iOS Widget Screenshot Automation

10 November 2022

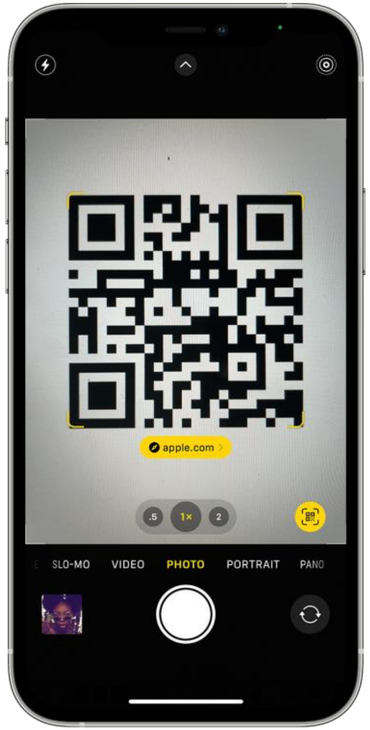
Rishab Sukumar
Senior Software Engineer

Monika Gorkani
Senior Staff Software Engineer

Alexander Botkin
Senior Mobile Engineering Manager

Watch that bottom right corner!

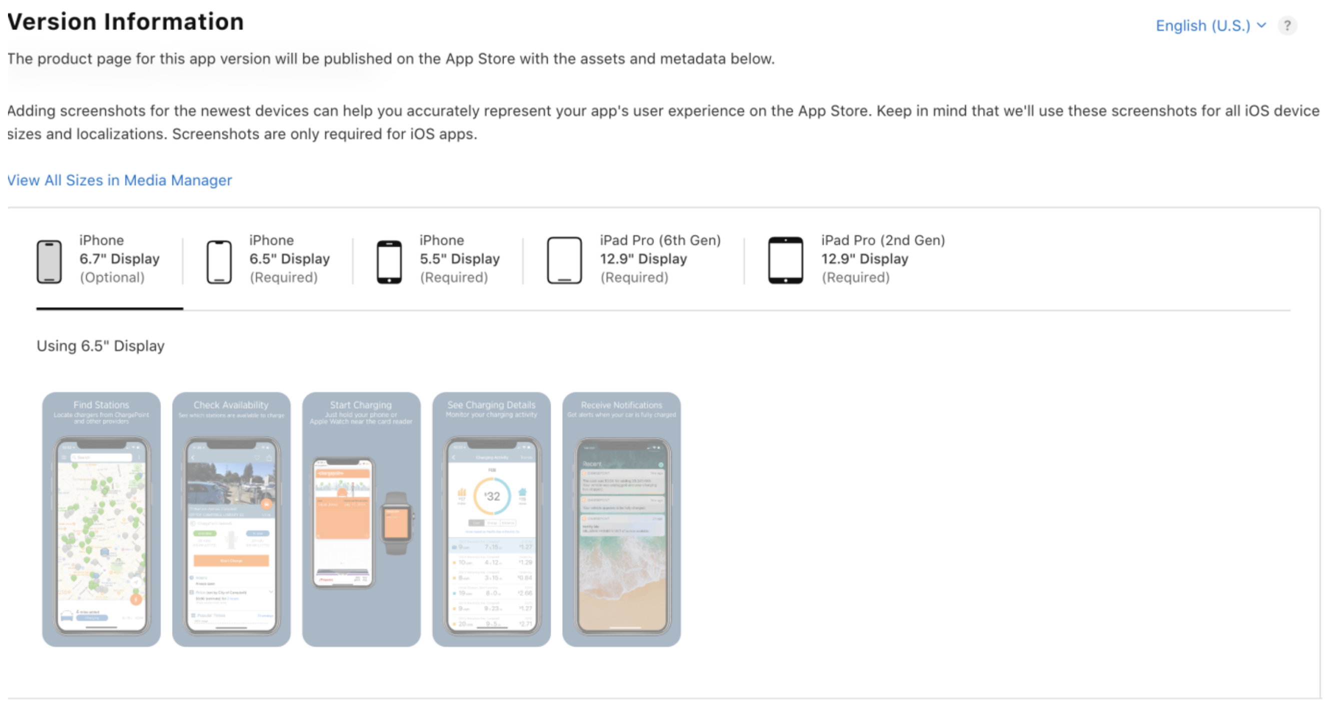
Open the iOS Camera app, point at the QR code, & tap the banner if you want to learn more!



Why Automate Screenshots?

App Store Connect Requirements

+ Stay ahead of ever-changing Apple screenshot requirements



Version Information

English (U.S.) ?

The product page for this app version will be published on the App Store with the assets and metadata below.

Adding screenshots for the newest devices can help you accurately represent your app's user experience on the App Store. Keep in mind that we'll use these screenshots for all iOS device sizes and localizations. Screenshots are only required for iOS apps.

View All Sizes in Media Manager

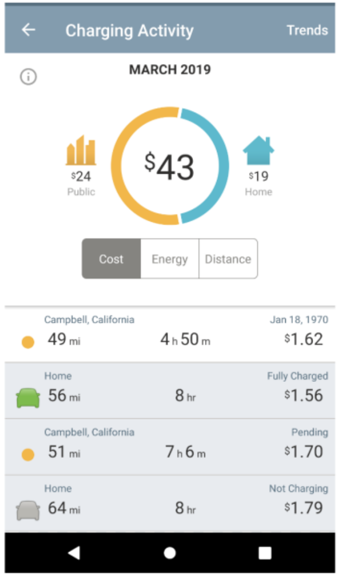
iPhone 6.7" Display (Optional)	iPhone 6.5" Display (Required)	iPhone 5.5" Display (Required)	iPad Pro (6th Gen) 12.9" Display (Required)	iPad Pro (2nd Gen) 12.9" Display (Required)

Using 6.5" Display

Find Stations
Locate chargers from ChargePoint and other providers
Check Availability
See which stations are available to charge
Start Charging
Just hold your phone or Apple Watch near the card reader
See Charging Details
Monitor your charging activity
Receive Notifications
Get alerts when your car is fully charged

Visual Baseline for Review & Comparison

Screenshot Name:"CPTAutomation_en_chargingActivity_step1.png"



Charging Activity   Trends

MARCH 2019

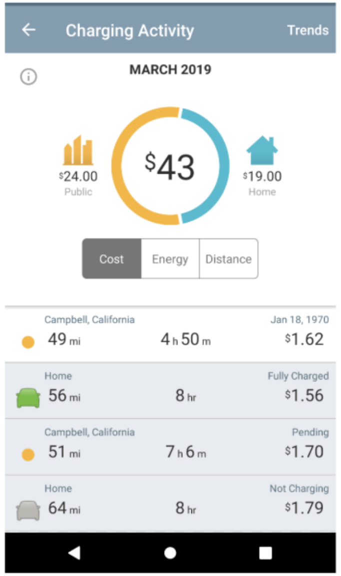
$43 — $24 Public — $19 Home

Cost | Energy | Distance

Campbell, California		Jan 18, 1970
49 mi	4 h 50 m	$1.62
Home		Fully Charged
56 mi	8 hr	$1.56
Campbell, California		Pending
51 mi	7 h 6 m	$1.70
Home		Not Charging
64 mi	8 hr	$1.79

Master

Master Screenshot taken On Device: "Pixel2",Theme "light" & Language: "en"



Charging Activity   Trends

MARCH 2019

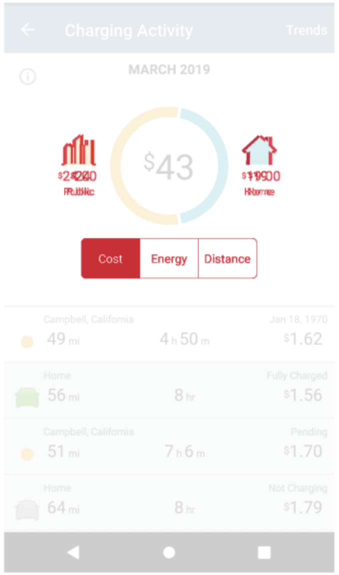
$43 — $24.00 Public — $19.00 Home

Cost | Energy | Distance

Campbell, California		Jan 18, 1970
49 mi	4 h 50 m	$1.62
Home		Fully Charged
56 mi	8 hr	$1.56
Campbell, California		Pending
51 mi	7 h 6 m	$1.70
Home		Not Charging
64 mi	8 hr	$1.79

Current(develop)

Branch Screenshot taken On Device: "Pixel2",Theme "light" & Language: "en"



Charging Activity   Trends

MARCH 2019

$43

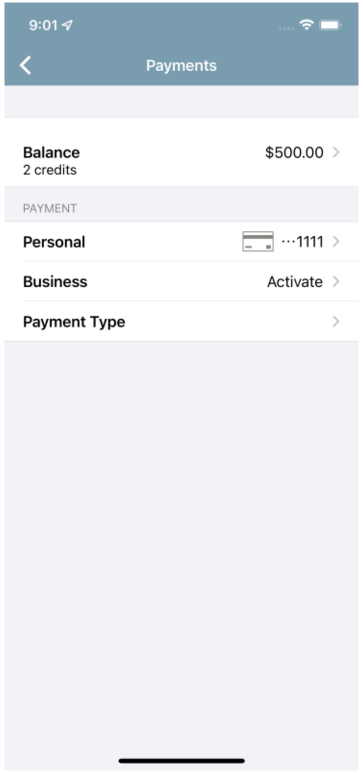
Cost | Energy | Distance

Campbell, California		Jan 18, 1970
49 mi	4 h 50 m	$1.62
Home		Fully Charged
56 mi	8 hr	$1.56
Campbell, California		Pending
51 mi	7 h 6 m	$1.70
Home		Not Charging
64 mi	8 hr	$1.79

Difference

Result After Comparison On Device: "Pixel2",Theme: "light" & Language: "en"

Guide Development of UI Changes


9:01

Payments

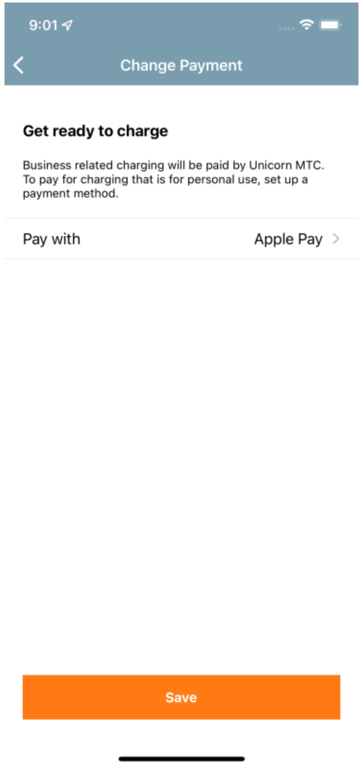
Balance $500.00
2 credits

PAYMENT

Personal ···1111

Business Activate

Payment Type


9:01

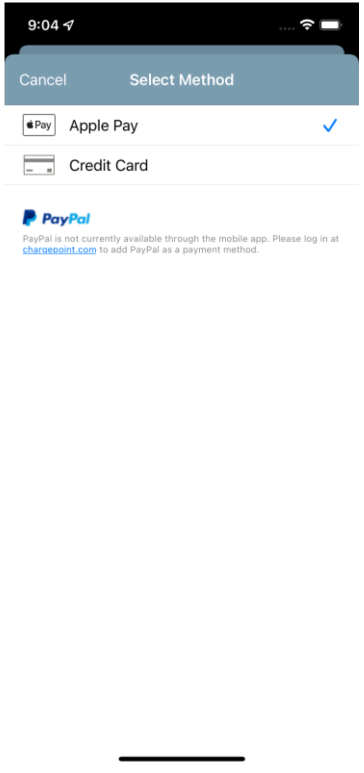
Change Payment

Get ready to charge

Business related charging will be paid by Unicorn MTC. To pay for charging that is for personal use, set up a payment method.

