Project FoSIBLE
Fostering Social Interactions for a Better Life of the Elderly

D4.1 – Early and advanced, stable prototypes of the main social media platform

Responsible

Kaasa solution

AIT

Participants

USI

Mauser

IMS

CURE

UTT

Deliverable

D4.1 - Early and advanced, stable prototypes of the main social media platform

Version: Final
Date: 19/11/2013
Dissemination level: (PU, PP, RE, CO): PU

Abstract

This document gives an overview on the design and development of the main social media platform as well as the development of the server side of the application and the client applications in the FoSIBLE project. In this context, the social media backend, the data broker, external data sources and the internet capable TV set are described followed by the presentation of the overall system architecture. Furthermore, a timeline illustrates the previous development steps. At the end of this document the realized quality assurance and performed tests for proving the platform are explained.

Table of Content

1. Introduction

1.1 Purpose of the Document

The purpose of this summary is to specify/report on the work on the interactive TV widget and its modules that are used for evaluation in the test households of the FoSIBLE project.

The technical infrastructure and the system architecture of the social platform are described in detail.

1.2 Document Scope

Objectives of the WP
Design and development of the final platform for the services, programs and devices which will be offered to the users. Developers, scientists and selected end-users test the developed material and verify that the development meets the defined goals.
Description of work
Task 4.1: Main social media platform development: Kaasa with support from AIT will develop the main social media platform. Task 4.1.1: Development of the server side of the application. Task 4.1.2: Development of the client applications, containing the elements: Client UI realized as overly to the television broadcast or the console menu; client connectivity based on standard internet protocols; framework for client input able to run with the new input devices developed in WP3 and WP5 and with standard input devices such as keyboards and mouse for faster evaluation of the software and as a fallback solution. Task 4.2: Virtual generational community application: UTT will develop an application aiming at formulating virtual generational communities, which permits to discuss/ to exchange feelings and experiences between elderly people. These generational communities will be the way to provide social support within peers. This development will be integrated with the other applications developed by the other partners. Task 4.3: Development of gaming and entertainment applications: Kaasa will develop Wii plug-ins aiming at innovative ludic ways of social interaction and mental and physical health of elder persons. Task 4.4: Development of communication tools: Uni Siegen develops a client application to provide messenger service aiming at supporting communication on a spectrum from peripheral awareness to active involvement which will be integrated with the other

applications. Task 4.5: Integration of the technical software and hardware solutions from WPs 3, 4 and 5: (Responsible: all participants) Prototype development of the new input/output devices which combine future concepts of input and different sensors for a new way of controlling interactive social media centre applications and extend towards natural Interaction: through speech, haptics, gestures to improve current user interfaces by leveraging human senses with other natural communication channels (e.g. emotional)-
Deliverables of the WP D4.1: Early and advanced, stable prototypes of the main social media platform. Delivery dates: M12 and M34 D4.2: Early and advanced, stable prototypes of software applications (generational community application, gaming and entertainment application, communication tools). Delivery date: M12 and M34

1.3 Partners

This section defines the contribution of each project partner to this document.

1. Kaasa solution has contributed information about: the social platform based on an Open Source Community software system (JomSocial) to attach various sensors in the future

2. University of Siegen has contributed information about: modules of the code of the integrated widget (TV recommendation/chat)

3. Mauser and Fraunhofer IMS have contributed information about: sensors and sensor data in furniture that are interpreted by the technical platform and can then be presented in the integrated widget or the social platform

4. CURE was involved in the UI creation of the community software and the TV widget

5. UTT was involved in the design of the prototypes based on the results of deliverable D2.1 (end-user requirements) and D2.2 (system functionality).

2. Management Summary

This document contains a description of deliverable D4.1 of the project FoSIBLE, the early and advanced, stable prototypes of the main social media platform. In this section, the components of the technical system regarding the different deliverables D4.1 and D4.2 will be listed and described in the following subsections.

Components of D4.1:

- Social media backend (JomSocial)

- Data broker

- External data sources (for TV Guide)

- Internet capable TV set

Components of D4.2 (explained in deliverable D4.2):

- Internet capable TV set

- TV widget

- Tablet PC with Android OS

- Sensors included in furniture

2.1 Social media backend (JomSocial)

After market research the decision has been made to use the Open Source community software JomSocial as a basis for the social media backend for the FoSIBLE project.

