

**Curso:** Análise e Desenvolvimento de Sistemas

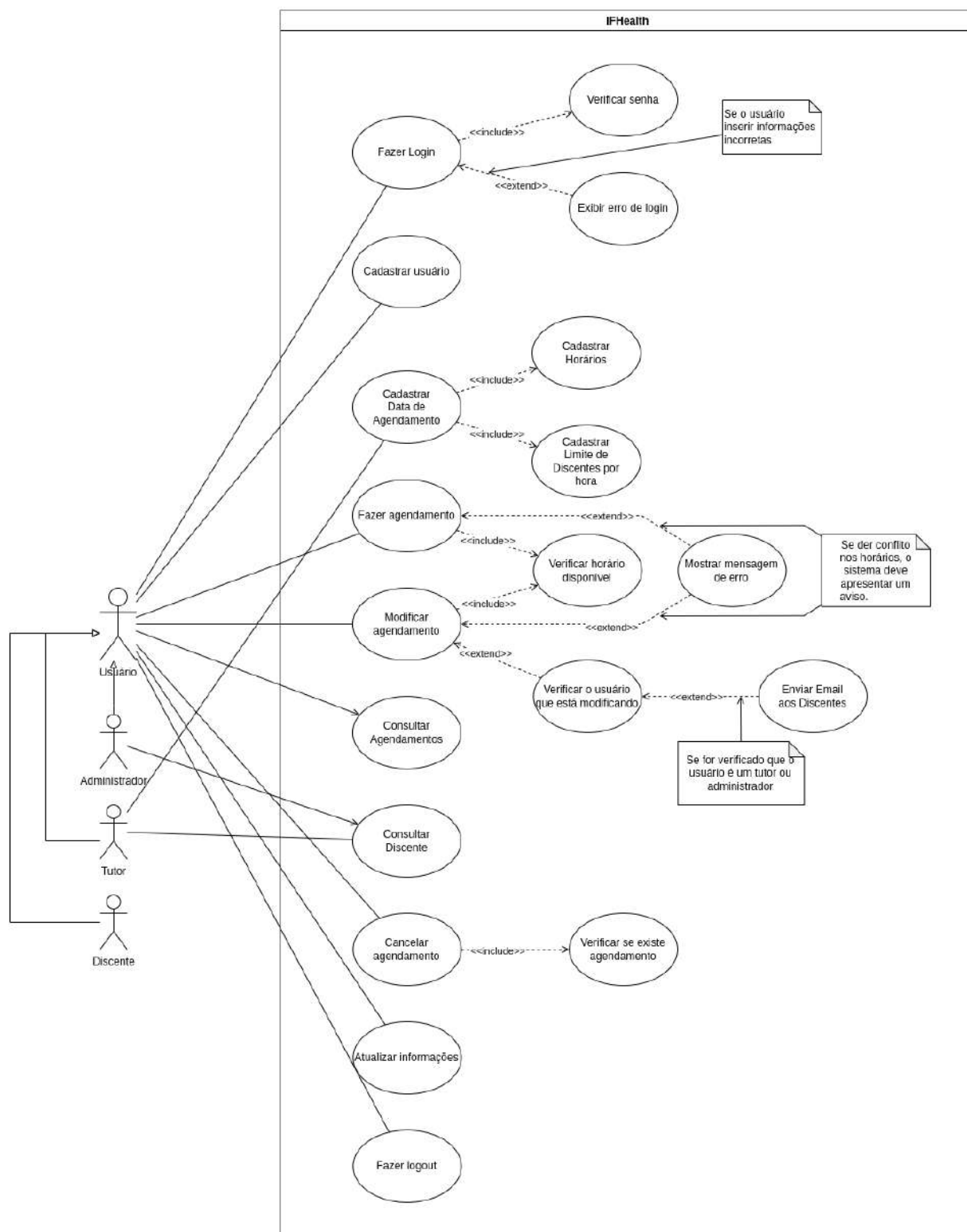
**Disciplina:** Laboratório de Web I      **Data:** 28/04/2022

**Docente:** Fábio dos Santos Lima

**Discentes:** Ana Caroline Silva Nogueira, Cloves de Brito Rodrigues Júnior, Giliarde Reis Pereira, Ícaro Dias dos Santos, Samara da Trindade Torquato

