



Plataforma de Estudo para Auxílio no Aprendizado de Banco de Dados

João Pedro Valladão Pinheiro

Projeto Final de Graduação

**Centro Técnico Científico
Departamento de Informática
Curso de Engenharia de Computação**

Orientador: Prof. Sérgio Lifschitz

Rio de Janeiro
Janeiro de 2014



João Pedro Valladão Pinheiro

**Plataforma de Estudo para Auxílio no
Aprendizado de Banco de Dados**

Relatório de Projeto Final, apresentado ao programa do Curso de Engenharia de Computação da PUC-Rio como requisito parcial para a obtenção do título de Engenheiro de Computação.

Prof. Sérgio Lifschitz

Orientador

Departamento de Informática — PUC-Rio

Rio de Janeiro, 16 de Janeiro de 2014

Agradecimentos

Agradeço ao apoio incondicional da minha família em todos os momentos da minha vida.

Também sou muito grato ao meu orientador Sérgio Lifschitz por ter me auxiliado em diversos momentos da minha graduação e, principalmente, no Projeto Final.

Faço um agradecimento especial a Vanessa Wagner, que acreditou nos meus sonhos e foi imprescindível para a realização dos mesmos.

Gostaria de destacar e agradecer três amigos que foram essenciais no período de elaboração desse projeto: Daniel Blando, João Leite e Jesué Junior.

Obrigado a todos.

Resumo

João Pedro, Pinheiro; Lifschitz, Sérgio. **Plataforma de Estudo para Auxílio no Aprendizado de Banco de Dados**. Rio de Janeiro, 2014. ??p. Relatório de Projeto Final — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Elaboração de uma plataforma web que auxilie o aprendizado do aluno da disciplina Banco de Dados. O tema surgiu a partir do conceito flipped classroom, que se trata de um modelo que sugere o aprendizado online. O aluno assiste, pratica e discute determinados assuntos em casa, trazendo as dúvidas para a sala de aula. Dessa forma, as aulas tornam-se mais dinâmicas e menos expositivas.

Palavras-chave

Plataforma de Estudo. Banco de Dados. SQL.

Abstract

João Pedro, Pinheiro; Lifschitz, Sérgio. **Study Platform to Aid Database Learning**. Rio de Janeiro, 2014. ??p. PhD Thesis — Department of Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Elaboration of web platform which aids student learning of database discipline. The topic came up from flipped classroom concept, whose model suggests online learning. The student attends, practice and discusses certain subjects at home, bringing doubts to classroom. Thus, classes became more dynamic and less expository.

Keywords

Study Platform. Database. SQL.

Sumário

| | | |
|----------|---|-----------|
| 1 | Introdução | 9 |
| 2 | Proposta | 11 |
| 2.1 | Análise dos Projetos Similares | 12 |
| 2.2 | Avaliação do Sistema perante os Concorrentes – Análise SWOT | 13 |
| 2.3 | Requisitos | 15 |
| 3 | Projeto | 17 |
| 3.1 | Modelagem Conceitual | 18 |
| 3.2 | Modelagem Lógica | 23 |
| 3.3 | Modelagem Física | 28 |
| 3.4 | Casos de Uso | 36 |
| 3.5 | Diagrama de Classes | 41 |
| 4 | Plataforma Web | 45 |
| 4.1 | Mockups de Telas | 45 |
| 4.2 | Framework Django | 50 |
| 4.3 | Screenshots do Sistema | 52 |
| 5 | Considerações Finais | 55 |
| | Referências Bibliográficas | 57 |

Lista de figuras

| | | |
|------|---|----|
| 1.1 | Inversão nos padrões de ensino sugerida pelo modelo flipped class-room; Retirada do infográfico contido em (FlippedLearning) | 10 |
| 2.1 | Ilustração que demonstra todos os critérios utilizados na Análise SWOT | 14 |
| 3.1 | Fases do projeto e da implementação de banco de dados sugerida no livro ' Sistemas de Banco de Dados ', 4a edição, Elsmari e Navathe | 18 |
| 3.2 | Primeira versão da Modelagem Conceitual | 19 |
| 3.3 | Segunda versão da Modelagem Conceitual; solução focada no fórum | 20 |
| 3.4 | Segunda versão da Modelagem Conceitual; solução focada nos exercícios | 21 |
| 3.5 | Terceira versão da Modelagem Conceitual; solução focada no fórum | 22 |
| 3.6 | Terceira versão da Modelagem Conceitual; solução focada nos exercícios | 22 |
| 3.7 | Diagrama de Classes do pacote Templates | 42 |
| 3.8 | Diagrama de Classes do pacote Views | 42 |
| 3.9 | Diagrama de Classes do pacote Models | 44 |
| 4.1 | Página para validação do usuário | 46 |
| 4.2 | Página inicial do sistema | 46 |
| 4.3 | Página que mostra o desempenho dos alunos | 47 |
| 4.4 | Página de busca e inserção de dúvidas no fórum | 48 |
| 4.5 | Página que lista os exercícios cadastrados | 49 |
| 4.6 | Página com as questões que compõem a lista selecionada | 50 |
| 4.7 | Arquitetura do Framework Django | 51 |
| 4.8 | Página para validação do usuário | 52 |
| 4.9 | Página inicial do sistema | 53 |
| 4.10 | Página de busca e inserção de dúvidas no fórum | 53 |
| 4.11 | Página que lista os exercícios cadastrados | 54 |
| 4.12 | Página com as questões que compõe a lista selecionada | 54 |

Lista de tabelas

| | | |
|------|--|----|
| 3.1 | Tabelas referentes a Lista de Exercícios (parte 1) | 24 |
| 3.2 | Tabelas referentes a Lista de Exercícios (parte 2) | 25 |
| 3.3 | Tabelas referentes ao Fórum (parte 1) | 26 |
| 3.4 | Tabelas referentes ao Fórum (parte 2) | 27 |
| 3.5 | Criação das Tabelas (parte 1) | 29 |
| 3.6 | Criação das Tabelas (parte 2) | 30 |
| 3.7 | Adição dos Comentários (parte 1) | 31 |
| 3.8 | Adição dos Comentários (parte 2) | 32 |
| 3.9 | Adição dos Comentários (parte 3) | 33 |
| 3.10 | Adição dos Comentários (parte 4) | 34 |
| 3.11 | Adição das Restrições (parte 1) | 35 |
| 3.12 | Adição das Restrições (parte 2) | 36 |

1

Introdução

Através do advento e popularização da Internet, pessoas envolvidas com educação encontraram uma possível solução para o déficit de ensino global. A diminuição das distâncias entre educadores e alunos gerou uma gama de possíveis serviços a serem prestados em prol de um melhor ensino.

A partir disso, algumas iniciativas foram desenvolvidas e estão inseridas com sucesso no mercado. Foram contempladas desde ideias sem fins lucrativos, como **Edx** (Edx), **Khanacademy** (Khanacademy) e **Gradiance** (Gradiance) (essa última idealizada pelo Prof. Jeffrey D. Ullman), até startups que oferecem parte dos cursos gratuitamente e outra parte paga, como o caso da **Code School** (CodeSchool).

Essa metodologia de ensino recebeu um nome: **flipped classroom** (FlippedLearning). Trata-se de uma inversão nos padrões de ensino adotados pelas escolas, no qual a passagem de conhecimento passa a ser online, através de vídeos de curta duração que devem ser assistidos pelos alunos fora da sala de aula. Dessa forma, a sala de aula se torna um ambiente de resolução de dúvidas e exercícios.

Baseado nesse modelo, foi proposta a elaboração de uma plataforma web que contempla alguns dos princípios sugeridos pela metodologia flipped classroom. Vale ressaltar que houve uma descontinuação de tema e troca de orientador entre os projetos finais I e II. Logo, todo o trabalho proposto e apresentado é referente ao desenvolvimento de apenas um semestre.

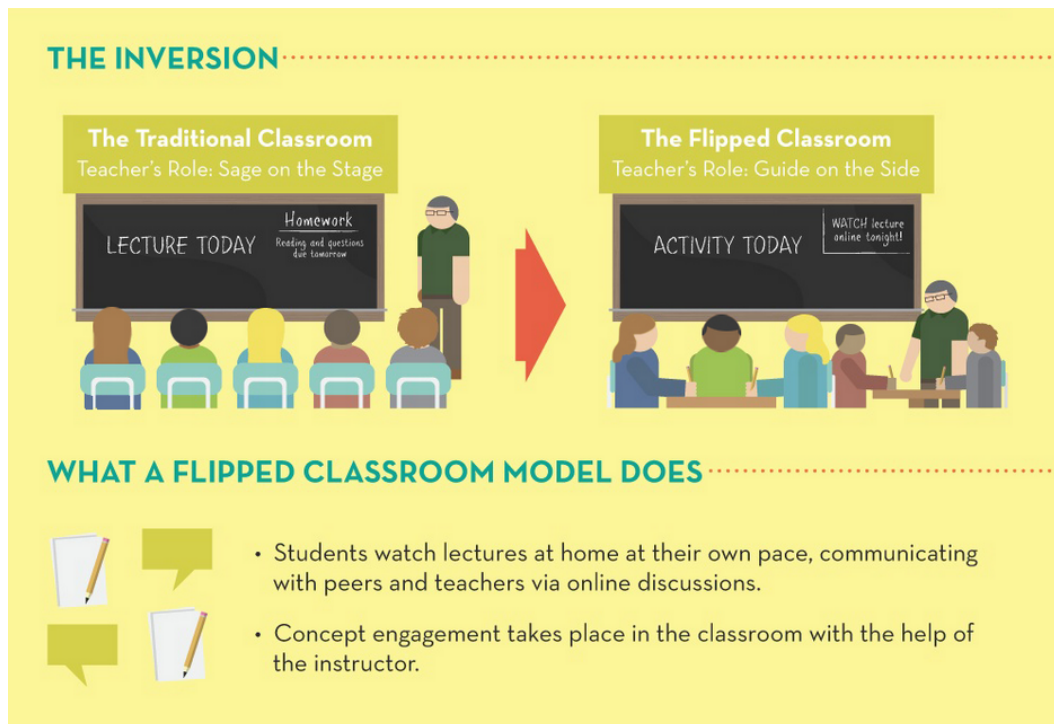


Figura 1.1: Inversão nos padrões de ensino sugerida pelo modelo flipped classroom; Retirada do infográfico contido em (FlippedLearning)

2

Proposta

O presente trabalho tem como objetivo desenvolver uma ferramenta, de cunho acadêmico e open-source, que auxilie o ensino de banco de dados. O escopo do projeto abrangerá o ensino da linguagem SQL, através da disponibilização de exercícios online e fórum para discussão dos mesmos ou de assuntos referentes à disciplina. Além disso, todas as listas de exercícios contidas no site serão, em um primeiro momento, as mesmas oferecidas na disciplina de Banco de Dados da PUC-Rio (INF 1383).

Atualmente, os alunos da disciplina podem resolver os exercícios propostos em algumas aulas práticas, durante horário de aula, ou utilizar um programa desktop que se conecta ao servidor da disciplina com o **SGBD** (ElmasriNavathe05) já configurado. Os exercícios estão vinculados, na maioria das vezes, a um ou mais esquemas contidos no SGBD em questão. Dessa forma, os alunos acessam um mesmo esquema para tentar solucionar determinada questão. Quando trata-se de uma lista de exercícios que envolve puramente o comando **SELECT**, não há maiores problemas quanto ao esquema compartilhado. Porém, quando há comandos **DML**(ElmasriNavathe05) do tipo **INSERT**, **UPDATE** ou **DELETE**, ou comandos **DDL**(ElmasriNavathe05) envolvidos na resolução da lista, pode-se gerar uma indisponibilização momentânea, caso um comando seja executado de forma incorreta.

Sendo assim, duas das principais características do sistema proposto serão a utilização de esquemas distintos por alunos a cada questão e a possibilidade de voltar em uma determinada modificação feita anteriormente no esquema. Logo, além de ter liberdade de executar qualquer comando na base, sabendo que não interferirá os demais alunos, o aluno poderá voltar para um estado anterior da base quando bem entender.

2.1

Análise dos Projetos Similares

2.1.1

Edx

A ideia do **Edx** é expandir o acesso à educação para todos, disponibilizando ensino e aprendizado do campus das respectivas faculdades participantes do projeto no meio online.

O **Edx** tem sua sede em Cambridge, Massachusetts e é comandado pela parceria das faculdades MIT e Harvard. Além de disponibilizar cursos das supracitadas universidades, outras faculdades americanas bem conceituadas resolveram participar do projeto, cedendo vídeos de aulas dadas e exercícios praticados online.

Vale ressaltar que a parceria não possui fins lucrativos.

2.1.2

Khanacademy

Esta plataforma de ensino foi iniciada por Salman Amin "Sal" Khan (SalmanKhan), que, inicialmente, produzia vídeos que contemplavam um amplo espectro de assuntos acadêmicos, com ênfase em matemática e ciências. O objetivo inicial era auxiliar o aprendizado de uma prima, que morava distante de sua residência.

Os vídeos postados no YouTube obtiveram milhares de visualizações em poucos anos. Após o reconhecimento obtido, Khan resolveu fundar a **KhanAcademy**. Trata-se também de uma plataforma web sem fins lucrativos e a abordagem adotada está de acordo com o conceito de flipped learning.

2.1.3

Gradiance

O **Gradiance** trata-se de um software web pago com o objetivo de oferecer serviços de aprendizado online focado em alguns temas de computação. Os temas abordados são: Sistemas de Banco de Dados, Compiladores, Teoria de Autômatos e Sistemas Operacionais.

Esse sistema é oferecido em algumas faculdades americanas e tem como idealizador o Prof. Jeffrey D. Ullman (Ullman), que possui livros e artigos publicados abrangendo os temas mencionados anteriormente.

2.1.4

Code School

O **Code School** é uma empresa startup que desenvolveu uma plataforma web com o objetivo de auxiliar o aprendizado de linguagens de programação. A empresa possui fins lucrativos e o interessado paga uma mensalidade para ter acesso a um curso específico, ou uma tarifa maior para abranger todos os cursos.

A empresa acredita que a melhor forma de aprendizado ocorre na prática. Sendo assim, seu objetivo é transferir conhecimento através da elaboração de diversos desafios intrigantes para o usuário.

2.2

Avaliação do Sistema perante os Concorrentes – Análise SWOT

A **Análise SWOT** (Fine09) é um sistema simples para posicionar ou verificar a posição estratégica da empresa no ambiente em questão. O termo SWOT é uma sigla oriunda dos termos ingleses Strengths (Forças), Weaknesses (Fraquezas), Opportunities (Oportunidades) e Threats (Ameaças).

- Strengths (forças) - vantagens internas da empresa em relação às concorrentes. Ex.: qualidade do produto oferecido, bom serviço prestado ao cliente, solidez financeira, etc.

- Weaknesses (fraquezas) - desvantagens internas da empresa em relação às concorrentes. Ex.: altos custos de produção, má imagem, instalações inadequadas, marca fraca, etc.;

- Opportunities (oportunidades) – aspectos externos positivos que podem potencializar a vantagem competitiva da empresa. Ex.: mudanças nos gostos dos clientes, falência de empresa concorrente, etc.;

- Threats (ameaças) - aspectos externos negativos que podem por em risco a vantagem competitiva da empresa. Ex.: novos competidores, perda de trabalhadores fundamentais, etc.

