Web Advice Module

COSMIC Case Study

www.nesma.org

Table of Content

Revision history

VERSION	DATE	DESCRIPTION	AUTHORS
1.0	January 2014	Final version	Frank Vogelezang Charles Symons Arlan Lesterhuis
1.1	February 2016	Updated to version 4.0.1 of the COSMIC method and updated versions of used Guidelines	Frank Vogelezang Charles Symons Arlan Lesterhuis

Reviewers of version 1.0

NAME	ORGANIZATION	COUNTRY
Alain Abran	École de technologie supérieure	Canada
Chris Woodward	DCG-SMS	United Kingdom
Cigdem Gencel	Deiser	Spain
Edwin van Gorp	Sogeti Nederland	the Netherlands
Eric van der Vliet*	CGI	the Netherlands
Fred de Wilde*	ING Domestic Banking	the Netherlands
Harold van Heeringen*	Metri group	the Netherlands
Jean-Marc Desharnais	École de technologie supérieure	Canada
Jolijn Onvlee*	Onvlee Opleidingen & Advies	the Netherlands
Luca Santillo	Agile Metrics	Italy
Peter Bellen*	Independent consultant	the Netherlands
Peter Fagg	Pentad-SE Ltd.	United Kingdom
Theo Prins	Sogeti Nederland	the Netherlands

member of the COSMIC working group of Nesma

Reviewers of version 1.1

NAME	ORGANIZATION	COUNTRY
Arlan Lesterhuis	COSMIC	the Netherlands
Charles Symons	COSMIC	United Kingdom

1   Foreword

This case study is an example of how the COSMIC method should be applied to measure a small web application. The case study has been assembled by members of the COSMIC working group of Nesma, based on real specifications, for the purpose of sharing our experience with the COSMIC community.

This application, although very small, contains a lot of discussion points about how the COSMIC method should be applied correctly. It is a real application, which could be encountered by any measurement professional. Because of its compact, yet complicated nature we decided to make a case study out of this application. We hope that it helps the COSMIC community to understand a number of aspects of the method better and assists in ensuring consistent interpretation of the COSMIC principles.

1.1   Authors

This document has been initiated by the COSMIC Working Group of Nesma, who served as the main reviewers, together with other volunteers from the COSMIC community.

1.2   Conformity to the COSMIC measurement principles

The conformity to the current version of the Measurement Manual, version 4.0.1 (MM), the Business Application Guideline, version 1.2 (BAG), the Guideline for Sizing Service Oriented Architecture Software, version 1.1 (GSOA) and the Guideline for Sizing Real-Time Software, version 1.1 (RTAG) has been verified by experienced COSMIC practitioners. Wherever one of these documents is referenced, the reference is made to the section title rather than to the section number.

1.3   Observations on the clarity of the documented requirements

Although based on real requirements, the original requirements have been edited for the purposes of this case study to make the requirements as unambiguous as possible to be able to focus on the measurement principles. From the length of the discussion section in chapter 4 one might get the impression that the COSMIC method is difficult to apply. However, the reader should bear in mind that a large portion of the discussion in chapter 4 deals with the interpretation of the requirements, rather than with the application of the COSMIC method. This is a deliberate choice of the authors, since in practice the interpretation of the requirements is an important part of the measurement process. The COSMIC method has included the Mapping Phase to extract the Functional User Requirements (FUR) in a form to which the COSMIC Generic Software Model can be applied (MM Applying the Generic Software Model).

1.4   Disclaimer

By publication of this user guide the Nesma wants to contribute to further understanding of applying the COSMIC method. Nesma is not responsible for the use of this publication, nor for the results obtained by using the described approach.

1.5 Contact information

Any suggestions, remarks or questions regarding this case study can be sent to cosmic@nesma.org.

1.6 More information on the COSMIC method

More information on the COSMIC method, including the latest versions of the mentioned manuals, and other case studies can be obtained from the COSMIC website at www.cosmic-sizing.org.

2 Requirements for the Web Advice Module

2.1 Context

The Web Advice Module is a special module on the website of a commercial bank to assist (young) customers with the choice whether they are going to rent a house or buy one with a mortgage. The customer fills in his or her opinion to a number of propositions and is presented rough advice from the Web Advice Module. Based on this advice the customer can request a rent or buy advice session via the Module if he or she wishes, or contact the local branch for advice.

2.2 Start page

On the start page the customer is presented with a short introduction text and five propositions with which he or she can agree, disagree or choose neutral. By default the neutral choice is activated. When the customer is satisfied with the choices, he or she can push the Give advice button to see the advice, based on the set of choices.

The other buttons are all active and start functionality that is described in later sections.

2.3 Advice page

Based on the business rules, the Start page is re-displayed with a general advice on the best option and the relevant advice text for each choice that has an answer in the category of the general advice. The advice texts shown below are example texts that need to be finalized on implementation. The lower part of the page shows the propositions again with the current choices. These can be changed to generate a new advice.

2.4 Business rules

Each agree or disagree corresponds to a score for renting (R) or buying (B) a house and for each choice an advice text is available to accompany the general advice:

1. agree (R, rent text 1)   disagree (B, buy text 1)
2. agree (B, buy text 2)   disagree (R, rent text 2)
3. agree (B, buy text 3)   disagree (R, rent text 3)
4. agree (B, buy text 4)   disagree (R, rent text 4)
5. agree (B, buy text 5)   disagree (R, rent text 5)

If the customer chooses neutral for a certain proposition, no advice text is shown in the advice for that proposition.

If the number of R-scores is greater than the number of B-scores, the customer is shown the general advice to rent a house together with the advice texts for the propositions that resulted in an R-score.

If the number of B-scores is greater than the number of R-scores, the customer is shown the general advice to buy a house together with the advice texts for the propositions that resulted in a B-score.

If the numbers are equal, then a mixed general advice is shown together with the advice texts of the questions that resulted in an R-score and the advice texts for the propositions that resulted in a B-score.

Three choices can change this logic:
- If the choice at proposition 2 is **disagree**, then the general advice is always to rent.
- If the choice at proposition 5 is **disagree**, then the general advice is always to rent.
- If the choice at proposition 1 is **agree**, then the general advice is to rent if the number of R-scores is equal to or higher than the number of B-scores. If the number of B-scores is higher than the number of R-scores, then the mixed general advice is given.

