



Curso Superior de Desenvolvimento de Software Multiplataforma

Beatriz Cortez Siqueira, 3011392423003

Gabriel Lupateli Campanholi, 3011392423031

João Gabriel Milone Vilela de Camargo, 3011392423032

João Victor Quaresma de Arruda, 3011392423012

Wallacy José Oliveira de Souza, 301139243029

Projeto Interdisciplinar II

Banco de Dados Relacional

Engenharia de Software

Estude.My

Orientadores

Profº JONES ARTUR GOLÇALVES

Profª CRISTIANE PALOMAR MERCADO

Resumo do Projeto

O software Estude.my é um ambiente de estudos descontraído e confortável onde os jovens podem estudar de forma divertida. Esse software pode ser utilizado por alunos aleatórios que querem estudar por conta, ou por alunos de uma sala de aula, juntamente com seu professor através da plataforma.

O principal objetivo do Estude.my é facilitar a aprendizagem e abrir novas oportunidades de estudos.

Na atualidade, os jovens têm muita dificuldade de ficar muito tempo a enormes textos e vídeos longos, o que dificulta sua aprendizagem, já que muitos conteúdos são mostrados assim. Esses jovens podem ficar para trás em certos conhecimentos graças a isso, e muitos acabam desistindo dos estudos. Com nosso software, o estudo será fácil e divertido, dando esperança para esses jovens.

Além disso, professores que querem compartilhar seu conhecimento fora das escolas e expandir sua visibilidade, ou até mesmo melhorar seus estudos comuns na sala de aula, terão a possibilidade de se cadastrar como professor e criar suas próprias trilhas (aulas interativas), tanto para um público aberto quanto para uma sala fechada.

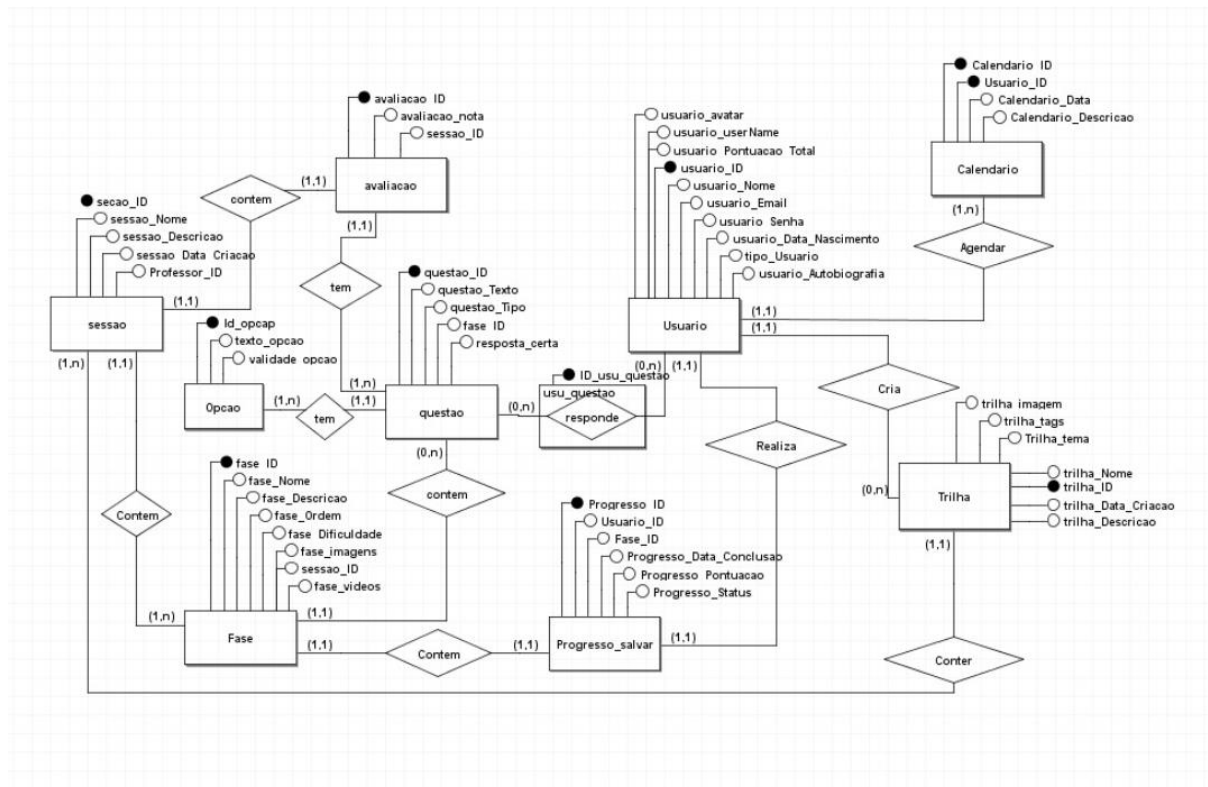
O Estude.my é liberado para todos, mas o principal público a ser atingido são os jovens que estão entre o ensino fundamental e ensino médio, sendo também útil a jovens mais novos.

