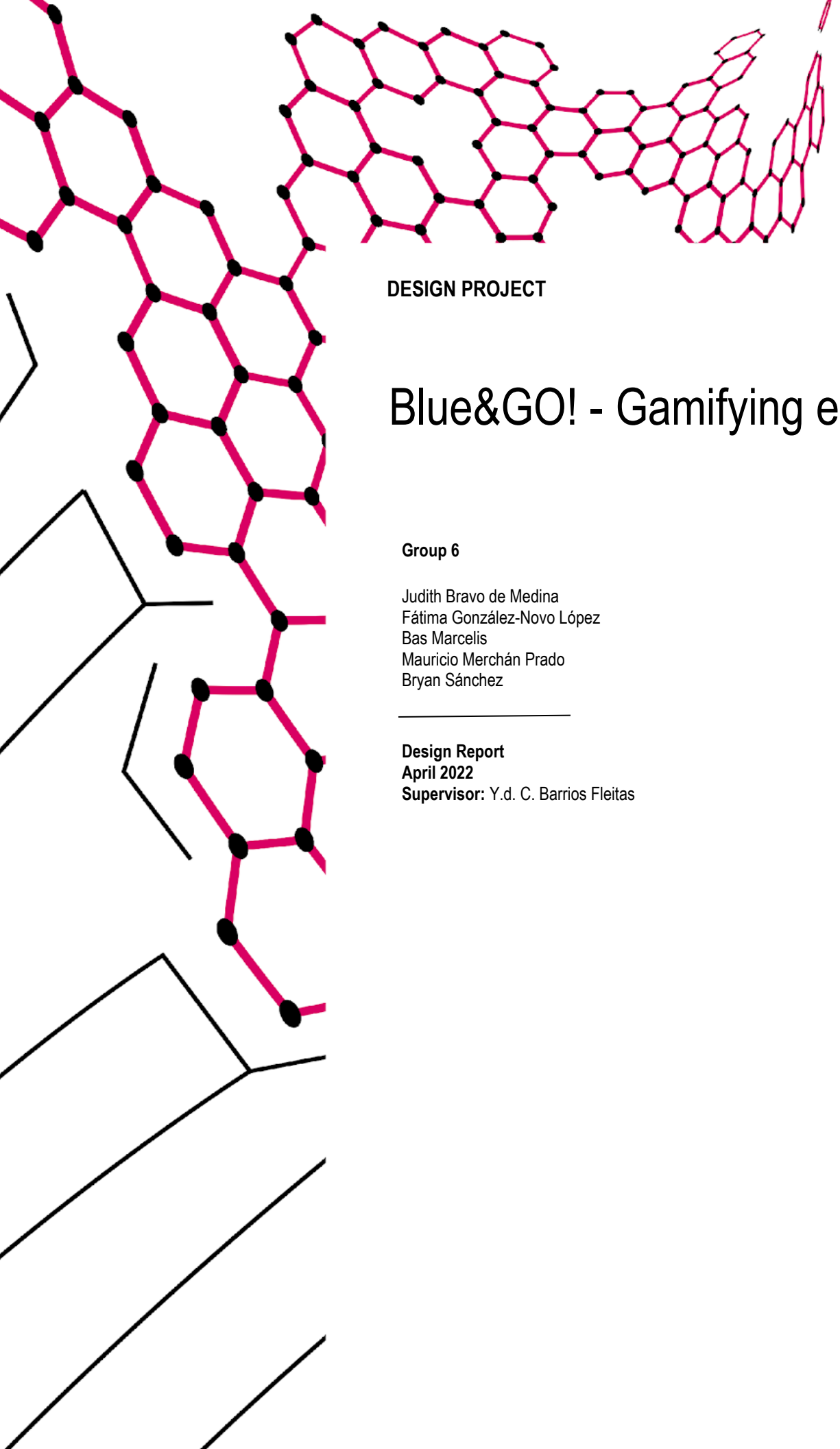


DESIGN PROJECT

Blue&GO! - Gamifying education

Group 6

Judith Bravo de Medina
Fátima González-Novo López
Bas Marcelis
Mauricio Merchán Prado
Bryan Sánchez

Design Report
April 2022
Supervisor: Y.d. C. Barrios Fleitas

Abstract

This paper has been written with the purpose of reporting and showing all the processes of the Blue&Go! Project. Moreover, there was a necessity to demonstrate the advances the team made during this period and to analyse the challenges the team faced and what would be the future work of the project itself. In this paper, we describe and analyse the current solutions within the field we wanted to cover, as well as what our stakeholders required to fill some needs they had while educating. In this way, the team discovered the importance of gamification and how it can improve education. Thus, Blue&Go! rises as an opportunity to help professors and teachers make their activities more interactive. It is done by the development of a webpage and a set of buzzers with a controller that works as one to create a completely interactive game where we aim to mix the physical interaction of buttons with the virtuality and engagement of a virtual platform with animations and sounds. First, the paper describes the current solutions for lecture gamification. These are the project solutions. Next, there are the functional and non-functional requirements from our stakeholders, and what is the planning by springs of each task of the project. Secondly, the team defined the preliminary designs for each component, such as the webpage, buzzers, and emitter. But also, the importance of such designs and how the system would interact. This gave an important overview of the actual design choices for the web interface, and the rest of the components. By the end of such sections, we succeed in having a set of functional buzzers that can connect with the principal emitter and interact together with the interface. It means a lot of hard work by the team that aims to achieve a complex project. Since it contains several parts: 1) the development of a web application that is responsive to the user's needs; 2) the 3D design of buzzers and their internal architecture and assemble. 3) the creation of protocols that unify the buzzers with the webpage and create an environment for learning and having fun. This project has important potential within education and in the way, it interacts with students and teachers. Different features can be improved, and others can be implemented to create a more engaging game.

Contents

Chapter 1

Introduction

Education is one of the most important contributors to the development and growth of an individual [6]. It provides knowledge and skill, which can later be used in a professional setting and can significantly contribute to how daily tasks are approached. Nevertheless, it cannot be said that learning is something that all individuals enjoy. When dealing with educational explanations, it is often seen in classrooms the confusion of students, which can later lead them to a loss of motivation. Gamification of education; a trend that is becoming more popular year after year, promises to motivate students while making them more engaged in the subject matter [7].

Besides the motivational stimulus, gamification allows learners to improve their class interaction, has a positive effect on their knowledge retention and enhances social and practical skills like problem-solving, collaboration, and communication [8]. During his TEDx talk, Professor Scott Hebert affirmed: “the education system is a system that is totally in an engagement crisis” [9]. Teachers tend to seek the advice of everyone aside from their students, without understanding that they are the second biggest untapped resource in education. During lessons, students do everything in their power to entertain themselves, but how can we blame them? At the end of the day, nobody is bored when making something beautiful or discovering new things, and that is what education should be, a place of discovery.

In this report, the proposal of a tool that aims to gamify education, in an attempt to eliminate boredom from lectures, will be explained. After analysing the current market and looking into the most famous educational tools, the proposed product design and the needed development of both software and hardware components will be described. Afterwards, an explanation of the web interface design, development and testing will be conducted.

1.1 Domain Analysis

Gamifying lessons is one of the many ways of eliminating class boredom. It promises to improve the overall development of students and its impact can already be observed from the use of popular tools in lessons. Nevertheless, there is still room for improvement. Current gamifying educational tools require an extremely large amount of preparational time to function properly. Teachers need to login into the platform, prepare questions in advance, and think of possible correct and incorrect answers. Due to all the benefits that gamification promises to bring, many professors still consider this time worth it for the sake of their students. But why should we stick with current methods when more efficient solutions can be developed?

In this section, an analysis of the most used educational gamifying tools already in the market will be conducted, with an emphasis on the problems that each platform is trying to tackle, the benefits that they apport and their weaknesses (what they lack or could be improved).

1.1.1 Existing Solutions

There are several game-based learning platforms that teachers use for interacting with their students. The most well-known ones are Kahoot! Quizlet, Quizalize and Wooclap. All these platforms share something in common, which is that teachers or any leading user must create the quiz questions beforehand, and it is what we consider to be the main shortcoming that the platforms face. In the table below, you can see the focus, weaknesses, and strengths of each platform.

Platform name	Engagement	Preparation time	Physical
Kahoot!	High	High	No
Quizlet	Medium	High	No
Wooclap	Low	Low	No
Quizalize	High	Medium	No

Table 1.1: Analysis of the market

The time that it takes to prepare a quiz varies per platform and per feature (for example, it takes longer to prepare a multiple-choice question than an open format one) but is considered by educational teachers to take too much time. Coming up with questions might seem relatively easy, but finding incorrect answers for those questions, which should still hold a grain of truth to mislead students, is much harder.

As you can see in the table, each platform has its own distinct identity. In other words, we can consider that each platform has its game mode. For example, Quizlet allows the students to join teams to play by matching terms and definitions, and the teacher can display the progress. Quizizz and Quizalize are similar since the students can play individually using their smart devices where they see each question with clickable options. But also, there is Kahoot, which differentiates itself by not having the questions on the student devices.

As it is observed, there are already services that provide resources to provide enjoyable gamification of education. However, they all have some limitations. Some require a high preparation time, others lack animations, some are hard to understand and most rely on the use of non-physical components. Educational tools should not just focus on either the teacher or the student, they should focus on all users equally.

1.2 Requirements Specification

To develop a more efficient tool to gamify education, the perspectives of the involved stakeholders need to be analysed. Product requirements and objectives can only be formed after understanding their needs, thus, the aim of this section.

1.2.1 Stakeholder

As stated in previous sections, the current focus of this project is to use gamification in classes at universities and, therefore, the main stakeholders to be taken into consideration are those in a college environment. For a better understanding of their needs and the benefits that can arise from this project to the different stakeholders, the power-interest (Figure 1.2) and the onion diagrams (Figure 1.3) will be used to help in the analysis of each group.

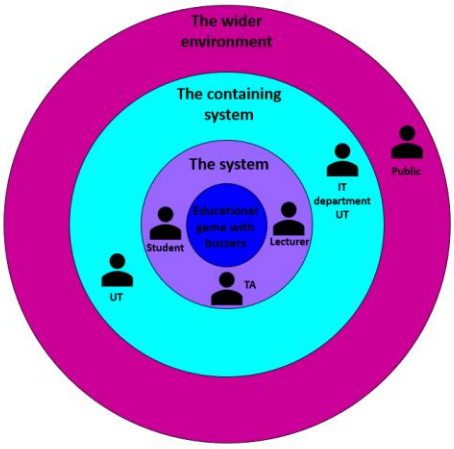


Figure 1.2. Stakeholder Onion Diagram

The System

Stakeholders involved in this section will be directly interacting with the game, either playing it, controlling it or helping out throughout the session. As can be observed in Figure 1.2. There are three direct stakeholders in this first layer:

Stakeholder	Power-interest
Lecturer	The ones in charge of giving the lectures are the ones taking the initiative to gamify their classes and they are, therefore, the prime stakeholders of this project. Thus, their power and interest in the project are high. They have the power to specify which features would not be helpful for them, which ones they would like to use, and how they can be as helpful as possible for them. Furthermore, the high position in interest lies in the fact that if they are the main target of the game, the ones in charge of gamification of their classes, thus, the ones using Blue&GO!. This group should be managed closely and cautiously, following as much as possible their specific requests, to optimise their experience so that they are willing to use the product.

Student	Attendees to the lectures and target group to use the system. They have a high power-interest balance, and the influence of their decision on which features, and settings are preferred is high, considering they would be the ones interacting and playing the game. Hence, if the features deployed on the game are not enjoyable enough for them, they would not be willing to use it.
Teacher Assistant (TA)	As part of the teaching staff, TA's power-interest scale is placed in the middle, neither low nor high. They should be kept informed about the evolution of the different game features, to consider what they, both as students and teaching staff, would be interested in and what would make it engaging for them to play the game. As they directly engage with the system and take part in the evolution of the lecture class, they can give new insights into the design and features of the system. Therefore, their decisions have enough influence to be considered.

Table 1.4. System Stakeholder definition

The Containing System

This second layer will be composed of indirect stakeholders, these won't be directly interacting with the system, however, they are in some way, or another affected by the use of the product. Described below are the two stakeholders involved in this level, based on the University of Twente since that will be the main institution focused on the development of the project, as was stated in previous sections.

Stakeholder	Power-interest
University (UT)	In high-level education institutions in which the lectures take place, the University buys the product, so they have to be kept informed about the development of the project/game. As can be observed in the diagram above, the interest of the university is high, as they are the ones that can decide to purchase the system for their use in the lectures of the different faculties. Therefore, their interest resides in the educational usefulness of the game, as well as how important they observe the different features to be for the educational purposes of the lectures.
Service IT Department (UT)	The IT department is in charge of any technical difficulties encountered with the development of the game and the connections with the university network. The interest-power the IT department has in the game is intermediate, leaning towards low since they would not be interacting with the system as players or executors do. Therefore, the focus is not so much on making them interested in the functionality of the game. However, they would be the ones managing the connections of the university network needed for the game, solving problems lecturers may have while using the game, and more.

Figure 1.5. Containing System Stakeholder definition

The Wider Environment

Lastly, we have the indirect stakeholders that are not directly affected by the utilisation of the game, but they do indirectly benefit from the product, as viewers or third parties.

Stakeholder	Power-interest
Public	People in this group will be considered as those in the class that would not be participating in the game, thus not being directly affected by the course of the gamification system. Hence, the public's power over the decisions to be taken in the project is low since they would not be interacting with the system. However, their interest in Blue&GO! is slightly higher than zero. As the public is an indirect stakeholder and is observing what is happening during the play of the game, it should be interactive enough for them to keep observing the interaction of the students and professor with the game.

