



# Locker API User Guide

Rev. 1.0 | November 2015

This document is subject to a Confidentiality Agreement and is intended solely for the person to whom it was delivered by Kannuu Pty Ltd. It is not to be reproduced or distributed to any other party without the express written consent of Kannuu.

# Contents

1. Overview.....	3
1.1 Service Architecture Diagram for Suggest Locker .....	3
2. Locker API Basics. ....	4
2.1 Creating and Maintaining Device Tokens .....	4
2.2 Creating and Linking User Tokens .....	6
2.3 Subsequent Launching of the Client.....	7
3. Basic Locker Setup .....	8
3.1 Create and List Lockers.....	8
3.2 Create and List Locker Items.....	9
3.3 Removing Locker Items .....	12
4. Use Case Scenarios.....	13
4.1 Recent Searches Locker .....	13
4.2 Bookmarks and Auto-Suggest Locker .....	14
4.3 User / Device Pair & Share.....	17

# 1. Overview

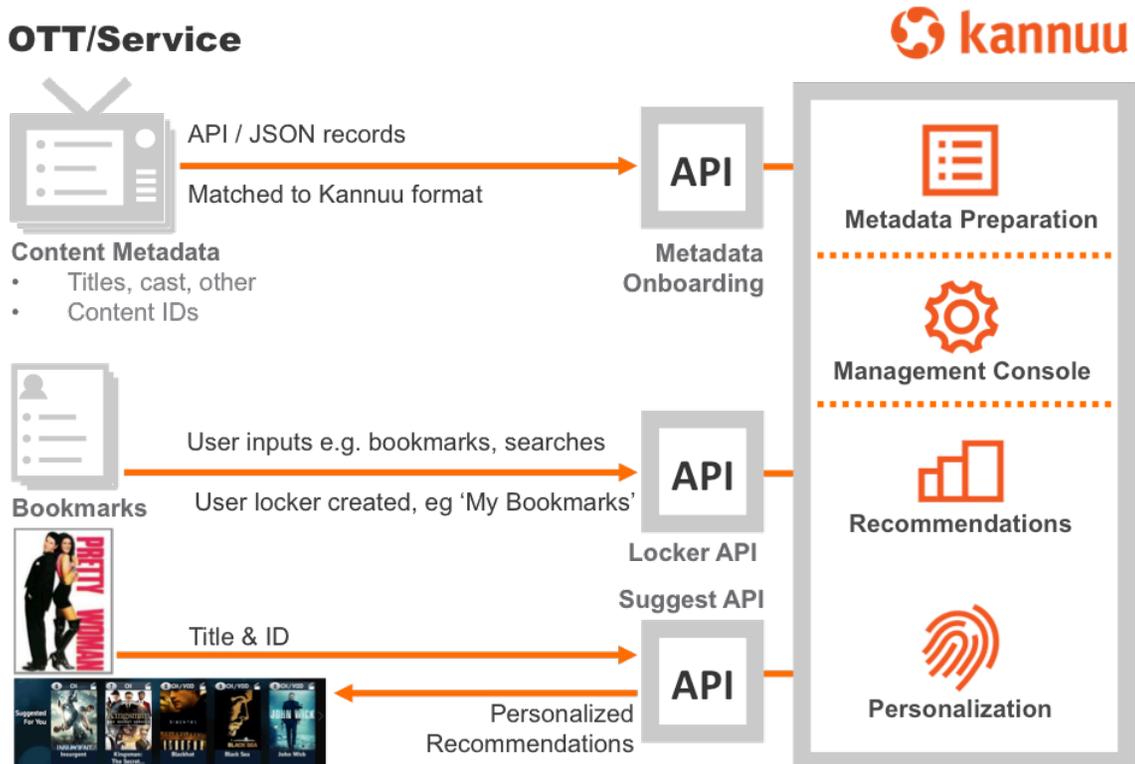
The Kannuu locker API is a RESTful web service that will give developers the ability to read and write movie, TV show, personality, and keyword data into organized “locker” containers.

