

CENTRO ESTADUAL DE EDUCAÇÃO  
TECNOLÓGICA PAULA SOUZA  
ESCOLA TÉCNICA ESTADUAL DE DIADEMA

LUCAS CARVALHO

MÁRCIA CRISTINA

TAINAN ARAÚJO

THAYNÁ EMANOELLY

SHUFFLE

DIADEMA - SP

2012

LUCAS CARVALHO  
MÁRCIA CRISTINA  
TAINAN ARAÚJO  
THAYNÁ EMANOELLY

SHUFFLE

Trabalho de Conclusão de Curso  
apresentado como exigência parcial, para  
obtenção do título técnico de Informática,  
ETEC Juscelino Kubitschek de Oliveira.

**Orientador:** Prof<sup>o</sup> Paulo Sérgio Félix da  
Silva

DIADEMA – SP

2012

LUCAS CARVALHO  
MÁRCIA CRISTINA  
TAINAN ARAÚJO  
THAYNÁ EMANOELLY

SHUFFLE

Orientador: \_\_\_\_\_

Nome:  
Instituição:

Examinador (a): \_\_\_\_\_

Nome:  
Instituição:

Examinador (a): \_\_\_\_\_

Nome:  
Instituição:

Examinador (a): \_\_\_\_\_

Nome:  
Instituição:

Diadema, \_\_\_\_/\_\_\_\_/\_\_\_\_.

## **DEDICATÓRIA**

Dedicamos esse trabalho aos nossos amigos, parentes e professores que colaboraram com sua paciência, carinho e compreensão para que superássemos os momentos difíceis e chegássemos ao fim desta longa caminhada.

## **Epígrafe**

“Os professores são parte fundamental em nossas vidas, se não os valorizarmos, quem vai ensinar? O ensino nunca morrerá, mas nós deixaremos de aprender.”

*Tainan Araújo*

## **RESUMO**

O Shuffle foi pensado para que o professor possa avaliar o desempenho do aluno de forma mais precisa possível, evitando a interferência de outros alunos na prova. O programa facilita a vida dos professores na hora de registrar as nota dos alunos, incentiva os alunos a estudarem para as provas e também tem acesso às notas logo após a prova.

## **ABSTRACT**

The Shuffle was designed so that the teacher can assess student performance more accurately as possible, avoiding interference with other students in the competition. The program makes life easier for teachers in time to record the notes of the students, encourages students to study for exams and also have access to the notes immediately after the race.

# SUMÁRIO

<b>1. INTRODUÇÃO</b> .....	12
1.1 MOTIVAÇÃO .....	12
1.2 METODOLOGIA .....	12
1.3 PROBLEMATIZAÇÃO.....	12
1.4 PESQUISA DE CAMPO .....	12
1.5 INTRODUÇÃO.....	15
<b>2. ANÁLISE DO SISTEMA</b> .....	16
2.1 O SOFTWARE (SHUFFLE) .....	16
2.2 DICIONÁRIO DE DADOS .....	16
2.3 DER- DIAGRAMA DE ENTIDADE RALACIONAL .....	17
2.4 CRONOGRAMA .....	18
2.5 RELATÓRIO DE CUSTOS.....	18
2.6 HORAS DO PROJETO .....	18
2.7 CUSTO DO PROJETO.....	19
<b>3. HISTÓRICO DA LINGUAGEM DE PROGRAMAÇÃO E TECNOLOGIAS</b> ....	19
3.1 VISUAL BASIC.....	19
3.2 SQL.....	20
3.3 DIAGRAMA DE CLASSES.....	21

3.4 DIAGRAMA DE CASOS DE USO.....	22
<b>4. ESTUDO DA VIABILIDADE.....</b>	<b>22</b>
4.1 VIABILIDADE .....	22
4.2 CRONOGRAMA FINAL .....	23
4.3 MANUAL DO USUÁRIO .....	23
4.3.1 Professor .....	23
4.3.2 Aluno .....	26
<b>5. CONSIDERAÇÕES FINAIS E REFERÊNCIAS BIBLIOGRAFICAS .....</b>	<b>27</b>
5.1 CONSIDERAÇÕES FINAIS.....	27
5.2 REFERÊNCIAS BIBLIOGRÁFICAS .....	28
<b>APÊNDICE.....</b>	<b>29</b>



## Lista de Tabelas

Tabela 1 Entidade TbAlunos.....	16
Tabela 2 Entidade TbRandon.....	16
Tabela 3 Entidade TbPerguntas.....	16
Tabela 4 Entidade TbDadosprova .....	17
Tabela 5 Cronograma.....	18
Tabela 6 Relatório de Custos.....	18
Tabela 7 Horas do Projeto.....	18
Tabela 8 Custos do projeto.....	19
Tabela 9 Cronograma final.....	23

## Lista de Ilustrações

Figura1 - Diagrama de Entidade Relacional- Feito.....	17
Figura2 - Diagrama de Classes- Feito em 25 de setembro de 2012- Visio.....	21
Figura3 - Diagrama de Casos de Uso- Feito em 24 de setembro de 2012- ArgoUml....	22
Figura4 -Tela Principal professor.....	23
Figura5 - Cadastrar perguntas.....	24
Figura6 - Consultar pergunta.....	24
Figura7 - Alterar pergunta.....	25
Figura 8 - Excluir perguntas.....	25
Figura 9 - Notas.....	26
Figura10 - Tela principal aluno.....	26
Figura11 - Tela prova.....	27

## **Lista de Gráficos**

Gráfico1 - Quantidade de entrevistados.....	12
Gráfico2 - Idade dos entrevistados.....	12
Gráfico3 - Importância do desempenho dos alunos.....	13
Gráfico4 - Utilização do programa.....	13
Gráfico5 - Aceitação do projeto.....	14
Gráfico6 - Programas destinados à professores.....	14

# **1. INTRODUÇÃO**

## **1.1 MOTIVAÇÃO**

Hoje a demora da correção e entrega das notas, é uma situação que prejudica não só os alunos como também os professores. A maioria dos educadores ministram aulas em diversas escolas e com isso tem cada vez mais testes para corrigir, isso levou-nos a desenvolver um programa que facilitasse e agilizasse esse processo, para registrar notas mais rapidamente e mais justas (sem erros).

## **1.2 METODOLOGIA**

Para o desenvolvimento deste trabalho foi realizada uma pesquisa bibliográfica através de sites escolares para facilitar na confecção do nosso sistema.

Além disso, foram feitas pesquisa quantitativa com professores de algumas instituições de ensino para obter informações de como o projeto desenvolvido será aceito no mercado.

## **1.3 PROBLEMATIZAÇÃO**

Como evitar “colas” e avaliar de forma rápida e correta o desempenho dos alunos?

## **1.4 PESQUISA DE CAMPO**

Pesquisa realizada em quatro instituições de ensino, totalizando quarenta professores entrevistados, sendo dez de cada instituição: ETEC Juscelino Kubitschek de Oliveira, E.E. Anecondes Alves Ferreira, E.E WallaceCockrane Simonsen e E.E. Simon Bolivar. Os gráficos abaixo demonstram o percentual de aceitação do software;