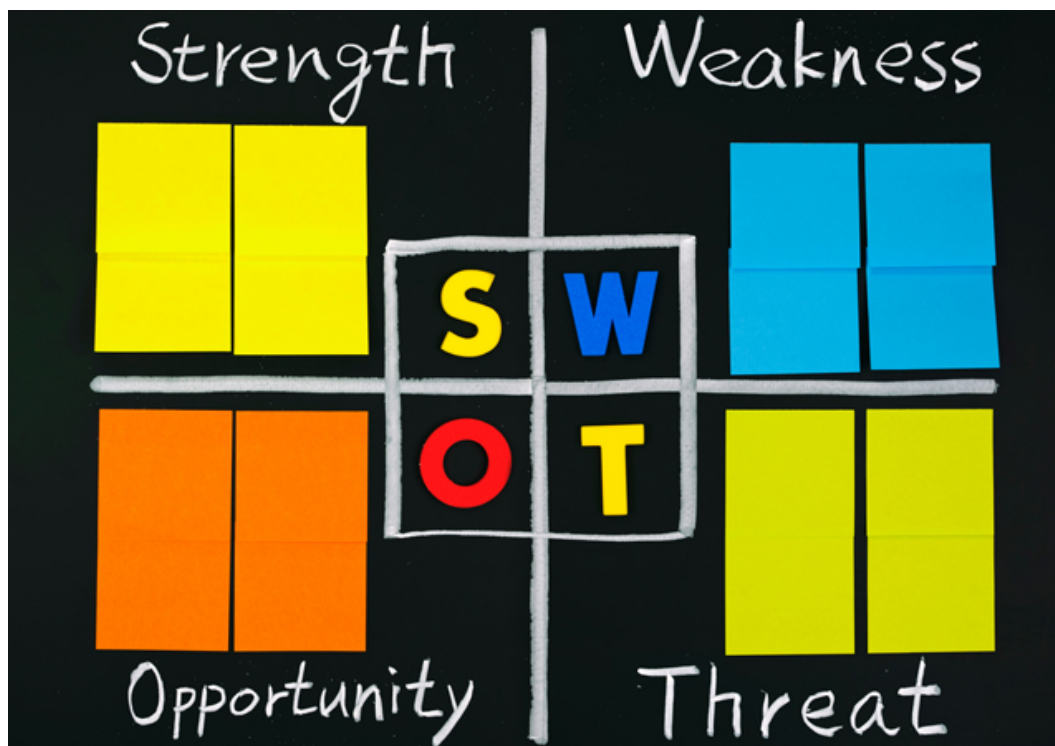


Figura 2.1: Ilustração que demonstra todos os critérios utilizados na Análise SWOT

Segue a seguir a análise do projeto perante os concorrentes:

| |
|--|
| Forças |
| Sistema focado no aprendizado de apenas uma disciplina |
| Ferramenta opensource e gratuita |
| Apenas o Gradiance possui curso específico para aprendizado de Banco de Dados |
| Listas de exercícios extensa e aplicada por anos com sucesso na disciplina de Banco de Dados |
| Exercícios com feedback instantâneo aos alunos |
| Fraquezas |
| Publico alvo restrito a alunos da PUC-Rio |
| Novo entrante no mercado |
| Não possui reconhecimento internacional como cursos do Edx |
| Oportunidades |
| Ferramenta facilmente acessível pelos alunos e professores |
| Apenas o Gradiance possui curso específico para aprendizado de Banco de Dados |
| Maior independência no aprendizado do aluno (aprender fora da sala de aula) |
| Gerar maior interesse nos alunos, pois aprende-se fazendo (aprendizado na prática) |
| Conceito de Flipped Learning ainda pouco desenvolvido no Brasil |
| Ameaças |
| Desinteresse dos alunos pelo aprendizado de Banco de Dados |
| Desinteresse dos alunos pela utilização da ferramenta |

2.3

Requisitos

2.3.1

Escopo

A lista a seguir se refere à elicitação de requisitos (Leite03) referentes a uma plataforma de ensino online. A mesma conterá tanto **requisitos funcionais** quanto **requisitos não funcionais**.

2.3.2

Público Alvo

Em um primeiro momento, a plataforma em questão se destina a estudantes da Pontifícia Universidade Católica do Rio de Janeiro que estejam cursando a disciplina de Banco de Dados. Em um futuro próximo, deseja-se disponibilizar o projeto a diversos cursos.

2.3.3

Requisitos Não Funcionais

[R1] Realização de um estudo sobre os principais concorrentes da plataforma, utilizando a Análise SWOT.

[R2] Documentação do sistema utilizando práticas conhecidas: Modelagem Conceitual, Modelagem Lógica, Modelagem Física, Casos de Uso e Diagrama de Classes.

[R3] Utilização de documentação interna do código, com o intuito de facilitar a modificação da plataforma no futuro.

[R4] Gerenciamento de versões do código utilizando a ferramenta Git com a plataforma GitHub.

[R5] Criação de Mockups das telas do sistema.

2.3.4

Requisitos Funcionais

[R6] A plataforma de ensino terá a funcionalidade de controle de acesso, existindo dois níveis de usuário: Aluno e Professor.

Funcionalidades Professor:

[R7] Controle dos alunos que possuem acesso à ferramenta.

[R8] Módulo de edição dos exercícios e questões.

[R9] Carga dos esquemas utilizados nos exercícios.

[R10] Monitoramento da evolução dos alunos.

[R11] Permissão para cancelar uma thread inadequada no fórum.

Funcionalidades Aluno:

[R12] Utilização de uma instância própria a cada exercício.

[R13] Possibilidade de compartilhar com seus colegas de turma suas instâncias.

[R14] Utilização de fórum como forma de sanar dúvidas dos exercícios e auxiliar os outros alunos.

[R15] Resolução dos exercícios com correção dos mesmos em tempo real.

[R16] Possibilidade de fazer upload de foto para facilitar a identificação do aluno.

[R17] Possibilidade de compartilhar as questões em redes sociais (Facebook, Google+ e Twitter).

3

Projeto

Neste tópico serão expostas todas as abstrações de projeto realizadas em ordem cronológica. Essa evolução sequencial permitiu um melhor entendimento do problema e, conseqüentemente, uma clara definição de escopo.

Iniciando pela Modelagem Conceitual [3.1], com a utilização do modelo entidade relacionamento, foi possível obter uma alta abstração da plataforma, assim como um melhor entendimento das regras de negócios envolvidas (Heuser09). Logo após, foram feitas as modelagens lógica [3.2] e física [3.3]. Na lógica, explicitamos a estrutura a ser utilizada para persistência dos dados, que no caso foi em tabelas, e os tipos de dados de cada atributo. Já na física, foi definido o SGBD utilizado, que no caso foi o **PostgreSQL 9.1**, e definidas os comandos SQL do tipo DDL para criação das tabelas, chaves primárias e estrangeiras, triggers, constraints, entre outros.

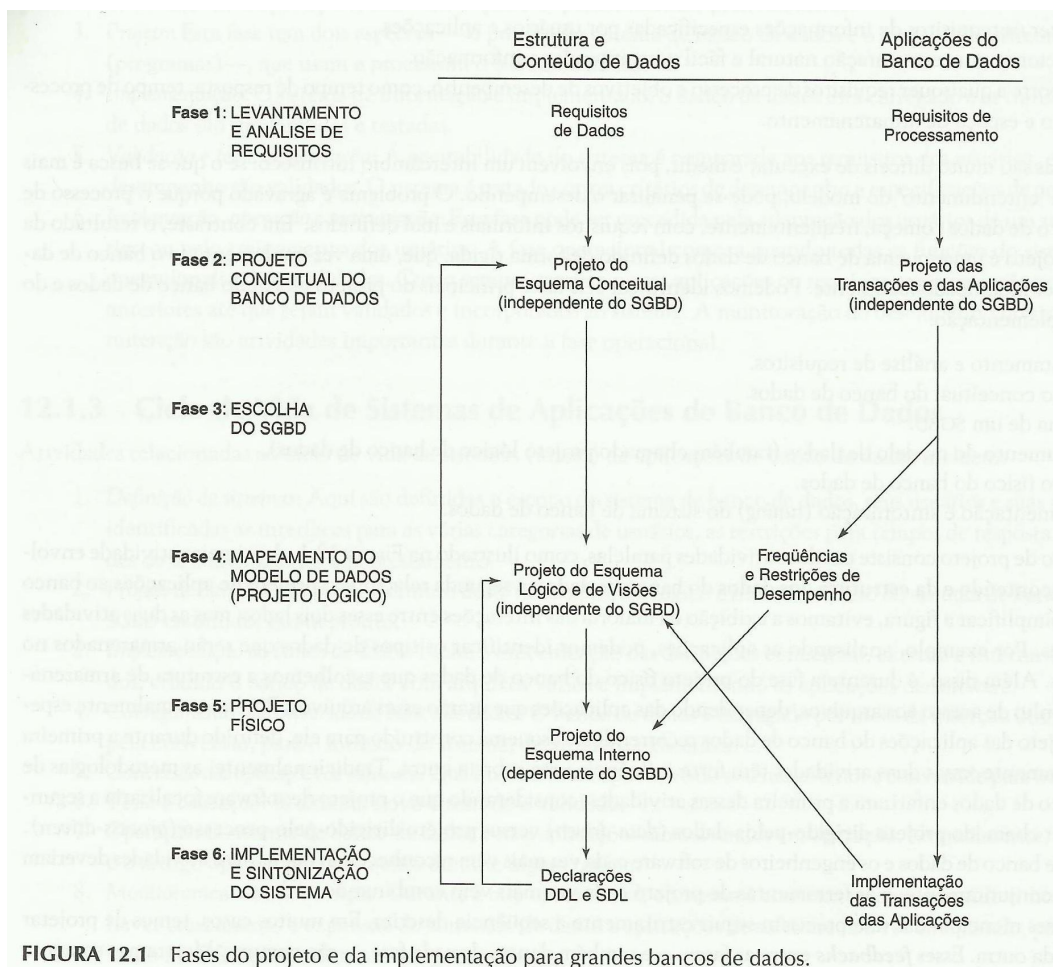


Figura 3.1: Fases do projeto e da implementação de banco de dados sugerida no livro **'Sistemas de Banco de Dados'**, 4a edição, Elsmari e Navathe

Após terminadas as abstrações referentes ao armazenamento dos dados, partimos para a definição da interação entre os usuários envolvidos e o sistema, através dos Casos de Uso [3.4] (Larman04), e para a definição da arquitetura a ser utilizada no desenvolvimento da plataforma, através do Diagrama de Classes [3.5].

3.1 Modelagem Conceitual

Nesta seção, vamos mostrar a evolução da concepção do projeto através das três versões elaboradas com o uso do diagrama entidade relacionamento.

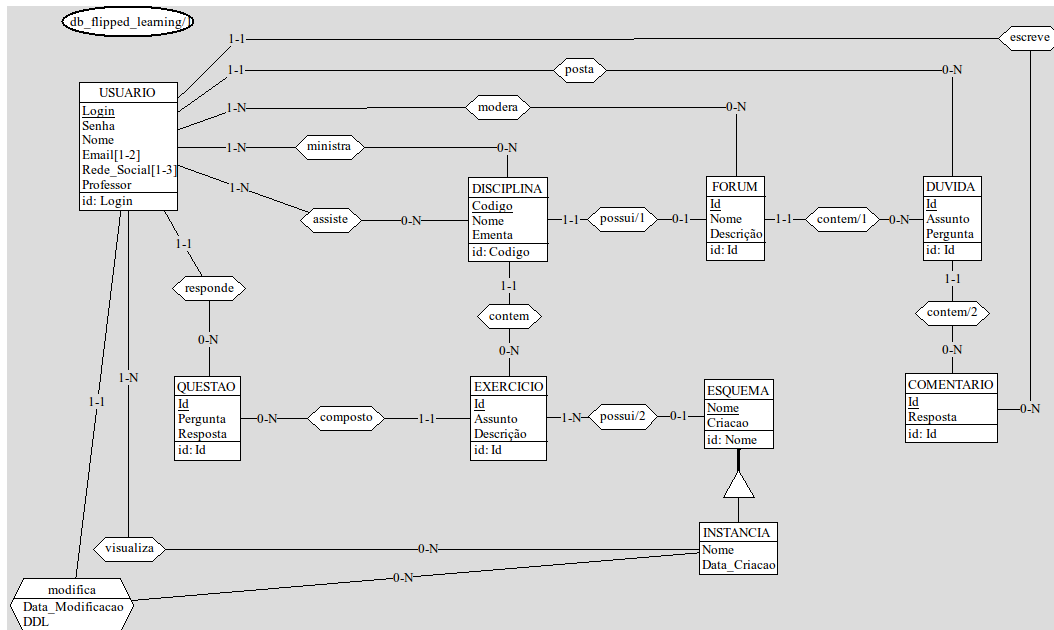


Figura 3.2: Primeira versão da Modelagem Conceitual

A entidade usuário serve como representação tanto para aluno, quanto para professor. Como não foram encontrados atributos únicos que fossem capazes de diferenciar as possíveis entidades aluno e professor, não houve necessidade da utilização de herança (também chamado de “IS-A”). No caso, haveria a possibilidade de aluno e professor herdarem características da entidade usuário. Para a diferenciação do papel, aluno ou professor, foi utilizado o atributo ‘professor’ (depois será possível perceber que o mesmo deixou de ser utilizado, por não se fazer necessária a distinção de papéis nesse nível de abstração do modelo).

Cada exercício proposto poderá, ou não, ser vinculado, a um esquema. Dessa forma, o sistema foi idealizado para permitir não só questões relacionadas a SQL, mas a qualquer tema abordado na disciplina Banco de Dados. Como características, a entidade esquema possui um nome, como identificador, e um campo destinado a seu DDL de criação, chamado criação. Quando um aluno inicia um novo exercício e caso o mesmo seja vinculado a um esquema para ser respondido, é criada uma instância. A instância nada mais é do que uma cópia, em seu estado inicial, do esquema associado ao exercício. Essa escolha foi tomada a fim de dar maior liberdade na utilização da base por parte dos alunos. É comum, em outras plataformas, a existência de bases compartilhadas para resolução dos exercícios em SQL. Porém, acredita-se que essa prática inibe o aluno (como mencionado em seção anterior), que passa a ter receio de errar e prejudicar a turma.

Ainda em relação à instância, foi dada a liberdade para que o aluno possa compartilhá-la com seus colegas de turma. O mesmo fica explícito ao analisarmos a cardinalidade do relacionamento “visualiza”, entre as entidades usuário e instância. Já em relação à ação de “modificação” da instância, apenas o usuário dono da mesma poderá realizá-la. Para haver maior interação dos usuários com a plataforma, foi possibilitada a criação uma área de fórum por disciplina. E o usuário com o papel de professor seria o moderador; já os alunos participariam com dúvidas e comentários relacionados à matéria dada.

A característica preponderante desta versão era fazer uma plataforma que englobasse diversas disciplinas. Porém, conforme foi sendo definido o escopo, pôde-se observar que um fator de diferenciação perante os demais sistemas seria o foco no tema banco de dados. Essa escolha provocou uma melhor visualização da solução, que pode ser percebida nas duas seguintes versões.

