

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA –
CAMPUS CAJAZEIRAS

CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS

**EDUCAÇÃO MOBILE – APRENDER SEMPRE, EM QUALQUER LUGAR, COM
QUALQUER DISPOSITIVO**

FÁTIMA FERREIRA DE SOUSA

CAJAZEIRAS – PB

2015

FÁTIMA FERREIRA DE SOUSA

**EDUCAÇÃO MOBILE – APRENDER SEMPRE, EM QUALQUER LUGAR, COM
QUALQUER DISPOSITIVO**

Documento apresentado a disciplina de Trabalho de Conclusão de Curso I, como requisito básico para a apresentação do Trabalho de Conclusão de Curso do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas.

ORIENTADOR: Prof. Esp. Janderson Ferreira Dutra.

CAJAZEIRAS – PB

2015

LISTA DE FIGURAS

Figura 1 - Síntese de Fluxo do Processo YP	17
Figura 2 - Desenho da Arquitetura Geral do Sistema.....	23
Figura 3 - Diagrama de Caso de Uso.....	24
Figura 4 - Diagrama de Classes.....	28
Figura 5 - Diagrama de Componentes	29
Figura 6 - Diagrama Entidade-Relacionamento	31
Figura 7 - Diagrama Lógico dos Dados	32

LISTA DE QUADROS

Quadro 1 - Cronograma	12
Quadro 2 - Requisitos Funcionais	20
Quadro 3 - Requisitos Não Funcionais	21
Quadro 4 - Detalhamento das User Stories	37
Quadro 5 - Caso de Uso Login	42
Quadro 6 - Caso de Uso Manter Exercício	42
Quadro 7 - Caso de Uso Manter Questão	43
Quadro 8 - Caso de Uso Manter Arquivo	43
Quadro 9 - Caso de Uso Manter Disciplina	44
Quadro 10 - Caso de Uso Comentar Tópico	44
Quadro 11 - Caso de Uso Participar de Grupo	45
Quadro 12 - Caso de Uso Download de Arquivo	45
Quadro 13 - Caso de Uso Responder Exercícios	45
Quadro 14 - Caso de Uso Manter Grupo de Discussão	46
Quadro 15 - Caso de Uso Manter Tópico	46

LISTA DE SIGLAS

HTML	HyperText Markup Language
M-Learning	Mobile Learning
YP	easYProcess
SGBD	Sistema Gerenciador de Banco de Dados
TCC	Trabalho de Conclusão de Curso

SUMÁRIO

1. Introdução.....	1
1.1. Motivação.....	2
1.2. Objetivos	3
1.3. Organização do texto.....	4
2. Fundamentação teórica.....	5
2.1. Computação ubíqua.....	5
2.2. Mobile learning	6
2.3. Objetos de aprendizagem	8
3. Metodologia	11
3.1. Atividades	12
3.2. Cronograma.....	12
3.2.1. Ferramentas e tecnologias	13
3.2.1.1. Ferramentas	13
3.2.1.2. Linguagens	14
3.2.1.3. Tecnologias	15
3.2.2. easYprocess – YP.....	16
4. Desenvolvimento do projeto	18
4.1. Apresentação do sistema	18
4.1.1. Módulo de exercícios	18
4.1.2. Módulo de transferência de arquivos	19
4.1.3. Módulo de criação de fóruns de discussões.....	19
4.1.4. Módulo de envio de mensagens	19
4.2. Especificação.....	20
4.2.1. Requisitos funcionais	20
4.2.2. Requisitos não-funcionais	21
4.3. Análise e projeto.....	22
4.3.1. Diagrama de caso de uso	23
4.3.2. Diagrama de classes	27
4.3.3. Diagrama de componentes	29
4.3.4. Projeto de banco de dados – diagramas.....	30
4.3.4.1. Diagrama entidade-relacionamento.....	30
4.3.4.2. Modelo lógico do sistema	32
5. Conclusão	33
6. Referências bibliográficas.....	34

1. INTRODUÇÃO

“As tecnologias de computação móvel encontram-se atualmente em franca evolução e parecem destinadas a transformar-se no novo paradigma dominante da computação” (Myers et al., 2003). Esse novo paradigma computacional aberto, dinâmico e adaptável surge principalmente com o objetivo de desenvolver soluções que permitam o acesso e o compartilhamento de informações em qualquer lugar, e a qualquer momento, de forma contínua, contextualizada e integrada ao cotidiano das pessoas.

A educação, assim como as outras áreas de conhecimento, faz uso dessas novas tecnologias para melhorar suas práticas e abordagens. A aplicação dessas tecnologias no aperfeiçoamento das estratégias de educação ocasionou o surgimento de uma frente de pesquisa denominada *Mobile Learning*.

A tecnologia *M-Learning* é um campo emergente, que engloba o uso de equipamentos portáteis, num novo cenário de “computação pervasiva” pautada pela mobilidade global do usuário. “Seu grande potencial encontra-se na utilização da tecnologia móvel como parte de um modelo de aprendizado integrado caracterizado principalmente pelo uso de dispositivos de comunicação sem fio de forma transparente e com alto grau de mobilidade” (Ahone, 2003; Syvanen, 2003), permitindo que a aprendizagem possa ocorrer sem hora e local pré-estabelecido, maximizando a liberdade dos alunos.

Neste contexto o uso das tecnologias móveis se apresenta como uma ideia inovadora e entusiasmante pelo o fato de o mesmo servir de auxílio para que os usuários possam buscar conhecimento, agregando a portabilidade e a vasta disponibilidade de informações relacionadas ao ensino, por parte de professores e alunos, e no âmbito acadêmico ressalta-se sua importância por despertar nos profissionais da área computacional o interesse no desenvolvimento de aplicações voltadas para dispositivos móveis direcionados à educação.

Mediante ao cenário exposto, é apresentado uma proposta de aplicação educacional para dispositivos móveis que permite a comunicação e troca de informações entre alunos e professores de forma rápida e prática, tornando-se uma alternativa interessante de modelo de aplicação em *M-Learning*.

1.1. MOTIVAÇÃO

A informática revolucionou a forma de processar e transmitir conhecimento. Todos os dias um demasiado volume de informações está sendo colocado a nossa frente, e o problema que surgiu, no entanto, é como assimilar todos esses dados, é preciso transformar esse grande volume de informação em conhecimento de forma que facilite o aprendizado dos alunos, porém muitas vezes, os mecanismos tradicionais de ensino se tornam estressantes tanto para os alunos, quanto para os professores.

No sistema educacional atual, principalmente no ensino fundamental e médio a função do professor é basicamente copiar e repassar o conteúdo de livros, sem acrescentar experiências novas. Como consequência, uma aula, muitas vezes se torna algo estressante e cansativo, fazendo com que o aluno sinta-se desmotivado e não aproveite adequadamente o tempo que passa com o professor.

Almeida (2001, apud BRASIL, 2007) afirma que “diante do atual contexto escolar brasileiro, os educadores necessitam de alternativas pedagógicas que auxiliem o processo de ensino/aprendizagem de forma mais eficiente”. Assim, fica claro, que a sociedade necessita de ferramentas que agreguem valor ao processo de ensino, de modo que os conteúdos sejam facilmente absorvidos pelos aprendizes.

Para evitar a sobrecarga cognitiva de alunos e tornar o ambiente escolar mais intuitivo algumas metas estão sendo adotadas, entre elas o uso das tecnologias de aprendizagem interativas mediadas pelo uso de computadores, permitindo assim a abordagem de diversas áreas de conhecimento de forma dinâmica e interativa tanto para educadores quanto para educandos.

Porém o uso desses computadores gera algumas dificuldades, tais como falta de máquinas para todos os alunos, indisponibilidade, ou até mesmo desperdício de tempo com deslocamento dos alunos até laboratórios de informática. Além disso o aprendizado ainda continua restrito ao ambiente escolar.

Dentro dos conceitos de aprendizagem até então utilizados surge o desafio de apresentar melhorias que possam ajudar na construção do conhecimento de alunos, a qual permita aos mesmos acessar matérias, assistir aulas, interagir com colegas de qualquer lugar e a qualquer tempo, fazendo uso das tecnologias disponibilizadas em conjunto com o *M-Learning* incorporadas aos dispositivos móveis. Porque em toda história das tecnologias da educação

nenhuma tecnologia esteve tão amplamente acessível aos cidadãos como a tecnologias móveis (Nix, etal. 2005).

1.2. OBJETIVOS

Para que exista uma aprendizagem continuada fora do ambiente escolar com qualidade faz-se necessário o uso de ferramentas altamente interativas, que possibilitem a criação de um ambiente virtual móvel ativo e aberto a aprendizagem, no qual estudantes possam trabalhar com seus interesses individuais e coletivos visando desenvolver suas habilidades, conhecimentos e interesses. Desta forma o *M-Learning* surge como uma importante alternativa de ensino e treinamento que contribui de maneira satisfatória para o ensino.

Motivado por essa grande aceitação das tecnologias móveis, principalmente por jovens, nosso projeto elegeu um objetivo geral a ser alcançado:

- Desenvolver um aplicativo que auxilie no processo de ensino-aprendizagem, através da consulta e troca de informações agregado a mobilidade dos dispositivos computacionais móveis apoiados pelo *M-Learning*. Tornando esse processo mais dinâmico e completo, e proporcionando aos alunos um ambiente informal de aprendizagem.

Para alcançar o objetivo geral, foram propostos os seguintes objetivos específicos:

- Apoiar o uso de tecnologia digital na educação por meio do objeto desenvolvido;
- Maximizar os recursos de aprendizagem dos alunos, permitindo aos mesmos contar com apoio de dispositivos computacionais para execução de tarefas, anotações de ideias, consulta e compartilhamento de informações via Internet;
- Prover acesso ao conteúdo didático de forma dinâmica, incremental, fazendo uso de dispositivos móveis, de acordo com a conectividade do dispositivo.
- Expandir a aprendizagem fora do ambiente escolar.
- Fornecer meios para o desenvolvimento de métodos inovadores de ensino, através dos novos recursos da computação móvel.

Além disso, este trabalho também preocupa-se em apresentar e impulsionar o desenvolvimento de aplicações que colocam em prática a questão da produção de conteúdos educacionais para dispositivos móveis, já que está é uma área de estudo inovadora e entusiasmante.

1.3. ORGANIZAÇÃO DO TEXTO

Este texto está organizado em capítulos, dos quais este é o primeiro e apresenta a ideia do sistema, incluindo os objetivos e a justificativa.

No Capítulo 2 está exposta o referencial teórico em que a análise se apoiou. Nele, são apresentadas as ideias principais dos expoentes de cada conceito que interessam mais de perto a esta investigação.

No Capítulo 3 está o método, que é a sequência geral de passos para o ciclo de vida do sistema, e os materiais utilizados. Os materiais se referem ao que é necessário para modelar e implementar o sistema, incluindo as tecnologias, as ferramentas e os ambientes de desenvolvimento utilizados.

O Capítulo 4 contém uma descrição detalhada do sistema, descrevendo os requisitos funcionais e não-funcionais que o compõe, bem como o projeto arquitetural e a metodologia de desenvolvimento, esta são complementadas através de diagramas padronizados seguindo uma linguagem de modelação de terceira geração, a UML (*Unified Modeling Language*).

No último capítulo, o 5 são apresentadas algumas considerações a respeito da pesquisa.

2. FUNDAMENTAÇÃO TEÓRICA

A fundamentação teórica desse trabalho está centrada na computação ubíqua, na tecnologia *Mobile Learning* e nos Objetos de aprendizagem, tendo em vista que estes conceitos foram utilizados na documentação da análise de projeto.

2.1. COMPUTAÇÃO UBÍQUA

O mundo moderno vem sofrendo constantes modificações com a crescente expansão das tecnologias móveis. Estamos vivenciando em um contexto de computação ubíqua, no qual dispositivos computacionais estão incorporados no nosso cotidiano de tal forma que, nós inconscientemente, aproveitamos suas comodidades digitais para alcançar certos benefícios em nossa vida.

Pesquisa realizada em 2011 pelo Instituto Brasileiro de Geografia e Estatística (IBGE) indica que o percentual de brasileiros com telefone celular subiu de 36,6%, em 2005, para 69%, em 2011. Em números absolutos, o total passou de 55,7 milhões de pessoas para 115,4 milhões, um crescimento de 107,2%. A importância destes aparelhos vai além da sua capacidade computacional: está também na facilidade de uso e principalmente na sua portabilidade.

