

SOCIABLE DELIVERABLE D3.1

“SOCIABLE Services Specification”



Project Acronym	SOCIABLE
Grant Agreement No.	238891
Project Title	Motivating platform for elderly networking, Mental reinforcement and social interaction
Deliverable Reference Number	SOCIABLE_WP3_D3.1
Deliverable Title	“SOCIABLE Services Specification”
Revision Number	V1.2
Deliverable Editor(s)	Singularlogic (SLG)
Authors	Stelios Pantelopoulos (SLG), Kostas Kalaboukas (SLG), Paolo Mattarelli (CEDAF), Michele Fabbri (CEDAF), Irene Zaragozá Álvarez (UPV), Miriam Morante (AIJU), Rocío Zaragoza (AIJU)

Project co-funded by the European Commission within the ICT Policy Support Programme		
Dissemination Level		
P	Public	P

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

PROPRIETARY RIGHTS STATEMENT

This document contains information, which is proprietary to the SOCIABLE Consortium. Neither this document nor the information contained herein shall be used, duplicated or communicated by any means to any third party, in whole or in parts, except with prior written consent of the SOCIABLE consortium

Revision History

Revision	Author(s)	Organization(s)	Date	Changes
0.1	Stelios Pantelopoulos, Kostas Kalaboukas	SLG	08/07/2010	Provision of Structure and Table of Contents
0.2	Stelios Pantelopoulos	SLG	20/09/2010	Addition of games specifications
0.3	Stelios Pantelopoulos,	SLG	29/09/2010	Description of the SOCIABLE care centre architecture
0.4	Stelios Pantelopoulos, Kostas Kalaboukas	SLG	04/10/2010	Description of the SOCIABLE implementation technologies and services design guidelines
0.5	Irene Zaragozá Álvarez	UPV	15/10/2010	Specifications of Social Activation Services
0.6	Paolo Mattarelli	CEDAF	15/10/2010	Technical Specification of Back-Office Services and Services to the health professionals
0.7	Rocío Zaragoza, Miriam Morante	AIJU	20/10/2010	Technical Specification and Design of the Cognitive Games
0.8	Stelios Pantelopoulos, Kostas Kalaboukas	SLG	29/10/2010	Revision of the services design guidelines
0.9	Rocío Zaragoza, Miriam Morante	AIJU	04/11/2010	Additional information on the games specifications
1.0	Stelios Pantelopoulos, Kostas Kalaboukas	SLG	15/11/2010	Editing of conclusions and executive summary
1.1	Stelios Pantelopoulos, Kostas Kalaboukas	SLG	17/11/2010	Matching of games with design principles, fine-tuning of the document
1.2	Stelios Pantelopoulos	SLG	03/12/2010	Incorporation of Quality Review Comments from FSL (Roberta Annicchiarico)

PROPRIETARY RIGHTS STATEMENT

This document contains information, which is proprietary to the SOCIABLE Consortium. Neither this document nor the information contained herein shall be used, duplicated or communicated by any means to any third party, in whole or in parts, except with prior written consent of the SOCIABLE consortium

Abstract

The present deliverable presents technical specifications of the SOCIABLE services, including services offered to the elderly (notably games and the book-of-life), as well as services offered to health professional (notably back-office services). The technical specifications of these services are presented on the basis of the general technical principles and guidelines that will be followed for implementing SOCIABLE services, but also in terms of the more specific characteristics of the SOCIABLE services (e.g., games workflows, back-office applications workflows, book-of-life implementation workflows). The main technical principles and guidelines outlined above stem from the nature and functionalities of the SOCIABLE surface computing platforms, and aim at maximizing the potential of the surface computing paradigm for the purposes of the SOCIABLE project. At the same time the more specific characteristics of the services, stem from the technical analysis of the definition of the SOCIABLE services (within WP2). This analysis has taken place in order to facilitate the later implementation and integration of the services in the SOCIABLE platform (as part of WP4). It should be noted that the list of eleven games presented in the deliverable is non-exhaustive; it is envisaged that the SOCIABLE platform will be gradually augmented with additional games, towards the ultimate target of twenty-seven games (defined in D2.2 of the project). Note however that additional games will also follow the main technological design and implementation guidelines provided in this deliverable.

PROPRIETARY RIGHTS STATEMENT

This document contains information, which is proprietary to the SOCIABLE Consortium. Neither this document nor the information contained herein shall be used, duplicated or communicated by any means to any third party, in whole or in parts, except with prior written consent of the SOCIABLE consortium

Table of Contents

Revision History	2
Abstract.....	3
Table of Contents.....	4
List of Figures	6
Executive Summary.....	8
1. Introduction	10
2. Overview of SOCIABLE Technical Architecture, Deployment Model and Implementation Technologies	12
2.1 Technical Architecture and Deployment Model.....	12
2.2 Specification of SOCIABLE Implementation Technologies.....	14
2.2.1 Microsoft Surface SDK	16
2.2.1.1 Overview	16
2.2.1.2 Visual Interaction	17
2.2.1.3 Tagged Object Recognition	18
2.2.1.4 Compatibility with Other Technologies	18
2.2.2 Windows Presentation Framework (WPF)	18
2.2.3 Surface Controls.....	19
2.2.4 Sample Microsoft Surface Components and Applications	20
2.2.5 The C# Programming Language	20
2.2.6 AJAX.....	20
2.2.7 SQL Server	20
2.2.8 Web Services.....	21
3 Surface Application Design Specifications and Interaction Guidelines	22
3.1 Spatial Interaction and 3D Environments.....	22
3.2 Super-Reality.....	22
3.3 Context-Awareness	22
3.4 Touch and Direct Manipulation	23
3.5 Virtual Objects like Physical Objects.....	23
3.6 Use of Tagged Objects	23
3.7 Use of UnTagged Objects.....	24
3.8 360° degree User Interface	24
3.9 Playful Pleasurable and Explanatory Touches	24
3.10 Progressive Disclosure	25
3.11 Multiple Senses.....	25
3.12 Continuous Input	25
3.13 The content should (as much as possible) be the interface	25
3.14 Integration with the Surface Shell	25
4. Technical Specification of Cognitive Training Games	26
4.1 Game #1- "Hide and Seek"	26
4.1.1 Detailed Specification	26
4.1.2 Relevant Surface Application Design Guidelines	27
4.1.3 Sample User Interfaces	27

4.2	Game #2 - Puzzles”	28
4.2.1	Detailed Specification	28
4.2.2	Relevant the Surface Application Design Guidelines	29
4.2.3	Sample User Interfaces	29
4.3	Game #3 - Analogies	29
4.3.1	Detailed Specification	29
4.3.2	Relevant Surface Application Design Guidelines	34
4.3.3	Sample User Interfaces	34
4.4	Game #4 – “Lost In Migration” (or “Lost in the City”)	35
4.4.1	Detailed Specification	35
4.4.2	Adoption of the Surface Application Design Guidelines.....	35
4.4.3	Sample User Interfaces	35
4.5	Game #5 – My Home	36
4.5.1	Detailed Specification	36
4.5.2	Adoption of the Surface Application Design Guidelines.....	36
4.5.3	Sample User Interfaces	37
4.6	Games #6 and #7 – Synonyms and Antonyms (respectively)	37
4.6.1	Detailed Specification	37
4.6.2	Relevant Surface Application Design Guidelines	38
4.6.3	Sample User Interfaces	39
4.7	Games #8 – Guess Who	39
4.7.1	Detailed Specification	39
4.7.2	Relevant Surface Application Design Guidelines	40
4.7.3	Interfaces to the Platform.....	40
4.8	Games #9 Par and Par Maxis	41
4.8.1	Detailed Specification	41
4.8.2	Relevant Surface Application Design Guidelines	41
4.8.3	Interfaces to the Platform.....	41
4.9	Games #10 Take Away Menu.....	42
4.9.1	Detailed Specification	42
4.9.2	Relevant Surface Application Design Guidelines	42
4.9.3	Interfaces to the Platform.....	43
4.10	Game #11 – Scrambled Story.....	43
4.10.1	Detailed Specification	43
4.10.2	Relevant Surface Application Design Guidelines	44
4.10.3	Interfaces to the Platform.....	44
4.11	Baseline Workflow of the Games	44
4.12	Specification of Additional Games.....	45
5.	Technical Specification of Social Interaction Services	47
5.1	Technical Specification of the “Book-of-Life”	47
5.1.1	Short Description	47
5.1.2	Workflow.....	47
5.1.3	Specification of Implementation Technologies and Interfaces to the Platform	48
5.2	Technical Specification of the Communication Services	49

5.2.1	Short Description	49
5.2.2	Workflow.....	49
5.2.3	Specification of Implementation Technologies and Interfaces to the Platform	50
6.	Technical Specification of SOCIABLE Services to Health Professionals	51
6.1	Configuration of Cognitive Training Services.....	51
6.1.1	Short Description	51
6.1.2	Workflow.....	52
6.1.3	Specification of Implementation Technologies and Interfaces to the Platform	54
6.2	Configuration of Social Activation Services	54
6.2.1	Short Description	54
6.2.2	Workflow.....	55
6.3	Monitoring of the Elderly Record (Cognitive Performance, Cognitive / Functional Assessment)	55
7.	Conclusions	56
8.	References	57
	Appendix – Workflows of the Cognitive Games Specified in the paragraph Technical Specification of Cognitive Training Games	58

List of Figures

Figure 1: Components Comprising the SOCIABLE Care Centre Technical Infrastructure	13
Figure 2: SOCIABLE Deployment Model – Multi-Domain / Across Care Centers	14
Figure 3: Overview of SOCIABLE implementation technologies.....	16
Figure 4: Screenshots of the “Hide and Seek” game (user interface design).....	28
Figure 5: Screenshots of the “Puzzles” game (user interface design).....	29
Figure 6: Screen shots of the “Analogies” Cognitive Training Game.....	34
Figure 7: Screenshots of the “Lost in Migration” Game	36
Figure 8: Screenshots of the “My Home” Game.....	37
Figure 9: Screenshots of the “Synonyms and Antonyms” Game	39
Figure 10: Screenshots of the “Guess Who” Game	40
Figure 11: Screenshots of the “Par and Par Max” Game.....	41
Figure 12: Screenshots for the “Take Away Menu” Game	43
Figure 13: Screenshots for the “Scrambled Story” Game.....	44
Figure 14: Workflow of the Hide and Seek Games.....	45
Figure 15: Workflow of the “Book-of-Life” application	48
Figure 16: Workflow of SOCIABLE Communication Services.....	50
Figure 17: Sequence Diagram Associated with the Creation of a SOCIABLE Cognitive Training Programmes.....	53
Figure 18: Component Diagram depicting the components of the Back-office applications offering services to the professionals	54
Figure 19: Workflow of the “Puzzles” Game	58
Figure 20: Workflow for the “Analogies” Cognitive Training Game.....	59

Figure 21: Workflow of the "Lost in Migration" Game.....	59
Figure 22: Workflow of the "My Home" Game	60
Figure 23: Workflow for "Synonyms and Antonyms" Cognitive Training Game	60
Figure 24: Workflow of the "Guess Who" Game.....	61
Figure 25: Workflow for the Par and Par Maxis Game	61
Figure 26: Workflow for the "Take Away Menu" Game.....	62
Figure 27: Workflow of the "Scrambled Story" Game.....	62

Executive Summary

The present document has been produced as part of work performed in the scope of WP3 (titled: "Services Specification") of the SOCIABLE project. It is devoted to the technical and technological specifications of the main SOCIABLE services comprising cognitive training games, the back-office applications, as well as the book-of-life. These services have been thoroughly defined in the scope of WP2 of the project. Hence, this deliverable is a natural evolution of the work done in WP2, given that it provides technical specifications enabling the later implementation of these services on the SOCIABLE platform.

The technical/technological specifications of the SOCIABLE services include:

- The overall technical specification of the structuring principles that drive the integration of SOCIABLE services within the platform. These structuring principles comprise the technical/technological architecture of the SOCIABLE care centre, which foresees that services are hosted and executed through the surface table and/or the surface laptop devices. These services access the care centre database.
- The implementation technologies of the SOCIABLE services, which include programming languages (i.e. C# as part of the MS. NET), Relational database management technologies (i.e. SQL Server), surface computing components and APIs (notably the Microsoft Surface SDK and its components), presentation layer technologies (notably AJAX and the Windows Presentation Framework), as well as integration technologies (notably Web Services). The pool of these technologies is compatible with the SOCIABLE technical architecture and hardware platforms. The later compatibility justifies the selection and orientation towards several Microsoft technologies.
- The technical specifications of the surface features to be used for implementing the services. These surface features are essential towards making the services motivating, ergonomic and pleasant to the end-user. To this end, best practices and guidelines of the surface vendor have been exploited.
- The specification of the detailed functionality of games, back-office applications and the book-of-life. This specification includes the main workflows of these applications, which will be used for implementing them. Note that the SOCIABLE games feature different scenarios and functionalities, which are not exhaustively presented as part of this deliverable. However, the deliverable presents the main (baseline) game workflow which is replicated across most of the SOCIABLE games, towards a uniform structuring of all the SOCIABLE games.

Overall, the above specifications constitute valuable input to service developers and integrators, which are working (as part of WP4) on the integration of the services on the SOCIABLE platform. In the scope of WP4 SOCIABLE adopts an iterative and evolutionary development approach, which emphasizes continuous improvement through multiple releases of the SOCIABLE platform and services. Due to this approach the present deliverable can be considered as a catalogue of the baseline and initial specification of the SOCIABLE services especially in terms of aspects (such

as application workflows) that are subject to change in future revisions of the games. Later WP4 deliverables, including the implementation of the prototypes might introduce changes/deviations to the games and book-of-life workflows. However, the SOCIABLE technical architecture and implementation technologies (presented in this document) can be regarded as more stable points of technological specifications.

Note that this deliverable represents the specifications of one of the two main pillars of the SOCIABLE pilot systems, namely the SOCIABLE services. The other pillar is the SOCIABLE platform, where these services are deployed. The SOCIABLE platform is specified (from a technical/technological perspective) in deliverable D3.2, which can be considered as a complementary document to the present one.

1. Introduction

SOCIABLE is piloting and evaluating a novel ICT based paradigm for cognitive training and social activation, which hinges on the deployment of cognitive games and social interaction applications over a novel surface computing platform. The services offered by the SOCIABLE platform are described in detailed in Deliverable D2.2 (titled "SOCIABLE services definition") and include:

- A wide range of cognitive games, which target a variety of cognitive skills for the elderly according to the SOCIABLE cognitive skills classification.
- A set of application facilitating social interaction between the elderly, as well as between the elderly and their family and friends.
- The "Book-of-Life" application, allowing older adults to create their own biography in a way that boosts experience sharing with other elderly individuals.
- Services supporting the professionals in creating, designing and customizing SOCIABLE cognitive training and social activation sessions, while at the same time monitoring the elderly performance and cognitive assessment.

From a technical viewpoint, the above services are designed and integrated over the Microsoft Surface platform. To this end, the Microsoft Surface SDK (Software Development Kit) is exploited based on the C# language. The design and technical integration of the services is based on the SOCIABLE technical architecture, which defines the structuring principles of the SOCIABLE platform and services. In line with this architecture the above-mentioned applications are designed as extensible modules that augment the SOCIABLE surface computing platform. This extensible design boosts the expandability of the SOCIABLE platform, facilitates the bundling of the SOCIABLE services (according to the SOCIABLE exploitation planning), while at the same time facilitating technological longevity.

