

SwimmPair-Www

v1.0

Generated by Doxygen 1.9.6

1 Model of SwimmPair web application	1
1.1 Utility Classes	1
1.2 Object Classes	1
1.3 Manager Classes	1
1.3.1 Github repo of this model	1
2 Class Index	3
2.1 Class List	3
3 Class Documentation	5
3.1 Auth Class Reference	5
3.1.1 Detailed Description	5
3.1.2 Member Function Documentation	5
3.1.2.1 requireRole()	5
3.2 Club Class Reference	6
3.2.1 Detailed Description	7
3.2.2 Constructor & Destructor Documentation	7
3.2.2.1 __construct()	7
3.2.3 Member Function Documentation	7
3.2.3.1 SerializeFull()	7
3.2.3.2 SerializeSlim()	8
3.3 ClubsManager Class Reference	8
3.3.1 Detailed Description	9
3.3.2 Constructor & Destructor Documentation	9
3.3.2.1 __construct()	9
3.3.3 Member Function Documentation	9
3.3.3.1 FindAllClubs()	9
3.3.3.2 GetClubAffiliationToRegion()	9
3.3.3.3 GetClubByID()	10
3.3.3.4 InsertNewClub()	10
3.3.3.5 UpdateClub()	11
3.4 Cup Class Reference	11
3.4.1 Detailed Description	12
3.4.2 Constructor & Destructor Documentation	12
3.4.2.1 __construct()	12
3.4.3 Member Function Documentation	13
3.4.3.1 SerializeFull()	13
3.4.3.2 SerializeSlim()	13
3.5 CupsManager Class Reference	14
3.5.1 Detailed Description	14
3.5.2 Constructor & Destructor Documentation	15
3.5.2.1 __construct()	15
3.5.3 Member Function Documentation	15

3.5.3.1 DeleteOldAvailability()	15
3.5.3.2 DeleteOldPairing()	15
3.5.3.3 FindAllPastCupsMostRecentFirst()	16
3.5.3.4 FindAllUpcomingCupsEarliestFirst()	16
3.5.3.5 FindPairingsForThisCup()	16
3.5.3.6 GetCupByID()	17
3.5.3.7 GetEarliestCupYear()	17
3.5.3.8 GetMaximumCupYear()	17
3.5.3.9 GetNewCupIDByInfo()	17
3.5.3.10 GetPairingHashForThisCup()	18
3.5.3.11 InsertNewAvailability()	18
3.5.3.12 InsertNewCup()	19
3.5.3.13 InsertNewPairing()	19
3.5.3.14 IsUserAvailableForTheCup()	19
3.6 Page Class Reference	20
3.6.1 Detailed Description	21
3.6.2 Constructor & Destructor Documentation	21
3.6.2.1 __construct()	21
3.6.3 Member Function Documentation	21
3.6.3.1 Serialize()	21
3.7 PagesManager Class Reference	22
3.7.1 Detailed Description	22
3.7.2 Constructor & Destructor Documentation	22
3.7.2.1 __construct()	22
3.7.3 Member Function Documentation	22
3.7.3.1 GetPageByID()	23
3.7.3.2 UpdatePage()	23
3.8 PairPositionUser Class Reference	24
3.8.1 Detailed Description	24
3.8.2 Constructor & Destructor Documentation	24
3.8.2.1 __construct()	24
3.8.3 Member Function Documentation	25
3.8.3.1 Serialize()	25
3.9 Position Class Reference	25
3.9.1 Detailed Description	26
3.9.2 Constructor & Destructor Documentation	26
3.9.2.1 __construct()	26
3.9.3 Member Function Documentation	26
3.9.3.1 Serialize()	26
3.10 PositionsManager Class Reference	27
3.10.1 Detailed Description	27
3.10.2 Constructor & Destructor Documentation	27

3.10.2.1 __construct()	27
3.10.3 Member Function Documentation	28
3.10.3.1 DeleteOldStatsPositions()	28
3.10.3.2 DisplayedLiveStatsConfiguredPositions()	28
3.10.3.3 FindAllPositions()	28
3.10.3.4 GetPositionNameById()	28
3.10.3.5 InsertNewStatPosition()	29
3.11 Post Class Reference	30
3.11.1 Detailed Description	30
3.11.2 Constructor & Destructor Documentation	30
3.11.2.1 __construct()	31
3.11.3 Member Function Documentation	31
3.11.3.1 NullCtor()	31
3.11.3.2 Serialize()	31
3.12 PostsManager Class Reference	32
3.12.1 Detailed Description	32
3.12.2 Constructor & Destructor Documentation	32
3.12.2.1 __construct()	32
3.12.3 Member Function Documentation	33
3.12.3.1 FindAllPostsOrderByIDDesc()	33
3.12.3.2 FindLastNPosts()	33
3.12.3.3 GetFollowingPost()	33
3.12.3.4 GetPostById()	34
3.12.3.5 InsertNewCupPSAPost()	34
3.12.3.6 InsertNewPost()	35
3.12.3.7 UpdatePost()	35
3.13 RefereeRank Class Reference	36
3.13.1 Detailed Description	36
3.13.2 Constructor & Destructor Documentation	36
3.13.2.1 __construct()	36
3.13.3 Member Function Documentation	37
3.13.3.1 Serialize()	37
3.14 Region Class Reference	37
3.14.1 Detailed Description	38
3.14.2 Constructor & Destructor Documentation	38
3.14.2.1 __construct()	38
3.14.3 Member Function Documentation	38
3.14.3.1 Serialize()	38
3.15 RegionsManager Class Reference	39
3.15.1 Detailed Description	39
3.15.2 Constructor & Destructor Documentation	39
3.15.2.1 __construct()	39

3.15.3 Member Function Documentation	40
3.15.3.1 FindAllRegions()	40
3.15.3.2 GetRegionById()	40
3.15.3.3 GetRegionNameOfClub()	40
3.15.3.4 InsertNewRegion()	41
3.15.3.5 UpdateRegion()	41
3.16 Sanitizer Class Reference	42
3.16.1 Detailed Description	42
3.16.2 Member Function Documentation	42
3.16.2.1 getGetInt()	42
3.16.2.2 getGetString()	43
3.16.2.3 getPostInt()	43
3.16.2.4 getPostString()	44
3.17 StatPositionCnt Class Reference	44
3.17.1 Detailed Description	45
3.17.2 Constructor & Destructor Documentation	45
3.17.2.1 __construct()	45
3.17.3 Member Function Documentation	45
3.17.3.1 Serialize()	45
3.18 StatUserCnt Class Reference	46
3.18.1 Detailed Description	46
3.18.2 Constructor & Destructor Documentation	46
3.18.2.1 __construct()	46
3.18.3 Member Function Documentation	47
3.18.3.1 Serialize()	47
3.19 User Class Reference	47
3.19.1 Detailed Description	48
3.19.2 Constructor & Destructor Documentation	48
3.19.2.1 __construct()	48
3.19.3 Member Function Documentation	49
3.19.3.1 SerializeFull()	49
3.19.3.2 SerializeSlim()	49
3.19.3.3 SerializeTag()	49
3.20 UserRights Class Reference	50
3.20.1 Detailed Description	50
3.20.2 Member Function Documentation	50
3.20.2.1 getRightsById()	50
3.21 UsersManager Class Reference	51
3.21.1 Detailed Description	52
3.21.2 Constructor & Destructor Documentation	52
3.21.2.1 __construct()	52
3.21.3 Member Function Documentation	53

3.21.3.1 CountClubSeasonalStatistics()	53
3.21.3.2 CountCupsAttendanceOfUserGivenYear()	53
3.21.3.3 CountOverallStatisticsOfUserGivenYear()	53
3.21.3.4 EmailNewPersonRegistered()	54
3.21.3.5 FindAllActiveUsersOrderByLastNameAsc()	54
3.21.3.6 FindAllInactiveUsersOrderByLastNameAsc()	54
3.21.3.7 FindAllRefereeRanks()	55
3.21.3.8 FindAllRegisteredTeamMembersForTheCup()	55
3.21.3.9 FindAllRegisteredUsersForTheCup()	55
3.21.3.10 FindAllTeamMembers()	56
3.21.3.11 FindAllUsers()	56
3.21.3.12 FindPairedUsersOnCupForPosition()	56
3.21.3.13 GetClubAbbreviationByAffiliationID()	57
3.21.3.14 GetClubNameByAffiliationID()	57
3.21.3.15 GetRefereeRank()	57
3.21.3.16 GetUserByID()	58
3.21.3.17 GetUserEmailByID()	58
3.21.3.18 GetUserFullNameByID()	58
3.21.3.19 IsComing()	59
3.21.3.20 IsEmailPresentAlready()	59
3.21.3.21 IsUserWithIDPresentAlready()	59
3.21.3.22 LoginCandidateToBeAuthorized()	60
3.21.3.23 RegisterUser()	60
3.21.3.24 RetComingCSSClass()	61
3.21.3.25 RetStringComingFlag()	61
3.21.3.26 SetApprovedForUser()	61
3.21.3.27 SetAvailabilityCanGo()	62
3.21.3.28 SetAvailabilityCantGo()	62
3.21.3.29 SetAvailabilityRegister()	63
3.21.3.30 SetLoginEmailForUser()	63
3.21.3.31 SetPasswordForUser()	63
3.21.3.32 SetRefereeRankForUser()	64

Chapter 1

Model of SwimmPair web application

This folder is dedicated to model of SwimmPair web application <https://github.com/KlosStepan/SwimmPair-Www> Via. file start.php in folder above all classes are included, Managers are instantiated with live database connection and this file is included all around the web application. Therefore we view these functions as API functions of our web application because they are sole mean of communication between web and database.

