



# **FoxInCloud**

# **MANUALLY INSTALLING A DEVELOPMENT FOXINCLOUD APPLICATION SERVER**

How-To Guide

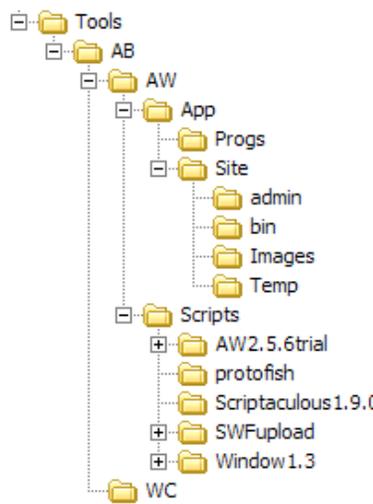
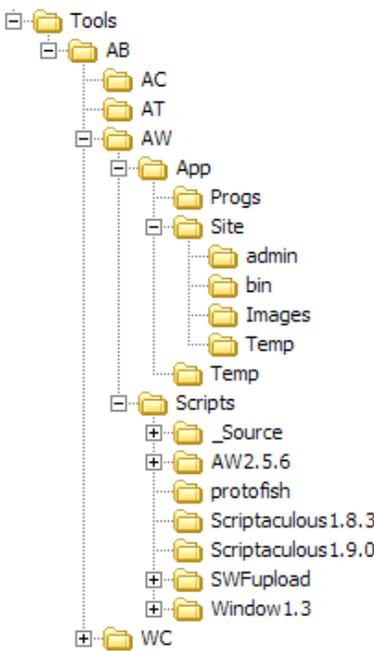
Applicable to FoxInCloud Application Server V. 1.20+

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# TARGET AUDIENCE

This document explains how to setup your FoxInCloud Application Server (**FAS**) and web site to developers who have:

- Adapted their application using FoxInCloud Adaptation Assistant (**FAA**),
- Downloaded and installed FoxInCloud Application Server, installing the following directory structure in `<VFP9>\Tools\`:

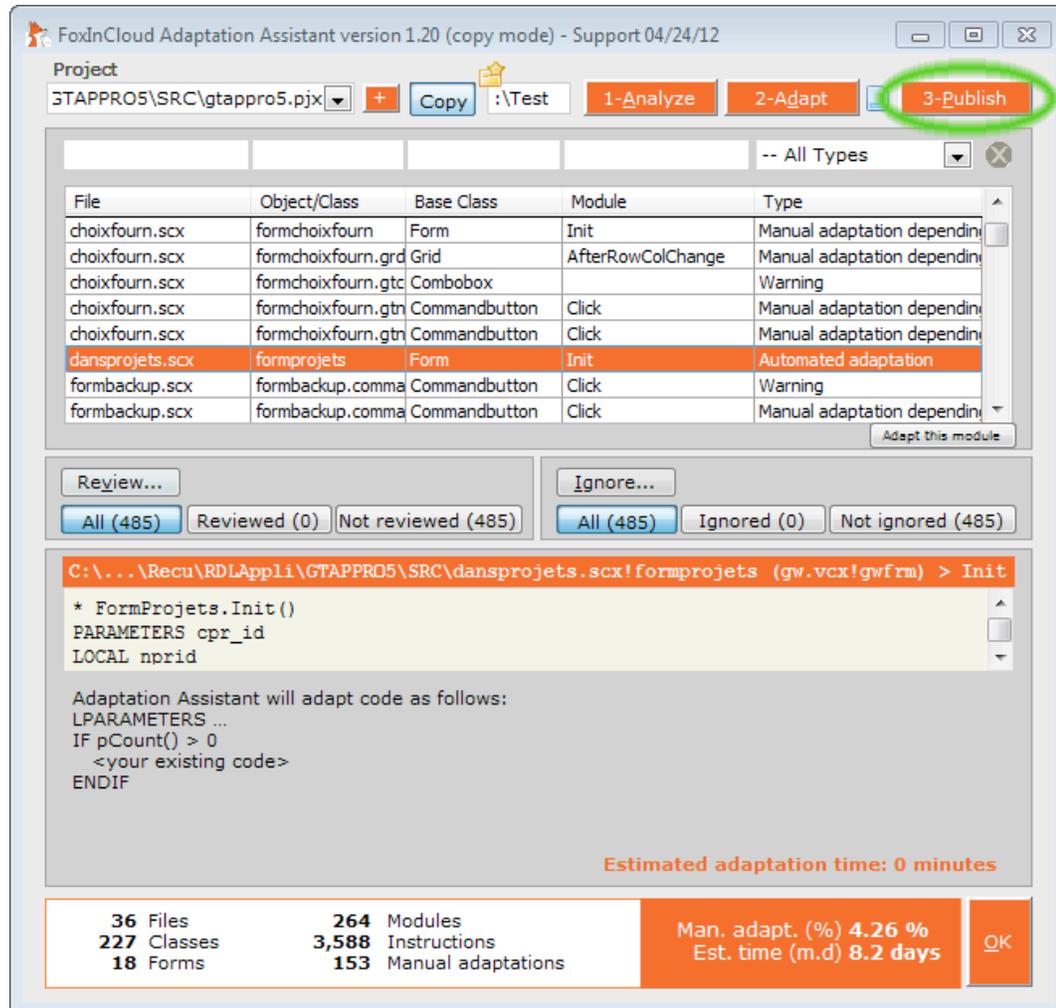
	Trial License	Full License
		
<code>AB\</code>	FoxInCloud public modules (previously installed by <b>FAA</b> )	
<code>\AC\</code>	N/A	FoxInCloud utilities source code
<code>\AT\</code>	N/A	FoxInCloud tools source code
<code>\AW\</code>	FoxInCloud & West-Wind Web Connect compiled code	FoxInCloud Application Server source code
<code>\Scripts\</code>	FoxInCloud JavaScript libraries	
<code>\WC\</code>	West-Wind Web Connect dlls	West-Wind Web Connect source code
<code>\App\</code>	Template files for your FoxInCloud Application Server	

[www.foxincloud.com](http://www.foxincloud.com) | [contact@foxincloud.com](mailto:contact@foxincloud.com)

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# AUTOMATED INSTALLATION

Starting with version 1.20, FoxInCloud Adaptation Assistant offers an **automated Development Server installation** feature with button “3-Publish”



This document focuses on manual installation if, for whatever reason, automated installation fails to deliver the expected result.

# THE BIG PICTURE

If ever you wonder what are the interactions between the browser, the web server, West-Wind Web Connect (wConnect), FoxInCloud and your application, this page will try to give you an overview.

First of all, keep in mind that 2 areas are clearly distinct: the site on one hand, your VFP-based Application Server on the other hand. wConnect acts as a link between these areas, we'll explain later how.

The web server receives requests HTTP on your site; an HTTP request basically consist in an URL, possibly a query string adding parameters to this URL, headers defining the context of the request, a command verb such as 'GET' or 'POST' indicating the action requested and, in case of a POST, data that should be posted to the site.

With the wiring explained further in this document, the web server transmits this request to the wConnect extension installed in your site, '[wc.dll](#)'.

[wc.dll](#) passes this request to the VFP-based Server built on top of wConnect's VFP base classes: wwServer, wwRequest and wwResponse.

