

AvanzaMe – Merchant Cash Advance

Solution Outline

Version 1.0 – 7/22/2014

# Document Status and Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Author | Issue date | Revisions |
| 1.0 | Yatish | 07-16-2014 | 1 |
| 1.1 | Yatish | 07-22-2014 | 2 |
|  |  |  |  |

**Document Authorizations**

|  |  |  |
| --- | --- | --- |
| Name | Position | Email |
|  |  |  |
|  |  |  |

# Document Purpose

This document is intended to provide high level requirements of *AvanzeMe.com*. This document contains outline of high level design of the solution. For the technical design document to be completed following should be completed and signed off with client manager:

1. Business requirements
2. Functional requirements
3. Relevant Enterprise standards

Each section of the document contains notes/suggestions these suggestions.

### Table of Contents

[Document Status and Revision History 2](#_Toc16117)

[Version 2](#_Toc21002)

[Author 2](#_Toc27857)

[Issue date 2](#_Toc18457)

[Revisions 2](#_Toc31831)

[Name 2](#_Toc29320)

[Position 2](#_Toc26538)

[Email 2](#_Toc4056)

[Document Purpose 3](#_Toc5249)

[Table of Contents 4](#_Toc4893)

[1. Project Summary 9](#_Toc9462)

[2. Application Design 10](#_Toc27667)

[2.1. Interfaces 10](#_Toc11530)

[2.1.1 Sales Representatives 10](#_Toc25284)

[2.1.2 New System Users 10](#_Toc25899)

[2.1.2.1 Prequal Work flow 10](#_Toc27939)

[2.1.2.1.1 List of Merchants Pending for Review 12](#_Toc31198)

[2.1.2.1.2 Document Scanning 13](#_Toc24662)

[2.1.2.1.3 Data Entry 14](#_Toc28283)

[2.1.2.1.4 - Credit Card Volumes 17](#_Toc24720)

[2.1.2.1.5 - Scoring Job, Merchant Evaluation 18](#_Toc28981)

[2.1.2.1.6 - Offer Creation 18](#_Toc5453)

[2.1.2.1.7 - Offers Pending for Acceptance 21](#_Toc18143)

[Additional Information 22](#_Toc14539)

[2.1.2.2 Contract Work flow 23](#_Toc28307)

[2.1.2.2.2 Verification Call 25](#_Toc3141)

[2.1.2.2.3 Data Entry 26](#_Toc22665)

[2.1.2.2.4 Verification Tasks 29](#_Toc29489)

[2.1.2.2.4.1 Bank Information Verification 29](#_Toc14369)

[2.1.2.2.4.2 Corporate Documents Verification 30](#_Toc12423)

[2.1.2.2.4.3 Commercial Name Verification 31](#_Toc8163)

[2.1.2.2.4.4 Landlord Call 31](#_Toc32148)

[2.1.2.2.5 Review Task 32](#_Toc230)

[2.1.2.2.6 Contract Task 32](#_Toc693)

[2.1.2.2.7 Funding Task 33](#_Toc30503)

[2.1.2.2.8 Final Validation 34](#_Toc5281)

[Notes on SalesForce 34](#_Toc21103)

[Final Notes 34](#_Toc582)

[2.1.2.3 Renewals Work flow 34](#_Toc29484)

[2.1.2.3.1 List of Merchants that Qualify for Renewal 35](#_Toc10919)

[2.1.2.3.2 Merchant Evaluation 36](#_Toc866)

[2.1.2.3.3 Document Verification 37](#_Toc25244)

[2.1.2.3.4 Offer Creation and Acceptance 37](#_Toc12704)

[2.1.2.3.5 Renewal Review 38](#_Toc10338)

[2.1.2.3.6 Contract 38](#_Toc26137)

[2.1.2.3.7 Funding 39](#_Toc20781)

[2.1.2.3.8 Final Validation 39](#_Toc17588)

[2.1.2.3.9 Notes on Credit Card Volumes 40](#_Toc10118)

[2.1.2.3.10 Notes on SalesForce 40](#_Toc29645)

[2.1.2.3.11 Final Notes 40](#_Toc27124)

[2.1.2.4 Merchant Screen 40](#_Toc2815)

[2.1.2.4.1 Business Tab 41](#_Toc8202)

[2.1.2.4.2 Landlord Tab 43](#_Toc12369)

[2.1.2.4.3 Contacts Tab 43](#_Toc10613)

[2.1.2.4.4 Owners Tab 45](#_Toc14228)

[2.1.2.4.5 Profiles 45](#_Toc31301)

[2.1.2.4.6 Documents Tab 46](#_Toc7666)

[2.1.2.4.7 Activity Tab 47](#_Toc5851)

[2.1.2.4.8 Statements Tab 47](#_Toc19138)

[2.1.2.4.9 History Tab 49](#_Toc13275)

[2.1.2.4.10 Risk Evaluation Tab 49](#_Toc2066)

[2.1.2.4.11 Contract Tab 50](#_Toc17170)

[2.1.2.4.11.1 General Tab 50](#_Toc30870)

[2.1.2.4.12 Contract Activity tab 51](#_Toc15412)

[2.1.2.4.13 Sales Representative Tab 51](#_Toc15820)

[2.1.2.4.14 History Tab 51](#_Toc10458)

[2.1.2.4.15 Collections Tab 52](#_Toc22688)

[2.1.2.4.16 Final Notes 52](#_Toc29997)

[2.1.2.5 Permissions controls: 55](#_Toc24535)

[2.1.2.5.1 Permissions and Roles 56](#_Toc21274)

[2.1.2.5.2 User Profiles 57](#_Toc17146)

[2.1.2.6 The Collections Work flow 57](#_Toc11277)

[2.1.2.6.1 The Collections Screen 59](#_Toc17837)

[2.1.2.6.2 Datacrédito 59](#_Toc15735)

[2.1.2.6.3 Affiliations 59](#_Toc32391)

[2.1.2.6.4 Documents 60](#_Toc7866)

[2.1.2.6.5 Landlord 60](#_Toc15852)

[2.1.2.6.6 Payment Agreement 60](#_Toc8998)

[2.1.2.6.7 Legal 62](#_Toc9337)

[2.1.2.6.8 Generate Letter 63](#_Toc24877)

[2.1.2.6.9 Activity 63](#_Toc11772)

[2.1.2.7 Finances 64](#_Toc12286)

[2.1.2.8 Transfer Files 67](#_Toc23830)

[2.1.2.9 Reports 67](#_Toc14115)

[2.2. User Management 68](#_Toc21746)

[2.3. Data Output 68](#_Toc18386)

[2.3.1 Processor activities 68](#_Toc22971)

[2.3.1.1 File Transfer 68](#_Toc13845)

[2.3.1.2 Upload 68](#_Toc13748)

[2.3.1.2.1 Withdrawal Setup File (Upload) 68](#_Toc22525)

[2.3.1.2.2 Contract Overview File (Upload) 68](#_Toc12321)

[2.3.1.2.3 Status report File (UPLOAd) 68](#_Toc17259)

[2.3.1.3 Download 68](#_Toc27742)

[2.3.1.3.1 Response Setup File (Download) 69](#_Toc26430)

[2.3.1.3.2 Processor Activity Files (Download) 69](#_Toc31455)

[General file exchange work flow 69](#_Toc17133)

[2.3.1.4 General file SCHEDULE information 69](#_Toc29413)

[2.3.1.4.1 General file information 70](#_Toc22471)

[2.3.2 Reports 71](#_Toc25392)

[2.4. Data Management 71](#_Toc2426)

[2.5. Coding Requirements 71](#_Toc6395)

[Appendix 72](#_Toc3984)

*[File Detailed Format and Information](#_Toc30711)* [72](#_Toc30711)

*[Withdrawal Setup File (WSF)](#_Toc26620)* [72](#_Toc26620)

*[Contract Overview File (cof)](#_Toc13106)* [75](#_Toc13106)

*[Status report File (sts)](#_Toc31473)* [78](#_Toc31473)

*[Response Setup File (RSF)](#_Toc30208)* [83](#_Toc30208)

*[Processor Activity File (PAF)](#_Toc25243)* [86](#_Toc25243)

# 1. Project Summary

The new system will automate existing manual work done during loan application processing. A series of tasks are performed by Organization Employees for giving loan to a merchant/customer. Currently, these tasks are assigned manually. Files are moved to next officer once a task is completed by one officer.

These tasks are assigned based on their roles. The new system will automate process of task assignments. The Completed tasks will be moved in queues of next person to review application. Credit check of the merchant will be automated by using credit reports. Also, collection of Credit Card Sales will be automated using VisaNet and CardNet processors APIs. However there will be provision to manually enter statements.

New system will also be integrated with SalesForce.com. SalesForce and New System will be synced on regular basis to provide updated status to sales reps. Sales reps will be able to track status of merchants/contracts in sales force.

New system will send out notifications in form of emails. Email notifications will be sent out to predefined email groups. System will also have provision to add new persons to email sender lists.

# 2. Application Design

The new system is intended to manage tasks for loan officers. New system should fetch new leads from the SalesForce.com.

## 2.1. Interfaces

The New System will provide interfaces for 2 types of users:

## 2.1.1 Sales Representatives

Sales representatives will not be interacting directly with New System. However, Inputs done by sales rep in SalesForce.com will be treated as initial input for new system. Data from SalesForce will be imported into new system. Status of imported merchants and their contracts will be

## 2.1.2 New System Users

New system will have different roles viz. Underwriters, contract administrators etc. These users interact with system directly. System users will work on tasks assigned to them by Job scheduler. Multiple tasks can be queued for a user.

Each user will have access to a number of screens depending on their role and permissions. There are two main sections of application:

## 2.1.2.1 Prequal Work flow

Prequal work flow screens lists tasks for merchant creation. The new merchants are fed into system from sales force. For each merchant a series of tasks need to be completed as a pre-requisite. This includes verification of merchant information like Gross Yearly Sales. Below diagram shows the list of steps and their relation:

**1. List of3. merchants pending for** review

**2. Document Scanning**

**3. Data Entry**

**4. Send and Receive VisaNet and CardNet Request**

**5. Merchant Evaluation**

**6. Offer Creation**

**7. Offers Pending for Acceptance**

**8. Offer Acceptance**

**\*Note:**

Actions: There will be four buttons in each screen:

* 1. Complete - Complete the current step.
  2. Save - Save data but current step is not completed.
  3. Kick Back - Go to a previous step in work flow. All the information should be filled in with the old information, but if the user edits it, it will use the new information. Usually to the person that completed the work flow tasks, but it should also let us choose among the available users when we choose to kickback.
  4. Decline - Decline contract.

1. Work flow application will create a task for each step. These task will be assigned to users.
   1. Tasks will have following status:
      1. Completed
      2. Pending
      3. Pending CC Volumes
   2. All tasks should be completed to assign the next one, the only tasks that appear together are the 4 verification tasks in the contract work f low and the need to be completed before the review task is assigned.
   3. the assignment will be done like the original assignment. Randomly among all the users that have permission to work the task.
2. Roles cannot be changed unless all tasks are completed or reassigned to another user.
3. User with permission we could let a certain Role see the tasks for another Role and assign viewing or editing rights for these tasks

### 2.1.2.1.1 List of Merchants Pending for Review

Sales reps enter information about leads into SalesForce.com. Sales Reps convert lead into a potential merchant after he signs form in SalesForce.com. Only leads converted into potential merchant will be evaluated in new system

The first step will be to check for merchants pending for review. The way it should work is once a Sales Representative creates a merchant in SalesForce (clients that fill our form, not leads) and the feeding from SalesForce to the new system occurs, we should see a list of the created Merchants that have not been reviewed. Then we select one of those Merchants and start the review process. Once this process starts, the system should assign an ID number to the new merchant. For example, the first merchant will be ID 1, the second ID 2, and so on, unless the Merchant is being recreated for reevaluation purposes, in which case it will keep the ID that was assigned to it the first time it was created.

Here are few important checks for merchant creation:

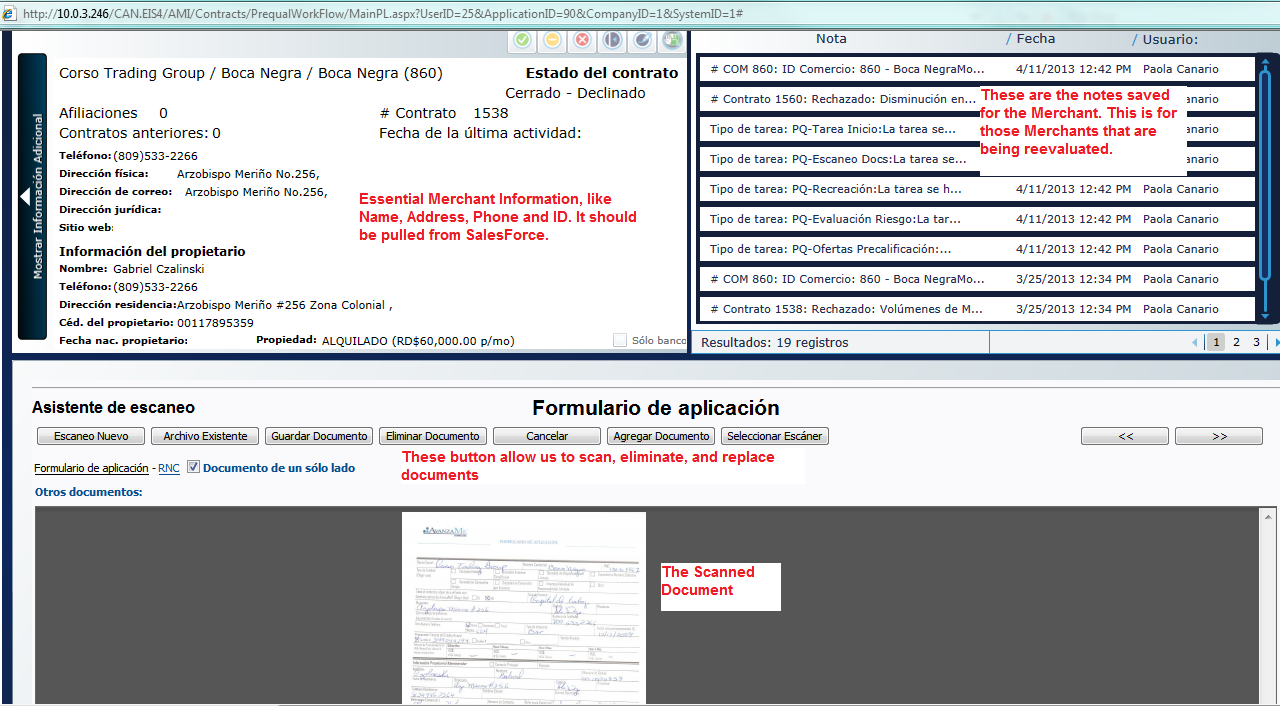
1. Before adding a new merchant in system user need to check if merchant already exists in system.
2. Existing record search can be done by RNC, Merchant Name, Processor Affiliate Number, Owner Id Number or Owner last name. If no record were found a pop-up could come up that asks "would you like to create a new merchant" when yes, the required information to add this merchant to the queue would be Merchant Name and Sales Rep. Then all the other information will be filled manually in the Data Entry.
3. We also need the option to do this step manually. That is by manually adding a Merchant to the list, and every field that requires information from SalesForce will be filled manually. Check for existing record will still be performed on merchant details being entered manually.
4. The merchant will be linked to same SalesRep who converted lead to Merchant in SalesForce.Sales Rep can be changed during the data entry process, but it could be change after the merchant is created. Sales Rep of a merchant cannot be changed after funding. Anyone can change it if they have these privileges.
5. There should be privilege to add two sales reps to a merchant. Both the sales reps should be able to work on same merchant/contract. Sales Reps can be divided as “Primary” and “Secondary”. Categorization of sales re will affect splitting their commissions.
6. Sales Rep will not affect merchant scoring or funding process.
7. Once merchant is created system will assign an ID to merchant. If merchant already exists his ID will be used for reference. The new or existing Merchant ID will be updated in SalesForce.com.

