Smart Classroom Manager

Algorithm – Web Service

Introduction

The algorithm takes the configuration and constraints as input and returns the booking records as output. The algorithm has two segments. In the first segment, it goes through all the booking patterns to calculate the interval date periods and generates optional parameters. In the second segment, the algorithm starts generating the booking records.

Segment 1

In this segment, the algorithm goes through all the booking pattern and determines

- The number of weeks to be used for generating the Amount of booking records based on the Interval for each booking pattern. If the Interval is 1, the algorithm will generate the Amount of booking records for each week inside the booking pattern From and To date period.
- The Minimum date and Maximum date for the algorithm to generate booking patterns. Though the Booking Period contains the actual From and To date for generating the booking records, It is not always efficient to use them. The algorithm selects the lowest From date from the given booking patterns as Minimum date and highest To date from booking patterns as Maximum date. By doing this, the algorithm eliminates the unnecessary dates from the process.
- Set the optional parameters such as Assigned Teacher, Course, Room, Room Type and Facility if not set in the booking pattern. These parameters are set by using the Booking ID from the Booking pattern.
- Access the database and fetch all the booking records created previously. These are used to check for collisions.
- Calculates the total number of booking records to be created.

Segment 2

In this segment, the algorithm starts generating the booking records. While generating the records it performs tasks such as selecting timeframes, selecting optimal bookings and checking for collisions.

The segment 2 is performed till the total number of booking records to create becomes 0. For each booking pattern, the algorithm performs the following steps.

1. Checks the Amount parameter. If it is 0, continue with next booking pattern.
2. For each WeeksFrame calculated using the Interval parameter, get the number of booking records created for this booking pattern. If it is equal to Amount parameter, go to next WeeksFrame. If not equal, continues with step 3
3. For each day in the WeeksFrame, set the currentBookingDate. Check if it is inside the PeriodStartDate and PeriodEndDate of the BookingPeriod. If it is inside, continue with step 4. Else create a booking record and set it as collision.
4. Check if the day of currentBookingDate is allowed in the AllowedWeekends. If the day is allowed, continue with step 5. Else create a booking record and set it as collision.
5. Check the Type parameter in the BookingPeriod. If it is true, the timeframes are selected within the boundary times set in the FlexibleTimeSlot. Else the timeframes defined in the StaticTimeSlots are used to create the booking records.
6. A Optimal Booking Patterns list is generated for the available time slot. If the current booking pattern is set as Optimal, then the timeframe is selected. If it is not an optimal booking pattern, then continue step 4 with the next day in the WeeksFrame.
7. Once a timeframe is selected, it is checked for collisions. A booking record is created if there no collision. If any collision occured, a new timeframe is selected.
8. This process is performed for each day in the WeeksFrame. If there is no timeframe to selected, a booking record is created and set as collision.
9. Finally the booking records are stored in the database.

Timeframe Selection

The timeframe selection varies for static and flexible types.

Static type:

For static type, the timeframe is selected from the StaticSlotsPerDay in BookingPeriod. The StaticSlotsPerDay is a list contianing slots for each day. The timeframe is selected from any one of the slot depending on the availability of the resources such as teacher, group and room.

Flexible type:

For flexible type, the timeframe is selected between the From and To boundary set in the FlexibleTimeframe for each day depending on the availability of the resources such as teacher, group and room. The lunch time is adjusted either before or after the time slot based on LunchTimeflexibility and the MaximumLunchTimeBuffer.

LunchTimeFlexibility in FlexibleTimeframe:

The lunch time is adjusted to allow creation of booking records in an optimal way. The algorithm creates a buffer of length specified in the MaximumLunchTimeBuffer before and after the lunch duration. So it is now possible to create booking records inside the buffer. For example, the LunchTimeFlexibility is 12:00 and the MaximumLunchTimeBuffer is 30 minutes. The algorithm creates a buffer from 11:30 to 12:30. If there are any booking pattern that can be created within the 12:30, then the lunch time is adjusted to 12:30. If there are no booking pattern and the last used timeframe falls inside the buffer, then end time of that timeframe is set as lunch time. For example, if the last used timeframe is 11:00 to 11:45, then the lunch time is moved to the front to start at 11:45.

NotAllowed Timeframe:

It is possible for a group to be set as not allowed for a duration in Not Allowed Timeframes. If the selected Timeframe collides with the Not Allowed TimeFrames for the given group and date, a new timeframe will be selected.

Optimal Booking Pattern Selection

The algorithm uses a simple and yet powerful process to select optimal booking patterns for a given slot. It maintains four baskets to store the booking patterns.The booking patterns are classified and stored into different baskets based on their lengths. The basket 1 is the highest priority and the basket 4 is the least priority. If the length of the booking pattern is equal to the size of the slot, then that booking pattern is stored in the basket 1. If the length of the booking pattern is smaller than the size of the slot, the algorithm will check if it is possible to create multiple without leaving any unused time. If possible, that booking pattern is stored in the basket 2. If it can create multiple and any time is unused, then that booking pattern is stored in the basket 3. If the booking can not be stored in any other basket, then it will be stored in basket 4. Finally the optimal booking pattern is returned based on the priority of the baskets. If the basket 1 contains any booking pattern then it will be returned. If basket 1 is empty, then the next basket is used. This continues till all the baskets are empty. When all the baskets are empty, it means no booking pattern is optimal for that slot.

CollisionReason

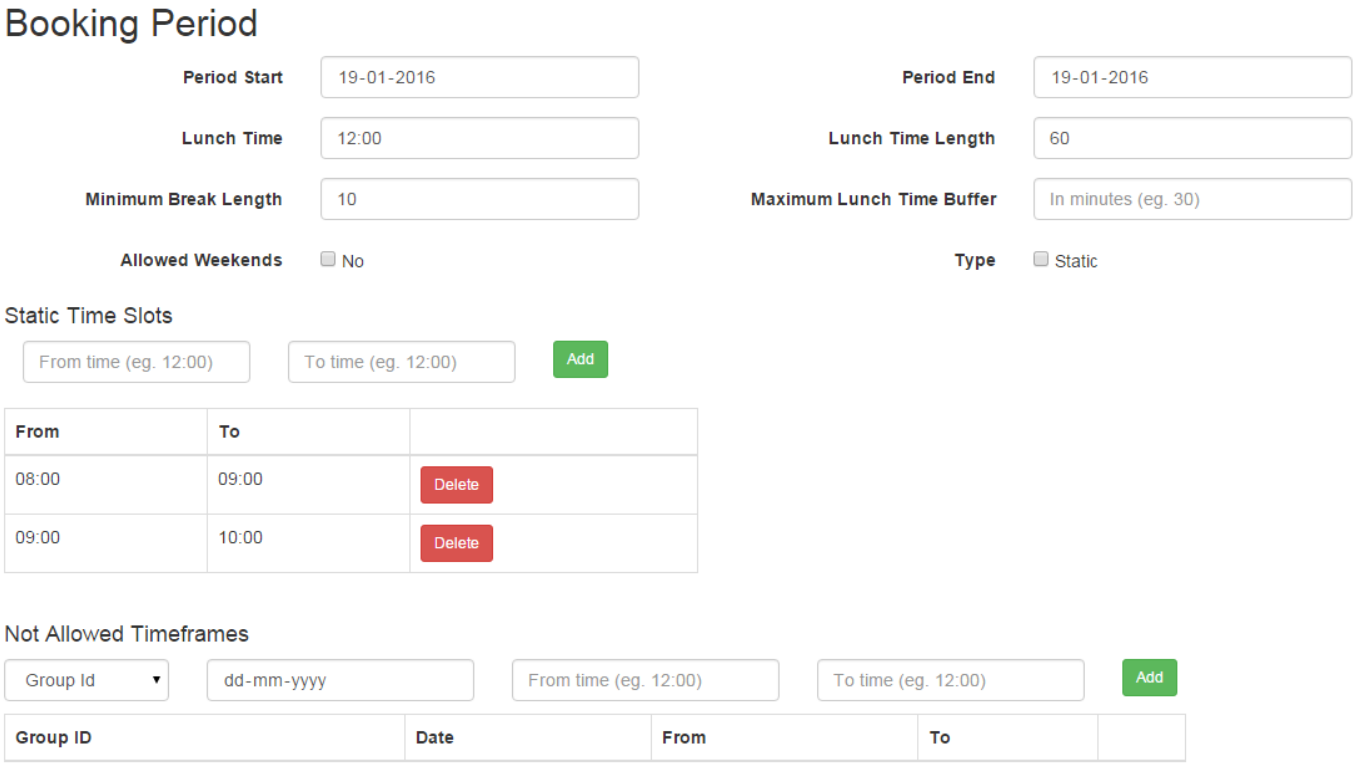
The algorithm has the ability to find multiple collisions.The algorithm uses flags to determine the collision reason. If any collision is detected while checking for collisions, the flags are turned on based on the collision. The collsions may occur for teacher, group, room. If there is no timeframe available, then it is also considered as collision.

Scenario 1 (Static type)

This scenario deals with the static type booking records in general.

Booking Period Selection

The booking period was selected for only one day to show how the algorithm selects the optimal booking pattern. The Minumum Break Length field specifies the length (in minutes) for the break in between booking records. Two time slots are created in the Static Time Slots. The first slot is from 08:00 to 09:00 and second slot is from 09:00 to 10:00.



Bookings Patterns

The booking patterns were created by giving the course Id,length of the booking, amount,From date,To date,Interval and room setting as optional parameter. From date and To date should be in between the period start date and period end date.

booking 1

length : 25 mins

Amount : 1

Interval : 1

From : 19-01-2016

To : 19-01-2016

booking 2

length : 60 mins

Amount : 1

Interval : 1

From : 19-01-2016

To : 19-01-2016

booking 3

Length : 15 mins

Amount : 1

Interval : 1

From : 19-01-2016

To : 19-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1	1	1	25	✔	1	1	2016-01-19	2016-01-19	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete
2	2	1	60	✔	1	1	2016-01-19	2016-01-19	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete
3	3	1	15	✔	1	1	2016-01-19	2016-01-19	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete

Booking Records

The booking 2 is selected as optimal booking for the slot 08:00 to 09:00 since it takes the full slot. In the next slot, there is 50 minutes free slot after including 10 minutes of minimum break length. It is used for booking 1 and booking 2 combinedly.

