

DESENVOLVIMENTO DO SEEA: sistema web de auxílio nas realizações de provas e apoio a aprendizagem em autoescola.

¹Hewerton Luís P. Santiago

²Kátya Araújo Pereira

RESUMO

A necessidade de agilizar processos no presente cotidiano é uma clara realidade e para gerenciar estes processos, diante de tal realidade, surgiu a oportunidade de desenvolver um software que irá atender as necessidades de uma autoescola, para que a mesma possa alcançar seus objetivos com precisão e qualidade em se tratando de melhoria do processo de ensino-aprendizagem nas realizações de provas. O software web tem o objetivo principal de cuidar destes processos, facilitando a elaboração de provas de treino, e dando apoio ao método de aprendizagem que é oferecido.

Palavras-chave: Avaliação. Aprendizagem. Processos. Qualidade.

ABSTRACT

The need to streamline processes in this routine is a clear reality and to manage these processes in the face of this reality, the opportunity arose to develop software that will meet the needs of a driving school, so that it can achieve its objectives with precision and quality when it comes to improving the teaching-learning process in the evidence of achievements. The web software has the main objective of taking care of these processes , facilitating the development of training events, and supporting the learning method that is offered.

¹ Mestre em Administração pela Universidade Metodista de Piracicaba (UNIMEP). Especialista em Administração de Sistemas de Informação pela Universidade Federal de Lavras (UFLA). Coordenador e professor do Curso de Sistemas de Informação – Faculdade Atenas.

² Acadêmica do Curso de Sistemas de Informação – Faculdade Atenas.

Keywords : Evaluation. Learning. Processes. Quality.

1 INTRODUÇÃO

Com o crescente avanço tecnológico, tudo tende a aderir novas formas de trabalhar processos e manipular dados para facilitar a vivência cotidiana. Dentro de uma autoescola o principal objetivo é habilitar pessoas/clientes para se tornarem condutores. Todo o processo de aprendizagem é feito dentro e fora da mesma que precisa ensinar e avaliar seus alunos da melhor forma possível. O problema é o tempo e o desgaste que o instrutor tem para montar, avaliar e acompanhar o desenvolvimento de seus alunos, fato que pode levá-los a desistir do curso ou simplesmente passar por ele sem aprender realmente o necessário para se tornarem bons condutores. Portanto, o ensino deve ser passado de forma clara e objetiva e as avaliações devem ser feitas com agilidade, eficiência e precisão. E é diante deste contexto que desenvolveu-se este trabalho. A proposta apresenta como resultado, um *software* com o apoio da tecnologia para facilitar os processos de profissionais de uma autoescola e saber se seus alunos/clientes estão realmente prontos para serem portadores de uma Carteira Nacional de Habilitação. Sabe-se que há uma dificuldade pelos alunos ao realizar provas de legislação, prova esta que dá permissão aos candidatos a liberação da licença para aulas de direção. Esta prova há um grande índice de reprovação, pois são específicas e exige muito raciocínio lógico. Diante dessa situação, o problema de pesquisa pauta-se em: **de que maneira um software, poderá auxiliar no processo de aprendizagem dos alunos de uma autoescola, por meio das realizações de simulados utilizando a web?**

Neste sentido, buscou-se modelar e desenvolver um *software* para elaborar provas e acompanhar o ensino e aprendizagem com a finalidade de auxiliar e avaliar o desempenho nos treinos dos clientes/alunos através dos dados alcançados nas provas. Como objetivos específicos, preocupou-se em oferecer maneiras de facilitar a elaboração de prova de acordo com níveis de dificuldades, possibilitar o acompanhamento o ciclo de aprendizagem dos alunos, oportunizar o desenvolvimento de atividades de forma mais rápida, proporcionar flexibilidade aos clientes/alunos e mensurar o processo de desempenho dos alunos.

O presente estudo parte de uma hipótese, levando em consideração a dificuldade dos alunos de uma autoescola a atingirem um bom resultado para aprovação nas provas de legislação. Acredita-se que a implantação deste software, possibilitará que as autoescolas façam um acompanhamento e ao mesmo deem treinamentos para reforçar o ensino e

aprendizagem destes alunos, de forma mais interativa e dinâmica. Assim poderá diagnosticar o desempenho das atividades dos alunos melhorando o índice de aprovação de alunos das

autoescolas.

