**Arduino Security Robot – The Kenobis**

**Contents**

[**Team Details and GitHub Account** 2](#_Toc382294766)

[**Project Proposal – Arduino Security Bot** 2](#_Toc382294767)

[**Requirements** 2](#_Toc382294768)

[**Running System Screenshots** 3](#_Toc382294769)

[**Production Issues (Technical or Group Management)** 3](#_Toc382294770)

[**Reflection** 3](#_Toc382294771)

[**Test Cases** 3](#_Toc382294772)

[**Review Meetings** 3](#_Toc382294773)

[**Team Member Contribution** 3](#_Toc382294774)

[**References** 3](#_Toc382294775)

**Team Details and GitHub Account**

**Team Name**

The Kenobis

|  |  |
| --- | --- |
| **Team Members** | **Student Number** |
| Andrew Spackman | s1309454 |
| Carl Berquist | s1310712 |
| Josh Ravenscroft | s1303892 |
| Reece Kelly | s1304388 |

**GitHub Account Details**

Username: TheKenobis

Email: s1309454@gmail.com

Password: TheKenobis1

URL: https://github.com/TheKenobis/ArduinoSecurityBot

**Project Proposal – Arduino Security Bot**

Our proposed idea for assignment 2 of CT4010 is the Arduino Security Bot. The security bot will be used to detect motion in a specified area and when motion is detected for a specific period of time, the security bot will then send an email to a specified email address as a response and keep a textual log of activity.

**Requirements**

Arduino with various components attached:

* Requirement 1: Ultrasonic Sensor for motion detection
* Requirement 2: LEDs for confirmation of Security Bot status
* Requirement 3: Button for switching the system on and off

Java Application which will send and receive status to and from the Arduino, with automatic port finder, including:

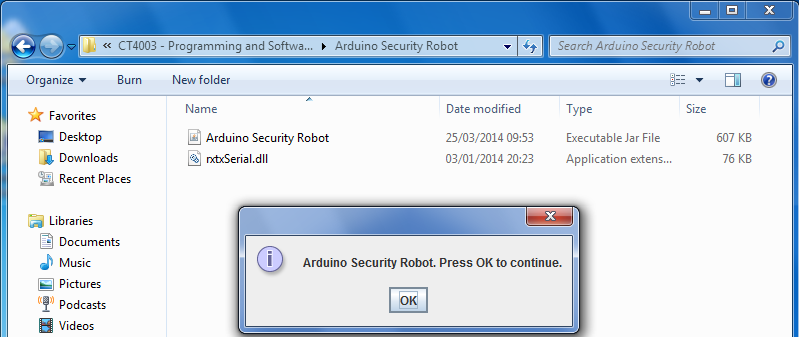
* Requirement 4: Receive Arduino status through serial port
* Requirement 5: Use serial port input as part of the program
* Requirement 6: Determine next steps in relation to input
* Requirement 7: Send an email to a recipient
* Requirement 8: Confirm sending status

The Java application will also send an email to a specified recipient in response to the input received from the Arduino. This will include the email recipient’s address, subject header and email contents and timestamp. The email component will work through java to ensure that is sent successfully in relation to the Arduino’s input. It will also maintain a text log of activity as a fail-safe to ensure that if an email isn’t sent successfully, a textual log can be accessed from the computer system. The textual log will also confirm email sending status (sent or not sent).

**Running System Screenshots**

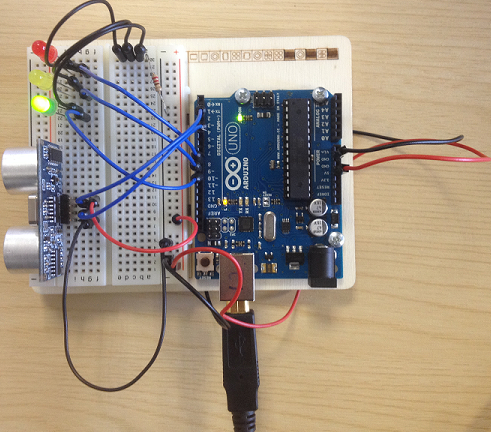
**Java Application** (Eclipse Developer)

**Java Application** (.jar Version)

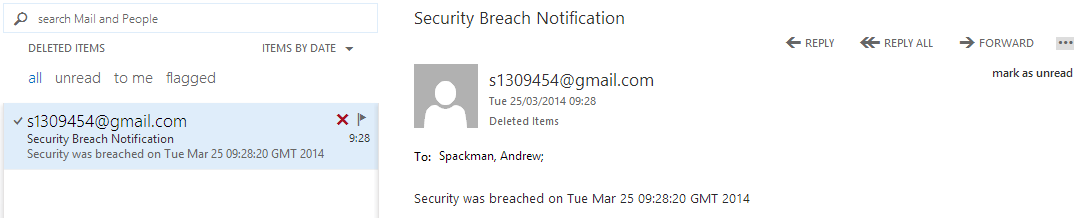


**Arduino** (Serial Monitor Pre Set-up)

**Arduino Circuit**



**Email Received**



**Production Issues (Technical or Group Management)**

**Reflection**

**Test Cases**

**Review Meetings**

**Team Member Contribution**

**Andrew**

**Carl**

**Josh**

**Reece**

**References**

Notes:

Word Count – As long as criteria is covered, this isn’t an issue.

Presentation – Use the video to showcase the project working. Explain in detail afterwards