Pay with Apple Pay

Save


9:04

Cancel Select Method

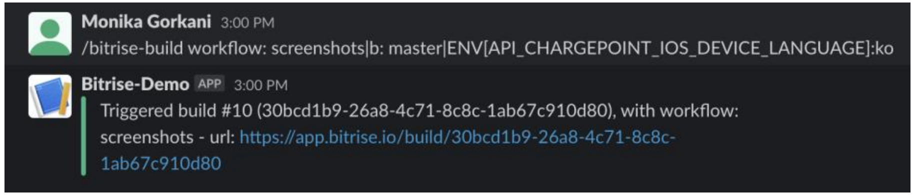
Apple Pay ✓

Credit Card

PayPal

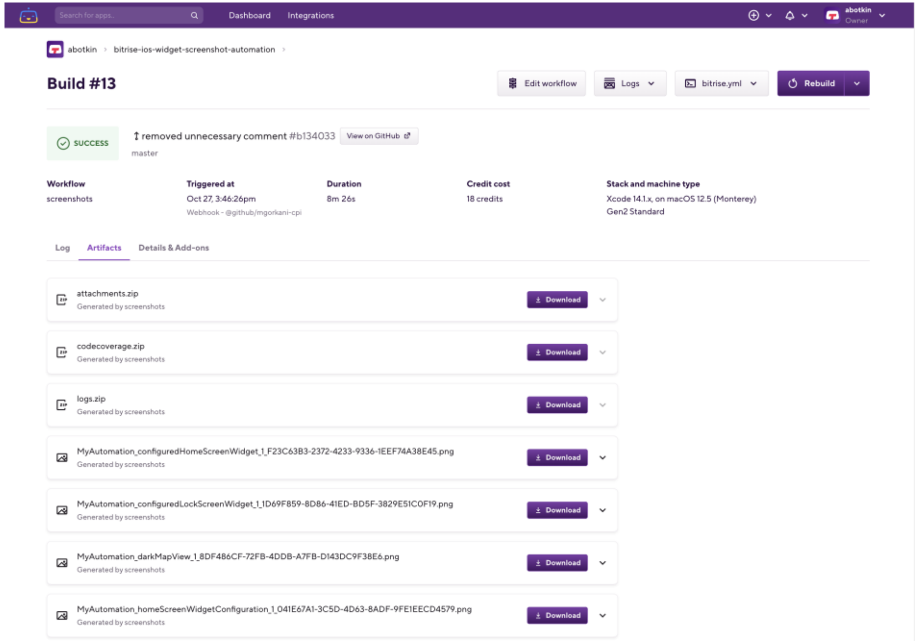
PayPal is not currently available through the mobile app. Please log in at chargepoint.com to add PayPal as a payment method.

Empower Developers To Move Faster



Monika Gorkani 3:00 PM
/bitrise-build workflow: screenshots|b: master|ENV[API_CHARGEPOINT_IOS_DEVICE_LANGUAGE]:ko

Bitrise-Demo APP 3:00 PM
Triggered build #10 (30bcd1b9-26a8-4c71-8c8c-1ab67c910d80), with workflow: screenshots - url: https://app.bitrise.io/build/30bcd1b9-26a8-4c71-8c8c-1ab67c910d80



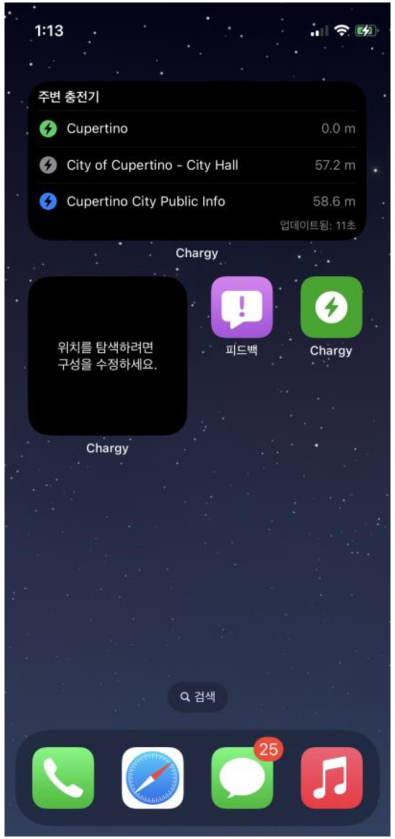
abotkin › bitrise-ios-widget-screenshot-automation ›

Build #13

SUCCESS ↥ removed unnecessary comment #b134033
master

Workflow	Triggered at	Duration	Credit cost	Stack and machine type
screenshots	Oct 27, 3:46:26pm Webhook - @github/mgorkani-cpi	8m 26s	18 credits	Xcode 14.1.x, on macOS 12.5 (Monterey) Gen2 Standard

attachments.zip Generated by screenshots	Download
codecoverage.zip Generated by screenshots	Download
logs.zip Generated by screenshots	Download
MyAutomation_configuredHomeScreenWidget_1_F23C63B3-2372-4233-9336-1EEF74A38E45.png Generated by screenshots	Download
MyAutomation_configuredLockScreenWidget_1_1D69F859-8D86-41ED-BD5F-3829E51C0F19.png Generated by screenshots	Download
MyAutomation_darkMapView_1_8DF486CF-72FB-4DDB-A7FB-D143DC9F38E6.png Generated by screenshots	Download
MyAutomation_homeScreenWidgetConfiguration_1_041E67A1-3C5D-4D63-8ADF-9FE1EECD4579.png Generated by screenshots	Download



주변 충전기

Cupertino	0.0 m
City of Cupertino - City Hall	57.2 m
Cupertino City Public Info	58.6 m

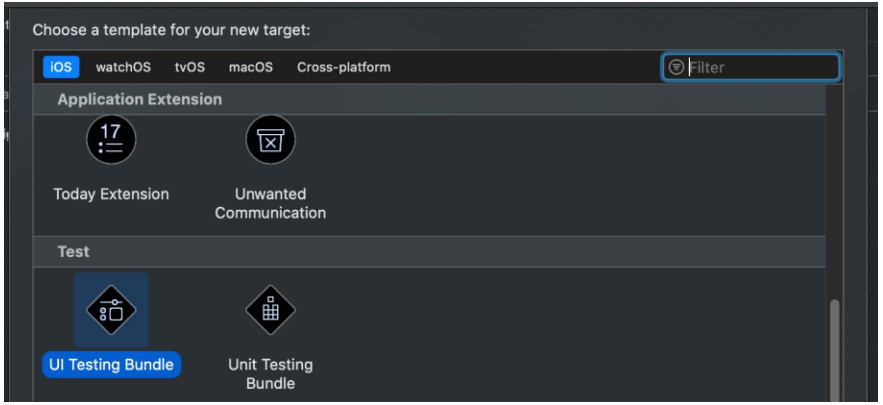
업데이트됨: 11초

위치를 탐색하려면 구성을 수정하세요.

How Do We Automate Screenshots?

How do we create screenshots?

+ UI Testing Bundle target
+ Xcode generates a <YourAppName>UITests.swift file with an XCTestCase subclass
+ We need to add test functions that save attachments onto the XCTestCase



How do we create screenshots?

```
func testMainViewScreenshot() throws {
    // UI tests must launch the application that they test.
    let app = XCUIApplication()
    app.launch()

    // Let's ensure the view has appeared by using the accessibility identifier
    // we set up in the storyboard
    let darkMapVCView = app.otherElements["Dark Map View"];
    XCTAssertTrue(darkMapVCView.waitForExistence(timeout: 3))

    // Now let's get a screenshot & save it to the xcresult as an attachment
    let screenshot = XCUIScreen.main.screenshot()

    let attachment = XCTAttachment(screenshot: screenshot)
    attachment.name = "MyAutomation_darkMapView"
    attachment.lifetime = .keepAlways

    self.add(attachment)
}
```

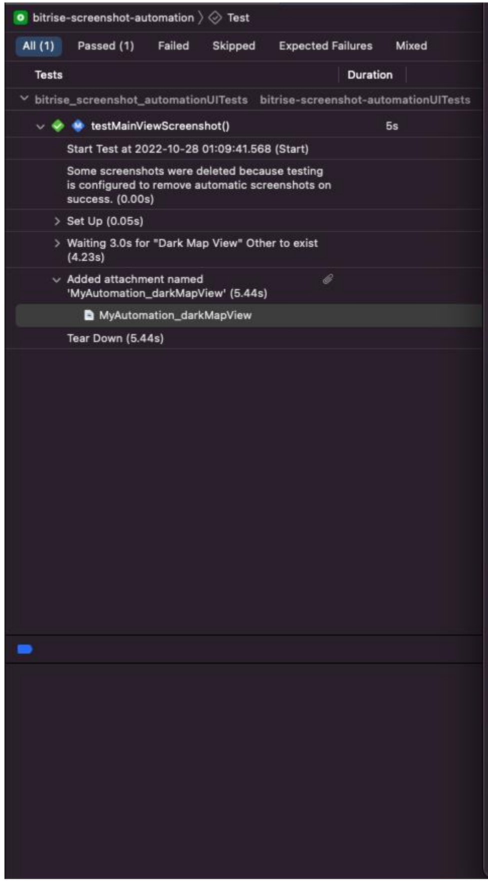
How do we create screenshots?

```swift
//
//  XCTestCase+Screenshots.swift
//  bitrise-screenshot-automationUITests
//

import XCTest

extension XCTestCase {
    func saveScreenshot(_ name: String) {
        let screenshot = XCUIScreen.main.screenshot()

        let attachment = XCTAttachment(screenshot: screenshot)
        attachment.name = name
        attachment.lifetime = .keepAlways

        self.add(attachment)
    }
}
```

How do we create screenshots?

```swift
func testMainViewScreenshot() throws {
    // UI tests must launch the application that they test.
    let app = XCUIApplication()
    app.launch()

    // Let's ensure the view has appeared by using the accessibility identifier
    // we set up in the storyboard
    let darkMapVCView = app.otherElements["Dark Map View"];
    XCTAssertTrue(darkMapVCView.waitForExistence(timeout: 3))

    // Now let's get a screenshot & save it to the xcresult as an attachment
    self.saveScreenshot("MyAutomation_darkMapView")
}
```

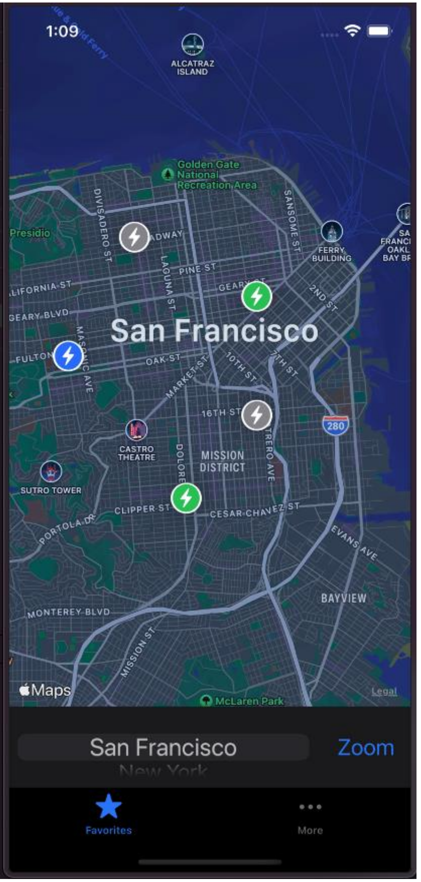
How do we create screenshots?



bitrise-screenshot-automation 〉 Test

All (1) | Passed (1) | Failed | Skipped | Expected Failures | Mixed

Tests	Duration
bitrise_screenshot_automationUITests bitrise-screenshot-automationUITests	
testMainViewScreenshot()	5s
Start Test at 2022-10-28 01:09:41.568 (Start)	
Some screenshots were deleted because testing is configured to remove automatic screenshots on success. (0.00s)	
Set Up (0.05s)	
Waiting 3.0s for "Dark Map View" Other to exist (4.23s)	
Added attachment named 'MyAutomation_darkMapView' (5.44s)	
MyAutomation_darkMapView	
Tear Down (5.44s)	



So...what is XCUIApplication doing?