A principal relevância no desenvolvimento deste trabalho é a necessidade de um sistema informatizado para autoescolas. Um sistema que auxilie nos processos de aprendizagem e principalmente das avaliações que mostram o desempenho dos seus alunos. E a partir das informações obtidas atingir uma melhor aprendizagem dos conhecimentos oferecidos, visando a qualidade na prática de aprender, potencializando os processos para a conquista da habilitação, focando na aprendizagem do aluno por meio de um programa de avaliação e aprendizagem. Percebe-se que as empresas desse setor, estão carentes de softwares que apoiem essas atividades, portanto, este trabalho é importante, pois, garante as instituições de ensino um processo de aprendizagem focado e simples que traga um excelente desempenho de cada aluno a partir de simulados de treino.

Para realização do desenvolvimento do sistema, inicialmente definiu-se o problema de pesquisa a ser investigado e explorado em busca de soluções, em seguida foi trabalhado a UML (*Unified Modeling Language*) para realizar a modelagem do sistema. Em seguida foi selecionada a linguagem de programação PHP para ser trabalhado juntamente com as ferramentas MySQL, HTML. Tais escolhas foram feitas a partir de embasamento teórico de vários autores renomados e pesquisas bibliográficas. Por termino do desenvolvimento do sistema foram realizados os devidos testes para confirmar a resolução do problema proposto.

2 TECNOLOGIAS E SUAS ABRANGÊNCIAS

Segundo Maximiniano (1993) uma organização é uma combinação de esforços individuais que tem por finalidade realizar propósitos coletivos. Por meio de uma organização torna-se possível perseguir e alcançar objetivos que seriam inatingíveis para uma pessoa. Neste sentido, Rezende (2005, p.28) diz que, “um sistema de informação pode ter um grande impacto na estratégia corporativa e no sucesso da empresa. Trazendo um grande suporte a tomada de decisão, vantagens estratégias, segurança nas informações e muitos outros fatores.” No mesmo contexto Audy (2005) afirma que o objetivo geral dos sistemas de informação é disponibilizar para a organização as informações necessárias para que ela atue em um determinado ambiente, sem que haja dificuldades de comunicação, pois de nada adianta ter a informação disponível sem que haja uma boa comunicação dentro da organização. É fundamental compreender os conceitos de linguagem de programação, assim como suas

características, de modo a adotar uma determinada linguagem para desenvolvimento de um software..

2.1 LINGUAGEM DE PROGRAMAÇÃO

Para Sebesta (2003) uma linguagem de programação é um conjunto de regras sintáticas e semânticas que são utilizadas para a comunicação entre programa e computador. A linguagem de programação definida para a criação do *software* é o PHP. De acordo com Niederauer (2011) o PHP é uma das linguagens mais utilizadas na web. Milhões de sites no mundo inteiro utilizam o PHP. A principal diferença em relação às outras linguagens é a capacidade que o PHP tem de interagir com o mundo web, transformando totalmente os sites que possuem páginas estáticas. As grandes vantagens do PHP podem ser descritas como: a facilidade de alterações nos arquivos que são direcionados para todas as páginas, é uma ferramenta gratuita, ele já vem embutido no HTML que é a ferramenta utilizadas junto com o PHP para escrever os códigos, quando é executado uma página no servidor todo o código é executado também, faz ligação com vários banco de dados como o SQLServer ou o MYSQL e oferece uma vasta portabilidade podendo ser executada em diversos sistemas operacionais como o Unix, Linux ou o Windows. Sebesta (2003) descreve:

Se uma linguagem tiver um grande número de diferentes construções, alguns programadores podem não estar familiarizados com todas elas. Isso pode levar ao uso inadequado de alguns recursos e ao desuso de outros que podem ser ou mais elegantes ou mais eficientes (ou ambos) do que aqueles usados. Pode, até mesmo, ser possível, como foi observado por Hoare (1973), usar recursos desconhecidos acidentalmente, com resultados bizarros. Portanto, um número menor de construções primitivas e um conjunto consistente de regras para combiná-las (isto é, ortogonalidade) é muito melhor do que, simplesmente, ter um número grande de primitivas. Um programador pode projetar uma solução para um problema complexo depois de aprender somente um conjunto simples de construções primitivas (SEBESTA, 2003, p.28-29).

