README DOC

Schedule Masters

Team Members:

Ishanvi Deodhar, Aditi Gokul, Gabby Moll

Revision Date: 5/14/25

Program Purpose:

The purpose of this program is to create and manage events with dates, times, and priorities. Users will be able to add, view, edit, and delete events from their schedule. The program helps users organize their time better by allowing them to categorize events and set reminders.

Target User Profile:

High school and college students who need help organizing their academic schedules. These students typically have multiple classes, assignments, and extracurricular activities to balance. They prefer simple, straightforward applications that don't require much setup or learning time.

Feature List:

- Add new events with title, description, and priority level
- Different types of events (Assignment, Meeting)
- View all scheduled events sorted by date
- Filter events by type and priority
- Edit existing events
- Delete events
- Mark events as completed

Instructions:

The application interface we want to create will consist of a display area in the center and buttons at the bottom:

Adding Events:

- Click the "Add Event" button
- Enter a title for the event when prompted
- Enter a description (optional)
- Select the event type (Assignment, Meeting, or Reminder)
- Select a category if you have created any
- Fill in the type-specific details:

Assignment: Course name, priority

Meeting: Location, attendee names

Viewing Events: Click the "View Events" button to see all events.

Events are displayed with:

- Event type emoji
- Status indicator
- ID number
- Title and description
- Category (if assigned)
- Type-specific details

Mark Complete:

- Click the "Mark Complete" button
- Enter the ID of the event you want to mark as completed

Exiting the Application:

- Click the "Exit" button to close the application

- If any time you want to skip entering something for the given prompts, you can always just click next, and it will let you move on without marking that information.

Class List:

Event (Abstract):

Role: This class defines the common properties and behaviors for all event types in the application. It includes methods for:

- Basic event information (title, description)
- Completion status
- Unique ID generation
- Notification handling

Meaning: Defines the essential properties that all events must have. As an abstract class, it can't be instantiated directly, but can be used as a template or blueprint for concrete class event types. Contains common fields: id, title, description, date, status, and category

Assignment:

Role: Represents the academic work assignments that need to be completed

Meaning: Extends the event class with specific features for tracking assignments, including course, priority level, and due dates

Meeting:

Role: Represents scheduled meetings in the system

Meaning: Extends the Event class to get meeting-specific information such as location and attendees

Priority (enum):

Role: Enumeration of priority levels for assignments

Meaning: Gets and adds the basic details of the importance of each event: Low, medium, high and urgent. Each level has associated notification behaviors and display properties.

EventStatus (enum):

Role: Tracks the current state of events

Meaning: An enumeration of the four possible statuses: Pending, in progress, completed, and cancelled

BasicNotification:

Role: Display event message and its source.

Meaning: Builds a notification using event message and event source and returns.

Notification:

Role: Represents an individual notification message for an event

Meaning: The class has all the information for alerting a user about an event. Contains the message, which is a reference to that event. The class provides methods to determine when the notification should be displayed and the process of it being displayed.

NotificationService:

Role: Manages the collection of notifications.

Meaning: Service keeps track of all the reminders for your events. It makes new notifications whenever you add an event

Schedule:

Role: Core data model for the PlanIt Pro application.

Meaning: This class maintains the collection of all events and provides methods to add, retrieve, and manipulate event data. It serves as the central data store and implements methods for:

- Adding new events
- Retrieving events by ID
- Listing all events
- Managing the notification service

ScheduleApp:

Role: This application provides a graphical user interface for managing events, tasks, and meetings using a simple and clean design.

Meaning:

- Create and manage assignments with priority levels
- Schedule meetings with location and attendee tracking
- Mark events as complete

- View all scheduled events in a readable format
- The application uses Java Swing for the GUI components

Team Responsibilities:

Aditi - UML Diagram/ReadMe File. Building relationships with classes and using inheritance to amplify them.

Ishanvi - ReadMe File/UML Diagrams. Learned about enum classes and applied them.

Gabby - ReadMe File/Coding of base classes with minimal information

Known Bugs/Workarounds:

- Miscommunication about pushing and pulling commits into GitHub
- Deciding on whether we wanted to do more console input or GUI

Key Learnings:

Credit List:

- Java Enums
- Custom Graphics Programming - Java Programming Tutorial
- GUI(html, other)
- GUI