2.5 Request form for an Advice Session

The customer may require advice on renting or on a mortgage to buy a house. To help the customer, he or she may send a request for an advice session from the Web Advice Module. The customer enters personal and contact data in a request form.

When the customer presses the [Advice on mortgage] or [Advice on renting] button the entered data, together with the answers from the start page and the general advice and advice texts from the advice page, are sent to a back office service of the bank to arrange the session. See also section 4.1.2.

2.6 Error messages

When the Web Advice Module detects erroneous processing, this must be communicated to the user as an error message in a separate window. Erroneous processing can be the result of either some (technical) error from a software component or an error of a human user. If a human user causes an error the following error message must be shown:
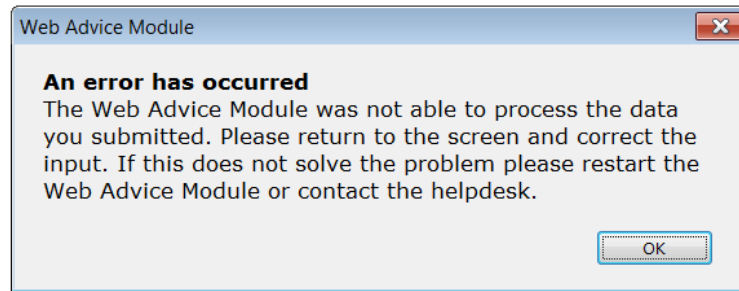
If a software component causes a (technical) error, the following error message must be shown:
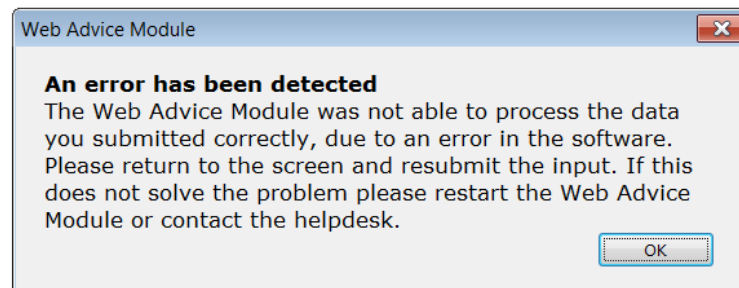
To this message additional (technical) information may be added to assist users of the Web Advice Module to solve the cause of the erroneous processing.

2.7 Inactivity message

If the customer has not used any functionality of the pages of the Web Advice Module for more than five minutes, a pop-up message will appear with a message that the customer has been inactive for too long and that the entered data and answers are not available anymore:
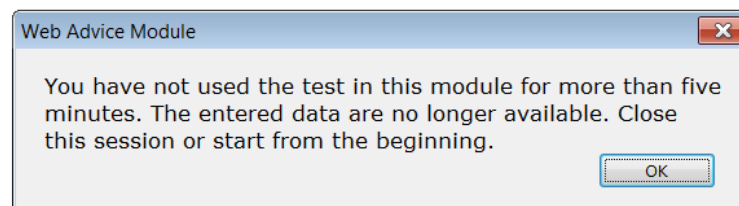
2.8 Mortgage assessment

If the customer presses the button the customer is directed away from the Web Advice Module to an existing mortgage assessment module. This existing module gives an indication of the maximum mortgage the bank is willing to provide, based on some financial data the customer has to provide in the mortgage assessment module. No data is sent to the mortgage assessment module from the Web Advice Module.

2.9   Requirements for all pages

Below each page are links to the Web Advice Module's principles, the bank's privacy statement and the bank's disclaimer for web pages. Both the privacy statement and the disclaimer are existing services that can be invoked from the Web Advice Module and are shown directly in the browser of the customer, without further interaction with the Web Advice Module. These existing services are available to all web applications from the bank. All entered data are subjected to basic validations for formatting and valid range. The customer is notified when invalid data is entered by an 'error has occurred' message. All entered data remain available within the Web Advice Module during the web session. When a customer requests an advice session these entered data are sent to a back office service. When the web session is closed by clicking on a 'close window' button, all entered data is no longer available.

