Éden Samuel

Felipe Carneiro

Fernando Lopes

Henrique Ricardo

Hugo Takeda

RELAÇÃO DE ARTEFATOS

ESPECIFICAÇÃO DO PROJETO

- FacilitaU-

[LINK TRELLO](https://trello.com/invite/b/67d2d7a463360c050382092f/ATTI0de97b2f0d5f5bdf1ab4dedfbbe6155d4423BE76/experiencia-criativa)

[LINK GITHUB](https://github.com/Riquehenri/FacilitaU)



Curitiba

2025

SUMÁRIO

[ARTEFATO 1: Quadro “3 Objetivos” 3](#_heading=h.gjdgxs)

[ARTEFATO 2: Quadro “é – não é – faz – não faz” 4](#_heading=h.3znysh7)

[ARTEFATO 3: Quadro “Visão de Produto”. 5](#_heading=h.3dy6vkm)

[ARTEFATO 4: Canvas PBB 6](#_heading=h.2s8eyo1)

[ARTEFATO 5: Relação de User Stories 7](#_heading=h.3rdcrjn)

[ARTEFATO 6: Modelo Relacional](#_heading=h.35nkun2) 14

[ARTEFATO 7: Diagrama de Classes](#_heading=h.44sinio) 22

[ARTEFATO 8: Diagrama de Atividades](#_heading=h.z337ya) 23

[REFERÊNCIA BIBLIOGRÁFICAS](#_heading=h.1y810tw) 24

ÍNDICE DE ILUSTRAÇÕES

[Figura 1 – Quadro “3 Objetivos”. 3](#_heading=h.1fob9te)

[Figura 2 – Quadro “é – não é – faz – não faz”. 4](#_heading=h.tyjcwt)

[Figura 3 – Quadro "Visão de Produto". Fonte: AGUIAR, F. 2018. 5](#_heading=h.4d34og8)

[Figura 4 – Canvas PBB: "Product Backlog Building". Fonte: AGUIAR, F. 2018. 6](#_heading=h.17dp8vu)

[Figura 5 - User Stories e Critérios de Aceite. Fonte: AGUIAR, F. 2018. 7](#_heading=h.lnxbz9)

[Figura 6 – Exemplo: Modelo Relacional gerado por engenharia reversa (MySQL Workbench).](#_heading=h.1ksv4uv) 14

[Figura 7 – Exemplo: Diagrama de Classes.](#_heading=h.2jxsxqh) 22

[Figura 8 – Exemplo: Diagrama de Atividades.](#_heading=h.3j2qqm3) 23

# ARTEFATO 1: Quadro “3 Objetivos”

| **ARTEFATO 1**: Quadro “3 Objetivos” | |
| --- | --- |
| **NOME DO PRODUTO**: FacilitaU | |
| **OBJETIVOS** | **DESCRIÇÃO** |
| 1 | Melhorar a organização dos estudantes permitindo o gerenciamento de tarefas e horários acadêmicos de forma simples e acessível. |
| 2 | Facilitar o acesso rápido a informações institucionais e suporte financeiro, como bolsas, auxílios e prazos de pagamento. |
| 3 | Oferecer suporte básico de bem-estar e ferramentas para controle pessoal. |

**Figura 1 – Quadro “3 Objetivos”.**

# ARTEFATO 2: Quadro “é – não é – faz – não faz”

| **ARTEFATO 2**: Quadro “é – não é – faz – não faz” | |
| --- | --- |
| **NOME DO PRODUTO**: FacilitaU | |
| **É**  Uma ferramenta digital de suporte acadêmico para estudantes universitários.  Simples, intuitiva e focada no aluno, com apoio à organização pessoal. | **Não é**  Um sistema complexo de gestão acadêmica ou financeiro para professores ou administradores da universidade.  Um software de contabilidade avançada ou um substituto para plataformas oficiais de pagamento da universidade. |
| **Faz**  Gerencia tarefas, envia lembretes de prazos e exibe opções de suporte (bolsas, auxílios).  Responde perguntas básicas sobre a faculdade e ajuda no planejamento acadêmico. | **Não faz**  Realiza transações financeiras diretas, como pagamentos de mensalidades ou solicitações formais de bolsas.  Substitui serviços oficiais da universidade ou fornece aconselhamento financeiro profissional. |

