

UNIVERSIDADE DO VALE DO ITAJAI
ESCOLA DO MAR, CIÊNCIA E TECNOLOGIA
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Disciplina: Engenharia de Requisitos

Professor: Alex Luciano Roesler Rese, MSc.

Alunos: Tânia Michel Pereira e Evandro Orlandini

FÓRUM TEMÁTICO

INTRODUÇÃO

Este documento tem por objetivo apresentar à disciplina de Hands On Work III os requisitos para construção de um software para um clube.

O clube deseja ter um controle efetivo de sócios, dependentes, aulas, pagamentos, e desfrutar de um software com tecnologia open-source.

Conforme documento de entrevista com o clube, apresentamos o processo de cadastro de novos sócios através do fluxograma BPMN (AS IS), três regras de negócios, dez requisitos funcionais, três requisitos não funcionais, diagramas de casos de uso, e documentação de quatro dos casos de uso do projeto do clube em questão.

FÓRUM TEMÁTICO

INTRODUÇÃO

Este documento tem por objetivo apresentar à disciplina de Hands On Work III os requisitos para construção de um software para um clube.

O clube deseja ter um controle efetivo de sócios, dependentes, aulas, pagamentos, e desfrutar de um software com tecnologia open-source.

Conforme documento de entrevista com o clube, apresentamos o processo de cadastro de novos sócios através do fluxograma BPMN (AS IS), três regras de negócios, dez requisitos funcionais, três requisitos não funcionais, diagramas de casos de uso, e documentação de quatro dos casos de uso do projeto do clube em questão.

REGRAS DE NEGÓCIOS

RN001	
Nome	Limite de dependentes
Módulo	Cadastro de Sócios
Data de Criação	28/08/2021
Autor	João Alexandre do Amaral Chaves
Versão	1.0
Dependência	RF001
Descrição	Cada sócio só pode ter no máximo 5 dependentes
Tipo	Restrição

RN002	
Nome	Limite de alunos por aula de natação
Módulo	Agendamento de aulas
Data de Criação	28/08/2021
Autor	João Alexandre do Amaral Chaves
Versão	1.0
Dependência	RF002
Descrição	Cada aula de natação só pode ter no máximo 8 alunos
Tipo	Restrição

RN003	
Nome	Limite de alunos por aula de tênis
Módulo	Agendamento de aulas
Data de Criação	28/08/2021
Autor	João Alexandre do Amaral Chaves
Versão	1.0
Dependência	RF002
Descrição	Cada aula de tênis só pode ter no máximo 2 alunos
Tipo	Restrição

REQUISITOS FUNCIONAIS

[RF001] - Cadastro de Sócios	
Ator	Funcionário recepção
Prioridade	Essencial
Entradas e pré-condições	Dados Pessoais dos Sócios
Saídas e pós-condições	Cadastro do sócio efetivado
Fluxo de eventos principal	Entrada na tela de cadastro de sócios <ol style="list-style-type: none"> 1. Digitação do nome 2. Digitação do CPF 3. Digitação do RG 4. Digitação da data de nascimento 5. Digitação do endereço 6. Digitação do telefone 7. Digitação do e-mail 8. Finalização do cadastro

[RF002] - Agendamento de Aulas	
Ator	Funcionário agendamento

Prioridade	Essencial
Entradas e pré-condições	As aulas, dias, horários, professores e quantidade de vagas disponíveis por aula devem estar cadastrados no sistema.
Saídas e pós-condições	Confirmar aula, gerar redução da quantidade de vagas disponíveis nas aulas, nos dias e horários.
Fluxo de eventos principal	Entrada na tela de agendamento de aulas <ol style="list-style-type: none"> 1. Digitação do nome da aula 2. Digitação dos dias 3. Digitação do horário 4. Digitação do professor 5. Finalização do agendamento

[RF003] - Cadastro de Dependentes	
Ator	Funcionário recepção
Prioridade	Essencial
Entradas e pré-condições	Dados do dependente que será cadastrado. O sócio deve estar cadastrado para poder cadastrar dependentes e com o limite de cadastrar apenas 5 dependentes.
Saídas e pós-condições	Cadastro do sócio dependente efetivado
Fluxo de eventos principal	Entrada na tela de cadastro de sócios dependentes <ol style="list-style-type: none"> 1. Digitação do nome 2. Digitação do CPF 3. Digitação do RG 4. Digitação da data de nascimento 5. Digitação do endereço 6. Digitação do telefone 7. Digitação do e-mail 8. Finalização do cadastro