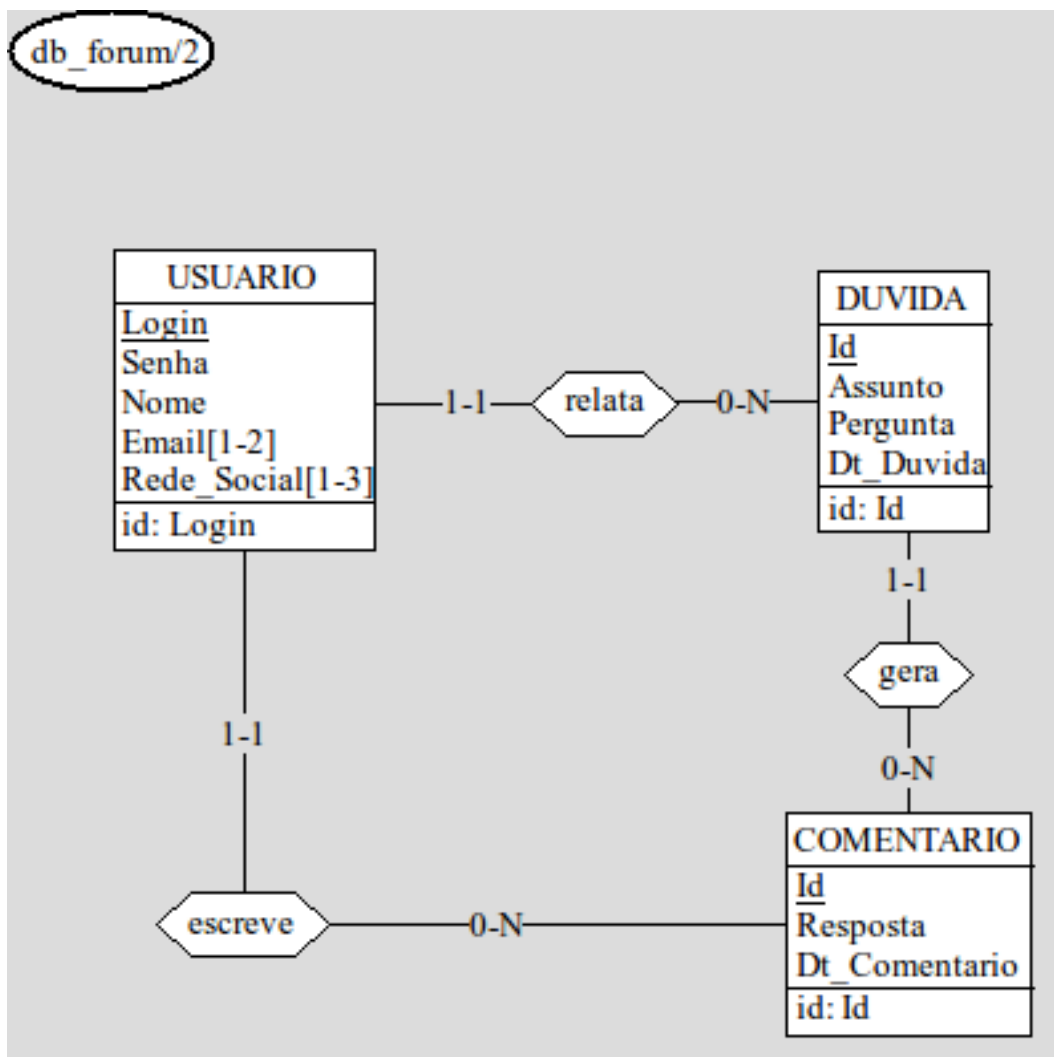


Figura 3.3: Segunda versão da Modelagem Conceitual; solução focada no fórum

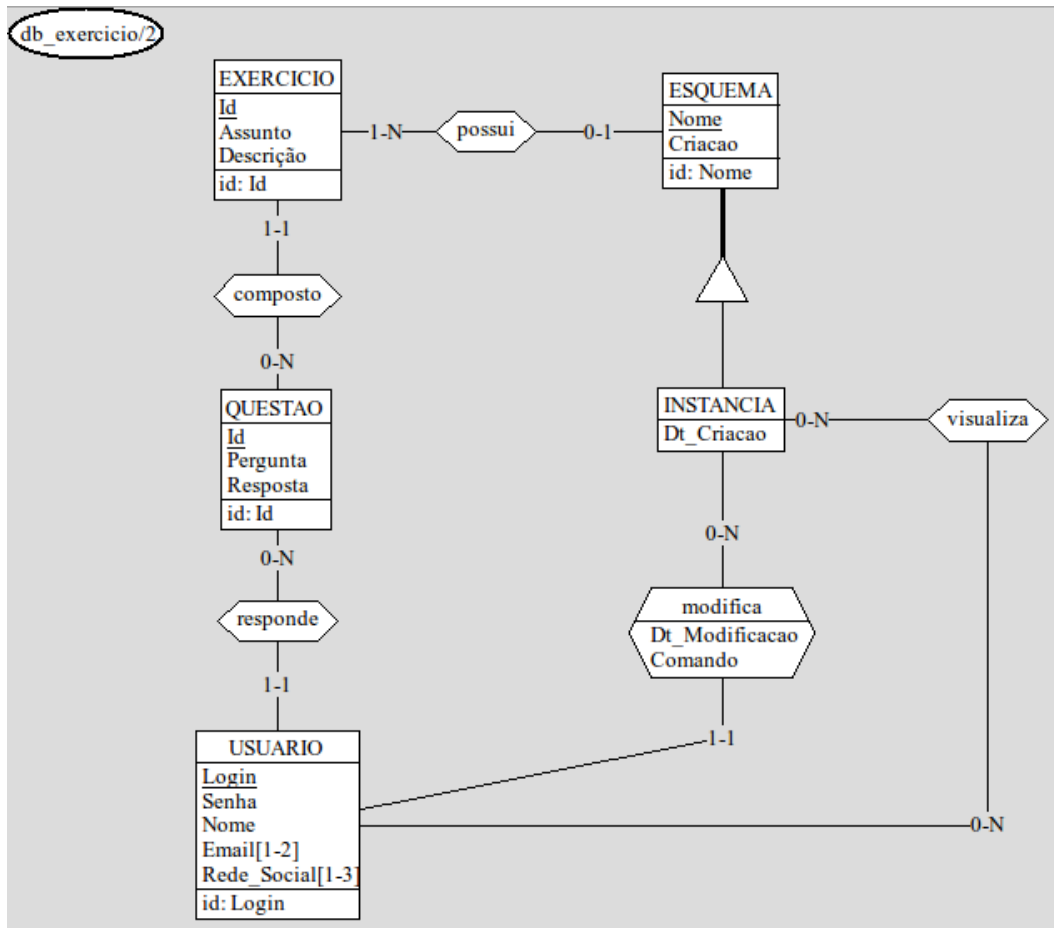


Figura 3.4: Segunda versão da Modelagem Conceitual; solução focada nos exercícios

Tendo em vista que a primeira versão havia ficado muito confusa, foi decidido particionar a solução em dois sistemas distintos: fórum e exercício. Dessa forma, houve uma simplificação do problema, além da melhor compreensão do mesmo, que certamente ajudará nas próximas etapas do projeto.

Alguns aspectos relevantes precisam ser ressaltados. A modelagem referente ao fórum apresenta um caminho fechado, o que deve ser evitado. Porém, um caminho fechado não necessariamente é um ciclo. Já um ciclo é, necessariamente, um caminho fechado. Assim sendo, devemos observar o sentido do relacionamento. E como um usuário, no papel de aluno, relata a dúvida e um usuário, no papel de aluno ou professor, escreve o comentário, podemos perceber que não há geração de ciclo.

Mesmo com a evolução entre as versões, alguns requisitos do sistema não foram contemplados, tais como os moderadores do fórum e a persistência das respostas dos alunos às questões dos exercícios. Além disso uma dúvida se fez presente: será mesmo que uma instância é um esquema?

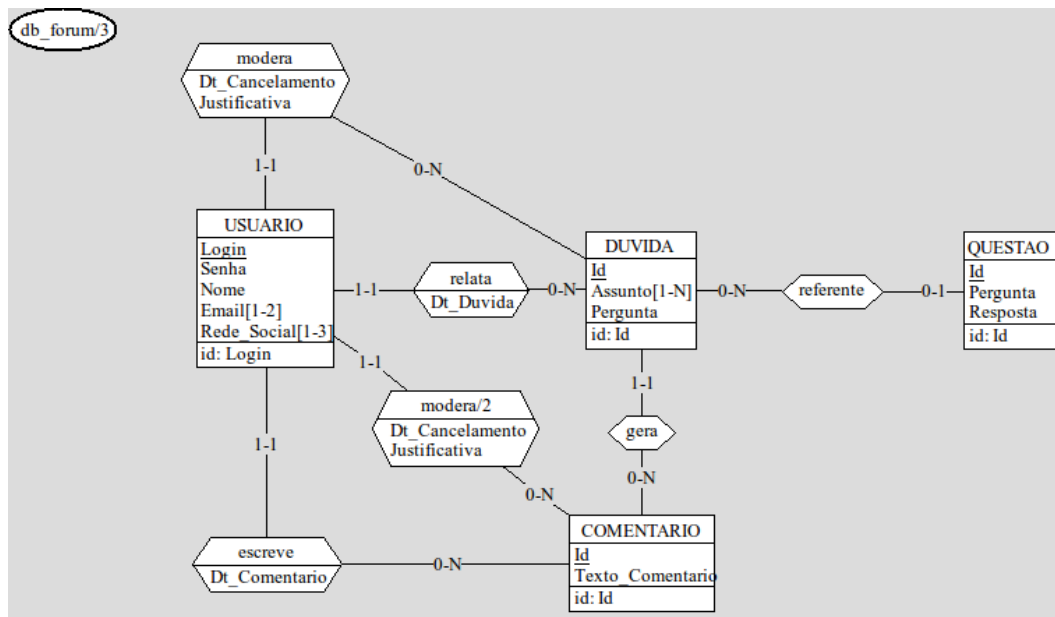


Figura 3.5: Terceira versão da Modelagem Conceitual; solução focada no fórum

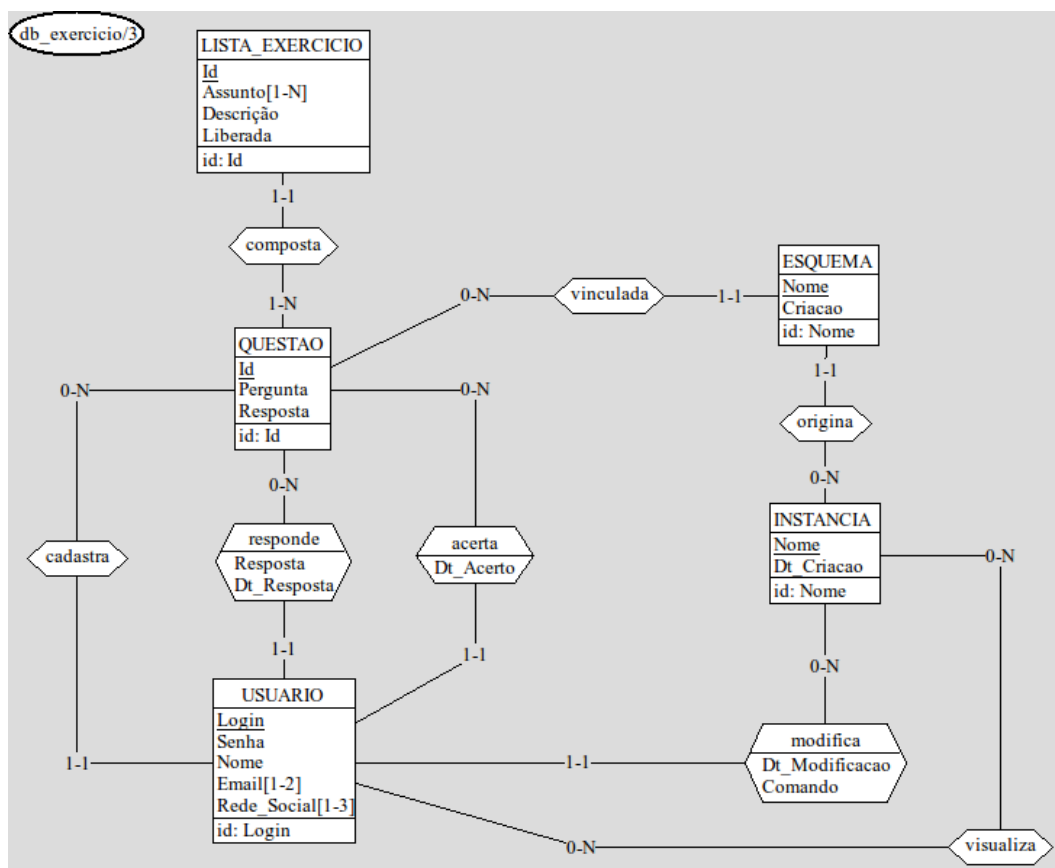


Figura 3.6: Terceira versão da Modelagem Conceitual; solução focada nos exercícios

Com relação à modelagem referente ao fórum, podemos perceber a adição das datas no relacionamento “relata” e “escreve”, ao invés de estarem presentes nas entidades dúvida e comentário, respectivamente. Também foram adicionadas as ações de “moderação” do fórum, que anteriormente não foram contempladas. Em ambas um campo de ‘justificativa’ foi inserido, para permitir que o moderador possa explicar os motivos de um possível cancelamento de dúvida ou comentário.

Além disso, percebe-se que a entidade dúvida agora pode, ou não, estar relacionada com a entidade questão. Dessa maneira, foi possibilitada a inserção de dúvidas não somente aos exercícios contidos na plataforma, mas como dúvidas teóricas, que tenham sido abordadas na disciplina, ou exercícios de fontes externas, por exemplo de outras faculdades.

Com relação à modelagem referente aos exercícios, as entidades esquema e instância deixaram de ser, respectivamente, superclasse e subclasse uma da outra. Logo, houve uma substituição da representação de herança por uma de relacionamento, com a ação “originar”.

Outro ponto importante é a persistência das respostas dos usuários às questões. Serão armazenadas tanto as respostas corretas como as incorretas. Houve a opção de representar a ação de “acerto” separado da ação de “resposta”. Logo, deve-se tomar cuidado para manter a restrição de integridade semântica nesse caso, já que devemos identificar quais das respostas do usuário são corretas.

Percebe-se que o relacionamento da ação “vínculo” era entre as entidades exercício e esquema. Todavia, uma lista de exercícios pode conter questões referentes a diferentes esquemas. Assim sendo, a entidade questão passa a estar vinculada a um determinado esquema.

3.2

Modelagem Lógica

Nesta fase, o mapeamento ainda não considera nenhuma característica específica ou casos especiais que se aplicam à implementação do modelo de dados do SGBD. Ou seja, tudo o que for explicado pode ser aplicado a qualquer escolha de SGBD na próxima etapa.

