# **Functional Design Report**

A line break

Trick question Inc. - Reverse Geocaching Box Development





**Embedded Systems Engineering Academy Engineering and Automotive HAN University of Applied Sciences**

**Authors**  
2158469 Federico Giovanni Accossato  
2153089 Henri Krinke  
2140725 Jabez Impano  
2158363 Mohammed Alkhomaish Hernandez  
2156231 Pedro von Sydow

**Tutor**

Remko Welling

**Customer**

Hugo Arends

**Date**

March 2025

**Version**

0.5

Revisions

|  |  |  |  |
| --- | --- | --- | --- |
| Version | When | Who | What |
| 0.1 | 13/03/2025 | Federico | Created the file transferred “functional design” from the product report. |
| 0.1.1 | 13/03/2025 | Federico | Created the “Sketch of the interface” Figures. |
| 0.1.3 | 16/03/2025 | Mohammed | Added a cover page, “revisions” section, “Table of contents” and “introduction’ |
| 0.1.4 | 20/04/2025 | Henri | Implemented changes to the specifications |
| 0.1.5 | 20/04/2025 | Jabez | User stories for puzzle games |
| 0.1.6 | 20/04/2025 | Pedro | User story flow chart |

Table of Contents

[1 Introduction 4](#_Toc193448619)

[1.1 Background 4](#_Toc193448620)

[1.2 Reason 4](#_Toc193448621)

[1.3 Report Structure 4](#_Toc193448622)

[2 Functional Design 5](#_Toc193448623)

[2.1 2.1 Functional specifications 5](#_Toc193448624)

[2.2 Technical specifications 7](#_Toc193448625)

[3 User interface 8](#_Toc193448626)

[3.1 Puzzle-Box: 8](#_Toc193448627)

[3.1.1 Appearance and Layout 8](#_Toc193448628)

[3.1.2 Sketch of the interface 8](#_Toc193448629)

[3.1.3 User Story 9](#_Toc193448630)

[3.1.4 Interaction flowchart 10](#_Toc193448631)

[3.1.5 Inputs 11](#_Toc193448632)

[3.1.6 Outputs 11](#_Toc193448633)

[3.1.7 Interaction and Changes 11](#_Toc193448634)

[3.1.8 Puzzles 12](#_Toc193448635)

[3.2 Companion laptop: 13](#_Toc193448636)

[3.2.1 Appearance and Layout: 13](#_Toc193448637)

[3.2.2 Sketch of the interface: 13](#_Toc193448638)

[3.2.3 Inputs: 14](#_Toc193448639)

[3.2.4 Outputs: 14](#_Toc193448640)

[3.2.5 Interactions and changes: 14](#_Toc193448641)

[4 References 15](#_Toc193448642)

# Introduction

The Functional Design of the Reverse Geocaching Box defines the essential behaviours and capabilities the system must exhibit to fulfil its purpose. This section outlines the specific requirements and functionalities that ensure the box operates as intended.

## Background

This document is part of the development process for a Reverse Geocaching Box, which is an interactive system that integrates GPS tracking and puzzle-solving elements to create a fun user experience. Unlike traditional geocaching, where users locate a hidden object using GPS coordinates, this system requires players to solve a series of challenges before unlocking the box they brought with them.

## Reason

The Functional Design is a crucial aspect of the project as it establishes the foundation for how the Reverse Geocaching Box interacts with users and processes data. Without well-defined functionalities, the system may not perform reliably or meet client expectations. By categorizing requirements using the MoSCoW (MoSCoW method , 2025) prioritization method (Must-have, Should-have, Could-have, and Won’t-have) and the SMART (SMART criteria , 2025)(Specific, Measurable, Achievable, Realistic, Timely) this section ensures that critical features such as GPS tracking, locking mechanisms, and puzzle elements are properly defined for successful implementation. The detailed requirements in combination with the user interface will serve as a structured foundation for the technical design and further development of the system.

