1. Introduction

Good afternoon. We are group 7 and we are going to present the analysis for project 1. Its name is Zephyrus and its intent is to implement a wrist motion-controlled remote car.

2. Agenda

Here is the agenda for the presentation.

- Problem Statement
- Market Study
- Requirements and Constraints
- System Overview
- System Architecture
- Dynamic System Modelling
- Project Planning

[Sugestion]: *It will start with the Problem Statement and Market Study. Then the Requirements and Constraints for the system will be presented. After that the System's Overview, Architecture and Dynamic Modelling will be analysed. The presentation will end with an overview of the plan for the timing on the different stages of the project.*

3. Problem statement

Motion-tracking and gesture-controlled robotics have been gaining popularity in various professional areas.

With that in mind, this project intends to serve as a research and development tool for wearable-controlled technologies and will consist of a car, remotely controlled by a wristband using natural wrist and finger movements.

4. Expected growth

In this image we can see the projected gesture recognition market growth from 2019 to 2024.

5. Market Study

There exist, in the market for products similar to Zephyrus, ones with appealing designs and impactful first impressions. However, these products are either uncomfortable to use or feel cheap and brittle.

6. Added Value

Zephyrus aims to match these products in those characteristics but beat them in comfort and ease of use.

7. Requirements

Functional

The product is then required to sense user inputs and turn them into movement in the car. This should be done through a wireless between the local and remote systems connection that the user can control.

Non-functional

All of it should be done as efficiently as possible, both in terms of resources (having low power consumption) and time (having low latency in the connection between the local and remote connection). Also, as mentioned before, it should be comfortable and practical to control.

8. Constraints

Technical

Here are the technical constraints, regarding the tools and technologies that must be leveraged in the development of the project...

Non-technical

... as well as the constraints in terms of the project management.

9, 10, 11. System overview [Intro]

This Figure depicts the system overview, where one can identify the main and remote boards and the ways in which they interact with each other and the external environment.

12. Hardware Architecture

The hardware in the Local Board is focused on power efficiency but also computing performance, hence the use of a faster processor with more capable peripherals. The hardware on the remote system concentrates more sensors but requires less computational power. For that reason, it's projected to use a less capable yet more energy-efficient processor.

13. Software Architecture

The software architecture for both devices include high abstraction for communicating with each other and sensing modules, as well as power management and task scheduling. The local system, specifically, includes two high-level layers of abstraction for the Reinforcement Learning model.

14. Reinforcement Learning

The main board hosts the Reinforcement Learning inference engine, based in an actor-critic model.

15. Reinforcement Learning

This diagram represents the stages in which the RL model will be trained, converted, and deployed. The training will be done in a more resource-capable device and then deployed on the local board, which is a resource-scarce edge device.

This will be done using a method that promotes the best results in terms of embedded optimization.

16. System Events

To first characterize the system, a model-focused approach was adopted, using a sub-set of UML models to describe the intended behaviour for each subsystem.

The remote system is mainly focused on periodically and synchronously sampling user inputs and sending information about them to the local system. The local system responds to these inputs, also periodically and synchronously, satisfying these commands as it travels through the environment.

17. Local System Use Cases

In the Local System's case, all the interactions happen with, and are dictated by, the Remote System.

18. Remote System Use Cases

The Remote System serves as a bridge between the user and the Local System. Because of that, its behaviour is defined by its interactions with them.

19. Local System State Diagram [1]

When starting up, the Local System begins by initializing the hardware peripherals, preparing the Reinforcement Learning Model and listening for a connection from the Remote System.

20. Local System State Diagram [2]

Upon establishing a connection, it starts listening for the environment through its own sensors and the Remote System's sensors as well. At the same time, it feeds that information to the Reinforcement Learning Inference Engine, which periodically computes the actuator values.

21. Remote System State Diagram

The Remote System starts up by initialising the sensors as well, and automatically requesting the connection for which the Local System is listening.

Once a connection is established, as previously discussed, it starts periodically sampling the user's movement to then send it to the car.

While doing this, it also allows the user to end and afterwards re-establish the connection.

22. Local System Sequence Diagrams [1]

This is what the Local System's event response looks like from a sequential perspective.

23., 24. Local System Sequence Diagrams [2] / Remote System Sequence Diagram

These sequence diagrams shine further light on the sequence and timing of events in the Local and Remote Systems.

25. Estimated Budget

Our estimated budget for hardware and structure of the project is 82 euros.

26. Gantt Diagram

The Gantt Diagram is presented in this slide and presents the tasks assigned to each week until the deadline.

[Sugestion]: *The Gantt diagram for the foreseen project organization and timing anticipates the end of the design phase by the beginning of December, kicking off the Implementation stage, which should last until two weeks before the project deadline. This time should be enough to test the implemented system.*