**\*Note:** All users can add notes for merchant in new system.

### 2.1.2.1.2 Document Scanning

The second step is the Document Scanning, here we should be able to scan any document we have, but the only required document will be the Application Form, which we should be able to upload in PDF or any image format (it should allow us to scan PDFs, docs, xlsx, and any image file):

Our current Doc Scan screen looks like this (my notes in red):



Uploaded documents will be linked to Merchants or contract depending upon type of document. Application form will be linked to a merchant. Users should be able to view documents in Merchants Profile screen.

**\*Note:** 1. Application Form is mandatory to upload.

2. The note should have a field above it that adds the default email address and lets us add any address we want.

### 2.1.2.1.3 Data Entry

The third step is the data entry. All this information should be filled from SalesForce, but we should be able to modify anything, in case we have to. The information we should have here is:

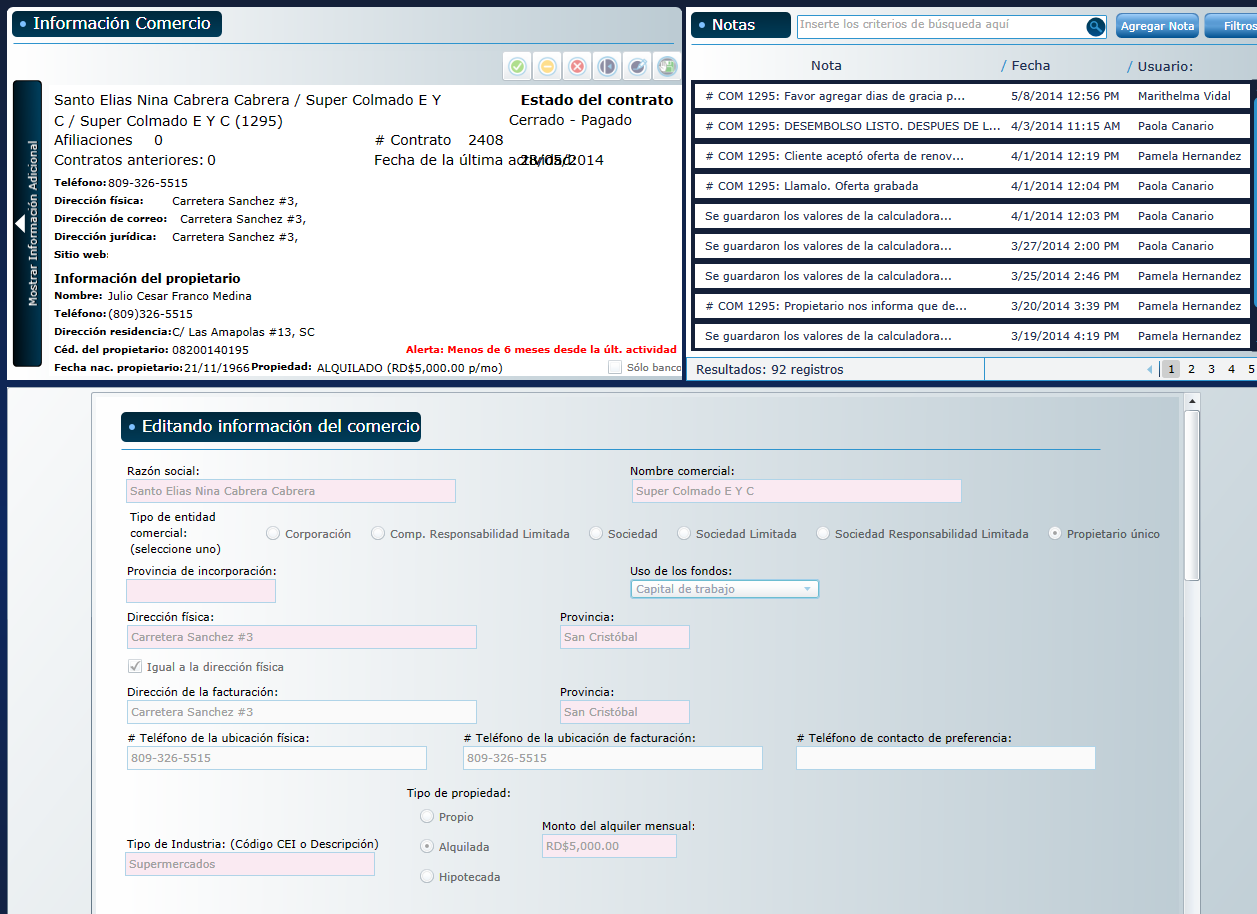
* Name of the Company
* Name of the Business
* RNC: This is the ID the government assigns to every company. It should have a maximum of 11 numbers.
* Address
* City
* Province: This should be a drop down list with the following options:
* Azua
* Bahoruco
* Barahona
* Dajabón
* Duarte
* Elías Piña
* El Seibo
* Espaillat
* Hato Mayor
* Independencia
* La Altagracia
* La Romana
* La Vega
* María Trinidad Sánchez
* Monseñor Nouel
* Montecristi
* Monte Plata
* Pedernales
* Peravia
* Puerto Plata
* Hermanas Mirabal
* Samaná
* Sánchez Ramírez
* San Cristóbal
* San José de Ocoa
* San Juan
* San Pedro de Macorís
* Santiago
* Santiago Rodríguez
* Santo Domingo
* Valverde
* Distrito Nacional
* Telephone Number: Dominican telephone numbers have 10 digits in the format: (xxx) xxx-xxxx.
* Email
* Industry Type: We will annex all the options that will make up this drop list. This comes from the predetermined MCC codes.
* Type of Property: This should be a drop down list with two options, Owned and Rented.
* Rent Amount: This should be a currency field.
* Gross Yearly Sales: This should be a currency field.
* Processor Company: Either CardNet or VisaNet.
* Affiliate Number: The affiliate number for the business with either CardNet or VisaNet (or both). We should be able to add more than one affiliate number. This is the number the Processor company identifies the business with. It affects the Merchant because the Processor Company will send us CC sales for that number exclusively, and it is through that number(s) we would be taking payments. We could add more than one processor number for more than one processor.

Then we should have the Owner’s Information in the same screen:

* Name
* Last Name
* ID or Passport Number: A maximum of 11 characters, either numbers or letters.
* Phone Number
* Cell Phone Number: In the format (xxx) xxx-xxxx.
* Address
* Email

Users should be able to add multiple owners for a merchant.

Our current Data Entry Screen looks like this:



As you can see, the top part is very similar to the Doc Scan screen, and bellow is all the information I have indicated above.

The name of the company is the legal name of the company that owns the business and the name of the business is the name the business uses for its operations. For example, Company A could own Restaurant Food, but it could also own Store One, which would be created as a separate merchant. So we would have two merchants, Store One and Restaurant Food, owned by the same company. This is called affiliations, and affiliation is when a Legal Name has more than one business. this way we can search a merchant by affiliations to see all their related business and performance.

All the information in Data Entry page is required. However, All fields can be edited by authorized users should be able to edit information on Data Entry screen.

**\*Note:** 1. OwnerID is unique, It's the number the Government assigns to every citizen. 2. The system should check for Merchants with similar information in this fields and show us a list so we can manually decided. Also, we should have a search box where we can type a RNC, Owner ID, Affiliate number, etc, and it will show us a list of every Merchant that matches any of these fields, and then we will decide if it's a duplicate or no. 3. See No.2.

2. In our system, every Merchant is one Business. A company or an owner might have more than one Merchant and, thus, get more than one loan. 2. The person signing the contract will be determined by the legal documents the client provides.

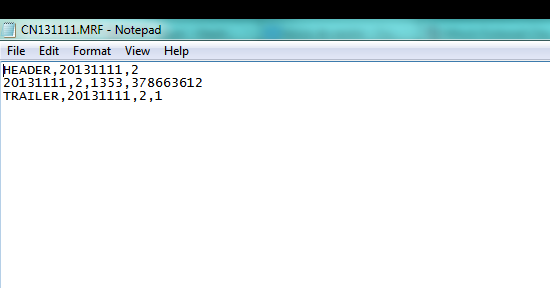
### 2.1.2.1.4 - Credit Card Volumes

At this point the system should send a request to VisaNet and CardNet for the credit card volumes, then, once we receive the request, the work flow continues. There are times when we receive the volumes outside the normal process, through email or some other way, which is why we need to have the option, in our system, to manually type the credit card volumes, so that it can go on to the scoring job.

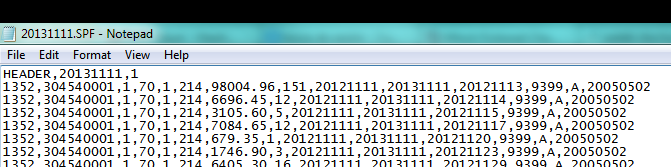
Once the Merchant is created, the system will automatically send a Request through an MRF file to the Credit Card Processor Company. This MRF file contains the Date of the Request, the ID of the Merchant in our System, the Merchant’s affiliate number with the Processor and the Company ID. System should request maximum available data from processors. Visanet usually sends no more than 12 months, but cardnet has sent more than 20 months of data

Also, at this point we would like the system to automatically send the physical Form we scanned earlier to CardNet (but not to VisaNet), perhaps through the same way we will send the Volumes request. (This is something we will work out in detail later)

Here is an example of an MRF file:



The Processor Company will respond with an SPF file containing the monthly credit card volumes of the merchant requested. Here is an example of an SPF file:



The SPF file will import into the system all the credit card volumes showing monthly. Also it should import the date the business started accepting credit cards, CC processor industry type.

### 2.1.2.1.5 - Scoring Job, Merchant Evaluation

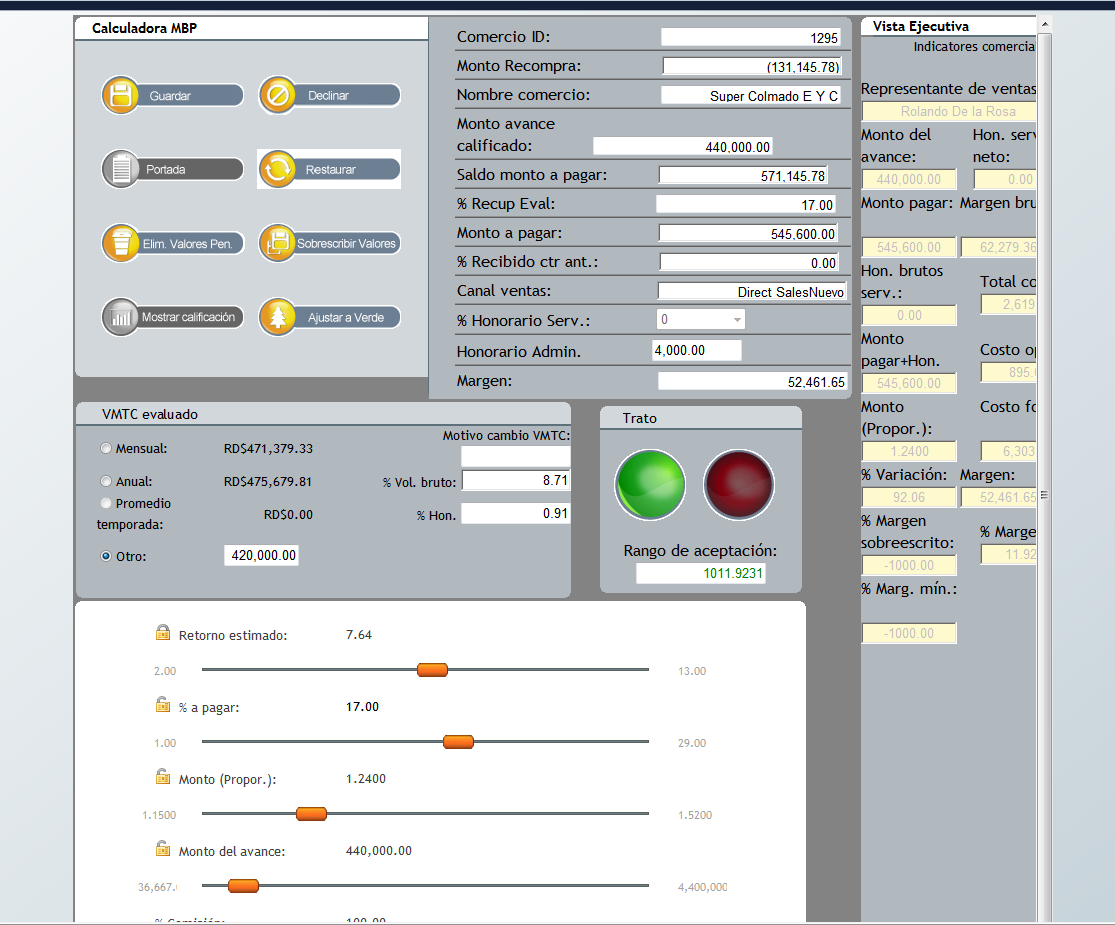
After we have received the Credit Card Volumes through the SPF file or manually introduced, the new task is going to be the Merchant Evaluation task. Here our officer will generate the score for the Merchant, take a look at the Datacrédito and the Credit Card volumes, and then decide whether to go to the Offer Creation task or decline the Pre-qualification.

Every Datacrédito report we pull should be saved as a Document in the Merchant. And we should have a list of every Datacrédito report we have ever pulled for the Merchant and be able to access them at any time.

### 2.1.2.1.6 - Offer Creation

Here we create the offers. One Merchant could have any number of offers saved. To create the offer we use a Calculator Application.

This is how the calculator currently looks:



This is an idea of what it could look like (simplified):



*This should pull the Gross Yearly Sales automatically from the information entered in the Merchant Creations process, but it should let us change it manually in case we need to. The Average Monthly Credit Card Sales should be calculated from the volumes. It should also let us change in manually if we need to.*

*This is the Offer Calculator. The components of the offer are the Time in months the contract is expected to last, the loaned and owed amount, the proportion (This is the Owed Amount divided by the Loaned Amount, for example, if the loan is going to be 100, and the owed amount will be 150, then the proportion is 1.5) and the retention percentage, which is the fixed percentage we will be discounting from the credit card sales of the Merchant as payment. Every field should have sliders to move around the amounts until the officer is comfortable with the offer.*

*This button is to save the current offer, so that the officer can do more than one offer to present the client.*

*Here it should display all the saved offers and the details of the offers.*

*Once the officer has saved all the offers he needs to, he can hit complete or decline (as usual). If he hits complete, then the system should generate a PDF file with all the saved offers and the details. This is the file the officer is going to send the Sales Representatives so that they can present them to the client.*

At the same time, it should send an email to the Sales Rep in the following format:

ID # XXX: Merchant Name

Sales Representative Name: John Doe

Business Owner Name: Jack Doe

Offer Date: DD/MM/YY

Expiration Date: DD/MM/YY

Offers

Offer 1

MCA Amount: RD$104,000.00

Owen Amount: RD$127,920.00

Price: 1.2300

% Retention: 29.00%

Repayment Time (months): 5.03

Pending Requirements:

Additional Notes:

**\*Notes:**

1. This is where we will model the decline letter and will be sent to the email the customer provided as well as sent to the sales rep, sales manager, also any other email address could be included manually.
2. If Gross Yearly Sales is modified while Offer generation is should be updated in merchant details.
3. If Gross yearly Sales is modified after generating an offer: The system should calculate if that offer still applies for the new Gross Yearly Sales, and if it doesn't, it should let us know, maybe we should have a Offer Does Not Apply sign in the offer. Remember the retention % can never be more than 9% of the total gross sales, so if for example the gross sales are less once we analyze the merchant, then there would be a probability that we overpass that 9% threshold.
4. Must have: Maintain log of change in Gross Yearly Sales changes while Offer generation.
5. If the last scoring is less than 60 days prior to when reevaluating, we can reactivate this merchant, if not it has to go through the complete PQ work flow once again.
6. On Offer expiration email notification should be sent to The Sales Rep, Sales Manager, General Manager and Risk Manager.
7. In the prequal offer everything could be changed thrue the calculator and if applies. We don't need Merchant's consent to change it. It would be good to keep track of the changes.

