

# 空き家ツールボックス

スマートフォンと無料のオープンソースツールを用いて空き家情報を収集・利用するための手引書



<http://www.akiyatoolbox.org>

第0.91版 (2017年2月28日)

# はじめに ..... 5

オープントーデータキット .....	5
インストール方法 .....	6
基本操作 .....	7

# サーバの設定 ..... 12

Google クラウドプラットフォームにインストールする方法 .....	12
サーバの導入にあたって .....	26

# 入力フォームの作成 ..... 31

簡単な入力フォームの作成 .....	31
作成した入力フォームを使ってみる .....	35
参考情報 .....	36

# XLSForm ..... 38

基本フォーマット .....	38
質問の種類 .....	39
ヒント .....	44
制約条件 .....	44
関連制御 (Relevant) .....	45
数式 .....	46
計算 .....	47
回答必須 .....	47
グループ作成 .....	48
繰り返し .....	49
多言語への対応 .....	51
メディア .....	52
位置情報 .....	53
CSVデータの事前読み込み .....	53

事前に読み込んだデータの動的選択.....	55
カスケード選択.....	57
外部選択.....	58
デフォルト .....	58
読み取り専用 .....	58
外観 (Appearance) .....	59
settings (設定) ワークシート .....	59
スタイルリング.....	63
XLSFormに対応したプラットフォーム／ツール.....	63
その他のリソース .....	64
XLSFormの歴史 .....	64
<b>セキュリティとプライバシー.....</b>	<b>65</b>
ライセンス .....	65
通信手段.....	65
サードパーティのソフトウェア .....	66
ODK Aggregateの通信.....	66
ODK AggregateをGoogle App Engineや他のホスティングサービス上で運用する場合	66
暗号化されたフォームのセキュリティ .....	67
ODK Aggregateサーバにおけるユーザ名による認証.....	68
Google Gmailによる認証.....	68
識別情報の送信と蓄積 .....	68
オンラインのXLSForm変換ツール .....	70
ウェブサイト .....	70
Google Playストア .....	70
参考文献 .....	71
<b>フォームの暗号化.....</b>	<b>72</b>

暗号化の準備.....	73
セキュリティ上の注意点 .....	73
設定方法.....	74
フォームの定義.....	75
RSA鍵ペアの作成 .....	76
運用について .....	79

## 第1章

# はじめに

この手引書はスマートフォンと無料のソフトウェアを使って、地域の空き家情報を効率的に収集・利用するための方法や注意点をまとめたものです。具体的には、まずオープンデータキットと呼ばれるソフトウェアを用いて、空き家調査フォームを作成します。次に、作成したフォームを用いてスマートフォン上で空き家情報の入力を行います。

Akiya Toolbox (<http://akiyatoolbox.org>) というウェブサイトから、空き家調査フォームをダウンロードすれば、すぐに調査を開始することができます。必要に応じて、空き家調査フォームをカスタマイズすることもできます。

オープンデータキットで集めたデータは、ウェブ上で表示したり、パソコンにダウンロードしたりすることができます。パソコンにダウンロードしたデータは、Open Office(<http://www.openoffice.org/ja/>)、Libre Office(<https://ja.libreoffice.org/>)、QGIS(<http://qgis.org/ja/site/>)、R(<https://www.r-project.org/>)、Jupyter(<http://jupyter.org/>)といった無料のソフトウェアに読み込んで、様々な可視化や分析を行うことができます。一般に公開しても構わないデータであれば、Tableau Public(<https://public.tableau.com/s/>)、Carto(<https://carto.com/>)といったクラウド上の無料ツールを利用して、とても手軽に可視化や分析を行うことができます。

## オープンデータキット

オープンデータキット（Open Data Kit: ODK）<sup>1</sup>とは、スマートフォンなどの携帯端末を使って情報を集めることができる無料のソフトウェアです。

スマートフォンをお持ちであれば、簡単に使うことができます。具体的に何ができるかというと、

1. 情報収集用の入力フォームを作成し、
2. スマートフォンなどの携帯端末を使って情報収集を行い、
3. 収集した情報をインターネットに接続されたコンピュータ（サーバ）に送信し、

---

<sup>1</sup> <https://opendatakit.org/>

4. サーバに集まってきたデータを集約し、使いやすい形式の情報に変換し、出力することができます。

また、オープンデータキットには次のような特徴があります：

- 無料で使って、機能が豊富です。
- 通常のアンケートに記入するような内容以外に、位置情報や写真、各種マルチメディア情報を簡単に入力することができます。また、目的や環境に応じて自由に機能をカスタマイズすることができます。
- 海外で利用されています。特に、欧米、アフリカ、アジアの一部の国など。
- 日本語のデータも扱うことができますので、今後日本でも利用が広がる可能性があります。
- インターネット接続がない場所でも使えます。
- 途中で入力を中断し、あとで残りを入力することができます。
- いわゆるオープンソースソフトウェアですので、システムのソースコードは公開されています<sup>2</sup>。
- 米国ワシントン大学の研究者が中心に開発を行っており、開発費は寄付等によって賄われています。

## インストール方法

スマートフォン<sup>3</sup>の「Playストア」で「ODK Collect」を検索し、インストールしてください。

---

<sup>2</sup> Apache License 2.0

<sup>3</sup> Android OSのみ。iOSは未対応。

## 基本操作

では、早速使ってみましょう。ひとつひとつ手順を説明していきます。

- (1) まず、スマートフォン上で、ODK Collect を起動します。少し待つと【メインメニュー】が表示されます。

【メインメニュー】



(2)次に、「空のフォームを取得する」を押します。【フォーム取得画面】が表示されます。取得するフォームにチェックを入れ「選択を取得」を押します。取得に成功したら「OK」ボタンを押します。再び【メインメニュー】が表示されます。

### 【フォーム取得画面】



(3) [メインメニュー]で「空のフォームに入力する」を押して、(2)で取得したフォームを選択します。すると以下の[入力画面]が表示されます。左にスワイプ（指で画面に触って、左に払うように指を動かす）して入力していきます<sup>4</sup>。最後に「フォームを保存して終了」します。再び[メインメニュー]が表示されます。

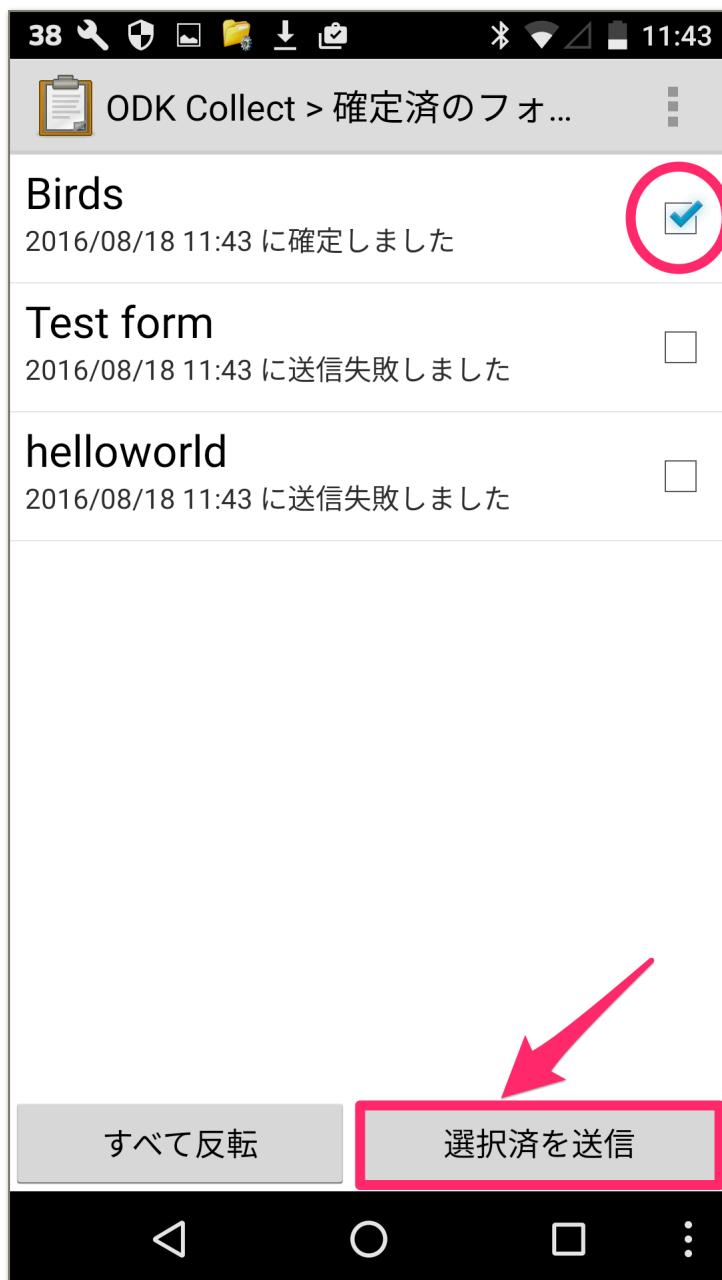
### [入力画面]



<sup>4</sup> 回答をいったん中断し、残りはあとで回答しても構いません。その場合は、「保存ボタン」を押して既に回答した内容を保存しておきます。

(4) [メインメニュー]で「確定済のフォームを送信」を押します。[送信画面]が表示されます。確定するフォームにチェックを入れて「選択済を送信」を押します<sup>5</sup>。

### [送信画面]



この例では、データはデモ用のサーバ（<https://opendatakit.appspot.com>）に送信されます<sup>5</sup>。

<sup>5</sup> 電波が来ていない場所でもデータ収集を行うことができます。その場合、電波が来ている場所に戻ってからまとめてデータをアップロードすることになります。

おめでとうございます！スマートフォンでデータを収集し、サーバに送信することができます<sup>6</sup>。

---

<sup>6</sup>別のサーバを指定するには、ODK Collectの全般設定-->プラットフォーム設定を構成--> URLをタップして設定します。

## 第2章

# サーバの設定

サーバの設定を行うには、ODK Aggregateというソフトウェアを、クラウド(Googleクラウドプラットフォームなど)か、自分たちで用意したサーバにインストールします。このソフトウェアがどういうものかは、デモ用のサーバ (<https://opendatakit.appspot.com>) を見るとなんとなくわかるのではないかと思います。

クラウドにインストールするか、自分で用意したサーバにインストールするかをまず決めなければなりません。初心者の場合はとりあえずGoogle クラウドプラットフォームにインストールしてみるのがおすすめです。たくさんのデータを送信する場合や、プライバシーなどの観点からクラウドで保管したくない情報を扱う場合には、自分で用意したサーバを使うことも検討した方が良いと思います。

## Google クラウドプラットフォームにインストールする方法

以下では、Google クラウドプラットフォームにODK Aggregateをインストールし、設定する手順を示します。

- (1) 手元のコンピュータにJava 7あるいはそれより新しいバージョンがインストールされていることを確認します。
- (2) Gmailアカウントを用意します。そのアカウントを使ってGoogleクラウドプラットフォームを利用します。なお、無料でGoogleクラウドプラットフォームを使う場合、大量のデータをアップロードするなどして使いすぎると、利用制限がかかって使えなくなります。しかし、ロサンゼルス時間の深夜にリセットされますので、それまで待てばまた使えるようになります。

(3)Googleクラウドプラットフォームのプロジェクトを設定します。画面右上の”Console”をクリックして下さい。

The screenshot shows the Google Cloud Platform homepage. At the top right, there is a user profile icon with the word "Console" next to it, which is circled in red. The main heading reads "Build What's Next Better software. Faster." Below it, there are three bullet points: "Use Google's core infrastructure, data analytics and machine learning.", "Secure and fully featured for all enterprises.", and "Committed to open source and industry leading price-performance.". There are two buttons at the bottom left: "TRY IT FREE" and "CONTACT SALES". At the bottom, there are three tabs: "Interactive Tutorial", "Customer Spotlight", and "VM Migration", with "Interactive Tutorial" being the active tab.

(4)今回初めてプロジェクトを設定する場合は「空のプロジェクトを設定」をクリックします。そうでない場合は画面最上部の真ん中あたりの▼からメニューを出して「プロジェクトの作成」を選択します。

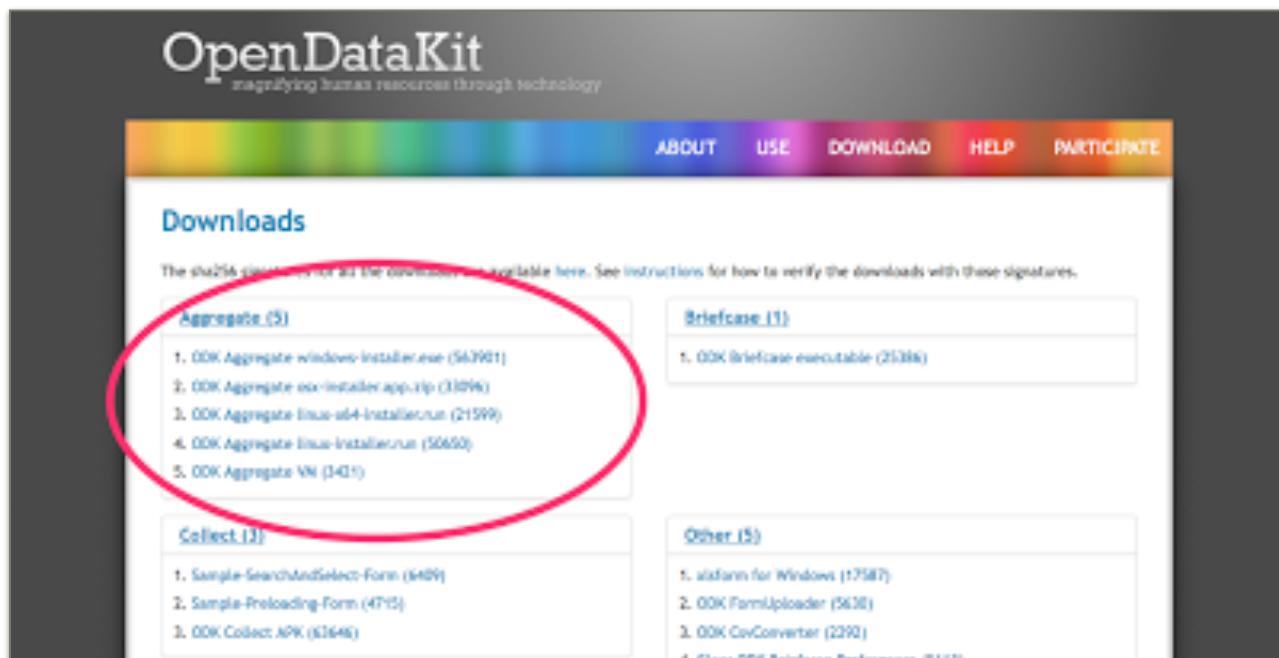
The screenshot shows the Google Cloud Platform dashboard for the project "My First ODK Aggregate". On the right side, there is a context menu with several options: "新しいプロジェクト", "オペレータープロジェクトを作成", "削除 (消去)", and "削除 (消去) [確認]". The option "オペレータープロジェクトを作成" is circled in red. The dashboard also displays sections for "リソース" (Resources), "API", and "エラー報告" (Error Reports).