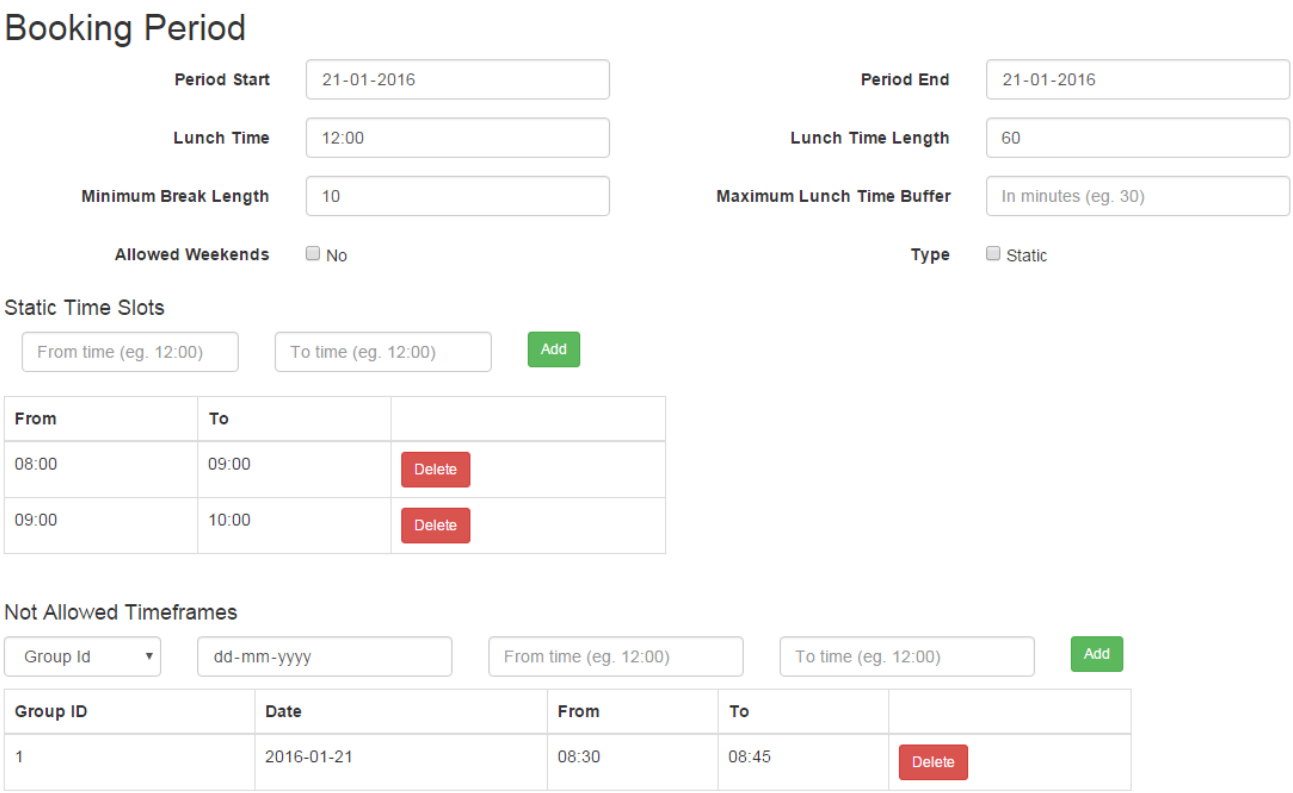
Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	2	2	1	2016-01-19	08:00	09:00	60 mins.	1	1	✔ No		1
2	1	1	1	2016-01-19	09:10	09:35	25 mins.	1	1	✔ No		1
3	3	3	1	2016-01-19	09:45	10:00	15 mins.	1	1	✔ No		1

Scenario 2 (Static type with Not Allowed Timeframes)

This scenario deals with the static type booking records with Not allowed Timeframes.

Booking Period Selection

Two static slots were created for a single day (21-01-2016). The first one is from 08:00 to 09:00 and the second one is from 09:00 to 10:00.Not allowed timeframe was created for a single day (21-01-2016) starts from 08:30 to 08:45 for group Id (1).



Booking Patterns

The booking pattern was created for a length time of 25 minutes for a single day.

booking 1

Length : 25 mins

Amount : 5

Interval : 1

From : 21-01-2016

To : 21-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1		1	25	✔	5	1	2016-01-21	2016-01-21	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete

Booking Records

From the table it shows that the booking was not created from 08:35 to 9:00 ,since the frame is in Not allowed Time frame from 08:30 to 08:45.

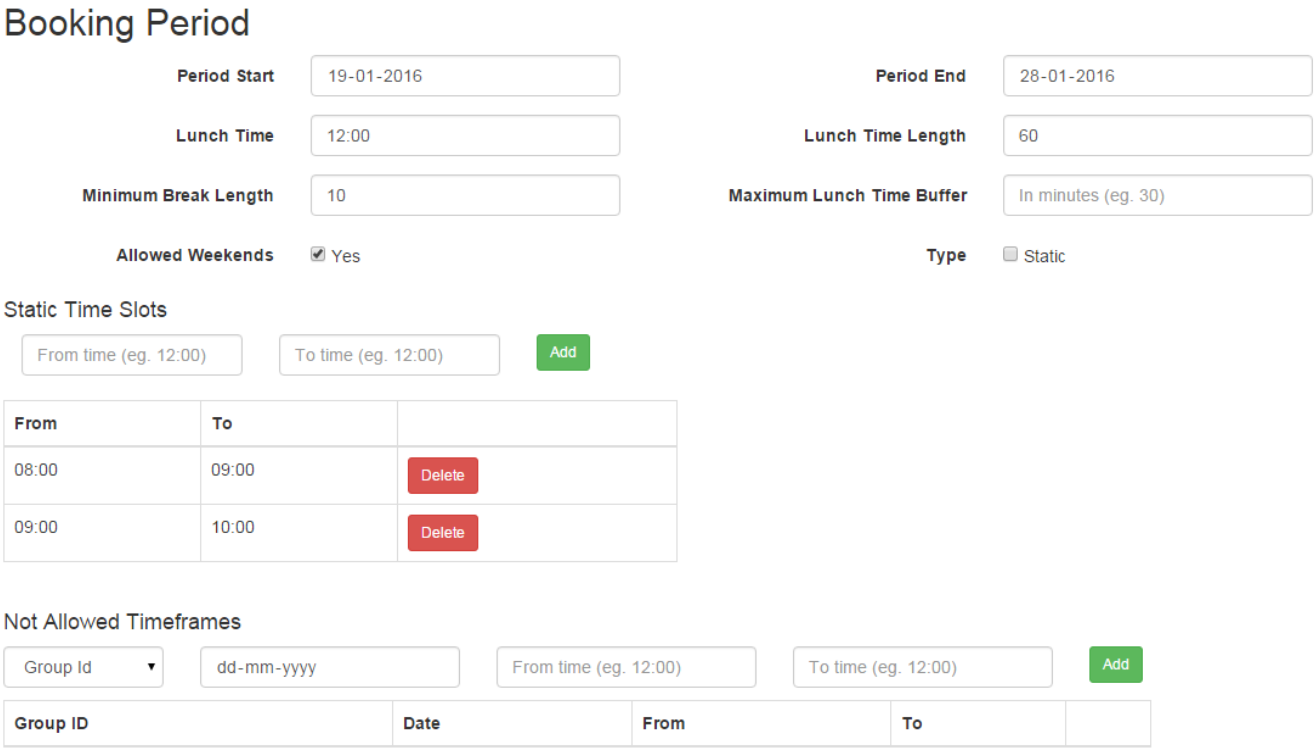
Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-21	08:00	08:25	25 mins.	1	1	✔ No		1
2	1	1	1	2016-01-21	09:00	09:25	25 mins.	1	1	✔ No		1
3	1	1	1	2016-01-21	09:35	10:00	25 mins.	1	1	✔ No		1
4	1	1	1	2016-01-21	00:00	00:00	25 mins.	1	1	✖ Yes	Group,Teacher,Room,Timeframe not available.	1
5	1	1	1	2016-01-21	00:00	00:00	25 mins.	1	1	✖ Yes	Group,Teacher,Room,Timeframe not available.	1

Scenario 3 (Static type with Allowed Weekends)

This scenario deals with the static type booking records with Allowed Weekends.

Booking Period Selection

From the figure you can see that Allowed Weekends field is set to be true for creating booking records in Allowed Weekends.The booking period is selected from Period Start (19-01-2016) to Period End (28-01-2016) of 2 weeks to work in Allowed Weekends. The Minumum Break Length field specifies the length (in minutes) for the break in between booking records.Two time slots are created in the Static Time Slots. The first slot is from 08:00 to 09:00 and second slot is from 09:00 to 10:00.



Bookings Patterns

The booking patterns were created by giving the course Id,length of the booking, amount,From date,To date,Interval and room setting as optional parameter. From date and To date should be in between the period start date and period end date.The amount value is 3 of which the total booking records count will be 12.

booking 1

length : 25 mins

Amount : 3

Interval : 1

From : 19-01-2016

To : 28-01-2016

booking 2

length : 20 mins

Amount : 3

Interval : 1

From : 19-01-2016

To : 28-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1	1	1	25	✔	3	1	2016-01-19	2016-01-28	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete
2	2	1	20	✔	3	1	2016-01-19	2016-01-28	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete

Booking Records

From the table you can see that 4 booking records were created on 24-01-2015(sunday) which was in Allowed Weekends.

