POLITECNICO DI TORINO

Master's Degree in Aerospace engineering

Master's Degree Thesis

Development of Virtual Reality Pilot Training for A320 Standard Operating Procedures in Unity/Pacelab WEAVR

Supervisors:
Prof. Giorgio GUGLIERI
Dr. Stefano PRIMATESTA

Candidate:
Beniamino BARBERIS

TXT Group Consultants:
Ing. Angelo GINO
Ing. Stefano MAPELLI

July 2023

Summary

This thesis focuses on the application of Virtual Reality (VR) technology to develop a pilot training program for the Standard Operating Procedures (SOPs) of the Airbus A320 aircraft. The project is a collaboration between Politecnico di Torino and TXT e-solutions; this collaboration builds upon a history of several Master's degree theses conducted since 2020, where TXT/PACE (now one of its subsidiaries) provided Pacelab WEAVR, a software application add-on to Unity 3D created by PACE for the development of extended reality (XR) training applications at an industrial level. Thesis students of Politecnico di Torino used this asset to produce a procedural training experience for A320 pilots in an immersive virtual environment.
The aim of this thesis is to improve upon the previous works by implementing six different training procedures in a more detailed and correctly scaled A320 cockpit provided by TXT. The aforementioned procedures are **FMGS Preparation**, **Cockpit Preparation**, **Before Start**, **Engine Start - Automatic**, **After Start** and **Taxi**. Such procedures are either ones that were not previously implemented or procedures already developed for previous theses, which were reorganized in a more cohesive structure and updated to align with the early 2022 Airbus amendments to SOPs and the new cockpit.
This document begins with an overview of VR technologies, covering general hardware, market size, and the limitations of applications for these technologies. It is followed by a review of VR training applications in various fields, such as aviation and healthcare. Furthermore, the document presents the results of an overview of scientific literature on the evaluation of VR training effectiveness compared to traditional methods, which showed no standardization but nevertheless provided a series of useful information on study design and data collected for these types of studies, mainly about performance and factors of the training that may affect performance, as well as on statistical analysis of the results. Subsequently, the thesis describes the hardware and software apparatus used for its stated main goal, with a specific focus on Unity and Pacelab WEAVR. It then provides details on the virtual training environment, specifically the A320 cockpit and its equipment. Additional information is provided on the preparation of the virtual training environment

with Unity. Lastly, the document provides information on flight procedures and describes the procedures implemented for training within the virtual environment.

Acknowledgements

I would like to thank my supervisors, Professor Giorgio Guglieri, who gave me the opportunity to work on a thesis on this topic, and Doctor Stefano Primatesta, for his patience and constant support along the way of this project.
I would also like to thank my family, who supported me during these long years of study at Politecnico di Torino, and my friends, both outside and inside of this university. I would like to give special thanks to the student team in which I worked for four years, Team S55, because it is thanks to its awesome people both from older and newer generations that I was able to complete my academic obligations with my sanity (mostly) intact.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Virtual reality (VR) is a versatile technology widely used for multiple applications, including gaming, therapy, training and simulation, design and education [1]. Among these applications, training and simulation have caught the attention of numerous industries and services [2]: comparisons with traditional training show how the application of virtual reality in this context can be more immersive, resulting in improved training outcomes with enhanced knowledge retention and user satisfaction. Additionally, VR-based training can be more cost efficient and less risky for the trainees; moreover, there are specific training environments which are difficult or impossible to create without virtual reality. All of these factors contribute to make it particularly suitable for the aerospace industry.
TXT e-solutions[1] is an international specialised provider of engineering software solutions supporting its customers in high-tech markets, with a growing presence in Aerospace, Aviation, Defense, Industrial, Government and Fintech. In the Aerospace sector[2], TXT supports business and critical engineering processes throughout the aircraft life-cycle, from preliminary aircraft & system design to training and simulation; among training and simulation applications TXT developed training solutions in extended reality (XR), with Pacelab WEAVR serving as its flagship XR product. Through a collaboration that began in 2020, Pacelab WEAVR was provided to the Fligh Simulation Laboratory of Politecnico di Torino.
This thesis, like previous works [3], [4], [5], [6], [7] and [8] uses Pacelab WEAVR to create training procedures in virtual reality that contribute to the discourse on the application of virtual reality in aviation pilot training.

1. https://www.txtgroup.com/company-values/
2. https://www.txtgroup.com/it/our-markets/aerospace-defense/

1.1 Thesis objectives

This thesis main objective was the development of 6 different training procedures using Pacelab WEAVR in the virtual environment of A320 cockpit developed in Unity; these six procedures, namely **FMGS Preparation**, **Cockpit Preparation**, **Before Start**, **Engine Start - Automatic**, **After Start** and **Taxi**, were taken from the Flight Crew Operating Manual [9] of the A320, and are all procedures (or part of procedures, in case of **FMGS Preparation**) performed by the A320 pilots before take-off.
A secondary objective for this thesis was to conduct a research on the available scientific literature on the tools and common practices in testing the effectiveness of virtual reality training, in perspective of future thesis works that may implement the findings reported in this document.

1.2 Thesis outline

This thesis is organized in five chapters, as follows:

- Chapter 2, **Virtual Reality: Overview and Training Applications**, contains a brief overview of hardware and market size for virtual reality (section 2.1), then a review of different virtual reality training applications (section 2.2) and finally a discussion on how to test the virtual reality training effectiveness on its own and compared to different forms of training (section 2.3).

- Chapter 3, **Hardware & Software** contains a description of all the hardware (section 3.1) and the software (section 3.2) elements used for the project described in the following chapters;

- Chapter 4, **A320 Cockpit**, is used to describe the A320 cockpit (section 4.1) in its main elements, the A320 flight simulator of Politecnico di Torino (section 4.2) and the A320 cockpit in Unity (section 4.3);

- Chapter 5, **A320 Flight Operations Procedures & VR Training**, contains an introduction to the standard operating procedures in aviation and for the specific case of an A320, and in section 5.1 a description of the procedures implemented in Unity/Pacelab WEAVR in terms of user experience and from a development point of view.

- Chapter 6, **Conclusions**: this chapter contains a discussion on future improvements and applications specifically regarding testing the effectiveness of VR training compared to computer-based training (section 6.1) and future improvements to the virtual training created in Unity/Pacelab WEAVR for the standard operating procedures of an A320 (section 6.2).

Chapter 2

Virtual Reality: Overview and Training Applications

Virtual reality (VR) is part of extended Reality (XR) techonologies, which is a general term used to refer to a wide range of technologies that blend real and virtual worlds. From a practical viewpoint, it can be said that major XR technologies today can be broadly classified in terms of the different level of integration between virtual and real environments [1] (figure 2.1[2]):

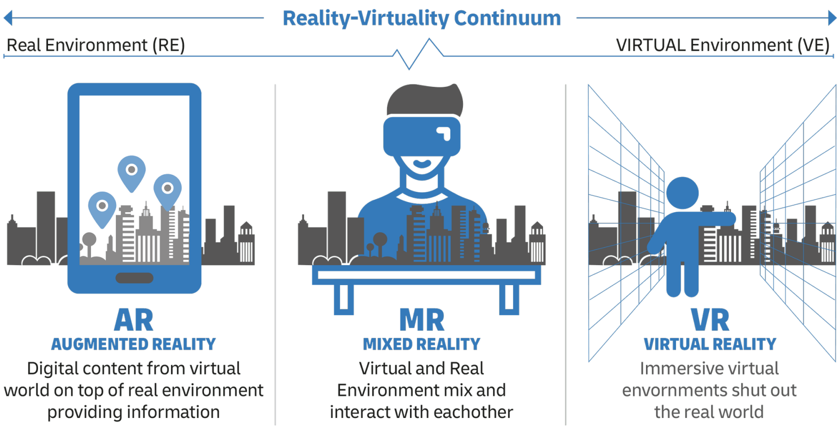


Figure 2.1: XR technologies: AR, MR and VR (left to right)

- Virtual Reality (VR) immerses users in a completely virtual environment;

1. https://pace.txtgroup.com/products/extended-reality/
2. https://www.dhl.com/global-en/home/insights-and-innovation/thought-leadership/trend-reports/augmented-and-extended-reality.html

- Augmented Reality (AR) projects virtual objects on top of the real world environment;
- Mixed Reality (MR) anchors virtual objects to real world environments; the virtual objects appear as a natural part of the real world, reacting to it and to users interactions.

This chapter focuses on providing an overview of VR technologies, on discussing a specific application of VR, which is training, and uses available literature to define an overview on common practices for evaluating the benefits of VR training per se or compared to computer based training (CBT) or training based on other technologies.

2.1 VR overview

VR technologies is a term that in literature is historically applied to a number of systems, with differences based on the level of sensory immersion. "Immersion" in this context is defined [10] as an objective measure of sensory engagement provided by VR systems related to the type and quality of hardware and software adopted, while "presence" is a related concept, referring to the subjective experience of one's sense of feeling of being part of the virtual world. Studies [11] show that presence and immersion are positively correlated (if one is high, the other tends to be high), and that together with an an high interactivity improve user satisfaction [12] and possibly learning, although studies on the matter proved to be inconclusive [13] or provided mixed results.

2.1.1 Hardware

One of the most marketable factors of VR today is its characteristic of being able to provide an immersive experience [14]**;** immersive VR (IVR) is based on first and foremost on head mounted displays (HMDs). These devices are used to exclude the real world view and substitute it with VR content: the most modern HDMs are equipped with LCD or OLED displays and special stereoscopic lenses that take a flat image from the displays and convert it into 3D visuals [15]**.** To guarantee an immersive experience and avoid cybersickness, the HMDs visuals must have:

- High refresh rate, meaning 90 Hz, 120 Hz or higher, with 90 Hz being today the most common figure for mid and high-end products and 120 Hz being both a figure for 2023 high-end devices [3] and a lower threshold recognized by

3. https://www.space.com/best-vr-headsets

studies to avoid simulator sickness and "to perform better in VR environments when visual accuracy and reaction speed are important" [16];

- Low latency time, with 20 ms being recognized as a treshold under which delays between head movement and changes in what is displayed by the HMD are imperceptible [4];
- Large horizontal field of view, to improve immersion, enhancing peripheral view and spatial awareness [17], with 90 to 130 degrees being used for mid to high-end devices today;
- High resolutions, to support an highly detailed visualization of both close and distant elements in the virtual scene.

Other important elements for a complete VR system [18] include tracking systems, which must translate the rotational movements (3DOF) or rotational and positional movements (6DOF) of the user in consistent changes in the scenery displayed with the HMDs, input devices to interact with the scene (which can also be part of the hand/body movement tracking systems) and, if HMDs are not stand-alone devices, desktop and mobile platforms that provide hardware and software elements to support VR experiences. An intuitive diagram on the relation between these components is provided in figure 2.2.

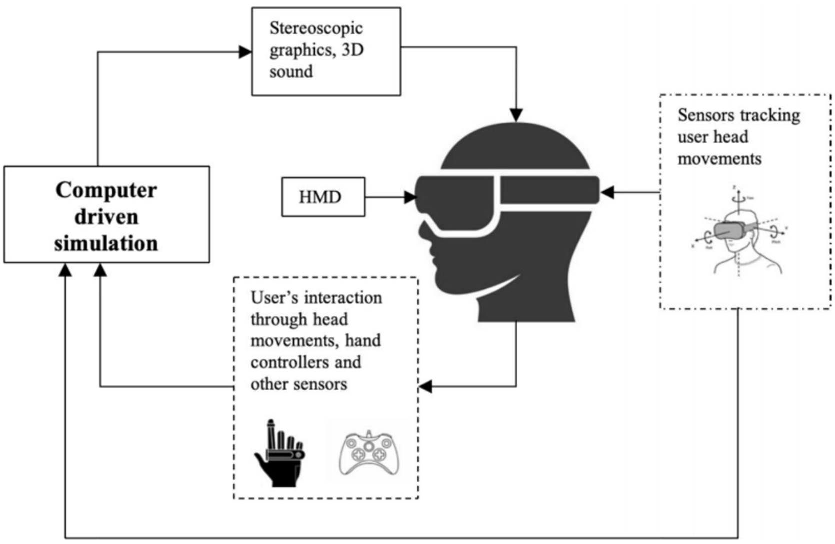


Figure 2.2: VR system diagram[2]

4. https://varjo.com/learning-hub/latency/

2.1.2 Market and limitations on adoption

The market for VR technologies is quite valuable and widespread across numerous applications and fields (figure 2.3); according to market analysis reports published by Grand View Research, the estimated value for VR market was USD 28.41 billion in 2022 [1], with USD 7.77 billion only associated with the HMDs market [19]. Projections from the same reports based on historical data from 2018 to 2021 show a compound annual growth rate (CAGR) from 2023 to 2030 of 13.3% for the VR market a a value of USD 87 billion.

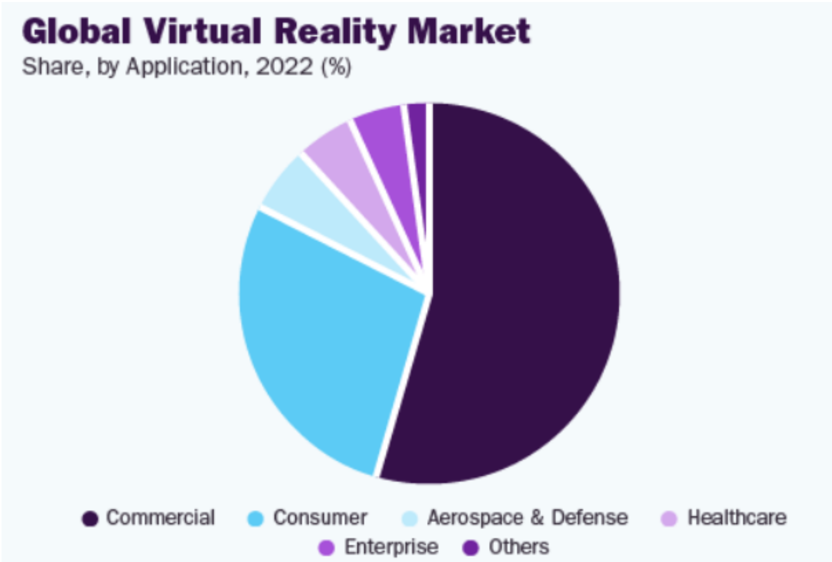


Figure 2.3: VR global market (share by application) [1]

According to research analysts [5] and indutry reports[6] [20][21] as of today VR technology must overcome some barriers to reach mass adoption, namely high prices of VR related hardware, lack of content, lack of consumer awareness, limited accessibility (meaning that VR hardware and software can be complex and unintuitive for some users leading to a steep learning curve), size and design of HMDs and general VR hardware (which can still pose problems such as discomfort related to extended use and portability), lack of standardization (which limits the ability of different VR devices and applications to work together seamlessly not allowing compatibility and interoperability), health issues (i.e. cybersickness, more

5. https://www.gartner.com/smarterwithgartner/3-reasons-why-vr-and-ar-are-slow-to-take-off

6. https://www.statista.com/statistics/1099109/barriers-to-mass-consumer-adoption-of-vr/

in section 2.3.2).

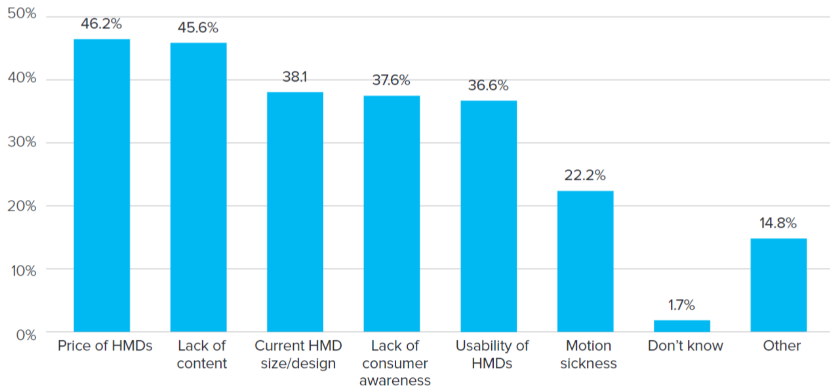


Figure 2.4: Barriers to VR technologies mass consumer adoption [21]

In figure 2.4 are provided the statistic for barriers to mass consumer adoption, as assessed by a 2019-2020 industry survey produced by VRX [21].
It is worthy of mention that the same survey asks to members of companies that deal in some capacity with XR technologies to provide reasons about what is stopping their company from further integrating XR technologies into their business; besides a cost factor, the two main reasons provided are a lack of awareness on benefits and options of these technologies and a lack of strong business cases that prove a return on investments (ROI). These results (figure 2.5) show how more studies or more successful business cases with more visibility on VR benefits and multiple applications are required to improve professional adoption of these technologies.

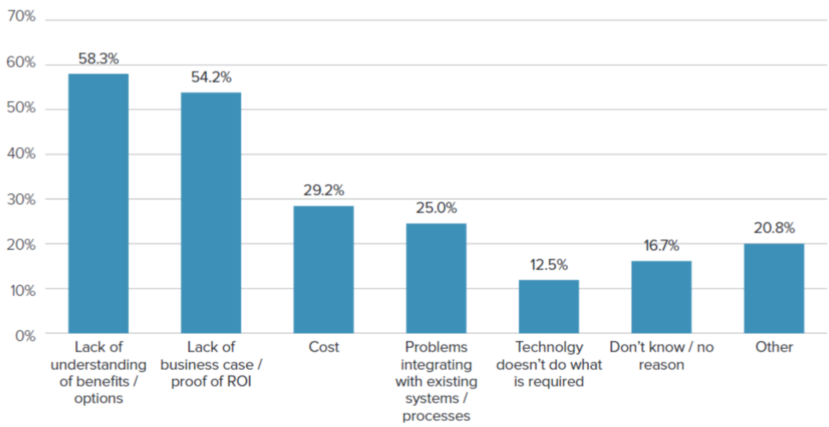


Figure 2.5: Barriers to XR technologies integration into industries business [21]

2.2 Training applications

VR technologies used as training tools can have multiple benefits compared to traditional training, computer-based or based on classroom lectures. Systematic reviews on the subject provided confirmation on the benefits of these tools [22] [2] or at least proved these tools as effective as traditional methods, although the diversity of applications and technologies used to assess the effectiveness of VR as a training tool, as well as the lack of standardization and possible publication bias on the matter are reasons for continuing research on the subject. In terms of specific benefits, multiple studies positively relate immersive VR applications with higher knowledge retention compared to computer based training [23] [24], higher user satisfaction [25] [12], and some even accelerated training time[7]. As general consideration, it can be said that VR can be more cost effective compared to traditional simulators and real world experience, limiting costs on the long run for expensive equipment and even instructors; the flexibility of VR applications in modifying the virtual environment dedicated for training mostly through software-based modifications compared to the acquisition of expensive hardware and the physical installation of new components in traditional simulators is a fundamental factor in terms of time and cost effectiveness of VR applications. VR allows practice for potentially dangerous tasks in hazardous environments without any real-world risks. It's particularly valuable in industries such as the aerospace industry, healthcare, and emergency services where mistakes can have severe consequences, and emergency situations cannot be practiced outside of dedicated simulators.

2.2.1 Aviation

Aviation is an industry where the benefits of VR training are evidently applicable, first and foremost in the flight simulation industry. Traditional flight simulators that replicate realistically the cockpit of an aircraft can have a price ranging from tens of thousand of dollars to millions of dollars in case of FAA-certified Flight Training Simulator Devices (FTSDs)[8]; these simulators usually require great spaces, which makes them very difficult to move, and replicate specific aircraft models of families, limiting their versatility. Moreover, for these simulators the visuals outside the cockpit can be simulated only with one or multiple screens onto which the outside world is projected, which cannot properly display the correct depth of

7. https://news.erau.edu/headlines/improved-pilot-training-program-yields-promising-results

8. https://www.aviationtoday.com/2019/08/01/training-brain-mind/

field (figure 2.6[9]), and possibly do not cover a 360 degrees FOV. Lack of depth of field can be critical especially in flight phases like the landing or take-off phases for a fixed-wing aircraft, where the pilot must be able to correctly evaluate the distances related to runways, or for a helicopter when the pilot must hover the vehicle at a distance close to terrain; combined with a limited FOV it can be said that these factor contribute to limit spatial and situational awareness.

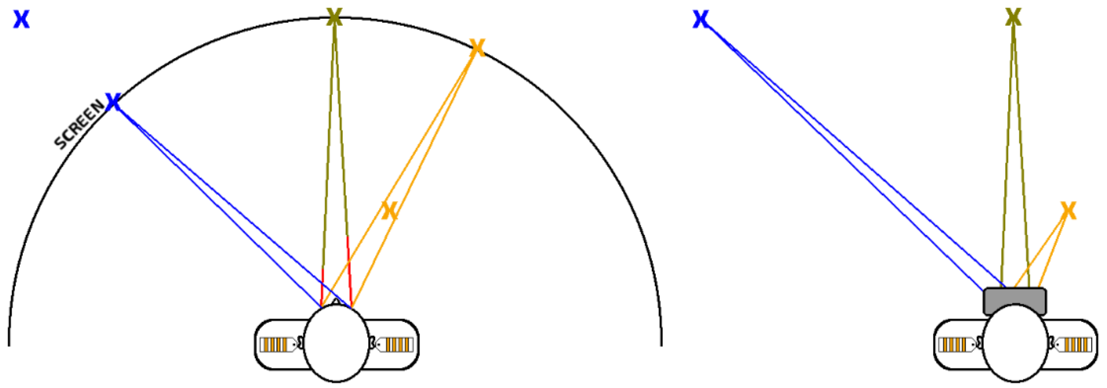


Figure 2.6: Depth of field between screen-based and VR simulator (left to right)

VR simulators have benefits in all the aforementioned factors; costs can be limited to a few thousands of dollars, considering the hardware (computer, HMD, controllers and joysticks, pedals) and software needed for the simulation. Dimensions and portability of a VR simulator are better than traditional simulators, as is the versatility factor, because different aircrafts can be simulated within the same flight simulator with software-based modifications and limited or none physical modification. The lack of depth of field and limited FOV problems are addressed: with HMDs in the virtual environment pilots can look around in 360 degrees, and with the optical system of displays and stereoscopic lenses on HMDs the images provided to the eyes guarantee a correct sense on depth. It must also be said that there are advantages to traditional high-end/certified flight simulators compared to VR systems, especially in terms of physical fidelity and realism of the cockpit in terms of the interactions with all the buttons, levers and switches, and in terms of full-body physical feedback provided by motion platforms, which can be valuable in simulating particular flight conditions like turbulence, or flight phases like takeoffs and landings, or other dynamic flight maneuvers.

Multiple organizations decided to took advantage of the benefits of VR; airlines and flight academies all over the world are right now using VR training not only

9. http://vrpilot.aero/virtual-reality-flight-training/

for pilots, but also to cabin crew and ground personnel, as shown with a partial list in table 2.1. Multiple major aircraft manufacturers and engine manufacturers such as Airbus, Boeing, Rolls Royce[10] and Pratt & Whitney[11] have developed or are developing training programs using VR.

Organizations	Description
KLM Cityhopper	Pilots training (Embraer 175 and 190)
American Airlines	Flight attendants training
Lufthansa	Flight attendants training
Qatar Airlines	Flight attendants training
All Nippon Airways	Flight attendants and maintenance training
Icelandair	Flight crew training (B737 MAX)
Alaska Airlines	Pilots training (B737-800SFP)
Baltic Aviation Academy	Pilots training (A320, B737 NG, B737 MAX)
Embry-Riddle Aeronautical University	Pilots training (Cessna 172)
Deutsche Post DHL Group	Pilots training (A300)
Philippine Airlines	Maintenance and cabin crew training (A320, A321)
Aviomar flight academy	Pilots training (B737 NG)
LifeFlight Training Academy	Pilots training (AW139, Bell 412)

Table 2.1: Airlines and flight academies adopting VR training

While talking about applications of VR training in civil aviation, the regulatory side of the matter cannot be overlooked; EASA has in recent years moved some step forward in recognizing the legitimacy of VR training for pilots, granting in 2021 the first certificate for a Virtual Reality (VR) based Flight Simulation Training Device (FSTD) developed by Varjo and Loftydynamics[12], and in 2022 certifying

10. https://www.vrowl.io/the-possibilities-5-examples-of-vr-training-in-aviation/

11. https://www.halldale.com/articles/11225-pratt-whitney-investing-in-virtual-reality-training-tools?v=preview

12. https://www.easa.europa.eu/en/newsroom-and-events/press-releases/easa-approves-first-virtual-reality-vr-based-flight-simulation/

the VR based Airbus H125 simulator as an FTD Level 3 simulator[13]. FAA on the other hand has not yet certified any VR training device.
Military organizations such as the US Air Force also started integrating VR in their training's structure; in 2018 with the "Pilots Training Next"[14] (PNT) program it was possible to instruct a 12-month class of 30 pilots with astonishing results: around 40% of the class was able to complete their training in one third of the time, with equipment far less expensive (USD 1000 per unit) than the traditional simulator (multi-million dollars system); other programs based on PNT were developed, "Undergraduate Pilot Training 2.5" and "Accelerated Path to Wings", with the purpose of graduating pilots faster[15]. Other disclosed applications were found, for example a VR system to train the flight crew of a AC-130U[16] for cockpit familiarization and checklist training (to avoid the use of traditional expensive full-mission simulators in early or simpler stages of training), and a maintenance program at Sheppard Air Force Base[17] which assessed comparable scores between VR and traditional training (through lessons, reading manuals and finally hands-on experience on an aircraft) but allowed students to complete a 27-day program in less than half the time.

2.2.2 Healthcare

While there are many applications areas for VR in healthcare, the two main areas can be easily identified in therapy and training. In terms of therapeutic applications, VR is commonly used for pain management [**26**], providing cognitive distractions as an alternative to drugs, for mental illness treatments [**27**] being used for exposure therapy, as an integration for physical therapy [**28**] and to treat visual impairments like strabism or amblyopia [**29**]. On the training side, there are multiple applications dedicated to surgical training; the benefits brought by VR training are a realistic, immersive and interactive learning environment where trainees can study the anatomy of the human body, taking advantage of the sense of depth provided by HMD technology to better understand the spatial relationship between different organs and bones, or they can practice more or less

13. https://www.airmedandrescue.com/latest/news/helitrans-acquires-new-h125-vr-simulator

14. https://taskandpurpose.com/news/air-force-vr-pilot-training/

15. https://www.military.com/daily-news/2021/03/26/air-forces-virtual-reality-fighter-training-working-best-5th-gen-pilots.html

16. https://bisimulations.com/customer-showcase/vr-based-gunship-crew-trainerl

17. https://www.texasmonthly.com/news-politics/air-force-virtual-reality-training/

complicated procedures without risks involving real patients, using special input devices that provide an haptic feedback to simulate the physical interactions of the procedures. These training session have an high level of repeatability because they do not rely on disposable medical equipment, patients or cadavers; this is important because multiple training sessions are needed to improve muscle memory, hand-eye coordination and procedural fluency, and can be more cost-effective than the alternatives mentioned.
There are multiple studies that present encouraging results for VR training programs. A study conducted at David Geffen School of Medicine at UCLA [30] in collaboration with OssoVR evaluated VR training in the context of teaching the surgical technique for tibial shaft fracture intramedullary nailing: results showed that the students belonging to the VR training group performed better than the traditionally trained group in terms of time needed (20% faster) and performance on completed tasks (38% more steps of the procedure completed correctly). Another study [31] in collaboration with PrecisionOS compared the results of traditional and VR training for a group of senior surgical residents in a training related to orthopedic surgical skills, specifically about performing a reverse shoulder arthroplasty for rotator cuff tear arthropathy; the VR training group completed practice in one fourth of the time needed for the control group to finish it, and the application of the training showed significantly better performance metrics for the VR group (around 70%), while also showing high transfer effectiveness ratio and a good transfer of training, and also that VR was 34 times more cost-effective than traditional training. In a similar application (for knee arthroplasty) another study [32] found the application of VR training beneficial in terms of number of steps of a procedure performed correctly (the control group performed about 20% worse) and time needed to complete a procedure (the VR group was about 25 % faster). While the examples provided may be limited, they show benefits confirmed by systematic reviews on a larger sample of studies found in literature on the subject [33][34][35]. The three specific examples indicated used medical students and doctors in training (residents), but applications of VR training can be found for all the medical personnel, for paramedics [36] and nurses [37][38], psychiatrists and psycologists [39].