## Report Structure

This section of the report focuses on the Functional Design of the Reverse Geocaching Box and is structured as follows:

1. **Functional Specifications** – Defines the core functionalities of the system, categorized based on priority using the MoSCoW method.
2. **Technical Specifications** – Describes the technical requirements discussed with the client.
3. **User Interface** – Details how users engage with the system, including input and output components such as buttons, sensors, and display elements. Also explains the interactions between inputs and outputs.

# Functional Design

## Functional specifications

The functional specifications define the behaviors and features the Reverse Geocaching Box must exhibit. These specifications are written using the (SMART criteria , 2025) SMART criteria to ensure they are Specific, Measurable, Assignable, Realistic, and Time-related. They are also categorized into must, should and could have according to the MoSCoW method (MoSCoW method , 2025).

Table 1: Functional Specifications

|  |  |  |
| --- | --- | --- |
| Nr. | MoSCow | Description |
| F1 | M | Must consist of an openable box |
| F1.1 | M | Must be able to be locked and unlocked |
| F1.1.2 | M | Must use an actuator for the locking mechanism |
| F1.2 | S | Should be portable |
| F1.3 | M | Must be able to be powered on and off |
| F2 | M | The box must be able to track its own position |
| F2.1 | M | Must be tracked every second |
| F2.2 | M | Must be able to compare its own position to a target position |
| F2.3.1 | S | Should be able to handle multiple target locations |
| F2.3.2 | M | Must be able to determine if it is at the target location |
| F2.3.3 | M | Must be able to determine the direct distance to the location |
| F2.3.4 | S | Should be able to determine in which compass direction the target position is located |
| F3 | M | Must have multiple puzzle elements before the box can be opened |
| F3.1 | S | There should be puzzles at 4 different locations |
| F3.2 | S | Location one should contain a memory game |
| F3.2.1 | S | A sequence of arrows should be displayed on a display |
| F3.2.2 | S | The sequence should be replicated using a joystick |
| F3.3 | S | Location two should contain a colour dependant game |
| F3.3.1 | S | Depending on a presented colour different information should be displayed |
| F3.3.1.1 | S | When green is presented the compass direction of the next location should be shown for one minute |
| F3.3.1.2 | S | When brown is presented the distance to the next location should be shown for 30 seconds |
| F3.3.1.3 | S | When White/Grey is presented the option to receive a hint about the next location will be shown for 15 seconds |
| F3.3.1.4 | S | Each colour option should only be able to be chosen three times |
| F3.4 | S | Location three should contain a balancing game |
| F3.4.1 | S | The box should be kept straight until reaching the final location |
| F3.4.2 | S | The compass direction of the final location should be shown |
| F3.5 | M | At each of the location a light corresponding to a number should display |
| F3.5.1 | M | At the final location all the numbers should be entered to unlock the box |
| F4 | M | The box must be able to display relevant information on an interface |
| F4.1 | M | Must be able to show the puzzles |
| F4.1.1 | S | Should be able to display distance to target coordinates |
| F4.1.2 | S | Should be able to display compass location of target location |
| F4.1.3 | S | Should be able to display arrows in vertical, horizontal and diagonal directions |
| F4.1.2 | C | Could display puzzle descriptions |
| F4.2 | S | Should be able to show the runtime of the batteries and |
| F4.3 | M | Must be able to show if a GPS connection has been established |
| F5 | M | The box must be able to log certain information |
| F5.1 | M | Must be able to log the GPS coordinates |
| F5.2 | M | Must be able to log the temperature outside the box |
| F5.3 | M | Must be able to store puzzle progress |
| F5.3.1 | S | Should be logged after every solved puzzle |
| F5.4 | M | Logged information must be stored in persistent memory |
| F6 | M | The box must be able to communicate with a companion laptop |
| F6.1 | S | Should only be accessible once the box has been opened |
| F6.2 | M | Should be able to set puzzle information |
| F6.3 | M | Must be able to set target locations |
| F6.4 | M | Must be able to display logged information |
| F6.4.1 | S | Should be able to clear logged information from persistent memory |

## Technical specifications

The technical specifications contain additional specifications that the customer/client requires of the product. They have been clarified in the project guide as well as in a private meeting with the client

