

Simulador de manutenção

Jogo interativo para complementar o ensino de gestão de manutenção

Vitor Nogueira de Abreu
Universidade federal de Itajubá
Campus Itabira - Itabira MG
v100abreu@gmail.com

Resumo— Este documento apresenta a estrutura de um “jogo educativo virtual” criado para complementar o ensino de disciplinas de manutenção. Através dele o usuário se depara com problemas presentes no ambiente de trabalho da área de manutenção, enquanto recebe informações relevantes sobre eles de forma a encontrar melhores soluções para gestão empresarial.

Palavras-Chave—Jogos Educativos; Manutenção; Gestão Empresarial.

Abstract— This document presents the structure of a “virtual educational game” made to supplement the teaching of maintenance classes. Through it, the user faces the problems inside the work environment of the maintenance field, while receiving relevant information about them in a way to find better solutions for the Business Management.

Keywords—Educative games; Maintenance; Business Management.

I. INTRODUÇÃO

É preciso entender o “Jogo Virtual” como uma modalidade atual, uma ferramenta educativa, e o “jogo” propriamente dito, como um elemento fundamental da existência humana. Segundo [1]: “O jogo é uma atividade ou ocupação voluntária, exercida dentro de certos e determinados limites de tempo e de espaço, segundo regras livremente consentidas, mas absolutamente obrigatórias, dotado de um fim em si mesmo, acompanhado de um sentimento de tensão e de alegria e de uma consciência de ser diferente da ‘vida cotidiana’”. Além de jogo, a palavra virtual, segundo [2], significa em filosofia algo que se opõe a atualidade, por exemplo a árvore está virtualmente contida no grão, é uma potencialidade do grão, assim como um programa virtual simula potencialmente a realidade de uma empresa.

Jogos Virtuais que focam na temática da gestão e gerenciamento são jogados atualmente por diversas pessoas,

mesmo fora das áreas técnicas, são exemplos de jogos de gestão para o público comum: SimCity™; Production Line™; Cities: Skylines™. Entretanto, nesses jogos os objetos que você adquire ou constrói não se danificam, portanto não necessitam de manutenção.

Dada a importância da manutenção no cotidiano do Engenheiro Eletricista, esse trabalho é proposto para atender à aulas relacionadas a gestão de manutenção, para trazer uma forma mais dinâmica de trabalhar pontos como: gerência da manutenção., formas de manutenção, análise do valor, fator humano na manutenção e elaboração de um plano de manutenção.

Como todo Jogo, é esperado um objetivo, e um processo/caminho para realizá-lo; no simulador de manutenção que será tratado aqui como Simulador, o objetivo é ter o menor gasto possível na empresa, bem como valores próximos em relação ao gasto/mês, através do uso consciente dos recursos (humanos e materiais). O jogador gerencia a manutenção das peças que a empresa possui, conforme elas envelhecem ou perdem qualidade. O jogador deve decidir quando comprar ou renovar cada peça da maneira que seja mais adequada.

II. CONCEITOS DO JOGO

A. Peças

O jogador, aqui denominado “usuário” deve comprar o material necessário para que a empresa funcione, esse valor é específico para cada Peça, um exemplo de sistema com peças seria um carro, que precisa de quatro rodas para andar, mas normalmente possui a quinta roda (step), assim sendo o usuário escolhe entre ter quatro ou cinco rodas de acordo com o custo da mesma. O fato de possuir apenas quatro rodas traz o risco de parada do veículo caso uma delas quebre. De maneira análoga, caso a empresa tenha uma produção que dependa de

dez peças e possui somente essas, na falha de uma haverá parada da produção e impossibilidade de funcionamento da empresa até que haja compra de uma nova.

Cada peça necessita de um número de horas de manutenção para manter sua qualidade, que deve ser equiparado pelo número de horas entregues pelos trabalhadores.

B. Calendário

O jogo possui um calendário onde ocorrem eventos esporádicos, devendo o usuário antecipar suas ações com base nos próximos eventos. Esses eventos podem ser tanto eventos climáticos, que interfiram na qualidade das peças como eventos de mercado que afetam o preço das peças, como exemplo período de alta umidade ou de calor excessivo podem danificar peças, bem como período de baixas temperaturas podem melhorar a vida útil das peças. Eles devem possuir uma breve descrição para instruir o usuário sobre seus efeitos.