Dominando essa linguagem, pode-se transformar aqueles sites estático, feitos em HTML puro, em sites interativos, utilizando todas as técnicas de programação que essa linguagem oferece. Entretanto, para desenvolver os mesmos com qualidade é necessário utilizar banco de dados.

2.2 BANCO DE DADOS

Segundo Elmasri e Navathe (2011, p.2-3), “os bancos de dados é uma coleção de

dados relacionados, são projetados e construídos com dados para uma finalidade específica.” Sendo assim, o banco de dados é responsável por todo armazenamento de dados para gerar uma determinada informação para a organização. De acordo com Beighley e Morrison (2010) o SQL do MySQL é uma linguagem de consulta usada para comunicar-se com um banco de dados de MySQL que por sua vez é onde é armazenado dados dentro de tabelas de banco de dados”. Portanto o gerenciamento de banco de dados do software utilizado será o MySQL. Para Siberschatz, Korth e Sudarshan (1999) um banco de dados deve oferecer a empresa informações claras, eficientes e seguras como descreve Sistemas de banco de dados são projetados para gerir grandes volumes de informações. O gerenciamento das informações implica a definição das estruturas de armazenamento das informações e a definição dos mecanismos para a manipulação dessas informações. Ao implementar o software de elaboração de provas, o banco de dados proporcionará diminuição do acúmulo de papel, agilidade na manipulação dos arquivos e eliminação de redundância, pois o sistema não permite o cadastro informações iguais. Apresentando também um padrão de armazenamento e a segurança na disponibilização de visualizações feitas somente para usuários que possuam senha de acesso e gerenciado pelo MySQL. Neste sentido, ainda ao se tratar de banco de dados, para facilitar o entendimento de como se dará a organização e interação com os dados no software, é necessário conhecer e compreender o papel de um diagrama de entidade e relacionamento.

2.2.1 DIAGRAMA DE ENTIDADE E RELACIONAMENTO

Para (Almeida, Oliveira e Coelho, 2010). Modelar conceitualmente uma realidade é uma das etapas da construção de um SI. Portanto um diagrama de entidade e relacionamento ou diagrama de classes é uma forma de representar as relações existentes entre as partes envolvidas na aplicação do *software* para facilitar o entendimento de qualquer indivíduo acerca do trabalho que está sendo desenvolvido. Estes precisam estar diretamente ligados a uma excelente Engenharia de Software.

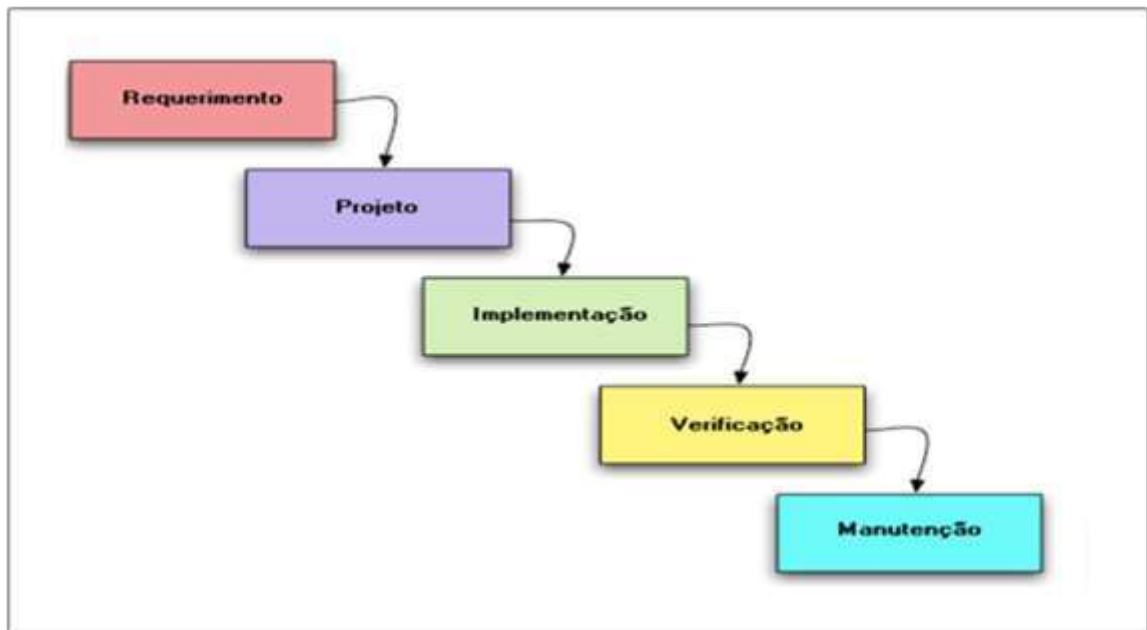
2.3 ENGENHARIA DE SOFTWARE

De acordo com Sommerville (2007) um modelo de processo de software é uma representação abstrata de um processo de *software*. O modelo de processo para criação do *software* deve ser único para cada projeto, pois, cada um apresenta suas particularidades. O

