

# Especificação de Software



## Módulo 1 Manutenção de Ocupação

<b>Versão</b>	<b>Autor</b>	<b>Data de Lançamento</b>
1.0.0	Leandro Alves	20/04/2023

## Conteúdo

1. Index	1
1.1. Scripts SQL	1
1.2. Interfaces	3
2. Inserir	7
2.1. Scripts SQL	7
2.2. Interfaces	10
3. Alterar	12
3.1. Scripts SQL	12
3.2. Interfaces	15

# 1. Index

## 1.1. Scripts SQL

select-usuario-logado.sql

```
USE db_Faculdade;

--Parâmetros de entrada
SET @Cod_Usuario_Logado = 1;

--Execução
SELECT I_Cod_Usuario, C_Perfil FROM Usuario
WHERE I_Cod_Usuario = @Cod_Usuario_Logado;
```

select-sem-filtro.sql

```
USE db_Faculdade;

--Parâmetros de entrada
SET @Ordenacao_Por = '';
SET @E_Ordenacao_Crescente = 1;
SET @Page = 1;

--Execução
SET @Results_Skipped = (@Page - 1) * 10;
SET @Desc_Text = IF(@E_Ordenacao_Crescente = 1, '', ' DESC');
PREPARE STMT FROM
    CONCAT(CONCAT('
SELECT o.I_Cod_Ocupacao, o.I_Cod_Usuario_Professor,
      o.I_Cod_Curso, o.S_Dia_Semana, o.D_Hora_Inicio,
      o.D_Hora_Fim, p.S_Nome as ProfesorS_Nome,
      c.S_Nome as CursoS_Nome
FROM Ocupacao o
      INNER JOIN Usuario p
          ON p.I_Cod_Usuario = o.I_Cod_Usuario_Professor
      INNER JOIN Curso c
          ON c.I_Cod_Curso = o.I_Cod_Curso
ORDER BY (CASE @Ordenacao_Por
  WHEN '' THEN 1
  WHEN ''ProfesorS_Nome'' THEN ProfesorS_Nome
  WHEN ''CursoS_Nome'' THEN CursoS_Nome
  WHEN ''S_Dia_Semana'' THEN S_Dia_Semana
  WHEN ''D_Hora_Inicio'' THEN D_Hora_Inicio
  WHEN ''D_Hora_Fim'' THEN D_Hora_Fim
END
)', @Desc_Text), ' LIMIT 10 OFFSET ?');
```

```
EXECUTE STMT USING @Results_Skipped;
```

delete.sql

```
USE db_Faculdade;

--Parâmetros de Entrada
SET @Cod_Ocupacao = 2;

--Execução
DELETE FROM Ocupacao
WHERE I_Cod_Ocupacao = @Cod_Ocupacao;
```

## 1.2. Interfaces

### Index de Ocupações no Computador:

**Ocupações**

Q Termos

> Filtros Avançados

Ordenar por: Nada

Pesquisar + Inserir...

Nome Professor	Nome Curso	Dia da Semana	Hora Início	Hora Fim	#
Lynnderson	React	Domingo	14:00	15:30	⋮
Lynnderson	React	Sábado	16:00	18:00	⋮
Mariana	C#	Domingo	16:00	17:30	⋮

< 1 >

### Evento de carregar a página

Passo 1) Rodar a `select-usuario-logado.sql`, com o `@Cod_Usuario_Logado` do usuário logado. Se a coluna perfil do resultado for diferente de 'S', não mostrar botão de inserir e nem a coluna # dos resultados. No caso de tela mobile, não mostrar o Dropdown.

Passo 2) Rodar a `select-sem-filtro.sql`, com `@Page = 1`

### Maqui\_Filtro\_Termos


### Maqui\_Filtro\_Avancado\_Wrapper

Conteúdo:

Maqui\_Filtro\_Avancado\_Texto "Nome Professor" X (ProfessorS\_Nome)

Maqui\_Filtro\_Avancado\_Texo "Nome Curso" X (CursoS\_Nome)

Maqui\_Filtro\_Avancado\_Literal "Dia da Semana" (Literal\_Modelo: Dia Semana) X (S\_Dia\_Semana)



Leandro
 

Segurança
 Planejamento
 Matrículas e Notas

### Ocupações

Filtros Avançados

Nome Professor
 Nome Curso
 Dia da Semana
 

Hora Início

 Hora Fim


Hora Início

Ordenar por
 Nada

Pesquisar
 + Inserir...

