



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

***D4.2: Early and advanced, stable prototypes of software applications
(generational community application, gaming and entertainment application,
communication tools)***



Responsible

Kaasa solution

USI

AIT

Deliverable

D4.2 – Early and advanced, stable prototypes of software applications (generational community application, gaming and entertainment application, communication tools)

Version: Final

Date: 28/02/2013

Dissemination level: (PU, PP, RE, CO): PU

Project: FoSIBLE

D4.2 Early and advanced, stable prototypes of software applications (generational community application, gaming and entertainment application, communication tools) - 28/02/2013

Abstract

In this document the previous development steps regarding the FoSIBLE platform were described including the internet capable TV set and the TV widget. Furthermore, the progress regarding the tablet implementation were explained.

Table of Content

1.	Introduction	4
1.1	Purpose of the Document.....	4
1.2	Document Scope	4
1.3	Partners.....	5
2.	Management Summary	6
2.1	Internet capable TV set and TV widget.....	6
2.2	TV widget modules and functions	7
2.2.1	Login to the community platform.....	8
2.2.2	TV channel recognition	9
2.2.3	Buddy list	12
2.2.4	Chat.....	12
2.2.5	TV recommendation	14
2.2.6	Private message.....	15
2.2.7	Clubs	15
2.2.8	Medisana VitaDock® data (Kaasa).....	18
2.2.9	Activity index (Mauser/Fraunhofer IMS).....	20
2.2.10	Stay in Touch.....	22
2.3	TV widget User interface and navigation.....	23
2.4	Tablet Application (optimized for Archos Tablet).....	24
2.4.1	User Authentication.....	24
2.4.2	General Aspects.....	25
2.4.3	Chat.....	26
2.4.4	Clubs	28
2.4.5	Private Message.....	29
2.4.6	TV Suggestion	30
2.4.7	Preferences.....	31
3.	Quality assurance.....	32
3.1	Timeline.....	32
3.2	Performed tests	35
4.	Conclusion.....	37

1. Introduction

1.1 Purpose of the Document

The purpose of this summary is to report on the technical achievements in the FoSIBLE project for deliverable D4.2.

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 the document.

Kaasa solution has contributed information about:

- social platform based on an Open Source Community software system (JomSocial)
- integrated TV widget for Samsung TVs
- interfaces to attach various sensors in the future

AIT has contributed information about:

- Android application (optimized for Acer Tablet PC) used for text input in the TV widget

University of Siegen has contributed information about:

- integrated TV widget for Samsung TVs
- modules of the code of integrated widget (TV recommendations/chat)

Mauser/Fraunhofer IMS have contributed information about:

- Sensors and sensor data that are interpreted by the technical platform and can then be presented in the integrated widget or the social platform
- furniture with integrated sensors

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

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

2. Management Summary

This document contains a description of deliverable D4.2 of the FoSIBLE project - early and advanced, stable prototypes of the software applications.

The technical system consists of various components:

- Internet capable TV set
- TV widget
- Tablet PC with Android OS
- Sensors included in furniture

These components and their use are described in detail.

2.1 Internet capable TV set and TV widget

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.

The SDK and other information about the Samsung Smart TV can be found at:

<http://www.samsungdforum.com/>

The main layout structure of the modules that needed to be implemented for evaluating the widget was done simultaneously with the research and first tests with the Samsung SDK. After the setup of the development environment, developing the first main layout focused on parts of the mockups like the buddy list and recent activities as well as the main navigation status bar.

As Samsung did not provide any working functionality in Europe to display content and live-TV information at the same time, we had to consider this in the second main layout version. Because of this was integrated live TV information into the widget and introduced a hotkey to quickly switch between pure live TV and content overlaying live TV. At this time, the main layout has been finished.

For the development the actual stable build of the SDK is used (at the moment version 2.5). The widget and also the version of the Samsung SDK that includes the emulator changed a couple of times and so did the Samsung TV models. Newer models have more available features but the FoSIBLE widget has to be programmed in a way that this new functions are

not used but can be added at a later stage, as not all households will have these new models available.

When the development of the widget started, the 2010 models by Samsung were on the market and the SDK that Samsung offered for download was optimized for this models. The 2011 models that came on the market were better equipped with memory and CPU power. The SDK itself did not change really. There were just some slight changes in the SDK that had no effect on the first version of the widget. The 2010 and 2011 models are compatible.

The latest version of the SDK that also supports the actual 2012 TV models is already available. The approach to program a widget for these new TV sets changed drastically. For older versions the TV widget has been programmed as a kind of webpage that offered different functions. These functions had to be executed on a backend server or had to be programmed for each module. The new SDK now allows programming the widget as a standalone app where all needed functions are implemented. This approach is very resource friendly while the older models struggle with the lack of sufficient hardware memory and CPU power.

Unfortunately, this new method cannot be used for the TV sets that are used in the FoSIBLE evaluation because the 2010 and 2011 TV models do not support this SDK. For a later commercial usage of the FoSIBLE system it would make sense to reprogram the whole widget in order to benefit of the new possibilities of the new Samsung SDK.

2.2 TV widget modules and functions

The development of the widget started in February 2011, whereby the first parts of the widget were programmed and tested in the emulator only. After the TV sets arrived, the software could also be tested on the TV sets. The widget is fully internationalized and offers the user interface either in English, French or German, depending on the system setting of the Samsung TV set.

The first modules that have been worked on were:

- Login to the community platform
- TV channel recognition
- Buddy list
- Private messages
- TV recommendations
- Chat
- Clubs

- Medisana VitaDock® data
- Activity index
- Stay in Touch

In the following, these components were described in detail.

2.2.1 Login to the community platform

The user identify management for the widget had to be developed first as this module is the basis for features of the TV widget like the buddy list and the channel chat.

User and password management are executed on the FoSIBLE platform. Once a user has logged in successfully (widget uses an API of the FoSIBLE platform server backend), a user id is generated and stored inside the widget during its lifetime. This id is send to the server in each call that the widget sends toward it.

Figure 1 shows a rough scheme of the FoSIBLE system. The yellow area represents the household with the TV system on which the widget is installed. After the login to the FoSIBLE backend, the user has access to the services and the data that are available to the FoSIBLE system (e.g. clubs, vital data etc). The same information is also available via a web browser, what is just an alternative way to access the FoSIBLE system (the web platform is not focused in the evaluation in the households).

