**UNIVERSIDADE PAULISTA – UNIP**

**CAIQUE SANTOS AMARAL**

**GABRIEL YMAMURA**

**JEFERSON FEREIRA SILVA**

**NILTON LUCIO CORREIA**

**RODOLFO GONÇALVES MACHADO**

**PROJETO DE PESQUISA:**

*framework* para jogos de corrida em *Android* com Lógica Paraconsistente

**SÃO PAULO**

**2016**

**CAIQUE SANTOS AMARAL**

**GABRIEL YMAMURA**

**JEFERSON FEREIRA SILVA**

**NILTON LUCIO CORREIA**

**RODOLFO GONÇALVES MACHADO**

**PROJETO DE PESQUISA:**

*framework* para jogos de corrida em *Android* com Lógica Paraconsistente

Trabalho de conclusão de curso para obtenção do título de graduação em Ciência da Computação apresentado à Universidade Paulista – UNIP.

Orientadora: Profa. Me. Amanda Luiza Pereira

**SÃO PAULO**

**2016**

**FICHA CATALOGRÁFICA**

**A VERIFICAR**

**CAIQUE SANTOS AMARAL**

**GABRIEL YMAMURA**

**JEFERSON FEREIRA SILVA**

**NILTON LUCIO CORREIA**

**RODOLFO GONÇALVES MACHADO**

**PROJETO DE PESQUISA:**

*framework* para jogos de corrida em *Android* com Lógica Paraconsistente

Trabalho de conclusão de curso para obtenção do título de graduação em Ciência da Computação apresentado à Universidade Paulista – UNIP.

Aprovado em:

BANCA EXAMINADORA

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_/\_\_\_\_/\_\_\_\_

Prof. 1

Universidade Paulista – UNIP

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_/\_\_\_\_/\_\_\_\_

Prof. M.e Fábio Luís Pereira

Universidade Paulista – UNIP

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_/\_\_\_\_/\_\_\_\_

Prof.ª M.e Amanda Luiza Pereira

Universidade Paulista – UNIP

**DEDICATÓRIA**

Dedicamos este trabalho às mulheres que fazem a nossa vida mais feliz, que são companheiras e que têm sustentado nosso humor nessa etapa conclusiva que é nossa conclusão de curso.

**AGRADECIMENTOS**

Agradecemos em primeiro lugar à Deus pelo dom das nossas vidas e daqueles que amamos;

Aos nossos pais pelo carinho e tempo dedicado nos incentivando a sermos pessoas melhores, esforçados e trabalhadores;

A esta universidade, ao corpo docente, direção e administração que nos deram a oportunidade de cursar neste campus, pela confiança e ética aqui presente;

Em especial aos professores Amanda Luiza Pereira, Fábio Luís Pereira e Marcos Rodrigues por todo esforço dispensado em prol da nossa educação, a todas as documentações e projetos a que fomos submetidos que nos abriram os olhos para como as coisas são, por tentarem fazer de nós pessoas e profissionais melhores, por acreditarem que poderíamos chegar até aqui e cá estamos;

Aos nossos colegas de turma pela companhia agradável nesses quatro anos de curso, pelas risadas e pelas discussões;

A todos aqueles que direta ou indiretamente colaboraram para que aqui chegássemos fica o nosso muito obrigado.

**EPÍGRAFE**

*“Você faz suas escolhas e suas escolhas fazem você”.*

*(William Shakespeare)*

**RESUMO**

O mercado de jogos está em ascensão no Brasil, gerando uma grande comoção no sentido do desenvolvimento de jogos, especialmente no segmento de dispositivos portáteis em plataforma *Android,* onde atualmente é a mais utilizada.

Partindo desses fatos e percebendo a grande evolução dos jogos no sentido de se parecerem cada vez mais com a realidade e o não aproveitamento dos jogos do gênero de corrida, o presente trabalho consiste no desenvolvimento de um *framework* para jogos de corrida em plataforma *Android* apoiado na Lógica Paraconsistente Anotada Evidencial E (LPA), utilizando o Método Paraconsistente de Decisão (MPD) como principal base para as decisões tomadas dentro do jogo e, uma vez aplicado, esse método irá trazer maior realidade as escolhas a serem efetuadas, proporcionando, assim, uma imersão maior ao jogador e, para demonstrar a utilização do *framework* será elaborado um protótipo de jogo de corrida.

De forma que, no decorrer deste trabalho serão apresentados os conceitos necessários para que se dê o desenvolvimento dos mesmos.

Palavras-chave: Mobilidade, tomada de decisão, *smartphone*.

**ABSTRACT**

The gaming market on Brazil is rising, making a great commotion in game development, specially on Android mobile devices where it is currently most used.

Starting on its facts and realizing a big game evolution in the way they look more like reality and the no use of racing games this paper consists on developing a race-gamming framework on Android platform devices supported on Paraconsistent Evidencial Noted Logic EPNL using the Paraconsistent Decision Method (PDM) as the main base for all in-game decisions and, once applied, this method will bring more reality for the choices to be made, providing a big immersion for the player and, to demonstrate the framework usage will be developed a race game prototype.

For that, during the development of this paper all the necessary concepts for develop them will be presented.

Keywords: Mobility, decision taking, smartphone.

**INDICE DE ILUSTRAÇÕES**

[Figura 1 – Ramificações da Lógica 20](#_Toc451789658)

[Figura 2 - Comparativo entre as Lógicas 22](#_Toc451789659)

[Figura 3 - Infográfico Consumo Brasileiro de Jogos 36](#_Toc451789660)

[Figura 4 - Plataforma mobile mais utilizada 36](#_Toc451789661)

[Figura 5 - Mais baixados na PlayStore 2016 37](#_Toc451789662)

**SUMÁRIO**

[1 INTRODUÇÃO 11](#_Toc452831039)

[2 REFERENCIAIS TEÓRICOS 16](#_Toc452831040)

[2.1 Lógica 16](#_Toc452831041)

[2.1.1 Lógica Clássica 17](#_Toc452831042)

[2.1.2 Lógica Não Clássica 18](#_Toc452831043)

[2.1.3 Lógica Paraconsistente Anotada Evidencial E (LPA) 19](#_Toc452831044)

[2.1.4 Método Paraconsistente De Decisão (MPD) 21](#_Toc452831045)

[2.2 Inteligência 22](#_Toc452831046)

[2.2.1 Inteligência Natural 22](#_Toc452831047)

[2.2.2 Inteligência Artificial 23](#_Toc452831048)

[2.3 Tecnologias 25](#_Toc452831049)

[2.3.1 Java 25](#_Toc452831050)

[2.3.2 Android 26](#_Toc452831051)

[2.3.3 Mobile 27](#_Toc452831052)

[2.3.4 Game Engine 28](#_Toc452831053)

[2.3.5 Cocos2D 28](#_Toc452831054)

[2.3.6 Framework 30](#_Toc452831055)

[2.4 Jogos 32](#_Toc452831056)

[2.4.1 Jogos Digitais 32](#_Toc452831057)

[2.4.2 Pesquisa Mercadológica 34](#_Toc452831058)

[3 DESENVOLVIMENTO 37](#_Toc452831059)

[4 CONSIDERAÇÕES FINAIS 38](#_Toc452831060)

[5 REFERÊNCIAS 39](#_Toc452831061)

# **INTRODUÇÃO**

A temática macro deste trabalho é a elaboração de um *framework* utilizando Lógica Paraconsistente Anotada Evidencial E (LPA), fundamentada em diversas pesquisas que abordam o tema dentro de um aspecto científico.