Nome Professor	Nome Curso	Dia da Semana	Hora Início	Hora Fim	#
Lynnderson	React	Domingo	14:00	15:30	⋮
Lynnderson	React	Sábado	16:00	18:00	⋮
Mariana	C#	Domingo	16:00	17:30	⋮

< 1 >



Leandro
 

Segurança
 Planejamento
 Matrículas e Notas

### Ocupações

Filtros Avançados

Nome Professor
 Nome Curso
 Dia da Semana
 

Hora Início

 Hora Fim

Hora Início

Ordenar por
 Nada

Pesquisar
 + Inserir...

Nome Professor	Nome Curso	Dia da Semana	Hora Início	Hora Fim	#
Lynnderson	React	Domingo	14:00	15:30	⋮
Lynnderson	React	Sábado	16:00	18:00	⋮
Mariana	C#	Domingo	16:00	17:30	⋮

< 1 >

**Filtro Avançado “Hora Início”:** possui o Chip “Hora Início”, que mostra e esconde o campo “Hora Início” que aceita apenas input de horário. Quando informada, a Hora Início exibe somente as ocupações que tem um horário colidindo com o período desde a Hora Início informada até 23:59

**Filtro Avançado “Hora Fim”:** possui o Chip “Hora Início”, que mostra e esconde o campo “Hora Início” que aceita apenas input de horário. Quando informada, a Hora Fim exibe somente as ocupações que tem um horário colidindo com o período desde a 00:00 até a Hora Fim Informada.

Ao informar a Hora Início e Hora Fim, ignorar regras anteriores para filtrar por horário e exibir as ocupações que tem um horário colidindo com o período desde a Hora Início até a Hora Fim.

De volta à interface de Index:

Maqui\_Ordernar\_Por X (Nome Professor, Nome Curso, Dia da Semana, Hora Início, Hora Fim)

Maqui\_Botao\_Lento “Pesquisar”

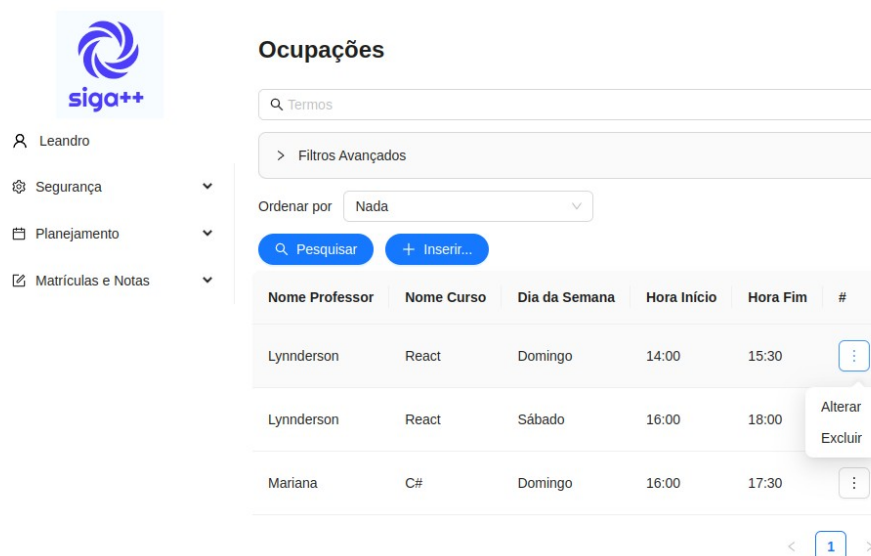
### Evento de Clique em Pesquisar

Passo 1) Colocar a página 1

Passo 2) Carregar os filtros e ordenações no endpoint de pesquisa com filtros, passando a página como 1.

Passo 3) Memorizar os filtros e ordenações que o usuário informou para ser utilizado novamente ao navegar entre as páginas.

