POSEiDON

Pers**O**nalized **S**mart **E**nvironments to increase **I**nclusion of people with **Down** sy**N**drome

Deliverable D4.1

Interface Strategy v3

Call:	FP7-ICT-2013-10
Objective:	ICT-2013.5.3 ICT for smart and personalised inclusion
Contractual delivery date:	31.10.2016 (M36)
Actual delivery date:	31.10.2016 (M36)
Version:	Final
Editor:	Silvia Rus (FhG)
Contributors:	Riitta Hellmann (Karde) Alexandra Covaci (MU) Lars Thomas Boye (Tellu) Fenglin Han (Karde)
Reviewers:	Dean Kramer (MU) Terje Grimstad (Karde)
Dissemination level:	Public
Number of pages:	58

Contents

1 Executive Summary

The deliverable D4.1 - Interface strategy is associated with Task 4.1 - Interface language from WP4. The deliverable is closely linked to D2.1 - Report on requirements and D2.3 - Report on Design of HW, Interfaces, and Software, taking the collected requirements and putting them into a common interface strategy for the different user interaction devices that are created in the scope of WP4.

The document is based on the design considerations presented in the first version of this deliverable, which are shaped by observations we made during the second group workshop. The deliverable will first introduce the topic in a few short paragraphs, including some references to the most important literature and how the deliverable is related to the other tasks and deliverables of the project. Afterwards, we introduce the rationale of the interface strategy, including the timeline and scope within the project, the collected requirements and their impact on the interface strategy, the current iteration of the hardware architecture, and best practice as gathered in the proposal phase and in the creation of this document.

In Chapter 3 we introduce the preliminary interface strategy that will drive the development of the user interfaces in the first phase of the project, including the introduction of some personas that will drive the strategy for the different types of interfaces that are subsequently introduced. In the next section, we list the technical implications of this strategy on the different interfaces that are developed in the first phase of POSEIDON.

Chapter 4 of this deliverable is dedicated to the description of the activities that were organized during the second user group workshop. We used the opportunity of this workshop to investigate certain aspects of interest in the design of our strategy. In order to deliver a stable system and to mitigate the device abandonment phenomenon, we have presented a series of demos meant to demonstrate the concept of this project and to gather feedback from both secondary and primary users. This feedback will be used as guidance information in the development process. We present our approach as resulted from this workshop in two subsections dealing with the interactive table, and the Virtual Reality system.

Chapter 5 presents the interface development guidelines extracted from D2.3 Report on HW, Interfaces and Software. These are extended by POSEIDON specific recommendations which will be the guide for each development while implementing the user interfaces presented in Chapters 9 and 10. Many of the developed POSEIDON components have as starting point paper prototypes or mock-ups. Guidelines on how to design these are presented in Chapter 6. Resulting from these two chapters, common terminology, POSEIDON specific logos, symbols and icons are presented in Chapter 7, while Chapter 8 is a concrete guideline for developers to follow the POSEIDON interface strategy.

In Chapter 9 we present the implemented interfaces strategy by describing the interfaces of POSEIDON components for pilot 1. Taking the feedback from pilot 1 into account we show in Chapter 10 how the improvements were implemented in pilot 2. The development of interfaces throughout the project was supported by many paper prototypes which are used to show the iterative development steps to gather input from users and secondary users before the actual implementation.

2 Introduction

User interface design (UID) is the design of websites, computers and software application with the focus on the user's experience and interaction. Hereby a user-centered-design has as goal to ensure an interaction is as simple and efficient as possible, while the users are accomplishing their goals.

To ensure the good quality of user interfaces, principles and guidelines have been formulated [1] which are applicable for all kinds of user interfaces, whether they are websites, applications or devices. These guidelines are especially concerned with accessibility.

"Worldwide, there are more than 750 million people with disabilities. As we move towards a highly connected world, it is critical that the Web be usable by anyone, regardless of individual capabilities and disabilities," said Tim Berners-Lee, Director of the W3C and inventor of the World Wide Web. [2]

The goal of the W3C is to make content accessible to a wider range of people with disabilities. They address blindness, low vision, deafness, hearing loss, learning disabilities, cognitive limitations, limited movement, speech disabilities, photosensitivity, and a combination of the above. The target group on which POSEIDON is focusing, people with Down syndrome, might present several of these impairments in different combinations and development. Thus, a user interface design for POSEIDON should strongly take the WCAG guidelines into account.

Due to the different nature and combination of the impairments people with Down syndrome are suffering from, customizable interfaces are needed. *Adaptive interfaces* change modality, layout and elements according to the user's preferences either manually or automatically, triggered by a changed context [3]. This allows the interface to be conformant to specific user needs. Mostly it is required to change presentation and navigation according to requirements. This research area has been in development for several years. An important topic is the modeling of the user either autonomously or via user input [4], allowing to control the interface according to specific context. Modeling in this context describes creating a profile of the individual capabilities that should be considered when designing the user interface. Considering our tackled scenarios, a combination of both methods is viable using a combination of prior knowledge as ground truth and autonomous adaptation based on the current context. Some evaluations have been done on the challenges posed by current computer interfaces to people with DS [5], [6] with some interesting outcomes on identifying what may be more useful in terms of communication and safety. However, there is still much to do, especially with the adaptation of interfaces to contexts, preferences and needs. Another aspect of adaptive interfaces is recently becoming more apparent in a paradigm called responsive web design [7]. This proclaims crafting sites to provide an optimal user experience.

Due to the user-centered design, the preferences have been investigated by evaluating the results of the Task 2.1 Requirements gathering. These results are presented in the deliverable D2.1 Report on Requirements, revised after user workshops and pilots.

2.1 Timeline of interface strategy development

The interface strategy strongly depends on the requirements gathered during the requirement analysis. The interviews of the primary user, the online questionnaires of the secondary and tertiary user and the first user workshop have all contributed to this process. From these requirements, an interface strategy is extracted, imminently followed by an implementation in form of an integrated prototype. In Figure 1, *(b)* and *(d),* the requirements gathering phase and the first user workshop followed by *f* when the first interfaces and interactive technology are set up, are represented

respectively. The created prototype *(g)* is evaluated in the second user workshop *(h)* and the outcome of the workshops is subsequently analyzed. The feedback is considered and the interface (l) is adjusted, when the interfaces are completely defined. This iterative process for the interfaces is finished in step r, when the improved interfaces are set up.

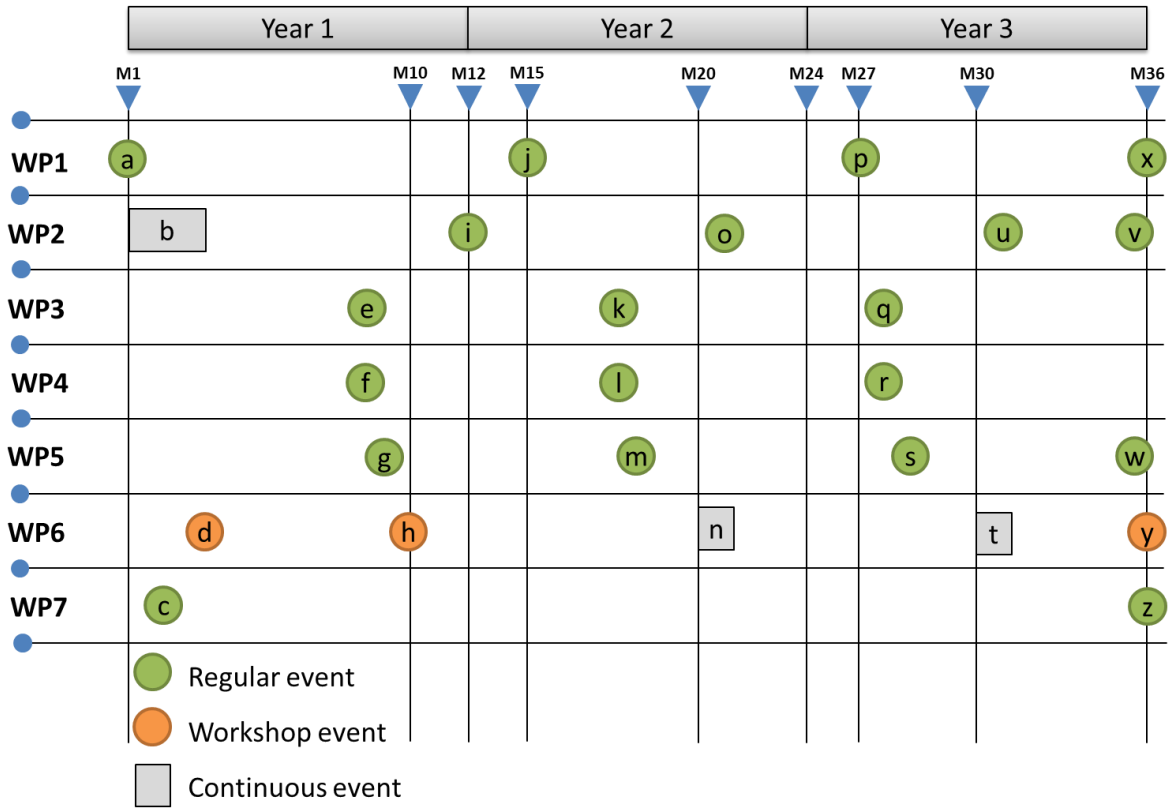


Figure 1 Project and work package milestones and events

3 Interface strategy

The interface strategy explains the overarching principles and the mission of designing the different user interfaces within the POSEIDON project. An interface strategy will describe the initial considerations regarding the user interfaces and how they are developed throughout the project run time. The initial considerations should include technical prerequisites, information about the potential users of the interface and the goals that should be achieved with the specific interface. In this section, we will describe all the different aspects of the interface strategy. It is structured by giving a short overview of the interface strategy template and its progression throughout the project runtime.

3.1 Strategy rationale

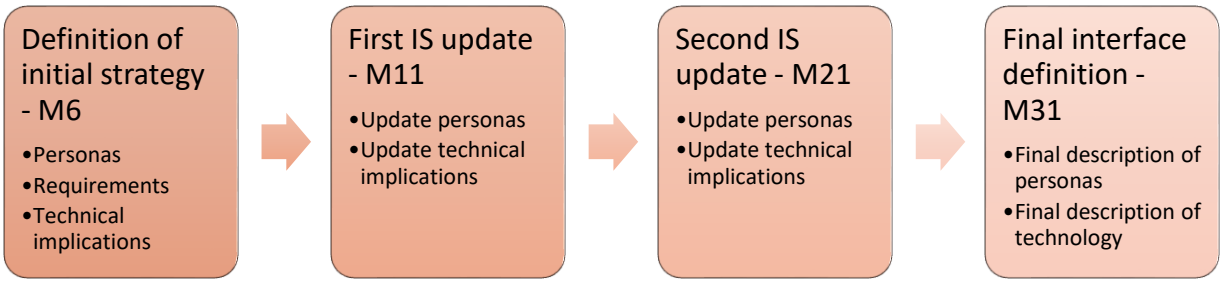


Figure 2 Development of interface strategy throughout project runtime

The interface strategy is based on the requirements created throughout the requirement gathering phase of the project as noted in D2.1 - Report on requirements. The interface strategy will be updated several times throughout the project run time. As shown in Figure , after the initial strategy presented in this document there will be two intermediate strategy updates, informed by the piloting and referring to the project plan as shown in Figure 1.

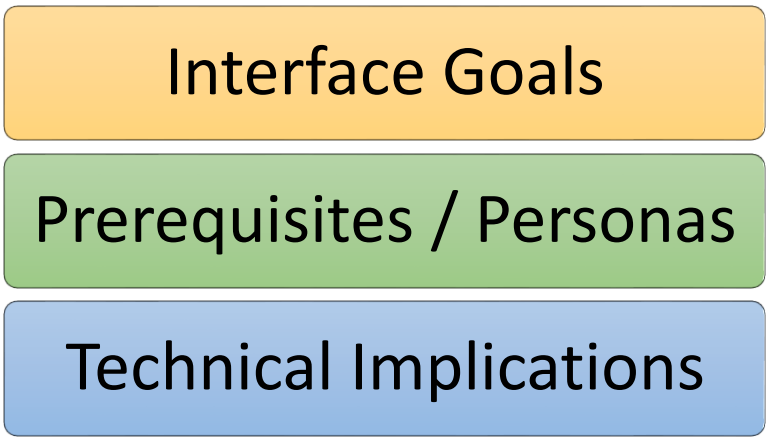


Figure 3 Layers of interface strategy

The interface strategy is distinguished into three different layers as shown in Figure . On the top level, we define a set of goals we like to achieve. This level is informed by the requirements collected and assumptions given in the description of work. The middle layer is comprised of prerequisite that are shared among all technical platforms and may also include Personas to drive the further description of scenarios and use cases and how they may influence the design of the user interface. The bottom layer is a collection of the concrete technical implications - that is both requirements on design, as well as specific decisions regarding the design of the different interface components.

3.2 Interface strategy adaptation method

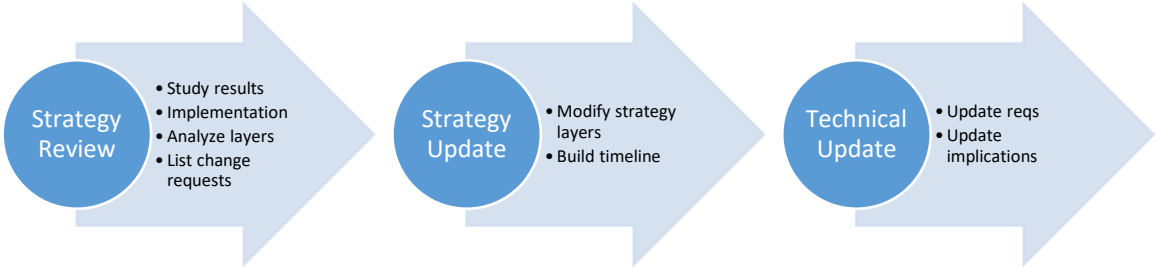


Figure 4 Strategy adaptation process

The adaptation of the interface strategy follows a simple three-step process that is outlined in Figure .

1. In the scope of the strategy review the results of performed user studies and experiences of the implementation process are taken to analyze the different layers of the interface strategy to check if it is still viable. Most likely the scope will be extended and certain parts of the strategy will be further detailed.
2. In the scope of the strategy update the list of change requests will be taken to modify the different layers of the interface strategy. The change requests should be incorporated as far as possible. There should also be some sort of prioritization that will define which parts of the changes are to be implemented first. This part is associated with the different updates of the interface strategy.
3. The technical update will update the concrete technical implications according to the updated strategy. Most likely this will add further parts of the interface to the different technical components.

The adaptation process was performed three times throughout the project duration. The first update occurred in M11, a second update occurred in M20, closely following the goals of the first update, preparing the interfaces for the piloting. The last update outlined all aspects of the interface and thus create the final interface strategy in M31.

The description of the adaptation method is abstract and not intended to change within the POSEIDON run time unless some extraordinary events occur. The adaptation will be performed primarily by persons involved in Task 4.1 Interface language, with informed collaboration to WP4 lead and the task leads of the remaining tasks in this work package. The circle of persons should be kept small, to prevent an overly specific strategy. The detailed specification of the interface strategy should instead be done in the scope of the single tasks.

3.3 Initial strategy

This section will outline the initial interface strategy. After briefly revisiting the hardware architecture, a set of Personas is introduced, a short overview of prerequisites is given and the goals of the interface are described. The resulting list of technical implications will be given in the following section.

Smart Environment (technology infrastructure layer)	**Tablets + Interactive Table + Virtual Reality Set**
Ambient Intelligence (software layer)	
Type of Services required	Friendly Interfaces, Personalization Options, Activity Planner, Mobility Support, Safety, Privacy, support for learning, work training, communication and socialization.
Areas to be used and further developed	Human-Computer Interfaces, Virtual Reality, Context-Awareness, Software Engineering, Artificial Intelligence, Cloud Services, Client-Server, Social Computing, Standardization, Ethics, Privacy.

Figure 5 DoW component architecture

To build the initial strategy, it is necessary to have the most recent hardware architecture for reference. The architecture is currently distinguished into two different components. Figure is the component view (both hardware and software) as specified by the DoW, focusing also on different services and new components that should be developed.

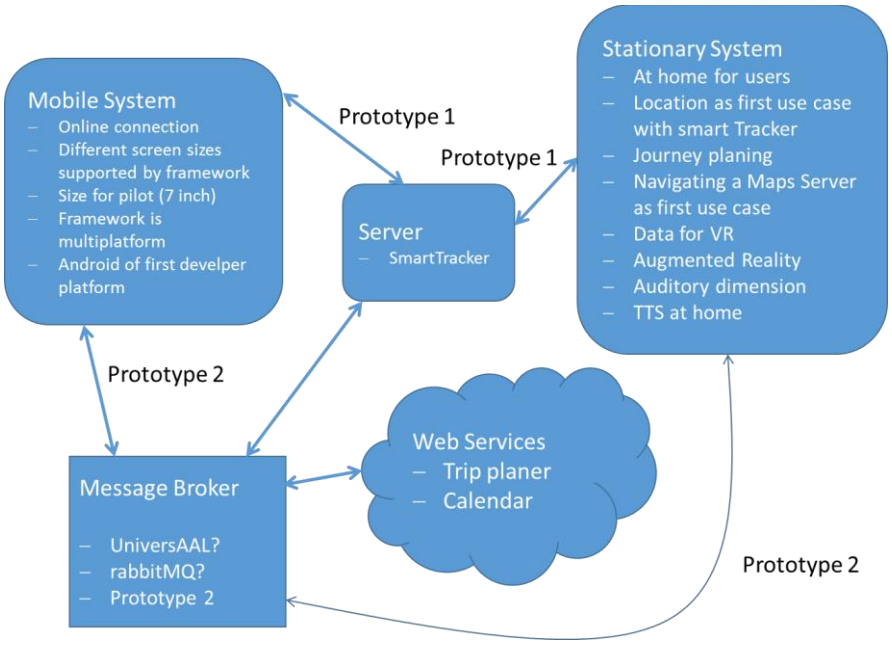


Figure 6 Initial technical component view and mapping to prototype iterations

In the technical kick-off an initial set of functions and components was determined and mapped to different iterations of the prototypes within the project run time, as shown in Figure . We can distinguish three different user centric systems: 1. The mobile system, intended for being carried around while the end user is out of his home. 2. The stationary system intended to be used by the end user, as well as family, while being at home; and 3. The server that provides a set of user interfaces for administration of the system and configuration of certain aspects.

