

# Visma.net IAM API

Version 1.4  
Revision date: 1/Apr/2023

# Introduction

Visma.Net.HRM has a suite of APIs available for integration with third party systems. This document focuses on the “IAM API” also known as User Management APIs.

## Getting Started with an integration

If you are interested in building an integration please contact [projectenbureau@visma.com](mailto:projectenbureau@visma.com) or your Customer Success Manager. They can provide assistance and a test environment.

Before you begin building the integration you will need the following details.

## Customer URL

Each customer has a unique URL to access the APIs. The unique URL will be communicated by the Visma Consultant when requesting the activation of the interface. With this URL the API's can be called after the Service user accounts are created and configured

## Service User Account

For this functionality the customer must set up 1 dedicated service user account in Visma.Net HRM. The service user account will have an email address and password. This will be required to authenticate each API call. See underneath example:

- 1 service user (API authentication) including a mailbox
  - Example: [VismaApi@comapnynome.nl](mailto:VismaApi@comapnynome.nl)

This user needs to be created by the IT admin of the customers company. The application admin or superuser will need to activate the service user in the Admin module of Visma.net.

After activation the names of the service user needs to be sent to the Visma consultant before starting the implementation ([Projectenbureau@visma.com](mailto:Projectenbureau@visma.com))

## Talent Manages Users

After receiving the user credentials the Visma consultant will be ready to start the configuration of the interface in HRM and add the proper rights. After configuring, the system will be ready for the scan implementation.

## Demo Program for developing the interface

A Visual.Basic.Net sample program including source is available that demonstrates the api calls. This can be used for testing and seeing how the calls are set up and what the results of the calls will be. The program will be communicated at the beginning of the development of the interface.

## SOAP wsdI

Some of the APIs are SOAP calls and require a wsdI file. This is available in the demo program. It is called EmplInfo.wsdI

## Json Calls

For all Json calls you need to have Content type = Text/Json

## User Management APIs

The following are the list of APIs suitable for building user management integration.

- Login.aspx
- Aut.UserCreate
- Aut.UserUpdate
- Aut.UserSearch
- Aut.GetUserInfo

The user management APIs should only be used when Visma.net HRM is managing users.

Basic Flow for new employee

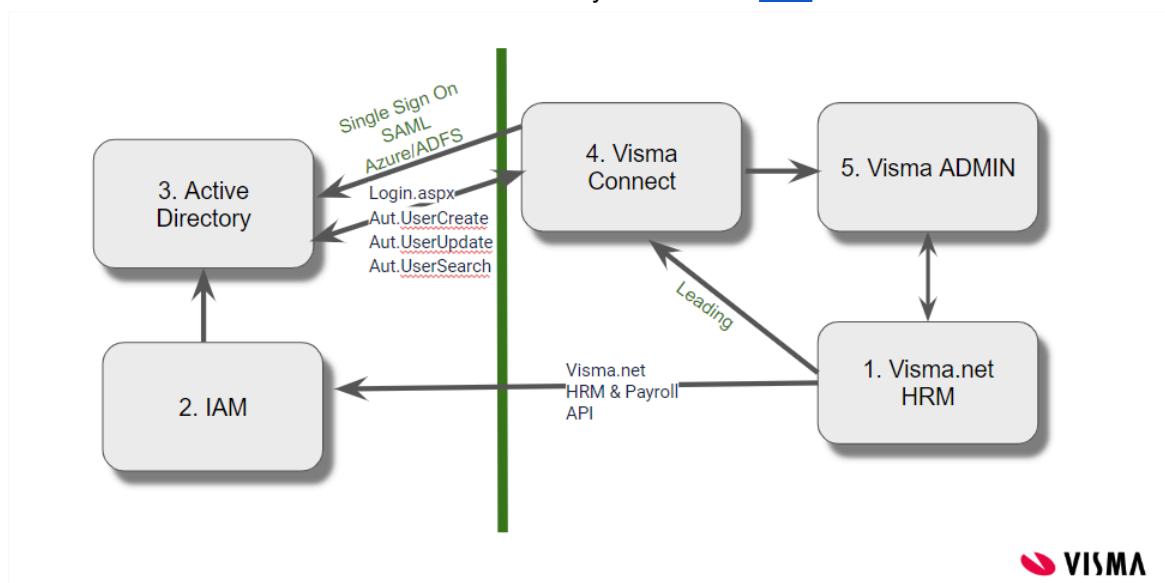
Step 1. Visma.net.HRM sends employee information using [Visma.net HRM & Payroll API](#).

