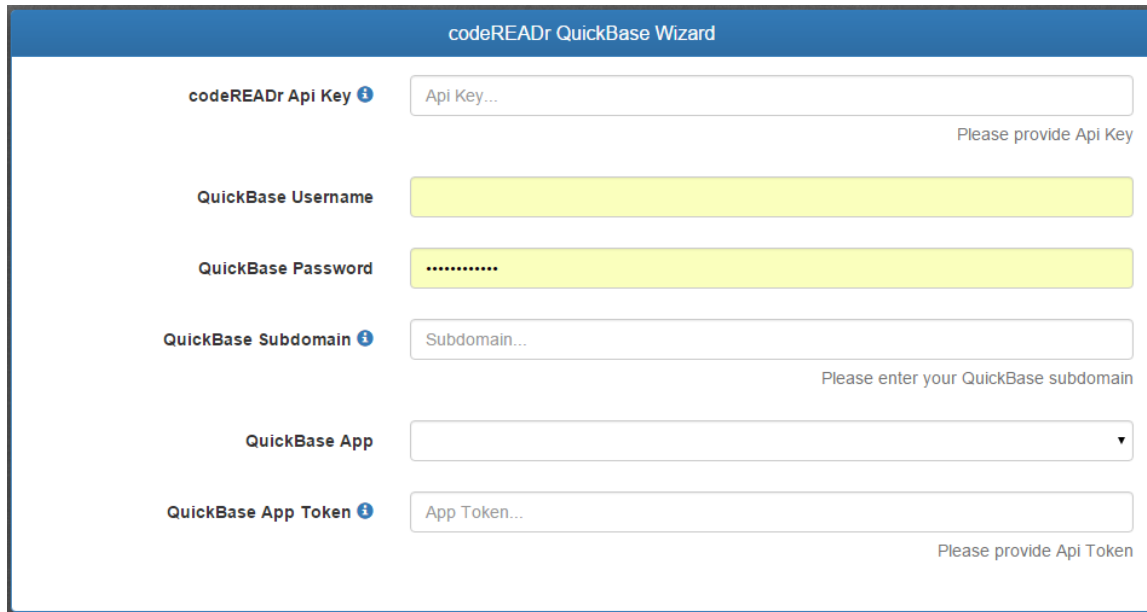


## Direct API Integration with QuickBase

With the 'wizard' you can auto-generate codeREADr services to enable app-users to view, add and edit records in your QuickBase tables directly from the codeREADr app. Click [here](#) to launch the wizard.

A screenshot of the "codeREADr QuickBase Wizard" form. The form has a blue header bar with the title "codeREADr QuickBase Wizard". Below the header, there are six input fields, each with a label and an information icon (a lowercase 'i' in a blue circle). The fields are: 1. "codeREADr Api Key" with a text input field containing "Api Key..." and a note "Please provide Api Key" below it. 2. "QuickBase Username" with a yellow highlighted text input field. 3. "QuickBase Password" with a yellow highlighted password input field showing ".....". 4. "QuickBase Subdomain" with a text input field containing "Subdomain..." and a note "Please enter your QuickBase subdomain" below it. 5. "QuickBase App" with a dropdown menu. 6. "QuickBase App Token" with a text input field containing "App Token..." and a note "Please provide Api Token" below it.