Figure 1.6. Wider Environment Stakeholder definition

Icon	Stakeholder
	Lecturer
	Student
	Teacher Assistant
	University
	Service IT department
	Public

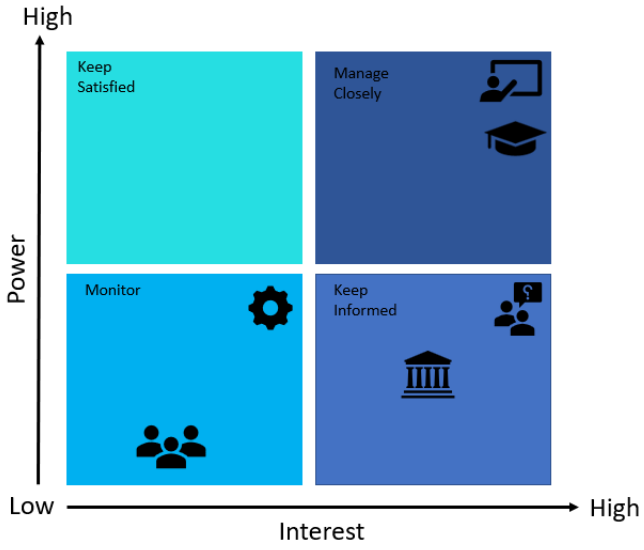


Figure 1.3. Icon reference and Stakeholder power-interest grid

1.2.2 Functional Requirements

The following section will give a detailed explanation of the functional requirements following the MoSCoW strategy to have a more complete view of which requirements should be taken into consideration in the earlier stages of the project development and the following development of it. These requirements have been built taking into account the stakeholders' needs.

Must

- The teacher must be able to initiate a new game through the web interface.
- The teacher must be able to visualise the players' progress within a web interface.
- The user must be able to identify the states of the game.
- The system must count with a web interface.
- The system must have synchronisation between the student buzzers and the emitter.
- The system must be safe to use for children.
- The system must safely store the gathered data.
- Every buzzer must have light states.
- The buzzers for students must be connected to an emitter system.
- The buzzers must generate a light state according to the game.
- The buzzer should have a power system.
- The interface should display the points awarded and players' progress.

Should

- The system should recognize which student is using which buzzer.
- The system should have a game settings interface, to modify features (e.g., the number of points awarded).
- The system should have a controller for the teacher connected to the interface.
- The system should have animations among the states of the game.
- The system should reproduce sounds throughout the game.
- The buzzers should have a charging system (batteries/cable).
- The interface should be responsive to different devices.
- The teacher could be able to modify and change the game settings (game mode, points awarded, time to reply…).

Could

- The final ranking could also be displayed as a canvas plugin.
- The teacher could be able to download the result of the game.
- The buzzers and controllers could be charged.
- The buzzer could have 4 buttons to allow the possibility of multiple-choice questions.
- The players could earn or lose points according to the teacher's decision (either correct or wrong answer).

Won't

- The system will not have the ability to display questions and answers on the web interface.

1.2.3 Quality Requirements

- The system must respond to the game rules.
- The system (Raspberry-based emitter and receiver) must be portable.
- The system could have the ability to divide the players into different teams, and rank points individually and per team.
- The system could track progress per period (the progress of each game is saved for each student).
- The Raspberry-based emitter must be responsive to the interface.
- The connection between emitters and receptors must be synchronised and constant.
- The time response of each buzzer should be up to 500ms.
- Student buzzers could have the possibility of wireless charging.
- The interface should be user-friendly and minimalistic to improve the user experience.
- Sounds and animations must engage the user's attention.

1.3 Solution Proposed - Blue&GO!

Based on all of the factors described above, it is clear that there is a need for a more efficient solution to gamify education. In this report, the use of Blue&GO! will be proposed, a physical tool that, like current educational platforms, aims to make students more interested in the learning experience by making lessons more enjoyable. Contrary to the existing tools, Blue&GO! aims to reduce lecture preparation time by building a tool that can be used on the go, therefore becoming a more sophisticated and well-rounded educational game.

Chapter 2

Planning

2.1 Sprint Planning

The planning and distribution of the project tasks and deliverables will be based on the agile scrum methodology. In this way, the deliverable project would meet and satisfy the needs of the customer. Moreover, the scrum team can be more communicative and dynamic through meetings, since the main objective of this methodology is to keep resolving problems among all sprints until the project meets the requirements. This method also encourages to show scrum values that are important for both the project and the team itself. These values are courage, focus, commitment, respect and openness; since it is a flexible methodology, each team member can work and exploit their skills and roles.

The distribution of the sprints has been displayed in Table 2.1, for each of them a sprint review will be presented at the end of it, making a balance of what went well and what should be fixed for the next sprint. Besides, for the majority of the sprint's deliverables will be presented to the Product Owner, to show progress on the development of the product and to obtain feedback on the already created features. For a detailed explanation of the sprints, their duration and the goal of each one, refer to Table 2.1.
Moreover, weekly meetings will be scheduled between the Product Owner and the Design group to discuss the progress of the project during the week, problems encountered and incoming ideas among other things. Additionally, the course of the project and the different features created will be documented throughout the whole project, providing drafts with every sprint to the Product owner to receive feedback, suggestions and future work-related sections.

Sprint	Sprint Duration	Sprint Goal

Sprint 1: Getting Started	**1 week** 7th - 13th February Deliverables: 2nd March → **Project proposal** 8th March → **Final Design**	Understand the project idea and start developing a project proposal
Sprint 2: Design	**2 weeks** 14th Feb - 6th March	Finalise the design of the interface and the game (lo-fi prototype). Finalise Project Proposal. Start researching the use of the Raspberry integrated into the project
Sprint 3: Development and creation of the product	**4 weeks** 7th March - 3rd April	Programming the game, web interface, system and inner connections.
Sprint 4: Testing	**2 weeks** 4th - 17th April Deliverables: 4th April → **Draft Design Report**	Finalising the game, and general Testing of the whole system. Developing Poster, finalising documentation.
Sprint 5: Closure	**1 week** 18th - 23rd April Deliverables: 18th April → **Poster** 29th April → **Working product and Design report**	Closure. Presenting the working product.

Table 2.1: Sprint Overview

2.1.1 Sprints

Following Table 2.1, this section will describe more in-depth each of the sprints and the deliverables to be handed in during each of the sprints. The total completion of the project is scheduled for 10 weeks divided into 5 different sprints guided by:

Sprint 1: The first sprint will be the kick-off stage of the project, getting to understand the project idea better, establishing the first project meeting with the Product Owner and understanding the idea of what the project should look like at the end of the last sprint.

Sprint 2: The design phase of this project will be held in two weeks, focusing on developing a lo-fi prototype of the web interface, buzzer and the game itself. Furthermore, interviews and surveys will be conducted in order to obtain a more in-depth perspective of what our stakeholders would like to interact with within an educational game, as well as already established insights that they may have of current similar products in the market. With this research and the design of the prototype, a project proposal will be developed and handed over to the Product owner in the middle of the sprint so that it can be discussed and adjusted to his idea of the project.
Lastly, during this sprint, the research on which materials may be needed for the project and how to use them (e.g., Raspberry pi, buzzer connections, etc.) will be held too so that it can be understood by sprint 3.

Sprint 3: The intermediate sprint will be the longest one, focusing on the development of the product, programming and creating the web interface (front-end), the internal connections (back-end), creating the buzzers, teacher controller and programming the raspberry pi's. Furthermore, testing for each of the components will be done regularly during this sprint, before going into the general testing in sprint 4.

Sprint 4: The first phase of this sprint will be testing the product with the stakeholders and testing the overall design with the participants, proving that the game satisfies the needs and perspectives of future users. Moreover, general testing of the whole system will be held during this sprint, fixing errors, and bugs and perfecting the complete game.
A poster for the final product will be created during the second phase of this sprint.

Sprint 5: For the last sprint of the module a presentation of the product should be held, including a poster and an MVP. Besides, the final documentation including the Design Report should be finalised by the end of the sprint, closing the development of the project.

2.2 Task Distribution

Once the definition of the project has been made, the general tasks to be accomplished were distributed following the management presented in table 2.2, these were the responsibilities assigned to each team member to be followed until project completion.

Team member	Responsibilities
Bas M.	Front-end; final presentation (assistant)
Bryan S.	Back-end; Hardware; Server; user testing (taking notes); final presentation (assistant);
Fátima G.	Front-end; Mock-up; final presentation (pitch)
Judith B.	Back-end; UX/UI designer; user testing (interviewer); final presentation (pitch)
Mauricio M.	Back-end; Hardware; user testing (taking notes); final presentation (assistant);

Table 2.2 Team tasks distribution

Chapter 3

Global Design

Since there is an expectation of interaction by the students within a class or lecture, Blue&GO! provides buzzers that the students can use to interact directly with the lecturer whenever he asks for it. The idea is that the teacher/lecturer would have an emitter that would host such buzzers, thus teachers can control when the students can participate and whether their answer is correct or incorrect. To register students' participation, the buzzer contains a card reader sensor that can be used to identify a group of students with a button. Thus, the teacher can keep track of the participation of all students in the whole course or module.

Focusing on the idea of having to register the participation of the students, the system requires the management of a database where the teacher would be able to register his courses with their corresponding number of students. As with the other platforms, there is a need for a web application where the teacher can sign in or log in to their account. In the web interface, the teacher has access to modify and create new courses linked to their account. This web interface should have a friendly and interactive interface where all the components and the page itself are interactive. This front end would have a landing page that displays general information about the game, its components, and the different game modes. Following that, there is a page where the teacher can import a CSV file to create courses, containing sets of students, which will be used to determine playing students in a game. These courses can still be modified later. There is also the game initialising page, where the teacher must select a game mode and a course to play with (or choose anonymous gameplay). This directly follows to the final page, the game playing page, where visual feedback on game status and scores of students is given.

3.1 Global Design Choices

3.1.1 Key Design Pillars

Before getting into the design specifics of Blue&GO!, it is important to determine key design pillars, which must be respected throughout the design process and on which the design will be based. These key design pillars are, therefore, represent the most important values that Blue&GO! must provide. The pillars are 1) Enjoyment, 2) Ease of use, and 3) Flexibility, which is in order of descending importance. In this chapter, it is discussed how these pillars shaped the global and architectural design, and in Chapter 5 it is discussed how they help shape specific design choices to further optimise them.

Enjoyment is the most valuable feature that Blue&GO! must provide. Blue&GO!'s primary objective is to optimise the learning process, which is achieved by increasing enjoyment through gamification, which enhances student participation and focuses as well as results in improved learning. Therefore, it is critical to keep in mind that most aspects of Blue&GO! should be fun and playful. This is reflected in using physical buttons with visual and audible feedback, which is more fun than using a mobile application, as well as in using an additional front end, which provides the option for even more colourful and animated feedback during the game.

Secondly, ease of use is of high importance. While the main objective revolves around optimising students' behaviour, it is in most cases up to the teacher whether certain educational tools are used, and it is, therefore, important that their experience with Blue&GO! is positive. To achieve this, ease of use is taken as an important value, to optimise teachers' experience and to increase the prevalence of Blue&GO! among educational environments. The most important feature in this domain is how the game is structured, and that it can be started requiring almost no preparation time at all, which increases time efficiency for teachers. Additionally, this ease of use for teachers is reflected through a wireless controller with few, but clear buttons which can control the game. For more complicated actions, such as changing settings and managing stored data, the web interface can be used. Here, again, ease of use and intuitive design is emphasised. In addition to teachers' experience, ease of use is also important for students, as ease of use can improve enjoyment, referring back to the first pillar. Ease of use for students is reflected in the physical

design of the button, with a big button on top and LED lights to provide feedback on the status of the game.

The last pillar regards flexibility. The flexibility of Blue&GO! is important as it can make its use applicable in more scenarios, which can further increase its prevalence. This flexibility can be seen in the way that physical buttons are integrated with the game. The buttons and the emitter are needed to host the game, and are, therefore, not limited to the classroom but can be played in multiple environments. Additionally, the use of buttons uses speech to ask and answer questions, which increases the flexibility of how questions and answers can be phrased. For example, providing the possibility for both open-ended questions, as well as multiple-choice questions.

3.1.2 Established In-game Workflow

The two most important processes in Blue&GO! regard the in-game experience and managing and storing courses in the web interface. The second is a more straightforward process, but the former needs some clarification on which steps need to be taken and in what order. Therefore, a workflow diagram is provided to illustrate what processes are expected to happen in which order to play a game, from initialising the game to ending the game. The workflow starts in the left bottom process, where the teachers select a game mode and course in the web interface.

