

UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE CIÊNCIAS MATEMÁTICAS E COMPUTAÇÃO

GRUPO 7

LUÍZA RUBIO - 10786121

LUCAS GABRIEL MENDES MIRANDA - 10265892

LEONARDO FONSECA PINHEIRO - 11219241

CESAR HENRIQUE DE ARAUJO GUIBO - 11218705

LUCA FARIA CURCIO - 10295502

TRABALHO DE SSC0124 – PARTE 1

SÃO CARLOS – SP

2020

SUMÁRIO

| | |
|--|-----------|
| 1 APRESENTAÇÃO GERAL DO SISTEMA | 3 |
| 1.1 ESTUDANTES | 3 |
| 1.2 EDUCADORES | 3 |
| 1.3 INFORMAÇÕES ADICIONAIS | 4 |
| 2 ANÁLISE DE REQUISITOS DO SISTEMA | 4 |
| 2.1 CASOS DE USO | 4 |
| 2.1.1 Alterações de termos e expressões | 4 |
| 2.1.2 Agrupamentos de casos de uso | 5 |
| 2.1.3 Adição de casos de uso | 5 |
| 2.1.4 Remoção de casos de uso | 6 |
| 2.1.5 Descrição textual de casos de uso críticos | 6 |
| 2.1.5.1 Assistir aula online | 6 |
| 2.1.5.2 Criar uma disciplina | 7 |
| 2.1.5.3 Disponibilizar aula de uma disciplina | 8 |
| 2.2 CENÁRIOS DE QUALIDADE | 8 |
| 2.2.1 Disponibilidade | 8 |
| 2.2.2 Interoperabilidade | 9 |
| 2.2.3 Segurança | 10 |
| 2.2.5 Inteligibilidade | 10 |
| 2.2.6 Acessibilidade | 11 |
| 2.2.7 Comportamento em relação ao tempo | 11 |
| 3 MODELO DA ARQUITETURA | 12 |
| 3.1 COMPONENTE DE GESTÃO | 12 |
| 3.2 COMPONENTE DE CONTEÚDO | 12 |
| 3.3 COMPONENTE DE USUÁRIO | 13 |
| 3.4 COMPONENTE DE ACESSO A DADOS DE USUÁRIOS | 13 |
| 3.5 COMPONENTE DE ACESSO A CONTEÚDO | 13 |
| 4 MODELO CONCEITUAL DO SISTEMA | 13 |
| 4.1 COMPONENTE DE GESTÃO | 13 |
| 4.2 COMPONENTE DE CONTEÚDO | 14 |
| 4.3 COMPONENTE DE USUÁRIOS | 15 |
| 5 DISCUSSÕES FINAIS | 16 |
| 6 REFERÊNCIAS BIBLIOGRÁFICAS | 16 |

1 APRESENTAÇÃO GERAL DO SISTEMA

Com a pandemia de COVID-19, ficou ainda mais evidente a necessidade de aplicações capazes de auxiliar educadores e alunos a manterem suas atividades educacionais à distância. Nesse documento, e em seus anexos, é descrito o funcionamento de uma dessas aplicações. A *Ensinet*, como é chamada, é uma aplicação web que tem como objetivo disponibilizar conteúdo educacional produzido por um educador. Para melhor detalhar o seu funcionamento, consideremos os dois principais tipos de usuário do sistema: *estudantes* e *educadores*.

1.1 ESTUDANTES

Carlos é um estudante universitário de 20 anos de idade que está sempre interessado em aprender coisas novas. Apesar de estar matriculado no curso de Ciências da Computação, ele também gosta de estudar outros assuntos, como História e Sociologia. Com a pandemia de COVID-19, foi forçado a continuar seus estudos a partir de casa. Por esse motivo, surgiu a necessidade de acompanhar as disciplinas que estava cursando (e as aulas dos outros assuntos que ele tem interesse) através de uma plataforma online. Dessa forma, ele pode se cadastrar na Ensinet como um *estudante*.