C. Trabalhadores

O usuário escolhe a rotina dos trabalhadores, considerando férias e fins de semana, bem como o valor recebido por eles. A consequência das escolhas é a permanência dos funcionários no corpo de trabalho da empresa. Fatores como saúde e salário do trabalhador influenciam a mantê-lo na empresa ou perdê-lo através de demissão.

O número de trabalhadores define quantas horas de manutenção são empregadas nas Peças, cada trabalhador entrega um número de horas de acordo com as férias e os finais de semana, gerando uma quantidade total de hora-homem que fornece às peças a fim de manter a qualidade em todas.

D. Estoque e Loja (não implementado)

Na loja, o usuário deve escolher qual Peça ele prefere ter na empresa, podendo escolher uma de maior qualidade e mais cara, ou de menor qualidade e mais barata. O usuário deve determinar quando trocar as Peças e quantas manter em estoque para que eventuais quebras não gerem parada total na produção. Enquanto que no estoque o espaço para Peças sobressalentes é limitado, devendo o usuário ser capaz de definir quais são as mais essenciais, por exemplo se seu estoque possui 100 unidades de armazenamento e um determinado motor possui 80 unidades, ele ocupa praticamente todo o espaço do estoque, e só deve ser mantido se extremamente necessário.

E. Métricas

O jogo utiliza duas métricas para avaliar o desempenho do usuário comuns no ambiente empresarial [3], Gasto total e Fator de gasto, a primeira soma todas as compras que o usuário fez durante o decorrer do jogo assim como os salários dos funcionários definidos pelo próprio usuário. Já a segunda

é bem semelhante a outros fatores de demanda e pode ser determinado de acordo com a seguinte equação:

$$F_{gasto} = \frac{Gasto_{max}}{Gasto_{medio}} \quad (1)$$

Onde $Gasto_{max}$ é o maior valor mensal registrado e $Gasto_{medio}$ é o valor médio gasto por mês e F_{gasto} o fator de gasto. O ideal, é que o usuário mantenha a equação (1) sempre o mais próximo de 1, garantindo que a empresa não precise exercer grandes gastos em um mês e tenha um gasto baixo em outro.

III. O JOGO ELETRÔNICO

O jogo, pelos seus conceitos em si, poderia ser desenvolvido de várias formas. Com o uso de cartas, dados e calculadora é possível montar uma versão rudimentar do mesmo. Porém foi escolhido o método de Simulador eletrônico, tanto para efeito de interatividade, onde o usuário apenas aperta alguns botões ao invés de jogar dados, como para efeito educativo, onde informações são exibidas durante o acontecimento dos eventos, melhorando o aprendizado.

O Simulador foi desenvolvido em MATLAB® com programação orientada a objetos e utilização da ferramenta GUIDE.

Para o desenvolvimento do programa foi utilizada a própria linguagem de programação do software, o código possui os objetos (*classes*) *parteQuebravel* que descreve atributos das Peças como nome, qualidade, idade, custo, etc; e *trabalhador* com atributos como nome, saúde, salário, limite mínimo de salário. Além dos objetos, o código possui as funções *geracalendario* e *atzcalendario*, responsáveis por criar um calendário com eventos aleatórios em determinados meses, e a função *chance* que calcula a probabilidade da Peça quebrar de acordo com sua idade. O programa principal chamado *TFGV3* contém as funções relativas aos botões e menus da interface gráfica, apresentada na Figura 1. A interface é composta de três áreas, a das tabelas informativas *Peças*, *Agenda* e *Trabalhadores*; a área de *Descrição*, no canto inferior esquerdo e a área correspondente às métricas e o botão *Próximo Mês*, no canto inferior direito. Sempre que o usuário clicar em qualquer item, detalhes específicos daquele item serão exibidos na área *Descrição*, detalhes esses que ajudarão na aprendizagem.