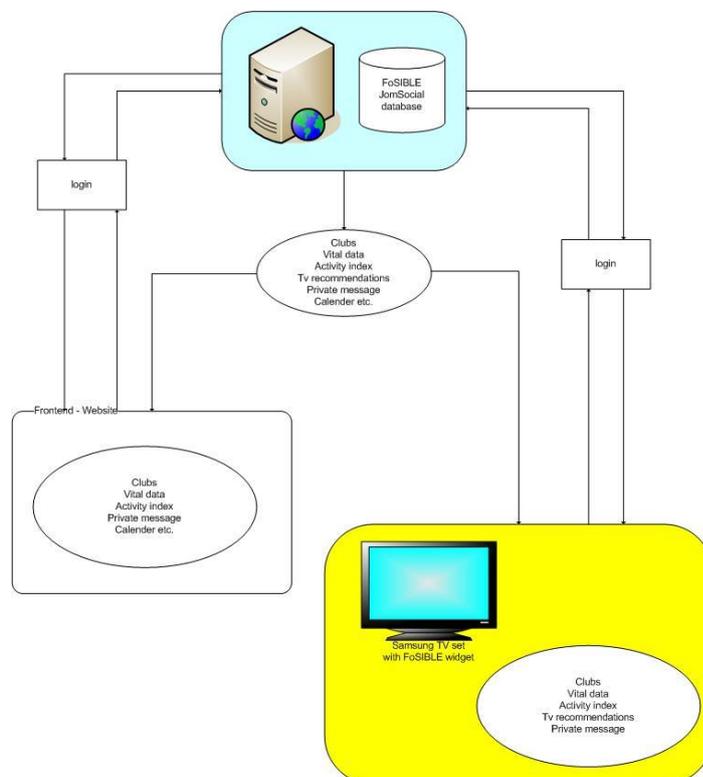


Figure 1. Rough structure of the FoSIBLE system

The login procedure uses the same technique and addresses the same interface as the web version of the community platform.



Figure 2. Login window

2.2.2 TV channel recognition

The TV widget is also able to determine the channel that the user is currently watching via standard interfaces that Samsung provides with the SDK.

The widget is programmed to fetch the channel and to post this information to the data broker (explained in D4.1) where it is stored and used to show this information in the buddy lists of the user's friends.

At the same time the widget also retrieves the TV program information from the website www.tvprogramm24.com and stores the available TV shows in the database of the backend system.

The information is parsed and added to the database through a php site that is addressed by the widget. The data fetched by the system is then used for TV recommendations (see 2.2.5) that are sent through the widget to other users.

Figure 3 shows the implementation of the channel recognition in the FoSIBLE system. TV set A collects the channel information and sends it together with the user information to the FoSIBLE backend. The information that is then stored in the database is fetched by other users and the widget on their TV set so that they know what channel one of their friends is watching.

Project: FoSIBLE

D4.2 Early and advanced, stable prototypes of software applications (generational community application, gaming and entertainment application, communication tools) - 28/02/2013

The information about the current active program is also used by the TV widget itself to retrieve the according TV program directly from www.tvprogram24.com that offers this service for free in the internet. The information gathered from that site is used to send TV recommendations to friends or to set TV reminders for oneself (see 2.1.5).

2.2.3 Buddy list

The Buddy List enables the users to see who of their added friends are online and what TV channel and program they are currently watching. The information about the online status of friends is provided by the backend via RSS feed that is fetched by the widget module and then shows the results about who is online.

The user in front of the internet TV set can select one of his or her friends on the Buddy List with the help of the TV remote control. In this case, a popup window would appear in the widget and the user can choose if he wants to write a private message, send a TV recommendation or remove the friend from his list (see Figure 4).



Figure 4. Buttons of the buddy list interface to choose an action¹

There are two possibilities to add friends to the Buddy List. One is to add the friends via the online backend website of the FoSIBLE project. After login it is possible to search for other users that are registered in the platform and add them to the user's own buddy list. All users who are added as friends on the website also appear in the Buddy List of the widget.

The other way to add a friend is to add him or her directly in the widget. This happens in the module "Stay in Touch". In this module, the user can search for a friend and then add him or her to his own list. For this process it helps to know the full name of the friend because in this case the search function of the widget can be used. Also when the user just knows parts of the name it is also possible to search for someone (e.g. the search for ,ste' would list all names containing this string like Stefano, Asterix etc).

2.2.4 Chat

Another function of the widget is the channel chat that enables the users to step in direct contact with one or more persons that are watching the same TV channel at the same time. In this way they can discuss about the program or just chat about different topics while watching TV. The chat window can be opened with the red button of the remote control.

The interaction between two and more users in the FoSIBLE network requires a central communication server because peer to peer connections are not possible on HBB TV at this state of time. Additionally, peer to peer connection would introduce some problems like

¹ "Nachricht senden" = "Send message", "Programm empfehlen" = "Recommend TV show", "Als Kontakt entfernen" = "Remove as contact"

firewall restrictions. This can easily be avoided with a central communication server which is based on an Apache HTTP server running PHP and MySQL. In the FoSIBLE project the JomSocial community platform serves as communication server which matches all these requirements.

First we developed a PHP class which is used for database communication. After that we determined that the communication between the widget and the server will be done in JSON, as both PHP and javascript (which is used in HBB TV) are able to encode and decode JSON data.

After setting up the development server we were able to start coding the server backend files for the channel chat. Therefore, a SQL table within the community database has been setup which stores the messages which users will send to a channel room.

For typing text into textboxes inside the widget, the text input interface IME provided by Samsung has been integrated. It is required for using both the remote control and an optional android app for text input. The result of these investigations was to use the ticker as it gives the possibility to switch channels while the widget is running. In full mode this is not possible.

However, the input via the TV remote is not user-friendly and takes too much time even for very short texts. To solve this problem the tablet PC will be used as an alternative input device for all text input related points in the TV widget (see 2.4). It has been added to the project at a later stage of the development.

In the current widget it is only possible to chat with people who are watching the same TV channel as the user. In a later stage it is planned to add a channel picker so the user can also chat with people in other channels. This option is already implemented in the Android application (see 2.4.3). The information about the channels is obtained from the online database.

To reduce the load on the widget backend server for the channel chat, we developed a two-step setup: (1) the first call is sent as soon as the user opens the channel chat inside the widget for the first time and then receives the reply of the backend server with the latest channel chat messages. (2) The second step is a regular AJAX call that is sent every five seconds towards the backend and which only calls for newest messages depending on the submitted last received message id.

Once the text is written and confirmed, the user sends the message to the previously chosen friend.

