

DESIGN AND IMPLEMENTATION OF REMOTELY MANAGED EMBEDDED DIGITAL SIGNAGE SYSTEM

Taner Arsan, Alp Parkan and Hakkı Konu

Department of Computer Engineering, Kadir Has University, Istanbul, Turkey

ABSTRACT

This paper is about a remotely managed digital signage system design considering embedded system design rules and characteristics. Digital signage system design is based on various methods of using computer and television screens as well as other kinds of display devices in ways that are as efficient as possible to provide advertising and information to people in public areas. In modern digital signage systems, the screens are divided into regions and layers, and the content on the screens is made up of several files. The goals of this study are broadcasting information, advertisements at display contents in public areas such as; subways, buses, malls, city squares and control these digital signs remotely.

KEYWORDS

Embedded System and Software, Multimedia Systems and Services, Software Engineering.

1. INTRODUCTION

Digital signage is the name of distributing digital content to different points over central headquarters. It is generally used for attracting prospects' attention in heavy traffic points and internal information distribution. Digital signage systems can run on simple hardware facilities. Simple advantages of digital signage solutions are; reducing expenses [1], being easy to reach a huge amount of people, integrating citizens with up-to-date technologies, dynamic and effective advertisement, being easy to deliver information when it is necessary. Main purpose of digital signage can be defined as "proper information at the proper time, for the proper mass."

Digital signage technologies are becoming very popular between many different business sectors [2]. Companies use digital signage for many different purposes. Digital signage solutions are creating opportunity for customizing the information that they want to deliver. Main usage purposes of digital signage are corporate communications, corporate messaging, corporate announcements, advertising and promoting products, entertainment, public information systems such as news, headlines, weather, and menu information such as digital menu boards with information on pricing, ingredients and nutritional facts. Effectiveness of Digital Signage Advertisements is one of the most important research topic of information engineering [3].

1.1 Related Works

There are lots of digital signage solutions on the market. They all use standard media file formats but they still have some problems about compatibility.

Most systems have been developed for specific digital signage projects and started to be run over small organizations and updated to larger ones. However, due to the lack of standardization, there are different ways to create and read the playlists, and scheduling files are different from each digital signage solution [4]. The ways the media players run playlists on the displays are also different from each other. In the long run, standardization will be necessary while the development of television and the Internet continues. There are huge improvements made for standardization. One of these initiatives is Point of Purchase Advertising International (POPAI), which has been around since the 1930s. POPAI has also developed a standard for playlist files. The playlists determine what is going to be played and the playlogs confirm the return of the acknowledgements about what has actually been played before. This kind of improvements have motivated and inspired us to develop an effective digital signage software.

Digital signage and advertising sectors get closer to each other in the last few years [5]. Advertisers are interested in reaching huge mass of people by spending a little amount of money. The United States is the largest market in the world; Asia-Pacific is the fastest-growing market all over the world. Digital technologies are rising in quality and falling in price so we are going to see digital signage solutions for marketing campaigns more often in daily life. There are several advantages of digital signage over traditional media like digital signage reduces advertisement expenses, enables users to customize their messages more dynamically and targeted. Traditional media is not as strict and efficient as digital signage. It is not open to sudden changes and costs much more than digital signage advertising techniques. Recently, mobile [6] and interactive [7] versions of digital signage systems have been developed.

Digital signage is a new growing sector in the world. There are lots of potential sectors which could be a market of digital signage such as advertisement, banking, health care, municipalities, municipal affiliates, hospitality and stores.

Organizations may prefer to use digital signage solutions to improve their prestige and reduce their marketing and digital signage expenses [8].

Digital signage solutions are commonly used all over the world. A 2011 study by Global Industry Analysts estimated that the global digital signage systems market would reach \$13.8 billion by 2017. Researchers said a retail boom in countries such as China, Singapore, Malaysia and Thailand, the United Arab Emirates, Hong Kong and India would help lead the way. The United States is the largest regional market worldwide; Asia-Pacific is the fastest-growing.

2. DEVELOPMENT TOOLS

The simplest definition of this study is “remotely managed digital display, typically tied in with sales, marketing and advertising”. In this paper, it is aimed to develop a user-friendly, web-based, cheap, effective and compact sized digital signage system which can be controlled and modified by the users. Digital signage management interface gives the opportunity to users, customize their own web marketing panel over any device which can access to Internet. This design brings practical solutions to digital signage sector. Simple web browsers can act like a digital advertising panel.

2.1 PHP

PHP is a server-side programming language. Created by Rasmus Lerdorf in 1995, PHP is a software development tool which is designed for filling the gap between SSI and Perl, intended to develop web applications [9]. Its basic principal is to bring web pages in dynamic content. We prefer to use PHP v5.3 in our project. As a web server, we have chosen Digital Ocean's Virtual

Private Server. When we have installed LAMP Stack v5.3, it comes as default. Version 5.3 is convenient to run this study.

2.2 Apache Web Server

Apache HTTP Web Server is a web server in basic manner [10]. On our cloud server we have LAMP stack. So we're using Apache as web server for our project. Also Apache supports a variety of features many of which implemented as compiled modules which extend the core functionality. These can range from server-side programming language support to authentication schemes.

2.3 Twitter Bootstrap

Bootstrap is a free development tool which contains HTML and CSS based templates. Bootstrap was published by Twitter which was designed for front-end development of web applications and web sites. We have used Twitter Bootstrap to reduce the time of development process. Also Bootstrap is a stable framework so it reduces the chances to make mistakes while developing the code [11]. We have chosen Bootstrap 2.3.2 because our template written by this version.

2.4 CSS3

CSS3 is used in this study to customize the front-end. CSS; in other words Cascading Style Sheet defines how to display HTML elements [12]. Cascading Style Sheets customize fonts, colours, margins, lines, height, width, background images, advanced positions and many other things. CSS gives us the advantage of controlling the layout of multiple documents from a single style sheet.

2.5 JavaScript

JavaScript (JS) is one of the most popular dynamic web programming language in the world [13]. JavaScript is used for making web pages interactive. Its syntax was inspired from C. JavaScript is used in this project because the following reasons;

- Loading and submitting data and posting new content to the server without refreshing the page.
- Customizing web page elements for resizing them and fading them in and out.
- Interactive content like audio and video.