Quanto à escolha do tema se faz necessário observar algumas demandas como, por exemplo, “[...] gostar do assunto, ter acesso a informações e dados necessários, ter tempo e outras condições materiais necessárias e que tal tema seja de interesse social” (DOMINGOS; SANTOS, 1998, p. 208).

O desenvolvimento de aplicativos *Android* está em ascensão atualmente, e a LPA ainda pode ser considerada um campo em âmbito exploratório quando o assunto é aplicá-la dentro das áreas de tecnologia e o acesso às informações bibliográficas sobre a mesma ainda é escasso, sendo o tema mais encontrado em artigos e monografias do que em livros.

[...]o assunto a ser escolhido, pode surgir de uma dificuldade prática, de uma curiosidade científica, independentemente de sua origem, necessariamente deve possuir um caráter amplo e descrever em linhas gerais o assunto que será base para a pesquisa[..] (LAKATOS;MARCONI, 2010, p. 201).

Apesar de ser a primeira etapa do plano da pesquisa científica são encontradas algumas dificuldades em sua elaboração:

A escolha de um tema é o primeiro passo no planejamento da pesquisa, não necessariamente o mais fácil. Não falta, evidentemente, tema para pesquisa: a dificuldade está em decidir-se por um deles. Para muitos pesquisadores, a decisão final é procedida por momentos de verdadeira angústia[...] (CERVO; BERVIAN; SILVA, 2007, p. 73).

De acordo com as definições acima, conclui-se que para um tema de amplo interesse, todos os envolvidos têm que estar em sintonia e gostar das ideias propostas, pois esta será a base para a realização da pesquisa.

Com a definição do tema é necessário delimitá-lo, para isso, segundo Domingos e Santos (1998), deve ser discutido levando em consideração os objetivos do trabalho, amplos ou restritos, em conjunto com a disponibilidade de recursos materiais e humanos. O processo da mesma só é finalizado quando se faz uma limitação completa, geográfica e espacial, com o objetivo da pesquisa (LAKATOS; MARCONI, 2010).

Partindo dessas definições a pesquisa está centrada no desenvolvimento de um *framework* aplicando LPA para jogos de corrida em *Android*, a escolha dos limites desta pesquisa foi definida após a validação dos conhecimentos dos recursos e a disponibilidade de fontes de pesquisas sobre os assuntos, para que assim a dificuldade sobre o tema seja atenuada, como descrevem Cervo, Bervian e Silva (2007, p. 74), “[...] convém superar a dificuldade muito comum de escolher temas, pelo seu nível de dificuldade e tamanho, delimitar é focar em partes para facilitar a operação de tarefas e as definições dos termos”.

A LPA possui um grande potencial para ser aplicada dentro do ramo tecnológico, devido a sua ampla capacidade para tomada de decisões, com isso será utilizada para o gerenciamento das decisões dentro do jogo e assim “[...] mostrar a importância do assunto, tendo em vista o conhecimento geral do mesmo e a temática proposta [...]” (DOMINGOS; SANTOS, 1998, p.210).

Com isso é possível determinar que o objetivo geral de acordo com Lakatos e Marconi (2010, p. 202), está sendo respeitado, pois conforme as autoras “[...] relaciona-se com o conteúdo intrínseco, quer dos fenômenos e eventos, quer das ideias estudadas. Vincula-se diretamente à própria significação da tese proposta pelo projeto”.

Assim sendo, a utilização da LPA dentro do sistema será empregada de forma que ela seja a responsável pela tomada das decisões, levando-se em conta a probabilidade de determinadas ações ocorrerem em um jogo de corrida, como por exemplo, ultrapassagens em curvas, efetuar ou não a troca dos pneus, efetuar decisões mediante à possibilidade de colisão, momento de realizar o *pit stop*, entre outras.

Para tanto se faz necessário esboçar os objetivos específicos, nos quais são apresentados de forma detalhada as ações que se pretendem alcançar, pois “[...] define o ponto central do trabalho. Isso quer dizer que, dentro de uma ideia geral do trabalho, deve-se ressaltar a ideia específica a ser desenvolvida [...]” (DOMINGOS; SANTOS, 1998, p. 210).

Já Lakatos e Marconi (2010, p. 202) dizem que os objetivos específicos são intrínsecos ao objetivo geral e ao tema, mas de outra forma, pois “[...] apresentam caráter mais concreto. Tem função intermediária e instrumental, permitindo, de um lado, atingir o objetivo geral e, do outro, aplicá-lo a situações particulares”.

Com isso conclui-se que o objetivo específico é o passo necessário para que se alcance a completude do objetivo geral e do trabalho (incluindo a verificação da hipótese).

Partindo desta premissa, os objetivos específicos abordados neste projeto de pesquisa e que serão detalhados mais à frente no trabalho são: Lógica, Lógica Clássica, Lógica não Clássica, Lógica Paraconsistente Anotada Evidencial E (LPA), Método Paraconsistente de Decisão (MPD), Inteligência, Inteligência Artificial, JAVA, *Android*, *mobile*, *game engine*, Cocos2D, *framework*, Jogos, Jogos Digitais e Pesquisa Mercadológica.

Para poder falar de LPA, antes de mais nada, é preciso compreender o que é a Lógica e como ela se aplica no cotidiano das pessoas. Logo, se tem que Lógica é a capacidade de argumentar, pensar, ter ideias, manifestar o conhecimento a procura da verdade e a partir dessa capacidade surge a Lógica Clássica onde se obtém como conceito que uma informação ou é falsa ou verdadeira, diferente da Lógica Não Clássica, na qual assume que a informação varia entre falsa e verdadeira.

Dentro da Lógica Não Clássica encontra-se a LPA, que apesar de ser uma Lógica Não Clássica, difere dos outros tipos uma vez que as outras Lógicas excluem a contradição, ela utiliza tanto da verdade como da contradição para montar o argumento lógico e auxiliar na tomada de decisão, como será detalhado posteriormente.

Uma vez conhecido o conceito da LPA, se faz necessário saber o conceito de Inteligência, essa que está ligada diretamente à forma de como toma-se decisões, que é a capacidade de escolher diferentes opções, ou mais especificamente nesse caso, de escolher por diferentes decisões e no decorrer dos anos surgiu a tentativa de reproduzir a Inteligência humana em máquinas criadas pelo homem, neste ponto nasce a Inteligência Artificial que tenta reproduzir a Inteligência de forma a automatizar processos comuns do dia-a-dia de forma autônoma, neste projeto a Inteligência Artificial irá comandar as ações dos outros jogadores do protótipo.

Desta forma, com a conceituação de Lógica e Inteligência este trabalho pode avançar para a segunda parte, onde será focado em falar do *Framework*, porém para um melhor entendimento do assunto, deve-se conceituar uma *game engine*, que é uma plataforma completa de desenvolvimento de jogos, no passo que *framework* é uma especialização, uma parte à parte da *engine* que referencia algo mais específico, nesse caso, será tratado o desenvolvimento de um *framework* utilizando LPA, demonstrado através de um jogo de corrida, que serão mais detalhados na seção de desenvolvimento desta pesquisa.