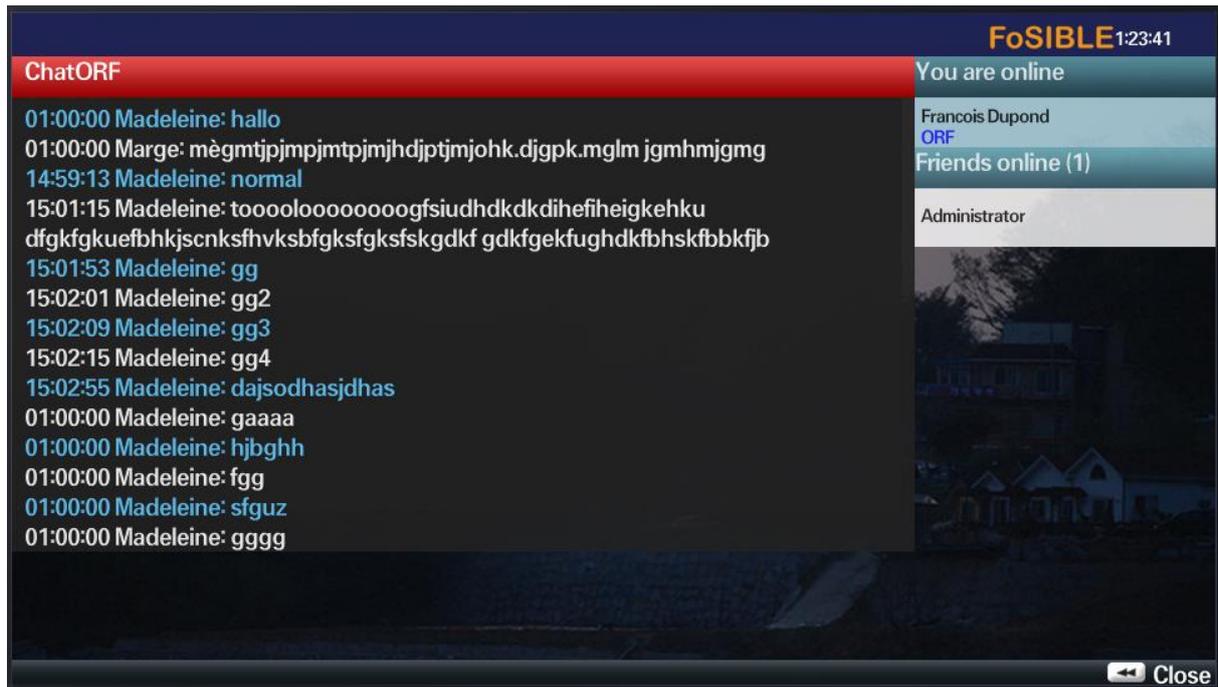


Figure 5. Screenshot of the channel chat

2.2.5 TV recommendation

The users have also the possibility to suggest a TV show to a friend of his Buddy List or to use this function as reminder functionality. In the first case, a popup window appears in the widget of the friend which shows the recommendation.

A TV recommendation can be send by selecting a friend from the Buddy List while watching TV. After selecting a popup appears (see Figure 4) and the user can choose the button TV recommendation. Then the program of the channel the user is currently watching is presented (Figure 6), whereupon the TV show can select that he wants to recommend.



Figure 6. Interface of the TV guide

The basis for a TV guide's proper implementation is the adequate implementation of getting EPG data. So our first step was to define an adequate method of acquiring those data. The Samsung SDK does provide access to the EPG data in the 2011 models, but not in 2010 models. As we do not have the chance to test this functionality on 2011 models, we decided to scrap the EPG data from one of the various TV guide websites. We always kept in mind that the source should be changeable, so the scrap routine consists of only one static function which is easily replaceable. This routine is then integrated in a much more complex php class which determines automatically if requested information has to be scraped first or an answer from the database to the client is possible. This has been done to save resources on the server backend and to reply information faster to the client.

After the scraper, the complex php class and the database structure have been built. The next step was designing and developing the user interface for the TV guide. The focus laid on the navigation part, as the whole navigation inside the TV guide has to be possible with left/right and up/down, keeping in mind that a future navigation may be a gestured one with external hardware devices.

At least we implemented the functionality with which clients are able to suggest a TV show to a friend or to use the function as reminder functionality. If they accept the suggestion, a popup will appear 15 minutes before the show goes on air.

2.2.6 Private message

Besides the communication in a channel chat, the users can also write private messages to friends. This is possible by selecting a friend from the Buddy List whereby a popup appears in which the user can select the button "send private message" (see 2.2.3). The friend will be informed about the message by a popup similar to the one that shows up when a TV suggestion is made.

While programming these modules we were able to use existing code elements as most code which was necessary for writing and receiving private messages between two FoSIBLE users could be reused from the TV programme suggestion feature (see 2.2.5).

2.2.7 Clubs

In the FoSIBLE community it is also possible to join clubs where various topics can be discussed with other participants of that club. Joining and creating clubs is already possible in a browser version. It is planned to be expanded to the TV widget as well.

All clubs in which the user is a member of are visible in the TV widget when the user opens the module 'Clubs' (with the yellow button on the remote control). The user can then scroll through the list with the arrow buttons on the TV remote (Figure 7) and can select a club with the appropriate button on the TV remote to see the single topics inside a club. Other clubs are not shown in the user's interface. The content of the clubs is offered to the widget with the help of RSS feeds that are polled by the widget periodically.

The structure in the widget is as follows:

| - Name of the club

| --- Title of the topic

| --- answers

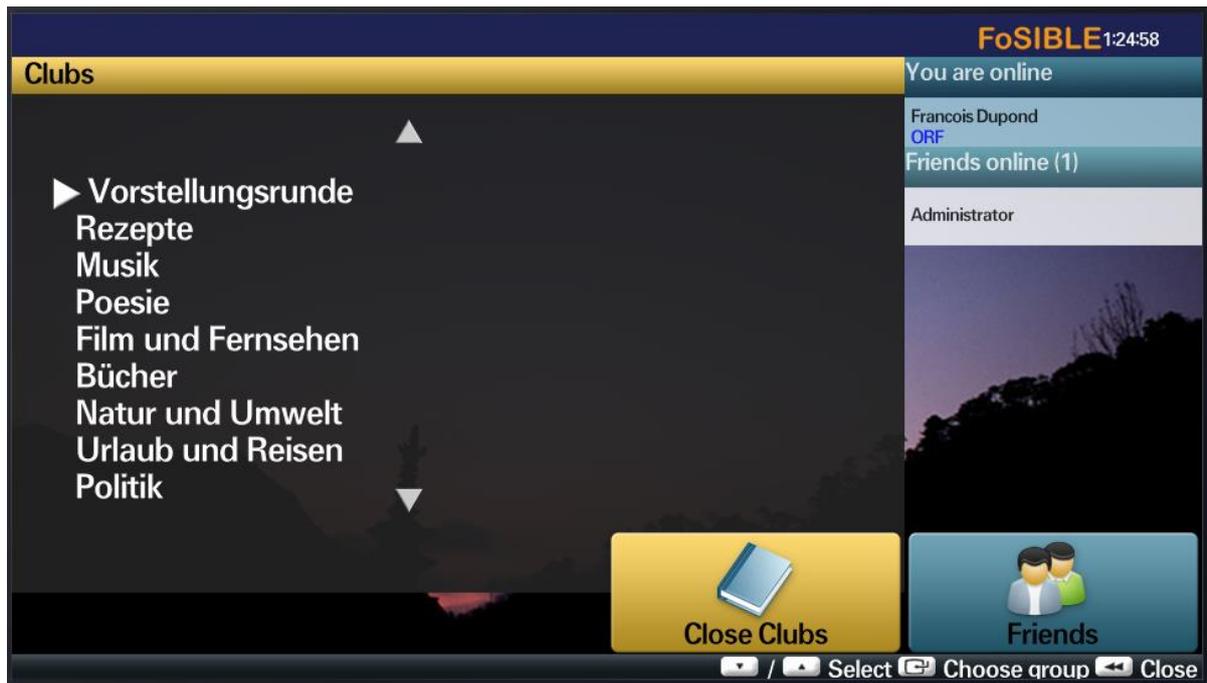


Figure 7. Screenshot of the interface showing the list of the user's clubs²

With the help of the TV remote control the user can scroll through the clubs and can read the topics and also the comments that other participants wrote. Pressing a preconfigured button on the TV remote control triggers a signal that is sent to the data broker. In this case the Android application gets this information via a RSS feed and an interface opens to enable the user to input the desired text into a textbox using the android software keyboard.

² Translation: Vorstellungsrunde (Introduce yourself), Repezpte (recipies), Musik (music), Poesie (poems), Film und Fernsehen (movies and television), Bücher (books), Natur und Umwelt (Nature and Environment), Urlaub und Reisen (holidays and travel), Politik (politics)



Figure 8. Screenshot of the interface showing one selected club



Figure 9. Screenshot of the interface showing some posts in the selected club

2.2.8 Medisana VitaDock® data (Kaasa)

VitaDock® is Medisana's new product revolution and consists of a free application as well as the appropriate measuring modules for reading and managing the user's vital data together with his iPhone®, iPod touch® or iPad®.

Currently Medisana is working together with a big German telecommunication company on an online database where the users can store their vital data that are fetched from the VitaDock modules. It is possible then to show the vital data in the TV widget and the community platform as well as to post the latest results to a club that only a certain group of people have access to so that private data is being kept secret.

This is all processed on the data broker and the backend. The TV widget acts as an additional output source.

VitaDock regulates the important vital functions in one app with the following modules:

GlucoDock® - automatic blood sugar measurements and diary



Figure 10. Screenshot GlucoDock® integration in the FoSIBLE interface

Project: FoSIBLE

D4.2 Early and advanced, stable prototypes of software applications (generational community application, gaming and entertainment application, communication tools) - 28/02/2013

CardioDock® - blood pressure measurements and self-management



Figure 11. Screenshot CardioDock® integration in the FoSIBLE interface

ThermoDock® - fever measurement without contact via infrared sensor



Figure 12. Screenshot ThermoDock® integration in the FoSIBLE interface

TargetScale® - body analysis scale



Figure 13. Screenshot TargetDock® integration in the FoSIBLE interface

The VitaDock App is available for free in the iTunes AppStore and will soon also be available for Android devices. The software and the hardware components are CE certified according to the German medical product law (“Medizinproduktgesetz”). All current and also older iOS versions are supported by the application. All new development stages are tested and test protocols are available as part of the certification.

NOTE:

As the test households in the project do not have the VitaDock sensors available, these modules are not needed in the Living Lab evaluations and have been deactivated for the evaluation version of the widget. Nevertheless, it can be reactivated at any later time when it is needed. The available interfaces are ready and can be addressed by the sensors.

2.2.9 Activity index (Mauser/Fraunhofer IMS)

The Activity Index is an index that shows the activity of a user in a household in the form of an abstract index. A low index means that the user is not very active inside his household what in the case of old people could mean that there is a problem.

The index is created from sensor data in the Fraunhofer InHaus in Duisburg. From all the movements and actions within a sensor equipped household the IMS databox (a little industry PC) is creating the index and sends the data to the data broker.

Every time this value changes to any direction, the value is sent to the online platform and is stored for each user in the database. This value can be posted to an activity club that is accessible only for a preconfigured group of people.

Detailed information about the sensor integration in furniture can be found in deliverable D3.3.



Figure 14. Screenshot Activity Index integration in the FoSIBLE interface

NOTE:

As it is not possible to equip all households that take part in the evaluation with furniture and the according sensors, the module has been deactivated for the widget that will be used by UTT in France and USI in Germany. This module was replaced by the 'Stay in Touch' module.

Nevertheless, as all interfaces are still available, it could be reactivated at any later time (e.g. in a commercial version of the widget). For the evaluation different versions of the widget are available according to the needs and the situation in the evaluation places.

2.2.10 Stay in Touch

This new module has been added to the widget to replace the modules 'Activity Index' and 'Medisana VitaDock' as they are not needed for the evaluation in the test households at UTT and USI. It is meant to be a kind of message centre where the user can have a look at the messages he received, can search for other users, add them to their Buddy List etc.

Adding friends to the buddy list

In the 'Stay in Touch' module it is possible to add a new friend to the buddy list. It can be opened by pressing the green button on the remote control. Then, by choosing the 'Add a friend' button, the user can enter a name of a friend he wants to add. After confirmation of the search string the database in the backend is checked and gives back all results matching the searched name. When there is just one user with the searched name, the name is added to the form and the user can confirm that he wants to add this user to the Buddy List.

When the search gives back multiple results a list is sent to the widget and the user can then search for the correct user.