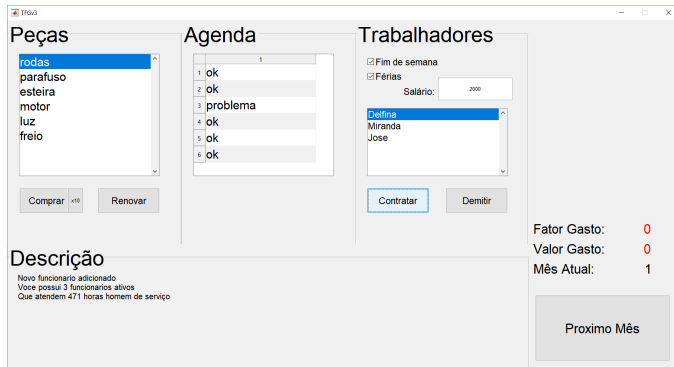


Fig. 1. Tela do Simulador com as áreas tabelas(acima), descrição (inferior esquerdo) e métricas(inferior direito).

IV. JOGANDO

Para iniciar o jogo basta abrir o aplicativo que então virá com as métricas todas zeradas. Em seguida o usuário escolhe os parâmetros para contratar um funcionário, clica em *Contratar* e observa qual a hora-homem entregue por aquele funcionário. A hora-homem de todos os funcionários deve ser maior ou igual a hora-homem necessária para manter as Peças, caso contrário, sua qualidade cairá com o passar dos meses. Em seguida, o usuário confere se estão faltando algumas das Peças essenciais para o funcionamento da empresa, clicando sobre as Peças na tabela *Peças* como observador na Figura 2. Ao clicar, aparecerá a descrição e se a quantidade é suficiente para o andamento da empresa sem paradas. Após todas as Peças verificadas, o usuário deve entrar no campo *Agenda* e checar se no próximo mês haverá algum evento que afete a qualidade da máquina.

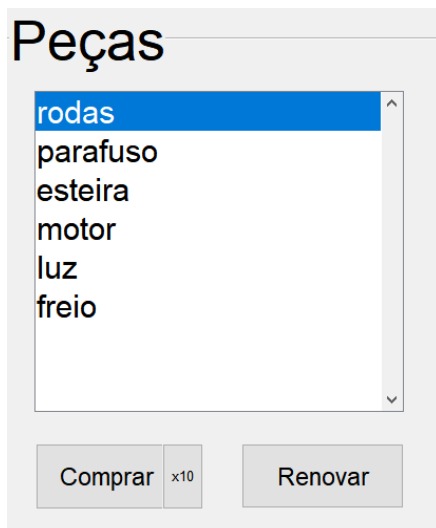


Fig. 2. Peças e informações das mesmas

Ao fim de todas as informações checadas, o usuário clica em próximo mês. Todo código referente às atualizações é

então executado, incrementando a idade dos equipamentos, pagando os funcionários atuais e caso haja evento na agenda, atualizando a qualidade do objeto. É gerado um relatório referente ao mês anterior (Fig. 3) com os dados gerais da empresa, as demissões que ocorreram no período e se houve ou não parada total por falta de Peças essenciais.

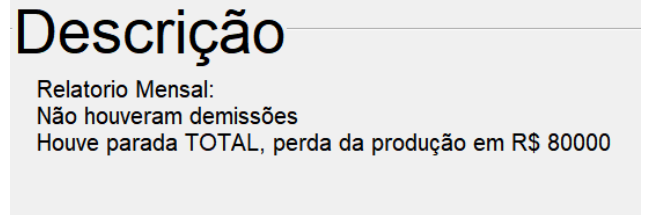


Fig. 3. Exemplo de relatório mensal de atividades.

O ciclo se repete, o usuário deve novamente checar as peças, a agenda, contratar funcionários caso necessário e clicar em próximo mês até que o objetivo proposto pelo criador seja atingido ou fracassado.

V. CRIANDO UMA FASE

O jogo permite que outro jogador, normalmente o professor, seja o criador da fase. Para tanto, ele deve alterar uma planilha que está anexa ao programa, estabelecendo as peças que compõem aquele determinado sistema, quais eventos podem acontecer e como eles afetam a qualidade dos equipamentos.

Para criar as peças, são necessários os seguintes atributos: nome, idade, custo, quantidade, quantidade necessária, qualidade, hora homem de serviço, e os fatores k_1 , k_2 e k_3 . Com exceção de idade, quantidade e qualidade, os atributos não mudam durante o jogo.

