



Zoom Mac-RTC-Stack

Zoom Mac-RTC-Stack

Modification History

Revision	Date	Originator	Comments
1	8/10/2016	Zoom engineering	First version

Note that the Mac-RTC-Stack from Zoom is distributed under a separate RTC-Stack agreement. Please make sure that you read the terms and conditions of the RTC-Stack agreement before using the RTC-Stack. Please visit [Zoom developer portal](#) if you have questions.

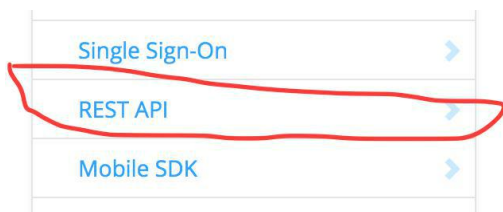


Zoom Mac-RTC-Stack

1 Introduction and Pre-Requisite

In order to use the RTC-Stack, you need to make sure that the REST API and the mobile SDK is enabled in your account. If not, please contact your account rep or contact Zoom support. Our mac RTC-Stack is written in object C.

- Get the SDK key and secret from your zoom account. This key/sec is same as the one used for mobile SDK



- Get the REST API Key/Sec from your zoom account

Credential	Playground
API Key:	KuqoKDxjQT2dtDb9Yis47A
API Secret:	***** Show Regenerate

- Call REST API “getbyemail” and this should return the user id and user token



Zoom Mac-RTC-Stack

API Playground

API Endpoint:

https://api.zoom.us/v1/user/getbyemail

API Key: *

KuqoKDxjQT2dtDb9Yis47A

API Secret: *

ylQrZ6uPduFwZ2QpPyhtGawPytNqAEdKYdGd

Data Type:

☒ JSON

☐ XML

[Clear](#)

User Email Address: *

abc@test.us

Login Type: *

☒ Work Email

☐ Google

☐ Facebook

☐ SSO

☐ API

[Send API Request](#)

Post Data:

[View Source](#)

```
{
  "id": "LeE4XRa8RkaCxtqgXts37w",
  "disable_jbh_reminder": false,
  "enable_cmrr": true,
  "enable_auto_recording": true,
  "enable_cloud_auto_recording": t
  "timezone": "America/Los_Angeles",
  "created_at": "2015-07-23T23:40:58Z",
  "token": "mL4KVYD-
8f12TtnE1nFM5leEMjNXT4xfgdFj62PmqNg.Bgl5b0NFOURZNStNZGJuNXhXSGY5SW
RydGovdCtldW5oV0tzQ0xwbnBFODRqTT1AN2QxM2I5NWlwZTVlZDliOTE5OTE1ZW
lwNDM4ZTNmMmVIY2E0MmMzJzJlE5MmUyZDdjNzY0YzQwOWU1NjliMjExYgA"
}
```



2 RTC-Stack Functions

2.1 RTC-Stack Initialization

Please follow the ZoomSDKSample.xcproject project that is part of our SDK bundle (AppDelegate.mm) and look at the initialization functions to find how to use ZoomSDK.

Interfaces:

- 1) `sharedSDK`
@return an ZoomSDK instance, e have done initialisation ourself.
- 2) `setZoomDomain:(NSString*)domain`
@param domain:set a web domain url you want to use.
- 3) `getAuthService`
@return ZoomSDKAuthService instance, use this to call Auth fuction.
- 4) `getMeetingService`
@return ZoomSDKMeetingService instance, use this to get meeting detail info and modify meeting UI.
- 5) `getSettingService`
@return ZoomSDKSettingService instance, use this to configure the meeting setting.
- 6) `getPremeetingService`
@return ZoomSDKPremeetingService instance, use this to schedule\list meeting.

2.2 Auth Service

Zoom SDK supports two options for user authentication.

- You can pass the user id of the user to the SDK initialization and all the meetings will start on that user's account. This is referred to as API user
- There are times it might be cumbersome to get the user id and you might want to distribute your app to all Zoom users – in this case, you can ask the user to enter the



Zoom Mac-RTC-Stack

Zoom login credentials (username /password) in your app and then pass it to the SDK.
This is referred to as normal user

Use ZoomSDKAuthService interface for the user authentication.

