



UPIS – UNIÃO PIONEIRA DE INTEGRAÇÃO SOCIAL

Bacharelado em Sistemas de Informação

JOÃO MARCELO DE JESUS MACEDO

MATHEUS REIS DE SOUZA TEIXEIRENSE

MATHEUS SENA VASCONCELOS

YGOR OLIVEIRA GONÇALVES

Autômatos celulares em um ambiente epidemiológico simulado

Relatório Técnico-Científico de Tópicos em Tecnologia da Informação e Comunicação

Brasília-DF
2021

SUMÁRIO

SUMÁRIO	2
RESUMO	4
ABSTRACT	5
INTRODUÇÃO	6
Considerações Gerais	6
Descrição do Problema	6
Justificativa e Importância	6
Objetivo geral	7
Objetivos específico	7
LEVANTAMENTO BIBLIOGRÁFICO E CONCEITOS	8
Modelo de Entidade e Relacionamento	8
Objetos, Atributos e Métodos	8
Padrão de Projetos ou Design Patterns	8
Null Object Design Pattern	9
Python	9
Numpy	10
Pandas	10
Matplotlib	10
Análise Exploratória de Dados	10
Gráfico de Média Móvel	10
DESENVOLVIMENTO	11
Modelo de Entidades e Relacionamentos	11
Core	11
Autômatos	13
Report	14
Atualizações no Modelo de Entidades e Relacionamento	15
Core	15
Autômatos	17
Report	17
Cálculos envolvidos	17
Probabilidade de infecção (PI)	17
Probabilidade de óbito (PO)	18
Simulação	18
Números de Mortes	19
Mortes por idade	19
Mortes por dia	19
Números de infectados	19
	2

Infectados por dia	19
RESULTADOS E DISCUSSÃO	20
TRABALHOS RELACIONADOS	21
TRABALHOS FUTUROS	22
CONSIDERAÇÕES FINAIS	23
REFERÊNCIAS BIBLIOGRÁFICAS	25
LEITURAS RECOMENDADAS	27

RESUMO

Inúmeras doenças já estiveram, e outras ainda estão, presentes no dia-a-dia da humanidade. Durante toda a história é possível identificar diversas epidemias causadas por um surto local, como por exemplo, em 2013 com o surto de Ebola na África Ocidental. Por outro lado, algumas patologias ultrapassam o contágio regional, chegando a disseminar em continentes ou até mesmo em todo o planeta Terra, como é o caso da Peste Negra e o recente Coronavírus. Este estudo busca mostrar através de uma simulação com autômatos celulares um ambiente epidemiológico, onde a população é representada pelos autômatos e suas interações, o meio de contágio. Utilizando dados disponibilizados em pesquisas e pelo Ministério da Saúde do Brasil sobre o Covid-19, a aplicação desenvolvida simula o contágio da doença disponibilizando dados, animações e gráficos sobre a epidemia. Ainda, a aplicação introduz algumas variáveis que podem influenciar na disseminação da doença, como medidas preventivas, qualidade de vida do indivíduo e sua idade. Com isso, é possível inferir que certas doenças possuem um alto grau de disseminação podendo causar um surto global. Além disso, é possível simular diferentes ambientes e, através dos resultados finais do modelo, realizar análises para melhores tomadas de decisões.

Palavras-chave: Autômatos celulares; probabilidade; epidemia; pandemia; simulação; programação.

ABSTRACT

Uncounted diseases have been, and others are still, present in the daily lives of humanity. Throughout history it is possible to identify several epidemics caused by local outbreaks, for example, in 2013 with the Ebola outbreak in West Africa. In contrast, some pathologies overtake regional contagion, spreading across continents or even across the planet, such as the case of Black Death and the recent Coronavírus. This study aims to show through a simulation with cellular automata an epidemiological environment, in which the population is represented by the automata and their interactions, means of contagion. Using data available in surveys and by the Ministry of Health of Brazil about Covid-19, the developed application simulates the contagion of the disease providing data, animations and graphics about the epidemic. In addition, the application introduces some variables that can influence the spread of the disease, such as preventive measures, the individual's quality of life and age. Thus, it is possible to infer certain diseases have a high degree of dissemination potentially causing a global outbreak. In addition, it is possible to simulate different environments and, through the final results of the model, perform analysis for better decision making.

Keywords: Cellular automata; Probability; Epidemic; Pandemic; Simulation; Programming.

INTRODUÇÃO

1. Considerações Gerais

Desde o início do ano de 2020 o mundo vem sofrendo com um grande problema devido a propagação do novo Coronavírus chamado de SARS-CoV 2 que é altamente contagioso. Este vírus causa doença respiratória denominada Covid-19, que vem deixando inúmeros óbitos ao redor do mundo. Curiosamente, no início do surto foi percebido que pessoas com a idade avançada e ou com doenças pré existentes, denominadas como grupo de risco, vinham a ter mais complicações devido à contração do vírus e posteriormente resultando em óbito. Com o passar do tempo, foram notados que pessoas de qualquer idade e sem nenhuma doença também contraíram a doença e por fim vindo a óbito. Portanto, foram impostas medidas protetivas com mais vigor, independentemente de pertencer ao grupo de risco ou não.

Este projeto estuda e simula um ambiente epidemiológico e viabiliza de forma visual a difusão e propagação de uma doença. Baseado nos métodos de prevenção da contração do vírus utilizados atualmente na pandemia do Covid-19 e até mesmo na qualidade de vida de cada ser humano, foram considerados os métodos mais conhecidos e recomendados pelos órgãos de saúde, tais como o uso de máscara, o isolamento social e a própria vacinação.

2. Descrição do Problema

Na pandemia, parte da população mundial não seguiu as recomendações da Organização Mundial da Saúde (OMS) como o isolamento social e outras prevenções necessárias para a contenção do vírus. Consequentemente, os números de infectados e óbitos aumentaram devido à alta negligência. A partir disso, fica o questionamento se grande parte da população mundial adotasse as medidas preventivas severamente ou prematuramente, os resultados finais seriam menos calamitosos.

3. Justificativa e Importância

Tendo em mente a atual pandemia, optou-se pelo desenvolvimento de um software capaz de simular diferentes ambientes epidemiológicos com diferentes

prevenção, a fim de realizar análises ao final com os dados gerados. A importância deste trabalho é demonstrar que medidas preventivas tomadas de formas diferentes e prematuras, os resultados poderiam ser mais satisfatórios.

4. Objetivo geral

O objetivo deste estudo é demonstrar de forma prática a aplicação de autômatos celulares em diferentes ambientes epidemiológicos simulados. Para isso, desenvolver uma aplicação que seja possível configurar os parâmetros da doença, como por exemplo a probabilidade de contágio e óbito. Além disso, que seja capaz de introduzir variáveis, como medidas preventivas, saúde do indivíduo e idade, que impactam diretamente no resultado da simulação.

Ao final da simulação e com os dados gerados, realizar análises e comparações dos ambientes simulados do Covid-19, com e sem as medidas preventivas.

5. Objetivos específico

- I. Desenvolver uma aplicação que simula a interação de indivíduos e a disseminação de uma dada doença.
- II. Configurar a aplicação com dados do Covid-19 e simular o ambiente epidemiológico.
- III. Gerar informações, animações, gráficos e análises sobre o ambiente.
- IV. Introduzir medidas preventivas, comparar os ambientes e realizar análises.

LEVANTAMENTO BIBLIOGRÁFICO E CONCEITOS

1. Modelo de Entidade e Relacionamento

A modelagem de dados é de suma importância para identificar os problemas organizacionais e projetar soluções, uma vez que com ela foi obtido uma visão única, não redundante e sucinta dos dados que serão armazenados. Ela tem como objetivo planejar as etapas de estruturação de qualquer projeto de banco de dados e facilitar manutenções do sistema quando necessário, além de ajudar a compreender o significado destes dados.

