Chapter 275

Software for monitoring of climate and vibration data collector in egg transporters in commercial aviaries

https://doi.org/10.56238/devopinterscie-275

Amanda dos Santos Dias
E-mail: as.dias@unesp.br

Mario Mollo Neto
E-mail: mario.mollo@unesp.br

ABSTRACT
During the entire automated transport in the egg production line, vibrations and short-term impacts (impacts suffered on the conveyors) cause damage (cracks/cracks) to this production, resulting in the loss of products and damage to the producers. Transport on automated egg production lines from the aviaries to the selection and packaging location should be carried out as soon as possible to reduce initial quality losses. The objective of this project is the creation of application software for the Windows platform, developed in Object Pascal language in Lazarus, for the communication of load and unload, interpretation, recording, editing, and analysis of the recorded data. In addition, having a user-friendly interface allows for optimal management and monitoring. The results obtained met the expectations of the program, achieving a friendly interface that can analyze vibrations and other data present in the transport of commercial eggs.

Keywords: Cracks, Fissures, Eggs, Software. Poultry.

1 INTRODUCTION

According to Gessulli Agribusiness (2017), Brazilian egg production totaled 39 billion units in 2016, a record that places Brazil as the seventh largest producer in the world. Despite having registered an increase of almost 40% since 2010, there is potential for a considerable increase in per capita consumption in the country. Currently, Brazilians consume 190 units per year. The global average is 230, but it reaches 300 eggs per person in many countries, including China, Denmark, and Mexico.

According to the VPA, of the Institute of Agricultural Economics (IEA), in a single city in the administrative region of Marília, beef poultry yielded R $ 1.5 billion that year, and in the State was R $ 2.8 billion, and the municipality in question, has a stock of 32 million head, of which 25 million chickens are producing and 6 million are young. The feed is made with 81 thousand tons of feed per month, 51 thousand tons of corn per month, and 18 thousand tons of soybean meal per month. The production of "in nature", liquid, frozen, powdered, and discarded poultry eggs in the largest egg-producing city in the state, generates 4,000 direct and 8,000 indirect jobs in poultry farming (Government of the State of São Paulo, 2018). It is, therefore, an important production chain in the region with great growth potential.

As an explanatory factor of the structures and behavior of firms, PIZZOLANTE et al. (2011) consider that technological innovations play a very important role. They say that in the activity of laying poultry, the technological innovations that have occurred recently can be considered radical or incremental in systems of breeding, management, and genetics of the birds and that the adoption of technologies by the producer has had as its main objective the reduction of costs through reduction of processing time, use of labor and improvements in the internal and external logistics of the company. Its permanence in the market depends on its efficiency in production.

Brazil has been one of the world's leading producers of table eggs, and with this, Brazilian poultry farming has been incorporating automated systems. The quality of the egg is determined by internal and external factors, so the importance of technologies (SILVA et al., 2015).

The sheds were modernized, having a length of 100 to 150 m and a width of 3 to 15 m. In them are adopted the Californian system (cages with two floors) or battery systems, with six sets of cages superimposed on each other, which requires great production effort (PIZZOLANTE et al., 2011).

According to the previous authors, automation has become present in many farms in the region and ranges from the supply of feed and water to the collection of eggs and, currently, it is possible that the final product, the egg, is collected in the sheds and reaches the consumer without manual contact, in a fully automated process. In addition, they point out that the eggs are collected on mats that lead them to the egg depot where they are taken directly to the washing, selecting, sorting, and packing machines.

The implementation of these new technologies for the automation of egg production environments does not always bring only positive results but may require adjustments and accessories to avoid losses during the production process.

In this research project, we highlight the problem arising from egg transport systems, which can cause productive and financial losses for farms. Nazareno (2012) reports that nowadays there are several losses in the egg production line of the automated transport from the production of these eggs in the sheds to the final process of packaging and shipping, in which the vibrations and impacts of short duration (impacts suffered in the conveyors) cause damage (cracks/cracks).

From the moment the egg is laid to the place of its commercialization, the main objective is to preserve its original quality until it reaches the consumer, therefore, the transport in the automated egg production lines from the aviaries to the place of selection and packaging should be carried out as soon as possible to reduce the initial losses of quality (MAGALHÃES et al., 2012).

Considering the scenario described above, the research project is focused on the search for an additional, incremental technological development, a technological innovation, in the form of a decision support software, amenable to use in the computers of the aviary management system. This can track and give information on the intensity of the impacts suffered by the eggs to give suggestions to managers to mitigate the productive losses arising from the transport systems between the production sheds and the processing and shipping units.

This tool will help companies to reduce the production losses caused by the vibrations of the path of the transporters and also of the exposure to sudden variations of temperature and, expand their market shares, thus gaining greater quality, efficiency, and consequently greater competitiveness and profitability.

2 GOALS

Construction of application software for the Windows platform for the communication of loading and unloading, interpretation, recording, editing, and analysis of the data recorded by the autonomous unit of sensor data collection, and that has a user-friendly graphical interface, which is an uncomplicated way allows efficient management and gives guidelines and suggestions for the mitigation of the problems encountered and facilitates the fight against the points generating productive losses.

3 MATERIALS AND METHODS

Initially, and throughout the development, a prospection of the bibliography in international and national databases of scientific articles will be carried out to deepen the understanding of the state of the art of research on this type of problem of vibrations and impacts on egg transport systems and the issues related to exposure to high temperatures and varying humidity, and the possible risks to production, as well as the development of *hardware and firmware* for sensor data collection, development of supervisory and decision support *software* and the techniques used by peers and the methods most commonly applied to obtain results in an egg production environment. The *Science Direct*, *Scielo,* and *Scopus* databases will prospect preferentially.