**Figura 2 – Quadro “é – não é – faz – não faz”.**

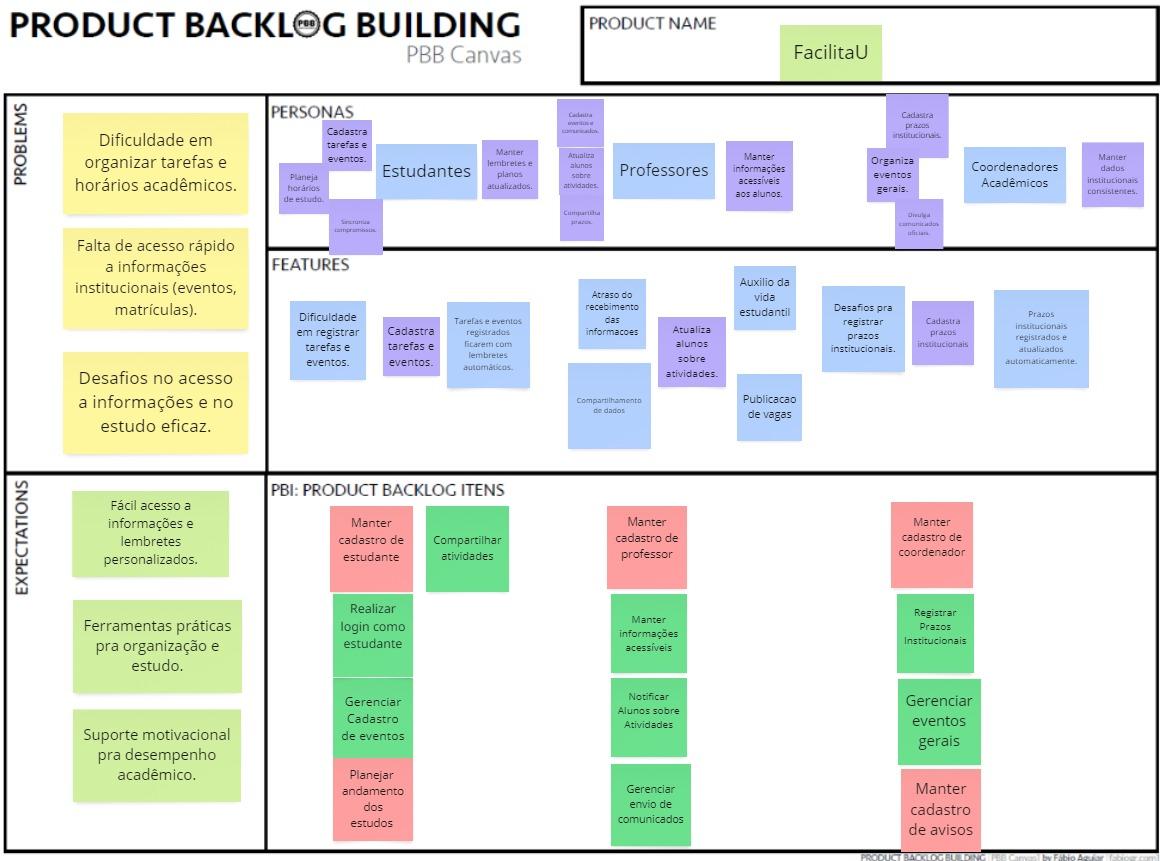
# ARTEFATO 3: Quadro “Visão de Produto”.

| **ARTEFATO 3**: Quadro “Visão de Produto” | |
| --- | --- |
| **NOME DO PRODUTO**: FacilitaU | |
| **CLIENTE-ALVO** | Quem são os nossos clientes mais importantes?  - Estudantes universitários que buscam organização e suporte acadêmico . |
| **CATEGORIA-SEGMENTO** | O que é o produto?  -Um aplicativo mobile e web de assistência pessoal para estudantes. |
| **BENEFÍCIO-CHAVE** | Qual é o principal benefício,​  Razão para adquiri-lo?​  - Organização simplificada, suporte e bem-estar acessível em tempo real. |
| **DIFERENCIADO-CHAVE** | Qual o diferencial do projeto? ​  Algo inovador...​  -Interface amigável com foco em bem-estar e gestão integrada à rotina acadêmica com auxílio de uma assistente virtual. |
| **META-VALOR.** | Qual impacto do valor?​  -Melhor suporte e auxílio acadêmico ao estudante |

**Figura 3 – Quadro "Visão de Produto". Fonte: AGUIAR, F. 2018.**

# ARTEFATO 4: Canvas PBB

**Link do Miro**: [LINK\_MIRO\_PBB](https://miro.com/welcomeonboard/eGdkU0hjMUNEWEU3MTc4cHZnKzNjMnJMU1A2WC85UWJZK2s3c2UvR0d4QUs1QWJWU3ZMVUhjRHZRNmF0eGM4ZjhuUTZxREdnVE9nalZuMkJnRC9qemlIOHlMVWZVUVBsc1dZSHg5VnRwWmZhblVhTkpkNnd3ZnZHajlXK0hhMWhyVmtkMG5hNDA3dVlncnBvRVB2ZXBnPT0hdjE=?share_link_id=219191628507)



**Figura 4 – Canvas PBB: "Product Backlog Building". Fonte: AGUIAR, F. 2018.**

# ARTEFATO 5: Relação de User Stories

| HISTÓRIA DO USUÁRIO 1 - PBI: Manter cadastro de estudante | |
| --- | --- |
| **COMO**: Estudante universitário.  **POSSO**: Manter cadastro de estudante.  **PARA**: Acessar todas as funcionalidades do sistema. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O aluno acessa a página de cadastro.  **QUANDO**: Ele preenche seus dados(E-mail,senha,nome) e confirma.  **ENTÃO**: O sistema cria seu perfil e envia uma confirmação. |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: O aluno deseja editar suas informações(E-mail,senha,nome).  **QUANDO**: Ele acessa as configurações de perfil.  **ENTÃO**: O sistema permite modificar seus dados(E-mail,senha,nome). |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O aluno esqueceu sua senha.  **QUANDO**: Ele solicita a recuperação.  **ENTÃO**: O sistema envia um link para redefinição no E-mail cadastrado . |

| HISTÓRIA DO USUÁRIO 2 - PBI: Realizar login como estudante | |
| --- | --- |
| **COMO**: Estudante universitário.  **POSSO**: Realizar login como estudante  **PARA**: Acessar informações acadêmicas | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: Acessa a página de login  **QUANDO**: insere senha e e-mail corretamente  **ENTÃO**: O sistema permite o acesso a conta |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: O estudante insere credenciais inválidas  **QUANDO**: tenta realizar o login  **ENTÃO**: O sistema exibe uma mensagem de erro informando que os dados estão incorretos |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O estudante esqueceu a senha  **QUANDO**: Acessa opção de recuperação de senha  **ENTÃO**: O sistema envia um e-mail com instruções para redefinir senha |