The purpose of the present deliverable is to illustrate the technical specifications of the various SOCIABLE applications, in terms of their implementation technology, their interfacing to the SOCIABLE platform, their interworking and interactions, as well as their extensible deployment. Hence, the present deliverable complements and evolves deliverable D2.2, through adding the technical/technological viewpoint. Indeed, while D2.2 has been focusing on the services offered by the SOCIABLE platform, the present deliverable emphasized on how these services will be technically implemented over the selected surface computing platform. Therefore the deliverable prescribes the technical architecture of the SOCIABLE ICT systems and services, the implementation technologies to be used, as well as the main functionalities to be implemented through the selected technologies.

It should be noted that the present deliverable falls in the scope of WP3 of the project, which focuses on the technical specification of the SOCIABLE platform and services. While the present deliverable deals with the SOCIABLE services, deliverable D3.2 (which is developed in parallel) focuses on the SOCIABLE platform i.e. the

infrastructure enables the deployment and execution of the services. The SOCIABLE platform infrastructure provides the surface hardware required, along with a collection of middleware libraries that allow the modular and extensible deployment of SOCIABLE services. Hence, distinguishing line between these two WP3 deliverables is that D3.1 presents specifications for individual services, while D3.2 elaborates on infrastructural components that are used by all services.

Note that the SOCIABLE infrastructural components make provisions for the platform's expansion with additional services (also beyond the end of the SOCIABLE project). Therefore, in addition to describing a set of services, the present deliverable illustrates the possibility of deploying additional services (i.e. notably additional cognitive games and social interaction applications) over the SOCIABLE platform. Overall, the list of services described in this deliverable is non-exhaustive. We foresee that SOCIABLE partners will endeavor to augment the platform with additional services, during the project but also following the end of the project i.e. as part of the implementation of the SOCIABLE sustainability strategies and associated business plans. The deliverable illustrates the possibility of implementing additional services over the SOCIABLE platform.

The deliverable is structured as follows: Section 2 following this introductory section provides an overview of the SOCIABLE technical architecture and associated deployment model. The purpose of this overview is to position (technically) the SOCIABLE services in the scope of the SOCIABLE pilot environment. Section 2 elaborates also on the implementation technologies used. Section 3 presents a set of guidelines, which will be taken into account in the scope of the technical implementation of the SOCIABLE services. These guidelines should be taken into account in order to meet the SOCIABLE goals for creating a motivating and ergonomic environment over a surface computing platform. Section 4 is devoted to the technical specification of cognitive training games. Note that the list of presented games is non-exhaustive, given that the process of specifying and implementing games will be on-going for almost the whole project duration. Rather than detailing the specifications of all cognitive game, Section 4 focuses on an indicative list. Additional games specifications will be provided as part of the WP4 deliverables documentation associated with the implementation of cognitive games. Section 5 focuses on the technical specifications of the social activation services of the platform, namely the book-of-life and the communication services. Likewise, section 6 elaborates on the technical specifications of the back-office services targeting health professionals. Finally, section 6 concludes the deliverable.

2. Overview of SOCIABLE Technical Architecture, Deployment Model and Implementation Technologies

2.1 Technical Architecture and Deployment Model

The third workpackage of the SOCIABLE project distinguishes between two main elements of the SOCIABLE architecture, namely the SOCIABLE platform and the SOCIABLE services. The SOCIABLE platform provides the hardware, software and middleware infrastructure required for the operation of the SOCIABLE services. This infrastructure provides a common functional denominator, which eases the task of integrating and deploying services on the platform. At the same time the SOCIABLE services comprising cognitive games and social activation applications, as well as services for the health professionals. Contrary to the SOCIABLE infrastructure, the SOCIABLE services will be extensible and expandable during the course of the project, but also beyond the end of the project. This is because the SOCIABLE system will be gradually enhanced with more cognitive training games and social activation services.

Figure 1 illustrates the technical architecture of the SOCIABLE ICT systems within the care centre, with emphasis on the software/middleware comprising the SOCIABLE platform and services. As shown in the figure the SOCIABLE technical developments consist of:

- A database server holding and managing the SOCIABLE data including data for the elderly, the games, as well as various system parameters. This database is part of the SOCIABLE platform and is described in more detail in deliverables D3.2 and D4.4.
- A web server enabling deployment of web applications along with web based access to the SOCIABLE data. The web server is an infrastructural component of any SOCIABLE deployment.
- A set of core services facilitating access to the SOCIABLE data, as well as flexible and modular deployment of the SOCIABLE services. These core services are also described in the scope of deliverable D3.2.
- A set of interfaces to external systems, notably to EHR (Electronic Health Record) systems. Though not directly used in SOCIABLE, these interfaces provide the means for a future integration of SOCIABLE to EHR systems or medical databases.
- The main surface application, which is a sandbox for executing applications over the Microsoft Surface table used in SOCIABLE. This application is described in deliverable D3.2.
- The main Surface PC application, which is a sandbox for executing applications over the SOCIABLE surface PC. This application is also described in deliverable D3.2.

- The back-office application, which provides the SOCIABLE services to the health professionals.
- A set of cognitive training and social activation applications, which are (partly) described in this deliverable. Note that an exhaustive description of all the SOCIABLE cognitive training services is out of the scope of the deliverable, since the SOCIABLE ICT platform will be continually augmented with additional applications and services.

Note that Figure 1 depicts also the possible end-users of the SOCIABLE services. These include: (a) The elderly, who engage in play activities through either the MS Surface Table and/or the Surface PC, (b) The medical experts, who are primarily using SOCIABLE through the back-office application in order to configure sessions and programmes for elderly users and patients. Note that medical experts could also use the surface table (e.g., in order to test the experience of level of difficulty of certain games), (c) The administrator of the infrastructure, who may setup and configure the care centre (software) infrastructure and services through a specialized portion of the back-office application.

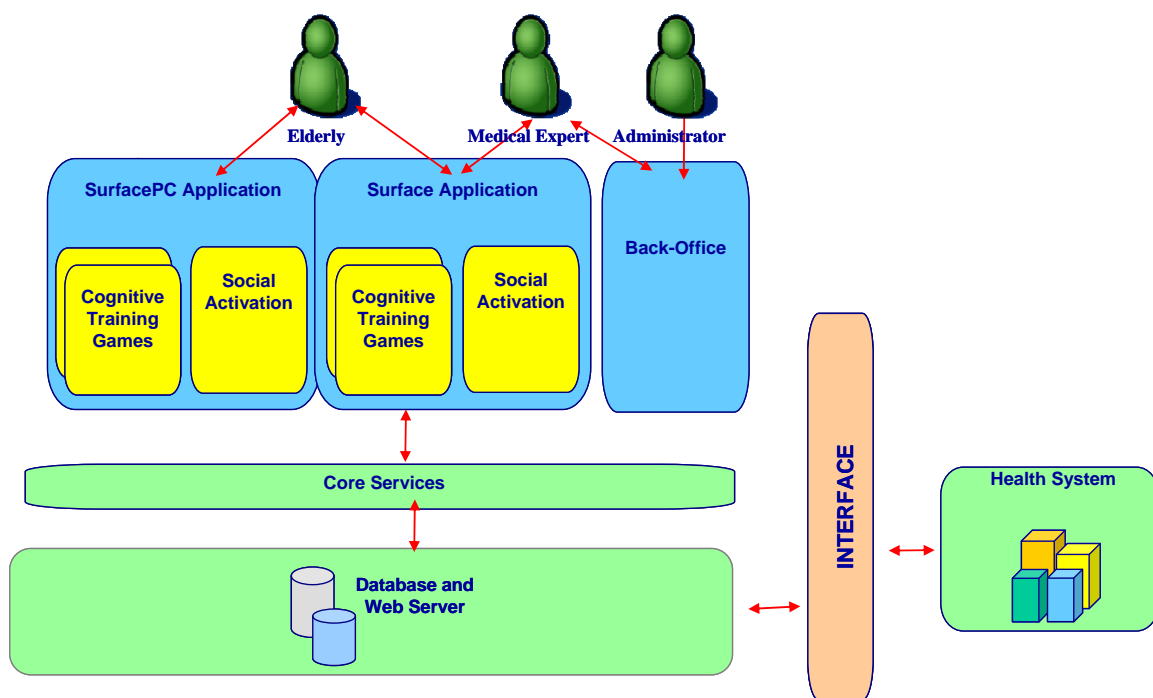


Figure 1: Components Comprising the SOCIABLE Care Centre Technical Infrastructure

Figure 1 depicts the SOCIABLE technical architecture within a single care centre. Note however that SOCIABLE is designed as a system that will be deployed across multiple care centers, given also that the SOCIABLE pilots will take place across seven distinct pilot sites. Figure 2 illustrates the deployment of SOCIABLE in a multi-domain environment comprising multiple care centers. It depicts as care centre network comprising multiple instances of the architecture depicted in Figure 1. A remote management server is providing directory services for the multiple care centers, while also enabling the remote management of the applications at the various care

centers (e.g., remote deployment of SOCIABLE services). The SOCIABLE deployment model is facilitated by the SOCIABLE platform, which is illustrated in deliverable D3.2.

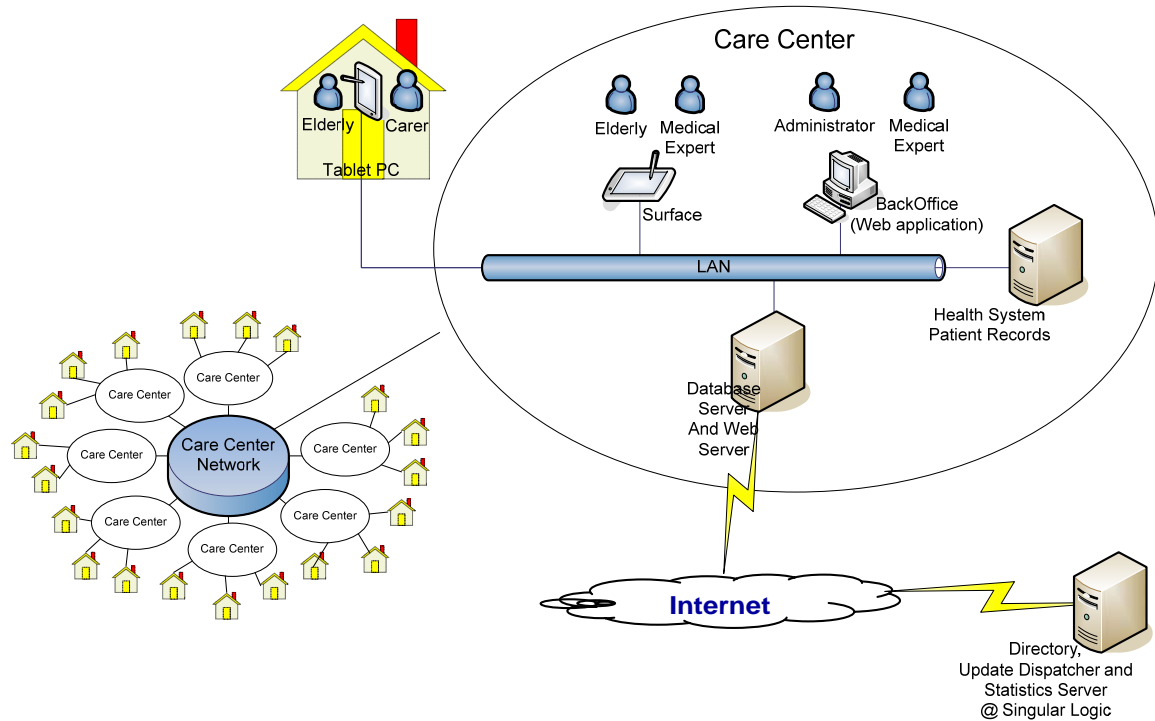


Figure 2: SOCIABLE Deployment Model – Multi-Domain / Across Care Centers

2.2 Specification of SOCIABLE Implementation Technologies

The previous paragraph has listed the main software and middleware components of the SOCIABLE platform and services, which will be more thoroughly described in the present deliverable, as well as deliverable D3.2. From an implementation perspective the SOCIABLE platform and services will be implemented based on a variety of technologies, which are depicted in Figure 3. These technologies have been selected on the basis of the following criteria:

- **Robustness and Maturity:** Due to the pilot nature of the project, SOCIABLE must rely on mature technologies with proven robustness, which are appropriate towards delivering reliable ICT systems for the pilots. SOCIABLE should deliver product-quality ICT services, rather than research prototypes. This is also fully in-line with the project's emphasis on sustainability and exploitability.
- **Compatibility with the selected surface/hardware platforms:** The selection of the surface hardware platform imposes certain choices with respect to implementation technologies. It imposes for example the use of the Microsoft Surface SDK, in order to develop MS Surface applications.
- **Single Platform Interoperability and Cost Efficiency:** All SOCIABLE developments are centered round the Microsoft Windows platform, as well as round Microsoft technologies. Hence, along with the Microsoft Surface Table platform and related technologies, the project will leverage other MS

technologies in order to realize the technical architecture of the project. The project opted for the exploitation of Microsoft technologies for the following main reasons: (a) In order to facilitate interoperability and homogeneity across the different platforms and services, (b) Due to the need to minimize the learning curve associated with the implementation technologies of the project (i.e. there was no need to rely on multiple technologies), while at the same time saving implementation effort through reusing components and implementation across the different surface platform (table, PC) used in the project and (c) Due to the fact that Microsoft technologies are associated with excellent support, technological longevity and robustness.

The SOCIABLE implementation technologies include:

- The Microsoft Surface SDK (Software Development Kit), which will be used in order to implement all surface based applications and services i.e. all services/applications accessing or running the surface platforms of the project.
- The Windows Presentation Framework (WPF), which will be exploited in the scope of the surface application sandbox implementation on both the surface table and on the surface PC.
- Microsoft C#, which will be used as the main programming language of the SOCIABLE platform and services. In particular, C# is used for the implementation of games, the book-of-life, the main sandbox applications, as well as the core middleware services of the SOCIABLE platform.
- SQL Server 2008 (Express), which will serve as the database management system hosting the care centre's database.
- AJAX technologies, which will be exploited in order to boost ease of use and user productivity in the scope of the back-office application.
- W3C Web Services technologies (developed and deployed over an MS .NET platform), which will be used for accessing the core services and/or external systems through a service-oriented way. Even though Web Services are implemented and deployed over a Microsoft .NET platform, the services will be usable/callable from any platform supporting the implementation of Web Services client applications.