1.1 Utility Classes

There are several utility classes, usually static classes with functions or of enumeration character. They serve a purpose of static functions for Authentication, Sanitization or keeping privileges and names of there appropriate ranks.

1.2 Object Classes

Several classes representing object in this system. They have data and some serialization functions as well.

1.3 Manager Classes

Manager class always moves around one class from Classes(as mentioned above) in order to provide API functions for this object.

1.3.1 Github repo of this model

This model resides in Github repo <https://github.com/KlosStepan/SwimmPair-Www/tree/master/model> and this documentation is generated against it.

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

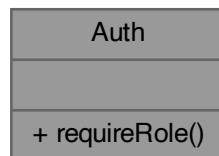
Auth	5
Club	6
ClubsManager	8
Cup	11
CupsManager	14
Page	20
PagesManager	22
PairPositionUser	24
Position	25
PositionsManager	27
Post	30
PostsManager	32
RefereeRank	36
Region	37
RegionsManager	39
Sanitizer	42
StatPositionCnt	44
StatUserCnt	46
User	47
UserRights	50
UsersManager	51

Chapter 3

Class Documentation

3.1 Auth Class Reference

Collaboration diagram for Auth:



Static Public Member Functions

- static [requireRole](#) (\$role)

3.1.1 Detailed Description

[Auth](#) authenticates [User](#) for access through the web application. It can check if [User](#)'s SESSION rights ([UserRights](#)) is set and appropriate.

3.1.2 Member Function Documentation

3.1.2.1 [requireRole\(\)](#)

```
static Auth::requireRole (
    $role ) [static]
```

Summary of [requireRole](#)

Parameters

int	<i>\$role</i>	
-----	---------------	--

Exceptions

<i>RuntimeException</i>	
-------------------------	--

Returns

void

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

- model/Auth.php

3.2 Club Class Reference

Collaboration diagram for Club:

**Public Member Functions**

- [__construct](#) (\$id, \$name, \$abbreviation, \$code, \$img, \$affiliation_region_id)
- [SerializeFull](#) ()
- [SerializeSlim](#) ()

Public Attributes

- `$id`
- `$name`
- `$abbreviation`
- `$code`
- `$img`
- `$affiliation_region_id`

3.2.1 Detailed Description

[Club](#) is an administrative unit of swimming club grouping bunch of users. One [User](#) is VedouciKlubu/1 from [UserRights](#), the rest is Rozhodci/0. It can organize [Cup](#) via its users.

3.2.2 Constructor & Destructor Documentation

3.2.2.1 `__construct()`

```
Club::__construct (
    $id,
    $name,
    $abbreviation,
    $code,
    $img,
    $affiliation_region_id )
```

Ctor of [Club](#) object for web application

Parameters

int	<i>\$id</i>	
string	<i>\$name</i>	
string	<i>\$abbreviation</i>	
int	<i>\$code</i>	
string	<i>\$img</i>	
int	<i>\$affiliation_region_id</i>	

3.2.3 Member Function Documentation

3.2.3.1 `SerializeFull()`

```
Club::SerializeFull ( )
```

SerializeFull outputs 6/6 these members: id, name, abbreviation, code, img, affiliation_region_id

Returns

string

3.2.3.2 SerializeSlim()

```
Club::SerializeSlim ( )
```

SerializeSlim outputs 3/6 these members: id, name, -NULL-, -NULL-, -NULL-, affiliation_region_id

Returns

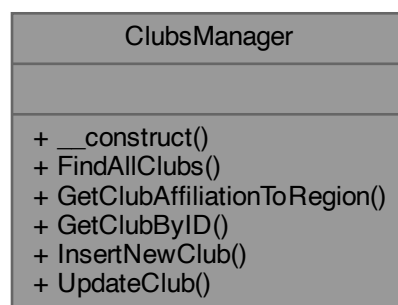
string

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

- model/Club.php

3.3 ClubsManager Class Reference

Collaboration diagram for ClubsManager:



Public Member Functions

- [__construct](#) (mysqli \$mysqli)
- [FindAllClubs](#) ()
- [GetClubAffiliationToRegion](#) (\$clubID)
- [GetClubByID](#) (\$clubID)
- [InsertNewClub](#) (\$name, \$abbreviation, \$club_id, \$img, \$kraj)
- [UpdateClub](#) (\$id, \$name, \$abbreviation, \$code, \$img, \$affiliation_region_id)

3.3.1 Detailed Description

[ClubsManager](#) has API functions to handle [Club](#) object/s and delivers it through web application.

3.3.2 Constructor & Destructor Documentation

3.3.2.1 __construct()

```
ClubsManager::__construct (
    mysqli $mysqli )
```

Initialize [ClubsManager](#) with live database connection.

Parameters

mysqli	<i>\$mysqli</i>	
--------	-----------------	--

3.3.3 Member Function Documentation

3.3.3.1 FindAllClubs()

```
ClubsManager::FindAllClubs ( )
```

Retrieve list of all clubs.

Returns

array<Club>

3.3.3.2 GetClubAffiliationToRegion()

```
ClubsManager::GetClubAffiliationToRegion (
    $clubID )
```

Get [Region](#) id based on [Club](#) id.

Parameters

int	<i>\$clubID</i>	
-----	-----------------	--

Returns

int

3.3.3.3 GetClubByID()

```
ClubsManager::GetClubByID (
    $clubID )
```

Get [Club](#) based on id.

Parameters

int	<i>\$clubID</i>	
-----	-----------------	--

Returns

Club|null

3.3.3.4 InsertNewClub()

```
ClubsManager::InsertNewClub (
    $name,
    $abbreviation,
    $club_id,
    $img,
    $kraj )
```

Create new [Club](#) in the web applicaton.

Parameters

string	<i>\$name</i>	
string	<i>\$abbreviation</i>	
int	<i>\$club_id</i>	
string	<i>\$img</i>	
int	<i>\$kraj</i>	

Returns

bool

3.3.3.5 UpdateClub()

```
ClubsManager::UpdateClub (
    $id,
    $name,
    $abbreviation,
    $code,
    $img,
    $affiliation_region_id )
```

Update [Club](#) information.

Parameters

int	<i>\$id</i>	
string	<i>\$name</i>	
string	<i>\$abbreviation</i>	
string	<i>\$code</i>	
string	<i>\$img</i>	
int	<i>\$affiliation_region_id</i>	

Returns

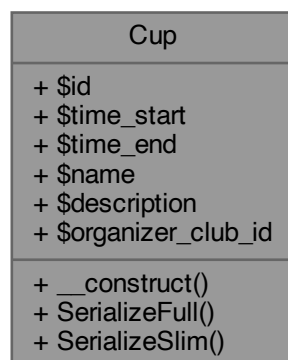
bool

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

- model/ClubsManager.php

3.4 Cup Class Reference

Collaboration diagram for Cup:



Public Member Functions

- [__construct](#) (\$id, \$time_start, \$time_end, \$name, \$description, \$organizer_club_id)
- [SerializeFull](#) ()
- [SerializeSlim](#) ()

Public Attributes

- **\$id**
- **\$time_start**
- **\$time_end**
- **\$name**
- **\$description**
- **\$organizer_club_id**

3.4.1 Detailed Description

[Cup](#) is swimming cup organized by [Club](#) with [User](#) entities signed up to it. Each [User](#) usually has some [Position](#) to perform at [Cup](#) when it takes place.

