



Hochschule Esslingen University of Applied Sciences
Atrato Technologies
Winter Semester 2022
Degree thesis, 30 Credits

Hochschule Esslingen
University of Applied Sciences

Progressive automation of disbursement processes through integration with the Mexican digital bank- ing system

DHIK Double Degree: ITESM - Hochschule Esslingen — Faculty of
Mechatronics and Electronics

Author: José Carlos Toscano Alcalá

Author:	José Carlos Toscano Alcalá
Main Supervisor:	Prof. Dr. Markus ENZWEILER, Department of Computer Science and Engineering, Hochschule Esslingen
Co-supervisor:	Juan Pedro Casián Porter Department of Product Development at Atrato, Atrato

“The first rule of any technology used in a business is that automation applied to an efficient operation will magnify the efficiency. The second is that automation applied to an inefficient operation will magnify the inefficiency.”

-Bill Gates

Abstract

With the growth in Atrato's customers and partners consuming its main financial product, the number of loans granted on a daily basis has increased and the manual money disbursement process has become an exhaustive, repetitive, and error-prone task. In order to ensure a scalable and reliable disbursement procedure, this should progressively migrate to a fully automated activity compatible with current processes. To track and monitor the complete disbursement procedure, a Balance System was built enabling an understanding of both a merchant's balance in a general and in a very granular way, implementing a logging system to ensure detailed visibility and traceability of the full process, integrated with existing Identity and Access Management modules for internal security. Connecting this Balance System to the entity provider of the access to the Mexican digital banking system through a custom API and web services for incoming updates and a private network. All this implementation considers certain flexibility on the disbursement modalities and was developed through Object Oriented Programing Principles, managing models and controllers, to ensure the best programming practices and a correct integration with the current Business rules, processes, and models. Keywords: Balance system, Buy Now Pay Later, disbursement, automation, compatibility

Preamble

Foreword

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin in diam. Nam dignissim facilisis lorem. Aliquam orci ipsum, egetas non, pretium at, venenatis in, felis. Pellentesque pellentesque. Vivamus in mi. Suspendisse cursus, augue quis malesuada mollis, massa lorem accumsan est, sed vulputate est nibh id pede.

Acknowledgements

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin in diam. Nam dignissim facilisis lorem. Aliquam orci ipsum, egetas non, pretium at, venenatis in, felis. Pellentesque pellentesque. Vivamus in mi. Suspendisse cursus, augue quis malesuada mollis, massa lorem accumsan est, sed vulputate est nibh id pede.

Notations and conventions

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin in diam. Nam dignissim facilisis lorem. Aliquam orci ipsum, egetas non, pretium at, venenatis in, felis. Pellentesque pellentesque. Vivamus in mi. Suspendisse cursus, augue quis malesuada mollis, massa lorem accumsan est, sed vulputate est nibh id pede.

Contents

List of Figures	2
1 Introduction	1
1.1 Motivation	1
1.2 Starting points	1
2 General overview	1
2.1 Current business pipelines	1
2.1.1 Understanding Atrato's Partners, Merchants and Stores	1
2.1.2 Disbursement process	1
Partner's dashboard	2
2.1.3 Compatibility with current business and application processes	2
2.2 Automation of disbursement process requirements	2
2.3 Balance system for partners	2
2.3.1 Balance Updates	3
2.3.2 Bank Transfers	3
2.3.3 Triggers	3
2.3.4 Understanding the correct handling of partner's commissions	3
2.4 Refactor and Changes	4
2.4.1 Registration of merchant's pipeline changes	4
2.4.2 Refactor in partiality selection architecture	4

List of Figures

—

1 Introduction

1.1 Motivation

Throughout the development of Atrato's main product, a Buy Now Pay Later (BNPL) financial service, many tasks and processes have been automated to enable a fast and efficient response to the final customers. Nonetheless, after-signature processes still were handled manually by an overcharged treasury team, handling more than 1,000 daily operations regarding money disbursements and credit cancelations. A process that became very time consuming, human-error prone and unsustainable at a larger scale. Atrato's BNPL was very efficient to the end users but started to generate plenty of friction and operational load to the partners and operational side.

As Atrato's market started expanding, new partnerships with bigger and more relevant merchants required efficient and flexible processes not only for the end user, but also for every new merchant that joins Atrato as a partner. Atrato's partnership with what today is their biggest merchant demanded immediate disbursements after every approved purchase. Not only immediate bank transfers needed to be handled, but also the process of managing credit updates and cancelations needed to be very efficient. To enable this partnership to happen, a balance system needed to be developed, capable of managing bank transfers with flexible triggers and handling changes in previous credits or cancelations.