2.6 jQuery

jQuery is a light-weight "write less, do more" JavaScript library. Main purpose of using jQuery in our project is to make the use of JavaScript easier while developing our web application [14]. jQuery is one of the most popular and extendable JavaScript framework. Many big companies like Google, Microsoft and IBM use jQuery. Digital signage system is developed with jQuery version 1.7.2 because template of the system supports the version. CSS frame work used in this study includes this version as default.

2.7 Ubuntu Server 12.04 LTS

Ubuntu is a Debian-based Linux operating system, with Unity as its default desktop environment (GNOME was the previous desktop environment). Ubuntu Server, made to be used in servers.

The server's installation CD allows the user to install Ubuntu permanently on a computer to be used as a server. It does not install a graphical user interface.

2.8 NetBeans

NetBeans [15] is an open source integrated development environment for developing; Java, PHP, C++, XML, Groovy and other different programming languages. NetBeans can run on different operating systems like Ubuntu which was our operating system while developing the project. NetBeans IDE 7.4 has been used on Ubuntu 64 bit version in our project. NetBeans can manage features like user settings, NetBeans visual library, and storage and framework wizard.

2.9 HTML5

HTML, in other words Hyper Text Markup Language, is a programming language for describing web sites. HTML5 is used in this project because Twitter Bootstrap theme supports HTML5 [13]. This language also brings practical solutions to development process. For example, <video> element was used very effectively in the project. Data attribute is also one of the crucial HTML5 features.

Admin panel theme of Digital Signage System is responsive, this is an important reason for working with HTML5 by its own.

2.10 Raspbian OS

Raspbian is an operating system which is distributed by Linux. Linux specially developed this operating system for Raspberry Pi devices [16]. Last version was published in January 2014. We used the latest version in this project. The most popular operating system for the Pi is Linux. Several Linux distributions are available for the Pi, and we chose Debian. Recently the Debian team has frozen the latest version named Wheezy, and because of the great efforts of the Raspbian team, it is available for the Pi already.

Raspbian supersedes Debian squeeze, which has been the reference operating system for the Pi for a long time. The Raspbian distribution has many advantages over all its predecessors. It is much faster, it has more recent software, and it will soon be more stable. Also, it is the preferred solution of the Raspberry team, so this book's focus is on Raspbian.

2.11 MYSQL

With over ten million installations, MySQL is probably the most popular database management system for web servers. MySQL was developed in the beginning of 1990s, now it's the most common used database engine [17]. Main reason for its success, like PHP, it's free to use. MySQL is fast and it can run on basic hardware easily. MySQL is a relational database system. Data is held on tables. We used MySQL in our project because it can run stable with Apache Web Server and PHP. It's one of the elements of LAMP stack.

2.12 Digital Ocean

Solid State Drive (SSD) only cloud hosting company – Digital Ocean is becoming very popular among web developers. Digital Ocean provides all users with high-performance SSD Hard Drives, flexible API, and the ability to select to nearest data center location. It is possible to rent a 512 MB of RAM and 20GB of SSD-powered disk space in Amsterdam for 5 dollars.

3. PROPOSED SYSTEM: IMPLEMENTATION AND METHODOLOGY

This study is very well prepared combinations of digital signage, Raspberry Pi, Raspbian, Linux Operating System, HTML, CSS, JavaScript and PHP programming technologies. The back-end side of the project is based on PHP language and the front-end side is HTML5 bootstrap. Server side of the project has to be developed on a web-server, client side is on ARM board. About the ARM board we research and decide to use Raspberry Pi. Raspberry Pi is a wallet-sized microcomputer, suitable for digital signage purposes. It is the cheapest solution for digital signage. Development stage is a major part of the project because of developing on PHP and Python. For comparison, other Linux based digital signage projects are searched, surveyed and tested. Development of this study is monitored on Github so it will be open source and progress of project will be on Web.

3.1. Technical Specifications of Digital Signage System

Main part of the study is the embedded system design which runs on Raspberry Pi. Raspberry Pi electronic board provides much more affordable digital signage system. User interface is easy to learn and creating basic media is very practical. It is not necessary to install an extra program on client device to run the mechanism. Unfortunately, Raspberry Pi has limited features in terms of speed and boot processes. Initial boot process is slow and multitasking is not available.

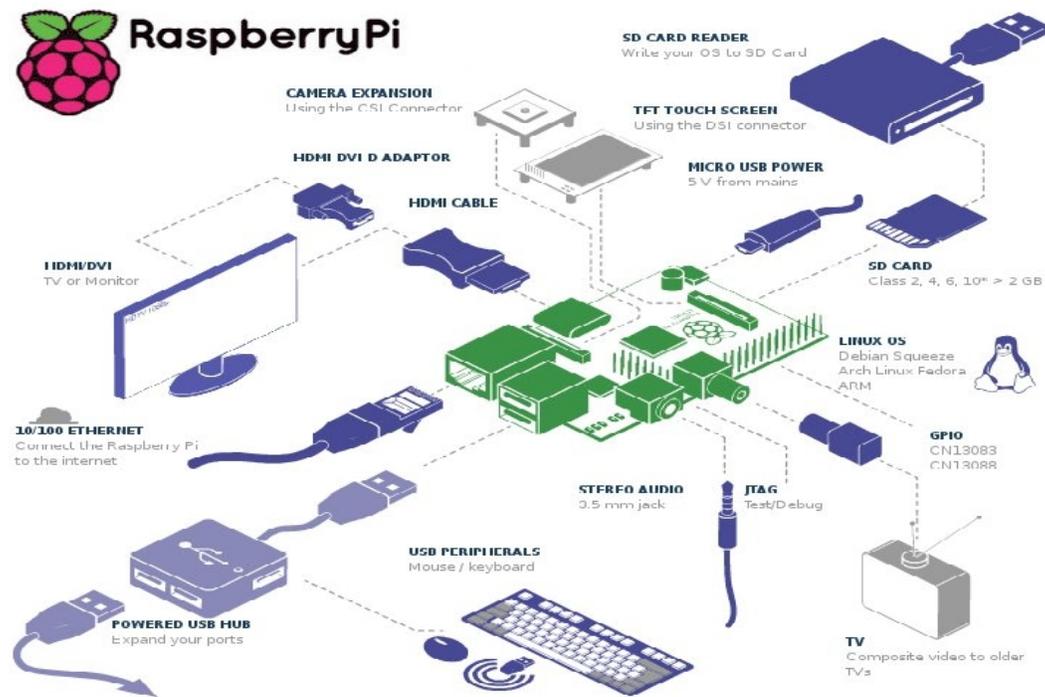


Figure 1. Raspberry Pi Model B System.