The API is divided into two primary definitions: Devices and Users. A device represents the physical device and may be associated with multiple users. A user represents an individual person using the system and may be associated with multiple devices. A user in turn defines one or more lockers to store one or more items, where an item is a movie, TV show, personality, keyword, or search term. Multiple lockers can be used to easily organize items of a similar nature - e.g. "watch list" or "recent searches".

Responses from the API are in JSON (JavaScript Object Notation) format. A JSONP callback can also be utilized for cross-domain data access purposes.

The API must be accessed using a developer specific API key. Failure to use a valid API key in any request will result in a failure, and no data will be returned. Contact your Kannuu representative to be provided an API key for use with this API.

## 1.1 Service Architecture Diagram for Suggest Locker



## 2. Locker API Basics

In order to begin using the Kannuu locker API, first follow the examples below for creating a device and user token. Creating the device and user tokens is only required when setting up a new client.

### 2.1 Creating and Maintaining Device Tokens

Before a “locker” container can be created, a device and user must be created and linked together. A device represents a browser, set top box, connected IpTV device, etc., which is the device that is running the application software that makes use of the Kannuu APIs.

Every device must allocate and store its own device token ID and save it in local store. Alternatively the device may pass in a locally unique ID (e.g. MAC address) in order to have that ID associated with the Kannuu device token ID.

Sample Device Token ID API Call:

`http://api.kannuu.com/api/2/device/token?op=create&name=Living%20Room&api_key=DEVELOPERAPIKEY&jsonp=test`

```
{ "fts" : 1448385660468,
  "stat" : "ok",
  "response" : {
    "image" : "",
    "name" : "Living Room",
    "state" : "offline",
    "token" : "f470bd5fbf5daacd1ac92a694b24a2f1"
  }
}
```

Clients should also register and maintain a device in an “online” status by periodically using the watchdog API. A timer value (in seconds) provides a time-to-live (TTL) for the registration.

Sample Register / Watchdog API Call:

`http://api.kannuu.com/api/2/device/token?op=register&timer=30&token=f470bd5fbf5daacd1ac92a694b24a2f1&api_key=DEVELOPERAPIKEY&jsonp=test`

```
{ "fts" : 60,
  "stat" : "ok",
  "response" : {
    "image" : "",
    "name" : "Living Room",
    "state" : "online",
    "total_results" : 1
  }
}
```

[http://api.kannuu.com/api/2/device/token?op=watchdog&timer=30&token=f470bd5fbf5daacd1ac92a694b24a2f1&api\\_key=DEVELOPERAPIKEY&jsonp=test](http://api.kannuu.com/api/2/device/token?op=watchdog&timer=30&token=f470bd5fbf5daacd1ac92a694b24a2f1&api_key=DEVELOPERAPIKEY&jsonp=test)

```
{ "fts" : 60,
  "stat" : "ok",
  "response" : {
    "sq" : 0
  }
}
```

Client applications maintain an “online” status for a device by periodically using the above watchdog API in order to reset the registration TTL (to not let it expire). The watchdog API response also includes details on the device queue for receiving notification and user initiated events (when “response.sq” > 0)

Client applications unregister a device with the system when the application is done.

#### Sample Unregister API Call:

[http://api.kannuu.com/api/2/device/token?op=unregister&token=f470bd5fbf5daacd1ac92a694b24a2f1&api\\_key=DEVELOPERAPIKEY&jsonp=test](http://api.kannuu.com/api/2/device/token?op=unregister&token=f470bd5fbf5daacd1ac92a694b24a2f1&api_key=DEVELOPERAPIKEY&jsonp=test)

```
{ "fts" : 1448386080936,
  "stat" : "ok",
  "response" : {
    "image" : "",
    "name" : "Living Room",
    "state" : "offline",
    "total_results" : 1
  }
}
```

## 2.2 Creating and Linking User Tokens

A user must be created for each device and should be created for each unique user using the client. Note that login via facebook can also create a user token ID (contact your Kannuu representative for more details). A separate API then links the user to the device.