### 2.1.2.1.7 - Offers Pending for Acceptance

This should be another list where we have all the pending offers with the option to accept any offer and kick off the contract work flow. If the offer is declined, the Agent will choose any of the decline reasons and the Merchant will be marked as declined, with the possibility of reactivation within a 60 day window. If the contract has not been reactivated after 60 days, then the Merchant must be recreated in the usual way. Depending on the decline reason the system should constantly and automatically revaluate merchants, it should put the merchant on the initial precualification task. Bellow is the amount of time after decline that it should be put in the prequalification list:

**Declined By Avanzame**:

* Unacceptable Credit Score – 90 days
* Merchant doesn’t fulfill Gross Yearly Sales requirement –90 days
* Merchant doesn’t have authorized Credit Card Processor – 120 days
* Merchant doesn’t fulfill minimum CC transactions requirement -
* CC sales have gone down – 90 days
* Less than 5 months accepting CC payments – 150 days
* CC volumes don’t fulfill requirements–120 days
* The person who signed the form is not an authorized person – 120 days
* The Merchant has already been created by another Rep
* Merchant stopped processing.

**Declined By the Client:**

* The client doesn’t want a loan at this time – 90 days
* The client declines because of the Administrative Expenses – 90 days
* The client declines because of the price – 90 days
* The client declines because of the Retention Percentage – 60 days
* The client wants a bigger loan – 120 days
* The client wants a higher repayment time – 90 days
* The client is unreachable / refuses to return calls – 90 days
* The client refuses to present required documents – 120 days
* Merchant has different financing option – 150 days

There should be a way when we decline that we tell the system to never revaluate a specific merchant.

Every pending offer has an expiration time of 30 days. If after the 30 days have passed, the offer has not been neither declined nor accepted, it will be marked as expired and no longer valid.

We need to be able to edit any Offer parameter before accepting it and, once it’s been accepted, the system should automatically send an email with the accepted offer to the Sales Rep and Management people we will indicate at a later time.

**\*Note:**

1. On Offer acceptance/denial system should send notification to a specific group of emails such as the sales manager, underwriting manager, general manager and ops team, as well as any manually entered email.
2. Offers will be generated by Under Writers and admin users.

### Additional Information

Every step of the work flow should have the following buttons on the top:

* **Complete Button**: This is the button that we will click once we are ready to complete the task and move on to the next task.
* **Decline Button**: This is the button that we will click if we want to decline the Merchant at any point in the process. It will ask the user to specify the Decline reason, which will come from the following list:

**Declined By Avanzame**:

* Unacceptable Credit Score
* Merchant doesn’t fulfill Gross Yearly Sales requirement
* Merchant doesn’t have authorized Credit Card Processor
* Merchant doesn’t fulfill minimum CC transactions requirement
* CC sales have gone down
* Less than 5 months accepting CC payments
* CC volumes don’t fulfill requirements
* The person who signed the form is not an authorized person
* The Merchant has already been created by another Rep
* Merchant stopped processing.

**Declined By the Client:**

* The client doesn’t want a loan at this time
* The client declines because of the Administrative Expenses
* The client declines because of the price
* The client declines because of the Retention Percentage
* The client wants a bigger loan
* The client wants a higher repayment time
* The client is unreachable / refuses to return calls
* The client refuses to present required documents
* Merchant has different financing option
* **Kick Back Button**: This is the button that we will click if we want to kick back the Merchant from a task to a previous task of choosing.
* **Notes Button**: This button should open the Notes application that will let us save any note or comment to the Merchant.
* **Documents Button**: This button should open a pop-up that will let us display any document that has been scanned and saved to the Merchant. We need to have this because sometimes our Agent needs to verify some information with the physical form.

## 2.1.2.2 Contract Work flow

Contract work flow screens lists tasks for contract review. Contracts of the merchants are reviewed in contracts work flow. Additional documents for contract are uploaded and contract completed. Diagram on next page shows the list of steps and their relation:

**2.1.2.2.1 Document Scanning**

**Contract Approval**

**Review**

**3. Data Entry**

**Contract being Signed**

**4d. Commercial Name Verification**

**4c. Corp Documents Verification**

**4b. Landlord Call (if rented)**

**4a. Bank Information Verification**

**8. Final Validation**

**Gathering Documents**

**1. Document Scanning**

**2. Verification Call**

**Processing Documents**

**5. Review Task**

**6. Contract**

**7. Funding**

**Funded**

**Note: The red squares signify a change in status for the contract.**

The first step of the Contract process should be a search screen much like the first screen in the Merchant Creation workflow. In this screen the Officer will look for the Merchants that have accepted offers and then begin the process of working the contract, which will start with the Contract Document scan. At this point the status of the Contract should be Gathering Documents, both in the system and in SalesForce.

The Contract Doc Scan Task should be exactly the same as the Prequalification Document Scan task, except for the documents that are going to be required to complete. The required documents are:

* Legal Documents of the Company (PDF, JPG, GIF, PNG, DOC)
* Commercial Name Verification Screenshot (JPG, PDF, PNG, GIF)
* RNC Screenshot (JPG, PDF, PNG, GIF)
* ID or Passport (JPG, PDF, PNG, GIF)
* Lease Contract or Land Title (PDF, JPG, GIF, PNG, DOC)
* Null Check (JPG, PDF, PNG, GIF)
* Bank Statements (PDF, JPG, GIF, PNG, DOC, XLSX)

Although these documents are required, we should have a way of marking any of these documents as pending (Except for the Legal Documents of the Company) to be uploaded later in the workflow.

After this task is completed, the Status will be automatically changed to *Processing Documents* both in our system and in SalesForce.

**\*Note:**

1. Its the same Document Scanning task, just that in PQ only the Application Form will be required, in the contract it should already have the Form uploaded and then let us upload all the other documents specified in the Contract Work flow document.
2. Yes, we specified this in the document. The legal Corporate document can never be marked as pending. As for the others, we can't complete the commercial name verification task without the commercial name screen shot, the Bank Verification task without the uploaded Null Check, the Landlord Verification without the Lease Contract uploaded.

### 2.1.2.2.2 Verification Call

The Verification Call task is where the Officer is going to call the Merchant owner and ask to confirm some information. So, this screen should consist of the script of the call (which we will provide you), the fields for the Officer to modify accordingly and a box with a list of the documents that were marked as pending in the document scanning task so that we can ask the client for the documents.

In this screen the Officer will see all the questions and confirm the information we already have. All the fields should be filled automatically if we already got the information in previous stages; if not, then the box should be left blank for the officer to fill. In every case, we should be able to modify all the fields except the loan and owed amount. As usual, the officer can complete or decline.

### 2.1.2.2.3 Data Entry

This is very similar to the PQ Data Entry. The information we will require in this screen will be:

* Name of the Company
* Name of the Business
* RNC: This is the ID the government assigns to every company. It should have a maximum of 11 numbers.
* Address
* City
* Province: This should be a drop down list with the following options:
  + - Azua
    - Bahoruco
    - Barahona
    - Dajabón
    - Duarte
    - Elías Piña
    - El Seibo
    - Espaillat
    - Hato Mayor
    - Independencia
    - La Altagracia
    - La Romana
    - La Vega
    - María Trinidad Sánchez
    - Monseñor Nouel
    - Montecristi
    - Monte Plata
    - Pedernales
    - Peravia
    - Puerto Plata
    - Hermanas Mirabal
    - Samaná
    - Sánchez Ramírez
    - San Cristóbal
    - San José de Ocoa
    - San Juan
    - San Pedro de Macorís
    - Santiago
    - Santiago Rodríguez
    - Santo Domingo
    - Valverde
    - Distrito Nacional
* Telephone Number: Dominican telephone numbers have 10 digits in the format: (xxx) xxx-xxxx.
* Email
* Industry Type: We will annex all the options that will make up this drop list. This comes from the predetermined MCC codes.
* Type of Property: This should be a drop down list with two options, Owned and Rented.
  + - Rent Amount: This should be a currency field.
    - Gross Yearly Sales: This should be a currency field.
    - Processor Company: Either CardNet or VisaNet.
    - Affiliate Number: The affiliate number for the business with either CardNet or VisaNet (or both). We should be able to add more than one affiliate number.
* Use of the Advance: This will be a drop list with the following options:
* Debt Payment
* Remodeling
* Tax Payment
* Equipment
* Cash Flow

Then we should have the Owner’s Information in the same screen:

* Name
* Last Name
* ID or Passport Number: A maximum of 11 characters, either numbers or letters.
* Phone Number
* Cell Phone Number: In the format (xxx) xxx-xxxx.
* Address
* Email

Then we should have the Landlord Information (this only applies if the Type of Property is marked as Owned):

* Company: This is the name of the company that owns the building.
* Name: This is the name of the person that represents the company or owns the building.
* Last Name: This is the last name of the person that represents the company or owns the building.
* Telephone Number

Then we should have the Bank Information:

* Name of the Bank
* Bank Account Number
* Bank Code

Then we should have the Trade References:

* Name of the Reference
* Telephone Number

**Note:**

This should have some sort of way for us to add more than one reference in case we need to.

There should also be a Gross Yearly Sales Calculator that will work through two methods.

Method 1

We will manually put in the Credit Card percentage of the monthly sales and it will use that to calculate the total Gross Yearly Sales. For example, if the Merchant sells US$100,000 through credit cards every month (or whatever we receive from CardNet and VisaNet) and we set the Credit Card monthly sales percentage at 40%, then the system will take the Cash Sales as 60% and calculate the total Yearly Sales. If credit card sales are US$100,000, that represent a 40% of the total monthly sales, then the total monthly sales would be US$250,000 (US$100,000 / 40%), and the total yearly sales would be US$3,000,000 (US$250,000 \* 12). This may seem confusing, please ask for further clarification if you need to. The goal here is to calculate the Gross Yearly Sales with the credit card volumes we already have and a percentage we will manually put in.

Method 2

*We will type in three months of Bank Statements and the system will use that and the credit card volumes to calculate the yearly sales. It will do so as in the previous method, except that the percentage will be determined with the Bank Statements amount. We will annex an excel sheet with the formulas we use to calculate the Gross Yearly Sales, please let us know if you don’t understand.*

After this task is complete, the system will simultaneously assign the following four verification tasks.

**\*Note:**

1. Information like the Trade References or the Landlord information will be entered here.
2. References are people the Merchant does business with, we need to have them in the system in case we want to check some information with them about the client.
3. Every field can be edited in DE screen.
4. Authorized owners will be the ones that sign the contracts and the system should let us know in the Contract Task who are the authorize owners so we know who we have to get to sign.
5. If the Merchant has affiliate numbers from both. You should pull data from ALL the affiliate numbers associated to the Merchant.
6. We will either manually type in the average percentage or we will put in three months of total monthly sales (we can get them from the bank statements) and you will divide the cc sales by the total sales and that should give you the cc percentage.\
7. Data entry step cannot be skipped by any user.

### 2.1.2.2.4 Verification Tasks

Note:

1. They all have to be completed and they are independent from each other. In the case of the Landlord Verification, this task will only show up if during the DE we select the type of property as Rented, otherwise the business does not have a landlord and we don't need to verify this.

#### 2.1.2.2.4.1 Bank Information Verification

In this task we will check the Bank Information in our system against the information on the Check we uploaded in the Documents Scanning task (if it was not uploaded and, instead, marked as pending, it should not let us complete this task without uploading the document). The screen could look something like this:



Note:

1. The Checks have written on them the bank account number and name, we upload a picture of the check and then write the information from that into our system.

#### 2.1.2.2.4.2 Corporate Documents Verification

This is task is similar to the previous task, except it should display the Legal Documents scanned in the Document Scanning task, and it should show two sets of fields, one for the Business and one for the Owners.

For the Business we will check the following:

* Name of the Company
* Address of the Company
* RNC

For the Owners we will check the following:

* Name
* Last Name
* ID or Passport number
* Telephone Number

For the owners, it should let us add as many owners as we need to, but we need a way to mark some of the owners as Authorized to Sign the Contract; maybe a check box.

#### 2.1.2.2.4.3 Commercial Name Verification

In this task the officer will make sure the information in the Commercial Name Verification Screenshot (which the officer will take from a webpage like yellow pages and upload it in the doc scan task) matches the information we have in the system. If the screenshot has not been uploaded because it was marked as pending, we should not be able to complete this task without uploading it. It should have the Business Name, Telephone Number, Address, City and Province.

It could look something like this:



#### 2.1.2.2.4.4 Landlord Call

In the Landlord Verification Call task, much like the regular Verification Call, the officer will follow a script (which we will provide) and then verify the required fields. The screen could look like this:



After these four tasks are completed, the system will change the status to *Review*.

**\*Note:**

1. This step will be skipped if the Type of Property was marked as owned during the Data Entry.

### 2.1.2.2.5 Review Task

This task will have all the previous tasks in different tabs in the same screen, so that another officer can review the first officer’s work. For example, it will have a document scan tab with all the documents the first officer uploaded, then a Verification Call tab with the fields the first officer filled, then a Data Entry tab with the data the first officer completed, and the rest of the tasks.

The goal in this task is for a different officer to review *everything* the first officer did from one screen. After this task is completed, the system should automatically change the status to *Contract Approval*, and push the change in status to SalesForce.

**Note:**

1. The review task should have all the tasks the officer completed organized in different tabs. For example, one tab for the documents, one tab for the data entry, one tab for the landlord info, etc. This will be done by a different officer than the one who completed the first part of the work flow. The goal here is for a different officer to review the first officer's work.
2. All previous tasks will be shows in each tab. Like we will have 5 Tabs, One for Document Scanning other 4 for all review tasks?

### 2.1.2.2.6 Contract Task

This task should be exactly the same as the Review Task (assigned to a different person) but with an extra tab that will let us print the Contract so that we can send it to the client. We will provide you with the format of the contract and all other documents we need to print in this tab.

It should also include another task called Administrative Expenses. In here we should see the Administrative Expenses we charge for each contract. They work in a scale, if the Loaned Amount is less than $500,000.00, we charge $4,000.00, if it’s from $500,000.00 to $999,999.99 we charge $8000, and anything greater we charge $12,000.00 (please note that this table could be updated at any moment). It should show the corresponding Expense by default, but it should let us manually change it, in case we want to override the scale.

After this task is completed, the status should be automatically changed to *Contracts being Signed* both in the system and in SalesForce.

Note:

1. If we have the credit card percentage we can calculate the cash percentage, and from there the total sales.

### 2.1.2.2.7 Funding Task

This is the task that the UW will complete once the money has been transferred to the client. It should have two checkbox, the first one “Contract Received and Reviewed”, to be marked once we receive the signed contracts from the client and the second one will be “Merchant Funded” to be marked once the merchant has the money. This task cannot be completed without both checkboxes marked and the system should keep a log of the user who marks the checkboxes.

This is the information the screen should display in this task:

* Bank Name
* Bank Account Number
* Bank Account Name
* Parameters of the Offer
  + MCA Amount
  + Administrative expenses
  + Total Funding Amount (MCA Amount – Administrative expenses)
* Owners that were marked as Authorized to Sign the Contracts back in the Corporate Documents task

After this task is completed, the status should be automatically changed to Funded both in the System and in Salesforce. One a contact is funded and status is updated this status could not be changed by anyone, only should change when the merchant has paid off the MCA.

**Note:**

1. Funding in the system will be just to mark the check boxes specified in the document. The actual funding will be done outside the system, and once done, we will go into the Funding task and mark the checkboxes.