The Raspberry Pi is a credit-card sized single-board low cost computer as shown in Figure 1. It is designed on an ARM board and have I/O ports. It can be used in electronics projects, and for many of other the things that any computer does. Besides it can play high-definition video. As shown in Table 1, there are two different models available in the market: Model A and Model B. In this study, the Model B is used, because Internet connectivity over Raspberry is desired. The

Model A does not have an Ethernet port and the RAM is less than Mode B. On the other hand, the Model B has an Ethernet port and has 512 MB of RAM.

3.1.1. Boot Process and Performance

All the files necessary for booting are installed in a FAT32 partition of the SD card. Raspberry Pi has an auto boot feature. When the Raspberry Pi is plugged into electricity, the boot process starts automatically, web browser–Midori starts after initial boot process.

The GPU provides Open GL ES 2.0, hardware–accelerated Open VG, and 1080p30 H.264 high–profile encode and decode. The GPU is capable of 1Gpixel/s, 1.5Gtexel/s or 24 GFLOPs of general purpose compute and features a bunch of texture filtering and DMA infrastructure. Overall real world performance is something like a 300MHz Pentium 2.

Raspberry Pi does not have a built–in camera. But, there is camera module for Raspberry. The Camera Board is a small PCB that connects to CSI–2 camera port on the Raspberry Pi using a short ribbon cable. The camera module is capable of up taking photos up to 5 megapixels and can record video at resolutions up to 1080p 30fps. It is not necessary to use a camera module in this study, but it is possible to extend this study to an interactive advertising.

Raspberry Pi has composite and HDMI out on the board, so you can hook it up to an old analogue TV through the composite or through a composite to scart connector, to a digital TV or to a DVI monitor. The Raspberry Pi can encode (record) and decode (play) h.264 (mp4/mkv) out of the box. Sound over HDMI port is supported. Also there is a standard 3.5 mm jack for audio out on board.

Table 1. Technical Specifications of Raspberry Pi.

	Model A	Model B
System on a Chip	Broadcom BCM2835	
CPU	700MHz ARM 1176JZF–S core	
GPU	Broadcom VideoCore IV, OpenGL ES 2.0 Device capable of MPEG–2 and VC–1, 1080p30 h.264/MPEG–4 AVC decoding and encoding	
Memory (SDRAM)	256MB, shared with GPU	512MB, shared with GPU
USB 2.0	1	2 (integrated USB hub)
Video Out	Composite RCA (PAL and NTSC), HDMI Display Serial Interface for LCD panel	
Audio Out	3.5mm jack, HDMI	
Storage	SD/MMC/SDIO card slot	
Network	No connector	RJ45 Ethernet
Peripheral Connectors	8 x GPIO, UART, I ² C bus, SPI bus	
Power Rating	300mA (1.5W)	700mA (3.5W)
Power Source	5 Volt via micro USB or GPIO header	

3.1.2. Operating System and Programming

There are several open source distributions (distros) available on Raspberry Pi's web page. The NOOBS installer is the easiest and the best way for beginners, as it walks you through the download and installation of a specific distro. The recommended distro is Raspbian, which is used in our project. Raspbian is a Linux based Debian distro, it can run on Raspberry Pi's ARM processor. There is a six main Linux distros which can run on the Pi. These six main distros are:

- Raspbian (Version: January 2014)
- Pidora (Version: 18)
- Openelec (Version: 3.2.0)
- Raspbmc (Version: December 2013)
- Risch OS (Version: July 2013)
- Arch Linux (Version: January 2014)

The recommended language for Raspbian is Python. Also, any language which will compile for ARMv6 can be used with the Raspberry Pi, though; so you are not limited to using Python. C, C++, Java, Scratch, and Ruby all come installed by default on the Raspberry Pi.

The back-end side of this study is based on PHP language and the front-end side is HTML5 bootstrap. Server side of the project has to be developed on a web-server, client side is on ARM board. About the embedded system side, it is decided to use Raspberry Pi. Raspberry Pi is a wallet-sized microcomputer, suitable for digital signage purposes. It is the cheapest solution for digital signage. Development stage is a major part of the project and software development is achieved on PHP and Python. Development of this study is monitored on Github so it will be open source and progress of project will be on Web.

3.2 Development of Digital Signage System

This study includes four main packages. These packages are: Research and Pre-Development, Development of Front-End, Development of Back-End and Testing and Adaptation.

3.2.1 Research and Pre-Development

The first part is research about digital signage technology, techniques, market and how to develop a digital signage application. We decided to develop a digital signage solution for every end user. We didn't develop a stand-alone application because it's not suitable for every platform, not elastic and not scalable. So, we have chosen to develop a digital signage solution as a web-application. It's completely scalable, elastic and also suitable for every platform on every device which can reach to web. Besides, update and new release of application are more simple and user friendly in web development. First of all, we decided the template of our website. It has to be suitable for every platform so the website should be responsive. Bootstrap is the best way to develop a responsive website.

3.2.2. Development of Front-End

Login page, Register page, Dashboard which allows users to see their devices, active slides and media library, Uploaded media which makes user to see media library, Add media which makes user to add new media by clicking the add media button, Add new device which makes user to see devices and also add new device features.

