

# COMP2432 Group Project Design Documentation

Star Poon

March 6, 2015

## 1 Data Structure

### 1.1 struct Appointment

The starting time of the appointment is stored in the type *time\_t* which is defined in the standard C library `<time.h>`. This library provides some time manipulation and conversion functions that are useful in the system design.

The main issue is how should we store the duration of the appointment. The input of the appointment contains the duration of the appointment in the format of n.n hours. We have two choice of storing the ending time:

```
time_t start
float duration
```

Or we could use:

```
time_t start
time_t end
```

During the insertion, we need to check whether there are collisions between the existing appointments and the new appointment. We need to determinate if the new appointment is in between the start time and the end time of the existing appointments. It is more convenient to store the ending time instead of the duration so that we can simply use *difftime()*. If we stored the duration instead of the time, we need to do more calculation each time we insert a new appointment. For more details about the conflict checking, please referee to [2.1](#).

## 2 Algorithm

### 2.1 Check appointment conflict

Each time a new appointment is added, we need to check if there is a conflict between the existing and the new appointment.