1.2 Starting points

Before the development of the balance system for merchants and stores, Atrato was able to receive payments through different providers and manage the money properly. In terms of taking money out of Atrato's banking account and sending it to a different account only the treasury team, composed by 2 people, was able to do this task. After confirming that the money was disbursed correctly, they proceeded to update every credit, adding a flag that indicates that the credit was already disbursed. After this, a manual spread sheet was generated to send it as a disbursement report to every merchant and store.

Regarding each merchant's banking information, none of it was saved as part of the information stores while adding a new merchant. All of it was later requested by the treasury team, making that the information needed for an automatic disbursement was not centralized. Additionally, every bank account needed to be authenticated before transferring a credit's disbursement for security purposes.

In terms of how much money needed to be transferred per credit, it could always vary depending on every merchant's current commission or if any special promotion or discount is active as part of the commercial team. All these factors needed to be considered before making a money disbursement. Furthermore, every new merchant could go through a trial period, where very specific commissions apply and would later change depending on its monthly origination.

This poorly structured process was the starting point before any automation could be developed, where current business processes should be able to continue operating in a similar way as they were already doing.

Concerning technical starting points, Atrato's codebase has been slowly migrating from functional programming, developed in JavaScript, to an Object-Oriented Programming structure in Typescript. All data is stored and handled in a single relational database with MySQL as the engine but consumed through TypeORM, an Object-relational mapping (ORM), allowing a Model-Controller-Router Structure for the server side.

2 General overview

2.1 Current business pipelines

It is very important to understand existing processes before attempting to automate these tasks. Atrato has partnered up with numerous merchants with possibly more than one store throughout Mexico. Every single partnership was discussed and agreed upon different terms, resulting in different expectations regarding how the money will be disbursed once a credit is granted through Atrato's application process. These processes could either be monthly, weekly, daily, or individually per credit. Additionally, merchants could be conditioned to specific commissions depending on different factors like their monthly origination or if they were on a specific trial period. All these rules were not specifically written down nor followed any guideline. This meant that every time a disbursement needed to be made, the person in charge should review every commission agreed and compute the merchant's monthly origination to know exactly what amount and in what conditions should the disbursement be made.

2.1.1 Understanding Atrato's Partners, Merchants and Stores

Atrato's partners are those commercial allies that offer its Buy Now Pay Later service as part of their payment options. These partners are internally classified as Merchants and each of these merchants could have 1 or more registered Stores. The application process for a loan with Atrato takes place through Atrato's web application by filling in information about the customer's personal profile and financial information, indicating the Merchant and Store in which the loan will take place. Once the application is fully authorized by Atrato and the customer has finished the signature of its digital contract, then the store proceeds to deliver the product to the customer. Now, depending on the type of disbursement agreed upon the merchant's registration the money regarding the purchase will be transferred to the merchants banking account. This describes the process for a single application, but multiple applications need to be handled both by Atrato's partners and Atrato's internal team.

2.1.2 Disbursement process

The disbursement of a credit could be as simple as taking the amount of the credit and transfer it to the bank account related to the merchant. The complexity is added when a merchant does not want to receive plenty of bank transfers per day or even per week and rather have all the money disbursed as a single transaction at the end of the month. Since every partner expects this task to be handled upon their own agreed terms every bank transfer needs to be carefully reviewed. Additionally, just like a partner can have their own terms, Atrato has its own specifications and requirements to every partner. Some commissions may be included to the partner depending on their specific monthly origination and type of partnership. Furthermore, already disbursed credits could be later canceled or a change in amount may happen for many reasons, leading to a negative balance with merchants. Taking all these factors into account, a money disbursement is made firstly depending on the expected disbursement periodicity, then the total amount to be transferred is computed depending on the number of credits and their commission and this amount is manually sent in a single bank transfer. Once the bank transfer is successfully confirmed a report is generated indicating every credit that was part of the disbursement, their corresponding commission and any change in amount or cancellation that was taken into account to compute the total amount. This report is then sent to every single merchant enabling complete transparency and understanding of every bank transfer that is sent.

Partner’s dashboard

Every partner has access to an internal dashboard where they can visualize and manage their customers’ applications and give follow ups on each product’s delivery or cancellation. The money regarding a new credit will not be disbursed until the partner notifies the delivery of the product through their dashboard. This notification works as Atrato’s triggering event to dispatch the money disbursement, the process that is intended to eventually be fully automated. This process is very important to be sure that no credit is disbursed before they are marked as “delivered”. Once the credit’s product is delivered the partner can visualize when the credit is disbursed. This dashboard is helpful to compare the information received from the disbursement’s report and what they visualize as the credits that were already delivered and disbursed.

