University of Minho
School of Engineering
Electronics Engineering department
Embedded systems

Project: Report

Marketing Digital Outdoor with gesture interaction — Problem statement

Group 8

José Pires	A50178
Hugo Freitas	A88258

Supervised by:
Professor Tiago Gomes
Professor Ricardo Roriz
Professor Sérgio Pereira

November 2, 2021

Contents

List of Figures

List of Abbreviations

Notation	Description	Page List
API	Application Programming Interface	4, 13, 15
BN	Billions	2, 3
BSP	Board Support Package	13–15
CAGR	Compound Annual Growth Rate	3
CLI	Command Line Interface	13
COTS	Commercial off-the-shelf	5
CPS	Cyber—Physical Systems	1, 3
CV	Computer Vision	15
DB	Database	12–15
DOOH	Digital Out-Of-Home	2
GIF	Graphics Interchange Format	2, 7, 10, 15
HW	Hardware	5, 8, 11, 12
IP	Internet Protocol	14
MCU	Micro Controller Unit	11
MDO	Marketing Digital Outdoor	2, 8–11
MDO-L	MDO Local System	9–11
MDO-RC	MDO Remote Client	9, 11
MDO-RS	MDO Remote Server	9–11

Notation	Description	Page List
OS	Operating System	13–15
OSI	Open Systems Interconnection	12
PCB	Printed Circuit Board	5
R&D	Research and Development	3
RDBMS	Relational Database Management System	13, 14
SW	Software	5, 11–16
TCP/IP	Transmission Control Protocol/Internet Protocol	9, 10, 12
UI	User Interface	9, 10, 12, 13, 15

1. Introduction

The present work, within the scope of the Embedded Systems course, consists in the project of a Cyber—Physical Systems (CPS), i.e., a system that provides seamless integration between the cyber and physical worlds [1]. The Waterfall methodology is used for the project development, providing a systematic approach to problem solving and paving the way for project's success.

In this chapter are presented the project's context and motivation, the problem statement — clearly defining the problem, the market research — defining the product's market share and opportunities, the project goals, the project planning and the document outline.

1.1. Context and motivation

COVID pandemics presented a landmark on human interaction, greatly reducing the contact between people and surfaces. Thus, it is an imperative to provide people with contactless interfaces for everyday tasks. People redefined their purchasing behaviors, leading to a massive growth of the online shopping. However, some business sectors, like clothing or perfumes, cannot provide the same user experience when moving online. Therefore, one proposes to close that gap by providing a marketing digital outdoor for brands to advertise and gather customers with contactless interaction.

Scenting marketing is a great approach to draw people into stores. Olfactory sense is the fastest way to the brain, thus, providing an exceptional opportunity for marketing [2] — "75% of the emotions we generate on a daily basis are affected by smell. Next to sight, it is the most important sense we have" [3].

Combining that with additional stimuli, like sight and sound, can significantly boost the marketing outcome. Brands can buy advertisement space and time, selecting the videoclips to be displayed and the fragrance to be used at specific times, drawing the customers into their stores.

Marketing also leverages from better user experience, thus, user interaction is a must-have, providing the opportunity to interact with the customer. In this sense, when users approach the outdoor a gesture-based interface will be provided for a brand immersive experience, where the user can take pictures or create GIFs with brand specific image filters and share them through their social media, with the opportunity to gain several benefits.

1.2. Problem statement

The first step of the project is to clearly define the problem, taking into consideration the problem's context and motivation and exploiting the market opportunities.

The project consists of a Marketing Digital Outdoor (MDO) with sound and video display, and fragrance emission selected by the brands, providing a gesture-based interface for user interaction to create pictures and Graphics Interchange Format (GIF)s, brand-specific, and share them on social media. It is comprised of several modes:

- normal mode (advertisement mode): the MDO will provide sound, video and fragrance outputs.
- interaction mode: When a user approaches the device, the MDO will go into interaction mode, turning on and displaying the camera feed and waiting for recognizable gestures to provide additional functionalities, such as brand-specific image filters.
- multimedia mode: in this mode the facial recognition is applied, enabling the user to select and apply different brand-specific image filters and take pictures or create a GIF.
- sharing mode: after a user take a picture or create a GIF, it can share it across social media.

Brands can buy advertisement space and time, selecting the videoclips to be displayed and the fragrance to be used at specific times, drawing the customers into their stores. Customers can be captivated by the combination of sensorial stimuli, the gesture-based interaction, the immersive user experience provided by the brands — feeling they belong in a TV advertisement, and the opportunity to gain several benefits, e.g., discount coupons.

1.3. Market research

A Digital Outdoor is essentially a traditional outdoor advertising powered up by technology. The pros of a digital outdoor compared to a traditional one is mostly the way that it captivates the attention of consumers in a more dynamic way. It can also change its advertisement according to certain conditions, such as weather and/or time. Some researches tells that the British public sees over 1.1 Billions (BN) digital outdoor advertisements over a week [4], which can tell how much digital marketing is valued nowadays.

When talking about numbers, "At the end of 2020, despite the Covid wipeout, the Digital Out-Of-Home (DOOH) market was estimated to be worth $41.06 BN, but by 2026, nearly two out of three (65%) advertising executives predict this will rise to between $50 BN and $55 BN. A further 16% expect it to be worth between

$55 BN and $60 BN, and 14% estimate it will be even bigger" [5].