Segundo Chen (1976), o Modelo de Entidades e Relacionamentos é a visão de uma dada realidade baseada no relacionamento entre os conceitos desta realidade. Ou seja, é definido em dois grupos sendo eles de Entidades e Relacionamentos, no qual representam um conjunto de objetos da realidade modelada que se deseja manter suas informações no banco de dados e um conjunto de associações entre as entidades, respectivamente.

2. Objetos, Atributos e Métodos

São conceitos básicos de Linguagem de Programação Orientada a Objetos. De acordo com Cáceres (2009), é uma forma de entender e representar sistemas através de estruturas hierárquicas de objetos que se comunicam. Os objetos representam quaisquer utensílios concretos da nossa realidade e cada uma contará com suas próprias características, que são chamadas de Atributos. Esses objetos, além das características, também contam com seus métodos, que são as atividades e as ações executadas pelos próprios objetos podendo utilizar seus atributos.

3. Padrão de Projetos ou Design Patterns

Padrões de Projeto ou design patterns conforme Guru, são soluções típicas para problemas comuns em projetos de software. Cada padrão é como uma planta de construção que pode ser customizada para resolver um problema de projeto recorrente em seu código. Ele não é simplesmente um pedaço de código pronto como as funções e bibliotecas encontradas na internet, mas sim um conceito geral que é aplicado e implementado a fim de solucionar um problema do seu programa.

Para construir softwares orientados a objetos de forma mais flexível e fácil de manter, são usados os padrões de projetos que tornam mais fáceis a reutilização de soluções e arquiteturas, reduzindo a complexidade do projeto e estabelecem um vocabulário comum, facilitando a comunicação, documentação e aprendizado dos sistemas.

Como um bom modelo de solução de problemas, existem características obrigatórias que um padrão de projetos deve ser seguido. E são compostos por quatro elementos:

- Devem possuir um nome que descreva o problema, as soluções e as consequências;
- Relatar de maneira clara o problema a ser resolvido;
- Uma solução concreta, que possa ser aplicado de maneiras diferentes; e
- Relatar quais as consequências para que possa ser analisada a solução aplicada e a compreensão dos benefícios da sua aplicação no projeto.

Os padrões de projeto podem ser divididos em três tipos: O criacional que é o modo como os objetos são criados, compostos e representados. O estrutural que se interessam em como classes e objetos são compostos em estruturas maiores. E o comportamental que são focados em algoritmos e na designação de responsabilidades entre objetos.

4. *Null Object Design Pattern*

Segundo Bgasparotto (2017), *Null Object* é um padrão de projeto do tipo comportamental baseado em herança que cria representações válidas de objetos null em um sistema, ou seja, é um objeto sem valor referenciado ou com um comportamento nulo definido. Por consequência, são geradas várias verificações para garantir que essas referências não sejam nulas antes de chamar qualquer método, assim evitando que exceções sejam lançadas e comportamentos inesperados sejam apresentados.

5. Python

Python é uma linguagem de programação de código aberto disponibilizada sob a licença da PSF License Agreement. Ela é uma linguagem multiplataforma, ou

seja, consegue desenvolver aplicações para desktops, web, dispositivos móveis, inteligência artificial e até mesmo para os games. Segundo o relatório de classificação da RedMonk (2021), o Python é a segunda linguagem de programação mais popular do mundo. Sua logo ficou conhecida por duas cobras entrelaçadas, na qual não tem nenhuma ligação com o nome. Existem alguns motivos que fazem o Python ser totalmente valorizada no mercado de trabalho, nos quais seriam:

- Facilidade de aprender a linguagem
- Multiplataforma
- Código aberto e gratuito
- Um leque de inúmeras possibilidades para o desenvolvimento

6. Numpy

O Numpy é uma biblioteca criada em Python muito conhecida e de forma geral é capaz de realizar cálculos numéricos em *Arrays* Multidimensionais. Genericamente um *Array* Multidimensional é uma técnica de armazenagem de dados com mais de duas dimensões. Consequentemente a biblioteca Numpy consegue percorrer todas as dimensões do *array* e com isso realizar inúmeros cálculos matemáticos trazendo mais performance para o código e em poucas linhas de código. Essa biblioteca é amplamente utilizada em modelos de *Machine Learning*, Processamento de imagem ou *Data Visualization* e para tarefas matemáticas de alta complexidade.

7. Pandas

Pandas também é uma biblioteca criada em Python muito famosa no âmbito dos desenvolvedores que utilizam grande quantidade de dados, no caso os Cientistas de Dados. Essa biblioteca é capaz de fazer análises exploratórias na qual permite ao desenvolvedor ler, manipular, agregar e plotar os dados. É feita uma combinação de modelos de Machine Learning com essa fantástica biblioteca para fazer análises de DataFrames e o resultado é extremamente assertivo, afinal o desenvolvedor tem liberdade de fazer todos os tipos de manipulação nos dados de uma forma mais prática e acessível.

8. Matplotlib

Segundo o autor do artigo, Thiago Coutinho, esta famosa biblioteca foi desenvolvida por John D. Hunter. O Matplotlib foi criado em Python e é amplamente utilizado para criação de gráficos e visualizações de dados em geral. Ela foi um facilitador enorme para os Cientistas de Dados que passaram a utilizá-la, pois a plotagem é mais simples e rápida. Antes do Matplotlib os desenvolvedores utilizavam a Matlab que é um software capaz de projetar produtos e sistemas, então o Matplotlib veio para viabilizar a plotagem dos dados de uma forma prática.

Se tratando em plotagem, esta biblioteca também é conhecida pelo leque de oportunidades que é oferecido ao desenvolvedor. Ela entrega gráficos de barra, gráficos de pizza, gráficos de dispersão, gráficos de linhas e até histogramas.

9. Análise Exploratória de Dados

São métodos ou conjuntos de estatísticas e gráficos destinados a revelar as características nos dados. O objetivo principal é descobrir padrões, características, tendências, comportamento anômalos e *outliers*, visando sempre maximizar a obtenção de informações ocultas em sua estrutura.

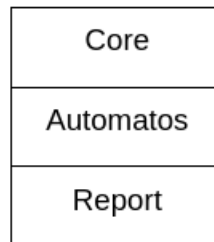
10. Gráfico de Média Móvel

A média móvel é um importante indicador da categoria dos rastreadores de tendência, no qual é feito o cálculo da média em um determinado período, neste trabalho será usado para calcular quantas pessoas foram infectadas, mortas ou se recuperaram em um determinado período de tempo com uma representação gráfica e posteriormente comparadas a outros resultados.

