



UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ

CAMPUS LUIZ MENEGHEL

RENAN LEON GARCIA

**DESENVOLVIMENTO DE UM APLICATIVO PARA
TREINO DE INIBIÇÃO COMPORTAMENTAL**

Bandeirantes

2013

RENAN LEON GARCIA

**DESENVOLVIMENTO DE UM APLICATIVO PARA
TREINO DE INIBIÇÃO COMPORTAMENTAL**

TCC apresentado à Universidade Estadual do Norte do Paraná – *campus* Luiz Meneghel – como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Me. Bruno Miguel Nogueira de Souza

Bandeirantes

2013

RENAN LEON GARCIA

**DESENVOLVIMENTO DE UM APLICATIVO PARA
TREINO DE INIBIÇÃO COMPORTAMENTAL**

TCC apresentado à Universidade Estadual do Norte do Paraná – *campus* Luiz Meneghel – como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

COMISSÃO EXAMINADORA

Prof. Me. Bruno Miguel Nogueira de Souza
UENP – *Campus* Luiz Meneghel

Prof. Me. Carlos Eduardo Ribeiro
UENP – *Campus* Luiz Meneghel

Prof. Estevan Braz Brandt Costa
UENP – *Campus* Luiz Meneghel

Bandeirantes, ___ de _____ de 2013

RESUMO

O Transtorno de Déficit de Atenção e Hiperatividade (TDAH) é uma patologia que faz com que a pessoa tenha que lidar com diversas dificuldades no seu cotidiano. Levando em consideração o alto custo financeiro para realizar um tratamento, o estresse de seus familiares, o prejuízo nas atividades acadêmicas e vocacionais, bem como efeitos negativos na autoestima das crianças e adolescentes com este tipo de transtorno, no presente trabalho foi desenvolvido um aplicativo como ferramenta de apoio ao profissional da saúde, que visa o treino da inibição comportamental, utilizando os conceitos de reforço diferencial de baixa taxa de resposta (DRL) para pessoas com diagnósticos de TDAH. O treino da inibição comportamental consiste em um teste padronizado que ao final deste, é possível gerar relatórios com os dados do treino do paciente, podendo assim, auxiliar o profissional quanto à tomada de decisão quanto ao tratamento.

Palavras-chave: TDAH, aplicativo, DRL, relatório, Sistemas de apoio à decisão.

ABSTRACT

Attention Deficit and Hyperactivity Disorder (ADHD) is a condition that causes a person to have to deal with various difficulties in their daily lives. Considering the high costs to perform a treatment, the stress of their family, loss academic and vocational activities, as well as negative effects on self-esteem of children and adolescents with this disorder, thus in this work an application was developed as a tool to support health professional training aimed at behavioral inhibition, using the concepts of differential reinforcement of low response rate (DRL) for people with diagnoses of ADHD. The training consists of behavioral inhibition in a standardized test that at the end of this, you can generate reports with the training data of the patient and can therefore assist the professional and decision making regarding treatment.

Keywords: ADHD, application, DRL, report, decision support system.

LISTA DE FIGURAS

FIGURA 1 Ambiente de Desenvolvimento do RPG <i>Maker</i>	16
FIGURA 2 Interface do <i>Construct 2</i>	18
FIGURA 3 Tabela de um Banco de Dados	20
FIGURA 4 Diagrama de Casos de Uso.....	27
FIGURA 5 Diagrama de Atividades do caso de uso Consultar Senha.....	28
FIGURA 6 Diagrama de Atividades do caso de uso Cadastrar Avaliador	29
FIGURA 7 Diagrama de Atividades do caso de uso Gerenciar Sujeito	29
FIGURA 8 Diagrama de Atividades do caso de uso Cadastrar Sujeito	30
FIGURA 9 Diagrama de Atividades do caso de uso Gerenciar Relatório.....	30
FIGURA 10 Diagrama de Atividades do caso de uso Treinar	31
FIGURA 11 Diagrama de Atividades do caso de uso Realizar Teste.....	31
FIGURA 12 Diagrama de Classes do Speed Car.....	33
FIGURA 13 Diagrama de Comunicação do caso de uso Consultar Senha.....	34
FIGURA 14 Diagrama de Comunicação do caso de uso Cadastrar Avaliador.....	35
FIGURA 15 Diagrama de Comunicação do caso de uso Gerenciar Sujeito.....	36
FIGURA 16 Diagrama de Comunicação do caso de uso Cadastrar Sujeito.....	37
FIGURA 17 Diagrama de Comunicação do caso de uso Gerenciar Relatório	38
FIGURA 18 Diagrama de Comunicação do caso de uso Treinar	39
FIGURA 19 Diagrama de Comunicação do caso de uso Realizar Teste	40
FIGURA 20 Diagrama de Máquina de Estado do Speed Car	41
FIGURA 21 Diagrama de Sequência do caso de uso Consultar Senha.....	42
FIGURA 22 Diagrama de Sequência do caso de uso Cadastrar Avaliador.....	43
FIGURA 23 Diagrama de Sequência do caso de uso Gerenciar Sujeito.....	44
FIGURA 24 Diagrama de Sequência do caso de uso Cadastrar Sujeito.....	45
FIGURA 25 Diagrama de Sequência do caso de uso Gerenciar Relatório	46
FIGURA 26 Diagrama de Sequência do caso de uso Treinar	47
FIGURA 27 Diagrama de Sequência do caso de uso Realizar Teste	48
FIGURA 28 Diagrama de Entidade e Relacionamento do Banco de Dados.....	49
FIGURA 29 Tela Inicial.....	51
FIGURA 30 Tela de cadastro de avaliador.....	51

FIGURA 31 Tela de recuperação de senha	52
FIGURA 32 Tela de seleção de sujeito	53
FIGURA 33 Tela de cadastro de sujeito.....	53
FIGURA 34 Tela de relatórios	54
FIGURA 35 Relatório de desempenho.....	55
FIGURA 36 Tela de seleção de modo de jogo.....	56
FIGURA 37 Tela de seleção de carro	57
FIGURA 38 Tela de seleção de pista.....	57
FIGURA 39 Tela de seleção de nível de dificuldade.....	58
FIGURA 40 Tela de corrida.....	59
FIGURA 41 PHP Cadastrar Avaliador.....	60

SUMÁRIO

1	INTRODUÇÃO.....	9
1.1	CONTEXTUALIZAÇÃO.....	9
1.2	FORMULAÇÃO E ESCOPO DO PROBLEMA.....	10
1.3	JUSTIFICATIVA.....	11
1.4	OBJETIVOS.....	11
1.4.1	Objetivo Geral.....	11
1.4.2	Objetivos Específicos.....	11
1.5	ORGANIZAÇÃO DO TRABALHO.....	11
2	FUNDAMENTAÇÃO TEÓRICA.....	13
2.1	TDAH.....	13
2.2	MOTORES DE JOGOS.....	15
2.2.1	RPG <i>Maker</i>	15
2.2.2	<i>Game Maker</i>	17
2.2.3	<i>Construct 2</i>	18
2.3	Banco de Dados MySQL.....	19
2.4	Linguagem de Programação PHP.....	21
2.5	Diagramas UML.....	21
3	MÉTODO.....	23
4	DESENVOLVIMENTO.....	26
4.1	Documentação.....	26
4.1.1	Diagrama de casos de uso.....	26
4.1.2	Diagramas de Atividades.....	28
4.1.3	Diagrama de Classes.....	32
4.1.4	Diagramas de Comunicação.....	34
4.1.5	Diagrama de Máquina de Estado.....	40
4.1.6	Diagramas de Sequência.....	42
4.2	Desenvolvimento do Aplicativo.....	49
4.3	Execução do Aplicativo.....	50
4.4	Comunicação entre o aplicativo e o banco de dados.....	59

CONCLUSÃO.....	62
REFERÊNCIAS.....	64
APÊNDICE A – Explicação dos casos de uso do Speed Car	67

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

O transtorno de déficit de atenção e hiperatividade (TDAH) é hoje um dos temas mais estudados em crianças em idade escolar. Estima-se que ele apresente uma das principais fontes de encaminhamento de crianças ao sistema de saúde. (SANTOS; VASCONCELOS, 2010).

De acordo com o Manual Diagnóstico e Estatístico de Transtornos Mentais (DSM) a característica principal do TDAH é um padrão persistente de desatenção e/ou hiperatividade, sendo mais frequente e severo do que aquele tipicamente observado em indivíduos com nível equivalente de desenvolvimento. Alguns sintomas hiperativo-impulsivos que causam prejuízo devem ter estado presentes antes dos 7 anos, mas muitos indivíduos são diagnosticados depois dessa idade, após a presença dos sintomas por alguns anos.

Rohde *et al.* (2000) define o TDAH como um distúrbio de desenvolvimento que se manifesta por meio de comportamentos considerados inadequados em que a pessoa não consegue controlar suas vontades e impulsos, com isso, acaba prejudicando seu nível de atenção. Suas principais características é a desatenção, a hiperatividade e impulsividade afetando negativamente o desempenho acadêmico, o relacionamento familiar e social e o ajustamento psicossocial.

Os estudos nacionais e internacionais situam a prevalência de TDAH entre 3% e 6%, sendo realizados com crianças em idade escolar na sua maioria. O impacto desse transtorno na sociedade é enorme, principalmente quando se apresenta em famílias de baixa renda, considerando o prejuízo nas atividades acadêmicas e vocacionais, o seu alto custo financeiro, o estresse nas famílias e bem como efeitos negativos de auto-estima das crianças e adolescentes. (ROHDE *et al.*, 2000)

Rohde e Mattos (2008) dizem que uma solução para auxiliar no tratamento de TDAH é a utilização de jogos para treino de inibição comportamental buscando o reforçamento diferencial de baixa taxa de resposta (DRL).

Para Dalpiaz (2004) DRL são testes realizados para que uma pessoa saiba esperar um momento oportuno para realizar determinada ação, por exemplo, uma criança que gosta muito de batatas fritas, ela só ganha as batatas para comer se ela pede para sua mãe a cada hora, se ela pede antes de uma hora ela não ganha as batatas e terá de esperar mais uma hora.

Baseado nas teorias apresentadas o aplicativo desenvolvido neste trabalho espera ajudar pessoas com déficit de atenção a esperar o momento mais adequado para realizar uma ação, mostrando assim, que quando se espera o tempo necessário existem mais vantagens do que realizar esta ação a todo momento sem analisar as circunstâncias em volta, com isso, espera-se que este treino possa transferir este tipo de comportamento ao seu cotidiano. Além do treino, o aplicativo irá gerar relatórios das partidas para serem analisados por um profissional da saúde e assim, auxiliar na tomada de decisão do profissional.

Para o desenvolvimento do aplicativo, foram utilizados: conceitos da Engenharia de Software para a documentação e projeto do software, o software *Construct 2* para a criação do aplicativo, a linguagem de programação PHP e banco de dados MySQL.

1.2 FORMULAÇÃO E ESCOPO DO PROBLEMA

Uma pesquisa realizada por Vasconcelos *et al.* (2003) com crianças da quarta série de uma escola para definir a prevalência do transtorno de déficit de atenção/hiperatividade (TDAH) aponta que de 403 alunos avaliados, sendo com idade entre 6 e 15 anos, 69 alunos (17,1%) apresentaram o diagnóstico de TDAH, e o tratamento, tanto psicológico quanto medicinal tem um custo muito elevado, principalmente para famílias de baixa renda. Para auxiliar nesse tratamento, o ideal seria uma ferramenta sistematizada para treino de inibição comportamental, que faria com que a criança pudesse transformar esse treino em seu cotidiano, auxiliando tanto no tratamento da TDAH quanto na sua vida psicossocial.

1.3 JUSTIFICATIVA

O desenvolvimento de um jogo para treino de inibição comportamental com base no conceito de testes de DRL vem como auxílio ao tratamento de TDAH e ferramenta de apoio ao profissional da área da saúde de modo a otimizar a geração de relatórios e a captura de dados dos pacientes, com isso, tornando possível um acompanhamento rotineiro do desempenho do paciente.

1.4 OBJETIVOS

1.4.1 Objetivo Geral

O objetivo deste trabalho é desenvolver um aplicativo para auxiliar o treino comportamental dos pacientes com TDAH.

1.4.2 Objetivos Específicos

Este trabalho tem como objetivos específicos:

- Analisar as ferramentas de apoio ao desenvolvimento de jogos;
- Desenvolver um aplicativo para treino comportamental;
- Documentar este aplicativo
- Tornar o aplicativo acessível por qualquer pessoa hospedando em um servidor web.

1.5 ORGANIZAÇÃO DO TRABALHO

O trabalho está dividido em cinco seções: Introdução, Fundamentação Teórica, Método, Desenvolvimento e Conclusão. A seção 1 traz uma contextualização do assunto que será abordado, a formulação e escopo do problema, o motivo para a realização deste projeto, seu objetivo geral e seus objetivos específicos.

Na Seção 2 são mostrados conceitos sobre TDAH, conceitos e exemplos de motores de jogos, conceitos de banco de dados MySQL, linguagem de programação PHP e diagramas UML.

Na Seção 3 é apresentado o método como o trabalho foi desenvolvido, descrevendo as ferramentas e metodologias utilizadas.

Na Seção 4 é apresentado o desenvolvimento do projeto, o levantamento de requisitos para a construção do aplicativo, a documentação do aplicativo, o desenvolvimento do aplicativo, a execução do aplicativo e a comunicação entre o aplicativo e o banco de dados.

Por último, na seção 5 é apresentada a conclusão, onde estão apresentadas as considerações finais do projeto.

2 FUNDAMENTAÇÃO TEÓRICA

No presente capítulo são abordados conceitos de TDAH, conceitos de motor (*engine*) de jogos e principais motores, conceitos de banco de dados MySQL, linguagem PHP e diagramas UML.

2.1 TDAH

O transtorno de déficit de atenção/hiperatividade (TDAH) é um distúrbio de desenvolvimento manifestado através de comportamentos considerados inadequados em que a criança não consegue controlar seus impulsos e vontades prejudicando seu nível de atenção (DALPIAZ, 2004). As características nucleares do transtorno de déficit de atenção/hiperatividade são a desatenção, a hiperatividade e impulsividade. Elas afetam de modo adverso o desempenho acadêmico, o relacionamento familiar e social, o ajustamento psicossocial e a vida laborativa e devem ser alvo de intervenção especializada (ROHDE; MATTOS, 2008).

De acordo com Rohde *et al.* (2000) os estudos nacionais e internacionais situam a prevalência de TDAH entre 3% e 6%, sendo realizados com crianças em idade escolar na sua maioria. O impacto desse transtorno na sociedade é enorme, considerando-se seu alto custo financeiro, o estresse nas famílias, o prejuízo nas atividades acadêmicas e vocacionais, bem como efeitos negativos na auto-estima das crianças e adolescentes. Estudos têm demonstrado que crianças com essa síndrome apresentam maior risco de desenvolverem outras doenças psiquiátricas na infância, adolescência e idade adulta.

O TDAH parece resultar de uma combinação complexa de fatores genéticos, biológicos, ambientais e sociais. Ao destacarem fatores genéticos no TDAH, alguns estudos indicam marcadores fenotípicos, bem como marcadores genéticos. (SANTOS; VASCONCELOS, 2010)

Costa *et al.* (2007) diz que na tarefa de reforçamento diferencial de baixas taxas (do inglês, “*differential reinforcement of low rates*”, DRL) há liberação de reforço apenas

quando a pressão a uma barra é espaçada por, no mínimo, um intervalo de tempo “ t ” definido pelo experimentador, denominado de tempo crítico. Caso a pressão na barra ocorra antes desse intervalo, não há liberação de reforço e o contador de tempo é zerado, reiniciando a contagem. Assim, a resposta deve ser omitida durante o intervalo de tempo crítico, podendo ser emitida depois de sua passagem, o que implica em um julgamento temporal.

Com a utilização de testes e atividades de treinamento de inibição comportamental de DRL no auxílio ao tratamento de TDAH, o paciente tem ao seu alcance ferramentas que irão auxiliá-lo de forma positiva no tratamento, buscando transferir esse reforçamento diferencial de baixa taxa de resposta para o seu cotidiano, com isso, ajudando-o no controle de suas vontades e impulsos.

Jogadores de videogame têm demonstrado um melhor desempenho em uma série de capacidades cognitivas, como atenção, processamento perceptual e funções executivas quando comparados a não jogadores de videogame, tais ganhos cognitivos podem ter implicações clínicas, especialmente com relação a transtornos nos quais a terapia medicamentosa tradicional não contempla todos os domínios cognitivos e comportamentais afetados, como no caso do TDAH. (BOOT *et. al.* 2008)

Segundo Rivero (2012), diversos estudos sugerem que os participantes que foram expostos a 10 horas ou mais de videogame apresentam melhora em diversas tarefas laborais que eram diferentes do videogame em si, levando a propor que a experiência de jogar videogame melhora funções cognitivas básicas, que podem ser generalizadas a tarefas e estímulos novos.

Para o desenvolvimento do jogo de treinamento inibição comportamental é necessário um motor de jogo que proporcione melhor auxílio para o trabalho. No próximo capítulo serão abordados alguns motores de jogos para que um deles seja utilizado no desenvolvimento do aplicativo.

2.2 MOTORES DE JOGOS

Para Bessa (2007), os motores de jogos visam a suprimir grande parte da demanda computacional, integrando diversas técnicas frequentemente necessárias nos jogos, simplificando muito a criação de um novo jogo.

O objetivo principal de um motor de jogos é agrupar funções fundamentais para o desenvolvimento de jogos, que podem se estender da interação com os periféricos de entrada até a renderização dos cenários e personagens. Isto certamente reduz o tempo total de produção, à medida que concentra a equipe de trabalho em atividades de mais alto nível. (BESSA, 2007).

A seguir serão apresentados três motores de jogos, *RPG Maker*, *Game Maker* e *Construct 2*.

2.2.1 RPG Maker

O *RPG Maker* é uma série de programa de computador ou conjunto de bibliotecas para desenvolvimento de RPGs (*role-playing games*) eletrônicos. Esta série foi criada pela ASC II e atualmente desenvolvida pela *Enterbrain*. (ROSA; MALTEMPI, 2007)

Lançado em 1988, o *RPG Maker* tem como sua última versão estável o *RPG Maker VX Ace* (15 de dezembro de 2011), possui suporte para os idiomas japonês e inglês, sendo um software proprietário de difícil utilização para construção de jogos gratuitos. (ROSA; MALTEMPI, 2003)

Bittencourt (2003) diz que o *RPG Maker* permite que os usuários criem seus próprios jogos de RPG e com algumas mudanças no sistema pode criar até outros tipos de jogos, utilizando o sistema permitido de 256 cores. No editor pode criar mapas tilesets (também conhecido como “chipset”). Ele apresenta uma linguagem de script embutida, o *Ruby Game Scripting System* (RGSS).

