UML Class Diagram

Scheduler

- Day days[31]
- int currentDay

- void initializeDays()
- void saveEventsTo_txt()
- void loadEventsFrom_txt()
- void option_list(int index)
- Scheduler(int currentDay)
- ~Scheduler()
- void scheduleEvent(int date, Event& event)
- void cancelEvent(int date, string& title, bool deleteRepeats)
- void shiftEvent(int date, string& title, int newDate)
- void setDayOff(int date)
- void viewDaySchedule(int day) const
- void viewWeekSchedule(int startDay) const
- void displayScheduler()
- bool isEventRepeating(int date, const string& title) const
- void displayScheduler_print(int today)

1 controls 1..31

Day

- int date
- bool isDayOff
- Event events[10]
- int eventCount
- string dayOfWeek

- sortEvents()
- Day(int date = 0, string dayOfWeek = "")
- void addEvent(Event& event)
- void deleteEvent(string& title)
- void shiftEvent(string& title, int newDate, Day* days)
- void clearEvents()
- string toString() const
- bool toString_print() const
- string formatDayDataToString() const
- void extractDayData(string& dayStr)

Time

- int hour
- int minute

- Time(int hour = 0, int minute = 0)
- bool isLessThan(Time& comparisonTime) const
- bool isGreaterThan(Time& comparisonTime) const
- bool isEqualTo(Time& comparisonTime) const
- string toString() const
- void fromString(string& timeString)

1

1 contains 1..*

1 contains 1..10

Event

- string title
- Time startTime
- Time endTime
- string repeatType

- Event(string title = "EVENT", Time startTime = Time(), Time endTime = Time(), string repeatType = "none")
- bool overlaps(Event& comparisonEvent) const
- string toString() const
- string formatEventDataToString() const
- void extractEventData(const string& eventString)

Exceptions

Exceptions

- int errorCode
- string errorMessage

- Exceptions(int code)
- virtual const char* what() const noexcept

throws

throws

throws

throws

throws

throws

std::exception

SchedulerExceptions

- SchedulerExceptions(int code)

TimeExceptions

- TimeExceptions(int code)

DayExceptions

- DayExceptions(int code)

EventExceptions

- EventExceptions(int code)

- public
- protected
- private