To work simultaneously on the computers of the managers, an integration interface will be prepared to interact with the Arduino platform of the prototype to ensure the acquisition of data collected from the sensors, and this *interface* will be developed with a focus on supervision, automatic control of the platform connected to the sensor devices, and the data collected, as well as generating information and recommendations to support the decision of managers.

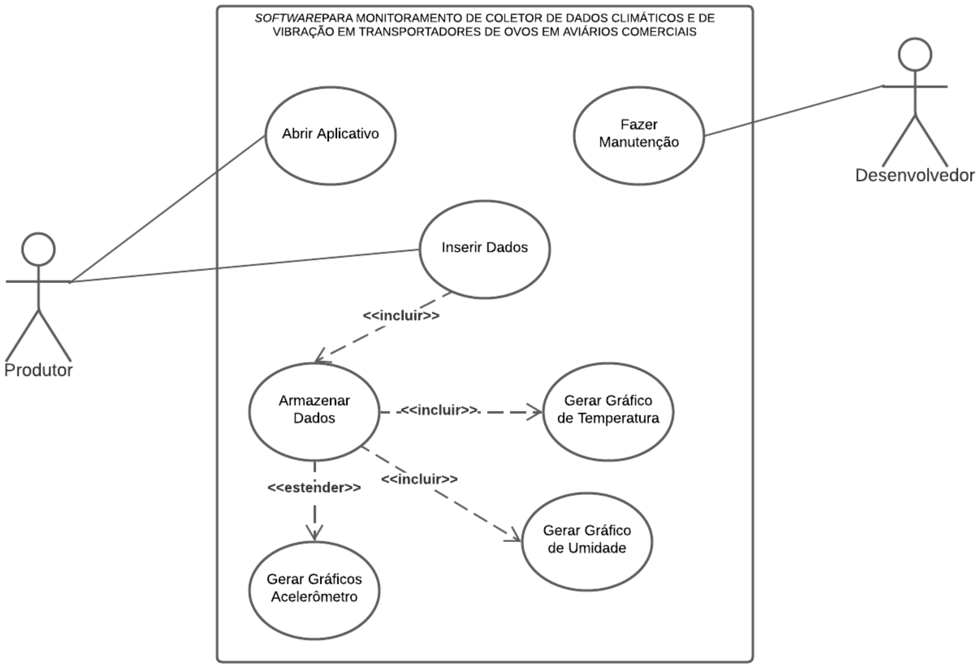
This platform will be developed in a LAZARUS *Rapid Application Development (RAD)* tool with Pascal programming language. This supervisory will allow the interaction of the System User with the developed acquisition hardware, allowing access and information to the users/managers of the system.

After development, tests will be carried out in the laboratory and the production environment directly with the users for the validation of the *software* modules, and the proof of compliance with their respective projected requirements.

4 RESULTS AND DISCUSSION

The system was modeled with UML. To visualize the main functionalities of the system and the interaction with users, we used the Use Case Diagram (Figure 4.1).

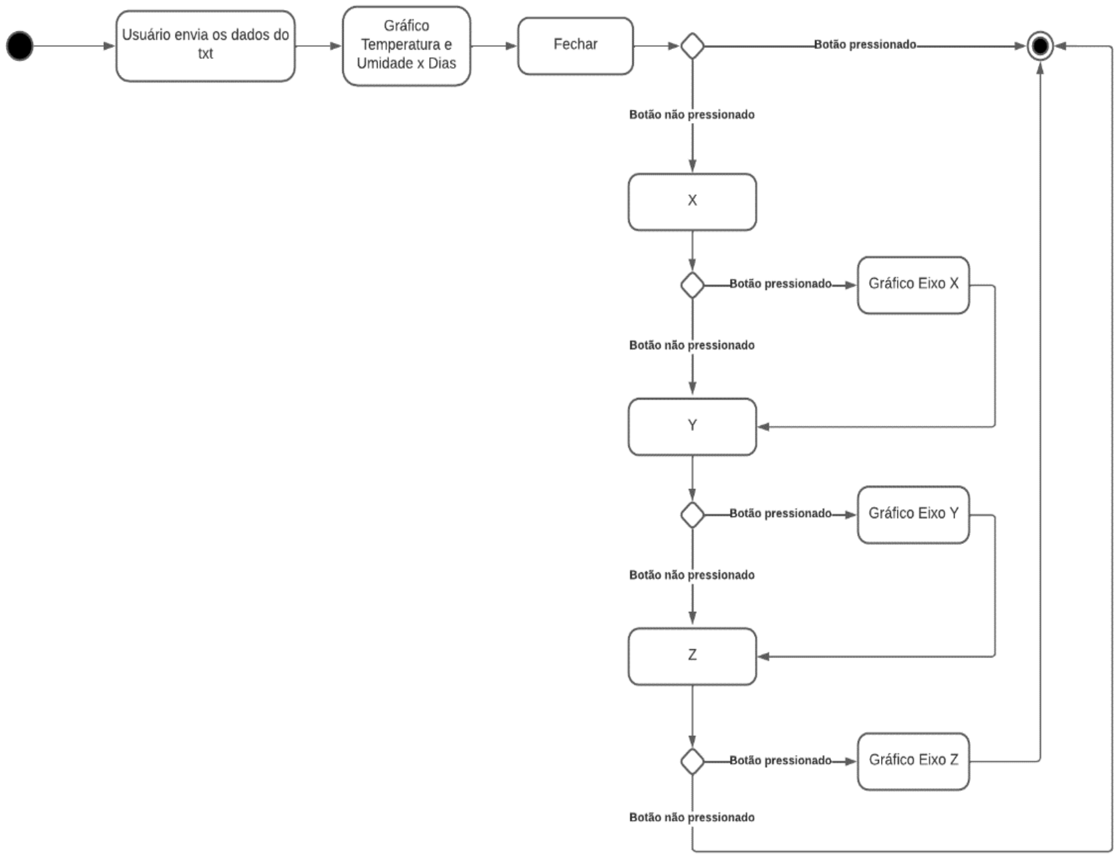
Figure 0. 1 - Use Case Diagram.