2.1.3 Compatibility with current business and application processes

An active customer’s application turns into a credit once the information is validated every required document is uploaded and an offer is sent. If the customer agrees to the term’s presented in the offer, then it goes through a digital signature process, where all the details of the credit can be confirmed. Once the signature is generated the credit is finally created. It is in this moment where the calculations for the credit’s commission need to be made to know exactly on what terms was this credit granted. The existing architecture supported a static commission per merchant, which was immediately assigned to the credit, but not necessarily represented the correct commission. A merchant’s commission could change every month depending on the previous month’s origination. Furthermore, depending on the type of merchant the commissions could vary even depending on the number of payments agreed for the credit. Temporary promotions should also be considered in the process before deciding the exact commission for every credit, where a specific deal for interest free months for the end customer could represent a higher commission for the merchant offering this promotion. All these factors must be considered before attempting to automate the disbursement process, without implying a change on current application or business processes. Every agreed commission, trial period and origination thresholds for every merchant should be saved in a standardized manner, enabling a better understanding and implementation of the agreed terms.

2.2 Automation of disbursement process requirements

The first step towards this automation should creating a detailed pipeline of the process. Every task and validation that was manually done before transferring a partner’s origination to their bank account will now need to be defined and structured in a regulated process. The automation of such a delicate process can be achieved through the development of a modular Balance system for each of Atrato’s partners. This balance system should be completely traceable. This means that we should be able to know what is happening all the time, exactly how the computations are made and every factor that is affecting a partner’s balance. It should be able to be handled independently for each partner with the possibility to disable the automation for some of them at any moment. A security step should be implemented, where some balance systems could require a manual confirmation of the bank transfer before it is sent. This could be helpful for transitioning from a manual process with some partners into the whole automation pipeline. Additionally, an integration with Banco de Mexico, the Mexican Banking System regulatory agency is needed to handle bank transfers properly. Since Atrato has not yet the technology to independently make bank transfers from one account to another this integration will be done through a third party that provides all access and security that is required.

2.3 Balance system for partners

This system should be modular enough to handle even different Bank Accounts per merchant. Since every merchant could have different stores, the modularity of the banking system will be done individually per every store. The system will describe every update to the store’s balance, enabling a complete understanding of how any contribution to the balance was made or how it was adjusted. It will relate every credit generation or cancellation, every money disbursement,

and every manual adjustment to the ongoing bank transfers. The balance system will be handled through Balance Update object which will eventually generate Bank Transfer objects all of which will be triggered by some specific requirements. In the following sections we will get a general overview of these models.

2.3.1 Balance Updates

A Balance Update will be every movement, transaction, adjustment, or cancelation that directly affects the system's balance. Balance Updates serve as the linking entity between a confirmed credit and a store's balance considering the appropriate commission. Every Balance Update will contribute in a way to the general balance, recording the previous balance and the new balance after its contribution. In this way, to effectively know the system's current balance it would be enough to inquire the last Balance Update. Balance updates can be created through the confirmation or cancellation of a new credit, through the confirmation of a successfully disbursed Bank Transfer or through the cancelation of a previously confirmed disbursement. Furthermore, a Balance Update could be generated manually by the treasury team whenever a Balance Adjustment is required. Furthermore, Balance Updates could have different specifications, expected behaviors or status updates depending on their type. These specifications will be furtherly discussed in Chapter 3.

2.3.2 Bank Transfers

A Bank Transfer object will be the entity describing the money movements between Atrato's bank account and every store's bank account. Bank Transfers will only be generated through and composed by Balance Updates and their status will be directly related. Once one or more Balance Updates are linked to one Bank Transfer, they will not be able to be related to any other Bank Transfer. Once the Bank Transfer is sent through the online banking system and the confirmation is received a new Balance Update will be generated indicating the money disbursement, or the corresponding updates in the Balance Updates' status will be handled according to any possible cancellation.

2.3.3 Triggers

Atrato's partners require specific disbursement processes, whether it is a daily, weekly, or monthly disbursement of the credits approved during this period or an immediate disbursement per credit. To allow this flexibility, the generation of a Bank Transfer will be initiated through specific triggers. Every store can have one of the following disbursement types: Instant, Hourly, Weekly or Daily. These four disbursement types will be triggered accordingly through Cron Jobs. Once these events are triggered, all pending Balance Updates will be considered for generating a new Bank Transfer only if the total amount of these pending Balance Updates is higher than the minimum amount required for a Bank Transfer. This threshold is defined to avoid sending Bank Transfers with a low amount of money, since every banking transaction also represents a cost for the company. There may be some edge cases where the total amount of the pending Balance Updates is even lower than \$0.00, this just means that for some reason the system's balance is negative and no money will be disbursed until the balance becomes positive again through the origination of more approved credits or any further update to the store's balance. Note that all these updates can only be done through Balance Update object.