[RF004] - Emissão Boleto	
Ator	Funcionário financeiro
Prioridade	Essencial
Entradas e pré-condições	Pagamento do boleto da matrícula,
Saídas e pós-condições	Geração dos boletos das mensalidades, acessos as aulas
Fluxo de eventos principal	Entrada na tela de cadastro financeiro dos sócios <ol style="list-style-type: none"> 1. Digitação do nome 2. Digitação do CPF 3. Digitação do RG 4. Digitação do telefone 5. Digitação do e-mail 1. Digitação da data de pagamento 6. Finalização do cadastro

[RF005] - Cadastro de Aula de Tênis	
Ator	Funcionário administrativo
Prioridade	Essencial
Entradas e pré-condições	Selecionar dias, horários e professores
Saídas e pós-condições	Gerar a possibilidade de agendar aulas
Fluxo de eventos principal	Entrada na tela de cadastro de aulas <ol style="list-style-type: none"> 1. Digitação da aula 2. Digitação do dia 3. Digitação do horário 4. Digitação do professor 5. Digitação da quantidade de vagas por aula 6. Finalização do cadastro

[RF006] - Cadastro de Aula de Natação	
Ator	Funcionário administrativo
Prioridade	Essencial
Entradas e pré-condições	Selecionar dias, horários e professores
Saídas e pós-condições	Gerar a possibilidade de agendar aulas
Fluxo de eventos principal	Entrada na tela de cadastro de aulas <ol style="list-style-type: none"> 1. Digitação da aula 2. Digitação do dia 3. Digitação do horário 4. Digitação do professor 5. Digitação da quantidade de vagas por aula 6. Finalização do cadastro

[RF007] - Cadastro de Quadro de Horários - Agendas	
Ator	Funcionário administrativo
Prioridade	Essencial
Entradas e pré-condições	Selecionar horários e Informar horários e limites
Saídas e pós-condições	Gerar o cadastro das aulas, dias, horários, e quantidade de vagas disponíveis por aula, o controle do fluxo de agendamentos e as possibilidades de agendamentos.
Fluxo de eventos principal	Entrada na tela de cadastro de quadro de horários <ol style="list-style-type: none"> 1. Digitação do nome de aulas 2. Digitação do dia 3. Digitação do horário 4. Digitação do limite de alunos 5. Digitação do nome dos professores de cada aula 6. Confirmar cadastro

[RF008] - Cadastro de funcionários	
Ator	Funcionário administrativo
Prioridade	Essencial
Entradas e pré-condições	Dados pessoais dos funcionários

Saídas e pós-condições	Controle de acessos dos funcionários no sistema .Gerar a possibilidade dos funcionários fazerem login com o seu e-mail e cadastrar uma senha para ter acesso segundo suas funções e responsabilidades
Fluxo de eventos principal	Entrada na tela de cadastro de aulas <ol style="list-style-type: none"> 1. Digitação do nome 2. Digitação do CPF 3. Digitação do RG 4. Digitação da data de nascimento 5. Digitação do endereço 6. Digitação do telefone 7. Digitação do e-mail 8. Digitação da área de atuação 9. Finalização do cadastro

[RF009] - Cadastro de professores	
Ator	Funcionário administrativo
Prioridade	Essencial
Entradas e pré-condições	Dados pessoais dos professores
Saídas e pós-condições	Gerar a possibilidade dos professor fazerem login com o seu e-mail e cadastrar uma senha para ter acesso, gerar a possibilidade de agendar aulas.
Fluxo de eventos principal	Entrada na tela de cadastro de professores <ol style="list-style-type: none"> 1. Digitação do nome 2. Digitação do CPF 3. Digitação do RG 4. Digitação da data de nascimento 5. Digitação do endereço 6. Digitação do telefone 7. Digitação do e-mail 8. Digitação da área de atuação 9. Finalização do cadastro

[RF010] - Cadastro de acesso	
Ator	Funcionário administrativo
Prioridade	Essencial
Entradas e pré-condições	Dados dos sócios
Saídas e pós-condições	Pagamentos em dia, acesso ao clube e aulas
Fluxo de eventos principal	Entrada na tela de cadastro de acesso <ol style="list-style-type: none"> 1. Digitação do nome do sócio 2. Digitação do número de matrícula 3. Cadastrar diagnóstico financeiro 4. Confirmar cadastro

REQUISITOS NÃO FUNCIONAIS

Identificador:	RNF-001	Categoria:	Financeiro
Nome:	Forma de pagamento		

Data de Criação:	08/09/2021	Autor:	Evandro Orlandini
Data da última alteração:	08/09/2021		
Versão:	1	Prioridade:	Essencial
Descrição:	A forma de pagamento deve ser em boleto		

Identificador:	RNF-002	Categoria:	Tecnologia
Nome:	Tipo de tecnologia		
Data de Criação:	08/09/2021	Autor:	Evandro Orlandini
Data da última alteração:	08/09/2021		
Versão:	1	Prioridade:	Essencial
Descrição:	A tecnologia deve ser open-source.		