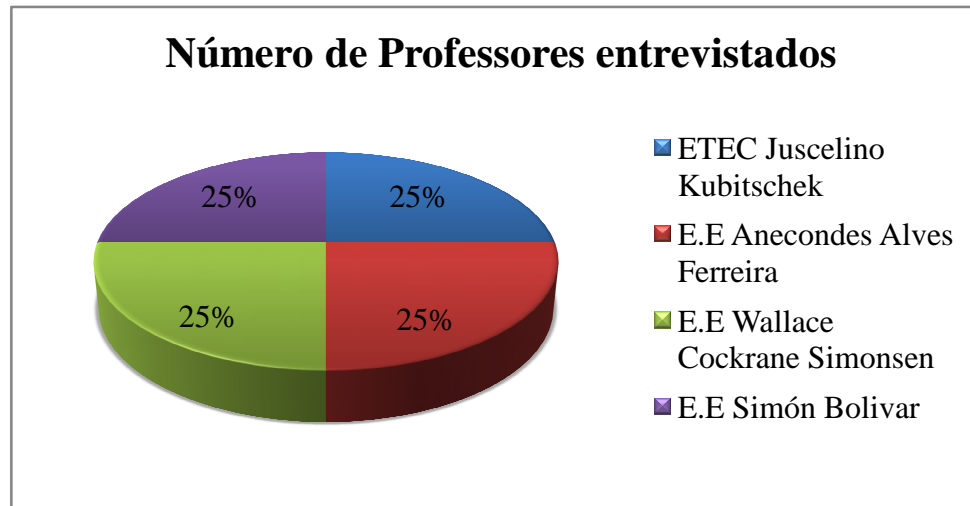


Gráfico1: Quantidade de entrevistados

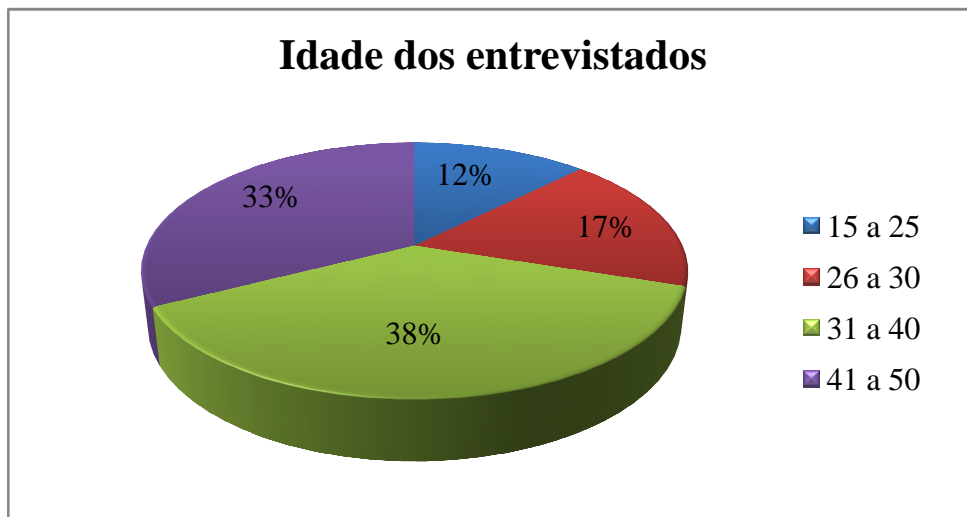
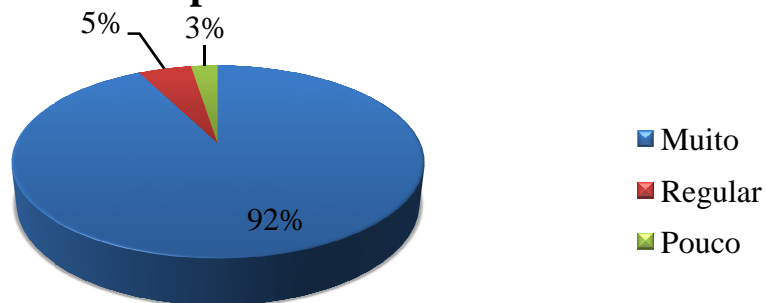


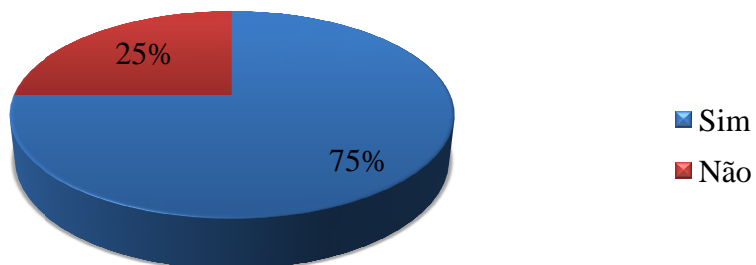
Gráfico 2: Idade dos entrevistados

**Você acha importante ter um conhecimento mais preciso do desempenho dos alunos?**



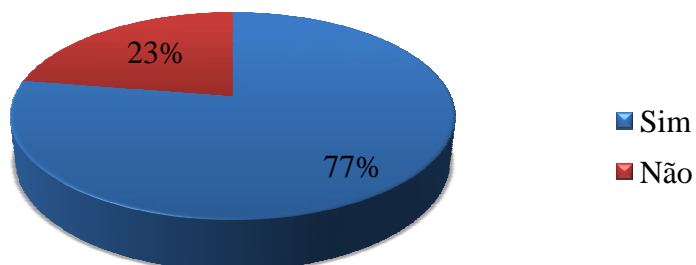
*Gráfico 3: Importância do desempenho dos alunos*

**Você usaria um programa que minimizaria as "colas" na aplicação das avaliações?**



*Gráfico 4: Utilização do programa*

**Você gostaria de um programa que facilitasse o processo de correção das provas?**



*Gráfico 5: Aceitação do projeto*

**A instituição em que trabalha possui um programa que te auxilie em dias de prova?**



*Gráfico 6: Programas destinados à professores*

## 1.5INTRODUÇÃO

Temos em vista que em todas as instituições de ensino possui alunos com dificuldades de aprendizado, por isso nas provas podem haver as famosas “colas”, fazendo com que os alunos recebam notas totalmente desproporcionais aos seus conhecimentos, sendo assim o programa tem o intuito de acabar com as “colas” e registrar notas mais proporcionais e justas a todos os estudantes. Temos também problemas nas correções, são demoradas e às vezes perdidas nesse processo por isso o programa vem com mais uma inovação, corrigir de forma ágil e prática e passar diretamente para o diário de notas do professor.

## 2. ANÁLISE DO SISTEMA

### 2.1 O SOFTWARE (SHUFFLE)

O software permitirá aos professores maior facilidade de precisão na aplicação de provas e na avaliação de desempenho de cada aluno. Será um aplicativo voltado para a agilização das correções das provas.

### 2.2 DICIONÁRIO DE DADOS

<b>Campo</b>	<b>Tipo</b>	<b>Nulo</b>	<b>Chave</b>
RG	Char (9)	Não	PK
Nome	Varchar (50)	Não	
Nota	Numeric (4,2)	Sim	
Respostas	Varchar (50)	Sim	

*Tabela1:Entidade TbAlunos*

<b>Campo</b>	<b>Tipo</b>	<b>Nulo</b>	<b>Chave</b>
Id Prova	Varchar (100)	Sim	
OrdemPerg	Varchar (200)	Sim	
OrdemResp	Varchar (200)	Sim	
RG	Char (9)	Sim	FK

*Tabela 2: Entidade TbRandom*

<b>Campo</b>	<b>Tipo</b>	<b>Nulo</b>	<b>Chave</b>
Id Perg	Int	Não	
Perguntas	Varchar (1024)	Não	
Pontos	Numeric (3,2)	Sim	
Alter Ok	Int	Sim	
Alter 1	Varchar (255)	Sim	
Alter 2	Varchar (255)	Sim	
Alter 3	Varchar (255)	Sim	
Alter 4	Varchar (255)	Sim	

*Tabela 3: Entidade TbPerguntas*



Campo	Tipo	Nulo	Chave
Nperguntas	Int	Não	
Matéria	Varchar (20)	Sim	
Professor	Varchar (50)	Sim	

Tabela 4: Entidade TbDadosprova

### 2.3 DER – DIAGRAMA DE ENTIDADE RELACIONAL

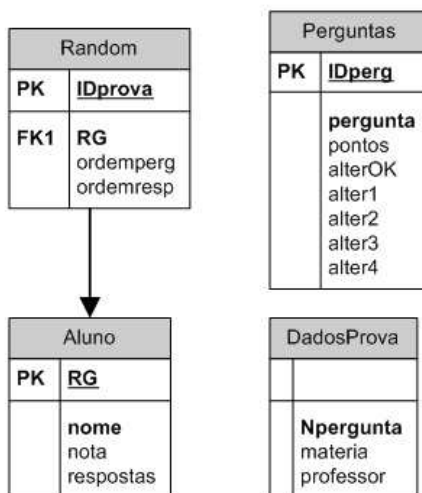


Figura 1: Diagrama de Entidade Relacional- Feito

## 2.4 CRONOGRAMA

	<b>Descrição</b>	<b>Prazo</b>
1	Introdução	13/08/2012
2	Análise de Sistema	04/09/2012
3	Histórico de Linguagem de Programação de Tecnologias Utilizadas	11/09/2012
4	Desenvolvimento do Sistema	25/09/2012
5	Estudo da Viabilidade ( Cronograma final)	09/10/2012
6	Considerações Finais	23/10/2012
7	Manual de Instalação	13/11/2012
8	Manual do Usuário	13/11/2012
9	Entrega	26/11/2012
10	Desenvolvimento de Banco de Dados	18/09/2012
11	Interface Visual da telas	09/10/2012
12	Desenvolvimento estrutural das telas	30/10/2012
13	Conexões com Banco de Dados	06/11/2012
14	Teste do Sistema	13/11/2012

*Tabela 5: Cronograma*

## 2.5 RELATÓRIO DE CUSTOS

<b>Recursos</b>	<b>Valor R\$</b>
Energia Elétrica	R\$ 144,00
Computadores	R\$ 3.000,00
Funcionários	R\$ 41.500,00
<b>Total</b>	<b>R\$ 44.644,00</b>

*Tabela 6: Relatório de Custos*

## 2.6 HORAS DE PROJETO

<b>Recurso</b>	<b>Horas Trabalhadas</b>	<b>Valor Total</b>
Lucas Carvalho	300 hrs	R\$ 10.000,00
Márcia Cristina	350 hrs	R\$ 9.500,00
Taina Araújo	450 hrs	R\$ 12.000,00
Thayná Emanuely	450 hrs	R\$ 10.000,00
<b>Total</b>	<b>1550 hrs</b>	<b>R\$ 41.500,00</b>

*Tabela 7: Horas do Projeto*

## 2.7 CUSTOS DO PROJETO

Software	Valor da Licença	
Visual Basic 2010	R\$	20.835,00
SQL Server	R\$	-
<b>Total</b>	<b>R\$</b>	<b>20.835,00</b>

*Tabela 8: Custos do projeto*

## 3. HISTÓRICO DA LINGUAGEM DE PROGRAMAÇÃO E TECNOLOGIAS

### 3.1 VISUAL BASIC

A linguagem Basic, criada por J.Kemeny e T. Kurtz em 1963 foi elaborada com o intuito de deixar claro o ensino dos conceitos da programação. As linguagens FORTRAN e Assembly empregavam rotinas de baixo nível para alguns controles e compilação, a Basic vinha então facilitar ao programador sua tarefa, não precisando se preocupar com métodos e algoritmos exigidos para construir e limpar os programas.

