

Webchat

Generated by Doxygen 1.8.2

Wed Jan 30 2013 08:53:04

Contents

1	Namespace Index	1
1.1	Packages	1
2	Hierarchical Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	Namespace Documentation	7
4.1	Package webchat	7
4.2	Package webchat.Communication	7
4.3	Package webchat.Controllers	7
4.4	Package webchat.Database	8
4.5	Package webchat.Filters	8
4.6	Package webchat.Helpers	8
4.7	Package webchat.Logging	8
4.8	Package webchat.Models	9
4.9	Package webchat.Models.Binders	9
4.10	Package webchat.Ping	9
4.11	Package webchat.Validators	9
5	Class Documentation	11
5.1	webchat.Filters.AuthenticationFilterAttribute Class Reference	11
5.1.1	Detailed Description	11
5.1.2	Member Function Documentation	11
5.1.2.1	OnAuthorization	11
5.2	webchat.Helpers.BindingHelper Class Reference	11
5.2.1	Detailed Description	12
5.2.2	Member Function Documentation	12
5.2.2.1	GetValue< T >	12
5.3	webchat.Controllers.ChatController Class Reference	12
5.3.1	Detailed Description	12

5.3.2	Member Function Documentation	13
5.3.2.1	Disconnect	13
5.3.2.2	Index	13
5.4	webchat.Models.ChatModel Class Reference	13
5.4.1	Detailed Description	13
5.4.2	Property Documentation	13
5.4.2.1	Rooms	13
5.4.2.2	Users	14
5.5	webchat.Database.Db Class Reference	14
5.5.1	Detailed Description	14
5.5.2	Constructor & Destructor Documentation	14
5.5.2.1	Db	14
5.5.3	Member Function Documentation	15
5.5.3.1	AddUser	15
5.5.3.2	Backup	15
5.5.3.3	DelUser	15
5.5.3.4	DelUserFromGlobalList	15
5.5.3.5	GetBackupRooms	15
5.5.3.6	GetRooms	15
5.5.3.7	GetRooms	16
5.5.3.8	GetUsers	16
5.5.3.9	IsPopulated	16
5.5.3.10	IsUser	16
5.6	webchat.Controllers.EventStreamController Class Reference	16
5.6.1	Detailed Description	17
5.6.2	Member Function Documentation	17
5.6.2.1	Get	17
5.7	webchat.Database.IDatabase Interface Reference	17
5.7.1	Detailed Description	18
5.7.2	Member Function Documentation	18
5.7.2.1	AddUser	18
5.7.2.2	Backup	18
5.7.2.3	DelUser	18
5.7.2.4	DelUserFromGlobalList	18
5.7.2.5	GetBackupRooms	18
5.7.2.6	GetRooms	19
5.7.2.7	GetRooms	19
5.7.2.8	GetUsers	19
5.7.2.9	IsPopulated	19
5.7.2.10	IsUser	19

5.8	webchat.Logging.ILogger Interface Reference	20
5.8.1	Detailed Description	20
5.8.2	Member Function Documentation	20
5.8.2.1	Log	20
5.8.2.2	Log	20
5.9	webchat.Controllers.IndexController Class Reference	20
5.9.1	Detailed Description	21
5.9.2	Member Function Documentation	21
5.9.2.1	Index	21
5.9.2.2	Index	21
5.10	webchat.Models.IndexModel Class Reference	21
5.10.1	Detailed Description	22
5.10.2	Property Documentation	22
5.10.2.1	Nick	22
5.11	webchat.Communication.IPublisher< T > Interface Template Reference	22
5.11.1	Detailed Description	22
5.11.2	Member Function Documentation	22
5.11.2.1	Publish	22
5.11.3	Property Documentation	23
5.11.3.1	Clients	23
5.12	webchat.Validators.JoinedRoomValidationAttribute Class Reference	23
5.12.1	Detailed Description	23
5.12.2	Constructor & Destructor Documentation	23
5.12.2.1	JoinedRoomValidationAttribute	23
5.12.3	Member Function Documentation	23
5.12.3.1	IsValid	23
5.13	webchat.Models.LeaveModel Class Reference	24
5.13.1	Detailed Description	24
5.13.2	Property Documentation	24
5.13.2.1	Room	24
5.14	webchat.Logging.Logger Class Reference	24
5.14.1	Detailed Description	24
5.14.2	Member Function Documentation	24
5.14.2.1	Log	24
5.14.2.2	Log	25
5.15	webchat.Controllers.MessageController Class Reference	25
5.15.1	Detailed Description	25
5.15.2	Member Function Documentation	25
5.15.2.1	Post	25
5.16	webchat.Models.MessageModel Class Reference	26

5.16.1 Detailed Description	26
5.16.2 Property Documentation	26
5.16.2.1 Message	26
5.16.2.2 Room	26
5.17 webchat.Models.Binders.MessageModelBinder Class Reference	26
5.17.1 Detailed Description	27
5.17.2 Member Function Documentation	27
5.17.2.1 BindProperty	27
5.18 webchat.MvcApplication Class Reference	27
5.18.1 Detailed Description	27
5.18.2 Member Function Documentation	27
5.18.2.1 Application_Start	27
5.18.3 Member Data Documentation	27
5.18.3.1 Db	27
5.18.3.2 Logger	28
5.18.3.3 Pub	28
5.19 webchat.Ping.Pinger Class Reference	28
5.19.1 Detailed Description	28
5.19.2 Constructor & Destructor Documentation	28
5.19.2.1 Pinger	28
5.19.3 Member Function Documentation	28
5.19.3.1 Ping	28
5.20 webchat.Controllers.PongController Class Reference	28
5.20.1 Detailed Description	29
5.20.2 Member Function Documentation	29
5.20.2.1 Index	29
5.21 webchat.Communication.Publisher Class Reference	29
5.21.1 Detailed Description	29
5.21.2 Member Function Documentation	30
5.21.2.1 Publish	30
5.21.3 Property Documentation	30
5.21.3.1 Clients	30
5.22 webchat.Controllers.RoomController Class Reference	30
5.22.1 Detailed Description	30
5.22.2 Member Function Documentation	30
5.22.2.1 Join	30
5.22.2.2 Leave	31
5.23 webchat.Models.RoomsModel Class Reference	31
5.23.1 Detailed Description	31
5.23.2 Property Documentation	31