(5) 「新しいプロジェクト」というタイトルのウィンドウが現れるので、プロジェクト名とプロジェクトIDを設定します。プロジェクトIDは、サーバを利用する際のURLの一部になるので、例えば「xxxcity-akiya」のような感じにしておくと良いと思います。



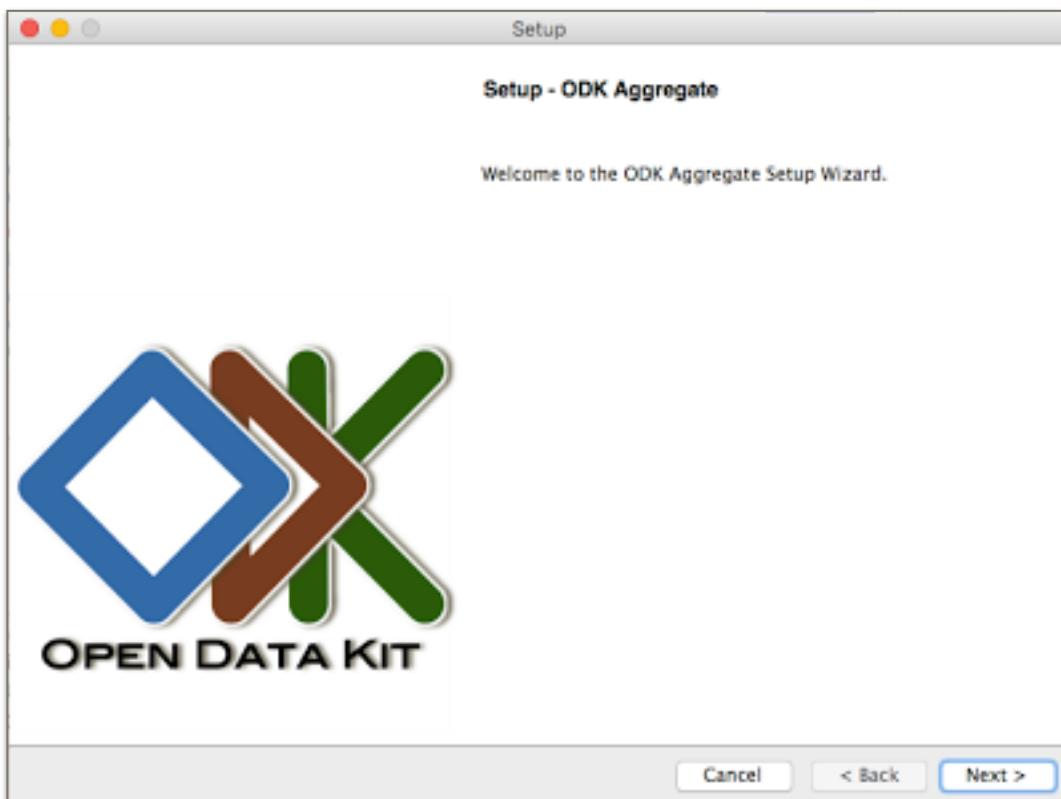
(6) ODK Aggregateを以下のURLからダウンロードします。Windows, OS X, Linuxなど、該当するものをダウンロードしてください。

<https://opendatakit.org/downloads/>

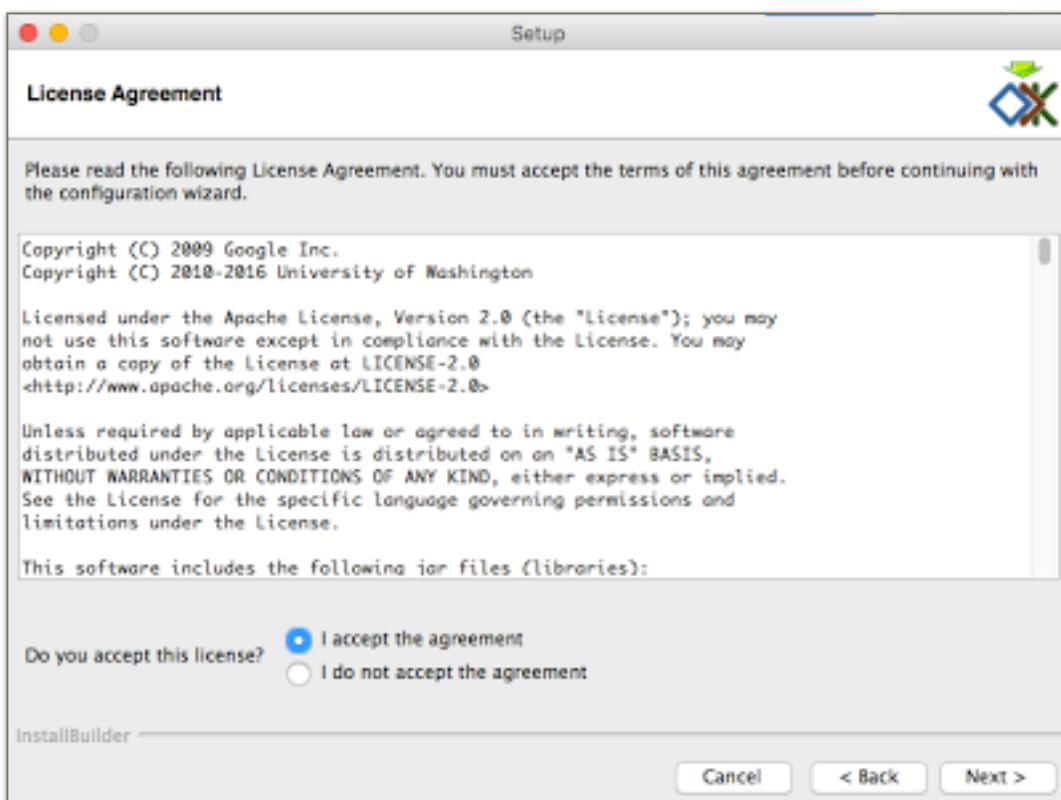


(7) ダウンロードしたファイルを起動するとインストールが始まります。以下、ステップ1から23まで順番に行って下さい。

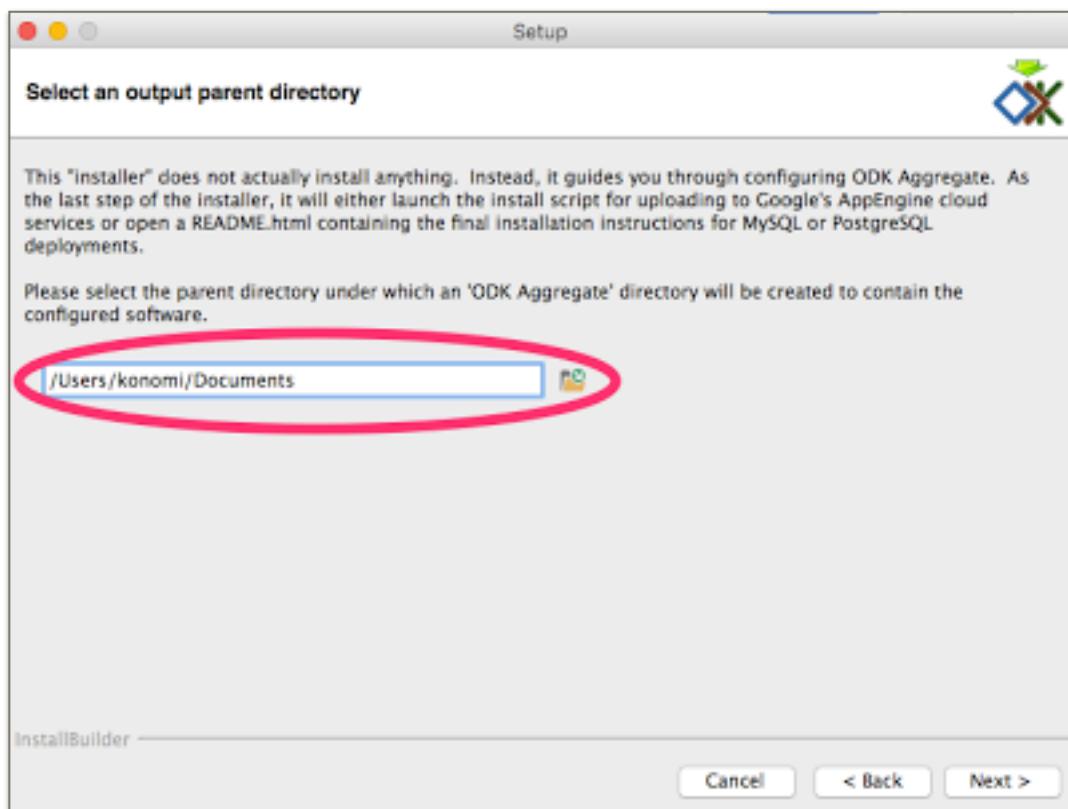
1. 起動画面が表示されます。「Next」を押します。



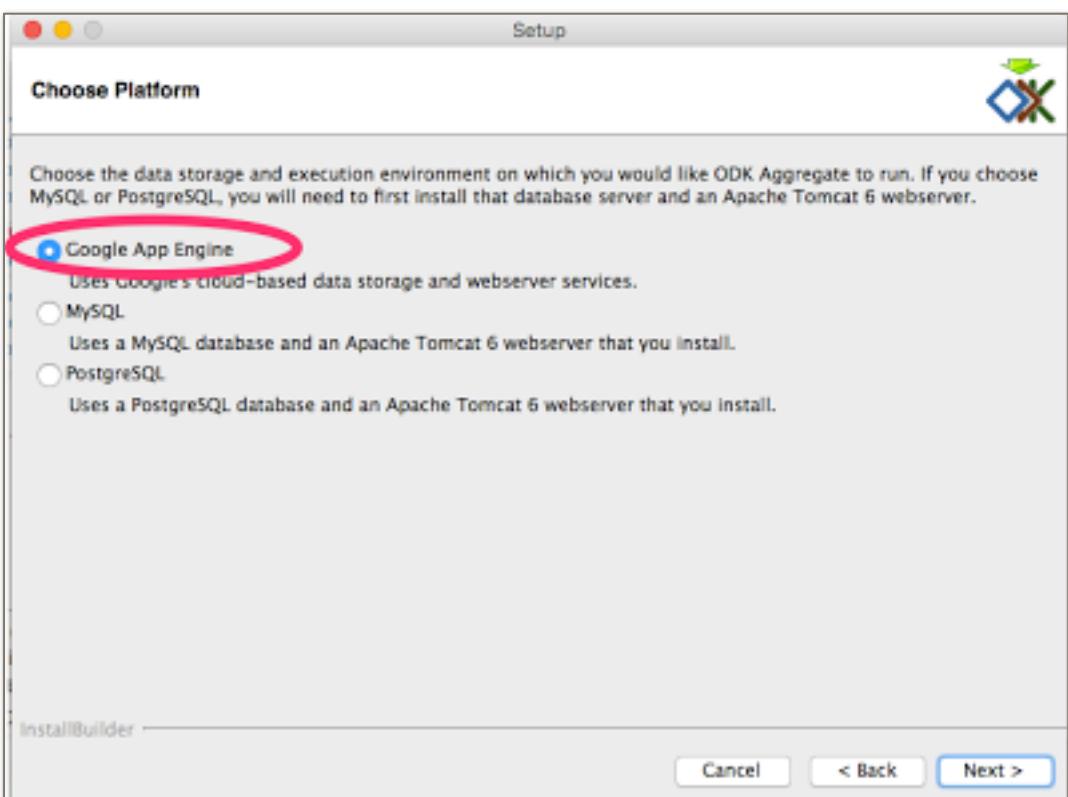
2. ライセンスに同意する(I accept the agreement)を選択し、「Next」を押します。



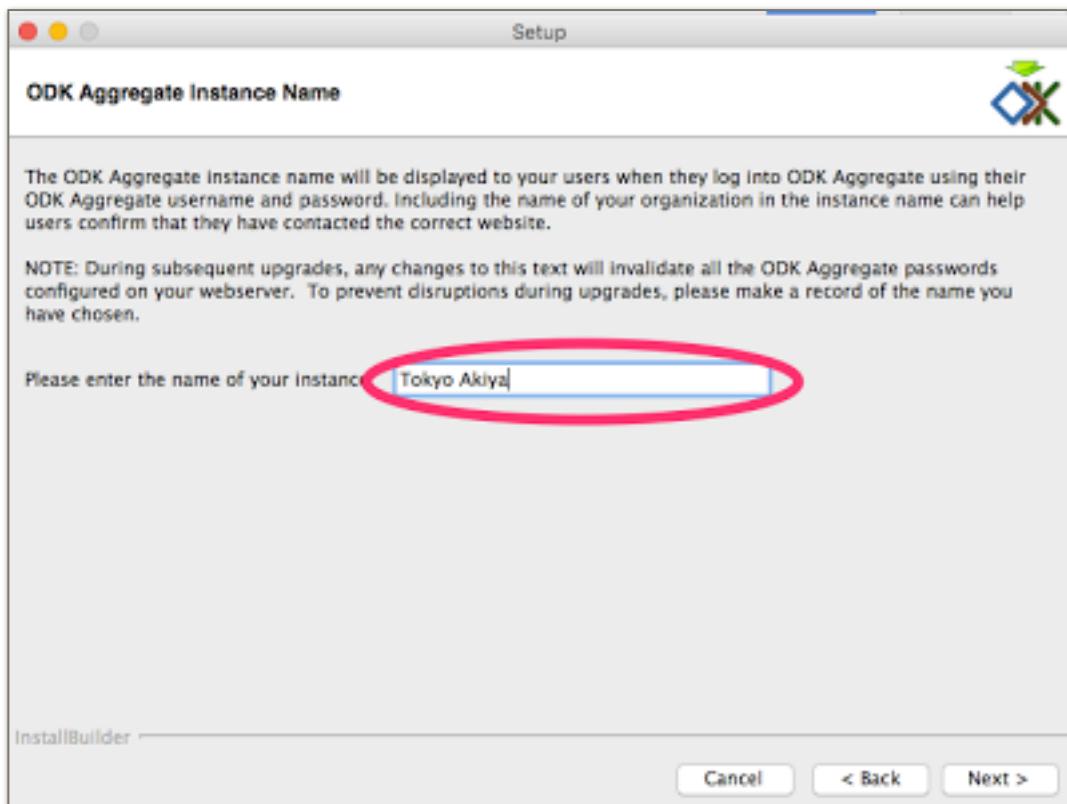
3. 設定したソフトウェアを保存するための、作業用フォルダを指定します。



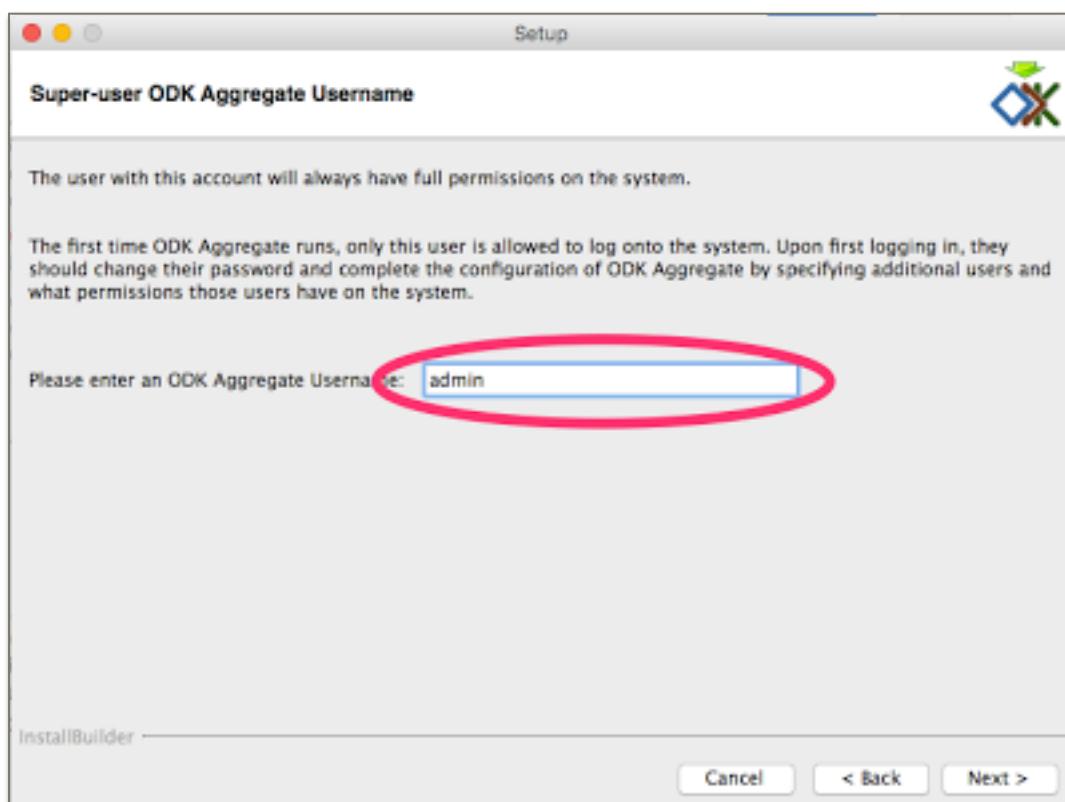
4. Google App Engineを選択します。



5. インスタンス名を指定します。何の情報収集ためのサーバかわかるような名前にする  
と良いと思います。例えば、Tokyo Akiyaとします。



6. 管理用アカウントのログイン名を指定します。例えば「admin」とします。



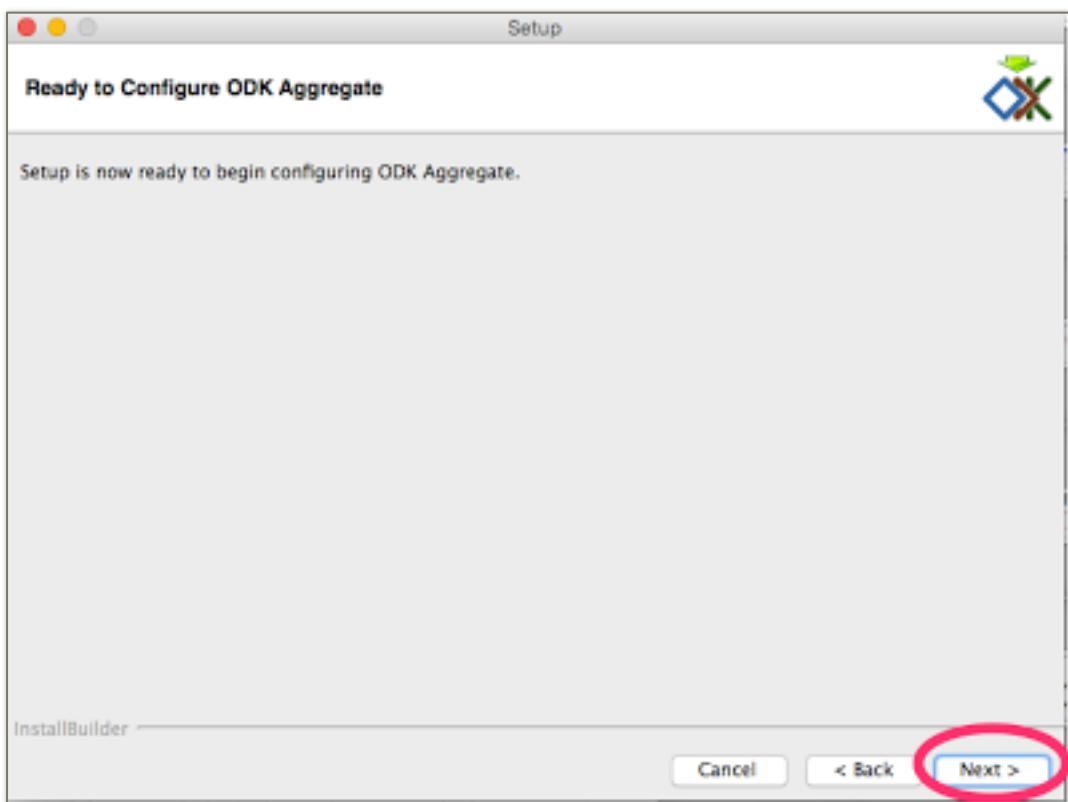
7. 赤丸部分に記載されているプロジェクトIDを確認します。

The screenshot shows the Google Cloud Platform dashboard. On the left sidebar, 'ダッシュボード' is selected. In the center, under 'プロジェクト: My Project', the project ID 'ID myproject-10000 (815-7348018364)' is displayed, with the ID part circled in red. To the right, there are sections for Compute Engine, Google API, and Cloud Storage, each with a 'Get Started' button.

