***CASE STUDY: Car rental***

Car rental companies own a number of cars and a number of sites where cars are parked when not in use. Customers rent cars for a period of time (having made a reservation, or not) and return them.

We focus on company CARS.

The current process (AS IS), is as follows.

A customer may reserve a car, using the company web site, or the call center (this step is optional).

A customer steps into the office close to the rental car parking site and completes the first step of the *check out*. The contract for the rental is defined (period of rental, name of driver, related ID document and driving license, insurances, damage deposit, partial and total fees, credit card), signed by both parties, and the payment for the rental is completed (payment has two parts, rental and damage deposit – the latter is normally returned at the end of the rental). Further, a specific car (identified by its tag) is assigned to the rental.

Then the customer walks to the car parking site. Here the second part of the check out happens.

An employee checks with the customer the car and lists all visible damages on the car in an annex to the contract. Also this annex is signed by both parties. Then the employee hands the car to the customer (this of course includes the keys) and the rental starts.

The final step is *check in*. The customer drives the car to parking site. An employee receives the car and the keys, checks with the customer for new damages. If there are damages another process starts (we leave this process out of this analysis). At this point the rental ends. The company issues an invoice and possibly returns the damage deposit to the customer.

TO BE process.

The idea is to improve the process by introducing the same innovations used by car sharing companies.

A customer has first to define an account with CARS. In this step the customer uploads his documents (ID, driving license) and a credit card. If all is right CARS approves and the customer can later rent cars. This step can be performed on a PC or smart phone. In any case the customer has to install the CARS app on her smartphone.

When a registered customer wants to rent a car she has to do a reservation (via app or PC).

Check out works as follows. The customer walks directly to the rental car parking, via the app she signals that she wants to start the rental. The app answers with position and tag of the assigned car.

When the customer is close to the car she asks, via the app, to open the car. The system opens the car (the car needs to be modified via a device connected to the cellular network and capable of controlling some car functions, like door open/close). The keys are inside the car. The customer starts the car, and the rental.

The check in is similar. The customer parks the car in the rental car parking, stops the car, exits, and asks the app to close the car. At this point the rental is over.

Invoicing and payments proceed through the credit card.

Damage deposits and possibly damage reimboursements are avoided, introducing by default an insurance to cover all.

*LAB*

Define the TCO for the project needed to obtain TO BE (requires to install device on each car, develop mobile app for customer and develop car remote control back end, and more). Use a 5 year time horizon.

For each cost item, define if it is direct / indirect, fixed / variable capex/ opex

Review unit cost KPIs considering the TCO

Contents

[A -- AS IS 3](#_Toc150244095)

[1 Organizational model 3](#_Toc150244096)

[2 Process 4](#_Toc150244097)

[Reservation 5](#_Toc150244098)

[Checkout 5](#_Toc150244099)

[3 KPIS – AS IS 5](#_Toc150244100)

[4 Data 6](#_Toc150244101)

[5 Technical model 6](#_Toc150244102)

[B – TO BE 8](#_Toc150244103)

[1 Organizational model 8](#_Toc150244104)

[2 Process 8](#_Toc150244105)

[Onboarding 9](#_Toc150244106)

[Checkout 9](#_Toc150244107)

[3 KPIS – TO BE 10](#_Toc150244108)

[4 Data 10](#_Toc150244109)

[5 Technical model 11](#_Toc150244110)

# A -- AS IS

## 1 Organizational model

In yellow the parts as described in the text above. Other parts are assumed considering a large rental company (at least active at country level, with many local agencies)

Besides the rental company (where is our focus), customer and payment circuits are listed since they interact with rental company.

Car rental company

Accounting

Finance

Human resources

IT area (manage web site and all IT services, manages call centers too?)

Legal department

Sales and marketing (will implement reservations)

Reservation office

Customer assistance (accidents, complaints..)

Car management (“manufacturing”)

Local agency (repeated many times, in each city, airport, railway station..)

