Green Recycle Bank



Emily Lomempow

1106068964

Gandhi Arssegi

1106000666

Yessy Anggraini

1106054782

**PREFACE**

First of all, let us send the highest bless to God who always give His mercy to us. Secondly, we will deliver the special thanks to all of the people who give the support to us, especially to Prof. Riri and Mr. Ruki who has given us tuition, and not forget to thanks to all of the people who give the support to us, morally, physically, directly or implied support. Because without all of them our project will not finish like it supposed to be.

G.R.B or Green Recycle Bank is a microcontroller based implementation project of water distributing. With the main objective is using it to help farmer or gardener irigate their field, making their work simpler and easier. Backed up by simple GUI (Graphical User Interface) and hardware system, it is easy to understood by the users because we already analyze and design it sole purposely to the user subject none as the farmer only.

For creating this G.R.B project, we worked in team and believes finished this application with hope to help other people with knowledge we implement in it. Otherwise we also know that in this application we have a lot of flaws and mistakes so we are open for input and critics, so in the future we can develop the project perfectly.

**TABLE OF CONTENT**

Cover 1

Preface 2

Table of Content 3

CHAPTER 1 : INTRODUCTION

1. Background 3
2. The Goals 3
3. Objectives 3
4. General Description of Software 3

CHAPTER II : PROJECT MANAGEMENT

1. Timeline 4
2. Task Division 5
3. Risk Management 6

CHAPTER III : DESIGN

1. UML Diagram 7
2. Hardware Specification 9

CHAPTER IV : IMPLEMENTATION

A. Operating System and Programming Language 10

B. GUI (Graphical User Interface) 31

CHAPTER V : TESTING AND ANALYSIS

A. Testing form ................................................................................................................33

B. Testing Result and Analysis....................................................................................35

CHAPTER VI : USER MANUAL 39

REFERENCES 40

**CHAPTER I**

**INTRODUCTION**

**A. BACKGROUND**

Nowadays, peoples get more ignorant. Forgetting everything beside their concerns, including their environment and the place they stand “The Earth” itself. Because of that there come pollution as the one of the most biggest issue in the world. We see, everything big start from small things and in this context it is trash. Where as we keep questioning, why we cannot just put the trash in its right place? Our team analyze it is because lack of reasoning. Ignorance makes people focus on their own and once again forgetting everything behind.

So we are here as the team making this project, to make the reason why everyone should put the trash in the right place while helping make the better systems. By simple thing, encouraging. We will encourage people who put their trash in the right place by giving them rewards in form of points and will be accumulated for each person in the end to at least show them at least there are some meaning when you do something good and that is what the true rewards is.

**B. THE GOALS**

* Help to raise people’s enthusiasm for environment.
* Provide a better systems of garbage collecting.
* To accomplish final project of project management subject.

**C. OBJECTIVE**

Objective of G.R.B project is to make a device that encourage people to support their recycling matters. By making a device that have a good usability, user-friendly and rewarding the users.

**D. GENERAL DESCRIPTION OF DEVICE**

G.R.B device are consist of 3 main components, Arduino microcontroller with external internet modem for wireless communication to integrate with other component. Personal Computer for database storage of the data and displaying user interface that integrated to the arduino. Also, force sensor as the main component attached to the arduino to extract the physial data of the main function which is garbage weight measuring. By combining these 3 components we will make a complete function to achieve our goals and objectives.