Os campos são descritos da seguinte forma: *Nome* é o nome ou abreviação do nome da peça; *Idade* é a idade com que a peça inicia quando se abre o jogo, caso de uma empresa que já funcionava quando o usuário inicia seu trabalho, ou de uma empresa que começou agora mas comprou determinado equipamento usado; *IdadeMax* é a idade máxima que o objeto pode alcançar em teoria, no programa 99% das peças costumam quebrar em 30% da idade máxima[4]; *Custo* é o valor da peça em Reais; *Quantidade* é número de peças presentes na empresa, somando as necessárias e as sobressalentes; *QTDnecessaria* é a quantidade mínima de peças na empresa para que não haja parada total da mesma; *Qualidade* é o estado das peças; *Horahomem* é o número de horas que um trabalhador gasta por mês para manutenção da peça; *fator k_1* é multiplicado a idade máxima, varia de]0 a 1[, normalmente 0,5 e é determinado por condições da empresa que fariam a peça “envelhecer” mais rapidamente ou lentamente; *fator k_2* é dado por fatores que influenciam na qualidade, varia de 0 a 1, normalmente 1, a grandeza é

multiplicada a qualidade; fator k3 é o fator que determina a quebra da peça por outros fatores não previstos, varia entre 0 e 1, no programa é usado 0,03, significando que ha 3% de chance da peça quebrar por eventos imprevisíveis, caso fosse implementado o item Loja, diferentes marcas teriam diferentes constantes k associadas aos seus produtos.

Os campos podem ser editados através do arquivo csv, um exemplo de peças e seus atributos pode ser observado na figura 4.

	nome	idade	idadeMax	custo	quantida...	qtdneces...	qualidade	horaho...	k1	k2	k3
Text	Number	Number	Number	Number	Number	Number	Number	Number	Number	Number	Number
1	nome	idade	idadeMax	custo	quantida...	qtdneces...	qualidade	horaho...	k1	k2	k3
2	rodas	2	100	1	200	180	1	2	0.6	1	0.03
3	parafuso	1	100	2	36	30	1	0	0.6	1	0.05
4	esteira	2	100	200	3	3	1	15	0.6	1	0.03
5	motor	1	100	1000	2	2	1	20	0.6	1	0.03
6	luz	2	100	12	120	100	1	10	0.6	1	0.07
7	freio	5	100	5	40	45	1	1	0.6	1	0.03

Fig. 4. Arquivo parteQuebravelTabela.csv aberto pelo MATLAB.

Outros fatores que são alterados pelo criador, a fim de aumentar o aprendizado do usuário, são os eventos esporádicos, devendo estes serem colocados em planilha também anexa ao programa. Como mostrado na figura 5, esses eventos envolvem o acontecimento em si, com uma breve descrição de como os mesmos afetam a qualidade das peças, a fim de ensinar, e na segunda coluna, o efeito negativo na qualidade dos equipamentos.

Outros elementos podem ser alterados internamente via programação, apesar de serem mais complexos para o criador editar, são passíveis de serem alterados.

	A	B
1	Problema	efeito
2	Nao haverá problemas no mes selecionado	0
3	Tempo quente Durante o periodo haverá problema de sobreaquecimento das pecas e as mesmas podem se danificar	0.05
4	Alta Humidade Durante o periodo as pecas podem sofrer oxidacao	0.1

Fig. 5. Arquivo problemasTabela.csv aberto pelo Google Sheets.

As variáveis e funções internas do jogo mais importantes são descritas a seguir, bem como o método de cálculo ou definição das mesmas adotadas pelo autor a fim de simular possíveis cenários das variáveis. O criador deve adequar esses parâmetros de acordo com o seu próprio cenário de jogo. Parada total (linha código 59), que possui valor 80 mil, o

criador escolhe considerando o valor que a empresa perderia se houvesse uma parada total da mesma no mês; Custo da renovação (linha código 222) é o custo que o usuário paga ao renovar uma peça, foi definido que se ela possui mais tempo que a idade máxima ou qualidade inferior a 30% ela não possui mais valor; Ultrapassagem do limite de quebra (linha código 411), caso seja ultrapassado as peças são reduzidas a uma quantidade aleatória entre 0 e 10% das peças necessárias, é considerado aqui que mais peças estavam em mal estado, porém foram canibalizadas para deixarem as restantes em bom estado; 'Sub atendimento' de horas homem (linha código 416). A qualidade é reduzida em 10% da diferença entre a hora total necessária às partes e a soma das horas entregues pelos trabalhadores normalizada, conforme demonstrado na equação 2:

$$Q_{nova} = Q_{antiga} - 0.1 * \frac{HHN - HHE}{HHN} \quad (2)$$

Onde HHN são as horas necessárias para manutenção das peças, HHE as horas entregues pelos funcionários e Q corresponde a qualidade, caso esse valor ultrapasse 30%, a qualidade é reduzida em 30%,. Qualidade abaixo de 0 (linha código 423), quando houver redução total da qualidade, também há canibalização das peças, para que as restantes possam ter uma qualidade residual de 40%; Condição de parada total (linha código 431), caso haja menos peças do que a quantidade mínima necessária, ou seja a empresa não possui uma peça no estoque e deve portanto parar a produção enquanto aguarda a chegada de uma nova peça; Influência de um evento em equipamentos (linha código 451), aplica a redução da qualidade do equipamento baseado apenas na coluna da tabela csv, podendo o criador alterar as condições de redução da qualidade; por fim, Texto de relatório mensal (linha código 481), é a mensagem do relatório mensal, e pode exibir outras informações que o criador julgue necessário para instruir o usuário no aprendizado da disciplina ou para adicionar outras informações relevantes ao jogo.

O criador também pode alterar os atributos dos objetos (classes) *parteQuebravel.m*, *trabalhador.m* esses atributos garantem que o jogo decida quando as peças quebram ou os funcionários se demitem. Por exemplo a classe trabalhador possui uma saúde inicial e um limite mínimo de saúde, caso seja atingido, o trabalhador se demite, fatores como salário e dias de trabalho influenciam na saúde positiva e negativamente, e através de uma função de atualização da classe que atualiza mensalmente.

No caso do limite para quebrar de *parteQuebravel.m*, é utilizado a função chance, criada a partir de uma metodologia que reproduzisse as curvas de confiabilidade de peças reais [4]. Para tanto, admitiu-se uma regra semelhante a probabilidade do aniversário de dois indivíduos ocorrerem no mesmo dia, ou princípio da casa dos pombos no caso do

trabalho, seria a probabilidade de o dia da falha coincidir com o dia atual. Dependendo do caso, a probabilidade de ocorrerem falhas com 10% da idade máxima é maior que 99%, entretanto em um lote de mil peças se é provável que pelo menos uma dure 100 anos, então essa deve ser considerada a idade máxima, mesmo que as outras durem 10 anos. Na figura 6 como exemplo da função chance, são aplicados os parâmetros 1000 para idade máxima e 120 para o último número exibido, é possível observar a probabilidade da peça quebrar naquele determinado período, devido ao arredondamento do programa o valor chega a 99,94% de chance de quebra.

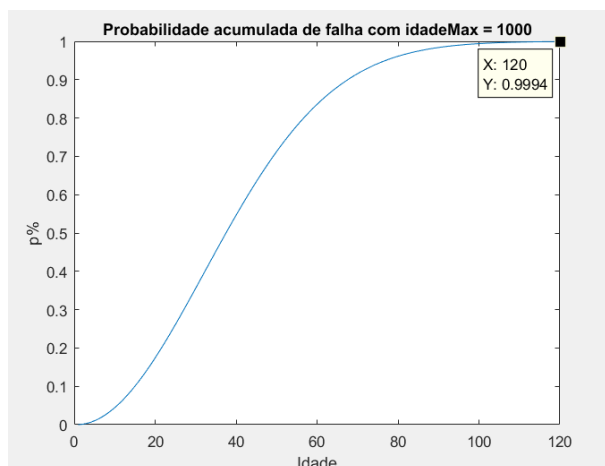


Fig. 6. Função chance.m para idadeMax = 1000, 120 primeiros pontos

Ao derivarmos a função chance que exibe a probabilidade acumulada, obtemos a probabilidade específica para cada idade pode ser observado na figura 7, demonstrando a probabilidade de ocorrer em cada idade. No caso de idadeMax = 1000 a idade onde ocorre mais falhas é em 32.