Nosso projeto abrange duas ODS's, sendo elas:

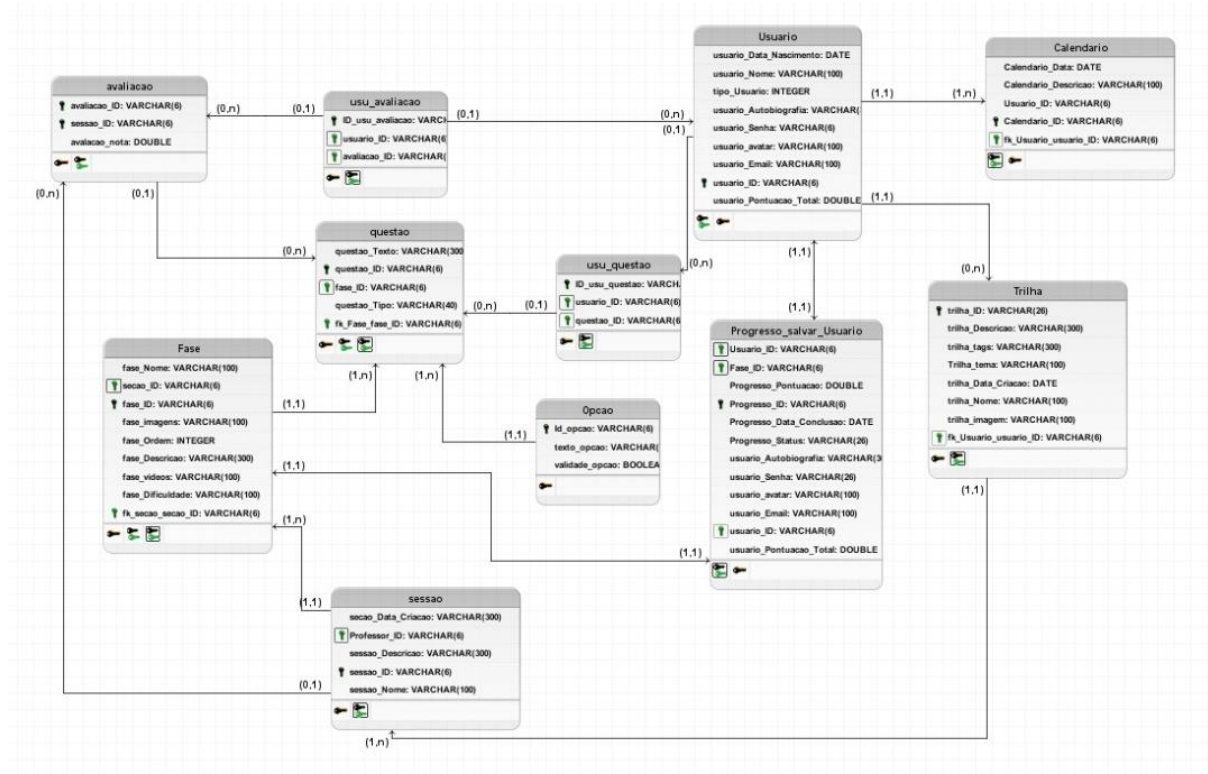
- Educação de qualidade;
- Trabalho decente, crescimento econômico;

A principal restrição desse software é relacionada ao cadastro de professores, não será permitido o cadastro de qualquer pessoa como professor, o candidato deve provar que está apto para dar sua aula.

