

Yuan Jiayi - Project Portfolio for MisterMusik

1. About the project	1
2. Summary of contributions	1
2.1. Enhancement added:	1
2.2. Code contributed:	2
2.3. Other contributions:	2
3. Contributions to the User Guide	2
4. Contributions to the Developer Guide	2

1. About the project

My team of 5 computer engineering students were tasked with enhancing a program Duke for our CS2113T project. We chose to design a scheduler application for serious music students pursuing a professional music career as a western classical music performer. The program is designed to automate and streamline most of the process in scheduling and organisation of materials, allowing the students to focus more on the important aspects of their education.

This is what our project looks like: image:: *Figure 1. The graphical user interface for **MisterMusik**.*

My role is to design and write the codes for the **recurring event** and **reschedule** features. The following sections illustrate these enhancements in more detail, as well as the relevant documentation I have added to the user and developer guides in relation to these enhancements.

Note the following symbols and formatting used in this document:

TIP This symbol indicates important information.

reschedule : A grey highlight (called a mark-up) indicates that this is a command that can be inputted into the command line and executed by the application.

addRecurringEvent : Italic text indicates a component, object, class, or method.

2. Summary of contributions

This section shows a summary of my coding, documentation, and other helpful contributions to the team project.

2.1. Enhancement added:

I added the ability to add recurring events, reschedule the events in the list, and add contacts of

tutors.

2.1.1. Add recurring events

1. What it does: The `lesson` and `practice` commands allow the user to add recurring events happen in 16 weeks, using input format `lesson/practice <description> /dd-MM-yyyy HHmm HHmm /period(in days)`.
2. Justification: Some events are recurring, like some lessons happen once a week, or practices happen every two days. The command with `period` enables users to add recurring events easily, instead of adding them one by one.
3. Highlight:

2.1.2. Reschedule

1. What it does: The `reschedule` command allows the user to change the date and time of an event in the list.
2. Justification: In the event that users have made a mistake or changed their minds about the date and time, the `reschedule` command enables them to change the start and end date and time of the event in the list.

2.1.3. Add contacts

1. What it does:
2. Justification:

2.2. Code contributed:

Please click these links to see a sample of my code:

2.3. Other contributions:

1. Project management:
 - a. dsf
2. Enhancements to existing features:
3. Documentation:
4. Community:
 - a. Reviewed Pull Requests:

3. Contributions to the User Guide

4. Contributions to the Developer Guide