In the following subsections, we will refer to these architectures when required and additionally we will structure the technical implications part according to the technical component view.

3.3.1 Personas

Regarding personas, we can distinguish three different groups of users. The first are persons with DS that are using the POSEIDON system to its full extent. The second are family and associates that use the POSEIDON system together with the end users. The final group are carers and care administrators that are assigned to the specific end user. The overview of the different personas can be found in Table .

Table 1 Overview of personas

Name (country)	**Type**	**Gender, age**	**Characteristics**
Jennifer (UK)	End-user	Female, 19	• Living with her parents • Moderate learning disability • Visits college • Problems in managing time and money • Uses PC
Erik (Norway)	End-user	Male, 25	• Living in supported living home • Mental age lower than physical • Very independent • Gets help from Assistive Technology • Struggles with concentration • Uses every kind of technology
Dorothy (UK)	Family	Female, 44	• Mother of Jennifer • Works part-time • Supports independence of Jennifer • Concerned about Jennifer when outside of home • Low use of technology - manages mail on PC
Anders (Norway)	Family	Male, 51	• Father of Erik • Engineer working out-of-town very frequently • Uses every kind of technology - owns smart phone, laptop, etc. • Uses different forms of communication to stay in touch with Erik daily
Michael (UK)	Carer	Male, 38	• Assigned carer for Jennifer's family • Visits once a week • Corresponds to Jennifer and family using the POSEIDON web system
Marit (Norway)	Carer	Female, 31	• Works in Erik's supported living home • Organizes several events • Manages assistive technology • Uses POSEIDON to stay in touch with several patients

3.3.2 Shared prerequisites

Table 2 Overview of user interface languages (UIL) *[12]*

UIL	Models	Methodology	Tools	Supported languages	Supported platforms	Level	Tags	Concepts
DISL	Presentation, dialog and control	Specification of a generic, platform-independent multimodal UI	Rendering engine	VoiceXML, Java MIDP, Java Swing, Visual C++	Mobile and limited devices	Model level	Not specified	Head element, interface classes (structure, style, behavior), state, generic widgets
GIML	Presentation, dialog, and domain	Specification of a generic interface description.	GITK (Generalized Interface Toolkit)	C++, Java, Perl	Not specified	Meta-model	15 tags	Interface, dialog, widget, objects
ISML	Presentation, task, dialog, domain	Specification of a generic UI description	Under construction	Java, Microsoft foundation class, Java swing classes	Desktop PC, 3D screen	Model level	Not specified	Mappings and constrains, action events, meta-objects, display parts, controller parts, interaction definition
RIML	There is no information	Specification of a generic UI description	There is no information	XHTML, XFORMS, XEvents, WML	Smart phone, pda, Mobile, Desktop Pc	Model level	There is no information	Dialog, Adaptation, layout, element
Seescoa XML	Task, Presentation, dialog	Specification of a generic UI description	CCOM (BetaVersion 1.0 2002) PacoSuite MSC Editor	Java AWT, Swing, HTML, java.microedition, applet, VoxML, WML Juggler	Mobile, desktop PC, Palm III	Model level	Not specified	Component, port, connector, contract, participant, blueprint, instance, scenario, pltform, user, device
SunML	Presentation, dialog, domain	Specification of a generic UI description	SunML Compiler	Java Swing, voiceXML, HTML, UIML,	Desktop Pc, pda	Model level	14 tags	Element, list, link, dialog, interface, generic events, synchronization
Teresa XML	Presentation, task, dialog	Specification of a generic UI description	CTTE Tool for task Models Teresa	Markup: Digital TV, VoiceXML, XHTML/SVG, X+V Programming: C#	DigitalTV, Mobile, Desktop PC,	Model level	19 tags	Mappings, models, , platform, task, input, output
UIML	Presentation, dialog, domain	Specification of a generic UI description	UIML.net, VoiceXML renderer, WML renderer, VB2UMIL	HTML, Java, C++, VoiceXML, QT, CORBA, and WML	desktop PC, a handheld device, tv, mobile	Model level	50 tags	interconnection of the user interface to business logic, services
WSXL	Presentation,dialog, domain	Specification of a generic UI description	Not specified	HTML	PC, Mobile phone,	Model level	12 tags	CUI=XForms, WSDL, Mapping=XLang Workflow=WSFL, Logic=XML event
XICL	Presentation,dialog,	Specification of a generic UI description	XICL STUDIO	HTML, ECMAScript, CSS e DOM.	desktop PC	Model level	Not specified	Component, structure, script, events, properties, interface
XIML	Presentation, task, dialog, domain	Specification of a generic UI description	XIML Schema	HTML, java swing, WLM	Mobile, desktop PC, PDA	Model level	32 tags	Mappings, models, sub models, elements, attributes and relations between the elements

As far as shared prerequisites are concerned we were looking primarily at two different aspects. We had already presented several sources and overview of best practice in interface design. For our initial strategy, we also considered two other sources that are specific to the POSEIDON solution. There is a multitude of different interface languages that support creating expressive, easy-to-use user interfaces. To find an informed selection of which language to choose we looked at previous surveys in this area. We considered several well-regarded publications that covered two different aspects.

The first aspect was the expressiveness of user interface languages. Guerrero-Garcia et al. [12], have performed a review of different user interface design languages. The results are shown in Table . They are distinguishing different models, methodology, available tools, supported languages, supported platforms, level of abstraction, number of tags and supported concepts. A similar web-centric analysis was performed by Pohja [13].

The second aspect was adaptive user interfaces. A recent work by Bongartz et al. [14] analyzes these in the scope of smart environments. We consider adaptive user interfaces separately, as they are important for the personalization aspect of POSEIDON. Again, there is a large body of literature available. Bongartz et al. describe a system the architecture of which "… is built upon the concept of

model-based UI design extended by context aware and adaptive features. Model-based languages provide the software development process with useful support for, building design prototypes and actual implementations for devices with various interaction resources. The proposed architecture can adapt to selected aspects of the context during run-time by communicating with a context server and applying the specified adaptation rules. In order to show the possibilities of the proposed solution, we report on its application in the development of an adaptive user interface prototype to be used in a warehouse picking system." [14]. Similar considerations are considered when building the context adaptive aspect of the POSEIDON system.

3.3.3 Interface goals

After we have specified personas and prerequisites we can create a list of interface goals that we would like to achieve in our first interface strategy. There is a fairly small set of main goals that are supported by a set of subordinate goals. They are collected in the following Table 3.

Table 3 Interface goals and sub-goals

Interface goal	Subgoal
Suitability for persons with DS	• Follow specified requirements from WP2 • Follow best practice for accessible systems • Define scenarios based on personas Jennifer and Erik
Suitability for family	• Follow specified requirements from WP2 • Design task efficient aspects • Define scenarios based on personas Dorothy and Anders
Suitability for carers	• Functional approach to designing interface • Task efficiency should be ensured • Define scenarios based on personas Michael and Marit
Support for adaptive user interfaces	• Analyze best practice for most suitable interface • Define adaptation aspects and context
Conformance to standards of the different technical systems	• Adhere if possible to standard practices on the different systems

3.4 List of technical implications

In the following Table 4 the different technical implications are listed. We were also following the set of different requirements that can be associated to the different interfaces as defined in D2.1.

Table 4 List of technical implications on the different systems

Technical component	Implication or feature
Mobile interface	• Create technology base for mobile interface • Define common aspects for different mobile platforms • Evaluate languages for platform conformity and adaptation potentials

	• Find suitable interface description language for user oriented interfaces • Early test of interface with end users
Stationary interface	• Create technology base for VR • Define suitable tools for VR interface creation • Create technology base for interactive table • Define interface language for interactive table interface • Find suitable interface description language for user oriented interfaces • Ensure alternative usage scenarios for personalization
Web interface	• Create technology base for web interface • Find suitable interface description language for task oriented interfaces • Early test with family and carers

3.5 Conclusion regarding the initial strategy

On the previous pages, we have outlined the interface strategy for POSEIDON in its initial version. There was additional versions of this deliverable in M11, M21 and M31, leading to the final interface strategy.

The document at first gave a rationale for the strategy, including a short presentation of the timeline, a collection of user requirements as specified in WP2 and a collection of best practices, focusing on aspects of interface development that will act as guide for the different tasks within WP4. There was a focus on presenting best practices already available in this domain, to avoid repeating work, already performed.

Finally, we presented the interface strategy, comprised of strategy rationale, methods and procedures to adapt the interface in the following iterations, an introduction to the interface strategy, a collection of Personas and shared prerequisites that are necessary for developing interfaces for the specified hardware devices. This part concluded with a concrete list of technical implications associated to the different interfaces that will be developed within POSEIDON.

This document acted as a basis for all further activities within WP4, driving the development of user interfaces in the different tasks for all different device categories that are part of the POSEIDON architecture.

4 Interface strategy after second User Group Workshop

The strategy we presented in the first version of D4.1 Interface strategy represented our first approach of this project. We emphasised the main characteristics of the technology we were developing based on user studies and general guidance indications. We considered the expressiveness of the user interface languages and the development of adaptive user interfaces.

We illustrated this initial strategy by the means of demos covering the main aspects of interest in developing the system. Thus, we have investigated topics related to the interactive table, the virtual and augmented reality system and the context awareness. The process consisted of presentations where the concepts and examples for these technologies were shown to primary and secondary users coming from 5 countries: Germany, Portugal, Luxembourg, Ukraine, and Switzerland. All these presentations were followed by feedback sessions where the developers and the users interacted with the aim to discover the best design and implementation decisions. We will present this process in the following subsections.

4.1 Prototype 1 – second User Group Workshop – Mainz, Germany

The second User Group Workshop took place in Mainz, Germany 29th – 31st of August 2014. Its goal was to bring developers and users together to discuss and conclude about the strategy update process.

The users who attended the workshop were both primary users (people with Down syndrome) and secondary users (families and carers). A short overview of some of the participants is presented in Table 5.

Table 5 Overview of the participants at the Second User Workshop

Name (country)	**Type**	**Gender, age**	**Characteristics**
Dennis (Germany)	End-user	Male, 29	• Living with his parents • Moderate learning disability • Visited college • Uses PC and tablet for games • Gives regular talks at universities
Rita (Germany)	Family	Female, 55	• Mother of Dennis • Works at a Down syndrome association • Supports Dennis' independence • Moderate concerned about Denis when he is outside the home • Moderate usage of technology
Frank (Germany)	End-user	Male, 36	• Living with his parents • Works with wood in a workshop • Travels by train to work • Plays guitar and has the brown belt in Thai Chi • Uses PC for games
Ana (Portugal)	End-user	Female, 21	• Lives with her parents • Completed 9th degree at 18 years of age • Moderate use of technology
Amandine (Luxembourg)	End-user	Female, 18	• Lives with parents but she will move soon to a supported living home • Uses every kind of technology

Ivan (Ukraine)	End-user	Male, 21	• Very independent • Lives with his parents, but plans to move alone • Works at jewelry workshop • Owns and uses smartphone, tablet
Tetyana (Ukraine)	Family	Female	• Ivan's mother • Supports Ivan's independence • Moderate usage of technology
Delfine (Luxembourg)	Family	Female	• Amandine's mother • She doesn't work – she dedicated her life to taking care of Amandine • Moderate concern about Amandine falling asleep when traveling • Moderate usage of technology
Tania (Luxembourg)	Carer	Female, 29	• Amandine's carer • Visits once a week • Uses every kind of technology - owns smartphone, laptop, etc.
Damian (Switzerland)	End-user	Male, 23	• Independent • Lives with his parents but plans to move alone • Travels around the world with a theater team • Uses smartphone, PC and tablet
Ursula (Switzerland)	Family	Female	• Damian's mother • She is working from home • Supports Damian's independence • Concern for every time he needs to go somewhere new
Claudia (Switzerland)	Carer	Female	• Damian's carer • Manages assistive technology

The sessions consisted of three main parts: personal presentations made by the primary users, technological presentations made by the developers, sessions where feedback about different aspects of the system was gathered.

The presentations made by the people with Down syndrome were meant to show different aspects of their personalities (degree of independence, level of cognitive disability, technology usage, hobbies, personal characteristics). Discussions with secondary users revealed details about their life, their concerns about the people with Down syndrome, their proficiency in using technology.

This phase was extremely important because it provided a set of initial information that has been used accordingly with other information gathered during this workshop in validating and updating our interface strategy.

4.2 Virtual and augmented Reality interface strategy

Virtual and Augmented Reality are emergent technologies with great potential in improving life for everybody, including people with cognitive disabilities.

During this workshop, we explained the concept behind these technologies and we presented how they could improve the daily life of our target users. This was accomplished by exemplifying with three

applications: the home navigation system, the planning and organisation system, the remote assistance application.

The presentation was followed by a session of questions and answers. To conclude, the users were asked to fill in a short questionnaire where they were asked to evaluate the potential of the presented applications in improving aspects of their life. All the applications were rated positively (maximum grade given by the secondary users and enthusiasm shown by the primary users). The discussions brought up other strategy aspects that we will expose below.

The main questions we wanted to address are related to:

1. Considerations about the usefulness of this type of application, where the users can train things at home, before exploring the real world
2. The interest of the participants in a potential use of this system
3. Design of the technology, essential in having a clearer view on how the information should be structured for people with Down syndrome
4. Interaction information, essential in choosing the right devices, able to maintain the attention and to entertain the users
5. The validation of the chosen tasks

4.2.1 Home Navigation System

The aim of this system is to speed up the learning process of new routes for people with Down syndrome. Every time when the primary users need to go to a new destination (e.g., a new place to meet friends, a new work place), they can rehearse the route before using our system. The carer can customise the system by adding information as preferred by the primary user. The primary user can train to navigate between two different points while she is supported by multi-sensory feedback provided in a safe context. We focus on walking routes, but we could also provide public transit indications, depending on the secondary users' customisation. The environment is displayed on a large screen and the users will have different ways to interact with it.

4.2.1.1 Usefulness and interest

When asked about the usefulness of the presented system, the users gave a positive answer. The present travel preparation routine for some of the users consists of exploring places on Google Street View with the help of the carer. Their visual memory is good; thus, they can easily remember buildings or landscapes.

However, the secondary users emphasised the need for this system to be alternated with real world practice. The people with Down syndrome should combine both methods of training (in virtual and real environment) to assure the transfer of knowledge and skill from one environment to the other.

The unanimity of the positive answers confirms and supports the technology we are proposing. Users were generally enthusiastic about the demo presenting the main idea of the navigation system. Moreover, they saw it as a means to help them in their daily lives.

4.2.1.2 Design of the technology

Another important aspect in the design of the application is choosing the right and most intuitive ways to provide feedback and to keep the user focused. Having this in mind, we demonstrated different types of information that can be provided by the proposed system (see Figure 7).

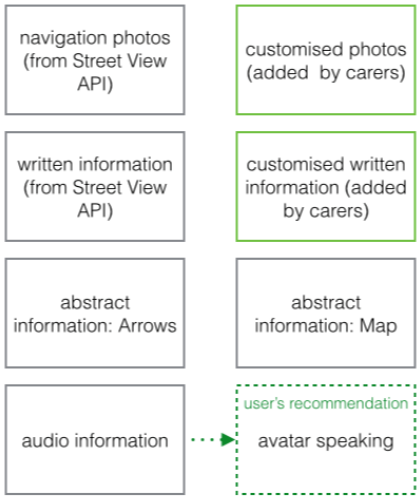


Figure 7 Types of information that were positively evaluated by the users

Figure 8 Snapshot of the Home Navigation Demo presented at the Second User Workshop

In Figure 8 we present a snapshot of the demo we showed at the workshop in Mainz. The snapshot illustrates the carer interface and presents the information offered by the system. The carer can set up different addresses of interest for the primary user and can configure the types of information to be delivered at each step.

The users appreciated the types of feedback we designed and were especially enthusiastic about the possibility to add customised information. Moreover, they suggested that the presence of an avatar could help them remain focused on important events (i.e. crossing streets).

Context awareness is also a desirable feature because extra information can be offered depending on the user's destination. Thus, the user could learn to take different decisions depending on external factors (i.e. weather, day of week). A conceptualised multi-layer architecture is presented in Figure 9.

4.2.1.3 User interaction

As we outlined in D4.4 Virtual reality system, our goal is to offer different interaction options that could enable the user to alternate between them. Based on the literature research, we proposed as interaction means the use of a Kinect sensor, of the interactive table and of the keyboard/mouse.

This workshop provided the opportunity to test the user's reaction to the interaction methods we proposed. The users were engaged in trying to use the interactive table. They looked motivated and expressed their enthusiasm about the interaction with the table. The Kinect sensor was also positively appreciated. These observations validate the chosen interfaces, showing that people with Down syndrome learn easily how to control such devices and their attention is maintained in this process.