Modelo Conceitual



Modelo Lógico



Script do Banco de Dados

```
create database EstudeMy
```

```
use EstudeMy
```

```
-- Tabela Usuario
```

```
CREATE TABLE Usuario
```

```
( usuario_ID VARCHAR(6) PRIMARY KEY, usuario_Data_Nascimento DATE,  
usuario_Nome VARCHAR(100), tipo_Usuario INTEGER, usuario_Autobiografia  
VARCHAR(300), usuario_Senha VARCHAR(26), usuario_avatar VARCHAR(100),  
usuario_Email VARCHAR(100), usuario_Pontuacao_Total DOUBLE PRECISION,  
usuario_userName VARCHAR(60)
```

```
);
```

```
-- Tabela Trilha
```

```
CREATE TABLE Trilha ( trilha_ID VARCHAR(26) PRIMARY KEY, trilha_Descricao  
VARCHAR(300), trilha_tags VARCHAR(300), trilha_tema VARCHAR(100),  
trilha_Data_Criacao DATE, trilha_Nome VARCHAR(100), trilha_imagem VARCHAR(100),  
fk_Usuario_usuario_ID VARCHAR(6), FOREIGN KEY (fk_Usuario_usuario_ID)  
REFERENCES Usuario(usuario_ID) );
```

```
-- Tabela Calendario
```

```
CREATE TABLE Calendario
```

```
( Calendario_ID VARCHAR(6) PRIMARY KEY, Calendario_Data DATE,  
Calendario_Descricao VARCHAR(100), Usuario_ID VARCHAR(6), FOREIGN KEY  
(Usuario_ID) REFERENCES Usuario(usuario_ID)
```

```
);
```

```
-- Tabela Secao
```

```
CREATE TABLE Secao
```

```
( secao_ID VARCHAR(6) PRIMARY KEY, secao_Nome VARCHAR(100), secao_Descricao  
VARCHAR(300), secao_Data_Criacao VARCHAR(50), Professor_ID VARCHAR(6)
```

);

-- Tabela Fase

CREATE TABLE Fase

(fase_ID VARCHAR(6) PRIMARY KEY, fase_Nome VARCHAR(100), secao_ID VARCHAR(6), fase_Descricao VARCHAR(300), fase_Ordem INTEGER, fase_imagens VARCHAR(100), fase_videos VARCHAR(100), fase_Dificuldade VARCHAR(10), fk_secao_secao_ID VARCHAR(6), FOREIGN KEY (fk_secao_secao_ID) REFERENCES Secao(secao_ID)

);

-- Tabela Avaliacao

CREATE TABLE Avaliacao

(avaliacao_ID VARCHAR(6) PRIMARY KEY, avaliacao_pergunta VARCHAR(200), avaliacao_nota DOUBLE PRECISION, secao_ID VARCHAR(6), FOREIGN KEY (secao_ID) REFERENCES Secao(secao_ID)

);

-- Tabela Usu_Avaliacao

CREATE TABLE Usu_Avaliacao

(usuario_ID VARCHAR(6), avaliacao_ID VARCHAR(6), PRIMARY KEY (usuario_ID, avaliacao_ID), FOREIGN KEY (usuario_ID) REFERENCES Usuario(usuario_ID), FOREIGN KEY (avaliacao_ID) REFERENCES Avaliacao(avaliacao_ID)

);