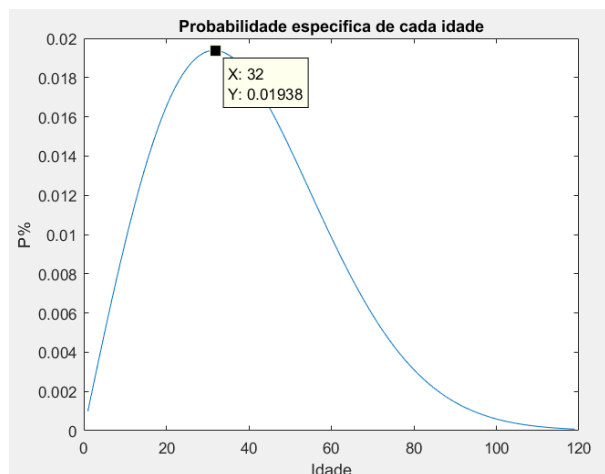


Fig. 7. Probabilidade específica de falhar em cada idade.

O gráfico se aproxima de uma função de Weibull, amplamente utilizada para prever falhas de equipamentos [4], através de uma aproximação para uma curva de Weibull, com

os parâmetros $\alpha = 44$ e $\beta = 2$, obteve-se a curva da figura 8, que ao ser plotada juntamente com a curva criada pela função chance praticamente a sobrepõe, como observado na figura 9.

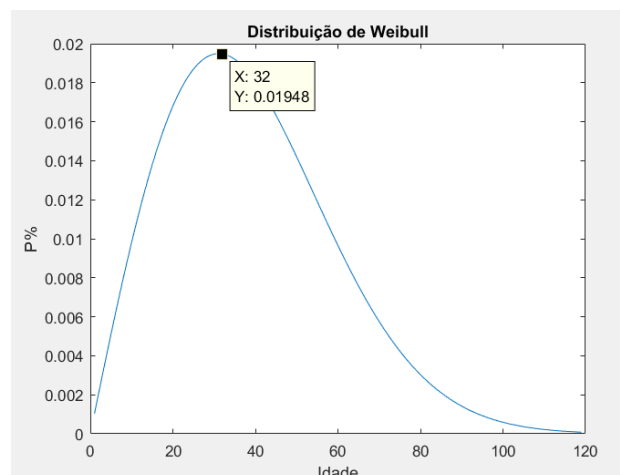


Fig. 8. Curva de Weibull $\alpha = 44$ e $\beta = 2$.

Não é escopo do trabalho, mas com a utilização da curva de Weibull, sabendo seus parâmetros é possível estimar quando as peças irão falhar e por que essas falhas ocorrem, α corresponde ao fator de escala, e é igual ao valor da idade que atinge 63,21% na probabilidade acumulada, enquanto β corresponde ao fator de curva. Quando $\beta < 1$ significa que as peças falham prematuramente, quando $\beta = 1$ as peças não sofrem influência com o tempo e para $\beta > 1$ tem-se o envelhecimento das peças. Como α e idade=44 possuem o valor de 63,2% de probabilidade acumulada e $\beta > 1$, a função chance reproduz a vida de uma peça ou máquina conforme sua probabilidade de falhar por envelhecimento.

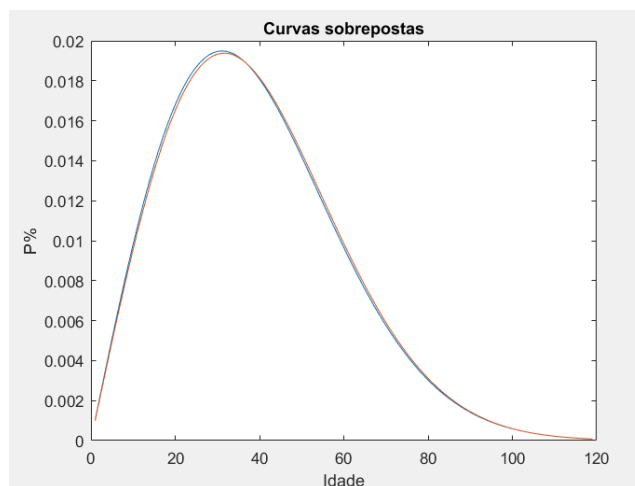


Fig. 9. Curvas das figuras 7 e 8 sobrepostas.

VI. CONCLUSÃO

O ambiente profissional hoje exige pessoas cada vez mais capacitadas e aptas a realizar as tarefas de manutenção, e quanto mais conhecimento adquirido através de forma dinâmica e lúdica, maior e mais rápida é a sua assimilação.

O simulador apresentado insere a capacidade de gamificação da gestão da manutenção, que traz benefícios para o processo bem como a possibilidade de avaliação das atitudes dos funcionários dentro do simulador pela gerência da empresa, fazendo-a optar por um gestor que esteja de acordo com os valores da mesma.