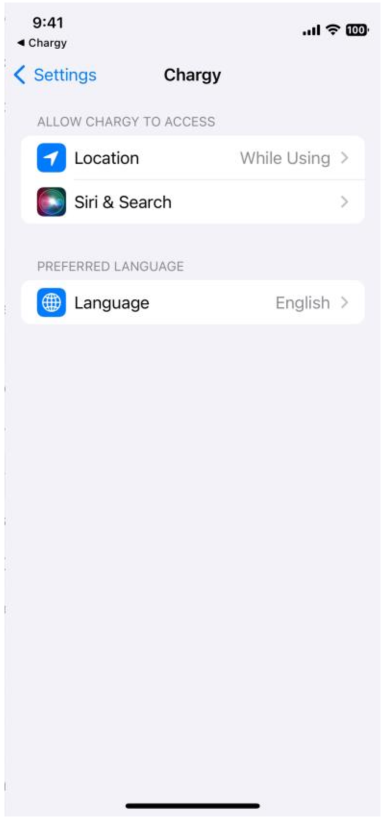
+ XCUIApplication is a proxy for a specified process that allows you to send events like launch, monitor, and terminate in a UITest
+ You can create a specific proxy for an app and interact with its elements by initializing XCUIApplication with the app's Bundle ID

```
let settings = XCUIApplication(bundleIdentifier: "com.apple.Preferences")
```

Settings

```
func testOpenAppSettings() throws {
    // Access the Settings app using its bundle identifier
    let settings = XCUIApplication(bundleIdentifier: "com.apple.Preferences")
    settings.activate()

    // Swipe until Chargy is visible
    settings.swipe(direction: .Up, numSwipes: 4)

    // Navigate to our app's settings
    settings.cells["Chargy"].tap()

    // Save App Settings Screenshot
    self.saveScreenshot("MyAutomation_AppSettings")
}
```

Settings


9:41
◀ Chargy

Settings Chargy

ALLOW CHARGY TO ACCESS

Location While Using

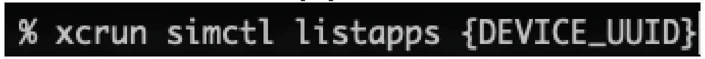
Siri & Search

PREFERRED LANGUAGE

Language English

Processes, not applications

+ Useful App Bundle IDs
  - com.apple.Preferences (Settings)
  - com.apple.mobileSafari (Safari)
  - com.apple.shortcuts (Shortcuts)
+ You can find Bundle IDs for installed apps on a device/simulator with the following command:



```
% xcrun simctl listapps {DEVICE_UUID}
```

+ Other useful processes
  - com.apple.springboard
  - com.apple.Posterboard
  - com.apple.WorkflowUI.WidgetConfigurationExtension

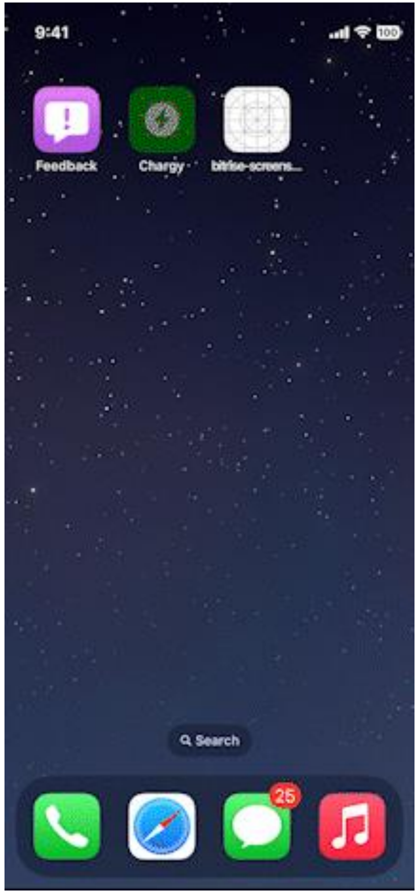
Processes, not applications

```
let springboard = XCUIApplication(bundleIdentifier: "com.apple.springboard")
```

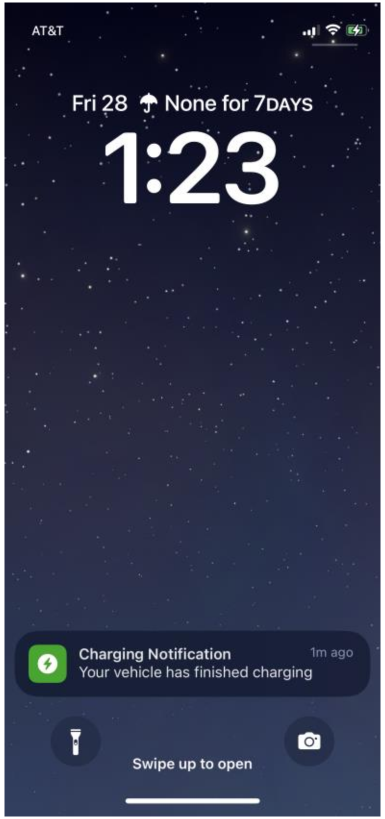
+ Springboard is the process that manages the iOS Home Screen
+ Creating a proxy for Springboard using XCUIApplication allows developers to write more powerful UI tests
  - Open Notification Center
  - Access Spotlight Search
  - Navigate to Today View
  - Add Widgets

Notification Center

```
func testNotificationCenter() {
    let app = XCUIApplication()
    let button = app.buttons["ZoomButton"]
    XCTAssertTrue(button.exists)
    button.tap()
    handleAlerts(app: app)

    // Springboard allows us to interact with the home screen
    let springboard = XCUIApplication(bundleIdentifier:
"com.apple.springboard")
    springboard.activate()

    // Swipe down from the top to get to the lock screen to view
notifications
    let coord1 = springboard.coordinate(withNormalizedOffset: CGVector(dx: 0,
dy: 0))
    let coord2 = springboard.coordinate(withNormalizedOffset: CGVector(dx: 0,
dy: 2))
    coord1.press(forDuration: 0.1, thenDragTo: coord2)

    self.saveScreenshot("MyAutomation_Notification")
}
```



Notification Center

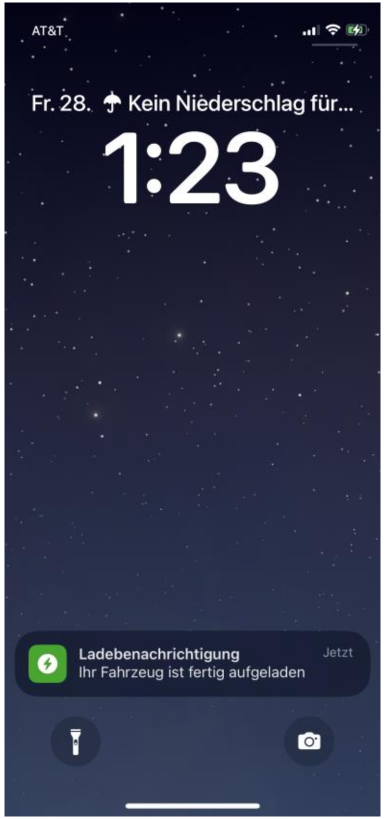

AT&T

Fri 28 None for 7DAYS

1:23

Charging Notification 1m ago
Your vehicle has finished charging

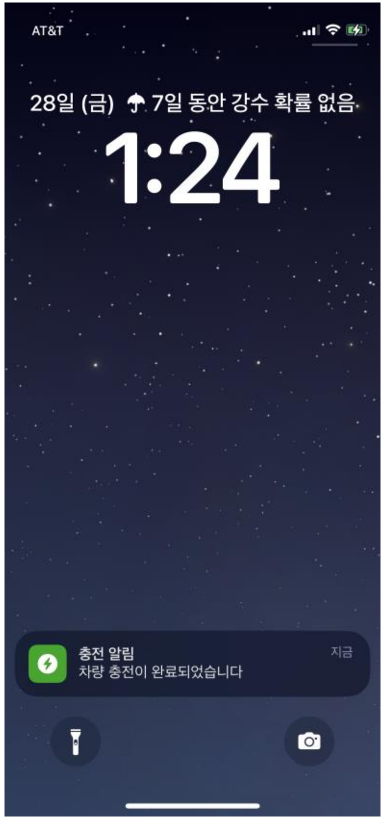
Swipe up to open


AT&T

Fr. 28. Kein Niederschlag für...

1:23

Ladebenachrichtigung Jetzt
Ihr Fahrzeug ist fertig aufgeladen


AT&T

28일 (금) 7일 동안 강수 확률 없음

1:24

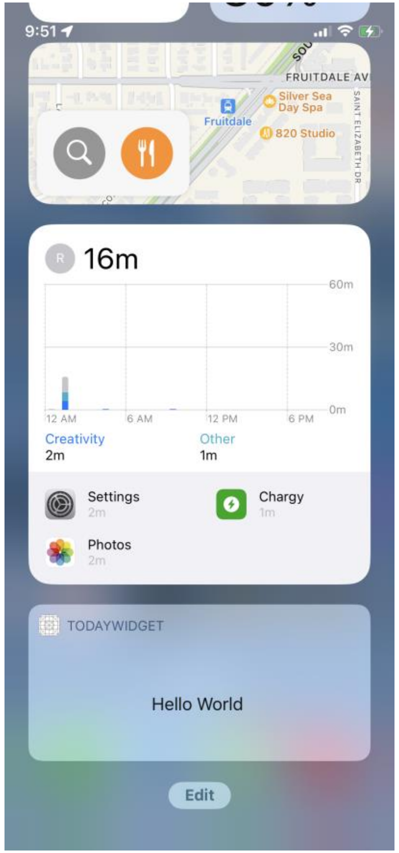
충전 알림 지금
차량 충전이 완료되었습니다

Today View

```
func testTodayWidgetScreenshot() throws {
    // Springboard allows you to navigate the home screen
    let springboard = XCUIApplication(bundleIdentifier: "com.apple.springboard")
    springboard.activate()

    // Swipe until we get to the Today View
    springboard.swipe(direction: .Right, numSwipes: 2)

    // Swipe until edit button is visible. Using XCUIApplication extension method here.
    springboard.swipe(direction: .Up, numSwipes: 2)

    let editButton = springboard.buttons.firstMatch
    XCTAssertTrue(editButton.waitForExistence(timeout: 3))
    editButton.tap()

    // Swipe until Customize button is visible
    springboard.swipe(direction: .Up, numSwipes: 3)

    // Using XCUIElementQuery extension to access second last match
    var customizeButton = springboard.buttons.secondLastMatch

    customizeButton.tap()

    // Find and tap to add the TodayWidget
    let widgetNamePredicate = NSPredicate(format: "label CONTAINS[c] 'TodayWidget'")
    let addWidgetCells = springboard.cells.matching(widgetNamePredicate)
    addWidgetCells.buttons.firstMatch.tap()

    let doneButton = springboard.navigationBars.buttons.secondMatch
    doneButton.tap()

    springboard.swipe(direction: .Up, numSwipes: 1)
    self.saveScreenshot("MyAutomation_todayWidget")
}
```

9:41
Feedback
Installing...
Search
26

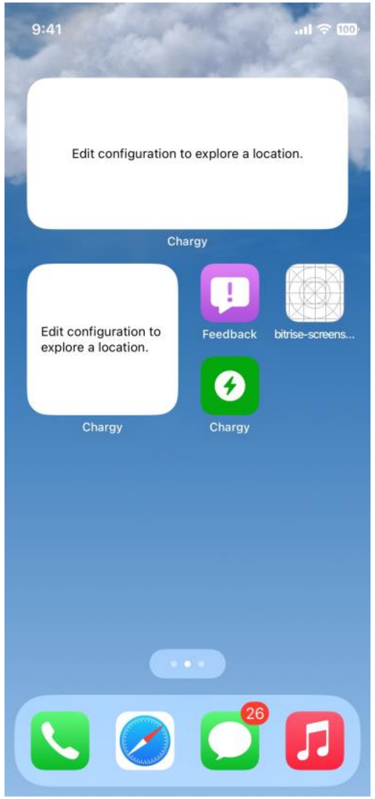
Today View


9:51
FRUITDALE AV
Silver Sea Day Spa
Fruitdale
820 Studio
SAINT ELIZABETH DR
16m
60m
30m
0m
12 AM
6 AM
12 PM
6 PM
Creativity
2m
Other
1m
Settings
2m
Chargy
1m
Photos
2m
TODAYWIDGET
Hello World
Edit