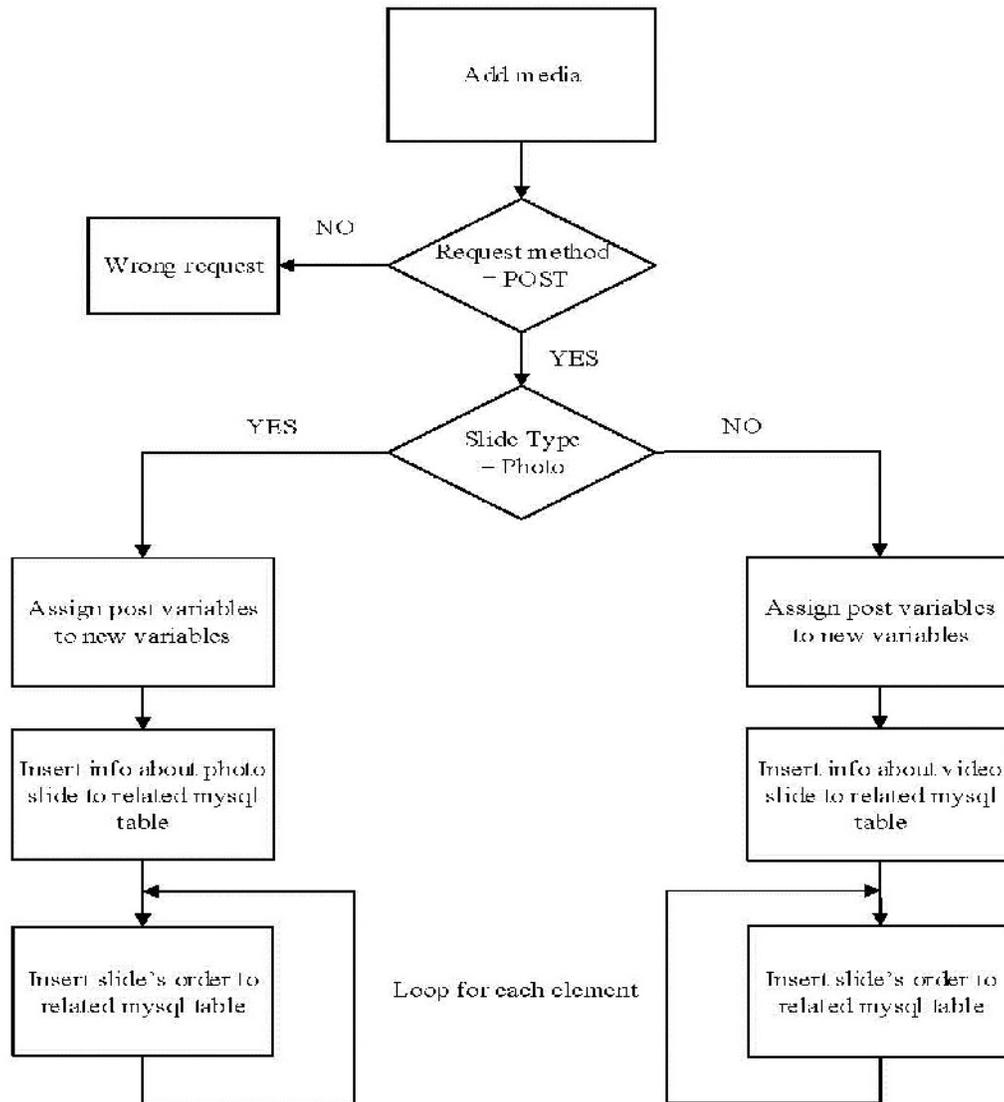


Figure 2. Flowchart of Create New Slide.

Create slides which make user to create slides at this page by selecting the media from his media library, Create video slide, Arrange media which makes user to arrange the slides in that page. Slide list which makes user to be able to see sides and assign them to any device, and create slide.

3.2.3. Development of back-end

These flowcharts are some examples of our back-end. The flowchart of creating new slide is given in Figure 2. Create new slide action working with the Post method, user selects the media from his library which will be shown in the related slide. After that system inserts slide info into database. In the final stage user arrange the slide order with the drag/drop feature and the system inserts the slide order to related database.

The flowchart of media upload is given in Figure 3. Media upload action is also working with the Post method. User selects the media which will be added to his library. After this, system checks the user's upload folder, if the user has not got one yet system creates an upload folder. Next phase, system controls the acceptability of the selected files by their extensions. If the files acceptable, system uploads the files to user's uploads folder in server.

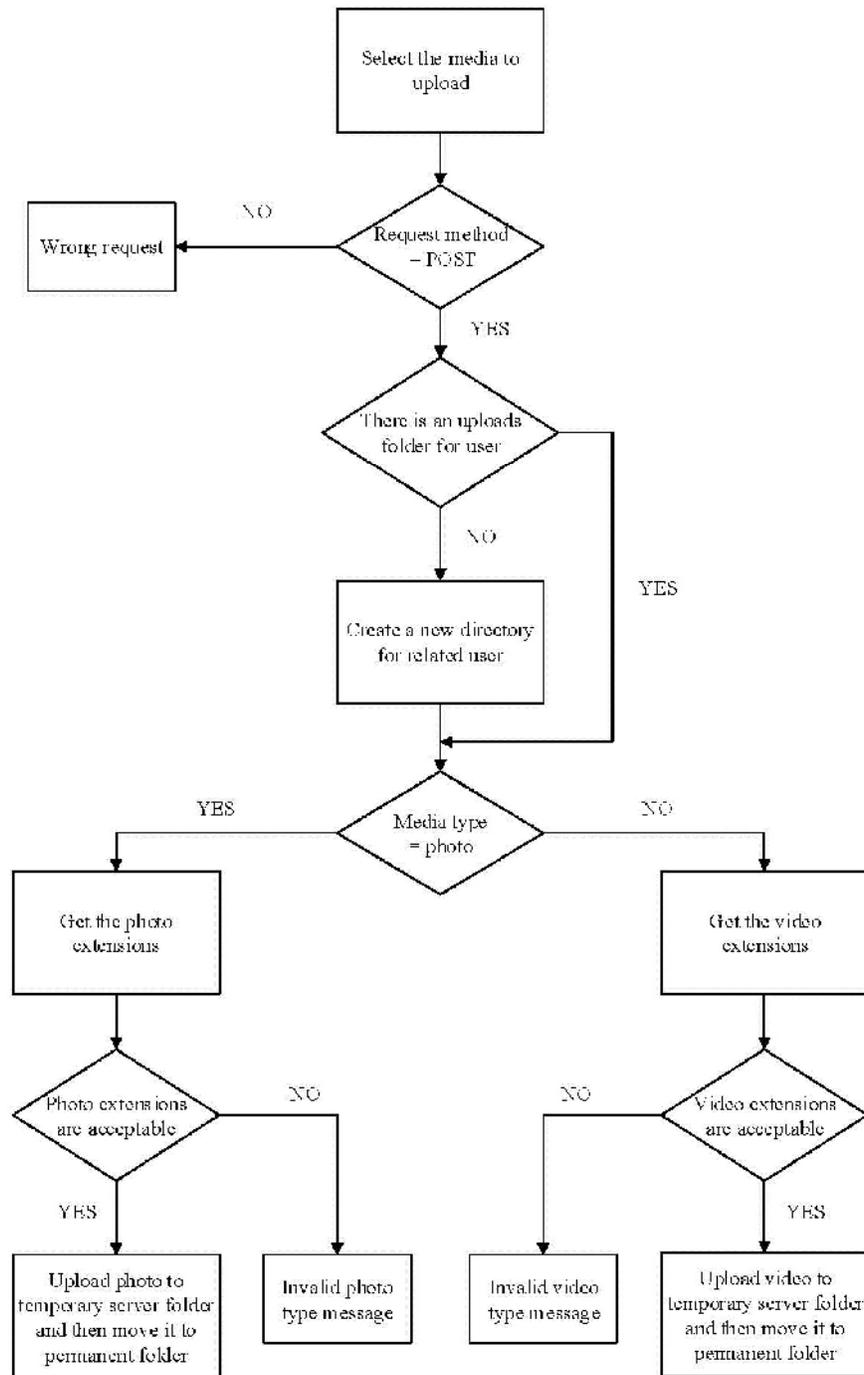


Figure 3. Flowchart of Media Upload.