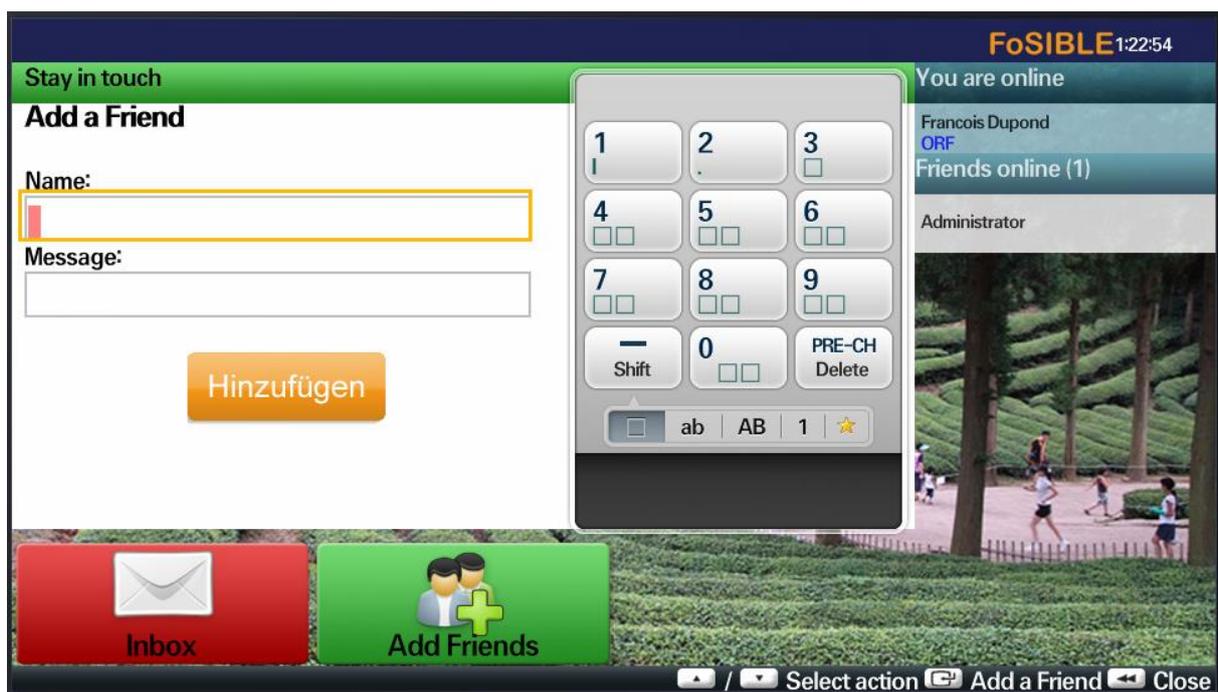


Figure 15: Add a friend interface

Inbox

Another feature inside the 'Stay in Touch' module is the inbox. It acts like an inbox of a mail programme. All messages that the user receives are listed here and the user can then select the message he wants to read.

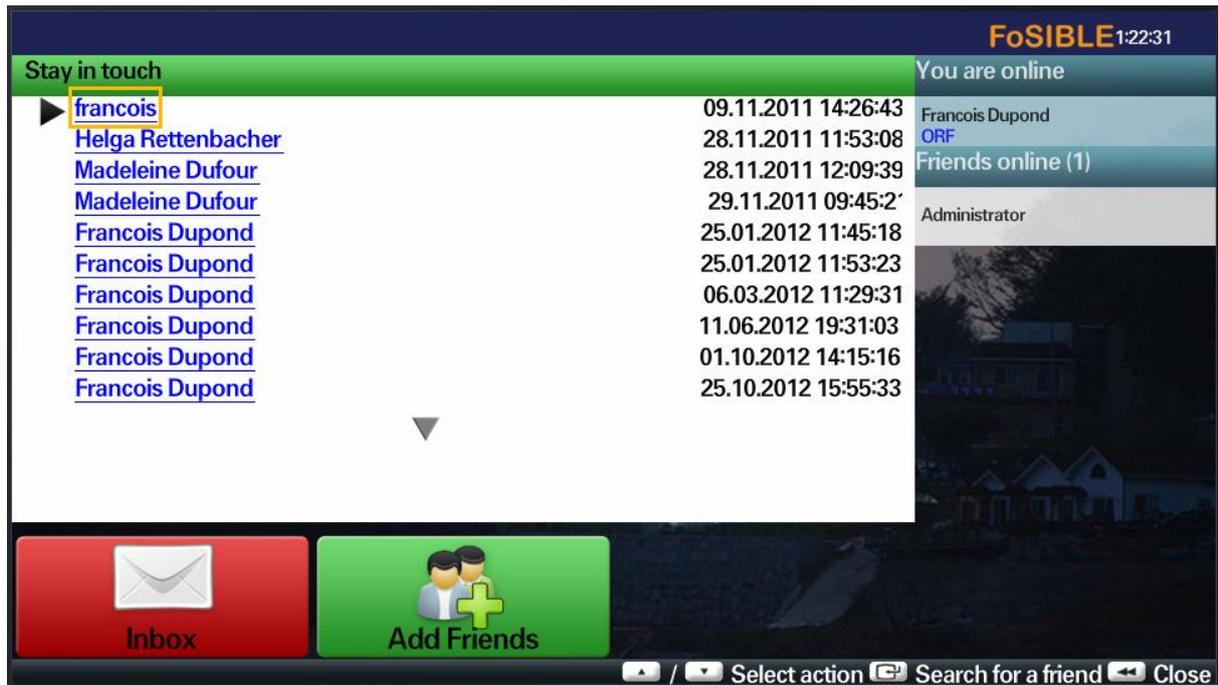


Figure 16: Messages inside the inbox.

2.3 TV widget User interface and navigation

All UI and navigation topics have been implemented in direct collaboration with the partner CURE who is very experienced in usability questions.

The interface changed quite a few times during the development process of the widget.

In version v0.1 the buttons were still placeholders and always visible. In version v0.2 the buttons that had no function on a certain page have been removed so that the user is not confused. In version v0.3 the buttons have been polished and the whole UI was redone from the graphical point of view.

The TV widget is designed in a way that just the four directions up, down, left and right and the ok and back buttons are used to navigate through the modules. Action buttons are the standard coloured buttons on the TV remote.

As the usage of the TV remote control is very time consuming when entering longer texts and also the switch between the TV remote and the tablet turned out to be confusing, it has been decided to add the major functions of the widget to the tablet application. The modules can now be opened by pressing the coloured buttons on the TV remote or alternatively by tapping the corresponding buttons of the tablet application (see details in 2.4)

2.4 Tablet Application (optimized for Archos Tablet)

This section describes the basic operations of the FoSIBLE tablet application and its interactions with the server backend featuring the gateway to the TV widget. The application is primarily used as a secondary user interface to the interfaces of the TV widget giving the user the opportunity to interact with the FoSIBLE framework more directly.

The tablet application features several aspects which are related to the TV widget application which are described in detail in the following sections.

The tablet application is fully internationalized and offers the user interface either in English, French or German, depending on the system setting of the Android tablet.

2.4.1 User Authentication

Since the application handles personal data which is associated with a certain user, the user has to authenticate him against the system with a username / password combination. The user can enter his credentials in respective user interface fields using the onscreen touch keyboard as depicted in the following figure.



Figure 17. User Interface for authentication

To ease the process of authentication, the user is given the opportunity to persist his credentials on the tablet and is automatically logged in to the system the next time the tablet application is started. This process is delayed by 5 seconds, to give the user the option to sign in with a different username / password combination. The remaining delay until the login process is triggered is visualized directly on the respective button of the user interface.

Following state of the art user interface guidelines, the user is presented with a descriptive message in case of an error during authentication. Respective user interface fields, which

could be the cause of the error, are marked so the user is directed on how to resolve the issue as depicted in the following figure.