modelo de processo escolhido para este *software* foi o modelo cascata. De acordo com Pressman (2006) no modelo cascata, as fases definidas para o desenvolvimento do software são sistematicamente seguidas de maneira sequencial.

Para Pressman (1995) no desenvolvimento de um software estão englobados três elementos importantes sendo eles: métodos (como fazer), ferramentas (apoio automatizado ou semiautomatizado aos métodos) e procedimentos (ligação que mantêm juntos os métodos e as ferramentas). Os mesmos possibilitam o desenvolvimento de um software de qualidade. Neste projeto foi utilizado o modelo cascata que apresenta um fluxo linear e sequencial das atividades que devem ser realizadas conforme ilustra a Figura 01, incluindo uma área muito importante no desenvolvimento de software sendo esta, a Análise e Projeto de Sistemas.

FIGURA 01- Modelo cascata.



Fonte: Pressman (2006).

2.4 ANÁLISE E PROJETO DE SISTEMAS

Quando se deseja construir um *software* é fundamental fazer uma modelagem dos dados e do desenvolvimento do programa para que se tenha uma melhor visão do que está entrando e qual o objetivo a ser alcançado, tendo como entrada os requisitos do programa e como saída o produto que é fornecido. Segundo Lawrence (2004) a construção de um modelo de processo e a discussão de seus subprocessos ajuda a equipar a entender essa diferença entre o que deveria ocorrer e o que realmente ocorre. Seguindo uma modelagem, fica claro entender todo o comportamento de um *software* e o que deve ser feito para evitar erros. Com mesmo

objetivo de evitar erros e de extrema importância o levantamento de requisitos. De acordo com Wazlawick (2001, p.21-22) “o levantamento de requisito é o processo de descobrir quais são as funções que o sistema deve realizar e quais são as restrições que existem sobre essas funções”.

A etapa de levantamento de requisito corresponde a buscar todas as informações possíveis sobre as funções que o sistema deve executar e as restrições sobre as quais o sistema deve operar. Para Sommerville (2007, p.80-81) “os requisitos de sistemas são divididos em requisitos funcionais: “são declarações de serviços que o sistema deve fornecer, como o sistema reagir a entradas específicas e como o sistema deve se comportar em determinadas situações. Em alguns casos, os requisitos funcionais podem também estabelecer explicitamente o que o sistema não deve fazer. E requisitos não funcionais: são restrições sobre os serviços ou as funções oferecidas pelo sistema. Eles incluem restrições de timing, restrições sobre o processo de desenvolvimento e padrões. Os requisitos não funcionais aplicam-se frequentemente, ao sistema como um todo. Em geral, eles não se aplicam às características ou serviços individuais de sistema”. Neste artigo, foram desenvolvidos os requisitos propostos para o usuário principal que é o Instrutor que com o sistema *web* irá cuidar da geração de provas ganhando tempo para passar mais conhecimentos para seus alunos acerca do que eles realmente têm dificuldades de aprender. Depois de levantar os requisitos tornou-se possível elaborar o diagrama de caso de uso do sistema. Caso de uso é uma técnica amplamente utilizada na indústria para descrever e capturar os requisitos funcionais de software (RIBU, 2001). Segundo Cockburn (2005) os casos de uso são fundamentalmente uma forma textual, embora possam consistir em fluxogramas ou diagramas de sequência, redes de Petri ou linguagens de programação. Servem como meio de comunicação entre pessoas que muitas vezes não possuem treinamento especial. Portanto, texto simples geralmente é a melhor escolha.

Para Wazlawick (2004) os casos de uso devem corresponder aos principais processos de negócio da empresa e possivelmente são associados a vários requisitos funcionais. Para melhor entendimento do desenvolvimento do sistema foi feito uso do diagrama de caso de uso. Com tal finalidade também é elaborado o diagrama de entidade e relacionamento.