A computação ubíqua coloca a computação na periferia da vida dos usuários, como uma ferramenta, e não em foco, isto é, fora de seus caminhos, permitindo assim, o verdadeiro cumprimento das tarefas que de fato, eles necessitam ou desejam concluir. O termo “ubíquo” é usado para exprimir que tanto os computadores, como a computação estarão presentes em qualquer lugar, e embutidos nas estruturas de nossas vidas, ao contrário da realidade virtual onde as pessoas são inseridas em mundos gerados dentro do próprio computador. Em outras palavras, o objetivo da computação ubíqua é mover os computadores do foco central da atenção dos usuários para um mundo invisível, onde eles são usados subconscientemente, para aumentar a eficiência das ferramentas e meios de comunicação existentes.

Para Araújo (2003), a computação ubíqua surge da necessidade da unidade da mobilidade com as funcionalidades tecnológicas da computação pervasiva para dar a um dispositivo computacional que está em movimento à habilidade de, dinamicamente, configurar seus serviços de acordo com a necessidade ou com o ambiente em que nos movemos.

Para melhor compreender a definição de computação Ubíqua, deve-se também explorar os conceitos da Computação pervasiva que se define por possuir os meios de computação

distribuídos no ambiente de trabalho dos usuários de forma perceptível ou imperceptível. Por esta definição o computador seria capaz de detectar e extrair dados e variações do ambiente, gerando automaticamente modelos computacionais controlando, configurando e ajustando aplicações conforme as necessidades dos usuários e dos demais dispositivos.

2.2. MOBILE LEARNING

O número de aplicações distribuídas criadas para serem usadas na Internet aumentou consideravelmente nos últimos anos. Uma das mais complexas questões nestas aplicações diz respeito à mobilidade das entidades envolvidas na computação distribuída. Neste contexto, define-se mobilidade como a capacidade das entidades computacionais deslocar-se de um ambiente para outro, através do sistema distribuído (AVELLIS, 2003).

Como o uso de dispositivos móveis está se tornando onipresente em nosso cotidiano e no mundo do trabalho, seu uso na educação é inevitável. Criar ambientes virtuais móveis para que alunos e professores possam interagir de tal maneira que o aprendizado não fique restrito somente as salas de aulas, parece ser algo cada vez mais próximo do nosso cotidiano.

Segundo Reinhard et all. (2007):

“Aprender com mobilidade não é uma ideia nova – a possibilidade de aprender em qualquer lugar e a qualquer momento sempre foi buscada e potencializada com ferramentas como livros, cadernos e outros instrumentos móveis (portáteis) que existem há muito tempo. O que hoje ocorre é que as Tecnologias Móveis Sem Fio podem contribuir para Aprendizagem com Mobilidade por disponibilizarem aos sujeitos o acesso rápido a uma grande e diversificada quantidade de informações, viabilizando seu recebimento e envio (quando associados à Internet); além disso, essas tecnologias promovem a comunicação e a interação entre pessoas distantes geograficamente e temporalmente.”

Segundo Antônio (2003) atualmente, os estudos sobre mobilidade vêm sendo impulsionados pelo crescimento dos dispositivos eletrônicos e móveis (*tablet PCs, notebooks, handhelds*, e telefones celulares) e o crescimento de uma rede de comunicação sem fio (*WinMAX, Bluetooth, Global System for Mobile (GSM)* e *WiFi*). Como já mencionado anteriormente, esse novo paradigma é denominado computação móvel. Os avanços das tecnologias da informação e comunicação centrados na mobilidade vêm transformando a maneira de como o usuário “enxerga” a tecnologia (OLIVEIRA, 2007). *M-Learning* é um desenvolvimento do *E-Learning* pois utiliza dispositivos móveis, como *smartphones* e celulares, para a utilização e uma maior interação com o acesso a recursos pedagógicos e com

os objetos de aprendizagem, independentemente de onde o usuário estiver com exceção da cobertura da rede sem fio (AHONE et.al., 2003).

O autor Ahonen ainda complementa:

“Seu grande potencial encontra-se na utilização da tecnologia móvel como parte de um modelo de aprendizado integrado, caracterizado pelo uso de dispositivos de comunicação sem fio, de forma transparente e com alto grau de mobilidade” (AHONEN et al.; 2003)

De acordo com Barbosa:

A aprendizagem móvel numa perspectiva pedagógica, aponta para uma nova dimensão na educação ao poder de atender a necessidade de aprendizagem imediata, com grande flexibilidade e interatividade. (BARBOSA, 2007)

Nesse sentido, a utilização da tecnologia móvel como parte de um novo modelo de aprendizagem integrado, caracterizado pelo o uso da computação sem fio, vem potencializar e facilitar ainda mais o conceito de *Mobile Learning* ou *M-Learning*. Esse conceito foi proposto inicialmente por Ahone e Swanen (2003 apud Marcal et al 2005), para definir processos de aprendizagem que ocorrem, necessariamente, apoiados pelo uso de Tecnologias da Informação Móvel e Sem Fio (TIMS), tendo como característica fundamental a mobilidade de seus atores, permitindo aos mesmos aprender a qualquer hora e em qualquer lugar sem a necessidade de uma conexão física permanente a uma rede cabeada.

Outra definição para *M-Learning* é a de Luciano Pelissoli e Waldomiro Loyolla (2004), segundo os referidos autores, a “união de várias tecnologias de comunicação e processamento de dados que possibilitam a alunos e educadores uma interação mais eficaz, através de dispositivos móveis como notebooks, celulares, smartphones, etc.”.

Contudo, considerando a mobilidade do ponto de vista do aluno, do que da tecnologia, ela se torna mais importante, visto que aprendizagem vai a qualquer lugar. Por exemplo, alunos fazem uma revisão do conteúdo enquanto estão no ônibus a caminho da escola, médicos atualizam seus conhecimentos enquanto estão andando pelo hospital, estudantes de idioma melhoras suas habilidades enquanto viajam por outros países. Todos esses exemplos permitem que aprendizagem ocorra enquanto seus agentes se movem.

Portanto uma definição de *M-Learning* deveria ser ampliada para: “Qualquer tipo de aprendizagem que ocorre enquanto o estudante não está fixo a um local predeterminado, ou

ainda aprendizagem que acontece quando o estudante obtém vantagem das oportunidades de aprendizagem oferecidas pelas tecnologias móveis” (Barbosa, 2010).

O *M-Learning* deve ser utilizado com o intuito de fornecer conhecimento e não meramente informações, alcançando assim objetivos educacionais. Sabe-se que entre informação e conhecimento há um estreito relacionamento, porém não são semelhantes: conhecimento é a informação num contexto.

“O interessante é que na utilização das tecnologias móveis para aprendizagem, o novo não é a telefonia com voz, ou câmera fotográfica, ou a capacidade de computação, ou o livro, ou televisão, o novo acontece na utilização integrada e convergente de todas essas tecnologias, estando permanentemente disponível quando o usuário decidir utiliza-las” (Valentim, 2009). Desta forma o *M-Learning* surge como uma importante alternativa de ensino e treinamento à distância, contribuindo para facilidade de acesso ao aprendizado sem hora e local pré-estabelecido, como já mencionando anteriormente.

2.3. OBJETOS DE APRENDIZAGEM

Considerando a construção de soluções de aprendizagem com mobilidade, é necessário que estas devam priorizar os critérios de usabilidade, acessibilidade, mobilidade, colaboração/cooperação (SCHLEMMER, 2007).

Objetos de Aprendizagem são “quaisquer recursos digitais que possam ser reutilizados para o suporte ao ensino” (WILEY, 2002). Como “recurso digital” entende-se, nessa situação, simulações, animações, vídeos, fotos, sons, figuras, texto.

Os Objetos de Aprendizagem são utilizados para apoiar educadores no processo de ensino/aprendizagem. Podem ainda ser definidos, conforme Fabre et all. (2003);

[...] como qualquer recurso, suplementar ao processo de aprendizagem, que pode ser reusado para apoiar a aprendizagem. O termo objeto educacional (learning object) geralmente aplica-se a matérias educacionais projetados e construídos em pequenos conjuntos com vistas a maximizar as situações de aprendizagem onde o recurso pode ser utilizado. A ideia básica é a de que os objetos sejam como blocos com os quais será construído o contexto de aprendizagem. [...]

David A. Wiley conceitua Objetos de Aprendizagem de uma forma um pouco diferente do conceito proposto por Fabre. Wiley concluiu que os objetos de aprendizagem são “quaisquer

recursos digitais que possam ser reutilizados para o suporte ao ensino”. Essa definição inclui tudo o que pode ser distribuído por toda a rede de computadores sob demanda, seja grande ou pequeno.

No entanto, são definições muito abrangentes que aceitam como um Objeto de Aprendizagem qualquer recurso que seja utilizado em algum processo de ensino com base tecnológica, como base tecnológica, como por exemplo, pessoas, organizações e eventos. Para restringir um pouco, será usado, neste trabalho, o conceito proposto por CLOE (2008), segundo o qual os Objetos de Aprendizagem são *“entidades digitais projetadas para auxiliar o aluno a atingir um determinado objetivo pedagógico e que podem ser reutilizadas em outros contextos para apoiar a aprendizagem”*.

Assim, somente os materiais digitais projetados especificamente para fins pedagógicos serão considerados como Objetos de Aprendizagem. Neste contexto eles podem apresentar diferentes formatos, por exemplo, textos, apresentações e animações.

Conforme descrito por Mendes, Souza e Caregnato (2004), para que blocos de informação sejam considerados um Objeto de Aprendizagem e este possa ser inserido em um ambiente de aprendizagem é necessário que atenda às seguintes características:

- Reusabilidade – Capacidade de ser reutilizado em outros contextos;
- Adaptabilidade – Capacidade de se adaptar a qualquer ambiente de ensino;
- Granularidade – Capacidade de fragmentar o conteúdo do Objeto de Aprendizagem para compor unidades maiores;
- Acessibilidade – Capacidade de estar disponível via Internet para ser usado em diversos locais;
- Durabilidade – Capacidade de continuar a ser usado independente da mudança de tecnologia;
- Interoperabilidade – Capacidade de ser usado em qualquer plataforma, independente de sistema operacional.

Tendo em vista que, os componentes de software para *M-Learning* devem levar em consideração a capacidade de armazenamento dos aparelhos móveis e o tráfego necessário de dados. Desta forma, objetos de aprendizagem encaixam-se perfeitamente neste contexto. Objetos de aprendizagem são partes menores, autossuficientes, de conteúdo maior, ou seja, permitem que um assunto seja modularizado, conforme referiram Beck (2001) e Canto (2004).

Estes objetos vêm facilitar o ensino, separando o conteúdo macro em conteúdos pequenos, de modo a facilitar o entendimento e absorção. Por meio deles, o professor consegue focar o processo de ensino em um determinado assunto e, com isso, o aprendiz consegue empregar seus esforços mais eficientemente (Microsoft, 2004; Wiley, 2000; Greenberg, 2000; Smith, 2004);

Objetos de Aprendizagem móveis podem auxiliar o processo cognitivo, promovendo o conhecimento, até mesmo, em ambiente extraclasse, de maneira involuntária ou imperceptível já que os mesmos podem ser distribuídos na Internet, o que possibilita o acesso por qualquer pessoa a qualquer hora e, com o advento do *M-learning*, acesso a partir de qualquer lugar.

Além da distribuição na Internet, os Objetos de Aprendizagem apresentam outras vantagens como: ser uma abordagem diferenciada de ensino, pelo fato de usarem vários recursos multimídia, facilitando a apresentação dos conteúdos; e, a granularidade, ou seja, o objeto é uma pequena parte de um conceito maior, mas que prevalece independentemente do restante. (Oliveira, 2007).

Outra vantagem dos objetos de aprendizagem é a obtenção do conteúdo *on-demand*, ou seja, o aprendiz adquire somente o conteúdo necessário para aquele momento de estudo. Isso facilita a utilização e otimização da banda de transmissão de dados e a utilização da capacidade de armazenamento móvel, pois será transmitido apenas o necessário para aquela etapa de ensino (Canto, 2004).

O método de aprendizagem deve ser voltado de forma a permitir uma integração com os dispositivos móveis e tecnologias e as ações de ensino utilizadas na educação. Quando a mobilidade é integrada a um usuário possibilita uma troca de conhecimento e de novas ideias, pois o método de interação em aprender em qualquer lugar e a qualquer momento sempre esteve à frente das tecnologias, mas isto tem buscado e potencializado juntamente com ferramentas de ensino, e não mais com livros, cadernos e outros instrumentos já existentes (GRAZIOLA, 2008). Devido à crescente utilização de dispositivos móveis, aplicações que eram conhecidas apenas em ambientes *desktops*, passaram a ser disponibilizadas neste novo tipo de plataforma.