Figure 18. Error handling on authentication issues

2.4.2 General Aspects

The tablet application is designed for two main purposes. First it can be used to control the TV and the respective widget by using it as an additional remote control device. The second main purpose is to ease the input from the user by presenting another view for the specific features of the widget. The tablet is closer to the user's eye and features one step actions to be sent to the widget reducing view point switches by the user from the TV to the keyboard back and forth.

To mimic the remote control feature the tablet application presents the four coloured buttons, known from the TV remote control, across the application in the upper region of the user interface. These buttons can be used to select the respective features from the TV widget.

The following figure (Figure 19) shows the main menu which is presented right after the user logs in to the system with a easy to use button style navigation to access the main features and the coloured button bar on top of the screen.

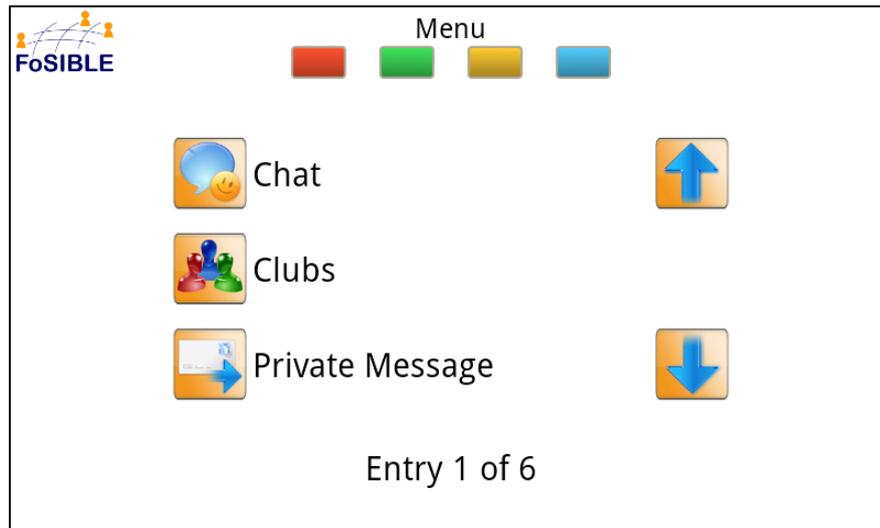


Figure 19. The main menu featuring button style navigation

To help the user navigate within the application there is primarily only one level to go away from the main menu to access the main features of the application. The user can always go back to the main menu by pressing a button in the upper right corner of each user interface screen. Each user interface screen presents the name of the section clearly on top of the screen.

The main menu offers an easy to use vertical navigation system, whereas the number of possibilities within the menu is presented in the lower section of the screen.

2.4.3 Chat

The application offers the logged in user the possibility to send messages to all other users viewing the same TV channel at the same time. To access this section of the application the user can simply press the button “Chat” from the main menu.

The user interface of the chat section presents the user a list of available TV channels, whereas the currently selected channel on the TV is also selected within the tablet application.

On the right side of the user interface, the user is presented with a list of recent messages that has been sent for this TV channel. The following figure (Figure 20) exemplifies the chat feature with the selected TV channel “ARD”.

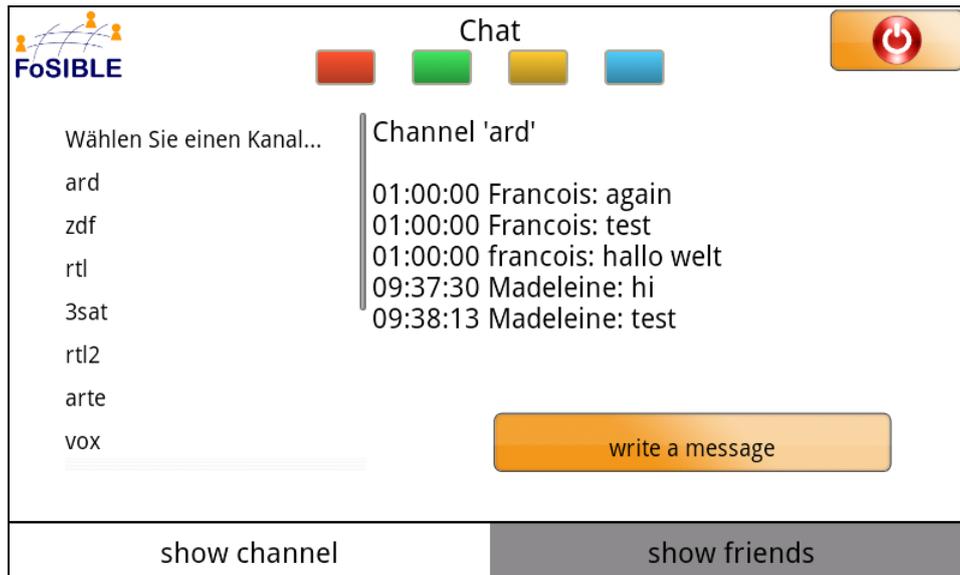


Figure 20. The TV chat section of the application

To switch the channel, the user can select one from the list on the left side of the user interface. To prevent unwanted channel switches by accidentally touching a channel name in the list, a confirmation dialog is presented to the user in such an event as Figure 21 demonstrates.

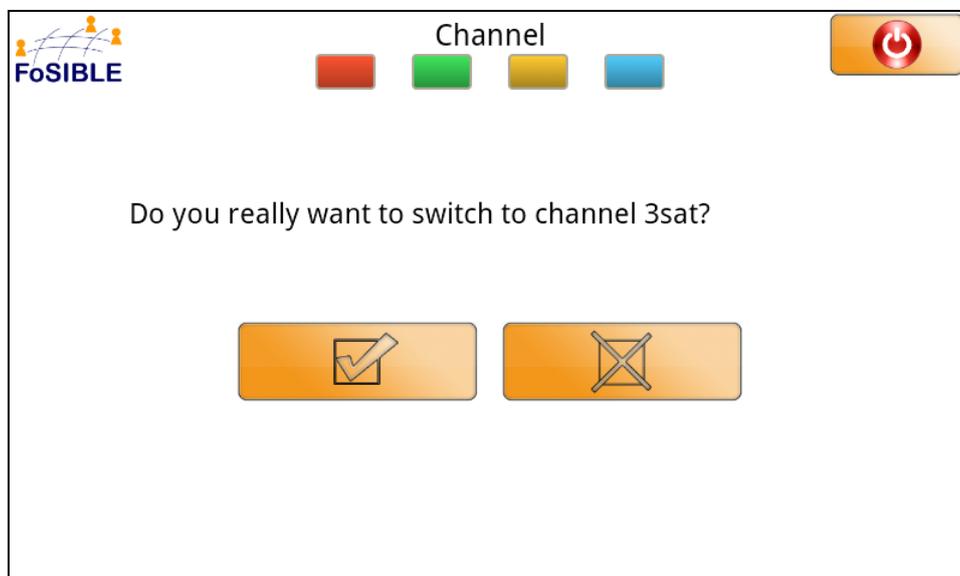


Figure 21. The confirmation dialog to prevent unwanted channel switching

To write a message to all users logged in the chat section and selected the same TV channel, the user simply selects the respective button in the user interface. He is then presented with a user interface field enabling him to compose the message (Figure 22).