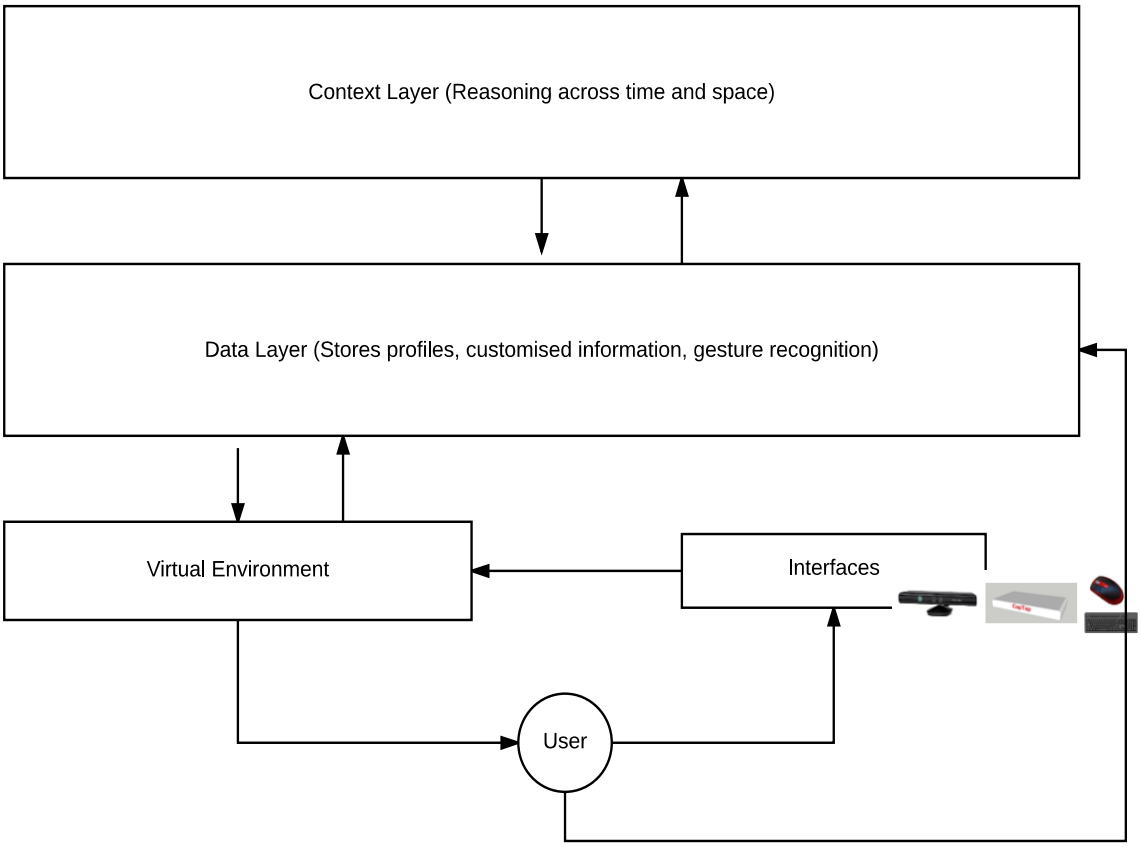


Figure 9 A conceptualised architecture for the Virtual Reality Home navigation system

4.2.2 Contextual reminders and Remote assistance

4.2.2.1 Usefulness and interest

The Augmented Reality technology can be integrated in a variety of ways in building solutions for both learning and support. The workshop held in Mainz provided the context to present demos and ideas related to the inclusion of this technology in the life of people with Down syndrome and to gather feedback from both primary and secondary users.

The Augmented Reality application ideas and demos presented at this workshop consist of: (i) a daily planner application that offers interactive feedback based on calendar activities, and (ii) a remote assistance application that can be used by the people with Down syndrome when they need extra information to enhance the real environment.

The feedback session we conducted showed the importance of a daily planner application in the life of a person with Down syndrome. Moreover, additional support can be added if the daily planner is aware of the context changes. Considering different contexts (location, time, weather) we can enhance the planning process showing primary users information about what objects to take with them or what to wear on a certain day.

4.2.2.2 Design of the technology

All the proposed applications are based on a similar architecture to the one described in Figure 10. The mobile devices we considered for these applications are the tablet and the smartphone. Observations made in the second user workshop confirm the results that the primary users show dexterity and enthusiasm when handling these types of devices.

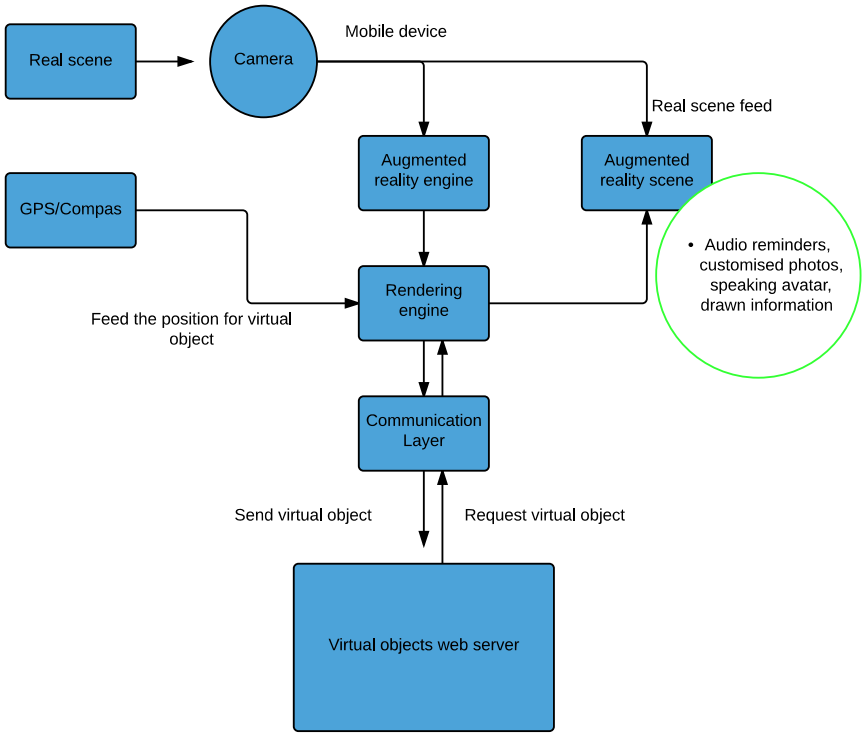


Figure 10 General architecture of the Augmented Reality applications and suggested feedback as resulted after this workshop

Users suggested as the most proper feedback we can provide:

- Custom photos
- Written notifications
- Audio notifications
- Speaking avatar who can help with focus maintenance

4.3 Interactive table interface strategy

The second user group workshop aimed at trying out an application using the interactive table, using different ways of interaction. For this purpose, we connected two applications to the mobile interactive table: a car racing game and Google Earth. In the following the mobile interactive table and the two applications will be described.

4.3.1 Mobile interactive table (Mobile CapTap)

For the second user group workshop and further development of applications we have designed a mobile version of the interactive table. At the end of the project, after the pilot evaluations, the interactive table is unobtrusively integrated into a common couch table, making hand movement interaction possible.

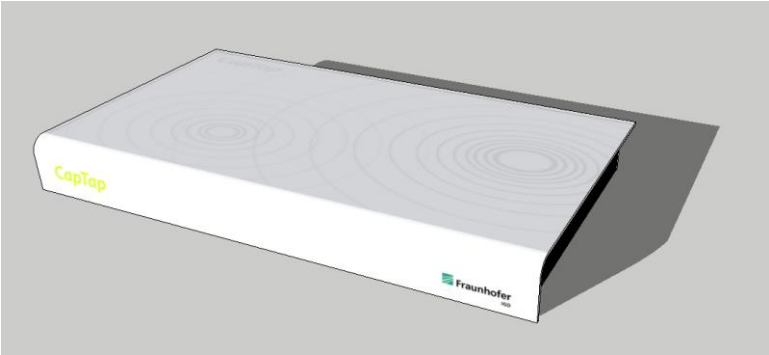


Figure 11. Overview of the design of the mobile interactive table

The mobile version of the interactive table is intended to be placed on top of a surface. It is smaller (90cm x 50cm) than the implementation of the interactive table in the regular living room table (115cm x 60cm) but still large enough to allow wide movements. In Figure 11 we show the initial design while Figure 12 depicts the final implementation of the mobile interactive table.

The design of the mobile interactive table is adjusted to the usage of a single user. We show the user from which side he is expected to interact by having that side of the edge bended. The table can be used single handed or multi handed. However, the two applications are used single handed. Further on, one can interact with the table on multiple levels: one level for touch and three levels for free-air gestures. On the touch level knocking, taping and swiping can be used as input while dwelling in the air, swiping in the air and the change between air interaction levels can be used as input for the free-air interaction. For more details refer to D4.1 Interactive Table, where the implementation and the functionality of the table is described in more detail. For more information on the mobile interactive table, please refer to the second version of D5.2 Prototypic systems where in Section 6 the manufacturing process and the software is described in detail.

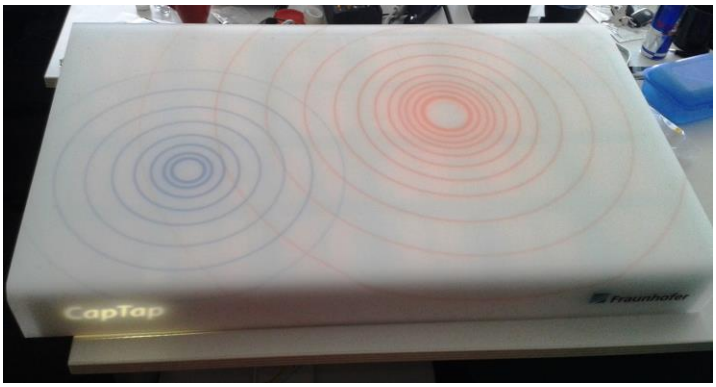


Figure 12 Completed mobile interactive table

4.3.2 Car racing

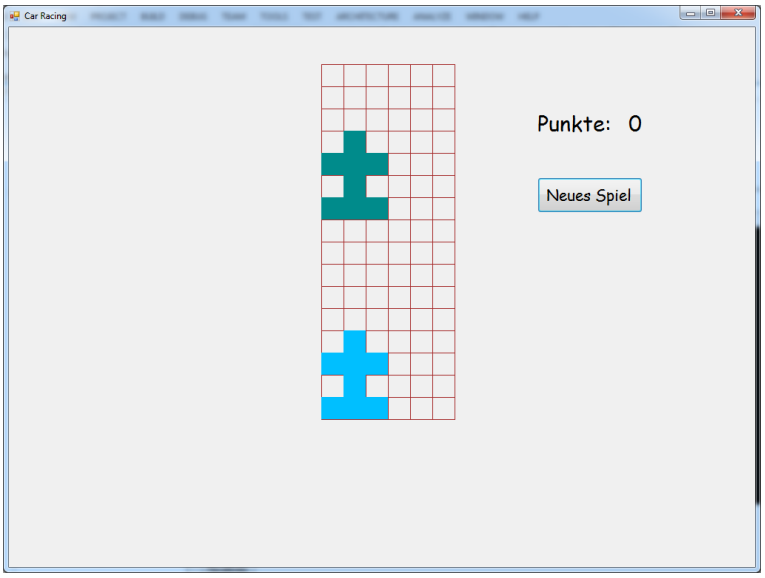


Figure 13 Car race game for interactive table

The car racing application is a simple game in which one must avoid crashing into the other cars. The speed of the cars constantly increases. We have adapted this game to be used with the input of the mobile CapTap. Figure 13 shows a screenshot from the game. The original game is controlled by using the left and right arrow of the keyboard. Translating these commands into the interaction possibilities of the table we have come up with two different solutions to control the game. These solutions are both based on the execution of the left and right swipe using one hand. However, for the second user workshop we tried out the swipe by touching the table or swiping in free-air.

These two interaction methods were compared by letting six primary users participating at the second user workshop try both while playing the game. The initial feedback after the session regarding which interaction method they would prefer for this game was mixed. Most of the primary users were in favour of controlling the game by touching the table, but one primary user very strongly made the point that he likes the free-air interaction since he does not have to touch the surface of the table. We found out that this primary user has some problems with the skin on the hands and fingertips. This is a quite common condition. Hence, we can understand the preferred free-air interaction.

This is an example where the diverse capabilities of the table offer a significant advantage in comparison to other technologies. From this initial feedback the importance of the adaptability of interaction is confirmed. There were no general issues during the workshop regarding the interaction.

4.3.3 Google Earth

Google Earth is a map browsing tool like Google Maps in which 3D buildings can also be seen. It can be controlled with the mouse, zooming in and out and moving in the desired direction. We translated these controls into controls of the table by stepwise moving into the desired direction when touching the desired areas of the table, like shown in Figure 14. Holding one's hand on the top area would move the map up. The same functionality is achieved by holding the hand above the table in the defined areas. The difference between browsing the map by touch and by dwelling in the air is that one is browsing in two predefined zoom settings. So, if one browses by touching, once the hand is held in the air, the view is zoomed out. Here one can browse in bigger steps and zoom in by touching the table again at the desired position.

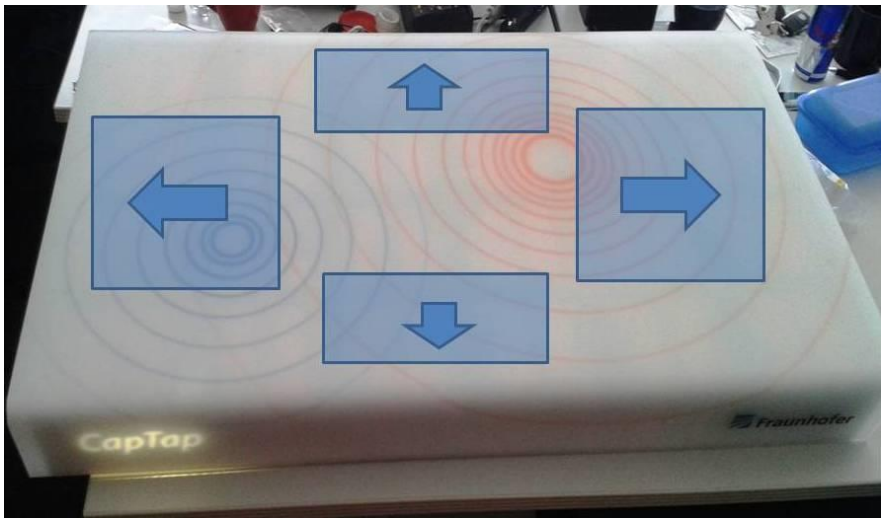


Figure 14 Tap and knock Interaction areas of Interactive table using Google Earth

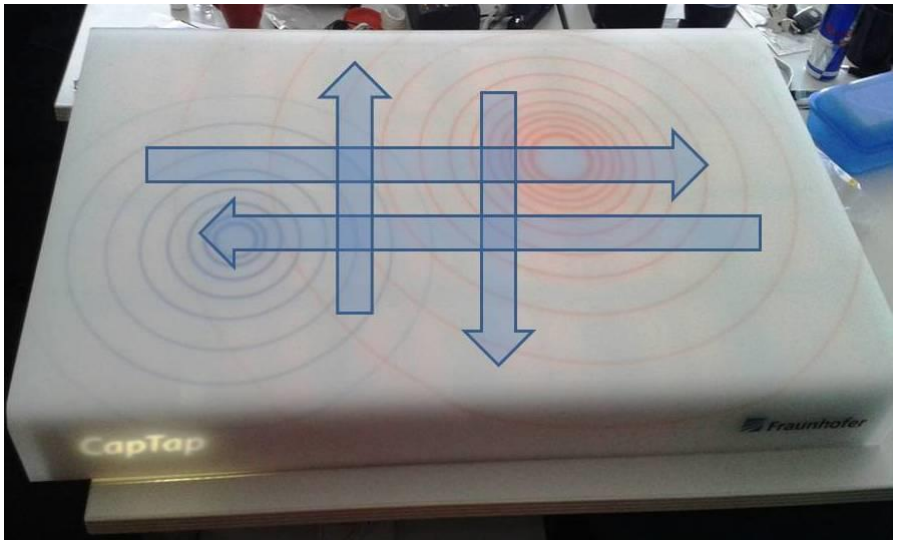


Figure 15 Swipe gestures on surface and in free-air of interactive table using Google Earth

These two predefined zoom areas were intended as a simplification, knowing that people with Down syndrome have difficulties reading maps. So the workshop was intended to see if this simplification would help and the primary users would manage to deal with all the commands. Unfortunately at the event site of the user group workshop the internet connection was too slow to allow a usable experience of Google Earth. Hence, this approach could not be tested there.

4.3.4 Follow-up questionnaire

As a follow up on the user group workshops we prepared a questionnaire for the primary users which should be filled out with the help of the secondary users. Through the questionnaire we would like to have the opinion of all primary user participants which tried out the table, adapting the controls of future applications to their needs.

5 POSEIDON guidelines for developing accessible user interfaces

This chapter provides guidance on usability and accessibility for the designers, developers and the ones responsible for testing and evaluating the POSEIDON system and its services. It provides a collection of practical advice of how to design the interaction so that it meets the requirements and capabilities of the target group of POSEIDON. The goal has been to try to keep these guidelines as simple as possible.

These guidelines are based on several existing guidelines which have been summarized in D2.3 Report on Hardware, Interfaces and Software. The most important references are:

- Principles of universal design[1]
- Information for all: European standards for making information easy to read and understand[2] (follows this document as separate attachment)
- Cognitive Accessibility User Research of W3C[3] Principles of universal design

Universal design is more than accessibility. With a focus on universal design we can have a more holistic approach to how we develop the POSEIDON system and services.

The concept of universal design is based on the design of products and environments to be usable by all people. Some of the principles are difficult to apply to ICT-based products and services. A subset of the principles for universal design will, however, apply for the POSEIDON project. In addition, we propose some implications these will have for the POSEIDON system:

Principle 1: Equitable use

The design is useful and marketable to people with diverse abilities.

Principle 2: Flexibility in use

The design accommodates a wide range of individual preferences and abilities.

POSEIDON recommendations

- All users should be allowed to adjust their preferences for how the system should communicate with them (e.g., size of fonts, contrasts, colours, etc.). Enabling assistive technology such as synthetic speech (i.e., multimodality) is important.
- If a user has tremor, or problems with fine motor skills, and tapping a tablet/smartphone with her fingers could be problematic, the system should be set up so that it is more tolerant for user errors, timing for accepting that a button is pressed etc.

Principle 3: Simple and intuitive use

Use of the design is easy to understand, regardless of the user's experience, knowledge, language skills, or current concentration level.

[1] http://www.ncsu.edu/project/design-projects/udi/center-for-universal-design/the-principles-of-universal-design/

[2] http://www.inclusion-europe.org/images/stories/documents/Project_Pathways1/Information_for_all.pdf

[3] https://w3c.github.io/coga/user-research/#down-syndrome

POSEIDON recommendations

- Only relevant actions should be displayed in the user interface.
- All (most) action buttons should be located at the bottom of the screen, and they should always be visible.
- Help/information should always be located at the bottom-right corner of the screen, or to the right from the specific location where help or additional information is available.
- Search (if relevant) should always be located at the top-right corner of the screen.
- Action buttons should be a combination of icons and text. Exceptions may be made for the user interface, such as lists of choices created by the end-user and where suitable icons are not available, or when uploading of an icon is not provided.
- Focus points of a video service should always be at the centre of the screen (important for video content).