As the first layer on top of wConnect, FoxInCloud receives request through the wwRequest object, decodes it, and identifies the application's form on which it applies. FoxInCloud instantiates this form if not already, runs the event method corresponding to the request received, identifies the changes in the form's appearance during the request, sends back display update orders to a wwResponse object, which sends back to [wc.dll](#), which sends it back to the web server, and ultimately to client browser.

The last missing piece is what the client browser actually displays: HTML built by FoxInCloud from the form itself, and javascript that implements form's events into the client browser. This HTML is part of a regular HTML page, built either with your subclass of FoxInCloud's awProcess class (AS wwProcess), or with any other tool such as ASP, ASPx, PHP, etc.

## PREREQUISITES

### Windows XP: turn on advanced security

By default, Win XP hides the 'Security tab' on folder and files property windows. This will be a handicap, as you'll often need to check NTFS access rights.

To switch to 'advanced security', go to Windows Explorer > Tools > Folder Options > Display tab; in the 'advanced parameters', uncheck 'simple security' near the bottom of the list.

### IIS installed on your machine

Because West-Wind Web Connect favors IIS, and because most VFP developers have a long experience with Windows as OS, this document focuses on IIS web server, bundled with all professional versions of Windows.

To check if IIS is installed on your development machine, go to 'Add and Remove programs', Windows Components, check 'Internet Services (IIS)' if not already. IIS versioning is as follows: Win2K: 5, WinXP: 5.5, Vista: 7, Win7: 7.5

If you run Windows Vista, 7 or 2008 server, IOW IIS $\geq$ 7, please check out wConnect documentation to make sure all the necessary components are installed: [http://www.west-wind.com/webconnection/docs/\\_22f0xkbmq.htm](http://www.west-wind.com/webconnection/docs/_22f0xkbmq.htm)

### Pick a decent development browser

While FoxInCloud supports any current browser, for development purpose it is more convenient to stick with a single browser, preferably with convenient development tools.

In this area FireFox is our preferred choice, with and because of the 'Firebug' extension. Take some time to play with Firebug, it looks like it was inspired by your VFP IDE.

## Prerequisites ...

### Understand the concept of virtual directory

A web server has its own view of folders and files on disk; actually it builds a shadow representation of what you include in your site, the 'metadata'.

By default the web server uses the original files/folders names and hierarchy, but you can at any point change these to turn into a 'virtual' directory or file.

Furthermore, you can attach properties to these 'virtuals', such as operations the web server is allowed to do, redirection to another virtual, headers that should be always be attached to the response, and so on...

Web servers treat this metadata hierarchically, in other words properties attached to a given element cascade to descendant virtual elements.

### Understand impersonation

A second, and probably the trickiest, abstraction is the concept of impersonation: under which NTFS account services and applications act. We don't intend here to give you a complete explanation on how Windows and Services manage impersonation, just give you some field-level bits of experience on how this should be taken care of, and how much attention should be dedicated to this issue that often reveals a pitfall.

Each member of the interaction chain described earlier is subject to impersonation, in other word may run with a different account than its predecessor. Tracking this chain of impersonations is crucial with wConnect.

The Web server, e.g. IIS, runs under a certain account. When IIS receives an anonymous GET request, it reads the required resource on disk (say a HTML file) using a dedicated account named '`IUSR_<machine name>`'. You should make sure this account has read credentials to the targeted resource.

### *Prerequisites ...*

When run by IIS, **wc.dll**, uses IIS's account if running in IIS process (standard with IIS5), or a different account if running under an 'application pool'. Knowing which account wc.dll runs is important as well.

In development, file-based mode, your FoxInCloud Application Server runs under the 'interactive account', yours, exactly like when you run your LAN application in VFP IDE or as a .app/.exe; this should not raise any credential issue.

In production, COM-based mode (our of the scope of this document but we raise this issue as a reference), the account under which your FoxInCloud Application Server will run will depend on the COM object settings. You will be able to manage these settings by running **DCOMcnfg.exe**.

We just tried to give you an overview of what impersonation involves and how you can apprehend and tweak possible issues. You will find extensive explanations and examples in [wConnect on line documentation](http://www.west-wind.com/webconnection/docs/) (<http://www.west-wind.com/webconnection/docs/>), take enough time to review this material if you don't feel familiar with impersonation.

### Use FoxInCloud Studio shortcut

Completing the steps in this documents requires having **CurDir()**, **Set('Path')**, **Set('Procedure')** and **Set('ClassLibrary')** adjusted to FoxInCloud resources, as achieved by the **aw\*.fpw** configuration file linked to the FoxInCloud Studio shortcut.

Please always launch VFP with the '**FoxInCloud Studio**' shortcut on your desktop and make sure you have no contradictory **SET**ting in either the Registry (VFP menu > Tools > Options > 'Set as Default'), and/or a **\_Startup** program like **VFPStart.prg**.

Preferably perform **SET ... ADDITIVE**.

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# STEPS INVOLVED

Most of the steps described hereafter shall probably seem obvious to developers already familiar with West-Wind Web Connect; this document rather targets developers having no or very little experience with West-Wind Web Connect.

Three meta steps are actually at stakes:

STEP	Skills involved	Timing est.
A- Building your FoxInCloud Application Server	VFP programming .ini file configuration	2 hours
B- Setting up your Application site	IIS virtual directories Script-mapped extensions NTFS security Impersonation	1 hour
C- Building and running your Application Forms in HTML pages	HTML CSS	1.5 hour

These steps are further detailed in the rest of this document.

## STEP A1: CHOOSE AN APP CODE

5 minutes

As you'll end up having many files in various directories for your installation, we encourage you to choose a 2 or 3-letters application code that will prefix most of the files and classes you're going to deal with.

Your choice is quite vast; just avoid `'AB'`, `'AC'`, `'AT'`, `'AW'` or `'WC'` as they are already used in the FoxInCloud namespace.

For instance, if your application deals with invoice payment, you may choose `'IP'`, or `'IPW'` if your LAN version already uses the `'IP'` as code/prefix.

Remember that FoxInCloud Adaptation Assistant already asked you for a class prefix? It might be a good idea to choose the same code or something derived from it. You can retrieve this code by browsing to the class library created while adapting your application.

So your application code is (*drumroll*) ...



In the rest of this document and the source files supplied, we'll repeatedly refer to `'xxx'` as a placeholder for the application code you've just chosen.

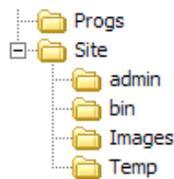
## STEP A2: COPY FILES TO YOUR APP'S DIR

10 minutes

All FoxInCloud Application Server files should reside in the **same folder** as you regular, LAN application. This is true for both VFP source code files and 'site' files in development <sup>1</sup>.

To achieve that, just copy (not move) files under `<VFP9>\Tools\AB\AW\App\` directory into the folder where your application project sits, and probably your `appMain.prg`.

Your app folder now has the following directories added:



... and several files added: `xxxTest*. *`

**Rename** all files in all directories by replacing '`xxx`' by your application code. Please make sure all other files are renamed accordingly in `Progs\` and `Site\` folders.

Also **rename** class name and file references in the various `.prg` accordingly.

After renaming files in the `Progs\` folder, move them to the folder where you usually store your app's `.prgs` and delete the now useless `Progs\` folder.

---

<sup>1</sup> Conversely, in production, Application and Site will reside in 2 isolated directories; we recommend installing Application into standard 'Program files' folder, and site into standard 'Inetpub' folder. This will be further explained by another how-to guide to come 'Installing a Production FoxInCloud Application Server.'

