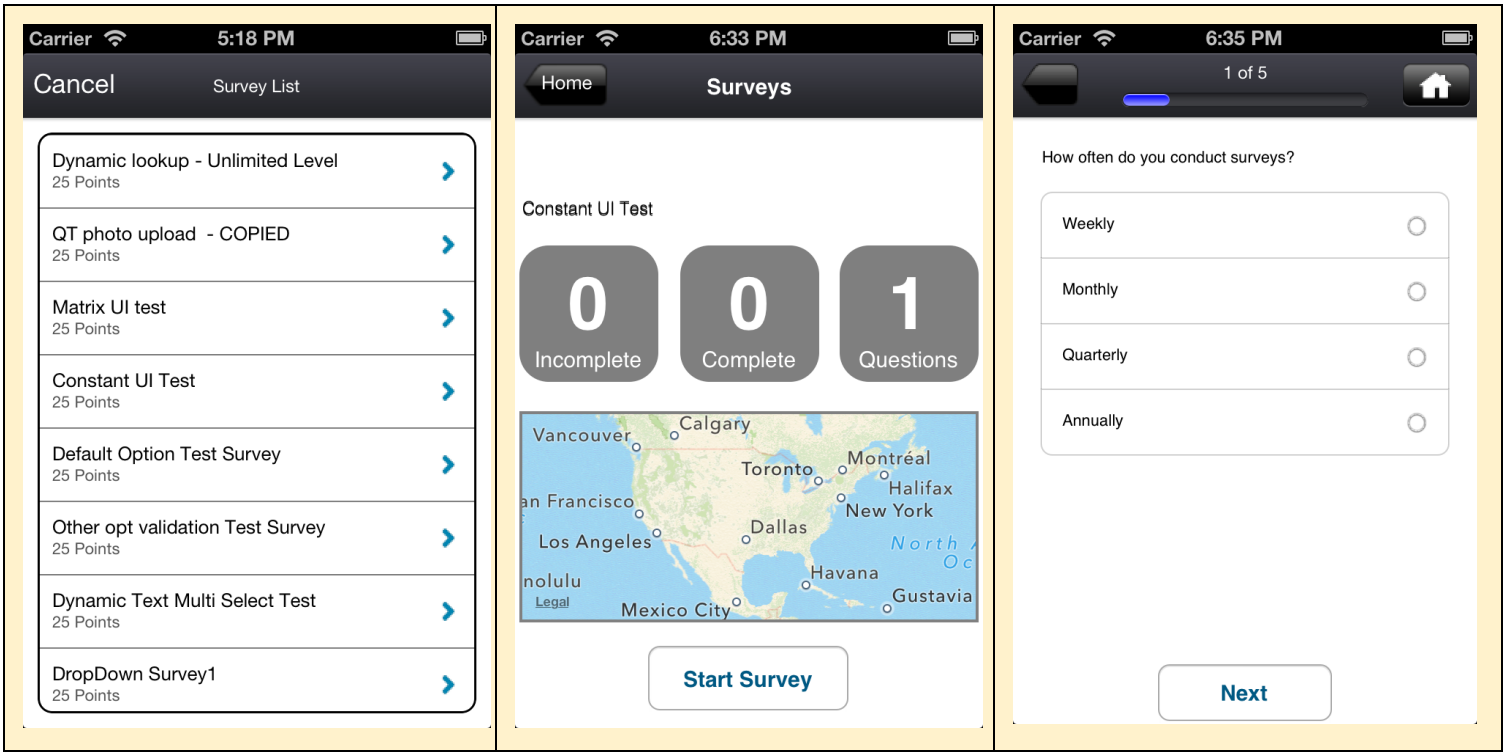
SurveyAnalytics iOS SDK Version1.0

Screenshots

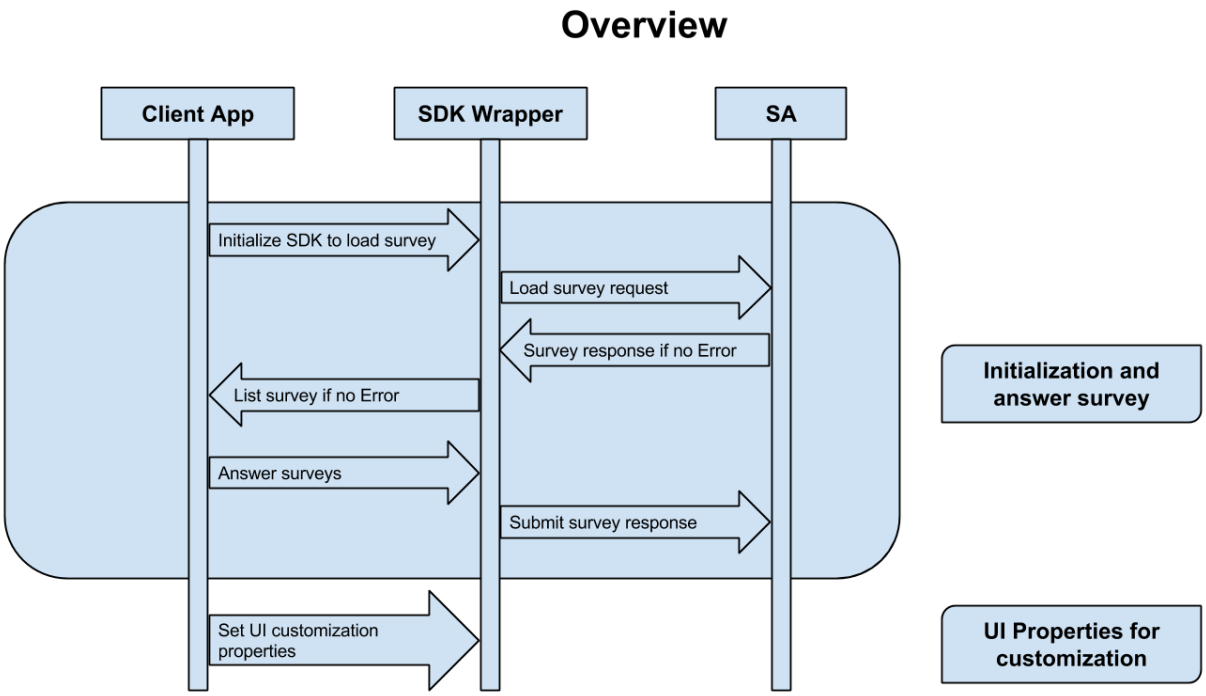


Contents

- Overview
- Use Cases
- Requirements
- Initial Setup
- How to use SDK
- Mandatory setup
- Credentials
- Optional setup
- International Support
- Testing
- Documentation
- Usability
- Next Steps

Overview