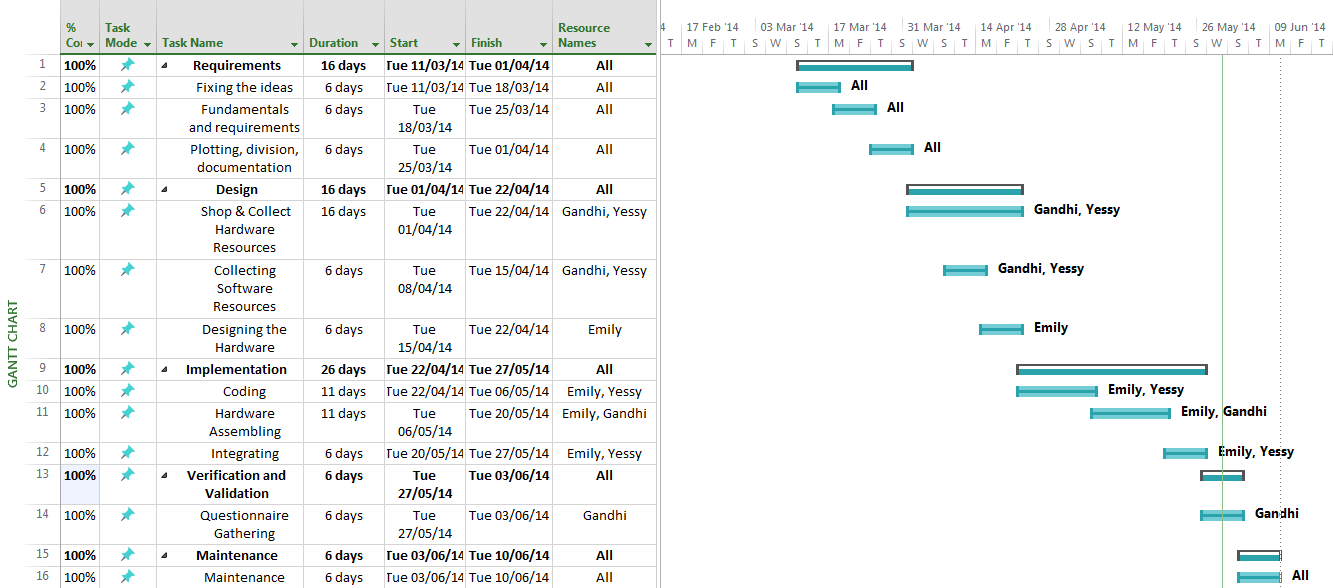
**BAB II**

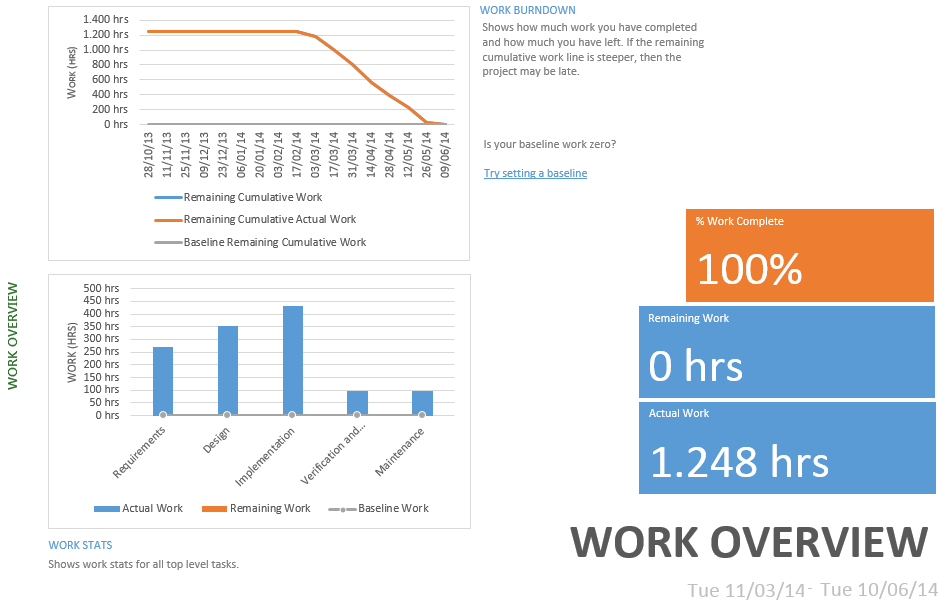
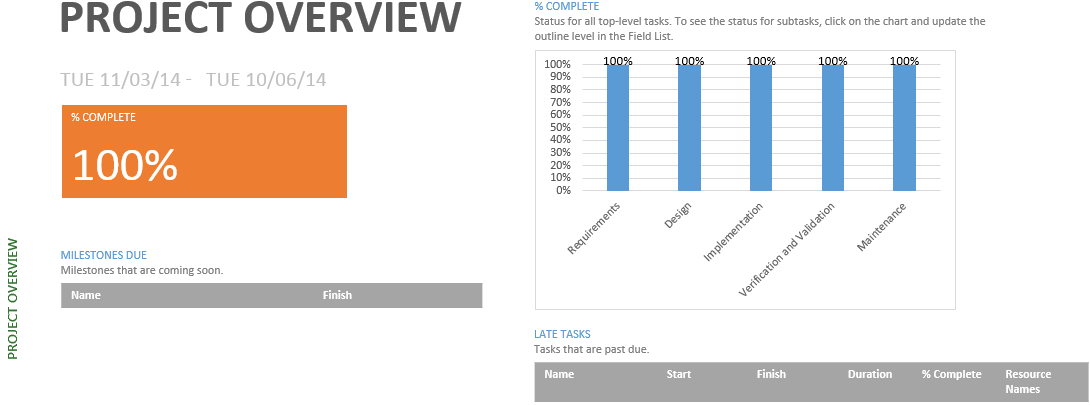
**PROJECT MANAGEMENT**