| HISTÓRIA DO USUÁRIO 3 - PBI: Gerenciar Cadastro de Eventos | |
| --- | --- |
| **COMO**: Estudante universitário.  **POSSO**: Gerenciar Cadastro de Eventos.  **PARA**: Manter as informações sempre atualizadas. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O Estudante universitário acessa a tela de cadastro de eventos.  **QUANDO**: Ele preenche o evento "Seminário de Biologia dia 20" e clica em "Cadastrar Tarefa/Evento".  **ENTÃO**:O sistema registra o evento e exibe "Evento cadastrado: Seminário de Biologia dia 20". |
| **Critério de**  **Aceite 2** | **DADO** **QUE**:O Estudante universitário cadastrou o evento "Seminário de Biologia dia 20"  **QUANDO**: Ele edita o evento para "Seminário de Biologia dia 21" e confirma.  **ENTÃO**: sistema atualiza o evento e exibe "Evento atualizado: Seminário de Biologia dia 21". |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O Estudante universitário deseja verificar os eventos cadastrados**.**  **QUANDO**: Ele acessa a lista de eventos.  **ENTÃO**: O sistema mostra o evento "Seminário de Biologia dia 21" na lista. |

| HISTÓRIA DO USUÁRIO 4 - PBI: Planejar Andamento dos Estudos | |
| --- | --- |
| **COMO**: Estudante Universitário.  **POSSO**: Planejar Andamento dos Estudos.  **PARA**: Acompanhar o progresso e garantir que está seguindo o planejamento adequado. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: o aluno deseja planejar seu estudo.  **QUANDO**: ele informa as matérias e a carga horária disponíveis na sua programação.  **ENTÃO**: o sistema gera um cronograma de estudos baseado no tempo disponível e na proximidade das avaliações. |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: o aluno está seguindo o plano de estudos.  **QUANDO**: ele marca um conteúdo como concluído.  **ENTÃO**: o sistema atualiza seu progresso e exibe a porcentagem de conclusão. |
| **Critério de**  **Aceite 3** | **DADO** **QUE**: o aluno deseja ajustar seu planejamento.  **QUANDO**: ele modifica a carga horária ou adiciona novas matérias.  **ENTÃO**: o sistema recalcula automaticamente o cronograma e sugere ajustes para manter o equilíbrio dos estudos. |

| HISTÓRIA DO USUÁRIO 5 - PBI: Gerenciar compartilhamento de atividades | |
| --- | --- |
| **COMO**: Estudante universitário.  **POSSO**: Gerenciar Compartilhamento de Atividades  **PARA**: Facilitar a colaboração acadêmica. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O aluno deseja compartilhar um material.  **QUANDO**: Ele seleciona um arquivo e um destinatário.  **ENTÃO**: O sistema envia o material ao colega. |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: Um aluno recebe um material.  **QUANDO**: Ele abre a notificação.  **ENTÃO**: O sistema permite visualizar e baixar o conteúdo. |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O aluno compartilhou um material por engano.  **QUANDO**: Ele acessa seu histórico de envios.  **ENTÃO**: O sistema permite excluir o compartilhamento. |

| HISTÓRIA DO USUÁRIO 6 - PBI: Manter cadastro de professor | |
| --- | --- |
| **COMO**: Professor.  **POSSO**: Manter cadastro de professor.  **PARA**: Permitir que utilizem funcionalidades do sistemas | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O Professor acessa a página de cadastro.  **QUANDO**: Ele preenche seus dados e confirma.  **ENTÃO**: O sistema cria seu perfil e envia uma confirmação. |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: O Professor deseja editar suas informações.  **QUANDO**: Ele acessa as configurações de perfil.  **ENTÃO**: O sistema permite modificar seus dados. |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O Professor esqueceu sua senha.  **QUANDO**: Ele solicita a recuperação.  **ENTÃO**: O sistema envia um link para redefinição. |

| HISTÓRIA DO USUÁRIO 7 - PBI: Manter informações acessíveis | |
| --- | --- |
| **COMO**: Professor.  **POSSO**: Manter informações acessíveis.  **PARA**: Garantir que os alunos recebam comunicados rapidamente. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O Professor acessa a tela de comunicados.  **QUANDO**: Ele preenche o comunicado "Aula extra dia 25" e clica em "Cadastrar Evento/Comunicado".  **ENTÃO**:O sistema registra o comunicado e exibe "Comunicado cadastrado: Aula extra dia 25". |
| **Critério de**  **Aceite 2** | **DADO** **QUE**:O Professor cadastrou o comunicado "Aula extra dia 25".  **QUANDO**: Ele confirma o envio para os alunos.  **ENTÃO**: O sistema notifica os alunos com "Novo comunicado: Aula extra dia 25". |
| **Critério de**  **Aceite 3** | **DADO** **QUE**: O Professor deseja verificar se os alunos receberam o comunicado.  **QUANDO**: Ele acessa a lista de comunicados enviados.  **ENTÃO**: O sistema mostra o comunicado "Aula extra dia 25" com status "Enviado". |