Identificador:	RNF-003	Categoria:	Horários
Nome:	Forma de apresentação de horários		
Data de Criação:	08/09/2021	Autor:	Evandro Orlandini
Data da última alteração:	08/09/2021		
Versão:	1	Prioridade:	Essencial
Descrição:	Os horários devem ser exibidos no formato de grade como uma agenda		

DIAGRAMA DE CASO DE USO

Figura 2. Diagrama Caso de Uso

DOCUMENTAÇÃO DOS CASOS DE USO

Identificador	UC001
Nome	Cadastro de sócios
Descrição	Cadastrar o sócio principal
Ator principal	Funcionário da recepção/Funcionário do financeiro
Atores secundários	Sócio requerente
Pré-condição	1. Cadastrar funcionário da recepção no sistema

	2. Funcionário da recepção estar logado no sistema
Fluxo principal	<ol style="list-style-type: none"> 1. Recepção solicita documentação do sócio requerente e dos dependentes (nome, RG, CPF, data de nascimento, endereço) e dos dados (e-mail, telefone) 2. Sócio requerente fornece a documentação e dados solicitados 3. Recepção cadastra no sistema os dados da documentação do novo sócio e de até 5 dependentes 4. Recepção informa o valor da matrícula, das mensalidades e datas de pagamento 5. Sócio requerente aceita o pagamento do valor da matrícula, das mensalidades e datas de pagamento 6. Sócio requerente escolhe uma das datas de pagamento das mensalidades disponíveis 7. Recepção cadastra a data de pagamento 8. Recepção no sistema solicita ao financeiro a liberação da emissão e envio dos boletos da matrícula e das mensalidades do novo sócio. 9. Financeiro libera a emissão dos boleto da matrícula e o envia no e-mail do novo sócio e avisa no sistema o funcionário da recepção 10. Recepção recebe a aviso que o boleto da matrícula do novo sócio foi enviados ao e-mail do novo sócio. 11. Recepção informa o novo sócio que lhe foi enviado por e-mail o boleto da matrícula e que após o pagamento a matrícula será efetivada e os boletos das mensalidades lhe serem enviados
Fluxo alternativo	<p>Sessão expirada</p> <ol style="list-style-type: none"> 1. Mensagem na tela requerendo: refazer login e senha <p>Sócio requerente não aceita pagar o valor da matrícula e/ou das mensalidades e/ou datas</p> <ol style="list-style-type: none"> 1. Finaliza o processo de matrícula como cadastro não

	efetivado
Pós-condição	Gerar os boletos das mensalidades, pagar os boletos das mensalidades.

Identificador	UC002
Nome	Emissão Boleto
Descrição	Liberação da emissão e envio dos boletos das mensalidades
Ator principal	Funcionário do financeiro/ Funcionário da recepção
Atores secundários	Sócio requerente
Pré-condição	Cadastro do funcionário de recepção; cadastro do sócio, verificar se o número de dependentes está abaixo de 5
Fluxo principal	<ol style="list-style-type: none"> 1. Financeiro abre e vê no sistema a solicitação do funcionário da recepção, de emissão e envio do boleto de matrícula do sócio requerente 2. Financeiro libera a emissão dos boleto da matrícula e o envia no e-mail do sócio requerente e avisa no sistema o funcionario da recepção 3. Financeiro confere no sistema sócio requerente cadastrado se a matrícula foi paga 4. Financeiro ao comprovar a efetivação do pagamento da matrícula, muda o status no sistema de sócio requerente para sócio cadastrado e libera a emissão dos boletos das mensalidades e os envia por e-mail ao novo sócio.
Fluxo alternativo	<p>Sessão expirada</p> <ol style="list-style-type: none"> 1. Mensagem na tela requerendo : refazer login e senha <p>Financeiro confere no sistema que a matrícula não foi paga</p> <ol style="list-style-type: none"> 1. Mandar e-mail ao sócio requerente solicitando que a matrícula seja paga para efetivar a matrícula.
Pós-condição	Emissão dos boletos da matrícula e das mensalidades, pagar