Há o potencial de refletir situações que acontecem em empresas reais como falha de peças ou desgaste dos trabalhadores devido a abusos de descansos oferecidos, em forma de simulação computacional, seja para uso didático de forma a instruir alunos ou novos funcionários à rotina de um engenheiro de manutenção ou para que gestores de empresas usem seus dados reais para alimentar o programa e assim simularem previsões futuras. O trabalho também possui alto nível de edição, podendo o professor ou outro criador de fase alterar os dados de acordo com um cenário específico ou até com os próprios dados da empresa que deseja-se simular.

A partir dessa obra abre-se a perspectiva para novos trabalhos como, inclusão do conceito Loja e Estoque ao programa. Inclusão de dados de uma empresa real, para

simulação de possíveis falhas na mesma e até reconstrução do simulador porém utilizando interface gráficas e elementos visuais, como visão explodida de um sistema e manutenção de seus componentes integrantes.

Os arquivos utilizados para o simulador, incluindo o código fonte e menu de interface (GUIDE) encontram-se disponíveis no GitHub através do link: <https://github.com/v100abreu/jogomanutencao>.

REFERENCIAS

- [1] HUIZINGA, Johan. "Homo ludens. O jogo como elemento da cultura". São Paulo: Perspectiva, 2014.
- [2] LÉVY, Pierre. "Cibercultura". São Paulo: 34 Ltda, 1999.
- [3] OLIVEIRA, Cláudia de; JORVINO, Juliana Jeronymo. "Compras: negociação, estratégia, redução de custos são elementos para agregar em sua empresa?". São Paulo: 2007, disponível em: <http://www.unisaesiano.edu.br/encontro2007/trabalho/aceitos/CC35111892842.pdf>
- [4] MARTINS, Maria P. G.; LEITÃO, Armando L. F.. "Predição de Falhas no Apoio à Decisão na Gestão da Manutenção". Bragança, disponível em: <https://bibliotecadigital.ipb.pt/bitstream/10198/1704/1/Artigo2.pdf>
- [5] KARDOSH, Suely. "Férias: uma necessidade real para todo profissional". Entrevista ao site RH.com.br disponível em: http://www.rh.com.br/Portal/Qualidade_de_Vida/Entrevista/7469/ferias-uma-necessidade-real-para-todo-profissional.html