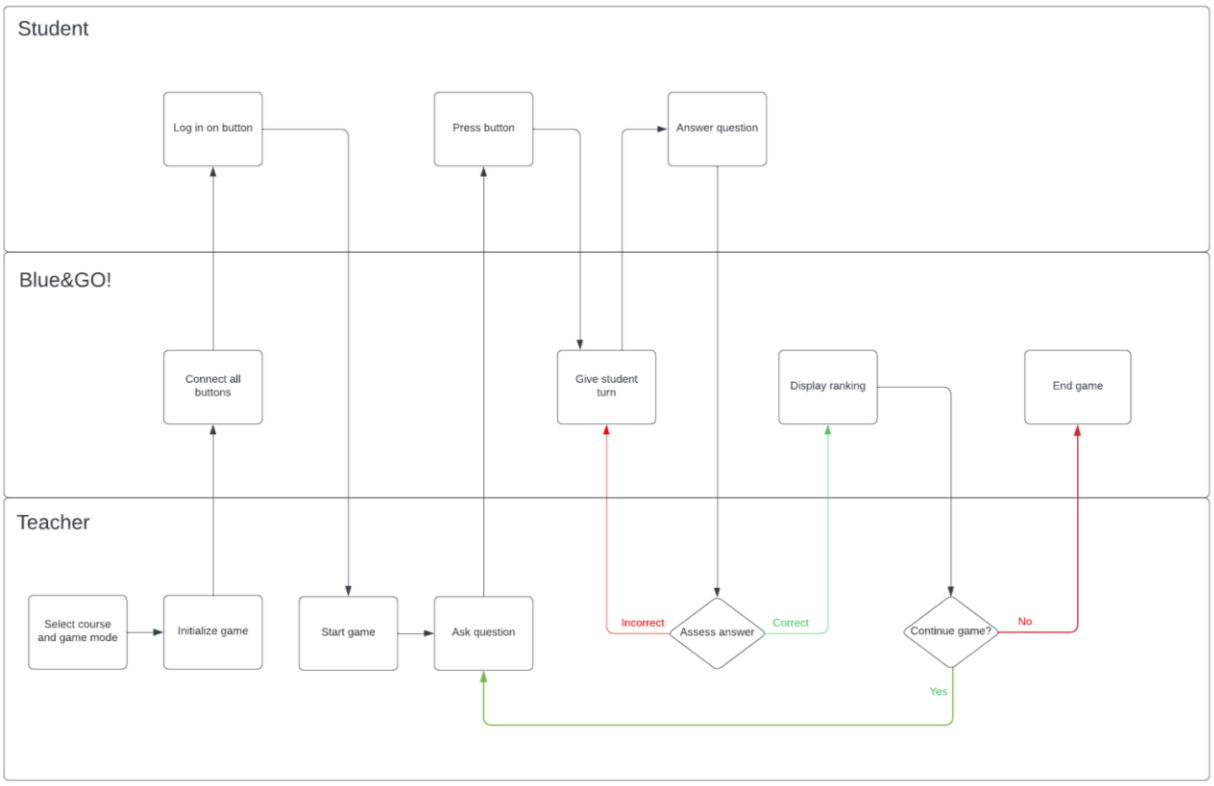


Figure 3.1. In-game workflow

3.2 Preliminary design

3.2.1 Programming Language, Frameworks and Libraries

3.2.1.1 Front-end

The interactive front end is provided using the usual languages HTML, CSS and JavaScript. HTML is used to create and structure components, CSS is to style these components, and JavaScript is used to generate relevant content and interact with the backend using RESTful services (CRUD). To improve the design of the front end by using predefined classes, the library Bootstrap v5.0 was used. Additionally, imports from LordIcon [1] and Bootstrap's icon library [2] were used to provide animated- and non-animated icons, respectively. Both Bootstrap and LordIcon were used because of their fewer cross-browser bugs and their compatibility with different browsers.

3.2.1.2 Back-end

For hosting the web interface, the flask environment was installed on the raspberry pi. It is because of its flexibility, the tools it provides and its little dependency on external libraries. This framework can be used to host applications and handle HTTP requests with Python which is the preferred one for using with the raspberry pi. The templates mentioned above are stored in the same directory as the flask server. There are two main Python classes that handle the user interface: Routes and Script. The first one is the class that contains the URL paths, and the other is where the main functions are. These functions are for fetching or updating information in the database or the protocols that control the game itself.

3.2.1.3 Emitter (Raspberry pi)

For controlling the Bluetooth connection of the Raspberry Pi. A system update had to be done, together with several libraries that helped to manage the connection itself. The principal library used is PyBluez, which is installed using the pip command within the terminal of the OS. This library was the most completed library among the others, but there is no documentation and functions are not developed yet. However, the functions are sufficient for this project. Additionally, PyBluez also needs GNU dependencies such as Python 3.5 or newer, Python distutils, and BlueZ libraries and header files (libbluetooth-dev).

3.2.1.4 Buzzer (ESP32)

The ESP32 has been decided as the core of the buzzer for two simple reasons. 1) it is more compact than an Arduino 2) it already contains Bluetooth on its own. Additionally, it is also compatible with Arduino IDE which makes its use of it easier.

Since the ESP32 is a low-power system that works with the Arduino software, the board managers should be implemented through 2 links of JSON files that also allow you to download the board of the ESP32 and use the device within the Arduino IDE.

The connection between the buzzer and emitter is through Bluetooth. For this reason, there is a library called BluetoothSerial.h that is compatible with the system but also controls the Bluetooth connection and interaction of the device.

3.2.2 3D modelling pieces

3.2.2.1 Modelling

For modelling the pieces, the SolidWorks software was used throughout this project. The prototypes had to be made accurately due to the fact that the electronic parts should fit with precision. Hence, the first task was to take the dimensions of every single component that the final product would have. Although, two millimetres were added to all measures as a margin error since the 3D printer is at the Design Lab at the University of Twente.

3.2.2.2 Assembly

After modelling all pieces, the assembly designs were also made with the same software mentioned in the above section. This was necessary for constructing and evaluating the design of the prototypes. In that way, the assembly could be made digitally before 3D printing.

3.3 System Overview

3.3.1 Web-interface

The web provides two main functionalities: 1) Starting and playing a game, and 2) Managing and storing data.

As for the first functionality, the front end can be used to manage most of the game features. Starting a game on the web interface should be intuitive (as will be further discussed), and, after it is selected, can be used to initialise the game. A predefined course from the internal database or anonymous gameplay and one of the three-game modes must be selected before initialising the game.

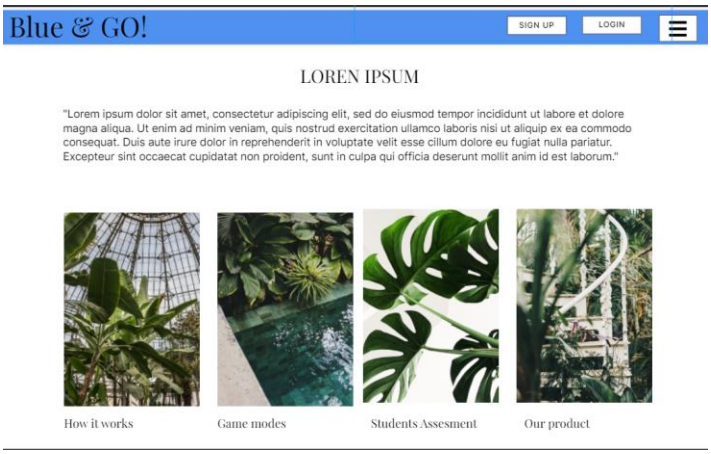


Figure 3.2. Landing page Mock-up

Followingly, an intermediary screen displays which buttons are connected and the status of the teams (if they are logged in/checked-in or not) before starting the game. When the teacher continues, the game will start, and the front end will provide visual feedback on how the game is going. This includes countdown timers before questions, names on whose turn it is, display (in)correctness of answers selected by teachers, and rankings in between questions. In between rounds, there is also the option to end the game or go to the next round using the web interface, this is provided by both the web interface as well as the physical controller.

The second functionality regards the managing, storing and reviewing of data that is stored inside the internal database. Logging in is required to get access to the data, which provides access to account-related data only. Logged in users can create and manage courses, which are sets of students used to play the game. This management includes, for example, adding or deleting students. Such courses can be created manually, name by name, but can also be imported using CSV files. Additionally, logged in users can review and download scores of previous games, which can be used to review rankings among students in the same course.

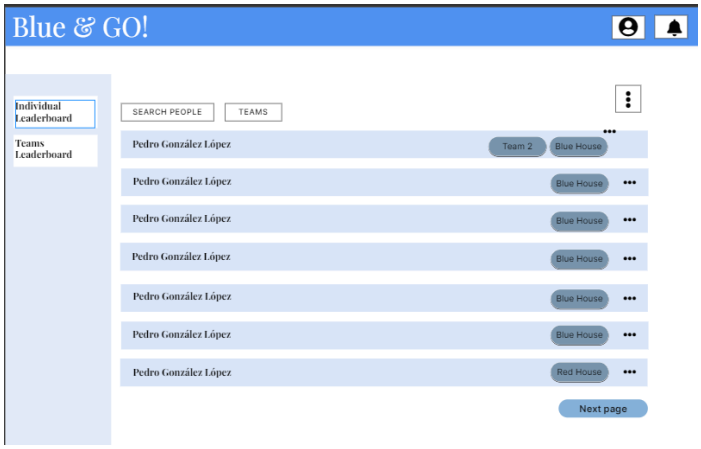


Figure 3.3. Course description (class) page Mock-up

Lastly, the web interface provides some additional features related to user experience. On the main page, there is an explanation of how the game can be started and how it can be played. Additionally, there is a FAQ page and a contact page when users who need help.

3.3.2 Hardware

There are two principal components that form the hardware, which are the buzzer and the emitter, both containing more features that allow the system to work ideally, according to the game specifications.

The emitter is a Raspberry Pi that will achieve two principal functions. It will be the server of the web application where the lecturer or teacher can initiate the game and control its configurations. In addition to that, there is an IR (infra-red) remote controller that the teacher would use to control certain steps of the game, such as initiation, finalisation, and deciding if an answer is correct or incorrect during the game. Moreover, it will be the host of the buzzers.

On the other hand, there is the buzzer, which contains several components within its 3D printed case. There is an ESP32, which is the core of the buzzer since it is a low-power CPU that manages the rest of the components. The ESP32 is powered by a battery of 10000 mAh. There is a wireless charging transmitter that helps the battery be charged. For the student card reading function, the buzzer contains an RFID sensor that is in charge of this function. Moreover, a led strip light will be at the top of the button that lights up the buzzer according to the state of the game being played.

Chapter 4

Design Specification

4.1 Design Choices

The following chapter describes the different design choices that were made throughout the design of Blue&GO! This chapter is divided into three parts: 1) Web interface, which includes design choices regarding the interactive part of the front and regards storing data and optimising user experience, 2) Controller & Buzzer, which regards the physical components of Blue&GO!, 3) Gameplay, which discusses design choices while playing the game.

4.1.1 Web interface

For the development of the web interface, some design choices were made, based on the feedback provided by our main stakeholders (professors). The web interface includes both the landing page with information about Blue&GO!, profiles, courses and scores, as well as it includes pages to accompany playing the game, which include rankings and giving turns. In this paragraph, design choices for keeping track of data and user experience will be discussed. However, the pages displayed while playing the game are not included in this paragraph, they have their own in 5.1.3.

Enjoyment

Because Blue&GO! the main goal is to use enjoyment to enhance the learning process, this enjoyment must also be reflected in the front end. Therefore, the front end must be easily understandable and have an intuitive design, to optimise the user experience. User testing contributed to the design for this user experience optimization. Functions that the front end provides are quite limited, it only keeps track of data (game data, course data etc.) and it provides the display while playing the game. The aim of the front end is, therefore, to simplify the design in most places to make the front end easier to use and cleaner. To further increase user enjoyment,

the front is made colourful, using a palette consisting of shades of blue with a hint of yellow, and dynamic, using animated and non-animated icons.

Implementation choices to enhance user experience can, for example, be seen in the design of the landing page. It was doubted if educational teachers would grant the same importance to modifying course settings compared to starting a game. Nevertheless, the results of the user testing revealed that teachers give both the same importance. Thus, to simplify the process as much as possible and make the website functional, the landing page was made with only two buttons of equal size that would lead the user to these two sites. The button to modify the settings of a course was first given the name “classes”, but after interviewing several professors, it was decided to be renamed “courses”, as that is the term used by them. Multiple small changes like the one above state were made to make the website completely understandable and intuitive.

Log-In

One of the things to consider when building an educational tool is how users will be able to log in to the site. This process should not only be secure but also easy for the users. That is why, it was decided to use the easy login features provided by Microsoft and Google, as these companies are the main providers of educational email accounts. Additionally, educational games should be easy to set up. Reducing the number of pages was therefore a big priority when designing the Blue&GO! website.

Storing Data

When educational games are being played during lectures, they are often played by large quantities of students. And as Blue&GO! is played using students' names and scores linked to them, it is important that such data can be stored. Therefore, a database is used to store courses, and of which students or of which groups of students they consist of. This simplifies starting the game and limits the time to prepare the game, as the teacher has to enter the set of students in a new course in the system only once and can easily select it later to continue playing. Not only this, but it provides the ability to keep track of scores of students or student groups over longer periods, which can be accessed and reviewed using the front end as well.

The process of adding the list of students that a course has before playing a personalised game can be quite tedious. With one of the objectives of Blue&GO! by reducing the preparation time of lectures, we needed to make this process as simple as possible. That is why the feature of importing lists of students through CVS files and making teams by pressing a single button on our page was created. With this design, an educational teacher can play a personalised game in just a few minutes. Moreover, the user testing round revealed that educational teachers gave importance to saving scores after every game, thus, the feature of downloading all stored information in multiple formats (.cvs, .pdf, .word…) was added.

Game Settings

Before initialising and starting a Blue&GO! game, some settings have to be determined, like which course to play with (or play anonymously), which game mode to play with, and what the settings for that game mode are. Initialising and starting the game is done using the web application, as it provides easier access to the settings and more possibilities to select and change settings. It also provides an easy way to oversee all buzzers and see if they are all properly connected before starting the game.

4.1.2 Controller and Buzzer

To develop Blue&GO! physical components and to respect the previously stated requirements in chapter 3, many design choices had to be made, which will be discussed here. The physical components include the students' buzzers/buttons, the Raspberry Pi which controls the game, and the lecturer’s controller.

Student identification

The greatest difficulty in the design choices regards how students can identify themselves. This identification is required to keep track of scores while playing the game, which stimulates competition. The priority here is the ease of use of Blue&GO!, which requires the identification to be done quickly and easily. Because of this, many easy options like teachers filling in names on the web interface already become inappropriate.

The scope for the first design of Blue&GO! is limited to university students. Most universities have a student identification card used for, for example, identification during exams. This card can also be used for identification while using Blue&GO!. To use this way of identification, the button will use a card scanner to scan the student card to identify the playing student. This will further be discussed in the button design.