G.R.B Project management adapting from waterfall model-like management for managing the production of a product within time and funding limits. We have project timeline, member jobdesk and risk management to sightseeing the future of the project. For things that not explained in this documentation the project management also involves technical, organizational and personal management of each member precisely to make sure the project fulfill its objectives.

**A. TIMELINE, PROGRESS AND ALLOCATION**

We use the timeline as guidance to keep us in touch for every step of project making. The timeline created using Microsoft Office Project Professional 2013. This program make the process planning and building project more efficient.





**B.** **TASK DIVISION**

In our project, we have a team consists of three persons. In the software engineering field, people management is essential. Then job division applied based on equality of task, skill specialization like designing the concept, the user interface, the content, the application, and the implementation. of this software project, the task division that can be done is as specified below.

**1. Emily Lomempow :**

- In charge for making the UML.

- Responsible for system integration.

- Responsible for hardware system.

**2. Gandhi Arssegi :**

- In charge for project documentation.

- Responsible for collecting resource and information (inc.software versioning).

- Responsible for maintenance and verification phase.

**3. Yessy Anggraini :**

- In charge for directing project.

- Responsible for database system.

- Responsible for communication system.

**C. RISK MANAGEMENT**

In the process of making G.R.B project, We are analyzing some problems that considerably will becomes our risks for the project. There are:

a. We are making hardware based project, the environment of implementing it, physical factors and everything else beside the technical factors can be obstacle sometimes. Because of that it is hard to analyze the flaw and problems that happen.

Solution : We are making this project modularly and extending maintenance time line.

b. We are code every function from scratch with minimal references and consists of many programming language.

Solution : We are extending the coding phase.

c. We are not having sufficient component and resources, so we are borrowing devices and component beside buying the new ones, impacting to the worst case function issues of the device.

Solution : We always prepared other option for alternating the way out.

**CHAPTER III**

**DESIGN**

Design and concepting is the main step before we go implementing the architecture. Its purpose is for preparing and mapping of what we should do in implementation in order to meet the objective precisely.

**A. UML DIAGRAM**

Unified Modelling Language Diagram are a kind of diagrams that which general-purpose is to make people to understand the design much easier with imaging from seeing the architecture of project from many perspective, and we made 5 of it by using Microsoft Office Visio 2013, there are :

1. Use Case Diagram



Figure 1. Use Case Diagram

In use case diagram, we can see what user can do from using our devices, in it divided into 2 main function such as funciton for garbage user and function for garbage collector.

2. Activity Diagram

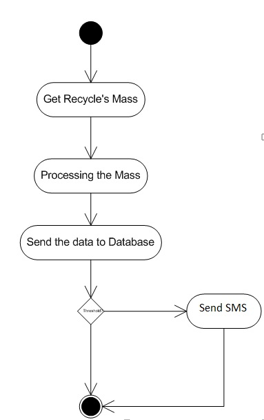


Figure 2. Activity Diagram

As illustrated, we make it like the flowchart. In this diagram we can see the whole activity from hardware perspective, from turning on the system until it ends.

3. Component Diagram



Figure 4. Component Diagram

In this component perspective there are component consists of arduino microcontroller, force sensor, database and how they integrated one to another.

**B. COMPONENTS SPECIFICATION**

**Hardware :**

* 1pc of Force sensitive resistors

Force Sensitive Square 1.75 x 1.5 inch.