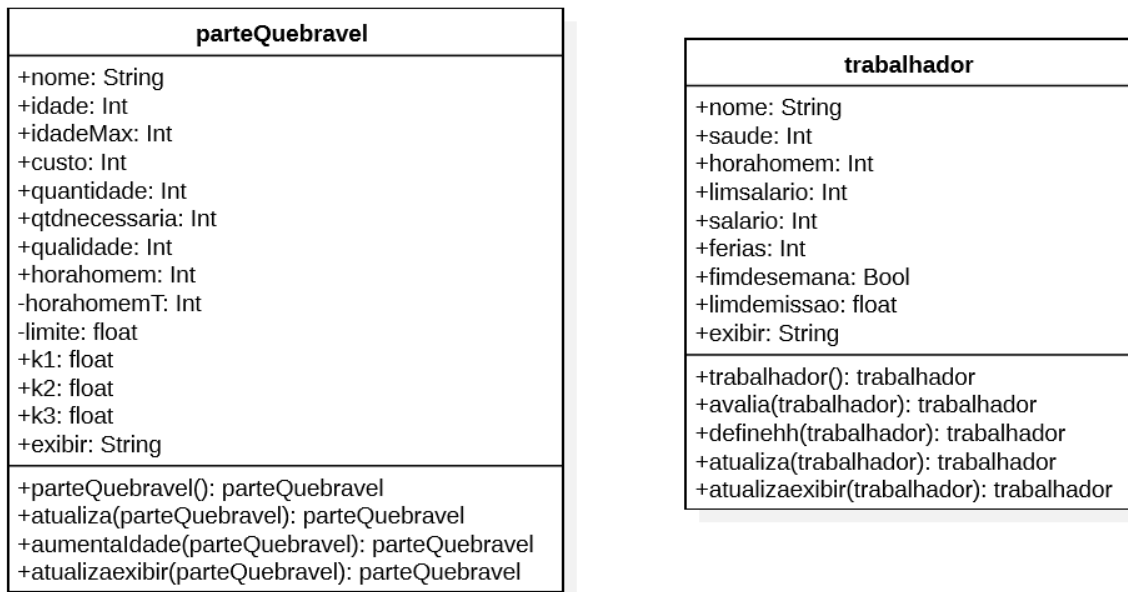


Fig. 10. Diagrama de classe UML das classes parteQuebravel.m e trabalhador.m

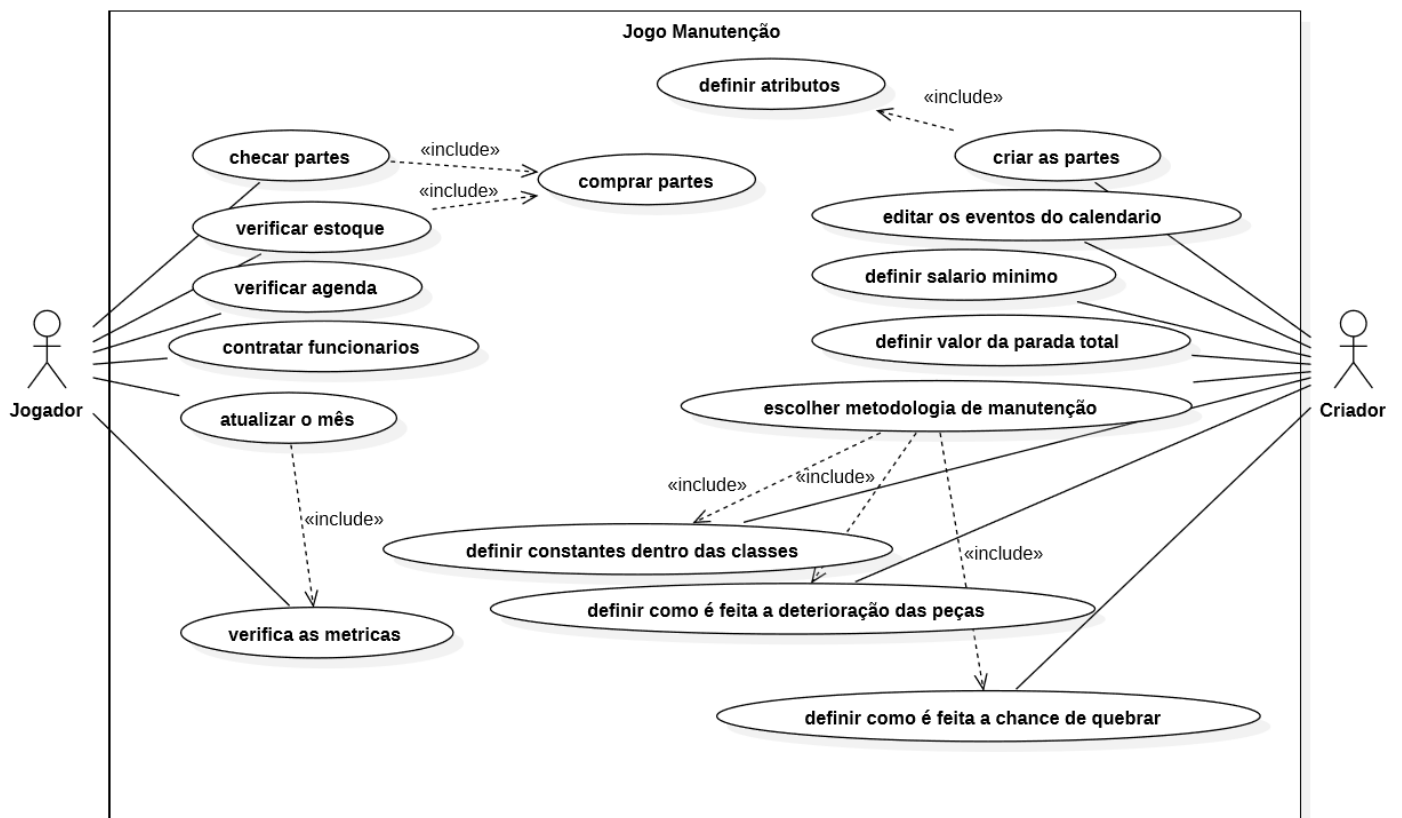


Fig. 11. Diagrama de uso de caso UML para o jogo das duas perspectivas