5.23.2.1	Rooms	31
5.24	webchat.Models.Binders.RoomsModelBinder Class Reference	32
5.24.1	Detailed Description	32
5.24.2	Member Function Documentation	32
5.24.2.1	BindProperty	32
5.25	webchat.Validators.RoomsValidationAttribute Class Reference	32
5.25.1	Detailed Description	32
5.25.2	Constructor & Destructor Documentation	33
5.25.2.1	RoomsValidationAttribute	33
5.25.3	Member Function Documentation	33
5.25.3.1	IsValid	33
5.26	webchat.Validators.UniqueNickValidationAttribute Class Reference	33
5.26.1	Detailed Description	33
5.26.2	Constructor & Destructor Documentation	33
5.26.2.1	UniqueNickValidationAttribute	33
5.26.3	Member Function Documentation	33
5.26.3.1	IsValid	33
	Index	34

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

webchat	7
webchat.Communication	7
webchat.Controllers	7
webchat.Database	8
webchat.Filters	8
webchat.Helpers	8
webchat.Logging	8
webchat.Models	9
webchat.Models.Binders	9
webchat.Ping	9
webchat.Validators	9

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

ApiController	
webchat.Controllers.EventStreamController	16
AuthorizeAttribute	
webchat.Filters.AuthenticationFilterAttribute	11
webchat.Helpers.BindingHelper	11
webchat.Models.ChatModel	13
Controller	
webchat.Controllers.ChatController	12
webchat.Controllers.IndexController	20
webchat.Controllers.MessageController	25
webchat.Controllers.PongController	28
webchat.Controllers.RoomController	30
DefaultModelBinder	
webchat.Models.Binders.MessageModelBinder	26
webchat.Models.Binders.RoomsModelBinder	32
HttpApplication	
webchat.MvcApplication	27
IAuthorizationFilter	
webchat.Filters.AuthenticationFilterAttribute	11
webchat.Database.IDatabase	17
webchat.Database.Db	14
webchat.Logging.ILogger	20
webchat.Logging.Logger	24
webchat.Communication.IPublisher< T >	22
webchat.Communication.Publisher	29
webchat.Models.LeaveModel	24
webchat.Models.MessageModel	26
webchat.Ping.Pinger	28
webchat.Models.RoomsModel	31
webchat.Models.IndexModel	21
ValidationAttribute	
webchat.Validators.JoinedRoomValidationAttribute	23
webchat.Validators.RoomsValidationAttribute	32
webchat.Validators.UniqueNickValidationAttribute	33

Chapter 3

Class Index

3.1 Class List

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

webchat.Filters.AuthenticationFilterAttribute	
Checks whether the user is logged in or not	11
webchat.Helpers.BindingHelper	
Binding helpers used by Models.Binders	11
webchat.Controllers.ChatController	
Handles the main page of the appliaction	12
webchat.Models.ChatModel	
Model for the Controllers.ChatController	13
webchat.Database.Db	
A concrete implementation of IDatabase	14
webchat.Controllers.EventStreamController	
Handle the EventStream connections	16
webchat.Database.IDatabase	
The interface that defines what a database class should do	17
webchat.Logging.ILogger	
An interface to define logging actions	20
webchat.Controllers.IndexController	
The controller that handles the authentication page	20
webchat.Models.IndexModel	
Model for the Controllers.IndexController	21
webchat.Communication.IPublisher< T >	
Defines behaviour of publisher classes	22
webchat.Validators.JoinedRoomValidationAttribute	
Checks whether the user has joined a certain room	23
webchat.Models.LeaveModel	
Model used by Controllers.RoomController when the user leaves rooms	24
webchat.Logging.Logger	
Concrete implementation of ILogger	24
webchat.Controllers.MessageController	
Handle the incoming messages	25
webchat.Models.MessageModel	
Model used by Controllers.MessageController	26
webchat.Models.Binders.MessageModelBinder	
Binder for MessageModel	26
webchat.MvcApplication	
27	
webchat.Ping.Pinger	
Class used for sending PING to the users of the application	28

webchat.Controllers.PongController	
Handles the PONG part of the PING PONG protocol	28
webchat.Communication.Publisher	
Concrete implementation of IPublisher<T>	29
webchat.Controllers.RoomController	
Handle the joning and leaving of rooms (channels)	30
webchat.Models.RoomsModel	
Model used by Controllers.IndexController and Controllers.RoomController for joining rooms .	31
webchat.Models.Binders.RoomsModelBinder	
Binder for RoomsModel	32
webchat.Validators.RoomsValidationAttribute	
Check the room names	32
webchat.Validators.UniqueNickValidationAttribute	
Check whether a nick is unique	33

Chapter 4

Namespace Documentation

4.1 Package webchat

Namespaces

- package [Communication](#)
- package [Controllers](#)
- package [Database](#)
- package [Filters](#)
- package [Helpers](#)
- package [Logging](#)
- package [Models](#)
- package [Ping](#)
- package [Validators](#)

Classes

- class [MvcApplication](#)

4.2 Package webchat.Communication

Classes

- interface [IPublisher< T >](#)
Defines behaviour of publisher classes
- class [Publisher](#)
Concrete implementation of [IPublisher<T>](#)

4.3 Package webchat.Controllers

Classes

- class [ChatController](#)
Handles the main page of the appliaction
- class [EventStreamController](#)

Handle the EventStream connections

- class [IndexController](#)

The controller that handles the authentication page

- class [MessageController](#)

Handle the incoming messages

- class [PongController](#)

Handles the PONG part of the PING PONG protocol

- class [RoomController](#)

Handle the joining and leaving of rooms (channels)

4.4 Package webchat.Database

Classes

- class [Db](#)

A concrete implementation of [IDatabase](#)

- interface [IDatabase](#)

The interface that defines what a database class should do

4.5 Package webchat.Filters

Classes

- class [AuthenticationFilterAttribute](#)

Checks whether the user is logged in or not

4.6 Package webchat.Helpers

Classes

- class [BindingHelper](#)

Binding helpers used by [Models.Binders](#)

- class **ErrorSummaryHelper**

Helper to display errors

4.7 Package webchat.Logging

Classes

- interface [ILogger](#)

An interface to define logging actions

- class [Logger](#)

Concrete implementation of [ILogger](#)

4.8 Package webchat.Models

Namespaces

- package [Binders](#)