2.5 GESTÃO DE PROJETOS

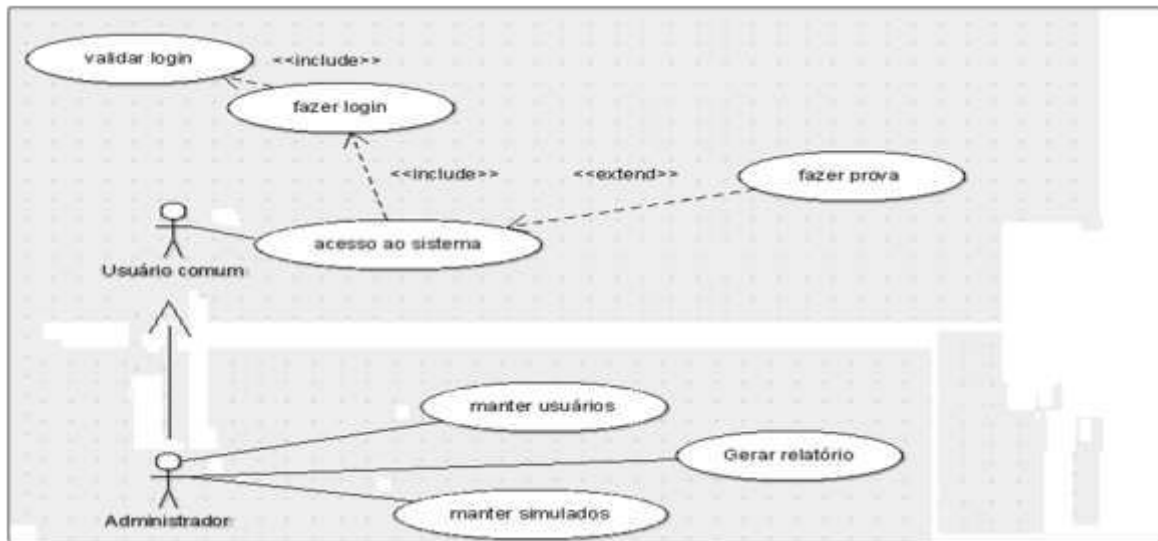
Segundo PMBOK (2004, p.8), “o gerenciamento de projeto é a aplicação de

conhecimentos, habilidades, ferramentas e técnicas às atividades do projeto a fim de atender os seus requisitos”. Os benefícios da gestão de um projeto de software é a redução no tempo de colocação de novos produtos no mercado, melhor controle e redução de custos, mais satisfação dos clientes e do próprio pessoal da empresa, Gestão de Projetos envolve ainda as pessoas, cliente, fornecedores. Fatores que vão além do simples uso de um *software*. Nos dizeres de Presman (1995, p.55) “para conduzir um processo de software bem sucedido, deve-se compreender o escopo do trabalho a ser feito, os riscos em que incorreremos os recursos exigidos, as tarefas a serem executados, os macros de referência a ser seguida. A gerência de projetos de *software* oferece essa compreensão.”. Para atingir o sucesso de um projeto deve-se sempre analisar os riscos e as incertezas que o envolve identificar os custos, tempo de elaboração e os conflitos que o cercam. Neste sentido, elaborou-se os documentos de Termo de Abertura e Termo de Encerramento. O Termo de Abertura é o documento que autoriza formalmente um projeto, concede aos gerentes do projeto a autoridade para aplicar os recursos organizacionais nas atividades do projeto, já o encerramento de contrato, aborda cada contrato aplicável ao projeto ou a uma de suas fases segundo (PMBOK, 2004).

3 PROJETO E DESENVOLVIMENTO DO SISTEMA SEEA

Neste capítulo, serão apresentados os diagramas e as principais interfaces do sistema como resultados deste trabalho. A interface do sistema se apresenta de forma clara e de fácil manuseio, permitindo que o usuário interaja com o sistema conseguindo entender facilmente o que cada elemento representa. Para que se tenha acesso ao sistema, é necessário que o usuário realize o *login* e suas informações sejam validadas. Diante dessa situação, realizou a modelagem de caso de uso conforme ilustra a Figura 01, na qual pode-se observar que existe dois atores, sendo um o usuário comum e o outro o administrador do sistema. Ambos podem realizar *login* e validar seu acesso.