## STEP A3: CREATE YOUR APP ENV. SETTER

30 minutes

It is important that your application evolves in **exactly the same environment in either LAN or Web mode.**

To achieve that, we recommend that you cut & paste the **SET** instructions you currently have at the beginning of your LAN project main program into a separate .prg shared by LAN and Web version.

FoxInCloud supplies you a template .prg called '**xxxSets.prg**', located in the folder where you store your application's .prg, and prefixed with your application code (step A2 above).

**MODIFY COMMAND xxxSets.prg** to see various implementations scenarios and what instructions should be found in this program.

Most important are **SET PATH/CLASSLIB/PROCEDURE ... ADDITIVE** to make sure FoxInCloud Application Server can retrieve your application files, visual classes and procedures at run time.

Once this .prg is created and populated with the necessary **SETs**, you need to instruct FoxInCloud Application Server of its location in order to run it at startup; **MODIFY COMMAND xxxServer.prg** to set the corresponding properties in your FoxInCloud Application Server:

- **cAppSetsLib**
- **cAppSets**
- **lAppSetsClass**

See **xxxSets.prg** and **xxxServer.prg** commented documentation for more details on how to setup these properties.

## STEP A4: CONFIGURE YOUR FIC SERVER

30 minutes

FoxInCloud installation has created 2 .prgs for your:

- **xxxServer.prg** in your app's **programs** sub-folder; **xxxServer.prg** is where you define your server classes: (i) **xxxServer** derived from **awServer**, (ii) **xxxProcess** derived from **awProcess**, etc.;
- **xxxTest.prg** in your app's **main** folder; **xxxTest.prg** is the main program for your Test project; it defines your Test server class **xxxTestServer AS xxxServer OLEPUBLIC<sup>2</sup>** and runs **xxxTestServer** in 'File mode' for development.

The reason for having 2 separate .prgs is that the same server can / should be deployed on several sites with the same code base and different data. Also it is very useful and even required to have a 'Test' and a 'Production' server sitting side by side on a live server and, why not, a 'Demo' server, etc.

While **xxxServer** settings are made through properties, **xxxTest** settings come from the **xxxTest.ini** file <sup>3</sup>. Open **<App folder>\xxxTest.ini** with a text editor <sup>4</sup> and adjust keys according to embedded documentation.

One important setting is the Folder in which wConnect and the FAS exchange files when running in 'file mode'. This setting is by default:

```
Tempfilepath=C:\Temp\xxx\
```

*We recommend you use the same directory structure; as mentioned in **xxxTest.ini**, any user should have full control on this folder; set NTFS security accordingly.*

---

<sup>2</sup> The **OLEPUBLIC** keyword is important to let you run your server in COM mode later on.

<sup>3</sup> You'll be able to modify **xxxTest.ini** on line with wConnect's Administration page.

<sup>4</sup> NotePad++ or PSpad is recommended for syntax coloring.

## STEP A5: RUN YOUR FIC APP SERVER

5 minutes

Make sure you launch VFP using the 'FoxInCloud Studio' **shortcut** created by FoxInCloud installation on your desktop: it ensures that **Set ( 'Path' )** points to the necessary FoxInCloud resource folders.

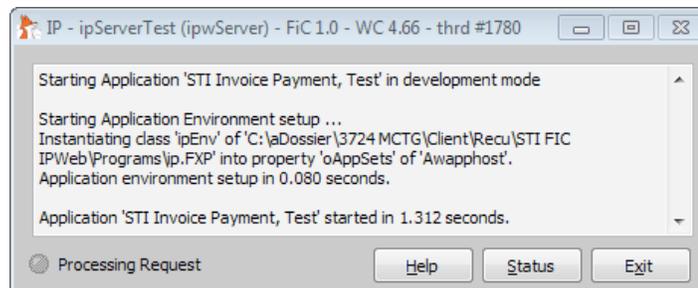
Then run your FoxInCloud Application Server by typing in the command Window:

**DO "<App folder>\xxxTest.prg"**

This should bring up a Web Connect window and describe in this window the operations carried out:

- Starting FoxInCloud Application Server
- Setting FoxInCloud Application Server environment according to what you did in Step 3 above,

When server startup completes, Server form's title bar reflects application code, server class, FoxInCloud and wConnect versions:



At this stage, your FoxInCloud Application Server is waiting for requests in the folder shared with wConnect ISAPI extension '**wc.dll**'. What you need to do next is setup IIS web server to have **wc.dll** capture request and write them into the above-mentioned shared folder. Please keep your Application Server running until all steps of this document are completed.

## STEP B1: SETUP '/AWSCRIPTS/' VIRTUAL

5 minutes

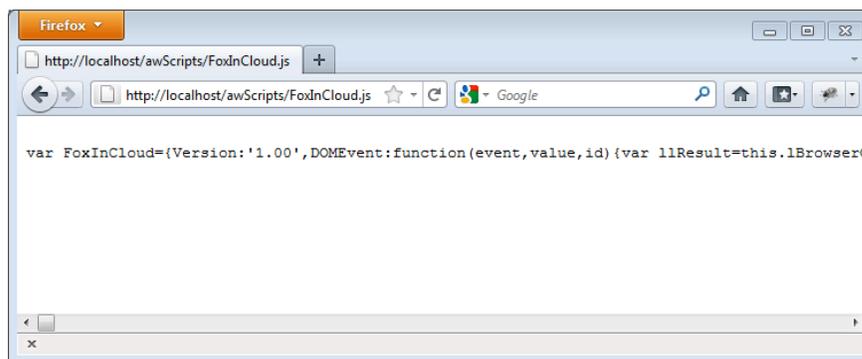
FoxInCloud comes with all the necessary JavaScript libraries located in `<VFP9>\Tools\AB\AW\Scripts\`. What you need to do now is create a virtual directory in `localhost` site <sup>5</sup> targeting this physical directory. To achieve that, proceed as follows <sup>6</sup>:

1. Open IIS management console (**RUN InetMgr.exe**)
2. Drill down to **Default Web site**
3. Right click on Default Web site, select New [>] Virtual Directory; the Virtual Directory wizard opens:
  - Alias: **awScripts**
  - Physical path: navigate to `<VFP9>\Tools\AB\AW\Scripts\`
  - Access: IIS<7: **read only**; IIS>=7: N/A

When done, type in **Firefox** address bar:

<http://localhost/awScripts/FoxInCloud.js>

You should see FoxInCloud minified JavaScript contents:



<sup>5</sup> Though IIS6 and higher allow creating several sites, we find it easier to work with `localhost` default site for development; production setup will of course support several sites.

<sup>6</sup> Note that this virtual directory is setup once for all: any of your FoxInCloud Applications will point to it.

## STEP B2: SETUP APP VIRTUAL

5 minutes

As in previous step, you now need to create a new virtual directory on **localhost** default Web Server pointing to your Application's '**Site\**' folder created in previous step A2.