Segundo Rosa et al. (2008) o *RPG Maker* é distribuído em dois pacotes, o programa em si e o RTP (*Run Time Package*), contendo sons, fontes e gráficos padrão. Ao executar o programa, o *software* apresenta um ambiente de desenvolvimento, como é apresentado na Figura 1:

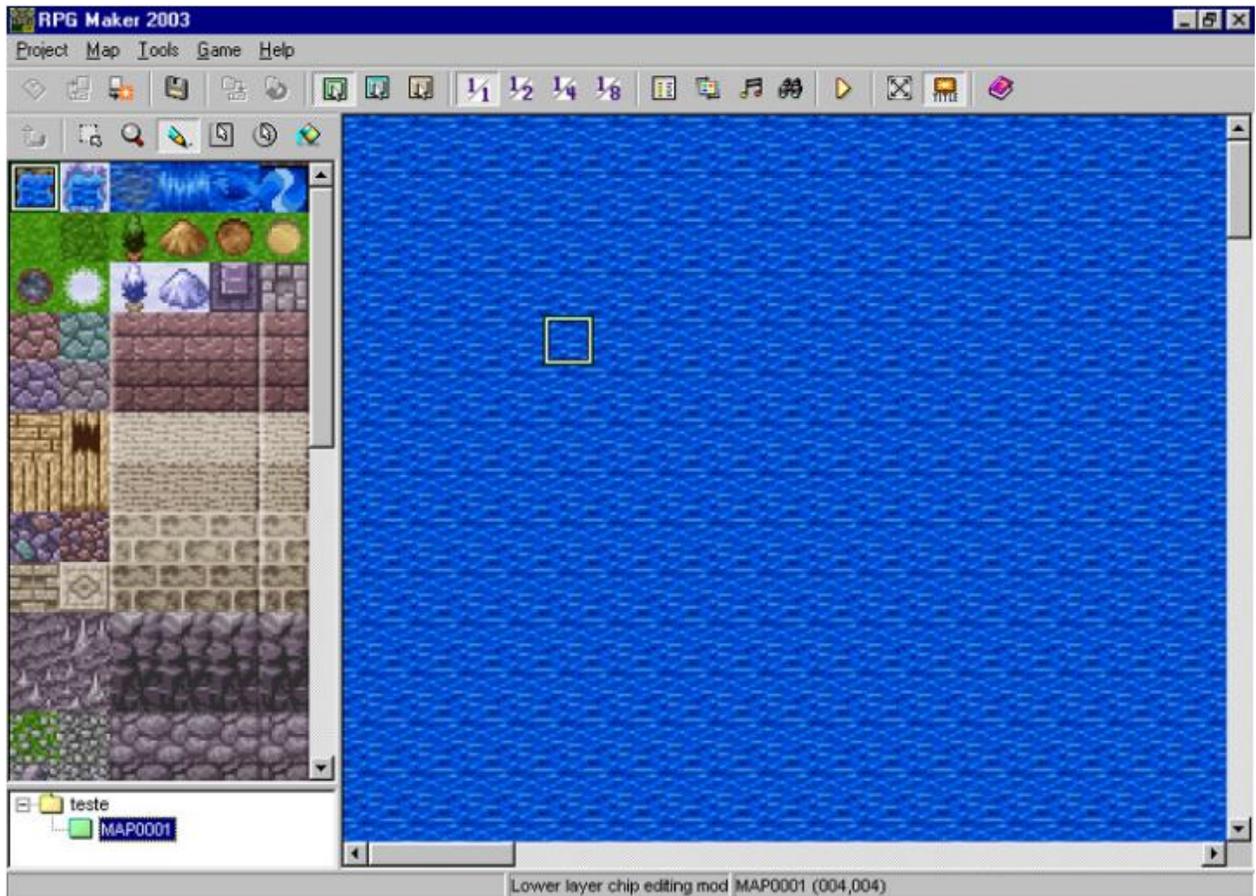


Figura 1 – Ambiente de Desenvolvimento do RPG *Maker*

Fonte: Rosa *et al.*, 2008.

Cavellucci (2003) apresenta quatro principais características do RPG *Maker* que fazem com que ele seja fácil de entender e de utilizar, como é mostrado na Tabela 1.

Características	Descrição
Eventos	A programação dos jogos gira em torno de eventos que são controlados por comandos de eventos e fazem qualquer coisa dentro do jogos, desde uma simples conversa entre heróis até um complicado mini-game de tiro.
Switches	<i>Switches</i> quer dizer interruptor, algo que liga ou desliga. Ele funciona como uma exceção para um evento realizado.
Variáveis	O princípio básico de uma variável é um espaço de memória identificado que seja capaz de realizar operações com números, utilizado principalmente para armazenar pontos obtidos no jogo.
Condições	São os comando de eventos que criam uma necessidade para que algo aconteça.

Tabela 1 – Características do RPG *Maker*

Fonte Adaptada: Cavellucci, 2003.

Apesar do RPG Maker ser uma ferramenta excelente para desenvolvimento de jogos, por ser um software proprietário e a linguagem de desenvolvimento não ser voltada para aplicações web, a utilização deste software se torna inviável, uma vez que o aplicativo desenvolvido será livre e disponível em um servidor web.

2.2.2 *Game Maker*

Segundo Francisco *et al.* (2012) o *Game Maker* foi criado em 1999, sendo baseado em uma linguagem de *scripts* (GML) e se destina a criação de jogos 2D e 3D.

Este motor de jogo é *software* proprietário e comercial, não possui seu código livre e sua versão completa deve ser comercialmente adquirida. Sua maior característica é a utilização de uma linguagem própria, a GML. É uma linguagem de sintaxe simples e de fácil aprendizagem, além de já possuir itens práticos exclusivos para o desenvolvimento de jogos eletrônicos. (MORAIS, 2009)

Morais (2009) define a GLM como uma linguagem orientada a objeto flexível, ou extensível, podendo atribuir novas funções que não estão naturalmente embutidas na linguagem. Sua sintaxe é parecida com a da linguagem C, embora o *Game Maker* tenha sido desenvolvido em *Delphi*, o qual é o motivo de apenas existir para a plataforma Windows.

O *Game Maker*, além de trabalhar com a GML, também utiliza um sistema facilitador, chamado de *drag'n'drop*, em que não é necessário utilizar linhas de código para programar as ações do jogo, mas sim, utiliza-se de comando pré-definidos, de fácil entendimento, que atuam como o código GML. (MORAIS, 2009)

2.2.3 Construct 2

Segundo Scirra (2012), o *Construct 2* é um criador de jogos em HTML 5, não sendo necessário a programação via linha de código, ele possui versões pagas para a criação de jogos comerciais e uma versão livre.

Para o desenvolvimento do jogo, o *Construct 2* utiliza um sistema *Drag'n'Drop* (arrastar e largar), que é a ação de clicar em um objeto virtual e arrastá-lo para uma posição diferente, ou sobre um outro objeto virtual.

Na Figura 2 é mostrado a interface inicial do *Construct 2*.

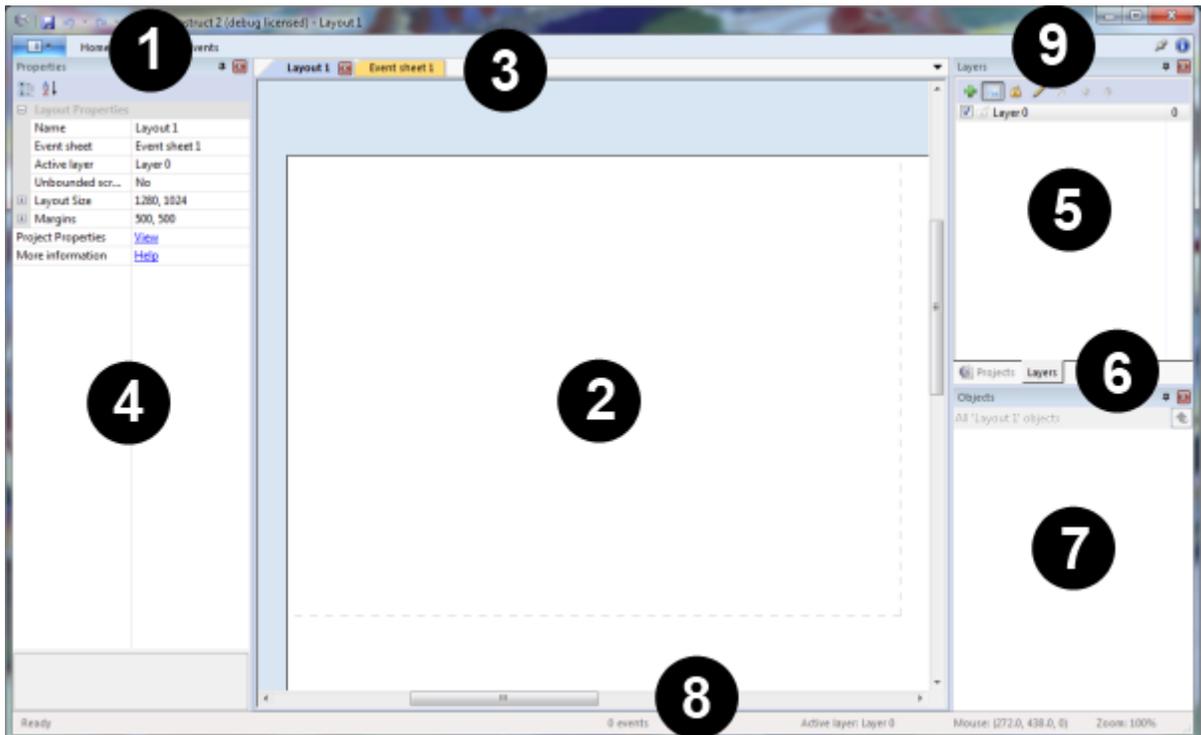


Figura 2 – Interface do *Construct 2*

Fonte: SCIRRA (2012).

De acordo com o criador deste motor de jogo, o *Construct 2* usa uma barra de abas semelhante ao *Office 2007* com comandos de Salvar, Desfazer, Refazer e

Visualizar, como mostrado o item 1 da Figura 3. O item 2 é o modo de exibição do *Layout*, que é onde são colocados objetos para os níveis de design. O item 3 é a visualização de guias, que permitem alternar entre as diferentes vistas de *layout* para definir a lógica baseada em eventos do jogo, dividida em *Layouts* e Folha de Eventos. O item 4 é a barra de propriedades, onde visualiza e altera propriedade para *layouts*, objetos, camadas e muito mais. O item 5 é o barra do projeto e a barra de comando, onde dá uma visão geral de tudo do projeto e é utilizado para adicionar novas camadas de um *layout*. O item 6 são guias para alternar entre a barra do projeto e a barra de comando. O item 7 é a barra de objeto, onde contém uma lista de objetos presentes no jogo. O item 8 é uma barra de status que mostra algumas propriedades que estão acontecendo no desenvolvimento. E por último, o item 9 são os botões como as de janelas usuais contendo minimizar, maximizar, e fechar.

Para a realização deste trabalho será utilizado a ferramenta *Construct 2* no desenvolvimento do aplicativo.

Um dos pontos principais da escolha desta ferramenta foi o fato de o *Construct 2* ser de código aberto e ser um aplicativo livre, tendo com isso uma versão gratuita para o desenvolvimento, além disso, os jogos são construídos com base nos novos padrões do HTML 5, sendo que o HTML é a linguagem base para a estruturação de sites e possui suporte na maior parte dos browsers e dispositivos móveis.

A interface do *Construct 2* é outro ponto importante, ela é bastante simples e fácil de manusear, utilizando um sistema *Drag'n'Drop* que faz o desenvolvimento ser mais interativo e mais fácil de construir os jogos, sem a necessidade de conhecer profundamente a linguagem HTML 5.

O *Construct 2* cria jogos para serem hospedados em servidores web, podendo ser acessado por qualquer pessoa em qualquer localização, e com ele podemos gerar relatórios para serem baixados pelos usuários.

2.3 Banco de Dados MySQL

Banco de dados é o conjunto de dados integrados que tem por objetivo atender a uma comunidade de usuário. (HEUSER, 1998)

Date (2003) afirma que para a manipulação dos dados em um banco de dados, se faz necessário a utilização de um Sistema Gerenciador de Banco de Dados (SGBD), que é basicamente um sistema computadorizado de manutenção de registros cuja a finalidade geral é armazenar informações e permitir que os usuários busquem e atualizem essas informações quando as solicitar.

SGBD é um software que incorpora as funções de definição, recuperação e alteração de dados de um banco de dados, e afirma que dentre os tipos de SGBD existentes, hoje há um claro predomínio dos SGBD relacionais, principalmente fora das plataformas de grande porte, além disso, muitos conceitos usados no projeto de banco de dados foram criados em combinação com a abordagem relacional. (HEUSER, 1998)

De acordo com Heuser (1998) um banco de dados relacional é composto por tabela, sendo uma tabela um conjunto não ordenado de linhas, cada linha é composta por uma série e campos e cada campo é identificado por um nome do campo. O conjunto de campos das linhas de uma tabela que possuem o mesmo nome formam uma coluna, como é mostrado na figura 3:

Emp		nome do campo (nome do atributo)	
CódigoEmp	Nome	CodigoDepto	CategFuncional
E5	Souza	D1	C5
E3	Santos	D2	C5
E2	Silva	D1	C2
E1	Soares	D1	—

coluna (atributo)

Linha (Tupla)

valor de campo
(valor de atributo)

Figura 3 – Tabela de um Banco de Dados

Fonte: Heuser, 1998.

Para Niederauer (2008), o MySQL é um SGBD relacional que utiliza a linguagem padrão SQL (*Structured Query Language*), e é largamente utilizado em aplicações para a internet. É o mais utilizado entre os banco de dados com código-fonte aberto.

Gonzaga e Birckan (2000) descrevem o MySQL como um servidor de banco de dados SQL multi-usuário e cliente-servidor, que consiste de um servidor e diferentes programas clientes e bibliotecas.

O MySQL é uma alternativa atrativa porque, mesmo possuindo uma tecnologia complexa de banco de dados, seu custo é baixo. Têm como destaque suas características de velocidade, escalabilidade e confiabilidade, o que vem fazendo com que ele seja adotado por departamentos de TI (Tecnologia da Informação), desenvolvedores web e vendedores de pacotes de softwares. (NIEDERAUER, 2008).

O servidor MySQL é também rápido e flexível o suficiente para permitir armazenar *logs* e figuras nele, As principais vantagens do MySQL são velocidade, robustez e facilidade de uso. (GONZAGA; BIRCKAN, 2000),

2.4 Linguagem de Programação PHP

De acordo com o Manual Oficial do PHP, PHP é uma linguagem de programação de ampla utilização, interpretada, que é especialmente interessante para desenvolvimento para a Web e pode ser mesclada dentro do código HTML. O objetivo principal da linguagem é permitir a desenvolvedores escreverem páginas que serão geradas dinamicamente e com rapidez.

O PHP é uma das linguagens mais utilizadas na web. Milhões de sites no mundo utilizam 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 websites que possuem páginas estáticas. Uma das grandes vantagens do PHP é que ele é gratuito. (NIEDERAUER, 2011)

2.5 Diagramas UML

A UML (Unifies Modeling Language) é uma linguagem-padrão para a elaboração da estrutura de projetos de software. Ela pode ser empregada para a visualização, a especificação, a construção e a documentação de artefatos que façam uso de sistemas complexos de software. (BOOCH *et al.*, 2005)

Fowler (2005) diz que UML é uma família de notações gráficas apoiada por um metamodelo único, que ajuda na descrição e no projeto de sistemas de *software*, particularmente daqueles construídos utilizando o estilo orientado a objetos.

As linguagens gráficas de modelagem existem há muito tempo na indústria do *software*. O propulsor fundamental por trás de todas elas é que as linguagens de

programação não estão em um nível de abstração suficientemente alto para facilitar as discussões sobre projeto. (FOWLER, 2005)

Booch *et al.* (2005) diz que a UML é adequada para a modelagem de sistemas, cuja abrangência poderá incluir sistemas de informação corporativos a serem distribuídos a aplicações baseadas em Web e até sistemas complexos embutidos de tempo real. Ela é uma linguagem muito expressiva, abrangendo todas as visões necessárias ao desenvolvimento e implantação desses sistemas.

Para Fowler (2005), a UML é indispensável na documentação de um *software*, um bom diagrama frequentemente pode ajudar a transmitir ideias sobre um projeto, particularmente quando você quer evitar muitos detalhes. Os diagramas também podem ajudá-lo a entender um sistema de *software*, ou um processo de negócio. Como parte de uma equipe tentando descobrir algo, os diagramas ajudam toda a equipe tanto a entender como comunicar esse entendimento.

Na próxima sessão será apresentado o modo como o trabalho será desenvolvido.

3 MÉTODO

Este trabalho foi desenvolvido baseado no conceito de DRL. Foi utilizado a ferramenta *Construct 2* para o desenvolvimento do aplicativo em linguagem de programação HTML 5. Para a documentação do aplicativo foram utilizados diagramas UML. Para o armazenamento dos dados extraídos do aplicativo, foi utilizado como sistema gerenciador de banco de dados o MySQL, e para a comunicação entre o aplicativo e o banco de dados foi utilizado a linguagem de programação PHP.

De acordo com Junior e Campos (2008) o levantamento de requisitos é a primeira etapa para o desenvolvimento de sistemas de informação, sendo ela responsável por identificar e modelar as necessidades do negócio a serem atendidas pelos sistemas de informação, e é, portanto, uma atividade cada vez mais relevante em um dinâmico cenário.

Para o desenvolvimento deste aplicativo, foram consultados pesquisadores e professores de psicologia da Universidade Estadual do Norte do Paraná (UENP) e da Universidade Federal de São Paulo (UNIFESP) que elencaram os requisitos principais do aplicativo.

Com base nas reuniões realizadas com os pesquisadores e professores, foram definidos alguns das principais características do aplicativo. Primeiramente o aplicativo deve trabalhar com um avaliador, que seria um profissional da área de saúde, como um psicólogo, para o acompanhamento dos resultados de seus pacientes, e sujeitos, que seriam os pacientes com prevalência de TDAH, sendo que um avaliador pode ter vários sujeitos cadastrados no aplicativo.

Portanto, ficou definido que o avaliador irá se cadastrar no sistema, em seguida irá cadastrar seus pacientes que utilizarão o aplicativo como treino de inibição comportamental.

Com o modo de cadastro do avaliador e de seus pacientes definido, o próximo passo foi definir o teste, baseado no conceito de DRL, que os pacientes irão realizar. Como o teste está voltado principalmente para crianças, foi definido que o teste seria

um jogo de corrida, com o nome de Speed Car, que chamará a atenção dos pacientes e incentivá-los a realizar o teste.

O aplicativo possui três pistas de corrida diferentes e três níveis de dificuldade e para realizar o teste o paciente deve correr nas três pistas, nas três dificuldades, totalizando nove corridas diferentes, sendo que cada corrida possui dez voltas, com um total de noventa voltas no teste completo. No início do teste, o paciente escolhe um tipo de carro que deseja utilizar, seu adversário será outro veículo que estará correndo ao mesmo tempo tentando vencê-lo comandado pelo aplicativo.

Neste aplicativo, o veículo do paciente possui turbo indicado por uma barra que é preenchida de forma gradativa e quando acionada o veículo do paciente terá um aumento de velocidade por um determinado tempo. É nessa barra de turbo que se concentra o conceito de DRL, sendo que, se o paciente acionar o turbo antes de preencher totalmente a barra, o veículo terá um aumento de sua velocidade pouco significativo por pouco tempo, em contrapartida, se ele esperar a barra encher completamente, seus benefícios serão maiores, uma vez que o aumento da velocidade de seu veículo será maior e o tempo de acréscimo perdura por mais tempo.

Com base nos conceitos de DRL, o aplicativo deve proporcionar um número significativo de chances para o paciente realizar um estímulo para que o avaliador possa visualizar uma evolução ou não dele durante o teste. A barra de turbo demora aproximadamente meia volta para se encher totalmente, conseguindo se encher por completo uma vez por volta, sendo que quando acionado o turbo ela só volta a começar a encher após o término do aumento de velocidade do veículo, o que resulta em aproximadamente noventa chances para o paciente acionar o turbo durante a execução do teste.

Para o monitoramento do desempenho dos pacientes, se faz necessário a extração de alguns dados do teste realizado pelo paciente, dentre os dados a serem extraídos estão o número de vezes que acionou o turbo antes de encher totalmente, o número de vezes que acionou o turbo cheio, o total de estímulos realizados, vezes que o veículo colidiu com qualquer objeto da pista, tempo para acionar o turbo depois de cheio, tempo para acionar o turbo antes de encher totalmente, tempo total de cada corrida e se no momento do estímulo a barra estava cheia ou não.

Após o jogo concluído e os testes realizados, ele foi postado em um servidor Web, no qual pode ser acessado que qualquer pessoa através do endereço <<http://ramses.uenp.edu.br/projetos/neurociencia/atentioncar/>>, principalmente por profissionais da área da saúde e seus pacientes, possam ter acesso e utilizar o aplicativo gratuitamente.

4 DESENVOLVIMENTO

No presente capítulo são expostos como foi realizada a documentação e o desenvolvimento do aplicativo, assim como, é mostrada a execução do aplicativo.

4.1 Documentação

Para a construção da documentação do aplicativo, foram utilizados alguns diagramas de UML, dentre eles está o diagrama de casos de uso, diagrama de atividades de cada caso de uso, diagrama de classe, diagrama de comunicação de cada caso de uso, diagrama de máquina de estado e diagrama de sequência de cada caso de uso.