Table 2: Technical Specifications

|  |  |
| --- | --- |
| **Nr.** | **Specifications** |
| T1 | Must be controlled using the FRDMMcxa153-development board |
| T2 | Must use the ATGM336H GPS module |
| T3 | Must have a tracking accuracy of a 10-meter radius |
| T4 | Must be programmed using C programming language |
| T5 | Must log information into persistent memory at specific intervals |
| T5.1 | Must log its position every 10 metres |
| T5.2 | Must log the temperature every 10 seconds |
| T6 | Should be a maximum of 20cm\*20cm, 30cm\* 30cm |
| T7 | Should have a separate compartment for the power supply |

# User interface

The user interface of the Reverse Geocaching Box plays a crucial role in how users interact with the system. It includes both input and output components that facilitate user engagement with the puzzle-solving and 2 process. Additionally, the interface is split into two parts, the outside of the box with its sensors, modules and the display and the companion laptop with its keyboard and screen.

## Puzzle-Box:

### Appearance and Layout

The Reverse Geocaching Box will have a compact and portable design, with a clear distinction between input and output elements. The enclosure will house the electronic components, including the GPS module, sensors, and microcontroller, while the interface elements will be positioned for intuitive user interaction.

### Sketch of the interface

Immagine che contiene testo, schermata, diagramma, Carattere

Descrizione generata automaticamente

Figure 1: Puzzle Box Sketch

### User Story

The Mystery Box Challenge

The OLED display on the compact box lit up after switching on the box, signaling the start of the challenge. The task was straightforward in concept but required precision and quick decision-making. To unlock the box, I needed to complete a series of interactive puzzles at multiple locations, each testing different skills. To start I had to find a specific location by using the only my distance to it, which was shown on the screen.

Once I was there first real challenge began as directional arrows appeared on the display, prompting movements with the joystick. As I responded, the game increased in speed, requiring more focus and accuracy. At one point, I misread an input, and a countdown timer appeared, providing only five seconds to correct the mistake. I adjusted my response in time, and an LED indicator illuminated, confirming successful completion of the task.

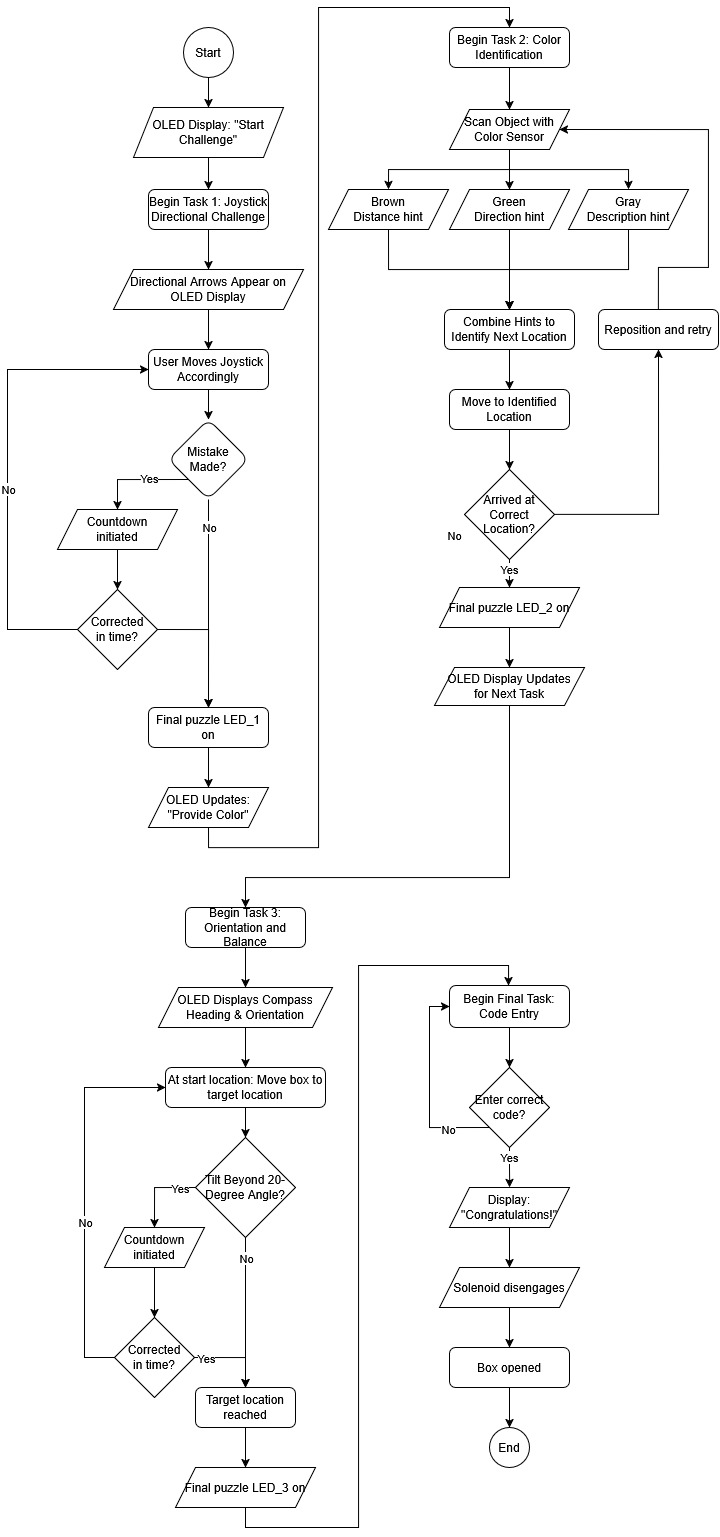
With the first stage complete, the display updated with the next instruction: **"Provide color."** Using the built-in sensor, I scanned objects in the environment. A brown object triggered a message displaying the distance to a target location. A green surface provided a directional hint, and scanning a gray surface revealed a descriptive clue about the location. These three hints, combined, guided me toward the next position.