Source: Prepared by the author.

Afterward, the Activity Diagram was executed (Figure 4.2), demonstrating the logic of the algorithm through the steps performed, simplifying and clarifying the system.

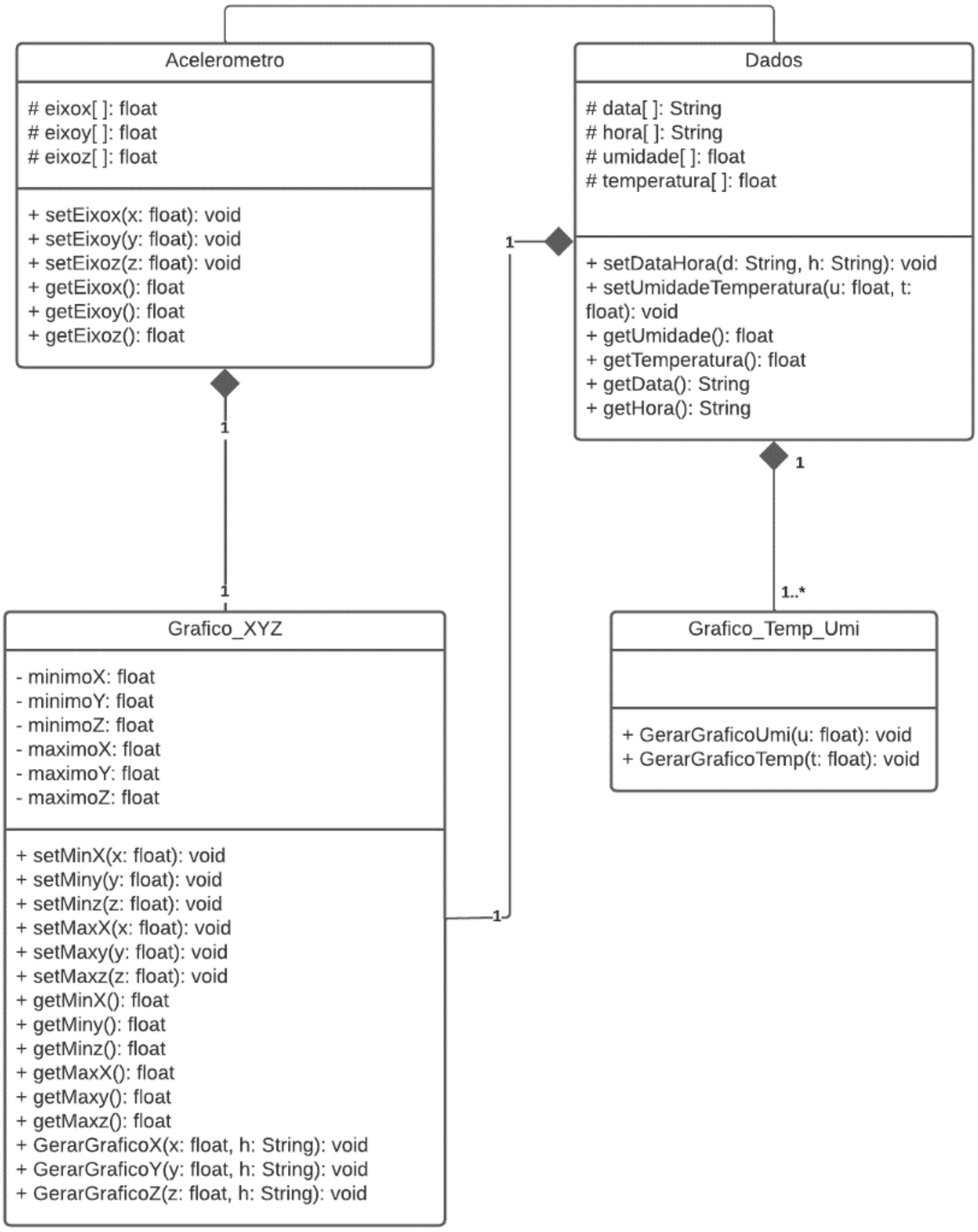
Figure 0. 2 - Activity Diagram.



Source: Prepared by the author.

The Class Diagram (Figure 4.3) was used to illustrate the model, express the needs, and better understand the overview of the schemes.

Figure 0. 3 - Class Diagram.

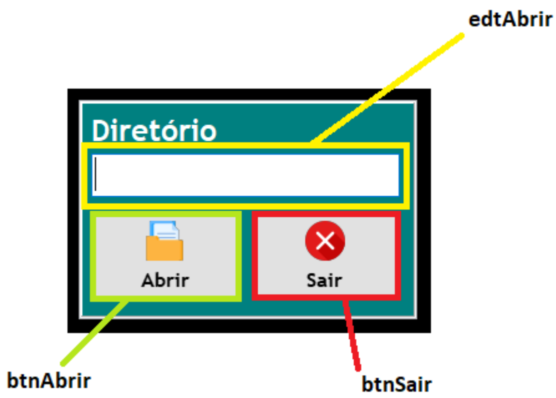


Source: Prepared by the author.