The flowchart of adding new device is given in Figure 4 and the flowchart of login check is given in Figure 5.

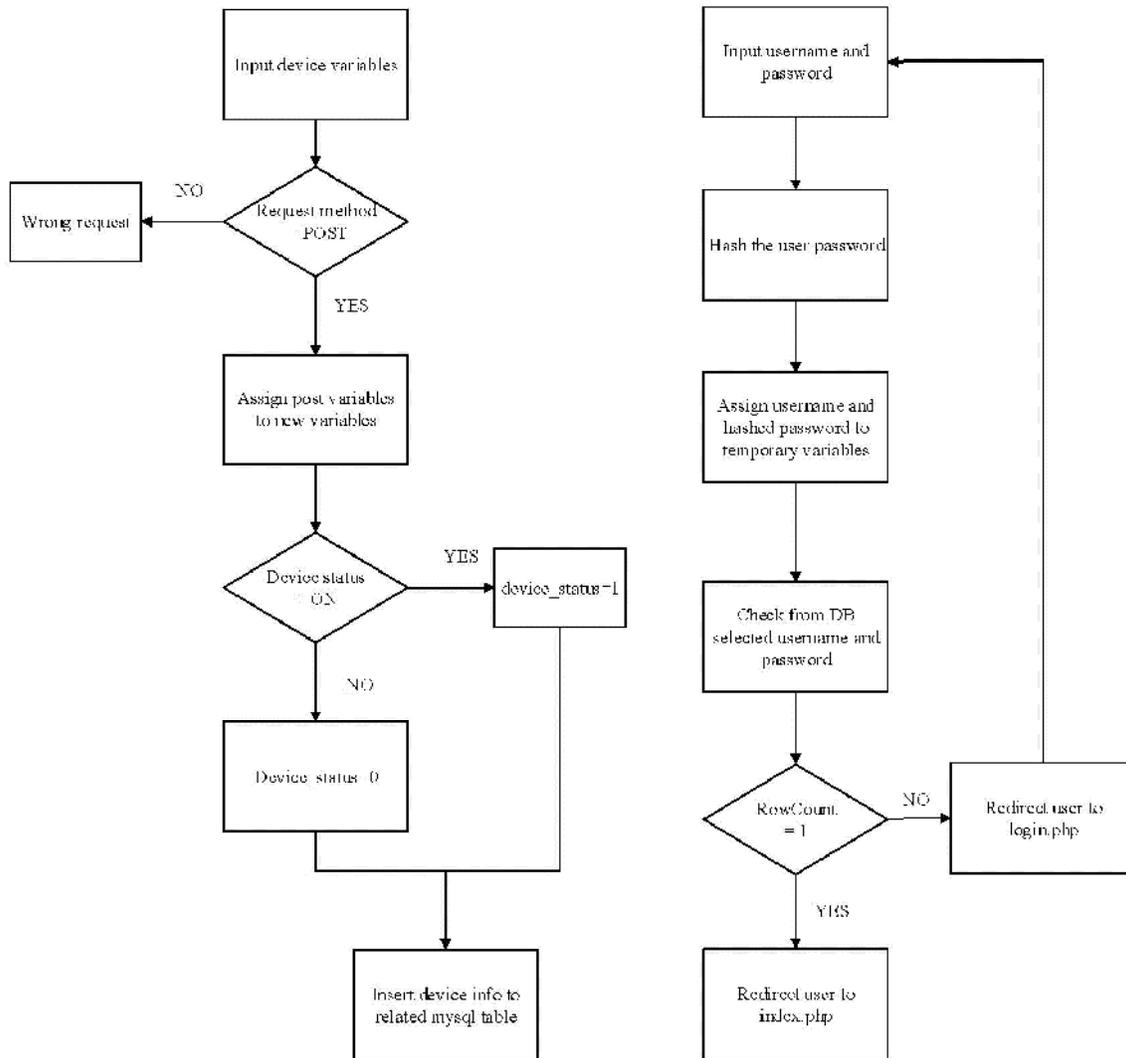


Figure 4. Flowchart of Add New Device. Figure 5. Flowchart of Login Check.

3.2.4. Development on Raspberry Pi

Firstly, it is necessary to change the boot process of Raspberry. Raspbian OS wants a username and a password from user to login. This phase removed, because it is not proper for end-user. So, changes are made in terminal by using some scripts which are developed by the project team. Raspberry Pi comes with a Midori web browser, but it does not have a capability of supporting HTML5's video element. So, that issue is solved with an open source project which called PiViD. PiViD is a browser extension which developed with JavaScript.

Step I: Repository addition provides; set up the repository as an additional package source on your Raspberry Pi and get the project public key. Finally, fetch the available packages from repository server.

Step II: Installing PiVid provides; Install PiVid and dependencies (other packages which PiVid requires to work).

Step III: Activating PiVid provides; Start the pivid service (it talks to your browser and to omxplayer to play your video)

Step IV: Installing PiViD in browser. To use the user script in Raspbian's default browser – Midori, it is necessary to activate the “User add-ons” extension. After that Midori can be started. Open the preferences dialog (Menu>Preferences), select the Extensions tab and tick “User add-ons”.

4. SOFTWARE ARCHITECTURE OF DIGITAL SIGNAGE SYSTEM

After these explanations and developments on raspberry Pi system, it is now possible to design suitable system architecture for successful and effective digital signage system. Dashboard design of digital signage system is given in Figure 6 and Uploaded media design of digital signage system is given in Figure 7. Dashboard allows users to see their devices, active slides and media library. Uploaded Media allows user to see their own media library at this page.