Principle 4: Perceptible information

The design communicates necessary information effectively to the user, regardless of ambient conditions or the user's sensory abilities.

POSEIDON recommendations

- Regardless of context of use, all information on the screen should be easily readable and understandable.
- Textual information presented to the end-user must be meaningful, and consider the end-users' capabilities and cultural frame of understanding.

Principle 5: Tolerance for error

The design minimises hazards and the adverse consequences of accidental or unintended actions.

POSEIDON recommendations

- When the end-user enters wrong data, or is using the system in a "non-predicted" way, the system should be "forgiving" and provide guidance to the end-user, and help to recover the error.
- If there are technical problems (disturbances in data communication, internet connection failure etc.), the system should still try to provide meaningful information to the end-users.

Principle 6: Low physical effort

The design can be used efficiently and comfortably and with a minimum of fatigue.

POSEIDON recommendations

- This principle is important when handling tools, opening doors etc. The POSEIDON system that will be operated on tablet PCs and/or smartphones will require very low physical effort.

Principle 7: Size and space for approach and use

Appropriate size and space is provided for approach, reach, manipulation, and use regardless of user's body size, posture, or mobility.

POSEIDON recommendations

- All action buttons both on the web and on tablet PCs and smartphones must be large and easy to click/tap.

6 Design methodology for mock-up development

To achieve the goal of a highly usable system, it was important that we started the interaction design by developing high-level mock-ups (paper prototypes). These could be pictures of the web-design of the POSEIDON family tools, or tablet PC functionality. This helped the design team to figure out how the POSEIDON system should work, what input does it require, and what kinds of output are expected. When the users perform an action "using" the mock-up, they do have some expectations for what will happen.

It is important that the POSEIDON system meets these expectations. The initial design cycle should work to identify the end-users expectations, and at the same time identify what information is needed from a developer's perspective to make the system work, and to provide the expected result.

For the mock-ups (high-fidelity paper prototypes), we identified a set of design principles that should be followed. These are presented in the next section. Later, these principles was applied to the development of the HCI of the final POSEIDON system.

When developing user interfaces there are some very central design principles to follow[4]:

Principle 1: Learnability

The user interface should be easy to use from the first time a user interacts with it. There should be no need to learn new functionality or new ways of user interaction. The system should be based on recognition rather than the need to recall previous experiences.

POSEIDON recommendations

- All action buttons should be located at the bottom of the screen, and they should always be visible, common tasks should be located at the same place all the time, such as help, search, information etc.
- Based on the experience of the system, the first time a user uses the application some help and guidance should be provided, and after some times of use the help and guidance should be less intrusive.

Principle 2: Efficiency

The number of steps a user takes to complete a task should be as few as possible. The need for horizontal and vertical scrolling should be kept to a minimum. Wizards should be used to simplify complex interactions. Real world metaphors should be used where applicable. Less is more: most likely we need to leave stuff out.

POSEIDON recommendations

- Main device-orientation is horizontal in the case of tablets. For smartphones the vertical orientation is preferable.
- All scrolling is vertical – no horizontal scrolling.

[4] Adapted from: http://www.slideshare.net/OpenRoad/mobile-ui-design-user-centered-design-and-ui-best-practices, http://www.google.com/about/company/philosophy/, http://developer.apple.com/library/ios/#documentation/userexperience/conceptual/mobilehig/UEBestPractices/UEBestPractices.html, http://designfestival.com/5-principles-of-user-centered-interface-design/, http://www.netmagazine.com/features/10-principles-mobile-interface-design, http://uxdesign.smashingmagazine.com/2011/07/18/seven-guidelines-for-designing-high-performance-mobile-user-experiences/

- All action buttons are located at the bottom of the screen, and are always visible. Exceptions to this may be buttons to handle the video content (e.g. Start-button in the middle of the screen and the like).
- Only relevant functionality should be visible.
- All information and help texts should be context sensitive, the information and help text should be relevant and to the point.

Principle 3: Error recovery

The system should be designed so that it is hard or even impossible for a user to make mistakes. However, when a user mistake occurs, this should be clearly communicated with information on which actions to take to continue the use of the system.

If there is a system error, this should also be communicated in a clear way, with simple and understandable information to the end-user. All error messages should be useful. The system should provide guidance on how the user should recover from the error.

POSEIDON recommendations

- There are three different types of errors that need to be addressed and have a consistent error recovery methodology.
    - User error
    - Device application error
    - Server application error (including error on services invoked from the POSEIDON system)
- It should not be necessary to re-enter any data when an error situation appears.
- When restarting the app or service, it should launch at the same state as it was when the error occurred.
- The system should be "forgiving" on user errors and provide mechanism for graceful degradation of functionality when an error situation occurs.
- E.g., if the video service is not responding, the system should continue to work (just not display the video), and the "space" used for the video should not be left blank.

Principle 4: Simplicity

Tasks frequently performed should be easy to do, and less common tasks should be possible to do. Unnecessary functionality should be avoided. The layout and design should be uncluttered.

The navigation should be ***narrow and shallow***, providing only necessary functionality. For this, we need to understand profoundly the context of when and where our users will use the system.

POSEIDON recommendations

- There should not be more than maximum three levels of navigation, ideally there should only be one level of navigation in the POSEIDON system.
- The design and design elements should be clean and simple.
- Where applicable, well-known principles and best practice should be applied.

Principle 5: Mapping

What the user expects to happen is what should happen. There should be a mapping between the conceptual model the user has of the system, and how the system actually works.

POSEIDON recommendations

- The conceptual model behind the POSEIDON system should be well documented and described.

Principle 6: Visibility

The most important information should be most visible, and less important information should be less visible. When using a touch interface, no button should be smaller than the user's fingertips plus "necessary margin" for users with tremor or problems with fine motor skills.

POSEIDON recommendations

- Only relevant actions should be displayed.
- All buttons should be of an easily press able size, with clear boundaries.
- Buttons should not be smaller than the size of the thumb.

Principle 7: Feedback

The user should be in control of the interface and not the other way around. The system should provide quick responses. If the response will take some time a progress bar or some other useful information should be provided. Speed and responsiveness are crucial for the user experience.

In today's computing environment one second is an "eternity" to wait for response from the system or application. If a system does not respond within a reasonable time frame, the users will assume there is an error and try again, or press other buttons that will null-out existing action causing confusion and a bad user experience.

POSEIDON recommendations

- If an action takes more than one second, a progress bar or other relevant information should be displayed to the end-user.

Principle 8: Consistency

Identical items, and identical functionality should always be displayed and behave the same way across the entire system/application.

POSEIDON recommendations

- All agreed common user actions should be tested and documented.
- All confirmation, information, help and error "pages" should have the same look-and-feel throughout the system.
- The system should be as predictable as possible.

Principle 9: Satisfaction

The users should enjoy using the POSEIDON system/software. The software should perform its expected tasks well and nothing more. If she would like to perform another task, she would most likely use another application (or system).

POSEIDON recommendations

- Only core-functionality should be provided by the POSEIDON system.
- For secondary functionality we should defer the end-users to other applications that solve those tasks well.

Principle 10: Predictability

When a system follows the principle of predictability, the user would know what to expect from the system: The behaviour is consistent throughout the application/system/service. With a consistent user interface the user will not experience surprises. When a user presses a button, or invokes a service, it should be evident for the user what to expect, and it should also be evident how the results will be presented.

To ensure a predictable user experience, it is important to understand the targeted users' expectations and the conceptual models[5] they have for the system they are using. If we design a system based on a different conceptual model than the one of the end-users', the user interaction and how they use the system will never match the anticipations of the developers, and the system will score low on usability and expectations of the users. If the system is designed following the conceptual model of the end-users, we will get a high score on usability, because the behaviour of the system is what the end-users predict. The system and the user interaction follow the users' expectations.

Ideally there should not be any surprises for the end-user when using the system. If something unexpected happens, the methods for solving the unexpected should be predictable and well known by all users.

Information for all

All information in the apps and other software products of the POSEIDON project should follow easy-to-read guidelines of Inclusion Europe[6], where appropriate, and possible to implement (i.e. Electronic information and Videos).

7 Terminology and symbols in POSEIDON

When developing an application that shall be used across several countries and regions it is important that the terminology used in the user interfaces are meaningful and accurate regarding what the end-user expects. It is also important that the terminology used in the user interface is easy to translate, and that the terms used are meaningful also in other languages. The terminology used should not only be meaningful in all languages, we should also thrive for language equivalency, and not merely a translation of terms.

When using symbols (icons) in the user interface to communicate different user actions it is important that the symbols used convey the same meaning across cultural borders – as far as possible. The symbols should mean the same for all people using the system. If some symbols could be interpreted differently in different cultures/languages an alternative symbol should be used.

It is also important that the symbols and terminology are used consistently throughout the applications and systems in POSEIDON.

All texts and icons/symbols should be tested and evaluated by relevant user groups in the partner countries of POSEIDON to ensure that they communicate well and convey the same meaning and expectations across culture and language.

In our user interfaces, we will also provide help texts and information texts that will be invoked in different parts of the system. It is important that the help and information texts are contextualised

[5] In this context, a conceptual model is the mental model an end-user has of how the systems works, and the end-users understanding of how different services and functions provided by the system works. An end-user will use the system based on the mental model she has on how the system works.

[6] http://www.inclusion-europe.org/images/stories/documents/Project_Pathways1/Information_for_all.pdf

and only provide relevant information for that specific part of the system. In addition, all texts should be in all POSEIDON partner languages, and the text provided in all languages should be semantically equivalent.

7.1 Visual appearance, icons and colour palettes in POSEIDON

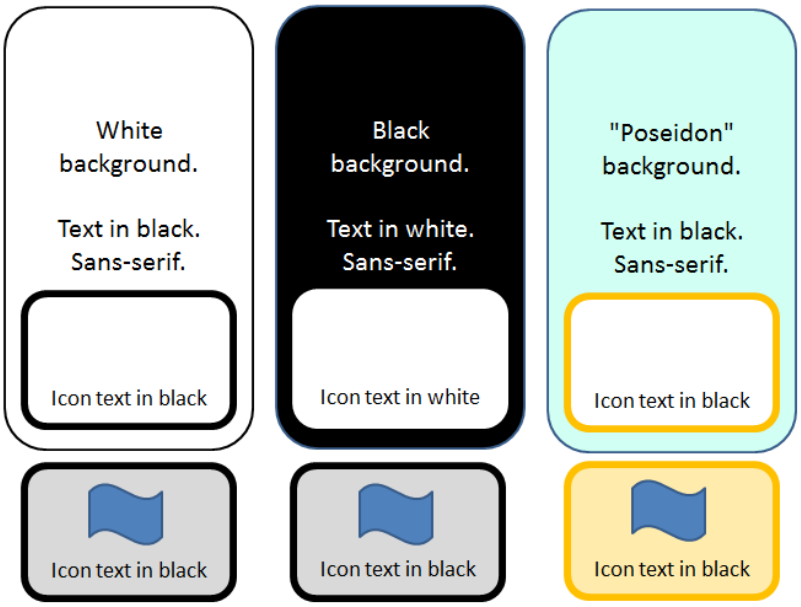
The POSEIDON apps and other software products (interactive table, carer's web) must be designed according to best possible practice of accessibility, and with special attention given to the requirements of persons with Down Syndrome.

The visual design must be very clear, without decorative elements, disturbing animations, or other design elements which may make the interaction difficult. Simple and clear icons, good contrasts everywhere, etc., are basic requirements. Faded edges should be avoided.

The visual appearance must also follow a "family resemblance" so that the user always understands that she in inside the POSEIDON system. Important elements here are choice of colour palettes and the idea behind the icon design. Icons should when possible be combined with text.

How users interpret icons is a difficult matter to handle, though, and ambiguities are certainly permanent to a certain extent. In order to avoid misunderstandings, it is recommended to choose commonly used icons, such as following examples (not necessarily relevant for POSEIDON as such):

For visually impaired users, the POSEIDON system must provide high contrast alternatives, such as black-and white palettes, in addition to the "branded" POSEIDON palette. This is very important for the readability of the text elements.



7.2 Basic accessibility requirements for web based systems

All web based services and information produced for the end-users of the POSEIDON project should ideally meet the following recommendations towards accessibility and usability:

- Separate content from presentation

- Use HTML5
- Use CSS3 for presentation
- Follow the W3C/WCAG2 guidelines7

7.3 Branding

Apart from the accessibility aspects, it may be important to brand the POSEIDOM with a logo. This may be useful for end users who need to recognise the apps and other software products. A sketch is drafted below:

Poseidon

Useful apps for special people

[7] http://www.w3.org/TR/WCAG20/#guidelines

8 POSEIDON interface design document for tests and pilots

The purpose of this chapter is to present the interface and interaction design elements and decisions made for the POSEIDON software. The overall goal is to provide as good accessibility as possible in all parts/apps/services of the software, and to guarantee a clean "look and feel" design.

8.1 The colour schema

- The main requirement is good contrasts everywhere, in all elements of the user interfaces.
- Make the colour "skins" available in an easily adjustable way, not hard-coded in all SW.
- For (at least) first prototypes, start with light backgrounds and dark text, buttons etc.

Selected colours:

System: RGB code		**System: Hex code**		
Black	**R:** 0 **G:** 0 **B:** 0	Black	#000000	
White	**R:** 255 **G:** 255 **B:** 255	White	#FFFFFF	
(POSEIDON) Turquoise	**R:** 53 **G:** 132 **B:** 140	(POSEIDON) Turquoise	#008080	
(POSEIDON) Orange	**R:** 241 **G:** 165 **B:** 50	(POSEIDON) Orange	#F1A532	
Red	**R:** 255 **G:** 0 **B:** 0	Red	#FF0000	
Blue	**R:** 28 **G:** 120 **B:** 204	Blue	#1C78CC	
Grey	**R:** 174 **G:** 167 **B:** 159	Grey	#AEA79F	

8.2 Text elements

- Font: Use only sans-serif screen fonts – no exceptions. **CALIBRI/Calibri**, or if not available, ARIAL/Arial. (Not Times New Roman or the like).
- High contrast choices everywhere, even if "neighbouring elements" would not follow the same principle, e.g.: two buttons beside each other where text is in different colour Black text on grey background White text on blue background.
- Always as large text as possible – even larger than what might feel "normal".
- All text must be implemented as character-based text, not images of text (for enabling synthetic speech later).
- No ALL CAPS text, no capital letters in the middle of expressions when not necessary (e.g. 'Start here', not 'Start Here') – this to harmonise the design and make it "calm". This can be changed later if necessary (NB. Conventions in different countries are different here!).
- No *Italics – at least long texts*. It is difficult to read.
- Normal, easy to understand language. No unfamiliar abbreviations or computer science jargon.

8.3 Action button design

Rounded edges:

"Flat design" (no outline): No 3D-effects: (to harmonise initial design)

Text and place(holder) for transparent icons:

Text

Text

8.4 Icon library

- An icon library contains all POSEIDON icons in necessary formats.
- All icons must be designed as transparent images, with high contrast against all possible backgrounds.
- All icons must follow "similar design". E.g., not a mixture of photo icons, pencil drawings and high quality graphics.

8.5 Decorative elements

- Do not use any unnecessary decorative elements such as boards, background images or the like.
- Do not use any animations (moving, blinking items) if not necessary for explaining something, e.g. a screen-cast connected to user support sections.

8.6 Navigation

- Allow the user to go back or "regret".
- Allow the user to start from beginning (Home, Start page).
- Show the user where s/he is "just now".

8.7 Scrolling

- Avoid scrolling
- Avoid horizontal scrolling

8.8 Actions and feedback to end user

- Let the user initiate actions, e.g. by clicking on buttons – even if not technically necessary.
- Acknowledge actions: show that a button is clicked on, that something may take time etc.

8.9 Language, national data formats

- Create a language library which allows alternative expressions and new languages be deployed without re-programming all SW.
- Dedicate a work force to create an as compact and easy-to-read language as possible, especially for buttons.
- Make it possible to use familiar (diverse) formats for date, time, monetary units etc.

8.10 Help and support

- Provide help. Enable help to be added to the SW "anywhere".
- Use commonly understandable, clickable symbols for more information, e.g. 
- Do not require the user to type information that the system already knows.

8.11 Look and feel

- We want the POSEIDON SW to be as familiar and self-explanatory as possible.
- Strive for realisation of look and feel. Look and feel is a term used in respect of a graphical user interface and comprises aspects of its design, including elements such as colours, shapes, layout, and typefaces (the "look"), as well as the behaviour of dynamic elements such as buttons, boxes, and menus (the "feel")[8].
- Create family resemblance between all parts of the POSEIDON SW by sticking to the rules in this document – even if you would personally disagree with something.

[8] http://en.wikipedia.org/wiki/Look_and_feel

- Brand all apps with the POSEIDON app logos or banners. However, do not let these dominate the interface.

8.12 Responsive design

- We want the POSEIDON software to be usable across a variety of devices, depending on the end user's preferences.
- Strive for realisation of responsive design. Responsive web design is an approach to web design aimed at crafting sites to provide an optimal viewing experience—easy reading and navigation with a minimum of resizing, panning, and scrolling—across a wide range of devices (from desktop computer monitors to mobile phones)[9].

8.13 Accessibility

- We want the POSEIDON software to be usable, useful and easy to use for a variety of end users with different levels of abilities.
- Strive for realisation of accessible solutions on all POSEIDON platforms. Accessibility can be viewed as the "ability to access" and benefit from some system or entity. The concept focuses on enabling access for people with disabilities, or special needs, or enabling access using assistive technology[10].
- This document is just a short "start package" for the SW team. For all more advanced accessibility issues, se accessibility guidelines such as[11].
- POSEIDON has a set of accessibility guidelines collected for use by SW developers.

[9] http://en.wikipedia.org/wiki/Responsive_web_design

[10] http://en.wikipedia.org/wiki/Accessibility

[11] http://www.w3.org/standards/webdesign/accessibility

9 Pilot 1 - Achieved interface strategy

For each component of the POSEIDON System we shortly describe in this section how the interface strategy defined by the interface guidelines, paper-prototype methodology, terminology and developer interface design in Chapter 5-8 has been implemented. We reference here the other parts of deliverables like "D4.5 HCI user and developer manuals" and other specific deliverables where the functionality of the interface is described in more detail.