FIGURA 01 – Representativa do caso de uso da tela de login.



Fonte: Elaborado pelos autores.

Conforme o modelo elaborado e apresentado anteriormente, desenvolveu-se a Tela de *Login* do sistema proposto conforme ilustra a Figura 02. Pode-se observar que a tela apresenta os campos “e-mail” e “senha” e um botão para o usuário enviar as informações para validação. Sendo validadas as informações de e-mail e senha do usuário, o mesmo terá acesso a tela principal do sistema SEEA. Tendo as informações validadas, o usuário terá acesso a tela Menu.

FIGURA 02 - Representativa da tela de login.

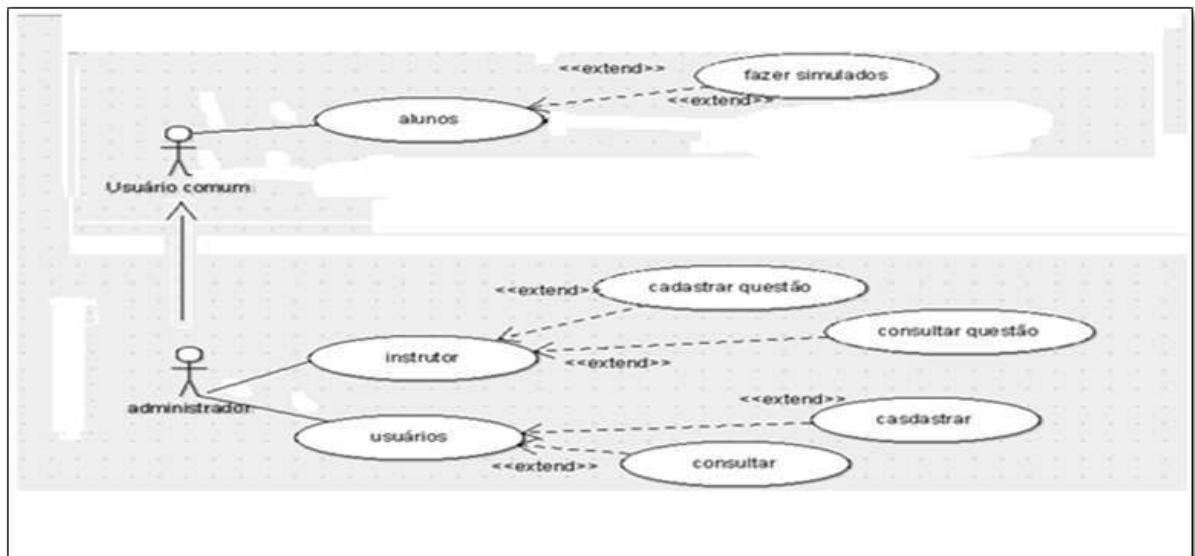


Fonte: Elaborado pelos autores.

A Figura 03 apresenta a modelagem de caso de uso da tela principal com dois

atores, sendo o usuário comum e o outro o administrador do sistema.

FIGURA 03 – Representativa do Caso de Uso da Tela Menu



Fonte: Elaborado pelos autores.

O usuário administrador tem acesso a todos os campos da tela, já o usuário comum tem acesso somente ao tópico permitido que é fazer simulado. Conforme o modelo elaborado e apresentado anteriormente, desenvolveu-se a Tela de Menu do sistema proposto, conforme ilustra a Figura 04.

FIGURA 04 – Representativa da tela menu.



Fonte: Elaborado pelos autores.

A tela apresenta uma barra e menus com os campos “Instrutor”, ”Simulados”, ”Relatórios” e ”Usuários”. Cada campo apresenta seus sub-tópicos. No campo instrutor eles são: “Cadastrar questão” e “Consultar questão”. No campo Simulado é: ”Fazer simulado”. Dentro do campo “Simulados” da barra de menu do sistema há o campo “Fazer simulado”. A Figura 05 apresenta a tela “Fazer simulado”.

FIGURA 05 – Representativa da tela fazer simulado.