Foram listadas todas as tabelas com suas chaves primárias e estrangeiras identificadas, atributos e tipos de dados. Além disso, foram identificados os campos: Entidade, Atributo e Relacionamento. Com esses campos, foi possível realizar o rastro entre a Modelagem Conceitual e a Modelagem Lógica.

| lista_exercicio | | | | | | | |
|-------------------------|----------------|----------------------|-----------------|------------------|-----------------|----------------|----------------|
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | LISTA.EXERCICIO | Id | - |
| 2 | id_assunto | inteiro | - | - | LISTA.EXERCICIO | Assunto[1-N] | - |
| 3 | descricao | cadeia de caracteres | 1024 | - | LISTA.EXERCICIO | Descrição | - |
| 4 | dt_liberada | data | dd-mm-aaaa | - | LISTA.EXERCICIO | Liberada | - |
| questao | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | QUESTAO | Id | - |
| 2 | id_exercicio | inteiro | - | - | LISTA.EXERCICIO | Id | composta |
| 3 | login_usuario | cadeia de caracteres | 20 | - | USUARIO | Login | cadastra |
| 4 | nome_esquema | cadeia de caracteres | 50 | - | ESQUEMA | Nome | - |
| 5 | pergunta | cadeia de caracteres | 1024 | - | QUESTAO | Pergunta | - |
| 6 | resposta | cadeia de caracteres | 1024 | - | QUESTAO | Resposta | - |
| usuario | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | login | cadeia de caracteres | 20 | - | USUARIO | Login | - |
| 2 | senha | cadeia de caracteres | 15 | - | USUARIO | Senha | - |
| 3 | nome | cadeia de caracteres | 100 | - | USUARIO | Nome | - |
| 4 | e-mail | cadeia de caracteres | 100 | - | USUARIO | Email | - |
| 5 | professor | booleano | 1 | - | - | - | - |
| resposta | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | - | - | - |
| 2 | id_questao | inteiro | - | - | QUESTAO | Id | - |
| 3 | login_usuario | cadeia de caracteres | 20 | - | USUARIO | Login | - |
| 4 | dt_resposta | data | dd-mm-aaaa | - | - | Dt_Resposta | responde |
| 5 | dt_acerto | data | dd-mm-aaaa | - | - | Dt_Acerto | acerta |
| 6 | resposta | cadeia de caracteres | 1024 | - | - | Resposta | responde |
| esquema | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | nome | cadeia de caracteres | 50 | - | ESQUEMA | Nome | - |
| 2 | criacao | cadeia de caracteres | 4096 | - | ESQUEMA | Criacao | - |
| instancia | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | nome | cadeia de caracteres | 50 | - | INSTANCIA | Nome | - |
| 2 | nome_esquema | cadeia de caracteres | 50 | - | ESQUEMA | Nome | - |
| 3 | login_usuario | cadeia de caracteres | 20 | - | USUARIO | Login | - |
| 4 | dt_criacao | data | dd-mm-aaaa | - | INSTANCIA | Dt_Criacao | - |
| modificacao | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | - | - | - |
| 2 | nome_instancia | cadeia de caracteres | 50 | - | INSTANCIA | Nome | - |
| 3 | login_usuario | cadeia de caracteres | 20 | - | USUARIO | Login | - |
| 4 | dt_modificacao | data | dd-mm-aaaa | - | - | Dt_Modificacao | modifica |
| 5 | comando | cadeia de caracteres | 1024 | - | - | Comando | modifica |
| visualizacao | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | nome_instancia | cadeia de caracteres | 50 | - | INSTANCIA | Nome | - |
| 2 | login_usuario | cadeia de caracteres | 20 | - | USUARIO | Login | - |
| assunto | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | LISTA.EXERCICIO | Assunto[1-N] | - |
| 2 | nome | cadeia de caracteres | 100 | - | LISTA.EXERCICIO | Assunto[1-N] | - |
| assunto_lista_exercicio | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id_assunto | inteiro | - | - | LISTA.EXERCICIO | Assunto[1-N] | - |
| 2 | id_exercicio | inteiro | - | - | LISTA.EXERCICIO | Id | - |

Tabela 3.1: Tabelas referentes a Lista de Exercícios (parte 1)

| | |
|-------------------------|---|
| lista_exercicio | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Informa quais temas são abordados na lista de exercício. |
| 3 | Possível sugestões do professor de como resolver a lista, com sugestão de referências bibliográficas. |
| 4 | Armazena em que dia a lista fora liberada aos alunos. |
| questao | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Identifica a qual lista de exercício a questão está atrelada. |
| 3 | Identificação do usuário responsável pela criação da questão. |
| 4 | Identifica a qual esquema a questão pode estar vinculada. |
| 5 | Informa o enunciado da questão a ser respondida pelo aluno. |
| 6 | Trata-se do gabarito dado pelo professor. |
| usuario | |
| ID | Observações |
| 1 | Identificador da tupla, criado pelo usuário e faz parte da sua chave de acesso ao sistema. |
| 2 | Criado pelo usuário e faz parte da sua chave de acesso ao sistema. |
| 3 | Nome completo do usuário. |
| 4 | Email para contato para notificações do fórum, liberação de exercícios e resgate de senha. |
| 5 | Refere-se ao papel do usuário em questão (1 = professor, 0 = aluno) |
| resposta | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Identifica qual questão se refere a resposta dada. |
| 3 | Identifica qual aluno respondeu a questão. |
| 4 | Identifica quando a resposta foi dada. |
| 5 | Identifica quando houve o acerto da resposta. |
| 6 | Armazena qual foi a resposta dada. |
| esquema | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Trata-se do comando DDL para criação do esquema. |
| instancia | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Identifica qual esquema que originou a presente instância. |
| 3 | Identifica qual o aluno responsável pelas modificações da instância. |
| 4 | Identifica quando a instância foi criada pelo aluno. |
| modificacao | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Identifica a instância que está sendo alterada. |
| 3 | Identifica o usuário autor da modificação. |
| 4 | Identifica quando a instância foi alterada. |
| 5 | Trata-se do comando DDL para alteração do esquema. |
| visualizacao | |
| ID | Observações |
| 1 | Identifica a instância que está sendo alterada. |
| 2 | Identifica o usuário autor da modificação. |
| assunto | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Nome do assunto relacionado. |
| assunto_lista_exercicio | |
| ID | Observações |
| 1 | Identifica quais assuntos são abordados na lista de exercícios. |
| 2 | Identificador da lista de exercício. |

Tabela 3.2: Tabelas referentes a Lista de Exercícios (parte 2)

| usuario | | | | | | | |
|---------------------------|------------------|----------------------|-----------------|------------------|-----------------|------------------|----------------|
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | login | cadeia de caracteres | 20 | - | USUARIO | Login | - |
| 2 | e-mail | cadeia de caracteres | 100 | - | USUARIO | Email | - |
| 3 | senha | cadeia de caracteres | 15 | - | USUARIO | Senha | - |
| 4 | nome | cadeia de caracteres | 100 | - | USUARIO | Nome | - |
| 5 | professor | booleano | 1 | - | - | - | - |
| assunto | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | - | - | - |
| 2 | nome | cadeia de caracteres | 100 | - | LISTA_EXERCICIO | Assunto[1-N] | - |
| questao | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | QUESTAO | Id | - |
| 2 | id_exercicio | inteiro | - | - | LISTA_EXERCICIO | Id | composta |
| 3 | login_usuario | inteiro | - | - | USUARIO | Login | cadastra |
| 4 | nome_esquema | cadeia de caracteres | 50 | - | ESQUEMA | Nome | - |
| 5 | pergunta | cadeia de caracteres | 1024 | - | QUESTAO | Pergunta | - |
| 6 | resposta | cadeia de caracteres | 1024 | - | QUESTAO | Resposta | - |
| duvida | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | DUVIDA | Id | - |
| 2 | id_assunto | inteiro | - | - | DUVIDA | Assunto[1-N] | - |
| 3 | login_usuario | cadeia de caracteres | 20 | - | USUARIO | Login | - |
| 4 | id_questao | inteiro | - | - | QUESTAO | Id | - |
| 5 | pergunta | cadeia de caracteres | 1024 | - | DUVIDA | Pergunta | - |
| 6 | dt_duvida | data | dd-mm-aaaa | - | - | Dt_Duvida | relata |
| comentario | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id | inteiro | - | auto incremental | COMENTARIO | Id | - |
| 2 | id_duvida | inteiro | - | - | DUVIDA | Id | - |
| 3 | login_usuario | cadeia de caracteres | 20 | - | USUARIO | Login | - |
| 4 | texto_comentario | cadeia de caracteres | 1024 | - | COMENTARIO | Texto_Comentario | - |
| 5 | dt_comentario | data | dd-mm-aaaa | - | - | Dt_Comentario | escreve |
| usuario_modera_duvida | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | login_moderador | cadeia de caracteres | - | - | USUARIO | Login | - |
| 2 | id_duvida | inteiro | 20 | - | DUVIDA | Id | - |
| 3 | justificativa | cadeia de caracteres | 1024 | - | - | Justificativa | modera |
| 4 | dt_cancelamento | data | dd-mm-aaaa | - | - | Dt_Cancelamento | modera |
| usuario_modera_comentario | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | login_moderador | cadeia de caracteres | - | - | USUARIO | Login | - |
| 2 | id_comentario | inteiro | 20 | - | COMENTARIO | Id | - |
| 3 | justificativa | cadeia de caracteres | 1024 | - | - | Justificativa | modera |
| 4 | dt_cancelamento | data | dd-mm-aaaa | - | - | Dt_Cancelamento | modera |
| assunto_duvida | | | | | | | |
| ID | Atributos | Tipo | Tamanho/Formato | Extra | Entidade | Atributo | Relacionamento |
| 1 | id_assunto | inteiro | - | - | DUVIDA | Assunto[1-N] | - |
| 2 | id_duvida | inteiro | - | - | DUVIDA | Id | - |

Tabela 3.3: Tabelas referentes ao Fórum (parte 1)

| | |
|---------------------------|--|
| usuario | |
| ID | Observações |
| 1 | Identificador da tupla, criado pelo usuário e faz parte da sua chave de acesso ao sistema. |
| 2 | Email para contato para notificações do fórum, liberação de exercícios e resgate de senha. |
| 3 | Criado pelo usuário e faz parte da sua chave de acesso ao sistema. |
| 4 | Nome completo do usuário. |
| 5 | Refere-se ao papel do usuário em questão (1 = professor, 0 = aluno) |
| assunto | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Nome do assunto relacionado. |
| questao | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Identifica a qual lista de exercício a questão está atrelada. |
| 3 | Identificação do usuário responsável pela criação da questão. |
| 4 | Identifica a qual esquema a questão pode estar vinculada. |
| 5 | Informa o enunciado da questão a ser respondida pelo aluno. |
| 6 | Trata-se do gabarito dado pelo professor. |
| duvida | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Informa quais temas são abordados na dúvida. |
| 3 | Identifica o aluno responsável pela dúvida. |
| 4 | Identifica um possível vínculo entre dúvida e questão. |
| 5 | Trata-se da pergunta feita pelo aluno no fórum. |
| 6 | Identifica quando a dúvida fora postada no fórum. |
| comentario | |
| ID | Observações |
| 1 | Identificador da tupla. |
| 2 | Identifica a qual dúvida se refere o comentário. |
| 3 | Identifica o aluno responsável pelo comentário. |
| 4 | Trata-se do comentário feito pelo aluno no fórum. |
| 5 | Identifica quando o comentário fora postada no fórum. |
| usuario_modera_duvida | |
| ID | Observações |
| 1 | Identifica o moderador que cancelou a dúvida. |
| 2 | Identifica qual dúvida foi cancelada. |
| 3 | Possui a justificativa do cancelamento dada pelo moderador do fórum. |
| 4 | Identifica quando a dúvida fora cancelada no fórum. |
| usuario_modera_comentario | |
| ID | Observações |
| 1 | Identifica o moderador que cancelou o comentário. |
| 2 | Identifica qual comentário foi cancelado. |
| 3 | Possui a justificativa do cancelamento dada pelo moderador do fórum. |
| 4 | Identifica quando o comentário fora cancelado no fórum. |
| assunto.duvida | |
| ID | Observações |
| 1 | Identifica quais assuntos são abordados nas dúvidas do fórum. |
| 2 | Identificador da dúvida. |

Tabela 3.4: Tabelas referentes ao Fórum (parte 2)

3.3

Modelagem Física

Durante esta fase, projetamos as especificações para o SGBD escolhido, no caso o **PostgreSQL 9.1** (Postgres), em termos de armazenamento físico das estruturas, da alocação de registros e dos índices.

Para melhor visualização da estrutura elaborada, os comandos SQL foram separados em três seções: **Tabelas**, que contém as sintaxes de criação dos elementos desvinculados, **Comentários**, que contém as sintaxes para adição de descrição dos campos e **Restrições**, que contém as sintaxes de criação de índices e chaves estrangeiras.

| |
|--|
| Tabela: usuario |
| <pre>CREATE TABLE public.usuario(login varchar(20) NOT NULL, senha varchar(15) NOT NULL, nome varchar(100) NOT NULL, email varchar(100) NOT NULL, professor bit NOT NULL DEFAULT b'0', CONSTRAINT pk_usuario PRIMARY KEY (login));</pre> |
| Tabela: usuario_modera_comentario |
| <pre>CREATE TABLE public.usuario_modera_comentario(login_moderador varchar(20) NOT NULL, id_comentario integer NOT NULL, justificativa varchar(1024) NOT NULL, dt_cancelamento timestamp NOT NULL, CONSTRAINT pk_forum PRIMARY KEY (login_moderador,id_comentario));</pre> |
| Tabela: duvida |
| <pre>CREATE TABLE public.duvida(id serial NOT NULL, id_assunto integer NOT NULL, login_usuario varchar(20) NOT NULL, id_questao integer, pergunta varchar(1024) NOT NULL, dt_duvida timestamp NOT NULL, CONSTRAINT pk_duvida PRIMARY KEY (id));</pre> |
| Tabela: comentario |
| <pre>CREATE TABLE public.comentario(id serial NOT NULL, id_duvida integer NOT NULL, login_usuario varchar(20) NOT NULL, texto_comentario varchar(1024) NOT NULL, dt_comentario timestamp NOT NULL, CONSTRAINT pk_comentario PRIMARY KEY (id));</pre> |
| Tabela: questao |
| <pre>CREATE TABLE public.questao(id serial NOT NULL, id_exercicio integer NOT NULL, login_usuario varchar(20) NOT NULL, nome_esquema varchar(50), pergunta varchar(1024) NOT NULL, resposta varchar(1024) NOT NULL, CONSTRAINT pk_questao PRIMARY KEY (id));</pre> |
| Tabela: usuario_modera_duvida |
| <pre>CREATE TABLE public.usuario_modera_duvida(login_moderador varchar(20) NOT NULL, id_duvida integer NOT NULL, justificativa varchar(1024) NOT NULL, dt_cancelamento timestamp NOT NULL, CONSTRAINT pk_usuario_modera_duvida PRIMARY KEY (login_moderador,id_duvida));</pre> |
| Tabela: assunto |
| <pre>CREATE TABLE public.assunto(id integer NOT NULL, nome varchar(100) NOT NULL, CONSTRAINT pk_assunto PRIMARY KEY (id));</pre> |