Home Screen Widgets

```swift
func addHomeScreenWidget(isMediumSize: Bool) throws {
    // The Springboard application allows us to interact with the home screen
    let springboard = XCUIApplication(bundleIdentifier: "com.apple.springboard")
    springboard.activate()

    // Press and hold to edit home screen
    springboard.press(forDuration: 3)
    // Tap on the add widgets button
    springboard.buttons.firstMatch.tap()
    // Find search field, search for app, and tap to add widgets
    springboard.searchFields.firstMatch.tap()
    springboard.typeText("Chargy")
    springboard.collectionViews.cells["Chargy"].tap()

    // If medium size, swipe to medium size widget
    if (isMediumSize) {
        springboard.swipe(direction: .Left, numSwipes: 1)
    }

    // Tap "Add Widget" button and then tap "Done" button
    springboard.buttons.thirdLastMatch.tap()
    springboard.buttons.secondMatch.tap()

    // Save Home Screen Widget Screenshot
    self.saveScreenshot("MyAutomation_homeScreenWidget")
}
```

Home Screen Widgets

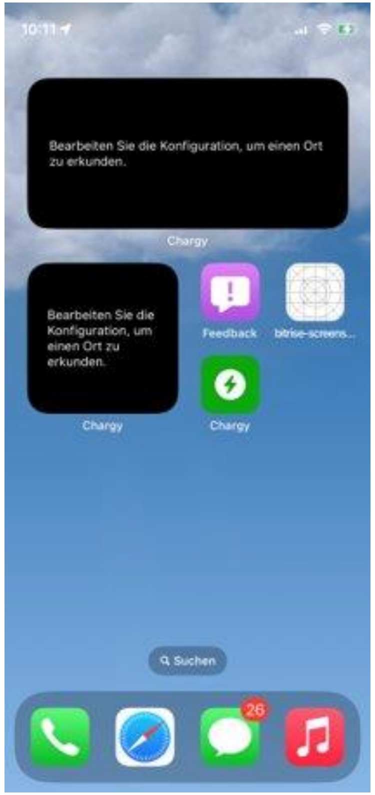


Edit configuration to explore a location.

Chargy

Edit configuration to explore a location.

Chargy



Bearbeiten Sie die Konfiguration, um einen Ort zu erkunden.

Chargy

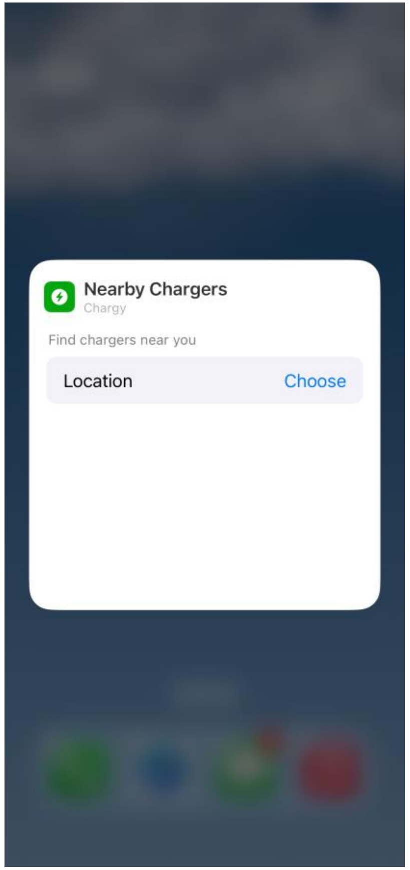
Bearbeiten Sie die Konfiguration, um einen Ort zu erkunden.

Chargy

Editing Home Screen Widgets

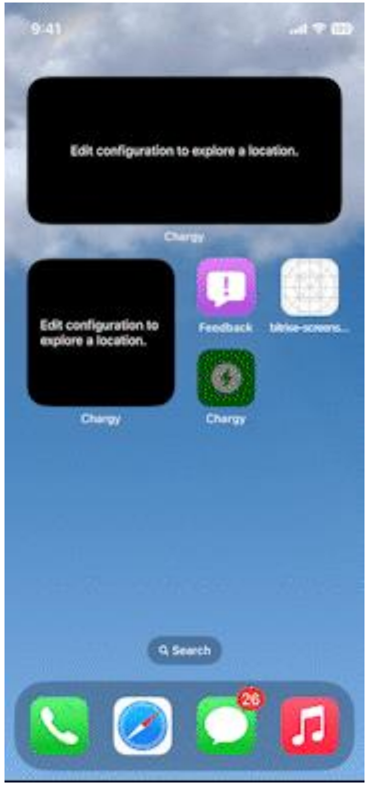
```
let widgetConfig = XCUIApplication(bundleIdentifier:
"com.apple.WorkflowUI.WidgetConfigurationExtension")
```

+ Elements of the Widget Configuration are not visible to Springboard.
+ The Process that handles Widget Configuration: WorkflowUI.WidgetConfigurationExtension
+ The WorkflowUI private framework contains other extensions as well.
  - AddShortcutExtension
  - CatalystContentExtension
  - FocusConfigurationExtension

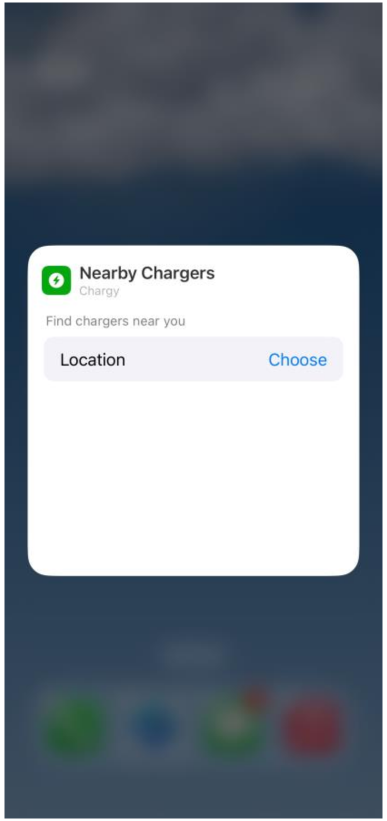


Editing Home Screen Widgets

```
func editHomeScreenWidget() {
    let springboard = XCUIApplication(bundleIdentifier: "com.apple.springboard")
    springboard.press(forDuration: 3)
    springboard.icons.matching(identifier: "Chargy").firstMatch.tap()

    // Use WidgetConfigurationExtension XCUIApplication to navigate the home screen
widget configuration
    let widgetConfig = XCUIApplication(bundleIdentifier:
"com.apple.WorkflowUI.WidgetConfigurationExtension")

    self.saveScreenshot("MyAutomation_homeScreenWidgetConfiguration")

    widgetConfig.buttons.firstMatch.tap()
    widgetConfig.searchFields.firstMatch.tap()
    widgetConfig.typeText("Cupertino\n")
    widgetConfig.cells.firstMatch.tap()
    widgetConfig.buttons.secondMatch.tap()

    // Dismiss configuration
    springboard.coordinate(withNormalizedOffset: CGVector(dx: 0, dy: 0)).tap()

    springboard.buttons.secondMatch.tap()

    self.saveScreenshot("MyAutomation_configuredHomeScreenWidget")
}
```


9:41
Edit configuration to explore a location.
Chargy
Edit configuration to explore a location.
Feedback
bitrise-screens...
Chargy
Chargy
Search
26

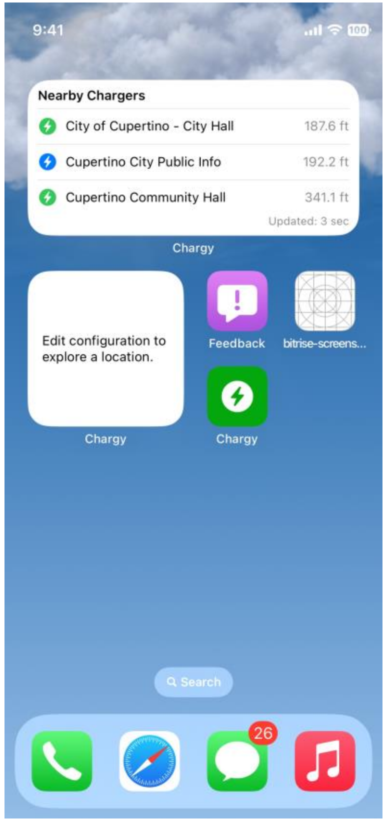
Editing Home Screen Widgets



Nearby Chargers
Chargy

Find chargers near you

Location Choose



9:41

Nearby Chargers

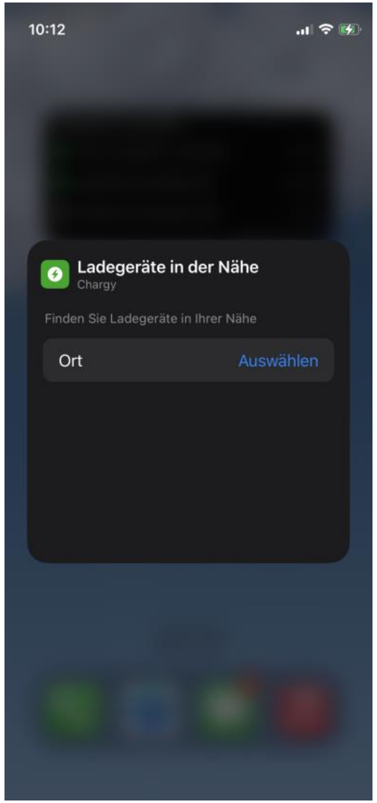
City of Cupertino - City Hall	187.6 ft
Cupertino City Public Info	192.2 ft
Cupertino Community Hall	341.1 ft

Updated: 3 sec

Chargy

Edit configuration to explore a location.

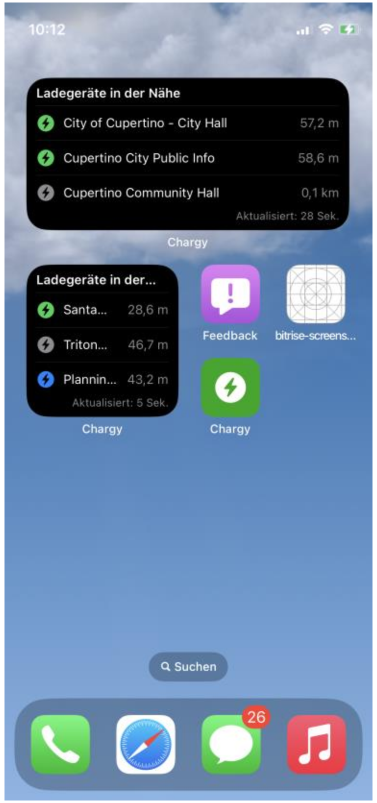
Editing Home Screen Widgets


10:12

Ladegeräte in der Nähe
Chargy

Finden Sie Ladegeräte in Ihrer Nähe

Ort	Auswählen


10:12

Ladegeräte in der Nähe

City of Cupertino - City Hall	57,2 m
Cupertino City Public Info	58,6 m
Cupertino Community Hall	0,1 km

Aktualisiert: 28 Sek.

Chargy

Ladegeräte in der...

Santa...	28,6 m
Triton...	46,7 m
Plannin...	43,2 m

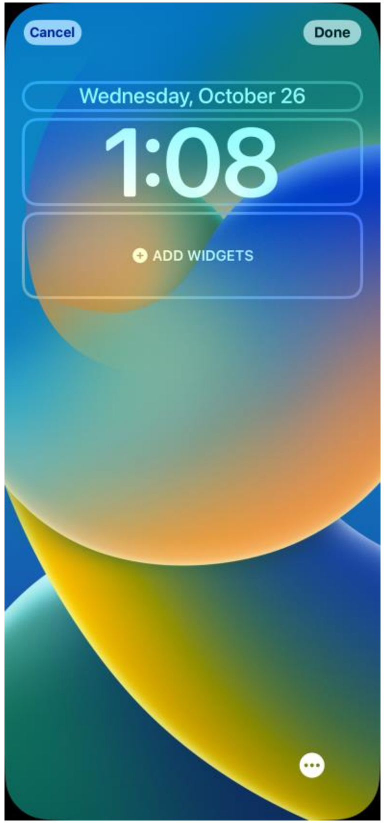
Aktualisiert: 5 Sek.