3.4.2 Constructor & Destructor Documentation

3.4.2.1 __construct()

```
Cup::__construct (
    $id,
    $time_start,
    $time_end,
    $name,
    $description,
    $organizer_club_id )
```

Ctor of [Cup](#) object for web application

Parameters

int	<i>\$id</i>	
string	<i>\$time_start</i>	
string	<i>\$time_end</i>	
int	<i>\$name</i>	
string	<i>\$description</i>	
int	<i>\$organizer_club_id</i>	

3.4.3 Member Function Documentation

3.4.3.1 SerializeFull()

```
Cup::SerializeFull ( )
```

SerializeFull outputs 6/6 these members: id, time_start, time_end, name, description, owningclub

Returns

string

3.4.3.2 SerializeSlim()

```
Cup::SerializeSlim ( )
```

SerializeSlim outputs 4/6 these members: id, time_start, time_end, name, -NULL-, -NULL-

Returns

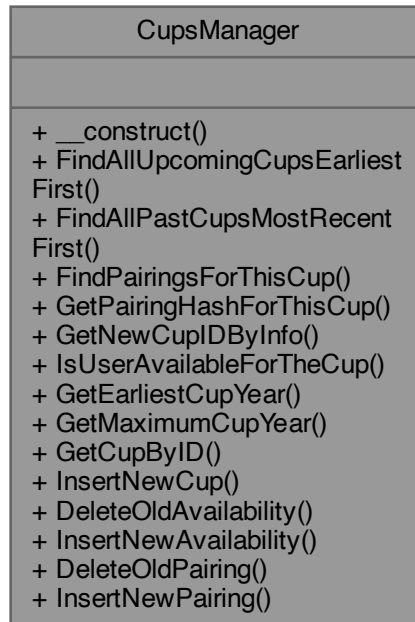
string

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

- model/Cup.php

3.5 CupsManager Class Reference

Collaboration diagram for CupsManager:



Public Member Functions

- [__construct](#) (mysqli \$mysqli)
- [FindAllUpcomingCupsEarliestFirst](#) ()
- [FindAllPastCupsMostRecentFirst](#) ()
- [FindPairingsForThisCup](#) (\$cupID)
- [GetPairingHashForThisCup](#) (\$cupID)
- [GetNewCupIDByInfo](#) (\$name, \$date_begin, \$date_end)
- [IsUserAvailableForTheCup](#) (\$userID, \$cupID)
- [GetEarliestCupYear](#) ()
- [GetMaximumCupYear](#) ()
- [GetCupByID](#) (\$cupID)
- [InsertNewCup](#) (\$name, \$date_begin, \$date_end, \$club, \$content)
- [DeleteOldAvailability](#) (\$cupID)
- [InsertNewAvailability](#) (\$cupID, \$userID, \$attendanceFlag)
- [DeleteOldPairing](#) (\$cupID)
- [InsertNewPairing](#) (\$cupID, \$posID, \$userID)

3.5.1 Detailed Description

[CupsManager](#) has API functions to handle [Cup](#) object/s and delivers it through web application.

3.5.2 Constructor & Destructor Documentation

3.5.2.1 __construct()

```
CupsManager::__construct (
    mysqli $mysqli )
```

Initialize [CupsManager](#) with live database connection.

Parameters

mysqli	<i>\$mysqli</i>	
--------	-----------------	--

3.5.3 Member Function Documentation

3.5.3.1 DeleteOldAvailability()

```
CupsManager::DeleteOldAvailability (
    $cupID )
```

Deletes availabilities for [Cup](#).

Parameters

int	<i>\$cupID</i>	
-----	----------------	--

Returns

bool

3.5.3.2 DeleteOldPairing()

```
CupsManager::DeleteOldPairing (
    $cupID )
```

Delete old pairings for [Cup](#).

Parameters

int	<i>\$cupID</i>	
-----	----------------	--

Returns

bool

3.5.3.3 FindAllPastCupsMostRecentFirst()

```
CupsManager::FindAllPastCupsMostRecentFirst ( )
```

Find past cups and sort by the most recent first.

Returns

array<Cup>

3.5.3.4 FindAllUpcomingCupsEarliestFirst()

```
CupsManager::FindAllUpcomingCupsEarliestFirst ( )
```

Find future cups and sort by earliest first.

Returns

array<Cup>

3.5.3.5 FindPairingsForThisCup()

```
CupsManager::FindPairingsForThisCup (
    $cupID )
```

Retrieve existing pairings list [PairPositionUser](#) that is ready for this [Cup](#).

Parameters

int	<i>\$cupID</i>	
-----	----------------	--

Returns

array<PairPositionUser>

3.5.3.6 GetCupByID()

```
CupsManager::GetCupByID (
    $cupID )
```

Get [Cup](#) by id.

Parameters

int	<i>\$cupID</i>	
-----	----------------	--

Returns

Cup|null

3.5.3.7 GetEarliestCupYear()

```
CupsManager::GetEarliestCupYear ( )
```

Get year of most early [Cup](#) in web application.

Returns

int

3.5.3.8 GetMaximumCupYear()

```
CupsManager::GetMaximumCupYear ( )
```

Get most recent year.

Returns

int

3.5.3.9 GetNewCupIDByInfo()

```
CupsManager::GetNewCupIDByInfo (
    $name,
    $date_begin,
    $date_end )
```

Return [Cup](#) based on its information.

Parameters

string	<i>\$name</i>	
string	<i>\$date_begin</i>	
string	<i>\$date_end</i>	

Returns

Cup|null

3.5.3.10 GetPairingHashForThisCup()

```
CupsManager::GetPairingHashForThisCup (
    $cupID )
```

Get hash of database result of pairing for this [Cup](#).

Parameters

int	<i>\$cupID</i>	
-----	----------------	--

Returns

string

3.5.3.11 InsertNewAvailability()

```
CupsManager::InsertNewAvailability (
    $cupID,
    $userID,
    $attendanceFlag )
```

Insert new availability.

Parameters

int	<i>\$cupID</i>	
int	<i>\$userID</i>	
bool	<i>\$attendanceFlag</i>	

Returns

bool

3.5.3.12 InsertNewCup()

```
CupsManager::InsertNewCup (
    $name,
    $date_begin,
    $date_end,
    $club,
    $content )
```

Create new cup in the web application.

Parameters

string	<i>\$name</i>	
string	<i>\$date_begin</i>	
string	<i>\$date_end</i>	
int	<i>\$club</i>	
string	<i>\$content</i>	

Returns

bool

3.5.3.13 InsertNewPairing()

```
CupsManager::InsertNewPairing (
    $cupID,
    $posID,
    $userID )
```

Insert new pairing for [Cup](#).

Parameters

int	<i>\$cupID</i>	
int	<i>\$posID</i>	
int	<i>\$userID</i>	

Returns

bool

3.5.3.14 IsUserAvailableForTheCup()

```
CupsManager::IsUserAvailableForTheCup (
    $userID,
    $cupID )
```

Check if [User](#) is available for the [Cup](#).

Parameters

int	<i><code>\$userID</code></i>	
int	<i><code>\$cupID</code></i>	

Exceptions

<i><code>RuntimeException</code></i>	
--------------------------------------	--

Returns

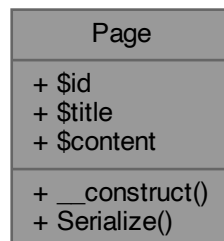
bool

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

- model/CupsManager.php

3.6 Page Class Reference

Collaboration diagram for Page:



Public Member Functions

- [__construct](#) (`$id`, `$title`, `$content`)
- [Serialize](#) ()

Public Attributes

- `$id`
- `$title`
- `$content`

3.6.1 Detailed Description

[Page](#) is static website page with information in web application. It has some title and content.

3.6.2 Constructor & Destructor Documentation

3.6.2.1 __construct()

```
Page::__construct (
    $id,
    $title,
    $content )
```

Ctor of [Page](#) object for web application

Parameters

int	<i>\$id</i>	
string	<i>\$title</i>	
string	<i>\$content</i>	

3.6.3 Member Function Documentation

3.6.3.1 Serialize()

```
Page::Serialize ( )
```

Serialize outputs 3/3 these members: id, title, content

Returns

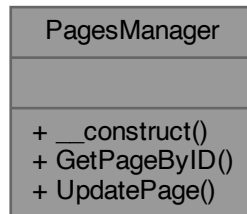
string

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

- model/Page.php

3.7 PagesManager Class Reference

Collaboration diagram for PagesManager:



Public Member Functions

- [__construct](#) (mysqli \$mysqli)
- [getPageById](#) (\$id)
- [updatePage](#) (\$id, \$title, \$content)

3.7.1 Detailed Description

[PagesManager](#) has API functions to handle [Page](#) object and delivers it through web application.

3.7.2 Constructor & Destructor Documentation

3.7.2.1 __construct()

```
PagesManager::__construct (
    mysqli $mysqli )
```

Initialize [PagesManager](#) with live database connection.

Parameters

mysqli	<i>\$mysqli</i>	
--------	-----------------	--

3.7.3 Member Function Documentation

3.7.3.1 GetPageByID()

```
PagesManager::GetPageByID (
    $id )
```

Get [Page](#) from database by its id.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

Page|null

3.7.3.2 UpdatePage()

```
PagesManager::UpdatePage (
    $id,
    $title,
    $content )
```

Updates [Page](#) in the web application.

Parameters

int	<i>\$id</i>	
string	<i>\$title</i>	
string	<i>\$content</i>	

Returns

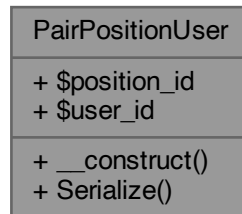
bool

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

- model/PagesManager.php

3.8 PairPositionUser Class Reference

Collaboration diagram for PairPositionUser:



Public Member Functions

- [__construct](#) (\$position_id, \$user_id)
- [Serialize](#) ()

Public Attributes

- [\\$position_id](#)
- [\\$user_id](#)

3.8.1 Detailed Description

[PairPositionUser](#) is tuple of [Position](#) and [User](#). It is used as pairing object for further use in web application.