Inicialmente, cada linha começava com um número de linha e as instruções não eram endentadas (estrutura semelhante a sub-níveis). Todos os caracteres utilizados eram em caixa alta e as instruções GOSUB e GOTO utilizavam o número de linha para desviar a sua execução. Estas primeiras versões foram consideradas não-profissionais, mas sua evolução de linguagem interpretada para uma rápida e bem estruturada linguagem compilada, mudou sua reputação.

Na década de 70, um novo Basic interpretado e residente em ROM foi introduzido pela Microsoft. Esta versão, GW-BASIC, podia ser encontrada no sistema operacional MS-DOS 4.01 e anteriores.

Em 1982, a Microsoft revoluciona a linguagem com seu Quick Basic. Os números de linha foram eliminados, foram adicionados novos recursos, subprogramas e dados estruturados agora eram definidos pelo usuário, novas capacidades gráficas e de som, e

velocidade aumentada. A maior vantagem consistia em que seus programas podiam ser executados nos modos interativos e interpretados, ou serem compilados em executáveis

### 3.2 SQL

Tudo começou em março de 1987 quando a Microsoft comprou os direitos do DataServer da Sysbase para o sistema operacional OS/2. O objetivo era provocar o interesse e chamar a atenção da comunidade do dBase. Por isso a Microsoft traçou um acordo com a Ashton-Tate como forma de endossar esse novo processo. Assim, a primeira versão do SQL Server se chamava Ashton-Tate/Microsoft SQL Server e chegou ao mercado na metade final de 1988.

Em maio do ano seguinte a versão 1.0 para OS/2 era lançada com parca participação da Microsoft (limitando-se apenas a poucas ferramentas, testes e o projeto visando tornar o aplicativo mais simples de ser instalado). Na segunda metade de 1990 a união da Microsoft com a Ashton-Tate se encerrou e a versão 1.1 do SQL Server passou a oferecer suporte para o Windows 3.0 (que era uma novidade na época). Apesar disso, a base do SQL era produzida pela Sysbase e a Microsoft sequer tinha acesso ao código-fonte. Qualquer defeito tinha que ser relatado para a Sysbase e corrigido apenas por ela. Só em 1991 a Microsoft passou a poder acessar o código fonte do SQL apenas para leitura.

Com o lançamento da versão 4.2 para OS/2 no fim de 1991 a Microsoft já desenvolvia o banco de dados em conjunto com a Sysbase, o que só mudou em agosto de 1993 quando a Microsoft finalmente conseguiu a totalidade da criação do SQL Server e lançou a versão para o Windows NT 3.1.

### 3.3 DIAGRAMA DE CLASSES

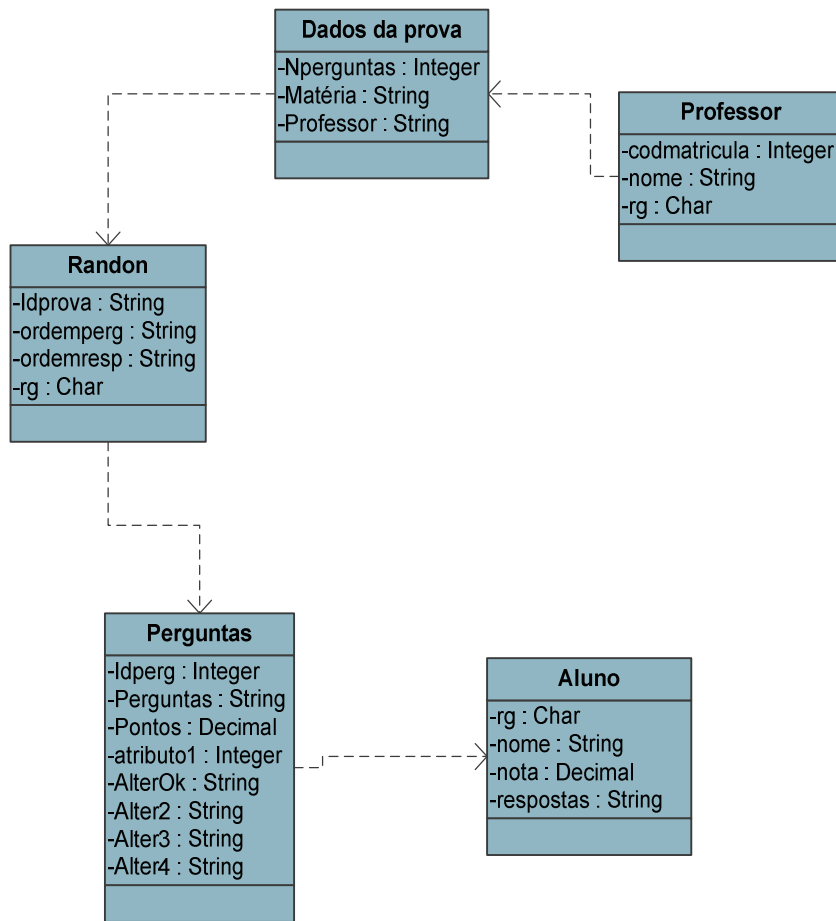


Figura 2: Diagrama de Classes- Feito em 25 de setembro de 2012- Visio.

### 3.4 DIAGRAMA DE CASOS DE USO

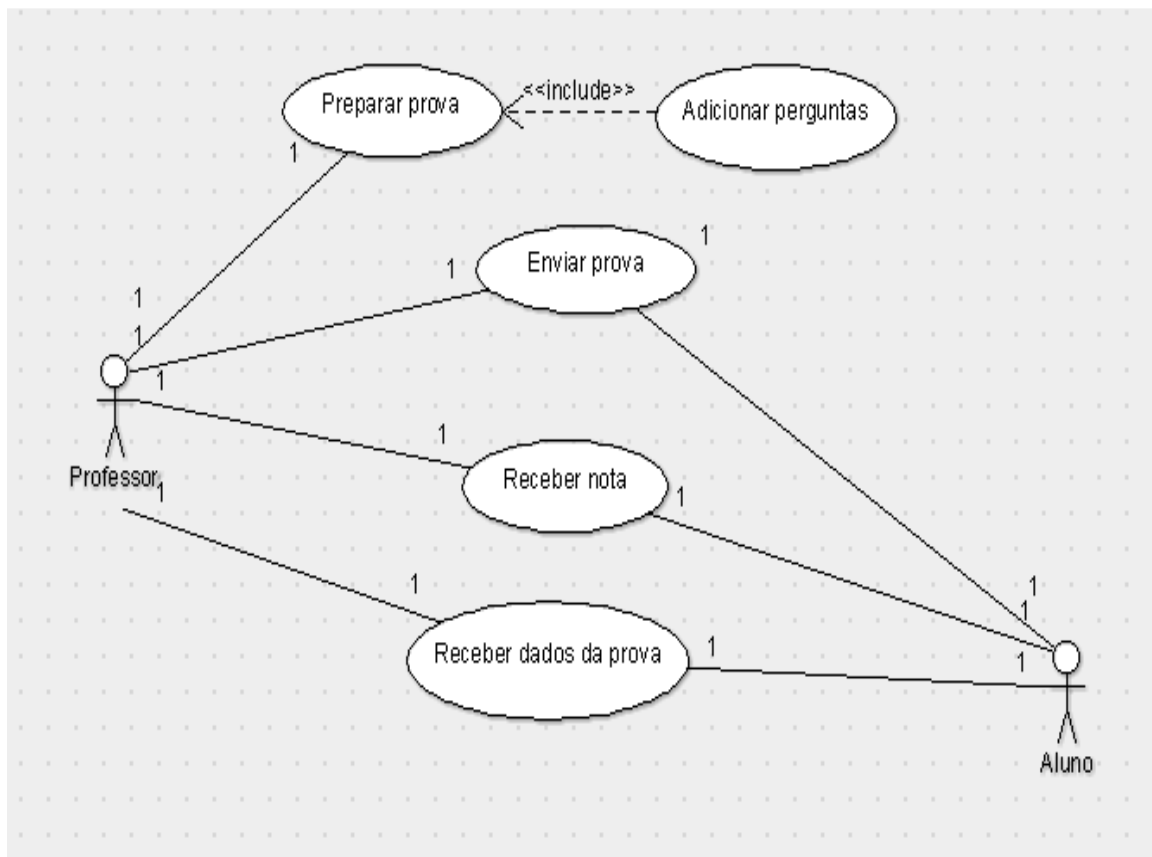


Figura3: Diagrama de Casos de Uso- Feito em 24 de setembro de 2012- ArgoUml.