We aim to deliver a solution that provides a "user friendly" experience, while being functional and succeeding in improving different problematic aspects in the life of people with Down syndrome. This workshop gave us the opportunity to present demos of our strategy and to gather immediate feedback from the users. We presented demos and examples to both secondary and primary users and we designed questionnaires meant to help us evaluate the proposed solutions.

The conclusions were significant showing that our approach is appreciated and considered useful and giving us information about future steps in implementation.

The main point mentioned by the participants was, that primary users should have a mobile device with them outdoors. Because of the space restrains the chosen mobile device was not as initially planned a tablet, but a quite large mobile device with 5.5-inch screen size. Figure 16 shows a graphic presented in D5.2 showing an overview of the POSEIDON system for pilot 1.

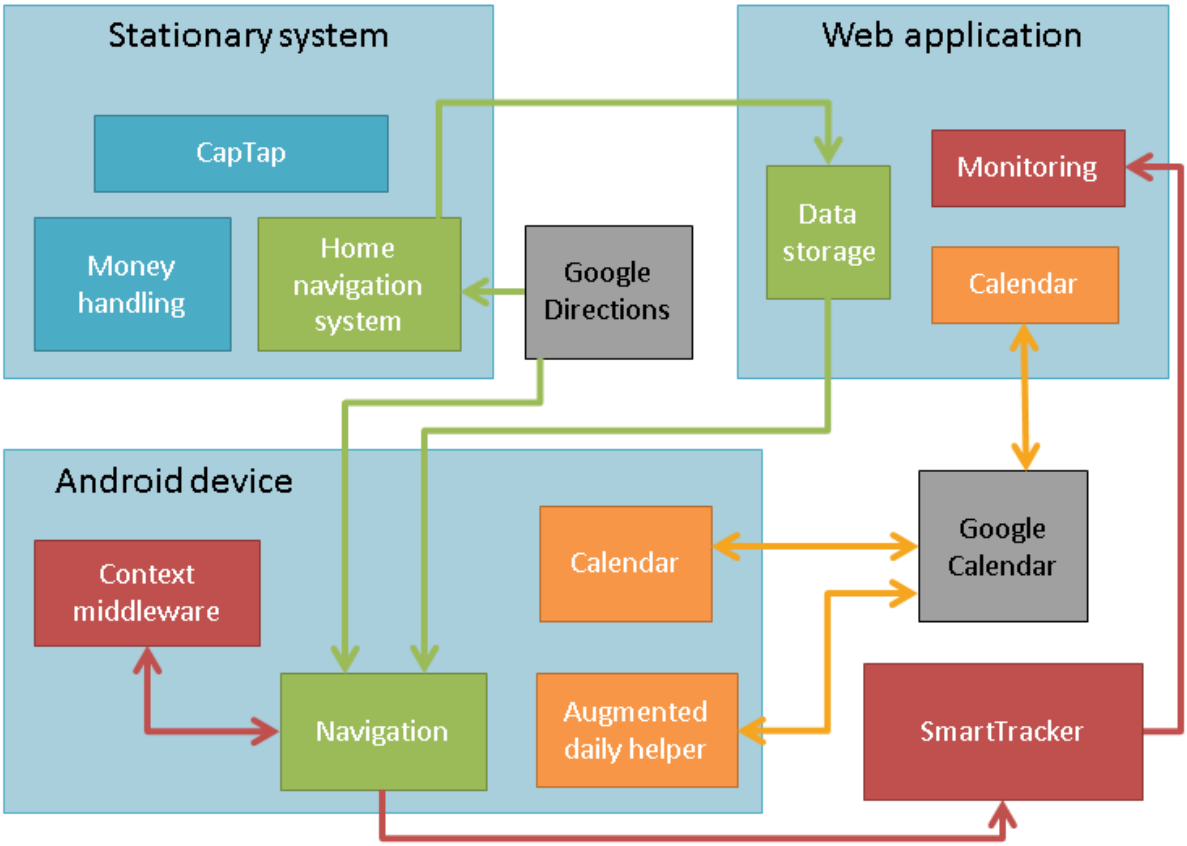


Figure 16 POSEIDON Pilot 1 components overview

In the following we are going to address the interface design process of the different components which are directly in contact with the Primary User and Secondary User: Home Navigation System,

Moneyhandling Training with Interactive Table, POSEIDON App, POSEIDON Web (for Secondary User). Where available, we are going to present paper prototypes and show examples of interface screenshots.

9.1 POSEIDON App

9.1.1 Paper prototypes

As part of the mobile app design process, there was an idea phase where paper prototypes were produced, with examples of functionality POSEIDON app(s) can include and how the user interfaces can be organised. These were sketches, so the visual elements such as icons and colours were not meant for a working prototype. Some design principles for increasing the accessibility of the prototypes were also found in this phase:

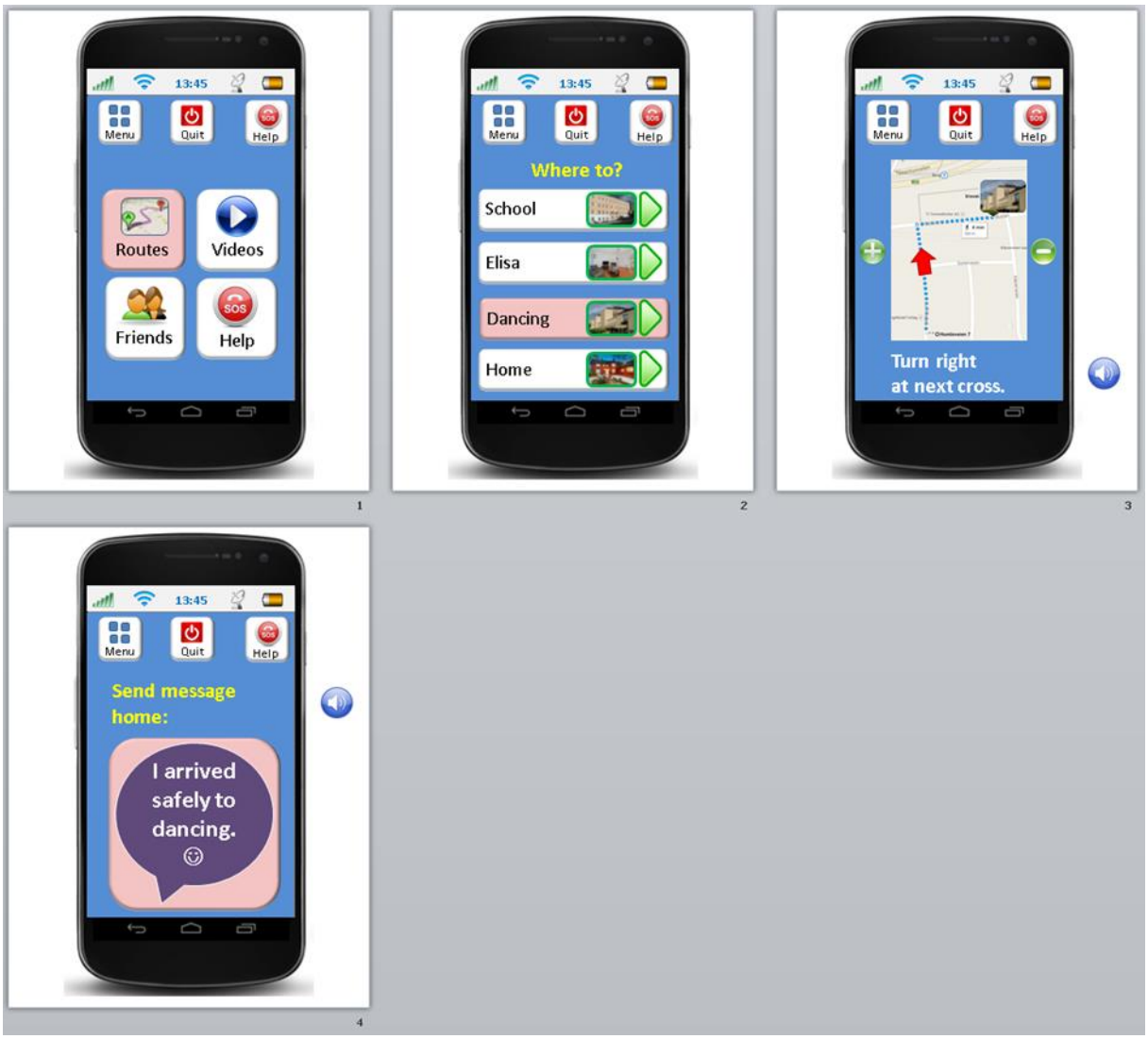
- Clean interface without any decorative or other disturbing elements
- Possibility to switch between normal and high contrast alternatives
- Possibility to choose between small caps and all caps
- Very short texts/sentences
- Clear and simple terminology
- Most parts are illustrated with icon/image, in addition to text
- Many places opportunity to have audio output added
- Possibility to mute sound (privacy)
- "Family resemblance" between the various apps
- Possibility for family members to customize apps on relatives' website (incl. selecting images from standard image sets)
- Use tactile properties where possible (vibration)
- Design interface elements which communicate well, for example that buttons look like objects that can be pressed/tapped, the scroll bars are wide and spacious and appear so that “here you can scroll” etc.

This application provides guidance when travelling outdoors, as well as tracking the user. The first set of screens shows preferences – toggling tracking and speech, changing the colour theme and adjusting font.

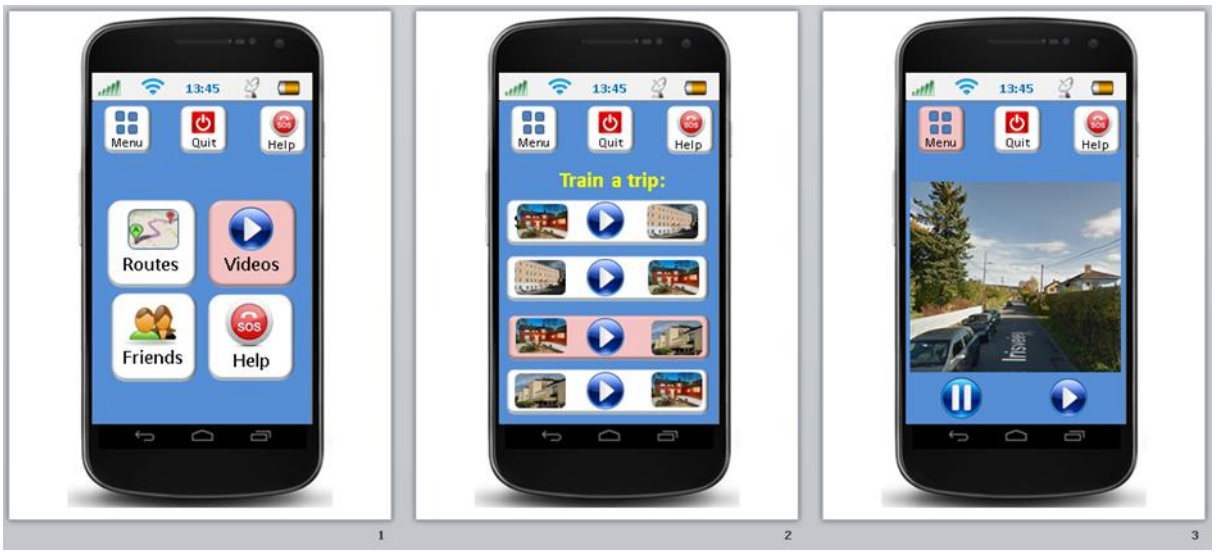

13:45
Menu
Quit
Help
Routes
Videos
Friends
Change
1
Change:
Speech
Colour
AB ab
Font
Tracking
2
Tracking is: ON
Parents can see where you are.
Turn OFF tracking
3
Change:
Speech
Colour
AB ab
Font
Tracking
4
High contrast is: OFF
abc
Turn ON
abc
Turn ON
5
High contrats is: ON
abc
Turn ON
abc
Turn OFF
6


13:45
Menu
Quit
Help
Change:
Speech
Colour
AB ab
Font
Tracking
7
Speech is: ON
People around you can hear messages.
Turn OFF speech
8
Change:
Speech
Colour
AB ab
Font
Tracking
9
Large letters are: OFF
ABCDEFG
Turn ON large
10
MENU
QUIT
HELP
LARGE LETTERS ARE: ON
abcdefg
TURN ON SMALL
11

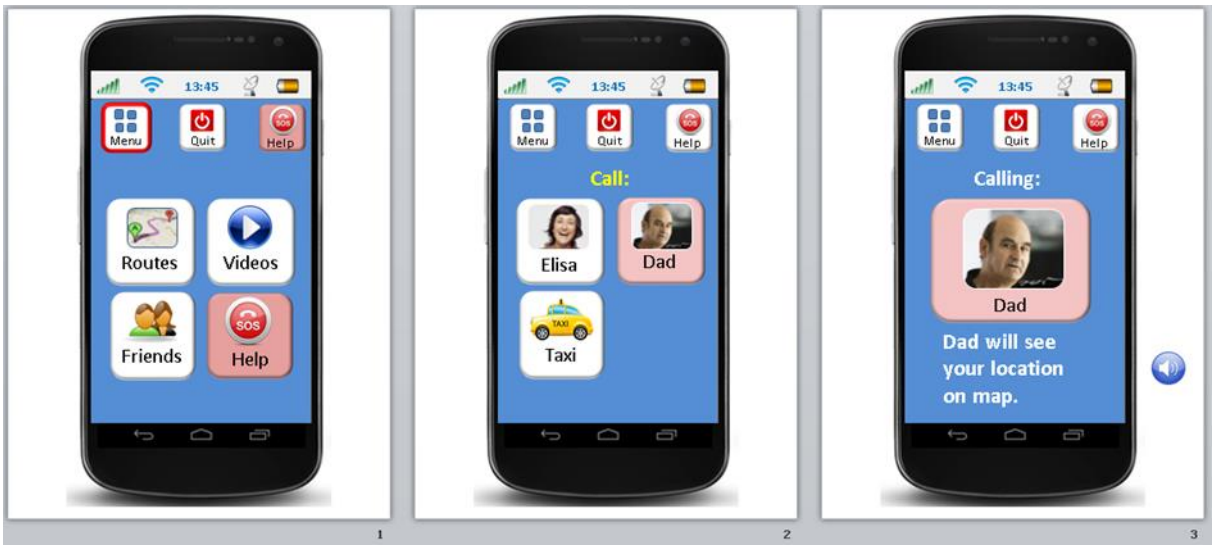
In the next part, the user wants navigation help. A list of destinations has been set up for the user to select from, and a route with navigation instructions is shown on a map. The user can send a message when reaching the destination.



This mock-up also shows the idea of playing back a video or sequence of images showing the way to the destination.

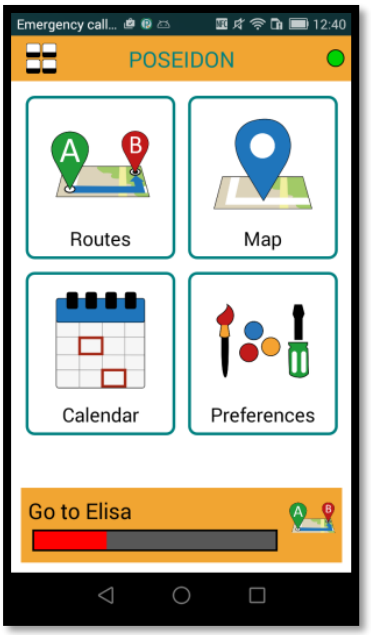


Finally, the app has a help function if the user requires further assistance. A phone call can be initiated, and the tracked location will be made available to the carer.



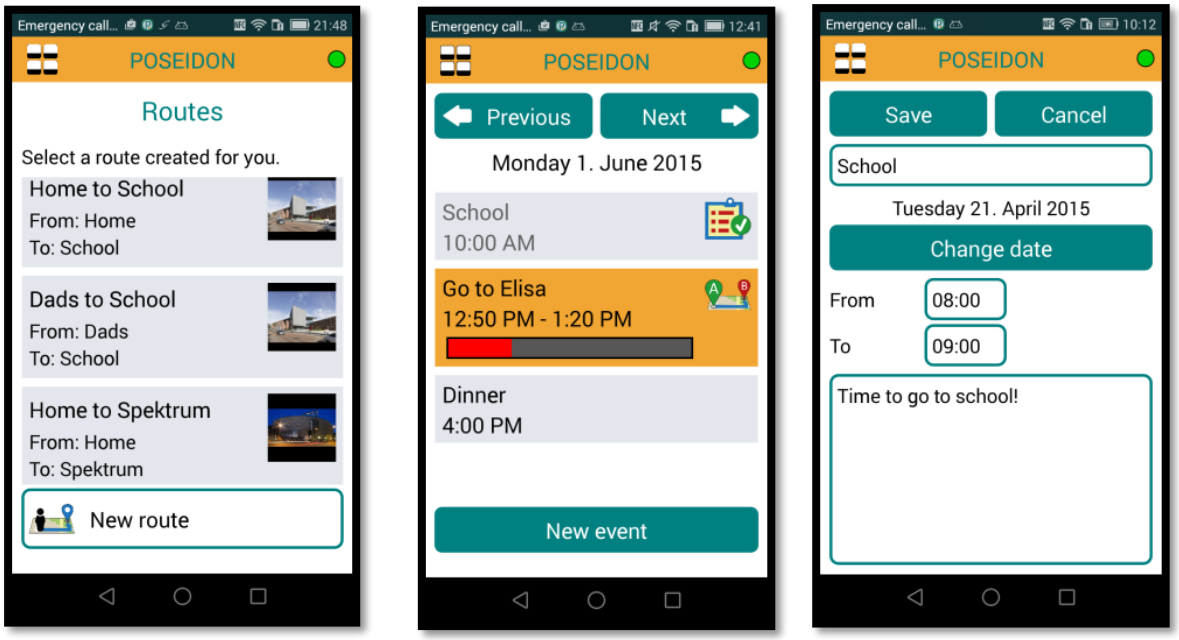
9.1.2 Implementation of POSEIDON App

A POSEIDON mobile app prototype was developed for the pilot, based on the early prototype demonstrated at the user workshop at the end of the first prototype iteration and on the presented paper prototypes. The main end-user functionalities are calendar with notifications and route navigation. More details regarding functionality of the app can be found in D5.2 Chapter 2. The primary goal with the mobile app pilot was to give the Down syndrome users access to this functionality, to assess their potential usefulness for this user group. The navigation is based on routes provided by Google and the calendar data is based on the simple model of events and reminders found in most online calendars. These data models are probably not ideal for our use, but their availability in existing online services as well as the relatively low level of effort required for carers to configure the data are important factors. We need to assess the data models to decide if the data which can be provided is sufficient or not, and for this it is important that the user interface isn't a bottleneck in the use of the pilot.



