

CAPPASITY iOS FRAMEWORK

v0.3

Contents

CAPPASITY iOS FRAMEWORK	1
License	1
Overview	2
The CappasityDelegate Protocol	3
@protocol CappasityDelegate	3
User Account Login	3
@protocol CappasityServiceDelegate	3
@interface CappasityService	3
Obtaining a List of User Models	4
@protocol CappasityModelsListDelegate	4
@interface CappasityModelsList	4
Working with the Model	5
@interface CappasityEmbedCodeParams	5
@protocol CappasityModelDelegate	6
@interface CappasityModel	6
Displaying the Model	9
@interface CappasityModelViewParams	9
@protocol CappasityModelViewDelegate	10
@interface CappasityModelView	10
Error List	10
Obtaining the Token	11

License

Copyright (c) 2017 - 2018, Cappasity Inc. All rights reserved.

Redistribution and use in binary form, without modification, are permitted provided that the following conditions are met:

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Cappasity Inc., nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Overview

CappasityFramework contains functionality necessary for working with models on the Cappasity platform, i.e.:

- obtaining the models,
- obtaining the properties of a given model, editing the model, deleting the model,
- displaying the model.

Logging into your user account is done by providing the CappasityService class with the generated token, by sending it to the loginWith: method. The login is effective until the application is closed.

The entire set of user models may be obtained by placing consecutive requests for the subsequent n models, where n should be no greater than 100. Functionality for obtaining a list of models is contained in the CappasityModelsList class.

A specific model may be obtained via its id or sku. To do this, you need to make an instance of the CappasityModel class, using the corresponding initializers initWithId:delegate or initWithSku:delegate and calling the receiveInfo: method.

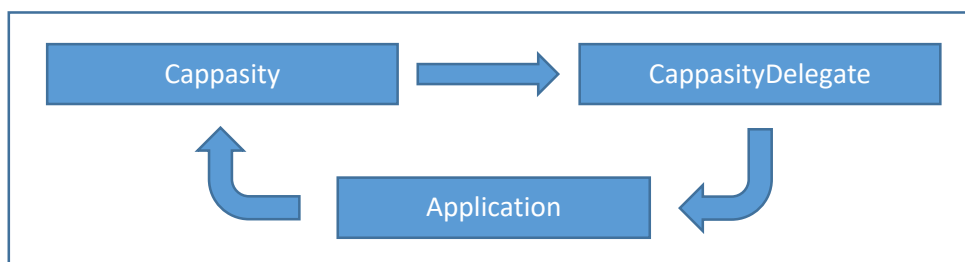
CappasityModel allows access to the following model information:

- sku,
- model title,
- model access type,
- model embed code.

The model's sku, title, and access type may be modified via the updateWithTitle:sku:access method or one of other overloaded methods. The remove: method deletes the model.

To display the model, you may use the embed code and incorporate it into your application yourself. Alternatively, you can use the CappasityModelView class. The CappasityModelView class inherits the UIView class, so it can be added into the application either through the storyboard or directly in the program.

Interaction with the classes is implemented through delegates.



Importing framework

```
#import "Cappasity/Cappasity-Swift.h"
```

The CappasityDelegate Protocol

@protocol CappasityDelegate

A protocol of shared methods.

All delegates for the main classes are inherited from the CappasityDelegate protocol.

Methods

- (void)cappasityErrorWithCode:(NSInteger)code description:(NSString * _Nonnull)description
Tells the delegate of an error during program execution.
Parameters:
 - code – error number
 - description – error description
- (void)cappasityMessage:(NSString * _Nonnull)message
Tells the delegate certain information regarding CappasityService state

User Account Login

@protocol CappasityServiceDelegate

A collection of methods that may be called by the CappasityService class object.

Overview

The loginCompleted: method is called in the case of a successful account login.
Inherits the CappasityDelegate protocol.

Methods

- (void)loginCompleted
Tells the delegate of the successful account login.

@interface CappasityService

Performs a token-based login to the Cappasity system.

Overview

The `loginWith:` method is used to log into the user account. In the case of a successful login, the `loginCompleted:` method of the `CappasityServiceDelegate` delegate is called, which instance is given during the initialization of the object.