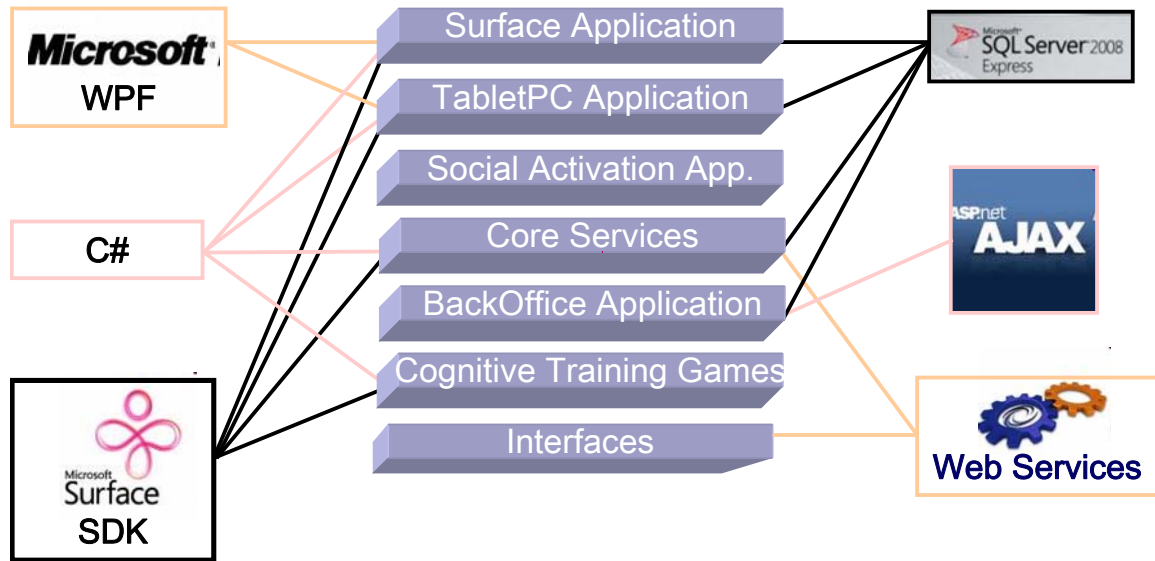


Figure 3: Overview of SOCIABLE implementation technologies

Brief information on these technologies and their use within SOCIABLE are provided in the following paragraphs. For more information, interested readers are also advised to access the official documentation of these technologies.

2.2.1 Microsoft Surface SDK

2.2.1.1 Overview

The Microsoft Surface platform comes as a combination of hardware and software platform, which allows for the development and integration of multi-input, touch-enabled applications. Among the main characteristics of the platform is that it facilitates the development of rich and visually appealing applications, which can offer a new ergonomic and friendly user experiences. Ergonomics and user friendliness hinge on the fact that end-users of surface applications can use only their hands and various objects to manipulate and interact with the applications. SOCIABLE capitalizes on the opportunities provided by Microsoft Surfaces towards natural and intuitive human-machine interaction. It should be noted that the Microsoft Surface unit hides the physical computer from the end-user, which reinforces the revolutionary experience.

From a technical perspective the Microsoft Surface software relies on the Windows Vista platform, which facilitates management, deployment and support of Microsoft Surface units. The Microsoft Surface SDK sits over the MS Vista platform. SOCIABLE uses currently Microsoft Surface 1.0 Service Pack 1, which makes provisions for:

- Enhanced user interfaces, which are a cornerstone to the SOCIABLE surface computing cognitive training concept.

- Improved manageability, which provides opportunities for implementing modular deployment and reducing the total cost of ownership (TCO) of SOCIABLE applications.
- Broader international support, which will be heavily exploited in order to localize the SOCIABLE applications to the various target languages.
- Faster, easier ways to design innovative applications, which will be particularly useful for game designers and developers wishing to augment the SOCIABLE applications.

A discussion of some key and novel functionalities enabled by the Microsoft Surface SDK follows.

2.2.1.2 Visual Interaction

The novel camera system of Surface enables identification of touches, in terms of touching fingers, hands, paintbrushes, tagged objects and many other real-world items. It also provides opportunities for grabbing digital information and interacting with the content through touch and gesture. Surface supports the recognition of many points of contact simultaneously, i.e. fifty-two points. Hence, multiple people can use Surface at the same time, which creates a more engaging and collaborative computing experience comparing to legacy computing devices (such as personal computers, mobile devices, or public kiosks).

The Microsoft Surface SDK recognizes three distinct kinds of contacts:

- Finger: One or more fingers are placed on, moved along, or lifted up from the Microsoft Surface screen.
- Blob: Any object is placed on, moved along, or lifted up from the Microsoft Surface screen.
- Tagged Objects: Any object that has an attached tag is placed on, moved along, or lifted up from the Microsoft Surface screen.

Furthermore, the SDK supports management of gestures i.e. user interactions that applications recognize without the context of what user interface elements they occur over. Gestures do not typically involve changes in the coordinates of the position of the user input. The Microsoft Surface SDK exposes events for two gestures:

- Tap
- Press-and-hold

The surface SDK supports also manipulations i.e. user interactions that an application can properly interpret only in the context of user interface elements. For example, an application could interpret two fingers that are moving away from each other as a "stretch" manipulation, as two separate "move" manipulations, or as several other actions. The actual interpretation depends on what user interface elements appear beneath the interactions. The Microsoft Surface SDK exposes manipulations, in the form of controls, which respond to the following manipulations: Drag, Pan, Flick, Move, Resizing and Rotating,

2.2.1.3 Tagged Object Recognition

Among the main innovative features of the surface is the recognition of Tagged object. The use of tag enables Surface to uniquely identify objects, allowing also the system to distinguish between two objects (due to their different tags). Applications can also use a tag to start a command or action, so simply placing a tagged object on the screen can open up a whole new experience. A tagged object might also identify a cardholder, which is a feature that will be used to enable the SOCIABLE login process.

The Surface SDK enables access to the capabilities of the Microsoft Surface Vision System, therefore enabling the recognition of tags. Tags are special tagged objects that are marked with a special pattern of dots. The Microsoft Surface platform can inherently recognize tags, which facilitates the development of applications that use tags. This is because each tag stores a distinct binary code value, which enables applications to distinguish one object from another. Microsoft Surface supports two types of tags:

- Byte tags store 8 bits of data (1 byte), so there are 256 unique tag values. The value of a byte tag is represented in code by a ByteTag structure.
- Identity tags store 128 bits of data (two 64-bit values), so there is an effectively unlimited range of unique tag values. The value of an identity tag is represented in code by an IdentityTag structure.

2.2.1.4 Compatibility with Other Technologies

The surface SDK enables the Microsoft Surface compatibility with familiar programs and tools, in particular:

- Microsoft® Windows Presentation Foundation (WPF)
- Microsoft® XNA™
- Microsoft® Visual C#® 2008 Express Edition
- Microsoft® Visual Studio® 2008 SP1
- Microsoft® Expression® Studio 2
- Microsoft® Expression Blend™ 2

2.2.2 Windows Presentation Framework (WPF)

SOCIABLE will employ the WPF technology as a graphic subsystem to be used in conjunction with the Microsoft Surface. WPF exploits DirectX and is a popular framework for rendering user interfaces in Windows-based applications. WPF imposes a programming model, which ensures a clear separation between the user interface and the business logic, in accordance to modern application development practices for multi-tier architecture.

From a technical viewpoint WPF is an XML-oriented object model, since it employs XAML (derived from XML), in order to define and link user interface elements. In SOCIABLE WPF will be used in the scope of the back-office services, but also for cognitive games development within the surface. While, it can be deployed as a standalone programme, in the scope of SOCIABLE WPF will be used for building web based interfaces (i.e. as an embedded object in SOCIABLE web based applications). It will be used to unify user interface elements (e.g., 2D/3D rendering, graphics, runtime animation) and accordingly link them and manipulated them according to events, user interactions, and data bindings.

Note that WPF is fully compatible with the MS Surface SDK, since its runtime libraries are included in Windows Vista.

2.2.3 Surface Controls

In order to maximize development productivity and cost efficiency, SOCIABLE will leverage the user interface (UI) controls of the Microsoft Surface SDK. Controls represent fundamental unit of interactions, which can be combined in order to compose an overall interaction scheme. Note that surface controls resemble traditional controls, yet they are customized for the surface environments i.e. they usually represent areas of the screen which users touch in order to complete their actions. The UI controls that are exploited in the scope of the SOCIABLE applications include:

- **LibraryStack**, which is a virtual stack of objects, one placed on top of another, which users can sort through and reorder.
- **LibraryBar**, which is a horizontal arrangement of content, which users can scroll horizontally.
- **ScatterView**, which displays freeform content where multiple users can sort, move, resize, rotate, and organize content simultaneously.
- **SurfaceButton**, which is a region of the screen that has been labeled with a command. When a users touches a this control, a command is invoked.
- **SurfaceCheckBox**, which enables users to choose between two clearly opposite choices.
- **SurfaceInkCanvas**, which translates touch gestures directly into ink strokes.
- **SurfaceListBox**, which enables users to select from a set of values that are presented in a list that is always visible.
- **SurfaceRadioButton**, which enables users to choose from a set of mutually exclusive, related options.
- **SurfaceScrollBar**, which allows users to move to a particular portion of a list, display, or other viewing area.
- **SurfaceScrollViewer**, which offers a scrollable area that can contain other visible elements. This control makes it easy for users to directly manipulate a large piece of content.
- **SurfaceSlider**, which allows users to choose from a continuous range of values.

- **SurfaceTextBox**, which is a typical text box where a user can enter text. Whenever the user touches this control, the on-screen keyboard appears to enable the user to enter text.
- **TagVisualizer**, which reacts to one or more tagged physical objects that are placed on the Microsoft Surface screen. This control facilitates the quick development of applications that use tags.

2.2.4 Sample Microsoft Surface Components and Applications

The SOCIABLE services will be built upon commercial features and sample applications of the Microsoft Surface Platform, which offers a variety of sample applications utilizing advanced visualization and interaction interfaces. A non-exhaustive list of sample developments (source code available), which include games is available at:

[http://msdn.microsoft.com/en-us/library/ee804866\(v=Surface.10\).aspx](http://msdn.microsoft.com/en-us/library/ee804866(v=Surface.10).aspx)

2.2.5 The C# Programming Language

The C# is one of the most prominent programming languages, which features imperative, declarative, functional, generic, object-oriented, and component-based characteristics. C# is part of the .NET initiative, as well as an ECMA (ECMA-334) and ISO (ISO/IEC 23270) standard. C# is one of the programming languages supported in the scope of the .NET Common Language Infrastructure (CLI) environment. The most recent version of the language, which is used in SOCIABLE is C# 4.0. C# is very robust and durable, while also facilitating programmer's productivity. Furthermore it provides strong support for internationalization. These properties make it appropriate for the SOCIABLE applications.

2.2.6 AJAX

AJAX stands for Asynchronous JavaScript and XML and signifies a new way to use existing standards in web based software development. AJAX is a key to building user-friendly GUI (Graphical User Interfaces), through facilitating updates on parts of a web page without reloading the whole page. To this end, AJAX enables web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. SOCIABLE will leverage AJAX in the implementation of web interfaces, notably interfaces of the back-office applications.

2.2.7 SQL Server

SQL Server is the database server that will be used in SOCIABLE. The main criterion for selecting SQL server as the Relational Database Management Software (RDBMS) of the project is its compatibility with Microsoft technologies and the MS operating systems. SOCIABLE will leverage the latest version of the SQL Server 2008 R2 in order to execute ANSI SQL queries for data management.

2.2.8 Web Services

SOCIABLE will leverage the popular Web Services model, in order to build application programming interface (API) that could be called from remote. Such APIs will be needed in the scope of the integration of SOCIABLE applications with external systems (such as Electronic Health Records Management systems). At the same time, such APIs will facilitate remote web-based access to the SOCIABLE databases.

3 Surface Application Design Specifications and Interaction Guidelines

In the sequel, we list the main application design guidelines that will be taken into account in the development of the cognitive games and other SOCIABLE services. These guidelines will facilitate one of the main technical development targets of the SOCIABLE project, which is to create a user-friendly, motivating and ergonomic environment.

3.1 Spatial Interaction and 3D Environments

The spatial interaction and 3D design guidelines for the SOCIABLE applications, aims at exploiting optimally the space of the surface, through exploiting both the plan of the display, but also through creating a true third dimension. Hence, SOCIABLE applications should as much as possible attempt to make objects look as if they have a depth, in order to achieve immersion and enable the objects to behave like 3D objects.

Traditional GUI interaction models are flat, planar, and two-dimensional (2-D). You can use some techniques (such as skewing planes, adding shadows, and overlapping elements) to make some objects look as if they have a depth. Spatial interactions and the creation of 3D environments can enable a sense of realism, which is a contributing factor towards a motivating play environment for the elderly.

3.2 Super-Reality

Super realism will be another vehicle for creating an experience that resembles natural interactivity with objects. Super realism pushes beyond what is physically natural so that experiences do more than what is possible in the real world. Practically, it enables direct interaction with objects (e.g., push), including however interactions that are literally not possible. As a typical example to be employed in games involving pictures, images, puzzles and pieces:

- Elderly users should be able to touch objects and push them around (like in the real world), but at the same time:
- Elderly users should be able to scale objects by using a two-finger gesture (unlike the real world).

3.3 Context-Awareness

Cognitive training games and social interaction applications must be designed in a way that makes them adapt to the context of the end-user. Hence applications must respond to users' actions according to the most proper way pertaining to the context of their action or behavior. Characteristic examples of context-aware behavior include:

- Making the available steps and choices very obvious/prominent. This may involve restricting users' choices according to his/her context i.e. previous steps or actions. Moreover, the applications should make the "right" next step very prominent.
- Avoid extremely automated steps in order to give users' the feeling that they control the application. Likewise, the project should provide users' with the opportunity to lead the experience, offering however options for remedying mistakes.

Overall, the SOCIABLE applications must be designed in a way that facilitates the elderly users to understand, control and anticipate the results of their actions.

3.4 Touch and Direct Manipulation

SOCIABLE applications and games should leverage as much as possible unique features provided by the Microsoft Surface. One of these unique features is direct manipulation i.e. the ability of surface applications to respond to one finger touch, many fingers touches, as well as physical objects. Touch actions and the use of physical objects become temporal and unmediated in the way that content is manipulated. Through direct manipulation, users can play games through many fingers, as well as through the use of physical objects.

SOCIABLE applications must leverage direct manipulation as much as possible, with a view to providing a natural environment for user interaction with the platform.

3.5 Virtual Objects like Physical Objects

The surface platform of the project makes it possible to use virtual objects. Whenever virtual objects are used, it must be attempted to make them behave like physical objects. To this end, transitions must be made fluid (i.e. objects smoothly move, animated and transition). Furthermore, transitions should be implemented in a way that exposes mass, acceleration, friction, viscosity, and gravity. These properties are readily available in some controls of the Microsoft Surface SDK (such as the ScatterView control).

In the scope of the use of virtual objects, the SOCIABLE services should make prominent the state changes of the virtual objects (including the starting and ending state). Also, special attention should be given to physical reactions, which should be customized to the needs of each application.

3.6 Use of Tagged Objects

As already outlined, a tag is a special pattern of dots, which enables mixing of the physical and virtual worlds in the scope of Microsoft Surface applications. SOCIABLE intends to use the tags in order to enable elderly end-users to interact with the platform using physical objects. A prominent example is the use of a tag for login to the platform, with a need to providing user names and passwords.

From a service design perspective SOCIABLE applications should deal with tagged objects based on the following principles:

- They should respond to every object and contact. The objective is to ensure that all tagged objects are interactive. No doubts should be created to end-users about the proper function of the objects.
- They should respond immediately to the presence of tagged objects, in order to facilitate the seamless blending of real and virtual objects.
- Allow inference of locations and shape of objects, through placing the tags in the same physical location of the surface. This should be taken into account in the design of the surface applications.
- Minimize reflection of IR objects, to avoid generation of contacts within the application.

3.7 Use of UnTagged Objects

In terms of untagged objects, Microsoft Surface can detect IR-reflective objects that are placed on the screen. Regarding the use of untagged objects, application designers should:

- Similarly to tagged objects, application have to respond to every object and contact. End-users should not have the wonder whether an untagged object is "functioning" or not.
- Feedback on the placement of objects should be provided (e.g., through applying effects), in order to indicate that the surface understands their presence, even if they are not identified.

