

# <faxapi>

The faxserver.net Fax API offers support for .NET, JAVA, PHP and RestAPI.

To see the full API package, please contact [sales@ingeniumsw.com](mailto:sales@ingeniumsw.com)

## .NET Fax Sample:

Please contact us for our .NET API code.

## Java Fax Sample:

```
/**
 * FaxApp.java
 * Copyright (c) 2009-2013, etherFAX, LLC
 *
 * Sample etherFAX client application.
 */

import java.util.ArrayList;

import etherfax.web.AccountStatus;
import etherfax.web.EtherFaxClient;
import etherfax.web.FaxReceive;
import etherfax.web.FaxResult;
import etherfax.web.FaxStatus;

public class FaxApp {

    /**
     * main
     */
    public static void main(String[] args) {
        // new etherfax client
        EtherFaxClient client = new EtherFaxClient();

        // set credentials and create client session
        client.setCredentials("my_account_#", "my_user_name", "my_password");
        client.create();

        // get account information
        AccountStatus account = client.getAccountStatus();
        System.out.println("name: " + account.Name + ", ports available: " + account.Ports);

        // get unread fax count
        int unread = client.getUnreadFaxCount();
    }
}
```

```

if (unread > 0) {
    // show unread fax count
    System.out.println("unread faxes: " + unread);

    // list all unread faxes
    ArrayList<FaxReceive> list = client.getUnreadFaxList();
    for (int n = 0; n < list.size(); n++) {
        // show job id
        System.out.println("job-id: " + list.get(n).JobId + ", received on: " +
list.get(n).ReceivedOn);

        // download the fax
        FaxReceive receive = client.getFax(list.get(n).JobId);
        if (receive != null) {
            // receive.FaxImage contains fax contents
        }

        // mark fax as read (only call if you have saved received fax successfully)
        client.setFaxReceived(list.get(n).JobId);
    }
}

// show pending outbound faxes
int pending = client.getPendingFaxes();
System.out.println("pending faxes: " + pending);

// send a fax
int pages_in_file = 1;
FaxStatus status = client.sendFax("my_phone_num",
    "my_document.tif", pages_in_file, "my_csid", "my_caller_id", "my_tag");

// was it accepted?
if (status.Result == FaxResult.InProgress)
    System.out.println("id: " + status.JobId);

// save job id
String id = status.JobId;

while (true) {
    // display result/status
    System.out.println("result: " + status.Result);

    // fax completed?
    if (status.Result != FaxResult.InProgress)
        break;

    try {
        // delay

```

```

        Thread.sleep(15 * 1000);
    }
    catch (Exception e) {
    }

    // update status
    status = client.getFaxStatus(id);
}

// close session
client.close();
}
}

```

## PHP Fax Sample:

```

setCredentials('efax-xxxx-xxxx/user_name', 'password'); // get account status $acc = $client-
>getAccountStatus(); echo"Account: ". $acc->Account .", ports: ". + $acc->Ports ."\n"; // get unread fax
count $unread = $client->getUnreadFaxCount(); if ($unread > 0) { // get unread fax list echo'Unread
Faxes: '. $unread ."\n"; $faxes = $client->getUnreadFaxList(); foreach ($faxes as $f) { // display fax
information echo'New fax from: '. $f->CallingNumber .' on '. $f->ReceivedOn ."\n"; // download fax $fax
= $client->getFax($f->JobId); echo'Fax image length: '. strlen($fax->FaxImage) ."\n"; // mark fax as
received $client->setFaxReceived($f->JobId); } } // send fax $fax = $client->sendFax('+18005551234',
'../../sample.tif', 0, 'MyCSID'); echo'Send Result: '. $fax->Result ."\n"; if ($fax->Result ==
FaxResult::InProgress) { // display job id if successful

```

## RestAPI Fax Sample:

```

using System;
using System.Collections.Generic;
using System.IO;
using System.Text;
using System.Threading;
using EtherFax.Web;

namespace EtherFax.Web.Test
{
    class Program
    {
        static void Main(string[] args)
        {
            // create etherfax instance
            EtherFaxConnection ef = new EtherFaxConnection("efax-9999-9999/jsmith",
"mypassword");

```

```

// get account status
string response = ef.GetAccountStatus();
Console.WriteLine(response);

// get number of pending faxes
Console.WriteLine("pending faxes: {0}", ef.GetPendingFaxes());

// get number of unread faxes
Console.WriteLine("unread faxes: {0}", ef.GetUnreadFaxCount());

// list unread faxes
List<FaxReceive> list = ef.GetUnreadFaxList();
foreach (FaxReceive r in list)
{
    // show fax
    Console.WriteLine("received from: {0}, id: {1}", r.RemotId, r.JobId);

    // get complete fax
    FaxReceive recv = ef.GetFax(r.JobId);
    if (recv != null)
    {
        // save to .tif file
        string file = "c:\\temp\\" + r.JobId.ToString() + ".tif";
        using (FileStream fs = new FileStream(file, FileMode.Create))
        {
            fs.Write(recv.FaxImage, 0, recv.FaxImage.Length);
            fs.Close();
        }
    }

    // mark as received
    ef.MarkFaxReceived(r.JobId);
}

// send a fax
FaxStatus status = ef.SendFax("8005551234", "c:\\temp\\sample.tif", "My
CSID", null, "123abc");
Console.WriteLine("SendFax result: {0}, id: {1}", status.Result, status.JobId);

// let's monitor, if in progress
while (status.Result == FaxResult.InProgress)
{
    // delay
    Thread.Sleep(15000);

    // get status
    status = ef.GetFaxStatus(status.JobId);
    if (status == null)

```

```
        {
            Console.WriteLine("failed to get status");
            break;
        }

        // display state
        Console.WriteLine("state: {0}, connect: {1}, speed: {2}, remote: {3},
delivered: {4}",
                        status.State, status.ConnectTime, status.ConnectSpeed,
status.Remoteld, status.PagesDelivered);
    }

    Console.WriteLine("done");
}
}
```