To build the *software*, a project was created in the Lazarus program and a main form, with the name of frmPrincipal. In addition, the form was divided into three parts: directory, graphics, and accelerometer.

The directory is the area where the user will upload the txt file, sending the data through edtOpen by clicking the btnOpen button (Figure 4.4). If you click the btnExit button, the application will close. To open the directory used a TOpenDialog component (dialogOpen).

Figure 0. 4 - Directory Components.



Source: Prepared by the author.

The first step in programming the algorithm was to declare the global variables (Figure 4.5), so you could use them anywhere in the project. In the OnClick event of the button (when the user clicks) the code in Figure 4.6 was programmed, which allows you to open the directory and choose the txt file.

Figure 0. 5 - Global variables used.

```
var
  frmPrincipal: TfrmPrincipal;
  data: array [0..197] of String;
  hora: array [0..197] of String;
  umidade: array [0..197] of double;
  temperatura: array [0..197] of double;
  eixox: array [0..197] of double;
  eixoy: array [0..197] of double;
  eixoz: array [0..197] of double;
  Linhas: TStringList;
  Colunas: TStringList;
  i: integer;
```

Source: Prepared by the author.

Figure 0. 6 - btnOpen OnClick Event.

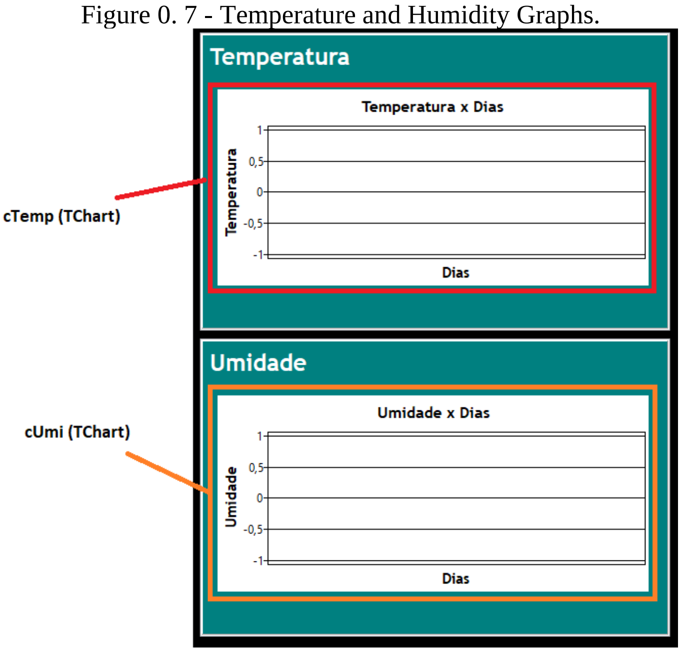
```
procedure TfrmPrincipal.btnAbrirClick(Sender: TObject);
begin
  btnX.Enabled := true;
  btnY.Enabled := true;
  btnZ.Enabled := true;
  Linhas := TStringList.Create;
  Colunas := TStringlist.Create;
  Colunas.Delimiter := ';';
  Colunas.StrictDelimiter := True;

  if (dialogAbrir.Execute) then
     begin
          if (dialogAbrir.FileName <> '') then
          begin
               edtAbrir.Text := dialogAbrir.FileName;
               try
                 Linhas.LoadFromFile(edtAbrir.Text);
                 for i := 1 to Linhas.Count-1 do
                 begin
                   Colunas.DelimitedText := Linhas[i];
                   data[i-1] := Colunas[0];
                   hora[i-1] := Colunas[1];
                   umidade[i-1] := StrToFloat(Colunas[2]);
                   temperatura[i-1] := StrToFloat(Colunas[3]);
                   eixox[i-1] := StrToFloat(Colunas[4]);
                   eixoy[i-1] := StrToFloat(Colunas[5]);
                   eixoz[i-1] := StrToFloat(Colunas[6]);
                 end;
               finally
                 Linhas.Free;
               end;
          end;
     end;
```

Source: Prepared by the author.

Two graphs will make the correlation of temperature x days and humidity x days (Figure 4.7). To assemble the graphs, two TChart (cTemp and cUmi), two TLineSeries (TempDias and UmiDias), and two TListChartSource (lcsTemp and lcsUmi) were used.

Figure 0. 7 - Temperature and Humidity Graphs.



Source: Prepared by the author.

For the charts to be created and shown as soon as you click on btnOpen, still in the OnClick event of the button, the code of Figure 4.8 was placed.

Figure 0. 8 - OnClick Event: Graphic.

```
  // Gráfico
  for i := 0 to cTemp.SeriesCount - 1 do
      begin
        if (cTemp.Series[i] is TBarSeries) then
        begin
           (cTemp.Series[i] as TBarSeries).Clear;
        end;
        if (cTemp.Series[i] is TPieSeries) then
        begin
           (cTemp.Series[i] as TPieSeries).Clear;
        end;
      end;
  lcsTemp.Clear;
  for i := 0 to 196 do
      begin
        lcsTemp.AddXYList(i,temperatura[i],data[i],RGBToColor(random(255),random(255),random(255)));
      end;

  for i := 0 to cUmi.SeriesCount - 1 do
      begin
        if (cUmi.Series[i] is TBarSeries) then
        begin
           (cUmi.Series[i] as TBarSeries).Clear;
        end;
        if (cUmi.Series[i] is TPieSeries) then
        begin
           (cUmi.Series[i] as TPieSeries).Clear;
        end;
      end;
  lcsUmi.Clear;
  for i := 0 to 196 do
      begin
        lcsUmi.AddXYList(i,umidade[i],data[i],RGBToColor(random(255),random(255),random(255)));
      end;
end;
```

Source: Prepared by the author.