DESENVOLVIMENTO

1. Modelo de Entidades e Relacionamentos

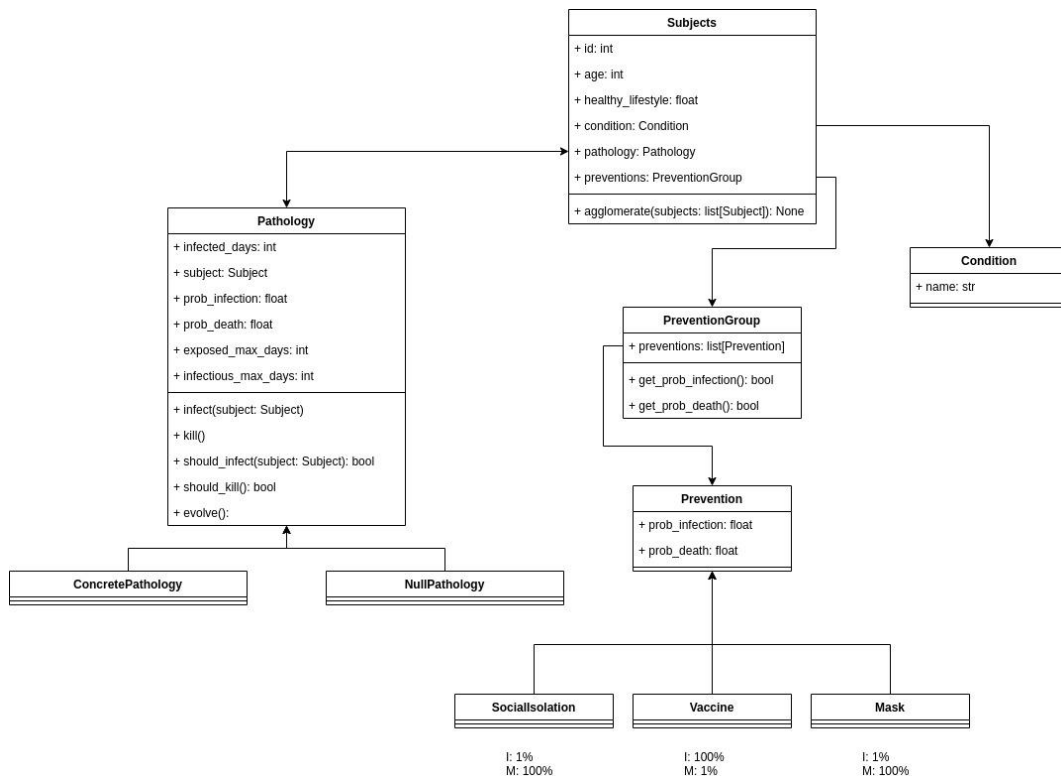
O início do estudo se deu no levantamento das entidades presentes em toda a aplicação e como elas interagem entre si. A abstração inicialmente foi dividida em três partes.



Domínios da aplicação

a. Core

A primeira representa a abstração das entidades reais em um sistema epidemiológico, chamados de core. Nessa camada estão presentes os indivíduos/sujeitos, condições do sujeito, doença e grupo de prevenção.



Primeira versão da camada Core

Os sujeitos (*Subjects*) representam cada indivíduo no ambiente e possuem atributos que estão associados aos estados do sujeito, como por exemplo sua idade (*age*), sua qualidade de vida (*healthy_lifestyle*) e seu identificador no ecossistema (*id*). Ainda, o sujeito contém referências da sua sua condição atual (*condition*), das prevenções adotadas (*preventions*) e da própria doença (*pathology*). Os indivíduos interagem entre si através do método aglomerar (*agglomerate*), podendo atingir o número máximo de 8 sujeitos por interação, ou seja, o número máximo de vizinhos que uma célula de uma matriz pode ter no eixo X e Y.

n1	n2	n3
n4	x	n5
n6	n7	n8

Vizinhos selecionados pelo método *agglomerate*

A condição (*Condition*) armazena o estado atual do indivíduo, podendo ser:

- NORMAL: Sujeito que não entrou em contato com a doença.
- EXPOSED: Sujeito que entrou em contato com a doença, mas ainda não desenvolveu os sintomas e que transmite a doença. Este passou pela condição NORMAL
- INFECTIOUS: Sujeito que entrou em contato com a doença, desenvolve os sintomas e que transmite a doença. Este passou pela condição EXPOSED.
- HEALED: Sujeito que entrou em contato com a doença, passou pelas condições infecciosas (EXPOSED e INFECTIOUS) e se recuperou após atingir o número máximo de dias com a doença.
- DEAD: Sujeito que entrou em contato com a doença, passou pelas condições infecciosas (EXPOSED e INFECTIOUS), mas faleceu antes de atingir a condição HEALED.

Já a doença (*Pathology*) possui atributos probabilísticos que interferem diretamente nas suas ações de infectar e levar o sujeito ao óbito. São elas: probabilidade de infecção (*prob_infection*), probabilidade de óbito (*prob_death*),

quantidade máxima de dias que um sujeito pode ficar exposto ao vírus sem sintomas (*exposed_max_days*) e com o sintomas (*infectious_max_days*). Além disso, a doença também armazena a referência do sujeito que está contaminado (*subject*) e a quantidade de dias desde a infecção (*infected_days*).

A doença ainda possui 2 (dois) métodos que são responsáveis por infectar ou levar a óbito um sujeito. O primeiro (*infect*), que utiliza de outro método (*should_infect*), calcula a probabilidade de infecção e modifica a condição do sujeito. Da mesma maneira, o segundo método (*kill*), que calcula a probabilidade de óbito (*should_kill*) e também modifica a condição do indivíduo.

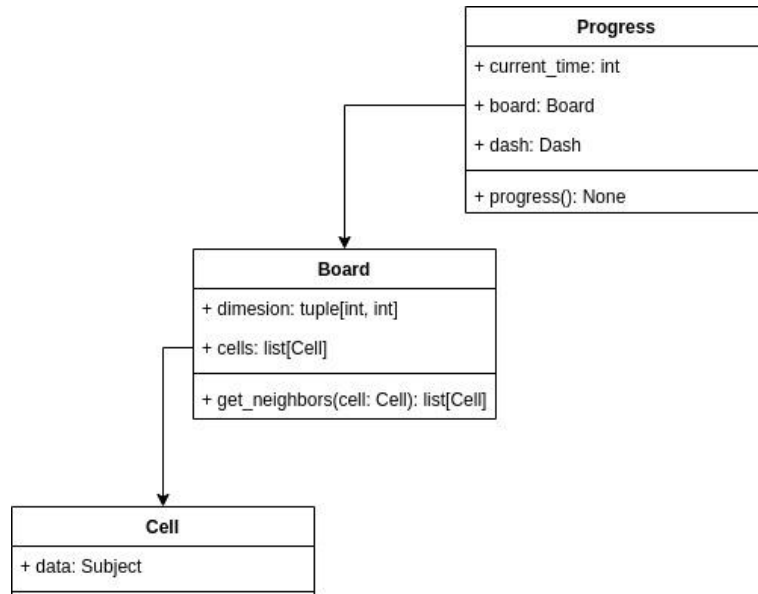
O sujeito, em seu método de interação com outros sujeitos, interage diretamente com o método de contágio da patologia.

Para melhor organização da aplicação foi utilizado o padrão de projeto *Null Object* na patologia. O indivíduo já nasce com a doença nula (*NullPathology*), mas ao ser contaminado recebe a doença concreta (*ConcretePathology*), ou seja, onde seus atributos e métodos possuem valores e ações.

Por fim, a prevenção (*Prevention*) possui atributos que representam a probabilidade de infecção (*prob_infection*) e a probabilidade de óbito (*prob_death*). Ela funciona como uma interface para diferentes tipos de prevenção com diferentes probabilidades (*SocialIsolation*, *Mask*, *Vaccine*). Ainda, é possível agrupar todas as prevenções e obter o resultado final de suas probabilidades (*PreventionGroup*).

b. Autômatos

Após a abstração das entidades principais do sistema, a segunda camada chamada autômatos busca uni-los de forma a proporcionar uma interação entre eles. Seus elementos são o *board*, as células de cada região do *board* e progresso.



Primeira versão da camada Autômato

A célula (*Cell*), a menor unidade desta camada, representa um indivíduo em toda a população. Seu único atributo (*data*) contém uma instância do sujeito (*Subject*).

O conjunto de indivíduos, ou seja, sua população (*Board*), contém um número n de células pré-definidas. Ela possui como atributos a dimensão da população (*dimension*) e a matriz quadrada com todas as células do ecossistema (*cells*). Ainda, possui um método que busca os 4 (quatro) vizinhos mais próximos dado uma célula (*get_neighbors*), necessário para a interação dos sujeitos.