- 1) (ZoomSDKError) sdkAuth: (NSString*)key appSecret: (NSString*)secret
 @param key: your client key, also known as API key.
 @param secret: your client secret
 @return a error enum to tell the call result directly,
 and also can get a callback from its delegate.
- 2) (BOOL) isAuthorized
 @return a BOOL value to tell you the ZoomSDK is authed or not.
- 3) (ZoomSDKError) login: (NSString*)userName
 Password: (NSString*)password RememberMe: (BOOL)rememberMe;
 @param userName: your login user name
 @param password: password related to your username
 @param rememberMe: if you want to login automatically next
 time, set YES, or NO to not remember your password. @return a
 error enum to tell the call result directly, and also can get
 a callback onZoomSDKLogin from its delegate.
- 4) - (ZoomSDKError) logout;
 @return a error enum to tell the call result directly, and
 also can get a callback onZoomSDKLogout from its delegate.
- 5) - (ZoomSDKAccountInfo*) getAccountInfo
 @return ZoomSDKAccountInfo object if call
 successfully. ZoomSDKAccountInfo
 interface: - (NSString*) getDisplayName
 return the display name of the login account.

Callback from delegate:

- 1) (void) onZoomSDKAuthReturn: (ZoomSDKAuthError) returnValue
 @return a error enum to tell the auth process successfully or not.
- 2) (void) onZoomSDKLogin: (ZoomSDKLoginStatus) loginStatus;
 @return a ZoomSDKLoginStatus enum to tell if
 client login successfully.
 ZoomSDKLoginStatus enum:
 ZoomSDKLoginStatus_Idle, // Not login yet
 ZoomSDKLoginStatus_Processing, // Login in progress
 ZoomSDKLoginStatus_Success, // Login Success
 ZoomSDKLoginStatus_Failed // Login Failed
- 3) (void) onZoomSDKLogout



Zoom Mac-RTC-Stack

2.3 Pre-Meeting Service

:

```
1) (ZoomSDKError)scheduleMeeting:(ZoomSDKScheduleMeetingItem*)meetingItem;
    @param meetingItem the specified meeting user want to schedule. ZoomSDKScheduleMeetingItem interface:
    You can set follow property of
    ZoomSDKScheduleMeetingItem NSString* meetingTopic;
    NSString* meetingPassword;
    time_t meetingStartTime;
    time_t meetingDuration;
    BOOL joinBeforeHost;
    BOOL usePMI;
    BOOL turnOffVideoForHost;
    BOOL turnOffVideoForAttendee;
    @return A ZoomSDKError to tell client whether the meeting has been
    scheudled successfully or not synchronously, also get callback
    onScheduleOrEditMeting from delegate asynchronously.
2) (ZoomSDKError)editMeeting:(ZoomSDKScheduleMeetingItem*)meetingItem
    m MeetingUniqueID:(unsigned int)meetingUniqueID;
* @param meetingItem the user create to edit meeting.
* @param meetingUniqueID the specified meeting Unique ID user want
    to edit.
    @return A ZoomSDKError to tell client whether the meeting has
    been scheudled successfully or not synchronously.also get
    callback onScheduleOrEditMeting from delegate asynchronously.
3) - (ZoomSDKError)deleteMeeting:(unsigned int)meetingUniqueID;
    @param meetingUniqueID the specified meeting Unique ID user
    want to delete.
    @return A ZoomSDKError to tell client whether the meeting has
    been deleted successfully or not synchronously, also get
    callback onDeleteMeeting from delegate asynchronously.
4) - (ZoomSDKError)listMeeting
    This method is used to list all meetings.
    @return A ZoomSDKError to tell client whether list meeting action
    successfully or not synchronously, also get callback
    onListMeeting from delegate asynchronously.
5) - (ZoomSDKMeetingItem*)getMeetingItem:(unsigned
    int)meetingUniqueID;
    @param meetingUniqueID the specified meeting Unique ID user want
    to get.
    @return A ZoomSDKMeetingItem object with meeeting info if function
```



Zoom Mac-RTC-Stack

```
call successfully.
ZoomSDKMeetingItem interface:
(BOOL)isPersonalMeeting;
(BOOL)isWebinarMeeting;
(BOOL)isRecurringMeeting;
(BOOL)isAllowJoinBeforeHost;
(BOOL)isUsePMIAsMeetingID;
(long long) getMeetingUniqueID;
(long long) getMeetingNumber;
(time_t) getMeetingStartTime;
(time_t) getMeetingDuration;
(NSString*) getMeetingTopic;
(NSString*) getMeetingPassword;
```

