Table of Content

1. General description
2. Aims
   1. General Aim
   2. Specific Aims
3. User Requirements
4. Static Model
5. Use Case Diagram
6. Activity Diagram
7. Data Dictionary
8. Database Model
9. Tracking spreadsheet
10. API Reference
11. User’s Manual
12. Anexo

General Description

The booking system is meant to be a tool to make reservations of teaching resources available at the school; these resources are buildings, furniture, mobile equipment, electronic devices such as computers, telescopes, etc. Teachers will be able to set apart any of these resources for specific periods of time, these last ones are tied to a schedule in accord to the timetable of the school.

The system will be developed in a web environment so that any teacher and administrator will be capable to access it from any computer with an internet connection.

System’s capabilities

The system will show available resources to the end user when he comes in to check out for a booking. It will handle a timetable of periods of classes this will make possible to make reservations to periods of time according to the school schedule.

Aims

General Aim

* To develop a web based system to set bookings over available school resources such as laptops, buildings, furniture, mobile equipment, etc.

Specific Aims

* To create a module to handle the timetable of the school activities.
* To provide the system with a module to handle availability of the resources at any given period of time.
* To manage user accounts to attach bookings to person in charge.

User Requirements

* Bookings where furniture and mobile equipment are involved need to have a specific location.
* Bookings like laptops must have a predefined list of rooms to be placed of.
* All resources need to have a person or group of person as responsible. This way they become administrators over that specific resource.
* All resources must have the possibility to be reserved 1 or more days ahead of its use.
* The booking system must allow to set one booking in multiple dates not necessarily following a pattern but to be able to fit according to the resources usage.
* Allow resources to be booked according to the timetable.

Static Model

Timetable

Booking System

Resources

Rooms

Staff

Class Deployment

Timetable

Description: This entity handles periods of classes for primary and secondary as well. It takes into consideration from 7am to 11:59pm. It divides the time in seven periods and after this last one divides time in periods of half an hour.

Attributes:

* Period name: Specifies the identifier of the period (Period 1, Period 2, Period 3, Period 4, Period 5, Period 6, Period 7, etc.)
* Start time: Indicates the time when the period starts.
* End time: Indicates the time when the period ends.
* Code: It links the name of the period with an identifier.
* Active: Is a flag which sets the period to be active or not.

Resources

Description: This entity manages the resources available at the school (Electronic devices, furniture, buildings, etc.). Contains all information linked to them.

Attributes:

* Name: Specify the identifier of the resource.
* Description: Gives a brief description of what the resource is about.
* Type: All resources belong to certain types or categories, this attribute specify which one this resource belongs to.
* Quantity: Indicates the quantity of the resource.
* Time before to be booked: Indicates the amount of time that the resource needs to be booked before its usage.
* Rooms available to: Indicates the room(s) that this resource is available to.
* Active: Is a flag which sets the resource to be active or not.

Rooms

Description: This entity handles the rooms inside the school where the resources are meant to be used.

Attributes:

* Name: Identifier of the room.
* Section: All rooms belong to different location. This attributes sets the section the room belong to.
* Person in charge: Contains de the name of the person responsible for the room. This is an optional attribute because not all the buildings are managed by a single person.

Staff

Description: This entity provides the information of all the staff in the school. It is useful to provide credential information.

Attributes:

* Name: Full name of the staff member.
* Username: Username assigned to the staff member.
* ID: Unique identifier of staff member.

Use Cases

Actors

Staff Member: Now on it will be named “staff”. This actor is meant to create bookings in the system. He has a username, password and an ID to login the system. He is able as well to perform actions over the resources such as to set and check bookings, manage resources and rooms, etc.

The System: Now on it will be named “system”. The system is one of the main actors; it serves all the requests from the staff.

Administrator: All people in charge of managing the site. They have full access to all features and are allowed to perform all tasks in it.

