Simple Smart Homes Web Interfaces for Blind People

Marina Buzzi[1], Barbara Leporini[2] and Clara Meattini[2]
[1]*Institute of Informatics and Telematics, IIT-CNR, via Moruzzi 1, Pisa, Italy*
[2]*Institute of Information Science and Technologies, ISTI-CNR, via Moruzzi 1, Pisa, Italy*

Keywords: Home Automation, Usability, Accessibility, Visually Impaired, Web Interfaces, Screen Reader.

Abstract: Last-decade great advances in technology have contributed to make home smarter and more comfortable, especially for people with disabilities. A lot of low cost solutions are available on the market, which can be controlled remotely by a Home Automation System (HAS). Unfortunately, the user interfaces are usually designed to be visually oriented which can exclude some user categories, like those who are blind. This paper focuses on the design of usable Web user interfaces for Home Automation Systems, with a special attention to the functions as well as the interface arrangement in order to enhance the interaction via screen reader. The proposed indications could inspire other designers to make the user experience more satisfying and effective for people who interact via screen reader.

1 INTRODUCTION

Visually-impaired people may experience obstacles and issues when interacting with software and hardware components. Speech technology, screen reading software and multimodal user interfaces have been proposed to overcome those access barriers (Stephanidis, 2009). However, the interaction is not always particularly easy for a person who is blind. The user interfaces (UIs) should be designed not only following accessible principles, but importantly offering a usable experience.

Several accessibility and usability guidelines have been proposed in the literature in order to enhance the interaction with UIs, including the Web (Boldú, et al. 2017). Nevertheless, accessibility issues still exist when interacting via assistive technology (Power et al., 2012). Consequently, research focuses on methodologies and tools for improving accessibility of user interfaces and related services.

Despite an increasing focus on the smart home environments within the human–computer interaction (HCI) field, there is still a lack of studies in the context for people with special needs. A Home Automation System (HAS) would enable blind people to perform everyday activities autonomously, which might be impossible or very difficult for them. For instance, checking or setting the h temperature could not be effectively possible for people who are blind due to the inaccessibility of the thermostat interface. Exploiting a remote control system based on a (Web) app, blind persons can perform checking/setting tasks autonomously. In order to enable blind users to fully and satisfactorily control their home environment, the (Web or mobile) HAS interfaces must be effectively accessible and usable via screen reader.

This study investigates how to design HAS Web interfaces to effectively support the screen reading users to handle their everyday home activities. Starting from users' preferences and requirements collected in (Leporini and Buzzi, 2018), and from the main accessibility and usability issues observed in (Buzzi et al., 2017), this work proposes a prototype of potential HAS Web Interfaces aimed at enhancing the interaction usability via screen reader. Although many studies investigated how to design an effective and usable user interface (Almeida et al., 2018; Carvalho et al., 2016; Velasco et al., 2008), Home Automation Systems have not been investigated in terms of UIs. Our research is aimed at overcoming this gap.

The main contribution of this work is to propose a methodology for a HAS interface in terms of (1) features and functions to include, and (2) arrangement and organization of the components in the interfaces for a suitable and satisfactorily interaction for screen reader users.

The paper is organized in six sections. Section 2 introduces the related work and section 3

Buzzi, M., Leporini, B. and Meattini, C.
Simple Smart Homes Web Interfaces for Blind People.
DOI: 10.5220/0006935602230230
In *Proceedings of the 14th International Conference on Web Information Systems and Technologies (WEBIST 2018)*, pages 223-230
ISBN: 978-989-758-324-7

summarises the method leading the study. Section 4 describes the study, including the requirements, the features of the prototypal user interface, and the examples illustrating the proposed solutions. A short discussion highlighting possible research direction is reported in section 5. Conclusions and future work end the paper.

2 RELATED WORK

Internet empowers home automation systems making them simpler and smarter, accessible anytime, anywhere. Smart Houses are a great valuable opportunity for people with disability to achieve independence, they encompass the mere usability and embrace the personal care and safety (Leporini at al., 2018). An accurate review of smart homes and for home health monitoring technologies show that currently the technology-readiness is still low and a strong evidence on their effectiveness as prevention tool for assisting old persons is still missing (Liu et al., 2016).