3.8.2 Constructor & Destructor Documentation

3.8.2.1 __construct()

```
PairPositionUser::__construct (
    $position_id,
    $user_id )
```

Ctor of [PairPositionUser](#) object for web application

Parameters

int	<i>\$position</i> ↔ _id	
int	<i>\$user_id</i>	

3.8.3 Member Function Documentation

3.8.3.1 Serialize()

```
PairPositionUser::Serialize ( )
```

Serialize outputs 2/2 these members: position_id, user_id

Returns

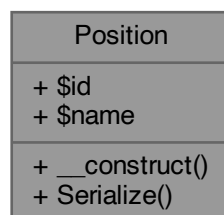
string

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

- model/PairPositionUser.php

3.9 Position Class Reference

Collaboration diagram for Position:



Public Member Functions

- [__construct](#) (\$id, \$name)
- [Serialize](#) ()

Public Attributes

- `$id`
- `$name`

3.9.1 Detailed Description

`Position` is object representing task for `Cup` that has to be performed by `User`. It has internal id based on which it is wired through the system internally.

3.9.2 Constructor & Destructor Documentation

3.9.2.1 `__construct()`

```
Position::__construct (
    $id,
    $name )
```

Ctor of `Position` object for web application

Parameters

int	<code>\$id</code>	
string	<code>\$name</code>	

3.9.3 Member Function Documentation

3.9.3.1 `Serialize()`

```
Position::Serialize ( )
```

Serialize outputs 2/2 these members: id, name

Returns

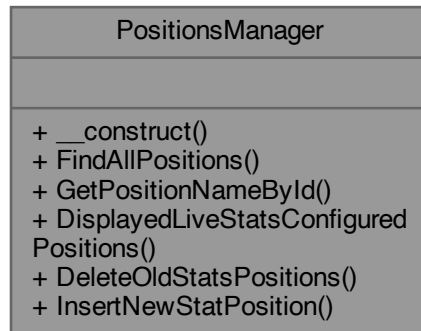
string

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

- `model/Position.php`

3.10 PositionsManager Class Reference

Collaboration diagram for PositionsManager:



Public Member Functions

- [__construct](#) (mysqli \$mysqli)
- [FindAllPositions](#) ()
- [GetPositionNameById](#) (\$posId)
- [DisplayedLiveStatsConfiguredPositions](#) ()
- [DeleteOldStatsPositions](#) ()
- [InsertNewStatPosition](#) (\$id)

3.10.1 Detailed Description

[PositionsManager](#) has API functions to handle [Position](#) object/s and delivers it through web application.

3.10.2 Constructor & Destructor Documentation

3.10.2.1 __construct()

```
PositionsManager::__construct (
    mysqli $mysqli )
```

Initialize [PositionsManager](#) with live database connection.

Parameters

mysqli	\$mysqli	
--------	----------	--

3.10.3 Member Function Documentation

3.10.3.1 DeleteOldStatsPositions()

```
PositionsManager::DeleteOldStatsPositions ( )
```

Delete setting of how positions should be displayed at public website.

Returns

bool

3.10.3.2 DisplayedLiveStatsConfiguredPositions()

```
PositionsManager::DisplayedLiveStatsConfiguredPositions ( )
```

List of positions in our specific order that we desire to display at public website for statistical purposes.

Returns

array<Position>

3.10.3.3 FindAllPositions()

```
PositionsManager::FindAllPositions ( )
```

Retrieve all work positions needed at [Cup](#) to be administered.

Returns

array<Position>

3.10.3.4 GetPositionNameById()

```
PositionsManager::GetPositionNameById (
    $posId )
```

Get [Position](#) name based on id.

Parameters

int	<i>\$pos</i> ↔ <i>Id</i>	
-----	-----------------------------	--

Returns

string

3.10.3.5 InsertNewStatPosition()

```
PositionsManager::InsertNewStatPosition (
    $id )
```

Insert new setting of how positions should be displayed at public website.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

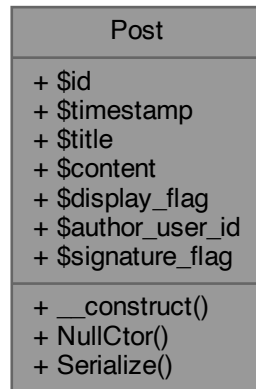
bool

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

- model/PositionsManager.php

3.11 Post Class Reference

Collaboration diagram for Post:



Public Member Functions

- [__construct](#) (\$id, \$timestamp, \$title, \$content, \$display_flag, \$author_user_id, \$signature_flag)
- [NullCtor](#) ()
- [Serialize](#) ()

Public Attributes

- `$id`
- `$timestamp`
- `$title`
- `$content`
- `$display_flag`
- `$author_user_id`
- `$signature_flag`

3.11.1 Detailed Description

[Post](#) is a static snippet of news for homepage of web application.

3.11.2 Constructor & Destructor Documentation

3.11.2.1 `__construct()`

```
Post::__construct (
    $id,
    $timestamp,
    $title,
    $content,
    $display_flag,
    $author_user_id,
    $signature_flag )
```

Ctor of [Post](#) object for web application

Parameters

int	<i>\$id</i>	
string	<i>\$timestamp</i>	
string	<i>\$title</i>	
string	<i>\$content</i>	
bool	<i>\$display_flag</i>	
int	<i>\$author_user_id</i>	
int	<i>\$signature_flag</i>	

3.11.3 Member Function Documentation

3.11.3.1 `NullCtor()`

```
Post::NullCtor ( )
```

TODO - Constructor of [Post](#) with nothing inside. Not working yet.

Returns

void

3.11.3.2 `Serialize()`

```
Post::Serialize ( )
```

Serialize outputs 7/7 these members: id, timestamp, title, content, display_flag, author_user_id, signature_flag

Returns

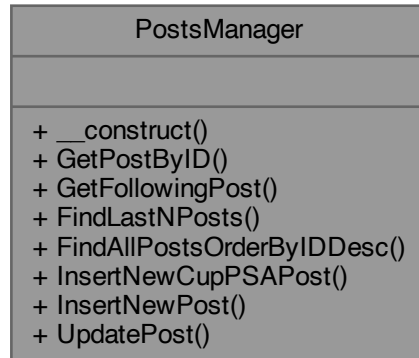
string

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

- model/Post.php

3.12 PostsManager Class Reference

Collaboration diagram for PostsManager:



Public Member Functions

- [__construct](#) (mysqli \$mysqli)
- [GetPostByID](#) (\$id)
- [GetFollowingPost](#) (\$id)
- [FindLastNPosts](#) (\$N)
- [FindAllPostsOrderByIDDesc](#) ()
- [InsertNewCupPSAPost](#) (\$cupTitle, \$cupID, \$date_start, \$date_end, \$authorID, \$clubAbbrev)
- [InsertNewPost](#) (\$title, \$content, \$display_flag, \$author, \$signature_flag)
- [UpdatePost](#) (\$id, \$title, \$content, \$display_flag, \$signature_flag)

3.12.1 Detailed Description

[PostsManager](#) has API functions to handle [Post](#) object/s and delivers it through web application.

3.12.2 Constructor & Destructor Documentation

3.12.2.1 `__construct()`

```

PostsManager::__construct (
    mysqli $mysqli )
  
```

Initialize [PostsManager](#) with live database connection.

Parameters

mysqli	<i>\$mysqli</i>	
--------	-----------------	--

3.12.3 Member Function Documentation

3.12.3.1 FindAllPostsOrderByIDDesc()

```
PostsManager::FindAllPostsOrderByIDDesc ( )
```

Finds all posts by descending id (earliest first).

Returns

array<Post>

3.12.3.2 FindLastNPosts()

```
PostsManager::FindLastNPosts (
    $N )
```

Finds last N posts based on provided N.

Parameters

int	<i>\$N</i>	
-----	------------	--

Returns

array<Post>

3.12.3.3 GetFollowingPost()

```
PostsManager::GetFollowingPost (
    $id )
```

Get following [Post](#) compared to one that we have by id.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

Post|null

3.12.3.4 GetPostByID()

```
PostsManager::GetPostByID (
    $id )
```

Get [Post](#) object by its id.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

Post|null

3.12.3.5 InsertNewCupPSAPost()

```
PostsManager::InsertNewCupPSAPost (
    $cupTitle,
    $cupID,
    $date_start,
    $date_end,
    $authorID,
    $clubAbbrev )
```

[Post](#) that new [Cup](#) was created as public state announcement via this function.

Parameters

string	<i>\$cupTitle</i>	
int	<i>\$cupID</i>	
string	<i>\$date_start</i>	
string	<i>\$date_end</i>	
int	<i>\$authorID</i>	
string	<i>\$clubAbbrev</i>	

Returns

void

3.12.3.6 InsertNewPost()

```
PostsManager::InsertNewPost (
    $title,
    $content,
    $display_flag,
    $author,
    $signature_flag )
```

Insert new news [Post](#) for visitors of web application.