Tabela 3.5: Criação das Tabelas (parte 1)

| |
|--|
| Tabela: lista_exercicio |
| <pre>CREATE TABLE public.lista_exercicio(id serial NOT NULL, id_assunto integer NOT NULL, descricao varchar(1024), dt_liberada timestamp NOT NULL, CONSTRAINT pk_lista_exercicio PRIMARY KEY (id));</pre> |
| Tabela: resposta |
| <pre>CREATE TABLE public.resposta(id serial NOT NULL, id_questao integer NOT NULL, login_usuario varchar(20) NOT NULL, dt_resposta timestamp NOT NULL, dt_acerto timestamp, resposta varchar(1024) NOT NULL, CONSTRAINT pk_resposta PRIMARY KEY (id));</pre> |
| Tabela: esquema |
| <pre>CREATE TABLE public.esquema(nome varchar(50) NOT NULL, criacao varchar(4096) NOT NULL, CONSTRAINT pk_esquema PRIMARY KEY (nome));</pre> |
| Tabela: instancia |
| <pre>CREATE TABLE public.instancia(nome varchar(50) NOT NULL, nome_esquema varchar(50) NOT NULL, login_usuario varchar(20) NOT NULL, dt_criacao timestamp NOT NULL, CONSTRAINT pk_instancia PRIMARY KEY (nome));</pre> |
| Tabela: modificacao |
| <pre>CREATE TABLE public.modificacao(id serial NOT NULL, nome_instancia varchar(50) NOT NULL, login_usuario varchar(20) NOT NULL, dt_modificacao timestamp NOT NULL, comando varchar(1024) NOT NULL, CONSTRAINT pk_modificacao PRIMARY KEY (id));</pre> |
| Tabela: visualizacao |
| <pre>CREATE TABLE public.visualizacao(nome_instancia varchar(50) NOT NULL, login_usuario varchar(20) NOT NULL, CONSTRAINT pk_visualizacao PRIMARY KEY (nome_instancia,login_usuario));</pre> |
| Tabela: assunto_lista_exercicio |
| <pre>CREATE TABLE public.assunto_lista_exercicio(id_assunto integer NOT NULL, id_exercicio integer NOT NULL, CONSTRAINT pk_assunto_lista_exercicio PRIMARY KEY (id_assunto,id_exercicio));</pre> |
| Tabela: assunto_duvida |
| <pre>CREATE TABLE public.assunto_duvida(id_assunto integer NOT NULL, id_duvida integer NOT NULL, CONSTRAINT pk_assunto_duvida PRIMARY KEY (id_assunto,id_duvida));</pre> |

Tabela 3.6: Criação das Tabelas (parte 2)

| |
|--|
| Comentários :: Tabela : usuario |
| COMMENT ON COLUMN public.usuario.login IS 'Identificador da tupla, criado pelo usuário e faz parte da sua chave de acesso ao sistema.'; COMMENT ON COLUMN public.usuario.senha IS 'Criado pelo usuário e faz parte da sua chave de acesso ao sistema.'; COMMENT ON COLUMN public.usuario.nome IS 'Nome completo do usuário.'; COMMENT ON COLUMN public.usuario.email IS 'Email para contato para notificações do fórum, liberação de exercícios e resgate de senha.'; COMMENT ON COLUMN public.usuario.professor IS 'distingue o papel do usuario: aluno ou professor.'; |
| Comentários :: Tabela : usuario_modera_comentario |
| COMMENT ON COLUMN public.usuario_modera_comentario.login_moderador IS 'Identifica o moderador que cancelou o comentário.'; COMMENT ON COLUMN public.usuario_modera_comentario.id_comentario IS 'Identifica qual comentário foi cancelado.'; COMMENT ON COLUMN public.usuario_modera_comentario.justificativa IS 'Possui a justificativa do cancelamento dada pelo moderador do fórum.'; COMMENT ON COLUMN public.usuario_modera_comentario.dt_cancelamento IS 'Identifica quando o comentário fora cancelado no fórum.'; |
| Comentários :: Tabela : duvida |
| COMMENT ON COLUMN public.duvida.id IS 'Identificador da tupla.'; COMMENT ON COLUMN public.duvida.id_assunto IS 'Informa quais temas são abordados na dúvida.'; COMMENT ON COLUMN public.duvida.login_usuario IS 'Identifica o aluno responsável pela dúvida.'; COMMENT ON COLUMN public.duvida.id_questao IS 'Identifica um possível vínculo entre dúvida e questão.'; COMMENT ON COLUMN public.duvida.pergunta IS 'Trata-se da pergunta feita pelo aluno no fórum.'; COMMENT ON COLUMN public.duvida.dt_duvida IS 'Identifica quando a dúvida fora postada no fórum.'; |
| Comentários :: Tabela : comentario |
| COMMENT ON COLUMN public.comentario.id IS 'identificador dos comentarios'; COMMENT ON COLUMN public.comentario.id_duvida IS 'Identifica a qual dúvida se refere o comentário.'; COMMENT ON COLUMN public.comentario.login_usuario IS 'Identifica o aluno responsável pelo comentário.'; COMMENT ON COLUMN public.comentario.texto_comentario IS 'Trata-se do comentário feito pelo aluno no fórum.'; COMMENT ON COLUMN public.comentario.dt_comentario IS 'Identifica quando o comentário fora postada no fórum.'; |

Tabela 3.7: Adição dos Comentários (parte 1)

| |
|--|
| Comentários :: Tabela : questao |
| COMMENT ON COLUMN public.questao.id IS 'identificador da questao'; COMMENT ON COLUMN public.questao.id_exercicio IS 'Identifica a qual lista de exercício a questão está atrelada.'; COMMENT ON COLUMN public.questao.login_usuario IS 'Identificação do usuário responsável pela criação da questão.'; COMMENT ON COLUMN public.questao.nome_esquema IS 'Identifica a qual esquema a questão pode estar vinculada.'; COMMENT ON COLUMN public.questao.pergunta IS 'Informa o enunciado da questão a ser respondida pelo aluno.'; COMMENT ON COLUMN public.questao.resposta IS 'Trata-se do gabarito dado pelo professor.'; |
| Comentários :: Tabela : usuario_modera_duvida |
| COMMENT ON COLUMN public.usuario_modera_duvida.login_moderador IS 'Identifica o moderador que cancelou a dúvida.'; COMMENT ON COLUMN public.usuario_modera_duvida.id_duvida IS 'Identifica qual dúvida foi cancelada.'; COMMENT ON COLUMN public.usuario_modera_duvida.justificativa IS 'Possui a justificativa do cancelamento dada pelo moderador do fórum.'; COMMENT ON COLUMN public.usuario_modera_duvida.dt_cancelamento IS 'Identifica quando a dúvida fora cancelada no fórum.'; COMMENT ON CONSTRAINT pk_usuario_modera_duvida ON public.usuario_modera_duvida IS 'Chave primaria da tabela usuario_modera_duvida.'; |
| Comentários :: Tabela : assunto |
| COMMENT ON COLUMN public.assunto.id IS 'Identificador da tupla.'; COMMENT ON COLUMN public.assunto.nome IS 'Nome do assunto relacionado.'; COMMENT ON CONSTRAINT pk_assunto ON public.assunto IS 'Chave primaria da tabela assunto.'; |

Tabela 3.8: Adição dos Comentários (parte 2)

| |
|---|
| Comentários :: Tabela : lista_exercicio |
| COMMENT ON COLUMN public.lista_exercicio.id IS 'Identificador da tupla.'; COMMENT ON COLUMN public.lista_exercicio.id_assunto IS 'Informa quais temas são abordados na lista de exercício.'; COMMENT ON COLUMN public.lista_exercicio.descricao IS 'Possível sugestões do professor de como resolver a lista, com sugestão de referências bibliográficas.'; COMMENT ON COLUMN public.lista_exercicio.dt_liberada IS 'Armazena em que dia a lista foi liberada aos alunos.'; COMMENT ON CONSTRAINT pk_lista_exercicio ON public.lista_exercicio IS 'Chave primaria da tabela lista_exercicio.'; |
| Comentários :: Tabela : resposta |
| COMMENT ON COLUMN public.resposta.id IS 'Identificador da tupla.'; COMMENT ON COLUMN public.resposta.id_questao IS 'Identifica qual questão se refere a resposta dada.'; COMMENT ON COLUMN public.resposta.login_usuario IS 'Identifica qual aluno respondeu a questão.'; COMMENT ON COLUMN public.resposta.dt_resposta IS 'Identifica quando a resposta foi dada.'; COMMENT ON COLUMN public.resposta.dt_acerto IS 'Identifica quando houve o acerto da resposta.'; COMMENT ON COLUMN public.resposta.resposta IS 'Armazena qual foi a resposta dada.'; COMMENT ON CONSTRAINT pk_resposta ON public.resposta IS 'Chave primaria da tabela resposta.'; |
| Comentários :: Tabela : esquema |
| COMMENT ON COLUMN public.esquema.nome IS 'Identificador da tupla.'; COMMENT ON COLUMN public.esquema.criacao IS 'Trata-se do comando DDL para criação do esquema.'; COMMENT ON CONSTRAINT pk_esquema ON public.esquema IS 'Chave primaria da tabela esquema.'; |
| Comentários :: Tabela : instancia |
| COMMENT ON COLUMN public.instancia.nome IS 'Identificador da tupla.'; COMMENT ON COLUMN public.instancia.nome_esquema IS 'Identifica qual esquema que originou a presente instância.'; COMMENT ON COLUMN public.instancia.login_usuario IS 'Identifica qual o aluno responsável pelas modificações da instância.'; COMMENT ON COLUMN public.instancia.dt_criacao IS 'Identifica quando a instância foi criada pelo aluno.'; COMMENT ON CONSTRAINT pk_instancia ON public.instancia IS 'Chave primaria da tabela instancia.'; |

Tabela 3.9: Adição dos Comentários (parte 3)

| |
|---|
| Comentários :: Tabela : modificacao |
| COMMENT ON COLUMN public.modificacao.id IS 'Identificador da tupla.'; |
| COMMENT ON COLUMN public.modificacao.nome_instancia IS 'Identifica a instância que está sendo alterada.'; |
| COMMENT ON COLUMN public.modificacao.login_usuario IS 'Identifica o usuário autor da modificação.'; |
| COMMENT ON COLUMN public.modificacao.dt_modificacao IS 'Identifica quando a instância foi alterada.'; |
| COMMENT ON COLUMN public.modificacao.comando IS 'Trata-se do comando DDL para alteração do esquema.'; |
| COMMENT ON CONSTRAINT pk_modificacao ON public.modificacao IS 'Chave primaria da tabela modificacao.'; |
| Comentários :: Tabela : visualizacao |
| COMMENT ON COLUMN public.visualizacao.nome_instancia IS 'Identifica a instância que está sendo alterada.'; |
| COMMENT ON COLUMN public.visualizacao.login_usuario IS 'Identifica o usuário autor da modificação.'; |
| Comentários :: Tabela : assunto_lista_exercicio |
| COMMENT ON COLUMN public.assunto_lista_exercicio.id_assunto IS 'Identifica quais assuntos são abordados na lista de exercícios.'; |
| COMMENT ON COLUMN public.assunto_lista_exercicio.id_exercicio IS 'Identificador da lista de exercício.'; |
| COMMENT ON CONSTRAINT pk_assunto_lista_exercicio ON public.assunto_lista_exercicio IS 'Chave primaria da tabela assunto_lista_exercicio.'; |
| Comentários :: Tabela : assunto_duvida |
| COMMENT ON COLUMN public.assunto_duvida.id_assunto IS 'Identifica quais assuntos são abordados nas dúvidas do fórum.'; |
| COMMENT ON COLUMN public.assunto_duvida.id_exercicio IS 'Identificador da dúvida.'; |
| COMMENT ON CONSTRAINT pk_assunto_duvida ON public.assunto_duvida IS 'Chave primaria da tabela assunto_duvida.'; |

Tabela 3.10: Adição dos Comentários (parte 4)

| |
|--|
| Chaves Estrangeiras :: Tabela : lista_exercicio |
| ALTER TABLE lista_exercicio ADD CONSTRAINT fk_lista_exercicio_assunto FOREIGN KEY (id_assunto) REFERENCES assunto (id); |
| Chaves Estrangeiras :: Tabela : questao |
| ALTER TABLE questao ADD CONSTRAINT fk_questao_lista_exercicio FOREIGN KEY (id_exercicio) REFERENCES lista_exercicio (id); ALTER TABLE questao ADD CONSTRAINT fk_questao_usuario FOREIGN KEY (login_usuario) REFERENCES usuario (login); ALTER TABLE questao ADD CONSTRAINT fk_questao_esquema FOREIGN KEY (nome_esquema) REFERENCES esquema (nome); |
| Chaves Estrangeiras :: Tabela : resposta |
| ALTER TABLE resposta ADD CONSTRAINT fk_resposta_questao FOREIGN KEY (id_questao) REFERENCES questao (id); ALTER TABLE resposta ADD CONSTRAINT fk_resposta_usuario FOREIGN KEY (login_usuario) REFERENCES usuario (login); |
| Chaves Estrangeiras :: Tabela : instancia |
| ALTER TABLE instancia ADD CONSTRAINT fk_instancia_esquema FOREIGN KEY (nome_esquema) REFERENCES esquema (nome); ALTER TABLE instancia ADD CONSTRAINT fk_instancia_usuario FOREIGN KEY (login_usuario) REFERENCES usuario (login); |
| Chaves Estrangeiras :: Tabela : modificacao |
| ALTER TABLE modificacao ADD CONSTRAINT fk_modificacao_instancia FOREIGN KEY (nome_instancia) REFERENCES instancia (nome); ALTER TABLE modificacao ADD CONSTRAINT fk_modificacao_usuario FOREIGN KEY (login_usuario) REFERENCES usuario (login); |
| Chaves Estrangeiras :: Tabela : visualizacao |
| ALTER TABLE visualizacao ADD CONSTRAINT fk_visualizacao_instancia FOREIGN KEY (nome_instancia) REFERENCES instancia (nome); ALTER TABLE visualizacao ADD CONSTRAINT fk_visualizacao_usuario FOREIGN KEY (login_usuario) REFERENCES usuario (login); |
| Chaves Estrangeiras :: Tabela : assunto_lista_exercicio |
| ALTER TABLE assunto_lista_exercicio ADD CONSTRAINT fk_assunto_lista_exercicio_assunto FOREIGN KEY (id_assunto) REFERENCES assunto (id); ALTER TABLE assunto_lista_exercicio ADD CONSTRAINT fk_assunto_lista_exercicio_exercicio FOREIGN KEY (id_exercicio) REFERENCES lista_exercicio (id); |

Tabela 3.11: Adição das Restrições (parte 1)

| |
|--|
| Chaves Estrangeiras :: Tabela : duvida |
| ALTER TABLE duvida ADD CONSTRAINT fk_duvida_assunto FOREIGN KEY (id_assunto) REFERENCES assunto (id); ALTER TABLE duvida ADD CONSTRAINT fk_duvida_usuario FOREIGN KEY (login_usuario) REFERENCES usuario (login); ALTER TABLE duvida ADD CONSTRAINT fk_duvida_questao FOREIGN KEY (id_questao) REFERENCES questao (id); |
| Chaves Estrangeiras :: Tabela : comentario |
| ALTER TABLE comentario ADD CONSTRAINT fk_comentario_duvida FOREIGN KEY (id_duvida) REFERENCES duvida (id); ALTER TABLE comentario ADD CONSTRAINT fk_comentario_usuario FOREIGN KEY (login_usuario) REFERENCES usuario (login); |
| Chaves Estrangeiras :: Tabela : usuario_modera_duvida |
| ALTER TABLE usuario_modera_duvida ADD CONSTRAINT fk_usuario_modera_duvida_usuario FOREIGN KEY (login_moderador) REFERENCES usuario (login); ALTER TABLE usuario_modera_duvida ADD CONSTRAINT fk_usuario_modera_duvida_duvida FOREIGN KEY (id_duvida) REFERENCES duvida (id); |
| Chaves Estrangeiras :: Tabela : usuario_modera_comentario |
| ALTER TABLE usuario_modera_comentario ADD CONSTRAINT fk_usuario_modera_comentario_usuario FOREIGN KEY (login_moderador) REFERENCES usuario (login); ALTER TABLE usuario_modera_comentario ADD CONSTRAINT fk_usuario_modera_comentario_comentario FOREIGN KEY (id_comentario) REFERENCES comentario (id); |
| Chaves Estrangeiras :: Tabela : assunto_duvida |
| ALTER TABLE assunto_duvida ADD CONSTRAINT fk_assunto_duvida_duvida FOREIGN KEY (id_duvida) REFERENCES duvida (id); ALTER TABLE assunto_duvida ADD CONSTRAINT fk_assunto_duvida_assunto FOREIGN KEY (id_assunto) REFERENCES assunto (id); |

Tabela 3.12: Adição das Restrições (parte 2)

3.4

Casos de Uso

3.4.1

Acesso à Plataforma

Este caso de uso demonstra como será realizada a verificação de cadastro do usuário, aluno ou professor. As credenciais informadas são verificadas com os dados cadastrais contidos no Banco de Dados e o usuário será redirecionado para a página de entrada no sistema.