Simple routinely operations involving lights, shutters and doors are very important for people with disability. B-Live is an example of a system designed for the motor impaired (tetraplegic, paraplegic and wheelchair users) and elderly (Santos et al., 2007).

Nowadays natural interaction that exploits voice and tact senses is expanding. T Vocal interaction to control home automation has been early investigated with telephone interface (Sandweg et al., 2000). The introduction of sounds in home automation may improve user experience by delivering information quickly (Liu et al., 2016).

The integration of voice engine technology is a promising research field, which needs to be carefully evaluated in actual contexts. It shows great potential as voice commands would be optimal for blind people. Vocal assistants such as Google or Alexa by Amazon, are recently emerging. However, in case of the elderly person a vocal interaction encourages a lazy lifestyle, that might provoke a rapid physical and cognitive degradation (Portet et al., 2013). Thus designing for elderly it is fundamental to promote healthy way of life particularly for the ageing population in Europe.

Various solutions have been proposed to overcome difficulties for blind people (Brady et al. 2013). In the last decade, low-cost built-in modular systems emerged such as Fibaro (www.fibaro.com) or easy-to-build Arduino solutions (www.arduino.cc, an open-source project). These systems enable control via a computer, smartphone or tablet using the Internet network infrastructure.

Alternatively sensors and actuators embedded in everyday objects, smart home appliances and furniture, RFID systems are also used. Smart homes empower people with disabilities when usable interfaces are offered, facilitating social activities and monitoring health conditions (Brady et al., 2013). Caregivers can be alerted when anomalies are detected. A multi-modal gestural and vocal interface system for controlling distributed smart homes appliances, has been proposed by (Jeet et al., 2015) to enable hands-free operation to people with motor disability. Blind people can also exploit these interactions.

3 METHOD

Starting from the users' requirements and preferences, and the accessibility issues and design suggestions investigated in the previous works (Buzzi and Leporini, 2017; Leporini and Buzzi, 2018), we identified some crucial needs and consequently features to include in the HAS design, and a simple template for enabling a usable interaction via screen reader. For instance, rather than providing a single and all comprehensive interface overview for showing the lighting on/off, different simplified views are made available for a more compact rendering better navigable via screen reader and keyboard as well.

Briefly, our methodology can be summarized in the following procedure:

1. Analysis of the users' requirements as well as main accessibility and usability issues experienced via screen reader interacting with a popular commercial system for Home Automation (Fibaro);
2. Identification of the main features as well as functions to include in the design of user interface of a HAS;
3. Selection of the components to include in the user interfaces especially in terms of arrangement and organization according to the activities related to a smart home.

For the last step, the main accessibility and usability guidelines have been considered: the W3C WCAG and WAI-ARIA (W3C, 2014), specific usability design suggestions proposed in (Buzzi et al., 2017), and (Leporini et al., 2018).

During the entire design-cycle two totally blind users have been involved in various pilot tests to evaluate the interface prototype. The Internet Explorer and Mozilla Firefox Browsers, and the screen reader Jaws for Windows (http://www.freedomscientific.com/) have been used to test the Web interfaces.

4 THE WEB INTERFACE

4.1 Requirements

In a survey involving 42 visually impaired people - 32 of them totally blind - (Leporini and Buzzi, 2018), the users expressed clear preferences for:

(1) handling the status of the devices such as lighting system in an effective and simple way (checking and turning). This interest was expressed by the 81% of the participants living alone or in specific situations. So we included this functionality in our system.

(2) Chance to manage home devices such as thermostat, washing machines, electric system, garden and any type of activity that can be controlled remotely.

Based on these preferences, the main functionalities as well as components to be included in the user interface have been identified. Customization, simple tasks and different devices/status views are important features driving the User Interface design.

The usability issues encountered when navigating the Fibaro Web interfaces via screen reader (Buzzi et al. 2017) derive from the lack of

(I) Interface partitioning in logical sections, to allow the user in better orientating among the page components;

(II) Meaningful context-independent labels of links and buttons (e.g. “lamp 1 on”, rather than just “on”). This is especially useful to easily and clearly understand the interactive items and corresponding tasks carried out;

(III) Compact contents and simple interaction to perform common tasks like reading elements (e.g., the device status) or performing simple actions (e.g., change status).