Classes

- class [ChatModel](#)
Model for the [Controllers.ChatController](#)
- class [IndexModel](#)
Model for the [Controllers.IndexController](#)
- class [LeaveModel](#)
Model used by [Controllers.RoomController](#) when the user leaves rooms
- class [MessageModel](#)
Model used by [Controllers.MessageController](#)
- class [RoomsModel](#)
Model used by [Controllers.IndexController](#) and [Controllers.RoomController](#) for joining rooms

4.9 Package webchat.Models.Binders

Classes

- class [MessageModelBinder](#)
Binder for [MessageModel](#)
- class [RoomsModelBinder](#)
Binder for [RoomsModel](#)

4.10 Package webchat.Ping

Classes

- class [Pinger](#)
Class used for sending PING to the users of the application

4.11 Package webchat.Validators

Classes

- class [JoinedRoomValidationAttribute](#)
Checks whether the user has joined a certain room
- class [RoomsValidationAttribute](#)
Check the room names
- class [UniqueNickValidationAttribute](#)
Check whether a nick is unique

Chapter 5

Class Documentation

5.1 webchat.Filters.AuthenticationFilterAttribute Class Reference

Checks whether the user is logged in or not

Inherits `AuthorizeAttribute`, and `IAuthorizationFilter`.

Public Member Functions

- override void [OnAuthorization](#) (`AuthorizationContext authorizationContext`)
Does the actual authorization

5.1.1 Detailed Description

Checks whether the user is logged in or not

5.1.2 Member Function Documentation

5.1.2.1 override void webchat.Filters.AuthenticationFilterAttribute.OnAuthorization (`AuthorizationContext authorizationContext`)

Does the actual authorization

Parameters

<i>authorization-Context</i>	Object that holds HTTP and Session data
------------------------------	---

If the user is not logged in he's automatically redirected to the Index

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

- `D:/localhost/github/webchat/webchat/Filters/AuthenticationFilterAttribute.cs`

5.2 webchat.Helpers.BindingHelper Class Reference

Binding helpers used by [Models.Binders](#)

Static Public Member Functions

- static T [GetValue](#)< T > (ModelBindingContext bindingContext, string key)
Get a value from a HTTP form

5.2.1 Detailed Description

Binding helpers used by [Models.Binders](#)

5.2.2 Member Function Documentation

- 5.2.2.1 static T webchat.Helpers.BindingHelper.GetValue< T > (ModelBindingContext *bindingContext*, string *key*)
[static]

Get a value from a HTTP form

Template Parameters

<i>T</i>	Return's value type
----------	---------------------

Parameters

<i>bindingContext</i>	The object that holds all the HTTP data
<i>key</i>	The needed value should be retrieved by it's key (the element's name attribute)

Returns

Returns the value stored at the key

See: <http://odetocode.com/blogs/scott/archive/2009/05/05/iterating-on-an-asp-net-mvc-mode.aspx>

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

- D:/localhost/github/webchat/webchat/Helpers/BindingHelpers.cs

5.3 webchat.Controllers.ChatController Class Reference

Handles the main page of the appliaction

Inherits Controller.

Public Member Functions

- ActionResult [Index](#) ()
Load the initial page where the user can send messages and join additional rooms
- ActionResult [Disconnect](#) ()
Log the user out

5.3.1 Detailed Description

Handles the main page of the appliaction

In order to enter the chat the user must be authenticated

See Also

[Filters.AuthenticationFilterAttribute](#)

5.3.2 Member Function Documentation

5.3.2.1 ActionResult webchat.Controllers.ChatController.Disconnect ()

Log the user out

Returns

Redirects the user to the [IndexController](#)

5.3.2.2 ActionResult webchat.Controllers.ChatController.Index ()

Load the initial page where the user can send messages and join additional rooms

Returns

Returns a [Models.ChatModel](#)

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

- D:/localhost/github/webchat/webchat/Controllers/ChatController.cs

5.4 webchat.Models.ChatModel Class Reference

Model for the [Controllers.ChatController](#)

Properties

- Dictionary< string, HashSet< string > > [Users](#) [get, set]
A copy of the database used to display the connected users on every room
- List< string > [Rooms](#) [get, set]
The rooms the user connected to

5.4.1 Detailed Description

Model for the [Controllers.ChatController](#)

5.4.2 Property Documentation

5.4.2.1 List<string> webchat.Models.ChatModel.Rooms [get], [set]

The rooms the user connected to

5.4.2.2 Dictionary<string, HashSet<string> > webchat.Models.ChatModel.Users [get], [set]

A copy of the database used to display the connected users on every room

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

- D:/localhost/github/webchat/webchat/Models/ChatModel.cs

5.5 webchat.Database.Db Class Reference

A concrete implementation of [IDatabase](#)

Inherits [webchat.Database.IDatabase](#).

Public Member Functions

- [Db](#) (IPublisher< ConcurrentQueue< StreamWriter >> p, [ILogger](#) I)
The Database's constructor
- void [AddUser](#) (IEnumerable< string > rooms, string nick)
Add a user to the database
- void [DelUser](#) (IEnumerable< string > rooms, string nick)
Delete a user from the database in case he leaves
- void [DelUserFromGlobalList](#) (string nick)
Delete a user form the global list
- Dictionary< string, HashSet< string > > [GetUsers](#) ()
Get the users currently connected
- List< string > [GetRooms](#) ()
Get all the rooms currently populated
- List< string > [GetRooms](#) (string nick)
Get all the rooms currently joined by a user
- List< string > [GetBackupRooms](#) (string nick)
Get the rooms the user was connected to before the PING
- void [Backup](#) ()
Do a backup of the database before sending a PING
- bool [IsPopulated](#) ()
Check if there are users connected
- bool [IsUser](#) (string nick)
Check if a user is connected on at least a room

5.5.1 Detailed Description

A concrete implementation of [IDatabase](#)

5.5.2 Constructor & Destructor Documentation

5.5.2.1 webchat.Database.Db.Db (IPublisher< ConcurrentQueue< StreamWriter >> p, [ILogger](#) I)

The [Database](#)'s constructor

Parameters

p	A concrete implementation of an Communication.IPublisher<T>
I	A concrete implementation of an Logging.ILogger

5.5.3 Member Function Documentation

5.5.3.1 void webchat.Database.Db.AddUser (IEnumerable< string > *rooms*, string *nick*)

Add a user to the database

Parameters

<i>rooms</i>	Which rooms the user joined
<i>nick</i>	The user's nickname

5.5.3.2 void webchat.Database.Db.Backup ()