Por fim, a entidade que faz abstração do tempo (*Progress*) e atua como um orquestrador do sistema. Ela é encarregada de inicializar a matriz, organizar a interação entre os indivíduos, evoluir a doença, ativar as patologias e iniciar os relatórios. Tudo isso através de seu único método (*progress*). Como atributos possui uma referência do tempo atual da simulação (*current_time*), da população (*board*) e dos relatórios (*report*).

c. Report

O relatório (*Report*), a última camada do sistema, é responsável pela monitoração de cada passo da simulação e pela geração de dados auxiliares para análises.

Dash
+ draw(board: board): None
+ report(board: board): None

Primeira versão da camada Report

Para representar a simulação de forma gráfica, o relatório possui 2 métodos responsáveis por desenhar os estados da matriz a cada passo do sistema e por gerar relatórios no formato CSV contendo informações de cada sujeito em sua célula e de suas condições ao longo da simulação.

Cada condição é representada por uma cor no relatório:

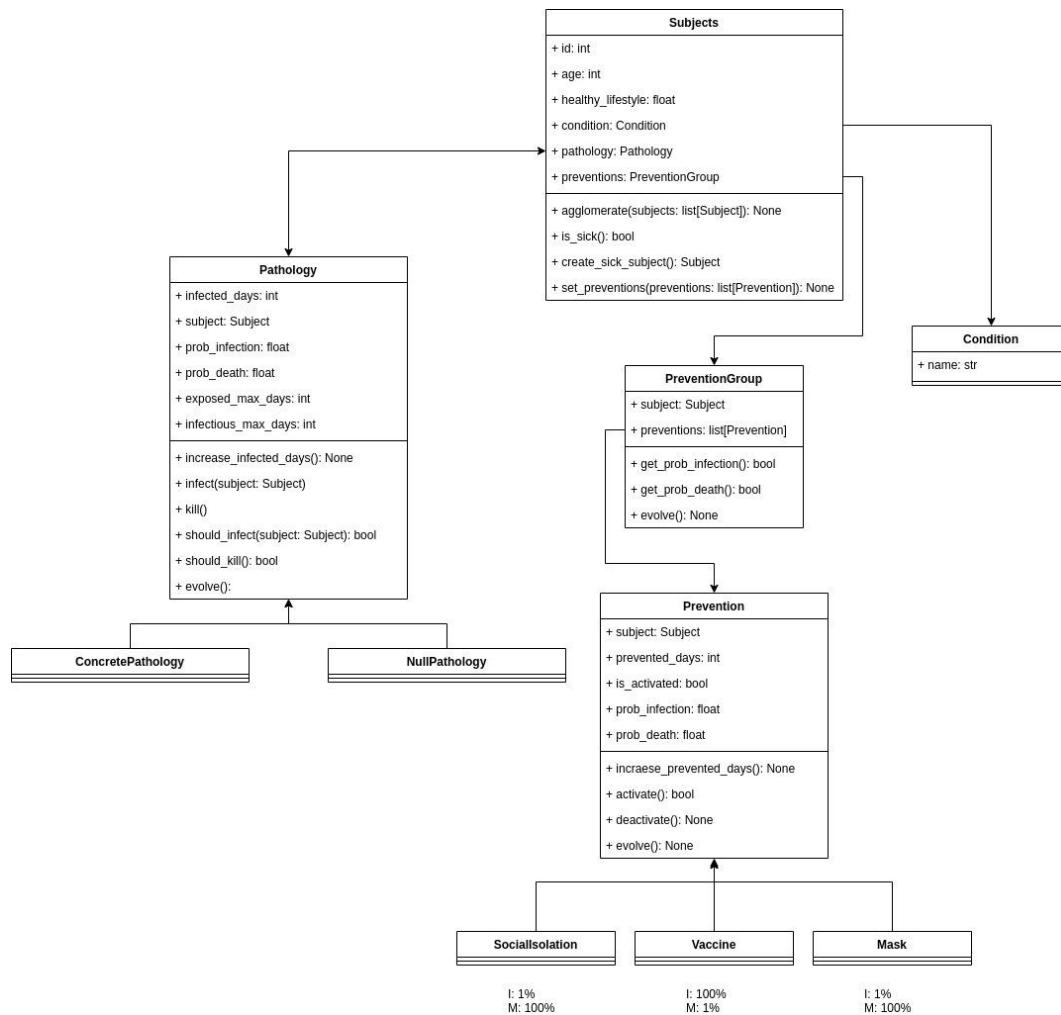
NORMAL	BRANCO
EXPOSED	LARANJA
INFECTIOUS	VERMELHO
HEALED	VERDE
DEAD	PRETO

Estrutura de cores dos *status* dos sujeitos

2. Atualizações no Modelo de Entidades e Relacionamento

Após o início da implementação do modelo desenvolvido, houve a necessidade de realizar mudanças nos 3 domínios da aplicação para atender melhor os requisitos.

a. Core



Última versão da camada Core

Para que as prevenções fossem atividades automaticamente e após o início da simulação, foram adicionados na classe *Prevention* atributos que representa se a mesma encontra-se ativa (*is_activated*) e a quantidade de dias desde sua ativação (*prevented_days*). Além disso, da mesma forma que a patologia armazena uma referência do sujeito contaminado, a prevenção passa a também referenciá-lo (*subject*).

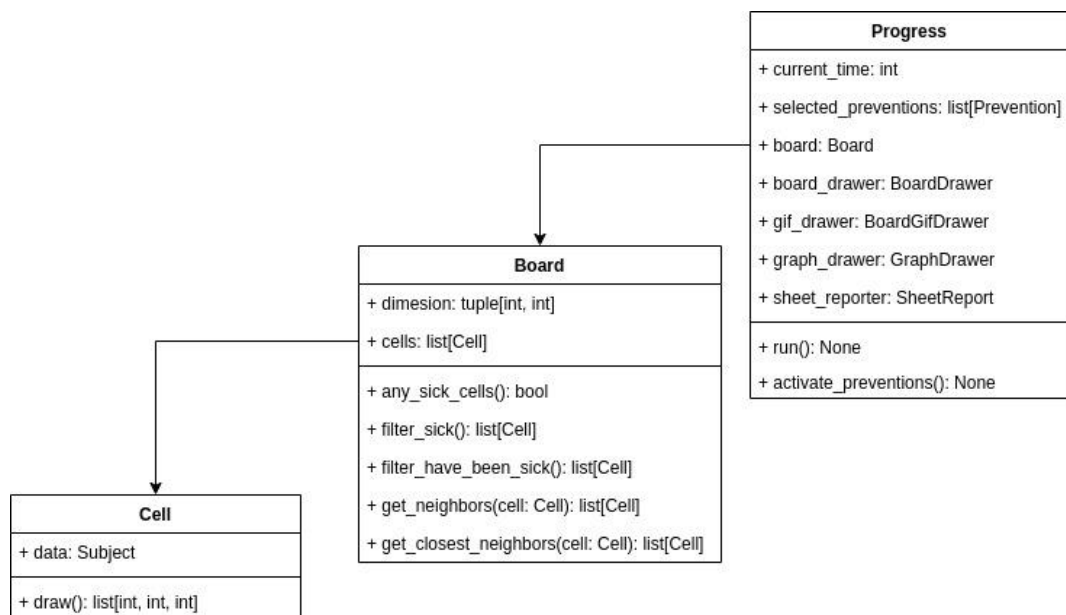
Além disso, o método *agglomerate* da classe *Subject* agora passa a selecionar somente os 4 principais vizinhos da célula, não mais 8. Essa alteração se deu pela a Taxa de Transmissão Real do vírus, que chegou a variar de 0 até 3.7 em alguns estados brasileiros, segundo dados apresentados pela Folha Vitória (2021). Isso significa que, em a cada 100 pessoas, pode haver uma transmissão para outras 37 pessoas.