Making the user interface suitable for persons with Down syndrome has been the primary goal in the user interface design. It has been important to keep the user interface as simple as possible. For instance, we have tried to avoid scrolling, which can be challenging due to the size constraint of a phone screen as well as the variability of some of the content. The development has been done with feedback from the end user organisations, which has been especially important regarding how to present information in a way which is understandable for the user group. An example of this is the visualisation of the time left until the start of an appointment using a coloured bar, which was added on request from the end user organisations as they knew that just writing the number of minutes left wouldn't work. For easy of navigation in the app, a start menu icon is always present, to return to the first screen. In addition, the back button of the phone can always be used to backtrack, as is the normal convention. The app has a POSEIDON colour theme, and a

high contrast theme. All text is translated to all three project languages and icons are generally understood and adequate for adult users, not childish. These few images taken from D4.5 show that



scrolling could not be avoided everywhere, like in the routes outline view. Here vertical scrolling had to be implemented. In the calendar view, where primary users set their appointments adding the time is provided by a scrollable wheel, while typing should be done via the standard keyboard using the phone word suggestion. Additional to written text, the app alerts appointments by reading the appointment text out loud.

The app user interface is further described in deliverable D4.2 chapter 6, which has a focus on adaptivity. A technical description of the app is found in D5.2 chapter 5, while the technical user manual is found in D4.5 chapter 9.

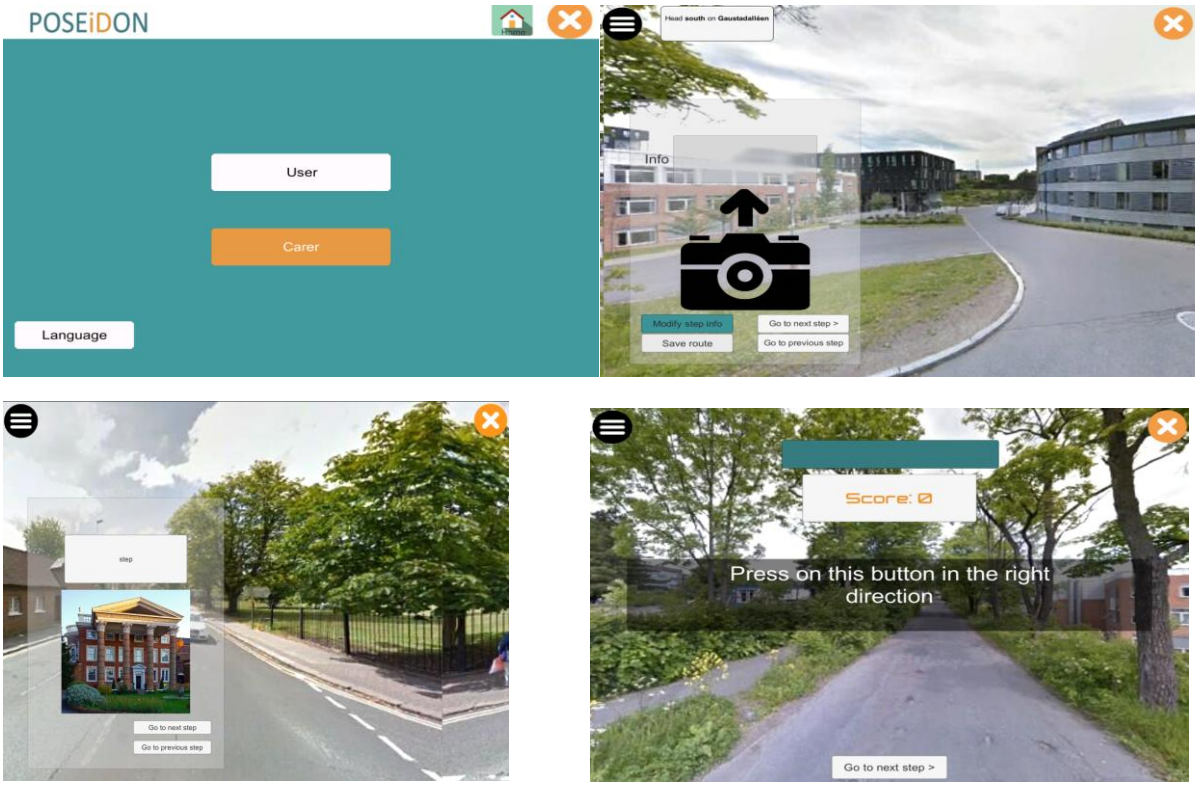
9.2 Home Navigation

A Home Navigation app prototype was developed for this pilot, based on the observation and feedback gathered after the second user workshop, at the end of the first prototype iteration. The product was designed having in mind different functionalities: it allows for route customization; it provides a platform for learning and rehearsing the details of a route.

The navigation is based on data provided by Google and the steps of a journey can be configured through custom information that can be added by the secondary users. The custom information consists of personal material (photos) or photos that exist in the Panoramio (www.panoramio.com) database, that are suggested based on user's location. Further details are provided in D4.4 Virtual Reality System and D4.5 HCI user and developer manuals.

The interface is simple and allows to both secondary and primary users to use the application. It was built as a result of various iterations with the involved users who made several recommendations about the interface design.

We show here an extract of some early screens form the Home Navigation System. One can observe the family resemblance with the POSEIDON colour scheme separating secondary user's navigation settings from primary user route rehearsal and training settings. A gaming approach was used to motivate primary users encouraging correct decisions with a higher score.



9.3 Moneyhandling Training with Interactive table

For pilot 1 we chose a learning application to improve the ability of people with Down syndrome to handle money. This was one of the most important domains where technology could support people with Down syndrome. More details on the interactive table and the Moneyhandling training can be found in D4.3 Interactive table and D5.2 Prototypic systems Chapter 7 and D4.5 HCI, user and developer manuals Chapter 5, 10.

To use the advantages of the interactive table to support the learning experience we developed an interaction strategy of combining screen visualization and real world interaction. The development and the rationale behind the learning application are explained in the following section.

9.3.1 Learning phases – of how to handle money

In the following the learning phase's children with learning disabilities need to acquire to be able to handle money in real-life are described. The main reference is "Lehrplan + Materialien für den Unterricht in der Schule für geistig Behinderte" vom Staatsinstitut für Schulpädagogik München". The success of learning one phase is a prerequisite for learning the next phase. The colours represent the learning goals touched by the different levels of the proposed money handling training application described in the next section.

Get to know coins and bills (train with real money)

- Name them
- Recognize
- Recognize specific attributes (shape, colour, size, material, text, image)
- Sort
- Recognize money symbols, on screens

Know value of coins and bills (relation value to object, why do we save)

- Connect objects/products of a certain value to the value of bills and coins
- Sort coins and bills according to their value
- Practice saving

- Compare price tags of products (cheap – expensive)
- Work to earn money

Change money

- Value of money puts the amount in relation to the number of coins and the value of the coin/bill. Take examples of daily life.
- Change 2 cent to 1 cent coin
- Change 5 cent to 1 cent coin
- Change 10 cent to 1 or 2 or 5 cent coin
- Change 10 cent to 1€ coin
- Change 10 cent to 50 cent coin
- Change 1€ to 50 cent
- Change 2€ to 1€ coins
- Change 5€ to 1€ coins …

Read & give right amount of money (understand comma and zero)

- Read and give with amounts of money form 1-100 cents, ex. 55 cent
- Read and give with amounts of money in €, without cents, ex. 2 €
- Read and give with full amounts and cent amounts, ex. 2,30 €

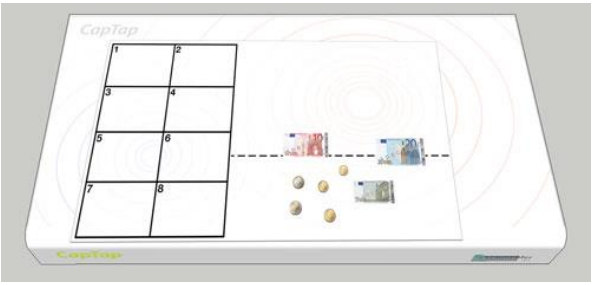
Pay with money

- Decide if money is enough to buy product
- Pay with exact amount of money
- Change money and pay with exact amount
- Pay with larger amount and await rest money
- Pay with larger amount and know how much was payed too much

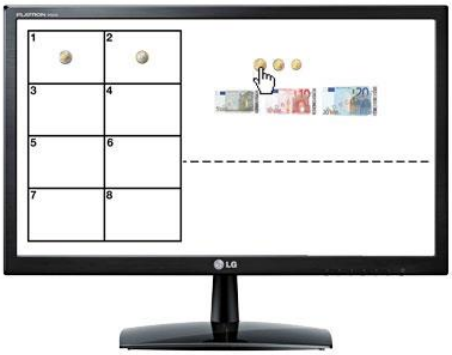
9.3.2 Paper prototypes for money handling application with CapTap

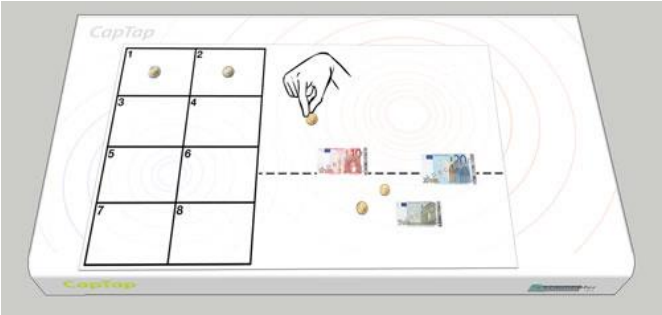
Step 1: Initialize application

The person with Down syndrome sits down in front of a screen and has the capacitive table in from of him. The learning application has been started. On the table a mat overlay is placed with the same basic layout as on the screen. Real money is already on the table.



The application asks the user to sort the money in the “boxes”. This could be done by asking the user to search for the first coin, in the first box and put it as described on the screen in the first box. This layout could be fix or configurable.





Levels of learning addressed with this step:

Get to know coins and bills (train with real money)

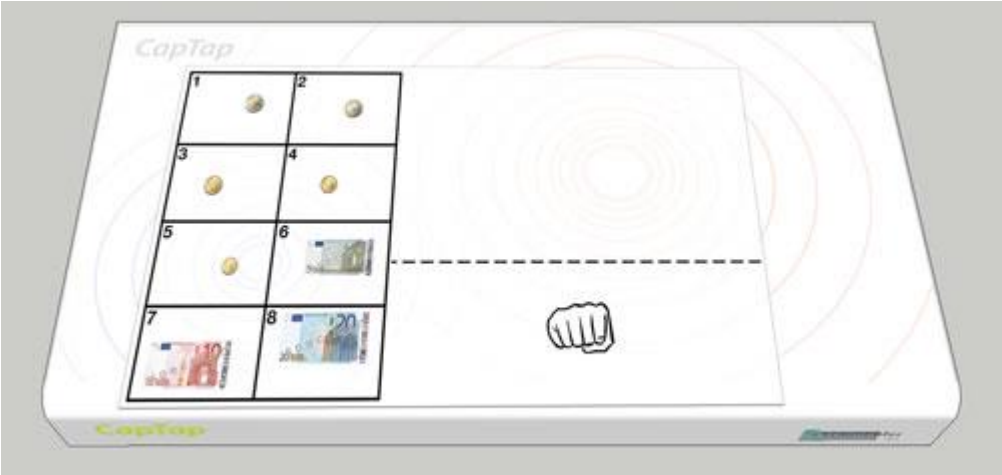
- Name them
- Recognize
- Recognize specific attributes (Shape, colour, size, material, text, image)
- Sort
- Recognize money symbols, on screens

Step 2: Play version 1: Pay for everyday products

Now the screen shows the sorted money and the sorting on the table is done.

In the next step a product appears with a price tag underneath. The program asks the user to select the right amount of money by knocking (gesture variable) on the table in the desired box and then in the area underneath the product. Then the selected money appears on the screen.

Starting screen of playful learning



Levels of learning addressed with this version 1:

Pay with money

- Pay with exact amount of money

Read & give right amount of money (understand comma and zero)

- Read and give with amounts of money in €, without cents, ex. 2 €

Variations of this level are imaginable. For example:

- Ask the user how much he would pay for the product?
- Ask the user to choose between two price tags, which is the appropriate one
- Have price tags with commas and pay with fix amount
- Some coins are not available. He has to pay and get change.

The variations address some further training steps:

Know value of coins and bills (relation value to object, why do we save)

- Connect objects/products of a certain value to the value of bills and coins
- Compare price tags of products (cheap – expensive)

Pay with money

- Change money and pay with exact amount
- Pay with larger amount and await rest money

Ideas for personalization and customization:

As an extension of this version the products and the price tag could be customized for the user. Also, the range of numbers could be set in the game.

The three currencies of UK, Norway and Germany are going to be implemented, as well as the languages.

If there would be text, the size of the text should be adaptable. Special help like a voice reading the text out loud (customizable btw. woman, men, carers voice) and the word syllables lightening up as the word is read. If not needed the function could be deactivated.

9.3.3 Overlay development

Here we show how the initial idea of having the money on the screen in boxes and moving it around has generated the idea to have an overlay placed on top of the table, where finally real money can be placed and moved around.

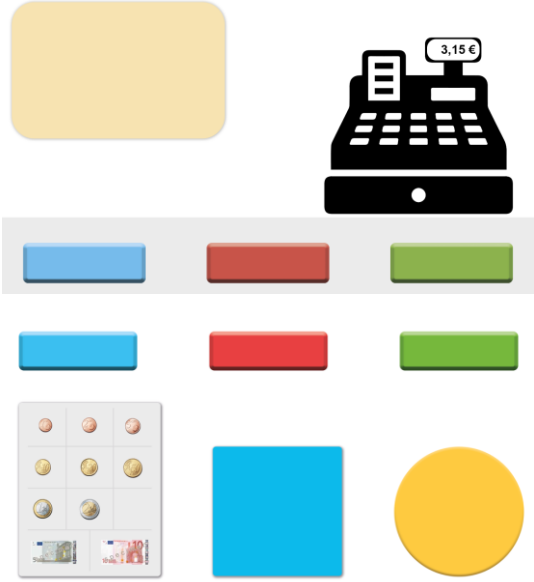


Figure 17 First mock-ups of the Moneyhandling screen and the interactive table overlay

Figure 18 Learning application screenshot and underneath the overlay of the interactive table. The colored fields activate the buttons on the screen.

The result is, that the money display has been moved in front of the user and the table recognizes the field where the hand is choosing a coin or bill. The user can acknowledge the selection of the coin or bill by taping or knocking.

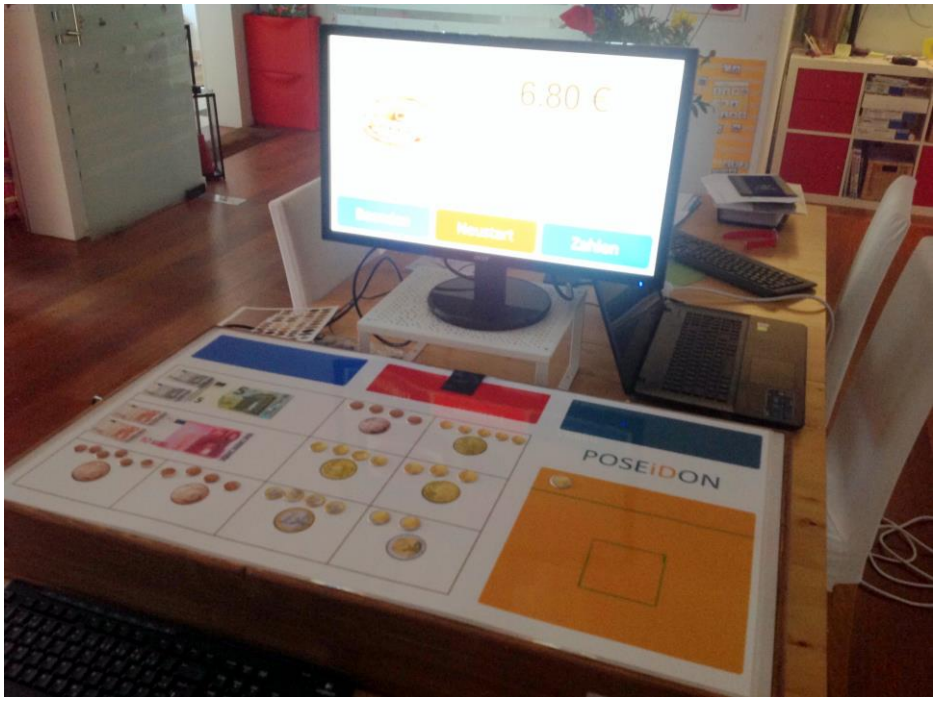


Figure 19 Interactive table with overlay and additional coins in front of screen of the application

9.4 POSEIDON web

Pilot 1 proved the usability and usefulness of the POSEIDON web. We also considered the users' learnability, efficiency in using the web system, i.e. we made the system easy to be used. One important aspect here is that complex interactions were avoided. Moreover, the software pilot environments was designed to reveal the most crucial usability, usefulness and accessibility aspects to be implemented in Pilot 2.

