REQUIREMENTS DOCUMENT

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ICARUS

Version: 2.0

November 27th, 2017

Developed by Team 08

**Project**: ECSE-211 Design Project: Game of Capture the Flag

**Task**: The goal of this project is to construct an autonomous machine that can play a one-on-one version of the game Capture the Flag, performing tasks such as navigation, localization etc.

**Document Version Number**: 2.0

**Date**: November 27th, 2017

**Author**: Ezz Aboulezz

**Edit History**:

|  |  |
| --- | --- |
| Date | Purpose |
| Oct 17 | Added most of the sections |
| Oct 18 | Completed document |
| Oct 25 | Removed incorrect statement from scope, added single-brick solution and modified spec diagram |
| Nov 27 | Added details to purpose and scope |
| Nov 27 | Added to user function, operation environment and compatibility |
| Nov 27 | Final edit and re-structure |

[1.TABLE OF CONTENTS](#_top)

[2. CAPABILITIES](#CAPABILITIES)

[2.1 Purpose…..............................................................................................3](#PURPOSE)

[2.2 Scope….................................................................................................3](#SCOPE)

[2.3 Constraints…........................................................................................4](#CONSTRAINTS)

[2.4 User Functions…..................................................................................4](#USERFUNCTIONS)

[2.5 Operating Environment….....................................................................5](#OPERATING)

[2.6 Performance…......................................................................................6](#PERFORMANCE)

[3. COMPATIBILITY](#COMPATIBILITY)

[3.1 Component Re-use…............................................................................6](#COMPONENTREUSE)

[3.2 Compatibility with Third Party Products…..........................................6](#COMPATIBILITYWITH3PARTY)

[4. GLOSSARY OF TERMS…...............................................................................6](#GLOSSARY)

1. **CAPABILITIES**

**2.1** **PURPOSE**

The purpose is to construct an autonomous robot capable of localizing and navigating to a flag in a given position, capture it and go back to the starting position. The enclosed area, which will be the robot's playing field, measures 12’ x 12’. At the start of a round, both players are placed in their respective corners at a random orientation. Each player waits for a set of game parameters to be downloaded from the game server. Once the parameters, which describe the layout of the playing field, are received each player must cross the river over to the opponent's area. One player always transits using the zip line and returns over water while the other player transits over water and returns using the zip line. Once the robot completes the initial transit it will encounter several blocks with different colors and must figure out how to determine which color represents the opponent's flag. Once it accomplishes that it beeps three times to declare a capture and performs the second transit. At the end of a round, the robot needs to return to its starting point. In summary the robot is required to do the following;

1. Receive parameters from the game controller
2. Localize; given the starting corner number, move to closest grid intersection and note initial position and alignment. This must be completed within 30 seconds.
3. Navigate to the ramp corresponding to the starting zone.
4. Traverse the river using the zip line or shallow crossing.
5. Search for opponent flag.
6. Indicate capture.
7. Navigate back to the start using the appropriate transit method and stop.

**2.2** **SCOPE**

The robot will be a single-brick design and it will be completing its task on a 12’ x 12’ foot hardwood-covered surface surrounded by walls. One competition round lasts for five minutes. During that time, the robot needs to localize, be able to navigate using both methods required, search for a block representing a flag and return to its starting point after completion. The robot must be able to complete the steps specified in the purpose section with both variations; performing transit using the zipline first then 'swim' or 'swim' first then traverse the zipline. The final competition will reward the following capabilities:

1. Ability to receive the game parameters
2. Localization
3. Navigation to mounting point
4. River traversal using both methods
5. Flag search
6. Flag capture (3 beeps)
7. Reverse traversal to starting position and stopping

The final competition follows a points-based system. The previously mentioned capabilities are each worth a point. A successful design is one that can perform each role successfully during the given number of rounds. Each robot will have four opportunities to perform the tasks and a grade will be assigned to the best run. If the robot fails at any point during the run the team does not lose all the points accumulated. The run with the most points will be assigned to each team and a winner will be picked at the end of the competition. The final product of this project is a one-shot operation.

Prior to the final competition, there is Beta demonstration where the team demonstrates the performance of their robot in a time slot of ten minutes. Information concerning the role and the zones will be transmitted by Wi-Fi. The robot is required to do the following:

1. Ability to receive the game parameters
2. Localization
3. Navigation to mounting point
4. River traversal using the zipline
5. Stopping after dismounting

**2.3** **CONSTRAINTS**

[\*See Constraints Document for more details on specific constraints.](Constraints%20Document.docx)

The following is a list of constraints imposed by the client:

1. Robot must compete in a 12’ x 12’ enclosure.
2. Robot must be able to receive parameters through Wi-Fi.
3. It must localize in 30 seconds or less.
4. It must be able to navigate to the mounting point.
5. It must avoid any collisions.
6. It must be able to traverse the river using both given methods.
7. It must be able to search for the block representing the flag.
8. It must be able to determine the color of the blocks and figure which one is the target.
9. It must beep 3 times to announce flag capture.
10. It has a 5-minute limit to complete the competition and return to its starting corner.
11. Up to 3 Mindstorms kits are allowed, adding material will result in an increase in cost.
12. Robot must not attempt to pick up the block.
13. Robot must not make any sounds other than the required capture beeps.