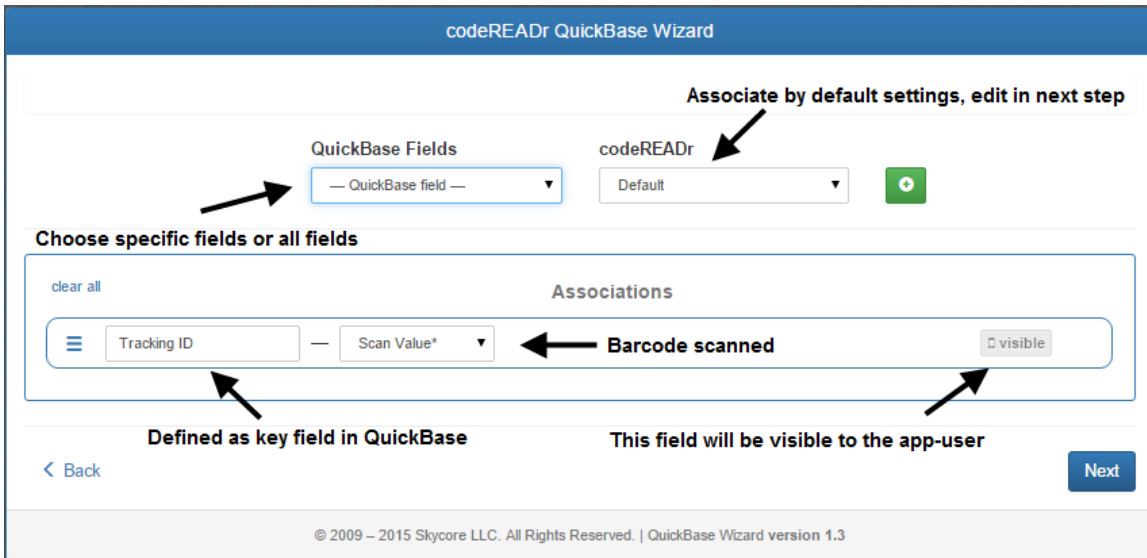
### The Wizard

The opening screen of the wizard is shown above. It connects the two platforms and allows you to select the target QuickBase app. Notice the 'i' icons – they will help you find the information you need. To use the wizard you will need the following:

- a) A codeREADr account with [paid plan](#) to get your API Key. For evaluation purposes you can sign up for the Free Plan and then request a temporary API key via [email](#).
- b) A QuickBase account with API access to copy your [application tokens](#) (included in their 30-day free trial).

### Auto-Generate Services

After entering this information, you can now associate specific fields within a table to allow your authorized app-users to view, add or edit them within the codeREADr app.

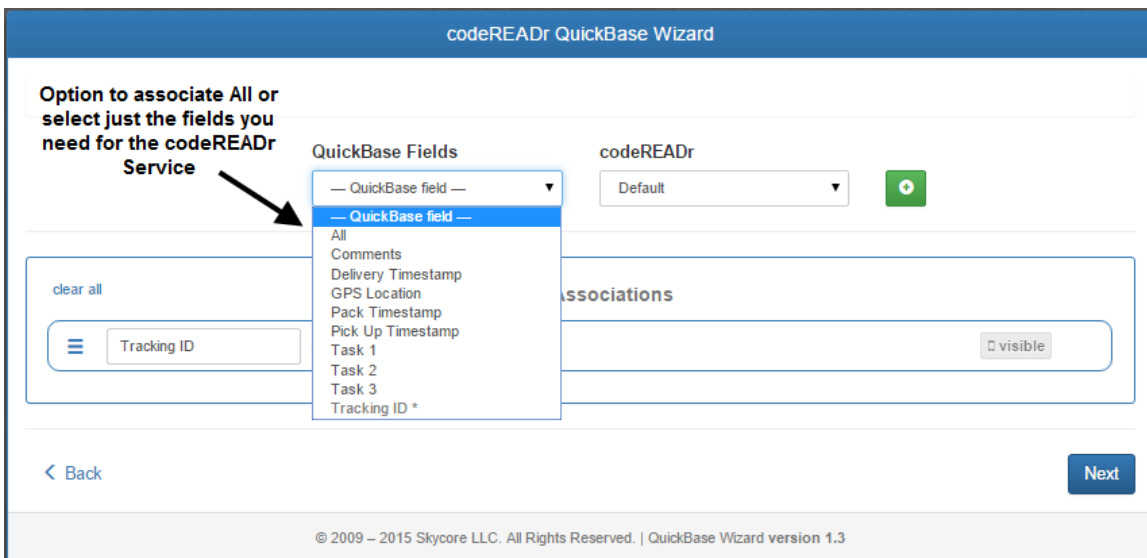


## Key Field

In the above example you'll see one association was automatically made because the QuickBase table had a key field defined. The key field is the value ('Scan Value') you expect the app-user to scan, manually enter or lookup ('Enable online search within the app') from the app.