4.1.1 Diagrama de casos de uso

O diagrama de casos de uso corresponde a uma visão externa do sistema e representa graficamente os atores, os casos de uso e os relacionamentos entre estes elementos e tem como objetivo ilustrar em um nível alto de abstração quais elementos externos interagem com que funcionalidade do sistema. A finalidade de um diagrama de casos de uso é apresentar um tipo de diagrama de contexto que apresenta os elementos externos de um sistema e as maneiras segundo as quais eles as utilizam. Na figura 4 é apresentado o diagrama de casos de uso do Speed Car:

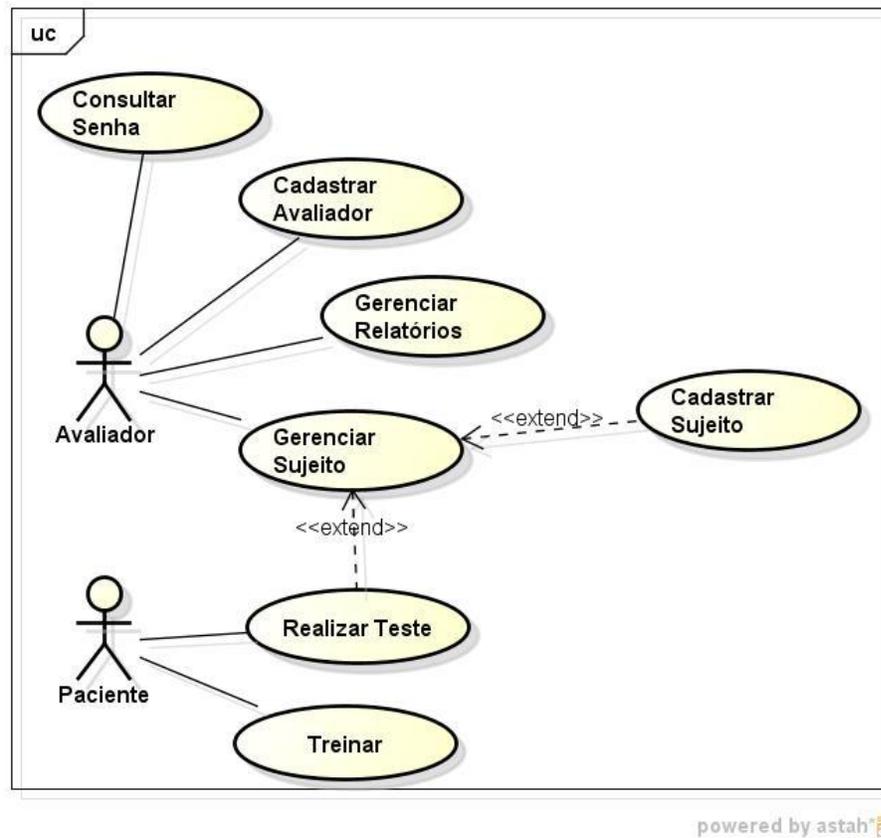


Figura 4 – Diagrama de Casos de Uso

Conforme a figura 4 apresentada, dois atores irão interagir com o aplicativo, o avaliador, como profissionais da área da saúde, e sujeito, como pacientes com diagnósticos de Transtorno de Déficit de Atenção e Hiperatividade.

No início, o avaliador irá se cadastrar no aplicativo, se caso já tiver cadastro e não lembrar sua senha, ele poderá consultar sua senha já cadastrada no sistema. Após logar no aplicativo, o avaliador poderá selecionar um sujeito para realizar o teste, excluir ou cadastrar seus pacientes, bem como gerar relatórios de testes já realizados por um sujeito selecionado.

O sujeito selecionado poderá realizar o teste ou simplesmente treinar para se familiarizar com o aplicativo. No modo treino, o sujeito seleciona um carro que deseja utilizar, escolhe a pista e o nível de dificuldade que deseja correr, sendo que no modo teste ele seleciona um carro, visualiza a pista e a dificuldade que irá correr, pois é obrigatório correr nas três pistas e nos três níveis de dificuldade em cada pista, com um total de nove corridas. Por fim, o sujeito realiza uma corrida de 10 voltas contra um carro guiado pelo sistema.

A explicação de cada caso de uso, com sua descrição, ator envolvido, pré-requisito para o caso de uso, fluxo principal e alternativo e pós-condição estão presentes no Apêndice A.

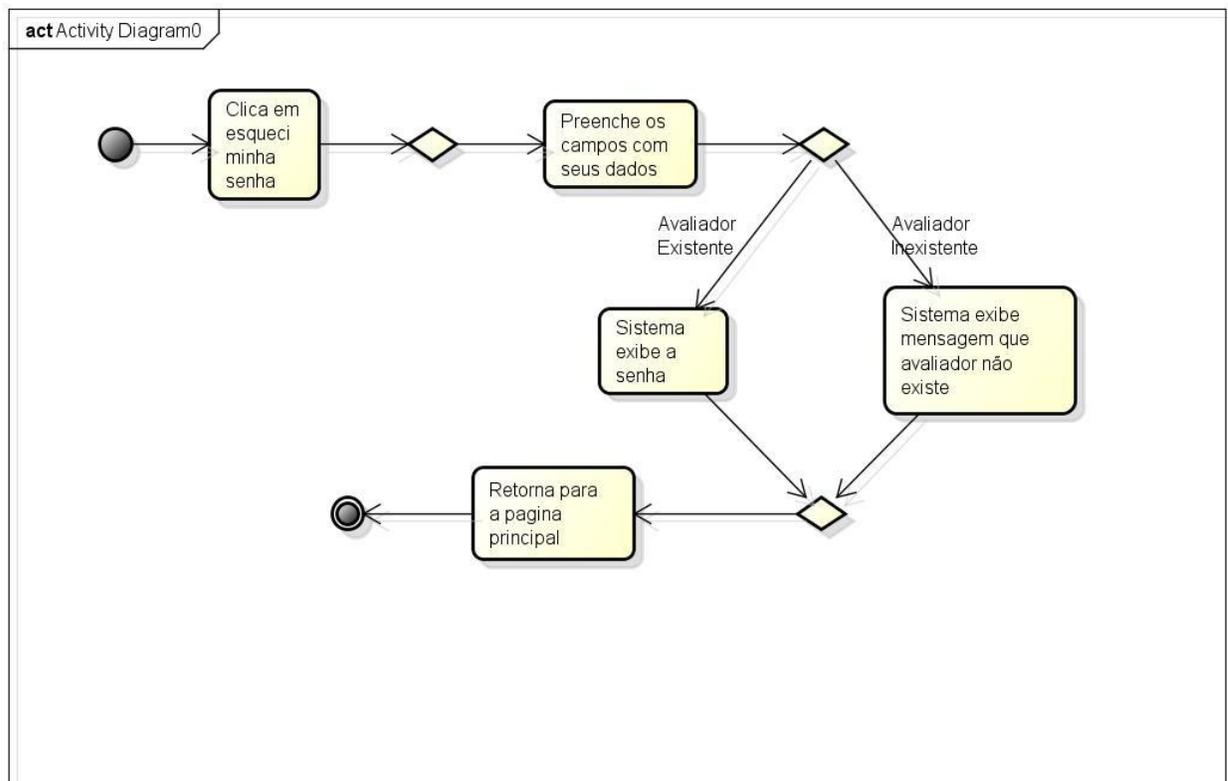
4.1.2 Diagramas de Atividades

O diagrama de atividades é utilizado para modelar o aspecto comportamental de processos. Neste diagrama, uma atividade é modelada como uma sequência estruturada de ações, controladas potencialmente por nós de decisão e sincronismo. (BOOCH *et al.*, 2005)

Para uma documentação completa do sistema, cada caso de uso deve ter um diagrama de atividades, a seguir serão apresentados os diagramas do aplicativo Speed Car.

Consultar Senha

Na figura 5 é mostrado o diagrama de atividades do caso de uso Consultar Senha.

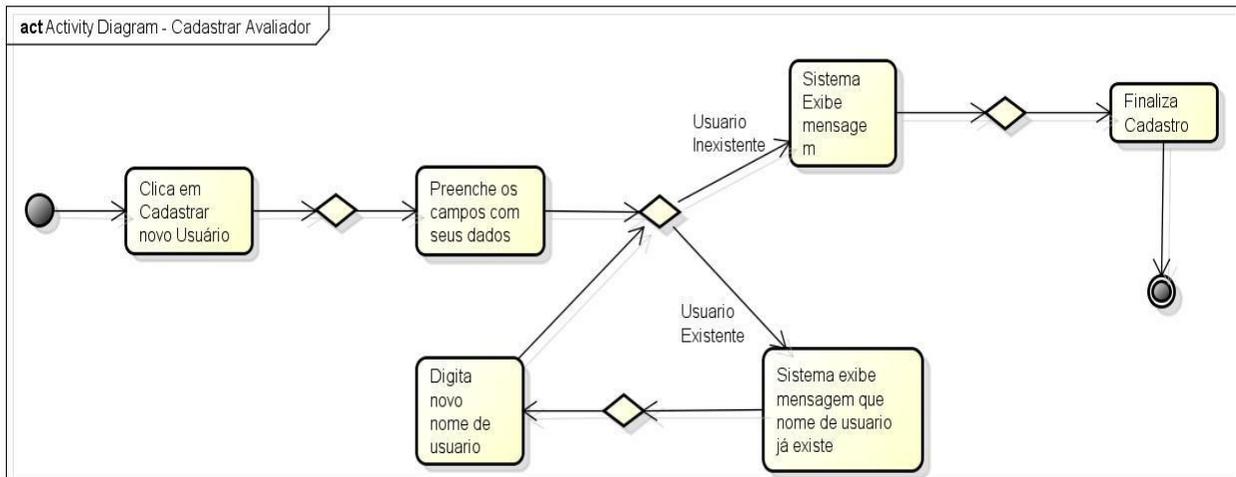


powered by astah

Figura 5 – Diagrama de Atividades do caso de uso Consultar Senha

Cadastrar Avaliador

Na figura 6 é apresentado o diagrama de atividades do caso de uso Cadastrar Avaliador.

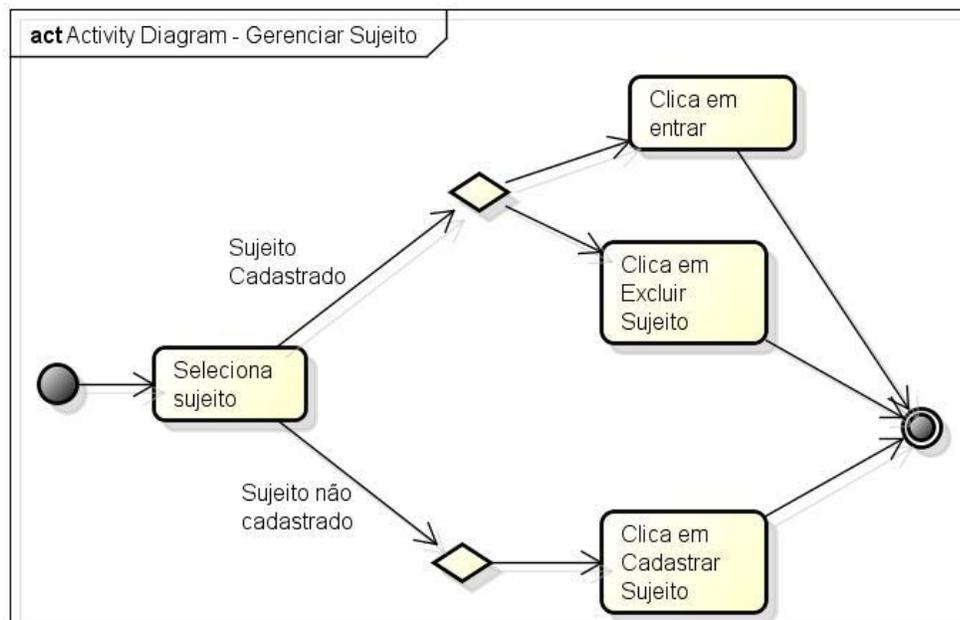


powered by astah®

Figura 6 – Diagrama de Atividades do caso de uso Cadastrar Avaliador

Gerenciar Sujeito

Na figura 7 é mostrado o diagrama de atividades do caso de uso Gerenciar Sujeito.



powered by astah®

Figura 7 – Diagrama de Atividades do caso de uso Gerenciar Sujeito

Cadastrar Sujeito

Na figura 8 é apresentado o diagrama de atividades do caso de uso Cadastrar Sujeito.

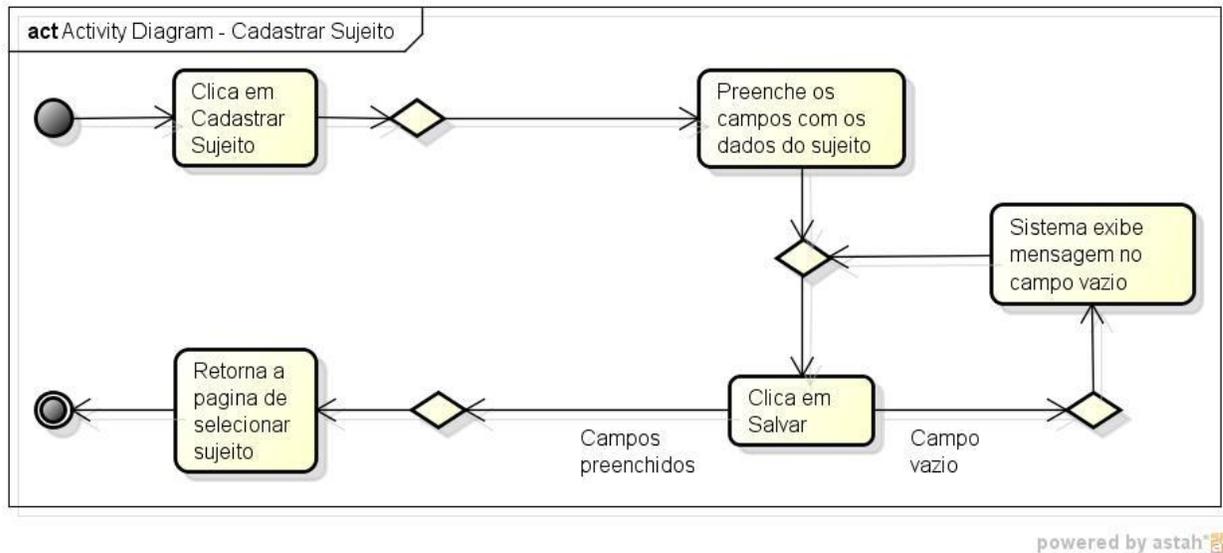


Figura 8 – Diagrama de Atividades do caso de uso Cadastrar Sujeito

Gerenciar Relatórios

Na figura 9 é apresentado o diagrama de atividades do caso de uso Gerenciar Relatórios.

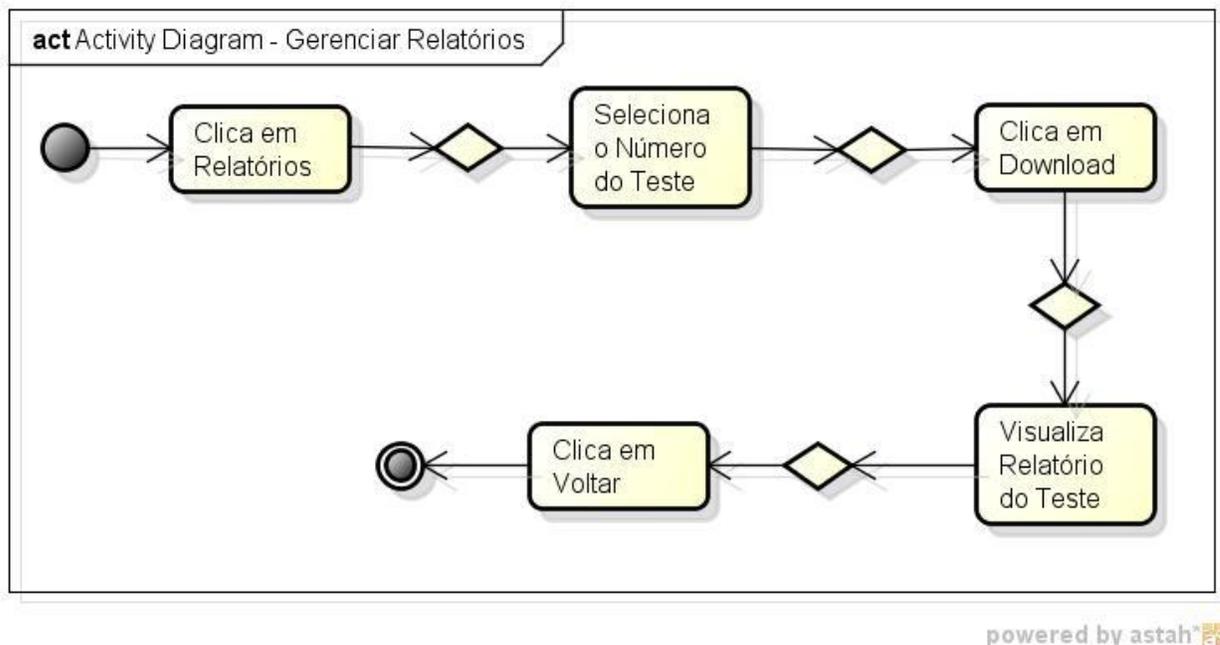


Figura 9 – Diagrama de Atividades do caso de uso Gerenciar Relatórios

Treinar

Na figura 10 é mostrado o diagrama de atividades do caso de uso Treinar.

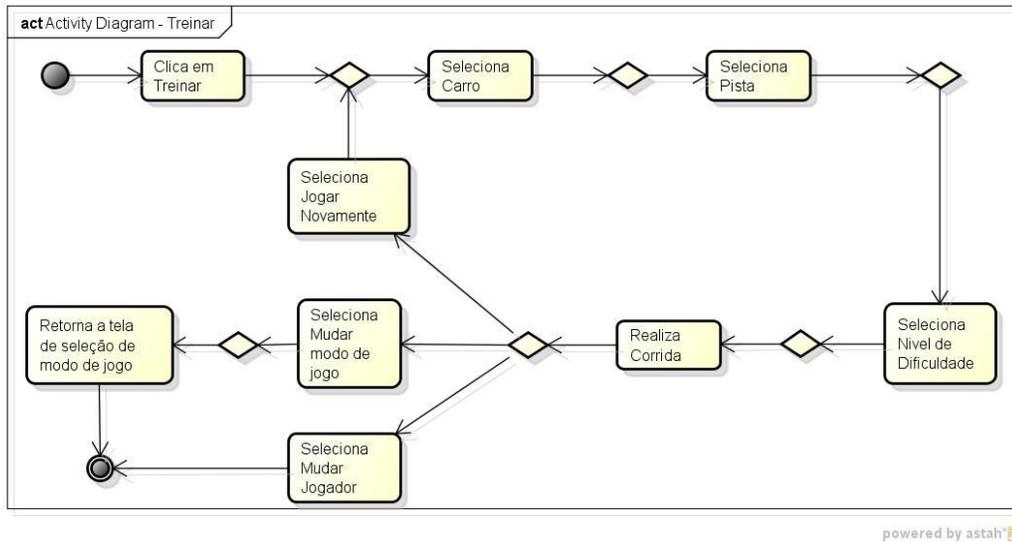


Figura 10 – Diagrama de Atividades do caso de uso Treinar

Realizar Teste

Na figura 11 é mostrado o diagrama de atividades do caso de uso Realizar Teste.