The title of this module is "Web Advice Module" which appears in the title bar of the internet browser. In the top of the page the bank's logo is shown, together with the title "Web Advice Module". Below the title in a blue bar the following proposition is shown: "What suits me most: to buy or to rent a house?" The figure below this section is an example of what the general page lay-out should look like:
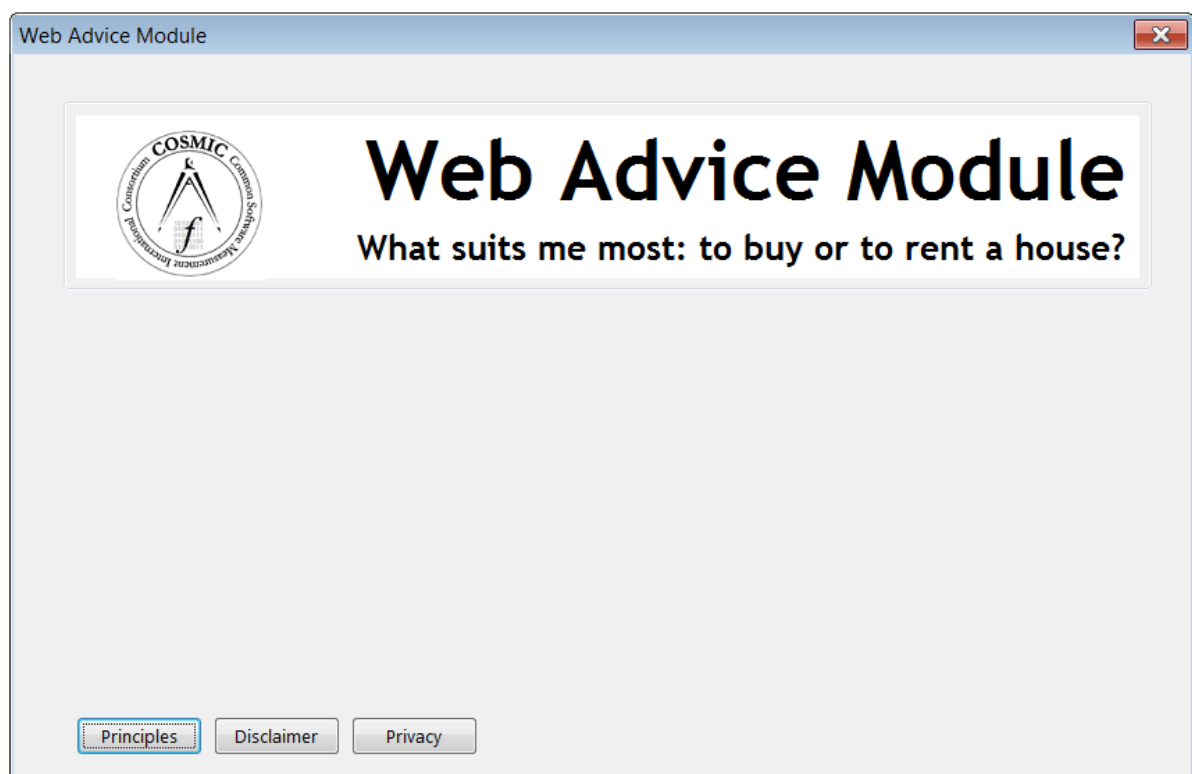
At the bottom of each page there is a "Principles" button. When this button is pressed, the Web Advice Module opens a new page with the following information:
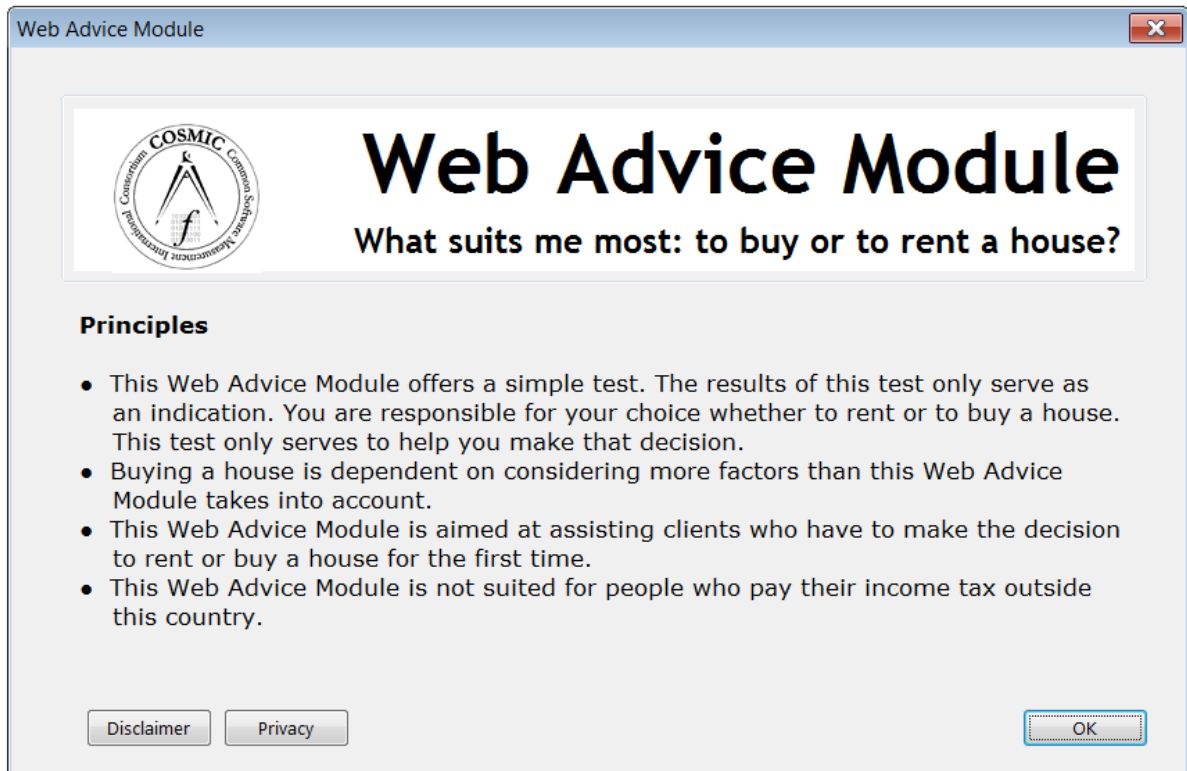
2.10 Maintenance of the Web Advice Module

The following items must be editable without the assistance of a programmer:
- The introduction text on the Start page (see section 2.2)
- The proposition texts (see section 2.2)
- The general advice texts (see section 2.3)
- The advice texts (see section 2.3)

3   Measurement strategy phase

In this chapter sections in *italics* are explanatory text from the authors that are not a part of the actual measurement process.

3.1   Determine the PURPOSE

The purpose of this measurement is to measure a functional size of the Web Advice Module that can be used as a basis to estimate the required effort to build the software.

3.2   Determine the SCOPE

The scope of the measurement is all of the FUR that are related to the Web Advice Module, i.e. which are derived from the requirements in chapter 2 of this document.

This implies that the functionality of existing services is out of scope for this measurement. However, the functionality within the Web Advice Module to start the existing services is within the measurement scope.

3.3   Determine the level of decomposition

The level of decomposition of this scope is that of a whole application. All the functionality described in the FUR that is in scope for this measurement resides in the application layer.

3.3.1   One or more layers

The privacy statement and the disclaimer are utility services that provide common functionality (business or non-business) independently of, but available to, other services (GSOA Utility Services). The Web Advice Module and all other services that it invokes are all in the one application layer.

3.4   Determine the LEVEL OF GRANULARITY

The level of granularity of the requirements is at the standard level of granularity, the functional process level.

3.4.1   Determining the level of granularity

The requirements are at the standard level of granularity, meaning that the functional users are individual humans (Customer, Application manager) or individual pieces of software (Back Office service, System Clock (as part of the Operating System), Privacy statement service, Disclaimer service) and not groups of these. The functional users that provide input data detect single occurrences of events that the Web Advice Module must respond to (MM The standard level of granularity).
By measuring at the standard level of granularity it is possible to use this measurement not only for the purpose of this measurement (see §3.1) but also for benchmarking purposes, since most benchmark data is available at the standard level of granularity.

3.5 Identify the FUNCTIONAL USERS and PERSISTENT STORAGE

3.5.1 Identifying functional users

The functional users are senders and/or intended recipients of data in the FUR of the Web Advice Module.

In this case study the functional users are:

- The human customers that use the Web Advice Module to get advice.
- The application manager that maintains the editable texts within the application.
- The system clock that controls the inactivity control process.