-- Tabela Questao

CREATE TABLE Questao

```
( questao_ID VARCHAR(6) PRIMARY KEY, questao_Texto VARCHAR(300), fase_ID  
VARCHAR(6), questao_Tipo VARCHAR(30), fk_Fase_fase_ID VARCHAR(6), FOREIGN  
KEY (fk_Fase_fase_ID) REFERENCES Fase(fase_ID)
```

```
);
```

```
-- Tabela Usu_Questao
```

```
CREATE TABLE Usu_Questao
```

```
( usuario_ID VARCHAR(6), questao_ID VARCHAR(6), PRIMARY KEY (usuario_ID,  
questao_ID), FOREIGN KEY (usuario_ID) REFERENCES Usuario(usuario_ID), FOREIGN  
KEY (questao_ID) REFERENCES Questao(questao_ID)
```

```
);
```

```
-- Tabela Opcao
```

```
CREATE TABLE Opcao ( id_opcao VARCHAR(6) PRIMARY KEY, texto_opcao  
VARCHAR(200), validade_opcao BIT
```

```
);
```

```
-- Tabela Progresso_Salvar_Usuario
```

```
CREATE TABLE Progresso_Salvar_Usuario
```

```
( usuario_ID VARCHAR(6), fase_ID VARCHAR(6), Progresso_Pontuacao DOUBLE  
PRECISION, Progresso_Data_Conclusao DATE, Progresso_Status VARCHAR(26),  
usuario_Data_Nascimento DATE, usuario_Nome VARCHAR(100), tipo_Usuario  
INTEGER, usuario_Autobiografia VARCHAR(300), usuario_Senha VARCHAR(26),  
usuario_avatar VARCHAR(100), usuario_Email VARCHAR(100),  
usuario_Pontuacao_Total DOUBLE PRECISION, PRIMARY KEY (usuario_ID, fase_ID),  
FOREIGN KEY (usuario_ID) REFERENCES Usuario(usuario_ID), FOREIGN KEY (fase_ID)  
REFERENCES Fase(fase_ID)
```

```
);
```

```
-- Inserção de dados das tabelas
```

--Insert de Usuario

INSERT INTO Usuario

```
( usuario_ID, usuario_Data_Nascimento, usuario_Nome, tipo_Usuario,
usuario_Autobiografia, usuario_Senha, usuario_avatar, usuario_Email,
usuario_Pontuacao_Total, usuario_userName ) VALUES ('U0001', '1990-01-01', 'Alice
Silva', 1, 'Autobio 1', 'senha123', 'avatar1.png', 'alice@email.com', 85.5, 'alice01'),
('U0002', '1985-03-15', 'Bruno Souza', 2, 'Autobio 2', 'senha456', 'avatar2.png',
'bruno@email.com', 72.0, 'bruno02'), ('U0003', '1992-07-22', 'Carla Mendes', 1, 'Autobio
3', 'senha789', 'avatar3.png', 'carla@email.com', 93.7, 'carla03'), ('U0004', '1995-12-30',
'Diego Rocha', 2, 'Autobio 4', 'senha101', 'avatar4.png', 'diego@email.com', 60.4,
'diego04'), ('U0005', '2000-05-10', 'Elaine Costa', 1, 'Autobio 5', 'senha202',
'avatar5.png', 'elaine@email.com', 78.9, 'elaine05');
```

--Insert de Seção

INSERT INTO Secao

```
( secao_ID, secao_Nome, secao_Descricao, secao_Data_Criacao, professor_ID )
VALUES ('S001', 'Matemática', 'Seção de matemática básica', '2023-01-10', 'P001'),
('S002', 'Português', 'Seção de gramática', '2023-02-11', 'P002'), ('S003', 'História',
'Seção de história geral', '2023-03-12', 'P003'), ('S004', 'Geografia', 'Seção de geografia
mundial', '2023-04-13', 'P004'), ('S005', 'Física', 'Seção de física básica', '2023-05-14',
'P005');
```