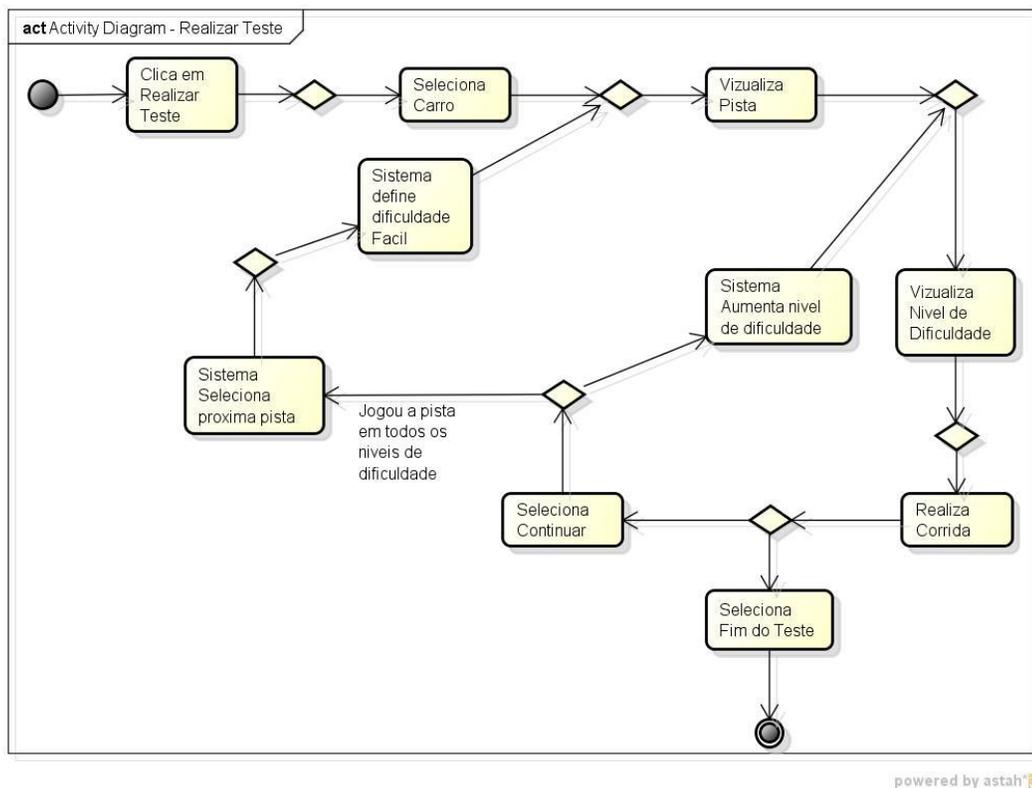


Figura 11 – Diagrama de Atividades do caso de uso Realizar Teste

4.1.3 Diagrama de Classes

O diagrama de classes representa a estrutura do sistema, recorrendo ao conceito de classe e suas relações. O modelo de classes resulta de um processo de abstração no qual são identificados os objetos relevantes do sistema. Um objeto é uma ocorrência que tem interesse para o sistema em estudo e que se pretende descrever no seu ambiente, contendo identidade e comportamento. O comportamento de um objeto define o modo como ele age e reage a estímulos externos e a identidade de um objeto é um atributo que o distingue de todos os demais, sendo preservada quando o seu estado muda. Um objeto não é mais do que uma instância de classe. (BOOCH *et al.*, 2005)

Na figura 12 é apresentado o diagrama de classes do aplicativo Speed Car.

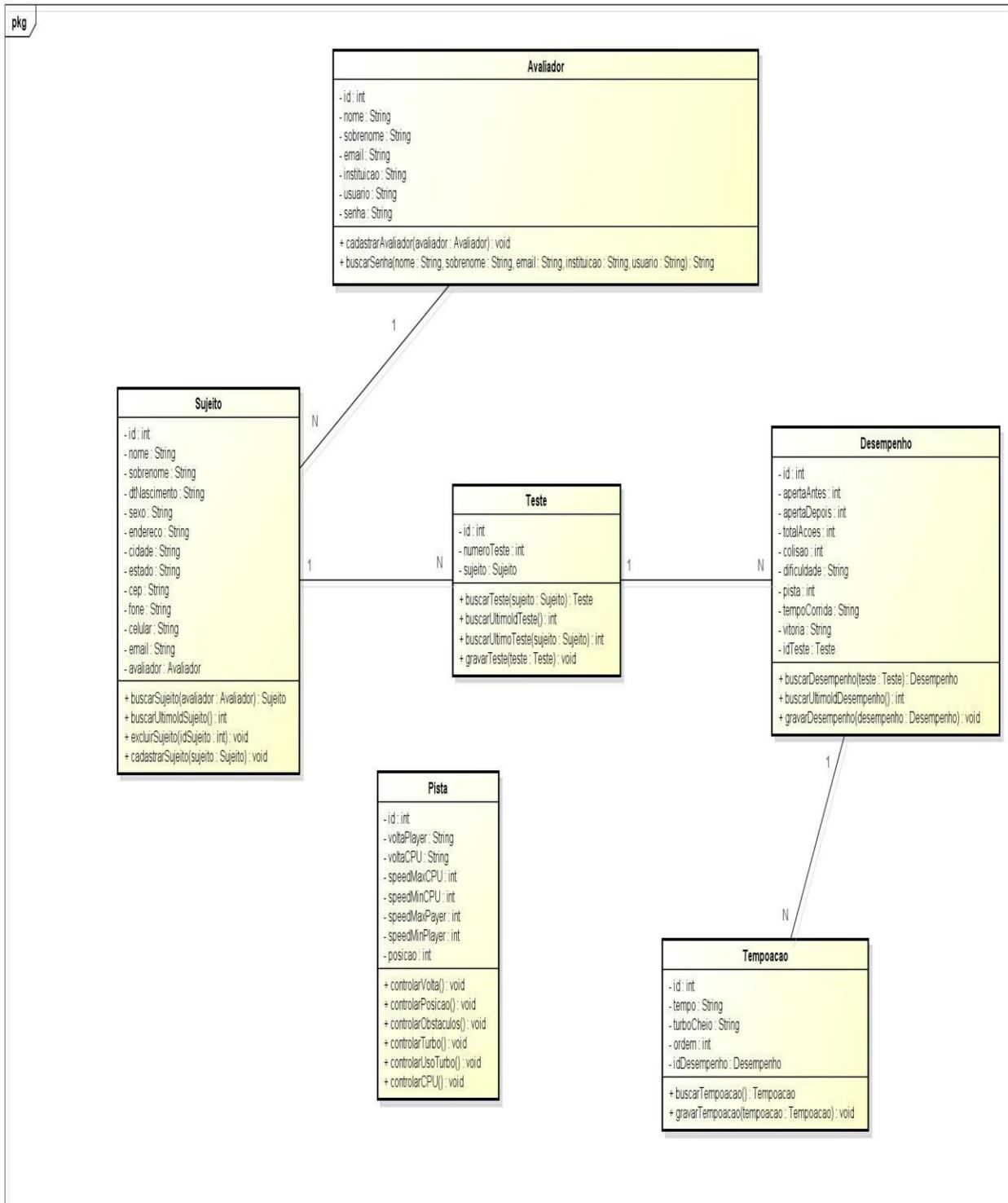


Figura 12 – Diagrama de Classes do Speed Car

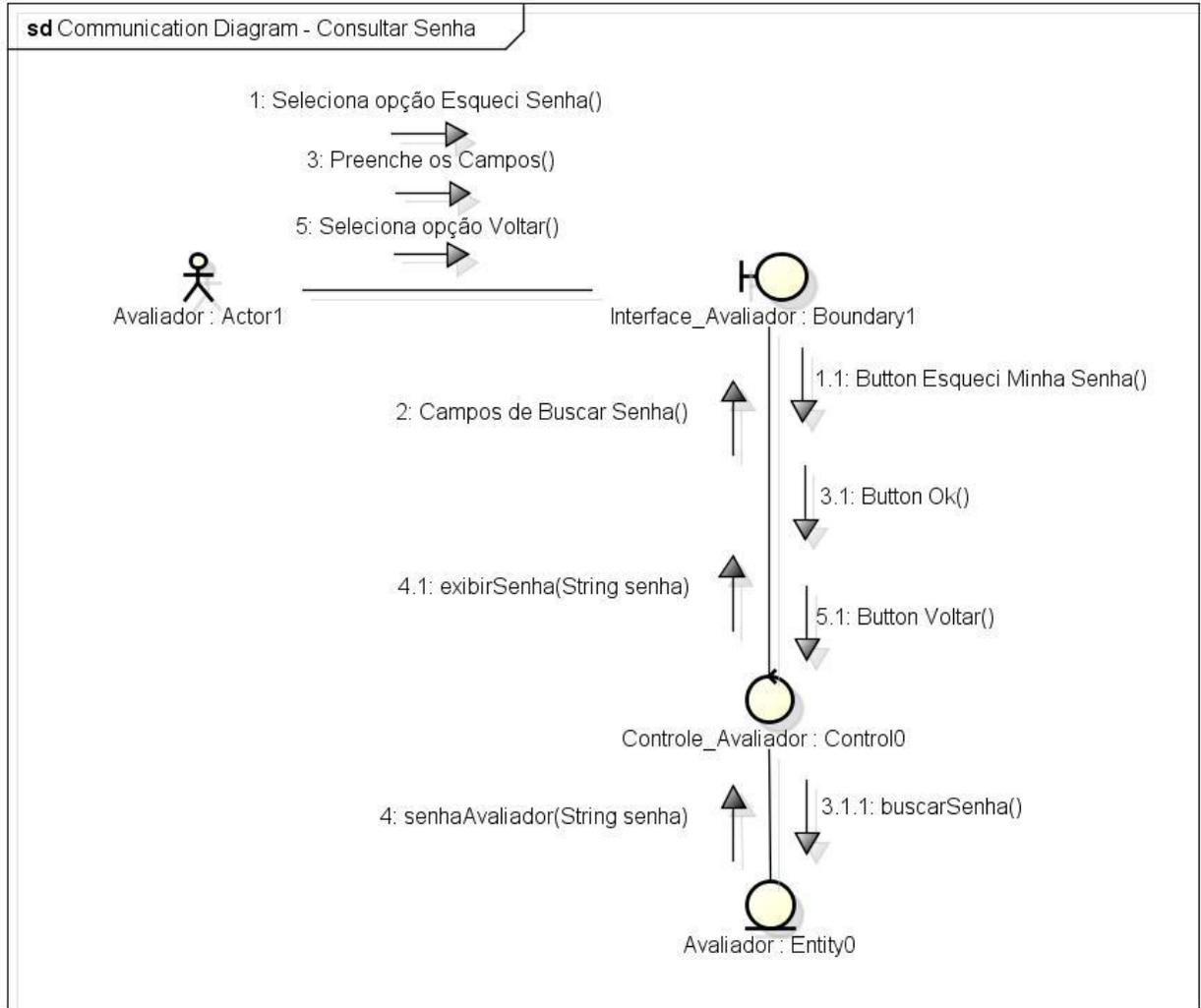
4.1.4 Diagramas de Comunicação

Segundo Booch *et. al.* (2005) o diagrama de comunicação exibe de maneira semelhante ao diagrama de sequência, a colaboração dinâmica entre os objetos. Ele é desenhado como um diagrama de objeto, no qual os diversos objetos são mostrados juntamente com seus relacionamentos.

A seguir são apresentados os diagramas de comunicação de cada caso de uso, auxiliando para um melhor entendimento do funcionamento de cada caso de uso.

Consultar Senha

Na figura 13 é mostrado o diagrama de comunicação do caso de uso Consultar Senha.

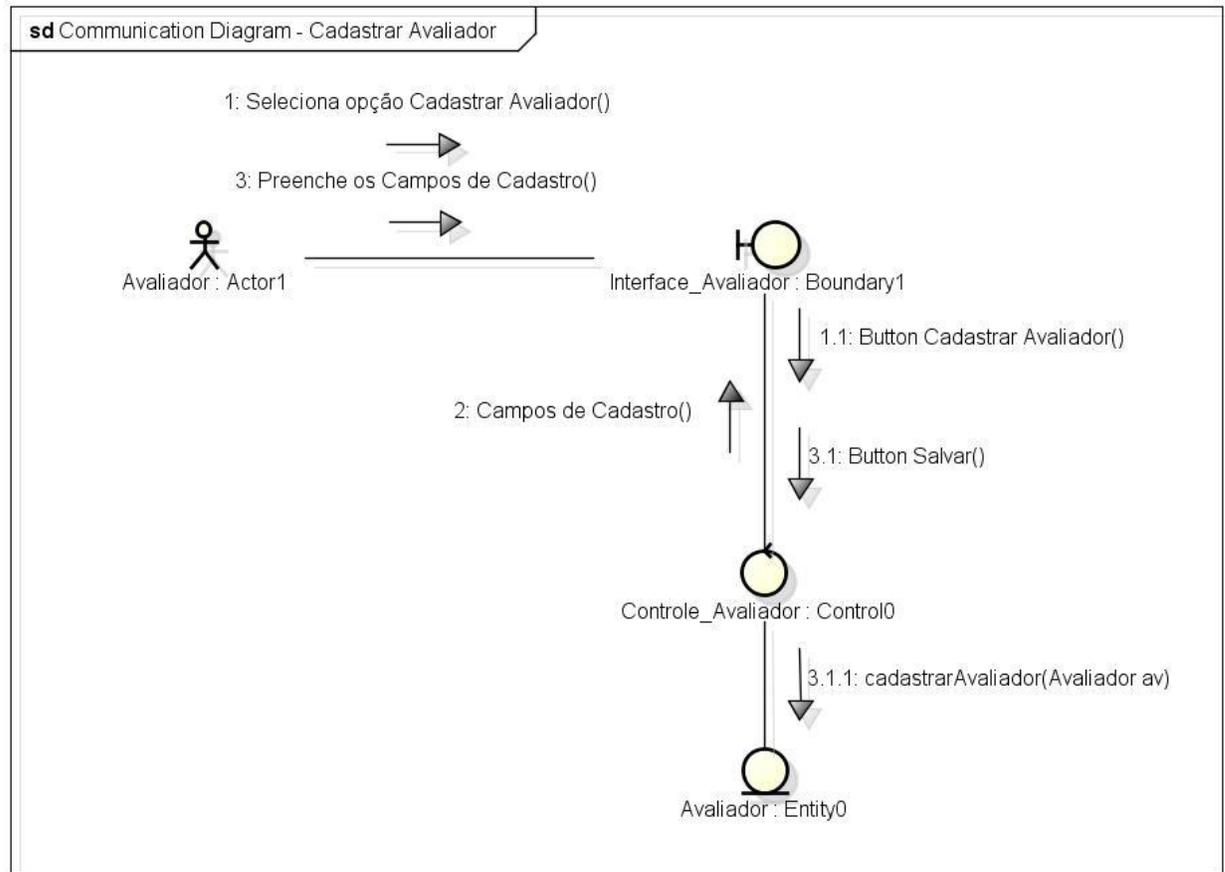


powered by astah

Figura 13 – Diagrama de Comunicação do caso de uso Consultar Senha

Cadastrar Avaliador

Na figura 14 é apresentado o diagrama de comunicação do caso de uso Cadastrar Avaliador.

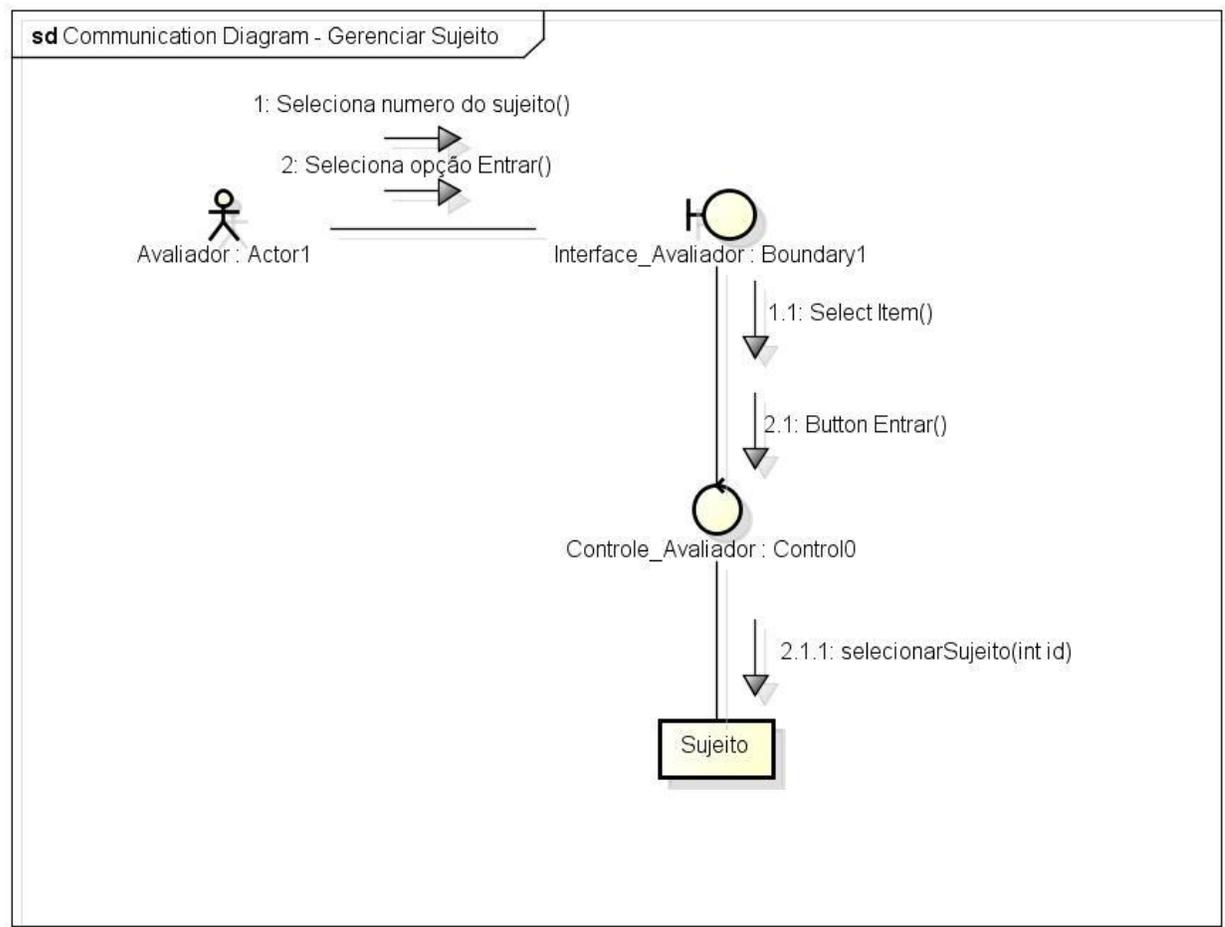


powered by astah®

Figura 14 – Diagrama de Comunicação do caso de uso Cadastrar Avaliador

Gerenciar Sujeito

Na figura 15 é apresentado o diagrama de comunicação do caso de uso Gerenciar Sujeito.