Also all identified pieces of software in this case study must be considered functional users of the Web Advice Module (GSOA The functional users of services):

- Privacy statement service
- Disclaimer service
- Back Office service

There is no exchange of data between the Web Advice Module and the Mortgage calculation service. By pressing the Calculate mortgage button the customer is directed to the Mortgage calculation service. From that point the customer can start the functionality of that service. The direction from the Web Advice Module to the Mortgage calculation is mere navigation that does not start a functional process. Consequently, this should be ignored in the measurement of Web Advice Module (BAG Menus and the triggering Entry).

The boundary of the Web Advice Module is a conceptual interface between this piece of software and its functional users. The boundary allows the measurer to distinguish, without ambiguity, what is included inside the measured software from what is part of the measured software's operating environment. (MM Functional Users). It is indicated by the dashed line:
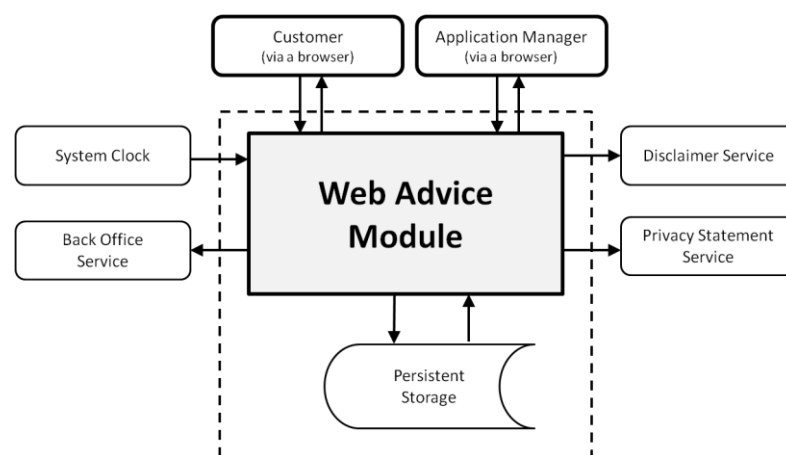
The arrows represent the exchange of data between functional users and the Web Advice Module. The arrows to the Disclaimer and the Privacy statement are only outward bound, since the data these services present are directly presented to the human user (by means of the web browser) and not to the Web Advice Module and no feed-back is required.

3.5.2 Identifying persistent storage

Persistent storage is storage which enables a functional process to store a data group beyond the life of the functional process and/or from which a functional process can retrieve a data group stored by another functional process, or stored by an earlier occurrence of the same functional process, or stored by some other process. In the COSMIC model, persistent storage has a specific meaning, namely that it exists only within the boundary of the software being measured, i.e. that it isn't a functional user of the software being measured. In other words, it is storage where the FUR are not concerned with how those data accesses are handled by any other software.

From the requirements with respect to retrieving and storing data it appears that no storage requires intervention of other software, therefore all storage mentioned is persistent storage. It implies that for all retrieve and store activities, Read and Write data movements must be identified, rather than Exit / Entry pairs.

4   Mapping phase

The Mapping Phase is intended to express the Functional User Requirements in a form to which the COSMIC Generic Software Model can be applied (MM Applying the Generic Software Model). Applying the COSMIC Generic Software Model means identifying the set of triggering events sensed by each of the functional user (types) identified in the FUR, and then identifying the corresponding functional processes, objects of interest, data groups, and data movements that must be provided to respond to those events.

In this chapter sections in *italics* are explanatory text from the authors that are not a part of the actual measurement process.

4.1   *Identify FUNCTIONAL PROCESSES*

The first step in the mapping phase is to determine the unique event types and the corresponding functional processes. The most important general advice is that it is almost always useful to try to identify first the separate events that the software must respond to ('triggering events'), since each such event gives rise to usually one (but sometimes more than one) functional process.

A functional process is a set of data movements, representing an elementary part of the Functional User Requirements for the software being measured, that is unique within these FUR and that can be defined independently of any other functional process in these FUR. It starts processing on receipt of a data group moved by the triggering Entry data movement of the functional process. Its set of data movements is needed to meet the FUR for all the possible responses to its triggering Entry. (MM Identifying functional processes).

A triggering event is an event that causes one or more functional users of this software to generate one or more data groups, each of which will subsequently be moved by a triggering Entry (MM Identifying functional processes)

In this case study the following functional processes are identified (recipients of the result of the functional process are shown in grey):

Triggering event	Functional user	The data moved by the Triggering Entry	Functional process
Request for rent or buy advice	Customer	Proposition choices	Display rent or buy advice
Request for an advice session (on renting)	Customer Back Office Service	Customer data	Advice session request form
Request for an advice session (on mortgage)	Customer Back Office Service	Customer data	Advice session request form
Need to select a text to be changed	Application Manager	Request to display all editable texts	Select text
Need to edit selected application text	Application Manager	Edited text for selected application text item	Edit text

Triggering event	Functional user	The data moved by the Triggering Entry	Functional process
Interest in principles	Customer	Principles button pressed	Display principles
Interest in disclaimer	Customer Disclaimer Service	Disclaimer button pressed	Invoke disclaimer service
Interest in privacy statement	Customer Privacy statement Service	Privacy statement button pressed	Invoke privacy statement service
Time interval elapsed	System Clock	Threshold exceeded message	Display time-out message
User decides to end Module session	Customer	Close Module session message	Close Module session

4.1.1   One or more functional processes to obtain an advice?

When an advice has been obtained based on the first time the advice module has been used, the advice module offers the possibility to modify the choices given the first time and to obtain a modified advice. Is this a second functional process or not?