SurveyAnalytics iOS Survey SDK provides facilities to integrate our survey module in your existing iPhone application. It provides an easy way to add and configure survey module in your mobile application and collect the survey responses.



Use Cases

The SDK supports two modes or two use cases: List of **Multiple surveys** and **Single Survey**.

Multiple Surveys

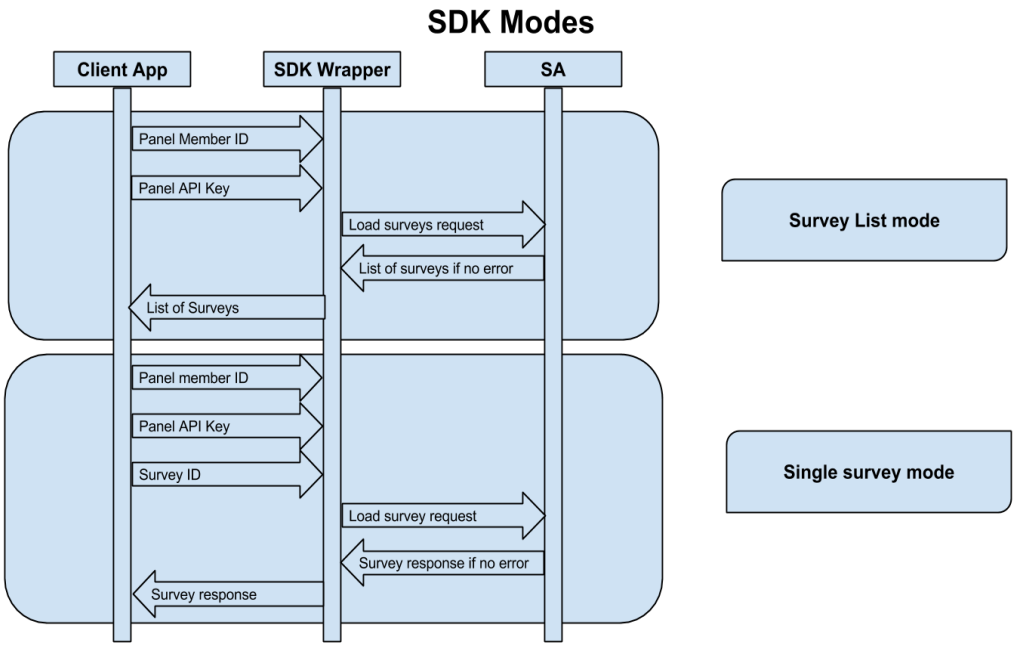
List of all surveys which is assigned to your panel member.