2.3.4 Understanding the correct handling of partner's commissions

Every merchant, as well as every store, has a standard commission. All the business logic regarding the correct amount for these commissions should be handled independently to the automatic disbursement process. Commissions can change dynamically depending on different factors, like special promotions, discounts or regarding the merchant's monthly origination. All these process and specifications should not interfere with the automatic disbursements. Every credit must have a commission assigned before it can be approved for disbursement. This will be the commission that will be considered for the computation of the amount for the Balance Update generated by the approval of a credit. Independently of how a partner's commissions

vary, since the commission is directly assigned to the credit, these two business processes will not interfere with one another.

2.4 Refactor and Changes

Current business processes and the existing models for merchants, stores and all the structure for selecting a credit's loan term were not able to support a new system where disbursements could be automated. Since the disbursement of a credit is the missing link that could complete the full automation of the credit application and lending process, all the entities involved converge in this point and some of them will require some very specific changes: A store's bank account should be validated and ready to receive payments, a credit's commission should be able to be determined according to the term selected, the amount requested and any applicable promotion that could be involved. Hence, additional to the changes in current pipelines that are required, a complete re-design of the architecture related to how a customer could select the conditions of the loan they are applying to must be done.

2.4.1 Registration of merchant's pipeline changes

During the registration process of any new merchant and their corresponding stores involved, the banking information of the stores should now be a required field. Once a merchant is added, its first store should be added as well, and every time a store is added its banking information should be included. The Treasury team oversees validating a merchant's commission depending on the terms agreed upon their registration. To do this, a new attribute, `isValidated`, will be added to both the Merchant and the Store models indicating if the information has already been validated by the Treasury team. This extra validation step is now included as part of the registration process of a merchant. Every time the information of a merchant or a store is added or edited, the `isValidated` flag will be set back to false, indicating that a new validation of the Treasury team is needed. The whole registration pipeline is mounted in a dashboard developed for internal use with CRUD operations. With any subtle change in the Update method, as shown in Figure N, the `isValidated` flag will always turn back to false when used, requiring an additional validation.

iiiiiiifigurelllllll

A merchant cannot be validated if it does not have at least a validated store. Furthermore, any merchant that is not validated will not be displayed as an option in the application form. Additionally, a new Model will be introduced: Payment Option. Its connection to the Merchant-Store model will be described in the following section.

2.4.2 Refactor in partiality selection architecture

During the development of Atrato's customers' application process the commissions for the merchants were not considered. Furthermore, with the iteration of commercial and partner acquisition processes a credit's commission became more complex. A credit's commission can depend on the number of payments, the credit's amount, and ongoing promotions. To enable this dynamism a refactor of the internal architecture of the term selection options needed to be done. Previously, whenever a merchant was incorporated to Atrato's partners a simple setup of the available terms was done. At this moment we only kept a record of the possible number of payments that were available for this merchant's stores. Be it a range from 3 to 24 monthly payments. We needed to elaborate every merchant's term options into a whole module of Payment Options, where every payment option includes specific interest rates depending on a customer's risk profile, the number of payments, the requested amount and had expiring dates in order to work as promotions. In this way the Payment Options module could support flexibility and the availability of every option could be determined through a customer's input. This Payment Option model will refer to all the possible terms that a store could have and will include details regarding specific interest rates that could vary depending on a customer's risk profile, and will include applicability filters for date and requested amount, the option to be tagged as a promotion, specifications on the specific merchant's commission, a trial period specification and very importantly a flag indicating if its commission could be automatically updated to integrate with the on-going changes in a merchant's commission due to their monthly origination. Please

refer to 2.4.2.2 Payments Options UML for specific details on the model. Payment Options will also go through the validation step from the Treasury team, following the same structure and process as before. Every store should have at least a validated Payment Option before being completely validated. Now, the complete pipeline goes as follows: Whenever a merchant is added, its first store should also be added, and the first payment option for the store should be added as well. Regarding the validation process, it goes the other way around, payment options should first be validated before attempting to validate a store, and once a store is validated the merchant can be validated as well. All of these is just as a general validation to make sure that there is no merchant that does not have a valid store and that every valid store has at least a valid payment option.