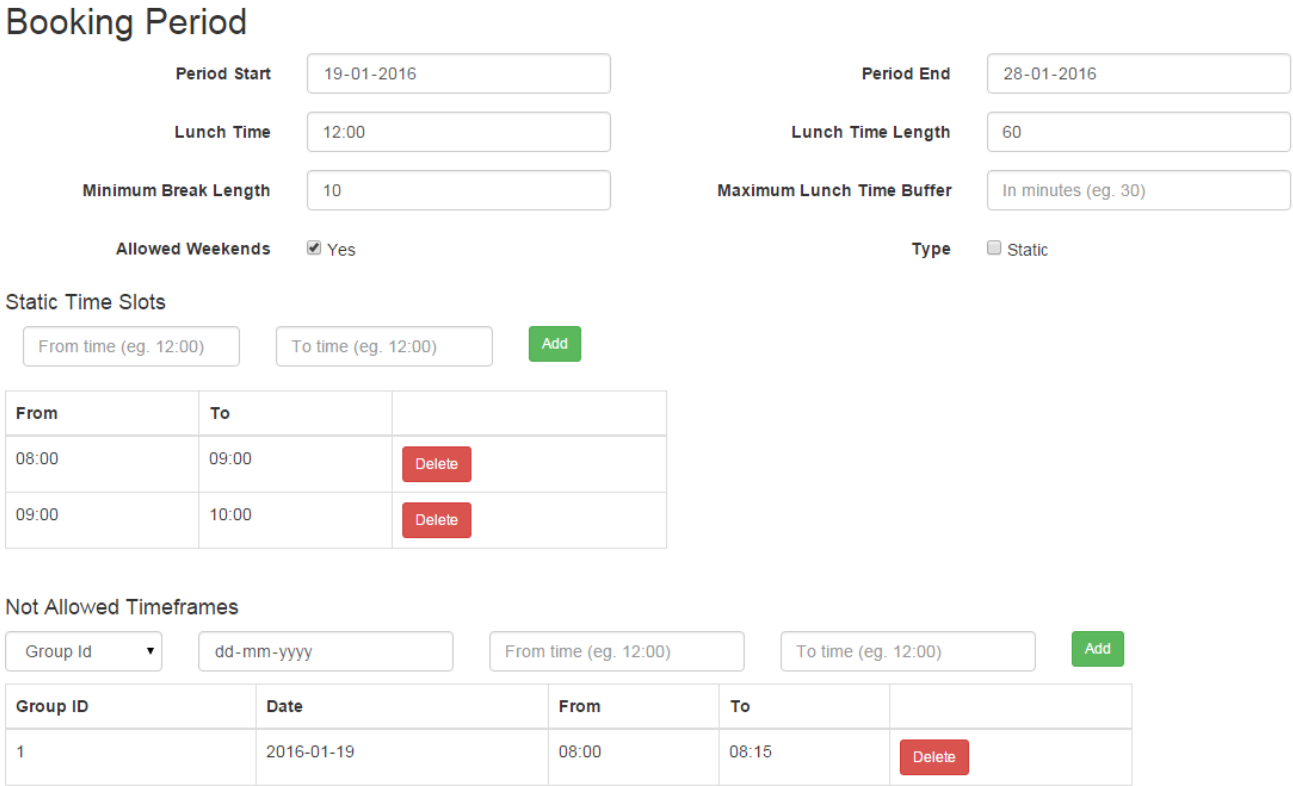
Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-19	08:00	08:25	25 mins.	1	1	✔ No		1
2	1	1	1	2016-01-19	08:35	09:00	25 mins.	1	1	✔ No		1
3	1	1	1	2016-01-20	08:00	08:25	25 mins.	1	1	✔ No		1
4	1	1	1	2016-01-24	08:00	08:25	25 mins.	1	1	✔ No		1
5	1	1	1	2016-01-24	08:35	09:00	25 mins.	1	1	✔ No		1
6	1	1	1	2016-01-25	08:00	08:25	25 mins.	1	1	✔ No		1
7	2	2	1	2016-01-19	09:10	09:30	20 mins.	1	1	✔ No		1
8	2	2	1	2016-01-19	09:40	10:00	20 mins.	1	1	✔ No		1
9	2	2	1	2016-01-20	08:35	08:55	20 mins.	1	1	✔ No		1
10	2	2	1	2016-01-24	09:10	09:30	20 mins.	1	1	✔ No		1
11	2	2	1	2016-01-24	09:40	10:00	20 mins.	1	1	✔ No		1
12	2	2	1	2016-01-25	08:35	08:55	20 mins.	1	1	✔ No		1

Scenario 4 (Static type with Not Allowed Timeframe and Allowed Weekends)

This scenario deals with the static type booking records with Not allowed Timeframes and with Allowed weekends.

Booking Period Selection

The booking period was selected from Period Start (19-01-2016) to Period End (28-01-2016) of 2 weeks to work in Allowed Weekends. The Minumum Break Length field specifies the length (in minutes) for the break in between booking records. From the figure you can see that Allowed Weekends field is set to be true and Not Allowed Timeframes Slot is filled up with Group Id, Date(19-01-2016), FromTime(08:00), ToTime(08:15) to demonstrate that the booking record will not be created in this particular duration.



Bookings Patterns

The booking patterns were created by giving the course Id,length of the booking, amount,From date,To date,Interval and room setting as optional parameter. From date and To date should be in between the period start date and period end date.The amount value is 3 of which the total booking records count will be 12.

booking 1

length : 25 mins
Amount : 3
Interval : 1
From : 19-01-2016
To : 28-01-2016

booking 2

length : 20 mins
Amount : 3
Interval : 1
From : 19-01-2016
To : 28-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1	1	1	25	✔	3	1	2016-01-19	2016-01-28	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete
2	2	1	20	✔	3	1	2016-01-19	2016-01-28	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete

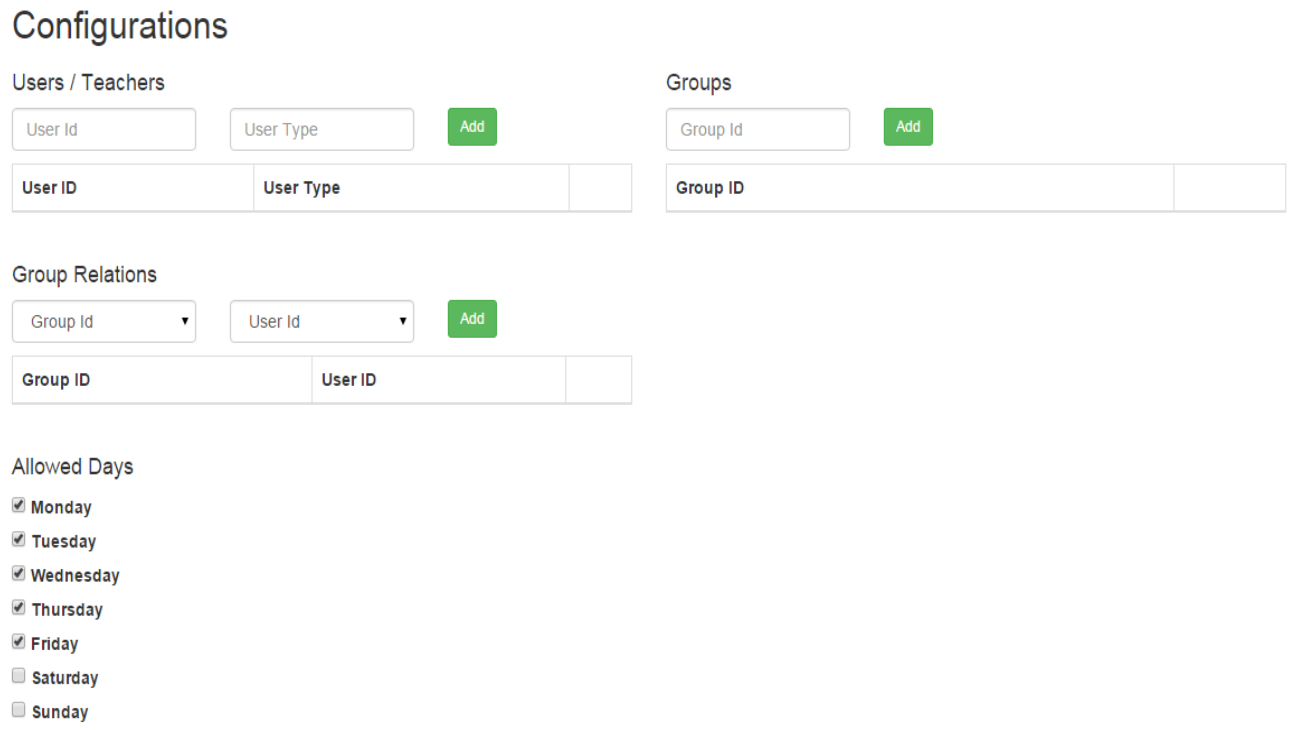
Booking Records

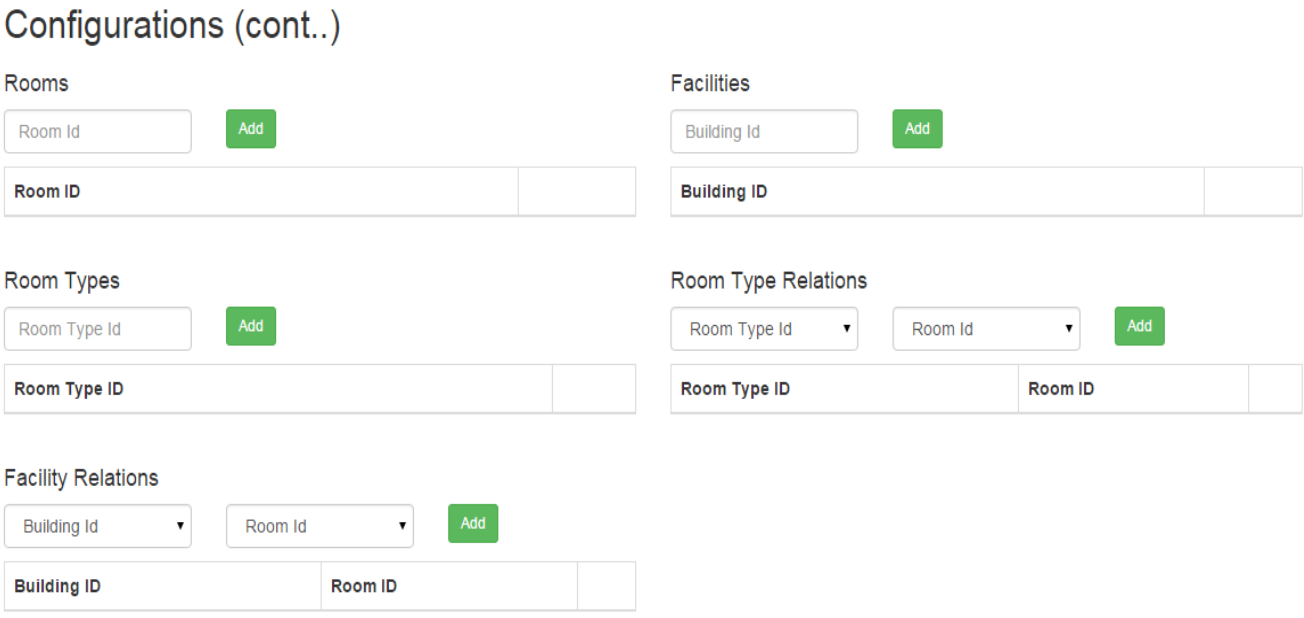
From below figure you can see that the booking record was not created from 08:00 to 08:15 on 19-01-2016 as we have given these parameters as Not Allowed Timeframe Slot.

Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-19	08:25	08:50	25 mins.	1	1	✔ No		1
2	1	1	1	2016-01-19	09:00	09:25	25 mins.	1	1	✔ No		1
3	1	1	1	2016-01-19	09:35	10:00	25 mins.	1	1	✔ No		1
4	1	1	1	2016-01-24	08:00	08:25	25 mins.	1	1	✔ No		1
5	1	1	1	2016-01-24	08:35	09:00	25 mins.	1	1	✔ No		1
6	1	1	1	2016-01-25	08:00	08:25	25 mins.	1	1	✔ No		1
7	2	2	1	2016-01-20	08:00	08:20	20 mins.	1	1	✔ No		1
8	2	2	1	2016-01-20	08:30	08:50	20 mins.	1	1	✔ No		1
9	2	2	1	2016-01-20	09:00	09:20	20 mins.	1	1	✔ No		1
10	2	2	1	2016-01-24	09:10	09:30	20 mins.	1	1	✔ No		1
11	2	2	1	2016-01-24	09:40	10:00	20 mins.	1	1	✔ No		1
12	2	2	1	2016-01-25	08:35	08:55	20 mins.	1	1	✔ No		1

Scenario 5 (Static Type without Input parameters)

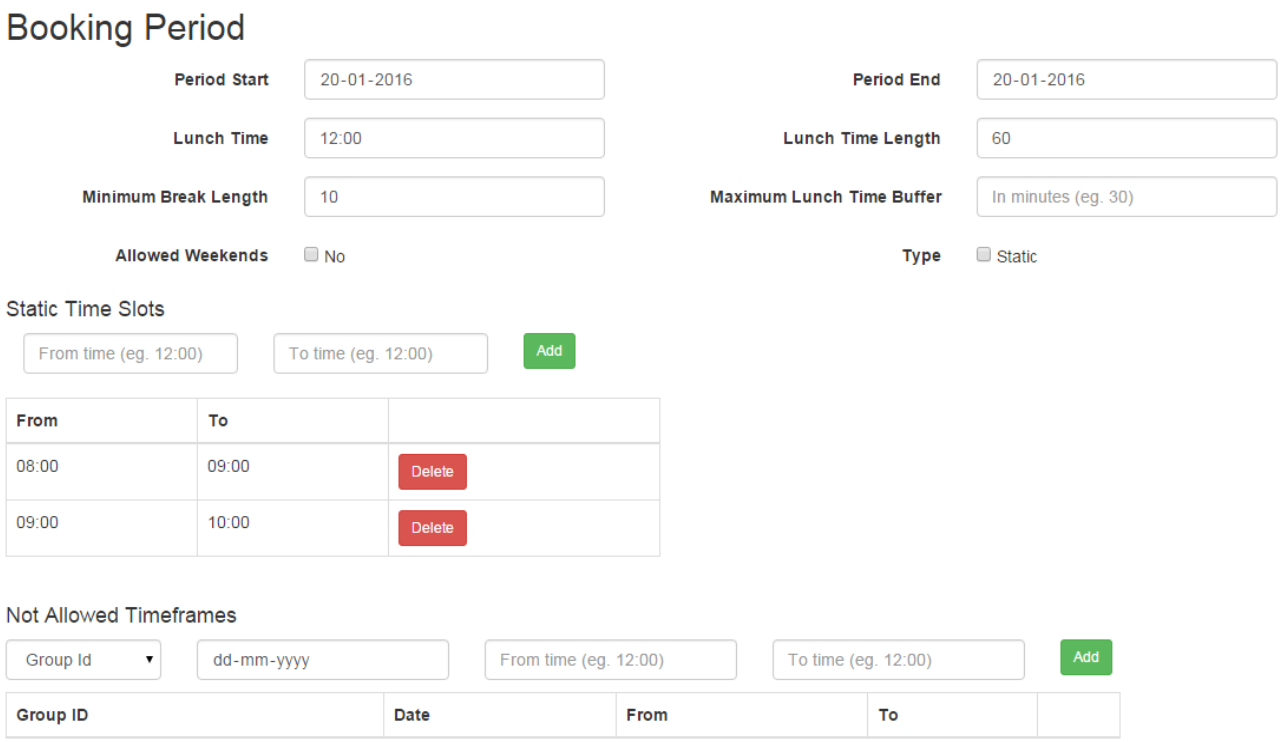
This scenario deals with the static type booking records without Input Parameters such as rooms,teachers,groups and building relations. From the figure it shows that the Input parameters are not mandatory for creating booking records.





Booking Period Selection

The booking period is selected for only one day to show how the algorithm works without providing input parameters. Two time slots are created in the Static Time Slots. The first slot is from 08:00 to 09:00 and second slot is from 09:00 to 10:00.



Bookings Patterns

Here the booking pattern was created without the input parameters such as course Id,Group Id,RoomType Relations, and Facility Relations.From the table it clears that input parameters were not provided.

booking 1

Length : 20 mins

Amount : 4

Interval : 1

From : 20-01-2016

To : 20-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1			20	✖	4	1	2016-01-20	2016-01-20	Rooms : Facilities : Room Types : Teachers :	Delete

Booking Records

As per the amount and interval values it created 4 booking records in a single day without input parameters. The values for Room Id, Building Id,Teachers Id,Group Id,Course Id were filled up with booking Id as default value.

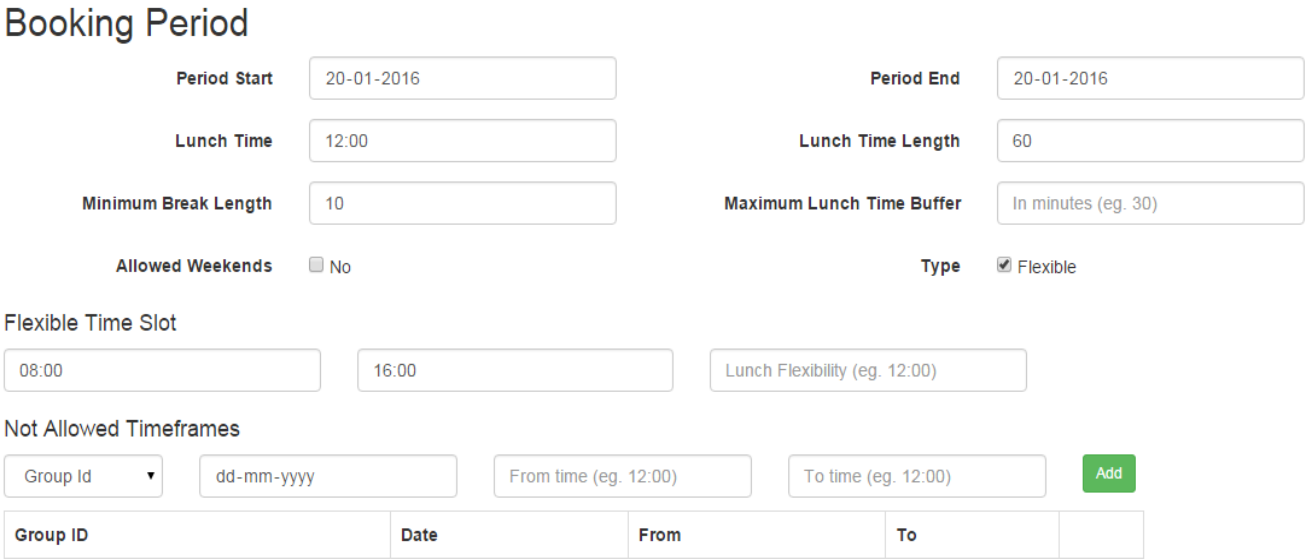
Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-20	08:00	08:20	20 mins.	1	1	✔ No		1
2	1	1	1	2016-01-20	08:30	08:50	20 mins.	1	1	✔ No		1
3	1	1	1	2016-01-20	09:00	09:20	20 mins.	1	1	✔ No		1
4	1	1	1	2016-01-20	09:30	09:50	20 mins.	1	1	✔ No		1

Scenario 6 (Flexible Type)

This scenario deals with the flexible type booking records in general.

Booking Period Selection

The booking period was selected for only one day to show how the algorithm works with the flexible type booking records. The Minumum Break Length field specifies the length (in minutes) for the break in between booking records.Flexible slot was created from 08:00 to 16:00 for one day (20-01-2016) .



Bookings Patterns

Two bookings were created with a LengthTime of 25 minutes to generate flexible type booking records from 08:00 to 16:00.

booking 1

Length : 25 mins
Amount : 10
Interval : 1
From : 20-01-2016
To : 20-01-2016

booking 2

Length : 25 mins
Amount : 10
Interval : 1
From : 20-01-2016
To : 20-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1	1	1	25	✔	10	1	2016-01-20	2016-01-20	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete
2	2	2	25	✔	10	1	2016-01-20	2016-01-20	Rooms : 2, Facilities : 1, Room Types : 2, Teachers : 2,	Delete

Booking Records

Based on the amount value from the booking patterns it generates 20 flexible type booking records. The LunchTime is from 12:00 to 13:00 of which the booking was not created,since there is no buffer time length specified.

