**estimote**私たちの
APIdevのコミュ
ニティ屋内
SDKお問い合わせ
わせ≡プロ
グ

APIドキュメント

APIドキュメント

入門

- セットアップdev環境
- サンプルコードをコンパイルします
- サンプルを変更します
- 独自のアプリを作成します
- あなたの成果を共有します

チュートリアル

- 距離
- 近接
- 通知

iOSのSDK

AndroidのSDK

APIリファレンス

小売アプリケーション

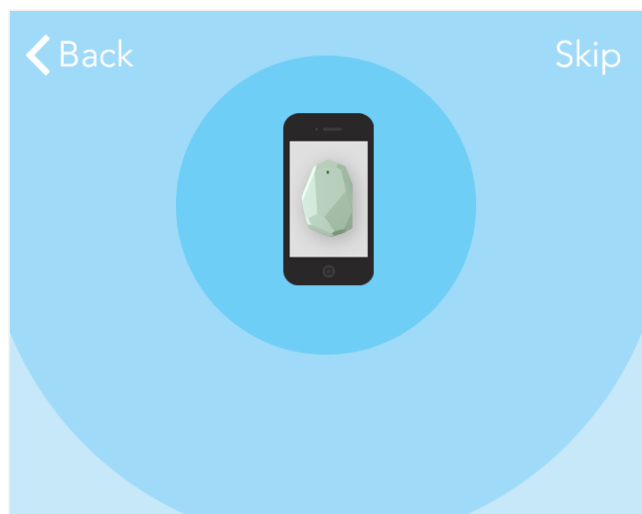
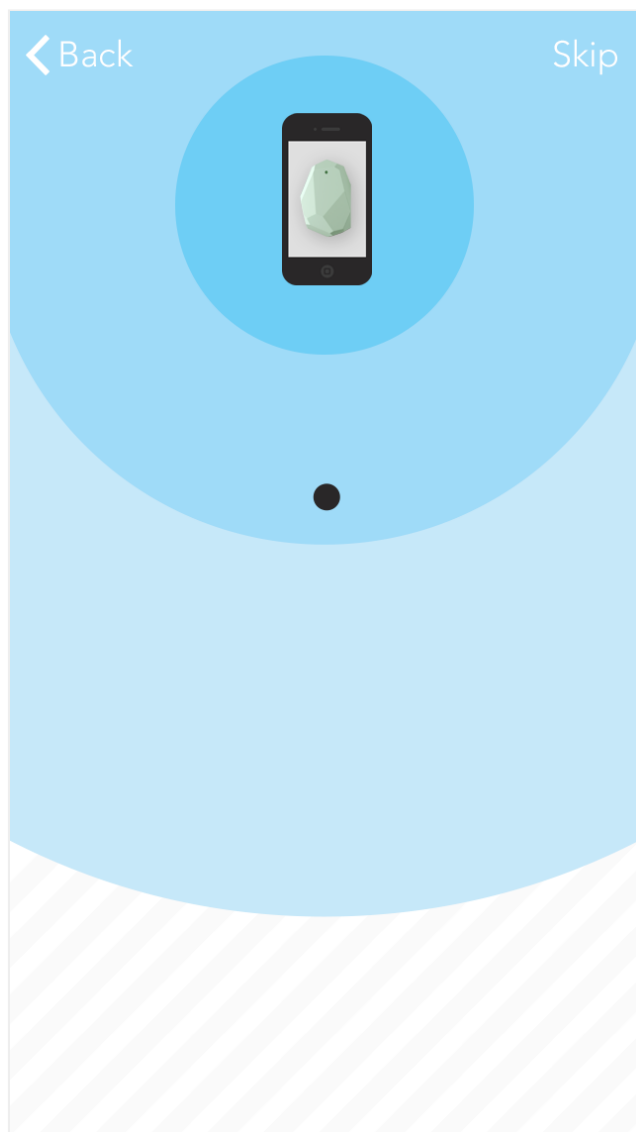
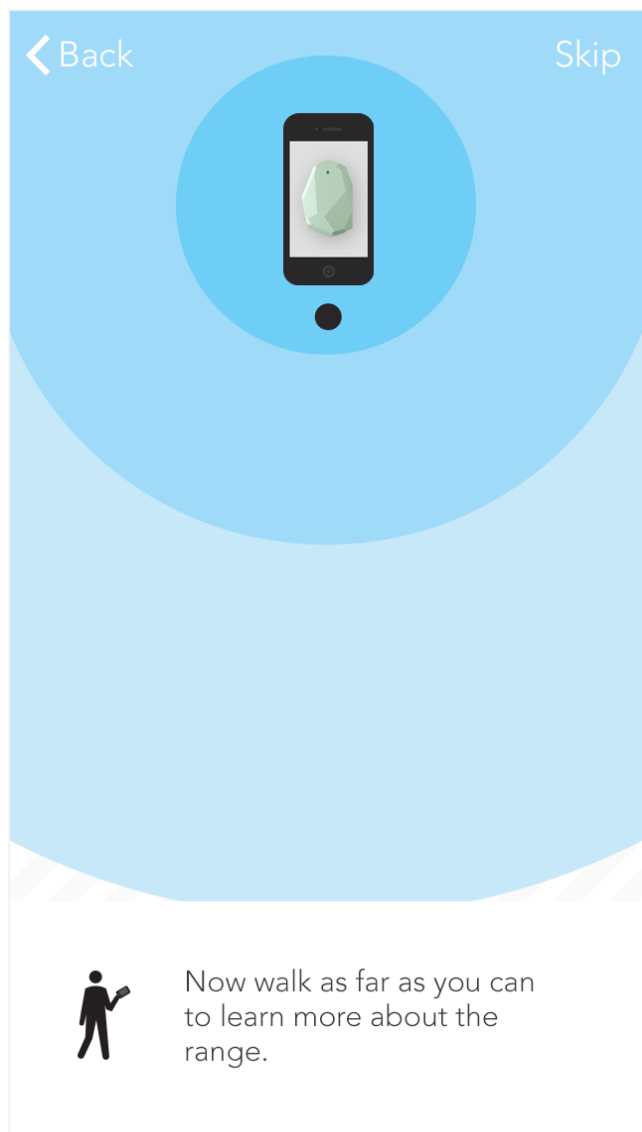
Estimoteについて