### 2.1.2.2.8 Final Validation

In this task we will upload the signed contract and final legal documents, so it should have a documents uploading app similar to the Doc Scan task. After this task is completed, the workflow is done.

Note:

1. This is where we upload the signed documents. It needs to happen after funding because in our process we receive the final legal documents after the client has the money in the bank

### Notes on SalesForce

The exchange between the system and SalesForce does not need to happen in real time. Maybe we can set up three or four cut times where the system will gather all the status, offers and information and feed it back to SalesForce.

### Final Notes

All the tasks should have a timestamp of when they were assigned and when completed and by whom. This way we can track the complete time of any case or group of cases, even identify by status changes.

Every screen should have the following buttons:

* Complete
* Decline
* Notes
* Documents
* Skip Step: This button can only be used by a user with admin privileges and will be used to skip any step we deem unnecessary. This should *not* be used by regular users.

## 2.1.2.3 Renewals Work flow

Merchants who has paid off their loan up to 60% (we should be able to change this %) of their owned amount can ask for more loan. To process those merchants a complete work flow will be added to new system. This is called Renewals Work Flow. Below diagram shows steps of Renewals work flow:

### 2.1.2.3.1 List of Merchants that Qualify for Renewal

**3. Document Verification**

**4. Offer Creation and Acceptance**

**5. Renewal Review**

**6. Contract**

**7. Funding**

**8. Final Validation**

**Review**

**Renewal Approval**

**Contracts being signed**

**Funded**

**Note: The red squares signify a change in status for the contract.**

**1. List of Merchants that qualify for Renewal**

**2. Renewal Evaluation**

**Applies for Renewal**

**Renewal Process**

This is the first step of the Renewal work flow. It should show a list of all the Merchants that qualify to be renewed. The requirement for an active contract to be renewed is that the contract has at least a 60% of the owed amount paid.

So, the list should show all the Merchants with 60% paid and it should give us the option to mark any of these Merchants as pending to be reevaluated at a later time. If we select this, it should ask us when (what exact date) we want it to start showing up in the list again or after what number of days or after a certain amount has been paid. For example, let’s say Merchant A has already paid 60% of his contract, so he shows up in the list. But when we see it in the list, we decide we don’t want to reevaluate Merchant A right now so we mark it as pending for reevaluation and the system shows us a pop up asking us if we want it to show up in the list after a certain date, number of days, or after the contract has a higher percentage paid.

If we want to evaluate it now, then we select the Merchant and kick off the process. At this point the status should be “Applies for Renewal”.

\*Note:

1. New contract will be created.
2. Status of new contracts will be "In Renewal Process".
3. Merchant status, it will be tied to the Active contract. If the active Contract is being paid well within the agreed time, it will be "Merchant paying below time". If the Contract is being paid around the time agreed then it will be "Merchant paying on time". If the Contract is being paid well above the time agreed then it will be "Merchant paying above time". If it has no active contract then the Merchant status will simply be "Inactive". So, in the Renewal process the Merchant status will be decided by the active contract. If the Merchant is in collection it will be inactive, but if the Merchant has not been paying for the last week then it should change the status to "Active Investigation".

### 2.1.2.3.2 Merchant Evaluation

Here our officer will generate the score for the Merchant, take a look at the Datacrédito and the Credit Card volumes, and then decide whether to go to the Offer Creation task or decline the Renewal.

Every Datacrédito report we pull should be saved as a Document in the Merchant. And we should have a list of every Datacrédito report we have ever pulled for the Merchant and be able to access them at any time.

If the renewal is not declined, it should change the status to “Renewal Process”.

\*Note:

* Evaluating officer will trigger credit report pull.
* New contract will be created before merchant evaluation starts.

### 2.1.2.3.3 Document Verification

This step will be like a regular Document Scanning screen, but it will be called Document Verification. It will work exactly like a doc scan screen, but with no required uploads. The officer will use this screen to verify that the required documents are all up to date so that he knows if he needs to ask the client for an updated document.

\*Note:

* Renewals officer will decide if existing documents can be used or new documents are required.

### 2.1.2.3.4 Offer Creation and Acceptance

This task will be the calculation of the new offer. The calculator will resemble the regular contract calculator; it will only change in that we need to show somewhere the information for the active contract, as in how much was the loaned amount, how much is owed, and all the other information. Since we need to pay off the remaining amount of the previous contract, it should show how much money the client will be actually getting. For example, let’s say Merchant A has $40,000.00 left to pay of his current contract, and we’re offering him a new loan of $100,000.00, we need to pay off first the $40,000.00 he owes us, so we would actually be giving him $60,000.00.

Since this task will be done with the client on the phone, we need to be able to accept in this task any of the saved offers. Once an offer is accepted, it will go to Renewal Verification.

If the offer is accepted, it should change the status to “Review”.

This screen could look like this:



**\*Note:**

1. The existing contract will remain untouched until the moment the new contract is funded and the previous contract is paid off with part of the amount of the new contract.
2. Status of contract after acceptance will be updated to "In Review".
3. Once the offer is created and saved in the offer creation task.

### 2.1.2.3.5 Renewal Review

This task will have the different tabs with the documents and all the information of the Merchant so that another officer checks the first officer’s work.

This is much like the Review task in the regular Contract Work flow. After it’s done, the status should be changed to “Renewal Approval”.

Note: Status of contract will be updated to "Renewal Waiting for Approval".

### 2.1.2.3.6 Contract

This task should be exactly the same as the Review Task (assigned to a different person) but with an extra tab that will let us print the Contract so that we can send it to the client. We will provide you with the format of the contract and all other documents we need to print in this tab.

It should also include another task called Administrative Expenses. In here we should see the Administrative Expenses we charge for each contract. They work in a scale, if the Loaned Amount is less than $500,000.00, we charge $4,000.00, if it’s from $500,000.00 to $999,999.99 we charge $8000, and anything greater we charge $12,000.00 (please note that this table could be updated at any moment). It should show the corresponding Expense by default, but it should let us manually change it, in case we want to override the scale.

After this task is completed, the status should be changed to “Contract being signed”.

Note: Status of contract will be update to "Contracts Being Signed".

### 2.1.2.3.7 Funding

This is the task that the UW will complete once the money has been transferred to the client. It should have two check box, the first one “Contract Received and Reviewed”, to be marked once we receive the signed contracts from the client and the second one will be “Merchant Funded” to be marked once the merchant has the money. This task cannot be completed without both check boxes marked and the system should keep a log of the user who marks the check boxes.

This is the information the screen should display in this task:

* Bank Name
* Bank Account Number
* Bank Account Name
* Parameters of the Offer
  + MCA Amount
  + Administrative expenses
  + Total Funding Amount (MCA Amount – Administrative expenses – Owed amount from previous contract)
* Owners that are marked as Authorized to Sign the Contracts

After this task is completed, the status should be automatically changed to “Funded” both in the System and in Salesforce. One a contact is funded and status is updated this status could not be changed by anyone, only should change when the merchant has paid off the MCA.

\*Note:

1. Status of new contract will be "Funded".
2. Status of existing contract will change to “Closed”.

### 2.1.2.3.8 Final Validation

In this task we will upload the signed contract and final legal documents, so it should have a documents uploading app similar to the Doc Scan task. After this task is completed, the work flow is done.

### 2.1.2.3.9 Notes on Credit Card Volumes

In this work flow the system should work with the Credit Card Volumes as calculated with our system and should not request new volumes from the Processor Companies unless the previous contract ended more than 30 days before the time of the renewal.

If the contract is active or ended less than 30 days before the renewal, it should be easy for the system to calculate the volumes. If an active contract has a retention percentage of 10% and is paying us 10,000 every month, then his credit card sales for the month must be 100,000.00 (which is 10,000 / 10%).

### 2.1.2.3.10 Notes on SalesForce

The exchange between the system and SalesForce does not need to happen in real time. Maybe we can set up three or four cut times where the system will gather all the status, offers and information and feed it back to SalesForce.

### 2.1.2.3.11 Final Notes

All the tasks should have a time stamp of when they were assigned and when completed and by whom. This way we can track the complete time of any case or group of cases, even identify by status changes.

Every screen should have the following buttons:

* Complete
* Decline
* Notes
* Documents
* Skip Step: This button can only be used by a user with admin privileges and will be used to skip any step we deem unnecessary. This should not be used by regular users.

## 2.1.2.4 Merchant Screen

This is the screen where we’ll go when we want to check or modify any information from an already created Merchant. This screen will be divided in many different tabs, which we will access depending on the information we want to see. To access this screen, we should be able to search the Merchant we want to see through ID number, company name, business name, owner name, or affiliate number.

**\*Note:**

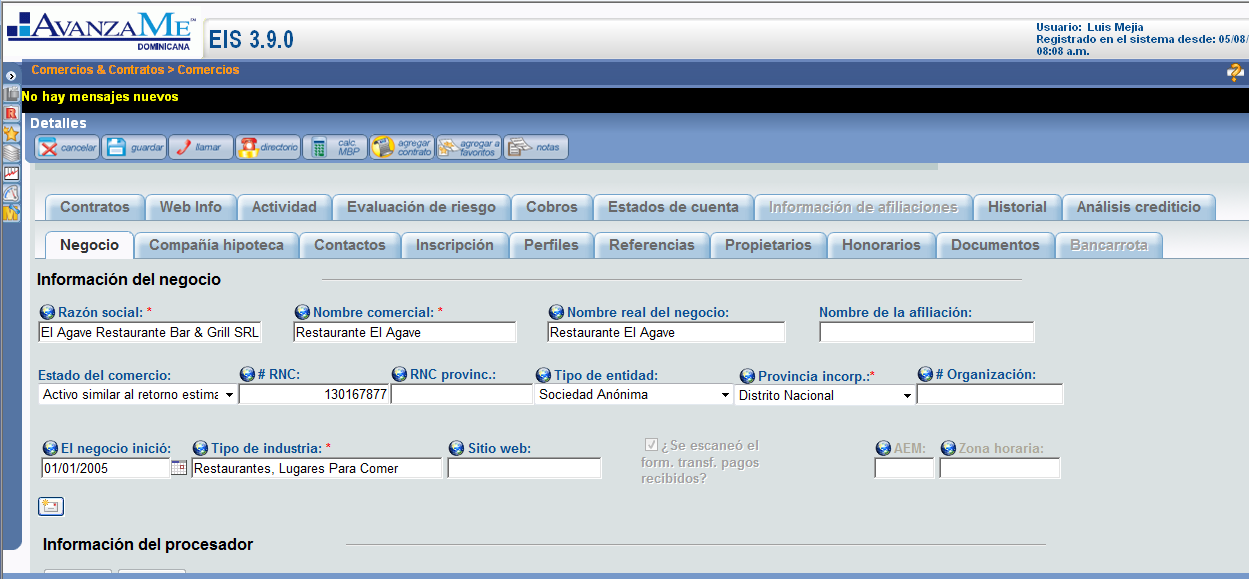
1. Status of merchant should be shown in screen.
2. Status contract should be shown to the user in screen.
3. We need to show on top any overcharge the contract might have (money we took after the contract was paid) so that we know what amount to refund the client.
4. Merchant screen will be accessible to all users.
5. All users cannot edit information in merchant screen.

### 2.1.2.4.1 Business Tab

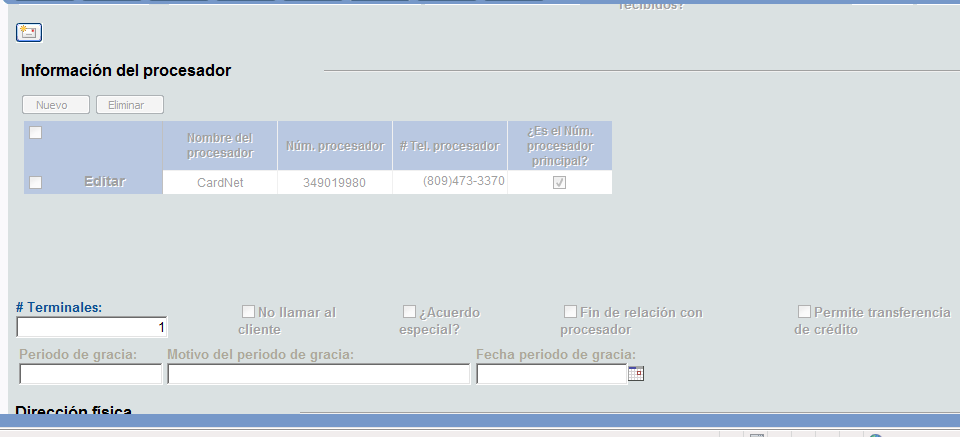
This is the main tab, and it will contain the following information:

* Name of the company
* Name of the business
* Business Status
* RNC
* Type of entity
* Address
* City
* Province
* Telephone Number
* Email
* Date the business started
* Date the business began accepting CC (This information should be pulled from the SPF file the Processor Companies send)
* Type of industry
* Processor Company and Affiliate Number (As many as the Merchant has)
* Legal Address, City and Telephone Number

Our current Merchant Screen looks like this:







### 2.1.2.4.2 Landlord Tab

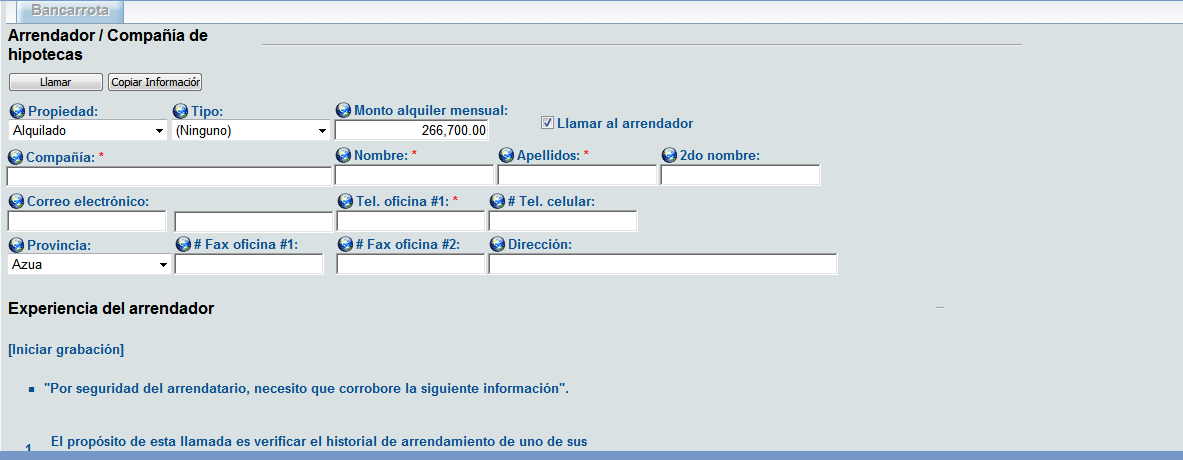
This is the tab that contains the information of the landlord, if it applies. If the business owns the local, then this tab should be empty.

It contains these fields:

* Type of property
* Landlord Company Name
* Landlord Name
* Telephone Number
* Cell Phone Number
* Email
* Address  
  City
* Monthly Rent Amount

Below this should be a copy of the script from the Landlord verification task and all the answers that were filled in (if the contract workflow has been completed).