In the OnClick event of the btnExit button, the code below was placed, to close the program.

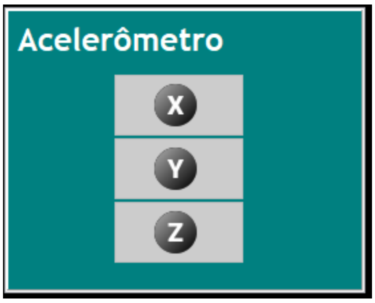
Figure 0. 9 - btnExit OnClick event.

```
procedure TfrmPrincipal.btnSairClick(Sender: TObject);
begin
  frmPrincipal.Close;
end;
```

Source: Prepared by the author.

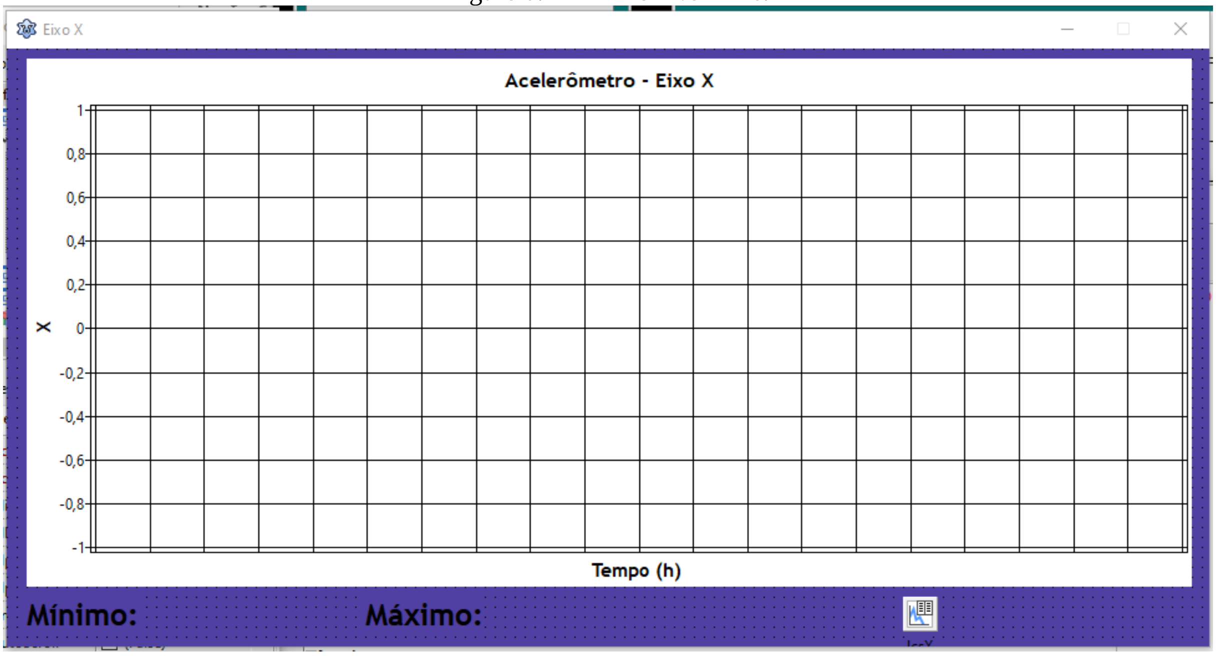
To visualize the axes of the accelerometer, 3 buttons were created: btnX, btnY, and btnZ (Figure 4.10). Each of these buttons will open a different form, showing the time x-axis chart (Figure 4.11) and the minimum and maximum values. With this, the producer can see if the impact has occurred, and if the values are varying.

Figure 0. 10 - Buttons of the accelerometer axes.



Source: Prepared by the author.

Figure 0. 11 - Axis x vs Time.



Source: Prepared by the author.

To open these axis forms, the code below was placed in the OnClick events of the X, Y, and Z buttons.

Figure 0. 12 - Event OnClick axes.

```
procedure TfrmPrincipal.btnXClick(Sender: TObject);
begin
  frmX.Show;
end;

procedure TfrmPrincipal.btnYClick(Sender: TObject);
begin
  frmY.Show;
end;

procedure TfrmPrincipal.btnZClick(Sender: TObject);
begin
  frmZ.Show;
end;
```

Source: Prepared by the author.

In addition, to have access to the other forms, you must add them in the frmPrincipal uses (Figure 4.13), and in the frmX, frmY, and frmZ uses, you must also add frmPrincipal (Figure 4.14).

Figure 0. 13 - Uses of frmPrincipal.

```
implementation
uses Splash, X, y, Z;
```

Source: Prepared by the author.

Figure 0. 14 - Uses of frmX.

```
uses Principal;
```

Source: Prepared by the author.

To plot the axes, the code was placed in the OnShow event of frmX (Figure 4.15), frmY (Figure 4.16), and frmZ (Figure 4.17).