The modular software offers the possibility to adjust the community according to the needs of the FoSIBLE project. New modules can easily be added and the open structure enables all involved partners to add new input sources (sensors, data etc.) at any time. This makes sure that additional modules can be added to the existing system if requested by the users in the later research stage.

After the initial installation, the software had to be updated a couple of times due to changes in the software on the server that is needed to run the backend software.

2.1.1 Technical details of the soft- and hardware

The FoSIBLE community platform is available at the link: fosible.kaasa.com. The system is installed on a Dell Server with the following technical configuration:

- CPU: Intel(R) Core(TM) i7 CPU 930 with 2.80GHz
- Memory: 24 GB RAM
- Operating system: Debian Squeeze Linux Kernel 2.6.32-5

Installed software that is necessary for the JomSocial system:

- Web Server Apache/2.2.16 (Debian)
- Database application: MySQL v5.1.66-0
- PHP 5.3.3-7

The current version of the installed JomSocial platform is v2.3.

2.1.2 Maintenance of the server and the software components

The latest version (v2.4.) is already available for download on jomsocial.com but the update will be done at a later time because it has to be done on the test server before the rollout on the live machine.

The operating system software on the server is updated on a regular basis (once a week). This includes all official patches for the OS components that are released by DEBIAN available via the 'apt-get update' function and that are official DEBIAN updates. Also critical patches for the web services that are running on the server and that are needed by the social media backend (MySQL, PHP) are included in the updates. All important components that are needed to restore the server in case of a hard or software failure are backed up every day (database, configuration files etc.).

In case of a hard or software problem, the server can be restored within a couple of hours in order to be available to the FoSIBLE community again.

2.1.3 Installation process

The basic installation of the JomSocial software has been made on a test machine in February 2011. After the first tests the social media backend has been moved to a productional server where it is available to the public now.

After the community software became available to all project partners the testing on the live server began and feedback was collected from all involved partners.

Adjustments to the community software and the installed modules fit the requirements of the planned scenarios and also meet the requirements of the partners involved in the later

field research. To fill the platform with content exemplary users have been created in the platform based on the personas used in the scenarios of FoSIBLE, and clubs were constructed to meet the scenarios of FoSIBLE.

The layout of the community platform has not changed since the first configuration as it turned out that the platform will not be needed for the evaluation at UTT and USI. The resources that have been allocated for this task have been transferred to more important tasks in the development of the TV widget.