Button design

To easily store the buttons, the buttons will have a cube shape so that they can easily be stored inside a storage unit. Buttons should be small enough so that many fit inside the storage unit, but not too small. It has been designed in such a way that all internal components are aligned and compacted, thus the dimensions of the housing and the push button are ideal in a way that 10 of these can be stored in one big case.

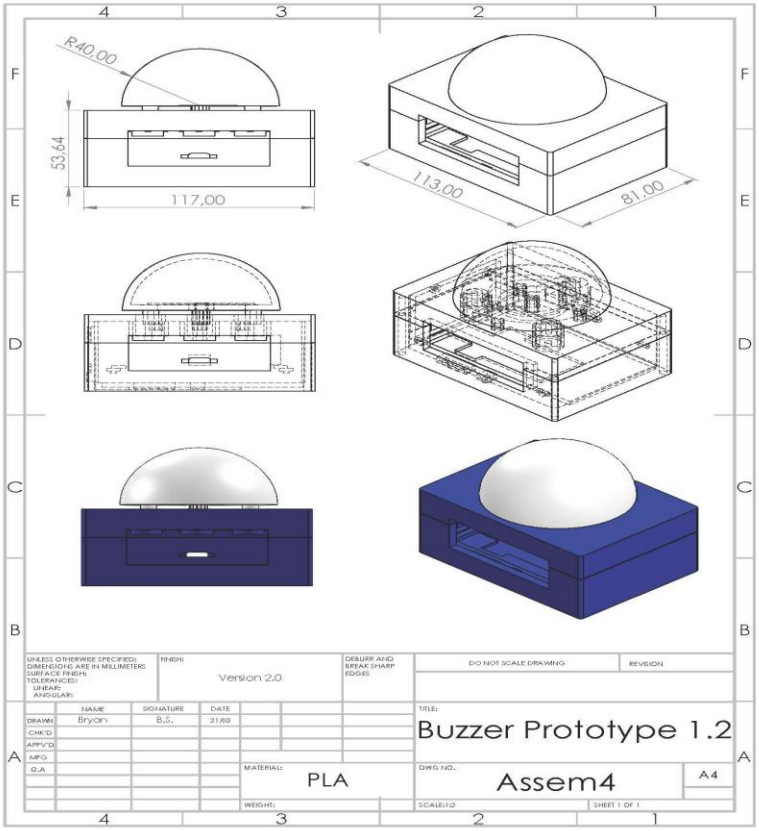
The button will include a scanner to read student cards for identification, as previously mentioned. A small screen will also be attached to the front of the button. This screen will provide small bits of game information, like which team is playing with this button and how many points they have, people joining the game, people joining groups etc. Moreover, the special button and the on/off switch are placed at the bottom so they cannot be touched by accident by the end-user.

The housing of the button was divided into three parts, the top one is the spherical push button and was designed to be transparent in order to let the user see clearly the intense light from the RGB led strip inside of it. The middle and bottom parts were designed to be blue and non-transparent in such a way that it would hide the electronic components.

Such pieces were 3D modelled and assembled in three different prototypes. In the first weeks, two prototypes were made to evaluate different designs and perspectives of what the final product would look like. The first design had a cube shape with three pieces that would be attached with a clip system. This scale drawing can be found in Appendix E.1 and it was designed for storing a battery of 4000 mAh and the other electronic components that are mentioned before. However, this design was discarded as the final product should have a battery with more capacity, 10 000 mAh.

The first version of the second prototype can be found in Appendix E.2 and has a circular shape with the same specifications that the first version of the prototype has. But this cannot store an LCD screen, so it was modified in such a way that it keeps the circular shape, and the display can fit in the front. Additionally, this second version can store a battery of 10 000 mAh. Scale drawing of this version can be found in Appendix E.3.

The two designs were compared, and the conclusion was that the second prototype with the rounded shape needs to be discarded as it is too big, and this could result in an inconvenience for storing and carrying 10 devices with this shape. Thus, the prototype needed to be modified to store the new battery and be compacted at the same time.



Therefore, the length and width were increased due to the battery, and to have a compact design, the height was increased so the components would be placed above the battery instead of being at

the sides. Scale drawings of each piece of this final version of the prototype can be found in Appendix E.4.

Charging and Storage

The buttons will carry a battery that can be charged in two ways: wireless by the receiving coil that is placed at the bottom of it, and with a normal charger via the micro-USB port on the side of the housing. It will contain the corresponding modules that allow the battery to be charged in both forms. A briefcase would store 10 of these buttons and their chargers, so the end-user can take away everything they need for gamifying a lecture easily.

4.1.3 Gameplay

The last domain of design choices to be discussed in this chapter is Gameplay. While this is also a part of the web interface, the web interface part of this chapter deals with data storage and front-end interaction outside of gameplay, while this chapter focuses on the gameplay.

Game Modes

Blue&GO! intends to enhance the learning process by increasing enjoyment in learning using gamification. To optimise enjoyment, Blue&GO! will include settings to play multiple game modes for different ways of learning and for classes to choose their favourites. Examples of game mode settings included are: Give turns in order of how fast the button is pressed or random selection, give points to correct answers or use lifelines and deduct lives when answered incorrectly, and playing with groups or with individuals. Settings also allow smaller rule changes, like changing the number of points per question or the time students have to answer if the respondent is chosen randomly.

Notifications

To allow the integration of Blue&GO! while using other educational materials like presentations or pdfs, notifications were added. With them, games can be played without the website being presently shown on the screen, while still being notified of the leader board scores through notifications.

Chapter 5

Results

5.1 Testing

The Blue&GO! implementation will consist of Arduino-based buzzers, Raspberry Pi to host the game, a controller for the teacher and a web application providing general information about the score and the game. Consequently, there will be isolated testing for both the buzzers and the controller, the web application, as well as general testing for the product as a whole.

Isolated testing of both the Arduino-based buzzer as well as the controller for the lecturer will include many physical tests. The primary focus here is on safety and robustness. Blue&GO! will operate in an environment involving children, who can use the device, especially the buzzer, in unorthodox ways, for which the device must remain safe to use.

Feature	**Test case**	**Expected result**
Safety regarding electricity	Manual	Not being able to touch any wires
Robustness	Dropping from 1m height	Remaining functional
Robustness	Forcefully handling (for example, forcefully pressing the button)	Remaining functional

Table 5.1: Testing Isolated Buzzer & Controller

Isolated testing of the web application will mostly involve testing direct human interaction with the web application. This involves, for example, ease of use and making sure the application is

complete. Besides interaction, ways of handling data will be thoroughly tested as well. This includes the safety of stored data, but also if the correct data is stored at the correct place.

Feature	Test case	Expected result
Ease of use	Give test subjects assignments to fulfil within the application	Assignments can be easily and quickly completed
Data safety	Manual	No gaps in data security
Data storage	Random values as input	Correctly stored data

Table 5.2: Testing Web Application

Lastly, general testing will include the remaining tests. The previous isolated tests focused on specific components, while general testing focuses on the use cases of the product as a whole. This, first of all, includes correct scenario handling of the complete product, to make sure the game plays out as intended. Additionally, to optimise the learning environment, enjoyable gameplay is of very high priority. To achieve this, the ease of use and enjoyability of the game must be thoroughly tested. To clarify, ease of use will be separated into tests for lecturers, which include testing with the controller and web application, as well tests for students, who will be tested with the buzzer and the web application.

Feature	Test case	Expected result
Giving turns in the correct order	Pressing buzzers in random order	Give turns in pressed order
Handle simultaneous presses	Press buzzers simultaneously	Both buzzers turn blue, indicating they were pressed at the same time
Ease of use for the lecturer	Giving assignments to test	Over 80% of assignments

	subjects	must be passed
Ease of use for the students	Giving assignments to test subjects	Over 80% of assignments must be passed
Enjoyability	Test subjects playing the game + interview	Over 70% of test students must find the game enjoyable

Table 5.3: General Testing

5.1.1 User Testing

5.1.1.1 Survey Quantitative Research

During the early stages of this project, in the middle of Sprint 2, we carried out quantitative survey research to analyse what the end-users would like to see in an educational oriented game, as well as what was lacking in current applications such as Kahoot or Wooclap that would make the gamification of lectures/classes even more entertaining and practical for students to learn while enjoying.

Approach

We developed two different surveys, one for professors and another one for students (refer to Appendix A), however, we took different approaches for each of them. The students' survey was spread and sent to as many people as possible (students), with a total number of responses of fifty-seven, which gave a complete insight into what students were interested in and what they wanted to have in an educational game.

On the other hand, we wanted to have a one-to-one conversation with professors so that they could give us a more complete picture of their experiences with educational games and what they would add/remove from such platforms for them to be even more practical and them being willing to use it. We managed to conduct three face-to-face online and two email-based interview responses from professors, which completed the perspective of the requirements and what the focus should be for the implementation of the project.

Results

Analysing the results obtained from the surveys, it was concluded that 94.6% of the students consider educational games to be enjoyable, with more than 70% assigning them more than an 8 out of 10 on a priority scale. Furthermore, out of the 57 participants, more than 50% do not have a formal opinion about whether it is important or not to see the question displayed on the screen, and more than 30% would still use a game where the questions are not shown in the lecture slides or game interface. Following the importance of displaying questions, around 85% of the students consider educational games a motive to improve their attention during lectures, helping them towards their learning goals.

Moreover, one of the most important factors to consider when designing our web interface and the dynamics of the game are the details that make the educational system enjoyable/annoying for the students, as well as which new features they would like to see in a new educational game. Presented below in table 5.4 are the positive/enjoyable and negative/annoying features considered by the students and marked in italic are those features that are considered lacking in current applications.

Positive/enjoyable Features	**Negative/annoying Features**
General ranking	Sounds
Ranking between questions	Music
Pictures and colours	*Countdown, a time limit to answer questions*
Working in groups	
Open questions	
Lifelines, the last one standing	
Different ways to login (e.g., QR Code)	
Tracking of the questions and the explanation of the correct answer	
Ranking of the whole module	

Table 5.4: Lacking features in existing educational games

On the other side of the balance, we have the professors, which we interviewed to gain more insight into the results and compare them to the results of the student's survey and what the interests of the main stakeholders are towards educational games.
As a first positive result we obtained that 100% of the interviewed use educational games during lectures, with only 80% of the professors using them sometimes, considering the time to prepare the game long but worth it; while the other 25% use it often and considers the time spent on the game beforehand is reasonable/short.

Similar to the student's survey, we also asked the professors which features already existing in educational games were the most enjoyable/annoying as well as those positive and negative aspects of already created games or those lacking features they would like to see in a newly developed solution. Displayed below in table 5.5 we have the results to those questions, and again, those marked in italic are the ones considered lacking and that professors would like to see in a new educational game.

Positive/enjoyable Features	**Negative/annoying Features**
General Ranking	Thinking of new questions/answers
Sounds and music	Pictures
Easy to use and integrate into lectures	Some existing educational games are too complicated to set up
Tracking of the questions and the explanation of the correct answer	
Pictures	
Possibility to create groups	
Display results/games in a web interface	
Store results of the game (leader board)	

Table 5.5: Lacking features in existing educational games

5.1.1.2 User Experiment

Approach

This first User Testing was carried out at the end of week 4, thus the beginning of the third Sprint to set a perspective of how the web interface should look, its intuitiveness of it and the different features that were missing. We reached out to six different professors at the University of Twente, from the EEMCS and BMS faculties to try out the Lo-fi prototype we developed. Each of the tests was taken individually, during each of them, two team members were involved in the session, one of them in the role of the interviewer and the other one taking notes and observing every movement and decision the participant was making.

First, the project idea was introduced to the participants, giving them a clear view of what the purpose of the web interface and the project itself were serving. Followed by an explanation of how the interview was going to be carried out, we gave full control of the interface to the tester while we were presenting him a list of challenges/tasks to be completed throughout the testing, giving them full freedom to give their sincere opinion of the interface and thinking out loud through each of the pages.

Results

During the experiment and the completion of the challenges by the participants, these were marked as completed ('yes') or incomplete ('no'), based on how difficult it was for the user to complete the task and if they were able to complete the challenge successfully, as can be observed in Table 5.5.

Challenge	1	2	3	4	5	6
Create an account to log into the system	No	Yes	Yes	Yes	Yes	Yes
Analyse correctly what each button is on the teacher's home page	No	Yes	Yes	No	Yes	No
View all courses	Yes	Yes	Yes	Yes	Yes	Yes
Understand course representation	No	No	Yes	No	Yes	Yes
Create a new course	Yes	Yes	Yes	Yes	Yes	Yes

Check the information about a specific course	Yes	No	Yes	No	Yes	Yes
Editing student	Yes	Yes	Yes	Yes	Yes	Yes
Deleting student	Yes	Yes	Yes	Yes	Yes	Yes
Choosing a course to play with	No	Yes	Yes	Yes	No	Yes
Choosing a game mode	No	No	Yes	Yes	Yes	Yes
Starting a game	Yes	Yes	Yes	Yes	Yes	Yes

Table 5.5 Challenges results per participant

During the different runs of the user testing, we could observe there were common points where the participants struggled with the challenges, or they pointed out negative features of the interface that would interfere with their use of the future application. Following the challenge in Table 6.5, the following points will be described in a general aspect, analysing the suggestions made by the participants. Furthermore, each bullet point will be completed with an explanation of the changes made based on the recommendations obtained during the testing, having in mind the possibilities of the project and its requirements of such.

- **Navigation bar:** the names in the navigation bar are not intuitive enough, the majority of the participants clicked on *home* expecting to go to the landing page, not to the teacher home page.

  The number of participants struggling with this feature and recommending the change of such features was not high enough as for considering the change of the names, therefore, the only adjustment made was making the Blue&GO! Signal clickable so that when the user clicks on it, it gets redirected to the landing page.
- **Teacher's home page buttons:** participants struggled to understand what 'play' and 'groups' stand for, they suggested the first one could be used to play individually and the second to play in groups, once they realised the real functionality of it, they suggested changing the name groups to courses since the name confuses with a team formation of the students.