Sample Create User API Call:

[http://api.kannuu.com/api/2/user/token?op=create&name=Eval%20User&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/token?op=create&name=Eval%20User&api_key=DEVELOPERAPIKEY)

```
{ "fts" : 1448385858976,
  "response" : { "image" : "",
                 "name" : "Eval User",
                 "token" : "0e7d16450c2cb410142bdf4440d81bb1"
               },
  "stat" : "ok"
}
```

Sample Link User API Call:

[http://api.kannuu.com/api/2/user/token?op=link&token=0e7d16450c2cb410142bdf4440d81bb1&cid=f470bd5fbf5daacd1ac92a694b24a2f1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/token?op=link&token=0e7d16450c2cb410142bdf4440d81bb1&cid=f470bd5fbf5daacd1ac92a694b24a2f1&api_key=DEVELOPERAPIKEY)

(Note “cid” is the client or device token ID)

```
{ "fts" : 1448385858976,
  "stat" : "ok",
  "response" : {
    "devices" : 1,
    "id" : "0e7d16450c2cb410142bdf4440d81bb1",
    "image" : "",
    "name" : "Eval User",
    "results" : [
      { "id" : "f470bd5fbf5daacd1ac92a694b24a2f1",
        "image" : "",
        "name" : "Living Room"
      }
    ]
  }
}
```

Users remain linked to devices until the unlink API operation is performed.

### 2.3 Subsequent Launching of the Client

On subsequent client launches, the device and user token do not need to be recreated. Using the stored device token ID the list of associated users may be easily retrieved:

Sample Get Device Details API Call:

[http://api.kannuu.com/api/2/device/token?op=get&token=f470bd5fbf5daacd1ac92a694b24a2f1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/device/token?op=get&token=f470bd5fbf5daacd1ac92a694b24a2f1&api_key=DEVELOPERAPIKEY)

```
{ "fts" : 1448386080936,
  "stat" : "ok",
  "response" : {
    "image" : "",
    "name" : "Living Room",
    "results" : [
      { "id" : "0e7d16450c2cb410142bdf4440d81bb1",
        "image" : "",
        "name" : "Eval User"
      }
    ],
    "state" : "offline",
    "total_results" : 1
  }
}
```

The desired user can be selected from the results and subsequent locker information for that user can be used.

### 3. Basic Locker Setup

Once the user token has been created / retrieved, locker containers for that user can be created.

#### 3.1 Create and List Lockers

Lockers are created with a particular type value which is currently one of "manual", "autosuggest", "saved", "recents", or "bookmarks". Typically "manual" is used to indicate that the items in the locker are manually added and deleted through client activity.

Sample Create Locker API Call:

[http://api.kannuu.com/api/2/user/locker?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&api\\_key=DEVELOPERAPIKEY&type=manual&name=Saved&limit=100](http://api.kannuu.com/api/2/user/locker?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&api_key=DEVELOPERAPIKEY&type=manual&name=Saved&limit=100)

```
{ "fts" : 1448403680231,
  "stat" : "ok",
  "response" : {
    "locker" : {
      "id" : "1",
      "image" : "",
      "limit" : 100,
      "name" : "Saved",
      "share" : false,
      "total_results" : 0,
      "type" : "manual"
    }
  }
}
```

Note that the locker ID is assigned by the API server and this ID is used in other various API calls. Once the user lockers have been created the user token is used to fetch the list of lockers for a user.