Callback from delegate:

- 1) - (void)onListMeeting:(ZoomSDKPremeetingError)error
MeetingList:(NSArray*)meetingList
@param error tell client related to this premeeting event whether list meeting success or not.
@param meetingList A NSArray contains elements of ZoomSDKMeetingItem
- 2) - (void)onScheduleOrEditMeeting:(ZoomSDKPremeetingError)error
MeetingUniqueID:(unsigned int)meetingUniqueID
@param error tell client related to this premeeting event whether schedule or edit meeting successfully or not.
@param meetingUniqueID the unique id of the meeting you schedule or edit.
- 3) - (void)onDeleteMeeting:(ZoomSDKPremeetingError)error
@param error tell client related to this premeeting event whether delete meeting successfully or not.

2.4 Meeting Service

After auth is successful, you can use ZoomSDKMeetingService to modify meeting and get meeting detail info.

2.4.1 Start, Join, Leave meeting interfaces

- 1) - (ZoomSDKError)startMeeting:(ZoomSDKUserType)userType
userID:(NSString*)userID userToken:(NSString*)userToken
displayName:(NSString*)username
meetingNumber:(NSString*)meetingNumber
isDirectShare:(BOOL)directShare
sharedApp:(CGDirectDisplayID)displayID;
@param userType: The userType depends on what the client account is, sdk user should use ZoomSDKUserType_APIUser or zoom user: ZoomSDKUserType_ZoomUser



Zoom Mac-RTC-Stack

```
for api user userId\userToken\userNam\meetingNumber is
requested, but not need for ZoomSDKUserType_ZoomUser
@param userId: The userId received as a result client
user account from Zoom site.
@param userToken: The userToken received as a result
client user account from Zoom site.

@param username: The username will be used as display name
in the Zoom meeting.

@param meetingNumber The meetingNumber may be generated from a
scheduled meeting or a Personal Meeting ID, api user can't start
instant meeting, if zoom user want to start a instant meeting, set
meetingNumber to nil.

@param directShare: set YES client will start desktop
share directly when meeting started.

@param displayID: set app display ID   client will start share this
app directly when meeting started.

@return A ZoomSDKError to tell client whether the meeting started
or not
callback: - (void)onMeetingReturn:(ZoomSDKMeetingError)error
internalError:(NSInteger)internalError
2) - (ZoomSDKError)joinMeeting:(ZoomSDKUserType)userType
toke4enfrocelogin:(NSString*)toke4enfrocelogin
participantId:(NSString*)participantId
meetingNumber:(NSString*)meetingNumber
displayName:(NSString*)username password:(NSString*)pwd
isDirectShare:(BOOL)directShare
sharedApp:(CGDirectDisplayID)displayID;
@param userType: The userType depends on what the client
account is, sdk user should use ZoomSDKUserType_APIUser or
zoom user: ZoomSDKUserType_ZoomUser
for api user you can input parameter: toke4enfrocelogin\
participantId, but not permit for ZoomSDKUserType_ZoomUser
@param toke4enfrocelogin when join a meeting need login,
this param is needed.
@param participateId
@param username The username will be used as display name in
the Zoom meeting.

@param meetingNumber The meetingNumber used to join the meeting.
@param pwd, The meeting password which used to join the
meeting, if there does not exist meeting password, just set pwd
to nil or @"".
@param directShare: set YES client will start desktop
share directly when meeting joined.

@param displayID: set app display ID   client will start share this
app directly when meeting joined.

@return A ZoomSDKError to tell client whether can join
the meeting or not
```




Zoom Mac-RTC-Stack

```
callback: - (void)onMeetingReturn:(ZoomSDKMeetingError)error
internalError:(NSInteger)internalError
3) - (void)leaveMeetingWithCmd:(LeaveMeetingCmd)cmd
@param cmd, leave meeting by the command type, only host can
user command LeaveMeetingCmd_End to end meeting, or you can
use LeaveMeetingCmd_Leave to leave meeting.
```

