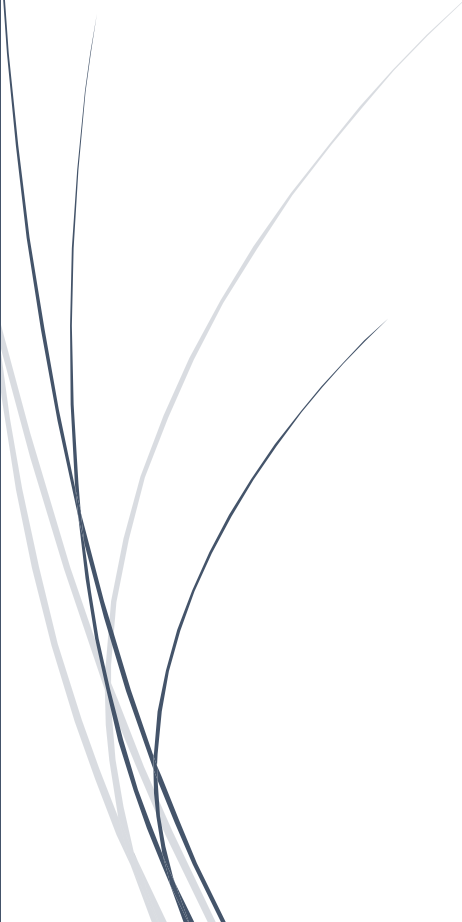


A dark blue vertical bar on the left side of the page. A blue arrow points to the right from the bar, containing the date.

12/17/2023

Sistema de Gestão e Simulação de Eventos

Documentação

Several thin, dark blue wavy lines originate from the bottom left and curve upwards and to the right.

David Maurício Ferreira, nº 202300443
José Luís Parreira de Oliveira, nº 202300558
IPOO, 1ºSEMESTRE

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

Índice

| | | |
|------------|--|-----------|
| 1 | <i>Introdução</i> | 3 |
| 2 | <i>Implementação</i> | 3 |
| 2.1 | Classes | 3 |
| 2.1.1 | Event | 3 |
| 2.1.2 | Edition | 5 |
| 2.1.3 | Lecture | 6 |
| 2.1.4 | Speaker..... | 7 |
| 2.1.5 | Place | 8 |
| 2.1.6 | Room | 8 |
| 2.1.7 | Manager | 9 |
| 2.1.8 | Console..... | 10 |
| 2.2 | Características e funcionalidades do programa | 13 |
| 3 | <i>Conclusões</i> | 13 |

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

1 Introdução

Neste projeto é desenvolver um programa que possibilita a gestão e simulação de eventos em que toda a informação é gerida por uma consola. Esta informação inclui palestrantes, palestras, eventos, edições de eventos, salas e locais. Também será possível a simulação destes eventos e calcular a popularidade cada vez que é feita uma simulação.

2 Implementação

2.1 Classes

2.1.1 Event

Esta classe permite armazenar as características de um evento.

2.1.1.1 Atributos

A classe Event tem os seguintes atributos:

- String name – Contém o nome do evento.
- Edition[] editionList – Guarda uma lista das edições do evento (uma lista do tipo de classe Edition).
- int numEditions – Variável auxiliar do array editionList, contendo o número de edições no array.
- PopularityLevel popularity – Contém a popularidade do evento do tipo PopularityLevel (um tipo enumerado).

2.1.1.2 Métodos

Esta classe contém os seguintes métodos:

- Métodos construtores
 - Event(String name) – inicializa os atributos com o nome fornecido, o array vazio e a popularidade de UNKNOWN.
 - Event(String name, PopularityLevel popularity) – igual ao anterior mas inicializa a popularidade com o valor fornecido.
- Métodos seletores
 - Para retornar o valor dos atributos referidos anteriormente.
 - String getName(), Edition[] getEditionList(), PopularityLevel getPopularity(), int getNumEditions()

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

- Métodos modificadores e outros
 - void setName(String name) – Alterar nome do evento
 - void addEdition(Edition edition) – Permite adicionar uma edição ao array editionList do evento, incrementando a variável auxiliar numEditions. Se o array atingiu a sua capacidade, os seus conteúdos são antes copiados para um array de maiores dimensões antes da adição da nova edição.
 - void show() – Permite mostrar ao utilizador a informação do evento.
 - void calculateEventPopularity() – Se uma edição tiver sido simulada, permite calcular a popularidade do evento. Para isso é calculado um valor médio com base no número de participantes e na capacidade das salas. Este valor é convertido em percentagem e usando no método descrito em seguida para obter a popularidade do evento.
 - PopularityLevel calculatePopularity(double average) – permite com base no valor médio em forma de percentagem (d) retornar o nível de popularidade correspondente.
 - $d = 0$ ou não válido – UNKNOWN
 - $0 < d < 0,25$ – WEAK
 - $0,25 \leq d < 0,5$ – NEW
 - $0,5 \leq d < 0,75$ – BIG
 - $0,75 \leq d \leq 1$ – TOP

2.1.1.3 Enum PopularityLevel

Tipo enumerado com valores constantes para o tipo de popularidade.