(geo structure)

Office

Parking site

Maintenance (cleaning, small repairs) (full maintenance is outsourced to external

workshops)

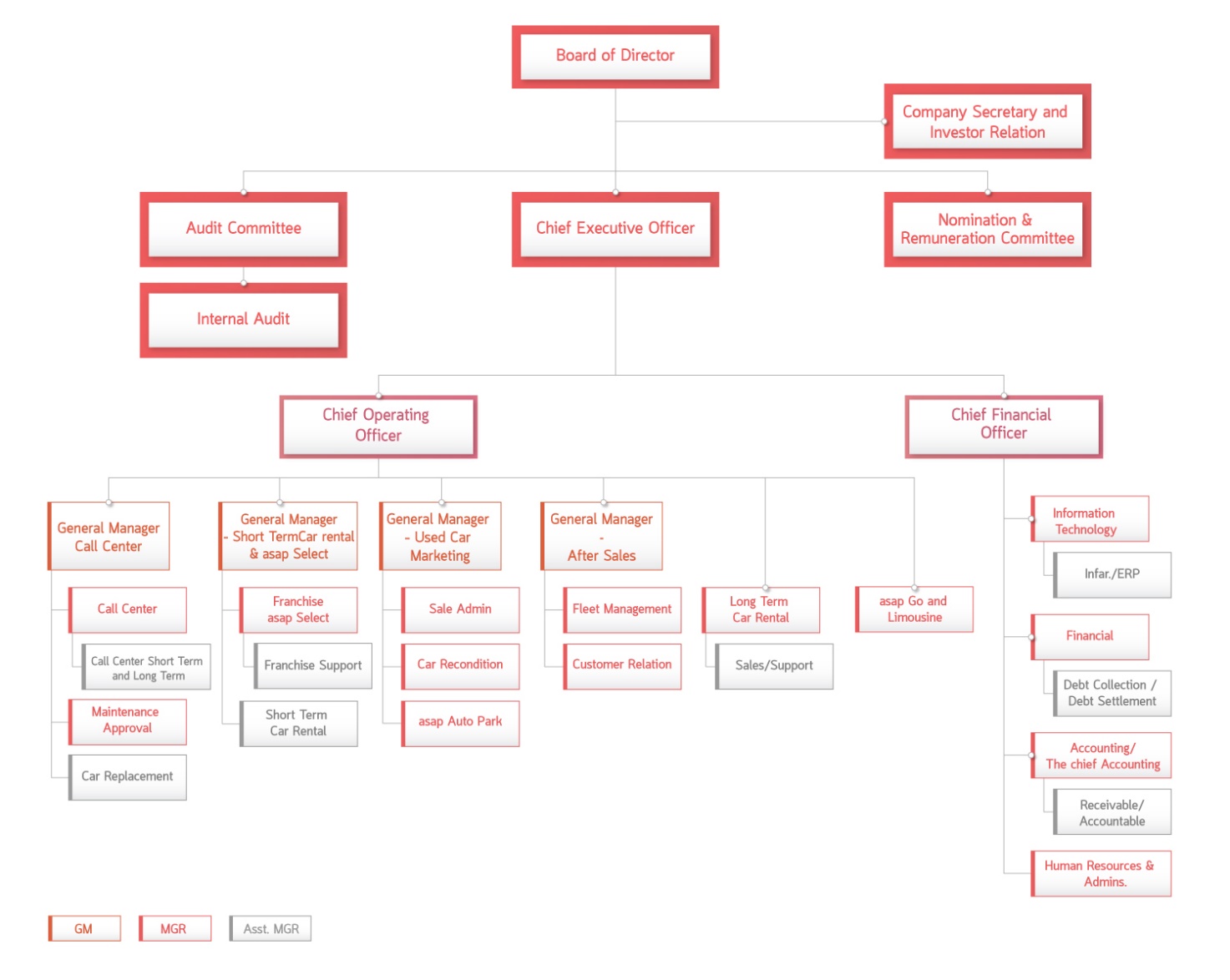
Purchase office (for cars)

Customer

Payment circuit (credit card , bank circuit)

(yellow = specific to the part actually described in case study. Non yellow, assumptions about part of CARS not described in case study)