**Botão Incluir:** ao ser clicado, redireciona para Inserção de Ocupação.



Nome Professor	Nome Curso	Dia da Semana	Hora Início	Hora Fim	#
Lynnderson	React	Domingo	14:00	15:30	⋮
Lynnderson	React	Sábado	16:00	18:00	Alterar Excluir
Mariana	C#	Domingo	16:00	17:30	⋮

**Coluna # do resultado:** Dropdown com as opções Alterar e Excluir. Alterar deve redirecionar para a tela de Alteração da ocupação, passando o código dessa ocupação, e Excluir deve abrir um Maqui\_Confirmacao\_Exclusao (

Pode: “Deseja excluir a Ocupação? A Ação não pode ser desfeita.”  
)

### Interface da Confirmação para excluir:


#### Evento de clique em Excluir

Passo 1) Rodar a `delete.sql`

Passo 2) Limpar os filtros, ordenações e voltar à página 1

Passo 3) Rodar a `select-sem-filtro.sql` passando `@Page = 1`

De volta à Interface de Index:  
Mostrando resultados no Celular:



Leandro

## Ocupações

> Filtros Avançados

Ordenar por 

Nada

Pesquisar

+ Inserir...

Lynnderson - React

Dia da semana: Domingo

Hora Início: 14:00

Hora Fim: 15:30

Lynnderson - React

Dia da semana: Sábado

Hora Início: 16:00

Hora Fim: 18:00

Mariana - C#

Dia da semana: Domingo

Hora Início: 16:00

Hora Fim: 17:30

< 1 >

Devem ser exibidos dois cards por linha, com os campos rearranjados conforme a captura de tela.

Ao final dos resultados:  
Maqui\_Paginacao



## 2. Inserção

### 2.1. Scripts SQL

select-usuario-logado.sql

```
USE db_Faculdade;

--Parâmetros de entrada
SET @Cod_Usuario_Logado = 1;

--Execução
SELECT I_Cod_Usuario, C_Perfil FROM Usuario
WHERE I_Cod_Usuario = @Cod_Usuario_Logado;
```

select-possivel-conflito-hora.sql

```
USE db_Faculdade;

--Parâmetros de entrada
SET @Cod_Usuario_Professor = 4;
SET @Sigla_Dia_Semana = 'SEG';
SET @Hora_Inicio = '2023-01-01 16:00:00';
SET @Hora_Fim = '2023-01-01 16:30:00';

--Execução
SET @Minutos_Dia_Semana =
    (CASE @Sigla_Dia_Semana
        WHEN 'DOM' THEN 0
        WHEN 'SEG' THEN 1
        WHEN 'TER' THEN 2
        WHEN 'QUA' THEN 3
        WHEN 'QUI' THEN 4
        WHEN 'SEX' THEN 5
        WHEN 'SAB' THEN 6
        END) * 24 * 60;
SET @Minutos_Inicio = @Minutos_Dia_Semana +
    HOUR(@Hora_Inicio) * 60 + MINUTE(@Hora_Inicio);
SET @Minutos_Fim = @Minutos_Dia_Semana +
    HOUR(@Hora_Fim) * 60 + MINUTE(@Hora_Fim);

SELECT I_Cod_Usuario_Professor, S_Dia_Semana, D_Hora_Inicio,
D_Hora_Fim, Minutos_Inicio, Minutos_Fim FROM (
SELECT I_Cod_Usuario_Professor, S_Dia_Semana, D_Hora_Inicio,
D_Hora_Fim,
    ( (CASE S_Dia_Semana
        WHEN 'DOM' THEN 0
        WHEN 'SEG' THEN 1
        WHEN 'TER' THEN 2
```

```

        WHEN 'QUA' THEN 3
        WHEN 'QUI' THEN 4
        WHEN 'SEX' THEN 5
        WHEN 'SAB' THEN 6
        END) * 24 * 60 +
        HOUR(D_Hora_Inicio) * 60 + MINUTE(D_Hora_Inicio)
    ) AS Minutos_Inicio,
    ( (CASE S_Dia_Semana
        WHEN 'DOM' THEN 0
        WHEN 'SEG' THEN 1
        WHEN 'TER' THEN 2
        WHEN 'QUA' THEN 3
        WHEN 'QUI' THEN 4
        WHEN 'SEX' THEN 5
        WHEN 'SAB' THEN 6
        END) * 24 * 60 +
        HOUR(D_Hora_Fim) * 60 + MINUTE(D_Hora_Fim)
    ) AS Minutos_Fim
FROM Ocupacao
WHERE I_Cod_Usuario_Professor = @Cod_Usuario_Professor
) sq
WHERE
    (@Minutos_Inicio BETWEEN sq.Minutos_Inicio AND sq.Minutos_Fim)
    OR (@Minutos_Fim BETWEEN sq.Minutos_Inicio AND sq.Minutos_Fim)
    OR (sq.Minutos_Inicio BETWEEN @Minutos_Inicio AND @Minutos_Fim)
    OR (sq.Minutos_Fim BETWEEN @Minutos_Inicio AND @Minutos_Fim);