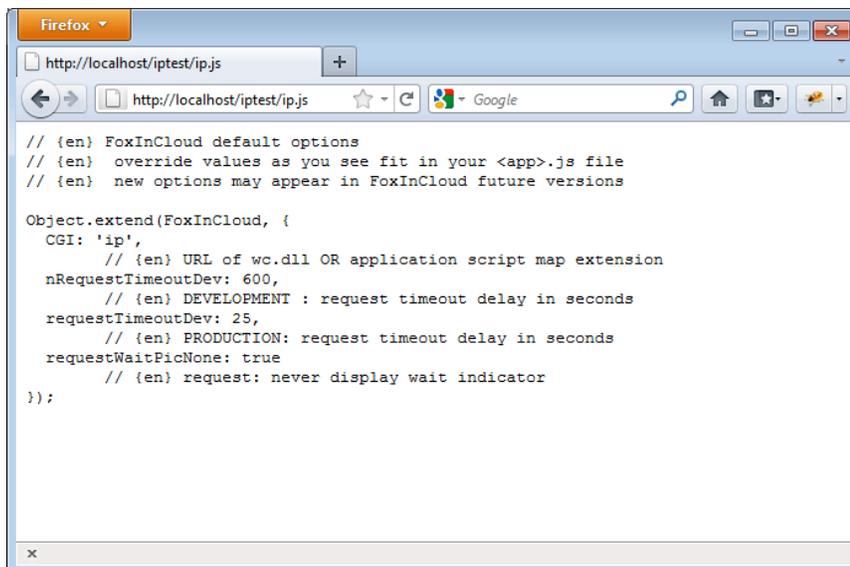
Repeat actions 1 to 3 in step B1, then setup your virtual directory:

- Alias: **xxx** (your application code)
- Physical path: navigate to **<App folder>\Site\**
- Access: **read**
- Execute: **scripts and executables**

When this is completed, type in FireFox address bar:

`http://localhost/xxx/xxx.js` && where 'xxx' is your application code

You should see the default application JavaScript file where FoxInCloud options are set for your Application (simplified in this example):



```
// {en} FoxInCloud default options
// {en} override values as you see fit in your <app>.js file
// {en} new options may appear in FoxInCloud future versions

Object.extend(FoxInCloud, {
  CGI: 'ip',
  // {en} URL of wc.dll OR application script map extension
  nRequestTimeoutDev: 600,
  // {en} DEVELOPMENT : request timeout delay in seconds
  requestTimeoutDev: 25,
  // {en} PRODUCTION: request timeout delay in seconds
  requestWaitPicNone: true
  // {en} request: never display wait indicator
});
```

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# STEP B3: CONFIGURE WC.INI

15 minutes

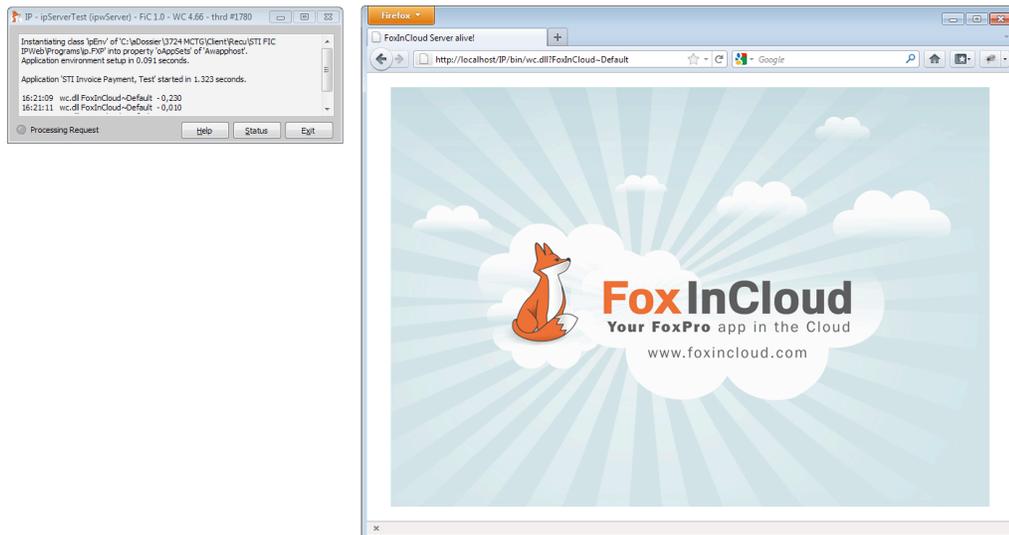
wConnect provides an ISAPI extension to IIS, `wc.dll`<sup>7</sup>, that you can configure with its `wc.ini` file. Both files sit in your `<App folder>\Site\bin\`.

In development mode, `wc.dll` and FoxInCloud/wConnect VFP server communicate in 'File mode'; in other words, both party poll the same folder for files written by the other party. To adjust this, open `<App folder>\Site\bin\wc.ini` with a text editor and complete settings as indicated by commented documentation in file.

One critical issue is having these 2 settings identical:

```
Tempfilepath=C:\Temp\xxx\    && in <App folder>\xxxTest.ini
Path=C:\Temp\xxx\           && in <App folder>\Site\bin\wc.ini
```

Typing `http://localhost/xxx/bin/wc.dll?FoxInCloud~Default:`



<sup>7</sup> The mechanism is: for each request against a wConnect site, IIS passes the request data through the 'ISAPI' interface to `wc.dll` which in turn passes the same request data to the wConnect VFP server it is linked to, then waits for response from the wConnect VFP server to pass back to IIS.

## STEP B4: SCRIPT-MAPPED EXTENSIONS

30 minutes

As you saw in previous step, typing URLs like

<http://localhost/xxx/bin/wc.dll?FoxInCloud~Default> is not really user-friendly and/or easy to remember.

What you need to do is declare, in IIS, an URL extension that both IIS and FoxInCloud will 'map' to [/bin/wc.dll?FoxInCloud](#)

With this 'Script-Mapped Extension', `'*.xxx'` for instance, typing

[/xxx/Default.xxx](#) will result in calling

[/xxx/bin/wc.dll?FoxInCloud~Default](#).

To do that, you need to go back to IIS admin, and select your `'xxx'` virtual directory. Then follow instructions indicated in wConnect documentation, under: '[Manually configuring IIS under Windows NT/2000/XP/Windows Server](http://www.west-wind.com/webconnection/docs/_sal0xeeat.htm)' ([http://www.west-wind.com/webconnection/docs/\\_sal0xeeat.htm](http://www.west-wind.com/webconnection/docs/_sal0xeeat.htm)) or '[Windows 7, Vista and Server 2008 Configuration](http://www.west-wind.com/webconnection/docs/_22f0xkbmq.htm)' ([http://www.west-wind.com/webconnection/docs/\\_22f0xkbmq.htm](http://www.west-wind.com/webconnection/docs/_22f0xkbmq.htm)) according to your Windows and IIS version.

Enter the following data:

- Extension: `'*.xxx'` (make sure you type `'.'`, `'*.'` for IIS>=7)
- Executable: `'<App folder>\Site\bin\wc.dll'`;  
With IIS <= 6, if ever your `<App folder>` path contains space(s), execute the following in VFP FoxInCloud Studio:

```
_cliptext = c83Address('<App folder>')
```

Then paste back to replace `<App folder>` containing space(s)

- For the rest, follow wConnect documentation indicated above.

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

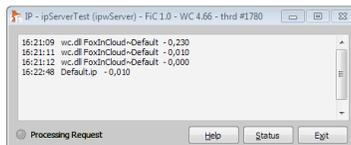
### Step B4: Script-Mapped Extensions...

FoxInCloud natively support the `'*.php'` extension; you may want to create another script mapping for this extension if ever you wish.