Chargy

Lock Screen Widgets

```
let posterboard = XCUIApplication(bundleIdentifier:
"com.apple.Posterboard")
```

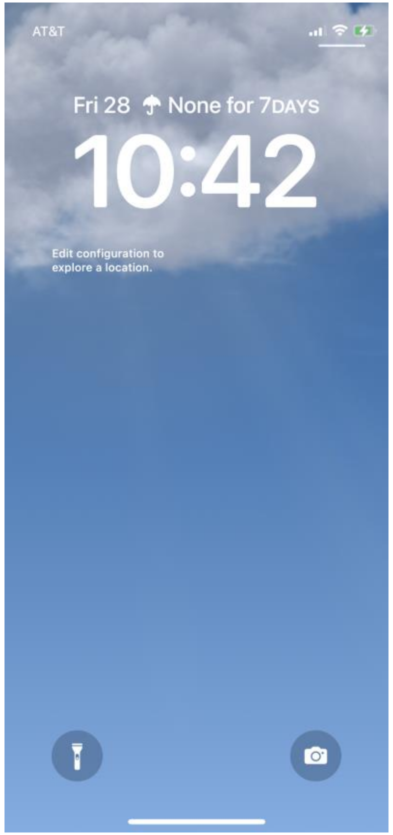
+ With iOS 16, the Lock Screen can now be customized
  – Font and Text Color
  – Wallpaper
  – Widgets
+ Lock Screen customization is handled by a new process called Posterboard



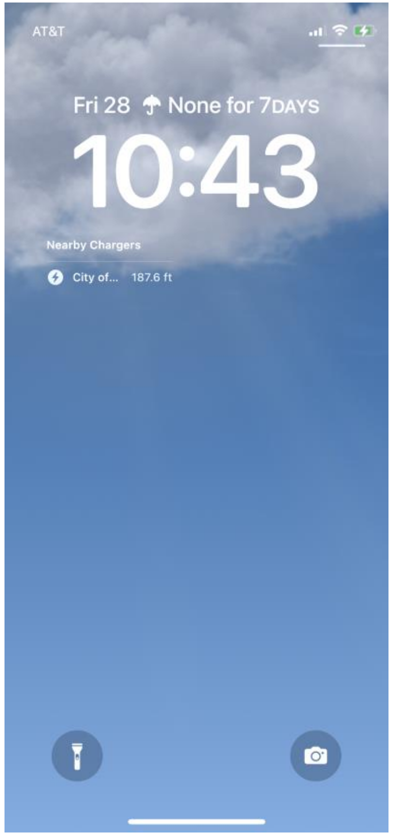
Lock Screen Widgets

```swift
func addLockScreenWidget() {
    let springboard = XCUIApplication(bundleIdentifier: "com.apple.springboard")
    springboard.activate()

    // Swipe down from the top to get to the lock screen
    let coord1 = springboard.coordinate(withNormalizedOffset: CGVector(dx: 0, dy: 0))
    let coord2 = springboard.coordinate(withNormalizedOffset: CGVector(dx: 0, dy: 2))
    coord1.press(forDuration: 0.1, thenDragTo: coord2)

    // Press and hold to edit lock screen
    springboard.press(forDuration: 3)

    // Tap Customize
    springboard.buttons.matching(identifier: "posterboard-customize-button").firstMatch.tap()

    // Tap to customize Lock Screen
    springboard.collectionViews["posterboard-collection-view"].cells.firstMatch.tap()

    // Use posterboard to navigate lock screen customization
    let posterboard = XCUIApplication(bundleIdentifier: "com.apple.PosterBoard")

    // Tap Add Widgets cell
    posterboard.buttons.lastMatch.tap()

    // Find and add widget
    posterboard.cells["Chargy"].tap()
    springboard.buttons.lastMatch.tap()

    // Dismiss lock screen editing views
    springboard.buttons.firstMatch.tap()
    springboard.buttons.firstMatch.tap()
    springboard.buttons.secondMatch.tap()
    springboard.tap()

    self.saveScreenshot("MyAutomation_lockScreenWidget")
}
```

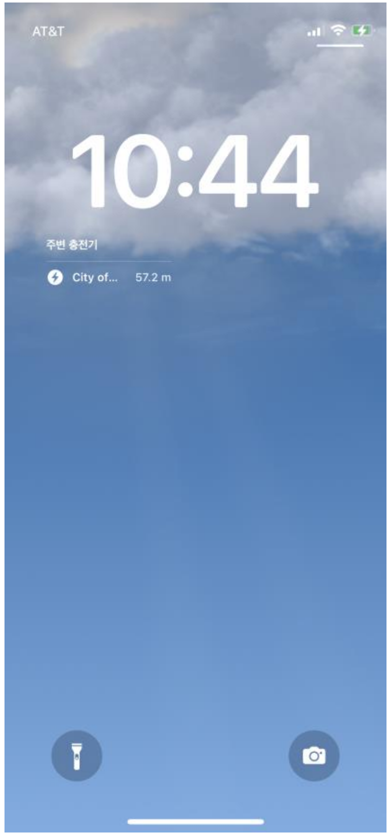
Lock Screen Widgets



Edit configuration to explore a location.



Nearby Chargers
City of... 187.6 ft



주변 충전기
City of... 57.2 m

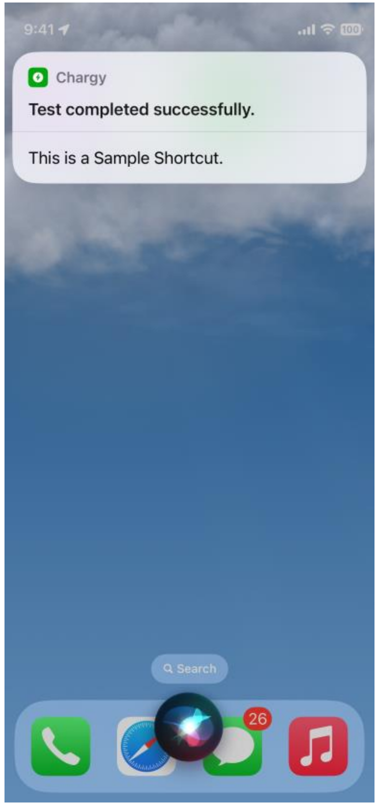
Siri Shortcuts

+ XCUIDevice is a proxy that is used to simulate physical buttons, device orientation, and Siri interaction
+ An XCUIDevice instance's *siriService* property represents the device's Siri interface
+ Using XCUISiriService
  - `func activate(voiceRecognitionText: String)`
    - Presents the Siri UI and accepts a string as if it were recognized speech (Uses the device's Siri input language setting)

Siri Shortcuts

```
func testSiriShortcut() throws {
    let springboard = XCUIApplication(bundleIdentifier:
"com.apple.springboard")
    springboard.activate()

    // Use XCUISiriService to pass text to Siri and invoke App Shortcut
    XCUIDevice.shared.siriService.activate(voiceRecognitionText: "Run
sample with Chargy")

    sleep(3)

    self.saveScreenshot("MyAutomation_sampleSiriShortcut")
}
```

Siri Shortcuts

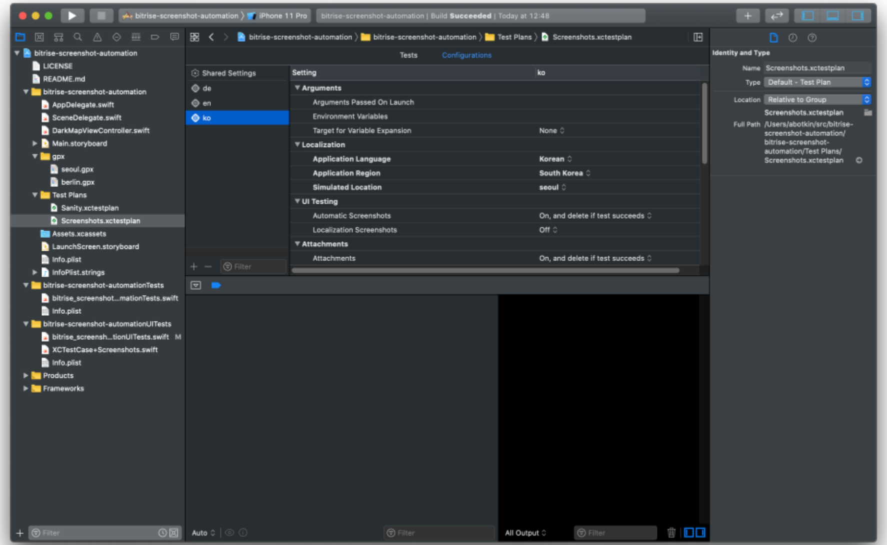


Test completed successfully.

This is a Sample Shortcut.

How do we create screenshots?

- How do we run the app in different languages & regions?
  - Make one test plan & add various configurations
- Create run configurations specific to an app language/region pairing
  - Does not affect device language & thus some system prompts
- Add GPX file in Xcode to simulate a custom GPS coordinate or route


bitrise-screenshot-automation 〉 iPhone 11 Pro
bitrise-screenshot-automation | Build Succeeded | Today at 12:48
bitrise-screenshot-automation 〉 bitrise-screenshot-automation 〉 Test Plans 〉 Screenshots.xctestplan
bitrise-screenshot-automation
LICENSE
README.md
bitrise-screenshot-automation
AppDelegate.swift
SceneDelegate.swift
DarkMapViewController.swift
Main.storyboard
gpx
seoul.gpx
berlin.gpx
Test Plans
Sanity.xctestplan
Screenshots.xctestplan
Assets.xcassets
LaunchScreen.storyboard
Info.plist
InfoPlist.strings
bitrise-screenshot-automationTests
bitrise_screenshot...mationTests.swift
Info.plist
bitrise-screenshot-automationUITests
bitrise_screensh...tionUITests.swift
XCTestCase+Screenshots.swift
Info.plist
Products
Frameworks
Tests
Configurations
Shared Settings
de
en
ko
Filter
Auto
All Output
Filter

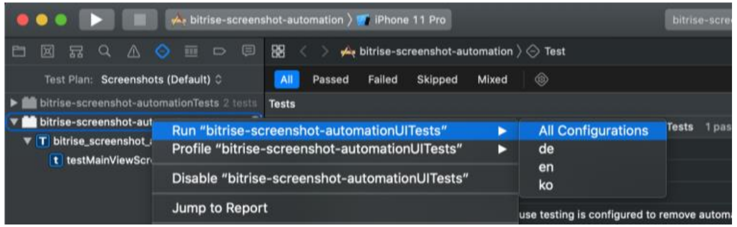
Setting	ko
Arguments	
Arguments Passed On Launch	
Environment Variables	
Target for Variable Expansion	None
Localization	
Application Language	**Korean**
Application Region	**South Korea**
Simulated Location	**seoul**
UI Testing	
Automatic Screenshots	On, and delete if test succeeds
Localization Screenshots	Off
Attachments	
Attachments	On, and delete if test succeeds

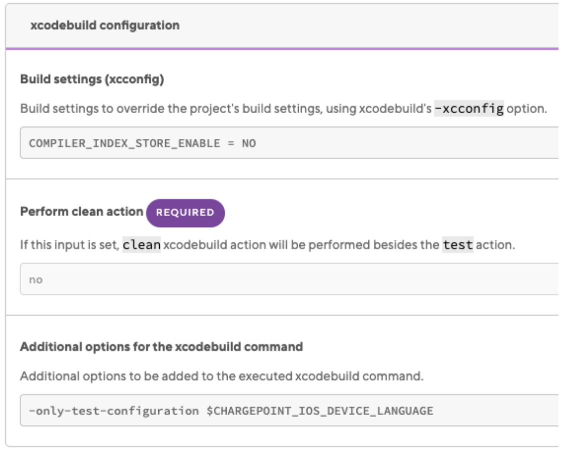
Identity and Type

Name	Screenshots.xctestplan
Type	Default - Test Plan
Location	Relative to Group
	Screenshots.xctestplan
Full Path	/Users/abotkin/src/bitrise-screenshot-automation/bitrise-screenshot-automation/Test Plans/Screenshots.xctestplan

How do we create screenshots?