```

insert.sql

```

USE db_Faculdade;

--Parâmetros de entrada
SET @I_Cod_Usuario_Professor = 4;
SET @I_Cod_Curso = 2;
SET @Sigla_Dia_Semana = 'SAB';
SET @Hora_Inicio = '2023-01-01 19:00:00';
SET @Hora_Fim = '2023-01-01 20:45:00';

--Execução
INSERT INTO Ocupacao (I_Cod_Usuario_Professor, I_Cod_Curso,
    S_Dia_Semana, D_Hora_Inicio, D_Hora_Fim)
VALUES (@I_Cod_Usuario_Professor, @I_Cod_Curso,
    @Sigla_Dia_Semana, @Hora_Inicio, @Hora_Fim);

```

select-professores.sql

```

USE db_Faculdade;

--Parâmetros de entrada

```

```
SET @Input_Nome = 'Ma';

--Execução
SELECT I_Cod_Usuario, C_Perfil, S_Nome
FROM Usuario
WHERE C_Perfil = 'P'
      AND LOWER(S_Nome) LIKE (
        SELECT GROUP_CONCAT('%', LOWER(@Input_Nome), '%')
      )
LIMIT 20;
```

select-cursos.sql

```
USE db_Faculdade;

--Parâmetros de entrada
SET @Input_Nome = 'Re';

--Execução
SELECT I_Cod_Curso, S_Nome FROM Curso
WHERE LOWER(S_Nome) LIKE (
  SELECT GROUP_CONCAT('%', LOWER(@Input_Nome), '%')
)
LIMIT 20;
```

## 2.2. Interfaces

### Inserção de Ocupação:



**Inserção de Ocupação**

\* Nome Professor  
Selecione...

\* Nome Curso  
Selecione...

\* Dia da Semana  
Selecione...

\* Hora Início

\* Hora Fim

[← Voltar](#) [Confirmar](#)

### Evento de carregar a página

Passo 1) Rodar a `select-usuario-logado.sql`, com o `@Cod_Usuario_Logado` do usuário logado. Se a coluna perfil do resultado for diferente de 'S', mostrar somente a navegação e mais nenhum elemento da Inserção.

**Maqui\_Campo\_FK "Nome Professor" (Coluna\_Label: S\_Nome) X `select-professores.sql`**

**@Input\_Nome:** validador final form de obrigatório.

**Maqui\_Campo\_FK "Nome Curso" (Coluna\_Label: S\_Nome) X `select-cursos.sql`**

**@Input\_Nome:** validador final form de obrigatório.

**Maqui\_Campo\_Literal "Dia da Semana" (Literal\_Modelo: Dia Semana, Com\_Selecione: 1):** validador final form de obrigatório.

**Campo "Hora Início":** aceita apenas campo de horário. Validador final form de obrigatório.

**Campo "Hora Fim":** aceita apenas campo de horário. Validadores final form de obrigatório e deve ser maior que a Hora Início.

**Maqui\_Botao\_Voltar\_Historico**

**Maqui\_Botao\_Lento "Confirmar"**

### Evento de Clique em Confirmar

Passo 1) Se tiver algum erro indicado por validadores final form, não continuar.

Passo 2) Rodar a `select-possivel-conflito-hora.sql`, passando o professor, dia da semana, hora início e hora fim. Se houver um registro, mostrar um Maqui\_Erro\_Estourado "Esse professor já está ocupado em algum momento do horário informado" e não continuar.

Passo 3) Rodar a `insert.sql`, passando: `@Sigla_Dia_Semana` de acordo com qual dia da semana está sendo inserido; `@Hora_Inicio` e `@Hora_Fim` usando a data 01/01/2023 mais o horário de cada respectivo campo; e os demais campos.