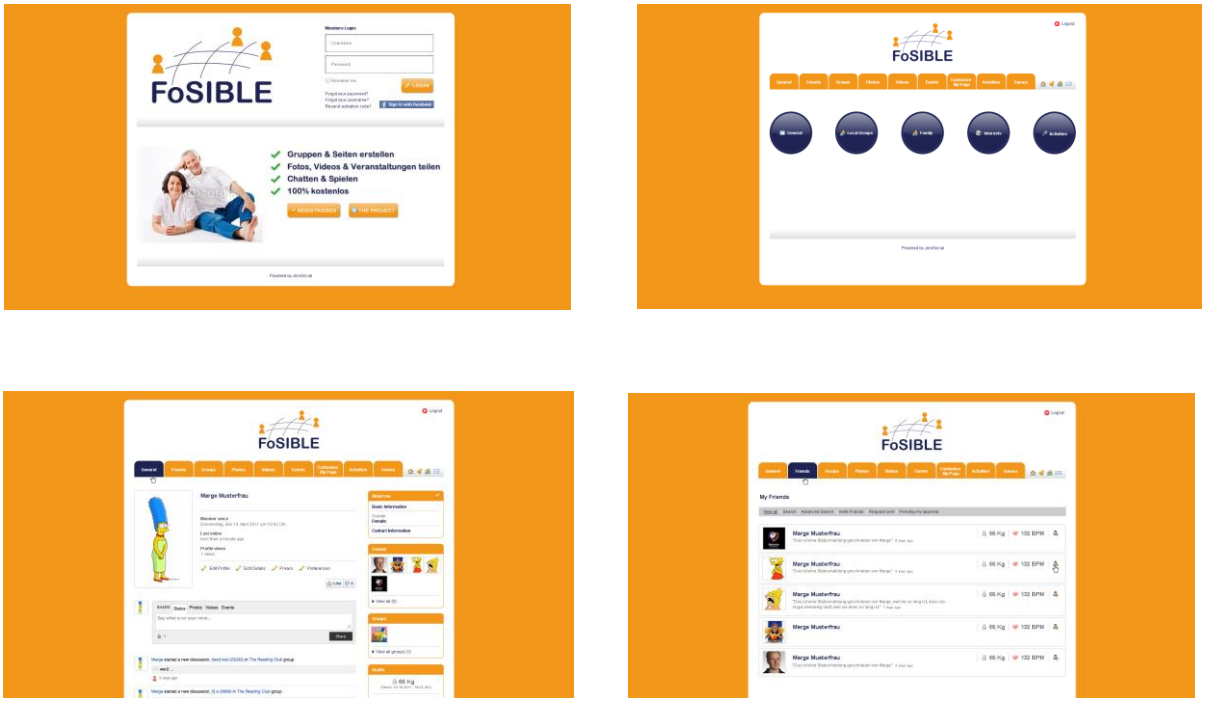

Figure 1. Screenshots of the test environment.

2.2 Data broker

Besides the community software, the server also acts as a data broker (see Figure 2). All input and output data is processed on the server.

Sensor data that is sent to the data broker can be processed by applying fixed rules so that the abstract data sent by the sensors are processed to human readable and reasonable information.

The incoming data can either be pre-processed giving (1) a final value that can directly be shown on the platform or (2) data that need some interpretation by the data broker before the final information is handed over to the social community platform.

The sensor data is posted to the server in form of an xml file. Every attached sensor has its own xml scheme that has to be matched so that the server can interpret the input and process it.

The incoming and processed information is transmitted directly to the social community database and can be accessed by the platform and the TV widget.

All interfaces were discussed with the relevant project partners and have been configured according to the technical needs of each partner.

As standard web interfaces are used to exchange data between the different elements of the whole system it is possible to add more input devices like sensors or other data sources at a later stage of the development.

2.3 External data sources

The system does not only use data like the user input or sensor data but also external EPC data. The source for the EPC data is an external TV guide website as Samsung does not provide access to the internal EPG data in the 2010 models. This data is relevant for the TV recommendations described in deliverable D4.2.

2.4 Internet capable TV set

After market research on the existing internet capable TV sets the decision has been made to use a Samsung Smart TV for the installation in the test households. The Samsung TVs are HBB TV enabled and developers can develop their own widgets for the TV with the help of the Samsung TV SDK.

Detailed information about the Internet capable TV set and the TV widget are described in deliverable D4.2.

3. Overview of the technical system

Figure 2 gives a general overview of the technical system showing the components described above as well as the user and the way data is exchanged. The grey area represents a test household with the internet capable TV set and other sensors.

The user in the household can interact with the widget on the TV set with the help of the TV remote. In the case that the user wants to comment on a book review or enter other text, a text input request is sent to the data broker. The tablet then fetches this request and opens the interface for text input. Once the text is written, it is sent to the data broker and stored to the database from where the widget receives that data and the text appears on the TV set.

The data from sensors inside the furniture is going a similar way. All data collected from the different sensors are pre-processed by the IMS data box and from there the files are sent to the data broker in a predefined form. The information is then stored to the database and can be obtained by the widget and presented there. As the data is available on the server it is possible to e.g. allow friends to see this data as well. In this way a friend can e.g. see that a friend is active.

Another data source is the VitaDock application that is running on iOS devices like the iPhone 5 (see also D4.2). The vital data that the user collects with the device (blood pressure, weight etc) is sent to the VitaDock server that has an open interface for the FoSIBLE backend. The information is fetched by the data broker and makes it available for the TV widget. The VitaDock data could be made public e.g. to a doctor or relatives who take care of the user.