You may also create any other extension provided you add it to:

```
xxxServer.prg!xxxServer::cScriptMapExts
&& defaults to 'php,pl,asp,vfp,zip'; extend with your own ...
cScriptMapExts = 'php,pl,asp,vfp,zip,any,other,ext,for,you'
```

Once your Script-Mapped Extensions are created, `'*.xxx'` for example, typing <http://localhost/xxx/Default.xxx> results in the same output as <http://localhost/xxx/bin/wc.dll?FoxInCloud~Default:>



Next steps will focus on putting all pieces together to run your App form(s) on the Web with FoxInCloud.

[www.foxincloud.com](http://www.foxincloud.com) | [contact@foxincloud.com](mailto:contact@foxincloud.com)

## STEP C1: BUILD YOUR PROJECT

10 minutes

FoxInCloud Application Server installation has installed the following program:

`<VFP9>\Tools\AB\AW\App\xxxTest.pjx.prg.`

Just copy this program into your application's root folder, rename it by replacing '`xxx`' by your application code chosen in step 2 above, and run it:

```
DO "<App folder>\xxxTest.pjx.prg"
```

This will result in creating or refreshing your application project:

`'<app root folder>\xxxTest.pjx'`.

*Note: because references build up progressively, just '**ignore all**' on build errors.*

**Trial users:** as many references lie in `aw.app`, building your project will result in various build errors; this is normal and does not jeopardize your FAS operations. *Build errors will disappear when you acquire your FoxInCloud license and get full source code.*

**Full licenced users:** It is very important that you keep the same .pjx from the moment you run your FoxInCloud Application Server in production as COM server; otherwise you'll have to re-register your server, as its GUID will have changed <sup>8</sup>.

You can safely to rerun your copy of `xxxTest.pjx.prg` several times.

---

<sup>8</sup> Though there might be some way to retrieve the GUID from a former project and force it into a new project, it is recommend to always keep the same project files; if you have several developers who can update the same COM production server object, think of copying the project on each development station.

## STEP C2: PICK A FORM LAUNCH STRATEGY

30 minutes

FoxInCloud Application Server acting as a wrapper around your application, it needs to 'launch' (instantiate) your forms to:

- Generate HTML that reflects your forms' object hierarchy,
- Implement your forms events into HTML DOM events,
- Route HTML events received from client to your forms events methods.

Form instantiation & processing may occur either (i) when server starts, or (ii) on the fly, just when needed to display your Form's HTML within a page.

As instantiating & processing a form may take several seconds, in order to speed-up Server startup, it might be a good idea to delay this task until your form is actually needed; this is particularly useful in development, when Server needs be restarted often, or even in production, when updating the server, to reduce the delay while users wait for server availability.

Actually, instantiating forms at server startup is required **only** for top-level forms, **and** if you choose to generate HTML pages containing these forms by a non-wConnect mechanism such as Server Side include directives in ASP, PHP, or any other web language. Furthermore, you can mix Startup and Just-In-time launch form by form.

### Launching Form(s) at Server startup

If you want to produce some form HTML page(s) with a non-wConnect tool such as ASP or PHP, you need to:

1. Declare a coma-delimited list of forms in your server's property named `'uFormsLaunchAtStartup'`, or .T. for all forms:

```
uFormsLaunchAtStartup = 'form1.scx, form2.scx, formClass1, formClass2'
```

## Step C2: Pick a form launch strategy...

2. Add some include directives to your target page(s) as explained in the [sample form.asp](#) page supplied with FoxInCloud installation:

```
<VFP9>\Tools\AB\AW\App\Site\Form.asp
```

### Launching Form(s) Just In Time

A preferable mechanism is to let FoxInCloud Application Server generate HTML forms for you, with standard `xxxServer.prg!xxxProcess::wForm*()` methods:

- `wFormStandardPage(form, parm1, parm2, etc.)` outputs a full HTML page including all the necessary JavaScript and CSS <sup>9</sup>;
- `wFormHTML(form, parm1, parm2, etc.)` returns form's HTML you can manually include in a page using standard wConnect's `MergeText()` and/or `ExpandTemplate()` methods, or any other VFP text generation mechanism such as `TEXT ... ENDTEXT`

What you need to do is:

- Implement one method per top-level form in:  
`xxxServer.prg!xxxProcess` (see sample code in this .prg)
- If you see fit, override `xxxProcess::wFormHTML_cHeader()` and/or `xxxProcess::wFormHTML_cFooter()` to customize form's header and footer, like adding menu and/or `<a href="yourForm.xxx">to your various forms</a>`.

On next page is presented a sample form generated by `xxxProcess::wFormStandardPage()`

---

<sup>9</sup> Full-licensed users may see `xxxServer.prg!xxxProcess` source code for more details.

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# Step C2: Pick a form launch strategy...

The screenshot shows a web browser window displaying the 'STI Invoice Payment System' interface. The page header includes 'THE SHERIDAN GROUP' and the 'STI TRACKING OUTSIDE THE BOX.' logo. A search bar and navigation buttons are visible. The main content is a table of invoices with the following columns: Division, Carrier, Carrier Account Number, Payer, Invoice Number, Invoice Date, Invoice Due Date, Invoice Amount, Details, G/L Coding, Edit Codes, Invoice Status, Status Date, Dispute, Approve, and Pay. The table lists multiple invoices from 'Sheridan Press' and 'FedEx' with various invoice numbers and amounts. At the bottom of the page, there are links for 'Export Data', 'View Invoices to Pay', and 'Pay Now'. The footer includes the 'FoxInCloud' logo and a status bar indicating 'Recherche terminée en 0.73 s (Internet 0.01 s, application 0.27 s, server 0.40 s, browser 0.05 s)'.

Figure 1 - Sample form generated by `xxxProcess::wFormStandardPage()`

## STEP C3: GENERATE FORMS HTML/CSS/JS

5-30 minutes

By design, FoxInCloud Application Server generates in your `<app folder>\Site\` folder <sup>10</sup>:

- One set of HTML/JS files for each form, named as follows:
  - . `"form.scx"`: `"form_scx.htm"` and `"form_scx.js"`;
  - . `"form"` class: `"form.htm"` and `"form.js"`;These files are created **each time** a form is instantiated.
- 2 `.css` files reflecting your application's classes and objects:
  - `awDefault<sys (2015)>.css` updated each time FoxInCloud Application Server discovers a new class during the course of user requests;
  - `awDefaultAll.css` generated when running FoxInCloud Application Server against all forms and classes in project or sub-folders with `.T.` as first parameter, or by running:

```
DO "<App folder>\xxxTest.css.prg"
```

**Full-licensed user:** make sure you project is open before issuing this command: FoxInCloud will search your project for all forms contained to make sure every class is properly converted into CSS.

**Instantiation errors:** some error may occur if your forms are not completely adapted, just ignore them and continue until all forms have gone through; you will get a complete warning log when FoxInCloud Application Server unloads.

If you chose to generate your form pages with `xxxProcess : wForm* ()` methods, you don't need to generate `awDefaultAll.css` until you go to production: in development mode, FAS will update CSS automatically; if you chose to generate your form pages with another mechanism such as ASP or PHP, then you need an updated `awDefaultAll.css`.

---

<sup>10</sup> Content is commented if you run your server in trial or development mode, minified otherwise.

## STEP C4: SETUP YOUR DEFAULT PAGE

10 minutes

When user closes a top-level form in the browser, FoxInCloud needs to know which page display next.