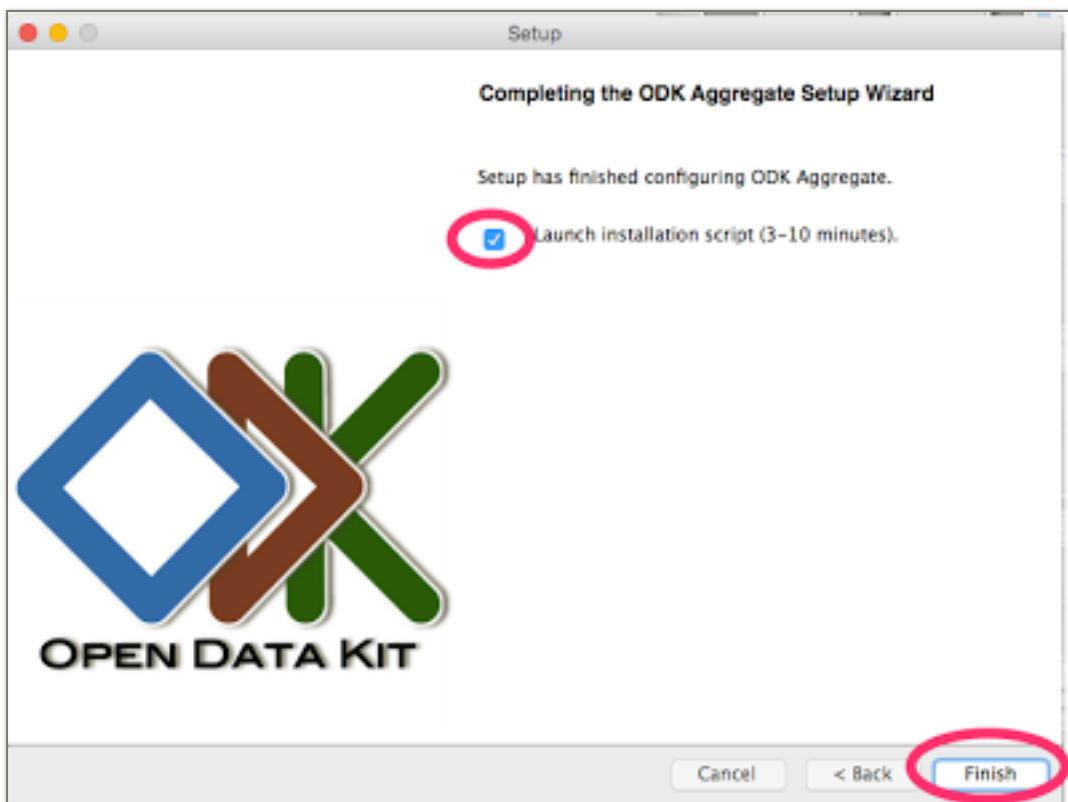
8. プロジェクトIDを入力します。

The screenshot shows a 'Setup' dialog for Google AppEngine. The title bar says 'Setup'. The main section is titled 'Google AppEngine application ID' and contains the following text:  
Before continuing, you should obtain an application identifier in Google's AppEngine cloud services. Click the link below and select the 'Create Application' button on that page to create an application identifier.  
If you are replacing an existing application already on appspot, the application identifier is the leftmost portion of the appspot.com URL. For example, if your application is at 'opendatakit.appspot.com' you would enter 'opendatakit' below.  
<https://appengine.google.com/>  
Please enter your application id:  (This input field is circled in red).  
 Yes, a Master/Slave application id is being (was) migrated to this application id  
At the bottom, it says 'InstallBuilder' and has 'Cancel', '< Back', and 'Next >' buttons.

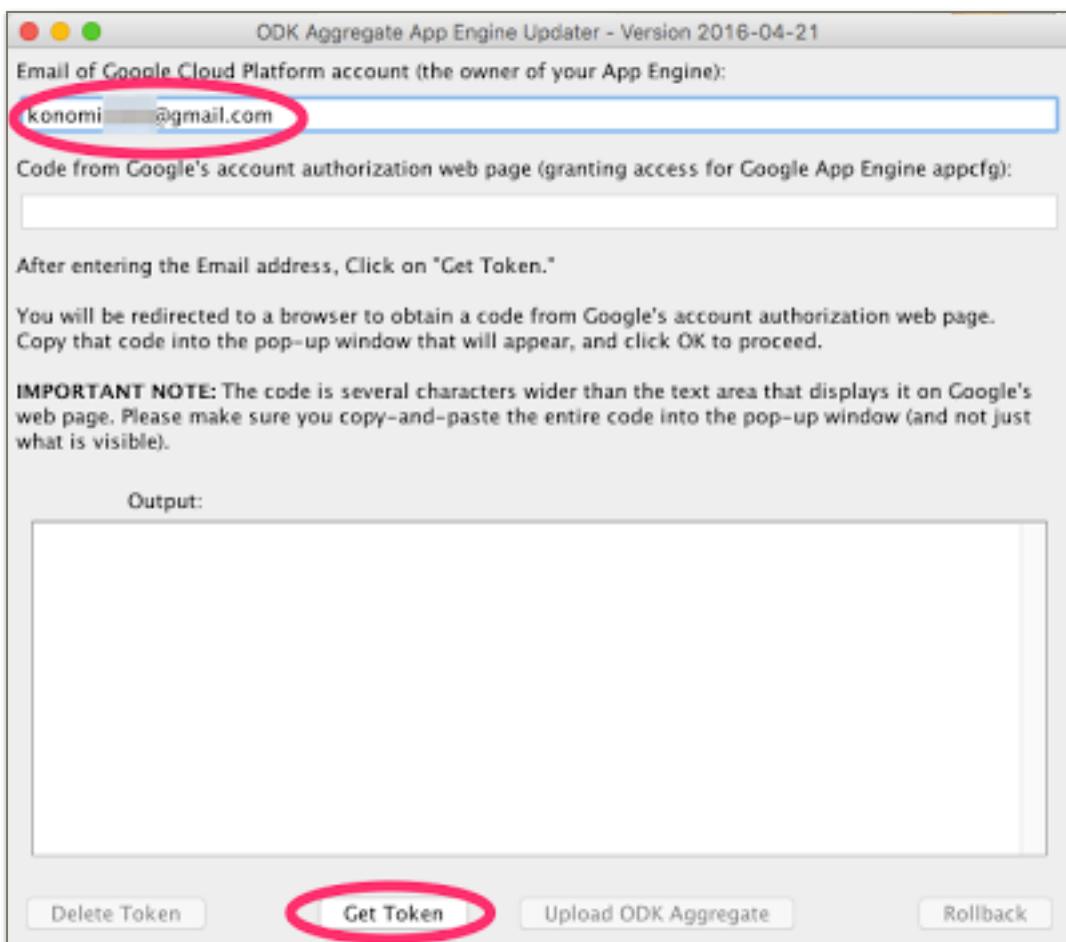
9. 「Next」を押して、次へ進みます。



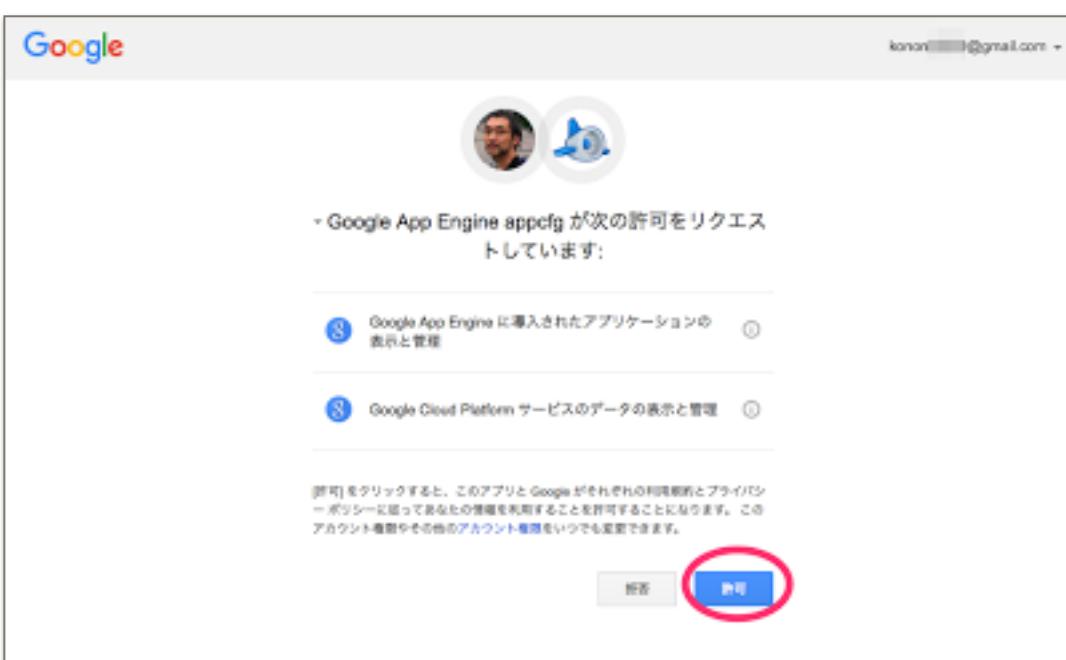
10. 「Finish」を押します。



11. Google クラウドプラットフォームのアカウント情報（Gmailアドレス）を入力し、「Get Token」を押します。



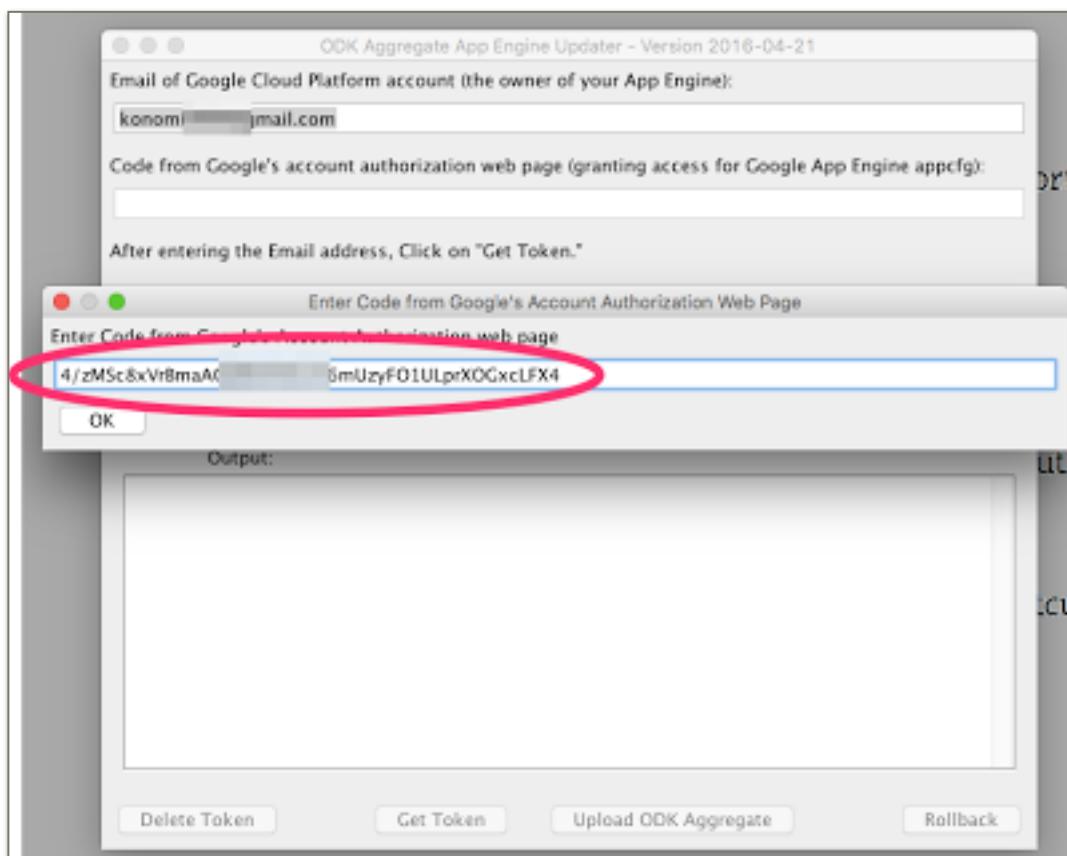
12. Webブラウザ上に、以下のようなアクセス許可を求めるメッセージが表示されます。「許可」を押します。