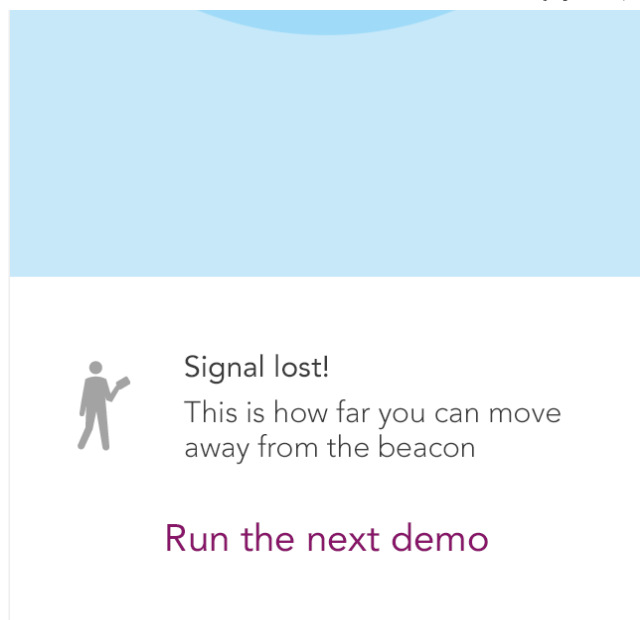
参加しませんか

プレスキット

はじめに

距離のアプリは、最も単純なEstimoteデモアプリの一つ、またまた、その機能を使用すると、Estimoteビーコンで行うことができます他のすべてのものの非常にコアであるとして理解することが最も有用なの一つです。





どのように動作します

このアプリを測定し、ユーザーの位置とビーコンとの間の距離を可視化します。この情報は、人は、彼らが停止した場合は離れて移動したり、近くにビーコンに来ている、と人はであるコンテキストに関連するイベントを作成する場合に指示するなどを使用することができます。

それが動作する方法は本当に簡単です - ビーコンは、そのIDと信号の強度、またはRSSI（受信信号強度表示）をブロードキャスト、連続的に、携帯電話によって監視されています。信号強度及び校正データに基づいて携帯電話がビーコンであるか遠く推測することが可能です。

干渉を排除

可能性があるため干渉するのではなく、読み出しに影響を与えるだろう、突然のスパイクやディップを、フィルタリングされますように、信号強度をサンプリングすることが重要です。これは、アカウントに、複数回の読み出しをとるローリング平均、1つだけでなく、を使用して行われます。これは、応答性のわずかな低下を犠牲にして、距離値を安定させます。

ユーザーインターフェイスでの表示結果

「即時」、「近い」と「遠い」：ユーザーインターフェイスでは、ビーコンの周囲は、3つの「ゾーン」に分割されています。これらのゾーンは、このデモで使用されていないが、他のデモアプリケーションで使用されています。彼らは完全に信号を失うまで、ユーザーができるだけ離れて行くように頼まれているようしかし、この「範囲

外」ストライプをデモゾーンで、使用されています。しかし実際的な理由から、ユーザーが十分本当にシグナルを失うこと歩くことができないように、メッセージが、かなり以前示されている「信号が失われました」。