- With test plans, can run all configurations & create one report (xcresult)
  - xcodebuild option: *-testPlan 'Screenshots'*
- Can also run one specific run configuration
  - xcodebuild option: *-only-test-configuration "de"*
- On Bitrise's Xcode Test step, these options have to be added in the 'xcodebuild configuration' section


bitrise-screenshot-automation 〉 iPhone 11 Pro
bitrise-scre
bitrise-screenshot-automation 〉 Test
Test Plan: Screenshots (Default)
All
Passed
Failed
Skipped
Mixed
bitrise-screenshot-automationTests 2 tests
Tests
bitrise-screenshot-aut
bitrise_screenshot_
testMainViewScr
Run "bitrise-screenshot-automationUITests"
Profile "bitrise-screenshot-automationUITests"
Disable "bitrise-screenshot-automationUITests"
Jump to Report
All Configurations
de
en
ko
Tests
1 pas
use testing is configured to remove autom



xcodebuild configuration

Build settings (xcconfig)

Build settings to override the project's build settings, using xcodebuild's `-xcconfig` option.

```
COMPILER_INDEX_STORE_ENABLE = NO
```

Perform clean action REQUIRED

If this input is set, `clean` xcodebuild action will be performed besides the `test` action.

```
no
```

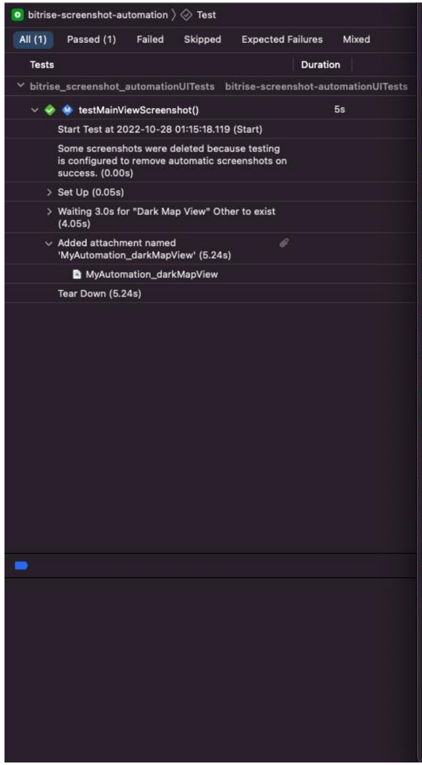
Additional options for the xcodebuild command

Additional options to be added to the executed xcodebuild command.

```
-only-test-configuration $CHARGEPOINT_IOS_DEVICE_LANGUAGE
```

How do we create screenshots?

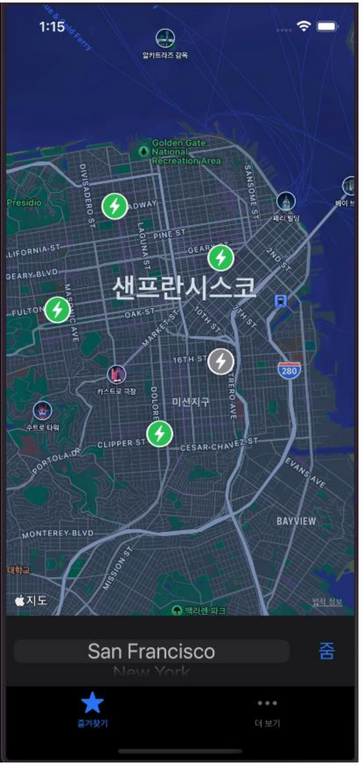
+ We have the report (xcresult), but how do we get all the screenshots out at once?



bitrise-screenshot-automation 〉 Test

All (1) | Passed (1) | Failed | Skipped | Expected Failures | Mixed

Tests	Duration
bitrise_screenshot_automationUITests bitrise-screenshot-automationUITests	
testMainViewScreenshot()	5s
Start Test at 2022-10-28 01:15:18.119 (Start)	
Some screenshots were deleted because testing is configured to remove automatic screenshots on success. (0.00s)	
Set Up (0.05s)	
Waiting 3.0s for "Dark Map View" Other to exist (4.05s)	
Added attachment named 'MyAutomation_darkMapView' (5.24s)	
MyAutomation_darkMapView	
Tear Down (5.24s)	

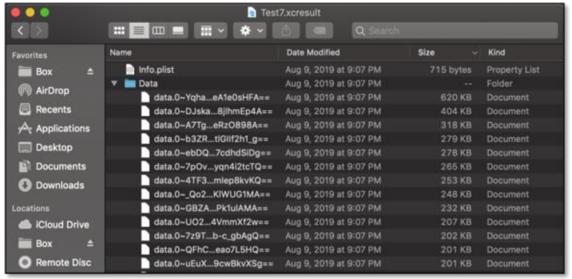


chargepoint®

Extracting Screenshots

xcresult & xcresulttool

+ *xcresulttool* is included in Xcode 11+ command line tools to do inspection & extraction of xcresult
  + Powerful tool
  + Not a one-line extractor



Test7.xcresult

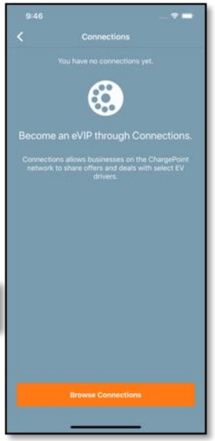
Name	Date Modified	Size	Kind
Info.plist	Aug 9, 2019 at 9:07 PM	715 bytes	Property List
Data	Aug 9, 2019 at 9:07 PM	--	Folder
data.0~Yqha...eA1e0sHFA==	Aug 9, 2019 at 9:07 PM	620 KB	Document
data.0~DJska...8jlhmEp4A==	Aug 9, 2019 at 9:07 PM	404 KB	Document
data.0~A7Tg...eRzO898A==	Aug 9, 2019 at 9:07 PM	318 KB	Document
data.0~b3ZR...tlGllf2h1_g==	Aug 9, 2019 at 9:07 PM	279 KB	Document
data.0~ebDQ...7cdhdSiDg==	Aug 9, 2019 at 9:07 PM	278 KB	Document
data.0~7pOv...yqn4i2tcTQ==	Aug 9, 2019 at 9:07 PM	265 KB	Document
data.0~4TF3...mlep8kvKQ==	Aug 9, 2019 at 9:07 PM	253 KB	Document
data.0~_Qo2...KIWUG1MA==	Aug 9, 2019 at 9:07 PM	248 KB	Document
data.0~GBZA...Pk1ulAMA==	Aug 9, 2019 at 9:07 PM	232 KB	Document
data.0~UO2...4VmmXf2w==	Aug 9, 2019 at 9:07 PM	207 KB	Document
data.0~7z9T...b-c_gbAgQ==	Aug 9, 2019 at 9:07 PM	202 KB	Document
data.0~QFhC...eao7L5HQ==	Aug 9, 2019 at 9:07 PM	201 KB	Document
data.0~uEuX...9cwBkvXSg==	Aug 9, 2019 at 9:07 PM	201 KB	Document

```
$ xcrun xcresulttool get --path $xcresultpath --format json
```

```
$ xcrun xcresulttool get --path $xcresultpath --format json --id 0~_6FLLXVxqNZFWTecoWAKeruyb1Jz8RMTQhv1WOY6rrDKD9fcctzLS71METqymkBgT27dA5y8aPQtO7QGqf22Aw==
```

```
$ xcrun xcresulttool get --path $xcresultpath --format json --id 0~kQ1eE6SCOHOqHvf_H1RgkahpjprRs3YG7BKlO_Twqzv4Cu3r7mbK3WSOSudizsrE7_4I6nPMnjdiXXdri8bH_w==
```

```
$ xcrun xcresulttool export --path $xcresultpath --output-path $destinationPath/$filename --id 0~JVQMKe6IgJ5d1MjzEtX2RGgfrtdmFxsP76yVkE-zoykhLqzSlva1VbnxkOVCNIjxXPfh2ldYqmCoyldz_YifmQ== --type file
```



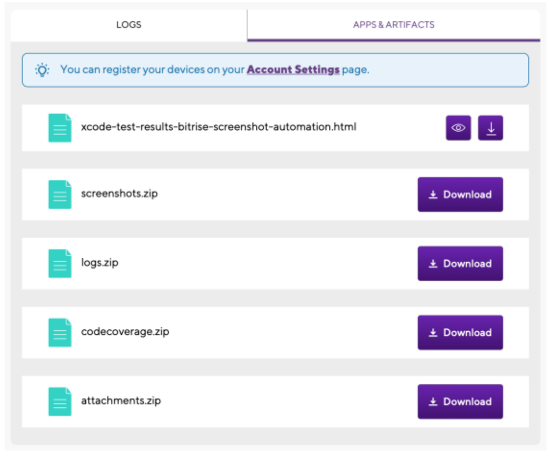
You have no connections yet.

Become an eVIP through Connections.

Connections allows businesses on the ChargePoint network to share offers and deals with select EV drivers.

xcparse

- Swift command line tool to parse Xcode xcresult
  - Extracts screenshots, attachments, logs (device, testmanagerd, etc.), & code coverage
  - *xcparse screenshots* makes extraction one command
- Options to separate screenshots by model, OS, language, test plan, test, & more
- Bitrise plugin support
  - Adds ZIPs with files directly to Bitrise Deploy
  - Provides the paths to the ZIPs as output for your own scripts to use



You can register your devices on your **Account Settings** page.

xcode-test-results-bitrise-screenshot-automation.html

screenshots.zip

logs.zip

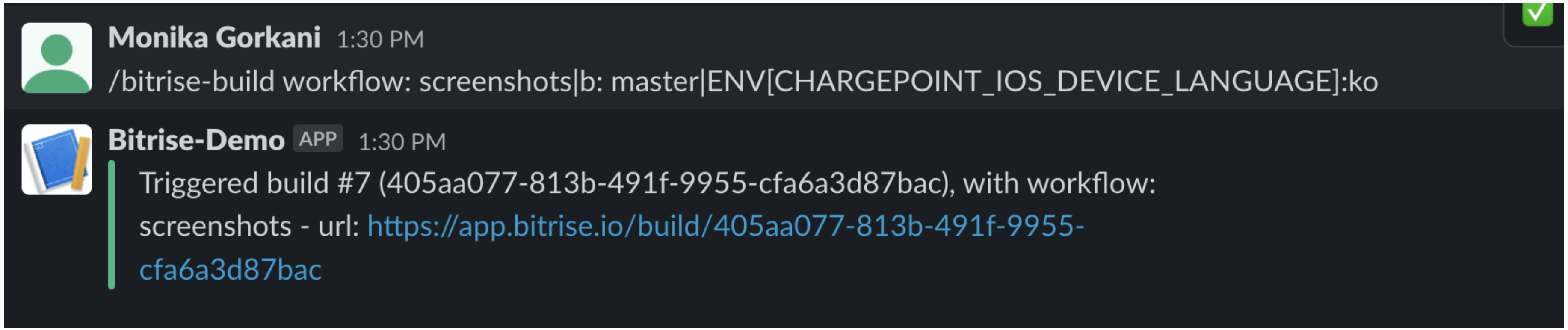
codecoverage.zip

attachments.zip

Integrating with Bitrise

What's our Goal?