Estudantes são os usuários que consomem o conteúdo educacional disponível no site. Um estudante pode:

- Se matricular em disciplinas públicas ou privadas através de um código disponibilizado pelo educador responsável;
- Visualizar e consumir o conteúdo de disciplinas públicas ou de disciplinas privadas em que estão matriculados.

Nota-se que, quando um estudante está matriculado em uma disciplina, ele tem acesso a conteúdo adicional, que consiste de provas e trabalhos. Além disso, estudantes matriculados recebem uma pontuação de 0 a 10, atribuída pelo educador, que corresponde ao seu rendimento na disciplina em questão.

1.2 EDUCADORES

Isabela é uma professora que precisa disponibilizar suas aulas em uma plataforma online. Ela precisa publicar conteúdo gravado, transmitir aulas ao vivo e organizar trabalhos e provas para a avaliação de seus alunos. Assim como Carlos, ela também pode se cadastrar na Ensinet, mas como uma educadora.

Educadores são usuários, que além de possuírem as funcionalidades de alunos, são responsáveis por distribuir conteúdo educacional através da plataforma. Tal conteúdo pode consistir de:

- Videoaulas gravadas;
- Videoaulas ao vivo;
- Trabalhos;
- Provas.

O conteúdo disponibilizado por um educador deve ser agrupado em disciplinas. Por exemplo: se um educador deseja publicar uma vídeoaula sobre autovetores, ele deve associá-la à disciplina de Álgebra Linear, que deve ter sido criada por ele anteriormente. Um educador deve poder criar várias disciplinas e escolher se cada uma delas será pública ou privada. Disciplinas públicas estarão disponíveis a todos os usuários do site, enquanto que disciplinas privadas estarão disponíveis apenas para usuários matriculados nela.

1.3 INFORMAÇÕES ADICIONAIS

Considerando o cenário de uso apresentado, pode-se sintetizar o sistema como uma plataforma semelhante à Netflix, com as áreas do conhecimento (como Geografia, Biologia e Matemática) sendo equivalentes aos gêneros de filme. Ou seja, uma plataforma que junta videoaulas, aulas de dúvida ou qualquer outro tipo de vídeo educacional da matéria e disponibiliza para o usuário de forma rápida e de fácil acesso.

Os vídeos serão armazenados na nuvem, sendo esse armazenamento feito com o auxílio de serviços como o *Amazon Web Services*, por exemplo. A transmissão deles será feita por streaming, inicialmente sem legendas, sendo possível realizar downloads.

2 ANÁLISE DE REQUISITOS DO SISTEMA

Nas subseções abaixo, vamos descrever os modelos de casos de uso e de cenários de qualidade do sistema.

2.1 CASOS DE USO

A imagem com o modelo de casos de uso (*ensinet.png*) se encontra em anexo, na raiz do arquivo .zip.

2.1.1 Alterações de termos e expressões

A diferença mais básica desse diagrama para o outro é puramente semântica: alguns termos foram alterados para sinônimos ou outros termos com significado mais compatível com o que o caso de uso realmente representa. Alguns exemplos são:

- O termo “aluno” foi trocado para “estudante”;
- O termo “transmissão ao vivo” foi trocado para “livestream”;
- O termo “visualizar conteúdo” foi trocado para “listar informações”.

Isso foi feito com o objetivo de unificar a linguagem de todos os documentos que compõem o projeto e diminuir ambiguidades.