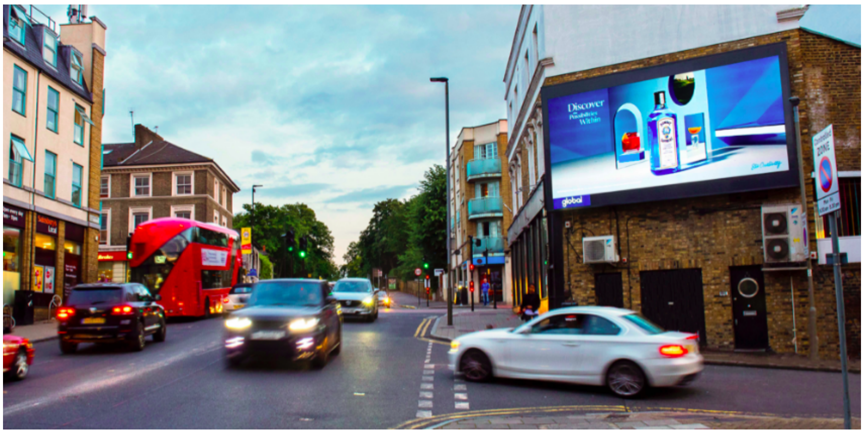

Figure 1.1.: Example of a Digital Outdoor, withdrawn from [4]

Scent market is the art of taking a company's brand identity, marketing messages, target audience and creating a scent that amplifies these values. That's because "a scent has the ability to influence behavior and trigger memories almost instantaneously. When smell is combined with other marketing cues, it can amplify a brand experience and establish a long lasting connection with consumers" [6].

Ambient scent uses fragrance to enhance the experience of consumers with different purposes, whereas scents in scent branding are unique to each company's identity. According to a Samsung study: "when consumers were exposed to a company scent, shopping time was increased by 26% and they visited three times more product categories" [7]. Also, "the digital scent technology market is expected to grow from $1.0 BN in 2021 to $1.5 BN by 2026, at a Compound Annual Growth Rate (CAGR) of 9.2%." [8].

The market growth can be attributed to several factors, such as expanding application and advancements in e-nose technologies, increasing use of e-nose devices for disease diagnostic applications, emerging Research and Development (R&D) activities to invent e-nose to sniff out COVID-19, and rising use of e-nose in food industry for quality assurance in production, storage, and display.

1.4. Project goals

The project aims to develop a CPS for multi-sensory marketing with contactless user interaction. The key goals identified and the respective path to attain them are:

1. devise a device with audio and video outputs, as well as fragrance diffusion: understand audio and video streaming and study fragrance nebulizer technologies.

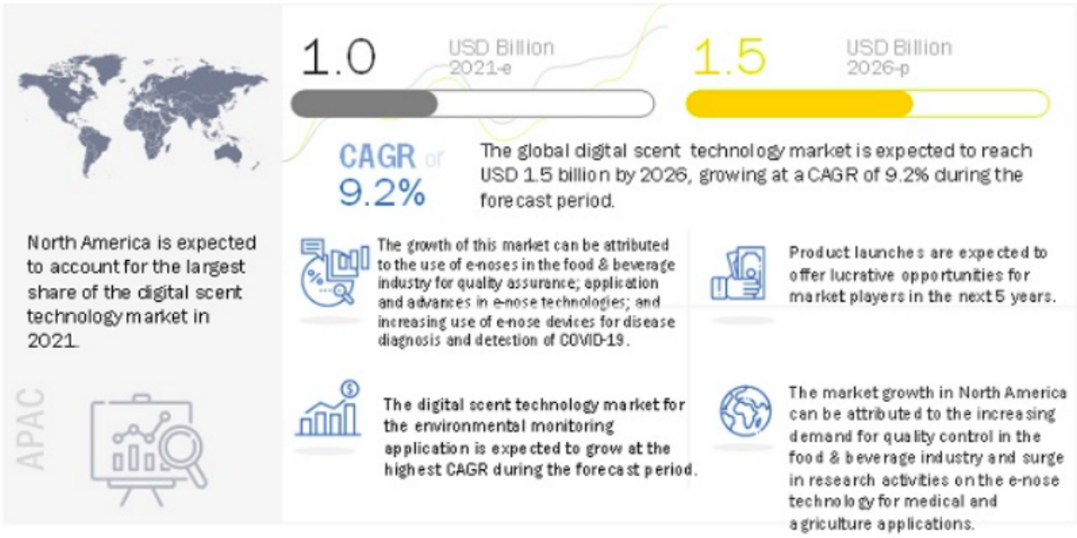

Figure 1.2.: Attractive Opportunities in the Digital Scent Technology Market, withdrawn from [8]

2. create a contactless user interface based on gestures through computer vision: identify user gestures through computer vision and match them to interface callbacks; a virtual keyboard may be required for user input.
3. devise a distributed architecture to convey brand advertisement information to the local device: understand distributed architectures and apply them for optimal data flow; create a remote client-server model to convey information from the brands to the device through remote cloud database services; devise adequate data frames to convey information to the local device; create a local server to respond to the remote server requests.
4. apply facial recognition to the camera feed and subsequently apply image filters specific to each brand: understand facial recognition algorithms and apply them to the camera feed; apply image filters on top of the identified faces through a specialized Application Programming Interface (API).
5. enable image and GIFs sharing to social media for increased brand awareness: understand how to use social media APIs for media sharing.

1.5. Project planning

In Appendix A is illustrated the Gantt chart for the project (Fig. A.1), containing the tasks' descriptions. It should be noted that the project follows the Waterfall project methodology, which is meant to be iterative.

The tasks are described as follows:

- Project planning: in the project planning, a brainstorming about conceivable devices takes place, whose viability is then assessed, resulting in the problem statement (Milestone 0). A market research is performed to assess the product's market space and opportunities. Finally, an initial version of the project planning is conceived to define a feasible timeline for the suggested tasks.
- Analysis: in this phase an overview of the system is conceived, presenting a global picture of the problem and a viable solution. The requirements and constraints are the elicited, defining the required features and environmental restrictions on the solution. The system architecture is then derived and subsequently decomposed into subsystems to ease the development, consisting of the events, use cases, dynamic operation of the system and the flow of events throughout the system. Finally, the theoretical foundations for the project development are presented.
- Design: at this stage the analysis specification is reviewed, and the HW and SW and the respective interfaces are fully specified. The HW specification yields the respective document, enabling the component selection, preferably Commercial off-the-shelf (COTS), and shipping. The SW specification is separately performed in the subsystems identified, yielding the SW specifications documentation (milestone).
- Implementation: product implementation which is done by modular integration. The HW is tested and the SW is implemented in the target platforms, yielding the SW source code as a deliverable (milestone). The designed HW circuits are then tested in breadboards for verification and the corresponding Printed Circuit Board (PCB) is designed, manufactured and assembled. After designing the PCB, the enclosure is designed to accomodate all HW components, manufactured and assembled. Lastly, the system configuration is performed, yielding prototype alpha of the product.
- Tests: modular tests and integrated tests are performed regarding the HW and SW components and a functional testing is conducted.
- Functional Verification/Validation: System verification is conducted to validate overall function. Regarding validation, it is conducted by an external agent, where a user should try to interact with the designed prototype.
- Documentation: throughout the project the several phases will be documented, comprising several milestones, namely: problem statement; analysis; design; implementation; and final.

