



x u a n

Xuan (The STACK Box) Application Development

Table of Contents

A) Xuan Platform.....	3
B) Basic Xuan Application Development Process.....	3
C) Basic UI Only Xuan Application.....	3
D) Basic UI and Daemon (Python) Xuan Application.....	4
E) Advance Xuan Application Development Process.....	8

A) Xuan Platform

1. First of all, if you are not familiar with the Xuan platform architecture, please refer to developer.thexuan.com.
2. You can develop HTML app (UI) running on the “Xuan App” (iOS/Android) and make HTTPS API request to the STACK Box (developer.thexuan.com/documentation.html).
3. You can develop Python app (Daemon) running on the STACK Box itself. Daemon app can extend the STACK Box features and expose more HTTPS API to external program. It can also call to other daemon app such as system app (2000100) through IPC.
4. You also can develop external program (e.g. iOS, Android and etc) and make HTTPS API request to the STACK Box. Xuan developer account is not required for this type of program.
5. For more complex and special corporate project business requirements, please email to xuan@cloudmedia.com for enquiries.

B) Basic Xuan Application Development Process

1. Get a STACK Box from cloudmedia.com.
2. Register Xuan developer account at apps.thexuan.com.
3. Develop a new Xuan application, zip and upload it to apps.thexuan.com.
4. Create the Private Token to install and test the application on the STACK Box.
5. Make changes on the application, zip and use the EDIT function to re-upload the app.
6. Use the Private Token to re-install and test the updated application.
7. Repeat steps 5 and 6 above to develop and test the application on STACK Box or use “Advance Xuan Application Development Process” found later on this document.
8. Submit the application for approval.
9. Once approved, the end users will be able to see and install the application from their “Xuan App” (iOS/Android) Apps Market to their STACK Box.
10. Then, use the UPGRADE function when new version of the application is developed.

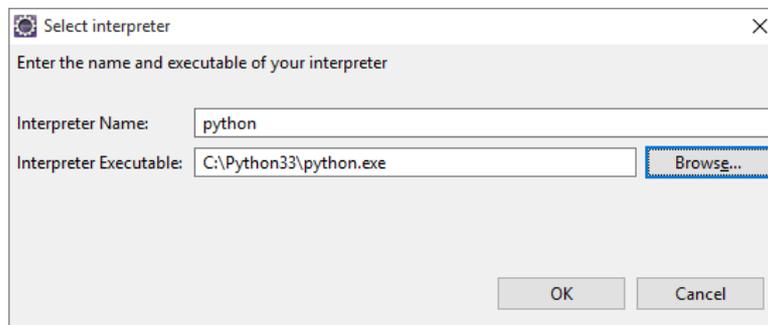
C) Basic UI Only Xuan Application

1. You can use any tools you are familiar with to develop your UI only application (e.g. HTML5) that can run on “Xuan App” (iOS/Android) embedded web-browser.
2. The UI can make HTTPS API request to the STACK Box (developer.thexuan.com/documentation.html). For example:

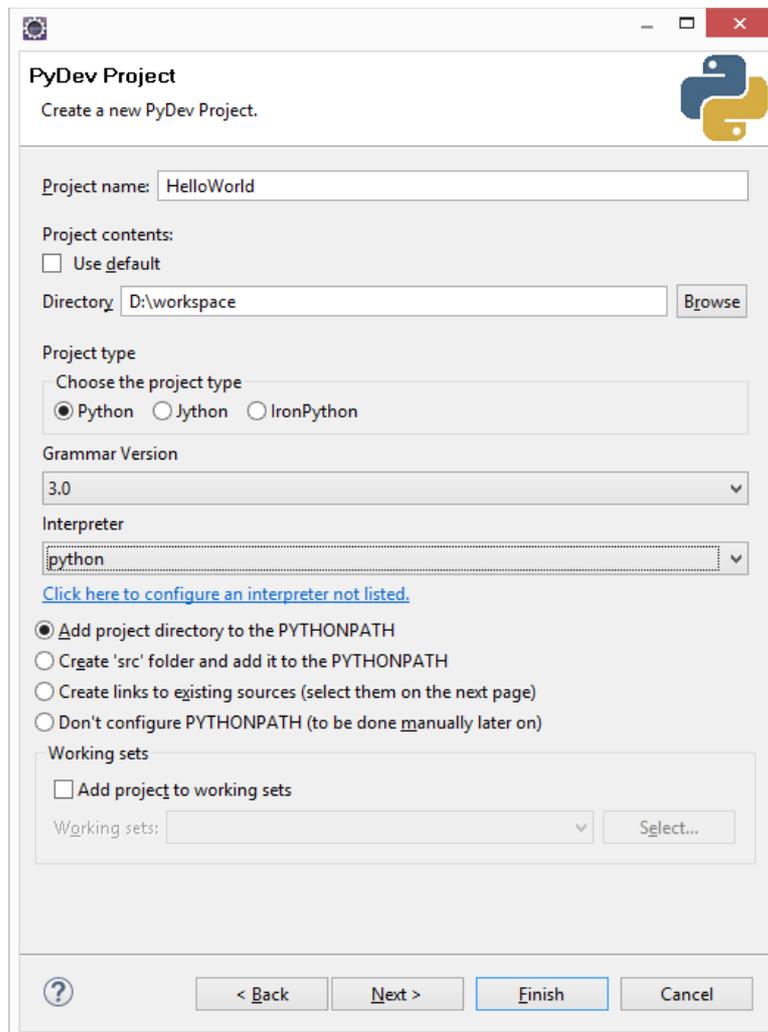
```
https://{BOX_IP}/cgi-bin/syb_reqresp.cgi?  
app_id=2000100&module={MODULE_NAME}&method={METHOD_NAME}
```
3. The entry file web/index.html is required for UI application. Download the [developer.thexuan.com/HelloWorld_ui_only_sample.zip](http://developer.thexuan.com>HelloWorld_ui_only_sample.zip) to see the example.
4. Finally, zip and upload it to your Xuan developer account (apps.thexuan.com).