## 4. ESTUDO DA VIABILIDADE

### 4.1 VIABILIDADE

O investimento no sistema como um todo (visual e programação) visa mostrar em curto prazo inúmeros benefícios não só para os professores, mas também para os alunos.

Por serem tecnologias inovadoras de programação de softwares, como Visual Studio 2010 e, SQL Server 2008, sendo para melhoria das aplicações escolares a implantação desse sistema se torna mais viável.

“Colas”, rasuras, trocar de provas durante a aplicação e possíveis furtos das provas originais que poderiam alterar o resultado das provas, será reduzido a índices bem menores, pois no nosso software as avaliações serão feitas digitalmente (alterando a respectiva ordem das questões e alternativas), permitindo mais confiança e tranquilidade para os professores nas aplicações dos testes, e para os alunos mais empenho nos estudos para um melhor desempenho nas avaliações.

## 4.2 CRONOGRAMA FINAL

<b>Id</b>	<b>Nome da tarefa</b>	<b>Início</b>	<b>Término</b>
1	Resumo	Qua 01/08/12	Seg 06/08/12
2	Introdução	Qua 01/08/12	Seg 06/08/12
3	Análise de Sistema	Qui 30/08/12	Ter 04/09/12
4	Histórico de linguagem	Qua 05/09/12	Seg 10/09/12
5	Diagrama de Casos de Uso	Sex 21/09/12	Ter 25/09/12
6	Diagrama de Classes	Sex 21/09/12	Ter 25/09/12
7	Cronograma Final	Seg 08/10/12	Seg 08/10/12
8	Estudo da Viabilidade		Seg 08/10/12
9	Início Programação	Ter 28/08/12	Ter 23/10/12
10	Considerações Finais	Sex 19/10/12	Ter 23/10/12
11	Interface Visual	Ter 24/07/12	Ter 02/10/12
12	Manual do Usuário	Qui 08/11/12	Ter 13/11/12
13	Banco de Dados	Ter 14/08/12	Ter 11/09/12
14	Testes do Sistema	Ter 13/11/12	Ter 13/11/12
15	Finalizando formatação/ Impressão		
16	Entrega		
17	Apresentação Final	Qua 05/12/12	Qua 05/12/12

Tabela 9: Cronograma Final

## 4.3 MANUAL DO USUÁRIO

### 4.3.1 Professor

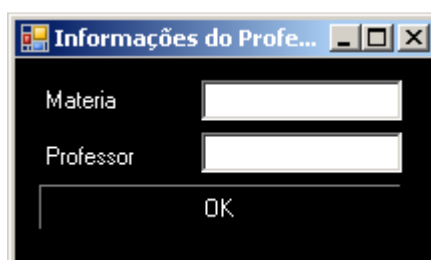


Figura 4. Tela Principal professor

Tela Principal: Campos “Matéria” e “Professor” para acessar o software.

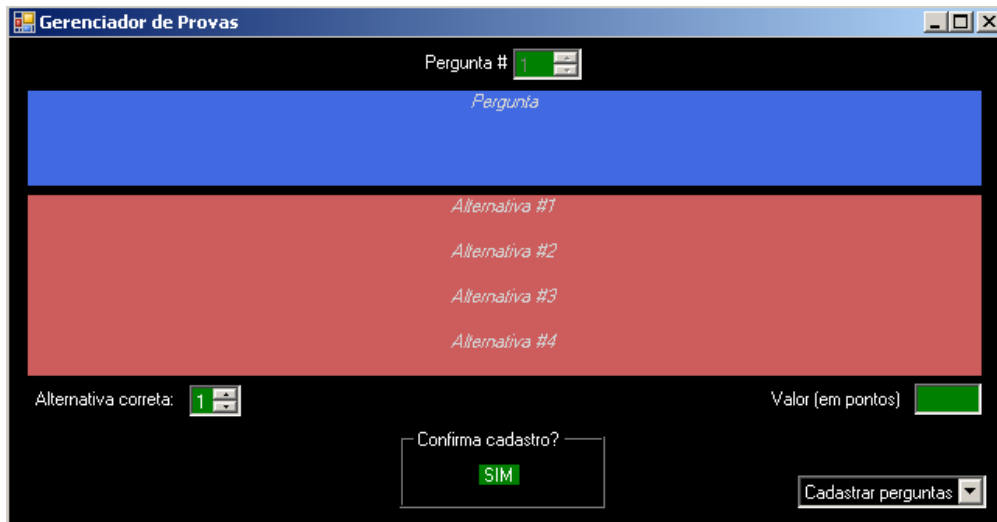


Figura 5. Cadastrar perguntas

Cadastrar: Cadastra a quantidade de questões, suas respectivas respostas, o valor de cada pergunta, e confirma cadastro.