2.1.2 Agrupamentos de casos de uso

Alguns casos de uso da versão anterior foram agrupados, e passaram a possuir relacionamentos <<extend>> de casos de uso criados na nova versão. Abaixo, explicitamos algumas instâncias em que isso ocorreu:

- Os casos de uso *“avaliar conteúdo”* foi criado, possuindo relacionamentos <<extend>> com *“avaliar uma aula positivamente”* e *“avaliar uma aula negativamente”*;
- O caso de uso *“responder conteúdo”* foi criado, possuindo relacionamentos <<extend>> com *“fazer provas”* e *“submeter trabalho”*;
- O caso de uso *“listar informações do usuário logado”* foi criado, possuindo relacionamentos <<extend>> com *“listar disciplinas matriculadas”* e *“listar informações de cadastro”*;
- O caso de uso *“listar informações de uma disciplina”* foi criado, possuindo relacionamentos <<extend>> com *“listar notas do usuário em uma disciplina”*, *“listar livestreams de uma disciplina”*, *“listar aulas de uma disciplina”* e *“listar material de apoio de uma disciplina”*;
- O caso de uso *“listar informações de um educador”* foi criado, possuindo relacionamentos <<extend>> com *“listar disciplinas de um educador”* e *“listar informações básicas”*.

Isso foi feito para melhor agrupar alguns casos de uso, de modo a criar processos mais bem definidos, com começo, meio e fim.

2.1.3 Adição de casos de uso

Alguns casos de uso foram identificados e adicionados ao diagrama. Alguns desses casos de uso são:

- *“Alterar informações de cadastro”*, do ator *“estudante”*, pois foi identificado que seria interessante que esse ator tivesse a capacidade de alterar as informações fornecidas no cadastro, como senha, por exemplo;
- *“Submeter trabalhos”*, do ator *“estudante”*, para que esse ator tenha a capacidade de submeter trabalhos para a correção do educador.

2.1.4 Remoção de casos de uso

Alguns casos de uso foram removidos, tais como:

- *“Assistir aulas de disciplinas públicas” e “assistir transmissões ao vivo de disciplinas públicas”, do ator “visitante”. Isso foi feito porque esses casos de uso foram identificados como redundantes, visto que já existe o caso de uso “assistir conteúdo”, do ator “estudante”.*
- *“Tirar dúvidas de uma aula”, do ator “estudante”, que também é redundante, por causa da existência do caso de uso “trocar mensagens com um educador”.*

2.1.5 Descrição textual de casos de uso críticos

Nas tabelas de 1 a 3, vamos apresentar a descrição textual concreta dos casos de uso mais críticos para o funcionamento do sistema.

2.1.5.1 Assistir aula online

Tabela 1 - Caso de uso “assistir aula online”

| | |
|-------------------------------------|--|
| Nome do caso de uso | Assistir aula online |
| Ator principal | Usuário do sistema |
| Atores secundários | Educador |
| Resumo | Usuário entra no sistema e consegue acessar aula para assistir |
| Pré-condições | Disciplina ser criada, aula ser disponibilizada pelo educador no sistema e usuário possuir acesso a essa disciplina (acontece quando ela é pública ou quando o estudante está matriculado nela). |
| Pós-condições | Sistema armazena que a aula foi assistida pelo usuário e sugere a próxima aula |
| Cenário de Sucesso Principal | 1. Estudante realiza o login no site 2. Estudante acessa a página da disciplina 3. Estudante clica na aula que deseja e a assiste |
| | 1. Visitante acessa a página de uma disciplina pública 2. Visitante clica na aula que deseja e a assiste |

| | |
|------------------------------|--|
| | <ol style="list-style-type: none"> 1. Educador realiza o login no site 2. Educador acessa a página da disciplina 3. Educador clica na aula que deseja e a assiste |
| | <ol style="list-style-type: none"> 1. Usuário acessa a página de uma disciplina 2. Sistema exibe mensagem informando o aluno que nenhuma aula foi adicionada à disciplina |
| Restrições/Validações | O usuário precisa estar matriculado na disciplina em questão, ou ela precisa ser pública. |

2.1.5.2 Criar uma disciplina

Tabela 2 - Caso de uso “criar uma disciplina”

| | |
|-------------------------------------|--|
| Nome do caso de uso | Criar uma disciplina |
| Ator principal | Educador |
| Atores secundários | Usuário do sistema |
| Resumo | Educador entra no sistema e consegue criar uma disciplina |
| Pré-condições | Usuário que quer criar a disciplina precisa ser um educador |
| Pós-condições | Sistema armazena a disciplina que foi criada pelo educador |
| Cenário de Sucesso Principal | <ol style="list-style-type: none"> 1. Educador realiza login no site 2. Educador clica no botão de nova disciplina 3. Educador dá um nome e uma descrição à nova disciplina 4. Educador clica no botão de criar disciplina |
| Restrições/Validações | N/A |