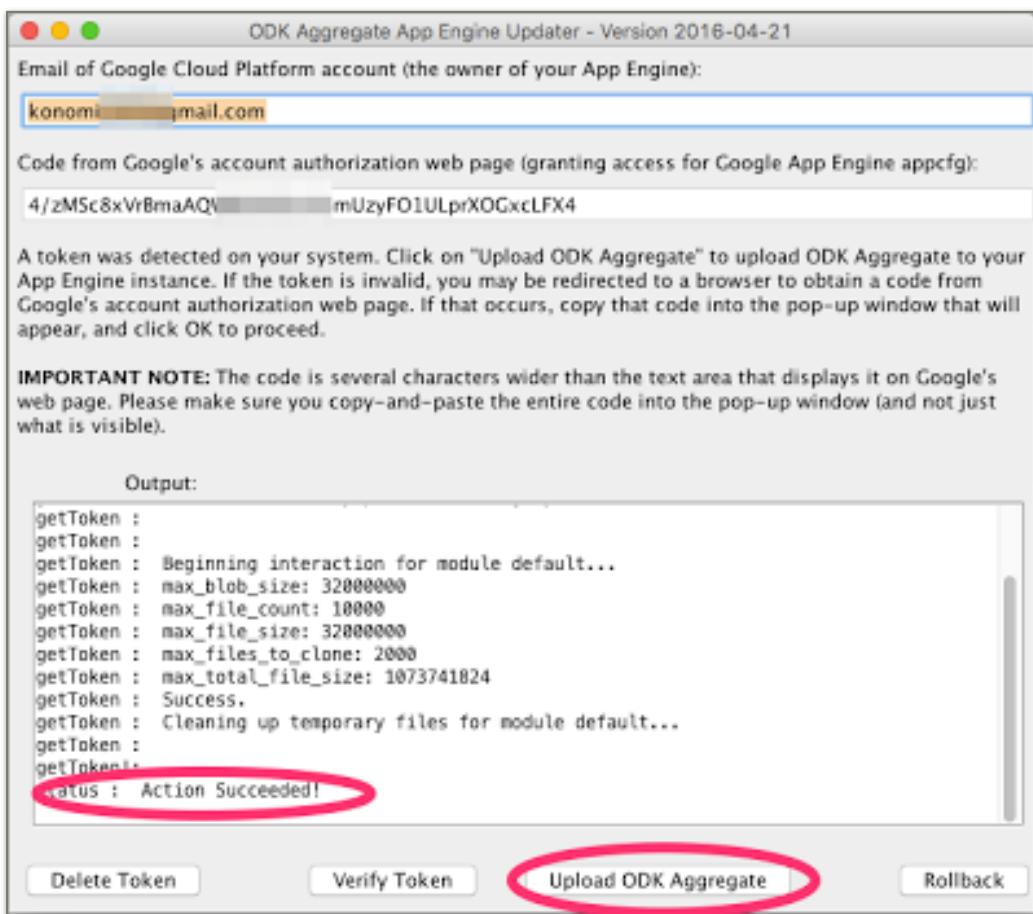
13. コードが表示されますので、コピーします。



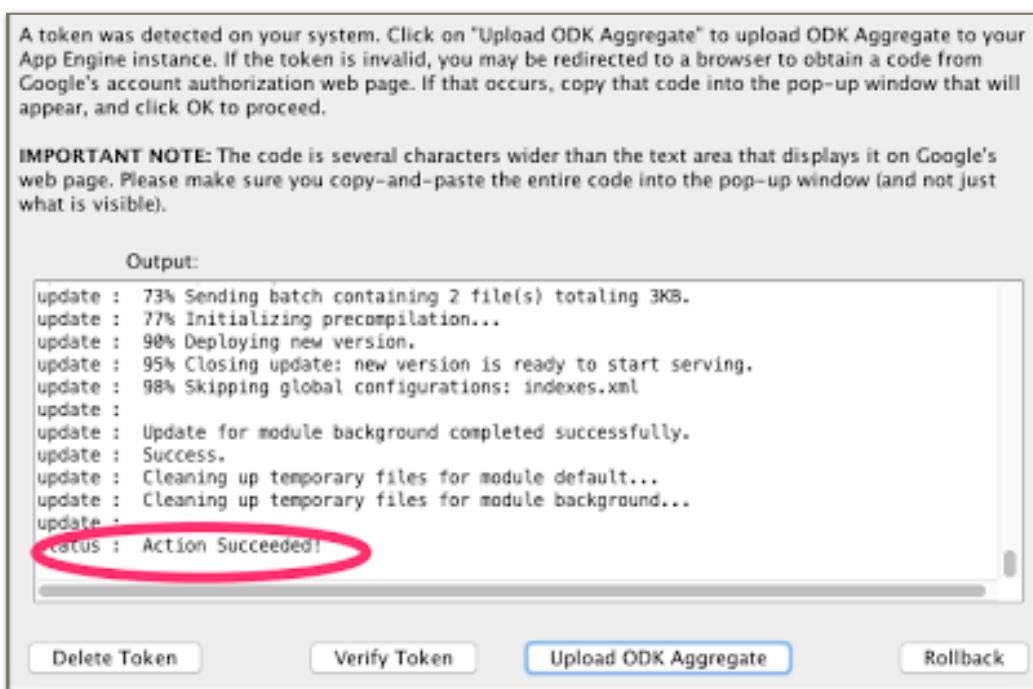
14. コピーしたコードを貼り付けます。



15. 「Upload ODK Aggregate」を押すと、Google クラウドプラットフォームにODK Aggregateをアップロードします。



16. このように「Action Succeeded!」と表示されれば、アップロードが完了したことになります。

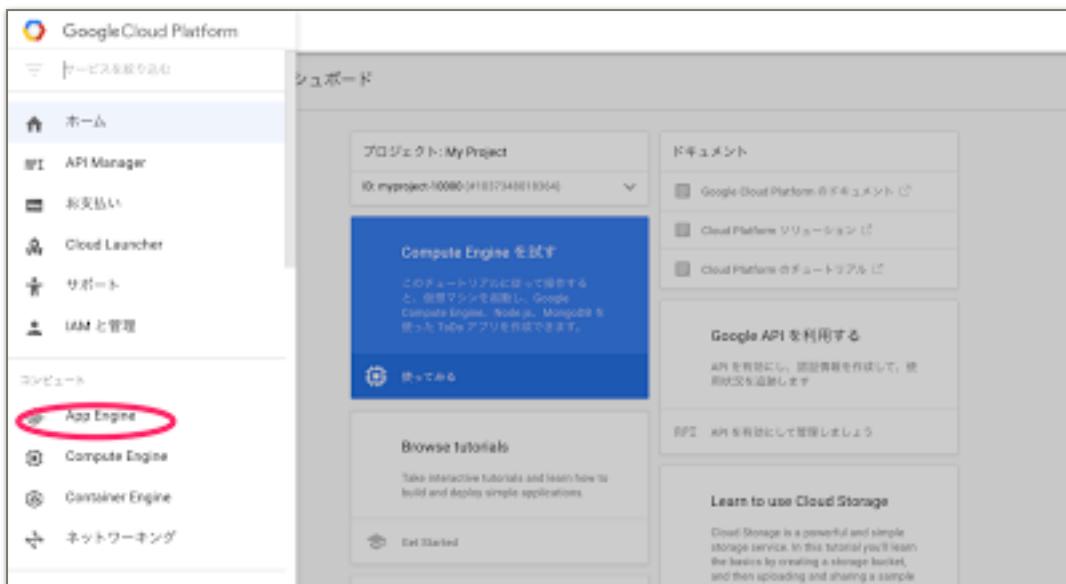


17. Google クラウド プラットフォームに戻り、左上をクリックします。



The screenshot shows the Google Cloud Platform dashboard. On the left, there's a navigation sidebar with icons for Home, Dashboard, and Analytics. The 'Dashboard' option is currently selected and highlighted in blue. At the top left of the main content area, there's a three-line menu icon, which is circled in red. The main content area displays a project summary for 'My Project' (ID: myproject-10080 | #1817348811034) and several quick-start sections: 'Compute Engine を試す', 'Browse tutorials', and 'Learn to use Cloud Storage'. To the right, there are sections for 'ドキュメント' (Documentation) and 'Google API を利用する' (Using Google APIs).

18. 「App Engine」を選択します



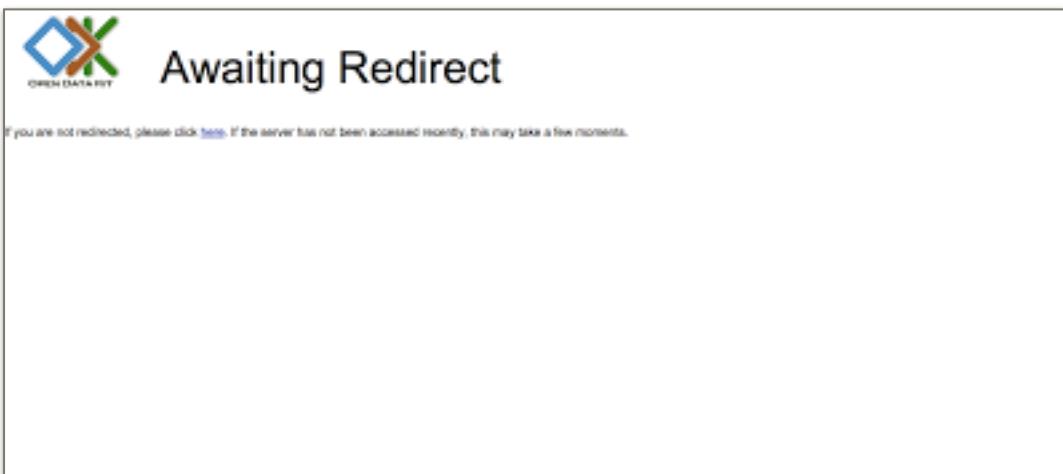
This screenshot shows the same Google Cloud Platform dashboard as the previous one. However, the sidebar has been modified to highlight the 'Compute' section, which contains the 'App Engine' option. This option is circled in red. The rest of the sidebar and the main content area are identical to the first screenshot, showing the 'Dashboard' selected and various quick-start guides for Compute Engine, Cloud Storage, and Google APIs.

19. 右上の赤丸部分をクリックして、サーバ（ODK Aggregate）を起動します<sup>7</sup>。



The screenshot shows the Google Cloud Platform App Engine dashboard. The left sidebar has options like 'サービス', 'バージョン', 'インスタンス', etc. The main area shows a summary with a note 'この期間にリクエストはありません' (No requests in this period). Below it, there's a section for 'インスタンス' (Instances) with a note 'このアプリはインスタンスをデプロイしていません' (This app does not have any instances deployed). At the top right, the URL 'http://project-18008.appspot.com' is displayed, with a red circle drawn around it.

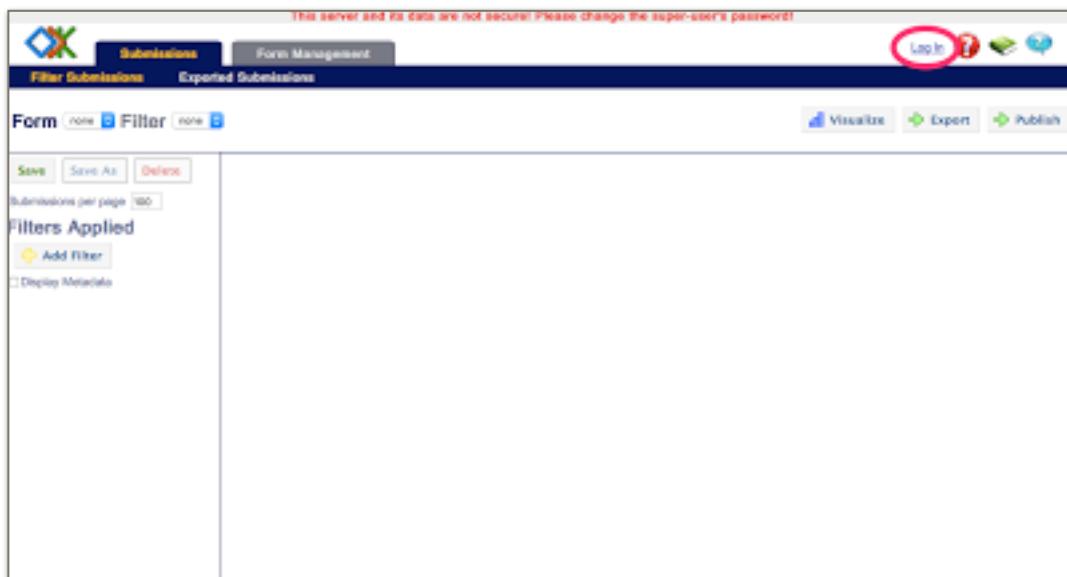
20. しばらくお待ちください。



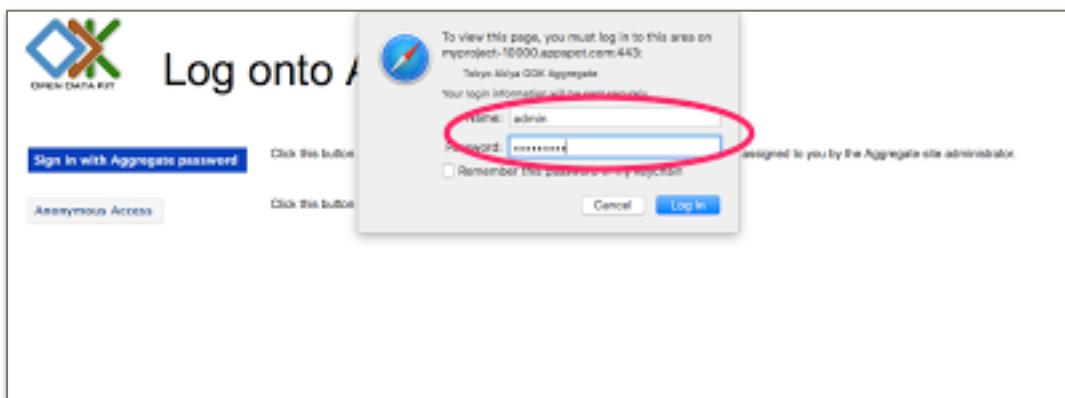
---

<sup>7</sup> クリックする部分にURLが表示されていますが、このURLを控えておいてください。ODK CollectでこのURLを指定すると、サーバと通信できるようになります。ODK Collectの「全般設定」-->「プラットフォーム設定を構成」-->「URL」をタップして設定します。

21. ログインします。



22. ログイン名は、ステップ(6)で「admin」と設定したのであれば、「admin」です。初期パスワードは「aggregate」です。



23. ログインしたら、すぐにパスワードを変更します。

The screenshot shows the 'Configure Site Access' page in the ODK Aggregate interface. At the top, there are tabs for Submissions, Form Management, and Site Admin, with Site Admin being the active tab and highlighted with a red circle. The main content area is titled 'Configure Site Access' and contains sections for 'Users access the site either:' and 'Capabilities are as follows:'. Under 'Users access the site either:', there is a list of three options: 'anonymously (i.e., as the anonymousUser)', 'via an OAuth 2.0 token (requires an Email account (e.g., user@gmail.com) and an OAuth 2.0 token issued by Google with access to userInfoEmail)', and 'via an ODK account, with a username and password that a site administrator has configured for them'. Under 'Capabilities are as follows:', there is a detailed list of permissions for different roles: Data Collector, Data Viewer, Form Manager, Synchronization Tables, Administrator Tables, and Site Administrator. Below this, there is a section titled 'Edit Users' with a table to manage user accounts. The table has columns for Username, Full Name, Account Type, Data Collector, Data Viewer, Form Manager, and Site Administrator. A row for the user 'admin' is shown, with a 'Change Password' link highlighted with a red circle. The right side of the interface includes standard navigation icons.

サーバの設定が完了しました。お疲れ様でした!

このウェブページを使って、データ収集用の入力フォームを登録します。登録した入力フォームを使って入力したデータは全てサーバに蓄積されます。また、このウェブページを使って、サーバに集まったデータを見たり、手元のパソコンなどにデータをダウンロードしたり、データを地図上に表示したり、不要なデータを削除したりすることができます。サーバにアクセスするユーザを管理することもできます。

右上のアイコンをクリックすると、説明が表示されますので、そちらもご覧ください。

サーバの導入にあたって<sup>8</sup>

ODK Aggregateは、GoogleのApp Engine、AmazonのEC2クラウドサービス、あるいはWebサーバ（TomcatとMySQL（またはPostgreSQL）が必要）上に導入・運用すること

<sup>8</sup> 本章は次のURLの内容に基づきます：<https://opendatakit.org/use/aggregate/deployment-planning/>

ができます。初心者の方には、GoogleのApp Engine（あるいはODK AggregateのVM）を使うことをお薦めします。慣れてきたら、AWS EC2やTomcat、その他のクラウドサーバの利用も検討してください。

ODK Collectアプリを用いて、ODK Aggregateサーバからフォームをダウンロードしたら、フォームにデータを入力し、確定したフォームをODK Aggregateサーバにアップロードし、ODK Aggregateサーバ上でデータの表示やエキスポート、公開を行うことができます。また、ODK Briefcaseというソフトウェアを用いて、ODK Aggregateサーバからデータをダウンロードし、手元のパソコン上でデータを表示、変換、分析、公開することもできます。

## 導入の際の検討事項

Google のクラウドサービスを用いるべきかどうかを判断するときに、以下の点について検討した方が良いかもしれません：

1. インターネットアクセス – Google App Engineを利用するにはインターネットにアクセスできる必要があります。もし、インターネットにアクセスできる環境になければ、ODK Briefcaseを使うことを検討してください。インターネットアクセスがない環境でTomcatサーバを使うことも可能ですが、その場合、ODK Collectアプリで収集したデータをアップロードするには、Tomcatサーバのあるネットワークに何らかの方法で接続する必要があります。
2. コンピュータのスキル – Tomcatサーバを使う方法は技術的にやや敷居が高いかもしれません（ある程度技術的な素養が必要です。）最低限やらなければならないこととしては(1) ネットワークの構成を変更する、(2) ウェブホスティングサービスを選定し設定する、あるいは自前のサーバとネットワークルータを選定し設定する、(3) サーバのソフトウェアをインストールする、(4) 利用するサーバ環境において適切な停電・バックアップ対策が施されているか確認し、施されていなければ対策を行う、などがあります。
3. サポート体制 – Tomcat サーバを使う場合、定期的にデータをバックアップすることが不可欠です。もし、データのセキュリティについて懸念がある場合は、システム／データベース管理者を短時間雇用するなどして、システムのログを定期的にレビューし、悪意のある行動が見られないかチェックする必要があります。

4. 可用性 – ODK Collectアプリにダウンロードした空のフォームは、調査期間中そのまま更新せずに使うことが多いでしょう。また、調査員は確定したフォームを適宜ODK Aggregateサーバにアップロードすることが多いでしょう。そうした場合であれば、Google App Engineに匹敵するような、可用性の高いサーバ環境は不要かもしれません。Tomcatサーバを使う場合、Google App Engine並みの可用性を実現することには多大なコストがかかります（あなたの組織が既にIT部門を持っていれば話は別かもしれません）。Tomcatサーバを使う場合、サーバが停止することなくいつも利用することのできる、可用性の高い環境を構築するには、大きな金銭的投資が必要になるでしょう。
5. データセットのサイズ – Google App Engineは、「無限に近い」量のデータを蓄積することができます（アップロードデータ数が100万以上になっても大丈夫です）。しかし、Google App EngineにしてもTomcatサーバにしても、データセットのサイズが大きい場合（例えば目安としてアップロードデータ数7000以上）は、サーバをカスタマイズ（バックグラウンドプロセスを実行するサーバのサイズを大きくし、より大きな仮想マシンを用いるなど）しなければ、データをCSV/KML/JSONファイルにエキスポートする機能がうまく動作しなくなります。Google App Engineにおいてはサーバを大きくするほど利用料金が高額になります。また、データセットが非常に大きい（例えばデータ数100,000以上）時は、MySQLデータベースやPostgreSQLデータベースによるシステム環境の方が処理性能が高くなる可能性があります。つまり、自前のサーバにデータベースサーバを導入した方が、Googleクラウドサービスを用いる場合よりもカスタマイズや最適化を行いやすいでしょう。
6. データのローカル性とセキュリティ – Google App Engineサーバの実体は、地球上のどこかにあるのですが、自分のデータが具体的にどこに保存されているか分かりにくくなっています。データのローカル性の保証や予防的なセキュリティ対策を行いたい場合、そのような分かりにくさのあるサーバ環境はあまり好ましくないでしょう。慎重な扱いが必要なデータを管理する場合や、国や組織において各種の規則や制限が設けられている場合、データのローカル性やセキュリティを考慮することは重要です。ただし、比較的厳しい規則や制限が設けられている場合でも、暗号化されたフォームを使うことによって、コンプライアンスを達成できる可能性はあります。Google App Engineにデータを保存する前に、関連する規則や制限事項について調べましょう。また、Google App Engine のサービス利用規約も是非参照して下さい。

7. 課金 – Google App Engineは、24時間単位の作業クオータを採用しています。一般に、調査を試行する場合や、小規模の調査を行う場合には、Google App Engineをオープンデータキットのサーバとして無料で利用することができます。もし、2000件以下のデータを集めることが目的で、24時間ではなく日中のみサーバにアクセスし、データのアップロードやダウンロードを柔軟なタイミングで行うことができるのであれば、Google App Engineを本格的な調査においても利用することができます。そうでない場合は、有料サービスの設定を行う必要が生じるでしょう。

Google App Engineのコストパフォーマンスについて少し補足します。 <http://opendatakit.appspot.com> のサーバは、通常は利用量が少ないとおり、Googleへの支払額は最低の課金額である週2.10米ドルを少し超える程度です。Open Data Kitの開発者・関係者が2011年9月から2012年12月にかけてムンバイで実施したデータ収集・調査では、最低の課金額である週2.10米ドルの支払いだけで済んだということです。なお、週単位で必要に応じて課金のオン・オフを行うことができますので、データ収集をしばらく行わない期間がある場合は、課金をオフにします。課金をオフにしていると、毎日の無料利用可能範囲を超えるとアクセスが制限されます。

## *Google App Engineの利用料金について*

利用料金を最小化するには、ODK Aggregateサーバのウェブサイトに不要なアクセスをしないことが重要です。特にODK Aggregateサーバのウェブサイト上で「submission」タブを表示すると非常に多くの通信が発生しますので、無料利用可能範囲をすぐに超えてしまう可能性があります。