1. Set API key of your account .
2. Set panel member ID of your account.

Single Survey

Get a specific survey which is assigned to your panel member.

1. Set API key of your account.
2. Set panel member ID of your account.
3. Set specific survey ID which you want to answer.



Requirements

- Xcode 5 and iOS SDK 7
- iOS 6.0+ target deployment
- armv6, armv7, armv7s, and arm64 devices, and the simulator
- iPhone and iPad of all sizes and resolutions

Initial Setup of SDK

Add the SDK to Your Project

1. Clone or download the SDK, which consists of header files, license acknowledgements, release notes, and a static library. It also includes a sample app.
2. Add the SAIMobile directory (containing several .h files and SAUICatalog.embeddedFramework) to your Xcode project. We recommend checking "Copy items..." and selecting "Create groups...".
3. In your project's **Build Settings** (in the TARGETS section, not the PROJECTS section):
    - add -all_load and -ObjC to Other Linker Flags
    - enable Link Frameworks Automatically
4. In your project's **Build Phases**, link your project with these libraries. Weak linking for iOS versions back to 6.0 is supported.
    - Mapkit.framework
    - AVFoundation.framework
    - CoreText.framework
    - COreMedia.framework
    - CoreLocation.framework
    - CoreVideo.framework
    - AudioToolbox.framework
    - CoreData.framework
    - COreGraphics.framework
    - MobileCoreServices.framework
    - MediaPlayer.framework
    - SystemConfiguration.framework
    - QuartzCore.framework
    - OpenGLES.framework
    - libiconv.dylib
    - libicucore.dylib
    - libstdc++.dylib
    - SAUICatalog.framework

How to use SDK:

To call SDK intent from your main application, use below lines of code:

1. **Import following classes in your application :**

   #import <SAUICatalog/SASurveyListView.h>

2. **Define and create SASurveyListView object into your class.**

```
@interface SingleSurveyNavigationViewController : UIViewController
{
        SASurveyListView *_saView;
}
@property(nonatomic,retain) IBOutlet  SASurveyListView *saView;
```

3. **Set the properties of created objects.**

```
// To set current controller for Root Observer of SDK callback events
[self.saView setRootViewController:self];

// To set Panel Nember - Replace with appropriate id
[self.saView setStrPanelMemberID:@"3479669"];

// To set Survey ID  - Replace with appropriate id
[self.saView setStrSurveyID:@"3135804"];

// To set ApiKey  - Replace with appropriate apiKey mapped to your panel
[self.saView setStrPanelApiKey:@"14db1eb8-8324-43db-85ef-4ab44f770b25"];
```

4. **Download and start the survey module into your app.**

```
// To download and Start specific Survey
[self.saView loadAndStartSurvey];
// To download and launch Survey list
[self.saView loadSurveys];
```

Mandatory setup

Add below permissions in main application's "viewController" file

1. Define and create SASurveyListView object into your class
2. To set current controller for Root Observer of SDK callback events in .m file
   `[self.saView setRootViewController:self];`
3. To set Panel Nember - Replace with appropriate id
   `[self.saView setStrPanelMemberID:@"2591355"];`

4. To set Panel API key - Replace with appropriate Key
   [self.saView setStrPanelApiKey:@"bddaf89e-eb17-4daa-983e-ebdb6cc92439"];

Credentials

Your mobile integration requires different PanelMemberID and PanelApiKey values for each environment: Live and Test (Sandbox).