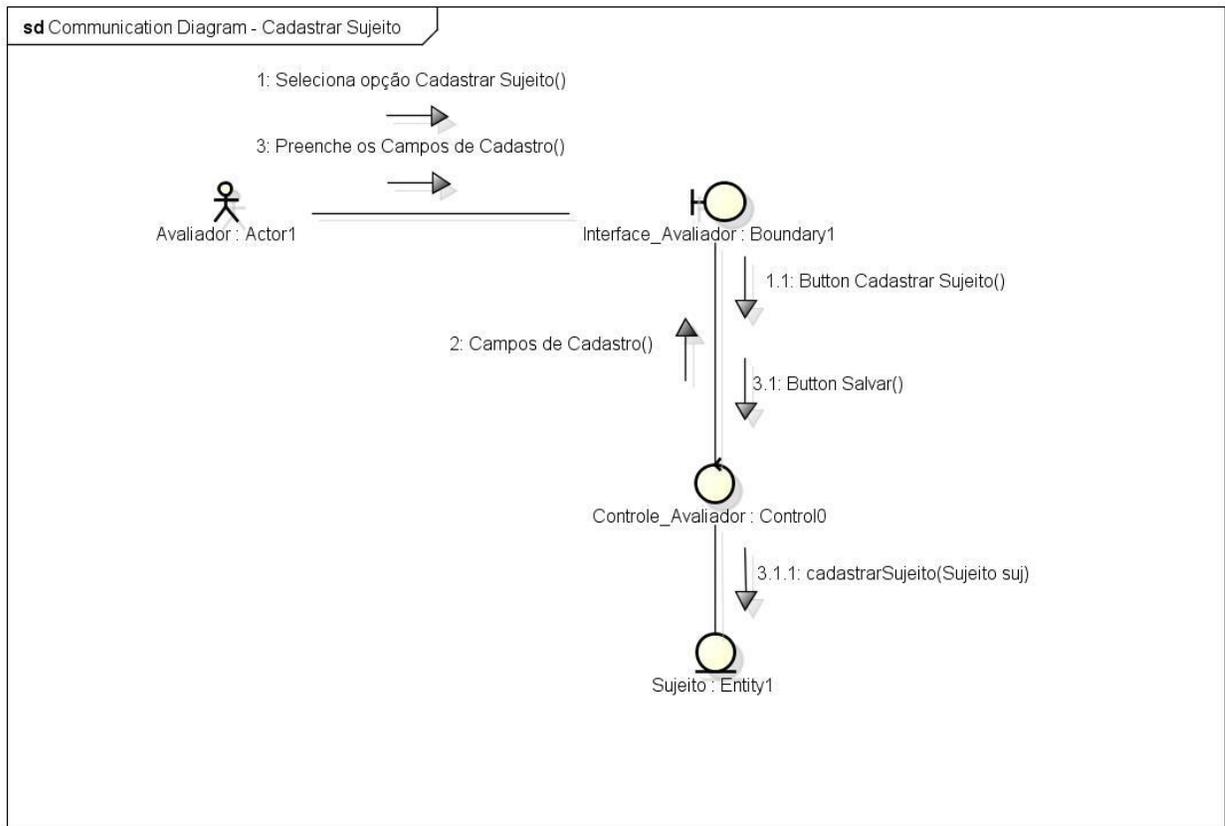


powered by astah

Figura 15 – Diagrama de Comunicação do caso de uso Gerenciar Sujeito

Cadastrar Sujeito

Na figura 16 é apresentado o diagrama de comunicação do caso de uso Cadastrar Sujeito.



powered by astah®

Figura 16 – Diagrama de Comunicação do caso de uso Cadastrar Sujeito

Gerenciar Relatórios

Na figura 17 é apresentado o diagrama de comunicação do caso de uso Gerenciar Relatórios.

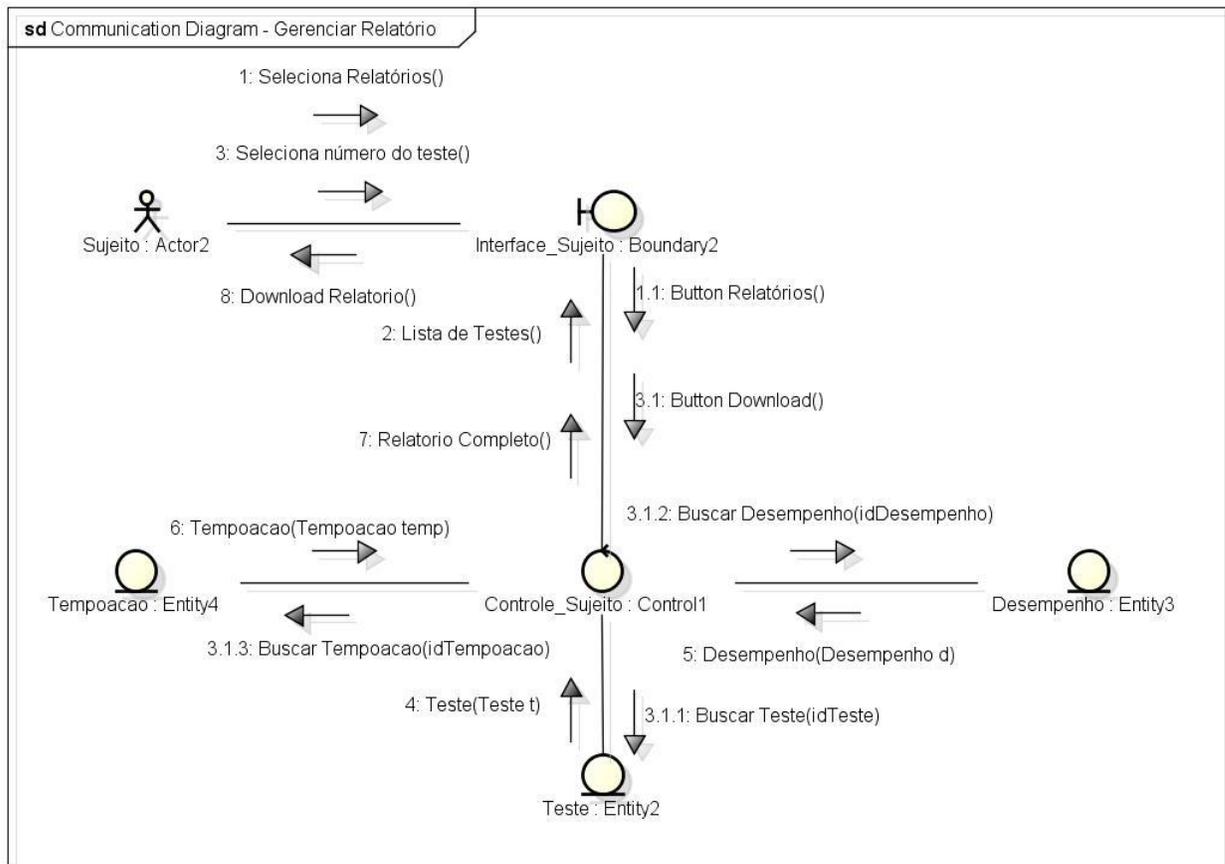
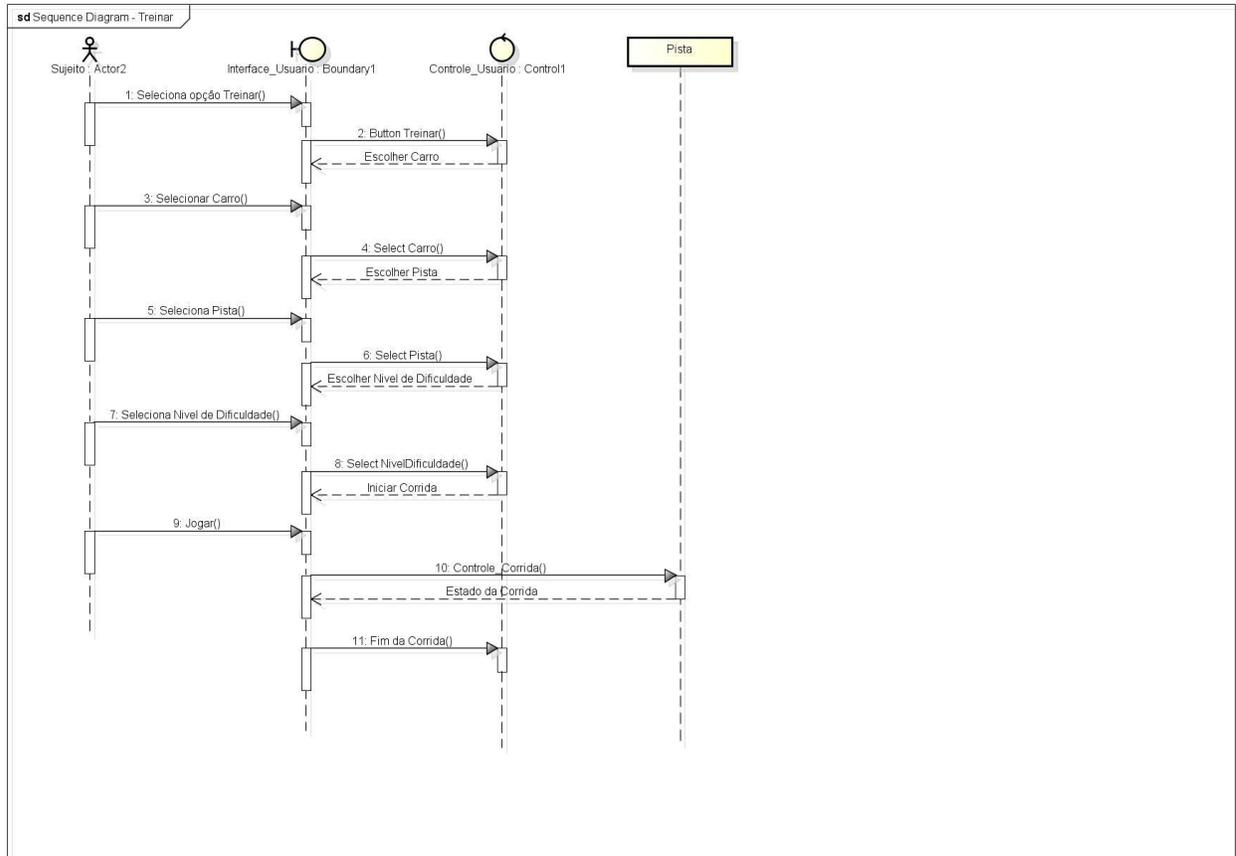


Figura 17 – Diagrama de Comunicação do caso de uso Gerenciar Relatórios

Treinar

Na figura 18 é apresentado o diagrama de comunicação do caso de uso Treinar.

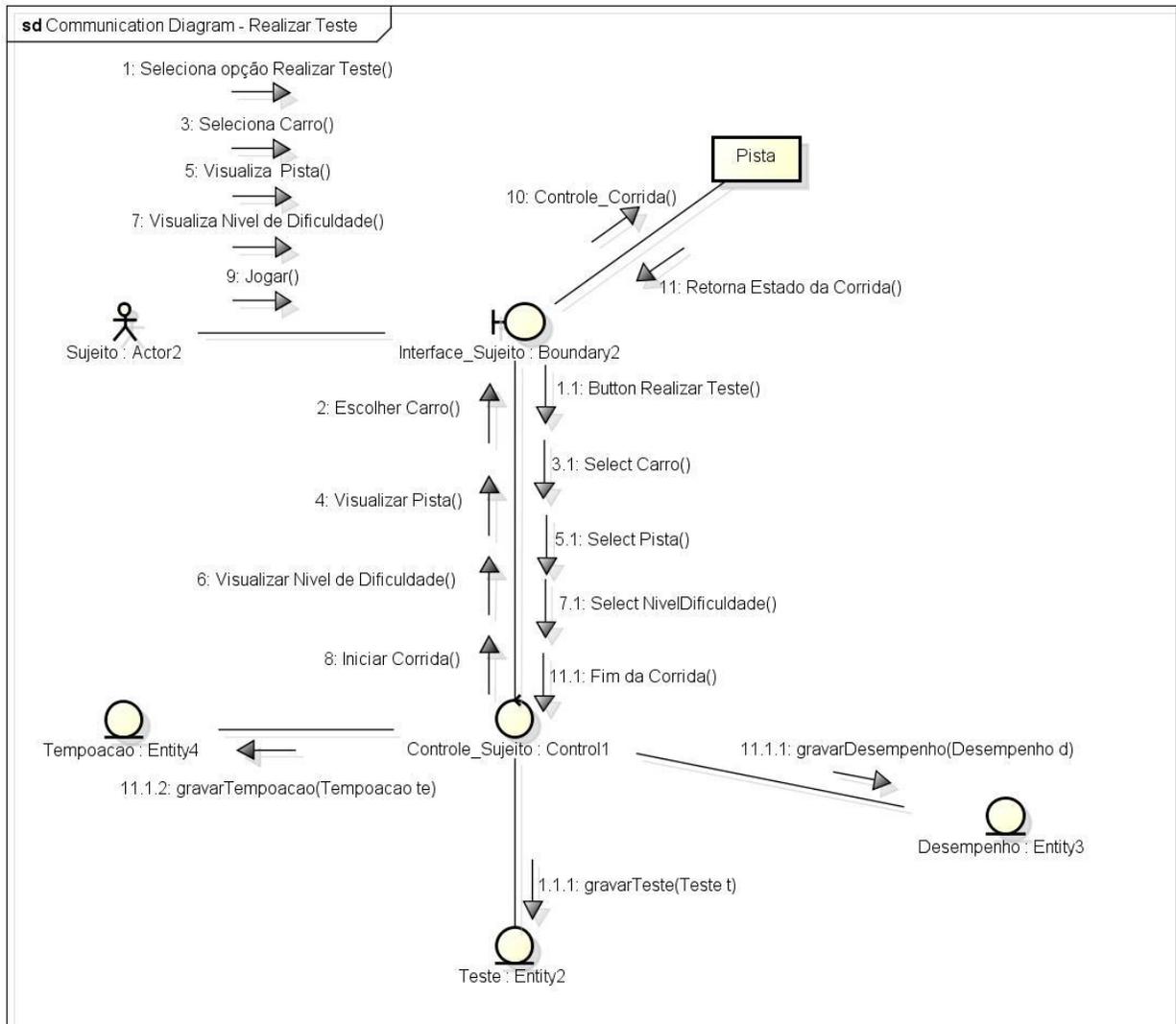


powered by astah

Figura 18 – Diagrama de Comunicação do caso de uso Treinar

Realizar Teste

Na figura 19 é apresentado o diagrama de comunicação do caso de uso Realizar Teste.



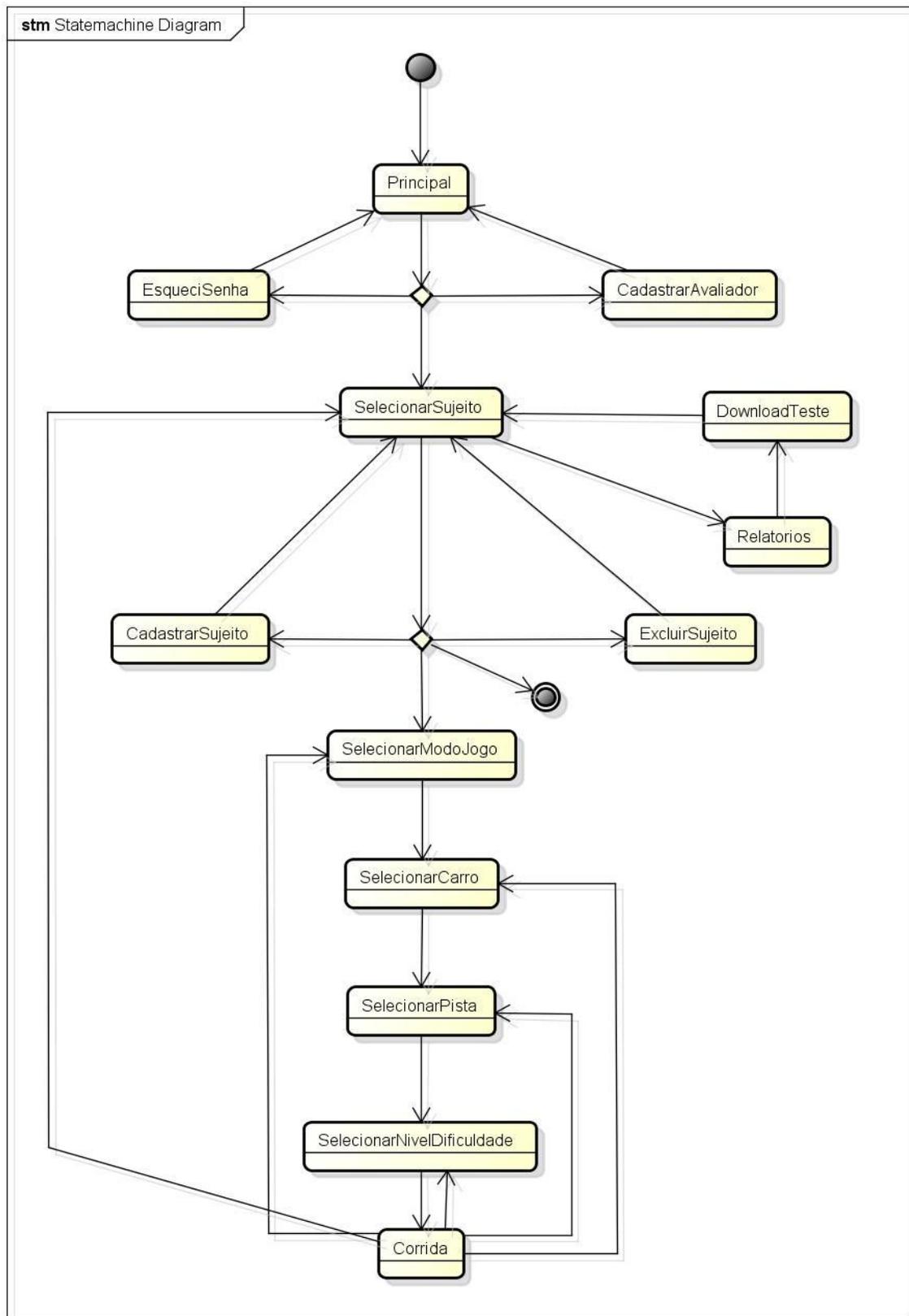
powered by astah

Figura 19 – Diagrama de Comunicação do caso de uso Realizar Teste

4.1.5 Diagrama de Máquina de Estado

O diagrama de máquina de estado procura acompanhar as mudanças sofridas nos estados de uma instância de uma determinada classe. Ele procura demonstrar o comportamento de um elemento por meio de transições de estado. (FOWLER, 2005)

Na figura 20 é apresentado o diagrama de máquina de estado do aplicativo Speed Car, auxiliando o entendimento dos estados dos objetos dentro do aplicativo.



powered by astah

Figura 20 – Diagrama de Máquina de Estado do Speed Car

4.1.6 Diagramas de Sequência

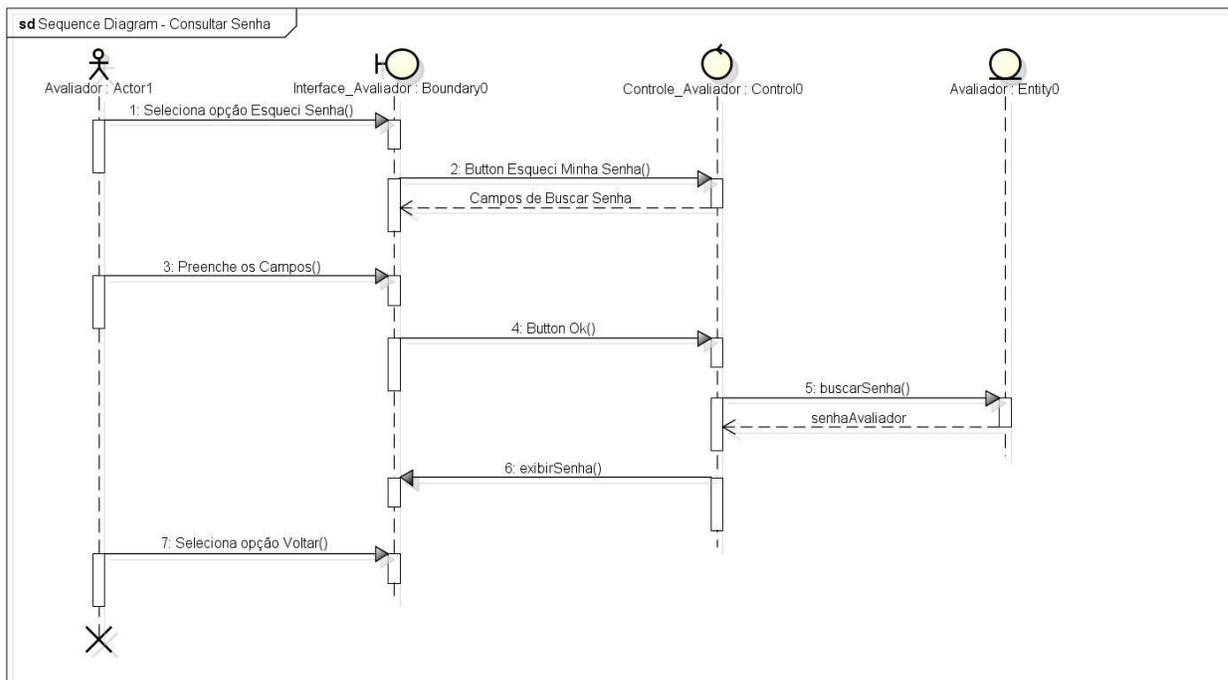
De acordo com Fowler (2005) o diagrama de sequência é utilizado para representar um cenário para um determinado caso de uso, ele mostra os eventos que partem do ator e chegam ao sistema. Para cada evento recebido, o sistema irá executar uma operação em resposta.