--Insert de Fase

INSERT INTO Fase

```
( fase_ID, fase_Nome, secao_ID, fase_Descricao, fase_imagens, fase_videos,
fase_Dificuldade, fase_Ordem ) VALUES ('F001', 'Fase 1', 'S001', 'Descrição fase 1',
'img1.png', 'vid1.mp4', 'Fácil', 1), ('F002', 'Fase 2', 'S001', 'Descrição fase 2', 'img2.png',
'vid2.mp4', 'Médio', 2), ('F003', 'Fase 3', 'S002', 'Descrição fase 3', 'img3.png',
'vid3.mp4', 'Difícil', 3), ('F004', 'Fase 4', 'S003', 'Descrição fase 4', 'img4.png', 'vid4.mp4',
'Fácil', 1), ('F005', 'Fase 5', 'S004', 'Descrição fase 5', 'img5.png', 'vid5.mp4', 'Médio', 2);
```

--Insert de Questão

INSERT INTO Questao

```
( questao_ID, questao_Texto, fase_ID, questao_Tipo ) VALUES ('Q001', 'Qual a capital
do Brasil?', 'F001', 'objetiva'), ('Q002', 'Resolva: 2+2', 'F001', 'objetiva'), ('Q003',
'Explique a Revolução Francesa.', 'F003', 'dissertativa'), ('Q004', 'O que é uma rocha
```

sedimentar?', 'F004', 'objetiva'), ('Q005', 'Defina velocidade média.', 'F005', 'dissertativa');

--Insert de Opção

INSERT INTO Opcao

(id_opcao, texto_opcao, validade_opcao) VALUES ('O001', 'Brasília', 1), ('O002', 'Rio de Janeiro', 0), ('O003', '4', 1), ('O004', '5', 0), ('O005', 'Paris', 1.5);

--Insert de Avaliação

INSERT INTO Avaliacao

(avaliacao_ID, avaliacao_Pergunta, avaliacao_Nota, secao_ID) VALUES ('A001', 'Qual a capital do Brasil?', 4.5, 'S001'), ('A002', 'Oque é insert no MySQL?', 4.0, 'S001'), ('A003', 'Qual a capital da França?', 5.0, 'S002'), ('A004', 'Oque é SELECT no MySQL?', 3.5, 'S003'), ('A005', 'Qual é a capital da Itália?', 4.7, 'S004');

--Insert de Usuario

Avaliação INSERT INTO

Usu_Avaliacao (usuario_ID, avaliacao_ID) VALUES ('U0001', 'A001'), ('U0002', 'A002'), ('U0003', 'A003'), ('U0004', 'A004'), ('U0005', 'A005');

--Insert de Usuario

Questão INSERT INTO

Usu_Questao (usuario_ID, questao_ID) VALUES ('U0001', 'Q001'), ('U0002', 'Q002'), ('U0003', 'Q003'), ('U0004', 'Q004'), ('U0005', 'Q005');

--Insert de Calendario

INSERT INTO

Calendario (calendario_ID, calendario_Data, calendario_Descricao, usuario_ID)
VALUES ('C001', '2024-01-01', 'Início do curso', 'U0001'), ('C002', '2024-01-15', 'Prova 1',
'U0002'), ('C003', '2024-02-01', 'Entrega de atividade', 'U0003'), ('C004', '2024-02-20',
'Revisão de conteúdo', 'U0004'), ('C005', '2024-03-01', 'Prova final', 'U0005');

--Insert de Trilha

INSERT INTO Trilha

(trilha_ID, trilha_Descricao, trilha_Tags, trilha_tema, trilha_Data_Criacao, trilha_Nome,
trilha_Imagem) VALUES ('T001', 'Trilha matemática básica', 'matematica,basica',
'Matemática', '2024-01-01', 'Matemática', 'img1.png'), ('T002', 'Trilha de português',
'portugues,gramatica', 'Português', '2024-01-05', 'Português', 'img2.png'), ('T003', 'Trilha
história moderna', 'historia,moderna', 'História', '2024-01-10', 'História', 'img3.png'),
('T004', 'Trilha de física', 'fisica,basica', 'Física', '2024-01-15', 'Física', 'img4.png'),
('T005', 'Trilha de geografia', 'geografia,mundo', 'Geografia', '2024-01-20', 'Geografia',
'img5.png');