Upon reaching the indicated area, the display changed again, presenting a new task. The screen showed a compass heading and a graphic representation of the box’s orientation. The goal was to carry the box to a precise location while keeping it level. As I moved, the tilt sensor tracked any shifts beyond about 20 degrees, triggering a five-second countdown if the box became unsteady. Careful adjustments were necessary to maintain balance, and after steady progress, I reached the target destination. A confirmation message signaled that I could proceed to the final step.

Now, the screen prompted for a code entry. Each previous puzzle had illuminated numbered LEDs, which served as the key. I entered the sequence using the buttons, ensuring accuracy. After a brief pause, a confirmation tone played, and the display showed: **"Congratulations."** A soft click followed as the solenoid actuator engaged, unlocking the side of the box.

The challenge was complete. The process had required a combination of reflexes, problem-solving, and careful execution. Each stage built upon the last, reinforcing the need for both speed and precision. With the box now open, the objective had been successfully met.

### Interaction flowchart



### Inputs

* **Joystick**: Used for the memory game, allowing users to replicate arrow sequences.
* **Color Sensor**: Detects different colors and triggers corresponding puzzle hints.
* **Tilt Sensor**: Monitors box stability for the balancing game. (Not visible)
* **Buttons:** Act as inputs for the unlocking sequence.

### Outputs

* **OLED Display:** Shows puzzle-related information, including directional hints, distances, and game progress.
* **LED Indicators:** Provide feedback on completed puzzles and display numerical clues at each location.
* **Solenoid-Actuator:** Provides final feedback by opening the door.

### Interaction and Changes

The level of influence each of the inputs has on the output modules depends on which puzzle the user is currently solving. This section will go through the respective interactions puzzle by puzzle. Information always visible on the OLED display is whether a GPS fix has been successfully acquired. This will be done through a “.” if there is no fix and an “!” if there is a fix.

If there is no fix a message asking the user to go a less obstructed area will be displayed.

The box is turned on through a power on switch and a message showing that the game has started will be displayed as soon as the GPS module has acquired coordinates. It will start showing the distance to a location and upon reaching it the first puzzle will start.