2.4.2 Modify Meeting interface

2.4.2.1 Show, hide, move and minimize UI window in Meeting Interface

Use - (ZoomSDKMeetingUIController*)getMeetingUIController in ZoomSDKMeetingService to get ZoomSDKMeetingUIController instance.

```
1) - (ZoomSDKError)showMeetingComponent:(MeetingComponent)component
window:(NSWindow**)window show:(BOOL)show InPanel:(BOOL)inPanel
frame:(CGRect)componentFrame;
@param component: a enum specify the meeting component
window you want to modify.
    MeetingComponent_MainWindow : main meeting window
    MeetingComponent_Audio: audio window
    MeetingComponent_Chat: chat window
    MeetingComponent_Participants: participate list
    window MeetingComponent_MainToolBar: toolbar at the
    bottom of main video window
    MeetingComponent_ShareToolBar: toolbar at the top
    of sharing window
    MeetingComponent_Setting setting window
@param window: you can set a NSWindow object, and it will
be assigned value after the function call successfully, it
only works when @param component select as follow enum:
    MeetingComponent_MainWindow
    MeetingComponent_Audio
    MeetingComponent_Chat
    MeetingComponent_Participan
    t s
    MeetingComponent_Setting
@param show: set YES if you want to show this component
window or No to hide.
@param inPanel: Just for Participants and chat window, set
YES the selected window will show in the left side of main
window, set No window will show independently.
@param rect: the frame of the component you want to show on
screen. Can be moved and resized Component: MainWindow
```



Zoom Mac-RTC-Stack

Just can be moved Component: the other components.

@return A ZoomSDKError enum to tell client whether the meeting UI modify successfully or not synchronously.

- 2) (ZoomSDKError)enterFullScreen:(BOOL)fullScreen
firstMonitor:(BOOL)firstView DualMonitor:(BOOL)secondView
@param fullScreen: set YES to enter full screen or No to exit.
@firstView: Set YES if u want modify first screen or NO not to modify.
@secondView: Set YES if u want modify second screen or NO not to modify.
@return A ZoomSDKError enum to tell client whether enter or exit full screen successfully or not synchronously.
- 3) - (ZoomSDKError)switchToVideoWallView
@return A ZoomSDKError enum to tell client whether main video view switch to video wall view successfully or not synchronously.
- 4) - (ZoomSDKError)switchToActiveSpeakerView;
@return A ZoomSDKError enum to tell client whether main video view switch to active speaker view successfully or not synchronously.
- 5) - (ZoomSDKError)minimizeShareFloatVideoWindow:(BOOL)bMin
@param bMin: set YES to minimize the float video window when share, or NO to maximize.
@return A ZoomSDKError enum to tell client whether minimize or maximize share float video window successfully or not synchronously.
- 6) - (ZoomSDKError)moveFloatVideoWindow:(NSPoint)position
@param position: the position you want the float video window move to when sharing.
@return A ZoomSDKError enum to tell client whether move share float video window successfully or not synchronously.
- 7) - (ZoomSDKError)switchFloatVideoToActiveSpeakerMode
@return A ZoomSDKError enum to tell client whether shaer float video view switch to active speaker view successfully or not synchronously.
- 8) - (ZoomSDKError)switchFloatVideoToGalleryMode
@return A ZoomSDKError enum to tell client whether shaer float video view switch to gallery view successfully or not synchronously.

2.4.2.2 Meeting configuration

Use - (ZoomSDKMeetingConfiguration*)getMeetingConfiguration in ZoomSDKMeetingService to get ZoomSDKMeetingConfiguration instance.



Zoom Mac-RTC-Stack

You can use this to set some meeting configuration before meeting start.

- 1) `(void)reset`
reset the meeting configuration.

Property: you can set follow property before you start meeting

- 1) app display ID for share
`CGDirectDisplayID _displayAppID;`
- 2) monitor ID for share
`CGDirectDisplayID _monitorID;`
- 3) FloatVideo position
`NSPoint _floatVideoPoint;`
- 4) Sharing ToolBar Visible
`BOOL _shareToolBarVisible;`
- 5) Main video position
`NSPoint _mainVideoPoint;`
- 6) waiting for host window visible
`BOOL _jbhWindowVisible;`
- 7) mute on entry
`BOOL _enableMuteOnEntry;`
- 8) play chime or not
`BOOL _enableChime;`
- 9) direct share when start meeting or not
`BOOL _isDirectShareDestop;`

2.4.2.3 Whiteboard Annotation

Use - (ZoomSDKAnnotationController*)getAnnotationController in ZoomSDKMeetingService to get ZoomSDKAnnotationController instance. You can use this to control annotation in share whiteboard.