4.2 Features and Functions

The usability definition by ISO 9241 (www.iso.org/) takes into account the specified users when interacting with a system in a context of use to reach specified goals. In our case, the goals are: to be able to easily and satisfactorily check the status of the devices as well as to set/perform specific actions such as turning lights on/off or activating predefined scenarios.

Accordingly, the main tasks and objectives to meet in the proposed interfaces are identified in:

(1) Checking which devices/sensors are on/off; for this activity the interface should provide quick functions and immediate control of the most used devices.

(2) Turning on/off the devices/sensors; as in the previous case, the user should be able to carry out specific tasks through a quick and easy interaction. The user should perform quickly very common activities, such as turning all lights.

(3) Getting an overview of information about the home/room/device status. The user interface should provide useful views and links to quickly read the status and to get a fast overview of the HAS. A summary of the status for the home and for each single room could be useful functionalities provided by the interface.

With this in mind the following main characteristics and functionalities have been included in the UIs design:

1. **Menus, regions and heading levels.** Menus and regions have been applied to deliver the information via a logical partitioning of the contents. Regions and heading levels have been introduced for structuring the interface content.
2. **Home summary and rooms details.** The user interface offers a quick summary of the home status (i.e., how many lights are on, which devices are active, etc.); besides, the user has the opportunity to visualize the details of a single room.
3. **Global search and “ready-functions” for checking devices status.** The user can perform any search about the devices status, but to simplify the interaction for frequent actions, the user interface provides “ready functions” to carry out specified tasks (e.g. which lights are still on).

Therefore the prototype includes the following user interfaces:

- Map view for the status of the home in summary or by rooms details
- Views by Devices status
- Items and quick actions for specific tasks like ‘turn on/off specific devices’ or ‘check the open elements’.

In the following we illustrate these features showing how the proposed solutions applied to develop the Web user interfaces can simplify the interaction via screen reader.

4.3 Web User Interfaces

In this section some UIs are presented to introduce our Web prototype. The features have been arranged into different pages that the user can quickly select by a simple click on the tab panel on the top of the interface. The navigation menu is composed of 4 items: Home, Devices, Scenarios, and Settings. In the following we describe the 'Home' and 'Devices' Interfaces.

4.3.1 Map View

Map views are usually used to show and interact with the home structure in a friendly graphical way. Frequently they are not so accessible via screen reader. In our study we intended to use a map view in order to consider how to make it both accessible and usable for screen reading users. Thus, let us to act on the map of a home with few rooms, to apply accessibility design (See Figure 1).

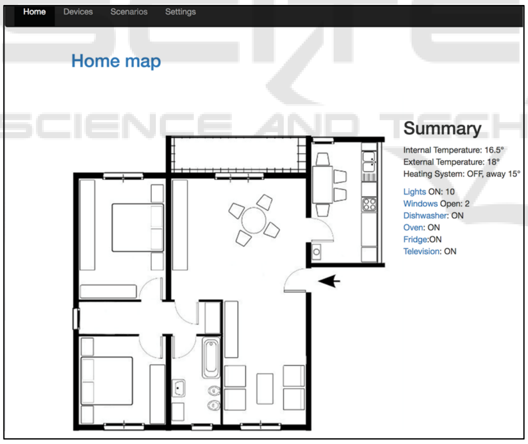

Figure 1: Home Automation Web Interface (map view): Home status.

When designing the user interface, we intended to: (1) reproduce a graphical attractive map for sighted people, (2) create a room list clearly detectable by the screen reader, and (3) show a summary of the home status (global overview) or room by room (details).

When the user clicks on the link "Home" from the "menu", the map of the home is automatically shown (See Figure 1). The user can now get a summary of the current home situation with the status for the most important devices. By clicking on a room, the summary for that room is instead displayed. In Figure 2 the kitchen room has been selected by the user.

In Table 1 and 2 the content announced by Jaws when accessing respectively the home map and the kitchen are reported. The italic indicates the content announced about user interface element typology (e.g., link, button, heading level, etc.).