Our current Landlord tab looks like this:



### 2.1.2.4.3 Contacts Tab

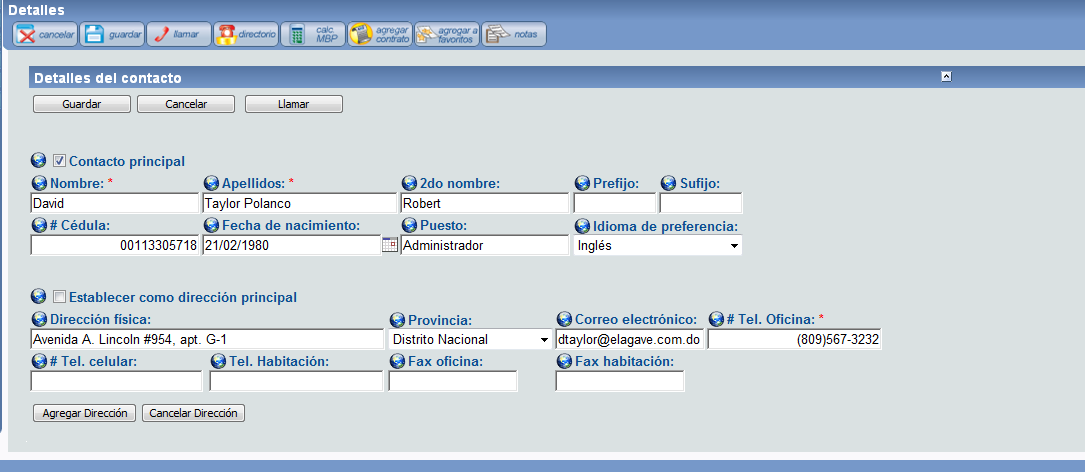
This is where we store all the information regarding contacts of a Merchant. A contact is any person within the business of the Merchant that we can call for information. It could be an owner or not.

It contains these fields:

* Name
* Last Name
* ID
* Date of Birth
* Position
* Address
* City
* Province
* Email
* Telephone Number
* Cell Phone Number

In this screen we should be able to add and delete Contacts.

Our current Contacts tab looks like this:



**\*Note:**

1. A contact is a person from the business of the merchant whom we can talk to if we need to. For example, let's say there's the Accountant of the business, if we talk to him, we create him as a contact (because he's not an owner) and then next time we need to talk to him we will have his information on the system.
2. Only authorized users should be able to add/edit contacts.
3. All fields are mandatory to create a contract.
4. History of deleted contacts should be logged.
5. A merchant can have multiple contacts.

### 2.1.2.4.4 Owners Tab

This tab is exactly like the Contacts tab except it needs to add a couple of fields:

* Name
* Last Name
* ID
* Date of Birth
* Position
* Address
* City
* Province
* Email
* Telephone Number
* Cell Phone Number
* Date became Owner
* Authorized to sign the contract (checkbox)

In the same way, we should be able to add and remove owners here.

### 2.1.2.4.5 Profiles

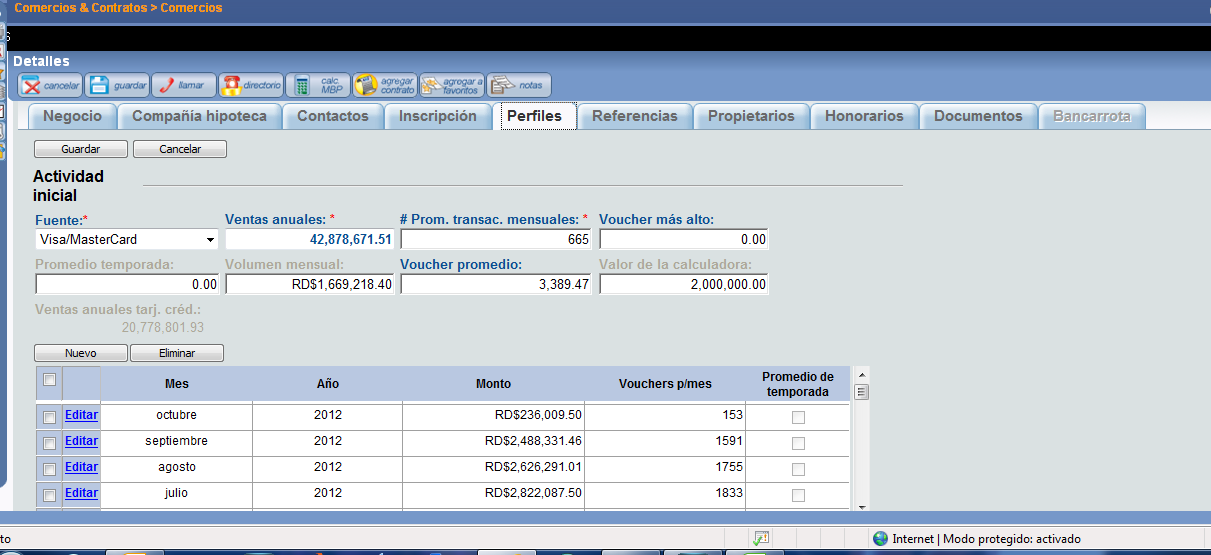
This is the tab that contains information about the credit card affiliates and volumes.

It should have the following fields:

* Gross Yearly Sales
* Average monthly transactions (number of tickets per month)
* Average Monthly Credit Card sales: If the Merchant has an active contract this field will be calculated with the payments. For example, if a merchant is paying 10,000 every month and the retention percentage of his contract is 10%, then his monthly sales must be 100,000 (10,000/10%).
* Average used in the Calculator: This is the average credit card sales that were used in the calculator when the offer was made (if the Merchant has a contract).
* Monthly credit card sales as calculated by our system (if the merchant has an active contract) detailed month by month.
* Monthly credit card sales received from the Processor companies detailed month by month.

In this screen it would be convenient to have a drop list in the top that would let us choose between VisaNet and CardNet or both, and then show just the information concerning the selected option. For example, if we choose VisaNet, the fields will be filled only with VisaNet information, if I select CardNet only with CardNet information and if I select both, with all the information we have.

Our current Profiles tab looks like this:



\*Note:

1. If both is selected as processors the values shown on screen will be sum of value for both processors.
2. Information should be shown for recent contract.

### 2.1.2.4.6 Documents Tab

This is a list of all the documents uploaded in the workflows. It should let us upload and delete documents right in this screen and it should show the date and user that uploaded every document.

The documents are:

* Legal Documents of the Company (PDF, JPG, GIF, PNG, DOC)
* Commercial Name Verification Screenshot (JPG, PDF, PNG, GIF)
* RNC Screenshot (JPG, PDF, PNG, GIF)
* ID or Passport (JPG, PDF, PNG, GIF)
* Lease Contract or Land Title (PDF, JPG, GIF, PNG, DOC)
* Null Check (JPG, PDF, PNG, GIF)
* Bank Statements (PDF, JPG, GIF, PNG, DOC, XLSX)
* Payment Receipt (PDF, JPG, GIF, PNG, DOC)
* Payment Agreement (PDF, JPG, GIF, PNG, DOC)

**\*Note:**

1. Documents for all contracts should be shown.
2. if it was uploaded before the workload, then it should be seen as already there in the work flow and we won't need to upload it again.
3. History of documents should be logged.

### 2.1.2.4.7 Activity Tab

This is the tab where we see all the payments the Merchant has done. You can organize this however you want, as long as we get to see in detail every financial activity in the Merchant, and organize it in a daily and monthly way.

It could look like this:



These fields should make more sense once we work through the Finances Module.

**\*Note:**

1. You will understand this once we go though the Finances Module, but there are different types of activity. Payments is the main one, but no the only one.

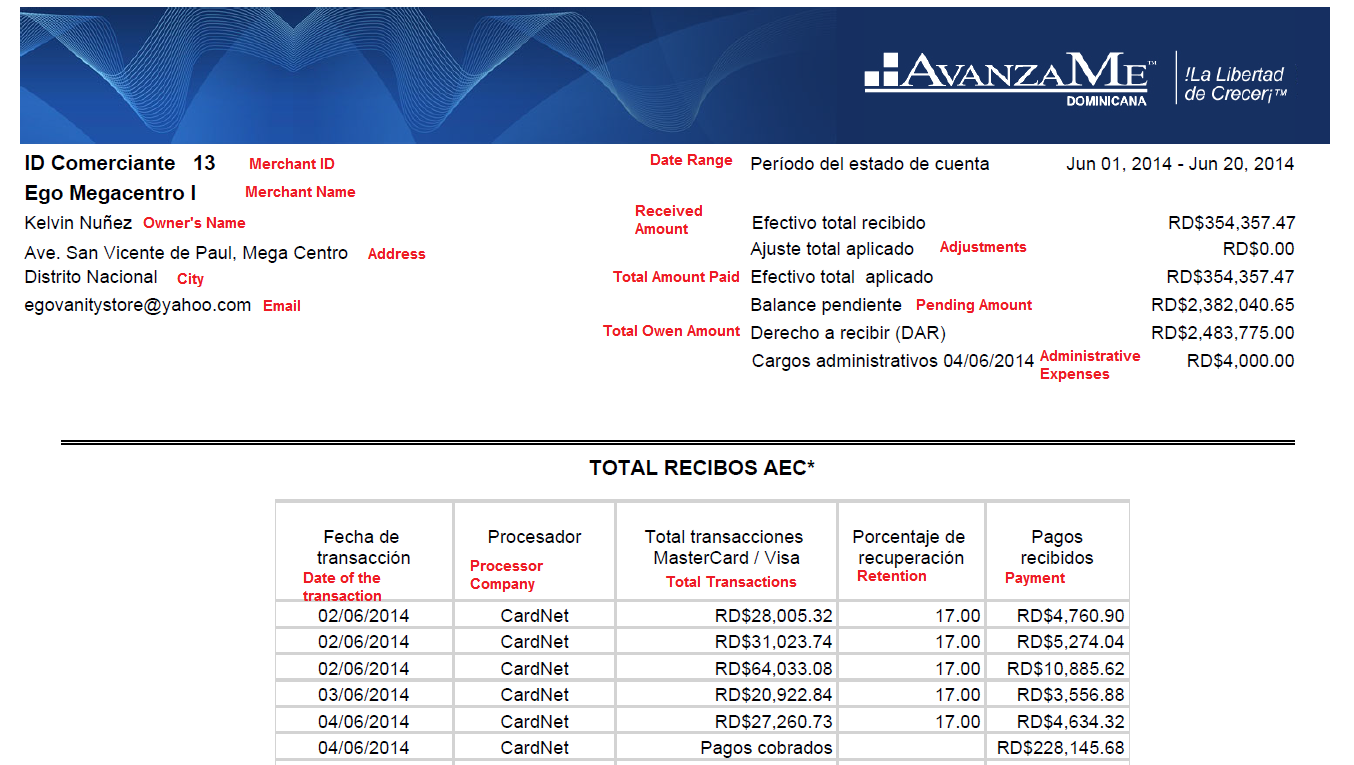
### 2.1.2.4.8 Statements Tab

This is the tab where we generate the statements for the client. We will provide you with the correct format, but the way it works is we select a range of dates and then we click one of two buttons. The first button will be “Generate Statements” which will bring up a pop up with a PDF file of the statements. The second button will be “Send Statements” which will automatically send the statements from the chosen range of dates to the Merchant’s email address along with a corresponding script that we will write.

Also in this tab is where we can generate the fiscal invoices, just the same method as the statements.

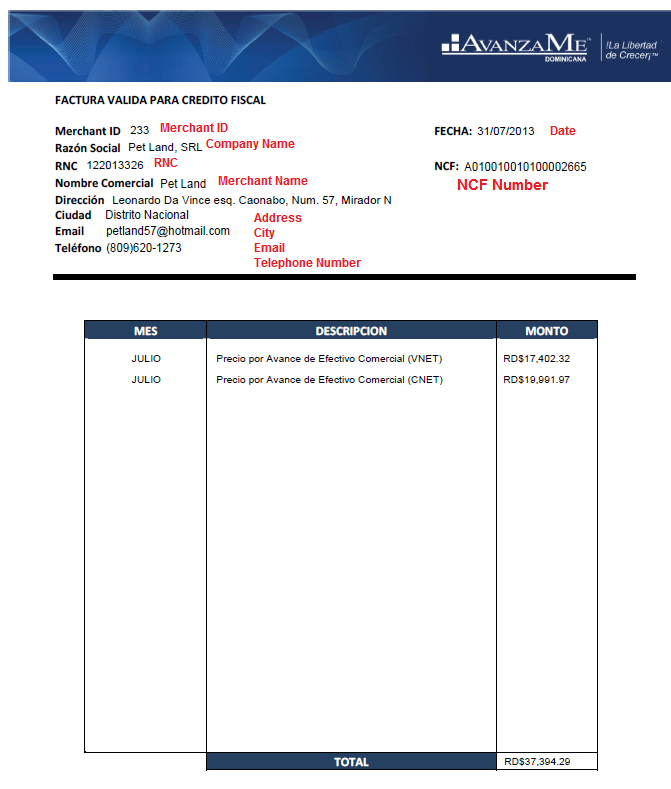
Apart from this manual generation, the system should automatically send the statements to the client on the first and 16th of every month. Also, the fiscal invoices (we will provide you with the correct format) will be sent along with the statements.

The statements look like this:



In red you’ll find the translation for every field. Some of these fields won’t make sense until we go through the Finances Module.

And this is how the Fiscal Invoices look:



The NCF number is a sequential number, once the system is running we will tell you from what number to start and from then on the system assigns the numbers sequentially.

Again, it won’t make sense until we go through the Finances Module.

**\*Note:**

1. By default the system should send the statements every 1st and 16th of every month, but maybe it could be useful to be able to change these dates for every individual merchant.
2. We should use changed values. In any case, we won't be able to change the parameters of the contract after it is funded but we could change any information from the merchant screen.

### 2.1.2.4.9 History Tab

This tab should show a log of all the changes in every field or status of the Merchant.

### 2.1.2.4.10 Risk Evaluation Tab

This tab should let us access all the generated Datacréditos as well as keep a list of every score the system has ever given the Merchant. This tab has all scoring evaluations made to this merchant, how they differ is through the contract number. Also it should have a stored all the credit reports pulled.

### 2.1.2.4.11 Contract Tab

This tab contains information about every contract the Merchant has ever had. It should show us a list of all the contracts with the parameters and the date they were funded. If we click a contract, it should popup a different screen with different tabs.

The list of contracts could look like this:



I believe you’re familiar with every field, except for “Real Time” which is the projected time the merchant will pay the contract in. For example, the offer was agreed to be paid in 8 months, so if he has to pay 175,000 in 8 months, then he would have to pay 21,875 per month to pay the contract in full in 8 months. But if he’s been paying 23,150 per month, then he will be projected to pay in 7.56 month (175,000 / 23,150). This is how the Real Time of the contract should be calculated, except that the calculations should be done based on what he has to pay *per day* and what he’s been really paying every day.

This new pop up should contain these tabs:

### 2.1.2.4.11.1 General Tab

In this tab is the general information of the contract.

The fields are:

* Contract Number
* MCA Amount
* Owed Amount
* Contract Status
* Funding Date
* Scanned contract and IOU (this button should show us a PDF of the contract and IOU)
* Retention Percentage (This field should be editable according to permissions and it should keep a log of changes)
* Administrative Expenses (This field should be editable according to permissions and it should keep a log of changes)

**\*Note:**

1. Status of contracts cannot be changed from contracts screen.
2. On the merchant no, of the contract nothing can be changed aside from the retrieval % (based on user permission).

### 2.1.2.4.12 Contract Activity tab

This is much like the previous activity tab, but for this specific contract.