Figure 0. 15 - frmX OnShow Event.

```
procedure TfrmX.FormShow(Sender: TObject);
var
  minimo: double;
  maximo: double;
begin
  minimo := eixox[0];
  maximo := eixox[0];
  // Gráfico
  for i := 0 to cX.SeriesCount - 1 do
      begin
          if (cX.Series[i] is TBarSeries) then
        begin
            (cX.Series[i] as TBarSeries).Clear;
        end;
        if (cX.Series[i] is TPieSeries) then
        begin
            (cX.Series[i] as TPieSeries).Clear;
        end;
      end;
  lcsX.Clear;
  for i := 0 to 196 do
      begin
        if (eixox[i] <= minimo) then
        begin
           minimo := eixox[i];
        end;
        if (eixox[i] >= maximo) then
        begin
           maximo := eixox[i];
        end;
        lcsX.AddXYList(i,eixox[i],hora[i],RGBToColor(random(255),random(255),random(255)));
      end;
  lblMinimo.Caption := 'Mínimo: ' + FloatToStr(minimo);
  lblMaximo.Caption := 'Máximo: ' + FloatToStr(maximo);
end;
```

Source: Prepared by the author.

Figure 0. 16 - frmY OnShow Event.

```
procedure TfrmY.FormShow(Sender: TObject);
var
  minimo: double;
  maximo: double;
begin
  minimo := eixoy[0];
  maximo := eixoy[0];
  // Gráfico
  for i := 0 to cY.SeriesCount - 1 do
      begin
          if (cY.Series[i] is TBarSeries) then
        begin
            (cY.Series[i] as TBarSeries).Clear;
        end;
        if (cY.Series[i] is TPieSeries) then
        begin
            (cY.Series[i] as TPieSeries).Clear;
        end;
      end;
  lcsY.Clear;
  for i := 0 to 196 do
      begin
        if (eixoy[i] <= minimo) then
        begin
           minimo := eixoy[i];
        end;
        if (eixoy[i] >= maximo) then
        begin
           maximo := eixoy[i];
        end;
        lcsY.AddXYList(i,eixoy[i],hora[i],RGBToColor(random(255),random(255),random(255)));
      end;
  lblMinimo.Caption := 'Mínimo: ' + FloatToStr(minimo);
  lblMaximo.Caption := 'Máximo: ' + FloatToStr(maximo);
end;
```

Source: Prepared by the author.

Figure 0. 17 - frmZ's OnShow Event.

```
procedure TfrmZ.FormShow(Sender: TObject);
var
  minimo: double;
  maximo: double;
begin
  minimo := eixoz[0];
  maximo := eixoz[0];
  // Gráfico
  for i := 0 to cZ.SeriesCount - 1 do
      begin
          if (cZ.Series[i] is TBarSeries) then
        begin
            (cZ.Series[i] as TBarSeries).Clear;
        end;
        if (cZ.Series[i] is TPieSeries) then
        begin
            (cZ.Series[i] as TPieSeries).Clear;
        end;
      end;
  lcsZ.Clear;
  for i := 0 to 196 do
      begin
        if (eixoz[i] <= minimo) then
        begin
           minimo := eixoz[i];
        end;
        if (eixoz[i] >= maximo) then
        begin
           maximo := eixoz[i];
        end;
        lcsZ.AddXYList(i,eixoz[i],hora[i],RGBToColor(random(255),random(255),random(255)));
      end;
  lblMinimo.Caption := 'Mínimo: ' + FloatToStr(minimo);
  lblMaximo.Caption := 'Máximo: ' + FloatToStr(maximo);
end;
```

Source: Prepared by the author.

To make the program more structured, a form was created to be the Splash screen, with the name of frmSplash (Figure 4.17). In this form were placed a TProgressBar and a TTimer to together starts the screen and open after a certain time, the frmPrincipal. The code in Figure 4.18 was programmed into the OnTimer event and the OnCreate.

Figure 0. 18 - frmSplash.

Source: Prepared by the author.

Figure 0. 19 - OnTimer and OnCreate event.

```
procedure TfrmSplash.Timer1Timer(Sender: TObject);
begin
  // A barra andará de 5 em 5
  ProgressBar1.Position := ProgressBar1.Position + 3;

  // Quando atingir 100 o outro forms abre
  if ProgressBar1.Position >= 80 then
  begin
    frmPrincipal.Show;
  end;
end;

procedure TfrmSplash.FormCreate(Sender: TObject);
begin
  Timer1.Enabled := true;
  ProgressBar1.Position := 0;
end;
```