Do a backup of the database before sending a PING

5.5.3.3 void webchat.Database.Db.DelUser (IEnumerable< string > *rooms*, string *nick*)

Delete a user from the database in case he leaves

Parameters

<i>rooms</i>	The rooms the user left
<i>nick</i>	The user's nickname

5.5.3.4 void webchat.Database.Db.DelUserFromGlobalList (string *nick*)

Delete a user from the global list

Parameters

<i>nick</i>	The user's nickname
-------------	---------------------

5.5.3.5 List<string> webchat.Database.Db.GetBackupRooms (string *nick*)

Get the rooms the user was connected to before the PING

Parameters

<i>nick</i>	The user's nickname
-------------	---------------------

Returns

Returns a List<string> of rooms

5.5.3.6 List<string> webchat.Database.Db.GetRooms ()

Get all the rooms currently populated

Returns

Returns a List<string> of rooms

5.5.3.7 List<string> webchat.Database.Db.GetRooms (string *nick*)

Get all the rooms currently joined by a user

Parameters

<i>nick</i>	The user's nickname
-------------	---------------------

Returns

Returns a List<string> of rooms

5.5.3.8 Dictionary<string, HashSet<string>> webchat.Database.Db.GetUsers ()

Get the users currently connected

Returns

Returns a Dictionary<string, HashSet<string>> of rooms as keys and users as values for the HashSet<string>

5.5.3.9 bool webchat.Database.Db.IsPopulated ()

Check if there are users connected

Returns

Returns true if there is at least one user, else returns false

5.5.3.10 bool webchat.Database.Db.IsUser (string *nick*)

Check if a user is connected on at least a room

Parameters

<i>nick</i>	The user's nickname
-------------	---------------------

Returns

Return true if the user is connected, else false

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

- D:/localhost/github/webchat/webchat/Database/Database.cs

5.6 webchat.Controllers.EventStreamController Class Reference

Handle the EventStream connections

Inherits ApiController.

Public Member Functions

- HttpResponseMessage [Get](#) (HttpRequestMessage request)

Prepare the response headers for the EventStream

5.6.1 Detailed Description

Handle the EventStream connections

The user cannot access this if he's not authenticated

See Also

[Filters.AuthenticationFilterAttribute](#)

5.6.2 Member Function Documentation

5.6.2.1 HttpResponseMessage webchat.Controllers.EventStreamController.Get (HttpRequestMessage request)

Prepare the response headers for the EventStream

Parameters

<i>request</i>	The request from which the response will be generated
----------------	---

Returns

Returns HttpResponseMessage whose content will be changed by [MvcApplication.Pub](#)

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

- D:/localhost/github/webchat/webchat/Controllers/EventStreamController.cs

5.7 webchat.Database.IDatabase Interface Reference

The interface that defines what a database class should do

Inherited by [webchat.Database.Db](#).

Public Member Functions

- void [AddUser](#) (IEnumerable< string > rooms, string nick)
Add a user to the daatbase
- void [DelUser](#) (IEnumerable< string > rooms, string nick)
Delete a user from the database in case he leaves
- void [DelUserFromGlobalList](#) (string nick)
Delete a user from the global user list
- Dictionary< string, HashSet< string > > [GetUsers](#) ()
Get the users currently connected
- List< string > [GetRooms](#) ()
Get all the rooms currently populated
- List< string > [GetRooms](#) (string nick)
Get all the rooms currently joined by a user
- List< string > [GetBackupRooms](#) (string nick)
Get the rooms the user was connected to before the PING

- void [Backup](#) ()
Do a backup of the database before sending a PING
- bool [IsPopulated](#) ()
Check if there are users connected
- bool [IsUser](#) (string nick)
Check if a user is connected on at least a room

5.7.1 Detailed Description

The interface that defines what a database class should do

5.7.2 Member Function Documentation

5.7.2.1 void webchat.Database.IDatabase.AddUser (IEnumerable< string > rooms, string nick)

Add a user to the daatbase

Parameters

<i>rooms</i>	Which rooms the user joined
<i>nick</i>	The user's nickname

5.7.2.2 void webchat.Database.IDatabase.Backup ()

Do a backup of the database before sending a PING

5.7.2.3 void webchat.Database.IDatabase.DelUser (IEnumerable< string > rooms, string nick)

Delete a user from the database in case he leaves

Parameters

<i>rooms</i>	The rooms the user left
<i>nick</i>	The user's nickname

5.7.2.4 void webchat.Database.IDatabase.DelUserFromGlobalList (string nick)

Delete a user from the global user list

Parameters

<i>nick</i>	The user's nickname
-------------	---------------------

5.7.2.5 List<string> webchat.Database.IDatabase.GetBackupRooms (string nick)

Get the rooms the user was connected to before the PING

Parameters

<i>nick</i>	The user's nickname
-------------	---------------------

Returns

Returns a List<string> of rooms

5.7.2.6 List<string> webchat.Database.IDatabase.GetRooms ()

Get all the rooms currently populated

Returns

Returns a List<string> of rooms

5.7.2.7 List<string> webchat.Database.IDatabase.GetRooms (string *nick*)

Get all the rooms currently joined by a user

Parameters

<i>nick</i>	The user's nickname
-------------	---------------------

Returns

Returns a List<string> of rooms

5.7.2.8 Dictionary<string, HashSet<string>> webchat.Database.IDatabase.GetUsers ()

Get the users currently connected

Returns

Returns a Dictionary<string, HashSet<string>> of rooms as keys and users as values for the HashSet<string>

5.7.2.9 bool webchat.Database.IDatabase.IsPopulated ()

Check if there are users connected

Returns

Returns true if there is at least one user, else returns false

5.7.2.10 bool webchat.Database.IDatabase.IsUser (string *nick*)

Check if a user is connected on at least a room

Parameters

<i>nick</i>	The user's nickname
-------------	---------------------

Returns

Return true if the user is connected, else false

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

- D:/localhost/github/webchat/webchat/Database/IDatabase.cs

5.8 webchat.Logging.ILogger Interface Reference

An interface to define logging actions

Inherited by [webchat.Logging.Logger](#).