1.6. Report Outline

This report is organized as follows:

- In Chapter 1 is presented the project's context and motivation, the problem statement, the market research, the project goals, and project planning.

- Lastly, the appendices (see Section 2.4.3) contain detailed information about project planning and development.

2. Analysis

In the analysis phase, the product requirements are derived — defining the client expectations for the product — as well as the project constraints — what the environments limits about the product. Based on the set of requirements and constraints, a system overview is produced, capturing the main features and interactions with the system, as well as its key components.

Finally, the theoretical foundations are outlined, providing the basic technical knowledge to undertake the project.

2.1. Requirements and Constraints

The development requirements are divided into functional and non-functional if they pertain to main functionality or secondary one, respectively. Additionally, the constraints of the project are classified as technical or non-technical.

2.1.1. Functional requirements

- Advertising through a screen and speakers;
- Have fragrance diffusion;
- Take pictures and GIFs;
- Detect a user in range of the device;
- Contactless user interaction through gesture recognition;
- Camera feed and facial recognition;
- Apply brand-specific image filters;
- Enable sharing multimedia across social media;
- Provide a remote user interface for brands to purchase and configure the advertisements;
- Provide a remote user interface for company staff to assess and control the MDO local system.

2.1.2. Non-functional requirements

- Low power consumption;
- Provide user-friendly interfaces;
- Have low latency between local system and remote server;
- Use wireless communication between the local and remote systems.

2.1.3. Technical constraints

- Use device drivers;
- Use Makefiles;
- Use C/C++;
- Use Raspberry Pi as the development board;
- Use compatible HW with the development board;
- Use buildroot;
- Social media APIs for sharing multimedia
- Image filtering through specialized APIs

2.1.4. Non-technical constraints

- Project duration: one semester (circa 20 weeks);
- Pair work flow;
- Limited budget.

2.2. System overview

The system overview presents a global view of the system, considering its main features, components and interactions. It is not intended to be complete, but rather provide a basis for the outline of the system architecture. Fig. 2.1 presents the MDO system overview.

Considering the system interactions, three main actors were identified:

1. Brand: represents the brands contracting the advertisement services;
2. Administrator: the development company staff, which can monitor and control the outdoor (administrative privileges).
3. User: the user (the target audience of the advertisiment) interacting with the system.

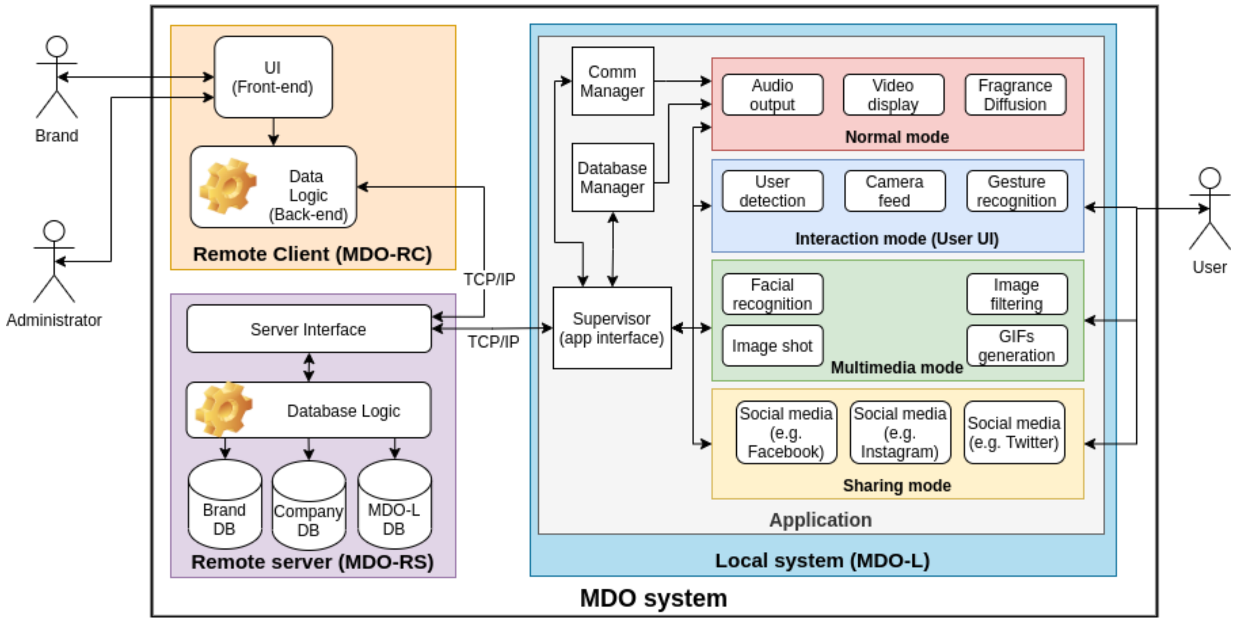

Figure 2.1.: MDO system overview

Considering the data flow across the **MDO system**, three main subsystems were identified: **MDO Remote Client (MDO-RC)**, **MDO Remote Server (MDO-RS)**, and **MDO Local System (MDO-L)**. The rational behind this initial decomposition is explained next.