By default, FoxInCloud redirects user to the main virtual directory, so it is important you define a default page to display.

FoxInCloud comes with an empty `'index.htm'` file located in `Site\`. `'index.htm'` is one of the files IIS considers as a potential default.

What you need to do is:

- In IIS management console, go to your virtual directory, then click on 'Default document'; you'll see `'index.htm'` in the default documents.

Then you have two options:

- Either implement some HTML content as you see fit in `'index.htm'`
- Or redirect `'index.htm'` to some other page either produced by FoxInCloud Application Server (such as `'index.xxx'`), or to some other page where you have, for instance, your application's menu.

To redirect `'index.htm'`, right-click on this file, choose 'display functionalities'; then click on `'HTTP redirections'`, then type in the URL you want to redirect to.

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# STEP C5: RUN YOUR FOXINCLOUD APP

Congratulations, you're done!

You can just type <http://localhost/xxx/someForm.xxx> to see one of your application's forms pop into the browser and begin interaction.

Here are some forms of applications running with FoxInCloud...



Figure 2 - Home page

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# Step C5: Run you FoxInCloud App...

The screenshot shows a web browser window with the URL <http://foxincloud.com/BSW/ordonnance.BSW>. The page title is "FICHE PRESCRIPTION-ORDONNANCE - BioSilWeb". The form is titled "FICHE PRESCRIPTION-ORDONNANCE - BioSilWeb" and contains the following information:

- Navigation:** Déconnecter, Agenda, Prescription, Technique, Validation, Consultation, Gestion, Statistiques, Paramétrage, A propos
- Menu:** BIOSILWEB -- SUPER
- Form Fields:**
  - Number: 2652
  - CREE: 04/08/2011
  - PRESCRIT: 04/08/2011
  - MODIFIE: 04/08/2011
  - OPERATEUR: 10
  - PATIENT: 2040 DENNEMENT
  - PRESCRIPTEUR: 202 RAKOTO
  - SEX: M, AGE: 65 An(s)
  - PRELEVEMENT, DOMICILE, DISTANCE
- Buttons:** ARCHIVE, ATTENTE, SEJOUR, IMPRIME, FACTURE, TRANSMIS, P.E.C., MULTIPLE
- Financial Summary:**
  - Total laboratoire: 25000.00
  - Total analyses: 25000.00
  - Total prélèvement: 25000.00
  - Total déplacement: 25000.00
  - Clé B: 250.00, Clé HN: 250.00, Solde patient: -30000.00
  - % état et montant règlement patient: 100 ERREUR, 25000.00
  - % état et montant règlement Tiers: 100 ERREUR, 25000.00
- Table:**

SUP.	CODE	CLE	SPEC.	%PAT
<input type="checkbox"/>	PSA	100	<input type="checkbox"/>	100
<input type="checkbox"/>	RQTP	0	<input type="checkbox"/>	100
- Footer:** GLS INFORMATIQUE :: FOXINCLOUD
- Search Log:** Recherche terminée en 0.76 s (Internet 0.04 s, application 0.13 s, serveur 0.53 s, browser 0.07 s)

Figure 3 - Order form

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# Step C5: Run you FoxInCloud App...

The screenshot displays the BioSilWeb application interface within a web browser. The browser's address bar shows the URL <http://foxincloud.com/BSW/validation.BSW>. The application title is "VALIDATION DOSSIER - BioSilWeb". The interface includes a navigation menu with options like "Déconnecter", "Agenda", "Prescription", "Technique", "Validation", "Consultation", "Gestion", "Statistiques", "Paramétrage", and "A propos". The main content area is titled "VALIDATION DOSSIER - BioSilWeb" and shows a list of dossiers on the left, with "2637 ANDRIANAHO Rantilo" selected. The central part of the screen is a table for biological analysis validation, with columns for "T", "V", "EXAMEN", "TEST", "VALEUR", and "ANTERIEUR(S)". The table contains several rows of data, including tests like "TPHAR", "VDRLR", "CONC", "RQ", "TUBE", "HIV", "HIVB", "HIVC", "ORIG", "ABON", "COUL", "KOH", "LEU", "HEM", "CEL", "TRI", "LEV", "FIL", "CLUE", "DODERL", "GRAM", and "COCOC". The "VALEUR" column shows results such as "Négatif", "En cours", and "COMMENTAIRE". The "ANTERIEUR(S)" column shows a date and result: "2011/03/23 Négatif". At the bottom of the interface, there are logos for "GLS INFORMATIQUE" and "FOXINCLOUD", and a status bar indicating "Recherche terminée en 1.54 s (Internet 0.49 s, application 0.11 s, serveur 0.62 s, navigateur 0.32 s)".

T	V	EXAMEN	TEST	VALEUR	ANTERIEUR(S)
<input checked="" type="checkbox"/>	BW	TPHAR	Négatif		
<input checked="" type="checkbox"/>	BW	VDRLR	Négatif		
<input checked="" type="checkbox"/>	BW	CONC			
<input checked="" type="checkbox"/>	BW	RQ			
<input checked="" type="checkbox"/>	BW	TUBE	2637		
<input checked="" type="checkbox"/>	HIVQ	HIV	Négatif		
<input checked="" type="checkbox"/>	HIVQ	HIVB	Négatif		
<input checked="" type="checkbox"/>	HIVQ	TUBE	2637		
<input checked="" type="checkbox"/>	HIVQ	CONC	COMMENTAIRE		
<input checked="" type="checkbox"/>	HIVQ	RQ			
<input type="checkbox"/>	PV	ORIG	En cours		
<input type="checkbox"/>	PV	ABON	En cours		
<input type="checkbox"/>	PV	COUL	En cours		
<input type="checkbox"/>	PV	KOH	En cours		
<input type="checkbox"/>	PV	LEU	En cours		
<input type="checkbox"/>	PV	HEM	En cours		
<input type="checkbox"/>	PV	CEL	En cours		
<input type="checkbox"/>	PV	TRI	En cours		
<input type="checkbox"/>	PV	LEV	En cours		
<input type="checkbox"/>	PV	FIL	En cours		
<input type="checkbox"/>	PV	CLUE	En cours		
<input type="checkbox"/>	PV	DODERL	En cours		
<input type="checkbox"/>	PV	GRAM	en cours		
<input type="checkbox"/>	PV	COCOC	En cours		

Figure 4 - Biological analysis validation form

MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# Step C5: Run you FoxInCloud App...