As a reference, below is an org chart of an existing car rental company

**

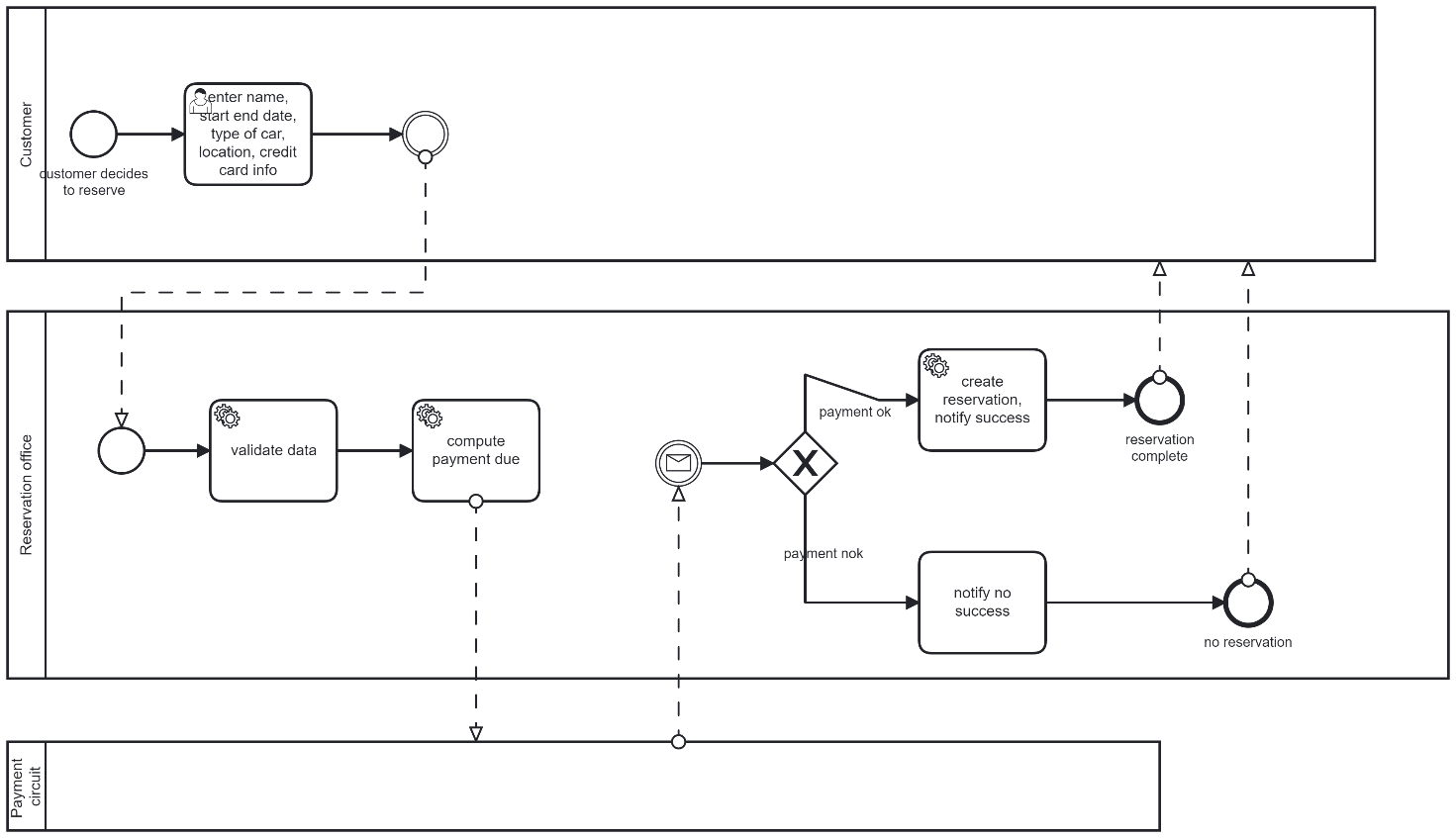
## 2 Process

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Process name | In | out | description | OU involved |
| Reservation | Customer decides to reserve | Reservation complete | Customer enters her data + car requested, dates, place; makes payment | Customer + reservation office + payment circuit |
| Check out | Customer requests car (having reservation or not) | Rental started | Customer gives id, driving licence, credit card, signs rental agreement, pays damage deposit, start rental | Local agency, customer, payment circuit |
| Check in | Customer returns car | Rental ended | Damage check, return deposit, close rental | Local agency, customer, payment circuit |