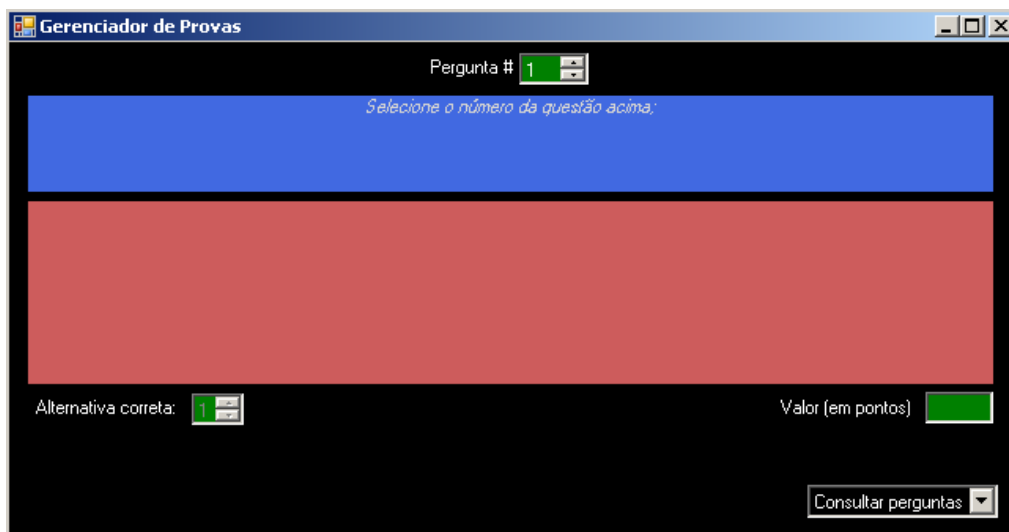


Figura 6. Consultar pergunta



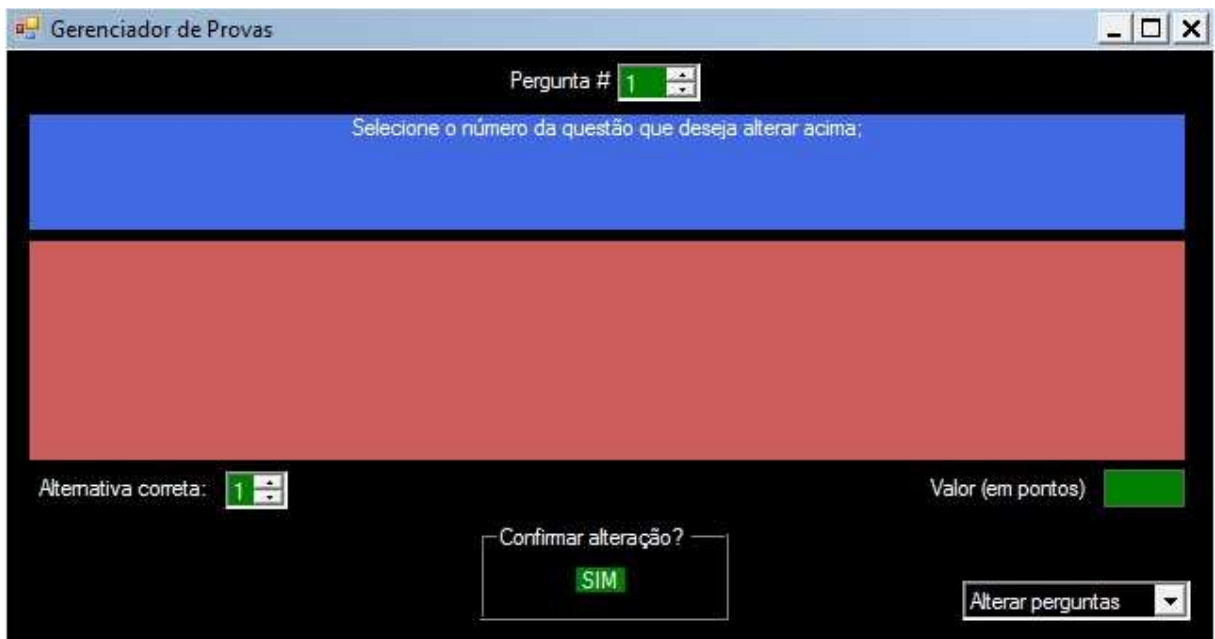


Figura 7. Alterar pergunta

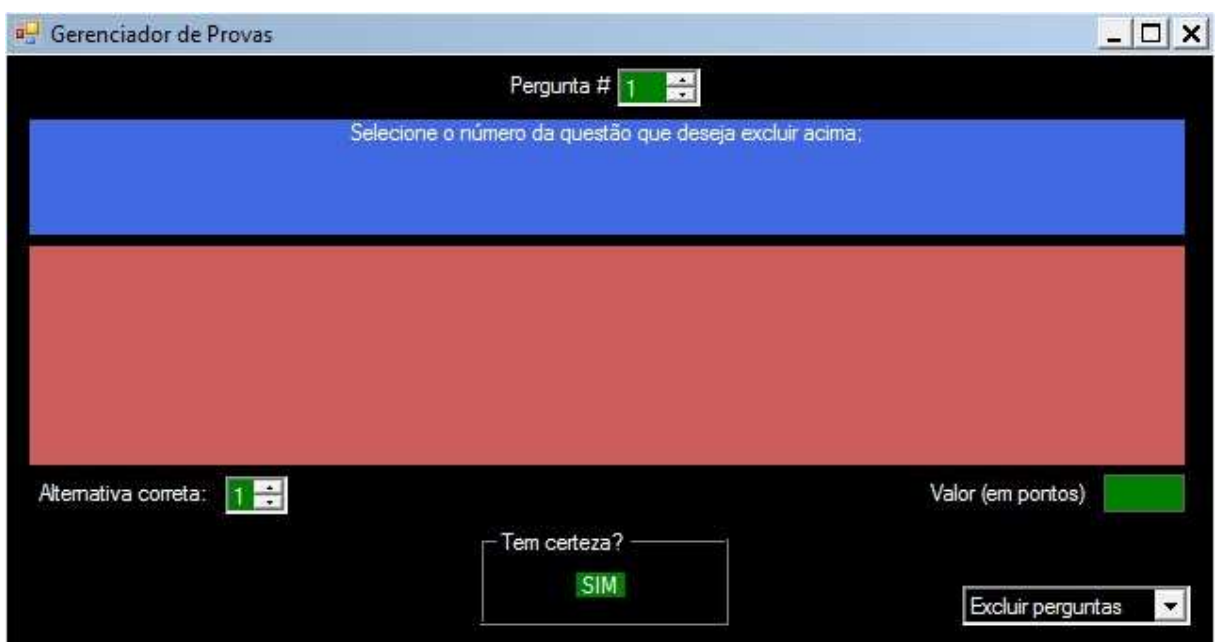


Figura 8. Excluir perguntas

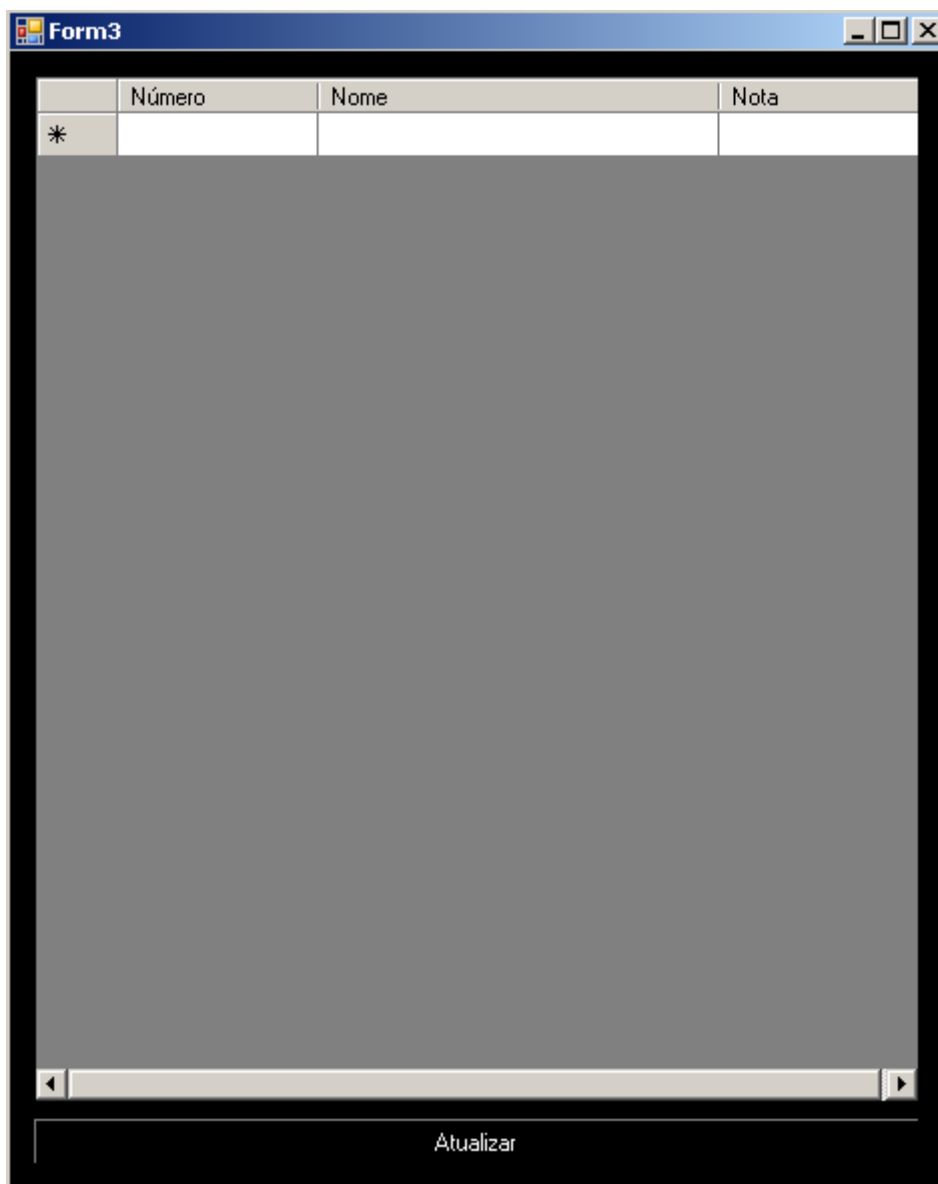


Figura 9. Notas

Visualização de Notas: Exibe as notas dos alunos que terminarem a prova.

#### 4.3.2 Aluno

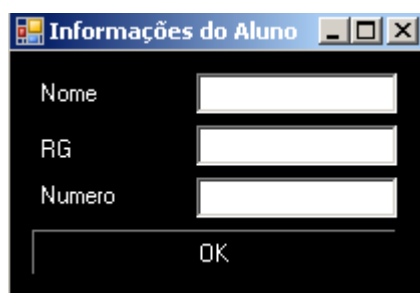


Figura 10. Tela principal aluno

Tela Principal: Campos “Nome” e “RG” para acessar o software.

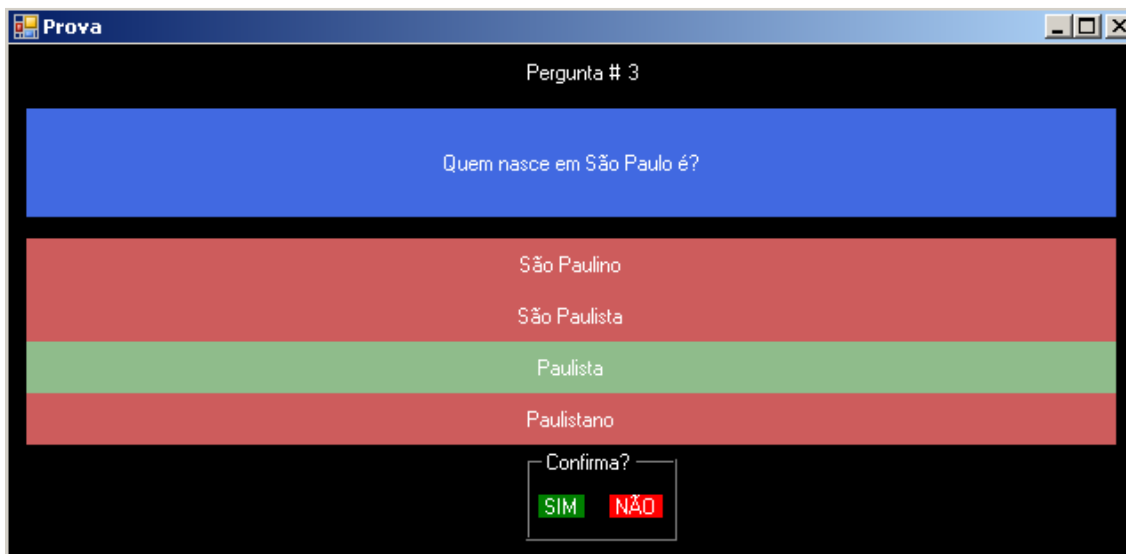


Figura 11. Tela prova

Tela Prova: O aluno seleciona a alternativa e confirma.