Segundo Booch *et al.* (2005), o diagrama de sequência é um diagrama de objetos que mostra o envio de mensagens entre eles que descrevem ao longo de uma linha de tempo a sequência de comunicação entre objetos. O decorrer do tempo é visualizado observando-se o diagrama no sentido vertical de cima para baixo, sendo as mensagens enviadas por cada objeto simbolizadas por setas entre os objetos que se relacionam.

A seguir são apresentados os diagramas de sequência de cada caso de uso, auxiliando o entendimento da sequência das informações enviadas de entre o ator e o sistema.

Consultar Senha

Na figura 21 é exibido o diagrama de sequência do caso de uso Consultar Senha.

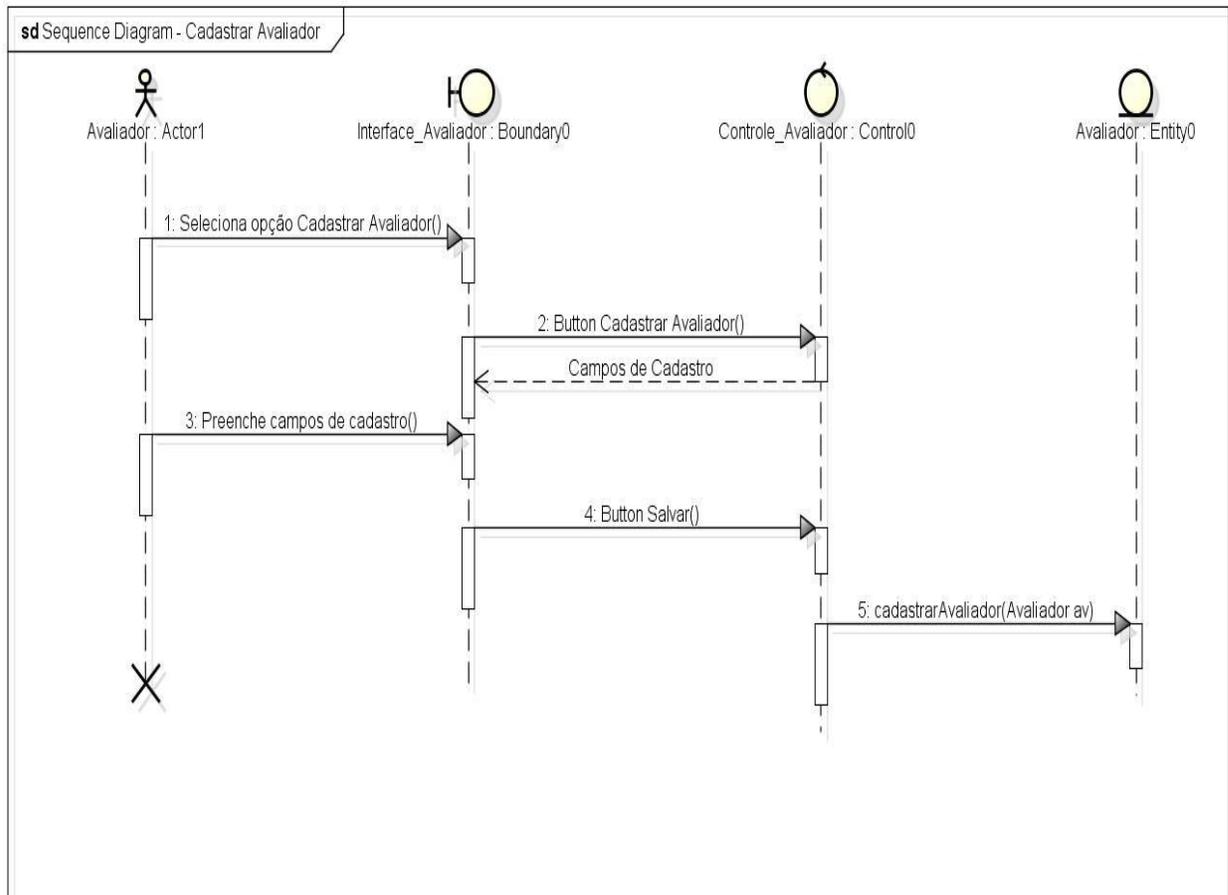


powered by astah

Figura 21 – Diagrama de Sequência do caso de uso Consultar Senha

Cadastrar Avaliador

Na figura 22 é apresentado o diagrama de seqüência do caso de uso Cadastrar Avaliador.

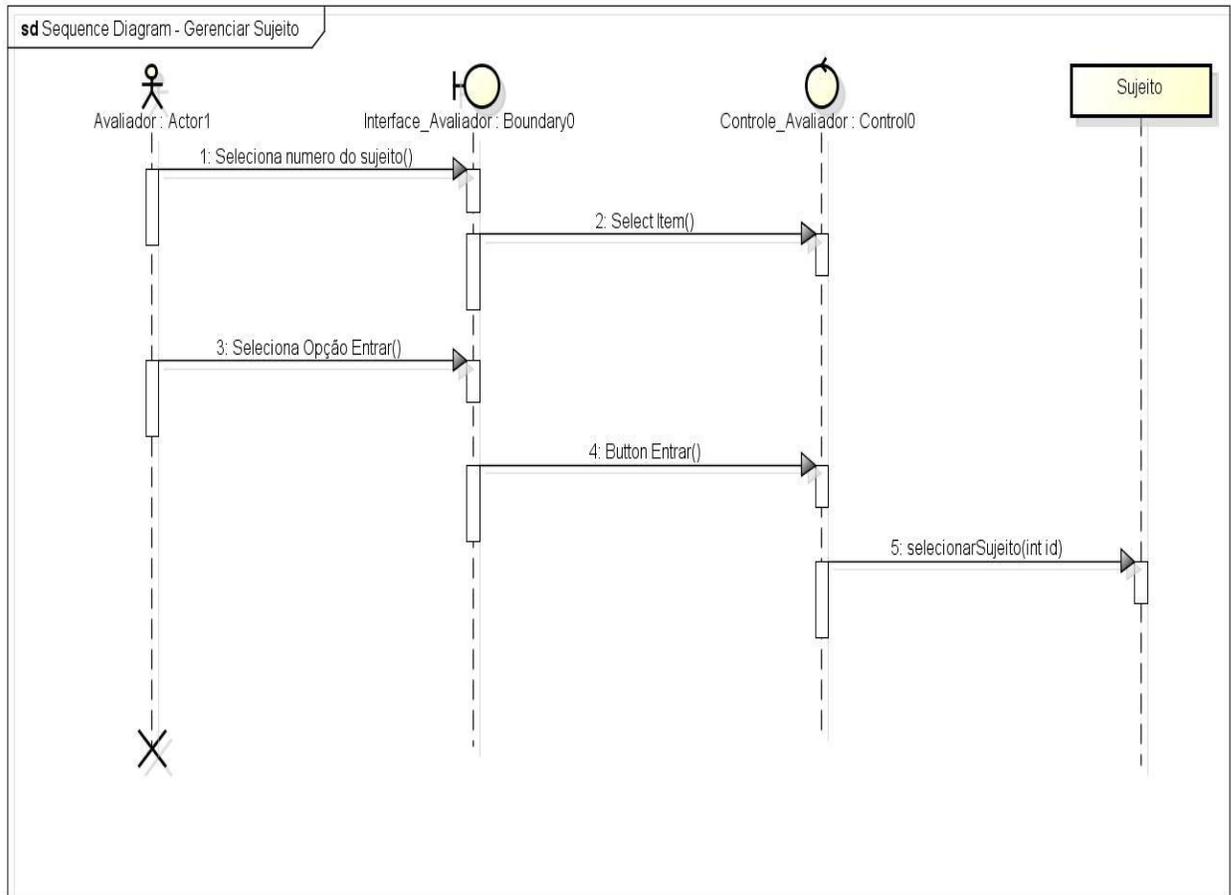


powered by astah

Figura 22 – Diagrama de Sequência do caso de uso Cadastrar Avaliador

Gerenciar Sujeito

Na figura 23 é mostrado o diagrama de sequência do caso de uso Gerenciar Sujeito.

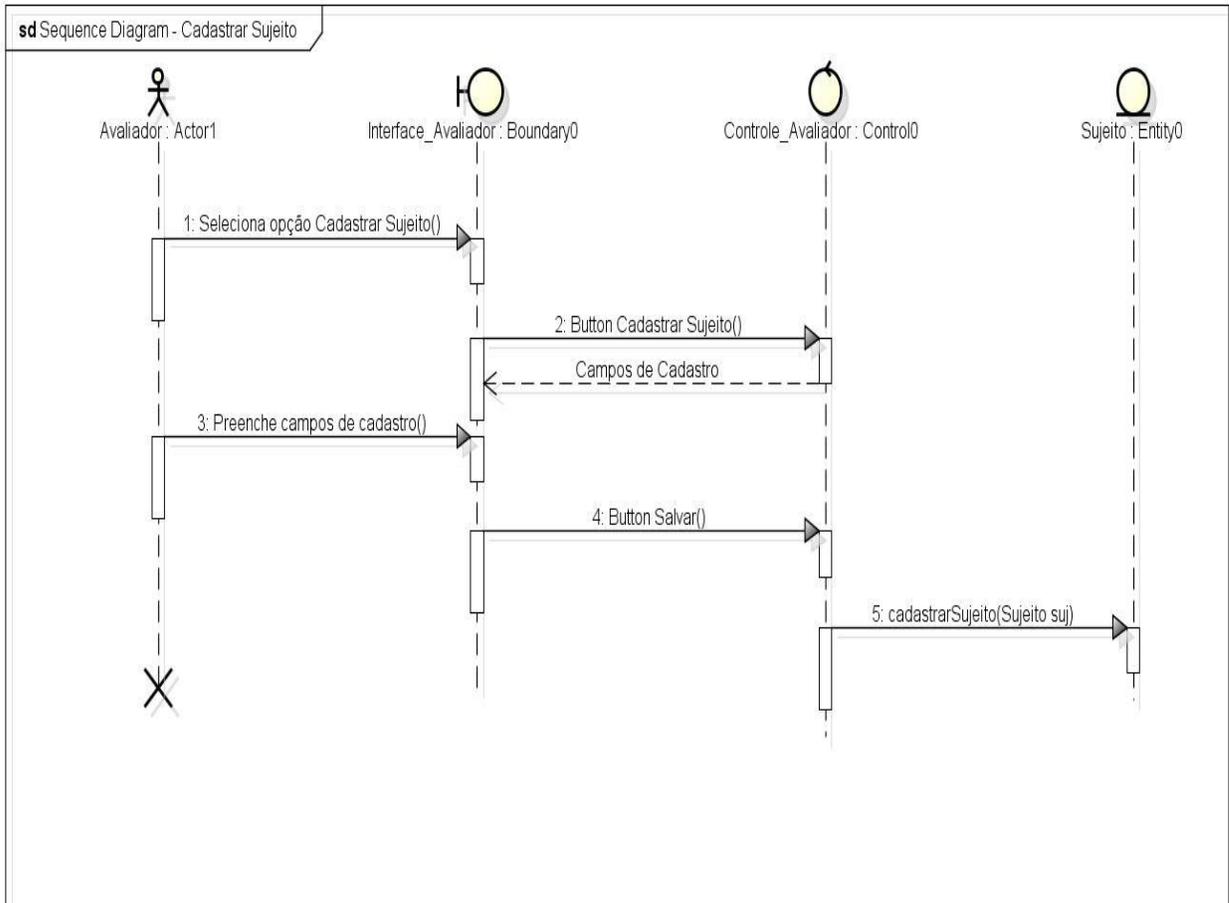


powered by astah

Figura 23 – Diagrama de Sequência do caso de uso Gerenciar Sujeito

Cadastrar Sujeito

Na figura 24 é apresentado o diagrama de seqüência do caso de uso Cadastrar Sujeito.



powered by astah

Figura 24 – Diagrama de Sequência do caso de uso Cadastrar Sujeito

Gerenciar Relatórios

Na figura 25 é exibido o diagrama de sequência do caso de uso Gerenciar Relatórios.

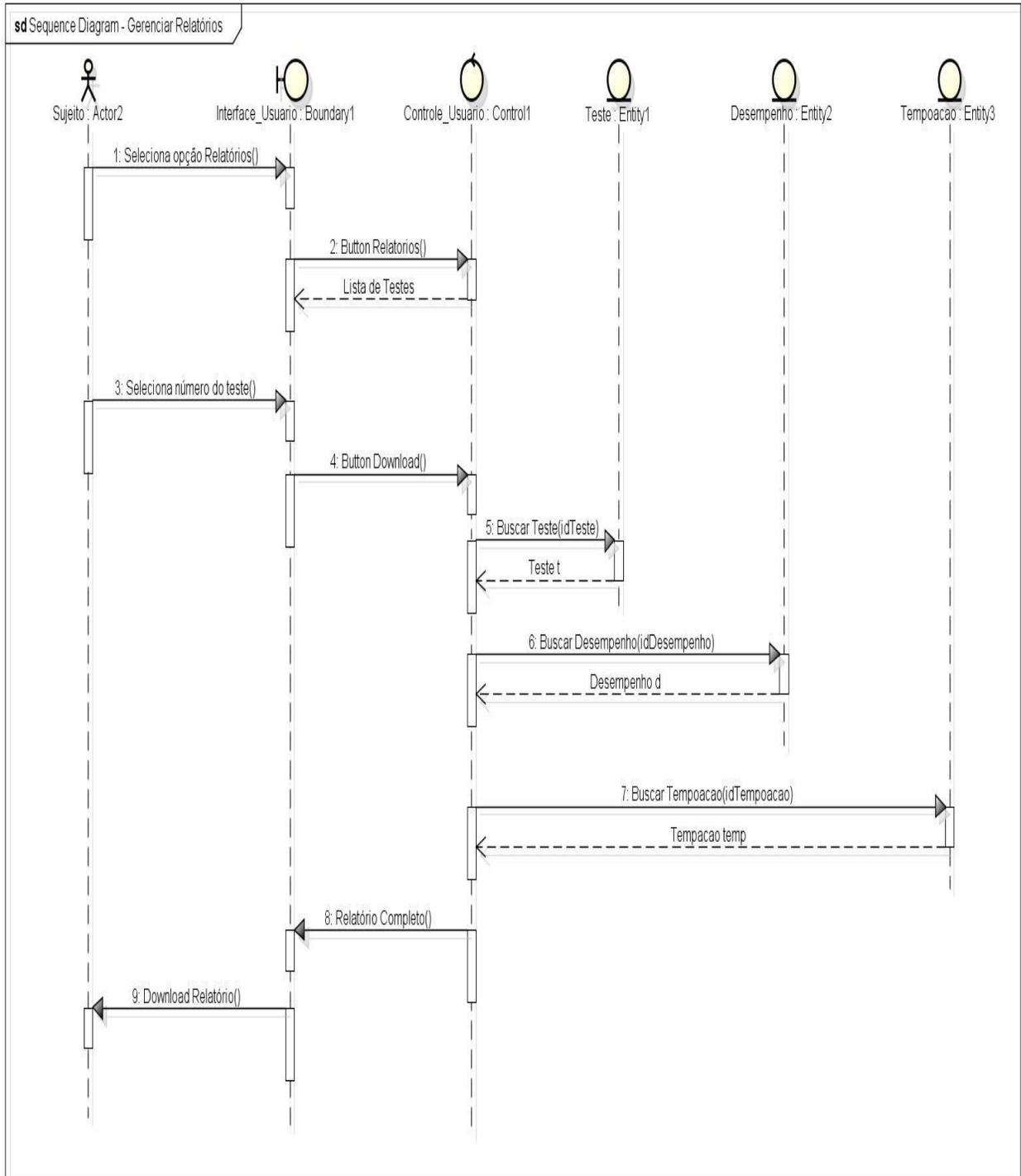
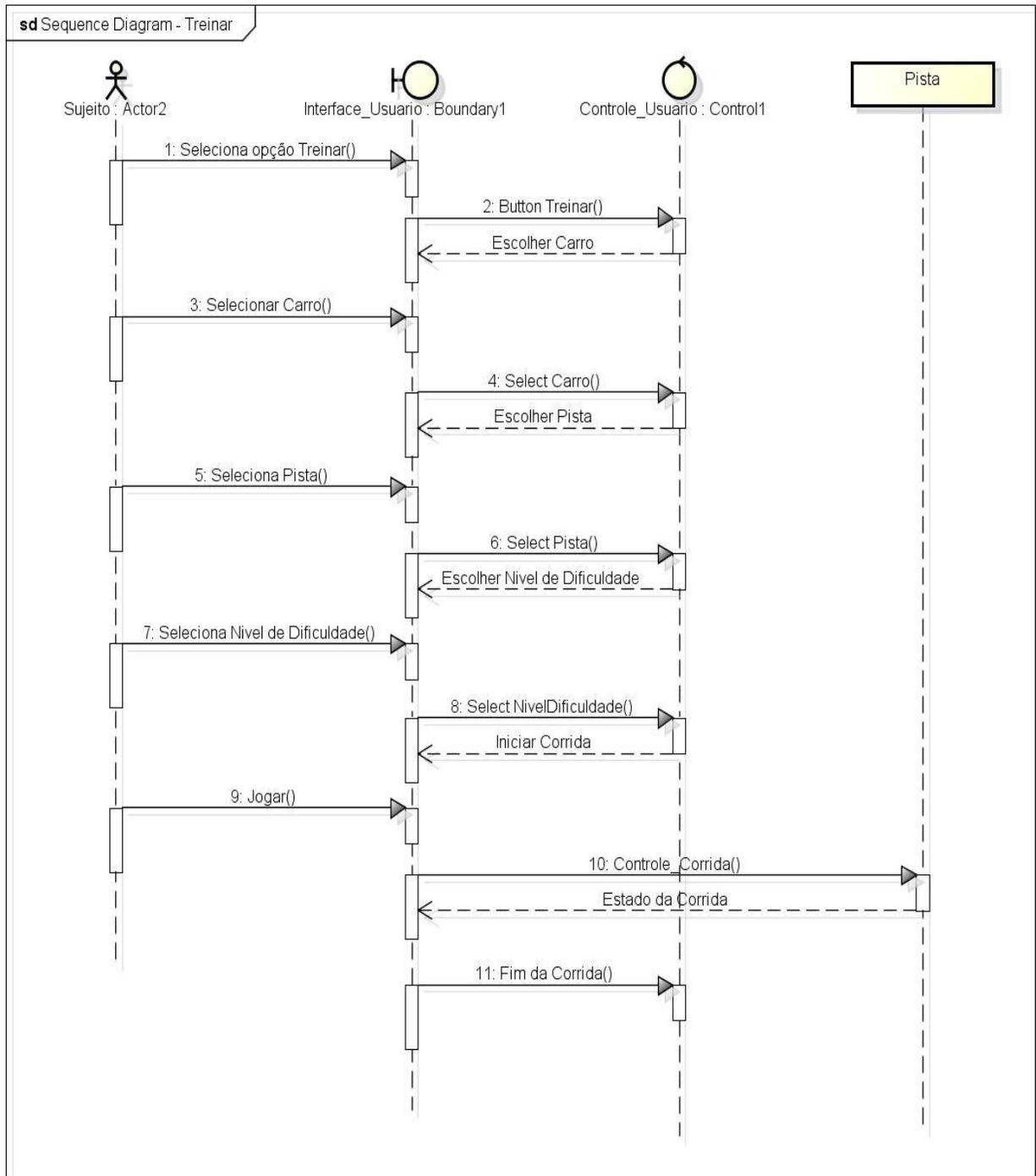


Figura 25 – Diagrama de Sequência do caso de uso Gerenciar Relatórios

Treinar

Na figura 26 é mostrado o diagrama de seqüência do caso de uso Treinar.



powered by astah

Figura 26 – Diagrama de Sequência do caso de uso Treinar

Realizar Teste

Na figura 27 é apresentado o diagrama de seqüência do caso de uso Realizar Teste.