Parameters

string	<i>\$title</i>	
string	<i>\$content</i>	
bool	<i>\$display_flag</i>	
int	<i>\$author</i>	
int	<i>\$signature_flag</i>	

Returns

bool

3.12.3.7 UpdatePost()

```
PostsManager::UpdatePost (
    $id,
    $title,
    $content,
    $display_flag,
    $signature_flag )
```

Update already existing [Post](#).

Parameters

int	<i>\$id</i>	
string	<i>\$title</i>	
string	<i>\$content</i>	
int	<i>\$display_flag</i>	
int	<i>\$signature_flag</i>	

Returns

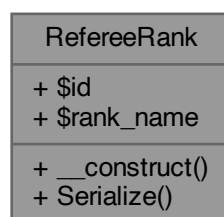
bool

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

- `model/PostsManager.php`

3.13 RefereeRank Class Reference

Collaboration diagram for RefereeRank:



Public Member Functions

- [__construct](#) (\$id, \$rank_name)
- [Serialize](#) ()

Public Attributes

- `$id`
- `$rank_name`

3.13.1 Detailed Description

[RefereeRank](#) is object representing rank in referee hierarchy. It has internal id and caption of appropriate level in Czech Swimming Federation hierarchy.

3.13.2 Constructor & Destructor Documentation

3.13.2.1 __construct()

```
RefereeRank::__construct (
    $id,
    $rank_name )
```

Ctor of [RefereeRank](#) object for web application

Parameters

int	<i>\$id</i>	
string	<i>\$rank_name</i>	

3.13.3 Member Function Documentation

3.13.3.1 Serialize()

```
RefereeRank::Serialize ( )
```

Serialize outputs 2/2 these members: id, rank_name

Returns

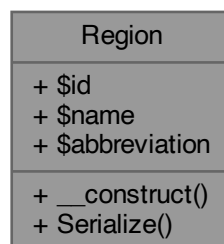
string

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

- model/RefereeRank.php

3.14 Region Class Reference

Collaboration diagram for Region:



Public Member Functions

- [__construct](#) (\$id, \$name, \$abbreviation)
- [Serialize](#) ()

Public Attributes

- `$id`
- `$name`
- `$abbreviation`

3.14.1 Detailed Description

[Region](#) is administrative object to which we affiliate each [Club](#) which each has some [User](#) objects. [Region](#) has no other purpose than just umbrella of regional clubs.

3.14.2 Constructor & Destructor Documentation

3.14.2.1 `__construct()`

```
Region::__construct (
    $id,
    $name,
    $abbreviation )
```

Ctor of [Region](#) object for web application

Parameters

int	<i>\$id</i>	
string	<i>\$name</i>	
string	<i>\$abbreviation</i>	

3.14.3 Member Function Documentation

3.14.3.1 `Serialize()`

```
Region::Serialize ( )
```

Serialize outputs 3/3 these members: id, name, abbreviation

Returns

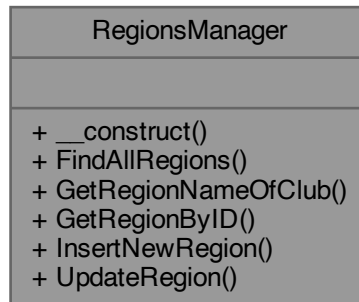
string

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

- `model/Region.php`

3.15 RegionsManager Class Reference

Collaboration diagram for RegionsManager:



Public Member Functions

- [__construct](#) (mysqli \$mysqli)
- [FindAllRegions](#) ()
- [GetRegionNameOfClub](#) (\$regionID)
- [GetRegionByID](#) (\$regionID)
- [InsertNewRegion](#) (\$name, \$abbrev)
- [UpdateRegion](#) (\$id, \$name, \$abbrev)

3.15.1 Detailed Description

[RegionsManager](#) has API functions to handle [Region](#) object/s and delivers it through web application.

3.15.2 Constructor & Destructor Documentation

3.15.2.1 `__construct()`

```
RegionsManager::__construct (
    mysqli $mysqli )
```

Initialize [RegionsManager](#) with live database connection.

Parameters

mysqli	<i>\$mysqli</i>	
--------	-----------------	--

3.15.3 Member Function Documentation

3.15.3.1 FindAllRegions()

```
RegionsManager::FindAllRegions ( )
```

Find all regions in under administration.

Returns

array<Region>

3.15.3.2 GetRegionByID()

```
RegionsManager::GetRegionByID (
    $regionID )
```

Return [Region](#) provided its id.

Parameters

int	<i>\$regionID</i>	
-----	-------------------	--

Returns

Region|null

3.15.3.3 GetRegionNameOfClub()

```
RegionsManager::GetRegionNameOfClub (
    $regionID )
```

Return name of [Region](#) in which [Club](#) is located.

Parameters

int	<i>\$regionID</i>	
-----	-------------------	--

Returns

string

3.15.3.4 InsertNewRegion()

```
RegionsManager::InsertNewRegion (
    $name,
    $abbrev )
```

Insert new [Region](#) into the web application.

Parameters

string	<i>\$name</i>	
string	<i>\$abbrev</i>	

Returns

bool

3.15.3.5 UpdateRegion()

```
RegionsManager::UpdateRegion (
    $id,
    $name,
    $abbrev )
```

Update already existing [Region](#) in the web application.

Parameters

int	<i>\$id</i>	
string	<i>\$name</i>	
string	<i>\$abbrev</i>	

Returns

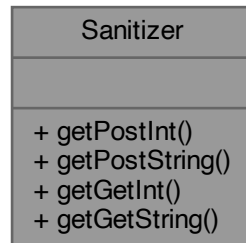
bool

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

- model/RegionsManager.php

3.16 Sanitizer Class Reference

Collaboration diagram for Sanitizer:



Static Public Member Functions

- static [getPostInt](#) (\$key)
- static [getPostString](#) (\$key)
- static [getGetInt](#) (\$key)
- static [getGetString](#) (\$key)

3.16.1 Detailed Description

[Sanitizer](#) class serves as anti XSS protection. It handles Int and String parameters provided via HTTP GET and HTTP POST.

3.16.2 Member Function Documentation

3.16.2.1 getGetInt()

```
static Sanitizer::getGetInt (
    $key ) [static]
```

Process Integer provided via. HTTP GET.

Parameters

int	\$key	
-----	-------	--

Exceptions

<i>RuntimeException</i>	
-------------------------	--

Returns

int

3.16.2.2 getGetString()

```
static Sanitizer::getGetString (
    $key ) [static]
```

Process String provided via. HTTP GET.

Parameters

string	<i>\$key</i>	
--------	--------------	--

Exceptions

<i>RuntimeException</i>	
-------------------------	--

Returns

string

3.16.2.3 getPostInt()

```
static Sanitizer::getPostInt (
    $key ) [static]
```

Process Integer provided via. HTTP POST.

Parameters

int	<i>\$key</i>	
-----	--------------	--

Exceptions

<i>RuntimeException</i>	
-------------------------	--

Returns

int

3.16.2.4 getPostString()

```
static Sanitizer::getPostString (
    $key ) [static]
```

Process String provided via. HTTP POST.

Parameters

string	\$key	
--------	-------	--

Exceptions

<i>RuntimeException</i>	
-------------------------	--

Returns

string

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

- model/Sanitizer.php

3.17 StatPositionCnt Class Reference

Collaboration diagram for StatPositionCnt:

StatPositionCnt
+ \$position_id + \$cnt
+ __construct() + Serialize()

Public Member Functions

- [__construct](#) (\$position_id, \$cnt)
- [Serialize](#) ()

Public Attributes

- [\\$position_id](#)
- [\\$cnt](#)

3.17.1 Detailed Description

[StatPositionCnt](#) is object providing count of statistics of [Position](#).

3.17.2 Constructor & Destructor Documentation

3.17.2.1 __construct()

```
StatPositionCnt::__construct (
    $position_id,
    $cnt )
```

Ctor of [StatPositionCnt](#) object for web application

Parameters

int	<i>\$position_id</i>	
int	<i>\$cnt</i>	

3.17.3 Member Function Documentation

3.17.3.1 Serialize()

```
StatPositionCnt::Serialize ( )
```

Serialize outputs 2/2 these members: position_id, cnt

Returns

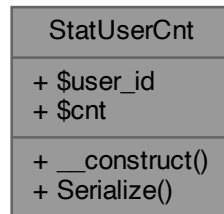
string

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

- model/StatPositionCnt.php

3.18 StatUserCnt Class Reference

Collaboration diagram for StatUserCnt:



Public Member Functions

- [__construct](#) (\$user_id, \$cnt)
- [Serialize](#) ()

Public Attributes

- [\\$user_id](#)
- [\\$cnt](#)

3.18.1 Detailed Description

[StatUserCnt](#) is object providing statistics count of [User](#).