3. METODOLOGIA

O desenvolvimento do projeto pode ser descrito em dois momentos principais: Inicialmente serão feitos estudos sobre a tecnologia *M-Learning*, através de referenciais bibliográficos. Também serão feitos estudos baseados em algumas aplicativos educacionais já existentes, e com base nas informações coletadas, serão definidos os requisitos necessários para o desenvolvimento da ferramenta.

A metodologia de desenvolvida utilizada nesse trabalho se fundamenta principalmente na pesquisa bibliográfica. Entende-se por pesquisa bibliográfica a revisão da literatura sobre as principais teorias que norteiam o trabalho científico.

Conforme esclarece Marconi e Lakatos (1992), a pesquisa bibliográfica é o levantamento de toda a bibliografia já publicada, em forma de livros, revistas, publicações avulsas e imprensa escrita. A sua finalidade é fazer com que o pesquisador entre em contato direto com todo o material escrito sobre um determinado assunto, auxiliando o cientista na análise de suas pesquisas ou na manipulação de suas informações. Ela pode ser considerada como o primeiro passo de toda a pesquisa científica.

A pesquisa bibliográfica é uma importante etapa na constituição eficaz de um conjunto de investigação e que representa uma técnica que traz diferentes finalidades, entre as quais, é claro, ajudar o pesquisador a ampliar seus projetos. Em uma outra faceta, esta modalidade de pesquisa tem a incumbência de fundamentar teorias necessárias ao entendimento e compreensão dos elementos estudados.

Segundo Gil (2007) a pesquisa bibliográfica é desenvolvida a partir de material já elaborado, constituído principalmente de livros e artigos científicos. Embora em quase todos os estudos seja exigido algum tipo de trabalho desta natureza, há pesquisas desenvolvidas exclusivamente a partir de fontes bibliográficas. Boa parte dos estudos exploratórios pode ser definida como pesquisas bibliográficas.

A principal vantagem da pesquisa bibliográfica reside no fato de permitir ao investigador a cobertura de uma gama de fenômenos muito mais ampla do que aquela que poderia pesquisar diretamente. Esta vantagem se torna particularmente importante quando o problema de pesquisa requer dados muito dispersos pelo espaço.

3.2.1. FERRAMENTAS E TECNOLOGIAS

Nesta seção são apresentadas as principais tecnologias, ferramentas e linguagens utilizadas no desenvolvimento do sistema.

3.2.1.1. FERRAMENTAS

- **Netbeans 8.2**

Ambiente de Desenvolvimento Integrado *open source* utilizado para o desenvolvimento de aplicações em diferentes plataformas, utilizando tecnologia Java para isto. Totalmente gratuita, essa ferramenta proporciona algumas facilidades para o desenvolvimento de projetos, como auto completar, sistema de depuração em tempo de desenvolvimento, mostrando falhas de digitação, variáveis não declaradas, métodos inexistentes e etc. Em matéria de desenvolvimento, essa IDE é muito madura, sendo uma excelente alternativa para aqueles que desejam desenvolver aplicações Java de forma simples e rápida (Gonçalves, 1998).

- **SGBD POSTEGRESQL**

Sistema gerenciador de banco de dados *open source* coordenado pelo POSTEGRESQL Global Development Group, criado em 1996 na Universidade de Berkeley, regularizado sob a licença BSD (Berkeley Software Distribution). O POSTEGRESQL será utilizado, no sistema, para persistência dos dados.

- **BrModelo**

Ferramenta *free* voltada para a criação do esquema de banco de dados. É um software simples que funciona como um editor, resultado de um trabalho de graduação do curso de pós-graduação de banco de dados de um aluno da Universidade de Várzea Grande – Univag. O BrModelo tem como base a metodologia defendida pelo professor Heuser, em seu livro. Ele possui as seguintes funcionalidade: construção do modelo de entidade e relacionamento. Esta ferramenta será utilizada para modelagem do diagrama Entidade-Relacionamento e do modelo lógico de dados.

- **Astah**

É uma ferramenta de modelagem gratuita para projetos de sistemas orientados a objetos. É baseado em diagramas e na notação UML. Desenvolvido na plataforma Java, o que garante sua portabilidade para qualquer plataforma que possui a Máquina Virtual Java. É flexível e extensível e contém vários recursos. Nela é possível desenvolver vários diagramas: diagrama de casos de uso, diagrama de classe, diagrama de sequência, diagrama de estados, diagrama de atividades, diagrama de componentes, diagrama de implantação, diagrama de estrutura composta, diagrama de comunicação e diagrama de pacote (ASTAH COMMUNITY, 2011).

- **Git**

Projetado e desenvolvido por Linus Torvalds, o Git é sistema *open source* de controle de versão distribuído e sistema de gerenciamento de código fonte projetado para lidar com os mais diversos tipos de projetos, com velocidade e eficiência. Por ser uma arquitetura distribuída o Git permite que cada desenvolvedor tenha uma cópia local do histórico que está sendo produzido, o git funciona mesmo sem acesso a rede ou ao servidor remoto (Torvalds, 2010).

3.2.1.2. LINGUAGENS

- **Java**

Linguagem de programação criada em 1994 pela Sun Microsystems, Java foi projetada para ser uma linguagem pequena, simples e portátil a todas as plataformas e sistemas operacionais. Por ser uma linguagem criada seguindo o paradigma da orientação a objetos, Java traz de forma nativa a possibilidade de o programador usar conceitos de herança, polimorfismo e encapsulamento, além de oferecer outras facilidades, como: portabilidade, reusabilidade, *garbage collection*, multitarefa e outros recursos já testados e consolidados no mercado que podem ser adquiridos sem nenhum custo.

- **Java Enterprise Edition – Java EE**

Plataforma Java voltada para o desenvolvimento Web na linguagem de programação Java. O Java EE é construído sobre a plataforma Java SE e oferece um conjunto de APIs (Interfaces de programação de aplicações) para o desenvolvimento e execução de aplicações portáteis, robustas, escaláveis, confiáveis e seguras no lado do servidor. Java EE é uma plataforma extremamente poderosa para criação de aplicativos para Web,

podemos criar qualquer tipo de aplicativo, desde e-commerces até sites.gov. Essa plataforma junta a poderosa linguagem de programação Java com suas APIs e Frameworks para dar grande produtividade em qualquer tipo de projeto. (Luckow, 2008)

- **JavaScript**

Criada pela Netscape em parceria com a Sun Microsystems, JavaScript é uma linguagem de programação interpretada, criada com a finalidade de fornecer um meio de adicionar interatividade a uma página web. Originalmente implementada como parte dos navegadores web para que *scripts* pudessem ser executados do lado do cliente e interagissem com o usuário sem a necessidade deste *script* passar pelo servidor, controlando o navegador, realizando comunicação assíncrona e alterando o conteúdo do documento exibido. JavaScript é atualmente a principal linguagem para programação *cliente-side* em navegadores Web. Funções escritas em JavaScript podem ser facilmente embutidas dentro de seu documento HTML, possibilitando o incremento das funcionalidades do seu documento HTML com elementos interessantes. Sendo possível: responder facilmente a eventos iniciados pelo usuário, incluir efeitos que tornem sua página mais dinâmica. Logo, podemos criar sofisticadas páginas com a ajuda desta linguagem. (Sammy, 2010). A escolha desta tecnologia se deu principalmente pela a série de vantagens que ela proporciona, tais como: alterar e controlar de forma dinâmica a apresentação de um documento HTML, manipular o navegador (criar janelas pop-up, apresentar mensagens ao usuário, alterar as dimensões do navegador) interagir com formulários e etc. A mesma será usada no módulo web do sistema com o objetivo de agradar o usuário criando páginas mais dinâmicas e interativas.

3.2.1.3. TECNOLOGIAS

- **EJB**

É uma tecnologia de componentes básica da plataforma Java Enterprise Edition, que roda em um *container* de um servidor de aplicação. Seu principal objetivo consiste em fornecer um desenvolvimento rápido e simplificado de aplicações Java, com base em componentes distribuídos, transacionais, seguros e portáteis. Atualmente encontra-se na versão 3.1.

- **JSF**

JavaServer Faces é uma tecnologia do mundo Java EE, desenhado para simplificar o desenvolvimento de aplicações Web. JSF torna fácil o desenvolvimento de componentes de interface de usuário (GUI) e conecta esses componentes a objetos de negócios. Também automatiza o processo de uso de JavaBeans e navegação de páginas (Gonçalves, 1998).

- **Hibernate**

O Hibernate é um framework que se relaciona com banco de dados, onde esse relacionamento é conhecido como mapeamento objeto/relacional para Java, deixando o desenvolvedor livre para se concentrar em problemas da lógica de negócio. Sua simplicidade em configuração, dá ao desenvolvedor algumas regras para que sejam seguidas como padrões de desenvolvimento ao escrever sua lógica de negócios e suas classes de persistência (Gonçalves, 1998). O Hibernate se integra suavemente ao sistema se comunicando com o banco de dados como se fosse diretamente feito por sua aplicação.

3.2.2. easYProcess – YP

Para alcançar sucesso em um projeto de software, obtendo um produto de qualidade e entregue no prazo previsto é necessário o uso de um processo de desenvolvimento.

Segundo Humphrey, 1990, “Um processo de desenvolvimento de software é um conjunto de ferramentas, métodos e práticas usados para construir um produto de software”. A produção de um software envolve aspectos teóricos e práticos, embasamento prévio de engenharia de software e a utilização de um processo de desenvolvimento, de um bom suporte ferramental, assim como de tecnologias aceitas pelo mercado.

O easYProcess é uma metodologia de desenvolvimento de software ágil criada pelo grupo PET¹ da Universidade Federal de Campina Grande (UFCG), sendo idealizada pela Professora Dr^a Francile Procópio Garcia (Silva, 2010). Esta metodologia foi criada com o intuito de auxiliar os alunos do curso de Ciência da Computação no desenvolvimento e

¹ O Grupo PET (Programa de Educação Tutorial) do Curso de Graduação em Ciência da Computação da Universidade Federal de Campina Grande (UFCG) foi criado em 1992.

Para a **Implementação**, o processo prega o uso de algumas práticas, tais como: Design Simples, Padrões de Codificação, Padrões de Projeto, Refatoramento e Propriedade Coletiva de Código, a fim de produzir um código com mais qualidade. Há uma grande ênfase na parte de testes, tanto de unidade, que validam pequenos módulos do sistema, como de aceitação, que de fato representam a satisfação ou não do cliente diante do que foi desenvolvido.

A priori, YP foi escolhido para a modelagem e implementação do desenvolvimento do sistema, por ser considerado uma Metodologia de Desenvolvimento de Software Ágil simples e eficiente, e pelo mesmo auxiliar na gerência e no desenvolvimento de aplicações de pequeno e médio porte.

4. DESENVOLVIMENTO DO PROJETO

Este capítulo apresenta o resultado prático da realização deste trabalho.

4.1. APRESENTAÇÃO DO SISTEMA

A aplicação de acesso aos dispositivos móveis em todo mundo tem promovido mudanças no modo de produção e compartilhamento do conhecimento e tem apresentado múltiplas possibilidades para a aprendizagem, baseadas na mobilidade dos aparelhos, dos alunos, dos conteúdos e no acesso ao conhecimento em qualquer hora e em qualquer lugar.

A aplicação desenvolvida consiste num conjunto de ferramentas com intuito de auxiliar a interação entre alunos e professores, provendo mecanismo que possibilitem a troca de informações através da criação de comunidades de aprendizagem ativa, interativa e colaborativa.

O sistema está organizado em quatro módulos básicos.

4.1.1. MÓDULO DE EXERCÍCIOS

Este módulo é responsável pela realização das avaliações informais. Para o professor, existirá uma página na *Web* para o cadastro das questões e alternativas das mesmas. Após o cadastro, o professor deverá sinalizar quando a lista estará disponível, o tempo que permanecerá disponível e para qual turma o grupo de alunos estará acessível.

Para o aluno, o sistema deverá prover um mecanismo capaz de verificar as listas de exercícios e simulados disponíveis e carregar a lista selecionada em uma tela para que possa

ser respondida. Após ser respondida, o sistema deverá integrar e validar as informações, e enviá-las para serem armazenadas no banco de dados.

Um submódulo comum a aluno e professor permitirá que ambos vejam o resultado alcançados nos exercícios.