--Insert de Progresso Slavar Usuario

INSERT INTO

Progresso_Salvar_Usuario (usuario_ID, fase_ID, progresso_Pontuacao,
Progresso_Data_Conclusao, progresso_Status, usuario_Data_Nascimento,
usuario_Nome, tipo_Usuario, usuario_Autobiografia, usuario_Senha, usuario_avatar,
usuario_Email, usuario_Pontuacao_Total) VALUES ('U0001', 'F001', 80.0, '2024-01-10',
'Concluído', '1990-01-01', 'Alice Silva', 1, 'Autobio 1', 'senha123', 'avatar1.png',
'alice@email.com', 85.5), ('U0002', 'F002', 70.0, '2024-01-12', 'Concluído', '1985-03-15',
'Bruno Souza', 2, 'Autobio 2', 'senha456', 'avatar2.png', 'bruno@email.com', 72.0),
('U0003', 'F003', 90.0, '2024-01-14', 'Concluído', '1992-07-22', 'Carla Mendes', 1,
'Autobio 3', 'senha789', 'avatar3.png', 'carla@email.com', 93.7), ('U0004', 'F004', 60.0,
'2024-01-16', 'Concluído', '1995-12-30', 'Diego Rocha', 2, 'Autobio 4', 'senha101',
'avatar4.png', 'diego@email.com', 60.4), ('U0005', 'F005', 75.0, '2024-01-18',
'Concluído', '2000-05-10', 'Elaine Costa', 1, 'Autobio 5', 'senha202', 'avatar5.png',
'elaine@email.com', 78.9);

--Exemplos de Select

--Listar todos os usuários

SELECT * FROM Usuario;

--Buscar todos os usuários com pontuação acima de 80

```
SELECT usuario_ID, usuario_Nome, usuario_Pontuacao_Total FROM Usuario WHERE
usuario_Pontuacao_Total > 80;
```

--Listar fases de uma seção específica (ex: Matemática - S001)

```
SELECT fase_ID, fase_Nome, fase_Dificuldade, fase_Ordem FROM Fase WHERE
secao_ID = 'S001' ORDER BY fase_Ordem;
```

--Listar todas as questões e suas opções

```
SELECT q.questao_ID, q.questao_Texto, o.texto_opcao, o.validade_opcao FROM
Questao q LEFT JOIN Opcao o ON q.questao_ID = o.questao_ID ORDER BY
q.questao_ID;
```

--Mostrar avaliações

```
SELECT a.avaliacao_ID, a.avaliacao_Pergunta, u.usuario_Nome FROM Avaliacao a
INNER JOIN Usu_Avaliacao ua ON a.avaliacao_ID = ua.avaliacao_ID INNER JOIN
Usuario u ON ua.usuario_ID = u.usuario_ID;
```

--Progresso de cada usuário por fase

```
SELECT u.usuario_Nome, f.fase_Nome, p.progresso_Pontuacao, p.progresso_Status
FROM Progresso_Salvar_Usuario p JOIN Usuario u ON p.usuario_ID = u.usuario_ID JOIN
Fase f ON p.fase_ID = f.fase_ID ORDER BY u.usuario_Nome;
```

--Trilhas criadas por usuários com filtro por categoria

```
SELECT t.trilha_Nome, t.trilha_tema, u.usuario_Nome FROM Trilha t JOIN Usuario u ON
u.usuario_ID = t.usuario_ID WHERE t.trilha_tema = 'História';
```

--Contar número de questões por fase

```
SELECT f.fase_Nome, COUNT(q.questao_ID) AS total_questoes FROM Fase f LEFT JOIN
Questao q ON f.fase_ID = q.fase_ID GROUP BY f.fase_Nome;
```

--Buscar todos os eventos do calendário de um usuário específico

```
SELECT c.calendario_Data, c.calendario_Descricao, u.usuario_Nome FROM  
Calendario c JOIN Usuario u ON c.usuario_ID = u.usuario_ID WHERE  
u.usuario_userName = 'alice01';
```