Methods

- (nonnull instancetype)initWithDelegate:(id <CappasityServiceDelegate> _Nullable)delegate
Returns an instance of the `CappasityService` class.
Parameters:
 - delegate – the delegate object.
- (void)loginWith:(NSString * _Nonnull)token
Sends a request for a token-based login to the user account.
Parameters:
 - token – the user token.

Obtaining a List of User Models

@protocol CappasityModelsListDelegate

A collection of methods that may be called by the `CappasityModelsList` class object.

Overview

The method `receivedWithModels:` is called in the case of a successful request.
Inherits the `CappasityDelegate` protocol.

Methods

- (void)receivedWithModels:(NSArray<CappasityModel *> * _Nonnull)models
Returns the models obtained.
Parameters:
 - models – the array of models

@interface CappasityModelsList

Obtaining of the user models.

Overview

The registered user may obtain a list of all their models sorted by the order in which the models were created.

Properties and methods

- (nullable instancetype)initWithDelegate:(id <CappasityModelsListDelegate> _Nonnull)delegate
Returns an instance of the `CappasityModelsList` class in case of a successful account login. Returns “null” otherwise.
Parameters:
 - delegate – the delegate object.
- (BOOL)nextModelsWithLimit:(NSInteger)limit offset:(NSInteger)offset
Request to obtain the set of models.
Parameters:
 - limit – the number of models requested. Range of values: [1, 100].
 - offset – the sequential rank of the model from which the next set commences. Range of values: [0,...)

- (BOOL)nextModelsWithLimit:(NSInteger)limit
Request to obtain the next set of models.
Parameters:
 - limit – the number of models requested. Range of values: [1, 100].
- (BOOL)isValid
Verification of class validity. The method returns “false” in the case of an error during the request, and returns “true” otherwise.
- (void)clearStatus
A reset of all object properties to their original state. Following this operation, the nextModelsWithLimit: method will return the models from the very beginning of the list.

Working with the Model

@interface CappasityEmbedCodeParams

Collection of embed-code parameters.

Overview

Used for customizing the embed code.

Properties

- @property (nonatomic, copy) NSString * _Nonnull width
Window width, given in pixels or percentages. For example, “600px” or “100%”.
- @property (nonatomic, copy) NSString * _Nonnull height
Window height, given in pixels or percentages.
- @property (nonatomic) BOOL autoRotate
Switch the automatic rotation of the object on/off.
- @property (nonatomic) BOOL autoRun
Switch autorun on/off.
- @property (nonatomic) BOOL closeButton
Show/hide the Close button.
- @property (nonatomic) BOOL showLogo
Show/hide the logo.
- @property (nonatomic) float autoRotateTime
Time for a single rotation of the object during automatic rotation. Value specified in seconds. Available when autoRotate is enabled.
- @property (nonatomic) float autoRotateDelay
Time delay prior to the start of the automatic rotation. Value specified in seconds. Available when autoRotate is enabled.
- @property (nonatomic) NSInteger autoRotateDir

Direction of the automatic rotation. 1 – clockwise, -1 – counterclockwise. Available when autoRotate is enabled.

@property (nonatomic) BOOL hideFullScreen
Show/hide the Full Screen button.

@property (nonatomic) BOOL hideAutoRotateOptions
Show/hide the Automatic Rotation button. Available when autoRotate is enabled.

@property (nonatomic) BOOL hideSettingsButton
Show/hide the Settings button.

@property (nonatomic) BOOL enableImageZoom
Switch zoom mode on/off.

@property (nonatomic) NSInteger zoomQuality
Zoom quality. Possible values:
1 – SD quality,
2 – HD quality.

@property (nonatomic) BOOL hideZoomOption
Show/hide zoom options.

@protocol CappasityModelDelegate

A collection of methods that may be called by the CappasityModel class object.

Overview

Inherits the CappasityDelegate protocol.

Methods

- (void)modelInfoReceived
Tells the delegate of having successfully obtained the requested model information.
- (void)modelUpdated
Tells the delegate of having successfully updated the model properties.
- (void)modelRemoved
Tells the delegate of having successfully deleted the model.
- (void)previewReceivedWithPreview:(UIImage * _Nonnull)preview userInfo:(id _Nullable)userInfo
Returns the requested model preview.
Parameters:
 - preview – model preview image
 - userInfo – user information entered during the preview request.