| HISTÓRIA DO USUÁRIO 8 - PBI: Notificar alunos sobre atividades | |
| --- | --- |
| **COMO**: Professor.  **POSSO**: Notificar alunos sobre atividades.  **PARA**: Manter os alunos informados e engajados nas atividades acadêmicas | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: Eu cadastrei uma atividade "Seminário de Física dia 22".  **QUANDO**: Eu clico no botão "Notificar Alunos".  **ENTÃO**:O FacilitaU envia uma notificação aos alunos com "Nova atividade: Seminário de Física dia 22". |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: Eu cadastrei a atividade e notifiquei os alunos.  **QUANDO**: Faltam 2 dias para a atividade (dia 20).  **ENTÃO**: O FacilitaU envia um lembrete com "Lembrete: Seminário de Física dia 22". |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:Um aluno recebeu a notificação  **QUANDO**: O aluno visualiza a notificação.  **ENTÃO**: O FacilitaU exibe a opção "Confirmar Participação" para o aluno responder. |

| HISTÓRIA DO USUÁRIO 9 - PBI: Gerenciar envio de comunicados | |
| --- | --- |
| **COMO**: Professor.  **POSSO**: **:** Gerenciar envio de comunicados.  **PARA**: Distribuir comunicados rapidamente aos alunos. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O Professor acessa a tela de comunicados.  **QUANDO**: Ele preenche o comunicado "Aula extra dia 25" e clica em "Cadastrar Evento/Comunicado".  **ENTÃO**:O sistema registra o comunicado e exibe "Comunicado cadastrado: Aula extra dia 25". |
| **Critério de**  **Aceite 2** | **DADO** **QUE**:O Professor cadastrou o comunicado "Aula extra dia 25".  **QUANDO**: Ele confirma o envio para os alunos.  **ENTÃO**: O sistema notifica os alunos com "Novo comunicado: Aula extra dia 25". |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O Professor deseja verificar os comunicados enviados.  **QUANDO**: Ele acessa a lista de comunicados.  **ENTÃO**: O sistema exibe o comunicado "Aula extra dia 25" com status "Enviado". |

| HISTÓRIA DO USUÁRIO 10 - PBI: Manter cadastro de coordenador | |
| --- | --- |
| **COMO**: Coordenadores acadêmicos.  **POSSO**: Manter cadastro de coordenador  **PARA**: Permitir que utilizem funcionalidades do sistema | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O Coordenador acessa a página de cadastro.  **QUANDO**: Ele preenche seus dados e confirma.  **ENTÃO**: O sistema cria seu perfil e envia uma confirmação. |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: O Coordenador deseja editar suas informações.  **QUANDO**: Ele acessa as configurações de perfil.  **ENTÃO**: O sistema permite modificar seus dados. |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O Coordenador esqueceu sua senha.  **QUANDO**: Ele solicita a recuperação.  **ENTÃO**: O sistema envia um link para redefinição. |

| HISTÓRIA DO USUÁRIO 11 - PBI:Registrar Prazos institucionais | |
| --- | --- |
| **COMO**: Coordenadores acadêmicos.  **POSSO**: Registrar prazos institucionais  **PARA**: Manter prazos institucionais registrados e atualizados automaticamente. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O Coordenador Acadêmico acessa a tela de cadastro de prazos.  **QUANDO**: Ele preenche o prazo "Matrícula até dia 25" e clica em "Cadastrar Prazo Institucional".  **ENTÃO**:O sistema registra o prazo e exibe "Prazo cadastrado: Matrícula até dia 25". |
| **Critério de**  **Aceite 2** | **DADO** **QUE**:O Coordenador Acadêmico cadastrou o prazo "Matrícula até dia 25".  **QUANDO**: O sistema atualiza os dados automaticamente.  **ENTÃO**: O sistema exibe na seção "Prazos" da tela inicial "Matrícula até dia 25". |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O Coordenador Acadêmico deseja verificar os prazos cadastrados**.**  **QUANDO**: Ele acessa a lista de prazos.  **ENTÃO**: O sistema mostra o prazo "Matrícula até dia 25" com status "Ativo". |

| HISTÓRIA DO USUÁRIO 12 - PBI: Gerenciar eventos gerais | |
| --- | --- |
| **COMO**: Coordenadores acadêmicos.  **POSSO**: Gerenciar eventos gerais.  **PARA**: Manter eventos gerais organizados e acessíveis aos alunos. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O Coordenador Acadêmico acessa a tela de eventos gerais.  **QUANDO**: Ele preenche o evento "Feira de Ciências dia 30" e clica em "Cadastrar Evento".  **ENTÃO**:O sistema registra o evento e exibe "Evento cadastrado: Feira de Ciências dia 30". |
| **Critério de**  **Aceite 2** | **DADO** **QUE**:O Coordenador Acadêmico cadastrou o evento "Feira de Ciências dia 30".  **QUANDO**: Ele confirma o envio para os alunos.  **ENTÃO**: O sistema notifica os alunos com "Novo evento: Feira de Ciências dia 30". |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O Coordenador Acadêmico deseja verificar os eventos cadastrados.  **QUANDO**: Ele acessa a lista de eventos gerais.  **ENTÃO**: O sistema mostra o evento "Feira de Ciências dia 30" com status "Enviado". |