- 1) - (ZoomSDKError)setTool:(AnnotationToolType)type
@ param type: select a tool to use.
`AnnotationToolType_None,`
`AnnotationToolType_Pen,`
`AnnotationToolType_HighLighter,`
`AnnotationToolType_AutoLine,`
`AnnotationToolType_AutoRectangle,`
`AnnotationToolType_AutoEllipse,`
`AnnotationToolType_AutoArrow,`
`AnnotationToolType_AutoRectangleFill`
`1`
`,`
`AnnotationToolType_AutoEllipseFill`
`, AnnotationToolType_SpotLight,`
`AnnotationToolType_Arrow,`
`AnnotationToolType_ERASER,`



Zoom Mac-RTC-Stack

```

@return A ZoomSDKError enum to tell client whether set
annotation tool successfully or not synchronously.
2) - (ZoomSDKError)clear:(AnnotationClearType)type @param
type: select a clear type to clear annotation.
    AnnotationClearType_All: clear all annotation in
    whiteboard AnnotationClearType_Self: clear your annotation
    in whiteboard AnnotationClearType_Other: clear other's
    annotation in Whiteboard
@return A ZoomSDKError enum to tell client whether
clear annotation successfully or not synchronously.
3) - (ZoomSDKError)setColor:(NSColor*)color
@param color: set color u want annotation to use
@return A ZoomSDKError enum to tell client whether
set color successfully or not synchronously.

4) - (ZoomSDKError)setLineWidth:(long long)lineWidth
@param lineWidth: set format line width u want annotation to
use @return A ZoomSDKError enum to tell client whether set line
width successfully or not synchronously
5) - (ZoomSDKError)undo
@return A ZoomSDKError enum to tell client whether revoke
last annotation action successfully or not synchronously.
6) - (ZoomSDKError)redo
@return A ZoomSDKError enum to tell client whether not to revoke
last annotation action successfully or not synchronously.

```

2.4.2.4 other meeting actions

this contains interfaces about audio\video\share\chat\recording action, and get meeting detail info.

Interfaces:

```

1) (ZoomSDKError)actionMeetingWithCmd:(ActionMeetingCmd)cmd
   userID:(unsigned int)userID
   @param cmd: select a command to specify a action
               to modify. //video cmd
               ActionMeetingCmd_MuteVideo,
               ActionMeetingCmd_UnMuteVideo,
               //audio cmd
               ActionMeetingCmd_MuteAudio,
               ActionMeetingCmd_UnMuteAudio,
               //lock meeting cmd
               ActionMeetingCmd_LockMeeting,
               ActionMeetingCmd_UnLockMeetin
g

```




Zoom Mac-RTC-Stack

```
//remote control cmd
ActionMeetingCmd_RequestRemoteControl,
ActionMeetingCmd_GiveUpRemoteControl,
ActionMeetingCmd_GiveRemoteControlTo,
ActionMeetingCmd_DeclineRemoteControlRequest ,
ActionMeetingCmd_RevokeRemoteControl,
ActionMeetingCmd_GetCurrentRemoteController,
//lock share cmd
ActionMeetingCmd_LockShare,
ActionMeetingCmd_UnlockShare,
//lower all hands
ActionMeetingCmd_LowerAllHands,
//switch share mode
ActionMeetingCmd_ShareFitWindowMode
,
ActionMeetingCmd_ShareOriginSizeMod
e
,
@return A ZoomSDKError enum to tell client whether
modify meeting action successfully or not synchronously.
```