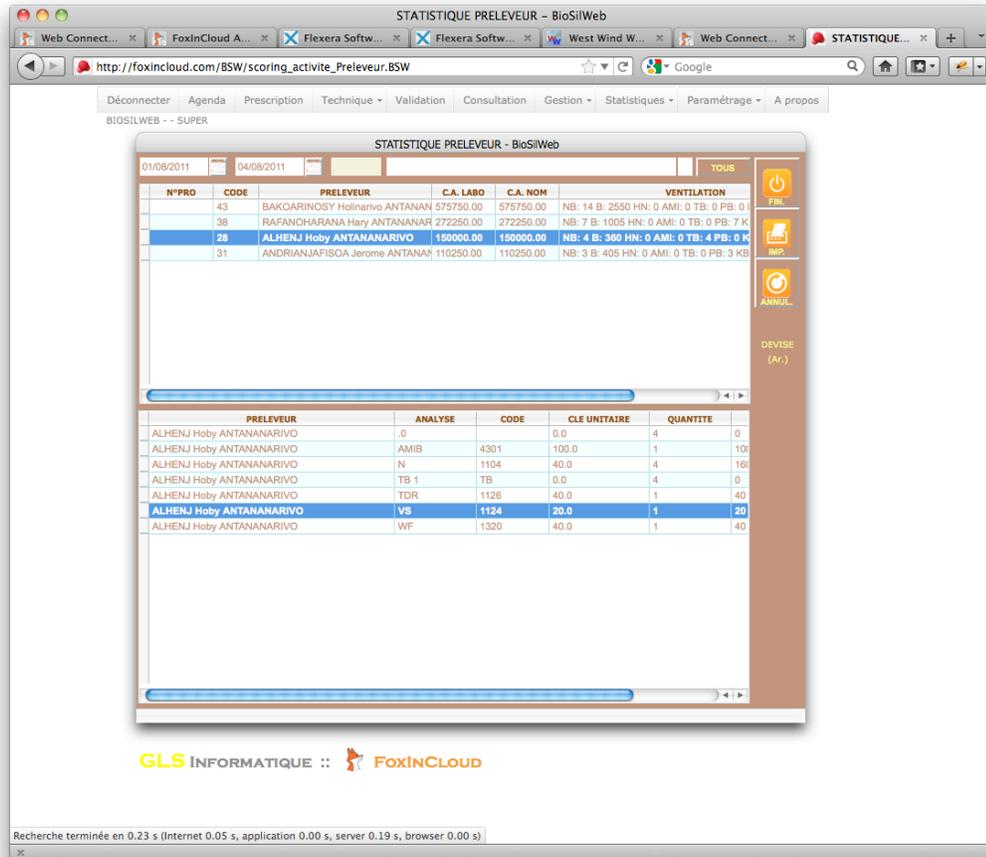


Figure 5 - client activity stats form

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

### Step C5: Run you FoxInCloud App...

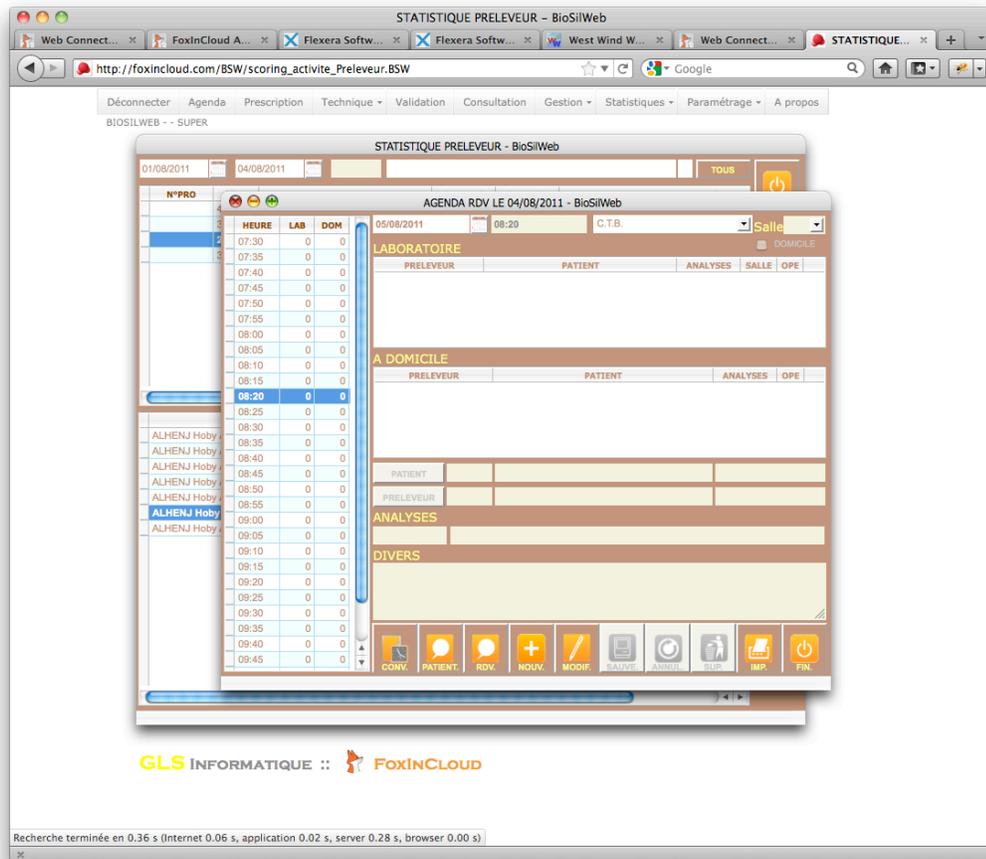


Figure 6 - Agenda form shown as modeless child form

## MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

# Step C5: Run you FoxInCloud App...

VALIDATION CONNEXION - BioSiWeb

NUMERATION: 0

PAVILLON SAINTE FLEUR

T0 (N) Proméiocytes %

IDENTIFICATION	TEST	EMIS	RESULTAT	PASSAGE 1	PASSAGE 2	PASSAGE 3	TUBE	ANTERIEUR(S)
2338 - RAKOTONDRA SOA Voahangy F 39An(s)	GR	<input type="checkbox"/>					0600709338	
	HGB	<input type="checkbox"/>					0600709338	
	HT	<input type="checkbox"/>					0600709338	
	VWC	<input type="checkbox"/>					0600709338	
	TCMH	<input type="checkbox"/>					0600709338	
	CCMH	<input type="checkbox"/>					0600709338	
	GB	<input type="checkbox"/>					0600709338	
	PN	<input type="checkbox"/>					0600709338	
	PE	<input type="checkbox"/>					0600709338	
	PB	<input type="checkbox"/>					0600709338	
	LYMP	<input type="checkbox"/>					0600709338	
	MICRO	<input type="checkbox"/>					0600709338	
	PLASM	<input type="checkbox"/>					0600709338	
	META	<input type="checkbox"/>						
	MYEL	<input type="checkbox"/>						
	PROM	<input type="checkbox"/>						
	NUCL	<input type="checkbox"/>						
	MYELB1	<input type="checkbox"/>						
	PMYEC	<input type="checkbox"/>						
	MYEC	<input type="checkbox"/>						
	MITMYC	<input type="checkbox"/>						
	BLAST1	<input type="checkbox"/>						
	BLAST2	<input type="checkbox"/>						
	PLAQ	<input type="checkbox"/>					0600709338	
	MOR	<input type="checkbox"/>						
	MOR2	<input type="checkbox"/>						
	MOR3	<input type="checkbox"/>						
	RQ	<input type="checkbox"/>						
2552 - RATSIMBAZAFY Ando M 25An(s)	PLASM	<input type="checkbox"/>						

GLS INFORMATIQUE :: FOXIN CLOUD

Recherche terminée en 0.68 s (Internet 0.08 s, application 0.03 s, serveur 0.48 s, browser 0.08 s)

Figure 7 - Connection with biological analysis instruments

# MANUALLY INSTALLING A DEVELOPMENT APPLICATION SERVER

## Step C5: Run you FoxInCloud App...

The screenshot shows a web browser window displaying the 'FACTURATION TIERS - BioSiWeb' application. The interface includes a navigation menu at the top, a search bar, and a main table of invoices. The table has columns for 'LOT', 'TIERS', 'DOSSIER', 'DATE', 'PATIENT', 'MONTANT', 'HONORAIRES', 'ACCOMPTÉ', and 'NET A PAYER'. The 'MONTANT' column is highlighted in orange, and the 'ACCOMPTÉ' column is highlighted in green. The 'NET A PAYER' column is highlighted in blue. The table contains 30 rows of invoice data.