Até o momento pode-se dizer que, neste projeto, será desenvolvido um *framework*, para jogos de corrida em plataforma *Android*, com Lógica Paraconsistente, onde a LPA irá dizer através de dados fornecidos se a opção desejada é a mais válida, por ser mais verdadeira. Para auxiliar no desenvolvimento do protótipo e do *framework*, será utilizado a game engine Cocos2D, uma multiplataforma de desenvolvimento de aplicativos e jogos para smartphones de código aberto que utiliza como base a linguagem de programação JAVA, linguagem essa oriunda das Linguagens C e C++, baseada em classes e orientada a objetos e que foi projetada de forma que seja simples e fácil de usar.

E com isso pode-se passar para a última, mas não menos importante, parte do projeto que é a prototipação de um jogo de corrida, mas antes de falar propriamente de jogos, é necessário dizer que esse conceito advém da teoria dos jogos, a qual observa a interação entre dois ou mais agentes de decisão interagindo entre si, com isso, será preciso definir o sistema básico de um jogo, como um jogo é dividido e, por fim, jogos digitais, utilizando todos os conceitos explicitados aqui e detalhados mais adiante.

Mas, para que todo nosso argumento seja válido, faz-se necessário justificá-lo e para isso, em relação à justificativa, tem-se que:

Como todo pesquisador normalmente trabalha ou é financiado por instituições, ele tem que justificar as razões para a realização do trabalho. Essa justificativa servirá também para mostrar o porquê de prováveis limitações à proposta de trabalho (DOMINGOS; SANTOS, 1998, p. 210).

Já Lakatos e Marconi (2010, p. 202) dizem que “é o único item do projeto que apresenta respostas à questão por quê? ”, uma vez que expõe de forma sucinta, porém completa, as razões teóricas e os motivos práticos que dão a importância da realização da pesquisa.

Para Cervo, Bervian e Silva (2007) é na justificativa que temos a clareza daquilo que será realizado ao longo do trabalho, tanto das questões teóricas como práticas.

A justificativa deste trabalho ainda não foi realizada devido a necessidade de um maior entendimento dos outros elementos de pesquisa, para uma base teórica que possibilite o desenvolvimento do mesmo.

# **REFERENCIAIS TEÓRICOS**

Este trabalho irá tratar sobre o desenvolvimento de um framework baseado em Lógica Paraconsistente Anotada Evidencial E (LPA) para jogos de corrida em plataforma *Android*.

Nesta seção serão abordados os conceitos pesquisados para que se dê o desenvolvimento posterior do projeto como um todo.

Encontra-se dividido da seguinte forma:

* Lógica:
  + Lógica Clássica;
  + Lógica Não Clássica;
  + Lógica Paraconsistente Anotada Evidencial E (LPA);
  + Método Paraconsistente de Decisão (MPD).
* Inteligência:
  + Inteligência Natural;
  + Inteligência Artificial.
* Tecnologias:
  + Java;
  + *Android*;
  + *Mobile*;
  + *Game engine*;
  + Cocos2D;
  + *Framework*.
* Jogos:
  + Jogos Digitais;
  + Pesquisa Mercadológica.

## Lógica

O primeiro item desse capítulo trata sobre os conceitos de Lógica Clássica e Não Clássica para dar entendimento a Lógica Paraconsistente Anotada Evidencial E (LPA) e o Método Paraconsistente de Decisão (MPD) que serão utilizados como parte do desenvolvimento futuro tanto do *framework* como do protótipo do jogo de corrida.

Existem várias formas de conceituar a lógica, mas segundo Copi (1977, p.15) a “Lógica é uma ciência do raciocínio”. Partindo desse princípio, pode-se verificar a evolução da Lógica através do tempo, para isso dividiremos aqui em três períodos:

1. Período Aristotélico: vai desde Aristóteles (384 – 322 a.C) até o início do século XIX, até então os avanços no campo da lógica não foram tão profundos;
2. Período Booleano: influenciado principalmente por George Boole (1815 - 1864), houve uma significativa evolução da lógica, especialmente por ideias algébricas sendo empregadas no domínio da lógica e, nessa época, pode-se destacar Gottlob Frege (1848 - 1925) que edificou um sistema lógico sem precedentes que deu origem a várias das características da lógica atual através do afastamento das ideias algébricas e aplicação de formas linguísticas ou proposicionais, como será visto mais à frente;
3. Período Contemporâneo: Período mais atual onde houveram evoluções significativas no que diz respeito a Lógica como um todo, aqui valem citar alguns nomes muito importantes para a época, tais como: Alfred North Whitehead (1861 - 1947) e Bertrand Russel (1872 - 1970) com a obra *Pincipia Mathematica* em três volumes, Kurt Gödel (1906 - 1978) com seu teorema da incompleteza, Alan Mathison Turing (1912 - 1954) com sua teoria geral dos processos computáveis, a teoria da recursão de Alonzo Church (1903 - 1995), entre outros.

### Lógica Clássica

A Lógica Clássica é caracterizada por seu núcleo, por possuir uma linguagem bem definida e uma semântica padrão, entre suas características estão:

* Baseando-se em uma proposição, através de uma expressão com linguagem consistente e bem definida, e da aplicação de diferentes técnicas, criar novas proposições derivadas da primeira;
* Através da lógica podem ser elaboradas inferências e formular teorias;
* É possível a obtenção de conclusões através de suas premissas, bem como deduzir sobre princípios básicos de uma ciência;
* Pode ser representada por: Verdadeiro (V ou ‘1’) ou Falso (F ou ‘0’);
* Sua linguagem possui símbolos de operações para conjunção (^), disjunção (v), implicação () e negação (~) tornando possível a formação de expressões que relacionam seus objetos permitindo assim, o cálculo de predicado clássico de primeira ordem;
* Princípios Fundamentais da Lógica Clássica:

Princípio da Identidade:

x = x,

Ou seja, todo objeto é idêntico a si mesmo.

Princípio do terceiro excluído:

p v ¬p,

Ou seja, de duas proposições contraditórias, (isto é, tais que uma é negação da outra), uma delas é verdadeira.

Princípio da contradição, ou não contradição:

¬(p ^ ¬p),

Isto é, entre duas proposições contraditórias, uma é falsa.

Princípio da identidade proposicional:

p p,

Isto é, citando Russell, “once true, always true; once false, always false”.

(COSTA, ABE, *et al.*, 1999, p. 13)