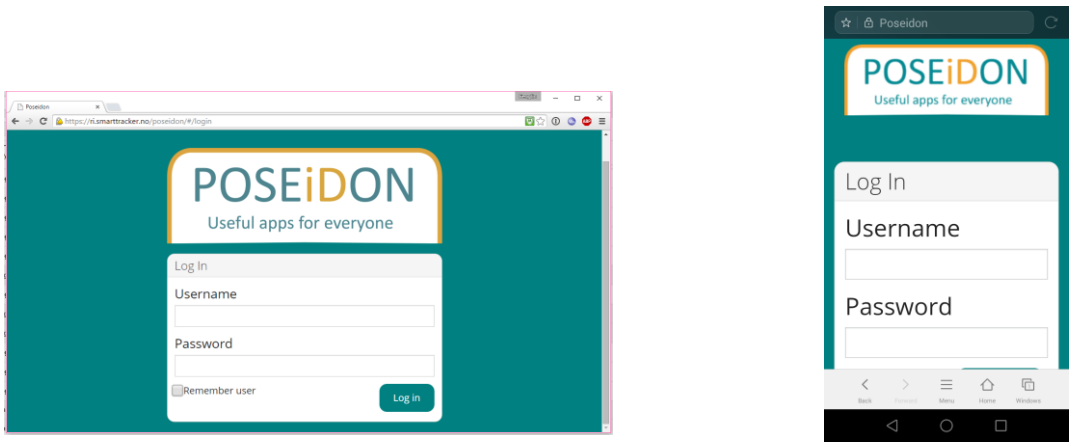


Figure 20 Responsive design of POSEIDON Web for PC and Smartphone

In the following we show several examples of how the user interface guidelines have been implemented in the POSEIDON Web.

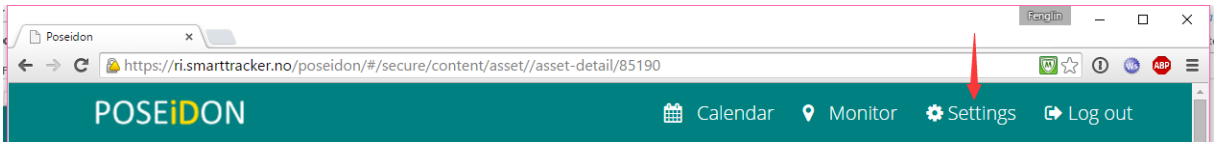


Figure 21 Shallow navigation levels

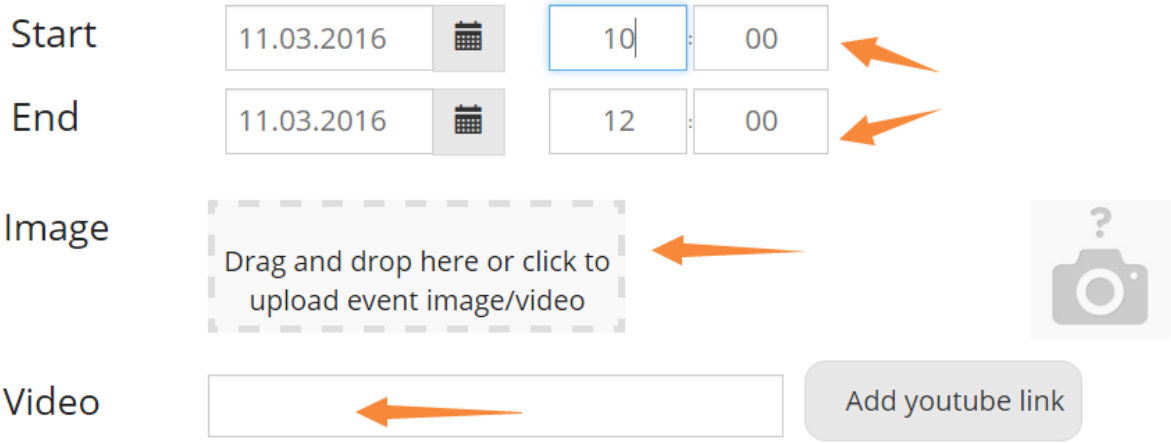


Figure 22 Easy input

9.5 Overarching interface aspects

For the past POSEIDON development period, we have put the accessibility and unified design of POSEIDON system as the first-class consideration, e.g. the unified logo, slogan, web system colour including good contrasts, help texts and large fonts. An important design perspective was to design a system that enables several essential accessibility features for POSEIDON's target group (persons with Down syndrome, carers and tertiary users). Already for pilot 1, a certain family resemblance between the POSEIDON apps has been both applied and enabled.

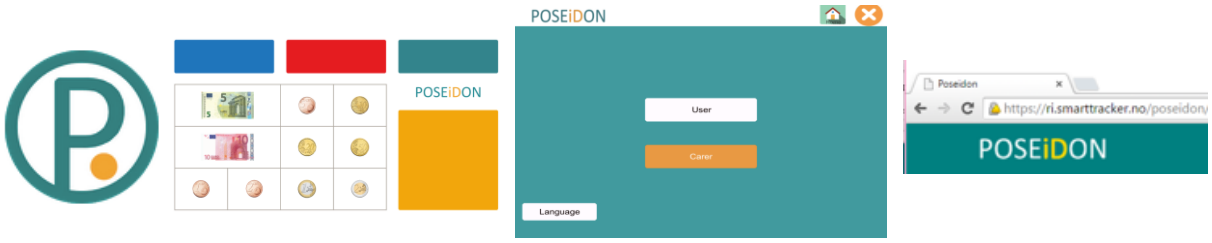


Figure 23 Icons and POSEIDON look-and-feel throughout the POSEIDON components

An important strategy was to design icons that are very easy to understand, nice (not silly, not complicated), and implemented in a graphics design style that allows the design to be continued in Pilot 2.

10 Pilot 2 - Achieved interface strategy

In this chapter, we provide some general comments for each of these POSEIDON components regarding the user interface strategy. The comments are feedback regarding the implementation from pilot 1. Furthermore, we show the improved interfaces and some interface design process of new PSOEIDON components.

10.1 POSEIDON App

Designing a user interface for people with Down syndrome is challenging, and we are sure there is more to learn and improve. Firstly, we got the results of the pilot 1 and analysed them to see how well the user interface worked. Any shortcomings were addressed, see Chapter 5.1 in D4.2 Adaptive tablet interface. New functionality was also developed for pilot 2, presented in D5.2 Prototypic systems.

10.1.1 Changes triggered by feedback from pilot 1

The screenshots below show interface changes in the POSEIDON App triggered by feedback from pilot 1. The image in the upper line represents the screenshot from pilot 1 and the image below the screenshot from pilot 2. In the case of the first image, which is the “Next activities view” the image from pilot 2 shows the rich POSEIDON icons base. Additionally, on request of the users, a button to browse to the newly implemented monthly calendar overview is placed in the upper buttons row.

The image in the middle shows the map view of the navigation function from the app. Here in the pilot 2 view we observe that the arrow in the middle which was very much error prone has been removed

and replaced by an arrow on the route pointing towards the desired direction of movement. The third image with its corresponding image underneath for pilot 2 shows the very much extended system status screen. To be able to debug better, this screen was used to identify status of media download, calendar connection and multiple other functions. The lights showed the quick overall status of the different parameters underneath.

10.1.2 Extract of additional functionality for pilot 1

Feedback from pilot 1 indicated that the inclusion of a monthly overview in the calendar functionality would be helpful. For pilot 2, this functionality was implemented and added. The concept was designed together with secondary users from the DSAs. The weekdays use special icons. For each pilot country, a set of weekday icons has been designed. Additionally, the time scale design was also altered using feedback from pilot 1. Initially it was designed as a progress bar, changing colour to red when little time as left. Because this did not have the wished impact everywhere, the linear bar concept changed from a bar to a circular bar, inspired by the shape of a clock. Here the time is running out.

In comparison to the POSEIDON app main screen from pilot 1, we now have 6 tiles instead of the previous 4. Three of these are completely new functionality including Video list, Moneyhandling training on the mobile phone (see Section 10.8), and the shopping assistance.

10.2 Shopping App

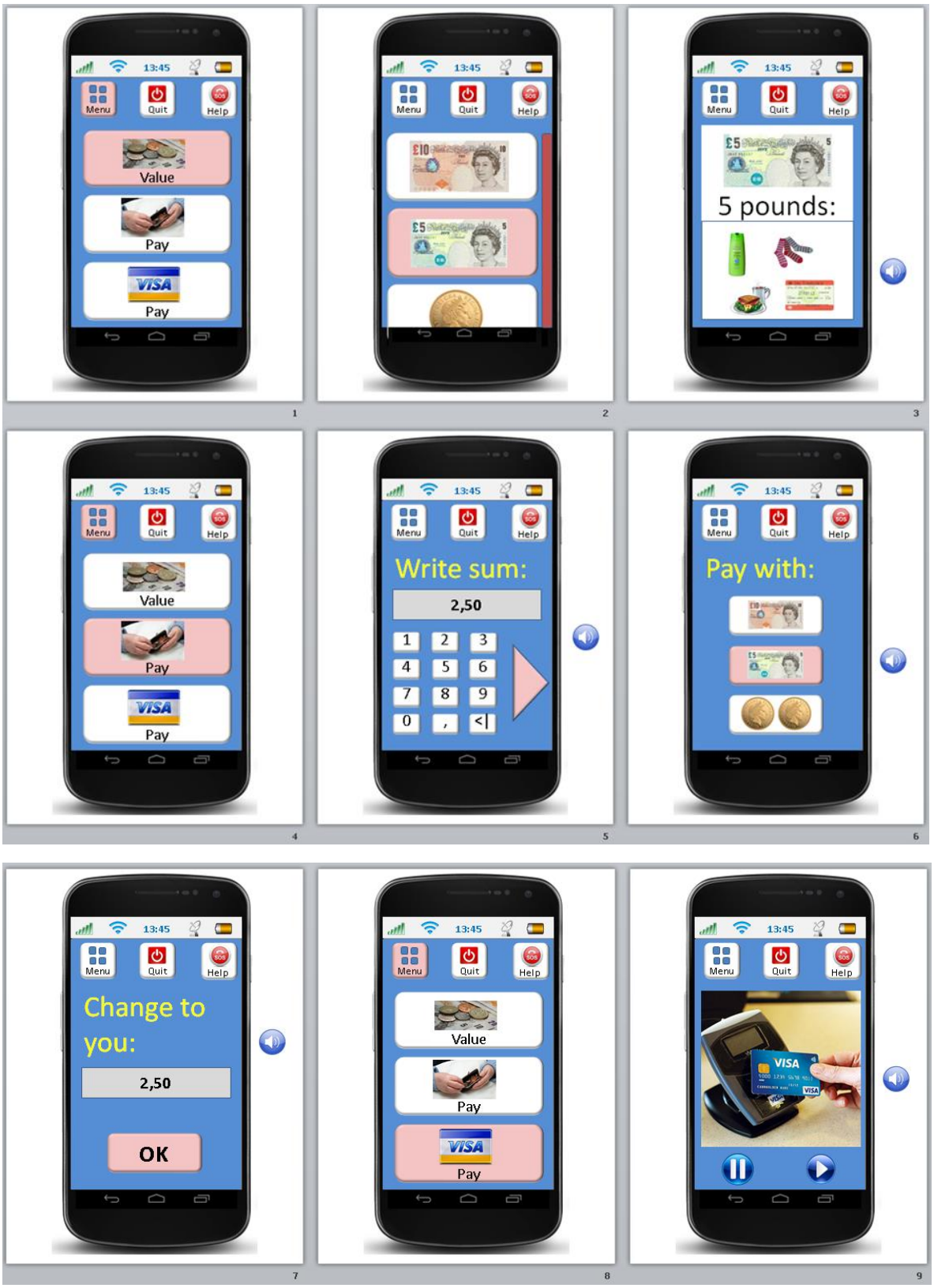
As suggested at the second review, more functionality should be added and integrated with the other POSEIDON components. Hence, the Shopping App was designed to extend the functionality of the Moneyhandling Training at home with the interactive table.

10.2.1 First paper prototypes

This application is meant to aid handling money and paying for things.

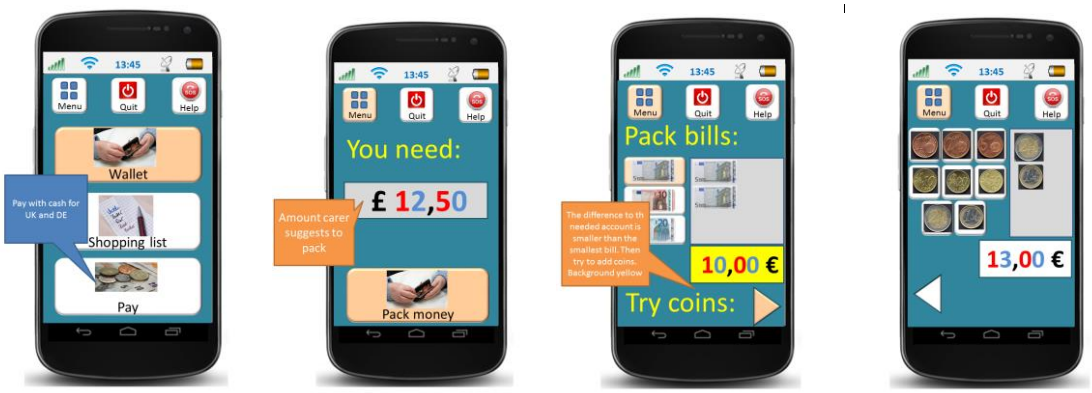
The “value”-function shows money of different values, and illustrates what you can buy with it.

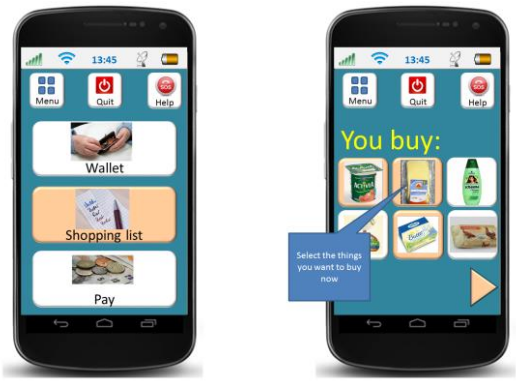
“Pay” by cash shows you what money you can use to pay a specific amount, and how much change you should get back, while “pay” by card can show instructions for this.



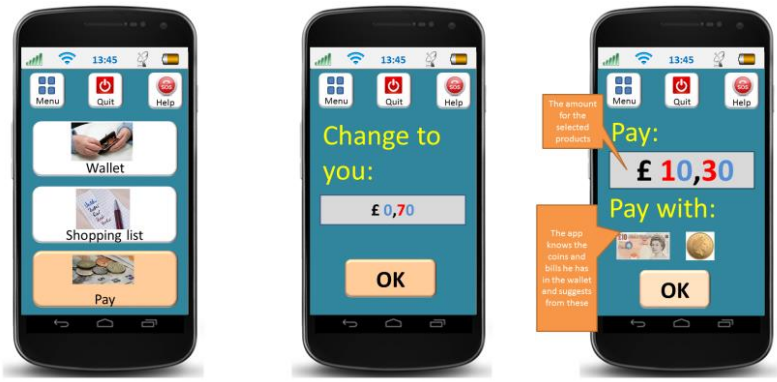
10.2.2 Refined paper prototypes

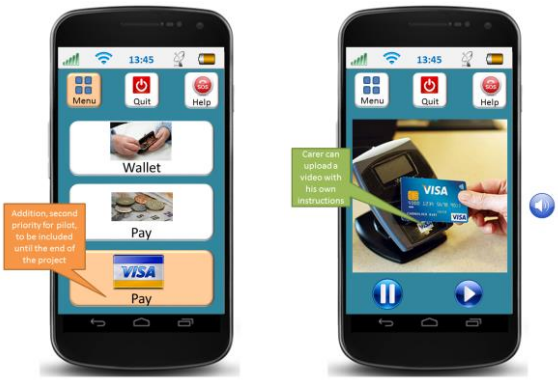
In the second iteration of paper prototypes, many aspects of the first iterations have been kept. However, there was a need expressed by the DSAs that a suitable app would need to know the content of the primary users' wallet to suggest the correct coins and bills when paying. Thus, an additional button "Wallet" has been foreseen. The app knows the value of the shopping list and guides the user to pack enough money to be able to pay for the goods.





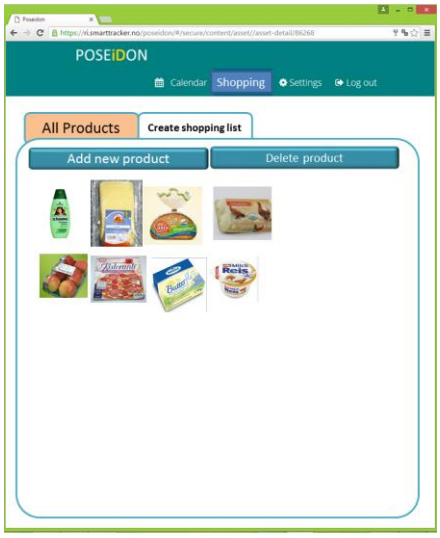
In the shop, the button “shoppinglist“ can be activated to see the products which have been added. After finishing one goes to the counter and pays with the suggested coins and bills, which are exactly the ones contained in the wallet.

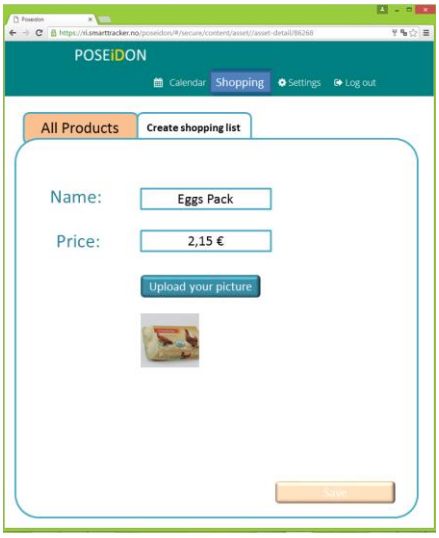


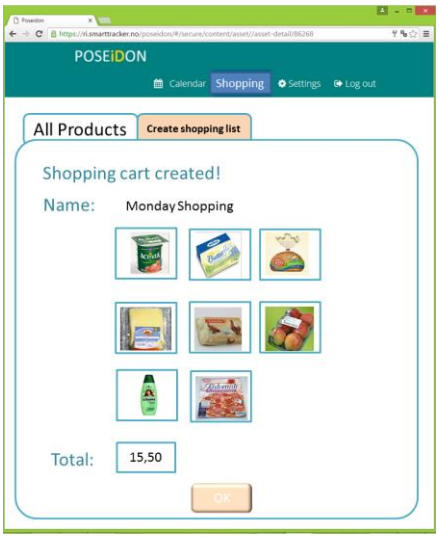


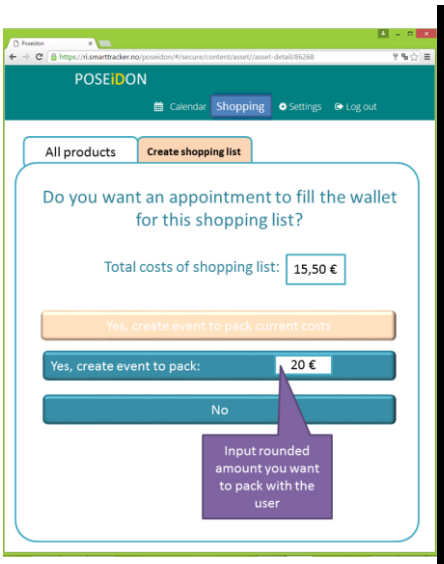
For Norway, because there paying by cash is not usual, instead of getting a paying suggestion an instructional video on how to pay with credit card is shown.