それを行う方法

距離のデモを再作成することは非常に簡単です。のは、このデモ・アプリのいくつかの重要な要素を見てみましょう。

まず、あなたが例のプロジェクトに含まれているプロジェクトだけでなく、libEstimateSDK.aにCoreBluetooth.frameworkとCoreLocation.frameworkを追加したことを確認してください。

```

1  - (ボイド) のviewDidLoad
2  {
3      [超のviewDidLoad ] ;
4
5      //////////////////////////////////////////
6      //設定Estimoteビーコンマネージャー
7
8      //ビーコン・マネージャー・インスタンスの作成
9      、自己を。 beaconManager = [ [ ESTBeaconManagerのアロケーション]のinit ] ;
10     自己。 beaconManager 。デリゲート=自己；
11     自己。 beaconManager 。 avoidUnknownStateBeacons = YES ;
12
13     //作成したサンプル領域オブジェクト（あなたはまた、メジャーまたはマイナー+主な引数を渡すこ
14     ESTBeaconRegion *地域= [ [ ESTBeaconRegionのallocは] initWithProximityUI
15
16     地域のEstimoteビーコンを探し始める//
17     //ビーコンは範囲、beaconManagerにありますdidRangeBeacons : inRegionは : 起動される
18     [自己を。 beaconManager startRangingBeaconsInRegion : 地域] ;
19
20     //////////////////////////////////////////
21     //セットアップビュー
22
23     [自己setupView ] ;
24 }

```

最初のステップは、使用してEstimoteビーコンManagerオブジェクトを作成することでESTBeaconManagerのクラスおよびデリゲートオブジェクトを設定します。この例では、使用しているavoidUnknownStateBeaconsのフラグを。コアロケーションフレームワークは、発見されたビーコンのリストからオブジェクトの除去にいくつかの慣性を持っており、このプロパティをYESに設定すると、私たちは、リストから未知の状態オブジェクトの削除を強制することができます。また、起動

startRangingBeaconsInRegionを：方法は基本的に記述されているEstimoteビーコンを探して開始するESTBeaconRegionのオブジェクト。

```

1  - (ボイド) setupView
2  {
3      //////////////////////////////////////////
4      //セットアップ背景画像
5
6      CGRect screenRect          = [ [ UIScreen mainScreen ]境界] ;
7      CGFloat screenHeight       = screenRect . サイズ. 高さ;
8      UIImageView *      backgroundImageの。
9
10     場合 ( screenHeight > 480 )
11         backgroundImageの= [ [ UIImageViewのalloc ] initWithImage :[ UIImage
12     他の
13         backgroundImageの= [ [ UIImageViewのalloc ] initWithImage :[ UIImage
14
15     [セルフ. ビュー addSubview : backgroundImageの] ;
16
17     //////////////////////////////////////////
18     //セットアップ点画像の
19
20     自己. positionDot = [ [ UIImageViewのalloc ] initWithImage :[ UIImage im
21     [セルフ. positionDot setCenter :自己. ビュー. センター] ;
22     [セルフ. positionDot setAlpha :1 . ] ;
23
24     [セルフ. ビュー addSubview :セルフ. positionDot ] ;
25
26     自己. dotMinPos = 150 ;
27     自己. dotRange =自己. ビュー. 境界. サイズ. 高さ - 220 ;
28 }

```

でsetupViewの方法我々は、背景画像と同様に、ビーコンと携帯電話との間の距離を示している点画像を定義します。

```

1  - (ボイド) beaconManager : ( ESTBeaconManager * ) マネージャー
2      didRangeBeacons : ( NSArrayの* ) ビーコン
3      inRegion : ( ESTBeaconRegion * ) 領域
4  {
5      ESTBeacon * closestBeacon 。
6
7      場合 ([ビーコンカウント] > 0 )
8      {
9          //ビーコン配列は、距離に基づいてソートされている
10         //最も近いビーコンが最初のものである
11         closestBeacon = [ビーコンobjectAtIndex :0 ] ;

```

```

12
13 //計算し、設定された新たなY位置
14     フロートnewYPos = self . dotMinPos + ( ( float ) closestBeacon . rssi
15     self . positionDot . center = CGPointMake ( self . view . bounds . s
16 }
17 }
```

私たちのUIとするとESTBeaconManagerが整備されている、実行するだけで、残りのものは発見されたビーコンを処理することです。我々は、使用didRangeBeacons : beaconManagerをinRegion : デリゲートメソッドは、それをします。発見されたビーコンのリストは、距離に基づいてソートされている（最も近いビーコンは、リストの先頭にあります）。私たちは、最寄りの取得ESTBeaconのインデックス0を使用してオブジェクトを私たちは、現在のRSSI値を得るためにそれを使用しており、それに基づいて、我々は、ドット位置を更新します。RSSI値は、-100から0までの範囲、およびビーコン近いほど、値が0に近いほど、デバイスにあります。

あなたは私たちのGitHubから（距離のデモを含む）の完全な例としてプロジェクトをダウンロードすることができます。

ソースコードをダウンロード



近接デモアプリ

ゾーンを横断する際に、メッセージを表示します



通知デモアプリ

ゾーンを離れる時に送信されるプッシュ通知

ホーム

それがどのように動作しますか？

エンタープライズソリューション

コンタクト

プレス&メディア

サポート

個人情報保護方針

Estimoteでのお仕事

ツイッター

フェイスブック

ブログ

コミュニティ

Estimote社

35 E 19セント3階

ニューヨーク、NY 10003

アメリカ合衆国

EstimoteパルスカSpの東京オブジェクト指向の

UL。Krakusa 11

30から535クラクフ、ポーランドの

欧州連合（EU）

著作権2012-2015 Estimote社によって©、ブルートゥース、アップル、Googleとその他の商標はそれぞれの商標所有者に帰属します。