4.1.2. MÓDULO DE TRANSFERÊNCIA DE ARQUIVOS

O módulo de transferência de arquivos será dividido em *download* e *upload*. O módulo de *upload* permitirá ao professor escolher um ou mais arquivos e torná-los disponíveis para grupos de alunos, que poderão acessar os arquivos pelo módulo de *download*. Por intermédio deles, serão disponibilizados conteúdo através de micro apresentações que podem variar desde *slides* até arquivos em formato de áudio e vídeo, com intuito de introduzir os alunos em um novo tema, ou então sanar possíveis dúvidas. São fáceis de serem criadas, e sua versatilidade permite que sejam utilizadas como complemento em qualquer assunto (BARROS, 2002).

4.1.3. MÓDULO DE CRIAÇÃO DE FÓRUMS DE DISCURSÕES

Este módulo será responsável pela criação e gerenciamento de fóruns de discursões. **Será permitido ao aluno criar um fórum de discursão**, onde professores e outros alunos poderão participar. Alunos podem criar tópicos em um fórum, e estes podem ser comentado por outros alunos ou professores.

4.1.4. MÓDULO DE ENVIO DE MENSAGENS

Funcionando em conjunto com o módulo de exercícios, o módulo de transferência de arquivos e o módulo de criação de fóruns estará o módulo de envio de mensagens. Todas as mensagens transmitidas, por motivos de *performance* e segurança, serão armazenadas no banco de dados, e quando o professor ou aluno “*logar*” no sistema, este carregará todas as mensagens dirigidas ao usuário “*logado*”, sinalizando as que não foram lidas ou respondidas ainda.

Sua integração com os outros módulos acontecerá em dois momentos: quando um professor deixar disponível uma atividade ou um arquivo para *download*. Juntamente com a disponibilização, deve ser disparada uma mensagem para todos os que terão acesso ao material ou o exercício, avisando sobre suas características, como data de entrega e conteúdo do material, entre outras. Quando um aluno ou professor comentar algum tópico em algum fórum de discursão todos os membros daquele fórum deverão ser notificados.

4.2. ESPECIFICAÇÃO

Os requisitos é um conjunto de especificações referentes ao projeto a ser desenvolvido, ou seja, são todas as funções e características que o sistema deverá apresentar depois de finalizado. Destacando basicamente duas categorias de requisitos, os funcionais que são responsáveis por todas as funções que o sistema deverá apresentar para efetivação das atividades administrativas, e não funcionais que são responsáveis por todas as características que o sistema deverá apresentar (Garcia, 2007).

A seguir, são apresentados detalhadamente os requisitos funcionais e não funcionais do sistema.

4.2.1. REQUISITOS FUNCIONAIS

Seguindo o padrão descrito por Garcia (2007), teremos abaixo os requisitos que descrevem as funcionalidades do sistema.

Quadro 2 - Requisitos Funcionais

Manter professor	O sistema deverá permitir o cadastro, alteração e remoção dos dados dos professores, tais como: nome completo, matrícula data de nascimento, e-mail, senha, nível de formação (Especialista, Mestre, Doutor), instituição em que trabalha.
Manter aluno	O sistema deverá permitir o cadastro, alteração e remoção dos dados dos alunos, para cada aluno deverá ser persistido: nome completo, matrícula, curso, instituição em que estuda, pontuação, e-mail e senha para acesso ao sistema.
Manter disciplina	O sistema deverá permitir o cadastro, atualização e remoção de uma determinada disciplina, para cada disciplina deverá ser persistido: nome, código e carga horária.
Manter questão	O sistema deverá permitir o cadastro, atualização e remoção de uma questão, para cada questão deverá ser persistido um código, o enunciado, a resposta e o peso de cada questão.
Manter Arquivo	O sistema deverá permitir o cadastro e remoção de arquivos. Para cada arquivo deverá ser mantido o nome, um código, tamanho.
Responder exercícios	O sistema deverá permitir que alunos respondam os exercícios elaborados pelo professor.

Consultar resultado dos exercícios.	Deverá ser permitido a alunos e professores consultarem os resultados dos exercícios e simulados.
Fazer <i>upload</i> de arquivos	Deverá ser permitido a professores fazerem <i>uploads</i> de arquivos.
Fazer <i>download</i> de arquivos	Deverá ser permitido a alunos fazerem <i>download</i> dos arquivos disponibilizados pelos professores.
Manter fóruns de discussão	O sistema deverá permitir o cadastro, alteração e remoção dos dados de um fórum de discussão, tais como: nome, data de criação, fundador.
Criar tópicos	Alunos poderão criar tópicos em fóruns de discussões. Cada tópico deverá ter um código, um nome e uma data de criação.
Comentar tópicos	Será permitido a alunos e professores comentarem tópicos criados em um fórum.
Criar Exercício	O Sistema deve permitir a professores criar exercícios. Para cada exercício deve ser mantido um código e um nome.



4.2.2. REQUISITOS NÃO-FUNCIONAIS

Os requisitos que descrevem os aspectos não-funcionais do sistema são apresentados a seguir:

Quadro 3 - Requisitos Não Funcionais

Performance	Tempo de resposta do sistema quando o mesmo for solicitado não deve ultrapassar 10s, um sistema lento é alvo de críticas dos usuários, mesmo que seja funcional;
Segurança	Restrições de acesso e privilégios em relação aos usuários, o sistema só permitirá acesso aos dados com autorização, esse procedimento consistirá no uso de login e senha, somente usuários autorizados terão acesso aos dados garantindo assim a segurança dos dados;
Perspectiva do usuário	Como o sistema estará estruturado (padrão das cores, disposição dos objetos na tela);
Usabilidade	O sistema deve ser apresentado com uma interface simples, para facilitar ao máximo o uso do mesmo. A navegação deverá ser

	intuitiva até por que sistemas difíceis de serem usados terão pouca aceitação em relação aos usuários;
Confiabilidade	A capacidade do sistema de lidar com eventos inesperados, ou seja, o sistema deve ser capaz de lançar exceções e de recuperar-se de falhas, sem que haja perda de dados.
Portabilidade	A interface Web do sistema deve funcionar para os browsers Mozilla Firefox e Google Chrome;
Usuários simultâneos	O sistema deve estar preparado para suportar 1000 usuários simultâneos;

Para facilitar o desenvolvimento do sistema e melhor acompanhar a implementação do projeto, a implementação será dividida 4 em releases.

O primeiro release contemplará o desenvolvimento do módulo de exercícios, e incluirá o desenvolvimento dos requisitos funcionais: RF01, RF02, RF03, RF04, RF06, RF07, RF12.

O segundo release contemplará o desenvolvimento de módulo de transferência de arquivos, e incluirá o desenvolvimento dos requisitos funcionais: RF05, RF08.

O terceiro release contemplará o desenvolvimento de módulo de criação de fóruns de discussões e incluirá o desenvolvimento dos requisitos funcionais: RF09, RF10, RF11.

O quarto release contemplará o desenvolvimento do módulo de envio de mensagens.

A descrição das User Stories, bem como os testes de aceitação estão detalhadas no Apêndice B.

4.3. ANÁLISE E PROJETO

O sistema é constituído por três camadas lógicas: *camada de apresentação*, *camada de negócio* e *camada de persistência*.

- **Camada de Apresentação** – Camada responsável por disponibilizar os meios para acesso do usuário as funcionalidades do sistema. Esse componente tem como características ser modular e configurável o que facilita a customização do sistema aos mais variados tipos de dispositivos. Esta camada interage diretamente com a camada de negócio. É composta por dois módulos: módulo móvel compartilhado entre alunos e professores, e o módulo web para professores. A módulo web será composto por páginas

dinâmicas, que serão implementadas utilizando o *framework PrimeFace*, que ajuda na geração de páginas dinâmicas baseadas em HTML.

- **Camada de Negócio** – Camada responsável por todo processamento de dados do sistema. Esta camada conterá todas as interações necessárias para estruturação do negócio do cliente, a mesma é composta por beans gerenciados que funcionam como controladores que governam o acesso e atualização desses dados, atualizando-os de uma maneira apropriada, baseado na interação do usuário. Essa camada se comunica com a camada de apresentação e com a camada de persistência.
- **Camada de Persistência** - Camada responsável pela persistência de dados, essa camada conterá todas as interações necessárias para interagir o banco de dados com os objetos de negócio, e separar todo tipo de SQL ou linguagens padrões de banco de dados, da camada de negócio. A mesma também é responsável por fornecer uma interface para realização de tarefas, tais como: salvar, excluir, atualizar e excluir dados. Esta camada está intimamente relacionada com a camada de negócio.

A figura 2 apresenta a descrição geral da arquitetura do sistema.

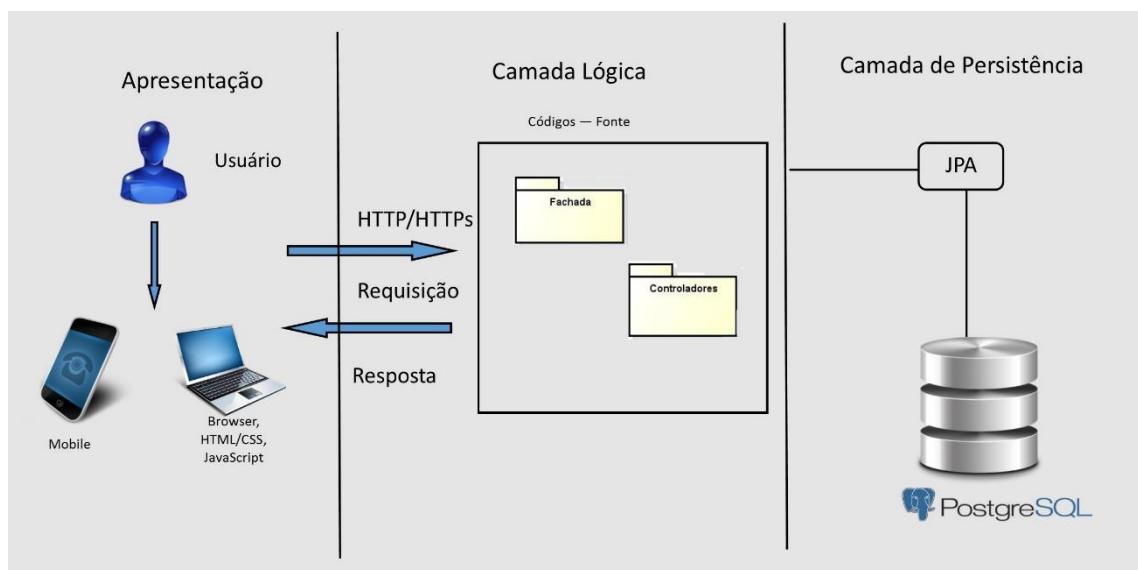


Figura 2 - **Desenho da Arquitetura Geral do Sistema**

Para representação da modelagem do sistema foram escolhidos os diagramas de caso de uso, classes e componentes.

4.3.1. DIAGRAMA DE CASO DE USO

Os diagramas de casos de uso são utilizados para descrever as funcionalidades propostas para o sistema. Esses diagramas fornecem um modo de descrever a visão externa do sistema e suas

interações com o mundo exterior. Assim podemos representar uma visão de alto nível de funcionalidades internacionais mediante requisições feitas pelo usuário.