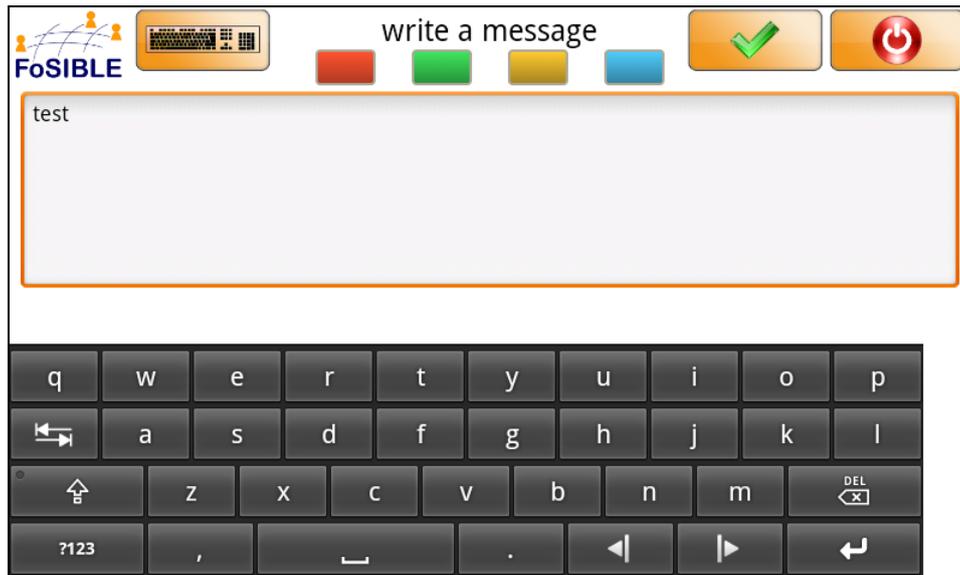


Figure 22. Composing the message within the chat section

After the message is composed, it can be sent by touching the well-known button in the upper right corner of the user interface. This leads to the transmission of the message to the server and a visual feedback to the user informs him about the progress.

2.4.4 Clubs

By selecting the respective button in the main menu, the user can access the “clubs” feature of the application featuring a horizontally scrollable list of all available threads on the server.



Figure 23. The clubs section of the application

By selecting the respective button, the user can enter the club, whereas a list of already composed messages within the club is shown. A new message can be composed by selecting the respective button and entering the needed information within the user interface.

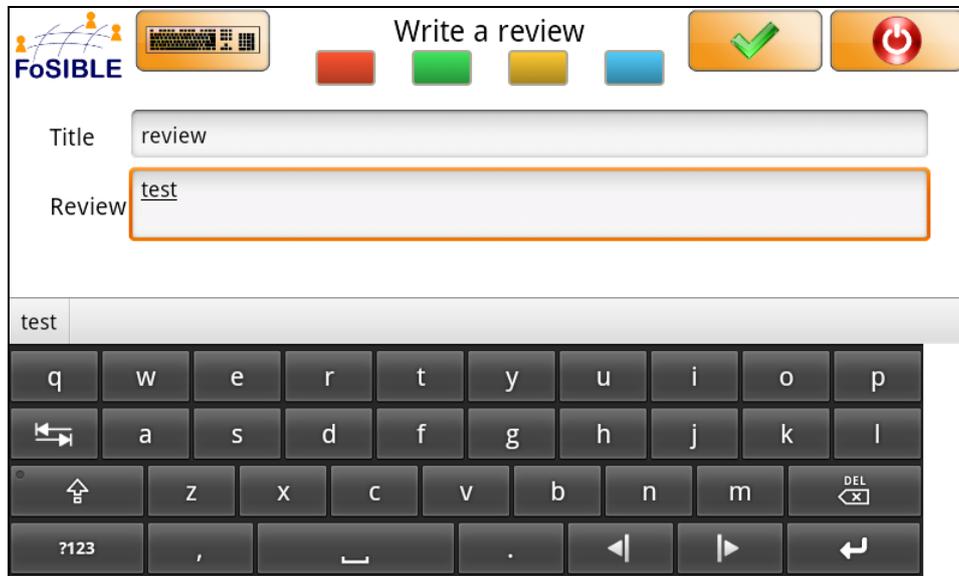


Figure 24. Composing an entry for the club section

After the user transmits the message by touching the well-known button in the upper right corner of the interface, the user is informed about the progress.

2.4.5 Private Message

A logged in user has the ability to send private messages to another user. When this is received by the other user it is displayed on the big screen. To be able to send a private message to another user, the user has to be added first to the users list of friends. This on the one hand limits the choices of available users to the relevant ones and on the other hand addresses privacy issues. It is not possible to spam a user with messages or send unsolicited messages to specific users.

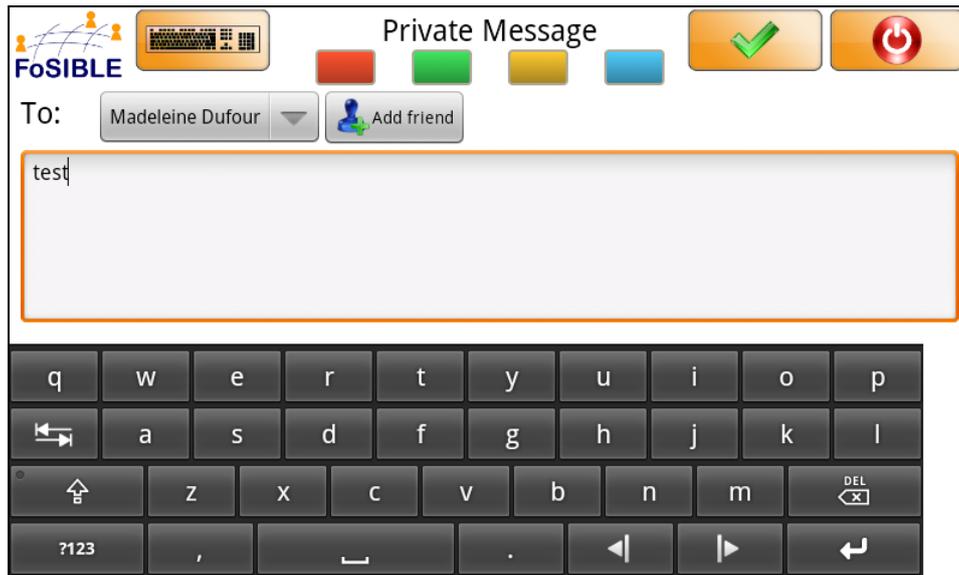


Figure 25. The user interface to compose a private message to a friend

To add a certain user to your list of friends, the user has to be added via a search feature. This feature can be accessed by selecting the respective button. This leads to a screen where all available users are listed. To ease the process of selecting a specific person, the user can limit the number of results by entering some letters from the intended person's name.