2.2.3 Emergency services

Personnel belonging to emergency services (e.g. firefighters,law enforcement and emergencies medical services, or EMS) are frequently subject to injuries while providing their services; a study from 2009 [40] shows how police officers and firefighter must deal with injury rates two to three times greater than the U.S. labor force. To improve the probability of success of a mission and to mitigate the risk of injuries, first responders must go through intensive training, which traditionally

comes in the form of more or less long and costly programs. While the majority of training sessions continue to adhere to the conventional approach of conveying knowledge through lessons with the support of books and slide presentations [**41**], for all first responders a great deal of importance is given to practical training.
First responders traditional practical training consists in real-life scenario based training, recreating scenarios that may occur while on duty, doing so in dedicated training facilities, preparing the environment in the most realistic way possible using gear that replicates what they would have at their disposal in a real-life situation. This approach has limits, related to costs for the resources needed for this type of training, like instructors, gear, and specific resources to set up a scene like fuel, water and wood for firefighters; the time needed to prepare a realistic simulation environment is also a limiting factor, as well as the lack of possible flexibility in the design of the training scenario.
VR training addresses potentially all the issues listed before for the practical training of emergency operators, limiting or eliminating the need for real-life gear and real training environments, which cuts the costs and simplify the logistics of the training, and providing a flexible design for the simulations with the possibility of simulation in a virtual environment which engage the operators in a hazardous scenario.
VR training for EMS can replicate the usual work environment for these operators, such as ambulances, and several studies have demonstrated the benefits of VR in associated applications. For instance, [**42**] described a VR training system designed to enhance the ability to configure the interior of an ambulance with the appropriate equipment. The study compared this VR training system with a physical simulation environment, and the results indicated a statistically significant difference in completion time (67% faster) in favor of VR training. Additionally, the study showed that the VR system was significantly more cost-effective, with a setup cost of USD 15,000 compared to USD 250,000 required for a physical environment equipped with all the necessary equipment. Another similar study [**41**] described the development and testing or a VR training application for operators of ambulance busses showing improvements compared to traditional training in terms of better accuracy in completing tasks (49 %) and shorter time on task (30%).
Police agencies all around the world (e.g. US[18], Mexico[19], China[20], Australia [21])

18. https://www.apexofficer.com/resources/police-departments-are-implementing-vr-training-technologies

19. https://www.euronews.com/next/2023/02/22/mexico-city-police-fight-crime-in-virtual-reality-in-futuristic-new-training-centre

20. http://www.chinadaily.com.cn/a/201804/17/WS5ad599a9a3105cdcf6518d04.html

21. https://www.police.wa.gov.au/About-Us/News/Virtual-Reality-Training

are using VR to train their personnel for a variety of situations, from de-escalation scenarios to active-shooters scenarios. There are several studies that support the adoption of this technology; for example a police officers real-life scenario based training was compared to VR training in terms of cognitive and physical response in a study [43] performed on 237 police officers of the Dutch National Police to a VR training in terms of cognitive and physical response. The study found that VR training elicited perceived stress, mental effort, and average heart rate in a similar manner to real-life training, but that physical activity was on average higher for the latter form of training; the study suggested that VR training may complement real-life training in more "static" scenarios (e.g. detection of suspects in unknown environments) while real-life training may be applied to scenarios where high-intensity physical activity is involved, for example when chasing a suspect. Another study [44] applied to 80 UK police officers which used VR training and real-life training showed no significant differences in learning outcomes between the two and a mixed form of training using both.

The value of firefighters VR training is also starting to be recognized internationally: a 2022 article on the National Fire Protection Association (NFPA) journal[22] (a no-profit international association with more than 50000 members worldwide which publishes codes and standards intended to minimize the possibility and effects of fire and other risks) recognized the role of FLAIM[23] and similar companies that develop VR training solutions for firefighters as an important player in the future of this profession, for the advantages that these training programs provide in terms of a safer training and a more environmentally friendly training, since live training facilities are great consumers of water and great carbon emitters, and materials used in live training can be contaminants, like some PFAS-containing firefighting foams. The NFPA article also mentions that it has yet to add guidance on this type of tools in its codes and standards, because it needs more data to properly assess VR technologies and associated immersive learning; a US research program[24] funded by the Federal Emergency Management Agency and the Department of Homeland Security is referenced as a step in the right direction to address the lack of data currently available. There is, however, some recent research on the matter, for instance [45], a research that studied the performance of firefighters trainees in a VR system capable of providing both visual and tactile information through the use of a special thermal haptic device, or also [46], which tested the usability of

22. https://www.nfpa.org/News-and-Research/Publications-and-media/NFPA-Journal/2022/Spring-2022/Features/VR

23. https://flaimsystems.com/

24. https://www.nfpa.org/News-and-Research/Publications-and-media/Blogs-Landing-Page/NFPA-Today/Blog-Posts/2022/01/03/Research

the FLAIM VR training systems for 91 brazilian firefighters with positive results.

2.2.4 Space industry

Space industry has a long history in using VR technology, and multiple space agencies all over the world have used it to some extent. NASA has used it at least since 1986 in an application related to telerobotics [47], and has used for training astronauts at least since the nineties [48], when immersive virtual environments were used for the training of extra vehicular activities (EVA) for the ground-support flight team for the Hubble Space Telescope mission completed in 1993. This VR application offered a unique and irreplaceable training experience by providing a fully immersive visual perspective of the entire vehicle-arm-telescope worksite, something that no other training environment could offer at the time; soon after what is known today as the Virtual Reality Laboratory (VRL) at the Johnson Space Center was formalized, an immersive training facility which had the initial goal of developing, maintaining and supporting astronaut training for the Hubble Telescope repair missions, and developed in the course of the years several VR training solutions [49], such as a VR training system for the Simplified Aid For EVA Rescue (SAFER), which is a "jetpack" backpack created to serve as a backup system in case an astronaut becomes untethered from the International Space Station (ISS) during a spacewalk; this system serves the purpose of training and certifying astronauts in the utilization of SAFER while on the ground, and at least up until 2020, it remained the sole VR training system accessible aboard the International Space Station (ISS). Another training system developed by VRL is the Mass Handling Training System using the McDonnell Douglas Charlotte IVA robot, modified for micro-gravity mass handling training with payloads above 500 pounds; one session of training usually consists in handling different payloads and install them into specific structural interfaces of the virtual environment. Astronauts complete at least one training session before going to the ISS.
Not only the VRL, but also other NASA training facilities have adopted VR training: the NASA Extreme Environment Mission Operation (NEEMO)[25], which is an underwater facility with the goal of simulating the internal environment of the ISS. NEEMO adopted VR to train the astronauts inside the facility in performing a number of tasks, from routine to emergency tasks like treating severe health conditions; this training had some astonishingly good results, as astronauts took one hour to complete the given tasks after the VR training, as opposed to the four hours needed after a traditional training based on studying written instructions and manuals.

25. https://skywell.software/blog/virtual-reality-for-space-exploration-astronaut-training/

There are applications of VR training related to space agencies other than NASA, for example Titan Lake[26] developed by Raytracer, a VR training application funded by the Australian Space Agency which aims to use the underwater environment of a swimming pool to make the astronauts in training experience the weightless environment of space.
The European Space Agency (ESA)[27] recently decided to fund seven projects related to XR technologies, among which there is a study for similar VR training compared to Titan Lake. Regarding activities on the ISS, ESA, in cooperation with the german DLR and the french CNES, published a paper [**50**] in which described the development of an on-board VR training (VR-OBT) system for the astronauts of the ISS for maintenance activities and remove & replace for the ESA Life Support Rack payload.
There is an interest for in the subject also outside of space agencies; VR training for astronauts is also being developed for the Boeing Starliner program[28], where astronauts will be trained in VR for each phase of the entire Starliner's mission, including launching, docking, re-entering the atmosphere and landing phases, using Varjo's VR devices.

26. https://www.industry.gov.au/news/virtual-reality-training-space-exploration

27. https://www.esa.int/ESA_Multimedia/Images/2022/06/Underwater_VR_for_astronaut_training

28. https://varjo.com/boeing-starliner/

2.3 VR training effectiveness evaluation

A number of studies can be found in literature for the purpose of showing the effectiveness of VR training against other means of training, like traditional computer-based or based on AR technologies, with not many studies available to the public for the specific application of VR training in aviation.
Most of the studies in literature that evaluate the effectiveness of VR training compared to traditional or alternative AR training methods have some points in common, or worthy of mention, as discussed in the following sections.

2.3.1 Study design

To evaluate the effectiveness of VR training, a systematic review [**22**] based on a large database of scientific articles (317) found as a common practice (43 % of the studies evaluated) to implement mixed-design studies, meaning that comparisons on the results of the training were done between two groups, usually called experimental (with VR training) and control (adopting computer-based or similar traditional training), and also that comparisons were made between the results of the same people but in different points in time. The same review reported that while pre-test and post-test measurements (before and after a training method is experienced by the two groups) were fairly common (respectively 57% and 93%), a measurement in a third point in time to assess knowledge retention was not so common (18%); this can be easily explained in terms of logistics, but nevertheless a performance test in a third point in time (days or weeks or months after the VR training) appears to be valuable, and should be performed when possible. The pre-test can be administered as an item of a personal information form if the knowledge assessment to be executed as post-test cannot be performed before a training session. A systematic review, dedicated to applications of VR in industrial skills training [**51**]**,** associated four main categories of measures to be performed during or after the VR training for the corresponding group, namely immersion/presence, cybersickness, usability and task load. These are all factors that contribute in some way to the overall VR experience, and a measurement of all of them can provide an useful interpretation tool for the results of the VR training group in the performance test. [**22**] showed a sample size of groups used for the evaluation of VR training effectiveness ranging from four participants to more than two thousands; it also showed how a few studies performed power analysis [29] to determine the sample size needed for an experiment, given a number of constraints; this was performed

29. https://www.tibco.com/reference-center/what-is-power-analysis

using software tools like G*Power [52] or through the IBM software SPSS. For the statistical analysis on the results of the performance test, it is considered good practice to avoid too little sample sizes for each group (smaller than 6) and to avoid extremely unbalanced sample sizes between the two groups that are being compared[30].

2.3.2 Data collected

1.3.2.1 Personal information

It was found to be common practice to make the candidates compile a survey to know them in terms of age, gender, occupation, prior knowledge of the technology/systems used in the experiment and on the subject of the training. These data can be used in combination with the performance test results for a regression analysis, to understand which factors can be correlated to more or less positive results.

1.3.2.2 Performance

Although it can be said that performance evaluation indices are to some extent tailored on the application, there are some elements in common between different references found in literature. The scoring system to evaluate the results of the training sessions (VR and traditional) often uses measures of performance indexes like total time to conclude the tasks previously trained [53][54][55], number of movements (referred to a reference) to establish the efficiency of a candidate [54], number of errors (solved and unsolved) associated with steps of the procedure[55], number of external interferences needed to complete the procedure [56] (if the participant got stuck on a particular step of it), or also number of steps performed with varying degrees of correctness [5][57][38]; all this data can be used as scoring parameters on its own or can be used to create a more complex scoring system [56].

1.3.2.3 System usability

According to ISO 9241-11, usability "is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use". Effectiveness refers to achieving the desired goals using the system, efficiency refers to the resources needed to achieve the desired goal, and satisfaction on how the user reacts to the system. Following this definition, the assessment of usability for any system should consider the

30. https://libguides.library.kent.edu/SPSS/IndependentTTest

specific context in which it is employed and determine its suitability within that context [58].
It can be useful to measure usability for several reasons. Usability is related to user satisfaction, because a system with high usability should be easier and more intuitive to use, which would mean an improved user experience; a better user experience can also give a competitive advantage to a system compared to its less-usable competitors. A better usability and user experience can also mean that a system enables users to complete tasks faster and more accurately; a measure of usability can identify areas where users are more likely to make errors and use these results to improve the system's design.
According to [51] and [22], the "System Usability Scale" (SUS) [58] is the most commonly adopted questionnaire for measuring VR system usability. The SUS is a 10-item, 5-point Likert scale questionnaire that is used to produce a global assessment of system usability, combining the results of each of the ten questions in a scoring system 0-100. An alternative found in literature by [51] is the "Questionnaire for the subjective consequences of intuitive use" (QUESI) [59].

1.3.2.4 Task Load

Measuring the task load for a training system can be important for a number of reasons; for example, task load is positively related to the level of challenge and provided by the training system, which should be adequately scaled to avoid that users experience either boredom or stress and anxiety. A high task load may also be an indicator of a need for training optimization, with actions such as providing clarifications or additional support in order to simplify the training and reducing the task load.
Multiple literature reviews ([22] and [51]) referenced the NASA task load index (NASA-TLX) [60] as a common tool used as a subjective way to measure workload during training. NASA-TLX is a 6-item questionnaire to measure how demanding a training system is by different point of views: mental load, physical load, how much pressure was put on the user in terms of time limit and performance, and what the user experienced in terms of frustration and needed effort to complete the task[31]. As an alternative, [51] references the simulation task load index (SIM-TLX) questionnaire [61], which was developed building upon the NASA-TLX as an adaptation specifically suitable for VR simulators. The 9-item questionnaire measures mental load, physical load, how much pressure was put on the user in terms of time limit, what the user experienced in terms of frustration, complexity, stress, perceptual strain, distraction and difficulty in terms of navigation and

31. https://humansystems.arc.nasa.gov/groups/TLX/

control in the virtual environment[32].

1.3.2.5 Presence

Presence measurements in literature are commonly performed in the form of self-report questionnaires administered to users after their VR experiences. While this method is certainly the easiest to apply, It must be noted that post-experience questionnaires may be inherently not reliable particularly for measuring presence, because of the time discrepancy between the compilation of the questionnaire and the VR experience, and because of the fact that they are not sensitive to state changes during the experience [62]. Alternatives that attempt to address these issues are found in literature [62] in different forms. For example, physiological measurements such as electrodermal activity, heart rate, and brain activity can be performed while the user is experiencing a virtual environment. These measurements are challenging to set up, which explains why they are uncommon methods for measuring presence [63] compared to questionnaires. A different approach worthy of mention to presence measurement is provided in [64], where commonly used presence questionnaires were provided to an experimental group inside the virtual environment, making the questionnaire itself a part of the VR experience. This resulted in no significant difference in comparison with the mean scores of a control group answering the same questionnaires outside of VR. However, the in-VR scores had significantly lower variance, therefore demonstrating greater consistency in the results among the experimental group.
A compendium which discusses applications and limitations of different types of questionnaires applied in this context is provided by [65]. Notable mentions are the "Presence Questionnaire" (PQ) and the "Immersive Tendencies Questionnaire" (ITQ) [66]; PQ is a 32-item list of 7-point Likert scale questions with the aim of measuring presence in a virtual environment, associating each question with factors and sub-scales to identify different elements of the experience that contribute to presence, while ITQ is a 29-item questionnaire which should measure differences in the tendencies of individuals to experience presence; ITQ can be used to interpret the results of PQ questionnaires. Igroup Presence Questionnaire (IPQ) [67] is another 14-item list of questions with three sub-scales specifically dedicated to spatial presence, involvement and experienced realism [33]. Another appreciated questionnaire is the Slater, Usoh and Steed (SUS) questionnaire [68] in the form of a 6-item list of questions on 7-point Likert scale.

32. https://osf.io/p6de4/
33. http://www.igroup.org/pq/ipq/index.php

1.3.2.6 Cybersickness

Cybersickness is commonly referred to as a visually induced motion sickness associated with a series of disorientation symptoms (e.g. nausea, dizziness, eye strain, general disorientation) which can affect VR users; the identified causes for this sickness include factors like discrepancies between expected sensory signals and experienced signals, displays characteristics and gameplay experience [63]; cybersickness, if present, can affect negatively the level of presence in the VR environment. according to [63] and [66].
Measurements for cybersickness are commonly carried on with multi-item questionnaires, like the Simulator Sickness Questionnaire [69], which is a 16-item list of symptoms to evaluate on a four-point scale (0 for none, 3 for severe) used to create a total discomfort score and how this discomfort is divided in three sub-classes (nausea, oculomotor disturbance, disorientation) [70]. SSQ is not specifically designed for VR experiences, because simulator sickness and cybersickness have a slightly different symptomatology; this was a factor that contributed to the development of other questionnaires, like the VR Sickness Questionnaire (VRSQ) [71], which is a derivation of SSQ limited only to the symptoms more consistent with cybersickness, or the Cybersickness in VR Questionnaire (CSQ-VR) [72], which tries to improve on the two previously mentioned.
Alternatives to subjective questionnaires are referenced in [63] as physiological measurements including bradygastric activity, respiration rate, heart rate, and skin conductance at the forehead.

2.3.3 Data analysis

Multiple studies found in literature, such as [57][54][38], mentioned the use of SPSS to perform any statistical analysis on the results.
From the performance data collected (task completion time, number of correct steps or something else) from the two groups, experimental and control, is usually calculated the mean values and the standard deviation or variance of the results; these data alone provide valuable information about the distribution and central tendency of a dataset, but alone they do not establish statistical significance of the results. According to Encyclopedia Britannica[34], statistical significance is "the determination that a result or an observation from a set of data is due to intrinsic qualities and not random variance of a sample". This means that a result is unlikely to be a product of chance or sampling error (the sample observed is not representative of the whole population). The hypothesis testing method is used to determine the significance of a result, in the following process:

34. https://www.britannica.com/topic/statistical-significance

- A null hypothesis is set up, typically positing no distinction or correlation between the datasets;
- A level of significance α is defined, which is the threshold beyond which the null hypothesis will be rejected;
- An appropriate statistical test is conducted on the samples of data to generate a measure known as the p-value, which represents the likelihood of the observed results occurring if the null hypothesis holds true; a lower p-value signifies a higher probability that the event is not a result of random chance;
- The p-value is compared with the significance level threshold α; if the p-value falls below this threshold, the null hypothesis is rejected, indicating that the observed difference is unlikely to be due to chance alone, and the outcome is deemed statistically significant.

Conventionally a threshold of 0.05 is assumed, based on the assumption made by the famous statistician Ronald Fischer that an occurrence with a one in twenty chance indicates an uncommon sampling event [**73**], but this threshold is actually not based on scientific evidence [**74**], and in general should be dependent on the field of application. Multiple studies found in literature [**57**][**53**][**54**][**38**] studying the effectiveness of VR training used the 0.05 threshold.
The most common statistical tests used for evaluating the effectiveness of VR training compared to traditional training were found to be the *Independent Samples t-test*, also known as *Student t-test* or *Two-sample t-test*, or the *Mann-Whitney U-test*.
The *Independent Samples t-test* can be used under a number of assumptions[35]:

- Independent and random sample groups;
- Equal variances for the results of the two groups;
- Normal distribution for the results of each group.

For the groups to be independent (or independence of observations condition) means that the elements of a group cannot be part of the other, and measures should be put in place to prevent one group influencing the other; the randomicity condition ensures that the observed sample is representative of the larger population from which it was drawn.
Equal variances, or homogeneity of variances, assumes that the population variances of the two groups being compared are approximately equal; if they are not, and/or

35. https://libguides.library.kent.edu/SPSS/IndependentTTest

the sample size between the two groups is different, the p-value calculated from t-test may not be trustworthy. The *Welch's t-test*, also known as *Unequal Variances t-test*, is more reliable than Independent Samples t-test in case of unequal variances and possibly unequal sample sizes. Equality of variance can be tested, for example through the *Levene's test*.
The necessity of the assumption of normal distribution may be mitigated by "large" sample sizes for both groups, but in general the meaningfulness of t-test results and the p-value obtained from them is significantly diminished when dealing with non-normal population distributions, particularly those that exhibit thick-tails or that are heavily skewed. Normality can be verified through tests like the *Shapiro-Wilk (S-W) test* or the *Kolmogorov-Smirnov (K-S) test*. According to [75], as a rule of thumb *K-S test* should be used for sample sizes greater 50, while for smaller sample sizes the *S-W test* should be appropriate. If these tests could not confirm the assumption of normality, instead of t-tests the *Mann-Whitney U-test* should be used.
The *Mann-Whitney U-test* shares with the *Student t-test* the assumption of independent and random groups, but being a non-parametric test it does not make assumptions on the distributions of data (for a specific shape or for homogeneity of variance). However, to apply the *Mann-Whitney U-test* correctly a comparison between the shapes of the two data distribution must be made: if the shapes are similar, it is possible to establish a comparison between the medians of the results of the two groups, and if they are not, it is possible to compare the mean ranks of the two data distributions.

Chapter 3

Hardware & Software

This chapter describes the hardware (3.1) and software (3.2) components used for the activities of this thesis; all these elements have been provided by the Flight Simulation Laboratory of Politecnico di Torino, DIMEAS.

3.1 Hardware

3.1.1 VR workstation PC

The PC used as VR workstation to support the software described in the next paragraphs and interface with the VR hardware is an MSI Trident X Plus 9th[1] with the specifications of table 3.1.

Specifications	Description
Processor	Intel Core i7-9700K, 3.6 GHz (4.9 GHz Turbo), 8 cores, 8 threads
RAM	16 GB
Graphic card	Nvidia GeForce RTX 2080 Super
Storage	PNY CS2130 500 GB SSD
Operating system	Windows 10 Professional, 64-bit

Table 3.1: VR workstation PC specifications

1. https://it.msi.com/Desktop/Trident-X-Plus-9th/Specification

3.1.2 HTC Vive Pro

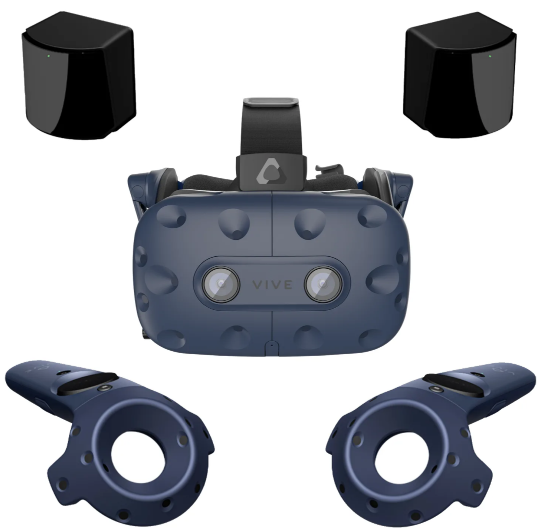


Figure 3.1: HTC Vive Pro headset, wands and base stations

HTC Vive Pro is a VR system developed by HTC in collaboration with Valve. The hardware needed for this system is portrayed in figure 3.1[2], and includes a headset, two base stations and two controllers. The headset is the user's window in the VR environment, the controllers are the means in which the user can interact with the VR world; both are equipped with sensors, which along the base stations define the tracking system. Base stations and controllers are connected to a PC via Bluetooth, while the headset can be connected to a PC through cables and a dedicated *Link Box* or with *Vive Wireless Adapter* for a cable-less solution.

2.1.2.1 Lighthouse tracking system

HTC Vive Pro implements room-scale tracking [3], which allows users to move with six degrees of freedom in the virtual environment. This is achieved through "Lighthouse", a laser-based inside-out positional tracking system [4]. In this system, each base station contain an array of LED lights, and two infrared lasers. The LED lights flash sixty times per second; after a flash occurs the lasers, mounted on spinners, sweep the play area vertically and horizontally. The receiver devices (headset and controllers) are covered with photosensors; when a flash occurs, the time for the laser to hit a specific photosensor is counted. Combining the information of when the photosensor is hit by the laser and where is the photosensor on the receiver device, and combining these information for multiple photosensors,

2. https://komete-xr.com/it/products/htc-vive-pro-full-kit-bws

3. https://xinreality.com/wiki/Room-scale_VR

4. https://gizmodo.com/this-is-how-valve-s-amazing-lighthouse-tracking-technol-1705356768

the position and orientation of the receiver devices can be calculated. To acquire an initial orientation and position 'lock', a receiver device must have at least five photosensors 'lit' by a base station (or three if two base stations are in view). Once locked, the devices can use data from internal inerial measurement unit (IMU) to track their own location in real-time, to provide supplementary motion tracking data during temporary interruptions in laser tracking, for example during the periods of time between laser sweeps or if (for any reason) all sensors have been temporarily blocked from view [5].

2.1.2.2 Headset

In the following table are presented the specifications of HTC Vive Pro headset[6].

Specifications	Description
Screen	Dual AMOLED 3.5" diagonal
Resolution	1440 × 1600 pixels per eye
Refresh rate	90 Hz
FOV	110°
Audio	Hi-Res certificate headphones
Input	Integrated microphones
Connections	Bluetooth, USB-C port for peripherals
Sensors	SteamVR Tracking, G-sensor, gyroscope, proximity, IPD sensor
Ergonomics	Eye relief with lens distance adjustment, adjustable IPD, adjustable headphones, adjustable headstrap
Weight	800 g

Table 3.2: HTC Vive Pro headset specifications

2.1.2.3 Controllers

Figure 3.2[7] shows all the buttons and switches of an HTC Vive Controller. The *Steam Menu* button is used to switch on and off the controller; relevant for the

5. https://pcper.com/2016/04/steamvr-htc-vive-in-depth-lighthouse-tracking-system-dissected-and-explored/2/

6. https://developer.vive.com/resources/hardware-guides/vive-pro-specs-user-guide/

7. https://survios.com/rawdata/guide/

activities of this thesis are also the *Trackpad* button and the *Trigger* button. In the Unity scene set up for the VR experience of this thesis, the former is used to control the teleport actions in the virtual scene, while the latter is used for every interactions with objects in the VR environment.

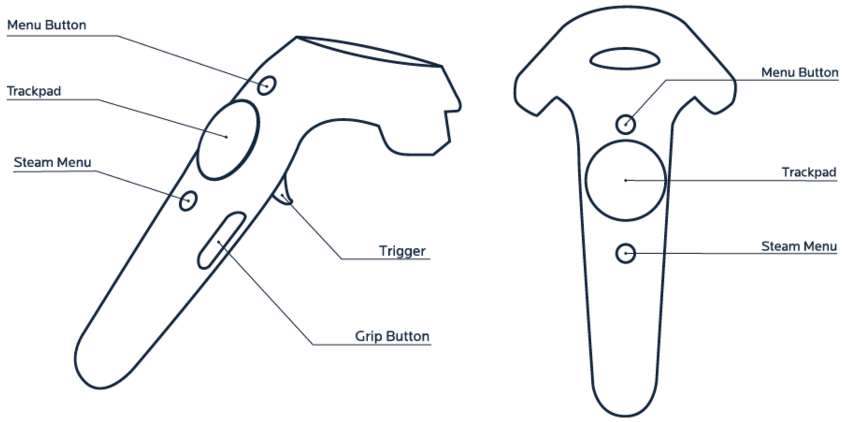


Figure 3.2: HTC Vive Pro controller

3.2 Software

3.2.1 SteamVR

SteamVR is a Virtual Reality Platform developed by Valve in partnership with HTC as an extension of Steam [8], designed to be compatible with multiple VR hardware devices, HTC Vive Pro included. SteamVR serves as a software platform that supports and works in conjunction with VR hardware devices, and is responsible for:

- Recognizing and communicating with the sensors of the tracking system, ensuring that the tracking data is correctly interpreted and used in the virtual environment;
- Integrating with the input devices or controllers, enabling communication between the controllers and the VR applications;
- Providing tools and guidance for users to calibrate their hardware, with features such as headset calibration, controller pairing, and tracking setup.

8. https://xinreality.com/wiki/SteamVR

For this thesis, SteamVR integrates with Pacelab WEAVR/Unity with a dedicated plugin, SteamVR Unity Plugin, allowing communication between the VR hardware and Unity; WEAVR also exploits some SteamVR components in some of its Prefab objects, described in paragraph 3.2.3.

3.2.2 Unity

Unity is a cross-platform game engine developed by Unity Technologies that supports a variety of desktop, mobile, console and virtual reality platforms, and supports C#, C++ and JavaScript as programming languages. The engine can be used to create three-dimensional (3D) and two-dimensional (2D) games, as well as interactive simulations [9]. The creation of a game or simulation starts with a Unity project, which acts as a workspace where users design, develop, and organize every element of their interactive experience. The Unity Editor is the application where a project can be edited, while Unity Hub is a management tool that helps users to organize and switch between Unity projects, install and manage different versions of the Unity editor, and install additional components. Starting a new project creates a local folder where users can organize and store all the project-related files, including:

- Assets which constitutes the building blocks of the Virtual scene, like 3D models, textures, audio files;
- Scripts, which define the behavior and functionality of objects and elements inside the project;
- Scenes, which are used by Unity to represent different levels, environments, or screens within your project, corresponding to different levels or sections of an application;
- Settings files, adjustable to optimize performance and further customize the behaviour of the applications.

What described above can also be found inside the project window, that opens when a project is opened or initialized. The project window (figure 3.3) displays all of the files related to the project and is the main way users can navigate and find assets and other project files inside the application[10].

A macro subdivision in six parts of the project window is shown in figure 3.3.