Force sensor or weight detector sensor that can measure weight within 100g-10Kg of range.  
Specification :  
- Output : Resistance (No load : 1M Ohm)  
- Bigger force input = smaller the resistance value.  
- Range : 100g - 10Kg.  
- Sensing area : 44.45mm x 38.1mm

* 1pc of Arduino-Due
* 1pc of Internet Modem
* 1pc of Personal Computer
* 1pc Breadboard
* 3pcs Jumper-wire
* 1pc 10k ohm resistor
* 1pc Micro-USB cable
* 1pc GSM SIM Card

**Software :**

* Eclipse IDE
* Arduino IDE 1.5.6 Beta
* Ozeki NG / GAMU
* Notepad++
* Windows OS w/ Java
* MySQL
* XAMPP

**BAB IV**

**IMPLEMENTATION**

**A.** **HARDWARE, OPERATING SYSTEMS AND PROGRAMMING LANGUAGES**

**1. Hardware**

This project are using arduino microcontroller, the DUE series The Arduino Due is a microcontroller board based on the Atmel SAM3X8E ARM Cortex-M3 CPU. It is the first Arduino board based on a 32-bit ARM core microcontroller. It has 54 digital input/output pins (of which 12 can be used as PWM outputs), 12 analog inputs, 4 UARTs (hardware serial ports), a 84 MHz clock, an USB OTG capable connection, 2 DAC (digital to analog), 2 TWI, a power jack, an SPI header, a JTAG header, a reset button and an erase button. With 32-bit proccessing provide faster execution and efficient in handling tasks, we also use this type because newer technology is encouraged to fulfill the project. As for communication the Arduino Due has a number of facilities for communicating with a computer, another Arduino or other microcontrollers, and different devices like phones, tablets, cameras and so on. The SAM3X provides one hardware UART and three hardware USARTs for TTL (3.3V) serial communication. Programmed with programming port that connected to an ATmega16U2, which provides a virtual COM port to software on a connected computer

**2. Operating System**

The operating system is installed in personal computer in order to operate it easier. We are using Windows operating system. Microsoft Windows is a series of [graphical interface](http://en.wikipedia.org/wiki/Graphical_user_interface" \o "Graphical user interface) [operating systems](http://en.wikipedia.org/wiki/Operating_system) developed, marketed, and sold by [Microsoft](http://en.wikipedia.org/wiki/Microsoft" \o "Microsoft) and we are using Windows 8 as our operating system. Windows 8 is a [personal computer](http://en.wikipedia.org/wiki/Personal_computer" \o "Personal computer) [operating system](http://en.wikipedia.org/wiki/Operating_system) developed by [Microsoft](http://en.wikipedia.org/wiki/Microsoft" \o "Microsoft) as part of [Windows NT](http://en.wikipedia.org/wiki/Windows_NT" \o "Windows NT) family of operating systems. Windows 8 introduced major changes to the operating system's platform and [user interface](http://en.wikipedia.org/wiki/Graphical_user_interface" \o "Graphical user interface) to improve its user experience on [tablets](http://en.wikipedia.org/wiki/Tablet_computer" \o "Tablet computer), where Windows was now competing with [mobile operating systems](http://en.wikipedia.org/wiki/Mobile_operating_system" \o "Mobile operating system), including [Android](http://en.wikipedia.org/wiki/Android_%28operating_system%29" \o "Android (operating system)) and [iOS](http://en.wikipedia.org/wiki/IOS" \o "IOS). In particular, these changes included a touch-optimized [Windows shell](http://en.wikipedia.org/wiki/Windows_shell" \o "Windows shell) based on Microsoft's ["Metro"](http://en.wikipedia.org/wiki/Metro_%28design_language%29" \o "Metro (design language)) [design language](http://en.wikipedia.org/wiki/Design_language), the [Start screen](http://en.wikipedia.org/wiki/Start_screen" \o "Start screen) (which displays programs and dynamically updated content on a grid of tiles), a new platform for developing [apps](http://en.wikipedia.org/wiki/Mobile_app" \o "Mobile app) with an emphasis on [touchscreen](http://en.wikipedia.org/wiki/Touchscreen" \o "Touchscreen) input, integration with online services (including the ability to sync apps and settings between devices), and [Windows Store](http://en.wikipedia.org/wiki/Windows_Store" \o "Windows Store), an online store for downloading and purchasing new software. Windows 8 added support for [USB 3.0](http://en.wikipedia.org/wiki/USB_3.0" \o "USB 3.0), [Advanced Format](http://en.wikipedia.org/wiki/Advanced_Format) hard drives, [near field communications](http://en.wikipedia.org/wiki/Near_field_communication" \o "Near field communication), and [cloud computing](http://en.wikipedia.org/wiki/Cloud_computing" \o "Cloud computing). Additional security features were introduced, such as built-in [antivirus software](http://en.wikipedia.org/wiki/Antivirus_software" \o "Antivirus software), integration with [Microsoft SmartScreen](http://en.wikipedia.org/wiki/Microsoft_SmartScreen" \o "Microsoft SmartScreen) [phishing filtering](http://en.wikipedia.org/wiki/Anti-phishing_software) service and support for [UEFI Secure Boot](http://en.wikipedia.org/wiki/UEFI_Secure_Boot" \o "UEFI Secure Boot) on supported devices with [UEFI firmware](http://en.wikipedia.org/wiki/Unified_Extensible_Firmware_Interface" \o "Unified Extensible Firmware Interface), to prevent [malware](http://en.wikipedia.org/wiki/Malware" \o "Malware) from infecting the boot process.