D) Basic UI and Daemon (Python) Xuan Application

1. Download and install Python 3.3.3 from python.org
2. Download and install latest Eclipse from eclipse.org and install “PyDev” plugin from Help => Eclipse Marketplace.
3. Go to **Window => Preferences => PyDev => Interpreters => New...**

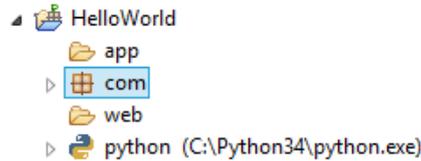


4. Go to “PyDev Package Explorer” view to create a PyDev Project.

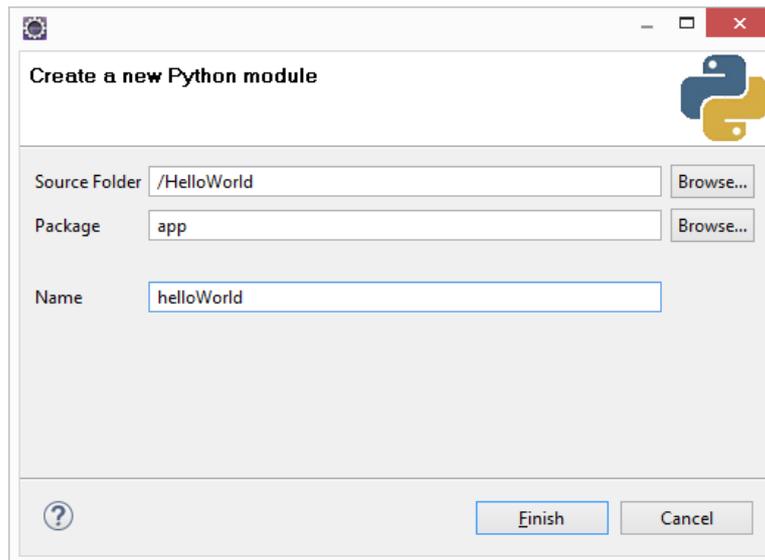


Xuan (The STACK Box) Application Development

5. Create 2 folders “web” for UI and “app” for daemon.
6. Download the latest Xuan SDK from <https://github.com/TheStackBox/xuansdk> and extract it. Copy the “com” folder from SDKLibrary to the HelloWorld Project.



7. Create a new PyDev module. Then select <Empty> Template.



8. Download developer.thexuan.com/HelloWorld_ui_daemon_sample.zip, extract it and copy the app/helloWorld.py content to your helloWorld.py. Indentation is important for Python, make sure the 4 and 8 spaces indentation are placed correctly.

```
*helloWorld.py
1 from com.cloudMedia.theKuroBox.sdk.app.application import Application
2 from com.cloudMedia.theKuroBox.sdk.paramTypes.kbxString import KBXString
3
4 class HelloWorld(Application):
5     def on_system_connected(self):
6         super().on_system_connected()
7         self.register_method("say_my_name", self.say_my_name, kbxMethodParams=[
8             self.register_shared_method("say_my_name", self.say_my_name, kbxMethodI
9
10    def post_system_connected(self):
11        super().post_system_connected()
12
13    def say_my_name(self, request):
14        name = request.get_arg("name")
15        print("Welcome " + name + "!")
16        self.send_response({"text": "Welcome " + name + "!"}, request.requestId)
```

9. When you extend “Application” class, “on_system_connected” and “post_system_connected” must be overridden and super().on_system_connected() and super().post_system_connected() must be included respectively.