In general, applications should (as much as possible) use tagged (rather than untagged) objects.

3.8 360° degree User Interface

In order to enhance user experience, SOCIABLE should pay special emphasis to the orientation of the applications and the respective user interfaces. The objective should be to allow users to approach/use applications from all possible directions. In order to achieve this, the Microsoft Surface SDK offers controls (notably the ScatterView control), which allow users to orient content towards any edge of the display.

3.9 Playful Pleasurable and Explanatory Touches

As mandated by the SOCIABLE user requirements (elaborated in WP1), surface applications should focus on the creation of fun and pleasurable experiences, which maximize both quality and enjoyment. To this end, applications should leverage smooth transitions of virtual objects, use high-quality & physically-oriented graphics in order to motivate and please users, make the experience as responsive as possible, while also ensuring a superior technical performance (i.e. without any delays or hiccups).

3.10 Progressive Disclosure

Progressive disclosure is a technique for creating natural and realistic interfaces, based on revealing deeper content and functionality. A classical example of progressive disclosure functionality is zooming. SOCIABLE applications should exploit progressive disclosure in order to ensure gradual exploration of content, based on natural metaphors such as zooming and flipping over objects.

3.11 Multiple Senses

SOCIABLE applications should appeal to multiple sensors, through visuals, sound, motion, and physical interactions. Multiple senses facilitate the motivation and pleasure of users. SOCIABLE designers should provide graphics, images, sounds etc. in order to facilitate the development of multi-sensory applications.

3.12 Continuous Input

A distinguishing characteristic of surface applications is the fact that they allow for continuous actions, rather than discrete actions which are typical in conventional applications. To this end, all touches should be associated with an immediate visual response (i.e. feedback), thus avoiding waiting time as results are computed. Hence, SOCIABLE applications should make touches appearing as a direct, physical reaction through placing visual responses immediately at the position of the contacts. Furthermore, applications should allow the use of many fingers and many objects, through ensuring appropriate response to many forms of simultaneous input.

3.13 The content should (as much as possible) be the interface

The user interface be designed in a way that it has only limited items that do not represent content. Controls should appear based on the content, overall creating and preserving the impression that users interact directly with the content. Users should also be able to directly move objects on the screen.

3.14 Integration with the Surface Shell

The surface shell of the Microsoft platform should be integrated with all applications in order to provide a baseline behavior for all applications. The shell should ensure the integration of introductory video, text, universal access points, as well as application previews.

Note that specialized version of the Microsoft Surface shell will be created and customized for the SOCIABLE applications, as specified and illustrated in deliverable D3.2.

4. Technical Specification of Cognitive Training Games

4.1 Game #1- "Hide and Seek"

4.1.1 Detailed Specification

The user is shown of a fully-furnished and fully-decorated room and is asked to hide 5-10 items in there. After a 15-20 minute delay, the user is asked to recall where he has hidden the various objects.

Difficulty levels

- Difficulty level 1. The user is asked to hide 5 items: car keys, wallet, glasses, TV remote control and pills.
- Difficulty level 2. The user is asked to hide 7 items: car keys, wallet, glasses, TV remote control, pills, identity card and camera.
- Difficulty level 3. The user is asked to hide 10 items: car keys, wallet, glasses, TV remote control, pills, identity card, camera, watch, umbrella and mobile phone.

Different versions of the game

The user can choose if he/she wants to hide the items in a) the kitchen, b) the bedroom or c) the living room. All the rooms must have an equal -more or less- number of furniture/appliances and decorative items (such as vases, flower pots, paintings, carpets, pillows, baskets etc). There should be a minimum of 20 different places (furniture/appliances and decorative items) to hide an item in each room.

Game flow

- **Screen 1:** A short sound is heard. On top, there is the name of the game. Below, there could be three buttons: info, example and start. Also, there could be a music on/off button at the bottom. If the user presses the info button, he/she gets a short description of the game (Train your memory by hiding objects in a room and then trying to remember where you have hidden them). If he presses the example button, he is presented with a short example showing how he can drag and drop an item in a certain place and how, later, he can indicate where it is hidden. If he presses the start button, the game begins.
- **Screen 2:** Difficulty level. We could indicate different difficulty levels in a funny way by showing a man with athletic outfit walking (level 1), running (level 2) and being exhausted (level 3). All of them could be moving images. Or we could have the profile of a face with the brain in its position. The brain is connected to a starting handle. In level 1 picture the hand turns the starting handle slowly, then faster and in level 3 very fast.
- **Screen 3:** Choose a room. The user is shown the three pictures of the rooms with their corresponding name underneath and is asked to press the picture with the room he prefers.

- **Screen 4:** Instructions. Hide your personal items in the room. To hide an object, drag and drop it wherever you would like it to be hidden. Try to keep in mind the hoard you have selected for each item! (is hoard the right word?). Once you have hidden all the items, click "Ready".
- **Screen 5:** The front view of a 3D room is presented. On the right there is a column with the objects to be hidden.
- **Screen 6:** When the user presses the button Ready, he gets the message: Great! Remember that you have to keep in mind where you have hidden each object!
- **Screen 7:** After 15 minutes, the user is presented with a screen showing the room he/she had selected. On the right there is a row with the various objects hidden. On top there is the following message: Do you remember where you have hidden each object? First click on the object you want to locate and then the place where you think you have hidden it. If the user is right, he gets an applause and a "correct" message appears on the screen together with a corresponding picture (e.g. a 3D green tick). The object he has correctly located disappears from the screen. If not, a red X appears on the screen together with the message "Try again". If the user fails to correctly locate the item again, the item disappears from the column, the red X appears again and he gets the message "Why not try to locate another object?".
- **Screen 8:** Accuracy, presented in percentage and with a picture. If we have used the man in the athletic outfit to indicate difficulty, now we can show him wearing a gold/silver/bronze medal. If we have used the brain image, we could show the face profile with a paper and pencil/calculator/laptop in the place of the brain. Apart from accuracy, the time needed to complete the game could also be shown. These two parameters are the most important and these should be used for other games as well.

4.1.2 Relevant Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "Hide and Seek" game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of spatial interactions in order to depict depth and increase realism.
- The development of the application as a multi-sensory application.
- The exploitation of the context-awareness guideline in order to enable progressive hiding and finding of the games.

4.1.3 Sample User Interfaces

A set of sample screens for the "Hide and Seek" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.



Figure 4: Screenshots of the "Hide and Seek" game (user interface design)

4.2 Game #2 - Puzzles"

4.2.1 Detailed Specification

The user is shown pieces of a puzzle¹ and is asked to put them onto the grid as fast as he can in order to make the picture. There is no time limit to complete the game, but the user gets a time bonus if he/she puts the puzzle together quickly enough.

Difficulty levels

- Difficulty level 1. Puzzles with 9 pieces
- Difficulty level 2. Puzzles with 15 pieces
- Difficulty level 3. Puzzles with 21 pieces

Game flow

Similar sequence to with the "Hide and Seek" game is followed:

- **Screen 1:** A short sound is heard. On top, there is the name of the game. Below, there could be two buttons: info, and start. Also, there could be a music on/off button at the bottom. If the user presses the info button, he gets a short description of the game and is presented with a short example showing how he can move the pieces of the puzzle. If he presses the start button, the game begins.
- **Screen 2:** Difficulty level. The medical expert chooses a difficulty level.
- **Screen 3:** Puzzle selection. The user chooses between puzzles of equal difficulty.

¹ See as an example: <http://www.puzzlepixies.com/easy/easy/octopus-jigsaw.html>

- **Screen 4:** Instructions. Put the puzzle together as fast as you can! Select a piece and then, drag and drop it onto the correct place. If you get it right, you will hear a noise and the piece will stick in the grid. If you get it wrong, your piece will zoom off back to its original position. You can restart the game at any time by clicking on the "Clear" button at the top right hand corner of the screen.
- **Screen 5:** On the left hand side, the full picture is presented. On the right hand side there is an empty grid and next to it, the pieces of the puzzle. On the screen there is also a "Clear" button and a timer.
- **Screen 6:** Accuracy (number of wrong movements) and time to complete the game.

4.2.2 Relevant the Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "Puzzles" game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of spatial interactions in order to depict depth and increase realism when manipulating objects.
- The exploitation of touch and direct manipulation in order for example to alter the size of the pieces, to manipulate them using fingers etc.
- A 360° degree interface enabling users to manipulate the puzzle regardless of orientation.

4.2.3 Sample User Interfaces

A set of sample screens for the "Puzzles" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.



Figure 5: Screenshots of the "Puzzles" game (user interface design)

4.3 Game #3 - Analogies

4.3.1 Detailed Specification

Analogies provide excellent training in seeing relationships between concepts. In analogies, the user is given one pair of related words/pictures and another word/picture without its pair. The user must choose a word/picture that has the same relationship to the word/picture as the first pair. For example, fire is to hot, as

ice is to cold. Pairs of words/picture can be related in many ways, including the following types: opposites, synonyms, things that go together (bow/arrow), object and classification (green/color), object and related object (dog/puppy), object and function (pen/write), problem and solution (tired/sleep), degrees of a characteristic (cold/freezing) etc.

The "tricky" thing in this game is how to present the analogies on the screen. I guess that the best possible solution would be to prepare 2-3 different versions to choose from.

Difficulty levels

The difficulty levels will be determined on the basis of the abstraction level and the frequency/familiarity of the stimuli (which could be words, pictures or even sounds). Three difficulty levels are envisaged:

Difficulty level 1

Verbal analogies

1. Car: road train: _____
track vehicle fast wheel

2. Grass: soil seaweed: _____
water salty river fish

3. Lion: animal flower: _____
plant grass roots rose

4. Stair: steps book: _____
title library pages knowledge

5. Top: bottom front: _____
back below right down

6. Fish: swim bird: _____
sing feather tree fly

7. Huge: big tiny: _____
dot thin small short

8. Foot: socks hand: _____
watch wrist gloves nails

9. Cat: meow (sound) dog: _____
SOUNDS: dog cow horse hen

10. CD: music (sound) football ground: _____

SOUNDS: waves playground traffic football ground

Picture analogies

1. Leopard: spots zebra: _____
cow skin rabbit fur zebra skin feathers
2. Pencil: workbook blackboard: _____
pencil brush sponge chock
3. Bow: arrow gun: _____
dart bullet target policeman
4. Rug: floor sheet: _____
drawer table pillow bed
5. Bell: ring (sound) phone: _____
SOUNDS: bee train phone boat
6. Glass: break (i.e. broken glass) paper: _____
cut put on fire crumple write on
7. Head: helmet eyes: _____
sunglasses eyebrows glasses bag
8. Deer: forest tiger: _____
lake tree jungle cage
9. Kite: string dog: _____
bone muzzle ball dog leash
10. Scissors: paper saw: _____
stone trunk bread flower

Difficulty level 2

Verbal analogies

1. Fish: egg plant: _____
leaf root seed stem
2. Boy: child man: _____
father uncle adult person

SOCIABLE: Motivating platform for elderly networking, mental reinforcement and social interaction
WP3- Services Specification
Deliverable D3.1: "SOCIABLE Services Specification"

3. car: 4 bike: _____
2 6 1 3
4. Clock: time thermometer: _____
temperature degrees weather centigrade
5. Butterfly: caterpillar frog: _____
fish amphibian tadpole toad
6. Quiet: sound darkness: _____
cellar sunlight noise stillness
7. Question: answer fire: _____
flame fireman water matches
8. Work: labor rich: _____
poor wealthy millionaire money
9. Group: people herd: _____
birds animals cars plants
10. December: January last: _____
month second first one

Picture analogies

1. Fish: submarine bird: _____
kite airplane cloud birdcage
2. Knob: door keys: _____
keychain wallet lock purse
3. Pink: red beige: _____
brown green white blue
4. Pyramid: triangle cube: _____
square rectangle polygon box
5. Snow: flake rain: _____
cloud weather drop rainbow

6. The rest could be figural analogies like the ones presented in the following link: <http://www.designsforstrongminds.com/game/game1.htm>, but much easier. Again, we should have four different possible answers.

Difficulty level 3

Verbal analogies

1. Violence: activity melancholy: _____
evening cruelty mood silence
2. Falling: gravitation collapse: _____
balloon electricity pressure destruction
3. Musician: entertainment professor: _____
pupils school homework instruction
4. Wave: tide moment: _____
time ocean tardiness clock
5. Work: dismissal marriage: _____
divorce death birth end
6. Monologue: dialogue king: _____
reign democracy oligarchy regime
7. Planet: universe part: _____
piece moon infinity total
8. Miracle: experiment faith: _____
church science God laboratory
9. Adult: teenager writing: _____
poem song draft report
10. Where: everywhere when: _____
mostly often never always

Picture analogies

Figural analogies like the ones presented in the following link: <http://www.designsforstrongminds.com/game/game1.htm>, with four different possible answers must be used.

Game flow

- **Screen 1:** A short sound is heard. On top, there is the name of the game. Below, there could be two buttons: info, and start. Also, there could be a music on/off button at the bottom. If the user presses the info button, he gets a short description of the game and is presented with a short example showing how he can move the pieces of the puzzle. If he presses the start button, the game begins.
- **Screen 2:** Difficulty level. The medical expert chooses a difficulty level.
- **Screen 3:** Instructions. Select the answer at the bottom of the screen that best completes the analogy. Try to be as fast as you can!
- **Screen 4:** The user is presented with the analogy. On the screen, there is also a timer and a "next" button. The user gets feedback depending on the answer he gives. If he is mistaken, then he gets the message "Wrong answer! Try again".
- **Screen 5:** Accuracy (number of mistakes) and time to complete the game.

4.3.2 Relevant Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "Analogies" game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of playful and explanatory touches in order to provide additional instructions for completing the games.
- Progressive disclosure of objects/words that relate to the analogies at hand.

4.3.3 Sample User Interfaces

A set of sample screens for the "Analogies" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.



Figure 6: Screen shots of the "Analogies" Cognitive Training Game

4.4 Game #4 – "Lost In Migration" (or "Lost in the City")

4.4.1 Detailed Specification

"Lost in migration" has been originally designed by www.lumosity.com (<http://www.lumosity.com/brain-games/attention-games/lost-in-migration>), where the game flow is illustrated. Attention should be given to the following matters:

- Instead of birds, in SOCIABLE we will use people showing to different directions.
- The three difficulty levels should be determined on the basis of the number of distractors, namely:
 - Difficulty level 1: 5 persons in total, all showing to the same direction except for the person in the middle.
 - Difficulty level 2: 5 persons in total, 3 of them showing to one direction, 1 showing to another and the person in the middle showing to a different direction from all the others.
 - Difficulty level 3: 5 persons in total, 2 of them showing to one direction (e.g. up), 2 of them showing to two different directions (left and right respectively) and the person in the middle showing to a different direction from all the others (e.g. down).

4.4.2 Adoption of the Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "Lost in migration" game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of progressive disclosure (e.g., zooming) as the end-users navigates in the game.
- The exploitation of multiple senses including several sounds and images.
- The exploitation of playful and explanatory touches in order to provide additional instruction for completing the game.

4.4.3 Sample User Interfaces

A set of sample screens for the "Lost-in-Migration" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.