収集したデータにアクセスする場合、ODK Aggregateのウェブサイトを使うよりも、ODK Briefcaseを使ってCSVファイルをダウンロードする方が、利用料金を節約することができます。

ODK Aggregateサーバに全てのデータを置いたままにしても良いですし、古いデータを逐次サーバから消していく構いません。データをサーバから手元のパソコンにダウンロードし、またサーバにアップロードするような場合には前者の方が効率よくなる可能性があります。データをサーバから手元のパソコンにダウンロードして利用するだけであれば、場合によっては後者を行なうのが良いかもしれません。

サーバからFusion Tableなどにデータを読み込んでFusion Table上で閲覧や分析を行ったりCSVをダウンロードするというような、間接的な方法を用いると、サーバの利用コストをより小さくすることができます。

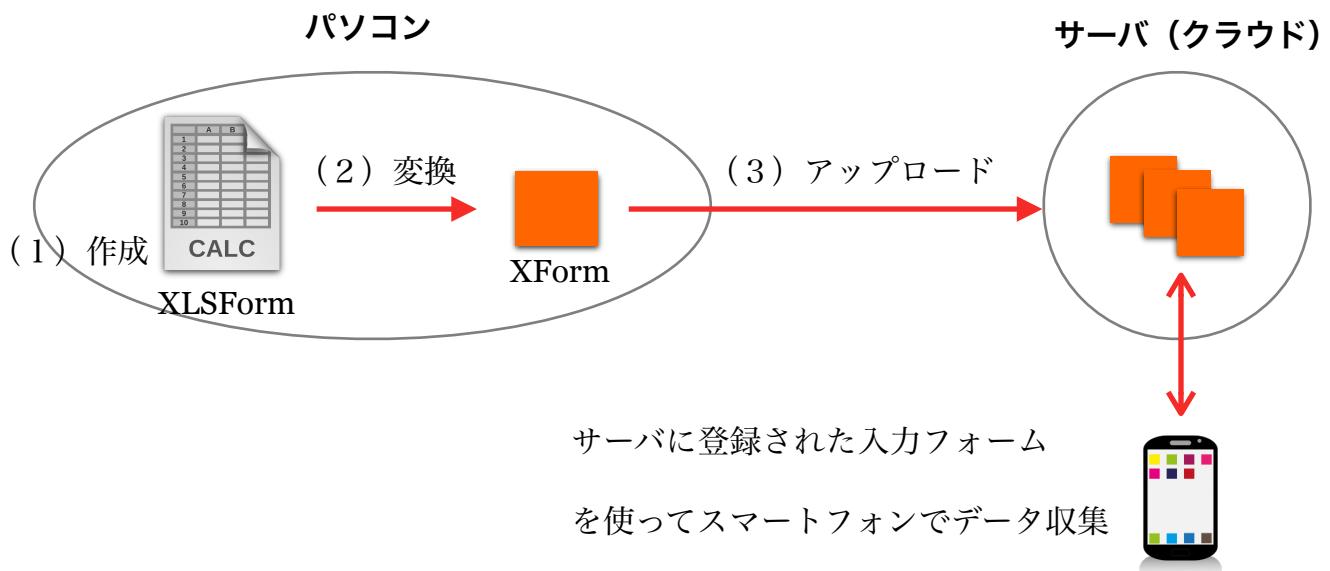
## 第3章

# 入力フォームの作成

本章では、入力フォームを作成し、サーバに登録する方法を説明します。

オープンデータキットでは、Excelなどの表計算ソフトを使って入力フォームを手軽に作成することができます。作成手順の概要は以下のとおりです：

- (1) 表計算ソフトを使って、集めたいデータ項目などを指定したテーブルを作成し、ファイルに保存します。このファイルのことをXLSFormと言います。
- (2) XLSFormファイルを変換して、XFormファイルというものを生成します。
- (3) XFormファイルをサーバにアップロードします。



入力フォーム作成・登録の流れ

## 簡単な入力フォームの作成

実際に簡単な入力フォームを作ってみましょう。以下に方法を説明しますので、是非やってみてください。

## (1) XLSFormファイルを作成する

年齢と名前を入力するフォームを作成してみましょう。

こんなのを作ります：



まず、Excelを起動して空のExcelブックを作成します（あるいはOpenOfficeやLibreOfficeなど、.xls,.xlsx形式でファイルを保存できる表計算ソフトならなんでもOKです）。ワークシートの名前を `survey` に変更します。

以下の通り入力してください：

	A	B	C	D
1	type	name	label	
2	text	name	名前を入力してください	
3	integer	age	年齢を入力してください	
4				
5				
6				
7				

1行目は列の名前(type, name, label)です。

2行目が最初の質問の定義です。「名前を入力してください」というメッセージを出して、テキスト(text)で回答してもらう質問です。

3行目が2番目の質問の定義です。「年齢を入力してください」というメッセージを出して、整数(integer)で回答してもらう質問です。

helloworld.xls (あるいはhelloworld.xlsx) というファイル名をつけて保存してください。お手本が以下のURLに置いてあります。

[https://drive.google.com/file/d/oB-Tqnx-A\\_Ee8aC1sVDNjUHg2c2M/view](https://drive.google.com/file/d/oB-Tqnx-A_Ee8aC1sVDNjUHg2c2M/view)

## (2) XLSFormファイルをXFormファイルに変換する

以下のURLにアクセスして、作成したXLSFormファイルをアップロードします。

<http://opendatakit.org/xiframe/>



「Download」ボタンを押してXMLファイルをダウンロードします



### (3) XFormファイルをサーバにアップロードする<sup>9</sup>

「Form Management」タブを選択し、「Add New Form」を押します。

The screenshot shows the 'Form Management' tab selected in the navigation bar. Below it is a table titled 'Forms List' with columns: Title, Form Id, Media files, User, Downloadable, and Accept S. The table lists several forms: test\_form, test10, test11, test12, test13, test14, and test15. A red arrow points to the 'Add New Form' button at the top left of the table area, with the text '(2)クリックするとXMLファイルをアップロードするための画面が表示される' (Clicking here displays the screen for uploading XML files). Another red arrow points to the 'Form Management' tab itself, with the text '(1)タブを選択' (Select the tab).

Forms List					
Submissions		Form Management			
Title	Form Id	Media files	User	Downloadable	Accept S
test_form	build_test-form_1467386720	0	admin	<input checked="" type="checkbox"/>	
test10	test10	0	anonymousUser	<input checked="" type="checkbox"/>	
test11	test11	0	anonymousUser	<input checked="" type="checkbox"/>	
test12	test12	0	anonymousUser	<input checked="" type="checkbox"/>	
test13	test13	0	anonymousUser	<input checked="" type="checkbox"/>	
test14	test14	0	anonymousUser	<input checked="" type="checkbox"/>	
test15	test15	0	anonymousUser	<input checked="" type="checkbox"/>	

<sup>9</sup> もし、画像や音声、ビデオなどのビデオファイルやCSVファイルをアップロードする場合は、ここで同時にアップロードすることができます。

XFormファイルをアップロードします。「Choose File」をクリックし、XFormファイルを指定します。次に、「Upload Form」ボタンを押すと、アップロードが始まります。なお、専門的な話になりますが、XFormファイルはxmlと呼ばれる言語で記述されています。下の図のxmlとは、XFormファイルのことを指しています。



無事アップロードできれば完成です！名前と年齢を入力するフォームを作成し、サーバに登録する作業を完了しました。

## 作成した入力フォームを使ってみる

1. ODK Collectアプリを起動して、メインメニューで「空のフォームを取得する」を選択し、helloworldを取得してください。次にメインメニューに戻って「空のフォームに入力」を選択し、helloworldを起動します。
2. 左にスワイプして、データを入力していきます。
3. 全て入力したら「フォームを保存して終了」をタップしてメインメニューに戻ります。
4. 「確定剤のフォームを送信」をタップして、チェックを入れ、「選択済を送信」をタップします。ODK Aggregateサーバにデータがアップロードされます。
5. アップロードしたデータは、ODK Aggregateで、次のようにして確認してください。

(1) タブを選択

(2) helloworldを選択

(3) 端末からアップロードされたデータが表示される

	name	age	meta
	instanceID		
	山田太郎	35	uuid:6bd3ba11-8bcf-40ea-9ed3-866ee978af79

「Submissions」タブを選択し、helloworldを選択し、端末からアップロードされたデータが表示されることを確認します。

XLSFormは機能豊富で、使いこなせば色々なことができます。例えば、位置情報やマルチメディア、多言語にも対応しており、お絵描きによる入力や、サインの入力もできます。

以下のURLには、もう少し複雑なXLSFormの例が置いてあります。ダウンロードして、参考にしてください。

<https://opendatakit.org/wp-content/uploads/2013/06/sample.xlsform.xls>

XLSFormを使いこなして、使いやすい入力フォームをデザインするには、第4章「XLSForm」へと読み進んでください。

## 参考情報

- 作成されたフォーム(XMLファイル) をサーバ(ODK Aggregate)にアップロードせずに使うこともできます。その場合、AndroidデバイスのSDカードの /odk/forms フォルダにコピーし、ODK Collectに直接追加します。

- XLSFormは、ODK Collect version 1.1.7もしくはそれよりも新しいバージョンに対応しています。
- XLSFormをアップロードして変換するときにenketo (<https://enketo.org/>) で確認（プレビュー）できますが、画像や音声は表示されません。
- XLSFormをXFormに変換する方法は、既に紹介した方法 (<http://opendatakit.org/xiframe/>) 以外にもいくつかあります。
  1. Windows用の変換ツール（<https://opendatakit.org/downloads/download-info/xlsform-for-windows/>、インターネット接続なしで使えます。）エラーレポートを作成するために、ODK Validate (<https://opendatakit.org/downloads/download-info/odk-validate-2/>) もダウンロードしてください。
  2. NafundiのWin/Mac用ツール（<https://gumroad.com/l/xlsform-offline>、インターネット接続なしで使えます。）
  3. Pythonライブラリpyxform（<https://github.com/uw-ictd/pyxform>、インターネット接続なしで使えます。）
  4. Formhub.org（<https://formhub.org/>、サポート終了）、Ona.io（<https://ona.io/home/>）（これらを使うにはインターネット接続が必要です。）
- XLSFormを使わずに入力フォームを作成する方法があります。Build (<https://opendatakit.org/use/build/>)、Kobo (<http://www.kobotoolbox.org/>)、Enketo (<https://enketo.org/>)、PurcForms (<https://code.google.com/archive/p/purcforms/>)、Vellum (<http://xforms.dimagi.com/>)などを試してみてください。XML言語をご存知の方は、直接XFormファイルを編集してフォームを作成することもできます。

## 第4章

# XLSForm<sup>10</sup>

XLSFormとは、Excelなどの表計算ソフトを使って手軽に入力フォームを作成することができる環境です。

XLSFormを使えば、単純な入力フォームなら、初心者でもすぐに作成することができます。以下に説明する内容に慣れ親しめば、高度な入力フォームを作成することもできます<sup>11</sup>。

オープンデータキット以外にも、XLSFormに対応したデータ収集プラットフォームがあります。

## 基本フォーマット

2つのワークシートを持つExcelファイルを作成します。これらのワークシートの名前は、surveyとchoicesです。なお、settingsという名前のワークシートも用いる場合がありますが、通常はsettingsワークシートは省略しても構いません。

surveyワークシートには、質問の内容や表示方法などを記入します。基本的には一つの質問が一つの行に対応しています。

choicesワークシートには、選択肢を用いた質問の、選択肢の内容を記入します。一つの列が一つの選択肢に対応しています。

両ワークシートには、必ず何か記入しなければならない必須の列があります。その他の列は空欄にしておいても構いません。

surveyワークシートには、3つの必須の列(type, name, label)があります。

- type列には、追加する項目の型を記入します

---

<sup>10</sup> 本章の内容は、XLSForm.orgに基づくものです。

<sup>11</sup> XLSFormで作成できる様々なフォームの例：<https://opendatakit.org/help/form-design/examples/>

- name列には、項目の名前を記入します。すべての項目に違う名前をつける必要があります。
- label列には、入力フォームに表示されるテキスト（質問の内容など）を記入します。

choicesワークシートには、3つの必須の列(list name, name, label)があります。

- list name列には、複数の選択肢をまとめたための名前を記入します。
- name列には、選択肢の名前を記入します。
- label列には、入力フォームに表示されるテキストを記入します。

ワークシート中、列の順番は自由です。また、空白の行があっても大丈夫です。色や罫線の太さ、フォントの大きさなどを変えても影響しませんので、ご自由にどうぞ。

Excelで入力フォームを作成するときには、正確に記入しなければならないことに注意してください。例えば、choicesではなく、Choicesやchoiceと入力してしまうと、うまくいきません。

## 質問の種類

XLSFormで使用することのできる質問の種類を以下の表に示します。これら以外の種類の質問も扱うことができますが、表に示すものが代表的です。なお、質問の種類は、surveyワークシートのtype列で指定します。

質問の種類	入力される回答
integer	整数
decimal	数値
text	テキストによる自由記述
select_one [オプション]	複数選択肢の質問。一つだけの答えを選択することができます
select_multiple [オプション]	複数選択肢の質問。複数の回答を選択することができます
note	テキスト情報を表示します。入力なし
geopoint	GPS座標
geotrace	複数のGPS座標からなる線のデータ
geoshape	複数のGPS座標からなる面（ポリゴン）のデータ。最後の座標は、最初の座標と同じです
date	日付
time	時間
dateTime	日付と時刻
image	写真
audio	音を録音
video	ビデオを録画
barcode	バーコードの読み取り。バーコード読み取りアプリをインストールする必要がある
calculate	計算を実行します
acknowledge	確認を促し、反応があれば「OK」が記録される

## GPS

例えば、店舗名とGPS座標を収集するには、次のように記述します。

survey			
	type	name	label
	text	store_name	この店の名前は何ですか？
	geopoint	store_gps	この店のGPS座標を収集します。

## GPS（位置精度のしきい値を指定する場合）

ODK Collectは通常GPSの精度が5m以下のときだけ位置情報を記録します。これを5m以外の値に変更することができます。surveyワークシートに、body::accuracyThresholdという列を追加し、好きな値を指定します。単位はメートルです。以下に例を示します。

survey				
	type	name	label	body::accuracyThreshold
	geopoint	store_gps	この店のGPS座標を収集します。	1.5

## 選択肢を用いる質問

单一回答の場合select\_oneを、複数回答の場合select\_multipleを使います。また、選択肢の内容をchoicesワークシートに記入します。

单一回答の場合（select\_one）の例を示します。

survey			
	type	name	label
	select_one yes_no	likes_pizza	ピザは好きですか？
choices			
	list name	name	label
	yes_no	yes	はい
	yes_no	no	いいえ

surveyワークシートのyes\_noは、choicesワークシートのlist nameのyes\_noと一致しなければならないことに注意してください。質間に該当する選択肢が正しく表示されるようになるには、これらが一致していかなければなりません。

次に、複数の回答を選択して回答する場合（select\_multiple）の例を示します。

<b>survey</b>			
	type	name	label
	select_multiple pizza_toppings	favorite_toppings	あなたの好きなピザのトッピングは何ですか？
<b>choices</b>			
	list name	name	label
	pizza_toppings	cheese	チーズ
	pizza_toppings	pepperoni	ペパロニ
	pizza_toppings	sausage	ソーセージ

## 「その他」の指定

選択肢を用いる質問では、「その他」という選択肢を設けることがよくあります。「その他」に該当する場合、具体的な内容を記入してもらうこともあります。このような入力をしてもらいたい時には、surveyワークシート中で、or\_otherを指定します。

<b>survey</b>			
	type	name	label
	select_multiple pizza_toppings or_other	favorite_topping	あなたの好きなピザのトッピングは何ですか？
<b>choices</b>			
	list name	name	label
	pizza_toppings	cheese	チーズ
	pizza_toppings	pepperoni	ペパロニ
	pizza_toppings	sausage	ソーセージ

\*or\_otherを使用してデータを収集すると、favorite\_topping欄に、otherという値が記録され、別の欄に実際の回答が記録されます。このため、特に大量にデータを集める場合には、分析作業がやや面倒になる可能性があります。大量にデータを集める場合は、or\_otherではなくrelevantを用いる方が良いかもしれません。relevantについては後ほど説明します。

## メタデータ

時刻や端末情報などのメタデータを自動的に収集することもできます。具体的には下記のメタデータを収集することができます。

メタデータの種類	意味
start	調査の開始日時
end	調査の終了日時
today	調査日
deviceid	IMEI（国際移動体装置識別番号）
subscriberid	IMSI（国際移動加入者識別番号）
simserial	SIMのシリアル番号
phononenumber	電話番号（利用可能な場合）

これらのお部は、電話機能を持つ端末でしか収集できないので注意してください。

これらすべてのメタデータを収集したいなら、surveyワークシートの冒頭に次のように記入します。

survey			
	type	name	label
	start	start	
	end	end	
	today	today	
	deviceid	deviceid	
	subscriberid	subscriberid	
	simserial	simserial	
	phononenumber	phononenumber	

メタデータは自動的に収集されるので、質問などは表示されません。従って、labelを記入する必要はありません。

## ヒント

質問への答え方を回答者に説明したい（けれども質問本文には説明を書きたくない）場合、hint列を使います。

survey				
	type	name	label	hint
	text	name	この店の名前は何ですか？	店に看板がある場合は、看板を確認してください。
	geopoint	geopoint	この店のGPS座標を収集します。	