### Puzzles

#### The first puzzle:

The game utilizes a joystick, and an OLED display. The OLED display will continuously show arrows pointing in various directions, prompting the player to move the joystick accordingly. Each time an arrow appears; the player must quickly push the joystick in the corresponding direction. If the correct movement is made within a set time frame, the game progresses smoothly. However, if the joystick is moved in the wrong direction or if the player fails to respond in time, a countdown timer will appear on the display. If the player does not correct their input before the timer expires, they will have to restart from the previous stage. Successfully following a sequence of directional prompts will trigger a transition to the next puzzle. Throughout the game, an LED indicator will remain illuminated, signaling active gameplay.

#### The second puzzle:

This game utilizes a color sensor and the OLED display. Upon leaving the starting location the display will read “provide color”, upon showing a color the display will update with a corresponding hint. Each color will have two variables: display time and maximum number of scans. The colors and corresponding hints can be edited on the companion app, by default they will be set to:

* Brown (e.g. tree bark) – This will display the distance to the target location (30 seconds)
* Green (e.g. grass) – This will display a compass direction to the target location (1 minute)
* Off-white/grey (e.g. concrete) – This will display a hint about the target location (e.g. near a small pond) (15 seconds)

If the target location cannot be found after all colors are used up, the player will be given the option to reset from the start location.

#### The Third Puzzle:

It uses the tilt sensor, the GPS module, the OLED display and a compass. The OLED-Display will constantly display the direction to the target location from the current location in degrees. This changes if the user walks to a significantly different GPS location with the box. At the same time the display will show a graphic representation of whether the box is being held leveled or whichever side it is shifted to according to the information received from the tilt sensor. If the box is tilted by more than 20 degrees a five second countdown will appear next to the representation. If the box is not level again after this a message asking the player to go to the previous target location to restart the puzzle will be shown on the display. If the box is successfully delivered to the target location while keeping it level a message informing the user of the transition to the final puzzle will be displayed. The entire time a numbered LED will be turned on.

#### The Final/Overarching Puzzle:

It uses the OLED display, the buttons and the solenoid actuator. The display will show a message asking for a code after solving all the puzzles and reaching the final location. During each of the previous puzzles a numbered LED has been turned on either during the puzzle or upon solving it. Once the corresponding numbers have been entered using the buttons the display will display a congratulations message, and the solenoid will open the box.

## Companion laptop:

### Appearance and Layout:

The laptop interface is contained in a window.

### Sketch of the interface:

Immagine che contiene elettronica, computer, Dispositivo di output, computer

Descrizione generata automaticamente

Figure 2: Laptop Menu

Immagine che contiene elettronica, computer, computer, Dispositivo di output

Descrizione generata automaticamente

Figure 3: GPS location

### Inputs:

The inputs used are going to be the laptop’s keyboard as well as its cursor

### Outputs:

The outputs are going to be displayed in a window on the laptop screen.

### Interactions and changes:

For these interactions to work the laptop must be plugged into the microcontroller. Upon opening the application, the user gets shown a menu in which they can decide whether to set something in the microcontroller, look at the logged information from the run or just exit the application. If the logging option is chosen the menu changes to display the names of the things that were logged, which are GPS location every 10 seconds and temperature every 10 meters. The temperature will be displayed in a graph of temperature over time. The GPS locations will be shown on a map as a highlighted path. If the setting option is chosen the menu will display the option to set the location, the hints for each color of the color sensor game and the option to reset the pers which would make it possible for the game to restart. If the color setting is chosen for each hint the first letter of the color now used for it has to be pressed with each color only being able to be assigned once. In the location setting option the user has the option to set them by manually typing longitude and latitude of the target locations or to move a pin on a map using WASD for vertical and horizontal movement. The menus can be traversed using W for up and S for down. Selecting an option or saving a setting is done using the enter key. In every state of the interface the user can press b to go back, and new settings are only saved if they represent a valid input.

# References

*MoSCoW method* . (2025, March 21). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/MoSCoW\_method

*SMART criteria* . (2025, March 21). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/SMART\_criteria