**< Radio-controlled data collecting plane/RCDCP>**

**Requirements analysis and idea formulation**

**Digital Media and Design course**

**Version 0.0.1**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 29.10.2024 | 0.0.1 | Theoretical calculation, parts buying, getting requested environment, searching for the software | Aleksander Sem.  Aleksander Ser.  Alex  Misha |

# Problem research and analysis

Research and record ideas, observations and insights relevant to intentions, reflecting critically on work and progress. Minimum 500 words.....

The idea is to create an RC plane that can record and send telemetric data, including air pressure and wind speed, to a computer.

Research shows plenty of ways to create RC controls and data transmission in real time, depending on your financial situation. We’ve stuck to the ESP-32 WIFI transmission system. It’s the best in our situation due to the cheapness and flexibility of this technology. The cheapness is reached thanks to not so big cover area and few channel system. The flexibility is achieved due to compatibility with Arduino architecture. Due to this compatibility, all the other electronics are Arduino-compatible. We are using Micro Servos to control movable surfaces. We are using two three-bladed propellers on each side. Ideally, the plane should weigh not more than 2 kg. For that purpose, we are using highly efficient brushless motors. For mobility and repairability sake, our plane will consist of several parts, to be exact, two separatable wings, a body and a nose cone. Also, for safety, we are planning on adding an emergency parachute, it’ll be shot once the plane loses connection or if the accelerometer detects the too-dangerous angle of the plane relative to the ground. The computer is connected using the transmitter with a connected antenna; the coverage distance is approximately 300 meters.

The software is designed to show all collected data on a screen, as well as give live-time video from a camera connected to the plane, as well as give a user the ability to control the aircraft. Some of the assembly process will happen at our homes, but most of it will happen in the RTU scientific laboratory. The lab can provide us with most of the equipment, such as a 3D printer soldering-iron devices, Arduino development boards and for a short period of time in the beginning, power supplies for tests.

Our project may face with different types of problems starting with software issues and ending with a physical plane disability to fly. The list of hypothetical issues we may face with: Plane body overweight or lack of hardness – plane must be light and at the same time lasting enough to endure possible overload either positive or negative therefore we will need to find the best material for the plane body probably it with be some kind of plastic; Issues with the correct electronics assembling and also a risk of losing some part while testing systems, motors will consume a lot of energy and in case of incorrect assembling or calculations we may lose motors or other components; Problems with the transmitter or receiver, we will do telecommunication system for the first time and we probably will get some issues with it; Writing code for the Arduino could become a problem because we don’t have experience on C++ language; Also we don’t have experience of the creating own application and UI, we will need to learn photoshop. Of course will may face other problems at every stage of the project, but I hope that problems will only disappear or get resolved.

# Software specification

**1.1 Epic definition (5 points)**

For explorers and normal/nerdy people.

Who want to get information about environmental qualities, about air and specific places, or just for fun!

The ability of getting into places and domains with high risks for humans or just to entertain.

It is a material flying object for research and an opportunity to get to know the world of aircraft with our own created software.

That allows our customers to easily achieve what they want.

Unlike others, we create our product, especially for each customer, which means that all parts of the plane are created with each of our customer’s needs in mind, which makes the plane special.

Our company can make planes suitable for any customer’s need.

**1.2. Business process model (10 points)**

<Description of the main process – think about what processes take place in the project and who is involved in them.>

First of all, to start with our plane you need to visit our site Meteo-Plane.net. Order the plane and download the software to be able to control and monitor the plane telemetry and data collecting. To use the plane safety we recommend reading our instructions.

First, you need to charge up the plane. You will need to find a place for Takeoff and Landing as well, it must be from 30 meters of asphalt(smooth☺).

**1.3. Process diagram (10 points)**

A diagram of a plane

Description automatically generated

**1.4. Defining requirements (40 points in total)**

<Can be in the form of user stories (User stories)>

Functional requirements: (15 points)

1. What the system does-
2. ‘
3. ‘
4. ‘
5. ‘
6. ‘
7. ‘
8. ‘
9. ‘
10. ‘
11. ‘
12. ‘
13. ‘
14. ‘

Non-functional requirements: (15 points)

<15 non-functional or technical requirements>

Selection and justification for initial programming tools, frameworks, languages ​​(10 points)

# References

1. r/esp32 Sufficient-Market940 3 months ago <https://www.reddit.com/r/esp32/comments/1es32ut/doing_a_rc_car_from_scratch/>
2. Brian Lough’s YouTube video 6 years ago

<https://www.youtube.com/watch?v=0zs-A_fC3Yg&t=14s>

1. 3JWings’s YouTube video 6 months ago

<https://www.youtube.com/watch?v=dOqChqk8AAA>