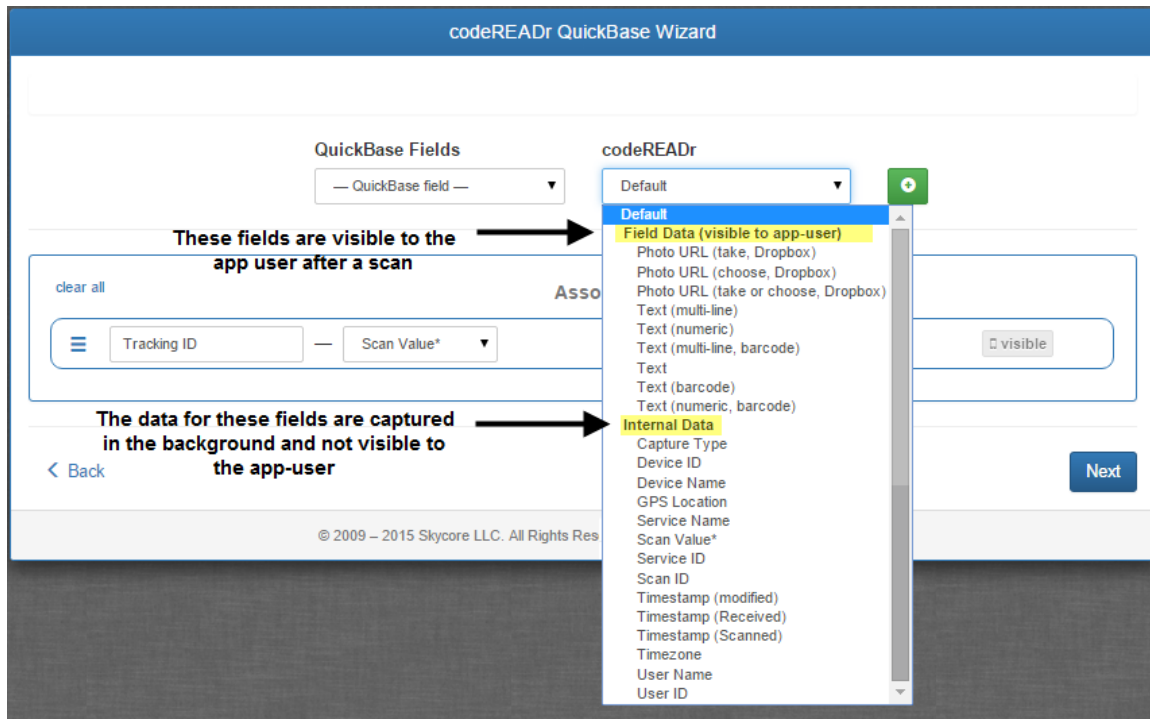
After a scan the app will first search your QuickBase table for that key field, then present that record's data to the app user for viewing and/or editing and then submit that to the table when the app user clicks 'submit'. If there is no match, then you can optionally allow the app-user to add that value.

You must have a key field defined and internet connectivity if you want the app user to be able to view and edit the data associated to that value. If no key field is defined then every scan will create a new record in the associated table.



Above is a view of the QuickBase fields in a sample table (in this case for package tracking). You can associate 'All' or any number of those fields.

Note: To simplify the association process you can create a special table within QuickBase that lists only those fields important for the data collection process.



Here's a view of the associated codeREADr fields. If you chose 'All' QuickBase fields and then choose 'Default' here, the wizard will try to automatically associate them. You will see a preview of each association in the next step for editing if required.

### Field Data Options

These fields will be visible to the app user. In the next step you can choose whether to allow the app user edit the field or 'read only'.

- Photo URL (choose, Dropbox)
- Photo URL (take or choose, Dropbox)
- Photo URL (take, Dropbox)
- Text (numeric, barcode)
- Text (multi-line barcode)
- Text
- Text (multi-line)
- Text (barcode)
- Text (numeric)
- Multiple Choice (select in the next step)

### Internal Data Options

- Capture Type (e.g. barcode scanned, value lookup, manual entry, NFC, mag stripe, etc.)
- Device ID (not commonly used)
- Device Name (not commonly used)

- GPS Location (option if tracking GPS location)
- Service ID (not commonly used)
- Service Name (commonly used)
- Scan ID (not commonly used)
- Scan Value \* (the barcode value or ID, **required whether key field or not**)
- Timestamp – Scanned (actual device time, commonly used)
- Timestamp – Received (server time, not commonly used)
- Timezone (not commonly used)
- Timestamp - Modified (not commonly used)
- User Name (the app user, commonly used)
- User ID (not commonly used)