10. "self.register_method" is used to register method that will be exposed through HTTPS. For example:

`https://{BOX_IP}/cgi-bin/syb_reqresp.cgi?app_id={APP_ID}&method=say_my_name&name=My Name`

11. "self.register_shared_method" is used to register method that will be exposed through IPC to allow other daemon Python app to call to.

`SharedMethod.call(kbxMethodAppId={APP_ID}, kbxMethodName="say_my_name", name="My Name")`

12. "print" is used to print text on the console or terminal for mainly debugging purposes. For more information, please refer to "Advance Xuan Application Development Process" found later on this document.
13. "self.send_response" is used to send response back to the UI through HTTPS or back to other Python app through IPC.
14. At this time, you need to create an application on apps.thexuan.com to get one {APP_ID} to allow UI to call to your registered method (self.register_method) through HTTPS.
15. After login through the apps.thexuan.com, go to "My Applications" and click on "Upload New Application".



16. Enter the information of your new application. (Some fields are optional)

A screenshot of the 'Upload New Application' form. The header says 'Welcome demo@thexuan.com!'. Below the header, it says 'Upload New Application'. A note reads: '* NOTE: If you wish to Update or Upgrade an existing application, please click on [Application Listing](#) and use the EDIT or UPGRADE button on the Details page'. The form has the following fields:

- Name * (89) : Hello World
- Type : UI and Daemon (dropdown menu)
- Version * (10) : 1.0.0
- Author * (91) : Xuan Demo
- Author URI : (empty field)

17. Then you can get the {APP_ID} from the Application Details page.

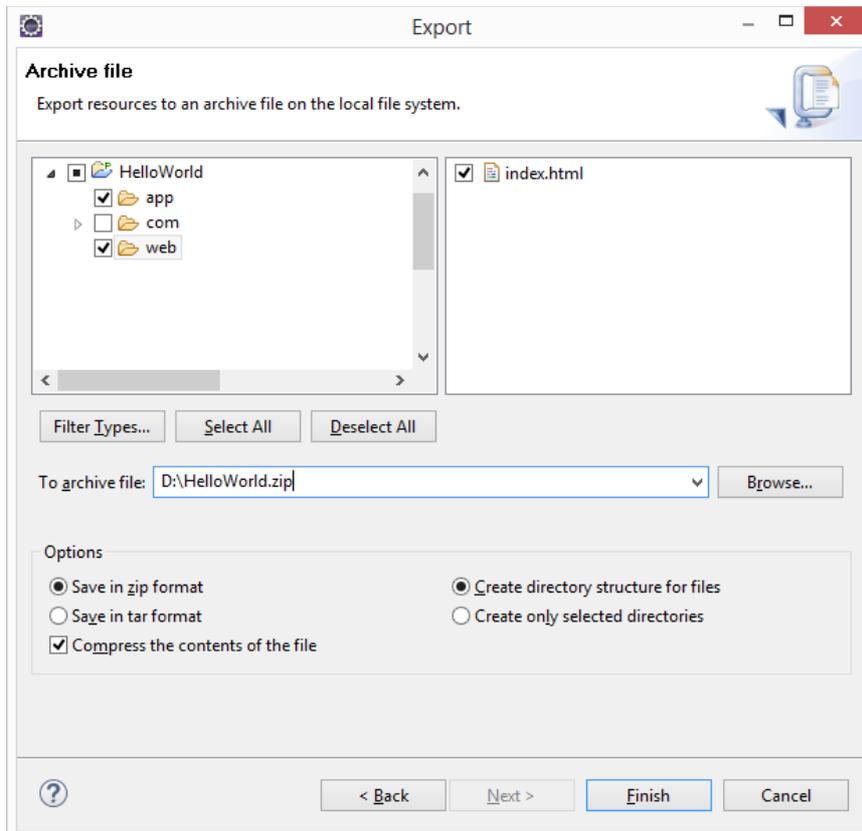
A screenshot of the 'Application Details' page. It shows the following information:

- App ID : 3178489
- App Version : Hello World - 1.0.0 (New) (dropdown menu)
- Private Token : (empty field)

18. Back to Eclipse, create entry file index.html in “web” folder and copy the content from HelloWorld_ui_daemon_sample.zip /web/index.html to your newly created index.html.



19. Change the {APP_ID} in the index.html to the App ID shown on your Application Details page and change the MY_NAME to your name.
20. Select “app” and “web” folders, right click and select “Export...”. Then, select **General** => **Archive File** and click on “Next >” button. (SDK library “com” should not be included)



21. Go back to the Application Details page, click on the EDIT button right in the middle of the page. On “Edit Application Details” page, select the HelloWorld.zip on “Application” field and type “app.helloWorld.HelloWorld” (packages.pyFilename.ClassName) on “App Entry Point” field. Finally, click on the UPDATE button at the bottom of the page.