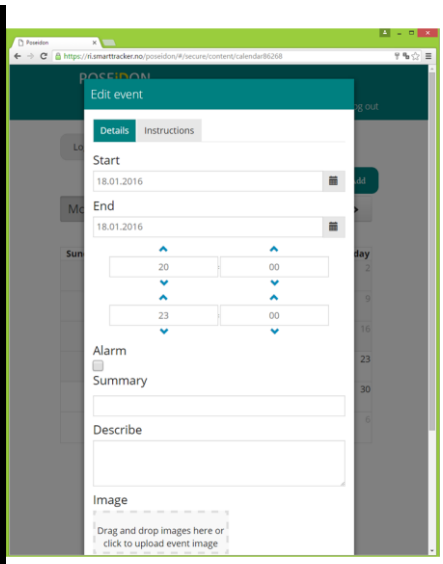
Because the phone needs to know the shopping list, the prices of the products and thus the price of the shopping cart, the secondary user must input this information to the POSEIDON Web. Therefore, paper prototypes have been also created.









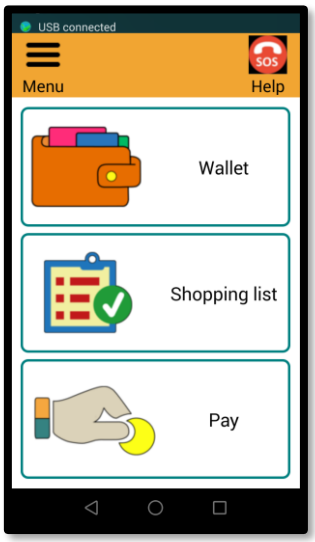


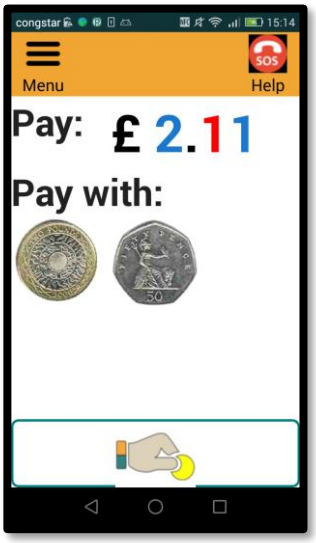
At first one must add products with names, price tag and upload the according picture. Doing this multiple times, a list of all products is created. A shopping list can then be created by clicking on the available products. It automatically calculates the shopping list total price. The secondary user can

specify a rounded sum, of how much the user should take with him and also create a calendar appointment for filling the wallet.

10.2.3 Shopping App for pilot 2

The implemented app interface is presented in D4.5 HCI user and developer manuals. However, we point at a specially designed feature for people with Down syndrome.





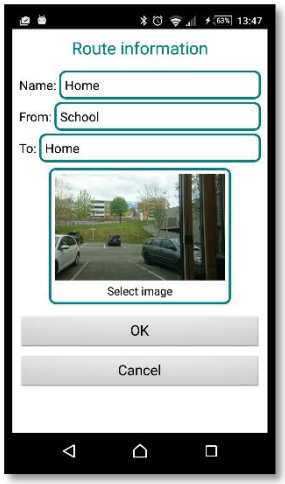
What we mean is the colour coded price tags. The colour code supports identifying the value of the digit, which makes it easier to read the number.

In general, the app is designed according to the POSEIDON interface design guidelines. We use icons wherever possible, reduce text, write it in a larger font and avoid scrolling where possible.

This app is not implemented yet with the high contrast colour scheme. It is not designed as part of the main POSEIDON App, but as an external app which must be installed separately. The usage however is fully integrated.

10.3 POSEIDON Route Creator App

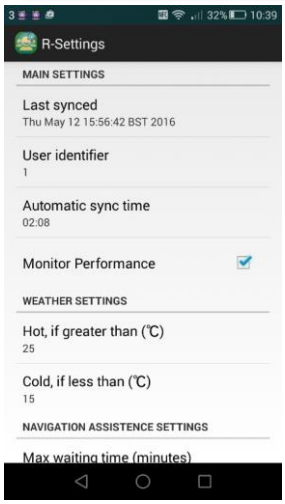
As a new component for pilot 2, the Route Creator App has been created. It is intended to support the secondary user to create personalised routes. These are customised with pictures, text instructions and recorded messages. At the point where the picture is taken, the GPS coordinates are used to mark this point on a map and create the route. More details regarding the functionality are presented in D5.2 Prototypic Systems and D4.5 HCI user and developer manuals Chapter 12.



The application is intuitive and easy use, eliminating the need to create routes indoors using the Home Navigation System.

10.4 POSEIDON Context Settings

The context reasoner which for pilot 1 was an app installed in the background has now a user interface for pilot 2. It is intended to be used by the secondary user. Here some parameters regarding the detectable contexts can be set. The context reasoner does not have the POSEIDON look and feel yet, however the settings have recently been integrated to the POSEIDON Web. It can be regarded like a future app which is developed for POSEIDON by an external interested developer.



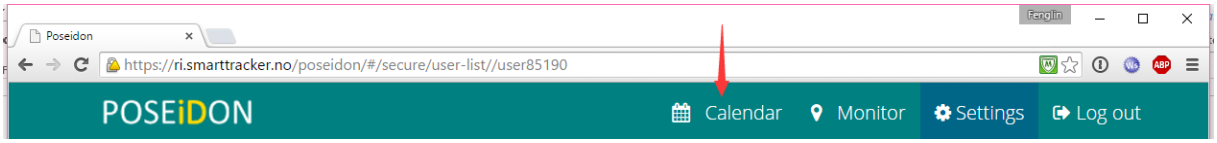
10.5 POSEIDON Web

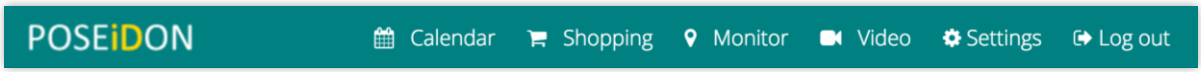
We improved the functional aspects of the POSEIDON Web by further simplifying graphics and usability.

Considering usability, we apply more drag and drop interaction in the calendar for easy reading, the fonts will be adapted to both mobile devices and PC-reader friendliness. Considering the end user's ability, we can include more functions in the POSEIDON web. One example of this is that the end user might be confused by complicated rules for creating multiple occurrence events. To avoid this, we use drag and drop for duplication of scheduled events. In the interaction design, we take advantage of drag and drop techniques and also apply touch screen interaction methods to make the POSEIDON Web more end user friendly and interaction friendly.

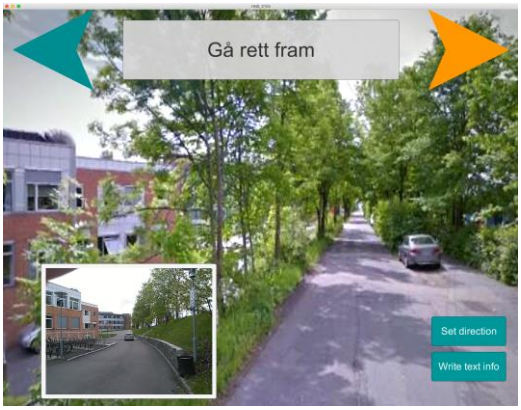
The POSEIDON Web interface is extended by creating the configuration for the Shopping and Video functionality on the smartphone. D4.5 HCI user and developer manuals shows the different interfaces.

For uploading an instruction video, drag and drop can be easily used. Additionally, a connection to YouTube videos is added. The Shopping functionality is inspired by the paper prototypes presented in Section 10.2.2.





10.6 Home Navigation System



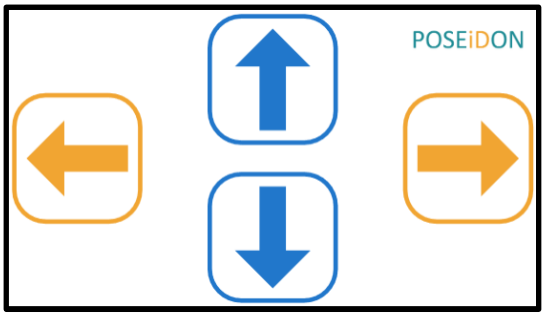
During pilot 1, we encountered several challenges and different aspects of the interface had to be modified according to users' recommendation.

In terms of suggested improvement for pilot 2, we were made aware of the fact that the system needs to support a higher degree of customisation of the route segments. This has been achieved by creating the specialised Route Creator App, solely for this purpose. However, the functionality to create routes at the home PC has been re-implemented consistent with the changes needed for a more customizable route.

In general, the interface has been improved. Choosing pictures from another folder in the PC has been made much easier.

In general, the app is more consistent with the POSEIDON guidelines.

For the full integration with the other POSEIDON components, the Home Navigation System has been connected to the interactive table. An overlay for navigating through the preconfigured route has been created. However, because the interactive table has not been user for pilot 2 no further adjustments were undertaken.

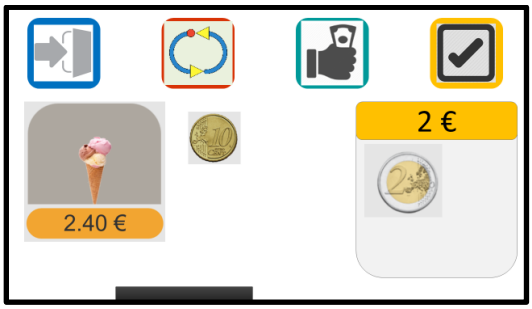


10.7 Moneyhandling Training with Interactive table

First trials of the Moneyhandling Application for pilot 1 have resulted in a few observations regarding the interface strategy. One feedback was regarding the size of the table and the second regarding the overlay of the table.

The pilot participants have already positively commented on the size of the table. The fact that the interaction area is so big (90 x 50 cm) seems to be extremely liked.

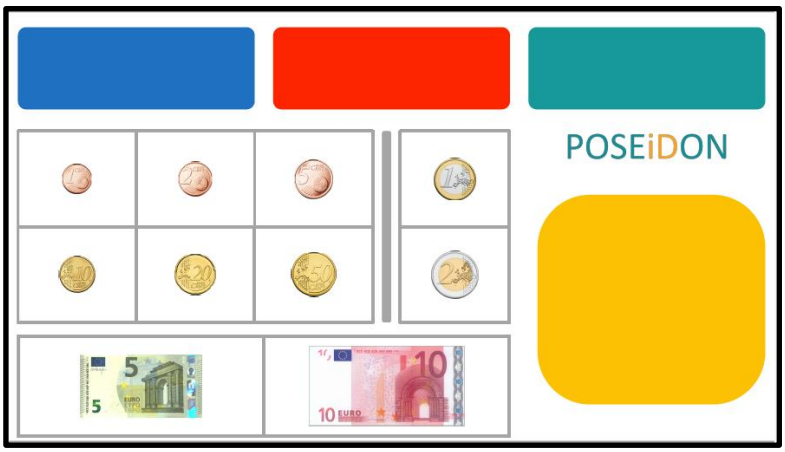
Families would prefer another order of the coins and bills on the overlay. However, it is not clear if they would agree to the same order, or this should be fully customisable. The reason for this seems to be the way people with Down syndrome learn to calculate and their difficulty to distinguish between whole coin value and cents. Thus, we have taken this feedback into account and tried to develop a new design which would make it easier to interact with the buttons further away. Some mock-ups of the finally not implemented ideas are shown here.



The adjusted overlay is shown here:

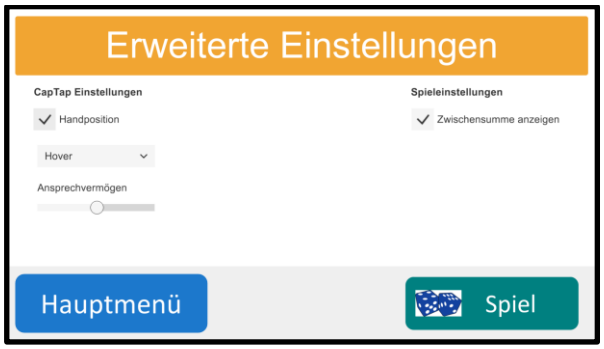
Characteristically are the now separated areas with coins. They enable easy differentiation between the coin types, grouping them together.

Additionally, the relative coin size has been adjusted.

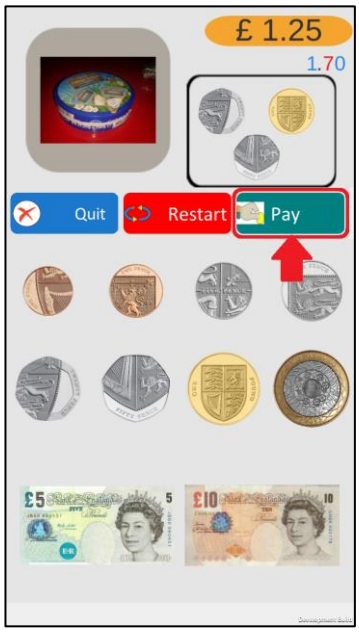


For pilot 2, the Money Handling App has gained from interface strategy point of view additional gestures. Hovering has been implemented and is used by holding the hand at close distance on top of the table (up to 15 cm) for a specific time interval which can be set. This process is shown by having a cursor on the screen in form of a hand which is filled while time elapses. Here we show a screenshot

of this hovering cursor and the personalization, settings interface. Here users can choose how to interact with the table, in mid-air or rather on the surface of the table or enable or disable summing aids. Further functionality is described in D4.4 Interactive Table.



10.8 Moneyhandling Training App



To make functionality independent of the device on which it is running, we have leveraged the capability of the development framework Unity to reuse the already developed gaming approach for Moneyhandling Training to create a smartphone app. The result is a slightly adjusted version of the Moneyhandling Training software used with the interactive table. D4.5 HCI user and developer manual presents the interface in Chapter 10.

For this app, we leverage the interface strategy used to develop the app. The idea was to have a palpable continuation of the screen by using the interactive table with the overlay. For the vertical alignment, how phones are usually hold this was beneficiary.

Because outcomes from the first and second pilot would be static, we embedded this app in pilot 2 instead of the app on PC and interactive table.

10.9 POSEIDON Social Network - Ecosystem

This POSEIDON component is web-based. It uses the Web-page service Wordpress to support three different functionalities: Information gathering with discussion forum, DSA websites and a personal social communication page. We use Wordpress with different plugins to enable the functionality and adapt the ecosystem to the POSEIDON colour themes.

Details are presented in D5.5 Social network and the user manual can be found in D4.5 HCI user and developer manuals Chapter 15.

11 Conclusion

After each encounter with the primary users, the user interface was refined. After implementing general guidelines, the details give the interface the very specific usability for persons with Down Syndrome. Such detailed examples are the time representation and the numbers representation used in our apps. Here we see that knowing about the learning strategy for the different domains is the decisive step to develop interface from which all user groups profit.

12 References

[1] "Web Content Accessibility Guidelines (WCAG) 2.0," 2015. [Online]. Available: http://www.w3.org/WAI/intro/wcag.php. [Accessed: 15-May-2015].

[2] World Wide Web Consortium, "Press Release: W3C Launches Web Accessibility Initiative," 1997. [Online]. Available: http://www.w3.org/Press/WAI-Launch.html. [Accessed: 15-May-2014].

[3] M. Schneider-Hufschmidt, U. Malinowski, and T. Kuhme, "Adaptive user interfaces: Principles and practice," 1993.

[4] P. Langley, "User Modeling in Adaptive Interfaces 1 The Need for Automated User Modeling."

[5] J. Feng, J. Lazar, L. Kumin, and A. Ozok, "Computer usage by children with down syndrome: Challenges and future research," *ACM Trans. Access. …*, 2010.

[6] J. Lazar, L. Kumin, and J. H. Feng, "Understanding the computer skills of adult expert users with down syndrome," in *The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility - ASSETS '11*, 2011, p. 51.

[7] E. Marcotte, "Responsive Web Design," *A List Apart*, 2010. [Online]. Available: http://alistapart.com/article/responsive-web-design. [Accessed: 13-May-2014].

[8] J. Nielsen, "Heuristic evaluation," *Usability Insp. methods*, vol. 24, p. 413, 1994.

[9] A. Kirijian, M. Myers, and S. Charland, "Web fun central: online learning tools for individuals with Down syndrome," *… Users. Chichester, UK John Wiley …*, 2007.

[10] Europe Inclusion, "Information for all," 2009.

[11] "The WCAG 2.0 Documents." [Online]. Available: http://www.w3.org/WAI/intro/wcag20. [Accessed: 15-May-2014].

[12] J. Guerrero-Garcia, J. M. Gonzalez-Calleros, J. Vanderdonckt, and J. Muoz-Arteaga, "A theoretical survey of user interface description languages: Preliminary results," in *Web Congress, 2009. LA-WEB'09. Latin American*, 2009, pp. 36–43.

[13] M. Pohja, "Comparison of common XML-based web user interface languages," *J. Web Eng.*, vol. 9, no. 2, pp. 95–115, Jun. 2010.

[14] S. Bongartz, Y. Jin, F. Paternò, J. Rett, C. Santoro, and L. D. Spano, "Adaptive User Interfaces for Smart Environments with the Support of Model-Based Languages," in *Ambient Intelligence*, Springer, 2012, pp. 33–48.