@interface CappasityModel

Working with the model.

Overview

The object may be initialized with the help of the model's id or sku. The `receiveInfo:` method should then be called to obtain the model information.

```
CappasityModel *model = [[CappasityModel alloc] initWithSku: @"123456789" delegate:self];
[model receiveInfo];
...
//the object is ready for use if the request has been successfull (the delegate received the
//modelInfoReceived: method).
printf("%s", [model.sku UTF8String]);
```

There is no need to call the `receiveInfo()` method if the model has been obtained via a request for a list of user models.

```
- (void)receivedWithModels:(NSArray<CappasityModel *> * _Nonnull)models {
    if ([models count] > 0) {
        printf("%s", [models[0].sku UTF8String]);
    }
}
```

Since the delegate has not been set during the initialization of the object in this case, this should be done directly if needed.

```
CappasityModel *model = models[0]
model.delegate = self
```

Properties and methods

`@property (nonatomic, weak) id <CappasityModelDelegate> _Nullable delegate`
The delegate object.

`@property (nonatomic, readonly, copy) NSString * _Nullable id`
Model id.

`@property (nonatomic, readonly, copy) NSString * _Nullable title`
Model title.

`@property (nonatomic, readonly, copy) NSString * _Nullable sku`
Model sku.

`@property (nonatomic, readonly) enum FileAccessType access`
Model visibility setting.
FileAccessType values:

<code>publicAccess</code>	model is visible to everyone
<code>unlistedAccess</code>	model is accessible only through a direct link
<code>privateAccess</code>	model is visible only to owner

`-(nonnull instancetype)initWithId:(NSString * _Nonnull)id delegate:(id <CappasityModelDelegate> _Nullable)delegate`
Returns an instance of the `CappasityModel` class
Parameters:

- id – model id.
 - delegate – the delegate object.
- (nonnull instancetype)initWithSku:(NSString * _Nonnull)sku delegate:(id <CappasityModelDelegate> _Nullable)delegate
Returns an instance of the CappasityModel class
Parameters:
- sku – model sku.
 - delegate – the delegate object.
- (void)receiveInfo
Sends a request to obtain the model information. Calls the delegate method modelInfoReceived: in the case of a successful execution.
- (void)remove
Sends a request to delete the model. Calls the delegate method modelRemoved: in the case of a successful execution.
- (void)getPreviewWithWidthPx:(NSUInteger)widthPx heightPx:(NSUInteger)heightPx userInfo:(id _Nullable)userInfo
Sends a request to obtain the model preview. Calls the delegate method previewReceivedWithPreview:userInfo: in the case of a successful execution.
Parameters:
- widthPx – width of requested preview in pixels.
 - heightPx – height of requested preview in pixels.
 - userInfo – user information, returned together with the obtained model preview.
- (void)getPreviewWithWidthPt:(NSUInteger)widthPt heightPt:(NSUInteger)heightPt userInfo:(id _Nullable)userInfo
Sends a request to obtain the model preview. Calls the delegate method previewReceivedWithPreview:userInfo: in the case of a successful execution.
Parameters:
- widthPt – width of requested preview in points.
 - heightPt – height of requested preview in points.
 - userInfo – user information, returned together with the obtained model preview.
- (void)updateWithTitle:(NSString * _Nonnull)title sku:(NSString * _Nonnull)sku access:(NSInteger)access
Sends a request to update the model information. Calls the delegate method modelUpdated: in the case of a successful execution.
Parameters:
- title – new model title
 - sku – new model sku
 - access – new model access type
- (void)updateWithTitle:(NSString * _Nonnull)title access:(NSInteger)access
- (void)updateWithTitle:(NSString * _Nonnull)title sku:(NSString * _Nonnull)sku access:(NSInteger)access
- (void)updateWithSku:(NSString * _Nonnull)sku access:(NSInteger)access
- (void)updateWithTitle:(NSString * _Nonnull)title sku:(NSString * _Nonnull)sku access:(NSInteger)access
- (void)updateWithTitle:(NSString * _Nonnull)title sku:(NSString * _Nonnull)sku
- (void)updateWithTitle:(NSString * _Nonnull)title sku:(NSString * _Nonnull)sku access:(NSInteger)access
- (void)updateWithTitle:(NSString * _Nonnull)title
- (void)updateWithTitle:(NSString * _Nonnull)title sku:(NSString * _Nonnull)sku access:(NSInteger)access