Pré-condições:

1. O computador deve estar com acesso à internet.
2. O sistema deve estar hospedado em algum servidor.

Cenários:

1. Verificação das credenciais de acesso realizadas com sucesso.

Fluxo Normal:

1. Usuário preenche o campo login.
2. Usuário preenche o campo senha.
3. Usuário pressiona a tecla 'Enter' ou clica no botão 'Entrar'.
4. Sistema recebe a requisição de entrada.
5. As credenciais digitadas são verificadas no Banco de Dados.
6. Sistema emite feedback positivo ao Usuário, através do acesso à página inicial.

Fluxos Alternativos:

5a. As credenciais de acesso digitadas pelo Usuário não são encontradas no Banco de Dados.

1. Usuário preenche o campo login.
2. Usuário preenche o campo senha.
3. Usuário pressiona a tecla 'Enter' ou clica no botão 'Entrar'.
4. Sistema recebe a requisição de entrada.
5. As credenciais digitadas não são encontradas no Banco de Dados.
6. Sistema emite feedback negativo ao Usuário, através de mensagem de erro.

3.4.2

Cadastro de Nova Dúvida no Fórum

Este caso de uso demonstra como os alunos postarão novas dúvidas no módulo de fórum. O usuário deverá clicar no botão 'Nova Dúvida' e redigir sua dúvida no campo disponível. Ao fim, basta pressionar o botão 'Salvar'. O sistema emitirá uma mensagem confirmando a operação.

Pré-condições:

1. O computador deve estar com acesso à internet.
2. O sistema deve estar hospedado em algum servidor.
3. O aluno deve estar logado no sistema.

Cenários:

1. Cadastro com sucesso de nova Dúvida no Fórum.

Fluxo Normal:

1. Usuário seleciona a opção 'Fórum' disponível no menu do Sistema.
2. Sistema recebe a requisição e retorna à página de Fórum.
3. Usuário deve pressionar o botão 'Nova Dúvida'.
4. Usuário redige sua Dúvida no campo disponível.
5. Usuário deve pressionar o botão 'Salvar'.
6. Sistema recebe a requisição e armazena a Dúvida no Banco de Dados.
7. Sistema emite feedback positivo ao Usuário, através de mensagem.

Fluxos Alternativos:

*a. Em qualquer instante, o Sistema falha.

1. Usuário reinicia o Sistema, pressionando F5 ou acessando o endereço inicial.
 - (a) Sistema continua em estado de falha.
 - i. Usuário deve redigir um e-mail informando o erro ao professor da disciplina.
- 7a. Sistema não consegue persistir a informação no Banco de Dados.

1. Usuário seleciona a opção 'Fórum' disponível no menu do Sistema.
2. Sistema recebe a requisição e retorna à página de Fórum.
3. Usuário deve pressionar o botão 'Nova Dúvida'.
4. Usuário redige sua Dúvida no campo disponível.
5. Usuário deve pressionar o botão 'Salvar'.
6. Sistema recebe a requisição, mas não consegue persistir a informação no Banco de Dados.
7. Sistema emite feedback negativo ao Usuário, através de mensagem.

3.4.3

Resolução das Listas de Exercícios

Este caso de uso demonstra como os alunos resolverão as listas de exercícios. Os alunos deverão selecionar uma das opções de lista para ser feita. Após, uma página com as questões da respectiva lista será exibida.

Pré-condições:

1. O computador deve estar com acesso à internet.
2. O sistema deve estar hospedado em algum servidor.
3. O professor ou monitor deve ter cadastrado as listas de exercícios e suas questões.
4. O aluno deve estar logado no sistema.

Cenários:

1. Praticar a linguagem SQL através da resolução das listas de exercícios.

Fluxo Normal:

1. Usuário seleciona a opção 'Exercícios' disponível no menu do Sistema.
2. Sistema recebe a requisição e acessa o Banco de Dados para buscar as Listas de Exercícios.
3. Sistema exibe a listagem de todas as Listas de Exercícios cadastradas.
4. Usuário seleciona a Lista de Exercícios desejada.

5. Sistema recebe a requisição e acessa o Banco de Dados para buscar as Questões.
 6. Sistema exibe a listagem de todas as Questões da respectiva Lista de Exercícios.
 7. Usuário escolhe uma das Questões para resolver.
 8. Usuário preenche o campo disponível com o comando SQL que acredita ser o correto.
 9. Usuário pressiona o botão 'Enviar'.
 10. Sistema recebe a requisição e acessa o Banco de Dados para obter o gabarito da Questão.
 11. Ambos os comandos SQL são executados e seus resultados comparados.
 12. Sistema exibe uma mensagem de feedback da Questão resolvida.
- I Usuário repete os passos 7–12 até o fim da Lista de Exercícios.

Fluxos Alternativos:

*a. Em qualquer instante, o Sistema falha.

1. Usuário reinicia o Sistema, pressionando F5 ou acessando o endereço inicial.
 - (a) Sistema continua em estado de falha.
 - i. Usuário deve redigir um e-mail informando o erro ao professor da disciplina.
- 11a. Erro não previsto pelo Sistema ao executar os comandos SQL.
 1. Usuário seleciona a opção 'Exercícios' disponível no menu do Sistema.
 2. Sistema recebe a requisição e acessa o Banco de Dados para buscar as Listas de Exercícios.
 3. Sistema exibe a listagem de todas as Listas de Exercícios cadastradas.
 4. Usuário seleciona a Lista de Exercícios desejada.
 5. Sistema recebe a requisição e acessa o Banco de Dados para buscar as Questões.

6. Sistema exibe a listagem de todas as Questões da respectiva Lista de Exercícios.
7. Usuário escolhe uma das Questões para resolver.
8. Usuário preenche o campo disponível com o SQL que acredita ser o correto.
9. Usuário pressiona o botão 'Enviar'.
10. Sistema recebe a requisição e acessa o Banco de Dados para obter o SQL gabarito da Questão.
11. Sistema tenta lidar como uma exceção não tratada, durante execução do comando SQL.
12. Sistema exibe ao Usuário uma tela descrevendo o erro.
13. Usuário reinicia o Sistema, pressionando F5 ou acessando o endereço inicial.

3.5

Diagrama de Classes

Para uma melhor visualização e entendimento do Diagrama de Classes, o mesmo foi dividido em três modelos. Cada um deles refere-se às classes contidas nos pacotes do projeto. Os pacotes são: **Templates**, **Views** e **Models**.

No pacote **Models**, estão contidos os arquivos a serem renderizados no browser. Perceba que foi realizado um template base para os demais templates. A exceção se dá na tela de Login.

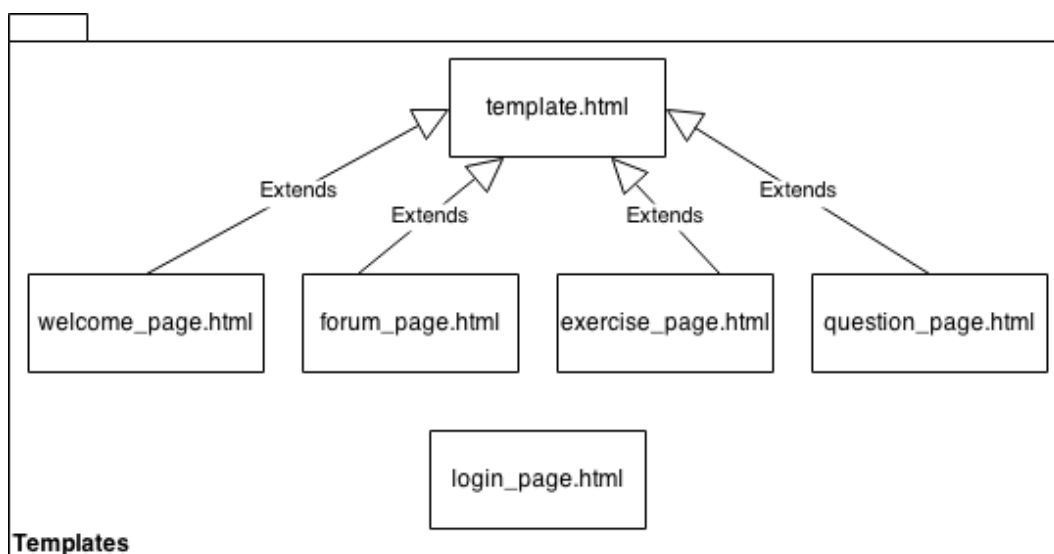


Figura 3.7: Diagrama de Classes do pacote Templates

No pacote **Views**, estão contidos os arquivos responsáveis por tratar as requisições realizadas pelos usuários do sistema.

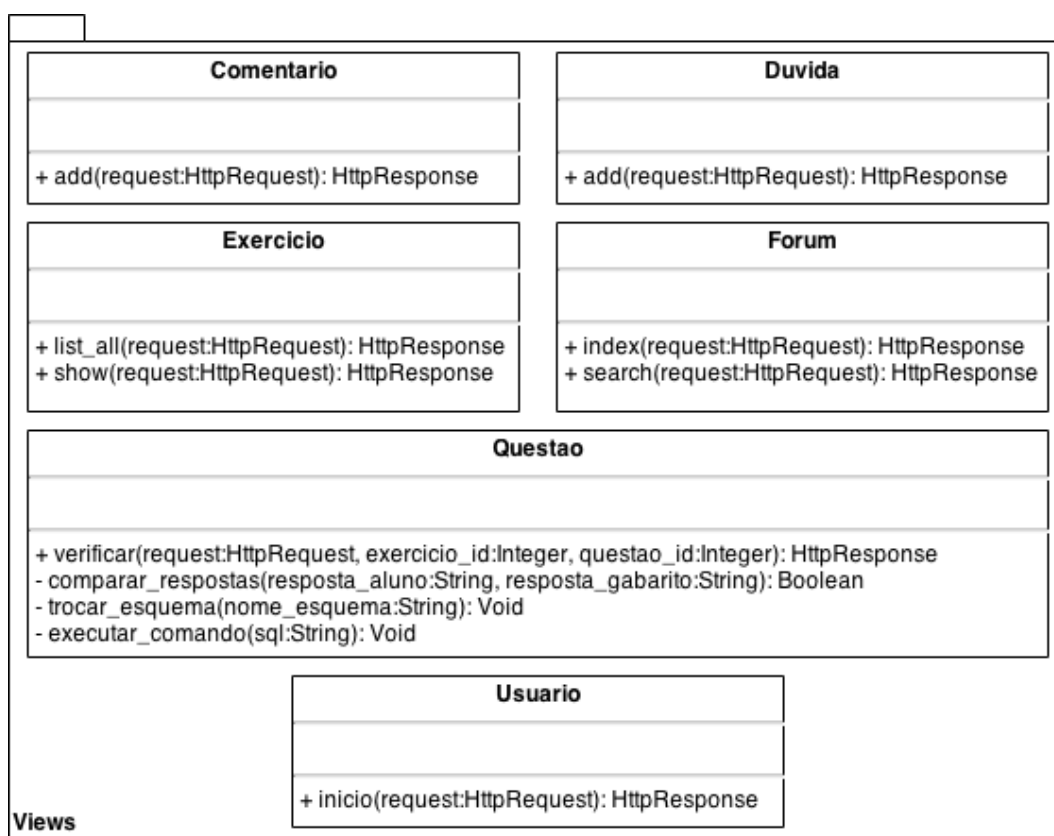
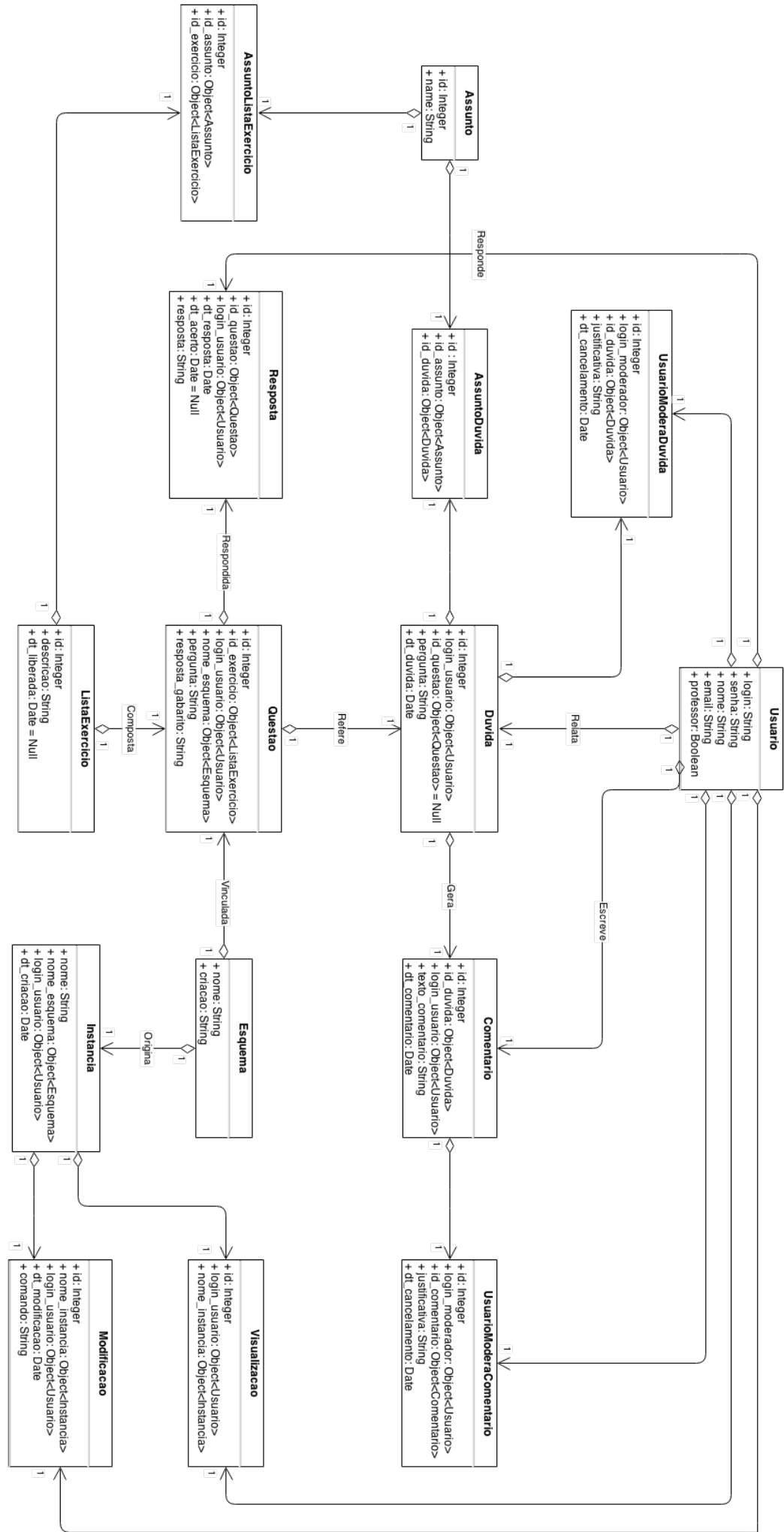


Figura 3.8: Diagrama de Classes do pacote Views

No pacote **Models**, estão contidos os arquivos que servem de mapeamento objeto relacional. Ou seja, uma representação dos elementos do banco de dados.