2.1.5.3 Disponibilizar aula de uma disciplina

Tabela 3 - Caso de uso “disponibilizar aula de uma disciplina”

| | |
|-------------------------------------|--|
| Nome do caso de uso | Disponibilizar aula de uma disciplina |
| Ator principal | Educador |
| Atores secundários | Usuário do sistema |
| Resumo | Educador entra no sistema e consegue adicionar aula a uma disciplina |
| Pré-condições | Usuário que quer criar a aula precisa ser um educador |
| Pós-condições | Sistema armazena a aula que foi criada pelo educador |
| Cenário de Sucesso Principal | <ul style="list-style-type: none">5. Educador realiza login no site6. Educador clica no botão da disciplina em que deseja adicionar a aula7. Educador clica no botão de adicionar aula8. Educador dá um nome e uma descrição à nova aula9. Educador faz o upload do vídeo que corresponde à aula10. Educador clica no botão de criar disciplina |
| Restrições/Validações | N/A |

2.2 CENÁRIOS DE QUALIDADE

Abaixo temos alguns atributos de qualidade que serão importantes para o projeto. Para cada um desses atributos, foi especificado um ou mais cenários correspondentes. Note que cada atributo corresponde a uma característica ou subcaracterística especificada na ISO/IEC 25010.

2.2.1 Disponibilidade

Trata-se de uma subcaracterística especificada na ISO/IEC 25010. Diz respeito ao grau em que um sistema se encontra operacional quando requisitado para uso. A tabela 4 descreve um cenário de qualidade concreto para esse atributo.

Tabela 4 - Cenário de qualidade para o atributo de disponibilidade

| | |
|---------------------------|--|
| Fonte | Processos do cliente |
| Estímulo | Conexão não estável à internet |
| Artefato | Processos |
| Ambiente | Em execução. |
| Reposta | Servir apenas os dados armazenados localmente em cache pelo usuário final |
| Medida da resposta | Dados em cache apresentam uma interface funcional para o usuário enquanto não se consegue realizar o download dos dados necessários. |

Esse cenário estava presente na versão anterior do modelo. Ele é importante para garantir que usuários do sistema com conexão não estável com a internet tenham uma experiência mais próxima do normal.

2.2.2 Interoperabilidade

Outra subcaracterística especificada na ISO/IEC 25010. Diz respeito ao grau em que o sistema é capaz de se comunicar com outros sistemas fundamentais para seu funcionamento (como os SGBDs, por exemplo). A tabela 5 descreve um cenário de qualidade concreto para esse atributo.

Tabela 5 - Cenário de qualidade para o atributo de interoperabilidade

| | |
|---------------------------|--|
| Fonte | Processos do cliente |
| Estímulo | Conteúdo disponível sofre alguma alteração |
| Artefato | Banco de dados |
| Ambiente | Em execução. |
| Reposta | Armazenar as mudanças corretamente e começar a responder <i>queries</i> com os dados mais atualizados. |
| Medida da resposta | Após algum dado ter sido editado, o próximo get request realizado deve ser respondido com a versão mais recente dos dados. |

Esse cenário também estava presente na versão anterior do modelo. Ele é importante para garantir que usuários do sistema recebam os dados mais atualizados do sistema.

2.2.3 Segurança

Uma característica especificada na ISO/IEC 25010. Diz respeito ao grau em que o sistema protege os dados com os quais está lidando. A tabela 6 descreve um cenário de qualidade concreto para esse atributo.