9. https://en.wikipedia.org/wiki/Unity_(game_engine)
10. https://docs.unity3d.com/Manual/ProjectView.html

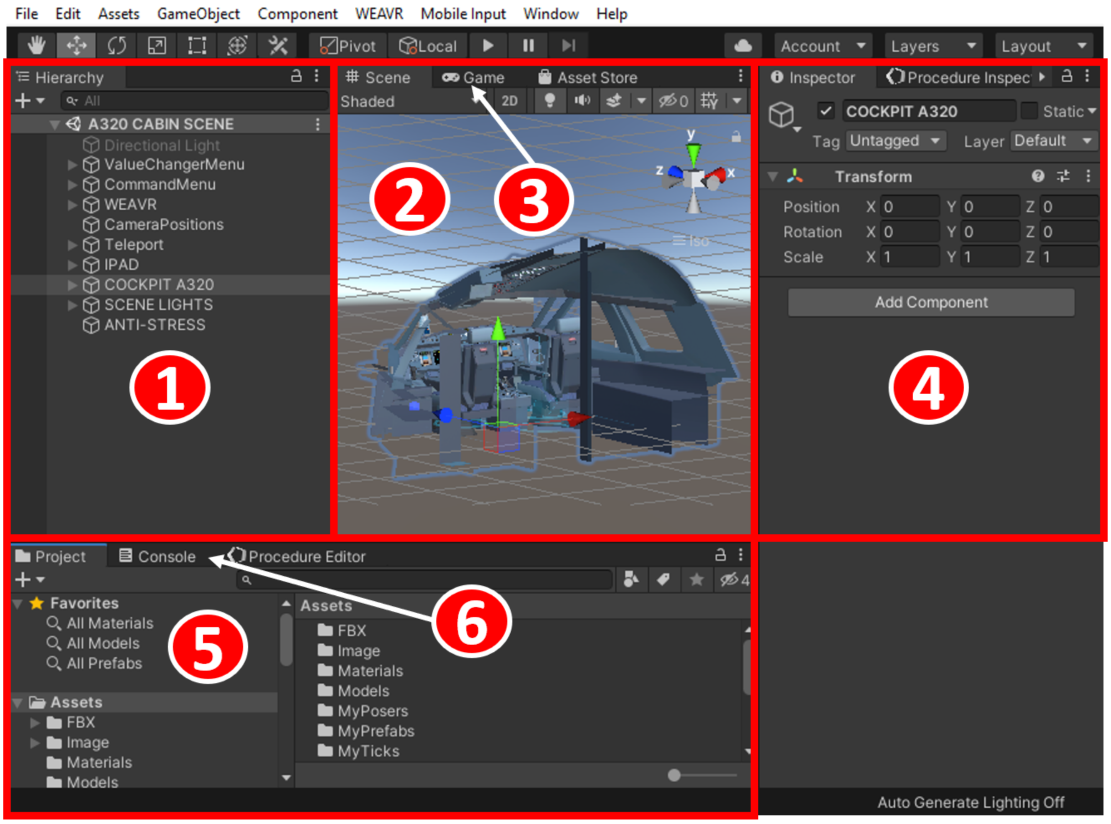


Figure 3.3: Unity project window

1. **Hierarchy**: this window displays every Game Object in a scene, such as models, cameras, or prefabs, and can contain multiple scenes. Unity uses the concept of parent-child hierarchies, or parenting, to group Game Objects; a child inherits the properties of a parent, so a user can associate a number of Game Objects to the same parent object for example to translate, scale, or rotate all of them at the same time;

2. **Scene**: The Scene view window is an interactive view that is normally used while editing a scene;

3. **Game**: The Game view window is rendered from the camera/s in the application. It represents how the application plays. When the user clicks on the "Play" button located in the toolbar above, Unity enters automatically Game view;

4. **Inspector**: this window can be used to edit and add Components to every Game Object; Components contain properties that characterize the behaviour

of a Game Object. Every Game Object possesses at least one Component, "Transform", that is used to define the position, orientation and scale of the Game Object inside a scene;

5. **Project**: This window contains every project-related file located in the local folder as described before;

6. **Console**: The Console Window displays errors, warnings, and other messages that the Unity Editor generates.

In figure 3.3 are displayed two additional windows, **Procedure Inspector** and **Procedure Editor** (section 5-6), which are present only after the installation of WEAVR in the Unity project. These two windows will be discussed in the next paragraph.

3.2.3 Pacelab WEAVR

Pacelab WEAVR [76] is a software toolbox for the design and development of virtual training systems developed by PACE, part of the TXT group. Its visual approach to design and development requires little to no programming or scripting skill, allowing subject matter experts to create virtual training systems without involving 3D Editors or software engineers. WEAVR is designed to promote a high level of reuse, integrating existing components and simulation modules, as well as deploying the same training content across multiple systems including desktop PC, mobile, and VR devices.
WEAVR comprises three main modules, Creator, Manager and Player. For the purpose of this thesis, the module of interest is Creator. WEAVR Creator is a plugin for Unity that enables the definition of behaviors and animations for Game Objects, implementing basic tools like Unity Components for animating buttons and levers, cameras and relative movement scripts, and a flow-chart editor to model procedures.

2.2.3.1 Installation

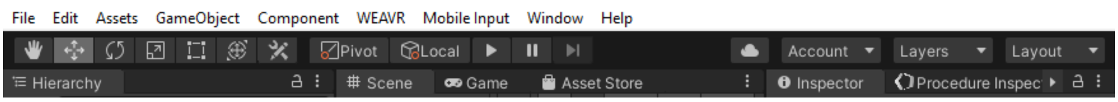


Figure 3.4: Unity Editor's upper toolbar

WEAVR can be installed in a Unity project importing the corresponding custom package, by the upper toolbar (figure 3.4) *Assets > Import package > Custom*

package; at the end of the installation, the Extension Manager window will appear, allowing the user to select the extensions needed for the project.
The same window can be accessed from the newly created WEAVR tab on the upper toolbar (figure 3.4), with *WEAVR > Setup > Manage Extensions*. To use WEAVR and its asset, is also necessary to set up the Setup scene window, with *WEAVR > Setup > Setup Scene*. The settings applied for the project described in chapter 5 are the ones displayed in figure 3.5.

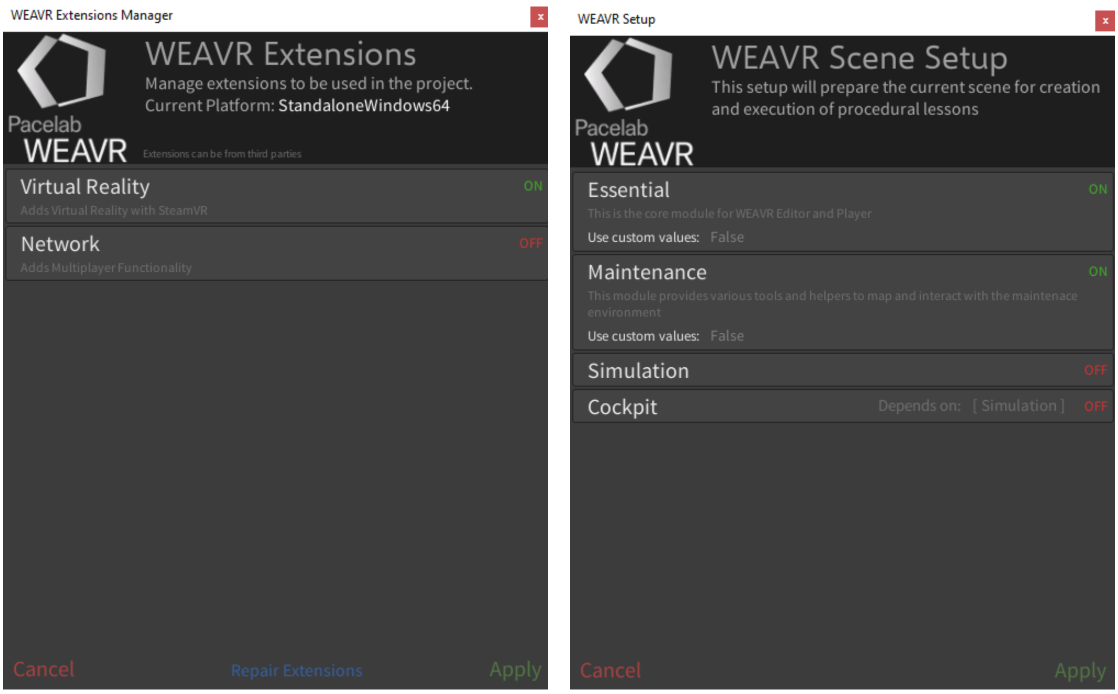


Figure 3.5: WEAVR Extensions Manager window (left) and WEAVR Setup Scene window (right)

At the end of these operations, with the Virtual Reality setting of the Extensions Manager window on "on" is enabled a toolkit for projects involving a Virtual Reality tool (using SteamVR and OpenVR Loader); with Essential and Maintenance on "on" in the Setup Scene window are enabled respectively the major Game Objects inside the newly generated WEAVR Game Object inside the hierarchy window of the project, and the main Components for objects interactions and behaviour. At this point the WEAVR Game Object is not completely set up, and needs more actions on its Components and its children objects components, specifically on the PlayerRIG Game Object and its children. For applications similar to the project described in chapter 5, the same project can be used as reference for the aforementioned components settings.

2.2.3.2 Components

As previously stated, WEAVR adds a number of Components to the Unity project when installed; Components relevant for this thesis are part of the Object Behaviour type (e.g. that allow Game Objects to behave in a specific way). The main ones used for the project of this thesis are presented in table 3.3.

Component	Description
VR_Object	Allows interacting with the Game Object using controllers
Interactions Controller	Keeps and controls the list of all the interactions associated with the Game Object
Grabbable	Enables the Game Object to be picked up and follow the movement of a controller
Executable	Triggers generic actions when interacting with the object
Push button	Allows to press the Game Object like a push button
2-Way Switch	Allows to interact with the Game Object like with a 2-way switch
3-Way Switch	Allows to interact with the Game Object like with a 3-way switch
N-Way Switch	Allows to interact with the Game Object like with a N-way switch

Table 3.3: Unity main Object Behaviour Components added by WEAVR

When a generic Component part of the Object Behaviour type is added to a Game Object, the Components "VR_Object" and "Interactions Controller" are automatically added to that same object.

2.2.3.3 SteamVR Components and prefabs

As mentioned in 3.2.1, WEAVR exploits some SteamVR components in some of its prefab objects. A prefab is a reusable template that defines a set of GameObjects with their components, properties, and settings. The specific prefabs mentioned are:

- **PlayerRIG**;
- **Teleport Area**;
- **Teleport Point**;

- **Hand Poses**.

The **PlayerRIG** prefab is included in the Hierarchy window as child of the WEAVR Game Object, and controls the player movements and interactions in the scene. PlayerRIG involves other components related to the player's ability to teleport in the scene and interact with objects through controllers/hands. Furthermore, the component includes the VR camera.
The **Teleport Area** prefab is used to define an area in which users can teleport using the controller.
The **Teleport Point** prefab allows the user to teleport to a specific point inside a Teleport Area using the controller. It can also be used to define a spawn point for the player when the interactive simulation in Unity is started.
The **Hand Poses** prefabs are used to render hands in the virtual scene when interacting with Game Objects. To set up a hand interaction when hovering over a specific Game Object, to that Game Object must be given the Component "Skeleton Poser". If this component is not assigned to a Game Object in the scene while the user hovers the controller over it is a hand grabbing a controller (figure 3.6 on the left), which is the default appearance of the hand/controller in game.

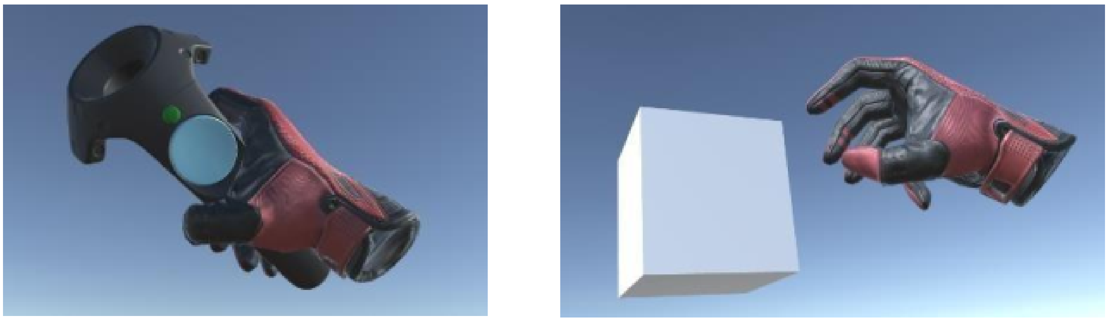


Figure 3.6: Example of hand pose [76]

2.2.3.4 Procedures

A WEAVR procedure consists of a orderly sequence of steps that determine its execution. Procedures are created with WEAVR Creator and executed with WEAVR Player; to create a procedure, the user needs to use the Unity upper toolbar (figure 3.4): *WEAVR > Procedures > Create Procedure.*
This opens the Create Procedure Wizard window (figure 3.7), where the user can set the procedure name and path inside the local folders, the configurations, the execution modes and the languages in which the procedure should be available.

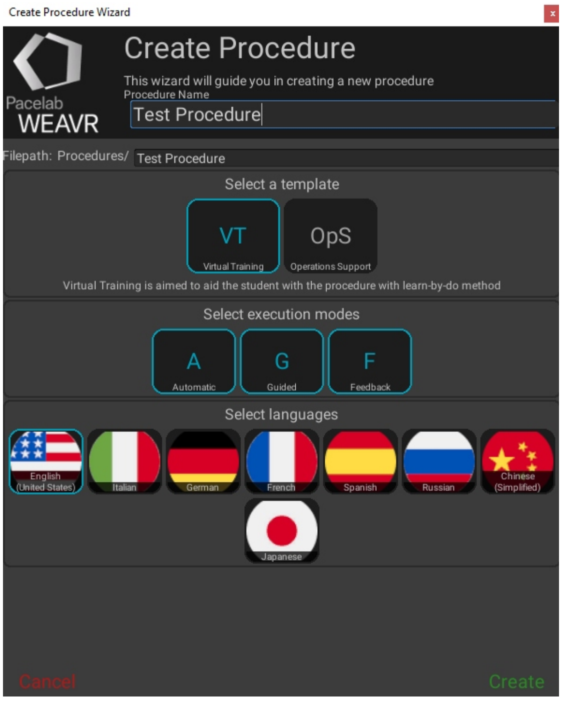


Figure 3.7: WEAVR Wizard Create Procedure window

Configuration and execution modes define the procedure type; their definition provide to WEAVR Player necessary informations like how to execute and if it can execute the procedure (configuration) and which steps to execute, what execute inside a step and how to execute it (execution modes). The procedure described inside this thesis uses the "Virtual Training (VT)" configuration, which usually allows user interactions and navigation with the virtual environment; the "VT" configuration is associated with three execution modes: "Automatic", "Guided" and "Feedback". The procedure of this thesis can be executed in any of these modes, but it was built considering specifically the "Guided" mode, which is specifically designed to help the user perform and complete a procedure, providing helping tools like guiding audios, billboards, images, videos, object highlighting, and animations.
Once the settings in the Create Procedure Wizard are applied, the **Procedure Editor** (figure 3.8) window is opened; the same window can be opened with: *WEAVR > Procedures > Procedure Editor.*

The **Procedure Editor** allows the user to create and edit a procedure. The toolbar of this window allows the user (among other things) to create or load a procedure, to start the procedure when the play button in the Unity Editor is clicked (activating the **Test** option) and select the execution mode.
The main view of the procedure is the procedure graph, where the user can create and edit the procedure steps, or Nodes. Nodes can be connected through links called Transitions, and multiple connected nodes can be grouped in structures called Super Steps. There are different types of nodes, but in the following description the focus will be on the most basic type. When a Node is created (for a basic Node, *mouse right click on the procedure graph > Create Node*), the **Procedure Inspector** window opens. This window allows you to see the details of procedure items (Nodes, Groups, Transitions, etc.) and modify them.

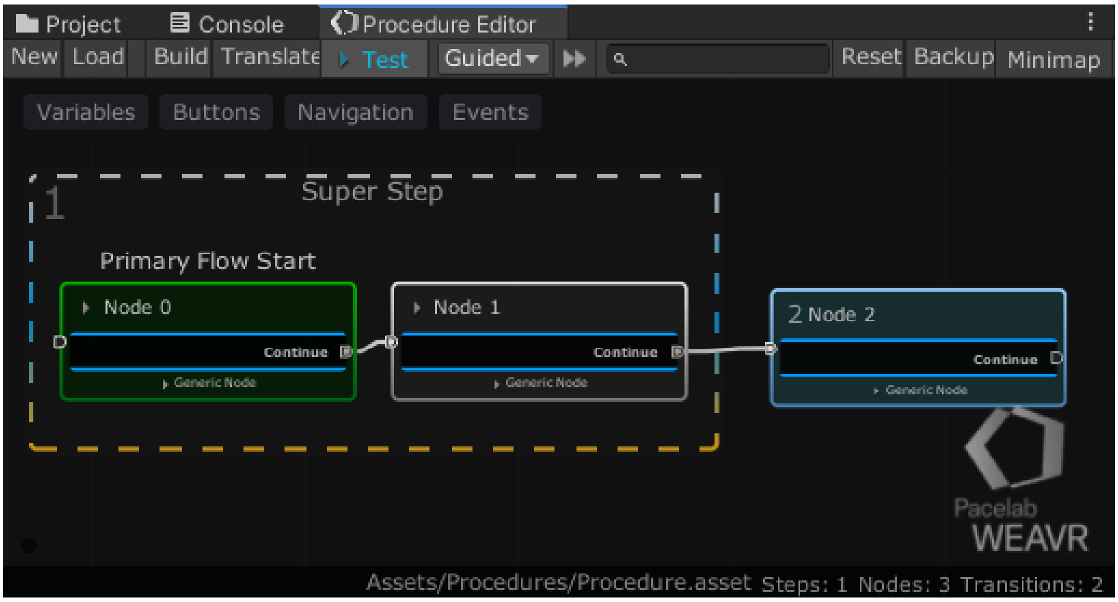


Figure 3.8: WEAVR Procedure Editor window

The Procedure Inspector for a Node is a Figure Inspector, as shown in 3.9. For a basic node there are three main items you can define:

- **Enter Actions**;
- **Exit Conditions**;
- **Exit Actions**.

An Action is used to inform the user about something through multiple possible means (audio clips, billboards and simple text messages, and others), to change the position of a camera (for example, to have a closer view of an object), to show or hide an object from a user, and many more. An **Enter Action** is an Action executed when the procedure flow arrives to the Node, and an **Exit Action** is executed when the procedure flow passes the Node. For the procedure flow to pass from one Node to the next, one or multiple **Exit Conditions** must be satisfied. An **Exit Condition** is an evaluation element that needs to be true/false to trigger the procedure flow to continue.
In the tables 3.4 and 3.5 are provided the main Actions and **Exit Conditions** used for this thesis, with the information related to the belonging group of these items, as described in [76].
The actions of the Hints group are used to provide support to the user in terms of visual pointers, while the "Text To Speech" Action is used to provide an audio

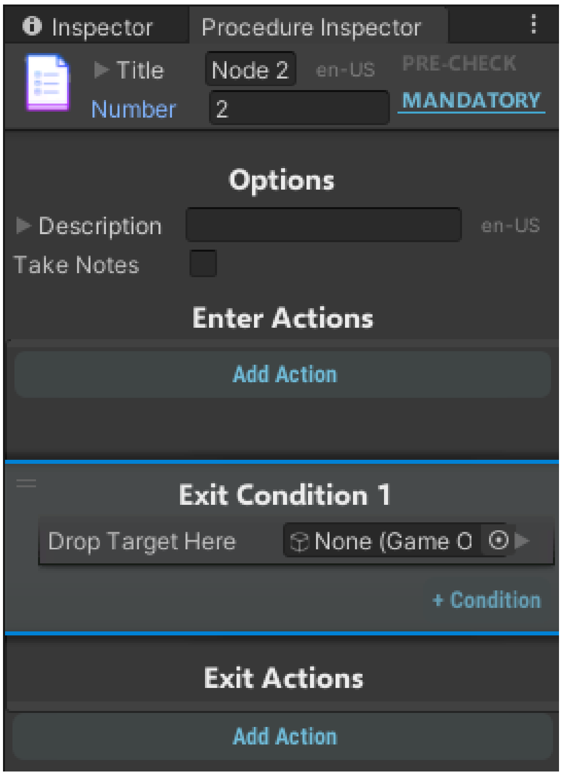


Figure 3.9: WEAVR Node Inspector

support for navigation. The Actions of the Object group are used to prepare the scene entering or exiting a procedure Node, in terms of Game Objects visible on the scene ("Toggle Object"), states of switches, levers and buttons ("Set Generic Value") and Game Objects behaviour ("Toggle Component"); this last Action in particular is used to enable or disable Components that allow a switch or button behaviour in different points of the procedure flow, to avoid inconsistencies with the actions that the user is asked to perform during the procedure and the state of the scene in terms of switches and buttons positions and what is shown on screen for the different displays on the scene. The **Exit Condition** "Generic Value" is used when to proceed in the procedure flow it is needed to confront the current state of an interactive Game Object with a desired state. This condition most of the times implies an interaction of the user with a Game Object. "Object is active" is used to continue inside a procedure when the actions of the user cause some Game Object inside the scene to activate or deactivate. "Visually Inspect" is used when the procedure requires the user to check visually any Game Object on the scene; the user can define a minimum distance for the visual inspection, and an inspection time. If the user is too distant a text message will prompt the user to get closer, otherwise a circular bar will appear, which will be filled in the inspection time set by the user (figure 3.10).

Action	Group	Description
Text to Speech	Audio	Converts a text written by the user into an audio clip and plays it
Wait Time	Control Flow	Creates a timeout of a desired amount of time between the Action before and the Action after
Show Billboard Expert	Hints	Assigns a billboard to a Game Object target with a text written by the user. It can be used to activate an outline on an object and create a 3D navigation arrow that points the Game Object target
Outline Object	Hints	Outlines a Game Object target with a desired color. It can be used to create a 3D navigation arrow that points the Game Object target
Hide Billboard and Outline	Hints	Deletes billboards and outline from a Game Object
Toggle Object	Object	Enables or disables a Game Object
Toggle Component	Object	Enables or disables a Component of a Game Object
Set Generic Value	Object	Edits the settings of a Component of a Game Object

Table 3.4: Main Actions for a WEAVR procedure

Exit Condition	Group	Description
Generic Value	Generic	Checks the settings of a component of a Game Object to compare them with control settings given by the user
Object is active	Object	Checks if a Game Object is enabled or disabled
Visually Inspect	Object	The condition is verified when a target Game Object is visualized in the VR Camera from a distance and for a time specified by the user

Table 3.5: Main Exit Conditions for a WEAVR procedure

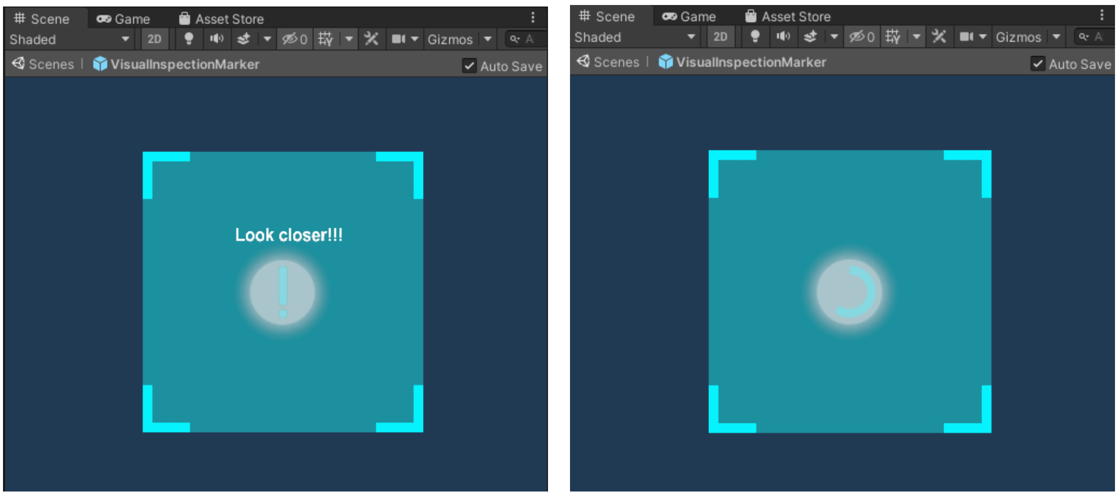


Figure 3.10: Visual inspection: user too distant (left) and user within the maximum distance radius for inspection (right)

3.2.4 X-plane

X-Plane is a flight simulation engine series developed and published by Laminar Research available as a desktop and mobile application. X-plane is highly regarded for accurate flight physics and realistic aircraft behaviour; this reputation stems from its modeling of aerodynamics and, if applicable, propeller performance using the Blade Element Theory[11]. With its wide database of airports and military and commercial planes, its detailed map environments based on real world geographic data, the possibility for users to add custom airports, planes, sceneries and custom made plugins, the uses of X-Plane range from gaming to professional training, including its application in FAA-certified simulators [12].
For the purpose of this thesis, X-plane 11 was used with the commercial aircraft A321 developed by Toliss, which was chosen for previous thesis works [4][3] because of:

- A good compromise between graphic fidelity of the cockpit and the frame rate needed for a real-time simulation, given the hardware described in 3.1.1;
- A huge number of *DataRefs* [13], which represent information published by the simulation. With custom plugins, scripts, and add-ons users can edit (if writable) or read *DataRefs* from an X-Plane simulation in real time.
- The Toliss Interactive Simulation Control System plugin, that allows the saving and loading of the simulation of specific points in time, for example the starting condition of a flight phase like landing or take-off;
- The compatibility between the model and the cockpit hardware for the A320 flight simulator inside the Flight Simulation Laboratory of Politecnico di Torino.

For the project described in chapters 4 and 5 X-plane was used to produce a number of screens for the displays of an A320 cockpit, specifically for the Engine and Warning Display (or upper ECAM), the System Display (or lower ECAM), the Navigation Display (ND) and the Multipurpose Control and Display Unit (MCDU). More details on these displays and the related assets produced for the Unity project are given in the next chapter.

11. `https://www.x-plane.com/desktop/how-x-plane-works/`
12. `https://www.x-plane.com/pro/`
13. `https://developer.x-plane.com/2009/04/datarefs-vs-commands-i-whats-the-difference/`

Chapter 4

A320 Cockpit

The Airbus A320 family[1] is a series of narrow-body commercial jet airliners developed and produced by Airbus. The family includes the A320 and its variants A321, A319 and A318 (in chronological order), which differ from the original mainly in terms of length and seating capacity. As of today, these aircraft are one of the highest-selling airliners of all time and are widely used by airlines around the world for short to medium-range flights.
The A320 Family benefits from the advantages of Airbus commonality philosophy (i.e. similarity and shared components among different aircraft models), one of which is that pilots can fly the aircraft of the A320 family with a Single Type Rating thanks to their identical cockpits and operating procedures[2]. Some of the notable design choices of the cockpit of the A320 family, according to [**77**] and [**78**], are:

- Functional grouping for controls and displays in dedicated panels, as well as an arrangement of panels based on frequency of use, importance, duplication of controls (if required);
- The practice of dimming or turning off non-essential lights, indicators, or displays in the cockpit during normal flight operations ("Dark Cockpit" philosophy), to create a less visually distracting environment for the pilots;
- Consistent colour coding for displays and lights: red and amber for critical and non-critical warnings, green, blue and white for normal operations.

All these choices contribute to enhance situational awareness, pilot efficiency, and safety.

1. https://en.wikipedia.org/wiki/Airbus_A320_family
2. https://www.airbus.com/en/products-services/commercial-aircraft/cockpits

This chapter provides a general description of the A320 cockpit main sections and panels (4.1), with a focus on the ones that are relevant for this thesis. The chapter then follows with a brief description of the A320 flight simulator of the Flight Simulation Laboratory of Politecnico di Torino (4.2). Finally, at paragraph 4.3 is given a description of the Cockpit used for the Unity project used for the training procedures in VR of chapter 5. The pictures used in paragraph 4.1 are from the site flybywiresim[3], from Glyn Chadwick's portfolio [4], from [77] or from the A320 cabin of paragraph4.3.

4.1 A320 cockpit description

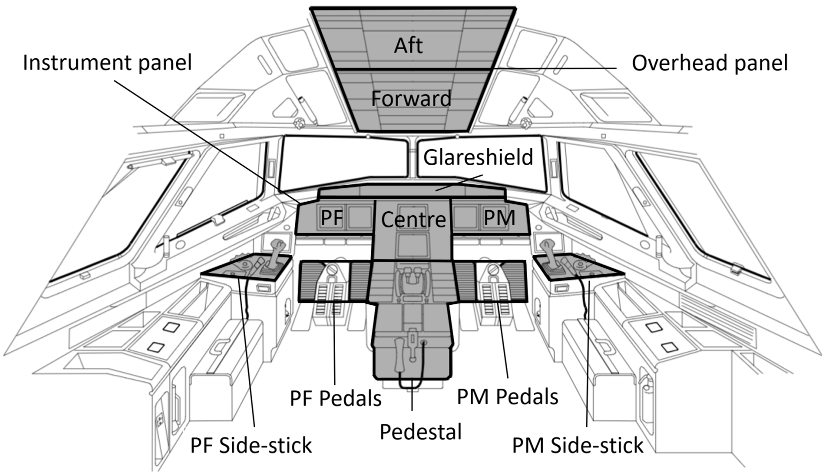


Figure 4.1: A320 Cockpit subdivision diagram

The A320 cockpit can be divided in a number of sections, as shown in figure 4.1:

- Side-stick and pedals (PF and PM side);
- Pedestal;
- Instrument panel, which can be divided in left/PF panel, right/PM panel, Centre panel and Glareshield;
- Overhead panel, which can be divided in forward and aft panels.