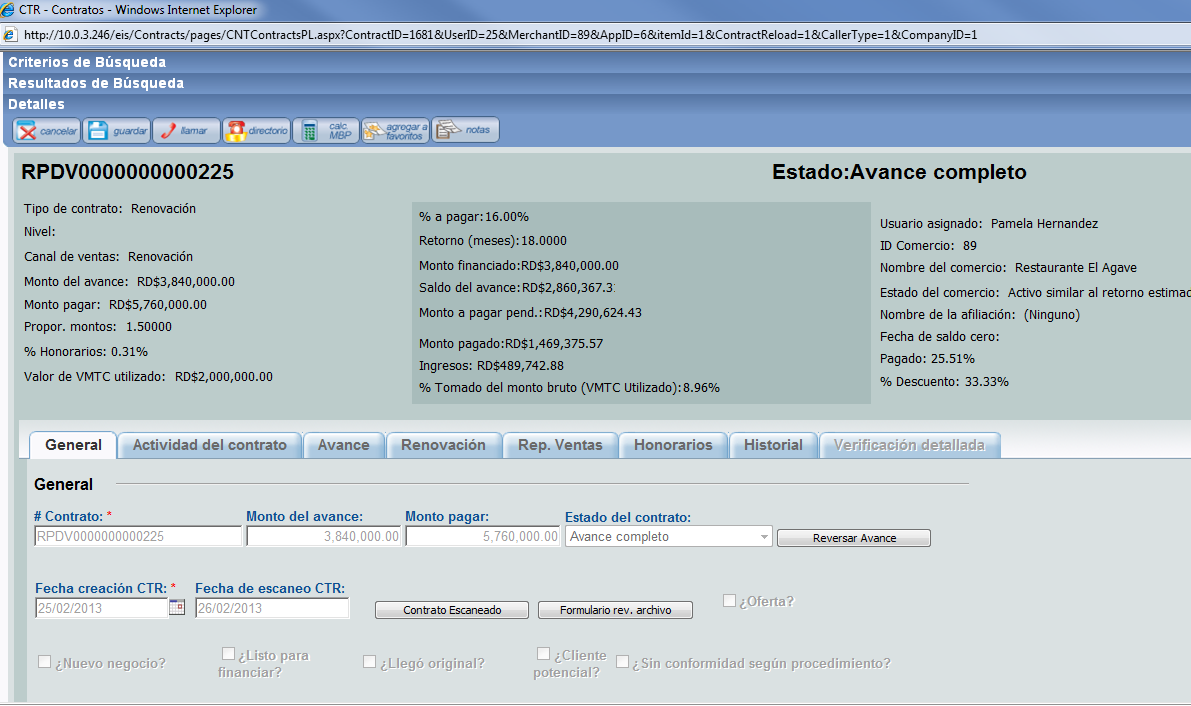
### 2.1.2.4.13 Sales Representative Tab

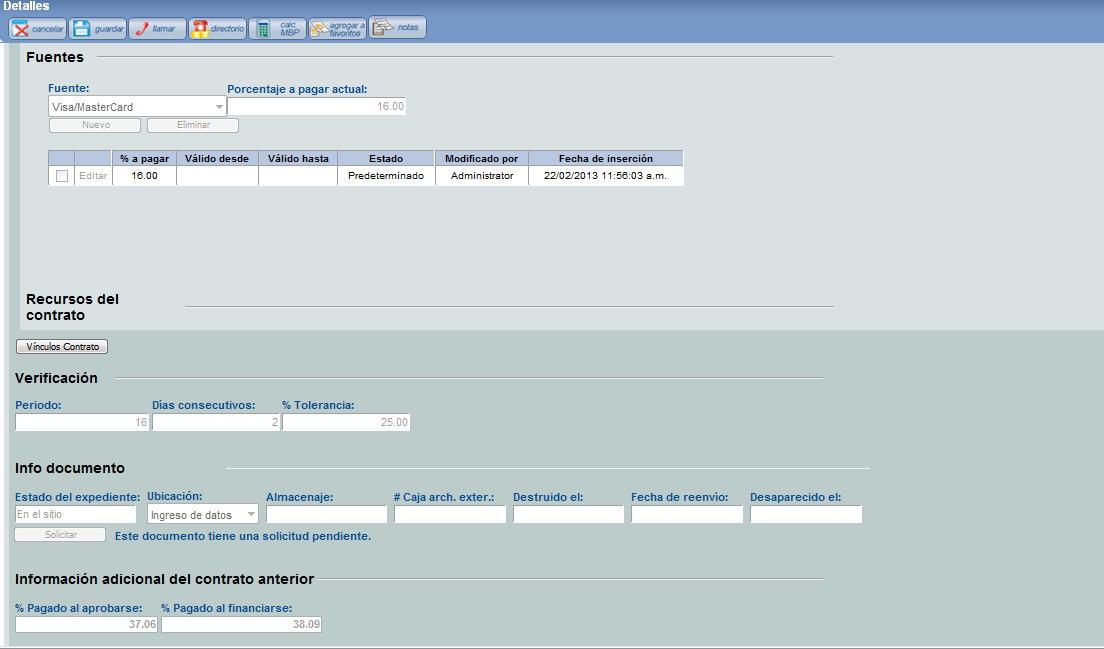
Here we should be able to add and remove Sales Reps associated with the contract.

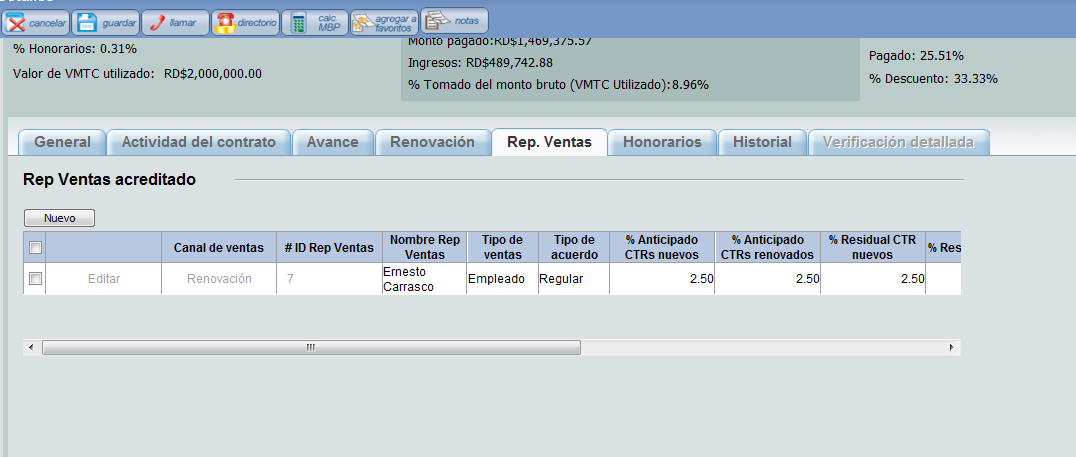
### 2.1.2.4.14 History Tab

This is a log of all the changes in every field.

Our Contract tab looks like this:







### 2.1.2.4.15 Collections Tab

In this tab we will see information if the Merchant has a contract (or has had a contract) in collections. It will show what amount entered collections, the status of the contract, the date it entered collection and other fields that we will go into detail when we complete the Collections workflow document.

# 2.1.2.4.16 Affiliation Tab

As we have specified before, a company or owner might own several different business which are created as separate Merchants in our system. But we still need a way to link them, so an affiliation is a link that exists between all the Merchants that have the same RNC number (this means they are owned by the same company) or the same Owner ID (which means they are owned by the same person).

This button will pull the Merchant’s affiliation screen, which could look something like this:



As you can see, there are two tabs. One tab is for the merchants with the same RNC and the other is for the Merchants with the same Owner.

There should be two boxes of information, one for all the merchants with the same RNC or Owner as the Merchant we are currently in that have Active Contracts and another box for all the affiliations that don’t have Active Contracts.

For the inactive contracts we need a button that will let us request the CC volumes for the merchants to the Processor Company. If we request the volumes, this list should show a Volumes Status that will be marked as pending until the Processor Company sends the requested volumes.

# 2.1.2.4.17 Credit Report Tab

In this screen we should see all the information from the Datacredito report, but imported as a proinformation per fields in our system. We will try to get Datecredito to help us map all the fields to the raw file (the XML file) you get from Datacredito so that you know where to get everything from.

The first thing this should have is a dropdown list of all the Credit Reports that have been generate, per contract. For example, a dropdown list with Contract 1, Contract 2 and Contract 3. If we switch between contracts it will show the from the Datacredito that was pulled for that specific contract.

We need to divide this tab in three: One for the Business’ Datacredito, one for the Owner Datacredito, and one for the sum of both. All three sections will have the same fields. Every section should have on the top the Xcore (The information for credit Score in the file is: "Xcore\_PD12M\_ALL\_PC\_NC\_Global":{"Xcore":"") for the selected contract and then it will have several subsections.

Every section will be subdivided like this:

### Total Credit

* Number of loans
* Approved Amount
* Owed Amount
* Monthly Payments
* Late Amount
* Amount in legal
* Late Index: This is the Late Amount divided by owed amount.
* Debt Index: This is the Monthly Payments amount divided by the total monthly sales of the merchant.

### Loans

* Number of loans
* Approved Amount
* Owed Amount
* Monthly Payments
* Late Amount
* Amount in legal

### Credit Cards

* Number of loans
* Approved Amount
* Owed Amount
* Monthly Payments
* Late Amount
* Amount in legal

### Others

* Number of loans
* Approved Amount
* Owed Amount
* Monthly Payments
* Late Amount
* Amount in legal

The Total Credits subsection will be the sum of all the other subsections.

It could look like this:

### 

### 2.1.2.4.18 Final Notes

The Merchant screen needs to have the following buttons:

* Save Button: To save any change made to any field.
* Cancel Button: To erase any change made.
* Call Button: With a list of all the stored telephone numbers in the Merchant, including owners, contacts, landlord and business numbers.
* Notes Button: A button to access the Notes application to write and read notes.

Once we enter a Merchant screen, the first seen we need to see on top is an information box on top with the essential information:

* Merchant ID
* Merchant Name
* Amount left to pay of the current contract
* Date of the last activity (this is the date of the last payment we received)
* Merchant Status

## 2.1.2.5 Permissions controls:

User permissions can be controlled by users with admin rights from user section. Details of pages is as below:

### 2.1.2.5.1 Permissions and Roles

The way we want this to work is, we create different user roles and assign permission to these roles to use different parts of the system, and then we assign the roles to the different users.

We need to have two types of permissions: Reading Rights and Modification Rights. Reading rights gives the user that has the role assigned to see and read the part of the system that has been assigned to the role with reading rights. For example, if the “Operations Officer” role has been assigned reading rights for the Finances Screen, any user with this role will be able to see the Finances Screen but won’t be able to make any modifications. Any user with a role that has Modification Rights for the Finances Screen will be able to both see and modify the information in the Finances Screen.

The only pre-created user role should be that of Administrator, that will have both Reading and Modifications rights to every part of the system. Any other role we want, we need to be able to create and then assign Reading, Modification, or No Rights to every module of the system individually (Workflows, Merchant Screen, Finances, Reports). Once we assign rights to a module, it should assign by default rights to everything IN the module, but we need to be able to select certain parts of the module with different rights. For example, in the Reports screen we want certain roles to have Reading Rights for some reports but No Rights for other reports.

It could be something like this:



To keep in mind that once every single component of the system is constructed it must be included in the permissions screen (calculator, scoring, etc).

### 2.1.2.5.2 User Profiles

The user profiles should contain information about the user such as Name, Last Name, and Date they joined the company, role, etc. We need to be able to assign any role to any user.

For those users that have roles with Rights to Workflows, we need a sort of timetable that will let us select the day and time we want workflows task to be assigned to that user. For example, let’s say User A and User B have rights to the Contract workflow, and in the time table in the User A’s profile we select Mondays, Wednesdays and Fridays from 8 AM to 5 PM for him to be able to work in the Contract Workflow, he will only be assigned tasks inside this time windows in those specific days, and User B will be assigned no tasks in these days unless we select the same days in the time table.

The time table could look like this:



Here we need to have an option to give leave permissions to some users. For example we can select a range of dates when the user won’t be in the office (like a vacation) and it will deactivate the user for that time without have to modify the time table.

**\*Note:**

1. Roles cannot be changed unless all tasks are completed or reassigned to another user.

## 2.1.2.6 The Collections Work flow

The Collections work flow will work in two branches. The first branch, which will be a sort of prevention work flow, will show us a list of all contracts that are paying 30% (or more) slower than expected and all contracts funded more than 30 days ago that are paying 20% (or more) slower than expected. This first branch will who us the information of the Merchant and the Contract and, if we click one of the Merchants, it will lead us to the Collections screen for that Merchant (a screen we will specify in the second branch).

It could be something like this:



The second branch, which will be the actual Collections workflow, will show a list of all the Merchants that have 4 or more days without registering any payment for their active contract by default, but there should be an option to select merchants that haven’t been paying for X amount of days. The merchants will be ordered by the amount of days without processing in descending order. In this list, if there’s a Merchant that has being showing for a monthly average of 2 times or more, it should be marked red as a warning. The monthly average will be calculated by dividing the number of times the merchant has showed up in the Collections workflow by the number of months the contract has been active for. For example, if Merchant A has an active contract that was funded 4 months ago and it has showed up in the Collections workflow 8 times, then it shows up an average of 2 times a month, and it should be marked red.

In this list we should also see the clients that have missed payments from a Payment Agreement.

It could look like this:



Once we click a merchant, it will take us to the Merchant’s Collection screen.

### 2.1.2.6.1 The Collections Screen

In this screen we will look at the information and decide what to do with the Merchant. This screen should have all the information we need plus a series of buttons that will let us take action.

It could look like this:



### 2.1.2.6.2 Datacrédito

If we click this button the system should ask us if we want to pull the latest generated credit report or if we want to generate a new one, and then show us the credit report in a popup.

### 2.1.2.6.3 Affiliations

As we have specified before, a company or owner might own several different business which are created as separate Merchants in our system. But we still need a way to link them, so an affiliation is a link that exists between all the Merchants that have the same RNC number (this means they are owned by the same company) or the same Owner ID (which means they are owned by the same person).

This button will pull the Merchant’s affiliation screen, which could look something like this:



As you can see, there are two tabs. One tab is for the merchants with the same RNC and the other is for the Merchants with the same Owner.

There should be two boxes of information, one for all the merchants with the same RNC or Owner as the Merchant we are currently in that have Active Contracts and another box for all the affiliations that don’t have Active Contracts.

For the inactive contracts we need a button that will let us request the CC volumes for the merchants to the Processor Company. If we request the volumes, this list should show a Volumes Status that will be marked as pending until the Processor Company sends the requested volumes.

### 2.1.2.6.4 Documents

This button should bring up the regular documents screen and let us upload or see any document we need and then let us return to the collections workflow.

### 2.1.2.6.5 Landlord

This button should bring up the regular landlord screen so that we can see the information of the landlord and the script of the landlord call as was filled out during the contract work flow.

### 2.1.2.6.6 Payment Agreement

This section will be used when we agree with the client to a payment plan where the client will pay us a fixed amount every agreed period until the owed amount is met.



In this screen the user will upload the signed physical agreement document, which will be stored along with all the other documents from the merchant.

The date of the agreement will be the date the agreement was signed and the start date and end date are the dates of the first and last payment, respectively. The start date will be set by the user, but the end date will, by default, depend on the contract. If the contract has been active for two months at the moment of the start date, and the offer had an estimated turn of 8 months, then the end date will be set by default in a date that will allow the contract to be paid within the 8 months. In the case of the example it would be 6 months from the start date. Although this will be set like this by default, we should be able to modify it.

The interval is the amount of the days between each payment. For example, if we set the interval to 15, then the client will pays us every 15 days.

The amount of the payments will be decided by dividing the pending amount of the contract by the number of periods. In the case of the example, if we have 6 months between the start and date, and payments every 15 days, then we have 12 periods ((6 \* 30) / 15). If this contract has a pending amount of $60,000, then the payments should be $5,000 each for 12 periods.

The date of the first payment will be the start date and the following dates will be the previous date plus the number of days of interval. For example, if the start date is January 1st and the interval is 15 days, then the first payment will be on January 1st, the second payment on January 16th, the third payment on February 1st and so on.

Every payment due should be detailed from start to finish, and every payment should have the status of the payment. This will make more sense once we go over the Finances module but if this contract receives the payment on the set date, it will be pulled out of the Collections workflow and return to normal. But if the merchant misses one payment, or does not pay the full amount of the period, it should go back into the Collections workflow.