To simplify it we are just use the operating system as a media for us to operate the PC easier, asding the facts for its many feature, none of them is essential except for important software compatibility that we use for our projects, for example Java compatibility. So we are considering older operating system for the real implementation of our project, because it will be more efficient in PC specification budget and data proccessing.

**3. Programming Language**

**DataField.java**

import java.awt.Color;

import java.awt.FlowLayout;

import java.awt.Graphics;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.image.BufferedImage;

import java.io.File;

import java.io.IOException;

import javax.imageio.ImageIO;

import javax.swing.JButton;

import javax.swing.JComponent;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JTextField;

class background extends JComponent {

Image i;

public background(Image i) {

this.i = i;

}

public void paintComponent(Graphics g) {

g.drawImage(i, 0, 0, null);

}

}

public class DataField extends JFrame {

// membuat frame dgn field nama, npm, berat

private JTextField namaField;

private JTextField npmField;

private JTextField beratField;

private String berat;

public DataField(String b) throws IOException {

this.setTitle("GRB Form");

BufferedImage bf = ImageIO.read(new File(

"C:\\Users\\Asus\\workspace\\GRBv2\\src\\wpp1.png"));

this.setContentPane(new background(bf));

namaField = new JTextField(20);

npmField = new JTextField(20);

beratField = new JTextField(b);

beratField.setEditable(false);

JPanel panel = new JPanel();

panel.setSize(300, 150);

JLabel namaLabel = new JLabel("Nama :");

JLabel npmLabel = new JLabel("NPM :");

JLabel beratLabel = new JLabel("Berat :");

JButton btn1 = new JButton("OK");

panel.add(namaLabel);

panel.add(namaField);

panel.add(npmLabel);

panel.add(npmField);

panel.add(beratLabel);

panel.add(beratField);

panel.add(btn1);

panel.setBounds(150, 200, 300, 150);

panel.setBackground(new Color(51, 153, 102));

add(panel);

ButtonHandler handler = new ButtonHandler();

btn1.addActionListener(handler);

berat = b;

}

private class ButtonHandler implements ActionListener {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

String nama = namaField.getText();

String npm = npmField.getText();

DataField.this.dispose(); // untuk mengclose frame

GRBApps.setData(npm, nama, berat);

}

}

}

**DBConnect.java**

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

import java.util.ArrayList;

public class DBConnect {

private Connection con;

private Statement st;

private ResultSet rs;

public DBConnect() {

// constructor to create dbconnector

String username = "root";

String password = "";

String url = "jdbc:mysql://localhost/grb\_berat";

try {

Class.forName("com.mysql.jdbc.Driver");

con = DriverManager.getConnection(url, username, password);

} catch (Exception ex) {

System.out.println("Error: " + ex);

}

}

public void insert(String npm, String nama, String berat) {

// to insert data

try {

st = con.createStatement();

float beratF = Float.parseFloat(berat);

String insertSql = "INSERT berat VALUES(" + "'" + npm + "', '"

+ nama + "', " + beratF + ")";

int val = st.executeUpdate(insertSql);

} catch (Exception ex) {

System.out.println("Error: " + ex);

}

}

public float cekTotal() {

// to get total mass

try {

st = con.createStatement();