This was a foreseen problem that was already taken into consideration for a change before the user testing, however, we left it on the lo-fi prototype to reiterate the idea that it was not intuitive enough for the users to deduce from 'Groups' that it was the classes page. As the results of the test were clearly showing this was something that should be changed, the name 'Groups' of the button on the teacher's site was renamed to 'Courses' as suggested by some of the participants.

- **Groups/courses site:** when wanting to select a group card, participants clicked on the images instead of clicking on the name of the group. Furthermore, the majority of the participants were confused when creating a group since the pop-up window shown after creating the group looks like an error message.

  Unfortunately, the decision to click on the image could not be implemented, and the only clickable part will be the name tag part, however, we managed to remove the alert message shown after creating a new course to avoid confusion.
- **Group/course information:** the majority of participants did not understand the meaning of the column teams in the table. Moreover, when looking at the leader board, participants expected to see a ranking of the top 5 groups and the overall ranking, not an individual ranking. As a last remark, the participants were unable to distinguish the download information button through the page thinking it was part of the search functionality.

  Once the groups' site was renamed to courses, we decided no change had to be made to the teams' column since the confusion was based on the groups' names, thus now that we changed that there is no need to rename the teams' column. Furthermore, the design of the leader board was changed, to have a more noticeable design, although the rankings did not change, and the decision is still based between individual ranking and group ranking.
- **Play site:** participants clicked on more than one group expecting to be the different groups that should be playing, instead of them being courses. Furthermore, for some participants, it was complicated to understand what the game modes were for or to remember the primary explanation given on the landing page, even after clicking on the dropdown and having read the definition of game modes.

  Following the basis of the groups' site, renamed to courses, and the difficulties presented to understand how many 'groups' were supposed to be chosen to play, we decided to rename the groups to courses on this site too. Moreover, the descriptions of each of the game modes

were changed to be less wordy and that could give a more understandable definition of each of the game modes.

5.1.2 System Testing

Testing is an important part of any project within the IT field, and this project contains several components that would require testing to work as it is required to. There are two ways to test: manually and through test functions.

However, due to a lack of time, we were not able to create functions, and it is because there were several parts that the team should have focused on. Thus, we did some manual tests for the different components. First, the web page required that all its features work as required, as well as the connection with the database. The login and sign-up pages were tested and compared with the information within the database. Thus, users that are not registered would not interact with the game functionalities. Second, manual tests were made for the responsive web interface that had to display and send data in real-time, so the user would interact with it smoothly and the game would not be affected by any delays or malfunctioning. Third, for the buzzers, we had 3 different steps: 1. The design and measures taken were based on each component like the ESP32, card reader, etc. however the 3D printer gives an offset of 2mm that makes a difference when an item is printed. 2. The actual measures with the new design were made and 3. The connections and testing of each component had to be individual to have an overview of each connection and its requirements.

Next, there is the connection by Bluetooth with the buzzers. Since there are protocols that handle the communication, we did tests with the devices and checked each time if the connection was successful. The use of a low prototype of the buzzer helped in the testing of the buzzer, as every single hardware component needed to be tested so it would have the functionality expected in the first design. However, it is hard to simulate the connection of other devices. Nevertheless, we did test each interaction that the devices should do and the time they are required to do it. Finally, there is the connection with the database and the capacity for change and updating information on it. Since the process is the same as the webpage, we did use the same function and developed new ones that were required for the game interaction. Each function was verified in terms of syntax by the PostgreSQL part to complete an interaction when it was needed.

It is important to highlight that the creation of functions was not possible because of a lack of time and the workload of the project. Nevertheless, this could be part of the future work of the project to improve the efficiency of the features of the project.

Chapter 6

Product Considerations

6.1 Privacy and Security

Privacy is what people consider the most important issue within any IT project. In this case, Blue & Go handles specific data via a variety of environments that are worthy of consideration. Since the project is focusing on the education system, teachers' and students' information is handled. This means that students' numbers, names, and courses of an educational institution are used to manage the gamification of the project. It means that this sensitive information is stored within a database. To accomplish this, all information within the database must be encrypted to ensure that no data is compromised. This also means that login methods and verification would be hashed and compared directly with the database. Moreover, different pages within the interface retrieve information from the database for displaying and managing any information about a course. In this case, the system must be aware of any SQL injection by any man-in-the-middle that would want to obtain any data or even change it for their purpose.

On the other hand, there is also the hardware that communicates with the database. There are the buzzers that are connected through Bluetooth to the emitter. In this case, the communication is in bytes. However, the system also transmits sensible data, which is the student number that helps the system identify participants in a specific buzzer. Thus, the communication should also be encrypted. Moreover, the emitter handles the communication with the online game of the teams or students of a game; thus, it also applies the rules for SQL injection.

It is important to highlight that these are the main issues within our system that are not covered yet, not because they are not important, but because the project was focusing on the development of the game and web interface. Nevertheless, it should be considered an important next step for future work.

As was mentioned before, the buzzers are connected to the emitter, which is a Raspberry Pi, via Bluetooth. It implies that the buzzers should be paired manually with a specific raspberry pi. It is because of the on-build configurations of the device that disabling such a feature would mean a vulnerability, and any device could connect to the emitter and corrupt our system.

6.2 Risk Analysis

6.2.1 SWOT

6.2.1.1 Strengths

The strengths of Blue&GO! are the main principles on which its efficacy is based. Therefore, the core of its strengths can be found in the key design pillars: enjoyment, ease of use, and flexibility (see 3.1.1).

The enjoyment aspect of (educational) games is a feature that is required in all (educational) games to excel and can also be seen in Blue&GO! in the provided competitiveness, the colourful palette, animations, and more. However, these enjoyment features are not what separates Blue&GO! and its competitors, as they are present in most games to optimise enjoyment. Blue&GO! separates itself by providing multiple game modes to deliver different gaming experiences, and also providing physical components which are designed to be more enjoyable to use in comparison to their digital equivalents.

The second strength, taken from the pillars, is the ease of use. In most educational games ease of use is of high importance, to make the game intuitive, easy to use and more enjoyable for students. This is also taken into account in Blue&GO!. However, Blue&GO! Additionally excels in the ease of use for teachers. This makes Blue&GO! more easily adopted in educational environments as teachers are in charge of starting and controlling educational games. This can increase Blue&GO!’s prevalence, as teachers most often decide whether a certain educational tool is used or not.

Lastly is flexibility. Blue&GO! Uses portable components, so it can be taken to other places and can be easily played outside of the classroom. Flexibility can also be seen in the gameplay, as multiple game modes can be appropriate in different settings and using speech to ask and answer questions provides flexibility in how answers and questions can be phrased.

6.2.1.2 Weaknesses

The main weakness of Blue&GO! regards its big reliance on physical components. While buttons are portable, they are still less flexible than having a mobile phone application. Additionally, there is a limited number of buttons available for every course, which can force the teacher to split classes into groups and play in groups instead of individually. With a bigger reliance on physical components also comes higher susceptibility to technical issues, such as broken components, connection issues or dead batteries.

6.2.1.3 Opportunities

Blue&GO! already has three game modes: classic, random, and lifelines. For every game mode, settings can also be modified to change the gaming environment as desired. As Blue&GO! is not limited to only one game mode, more game modes and more advanced settings for each mode can be implemented to make the game even more adjustable. This can further increase Blue&GO!'s flexibility and can also increase enjoyment, as games can be played just as desired.

Additionally, Blue&GO! has an internal database keeping track of scores of students over longer periods. Currently, this is used to show rankings among students within courses for every game and can be reviewed and downloaded. However, for the future of Blue&GO! This feature can be expanded and improved, which can increase the competitive drive for students. This can be done by not only limiting comparing scores in one game but being able to see comparisons of total scores over longer periods of multiple played games. Additionally, the ranking system can be changed to enable comparing rankings between different courses. If a later version of Blue&GO! stores data online instead of internally, it could be used to compare even more different scores between students, as it is not limited to the courses belonging to one teacher.

6.2.1.4 Threats

As was mentioned in opportunities, competitiveness can contribute to students' drive and can easily be increased in future versions. However, too much of a good thing is bad, which is especially the case for competitiveness. With just the right amount it can increase drive, but too much of it can backfire in toxic behaviour among students. Therefore, the competitive nature of Blue&GO! can be experimented with but must be monitored carefully to avoid toxic competition.

Chapter 7

Evaluation

7.1 Planning

From the very beginning, we divided the project into sprints, setting up our deadlines and the ones with the supervisor, to have a more structured view of what would have to be achieved and the time we had to complete it. The first weeks were right on track, designing Blue&GO!, based on the specifications of the product owner and following ideas from lacking features of existing games, we accomplished the deadlines of the first two sprints, however, the time for the following two sprints (sprint 3 and 4) couldn't be managed as expected. Developing the front-end and its basic structure fell within the proposed schedule, nevertheless, for the back-end and development of the hardware we had to postpone the deadlines later in the sprints. Due to unforeseen circumstances related to the connections of the raspberry pi and the ESP32 with the rest of the project, we had to work more than expected at the beginning, to achieve the MVP right in time. With this, delays on the rest of the project were soon arriving, we had to cancel the possibility of making a second user experiment with the final MVP prototype, and the backend was put on hold until the connections between the hardware components could be achieved.

To sum up, the planning developed at the beginning of the project was followed by small adjustments and having to deal with unforeseen situations, contributing to the fact that in the end the final deadline could not be reached. Nevertheless, the work done by all the team members and the good communication with the product owner, made it possible that in the end, we are satisfied with the project result and the evolution of the team throughout the completion of the pre-established planning and the different sprints.

7.2 Requirements Completion

In the table below (7.1), there is an evaluation of the accomplished requirements, following the pattern that *state in green means accomplished and state in red means not developed*.

MoSCoW level	Requirement	State
MUST	The teacher must be able to initiate a new game through the web interface.	
	The teacher must be able to visualise the players' progress within a web interface.	
	The user must be able to identify the states of the game	
	The system must count with a web interface.	
	The system must have synchronisation between the student buzzers and the emitter.	
	The system must be safe to use for children.	
	The system must safely store the gathered data.	
	Every buzzer must have light states.	
	The buzzers for students must be connected to an emitter system.	
	The buzzers must generate a light state according to the game.	
	The buzzer must have a power system.	
	The interface must display the points awarded and players' progress.	
SHOULD	The system should recognize which student is using which buzzer.	
	The system should have a game settings interface, to modify features (e.g., the number of points awarded).	
	The system should have a controller for the teacher connected to the interface.	

	The system should have animations among the states of the game.	
	The system should reproduce sounds throughout the game.	
	The buzzers should have a charging system (batteries/cable).	
	The interface should be responsive to different devices.	
	The teacher could be able to modify and change the game settings (game mode, points awarded, time to reply…).	
COULD	The final ranking could also be displayed as a canvas plugin.	
	The teacher could be able to download the result of the game.	
	The buzzers and controllers could be charged.	
	The buzzer could have 4 buttons to allow the possibility of multiple-choice questions.	
	The players could earn or lose points according to the teacher's decision (either correct or wrong answer).	
WOULDN'T	The system will not have the ability to display questions and answers on the web interface.	

Table 7.1 Requirements for completion

Chapter 8

Conclusion

8.1 Future Work

During the evolution of the project, many ideas came into place as to what implementations could be done to have a more unique and appealing educational game, nevertheless, due to the fixed time to complete the project, most of the implementations had to be left out hoping to be added as a future implementation of Blue&GO!

8.1.1 Web interface

There are some things about the web interface that can be improved for future versions. First of all, the code itself can be more organised and better documented. As many things were figured out using trial-and-error, there are parts of the code that are quite messy. Therefore, separating all divisions and styles into proper sections and classes, with additional and improved documentation can make the code easier to understand and work with. This could, for example, be improved by making separate files for components of one page. For example, a separate file could be made for the navigation bar, and each HTML file that contains this navigation bar can refer to it.

Secondly, the design of the front end could be improved in the coherence of its design and the flexibility according to the view height and view width. Many pages already use the same colour palette, but still lack some coherence in design. This applies to the differences between the page initialising the game, the profile pages and the rest. Additionally, problems in the design occur when changing the window size, where some parts are not flexible or are less flexible than others. This causes components to shrink or expand at different rates, changing the layout of the screen. This applies to pages containing generated content such as course cards, for example, the initialising game page and the courses page.

Additionally, the front end can be made more fun and interactive by, for example, integrating more (animated) icons, animations, changing colours etc. This especially applies to the pages providing visual feedback during the game, as they are most important to be fun, to contribute to the enjoyment of playing the game. Not only visual feedback can be improved, but also audible feedback. A song during the game, the correct/incorrect sounds, or the sounds when loading the ranking can all be improved to be made catchier and more distinctive. This could make gameplay more memorable and more enjoyable.

Finally, most improvements in the web interface can be obtained through user testing, as they directly interact with the web interface and their preferences and problems are, therefore, direct information on how to improve. This will be further discussed in 9.1.4.

8.1.2 Game settings

Improvements in the functionality of the game and its different settings can be developed in future steps of the project. First, game modes and settings can be improved. This, first of all, includes the random and lifelines game modes to be fully implemented and connected with the database to make them fully functional in the backend. To add to this, additional game modes and extra game settings can be introduced. An example of an additional game mode is a game that is only played with one buzzer and the one that gets to answer the question is the student that clicks the buzzer first. This can be a game mode with an even more increased competitive feel.

Furthermore, Blue&GO! can be changed so that it takes into account that not everyone will be playing the game or that people outside the course will play. As in traditional lectures, there will be students not participating in a class at all, people attending class just as a listening audience, or participants who are not registered for the course. In these cases, Blue&GO! could be more flexible in both playing the game as well as managing scores.