Sample Get Locker Details API Call:

[http://api.kannuu.com/api/2/user/token?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/token?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&api_key=DEVELOPERAPIKEY)

```

{ "fts" : 1448403680231,
  "stat" : "ok",
  "response" : {
    "image" : "",
    "name" : "Eval User",
    "results" : [ {
      "id" : "1",
      "limit" : 100,
      "name" : "Saved",
      "share" : false,
      "total_results" : 0,
      "type" : "manual"
    } ],
    "total_results" : 1
  }
}

```

Lockers are persisted for a user until they are deleted. Removing a locker automatically removes all the items in that locker as well.

### 3.2 Create and List Locker Items

In the process of using the Kannuu Lookup API to search and browse items it may become desirable to store items / searches performed in a locker for later retrieval. From the lookup API results the ID of the item to save is used as well as a type value which is a combination of "{type}:{subtype}" where "type" is one of 'movie', 'tv', or 'search' and "subtype" is one of 'title', 'person', 'keyword', or 'channel'. If subtype is not specified then "title" is assumed.

Sample Create Locker Item API Call:

[http://api.kannuu.com/api/2/user/locker/item?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&locker=1&api\\_key=DEVELOPERAPIKEY&type=movie:title&id=12041904](http://api.kannuu.com/api/2/user/locker/item?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&locker=1&api_key=DEVELOPERAPIKEY&type=movie:title&id=12041904)

```

{ "fts" : 1448430252117,
  "response" : {
    "id" : "1",
    "item" : {
      "cast" : [ ],
      "category" : "movie",
      "cert" : "R",
      "director" : [ ],
      "genres" : [ "Crime" ],
      "id" : "12041904",
      "ids" : { },
      "image" : " http://image.kannuu.com/sWVslxlmw0770.jpg ",
      "label" : "{title}",
      "provider" : { },

```

```

        "rating_expl" : [ "Violence", "Adult Language"],
        "ratings" : { },
        "released" : "2015-01-16",
        "runtime" : 133,
        "synopsis" : "A brilliant hacker facing a 15-year prison
sentence is needed to prevent a deadly hacker's terrorist attack after
the code he wrote years ago is used to target China, putting him in a
power struggle between the U.S. and Chinese governments.",
        "title" : "Blackhat",
        "trailer" : { },
        "year" : 2015,
        "ymal" : [ ]
    },
    "type" : "movie:title"
},
"limit" : 100,
"name" : "Saved",
"share" : false,
"total_results" : 1,
"type" : "manual"
},
"stat" : "ok"
}

```

Items in the locker can be retrieved through the locker get operation. Summary JSON objects are returned for each item in the locker. To fetch the full item the locker item get operation is used.

#### Sample Locker Get API Call:

[http://api.kannuu.com/api/2/user/locker?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&id=1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/locker?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&id=1&api_key=DEVELOPERAPIKEY)

```

{ "fts" : 1448430252117,
  "response" : { "locker" : {
    "id" : "1",
    "limit" : 100,
    "name" : "Saved",
    "results" : [ {
      "id" : "12041904",
      "type" : "movie:title",
      "summary" : {
        "id" : "12041904",
        "image" : "http://image.kannuu.com/sWVslxlmw0770.jpg",
        "label" : "{title}",
        "released" : "2015-01-16",
        "title" : "Blackhat"
      }
    }
  ]
}
}

```

```

        } ],
        "share" : false,
        "total_results" : 1,
        "type" : "manual"
    } },
    "stat" : "ok"
}