É usado para definir a popularidade do evento e a popularidade do palestrante (speaker).

Contém um método construtor para atribuir a cada constante:

- um valor String contendo o texto a mostrar ao utilizador do programa
- um valor double contendo o peso da constante no cálculo do número de participantes nas simulações

Contém um método para retornar o valor double da constante para os cálculos nas simulações.

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

Contém também um método personalizado toString() para retornar o valor String da constante, texto que será para mostrar ao utilizador a popularidade definida.

2.1.2 Edition

Esta classe permite armazenar as características de uma edição de um evento.

2.1.2.1 Atributos

A classe Edition tem os seguintes atributos:

- int editionNumber;
- int totalEditions;
- LocalDate startDate; - Contém a data de início da edição.
- LocalDate endDate; - Contém a data de fim da edição
- Place place; - Contém o local onde se realiza a edição (classe Place).
- Lecture[] lectureList; - Guarda uma lista de palestras.
- int numLectures; - Variável auxiliar do array lectureList, contendo o número de palestras.
- int numRegistrationsLectures;
- int numRegistrationsEdition;

2.1.2.2 Métodos

Esta classe contém os seguintes métodos:

- Métodos construtores
 - Edition(Place place)
 - Edition(Place place, LocalDate startDate, LocalDate endDate)
- Métodos seletores
 - Para retornar o valor dos atributos referidos
 - getEditionNumber(), getTotalEdition(), getStartDate(), getEndDate(), getPlace(), getLectureList(), getNumLectures(), getNumRegistrationsEdition(), getNumRegistrationsLectures(), getEvent()
- Métodos modificadores e outros
 - setStartDate(int year, int month, int day)
 - setEndDate(int year, int month, int day)
 - setEvent(Event event)
 - void addLecture(Lecture lecture) - Permite adicionar uma palestra ao array lectureList da edição, incrementando a variável auxiliar numLectures. Se o array atingiu a sua capacidade, os seus conteúdos

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

são antes copiados para um array de maiores dimensões antes da adição da nova palestra.

- `boolean findIntersectingLecturesBool(Lecture lecture)` – Verificar se existem outras palestras a ocorrer ao mesmo tempo. Retoma um valor booleano.
- `Lecture findIntersectingLectures(Lecture lecture)` – Igual ao anterior mas retoma a palestra.
- `void show()` - Permite mostrar ao utilizador a informação da edição.
- `void simulateRegistrations()` – Simulação da

2.1.3 Lecture

Esta classe permite armazenar as características de uma palestra.

2.1.3.1 Atributos

A classe `Lecture` tem os seguintes atributos:

- `String title;` - Contém o nome/título da palestra.
- `Speaker[] speakersList;` - Guarda uma lista de palestrantes.
- `int numSpeakers;` - Variável auxiliar do array `speakersList`, contendo o número de palestrantes.
- `LocalDateTime startDateTime;` - Contém a data e hora de início da palestra.
- `LocalDateTime endDateTime;` - Contém a data e hora do fim da palestra.
- `Room room;` - Contém a sala onde decorre a palestra.
- `int numRegistrationsLecture;`
- `Event event;`
- `Edition edition;`

2.1.3.2 Métodos

Esta classe contém os seguintes métodos:

- Métodos construtores
 - `Lecture(String title)`
 - `Lecture(String title, LocalDateTime startDateTime, LocalDateTime endDateTime, Room room)`
- Métodos seletores
 - Para retornar o valor dos atributos referidos
 - `getTitle()`, `getSpeakersList()`, `getNumSpeakers()`, `getStartDateTime()`, `getEndDateTime()`, `getRoom()`, `getNumRegistrationsLecture()`, `getEvent()`, `getEdition()`

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

- Métodos modificadores e outros
 - void setStartDateTime(int year, int month, int day, int hours, int minutes)
 - void setEndDateTime(int year, int month, int day, int hours, int minutes)
 - void setEvent(Event event)
 - void setEdition(Edition edition)
 - void setRoom(Room room)
 - void addSpeaker(Speaker speaker) – Adiciona um palestrante à palestra
 - void show() – Mostra toda a informação da palestra
 - boolean isPossibleToSimulate() – Verifica se é possível simular a palestra
 - double highestSpeakerPopularity(Lecture lecture) – verifica quem é palestrante com maior popularidade
 - void simulateRegistrations() – Simula a palestra.