## 5. CONSIDERAÇÕES FINAIS E REFERÊNCIAS BIBLIOGRÁFICAS

### 5.1 CONSIDERAÇÕES FINAIS

Demonstramos a viabilidade da implantação de um sistema eletrônico para aplicação de provas, que visa evitar os problemas encontrados com “colas”, dando maior confiabilidade ao processo, pois o programa irá proporcionar notas mais justas e uma correção mais rápida, evitando assim a perda das provas durante a sua correção. Sendo assim esperamos uma boa aceitação do projeto no mercado, e que ele realmente auxilie na realização das provas: agilizando as correções e influenciando o empenho cada vez maior dos alunos.

## 5.2 REFERÊNCIAS BIBLIOGRÁFICAS

**Microsoft Visual Studio.** Acesso em 11 set. de 2012, disponível em Wikipédia, A Enciclopédia Livre:

[http://pt.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](http://pt.wikipedia.org/wiki/Microsoft_Visual_Studio)

**Microsoft Visual Studio 2010 Professional.** (04 abr. de 2010). Acesso em 11 set. de 2012, disponível em Baixaki:

<http://www.baixaki.com.br/download/visual-studio-2010-professional.htm>

**Microsoft Visual Studio.** Acesso em 17 ser. De 2012, disponível em

Cotia net: <http://www.cotianet.com.br/bit/hist/basic.htm>

**Microsoft SQL Server.** Acesso em 11 set. de 2012, disponível em Elder Stroparo:

<http://elderstroparo.blogspot.com.br/2010/01/historia-do-sql-server.html>

**Microsoft SQL Server.** Acesso em 11 set. de 2012. Disponível em Oficina da net:

[http://www.oficinadanet.com.br/area/13/sql\\_server](http://www.oficinadanet.com.br/area/13/sql_server)

**Toda a história do sistema SQL da Microsoft.** Disponível em Artigonal:

<http://www.artigonal.com/advertising-artigos/toda-a-historia-do-sistema-sql-da-microsoft-1845126.html>. Acesso em 18 de setembro de 2012.

## APÊNDICE

### Implementação do Código da Programação – Shuffle

#### PROGRAMA DO PROFESSOR

Tela de carregamento

```
PublicClassLoad
PrivateSub Load_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
    System.IO.File.Copy(Application.StartupPath & "\data.accdb",
Application.StartupPath & "\RandomDB.accdb", True)
Form2.Show()
Me.Hide()
EndSub
EndClass
```

Tela de Login

```
PublicClassForm2
Dim Constring AsString = "provider=Microsoft.ACE.OLEDB.12.0;Password='';User
ID=Admin;Data Source = " + Application.StartupPath & "\RandomDB.accdb"
Dim DBCon AsNew OleDb.OleDbConnection(Constring)
Dim dr As OleDb.OleDbDataReader
Public prof, materia AsString
PrivateSub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
If TextBox1.Text <> "" And TextBox2.Text <> "" Then
    materia = TextBox1.Text
    prof = TextBox2.Text
Dim strsql AsString = "INSERT INTO dadosprova (materia, professor, x) VALUES
(@materia, @professor, @x)"
Dim cm AsNew OleDb.OleDbCommand(strsql, DBCon)
cm.Parameters.AddWithValue("@materia", materia)
cm.Parameters.AddWithValue("@professor", prof)
cm.Parameters.AddWithValue("@x", 1)
Try
    DBCon.Open()
    cm.ExecuteNonQuery()
    DBCon.Close()
Form1.Show()
Me.Hide()
Catch ex AsException
MessageBox.Show(ex.Message)
DBCon.Close()
EndTry
```

```

Else
MessageBox.Show("Não deixe campos em branco.", "Atenção", MessageBoxButtons.OK,
MessageBoxIcon.Error)
EndIf
EndSub
EndClass

```

Tela de Funções

```

PublicClassForm1
Dim Constring AsString = "provider=Microsoft.ACE.OLEDB.12.0;Password="";User
ID=Admin;Data Source = " + Application.StartupPath &"\RandomDB.accdb"
Dim DBCon AsNew OleDb.OleDbConnection(Constring)
Dim dr As OleDb.OleDbDataReader
Dim disable AsBoolean = False
Dim modo AsInteger = 1
Dim x AsInteger = 1
Dim y AsInteger = 1
Dim check AsBoolean = False
Dim materia AsString = Form2.materia
PrivateSub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
ComboBox1.Text = "Cadastrar perguntas"
resetlabels()
modo = 1
GroupBox2.Text = "Confirma cadastro?"
NumericUpDown2.ReadOnly = True
GroupBox2.Visible = True
proxperg()
EndSub
PrivateSub ComboBox1_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ComboBox1.SelectedIndexChanged
If ComboBox1.Text = "Cadastrar perguntas"Then
resetlabels()
modo = 1
TextBox1.Clear()
GroupBox2.Text = "Confirma cadastro?"
disable = False
NumericUpDown2.ReadOnly = True
GroupBox2.Visible = True
proxperg()
ElseIf ComboBox1.Text = "Consultar perguntas"Then
resetlabels()
modo = 2
TextBox1.Text = "Selecione o número da questão acima;"
GroupBox2.Text = ""
disable = True
NumericUpDown2.ReadOnly = False
GroupBox2.Visible = False
ElseIf ComboBox1.Text = "Excluir perguntas"Then
resetlabels()
modo = 3
TextBox1.Text = "Selecione o número da questão que deseja excluir acima;"
GroupBox2.Text = "Tem certeza?"

```

```

        disable = True
        NumericUpDown2.ReadOnly = False
        GroupBox2.Visible = True
    ElseIf ComboBox1.Text = "Alterar perguntas" Then
        resetlabels()
        modo = 4
        TextBox1.Text = "Selecione o número da questão que deseja alterar acima;"
    GroupBox2.Text = "Confirmar alteração?"
        disable = False
        NumericUpDown2.ReadOnly = False
        GroupBox2.Visible = True
    ElseIf ComboBox1.Text = "Finalizar prova" Then
        check = True
        proxperg()
    If MessageBox.Show("Deseja finalizar a prova? atualmente há "& x - 1 &" questões cadastradas.",
    "Atenção", MessageBoxButtons.YesNo, MessageBoxIcon.Question) Then
    Dim strSQL As String = "UPDATE dadosprova SET Npergunta=@Npergunta WHERE materia=" &
    materia & ""
    Dim cm As New OleDb.OleDbCommand(strsql, DBCon)
        cm.Parameters.AddWithValue("@Npergunta", x - 1)
    Try
        DBCon.Open()
        cm.ExecuteNonQuery()
        DBCon.Close()
        System.IO.Directory.CreateDirectory("C:\data")
        System.IO.File.Copy(Application.StartupPath & "\RandomDB.accdb",
"C:\data\data.accdb", True)
    MessageBox.Show("Prova finalizada.", "Completo")
    Form3.Show()
    Me.Hide()
    Catch ex As Exception
    MessageBox.Show(ex.Message)
        DBCon.Close()
    EndTry
    EndIf
    EndIf
    EndSub
    PublicSub resetlabels()
        NumericUpDown1.Value = 1
        NumericUpDown2.Value = 1
        TextBox1.Clear()
        TextBox2.Clear()
        TextBox3.Clear()
        TextBox4.Clear()
        TextBox5.Clear()
        TextBox6.Clear()
        x = 1
        y = 1
        check = False
    EndSub
    PublicSub proxperg()
        x = 1
    Dim flag = True
    While flag = True

```

```

Dim strsql AsString = "SELECT * FROM perguntas WHERE IDperg=" & x
Dim cm AsNew OleDb.OleDbCommand(strsql, DBCon)
Try
    DBCon.Open()
    dr = cm.ExecuteReader
If dr.HasRows Then
While dr.Read
If (dr.Item("IDperg") = x) Then
If x = 100 Then
MessageBox.Show("Limite de perguntas atingido")
Exit Sub
EndIf
                x = x + 1
EndIf
EndWhile
Else
    flag = False
EndIf
    DBCon.Close()
Catch ex AsException
MessageBox.Show(ex.Message)
    DBCon.Close()
EndTry
EndWhile
IfNot check Then
    NumericUpDown2.Value = x
EndIf
EndSub
PrivateSub TextBox_GotFocus(ByVal sender AsObject, ByVal e As System.EventArgs) Handles
TextBox1.GotFocus, TextBox2.GotFocus, TextBox3.GotFocus, TextBox4.GotFocus,
TextBox5.GotFocus
If disable = TrueThen
    Label1.Focus()
EndIf
EndSub
PrivateSub NumericUpDown1_GotFocus(ByVal sender AsObject, ByVal e As System.EventArgs)
Handles NumericUpDown1.GotFocus
If disable = TrueThen
    Label1.Focus()
EndIf
EndSub
PrivateSub NumericUpDown1_ValueChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles NumericUpDown1.ValueChanged
If disable = TrueThen
    NumericUpDown1.Value = y
EndIf
EndSub
PrivateSub NumericUpDown2_GotFocus(ByVal sender AsObject, ByVal e As System.EventArgs)
Handles NumericUpDown2.GotFocus
If modo = 1 Then
    Label1.Focus()
EndIf
EndSub