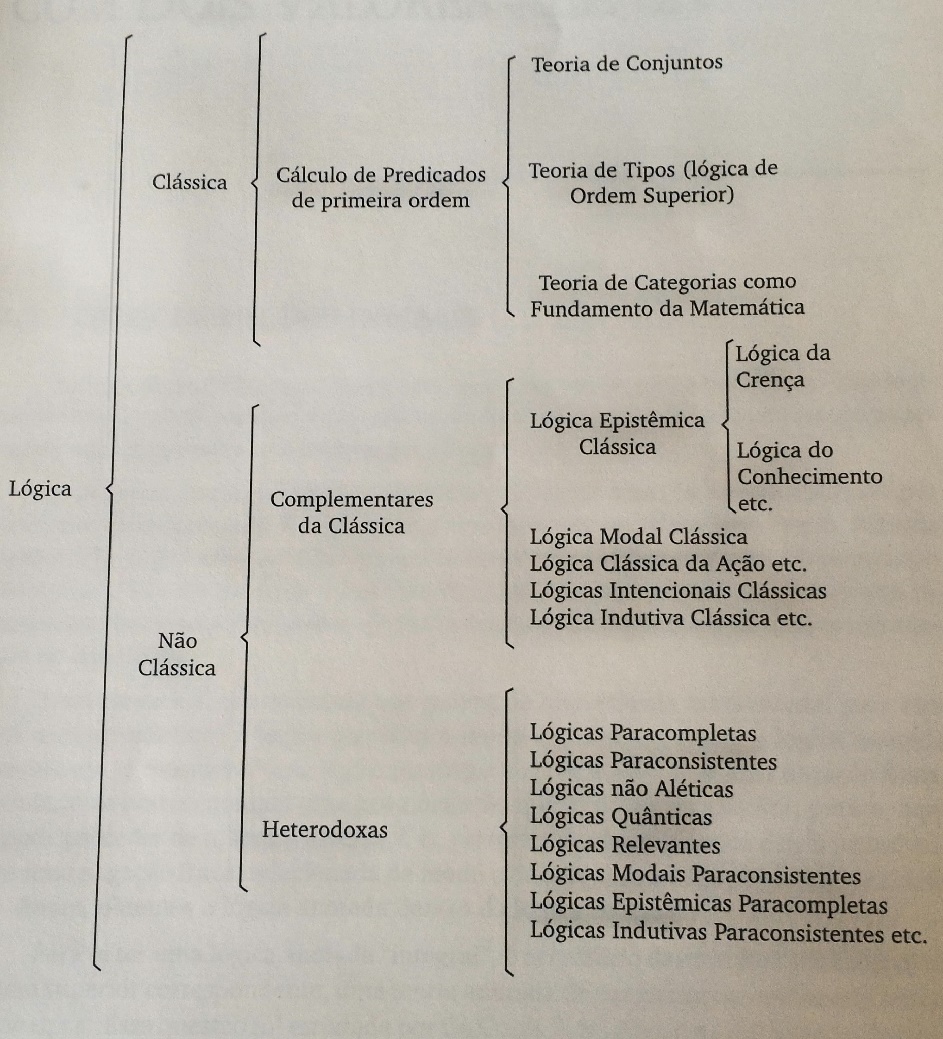
Tendo elucidado um pouco sobre a Lógica Clássica, antes de falarmos da LPA propriamente dita, será abordada a Lógica Não Clássica.

### Lógica Não Clássica

Antes de mais nada, é preciso saber que a diferença entre a Lógica Clássica e Não Clássica está em que na Lógica Não Clássica existem modificações ou restrições dos princípios da Lógica Clássica, se dividem em Complementares da Clássica e Heterodoxas, das quais iremos nos aprofundas somente na Lógica Paraconsistente, que faz parte das Heterodoxas, uma vez que ignoram a lei da contradição, explicada anteriormente.

A seguir segue uma imagem para melhor compreensão sobre como a Lógica está dividida:

Figura 1 – Ramificações da Lógica



Fonte: (da Costa, 1999)

Agora, com esses conceitos esclarecidos, será abordado um dos temas principais deste trabalho que é a LPA.

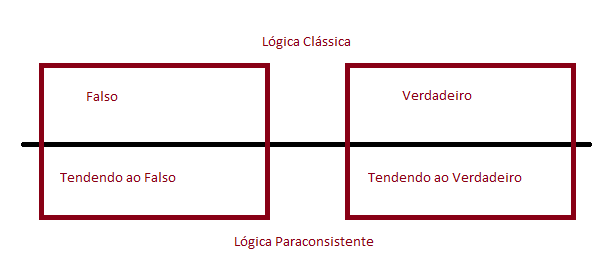
### Lógica Paraconsistente Anotada Evidencial E (LPA)

Sabe-se que a LPA pertence à família das Lógicas Não Clássicas Heterodoxas, ou seja, vai contra algum fundamento da Lógica Clássica.

Partindo dos trabalhos dos lógicos russo Nicolai A. Vasiliev e o polonês Jan Lukasiewicz, que, de forma independente publicaram trabalhos sobre a possibilidade da existência de uma lógica que não excluísse as contradições, diferentemente da Lógica Clássica, no final da década de 50 o lógico brasileiro Prof. Dr. Newton Carneiro Affonso da Costa, criou a Lógica Paraconsistente (inicialmente nomeada como Sistemas Cn), ainda que de forma sintática. Mesmo havendo certo ceticismo por parte do mundo acadêmico entre os matemáticos, lógicos e cientistas pela não existência de casos práticos de uso, por volta de 1965 às Lógicas Cn foi acrescentado o processo de algebrização, dando origem ao conceito de álgebra de Curry, e por último, mas não menos importante, surgiram propostas quanto à semântica para as lógicas Cn também conhecida como teoria das valorações por Abe, e finalmente veio a receber o nome de Lógica Paraconsistente por sugestão do Filósofo peruano Francisco Miro Quesada, onde na utilização do termo paraconsistente temos que significa próximo à consistência, ou além da consistência (ABE, et al., 2011).

Para poder continuar é preciso relembrar o princípio da não contradição da Lógica Clássica, na qual, entre duas proposições contraditórias, somente uma delas é verdadeira, princípio esse que firma a consistência da Lógica Clássica de que somente é possível tomar toda e qualquer proposição como verdadeira ou falsa, não existe o meio-termo, de forma diferente do que ocorre na vida real, onde todos se deparam com várias situações contraditórias onde há a necessidade da tomada de decisões. Para exemplificar essa diferença temos a seguinte imagem:

Figura 2 - Comparativo entre as Lógicas

****

Fonte: Próprio autor

Dessa forma, temos que a LPA permite, diferentemente da Lógica Clássica, associa mais valores além de Falso e Verdadeiro, uma vez que usa de Graus de Certeza e Incerteza para cada proposição em pares (μ; λ), normalmente referenciados como constantes de anotação que podem assumir valores entre 0 e 1, assim sendo, a notação de uma proposição ***p*** é tal que ***p*** (μ; λ) (ABE, et. al., 2011).

Falta ainda um melhor entendimento sobre os processos de Maximização e Minimalização que envolvem a LPA para incorporar a este trabalho.

### Método Paraconsistente De Decisão (MPD)

O Método Paraconsistente de Decisão, para ser acrescido a este trabalho. Mas sabe-se que ele é o método que será utilizado para as tomadas de decisão dos *framework* e protótipo a serem desenvolvidos.

## Inteligência

No Capítulo anterior foram vistos os conceitos da Lógica e sua aplicação, já neste capítulo se faz necessário uma abordagem singela do que seja a Inteligência, pois se a Lógica é uma ciência do raciocínio, então ela é um atributo do que seria a Inteligência, pois é um atributo humano e, dentre algumas de suas características estão o ato de compreender, raciocinar, pensar e interpretar, diferenciando a pessoa (o ser humano) do animal.

A área de pesquisa em Inteligência tem mostrado significantes resultados no comportamento social humano.

[...] especialmente porque elas são frequentemente relacionadas a comportamentos socialmente valorizados, tais como desempenho acadêmico, desenvolvimento profissional, traços de personalidade, entre outros(SANTOS;NORONHA, 2005, p.192).

A palavra "inteligência" vem do latim *intelligentia*​, originário de *intelligere*, onde a palavra *inter* significa "entre", e *legere* quer dizer "escolha", portanto, é a capacidade de um indivíduo escolher entre as possibilidades ou opções que lhe são apresentadas.

### Inteligência Natural

Na escolha da mais adequada oportunidade entre as diversas opções, o indivíduo tem que pensar ao máximo em todas as vantagens e desvantagens apresentadas e, para tanto, precisa compreender, pensar e raciocinar para se ter a base do que forma a inteligência.