Figure 7: Screenshots of the "Lost in Migration" Game

4.5 Game #5 – My Home

4.5.1 Detailed Specification

For the purposes of this orientation game, the user has to find his way in his new home. A person standing in the center of a red circle is placed on the starting point. A different indication is used to show the room where the user must go.

Instructions

You have just moved in your new house. Try to familiarize with it by going from one room to another as fast as you can! Use your finger to draw your route, without bumping into the furniture.

Difficulty levels

The difficulty levels are determined on the basis of the number of rooms, furniture and decorative items.

- Difficulty level 1: Five rooms. Bedroom, kitchen, living room, wc, garden. Few furniture and decorative items.
- Difficulty level 2: Seven rooms. Bedroom, office, kitchen, living room, play room, wc, garden. More furniture and decorative items.
- Difficulty level 3: Nine rooms. Master bedroom, kids' bedroom, kitchen, living room, play room, office, wc, gym, garden. Many furniture and decorative items.

The rooms should look more or less like the ones in the pictures attached, but the doors should be clearly stated since the user should go through them to move from the one room to another.

Last screen: Number of mistakes (e.g. bumping into walls, furniture etc) and time to complete the game.

4.5.2 Adoption of the Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "My Home" game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of spatial interaction and 3D environments to make the home objects look more natural.

- The exploitation of playful and explanatory touches in order to provide additional instruction for completing the game, especially as the end-user attempts to navigate through the home.

4.5.3 Sample User Interfaces

A set of sample screens for the "My Home" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.

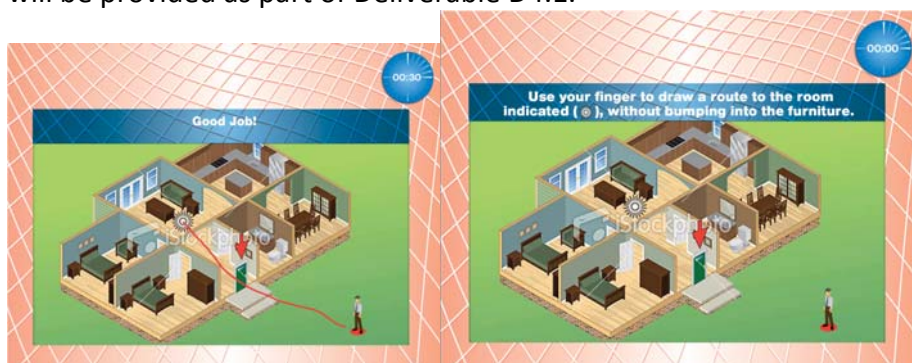


Figure 8: Screenshots of the "My Home" Game

4.6 Games #6 and #7 – Synonyms and Antonyms (respectively)

4.6.1 Detailed Specification

The games "Synonyms" and "Antonyms" are presented together as the games' design will be exactly the same. In the case of these two games, the words are presented in two different lists. The user is required to draw a line with his finger between the words on the left and their synonyms/antonyms on the right.

Difficulty levels

Difficulty level 1: list of 10 simple, concrete, everyday words

Synonyms

Woman	Weird
Happy	Hear
Noisy	Regular
Shiny	Dirty
Tiny	Talk
Speak	Joyful
Normal	Loud
Strange	Bright
Listen	Miniature
Filthy	Lady

Antonyms

Large	Fast
Smooth	Wrong
Wonderful	Over
Slow	Small
Early	Difficult
Correct	Hot
Under	Walk
Easy	Late
Cold	Rough
Run	Awful

Difficulty level 2: list of 15 both concrete/frequently used and abstract/less frequently used words

Synonyms		Antonyms	
Hard	Toss	Accept	Minority
Throw	Tutor	Succeed	Guilty
Eat	Wages	Simple	Wealth
Way	Challenge	Majority	Help
Teach	Anxious	Join	Complicated
Salary	Path	Increase	Sensible
Alter	Lend	Afraid	Reasonable
Argument	Change	Poverty	Expand
Permanent	Devour	Foolish	Selfish
Blend	Difficult	Innocent	Refuse
Borrow	Notice	Aid	Rejected
Observe	Proposal	Extreme	Fearful
Offer	Combine	Shrink	Decrease
Impatient	Temporary	Considerate	Separate
Dare	Dispute	Accepted	Fail

Difficulty level 3: list of 15 more "difficult"/abstract/less frequently used words.

Synonyms		Antonyms	
Elude	Employ	Economical	Wasteful
Raise	Custom	Knowledgeable	Improve
Crisis	Cure	Energetic	Flourish
Charming	Huge	Dainty	Accept
Use	Harvest	Danger	Fake
Deteriorate	Repair	Deceitful	Lethargic
Habit	Wander	Genuine	Ignorant
Harden	Escape	Decadence	Obscene
Reap	Weaken	Decline	Security
Heal	Chase	Lazy	Expose
Immense	Quest	Robust	Confirm
Roam	Emergency	Decent	Gross
Mend	Solidify	Damage	Feeble
Pursue	Enchanting	Bury	Trustworthy
Adventure	Elevate	Cancel	Diligent

4.6.2 Relevant Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "Synonyms and Antonyms" game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of playful and explanatory touches in order to provide additional instruction for completing the game, especially as the end-user attempts to match synonyms and antonyms.
- The use of touch and direct manipulation (e.g., in order to be able to alter/size the words).

4.6.3 Sample User Interfaces

A set of sample screens for the "Synonyms and Antonyms" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.

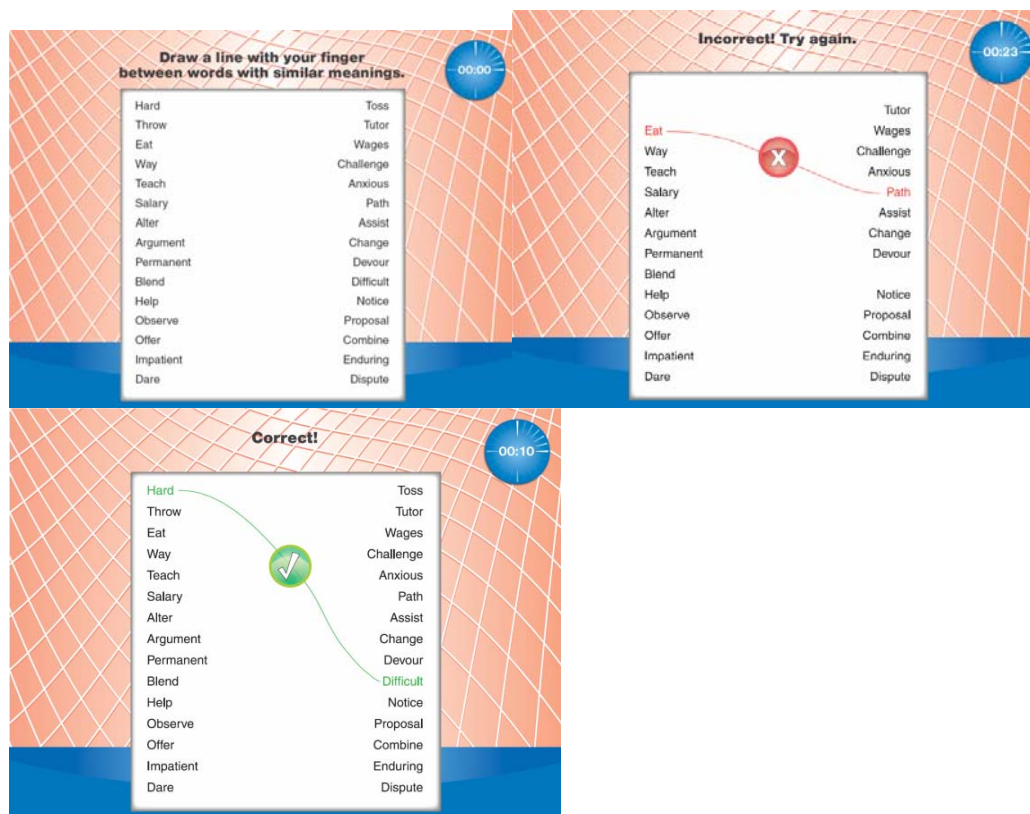


Figure 9: Screenshots of the "Synonyms and Antonyms" Game

4.7 Games #8 – Guess Who

4.7.1 Detailed Specification

Each player starts the game with a board that includes images of people and their first names. The PC gives clues (one by one) to eliminate candidates, starting with clues on obvious outward appearance characteristics of the person selected (e.g. the character selected is a man / wears glasses etc.) and continuing with questions based on the personalities (the character seems angry) or personal histories (the character is a wealthy lawyer) of the character. The user identifies (touching with his finger) images one by one to eliminate candidates.

Difficulty levels

- **Difficulty level 1.** The user is presented with the images of 10 people and is given clues only on obvious outward appearance characteristics of the person selected.

- **Difficulty level 2.** The user is presented with the images of 15 people and is given clues both on outward appearance and personality/personal history of the character selected.
- **Difficulty level 3.** The user is presented with the images of 20 people and is given clues both on outward appearance and personality/personal history of the character selected.

Instructions

You'll be presented with images of 10/15/20 different people. Based on the cues provided, try to eliminate candidates and correctly guess the mystery person chosen! Time matters, so be as fast as you can!

Parameters measured

Accuracy (defined on whether the user found the mystery person chosen, as well as on the number of mistakes) and time. Notice that during game play, the user is notified if he eliminates a person that does not fulfill the criteria determined by the cue provided (e.g. the cue is "the person wears glasses" and the users eliminates a person not wearing glasses).

4.7.2 Relevant Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "Guess Who" game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of playful and explanatory touches in order to provide additional instruction for the faces comprising the game.
- The use of touch and direct manipulation (e.g., in order to be able to rotate the faces).
- The use of multiple senses (audio, video and animation).

4.7.3 Interfaces to the Platform

A set of sample screens for the "Guess who" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.



Figure 10: Screenshots of the "Guess Who" Game

4.8 Games #9 Par and Par Maxis

4.8.1 Detailed Specification

Instructions

The user has to find the pairs that match the images. Cards are randomly dealt with the picture side down. Flip over any two cards by tapping on them. If the pictures on the cards are identical they will disappear. If they are not identical they will be flipped back picture side down.

At the start of the game the images have to show to the patient one by one and called. Then the images have to face down and placed on a grid

The game is made up of 38 pieces having different images. We can decide if we want to use images or photographs. The patient has to find the pair to match the image. He touches the screen to choose the pair. The computer will give feedback if the answer is right or wrong. If the answer is correct the images disappear.

The difficulty level will be determined by the number of the cards on the grid and the similarities within the images.

4.8.2 Relevant Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "Par and Par Maxis" game could adopt:

- The use of tagged object for the login of the patient.
- The use of an appropriately tagged object (e.g., cards) for matching a given picture.
- The use of multiple senses (audio, video and animation).

4.8.3 Interfaces to the Platform

A set of sample screens for the "Par and Par Max" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.



Figure 11: Screenshots of the "Par and Par Max" Game

4.9 Games #10 Take Away Menu

4.9.1 Detailed Specification

Instruction

You are on holiday in a hotel with your family. In order to offer you the best and quickest service you are kindly invited to compose your menu for the dinner. Choose your dishes from our Chef's list of the day!

Difficulty levels

The difficulty levels that can be introduced are based on the number of guests, the number of dishes, as well as on the number of selection criteria. In particular, the following levels are specified:

- **Difficulty level 1:** The user has to select four dishes (hors-d'oeuvre, first course, main course, side dishes/vegetables, dessert etc). In this level we have 1 criterion for selecting the dishes (for example don't have mushroom or meat, shouldn't spend more than etc., cold dish etc). The Chef's list should have three dishes for each type of food
- **Difficulty level 2:** The user has to select four dishes. In this level the user has to order also for another guest and with two criteria (one criterion for each person). The Chef's list should have four dishes for each type of food.
- **Difficulty level 3:** The user has to select four dishes. In this level the user has to order also for another guest and two more criteria (i.e. one criterion for each person), while not being allowed to spend more than a given amount. The Chef's list should have 5 dishes for each type of food

Parameters

Accuracy (whether the user completed the game and number of mistakes) and time to complete the game.

At the end of the game the user is invited to memorize his order and to recall it in the scope of another SOCIABLE game, namely the game "*Do you remember your order?*". The user should be given an interference task between the game such as "find the difference" or another game.

Obviously the menu mentioned above should be modified for each country.

4.9.2 Relevant Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the "Take Away Menu" game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of context awareness for the users' navigation across the different choices/options of the menu.
- The use of pleasurable and explanatory touches.

4.9.3 Interfaces to the Platform

A set of sample screens for the "Take Away Menu" game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.



Figure 12: Screenshots for the "Take Away Menu" Game

4.10 Game #11 – Scrambled Story

4.10.1 Detailed Specification

The users are presented with separate paragraphs of a story. The paragraphs are shuffled randomly on the screen. The PC can read the sentences in order of appearance. The users put (touching with his finger) in the right order the paragraphs in order to make a coherent story.

Instructions

You'll be presented with 6/8/10 sentences that are part of a story. Try to put the sentences in order to make a coherent story. Time matters, so be as fast as you can!

Difficulty levels

- Difficulty level 1. The user is presented with 6 pieces (paragraph of a story).
- Difficulty level 2. The user is presented with 8 pieces (paragraph of a story).
- Difficulty level 3. The user is presented with 10 pieces (paragraph of a story).

If the user isn't able to complete the story alone the PC can read the whole story and then the user can try to complete the game again (for example after five errors).

Example

Mary wakes every morning at 7 a.m.. She has breakfast with her husband Antonio and two children, Sara and Matthew. Then she gets dressed and accompanied children to school. Sara, who is seven years old, attends primary school. Matthew, who is twelve years old, attends high school. After having accompanied their children to school Mary goes to work.

Parameters measured

Accuracy (i.e. whether the user completed the story and number of mistakes) and time.

4.10.2 Relevant Surface Application Design Guidelines

In terms of the surface application design guidelines presented in the previous section, it is recommended that the “Scrambled Story” game could adopt:

- The use of tagged object for the login of the patient.
- The exploitation of progressive disclosure in the case of long stories.
- The use of touch and direct manipulation for increasing or decreasing the font size of the story.

4.10.3 Interfaces to the Platform

A set of sample screens for the “Scrambled story” game are provided below. Note that the full set of user interfaces, along with the methodology followed towards their design will be provided as part of Deliverable D4.1.



Figure 13: Screenshots for the “Scrambled Story” Game

4.11 Baseline Workflow of the Games

All the games outlined above are designed and implemented according to the some baseline workflow, which includes the following steps:

- **First step – Selection of the level of difficulty for the game:** In this step the therapist chooses a level of difficulty (easy, intermediate, and difficult) for the user to play.
- **Second step – Instructions:** In this step the screen displays two options:
 - *Information:* If this is the first time the game is played, the user has to click on the “information” button to discover the aim of the game.
 - *Start:* If the game has been played before, the user can press Start to begin the game.
- **Third step – Playing the Game:** The game will begin once the user has chosen a game scene. The aim of the games vary, according to the games scenarios and specification provided in the previous paragraphs of this section.
- **Fourth step: Scoring:** When the game is over, the results that the user obtains during the game are displayed on a screen.

Figure 14 depicts this game workflow for the “Hide and Seek” game. The workflows for the rest games are very similar (since they are organized around the baseline

workflow outlined above). The workflows of the other games are provided in the Appendix of this document.