1. Create category of resource
   1. Actors: Administrator, System
   2. Description: The administrator creates a category by supplying a name. The system takes this information and save it to database.
2. Create a resource
   1. Actors: Administrator, System
   2. Description: The administrator creates a resource into the system. To do so he specifies all the attributes of the resource by filling in the form in the system. Then the system takes all the information and puts it into the database.
3. Create a room
   1. Actors: Administrator, System
   2. Description: The administrator creates a location for resources to be used. These locations are rooms which belong to the campus of the school. The administrator must specify all room attributes and then the system keeps this entire info safe into database.
4. Manage timetable
   1. Actors: Administrator, System
   2. Description: The administrator sets up the timetable of the school to make it available at the system. The system splits the timetable into slots this way staff members can posts bookings into them.
5. Authentication process
   1. Actors: Administrator, Staff, System
   2. Description: The administrator or the staff member request access to the system. The system handles the request by providing two options of authentication:
      1. Open ID: Through the account in google apps the staff member of the administrator can give log in information.
      2. Traditional log in: Through the school username and password the staff member or administrator can have access to the system.
   3. After having verified user’s credentials the system decides whether to allow access or not.
6. Create a booking
   1. Actors: Staff member, System
   2. Description: The staff member looks for the resources that he wants to book. Then staff member sets up a quantity if necessary and write down details. The system collects all of this information and keeps it save in sessions variables until it is saved into database. Then the system shows the slots of time available for the set of resources previously picked up by the staff member. When the slots are available in the screen the staff can easily drag and drop it to all the available slots.
7. Delete a booking
   1. Actors: Staff member, administrator, system
   2. Description: Staff member or administrator look for the booking he wants to delete. When find it, he asks for deletion. The system responds the petition by deleting the booking, setting the slot as empty and available for other bookings.
8. Update a booking
   1. Actors: Staff member, system
   2. Description: Staff member looks for the booking to update. Then modify all information about it and save changes. If staff member needs to change the time and date, then he needs to create a new booking.

Activity Diagrams

|  |  |
| --- | --- |
| Create a category of resources | |
| Administrator | System |
| Requests to create a new resource category  Asks resource’s category features  Provide name, description  Takes resource category information and puts it into database |  |

|  |  |
| --- | --- |
| Create a resource | |
| Administrator | System |
| Requests to create a new resource  Asks resource’s features  Provide name, quantity, location, person in charge, availability  Takes resource information and puts it into database |  |

|  |  |
| --- | --- |
| Create a room | |
| Administrator | System |
| Requests to create a new room  Asks room’s features  Provide name, location, person in charge  Takes room information and puts it into database |  |

|  |  |
| --- | --- |
| Manage timetable | |
| Administrator | System |
| Requests to create a slot of time  Asks start and end times  Provide start and end time  Creates a timetable from slots of time |  |

|  |  |
| --- | --- |
| Authentication process | |
| User | System |
| Requests access to the system  Provides user with two options of authentication  Choose an option and provides username and password  Verifies user credentials and gives access or denies it. |  |

|  |  |
| --- | --- |
| Create a booking | |
| User | System |
| Looks for the resources that he wants to book. Then staff member sets up a quantity if necessary and write down details  Collects all of this information and keeps it save in sessions variables until it is saved into database  Shows to the user the slots available to set the booking according to the chosen resources  Drag the set to the available slots and set the booking |  |
| Delete a booking | |
| User | System |
| Looks for the booking to delete  Delete the booking and cleanse the slot to make it available  Gets a notification confirming the deletion |  |

|  |  |
| --- | --- |
| Update a booking | |
| User | System |
| Looks for the booking to update  Make changes and keep them on database  Gets a notification confirming the update |  |