Diferentemente das modelagens anteriores, a ilustração contempla uma solução unificada entre **Fórum** e **Exercícios**. A figura da página seguinte segue na horizontal para facilitar o entendimento.



4

Plataforma Web

Este capítulo tem como objetivo descrever a metodologia de desenvolvimento do presente projeto, assim como destacar todas as ferramentas utilizadas.

Como o sistema, desde seu início, possui cunho opensource, foram utilizadas ferramentas do mesmo tipo para elaboração do projeto. Para facilitar a documentação, evolução da plataforma e acompanhamento pelo Prof. Sérgio Lifschitz, foi criado um **blog** (Blog) referente ao projeto final.

Para controle de versão e futura participação de outros interessados no projeto, foi utilizado o **git**, através da plataforma **GitHub**. Já para organização das tarefas, foi utilizado o **Trello**.

A fim de minimizar o tempo de desenvolvimento do sistema, optou-se por um framework para sistemas web. O escolhido foi o **Django**, que tem **Python** como sua linguagem de programação. E, por fim, para armazenamento dos dados e aprendizado dos alunos, utilizou-se o SGBD **Postgres**, por ser gratuito, opensource e de fácil utilização.

4.1

Mockups de Telas

Para facilitar a etapa de desenvolvimento da plataforma, foram elaborados mocks das telas. Dessa forma, foi possível definir, juntamente com os **Casos de Uso** (apresentados anteriormente), como se daria a interação dos usuários com o sistema e o fluxo de informação no mesmo.

Seguem os protótipos das telas:

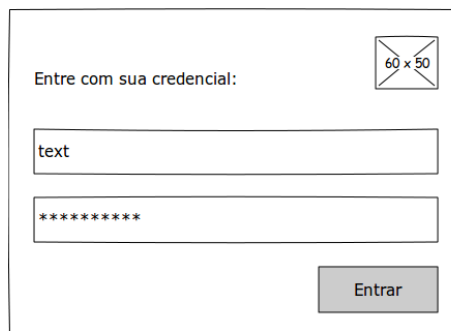


Diagram of a user login form. It contains a label "Entre com sua credencial:" followed by a text input field labeled "text" and a password input field labeled "*****". A "60 x 50" icon is positioned to the right of the label. Below the input fields is a button labeled "Entrar".

Figura 4.1: Página para validação do usuário

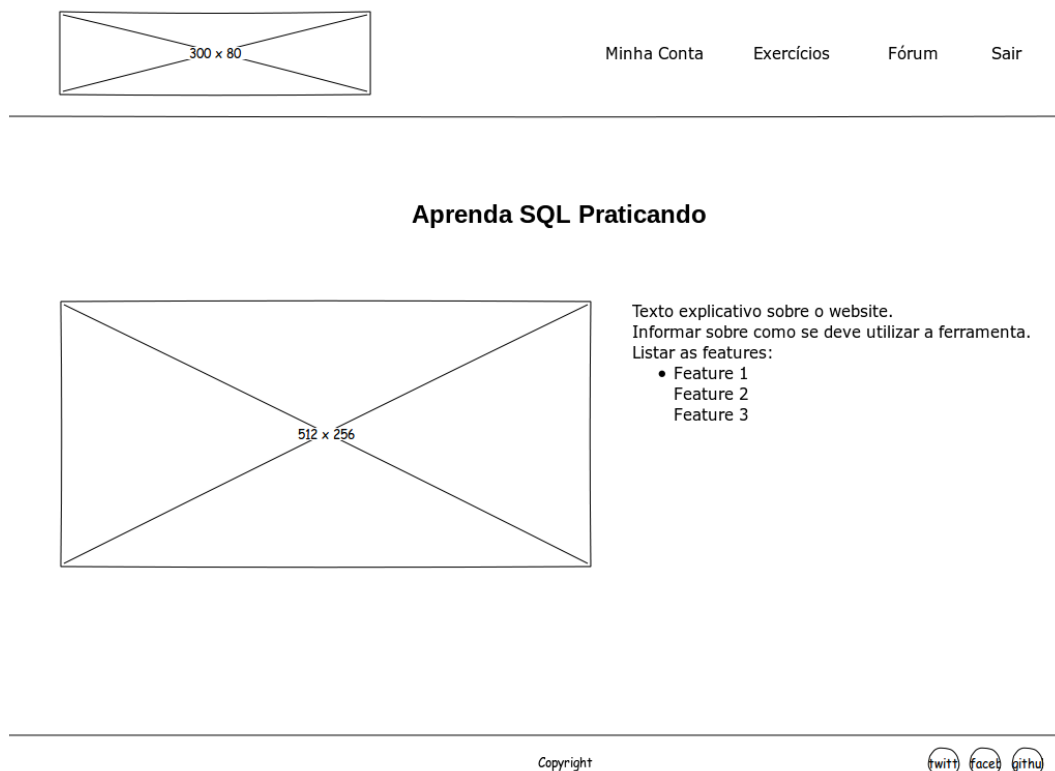


Diagram of the main page layout. It features a header with a navigation bar containing links: "Minha Conta", "Exercícios", "Fórum", and "Sair". Below the navigation bar is a horizontal line. The main content area is titled "Aprenda SQL Praticando". It includes a large rectangular area labeled "512 x 256" and a list of features: "Feature 1", "Feature 2", and "Feature 3". To the right of the list is a text block: "Texto explicativo sobre o website. Informar sobre como se deve utilizar a ferramenta. Listar as features:". The footer contains a "Copyright" notice and social media icons for "twitt", "faceb", and "githu".

Figura 4.2: Página inicial do sistema

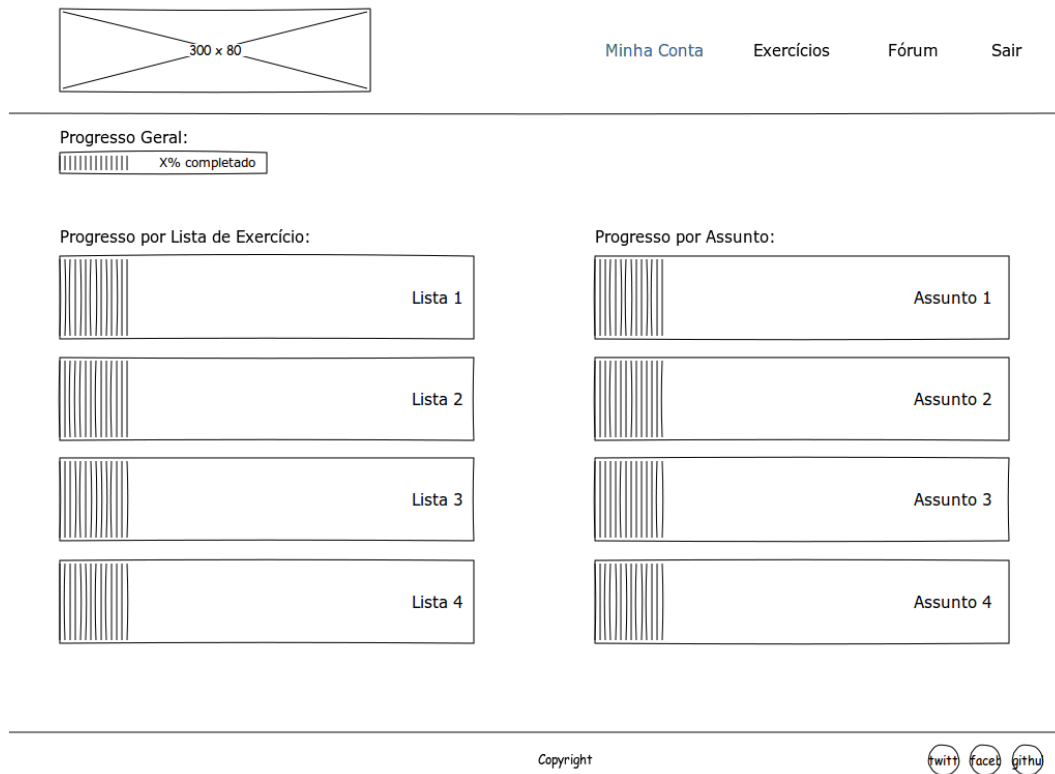


Figura 4.3: Página que mostra o desempenho dos alunos

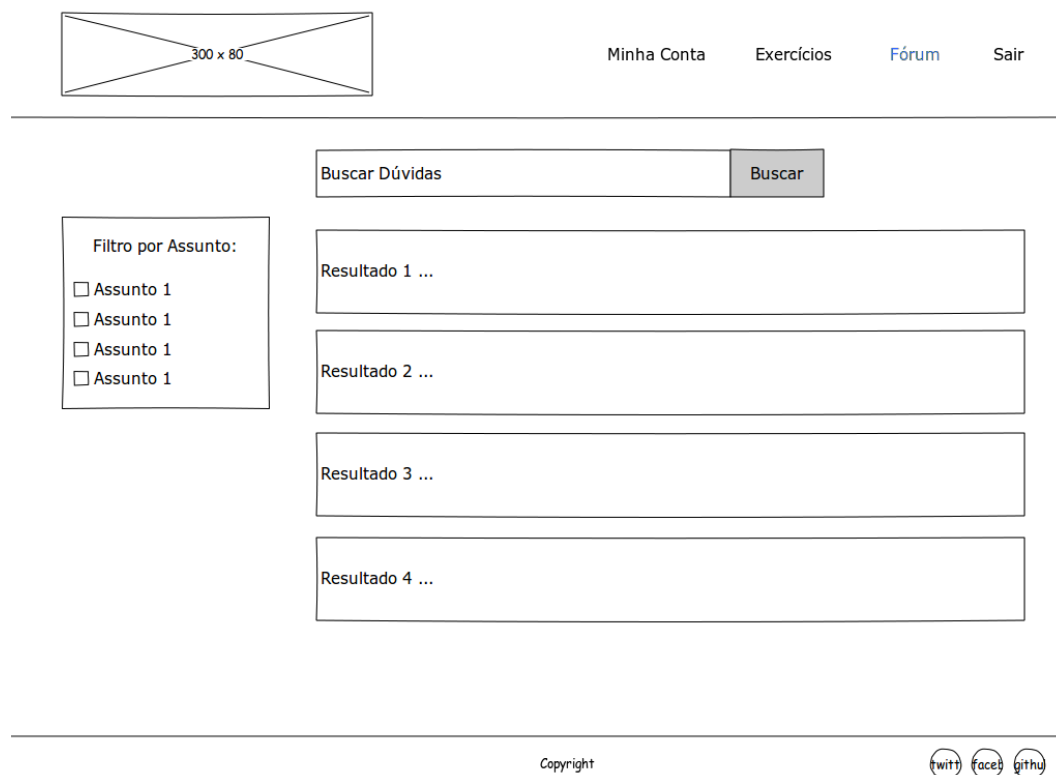


Figura 4.4: Página de busca e inserção de dúvidas no fórum

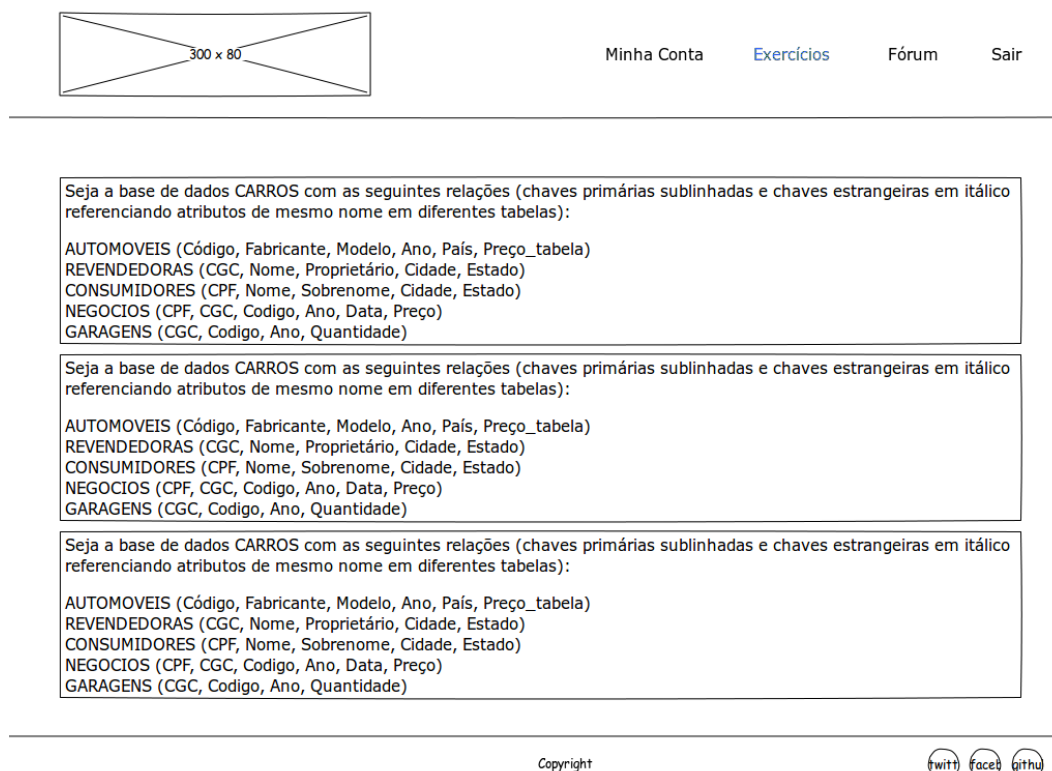


Figura 4.5: Página que lista os exercícios cadastrados



Minha ContaExercíciosFórumSair

Lista de Exercícios X

Seja a base de dados CARROS com as seguintes relações (chaves primárias sublinhadas e chaves estrangeiras em itálico referenciando atributos de mesmo nome em diferentes tabelas):

AUTOMOVEIS (Código, Fabricante, Modelo, Ano, País, Preço_tabela)
REVENDEDORAS (CGC, Nome, Proprietário, Cidade, Estado)
CONSUMIDORES (CPF, Nome, Sobrenome, Cidade, Estado)
NEGOCIOS (CPF, CGC, Código, Ano, Data, Preço)
GARAGENS (CGC, Código, Ano, Quantidade)

1. Listar os nomes dos fabricantes dos automóveis na base de dados e os respectivos países de fabricação originalmente.

Enviar

Copyright






Figura 4.6: Página com as questões que compõem a lista selecionada

4.2

Framework Django

A escolha do framework Django (Django) se deu, principalmente, pela sua arquitetura bastante modularizada. O código, conseqüentemente, fica fortemente coeso e fracamente acoplado. Além disso, trata-se de uma ferramenta que vem sendo bastante utilizada, o que pode facilitar a aceitação da ferramenta pela comunidade de desenvolvedores.