**Turma:** 4º semestre

## **2º Entrega Projeto WEB I**



**Curso:** Análise e Desenvolvimento de Sistemas

**Disciplina:** Laboratório de Web I      **Data:** 28/04/2022

**Docente:** Fábio dos Santos Lima

**Discentes:** Ana Caroline Silva Nogueira, Cloves de Brito Rodrigues Júnior, Giliarde Reis Pereira, Ícaro Dias dos Santos, Samara da Trindade Torquato

**Turma:** 4º semestre

## **2º Entrega Projeto WEB I**

```
CREATE DATABASE IFhealth;
```

```
USE IFhealth;
```

```
CREATE TABLE Discente(  
idDiscente INT PRIMARY KEY,  
nome VARCHAR(45),  
senha VARCHAR(45),  
numMatricula VARCHAR(14),  
peso FLOAT,  
sexo VARCHAR(20),  
idade INT,  
altura FLOAT,  
bloqueado BOOLEAN);
```

```
CREATE TABLE
```

```
CREATE TABLE Agendamento(  
idAgendar INT PRIMARY KEY,  
idtutor INT,  
idDiscente INT,  
idDisponivel INT);
```

```
ALTER TABLE Agendamento ADD CONSTRAINT `fk_Agendarmento_idTutor`  
FOREIGN KEY (idtutor) REFERENCES Tutor(idTutor);
```

```
ALTER TABLE Agendamento ADD CONSTRAINT `fk_Agendarmento_idDisponivel`  
FOREIGN KEY (idDisponivel) REFERENCES Disponibilidade(idDisponivel);
```

```
ALTER TABLE Agendamento ADD CONSTRAINT `fk_Agendarmento_idDiscente`  
FOREIGN KEY (idDiscente) REFERENCES Discente(idDiscente);
```

```
CREATE TABLE Disponibilidade(  
idDisponivel INT PRIMARY KEY,  
nome VARCHAR(45),  
senha VARCHAR(45),  
numMatricula VARCHAR(14),  
peso FLOAT,  
sexo VARCHAR(20),  
idade INT,  
altura FLOAT);
```

```
CREATE TABLE Tutor(  
idTutor INT PRIMARY KEY,  
nome VARCHAR(45),  
senha VARCHAR(45),  
numRegistro VARCHAR(14),  
idAdmin INT);
```

```
ALTER TABLE Tutor ADD CONSTRAINT `fk_Tutor_idAdmin`  
FOREIGN KEY (idAdmin) REFERENCES Administrador(idAdmin);
```

```
CREATE TABLE Administrador(  
idAdmin INT PRIMARY KEY,  
nome VARCHAR(45),  
senha VARCHAR(45),  
numRegistro VARCHAR(14));
```

```
CREATE OR REPLACE FUNCTION apagarAgendamento(nomDiscente VARCHAR(45))  
RETURNS INTEGER AS  
$$  
DECLARE idDiscente INT;  
DECLARE idDisponivel INT;  
BEGIN  
    idDiscente = (SELECT idDiscente FROM Discente WHERE nome =  
nomeDiscente);  
    idDisponivel = (SELECT idDisponivel FROM Agendamento ag WHERE  
ag.idDiscente = idDiscente);  
  
    DELETE FROM Agendamento ag WHERE ag.idDiscente = idDiscente;  
    DELETE FROM Disponibilidade dp WHERE dp.idDisponivel = idDisponivel;  
  
    RETURN idDiscente;  
END;  
$$  
LANGUAGE plpgsql;
```

```
CREATE OR REPLACE PROCEDURE apagarAgendamento(nomDiscente VARCHAR(45))  
LANGUAGE SQL  
AS  
$$  
DECLARE idDiscente INT;  
DECLARE idDisponivel INT;  
  
    idDiscente = (SELECT idDiscente FROM Discente WHERE nome =  
nomeDiscente);  
    idDisponivel = (SELECT idDisponivel FROM Agendamento ag WHERE  
ag.idDiscente = idDiscente);  
  
    DELETE FROM Agendamento ag WHERE ag.idDiscente = idDiscente;  
    DELETE FROM Disponibilidade dp WHERE dp.idDisponivel = idDisponivel;  
  
$$;
```