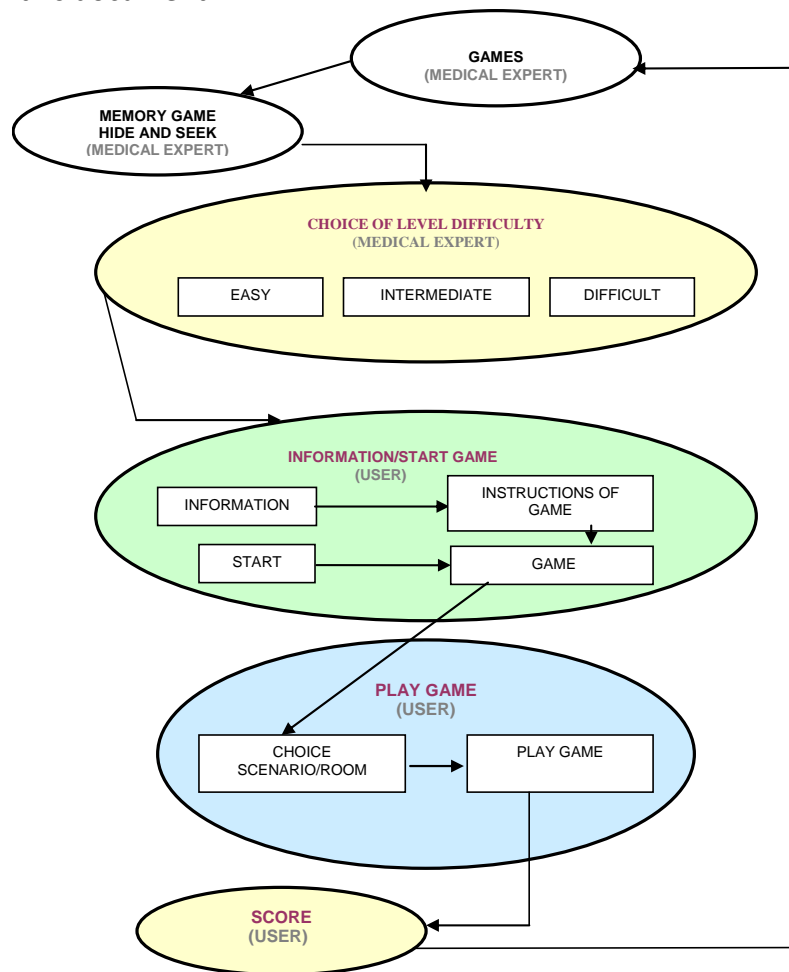


Figure 14: Workflow of the Hide and Seek Game

4.12 Specification of Additional Games

As already outlined previous paragraphs have provided specifications for eleven out of the twenty seven games defined in WP2. These eleven games are the ones that will be first integrated into the SOCIABLE platform, as part of WP4 activities. The rest games will be gradually specified and implemented on the platform, during the evolution of the activities of the project. Furthermore, we envisage that the set of SOCIABLE games will be further enhanced after the end of the project. To this end, this paragraph has provided a blueprint for creating games in terms of:

- o A baseline workflow including four main steps namely selection of difficult level, reception of instruction, play of the game and scoring. These four steps/processes are replicated within the above games and will be also replicated for future games. The aim of this "best practice" workflow is to allow elderly end-users acquaint themselves with a common flow across all different games. This will facilitate users' in switching between games (without any need for a "learning curve").

- An initial "look and feel" (as reflected in the screenshots), which will be further refined and explained in the scope of D4.1 (dealing with the user interfaces).
From a technical/technological perspective all the games will be implemented based on the implementation technologies and principles outlined in earlier sections.
Through not exhaustive, we believe that the above list of eleven games set a basis for the later implementation and elaboration of all the games to be integrated in SOCIABLE. This will be reflected in coming WP4 deliverables dealing with the technological implementation of the services.

5. Technical Specification of Social Interaction Services

5.1 Technical Specification of the "Book-of-Life"

5.1.1 Short Description

The "Book of Life" application is a personal diary, containing life experiences, emotions and thoughts, which is created by the elderly user through the SOCIABLE ICT platforms. Through the book of life, elderly users are able to share information about their life (e.g., photos, sounds) with other users. The book serves as both a collective memory and an individual memory.

The main objectives of this application are:

- To enhance self-concept and self-esteem, improve mood states, and prevent social isolation
- To find other users with the same kind of hobbies, interests, etc.
- To help the elderly to improve his/her communication capabilities and/or increase the number of his/her social contacts.

Each book has 4 different chapters, corresponding to different life stages (infancy, adolescence, adult, maturity). The application enables the customization of the content of the different chapters (texts, photos, sounds). For each chapter the user has a set of default questions, photos and sounds, but it is possible to include new ones provided by the user or his relatives.

5.1.2 Workflow

In this section the workflow of the "Book of Life" application is described.

Step 1: Pre-session assessment. When starting the application the elderly is asked for his current mood and level of isolation.

Step 2: Menu. After the assessment the main menu of the application appears: The elderly has 4 different possibilities:

- **New book of life:** The elderly (alone or in a group) can create a new book of life. He has to select a chapter and then complete it by adding text, photos and sounds. The elderly can indicate if he wants to share this chapter with other SOCIABLE users.
- **See books of life:** The elderly can see an existing book of life. He can see one of his books (individual or group) or a book of other user who has decided to share his book with other SOCIABLE users. The different chapters of the selected book are shown.

- **Modify book of life:** The elderly can choose one of his books (individual or group) for modifying a chapter. The different chapters of the book are shown. At each chapter the user has the possibility of modifying the text, the photos and the sounds included. The user can also indicate if he wants to share this chapter with other SOCIABLE users.
- **Friend management:** The elderly can find users with common hobbies and become friends. The elderly can share his books with friends.

Step 3: Post-session assessment. When exit the menu the elderly is asked again about his current mood and level of isolation. After that the application is closed.

The following diagram shows graphically the workflow previously described.

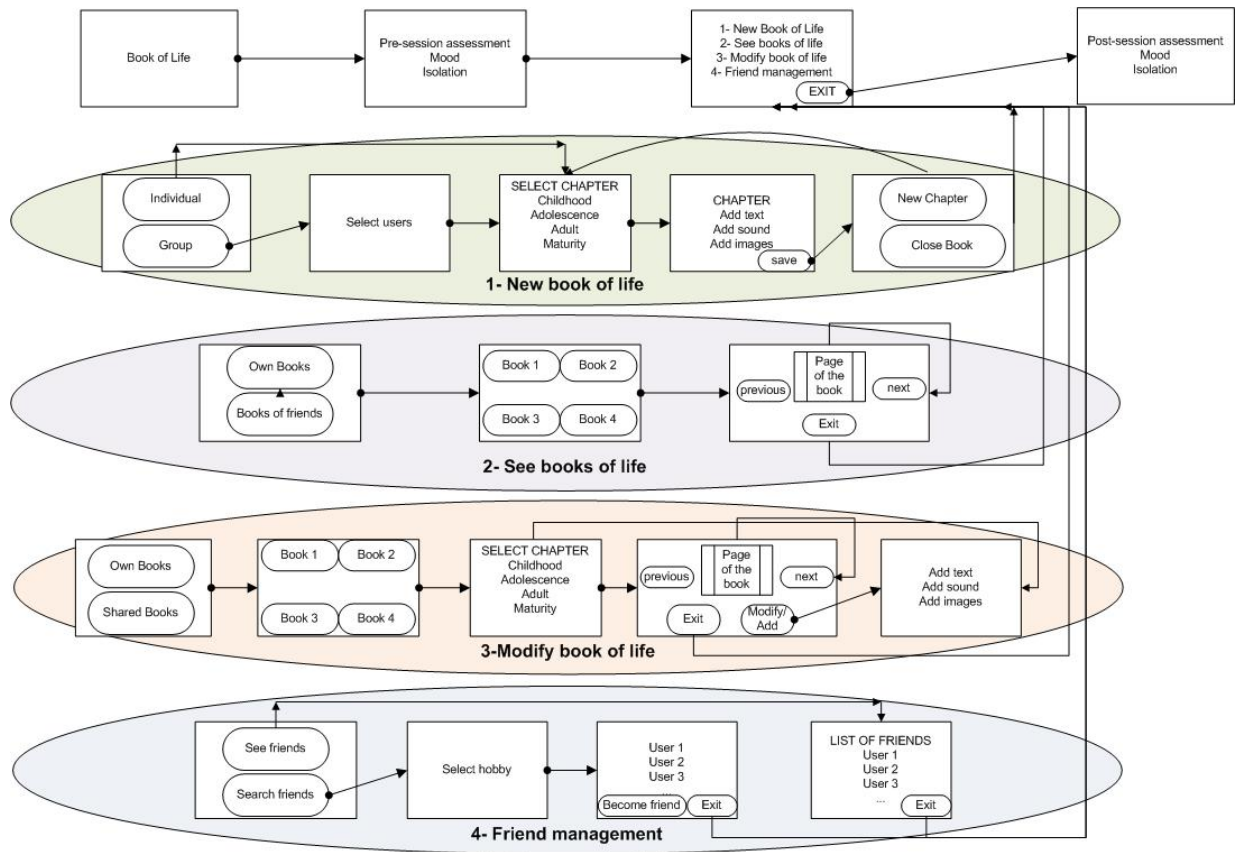


Figure 15: Workflow of the “Book-of-Life” application

5.1.3 Specification of Implementation Technologies and Interfaces to the Platform

The “Book of Life” application should be compatible with both platforms used in SOCIABLE: Microsoft Surface and TabletPC. For this reason it is based on the SOCIABLE technologies previously mentioned in this document. These technologies are:

- The Microsoft Surface SDK (Software Development Kit), used in order to implement surface based applications and services. The selection of the surface hardware platform imposes certain choices with respect to implementation technologies. It imposes for example the use of the Microsoft Surface SDK, in order to develop MS Surface applications.
- The Windows Presentation Framework (WPF), exploited in the scope of the surface applications implementation on both the surface table and on the surface PC.
- Microsoft C#, used as the main programming language of the SOCIABLE platform and services.

5.2 Technical Specification of the Communication Services

5.2.1 Short Description

The "Communication Services" enable the elderly to communicate with other SOCIABLE elderly/users including family, friends and caregivers.

The main objective of this application is to promote social activation helping to maintain the already existing relationships, and also to start new relationships with other users.

The application offers the elderly users the following possibilities:

- **Sending and receiving messages:** the users can communicate by email.
- **Video-conferencing:** the user is able to call other users or relatives.
- **Internet Navigation:** the user is able to open a browser and navigate through internet.

5.2.2 Workflow

In this section the workflow of the "Communication services" application is described.

Step 1: Menu. When starting the application the main menu appears: The elderly has 2 different possibilities:

- **Call/Video-call:** This option opens the skype program and allows the users to call other user, to start a video-conference with other user or to chat with other user.
- **Internet:** with this option the elderly can navigate through internet.

Step 2: Exit. This option closes the "communication services" application.

The following diagram shows graphically the workflow previously described.

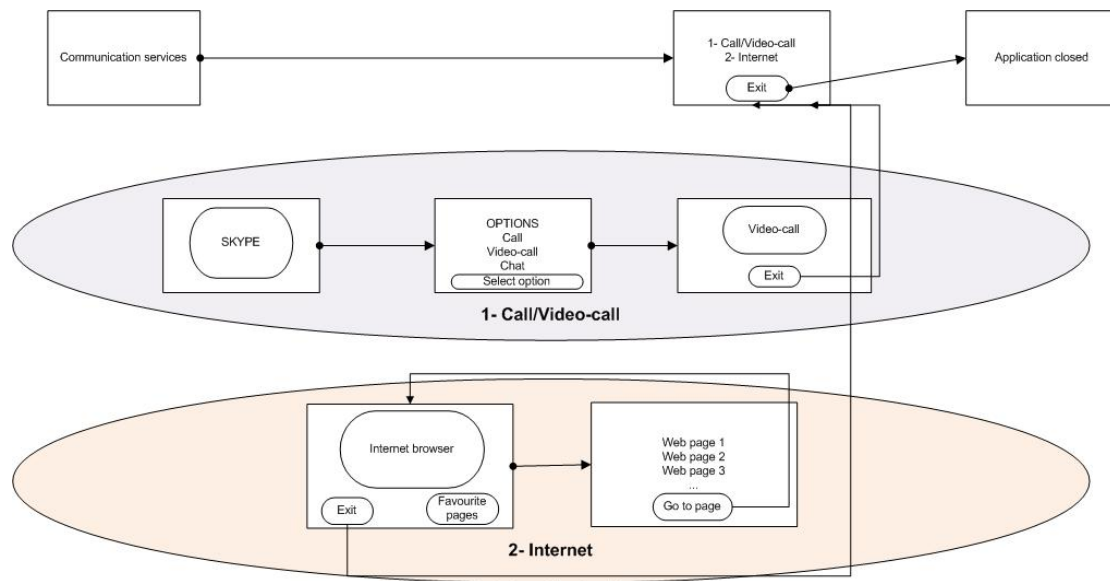


Figure 16: Workflow of SOCIABLE Communication Services

5.2.3 Specification of Implementation Technologies and Interfaces to the Platform

Some of the services previously described (e-mail and video-conference) imply the need of paying a server. On one hand the e-mail service requires a mail server for the creation of accounts, message management, storage, etc. On the other hand the video-conference service has the problem that is not possible to know the customer infrastructure (i.e. we do not know if he is behind a firewall, router, the kind of security he has, etc). For this reason is not possible to use a client-to-client connection since we had to open ports on a network not controlled by us. To solve this and skip the infrastructure problems a server (e.g. flash server) is needed.

To avoid paying for the mentioned servers the integration of SKYPE is being considered instead of e-mail and video-conference.

As commented in the case of the "Book of Life" the "Communication services" application is based on the SOCIABLE technologies: Microsoft surface SDK, Windows Presentation Framework and Microsoft C# as programming language. These technologies have been previously described in this document.

6. Technical Specification of SOCIABLE Services to Health Professionals

The services for Health Professionals will supply all features needed to manage the Sociable system, including features to manage users and patients' data, games, training programs and "book of life" management.

Back-Office application supplies the following features to the two actors of the system:

- Administrator
- Medical Expert

Medical expert will access to:

- Patient management
- Training program management (including customization of game difficulty level)
- Patient Social activation management

Administrator will access to:

- Game management
- Back-office users and permissions management
- Generic Management (Care centers, cities, region, country, etc.)

The back-office application consists of two parts:

- A web application, the "Back-office Console", used by medical expert and administrator to manage all parts of the system described above.
- A core of services (implemented in a web-service) used by all applications: Games, Book of Life and Back-office console.

6.1 Configuration of Cognitive Training Services

6.1.1 Short Description

The configuration of training services consists of the following activities:

- Patient management:
 - o creating and editing
 - o statistics consultation
- Training program management:
 - o definition of activities for a program with game assignment with a specific level of difficulty
 - o consultation and editing of the results of an activity
- Game management:
 - o creation and editing of levels of difficulty
 - o resource customization (texts, images and sounds)
 - o cognitive skills consultation

The main activity is the creation of the training program and its training session. Each training program includes sessions which could be performed in the care center or at home. Each session consists of activities that include the execution of a game or a social activation activity. To define an activity the medical expert selects a game for the elder defining a specific difficulty level, depending on the cognitive skills of the elder.

After execution of the activity by the elder, the medical expert can examine the result of the performance.