8.1.3 Buzzers

Regarding the physical components of Blue&GO! some features could be added and improved. The first is to increase its robustness, to remain functional after falling, being underwater or other potential damages. Additionally, enjoyment of the buttons can be increased. This could be done

by making the button press more satisfying, as right now it can only be pressed a small bit and does not have an overly noticeable spring to it. To add to the enjoyment, the buttons can be made more aesthetic. This could be done by shaping it better, choosing different materials, and changing the colours. To optimise this, user experience can be used, as will be discussed in 8.1.4.

Additional buzzers can be built with the same or an improved design which will keep the same functionality and the enjoyment of the previous buzzers. However, the server protocols for handling the Bluetooth connections with these buzzers can be enhanced to optimise the connection and the protocols. In this way, the server would handle more than 15 or more buzzers without problems of interrupted connections or delays between the action of a user with the server.

8.1.4 User Testing

Lastly, and most influential and most time-intensive, regards more user experience testing. Blue&GO! as a product is based on the user experience of students and teachers and it is vital that optimising this is important in future concerns. Values of important stakeholders were already valued high throughout the current design of Blue&GO! and must remain of critical importance in its future. Therefore, future work can be done in testing all components of Blue&GO! to improve them according to the stakeholders' wishes and needs.

8.2 Conclusion

Blue&Go! has been the result of lecture gamification and how it can still be possible to take advantage of physical hardware for engaging students. In this context, this project aims to educate students through the use of buzzers as part of it. Thus, the professor can verify what his students have learned, but it also encourages students to be active in a lecture and reinforces their knowledge. Blue&Go! is not only designed to help students but also professors with the main idea of saving time. It is because the game does not need any previous planification, such as questions or even predefined answers to verify. Thus, the game is more focused on asking any open or closed questions at any time during a lecture while still enjoying different features of a virtual game such

as leader boards, sounds, and visual animations. As was discussed in the previous sections, The game itself contains a webpage where the teacher can manage the game setting as well as the information about the course and students. This helps the teacher to keep track of his activity with each course, additionally, it gives the option to export such results. On the other hand, there is the emitter (Raspberry Pi) that handles the connection among the buzzers. These buzzers are capable of identifying students when they pass their student cards in front of it. These features together with the game modes make Blue&Go! unique among other gamified games in the market. This project has the potential to grow and change the way of education within institutions. The basis for a complete game has been placed where future groups can continue improving to make it more efficient and more interactive for all users.

References

[1] *5,600+ Animated Icons - Lordicon*. (n.d.). Lordicon. https://lordicon.com/

[2] *Bootstrap Icons*. (n.d.). Bootstrap. https://icons.getbootstrap.com/

[3] P. (n.d.). *GitHub - pybluez/pybluez: Bluetooth Python extension module*. GitHub. https://github.com/pybluez/pybluez

[4] *Pair a Raspberry Pi and Android phone — bluedot 2.0.0 Documentation*. (n.d.). Bluedot. https://bluedot.readthedocs.io/en/latest/pairpiandroid.html#using-the-command-line

[5] Instructables. (2021, October 9). *Easy Setup IR Remote Control Using LIRC for the Raspberry PI (RPi) - Updated Oct 2021 [Part 1]*. https://www.instructables.com/Setup-IR-Remote-Control-Using-LIRC-for-the-Raspber/

[6] a Href=/Team/Elizabeth-King Hreflang=En>Elizabeth KingEducation is Fundamental to Development and Growth. World Bank Blogs. Retrieved April 29, 2022, from https://blogs.worldbank.org/education/education-is-fundamental-to-development-and-growth

[7] *Gamification in Education*. (n.d.). Frontiers. Retrieved April 29, 2022, from https://www.frontiersin.org/research-topics/30148/gamification-in-education#:%7E:text=The%20use%20of%20gamification%20in,transferred%20in%20an%20educational%20context.

[8] Putz, L. M., Hofbauer, F., & Treiblmaier, H. (2020). Can gamification help to improve education? Findings from a longitudinal study. *Computers in Human Behavior*, *110*, 106392. https://doi.org/10.1016/j.chb.2020.106392

[9] *The Power of Gamification in Education | Scott Hebert | TEDxUAlberta*. (2018, May 7). [Video]. YouTube. https://www.youtube.com/watch?v=mOssYTimQwM&feature=youtu.be&ab_channel=TEDxTalks

Appendix A

Surveys

A.1 Student Survey Questions

1. Filling this questionnaire form means approving the processing of your data for this project. The answers are completely anonymous.
   a. I agree
   b. I disagree
2. Do you find educational games to be enjoyable? E.g., Kahoot or an online quiz. *(On a scale from 1 to 10, being 1 'Not enjoyable at all' and 10 'Very enjoyable')*

<u>Referring to educational quizzes</u>

3. How important is for you the ability to reread the question or multiple-choice answers on the screen? *(On a scale from 1 to 10, being 1 'Not important at all' and 10 'Very important)*
4. How would not displaying the questions on the screen affect your overall enjoyment of the game? *(On a scale from 1 to 10, being 1 'I would not like to use the system in lessons anymore' and 10 'I would be more likely to use it)*
5. What are enjoyable features of such games?
   Mark all that apply
   a. Sounds
   b. Music
   c. Ranking in between questions
   d. General ranking
   e. Colours
   f. Pictures
   g. Other *(specify)*
6. What are the annoying features of such games?
   Mark all that apply

a. Sounds
b. Music
c. Ranking in between questions
d. General ranking
e. Colours
f. Pictures
g. Other *(specify)*

7. What positive or negative features of educational games stand out the most?
8. Do you find educational games to improve your attention? *(On a scale from 1 to 10, being 1 'No, not at all' and 10 'Yes, a lot)*
9. Do you find educational games to improve your learning? *(On a scale from 1 to 10, being 1 'No, not at all' and 10 'Yes, a lot')*
10. What features do you feel like educational games are lacking? What features would you like to see in a new educational game?
11. How would you like to gain the right to respond?

    Mark all that apply

    a. Via a web interface
    b. Via a phone interface
    c. Via a physical button
    d. Other *(specify)*
12. If you would like to help us during further research, please leave your email address

A.2 Teacher Survey Questions

1. Filling this questionnaire form means approving the processing of your data for the purpose of this project. The answers are completely anonymous.
    c. I agree
    d. I disagree
2. Do you use educational games in class? E.g., Kahoot or an online quiz.
    a. Yes, often
    b. Yes, sometimes

c. Rarely

d. Never

3. Why do you think these games are useful?

   Mark all that apply

   a. They improve students' attention

   b. They improve students' ability to learn

   c. They are enjoyable

   d. I do not think they are useful

   e. Other *(specify)*

4. What do you think about the preparation time for these educational games?

   a. Too long, not worth it

   b. Long, but worth it

   c. Reasonable / short

   d. I never use such applications

   e. Other *(specify)*

5. What are the enjoyable/engaging features of such games?

   a. Sounds

   b. Music

   c. Ranking in between questions

   d. General ranking

   e. Colours

   f. Pictures

   g. Other *(specify)*

6. What are the annoying/distracting features of such games?

   a. Sounds

   b. Music

   c. Ranking in between questions

   d. General ranking

   e. Colours

   f. Pictures

   g. Thinking of new questions and possible answers

h. Other *(specify)*

7. What positive or negative features of educational games stand out the most?
8. What features do you feel like educational games are lacking? What features would you like to see in a new educational game?
9. Looking into the product that we will develop. To visualise the results and store the student interaction, what would you desire:
    a. Web interface (website)
    b. Canvas plugin (if applicable)
    c. Both
    d. Other *(specify)*
10. How would you like to start the system and ask a question?

    Mark all that apply
    a. By pressing a button on a controller
    b. By pressing a button on the web interface/canvas plugin
    c. By pressing a button on my phone (phone application)
    d. Other *(specify)*
11. Which features would you consider important when talking about the web interface?

    Mark all that apply
    a. Ability to track student interaction
    b. Gamification of lectures
    c. Leader score for the course
    d. Other *(specify)*
12. How would you like your students to gain the right to respond?
    a. Via a physical button
    b. Via a web interface
    c. Via a phone interface
    d. Other *(specify)*
13. If you would like to help us during our further research, please leave your email address

Appendix B

Web Interface Mock-up

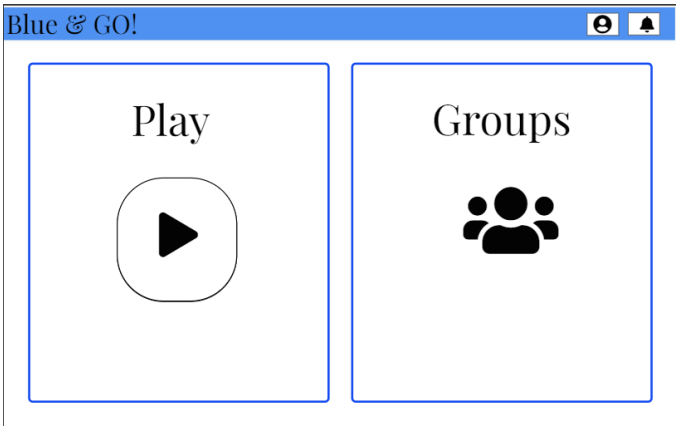


Figure A.1: Teacher site

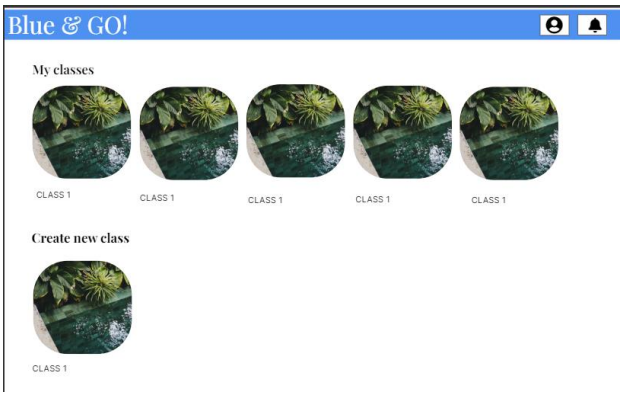


Figure A.2: Courses overview site

Figure A.3:Play site

Appendix C

Links of ESP32 board managers

These links are needed in order to use ESP32 in the Arduino IDE. They should be pasted into preferences as additional Boards manager URLs.

https://dl.espressif.com/dl/package_esp32_index.json,

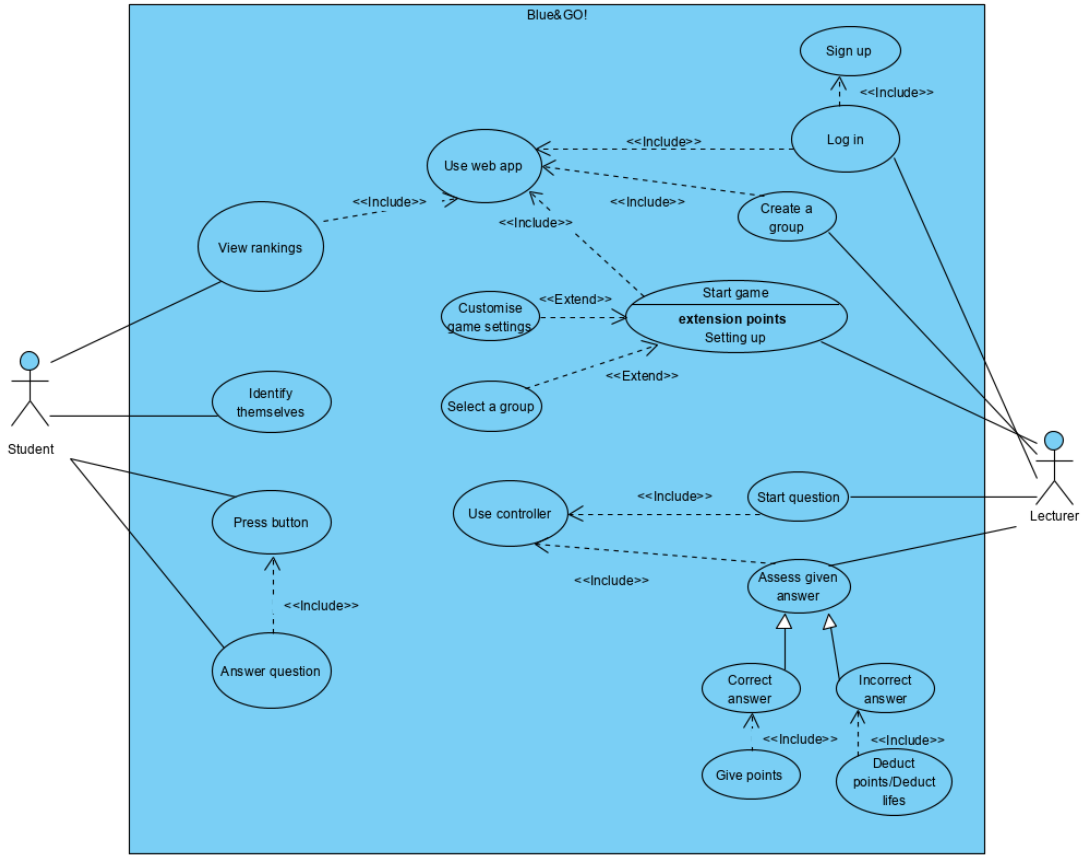
http://arduino.esp8266.com/stable/package_esp8266com_index.json

After placing the links, we are able to download the correspondent libraries for ESP32.