Step 2. IMS identify new employee and assign business email address

Step 3. IMS calls Aut.UserCreate to create a user in Visma.net.HRM and Visma.Admin with the new email address. If you include the employeeID in the message it will automatically assign the user to the employee and assign the email address in the employee's email as the sign-in email address.

For more information on Single Sign On click [here](#).

For more information on the Visma.net HRM & Payroll API click [here](#).



# Authentication

All calls will need a valid cookie. To get a Valid Cookie (gsID) you need to call Login.aspx.

	Login.aspx
URL	<p>https://&lt;customername&gt;Talent.net/WebFramework/Login.aspx</p> <p>You should always use the Customer URL rather than the internal name. For example, ABC BV will have an external url called <a href="https://ABCTalent.visma.net">ABCTalent.visma.net</a></p>
Method	POST
Description	This will return a valid cookie. (gsID)
FunctionID	N/A
Parameters	<p>eMailAddress: Email address of the service user</p> <p>password: Password of the service user</p>
Sample Call	<pre>{   "eMailAddress" : "x@y.z" ,   "password" : "something" }</pre>
Sample Results	<pre>{   "gsId": "x-y-z-r-t" }</pre>
Failed login	If the login is not successful a 4xx or 5xx error will be returned and the connection terminated.

## Available APIs

### Aut.UserCreate

<b>Aut.UserCreate</b>	
<p>This function should only be used if the flag Talent is leading is set on in the global settings tab in system company settings.</p> <p>This will create the user in HRM, Visma.net</p> <p>If the parameter employeeID is passed in it will add the user email as a signin email address for the employee</p>	
<b>URL</b>	<a href="https:&lt;customername&gt;Talent.net/GenImport/PostReceiver.aspx/Aut.UserCreate?companyid=">https:&lt;customername&gt;Talent.net/GenImport/PostReceiver.aspx/Aut.UserCreate?companyid=</a>
<b>FunctionID</b>	SYS.131 The service user must be authorised for this.
<b>Parameters</b>	
companyID	companyID in the customers database (also specified on the URL during POST)
<b>Parameters in Body</b>	
emailAddress	The email address of the new user. This must be unique for the user and not assigned to a different user.
DefaultCompanyID	For on-premises customers this sets the company to open when you start Talent. This is not used for the Visma.net customers.
Firstname	The first name of the user
Lastname	The last name of the user
PreferredlanguageID	This options are 'NED' for Dutch and 'ENG' for English
UserType	This should be set to 'N' for normal user.
expirationDate	The date the user is allowed to sign in until.

employeeID	The employeeld of the user. This link the user to the employee and automatically add the email to the employee email
domainName	Domain of the user. Must be unique in conjunction with the loginname
loginname	Loginname of the user. Must be unique in conjunction with the domainname. This may be used for synchronizing with MS Exchange

## Aut.UserUpdate

<b>Aut.UserUpdate</b>	
<p>This function should only be used if the flag Talent is leading is set on in the global settings tab in system company settings.</p> <p>This will update the user in HRM, Visma.net.</p> <p>If the parameter employeeID is passed in, it will add the user email as a signin email address for the employee.</p>	
URL	<a href="https:&lt;customername&gt;Talent.net/GenImport/PostReceiver.aspx/Aut.UserUpdate?companyid=">https:&lt;customername&gt;Talent.net/GenImport/PostReceiver.aspx/Aut.UserUpdate?companyid=</a>
FunctionID	SYS.131  The service user must be authorised for this.
<b>Parameters</b>	
companyID	companyID in the customers database (also specified on the URL during POST)
<b>Parameters in Body</b>	
emailAddress	The email address of the user. This identifies the user to change.
DefaultCompanyID	For on-premises customers this sets the company to open when you start Talent. This is not used for the Visma.net customers.
expirationDate	The date the user is allowed to sign in until. The date format is yyyy-mm-ddThh:mm:ss
employeeID	The employeeld of the user. This link the user to the employee and automatically add the email to the employee email
domainName	The new Domain of the user. Must be unique in conjunction with the loginname
loginname	The new Loginname of the user. Must be unique in conjunction with the domainname. This may be used for synchronizing with MS Exchange