Public Member Functions

- void [Log](#) (string message)
Overloaded method to log a message
- void [Log](#) (string message, string category)
Overloaded method to log a message

5.8.1 Detailed Description

An interface to define logging actions

5.8.2 Member Function Documentation

5.8.2.1 void webchat.Logging.ILogger.Log (string message)

Overloaded method to log a message

Parameters

<i>message</i>	The string to be logged
----------------	-------------------------

5.8.2.2 void webchat.Logging.ILogger.Log (string message, string category)

Overloaded method to log a message

Parameters

<i>message</i>	The string to be logged
<i>category</i>	The category in which the logged message belongs

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

- D:/localhost/github/webchat/webchat/Logging/ILogger.cs

5.9 webchat.Controllers.IndexController Class Reference

The controller that handles the authentication page

Inherits Controller.

Public Member Functions

- ActionResult [Index](#) ()
Display the authentication page or the chat page, depending if the user has authenticated or not
- ActionResult [Index](#) ([IndexModel](#) indexModel, bool captchaValid, string captchaErrorMessage)
Process the authentication of the user

5.9.1 Detailed Description

The controller that handles the authentication page

See Also

[Filters.AuthenticationFilterAttribute](#)

5.9.2 Member Function Documentation

5.9.2.1 ActionResult webchat.Controllers.IndexController.Index ()

Display the authentication page or the chat page, depending if the user has authenticated or not

Returns

Redirects the user to the chat if he's already authenticated or displays the login page

5.9.2.2 ActionResult webchat.Controllers.IndexController.Index ([IndexModel](#) indexModel, bool captchaValid, string captchaErrorMessage)

Process the authentication of the user

Parameters

<i>indexModel</i>	The model that has the user's data binded to
<i>captchaValid</i>	Whether the captcha is valid or not
<i>captchaError-Message</i>	If the user failed to provide a valid captcha this holds the reason

Returns

Returns an [Models.IndexModel](#) populated with data if the user submitted invalid data or redirects the user to the chat if he logged in

This also handles the CSRF token which prevents an attacker to submit the form remotely. If the user doesn't provide some rooms to join, he's automatically connected to a default one

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

- D:/localhost/github/webchat/webchat/Controllers/IndexController.cs

5.10 webchat.Models.IndexModel Class Reference

Model for the [Controllers.IndexController](#)

Inherits [webchat.Models.RoomsModel](#).

Properties

- string [Nick](#) [get, set]
The user's nickname

5.10.1 Detailed Description

Model for the [Controllers.IndexController](#)

5.10.2 Property Documentation

5.10.2.1 string [webchat.Models.IndexModel.Nick](#) [get], [set]

The user's nickname

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

- D:/localhost/github/webchat/webchat/Models/IndexModel.cs

5.11 [webchat.Communication.IPublisher< T >](#) Interface Template Reference

Defines behaviour of publisher classes

Inherited by [webchat.Communication.Publisher](#).

Public Member Functions

- void [Publish](#) (string channel, string message)
Define how the message is actually sent to the clients

Properties

- T [Clients](#) [get]
A collection that holds the clients to whom messages will be published

5.11.1 Detailed Description

Defines behaviour of publisher classes

Template Parameters

<i>T</i>	The type of the Clients property that the class implementing this interface will hold
----------	---

5.11.2 Member Function Documentation

5.11.2.1 void [webchat.Communication.IPublisher< T >.Publish](#) (string channel, string message)

Define how the message is actually sent to the clients

Parameters

<i>channel</i>	The channel to publish messages on, this may be used for categorizing messages
<i>message</i>	The message to be sent to every client

5.11.3 Property Documentation

5.11.3.1 T webchat.Communication.IPublisher< T >.Clients [get]

A collection that holds the clients to whom messages will be published

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

- D:/localhost/github/webchat/webchat/Communication/IPublisher.cs

5.12 webchat.Validators.JoinedRoomValidationAttribute Class Reference

Checks whether the user has joined a certain room

Inherits ValidationAttribute.

Public Member Functions

- [JoinedRoomValidationAttribute](#) ()
The constructor
- override bool [IsValid](#) (object value)
Do the actual checking of the room

5.12.1 Detailed Description

Checks whether the user has joined a certain room

5.12.2 Constructor & Destructor Documentation

5.12.2.1 webchat.Validators.JoinedRoomValidationAttribute.JoinedRoomValidationAttribute ()

The constructor

5.12.3 Member Function Documentation

5.12.3.1 override bool webchat.Validators.JoinedRoomValidationAttribute.IsValid (object value)

Do the actual checking of the room

Parameters

<i>value</i>	The room's name as a string
--------------	-----------------------------

Returns

Returns true if the user joined the room, otherwise false

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

- `D:/localhost/github/webchat/webchat/Validators/JoinedRoomValidationAttribute.cs`

5.13 webchat.Models.LeaveModel Class Reference

Model used by [Controllers.RoomController](#) when the user leaves rooms

Properties

- string [Room](#) [get, set]
The room the user left

5.13.1 Detailed Description

Model used by [Controllers.RoomController](#) when the user leaves rooms

5.13.2 Property Documentation

5.13.2.1 string webchat.Models.LeaveModel.Room [get], [set]

The room the user left

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

- `D:/localhost/github/webchat/webchat/Models/LeaveModel.cs`

5.14 webchat.Logging.Logger Class Reference

Concrete implementation of [ILogger](#)

Inherits [webchat.Logging.ILogger](#).

Public Member Functions

- void [Log](#) (string message)
Overloaded method to log a message to Trace
- void [Log](#) (string message, string category)
Overloaded method to log a message to Trace

5.14.1 Detailed Description

Concrete implementation of [ILogger](#)

5.14.2 Member Function Documentation

5.14.2.1 void webchat.Logging.Logger.Log (string message)

Overloaded method to log a message to Trace

Parameters

<i>message</i>	The string to be logged
----------------	-------------------------

5.14.2.2 void webchat.Logging.Logger.Log (string *message*, string *category*)

Overloaded method to log a message to Trace

Parameters

<i>message</i>	The string to be logged
<i>category</i>	The category in which the logged message belongs

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

- D:/localhost/github/webchat/webchat/Logging/Logger.cs

5.15 webchat.Controllers.MessageController Class Reference

Handle the incoming messages

Inherits Controller.