- 2) - (ZoomSDKError)sendChat:(NSString*)content
toUser:(unsigned int)userID
@param content, message content you want to send.
@param userID, userID of the user you want to send chat
to. @return A ZoomSDKError to tell client whether send
message successful or not.
- 3) - (ZoomSDKError)startRecording:(time_t)startTimestamp
saveFilePath:(NSString*)filePath
@param startTimestamp, start recording timestamp.
@param filePath, the path u want to save recording
file. @return A ZoomSDKError to tell client whether
start recording successful or not.
- 4) - (ZoomSDKError)stopRecording:(time_t)stopTimestamp
@param stopTimestamp, start recording timestamp.
@return A ZoomSDKError to tell client whether stop
recording successful or not.
- 5) - (ZoomSDKError)startAppShare:(CGWindowID>windowID
displayID:(CGDirectDisplayID)displayID
@param shareAppWindow, the app window u want to
share. @return A ZoomSDKError to tell client
whether start app share successful or not
callback from delegate:
- 6) - (ZoomSDKError)startMonitorShare:(NSString*)monitorID
@param monitorID, the indentify of the monitor u want to
share. @return A ZoomSDKError to tell client whether start
monitor share successful or not.
- 7) - (ZoomSDKError)startAnnotation:(NSPoint)position @param

position, the position of annotation first show.



Zoom Mac-RTC-Stack

```

@return A ZoomSDKError to tell client whether
start annotation share successful or not.
8) - (ZoomSDKError)stopAnnotation
@return A ZoomSDKError to tell client whether stop
annotation share successful or not
9) - (ZoomSDKError)hideOrShowNoVideoUserOnVideoWall:(BOOL)hide
@param hide: set YES if want hide no video user on wall view or
NO to show.
@return A ZoomSDKError to tell client whether function
call successfully or not
10) - (NSArray*)getParticipantsList
@return An array contains participant id.
11) - (ZoomSDKUserInfo*)getUserByUserID:(unsigned
int)userID @param userID, userID of the selected user.
@return A user info interface when function call successful,

                                user name of
                                user email of
                                user userid
                                of user
                                user's host or not
                                user's video is on or
                                not

or return nil when failed.
ZoomSDKUserInfo interfaces:
- (NSString*)getUserName
- (NSString*)getEmail
- (unsigned int)getUserID
- (BOOL)isHost;
- (BOOL)isVideoOn;
- (BOOL)isAudioMuteduser's audio is muted or not
12) - (NSString*)getMeetingProperty:(MeetingPropertyCmd)command
@param command, user command to get differnt property.
      MeetingPropertyCmd_Topic,
      MeetingPropertyCmd_InviteEmailTemplat
      e
      , MeetingPropertyCmd_InviteEmailTitle,
      MeetingPropertyCmd JoinMeetingUrl,
@return A meeting configuration when function
call successful, or return nil when failed.
13) - (ZoomSDKConnectionQuality*)getConnectionQuality
@return A enum that discribe the Connection Quality when
function call successful.
      ZoomSDKConnectionQuality Unknow,
      ZoomSDKConnectionQuality VeryBad,
      ZoomSDKConnectionQuality_Bad,
      ZoomSDKConnectionQuality_NotGood,
      ZoomSDKConnectionQuality_Normal,
      ZoomSDKConnectionQuality_Good,
      ZoomSDKConnectionQuality_Excellent
      ,
14) - (ZoomSDKMeetingStatus)getMeetingStatus
      //Idle
      ZoomSDKMeetingStatus Idle = 0,
      //Connecting
      ZoomSDKMeetingStatus_Connecting = 1,
      //waiting for the host
      = 2,

```



```
ZoomSDKMeetingStatus_WaitingForHost
//In Meeting
ZoomSDKMeetingStatus_InMeeting      = 3,
//Disconnecting
ZoomSDKMeetingStatus_Disconnecting  = 4,
//Reconnecting
ZoomSDKMeetingStatus_Reconnecting   = 5,
//Connect Failed                     = 6,
```

ZoomSDKMeetingStatus_Failed



Zoom Mac-RTC-Stack

```

//Meeting Ended
ZoomSDKMeetingStatus_Ended = 7,
15) (ZoomSDKError)getCurrentRemoteController:(unsigned int*)userID
@param userID, input a unsigned int* value to this function
if fuction call successfully will set the userset id of
current remote controller in userID.
@return A enum that discribe function call successful or not.

16) - (ZoomSDKError)enableMuteOnEntry:(BOOL)bEnable
@param bEnable, set YES to mute attendee when they join
meeting. @return A enum that discribe function call successful
or not. Make sure call this function after u get meeting status
change: ZoomSDKMeetingStatus_AudioReady.

17) enablePlayChimeWhenEnterOrExit:(BOOL)bEnable
@param bEnable, set YES to play chime when user join or
leave meeting. But the first attendee join meeting will
always play chime for echo test so this will take effect
from the second attendee
@return A enum that function call successful or not.