Tabela 6 - Cenário de qualidade para o atributo de segurança

| | |
|---------------------------|--|
| Fonte | Usuário ou administrador com intenções maliciosas |
| Estímulo | Tentativa não autorizada de remover ou alterar dados disponibilizados por outro usuário. |
| Artefato | Banco de dados |
| Ambiente | Em execução. |
| Reposta | Impedir a operação e impedir que o usuário acesse novamente o sistema. |
| Medida da resposta | Existem poucas reclamações dos usuários a respeito da segurança de seus dados. |

Esse cenário também estava presente na versão anterior do modelo. Ele é importante para garantir que usuários tenham os seus dados protegidos e possam confiar no sistema.

2.2.5 Inteligibilidade

Uma subcaracterística especificada na ISO/IEC 25010. Diz respeito ao grau de facilidade com que o usuário consegue aprender a utilizar o sistema. A tabela 7 descreve um cenário de qualidade abstrato para esse atributo.

Tabela 7 - Cenário de qualidade para o atributo de inteligibilidade

| | |
|-----------------|--|
| Fonte | Usuário |
| Estímulo | Usuário acessa o sistema pela primeira vez |
| Artefato | O sistema todo |

| | |
|---------------------------|---|
| Ambiente | Em execução. |
| Reposta | O sistema apresenta uma organização que permite que o usuário aprenda a utilizá-lo eficientemente |
| Medida da resposta | Existem poucas dúvidas dos usuários a respeito de como utilizar o sistema |

Esse cenário não estava presente na versão anterior do modelo. Ele é importante para garantir que os usuários não desistam do sistema durante um primeiro contato. Se o sistema for difícil de se utilizar e de se aprender a utilizar, as chances dos usuários migrarem para outras plataformas aumentam.

2.2.6 Acessibilidade

Outra subcaracterística especificada na ISO/IEC 25010. Diz respeito ao grau de facilidade com que usuários com necessidades especiais conseguem utilizar o sistema. A tabela 8 descreve um cenário de qualidade abstrato para esse atributo.

Tabela 8 - Cenário de qualidade para o atributo de acessibilidade

| | |
|---------------------------|--|
| Fonte | Usuário com necessidades especiais |
| Estímulo | Usuário tenta utilizar o sistema eficientemente |
| Artefato | O sistema todo |
| Ambiente | Em execução. |
| Reposta | O sistema apresenta características que permita que o usuário com necessidades especiais o utilize sem grandes problemas |
| Medida da resposta | Existem usuários com necessidades especiais utilizando o sistema, e poucas reclamações desses usuários |

Esse cenário também não estava presente na versão anterior do modelo. Ele é importante pois a ideia da plataforma é facilitar o aprendizado de *todas* as pessoas, sem exceções. Um sistema não acessível implica em abrir mão de parte do público-alvo.

2.2.7 Comportamento em relação ao tempo

Outra subcaracterística especificada na ISO/IEC 25010. Mede o tempo em que o sistema leva para processar suas funções. A tabela 9 descreve um cenário de qualidade concreto para esse atributo.

Tabela 9 - Cenário de qualidade para o atributo de comportamento em relação ao tempo

| | |
|---------------------------|---|
| Fonte | Usuários |
| Estímulo | Múltiplos usuários acessam o sistema simultaneamente |
| Artefato | O sistema todo |
| Ambiente | Em execução. |
| Resposta | O sistema responde a todos os usuários em um tempo razoável |
| Medida da resposta | Existem poucas reclamações dos usuários a respeito de lentidões no sistema; o tempo que o sistema demora para processar requisições de múltiplos usuários é pequeno |

Esse cenário também não estava presente na versão anterior do modelo. Ele é importante pois pretende-se que vários usuários utilizem o sistema ao mesmo tempo. Desse modo, é necessário que o sistema responda a todos dentro de um intervalo de tempo aceitável.

3 MODELO DA ARQUITETURA

A imagem com o modelo se encontra em anexo na raiz do arquivo .zip, com o nome de *componentes.png*.