Live

Your need to generate this panel key at server end in your account for specific panel in surveyswipe module.
For live mode you need to use generate following details from your SA enterprise account..

1. On Panel screen, you need create panel API key for iOS platform. This generated key will use in client app for setting PanelApiKey property.
   Screenshot: Panel API key
   Note: It is mandatory field.
2. On Panelist screen, you need find the PanelMemberID of user account. This ID will use in client app for setting PanelMemberID property.
   Screenshot: Panel Member ID
   Note: It is mandatory field.
3. On Survey screen, you need find the SurveyID of specific survey if you want lauch specific survey into client app. This ID will use in client app for setting SurveyID property.
   Screenshot: Survey ID
   Note: It is optional field.

Sandbox

For sandbox mode you need to use following provided details.

panel_api_key: f927c3e6-2344-4f6c-84ac-ae3523d74065
panel_member_id: 2108414
survey_id:

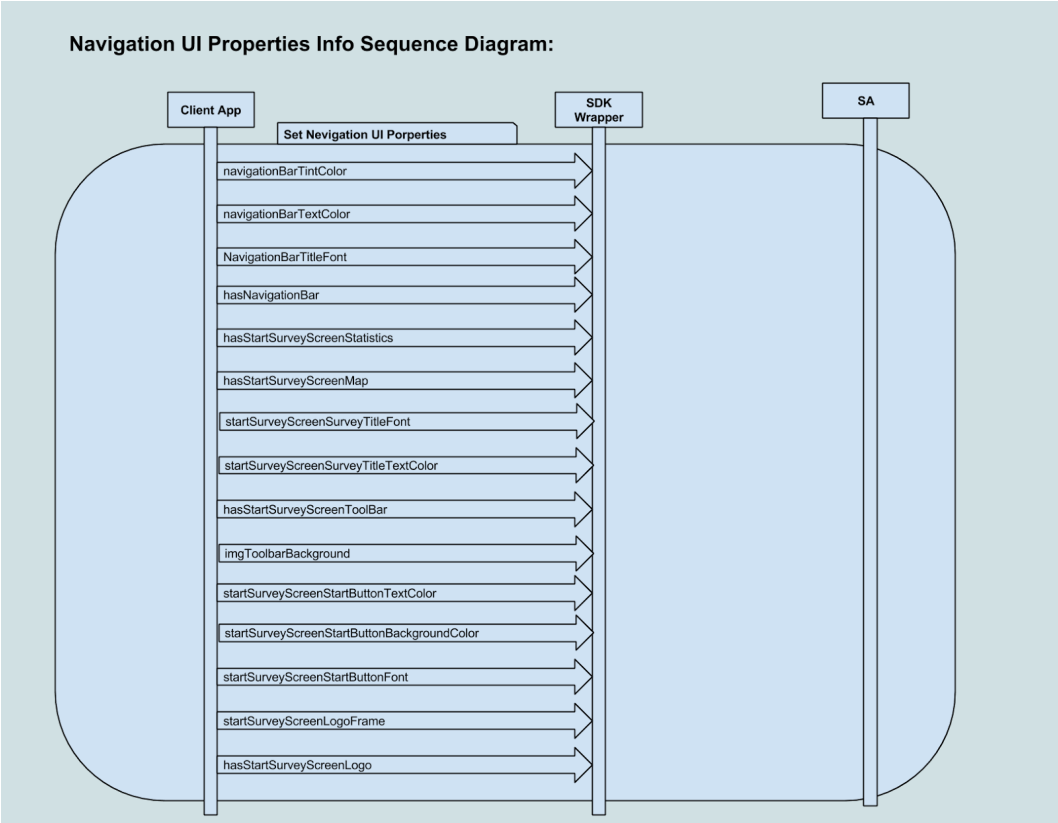
Optional setup

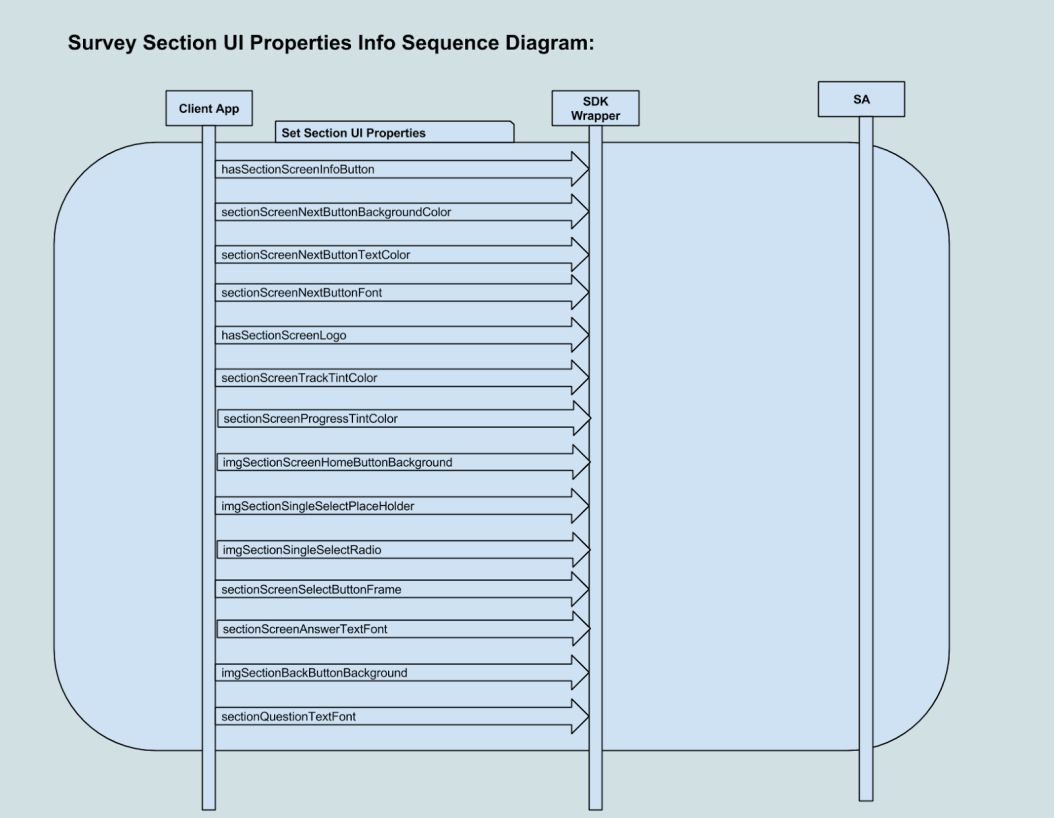
iOS Survey SDK is provide options to customize UI as per requirement. SDK expose some classes and methods which is used to set UI as per requirement. like you want to modify your survey listing(change font, size, colour etc).

for eg: you want to changes survey title font then you can set like

```
// To set NavigationBar Title Font style
self.saView.NavigationBarTitleFont = [UIFont systemFontOfSize:13];

// To set Main SurveyList screen title
//   self.saView.navigationBarTitle = @"Demo App";
```





International Support

Localizations

The SDK has built-in translations for only English languages.

Testing

During development and testing, set the demo panel key and demo survey id to API. See the header files for more information.

Documentation

- These docs in the SDK, which include an overview of usage, step-by-step integration instructions, and sample code.
- The sample app included in this SDK.
- Header files are thoroughly documented; refer to them as needed for extra details about any given property or parameter.

Usability

User interface appearance and behavior is set within the library itself. For the sake of usability and user experience consistency, apps should not adjust appearance properties or attempt to modify the SDK's behavior beyond the documented methods in the provided headers. Specifically, if you are using UIAppearance to modify the appearance of any UI elements in your app, you should reverse those changes prior to presenting our viewcontroller, and set them again after dismissing the viewcontroller.

Next Steps

Depending on your use case, you can now use following API calls load survey:

- Single Survey mode
- Survey List mode