A figura 3 apresenta o diagrama de caso de uso do sistema, contendo todos os casos de uso do

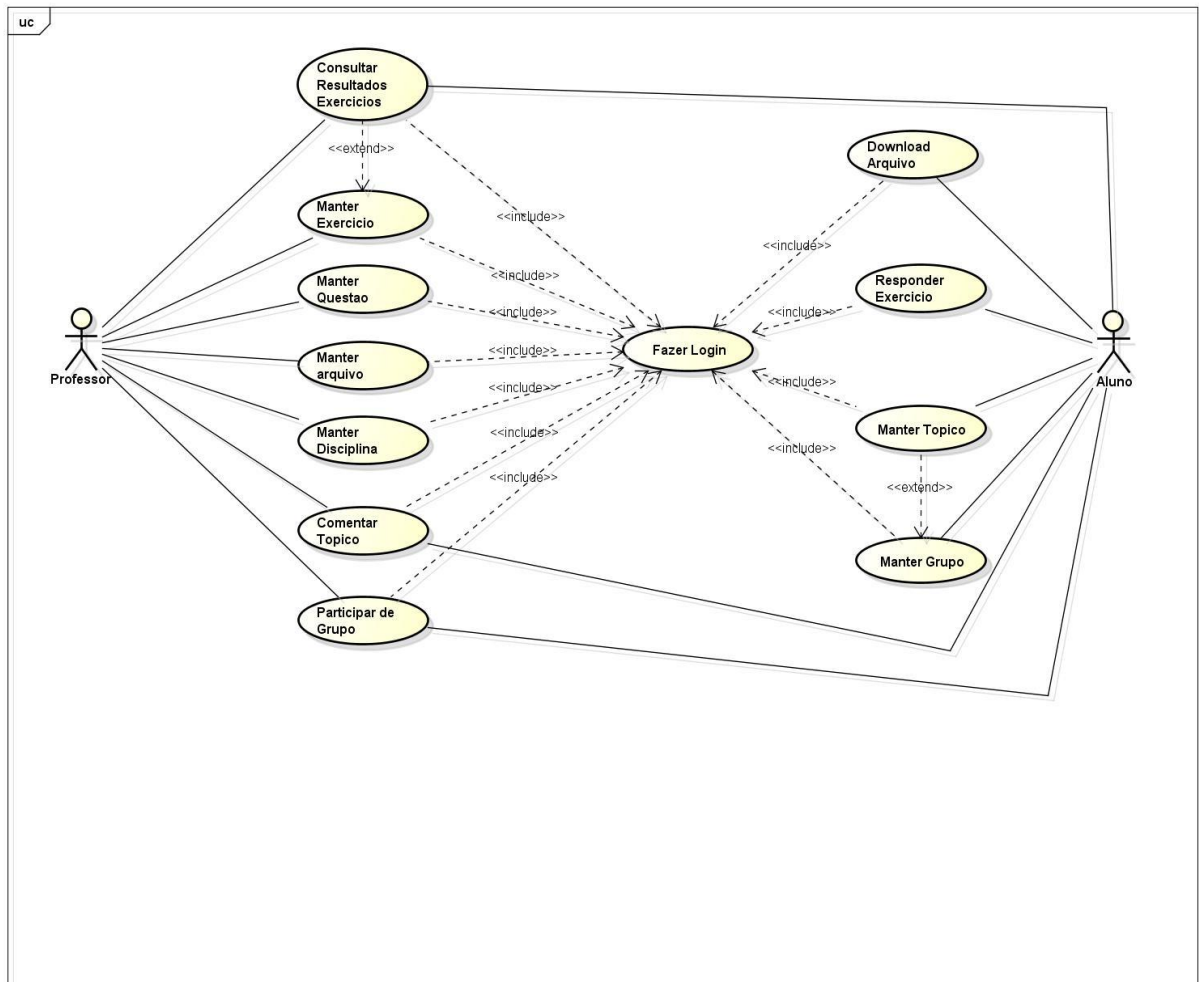


Figura 3 - Diagrama de Caso de Uso

4.3.1.1. Descrição dos Casos de Uso

Abaixo a descrição de cada um dos casos de uso descrito no diagrama, mas esta descrição está detalhada com o passo a passo no apêndice B.

UC01 Login – Este caso de uso especifica a ação de autenticação que um usuário executa no sistema, com objetivo de se conectar na aplicação. Este caso de uso é iniciado assim que o usuário tenta entrar no sistema. Caso o usuário nunca tenha entrado no sistema, o mesmo deve ser redirecionado para uma página de cadastro. O usuário fornece os dados básicos de autenticação e, após validação no sistema, o usuário torna-se apto a realizar operações da área restrita do sistema, caso contrário o mesmo será redirecionado para a mesma página. Apenas usuários cadastrados podem se autenticar no sistema.

UC02 Manter Disciplina – Este caso de uso possibilita que usuário do tipo Professor, devidamente identificado no sistema, cadastre uma disciplina. Possibilita também que o usuário realize busca de disciplinas, alterações dos dados cadastrais e inclusive remova uma disciplina do sistema, o mesmo é iniciado sempre que um professor escolhe a opção Disciplina, o professor fornece os dados básicos necessários e escolhe a opção adequada. Caso os dados inseridos não sejam devidamente corretos uma mensagem de erro e retornada ao usuário, caso contrário a operação é concluída com sucesso e uma mensagem é enviada ao usuário.

UC03 Consultar Resultado dos Exercício – Este caso de uso é válido tanto para usuários do tipo professores quanto para usuários do tipo aluno, ambos devidamente identificados no sistema, o mesmo possibilita que estes usuários consultem os resultados de um exercício. Para tanto, o sistema deve possuir ao menos um exercício cadastrado, respondido e corrigido. Caso não exista nenhum exercício cadastrado no sistema será mostrado uma mensagem informando aos usuários que não existem nenhum resultado para consulta.

UC04 Manter Arquivo – Este caso de uso possibilita que usuário do tipo Professor, devidamente identificado no sistema, faça upload de um arquivo e o disponibilize para seus alunos. O mesmo é iniciado sempre que um professor escolhe a opção Arquivos o professor fornece os dados básicos necessários e escolhe a opção adequada. Caso os dados inseridos não sejam devidamente corretos uma mensagem de erro e retornada ao usuário, caso contrário a operação é concluída com sucesso e uma mensagem é enviada ao usuário.

UC05 Fazer Download de Arquivo – Este caso de uso possibilita que um usuário do tipo Aluno, devidamente identificado no sistema faça download de algum arquivo disponibilizado por algum professor. O mesmo inicia sempre que um aluno seleciona a opção Minhas Disciplinas – Arquivos. Se houver algum arquivo para download o sistema habilita a funcionalidade de Download, caso contrário uma mensagem será enviada ao usuário informando que não existe nenhum arquivo.

UC06 Comentar Tópico - Assim como o caso de uso UC03 Consultar Resultado de Exercício, o mesmo é compartilhado por alunos e professores. O mesmo possibilita que usuários do tipo Aluno ou Professor comentem algum tópico em algum grupo de discussão. Este caso de uso inicia sempre que um professor ou aluno comenta algum tópico em algum grupo de discussão. Esse caso de uso permite também que ambos os usuários alterem ou remova algum comentário,

sempre que alguma dessas operações forem realizadas, uma mensagem é enviada ao usuário informando se a operação foi efetuada com sucesso ou não.

UC07 Manter Exercício - Este caso de uso possibilita que usuário do tipo Professor, devidamente identificado no sistema, cadastre uma Exercício. Possibilita também que o usuário realize busca de um exercício, alterações dos dados cadastrais e inclusive remova um exercício cadastrado no sistema, o mesmo é iniciado sempre que um professor escolhe a opção Exercícios, o professor fornece os dados básicos necessários e escolhe a opção adequada. Caso os dados inseridos não sejam devidamente corretos uma mensagem de erro é retornada ao usuário, caso contrário a operação é concluída com sucesso uma mensagem é enviada ao usuário informando o sucesso da operação.

UC08 Manter Grupo de Discursão - Este caso de uso possibilita que usuário do tipo Aluno, devidamente identificado no sistema, crie um novo Grupo de Discursão. Possibilita também que o usuário realize busca de um determinado Grupo, bem como altere os dados cadastrais e inclusive remova um grupo cadastrado no sistema. Caso os dados inseridos não sejam devidamente corretos uma mensagem de erro é retornada ao usuário, caso contrário a operação é concluída com sucesso e uma mensagem é enviada ao usuário.

UC09 Responder Exercício - Esse caso de uso possibilita que usuários do tipo Aluno, devidamente identificado no sistema resolver as questões compõem um exercício, permitindo ao aluno assinalar as alternativas consideradas por ele corretas e confirmar suas respostas para posterior verificação de desempenho.

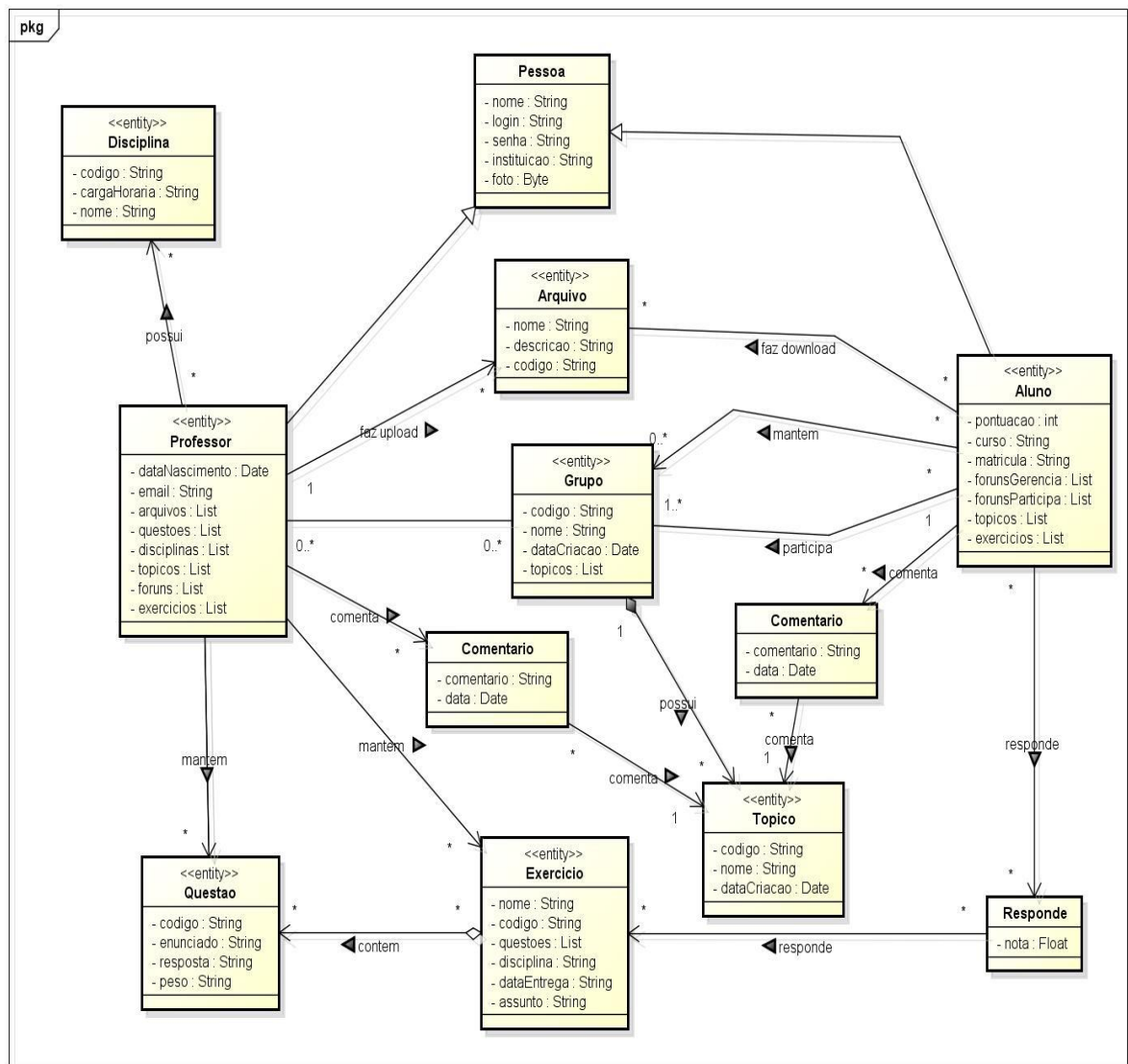
UC10 Manter Tópico - Este caso de uso possibilita que usuário do tipo Aluno, devidamente identificado no sistema, crie um novo tópico em um Grupo de Discursão. Possibilita também que o usuário altere o conteúdo de um tópico, bem como remova um tópico de um determinado grupo. Caso os dados inseridos não sejam devidamente corretos uma mensagem de erro é retornada ao usuário, caso contrário a operação é concluída com sucesso e uma mensagem é enviada ao usuário.

UC11 Participar Grupo – Esse caso de uso é compartilhado entre alunos e professores, devidamente autenticados no sistema, o mesmo permite que ambos usuários participem de um grupo de discursão. O mesmo é iniciado sempre que algum usuário solicita participar de um grupo, após solicitar o usuário fica aguardando uma confirmação do administrador do grupo, caso a solicitação seja aceita uma mensagem é enviada ao usuário e o mesmo pode ter acesso às informações do grupo.

UC11 Manter Questão - Este caso de uso possibilita que usuário do tipo Professor, devidamente identificado no sistema, crie um novo tópico em um Grupo de Discursão. Possibilita também que o usuário altere o conteúdo de um tópico, bem como remova um tópico de um determinado grupo. Caso os dados inseridos não sejam devidamente corretos uma mensagem de erro é retornada ao usuário, caso contrário a operação é concluída com sucesso e uma mensagem é enviada ao usuário.