Com a redução da seleção dos vizinhos para no máximo 4, a contaminação no sistema de simulação reduz em até 2 vezes, ficando mais próxima da Taxa de Transmissão Real.

n1	n2	n3
n4	x	n5
n6	n7	n8

Vizinhos selecionados pelo método *agglomerate* reduzido

b. Autômatos



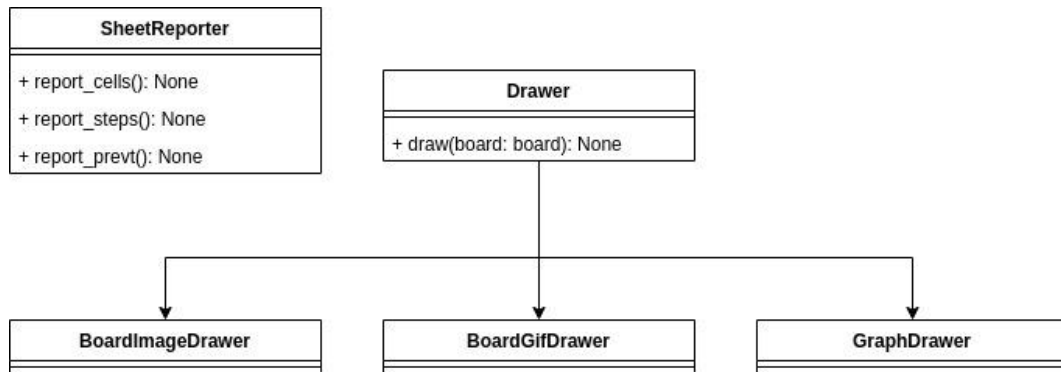
Última versão da camada Autômato

No domínio dos autômatos, a classe *Board* passou a receber novos métodos para filtragem de indivíduos presentes na matriz, como por exemplo filtrar qualquer célula com indivíduo doente (*filter_sick*), células com indivíduos que já estiveram doentes (*filter_have_been_sick*) e verificar se ainda existem sujeitos contaminados na matriz (*any_sick_cells*).

Além disso, a classe *Progress* que representa o progresso da simulação, agora passa a controlar a ativação das medidas preventivas e responsável por

chamar as classes do domínio *Report* e passar os *status* atual da simulação para que sejam gerados os gráficos e tabelas.

c. Report



Última versão da camada Report

Com a necessidade de criar novos gráficos, a classe *Report* passou a ser dividida em duas. A primeira responsável por criar as tabelas com informações de cada evolução da simulação, das prevenções e dos indivíduos (*SheetReport*). Já a segunda responsável por desenhar (*Drawer*), seja o gif da simulação sejam os gráficos, sendo cada subclasse responsável por um tipo de gráfico específico.

3. Cálculos envolvidos

a. Probabilidade de infecção (PI)

$$PI_T = PI_D * (PI_{P1} * PI_{Pn})$$

A probabilidade de infecção total (PI_T) é utilizada na aplicação para calcular se um indivíduo em contato com outro deve, ou não, infectá-lo. Essa variável é composta pela probabilidade de infecção da doença (PI_D) e pela multiplicação das probabilidades das prevenções (PI_{Pn}).

b. Probabilidade de óbito (PO)

$$PO_T = (PO * (I_S / I_{MS})) / QV * (POP_1 * POP_n)$$

A probabilidade de óbito total (PO_T) é utilizada na aplicação para calcular a probabilidade de óbito de um indivíduo infectado. Essa variável é composta pela multiplicação da probabilidade de óbito da doença simulada (PO) pela divisão da idade do sujeito (I_S) com a idade máxima que um sujeito pode ter (I_{MS}). O resultado é dividido pela multiplicação da qualidade de vida (QV) com a probabilidade de óbito de cada prevenção ativa (POP_n).

A parte da equação que representa a idade do indivíduo (I_S / I_{MS}) tem como objetivo adicionar maior probabilidade de óbito para sujeitos que possuem uma idade mais elevada, conforme mostram estudos sobre a menor imunidade de pessoas mais velhas.

4. Simulação

Os dados utilizados nas simulações são os do recente Coronavírus. Foi realizado em um *Board* de dimensão tamanho 51x51, onde é introduzido 1 (um) indivíduo infectado e o mesmo se encontra no meio da população, diante disso a matriz precisa ser de número ímpar. O experimento se dá início com a *Cell* infectada através do método *get_neighbors*, no qual ele vai buscar as suas quatro células vizinhas para interagir com elas pelo método *agglomerate*, em seguida, é feito os cálculos de infecção para descobrir se algum vizinho será infectado ou não e este ciclo se repetirá sucessivamente até todos os indivíduos estiverem curados.

Durante o avanço deste estudo foram feitas algumas simulações em diversos contextos para que pudessem mostrar eventuais efeitos reais de uma experiência forjada. Posteriormente, nestes mesmos experimentos, foram incluídas as prevenções (*Prevention*) com seus atributos e em sequência comparadas entre si. Nelas podem ser observadas desfechos com diferentes possibilidades através dos gráficos gerados.

Os contextos que foram estabelecidos nas simulações criadas pela aplicação e que serão apresentadas a seguir são:

Doença

Nome	Configuração
Probabilidade de Infecção	20%
Probabilidade de Óbito	3%
Quantidade de dias Incubado	7
Quantidade de dias Infectado	14

Prevenções

Nome	Máscara	Isolamento	Vacina
Probabilidade de Infecção	40%	50%	-
Probabilidade de Óbito	-	-	5%
Número de infectados para ativação	30%	20%	50%

Números de Mortes

Nessa circunstância, foram contabilizados o total de indivíduos mortos causadas pela patologia durante todo o seu ciclo de infecção no Board.

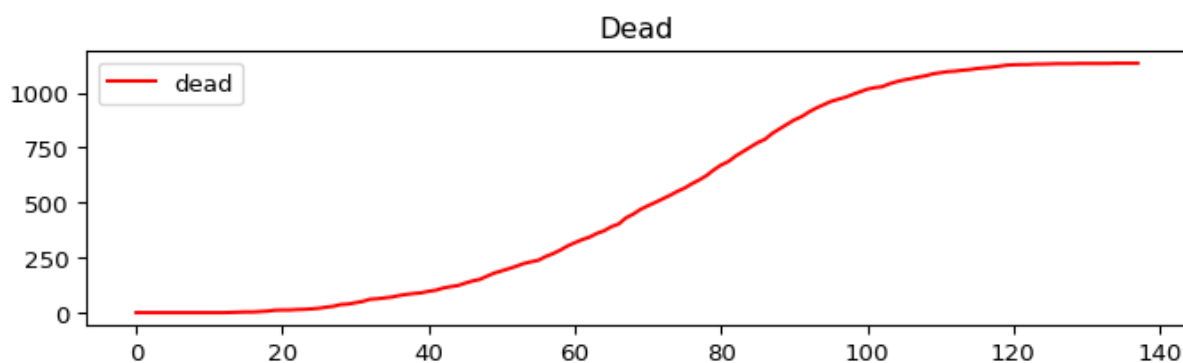


Gráfico do número de mortos

Pode-se observar que após quase 140 dias, houve mais de 1000 mortes devido às interações dos indivíduos infectados na simulação. Após visto este resultado, foi refeita a demonstração com as devidas prevenções aplicadas com o intuito de analisar a diferença e obteve o seguinte resultado:

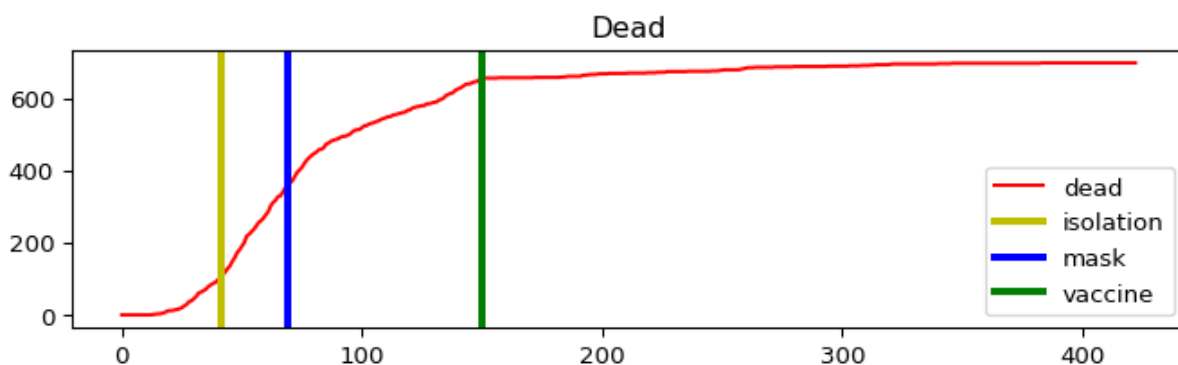


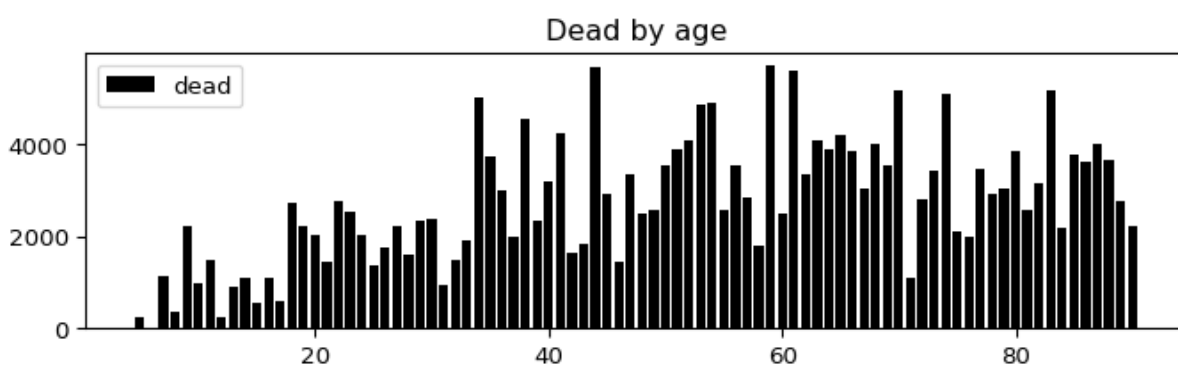
Gráfico do número de mortos com prevenções

As mortes foram significativamente reduzidas diante de todo o processo, houve aproximadamente 640 mortes em um período de um pouco mais de 400 dias. Nesta circunstância, as prevenções foram aplicadas separadamente após alguns dias e não todas de uma vez.

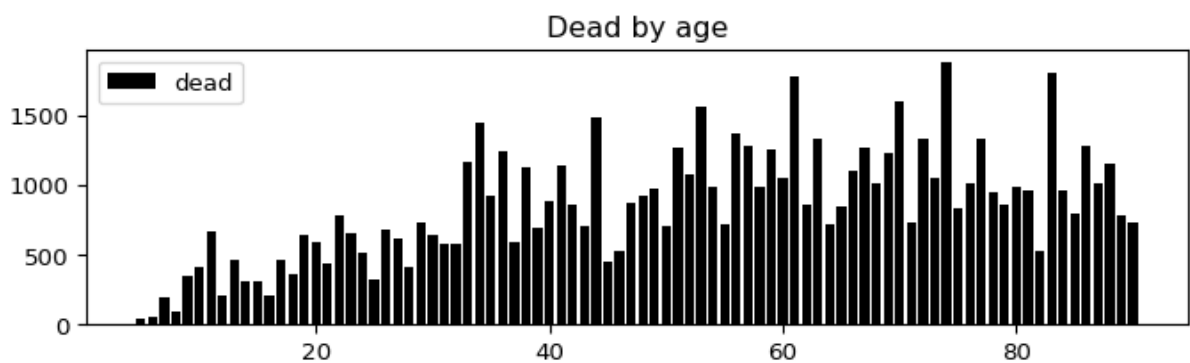
Primeiramente foi ativado o isolamento social que não obteve bons resultados, logo em seguida foi acionado a prevenção do uso de máscara e nota-se que a inclinação da linha vermelha diminui. Depois de um período maior foi aplicado as vacinas e fica nítido a redução das mortes que se estabilizou na casa dos 600.

Mortes por idade

Neste caso, foram calculados a quantidade de mortes por faixa etária causadas pela patologia durante todo o seu ciclo de infecção no *Board* e obteve-se o seguinte:



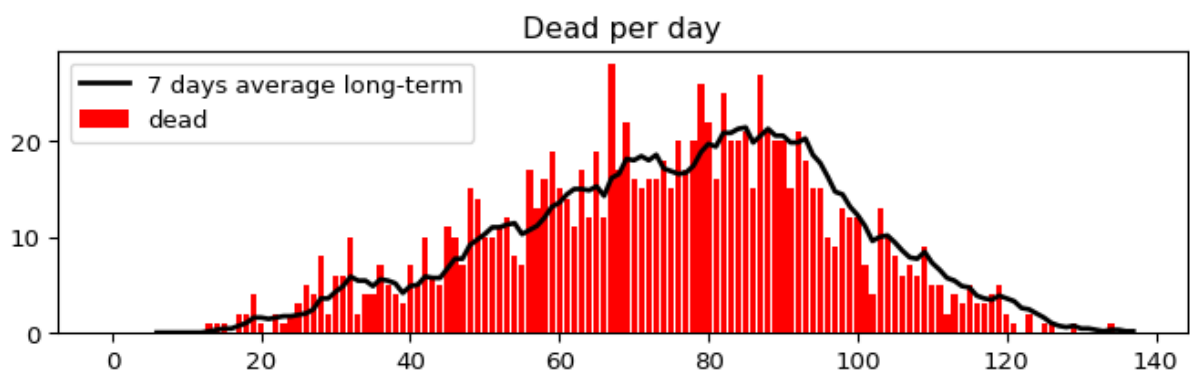
Após analisar o gráfico gerado, verifica-se que nas idades mais elevadas, os números de mortes ultrapassam os 4 mil, especificamente nos indivíduos de 44, 59 e 61 anos, que são as idades com maiores índices de mortos. Foi repetido a simulação, mas agora com as prevenções ativadas.



É notável a diferença da quantidade de mortos comparado com a primeira simulação, foram estimados menos de 2000 (dois mil) mortos, e as faixas etárias com maior número de mortes foram 74, 61 e 83 respectivamente. No total, foi reduzido em mais da metade de mortos, mas mantiveram a proporção, onde os indivíduos de idades menores morreram menos contrastado com as idades mais elevadas.

Mortes por dia

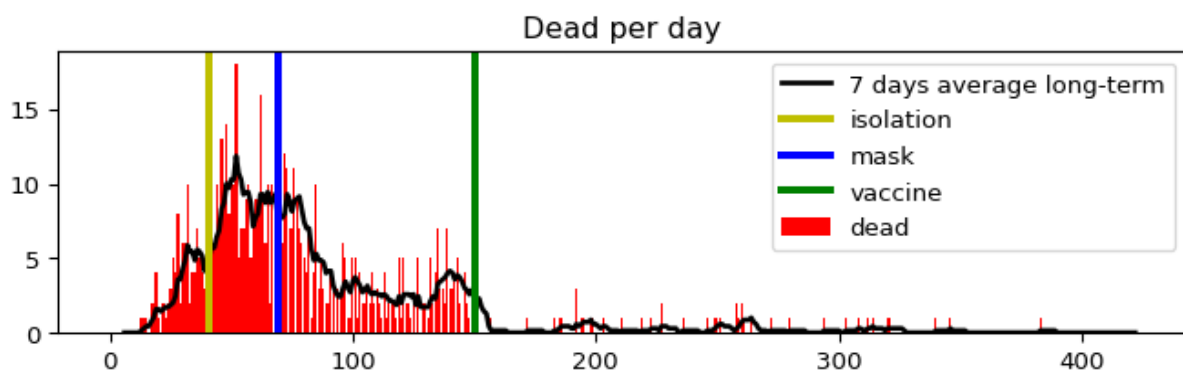
Nesta conjuntura, os dados estabelecidos foram o total de mortes que ocorreram por dia, de acordo com o ciclo de infecção patológica escolhida.