Public Member Functions

- HttpStatusCode [Post](#) ([MessageModel](#) m)
Publish the message to every chat user on the corresponding channel

5.15.1 Detailed Description

Handle the incoming messages

In order to send messages the user must be authenticated

See Also

[Filters.AuthenticationFilterAttribute](#)

5.15.2 Member Function Documentation

5.15.2.1 HttpStatusCode webchat.Controllers.MessageController.Post ([MessageModel](#) m)

Publish the message to every chat user on the corresponding channel

Parameters

<i>m</i>	The model which contains the intended message and channel See Also Models.MessageModel
----------	--

Returns

Returns a HttpStatusCode which represents whether the operation was successful or not

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

- D:/localhost/github/webchat/webchat/Controllers/MessageController.cs

5.16 webchat.Models.MessageModel Class Reference

Model used by [Controllers.MessageController](#)

Properties

- string [Message](#) [get, set]
Users message
- string [Room](#) [get, set]
The room on which the message was sent

5.16.1 Detailed Description

Model used by [Controllers.MessageController](#)

5.16.2 Property Documentation

5.16.2.1 string webchat.Models.MessageModel.Message [get], [set]

Users message

5.16.2.2 string webchat.Models.MessageModel.Room [get], [set]

The room on which the message was sent

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

- D:/localhost/github/webchat/webchat/Models/MessageModel.cs

5.17 webchat.Models.Binders.MessageModelBinder Class Reference

Binder for [MessageModel](#)

Inherits DefaultModelBinder.

Protected Member Functions

- override void [BindProperty](#) (ControllerContext controllerContext, ModelBindingContext bindingContext, PropertyDescriptor propertyDescriptor)
Overridden method to customly bind a string by trimming and escaping it

5.17.1 Detailed Description

Binder for [MessageModel](#)

5.17.2 Member Function Documentation

5.17.2.1 `override void webchat.Models.Binders.MessageModelBinder.BindProperty (ControllerContext controllerContext, ModelBindingContext bindingContext, PropertyDescriptor propertyDescriptor)` `[protected]`

Overridden method to customly bind a string by trimming and escaping it

Parameters

<i>controllerContext</i>	
<i>bindingContext</i>	
<i>property-Descriptor</i>	

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

- `D:/localhost/github/webchat/webchat/Models/Binders/MessageModelBinder.cs`

5.18 webchat.MvcApplication Class Reference

Inherits `HttpApplication`.

Static Public Attributes

- static `IPublisher`
`< ConcurrentQueue`
`< StreamWriter > > Pub = new Publisher()`
A singleton of a [Communication.IPublisher<T>](#) to use throughout the application
- static `ILogger Logger = new Logger()`
A singleton of an [Logging.ILogger](#) to use throughout the application
- static `IDatabase Db = new Db(Pub, Logger)`
A singleton of an [Database.IDatabase](#) to use throughout the application

Protected Member Functions

- void `Application_Start ()`

5.18.1 Detailed Description

5.18.2 Member Function Documentation

5.18.2.1 `void webchat.MvcApplication.Application_Start ()` `[protected]`

5.18.3 Member Data Documentation

5.18.3.1 `IDatabase webchat.MvcApplication.Db = new Db(Pub, Logger)` `[static]`

A singleton of an [Database.IDatabase](#) to use throughout the application

5.18.3.2 ILogger webchat.MvcApplication.Logger = new Logger() [static]

A singleton of an [Logging.ILogger](#) to use throughout the application

5.18.3.3 IPublisher<ConcurrentQueue<StreamWriter> > webchat.MvcApplication.Pub = new Publisher() [static]

A singleton of a [Communication.IPublisher<T>](#) to use throughout the application

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

- D:/localhost/github/webchat/webchat/Global.asax.cs

5.19 webchat.Ping.Pinger Class Reference

Class used for sending PING to the users of the application

Public Member Functions

- [Pinger](#) (int i=60000)
The constructor
- void [Ping](#) ()
Send the PING by using the [MvcApplication.Pub](#) and log the action using [MvcApplication.Logger](#)

5.19.1 Detailed Description

Class used for sending PING to the users of the application

5.19.2 Constructor & Destructor Documentation

5.19.2.1 webchat.Ping.Pinger.Pinger (int i = 60000)

The constructor

Parameters

<i>i</i>	The time interval between two PINGs
----------	-------------------------------------

5.19.3 Member Function Documentation

5.19.3.1 void webchat.Ping.Pinger.Ping ()

Send the PING by using the [MvcApplication.Pub](#) and log the action using [MvcApplication.Logger](#)

The backup is done by [MvcApplication.Db](#)

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

- D:/localhost/github/webchat/webchat/Ping/Pinger.cs

5.20 webchat.Controllers.PongController Class Reference

Handles the PONG part of the PING PONG protocol

Inherits Controller.

Public Member Functions

- [HttpStatusCode Index](#) ()

Restore the user's state in the application if he has an active connection to it

5.20.1 Detailed Description

Handles the PONG part of the PING PONG protocol

In order to be able to respond to pings the user must be authenticated

See Also

[Filters.AuthenticationFilterAttribute](#)

5.20.2 Member Function Documentation

5.20.2.1 [HttpStatusCode webchat.Controllers.PongController.Index](#) ()

Restore the user's state in the application if he has an active connection to it

Returns

Returns a [HttpStatusCode](#) which represents whether the operation was successful or not

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

- D:/localhost/github/webchat/webchat/Controllers/PongController.cs

5.21 webchat.Communication.Publisher Class Reference

Concrete implementation of [IPublisher<T>](#)

Inherits [webchat.Communication.IPublisher< T >](#).

Public Member Functions

- void [Publish](#) (string channel, string message)

Publish a certain message on a certain channel to all clients by using the stream set up in [Controllers.EventStreamController](#)

Properties

- [ConcurrentQueue< StreamWriter > Clients](#) [get]

Get a list of all connected clients

5.21.1 Detailed Description

Concrete implementation of [IPublisher<T>](#)

5.21.2 Member Function Documentation

5.21.2.1 void webchat.Communication.Publisher.Publish (string *channel*, string *message*)

Publish a certain message on a certain channel to all clients by using the stream set up in [Controllers.EventStream-Controller](#)

Parameters

<i>channel</i>	The channel to publish messages on, this may be used for categorizing messages
<i>message</i>	The message to be sent to every client

5.21.3 Property Documentation

5.21.3.1 ConcurrentQueue<StreamWriter> webchat.Communication.Publisher.Clients [get]

Get a list of all connected clients

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

- D:/localhost/github/webchat/webchat/Communication/Publisher.cs

5.22 webchat.Controllers.RoomController Class Reference

Handle the joning and leaving of rooms (channels)

Inherits Controller.