A inteligência, também está relacionada ao funcionamento e uso da abstração, da memória, do juízo, da concepção e da imaginação. No ramo da psicologia, a chamada “inteligência psicológica”, é a capacidade de aprender e relacionar, é a cognição de um indivíduo e a capacidade de se adaptar a novos habitats ou situações.

A inteligência humana está ligada à sua capacidade de interagir com o meio através de habilidades cognitivas (sentidos) e conotativas (ação), isto é, se movimentar, reconhecer sons (fala) e imagens, se expressar, etc. Quando se procura reproduzir estas características em dispositivos construídos pelo homem, referindo-se ao ramo da Ciência da computação conhecido como Inteligência Artificial.

### Inteligência Artificial

A Inteligência Artificial ouIA se preocupa em desenvolver mecanismos e dispositivos tecnológicos que consigam simular o cérebro dos seres humanos, isto é, a sua inteligência, como detalhado:

[...] IA é a parte da ciência da computação voltada para o desenvolvimento de sistemas de computadores inteligentes, i.e. sistemas que exibem características, as quais nós associamos com a inteligência no comportamento humano - e.g. compreensão da linguagem, aprendizado, raciocínio, resolução de problemas, etc. (FEIGENBAUM; EDWARD, 1981, p. 3)

O início da Inteligência Artificial se dá em torno dos anos de 1940, as pesquisas para essa ciência foram desenvolvidas apenas para encontrar novas funcionalidades para o computador, ainda em projeto. Com o surgimento da Segunda Guerra Mundial, surgiu também a necessidade de desenvolver a tecnologia para motivar a indústria bélica.

No decorrer do tempo, surgem técnicas da IA, como por exemplo relacionados ao paradigma biológico, que usa a IA para desenvolver conceitos com o objetivo de reproduzir as redes neurais humanas. Com isso nos anos 60 recebe o nome de Inteligência Artificial, fazendo com que os pesquisadores biológicos acreditassem ser possível a criação de máquinas que realizassem tarefas, como raciocinar.

Atualmente existem muitas aplicações que utilizam IA, como por exemplo: jogos, softwares, aplicativos de segurança, robótica (robôs auxiliares), reconhecimento de voz, programas de diagnósticos médicos, entre outros. A IA é um tema bastante recorrente em ficção científica, estando presente em livros, desenhos animados, filmes, entre outros. Um autor de grande destaque nesta área é o russo Isaac Asimov, autor de histórias de sucesso como “O Homem Bicentenário” e “Eu, Robô”. Ambas receberam adaptação para o cinema. Outra história que caiu nas graças do público foi “A.I: Inteligência Artificial”, dirigido por Steven Spielberg.

As pesquisas referentes a IA são lentas, mas já mostraram resultados positivos de como dispositivos e softwares podem interpretar e sintetizar a voz ou os movimentos humanos, por exemplo. Mas, ainda falta muito para que as máquinas atinjam um estado próximo do que seria a inteligência humana.

Neste trabalho a Inteligência Artificial será aplicada nos NPCs, *Non Playable Characters* (parte da Lógica do jogo a ser desenvolvido) conceitos que serão vistos mais à frente dentro do próximo capítulo que abordará sobre as tecnologias que serão utilizadas.

## Tecnologias

Anteriormente foi explicado sobre Lógica e Inteligência e para que elas se apliquem sente trabalho, é preciso falar um pouco sobre as tecnologias que estarão envolvidas nesse processo.

O primeiro item a ser abordado será a linguagem de programação que será utilizada para auxiliar no desenvolvimento posterior do framework e do protótipo do jogo que é Java.

### Java

A linguagem de programação Java™ foi criada na década de 90 por James Gosling na Sun Microsystems e, conforme o próprio Gosling (2013, p. 1) pode-se definir como “[...] uma linguagem de programação de propósito geral, concorrente, baseada em classes e orientada a objetos. ”.

Java possui uma ligação direta com C e C++ que são outras duas linguagens de programação, entretanto possui outra forma de organização e diferentes aspectos, na qual se optou por omitir algumas das ideias destas linguagens e incluir ideias de outras, além disso, foi projetada para ser simples e facilitar a aprendizagem dos desenvolvedores, desta forma programadores que conhecem as linguagens citadas possuem uma maior facilidade no aprendizado da Java.

A orientação a objetos está em quase todas as formas de representação na linguagem, com exceção dos tipos de dados primitivos, tudo são objetos sendo que até mesmo os tipos de dados primitivos podem ser encapsulados em objetos quando se julgar necessário. Estes objetos são construídos através de classes, que possuem suas características e ações (métodos).

É uma linguagem de alto nível, ou seja, onde detalhes das representações das maquinas não estão disponíveis através dela, entre estes detalhes considera-se o gerenciamento de memória, execução de pilhas, threads, etc.

Possui importantes características que a diferencia de seus concorrentes dentro do mercado, dentre elas:

* **JVM (Java Virtual Machine):** Este recurso é o responsável por permitir que a linguagem Java possa ser executada em diversos sistemas operacionais independentemente do *hardware*, ela não conhece a forma como o Java trabalha, sua função é retirar de arquivos *“.class*” instruções e traduzi-las para linguagem de máquina, gerenciar a memória, corrigir possíveis problemas durante a execução, executar *threads*, entre outros. Com isso o desenvolvedor limita-se a desenvolver as funções sem se preocupar com as diferentes formas que cada sistema trabalha.
* **Multiprocessamento:** É capaz de executar um ou mais processos simultaneamente.
* **Independência de plataforma:** Com o uso da JVM o mesmo código pode ser utilizado independente do S.O, desde que este tenha uma JVM especifica para seu sistema.

Como dito anteriormente o JAVA irá ser a linguagem de programação que dará a base para o desenvolvimento do framework e do protótipo, e será aplicada para a plataforma *Android*, próximo item a ser conceituado.

### Android

A palavra *Android* possui mais de uma definição, a que será utilizada como conceito no decorrer da pesquisa é: o *Android* é uma plataforma para tecnologia móvel completa, ou seja, é um dispositivo móvel que envolve um pacote de programas, um sistema operacional que pode ser definido como “[...]quem controla todos os recursos do computador e fornece a base sobre a qual os programas podem ser escritos” (TANEMBAUM;WOODHULL, 200, p. 17), middleware que é a camada intermediaria que fica entre o que o usuário visualiza e a fonte de dados e interface do usuário.

A plataforma foi criada em conjunto com a *Open Handset Alliance (OHA)*, que é uma parceria com mais de 40 empresas do ramo de telefonia móvel, consiste em todas as estruturas envolvidas no processo de telefonia móvel, operadores de celular, fabricantes de dispositivos, empresas de semicondutores, *softwares* e comercialização.

O *Android* foi construído com o intuito de ser um sistema operacional de código aberto, onde todos os *softwares* que fossem desenvolvidos para ele teriam acesso ao mesmo privilégio e pudessem tirar total proveito da estrutura do aparelho por exemplo, qualquer aplicação pode ter acesso as funcionalidades do núcleo do dispositivo tais como, ler mensagens, ou realizar ligações, desde que o usuário deseje conceder tais permissões ao aplicativo.

Ele é baseado no sistema operacional *Linux*, mas apesar disto não pode ser classificado como um sistema *Linux*, pois não possui diversos padrões e contém diversas diferenças como por exemplo, a sua interface gráfica.