newEmailAddress	The new email address of the user when the email address has changed.
<b>Sample Call</b>	<p><a href="https://talent3a.stag.visma.net/GenImport/PostReceiver.aspx/Aut.UserUpdate?CompanyID=1">https://talent3a.stag.visma.net/GenImport/PostReceiver.aspx/Aut.UserUpdate?CompanyID=1</a></p> <p><b>Body to update emailaddress</b>  {"Aut.UserUpdate": { "CompanyID": "1", "eMailAddress": "iehrm.test4@visma.com", "newEmailAddress": "iehrm.test5@visma.com"}}</p> <p><b>Body to update expire the user account</b>  {"Aut.UserUpdate": { "CompanyID": "1", "eMailAddress": "iehrm.test4@visma.com", "expirationdate": "2019-11-12T11:18:32"}}</p>
<b>Updating email address</b>	<p>If the user is updating the email address. The change will be immediate if the email domain is registered in Visma Connect. For example the company registered @jobloggs.com and you update the email address  {"Aut.UserUpdate": { "CompanyID": "1", "eMailAddress": "john@jobloggs.com", "newEmailAddress": "John.Murphy@jobloggs.com"}}</p> <p>The change will be applied immediately</p> <p>If the domain is not registered  {"Aut.UserUpdate": { "CompanyID": "1", "eMailAddress": "john@jobloggs.com", "newEmailAddress": "JohnO@gmail.com"}}</p> <p>The change will be pending until the user logs into the new email address and confirms the change. The API call will return a message  {"message": "IMS050: eMailAddress update requires confirmation by user", "error": ""}</p> <p>When you change the email address it will only process the email address change and the expirationdate. All other changes in the api call are ignored</p> <p>To change the email address a second time when the first change is not complete, You must use the user ID or the original email address to identify the user. The system will not accept the pending email address in the emailaddress field</p>



## Aut.UserSearch

<b>Aut.UserSearch</b>	
URL	<a href="https:&lt;customername&gt;Talent.net/GenImport/PostReceiver.aspx/Aut.UserSearch?companyid=">https:&lt;customername&gt;Talent.net/GenImport/PostReceiver.aspx/Aut.UserSearch?companyid=</a>
Description	This will search for a list of users in Visma.net HRM based on a number of filters. It returns the Userids that meet the search. This is used in conjunction with Aut.GetUserInfo where you can get the full details of a single user.
Method	REST
FunctionID	SYS.131 The service user must be authorised for this.
<b>Parameters</b>	
companyID	companyID in the customers database (also specified on the URL during POST)
emailAddress	All or part of the email address of the user to search.
expirationDate	The date the user expires
employeeID	The employeeld of the user.
domainName	The Domain of the user.
loginname	All or part of the Loginname of the user.
ReturnUserDetails	Y then return the full details for the user information, N or empty then it only returns the userID
<b>Sample Call</b>	<a href="https://talent3a.stag.visma.net/GenImport/PostReceiver.aspx/Aut.UserUpdate/CompanyID=1/emailaddress=paul/ReturnUserDetails=Y">https://talent3a.stag.visma.net/GenImport/PostReceiver.aspx/Aut.UserUpdate/CompanyID=1/emailaddress=paul/ReturnUserDetails=Y</a>

## Aut.GetUserInfo

<b>Aut.GetUserInfo</b>	
URL	<a href="https:&lt;customername&gt;Talent.net/GenImport/PostReceiver.aspx/Aut.GetUserInfo?companyid=">https:&lt;customername&gt;Talent.net/GenImport/PostReceiver.aspx/Aut.GetUserInfo?companyid=</a>
Description	This will retrieve the user details of a single user based on the UserID or emailaddress. This can be used in conjunction with Aut.UserSearch where you can find the userid based on several filters
Method	REST
FunctionID	SYS.131 The service user must be authorised for this.
<b>Parameters</b>	
CompanyID	The Visma.Net.HRM companyID
UserID	The UserID of the user. Integer field.
emailAddress	The email address of the user
<b>Sample Call</b>	
<a href="https://talent3a.stag.visma.net/GenImport/PostReceiver.aspx/Aut.GetUserInfo/CompanyID=1/emailaddress=Paul.bradley@visma.com">https://talent3a.stag.visma.net/GenImport/PostReceiver.aspx/Aut.GetUserInfo/CompanyID=1/emailaddress=Paul.bradley@visma.com</a>	