The centre of the system is the private household with the TV set/widget. The end user creates data sets with the VitaDock application and the sensors within the household that are sent to the external FoSIBLE backend/data broker. The same happens with text input through the tablet PC that is sent to the FoSIBLE backend. After the input data is processed it is visualized on the TV widget (not only in the widget of the user but also in widgets of other users using the FosIBLE system). The content of clubs is available to everyone, while vital data is planned to be available to a closed user group only (relatives/doctor) in the future. This is not possible in the prototype at the moment. As this feature will not be evaluated in the households, this feature is put on hold for now. Nevertheless, an implementation of this functionality is technically possible at a later stage as the JomSocial software offers possibilities to create closed user groups.

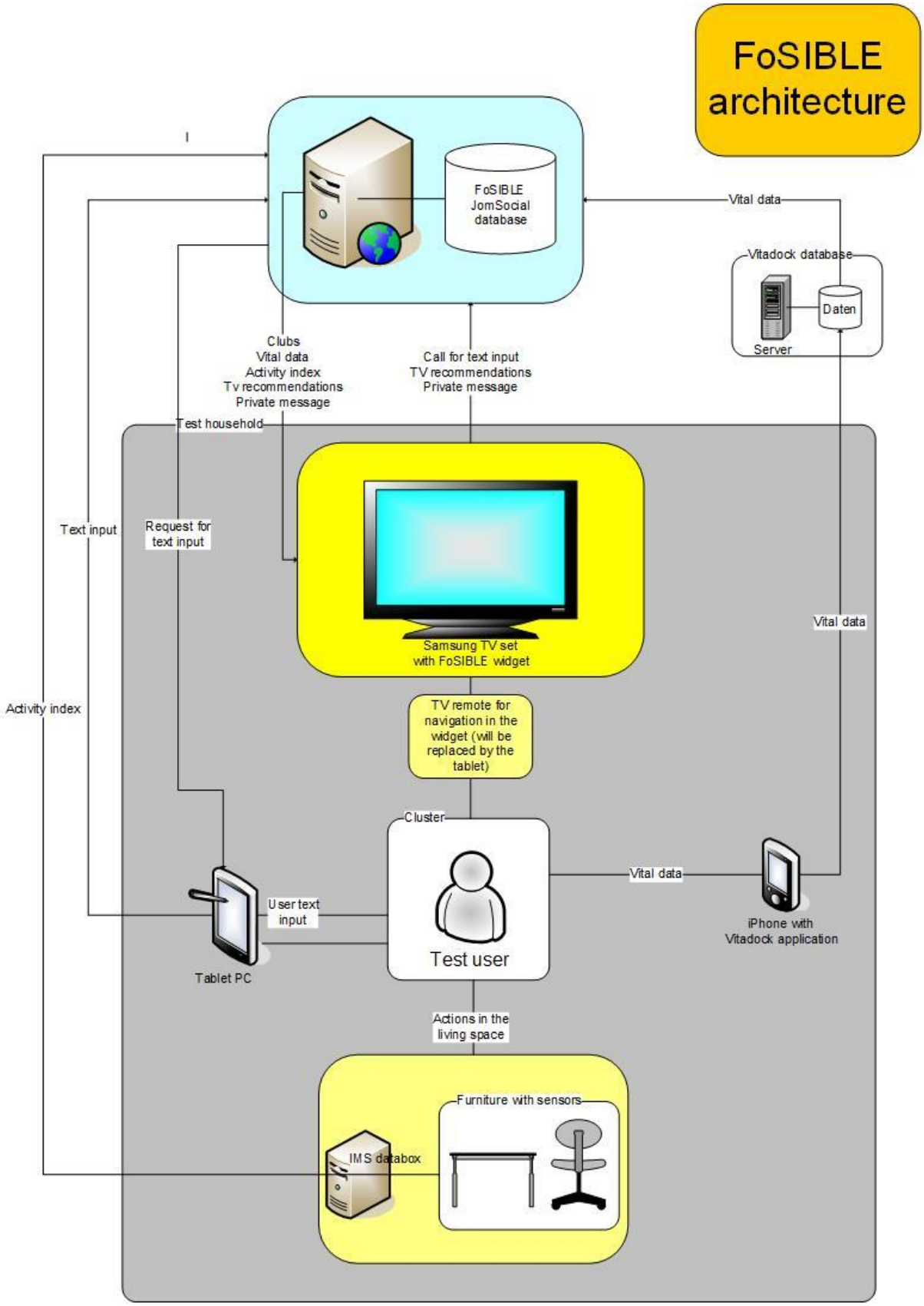

Figure 2. Overview of the system architecture

4. Timeline

The following timeline illustrates the development steps that were done for developing the before mentioned components und communication ways.