6.1.2 Workflow

In this section the workflow of the "Training Program creation" and "Game difficulty customization" is described.

The workflow of the **Training Program creation** is shown in the diagram below.

To create a Training Program the medical expert will execute the following steps:

Step 1: Login. (Not showed in the diagram) The medical logs in the console application.

Step 2: Patients selection. The medical expert selects the elder to be treated with the training program (1).

Step 3: Training Program Creation. The medical expert inserts the training program data(5).

Step 4: Session Creation. The medical expert inserts the session data: "execution date" and "name of session".

Step 5: Activity Creation. The medical expert creates an activity selecting a game for the activity and the difficult level of the game (11).

Step 6: Save Training Program. The medical expert may add additional sessions and activities by repeating steps 4 and 5. At the end, the medical expert saves the training program.

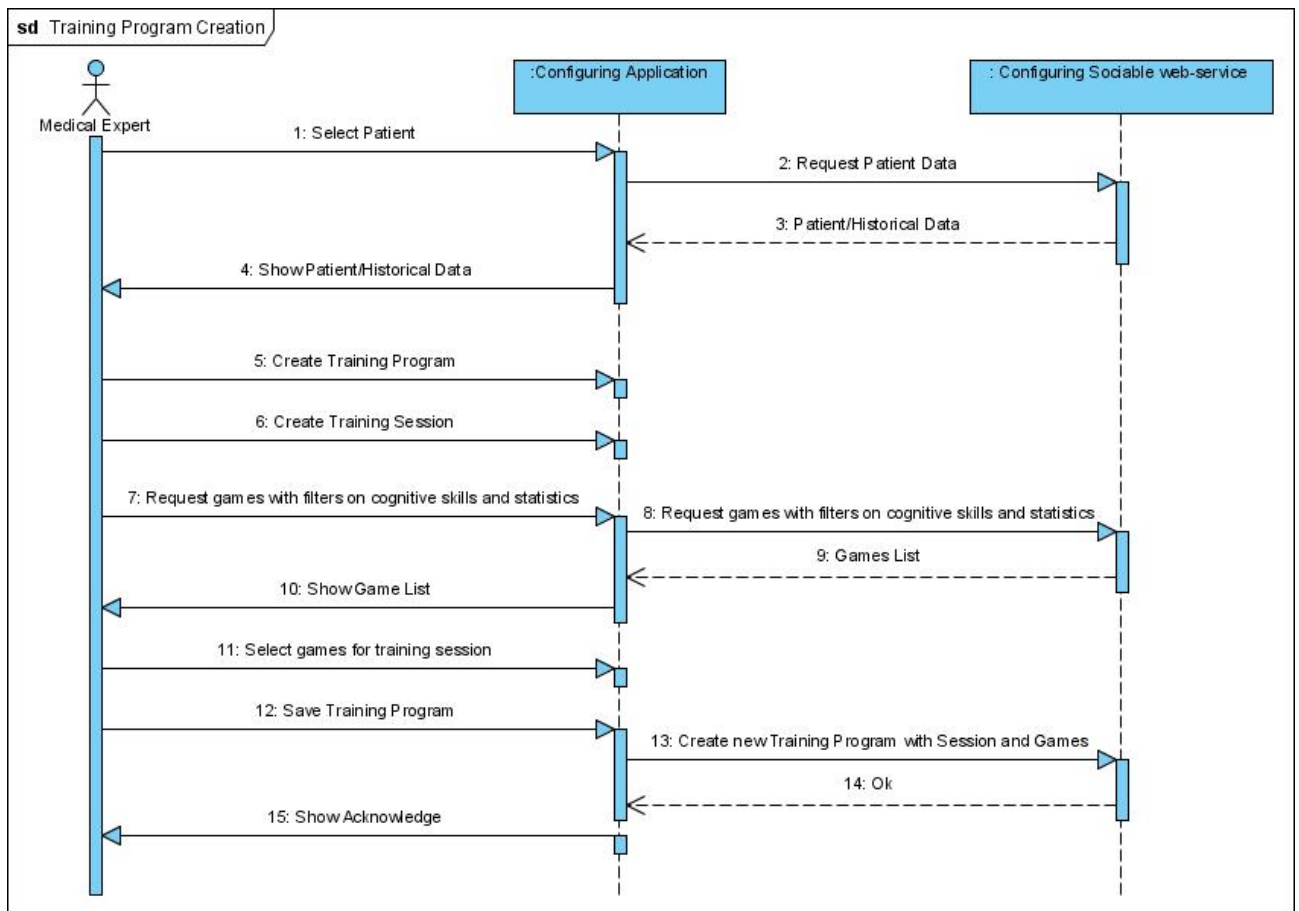


Figure 17: Sequence Diagram Associated with the Creation of a SOCIABLE Cognitive Training Programme

To customize the game difficulty, the medical expert will execute the following steps:

Step 1: Login. The medical expert logs in the console application, if not yet authenticated.

Step 2: Game selection. The medical expert selects a game from the list of games installed in the system.

Step 3: New Parameter Group creation. The medical expert clicks on the tab "Parameter Group", the application shows the list of Game Parameters, each of which represents a specific level of difficulty; in this tab the medical expert clicks on the button "Create new".

Step 4: Parameter values input. The medical expert fills the parameter with some values to create a new difficulty level (i.e. a game "hide and seek" can be customized with the number of items to find).

Step 5: Game saving. Finally the medical expert clicks on the "Save" button to confirm the changes.

6.1.3 Specification of Implementation Technologies and Interfaces to the Platform

This section will describe the implementation options of the console and of the web service.

The system is based on the Service-Oriented Architecture; all applications: Games, Book of Life and Back-office console, will access to the core of service via web-service, like shown in the diagram below.

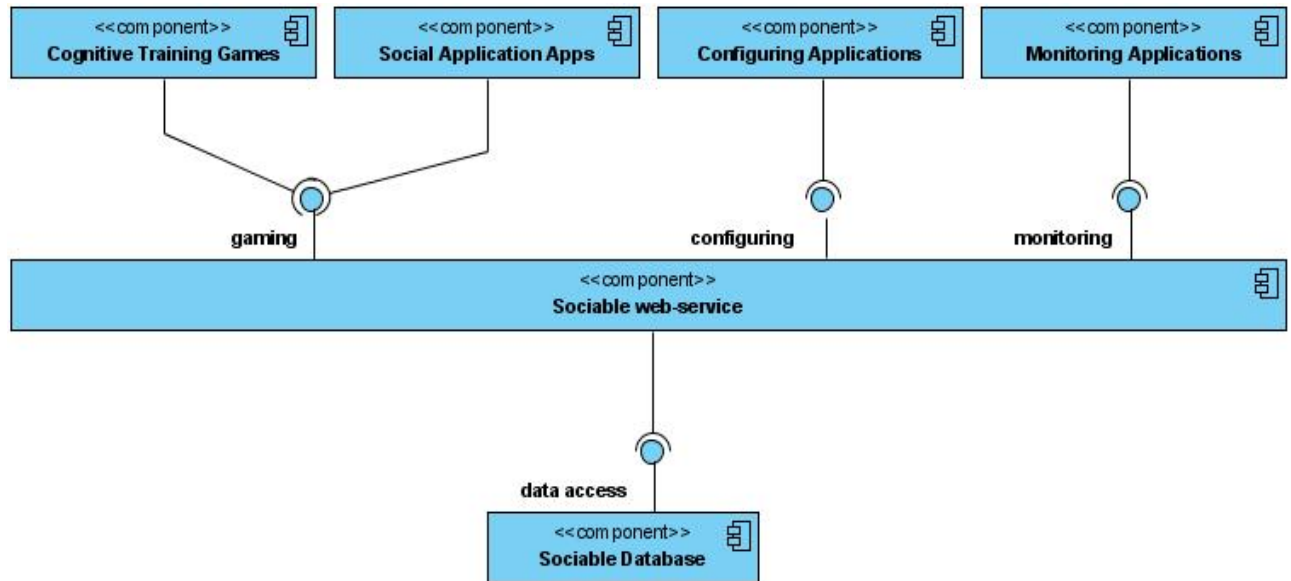


Figure 18: Component Diagram depicting the components of the Back-office applications offering services to the professionals

The SociableConsole is a web-application built using the MVC (Model View Controller) design pattern, with the version provided by Microsoft in .Net framework. The MVC pattern isolates the "domain logic" (the application logic for the user) from input and presentation (GUI), permitting independent development, testing and maintenance of both parts.

The Sociable Console accesses the service provided by the Sociable web-service. The Sociable web-service provides all features of the back-office; the web-service technology allows the use of services from different systems, like Java, PHP, etc. The access to the database is made using an ORM (Object Relational Mapper) framework (NHibernate), that presents the following advantages:

- Allows the use of different DBMS systems, like SQL Server, MySql, etc.
- Reduces the amount of code to be written.

6.2 Configuration of Social Activation Services

6.2.1 Short Description

The configuration of social activation services consists of two features:

- uploading multimedia contents on the system (images, sounds and texts), contents that will be available to patients and medical experts for the use in the book of life;
- specifying characteristics of a patient (hobbies, friends, relatives).

6.2.2 Workflow

To upload multimedia contents the medical expert will execute the following steps:

Step 1: Login. The medical expert logs in the console application, if not yet authenticated.

Step 2: Multimedia type selection. The medical expert selects the kind of content to upload (images, sounds or texts); the system displays the content already uploaded on the system.

Step 3: Patient selection. The medical expert selects the elder owner of the specific contents.

Step 4: Multimedia type selection. The medical expert selects the resource (image, sound or text) browsing the local file system.

Step 5: Upload confirmation. The medical expert confirms the upload of the resource.

To specify the characteristics of a patient, the medical expert will execute the following steps:

Step 1: Login. The medical expert logs in the console application, if not yet authenticated.

Step 2: Patient selection. The medical expert selects the patient to edit.

Step 3: Patient edit. The medical expert adds hobbies, specifies friends and relatives of the patient.

Step 4: Patient saving. Finally the medical expert clicks on the "Save" button to confirm the changes.

6.3 Monitoring of the Elderly Record (Cognitive Performance, Cognitive / Functional Assessment)

The console of the back-office will provide the tools for monitoring the elderly performance and trends of cognitive skill levels; these tools will be based on the capabilities of "Crystal Reports", the application embedded in the development environment "Visual Studio", used to design and generate reports. Upon instruction of the medical experts, the ICT experts will design and implement a series of reports specifically oriented to the monitoring and evaluation of the results of the trials; these reports will be the same for all the pilot sites in order to allow a correct comparison of data.

7. Conclusions

SOCIABLE is characterized by two very broad threads of technological development. One of the threads relates to the technological development of the SOCIABLE surface platforms, which includes a rich set of middleware and software functionalities that provide the baseline infrastructure for the execution of the SOCIABLE services. The other thread relates to the technological development of the SOCIABLE services themselves, i.e. the services offered by the SOCIABLE platform to elderly users and professionals. This deliverable has described the main technical specifications of the SOCIABLE services, which will be taken into account for their integration/implementation over the SOCIABLE surface platforms.

The technical specification of the SOCIABLE services lies on:

- The specification of the technologies to be used for integration and implementation. Given the choice of the Microsoft Surface platform, these technologies were inevitably centered around Microsoft-oriented technologies and frameworks, such as WPF, .NET and SQL Server. In addition to being compatible with the surface platform, these technologies are state-of-the-art, and robust, while also featuring longevity, which are good properties for the sustainability of the SOCIABLE results.
- The specification of technological components that could be reused to accelerate implementation and integration, a practice which is full in-line with the pilot-centered character of the SOCIABLE project. Several components of the Surface SDK will be reused to this end, in order to obviate the need for very low-level programming of the surface capabilities.
- The specification of the main technical principles that will be followed for the implementation of the services. These technical principles aim at maximizing the added-value of the surface platform, through exploiting features that are not available or even possible through other devices.
- The specification of an initial set of cognitive games along with their workflows. Most important, this specification is accompanied by a presentation of the baseline workflow that will be followed / inherited by all SOCIABLE games.
- The specification of the technological implementation of the Book-of-Life application. This can be seen as another game implementation.
- The specification of the back-office application in terms of its main functionalities.

These specifications have all been provided as part of this deliverable. They are compatible to the services definition in WP2 and the SOCIABLE platform selection. Hence, they provide a sound basis for the integration activities within WP4 of the project.

8. References

- [1]. Microsoft Corporation, "User Experience Guidelines - User Interaction and Design Guidelines for Creating Microsoft Surface Applications", July, 2, 2009.
- [2]. "Benefits of touch experiences." <http://www.elotouch.com/Solutions/CaseStudies/benefitswp.asp>. Berk, Laura E., and A. Winsler. Scaffolding Children's Learning: Vygotsky and Early Childhood Education. National Association for the Education of Young Children, 1995.
- [3]. Buxton, Bill. "Multi-Touch Systems that I Have Known and Loved." <http://www.billbuxton.com/multitouchOverview.html>. Buxton, Bill. Sketching User Experiences. Morgan Kaufmann, 2007. Csíkszentmihályi, Mihaly. Flow: The Psychology of Optimal Experience. 1991.
- [4]. Deep-zoom program. Available from Hard Rock Memorabilia. <http://memorabilia.hardrock.com>.
- [5]. Eames Foundation. "Powers of Ten." <http://www.eamesfoundation.org>. Jacob, R.J.K., A. Girouard, L.M. Hirshfield, M.S. Horn, O. Shaer, E.T. Solovey, and J. Zigelbaum, "Reality-Based Interaction: A Framework for Post-WIMP Interfaces." In Proc. ACM CHI 2008 Human Factors in Computing Systems Conference, 201–210. ACM Press, 2008. Koster, Ralph. Theory of Fun for Game Design. Paraglyph, 2004. Lidwell, William, Kristina Holden, and Jill Butler. Universal Principles of Design. Rockport Publishers, 2003. Norman, D. The Design of Everyday Things. Basic Book, 1988. Perkins, D. N. "What Constructivism Demands of the Learner." In Educational Technology, 19–21. September 1991.
- [6]. "Qualitative versus Quantitative Research." <http://wilderdom.com/research/QualitativeVersusQuantitativeResearch.html>. Raven, M.E. and A. Flanders. "Using Contextual Inquiry to learn about your Audiences." In Journal of computer documentation, 1–13. 20(1), 1996.
- [7]. "Research methods." http://en.wikipedia.org/wiki/Research_methods. Schwartz, Barry. The Paradox of Choice. Harper Collins, 2005.
- [8]. Usability descriptions. <http://www.usabilityfirst.com/methods/cogwalk.txt>.

Appendix – Workflows of the Cognitive Games Specified in the paragraph Technical Specification of Cognitive Training Games

In the following figures we provide the workflows of the earlier presented games. Note that all the games follow the same workflow structure (i.e. the baseline workflow), which is also described in section 4 above.