## 制約条件

データ品質を保証する一つの方法は、入力フォームの項目に制約条件を追加することです。例えば、人の年齢を尋ねたときに、-22または200のような誤ったデータが入力されることを避けるために、制約条件を使うことができます。制約条件を追加するのは簡単で、constraints列に条件を記入するだけです。以下の例では、年齢は150を超えてはならないという条件を記入しています。条件中の.は質問変数を参照しています。

survey				
	type	name	label	constraint
	integer	age	あなたは何歳ですか？	. <= 150

こうしておくと、例えばユーザが151を入力した場合は、次の質間に移動することや、保存・終了することができません。

条件中で使うことができる演算子については、以下のURLのOperatorsの欄をご覧ください。

<https://opendatakit.org/help/form-design/binding/>

## 制約のメッセージ

制約によって回答が受け付けられなかった場合に、その理由を説明するには、constraint\_message列に記入します。

survey					
	type	name	label	constraint	constraint_message
	integer	respondent_age	回答者の年齢	. >= 18	回答者は18歳以上でなければなりません

ユーザーが18未満の年齢を入力した場合、constraint\_message列で指定したメッセージが表示されます。

## 関連制御 (Relevant)

relevant列を使えば、質問の回答に基づいて、次の質問を表示したりしなかったりすることができます。例えば、ピザのトッピングについて単一選択(select\_one)のアンケートを行う場合、次のようにしてrelevantを利用することができます。

survey				
	type	name	label	relevant
	select_one yes_no	likes_pizza	ピザは好きですか？	
	select_multiple pizza_toppings or_other	favorite_topping	好きなトッピング	\$ {likes_pizza} = 'yes'

「ピザは好きですか？」の回答がyesである場合のみ、「好きなトッピング」を尋ねる質問が表示されます。likes\_pizzaという変数が \${ } で囲まれていますが、前の質問の変数を参照するためにこの記号を用いています。

複数選択(select\_multiple)のアンケートを行う場合は、次のようにしてrelevantを利用することができます。

「好きなトッピング」は複数選択により回答しますが、cheeseが回答の一つとして選ばれた場合は（他の選択肢が選ばれたかどうかにはかかわらず）必ず好きなチーズの種類を質問したいので、selected(\${favorite\_topping}, 'cheese')のように指定します。

survey				
	type	name	label	relevant
	select_one yes_no	likes_pizza	ピザは好きですか？	
	select_multiple pizza_toppings or_other	favorite_topping	好きなトッピング	<code>\${likes_pizza} = 'yes'</code>
	text	favorite_cheese	好きなチーズの種類	<code>selected(\${favorite_topping}, 'cheese')</code>
choices				
	list name	name	label	
	pizza_toppings	cheese	チーズ	
	pizza_toppings	pepperoni	ペパロニ	
	pizza_toppings	sausage	ソーセージ	

「その他」の説明で、大量にデータを集める場合はor\_otherではなくrelevantを用いる方が良いだろうと述べました。or\_otherの代わりにrelevantを用いるには、以下のようにします。

survey				
	type	name	label	relevant
	select_multiple pizza_toppings	favorite_toppings	好きなピザのトッピングは何ですか？	
	text	favorite_toppings_other	他の指定：	<code>selected(\${favorite_toppings}, 'other')</code>
choices				
	list name	name	label	
	pizza_toppings	cheese	チーズ	
	pizza_toppings	pepperoni	ペパロニ	
	pizza_toppings	sausage	ソーセージ	
	pizza_toppings	other	その他	

choicesワークシートの選択肢にotherを入れる必要があることに注意してください。

## 数式

制約条件(constraint)列、関連制御(relevant)列、計算(calculation)列で数式を利用するすることができます。実はすでに、制約条件と関連制御の例題の中で、数式を使いました。数式は、関数や演算子(+, \*, div, など)を使って記述することができます。使うことができる演算子のリストは以下のURLにあります。

[https://www.w3schools.com/xml/xpath\\_operators.asp](https://www.w3schools.com/xml/xpath_operators.asp)

## 計算

先行する質問のデータを用いて計算を行うには「calculate」を用います。例えば、以下のように指定すれば、食事のチップを計算して表示することができます。

survey				
	type	name	label	calculation
	decimal	amount	靴の値段はいくらでしたか？	
	calculate	tax		$\$ \{ amount \} * 0.08$
	note	display	8%の消費税は $\$ \{ tax \}$ です	

入力フォームを利用するとき、最下行の $\$ \{ tax \}$ は実際の消費税額に置き換えられます。

## 回答必須

必ず回答しなければならない質問を指定する方法を説明します。回答必須の質間に答えないで次の質間に進むこと（あるいはフォームを提出すること）はできません。

質問を回答必須にするには、required列をワークシートに追加し、回答必須の質問に対して yesを指定します。

survey					
	type	name	label	constraint	required
	integer	age	あなたは何歳ですか？	. <= 150	yes

## 回答必須メッセージ

回答必須項目が空欄になっている場合に表示するメッセージを指定することができます。  
required\_message列をsurveyワークシートに追加し、メッセージの内容を指定します。  
以下の例をご覧ください。

survey					
	type	name	label	required	required_message
	integer	respondent_age	回答者の年齢	yes	申し訳ありませんが、この項目は回答が必須です。

## グループ作成

以下のようにして、質問のグループを作成することができます。

survey			
	type	name	label
	begin group	respondent	回答者
	text	name	回答者の名前を記入してください
	text	position	回答者の学内における役職を記入してください
	end group		

こうしたグループを作成しておくと、データの書き出しや分析を行うときに便利です。  
end groupはフォームに表示されないので、nameもlabelも不要です。

## 入れ子グループ

グループ中にグループを作成することができます。

begin groupとend groupの対応関係を赤と青で示していますが、このように、最後にbeginしたグループから順にendしていく必要があります。全てのbegin groupは、対応するend groupを持たなければなりません。そうでないと、フォームをアップロードした際に、正常に処理することができず、エラーメッセージが表示されます。

survey			
	type	name	label
	begin group	hospital	病院
	text	名	この病院の名前は何ですか？
	begin group	habu_medication	ハブの血清
	select_one yes_no	have_habu_medication	この病院にはハブの血清がありますか？
	end group		
	end group		

## スキップ<sup>°</sup>

グループ単位で質問をスキップするには、begin groupのrelevant列に次のようにして条件を記入します。このような巧妙なことができるのが、XLSFormの特徴です。

type	name	label	relevant
integer	age	何歳ですか？	
begin group	child	子供	$\$age \leq 5$
integer	MUAC	子供の上腕周囲径を記録します	
select_one yes_no	mrdr	子供の迅速診断テストの結果は陽性ですか？	
end group			

この例では、下位グループの2つの質問(MUACとmrdr) は、最初の質問で子供の年齢が5才以下出会った場合にのみ表示されます。

## 繰り返し

begin repeatとend repeatに挟まれた（複数の）質問を繰り返すことができます。

この例では、name, birthweight, sexの3つの質問が繰り返されます。回答者は「グループを追加」を選択することにより、必要な回数この3つの質問を表示・回答することができます。

<b>survey</b>			
	type	name	label
	begin repeat	child_repeat	
	text	name	子供の名前
	decimal	birthweight	子供の出生時体重
	select_one male_female	sex	子供の性別
	end repeat		
<b>choices</b>	list name	name	label
	male_female	male	男
	male_female	female	女

begin repeatの行のlabel列は空欄でも構いませんが、ラベルをつけておけば入力フォームの対応部分に表示されます。

他の例をご覧になりたい方は以下のURLをご覧下さい。

[https://docs.google.com/spreadsheets/d/1\\_gCJml\\_FzJ4qiLU-yc67x1iu\\_GL-hfU3H8-HvINsIoE/edit](https://docs.google.com/spreadsheets/d/1_gCJml_FzJ4qiLU-yc67x1iu_GL-hfU3H8-HvINsIoE/edit)

繰り返しの回数を指定するには、repeat\_count列を追加し、回数を記入します。（指定しない場合は、回答者が好きなだけ何度でも繰り返し回答することができます。）

<b>survey</b>				
	type	name	label	repeat_count
	begin repeat	child_repeat		3
	text	name	子供の名前	
	decimal	birthweight	子供の出生時体重	
	select_oneのmale_female	sex	子供の性別	
	end repeat			
<b>choices</b>	list name	name	label	
	male_female	male	男	
	make_female	female	女	

上の例では、begin repeatとend repeatに挟まれた部分を、ぴったり3回繰り返します。

一部のプラットフォームでは、動的に繰り返し回数を指定することができます。下の例では、アンケート回答中に入力されるnum\_hh\_membersの値によって、hh\_memberのグループのくり返し回数が決まります。

survey				
	type	name	label	repeat_count
	integer	num_hh_members	世帯員の数？	
	begin repeat	hh_member		\${num_hh_members}
	text	name	名前	
	integer	age	年齢	
	select_one male_female	gender	性別	
	end repeat			
choices	list name	name	label	
	male_female	male	男	
	male_female	female	女	

## 多言語への対応

複数の言語に対応した入力フォームを簡単に作成することができます。

label::language1, label::language2のように複数の言語でラベルを準備しておくだけでOKです。以下の例では、英語と日本語を切り替えることができます。入力フォームを表示した状態で、メニューから「言語を変更」を選択し、表示言語を切り替えます。メニューやボタンの言語を切り替えるには、携帯端末自体の言語を切り替えてください。携帯電話の場合メニューキーを押して「言語を変更する」を選択する、スマートフォンの場合は「設定-->言語と入力」を選択する、などして切り替えます。

survey					
	type	name	label::英語 (English)	label::日本語 (Japanese)	constraint
	integer	age	How old are you?	あなたは何歳ですか？	. <= 150

## 参考情報

ヒントやメディアファイルも::languageで多言語に対応させることができます。

### メディア

画像を表示したりビデオや音声を再生した後に回答を入力してもらうような質問を作成することができます。

画像、音声、ビデオなどのメディアファイルを、ODK Collectアプリを利用する携帯端末の /odk/forms/formname-media フォルダにコピーします。そして、surveyワークシートの media列に、正確なファイル名を記入します。以下に例を示します。

survey					
	type	name	label	media::image	media::video
	note	media_example	メディアの例	example.jpg	example.mp4

メディアファイルをスマートフォンに手作業でコピーすることができますが、やや手間がかかりますので、サーバ（ODK Aggregate）にXFormファイルをアップロードするときに、一緒にメディアファイルをアップロードする方法が便利です<sup>12</sup>。そうしてアップロードしておくと、フォームを端末にダウンロードするときに、自動的にメディアファイルも一緒にダウンロードされ、上記フォルダーに格納されます。

以下のURLに鳥の写真などを使ったXLSFormの例がありますので、参考にしてください。

<https://docs.google.com/spreadsheets/d/1Rxft3H3xl3M9bLFGR2XhXzt1ucyFmndoqFmOQ6FaqJw4/edit#gid=246559171>

<sup>12</sup> ファイルが多数あるときはまとめてZIP圧縮したファイルをアップロードして下さい（サブフォルダは作らないでください）。

## 位置情報

質問の種類としてgeopointを指定すると、位置情報を取得する質問を作成できます。回答者は、GPS等を用いて自動的に取得した位置情報を記録します。

GPSの位置情報が誤っている場合や、現在地とは違う場所を入力したい場合には、地図を表示して手作業で位置情報を修正したり指定することができます。質問の種類としてgeopointを指定し、さらにappearance列に placement-mapを指定してください。

また、質問の種類としてdecimalを指定し、appearance列に bearingを指定すると、センサを用いて自動的に方向を入力する質問を作成することができます。

質問の種類として、imageを指定し、appearance列に annotateを指定すると、例えば、風景の写真を撮ってその上にお絵描きして印をつけることもできます。

survey				
	type	name	label	appearance
	geopoint	location	位置を入力	placement-map
	decimal	orientation	方角を入力	bearing
	image	img-data	写真を撮影し、該当箇所をまるで囲んでください	annotate

## CSVデータの事前読み込み

入力フォームを提示する際に既存のデータを併せて示したい場合は、事前読み込み機能を利用します。例えば、ある住宅の各種情報を既に昨年収集しており、今年は住宅が適切に管理されているかどうかの5段階評価データだけを更新すれば良いような場合には、昨年収集した各種情報を参照することができると便利です。

データの事前読み込みを行うには、.csvファイルを、メディアファイルと同じようにして、端末にコピーするなりODK Aggregateにアップロードするなりします。.csvファイル

は1つでも複数でも構いません。ZIP圧縮してからアップロードすると、高速にアップロードできます。

.csvファイルの最初の行は列の名前です。それ以降の行は、実際のデータです。.csvファイルには、すべての行を区別することのできる列（「キー列」と呼ぶ）が少なくとも1つ存在する必要があります。フォームを回答者に提示するときに該当する行のデータを読み込みますが、そのときにキー列を利用します。キー列の名前の末尾には`_key`という文字を追加してください。列の名前の末尾が`_key`となっていると、索引が自動的に作成され、高速に事前読み込みを行うようになります。

例を示します。このデータを格納したCSVファイルの名前を `fruits.csv` とします。

<code>name_key</code>	<code>name</code>
mango	マンゴー
orange	オレンジ

## CSVデータを読み込む方法

計算(calculate)を使って、CSVファイルから入力フォームにデータを読み込みます。以下に例を示します。

`calculate`の行を追加し、`fruit`という名前を付け、`calculation`列のところに `pulldata()` という関数を使って読み込むデータを指定しています。

<code>survey</code>				
	<code>type</code>	<code>name</code>	<code>label</code>	<code>calculation</code>
	<code>calculate</code>	<code>fruit</code>		<code>pulldata ('fruits', 'name', 'name_key', 'mango')</code>
	<code>note</code>	<code>note_fruit</code>	<code>\$({fruit}) を csv ファイルから読み込んだ</code>	

`pulldata()`関数を使って.csvデータを読み込んだら、それ以降の関連制御(relevant列)、制約条件(constraint列)、ラベル(label列)において、そのデータを参照することができます

す。（回答者が既にフォームに入力したデータを、それ以降の入力項目において参照するのと同じような感じです。）

## CSVデータの事前読み込みに関する重要事項

- ・ サイズの大きな.csvファイルは、zip形式で圧縮してください。
- ・ .csvファイルに英語以外の言語（例えば日本語）や特殊文字が含まれる場合は、UTF-8形式で保存してください。そうしておけば、Androidデバイス上で正しくテキストを表示することができます。
- ・ .csvから読み込んだデータはテキスト形式で扱われます。整数や数値として扱いたい場合は、int()やnumber()関数を使ってデータを変換してください。
- ・ 秘匿すべきデータが含まれているなどの理由で、.csvファイルをサーバにアップロードしたくない場合は、代わりに空の.csvファイルをアップロードしてください。端末にフォームをダウンロードした後に、手作業で本物の.csvファイルと差しかえればOKです。

## 事前に読み込んだデータの動的選択

.csvファイルを事前読込した場合、選択肢を用いる質問（select\_oneやselect\_multiple）の選択肢を.csvファイルから動的に生成することができます。

通常の選択肢質問と同様に、type列を、select\_one listnameあるいはselect\_multiple listnameとします。なお、listnameは選択肢リストの名前です。次に、appearance列を指定します。最後に、choicesワークシートに、listnameに対応する行を1つあるいは複数追加します。

survey				
	type	name	label	appearance
	select_one fruits_list	fruits_choice	フルーツを選択	search('fruits')

appearance列には、search()式を記入し、.csvデータのどの行を選択肢リストに含めるか指定します。上の例のようにsearch(csvName)とした場合、.csvファイルのすべての行からなる選択肢リストが生成されます。ただし、同じデータが2つある場合は、一つにまとめられます。特定のキーワードを含むデータだけに絞ることなどもできます<sup>13</sup>。

既定と異なる外観スタイル(appearance style)を指定する場合は、先にそちらを指定してから、空白を1つ入れて、その次に search()式を記入します。外観スタイルについては後述しますが、例えば quick search() のようになります。

選択肢のラベルと値を指定するために、choicesワークシートに行を追加します。

- list\_name列は選択肢リストの名前です。
- name列は.csvファイルのキー列の名前を指定します。
- label列は、csvファイルの列の名前を指定します<sup>14</sup>。

choices			
	list name	name	label
	fruits_list	name_key	name

このような行を作成しておけば、choicesワークシートのname列、label列のデータは自動的に挿入されます。

以下のURLに例題がありますので、参考にして下さい。

[https://docs.google.com/spreadsheets/d/  
1YovWocjl1nbkZczXRmcTC71Pso8dRbouPSYWGBdvBWU/edit](https://docs.google.com/spreadsheets/d/1YovWocjl1nbkZczXRmcTC71Pso8dRbouPSYWGBdvBWU/edit)  
[https://docs.google.com/spreadsheets/d/  
1gprb7ocTYlT\\_seOBFY5CuoxyodcXwWOuVxmp38OX1dE/edit#gid=1405027019](https://docs.google.com/spreadsheets/d/1gprb7ocTYlT_seOBFY5CuoxyodcXwWOuVxmp38OX1dE/edit#gid=1405027019)  
(CSVファイル)

<sup>13</sup> 詳しくは、<https://opendatakit.org/help/form-design/data-preloading/>を参照

<sup>14</sup> 指定したcsvファイルの列名に従って、choicesワークシートのname列、label列は自動的に挿入されます。列は、コンマで区切って複数の列名を指定することができます。

選択肢の表示順は、.csvファイル中の順番と一致します。明示的に順番を指定する場合は、.csvファイルに sortby という列を追加し、数値で順番を指定すれば、選択肢がその順番で表示されます。

.csvから読み込む動的な選択肢と、通常の静的な選択肢を混在させることができます。静的な選択肢は通常の方法で定義します。.csvのための行の上に追加しても下に追加しても構いません。ただし、静的な選択肢の値は数値にしなければなりません。