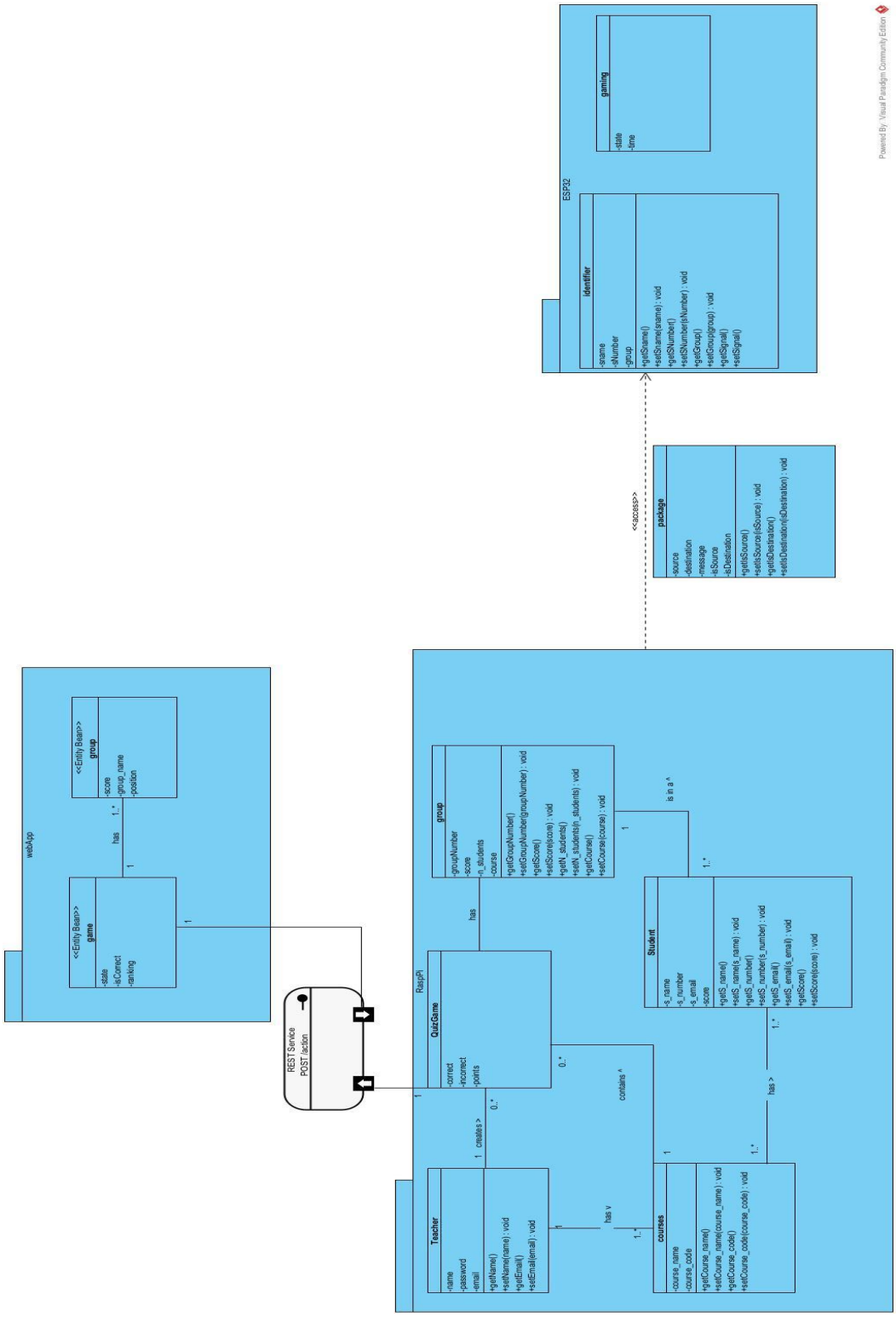
Appendix D

UML Diagrams

D.1 Use a case diagram



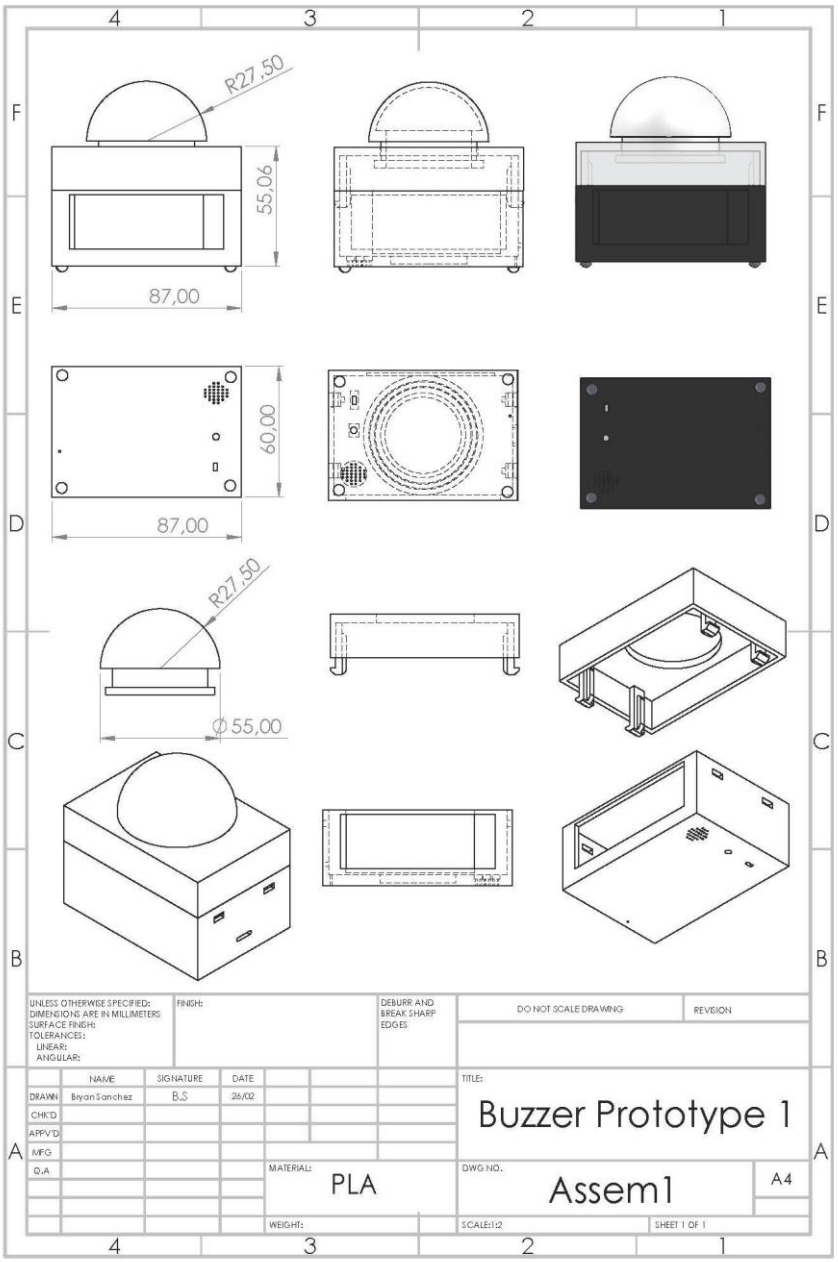
D.2 Class diagram



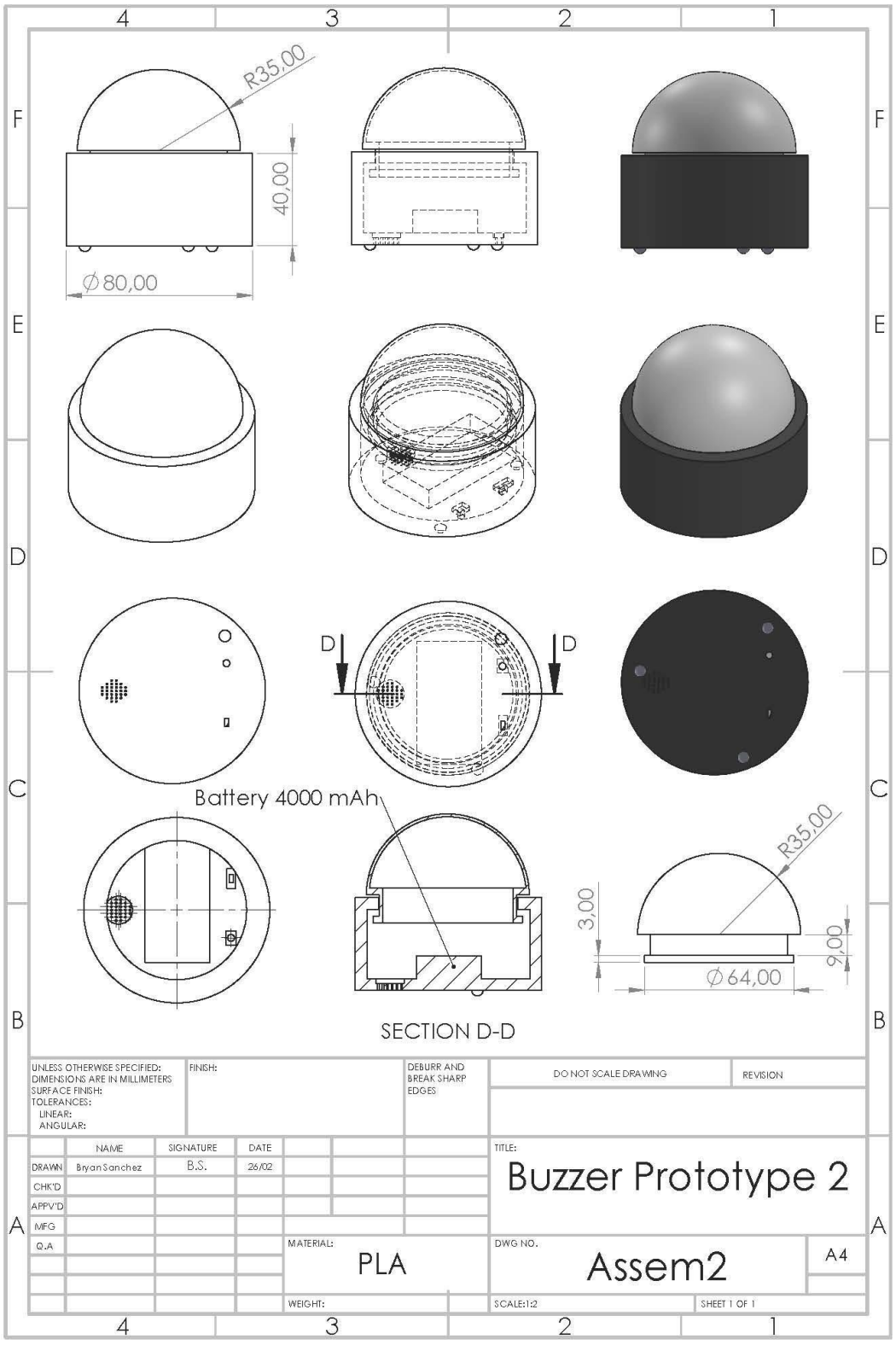
Appendix E

Scale drawings

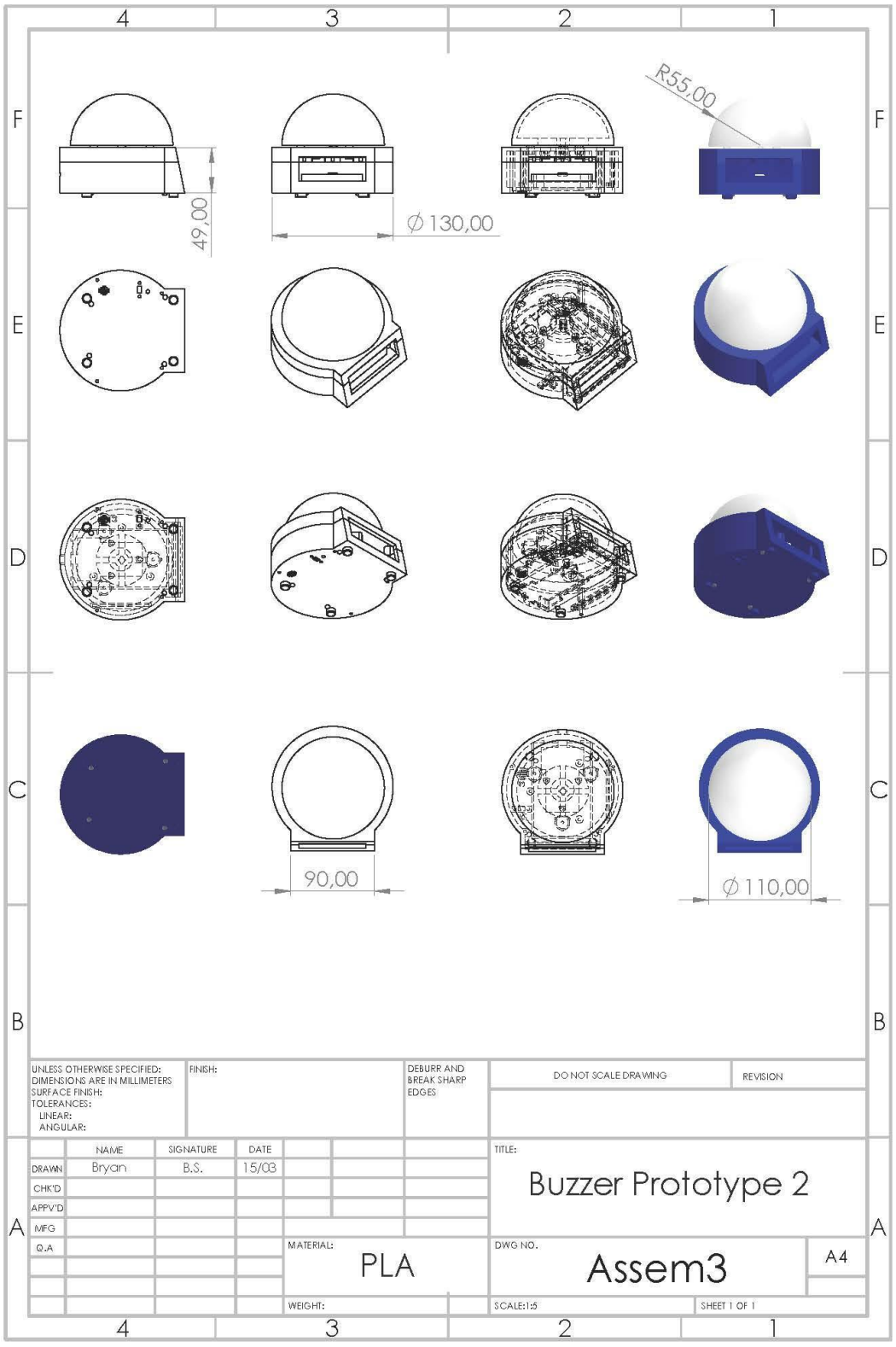
E.1 First prototype version 1.0



E.2 Second prototype version 1.0

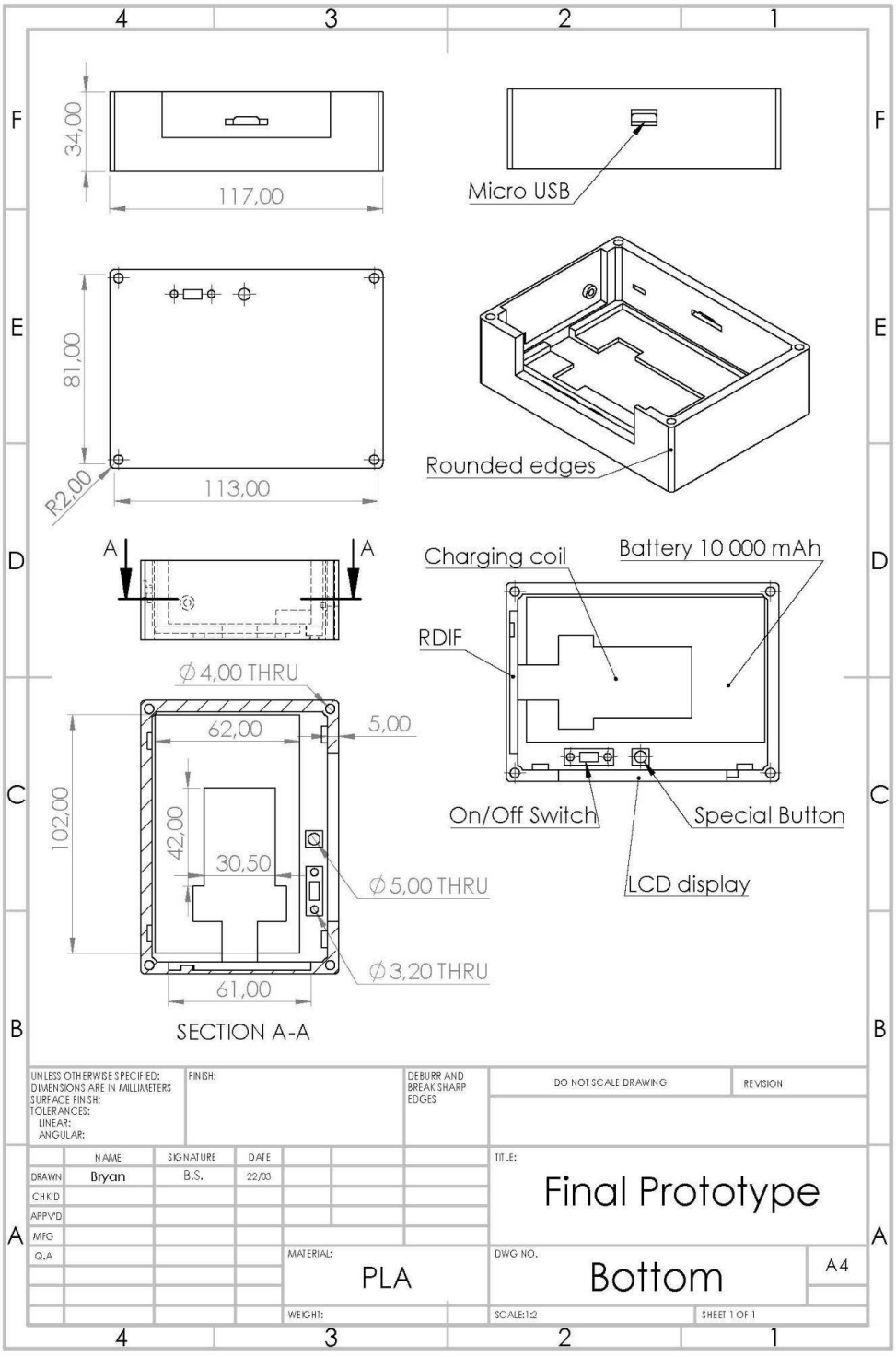


E.3 Second prototype version 2.1

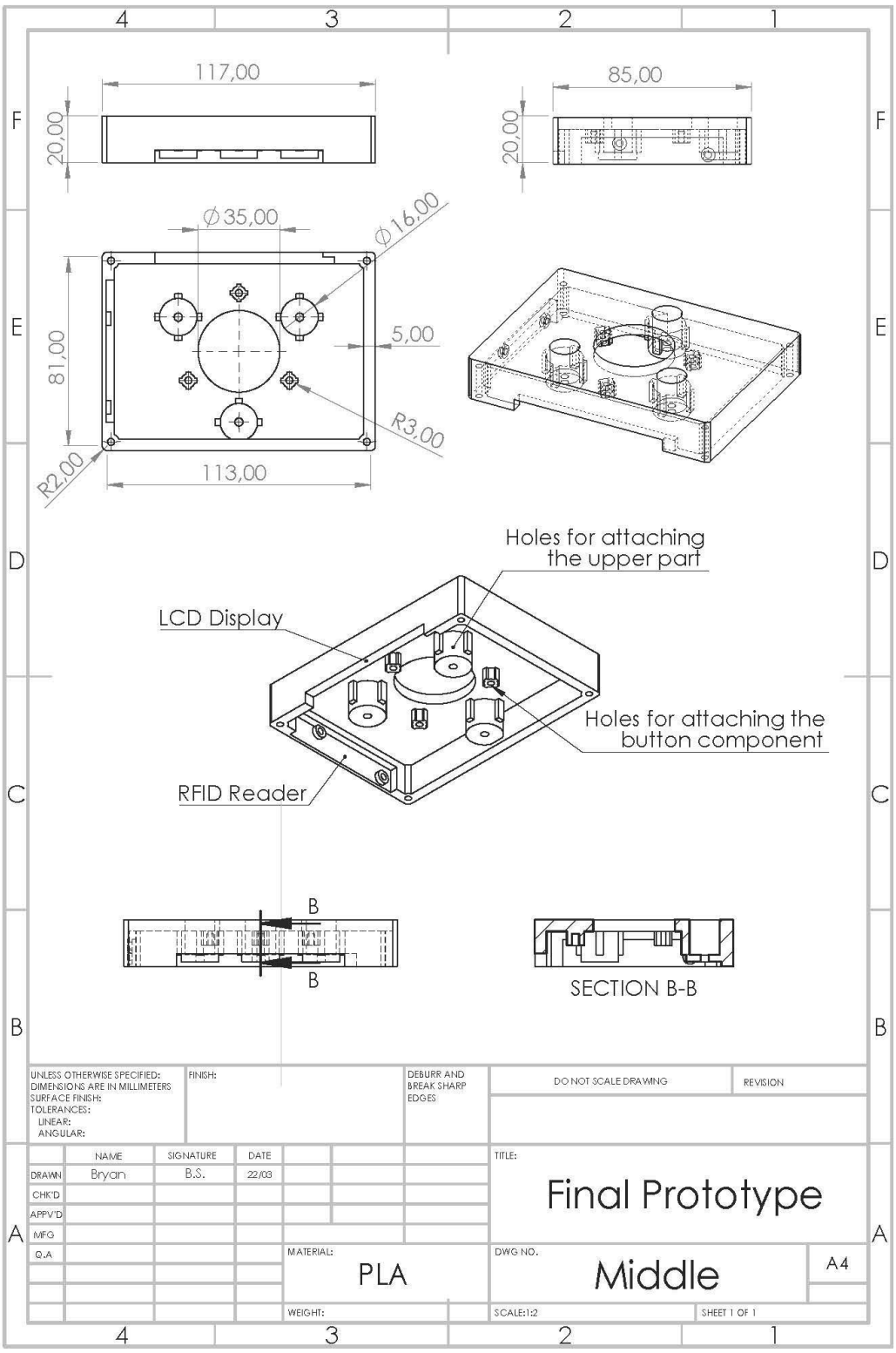


E.4 First prototype version 2.0

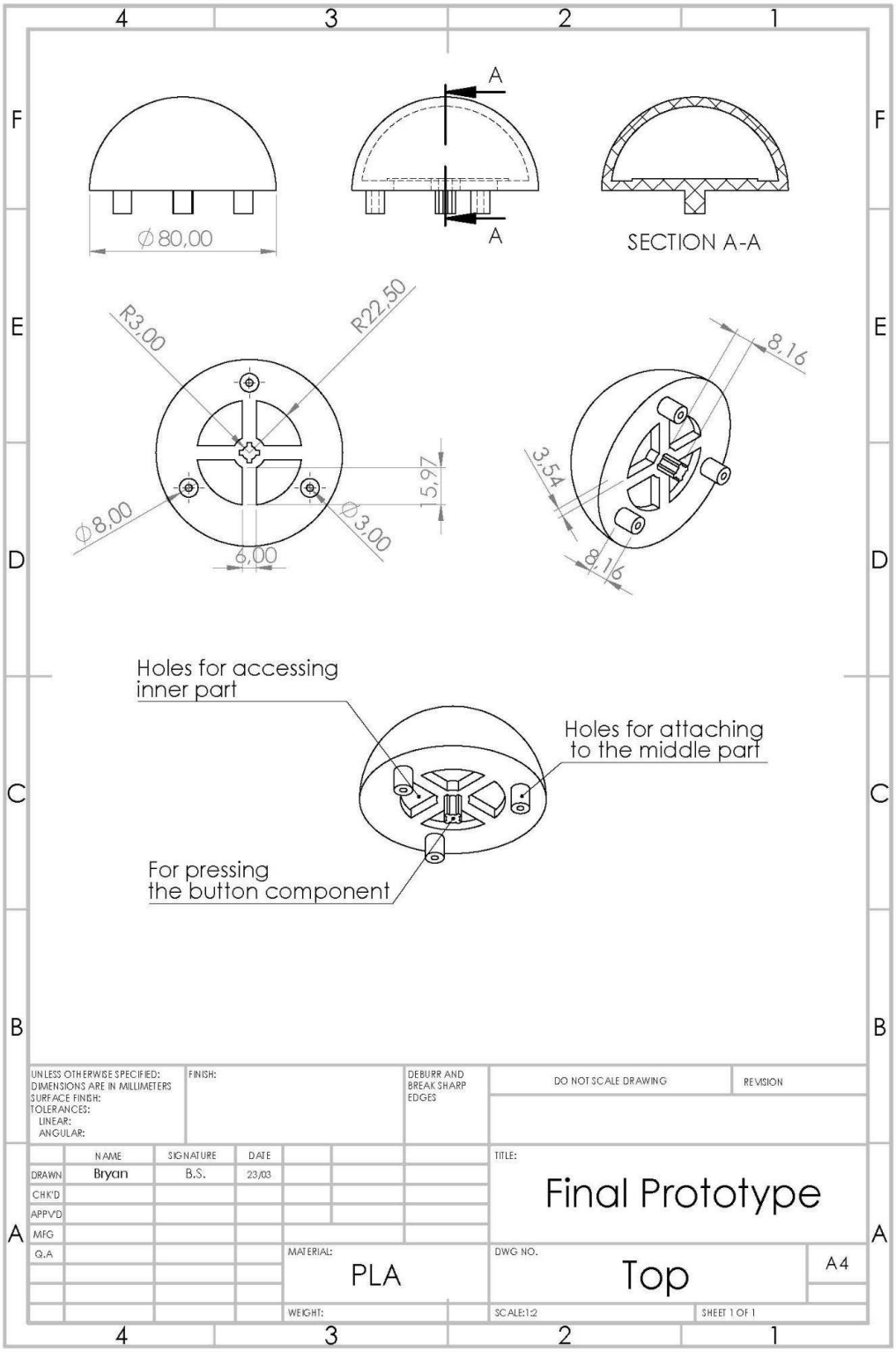
E.4.1 Bottom Part



E.4.2 Middle Part



E.4.3 Top Part


4
3
2
1
F
E
D
C
B
A
A
A
SECTION A-A
∅80,00
R3,00
R22,50
∅8,00
6,00
15,97
∅3,00
8,16
3,54
8,16
Holes for accessing inner part
Holes for attaching to the middle part
For pressing the button component
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:
FINISH:
DEBURR AND BREAK SHARP EDGES
DO NOT SCALE DRAWING
REVISION
NAME
SIGNATURE
DATE
DRAWN
Bryan
B.S.
23/03
CHK'D
APPV'D
MFG
Q.A
TITLE:
Final Prototype
MATERIAL:
PLA
DWG NO.
Top
A4
WEIGHT:
SCALE:1:2
SHEET 1 OF 1

Appendix F

Source Code & Manual

Link: https://github.com/mauleomp/Blue-Go.git

Instructions for using this system:

- Connect the raspberry pi to a power supply, then wait around 2 minutes until the system has everything prepared before starting playing.
- Then, search for a WiFi network with the name ‘Blue&Go’, and type the following password: ‘@BlueAndGo’
- After that, type the following ‘localhost:8080’ in a web browser such as Google Chrome, so it will be displayed on the landing page of Blue&Go.
- Finally, sign up or login in order to manage the courses and results, or start playing with the buzzers.