2.2.1. MDO Remote Client

The Brand and Administrator members require a remote User Interface (UI) (front-end) to interact with the system: the former to configure the advertisements being displayed at the MDO and purchase them; the latter to remotely monitor and control the operation of the MDO. Thus, it is clear that an authentication mechanism must be provided for the remote UI.

The data is then dispatched to the back-end, where it is processed and feed back to the UI user and/or sent to the remote server, via Transmission Control Protocol/Internet Protocol (TCP/IP) comprising the data logic component of the UI.

2.2.2. MDO Remote Server

Although the MDO-RC could communicate directly with the MDO-L, this is not desirable or a good architecture mainly due to: communications failure could result in data loss, compromising the system's integrity; the remote client and the local system become tightly coupled, meaning the remote client must be aware of all the available local systems; if the data storage in the local system fails, the remote client would have to provide the backup information.

Thus, a remote server component is included, providing the access and management of the system databases, pertaining to the Brand, Company, and MDO Local system. The first two provide the historical information of the **Brand** and **Administrator** entities, and the last one the information related to all of the **MDO-L** systems in operation.

The main functions of the **MDO-RS** are:

- UI requests responses: when a UI user requests/modifies some information from the database, the server must provide/update it.
- MDO-L monitoring and control: provide command dispatch and feedback to the **Administrator** staff for remote monitoring and control of the device.
- MDO-L update: periodically check for start times of each MDO-L device and transfer the relevant data to it.

The server interface is the responsible for managing the requests and respective responses from the remote client and for periodically send the update data to all MDO-L devices.

2.2.3. MDO Local system

The MDO local system (MDO-L) is the marketing device, interacting with the user to display multi-sensory advertisements. As aforementioned in Section 1.2, it is comprised of four modes:

- normal mode: the MDO provides sound, video and fragrance outputs. It is the default mode.
- interaction mode: When a user approaches the device, the MDO will go into interaction mode, turning on and displaying the camera feed and waiting for recognizable gestures to provide additional functionalities, such as brand-specific image filters. This is the **User** UI.
- multimedia mode: in this mode the facial recognition is applied, enabling the user to select and apply different brand-specific image filters and take pictures or create a GIF.
- sharing mode: after a user take a picture or create a GIF, it can share it across social media.

The user interaction is considered to be a higher priority activity than the advertisements, so when a **User** interacts with the system, the **normal mode** is overriden by the **Interaction mode**, thus, halting the advertisements.

The MDO-L application communicates with the remote server (**MDO-RS**) through the **Supervisor** via TCP/IP to handle requests from **Administrator** members to monitor and control the device through the **Supervisor** or to update the advertisements. Additionally, the **Supervisor** oversees the application mode and the communication (**Comm Manager**) and database (**Database manager**) managers to handle system events.

2.3. System architecture

In this section, the system architecture is devised in the HW and SW components, using the system overview as a starting point.

2.3.1. Hardware architecture

The diagram in Fig.2.2 represents an initial hardware big picture in order to facilitate the objective identification. As it can bee seen, the diagram is divided in four distinguished parts: External Environment, Local System, Remote Server and Remote Client.

Firstly, the **External Environment** represents all the environment that interacts with the system. In this case, these are all its users - normal users, brands and staff.

Secondly, the **Local System** is composed for the main controller, which is the Raspberry Pi 4B. This Micro Controller Unit (MCU) is responsible to controll all the Local System and to establish connection with the remote server through its included WiFi module. The board is powered connecting it to the electrical network. Then, it has several blocks connected to it:

- Motion Detection: Used to detect the users and switch from normal mode to interaction mode;
- Fragrance Diffusion Actuator: used to diffuse the fragrance onto the air;
- Camera: Used to capture image that is then processed;
- Speakers: Used to produce advertisements sounds;
- Screen: Used to produce video clips of advertisements.

In third place, the **Remote Server** has a server node running in another machine that can be one computer or a main frame. The remote server establishes connection with the cloud that has stored all the data from all databases.

Lastly, the **Remote Client** which can be a computer, a tablet or a smart phone to run the MDO management application.

2.3.2. Software architecture

In this section the SW architecture for MDO-RC, MDO-RS, and MDO-L subsystems is presented, defining its SW stack.

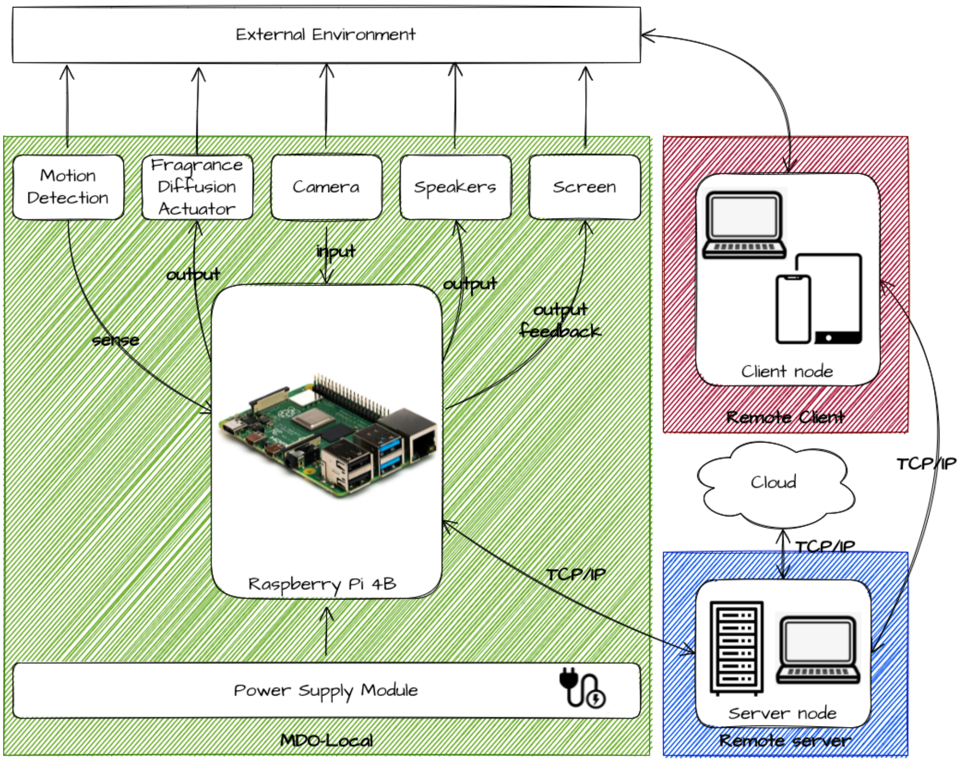

Figure 2.2.: HW architecture diagram

MDO remote client

Fig. 2.3 illustrates the SW architecture for the remote client, representing its SW stack. It is comprised of the following layers:

- Application: contains the remote client application. The **Brand** and **Admin** members interact with the UI, which is the visual part of the interface. The **UI engine** is notified and handles all UI events — internal or external — providing the **UI** with feedback for its users. The relevant commands are then parsed — **Parser** component — and responded. The commands are then translated to the appropriate Database (DB) queries and responded through the **DB Manager**. The **Comm Manager** is responsible for encapsulating the DB queries into the respective TCP/IP frames to be sent to the **Remote Server** as well as unwrap the incoming server responses.
- Middleware: contains the TCP/IP framework supporting these communication protocols as part of Open Systems Interconnection (OSI) model for internet applications. It manages the incoming/outgoing TCP/IP frames by providing the adequate protocol handshaking and queueing and timing aspects of the bytes to send/receive.

- OS & BSP — Operating System (OS) & Board Support Package (BSP): it contains the low-level and communication drivers required to handle input (keyboard/touch), output (screen) and communication to the **Remote Server**.

It should be noted that for desktop and mobile applications, the **Middleware** and **OS & BSP** layers are usually abstracted by the OS, thus, the relevant APIs should be used.

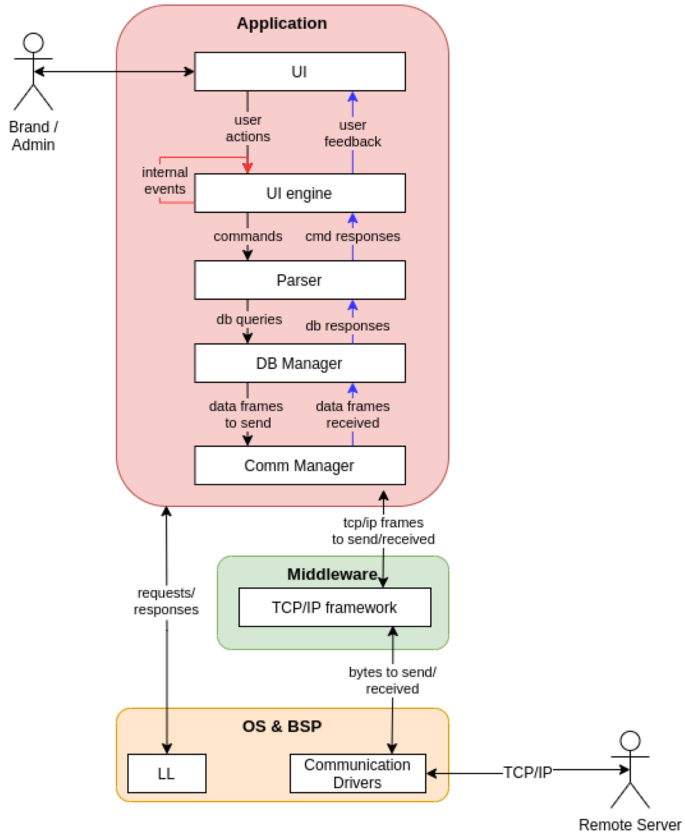

Figure 2.3.: SW architecture diagram: remote client

MDO remote server

Fig. 2.4 illustrates the SW stack for architecture for the remote server. It is comprised of the following layers:

- Application: contains the remote server application. It provides a Command Line Interface (CLI) to handle **Remote client** requests. The CLI engine is notified and handles all UI events — internal or external — providing the appropriate feedback. The relevant commands are then parsed — **Parser** component — and responded: DB queries are handled by the **Relational Database Management System (RDBMS)** issuing DB transactions; other commands received from the **Remote Client** are handled internally and translated, being dispatched to the **Local System** by the **Comm Manager** (via

Communication drivers). Internal events can also trigger the **RDBMS** to issue database transactions for the **Remote Client** or **Local System**. The **Comm Manager** is responsible for wrapping/unwrapping the data frames received by or sent to the **Remote Client** or **Local System**.

- Middleware: contains the RDBMS framework supporting the management of the relational databases using database transactions.
- OS & BSP — OS & BSP: it contains the **Communication** drivers to the handle requests from the **Remote Client**, and the **File I/O** drivers to manipulate DB transactions from/to storage.

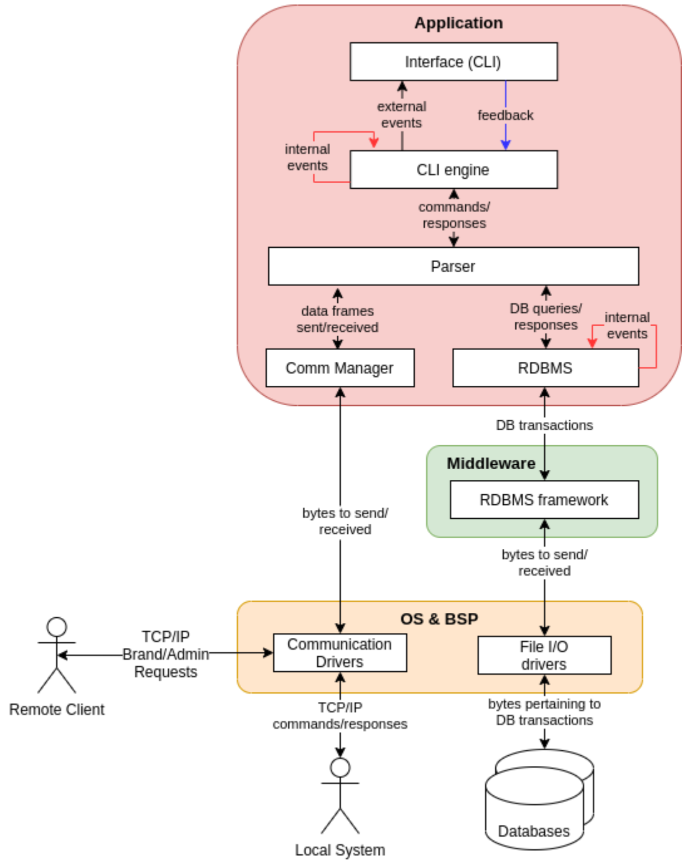

Figure 2.4.: SW architecture diagram: remote server

It should be noted that the **Remote Server** main functions are:

- provide relational databases for easier management of all entities and respective data in the system;
- decompose the relationship many-to-many, between the remote clients and local systems — many remote clients may want to connect to different local systems;
- decouple the architecture as the **Remote Client** should not know the Internet Protocol (IP) address of every local system it may potentially try to access, acting as a proxy server.

MDO local system

Fig. 2.5 illustrates the SW stack for architecture for the **Local System**. It is comprised of the following layers:

- Application: contains the local system application. It provides a UI to handle **User** interaction. The **Interface** engine is notified and handles all UI events — internal or external — through gesture recognition, providing the appropriate feedback. The relevant commands are then parsed — **Supervisor** component — and responded: DB queries are handled by the **Database manager** issuing DB transactions for internal databases; commands received from the **Remote Server** to monitor or control the system are handled internally and responded back by the **Comm Manager** (via **Communication drivers**); mode management is performed. Internal events can also trigger the **Database manager** to issue database transactions to update the **Local System**. The **Comm Manager** is responsible for wrapping/unwrapping the data frames received by or sent to the **Remote Server**.
- Middleware: contains: the DB framework supporting the management of the internal databases using database transactions; the Computer Vision (CV) framework that handles gesture and facial recognition; image filtering and GIF frameworks for multimedia; social media framework.
- OS & BSP — OS & BSP: contains: the **Communication** drivers to the handle requests from the **Remote Server** and for social media sharing, and, potentially the API calls to cloud-based image filtering frameworks, depending on the application profiling; the **File I/O** drivers to manipulate internal DB transactions from/to storage; audio, video and fragrance diffuser actuator drivers for normal mode; the camera driver for camera feed; the detection sensor driver to signal a **User** is in range, triggering the switch from normal mode to interaction mode.

2.4. Subsystem decomposition

For each subsystem, do:

1. User mockups
2. Events
3. Use cases
4. State machine diagram
5. Sequence diagram

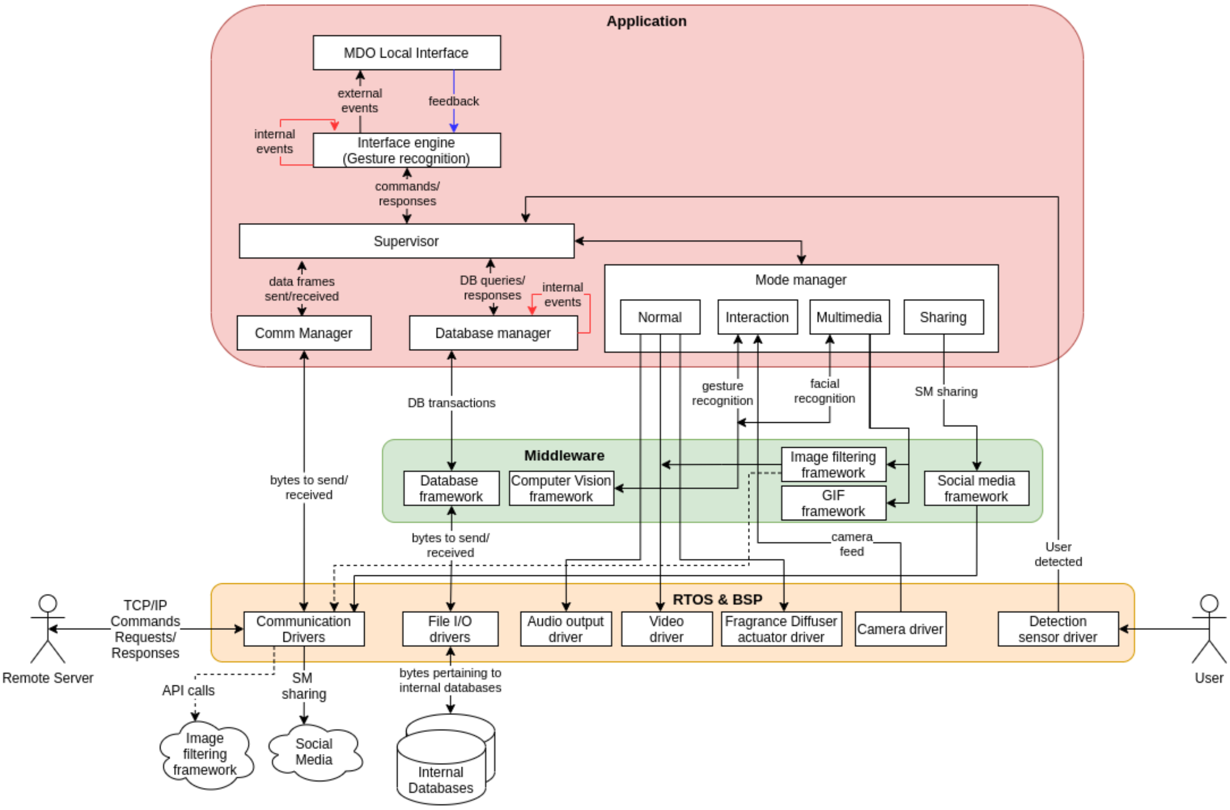

Figure 2.5.: SW architecture diagram: local system

2.4.1. Remote Client

User mockups

Events

Use cases

Dynamic operation

State machine diagram

Flow of events

Sequence diagram

2.4.2. Remote server

User mockups

Events

Use cases

Dynamic operation

State machine diagram

Flow of events

Sequence diagram

2.4.3. Local system

In this section the local system is analyzed, considering its events, use cases, dynamic operation and the flow of events.