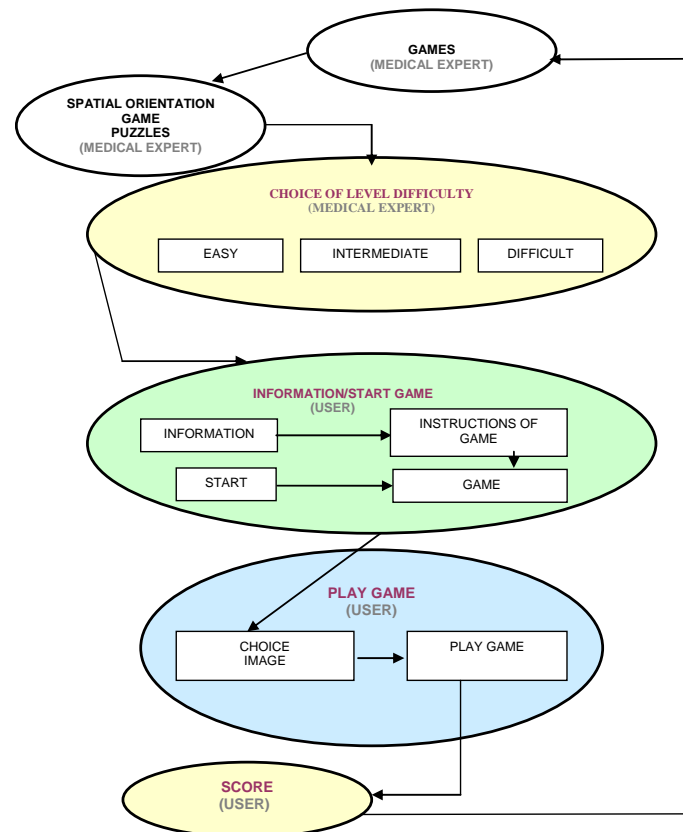


Figure 19: Workflow of the "Puzzles" Game

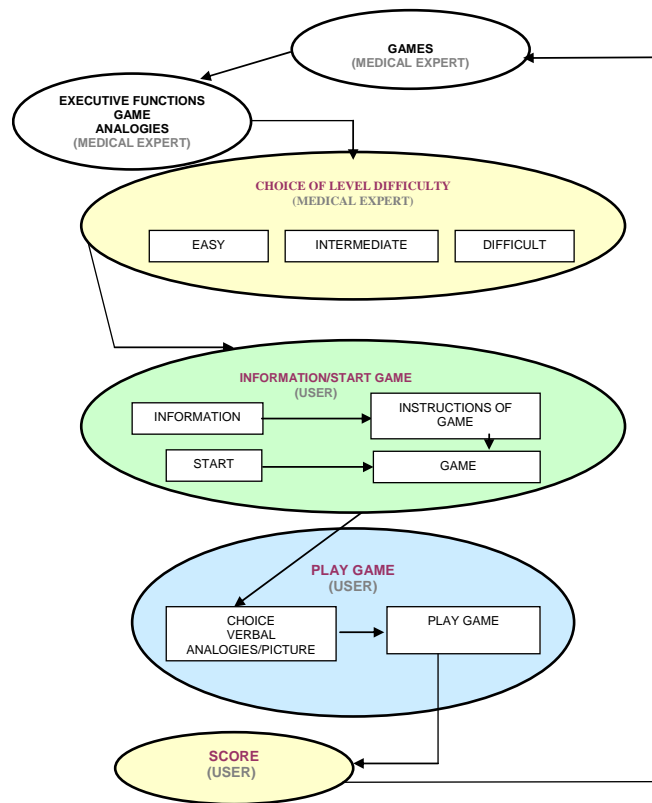


Figure 20: Workflow for the "Analogies" Cognitive Training Game

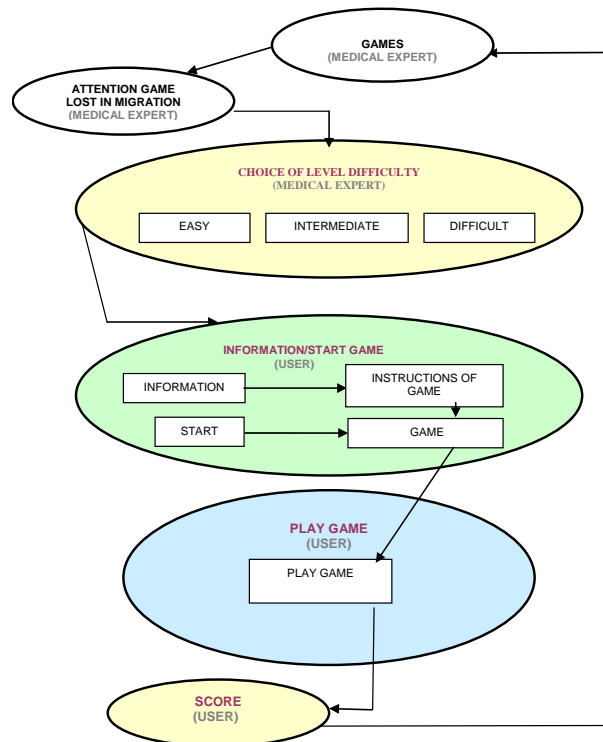


Figure 21: Workflow of the "Lost in Migration" Game

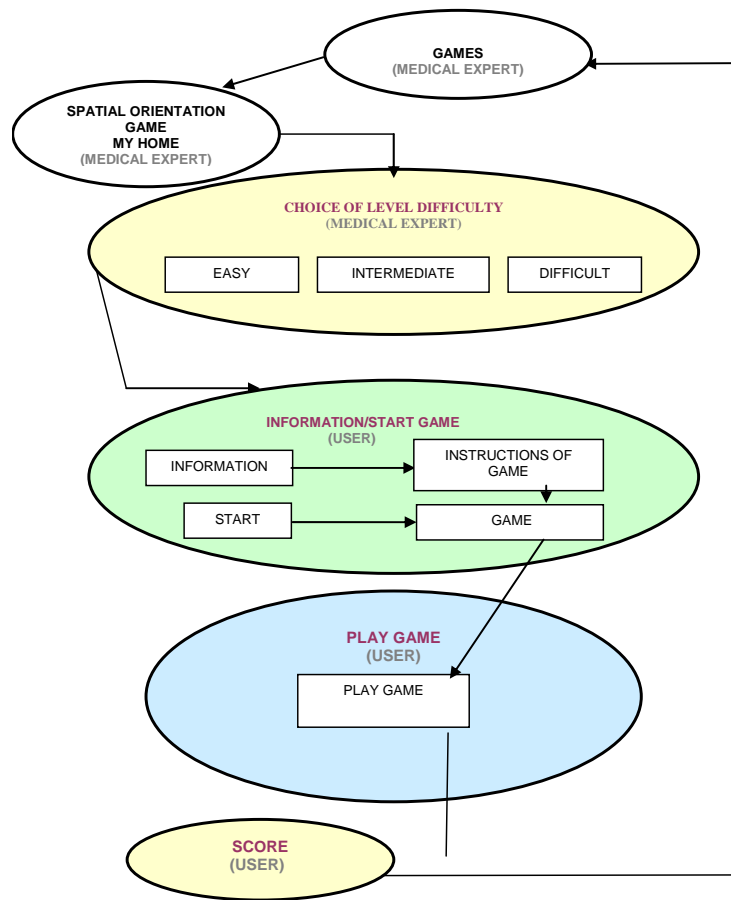


Figure 22: Workflow of the “My Home” Game

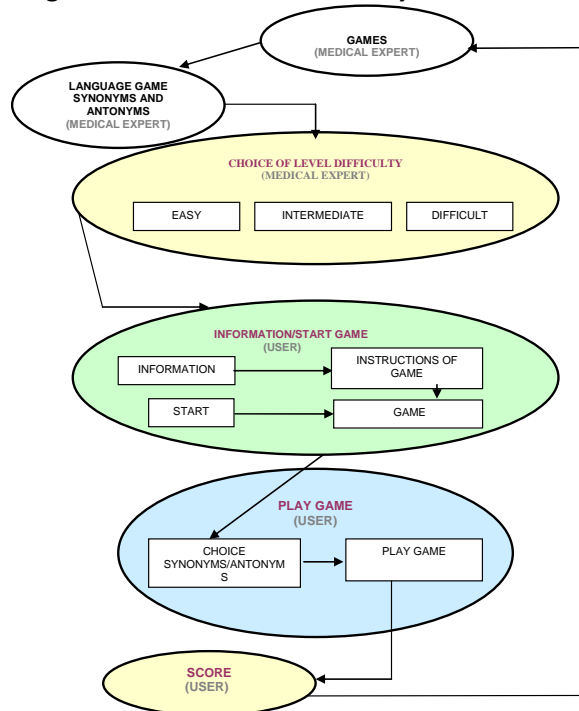


Figure 23: Workflow for “Synonyms and Antonyms” Cognitive Training Game

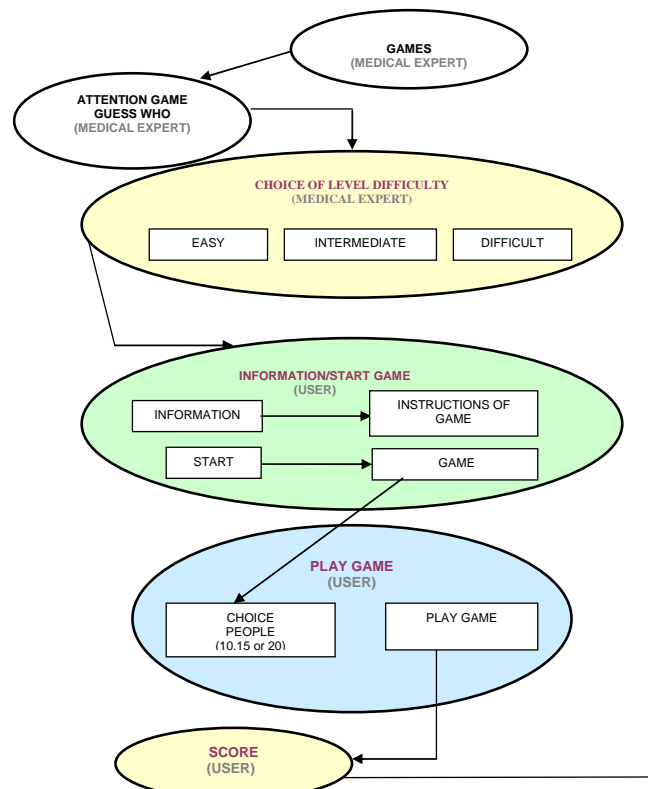


Figure 24: Workflow of the "Guess Who" Game

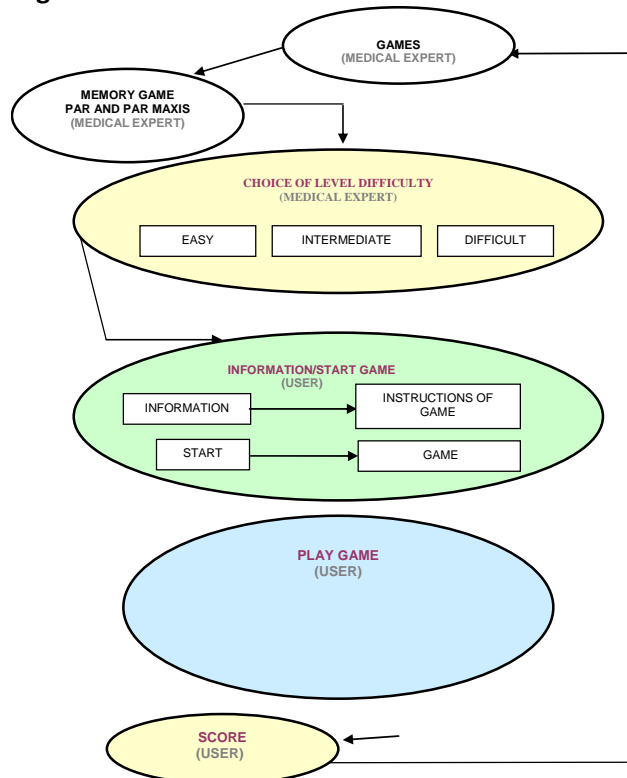


Figure 25: Workflow for the Par and Par Maxis Game

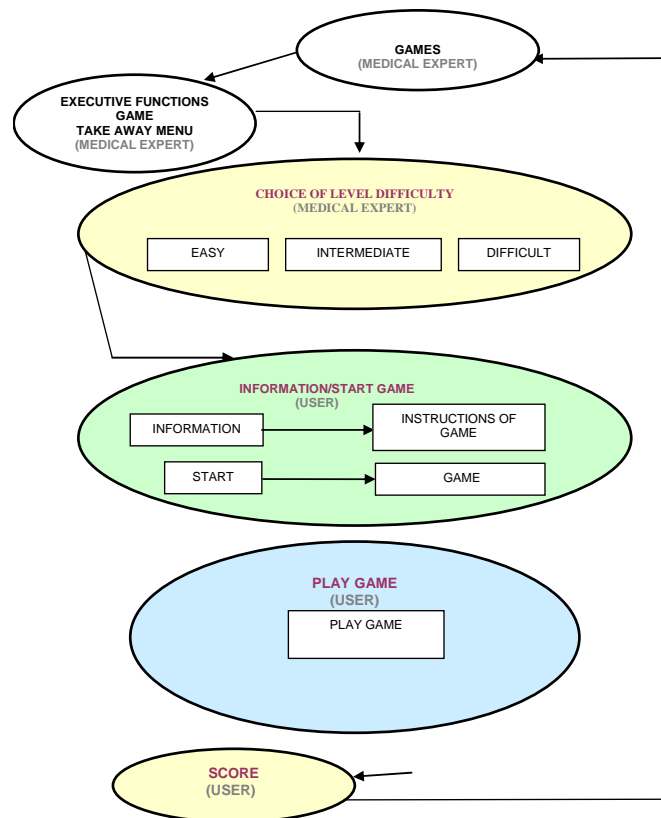


Figure 26: Workflow for the "Take Away Menu" Game

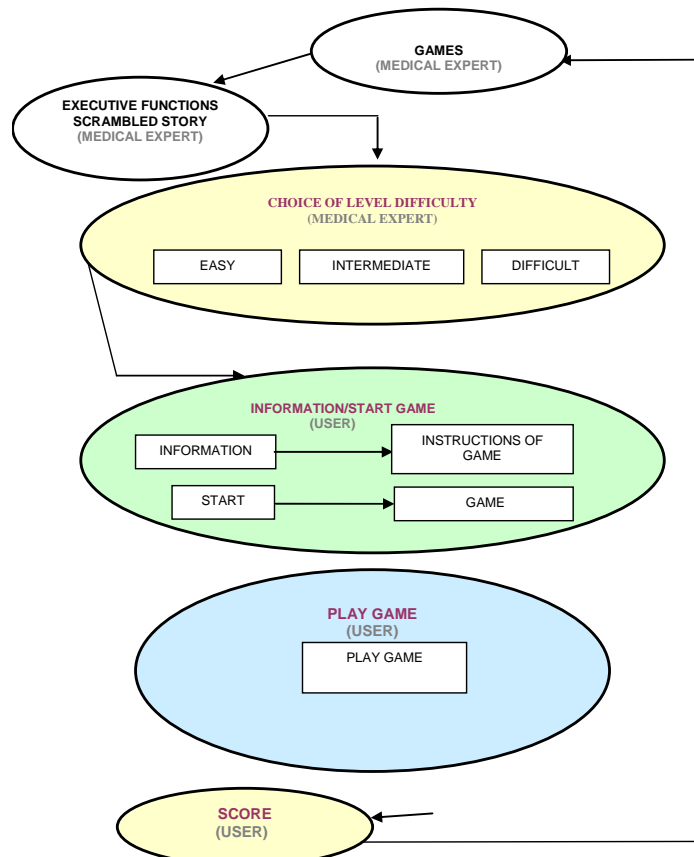


Figure 27: Workflow of the "Scrambled Story" Game