	os boletos da matrícula e das mensalidades
--	--

Identificador	UC003
Nome	Cadastro de professores.
Descrição	Cadastrar os professores de cada aula
Ator principal	Funcionário do administrativo
Atores secundários	Professores
Pré-condição	Cadastro do funcionário do administrativo.
Fluxo principal	<ol style="list-style-type: none"> 1. Administrativo solicita documentação do professor (nome, RG, CPF, data de nascimento, endereço) e dados (e-mail, telefone, área de atuação) 2. Professor fornece a documentação e dados solicitados 3. Administrativo cadastra no sistema os documentos e dados do professor. 4. Administrativo envia e-mail para o professor avisando da efetivação do cadastro e solicita que o professor cadastre na interface do sistema o seu e-mail no login e uma senha para acesso 5. Professor cadastra na interface do sistema o seu e-mail no login e uma senha. 6. Sistema emite mensagem: login e senhas cadastrados com sucesso.
Fluxo alternativo	<p>E-mail incorreto</p> <ol style="list-style-type: none"> 1. Mensagem na tela informando: e-mail incorreto
Pós-condição	Gerar disponibilidade de agendamento das aulas .

Identificador	UC004
Nome	Agendamento de aulas
Descrição	Agendar aulas
Ator principal	Funcionário do agendamento/Funcionário administrativo
Atores secundários	Sócio principal, sócios dependentes
Pré-condição	1. Cadastrar funcionário da agendamento;

	<ol style="list-style-type: none"> 2. Funcionário do agendamento estar logado; 3. Cadastrar o sócio principal e seus dependentes; 4. Pagamento das mensalidades dos sócios em dia; 5. Administrativo cadastrar aulas, turmas, professores, dias e horários disponíveis;
Fluxo principal	<ol style="list-style-type: none"> 1. Agendamento confere no sistema as aulas, turmas, professores, dias e horários disponíveis 2. Agendamento avisa o sócio sobre as aulas, turmas, professores, dias e horários disponíveis para agendar 3. O sócio escolhe a aula, turma, professor, dias e horários para fazer as aulas 4. Agendamento agenda a aula, turma, professor, dias e horários escolhido pelo sócio no sistema 5. Agendamento avisa o sócio da confirmação do agendamento. 6. Agendamento envia e-mail para o sócio avisando da efetivação do agendamento da aula, turma, professor, dias e horários escolhido.
Fluxo alternativo	N/A
Pós-condição	Gerar disponibilidade do sócio participar das aulas

Segue o código do projeto:

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using MySql.Data.MySqlClient;

namespace clube
{
    public partial class Form1 : Form

```

```

{
    public Form1()
    {
        InitializeComponent();
    }

    private MySqlConnectionStringBuilder conexaoBanco()
    {
        MySqlConnectionStringBuilder conexaoBD = new
        MySqlConnectionStringBuilder();
        conexaoBD.Server = "localhost";
        conexaoBD.Server.Database = "clube";
        conexaoBD.UserID = "root";
        conexaoBD.Password = "";
        return conexaoBD;
    }

    private void atualizarGrid()
    {
        MySqlConnectionStringBuilder conexaoBD = conexaoBanco();
        MySqlConnection realizaConexaoBD = new
        MySqlConnection(conexaoBD.ToString());
        try
        {
            realizaConexaoBD.Open();

            MySqlCommand comandoMySQL = realizaConexaoBD.CreateCommand();
            comandoMySQL.CommandText = "SELECT * FROM membro";
            MySqlDataReader reader = comandoMySQL.ExecuteReader();

            dgClube.Rows.Clear();
            //ou membro

            while (reader.Read());
            {
                DataGridViewRow row = (DataGridViewRow)dgClube.Rows[0].Clone();

                row.Cells[0].value = reader.GetInt32(0); //ID
                row.Cells[1].value = reader.GetString(1); //NOME
                row.Cells[2].value = reader.GetString(2); //TELEFONE
                row.Cells[3].value = reader.GetString(3); //EMAIL
                row.Cells[4].value = reader.GetString(4); // AULA
                row.Cells[5].value = reader.GetString(5); //MENSALIDADE
                row.Cells[6].value = reader.GetString(6); //ATIVO
                dgClube.Rows.Add(row);
            }
            realizaConexaoBD.Close();
        }

        catch (Exception ex)
        {
            MessageBox.Show("Can not open connection! ");
            Console.WriteLine(ex.Message);
        }
    }

    private void Form1_Load(object sender, EventArgs e)
    {
        atualizarGrid();
    }
}

```