Public Member Functions

- string [Join](#) ([RoomsModel](#) roomsModel)
Join a list of rooms
- string [Leave](#) ([LeaveModel](#) leaveModel)
Leave a room

5.22.1 Detailed Description

Handle the joning and leaving of rooms (channels)

In order to be able to join or leave rooms the user must be authenticated

See Also

[Filters.AuthenticationFilterAttribute](#)

5.22.2 Member Function Documentation

5.22.2.1 string webchat.Controllers.RoomController.Join (RoomsModel *roomsModel*)

Join a list of rooms

Parameters

<i>roomsModel</i>	The list of rooms to be joined
-------------------	--------------------------------

Returns

Returns the user's currently joined rooms as JSON

This may also return a localized error string if the room names are invalid

See Also

[Validators.RoomsValidationAttribute](#)

5.22.2.2 string webchat.Controllers.RoomController.Leave (LeaveModel leaveModel)

Leave a room

Parameters

<i>leaveModel</i>	The room to leave
-------------------	-------------------

Returns

If the user is still connected on other rooms returns them as JSON or returns "" if the user left the only room he was connected to

This may also return a localized error string if the user tries to leave a room he's not connected to

See Also

[Validators.JoinedRoomValidationAttribute](#)

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

- D:/localhost/github/webchat/webchat/Controllers/RoomController.cs

5.23 webchat.Models.RoomsModel Class Reference

Model used by [Controllers.IndexController](#) and [Controllers.RoomController](#) for joining rooms

Inherited by [webchat.Models.IndexModel](#).

Properties

- List< string > [Rooms](#) [get, set]
The rooms joined

5.23.1 Detailed Description

Model used by [Controllers.IndexController](#) and [Controllers.RoomController](#) for joining rooms

5.23.2 Property Documentation**5.23.2.1 List<string> webchat.Models.RoomsModel.Rooms [get], [set]**

The rooms joined

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

- D:/localhost/github/webchat/webchat/Models/RoomsModel.cs

5.24 webchat.Models.Binders.RoomsModelBinder Class Reference

Binder for [RoomsModel](#)

Inherits [DefaultModelBinder](#).

Protected Member Functions

- override void [BindProperty](#) (ControllerContext controllerContext, ModelBindingContext bindingContext, System.ComponentModel.PropertyDescriptor propertyDescriptor)
Overridden method to customly bind a string to a List<string> after splitting and sanitizing it

5.24.1 Detailed Description

Binder for [RoomsModel](#)

5.24.2 Member Function Documentation

- 5.24.2.1 override void webchat.Models.Binders.RoomsModelBinder.BindProperty (ControllerContext *controllerContext*, ModelBindingContext *bindingContext*, System.ComponentModel.PropertyDescriptor *propertyDescriptor*)
[protected]

Overridden method to customly bind a string to a List<string> after splitting and sanitizing it

Parameters

<i>controllerContext</i>	
<i>bindingContext</i>	
<i>property-Descriptor</i>	

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

- D:/localhost/github/webchat/webchat/Models/Binders/RoomsModelBinder.cs

5.25 webchat.Validators.RoomsValidationAttribute Class Reference

Check the room names

Inherits [ValidationAttribute](#).

Public Member Functions

- [RoomsValidationAttribute](#) ()
The constructor
- override bool [IsValid](#) (object value)
Do the actual checking against the rooms

5.25.1 Detailed Description

Check the room names

5.25.2 Constructor & Destructor Documentation

5.25.2.1 webchat.Validators.RoomsValidationAttribute.RoomsValidationAttribute ()

The constructor

5.25.3 Member Function Documentation

5.25.3.1 override bool webchat.Validators.RoomsValidationAttribute.IsValid (object *value*)

Do the actual checking against the rooms

Parameters

<i>value</i>	A List<string> of rooms to be checked
--------------	---------------------------------------

Returns

Returns true if all rooms are valid, else false

A room name is considered valid if it matches the regex: `^\[\w] + $`

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

- `D:/localhost/github/webchat/webchat/Validators/RoomsValidationAttribute.cs`

5.26 webchat.Validators.UniqueNickValidationAttribute Class Reference

Check whether a nick is unique

Inherits ValidationAttribute.

Public Member Functions

- [UniqueNickValidationAttribute](#) ()
The constructor
- override bool [IsValid](#) (object *value*)
Do the actual checking of the uniqueness of the nick

5.26.1 Detailed Description

Check whether a nick is unique

5.26.2 Constructor & Destructor Documentation

5.26.2.1 webchat.Validators.UniqueNickValidationAttribute.UniqueNickValidationAttribute ()

The constructor

5.26.3 Member Function Documentation

5.26.3.1 override bool webchat.Validators.UniqueNickValidationAttribute.IsValid (object *value*)

Do the actual checking of the uniqueness of the nick

Parameters

<i>value</i>	The user's nick as string
--------------	---------------------------

Returns

Return true if the nick is unique

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

- D:/localhost/github/webchat/webchat/Validators/UniqueNickValidationAttribute.cs