clear all Associations **Allow app-user to edit the field after each scan**

☰	Tracking ID	—	Scan Value*	▼		visible	
☰	Comments	—	Text (multi-line)	▼	☑️ ✎️ editable	visible	🗑️
☰	Delivery Timestamp	—	Text	▼	☑️ ✎️ editable	visible	🗑️
☰	GPS Location	—	Text (multi-line)	▼	☑️ ✎️ editable	visible	🗑️
☰	Pack Timestamp	—	Text	▼	☑️ ✎️ editable	visible	🗑️
☰	Pick Up Timestamp	—	Text	▼	☑️ ✎️ editable	visible	🗑️
☰	Task 1	—	Text (multi-line)	▼	☑️ ✎️ editable	visible	🗑️
☰	Task 2	—	Text (multi-line)	▼	☑️ ✎️ editable	visible	🗑️
☰	Task 3	—	Text (multi-line)	▼	☑️ ✎️ editable	visible	🗑️

In the next steps you first review your choices and default associations. Then you should rename the codeREADr Template (optional) and Service. The Service is important to rename as that is what the app user will see when they sign into the app. You should also be sure to select only those app users you authorize to scan when they sign into the codeREADr app.

## Review Summary

Template Name	Status Tracking 30-Sep-2015 14:33:35	←
		<b>Rename</b>
Service Name	Status Tracking 30-Sep-2015 14:33:35	←

App user access

All ▾

← **Select authorized app-users**

App user will be working...

- Online — Internet connection required.
- Offline — Has some limitations. [Learn more.](#)

← **Select option**

With each scan...

- Add new record.
- Edit existing record.
- Edit existing or add new record.
- Edit existing or add new record IF user inputs at least one value.

← **Select option**

Enable online search within app.









← **Default**

### Online vs. Offline

You have the option to scan and collect data online or offline. A stable Internet connection is required for online Services – the faster the better. The app user will be alerted if connectivity is lost and they can save scan records on the device. However they will not be able to view or edit the ‘visible’ fields you selected. The reason is simple: the app is looking into your QuickBase table online to know what to show.

If you choose offline, your app users will never see any visible fields except the Scan Value. Therefore offline Services are typically used when you simply want the app user to scan and collect ‘internal’ data (GPS, timestamps, etc.).

**IMPORTANT:** To show QuickBase fields to the app user when offline you must create those Services on codeREADr.com. Those offline scans can be exported from codeREADr.com and imported to QuickBase as a CSV file. It’s technically possible to auto-insert them into QuickBase but the wizard doesn’t yet support that option.

4 Scan records									All
	Tracking ID	Task 1	Pack Timestamp	Task 2	Pick Up Timestamp	Task 3	Delivery Timestamp	GPS Location	Comments
 	1Z47Y30A0394867186	Pack - Status Tracking	09-22-2015 06:27 PM	PickUp - Status Tracking	09-22-2015 06:45 PM	Deliver - Status Tracking	09-22-2015 06:47 PM	112-168 Everett St Allston, MA 02134 United States (42.357801,-71.137985)	
 	1Z6909230361357882	Pack - Status Tracking	09-22-2015 06:27 PM	PickUp - Status Tracking	09-22-2015 06:44 PM	Deliver - Status Tracking	09-22-2015 06:47 PM	112-168 Everett St Allston, MA 02134 United States (42.357812,-71.137964)	
 	1Z272E180370821829	Pack - Status Tracking	09-22-2015 06:27 PM	PickUp - Status Tracking	09-22-2015 06:44 PM	Deliver - Status Tracking	09-22-2015 06:47 PM	112-168 Everett St Allston, MA 02134 United States (42.357799,-71.137955)	
 	1Z7R44E20205300748	Pack - Status Tracking	09-22-2015 06:27 PM	PickUp - Status Tracking	09-22-2015 06:44 PM	Deliver - Status Tracking	09-22-2015 06:47 PM	112-168 Everett St Allston, MA 02134 United States (42.357804,-71.137926)	

The table above is an example of an offline service where the scans were auto-inserted into QuickBase. In this example, when scanning and collecting data (visible and internal) the app user didn't need to see any pre-populated data from the QuickBase online table. All they needed to see was the Scan Value (Tracking ID).

### Service Created

When finished with the wizard you can sign in to the codeREADr app with the authorized app use credentials selected for this Service. Note: making Service changes requires the app-user to exit the app and sign in again to see the new Service properties.

To see scans auto-populate to your QuickBase table, you'll need to refresh the screen.

### Contact

If you have any questions please email [support@codereadr.com](mailto:support@codereadr.com).

#