LOT	TIERS	DOSSIER	DATE	PATIENT	MONTANT	HONORAIRES	ACCOMPTÉ	NET A PAYER
0	AMADIA	2325	08/07/11	RAZAFINDRAKOTO Xavier Michel né(e) le 18/05/1964 rang 1	108750.00	108750.00	0.00	108750.00
0	CENTRE MEDICAL ANOSINAWAKA	2334	08/07/11	ANDRIANO Nadine né(e) le 30/11/1983 rang 1	30000.00	-3000.00	0.00	-3000.00
0	CENTRE MEDICAL ANOSINAWAKA	2367	12/07/11	ANDRIANO Nadine né(e) le 30/11/1983 rang 1	56250.00	-3000.00	0.00	-3000.00
0	CENTRE MEDICAL ANOSINAWAKA	2434	19/07/11	ANDRIANAIVOARUAONA Fabien né(e) le 07/10/1991 rang 1	15000.00	-3000.00	0.00	-3000.00
0	CLINIQUE MEDICO-CHIRURGICALE ADV	2317	07/07/11	RAVAIVOMANOELINA Michee né(e) le 10/03/1990 rang 1	10000.00	8000.00	0.00	8000.00
0	CLINIQUE MEDICO-CHIRURGICALE ADV	2349	11/07/11	KARMALY Fidahoussen né(e) le 01/01/1943 rang 1	38750.00	36000.00	0.00	36000.00
0	DHL	2422	18/07/11	ANDRIANIMANANA Marry né(e) le 07/01/1977 rang 1	91750.00	18350.00	0.00	18350.00
0	DR MAMELOMANA MYRIAME	2245	01/07/11	RAHANTAVOLOLONA Justine né(e) le 12/02/1945 rang 1	75750.00	-3000.00	0.00	-3000.00
0	DR MAMELOMANA MYRIAME	2247	01/07/11	RAMANAVELONANDRIANO né(e) le 06/10/1955 rang 1	53750.00	-3000.00	0.00	-3000.00
0	DR MAMELOMANA MYRIAME	2271	04/07/11	RAHARISON Malakotiana né(e) le 27/11/1994 rang 1	20000.00	-3000.00	0.00	-3000.00
0	DR MAMELOMANA MYRIAME	2279	05/07/11	BODONAVOLONA Zoely né(e) le 22/08/1985 rang 1	53750.00	-3000.00	0.00	-3000.00
0	DR MAMELOMANA SIMON	2284	04/07/11	MAHAFALY Brillante né(e) le 10/11/1973 rang 1	13000.00	-3000.00	0.00	-3000.00
0	DR RAMANDIMBISOA TOMBO	2347	11/07/11	BERANTO Alexandre né(e) le 27/02/1952 rang 1	38750.00	-18000.00	0.00	-18000.00
0	GROUPE TELMA	2421	18/07/11	ANDRIANIRINA Florian Herve né(e) le 30/09/1952 rang 1	26750.00	-3000.00	0.00	-3000.00
0	GROUPE TELMA	2559	28/07/11	RAZAFIARISOLO Fano Manatena né(e) le 22/10/1981 rang 1	43000.00	-3000.00	0.00	-3000.00
0	PAVILLON SAINTE FLEUR	2255	02/07/11	RAZANADRAMANANA Nirina Synthia né(e) le 01/01/1988 rang 1	20000.00	18000.00	0.00	18000.00
0	PAVILLON SAINTE FLEUR	2257	02/07/11	ARFADJA né(e) le 01/01/1991 rang 1	20000.00	18000.00	0.00	18000.00
0	PAVILLON SAINTE FLEUR	2258	02/07/11	RAZAFINDRABETSIAVALONA Christianah né(e) le 01/01/1990 rang 1	20000.00	18000.00	0.00	18000.00
0	PAVILLON SAINTE FLEUR	2285	04/07/11	RAMANOARISOA Sashondra Angela né(e) le 13/11/1982 rang 1	47500.00	42750.00	0.00	42750.00
0	PAVILLON SAINTE FLEUR	2287	04/07/11	RATVOARISOLO Helino Sandra né(e) le 24/10/1978 rang 1	10000.00	9000.00	0.00	9000.00
0	PAVILLON SAINTE FLEUR	2289	04/07/11	RATISSONNE Nadia Sandrine né(e) le 13/03/1982 rang 1	10000.00	9000.00	0.00	9000.00
0	PAVILLON SAINTE FLEUR	2283	05/07/11	RAKOTO ARIMANANA Juane Tiavina W né(e) le 01/07/2001 rang 1	20000.00	18000.00	0.00	18000.00
0	PAVILLON SAINTE FLEUR	2284	05/07/11	RAHARIJAONA Angela né(e) le 01/01/1983 rang 1	20000.00	18000.00	0.00	18000.00
0	PAVILLON SAINTE FLEUR	2285	05/07/11	RAKOTOMALALA Naholisoa né(e) le 24/08/1989 rang 1	10000.00	9000.00	0.00	9000.00
0	PAVILLON SAINTE FLEUR	2286	05/07/11	ANDRIANAVALOTSEHENO Haritiana né(e) le 10/12/1978 rang 1	17500.00	15750.00	0.00	15750.00
0	PAVILLON SAINTE FLEUR	2285	06/07/11	SIDONIE Fara né(e) le 01/01/1971 rang 1	10000.00	9000.00	0.00	9000.00
0	PAVILLON SAINTE FLEUR	2286	06/07/11	RAMANOARISOA Sashondra Angela né(e) le 13/11/1982 rang 1	10000.00	9000.00	0.00	9000.00
0	PAVILLON SAINTE FLEUR	2287	06/07/11	RAZANANANDRESY Claudis né(e) le 01/01/1991 rang 1	10000.00	9000.00	0.00	9000.00
0	PAVILLON SAINTE FLEUR	2288	06/07/11	NIVOMALALA Marie Adeline né(e) le 17/01/1985 rang 1	10000.00	9000.00	0.00	9000.00
0	PAVILLON SAINTE FLEUR	2289	06/07/11	RAKOTOARISOA Veronique né(e) le 09/06/1975 rang 1	10000.00	9000.00	0.00	9000.00
0	PAVILLON SAINTE FLEUR	2300	06/07/11	RAKOTOARISENDRA Felana né(e) le 18/02/1985 rang 1	10000.00	9000.00	0.00	9000.00

Recherche terminée en 1.43 s (Internet 0.13 s, application 0.28 s, serveur 0.61 s, navigateur 0.41 s)

Figure 8 - Invoicing Form

www.foxincloud.com | contact@foxincloud.com



# Fox InCloud

[www.foxincloud.com](http://www.foxincloud.com) | [contact@foxincloud.com](mailto:contact@foxincloud.com)

---

Abaque | SARL au capital de 41 200 € | RCS Paris B 379 373 566 | 66, rue Michel Ange  
75016 Paris +33 (0)9 53 41 90 90 | [www.zenbuyer.com](http://www.zenbuyer.com) | [info@zenbuyer.com](mailto:info@zenbuyer.com)