Index

- AddUser
 - webchat::Database::Db, [15](#)
 - webchat::Database::IDatabase, [18](#)
- Application_Start
 - webchat::MvcApplication, [27](#)
- Backup
 - webchat::Database::Db, [15](#)
 - webchat::Database::IDatabase, [18](#)
- BindProperty
 - webchat::Models::Binders::MessageModelBinder, [27](#)
 - webchat::Models::Binders::RoomsModelBinder, [32](#)
- Clients
 - webchat::Communication::IPublisher< T >, [23](#)
 - webchat::Communication::Publisher, [30](#)
- Db
 - webchat::Database::Db, [14](#)
 - webchat::MvcApplication, [27](#)
- DelUser
 - webchat::Database::Db, [15](#)
 - webchat::Database::IDatabase, [18](#)
- DelUserFromGlobalList
 - webchat::Database::Db, [15](#)
 - webchat::Database::IDatabase, [18](#)
- Disconnect
 - webchat::Controllers::ChatController, [13](#)
- Get
 - webchat::Controllers::EventStreamController, [17](#)
- GetBackupRooms
 - webchat::Database::Db, [15](#)
 - webchat::Database::IDatabase, [18](#)
- GetRooms
 - webchat::Database::Db, [15](#)
 - webchat::Database::IDatabase, [19](#)
- GetUsers
 - webchat::Database::Db, [16](#)
 - webchat::Database::IDatabase, [19](#)
- GetValue< T >
 - webchat::Helpers::BindingHelper, [12](#)
- Index
 - webchat::Controllers::ChatController, [13](#)
 - webchat::Controllers::IndexController, [21](#)
 - webchat::Controllers::PongController, [29](#)
- IsPopulated
 - webchat::Database::Db, [16](#)
 - webchat::Database::IDatabase, [19](#)
- IsUser
 - webchat::Database::Db, [16](#)
 - webchat::Database::IDatabase, [19](#)
- IsValid
 - webchat::Validators::JoinedRoomValidationAttribute, [23](#)
 - webchat::Validators::RoomsValidationAttribute, [33](#)
 - webchat::Validators::UniqueNickValidationAttribute, [33](#)
- Join
 - webchat::Controllers::RoomController, [30](#)
- JoinedRoomValidationAttribute
 - webchat::Validators::JoinedRoomValidationAttribute, [23](#)
- Leave
 - webchat::Controllers::RoomController, [31](#)
- Log
 - webchat::Logging::ILogger, [20](#)
 - webchat::Logging::Logger, [24](#), [25](#)
- Logger
 - webchat::MvcApplication, [27](#)
- Message
 - webchat::Models::MessageModel, [26](#)
- Nick
 - webchat::Models::IndexModel, [22](#)
- OnAuthorization
 - webchat::Filters::AuthenticationFilterAttribute, [11](#)
- Ping
 - webchat::Ping::Pinger, [28](#)
- Pinger
 - webchat::Ping::Pinger, [28](#)
- Post
 - webchat::Controllers::MessageController, [25](#)
- Pub
 - webchat::MvcApplication, [28](#)
- Publish
 - webchat::Communication::IPublisher< T >, [22](#)
 - webchat::Communication::Publisher, [30](#)
- Room
 - webchat::Models::LeaveModel, [24](#)
 - webchat::Models::MessageModel, [26](#)
- Rooms
 - webchat::Models::ChatModel, [13](#)
 - webchat::Models::RoomsModel, [31](#)

- RoomsValidationAttribute
 - webchat::Validators::RoomsValidationAttribute, 33
- UniqueNickValidationAttribute
 - webchat::Validators::UniqueNickValidationAttribute, 33
- Users
 - webchat::Models::ChatModel, 13
- webchat, 7
- webchat.Communication, 7
- webchat.Communication.IPublisher< T >, 22
- webchat.Communication.Publisher, 29
- webchat.Controllers, 7
- webchat.Controllers.ChatController, 12
- webchat.Controllers.EventStreamController, 16
- webchat.Controllers.IndexController, 20
- webchat.Controllers.MessageController, 25
- webchat.Controllers.PongController, 28
- webchat.Controllers.RoomController, 30
- webchat.Database, 8
- webchat.Database.Db, 14
- webchat.Database.IDatabase, 17
- webchat.Filters, 8
- webchat.Filters.AuthenticationFilterAttribute, 11
- webchat.Helpers, 8
- webchat.Helpers.BindingHelper, 11
- webchat.Logging, 8
- webchat.Logging.ILogger, 20
- webchat.Logging.Logger, 24
- webchat.Models, 9
- webchat.Models.Binders, 9
- webchat.Models.Binders.MessageModelBinder, 26
- webchat.Models.Binders.RoomsModelBinder, 32
- webchat.Models.ChatModel, 13
- webchat.Models.IndexModel, 21
- webchat.Models.LeaveModel, 24
- webchat.Models.MessageModel, 26
- webchat.Models.RoomsModel, 31
- webchat.MvcApplication, 27
- webchat.Ping, 9
- webchat.Ping.Pinger, 28
- webchat.Validators, 9
- webchat.Validators.JoinedRoomValidationAttribute, 23
- webchat.Validators.RoomsValidationAttribute, 32
- webchat.Validators.UniqueNickValidationAttribute, 33
- webchat::Communication::IPublisher< T >
 - Clients, 23
 - Publish, 22
- webchat::Communication::Publisher
 - Clients, 30
 - Publish, 30
- webchat::Controllers::ChatController
 - Disconnect, 13
 - Index, 13
- webchat::Controllers::EventStreamController
 - Get, 17
- webchat::Controllers::IndexController
 - Index, 21
- webchat::Controllers::MessageController
 - Post, 25
- webchat::Controllers::PongController
 - Index, 29
- webchat::Controllers::RoomController
 - Join, 30
 - Leave, 31
- webchat::Database::Db
 - AddUser, 15
 - Backup, 15
 - Db, 14
 - DelUser, 15
 - DelUserFromGlobalList, 15
 - GetBackupRooms, 15
 - GetRooms, 15
 - GetUsers, 16
 - IsPopulated, 16
 - IsUser, 16
- webchat::Database::IDatabase
 - AddUser, 18
 - Backup, 18
 - DelUser, 18
 - DelUserFromGlobalList, 18
 - GetBackupRooms, 18
 - GetRooms, 19
 - GetUsers, 19
 - IsPopulated, 19
 - IsUser, 19
- webchat::Filters::AuthenticationFilterAttribute
 - OnAuthorization, 11
- webchat::Helpers::BindingHelper
 - GetValue< T >, 12
- webchat::Logging::ILogger
 - Log, 20
- webchat::Logging::Logger
 - Log, 24, 25
- webchat::Models::Binders::MessageModelBinder
 - BindProperty, 27
- webchat::Models::Binders::RoomsModelBinder
 - BindProperty, 32
- webchat::Models::ChatModel
 - Rooms, 13
 - Users, 13
- webchat::Models::IndexModel
 - Nick, 22
- webchat::Models::LeaveModel
 - Room, 24
- webchat::Models::MessageModel
 - Message, 26
 - Room, 26
- webchat::Models::RoomsModel
 - Rooms, 31
- webchat::MvcApplication
 - Application_Start, 27
 - Db, 27
 - Logger, 27
 - Pub, 28
- webchat::Ping::Pinger

- Ping, [28](#)
- Pinger, [28](#)
- webchat::Validators::JoinedRoomValidationAttribute
 - IsValid, [23](#)
 - JoinedRoomValidationAttribute, [23](#)
- webchat::Validators::RoomsValidationAttribute
 - IsValid, [33](#)
 - RoomsValidationAttribute, [33](#)
- webchat::Validators::UniqueNickValidationAttribute
 - IsValid, [33](#)
 - UniqueNickValidationAttribute, [33](#)