By navigating the map via Tab key, the screen reader informs the user about typology for each element. For example, Jaws reads "graphic living room map image", so the user becomes aware is navigating a map. The user can interact with that element to get more details on it. By navigating via arrow keys, the home map is detected by the screen reader as a list of elements, i.e. the names of the rooms. Each room is announced in a single line with its name followed by the 'clickable' attribute. The user can quickly go along the list and select one room by just pressing the space bar key (i.e. to click on that item).

Table 1: Text announced by Jaws: the home status.

My home: Status
Menu navigation region
list of 4 items
Link Home
Link Devices
Link Scenarios
Link Settings
list end
Menu navigation region end
main region
Home Map *clickable*
Double bedroom *clickable*
Kitchen *clickable*
Living room *clickable*
Bedroom *clickable*
Bathroom *clickable*
Terrace *clickable*
Heading level 2 Summary
list of 9 items
Internal Temperature: 16.5°
External Temperature: 18°
Heating System: OFF, away 15°
Lights ON: 10
Windows Open: 2
Dishwasher: ON
Oven: ON
Fridge: ON
Television: ON
list end
main region end

For the general overview, we included an interface enhancement with a summary of the home status in the home page, such as the internal and external temperature, heating system status, number of the total lights on, number of the opened shutters, status of the washing and Dishwasher machines. This summary should be configurable according to user preferences. By clicking on a room (for instance the kitchen) you can see the turned on devices for that room: the detailed list is shown in Figure 2.

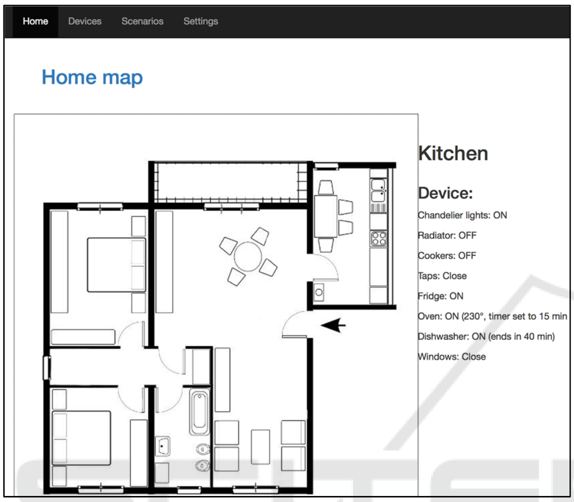

Figure 2: Home Automation Web interface (map view): Kitchen status.

Table 2: Text announced by Jaws: the kitchen status.

My home: Devices
…
Heading level 2 Kitchen
Device:
Chandelier lights: ON
Radiator: OFF
Cookers: OFF
Taps: Close
Fridge: ON
Oven: ON (230°, timer set to 15 minutes),
Dishwasher: ON (ends in 40 min)
Windows: Close
main region end

4.3.2 Device Status

The user could be interested to get the status of all devices belonging to a same category (e.g., shutters), or getting an overview about all the devices in a specific same status (e.g., lights turned on). To this end, the user interface should allow to perform a search by "category" or "status". Usually, like the system described in (Buzzi et al., 2017) this information is shown all together in the same user interface so that looking at a glance the user can get a lot of information. Unfortunately this approach is not appropriate for VIP because of too many links, buttons and contents. There is not an overview of the contents and sequential navigation requires a lot of effort via arrow and Tab keys. More flexibility interfaces aid VI users in make interaction faster and satisfying. Thus, from the menu, the link "Devices" gives the opportunity to perform various types of searches.

To check windows if it rains a blind person usually needs to move into all the rooms and "touch by hand" window by window to check if it is open. Otherwise, if a Home Automation system provides a support to check all the open windows, the user can go directly only towards those to be closed. Thus the user interface should provide a very quick way to carry out this frequent daily activity.

In this perspective, our prototype has been designed to offer this opportunity: (1) performing a quick search according to the different parameters (typology, status and room), and (2) showing the results so that the user can check in a list the results and quickly change their status in one click.

Figure 3 shows the results related to the search performed according to "Windows" devices, "On" status, and "all rooms". Table 3 reports how the screen reader Jaws reads the user interface and the results. All devices related to search settings are listed each by line. The space bar can be used to change the status (when possible).