3.1 COMPONENTE DE GESTÃO

O componente de gestão tem como responsabilidade conter as funcionalidades relativas à administração do sistema e do conteúdo, tais como adição/remoção de novos usuários, escolher se uma nova disciplina vai ser pública ou privada, etc. Além disso, ele tem como dependência os componentes de acesso aos dados de usuários e de acesso ao conteúdo, uma vez que os utiliza para acesso ao banco de dados.

3.2 COMPONENTE DE CONTEÚDO

O componente de conteúdo tem como responsabilidade a adição, remoção ou alteração de qualquer coisa do programa que tenha relação com o conteúdo apresentado, cuidando de uma variedade de coisas, que vai desde a criação de uma disciplina, até a remoção de uma prova. Além disso, ele tem como dependência os componentes de acesso aos dados de usuários e de acesso ao conteúdo, uma vez que utiliza-os para acesso ao banco de dados.

3.3 COMPONENTE DE USUÁRIO

O componente de usuário têm como responsabilidade prover as funcionalidades que abrangem o uso comum do sistema, ou seja as ações que não provocam emoções, nem adições de registros nas bases de dados de usuário ou conteúdo, mas, no máximo, atualizações. Tem como dependência os componentes de acesso aos dados de usuários e de acesso ao conteúdo, uma vez que utiliza-os para acesso ao banco de dados.

3.4 COMPONENTE DE ACESSO A DADOS DE USUÁRIOS

O componente de acesso a dados de usuários têm como responsabilidade prover ao sistema funcionalidades que gerenciam a comunicação do sistema com a base de dados de usuários. Essa última contém informações gerais sobre os usuários, como nome completo, nome de usuário, idade, etc. Cabe notar, também que ele independe dos outros componentes explícitos.

3.5 COMPONENTE DE ACESSO A CONTEÚDO

O componente de acesso a conteúdo têm como responsabilidade prover ao sistema funcionalidades que gerenciam a comunicação do sistema com a base de dados de conteúdo. Essa última armazena o conteúdo (vídeos, provas e trabalhos) oferecido pela aplicação. Cabe notar, também que ele independe dos outros componentes explícitos.

4 MODELO CONCEITUAL DO SISTEMA

4.1 COMPONENTE DE GESTÃO

A versão inicial do modelo se encontra em *gestao.png*. Abaixo, elaboramos alguns detalhes sobre as suas classes.

```
class GerenciarEstudantes{  
    string Nome;  
    integer Identificador;  
    array DisciplinasMatriculadas;  
    array HistóricoNotas;  
}
```

A classe `GerenciarEstudantes` permite a execução de funções administrativas em relação aos estudantes, e garante acesso ao seu nome (String), número Identificador (Integer), lista de disciplinas matriculadas (Array) e histórico de notas (Array).

```
class GerenciarDisciplinas{
    string Nome;
    integer Identificador;
    string DocenteResponsável;
    list AlunosMatriculados;
    array Horários;
}
```

A classe Gerenciar disciplinas é responsável por garantir acesso a todas as disciplinas criadas, e informações pertinentes a elas, permitindo a edição dessas informações: Nome da disciplina (String), ID da disciplina (Integer), professor responsável pela disciplina (String), lista de alunos matriculados atualmente (List), tabela com horários e datas de aulas da disciplina (Array).

4.2 COMPONENTE DE CONTEÚDO

```
class CriarDisciplina{
    String nome;
    Integer participantes;
    String educador;
}
```

A classe CriarDisciplina é responsável por criar uma disciplina que contenha os atributos: “nome”, string que contém o nome da disciplina; “participantes”, inteiro que contém o número de participantes máximo; “educador”, string que contém o nome do educador responsável. A classe CriarDisciplina é mãe das classes LiveStream, DisponibilizarMaterial e RemoverMaterial. Apenas os educadores ou administradores do sistema tem autorização para criar disciplinas.

```
class LiveStream{
    String horario;
    String conteudo;
}
```