January 2011:

Online platform:

- installation of the online community software on test server
- configuration of the test installation

TV widget:

- TV sets ordered
- widget programming started by University of Siegen
- tests in the Samsung Emulator

February 2011

- finalization of the configuration of the online platform on the test server (Kaasa)
- transfer of the test installation to a live system (Kaasa)
- finalization of the interfaces needed for the project and the various partners (Kaasa)
- test of the interfaces (Kaasa)
- first test of communication between TV widget and the online platform (USI/Kaasa)

March 2011

- configuration of the online platform to match requirements of the project (Kaasa)
- adding data to the online platform (Kaasa/UDE/UTT)
- optimization of the TV widget (USI)

April 2011

- online platform finished in version v0.1 (Kaasa)
- TV widget finalized in version v0.1 (USI)
    - modules: Buddy list, Channel chat, TV recommendation
- first prototype for tablet application available in version v0.1 (AIT)

May - July 2011

- Tests with online platform and the TV widget (USI/Kaasa)
- fixing of bugs ((USI/Kaasa)
- adjustments to the interfaces (Kaasa/CURE/UTT)
- collecting feedback from partners

August 2011

<u>Online platform</u>

- adding new feeds for the club function of the TV widget (Kaasa)

<u>TV widget</u>

- purchase of the Acer tablet PCs (AIT)
- definition of interfaces and functionality for the Android application (AIT/Kaasa)
- chat functionality fully functions in tablet application in version v0.2 (AIT)
    - test of the functionality (AIT)
- adding clubs to the TV widget (Kaasa)
- code optimization and merging of the modules of the TV widget (Kaasa)
- finalization of version v0.2 of the TV widget (Kaasa)

September 2011

Online platform

- preparation of the access to the Vitadock database (Kaasa)
- connectivity tests (Kaasa)

TV widget

- adding the VitaDock function to the TV widget (Kaasa)
- UI re-design and optimization of button appearance (Kaasa/CURE/UTT)
- optimization of the behaviour when interacting with the tablet PC (Kaasa/AIT)
- optimization of the Android application (AIT)
- finalization of version v0.2.2 of the TV widget (Kaasa/USI)

November 2011

Online platform

- Activity Index function was added to the data broker (Kaasa)

TV widget

- logic for the Activity index was set up in Fraunhofer InHaus connecting the Mauser/IMS box to the online platform (Kaasa/Mauser/IMS)
- added the Activity Index function to the TV widget (Kaasa/Mauser/IMS)
- added the function to post all values to preconfigured clubs (Kaasa)
- optimization of club functionality (Kaasa/CURE/UTT)
- testing of the integrated widget and the attached components and sensors (Kaasa/USI/AIT/Mauser/IMS)
- finalization of the TV widget (v0.3) (Kaasa)
- further interaction integration in tablet application in version v0.3 (AIT)

April 2012 - January 2013

Online platform

Adjustments to the interfaces have been done to fit additional requirements of the interaction with the tablet application. This enables the user to control functions of the widget with buttons on the tablet. This eliminates the normal TV remote control in a lot of ways as its usage is not very user friendly, especially when the user tries to write a longer text (comment to a topic or a chat message).

As the platform itself is not part of the evaluation process in the test households, the UI has not been altered and the allocated resources for this task have been switched to further development of the widget itself.

TV widget

As the Activity Index will not be evaluated in the test households at UTT and USI, the widget had been altered to meet the special requirements in these households. The module with the Activity Index has been replaced by the module "Stay in Touch" that is meant to be a kind of a communication centre within the widget. Further explanations about the available functions in this module are described in deliverable D4.2.

Furthermore, the partners performed tests in the households and their labs with the widget and send the feedback to the Kaasa's bugtracking system. After each test the bugs have been prioritized and then fixed according to their importance for the project. Every fixed issue has been tested again by the partners. The testing included a lot of subversions. At the moment the current version of the widget is v 0.6.