When used for the first time this page presents default choices that come from the application itself. In any round of use of the advice module the user can change the choices to one or more questions to generate an advice. The only difference between the first use of the module and any following use is that no advice is shown when the module is used the first time. With any following use the previous advice, corresponding to the previously entered choices, is still displayed. As these uses are repetitions of the same functionality, the uses are occurrences of one functional process (type). In consequence there is only one triggering event type that represents its many triggering event occurrences. (See MM Types versus Occurences)

4.1.2   One or two advice session request form functional processes?

There is a requirement for a request form for an advice session on renting or mortgage, which leads to one functional process. Although the request has been implemented in two (slightly different) ways, the requirement is still for one functional process, i.e. we count the requirements, not their implementation. In the delivered software the FUR have been implemented in the following two web pages:

4.1.3 Maintenance functional processes

In §2.10 the requirements state that a number of texts used by the Web Advice Module must be maintainable without the assistance of a programmer. This requirement is not very precise. We therefore assume the following:

- *The maintenance functional processes have a human functional user that is different from a regular bank customer user, who we call the Application Manager.*
- *Although the functionality of the Web Advice Module differentiates the application texts into 1 introductory text, 3 general advice texts, 5 proposition texts and 10 advice texts this differentiation is irrelevant to the Application Manager functional user. Therefore no distinctions are made between the maintenance of different types of application texts. Whether or not something is an object of interest can be different for different functional users and must be determined anew for each functional process.*
- *The requirements do not mention the need to create or delete new application texts. This implies that the number of application texts is fixed. This means that all application texts exist at all times, so there is no need to identify functionality to create or delete application texts, only to select and display the right text and to edit that text.*
- *It is assumed that the Application Manager maintains the data via a web browser and that maintenance via a web browser or via application software gives no difference in functional size of the Web Advice Module.*

Elaborating from these assumptions the requirements from section §2.10 lead to two (maintenance) functional processes: Show text (see §0) and Edit text (see §5.4).

4.1.4 Display principles functional process

The question is whether this functional process violates rule c) of a functional process ([MM](#) The approach to identifying functional processes) that requires a functional process to comprise at least two data movements. As the requirements do not mention any need to maintain the Principles statement, we assume that the information text of the principles is hard-coded as part of the web page. This only means that no Read is necessary to access it in a functional process. But when the Principles are displayed it is a data group that is moved across a boundary. Moving a data group across a boundary means processing data about an object of interest.

According to rule a) of the Exit ([MM](#) Identifying Exits) output of fixed text (the display of the Principles) shall be modeled as one Exit Together with the request to display the Principles, the functional process of displaying the Principles of the Web Advice Module satisfies the requirements of rule c) that a functional process shall comprise at least two data movements.

The display of the Principles of the Web Advice Module can be considered identical to a Utility service ([GSOA](#) Utility services), like the Privacy statement service and the Disclaimer service. However, the Privacy statement service and the Disclaimer service are existing services, so only the functionality to invoke the existing services is part of the scope of this measurement (see also the next section). The display of the Principles of the Web Advice Module is new functionality and should be measured as part of the estimate of the required effort to build the software.

4.1.5   Functional processes that invoke existing services

The Web Advice Module uses two existing services that only need to be invoked without any data entered by the functional user of the Web Advice Module:

- *Disclaimer service*
- *Privacy statement service*

The functionality of these services is not in scope for the measurement, but the Web Advice Module must have functionality to invoke (or call upon) the functionality of these services. That is why we need to identify two functional processes to invoke existing services.

These functional processes appear to the functional user of the Web Advice Module in a similar way as the Display principles functional process. The measurement result is different, however, since the Display principles functional process is in scope for the measurement. The Invoke functional processes only invoke existing SOA services that are out of scope for this measurement.

*Communication with other services uses the standard COSMIC model for the exchange of data between two peer pieces of software: the Entry/Exit pair (GSOA Services). For these invoke functional processes we assume that no feed-back is required (see §***Fout! Verwijzingsbron niet gevonden.***) so these invoke functional processes of the Web Advice Module therefore only consist of a triggering entry and an exit to send a request to invoke a SOA service.*

4.1.6   Invoking existing services

The functionality of the Web Advice Module consists of both a business application (governed by the BAG) and SOA-services (governed by the GSOA). Invoking of services via buttons has some resemblance to menu functionality (BAG Menus and the triggering Entry). According to this section of the BAG, menu functionality that "enables the user to move around the software, but which does not launch any functional process" is to be ignored. Pressing the button that enables the user to move towards the Mortgage calculation service is therefore ignored in this measurement. However, pressing either the Disclaimer and Privacy buttons invokes, i.e. launches, a functional process. Counting these two invoke functional processes is therefore in line with both the BAG and the GSOA, even though most of the functionality of these functional processes is outside the scope of the software to be measured.

4.1.7   Display time-out message

Measuring timer functionality requires clear specifications on what functionality is allocated to the hardware or the operating system, and what is specifically allocated to the software part (RTAG Timer functionality). We do not have this information, so we have to make some assumptions.

- *We assume that all functional processes take place via the web server and that the inactivity time relates to the interactions of human users with these functional processes.*
- *We also assume that the inactivity timing is existing functionality of the web server. When the inactivity limit of a customer has been reached, the system clock notifies the web server, which triggers the Web Advice Module to show the inactivity message and delete the customer's data. This is the logical cause of events, since the display of the time-out message and the delete of data is Web Advice Module functionality. Physically, this may be solved in a different way.*
- *The system clock is assumed to be the functional user that triggers the time-out functional process.*

4.1.8   Close Module Session

The user can at any time end the Web Advice Module session by clicking on the 'close window' button of any screen that is headed 'Web Advice Module'. We assume this action causes a functional process of the Web Advice Module to close all windows and to delete any user data that has been entered.

4.2   Identify OBJECTS OF INTEREST and DATA GROUPS

The second step in the mapping phase is to identify the object(s) of interest for each functional process and the corresponding data groups.

Objects of interest are defined as:
Any 'thing' in the world of the functional user that is identified in the Functional User Requirements, about which the software is required to process and/or store data. It may be any physical thing, as well as any conceptual object or part of a conceptual object. (MM Identifying objects of interest and data groups).

Note that an object of interest is not necessarily the same for all functional users throughout the functionality to be measured. In this Case Study this becomes most visible in the texts that are used for the advice and advice details. The same attributes are part of multiple data groups and multiple objects of interest, depending on the functional user. (See also [MM](#) Parameter (code) tables and objects of interest, which deals with this subject)

Data groups are defined as:
A data group is a distinct, non empty and non ordered set of data attributes where each included data attribute describes a complementary aspect of the same object of interest ([MM](#) Identifying objects of interest and data groups).The following objects of interest and corresponding data groups can be identified Note that according to the Data movement uniqueness Rule a) all data describing any one object of interest that is required to be entered into one functional process shall be identified as one data group moved by one Entry (in exceptional cases multiple Entries must be identified). The same equivalent rule applies to any Read, Write or Exit data movement in any one functional process (MM Data movement uniqueness and possible exceptions). In the measurement practice it suffices therefore to only identify the data group consisting of all data attributes of each object of interest, or only their objects of interest.