Em vermelho expressa a quantidade de mortes por dia, sem computar com as mortes do gráfico anterior. Já a linha mostra a média móvel de mortos dos últimos 7 dias, para se ter a percepção da diferença de mortes de uma semana para a outra.

A partir do dia 20, o número de mortes ultrapassa a da média móvel e desde este ponto, constata-se que há uma crescente que consegue superar a faixa de 20 mortes por dia. Após o 3º mês de contágio do vírus é que as mortes passam a cair.

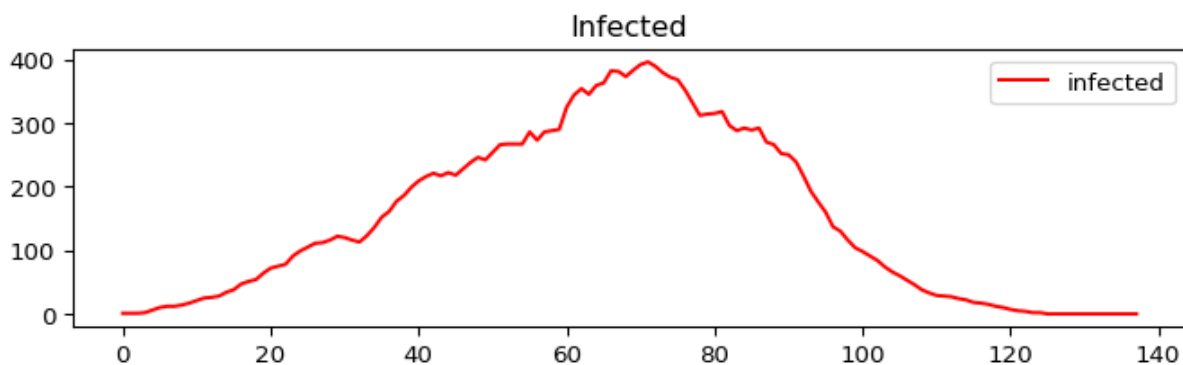
O mesmo experimento foi feito mas com as aplicações das prevenções ativadas e obteve-se o seguinte:



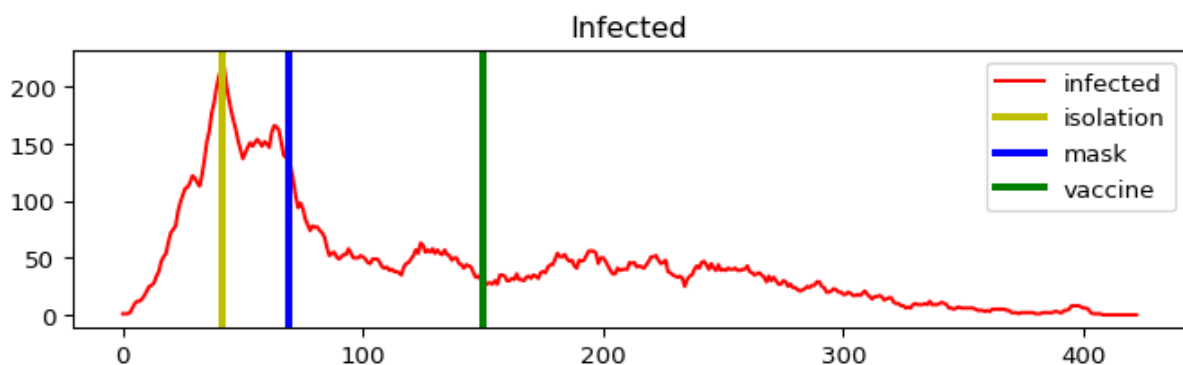
Inicialmente foi adicionado o mesmo padrão para adicionar as prevenções, onde o isolamento social foi empregado no começo da epidemia, mas não foi efetivo visto que as mortes não diminuíram. Logo depois foi utilizado as máscaras, e já é notado uma queda na média de mortes diárias de 10 para 5. E por fim, a aplicação da vacina fez com que as mortes fossem reduzidas drasticamente para quase zero.

Números de infectados

Neste cenário, foi estimado o número total de indivíduos infectados no *Board* durante todo o ciclo de vida do vírus.



Percebe-se que no início, o número de infecções são baixas e vão aumentando com o decorrer do tempo, na metade do decurso do agente patogênico, há um pico na quantidade de infectados, que alcança aproximadamente 400 indivíduos, e no final da progressão volta a cair. Esse tipo de comportamento, de qualquer vírus, é comum devido às pessoas que após entrar em contato com o vírus, adquirem anticorpos, mas o fato de ser infectado não garante imunidade total, o que torna possível a contaminação mais uma vez.

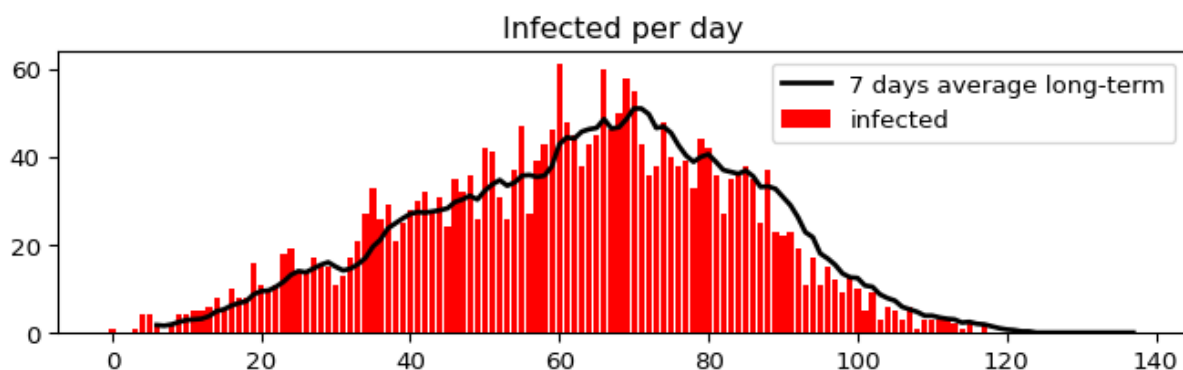


No mesmo cenário, mas agora com as aplicações das prevenções, nota-se que a alta taxa de infecções ocorre no início da experiência e não no meio, como antes. Primeiro foi aplicado o isolamento social quando atingiu um total de 200 infectados, como consequência, houve uma queda no índice de infecções. Após alguns dias, o mesmo voltou a crescer e logo em seguida foi aplicada outra prevenção, no caso o uso de máscara pelos indivíduos, o que de fato diminuiu mais ainda a quantidade de infecções ocorridas durante todo o processo de contaminação.