```



```

PrivateSub NumericUpDown2_ValueChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles NumericUpDown2.ValueChanged
If modo = 1 Then
    NumericUpDown2.Value = x
Exit Sub
EndIf
Dim strsql AsString = "SELECT * FROM perguntas WHERE IDperg=" &
NumericUpDown2.Value
Dim cm AsNew OleDb.OleDbCommand(strsql, DBCon)
Try
    DBCon.Open()
    dr = cm.ExecuteReader
If dr.HasRows Then
While dr.Read
If (dr.Item("IDperg") = NumericUpDown2.Value) Then
TextBox1.Text = dr.Item("pergunta")
    TextBox6.Text = dr.Item("pontos")
NumericUpDown1.Value = dr.Item("alterOK")
    y = NumericUpDown1.Value
If disable = TrueThen
        disable = False
        NumericUpDown1.Value = dr.Item("alterOK")
        y = NumericUpDown1.Value
        disable = True
EndIf
        TextBox2.Text = dr.Item("alter1")
        TextBox3.Text = dr.Item("alter2")
        TextBox4.Text = dr.Item("alter3")
        TextBox5.Text = dr.Item("alter4")
EndIf
EndWhile
Else
    TextBox1.Text = ""
    TextBox6.Text = ""
    NumericUpDown1.Value = 1
    y = 1
    TextBox2.Text = ""
    TextBox3.Text = ""
    TextBox4.Text = ""
    TextBox5.Text = ""
EndIf
    DBCon.Close()
Catch ex AsException
    MessageBox.Show(ex.Message)
    DBCon.Close()
EndTry
EndSub
PrivateSub Label8_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Label8.Click
SelectCase modo
Case 1
If TextBox6.Text = ""Then
    MessageBox.Show("Atribua um valor para a pergunta.", "Atenção")
    TextBox6.Focus()

```

```

Exit Sub
ElseIfNot IsNumeric(TextBox6.Text) Then
    MessageBox.Show("O valor da pergunta deve ser numérico.", "Atenção")
    TextBox6.Focus()
Exit Sub
ElseIf TextBox1.Text = ""Then
    MessageBox.Show("O campo de pergunta não deve estar em branco.", "Atenção")
    TextBox1.Focus()
Exit Sub
Else
    Dim strSQL AsString = "INSERT INTO perguntas (IDperg, pergunta, pontos, alterOK, alter1, alter2,
alter3, alter4) VALUES (@IDperg, @pergunta, @pontos, @alterOK, @alter1, @alter2, @alter3,
@alter4)"
    Dim cm AsNew OleDb.OleDbCommand(strSQL, DBCon)
        cm.Parameters.AddWithValue("@IDperg", NumericUpDown2.Text)
        cm.Parameters.AddWithValue("@pergunta", TextBox1.Text)
        cm.Parameters.AddWithValue("@pontos", TextBox6.Text)
        cm.Parameters.AddWithValue("@alterOK", NumericUpDown1.Text)
        cm.Parameters.AddWithValue("@alter1", TextBox2.Text)
        cm.Parameters.AddWithValue("@alter2", TextBox3.Text)
        cm.Parameters.AddWithValue("@alter3", TextBox4.Text)
        cm.Parameters.AddWithValue("@alter4", TextBox5.Text)

    Try
        DBCon.Open()
        cm.ExecuteNonQuery()
        DBCon.Close()
    MessageBox.Show("Pergunta #" & NumericUpDown2.Text & " adicionada", "Informação",
    MessageBoxButtons.OK, MessageBoxIcon.Information)
    resetlabels()
    Catch ex AsException
    MessageBox.Show(ex.Message)
        DBCon.Close()

    EndTry
        proxperg()

    EndIf
    Case 3
    check = True
        proxperg()
    IfMessageBox.Show("Deseja mesmo excluir a pergunta? A pergunta "& x - 1 &" tomará o lugar
desta.", "Atenção", MessageBoxButtons.YesNo, MessageBoxIcon.Warning) Then
    Dim strSQL AsString = "DELETE FROM perguntas WHERE IDperg=" & NumericUpDown2.Value
    Dim cm AsNew OleDb.OleDbCommand(strSQL, DBCon)
    Try
        DBCon.Open()
        cm.ExecuteNonQuery()
        DBCon.Close()
    Catch ex AsException
    MessageBox.Show(ex.Message)
        DBCon.Close()

    EndTry
    EndIf
    Dim strSQL2 AsString = "UPDATE perguntas SET IDperg=@IDperg WHERE IDperg=" & x - 1
    Dim cm2 AsNew OleDb.OleDbCommand(strSQL2, DBCon)
    Try

```

```

        DBCon.Open()
        cm2.Parameters.AddWithValue("@IDperg", NumericUpDown2.Value)
cm2.ExecuteNonQuery()
        DBCon.Close()
MessageBox.Show("Pergunta excluída.", "ATENÇÃO", MessageBoxButtons.OK,
MessageBoxIcon.Information)
resetlabels()
Catch ex As Exception
MessageBox.Show(ex.Message)
        DBCon.Close()
EndTry
Case 4
If MessageBox.Show("Confirmar alterações?", "Atenção", MessageBoxButtons.YesNo,
MessageBoxIcon.Warning) Then
Dim strSQL As String = "UPDATE perguntas SET pergunta=@pergunta, pontos=@pontos,
alterOK=@alterOK, alter1=@alter1, alter2=@alter2, alter3=@alter3, alter4=@alter4 WHERE
IDperg =" & NumericUpDown2.Value
Dim cm As New OleDb.OleDbCommand(strsql, DBCon)
        cm.Parameters.AddWithValue("@pergunta", TextBox1.Text)
        cm.Parameters.AddWithValue("@pontos", TextBox6.Text)
        cm.Parameters.AddWithValue("@alterOK", NumericUpDown1.Text)
        cm.Parameters.AddWithValue("@alter1", TextBox2.Text)
        cm.Parameters.AddWithValue("@alter2", TextBox3.Text)
        cm.Parameters.AddWithValue("@alter3", TextBox4.Text)
        cm.Parameters.AddWithValue("@alter4", TextBox5.Text)

Try
        DBCon.Open()
        cm.ExecuteNonQuery()
        DBCon.Close()
MessageBox.Show("Dados alterados.", "ATENÇÃO", MessageBoxButtons.OK,
MessageBoxIcon.Information)
resetlabels()
Catch ex As Exception
MessageBox.Show(ex.Message)
        DBCon.Close()
EndTry
EndIf
EndSelect
EndSub
EndClass

```

Tela de Notas

```

Public Class Form3
Dim Constring As String = "provider=Microsoft.ACE.OLEDB.12.0;Password='';User
ID=Admin;Data Source = C:\data\data.acddb"
Dim DBCon As New OleDb.OleDbConnection(Constring)
Dim dr As OleDb.OleDbDataReader
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Me.DataGridView1.Rows.Clear()
Dim strSQL As String = "SELECT * FROM aluno"

```

```

Dim cm AsNew OleDb.OleDbCommand(strsql, DBCon)
Try
    DBCon.Open()
    dr = cm.ExecuteReader
If dr.HasRows Then
While dr.Read
Me.DataGridView1.Rows.Add(dr.Item("numero"), dr.Item("nome"), dr.Item("nota"))
EndWhile
EndIf
    DBCon.Close()
Catch ex AsException
MessageBox.Show(ex.Message)
    DBCon.Close()
EndTry
EndSub
EndClass

```

## PROGRAMA DO ALUNO

### Tela de Login

```

PublicClassForm2
Dim Constring AsString = "provider=Microsoft.ACE.OLEDB.12.0;Password="";User
ID=Admin;Data Source = C:\data\data.accdb"
Dim DBCon AsNew OleDb.OleDbConnection(Constring)
Dim dr As OleDb.OleDbDataReader
Public nome, rg, numero AsString
Public x AsInteger
PublicSub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
If TextBox1.Text <>""And TextBox2.Text <>""And TextBox2.Text.Length = 9 And
IsNumeric(TextBox3.Text) Then
    nome = TextBox1.Text
    rg = TextBox2.Text
    numero = TextBox3.Text
Dim strsql AsString = "SELECT * FROM dadosprova WHERE x = 1"
Dim cm AsNew OleDb.OleDbCommand(strsql, DBCon)
Try
    DBCon.Open()
    dr = cm.ExecuteReader
If dr.HasRows Then
While dr.Read
        x = dr.Item("Npergunta")
EndWhile
EndIf
    DBCon.Close()
Catch ex AsException
MessageBox.Show(ex.Message)
    DBCon.Close()