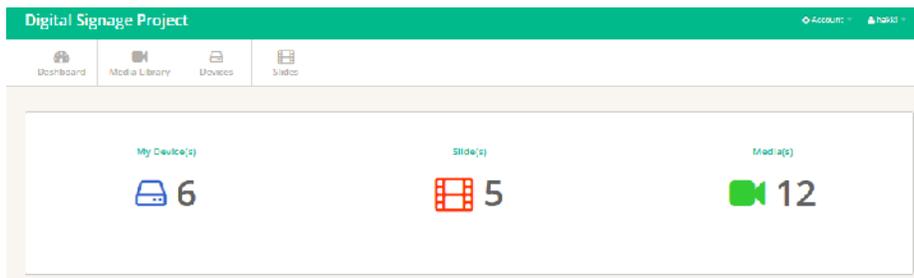


Figure 6. Dashboard Design.

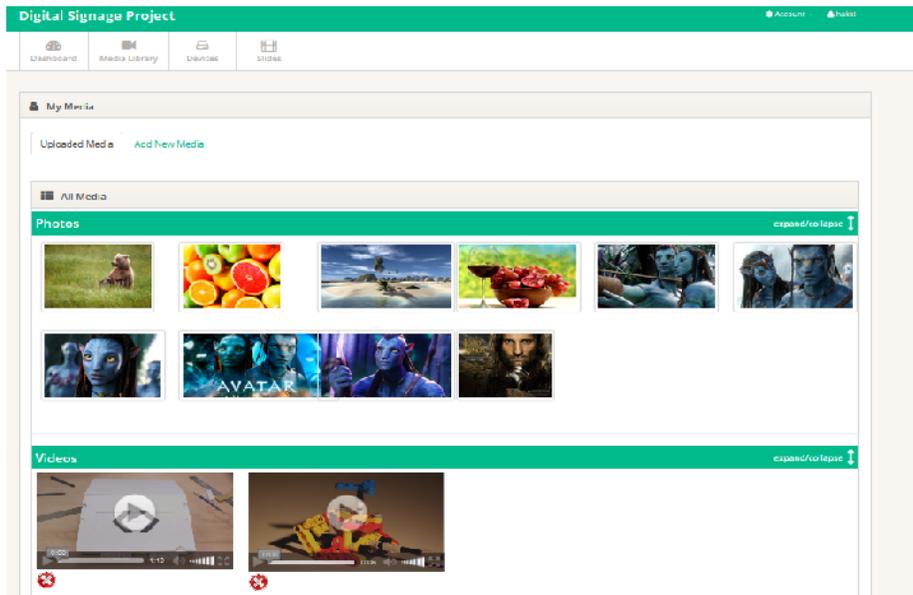


Figure 7. Uploaded Media.

End-to-end digital signage system integration structure is given in Figure 8. Digital Signage system integration consists of basic elements. There is a display connected to HDMI out of Raspberry pi, power cable and Ethernet also connected to related ports of Raspberry pi. So the Raspberry pi can reach to digital signage web server. The administrative side of the system accessible by any device which can be connected to Internet. It is simple to manage the digital signage system by our responsive design for laptops, smartphones and tablets.

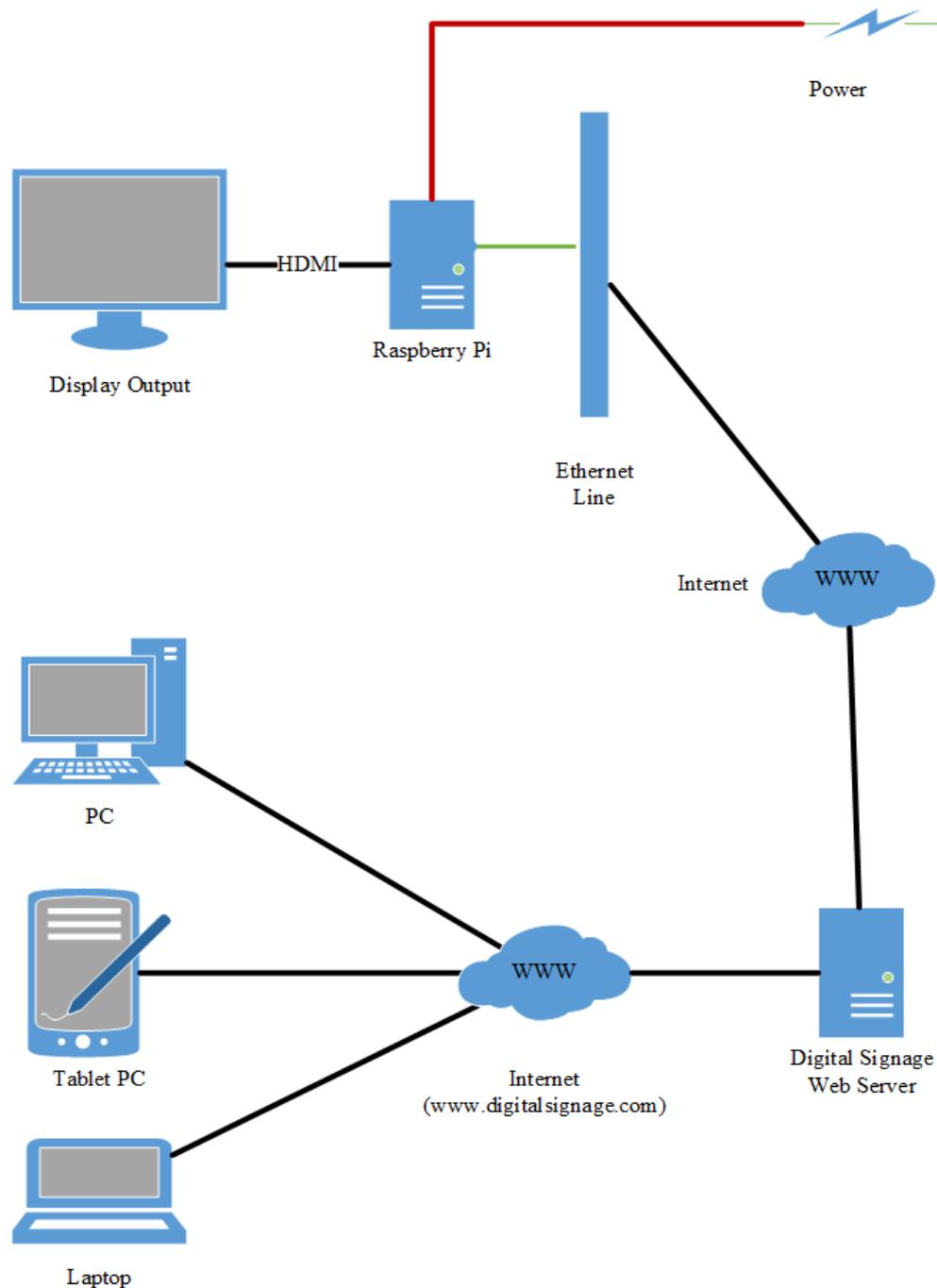


Figure 8. Digital Signage System Integration.