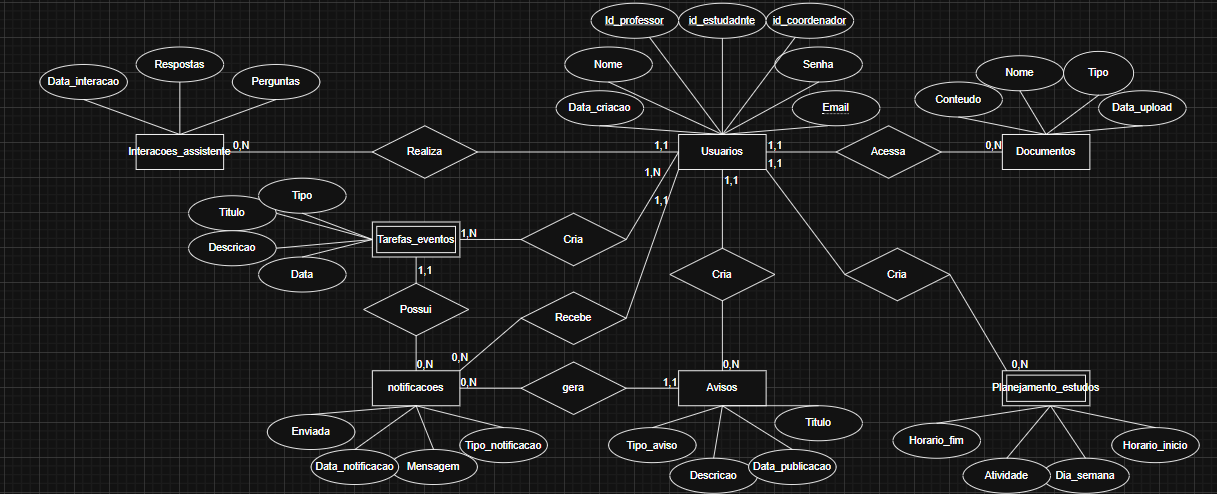
| HISTÓRIA DO USUÁRIO 13 - PBI: Manter cadastro de avisos | |
| --- | --- |
| **COMO**: Coordenadores acadêmicos.  **POSSO**: Manter cadastro de avisos  **PARA**: Manter avisos consistentes e acessíveis aos alunos. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: O Coordenador Acadêmico acessa a tela de cadastro de avisos.  **QUANDO**: Ele preenche o aviso "Suspensão de aulas dia 28" e clica em "Cadastrar Aviso".  **ENTÃO**:O sistema registra o aviso e exibe "Aviso cadastrado: Suspensão de aulas dia 28". |
| **Critério de**  **Aceite 2** | **DADO** **QUE**:O Coordenador Acadêmico cadastrou o aviso "Suspensão de aulas dia 28".  **QUANDO**: Ele confirma o envio para os alunos.  **ENTÃO**: O sistema notifica os alunos com "Novo aviso: Suspensão de aulas dia 28". |
| **Critério de**  **Aceite 3** | **DADO** **QUE**:O Coordenador Acadêmico deseja verificar os avisos cadastrados**.**  **QUANDO**: Ele acessa a lista de avisos.  **ENTÃO**: O sistema mostra o aviso "Suspensão de aulas dia 28" com status "Enviado". |

**Figura 5 - User Stories e Critérios de Aceite. Fonte: AGUIAR, F. 2018.**

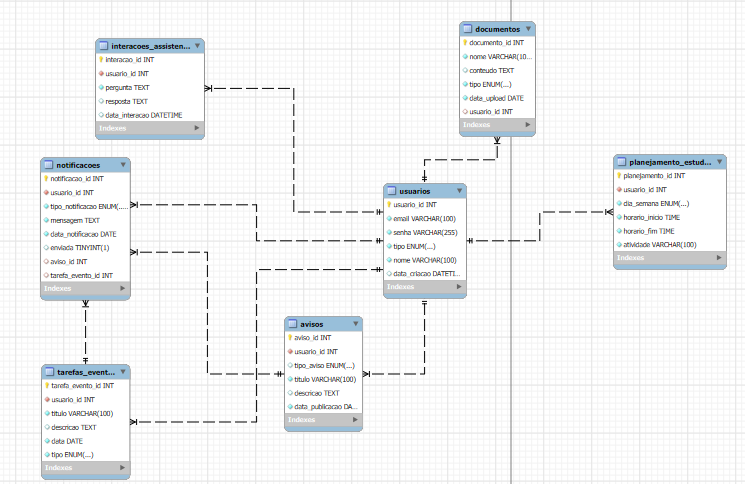
# ARTEFATO 6: Modelo Relacional

[LINK DRAW.IO](https://drive.google.com/file/d/11AQUq5HqftuOoRy0qSh2RrmtO7v0puyN/view?usp=sharing)