3. https://docs.flybywiresim.com/pilots-corner/
4. https://gchadwick.myportfolio.com/a320-cockpit-diagram

The side-stick and pedals are used by the pilot flying (PF) or the pilot monitoring (PM) to control pitch, roll and yaw of the aircraft. Further details on the other sections are provided in the next paragraphs.

4.1.1 Pedestal

The pedestal serves as a central control area for various systems and functions of the aircraft, as shown in figure 4.2. The panels and sections of the pedestal that are described in the next sub-paragraphs are specifically the ECAM control panel, the MCDU, the Radar panel, the Speed Brake, the Parking Brake, the Rudder trim panel, the Engine panel, the thrust levers and Pitch Trim Wheel, the Flaps lever and the ATC panel.

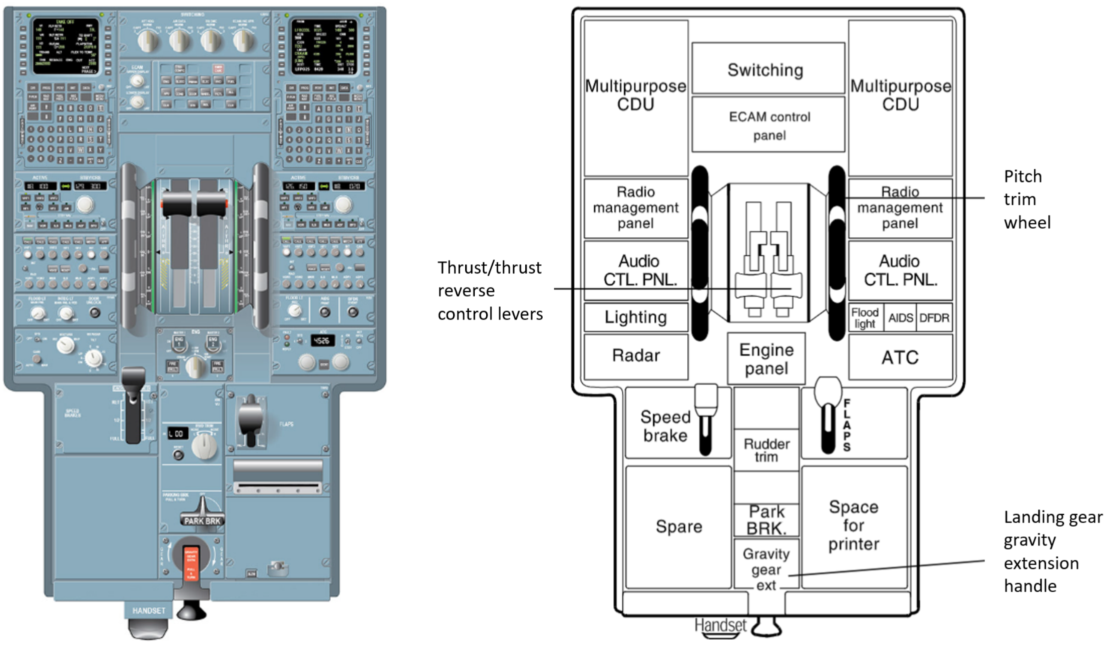


Figure 4.2: A320 Cockpit: pedestal (left) and panels diagram (right)

3.1.1.1 ECAM control panel

This panel contains a series of buttons that allow the user to select the page displayed in the lower ECAM, or System Display. Specifically, ENG shows the engine page, FUEL shows the fuel page, DOOR opens the doors and cockpit pressure page, and STS the status page. The TO CONFIG button is used to verify that the airplane is in the take-off configuration.

Figure 4.3: ECAM control panel

3.1.1.2 MCDU

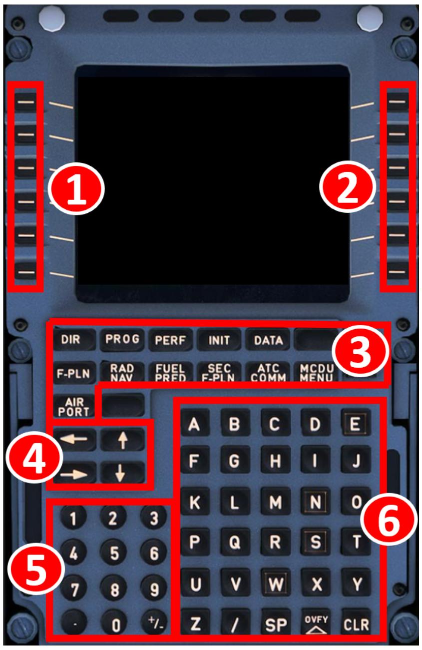


Figure 4.4: MCDU

The Multipurpose Control and Display Unit (MCDU) is the main interface for the pilots with the Flight Management Guidance System (FMGS). The Pedestal mounts two independent MCDUs for the PF and PM. The MCDU is used mainly for the selection of a flight plan for in terms of trajectories as well as speed profiles; the users can also modify performance and navigation data in the appropriate pages. The main pages used for this purpose are shown and described in paragraph 5.1.2.

To navigate through this device, users can interact with the numerated elements of figure 4.4:

1. Line Selector Keys Left (LSKL): ordered from 1 (upper) to 6 (lower), they are used to select lines in the MCDU display to navigate through the MCDU pages or to insert data;

2. Line Selector Keys Right (LSKR): same as the LSKL;

3. MCDU Pages Keys: each of these buttons opens one of the main pages of the MCDU;

4. Slew Keys: these buttons are used to scroll through the same page, for example to visualize the A and B sections of the INIT page (horizontal keys) or to visualize all the waypoints of a flight plane (vertical keys);

5. Numeric Keyboard: Together with the alphabetic keyboard, this is used to write data on the MCDU's current page displayed. The data is then inserted in the appropriate lines of the page through LSKR and LSKL;

6. Alphabetic Keyboard.

3.1.1.3 Weather Radar Panel

Figure 4.5: Weather Radar panel

This panel controls the weather radar system of the A320, with controls that allow pilots to adjust the radar's operational modes, tilt angle, range, and gain/sensitivity settings. The radar is also equipped with a Predictive Wind-shear System (PWS), which can be activated with the dedicated switch set on AUTO. The radar can be activated with the SYS switch set on 1 or 2.

3.1.1.4 Speed brake lever

The Speed Brake lever (figure 4.6 on the left) deploys the spoilers, and can be specifically used to set the ground spoilers if set to GND SPLRS ARMED (i.e. set to RET, and then pulled up).

3.1.1.5 Flaps lever

The Flaps lever (figure 4.6 on the right) controls the flaps and slats of the aircraft. It has five modes, 0, 1, 2, 3 and FULL, given in order of growing flaps/slats angle.

Figure 4.6: Brake Speed lever (left) and Flaps lever (right)

3.1.1.6 Parking Brake Handle

The Parking Brake handle (figure 4.7 on the right) can be rotated between the ON and OFF position to apply or remove the parking brake. When the Parking Brake is on, a memo is displayed on the upper ECAM.

3.1.1.7 Rudder Trim panel

This panel (figure 4.7 on the left) can be used to set the trim settings for the rudder. The RUD TRIM rotatory switch controls the rudder trim actuator, which moves the neutral point of the artificial feel; the RESET button sets it to zero. After having pushed the RESET button the pilot may observe a residual indication up to 0.3 ° left or right in the display.

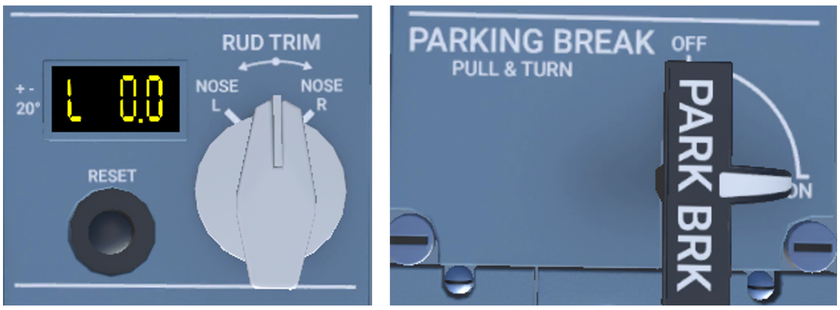


Figure 4.7: Rudder Trim panel (left) and Parking Brake handle (right)

3.1.1.8 Engine panel

The Engine panel (figure 4.8 on the right) is used for the control and start of the engines. The Engine Master Switches are set to ON with the purpose of communicating to the Full Authority Digital Engine Control (FADEC) to begin the automatic or manual start sequence for engine 1 or 2. The engine mode selector rotary switch is set to NORM during normal operations, to START to begin the automatic or manual start sequence; finally, the CRANCK mode is used predominantly by maintenance personnel for performing various engine tests and for troubleshooting purposes[5] (in this mode the engines igniters are deactivated and fuel cannot be introduced in the engines).

3.1.1.9 Thrust Levers and Pitch Trim Wheel

The two thrust levers (one per engine) are used by the pilots to set the thrust level. When the thrust levers are moved, they transmit signals to the FADEC, which computes the thrust rating limit and engine pressure ratio (EPR). The levers can be moved between the detents TO/GA (take-off/go around), FLX/MCT (flex/maximum continuous thrust), CL (climb), Idle, FULL REV (Reverse thrust), each one corresponding to specific thrust settings for important phases of flight. The pitch trim wheels at the side of the thrust levers provides mechanical control over the Trimmable Horizontal Stabilizer (THS).

Figure 4.8: Thrust levers and Pitch Trim wheel (left) and Engine panel (right)

3.1.1.10 ATC panel

The ATC/TCAS (Air Traffic Control/Traffic Collision Avoidance System) panel provides controls and indications related to communication with air traffic control

5. https://aviationinfo.net/airbus-engine-mode-selector-crank/

and the TCAS system. The aircraft has two ATC transponders, and one can be activated with the central left lateral knob, provided that the upper knob is set to ON or AUTO (figure 4.9). The keyboard is needed to enter the ATC four-digits code provided by ground-based air traffic controllers (2000 is the code for airplanes which have not been assigned a transponder code).

Figure 4.9: ATC/TCAS panel

4.1.2 Instrument Panel

The main Instrument Panel is a display area that provides essential flight information to the pilots. As shown in figure 4.1, the Instrument Panel can be divided in four sections, each one with a number of notable elements that will be described in the following paragraphs:

- Centre Instrument Panel, which contains the ECAM and the Autobrake and Gear indications and lever;
- Left (PF) and right (PM) Instrument Panels, which both contain a Navigation Display (ND) and a Primary Flight Display (PFD);
- Glareshield, which contains the EFIS control system and the FCU.

A complete visualization of the elements of the main Instrument panel is shown in figure 4.10.

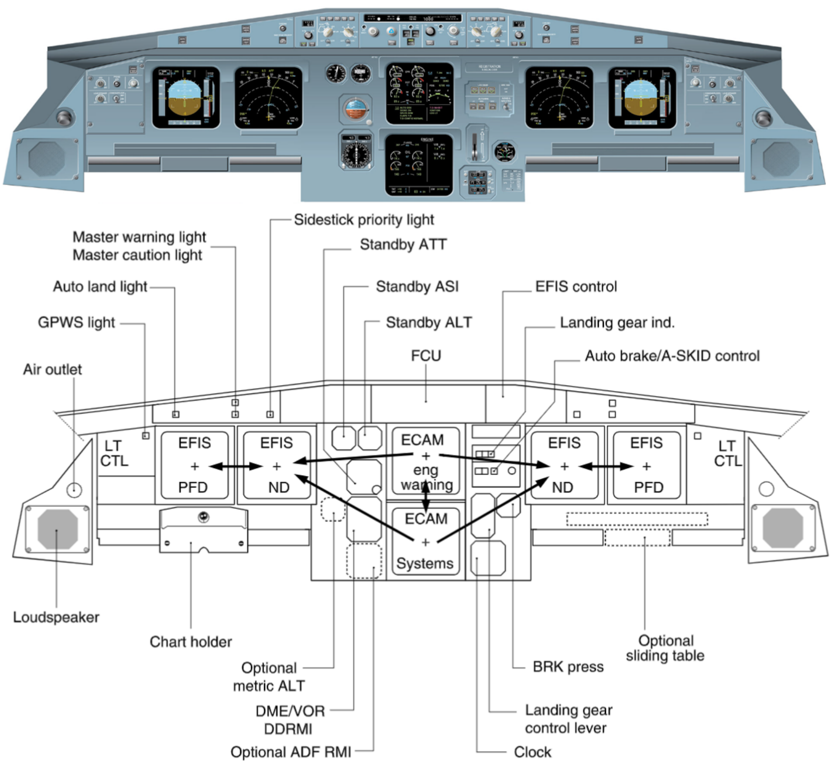


Figure 4.10: A320 Cockpit: Instrument panel (up) and panels diagram (down)

3.1.2.1 ECAM

The Electronic Centralized Aircraft Monitoring (ECAM) provides pilots with centralized monitoring, control, and alerting of various aircraft systems and parameters. The ECAM is made up of::

- The Engine & Warning Display (E/WD), or upper ECAM;
- The System Display (SD), or lower ECAM.

The E/WD (figure 4.11 on the left) contains primary engines indications (N1, N2, fuel flow and EGT), fuel quantity on board, flaps and slats position, and warning and caution alerts or memos. The SD (figure 4.11 on the right, Engine page) contains in its pages diagrams and information on numerous aircraft systems and flight permanent data; the different pages can be displayed using the ECAM control panel.

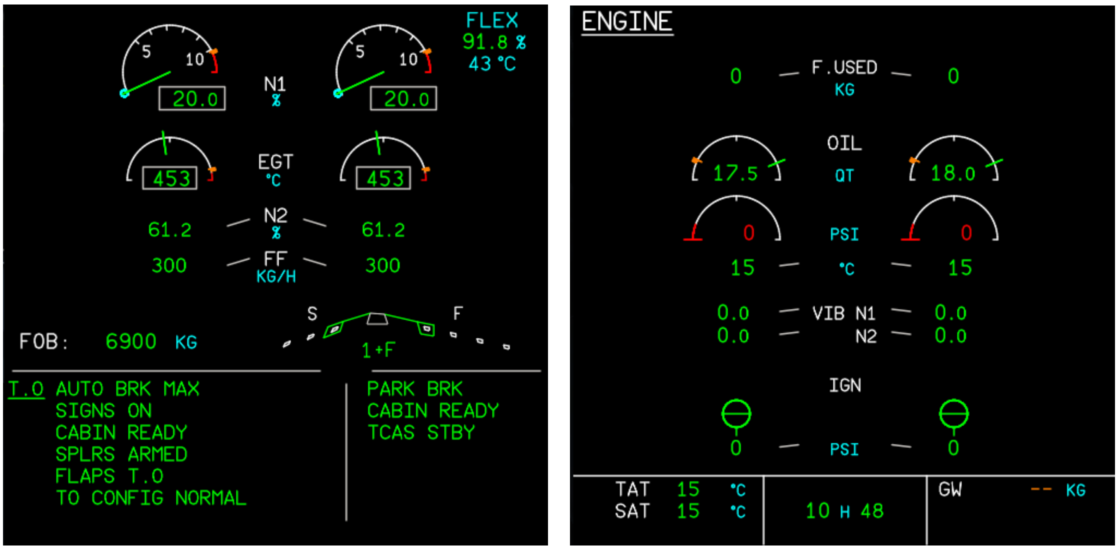


Figure 4.11: Upper ECAM (left) and lower ECAM - Engine page (right)

3.1.2.2 Autobrake and Gear indications panel

The A320 has an autobrake system which is usually used in case of an aborted takeoff (MAX button) or after landing (LO and MED buttons) and maintains a selected deceleration rate. The buttons when the correspondent autobrake mode is activated display a blue ON; if the deceleration reaches the 80 % of the corresponding deceleration for the selected mode, the buttons will display a green DECEL. The landing gear lever (figure 4.12 on the right) is used to retract (up) and deploy (down) the landing gear.

Figure 4.12: Autobrake and Gear indications panel

3.1.2.3 Navigation Display & EFIS control panel

The Navigation Display shows navigation data provided by the FMGS; ground speed (GS), true air speed (TAS), wind direction and speed, magnetic heading, selected heading and actual track are information always displayed. There are five different modes [6] selectable with a dedicated rotary selector in the EFIS control panel:

- ROSE ILS provides ILS frequency, course and identification, and Localizer and Glide slope deviation;
- ROSE VOR provides VOR frequency, course and identification, and VOR radial deviation;
- ROSE NAV shows the current route flown and waypoints with the aircraft centered. This mode is usually used to achieve a global view for a landing/take-off procedure that requires a 180°;
- ARC is the most used mode during flight; it shows the aircraft on its route with its waypoints in a limited field of view;
- PLAN is a mode used to verify the flight-plan; the screen is centered on the waypoint selected through the MCDU.

The range of these display pages can be modified with another rotary selector in the EFIS control panel. These pages are shown in figure 4.13[7].

6. https://wiki.ivao.aero/en/home/training/documentation/Navigation_Display_-_ND

7. http://www.airbusdriver.net/EFIS5.pdf

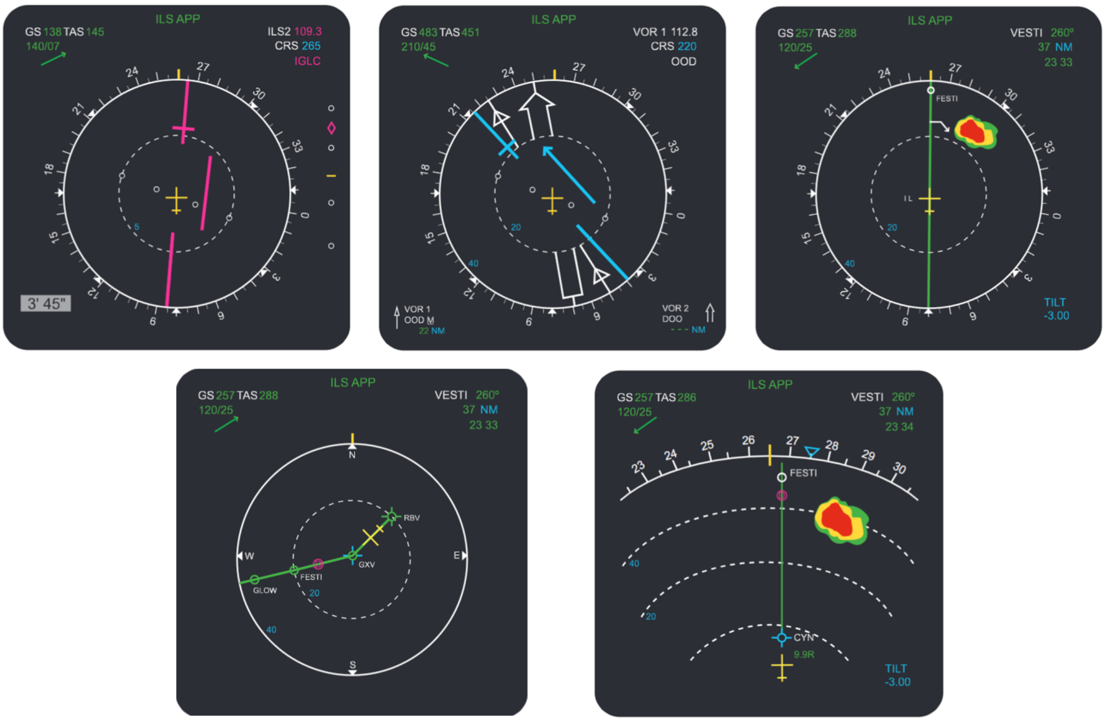


Figure 4.13: Navigation Display modes: ROSE ILS, VOR, NAV and ARC and PLAN (left to right, up to down)

Other then controlling the modes and range of the Navigation Display, the EFIS control panel can be used to set the barometric reference shown in the Barometer Reference Display Window with the left knob (figure 4.14).

Figure 4.14: EFIS control panel

3.1.2.4 Primary Flight Display

The Primary Flight Display (PFD) provides essential flight information for both pilots. This display can be roughly divided as shown in figure 4.15, in Flight Mode Annunciator columns (1), Indicated Airspeed indicator (2), Attitude and Guidance window, with a graphical representation of the aircraft's pitch and roll attitude (3), altitude indicator (4), vertical speed indicator (5), ILS information (6), magnetic heading indicator (7) and barometric reference (8).

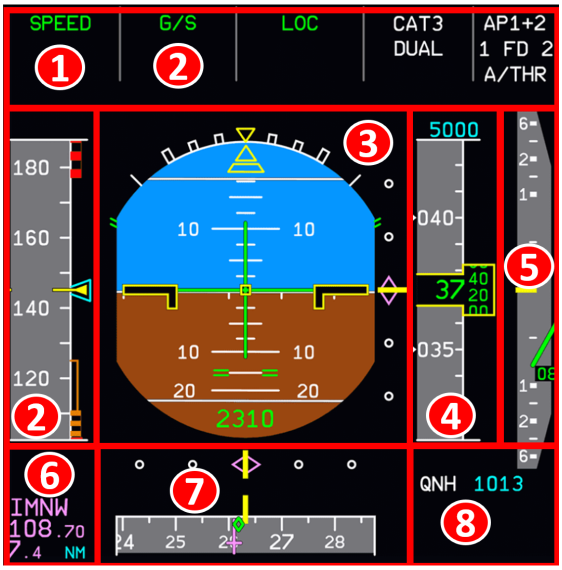


Figure 4.15: Primary Flight Display

3.1.2.5 Flight Control Unit

The Flight Control Unit (FCU) is an interface between the pilots and the Flight Management and Guidance Computer (FMGC), for example for activating and deactivating the autothrust (A/THR button), for engaging/disengaging the autopilots (AP1 and AP2 buttons) or for activating/deactivating the approach modes (APPR button). The four knobs of the FCU, SPD-MACH (speed), HDG-TRK (heading), ALT (altitude), V/S-FPA (vertical speed) can be used by the pilots for the selection of the reference flight parameters for the autopilot.

Figure 4.16: Flight Control Unit

4.1.3 Overhead Panel

Overhead panel plays a crucial role in controlling and monitoring multiple aircraft systems. For this purpose the Overhead forward panel (figure 4.17) is certainly the most important, and in the next paragraphs will be provided a brief overview of some of its main panels relevant for this thesis: ADIRS panel, Voice Recorder panel, Calls panel, Fire Control panel, Air condition control panel, Anti-ice panel, Exterior Lighting panel, APU control panel, Signs panel.

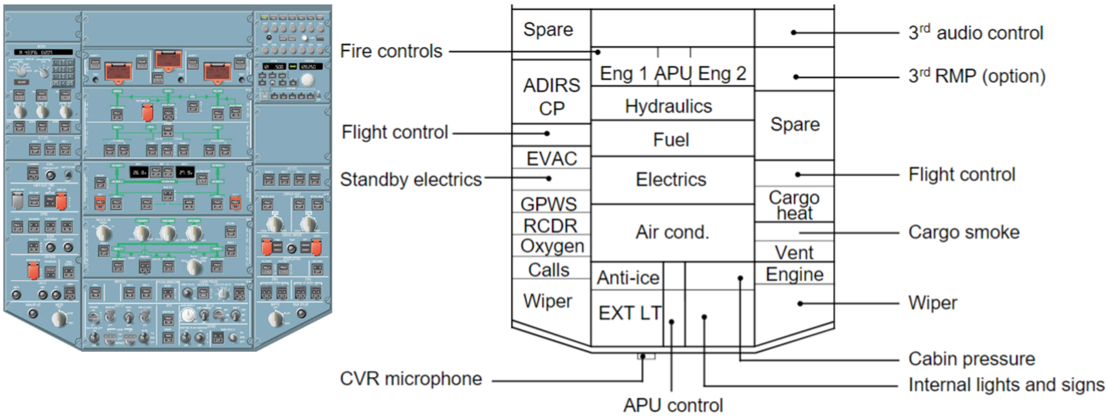


Figure 4.17: A320 Cockpit: forward Overhead panel (left) and panels diagram (right)

3.1.3.1 ADIRS panel

The Air Data Inertial Reference System (ADIRS) panel (figure 4.18) controls three air data/inertial reference units (ADIRU) present on the A320. Each ADIRU has an Air Data Reference system (ADRS), which provides data like barometric altitude, airspeed, Mach number, angle of attack, temperature, and an Inertial Reference (IR) system , which provides attitude, flight path vector, heading, track, acceleration (speed trend), vertical speed, groundspeed, and an inertial position input to provide to the FMGC for navigation computations together with the GPS. In aircraft normal operations, the IR mode selectors 1, 2 and 3 must be set to NAV, in this order; when the selectors are set from OFF to NAV, an amber light will appear on the ALIGNING indicator above, segnaling an alignment of the IRs in progress.

Figure 4.18: ADIRS panel

3.1.3.2 Voice Recorder panel

The Voice Recorder panel (figure 4.19) is made up of three buttons, namely the GND CTL, the CVR ERASE and the CVR TEST buttons. GND CTL is set to ON on the ground to energize to energize the cockpit voice recorder (CVR) and the digital flight data recorder (DFDR). CVR TEST, if pressed and maintained, activates the CVR test, in which a low frequency sound is emitted from the cockpit's loudspeakers; the CVR ERASE button if pressed and maintained deletes the CVR recordings.

Figure 4.19: Voice Recorder panel

3.1.3.3 Calls panel

The Calls panel (figure 4.20) allows the cockpit to initiate calls with the flight attendants and the ground crew. Specifically, the MECH button allows communication with the external service interphone panel, located near the nose gear on the External Power Panel. The ALL, FWD and AFT buttons allow the pilots to communicate with the flight attendants in the cabin, FWD for the forward call area, AFT for the AFT call area, ALL for both.

Figure 4.20: Calls panel

3.1.3.4 Fire Control panel

The Fire Control panel (figure 4.21) provides the pilots an interface to the fire and smoke detection system of the A320, specifically for fires or overheating of the APU and the two engines. The TEST buttons are used to test the fire detection and extinguishing system; when pushed and maintained, a repetitive chime sound is emitted from the loudspeakers of the cockpit, the AGENT buttons for the corresponding device and the ENG 1/ENG 2/APU FIRE red buttons lighten up; this happens also to the glareshield MASTER WARNINING indicators, and in the case of ENG 1 or 2 also the FIRE indicator on the engine panel; for all three buttons the upper ECAM produces warning messages if they are pushed.
The AGENT buttons are used to activate the fire extinguishers, while the FIRE buttons when pushed arm the fire extinguishers squibs, cut off energy for the FADEC and closes low-pressure fuel valve, hydraulic fire shut off valve, engine bleed valve, pack flow control valve.

Figure 4.21: Fire Control panel

3.1.3.5 Air Condition control panel

The Air Condition control panel (figure 4.22) allows the control of the two independent Pneumatic Air Conditioning Kit (PACKS) , which are responsible for controlling and regulating the temperature and pressure of the conditioned air supplied to the cockpit and the cabin. This panel also has buttons to control engines 1 and 2 bleed (ENG 1 BLEED, ENG 2 BLEED), a cross bleed selector to control the transfer of bleed air between engines, and the APU BLEED button, which is used to control if the flow of bleed air from the Auxiliary Power Unit

(APU) is available for use (on ON, blue light) for various aircraft systems such as air conditioning, anti-icing, and pressurization; the APU's bleed air can also be used for the engine start sequence.

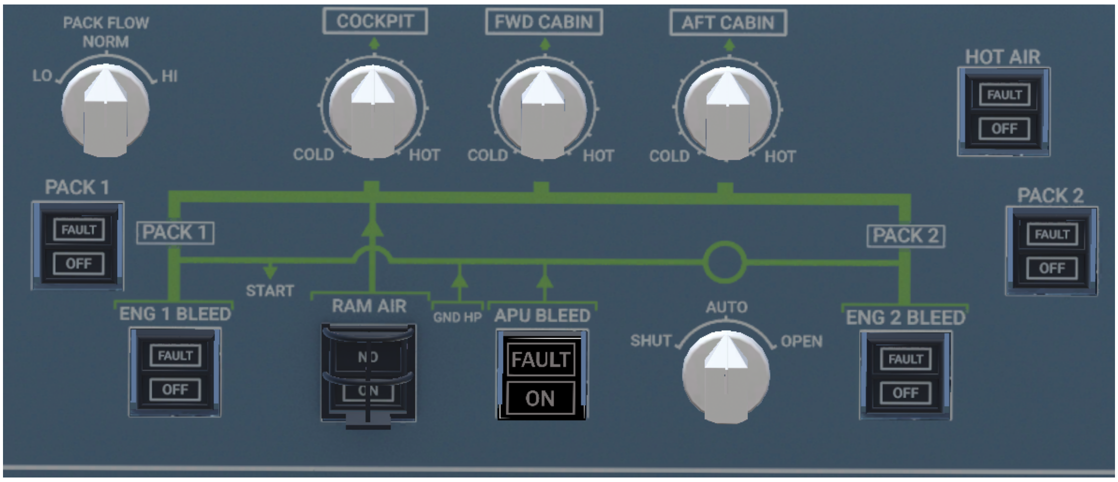


Figure 4.22: Air Condition control panel

3.1.3.6 Anti-ice panel

The Anti-ice panel (figure 4.23) controls the activation of anti-icing for critical areas of the aircraft, specifically for the three outboard leading-edge slats of each wing (WING button) and for the engines air intakes (ENG 1 and ENG 2 buttons). When on ON, the three buttons show a blue light.

