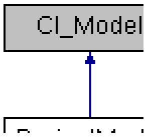
ProjectModel Class Reference

Inheritance diagram for ProjectModel:



Public Member Functions

- **__construct** ()
- **addProject** ($userID, $project=NULL)
- **addUserProject** ($userID, $projID, $userProject)
- **getProject** ($projID)
- **getProjects** ($filter=NULL, $and=false)
- **getProjectMembers** ($projID, $filter=NULL, $and=false)
- **getUserProjects** ($userID, $filter=NULL, $and=false)
- **deleteProject** ($projID)
- **deleteUserProject** ($userID, $projID)
- **deleteAllProjectsUser** ($userID)
- **deleteAllUsersProject** ($projID)
- **updateProject** ($projID, $userID=NULL, $project=NULL)
- **updateUserProject** ($projID, $userID, $userProject=NULL)

Constructor & Destructor Documentation

__construct ()

__construct() is the constructor which initiates the connection to the database

---

Member Function Documentation

addProject (*$userID*, *$project* = `NULL`)

addProject() is a method for adding a project

Parameters:

$userID	is is the id of the owner of the project
$project	is an array containing project data
$project['pname']	(required) contains the name of the project
$project['pdescription']	(optional) contains the description of the project
$project['pdate_start']	(required) contains the date of the beginning of the project
$project['pdate_end']	(required) contains the date of the end of the project

Returns:

new project id if insert succeeded and FALSE if not + name of the creator of the project

addUserProject (*$userID*, *$projID*, *$userProject*)

addUserProject() is a method for adding a member into a project

Parameters:

$userID	(required) is the id of a participant
$projectID	(required) is the id of the project
$userProject	['role_project'] (optional) contains the role of the participant in the project
$userProject	['gestion'] (required) means if a participant may manage the project (1) or not (0) =default

Returns:

TRUE if insert succeeded and FALSE if not

deleteAllProjectsUser (*$userID*)

deleteAllProjectsUser() remove all projects for a specific user

Parameters:

$userID	

Returns:

a boolean (TRUE if deletion has been applied, FALSE if not)

deleteAllUsersProject (*$projID*)

deleteAllUsersProject() remove all users for a specific project

Parameters:

$projID	

Returns:

a boolean (TRUE if deletion has been applied, FALSE if not)

deleteProject (*$projID*)

deleteProject() delete a project based on its id

Parameters:

$projID	

Returns:

a boolean (TRUE if deletion has been applied, FALSE if not)

deleteUserProject (*$userID*, *$projID*)

deleteUserProject() remove a project for a specific user

Parameters:

$userID	
$projID	

Returns:

a boolean (TRUE if deletion has been applied, FALSE if not)

getProject (*$projID*)

getProject() this method returns a project based on its id

Parameters:

$projID	project id

Returns:

a project with the firstname and lastname of project creator
$response['id'] is the project id
$response['project_name'] is the name of the project

$response['date_start'] is the date when the projet begin
$response['date_end'] is the date when the projet end
$response['owner_lastname'] is the lastname of creator
$response['owner_firstname'] is the firstname of creator
$response['project_description'] is the description of the project

getProjectMembers (*$projID*, *$filter* = `NULL`, *$and* = `false`)

getProjectMembers() this method returns the members of a project based on its id

Parameters:

$projID	project id
$filter	is optional and is an array containing search criterions
$filter['member_lastname']	is the lastname of a member
$filter['member_firstname']	is the firstname of a member
$filter['member_role']	is the role of a member
$filter['member_gestion']	means 1 or 0 (if the participant may not manage the project)
$filter['owner_lastname']	is the lastname of the project's owner
$filter['owner_firstname']	is the firstname of the project's owner
$and	is optional and is an boolean which is FALSE (default behavior) for processing teh search query with OR operators and TRUE for AND operators

Returns:

an array of members for a project (name ascending order)

See also:

getProjectMembers() for the data structure of returned users

getProjects (*$filter* = `NULL`, *$and* = `false`)

getProjects() this method returns a project based on its id

Parameters:

$projID	project id
$filter	is optional and is an array containing search criterions
$filter['id']	is the project id
$filter['project_name']	is the name of the project
$filter['project_description']	is the description of the project
$filter['date_start']	is the date when the projet begin
$filter['date_end']	is the date when the projet end
$filter['owner_lastname']	is the lastname of creator
$filter['owner_firstname']	is the firstname of creator
$and	is optional and is an boolean which is FALSE (default behavior) for processing teh search query with OR operators and TRUE for AND operators

Returns:

an array of projects (date start descending order)

See also:

getProjects() for the data structure of returned users

getUserProjects (*$userID*, *$filter* = `NULL`, *$and* = `false`)

getUserProjects() this method returns the projects for which a user participates

Parameters:

$userID	the id of the user
$filter	is optional and is an array containing search criterions
$filter['project_id']	is the id of the project
$filter['project_name']	is the name of the project
$filter['project_date_start']	is the date when the project start
$filter['project_date_end']	is the date when the project end
$filter['project_role']	is the role of the member for the project
$filter['project_gestion']	means 1 or 0 (if the participant may not manage the project)
$filter['project_owner']	means 1 or 0 (if the participant is not the owner of the project)
$and	is optional and is an boolean which is FALSE (default behavior) for processing teh search query with OR operators and TRUE for AND operators

Returns:

an array of projects for a user (date start descending order)

See also:

getUserProjects() for the data structure of returned projects

updateProject (*$projID*, *$userID* = `NULL`, *$project* = `NULL`)

updateProject() is a method for updating a project

Parameters:

$userID	is the id of the owner of the project
$project	is an array containing project data
$project['id']	contains the id
$project['pname']	contains the name
$project['pdescription']	contains the description of the project
$project['pdate_start']	contains the date of the beginning of the project
$project['pdate_end']	contains the date of the end of the project

Returns:

TRUE if update succeeded and FALSE if not

updateUserProject (*$projID*, *$userID*, *$userProject* = `NULL`)

updateUserProject() is a method for updating a project

Parameters:

$userID	is the member of the project
$projID	contains the id of the project
$userProject['role']	contains the role of the member for the project
$userProject['manage']	contains 1 or 0 (if the member may not manage the project)

Returns:

TRUE if update succeeded and FALSE if not

The documentation for this class was generated from the following file:

- ProjectModel.php