4.3.2. DIAGRAMA DE CLASSES

Juntamente com os requisitos elicitados, foi desenvolvido o diagrama de classes, representando a modelagem inicial do sistema. Este diagrama serve para compreender os relacionamentos entre as partes que o compõem. O objetivo do diagrama é atender as necessidades mostradas no diagrama de Entidade-Relacionamento. A figura 3 apresenta o diagrama de classes do sistema onde serão mostrados os relacionamentos entre as classes suas multiplicidades e atributos.



powered by Astah

Figura 4 - Diagrama de Classes

As classes *Pessoa*, *Aluno*, *Professor*, *Disciplina*, *Exercicio*, *Questao*, *Topico*, *Forum*, *Arquivo*, *Comentario* e *Responde* atuam como *beans*, formando as estruturas dos objetos que, em junção com o JPA, irão gerar as tabelas no banco de dados.

A entidade *Aluno* é uma especialização da entidade *Pessoa*, um aluno pode criar e participar de vários grupos, o que reflete a sua relação com essa classe. Um aluno também pode criar vários tópicos nos grupos aos quais ele participa, bem como pode comentar esses tópicos, fazer download de arquivos e responder exercícios, esse comportamento justifica o relacionamento dessa classe com as demais classes citadas.

A entidade *Professor* também é uma especialização da entidade *Pessoa*, um professor tem controle sobre os exercícios e as questões que compõem esses exercícios, ou seja, ele possui uma relação com a classe *Exercício* e com a classe *Questão*. A associação de um professor a classe *Arquivo* se deve ao fato de que um professor pode disponibilizar arquivos para download. A associação de um professor com a classe *Disciplina* ocorre pelo fato do professor ser responsável por uma ou várias disciplinas. Além disso um professor também pode participar de

vários grupos e comentar vários tópicos, por isso essa entidade tem relacionamento com as classes Grupo e Topico.

As demais classes apresentam comportamento semelhante e servem para armazenar e transmitir informações nas camadas da aplicação.

4.3.3. DIAGRAMA DE COMPONENTES

Um dos principais objetivos do padrão MVC é a organização do código de uma aplicação em camadas, realizando assim uma separação física dos componentes do software. Esse modelo consiste em uma tríade de classes frequentemente usadas em sistemas interativos para construção de interfaces com o usuário. A implementação desse modelo mantém o núcleo funcional do sistema independente da interface. Assim, as interfaces internas podem permanecer estáveis, mesmo quando a interface necessita ser alterada para se adaptar a novas plataformas e dispositivos de interação (Buschmann, 2009).

Com isso a apresentação, a lógica e o acesso ao banco de dados estão separados em camadas específicas, tornando os sistemas mais manuteníveis e garantindo a independência entre as camadas. Desta forma, as camadas de negócio podem ser divididas em classes podendo ser agrupadas em pacotes ou componentes reduzindo as dependências entre as mesmas.

A figura 4 apresenta o diagrama de componentes deste projeto, o mesmo é composto pelas três partes do MVC – Modelo, Visão e Controladores;

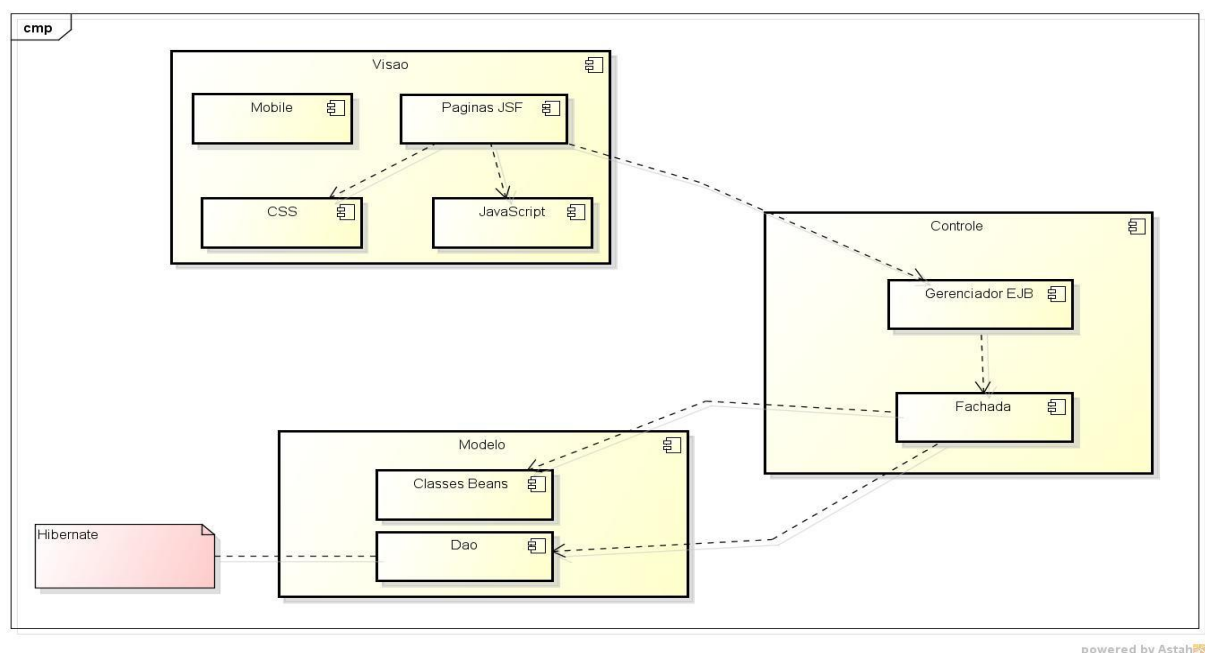


Figura 5 - Diagrama de Componentes

4.3.4. PROJETO DE BANCO DE DADOS – DIAGRAMAS

A abordagem de modelagem de dados utilizada se concentrou nas estruturas e nas restrições da base de dados, utilizando os conceitos do Modelo Entidade-Relacionamento. Este modelo conceitual de dados de alto nível é frequentemente usado para o projeto conceitual de aplicações de bases de dados e se baseia na percepção de um universo constituído por um grupo básico de objetos chamados de entidades, e pelos relacionamentos entre eles.

Serão apresentadas as fases de modelagem do banco de dados, conforme as figuras 6 e 7. A figura 6 demonstra o modelo conceitual do banco de dados, a sua estrutura; e a figura 7 apresenta o modelo lógico, onde são demonstrados os detalhes de cada campo das tabelas a partir do modelo conceitual.

4.3.4.1. DIAGRAMA ENTIDADE-RELACIONAMENTO

O diagrama Entidade-Relacionamento é utilizado para representar, de forma sucinta e bem estruturada, todos os elementos abstraídos no processo de análise de sistemas. Este diagrama consiste na representação principal do modelo de dados de um sistema com alto nível de abstração. Sua maior aplicação é para visualizar o relacionamento entre tabelas, no qual relações são construídas através de associação de um ou mais atributos destas tabelas que farão parte do projeto físico do banco de dados.

Após a fase de elicitação dos requisitos, foi elaborado o diagrama Entidade-Relacionamento, mostrado na figura 6. Tal diagrama representa a modelagem inicial do sistema, atendendo completamente aos requisitos elicitados.

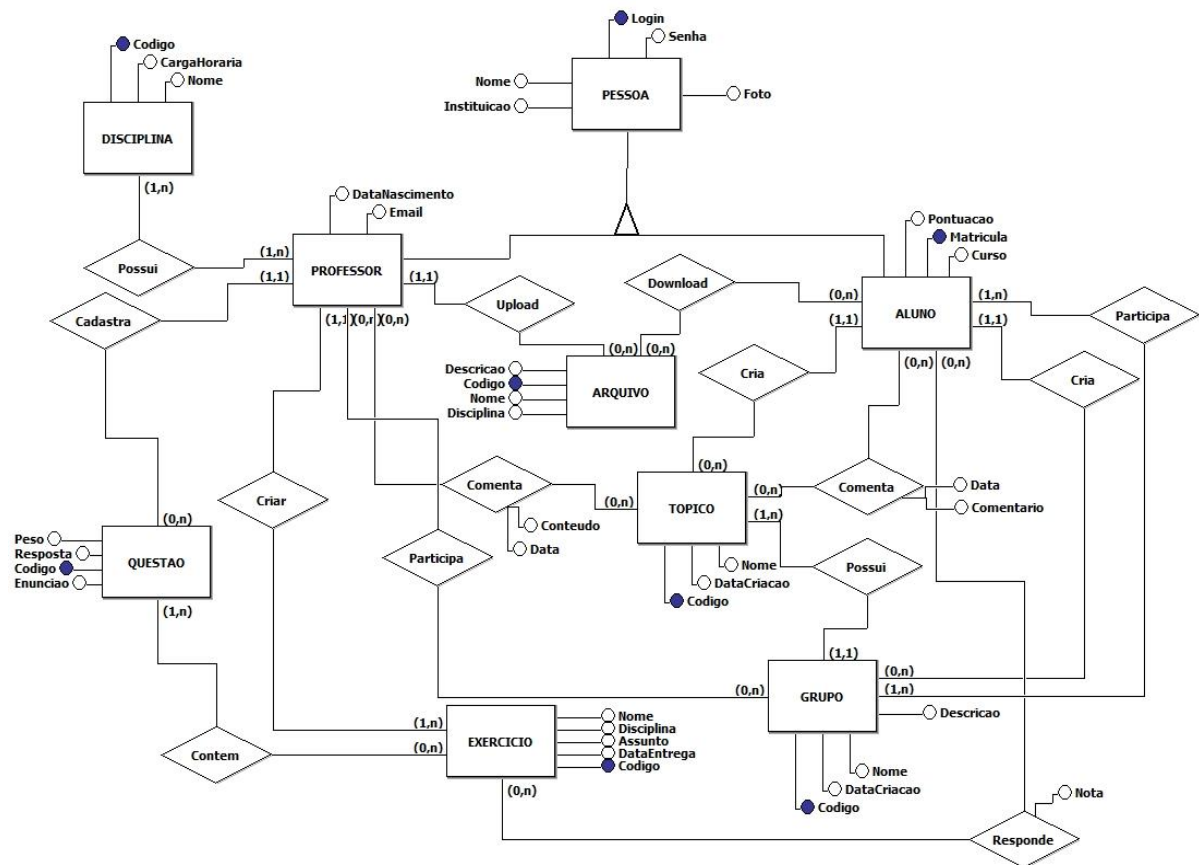


Figura 6 - Diagrama Entidade-Relacionamento

5. CONCLUSÃO

Ferramentas que complementem o ensino e aprendizagem de alunos são sempre bem-vindas. Muito tem se desenvolvido no cenário computacional para ajudar nesse processo, mas pouco tem se aproveitado. Devido a sua popularização, o uso de dispositivos móveis no processo de ensino e aprendizagem no momento tem se caracterizado como uma promissora possibilidade de melhorar o ensino, focando esses recursos como um elemento que propicie o aprendizado, que estimule o aluno a aprender, a se tornar independente e colaborativo.

Softwares educacionais e bem projetados podem ser criados e utilizados para unir práticas educativas com recursos multimídias em ambientes portáteis, tornando o *M-Learning* uma possibilidade viável, uma vez que, permite a professores e alunos desfrutar de maior eficiência na comunicação já que essas aplicações agregam aos usuários a facilidade de se manter conectado a sala de aula sem restrições de horário e local.

O sistema proposto neste trabalho objetiva transmitir ao usuário informações de caráter enriquecedor no processo de ensino/aprendizagem aliados as tecnologias móveis, fornecendo assim, conhecimento por meio de fontes diferenciadas, permitindo que a aprendizagem ocorra mesmo com alunos e professores, interagindo em tempos e espaços diferentes. Mas acima de tudo o seu desenvolvimento será guiado pelos conhecimentos adquiridos no decorrer do curso, presando pelas boas práticas de análise e desenvolvimento.

6. REFERÊNCIAS BIBLIOGRÁFICAS

Barbosa, D. N. F. (2007), **Um modelo de educação ubíqua orientado à consciência do contexto do aprendiz**. Tese Doutorado. Universidade Federal do Rio Grande do Sul. Porto Alegre.

Beck, Robert J. **“Object-Oriented Content: Importance, Benefits, and Costs**. University of Winsconsin, Estados Unidos da América, 2001. Disponível em: <http://www.educause.edu/Library/DetailPage/666?Redirect=True&ID=EDU0297>, consultado em dezembro de 2014.

Behar, P. A.; Batista, S. C. F. **Dispositivos Móveis na Educação. In: Pátio Ensino Fundamental**, n. 56, p. 26-29. Nov. 2010 - Jan. 2011. p. 26-29.