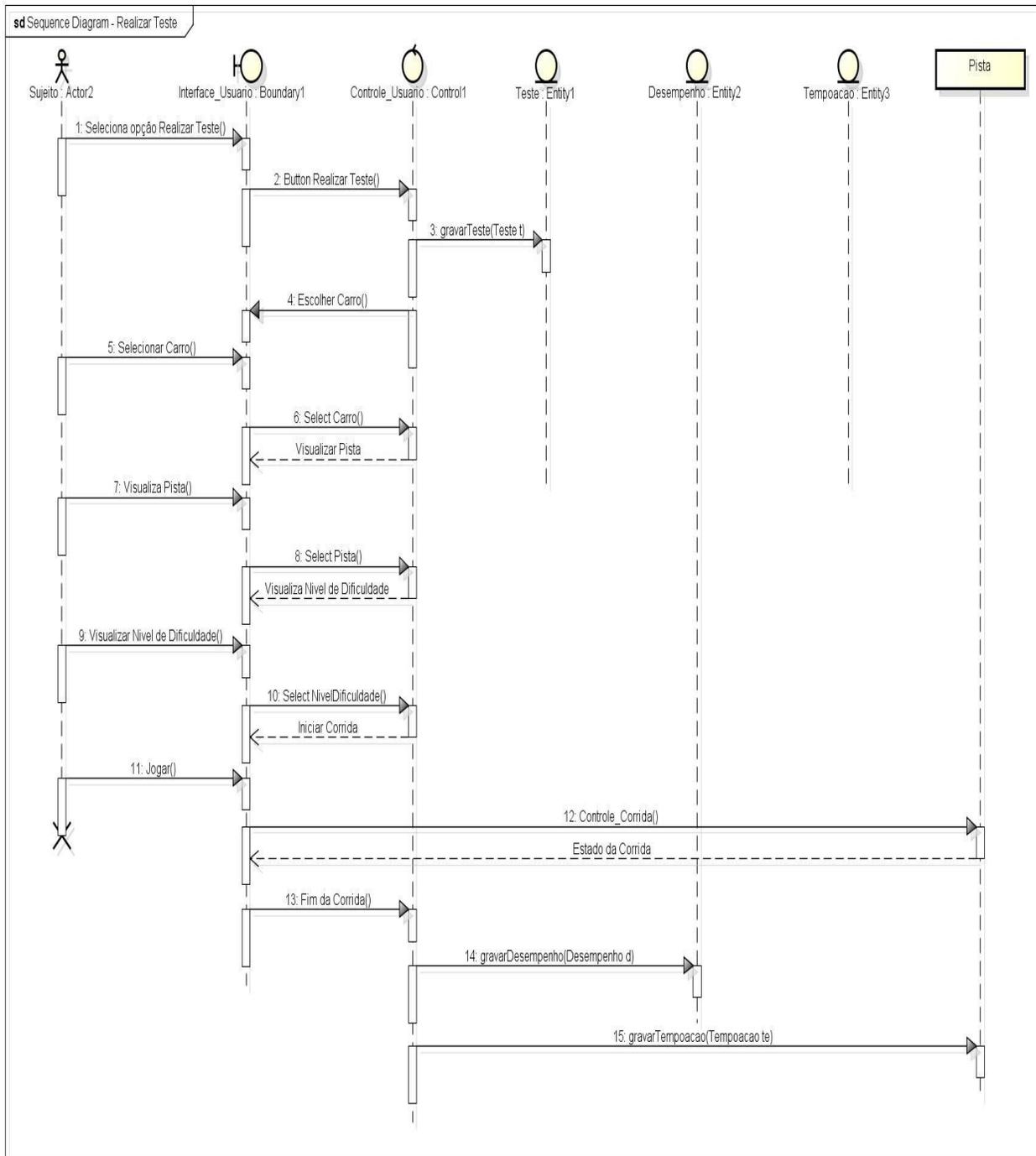


Figura 27 – Diagrama de Sequência do caso de uso Realizar Teste

4.2 Desenvolvimento do Aplicativo

Na fase de desenvolvimento, foi utilizado o motor de jogo *Contract 2* para a construção do aplicativo, o sistema gerenciador de banco de dados (SGBD) MySQL e a linguagem PHP para a comunicação entre o aplicativo e o banco de dados.

Na construção do banco de dados para o armazenamento das informações extraídas do aplicativo, foi criado um diagrama de entidade e relacionamento (DER) do aplicativo.

De acordo com Date (2003), o diagrama de entidade e relacionamento é um modelo abstrato que tem como objetivo descrever, de maneira conceitual, os dados a serem utilizados em um sistema de informações ou que pertencem a um domínio.

Na figura 28 é apresentado o diagrama de entidade e relacionamento do aplicativo Speed Car.

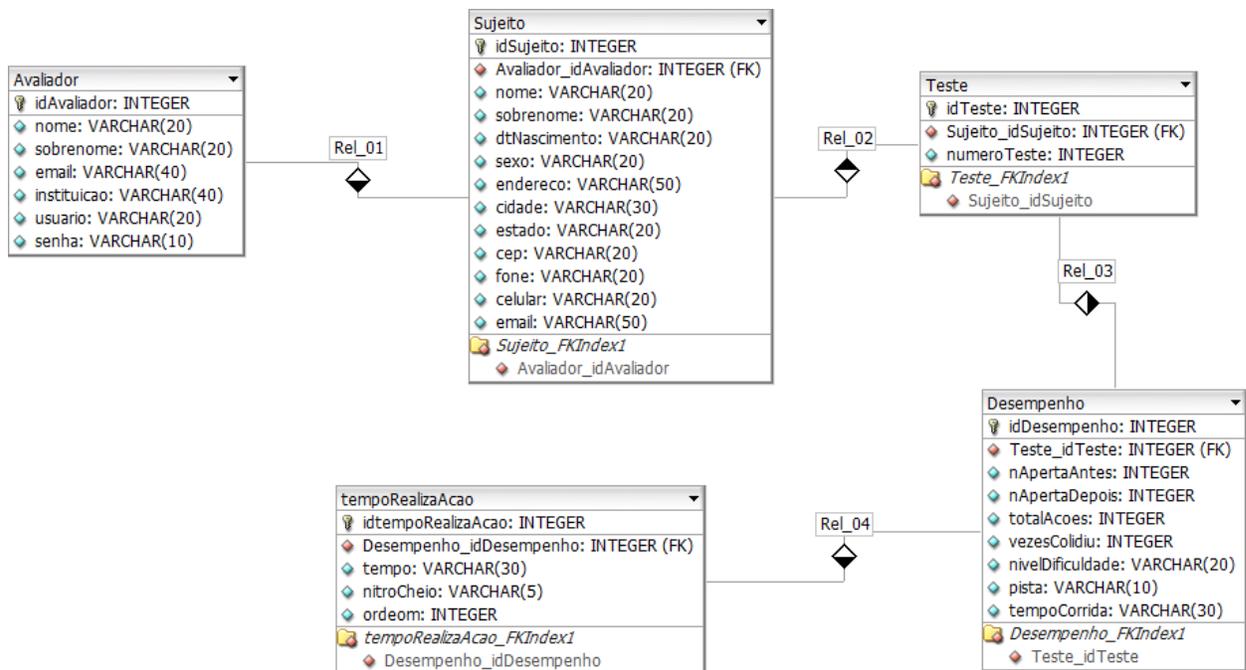


Figura 28 – Diagrama de Entidade e Relacionamento do Banco de Dados

No diagrama de entidade e relacionamento da figura 28, tem-se como está estruturado o banco de dados, sendo que este possui um total de cinco tabelas, no qual um avaliador pode ter vários sujeitos, um sujeito pode realizar vários testes, um teste é composto por várias corridas, representada pela tabela desempenho, e dentro de cada corrida pode ter vários tempo de demora para o sujeito realizar o estímulo.

O aplicativo é recomendado para avaliadores utilizar com seus pacientes. Para utilizá-lo, o avaliador deverá se cadastrar caso não tiver o cadastro. Em seguida ele escolhe o sujeito que irá realizar o teste, se caso o sujeito não estiver cadastrado, o avaliador cadastra o sujeito. Nesta mesma tela, o avaliador poderá excluir um sujeito e fazer o download do relatório de testes já realizados pelos sujeitos. Após o avaliador selecionar o sujeito, é o sujeito quem utilizará o aplicativo, escolhendo o modo de jogo, sendo o modo treino para se familiarizar com o aplicativo e o modo realizar teste para iniciar o teste. Após selecionar uma das opções o sujeito escolhe o carro que deseja utilizar. Se ele escolheu a opção treino, em seguida ele irá selecionar a pista e o nível de dificuldade que deseja correr, porém, se ele escolheu a opção teste, ele irá somente visualizar a pista e o nível de dificuldade que irá correr antes de realizar uma corrida, sendo que o teste completo é composto por nove corridas de dez voltas, três em cada pista, sendo cada uma em um nível de dificuldade. Após a realização do teste, retorna a tela de seleção de sujeito, e o avaliador poderá fazer o download do relatório do teste que o sujeito realizou.

4.3 Execução do Aplicativo

Na figura 29 é apresentada a tela inicial do aplicativo, nela o avaliador poderá logar-se, realizar o seu cadastro, caso ainda não tenha, ou recuperar sua senha se estiver esquecido, a mensagem de usuário ou senha inválido e o botão de esqueci minha senha só aparecem se caso o avaliador tentar se logar e o usuário e a senha não conferirem com as cadastradas.



Figura 29 – Tela Inicial

Ao clicar no botão cadastrar novo usuário, o sistema vai para a tela de cadastro de avaliador no qual serão exibidos os campos para realizar o cadastro de um novo avaliador, como é apresentado na figura 30.

Figura 30 – Tela de cadastro de avaliador

Na tela inicial do aplicativo, se o avaliador clicar no botão esqueci minha senha, serão exibidos alguns campos para que o avaliador preencha com seus dados, em

seguida o sistema verifica se os dados estão corretos, se estiverem ele exibe a senha do avaliador na tela, como é apresentado na figura 31.



A imagem mostra uma interface de usuário para a recuperação de senha. O fundo é verde escuro com uma textura de grama. No topo, o título "Preencha os Campos:" está em amarelo. Abaixo dele, há cinco campos de entrada brancos com rótulos em cinza: "Nome:" (contendo "Renan"), "Sobrenome:" (contendo "Leon"), "E-mail:" (contendo "renanleon@uenp.edu.br"), "Instituição:" (contendo "UENP") e "Usuario:" (contendo "renan"). À direita dos campos "E-mail:" e "Usuario:" há um botão "OK" cinza. Abaixo dos campos, o texto "Sua senha é: leon" está em amarelo. No centro da tela, há um botão "Voltar" cinza.

Figura 31 – Tela de recuperação de senha

Se caso o avaliador já estiver cadastrado e lembrar sua senha, ele simplesmente irá se logar na tela principal. Após entrar no sistema, aparecerá a tela de seleção de sujeito, nela o avaliador pode selecionar um sujeito para realizar o teste, cadastrar um novo sujeito, excluir um sujeito ou visualizar os relatórios de testes já realizados por um sujeito selecionado, como é mostrado na figura 32.



Olá Nome do Avaliador

Lista de Jogadores

Não há jogadores cadastrados

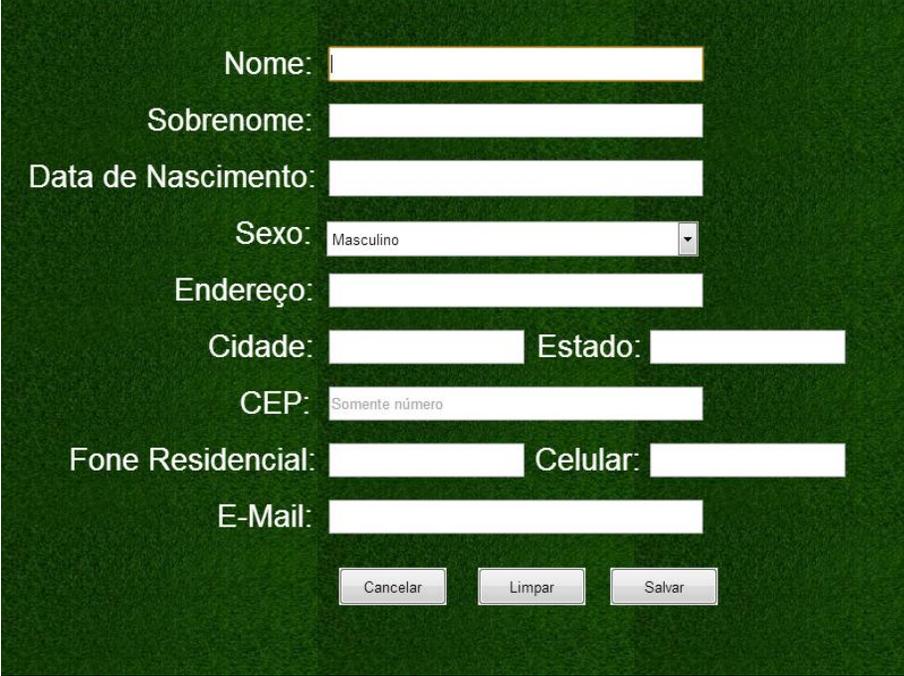
Selecione o número do jogador:

Excluir Cadastrar Jogador Entrar

Relatórios

Figura 32 – Tela de seleção de sujeito

Para excluir um sujeito, o avaliador selecionará o número do sujeito e clicar no botão excluir. Se o avaliador quiser cadastrar um novo sujeito, ele irá clicar no botão cadastrar jogador, que irá exibir uma tela com os campos de cadastro de sujeito, como é mostrado na figura 33.



Nome:

Sobrenome:

Data de Nascimento:

Sexo:

Endereço:

Cidade: Estado:

CEP:

Fone Residencial: Celular:

E-Mail:

Cancelar Limpar Salvar

Figura 33 – Tela de cadastro de sujeito

Para visualizar os relatórios dos testes já realizados por algum sujeito, o avaliador deve selecionar um sujeito na tela de seleção de sujeito e em seguida clicar no botão relatórios. O sistema irá exibir uma lista dos testes realizados por aquele sujeito selecionado e o avaliador poderá selecionar um e fazer o *download* do relatório clicando no botão *download*, como é apresentado na figura 34.



Figura 34 – Tela de relatórios

Cada teste realizado por um sujeito gera um relatório de desempenho para ser monitorado pelo avaliador, neste relatório contém os dados do avaliador, os dados pessoais do sujeito, o teste que ele realizou, bem como as informações extraídas do teste, contendo informações gerais do teste e informações de cada corrida, como é apresentado na figura 35.

Speed Car

Avaliador

Nome: Renan Leon

E-mail: renanleon@uenp.edu.br

Instituição: UENP

Sujeito

Nome: Thielle Claro

Endereço: Rua Com. Luiz Meneghel, 1136

CEP: 86.360-000

Cidade: Bandeirantes

Estado: PR

Fone: (43) 3542-1111

Celular: (43) 9659-1111

E-mail: thielle@uenp.edu.br

Teste 2

14/11/2013 10:32 PM

Tempo Total do Teste: 1m 21s 0ms

Número de Vitórias: 6 de 9 corridas

Total de Estímulos: 54

Total de Colisões: 63

Estímulos com Turbo Cheio: 45

Estímulos Antecipados: 45

> Pista = 1 | Dificuldade = Fácil

Tempo Total da Corrida: 5 segundos

Vitória: sim

Total de Estímulos: 2

Total de Colisões: 3

Estímulos com Turbo Cheio: 1

Estímulos Antecipados: 1

Estímulos	Turbo Cheio	Tempo de Resposta
1	Não	2,045 segundos
2	sim	2,100 segundos

> Pista = 1 | Dificuldade = Médio

Tempo Total da Corrida: 6 segundos

Vitória: sim

Total de Estímulos: 3

Total de Colisões: 4

Estímulos com Turbo Cheio: 2

Estímulos Antecipados: 2

* Tempo para encher totalmente a barra de turbo = 8,1 segundos

Página 1/3

Figura 35 – Relatório de desempenho

Depois de selecionado um sujeito na tela de seleção de sujeito, quem irá utilizar o aplicativo é o sujeito. O aplicativo exibe uma tela de seleção de modo de jogo, sendo que o sujeito pode escolher o modo treino e iniciar teste, como é mostrado na figura 36.



Figura 36 – Tela de seleção de modo de jogo

A diferença entre o modo treino e iniciar teste é que no modo treino o sujeito pode escolher em qual pista ele quer correr e em qual nível de dificuldade, e no modo iniciar teste, o sujeito deve obrigatoriamente correr nas três pistas três vezes em cada, uma vez em cada nível de dificuldade. Selecionando uma das opções, o sistema exibe a tela de seleção de carro, onde o sujeito deverá escolher o carro que deseja utilizar, como é apresentado na figura 37.



Figura 37 – Tela de seleção de carro

Após o sujeito selecionar um dos carros apresentados na figura 37, o sistema exibe a tela de seleção de pista, se caso estiver no modo iniciar teste, o sujeito irá apenas visualizar a pista que irá realizar a corrida, como é apresentado na figura 38.

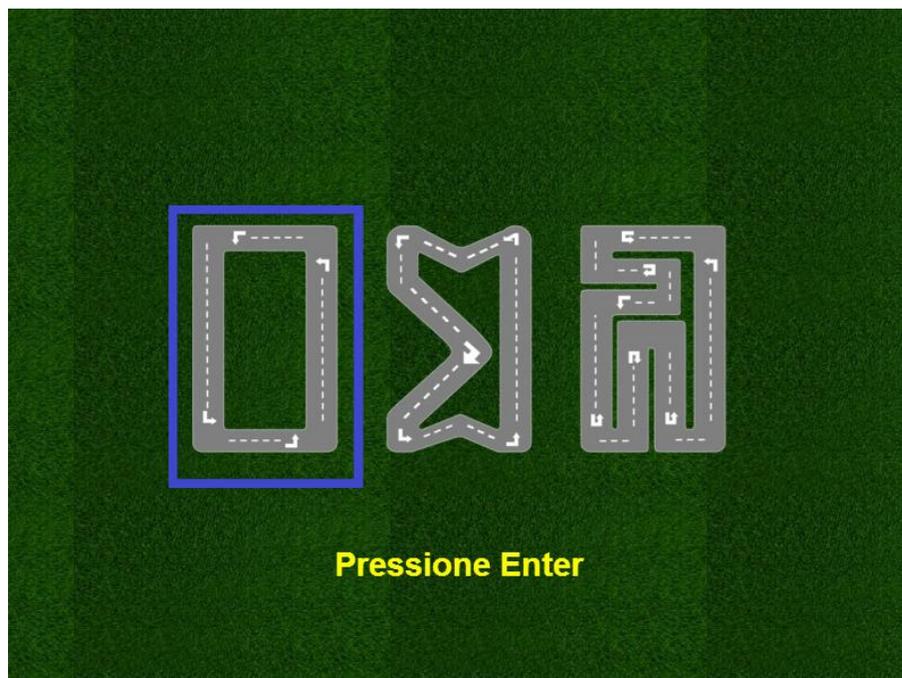


Figura 38 – Tela de seleção de pista

Com a pista selecionada, o sistema exibe a tela de seleção de nível de dificuldade, se caso estiver no modo iniciar teste, o sujeito irá apenas visualizar o nível de dificuldade que irá realizar a corrida, como é apresentado na figura 39.



Figura 39 – Tela de seleção de nível de dificuldade

Em seguida, o sistema vai para a tela de corrida, onde o sujeito irá realizar uma corrida de 10 voltas contra um outro carro guiado pelo sistema, como mostra a figura 40.



Figura 40 – Tela de corrida

Nesta tela, o veículo da esquerda é controlado pelo sujeito e o da direita é controlado pelo sistema. No canto superior direito está a barra de turbo, que está vazia antes do início da corrida. No canto superior esquerdo está a contagem de voltas e o total de voltas e no canto inferior direito a posição atual do sujeito na corrida.

Se estiver no modo teste, ao final de cada corrida são exibidas as características da próxima corrida, como a pista e o nível de dificuldade, até que realize todo o teste, porém, se estiver no modo treino, ao final da corrida o sujeito pode escolher se quer jogar novamente, voltando para a tela de seleção de carro, mudar o modo de jogo, voltando para a tela de seleção de modo de jogo ou mudar jogador, voltando para a tela de seleção de sujeito, onde o avaliador poderá escolher outro sujeito para utilizar o aplicativo.

4.4 Comunicação entre o aplicativo e o banco de dados

A linguagem PHP utilizada para realizar a comunicação entre o aplicativo e o banco de dados é uma linguagem Web, que trabalhar com funcionalidades GET e POST.