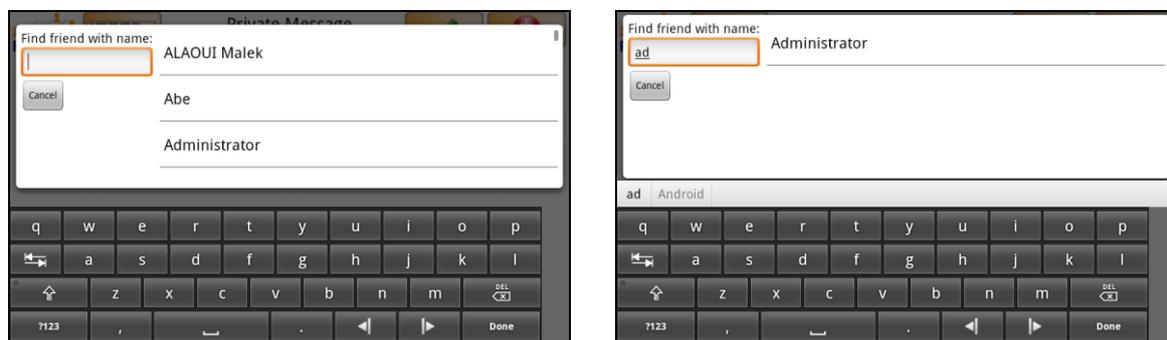


Figure 26. Searching for new friends - filter for results on the right side

The last figure (Figure 26) shows the unfiltered list of users within the system who can be added to the user's list of friends as well as the search for a specific user by entering a part of the person's name.

2.4.6 TV Suggestion

The system offers the possibility to recommend a specific TV show to a friend. To access this feature, the user can enter the TV suggestion part of the tablet application by touching the respective button in the main menu.

After entering this section of the tablet application, the user can select a TV channel on which the show is scheduled, the actual show he likes to recommend and the friend to whom the recommendation is sent. Additionally an extra message to personalize the

recommendation can be added and finally the recommendation can be sent by pressing the respective button in the upper right corner.

The following figure exemplifies a completely filled TV show recommendation.



Figure 27. User Interface for recommending a specific TV program to a friend

Following state of the art user interface guidelines, the user is guided through the process of sending a recommendation by enabling the relevant user interface items as needed. For example, the drop down menu for selecting the show is only enabled after it has been populated with concrete values after selecting the desired TV channel. The button for sending the recommendation is only enabled after all the required information is present, otherwise it is disabled and visually greyed out.

Worth to note is also that the recommendation can only be sent to a user's friend. Hence the list of possible recipients is limited and easier to navigate through.

2.4.7 Preferences

Within this section of the application the user is presented with the opportunity to define some default values which are used in functionality previously described.

The most important ones are the extended user credentials containing the username and password combination used within the auto login functionality as well as the default language setting which is sent to the TV widget so all the devices in the system are showing text messages to the user in a common language.

Within the preference screen the delay before the auto login process is triggered can be configured. This is helpful for the situation where a user commonly switches the user of one tablet but needs more time to interact with the system than the default of 5 seconds.

3. Quality assurance

As described in D4.1, tests have been done at every stage of the technical development by experienced software testers to find bugs and problems in the various components. The tests have been performed by the developing partners themselves and also by partners that were not involved in the direct development to get more external feedback.

The partner CURE checked each main development stage of the online platform and especially the TV widget to provide feedback about usability issues.

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

3.1 Timeline

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 Uni Siegen
- tests in 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 need 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)
- configuration of the online platform to match requirements of the project (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 2012

- online platform finished in version v0.1 (Kaasa)
- TV widget finalized in version v0.1 (USI)
 - o 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

Online platform

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

TV widget

- 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)

Project: FoSIBLE

D4.2 Early and advanced, stable prototypes of software applications (generational community application, gaming and entertainment application, communication tools) - 28/02/2013

- 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 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

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

TV widget

- logic for the Activity Index was set up in Fraunhofer TestLab
- connecting the Mauser/IMS box to the online platform
- 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
- further interaction integration in tablet application in version v0.3 (AIT)

December 2011- January 2013

- further implementation of the tablet into the system in order to improve the usability for the user and to exclude the TV remote control from the test
- customization of the widget for the usage in the different test environments (Test Lab, InHaus, UTT, USI)
- adding features after evaluation with the users to improve the usage and the functionality of the widget
- change of features to improve the usability for the end users
- bugfixing after feedback from the testing partners

3.2 Performed tests

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 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).

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.

Chat function

User A writes a message to the chat.

Project: FoSIBLE

D4.2 Early and advanced, stable prototypes of software applications (generational community application, gaming and entertainment application, communication tools) - 28/02/2013

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

Opening 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 on 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.

4. Conclusion

After the main community platform has been installed and configured the next step was focusing on additional software used for communication etc. in the FoSIBLE project.

The main component of the FoSIBLE project is the TV widget which cannot run on its own. Everything that is visible inside the widget is based on the social media backend that is installed on the main server. It is just an additional output method for the data and information that is stored on the server. The social media backend can also be accessed via a standard browser on any PC or laptop. However, the goal of the project was to have the TV set as the main in and output device. But it was not possible to program a widget that offers the possibilities of the current backend due to technical restrictions (database needed, performance of the TV set OS). So all components are based on the community platform and all information is accessible via standard web interfaces.

The communication between the different components is based on standard web protocols like RSS, http, ssl.

The TV widget and also the platform itself have not reached the final stage yet as there are still some open tasks. Also the upcoming research in the test households is supposed to be a key source for input regarding the functions of the platform, the TV widget and the various technical devices used in the project.

The next planned tasks are:

- Further optimization of Android tablet application so that it can also be used on other Android devices
- Adding more TV remote control function to the tablet PC application in order to remove the normal TV remote completely
- Providing the output of the online platform to the tablet PC so that the user is not forced to use the widget but can also do all actions on the tablet itself
- Adding a function to the TV widget to search the available clubs and to add new ones or delete some from the list
- Adding a function for chatting with non-friends (global chat)
- A setup of authorization lists for created clubs (e.g. for vital data)