2.1.4 Speaker

Esta classe permite armazenar as características de um palestrante.

2.1.4.1 Atributos

A classe Speaker tem os seguintes atributos:

- String name; - Contém o nome do palestrante.
- Lecture[] lectureList; - Contém a lista de palestras em que o palestrante está inscrito.
- int numLectures; - Variável auxiliar do array lectureList, contendo o número de palestras.
- Event.PopularityLevel popularity; - Contém a popularidade do palestrante.

2.1.4.2 Métodos

Esta classe contém os seguintes métodos:

- Métodos construtores
 - Speaker(String name)
 - Speaker(String name, Event.PopularityLevel popularity)
- Métodos seletores

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

- Para retornar o valor dos atributos referidos
- getName(), getLectureList(), getNumLectures(), getPopularity()
- Métodos modificadores e outros
 - void setName(String name)
 - void addLecture (Lecture lecture) – Adiciona uma palestra a um palestrante
 - void show() – Mostra todas as informações relacionadas com o palestrante
 - void calculateSpeakerPopularity() – Calcula a popularidade do palestrante

2.1.5 Place

Esta classe permite armazenar as características de um local.

2.1.5.1 Atributos

A classe Place tem os seguintes atributos:

- Room[] roomList; - Lista de salas
- int numRooms; - Variavel auxiliar para contagem de salas existentes
- String name; - Nome do local.

2.1.5.2 Métodos

Esta classe contém os seguintes métodos:

- Métodos construtores
 - Place(String name)
- Métodos seletores
 - Para retornar o valor dos atributos referidos
 - getRoomList(), getNumRooms(), getName()
- Métodos modificadores e outros
 - void setName(String name)
 - void addRoom(Room room) – Adiciona uma sala ao local.
 - void show() – Mostra todas as informações relacionadas com a sala.

2.1.6 Room

Esta classe permite armazenar as características de uma sala.

2.1.6.1 Atributos

A classe Room tem os seguintes atributos:

- String name; - Nome da sala
- int capacity; - Capacidade da sala

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

2.1.6.2 Métodos

Esta classe contém os seguintes métodos:

- Métodos construtores
 - Room(String name, int capacity)
- Métodos seletores
 - Para retornar o valor dos atributos referidos
 - getName(), getCapacity()
- Métodos modificadores e outros
 - void setName(String name)
 - void setCapacity(int capacity)
 - void show() – Mostrar todas as informações relacionadas com a sala.

2.1.7 Manager

Esta classe permite gerir os eventos.

2.1.7.1 Atributos

A classe Manager tem os seguintes atributos:

- Speaker[] speakerList; - Lista de palestrantes
- Event[] eventList; - Lista de eventos.
- Place[] placeList; - Lista de locais.
- int numSpeakers; - Variável auxiliar para contagem de palestrantes existentes
- int numEvents; - Variável auxiliar para contagem de eventos existentes
- int numPlaces; - Variável auxiliar para contagem de locais existentes

2.1.7.2 Métodos

Esta classe contém os seguintes métodos:

- Método construtor
 - Manager()
- Métodos seletores
 - Para retornar o valor dos atributos referidos
 - getSpeakerList(), getEventList(), getPlaceList(), getNumSpeakers(), getNumEvents(), getNumPlaces()
- Métodos modificadores e outros
 - void createEvent(String name) – Cria um evento recebendo um nome
 - void createEvent(String name, Event.PopularityLevel popularity) – Cria um evento recebendo a popularidade e o nome.

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

- void createSpeaker(String name) – Cria um palestrante recebendo um nome
- void createSpeaker(String name, Event.PopularityLevel popularity) – Cria um palestrante recebendo um nome e popularidade.
- void createPlace(String name) – Cria um local recebendo um nome.
- void createEdition(Event event, Place place) – Cria uma edição recebendo um nome e um local.
- void createEdition(Event event, Place place, LocalDate startDate, LocalDate endDate) - Cria uma edição recebendo um nome, um local, data de inicio e fim.
- void createRoom(Place place, String name, int capacity) – Cria uma sala recebendo um local, nome e capacidade
- Edition[] getEditionList(Event event)
- void createLecture(Event event, int edition, String title) – cria uma palestra recebendo um evento, edicao e titulo.
- void createLecture(Event event, int edition, String title, LocalDateTime startDateTime, LocalDateTime endDateTime, Room room) – cria uma palestra recebendo um evento, edição, titulo, data de inicio e fim e uma sala.
- void addSpeaker(Lecture lecture, Speaker speaker) – adiciona um palestrante a uma palestra.
- void addRoom(Lecture lecture, Room room) – adiciona uma sala a uma palestra.
- void simulateLecture(Lecture lecture) – simula os registos a uma palestra.
- void simulateEdition(Edition edition) – simula os registos a uma edição e calcula a popularidade do evento associado à edição

2.1.8 Console

Esta classe permite criar um gestor e uma consola de modo ao utilizador conseguir interagir com o programa.