4.1. Entity – Relationship Diagrams

Entity – Relationship diagram of database design for digital signage system is given in Figure 9. The structure consists of five tables as User, Media, Device, Slide and Slide Order. User information is stored in the USER table, every user has their own media so media_owner attribute is stored in MEDIA table. System can access to devices through the owner_id attribute in the DEVICE table and active slide of the device is also stored in the Device table. Slides are stored in the SLIDE table with their slide owner attribute. Finally order of the slides is stored different table which is SLIDE ORDER.

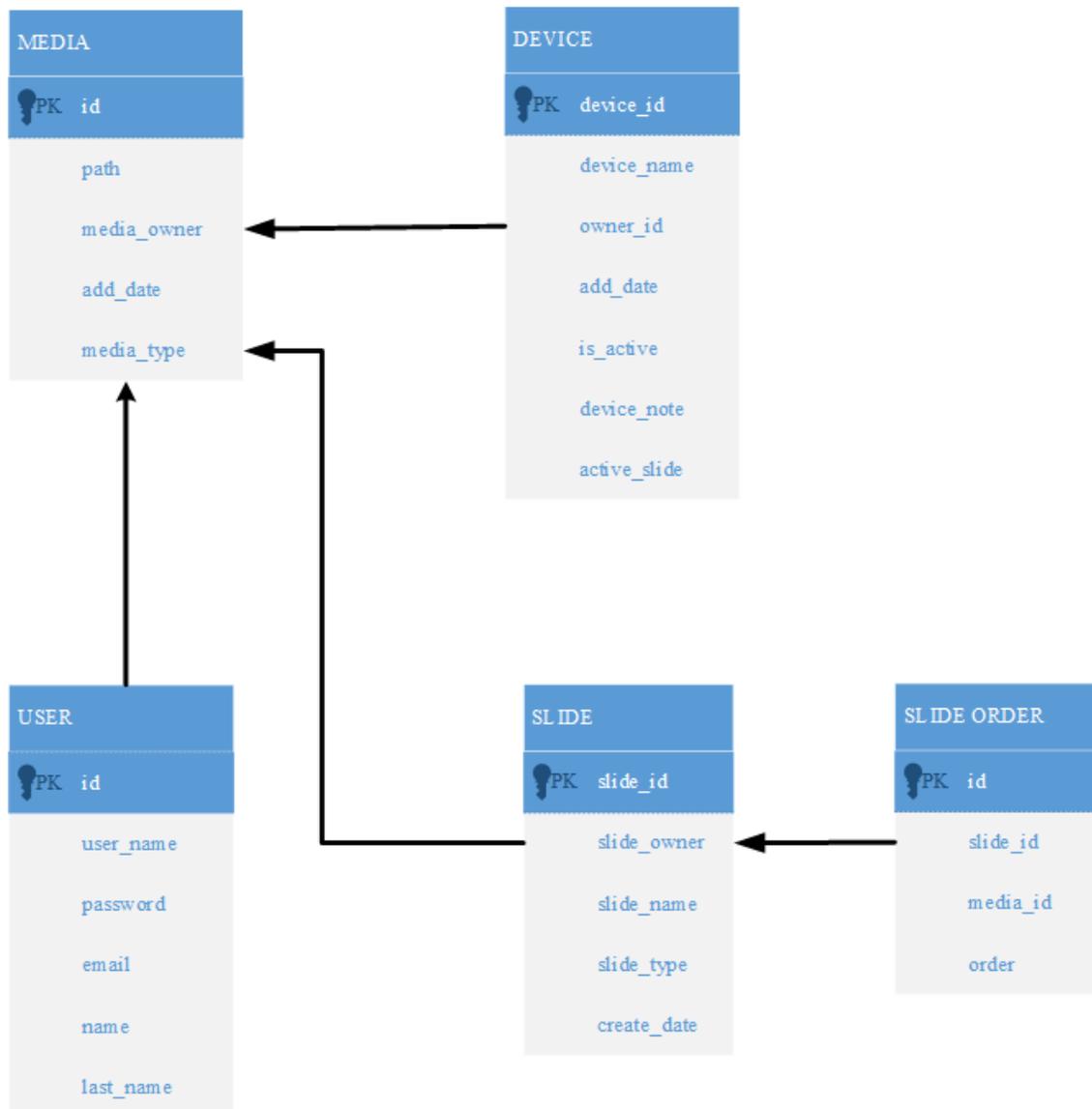


Figure 9. E-R Diagram.

4.2. Use Case Diagram

Use Case diagram of digital signage system is given in Figure 10. User can use seven different facilities of digital signage system. These are log in, register, Add/Delete Media, Add/Edit Device, Create Slide, Profile Edit and Dashboard Media, Device, User, Slide and Slide-order. User can add or delete photo or video files, and also add or edit device activation or select slide for related device. User can also create slides by selecting or arranging media or widgets.

Proposed Remotely Managed Embedded Digital Signage System can be usable at many different areas. It has variety of solutions for different sectors by simple modifications. Digital signage systems are mostly used at communal areas like public transportation. Local buses and bus stations, subways and ferries can use digital signage technology. By using this method; informing citizens about the next departure, daily news, weather, traffic conditions and exchange rates can be easier. There are also some alternatives such as indoor and outdoor advertisements, entertainment; cinema, theatre and exhibition centres, education, and corporate information. Digital signage solutions are preferred by tourism sector too. Daily activities, specific tours, historical information about the local area can be easily explained to the visitors.