String query = "SELECT \* FROM berat";

rs = st.executeQuery(query);

float beratTotal = 0;

while (rs.next()) {

String nS = rs.getString("berat");

float n = Float.parseFloat(nS);

beratTotal += n;

}

return beratTotal;

} catch (Exception ex) {

System.out.println("Error: " + ex);

}

return 0;

}

public void update(String npm, String berat) {

// to update table

try {

st = con.createStatement();

float beratF = Float.parseFloat(berat);

String updateSql = "UPDATE berat SET berat = berat + " + beratF

+ " WHERE npm = '" + npm + "'";

int val = st.executeUpdate(updateSql);

} catch (Exception ex) {

System.out.println("Error: " + ex);

}

}

public void flush() {

// to flush mass records

try {

st = con.createStatement();

String updateSql = "UPDATE berat SET berat = 0";

int val = st.executeUpdate(updateSql);

} catch (Exception ex) {

System.out.println("Error: " + ex);

}

}

public ArrayList<String> cekNPM() {

// to get all NPM

try {

st = con.createStatement();

String query = "SELECT \* FROM berat";

rs = st.executeQuery(query);

ArrayList<String> record = new ArrayList<String>();

int i = 0;

while (rs.next()) {

String nS = rs.getString("npm");

record.add(nS);

i++;

}

System.out.println();

return record;

} catch (Exception ex) {

System.out.println("Error: " + ex);

}

return null;

}

}

**GRBApps.java**

import java.awt.Toolkit;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.net.MalformedURLException;

import java.net.URL;

import java.net.URLConnection;

import java.util.ArrayList;

import javax.swing.JFrame;

public class GRBApps {

private static ArrayList<String> recordNPM = new ArrayList<String>();

private static float threshold = (float) .1;

private static SerialTest serial = new SerialTest();

private static double result = 0;

private static int count = 0;

private static boolean state = true;

private static final float incThreshold = (float) .1;

public static void main(String[] args) {

serial.run();

}

public static void tampilWindow(String berat) throws IOException {

// untuk menampilkan frame yang menerima input nama dan npm, disertakan

// field berisi berat yang tidak bisa diedit

DataField dataField = new DataField(berat);

dataField.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

/\*

Toolkit tk = Toolkit.getDefaultToolkit();

int xSize = ((int) tk.getScreenSize().getWidth());

int ySize = ((int) tk.getScreenSize().getHeight());

dataField.setSize(xSize, ySize);

\*/

dataField.setSize(600, 600);

dataField.setVisible(true);

}

public static void setData(String npm, String nama, String berat) {

if (npm.equals(""))

try {

GRBApps.tampilWindow(berat);

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

else {

DBConnect connect = new DBConnect(); // inisialisasi object

// dbconnect

recordNPM = connect.cekNPM(); // mengambil semua npm yang ada di db

boolean state = true;

for (int i = 0; i < recordNPM.size(); i++) {

// kalau npm sudah ada, maka db akan diupdate

if (npm.equals(recordNPM.get(i))) {

connect.update(npm, berat);

state = false;

}

}

// kalau npm belum ada, akan membuat record baru pada db

if (state == true)

connect.insert(npm, nama, berat);

cekThreshold();

}

}

public static void cekThreshold() {

DBConnect connect = new DBConnect();

float total = connect.cekTotal();

System.out.println(total);

if (total > threshold) {

sms();

threshold = (int) (Math.ceil(total / incThreshold) \* incThreshold);

}

if (threshold > 100) {

connect.flush();

threshold = incThreshold;

}

}

public static void setBerat(double asDouble) {

// TODO Auto-generated method stub

if (result == asDouble)

count++;

else {

result = asDouble;

count = 1;

}

if (count == 3) {

state = false;

String beratSerial = Double.toString(asDouble);

try {

tampilWindow(beratSerial);

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

serial.kill();

}

}

public static void sms() {

try {

URL smsGateway = new URL("http://localhost/grb/sms.php");

URLConnection sg = smsGateway.openConnection();

BufferedReader in = new BufferedReader(new InputStreamReader(

sg.getInputStream()));

String inputLine;

while ((inputLine = in.readLine()) != null)

System.out.println(inputLine);

in.close();