Modelo Conceitual



Modelo Lógico



Modelo Físico

| Views.sql |
| --- |
| CREATE VIEW AvisosComAutor AS  SELECT  a.aviso\_id,  a.tipo\_aviso,  a.titulo,  a.descricao,  a.data\_publicacao,  u.nome AS autor,  u.tipo AS tipo\_autor  FROM Avisos a  JOIN Usuarios u ON a.usuario\_id = u.usuario\_id;  CREATE VIEW NotificacoesPendentes AS  SELECT  n.notificacao\_id,  n.usuario\_id,  u.nome AS nome\_estudante,  n.tipo\_notificacao,  n.mensagem,  n.data\_notificacao,  n.aviso\_id,  n.tarefa\_evento\_id  FROM Notificacoes n  JOIN Usuarios u ON n.usuario\_id = u.usuario\_id  WHERE n.enviada = FALSE;  CREATE VIEW PlanejamentoPorEstudante AS  SELECT  p.planejamento\_id,  p.usuario\_id,  u.nome AS nome\_estudante,  p.dia\_semana,  p.horario\_inicio,  p.horario\_fim,  p.atividade  FROM Planejamento\_Estudos p  JOIN Usuarios u ON p.usuario\_id = u.usuario\_id  ORDER BY  u.nome,  FIELD(p.dia\_semana, 'segunda', 'terca', 'quarta', 'quinta', 'sexta', 'sabado', 'domingo'),  p.horario\_inicio;  CREATE VIEW TarefasEventosProximos AS  SELECT  te.tarefa\_evento\_id,  te.usuario\_id,  u.nome AS nome\_estudante,  te.titulo,  te.descricao,  te.data,  te.tipo  FROM Tarefas\_Eventos te  JOIN Usuarios u ON te.usuario\_id = u.usuario\_id  WHERE te.data BETWEEN CURDATE() AND DATE\_ADD(CURDATE(), INTERVAL 3 DAY);  CREATE VIEW DocumentosPorTipo AS  SELECT  d.documento\_id,  d.nome,  d.conteudo,  d.tipo,  d.data\_upload,  u.nome AS autor  FROM Documentos d  LEFT JOIN Usuarios u ON d.usuario\_id = u.usuario\_id;  CREATE VIEW InteracoesPorEstudante AS  SELECT  i.interacao\_id,  i.usuario\_id,  u.nome AS nome\_estudante,  i.pergunta,  i.resposta,  i.data\_interacao  FROM Interacoes\_Assistente i  JOIN Usuarios u ON i.usuario\_id = u.usuario\_id  ORDER BY i.data\_interacao DESC;  CREATE VIEW UsuariosAtivos AS  SELECT  u.usuario\_id,  u.email,  u.tipo,  u.nome,  u.data\_criacao  FROM Usuarios u  WHERE u.data\_criacao >= DATE\_SUB(CURDATE(), INTERVAL 30 DAY);  CREATE VIEW AvisosPorTipo AS  SELECT  tipo\_aviso,  COUNT(\*) AS total,  MAX(data\_publicacao) AS ultima\_publicacao  FROM Avisos  GROUP BY tipo\_aviso; |

| codigo base.sql |
| --- |
| -- Banco de dados  CREATE DATABASE facilitau\_db;  USE facilitau\_db;  -- Tabela Usuarios (estudantes, professores e coordenadores)  CREATE TABLE Usuarios (  usuario\_id INT AUTO\_INCREMENT PRIMARY KEY,  email VARCHAR(100) UNIQUE NOT NULL,  senha VARCHAR(255) NOT NULL,  tipo ENUM('estudante', 'professor', 'coordenador') NOT NULL,  nome VARCHAR(100) NOT NULL,  data\_criacao DATETIME DEFAULT CURRENT\_TIMESTAMP  );  -- Tabela Avisos (avisos gerais e oportunidades)  CREATE TABLE Avisos (  aviso\_id INT AUTO\_INCREMENT PRIMARY KEY,  usuario\_id INT NOT NULL,  tipo\_aviso ENUM('aviso', 'oportunidade') DEFAULT 'aviso',  titulo VARCHAR(100) NOT NULL,  descricao TEXT,  data\_publicacao DATE NOT NULL,  FOREIGN KEY (usuario\_id) REFERENCES Usuarios(usuario\_id) ON DELETE CASCADE  );  -- Tabela Documentos (para suporte à assistente virtual)  CREATE TABLE Documentos (  documento\_id INT AUTO\_INCREMENT PRIMARY KEY,  nome VARCHAR(100) NOT NULL,  conteudo TEXT,  tipo ENUM('contrato', 'regulamento', 'outro') NOT NULL,  data\_upload DATE NOT NULL,  usuario\_id INT,  FOREIGN KEY (usuario\_id) REFERENCES Usuarios(usuario\_id) ON DELETE SET NULL  );  -- Tabela Perguntas\_Respostas (para a assistente virtual)  CREATE TABLE Perguntas\_Respostas (  pergunta\_resposta\_id INT AUTO\_INCREMENT PRIMARY KEY,  pergunta TEXT NOT NULL,  resposta TEXT NOT NULL,  categoria VARCHAR(50),  data\_criacao DATETIME DEFAULT CURRENT\_TIMESTAMP  );  -- Tabela Tarefas\_Eventos (tarefas e eventos dos estudantes)  CREATE TABLE Tarefas\_Eventos (  tarefa\_evento\_id INT AUTO\_INCREMENT PRIMARY KEY,  usuario\_id INT NOT NULL,  titulo VARCHAR(100) NOT NULL,  descricao TEXT,  data DATE NOT NULL,  tipo ENUM('tarefa', 'evento') NOT NULL,  FOREIGN KEY (usuario\_id) REFERENCES Usuarios(usuario\_id) ON DELETE CASCADE  );  -- Tabela Planejamento\_Estudos (rotina acadêmica dos estudantes)  CREATE TABLE Planejamento\_Estudos (  planejamento\_id INT AUTO\_INCREMENT PRIMARY KEY,  usuario\_id INT NOT NULL,  dia\_semana ENUM('segunda', 'terca', 'quarta', 'quinta', 'sexta', 'sabado', 'domingo') NOT NULL,  horario\_inicio TIME NOT NULL,  horario\_fim TIME NOT NULL,  atividade VARCHAR(100) NOT NULL,  FOREIGN KEY (usuario\_id) REFERENCES Usuarios(usuario\_id) ON DELETE CASCADE  );  -- Tabela Notificacoes (para lembretes e avisos automáticos)  CREATE TABLE Notificacoes (  notificacao\_id INT AUTO\_INCREMENT PRIMARY KEY,  usuario\_id INT NOT NULL,  tipo\_notificacao ENUM('lembrete', 'aviso') NOT NULL,  mensagem TEXT NOT NULL,  data\_notificacao DATE NOT NULL,  enviada BOOLEAN DEFAULT FALSE,  aviso\_id INT NULL,  tarefa\_evento\_id INT NULL,  FOREIGN KEY (usuario\_id) REFERENCES Usuarios(usuario\_id) ON DELETE CASCADE,  FOREIGN KEY (aviso\_id) REFERENCES Avisos(aviso\_id) ON DELETE SET NULL,  FOREIGN KEY (tarefa\_evento\_id) REFERENCES Tarefas\_Eventos(tarefa\_evento\_id) ON DELETE SET NULL  );  -- Tabela Interacoes\_Assistente (histórico de interações com a assistente virtual)  CREATE TABLE Interacoes\_Assistente (  interacao\_id INT AUTO\_INCREMENT PRIMARY KEY,  usuario\_id INT NOT NULL,  pergunta TEXT NOT NULL,  resposta TEXT,  data\_interacao DATETIME DEFAULT CURRENT\_TIMESTAMP,  FOREIGN KEY (usuario\_id) REFERENCES Usuarios(usuario\_id) ON DELETE CASCADE  ); |