2.1.8.1 Atributos

A classe tem os seguintes atributos:

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

- Scanner sc; - Objeto de scanner que permite a interação entre o utilizador e o programa.
- Manager manager; - Objeto manager para aceder a métodos dessa classe

2.1.8.2 Métodos

Esta classe contém os seguintes métodos:

- Método construtor
 - Console()
- Outros métodos
 - void startConsole() – Começar o loop da consola e fechar consola se for terminada
 - int mainMenu() – Inicia o menu principal
 - int manageEvents() – Inicia o menu para gerir eventos
 - int editEvent(Event event) - Inicia o menu para editar eventos
 - void managePlaces() - Inicia o menu para gerir locais
 - int manageSpeakers() - Inicia o menu para gerir palestrantes
 - void createEvent() – Cria um evento
 - void createPlace() – Cria um local
 - int editPlace() – abre o menu de edição do local
 - void searchPlace() – Procura um local
 - Place findPlace() – Procura um local mas retorna o objeto do local
 - void removePlace() – Remove um local
 - void editPlaceName(Place place) – Edita o nome do local
 - void addRoom(Place place) – Adiciona uma sala a um local
 - void editRoom(Place place) – Edita a sala de um local
 - void removeRoom(Place place) – Remove um sala de um local
 - void listAllPlaces() – Lista todos os locais existentes
 - void createSpeaker() – Cria um palestrante
 - void void editSpeakerName(Speaker speaker) - Permite a edição de um nome de um palestrante
 - void searchSpeaker() – Procura um palestrante
 - Speaker findSpeaker() – Procura um palestrante mas retorna o objeto de palestrante

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

- void listAllSpeakers() – Lista todos os palestrantes existentes
- void removeSpeaker() – Remove um palestrante
- void searchEvent() – Procura um evento
- Event findEvent() – Procura um evento mas retorna o objeto de evento
- Void listAllEvents() – Lista todos os eventos existentes
- int editEdition(Event event, Edition edition) – Abre o menu de edição de edições.
- void createEdition(Event event) – Cria um edição
- void editEdition(Event event) - Permite a edição de uma edição
- void removeEvent() – Remove um evento
- place createNewPlaceOrNot() – Verifica se o utilizador quer criar um novo local ou usar um já definido.
- void searchEdition(Event event) – Procura por uma edição.
- void listAllEditions(Event event) – Lista todas as edição existentes.
- void simulateEdition(Event event) – Simula uma edição.
- Void next() – Usado para a navegação pelos vários menus.
- int logout() – Permite o fechar da consola
- void createLecture(Event event, Edition edition) – Cria uma palestra
- void searchLecture(Event event, Edition edition) – Procura uma palestra
- void simulateLecture(Event event, Edition edition) – Simula uma palestra.
- void setStartDate(Edition edition) – Definir a data de inicio de uma edição
- void setEndDate(Edition edition) - Definir a data de fim de uma edição
- void setPlace(Edition edition) - Definir o local de uma edição
- Lecture findLecture(Event event, Edition edition) – Procura de uma palestra mas retorna o objeto da palestra
- int editLecture(Event event, Edition edition, Lecture lecture) – Permite editar uma palestra

SISTEMA DE GESTÃO E SIMULAÇÃO DE EVENTOS

- void addSpeaker(Lecture lecture) – Adiciona um palestrante a uma palestra
- void setTitle(Lecture lecture) – define o título de uma palestra
- void setRoom(Edition edition, Lecture lecture) – define a sala de uma palestra
- void setStartDateTime(Lecture lecture) – define a data de início de uma palestra
- void setEndDateTime(Lecture lecture) – define a data de fim de uma palestra

2.2 Características e funcionalidades do programa

O programa permite a um gestor pela consola criar, editar, ver e simular eventos, palestras e edições. Permite também a criação e edição de palestrantes e adicionar esses palestrantes a palestras.

Os palestrantes e eventos têm uma popularidade que é atualizada a cada vez que se simula uma palestra ou edição. Podemos criar locais e salas nesses locais que as mesmas têm uma capacidade que afeta nos cálculos de popularidade.

3 Conclusões

O objetivo do trabalho foi concluído que foi criar um programa que um utilizador poderia gerir um sistema de eventos com palestrantes, palestras, edições e locais por meio de uma consola. As maiores dificuldades sentidas foi na parte da simulação de eventos e palestras.