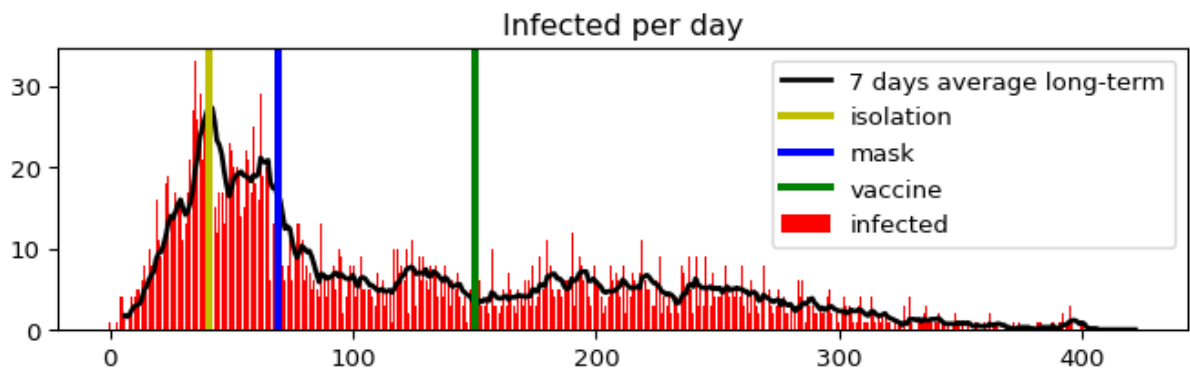
Aproximadamente no quinto (5º) mês, as vacinas foram aplicadas nos indivíduos e obteve-se uma média de 50 contaminações pelos próximos dias, até a sua redução gradativa e atingir 0 infectados.

Infectados por dia

Neste contexto, foi simulado a quantidade de indivíduos infectados por dia, de acordo com a patologia escolhida em todo o seu período de infecção.



Percebe-se que há um elevado número de casos de indivíduos infectados no final do segundo mês, que atinge o número de 60 infectados e a quantidade de infecções ultrapassam a média móvel da semana anterior. Após análise, foi simulado a versão com as prevenções ativadas.



Foram aplicadas as prevenções, primeiramente o isolamento social, que trouxe uma redução no número de infecções por dia. Logo em seguida foi aplicado o uso de máscaras pelos indivíduos, que somado com a prevenção anterior, a taxa de infecção caiu bruscamente. E após outro período de tempo, foi aplicado a vacina, a partir desse período a média de infecções foi mantida, mas a longo prazo houve mais diminuição nos números de infectados diariamente.

TRABALHOS FUTUROS

Como todo o sistema/projeto foi sustentado em cima da pandemia do Covid-19, é possível com alguns ajustes, modificações e criações de variáveis, a simulação do cenário de outras doenças. Certamente que para essas modificações surtirem os efeitos desejados na simulação, é essencial o estudo sobre a doença e entendimento de todas as variáveis possíveis.

Outra melhoria cabível seria a implementação de um software executável com uma interface gráfica amigável que auxilie o usuário a entender mais como são embasados os dados calculados e plotados.

CONSIDERAÇÕES FINAIS

O objetivo deste trabalho foi desenvolver uma aplicação onde é possível simular um cenário de pandemia da Covid-19. Para vários cenários diferentes, é possível ativar ou desativar prevenções como o uso de máscara e o isolamento social e também ativar a vacina. As prevenções, se ativadas, são disparadas automaticamente após cerca de 25% (25 por cento) do total de pessoas infectadas. Para mostrarmos a importância das prevenções, realizamos algumas simulações.

Foi levado em consideração como prevenção da contração do vírus a qualidade de vida de cada usuário ou visualmente exemplificando, cada quadrado branco no board. Também foi agregado ao mesmo as prevenções de isolamento social, o uso de máscara e a vacinação. No caso da vacinação, foi passado como parâmetro a idade de cada pessoa do board.

REFERÊNCIAS BIBLIOGRÁFICAS

BARROS, Rodrigo. Saiba por que algumas pessoas morrem de Covid-19 mesmo tendo completado o esquema vacinal. Folha de Pernambuco, [S. l.], p. 1, 12 ago.

2021. Disponível em: <https://www.folhape.com.br> Acesso em: 14 set. 2021.

BGASPAROTTO. Null Object. Disponível em:

<https://bgasparotto.com/pt/design-patterns/null-object>. Acesso em 25 out. 2021.

CÁCERES, Sheila. Linguagem de Programação Orientada a Objeto. UNIP - Curso: Ciência da Computação. (1/sem.2009). Disponível em:

<http://sheilacaceres.com/courses/lpoo/aulas/LPOO2-classes.pdf> Acesso em: 28 set. 2021.

CHEN, Peter Pin-Shan. The Entity-Relationship Model-Toward a Unified View of Data. (01. Vol). Massachusetts: 1976.

CORONAVÍRUS: O isolamento social começou um mês após o primeiro caso confirmado no Brasil. Jdv, [S. l.], p. 1, 16 jun. 2020. Disponível em:

<https://www.jdv.com.br>. Acesso em: 14 set. 2021.

COUTINHO, Thiago. O que é python matplotlib? Conheça a biblioteca de gráficos!.

Disponível em: <https://www.voitto.com.br/blog/artigo/o-que-e-python-matplotlib>.

Acesso em: 15 out. 2021.

CALDAS, Edmar: Análise Exploratória de Dados. Disponível em:

<https://www.linkedin.com/pulse/o-que-%C3%A9-an%C3%A1lise-explorat%C3%B3ria-de-dados-edmar-caldas/?originalSubdomain=pt>. Acesso em: 6 nov. 2021.

FIOCRUZ. O que é uma pandemia. Fiocruz, [S. l.], p. 1, 28 jul. 2021. Disponível em:

<https://www.bio.fiocruz.br>. Acesso em: 26 set. 2021.

KELLYANN, Fitzpatrick. RedMonk Top 20 Languages Over Time. Disponível em: <https://redmonk.com/kfitzpatrick/2021/03/02/redmonk-top-20-languages-over-time-january-2021/>. Acesso em: 03 out. 2021.

GLOBO, Uso de máscara reduz chance de infecção por coronavírus em 87%, diz estudo de universidades do RS. G1 RS, [S. l.], p. 1, 5 mar. 2021. Disponível em: <https://g1.globo.com>. Acesso em: 14 set. 2021.

GURU, Refactoring. Padrões de Projeto. Disponível em: <https://refactoring.guru/pt-br/design-patterns>. Acesso em 23 out. 2021.

HORN, Michelle. Python: o que é, como usar, guia pra aprender a linguagem. Disponível em: <https://blog.betrybe.com/python/>. Acesso em: 03 out. 2021.

LANDIM, Raquel et al. Uma a cada 25 mil pessoas morre de Covid-19 mesmo após 2ª dose da Coronavac. CNN Brasil, [S. l.], p. 1, 11 mai. 2021. Disponível em: <https://www.cnnbrasil.com.br>. Acesso em: 14 set. 2021.

PEDAMKAR, Priya. Multidimensional Array in Python. Disponível em: <https://www.educba.com/multidimensional-array-in-python/>. Acesso em 03: out. 2021

RIBEIRO, Lucas. Introdução a Biblioteca Pandas. Disponível em: <https://medium.com/tech-grupozap/introdu%C3%A7%C3%A3o-a-biblioteca-pandas-89fa8ed4fa38>. Acesso em: 03 out. 2021.

SANTIAGO, Luiz. Entendendo a biblioteca NumPy. Disponível em: <https://medium.com/ensina-ai/entendendo-a-biblioteca-numpy-4858fde63355>. Acesso em: 03 out. 2021.

VITÓRIA, Folha. Taxa de transmissão do coronavírus no ES é de 1.8; saiba o que isso significa. Folha Vitória, [S. l.], p. 1, 4 jun. 2020. Disponível em: <https://www.folhavoria.com.br/saude/noticia/06/2020/taxa-de-transmissao-do-coronavirus-no-es-e-de-1-8-saiba-o-que-isso-significa>. Acesso em: 31 out. 2021.

LEITURAS RECOMENDADAS

NULL Object Design Pattern. In: Source Making. [S. l.], 5 mar. 2021. Disponível em: <https://sourcemaking.com/>. Acesso em: 26 set. 2021.