- (void)updateWithSku:(NSString * _Nonnull)sku
- (void)updateWithTitle:(NSString * _Nonnull)title sku:(NSString * _Nonnull)sku access:(NSInteger)access
- (void)updateWithAccess:(NSInteger)access
- (void)updateWithTitle:(NSString * _Nonnull)title sku:(NSString * _Nonnull)sku access:(NSInteger)access
- (NSString * _Nullable)embedCodeWith:(CappasityEmbedCodeParams * _Nonnull)params
Returns the embed code of the model with the specified parameters.
Parameters:
 - params – collection of embed-code parameters.

Displaying the Model

@interface CappasityModelViewParams

Collection of model display parameters.

Overview

Used for customizing the model display.

Properties

- @property (nonatomic) BOOL autoRotate
Switch the automatic rotation of the object on/off.
- @property (nonatomic) BOOL autoRun
Switch autorun on/off.
- @property (nonatomic) BOOL showLogo
Show/hide the logo.
- @property (nonatomic) float autoRotateTime
Time for a single rotation of the object during automatic rotation. Value specified in seconds. Available when autoRotate is enabled.
- @property (nonatomic) float autoRotateDelay
Time delay prior to the start of the automatic rotation. Value specified in seconds. Available when autoRotate is enabled.
- @property (nonatomic) NSInteger autoRotateDir
Direction of the automatic rotation. 1 – clockwise, -1 – counterclockwise. Available when autoRotate is enabled.
- @property (nonatomic) BOOL hideAutoRotateOptions
Show/hide the Automatic Rotation button. Available when autoRotate is enabled.
- @property (nonatomic) BOOL hideSettingsButton
Show/hide the Settings button.
- @property (nonatomic) BOOL enableImageZoom
Switch zoom mode on/off.
- @property (nonatomic) NSInteger zoomQuality

Zoom quality. Possible values:

- 1 – SD quality,
- 2 – HD quality.

@property (nonatomic) BOOL hideZoomOption
Show/hide zoom options.

@protocol CappasityModelViewDelegate

A set of methods that may be called by the CappasityModelView class object.

Overview

Inherits the CappasityDelegate protocol.

@interface CappasityModelView

Displays the model with the specified parameters.

Overview

The CappasityModelView class inherits UIView. It can be appointed to an object directly in the storyboard. The “set” method is used to specify the model to be displayed, as well as the display parameters.

In order to enable transition to Cappasity's official website when clicking on the logo, viewController property must be set.

Properties and methods

@property (nonatomic, weak) id <CappasityModelViewDelegate> _Nullable delegate
The delegate object.

@property (nonatomic, weak) UIViewController * _Nullable viewController
The UIViewController object. Must be set for working logo button.
Clicking on the Cappasity logo will have no effect if this object is not specified.

- (void)stopLoading
Stops the loading of the model.

- (void)setWithModel:(CappasityModel * _Nonnull)model params:(CappasityModelViewParams * _Nonnull)params
Specifies the model to be displayed and the display parameters.
Parameters:
- model – model to be viewed
- params – collection of model display parameters.

Error List

Codes of potential errors encountered during program operation.