```

```

EndTry
Dim strsql2 = "INSERT INTO aluno (rg, nome, numero) VALUES (@rg, @nome,
@numero)"
Dim cm2 AsNew OleDb.OleDbCommand(strsql2, DBCon)
    cm2.Parameters.AddWithValue("@rg", rg)
cm2.Parameters.AddWithValue("@nome", nome)
    cm2.Parameters.AddWithValue("@numero", numero)
Try
    DBCon.Open()
    cm2.ExecuteNonQuery()
    DBCon.Close()
Form1.Show()
Me.Hide()
Catch ex AsException
    MessageBox.Show(ex.Message)
    DBCon.Close()
EndTry
Elseif TextBox1.Text = "" Or TextBox2.Text = "" Then
    MessageBox.Show("Não deixe campos em branco.", "Atenção", MessageBoxButtons.OK,
    MessageBoxIcon.Error)
Elseif TextBox2.Text.Length <> 9 Then
    MessageBox.Show("Digite seu RG sem pontos ou traços (Ex: 123456789)", "Atenção",
    MessageBoxButtons.OK, MessageBoxIcon.Error)
Else
    MessageBox.Show("Digite seu número.", "Atenção", MessageBoxButtons.OK,
    MessageBoxIcon.Error)
EndIf
EndSub
PrivateSub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
Dim strsql AsString = "SELECT * FROM dadosprova WHERE x = 1"
Dim cm AsNew OleDb.OleDbCommand(strsql, DBCon)
Try
    DBCon.Open()
    dr = cm.ExecuteReader
If dr.HasRows Then
While dr.Read
    x = dr.Item("Npergunta")
EndWhile
EndIf
    DBCon.Close()
Form1.Show()
Me.Hide()
Catch ex AsException
    MessageBox.Show(ex.Message)
    DBCon.Close()
EndTry
EndSub
EndClass

```

Tela da Prova

PublicClassForm1

```
Dim Constring AsString = "provider=Microsoft.ACE.OLEDB.12.0;Password="";User  
ID=Admin;Data Source = C:\data\data.accdb"
```

```
Dim DBCon AsNew OleDb.OleDbConnection(Constring)
```

```
Dim dr As OleDb.OleDbDataReader
```

```
Dim random, random2, resp, jafoi2(4), nota, y AsInteger
```

```
Public x AsInteger = Form2.x
```

```
Public rg AsString = Form2.rg
```

```
Dim jafoi(x - 1) AsInteger
```

```
Dim pergunta(x - 1, 6) AsString
```

```
PublicSub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
```

```
Handles MyBase.Load
```

```
    jafoi2(0) = 1
```

```
For count = 0 To x - 1
```

```
Dim strsql AsString = "SELECT * FROM perguntas WHERE IDperg=" & count + 1
```

```
Dim cm AsNew OleDb.OleDbCommand(strsql, DBCon)
```

```
Try
```

```
    DBCon.Open()
```

```
    dr = cm.ExecuteReader
```

```
If dr.HasRows Then
```

```
While dr.Read
```

```
If (dr.Item("IDperg") = count + 1) Then
```

```
pergunta(count, 0) = dr.Item("pergunta")
```

```
    pergunta(count, 1) = dr.Item("alter1")
```

```
    pergunta(count, 2) = dr.Item("alter2")
```

```
    pergunta(count, 3) = dr.Item("alter3")
```

```
    pergunta(count, 4) = dr.Item("alter4")
```

```
    pergunta(count, 5) = dr.Item("alterOK")
```

```
    pergunta(count, 6) = dr.Item("pontos")
```

```
EndIf
```

```
EndWhile
```

```
EndIf
```

```
    DBCon.Close()
```

```
Catch ex AsException
```

```
    MessageBox.Show(ex.Message)
```

```
    DBCon.Close()
```

```
EndTry
```

```
Next
```

```
    Randomize()
```

```
    random = Int(Rnd() * x)
```

```
    Label3.Text = pergunta(random, 0)
```

```
While jafoi2(random2) = 1
```

```
    Randomize()
```

```
    random2 = Int(Rnd() * 4 + 1)
```

```
EndWhile
```

```
Label4.Text = pergunta(random, random2)
```

```
    jafoi2(random2) = 1
```

```
If pergunta(random, 5) = random2 Then
```

```

        pergunta(random, 5) = 1
    EndIf
    random2 = 0
    While jafoi2(random2) = 1
        Randomize()
        random2 = Int(Rnd() * 4 + 1)
    EndWhile
    Label5.Text = pergunta(random, random2)
    jafoi2(random2) = 1
    If pergunta(random, 5) = random2 Then
        pergunta(random, 5) = 2
    EndIf
    random2 = 0
    While jafoi2(random2) = 1
        Randomize()
        random2 = Int(Rnd() * 4 + 1)
    EndWhile
    Label6.Text = pergunta(random, random2)
    jafoi2(random2) = 1
    If pergunta(random, 5) = random2 Then
        pergunta(random, 5) = 3
    EndIf
    random2 = 0
    While jafoi2(random2) = 1
        Randomize()
        random2 = Int(Rnd() * 4 + 1)
    EndWhile
    Label7.Text = pergunta(random, random2)
    jafoi2(random2) = 1
    If pergunta(random, 5) = random2 Then
        pergunta(random, 5) = 4
    EndIf
EndSub
PublicSub redall()
    Label4.BackColor = Color.IndianRed
    Label5.BackColor = Color.IndianRed
    Label6.BackColor = Color.IndianRed
    Label7.BackColor = Color.IndianRed
EndSub
PrivateSub Label4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Label4.Click
    If random < x Then
        redall()
        Label4.BackColor = Color.DarkSeaGreen
        resp = 1
        GroupBox2.Visible = True
    EndIf
EndSub
PrivateSub Label5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Label5.Click

```

```

If random < x Then
    redall()
    Label5.BackColor = Color.DarkSeaGreen
    resp = 2
    GroupBox2.Visible = True
EndIf
EndSub
PrivateSub Label6_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Label6.Click
If random < x Then
    redall()
    Label6.BackColor = Color.DarkSeaGreen
    resp = 3
    GroupBox2.Visible = True
EndIf
EndSub
PrivateSub Label7_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Label7.Click
If random < x Then
    redall()
    Label7.BackColor = Color.DarkSeaGreen
    resp = 4
    GroupBox2.Visible = True
EndIf
EndSub
PrivateSub Label8_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Label8.Click
If random < x Then
If resp = pergunta(random, 5) Then
    nota = nota + pergunta(random, 6)
EndIf
    Label2.Text = Label2.Text + 1
    jafoi(random) = 1
    y = y + 1
EndIf
If y < x Then
While jafoi(random) = 1
    Randomize()
    random = Int(Rnd() * x)
EndWhile
    random2 = 0
jafoi2(1) = 0
    jafoi2(2) = 0
    jafoi2(3) = 0
    jafoi2(4) = 0
    Label3.Text = pergunta(random, 0)
While jafoi2(random2) = 1
    Randomize()
    random2 = Int(Rnd() * 4 + 1)
EndWhile

```



```

Label4.Text = pergunta(random, random2)
    jafoi2(random2) = 1
If pergunta(random, 5) = random2 Then
    pergunta(random, 5) = 1
EndIf
    random2 = 0
While jafoi2(random2) = 1
    Randomize()
    random2 = Int(Rnd() * 4 + 1)
EndWhile
Label5.Text = pergunta(random, random2)
    jafoi2(random2) = 1
If pergunta(random, 5) = random2 Then
    pergunta(random, 5) = 2
EndIf
    random2 = 0
While jafoi2(random2) = 1
    Randomize()
    random2 = Int(Rnd() * 4 + 1)
EndWhile
Label6.Text = pergunta(random, random2)
    jafoi2(random2) = 1
If pergunta(random, 5) = random2 Then
    pergunta(random, 5) = 3
EndIf
    random2 = 0
While jafoi2(random2) = 1
    Randomize()
    random2 = Int(Rnd() * 4 + 1)
EndWhile
Label7.Text = pergunta(random, random2)
    jafoi2(random2) = 1
If pergunta(random, 5) = random2 Then
    pergunta(random, 5) = 4
EndIf
Else
    Label2.Text = ""
    Label3.Text = "Sua nota: " & nota
    Label4.Text = ""
    Label5.Text = ""
    Label6.Text = ""
    Label7.Text = ""
    random = x + 1
Dim strSql AsString = "UPDATE aluno SET nota=@nota WHERE rg=" & rg & """"
Dim cm AsNew OleDb.OleDbCommand(strsql, DBCon)
Try
DBCon.Open()
    cm.Parameters.AddWithValue("@nota", nota)
cm.ExecuteNonQuery()
    DBCon.Close()

```

```

Catch ex As Exception
    MessageBox.Show(ex.Message)
        DBCon.Close()
EndTry
EndIf
EndSub
PrivateSub Label9_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Label8.Click, Label9.Click
    Label4.BackColor = Color.IndianRed
    Label5.BackColor = Color.IndianRed
    Label6.BackColor = Color.IndianRed
    Label7.BackColor = Color.IndianRed
    GroupBox2.Visible = False
EndSub
PrivateSub Label3_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Label3.TextChanged, Label4.TextChanged, Label5.TextChanged,
Label6.TextChanged, Label7.TextChanged
    ToolTip1.SetToolTip(Label3, Label3.Text)
    ToolTip1.SetToolTip(Label4, Label4.Text)
    ToolTip1.SetToolTip(Label5, Label5.Text)
    ToolTip1.SetToolTip(Label6, Label6.Text)
    ToolTip1.SetToolTip(Label7, Label7.Text)
EndSub
EndClass

```