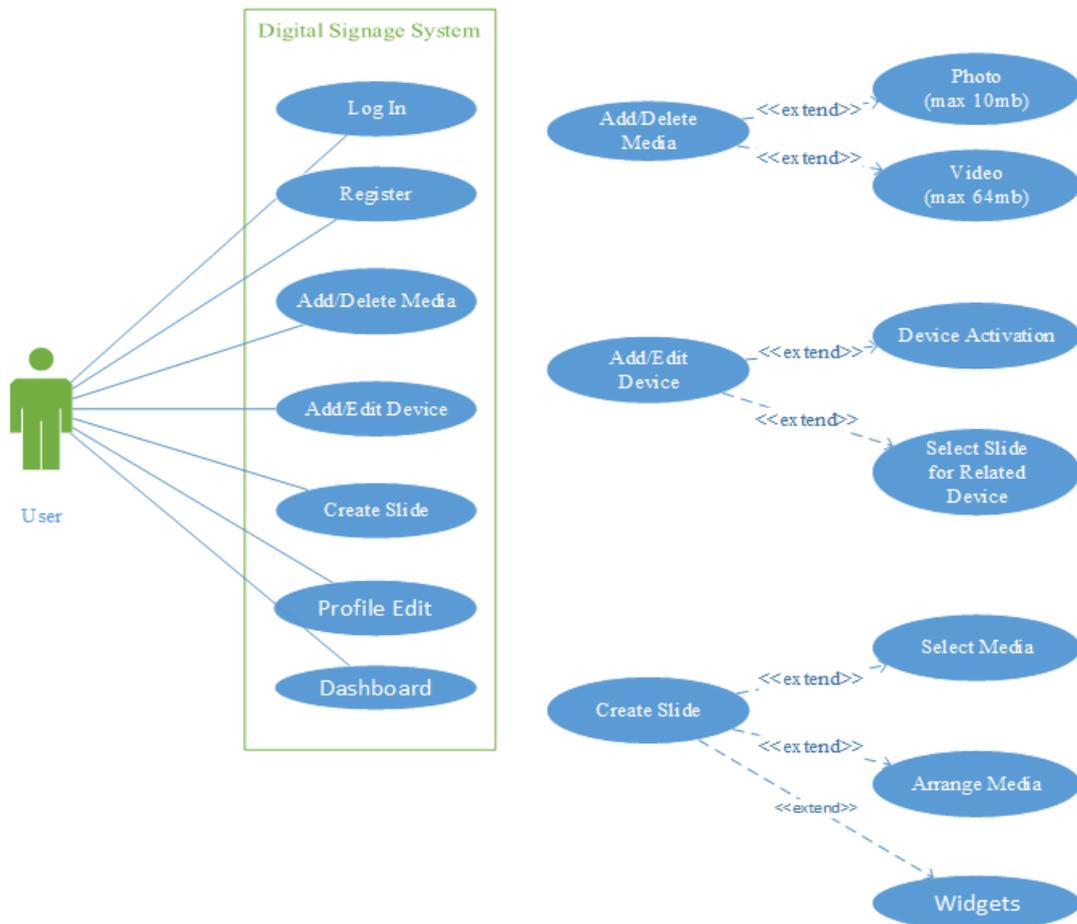


Figure 10. Use Case Diagram.

5. CONCLUSIONS

Current Digital Signage technologies are explained, relevant concepts and architectures are considered and specifications of digital signage systems are discussed.

A novel, effective, fast and cheap remotely managed embedded Digital Signage System architecture is designed, implemented, developed and presented in detail. This level of design detail does not commonly appear in the literature. This is a significant contribution especially for compact size embedded digital signage systems.

The strength of our work lies not only in developing a compact size embedded digital signage systems, but also in implementing and presenting this within the hardware and software engineering technology.

REFERENCES

- [1] C. Bauer, (2011) "Interactive Digital Signage – An Innovative Service and Its Future Strategies", Tirana, 2011 International Conference on Emerging Intelligent Data and Web Technologies (EIDWT), 7-9 September 2011, pp 137-142.
- [2] Jimmy Schaeffler, (2008) "Digital Signage", Focal Press Media Technology Professional, ISBN: 978-0-240-81041-6.
- [3] Kuo-cheng Yin, Hsin-chieh Wang & Don-lin Yang, (2012) "A Study on the Effectiveness of Digital Signage Advertisement", Hangzhou, 2012 International Symposium on Computer, Consumer and Control, pp 169-172.
- [4] Lars-Ingemar Lundstrom, (2008) "Digital Signage Broadcasting: Content Management and Distribution Techniques", Focal Press Media Technology Professional, ISBN: 978-0-240-80976-2.
- [5] Keith Kelsen, (2010) "Unleashing the Power of Digital Signage", Focal Press Media Technology Professional, ISBN: 978-0-240-81302-8.
- [6] Kun-Ming Yu, Cheng-Yan Yu, Bo-Han Yeh, Ching-Hsien Hsu & Hung-Nien Hsieh, (2010) "The Design and Implementation of a Mobile Location-Aware Digital Signage System", Hangzhou, 2010 Sixth International Conference on Mobile Ad-hoc and Sensor Networks, pp 235-238.
- [7] Roy Want & Bill N. Schilit, (May 2012) "Interactive Digital Signage", Computer, Vol. 45, No. 5, pp 21-24.
- [8] Yi-Ting Lee, Chia-Hung Lien, Hung, A., Jhe-Hong Ren & Ting-Kai Chang, (2012) "Design of a low cost interactive digital signage", 2012 IEEE 1st Global Conference on Consumer Electronics (GCCE), Tokyo, pp 120-124.
- [9] W. Jason Gilmore, (2010) "Beginning PHP and MySQL. From Novice to Pro", ISBN: 978-1-4302-3115-8.
- [10] Ben Laurie, (2002) "Apache: The Definitive Guide (3rd Edition)", ISBN: 978-0-596-00203-9.
- [11] Ryan Flores, (2014) "Getting Started with Bootstrap 3", ISBN: 978-1-310-97891-3.
- [12] Eric A. Meyer, (2006) "CSS: The Definitive Guide", ISBN: 978-0-596-52733-4.
- [13] Kyle Simpson, (2012) "JavaScript and HTML5 Now", ISBN: 978-1-449-33909-8.
- [14] David Sawyer McFarland, (2010) "JavaScript & jQuery: The Missing Manual", ISBN: 978-1-4493-9902-3.
- [15] Heiko Bck, (2011) "The Definitive Guide to NetBeans Platform 7", ISBN-13: 978-1-430-241010.
- [16] Maik Schmidt, (2012) "Raspberry Pi - A Quick Start Guide", ISBN: 978-1-93778-580-2.
- [17] Paul DuBois, (2013) "MySQL (5th Edition) (Developer's Library)", ISBN: 978-032-1833877.