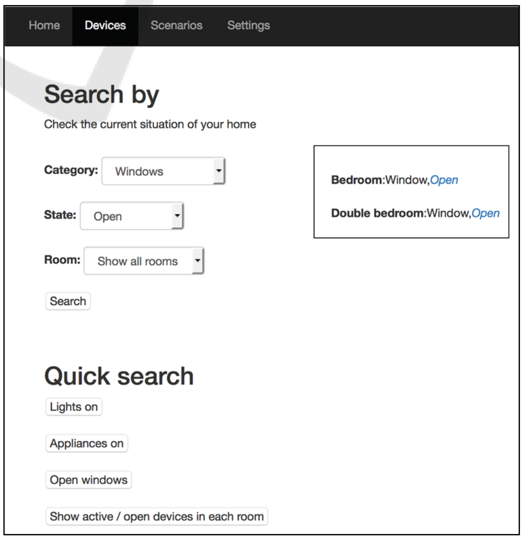

Figure 3: Home Automation Web interface: Search by devices.

Table 3: Content announced by Jaws: search by devices.

My home: Devices ... *Heading level 2* Search by Check the current situation of your home Windows *Listbox item selected* Open *Listbox item selected* Show all rooms *Listbox item selected* Search *Button* *Heading level 2* Results: Bedroom:Window, Open Double bedroom:Window, Open *Heading level 2* Quick search Lights on *Button* Home appliances on *Button* Open windows *Button* Show active / open devices in each room *Button* ...

The user interface also offers ready buttons to perform some more common queries, in order to get very quickly a list of lights on, open shutters and windows, and so on. Specifically, the available functionalities we proposed for a "quick search" are: "Lights on", "Home appliances on", "Open windows", and the opportunity to "Show active/open devices in each room". These functionalities are aimed at simplifying some checks avoiding to set up the queries for very common tasks each time.

For example, by clicking on the "Lights on" button, a list of the lights turned on is shown after the button it-self (Figure 4).

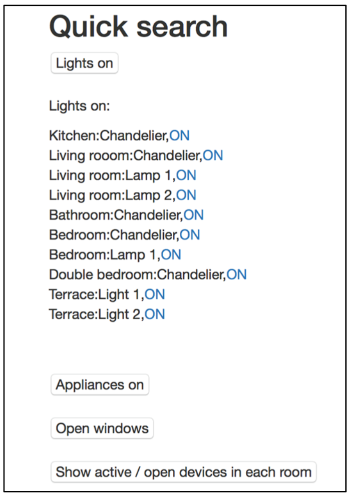

Figure 4: Quick Search: results shown when pressing the "Lights on" button.

The user can immediately read it just via the arrow keys. To turn off a light it is enough pressing the space bar on the corresponding item. Table 4 reports the text announced by Jaws when pressing the "lights on" button.

Table 4: Content announced by Jaws when pressing the "Lights on" button.

My home: Devices ... *Heading level 2* Quick search Lights on *Button* Lights on: *list of 10 items* Kitchen:Chandelier, ON Living rooom:Chandelier, ON Living room:Lamp 1, ON Living room:Lamp 2, ON Bathroom:Chandelier, ON Bedroom:Chandelier, ON Bedroom:Lamp 1, ON Double bedroom:Chandelier, ON Terrace:Light 1, ON Terrace:Light 2, ON *list end* Home appliances on *Button* Open windows *Button* Show active / open devices in each room *Button* *main region end*

4.3.3 Partitioning and Info

In our prototype we introduced the page content partitioning and additional information to provide to the screen reader. For this purpose, in the design we specifically used:

- The WAI-Aria regions for partitioning the Web content within the page. This allows the user to get an overview of the page via a specified command (Ctrl+JawsKey+R). An example is reported in Figure 5: four regions have been used for structuring the "Devices search" page. This has been designed by using the 'region role' with the title attribute: <div role="region" title="results">.
- The WAI-Aria live regions to inform the screen reader (and so the user) when a dynamic region updates. In this way as soon as the content changes the screen reader automatically reads it. In this way the user does not have to explore the page to detect if something has changed. In our prototype we used the live regions both in the 'home status' and 'devices' pages: in the first

case the live region has been used for the ‘summary’ area; in the second one, for the results content. An example of the code is <div role= "contentinfo" aria-live="polite" aria-atomic="true" aria-relevant="additions text">.