Functional process	Object of interest	Data group(s)
Display rent or buy advice	Customer advice	Customer proposition choices General advice Customer advice details
Advice session request form	Customer	Customer data
	Customer advice	Customer proposition choices General advice Customer advice details
Select text	Module application texts	Application texts
Edit text	Module application texts	Application texts
Display principles	Principles	Principles text
Invoke disclaimer service	Disclaimer	Invoke disclaimer request
Invoke privacy statement service	Privacy statement	Invoke privacy statement request
Display time-out message	Inactivity of a specific user	Threshold exceeded message
Close WAM session	WAM session	Close WAM session message

4.2.1   OOI Customer advice

The central object of interest in the Web Advice Module is the Customer advice whether to buy or rent a house. This is the central concept and there are three data groups that describe complementary aspects of the advice:

▪ *The Customer proposition choices the advice is based on. By default the choice is neutral for each proposition, but can be changed by the customer to reflect his or her personal situation. For each customer five proposition choices are always stored.*

▪ *The General advice whether to buy or rent a house. Based upon the proposition choices the business rules described in §2.4 lead to an advice to the customer to buy or to rent. For each customer there is only one occurrence (out of the three possible values) at a time.*

▪ *For each proposition choice the Web Advice Module contains Customer advice details that can be shown to the customer, if relevant, for each proposition choice that led to the general advice. The example in the Advice page in §2.3 shows two advice details that have led to the general advice to rent a house. For each customer up to five occurrences (out of the ten possible values) apply, based on the selected proposition choices.*

These three aspects correspond to different data groups because the aspects have different frequencies of occurrence (BAG Identification of objects of interest, data groups and data movements). The software needs to store all the data groups in a 'web session' to be able to send them to the functional user Back Office Service if the customer wants to have an advice session.

4.2.2   OOI Customer

In order to be able to process a request for an advice session the functional user Back Office Service needs to have information about the Customer. This object of interest only has one data group, containing the Customer data that is to be sent with the request for an advice session. This includes the type of session the Customer requires.

4.2.3   OOI Module application texts

For this object of interest there is one data group that contains all attributes of the dynamic application texts of the Web Advice Module from the perspective of the Application Manager. For the Application Manager, it is irrelevant to distinguish the different types of text as four different data groups describing the OOI Generic advice texts.

4.2.4   OOI Principles

For this object of interest there is only one data group, describing the principles of the Web Advice Module.

4.2.5   OOI Disclaimer

This object of interest only contains one data group, describing the general disclaimer of the bank.

4.2.6   OOI Privacy statement

This object of interest only contains one data group, describing the privacy statement of the bank.

4.2.7   OOI Inactivity of specific user

In order to fulfil the time-out requirement the Web Advice Module receives a signal for every distinct user of which the inactivity time has passed the threshold value.

4.2.8   OOI WAM session

When a user closes a WAM session, the WAM must delete any data stored for that user. The command to close a session is data that changes the state of the OOI WAM session.

4.3   Identify DATA ATTRIBUTES

The third step in the mapping phase is optional, since it has no effect on the measurement of new software. It is very useful as a basis for functional size measurements of future maintenance of the software. When the data attributes are known in detail, it is easier to determine whether a data group has been changed or not.

It is also very useful though to understand the functionality in detail. That is why this optional step is included in the case study.

The data groups identified in the previous section contain the following data attributes:

Data group	Data attributes
Customer proposition choices	Choice #1, choice #2, choice #3, choice #4, choice #5
Customer data	Initials, family name, gender, date of birth, street, number, ZIP-code, city, country, phone number, e-mail address, account number, requested type of session
Module application texts	Introductory text, rent advice, buy advice, mixed advice, proposition #1, proposition #2, proposition #3, proposition #4, proposition #5, buy detail #1, buy detail #2, buy detail #3, buy detail #4, buy detail #5, rent detail #1, rent detail #2, rent detail #3, rent detail #4, rent detail #5
Introductory text *(subset of application texts)*	Introductory text
General advice *(subset of application texts)*	Rent advice, Buy advice, Mixed advice
Generic proposition texts *(subset of application texts)*	Proposition #1, proposition #2, proposition #3, proposition #4, proposition #5
Generic advice details *(subset of application texts)*	Buy detail #1, buy detail #2, buy detail #3, buy detail #4, buy detail #5, rent detail #1, rent detail #2, rent detail #3, rent detail #4, rent detail #5
Customer advice details *(subset of generic details)*	Buy detail #1 *or* rent detail #1 *or* empty Buy detail #2 *or* rent detail #2 *or* empty Buy detail #3 *or* rent detail #3 *or* empty Buy detail #4 *or* rent detail #4 *or* empty Buy detail #5 *or* rent detail #5 *or* empty
Principles text	Principles of the Web Advice Module
Invoke disclaimer request	Request to show the disclaimer
Invoke privacy statement req.	Request to show the privacy statement
Threshold exceeded message	Message that a distinct uses exceeded the inactivity threshold
Close WAM session	Close WAM session message

4.4   Discussion on the mapping phase

From the length of the discussion sections in this chapter one might get the impression that the COSMIC method is difficult to apply. The reader should bear in mind that a large portion of this discussion deals with the correct mapping of the requirements to the COSMIC model. This is an important step in the COSMIC measurement process. It is a deliberate choice of the authors to discuss this part in depth, since in practice the interpretation of the requirements is the most important part of the measurement process.

5 Measurement phase

With the functional processes, objects of interest and data groups identified in the mapping phase, the actual measurement can take place by identifying all data movements in each functional process. In the Business Application domain, when a piece of software to be measured can generate messages without user data, by convention one single Exit is identified to represent all of those software messages (MM "Identifying Exits", BAG "Error messages and conditions").

In this chapter sections in *italics* are explanatory text from the authors that are not a part of the actual measurement process.

Each functional process identified in §4.1 will be described in detail by a message sequence diagram, describing the data movements that make up the functional process.

5.1 *Display rent or buy advice*
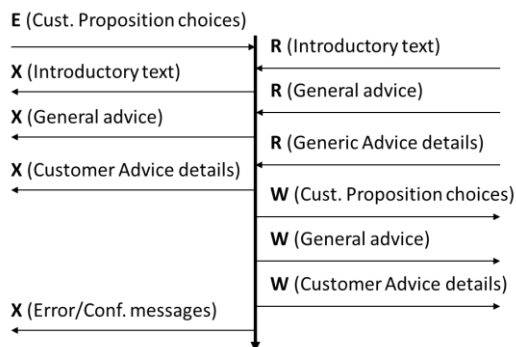
The functional process starts when the customer wants an advice from the Web Advice Module (triggering event) and enters the proposition choices and the trigger to generate an advice, based on the given choices (triggering Entry).