## カスケード選択

最初に国名を選択肢、次に県名、次に市町村名を指定するような場合のことを、カスケード選択と呼びます。カスケード選択の場合、choice\_filter を使うと便利です。以下のURLに例題があるので参考にして下さい。

<https://sites.google.com/site/akiyatoolbox/opundetakitto-open-data-kit-toha/xlsformno-xiangshii-shuo-ming>

### survey

type	name	label	choice_filter
select_one states	state	state	
select_one counties	county	county	state=\${state}
select_one cities	city	city	state=\${state} and county=\${county}

### choices

list_name	name	label	state	county
states	texas	Texas		
states	washington	Washington		
counties	king	King	washington	
counties	pierce	Pierce	washington	
counties	king	King	texas	
counties	cameron	Cameron	texas	
cities	dumont	Dumont	texas	king
cities	finney	Finney	texas	king
cities	brownsville	brownsville	texas	cameron
cities	harlingen	harlingen	texas	cameron
cities	seattle	Seattle	washington	king
cities	redmond	Redmond	washington	king
cities	tacoma	Tacoma	washington	pierce
cities	puyallup	Puyallup	washington	pierce

## 外部選択

長大な選択肢（数百～数千）がある場合、フォームの動作が遅くなる可能性があります。外部選択を利用すれば、動作速度の低下をある程度回避できます。

外部選択を利用するには、select\_oneの代わりにselect\_one\_externalとし、choicesワークシートの代わりに、external\_choicesワークシートを作成します。以下のURLに例題があるので参考にして下さい。

<https://docs.google.com/spreadsheets/d/12qZL34kuHSZGWDvoBBJ1qf7dSmml-d2VnMWHoVtg-O4/edit>

外部選択を利用すると、external\_choicesとして定義された選択肢は、itemsets.csvというcsvファイルに変換されます。このcsvファイルを、メディアファイルと同じようにODK Aggregateにアップロードします。

## デフォルト

回答者が質問を始めて見た時に、典型的な回答などを予め表示しておくと、回答が楽になることがあります。こうした回答を予め準備しておくことを「デフォルトを設定する」といいます。デフォルトを設定するには、default列を追加します。次の例では、体重のデフォルトを51.3 kgに設定しています。回答者はタップ操作により簡単に値を変更することができます。

## 読み取り専用

読み取り専用にするには、read\_only列を追加し、yesを指定します。デフォルトを設定し、読み取り専用とすることで、回答者に情報を提供することができます。

調査					
	type	name	label	read_only	default
	integer	num	患者の人数：	yes	5

## 外観 (Appearance)

質問をどのように表示すべきか指定することができます。appearance列を追加して、そこに適切な値を設定しましょう。指定することができる値を以下の表に示します。質問の種類と対応していますので、注意してください。

appearance 列の値	対応する質問の 種類	説明
multiline	text	Webクライアント向き。テキストボックスを複数行にわたる大きさにする。
minimal	select_one, select_multiple	選択肢がポップアップメニューで表示される。
quick	select_one	モバイルクライアントのみに関連。回答が選択された後、自動的に次の質問の画面に切り替わる。
no-calendar	date	モバイルクライアントのみ。カレンダーを表示しない。
month-year	date	日付の選択において、年月のみを選択するようにする。
year	date	日付の選択において、年のみを選択するようにする。
horizontal-compact	select_one, select_multiple	Webクライアントのみ。選択肢を水平方向に表示する。
horizontal	select_one, select_multiple	Webクライアントのみ。選択肢を水平方向に、列を用いて表示する。
likert	select_one	Weクライアント向き。回答をLikertスケールのようにして表示する。
compact	select_one, select_multiple [choices with images]	画像による選択肢を横に並べる
quickcompact	select_one [choices with images]	上と同じだが、次の質問の画面に自動的に切り替わる（モバイルクライアントのみ）
field-list	groups	モバイルクライアントのみ。1画面に質問のグループがすべて表示される。
label	select_one, select_multiple	選択肢のラベルだけを表示する（入力部分を省略）
list-nolabel	select_one, select_multiple	上のlabelと一緒に使う。ラベルを省略し入力部分のみ表示する。モバイルクライアントを使用する場合、labelとlist-nolabelは、field-listを指定したグループの中で使用する。
table-list	groups	上のような表示を行う勘弁な方法。少し動作が遅くなる可能性がある。
signature	image	モバイルクライアントのみ。手書きでサインできるようにする。
draw	image	モバイルデバイスの画面上で、指を使って描画できるようにする。

これらを全て実際に試してみることができるXLSFormが以下のURLにありますので、参考にして下さい。

<https://docs.google.com/spreadsheets/d/159tf1wNeKGRccgizZBlU3arrOM--OpxWo26UvZcDEMU/edit>

## settings (設定) ワークシート

ここまで説明してきた設定内容は全てsurveyワークシートかchoicesワークシートで行いますが、XLSFormにはもう一つ別のワークシート（settingsワークシート）があります。settingsワークシートは無くても構わないのですが、このワークシートを使うとフォームの暗号化や全体的なスタイルの設定などを行うことができます。

まずsettingsワークシートの例を示します：

settings					
	form_title	form_id	public_key	submission_url	default_language
	例題のタイトル	example_id	IIBIjANBg ...	<a href="https://example-odk-aggregate.appspot.com/submit">https://example-odk-aggregate.appspot.com/submit</a>	日本語

- **form\_title** : 回答者に対して表示するフォームのタイトルです。form\_titleを指定しない場合や、form\_titleに空白が指定されている場合は、form\_idに基づいてフォームのタイトルを表示します。
- **form\_id** : フォームを識別するための名前。form\_idを指定しない場合や、form\_idに空白が指定されている場合は、.xls (.xlsx)のファイル名に基づいてフォームを識別します。
- **public\_key** : フォームを暗号化する場合の公開鍵です。
- **submission\_url** : フォームを暗号化する場合に、完成したフォームの送信先を指定するURLです。
- **default\_language** : 多国語に対応したフォームでデフォルトの言語を指定します。

## フォームの暗号化

フォームを暗号化すれば、ODK Aggregateサーバに送られるデータは暗号化されますので、秘密鍵を所有する人だけしか内容を見ることができません。ODK Aggregateサーバとの間の通信路が暗号化されていない場合（HTTPSではなくHTTPによる通信の場合など）でも、フォームを暗号化していればデータの内容を秘密にすることができます。

XLSFormを暗号化するには、settingsワークシートにid\_string, submission\_url, public\_keyの列を追加します。

- **id\_string** - フォームを識別するための名前です
- **submission\_url** - フォームの送信先を指定するURLです
- **public\_key** - base64RsaPublicKeyです

フォームの暗号化については、第6章で説明していますので、そちらをご覧ください。また、以下のURLに、暗号化を行うXLSFormの例がありますので、参考にして下さい。

[https://docs.google.com/spreadsheets/d/1O2VW5dNxXeyr-V\\_GB3spS6QPX4rtqtt7ijqP\\_uZLU3I/edit#gid=390337726](https://docs.google.com/spreadsheets/d/1O2VW5dNxXeyr-V_GB3spS6QPX4rtqtt7ijqP_uZLU3I/edit#gid=390337726)

## フォーム送信名の指定

form\_idでフォームに名前をつけましたが、instance\_nameを使うとフォーム送信毎に別の名前をつけることができます。毎回違う名前を生成する必要がありますので、回答者がフォームに入力する情報を用いて一意に識別可能な名前を自動生成します。以下のURLに示されている例では、回答者が入力する名前とuuidを連結して唯一の名前を生成しています。

<https://sites.google.com/site/akiyatoolbox/opundetakitto-open-data-kit-toha/xlsformno-xiangshii-shuo-ming>

## 複数Webページへの分割

style列にpagesを指定することで、Webフォームを複数のページに分割することができます。

settings			
	form_title	form_id	style
	タイトルの例	example_id	pages

surveyタブ上で、ページごとに質問をグループ化して下さい。そして、そのグループの外観（appearance）をfield-listとして下さい。以下の例を参照してください。

以下のURLに、関連するブログの記事とXLSFormの例がありますので、参考にして下さい。

<https://blog.enketo.org/pages/>

<https://docs.google.com/spreadsheets/d/1yZqG2XtoI4duVxPqx-Snyot860iKtjHuBKXTRzCht6E/edit#gid=1495622120>

survey				
	type	name	label	appearance
	begin group	group1		field-list
	text	name	回答者の名前	
	integer	age	回答者の年齢	
	string	address	回答者のアドレス	
	end group			

## グリッド表示のフォーム

style列でtheme-gridを指定すると、複数の質問を1行に詰め込んで表示します。つまり、紙を用いた伝統的な調査用紙と似たような感じでフォームを表示することができます。パソコンや大型のタブレットにフォームを表示したり、紙にフォームを印刷する場合に便利です。

settings			
	form_title	form_id	style
	タイトルの例	example_id	theme-grid

surveyワークシート中で、各セクションと一緒に表示した質問をグループにまとめてください。appearance列に各質問の横幅を指定します。デフォルトの横幅は4です。

survey				
	type	name	label	appearance
	begin group	group1		
	text	name	回答者の名前	w3
	integer	age	回答者の年齢	w1
	string	address	回答者の住所	w4
	end group			

以下のURLに、関連するブログの記事と XLSFormの例がありますので、参考にして下さい。

<https://blog.enketo.org/gorgeous-grid/>

<https://docs.google.com/spreadsheets/d/1Z4gHZQTr5FibRK-Aj198WlNdMZghEBZlyWhmPZXjzJQ/edit#gid=74114172>

## スタイリング

以下の記法を用いて、太字、斜体、見出し、各種フォントスタイル・色、Webリンクなど（ODK Collect 1.4.9とEnketo）を使ったフォームを手軽に作成することができます。

- \_か\*で語を囲むと **強調** できます。
- \_か\*\*で語を囲むと **もっと強調** できます。
- [name of link](url)でWebリンクを追加。
- #を語の前につけると見出しになる。#(最大)から #####(最小)まで、#の数を増やすと文字が小さくなる。
- HTMLのspanタグにより色やフォントを指定する(例えば, <span style="color:#f58a1f">オレンジ<span>, <span style="color:red; font-family:cursive">赤い筆記体 (red and cursive)<span>)
- Ctrl + Enter、あるいはCtrl + Alt + Enterで（表計算ソフトにより異なる）改行を入れる。
- お気に入りの絵文字を追加できます😊📝😊！

## XLSFormに対応したプラットフォーム／ツール

- Open Data Kit (ODK)
- Formhub
- Ona

- SurveyCTO
- Enketo
- Kobo ToolBox
- CommCare
- Nafundi's XLSForm Offline
- Nathan Breit's xlsform.exe
- DataWinners

## その他のリソース

XLSFormをXMLに変換したものをXFormと呼びますが、XFormを直接読んだり編集したりしたい人は、以下のページを参考にして下さい。

- フォームのガイドライン: <http://code.google.com/p/opendatakit/wiki/XFormDesignGuidelines>
- フォームデザインのチュートリアル: <https://bitbucket.org/javarosa/javarosa/wiki/buildxforms>
- サンプルフォーム: <http://code.google.com/p/opendatakit/source/browse/?repo=forms>
- Formhub University: [http://formhub.org/formhub\\_u](http://formhub.org/formhub_u)
- Ona Form Gallery: <https://ona.io/home/>
- JavaRosaがサポートするXForms: <https://bitbucket.org/javarosa/javarosa/wiki/xform-jr-compat>

## XLSFormの歴史

XLSFormはもともとコロンビア大学の持続可能なエンジニアリング研究所(SEL)のAndrew MarderとAlex Doreyが開発したものです。ODKコミュニティがXLSFormを採用し、SELとワシントン大学のODKチームが共同で現在の仕様を開発しました。XLSFormsをXFormに変換するライブラリPyXFormは、ODK、SEL、Ona、SurveyCtO、Koboのメンバーによってサポートされているオープンソースプロジェクトです。

## 第5章

# セキュリティとプライバシー<sup>15</sup>

空き家の調査で得られる情報には、一般に公開すべきでないメモ、写真、個人情報などが含まれる可能性があります。こうした情報を扱う場合は、空き家の調査のために利用する情報収集ツールのセキュリティとプライバシーについて理解しておくことが重要です。本章では、オープンデータキットのセキュリティとプライバシーについて説明します。

## ライセンス

ODKソフトウェアはApache 2 ライセンスで公開されています。インストーラ、プログラム、ソースコードはありのまま提供していますので、誰かがその内容を保証したり責務を負うものではありません。詳しくはライセンスの説明をご覧ください。

## 通信手段

ODKソフトウェアが開発者に情報を送信することはありません。また、利用者のデバイス（スマートフォン等）に開発者が侵入したり遠隔操作するための仕組みは組み込まれていません。

しかし、悪意のあるハッカー等が、ODKソフトウェアやサードパーティのライブラリの脆弱性を発見・悪用し利用者のデバイスに侵入したり遠隔操作する可能性はあります。

ODKのウェブサイト([opendatakit.org](https://opendatakit.org))は、故意にクッキーを使用したり、サードパーティの広告を掲載したり、一般の人々から個人情報を収集することはありません。

---

<sup>15</sup> 本章は次のURLの内容に基づきます：<https://opendatakit.org/about/security-and-privacy-statement/>

## サードパーティのソフトウェア

ODKソフトウェアはサードパーティのライブラリを多用しています（著名なライブラリもそうでないものも利用しています）。それらのライブラリのセキュリティについて特に審査等は行っていません。

セキュリティ担当者の方などがライブラリやソースコードを見たい場合、以下のサイトにアクセスしてください。

<https://github.com/opendatakit/opendatakit>

## ODK Aggregateの通信

自前のWebサーバでODK Aggregateを動かす場合、サーバとODK AggregateでSSL証明書の設定を行わなければ、サーバとやりとりするデータは、見ようとすれば全て丸見えになります。

見ようとしても見えないようにするために、<https://>によりデータをやりとりする必要があります。

暗号化されたフォームの定義自体は、平文で（暗号化しないで）デバイスに送られます。暗号化されたフォームにデータを入力して確定させると、デバイス上で暗号化が行われ、暗号化された状態でサーバに送信されます。暗号化されたデータを送信するため、<http://>のような安全でない通信を用いたとしても、データ見えないようにすることができます。しかし、安全でない通信を用いると、第三者がフォームの定義を改ざんして暗号を無効化し、フォームに記入されたデータを取得したり、サーバとの通信を妨害する可能性があります。

## ODK AggregateをGoogle App Engineや他のホスティングサービス上で運用する場合

ホスティングサービスを利用してODK Aggregateを運用する場合、ホスティングサービスの従業員等はあなたのデータを見ることができると思っておく必要があります。データセンターにおいて、データと通信の安全性を確保する対策（アクセス制限、暗号化など）をどれくらいしっかりとやってくれるかについては、ホスティングサービスごとに異なります。

暗号化されたフォームであっても、フォーム定義や関連ファイルは、サーバ上に平文で（暗号化されないで）保存されます。暗号化されたフォームにデータを入力し、フォームを確定すると、データはデバイス上で暗号化され、暗号化された状態でサーバに送信されます。そして、サーバは受け取ったデータを保存します。暗号の解読に必要な秘密鍵は、サーバには保存されていません。従って、ホスティングサービスの従業員等であっても、暗号を破らない限りフォームに入力された内容や添付データを見ることができません。

## 暗号化されたフォームのセキュリティ

暗号化されたフォームの定義自体は、平文で（暗号化しないで）サーバに保存されており、平文のままデバイスに送られ、平文のままデバイスに保存されます。

フォームを確定させる前は、すべてのフォームデータや添付データは、デバイス上に平文で（暗号化なしで）保存されています。

フォームが確定されると、そのフォームに対して、SecureRandom数値ジェネレータを用いて256ビットの暗号化/解読キーが生成されます。こうして、すべての確定されたフォームに、固有の256ビット暗号化/解読キーが割り当てられます。

フォームに入力されたデータと、全ての添付データ（メディアファイル）は、256bitのAES CFB（Cipher Feedback）ストリーミングブロック暗号を用いて、暗号化されます。暗号化が完了すれば、そのフォームの平文データ（フォームに入力されたデータと添付データ）は全て削除されます。

256ビットの暗号化/解読キーにはパディングが施され、フォーム定義中に指定されているRSA公開鍵（2048 bitを推奨）と、OAEPWithSHA256AndMGF1Paddingアルゴリズムを用いて、暗号化されます。こうして暗号化されたキーは、暗号化された入力データおよび添付データとともにサーバに送信されます。送信時にシグネチャフィールドを付与することにより、入力データや添付データが不正に書き換えられた場合、ソフトウェアによって検知することができるようになります。

デバイス上の、削除された（平文の）入力データや添付データは、新しいデータで上書きされるまでの間、SDカードのフリーリストに残っている可能性があります。

サーバ上で、仮に何者かが暗号化されたあなたのデータにアクセスできたとしても、それぞれのフォームの入力データに対して異なるキーを使用していますので、それぞれのフォームのデータは一つずつ解読する必要があります。

現時点では、最も先進的な政府機関（例えばNSA）を除いては、AES暗号は不正に解読することは不可能であると考えられています。

## ODK Aggregateサーバにおけるユーザ名による認証

ODK Aggregateサーバでは、ユーザ名とパスワードによる認証を行うことができますが、認証を行う際にODKツールは DigestAuth を利用します。従って、<http://> で通信を行っていたとしても安全にユーザ名／パスワード認証を行うことができます（DigestAuth ではパスワードを単純にネットワークに送信することはありません）。