| dados falsos.sql |
| --- |
| -- Inserir usuários de teste  INSERT INTO Usuarios (email, senha, tipo, nome) VALUES  ('estudante1@facilitau.com', '$2y$10$z1X2Y3W4Q5E6R7T8U9I0O.kJ2L3M4N5P6Q7R8S9T0U1V2W3X4Y5Z6', 'estudante', 'João Silva'),  ('professor1@facilitau.com', '$2y$10$z1X2Y3W4Q5E6R7T8U9I0O.kJ2L3M4N5P6Q7R8S9T0U1V2W3X4Y5Z6', 'professor', 'Maria Oliveira'),  ('coordenador1@facilitau.com', '$2y$10$z1X2Y3W4Q5E6R7T8U9I0O.kJ2L3M4N5P6Q7R8S9T0U1V2W3X4Y5Z6', 'coordenador', 'Carlos Souza');  -- Inserir avisos de teste  INSERT INTO Avisos (usuario\_id, tipo\_aviso, titulo, descricao, data\_publicacao) VALUES  (3, 'aviso', 'Reunião Geral', 'Reunião de coordenadores na próxima sexta-feira.', '2025-03-30'),  (2, 'oportunidade', 'Vaga de Estágio', 'Oportunidade de estágio em TI. Inscrições até 2025-04-05.', '2025-03-28');  -- Inserir documentos de teste  INSERT INTO Documentos (nome, conteudo, tipo, data\_upload, usuario\_id) VALUES  ('Contrato de Matrícula', 'Texto do contrato de matrícula...', 'contrato', '2025-03-01', 3),  ('Regulamento Acadêmico', 'Regras gerais do curso...', 'regulamento', '2025-03-01', NULL);  -- Inserir perguntas/respostas para a assistente virtual  INSERT INTO Perguntas\_Respostas (pergunta, resposta, categoria) VALUES  ('Como acessar o contrato?', 'Você pode acessar o contrato na seção de documentos.', 'documentação'),  ('Qual a data da próxima prova?', 'Consulte sua agenda ou os avisos publicados.', 'calendário');  -- Inserir tarefas/eventos de teste  INSERT INTO Tarefas\_Eventos (usuario\_id, titulo, descricao, data, tipo) VALUES  (1, 'Prova de Cálculo', 'Estudar capítulos 1 a 3.', '2025-04-01', 'tarefa'),  (1, 'Palestra de Carreira', 'Evento no auditório principal.', '2025-04-02', 'evento');  -- Inserir planejamento de estudos de teste  INSERT INTO Planejamento\_Estudos (usuario\_id, dia\_semana, horario\_inicio, horario\_fim, atividade) VALUES  (1, 'segunda', '09:00:00', '11:00:00', 'Estudar Cálculo'),  (1, 'terca', '14:00:00', '16:00:00', 'Revisar Notas de Física');  -- Inserir notificações de teste  INSERT INTO Notificacoes (usuario\_id, tipo\_notificacao, mensagem, data\_notificacao, aviso\_id, tarefa\_evento\_id) VALUES  (1, 'lembrete', 'Lembrete: Prova de Cálculo em 2025-04-01', '2025-03-30', NULL, 1),  (1, 'aviso', 'Novo aviso: Reunião Geral', '2025-03-30', 1, NULL);  -- Inserir interações com a assistente virtual  INSERT INTO Interacoes\_Assistente (usuario\_id, pergunta, resposta) VALUES  (1, 'Como acessar o contrato?', 'Você pode acessar o contrato na seção de documentos.'); |