Nos dias de hoje o *Android*, está em constante evolução e um dos principais fatores que permitem isso é que sua plataforma é *open-source*, provendo assim uma maior facilidade para que os desenvolvedores trabalhem em conjunto, de uma forma mais ágil. Mais à frente na parte da Pesquisa Mercadológica, será explicado o motivo do uso do *Android* como plataforma.

Uma vez conceituado o que é *Android* e que ele foi concebido para dispositivos móveis, será visto no próximo item o conceito de *mobile*.

### Mobile

“*Mobile*” é uma tecnologia de dispositivos móveis, que são dispositivos, geralmente, pequenos, com uma bateria recarregável e com pelo menos um tipo de conexão, para esta pesquisa serão abordados o conceito mobile de telefonia, ou seja, os celulares, *smartphones* e *tablets* que se utiliza no dia-a-dia da população.

Com o avanço da internet e das novas formas de tecnologia, não era de se espantar que os celulares fossem atingidos por esse processo de transformação, de onde passaram a além de efetuar ligações e mandar mensagens de texto, os celulares atuais, mais conhecidos como *smartphones*, possuem arquitetura de *hardware* e *software* definidas, onde temos diversos sistemas operacionais, ou plataformas, para celular.

Possibilitam às pessoas, principalmente jovens e executivos, acessarem suas informações pessoais e a internet com muito mais velocidade e em aparelhos menores, como se fossem computadores de bolso, dentre as plataformas se destacam: Symbian da Nokia, BlackBerryOS da BlackBerry, Windows Phone da Microsoft, Android da Google e iOS da Apple.

Como informado anteriormente, mais à frente será explicada a razão da escolha do *Android* como plataforma. Agora para que se dê o desenvolvimento do *framework* e antes de falar propriamente dele, será abordado sobre *game engine*, o próximo item da lista.

### Game Engine

*Game engine* ou motor gráfico é um conjunto de ferramentas composta por pacotes de textura, sons, cenários, física, inteligência artificial, interface 2D/3D, entre outros componentes também conhecidos como frameworks que auxiliam no desenvolvimento de jogos para diversas plataformas, sejam elas, consoles (PS4, XOne, Wii), PC, *mobile* (celulares, *smartphones, tablets*).

As plataformas mais famosas na atualidade são *Unity* e *Unreal Engine*, porém neste trabalho a *engine* que será a base para o framework e protótipo futuro será a *Cocos2D*, que será explicada no próximo item.

### Cocos2D

Como informado anteriormente, Cocos2D é a *engine* que será a base para o desenvolvimento futuro do framework e do protótipo e, trata-se de uma família de *engines* de desenvolvimento de jogos e aplicativos de código aberto multi-plataforma.

A família Cocos2D consiste dos:

* Cocos2D-X – *Engine* de desenvolvimento de jogos e aplicativos de código aberto multi-plataforma escrita em C++ / JavaScript / Lua. É utilizada para construir jogos, aplicativos e outros programas interativos. Utiliza-se das Linguagens C++, JavaScript ou Lua, e mesmo que um projeto seja desenvolvido em JavaScript ou Lua o próprio Ccos2D-X traduz para C++, pois ao compilar um código nele, já é disponibiliza a base para as outras linguagens;
* Cocos2D-XNA - *Engine* de desenvolvimento de jogos e aplicativos *open-source* multi-plataforma 2D/3D desenvolvida especificamente para XNA (Ambiente de desenvolvimento próprio para plataformas baseadas em Windows / Windows Phone) em C#. Suporta o ambiente MonoGame (Para Windows e Windows Phone).
* Cocos2D-ObjC também conhecida como Cocos2D-iPhone - *Engine* de desenvolvimento de jogos e aplicativos open-source multi-plataforma desenvolvida em Xcode e Objective-C para iOS.
* Cocos2D (Phyton) - *Engine* de desenvolvimento de jogos e aplicativos open-source multi-plataforma, é o “pai” das outras *engines* só que desenvolvida em Python.

O Cocos2D utilizado para referências futuras é a biblioteca escrita para JAVA que deriva do original desenvolvido em Phyton, ele parte de uma classe de desenvolvimento base chamada Diretor (cocos2d::Director) e representa os conjuntos de elementos na tela através de um Cenário (cocos2d::Scene). É responsável pelo direcionamento do jogo, realiza a troca de cenários, dita o ritmo do jogo em si.

Além do diretor e do cenário nós temos outros elementos que são utilizados no decorrer do desenvolvimento do jogo, que são:

* Nós (cocos2d::Node);
* *Sprites (cocos2d::Sprite);*
* Ações (cocos2d::Action);
* Camadas (cocos2d::Layer).

Utiliza-se da estrutura de árvore para representar seus elementos. Nessa estrutura cada elemento é chamado de Nó, onde o Nó que estiver no topo da estrutura é a Raiz e os que se encontram na parte mais inferior quando não possuem Nós Filhos são as Folhas. Essa estrutura é também conhecida como grafo da cena e a ordem dos elementos na tela estão diretamente relacionados com essa estrutura.

Além disso, possui também as ações, que são objetos capazes de gerar transformações em nós específicos. Alguns exemplos de ações são:

* Mover (cocos2d::MoveTo e cocos2d::MoveBy);
* Rotacionar (cocos2d::RotateTo e cocos2d::RotateBy);
* Modificar a escala, ou seja, aumentar ou diminuir o tamanho (cocos2d::ScaleTo e Cocos2d::ScaleBy);
* Animar a visualização (cocos2d::FadeIn e cocos2d::FadeOut);
* Modificar a cor (cocos2d::TintTo e cocos2d::TintBy);
* Entre outras.

Todas as ações são realizadas em determinados intervalos de tempo, simplificando o trabalho de programá-las.

Para as ações terminadas com *By* as transformações realizadas são baseadas no estado atual do nó, já as terminadas com *To* o estado final é o estado desejado pelo programados, ou seja, sem muitas complicações para o cálculo referente ao estado atual, uma vez que informando essas duas ações o próprio Cocos2D realiza essas transformações.

Sendo assim, as ações são bastante completas, sendo possível executar diversas ações ao mesmo tempo ou de forma única, uma após a outra, e, para tanto, utiliza-se os comandos *Spawn* (cocos2d::Spawn) e *Sequence* (cocos2d::sequence) quando queremos conduzir de forma única ou agrupar as ações respectivamente.

Além disso, contém também como criar ações reversas e repetição. Para a primeira utilizamos o comando *reverse()* e para a segunda utilizamos de duas formas distintas, sendo elas:

* Para repetir um determinado número de vezes (cocos2d::Repeat);
* Para repetir indeterminadamente (cocos2d::RepeatForever) .

Assim dispõe que o Cocos2D é a game engine que auxiliará na construção do protótipo do jogo e na construção do framework.

### Framework

O *framework* é considerado uma estrutura, que está interligada em outro *software* oferecendo recursos para que esse apresente melhorias, sempre que conectado auxilia como: programa de suporte, biblioteca de código, linguagem de script entre outros recursos que auxiliam o desenvolvimento de um projeto.

Algumas características precisam ser consideradas para termos um *framework*: Ser reutilizável, facilitar o desenvolvimento de sistemas, ser documentável, ser completo naquilo em que se propõe, ser eficiente e eficaz. Sua utilização proporciona várias vantagens, uma delas, e talvez a principal, é a redução de custo, uma vez que reduz o tempo estimado do projeto.