Android tablet applications

During the first evaluations in the households the partners requested for a better method for text input as the usage of the normal remote control is very limited. On the one hand it is time consuming and on the other hand not suitable for users that are not familiar with text input with on a telephone. After deciding to use a tablet PC an Acer Tablet was chosen as medium and AIT started to develop the Android application.

In the first step the input of text for a book review and the possibility to answer to topics have been implemented. In later development stages other functions have been implemented. Now text input works for all parts where text can be added (TV recommendation, small text input etc.). Also the chat messages are now displayed and the listing of clubs can be show on the tablet.

The last step is to also emulate the coloured buttons of the TV remote control that are used for various interactions with the widget (e.g. open/close modules, navigation). Also it is planned to remove the TV remote control at all from the hardware setting and to use the tablet only as the remote control turned out to be the weakest point in the usability of the widget.

Additional information

During the first test in the labs and households the request was raised to remove the modules "VitaDock Vital Data" and the "Activity Index". This was done because the households are not equipped with the iOS hardware to be able to fetch the needed data. Also it is not possible to install the sensor equipped furniture in all households due to the prototype phase of the produced elements.

The modules have been removed from the widget and have been replaced by the "Stay in Touch" module that acts as a kind of message centre for the user.

For further research and evaluation of the Mauser furniture a special widget is available for the use in the InHaus where the furniture is installed.

5. Quality assurance

At every stage of the technical development, tests have been done by experienced software testers to find bugs and problems in the various components. The tests have been performed by the developing partners themselves and but also by partners that were not involved in the direct development to get more external feedback.

The partners CURE, UTT and USI checked each main development stage (see timeline) of the online platform and especially the TV widget to provide feedback about usability issues. Their feedback was based on the deliverables D2.1 (end-user requirements) and D2.2 (system functionality).

After finalization of version v0.3 of the TV widget it was tested as a whole in a test environment including two TV sets.

Further tests have been performed throughout the development in the test households and in the local labs of the universities. The feedback of the test has been used to improve the performance and the features of the widget and also the tablet application.

5.1 Performed tests

Chat function

User A writes a message to the chat.

Expected result: User B can read the text within a short period of time after sending.

The test was repeated various times. All tests have been positive.

TV recommendation

User A sends a TV recommendation to user B.

Expected results:

- The TV list shows the program of the current TV channel.
- The user can choose an item from the list and send the TV recommendation.
- User B gets the recommendation within a short period of time after sending.

All tests have been positive.

Clubs

Open a club, close clubs, reading the topics in a club, adding comments to a topic

Expected results:

- All entries are visible.
- The text input page opens on tablet and also a popup on the TV appears to inform the user to use the tablet.
- The text that is written on the tablet PC should appear after a short time after tapping the 'send' button.

All tests have been positive.

Vitadock Data®

Measurement tests with all available modules, posting the measurement in a preconfigured topic in the health club

Expected results:

- After a measurement and the storage to the online database the new value should appear on the screen after a short time.
- The value is readable in the activity club where other authorized users can read it.

All tests have been positive.

6. Conclusion

Choosing and installing the social media backend was the first step in the technical implementation of the FoSIBLE project. The main goal was an open architecture and the possibility to add new sensors and other inputs at a later stage of the project without being forced to change the whole system. It was important that open standards are supported and no proprietary protocols are used by any part of the backend software so that communication between the elements can be easily implemented. As research in the test households is an on-going task the state of the widget is not final. New findings are expressed in feature requests to make the widget meet the requirements of the users in the test households (e.g. module 'Stay in Touch').

In the longer term tests in the households it turned out that the performance of the TV set is very limited due to the lack of memory and CPU power in the available 2010 and 2011 TV sets. Because of this, some major changes had to be done in the code of the widget to improve the stability and to increase the user experience with the FoSIBLE system.

Some tasks are still pending and the evaluation in the households will produce additional feedback about parts of the widget that still need improvement.

It is planned to implement some open feature requests to improve the usability of the widget and later, once the widget is installed in all households and the evaluation is starting, to implement additional requests that were identified in the evaluation.

After the start of the evaluation in the households the feedback that is collected from the users will be used to do changes in the widget according to real end user requirements. Also bugs that will occur will be fixed so that the users can use the widget without problems.