サーバ上には符号化されたユーザ名とパスワードが保存されています。これらの符号化されたデータが盗まれたり漏洩すると、そのユーザとしてサーバに侵入される可能性があります。

## Google Gmailによる認証

Googleアカウント（GmailやGoogle Apps）の認証に関しては、ODK Aggregateはユーザの電子メールアドレス（単にアドレスだけで、email/ユーザプロファイル情報は含みません）を閲覧する権限を持つOAuth2トークンを、本人であることを証明するために受理します。

この方法による本人認証の仕組みは、非常に弱いものであると言えます。あなたの電子メールアドレスを他のサイトやアプリケーションと共有することをGoogleに許可する度に、それらのサイトやアプリケーションは、あなたとしてODK Aggregateを操作する権限を持つことになります。

ですから、Googleの電子メールアドレスによるサイトへのアクセスを宣言し許可することは不適切かもしれません。なお、この方法によるアクセスは、ODK 2.0 Sync (rev 128かそれ以前のバージョン) の機能を利用する場合、必要になります。

## 識別情報の送信と蓄積

データ送信するときに、識別情報の一部がサーバに送信され、またサーバに保存されます。

1. ODK Collectは、データ送信するときに端末のdeviceIDをサーバに渡します。（送信開始時のHEADリクエストに .../submission?deviceID=imei%3A9117DD011813771 のようなURL形式を用います。）ODK Aggregateサーバは、このdeviceIDをデータベースのテーブルに保存することはありません。しかし、この情報は通常Webサーバのアクセスログに記録されます。deviceIDはデータを送ってきたデバイスを一意に識別することができる情報です。この情報を用いれば、サーバ上でのイベントを特定のデバイスと関連付けることができます。この情報がロギングされるため、フォームのデータがどのデバイスから（つまり誰から）送られてきたかが特定されてしまう可能性があります。
2. ODK Aggregateで（ユーザ名／パスワードあるいはGoogleアカウントによる）認証の設定を行った場合、つまりData Collectorの権限をanonymousUserに与えない場合ということになりますが、送信されたデータを保持するテーブルの監査情報フィールドに、認証済みのユーザ名（あるいはGoogleアカウント）の情報が書き込まれます。anonymousUserにData Collectorの権限を与える場合は、認証は行われませんので、'anonymousUser'がそれらのフィールドに書き込まれます。監査情報フィールドに保存された情報は、エキスポートしたCSVファイルや、プルしたODK Briefcaseのデータには含まれませんし、下流のシステムに公開することもありません。しかし、データベースのテーブルに格納はされていますので、当然ながら、認証済みのユーザ名やGoogleアカウントの情報と、送られてくるデータを関連付けることができます。

ODK Aggregateをウェブ上で利用する場合、サーバの設定、フォームの定義、フィルタ、エキスポート、パブリッシャー、データテーブルなどを変更する、認証を必要とする操作が行われると、更新されているデータベーステーブルの監査情報フィールドに、認証ユーザ名またはGoogleアカウントの情報が書き込まれます。変更操作によりデータベーステーブルに対する削除が行われる場合は、認証ユーザ名もしくはGoogleアカウントの情報が、削除された情報の概要とともに、サーバのログの一部として記録されます。

## オンラインのXLSForm変換ツール

オンラインのXLSForm変換ツール (<http://opendatakit.org/xiframe/>) は、安全性の低い接続方式を使っています。つまり、サイトとの間でアップロード・ダウンロード通信が発生する際に、第三者があなたの（XLS / XML形式による）フォーム定義ファイルを見ようと思えば見ることができるということです。フォームのエラー情報についても同様です。

さらに、オンラインのXLSFormツールは、あなたがアップロードしたXLSファイルと、生成されたXMLフォームを、削除されるまでのしばらくの間ディスクドライブに保存します（ツールの運用上必要なため。）

オフラインのXLSFormツールとODK Validateツールは、ネットワーク通信を全く行わずに動作しますので、オンラインツールは便利ではありますが、オフラインツールのほうが安全です。

## ウェブサイト

OpenDataKitのウェブサイト、ダウンロードサイト、オンラインXLSFormツールはクッキーを使っています（あるいは使う可能性があります）そして、すべての操作を記録しています（あるいは記録する可能性があります）。また、セキュリティソフトウェア、スパム防御、ウェブアナリティクスツール（例えばGoogleウェブアナリティクス）を使っており、これらはウェブサイトの訪問者やそのアクセスパターンを追跡している可能性があります。

## Googleプレイストア

OpenDataKit関係者がアクセスすることのできる管理ポータル上に、Googleプレイストアにおけるダウンロード操作の情報が、集約された利用統計として示されます。

あなたが、アプリがクラッシュした場合にレポートを送信することにしている場合は、クラッシュレポートは匿名で提供されます。調査データや、デバイス及びユーザに固有のデータは含まれないようになっています。

## 参考文献

Samson Hussien Gejibo, *Towards a Secure Framework for mHealth. A Case Study in Mobile Data Collection Systems.* PhD Thesis, ベルゲン大学, 2015年.

(<http://bora.uib.no/handle/1956/10652>)

## 第6章

# フォームの暗号化<sup>16</sup>

フォームの暗号化を行えば、<http://>による通信を行う場合（つまり、SSL証明書がない場合や<https://>が使えない場合など）でもデータを秘匿化することができます。また、従来、プライバシーに関する法律や規則によってGoogle Appエンジン等のクラウドサーバを使うことができない場合であっても、フォームの暗号化を行えばクラウドサーバを使うことができるようになる可能性があります。

フォームを暗号化すると、ODK Collectアプリでフォームを確定する時に、非対称の公開鍵暗号が適用されます。暗号化されたデータがODK Aggregateサーバにアップロードされ、サーバに集められたデータはODK Briefcaseソフトウェアで手元のパソコンにダウンロードすることができます。ODK Briefcaseは、暗号化されたフォームのデータを解読し、CSVファイルとして書き出します。CSVファイルに書き出されたデータは、自由に分析や可視化を行うことができます。なお、ODK Briefcaseソフトウェアは、データを解読するときに秘密鍵を読み込みますが、読み込んだ秘密鍵の情報を保存することは決してありません。

確定したフォームのデータ（及び添付されたメディアデータ）は、ODK Aggregateサーバに送信する前に暗号化されます。ODK Aggregateサーバは暗号化された状態のデータを保持しますし、ODK Briefcaseソフトウェアがダウンロードし保持するデータも暗号化されたままの状態です。

ODK Aggregateサーバは、暗号化されたデータをうまくGoogleスプレッドシートやフュージョンテーブルに書き出すことができません。暗号化によってフォームのデータ全体が難読化されているためです。ODK Aggregateサーバには秘密鍵がないので、サーバ上ではデータを解読できません。つまり、暗号化されたフォームを利用する場合、ODK Aggregateサーバはデータを集めるだけの機能を提供することになります。データを利用するには、ODK Briefcaseを用いて手元のパソコンにデータをダウンロードし、秘密鍵を用いて解読する必要があります。

---

<sup>16</sup> 本章は以下のURLの内容に基づきます：<https://opendatakit.org/help/encrypted-forms/>

ODK Collectアプリでデータを集めている際に、フォームがまだ確定されていない状態でデータが保存されている間は、データはまだ暗号化されていません。フォームが確定された瞬間に ODK Collectアプリは 256ビットの対称鍵を生成し、この鍵を用いてフォームのデータと全ての添付メディアデータを暗号化し、暗号化されたデータと暗号化に用いた対称鍵を非対称鍵で暗号化した値を記述した送信管理票(submission manifest)を作成します。この送信管理票がODK Aggregateにアップロードされる「フォーム」であり、暗号化されたフォームのデータと暗号化された添付メディアデータは、この送信管理票（「フォーム」）に添付されます。

## 暗号化の準備

フォームの暗号化を行うには、以下のバージョンかそれより新しいODKツールを使用してください:

- ODK Collect 1.2 Release Candidate 1 (RC1)
- ODK Aggregate 1.0.4 Production
- ODK Briefcase 1.0 Production

暗号化されたフォームの定義は、一意のOpenRosa instanceIDと、明示的な<submission/>要素を含まなければなりません。この点については以下でより詳しく説明します。

## セキュリティ上の注意点

ODK Collectは、フォームが確定されると、暗号化されていない元のデータをデバイスから削除しようとします。しかし、ODK Collectは、例えば写真を撮影するときなどにはサードパーティのアプリケーションを呼び出して利用しますし、またデータを削除している最中に強制終了イベントが発生する可能性もありますので、全ての元データを常に確実に削除できると保証することはできません。更に、SDカードが採用しているデータ書き込み・削除の方式に関係することですが、削除されたデータがSDカードの解放領域から復元できる可能性があります。あなたが所属する組織の取り組みとして、ODK CollectデバイスのSDカードから全てのデータを確実に削除するための方法を調査し、定期的にデ

バイスの初期化や再インストールを行うなど、セキュリティ確保のための手順を確立することが必要です。

フォームの暗号化は、確定したフォームのデータを不正に閲覧したり書き換えたりすることを防ぐことができます。しかし、悪意のある者が、確定したフォームのデータを偽のデータとそっくり差し替えたり、余分なデータを合成・付加することを防ぐことはできません。これらはフォームの暗号化によるセキュリティ対策の守備範囲外です。

## 設定方法

ODK CollectとODK Aggregateで特別な設定を行う必要はありません。

ODK Briefcaseについては、まずJava Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files をJavaのダウンロードサイトから取得し、インストールする必要があります。暗号解読を行うためにはJCEが不可欠です。

JCEをインストールするには、まずダウンロードしたJCEのzipアーカイブを解凍します。解凍後に現れたフォルダ（ディレクトリ）の中にある local\_policy.jar と US\_export\_policy.jar という 2つのファイルを、あなたのコンピュータのJava Runtime Environment (JRE)がインストールされたフォルダの中の security フォルダにコピーし、これらの 2つのファイルの古いバージョンと置き換えます<sup>17</sup>。

---

<sup>17</sup> Windowsマシンでは、JREは通常 C:\Program Files\Java\jre6\lib\securityのような場所にインストールされます。Mac OS X上で ODK BriefcaseのGUIでフォームデータを解読しようとすると FAILすることができます。その場合、ODK Briefcaseをコマンドラインで実行するとうまくいくことがあります。

## フォームの定義

まず、(XMLにより定義した)暗号化フォームの例を示します<sup>18</sup>。

```
<h:html xmlns="http://www.w3.org/2002/xforms"
         xmlns:h="http://www.w3.org/1999/xhtml"
         xmlns:orx="http://openrosa.org/xforms/">
<h:head>
    <h:title>Sample Form</h:title>
    <model>
        <itext>
            <translation lang="English" default="">
                <text id="ask_name">
                    <value form="long">Please enter your name:</value>
                    <value form="short">Respondent's name</value>
                </text>
            </translation>
        </itext>
        <instance>
            <sample id="sample-v1.0">
                <orx:meta>
                    <orx:instanceID/>
                </orx:meta>
                <name/>
            </sample>
        </instance>
        <submission method="form-data-post"
                    action="https://my-opendatakit.appspot.com/submit"
                    base64RsaPublicKey="MIIBIjANB...JCwIDAQAB" />
        <bind nodeset="/sample/meta/instanceID" type="string"
              readonly="true()">
            <calculate>concat('uuid:', uuid())</calculate>
        <bind nodeset="/sample/name" type="string" />
    </model>
</h:head>
<h:body>
    <input ref="name">
        <label ref="jr:itext('ask_name')"/>
    </input>
</h:body>
</h:html>
```

---

<sup>18</sup> base64RsaPublicKeyの属性値は略記しています。

XLSFormを使って暗号化フォームを定義するには、settingsワークシートを利用します。具体的にはsettingsワークシートの中に submission\_url と public\_key を指定します。これらが指定されていれば、XLSFormは上に示したような暗号化フォームの定義を自動的に生成します。他に何もする必要はありません。以下の段落は読み飛ばして、次節（RSA鍵ペアの作成）へと読み進め、公開・秘密鍵ペアを作成・指定してください。

OpenRosa 1.0 メタデータスキーマを定義したOpenRosaメタデータブロックが存在していることに注意してください。これは必須です。instanceIDフィールドはどのような値を指定しても構いませんが、調査データ中、一意に識別することができる値を指定してください。<bind/>の部分は、フォームがinstanceIDフィールドを持たない場合にODK Aggregateが生成するinstanceID値と同等なinstanceID値を計算しています。もし、コロン記号とダッシュ記号以外の区切り記号やその他の特殊記号を使う場合には、十分にテストを行い、ODK AggregateへのアップロードやODK Briefcaseへのダウンロードの際に、正しくそれらの記号が処理されることを確認してください（正しく処理されるはずですが）。

フォームを暗号化するためには<submission/>タグも必要です。このタグの中の method 属性は必ず form-data-post とします。また、action 属性はデータの送り先の URL を指定しなければなりません（ODK Aggregate ウェブサイトの URL の Aggregate.html を submission と置き換えたものになります）。最後に、base64RsaPublicKey 属性を指定します。ODK Collect が（確定したフォームのデータを暗号化するために生成する<sup>19</sup>）対称鍵を暗号化するために使用する RSA 公開鍵を base64 で符号化した値を指定します。

## RSA鍵ペアの作成

RSA公開鍵・秘密鍵のペアは、OpenSSLソフトウェアパッケージを用いて作成することができます。このソフトウェアパッケージは Mac OS X や Linux には同梱されていますので、これらの OS を使用していればわざわざインストールする必要はありません。Windows の場合は、まずダウンロードとインストールを行う必要があります。

---

<sup>19</sup> 確定した個々のフォームに対して異なる対称鍵を生成します。

## OpenSSLのインストール (*windows*のみ)

自分のシステムに適したWindows用OpenSSLのインストーラーをダウンロードし、起動してください。DLLを Windowsシステムディレクトリにコピーするか、もしくは /bin ディレクトリにコピーするか聞かれたら、/bin と答えてください<sup>20</sup>。

## RSA鍵ペアの構築

Windowsの場合は、Windowsの'cmd'ウィンドウを開きます。下記の例のように、OpenSSLディレクトリ配下のbinディレクトリに移動します。

```
cd C:\OpenSSL-Win32\bin
```

Macの場合は、Terminal.appを開きます。下記の例のように Desktopディレクトリに移動します。

```
cd ~/Desktop
```

## 秘密鍵の作成

以下のコマンドを用いて 2048ビットの秘密鍵を作成し MyPrivateKey.pem という名前のファイルに書き込みます。構成ファイルがありません、というような警告メッセージが表示されるかもしれません、無視して結構です。

Windowsの場合：

```
openssl genpkey -out MyPrivateKey.pem -outform PEM -algorithm RSA  
-pkeyopt rsa_keygen_bits:2048
```

Macの場合：

```
openssl genrsa -out MyPrivateKey.pem 2048
```

---

<sup>20</sup>どちらでもインストールはうまくいきますが、/binにしたほうが Windowsシステムディレクトリが「散らからない」のでお薦めです。

## 公開鍵の作成（抽出）

次に、秘密鍵から公開鍵を抽出します。以下のコマンドを実行してください：

```
openssl rsa -in MyPrivateKey.pem -inform PEM -out MyPublicKey.pem  
-outform PEM -pubout
```

構成ファイルがありません、というような警告メッセージが表示されるかもしれません、ここでも無視して結構です。

## 鍵の保存と利用

MyPrivateKey.pemファイルを安全な場所に移してください。パスワードで暗号化されていませんので、このファイルにアクセスできる者は誰でもあなたのデータを解読することができます。ODK Briefcaseを用いて暗号化されたデータを解読するとき、あなたはこの秘密鍵ファイルを利用することになります。

## XLSFormのsettingsワークシートのpublic\_key列の更新

メモ帳（Windowsの場合）やテキストエディット（Macの場合）を用いて、MyPublicKey.pemファイルを開いてください。そしてまず、冒頭と末尾の'-----'となっている行を削除してください。次に、改行コード（キャリッジリターン）を全て削除してください。1行の長い文字列になったと思いますが、その文字列をXLSFormのsettingsワークシートのpublic\_key列にコピー＆ペーストしてください。この長い文字列が、暗号化フォーム定義におけるbase64RsaPublicKey属性の値になります。なお、公開鍵の情報だけを使うように細心の注意を払ってください。絶対に元の公開-秘密鍵ファイルの内容をペーストしないようにしてください。元の公開-秘密鍵ファイルの内容を使った場合、一見問題なく動作するように見えますが、セキュリティは保護されなくなります。

## 運用について

暗号化フォームを利用するには、暗号化フォーム（の定義）を<submission/>タグのaction属性で指定したODK Aggregateサーバに追加します。そして、ODK Collect 1.2 RC1かそれ以降のバージョンを利用して、暗号化フォームを各デバイスにダウンロードし、定期的にSDカードを消去するための手順や約束事をどのようにデザイン・実施するか検討します。

フォームを確定すると、フォームに記入したデータは暗号化されます。ODK Aggregateサーバは暗号化されたデータだけを保持し、秘密鍵にアクセスすることはできません。ODK Aggregateサーバにデータが集まつたら、ODK Briefcaseソフトウェアを使って暗号化されたフォームのデータをあなたのパソコンにダウンロードします。次に、秘密鍵を指定してデータを解読し、CSVファイルに平文のデータを書き出します。

ODK BriefcaseソフトウェアがCSVファイルを生成する時、データに列を一つ追加します。この列には、暗号化ファイルのシグネチャの良し悪しが記録されます。添付データが欠損している場合や、データが不正に書き換えられた場合は、シグネチャによって不正が検知されたとして、この列に記録されます。ただし、全部そっくり不正にデータが書き換えられてしまった場合は、シグネチャによる検知はできませんので、注意が必要です。