3.18.2 Constructor & Destructor Documentation

3.18.2.1 __construct()

```
StatUserCnt::__construct (
    $user_id,
    $cnt )
```

Ctor of [StatUserCnt](#) object for web application

Parameters

int	<i>\$user</i> ↔ _id	
int	<i>\$cnt</i>	

3.18.3 Member Function Documentation

3.18.3.1 Serialize()

```
StatUserCnt::Serialize ( )
```

Serialize outputs 2/2 these members: user_id, cnt

Returns

string

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

- model/StatUserCnt.php

3.19 User Class Reference

Collaboration diagram for User:



Public Member Functions

- [__construct](#) (\$id, \$first_name, \$last_name, \$email, \$approved_flag, \$rights, \$referee_rank_id, \$affiliation_↔club_id)
- [SerializeFull](#) ()
- [SerializeSlim](#) ()
- [SerializeTag](#) ()

Public Attributes

- \$id
- \$first_name
- \$last_name
- \$email
- \$approved_flag
- \$rights
- \$referee_rank_id
- \$affiliation_club_id

3.19.1 Detailed Description

[User](#) is referee in the system. [User](#) is affiliated to [Club](#) has one of [UserRights](#) and one of tier from [RefereeRank](#). His statistics are presented via [StatUserCnt](#) in public sites via [PositionsManager](#).

3.19.2 Constructor & Destructor Documentation

3.19.2.1 __construct()

```
User::__construct (
    $id,
    $first_name,
    $last_name,
    $email,
    $approved_flag,
    $rights,
    $referee_rank_id,
    $affiliation_club_id )
```

Ctor of [User](#) object for web application

Parameters

int	<i>\$id</i>	
string	<i>\$first_name</i>	
string	<i>\$last_name</i>	
string	<i>\$email</i>	
bool	<i>\$approved_flag</i>	
int	<i>\$rights</i>	
int	<i>\$referee_rank_id</i>	
int	<i>\$affiliation_club_↔id</i>	

3.19.3 Member Function Documentation

3.19.3.1 SerializeFull()

```
User::SerializeFull ( )
```

Serialize outputs 8/8 these members: id, first_name, last_name, email, approved_flag, rights, referee_rank_id, affiliation_club_id

Returns

string

3.19.3.2 SerializeSlim()

```
User::SerializeSlim ( )
```

Serialize outputs 5/8 these members: id, first_name, last_name, -N/A-, -N/A-, -N/A-, referee_rank_id, affiliation_↔ club_id

Returns

string

3.19.3.3 SerializeTag()

```
User::SerializeTag ( )
```

Serialize outputs 2/8 these members: -NULL-, first_name, last_name, -NULL-, -NULL-, -NULL-, -NULL-, -NULL-

Returns

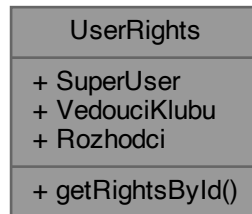
string

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

- model/User.php

3.20 UserRights Class Reference

Collaboration diagram for UserRights:



Static Public Member Functions

- static [getRightsById](#) (\$value)

Public Attributes

- const **SuperUser** = 2
- const **VedouciKlubu** = 1
- const **Rozhodci** = 0

3.20.1 Detailed Description

[UserRights](#) has roles with rights numbers in descending order. The larger the number the higher the access. It can also return name of the role based on its number. [User](#) can access page of same and lower rights.

3.20.2 Member Function Documentation

3.20.2.1 [getRightsById\(\)](#)

```
static UserRights::getRightsById (
    $value ) [static]
```

Returns name of the role based on its number from this [UserRights](#) enumeration class.

Parameters

int	<i>\$value</i>	
-----	----------------	--

Exceptions

<i>RuntimeException</i>	
-------------------------	--

Returns

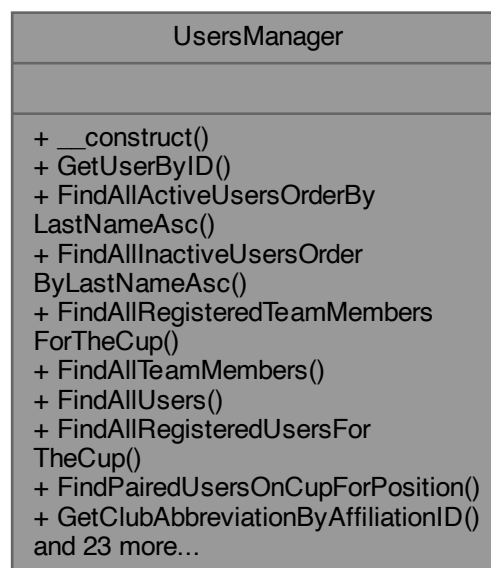
string

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

- model/Auth.php

3.21 UsersManager Class Reference

Collaboration diagram for UsersManager:



Public Member Functions

- [__construct](#) (mysqli \$mysqli)
- [GetUserById](#) (\$id)
- [FindAllActiveUsersOrderByLastNameAsc](#) ()
- [FindAllInactiveUsersOrderByLastNameAsc](#) ()
- [FindAllRegisteredTeamMembersForTheCup](#) (\$cupId, \$teamId)
- [FindAllTeamMembers](#) (\$teamId)
- [FindAllUsers](#) ()

- [FindAllRegisteredUsersForTheCup](#) (\$cupId)
- [FindPairedUsersOnCupForPosition](#) (\$cupId, \$position)
- [GetClubAbbreviationByAffiliationID](#) (\$id)
- [GetClubNameByAffiliationID](#) (\$id)
- [GetUserFullNameByID](#) (\$userId)
- [FindAllRefereeRanks](#) ()
- [GetRefereeRank](#) (\$id)
- [IsComing](#) (\$cupID, \$userID)
- [RetComingCSSClass](#) (\$cupID, \$userID)
- [RetStringComingFlag](#) (\$cupID, \$userID)
- [GetUserEmailByID](#) (\$userID)
- [SetPasswordForUser](#) (\$uid, \$passwd)
- [SetLoginEmailForUser](#) (\$uid, \$email)
- [SetRefereeRankForUser](#) (\$uid, \$rank)
- [SetAvailabilityRegister](#) (\$uid, \$cid)
- [SetAvailabilityCanGo](#) (\$uid, \$cid)
- [SetAvailabilityCantGo](#) (\$uid, \$cid)
- [IsEmailPresentAlready](#) (\$email)
- [IsUserWithIDPresentAlready](#) (\$id)
- [RegisterUser](#) (\$first_name, \$last_name, \$email, \$password, \$rights, \$refRank, \$klubaffil)
- [EmailNewPersonRegistered](#) (\$email, \$password)
- [SetApprovedForUser](#) (\$userID)
- [CountCupsAttendanceOfUserGivenYear](#) (\$userID, \$year)
- [CountOverallStatisticsOfUserGivenYear](#) (\$userID, \$year)
- [CountClubSeasonalStatistics](#) (\$clubID, \$year)
- [LoginCandidateToBeAuthorized](#) (\$email)

3.21.1 Detailed Description

[UsersManager](#) has API functions to handle [User](#) object/s and delivers is through web application.

3.21.2 Constructor & Destructor Documentation

3.21.2.1 `__construct()`

```
UsersManager::__construct (
    mysqli $mysqli )
```

Initialize [UsersManager](#) with live database connection.

Parameters

mysqli	<i>\$mysqli</i>	
--------	-----------------	--

3.21.3 Member Function Documentation

3.21.3.1 CountClubSeasonalStatistics()

```
UsersManager::CountClubSeasonalStatistics (
    $clubID,
    $year )
```

Stats function that returns [User](#) attendances for cups in given year.

Parameters

int	<i>\$clubID</i>	
int	<i>\$year</i>	

Returns

array<StatUserCnt>

3.21.3.2 CountCupsAttendanceOfUserGivenYear()

```
UsersManager::CountCupsAttendanceOfUserGivenYear (
    $userID,
    $year )
```

Stats function that returns howmany cups [User](#) has attended for given year.

Parameters

int	<i>\$userID</i>	
int	<i>\$year</i>	

Returns

int

3.21.3.3 CountOverallStatisticsOfUserGivenYear()

```
UsersManager::CountOverallStatisticsOfUserGivenYear (
    $userID,
    $year )
```

Stats function that returns pairs [StatPositionCnt](#) of [User](#) for given year.

Parameters

int	<i>\$userID</i>	
int	<i>\$year</i>	

Returns

array<StatPositionCnt>

3.21.3.4 EmailNewPersonRegistered()

```
UsersManager::EmailNewPersonRegistered (
    $email,
    $password )
```

TODO - function that e-mails person and notifies him/her that registration to our web application happened.

Parameters

mixed	<i>\$email</i>	
mixed	<i>\$password</i>	

Returns

bool

3.21.3.5 FindAllActiveUsersOrderByLastNameAsc()

```
UsersManager::FindAllActiveUsersOrderByLastNameAsc ( )
```

Get list of active users alphabetically.