Figure 4.23: Anti-ice panel

3.1.3.7 Exterior lighting panel

The Exterior lighting panel is used to control the external lights of the A320, with the switches in figure 4.24: BEACON (1), WING (2), NAV&LOGO (3), NOSE (4), LAND (5), RWY TURN OFF (6), STROBE (7).

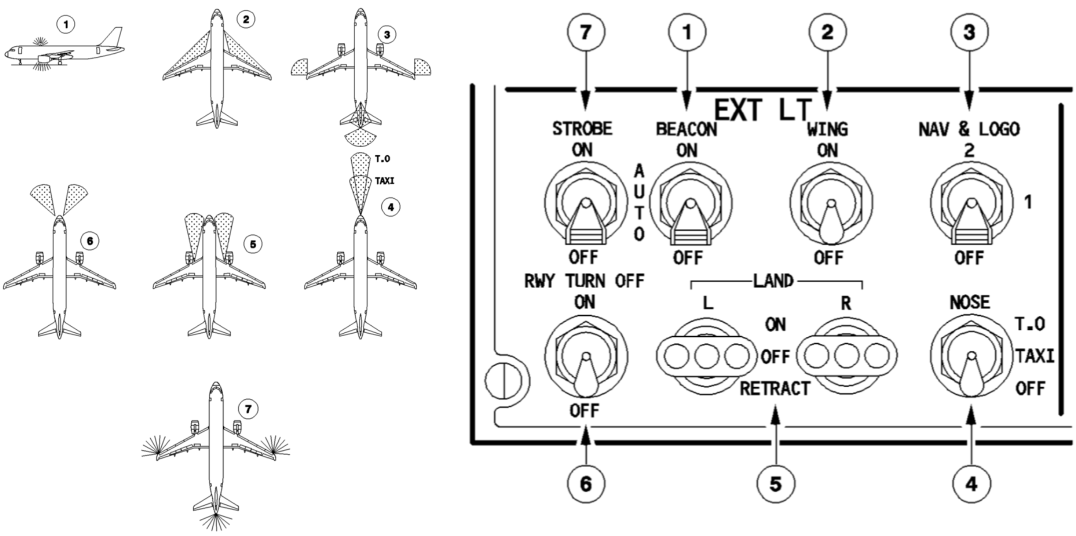


Figure 4.24: A320 Exterior lighting (left) and Exterior lighting panel (right) [9]

3.1.3.8 Signs panel

The Signs panel (figure 4.25) is used to activate signs inside the passenger cabin: the SEATBELTS switch activates (on ON) the FASTEN SEATBELTS and RETURN TO SEAT signs in the passenger cabin. The NO SMOKING switch activates the "NO SMOKING" and "EXIT" signs in the cabin on ON, while on AUTO (middle/neutral position) the same signs are activated only with the landing gear deployed. Finally, the EMER EXIT LT switch is set on ON to activate the "EXIT" signs, the overhead emergency lights and the floor light band system in the cabin; if instead is set to ARM, these signs are activated only in case of a failure of the aircraft electrical power system. The indicator besides the EMER EXIT LT switch shows an amber light in case the switch is set to OFF.

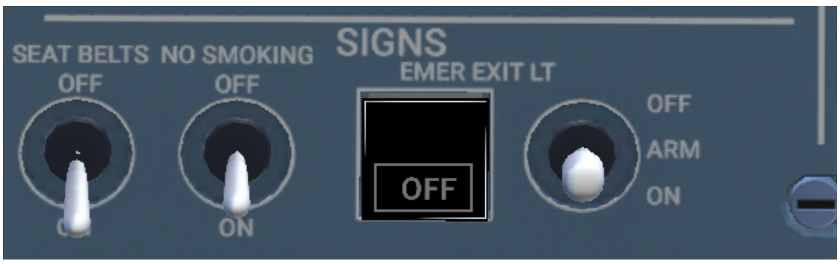


Figure 4.25: Signs panel

3.1.3.9 APU control panel

Figure 4.26: APU control panel

In this panel (figure 4.26), the MASTER SW button controls the supply of electrical power for APU operation (active on ON, blue light); the START button is used to begin the start sequence for the APU; when pushed is set to ON (blue light) until the APU has completed the start sequence and can supply electric power and bleed air to the aircraft's systems; when this happens, a green light lightens the AVAIL indicator on the button.

4.2 A320 Flight Simulator

The Flight Simulation Laboratory of Politecnico di Torino is equipped with an A320 Flight Simulator. This simulator uses as cockpit environment a 1:1 replica of the main Instrument Panel and Pedestal of an A320, with their main panels and components. The main instrument panel is equipped with master warning and caution warning lights, EFIS control panel, FCU, NDs, PFDs, E/WD, SD. The Pedestal is equipped with one MCDU (at the pilot flying side), speed brake and flap levers, parking brake handle and throttle levers. Notable elements missing are the complete overhead panel, and a functioning Engine panel on the pedestal, which means at least that any simulation must start with engines already active.
The hardware architecture of this simulator is presented in figure 4.27.
The primary computer hosts and runs the flight simulator software, X-Plane, with the A320 flight model. This PC handles the input from various hardware devices, for example the components of the blue line in figure 4.27, side-sticks, pedals, throttle and FCU, allowing pilots to interact with the virtual cockpit and control the aircraft systems, and provides outputs for the Navigation Displays, the Primary Flight Displays and for two projectors appropriately aligned, which are used to create the visual scenery of the flight simulator rendered by the PC in a wide field of view.
The secondary computer plays the role of the instructor station, where it is possible to introduce failures on the systems of the aircraft to evaluate the pilot's response to

an emergency situations; it also generates the ECAM (E/WD and SD), and allows the insertion of data necessary for programming the MCDU, which is managed by its own dedicated mini-computer. All the aforementioned computers are connected to allow data synchronization and communication (green line in figure 4.27).
There are four micro-controllers that allow the control of switches and the interface potentiometers with the levers and lights of the Pedestal and the main Instrument Panel.

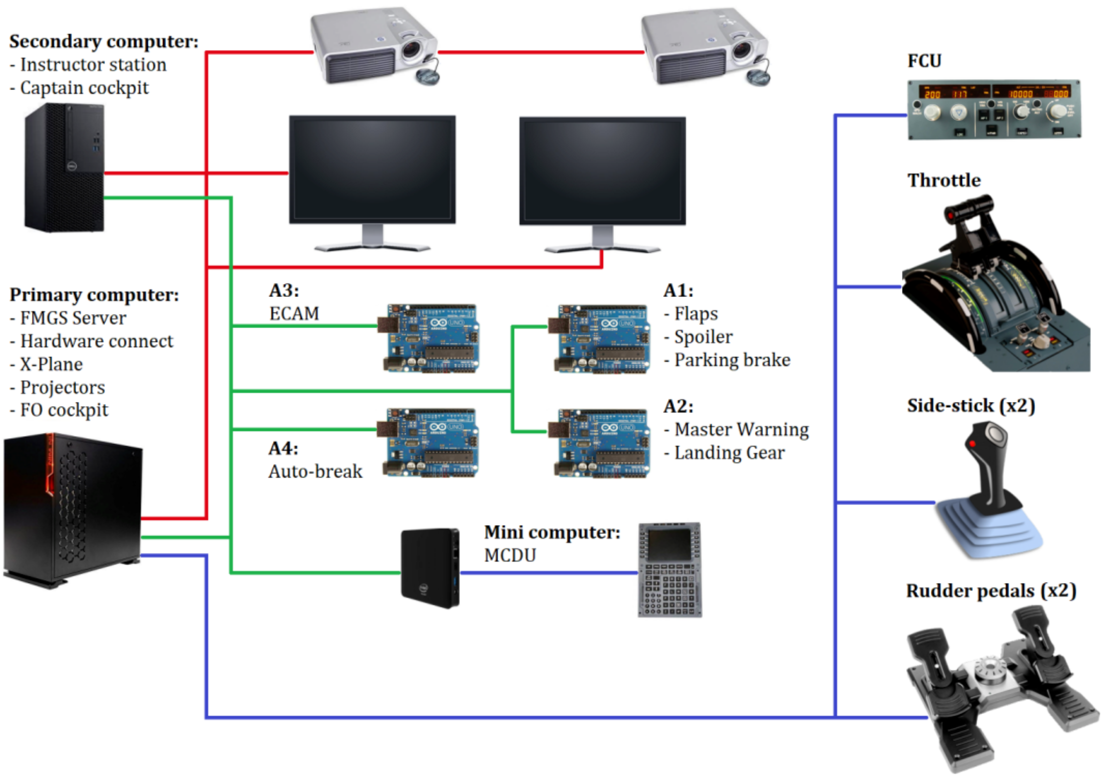


Figure 4.27: A320 flight simulator hardware (Flight Simulation Laboratory, Politecnico di Torino) [79]

While not directly relevant for the purpose of this thesis, the A320 flight simulator was used in previous similar thesis work [5] as a platform for evaluating the results of computer based training versus VR training in flight operations procedures; if or when a similar test will be performed again with the procedures described inside this thesis or others, it will be necessary a comparison between the capabilities and the limitations of the A320 flight simulator (lack of the overhead panel, lack of functioning engine panel), and the limits of both computer based training and what can be programmed in Unity/WEAVR environment in terms of VR training procedures.

4.3 A320 cockpit in Unity

The creation of VR training procedures required the setup of an appropriate scene in a Unity project, *"VT-S-V15"*. The starting point of this setup was importing a 3D model of an A320 cockpit provided by TXT, *"Cockpit_A320_Correct.unitypackage"*. This model (figure 4.28) improves on the models used in previous work of thesis, specifically adds panels and various items missing from the cockpit model used for [7], [8], [5] and [6], and corrects errors in the scale of components for the cockpit model used for a part of the project described in [3] and [4].

Figure 4.28: A320 cockpit in Unity (pilot seats hidden)

After having imported the 3D model of the cockpit and having installed WEAVR, it was possible to operate to a large number of Game Objects corresponding to buttons, switches and levers to apply the Unity Components described in table 3.3, usually with "2-Way Switch", "3-Way Switch" and "N-Way Switch" for switches, levers and rotary selector switches, and "Executable" and "Push button" for buttons in the cockpit. Additionally, the same Game Objects were equipped with a "Skeleton Poser" Component, in order to visualize on the scene a pointing index finger for interactions with buttons and a hand in a position to grab a cylindrical object for interactions with switches and similar Game Objects, instead of a hand holding a controller as per default visualization. Further additions to the scene were made, in order to make the scene more immersive (paragraph 4.3.1), to allow a easier navigation through the scene (paragraph 4.3.2) and to give an interface for the user inside the virtual scene to start different training procedures (paragraph 4.3.3).

4.3.1 Displays and buttons

It was deemed necessary to create a number of assets to complement the cockpit model, specifically images of different pages of some of the main displays: MCDU, Navigation Display and upper and lower ECAM. These images were produced with screenshots of these displays in X-Plane, and were used to represent the state of the belonging display after specific actions performed by the user in the virtual scene. The mentioned images were then loaded in the Unity project's local folder, and from Unity Project window (section 5 of figure 3.3) they were transformed in *Sprites*, 2D images that were then applied to Game Objects (one for each *Sprite*) with the Component "Sprite Renderer"; finally, these Game Objects were set as children in the Unity project's hierarchy to parent Game Objects associated with the displays.

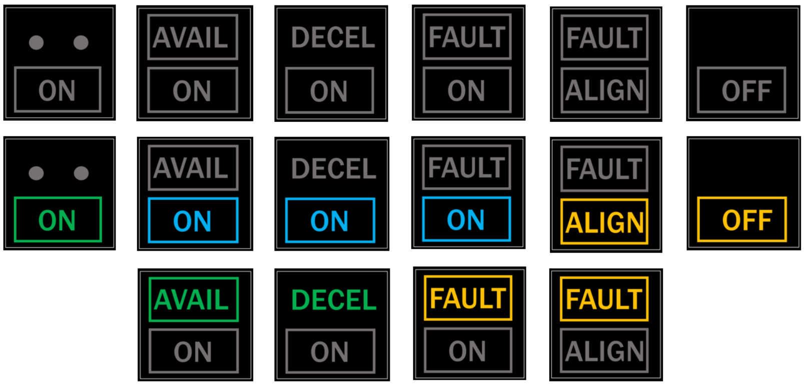


Figure 4.29: Main *Sprites* for the cockpit buttons and indicators

A similar activity was performed to create masks for numerous buttons on the scene; this was done with the purpose of creating assets to complement the buttons and switches behaviours with a visually immersive addition: if a button is pushed or the position of a switch is changed, than in the A320 real cockpit a light on the button or on a nearby indicator is activated. To replicate this behaviour, *Sprites* were assigned to multiple Game Objects, as many as necessary to represent all the states of the lights on the buttons; these Game Objects were then assigned to specific Game Objects of buttons and indicators. Some of the masks produced for the buttons on the scene are displayed in figure 4.29, while the displays different pages are shown in the next chapter.
The activation of a specific Game Object representing the state of a button or a display page that changes when the user interact with elements on the scene was programmed in two ways, either through the use of the "Executable"/"Push Button" Component on a button (which can be used to activate and deactivate Game Objects when the object is interacted with) or through the use of the WEAVR

Action "Toggle Object"; in the latter case, only during the execution of a procedure in a specific step the behaviour of the display or the button is consistent with the interactions of the player with the scene.

4.3.2 Teleport functions

Using the SteamVR prefabs described in paragraph 3.2.3, teleport points and areas were established in the pilot flying (left) position, pilot monitoring (right) position, and behind the pilots seats (figure 4.30). Additionally, the pilot flying position was used to define a spawn point for the execution of the VR application built in Unity.

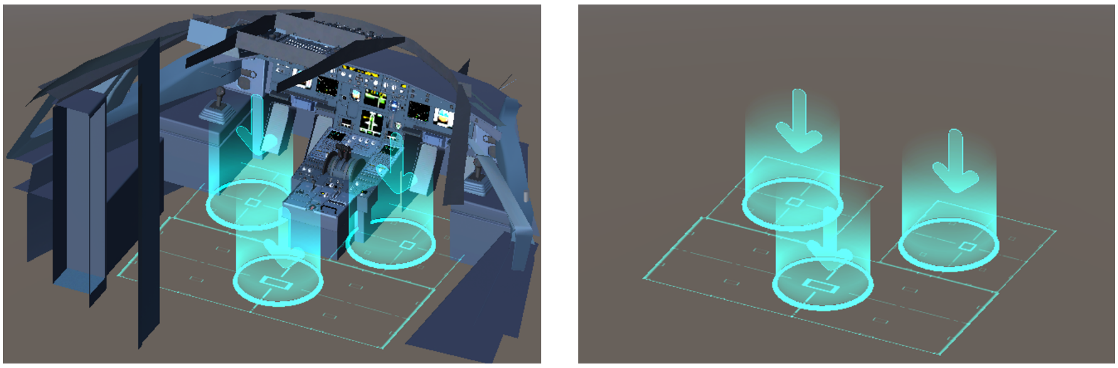


Figure 4.30: Teleport area and points

4.3.3 Tablet for procedures selection

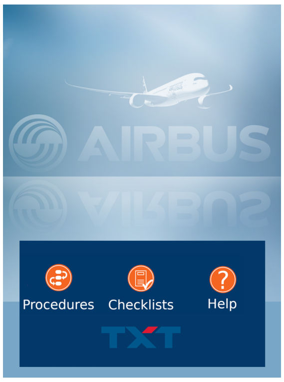


Figure 4.31: Tablet **Home** page

An additional Game Object was used to control the execution of the procedures described in the next chapter. A 3D model of a tablet was initially provided by TXT for the projects of previous thesis; the same Game Object was imported in *"VT-S-V15"*. The graphics of all pages were revamped, but the functions of different pages and buttons were kept consistent with previous thesis.

A **Home** page is used to access three more pages, **Procedures**, **Checklists** and **Help** (figure 4.32).

- **Help**: this page provides a general description of the main functions of the tablet;
- **Checklists**: this page shows a checklist for the A320 flight normal procedures;
- **Procedures**: this page allows the user a first choice between which kind of procedure he wants to execute, between "Normal" and "Abnormal & Emergency" procedures.

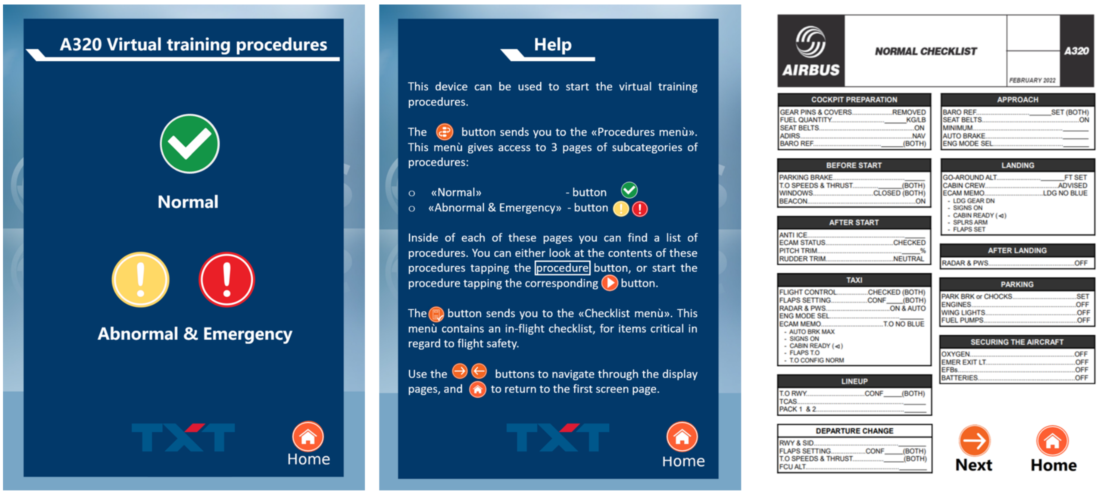


Figure 4.32: Tablet **Procedures**, **Help** and **Checklists** pages (left to right)

The **Normal procedures** pages 1 and 2 (figure 4.33) can be used to start a normal procedure (pushing the right orange button) and open the dedicated page with a list of the main steps to be taken to complete the procedure, or to see the same page for a specific procedure (pushing the cell corresponding to the procedure the user is interested in). When a step of a procedure is executed, a green check mark appears to the left side of the step; when a procedure is completed, a green check mark appears to the left side of the cell corresponding to the procedure in the **Normal procedures** page, and if said procedure has a reference checklist, a green check mark appears on the **Checklists** page for the completed procedure.
The navigation through the pages described is made possible by the application of an "Executable" Component to all the buttons of all the pages; this "Executable" is programmed to activate when the button is puhed the Game Object corresponding to another page of the tablet, with its children Game Objects (mainly buttons), and deactivate the parent page Game Object (and therefore deactivate itself). The

system of check marks appearing on the execution of specific procedure steps or the completion of procedures is entirely governed by Actions defined with the WEAVR Procedure described in the next chapter.

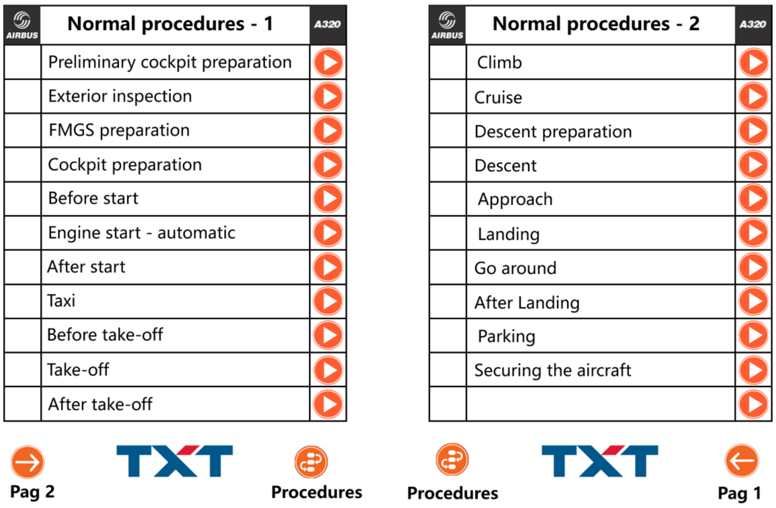


Figure 4.33: Tablet **Normal procedures** pages 1 and 2

Chapter 5

A320 Flight Operations Procedures & VR Training

In flight operations, Standard Operating Procedures (SOPs) are a set of procedures defined by an aircraft manufacturer or an airline which supports pilots in operating an aircraft safely and consistently, providing step-by-step instructions for how to carry out tasks and operations in a specific order. The application of SOPs ensures that the same tasks are carried on in the same way consistently by different people, therefore allowing pilots who may have not flown with each other before to fly together as a crew [1]. To summarise, the purpose of SOPs is to:

- Promote adherence to the manufacturer's and airline's operating philosophy;
- Promote operational safety and efficiency;
- Standardize the crew behaviour in dealing with a broad range of airline operations and operating environments.

The broad range of operations and operating environments that an aircraft could face must be addressed with SOPs that cover normal operations and abnormal & emergency situations, defined as follows [2]:

- An abnormal situation is one in which the safety of the aircraft or people on board or on the ground is not in danger, but the normal procedures cannot be used to continue the flight.

1. https://www.flightdeckfriend.com/ask-a-pilot/what-is-an-sop/
2. https://www.skybrary.aero/articles/emergency-or-abnormal-situation

- An emergency situation is one in which the safety of the aircraft or of the people on board or on the ground is endangered for any reason.

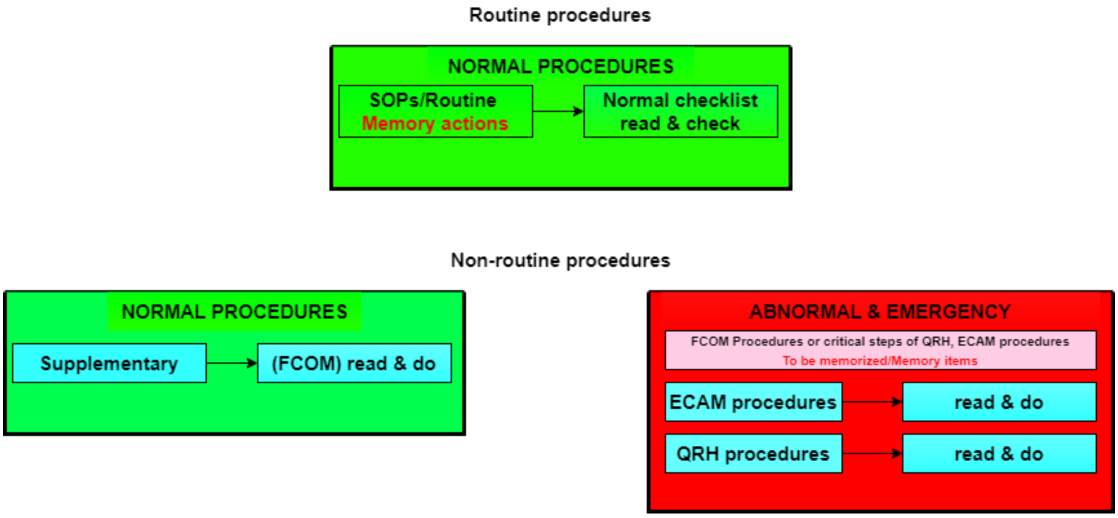


Figure 5.1: Procedures subdivision as described in A320 FCOM

The A320 flight crew techniques/training manual (FCTM) [80] uses the term SOPs referred to just a subset of normal procedures, namely the ones that are performed frequently in the daily normal operations of the aircraft. As described in the A320 FCTM, routine normal procedures are usually designed considering the following factors:

- The procedure must correspond to a specific flight phase, and the two must have an univocal relationship;
- The steps of the procedure must be described in chronological order;
- The steps of the procedure must be easy to memorize.

The application of these procedures presumes that all systems operate normally and all automatic functions can be used normally. Some of these procedures are checked against checklists, provided by the airline or the aircraft manufacturer. More specifically, Airbus normal checklist includes specifically the actions that can affect flight safety and efficiency. These checklists are of a "non-action" type (i.e. all actions should be completed from memory before the flight crew uses the checklist to check on the actions performed).
The non-routine normal procedures or supplementary procedures are operations that are performed during the daily normal operations of the aircraft but not

frequently (e.g. manual engine start). These procedures should be performed by the flight crew following the principle read & do, reading the steps of the procedures in the A320 flight crew operating manual (FCOM) [9].
An abnormal & emergency procedure is initiated usually following a system failure or in case of a dangerous operational context. These procedures should be performed by the crew either as memory items or read & do, depending on if the flight crew has or not the time to refer to the ECAM, or the quick reference book (QRH), or the FCOM.
The subdivision of procedures in routine and non-routine, normal and non-normal is summed up with figure 5.1.

5.1 VR training procedures

This thesis primary focus is on the implementation in Unity/Pacelab WEAVR environment of some of the routine normal procedures, following the reference [9], the latest (dated 2022) Airbus checklist for normal routine procedures (figure 4.32[3], for simulation purposes only), and two documents available online, describing the procedures flow of operations with the supplement of pictures [4] and providing a list similar to the one provided by the FCOM for each one of the single operations inside a procedure [5].
The procedures implemented for this thesis are:

- **FMGS preparation**;
- **Cockpit preparation**;
- **Before start**;
- **Engine start - automatic**;
- **After start**;
- **Taxi**.

All of these procedures have a dedicated checklist except for **Engine start - automatic** and **FMGS preparation**; according to the FCOM the latter is part of the **Cockpit preparation** procedure, but for this thesis **FMGS preparation**

3. https://x-plane.to/file/44/airbus-a32x-checklist-2022

4. https://www.theairlinepilots.com/forumarchive/a320/a320-normal-procedures.pdf

5. https://docs.flybywiresim.com/pilots-corner/SOP/

was separated from it because of its length and limited interactions with the A320 cabin. In previous similar thesis [8][6][5][7][3][4] were implemented the normal routine procedures **After start**, **Before start** and **Cockpit preparation**. From the non-routine normal procedures the **Engine start - manual** procedure was implemented, which is not much different from the **Engine start - automatic** procedure implemented for this thesis in terms of active interactions that the user must have with the virtual environment. For the procedures implemented in this thesis, as for the procedures implemented in the previous references, it must be noted that these are not completely accurate compared to the ones described in [9]; this is due both to limitations of the virtual environment and to a choice to provide to the player only the most meaningful interactions with the virtual environment for each procedure implemented.
In terms of WEAVR tools, the procedures presented in the following paragraphs are all part of a single procedure or *.asset* file. This means that they all have a starting point in common, and they are built in a way to allow the user to carry out all procedures in a single training session; they can also be performed in any order, and the same procedure can be performed more than once, since the procedures and the Unity scene are set in a way to reset the changes to the scene produced by the user's interactions with it. In terms of user experience, all steps of all the procedures are guided by WEAVR actions, specifically "Show Billboard Expert", "Outline " and "Text To Speech" (table 3.4); when combined, these actions can improve the user experience providing multiple visual and audio guidance to the completion of each task of a procedure.

5.1.1 Starting the VR procedure

To start the WEAVR procedure, users must push the *Play* button in the Unity Editor's upper toolbar (figure 3.4); the same button can be pushed to stop the execution of the procedure.
To start any of the SOPs virtual training, users must interact with the tablet in the virtual environment at least 3 times: from the **Home** page (figure 4.31), they must push the "Procedures" icon to access the **Procedures** (figure 4.32 on the left), then they must select the "Normal" icon to access the **Normal Procedures** pages (figure 4.33). From this point onward, any of the normal procedures listed in the previous section can be started, by pushing the orange and white "Play" icon at the side of the respective procedure. In figure 5.2 is shown on the left the interaction with the "Normal" icon of the **Procedures** page, and on the right the interaction with the **Normal Procedures** first page to start the **Taxi** procedure.