A classe LiveStream é filha da classe CriarDisciplina, por conseguinte ela herda todos os seus atributos “nome”, “participantes” e “educador”. Além destes ela possui também os atributos “horario”, string que contém o horário da aula ao vivo e o atributo “conteudo”, string com o nome do conteúdo que será ensinado durante a aula. Apenas os educadores ou administradores do sistema tem autorização para criar livestreams.

```
class DisponibilizarMaterial{
    String tipo;
    String dataDePostagem;
    String dataDeEntrega;
    String titulo;
```

```
}
```

A classe `DisponibilizarMaterial` é utilizada para adicionar conteúdos as disciplinas, ela possui os atributos: “tipo”, uma string que pode ser “prova”, “aula” ou “material de apoio”; “dataDePostagem”, string que contém a data que o material foi adicionado ao site; “dataDeEntrega”, string que contém a data limite para a atividade ser realizada. A classe `DisponibilizarMaterial` é filha da classe `CriarDisciplina`, assim ela herda todos os seus atributos “nome”, “participantes” e “educador”. Apenas os educadores ou administradores do sistema tem autorização para disponibilizar materiais.

```
class RemoverMaterial{
    String tipo;
    String titulo;
}
```

A classe `RemoverMaterial` auxilia na funcionalidade de remover um determinado material, o atributo “tipo” é uma string que identifica se é uma “prova”, “aula” ou “material de apoio” e o atributo “titulo”, é uma string que identifica o nome do documento a ser removido. Ela também é filha da classe `CriarDisciplina`, então ela herda todos os seus atributos “nome”, “participantes” e “educador”. Apenas os educadores ou administradores do sistema tem autorização para remover materiais.

4.3 COMPONENTE DE USUÁRIOS

A versão inicial do modelo se encontra em *usuarios.png*. Abaixo, elaboramos alguns detalhes sobre as suas classes.

```
class RealizarCadastro{
    string Nome;
    string e-Mail;
    string Senha;
    integer Identificador;
}
```

A classe `RealizarCadastro` permite que um visitante realize um cadastro no site fornecendo as informações obrigatórias: nome completo (String), email usado para login (String), senha de acesso ao site (String), e ao final desse processo o novo usuário terá um ID atribuído ao seu cadastro (Integer).

```
class AssistirConteudo{
    list Videoaulas;
    list Livestreams;
```

```
}
```

A classe `AssistirConteudo` permite que um estudante visualize qualquer videoaula que esteja disponível no sistema ou assista livestreams que estão em andamento.

```
class BuscarConteudo{  
    string StringdeBusca;  
    integer TipoConteudo;  
}
```

A classe `BuscarConteudo` permite que um estudar faça uma busca utilizando qualquer keywords que desejar (`String`), e pode selecionar se está buscando uma disciplina, aula, ou educador. Essa seleção vai definir o `Integer TipoConteudo`.

5 DISCUSSÕES FINAIS

Durante o projeto, algumas etapas se mostraram mais desafiadoras do que outras. De início, foi difícil decidir uma ideia concreta para o projeto e, dentre algumas opções, gostamos da Ensinet.

Depois disso, a próxima dificuldade a ser ultrapassada foi na arquitetura do sistema, onde tivemos algumas dificuldades para formular a melhor forma de representar efetivamente nosso projeto. Contudo, foi possível superá-la por meio de discussões e união entre o grupo.

Fora isso, foi possível navegar o projeto com uma certa facilidade, com todos os integrantes do grupo contribuindo ao máximo e ajudando a lidar com as dificuldades. Além disso, foi importante o uso da comunicação com a professora, que possibilita a resolução de qualquer dúvida em alguma etapa.

6 REFERÊNCIAS BIBLIOGRÁFICAS

Len Bass, Paul Clements, and Rick Kazman. 2012. *Software Architecture in Practice* (3rd. ed.). Addison-Wesley Professional.

ISO/IEC 25010:2011 [ISO/IEC 25010:2011]. Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models. Disponível em: <<https://www.iso.org/standard/35733.html>>. Acesso em: 14/10/2020