```

2.4.3 Callback from delegate

Callbacks used to mention status change of chat\video\audio\share\recording\meeting
 You should inherit the ZoomSDKMeetingServiceDelegate and implement each interface if you
 want to monitor those notifications.

```

1) - (void)onMeetingReturn:(ZoomSDKMeetingError)error
internalError:(NSInteger)internalError
@param error: tell client related to this meeting event.
ZoomSDKMeetingError_Success = 0,
//Network Unavailable
ZoomSDKMeetingError_NetworkUnavailable = 1,
//Reconnect Fail
ZoomSDKMeetingError_ReconnectFailed = 2,
// MMR Errorr
ZoomSDKMeetingError_MMRError = 3,
// Password Error
ZoomSDKMeetingError_PasswordError = 4,
// Create Seession Error
ZoomSDKMeetingError_SessionError = 5,
//Meeting is over
ZoomSDKMeetingError_MeetingOver = 6,
//Meeting haven't start
ZoomSDKMeetingError_MeetingNotStart = 7,
//Meeting does not exist
ZoomSDKMeetingError_MeetingNotExist = 8,
//User is Full
ZoomSDKMeetingError_UserFull = 9,
//Client Version less than Min Version
ZoomSDKMeetingError_ClientIncompatible = 10,

```



Zoom Mac-RTC-Stack

```

//No MMR ready for service
ZoomSDKMeetingError_NoMMR = 11,
//Meeting has been locked
ZoomSDKMeetingError_MeetingLocked = 12,
//Meeting Restricted
ZoomSDKMeetingError_MeetingRestricted = 13,
//JBH Meeting Restricted
ZoomSDKMeetingError_MeetingJBHRestricted = 14,
//Failed to send create meeting command to our web server. HTTP
post fail
ZoomSDKMeetingError_EmitWebRequestFailed = 15,
//identity token send from client expired
ZoomSDKMeetingError_StartTokenExpired = 16,
//
ZoomSDKMeetingError_VideoSessionError = 17,
ZoomSDKMeetingError_AudioAutoStartError = 18,
ZoomSDKMeetingError_RegisterWebinarFull = 19,
//Webinar
ZoomSDKMeetingError_RegisterWebinarHostRegister = 20,
ZoomSDKMeetingError_RegisterWebinarPanelistRegister = 21,
ZoomSDKMeetingError_RegisterWebinarDeniedEmail = 22,
ZoomSDKMeetingError_RegisterWebinarEnforceLogin = 23,
//ZC certificate changed
ZoomSDKMeetingError_ZCCertificateChanged = 24,
//Failed to write to config file
ZoomSDKMeetingError_ConfigFileWriteFailed = 50,
//Unknown error
ZoomSDKMeetingError_Unknown = 100,
@param internalError Zoom internal error code
2) - (void)onMeetingStatusChange:(ZoomSDKMeetingStatus)state
    @param state tell client meeting state change.
        //Idle
        ZoomSDKMeetingStatus_Idle = 0,
        //Connecting
        ZoomSDKMeetingStatus_Connecting = 1,
        //waiting for the host
        ZoomSDKMeetingStatus_WaitingForHost = 2,
        //In Meeting
        ZoomSDKMeetingStatus_InMeeting = 3,
        //Disconnecting
        ZoomSDKMeetingStatus_Disconnecting = 4,
        //Reconnecting
        ZoomSDKMeetingStatus_Reconnecting = 5,
        //Connect Failed
        ZoomSDKMeetingStatus_Failed = 6,
        //Meeting Ended
        ZoomSDKMeetingStatus_Ended = 7,
        ZoomSDKMeetingStatus_AudioReady = 8

3) (void)onUserAudioStatusChange:(NSArray*)userAudioStatusArray
    @param userAudioStatusArray a array contains
    ZoomSDKUserAudioStauts elements tell client audio stauts
    change of each user.
    ZoomSDKUserAudioStauts interfaces:

```



Zoom Mac-RTC-Stack

```
- (unsigned int)getUserID;
- (ZoomSDKAudioStatus)getStatus
ZoomSDKAudioStatus_None,
ZoomSDKAudioStatus_Muted,
ZoomSDKAudioStatus_UnMuted,
ZoomSDKAudioStatus_MutedByHost,
ZoomSDKAudioStatus_UnMutedByHost,
ZoomSDKAudioStatus_MutedAllByHost,
ZoomSDKAudioStatus_UnMutedAllByHos
t
,
```