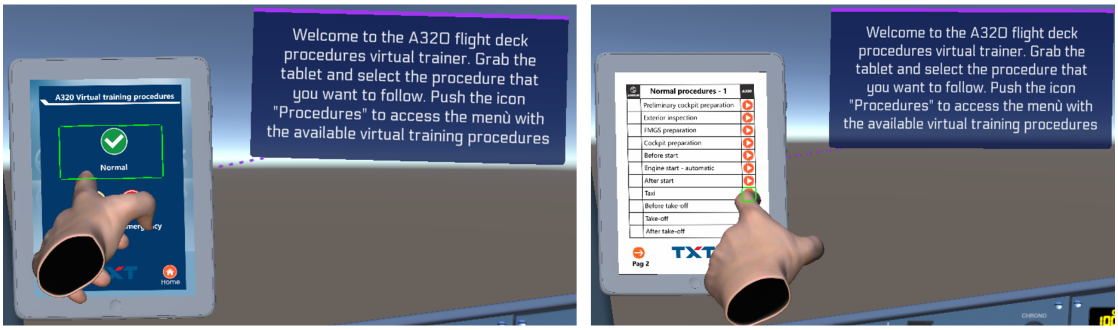


Figure 5.2: Interactions with the tablet to start an SOP virtual training: **Procedures** page (left) and **Normal Procedures** first page (right)

When a procedure ends, its ending node is linked to node 2; this is the only node from which a procedure can be started, because it is only in this node where the "Play" icons for all the available procedure procedures function as button-type objects that when executed or pushed start a procedure; their behaviour is governed by an "Executable" component assigned to each of them, activated at the start of node 2 through a "Toggle Component" action, used as Exit Condition from the node and deactivated though another "Toggle Component" action.

Node	Object	Exit Condition	Notes
1	-	-	Starting node of the WEAVR procedure, creates the billboard shown in figure 5.2
2	"Play" icons for all the available procedures in the **Normal Procedures** pages	Has Executed	Each "Play" icon defines a different Exit Condition, which links with different nodes (starting nodes for every different SOP)

Table 5.1: WEAVR procedure: starting Nodes and Exit Conditions

5.1.2 FMGS preparation

The **FMGS preparation** procedure is based on a number of interactions limited to the MCDU. As proposed by [9], the pages with which the user must interact to complete the FMGS preparation are, in chronological order:

- **D**ATA;
- **I**NIT-A;
- **F**-PLN;
- **S**EC F-PLN;
- **R**AD-NAV;
- **I**NIT-B;
- **P**ERF.

Specifically, the page accessible from the DATA menu page is A/C STATUS, which is used to check on engine and aircraft type and navigation database validity. The INIT-A page is used to set data like airport of origin and destination, flight number, cost index, cruise flight level and temperature; it is also used to complete the IRS alignment.

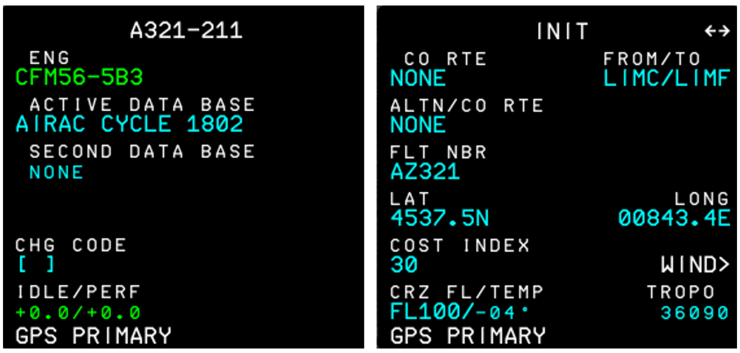


Figure 5.3: MCDU pages - DATA A/C STATUS (left) and INIT-A (right)

The F-PLN and SEC F-PLN are used to define a main route from an origin to a destination airport and an alternative. The RAD-NAV page is used to verify identifiers and frequencies of radio NAVAIDs, such as VOR, VOR/DME, ILS, and ADF. The INIT-B page is used to insert data like zero fuel weight (ZFW), zero fuel weight center of gravity (ZFWCG) and block fuel.

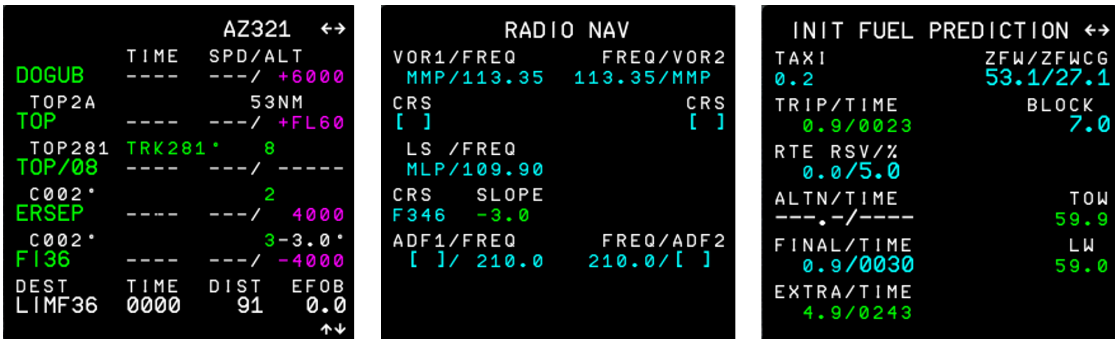


Figure 5.4: MCDU pages - F-PLAN (left), RAD-NAV (middle) and INIT-B (right)

Finally, the PERF page allows the user to set performance figures and parameters that affect the aircraft performance related to different phases of flight, such as take-off and approach.

Figure 5.5: MCDU pages - PERF TAKE OFF (left) and PERF APPROACH (right)

The **FMGS preparation** procedure implemented for this thesis considers active interactions with the MCDU for the INIT-A (nodes 6.3 to 11.3), F-PLN (nodes 12.3 to 27.3), INIT-B (nodes 30.3 to 32.3) and PERF (nodes 33.3 to 46.3) pages, simple visual inspections for DATA (node 5.3) and RAD-NAV (node 28.3) pages and does not set a secondary flight plan with the SEC F-PLN page. The active interactions consist in using the line selector keys left/right (LSKL/LSKR), the slew keys and the MCDU control keys to navigate through the MCDU's pages and insert data. To avoid an unnecessary lenghty and unpractical procedure, the necessity of writing something with the MCDU's alpha-numeric keyboard is completely removed. Billboards are used to tell users which buttons of the MCDU push for advancing the procedure step by step; in these billboards the information that should be entered into the MCDU normally by typing on the alpha-numeric keyboard and inputting through the line selector keys is also provided.

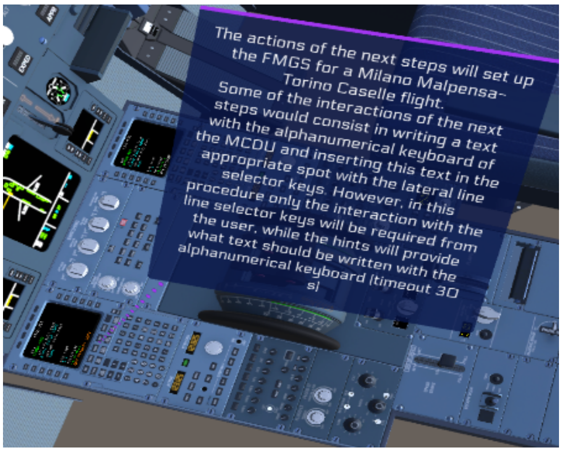


Figure 5.6: FMGS Preparation: node 4.3

In the virtual environment designed for this thesis, users do not type on the alpha-numeric keyboard, they just input them using line selector keys. This information is provided to the user in the node 4.3 of the procedure (figure 5.6). The data inserted in the MCDU during this procedure sets up a flight plan from Milano Malpensa to Torino Caselle airports, using the data used to program the MCDU in X-Plane in previous similar work of thesis, [3] and [4].

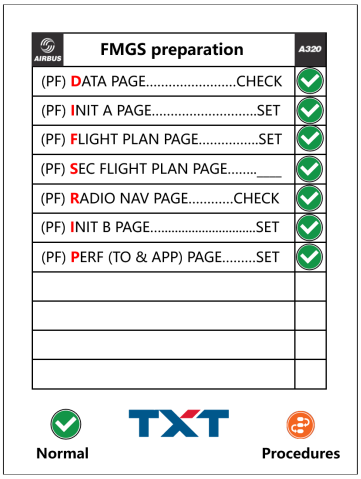


Figure 5.7: FMGS Preparation: "FMGS Preparation" page of the tablet when the procedure is complete

When all the previously mentioned pages of the MCDU have been compiled, the tablet in the virtual environment looks like in figure 5.7; at this point, the user can start another procedure by simply leaving the "FMGS Preparation" page of the tablet (node 47.3).

Node	Object	Exit Condition	Notes
3.3	-	-	Scene preparation
4.3	-	-	Billboard in figure 5.6
5.3	MCDU display	Visually inspect	Inspects the A/C STATUS page
6.3	INIT button	Has Executed	Opens the INIT-A page
7.3	LSKR1 button	Has Executed	Sets the ICAO codes for airports of origin and destination
8.3	LSKL3 button	Has Executed	Sets the flight number
9.3	LSKL4 button	Has Executed	Sets the Cost Index
10.3	LSKL5 button	Has Executed	Sets the cruise flight level and temperature
11.3	LSKR3 button	Has Executed	Complete the alignment procedure of the IRS
12.3	F-PLN button	Has Executed	Opens the F-PLAN page
13.3	LSKL1 button	Has Executed	-
14.3	LSKL1 button	Has Executed	The button is pushed twice to open a page where departure informations can be set
15.3	LSKL5 button	Has Executed	Sets the departure runway
16.3	LSKL3 button	Has Executed	Sets the departure route (SID)
17.3	LSKL6 button	Has Executed	Confirm the departure data
18.3	LSKL6 button	Has Executed	-

Table 5.2: FMGS Preparation procedure Nodes and Exit Conditions (part 1)

Node	Object	Exit Condition	Notes
19.3	LSKR1 button	Has Executed	This button and the previous one are pushed to open a page where departure informations can be set
20.3	LSKL5 button	Has Executed	Sets the instrument approach procedure
21.3	Slew key up	Has Executed	Navigate through the available arrival routes (STAR)
22.3	LSKL5 button	Has Executed	Sets the arrival route
23.3	LSKL4 button	Has Executed	Sets the arrival VIA
24.3	LSKR6 button	Has Executed	Confirms the arrival data
25.3	Slew key up	Has Executed	Navigate through the F-PLAN page to find a discontinuity
26.3	LSKL2 button	Has Executed	Deletes the discontinuity
27.3	LSKR6 button	Has Executed	Confirms the flight plan
28.3	RAD-NAV button	Has Executed	Opens the RAD-NAV page
29.3	INIT button	Has Executed	Opens the INIT-A page
30.3	Slew key left	Has Executed	Opens the INIT-B page
31.3	LSKR1 button	Has Executed	Sets the ZFW and the ZFWCG
32.3	LSKR2 button	Has Executed	Sets the block fuel
33.3	PERF button	Has Executed	Opens the PERF - TAKE OFF page

Table 5.3: FMGS Preparation procedure Nodes and Exit Conditions (part 2)

Node	Object	Exit Condition	Notes
34.3	LSKR3 button	Has Executed	Sets the flaps vonfiguration to be used for take-off
35.3	LSKL1 button	Has Executed	Sets the V_1 speed
36.3	LSKL2 button	Has Executed	Sets the V_R speed
37.3	LSKL3 button	Has Executed	Sets the V_2 speed
38.3	LSKR4 button	Has Executed	Sets the flex temperature
39.3	LSKR6 button	Has Executed	Opens the PERF - CLB (climb phase) page
40.3	LSKR6 button	Has Executed	Opens the PERF - CRZ (cruise phase) page
41.3	LSKR6 button	Has Executed	Opens the PERF - DES (descent phase) page
42.3	LSKR6 button	Has Executed	Opens the PERF - APPR (approach phase) page
43.3	LSKL1 button	Has Executed	Sets the QNH air pressure
44.3	LSKL2 button	Has Executed	Sets the air temperature
45.3	LSKL3	Has Executed	Sets the wind magnitude and direction
46.3	LSKR3 button	Has Executed	Sets the decision height
47.3	"FMGS Preparation" page (figure 5.7)	Is Not Active	Resets the virtual scene

Table 5.4: FMGS Preparation procedure Nodes and Exit Conditions (part 3)

5.1.3 Cockpit Preparation

The **Cockpit Preparation** procedure begins by asking the user to test the functionality of the cockpit voice recorder. To do so, he must push in the Voice Recorder panel the GND CTL button (node 4.4, figure 5.8 on the left) and then push and hold the CVR TEST button (node 5.4, figure 5.8 on the right); this action is programmed to trigger a looping sound in the virtual environment, to mimick the behaviour of the button in the real A320 cockpit.

Figure 5.8: Cockpit Preparation: node 4.4 (left) and node 4.5 (right)

After these actions, the user must turn all 3 of the IR mode selectors to NAV in the ADIRS panel to start the IR system alignment (node 6.4, figure 5.9 on the left). After that, the user must switch to AUTO the STROBE light switch in the Exterior Lighting panel (node 7.4, figure 5.9 on the left).

Figure 5.9: Cockpit Preparation: node 6.4 (left) and node 7.4 (right)

At this point, the user is asked to interact with the Signs panel to set the SEAT-BELTS LT switch ON (node 8.4, figure 5.10 on the left), the NO SMOKING LT switch to AUTO (node 9.4, figure 5.10 on the middle), and the EMER EXIT LT switch to ARM (node 10.4, figure 5.10 on the right).

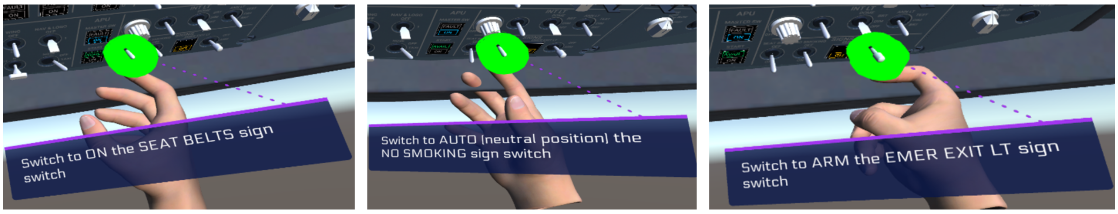


Figure 5.10: Cockpit Preparation: node 8.4 (left), node 9.4 (middle) and node 10.4 (right)

Subsequently, the user must push and hold the ENGINE 1 FIRE TEST (node 11.4, figure 5.11 on the left) and the ENGINE 2 FIRE TEST (node 12.4) buttons in the Fire Control panel, in this order; these actions trigger several changes to the virtual scene: while the buttons are pushed, the ECAM displays a warning memo related to an engine fire (figure 5.11 on the right), a repetitive chime sound can be heard, the AGENT buttons lighten up, like the FIRE indicators on the Engine panel on the pedestal and the MASTER WARNING indicators on the glare-shield.

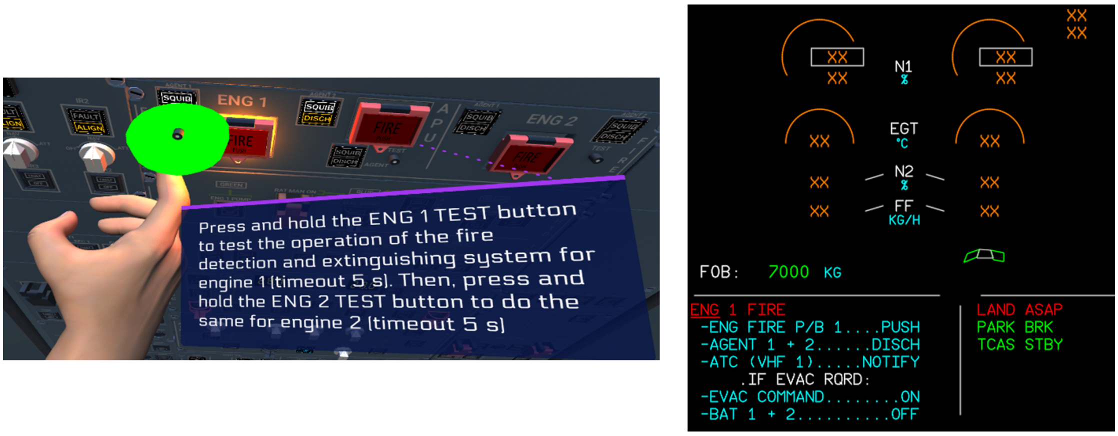


Figure 5.11: Cockpit Preparation: node 11.4 (left), and upper ECAM with the warning memo about the engine fire (right)

At this point a step is dedicated to the FMGS preparation, where the user is asked to visually inspect the MCDU (node 13.4) to "complete" this phase; a more in-depth focus on FMGS preparation is provided by the procedure described in the previous section.

As second to last step, the user is asked to check on the MCDU that the IR systems are set to NAV and their measures are coherent with the GPS; to do so, the user is asked to navigate through two pages of the MCDU, the DATA page and the DATA - POSITION MONITOR by pushing the DATA button (node 14.4, figure 5.12 on the left), then the LSKL1 button (node 15.4, figure 5.12 on the right); after that the user must visually inspect the DATA - POSITION MONITOR page (16.4).

Figure 5.12: Cockpit Preparation: node 14.4 (left) and node 15.4 (right)

The DATA page and DATA-POSITION monitor pages are shown with the figure 5.13 below.

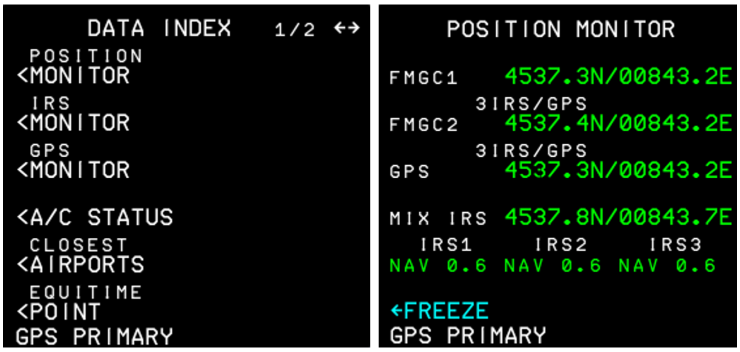


Figure 5.13: Cockpit Preparation: MCDU DATA page (left) and DATA - POSITION MONITOR page (right)

Finally, the user is asked to visually inspect the Barometer Reference Display Window in the left EFIS Control panel (node 17.4, figure 5.14 on the left). A step is then dedicated to the visual inspection of the right Barometer Reference Display Window performed by the pilot monitoring; the user must only wait 15 seconds (node 18.4, figure 5.14 on the right).

Figure 5.14: Cockpit Preparation: node 17.4 (left) and node 18.4 (right)

At this point, the user is asked to check on the **Checklists** page of the tablet navigating through the available pages of the tablet (node 19.4), and once opened they must wait 20 second before the navigation buttons of the page are active; subsequently, they can start another procedure from the **Normal procedure** pages of the tablet (node 20.4).

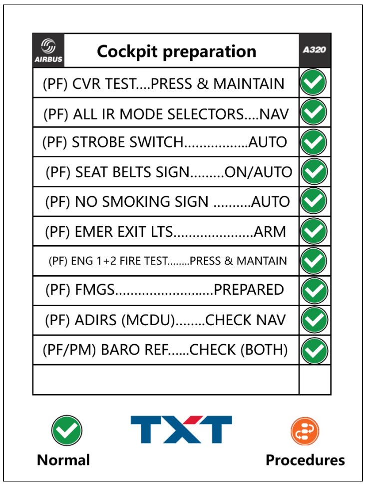


Figure 5.15: Cockpit Preparation: "Cockpit Preparation" page of the tablet when the procedure is complete

Node	Object	Exit Condition	Notes
3.4	-	-	Scene preparation
4.4	GND CTL button	Current state = down (Push Button)	Figure 5.8 (left)
5.4	CVR TEST button	Is Playing (Push Button)	Figure 5.8 (right)
6.4	IR mode selectors 1, 2, 3	Current state = middle (3-Way Switch)	Figure 5.9 (left)
7.4	STROBE lights switch	Current state = middle (3-Way Switch)	Figure 5.9 (right)
8.4	SEATBELTS lights switch	Current state = down (2-Way Switch)	Figure 5.10 (left)
9.4	NO SMOKING lights switch	Current state = middle (3-Way Switch)	Figure 5.10 (middle)
10.4	EMERGENCY EXIT lights switch	Current state = middle (3-Way Switch)	Figure 5.10 (right)
11.4	ENGINE 1 FIRE TEST button	Is Playing (Push Button)	Figure 5.11 (left)
12.4	ENGINE 2 FIRE TEST button	Is Playing (Push Button)	-
13.4	MCDU Display	Visually Inspect	-
14.4	DATA button	Has Executed	Figure 5.12 (left)
15.4	LSKL1 button	Has Executed	Figure 5.12 (right)
16.4	MCDU Display	Visually Inspect	Check the DATA - POSITION MONITOR page
17.4	PF Barometer Reference Display Window	Visually inspect	Figure 5.14 (left)

Table 5.5: Cockpit Preparation procedure Nodes and Exit Conditions (part 1)

Node	Object	Exit Condition	Notes
18.4	PM Barometer Reference Display Window	-	(PM action) figure 5.14 (right)
19.4	**Checklists** page	Is Active	-
20.4	**Checklists** page	Is Not Active	Resets the virtual scene

Table 5.6: Cockpit Preparation procedure Nodes and Exit Conditions (part 2)

5.1.4 Before Start

The **Before Start** procedure starts with the task for the user to set the BEACON lights switch to ON (node 4.5, figure 5.16 on the left); after that, the user is asked to visually inspect the left window operating handle (node 5.5, figure 5.16 on the right).

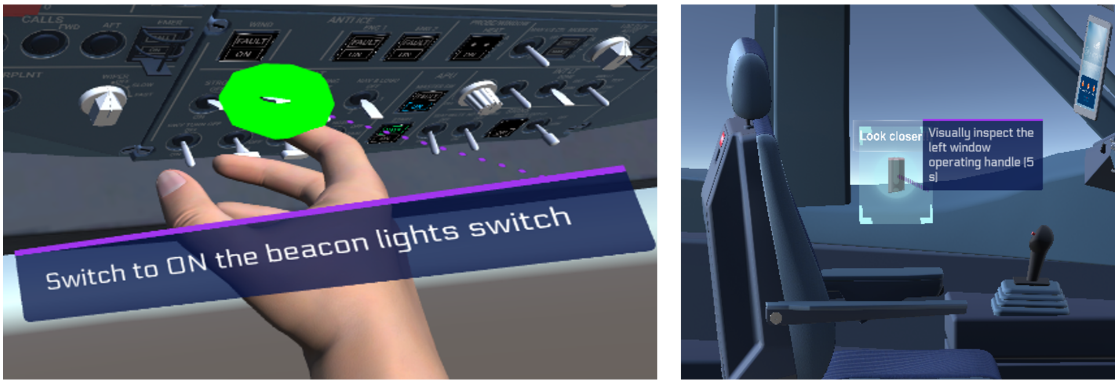


Figure 5.16: Before Start: node 4.5 (left) and node 5.5 (right)

After that, the user is asked to push the DOOR button in the ECAM control panel (node 6.5, 5.17 on the left), and then to visually inspect (node 7.5) the lower ECAM display opened on the DOOR page (figure 5.17 on the right), to check that the doors of the cabin are all closed and armed (green boxes).

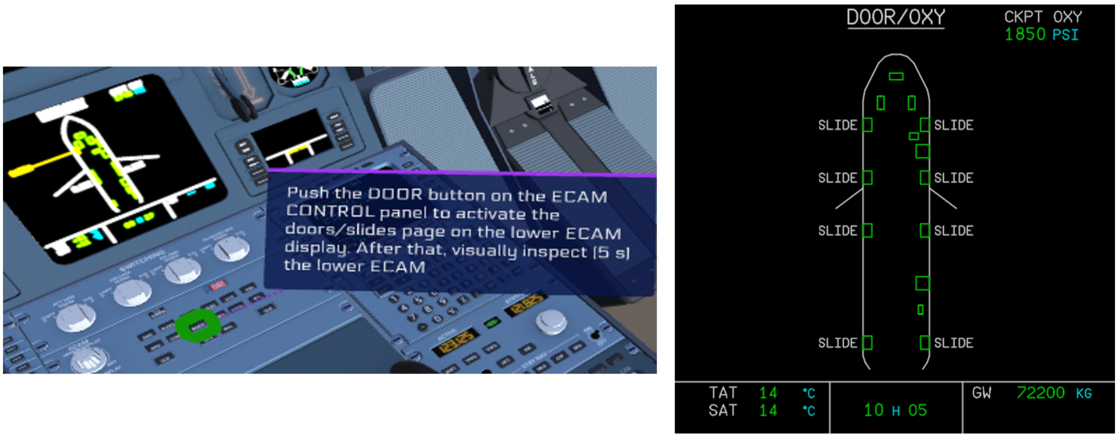


Figure 5.17: Before Start: node 6.5 (left) and lower ECAM DOOR page (right)

At this point the user is asked to visually inspect the engine thrust levers (node 8.5, figure 5.18 on the left) on the pedestal to check that they are set to the IDLE position, and after that to visually inspect the parking brake (node 9.5, figure 5.18 on the right) to check that is set in the ON position.

Figure 5.18: Before Start: node 8.5 (left) and node 9.5 (right)

The user is then asked to visually inspect the ATC panel (node 10.5, figure 5.19 on the left) to check that is set for operations. As final nodes, one node is dedicated to the pilot monitoring's visual inspection of the right window operating handle (node 11.5, figure 5.19 on the right).

Figure 5.19: Before Start: node 10.5 (left) and node 11.5 (right)

Another node is dedicated to the pilot monitoring's visual inspection of the lower ECAM DOOR page (node 12.5, figure 5.20). For both nodes 11.5 and 12.5 the user must wait 10 seconds to proceed to the next step

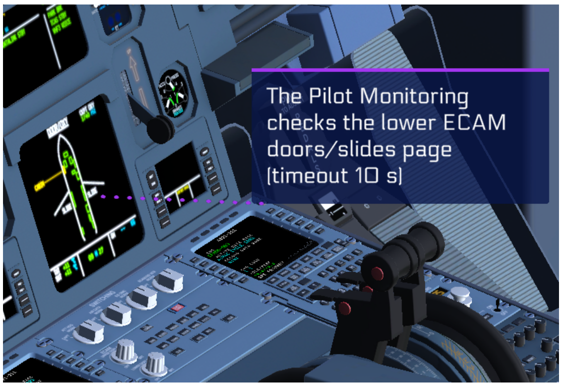
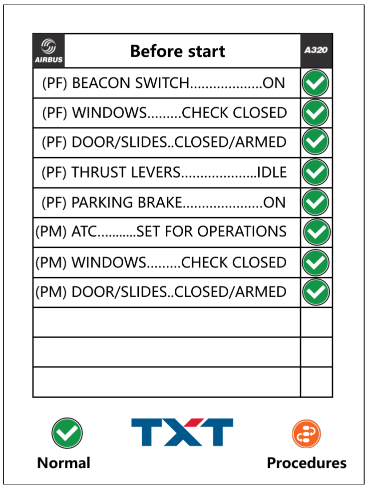


Figure 5.20: Before Start: node 12.5

At node 13.5 the procedure is complete, as shown in the "Before Start" tablet page in figure 5.21. To start another procedure, the user must first open the **Checklists** page on the tablet and wait 20 seconds; after this time, the user is able to navigate through the tablet's pages and eventually select another procedure to be performed.



AIRBUS Before start	A320
(PF) BEACON SWITCH...................ON	
(PF) WINDOWS.........CHECK CLOSED	
(PF) DOOR/SLIDES..CLOSED/ARMED	
(PF) THRUST LEVERS....................IDLE	
(PF) PARKING BRAKE.....................ON	
(PM) ATC...........SET FOR OPERATIONS	
(PM) WINDOWS.........CHECK CLOSED	
(PM) DOOR/SLIDES..CLOSED/ARMED	

Figure 5.21: Before Start: "Before Start" page of the tablet when the procedure is complete

Node	Object	Exit Condition	Notes
3.5	-	-	Scene preparation
4.5	BEACON lights switch	Current state = down (2-Way Switch)	Figure 5.16 (left)
5.5	Left window operating handle	Visually Inspect	Figure 5.17 (right)
6.5	DOOR button	Has Executed	Figure 5.17 (left)
7.5	Lower ECAM (DOOR page) Display	Visually Inspect	Figure 5.17 (right)
8.5	Engine thrust levers	Visually Inspect	Figure 5.18 (left)
9.5	Parking brake switch	Visually Inspect	Figure 5.18 (right)
10.5	ATC panel	Visually Inspect	Figure 5.19 (left)
11.5	Right window operating handle	-	(PM action) figure 5.19 (right)
12.5	Lower ECAM Display	-	(PM action) figure 5.20
13.5	**Checklists** page	Is Active	-
14.5	**Checklists** page	Is Not Active	Resets the virtual scene

Table 5.7: Before Start procedure Nodes and Exit Conditions

5.1.5 Engine Start - automatic

The **Engine start - automatic** procedure allows the user to simulate in the virtual environment the steps necessary to start both engines of the A320; for this thesis the turbofan engine used as reference is the high bypass-ratio CFM56-5B.
The procedure starts by asking the user to set the engine mode selector to the START position (step 4.6, figure 5.22 on the left) and then to set the ENGINE 2 MASTER switch to ON (step 5.6, figure 5.22 on the right).