If the client does not pay the full amount in a certain period, the remaining amount should be added up to the next period. For example, if a client has a payment of $5,000 due February 1st, but come the day he only pays $4,000, then the next payment, say it’s supposed to be on February 16th, should be $6,000, which is the corresponding $5,000 plus the remaining $1,000 from the previous period. The status should be “Paid”, “Incomplete” and “Missed Payment”.

Every payment should detail the date the payment was received. If the client has missed payments and pays on a later date, the system will take the money and apply it to the earliest missed payment and the date of the payment will be the date the client actually pays.

### 2.1.2.6.7 Legal

We will use this screen if we assign the client to an external legal firm. In this screen we should store the information of who has the case and what documents they have.

It could look like this:



It should have the information of the Lawyer, and below, we should see a list of all the documents uploaded for this particular merchant, with a checkbox mark beside the document that we will mark if the external legal firm is in possession of said document. For example, if we handed the Leasing Agreement, Corporative Documents and Contract for this merchant to an external legal firm, we would mark the corresponding checkbox of these documents in this screen.

The Lawyer will be selected through a drop list that will have all the lawyers created in the system.

There should be a Create New Lawyer button that will let use create and save new lawyers.

This button should bring up a screen like this:



As you can see it could show us a list of all the created lawyers and let us edit or delete them, as well as let us save new lawyers to the system.

### 2.1.2.6.8 Generate Letter

With this button we will access a letter that we will send to the Merchant as a physical letter or through email. We will give you a template of the letter and tell you what information the system should populate automatically.

### 2.1.2.6.9 Activity

Here the user will mark what he or she had done in the Collections screen and write a note. It should be a drop list for the user to choose with the following options:

* Contacted by telephone
* Contacted by email
* Investigative visit
* Letter
* Legal warning
* Contacted landlord
* Notify Payment
* Payment Agreement
* Other

In order for the system to be able to save an activity, some conditions have to be met. For all activities a note should be written and saved. In the case of “Notify Payment” the system should check if the Payment Receipt document has been uploaded before it lets us save this activity. For “Payment Agreement” it shouldn’t let us save unless a Payment Agreement has been activated through the Payment Agreement button and the Payment Agreement document has been uploaded.

**\*Note:** For collections:

1. Real turn for contract should be calculated on real time basis.
2. Contract will be marked as “Writtten Off” after 180 days of inactivity automatically.
3. there should be a screen that’s called Workflows and there should appear all the individual workflows for the corresponding agent to enter their workflow.
4. If a merchant is in collections and we get that mercahnt to process once again (thru credit card processors not payment agreements) the system should remove him from the collections status after we have recieved at leat 10 payments (thru the processor). If the merchant is in a payment agreement it will always stay in collections as a merchant status. If he is paying the agreement as expeted the merchant will not appear in the collections workflow, but if one payment is missed than it should show up as a reminder to the agent to call this merchant and request payment. But a merchant in a payment agreement will always be in collections, only merchants that pop out of collections are the once that star processing again.
5. **Questions:** If a merchant has made payment agreement to pay $5000 after how much time it will be pulled out of collecctions workflow?

**Answer:** After the first payment, if the merchant pays on time. Then it will remain outside the collections workflow as long as the merchant keeps making the payments on time. This is regarding the collections workflow only, the status of the merchant will always be in Collections. If it has a payment agreement, yes. We need to check that every payment is made on time, otherwise it should go right back to the collections workflow.

1. if the merchant process thru the credit card it should be included in his payments aswell. This way we have different ways of getting our money back.
2. Processor activities will not be removed until payment agreement is signed.

## 2.1.2.7 Finances

This module is the one that will gather all the payments and activity and will assign them to the corresponding merchants.

There are several kinds of activities:

* **Regular Payments:** These are the payments we receive through Credit Card transactions.
* **Processor Adjustments:** These are payments we apply in the system in cases when the system does not capture a regular payment that was made by the merchant.
* Cash/Check/Transference Payments: These are payments the merchant makes, either because they have a Payment Agreement or they want to pay the loan in cash to end the contract sooner.
* **Refund:** This is money we give the merchant in cases when our system takes more money than it should at the end of the loan (because the Processor company didn’t mark the loan as paid in full or whatever reason).
* **Renewal Repurchase:** Since we can renew a contract as soon as it has paid 60% of the total owed amount, most of the times the previous contract has a pending amount once we renew it, so we use part of the new MCA to pay off the pending amount of the previous contract. This is a repurchase.
* **Credit Transference:** This is transference of a certain amount that has been paid in one merchant to another merchant. For example if Merchant A paid $100 of his contract, we can transfer that credit to Merchant B. It should be possible to transfer all or a part of payment from merchant A to merchant B.
* **Cash Processing:** There are certain merchant that have special agreements with us and don’t make regular payments through Credit Card transactions. Instead, they pay us in cash the same amount we expected to receive through Credit Cards. This is not the same as a regular Cash Payment, because the Cash Processing is done within the parameters of the offer. The way this should work is, at the time that we add the Processor Company affiliate numbers to the merchants, we should have an option to mark the merchant as Cash Processing, even if we still add the affiliate number. For example, let’s say Merchant A has an affiliate number for VisaNet and one for CardNet, and we have both in the system. If we have a special agreement with the client we can mark the client as Cash processing and the system will stop expecting payments from Credit Cards and will start expecting Cash payments. If at any time we want to stop receiving cash and start receiving Credit Card transactions, we would just need to mark the affiliate numbers as active.

Cash processing merchant will be paying back a certain amount after X days. Special contracts will be generated for Cash processing merchants.

Collections check for these merchants will be 30 days of inactivity.

There should be a finances screen that will let us add these activities (except for the Regular payments, which can only come from Credit Card transactions). In this screen, we would click a button called Add Activity, that will ask us what kind of activity we want to add and will show us a drop list of the available options. We need to specify the ID of the merchant, the amount, and the date of the activity. In the case of Processor Adjustments, we need to specify the affiliate number that processed that specific payment.

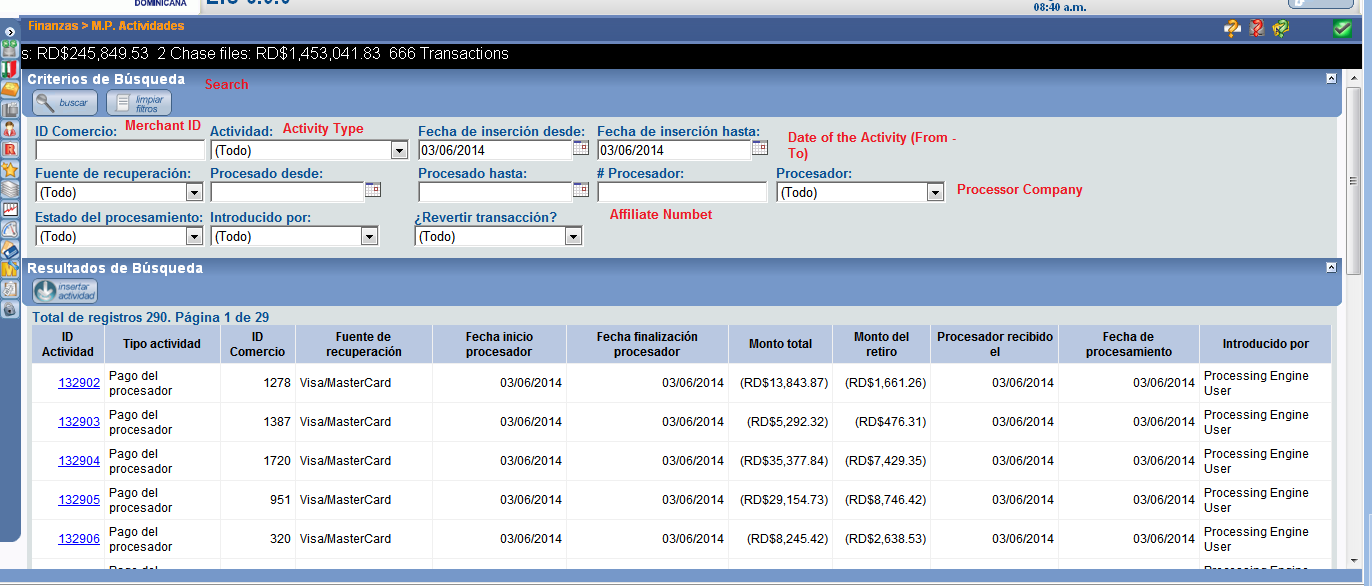
Once an activity has been added, the system needs to automatically save a note to the merchant with the type of activity, the amount and the date of the activity.

This is the screen where we add the activities:



In addition to this, in this module we should be able to see a list of all the activities, by doing a search with specific parameters of the payments we want to see. This list should be exportable to Excel.

This is how the activities screen looks now:



These are the parameters we need to see in the list:

* Activity ID
* Activity type
* Merchant ID
* Processor Company
* Date of the activity
* Total Amount
* Price Amount
* Capital Amount
* Processing date
* Added by (User)

So, in the end, this screen should look much like the Activity tab in the Merchant screen:



Except that we the first three columns should be the Activity ID, Type of activity and Merchant ID.

## 2.1.2.8 Transfer Files

Much like the request and answer of the Credit Card volumes, we request and receive the transactions for the funded affiliate numbers. Every day our system sends a WSF files with all the active Merchants, the retention percentage and the pending amount to the Processor Companies, who will, in turn, answer with a PAF files with the transactions for every Merchant. Is from this PAF file that our system pulls the Credit Card transactions.

## 2.1.2.9 Reports

New system will have a section for reporting. It should be able to generate reports on demand and automatically. Reports can be generated after X days periodically. Here is the list of reports:

## 2.2. User Management

New System provides a way to manage users. User management includes creating, editing and deleting users. There will be different roles in the system. User rights can change in system. Permissions of user roles can also change.

## 2.3. Data Output

## 2.3.1 Processor activities

New system will exchange data with processor servers to add new merchants for payments.

## 2.3.1.1 File Transfer

* Our preferred method for file transfer is SFTP (under special circumstances we might receive –or send- files via e-mail attachment).
* Our process to download files from the processor and upload up to four files in return is automated. This process is set to happen on a daily basis as agreed with each processor.
* The computer accepting the SFTP connection on our processor side should have two folders: CAdownload and CAupload.

## 2.3.1.2 Upload

These files are generated by our system and will be available to the processor for upload.

### 2.3.1.2.1 Withdrawal Setup File (Upload)

Generated by Avanzame and delivered to the processor (also known as [WSF files](#_Withdrawal Setup File (WSF))). This file includes information to let the processor know when to start or stop the withdrawal process on specific merchants. It also includes running balances at a merchant level and the actual Specified Percentage rate for withdrawals with respect to the merchant.

### 2.3.1.2.2 Contract Overview File (Upload)

This file is the response file generated by Avanzame. It is delivered to the processor in order to give the processor the option of keeping a running balance of specific contracts signed with the merchants. This file is also known as a [COF file](#_Contract Overview File (cof)).

## 2.3.1.2.3 Status report File (UPLOAd)

This file is generated by Avanzame to notify the processor of the status of applications that are in process. This file is optional (also known as [STS file](#_Status report File (sts))).

## 2.3.1.3 Download

The processor activity and adjustment files will be picked up from the CAdownload folder. This file will be processed and archived by our system. All encountered errors and exceptions are logged in and researched by our team.

### 2.3.1.3.1 Response Setup File (Download)

Generated by the Processor to acknowledge the Withdrawal Setup File and indicate if the “Merchant setup process” was successful or not.

### 2.3.1.3.2 Processor Activity Files (Download)

This file is generated by the processor and delivered to Avanzame on a daily basis (also known as [PAF files](#_Processor Activity File (PAF))). This file will have all merchant activity for the day and will be used by Avanzame to keep our system up to date. This file is mission critical to AdvanceMe, Inc.’s business.

### General file exchange work flow



(\*) Optional files are not shown on the exchange work flow diagram.

## 2.3.1.4 General file SCHEDULE information

The file exchange will be done, using the following schedule:

|  |  |
| --- | --- |
| AdvanceMe Inc, would pick up, on a daily basis, the following file … | |
| File Type | Schedule |
| Processor Activity Files ([PAF Files](#_Processor Activity File (PAF))) | 11:30 AM EST |
| Response Setup Files ([RSF Files](#__x000F_Response Setup File (RSF))) | 11:30 AM EST |

|  |  |
| --- | --- |
| Each Processor would pick up, on a daily basis, the following files … | |
| File Type | Schedule |
| Withdrawal Setup Files ([WSF Files](#_Withdrawal Setup File (WSF))) | 5:30 PM EST |
| Contract Overview Files ([COF Files](#_Contract Overview File (cof))) - Optional | 5:30 PM EST |
| Status Report Files ([STS Files](#_Status report File (sts))) - Optional | 5:30 PM EST |

As stated, the file exchange would be done on a daily basis, including Saturdays and Sundays (including holidays).

Changes to this outline schedule of procedures must be agreed upon in writing or electronic communication by the Processor and AdvanceMe, Inc.

### 2.3.1.4.1 General file information

All files are CSV:

* Comma delimited
* Dates have to have to format YYYYMMDD
* The filename is composed of:
* Processor code: char (2)
* Processed date: char (6): YYMMDD
* Specific extension: char(3) according to file type. e.g. CS010514**.PAF**
* All files have a header record and a trailer record (this would be explained in detail with each file type).

## 2.3.2 Reports

New system will have a section for reporting. It should be able to generate reports on demand and automatically. Reports can be generated after X days periodically. Here is the list of reports:

1. [Contracts](#_Contracts Report:)
2. [Database report](#_Database report:)
3. [Monthly MCA](#_Monthly MCA:)
4. [Notes](#_Notes:)
5. [Pre-qualifications](#_Prequalifications:)
6. [Turn Analysis](#_Turn Analysis:)

## 2.4. Data Management

## 2.5. Coding Requirements

## Appendix

# *File Detailed Format and Information*

# *Withdrawal Setup File (WSF)*

**Extension:** WSF

**Creator:** AdvanceMe, Inc.

**Purpose:** To notify the processor for which merchants withdrawals needs to start or stop. Also provides running balances at merchant level and the actual Specified Percentage rate at which withdrawals for the merchant should take place.

**Header record:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** |  |
| Header Constant | Varchar(6) | HEADER | Required |
| AMI Processed date | Varchar(8) | Date that AMI processed the file. | Required |
| AMI Processor Code | Varchar(9) | Processor code as existing on AMI System | Required |

**File Detailed Structure:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** |  |
| AMI Processed date | Varchar(8) | Date that AMI. processed the record | Required |
| AMI Processor Code | Varchar(9) | Processor code as existing on AMI System | Required |
| AMI Merchant Number | Varchar (9) | MerchantID as existing on AMI System. | Required |
| Processor Merchant Number | Varchar (25) | MerchantID as assigned by the processor. | Required |
| Balance | Numeric(14,2) | Total amount purchased by AMI from the Merchant. This includes Renewal contracts in process | Required |
| (Specified Percentage) Rate | Numeric(14,2) | Percent to be withheld. A Specified Percentage of 0.0 indicates the Merchant has completed its contract obligations. | Required |

**Trailer record:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** | |  |
| Trailer Constant | Varchar(7) | TRAILER | | Required |
| AMI Processed date | Varchar(8) | Date that AMI processed the file | | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | | Required |
| Sum of Balances | Numeric(14,2) | | Total amount of Balances of unremitted purchased receivables included on the detail of the file. | Required |
| Number of Rows | Numeric(5) | | Total number of rows included on the file (it does not include the header and trailer rows) | Required |

**Special Considerations:**

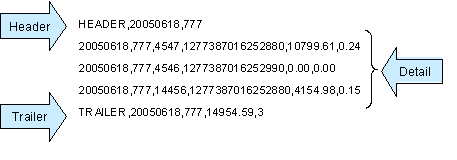
1. As a general rule, this file will include active Merchants only.
2. The Specified Percentage rate can change on a daily basis for any given merchant
3. Balances of unremitted purchased receivables will change daily and might increase or decrease depending on new contracts being created for the merchant.
4. If a Merchant has completely satisfied its contract obligations, the day the balance of unremitted purchased receivables becomes $0, a record will be sent on this file, with the Balance of unremitted purchased receivables in 0 and the Specified Percentage Rate in 0 to indicate so. Subsequent files will no longer include that merchant until new contracts are issued.
5. New Merchants could be added to this file daily.
6. If Avanzame does not submit a file, the processor should revert to the last received file.