| procedure.sql |
| --- |
| DELIMITER //  CREATE PROCEDURE InserirNotificacaoAviso(  IN p\_aviso\_id INT,  IN p\_usuario\_id INT,  IN p\_titulo VARCHAR(100),  IN p\_data\_publicacao DATE  )  BEGIN  DECLARE v\_mensagem TEXT;  SET v\_mensagem = CONCAT('Novo aviso: ', p\_titulo);  INSERT INTO Notificacoes (usuario\_id, tipo\_notificacao, mensagem, data\_notificacao, enviada, aviso\_id)  SELECT  u.usuario\_id,  'aviso',  v\_mensagem,  p\_data\_publicacao,  FALSE,  p\_aviso\_id  FROM Usuarios u  WHERE u.tipo = 'estudante';  END //  DELIMITER ;  DELIMITER //  CREATE PROCEDURE InserirNotificacaoLembrete()  BEGIN  DECLARE v\_data\_amanha DATE;  SET v\_data\_amanha = DATE\_ADD(CURDATE(), INTERVAL 1 DAY);  INSERT INTO Notificacoes (usuario\_id, tipo\_notificacao, mensagem, data\_notificacao, enviada, tarefa\_evento\_id)  SELECT  te.usuario\_id,  'lembrete',  CONCAT('Lembrete: ', te.titulo, ' em ', te.data),  CURDATE(),  FALSE,  te.tarefa\_evento\_id  FROM Tarefas\_Eventos te  WHERE te.data = v\_data\_amanha;  END //  DELIMITER ;  DELIMITER //  CREATE PROCEDURE LimparNotificacoesAntigas()  BEGIN  DELETE FROM Notificacoes  WHERE enviada = TRUE  AND data\_notificacao < DATE\_SUB(CURDATE(), INTERVAL 30 DAY);  END //  DELIMITER ;  DELIMITER //  CREATE PROCEDURE AtualizarSenhaUsuario(  IN p\_email VARCHAR(100),  IN p\_nova\_senha VARCHAR(255)  )  BEGIN  UPDATE Usuarios  SET senha = p\_nova\_senha  WHERE email = p\_email;  END //  DELIMITER ;  DELIMITER //  CREATE PROCEDURE CadastrarTarefaEvento(  IN p\_usuario\_id INT,  IN p\_titulo VARCHAR(100),  IN p\_descricao TEXT,  IN p\_data DATE,  IN p\_tipo ENUM('tarefa', 'evento')  )  BEGIN  DECLARE v\_tarefa\_evento\_id INT;  DECLARE v\_data\_amanha DATE;  -- Inserir a tarefa/evento  INSERT INTO Tarefas\_Eventos (usuario\_id, titulo, descricao, data, tipo)  VALUES (p\_usuario\_id, p\_titulo, p\_descricao, p\_data, p\_tipo);  -- Obter o ID da tarefa/evento inserida  SET v\_tarefa\_evento\_id = LAST\_INSERT\_ID();  -- Verificar se a data é amanhã para criar um lembrete  SET v\_data\_amanha = DATE\_ADD(CURDATE(), INTERVAL 1 DAY);  IF p\_data = v\_data\_amanha THEN  INSERT INTO Notificacoes (usuario\_id, tipo\_notificacao, mensagem, data\_notificacao, enviada, tarefa\_evento\_id)  VALUES (  p\_usuario\_id,  'lembrete',  CONCAT('Lembrete: ', p\_titulo, ' em ', p\_data),  CURDATE(),  FALSE,  v\_tarefa\_evento\_id  );  END IF;  END //  DELIMITER ;  DELIMITER //  CREATE PROCEDURE RegistrarInteracaoAssistente(  IN p\_usuario\_id INT,  IN p\_pergunta TEXT,  IN p\_resposta TEXT  )  BEGIN  INSERT INTO Interacoes\_Assistente (usuario\_id, pergunta, resposta)  VALUES (p\_usuario\_id, p\_pergunta, p\_resposta);  END //  DELIMITER ;  DELIMITER //  CREATE PROCEDURE MarcarNotificacaoEnviada(  IN p\_notificacao\_id INT  )  BEGIN  UPDATE Notificacoes  SET enviada = TRUE  WHERE notificacao\_id = p\_notificacao\_id;  END //  DELIMITER ;  DELIMITER //  CREATE PROCEDURE ExcluirUsuario(  IN p\_usuario\_id INT  )  BEGIN  DELETE FROM Usuarios  WHERE usuario\_id = p\_usuario\_id;  END //  DELIMITER ; |

# ARTEFATO 7: Diagrama de Classes

O Diagrama de Classes da UML representa a estrutura e relações entre as classes de um produto de software orientado a objetos.

**Importante**:

* Esta seção é opcional apenas se o produto de software for orientado a objetos.
* A equipe deve combinar com a banca de professores todos os artefatos de especificação.

A figura a seguir apresenta uma instância exemplo, como padrão para entrega.

| **ARTEFATO 7:** Diagrama de Classes |
| --- |
| Diagrama  Descrição gerada automaticamente |

**Figura 7 – Exemplo: Diagrama de Classes.**

# ARTEFATO 8: Diagrama de Atividades

Incluir Diagrama de Atividades (da disciplina de Criação de Modelos Computacionais)

| **ARTEFATO 8.1:** Diagrama de Atividades |
| --- |
|  |

**Figura 8 – Exemplo: Diagrama de Atividades.**

# REFERÊNCIA BIBLIOGRÁFICAS

AGUIAR, F. **Product backlog building: concepção de um product backlog efetivo**. 2018. Disponível em: <https://speakerdeck.com/fabiogr/product-backlog-building>. Acesso em: 10 fevereiro 2022.

AGUIAR, F. **PBB\_Canvas Template**. 2018. Disponível em: <http://www.productbacklogbuilding.com/canvas/PBB_Canvas.pdf>. Acesso em 10 de fevereiro de 2022.

SCHWABER, K.; SUTHERLAND, J. **Guia do SCRUM - o guia definitivo para o Scrum: as regras do jogo**. 2020. Disponível em: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Portuguese-European.pdf>. Acesso em: 10 fevereiro 2022.