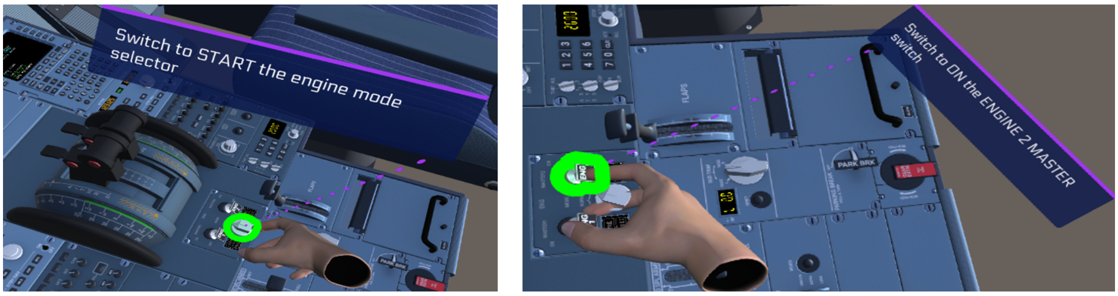


Figure 5.22: Engine start - automatic: node 4.6 (left) and node 5.6 (right)

In an A320 the latter action communicates to the FADEC of the engine to begin the automatic start sequence. To start each of the turbofan engines, a source of high pressure air is needed, for example the APU. When the ENGINE 2 MASTER switch is set to ON, the FADEC of the engine opens the start valve, which provides pressurized air to the accessory gearbox; this is used to feed an air starter unit, which is a turbine linked to a a shaft connected to the N2/high pressure shaft of the turbofan. The turbine of the air starter rotates thanks to the air provided to the air flow received from the start valve, and makes the high-pressure shaft rotate.
When N2 reaches a certain speed, the igniters are activated inside the combustion chamber of the engine; at higher speeds, a fuel flow (FF) is introduced to create an adequate fuel/air mixture; the temperature of the exhaust gas (EGT) exiting the turbine group starts increasing. Fuel flow and igniters are also controlled by the FADEC. The N2 shaft accelerates further, and slowly the N1/low pressure shaft starts to rotate, which causes more air suction, which increases N1 and N2 speeds further ; at a certain N2 speed, the ignition is cut-off, since the flame in the combustion chamber is self-sustained. At even higher N2 speeds, the air starter shaft is disengaged from the N2 shaft and the start valve is closed; after a little time the engine reaches the idle condition.
For the turbofan engine CFM56-5B, these steps of the starting sequence are associated to the following N2 speed values:

- N2 = 16 %: ignition is fired;
- N2 = 22 %: FF is introduced, EGT increases, N1 increases;

- N2 = 50 %: igniters are shut off, start valve is closed;
- N2 = 60 %: the engine has reached the idle condition, the starting sequence for the engine has finished.

The state of the start valve and the ignition can be assessed from the System display, which opens the ENGINE page when an ENGINE MASTER switch is set to ON. The N2 and N1 speeds, the EGT and the FF are shown to the pilot in the upper ECAM. The **Engine start - automatic** procedure uses the steps from 6.6 to 10.6 to show in the virtual environment the steps of the starting sequence for engine 2, giving users the tasks of visually inspecting the lower and upper ECAM to verify the progress of the sequence at increasing values of N2 speed.

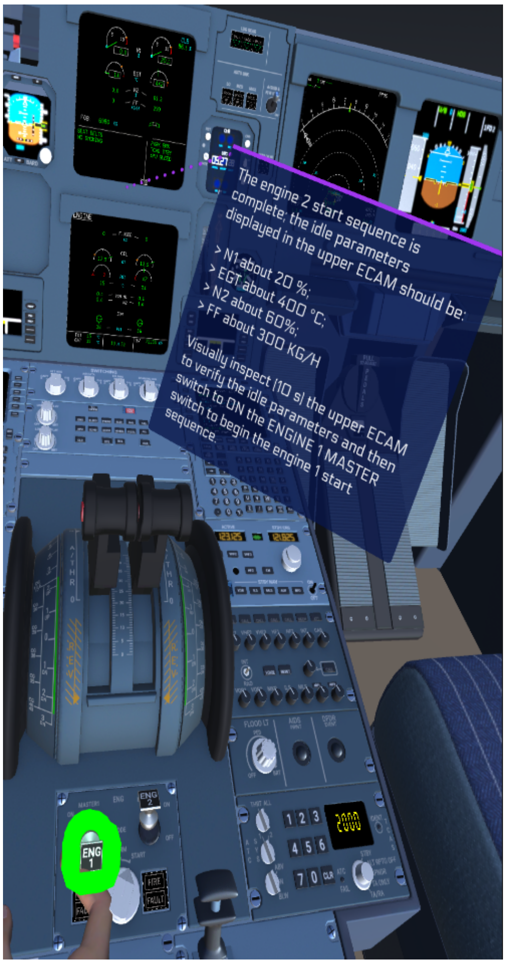


Figure 5.23: Engine start - automatic: node 11.6

At step 11.6, engine 2 has reached the idle condition (figure 5.23), approximately defined as follows at ISA sea level [**9**]**:**

- N2 = 60 %;
- N1 = 20 %;
- FF = 275 Kg/H;
- EGT = 390 °C.

At this point, the user is given the task to switch ON the ENGINE 1 MASTER switch; this starts the starting sequence for engine 1, and since the steps are the same than the ones described for engine 2, the procedure uses node 12.6 to explain the similarity between the two starting sequences, then jumps to engine 1 in idle state (node 13.6). After that, the procedure is complete, and the user is able to change the tablet page displayed in figure 5.24 and start another procedure.

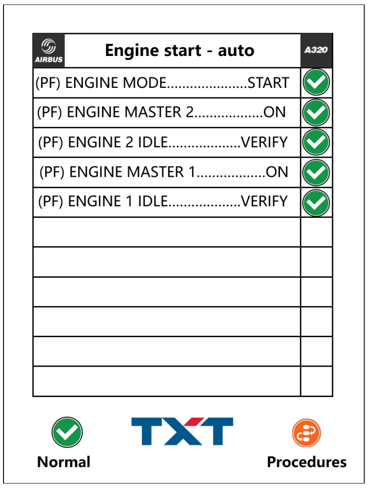


Figure 5.24: Engine start - automatic: "Engine start - automatic" page of the tablet when the procedure is complete

Node	Object	Exit Condition	Notes
3.6	-	-	Scene preparation
4.6	Engine mode selector	Current state = up (3-Way Switch)	Figure 5.22 (left)
5.6	ENGINE 2 MASTER switch	Current state = up (2-Way Switch)	Figure 5.22 (right)
6.6	Upper and Lower ECAM (ENGINE page) Display	Visually Inspect	N2 = 0 %, start valve open
7.6	Upper and Lower ECAM (ENGINE page) Display	Visually Inspect	N2 = 16 %, ignition fired
8.6	Upper ECAM Display	Visually Inspect	N2 = 22 %, FF activated
9.6	Upper and Lower ECAM (ENGINE page) Display	Visually Inspect	N2 = 50 %, ignition shut off, start valve closed
10.6	Upper ECAM Display	Visually Inspect	N2 = 60 %
11.6	Upper ECAM Display & ENGINE 1 MASTER switch	Visually inspect & Current state = down (2-Way Switch)	Engine 2 in idle condition (figure 5.23)
12.6	Upper ECAM Display	-	Engine 1 starting sequence summed up in a billboard
13.6	Upper ECAM Display	Visually Inspect	Engine 1 in idle condition
14.6	"Engine Start - automatic" page (figure 5.24)	Is Not Active	Resets the virtual scene

Table 5.8: Engine start - automatic procedure Nodes and Exit Conditions

5.1.6 After Start

The **After start** procedure starts by giving the task to the user of set the engine mode selector to the NORM position (node 4.7, figure 5.25 on the left), since chronologically at this point of the flight operations the start sequence is complete; for the same reason the user is given the task to shut off the APU bleed pushing the APU BLEED button to set it off (node 5.7, figure 5.25 on the right).

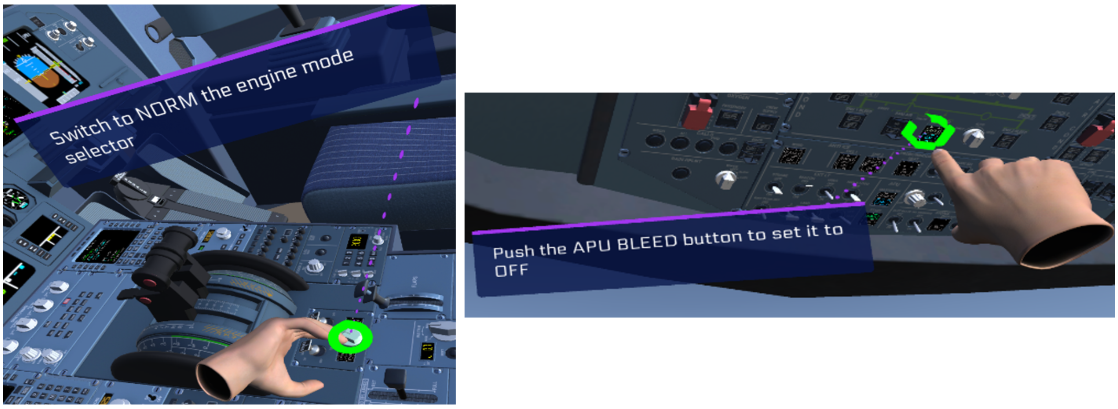


Figure 5.25: After start: node 4.7 (left) and node 5.7 (right)

After that, the user must visually inspect the Anti-Ice panel to check that all the anti-ice buttons are set as required (node 6.7, figure 5.26 on the left), then the APU must be shut off pushing the APU MASTER button (node 7.7, figure 5.26 on the right).

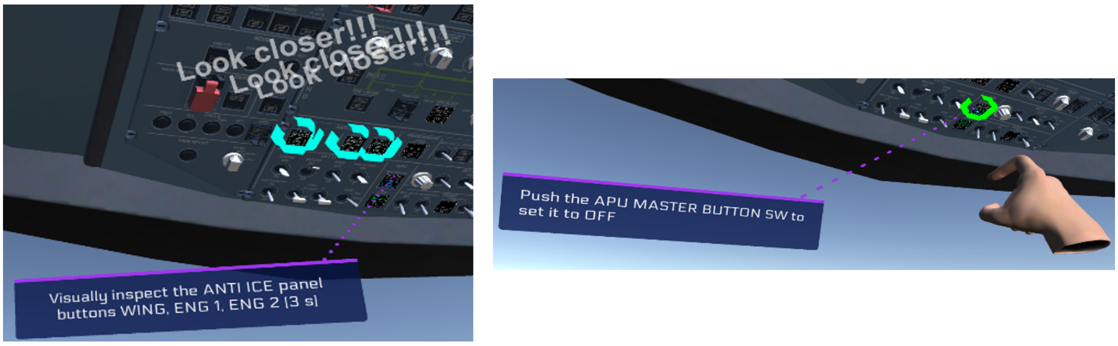


Figure 5.26: After start: node 6.7 (left) and node 7.7 (right)

At this point, the procedure asks the user to check the lower ECAM STATUS page pushing the STS button and to visually inspect the page (nodes 8.7 and 9.7, figure 5.27), and then to do the same with the upper ECAM to check that the nose-wheel steering disconnected memo is not displayed (node 10.7)

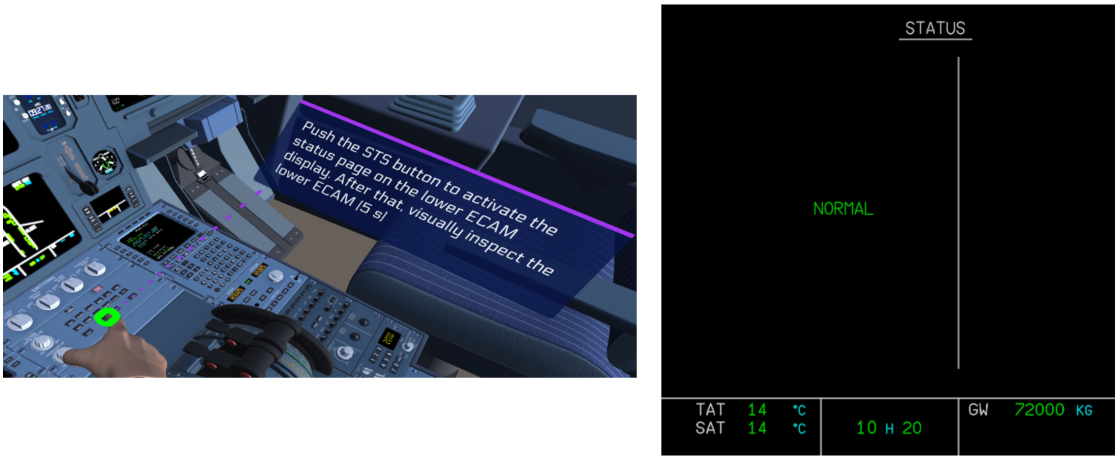


Figure 5.27: After start: node 8.7 (left) and lower ECAM STATUS page

The procedure then asks the user to arm the ground spoilers with the speed brake lever (node 11.7, figure 5.28 on the left), and to visually inspect the Rudder Trim display on the pedestal, and to push the RESET button if the number shown on the display is larger than 0.3 (nodes 12.7, figure 5.28 on the right).

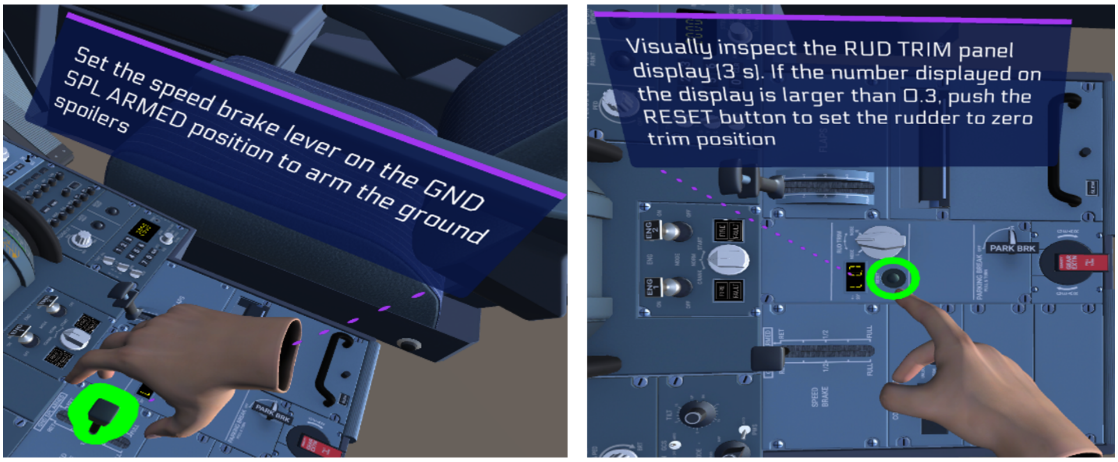


Figure 5.28: After start: node 11.7 (left) and node 12.7 (right)

The last steps of the procedure ask users to set the flap lever on the take-off (1) position (node 13.7, figure 5.29 on the left), then to visually inspect the pitch trim wheel (node 14.7, figure 5.29 on the right); finally, a step is dedicated to the lower ECAM check that should be done by the pilot monitoring (node 15.7), where the user must only wait several seconds to continue the procedure.

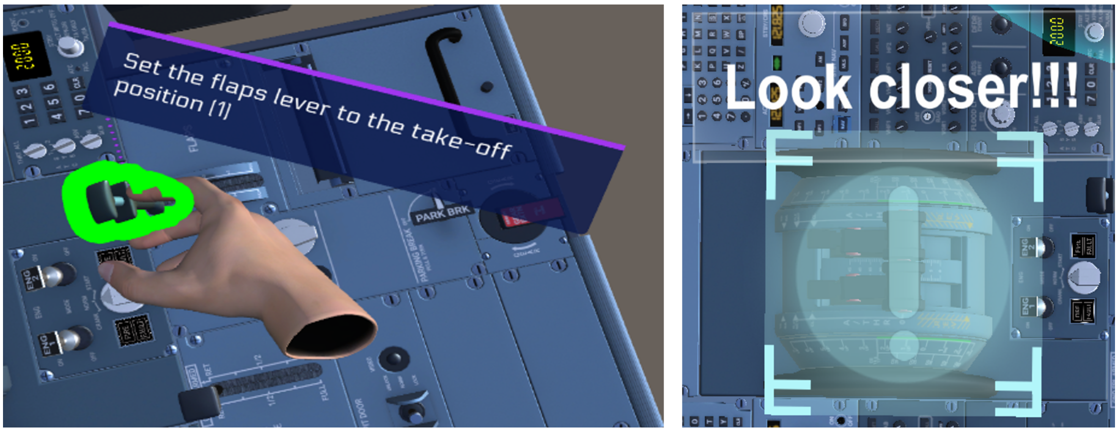


Figure 5.29: After start: node 13.7 (left) and node 14.7 (right)

Since the **After start** procedure has a checklist dedicated to its main items, the user is asked to navigate through the pages of the tablet in the virtual environment to open the **Checklists** page (node 16.7). The user starts from the page shown in figure 5.30 and navigates to the checklist shown in figure 4.32 on the right; once opened this page, they must wait 20 seconds before being able to change the page displayed and start another procedure.

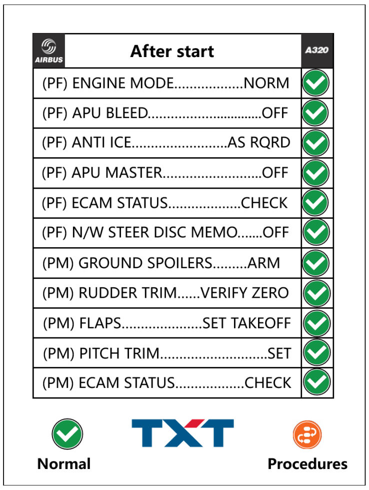


Figure 5.30: After start: "After start" page of the tablet when the procedure is complete

Node	Object	Exit Condition	Notes
3.7	-	-	Scene preparation
4.7	Engine mode selector	Current state = middle (3-Way Switch)	Figure 5.25 (left)
5.7	APU BLEED button	Current state = up (Push Button)	Figure 5.25 (right)
6.7	WING, ENG 1, ENG 2 anti-ice buttons	Visually Inspect	Figure 5.26 (left)
7.7	APU MASTER button	Current state = up (Push Button)	Figure 5.26 (right)
8.7	STS button	Has Executed	Figure 5.27 (left)
9.7	Lower ECAM (STATUS page) Display	Visually Inspect	Figure 5.27 (right)
10.7	Upper ECAM Display	Visually Inspect	-
11.7	Speed brake lever	Current state = up (3-Way Switch)	Figure 5.28 (left)
12.7	Rudder Trim display & RESET button	Visually Inspect & Current state = down (Push Button)	Figure 5.28 (right)
13.7	Flaps lever	Current state index = 1 (N-Way Switch)	Figure 5.29 (left)
14.7	Pitch trim wheel	Visually Inspect	Figure 5.29 (right)
15.7	Lower ECAM Display	-	(PM action) check the STATUS page
16.7	**Checklists** page	Is Active	-
17.7	**Checklists** page	Is Not Active	Resets the virtual scene

Table 5.9: After Start procedure Nodes and Exit Conditions

5.1.7 Taxi

The **Taxi** procedure begins with the task of pushing the AUTBRK MAX button to set it ON to activate the autobrake system in the correct mode for take-off (node 4.8, figure 5.31 on the left); in the next step, the user is asked to visually inspect the ATC panel to confirm the ATC code displayed (node 5.8, figure 5.31 on the right)

Figure 5.31: Taxi: node 4.8 (left) and node 5.8 (right)

Subsequently, the user must visually inspect the engine mode selector to check that the rotary switch is set to NORM (node 6.8), and must set the switches for the radar and its Predictive Wind-shear System respectively to 1 (on) (node 7.8, figure 5.32 on the left) and AUTO (node 8.8, figure 5.32 on the right).

Figure 5.32: Taxi: node 7.8 (left) and node 8.8 (right)

The following step consists in pushing the the ALL button (node 9.8, figure 5.33 on the left) in the Calls panel on the Overhead panel to receive the "Cabin ready" report from the crew. After that, the user must push the TO CONFIG button (node 10.8, figure 5.33 on the right) and subsequently visually inspect the upper ECAM (node 11.8) to check that among the take-off actions listed on the page there are no blue items (i.e. that have yet to performed).

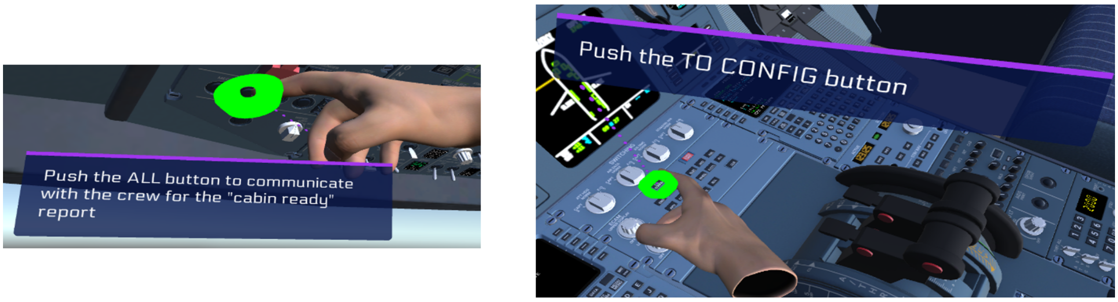


Figure 5.33: Taxi: node 9.8 (left) and node 10.8 (right)

The appearance of the upper ECAM at the start and at the end (node 11.8) of this procedure is shown in figure 5.34 below. The starting condition of this page (as shown in figure 5.34 on the left) has some take-off items still with blue memos beside them, and some have already been completed in previous procedures: the flaps and spoilers settings in **After start** (nodes 11.7 and 13.7) and the signs ON in **Cockpit preparation** (nodes 8.4, 9.4 and 10.4). In this procedure, after each item is dealt with, the page in the virtual enviroment changes, removing the corresponding blue memos from the page; after pushing the TO CONFIG button, the "TEST" blue memo disappears and the upper ECAM appears as shown in 5.34 on the right.

Figure 5.34: Taxi: Upper ECAM at the start (left) and at the end (right) of the procedure

Like some of the previous procedures, **Taxi** has also a checklist dedicated to its main items; because of this the user is asked to navigate through the pages of the tablet in the virtual world to open the **Checklists** page (node 12.8), starting from the page in figure 5.35 up to the checklist page; on this page the user must wait 20 seconds before being able to proceed to another procedure.

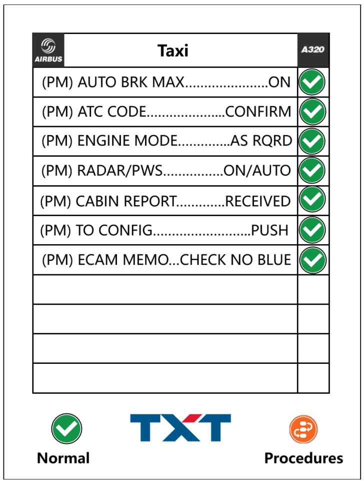


Figure 5.35: Taxi: "Taxi" page of the tablet when the procedure is complete

Node	Object	Exit Condition	Notes
3.8	-	-	Scene preparation
4.8	AUTO BRK MAX button	Current state = down (Push Button)	Figure 5.31 (left)
5.8	ATC panel display	Visually Inspect	Figure 5.31 (right)
6.8	Engine mode selector	Visually Inspect	-
7.8	PWS radar switch	Current state = down (2-Way Switch)	Figure 5.32 (left)
8.8	SYS radar switch	Current state = middle (3-Way Switch)	Figure 5.32 (right)
9.8	ALL button	Has Executed	Figure 5.33 (left)
10.8	TO CONFIG button	Has Executed	Figure 5.33 (right)
11.8	Upper ECAM Display	Visually Inspect	Figure 5.34 (right) - No blue memos condition
12.8	**Checklists** page	Is Active	-
13.8	**Checklists** page	Is Not Active	Resets the virtual scene

Table 5.10: Taxi procedure Nodes and Exit Conditions

Chapter 6

Conclusions

This thesis main results are a thorough research on how to test the effectiveness of VR training compared to traditional, computer-based training, as discussed in section 2.3, and 6 virtual training procedures modeled after the A320 SOPs, as discussed in section 5.1. Both these results can be discussed in terms of future developments, as proposed in the next sections.

6.1 VR training effectiveness

The evaluation of scientific literature that compares VR training effectiveness with other more or less traditional forms of training highlighted some key common factors that should be considered in future work of thesis with the goal of assessing VR training effectiveness, such as number of participants, number of performance tests to be administered to the participants, factors to be measured specifically for VR training and means to do so, and statistical analysis to be applied to the collected data. This research also highlighted some structural limitations to one of the previous work of thesis [5] that aimed to test the effectiveness of VR training. The most important limitation was probably the small sample size (5 people in total), caused by the restrictions imposed by the COVID-19 emergency; this limitation rendered a statistical analysis on the performance tests results meaningless, and as a result was not performed. Another significant limitation was that VR training was evaluated with a questionnaire for measuring mainly presence, but literature shows that there are other factors associated with presence which may be worth measuring to better contextualize a certain level of performance produced by VR training, such as cybersickness, system usability and task load. Furthermore, although the referenced thesis used a performance test a week after the VR training and the first performance test to assess knowledge retention, it does not provide any information on a measure of the participants prior knowledge of the technology/systems used in the experiment and on the subject of the training, which is another factor that can help to contextualize the performance results of both VR and computer-based training groups.

6.2 VR training procedures

The procedures developed for this thesis in the virtual environment improve upon the work of previous thesis in terms of user experience, with a better A320 cockpit as mentioned in section 4.3, which provides more equipment actually available in a real A320 cockpit; this helps in defining procedures more realistic as compared to the ones that should be performed by pilots on an A320. A particular care was given to create a scene that reacts consistently with the actions of the user in terms of displays, indicators and button lights; this was done in a consistent way for all the procedures mentioned in this thesis. For each and every node of the procedures defined in the previous chapter, billboards and audio instructions were created, to facilitate the interactions with the scene.
The user interface used to start the procedures inside the virtual environment, the tablet, was graphically modified to:

- Add information to the **Help** page;
- Substitute the **Checklist** page to address the latest Airbus amendments;
- Separate the "Normal" procedures from the "Abnormal & Emergency" procedures;
- Organize in chronological order and define the main SOPs in the **Normal procedures** pages.

The procedures implemented in this thesis were all on-ground SOPs to be performed before take-off; among those that can be realistically performed on-ground with a static aircraft and that have yet to be implemented in the Unity project created for this thesis, notable mentions from the A320 FCOM [9] are the **Preliminary Cockpit Preparation** procedure for the the before take-off operations, and the **After Landing**, **Parking** and **Securing the Aircraft** for the after landing operations; these last three procedures were implemented as part of previous works of thesis, [6] and [8]. Among the SOPs displayed in the **Normal procedures** pages of the tablet (figure 4.33), the **Exterior Inspection** is an on-ground procedure with a static aircraft that would require a model for the external surfaces of the A320, and not just the interiors of the cockpit. Outside of SOPs, other on-ground procedures could be implemented from the Supplementary/Non-routine normal procedures like the **Engine Start - manual** procedure or from the Abnormal

& Emergency procedures, like the **On Ground Emergency Evacuation**, both described in [3] and [4].
Other procedures among the SOPs that have yet to be implemented are the procedures in which the aircraft is not static, i.e. in which it is moving on-ground or flying. An implementation for some of these procedures was tried in [3] and [4], specifically for the **Take-off**, **Approach** and **Landing** procedures. These thesis used the WEAVR Simulation Hub (SimHub) module [1] provided by TXT/PACE to establish a connection between two simulation endpoints, X-Plane and WEAVR, and keep an information exchange between them. Its main components consist in:

- **SimHub server**: establishes connections between Clients using a Subscribe/Publish paradigm;
- **SimHub Client**: manages the data exchange between other clients; it connects to the server an provides to it an Interface Control Document (ICD), which is used by the server to connect it with other clients; the SimHub Client also can access the shared memory to read/write data used by the host simulation.
- **Shared Memory**: a dedicated library for shared memory access and management. Data can be exchanged locally (on the same host) between SimHub Client and customer's simulation.