Making it easy for any developer on the team to launch customized screenshot builds



Monika Gorkani 1:30 PM

/bitrise-build workflow: screenshots|b: master|ENV[CHARGEPOINT_IOS_DEVICE_LANGUAGE]:ko

Bitrise-Demo APP 1:30 PM

Triggered build #7 (405aa077-813b-491f-9955-cfa6a3d87bac), with workflow: screenshots - url: https://app.bitrise.io/build/405aa077-813b-491f-9955-cfa6a3d87bac

Environment Variables

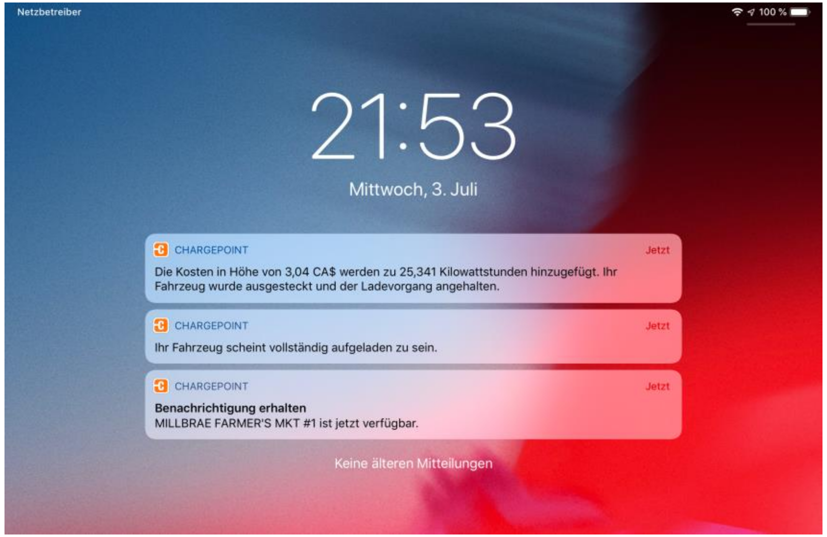
+ Use environment variables to configure options like test plan, configuration, device, interface style
+ Allows you to have one workflow that can do all the work
+ You can set default values for the environment variables when workflow is invoked or customize it if you make a workflow clone
+ Allows for replacement of values via curl or Bitrise Slack slash commands
  + Need a script step to set environment variable from those passed via APIs

```bash
#!/bin/bash

# Pull any arguments that may have been set from Slack slash commands and such
if [ ! -z "$API_CHARGEPOINT_IOS_DEVICE_LANGUAGE" ] ; then
  envman add --key CHARGEPOINT_IOS_DEVICE_LANGUAGE --value "$API_CHARGEPOINT_IOS_DEVICE_LANGUAGE"
fi
```

Device Language in Simulator

+ Test plan language settings only affect the app language & region, not the device
  + Leaves OS dialogues that don't happen in your process unlocalized
+ Modifying the Simulator's defaults can be a workaround for this

```
#!/bin/bash

# Iterate over every Simulator & modify the global
preferences for lang/region
find ~/Library/Developer/CoreSimulator/Devices/* -
type d -maxdepth 0 -exec /usr/libexec/PlistBuddy -c
"Delete :AppleLanguages" -c "Add :AppleLanguages
array" -c "Add :AppleLanguages:0 string
$CHARGEPOINT_IOS_DEVICE_LANGUAGE" -c "Delete
:AppleLocale" -c "Add :AppleLocale string
$CHARGEPOINT_IOS_DEVICE_REGION"
{}/data/Library/Preferences/.GlobalPreferences.plist
\;
```


Netzbetreiber

100 %

21:53

Mittwoch, 3. Juli

CHARGEPOINT Jetzt

Die Kosten in Höhe von 3,04 CA$ werden zu 25,341 Kilowattstunden hinzugefügt. Ihr Fahrzeug wurde ausgesteckt und der Ladevorgang angehalten.

CHARGEPOINT Jetzt

Ihr Fahrzeug scheint vollständig aufgeladen zu sein.

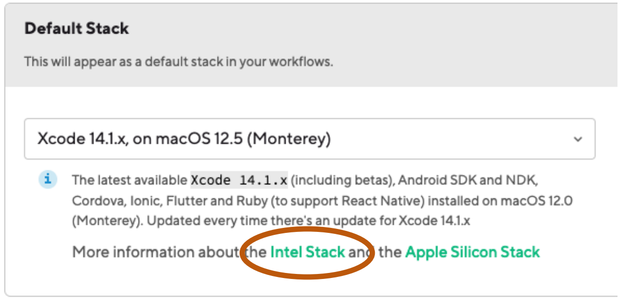
CHARGEPOINT Jetzt

Benachrichtigung erhalten
MILLBRAE FARMER'S MKT #1 ist jetzt verfügbar.

Keine älteren Mitteilungen

Stack Simulator Availability

+ Bitrise stacks may not have all the Simulators you want
+ Check the stack information to see what's in the VM by default



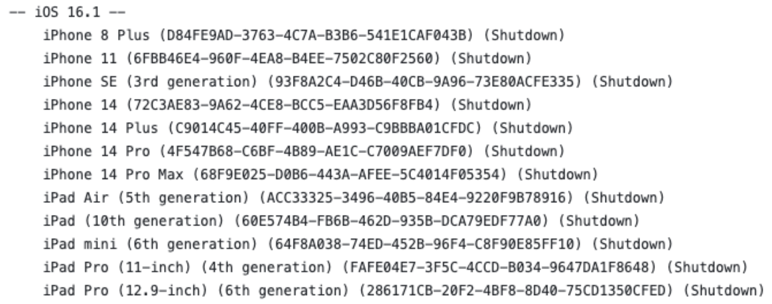
Default Stack

This will appear as a default stack in your workflows.

Xcode 14.1.x, on macOS 12.5 (Monterey)

The latest available `Xcode 14.1.x` (including betas), Android SDK and NDK, Cordova, Ionic, Flutter and Ruby (to support React Native) installed on macOS 12.0 (Monterey). Updated every time there's an update for Xcode 14.1.x

More information about the Intel Stack and the Apple Silicon Stack

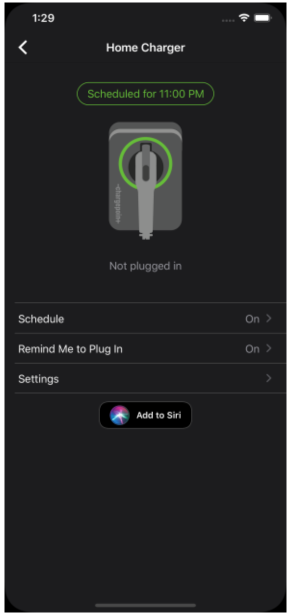


```
-- iOS 16.1 --
    iPhone 8 Plus (D84FE9AD-3763-4C7A-B3B6-541E1CAF043B) (Shutdown)
    iPhone 11 (6FBB46E4-960F-4EA8-B4EE-7502C80F2560) (Shutdown)
    iPhone SE (3rd generation) (93F8A2C4-D46B-40CB-9A96-73E80ACFE335) (Shutdown)
    iPhone 14 (72C3AE83-9A62-4CE8-BCC5-EAA3D56F8FB4) (Shutdown)
    iPhone 14 Plus (C9014C45-40FF-400B-A993-C9BBBA01CFDC) (Shutdown)
    iPhone 14 Pro (4F547B68-C6BF-4B89-AE1C-C7009AEF7DF0) (Shutdown)
    iPhone 14 Pro Max (68F9E025-D0B6-443A-AFEE-5C4014F05354) (Shutdown)
    iPad Air (5th generation) (ACC33325-3496-40B5-84E4-9220F9B78916) (Shutdown)
    iPad (10th generation) (60E574B4-FB6B-462D-935B-DCA79EDF77A0) (Shutdown)
    iPad mini (6th generation) (64F8A038-74ED-452B-96F4-C8F90E85FF10) (Shutdown)
    iPad Pro (11-inch) (4th generation) (FAFE04E7-3F5C-4CCD-B034-9647DA1F8648) (Shutdown)
    iPad Pro (12.9-inch) (6th generation) (286171CB-20F2-4BF8-8D40-75CD1350CFED) (Shutdown)
```

+ If there's a Simulator you need, create it in a script step using simctl

```
#!/bin/bash

# Create the iPad Pro (12.9-inch) (5th generation) simulator
xcrun simctl create "iPad Pro (12.9-inch) (5th generation)" "com.apple.CoreSimulator.SimDeviceType.iPad-Pro-12-9-inch-5th-generation" "com.apple.CoreSimulator.SimRuntime.iOS-16-0"
```

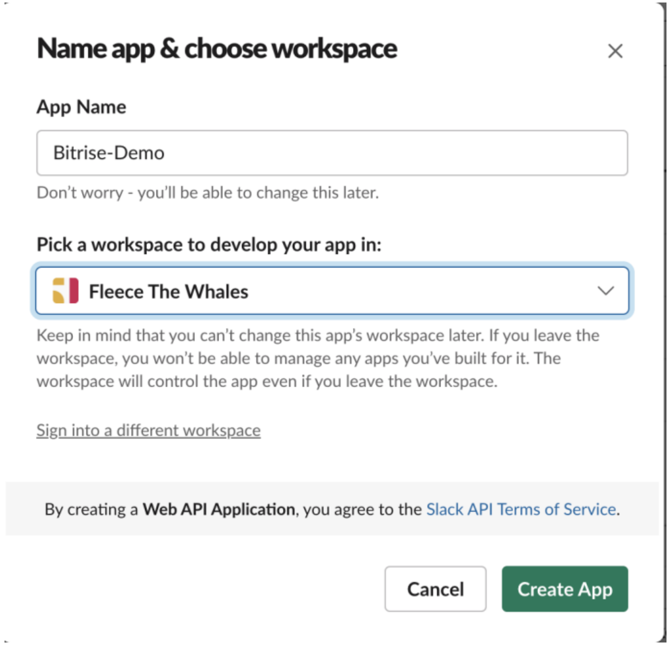
Dark Mode Screenshots



- Xcode 11.4+ added simctl command to change device's UI style
  - *xcrun simctl ui <DEVICE> appearance dark*
- For Xcode 11.0 to 11.3, this option doesn't exist
  - Can modify the Info.plist of your app in a script step
    - Real device or simulator
    - Affects app UI style only
  - Can modify default in *com.apple.uikitservices.userInterfaceStyleMode*
    - Simulator only
    - Affects device & app UI style

Slack Slash Commands for Bitrise

+ Create an app in Slack (api.slack.com)
+ Set up Slack commands using the outgoing webhook from Bitrise



Name app & choose workspace

App Name

Bitrise-Demo

Don't worry - you'll be able to change this later.

Pick a workspace to develop your app in:

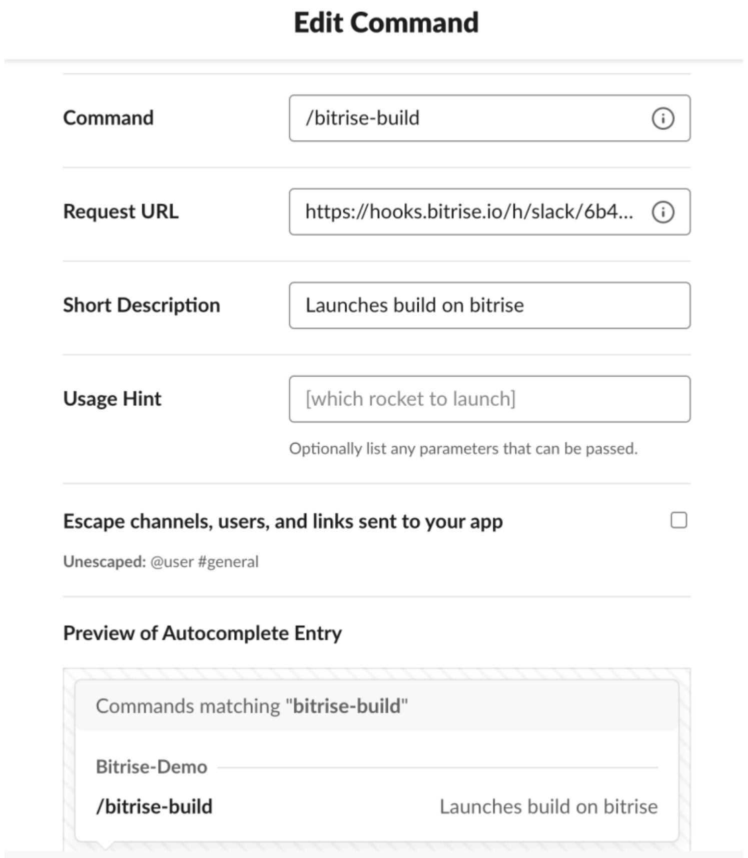
Fleece The Whales

Keep in mind that you can't change this app's workspace later. If you leave the workspace, you won't be able to manage any apps you've built for it. The workspace will control the app even if you leave the workspace.