User mockups

Fig. 2.6 illustrates the user mockups for the local system. It intends to mimic the user interaction with the local system, clarifying the user actions (gestures) and the respectives responses, as well as the workflow, comprising its four modes.

The initial state of the MDO-L's UI is depicted in thick border outline, after a **User** has been detected – **Interaction mode**. On the left it is the camera feed and on the right the commands ribbon, containing the hints to use the system and the available options. As it can been, the **User** can choose an option by hovering with pointing finger over the desired option for a designated amount of time (e.g., 3 seconds).

The workflow can be as follows:

- If the **User** selects the **Image filter** option, the **Image filtering** view is shown, presenting the options to select filters (which can be scrolled through palm raising/lowering), to cancel or accept the image filter. If a filter is selected **filter1_pressed**, it is applied, and if accepted it will return to **Interaction mode**, keeping the filter on.

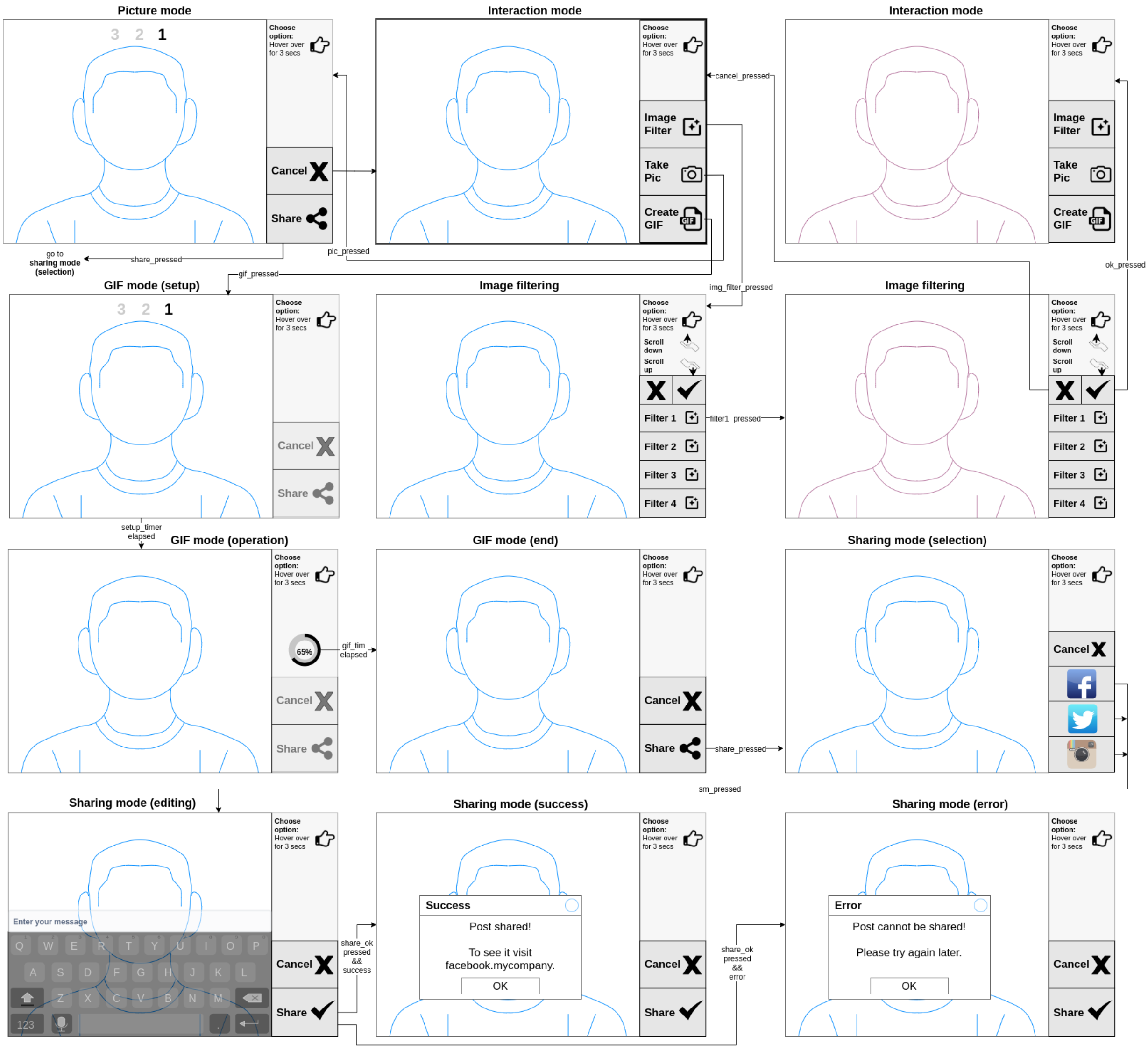

Figure 2.6.: User mockups: local system

- If the **User** selects the **Take Pic** option, **Picture mode** is started with a timer to allow the **User** to get ready before actually taking the picture. The **User** can **Cancel** — returning to main menu — or **Share** — starting **Sharing mode**.
- If the **User** selects the **Create GIF** option, **GIF mode (setup)** is started with a timer to allow the **User** to get ready before actually creating the GIF. After the **setup_timer** is elapsed, the **GIF mode (operation)** starts, displaying a dial with the GIF duration until being complete. When the **gif_timer** elapses, the GIF is created, enabling the **User** to **Cancel** — returning to main menu — or to **Share** — starting **Sharing mode**.

- Lastly, in the Sharing mode, the User can Cancel — returning to main menu — or select the social media network. After selecting the social media, the User can edit the post by entering its customized message and, if Share is pressed, a message box will appear displaying the status of the post sharing — Success or Error.