You can use this

```
for (ZoomSDKUserAudioStauts* key in userAudioStatusArray) {
    unsigned int userID = [key getUserID];
    ZoomSDKAudioStatus status = [key getStatus];
}
```

- 4) - (void)onChatMessageNotification:(ZoomSDKChatInfo*)chatInfo
@param chatInfo tell client the info of the chat message
info user received, you can get each info from its interface
ZoomSDKChatInfo interfaces:
- (unsigned int)getSenderUserID;
- (NSString*)getSenderDisplayName;
- (unsigned int)getReceiverUserID;
- (NSString*)getReceiverDisplayName;
- (NSString*)getMsgContent;
- (time_t)getTimeStamp;
- 5) - (void)onRecord2MP4Done:(BOOL)success
Path:(NSString*)recordPath @param success tell client
the conversion finish successfully or not .
@param recordPath tell client the mp4 file path.
- 6) - (void)onRecord2MP4Progressing:(int)percentage
@param percentage tell client the process record convert to mp4
- 7) - (void)onRecordStatus:(ZoomSDKRecordingStatus)status
@param status tell client record status.
- 8) - (void)onUserJoin:(NSArray*)array
@param array tell client the joined user array, contains
userid (unsigned int)
- 9) - (void)onUserLeft:(NSArray*)array
@param array tell client the left user array, contains
userid (unsigned int)
- 10) - (void)onRemoteControlStatus:(ZoomSDKRemoteControlStatus)status
User:(unsigned int)userID
@param status the remote controll user's status.
@param userID the remote controll user's identity.
ZoomSDKRemoteControlStatus_None,
//you can request other screen remote control
right ZoomSDKRemoteControlStatus_CanRequest,
//other decline the remote control request
ZoomSDKRemoteControlStatus_Decline,
//other accept the remote control request
ZoomSDKRemoteControlStatus_Accept,

```
//you give up the remote control right of other's  
screen ZoomSDKRemoteControlStatus_GiveUp,
```

```
//you receive the remote control request
```



Zoom Mac-RTC-Stack

```
ZoomSDKRemoteControlStatus_Request,
11) - (void)onSharingStatus:(ZoomSDKShareStatus)status User:(unsigned
    int)userID
    @param status the sharing user's status.
    @param userID the sharing user's identity.
```

2.5 Setting Service

Use - (ZoomSDKSettingService*)getSettingService to get ZoomSDKSettingService instance, and modify meeting setting.

Interfaces:

```
1) - (NSArray*)getCameraList
    @return a array contains CameraInfo
    element. CameraInfo Interfaces:
    - (NSString*)getDeviceID, device ID of camera
    - (NSString*)getDeviceName; device name of camera
    - (BOOL)isSelectedDevice; is the camera has been selected or not
2) - (ZoomSDKError)selectCamera:(NSString*)deviceID
    @param deviceID, the device indentify of camera you want
    select. @return A ZoomSDKError to tell client whether
    select camera successfully or not.
3) - (ZoomSDKError)setRecordingPath:(NSString*)path
    @param path, the default recording path you want to
    set. mode.
    @return A ZoomSDKError to tell client whether function
    call successfully or not
4) - (ZoomSDKError)enableMeetingSetting:(BOOL)enable
    SettingCmd:(MeetingSettingCmd)cmd;
    @param enable, set YES to enable, or NO to disable
    the specific setting.
    @param cmd, MeetingSettingCmd enum contains the
    setting choices you want to modify.
    MeetingSettingCmd_DualScreenMode,
    MeetingSettingCmd_AutoJoinAudio,
    MeetingSettingCmd_AeroModeInShare,
    MeetingSettingCmd_AutoFitToWindowWhenViewShare
    ,
    MeetingSettingCmd_AutoFullScreenWhenJoinMeetin
    g
    ,
    @return A ZoomSDKError to tell client whether
    function call successfully or not.
```




3 SDK Sample

Place ZoomSDK folder in the same directory as ZoomSDKSample.xcodeproj



Sample.APP will be created in Bin/Realse folder.