```

#### Sample Locker Item Get API Call:

[http://api.kannuu.com/api/2/user/locker/item?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&locker=1&api\\_key=DEVELOPERAPIKEY&id=12041904](http://api.kannuu.com/api/2/user/locker/item?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&locker=1&api_key=DEVELOPERAPIKEY&id=12041904)

```

{ "fts" : 1448430252117,
  "response" : {
    "item" : {
      "cast" : [ ],
      "category" : "movie",
      "cert" : "R",
      "director" : [ ],
      "genres" : [ "Crime" ],
      "id" : "12041904",
      "ids" : { },
      "image" : " http://image.kannuu.com/sWVslxlmw0770.jpg ",
      "label" : "{title}",
      "provider" : { },
      "rating_expl" : [ "Violence", "Adult Language"],
      "ratings" : { },
      "released" : "2015-01-16",
      "runtime" : 133,
      "synopsis" : "A brilliant hacker facing a 15-year prison sentence
is needed to prevent a deadly hacker's terrorist attack after the code he
wrote years ago is used to target China, putting him in a power struggle
between the U.S. and Chinese governments.",
      "title" : "Blackhat",
      "trailer" : { },
      "year" : 2015,
      "ymal" : [ ]
    } },
    "stat" : "ok"
  }
}

```

The ID of the locker item is used to retrieve the full item details. The details returned are specific to the locker item that was stored.

### **3.3 Removing Locker Items**

Locker items must be removed with the same type and ID value that was used to create them.

## 4. Use Case Scenarios

### 4.1 Recent Searches Locker

Create a locker that will contain recent searches.

Sample Create Locker API Call:

[http://api.kannuu.com/api/2/user/locker?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&api\\_key=DEVELOPERAPIKEY&type=recent&name=Recent&limit=100](http://api.kannuu.com/api/2/user/locker?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&api_key=DEVELOPERAPIKEY&type=recent&name=Recent&limit=100)

```
{ "fts" : 1448403680231,
  "stat" : "ok",
  "response" : {
    "locker" : {
      "id" : "2",
      "image" : "",
      "limit" : 100,
      "name" : "Recent",
      "share" : false,
      "total_results" : 0,
      "type" : "recent"
    }
  }
}
```

The locker only needs to be created the first time. On subsequent uses the locker can be retrieved from the user token details:

[http://api.kannuu.com/api/2/user/token?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/token?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&api_key=DEVELOPERAPIKEY)

Locate the desired locker in the result list and the ID associated with it.

When determining that a search term is to be saved the client then executes the create locker item API to save the recent search in the locker.

Sample Add Recent Search Locker Item API Call:

[http://api.kannuu.com/api/2/user/locker/item?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&locker=2&id=IRON+M&type=search%3Atitle&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/locker/item?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&locker=2&id=IRON+M&type=search%3Atitle&api_key=DEVELOPERAPIKEY)

```
{
  "stat": "ok",
  "fts": 1448468518226,
  "response": {
    "name": "Recent",
```

```

    "total_results": 1,
    "share": false,
    "item": {
      "type": "search:title",
      "id": "IRON M",
      "summary": {
        "ts": 1448468518226
      }
    },
    "limit": 100,
    "type": "recents",
    "id": "2"
  }
}

```

When saving search terms use a type value with “search” as the first part of the type, and put the search query string in the ID field.

Once the locker limit is reached, old items will be automatically removed in a FIFO fashion as new items continue to be added.

## 4.2 Bookmarks and Auto-Suggest Locker

Create a locker to store bookmarked items, and then create an auto-suggest locker that gets updated automatically when items are added / removed from the bookmark locker.

Sample Create Locker API Call:

[http://api.kannuu.com/api/2/user/locker?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&api\\_key=DEVELOPERAPIKEY&type=bookmarks&name=Bookmarks&limit=100](http://api.kannuu.com/api/2/user/locker?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&api_key=DEVELOPERAPIKEY&type=bookmarks&name=Bookmarks&limit=100)

```

{
  "stat": "ok",
  "fts": 1448469438085,
  "response": {
    "locker": {
      "name": "Bookmarks",
      "total_results": 0,
      "image": "",
      "share": false,
      "limit": 100,
      "type": "bookmarks",
      "id": "3"
    }
  }
}

```

On subsequent executions the bookmarks locker may simply be retrieved from the user token details.