Canto, Nilton Furtado. **“O uso de ferramentas de autoria para construção de sistemas tutores inteligentes (The use of authoring tools for construction of intelligent tutoring systems)”**. Escola Politécnica da Universidade de São Paulo, Brasil, 2004. Disponível em: <http://www.conahpa.ufsc.br/2004/artigos/Tema5/03.pdf>, consultado em dezembro de 2014.

Casas, Luis Alberto Alfaro. **“Contribuições para a modelagem de um ambiente de educação baseado em realidade virtual”**. Universidade Federal de Santa Catarina, Brasil, 1999. Disponível em: <http://www.eps.ufsc.br/teses99/casas/>, consultado em dezembro de 2014.

Castilo, S., Ayala, G.: ARMOLEO (2008) **“An Architecture for Mobile Learning Object”**, In: **18th International Conference on Electronics, Communications and Computers (CONIELECOMP)**, p. 53-58, IEEE Computer Society, Los Alamitos.

Garcia, Francile Procópico; Lima, Aliandro Higino Guedes; Ferreira, Danilo de Sousa; Júnior, Fábio Luiz Leite; Rocha, Giselle Regina Chaves da; Mendes, Gustavo Wagner Diniz; Pontes, Renata França de; Rocha, Verlayanne Kelley da Hora; Dantas, Vinicius Farias, **easYProcess – Um Processo de Desenvolvimento de Software**. Universidade Federal de Campina Grande. Campina Grande: 2007.

Graziola JR, P. G. (2008), **Aprendizagem com Mobilidade (M-learning): Novas Possibilidades para as Práticas Pedagógicas e a Formação Docente?** Mestrado em Práticas Pedagógicas e a Formação Docente Unisinos / Bolsa CNPq Profa. Dra. Eliane Schlemmer – Unisinos.

Greenberg, Jane. **Metadata and organizing educational resources on the Internet**. Binghamton: Haworth Information Press, 2000.

Lakatos, Eva Maria; Marconi, Marina de Andrade. **Fundamentos de metodologia científica**. São Paulo: Atlas, 1993.

Meirelles, Luiz F.T.; Tarouco, Liane M.R. & Alves, Carlos V.R. **“Telemática aplicada à aprendizagem com mobilidade”**. Universidade Federal do Rio Grande do Sul, Brasil, 2004. Disponível em: http://www.cinted.ufrgs.br/renote/nov2004/artigos/a13_telematica-aplicada.pdf, consultado em dezembro de 2014.

Oliveira, L. R. **Desenvolvimento de objetos de aprendizagem para dispositivos móveis: iniciação ao m-learning**. 2008. 63 f. Monografia (Trabalho de Conclusão do Curso de Ciência da Computação) – Centro de Tecnologia, Universidade Federal de Santa Maria, Santa Maria, 2008. PILETTI, C. Didática Geral. 21.ed. São Paulo: Ática, 1997. 258 pg.

Parra Filho, D. e Santos J. A. **Metodologia científica**. São Paulo: Futura, 1998.

Pelissoli, Luciano & Ioyolla, Waldomiro. **“Aprendizado móvel (M-Learning): dispositivos e cenários”**. Salvador, Brasil, 2004. Disponível em: <http://www.abed.org.br/congresso2004/por/pdf/074-TCC2.pdf>, consultado em dezembro de 2014.

Pretto, Nelson de Luca, serpa Luis Felipe, **A Educação e a Sociedade da Informação, II Conferência Internacional de Tecnologias de Informação e Comunicação na Educação**, Universidade Federal da Bahia, Salvador: 2001.

Ribas, L. M. L. R. **Iniciação à pesquisa científica: questões para reflexão e estudo**. Campo Grande: Universidade Católica Dom Bosco - Biblioteca, 1999.

Rudio, Franz Victor. **Introdução ao projeto de pesquisa científica**. Petrópolis: Vozes, 2000.

Salvador, Angelo Domingos. **Métodos e técnicas de pesquisa bibliográfica**. Porto Alegre: Sulina, 1978.

Severino, Antônio Joaquim. **Metodologia do trabalho científico**. São Paulo: Cortez, 2000.

Schlemmer, E.; Zanela, S. A.; Barbosa, J.; Reinhard, N. (2007), **M-Learning ou Aprendizagem com Mobilidade: Casos no contexto Brasileiro**. 13º Congresso Internacional de Educação à Distância. Curitiba.

Schwartz, Alice B. “**Aprendizagem por observação**”. Universidade Federal de Santa Catarina, Brasil, 1996. Disponível em: <http://www.inf.ufsc.br/iad/users/a/alice/aliceiad.htm>, consultado em dezembro de 2014.

Tarouco, Liane M.R.; FABRE, Marie C.J.M.; grando, Anita R.S. & konrath, Mary L.P. “**Objetos de aprendizagem para M-Learning**”. Universidade Federal do Rio Grande do Sul, Brasil, 2004. Disponível em: http://www.cinted.ufrgs.br/CESTA/objetosdeaprendizagem_sucesu.pdf, consultado em dezembro de 2014.

Viviane de F. Bartholo; Marília A. Amaral; Maria I. Cagnin (2009). **M-AVA: Modelo de Adaptabilidade para Ambientes Virtuais Móveis de Aprendizagem**. XX Simpósio Brasileiro de Informática na Educação.

ZANELLA, A. ; SCHLEMMER, Eliane ; BARBOSA, Jorge Luis Victória ; REINHARD, Nicolau . MLearning ou Aprendizagem com Mobilidade: um estudo exploratório sobre sua utilização no Brasil. In: XXX Encontro Nacional dos Programas de Pós-Graduação em Administração, 2007, Rio de Janeiro. ANPAD, 2007. v. 1. p. 1-17. Disponível em: http://gpedunisinos.files.wordpress.com/2009/06/art_m-learning-ou-aprendizagem-commobilidade.pdf. Acesso em: 06 Fevereiro 2015.

APÊNDICE A – USER STORIES

Quadro 4 - Detalhamento das User Stories

Lista de User Stories	
US01	Implementar funcionalidade de cadastro de usuários (professores e alunos)
	<ul style="list-style-type: none"> • TA1.1 – Cadastrar um professor ou aluno com todos os dados corretos (Cadastro efetuado com sucesso); • TA1.2 – Cadastrar um professor ou aluno sem informar todos os campos obrigatórios (Cadastro não deve ser efetuado); • TA1.3 – Cadastrar um professor ou aluno informando dados inválidos (Cadastro não deve ser efetuado);
US02	Implementar funcionalidade de atualização de usuários (Professores/Alunos)
	<ul style="list-style-type: none"> • TA02.1 – Atualizar dados informando todos os campos corretamente (Atualização deve ser efetuada com sucesso); • TA02.2 – Atualizar dados informando dados inválidos (Atualização não deve ser efetuada); • TA02.3 – Atualizar dados sem informar todos os campos corretamente (Atualização não deve ser efetuada);
US03	Implementar funcionalidade de remoção de usuários (Professores/Alunos)
	<ul style="list-style-type: none"> • TA03.1 – A conta deve ser excluída com sucesso;
US04	Implementar funcionalidade de cadastro de disciplina
	<ul style="list-style-type: none"> • TA04.1 – Cadastrar uma disciplina com todos os dados corretos (Cadastro efetuado com sucesso); • TA04.2 – Cadastrar uma disciplina sem informar todos os campos obrigatórios (Cadastro não deve ser efetuado); • TA04.3 – Cadastrar uma disciplina informando dados inválidos (Cadastro não deve ser efetuado);
US05	Implementar funcionalidade de Consulta de disciplinas
	<ul style="list-style-type: none"> • TA05.1 - Consultar dados de uma disciplina a partir de uma chave existente (Dados devem ser retornados com sucesso); • TA05.2 - Consultar dados de uma disciplina a partir de uma chave inválida (Mensagem de erro deve ser retornada); • TA05.3 - Recuperar todos as disciplinas de um professor;
US06	Implementar funcionalidade de atualização de disciplinas
	<ul style="list-style-type: none"> • TA06.1 – Atualizar dados informando todos os campos corretamente (Atualização deve ser efetuada com sucesso); • TA06.2 – Atualizar dados informando dados inválidos (Atualização não deve ser efetuada);

	<ul style="list-style-type: none"> TA06.3 – Atualizar dados sem informar todos os campos corretamente (Atualização não deve ser efetuada);
US07	Implementar funcionalidade de remoção de disciplinas
	<ul style="list-style-type: none"> TA07.1 – Excluir uma determinada disciplina a partir de uma chave existente (Registro deve ser excluído); TA07.2 – Fazer busca pela chave referente a uma disciplina X (Nenhum registro deve ser retornado);
US08	Implementar funcionalidade de cadastro de questão
	<ul style="list-style-type: none"> TA08.1 – Cadastrar uma questão com todos os dados corretos (Cadastro efetuado com sucesso); TA08.2 – Cadastrar uma questão sem informar todos os campos obrigatórios (Cadastro não deve ser efetuado); TA08.3 – Cadastrar uma questão informando dados inválidos (Cadastro não deve ser efetuado);
US09	Implementar funcionalidade de consulta de questão
	<ul style="list-style-type: none"> TA09.1 - Consultar dados de uma questão a partir de uma chave existente (Dados devem ser retornados com sucesso); TA09.2 - Consultar dados de uma questão a partir de uma chave inválida (Mensagem de erro deve ser retornada); TA09.3- Recuperar todas as questões cadastradas por um professor;
US10	Implementar funcionalidade de atualização de questão
	<ul style="list-style-type: none"> TA10.1 – Atualizar dados informando todos os campos corretamente (Atualização deve ser efetuada com sucesso); TA10.2 – Atualizar dados informando dados inválidos (Atualização não deve ser efetuada); TA10.3 – Atualizar dados sem informar todos os campos corretamente (Atualização não deve ser efetuada);
US11	Implementar funcionalidade de remoção de questão
	<ul style="list-style-type: none"> TA11.1 – Excluir uma determinada questão a partir de uma chave existente (Registro deve ser excluído); TA12.2 – Fazer busca pela chave referente a uma questão X (Nenhum registro deve ser retornado);
US12	Implementar funcionalidade de upload de arquivos
	<ul style="list-style-type: none"> TA12.1 – Anexar um arquivo sem informar todos os campos obrigatórios (Operação não deve ser concluída); TA12.2 – Anexar um arquivo informando todos os campos obrigatórios (Operação deve ser concluída);
US13	Implementar funcionalidade de remoção de arquivo
	<ul style="list-style-type: none"> TA13.1 – Remover um arquivo a partir de uma chave existente (Arquivo deve ser removido); TA13.1 – Fazer busca pela chave referente ao arquivo X (Nenhum registro deve ser retornado);

US14	Implementar funcionalidade de responder exercício
	<ul style="list-style-type: none"> TA14.1 – Responder lista sem responder todas as perguntas (Operação deve ser concluída);
US15	Implementar funcionalidade de consultar resultados dos exercícios
	<ul style="list-style-type: none"> TA15.1 – Fazer consulta dos resultados;
US16	Implementar funcionalidade de download de arquivo
	<ul style="list-style-type: none"> TA7.1 – Fazer download de um arquivo (Arquivo deve ser baixado);
US17	Implementar funcionalidade de cadastro de Fóruns
	<ul style="list-style-type: none"> TA17.1 – Cadastrar fórum com todos os dados corretos (Cadastro efetuado com sucesso); TA17.2 – Cadastrar fórum sem informar todos os campos obrigatórios (Cadastro não deve ser efetuado); TA17.3 – Cadastrar fórum informando dados inválidos (Cadastro não deve ser efetuado);
US18	Implementar funcionalidade de consulta de fóruns
	<ul style="list-style-type: none"> TA18.1 - Consultar dados de um fórum a partir de uma chave existente (Dados devem ser retornados com sucesso); TA18.2 - Consultar dados de um fórum a partir de uma chave inválida (Mensagem de erro deve ser retornada); TA18.3 - Recuperar todos os fóruns ao qual um usuário faz parte;
US19	Implementar funcionalidade de atualização de fóruns;
	<ul style="list-style-type: none"> TA19.1 – Atualizar dados informando todos os campos corretamente (Atualização deve ser efetuada com sucesso); TA19.2 – Atualizar dados informando dados inválidos (Atualização não deve ser efetuada); TA19.3 – Atualizar dados sem informar todos os campos corretamente (Atualização não deve ser efetuada);
US20	Implementar funcionalidade de remoção de fóruns;
	<ul style="list-style-type: none"> TA20.1 – Excluir um determinado fórum a partir de uma chave existente (Registro deve ser excluído); TA20.2 – Fazer busca pela chave referente ao fórum X (Nenhum registro deve ser retornado);
US21	Implementar funcionalidade de cadastrar exercício
	<ul style="list-style-type: none"> TA21.1 – Cadastrar exercício informando todos os dados corretos (Cadastro deve ser efetuado com sucesso); TA21.1 – Cadastrar exercício sem informar todos os dados corretos (Cadastro não deve ser efetuado); TA21.2 – Cadastrar exercício informando dados inválidos (Cadastro não deve ser efetuado);
US22	Implementar funcionalidade de consulta de exercício
	<ul style="list-style-type: none"> TA22.1 - Consultar dados de um exercício a partir de uma chave existente (Dados devem ser retornados com sucesso);