Como visto, o propósito do *framework* é de facilitar no desenvolvimento de *softwares*, elevando o nível de detalhamento de programadores em seus projetos e partindo dessa definição, este trabalho propõe o desenvolvimento de um *framework* que irá utilizar o MPD para tornar mais realista as escolhas a serem feitas num jogo de corrida, dado que atualmente as empresas buscam cada vez mais a realidade ou o que mais possa se aproximar dela em termos de jogos, que serão destacadas no próximo tópico.

Para exemplificar melhor, será utilizada aqui uma peça fundamental para um carro de corrida, o pneu, sabe-se que ele possui vários tipos dependendo da pista em que se dará a corrida, então, partindo dessas informações será necessário pelo menos três especialistas para dizer qual tipo de pneu é bom em qual tipo de pista, e aplicando os métodos de Maximização e Minimalização da LPA ao final será dado uma porcentagem para cada pneu, assim o jogador irá poder escolher de maneira mais realista o pneu que melhor se enquadrar, cabendo ao jogador e somente a ele a decisão ou não auxiliada pela indicação.

## Jogos

Antes de falar propriamente de jogos digitais, é necessário dizer que esse conceito advém da teoria dos jogos, a qual, interligada com a teoria matemática, foi criada para estruturar fenômenos que, quando observados, percebem-se dois ou mais agentes de decisão interagindo entre si. Essa linguagem é fornecida para ser feita uma descrição de processos de decisão conscientes e com o objetivo de trazer e envolver mais do que uma pessoa, podendo ser aplicada em vários outros assuntos, tais como eleições, leilões, balança de poder, evolução genética, entre outros.

Poucas pessoas acreditam que a teoria dos jogos em algum determinado tempo irá formar um alicerce de conhecimento técnico estrito de decisões que serão feitas para tentar enxergar um modo de a economia funcionar. Apesar de seu grande desenvolvimento a teoria ainda não chegou nesse patamar e, hoje, tem o seu foco voltado para a matemática pura e, nas aplicações, algumas vezes utilizada como ferramenta ou alegoria que auxiliam no aprendizado de sistemas com maiores níveis de dificuldade.

Conforme explicado anteriormente, a teoria dos jogos juntamente com a teoria matemática tem sua finalidade na escolha de decisões sobre a condição de conflito, e a partir dessas informações tem-se o conceito de jogos que será utilizado para falar de jogos.

Para tanto, é preciso definir o sistema básico de um jogo que é o número de jogadores que dele participam, onde cada jogador possui vários conjuntos de estratégias e quando um jogador decide sua estratégia, tem-se uma situação ou perfil em um espaço repleto de situações possíveis. De acordo com o andar do jogo o jogador cria interesse ou preferências para cada situação. Nos termos matemáticos, cada jogador absorve uma função que será atribuída a um número real a cada situação.

### Jogos Digitais

Com a definição de jogo esclarecida, este tópico irá falar da parte digital do jogo, que nada mais é do que os conceitos de jogos aplicados em dispositivos computacionais.

Para tanto, um jogo é dividido em três etapas: enredo ou trama, motor (ou *game engine*) e interface interativa (ou interface gráfica). Os objetivos do jogo são definidos através do enredo, tendo a opinião de vários especialistas, tais como, escritores, psicólogos, historiadores, pesquisadores, entre outros.

O motor faz a interação entre o jogador e a sua interface. Na colocação de um motor é envolvido diversos conceitos de várias áreas da computação, tais como, computação gráfica, redes de computadores, engenharia de *software*, Inteligência Artificial, entre outras. A interatividade da interface apresenta um estado decorrente do jogo e a viabiliza a interação entre o jogo e o jogador. A sua implementação envolve vários tipos de aspectos artísticos, cognitivos e técnicos.

As características de uma interface de jogo são o atrativo principal, a interatividade com a trama e a jogabilidade desenvolvida em alto nível, no qual representa a capacidade do jogador se adaptar fácil e rapidamente ao ambiente do jogo e acionar alguns recursos para efetuar uma nova jogada.

Apesar de diversas características de um jogo não serem notadas, é possível obedecer alguns critérios gerais para classificar a sua interface: Visão do jogador no ambiente do jogo, o próprio ambiente do jogo, projeção dos gráficos, entre outros.

O sucesso de um jogo nasce de uma combinação precisa de seus componentes, ou seja, o aspecto desafiador não pode ficar só preso na interface do jogo, pois não é a total garantia de que o jogador irá ficar totalmente estimulado, bem como um enredo tem que ser desafiador para seu jogador e ao mesmo tempo causar divertimento, além da criação de um enredo, que é planejada em cima de orientações que envolvem roteiros parecidos com os de teatro e cinema, assim, esses são os fundamentos para o sucesso de um jogo.

Conforme Adams (2009) o processo da criação, do *design*, de jogos tem como base imaginar um jogo, definir como o jogo funcionará, descrever os elementos que fazem parte do jogo (conceituação, funcionalidades, arte e outros) e passar essas informações para quem irá construí-lo.

Para o desenvolvimento do jogo, em geral, é preferível a utilização de metodologias de iterações ágeis. Essa metodologia de desenvolvimento depende do *feedback* e refinamento das iterações do jogo, e segundo Bates (2004), este método é efetivo pois a maioria dos projetos não se iniciam com uma definição clara dos requisitos.

Anteriormente foi comentado sobre a Inteligência Artificial dos Personagens não jogáveis, os NPCs, eles são de grande importância para o projeto, uma vez que serão eles que serão os competidores no protótipo do jogo a ser desenvolvido.

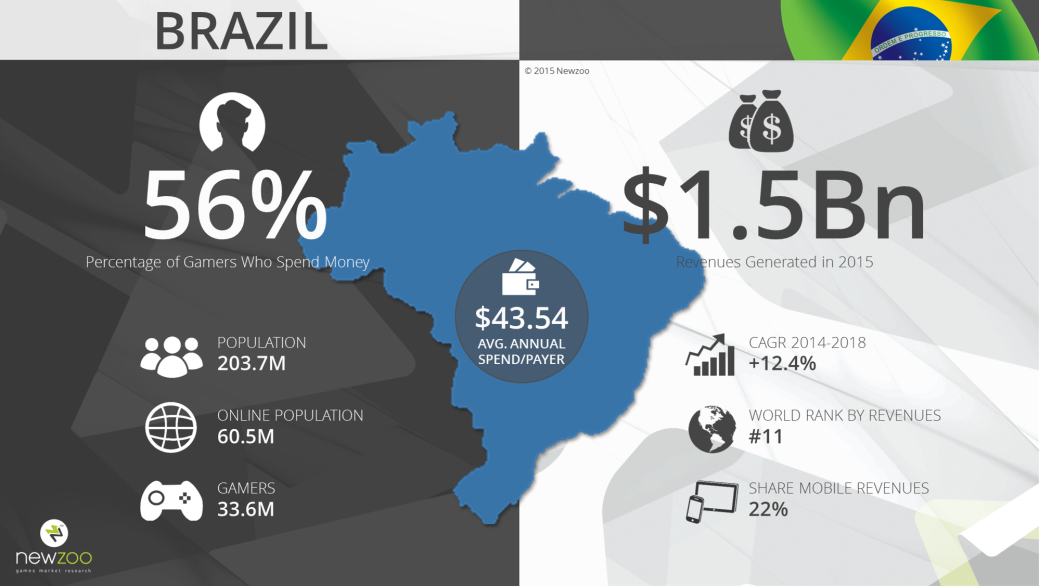
Os jogos são divididos em gêneros, nos quais os mais acessados no Brasil são os de esporte (futebol), ação/aventura, tiro, casuais e corrida. E partindo dessas definições o tipo de jogo abordado no projeto será de corrida. No próximo item, será explicitada a pesquisa mercadológica efetuada para este projeto.