Fonte: Elaborado pelo autor da pesquisa

A tela solicita ao usuário que informe o tipo de questão que deseja fazer dentre eles: “Legislação de trânsito”, “Sinalização de trânsito”, “Normas de circulação”, “Infrações e penalidades”, “Direção Defensiva”, “Primeiros socorros”, “Meio ambiente e convívio social” e “Noções de veículos”. Logo após o sistema solicita ao usuário que informe a quantidade de questões que deseja responder. Outra funcionalidade importante do sistema é gerar relatórios. Dentro do campo “Relatórios” da barra de menu do sistema há o campo “Gerar relatório”. A Figura 06 apresenta a modelagem da tela “Gerar relatório”. Nesta tela o usuário poderá gerar relatórios de desenvolvimentos de usuários. Nesta tela é apresentado apenas um ator que é o Administrador. Somente o administrador pode ter acesso a esta tela após realizar login. A tela solicita ao usuário que informe o código do aluno para trazer o relatório dos simulados.

FIGURA 06 - Representativa da tela gerar relatório.



Fonte: Elaborado pelo autor da pesquisa

3 CONCLUSÕES

Ao longo do desmembramento do projeto foi trabalhado o problema encontrado dentro de uma autoescola relacionado ao baixo nível de gerenciamento e elaboração das avaliações de treino e aprendizagem. Dentro deste conceito foi apresentada uma proposta para solucionar a falha que ali existia visando o bom andamento da empresa e satisfação dos clientes. A proposta para solucionar esta falha é apresentada em forma de um *software* desenvolvido para web que vem proporcionar a empresa mais eficiência e agilidade como geração de simulados de treino buscando um aprendizado perfeito e concreto. O resultado esperado foi obtido e as dificuldades sanadas com clareza e eficiência. O sistema funciona a partir de um banco de dados, onde, estão cadastradas questões divididas por tipo no qual ficam à disposição dos alunos via web para fazer simulados de treino. Por meio do sistema, a autoescola oferece para o aluno a oportunidade de se preparar embasado nos simulados para a tão esperada prova que é aplicada pelo DETRAN. O *software* está formulado e pode ser trabalhado dentro de autoescolas que é a empresa foco, como também dentro de outras instituições que necessite agilizar seus processos de elaboração de provas e aprendizado, bastando apenas montar um banco de dados acerca do assunto que será trabalhado e aproveitar todas as oportunidades que podem ser oferecidas pelo sistema. Garantindo ainda

que, seus alunos/ clientes tenham um ensino de qualidade, pois, gozaram de mais treino e tempo para tal. Tendo o *software* pronto, surgiram novas ideias para sanar novas deficiências e para tais projetos futuros pode-se acoplar ao sistema, funções como a geração de relatórios de satisfação dos alunos e instrutores.

REFERÊNCIAS

BEIGHLEY, Lynn; MORRISON, Michael. **Use a cabeça! PHP E MySQL**. 1. ed. Rio de Janeiro, 2010.

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de banco de dados**. 4. ed. São Paulo: Addison Wesley, 2005.

LAWRENCE, Shari Pfleeger. **Engenharia de Software**. 2 ed. São Paulo: Prentice Hall,2004.

MAXIMIANO, Antônio Cesar A. **Introdução a administração**. 3 ed. São Paulo: Editora Atlas, 1993.

NIEDERAUER, Juliano. **Desenvolvendo Websites com PHP**. 2 ed. São Paulo: Novatec, 2011.

PMBOK, Guia. **Um guia do Conjunto de Conhecimento de Projetos**. *Project Management Institute*. 3.ed. *NewtonSqaure, Pennsylvania: Global Standard*,2004.

PRESSMAN, Roger S. **Engenharia de software**. 5. ed. São Paulo: Makron Books, 2002.

REZENDE, Denis Alcides. **Engenharia de Software e sistemas de informação**. 3.ed. Rio de Janeiro: Brasport, 2005.

SEBESTA, Roberto W. **Conceitos de linguagem de programação**. 5 ed. Porto Alegre: Editora Bookman, 2003.

SILBERCHATZ, Abraham; KORTH Henry; SAUDASHAN, S. **Sistema de Banco de Dados**. Pearson Makron Books, 1999.

SOMMERVILLE, Ian. **Engenharia de Software**. 8 ed. São Paulo: Person Hall, 2007.

WAZLAWICK, Raul Sidnei. **Análise e Projeto de Sistemas de Informação Orientado a Objetos**. Rio de Janeiro: Elsevier, 2004.

RIBU, K. *Estimating Object-Oriented Software Projects with Use Cases*. Master of Science Thesis, University of Oslo. Department of Informatics, 2001.