[http://api.kannuu.com/api/2/user/token?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/token?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&api_key=DEVELOPERAPIKEY)

```
{
  "stat": "ok",
  "fts": 1448469438085,
  "response": {
    "total_results": 3,
    "image": "",
    "name": "Eval User",
    "results": [
      {
        "name": "Saved",
        "total_results": 0,
        "share": false,
        "limit": 100,
        "type": "manual",
        "id": "1"
      },
      {
        "name": "Bookmarks",
        "total_results": 0,
        "share": false,
        "limit": 100,
        "type": "bookmarks",
        "id": "3"
      },
      {
        "name": "Recent",
        "total_results": 1,
        "share": false,
        "limit": 100,
        "type": "recents",
        "id": "2"
      }
    ]
  }
}
```

Creating an auto-suggest locker occurs in a similar manner, though in this situation the auto-suggest locker needs to be told which other locker to watch for updates.

Specify the locker ID to watch and the autosuggest locker refresh interval (in seconds) in the 'tcfg' parameter (comma delimited).

Sample Create Auto-Suggest Locker API Call:

[http://api.kannuu.com/api/2/user/locker?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&api\\_key=DEVELOPERAPIKEY&type=autosuggest&name=AutoSuggest&tcfg=3,86400](http://api.kannuu.com/api/2/user/locker?op=create&token=0e7d16450c2cb410142bdf4440d81bb1&api_key=DEVELOPERAPIKEY&type=autosuggest&name=AutoSuggest&tcfg=3,86400)

```
{
  "stat": "ok",
  "fts": 1448470265485,
  "response": {
    "locker": {
      "name": "AutoSuggest",
      "total_results": 0,
      "image": "",
      "share": false,
      "limit": 25,
      "type": "autosuggest",
      "id": "4"
    }
  }
}
```

As things are added to the bookmarks folder then related, suggested items are automatically added to the auto-suggest locker as well (this is the function of the auto-suggest locker type). The locker may simultaneously store many different items from movie and tv shows to people, keywords, and channels. In order to retrieve specific types of items the “filter” parameter may be used with the get locker details API call. Also specifying “type=autosuggest” provides an alternative way to retrieve the auto-suggest locker items in lieu of looking up the actual ID of the locker.

Sample Retrieve Auto-Suggest Locker Items API Call:

[http://api.kannuu.com/api/2/user/locker?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&type=autosuggest&filter=movie:title&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/locker?op=get&token=0e7d16450c2cb410142bdf4440d81bb1&type=autosuggest&filter=movie:title&api_key=DEVELOPERAPIKEY)

```
{
  "stat": "ok",
  "fts": 1448470265485,
  "response": {
    "locker": {
      "name": "AutoSuggest",
      "total_results": 25,
      "share": false,
      "results": [
        {
          "type": "movie:title",
          "id": "12041883",

```

```

        "summary": {
            "released": "2014-12-05",
            "image": "http://image.kannuu.com/ 4MlKSLlFUus98N.jpg",
            "label": "{title}",
            "id": "12041883",
            "title": "Black Sea"
        }
    },
    .
    .
    .
],
"limit": 25,
"type": "autosuggest",
"id": "4"
}
}
}
}

```

Autosuggest lockers default to a limit of 25 of each type of item (movie:title, tv:title, movie:person, etc.) that is stored in the locker. The limit can be adjusted using the locker update operation API.

### 4.3 User / Device Pair & Share

The pair & share API behavior allows a user to be easily added to another device (e.g. TV) by requesting a PIN from that device and providing a means for the user to enter the PIN value. PIN values are temporarily allocated by the server and associated with the device token.

On the device to link to, allocate a PIN using the following API and display it for the user.