Sign into a different workspace

By creating a **Web API Application**, you agree to the Slack API Terms of Service.

Create slash command



Edit Command

Command	/bitrise-build
Request URL	https://hooks.bitrise.io/h/slack/6b4...
Short Description	Launches build on bitrise
Usage Hint	[which rocket to launch]

Optionally list any parameters that can be passed.

Escape channels, users, and links sent to your app ☐

Unescaped: @user #general

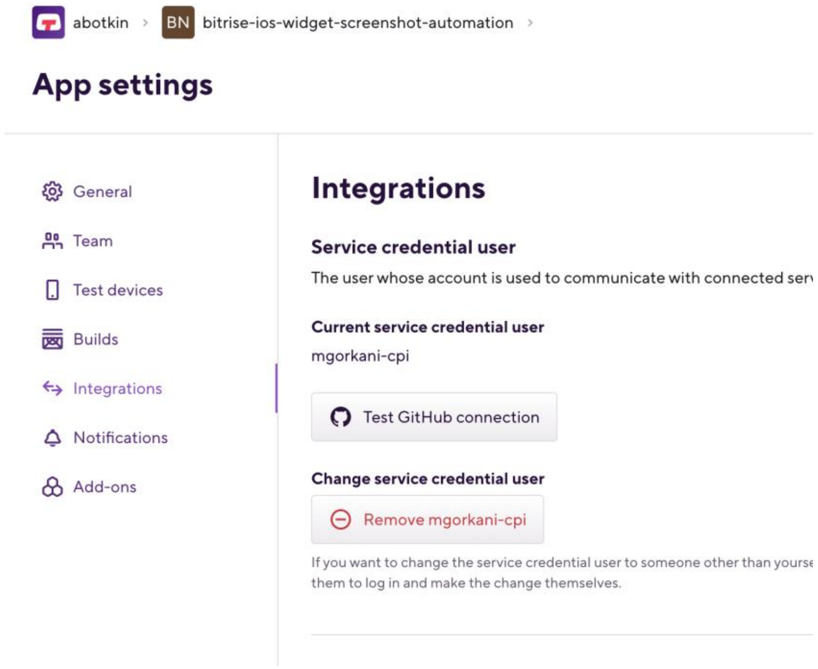
Preview of Autocomplete Entry

Commands matching "**bitrise-build**"

Bitrise-Demo

/bitrise-build Launches build on bitrise

Getting outgoing webhook from Bitrise



abotkin › BN bitrise-ios-widget-screenshot-automation ›

App settings

- General
- Team
- Test devices
- Builds
- Integrations
- Notifications
- Add-ons

Integrations

Service credential user

The user whose account is used to communicate with connected ser

Current service credential user

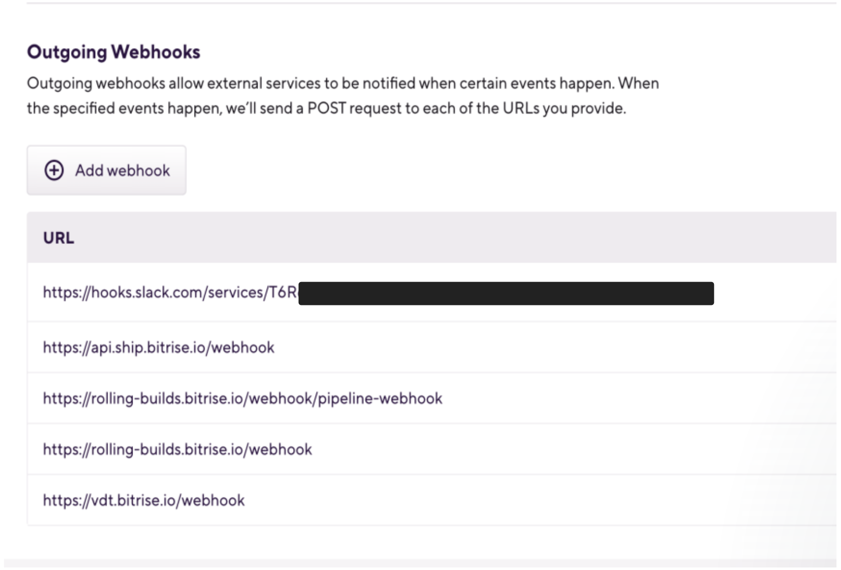
mgorkani-cpi

Test GitHub connection

Change service credential user

Remove mgorkani-cpi

If you want to change the service credential user to someone other than yourse them to log in and make the change themselves.



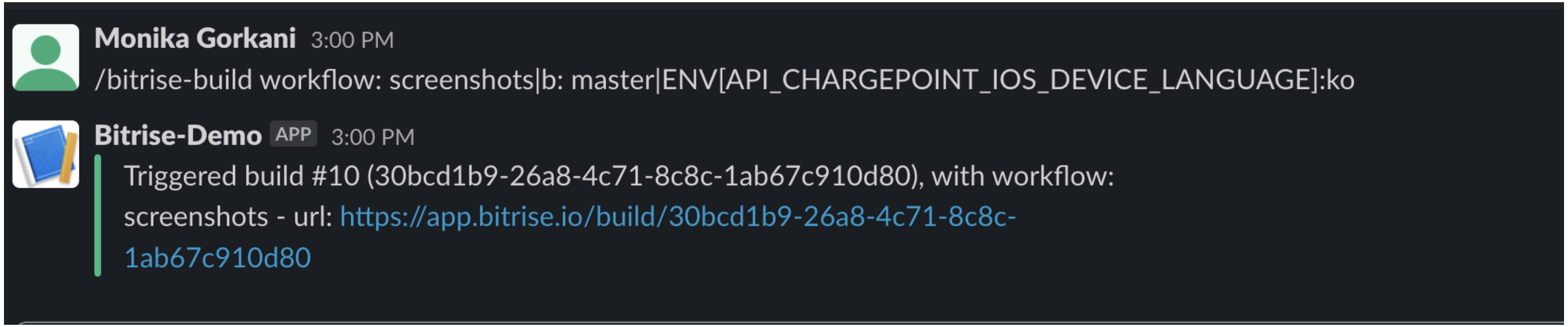
Outgoing Webhooks

Outgoing webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide.

Add webhook

URL
https://hooks.slack.com/services/T6R[redacted]
https://api.ship.bitrise.io/webhook
https://rolling-builds.bitrise.io/webhook/pipeline-webhook
https://rolling-builds.bitrise.io/webhook
https://vdt.bitrise.io/webhook

Sample slack command



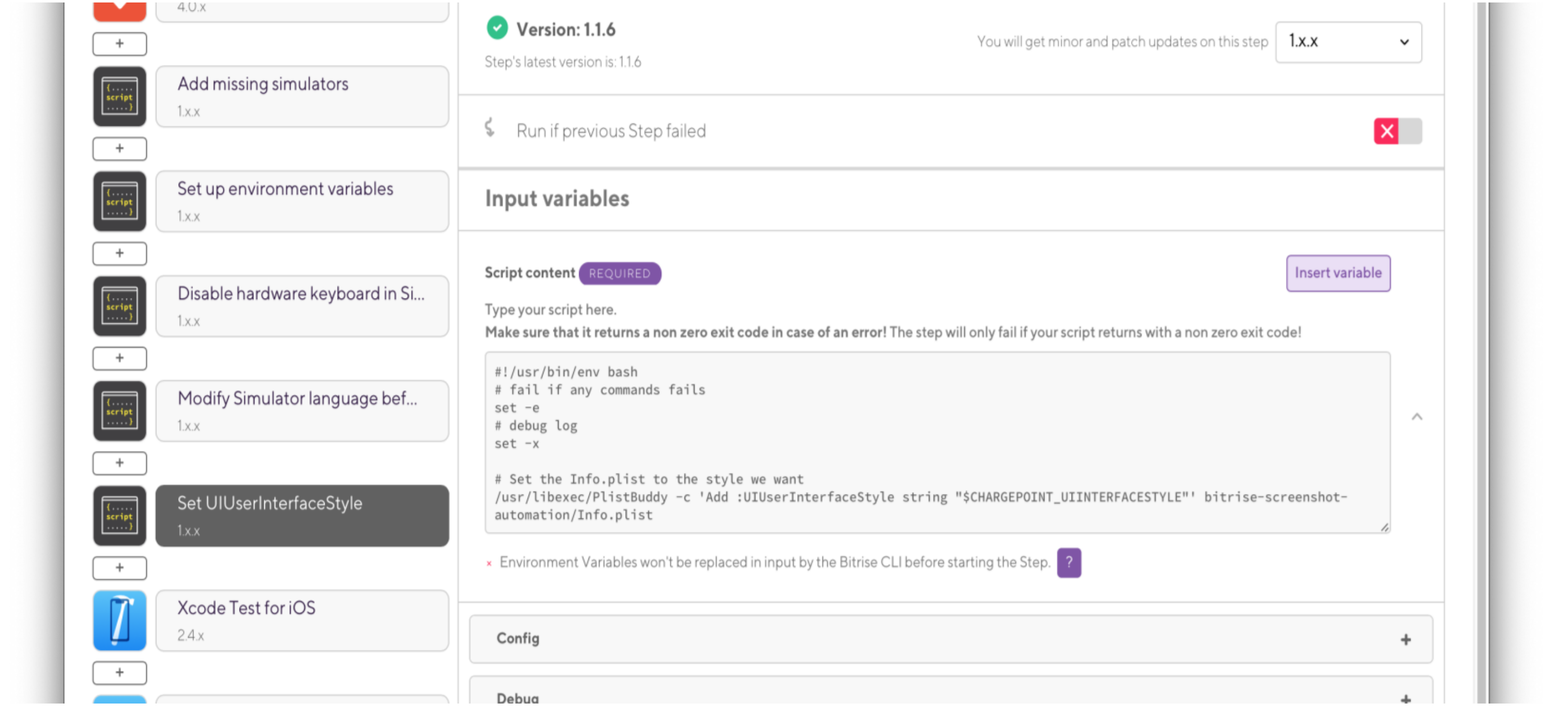
Monika Gorkani 3:00 PM

```
/bitrise-build workflow: screenshots|b: master|ENV[API_CHARGEPOINT_IOS_DEVICE_LANGUAGE]:ko
```

Bitrise-Demo APP 3:00 PM

Triggered build #10 (30bcd1b9-26a8-4c71-8c8c-1ab67c910d80), with workflow: screenshots - url: https://app.bitrise.io/build/30bcd1b9-26a8-4c71-8c8c-1ab67c910d80

Sample App: bitrise-ios-widget-screenshot-automation



Version: 1.1.6

Step's latest version is: 1.1.6

You will get minor and patch updates on this step: 1.x.x

Run if previous Step failed

Input variables

Script content REQUIRED

Insert variable

Type your script here.

Make sure that it returns a non zero exit code in case of an error! The step will only fail if your script returns with a non zero exit code!

```
#!/usr/bin/env bash
# fail if any commands fails
set -e
# debug log
set -x

# Set the Info.plist to the style we want
/usr/libexec/PlistBuddy -c 'Add :UIUserInterfaceStyle string "$CHARGEPOINT_UIINTERFACESTYLE"' bitrise-screenshot-
automation/Info.plist
```

× Environment Variables won't be replaced in input by the Bitrise CLI before starting the Step. ?

https://github.com/ChargePoint/bitrise-ios-widget-screenshot-automation

Hands-on

Thank You

For further information on this topic,
check out the sample code on GitHub

https://github.com/ChargePoint/bitrise-ios-widget-screenshot-automation