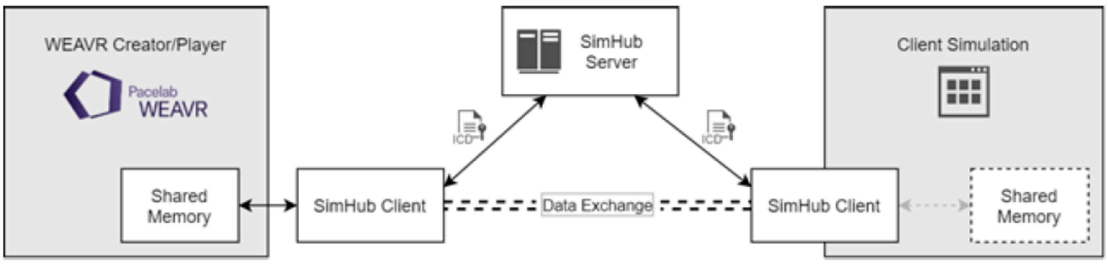


Figure 6.1: Simulation Hub module structure

The ICDs are *.xml* files which contain information about the Client (port, address and name), about the server (port and address) and the subscribed and published data, respectively the data that the Client receives from other Clients and the data sent to other Clients. In the ICDs for each of these data a label name, a description (optional) and a type are defined.

1. https://help.pace.de/xr/weavr/2.0.0/advanced-features/weavr-simulation-hub/index.html

WEAVR Creator uses the ICD file to define variable bindings to Game Objects using the "Element" component provided by the SimHub module; the binding can be of different types, *Read* (the Game Object is modified by the variable changing), *Write* (the user interacts with the Game Object to change its state, and this triggers a change in the bound variable) or *Both*.
The project described in [3] and [4] used the SimHub module with a Unity Client, provided by TXT, and an X-Plane Client, created by Sgarra and Tartaglia (the authors of the thesis and project) in C++, to exchange data between X-Plane and Unity/WEAVR. X-Plane is used with the X-Plane Connect[2] (XPC) plug-in installed (allows to control the simulation by sending commands and writing or reading *Datarefs* using functions written in C, C++, Java, MATLAB or Python), and should provide to Unity/WEAVR a number of data from the Primary Flight Display and others, like landing gear force. The Unity Client should send to X-Plane data related to the state of some of the components, like the side-stick, the thrust lever, the flaps and spoilers levers.
A file *.bat* was used to start the Simulation Hub server and the two Clients, Unity and X-Plane; inside the same file is written an identification number to select the procedure, which can be edited by the user to start in Unity one of the three procedures listed before while in game (**Take-off**=1, **Approach**=2, **Landing**=3).
The attempt at creating the communication between X-Plane and Unity with Simulation Hub was partially successful: a communication from X-Plane to Unity was established successfully, while the communication from Unity to X-Plane was not, which severely limited the functioning of the procedures built in Unity/WEAVR.
During the course of the thesis described in the previous chapters an attempt was made to solve this problem, but it was unsuccessful; on the project described in [3] and [4] a WEAVR Creator version with its latest update at November 2022 was substituted to the one used originally, and the X-Plane Client code was slightly modified to display the data published by Unity/WEAVR, which showed how no output data from Unity reached the X-Plane Client. As reported in the conclusions of the both [3] and [4], future work of thesis should address this issue, and could further improve the simulation by:

- Integrating the selection of the procedure in an element present in the virtual scene, i.e. the tablet;
- Transfer video streams of the A320 Cockpit in X-Plane to Unity, to visualize in the virtual environment the main displays and the external environment updated in real-time consistently with the X-Plane simulation environment.

2. https://software.nasa.gov/software/ARC-17185-1

Bibliography

[1] *Virtual Reality Market Size, Share & Trends Analysis Report By Technology (Semi & Fully Immersive, Non-Immersive), By Device (HMD, GTD), By Component (Hardware, Software), By Application, By Region, And Segment Forecasts, 2023 - 2030.* Tech. rep. GVR-1-68038-831-2. Grand View Research, 2022. URL: https://www.grandviewresearch.com/industry-analysis/virtual-reality-vr-market (cit. on pp. 1, 7).

[2] Sathiya Kumar Renganayagalu, Steven Mallam, and Salman Nazir. «Effectiveness of VR Head Mounted Displays in Professional Training: A Systematic Review». In: *Technology, Knowledge and Learning* 26 (Dec. 2021), pp. 1–43. DOI: 10.1007/s10758-020-09489-9 (cit. on pp. 1, 6, 9).

[3] Giovanni Sgarra. «Integration of Pacelab WEAVR software with X-Plane flight simulator for pilot training in virtual reality environment». Dec. 2021. URL: http://webthesis.biblio.polito.it/20934/ (cit. on pp. 1, 40, 61, 69, 73, 102, 103).

[4] Francesco Salvatore Tartaglia. «Digitalization of operational procedures in virtual reality for pilot training by integrating Pacelab WEAVR with the X-Plane flight simulator». Dec. 2021. URL: http://webthesis.biblio.polito.it/20935/ (cit. on pp. 1, 40, 61, 69, 73, 102, 103).

[5] Andrea Pavone. «Digitization and analysis of aeronautical operating procedures for the training of pilots in a virtual reality environment.» Dec. 2020. URL: http://webthesis.biblio.polito.it/16838/ (cit. on pp. 1, 19, 60, 61, 69, 100).

[6] Frederic Vianale. «Analysis of operative procedures for aeronautical personnel in virtual reality environment». July 2020. URL: http://webthesis.biblio.polito.it/15168/ (cit. on pp. 1, 61, 69, 101).

[7] Giuseppe Brizzi. «Digitalisation and analysis of the training of aircraft personnel through the use of virtual reality instruments». Dec. 2020. URL: http://webthesis.biblio.polito.it/16831/ (cit. on pp. 1, 61, 69).

[8] Lorenzo Fava. «Digitalization of operating procedures for the training of aviation personnel in a virtual reality environment.» Apr. 2020. URL: http://webthesis.biblio.polito.it/14634/ (cit. on pp. 1, 61, 69, 101).

[9] AIRBUS. *GLG A318/A319/A320/A321 For A/C: HC-CLF*. 2017 (cit. on pp. 2, 58, 68, 69, 71, 89, 101).

[10] Mehmet Ilker Berkman and Ecehan Akan. «Presence and Immersion in Virtual Reality». In: *Encyclopedia of Computer Graphics and Games*. Ed. by Newton Lee. Cham: Springer International Publishing, 2019, pp. 1–10. ISBN: 978-3-319-08234-9. DOI: 10.1007/978-3-319-08234-9_162-1. URL: https://doi.org/10.1007/978-3-319-08234-9_162-1 (cit. on p. 5).

[11] Doug A Bowman and Ryan P McMahan. «Virtual reality: how much immersion is enough?» In: *Computer* 40.7 (2007), pp. 36–43. URL: https://doi.org/10.1109/MC.2007.257 (cit. on p. 5).

[12] Joschka Mütterlein. «The Three Pillars of Virtual Reality? Investigating the Roles of Immersion, Presence, and Interactivity». In: Jan. 2018. DOI: 10.24251/HICSS.2018.174 (cit. on pp. 5, 9).

[13] Carli Ochs and Andreas Sonderegger. «The Interplay Between Presence and Learning». In: *Frontiers in Virtual Reality* 3 (Feb. 2022). DOI: 10.3389/frvir.2022.742509 (cit. on p. 5).

[14] Ayah Hamad and Bochen Jia. «How Virtual Reality Technology Has Changed Our Lives: An Overview of the Current and Potential Applications and Limitations». In: *International Journal of Environmental Research and Public Health* 19 (Sept. 2022), p. 11278. DOI: 10.3390/ijerph191811278 (cit. on p. 5).

[15] Z. Tacgin. *Virtual and Augmented Reality: An Educational Handbook*. Cambridge Scholars Publishing, 2020. ISBN: 9781527548725. URL: https://books.google.it/books?id=3UbhDwAAQBAJ (cit. on p. 5).

[16] Jialin Wang, Rongkai Shi, Wenxuan Zheng, Weijie Xie, Dominic Kao, and Hai-Ning Liang. «Effect of Frame Rate on User Experience, Performance, and Simulator Sickness in Virtual Reality». In: *IEEE Transactions on Visualization and Computer Graphics* PP (May 2023), pp. 1–11. DOI: 10.1109/TVCG.2023.3247057 (cit. on p. 6).

[17] Julian Keil, Dennis Edler, Thomas Schmitt, and Frank Dickmann. «Creating Immersive Virtual Environments Based on Open Geospatial Data and Game Engines». In: *KN - Journal of Cartography and Geographic Information* 71 (Jan. 2021). DOI: 10.1007/s42489-020-00069-6 (cit. on p. 6).

[18] Dimiter Velev and Plamena Zlateva. «Virtual Reality Challenges in Education and Training». In: *International Journal of Learning and Teaching* 3 (Mar. 2017), pp. 33–37. DOI: 10.18178/ijlt.3.1.33-37 (cit. on p. 6).

[19] *Virtual Reality Headset Market Size, Share & Trends Analysis Report By End-device (Low-end, Mid-range, High-end), By Product Type (Standalone, Smartphone-enabled), By Application, By Region, And Segments Forecasts, 2023 - 2030*. Tech. rep. GVR-4-68038-030-9. Grand View Research, 2022. URL: https://www.grandviewresearch.com/industry-analysis/virtual-reality-vr-headset-market (cit. on p. 7).

[20] Yugandhara R. Y. «Augmented and Virtual Reality Market Size, Share Report 2023». In: (Apr. 2023) (cit. on p. 7).

[21] Alex Hadwick. *VRX Industry Insight Report 2019-2020*. Tech. rep. VRX, 2020. URL: https://www.reutersevents.com/technology/xr-industry-report-2019-free-download (cit. on pp. 7, 8).

[22] Paweł Strojny and Natalia Dużmańska-Misiarczyk. «Measuring the effectiveness of virtual training: A systematic review». In: *Computers & Education: X Reality* 2 (2023), p. 100006. ISSN: 2949-6780. DOI: https://doi.org/10.1016/j.cexr.2022.100006. URL: https://www.sciencedirect.com/science/article/pii/S294967802200006X (cit. on pp. 9, 18, 20).

[23] Eric Krokos, Catherine Plaisant, and Amitabh Varshney. «Virtual memory palaces: immersion aids recall». In: *Virtual Reality* 23 (Mar. 2019). DOI: 10.1007/s10055-018-0346-3 (cit. on p. 9).

[24] Jun Zhang. «Immersive Virtual Reality Training to Enhance Procedural Knowledge Retention». In: 2019 (cit. on p. 9).

[25] Chong Chae, Daegun Kim, and Hyeong-Tak Lee. «A Study on the Analysis of the Effects of Passenger Ship Abandonment Training Using VR». In: *Applied Sciences* 11 (June 2021), p. 5919. DOI: 10.3390/app11135919 (cit. on p. 9).

[26] Lisa Goudman, Julie Jansen, Maxime Billot, Nieke Vets, Ann De Smedt, Manuel Roulaud, Philippe Rigoard, and Maarten Moens. «Virtual Reality Applications in Chronic Pain Management: Systematic Review and Meta-analysis». In: *JMIR serious games* 10 (May 2022), e34402. DOI: 10.2196/34402 (cit. on p. 12).

[27] Debra Boeldt, Elizabeth McMahon, Mimi McFaul, and Walter Greenleaf. «Using Virtual Reality Exposure Therapy to Enhance Treatment of Anxiety Disorders: Identifying Areas of Clinical Adoption and Potential Obstacles». In: *Frontiers in Psychiatry* 10 (Oct. 2019). DOI: 10.3389/fpsyt.2019.00773 (cit. on p. 12).

[28] Afsoon Asadzadeh, Taha Samad-Soltani, Zahra Salahzadeh, and Peyman Rezaei-Hachesu. «Effectiveness of virtual reality-based exercise therapy in rehabilitation: A scoping review». In: *Informatics in Medicine Unlocked* 24 (2021), p. 100562. ISSN: 2352-9148. DOI: 10.1016/j.imu.2021.100562. URL: https://www.sciencedirect.com/science/article/pii/S2352914821000526 (cit. on p. 12).

[29] Arata Andrade Saraiva, Alexandre Tolstenko Nogueira, N. M. Fonseca Ferreira, and Antonio Valente. «Application of virtual reality for the treatment of Strabismus and Amblyopia». In: *2018 IEEE 6th International Conference on Serious Games and Applications for Health (SeGAH)*. 2018, pp. 1–7. DOI: 10.1109/SeGAH.2018.8401357 (cit. on p. 12).

[30] Gideon Blumstein et al. «Randomized Trial of a Virtual Reality Tool to Teach Surgical Technique for Tibial Shaft Fracture Intramedullary Nailing». In: *Journal of Surgical Education* 77.4 (2020), pp. 969–977. ISSN: 1931-7204. DOI: 10.1016/j.jsurg.2020.01.002. URL: https://www.sciencedirect.com/science/article/pii/S1931720420300027 (cit. on p. 13).

[31] Ryan Lohre, Aaron Bois, J. Pollock, Peter Lapner, Katie McIlquham, George Athwal, and Danny Goel. «Effectiveness of Immersive Virtual Reality on Orthopedic Surgical Skills and Knowledge Acquisition Among Senior Surgical Residents: A Randomized Clinical Trial». In: *JAMA Network Open* 3 (Dec. 2020), e2031217. DOI: 10.1001/jamanetworkopen.2020.31217 (cit. on p. 13).

[32] Brandon McKinney, Ammer Dbeis, Ashley Lamb, Petros Frousiakis, and Stephan Sweet. «Virtual Reality Training in Unicompartmental Knee Arthroplasty: A Randomized, Blinded Trial». In: *Journal of Surgical Education* 79.6 (2022), pp. 1526–1535. ISSN: 1931-7204. DOI: 10.1016/j.jsurg.2022.06.008. URL: https://www.sciencedirect.com/science/article/pii/S1931720422001532 (cit. on p. 13).

[33] Eshita Dhar, Umashnkar Upadhaya, Yaoru Huang, Mohy Uddin, George Manias, Dimosthenis Kyriazis, Usman Wajid, Hamza Alshawaf, and Syed Abdul Shabbir. «A scoping review to assess the effects of virtual reality in medical education and clinical care». In: *Digital health* 9 (Feb. 2023), p. 20552076231158022. DOI: 10.1177/20552076231158022 (cit. on p. 13).

[34] Randi Q. Mao, Lucy Lan, Jeffrey Kay, Ryan Lohre, Olufemi R. Ayeni, Danny P. Goel, and Darren de SA. «Immersive Virtual Reality for Surgical Training: A Systematic Review». In: *Journal of Surgical Research* 268 (2021), pp. 40–58. ISSN: 0022-4804. DOI: 10.1016/j.jss.2021.06.045. URL: https://www.sciencedirect.com/science/article/pii/S0022480421004169 (cit. on p. 13).

[35] Bhone Kyaw et al. «Virtual Reality for Health Professions Education: Systematic Review and Meta-Analysis by the Digital Health Education Collaboration (Preprint)». In: (Nov. 2018). DOI: 10.2196/preprints.12959 (cit. on p. 13).

[36] Martin Boroš, Eva Sventeková, Anna Cidlinova, Marek Bardy, and Katerina Batrlova. «Application of VR Technology to the Training of Paramedics». In: *Applied Sciences* 12 (Jan. 2022), p. 1172. DOI: 10.3390/app12031172 (cit. on p. 13).

[37] Bethany Cieslowski, Tanya Haas, Kyeung Mi Oh, Kathleen Chang, and Cheryl A. Oetjen. «The Development and Pilot Testing of Immersive Virtual Reality Simulation Training for Prelicensure Nursing Students: A Quasi-Experimental Study». In: *Clinical Simulation in Nursing* 77 (2023), pp. 6–12. ISSN: 1876-1399. DOI: https://doi.org/10.1016/j.ecns.2023.02.001. URL: https://www.sciencedirect.com/science/article/pii/S1876139923000087 (cit. on p. 13).

[38] Elif İsmailoglu, Nilay Orkun, İsmet Rn, and Ayten Zaybak. «Comparison of the effectiveness of the virtual simulator and video-assisted teaching on intravenous catheter insertion skills and self-confidence: A quasi-experimental study». In: *Nurse Education Today* 95 (Dec. 2020), p. 104596. DOI: 10.1016/j.nedt.2020.104596 (cit. on pp. 13, 19, 22, 23).

[39] Fabrizia Mantovani, Gianluca Castelnuovo, Andrea Gaggioli, and Giuseppe Riva. «Virtual Reality Training for Health-Care Professionals». In: *Cyberpsychology & behavior : the impact of the Internet, multimedia and virtual reality on behavior and society* 6 (Sept. 2003), pp. 389–95. DOI: 10.1089/109493103322278772 (cit. on p. 13).

[40] Audrey Reichard and Larry Jackson. «Occupational Injuries Among Emergency Responders». In: *American journal of industrial medicine* 53 (Nov. 2009), pp. 1–11. DOI: 10.1002/ajim.20772 (cit. on p. 13).

[41] George Koutitas, Scott Smith, and Grayson Lawrence. «Performance evaluation of AR/VR training technologies for EMS first responders». In: *Virtual Reality* 25 (Mar. 2021). DOI: 10.1007/s10055-020-00436-8 (cit. on p. 14).

[42] Osama Halabi, Tooba Salahuddin, Abdel Karkar, and Guillaume Alinier. «Virtual reality for ambulance simulation environment». In: *Multimedia Tools and Applications* 81 (Sept. 2022). DOI: 10.1007/s11042-022-12980-3 (cit. on p. 14).

[43] Lisanne Kleygrewe, Vana Hutter, Matthijs Koedijk, and Raôul Oudejans. «Virtual reality training for police officers: a comparison of training responses in VR and real-life training». In: *Police Practice and Research* (Feb. 2023). DOI: 10.1080/15614263.2023.2176307 (cit. on p. 15).

[44] Jonathan Saunders, Steffi Davey, Petra Saskia Bayerl, and Philipp Lohrmann. «Validating Virtual Reality as an Effective Training Medium in the Security Domain». In: *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 2019, pp. 1908–1911. DOI: 10.1109/VR.2019.8798371 (cit. on p. 15).

[45] Pedro Monteiro, Miguel Melo, Antonio Valente, José Vasconcelos-Raposo, and Maximino Bessa. «Delivering Critical Stimuli for Decision Making in VR Training: Evaluation Study of a Firefighter Training Scenario». In: *IEEE Transactions on Human-Machine Systems* (Nov. 2020). DOI: 10.1109/THMS.2020.3030746 (cit. on p. 15).

[46] Rosângela Bail, Ariel Mi, Renan Bortolassi de Oliveira, Ilona Heldal, Cecilia Hammar Wijkmark, Eduardo Jose, and Slomp Aguiar. «Usability of immersive technology for the training and training of firefighters in Brazil». In: (June 2022) (cit. on p. 15).

[47] Scott Fisher, Monika McGreevy, J. Humphries, and W. Robinett. «Virtual environment display system». In: *Proceedings of the Workshop on Interactive 3-D Graphics* 1 (Jan. 1987), pp. 77–87. DOI: 10.1145/319120.319127 (cit. on p. 16).

[48] R.B. Loftin and P. Kenney. «Training the Hubble space telescope flight team». In: *IEEE Computer Graphics and Applications* 15.5 (1995), pp. 31–37. DOI: 10.1109/38.403825 (cit. on p. 16).

[49] Angelica D. Garcia, Jonathan Schlueter, and Eddie Paddock. «Training Astronauts using Hardware-in-the-Loop Simulations and Virtual Reality». In: *AIAA Scitech 2020 Forum*. DOI: 10.2514/6.2020-0167. eprint: https://arc.aiaa.org/doi/pdf/10.2514/6.2020-0167. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2020-0167 (cit. on p. 16).

[50] Stephen Ennis et al. «Astronaut training on-board the International Space Station using a standalone Virtual Reality headset». In: *72nd International Astronautical Congress (IAC 2021)*. 2021. URL: https://elib.dlr.de/147953/ (cit. on p. 17).

[51] Unnikrishnan Radhakrishnan, Koumaditis Konstantinos, and Francesco Chinello. «A systematic review of immersive virtual reality for industrial skills training». In: *Behaviour and Information Technology* 40 (July 2021). DOI: 10.1080/0144929X.2021.1954693 (cit. on pp. 18, 20).

[52] Franz Faul, Edgar Erdfelder, Albert-Georg Lang, and Axel Buchner. «G*Power 3: A flexible statistical power analysis program for the social, behavior, and biomedical sciences». In: *Behavior Research Methods Instruments & Computers* 39 (May 2007), pp. 175–191. DOI: 10.3758/BF03193146 (cit. on p. 19).

[53] Ariel Caputo, Sergiu Jacota, Serhiy Krayevskyy, Marco Pesavento, Fabio Pellacini, and Andrea Giachetti. «XR-Cockpit: a comparison of VR and AR solutions on an interactive training station». In: Sept. 2020. DOI: 10.1109/ETFA46521.2020.9212043 (cit. on pp. 19, 23).

[54] Sem Hardon, Anton Kooijmans, Roel Horeman, Maarten Elst, Bobby Bloemendaal, and Tim Horeman. «Validation of the portable virtual reality training system for robotic surgery (PoLaRS): a randomized controlled trial». In: *Surgical Endoscopy* 36 (July 2022), pp. 1–11. DOI: 10.1007/s00464-021-08906-z (cit. on pp. 19, 22, 23).

[55] Frederik Winther, Linoj Ravindran, Kasper Svendsen, and Tiare Feuchtner. «Design and Evaluation of a VR Training Simulation for Pump Maintenance Based on a Use Case at Grundfos». In: Mar. 2020, pp. 738–746. DOI: 10.1109/VR46266.2020.1580939036664 (cit. on p. 19).

[56] Nirit Gavish, Teresa Gutierrez, Sabine Webel, Jorge Rodríguez-Arce, Matteo Peveri, and Franco Tecchia. «Evaluating virtual reality and augmented reality training for industrial maintenance and assembly tasks». In: *Interactive Learning Environments* 23 (Dec. 2013), pp. 1–21. DOI: 10.1080/10494820.2013.815221 (cit. on p. 19).

[57] Elif İsmailoglu and Ayten Zaybak. «Comparison of the Effectiveness of a Virtual Simulator With a Plastic Arm Model in Teaching Intravenous Catheter Insertion Skills». In: *CIN: Computers, Informatics, Nursing* 36 (Nov. 2017), p. 1. DOI: 10.1097/CIN.0000000000000405 (cit. on pp. 19, 22, 23).

[58] John Brooke. «SUS: A quick and dirty usability scale». In: *Usability Eval. Ind.* 189 (Nov. 1995) (cit. on p. 20).

[59] Jörn Hurtienne and Anja Naumann. «QUESI-A questionnaire for measuring the subjective consequences of intuitive use». In: *Interdisciplinary College 2010* (Jan. 2010) (cit. on p. 20).

[60] Sandra G. Hart and Lowell E. Staveland. «Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research». In: *Human Mental Workload*. Ed. by Peter A. Hancock and Najmedin Meshkati. Vol. 52. Advances in Psychology. North-Holland, 1988, pp. 139–183. DOI: https://doi.org/10.1016/S0166-4115(08)62386-9. URL: https://www.sciencedirect.com/science/article/pii/S0166411508623869 (cit. on p. 20).

[61] David Harris, Mark Wilson, and Samuel Vine. «Development and validation of a simulation workload measure: The Simulation Task Load Index (SIM-TLX)». In: (Dec. 2019). DOI: 10.31234/osf.io/pm5ht (cit. on p. 20).

[62] Susanne Putze, Dmitry Alexandrovsky, Felix Putze, Sebastian Höffner, Jan Smeddinck, and Rainer Malaka. «Breaking The Experience: Effects of Questionnaires in VR User Studies». In: Apr. 2020, pp. 1–15. DOI: 10.1145/3313831.3376144 (cit. on p. 21).

[63] Séamas Weech, Sophie Kenny, and Michael Barnett-Cowan. «Presence and Cybersickness in Virtual Reality Are Negatively Related: A Review». In: *Frontiers in Psychology* 10 (Feb. 2019), p. 158. DOI: 10.3389/fpsyg.2019.00158 (cit. on pp. 21, 22).

[64] Valentin Schwind, Pascal Knierim, Nico Haas, and Niels Henze. «Using Presence Questionnaires in Virtual Reality». In: May 2019. DOI: 10.1145/3290605.3300590 (cit. on p. 21).

[65] «OmniPres project IST-2001-39237 Deliverable 5 Measuring Presence : A Guide to Current Measurement Approaches». In: 2004 (cit. on p. 21).

[66] Bob G. Witmer and Michael J. Singer. «Measuring Presence in Virtual Environments: A Presence Questionnaire». In: *Presence: Teleoperators and Virtual Environments* 7.3 (June 1998), pp. 225–240. DOI: 10.1162/105474698565686. eprint: https://direct.mit.edu/pvar/article-pdf/7/3/225/1836425/105474698565686.pdf. URL: https://doi.org/10.1162/105474698565686 (cit. on pp. 21, 22).

[67] Thomas W. Schubert. «The sense of presence in virtual environments: A three-component scale measuring spatial presence, involvement, and realness». In: *Z. für Medienpsychologie* 15 (2003), pp. 69–71. DOI: 10.1026//1617-6383.15.2.69. URL: http://dx.doi.org/10.1026//1617-6383.15.2.69 (cit. on p. 21).

[68] Martin Usoh, Ernest Catena, Sima Arman, and Mel Slater. «Using Presence Questionnaires in Reality». In: *Presence: Teleoperators and Virtual Environments* 9 (Apr. 2000). DOI: 10.1162/105474600566989 (cit. on p. 21).

[69] Robert S. Kennedy, Norman E. Lane, Kevin S. Berbaum, and Michael G. Lilienthal. «Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness». In: *The International Journal of Aviation Psychology* 3.3 (1993), pp. 203–220. DOI: 10.1207/s15327108ijap0303_3. eprint: https://doi.org/10.1207/s15327108ijap0303_3. URL: https://doi.org/10.1207/s15327108ijap0303_3 (cit. on p. 22).

[70] Pauline Bimberg, Tim Weissker, and Alexander Kulik. «On the Usage of the Simulator Sickness Questionnaire for Virtual Reality Research». In: Mar. 2020, pp. 464–467. DOI: 10.1109/VRW50115.2020.00098 (cit. on p. 22).

[71] Hyun K. Kim, Jaehyun Park, Yeongcheol Choi, and Mungyeong Choe. «Virtual reality sickness questionnaire (VRSQ): Motion sickness measurement index in a virtual reality environment». In: *Applied Ergonomics* 69 (2018), pp. 66–73. ISSN: 0003-6870. DOI: https://doi.org/10.1016/j.apergo.2017.12.016. URL: https://www.sciencedirect.com/science/article/pii/S000368701730282X (cit. on p. 22).

[72] Panagiotis Kourtesis, Josie Linnell, Rayaan Amir, Ferran Argelaguet, and Sarah E. MacPherson. «Cybersickness in Virtual Reality Questionnaire (CSQ-VR): A Validation and Comparison against SSQ and VRSQ». In: *Virtual Worlds* 2.1 (2023), pp. 16–35. ISSN: 2813-2084. DOI: 10.3390/virtualworlds2010002. URL: https://www.mdpi.com/2813-2084/2/1/2 (cit. on p. 22).

[73] R. A. Fisher. «The Arrangement of Field Experiments». In: *Breakthroughs in Statistics: Methodology and Distribution*. Ed. by Samuel Kotz and Norman L. Johnson. New York, NY: Springer New York, 1992, pp. 82–91. DOI: 10.1007/978-1-4612-4380-9_8. URL: https://doi.org/10.1007/978-1-4612-4380-9_8 (cit. on p. 23).

[74] E. L. Lehmann and Joseph P. Romano. *Testing statistical hypotheses*. Third. Springer Texts in Statistics. New York: Springer, 2005, pp. xiv+784. ISBN: 0-387-98864-5 (cit. on p. 23).

[75] Prabhakar Mishra, ChandraM Pandey, Uttam Singh, Anshul Gupta, Chinmoy Sahu, and Amit Keshri. «Descriptive Statistics and Normality Tests for Statistical Data». In: *Annals of Cardiac Anaesthesia* 22 (Jan. 2019), pp. 67–72. DOI: 10.4103/aca.ACA_157_18 (cit. on p. 24).

[76] PACE-TXT. «PACELAB WEAVR, WEAVR – User Manual, Version: 2.0.3 (LTS)» (cit. on pp. 31, 34, 36).

[77] Airbus. «A319/A320/A321 Flightdeck and systems briefing for pilots, STL 945.7136/97». Sept. 1998. URL: https://www.smartcockpit.com/docs/A320_Flight_Deck_and_Systems_Briefing_For_Pilots.pdf (cit. on pp. 41, 42).

[78] Airbus. «Flight Operations Safety Awareness Seminar (FOSAS) - The Airbus Cockpit Philosophy». Sept. 2017 (cit. on p. 41).

[79] Giorgio Guglieri. «Flight Simulation. Dispense del Corso di Simulazione del Volo». 2021 (cit. on p. 60).

[80] AIRBUS. *GLG A318/A319/A320/A321 FLEET FCTM*. 2017 (cit. on p. 67).