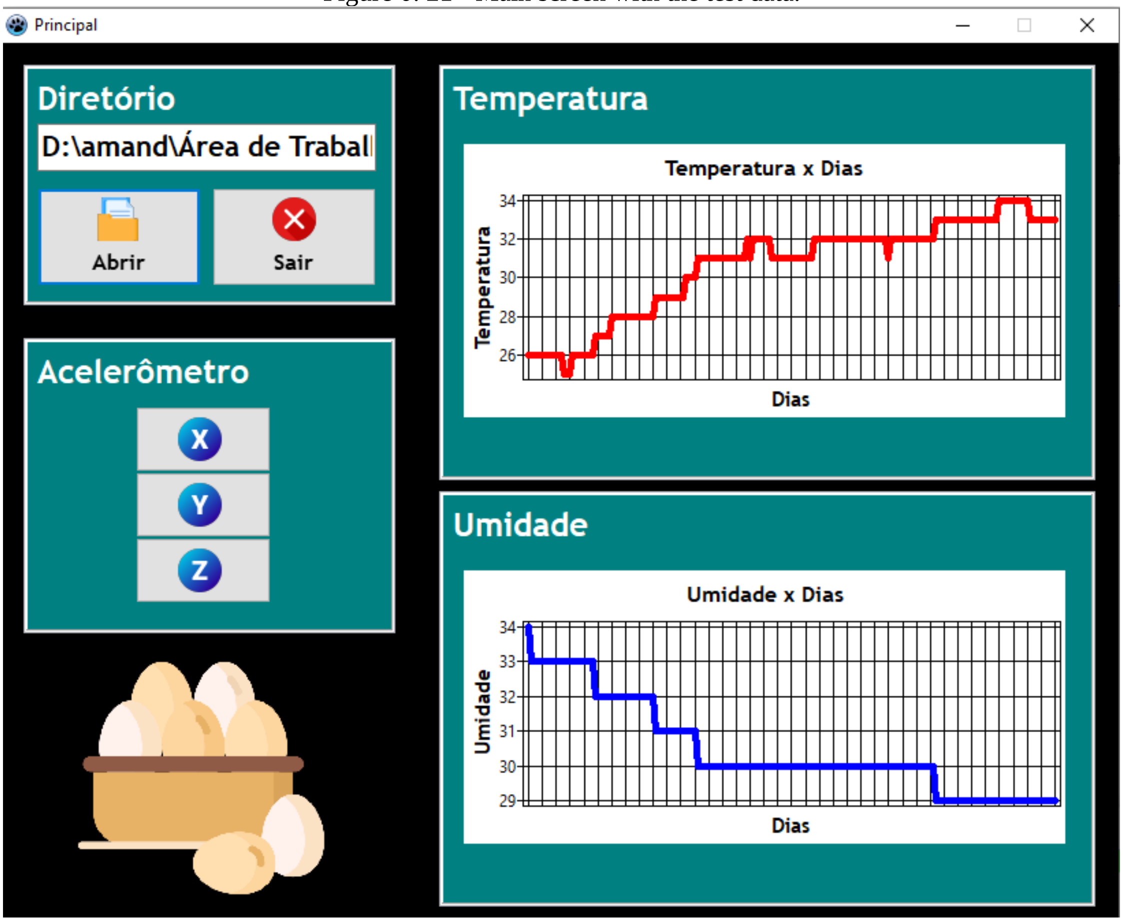
Source: Prepared by the author.

Data were obtained through tests carried out by Arduino at Unesp Tupã, with this obtained the following results:

Figure 0. 20 - Splash screen.

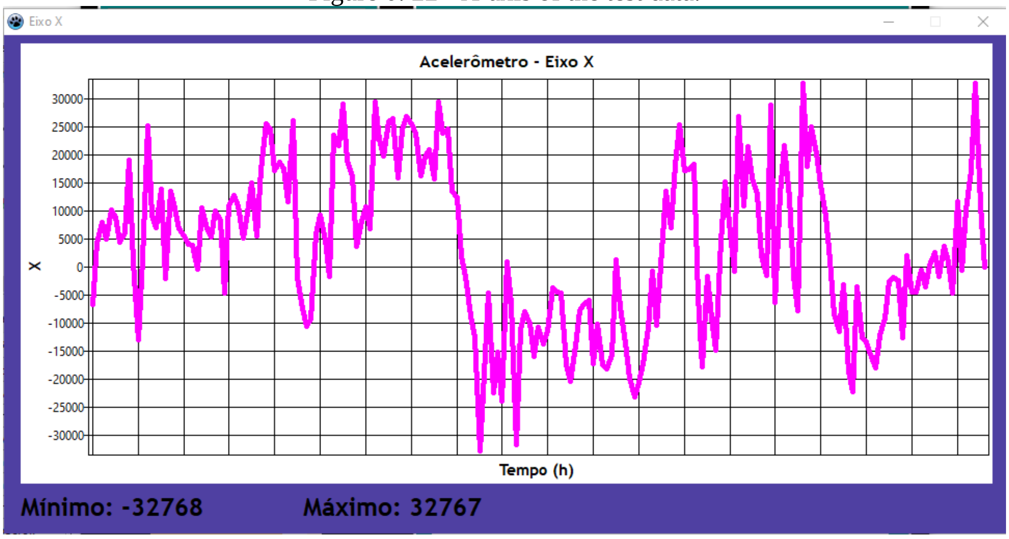
Source: Prepared by the author.

Figure 0. 21 - Main screen with the test data.



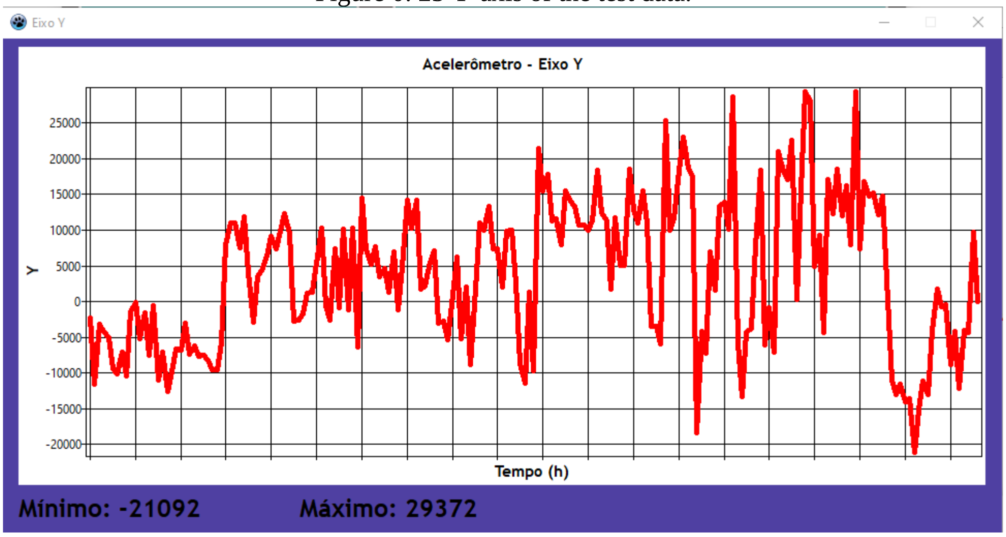
Source: Prepared by the author.