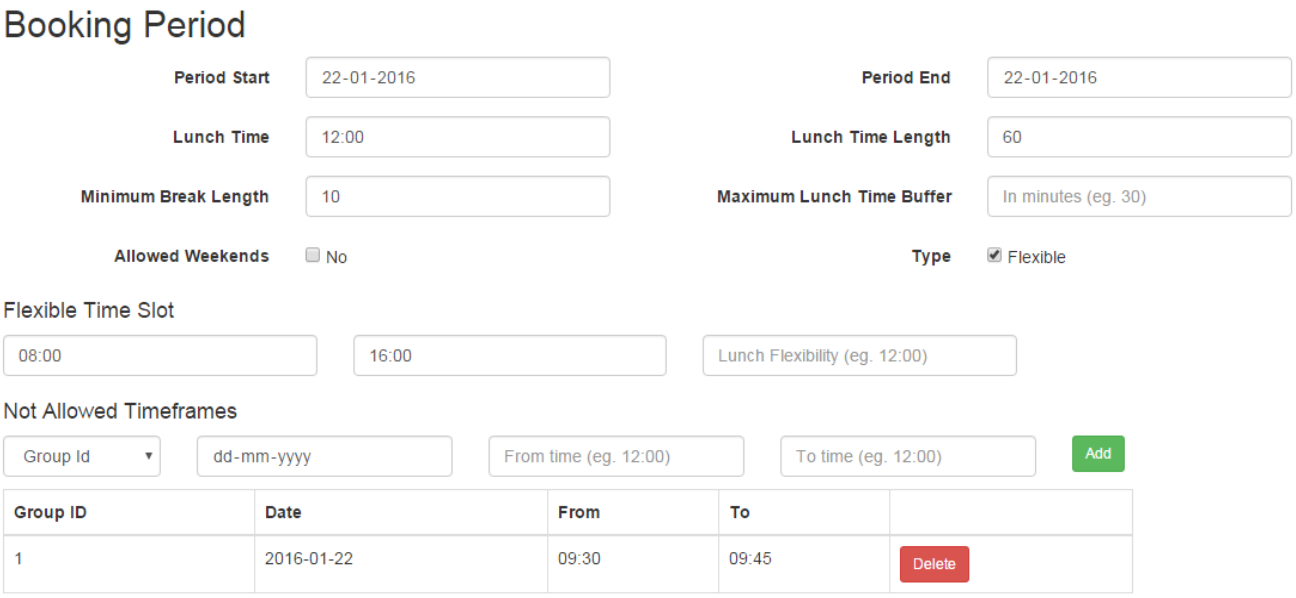
Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-20	08:00	08:25	25 mins.	1	1	✔ No		1
2	2	2	2	2016-01-20	08:00	08:25	25 mins.	1	2	✔ No		2
3	1	1	1	2016-01-20	08:35	09:00	25 mins.	1	1	✔ No		1
4	2	2	2	2016-01-20	08:35	09:00	25 mins.	1	2	✔ No		2
5	1	1	1	2016-01-20	09:10	09:35	25 mins.	1	1	✔ No		1
6	2	2	2	2016-01-20	09:10	09:35	25 mins.	1	2	✔ No		2
7	1	1	1	2016-01-20	09:45	10:10	25 mins.	1	1	✔ No		1
8	2	2	2	2016-01-20	09:45	10:10	25 mins.	1	2	✔ No		2
9	1	1	1	2016-01-20	10:20	10:45	25 mins.	1	1	✔ No		1
10	2	2	2	2016-01-20	10:20	10:45	25 mins.	1	2	✔ No		2
11	1	1	1	2016-01-20	10:55	11:20	25 mins.	1	1	✔ No		1
12	2	2	2	2016-01-20	10:55	11:20	25 mins.	1	2	✔ No		2
13	1	1	1	2016-01-20	11:30	11:55	25 mins.	1	1	✔ No		1
14	2	2	2	2016-01-20	11:30	11:55	25 mins.	1	2	✔ No		2
15	1	1	1	2016-01-20	13:00	13:25	25 mins.	1	1	✔ No		1
16	2	2	2	2016-01-20	13:00	13:25	25 mins.	1	2	✔ No		2
17	1	1	1	2016-01-20	13:35	14:00	25 mins.	1	1	✔ No		1
18	2	2	2	2016-01-20	13:35	14:00	25 mins.	1	2	✔ No		2
19	1	1	1	2016-01-20	14:10	14:35	25 mins.	1	1	✔ No		1
20	2	2	2	2016-01-20	14:10	14:35	25 mins.	1	2	✔ No		2

Scenario 7 (Flexible type with Not allowed Time frames)

This scenario deals with the flexible type booking records with Not allowed Time frames.

Booking Period Selection

The booking period was selected for only one day (22-01-2016) with a flexible slot from 08:00 to 16:00. Not allowed Time frame was created for a single day(22-01-2016) from 09:30 to 09:45 for the group Id (1).



Booking Patterns

Two booking patterns were created for flexible type booking records.The first one is for the length of 25 minutes and the second one is for the length of 20 minutes.

booking 1

Length : 25 mins
Amount : 5
Interval : 1
From : 22-01-2016
To : 22-01-2016

booking 2

Length : 20 mins

Amount : 5

Interval : 1

From : 22-01-2016

To : 22-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1	1	1	25	✔	5	1	2016-01-22	2016-01-22	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete
2	2	1	20	✔	5	1	2016-01-22	2016-01-22	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete

Booking records

From the table it shows that booking record was not created for the period in between 09:30 to 09:45 , since the collision happens with not allowed time frame.

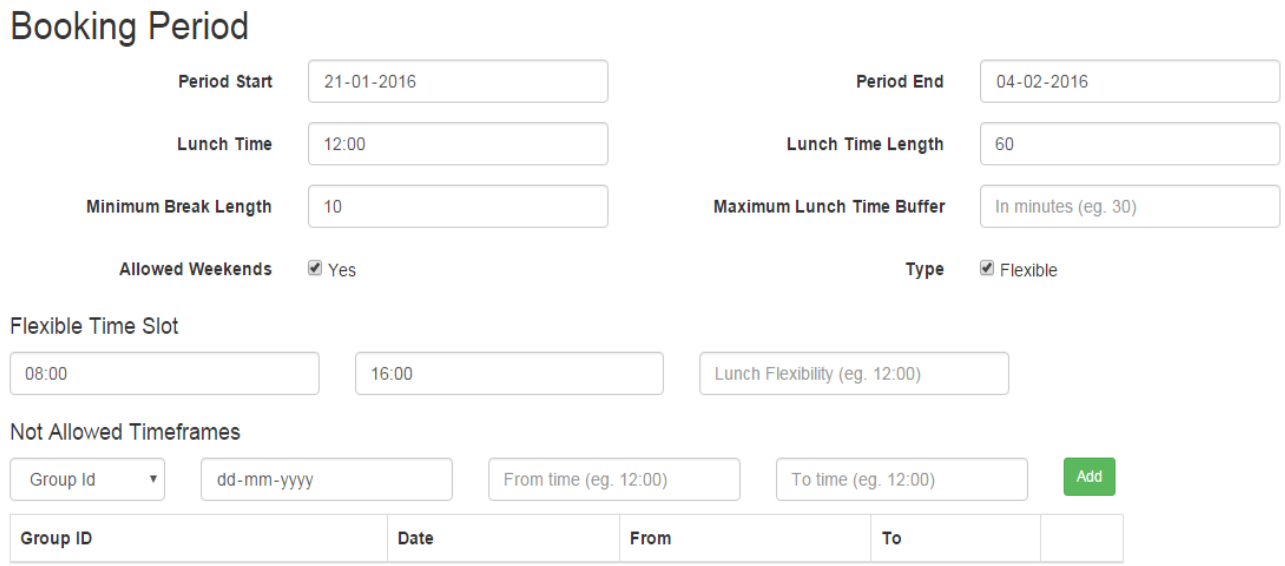
Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-22	08:00	08:25	25 mins.	1	1	✔ No		1
2	2	2	1	2016-01-22	08:35	08:55	20 mins.	1	1	✔ No		1
3	1	1	1	2016-01-22	00:00	00:00	25 mins.	1	1	✖ Yes	Group,Teacher,Room not available.	1
4	2	2	1	2016-01-22	09:05	09:25	20 mins.	1	1	✔ No		1
5	1	1	1	2016-01-22	09:55	10:20	25 mins.	1	1	✔ No		1
6	2	2	1	2016-01-22	10:30	10:50	20 mins.	1	1	✔ No		1
7	1	1	1	2016-01-22	11:00	11:25	25 mins.	1	1	✔ No		1
8	2	2	1	2016-01-22	00:00	00:00	20 mins.	1	1	✖ Yes	Group,Teacher,Room,Timeframe not available.	1
9	1	1	1	2016-01-22	11:35	12:00	25 mins.	1	1	✔ No		1
10	2	2	1	2016-01-22	13:00	13:20	20 mins.	1	1	✔ No		1

Scenario 8 (Flexible Type with Allowed Weekends)

This scenario deals with the flexible type booking records with allowed weekends.

Booking Period Selection

The booking period was selected from Period Start (19-01-2016) to Period End (04-02-2016) of 3 weeks to work in Allowed Weekends. The Minumum Break Length field specifies the length (in minutes) for the break in between booking records.



Bookings patterns

Two booking patterns were created with a LengthTime of 25 minutes.Both of the patterns starts from 21-01-2016 to 04-02-2016.

booking 1

Length : 25 mins

Amount : 3

Interval : 1

From : 21-01-2016

To : 04-02-2016

booking 2

Length : 25 mins

Amount : 3

Interval : 1

From : 21-01-2016

To : 04-02-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1	1	1	25	✔	3	1	2016-01-21	2016-02-04	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete
2	2	2	25	✔	3	1	2016-01-21	2016-02-04	Rooms : 2, Facilities : 2, Room Types : 2, Teachers : 2,	Delete

Booking records

The Algorithm generates a total of 18 records based on the amount value and interval for the 3 weeks from 21-01-2016 to 04-02-2016. From the table it shows the booking record was created on sunday(24-01-2016) which was in Allowed Weekends.

Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-21	08:00	08:25	25 mins.	1	1	✔ No		1
2	2	2	2	2016-01-21	08:00	08:25	25 mins.	2	2	✔ No		2
3	1	1	1	2016-01-21	08:35	09:00	25 mins.	1	1	✔ No		1
4	2	2	2	2016-01-21	08:35	09:00	25 mins.	2	2	✔ No		2
5	1	1	1	2016-01-21	09:10	09:35	25 mins.	1	1	✔ No		1
6	2	2	2	2016-01-21	09:10	09:35	25 mins.	2	2	✔ No		2
7	1	1	1	2016-01-24	08:00	08:25	25 mins.	1	1	✔ No		1
8	2	2	2	2016-01-24	08:00	08:25	25 mins.	2	2	✔ No		2
9	1	1	1	2016-01-24	08:35	09:00	25 mins.	1	1	✔ No		1
10	2	2	2	2016-01-24	08:35	09:00	25 mins.	2	2	✔ No		2
11	1	1	1	2016-01-24	09:10	09:35	25 mins.	1	1	✔ No		1
12	2	2	2	2016-01-24	09:10	09:35	25 mins.	2	2	✔ No		2
13	1	1	1	2016-01-31	08:00	08:25	25 mins.	1	1	✔ No		1
14	2	2	2	2016-01-31	08:00	08:25	25 mins.	2	2	✔ No		2
15	1	1	1	2016-01-31	08:35	09:00	25 mins.	1	1	✔ No		1
16	2	2	2	2016-01-31	08:35	09:00	25 mins.	2	2	✔ No		2
17	1	1	1	2016-01-31	09:10	09:35	25 mins.	1	1	✔ No		1
18	2	2	2	2016-01-31	09:10	09:35	25 mins.	2	2	✔ No		2

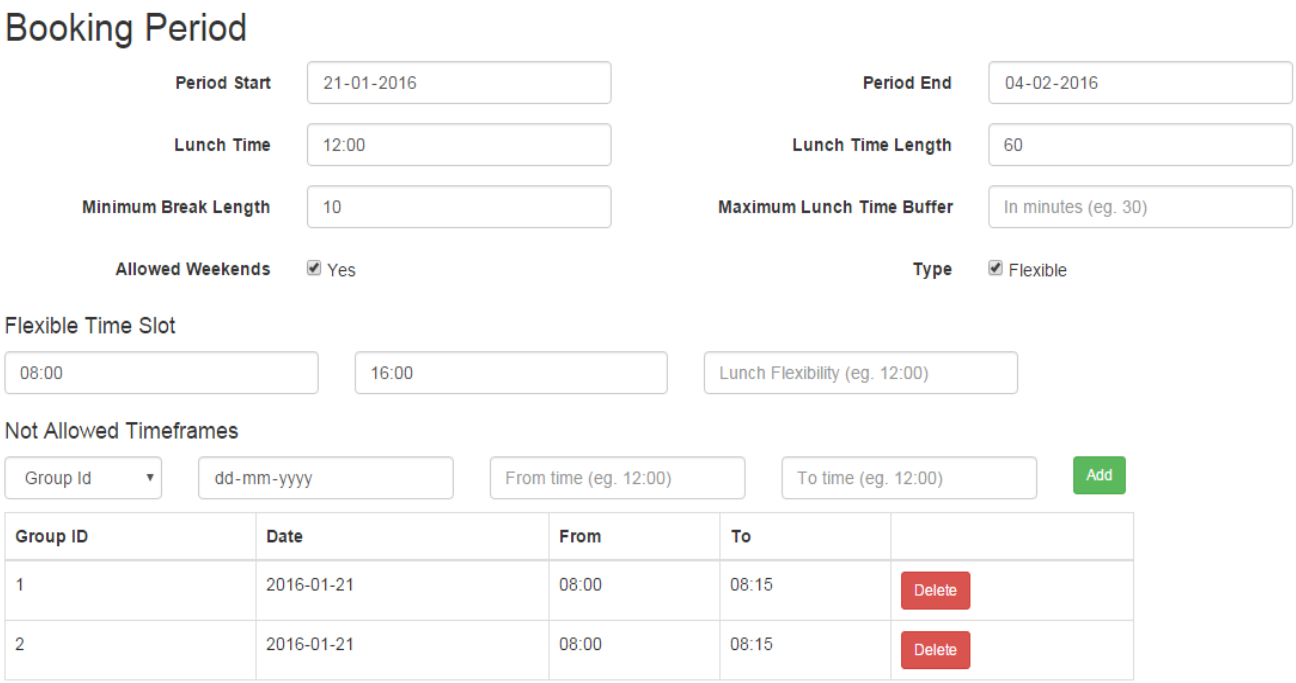
Scenario 9 (Flexible Type with Not Allowed TimeFrame and AllowedWeekends)

This scenario deals with the flexible type booking records with Not allowed Timeframe and

Allowed Weekends.

Booking Period Selection

The booking period is selected from Period Start (19-01-2016) to Period End (04-02-2016) of 3 weeks to work in Allowed Weekends. The Minumum Break Length field specifies the length (in minutes) for the break in between booking records.Not Allowed Timeframe is filled up with 2 frames.The firstframe is for the Group Id (1) and it starts from 08:00 to 08:15 on 21-01-2016.The secondframe is for the Group Id (2) and it starts from 08:00 to 08:15 on 21-01-2016.



Bookings patterns

Two booking patterns were created with a LengthTime of 25 minutes.Both of the patterns starts from 21-01-2016 to 04-02-2016.

booking 1

Length : 25 mins
Amount : 3
Interval : 1
From : 21-01-2016
To : 04-02-2016

booking 2

Length : 25 mins
Amount : 10

Interval : 1

From : 21-01-2016

To : 04-02-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1	1	1	25	✔	3	1	2016-01-21	2016-02-04	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete
2	2	2	25	✔	3	1	2016-01-21	2016-02-04	Rooms : 2, Facilities : 2, Room Types : 2, Teachers : 2,	Delete

Booking records

The Algorithm generates a total of 18 records based on the amount value and interval for the 3 weeks from 21-01-2016 to 04-02-2016.From the table the first and second records shows that the booking was created from 08:25 to 08:50 for a lengthtime of 25 minutes.

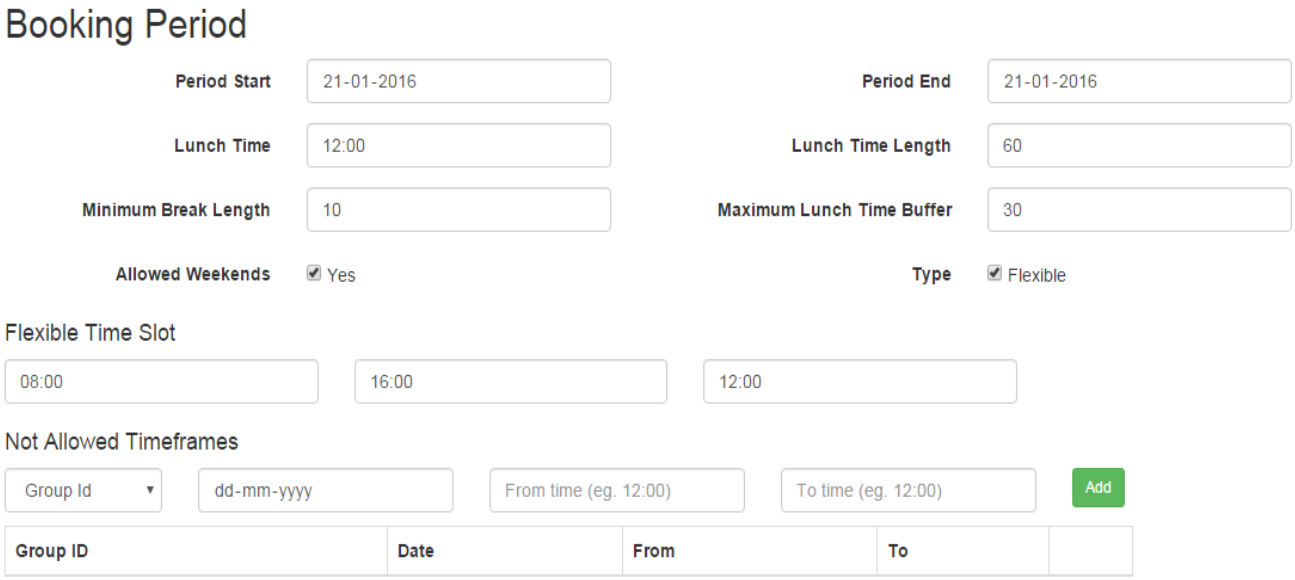
Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-21	08:25	08:50	25 mins.	1	1	✔ No		1
2	2	2	2	2016-01-21	08:25	08:50	25 mins.	2	2	✔ No		2
3	1	1	1	2016-01-21	09:00	09:25	25 mins.	1	1	✔ No		1
4	2	2	2	2016-01-21	09:00	09:25	25 mins.	2	2	✔ No		2
5	1	1	1	2016-01-21	09:35	10:00	25 mins.	1	1	✔ No		1
6	2	2	2	2016-01-21	09:35	10:00	25 mins.	2	2	✔ No		2
7	1	1	1	2016-01-24	08:00	08:25	25 mins.	1	1	✔ No		1
8	2	2	2	2016-01-24	08:00	08:25	25 mins.	2	2	✔ No		2
9	1	1	1	2016-01-24	08:35	09:00	25 mins.	1	1	✔ No		1
10	2	2	2	2016-01-24	08:35	09:00	25 mins.	2	2	✔ No		2
11	1	1	1	2016-01-24	09:10	09:35	25 mins.	1	1	✔ No		1
12	2	2	2	2016-01-24	09:10	09:35	25 mins.	2	2	✔ No		2
13	1	1	1	2016-01-31	08:00	08:25	25 mins.	1	1	✔ No		1
14	2	2	2	2016-01-31	08:00	08:25	25 mins.	2	2	✔ No		2
15	1	1	1	2016-01-31	08:35	09:00	25 mins.	1	1	✔ No		1
16	2	2	2	2016-01-31	08:35	09:00	25 mins.	2	2	✔ No		2
17	1	1	1	2016-01-31	09:10	09:35	25 mins.	1	1	✔ No		1
18	2	2	2	2016-01-31	09:10	09:35	25 mins.	2	2	✔ No		2

Scenario 10 (Flexible type with LunchTime Flexibility and Buffer Time)

This scenario deals with the flexible type booking records with Lunch Time Flexibility and Buffer Time.

Booking Period Selection

The booking period was selected for only one day to show how the algorithm works with lunch time buffer. Flexible slot starts from 08:00 to 16:00 with lunchtime flexibility as 12:00.Lunch Time Buffer is set to 30 minutes to adjust the Lunch timing.