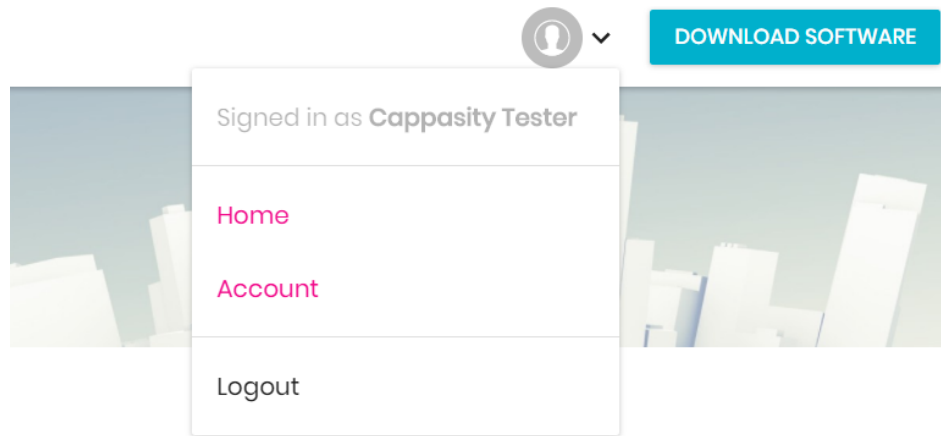
Code	Description
102	The login was unsuccessful.

103	You must log into the account first.
104	The token should not be empty.
105	The model has not been initialized. Call receiveInfo to obtain the model parameters.
106	The sku should not be empty.
107	The account name should not be empty.
108	The account name should only have 0-9, a-z/A-Z, and '.' characters.
109	The sku should only have 0-9, a-z/A-Z, '.', '_', and '-' characters.
110	The account name should contain at least 3 characters.
111	The password should contain at least 6 characters.
112	The title should not be empty or contain more than 75 characters.
113	The sku should contain no more than 50 characters.
150	The request was unsuccessful.
151	The request to obtain the model was unsuccessful.
152	The request to obtain the list of models was unsuccessful.
153	The request to obtain the model preview was unsuccessful.
154	The request to obtain the model parameters was unsuccessful.
155	The deletion of the model was unsuccessful.
156	The model parameter update was unsuccessful.
157	The modification of the model access type was unsuccessful.
158	The request to obtain the model's embed code was unsuccessful.
159	The video could not be processed.
160	Uploading model to the cloud server was unsuccessful.
161	The model's embed code could not be obtained.
162	The user search request was unsuccessful.
163	The file is being processed. Please wait for up to 15 minutes.
164	You can modify the file access settings if your activated paid account is Basic or higher.
300	Not enough memory.
301	Your request cannot be processed.

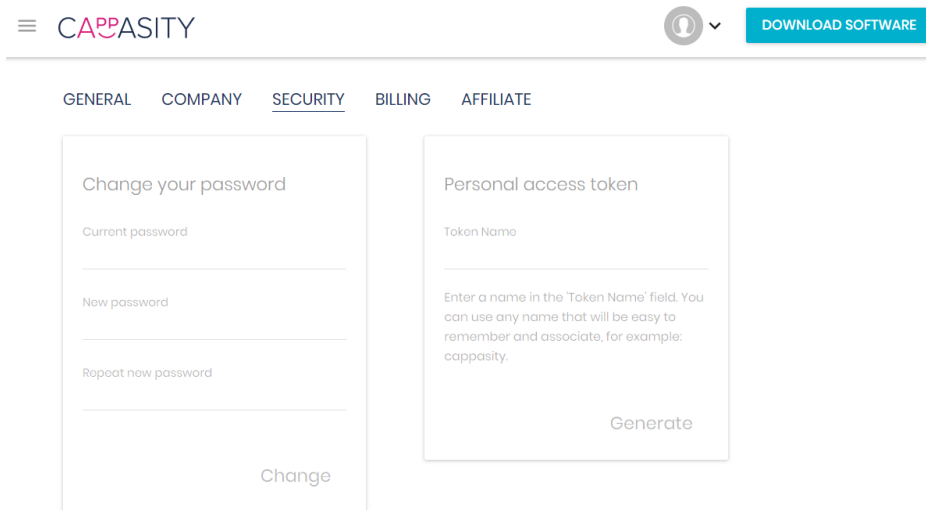
Obtaining the Token

To generate the token, you need to perform the following steps:

- 1) sign up for an account on the Cappasity platform at <https://3d.cappasity.com/>, or log into your existing account,
- 2) click on the Account tab in the menu,



3) click on the Security tab,



- 4) enter the token name and click on the Generate button,
- 5) copy the token obtained.