Events

Use cases

Dynamic operation

State machine diagram

Flow of events

Sequence diagram

Bibliography

[1] Fei Tao, Meng Zhang, and A.Y.C. Nee. Chapter 12 - digital twin, cyber–physical system, and internet of things. In Fei Tao, Meng Zhang, and A.Y.C. Nee, editors, Digital Twin Driven Smart Manufacturing, pages 243–256. Academic Press, 2019. ISBN 978-0-12-817630-6. doi: https://doi.org/10.1016/B978-0-12-817630-6.00012-6. URL https://www.sciencedirect.com/science/article/pii/B9780128176306000126.

[2] What the nose knows. URL https://news.harvard.edu/gazette/story/2020/02/how-scent-emotion-and-memory-are-intertwined-and-exploited/. accessed: 2021-10-23.

[3] Martin Lindstrom. Brand sense: How to build powerful brands through touch, taste, smell, sight and sound. Strategic Direction, 2006.

[4] Digital outdoor advertising: The what, the why, the how. URL https://bubbleoutdoor.com/digital-outdoor-advertising-what-why-how/. accessed: 2021-10-24.

[5] Digital outdoor market to be worth $55 BN in 5 years. URL https://www.decisionmarketing.co.uk/news/digital-outdoor-market-to-be-worth-55bn-in-5-years. accessed: 2021-10-24.

[6] Scent marketing; type of sensory marketing targeted at the olfactory sense, . URL https://www.air-aroma.com/scent-marketing/. accessed: 2021-10-24.

[7] Scent basics: What is scent marketing, scent branding and ambient scent, . URL https://reedpacificmedia.com/scent-basics-what-is-scent-marketing-scent-branding-and-ambient-scent/. accessed: 2021-10-24.

[8] Digital scent technology market with covid-19 impact analysis, . URL https://www.marketsandmarkets.com/Market-Reports/digital-scent-technology-market-118670062.html. accessed: 2021-10-24.

Appendices

A. Project Planning — Gantt diagram

In Fig. A.1 is illustrated the Gantt chart for the project, containing the tasks' descriptions.

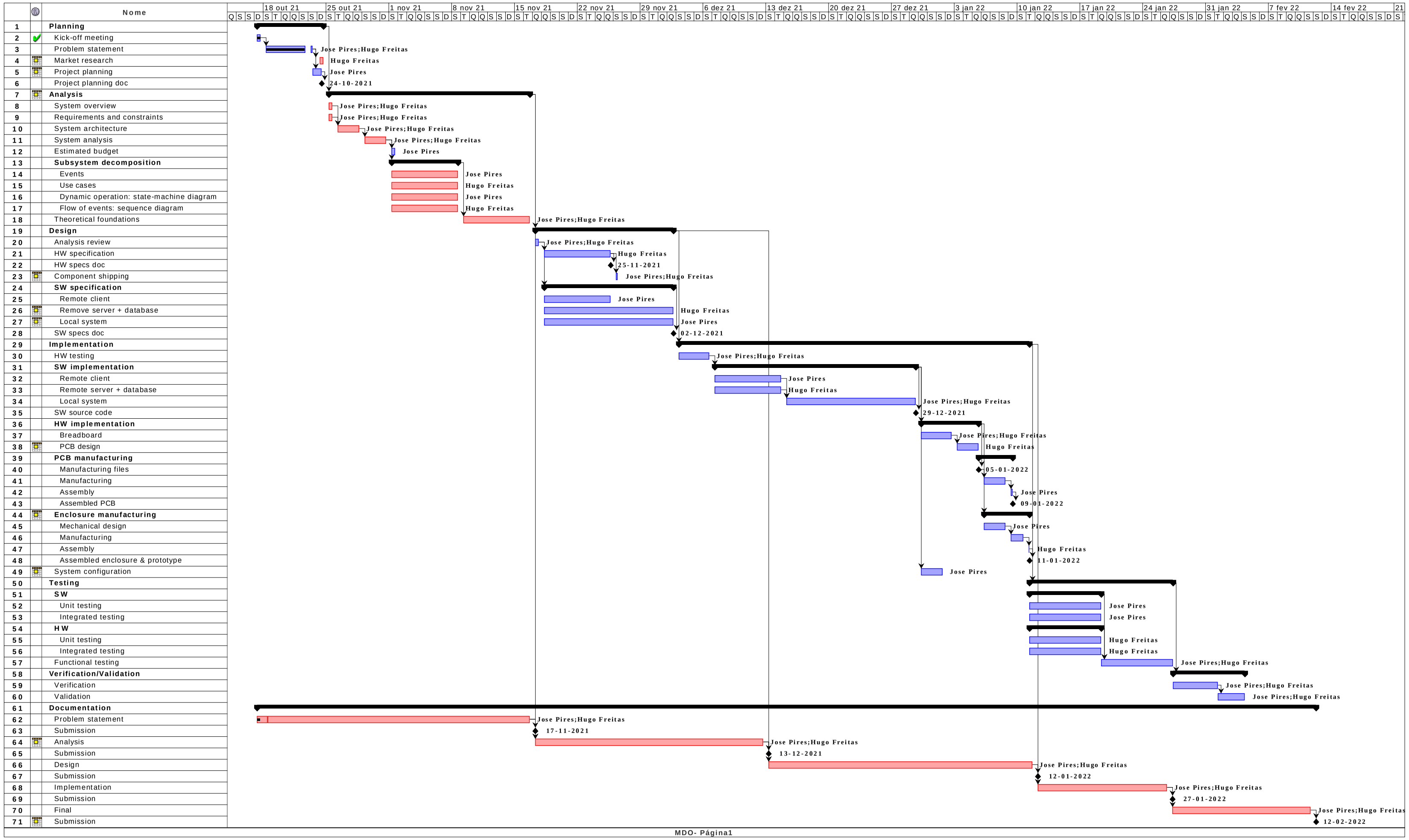

Figure A.1.: Project planning — Gantt diagram