} catch (MalformedURLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

**SerialTest.java**

import jssc.SerialPort;

import jssc.SerialPortEvent;

import jssc.SerialPortEventListener;

import jssc.SerialPortException;

public class SerialTest {

static SerialPort serialPort;

public void run() {

serialPort = new SerialPort("COM15");

try {

//

System.out.println("Port opened: " + serialPort.openPort());

serialPort.setParams(SerialPort.BAUDRATE\_9600,

SerialPort.DATABITS\_8, SerialPort.STOPBITS\_1,

SerialPort.PARITY\_NONE);

int mask = SerialPort.MASK\_RXCHAR + SerialPort.MASK\_CTS

+ SerialPort.MASK\_DSR;// Prepare mask

serialPort.setEventsMask(mask);// Set mask

serialPort.addEventListener(new SerialPortReader());// Add

// SerialPortEventListener

} catch (SerialPortException ex) {

System.out.println(ex);

}

}

public void kill() {

try {

serialPort.closePort();

} catch (SerialPortException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

static class SerialPortReader implements SerialPortEventListener {

public void serialEvent(SerialPortEvent event) {

if (event.isRXCHAR()) {// If data is available

// System.out.println(event.getEventValue());

if (event.getEventValue() > 4) {// Check bytes count in the

// input buffer

// Read data

try {

byte buffer[] = serialPort.readBytes(4);

int asInt = (buffer[0] & 0xFF)

| ((buffer[1] & 0xFF) << 8)

| ((buffer[2] & 0xFF) << 16)

| ((buffer[3] & 0xFF) << 24);

float asFloat = Float.intBitsToFloat(asInt);

System.out.println(asFloat);

double cuttedFloat = Math.floor(asFloat \* 1000) / 1000;

GRBApps.setBerat(cuttedFloat);

// System.out.println(buffer[0]+" "+buffer[1]+" "+buffer[2]+" "+buffer[3]);

} catch (SerialPortException ex) {

System.out.println(ex);

}

}

}

}

}

}

**Arduino Side Code**

/\*

Proyek "Garbage Reader"

Ver 1.2, 23 March 2014

By Emily

Serial Comm test [OK]

\*/

int fsrPin = A0; //pin yang terhubung dengan fsr

int fsrReading; //bacaan analog

int fsrVoltage; //hasil convert ke tegangan

unsigned long fsrResistance; //hasil convert ke hambatan

unsigned long fsrConductance; //hasil convert ke konduktansi

double fsrForce; //hasil convert ke tekanan dalam Newton

float fsrMass;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600); //inisialisasi komunikasi serial

}

void loop() {

// put your main code here, to run repeatedly:

fsrReading = analogRead(fsrPin);

fsrVoltage = map(fsrReading, 0, 1023, 0, 3300); //due 3,3 volt, bacaan daam mV

if(fsrVoltage == 0) ;

else {

fsrResistance = 3300 - fsrVoltage;

fsrResistance \*= 10000; //10k ohm

fsrResistance /= fsrVoltage; //hitungan voltage divider

fsrConductance = 1000000; //dalam microMho

fsrConductance /= fsrResistance;

//sesuai grafik FSR untuk pendekatan tekanan

if (fsrConductance <= 1000) {

fsrForce = fsrConductance / 80.0;

}

else {

fsrForce = fsrConductance - 1000.0;

fsrForce /= 30.0;

}

fsrMass = fsrForce / 9.8;

if (fsrMass < 0.15) ;

else {

fsrMass -= 0.15;

byte\* b = (byte\*) &fsrMass;

Serial.write(b[0]);

Serial.write(b[1]);

Serial.write(b[2]);

Serial.write(b[3]);

}

}

delay(1000);

}

**PHP Side Code**

<?php

//header('Refresh: 3;url=home.php');

########################################################

# Login information for the SMS Gateway

########################################################

$ozeki\_user = "admin";

$ozeki\_password = "admin";

$ozeki\_url = "http://127.0.0.1:9501/api?";

########################################################

# Functions used to send the SMS message

########################################################

function httpRequest($url){

$pattern = "/http...([0-9a-zA-Z-.]\*).([0-9]\*).(.\*)/";

preg\_match($pattern,$url,$args);

$in = "";

$fp = fsockopen("$args[1]", $args[2], $errno, $errstr, 30);

if (!$fp) {

return("$errstr ($errno)");

} else {

$out = "GET /$args[3] HTTP/1.1\r\n";

$out .= "Host: $args[1]:$args[2]\r\n";

$out .= "User-agent: Ozeki PHP client\r\n";