Based on the given choices an advice is generated with advice details.

This functional process has a size of **11 CFP**.

See also the discussion in §4.1.1.

5.1.1 Measuring display rent or buy advice

The advice details are read from persistent storage, based on the entered proposition choices. Since the occurrences of the general advice and the advice details describe different objects of interest (BAG Identification of objects of interest, data groups and data movements) and have different frequencies of occurrence, they are counted as separate data movements.

If a data group survives the functional process using it, it needs to be stored. The fact that the proposition choices from the user are known when a customer fills in an advice form indicates that this data group survives the functional process.

5.2 Request form for an advice session (back office service)

From the application two kinds of advice sessions can be requested (see §2.5). Both requests are handled by the back office and are functionally identical. So only one functional process is identified (MM Identifying functional processes).
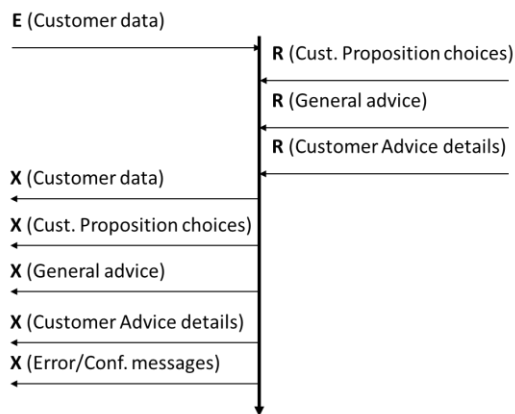
The functional process starts when the customers decides to request an advice session (triggering event) and activates a back office form, either to request a rent advice session or a mortgage advice session in which the customer enters his or her data.

This functional process has a size of **9 CFP**.

See also the discussion in §4.1.2.

5.2.1   Measuring advice session forms

The functional process reads the (customer) proposition choices, the general advice and the advice details that have been stored to be available for this functional process (see §5.1).

The functional process Exits the customer data, the (customer) proposition choices, the general advice and the advice details to the back office. These represent four different objects of interest, so four separate Exits are identified. The requirements deliberately specify that not only the proposition choices, but also the general advice and the advice details have to be sent to the back office. Due to the flexibility to maintain the texts which the customer receives from the Web Advice Module, given a certain set of proposition choices, the requirements want to ensure that the human advisor receives exactly the same input values and advice texts the customer has received.

5.2.2   Measuring the cancel request

Pressing the button to cancel the request on the advice session form screen is not a separate functional process. It is a way to terminate the functional process of requesting an advice session without sending out the request. In the COSMIC method this is called a control command (MM Control commands) which is to be ignored in the measurement. A control command is specific to the business application domain.

5.3 Select text (maintenance functional process)

E (Display texts)
R (Application texts)
X (Application texts)
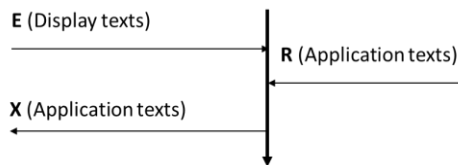
The Application manager can decide to change one or more editable texts within the Web Advice Module (triggering event) and requests the functional process to read and display all application texts that can be edited.

This functional process has a size of **3 CFP**.

See also the discussion in §4.1.3.

Note that there is no requirement for error or confirmation messages, so they are not counted.

5.3.1 Measuring Select text

The functional process reads all the texts from persistent storage and displays them to the Application manager so he or she can select an application text for editing via the browser. This functional process is the first so-called 'enquire-before-update' process which precedes the 'edit text' step. Both steps are identified as separate functional processes (BAG Separate functional processes).

5.4 Edit text (maintenance functional process)

E (Changed text)
W (Application text)

The decision to change one of the editable application texts (triggering event) and the subsequent update of an application text is a separate functional process (BAG "Separate functional processes"). The changed text is the triggering entry for this functional process. The changed text is written to persistent storage.

This functional process has a size of **2 CFP**.

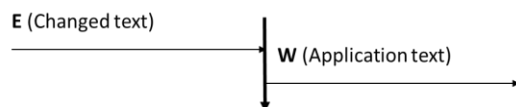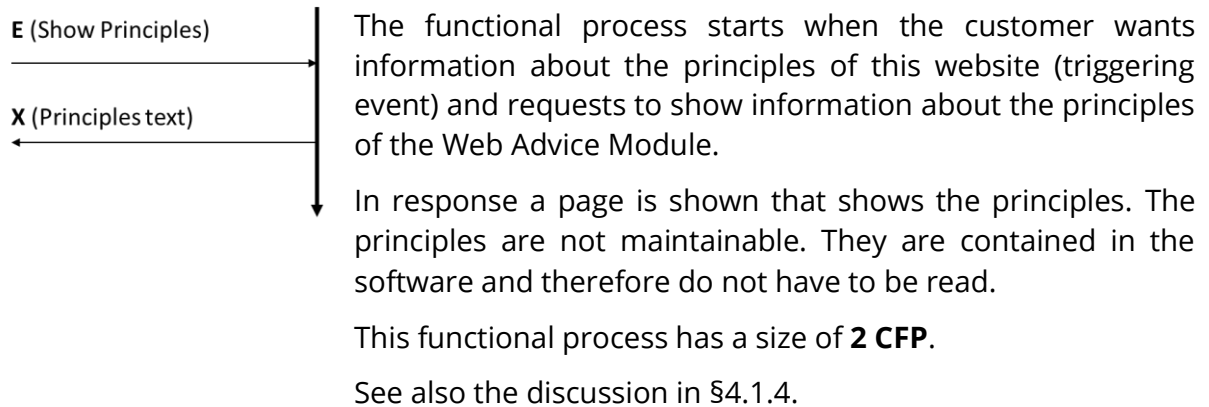See also the discussion in §4.1.3.

Note that there is no requirement for error or confirmation messages, so they are not counted.