Returns

array<User>

3.21.3.6 FindAllInactiveUsersOrderByLastNameAsc()

```
UsersManager::FindAllInactiveUsersOrderByLastNameAsc ( )
```

Get list of inactive user alphabetically.

Returns

array<User>

3.21.3.7 FindAllRefereeRanks()

```
UsersManager::FindAllRefereeRanks ( )
```

Find all referee ranks in the web application.

Returns

array<RefereeRank>

3.21.3.8 FindAllRegisteredTeamMembersForTheCup()

```
UsersManager::FindAllRegisteredTeamMembersForTheCup (
    $cupId,
    $teamId )
```

Get list of inactive users alphabetically.

Parameters

int	<i>\$cupId</i>	
int	<i>\$teamId</i>	

Returns

array<User>

3.21.3.9 FindAllRegisteredUsersForTheCup()

```
UsersManager::FindAllRegisteredUsersForTheCup (
    $cupId )
```

List of registered users for [Cup](#).

Parameters

int	<i>\$cupId</i>	
-----	----------------	--

Returns

array<User>

3.21.3.10 FindAllTeamMembers()

```
UsersManager::FindAllTeamMembers (
    $teamId )
```

List of registered [User](#) teammates for desired [Cup](#).

Parameters

int	<i>\$teamId</i>	
-----	-----------------	--

Returns

array<[User](#)>

3.21.3.11 FindAllUsers()

```
UsersManager::FindAllUsers ( )
```

List of all users from the web application.

Returns

array<[User](#)>

3.21.3.12 FindPairedUsersOnCupForPosition()

```
UsersManager::FindPairedUsersOnCupForPosition (
    $cupId,
    $position )
```

Get list of [User](#) s that are on [Cup](#) on [Position](#).

Parameters

int	<i>\$cupId</i>	
int	<i>\$position</i>	

Returns

array<[User](#)>

3.21.3.13 GetClubAbbreviationByAffiliationID()

```
UsersManager::GetClubAbbreviationByAffiliationID (
    $id )
```

Retrieve club abbreviation - short name code, provided [Club](#) id.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

string

3.21.3.14 GetClubNameByAffiliationID()

```
UsersManager::GetClubNameByAffiliationID (
    $id )
```

Retrieve [Club](#) name by id.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

string

3.21.3.15 GetRefereeRank()

```
UsersManager::GetRefereeRank (
    $id )
```

Get referee rank name by its id.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

string

3.21.3.16 GetUserByID()

```
UsersManager::GetUserByID (
    $id )
```

Get [User](#) by provided id.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

User|null

3.21.3.17 GetUserEmailByID()

```
UsersManager::GetUserEmailByID (
    $userID )
```

Retrieve e-mail of [User](#) based on his id.

Parameters

mixed	<i>\$userID</i>	
-------	-----------------	--

Returns

mixed

3.21.3.18 GetUserFullNameByID()

```
UsersManager::GetUserFullNameByID (
    $userID )
```

Retrieve full name (first_name + last_name) of [User](#) by provided id.

Parameters

int	<i>\$userID</i>	
-----	-----------------	--

Returns

string

3.21.3.19 IsComing()

```
UsersManager::IsComing (
    $cupID,
    $userID )
```

Answer T/F if [User](#) is coming to [Cup](#).

Parameters

int	<i>\$cupID</i>	
int	<i>\$userID</i>	

Returns

bool

3.21.3.20 IsEmailPresentAlready()

```
UsersManager::IsEmailPresentAlready (
    $email )
```

Ask if [User](#) with this e-mail is already present in the system.

Parameters

string	<i>\$email</i>	
--------	----------------	--

Returns

bool

3.21.3.21 IsUserWithIDPresentAlready()

```
UsersManager::IsUserWithIDPresentAlready (
    $id )
```

Ask if [User](#) with this id is already present in the system.

Parameters

mixed	<i>\$id</i>	
-------	-------------	--

Returns

bool

3.21.3.22 LoginCandidateToBeAuthorized()

```
UsersManager::LoginCandidateToBeAuthorized (
    $email )
```

TODO LoginCandidateToBeAuthorized / mby depr.

Parameters

string	<i>\$email</i>	
--------	----------------	--

Returns

array|bool|null

3.21.3.23 RegisterUser()

```
UsersManager::RegisterUser (
    $first_name,
    $last_name,
    $email,
    $password,
    $rights,
    $refRank,
    $klubaffil )
```

Register new user in the web application.

Parameters

string	<i>\$first_name</i>	
string	<i>\$last_name</i>	
string	<i>\$email</i>	
string	<i>\$password</i>	
int	<i>\$rights</i>	
int	<i>\$refRank</i>	
int	<i>\$klubaffil</i>	

Returns

bool

3.21.3.24 RetComingCSSClass()

```
UsersManager::RetComingCSSClass (
    $cupID,
    $userID )
```

Return CSS class for element if [User](#) is/not coming for [Cup](#).

Parameters

int	<i>\$cupID</i>	
int	<i>\$userID</i>	

Returns

string

3.21.3.25 RetStringComingFlag()

```
UsersManager::RetStringComingFlag (
    $cupID,
    $userID )
```

Return string flag of coming/not coming as "1" or "0"

Parameters

int	<i>\$cupID</i>	
int	<i>\$userID</i>	

Returns

string

3.21.3.26 SetApprovedForUser()

```
UsersManager::SetApprovedForUser (
    $userID )
```

Set approved flag for new [User](#) so he/she can use the system.

Parameters

int	<i>\$userID</i>	
-----	-----------------	--

Returns

bool

3.21.3.27 SetAvailabilityCanGo()

```
UsersManager::SetAvailabilityCanGo (
    $uid,
    $cid )
```

Flag available [User](#) as can go - TRUE.

Parameters

int	<i>\$uid</i>	
int	<i>\$cid</i>	

Returns

bool

3.21.3.28 SetAvailabilityCantGo()

```
UsersManager::SetAvailabilityCantGo (
    $uid,
    $cid )
```

Flag available [User](#) as can't go - FALSE.

Parameters

int	<i>\$uid</i>	
int	<i>\$cid</i>	

Returns

bool

3.21.3.29 SetAvailabilityRegister()

```
UsersManager::SetAvailabilityRegister (
    $uid,
    $cid )
```

Set [User](#)'s availability for cup.

Parameters

int	<i>\$uid</i>	
int	<i>\$cid</i>	

Returns

bool

3.21.3.30 SetLoginEmailForUser()

```
UsersManager::SetLoginEmailForUser (
    $uid,
    $email )
```

Set new login email for [User](#) as sysadmin.

Parameters

int	<i>\$uid</i>	
string	<i>\$email</i>	

Returns

bool

3.21.3.31 SetPasswordForUser()

```
UsersManager::SetPasswordForUser (
    $uid,
    $passwd )
```

Set new password for [User](#) as sysadmin.

Parameters

int	<i>\$uid</i>	
string	<i>\$passwd</i>	

Returns

bool

3.21.3.32 SetRefereeRankForUser()

```
UsersManager::SetRefereeRankForUser (
    $uid,
    $rank )
```

Set new referee rank for [User](#) as sysadmin.

Parameters

int	<i>\$uid</i>	
int	<i>\$rank</i>	

Returns

bool

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

- model/UsersManager.php

Index

- __construct
 - Club, [7](#)
 - ClubsManager, [9](#)
 - Cup, [12](#)
 - CupsManager, [15](#)
 - Page, [21](#)
 - PagesManager, [22](#)
 - PairPositionUser, [24](#)
 - Position, [26](#)
 - PositionsManager, [27](#)
 - Post, [30](#)
 - PostsManager, [32](#)
 - RefereeRank, [36](#)
 - Region, [38](#)
 - RegionsManager, [39](#)
 - StatPositionCnt, [45](#)
 - StatUserCnt, [46](#)
 - User, [48](#)
 - UsersManager, [52](#)
- Auth, [5](#)
 - requireRole, [5](#)
- Club, [6](#)
 - __construct, [7](#)
 - SerializeFull, [7](#)
 - SerializeSlim, [8](#)
- ClubsManager, [8](#)
 - __construct, [9](#)
 - FindAllClubs, [9](#)
 - GetClubAffiliationToRegion, [9](#)
 - GetClubByID, [10](#)
 - InsertNewClub, [10](#)
 - UpdateClub, [10](#)
- CountClubSeasonalStatistics
 - UsersManager, [53](#)
- CountCupsAttendanceOfUserGivenYear
 - UsersManager, [53](#)
- CountOverallStatisticsOfUserGivenYear
 - UsersManager, [53](#)
- Cup, [11](#)
 - __construct, [12](#)
 - SerializeFull, [13](#)
 - SerializeSlim, [13](#)
- CupsManager, [14](#)
 - __construct, [15](#)
 - DeleteOldAvailability, [15](#)
 - DeleteOldPairing, [15](#)
 - FindAllPastCupsMostRecentFirst, [16](#)
 - FindAllUpcomingCupsEarliestFirst, [16](#)
 - FindPairingsForThisCup, [16](#)
 - GetCupByID, [16](#)
 - GetEarliestCupYear, [17](#)
 - GetMaximumCupYear, [17](#)
 - GetNewCupIDByInfo, [17](#)
 - GetPairingHashForThisCup, [18](#)
 - InsertNewAvailability, [18](#)
 - InsertNewCup, [18](#)
 - InsertNewPairing, [19](#)
 - IsUserAvailableForTheCup, [19](#)
- DeleteOldAvailability
 - CupsManager, [15](#)
- DeleteOldPairing
 - CupsManager, [15](#)
- DeleteOldStatsPositions
 - PositionsManager, [28](#)
- DisplayedLiveStatsConfiguredPositions
 - PositionsManager, [28](#)
- EmailNewPersonRegistered
 - UsersManager, [54](#)
- FindAllActiveUsersOrderByLastNameAsc
 - UsersManager, [54](#)
- FindAllClubs
 - ClubsManager, [9](#)
- FindAllInactiveUsersOrderByLastNameAsc
 - UsersManager, [54](#)
- FindAllPastCupsMostRecentFirst
 - CupsManager, [16](#)
- FindAllPositions
 - PositionsManager, [28](#)
- FindAllPostsOrderByIDDesc
 - PostsManager, [33](#)
- FindAllRefereeRanks
 - UsersManager, [54](#)
- FindAllRegions
 - RegionsManager, [40](#)
- FindAllRegisteredTeamMembersForTheCup
 - UsersManager, [55](#)
- FindAllRegisteredUsersForTheCup
 - UsersManager, [55](#)
- FindAllTeamMembers
 - UsersManager, [55](#)
- FindAllUpcomingCupsEarliestFirst
 - CupsManager, [16](#)
- FindAllUsers
 - UsersManager, [56](#)
- FindLastNPosts