**2.4 USER FUNCTIONS**

The user can make changes to the device before it operates but can't interact with it during operation, except by stopping the program, since it is autonomous. The robot will be placed in a corner of the field in a random orientation before the competition commences. The user will start the robot by pressing the Enter button on the EV3 brick where it will then wait for the parameters to be passed through Wi-Fi. The user’s interactions with the robot will then end until the 5-minute round is over. The robot functions in batch mode, which means it functions without manual control (no interaction).

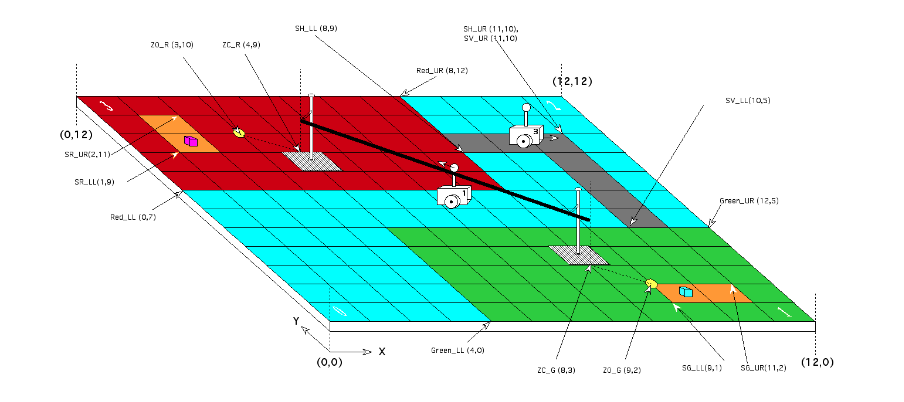
**2.5 OPERATING ENVIRONMENT**

[\*See Constraints Document; section 2.0 Environmental Issues for further details on the problems that can be caused by the operating environment.](Constraints%20Document.docx)

The final competition will be held on the second floor of the Lorne M. Trottier building. The robot will be moving inside of a 12’ x 12’ feet enclosure.

1. The floor is made of nine 4’ x 4’ feet hardwood-covered metal panels that lock together.
2. The surface of each panel is marked with a 4’ x 4’ grid that aligns precisely with adjacent panels. This grid consists of 30.48 cm sized squares bordered by black lines with a thickness of 7mm.

\***Refer to figure below for the map of the competition**



The map consists of three zones. Two zones have the corners where the players will start (red, green) and the blue zone is the virtual river, which players will cross either using the zipline or by 'swimming' in the grey area.

**2.6 PERFORMANCE**

[\*See Section 2.2 Scope and 2.3 Constraints for more details on performance.](#SCOPE)

Since the robot must function without human interaction, it needs to be able to complete several tasks simultaneously for the 5-minute period. It should be able to navigate the field to the target destination (mounting point) and then scan the opponent's side using a searching algorithm for finding the block. While navigating, it must also continuously perform odometry and avoid the walls of the arena; hence the robot needs a distance and color sensing system. Since it must travel for 5 minutes, it must accumulate the least amount of error possible, which is why localization and accurate navigation are crucial.

1. **COMPATIBILITY**

**3.1 COMPONENT RE-USE**

Existing components including the ultrasonic light sensor, the color sensor, and the motors are to be used with the EV3 controller. The software utilized to run these sensors in order to accomplish specific tasks such as localization, odometry and wall following has been developed during the research and development phase of the labs. Implementation of this code will greatly reduce the time needed for further tasks, such as integration, since this code forms the basis of our software architecture. Previous mechanical designs will also be useful as we got a good idea of the robot's mechanics and how we can add components to the existing designs we have.

**3.2 COMPATIBILITY WITH THIRD PARTY PRODUCTS**

The client mandates that the robot must travel on a wooden arena-floor, navigate across a river and capture the flag. Lego does not provide all the materials needed. The ECSE department will provide most of the material needed to build the competition environment. The brick is compatible with Eclipse/Java and has Wi-Fi and Bluetooth compatibility. The programming will be done on Eclipse with the LeJOS plugin. The robot also has to communicate by Wi-Fi connection to retrieve the input parameters.

1. **GLOSSARY OF TERMS**

Eclipse: an integrated development environment (IDE) used for [computer programming](https://en.wikipedia.org/wiki/Computer_programming).

EV3 brick: Lego Mindstorms EV3 is the third generation robotics kit in [Lego](https://en.wikipedia.org/wiki/Lego)'s Mindstorms line.

LeJOS: leJOS is a [firmware](https://en.wikipedia.org/wiki/Firmware) replacement for [Lego Mindstorms](https://en.wikipedia.org/wiki/Lego_Mindstorms) programmable bricks. It includes a [Java virtual machine](https://en.wikipedia.org/wiki/Java_virtual_machine), which allows Lego Mindstorms robots to be programmed in Java.

Mindstorms: The Lego Mindstorms series of kits contains software and hardware to create customizable, programmable robots.