Passo 4) Redirecionar para a Index de Ocupações.

## 3. Alterar

### 3.1. Scripts SQL

select-usuario-logado.sql

```
USE db_Faculdade;

--Parâmetros de entrada
SET @Cod_Usuario_Logado = 1;

--Execução
SELECT I_Cod_Usuario, C_Perfil FROM Usuario
WHERE I_Cod_Usuario = @Cod_Usuario_Logado;
```

select-detalhes.sql

```
USE db_Faculdade;

--Parâmetros de entrada
SET @Cod_Ocupacao = 1;

--Execução
SELECT I_Cod_Ocupacao, I_Cod_Usuario_Professor,
       I_Cod_Curso, S_Dia_Semana, D_Hora_Inicio, D_Hora_Fim
FROM Ocupacao
WHERE I_Cod_Ocupacao = @Cod_Ocupacao;
```

select-possivel-conflito-hora-outro.sql

```
USE db_Faculdade;

--Parâmetros de entrada
SET @Cod_Usuario_Professor = 4;
SET @Sigla_Dia_Semana = 'SEG';
SET @Hora_Inicio = '2023-01-01 16:00:00';
SET @Hora_Fim = '2023-01-01 16:30:00';
SET @Cod_Ocupacao_Ignorada = 2;

--Execução
SET @Minutos_Dia_Semana =
  (CASE @Sigla_Dia_Semana
    WHEN 'DOM' THEN 0
    WHEN 'SEG' THEN 1
    WHEN 'TER' THEN 2
    WHEN 'QUA' THEN 3
    WHEN 'QUI' THEN 4
    WHEN 'SEX' THEN 5
```

```

        WHEN 'SAB' THEN 6
        END) * 24 * 60;
SET @Minutos_Inicio = @Minutos_Dia_Semana +
    HOUR(@Hora_Inicio) * 60 + MINUTE(@Hora_Inicio);
SET @Minutos_Fim = @Minutos_Dia_Semana +
    HOUR(@Hora_Fim) * 60 + MINUTE(@Hora_Fim);

SELECT I_Cod_Ocupacao, I_Cod_Usuario_Professor, S_Dia_Semana,
    D_Hora_Inicio, D_Hora_Fim, Minutos_Inicio, Minutos_Fim FROM (
SELECT I_Cod_Ocupacao, I_Cod_Usuario_Professor, S_Dia_Semana,
    D_Hora_Inicio, D_Hora_Fim,
    ( (CASE S_Dia_Semana
        WHEN 'DOM' THEN 0
        WHEN 'SEG' THEN 1
        WHEN 'TER' THEN 2
        WHEN 'QUA' THEN 3
        WHEN 'QUI' THEN 4
        WHEN 'SEX' THEN 5
        WHEN 'SAB' THEN 6
        END) * 24 * 60 +
        HOUR(D_Hora_Inicio) * 60 + MINUTE(D_Hora_Inicio)
    ) AS Minutos_Inicio,
    ( (CASE S_Dia_Semana
        WHEN 'DOM' THEN 0
        WHEN 'SEG' THEN 1
        WHEN 'TER' THEN 2
        WHEN 'QUA' THEN 3
        WHEN 'QUI' THEN 4
        WHEN 'SEX' THEN 5
        WHEN 'SAB' THEN 6
        END) * 24 * 60 +
        HOUR(D_Hora_Fim) * 60 + MINUTE(D_Hora_Fim)
    ) AS Minutos_Fim
FROM Ocupacao
WHERE I_Cod_Usuario_Professor = @Cod_Usuario_Professor
    AND I_Cod_Ocupacao <> @Cod_Ocupacao_Ignorada
) sq
WHERE
    (@Minutos_Inicio BETWEEN sq.Minutos_Inicio AND sq.Minutos_Fim)
    OR (@Minutos_Fim BETWEEN sq.Minutos_Inicio AND sq.Minutos_Fim)
    OR (sq.Minutos_Inicio BETWEEN @Minutos_Inicio AND @Minutos_Fim)
    OR (sq.Minutos_Fim BETWEEN @Minutos_Inicio AND @Minutos_Fim);

```

update.sql

```

USE db_Faculdade;

--Parâmetros de entrada
SET @Cod_Ocupacao = 2;