- PostsManager, [33](#)
- FindPairedUsersOnCupForPosition
 - UsersManager, [56](#)
- FindPairingsForThisCup
 - CupsManager, [16](#)
- GetClubAbbreviationByAffiliationID
 - UsersManager, [56](#)
- GetClubAffiliationToRegion
 - ClubsManager, [9](#)
- GetClubByID
 - ClubsManager, [10](#)
- GetClubNameByAffiliationID
 - UsersManager, [57](#)
- GetCupByID
 - CupsManager, [16](#)
- GetEarliestCupYear
 - CupsManager, [17](#)
- GetFollowingPost
 - PostsManager, [33](#)
- getGetInt
 - Sanitizer, [42](#)
- getGetString
 - Sanitizer, [43](#)
- GetMaximumCupYear
 - CupsManager, [17](#)
- GetNewCupIDByInfo
 - CupsManager, [17](#)
- GetPageByID
 - PagesManager, [22](#)
- GetPairingHashForThisCup
 - CupsManager, [18](#)
- GetPositionNameById
 - PositionsManager, [28](#)
- GetPostByID
 - PostsManager, [34](#)
- getPostInt
 - Sanitizer, [43](#)
- getPostString
 - Sanitizer, [44](#)
- GetRefereeRank
 - UsersManager, [57](#)
- GetRegionByID
 - RegionsManager, [40](#)
- GetRegionNameOfClub
 - RegionsManager, [40](#)
- getRightsById
 - UserRights, [50](#)
- GetUserByID
 - UsersManager, [58](#)
- GetUserEmailById
 - UsersManager, [58](#)
- GetUserFullNameById
 - UsersManager, [58](#)
- InsertNewAvailability
 - CupsManager, [18](#)
- InsertNewClub
 - ClubsManager, [10](#)
- InsertNewCup
 - CupsManager, [18](#)
- InsertNewCupPSAPost
 - PostsManager, [34](#)
- InsertNewPairing
 - CupsManager, [19](#)
- InsertNewPost
 - PostsManager, [35](#)
- InsertNewRegion
 - RegionsManager, [41](#)
- InsertNewStatPosition
 - PositionsManager, [29](#)
- IsComing
 - UsersManager, [59](#)
- IsEmailPresentAlready
 - UsersManager, [59](#)
- IsUserAvailableForTheCup
 - CupsManager, [19](#)
- IsUserWithIDPresentAlready
 - UsersManager, [59](#)
- LoginCandidateToBeAuthorized
 - UsersManager, [60](#)
- NullCtor
 - Post, [31](#)
- Page, [20](#)
 - __construct, [21](#)
 - Serialize, [21](#)
- PagesManager, [22](#)
 - __construct, [22](#)
 - GetPageByID, [22](#)
 - UpdatePage, [23](#)
- PairPositionUser, [24](#)
 - __construct, [24](#)
 - Serialize, [25](#)
- Position, [25](#)
 - __construct, [26](#)
 - Serialize, [26](#)
- PositionsManager, [27](#)
 - __construct, [27](#)
 - DeleteOldStatsPositions, [28](#)
 - DisplayedLiveStatsConfiguredPositions, [28](#)
 - FindAllPositions, [28](#)
 - GetPositionNameById, [28](#)
 - InsertNewStatPosition, [29](#)
- Post, [30](#)
 - __construct, [30](#)
 - NullCtor, [31](#)
 - Serialize, [31](#)
- PostsManager, [32](#)
 - __construct, [32](#)
 - FindAllPostsOrderByIDDesc, [33](#)
 - FindLastNPosts, [33](#)
 - GetFollowingPost, [33](#)
 - GetPostByID, [34](#)
 - InsertNewCupPSAPost, [34](#)
 - InsertNewPost, [35](#)

- UpdatePost, 35
- RefereeRank, 36
 - __construct, 36
 - Serialize, 37
- Region, 37
 - __construct, 38
 - Serialize, 38
- RegionsManager, 39
 - __construct, 39
 - FindAllRegions, 40
 - GetRegionByID, 40
 - GetRegionNameOfClub, 40
 - InsertNewRegion, 41
 - UpdateRegion, 41
- RegisterUser
 - UsersManager, 60
- requireRole
 - Auth, 5
- RetComingCSSClass
 - UsersManager, 61
- RetStringComingFlag
 - UsersManager, 61
- Sanitizer, 42
 - getGetInt, 42
 - getGetString, 43
 - getPostInt, 43
 - getPostString, 44
- Serialize
 - Page, 21
 - PairPositionUser, 25
 - Position, 26
 - Post, 31
 - RefereeRank, 37
 - Region, 38
 - StatPositionCnt, 45
 - StatUserCnt, 47
- SerializeFull
 - Club, 7
 - Cup, 13
 - User, 49
- SerializeSlim
 - Club, 8
 - Cup, 13
 - User, 49
- SerializeTag
 - User, 49
- SetApprovedForUser
 - UsersManager, 61
- SetAvailabilityCanGo
 - UsersManager, 62
- SetAvailabilityCantGo
 - UsersManager, 62
- SetAvailabilityRegister
 - UsersManager, 62
- SetLoginEmailForUser
 - UsersManager, 63
- SetPasswordForUser
 - UsersManager, 63
- SetRefereeRankForUser
 - UsersManager, 64
- StatPositionCnt, 44
 - __construct, 45
 - Serialize, 45
- StatUserCnt, 46
 - __construct, 46
 - Serialize, 47
- UpdateClub
 - ClubsManager, 10
- UpdatePage
 - PagesManager, 23
- UpdatePost
 - PostsManager, 35
- UpdateRegion
 - RegionsManager, 41
- User, 47
 - __construct, 48
 - SerializeFull, 49
 - SerializeSlim, 49
 - SerializeTag, 49
- UserRights, 50
 - getRightsById, 50
- UsersManager, 51
 - __construct, 52
 - CountClubSeasonalStatistics, 53
 - CountCupsAttendanceOfUserGivenYear, 53
 - CountOverallStatisticsOfUserGivenYear, 53
 - EmailNewPersonRegistered, 54
 - FindAllActiveUsersOrderByLastNameAsc, 54
 - FindAllInactiveUsersOrderByLastNameAsc, 54
 - FindAllRefereeRanks, 54
 - FindAllRegisteredTeamMembersForTheCup, 55
 - FindAllRegisteredUsersForTheCup, 55
 - FindAllTeamMembers, 55
 - FindAllUsers, 56
 - FindPairedUsersOnCupForPosition, 56
 - GetClubAbbreviationByAffiliationID, 56
 - GetClubNameByAffiliationID, 57
 - GetRefereeRank, 57
 - GetUserById, 58
 - GetUserEmailById, 58
 - GetUserFullNameById, 58
 - IsComing, 59
 - IsEmailPresentAlready, 59
 - IsUserWithIDPresentAlready, 59
 - LoginCandidateToBeAuthorized, 60
 - RegisterUser, 60
 - RetComingCSSClass, 61
 - RetStringComingFlag, 61
 - SetApprovedForUser, 61
 - SetAvailabilityCanGo, 62
 - SetAvailabilityCantGo, 62
 - SetAvailabilityRegister, 62
 - SetLoginEmailForUser, 63
 - SetPasswordForUser, 63
 - SetRefereeRankForUser, 64