- Each page title contains a sort of current path. For example, the pages have the titles like “My home: Status”, for the “Home” page; “My home: Devices”, for the page designed for the search; “My home: Scenarios”, to select predefined scenarios. The page title is the first element read by the screen reader when a new page is loaded; so it can help the user in understanding more easily the current page. This can be designed by simply writing a sort of path in the <title> tag (we used <title>My home: Devices</title>, for the ‘Devices page’).

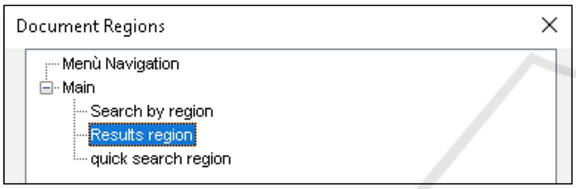

Figure 5: List of the ARIA regions captured by Jaws.

4.4 Pilot Evaluation

In order to understand if our approach can be appropriate, three visually-impaired skilled users have been involved to test the proposed Web interfaces. The users were asked to interact with the system by assigning them three tasks: (a) check (if any) what are the devices on in the living room; (b) Turn off all the lights, and (c) detect what is the internal temperature. The users accomplished all tasks in a natural setting, using their desktop to navigate the prototype, one at time. A think-aloud protocol was been applied and researchers observed and recorded any comments. Interesting suggestions about the interfaces emerged during the pilot, useful to enhance their usability.

Shortcuts and additional predefined buttons could further improve the use of the system; the search by devices and status could be better enhanced by adding specific links to simplify the selection. Some commands like “turn off all lights” should be available near the results of a search in order to switch off the devices listed. Some scenarios and opportunity to set them should be added to the user interface.

5 DISCUSSION

In this work we focus on how to propose and arrange the interface components to perform easily and quickly common tasks such as checking and changing the device status, to have different views and to simplify the interaction. The user interface can show the status summary for the entire home, and a more specific details once a room has been selected. This enables the user to get a quick home overview about the device status. Briefly, three main keys driven our Web interface design:

(1) reducing the irrelevant information; This can affect positively especially the sequential reading and interaction via screen reader and keyboards;

(2) showing only the contents related to the current goal; This enhances the user interaction especially with the common commands. In addition, the user can focus on the useful functionalities and elements related to the context.

(3) simplifying the repeated tasks via specific functionalities and commands. By providing predefined tasks can widely improve those activities which may be carried out frequently.

Through the proposed prototype, our goal was focused on simplifying the Web interfaces in order to reduce the contents to show, and at the same time to avoid elements of no use. The advantages of such an approach is confirmed by the study (Giraud et al., 2018): by avoiding redundant and irrelevant information there are substantial benefit regarding participants' cognitive load, performance, and satisfaction.

The short pilot test revealed the feasibility of the proposed approach, even if further improvements are needed. However, it is very important to check usability of prototype via smartphone since it may be more convenient for the blind when moving around the home.

6 CONCLUSION

In this work a prototype of Web user interfaces for a Home Automation System has been proposed to enhance interaction via screen reader. Smart homes have the potential to empower the individual independence, provided that the interaction with the devices and services are simple for people, regardless their abilities.

The design of our prototype suggested to have a simple and interactive interface by (1) reducing the page content to the current task and (2) arranging the functions and commands according to the main goals. The main goals in Home Automation can be summarized in checking and update the device status, and providing information in a very simple manner.

IoT technology offers new opportunities. More research is needed to investigate new chances and main challenges for simplifying interaction for people with disability and building cheap accessible and inclusive smart homes: the optimal positioning of intelligent objects in the home, the range of valuable services for special need users, the seamlessly combination of smart devices and personal health systems. As a consequence, the interfaces are clearly affected by the integrated system to deliver the services and contents to the users. Predefined scenarios and configurations can become two important enabling keys in such a context.

As future work smart components and Web interfaces to support context-aware scenarios and customized settings will be investigated.

In order to enhance usability interaction of smart homes for all, and to favour the individual autonomy additional research on smart home accessibility has to address different areas such as intellectual disability and learning problems by exploiting cognitive psychology and social principles, and investigating new ways for enhancing safety and security.