Data Dictionary

|  |  |  |  |
| --- | --- | --- | --- |
| Resource’s Categories: This table contains all categories and their full features. Categories behold similar resources. (**resource\_category**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. Identifies each category by a unique number. |
| name | Varchar(85) | Not null | Name of the category. |
| description | Varchar(150) | null | Description of the category |
| created\_date | timestamp | Not null | Timestamp when the category was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Resources: This table beholds all the resources available at the school with full detail. (**resource**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. Identifies each resource by a unique number. |
| name | Varchar(85) | Not null | Name of the resource. |
| description | Varchar(150) | null | Description of the resource |
| category\_id | integer | Not null | Id of the category which the resource belongs to. |
| quantity | integer | Not null, default 1 | Total amount of available resources. |
| time\_before\_booking | integer | Not null, default 0 | Indicates the time you need to set the booking before it starts |
| Is\_active | Enum(‘t’,’f’) | Not null, default ‘t’ | Indicates if the resource is available or not. |
| created\_date | timestamp | Not null | Timestamp when the resource was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Sections: This table beholds information about section of the school (Administration, Lower Primary, Upper Primary and Secondary). (**section**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. Identifies each section by a unique number. |
| name | Varchar(85) | Not null | Name of the section. |
| description | Varchar(150) | null | Description of the resource |
| created\_date | timestamp | Not null | Timestamp when the resource was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Rooms: Beholds all rooms at the school. (**room**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. Identifies each room by a unique number. |
| name | Varchar(85) | Not null | Name of the room. |
| description | Varchar(150) | null | Description of the room. |
| section\_id | integer | Not null | Id of the category which the resource belongs to. |
| created\_date | timestamp | Not null | Timestamp when the resource was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Link rooms and resources: This table links a resource with multiple rooms. A resource can be picked to be used in one of the several rooms linked to it. (**link\_resource\_room**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. |
| Resource\_id | integer | Not null | Id of the resource which the room makes reference. |
| Room\_id | integer | Not null | Id of the room where the resource can be used. |
| created\_date | timestamp | Not null | Timestamp when the resource was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Link administrators and resources: This table links a resource with multiple or single administrators of the resource. A resource can be managed by one or several people. (**link\_resource\_admin**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. |
| Resource\_id | integer | Not null | Id of the resource which the admin makes reference. |
| owner | Varchar(8) | Not null | Id of the admin where the resource can be used. |
| created\_date | timestamp | Not null | Timestamp when the resource was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Timetable: This table will manage the periods of time that make the timetable. (**timetable**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. |
| Name | Varchar(80) | Not null | Name of the period |
| Nickname | Varchar(35) | Not null | Short name to be used in the period |
| Description | Varchar(150) | Null | Description of the period of time |
| Start\_time | Time | Not null | Time of start of the period |
| End\_time | Time | Not null | Time of end of the period |
| created\_date | timestamp | Not null | Timestamp when the resource was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Booking Header: Beholds all the information related to the booking itself.(**bookingheader**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. |
| Subject | Varchar(80) | Not null | Name of the period |
| Location | integer | Not null, default -1 | Short name to be used in the period. When default value it means it can be anywhere. |
| Description | Varchar(150) | Null | Description of the period of time |
| Owner | Varchar(8) | Not null | Reference to the owner of the booking |
| Is\_active | Enum(‘t’,’f’) | Not null, default ‘t’ | Indicates if the booking is currently active or not. (There are no deletes, just active false) |
| ip | Varchar(15) | Not null | Contains IP address from the computer that generates the booking |
| created\_date | timestamp | Not null | Timestamp when the resource was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Booking Details: It contains the periods where the booking has been set up.(**bookingchild**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. |
| Booking\_id | integer | Not null | Reference to the booking |
| Start\_time | Time | Not null | Time of start of the period |
| End\_time | Time | Not null | Time of end of the period |
| bdate | date | Not null |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Link booking and resources: this table contains a reference between the booking and the resources to be used. (**link\_booking\_resource**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. |
| Resource\_id | integer | Not null | Id of the resource which the resource makes reference. |
| Booking\_id | integer | Not null | Id of the booking which owns the resource |
| created\_date | timestamp | Not null | Timestamp when the resource was created. |

|  |  |  |  |
| --- | --- | --- | --- |
| Notifications: this table contains a full and detailed record of activities that the user performs over the events created in the booking system. (**notification**) | | | |
| Field | Type | Extras | Description |
| id | integer | Primary\_key, auto\_increment, not null | Primary key of the table. |
| Resource\_id | integer | Not null | Id of the resource which the resource makes reference. |
| Booking\_id | integer | Not null | Id of the booking which owns the resource |
| Event\_id | integer | Not Null | Id of the envolved event. |
| performer | string | Not Null | ID Card of the performer of the activity |
| Activity\_type | Enum(‘new’,’update’,’delete’) | Not null default ‘new’ | Records the type of the activity |
| Message | string | Not null | A message to describe further the activity. |
| Performed\_date | timestamp | Not null | Timestamp when the activity is done. |

Data Base Model [Relational Schema]

Staff members (External data)

Resources

Sections

Rooms

Booking Headers

Booking Details

Categories for Resources

Timetable

Data Base Model [Physical Schema]