### Pesquisa Mercadológica

Para que este projeto se tornasse possível, alguns itens foram observados antes, como levantamento para esta pesquisa, como o desenvolvimento em plataforma *Android* por exemplo.

Segundo pesquisa realizada no site NewZoo ao final de 2015 o Brasil obteve um gasto de 1.5 Bilhão de dólares no mercado de games, o que dá mais ou menos 58.5 Bilhões de reais, sabendo-se que o dólar fechou o ano de 2015 em R$ 3,90, sendo o 11º país no mundo e o primeiro da América Latina em consumo de games, com uma marca de 203.7 Milhões de habitantes, com cerca de 60.5 Milhões pessoas online, sendo 33.6 Milhões jogadores, ou seja, 56% da população online consomem jogos digitais, é uma fatia um tanto grande, levando-se em conta que há um crescimento estimado entre 2014 a 208 de 12,4%, em resumo, um mercado que mesmo em meio a uma crise, continua em expansão.

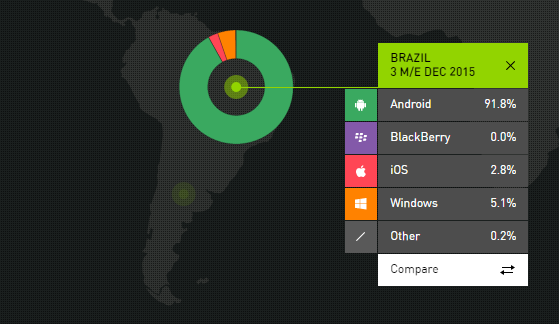
Figura 3 - Infográfico Consumo Brasileiro de Jogos



(Fonte: NewZoo)

Não obstante, no âmbito dos *mobiles*, o campeão de 2015 foi o *Android*, que fechou o ano com uma quantidade de 91,8% do mercado, como mostra a imagem abaixo, retirada do site Kantar World Panel:

Figura 4 - Plataforma mobile mais utilizada



(Fonte: Kantar World Panel)

Conforme nota de lançamento atual da própria Google, os jogos e aplicativos mais baixados seguem na imagem abaixo:

Figura 5 - Mais baixados na PlayStore 2016



(Fonte: Google Playstore)

Daqui percebe-se que os jogos de corrida para mobile já não estão mais tanto assim em destaque, e por conta disso, este projeto irá tratar dos jogos de corrida, e uma vez que for aplicado o framework com a Lógica Paraconsistente não será só mais um jogo de corrida simples e sim estratégia/corrida, misturas essas que atualmente vem ganhando popularidade, basta ver os jogos acima “Clash Royale” e “Pokémon Shuffle”, que misturam estratégia em sua composição.

# **DESENVOLVIMENTO**

O desenvolvimento deste projeto com a criação do framework e prototipação do jogo não será abordado no estado atual da pesquisa.

# **CONSIDERAÇÕES FINAIS**

No capítulo 2.1 foi visto a importância da Lógica para este projeto e como se dará o cerne do projeto em relação ao Método Paraconsistente de Decisão, como principal ingrediente na composição do framework.

No capítulo 2.2 elucidou-se o que é Inteligência e que ela será parte essencial do protótipo, uma vez que será a inteligência artificial dos competidores NPCs.

Já no capítulo 2.3 quais as tecnologias que serão utilizadas para que se deem o desenvolvimento tanto do framework como do protótipo do jogo, alvos deste trabalho.

E por fim, verificou-se o conceito de jogo, trazendo para a questão dos jogos digitais e da razão por trás de tudo isso pela pesquisa mercadológica efetuada.

Até aqui, foram levantadas as bases deste projeto, e com isso espera-se alcançar a continuidade do mesmo pelo desenvolvimento final tanto do framework como do protótipo.

# REFERÊNCIAS

ABE, Jair Minoro et al. **Lógica Paraconsistente Anotada Evidencial Et**. São Paulo: Comunicar, 2011.

ADAMS, Ernest. **Fundamentals of Game Design**. 2. ed. California: New Riders, 2010.

AVRON, Barr; FEIGENBAUM, Edward A.. ***The Handbook of Intelligence* - Vol I**. Willian Kaufmann, Inc., California. 1981.

BATES, Bob. **Game Design**. 2. ed. Boston: Thomson Course Technology, 2004.

COCOS2D. **Cocos2D**. Disponível em <http://cocos2d.org/#home>. Acesso em: 06 abr. 2015.

COCOS2D. **Cocos2D-X**. Disponível em <https://github.com/cocos2d/cocos2d-x>. Acesso em: 06 abr. 2015.

COPI, Irving Marmer. **Introdução à Lógica.** São Paulo. Mestre Jou, 1977.

DA COSTA, Newton C. A. et al. **Lógica Paraconsistente Aplicada**. 5. ed. São Paulo: Atlas, 1999.

DE CARVALHO, Fábio Romeu; ABE, Jair Minoro. **Tomadas de decisão com ferramentas da Lógica Paraconsistente Anotada**. São Paulo: Blucher, 2011.

DOS SANTOS, Acácia A. Angeli; NORONHA, Ana Paulo Porto; SISTO, Fermino Fernandes. **Teste de Inteligência R1-Forma B G36: Evidência de validade convergente.** Itatiba: Universidade São Francisco, 2005. Disponível em:<http://www.scielo.br/scielo.php?pid=S1413-294X2005000200005&script=sci\_abstract&tlng=pt>. Acesso em: 23 abr. 2016.

E SILVA, Ricardo Pereira. **Suporte ao desenvolvimento e uso de frameworks e componentes.** Tese para a obtenção do grau de Doutor em Ciências da Computação. Universidade Federal do Rio Grande do Sul, UFRGS. Porto Alegre, 2000. Disponível em: < http://www.inf.ufsc.br/~ricardo/download/tese.pdf >. Acesso em: 06 abr. 2015.

GOSLING, James. et. al. **The Java™ Language Specification Java SE 7 Edition**.Califórnia. 2013.

KANTAR WORLDPANEL COMTECH. **Smartphone OS sales market share.** Disponível em: <http://www.kantarworldpanel.com/global/smartphone-os-market-share/ >. Acesso em: 14 mai. 2016.

KLEINA, Nilton. **O que é engine ou motor gráfico?**. 2011. Disponível em <http://www.tecmundo.com.br/video-game/9263-o-que-e-engine-ou-motor-grafico-.htm>. Acesso em: 23 abr. 2016.

LINDHOLM, Tim. et. al. **The Java® Virtual Machine Specification Java SE 7 Edition.** Califórnia. 2011.

NEWZOO. **Newzoo Summer Series #13:** Brazilian Games Market.Disponível em: <https://newzoo.com/insights/infographics/newzoo-summer-series-13-brazilian-games-market/>. Acesso em: 14 mai. 2016.

PEREIRA, Lúcio Camilo Oliva; SILVA, Michel Lourenço. **Android para desenvolvedores**. Rio de Janeiro: Brasport, 2009.

TANEMBAUM, Andrew S. WOODHULL, Albert S. **Sistemas Operacionais:** projeto e implantação. Tradução Edson Furmankiewicz. 2. São Paulo: Bookman, 2000.

VINOSKI, Steve. **Where is Middleware?IEEE Internet Compuiting**. Março/Abril. 2002. p. 83.