articles for recording in DB
         */
        public function filterByDate($articles) {

            $filteredArticles = array();
            $old_date = $this->lastDate;

            foreach ($articles as $article) {
                $new_date = new DateTime($article->pubDate);


                if ($new_date->format('Y-m-d H:i:s') > $old_date->format('Y-m-d H:i:s')){
                    // Debugger::Dump($new_date);
                    $filteredArticles[] = $article;
                }
            }
            return $filteredArticles;
        }

        /**
         * Select latest article from DB
         */
        public function lastArticle(){
            $item = $this->getConnection()->table('articles')->order('id DESC')->limit(1)->fetch();
            return $item;
        }

        public function lastArticleID() {
            if ($this->lastArticle) {
                return $this->lastArticle->id;
            }
            else return 0;
        }

        public function lastArticleDate() {
            if ($this->lastArticle) {
                return $this->lastArticle->pub_date;
            }
            else return new DateTime('2013-01-01');
        }

}


// ************************************************************************
// ARTICLE CLASS
// ************************************************************************


/**
 * Class for working with article and storing it to database
 */
class Article {
        /** @var array content */
        public $content;
        /** @var Nette\Database\Connection connection */
        public $connection;

        public function __construct(Nette\Database\Connection $connection) {
            $this->connection = $connection;
        }

        /**
         * Treat external links
         * @param html code
         * @return html code without external links
         */
        public function externalLinks($html) {
            $html = phpQuery::newDocument($html);
            pq('a')->attr("href", "#")->attr("target", "")->attr("class", "external");
            return $html->html();
        }


        /**
         * Save images to local folder
         */
```

```php
        public function saveImages($id) {

            /** Handle big title image */
            if(!is_dir(IMAGES_DIR . $id)){

                // Create new image directory for current article
                mkdir(IMAGES_DIR . $id);

                $src = is_array($this->content->imgs) ?
                    $this->content->imgs[0]->src : $this->content->imgs->src;
                $titleImage = Image::fromFile($src);

                // Title image gets new name "title.jpg"
                $newName = $id . "/title.jpg";
                $titleImage->save( IMAGES_DIR . $newName, 80, Image::JPEG);

                // Change original url to the local url
                is_array($this->content->imgs) ?
                    $this->content->imgs[0]->src = $newName : $this->content->imgs->src = $newName;

            /** Handle smaller body images */

                // Load body images
                $bodyImages = phpQuery::newDocument($this->content->body);
                $images = pq('img');

                    // Save body images to the local folder and change src urls
                $imageID = 0;
                foreach ($images as $image)
                {
                    $imageID++;
                    $imageSelector = pq($image);
                    $image = Image::fromFile($imageSelector->attr('src'));

                    // Save Image with a new name
                    $smallImageName = $id . "/" . $imageID . ".jpg";
                    $image->save( IMAGES_DIR . $smallImageName, 80, Image::JPEG);

                    // Change src attributes to local
                    $imageSelector->attr('src', $smallImageName);
                }
                $this->content->body = $bodyImages->html();
            }
        }

        /**
         * Save article to the database
         */
        public function save()
        {
            $src = is_array($this->content->imgs) ?
                $this->content->imgs[0]->src : $this->content->imgs->src;
            $cat = is_array($this->content->category) ?
                $this->content->category[0] : $this->content->category;

            return $this->connection->table('articles')->insert(array(
                'guid'          => $this->content->guid,
                'title'         => $this->content->title,
                'link'          => $this->content->link,
                'category'      => $cat,
                'description'   => $this->content->description,
                'images'        => $src,
                'teaser'        => $this->content->teaser,
                'body'          => $this->content->body,
                'pub_date'    => date_format(new DateTime($this->content->pubDate), 'Y-m-d H:i:s')
            ));
        }

}

// **********************************************************************
// LAUNCHER
// **********************************************************************

        // Create DI container
        $container = new DIContainer($mysql, $jsonFeed);
        $last_art = $container->lastArticle;
        $last_ID  = $container->lastID;

        // Retrieve the RSS feed as a PHP variable
        $articles = $container->getRssJson();
        Debugger::dump($articles);
```

```php
    // Filter out articles with body failure (body NULL)
    $articles = $container->filterBodyFailures($articles);

    // Filter out articles with images <= 100px width
    $articles = $container->filterByImageSize($articles, 100);

    // Filter out articles already in DB
    $articles = $container->filterByDate($articles); //, $last_art->pub_date);

    // Set the proper counter for new articles in DB
    // Debugger::dump($articles);
    // Debugger::dump($last_ID);

    if ( sizeof($articles) == 0 ) {
        echo "Nothing to record";
    }


    foreach ($articles as $newArticle) {

        $last_ID++;

        $art            = $container->createArticle();
        $art->content   = $newArticle;

        $art->saveImages($last_ID);
        $art->content->body = $art->externalLinks($art->content->body);
        $art->content->guid = isset($art->content->pretitle) ? $art->content->pretitle : "";
        //echo $art->content->guid;
        $art->save();

        echo " - Article id#" . $last_ID . " Sucessfully recorded! <br />";

    }
```

Figure A3 – /rssfeeder/record_artices.php

6. Auto Update

If you want to execute this PHP script periodically, you can configure your /etc/crontab to include a similar entry to this example:

```
-------------------------
/etc/crontab
-------------------------

00 6    * * *   root    lynx -dump http://localhost/kepler/rssfeeder/record_articles.php
00 11   * * *   root    lynx -dump http://localhost/kepler/rssfeeder/record_articles.php
00 16   * * *   root    lynx -dump http://localhost/kepler/rssfeeder/record_articles.php
```

Figure A4 – Cron Daemon

ANEXO IV
Instrucciones para la realización del experimento

Este anexo recoge el documento entregado a los participantes previamente a la realización del experimento. En él se describe su duración, las fases con las que cuenta y las tareas a seguir por el participante.

Pruebas de Calidad en Navegación Web

Instrucciones

Las pruebas que está a punto de realizar tienen una duración aproximada de 100 minutos, distribuidos de la siguiente manera:

· Lectura de Instrucciones	5 min
· Registro de Usuario	4 min
· Fase de Entrenamiento	9 min
· Fase de Test 1	36 min
· Descanso	10 min
· Fase de Test 2	36 min
	100 min

Para realizar estas pruebas se hará uso del navegador **FIREFOX 24**, situado en el escritorio del equipo en el que hará las pruebas.

Al abrirlo, verá que aparece ya cargada una página de menú en el que se irán habilitando los botones según vaya completando las fases del test.

BAJO NINGÚN CONCEPTO CIERRE LA VENTANA DE MENÚ DURANTE LA DURACIÓN TOTAL DE LA PRUEBA O HABRÁ QUE VOLVER A EMPEZAR DESDE EL PRINCIPIO.

REGISTRO

Haga clic en el botón **REGISTRO**. Se abrirá una ventana nueva, en la que se le solicitará rellenar una serie de datos. No hay tiempo límite para esta sección así que asegúrese de que todos los datos son correctos.

Cuando haya terminado, le aparecerá un botón de **CERRAR**. Al clicar, volverá a ver la página de menú que había quedado debajo de ésta.

FASE DE ENTRENAMIENTO

Haga clic en el botón **ENTRENAMIENTO**. Se abrirá una ventana nueva, en la que se le mostrarán, en tres rondas, un ejemplo de navegación web a una velocidad determinada, en tres casos: El de la velocidad máxima, la mínima y una intermedia. Aparecerán de manera aleatoria.

Puede navegar libremente por la web (NOTICIAS Y GALERIA DE FOTOS) durante los 150 segundos de cada ronda. Al finalizar cada ronda, se visualizará automáticamente un formulario en el que podrá valorar su percepción con dos sencillas preguntas.

TIENE 30 SEGUNDOS MÁXIMO (tanto aquí como en las fases de test) para responder a las preguntas. Por favor, no se demore en contestar. Es fundamental que haya respondido a todo para dar por válido el test.

Al responder **en la fase de ENTRENAMIENTO**, se le mostrará la valoración que debería haber hecho de la prueba, comparada con la suya. Utilice estas referencias para las fases de test.

Una vez finalizado el entrenamiento, se mostrará el botón de **CERRAR**. Al clicar, volverá al menú anterior, al igual que en la fase de registro.

> **NOTA:** Las webs de NOTICIAS y GALERÍA tienen usuario y password. Sólo hay que introducirlo una vez en toda la prueba.

> **USUARIO:**   **nqas**
> **PASSWORD:**   **nqas**

FASES DE TEST

Haga clic en el botón **FASE 1**. Se abrirá una ventana nueva, en la que se le mostrarán, en sucesivas rondas dos tipos de página web (de noticias y de fotos) en las que podrá navegar libremente con algunas excepciones que se explican a continuación:

En la ronda de entrenamiento podía pasar de NOTICIAS a GALERÍA y viceversa sin limitación. Ahora, en las fases de test, se pide que:

Si está en la web de NOTICIAS:

> **NO clique en SALIR ni cierre la ventana bajo ningún concepto**
> **NO acceda a la sección de GALERÍA**
> **NO visualice ningún VÍDEO**

Si está en la web de FOTOGRAFÍAS (GALERÍA):

> **NO clique en SALIR ni cierre la ventana bajo ningún concepto**
> **NO acceda a la sección de NOTICIAS**

Cada ronda durará 150 segundos, como en el entrenamiento, y vendrá seguida, de nuevo por un par de preguntas acerca de su percepción. De nuevo:

RESPONDA OBLIGATORIAMENTE LAS PREGUNTAS EN MENOS DE 30 SEGUNDOS

Espere después a que automáticamente se inicie la siguiente ronda.

Al finalizar todas las rondas de la fase y tras contestar la última pregunta, se le mostrará el botón de **CERRAR**, y volverá al menú anterior.

Si estaba en la **FASE 1**, ahora podrá descansar 10 minutos antes de clicar en **FASE 2**. En esta otra fase, se mostrarán las mismas páginas **en otro orden aleatorio de velocidades**.

Al terminar la **FASE 2**, por favor, clique en el botón de **RESET** para finalizar el test.

SI TIENE ALGUNA DUDA, POR FAVOR, COMÉNTELA ANTES DE COMENZAR
MUCHAS GRACIAS POR SU COLABORACIÓN