Figure 0. 22 - X-axis of the test data.



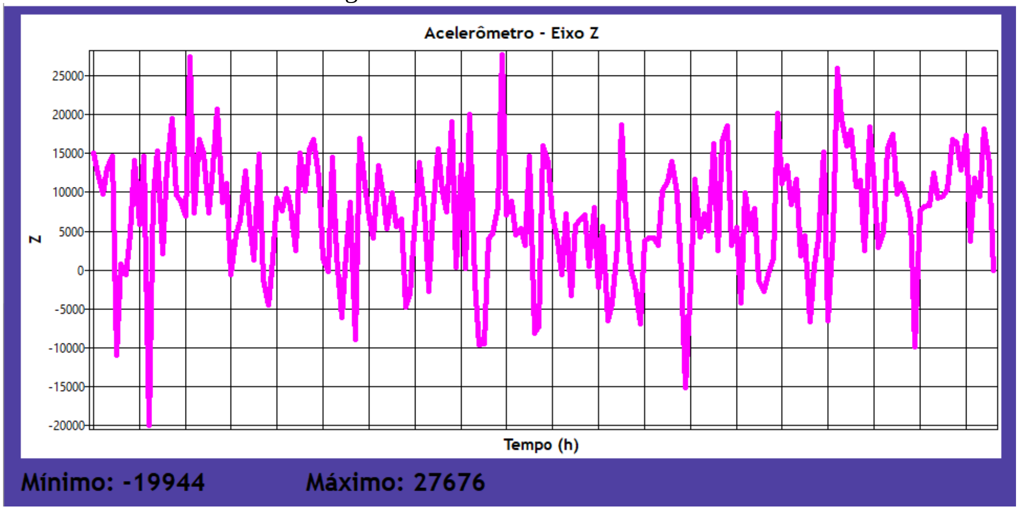
Source: Prepared by the author.

Figure 0. 23 Y-axis of the test data.



Source: Prepared by the author.

Figure 0. 24 Z-axis of the test data.



Source: Prepared by the author.

5 CONCLUSIONS

Performing the tests with the data from Unesp Tupã, it can be concluded that the objectives of the project were met since it was possible to analyze and show the data regarding temperature, humidity, and accelerometer. With the analysis of the data we can conclude that the temperature was increasing and the humidity decreasing, in addition, vibration occurred in the x, y, and z axes.

ACKNOWLEDGMENT

The authors thank CNPq for the PIBIT initiation scholarship (Amanda) and the productivity scholarship in technological development DT-II Process: 313339/2019-8 (Mario).

REFERENCES

GESSULLI AGRIBUSINESS. **Brasil é sétimo maior produtor mundial de ovos. Revista digital Avicultura Industrial.com.br**. Disponível em: <https://www.aviculturaindustrial.com.br/imprensa/brasil-e-setimo-maior-produtor-mundial-deovos/20171113-144114-a215> Acesso em: 17 set. 2019.

GOVERNO DO ESTADO DE SÃO PAULO. **Com 5 bilhões de ovos por ano, Bastos é maior produtora de SP.** Seg, 22/07/2019 - 12h01 - Do Portal do Governo. Disponível em: <http://www.saopaulo.sp.gov.br/spnoticias/com-5-bilhoes-de-ovos-por-ano-bastos-e-maior-produtordo-estado/> Acesso em: 17 set. 2019.

GOVERNO DO ESTADO DE SÃO PAULO. IEA divulga o Valor da Produção Agropecuária Paulista Regional: Resultado Final 2018. **Secretaria de Agricultura e Abastecimento - Instituto de Economia Agrícola (IEA)**, 2019. Disponível em: <http://www.iea.sp.gov.br/out/TerTexto.php?codTexto=14613> Acesso em: 22 set. 2019.

MAGALHÃES, A. P. C; CURVELLO, F. A.; MORENZ, M. J.; CALIXTO, L. F.; REZENDE, S. R. F. Qualidade de ovos comerciais de acordo com a integridade da casca, tipo de embalagem e tempo de armazenamento. **Revista de Ciência da Vida**, v.32, p.51-62, 2012.

NAZARENO, A. C.; **Ambiência pré-porteira; avaliação das condições bioclimáticas e das operações pré-eclosao na qualidade de pintos de corte**. Tese (Doutorado em física do Ambiente Agrícola) – Escola Superior de Agricultura "Luiz de Queiros", Universidade de São Paulo, Piracicaba, 2012, p. 85-117.

PIZZOLANTE, C.C.; SALDANHA, E.S.P.B.; MORAES, J.E.; KAKIMOTO, S.K. A trajetória tecnológica na avicultura de postura. **APTA - Pesquisa & Tecnologia**, vol. 8, n. 2, Jul-Dez 2011. Disponível em: <http://aptaregional.sp.gov.br/acesse-os-artigos-pesquisa-e-tecnologia/edicao-2011/2011-julho-dezembro/1010-a-trajetoria-tecnologica-na-avicultura-de-postura/file.html> Acesso em: 17 set. 2019.

SILVA, R.C.; NASCIMENTO, J. W. B.; OLIVEIRA, D. L.; FURTADO, D. A. Termohigrometria no transporte e na qualidade de ovos destinados ao consumo humano. **Revista Brasileira de Engenharia Agrícola e Ambiental**, Campina Grande, PB, UAEA/UFCG v.19, n.7, p.668–673, 2015.