```

```

SET @I_Cod_Usuario_Professor = 4;
SET @I_Cod_Curso = 2;
SET @Sigla_Dia_Semana = 'SAB';
SET @Hora_Inicio = '2023-01-01 19:00:00';
SET @Hora_Fim = '2023-01-01 20:40:00';

--Execução
UPDATE Ocupacao SET
    I_Cod_Usuario_Professor = @I_Cod_Usuario_Professor,
    I_Cod_Curso = @I_Cod_Curso,
    S_Dia_Semana = @Sigla_Dia_Semana,
    D_Hora_Inicio = @Hora_Inicio,
    D_Hora_Fim = @Hora_Fim
WHERE I_Cod_Ocupacao = @Cod_Ocupacao;

```

#### select-professores.sql

```

USE db_Faculdade;

--Parâmetros de entrada
SET @Input_Nome = 'Ma';

--Execução
SELECT I_Cod_Usuario, C_Perfil, S_Nome
FROM Usuario
WHERE C_Perfil = 'P'
    AND LOWER(S_Nome) LIKE (
        SELECT GROUP_CONCAT('%', LOWER(@Input_Nome), '%')
    )
LIMIT 20;

```

#### select-cursos.sql

```

USE db_Faculdade;

--Parâmetros de entrada
SET @Input_Nome = 'Re';

--Execução
SELECT I_Cod_Curso, S_Nome FROM Curso
WHERE LOWER(S_Nome) LIKE (
    SELECT GROUP_CONCAT('%', LOWER(@Input_Nome), '%')
)
LIMIT 20;

```



## 3.2. Interfaces

### Alteração de Ocupação:



**Alteração de Ocupação**

\* Nome Professor  
Lynnerson

\* Nome Curso  
C#

\* Dia da Semana  
Sábado

\* Hora Início  
21:00

\* Hora Fim  
22:15

[← Voltar](#) [Confirmar](#)

#### Evento de carregar a página

Passo 1) Rodar a `select-usuario-logado.sql`, com o `@Cod_Usuario_Logado` do usuário logado. Se a coluna perfil do resultado for diferente de 'S', mostrar somente a navegação e mais nenhum elemento da Alteração, e não continuar.

Passo 2) Recuperar o id da ocupação a ser alterada (que é o `Cod_Ocupacao`), por url ou por estado da aplicação.

Passo 3) Rodar a `select-detalhes.sql`, com o `@Cod_Ocupacao` sendo o id recuperado no passo 2. Usar o resultado para preencher os campos Nome professor, Nome Curso, Dia da Semana, Hora Início e Hora Fim.

**Maqui\_Campo\_FK "Nome Professor" (Coluna\_Label: S\_Nome) X select-professores.sql**

@Input\_Nome: validador final form de obrigatório.

**Maqui\_Campo\_FK "Nome Curso" (Coluna\_Label: S\_Nome) X select-cursos.sql**

@Input\_Nome: validador final form de obrigatório.

**Maqui\_Campo\_Literal "Dia da Semana" (Literal\_Modelo: Dia Semana):** validador final form de obrigatório.

**Campo "Hora Início":** aceita apenas campo de horário. Validador final form de obrigatório.

**Campo "Hora Fim":** aceita apenas campo de horário. Validadores final form de obrigatório e deve ser maior que a Hora Início.

**Maqui\_Botao\_Voltar\_Historico**

**Maqui\_Botao\_Lento "Confirmar"**

#### Evento de Clique em Confirmar

Passo 1) Se tiver algum erro indicado por validadores final form, não continuar.

Passo 2) Rodar a `select-possivel-conflito-hora-outro.sql`, passando o id da ocupação a ser alterada no `@Cod_Ocupacao_Ignorada`, professor, dia da semana, hora início e hora fim. Se houver um registro, mostrar um `Maqui_Erro_Estourado` “Esse professor já está ocupado em algum momento do horário informado” e não continuar.

Passo 3) Rodar a `update.sql`, passando: `@Sigla_Dia_Semana` de acordo com qual dia da semana está sendo inserido; `@Hora_Inicio` e `@Hora_Fim` usando a data 01/01/2023 mais o horário de cada respectivo campo; e os demais campos.

Passo 4) Redirecionar para a Index de Ocupações.