**Example:**

For this example we could consider the merchant pool of “Processor StarFleet” (processor initials “ST” and AMI Processor code 777). Let’s say this processor has the following Merchant Agreements:

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Case # 1** | **Case # 2** | **Case # 3** |
| AMI Merchant Number | 4547 | 4546 | 14456 |
| Processor Merchant Number | 1277387016252885 | 1277387016252999 | 1277387016252888 |
| Balance | $10799.61 | $0 | $4154.98 |
| (Specified Percentage) Rate | 24% | 25% | 15% |
| Observations |  | This merchant just satisfied all of its obligations under its agreement and there are no other contracts, or the merchant doesn’t belong to an affiliation that has outstanding balances of unremitted purchased receivables. |  |

For this example, the processing date would be June 18th, 2005. This would mean that AMI would create a file called ST050618.WSF, and would have the following data:



Please note the "zero record" (zero Balance of unremitted purchased receivables, zero Specified Percentage Rate) for merchant 4546 as an indication that Specified Percentage rates must stop on that merchant.

## 

## Contract Overview File (cof)

**Extension:** COF

**Creator:** AdvanceMe, Inc.

**Purpose:** To notify the processor of running balances for individual contracts signed with AMI Merchants.

**Header record:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** |  |
| Header Constant | Varchar(6) | HEADER | Required |
| AMI Processed date | Varchar(8) | Date that AMI processed the file | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | Required |

**File Detailed Structure:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** |  |
| AMI Processed date | Varchar(8) | Date that AMI processed the record | Required |
| AMI Processor Code | Varchar(9) | Processor code as existing on AMI System | Required |
| AMI Merchant Number | Varchar(9) | MerchantID as existing on AMI System. | Required |
| Processor Merchant Number | Varchar(25) | MerchantID as assigned by the processor. | Required |
| Contract | Numeric(14) | Contract Number existing on AMI System | Required |
| Balance | Numeric(14,2) | Total amount purchased by AMI from the Merchant. This includes Renewal contracts in process | Required |
| (Specified Percentage) Rate | Numeric(14,2) | Percent to be withheld. A Percentage of 0.0 indicates the Merchant has completed its contract obligations. | Required |
| FundedFlag | Varchar(1) | Indicates whether the contract has already been funded by AMI turning it into an Active Contract | Required |

**Trailer record:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** | |  |
| Trailer Constant | Varchar(7) | TRAILER | | Required |
| AMI Processed date | Varchar(8) | Date that AMI processed the file | | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | | Required |
| Sum of Balances | Numeric(14,2) | | Total amount of Balances of unremitted purchased receivables included on the detail of the file. | Required |
| Number of Rows | Numeric(5) | | Total number of rows included on the file (it does not include the header and trailer rows) | Required |

**Special Considerations:**

1. As a general rule, this file is to include Funded Contracts and those in process of being funded (passed approval review for new contracts and all for renewal contracts).
2. If a Contract is completely closed out, the day the balance becomes $0, a record will be sent on this file, with the Balance in 0 and the (Specified Percentage) Rate in 0 to indicate so. Subsequent files will no longer include that contract.
3. Contracts might go in and out of this file if they are newly created or canceled/declined.

**Example:**

For this example we could consider the merchant pool of “Processor StarFleet” (processor initials “ST” and AMI Processor code 777). Let’s say this processor has the following Merchant Agreements:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Case # 1** | **Case # 2** | **Case # 3** |
| AMI Merchant Number | 4547 | 4546 | 14456 |
| Processor Merchant Number | 1277387016252885 | 1277387016252999 | 1277387016252888 |
| Balance | $10799.61 | $0 | $4154.98 |
| (Specified Percentage) Rate | 24% | 25% | 15% |
| Observations |  | This merchant just satisfied all of its obligations under its agreement and there are no other contracts, or the merchant doesn’t belong to an affiliation that has outstanding balances of unremitted purchased receivables. |  |

For this example, the processing date would be June 18th, 2005. This would mean that AMI would create a file called ST050618.COF, and would have the following data:



## Status report File (sts)

**Extension:** STS

**Creator:** AdvanceMe Inc.

**Purpose:** To notify the processor of the status of applications that are in process. Accounts will remain on the report for 30 days following any status that would assume the account has finished the application process. This file only applies if the processor is also submitting contract applications to Avanzame and is created only if specifically requested.

**Header record:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** |  |
| Header Constant | Varchar(8) | HEADER | Required |
| AMI Processed date | Varchar(8) | Date that AMI processed the file | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | Required |

**File Detailed Structure:**

| **Name** | **Data Type** | **Description** |  |
| --- | --- | --- | --- |
| AMI Processor Code | Varchar(9) | Processor code as existing on AMI System | Required |
| AMI Merchant Number | Varchar(9) | MerchantID as existing on AMI System. | Required |
| Processor Merchant Number | Varchar(25) | Merchant Number as assigned by the processor. | Required |
| Application Received Date | Varchar(8) | The date AMI received the application | Required |
| Application Delayed Date | Varchar(8) | The date an application gets Delayed for additional documentation | Required – if pended, blank if not. |
| Pending Items | Text | Free text of all missing documentation | Required – if pended, blank if not. |
| Application Out Of Delay Date | Varchar(8) | The date the missing documentation requirements have been met. | Required – if Pending Items have been satisfied, blank if not. |
| Application Approved Date | Varchar(8) | The date the application is approved for funding | Required – if approved, blank if not. |
| Application Declined Date | Varchar(8) | The date the application gets declined | Required – if declned, blank if not. |
| Funding Date | Varchar(8) | Date the account gets funded | Required – if funded, blank if not. |
| Cash Advance Amt | Numeric(14,2) | The amount of the initial cash advance to the customer | Required - if the account is funded |
| Contract Number | Varrchar(20) | This corresponds to the Contract Number as existing on the AMI system. | Required - if the account is funded |

**Trailer record:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** | |  |
| Trailer Constant | Varchar(7) | TRAILER | | Required |
| AMI Processed Date | Varchar(8) | Date that AMI processed the file | | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | | Required |
| Number of Rows | Numeric(5) | | Total number of rows included on the file (it does not include the header and trailer rows) | Required |

**Example:**

For this example we could consider the merchant pool of “Processor StarFleet” (processor initials “ST” and AMI Processor code 777). Let’s say this processor has the following Merchant Agreements**.**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Case # 1** | **Case # 2** | **Case # 3** |
| AMI Merchant | 4547 | 4546 | 14456 |
| Processor Merchant Number | 1277387016252880 | 1277387016252990 |  |
| Application Received Date | 6/13/2005 | 6/10/2005 | 9/6/2005 |
| Cash Advance Amt | $17,000.00 | $7,400.00 | $60,000.00 |
| Application Delayed Date |  |  | 11/6/2005 |
| Pending Items |  |  | Landlord ref revise with option to complete file |
| Application Out Of Delay Date |  |  |  |
| Application Approved Date |  | 6/12/2005 |  |
| Application Declined Date | 6/15/2005 |  |  |
| Funding Date |  | 6/13/2005 |  |
| Contract Number | E10112121111 | E10112121112 | E10112121113 |

For this example, the processing date would be June 15th, 2005. This would mean that AMI would create a file called ST050615.STS, and would have the following data:

## Response Setup File (RSF)

**Extension:** RSF

**Creator:** Processor

**Contents:** To acknowledge the WSF File and include additional information the processor needs to exchange with Advanceme, Inc.

**Header record:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** |  |
| Header Constant | Varchar(8) | HEADER | Required |
| Processor Processed date | Varchar(8) | Date the Processor processed the file | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | Required |

**File Detailed Structure:**

| **Name** | **Data Type** | **Description** |  |
| --- | --- | --- | --- |
| Processor Processed Date | Varchar(8) | Date the Processor processed the record | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | Required |
| AMI Merchant Number | Varchar (9) | MerchantID as existing on AMI System. | Required |
| Processor Merchant Number | Varchar (15) | Merchant Number as assigned by the processor. | Required |
| Balance | Numeric(14,2) | Total amount due by Merchant to AMI. This includes Renewal contracts in process | Required |
| (Specified Percentage) Rate | Numeric(14,2) | Percent to be withheld. A Percentage of 0.0 indicates the Merchant has remitted the full Specified Amount of receivables purchased by AMI. | Required |
| Terminal | Varchar (3) | Number of credit card terminals at the merchant location | Required |
| Setup Status | Varchar(2) | States if the merchant setup was Successful or not. Possible values are :   * 0 : “Successful” * 1 : “Failed” or * 2 : “No change” | Required |
| INFO | Varchar(200) | Information Field from the Processor. In case of a failure in the merchant setup, it explains why the setup failed. |  |

**Trailer record:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** | |  |
| Trailer Constant | Varchar(7) | TRAILER | | Required |
| Processor Processed date | Varchar(8) | Date the Processor processed the file | | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | | Required |
| Sum of Balances | Numeric(14,2) | | Total amount of Balances included on the detail of the file. | Required |
| Number of Rows | Numeric(5) | | Total number of rows included on the file (it does not include the header and trailer rows) | Required |

**Special Considerations:**

The fields Processor Processed Date, AMI Processor Code, AMI Merchant Number, Processor Merchant Number, Balance and Specified Percentage Rate should be replicated from the previous file sent to the processor as acknowledgement with :

* status of “0” , if the merchant’s setup was successful.
* status of “1”, if the merchant’s setup was unsuccessful, and on the text field an explanation should be given indicating the setup failure reason.
* status of “2”, if the merchant’s current setup did not change.

**Example:**

For this example we could consider the merchant pool of “Processor StarFleet” (processor initials “ST” and AMI Processor code 777). Let’s say this processor has the following Merchant Agreements:

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Case # 1** | **Case # 2** | **Case # 3** |
| AMI Merchant Number | 4547 | 4546 | 14456 |
| Processor Merchant Number | 1277387016252885 | 1277387016252999 | 1277387016252888 |
| Balance | $10799.61 | $0 | $4154.98 |
| (Specified Percentage) Rate | 24% | 25% | 15% |
| Observations |  | This merchant just finished performing on its agreement and there are no other contracts, or the merchant doesn’t belong to an affiliation that has outstanding balances of unremitted purchased future receivables. |  |

For this example, the processing date would be June 18th, 2005. This would mean that AdvanceMe would create a file called ST050618.WSF, and would have the following data:

  
In this case, the processor would receive the WSF and generate the Response file, to acknowledge the success of failure of the merchant’s setup transactions submitted on the WSF file. In this scenario, let’s say that the first two transactions are successfully set up, but in the last one, the processor didn’t recognize the merchant’s processor number. In this case, Advanceme Inc would receive the following file (ST050618.RSF) from the processor, and this file would have the following data:   


## Processor Activity File (PAF)

**Extension**: PAF

**Creator**: Processor

**Purpose**: To notify AdvanceMe of all activity processed by the processor for an activity period. It also indicates the Specified Percentage rates taken for each processor and the amount withheld for AdvanceMe, Inc.

**Header record:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** |  |
| Header Constant | Varchar(8) | HEADER | Required |
| Processor Processed date | Varchar(8) | Date the Processor processed the file | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | Required |

**File Detailed Structure:**

| **Name** | **Data Type** | **Description** |  |
| --- | --- | --- | --- |
| Processor Processed date | Varchar(8) | Date the Processors processed the record | Required |
| AMI Processor Code | Varchar(9) | Processor code as existing on AMI System | Required |
| Processor Merchant Number | Varchar(25) | Processor assigned merchant number | Required |
| Retrieval source | Numeric(1) | Retrievalsource code  VISA / MC: 1 | Required |
| Total Amount | Numeric(14,2) | Total amount processed by the Merchant during the activity period specified by Start Date and End Date fields. Money going from AdvanceMe inc. to the merchant is positive | Required |
| Withdrawal Amount | Numeric(14,2) | Withdrawal amount withheld for AMI. Money going from the merchant to Avanzame is negative. Money going from AdvanceMe inc. to the merchant is positive | Required |
| Activity type | Numeric(3) | Withdrawal: 70  Adjustment: 20 | Required |
| (Specified Percentage) Rate | Numeric(14,2) | Percent withheld. (1.00 means 100%, 0.50 means 50% and so on) | Required |
| Start date | Varchar(8) | Start date of the activity period. | Required |
| End date | Varchar(8) | End date of the activity period. | Required |

**Trailer record:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Data Type** | **Description** | |  |
| Trailer Constant | Varchar(7) | TRAILER | | Required |
| Processor Processed date | Varchar(8) | Date the Processor processed the file | | Required |
| AMI Processor Code | Varchar (9) | Processor code as existing on AMI System | | Required |
| Sum of Withdrawal Amounts | Numeric(14,2) | | Total amount of Withdrawal Amounts included on the detail of the file. | Required |
| Number of Rows | Numeric(5) | | Total number of rows included on the file (it does not include the header and trailer rows) | Required |

**Example:**

For this example we could consider the merchant pool of “Processor StarFleet” (processor initials “ST” and AMI Processor code 777). Let’s say this processor has the following Merchant Agreements and where sent via WSF files to the processors (see Withdrawal setup files for more details)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Case # 1** | **Case # 2** | **Case # 3** |
| AMI Merchant Number | 4547 | 4546 | 14456 |
| Processor Merchant Number | 1277387016252885 | 1277387016252999 | 1277387016252888 |
| Balance | $10799.61 | $0 | $4154.98 |
| (Specified Percentage) Rate | 24% | 25% | 15% |
| Observations |  | This merchant just finished performing its agreement and there are no other contracts, or the merchant doesn’t belong to an affiliation that has outstanding balances of unremitted purchased future receivables. |  |

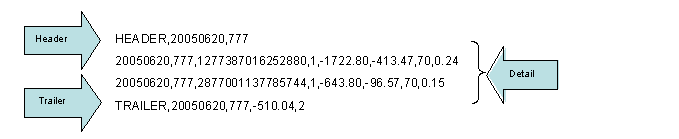
Let’s say that those merchants have the following amounts from the Processor’s Master Card / Visa Transaction Settlement File :

|  |  |  |
| --- | --- | --- |
|  | **Case # 1** | **Case # 3** |
| Processor Merchant Number | 1277387016252885 | 1277387016252888 |
| Settlement Amount on MC/Visa Settlement File | $1722.8 | $643.8 |

This means the withdrawal amount that AMI should receive would be :

|  |  |  |
| --- | --- | --- |
|  | **Case # 1** | **Case # 3** |
| Withdrawal amount (settlement amount \* Rate) | $413.47 | $96.57 |

If the processing date for the processor would June 20th, 2005, it means the processor would create a file ST050620.PAF, and would have the following data:



Notice than the total amount on the file (in this case $510.04) should match the ACH deposit to AdvanceMe bank's account for the money withheld on this file

##### Report Format:

|  |  |
| --- | --- |
| **Report Name** | **Format** |
| Contracts Report |  |
| Collection |  |
| Database report |  |
| Monthly MCA |  |
| Notes |  |
| Prequalifications |  |
| Turn Analysis |  |
| Executive Report Daily |  |
| Executive Report monthly |  |
| Investigation |  |
| Monthly Contracts |  |
| Monthly Income |  |
| PAF Files Reports |  |
| Refunds |  |
| Scoring Report |  |
| Status Report |  |
| Volume Reports |  |
| Work flow reports |  |
| Written Off Contracts |  |
| WSF Report |  |