22. On Application Details page, click on the “Create” link to create a Private Token.

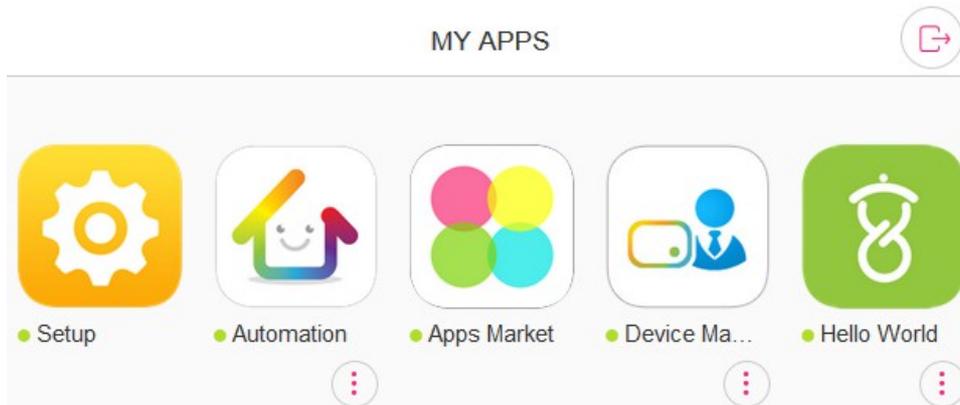
App ID : 3178489
App Version : Hello World - 1.0.0 (New) ▾
Private Token : (Create)
Category : App

23. Go to the “Xuan App” **MY APPS** => **Apps Market**, click on the top right icon  and key in the PRIVATE APP Token to install the app onto the STACK Box.



A screenshot of a form titled "PRIVATE APP". It features a text input field with a key icon on the left and the placeholder text "|Token". Below the input field is a "Submit" button.

24. Go to MY APPS and test the newly created app.



E) Advance Xuan Application Development Process

1. Get a STACK Box from cloudmedia.com.
2. Register Xuan developer account at apps.thexuan.com.
3. Develop a new Xuan application, zip and upload it to apps.thexuan.com.
4. Create the Private Token to install the application on the STACK Box.
5. Create NFS shared folder of your app on your PC. For Windows, you can try some tools like freenfs.sourceforge.net.
6. Use your developer account email to send your STACK Box's MAC Address to xuan@cloudmedia.com to request for telnet password.
7. Telnet (e.g. using putty) to the STACK Box to access to the console/terminal.
8. Mount to your PC NFS shared folder. For example, if the HelloWorld project path is D:\workspace\HelloWorld, and NFS Root Folder Path is set to D:\workspace:

```
# mkdir /tmp/HelloWorld  
# mount -o nolock {PC_IP}:/HelloWorld /tmp/HelloWorld
```

9. Then make symbolic links to the mounted PC NFS shared folders:

```
# mv /install/python/{APP_ID}/app /install/python/{APP_ID}/app_ori
# ln -s /tmp/HelloWorld/app /install/python/{APP_ID}/app
# mv /install/python/{APP_ID}/web /install/python/{APP_ID}/web_ori
# ln -s /tmp/HelloWorld/web /install/python/{APP_ID}/web
```

10. Put a thumbnail image to your web folder (e.g. D:\workspace\HelloWorld\web\thumbnail.png or thumbnail.jpg) just in case you want to see a proper thumbnail on the “Xuan App” MY APP.
11. At this time, changes made on folders and files under “web” (e.g. D:\workspace\HelloWorld\web) should be immediately reflected on the “Xuan App” or web-browser once refreshed.
12. Restart the python app whenever you make any changes to the Python file before testing. Grab the Process ID (PID) and python command from “ps”. Then, “kill” the app and re-run the python command after making sure the app is killed. For example:

```
# ps -ef

12152 root      python /install/python/3178489/index -uid 3178489 -appdir
/install/python/3178489

# kill 12152
# ps -ef
# python /install/python/3178489/index -uid 3178489 -appdir
/install/python/3178489
```

13. After re-run the python command, the app will be running on the console/terminal foreground, meaning you will be able to see all the python “print” function output on the console/terminal. Use CTRL+c to break the app from the console/terminal foreground.
14. Repeat steps 11 to 13 above to develop and test the application on the STACK Box.
15. Submit the application for approval.
16. Once approved, the end users will be able to see and install the application from their “Xuan App” (iOS/Android) Apps Market to their STACK Box.
17. Then, use the UPGRADE function when new version of the application is developed.