$out .= "Accept: \*/\*\r\n";

$out .= "Connection: Close\r\n\r\n";

fwrite($fp, $out);

while (!feof($fp)) {

$in.=fgets($fp, 128);

}

}

fclose($fp);

return($in);

}

function ozekiSend($phone, $msg, $debug=false){

global $ozeki\_user,$ozeki\_password,$ozeki\_url;

$url = 'username='.$ozeki\_user;

$url.= '&password='.$ozeki\_password;

$url.= '&action=sendmessage';

$url.= '&messagetype=SMS:TEXT';

$url.= '&recipient='.urlencode($phone);

$url.= '&messagedata='.urlencode($msg);

$urltouse = $ozeki\_url.$url;

if ($debug) { }//echo "Request: <br>$urltouse<br><br>"; }

//Open the URL to send the message

$response = httpRequest($urltouse);

if ($debug) {}

/\* echo "Response: <br><pre>".

str\_replace(array("<",">"),array("&lt;","&gt;"),$response).

"</pre><br>"; }

\*/

return($response);

}

########################################################

# GET data from sendsms.html

########################################################

$phonenum = 'PHONE\_NUMBER';

$message = 'Sampah telah mencapai 0.1 kg, silahkan diambil';

$debug = true;

ozekiSend($phonenum,$message,$debug);

?>

Green Recycle Bank telah mencapai batas maksimum.

Mengirimkan SMS kepada petugas.

**BAB V**

**TESTING AND ANALYSIS**

Testing is the last step of building a project. We cannot test an application for every possibility, because the number of ways that a non-trivial computer program may execute is limitless. Thus, testing cannot prove that an application is free from defects, as proofs of correctness can. Testing can only show presence of defect.

Testing is often misunderstood as being primarily a reassurance process, as in “test it to make sure it is right”. Sometimes, this is indeed the goals of testing, especially shortly before delivery, or in regression testing (explained in the next chapter). A major purpose of testing is quite the opposite from reassurance, howewer. The purpose is not to show that the application is satisfactory, but to vigorously determine where the application is not satisfactory. The goals of testing is structural, whereas the goals of other kinds of testing is typically functional.

**A. Testing Form**

Here is the testing form to make sure that G.R.B is ready to be used. We are making this in Bahasa Indonesia, because the form is addressed to Indonesian people.

1. **Testing Sistem**

Sistem harus dapat memberikan informasi kepada user tentang apapun yang sedang terjadi pada system melalui respons waktu yang tepat.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | **Testing System** | **Yes** | **No** | **Komentar** |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |
| 6 |  |  |  |  |
| 7 |  |  |  |  |
| 8 |  |  |  |  |
| 9 |  |  |  |  |
| 10 |  |  |  |  |

1. **Konsistensi dan Standar**

Sistem seharusnya menggunakan penamaan yang standard dan konsisten pada keseluruhan sistem, agar tidak menimbulkan ambiguitas ketika digunakan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | **Consistency Testing** | **Yes** | **No** | **Komentar** |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |

1. **User Interface**

Desain yang dibuat harus mudah dimengerti dan menarik perhatian pengguna untuk menggunakan sistem.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | **User Interface** | **Yes** | **No** | **Komentar** |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |

**BAB VI**

**USER MANUAL**

1. **Installation**
   1. Plug the Green Recycle Bank system in laptop USB port
   2. Make sure laptop already has software OzekiNG, phpmyadmin, and XAMPP installed
   3. On phpmyadmin, create table 'berat' that contains collumn 'npm', 'nama', 'beratF'
   4. Run grb.exe
2. **How-to-Use**

1. Put the empty bottle to Green Recycle Bank

2. Green Recycle Bank will automatically calculate the mass

3. Input your name and NPM to the windows appeared

4. Click OK and the data will automatically transferred into database

5. Repeat first until forth steps to inout another data

1. **Notes**

If total mass of the bottles reach the trigger (trigger can be modified), system will send SMS automatically to the officer's number to take the empty bottles

References :

<http://learn.adafruit.com/force-sensitive-resistor-fsr/overview>

<http://www.digiwarestore.com/dw.php?p=search_result&keyword=sensor&method=contain&category=all&vendor=all&pg=24>

<http://arduino.cc/en/Main/arduinoBoardDue>

<http://git-scm.com/doc>