	<ul style="list-style-type: none"> • TA22.2 - Consultar dados de um exercício a partir de uma chave inválida (Mensagem de erro deve ser retornada); • TA22.3 - Recuperar todos os exercícios que um professor cadastrou;
US23	Implementar funcionalidade de atualização de exercício
	<ul style="list-style-type: none"> • TA23.1 – Atualizar dados informando todos os campos corretamente (Atualização deve ser efetuada com sucesso); • TA23.2 – Atualizar dados informando dados inválidos (Atualização não deve ser efetuada); • TA23.3 – Atualizar dados sem informar todos os campos corretamente (Atualização não deve ser efetuada);
US24	Implementar funcionalidade de remoção de exercício;
	<ul style="list-style-type: none"> • TA24.1 – Excluir um determinado exercício a partir de uma chave existente (Registro deve ser excluído); • TA24.2 – Fazer busca pela chave referente ao exercício X (Nenhum registro deve ser retornado);
US25	Implementar funcionalidade de criar tópico
	<ul style="list-style-type: none"> • TA25.1 – Cadastrar tópico informando todos os dados corretos (Cadastro deve ser efetutado com sucesso); • TA25.2 – Cadastrar tópico sem informar todos os dados corretos (Cadastro não deve ser efetutado); • TA25.3 – Cadastrar tópico informando dados inválidos (Cadastro não deve ser efetutado);
US26	Implementar funcionalidade de consulta de tópicos
	<ul style="list-style-type: none"> • TA26.3 - Recuperar todos os tópicos de um grupo;
US27	Implementar funcionalidade de atualização de tópico
	<ul style="list-style-type: none"> • TA27.1 – Atualizar dados informando todos os campos corretamente (Atualização deve ser efetuada com sucesso); • TA27.2 – Atualizar dados informando dados inválidos (Atualização não deve ser efetuada); • TA27.3 – Atualizar dados sem informar todos os campos corretamente (Atualização não deve ser efetuada);
US28	Implementar funcionalidade de remoção de tópico;
	<ul style="list-style-type: none"> • TA28.1 – Excluir um determinado tópico a partir de uma chave existente (Registro deve ser excluído);
US29	Implementar funcionalidade de comentar tópico (Alunos/Professores)
	<ul style="list-style-type: none"> • TA29.1 – Comentar tópico informando todos os dados corretos (Comentário deve ser efetutado com sucesso); • TA29.2 – Comentar tópico sem informar todos os dados corretos (Comentário não deve ser efetutado);
US30	Implementar funcionalidade de atualização de comentário
	<ul style="list-style-type: none"> • TA30.1 – Atualizar dados informando todos os campos corretamente (Atualização deve ser efetuada com sucesso);

	<ul style="list-style-type: none"> • TA30.2 – Atualizar dados informando dados inválidos (Atualização não deve ser efetuada); • TA30.3 – Atualizar dados sem informar todos os campos corretamente (Atualização não deve ser efetuada);
US31	Implementar funcionalidade de consulta de comentários
	<ul style="list-style-type: none"> • TA26.3 - Recuperar todos os comentários de um tópico;
US32	Implementar funcionalidade de remoção de comentário;
	<ul style="list-style-type: none"> • TA28.1 – Excluir um determinado comentário a partir de uma chave existente (Registro deve ser excluído);
US33	Incluir funcionalidades de autenticação para usuários
	<ul style="list-style-type: none"> • TA33.1 – Acessar o sistema a partir de um login válido (autenticação feita com sucesso); • TA34.2 – Acessar o sistema a partir de um login inválido (Mensagem de erro deve ser gerada);

APÊNDICE B – DESCRIÇÃO DOS CASOS DE USO

Quadro 5 - Caso de Uso Login

Nome	Login
Atores	Todos os tipos de usuários
Pré- Condições	Está Cadastrado no sistema
Fluxo Principal	<ol style="list-style-type: none"> 1. O Usuário entra no sistema; 2. O sistema disponibiliza a interface para realizar o login e senha. 3. O Usuário deve informar no local indicado o login e a senha e clicar no botão “entrar”; 4. O sistema verifica as informações de login e senha; 5. O sistema envia uma mensagem de confirmação; 6. O sistema carrega os dados da aplicação e exibe a tela principal;
Fluxo Alternativo	<ol style="list-style-type: none"> 1. Caso o login ou senha sejam inválidos, o sistema envia uma mensagem informando o erro. 2. O usuário confirma a informação e retorna ao passo 1 do fluxo principal.

Quadro 6 - Caso de Uso Manter Exercício

Nome	Manter Exercício
Atores	Professor
Pré- Condições	O Usuário deve ser professor e estar cadastrado e logado no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 1. O professor seleciona a opção “Exercício” no menu de opções; 2. Em seguida o sistema retornará a página de cadastro de Exercício, caso o usuário deseje realizar operações de atualização ou exclusão, deverá pesquisar por tal primeiro; 3. O Professor preenche os dados corretamente e escolhe a opção “Cadastrar”; 4. Se não houver erros, o sistema salva os dados preenchidos; 5. O Sistema envia uma mensagem de confirmação;
Fluxo Alternativo	<ol style="list-style-type: none"> 1. Caso o professor deixe algum campo em branco ou tente inserir algum dado inválido, o sistema envia uma mensagem de erro; 2. O usuário confirma a informação e retorna ao passo 3 do fluxo principal.

Quadro 7 - Caso de Uso Manter Questão

Nome	Manter Questão
Atores	Professor
Pré-Condições	O Usuário deve ser professor e estar cadastrado e logado no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 6. O professor seleciona a opção “Questões” no menu de opções; 7. Em seguida o sistema retornará a página de cadastro de questão, caso o usuário deseje realizar operações de atualização ou exclusão, deverá pesquisar por tal primeiro; 8. O Professor preenche os dados corretamente e escolhe a opção “Cadastrar”; 9. Se não houver erros, o sistema salva os dados preenchidos; 10. O Sistema envia uma mensagem de confirmação;
Fluxo Alternativo	<ol style="list-style-type: none"> 3. Caso o professor deixe algum campo em branco ou tente inserir algum dado inválido, o sistema envia uma mensagem de erro; 4. O usuário confirma a informação e retorna ao passo 3 do fluxo principal.

Quadro 8 - Caso de Uso Manter Arquivo

Nome	Manter Arquivo
Atores	Professor
Pré-Condições	O Usuário deve ser professor e estar cadastrado e logado no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 11. O professor seleciona a opção “Arquivos” no menu de opções; 12. Em seguida o sistema retornará a página de cadastro de arquivos, caso o usuário deseje realizar operações de atualização ou exclusão, deverá pesquisar por tal primeiro; 13. O Professor preenche os dados corretamente e escolhe a opção “Enviar”; 14. Se não houver erros, o sistema salva os dados preenchidos; 15. O Sistema envia uma mensagem de confirmação;

Fluxo Alternativo	<p>5. Caso o professor deixe algum campo em branco ou tente inserir algum dado inválido, o sistema envia uma mensagem de erro;</p> <p>6. O usuário confirma a informação e retorna ao passo 3 do fluxo principal.</p>
--------------------------	---

Quadro 9 - Caso de Uso Manter Disciplina

Nome	Manter Disciplina
Atores	Professor
Pré-Condições	O Usuário deve ser professor e estar cadastrado e logado no sistema.
Fluxo Principal	<p>16. O professor seleciona a opção “Disciplina” no menu de opções;</p> <p>17. Em seguida o sistema retornará a página de cadastro de disciplinas, caso o usuário deseje realizar operações de atualização ou exclusão, deverá pesquisar por tal primeiro;</p> <p>18. O Professor preenche os dados corretamente e escolhe a opção “Cadastrar”;</p> <p>19. Se não houver erros, o sistema salva os dados preenchidos;</p> <p>20. O Sistema envia uma mensagem de confirmação;</p>
Fluxo Alternativo	<p>7. Caso o professor deixe algum campo em branco ou tente inserir algum dado inválido, o sistema envia uma mensagem de erro;</p> <p>8. O usuário confirma a informação e retorna ao passo 3 do fluxo principal.</p>

Quadro 10 - Caso de Uso Comentar Tópico

Nome	Comentar Tópico
Atores	Todos os usuários cadastrados no sistema
Pré-Condições	Estar cadastrado e logado no sistema.
Fluxo Principal	<p>21. O usuário seleciona o grupo em que está o tópico;</p> <p>22. Em seguida o sistema retornará a página principal do grupo</p> <p>23. O usuário digita as informações necessárias e clica em “Comentar”, caso o mesmo queira alterar ou excluir deverá escolher a opção apropriada;</p> <p>24. Se não houver erros, o sistema salva os dados preenchidos;</p>

	25. O Sistema envia uma mensagem de confirmação;
Fluxo Alternativo	N/A

Quadro 11 - Caso de Uso Participar de Grupo

Nome	Participar de Grupo
Atores	Todos os usuários cadastrados no sistema
Pré-Condições	Estar cadastrado e logado no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 1. O usuário seleciona o grupo em que deseja participar; 2. Em seguida o sistema retornará a página principal do grupo; 3. O usuário escolhe a opção “Participar”;
Fluxo Alternativo	N/A

Quadro 12 - Caso de Uso Download de Arquivo

Nome	Download de Arquivo
Atores	Aluno
Pré-Condições	Estar cadastrado e logado no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 1. O usuário seleciona a opção “Minhas Disciplinas” 2. Em seguida o sistema retornará a página principal; 3. O usuário escolhe a opção “Arquivos”; 4. Clica em “Baixar”;
Fluxo Alternativo	N/A

Quadro 13 - Caso de Uso Responder Exercícios


Nome	Responder Exercícios
Atores	Aluno
Pré-Condições	Estar cadastrado e logado no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 1. O usuário seleciona a opção “Minhas Disciplinas” 2. Em seguida o sistema retornará a página principal; 3. O usuário escolhe a opção “Exercícios”; 4. O usuário escolhe o exercício que deseja responder; 5. O usuário responde às perguntas e clica em “Enviar”;
Fluxo Alternativo	

Quadro 14 - Caso de Uso Manter Grupo de Discursão

Nome	Manter Grupo de Discursão
Atores	Aluno
Pré-Condições	O Usuário deve ser aluno e estar cadastrado e logado no sistema.
Fluxo Principal	<p>26. O aluno seleciona a opção “Grupos” no menu de opções;</p> <p>27. Em seguida o sistema retornará a página de cadastro de grupos, caso o usuário deseje realizar operações de atualização ou exclusão, deverá pesquisar por tal primeiro;</p> <p>28. O aluno preenche os dados corretamente e escolhe a opção “Cadastrar”;</p> <p>29. Se não houver erros, o sistema salva os dados preenchidos;</p> <p>30. O Sistema envia uma mensagem de confirmação;</p>
Fluxo Alternativo	<p>9. Caso o aluno deixe algum campo em branco ou tente inserir algum dado inválido, o sistema envia uma mensagem de erro;</p> <p>10. O usuário confirma a informação e retorna ao passo 3 do fluxo principal.</p>

Quadro 15 - Caso de Uso Manter Tópico

Nome	Manter Tópico
Atores	Aluno
Pré-Condições	O Usuário deve ser aluno e estar cadastrado e logado no sistema.
Fluxo Principal	<p>31. O aluno seleciona a opção “Meus Grupos” no menu de opções;</p> <p>32. Em seguida o sistema retornará a página de inicial do grupo;</p> <p>33. O aluno preenche os dados corretamente e escolhe a opção “Enviar”, caso o mesmo queira alterar ou remover o tópico, o mesmo deve escolher;</p> <p>34. Se não houver erros, o sistema salva os dados preenchidos;</p> <p>35. O Sistema envia uma mensagem de confirmação;</p>
Fluxo Alternativo	<p>11. Caso o aluno deixe algum campo em branco ou tente inserir algum dado inválido, o sistema envia uma mensagem de erro;</p>



12. O usuário confirma a informação e retorna ao passo 3 do fluxo principal.