Bookings patterns

The booking pattern is created for a single day (21-01-2016) with a lengthtime of 25 minutes.

booking 1

Length : 25 mins

Amount : 10

Interval : 1

From : 14-01-2016

To : 14-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1	1	1	25	✔	10	1	2016-01-21	2016-01-21	Rooms : 1, Facilities : 1, Room Types : 1, Teachers : 1,	Delete

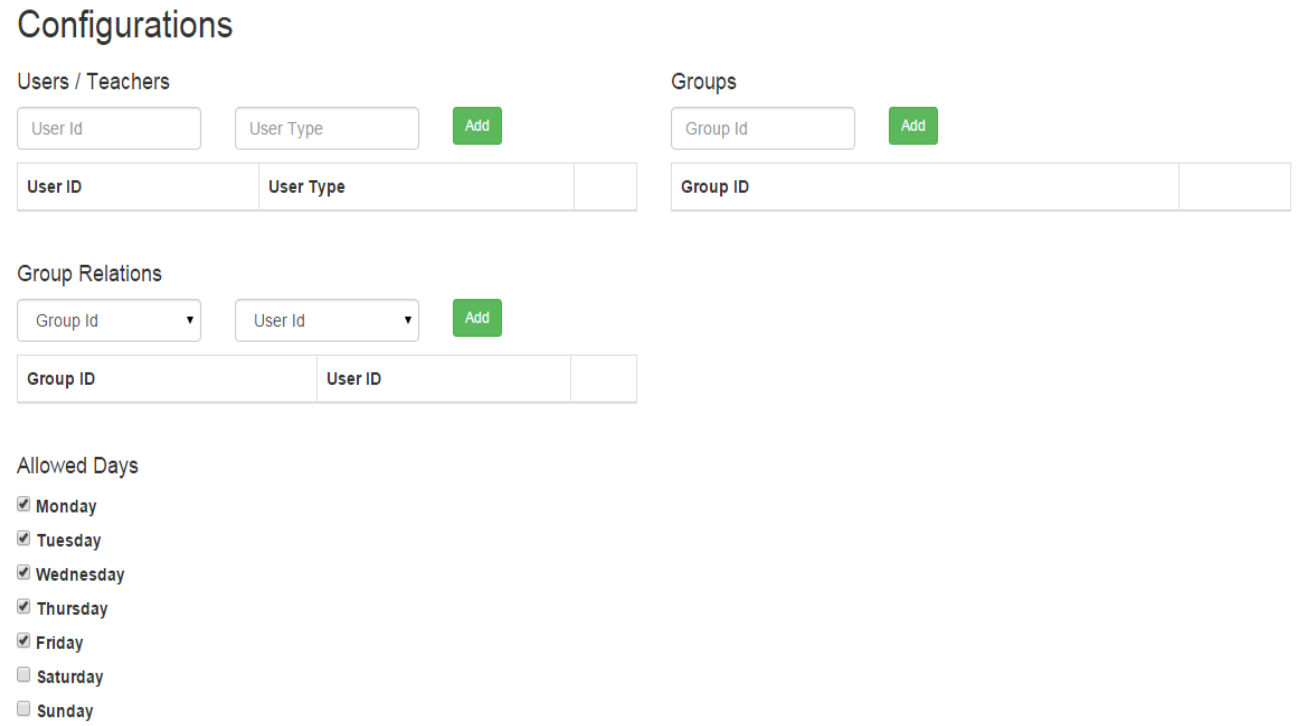
Booking records created

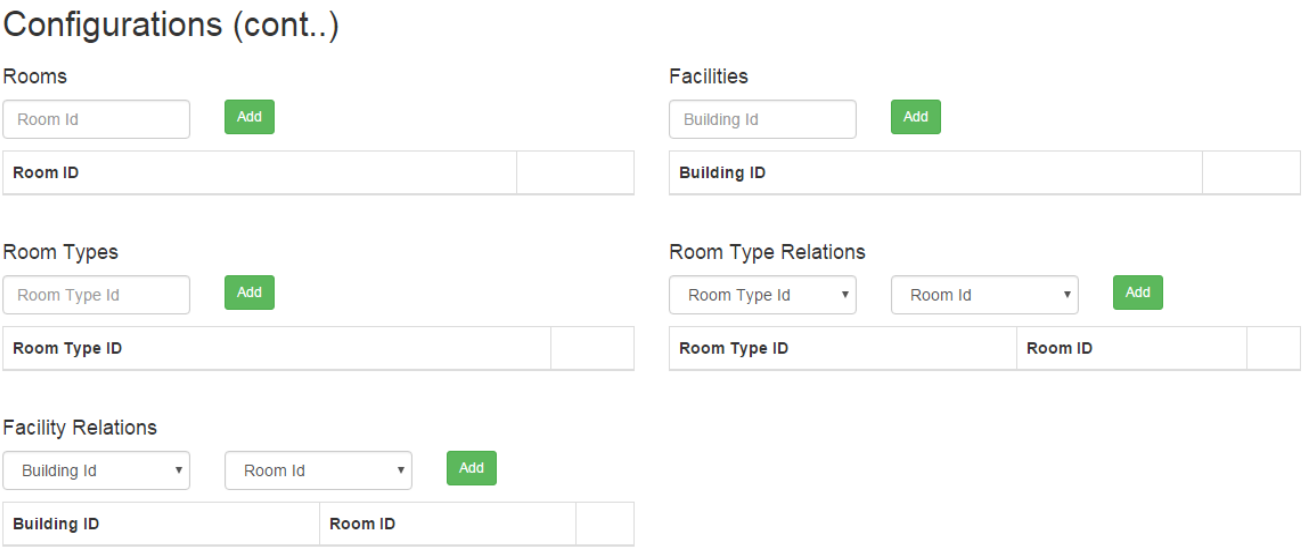
From the table it shows that the actual lunch time (12:00-13:00) is adjusted to (12:30 – 13:30), since the booking record is created using the buffer length of 30 minutes.

Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-21	08:00	08:25	25 mins.	1	1	✔ No		1
2	1	1	1	2016-01-21	08:35	09:00	25 mins.	1	1	✔ No		1
3	1	1	1	2016-01-21	09:10	09:35	25 mins.	1	1	✔ No		1
4	1	1	1	2016-01-21	09:45	10:10	25 mins.	1	1	✔ No		1
5	1	1	1	2016-01-21	10:20	10:45	25 mins.	1	1	✔ No		1
6	1	1	1	2016-01-21	10:55	11:20	25 mins.	1	1	✔ No		1
7	1	1	1	2016-01-21	11:30	11:55	25 mins.	1	1	✔ No		1
8	1	1	1	2016-01-21	12:05	12:30	25 mins.	1	1	✔ No		1
9	1	1	1	2016-01-21	13:30	13:55	25 mins.	1	1	✔ No		1
10	1	1	1	2016-01-21	14:05	14:30	25 mins.	1	1	✔ No		1

Scenario 11 (Flexible type without input parameters)

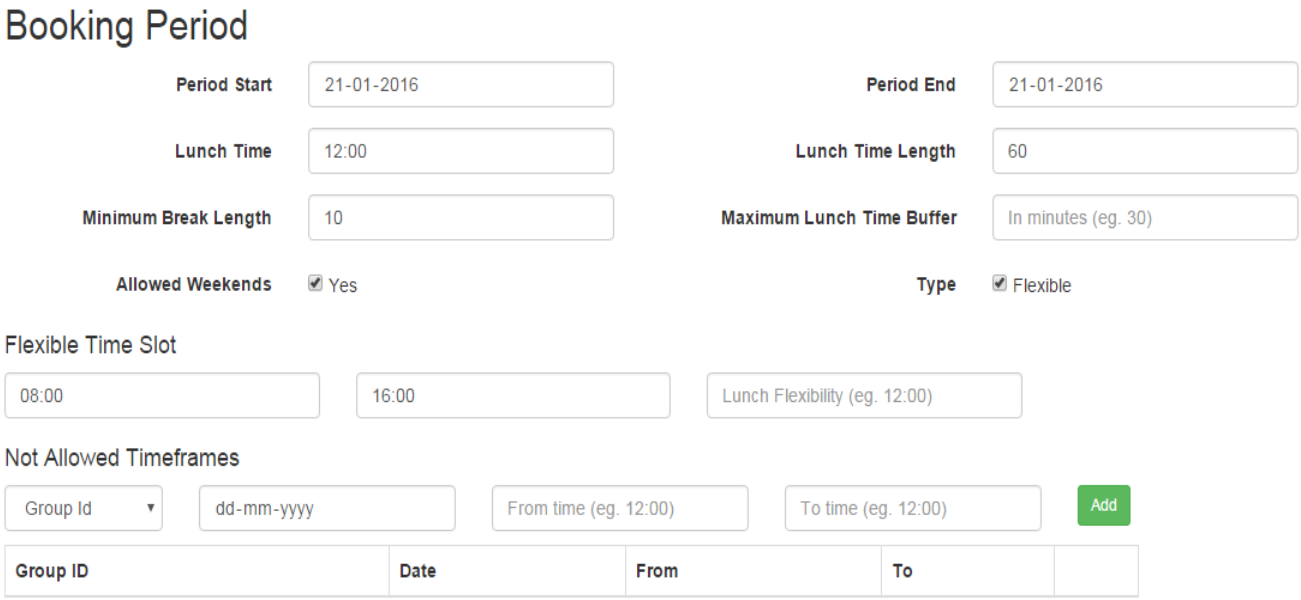
This scenario deals with the flexible type booking records without providimg input parameters such as Group Id, Room Id, Teachers, building relations, roomtype relations and facility relations.





Booking Period Selection

The booking period was selected for one day to show how the algorithm works without providing input parameters. Flexible slot starts from 08:00 to 16:00.The lunch time starts from 12:00 to 13:00 with a span of 60 minutes.



Booking Patterns

Two booking patterns were created for the flexible type booking records.The frist one is of length 20 minutes and second one is of length 25 minutes. From the table it clears that input parameters were not provided.

booking 1

Length : 20 mins
Amount : 3
Interval : 1
From : 20-01-2016
To : 20-01-2016

booking 2

Length : 25 mins
Amount : 3
Interval : 1
From : 20-01-2016
To : 20-01-2016

Booking Id	Course Id	Group Id	Length	Room Setting	Amount	Interval	From	To	Booking Info	
1			20	✖	3	1	2016-01-21	2016-01-21	Rooms : Facilities : Room Types : Teachers :	Delete
2			25	✖	3	1	2016-01-21	2016-01-21	Rooms : Facilities : Room Types : Teachers :	Delete

Booking Records

The flexible type booking records were created without providing input parameters. In default it takes the booking Id as a value for the course Id,group Id,building Id,room Id and teachers.

Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-21	08:00	08:20	20 mins.	1	1	✔ No		1
2	2	2	2	2016-01-21	08:00	08:25	25 mins.	2	2	✔ No		2
3	1	1	1	2016-01-21	08:30	08:50	20 mins.	1	1	✔ No		1
4	2	2	2	2016-01-21	08:35	09:00	25 mins.	2	2	✔ No		2
5	1	1	1	2016-01-21	09:00	09:20	20 mins.	1	1	✔ No		1
6	2	2	2	2016-01-21	09:10	09:35	25 mins.	2	2	✔ No		2

Input and Output Specifications

Method

1. GetSchedule

Input

Configuration

- CurrentBookingRecords
  - bookingRecordID
  - groupID
  - date
  - startTime
  - endTime
  - roomID
  - buildingID
  - AssignedTeachers
    - userID
  - NotAllowedTimeframes
    - date
    - fromTime
    - toTime
- Users
  - userID
  - userType
- Groups
  - groupID
- GroupRelations
  - userID
  - groupID
- Facilities
  - buildingID
- Rooms
  - roomID

AllowedWeekends :

monday

tuesday

wednesday

thursday

friday

saturday

sunday

RoomTypeRelations

roomTypeID

roomID

FacilityRelations

buildingID

roomID

RoomTypes

roomTypeID

Constraints

BookingPeriod

periodStart

periodEnd

lunchTime

lunchTimeLength

minimumBreakLength

maximumLunchTimeBuffer

allowedWeekends

type

NotAllowedDateTimeframes:

groupID

date

fromTime

toTime

StaticSlotsPerDay

fromTime

toTime

FlexibleTimeframe

fromTime

toTime

lunchFlexibility

Booking:

bookingID

courseID

groupID

length

amount

interval

from

to

description

roomSetting

AssignedTeachers

teacherID

AcceptableFacilities

buildingID

AcceptableRooms

roomID

AcceptableRoomTypes

roomTypeID

Output

BookingRecords

bookingRecordID

bookingID

courseID

groupID

length

date

startTime

endTime

buildingID

roomID

description

collissionType

collisionReason

AssignedTeachers

userID

Sample Input


```
{"Configuration":{
"CurrentBookingRecords":[],
"Users":[
{"userID":1,"userType":"1"},
{"userID":2,"userType":"1"},
{"userID":3,"userType":"1"}
],
"Groups":[
{"groupID":1},
{"groupID":2},
{"groupID":3}
```

```
        ],
"GroupRelations":[
        {"groupID":"1","userID":"1"},
        {"groupID":"2","userID":"2"},
        {"groupID":"3","userID":"3"}
        ],
"Facilities":[
        {"buildingID":1}
        ],
"Rooms":[
        {"roomID":1},
        {"roomID":2},
        {"roomID":3}
        ],
"AllowedWeekends":{
        "monday":true,
        "tuesday":true,
        "wednesday":true,
        "thursday":true,
        "friday":true,
        "saturday":false,
        "sunday":false
        },
"RoomTypeRelations":[
        {"roomTypeID":"1","roomID":"1"},
        {"roomTypeID":"1","roomID":"2"},
        {"roomTypeID":"1","roomID":"3"}
        ],
"FacilityRelations":[
        {"buildingID":"1","roomID":"1"},
        {"roomID":"2","buildingID":"1"},
        {"buildingID":"1","roomID":"3"}
        ],
"RoomTypes":[
        {"roomTypeID":1}
```

```
            ]
        },
"Constraints":{
        "BookingPeriod":{
            "periodStart":"2016-01-07",
            "periodEnd":"2016-01-16",
            "lunchTime":"12:00",
            "lunchTimeLength":60,
            "minimumBreakLength":10,
            "maxLunchTimeBuffer":20,
            "allowedWeekends":false,
            "type":false,
            "NotAllowedDateTimeframes":[
                {"groupID":"1","date":"2016-01-07","fromTime":"08:00","toTime":"08:30"},
                {"groupID":"2","date":"2016-01-08","fromTime":"08:00","toTime":"08:30"},
                {"groupID":"3","date":"2016-01-11","fromTime":"08:00","toTime":"08:30"}
                ],
            "StaticSlotsPerDay":[
                {"fromTime":"08:00","toTime":"09:00"},
                {"fromTime":"09:00","toTime":"10:00"},
                {"fromTime":"10:00","toTime":"11:00"}
                ],
            "FlexibleTimeframe":{},
            "Bookings":[
                {
                "courseID":1,
                "groupID":"1",
                "length":20,
                "amount":2,
                "from":"2016-01-07",
                "to":"2016-01-16",
                "interval":1,
                "description":"",
                "roomSetting":true,
                "AssignedTeachers":[{"userID":"1"}],
```

"AcceptableFacilities":[{"buildingID":"1"}],
"AcceptableRooms":[{"roomID":"1"}],
"AcceptableRoomTypes":[{"roomTypeID":"1"}],
"bookingID":1
},
{
"courseID":2,
"groupID":"2",
"length":30,
"amount":2,
"from":"2016-01-07",
"to":"2016-01-16",
"interval":1,
"description":"",
"roomSetting":true,
"AssignedTeachers":[{"userID":"2"}],
"AcceptableFacilities":[{"buildingID":"1"}],
"AcceptableRooms":[{"roomID":"2"}],
"AcceptableRoomTypes":[{"roomTypeID":"1"}],
"bookingID":2
},

{
"courseID":3,
"groupID":"3",
"length":20,
"amount":2,
"from":"2016-01-07",
"to":"2016-01-16",
"interval":2,
"description":"",
"roomSetting":true,
"AssignedTeachers":[{"userID":"3"}],
"AcceptableFacilities":[{"buildingID":"1"}],

```
                    "AcceptableRooms":[{"roomID":"3"}],
                    "AcceptableRoomTypes":[{"roomTypeID":"1"}],
                    "bookingID":3}
                    ]
                }
        }
}
```


Sample Output


```
[
        {
        "AssignedTeachers":[{"userID":1}],
        "bookingID":1,
        "bookingRecordID":1,
        "buildingID":1,
        "collisionReason":null,
        "collisionType":false,
        "courseID":1,
        "date":"2016-01-07",
        "description":"",
```

"endTime":"09:00",
"groupID":1,
"length":20,
"roomID":1,
"startTime":"08:40"
},
{
"AssignedTeachers":[{"userID":2}],
"bookingID":2,
"bookingRecordID":2,
"buildingID":1,
"collisionReason":null,
"collisionType":false,
"courseID":2,
"date":"2016-01-07",
"description":"",
"endTime":"08:30",
"groupID":2,
"length":30,
"roomID":2,
"startTime":"08:00"
},

{
"AssignedTeachers":[{"userID":3}],
"bookingID":3,
"bookingRecordID":3,
"buildingID":1,
"collisionReason":null,
"collisionType":false,
"courseID":3,
"date":"2016-01-07",
"description":"",
"endTime":"08:20",
"groupID":3,

"length":20,
"roomID":3,
"startTime":"08:00"
},
{
"AssignedTeachers":[{"userID":1}],
"bookingID":1,
"bookingRecordID":4,
"buildingID":1,
"collisionReason":null,
"collisionType":false,
"courseID":1,
"date":"2016-01-07",
"description":"",
"endTime":"09:30",
"groupID":1,
"length":20,
"roomID":1,
"startTime":"09:10"
},

{
"AssignedTeachers":[{"userID":2}],
"bookingID":2,
"bookingRecordID":5,
"buildingID":1,
"collisionReason":null,
"collisionType":false,
"courseID":2,
"date":"2016-01-07", "description":"",
"endTime":"09:30",
"groupID":2,
"length":30,

"roomID":2,
"startTime":"09:00"
},
{
"AssignedTeachers":[{"userID":3}],
"bookingID":3,
"bookingRecordID":6,
"buildingID":1,
"collisionReason":null,
"collisionType":false,
"courseID":3,
"date":"2016-01-07",
"description":"",
"endTime":"08:50",
"groupID":3,
"length":20,
"roomID":3,
"startTime":"08:30"
},

{
"AssignedTeachers":[{"userID":1}],
"bookingID":1,
"bookingRecordID":7,
"buildingID":1,
"collisionReason":null,
"collisionType":false,
"courseID":1,
"date":"2016-01-11",
"description":"",
"endTime":"08:20",
"groupID":1,

"length":20,
"roomID":1,
"startTime":"08:00"
},
{
"AssignedTeachers":[{"userID":2}],
"bookingID":2,
"bookingRecordID":8,
"buildingID":1,
"collisionReason":null,
"collisionType":false,
"courseID":2,
"date":"2016-01-11",
"description":"",
"endTime":"08:30",
"groupID":2,
"length":30,
"roomID":2,
"startTime":"08:00"
},

{
"AssignedTeachers":[{"userID":1}],
"bookingID":1,
"bookingRecordID":9,
"buildingID":1,
"collisionReason":null,
"collisionType":false,
"courseID":1,
"date":"2016-01-11",
"description":"",
"endTime":"09:20",
"groupID":1,

```
        "length":20,
        "roomID":1,
        "startTime":"09:00"
        },
        {
        "AssignedTeachers":[{"userID":2}],
        "bookingID":2,
        "bookingRecordID":10,
        "buildingID":1,
        "collisionReason":null,
        "collisionType":false,
        "courseID":2,
        "date":"2016-01-11",
        "description":"",
        "endTime":"09:30",
        "groupID":2,
        "length":30,
        "roomID":2,
        "startTime":"09:00"
        }
]
```

Booking Record Id	Booking Id	Course Id	Group Id	Date	Start Time	End Time	Length	Building Id	Room Id	Collision	Collision Reason	Assigned Teachers
1	1	1	1	2016-01-07	08:40	09:00	20 mins.	1	1	✔ No		1
2	2	2	2	2016-01-07	08:00	08:30	30 mins.	1	2	✔ No		2
3	3	3	3	2016-01-07	08:00	08:20	20 mins.	1	3	✔ No		3
4	1	1	1	2016-01-07	09:10	09:30	20 mins.	1	1	✔ No		1
5	2	2	2	2016-01-07	09:00	09:30	30 mins.	1	2	✔ No		2
6	3	3	3	2016-01-07	08:30	08:50	20 mins.	1	3	✔ No		3
7	1	1	1	2016-01-11	08:00	08:20	20 mins.	1	1	✔ No		1
8	2	2	2	2016-01-11	08:00	08:30	30 mins.	1	2	✔ No		2
9	1	1	1	2016-01-11	09:00	09:20	20 mins.	1	1	✔ No		1
10	2	2	2	2016-01-11	09:00	09:30	30 mins.	1	2	✔ No		2