O Speed Car é um aplicativo Web que estará hospedado em um servidor web. Para o armazenamento das informações do teste, o aplicativo realiza um POST na url,

comunicando com um arquivo PHP criado e hospedado no mesmo servidor, que utiliza o GET para pegar essas informações fornecidas pelo aplicativo, com isso o PHP faz a conexão com o banco de dados e guarda as informações nele, como é apresentado na figura 41 a estrutura do arquivo PHP responsável por salvar as informações de cadastro do avaliador no banco de dados.

```

1 <?php
2
3 $db = "speedcar";//Your database name
4 $dbu = "root";//Your database username
5 $dbp = "";//Your database users' password
6 $host = "localhost";//MySQL server - usually localhost
7
8 $dblink = mysql_connect($host,$dbu,$dbp);
9 $seldb = mysql_select_db($db);
10
11 if(isset($_GET['nome']) && isset($_GET['sobrenome']) && isset($_GET['email']) && isset($_GET['instituicao']) && isset($_GET['usuario']) &&
12 isset($_GET['senha'])){
13
14     $nome = strip_tags(mysql_real_escape_string($_GET['nome']));
15     $sobrenome = strip_tags(mysql_real_escape_string($_GET['sobrenome']));
16     $email = strip_tags(mysql_real_escape_string($_GET['email']));
17     $instituicao = strip_tags(mysql_real_escape_string($_GET['instituicao']));
18     $usuario = strip_tags(mysql_real_escape_string($_GET['usuario']));
19     $senha = strip_tags(mysql_real_escape_string($_GET['senha']));
20     $sql = mysql_query("INSERT INTO `db`.`avaliador` (`id`,`nome`,`sobrenome`,`email`,`instituicao`,`usuario`,`senha`) VALUES
21 ('','$nome','$sobrenome','$email','$instituicao','$usuario','$senha');");
22
23     if($sql){
24         echo 'Your score was saved. Congrats!';
25     }else{
26         echo 'There was a problem saving your score. Please try again later.';
27     }
28 }else{
29     echo 'Your name or score wasnt passed in the request. Make sure you add ?name=NAME&
30     sobrenome=SOBRENOME&email=EMAIL&instituicao=INSTITUICAO&usuario=USUARIO&senha=SENHA to the tags.';
31 }
32 mysql_close($dblink);//Close off the MySQL connection to save resources.
33 ?>

```

Figura 41 – PHP Cadastrar Avaliador

Na figura 41, primeiro são guardadas as informações do banco de dados, na linha 8 o PHP inicia a conexão com o banco de dados, na linha 11 é verificado se todas as informações necessárias para o cadastro do avaliador foram passadas através dos métodos GET e POST, na linha 20 é inserido no banco de dados as informações que o aplicativo forneceu, e para finalizar, na linha 32 o PHP fecha a conexão com o banco de dados.

E para a recuperação de dados do banco de dados para o aplicativo, o aplicativo utiliza o GET, no qual ele envia para o PHP as informações que necessita, o PHP busca no banco de dados e retorna para o aplicativo as informações solicitadas.

Nesta seção foi apresentado a forma de desenvolvimento do aplicativo, assim como, foi mostrado todos os recursos necessários para o desenvolvimento do mesmo. As maiores dificuldades foram o entendimento da ferramenta *Construct 2*, por ser uma ferramenta estrangeira e algumas limitações trazidas, como a integração com o banco de dados ser exclusivamente por meio da linguagem de programação PHP, sendo necessário criar vários arquivos em PHP para que possam dar suporte ao aplicativo. Porém, apesar destas dificuldades, o aplicativo foi criado cumprindo assim os objetivos do trabalho. A seguir será apresentada a Conclusão do trabalho, no qual serão apresentadas as discussões finais, assim como, as sugestões de desenvolvimento de trabalhos futuros.

CONCLUSÃO

O objetivo deste trabalho foi propor e desenvolver um aplicativo para treino de inibição comportamental de pessoa com diagnósticos de Transtorno de Déficit de Atenção e Hiperatividade (TDAH). Foi criado um aplicativo baseado nos conceitos de Reforçamento Diferencial de Baixa Taxa de Resposta (*Differential Reinforcement of Low Response Rate – DRL*) no qual o avaliador se cadastra, em seguida cadastra seus pacientes e o paciente realiza o teste como um jogo de corrida. As principais informações do teste são armazenadas no banco de dados, sendo que o avaliador pode fazer o *download* dos relatórios de desempenho dos seus pacientes nos testes já realizados.

Durante o desenvolvimento, o entendimento da ferramenta *Construct 2* foi a maior restrição encontrada, sendo necessário dedicar um bom tempo em vídeos aula e tutoriais. O objetivo do aplicativo não era simplesmente o desenvolvimento de um aplicativo qualquer para treino de inibição comportamental, mas sim, um aplicativo bonito, que chame a atenção dos profissionais da área da saúde e de seus pacientes e que seja completo, gratuito e de grande utilidade.

Apesar das dificuldades no desenvolvimento, o aplicativo está inteiramente desenvolvido e em pleno funcionamento, podendo ser acessado por meio do link <<http://ramses.uenp.edu.br/projetos/neurociencia/atentioncar/>>.

Espera-se que com este aplicativo os profissionais da área da saúde possuam uma ferramenta computadorizada para o treino de inibição comportamental de seus pacientes capaz de capturar informações precisas dos testes para um melhor monitoramento do desempenho de cada um. Com relação aos pacientes, espera-se que o treino de inibição comportamental possa ser transferido para o seu cotidiano, e fazer com que eles consigam controlar melhor seus impulsos e vontades e saibam esperar o momento mais oportuno para realizar uma determinada ação. O aplicativo é também uma maneira de tornar a consulta ao psicólogo, por exemplo, mais animadora para os pacientes, deixando-os com vontade de voltar ao consultório sabendo que irão realizar o teste inibitório em um jogo de corrida de computador.

Uma das grandes vantagens da utilização do aplicativo é que ele é capaz de capturar todos os estímulos realizados pelos pacientes durante o teste computadorizado, sem que haja perda de informação, conseqüentemente gerando relatórios extremamente precisos do desempenho dos pacientes, tirando a responsabilidade dos avaliadores de ficar o tempo todo atento para capturar os dados enquanto o paciente realiza teste de inibição comportamental manual.

Como trabalhos futuros, deverão ser realizados testes para verificar a eficácia do aplicativo no que diz respeito principalmente ao treino de inibição comportamental em pessoas com TDAH e realizar pesquisas para verificar se o aplicativo contribui de forma significativa para o tratamento de pacientes com TDAH, transferindo eficientemente os resultados obtidos no teste para o cotidiano dos pacientes.

REFERÊNCIAS

ASSOCIAÇÃO AMERICANA DE PSIQUIATRIA. **Manual Diagnóstico e Estatístico de Transtornos Mentais**. Tradução de Dayse Batista. Artes Médicas Sul, 1994, 886 p.

BESSA, Aline; SOUZA, Caio T.; BEZERRA, Carlos E.; MONTEIRO, Ivan; BANDEIRA, Humberto; SOUZA, Rodrigo. **Desenvolvimento de um moto multiplataforma para Jogos 3D**. Disponível em: <<http://www.inf.ufrgs.br/~cebbezerra/publications/bessa2007dmm.pdf>> Acesso em: 10 jun. 2013.

BITTENCOURT, João Ricardo; GIRAFFA, Lucia Maria. **Modelando Ambiente de Aprendizagem Virtuais utilizando Role-Playing Games**. Disponível em: <<http://ceie-sbc.educacao.ws/pub/index.php/sbie/article/view/299/285>> Acesso em: 10 jun. 2013.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML: guia do usuário**. 2ª Edição. Rio de Janeiro: Elsevier, 2005, 480p.

BOOT, W. R.; KRAMER, A. F.; SIMONS, D. J.; FABIANI, M.; GRATTON, G. **The effects of video game playing on attention, memory, and executive control**. Acta Psychologica, 2008, 129, 387-398.

CAVELLUCCI, Lia Cristina B. **Vivenciando e Refletindo sobre preferências de Aprendizagem**. Disponível em: <http://www.iar.unicamp.br/disciplinas/am540_2003/lia/artigo_lia.pdf> Acesso em: 15 maio 2013.

COSTA, Valéria Catelli Infantozzi; PAULA, Eldereis; XAVIER, Gilberto Fernando; BUENO, José Lino Oliveira. **Programa “DRL” para controle de pesquisas em julgamento temporal**. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0102-79722007000300019&lng=en&nrm=iso> Acesso em: 10 jun. 2013.

DALPIAZ, Marlei Teixeira; SILVEIRA, Cátia de Oliveira; TEIXEIRA, Letícia de Almeida; RIBEIRO, Luciane; TORRES, Santa Clori Nunes Ribeiro. **Transtorno de Déficit de Atenção/Hiperatividade**. Disponível em: <<http://revista.ulbratorres.com.br/artigos/artigo15.pdf>> Acesso em: 15 maio 2013.

DATE, C. J. **Introdução a Sistemas de Bancos de Dados**. Tradução de Daniel Vieira. Rio de Janeiro: Elsevier, 2003, 851p.

FOWLER, Martin. **UML Essencial: Um Breve Guia para Linguagem Padrão**. 3ª Edição. Porto Alegre: Bookman, 2005, 166p.

FRANCISCO, Thiago; CARMO, Joyce C.; ROCHA, Mariane A.; GARONE, Pricilla M. C.. **Parking Paradise: Uma proposta de jogo de estacionamento**. Disponível em:

<http://sbgames.org/sbgames2012/proceedings/papers/artedesign/AD_Short4.pdf> Acesso em: 14 jun. 2013.

GONZAGA, Flávio S.; BIRCKAN, Guilherme. **Curso de PHP e MySQL**. Disponível em: <<http://4websystems.com.br/portal/apostila/33/apostiladecursophpmysqlservidordebancodedados.pdf>> Acesso em: 09 nov. 2013

HEUSER, C. A. **Projeto de Banco de Dados**. 4ª Edição. Sagra, 1998, 206p.

JUNIOR, Delmir Peixoto de Azevedo; CAMPOS, Renato. **Definição de requisitos de software baseada numa arquitetura de modelagem de negócios**. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-65132008000100003> Acesso em: 09 nov. 2013.

MORAIS, Felipe Castanheira. **Desenvolvimento de Jogos Eletrônicos**. Disponível em: <<http://revistas.unibh.br/index.php/dcet/article/view/242/134>> Acesso em: 14 jun. 2013.

NIEDERAUER, Juliano. **Desenvolvendo Websites com PHP**. 2ª Edição. Novatec Editora Ltda. 2011, 26p.

NIEDERAUER, Juliano. **Integrando PHP 5 com MySQL**. 2ª Edição. Novatec Editora Ltda. 2008, 101p.

OFFICIAL PHP MANUAL. Disponível em < http://www.php.net/manual/pt_BR/preface.php> Acesso em: 09 nov. 2013.

RIVERO, Thiago S.; QUERINO, Emanuel H. G.; ALVES, Isabella Starling. **Videogame: seu impacto na atenção, percepção e funções executivas**. Revista Neuropsicologia Latinoamericana. Vol. 4. No. 2. 2012, 38-47.

RODRIGUES, Herbet F.; MACHADO, Liliane dos S. Machado; VALENÇA, Ana Marian G.. **Definição e Aplicação de um Modelo de Processo para Desenvolvimento de Serious Games na Área de Saúde**. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/wim/2010/009.pdf>> Acesso em: 10 jun. 2013.

ROHDE, Luis Augusto; BARBOSA, Genário; TRAMONTINA, Silzá; POLANCZYK, Guilherme. **Transtorno de Déficit de Atenção/Hiperatividade**. Disponível em: <http://www.scielo.br/scielo.php?pid=S1516-44462000000600003&script=sci_arttext> Acesso em: 15 maio 2013.

ROHDE, Luis Augusto; MATTOS, Paul. **Princípios e Práticas em TDAH**. Artmed Editora S. A., 2008.

ROSA, Maurício; MALTEMPI, Marcos Vinicius. **A avaliação sob o aspecto da educação a distancia**. Disponível em: <<http://www.scielo.br/pdf/ensaio/v14n50/30407.pdf>> Acesso em: 10 jun. 2013.

ROSA, Maurício; MALTEMPI, Marcos Vinicius. **RPG Maker: Uma proposta para unir jogo, informática e educação.** Disponível em: <<http://www.rc.unesp.br/igce/demac/maltempi/Publicacao/Rosa-Maltempi-sipem03.pdf>> Acesso em: 10 jun. 2013.

SANTOS, Letícia de Faria; VASCONCELOS, Laércia Abreu. **Transtorno de Déficit de Atenção e Hiperatividade em Crianças: Uma Revisão Interdisciplinar.** Disponível em: <<http://www.scielo.br/pdf/ptp/v26n4/15.pdf>> Acesso em: 10 jun. 2013.

SCIRRA. **Official Construct 2 Manual.** Disponível em <<https://www.scirra.com/manual/1/construct-2>> Acesso em: 15 jun. 2013.

VASCONCELOS, Mario M.; WERNER, Jairo; MALHEIROS, Ana Flávia de Araújo; LIMA, Daniel Fampa Negreiros; SANTOS, Ítalo Souza Oliveira; BARBOSA, Jane Bardawil. **Prevalência do Transtorno de Déficit de Atenção/Hiperatividade numa Escola Pública Primária.** Disponível em: < <http://www.scielo.br/pdf/anp/v61n1/15018.pdf>> Acesso em: 19 jun. 2013.

APÊNDICE A – Explicação dos casos de uso do Speed Car

1. Consultar Senha

1.1. Breve Descrição

Este caso de uso ocorre quando o avaliador esqueceu sua senha e quer recuperá-la.

1.2. Ator

Avaliador.

1.3. Pré-Requisito

O avaliador deve estar cadastrado no sistema.

1.4. Fluxo Principal

1. O avaliador clica no botão “Esqueci minha senha” na página principal.
2. O sistema exibe os campos necessários para a busca da senha.
3. O avaliador preenche os campos com seus dados.
4. O sistema exibe a senha do avaliador.
5. O avaliador volta à página principal.

1.5. Fluxo Alternativo

- 3a. O avaliador não está cadastrado.
 - 3a. 1 O sistema exibe uma mensagem que o avaliador não existe.
 - 3a. 2 O avaliador volta à página principal.

1.6. Pós-Condição

O avaliador recupera sua senha.

2. Cadastrar Avaliador

2.1. Breve Descrição

Este caso de uso ocorre quando o avaliador vai se cadastrar no sistema.

2.2. Ator

Avaliador.

2.3. Pré-Requisito

O avaliador não deve estar cadastrado no sistema.

2.4. Fluxo Principal

1. O avaliador clica no botão “Cadastrar novo usuário” na página principal.
2. O sistema exibe os campos necessários para o cadastro do avaliador.
3. O avaliador preenche os campos com seus dados.
4. O sistema verifica se já existe outro avaliador com o mesmo usuário.
5. O avaliador finaliza o cadastro.

2.5. Fluxo Alternativo

- 4a. O usuário já existe.
 - 4a. 1 O sistema exibe uma mensagem que este usuário já existe.
 - 4a. 2 O avaliador digita outro nome no campo “usuário”.
 - 4a. 3 Retorna ao passo 5.

2.6. Pós-Condição

O avaliador está cadastrado no sistema.

3. Gerenciar Sujeito

3.1. Breve Descrição

Este caso de uso ocorre quando o avaliador vai selecionar um sujeito.

3.2. Ator

Avaliador.

3.3. Pré-Requisito

O avaliador deve estar logado.

3.4. Fluxo Principal

1. O avaliador seleciona o sujeito que irá realizar o teste.

2. O avaliador clica no botão “entrar”.

3.5. Fluxo Alternativo

- 1a. Sujeito não cadastrado.
 - 1a. 1 O avaliador clica no botão cadastrar sujeito.
 - 1a. 2 Vai para o caso de uso Cadastrar Sujeito.

3.6. Pós-Condição

O sujeito que irá realizar o teste está selecionado.

4. Cadastrar Sujeito

4.1. Breve Descrição

Este caso de uso ocorre quando o avaliador vai cadastrar um novo sujeito.

4.2. Ator

Avaliador.

4.3. Pré-Requisito

O avaliador deve estar logado.

4.4. Fluxo Principal

1. O avaliador clica no botão “Cadastrar Jogador”.
2. O sistema exibe os campos necessários para o cadastro do sujeito.
3. O avaliador preenche os campos com o dados do sujeito.
4. O avaliador clica em “Salvar”.
5. O sistema verifica se existe algum campo vazio.
6. O sistema volta à página de selecionar sujeito

4.5. Fluxo Alternativo

- 5a. Campo vazio.
 - 5a. 1 O sistema exibe uma mensagem no campo vazio.
 - 5a. 2 Retorna ao item 3.

4.6. Pós-Condição

O sujeito foi cadastrado na lista de sujeitos do avaliador.

5. Gerar Relatório

5.1. Breve Descrição

Este caso de uso ocorre quando o avaliador quer fazer o download do relatório do teste realizado por um sujeito.

5.2. Ator

Avaliador.

5.3. Pré-Requisito

O avaliador deve estar logado e o sujeito selecionado.

5.4. Fluxo Principal

1. O avaliador clica no botão Relatórios.
2. O sistema exibe os testes realizados pelo sujeito.
3. O avaliador escolhe o teste que quer fazer o download do relatório.
4. O avaliador clica no botão "Download".
5. O sistema exibe o relatório em PDF.
6. O avaliador imprimir o relatório ou salva em seu computador.
7. O avaliador clica no botão "Voltar".

5.5. Fluxo Alternativo

- 3a. Não há testes realizados
 - 5a. 1 O sistema exibe uma mensagem de não há testes.
 - 5a. 2 Retorna ao item 7.

5.6. Pós-Condição

Um relatório é gerado para *download* do avaliador.

6. Treinar

6.1. Breve Descrição

Este caso de uso ocorre quando o sujeito escolhe a opção “treinar”.

6.2. Ator

Sujeito.

6.3. Pré-Requisito

O avaliador deve estar logado e o sujeito selecionado.

6.4. Fluxo Principal

1. O sujeito clica no botão “treinar”.
2. O sujeito seleciona o carro que deseja utilizar.
3. O sujeito seleciona a pista que deseja correr.
4. O sujeito seleciona o nível de dificuldade da corrida.
5. O sujeito realiza uma corrida de dez voltas contra um carro guiado pelo aplicativo.
6. O sujeito seleciona a opção “Mudar modo de jogo”.
7. O sistema volta para a tela de seleção de modo de jogo.

6.5. Fluxo Alternativo

- 6a. Sujeito seleciona a opção “Jogar novamente”.
 - 6a. 1 Retorna ao item 2.
- 6b. Sujeito seleciona a opção “Mudar Jogador”.
 - 6b. 1 Volta ao caso de uso Selecionar Sujeito.

6.6. Pós-Condição

O sujeito treinou para se familiarizar com o aplicativo.

7. Realizar Teste

7.1. Breve Descrição

Este caso de uso ocorre quando o sujeito escolhe a opção “realizar teste”.

7.2. Ator

Sujeito.

7.3. Pré-Requisito

O avaliador deve estar logado e o sujeito selecionado.

7.4. Fluxo Principal

1. O sujeito clica no botão “realizar teste”.
2. O sujeito seleciona o carro que deseja utilizar.
3. O sujeito visualiza a pista que irá correr.
4. O sujeito visualiza o nível de dificuldade da corrida.
5. O sujeito realiza uma corrida de dez voltas contra um carro guiado pelo aplicativo.
6. O sujeito seleciona a opção “Continuar”.
7. O sistema aumenta o nível de dificuldade da corrida.
8. O sistema volta para o item 4.

7.5. Fluxo Alternativo

6a. Fim do teste.

6a. 1 O sujeito seleciona a opção “Fim do Teste”.

6a. 2 Volta ao caso de uso Selecionar Sujeito.

7a. Próxima pista.

7a. 1 O sistema seleciona a próxima pista.

7a. 2 O sistema seleciona o nível de dificuldade “fácil”.

7a. 3 Retorna ao item 3.

7.6. Pós-Condição

O sujeito realizou o teste neuropsicológico.