REFERENCES

Almeida, L. D., & Baranauskas, M. C. C. (2018). A Roadmap on Awareness of Others in Accessible Collaborative Rich Internet Applications. In *Application Development and Design: Concepts, Methodologies, Tools, and Applications (pp. 479-500). IGI Global*.

Boldú, M., Paris, P., Térmens i Graells, M., Porras Serrano, M., Ribera, M., & Sulé, A. (2017). *Web content accessibility guidelines: from 1.0 to 2.0*.

Brady E., Morris M. R., Zhong Y., White S., & Bigham J.P. (2013). Visual challenges in the everyday lives of blind people. In *Proc. of the SIGCHI Conference (pp. 2117-2126). ACM*.

Buzzi, M., Gennai, F., & Leporini, B. (2017, November). How Blind People Can Manage a Remote Control System: A Case Study. In *Int. Conference on Smart Objects and Technologies for Social Good* (pp. 71-81). Springer, Cham.

Carvalho, L. P., Ferreira, L. P., & Freire, A. P. (2016). Accessibility evaluation of rich internet applications interface components for mobile screen readers. In *Proc. of the 31st Annual ACM Symposium on Applied Computing (pp. 181-186). ACM*.

Domingo MC. (2012). An overview of the Internet of Things for people with disabilities. *Journal of Network and Computer Applications*, 35(2): 584-596.

Giraud, S., Thérouanne, P., & Steiner, D. D. (2018). Web accessibility: Filtering redundant and irrelevant information improves website usability for blind users. *Int. Journal of Human-Computer Studies, 111, 23-35*.

Hwang, I., Kim, H. C., Cha, J., Ahn, C., Kim, K., & Park, J. I. (2015, January). A gesture based TV control interface for visually impaired: Initial design and user study. In *Frontiers of Computer Vision (FCV), 2015 21st Korea-Japan Joint Workshop on* (pp. 1-5). IEEE.

Javale D., Mohsin M., Nandanwar S., & Shingate M. Home automation and security system using Android ADK. *Int. journal of electronics communication and computer technology* 2013, 3(2): 382-385.

Jeet, V., Dhillon, H. S., & Bhatia, S. (2015, April). Radio frequency home appliance control based on head tracking and voice control for disabled person. In *Communication Systems and Network Technologies (CSNT), 2015 Fifth Int. Conf. on* (pp. 559-563). IEEE.

Leporini, B., Buzzi, M. (2018) Home Automation for an Independent Living: Investigating Needs of the Visually Impaired People. W4A 2018.

Liu, L., Stroulia, E., Nikolaidis, I., Miguel-Cruz, A., & Rincon, A. R. (2016). Smart homes and home health monitoring technologies for older adults: A systematic review. *Int. journal of medical informatics*, *91*, 44-59.

Portet F., Vacher M., Golanski C., Roux C., Meillon B. (2013). Design and evaluation of a smart home voice interface for the elderly: acceptability and objection aspects. *Personal and Ubiquitous Computing*, 17(1): 127-144.

Power, C., Freire, A., Petrie, H., & Swallow, D. (2012, May). Guidelines are only half of the story: accessibility problems encountered by blind users on the web. In *Proc. of the SIGCHI conference (pp. 433-442). ACM*.

Sandweg N., Hassenzahl M., & Kuhn K. (2000). Designing a telephone-based interface for a home automation system. *International Journal of Human-Computer Interaction*, 12(3-4): 401-414.

Santos V., Bartolomeu P., Fonseca J., Mota A. (2007, July). B-live-a home automation system for disabled and elderly people. In *Industrial Embedded Systems, 2007. SIES'07. Int. Symposium on (pp. 333-336). IEEE*.

Stephanidis, C. (Ed) (2009). The Universal Access Handbook. *CRC Press, 2009. Pp. 1034*

Velasco, C. A., Denev, D., Stegemann, D., & Mohamad, Y. (2008, April). A web compliance engineering framework to support the development of accessible rich internet applications. In *Proc. of W4A (pp. 45-49). ACM*.

W3C, Web Accessibility Initiative (2014). Accessible Rich Internet Applications (WAI-ARIA) 1.0. W3C Recommendation 20 March 2014.