Uma característica do framework é ser baseado na arquitetura Model–View–Template, ao invés da tradicional Model–View–Controller. Porém, será possível perceber que trata-se apenas de uma escolha no nome, e que, na verdade, ambas as arquiteturas são bastante similares.

Segue abaixo uma figura que demonstra a estrutura do framework e suas devidas explicações:

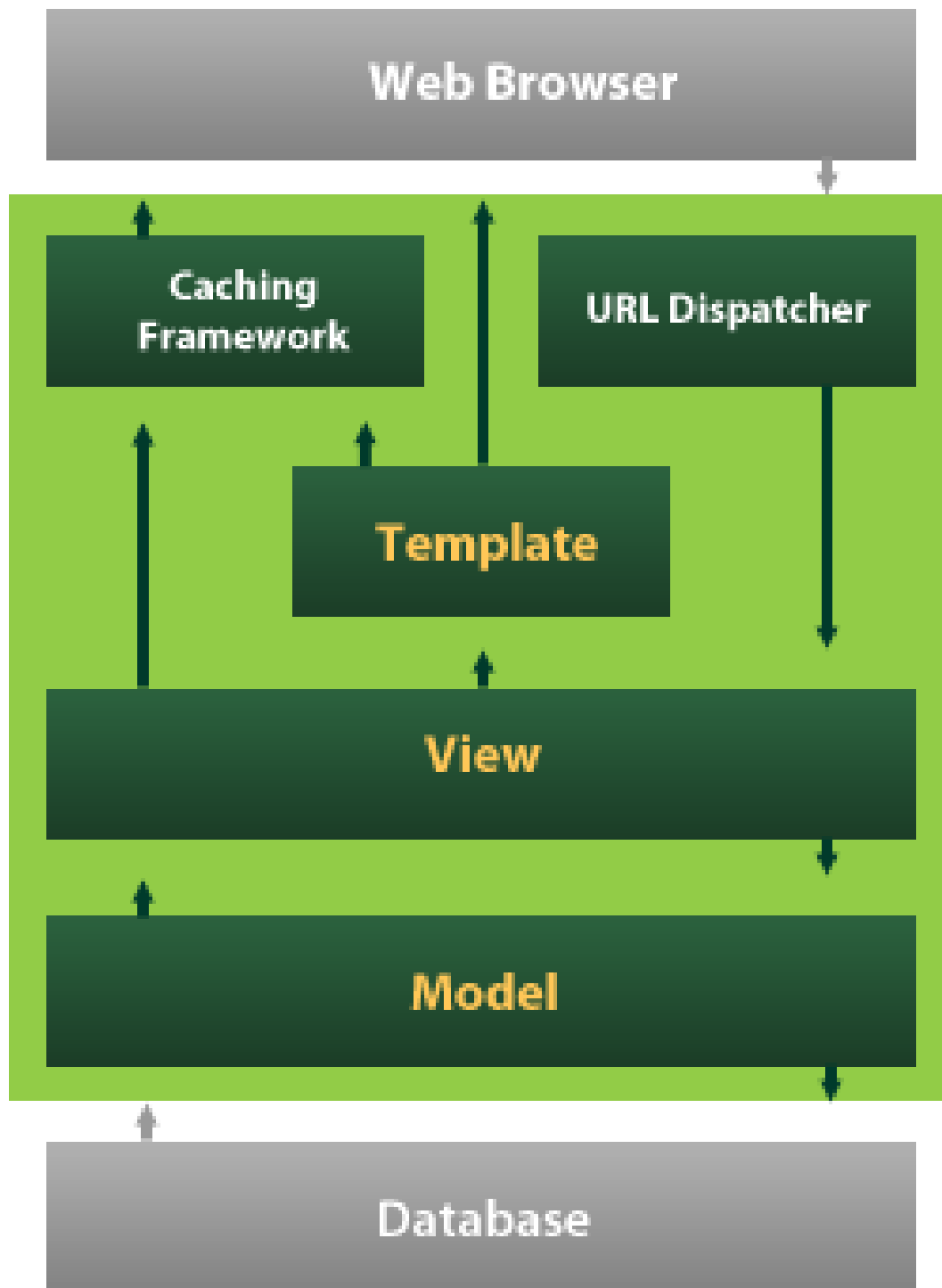


Figura 4.7: Arquitetura do Framework Django

1. URL Dispatcher

- Responsável pelo mapeamento entre as requisições feitas pelos usuários, baseadas na URL, e os métodos localizados nas views, que tem por função atender às solicitações.

2. View

- Responsável por executar as ações das requisições realizadas, que, tipicamente, envolvem leitura ou escrita no banco de dados. Pode incluir outras funcionalidades também.

3. Model

- Responsável pela definição dos dados e a interação entre os mesmos. Esse módulo trata do mapeamento objeto relacional (da sigla ORM em inglês). Além de mapear objetos provenientes do banco de dados, permite mapeamento em outros mecanismos de armazenamento.

4. Template

- Responsável pela renderização das páginas HTML. Permite a utilização de sintaxe própria, de fácil aprendizado e similares a linguagem de expressão (da sigla UEL em inglês).

4.3

Screenshots do Sistema

Para ilustrar o resultado final do projeto, seguem alguns screenshots do sistema. As telas servem de comparativo com o que foi sugerido na seção referente aos Mockups de Telas.

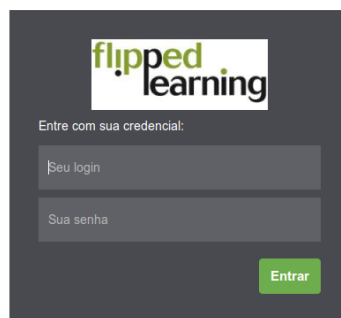


Figura 4.8: Página para validação do usuário



Figura 4.9: Página inicial do sistema

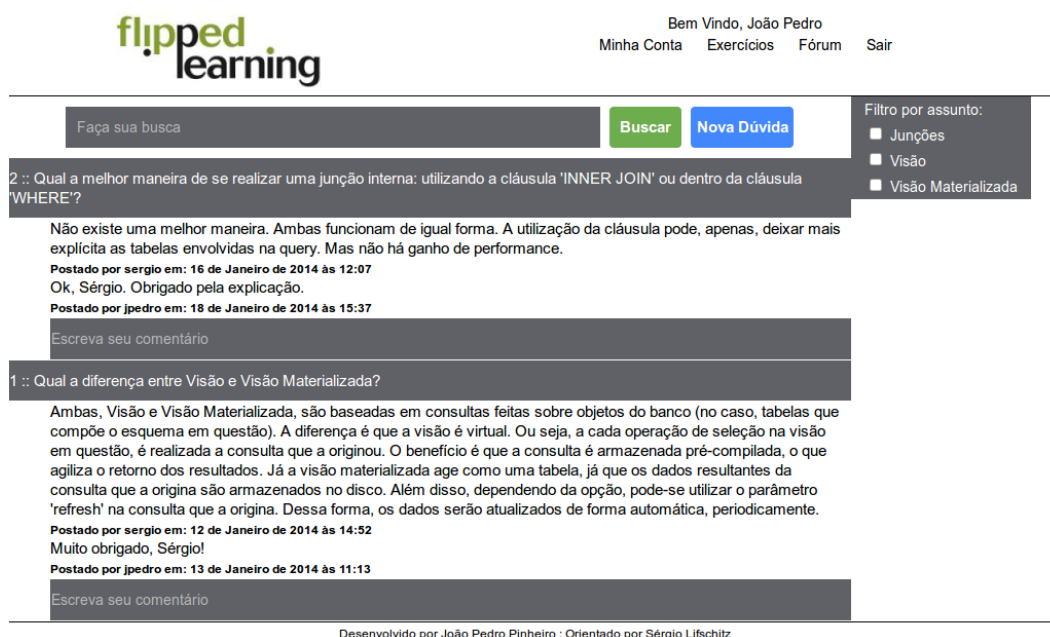



Figura 4.10: Página de busca e inserção de dúvidas no fórum



Bem Vindo, João Pedro
[Minha Conta](#) [Exercícios](#) [Fórum](#) [Sair](#)

Seja a base de dados CARROS com as seguintes relações:

AUTOMOVEIS ({Código}, Fabricante, Modelo, {Ano}, País, Preço_tabela)

REVENDEDORAS ({CGC}, Nome, Proprietário, Cidade, Estado)

CONSUMIDORES ({CPF}, Nome, Sobrenome, Cidade, Estado)

NEGOCIOS ({CPF}, {CGC}, {Codigo}, {Ano}, Data, Preço)

GARAGENS ({CGC}, {Codigo}, {Ano}, Quantidade)

* chaves primárias entre {}

* chaves estrangeiras entre []

Seja o esquema relacional da base de dados de uma Empresa:

EMPREGADO ({Ident}, Nome, Sal, End, Sexo, DataNasc, [DepNum], [SuperIdent])

DEPARTAMENTO ({Num}, Nome, [IdentGer], DataIni)

PROJETO ({Num}, Nome, Local, [DepNum])

TRABALHANO ({IdentEmp}, {[ProjNum]}, HRS)

DEPENDENTE ({IdentEmp}, {Nome}, Sexo, DataNasc, Parentesco)


DEPLOC ({DepNum}, {Local})

* chaves primárias entre {}

* chaves estrangeiras entre []

Desenvolvido por João Pedro Pinheiro ; Orientado por Sérgio Lifschitz

Figura 4.11: Página que lista os exercícios cadastrados



Bem Vindo, João Pedro
[Minha Conta](#) [Exercícios](#) [Fórum](#) [Sair](#)

Lista de Exercícios 1

Seja a base de dados CARROS com as seguintes relações:

AUTOMOVEIS ({Código}, Fabricante, Modelo, {Ano}, País, Preço_tabela)

REVENDEDORAS ({CGC}, Nome, Proprietário, Cidade, Estado)

CONSUMIDORES ({CPF}, Nome, Sobrenome, Cidade, Estado)

NEGOCIOS ({CPF}, {CGC}, {Codigo}, {Ano}, Data, Preço)

GARAGENS ({CGC}, {Codigo}, {Ano}, Quantidade)

* chaves primárias entre {}

* chaves estrangeiras entre []

- Listar os nomes dos fabricantes dos automóveis na base de dados e os respectivos países de fabricação originalmente.

Enviar

- Listar os estados onde se vende o modelo Elba, do fabricante Fiat.

Enviar

- Quais revendedoras não vendem automóveis de origem francesa?

Figura 4.12: Página com as questões que compõe a lista selecionada

5

Considerações Finais

Esse trabalho teve como objetivo criar uma ferramenta online com viés acadêmico e opensource, focado, inicialmente, em facilitar o ensino do conteúdo da disciplina Banco de Dados. Todas as seções anteriores mostraram as etapas de desenvolvimento da plataforma, incluindo um estudo de outros softwares similares.

Através desse levantamento de dados, pôde-se perceber que a utilização desse aprendizado às avessas (FlippedLearning) nada mais é que uma tentativa de aproximar os estudantes dos temas abordados em sala. Porém, fazendo essa tentativa de aproximação via tecnologia web e móvel.

Assim sendo, teremos uma real validação do projeto elaborado com a utilização do mesmo pelos alunos nos próximos períodos. Haverá possíveis ajustes a serem feitos, assim como a implementação de algumas funcionalidades que foram propostas, mas não contempladas na presente versão.

Segue a lista dos requisitos que não foram contemplados:

R10 Monitoramento da evolução dos alunos.

- Será uma tela importante quando o sistema estiver em pleno funcionamento. O professor da disciplina terá possibilidade de ver um relatório de evolução da turma, em uma visão geral, e um relatório filtrando por alunos.

R11 Permissão para cancelar uma thread inadequada no fórum.

- Essa funcionalidade será importante, pois auxiliará a manter o fórum com temas relevantes a disciplina.

R13 Possibilidade de compartilhar com seus colegas de turma suas instâncias.

- Nesta versão, cada aluno possui sua instância, referente a cada lista de exercícios feita. Porém, ainda não é possível compartilhar suas instâncias. Além disso, não é possível recuperar estados anteriores da base após execução de algum comando **DDL** incorreto. Essa será uma funcionalidade diferencial.

R16 Possibilidade de fazer upload de foto para facilitar a identificação do aluno.

- Essa funcionalidade auxiliará, principalmente, no acompanhamento dos relatórios por alunos, assim como no acompanhamento da participação dos mesmos no fórum.

R17 Possibilidade de compartilhar as questões em redes sociais (Facebook, Google+ e Twitter).

- Apenas será interessante se, algum dia, a plataforma crescer e obtiver um alcance maior de alunos e interessados.

Referências Bibliográficas

- [Blog] Disponível em: <http://dbflippedlearning.blogspot.com.br/>; Acessado em 19 de jan. de 2014. 4
- [CodeSchool] Disponível em: <http://www.codeschool.com/>; Acessado em 19 de jan. de 2014. 1
- [Django] Disponível em: <https://www.djangoproject.com/>; Acessado em 19 de jan. de 2014. 4.2
- [Edx] Disponível em: <https://www.edx.org/>; Acessado em 19 de jan. de 2014. 1
- [ElmasriNavathe05] RAMEZ ELMASRI, S. B. N.. **Sistemas de Banco de Dados**. Pearson, São Paulo, 2005. 2
- [Fine09] FINE, L. G.. **The SWOT Analysis: Using your Strength to overcome Weaknesses, Using Opportunities to overcome Threats**. Kick It, 2009. 2.2
- [FlippedLearning] Disponível em: <http://www.knewton.com/flipped-classroom/>; Acessado em 19 de jan. de 2014. (document), 1, 1.1, 5
- [Gradiance] Disponível em: <http://www.gradiance.com/>; Acessado em 19 de jan. de 2014. 1
- [Heuser09] HEUSER, C. A.. **Projeto de Banco de Dados, 6a edição**. Bookman, 2009. 3
- [Khanacademy] Disponível em: <https://www.khanacademy.org/>; Acessado em 19 de jan. de 2014. 1
- [Larman04] LARMAN, C.. **Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development**. Prentice Hall, 2004. 3
- [Leite03] JULIO CESAR SAMPAIO DO PRADO LEITE, J. H. D.. **Perspectives on Software Requirements**. Kluwer Academic Publishers, 2003. 2.3.1

[Postgres] Disponível em: <http://www.postgresql.org/docs/9.1/static/>;
Acessado em 19 de jan. de 2014. 3.3

[SalmanKhan] Disponível em: http://pt.wikipedia.org/wiki/Salman_Khan;
Acessado em 19 de jan. de 2014. 2.1.2

[Ullman] Disponível em: <http://infolab.stanford.edu/~ullman/>; Acessado
em 19 de jan. de 2014. 2.1.3