Sample Create Auto-Suggest Locker API Call:

[http://api.kannuu.com/api/2/device/token?op=pin&token=f470bd5fbf5daacd1ac92a694b24a2f1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/device/token?op=pin&token=f470bd5fbf5daacd1ac92a694b24a2f1&api_key=DEVELOPERAPIKEY)

```

{
    "stat": "ok",
    "fts": 1448475099552,
    "response": {
        "expires": 1200000,
        "pin": "166310"
    }
}

```

On the secondary device (e.g. tablet) that the user is currently making use of a form should be displayed where the user can enter the PIN value for the device

they want to link to. Once the user enters the PIN value the following API must be called to link the user to the device. Note in this API call that the "token" is the user token ID value.

Sample Link User by PIN API Call:

[http://api.kannuu.com/api/2/device/auth?token=073a220859b81ac69fa89464e0cd2bbe&pin=166310&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/device/auth?token=073a220859b81ac69fa89464e0cd2bbe&pin=166310&api_key=DEVELOPERAPIKEY)

```
{
  "stat": "ok",
  "fts": 1448474457675,
  "response": {
    "device": {
      "image": "",
      "id": "f470bd5fbf5daacd1ac92a694b24a2f1",
      "name": "Living Room",
      "users": 2
    }
  }
}
```

This links the user to the secondary device and provides this user with the device token ID of the device. The device itself may query its users and would discover the new user that has been linked.

[http://api.kannuu.com/api/2/device/token?op=get&token=f470bd5fbf5daacd1ac92a694b24a2f1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/device/token?op=get&token=f470bd5fbf5daacd1ac92a694b24a2f1&api_key=DEVELOPERAPIKEY)

```
{
  "stat": "ok",
  "fts": 1448474469354,
  "response": {
    "total_results": 2,
    "image": "",
    "results": [
      {
        "image": "",
        "id": "0e7d16450c2cb410142bdf4440d81bb1",
        "name": "Eval User"
      },
      {
        "image": "",
        "id": "073a220859b81ac69fa89464e0cd2bbe",
        "name": "Second User"
      }
    ],
    "name": "Living Room",
  }
}
```

```
    "state": "online"
  }
}
```

With the newly connected device token ID the user may choose to share search results with that device or have it play a trailer, etc. In order to share an item the /user/send API is used, where 'token' is the user token ID, 'cid' is the device token ID, with 'id' and 'type' representing the item that is to be sent to the device. An 'op' parameter value indicates the type of operation the device should perform with the item.

Note that the device must be in an "online" state for the send API to be successful.

Sample User Item Share API Call:

[http://api.kannuu.com/api/2/user/send?op=show&token=073a220859b81ac69fa89464e0cd2bbe&cid=f470bd5fbf5daacd1ac92a694b24a2f1&id=12041904&type=movie:title&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/user/send?op=show&token=073a220859b81ac69fa89464e0cd2bbe&cid=f470bd5fbf5daacd1ac92a694b24a2f1&id=12041904&type=movie:title&api_key=DEVELOPERAPIKEY)

```
{
  "stat": "ok",
  "response": {
    "remaining": 1
  }
}
```

Any additional parameters passed with the send API will be forwarded to the device.

The device is required to poll for items in its share queue:

[http://api.kannuu.com/api/2/device/sq?token=f470bd5fbf5daacd1ac92a694b24a2f1&api\\_key=DEVELOPERAPIKEY](http://api.kannuu.com/api/2/device/sq?token=f470bd5fbf5daacd1ac92a694b24a2f1&api_key=DEVELOPERAPIKEY)

```
{
  "stat": "ok",
  "response": {
    "remaining": 0
  }
}
```

When there are items on the queue to process the response will return the top item, who it is from, and indicate the remaining count.

```
{
  "stat": "ok",
  "response": {
    "uid": "073a220859b81ac69fa89464e0cd2bbe",
    "type": "movie:title",
    "id": "12041904",
    "op": "show"
  }
}
```

```
    "item": {  
        ...  
    },  
    "remaining": 0  
  }  
}
```

The items details are also retrieved and returned with the response. Note that any 'op' value understood between the two clients may be used.