5.4.1 Measuring Edit text

This functional process is the second step following the so-called 'enquire-before-update' process in which the text is edited and updated. Both steps are identified as separate functional processes (BAG Separate functional processes).
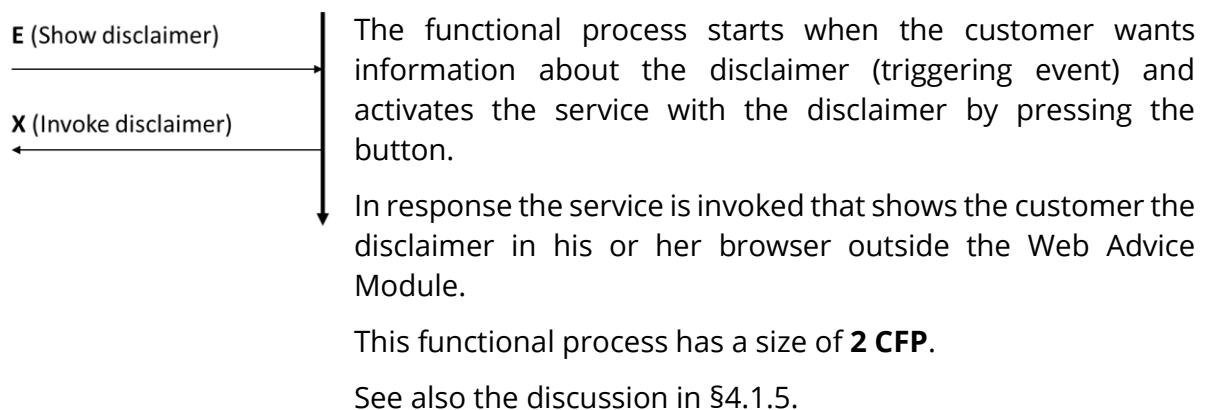
5.5   Display principles of this module

E (Show Principles)

X (Principles text)

The functional process starts when the customer wants information about the principles of this website (triggering event) and requests to show information about the principles of the Web Advice Module.

In response a page is shown that shows the principles. The principles are not maintainable. They are contained in the software and therefore do not have to be read.

This functional process has a size of **2 CFP**.

See also the discussion in §4.1.4.

5.5.1   Measuring display principles of this module

The facts that the triggering Entry is the result of pressing a button and the output is fixed, non-maintainable text do not affect that this is a simple enquiry.

5.6   Invoke Disclaimer service

E (Show disclaimer)

X (Invoke disclaimer)

The functional process starts when the customer wants information about the disclaimer (triggering event) and activates the service with the disclaimer by pressing the button.

In response the service is invoked that shows the customer the disclaimer in his or her browser outside the Web Advice Module.

This functional process has a size of **2 CFP**.

See also the discussion in §4.1.5.

5.7   Invoke Privacy statement service

E (Show privacy statement)

X (Invoke privacy statement)

The functional process starts when the customer wants information about the privacy statement (triggering event) and by pressing the button activates the service with the privacy statement.

In response the service is invoked that shows the customer the privacy statement in his or her browser outside the Web Advice Module.

This functional process has a size of **2 CFP**.

See also the discussion in §4.1.5.

5.8   Display time-out message

E (Time interval)

W (Proposition choices)

W (General advice)

W (Advice details)

X (Time-out screen)

As stated in §4.1.7 the Web Advice Module uses existing functionality of the web server, which uses the system clock of the operating system. If the inactivity time exceeds the threshold the system clock notifies the web server, which triggers the Web Advice Module to show the inactivity message and deletes the customer's data.

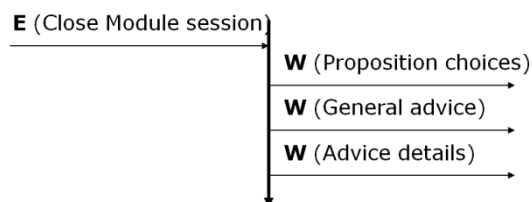A requirement to delete data is represented by a Write data movement.

This functional process has a size of **5 CFP**.

See also the discussion in §4.1.7.

5.8.1   Measuring display time-out message

We have assumed that the Web Advice Model must delete any data entered by a user if the user's session closes as a result of the Web Advice Module receiving a timeout message ('threshold exceeded') from the system clock. If our assumption is wrong and the deletion of user data is performed by the webserver or some other function of the operating environment, then this functional process would not need the three Write data movements for the deletion of user data.

5.9   Close Module session

E (Close Module session)

W (Proposition choices)

W (General advice)

W (Advice details)

The user can close the session of the Web Advice Module by closing any window headed 'Web Advice Module' or by closing his or her browser. As stated in §2.9 the entered data is no longer available. This is represented by Write data movements to delete these data from permanent storage.

This functional process has a size of **4 CFP**.

5.9.1   Measuring close Module session

We have assumed that the Web Advice Model must delete any data entered by a user if the user's session closes as a result of a user clicking on a 'close window' button. If our assumption is wrong and the deletion of user data is performed by the webserver or some other function of the operating environment, then this functional process would not be needed at all.

5.10 List of data movements

Functional process	Data movement	Type	CFP
Display rent or buy advice	Customer proposition choices	E	1
	Introductory text	R	1
	General advice	R	1
	Generic advice details	R	1
	Introductory text	X	1
	General advice	X	1
	Customer advice details	X	1
	Customer proposition choices	W	1
	General advice	W	1
	Customer advice details	W	1
	Error/Confirmation messages	X	1
Advice session request forms	Customer data	E	1
	Customer proposition choices	R	1
	General advice	R	1
	Customer advice details	R	1
	Customer data	X	1
	Customer proposition choices	X	1
	General advice	X	1
	Customer advice details	X	1
	Error/Confirmation messages	X	1
Select text	Display texts	E	1
	Module application texts	R	1
	Module application texts	X	1
Edit text	Selected text	E	1
	Module application text	W	1
Display principles of this module	Principles button	E	1
	Principles	X	1
Invoke Disclaimer service	Disclaimer button	E	1
	Invoke disclaimer	X	1
Invoke Privacy statement service	Privacy statement button	E	1
	Invoke privacy statement	X	1
Display time-out message	Time interval	E	1
	Customer proposition choices	W	1
	General advice	W	1
	Customer advice details	W	1
	Time-out screen	X	1
Close WAM session	Close window	E	1
	Customer proposition choices	W	1
	General advice	W	1
	Customer advice details	W	1
Total functional size			**40**