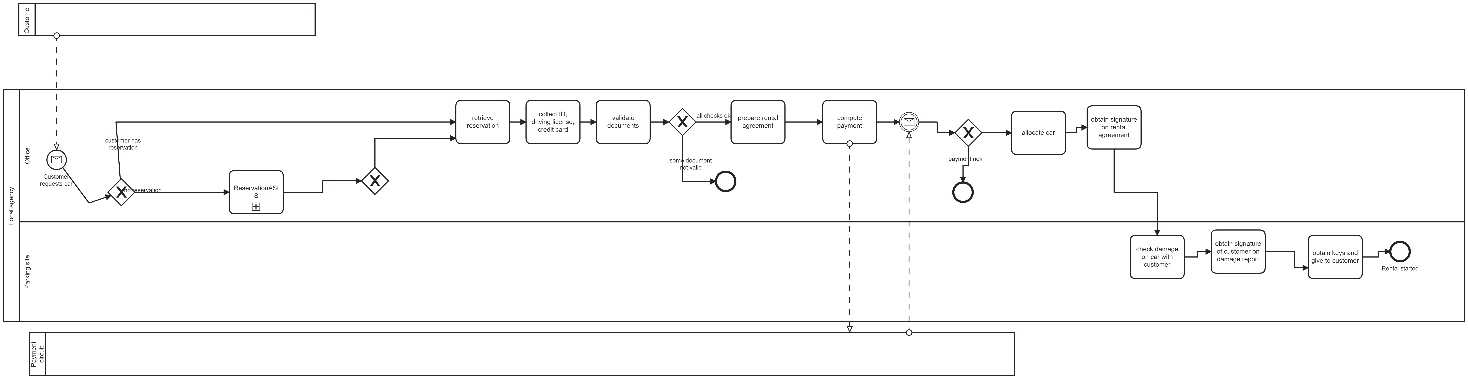
The 3 processes above are in fact part of a larger process (from reservation to end of rental). Modeling 3 processes instead of a single (big) one has the advantage of splitting the large model in 3 smaller ones. Furthermore, the three processes happen in 3 very well separated moments in time (ex reservation today, checkout in 2 months, checkin a week after checkout).

### Reservation

Only via web site (no call center)



### Checkout



### Checkin

TBD

## 3 KPIS – AS IS

Reservation

|  |  |  |
| --- | --- | --- |
| General | Time interval 1 year  Input volume : #customer decides to reserve  Output volume # reservation complete  Resources: human no  Non human: IS (reservation back end + main server CARS, as in deployment diagram) , payment circuit  Inventory: cars available |  |
| Efficiency | Unit cost = total cost / #reservation complete  Total cost : cost of IS (pro rata for this process only) + payment fees from payment circuit |  |
| Service | Stot = Time from ‘customer decides to reserve’ to ‘reservation complete’  (could include some queue time from web site, lead time + response time)  Stot = Sinternal + Spaymentcircuit  Important to know service time of payment circuit |  |
| Quality | Conformity  #reservation complete with defects / # reservation complete  Defects: wrong location, wrong car type, wrong customer name, allocate car that is not available  Customer satisfaction : survey about usability of reservation process on web site |  |

Checkout AS IS

|  |  |  |
| --- | --- | --- |
| General | Input volume #customer requests car  Output volume #rental started  Resources: employees in office, employees in parking site, IS, parking lot , copy machine, office, appliances in office ( paper for copies, storage space for paper, computers), payment circuit | Lets consider scenario ‘customer has reservation’ and nominal scenario (no exceptions) |
| Efficiency | Unit cost = total cost / #rental started  Total cost: cost of employees in office + cost of employees in parking lot + cost of IS, cost of parking lot, cost of office and appliances in office , payment circuit fees |  |
| Service | Time from ‘customer requests car’ to ‘rental started’ |  |
| Quality | Conformity : #rental started with defects / #rental started  Defects: car not available, car of different type as promised in reservation, wrong keys, wrong position of car,  Customer satisfaction survey about checkout process | Car is broken should be a defect for the car maintenance process |

Checkout TO BE

|  |  |  |
| --- | --- | --- |
| General | Input volume #customer requests car  Output volume #rental started  Resources: ~~employees in office~~, ~~employees in parking site~~, IS, parking lot , ~~copy machine, office, paper for copies, storage space for paper~~, ~~payment circuit~~, device on cars and remote control system | Here customer must always have reservation |
| Efficiency | Unit cost = total cost / #rental started  Total cost: cost of employees in office + cost of employees in parking lot + cost of IS, cost of parking lot, cost of office and appliances in office , payment circuit fees + cost of device on cars + cost or remote control system + cost of app | Total cost should be computed considering the pro rata quota of all expenses for this process. Ex cost of employees in office, considering only the effort they spend for this process. Cost of employees in parking lot, considering only the effort they spend for this process |
| Service | Same as AS IS |  |
| Quality | Same as AS IS |  |

Checkin

|  |  |  |
| --- | --- | --- |
| General |  |  |
| Efficiency |  |  |
| Service |  |  |
| Quality |  |  |

## 4 Data

A diagram of a diagram

Description automatically generated

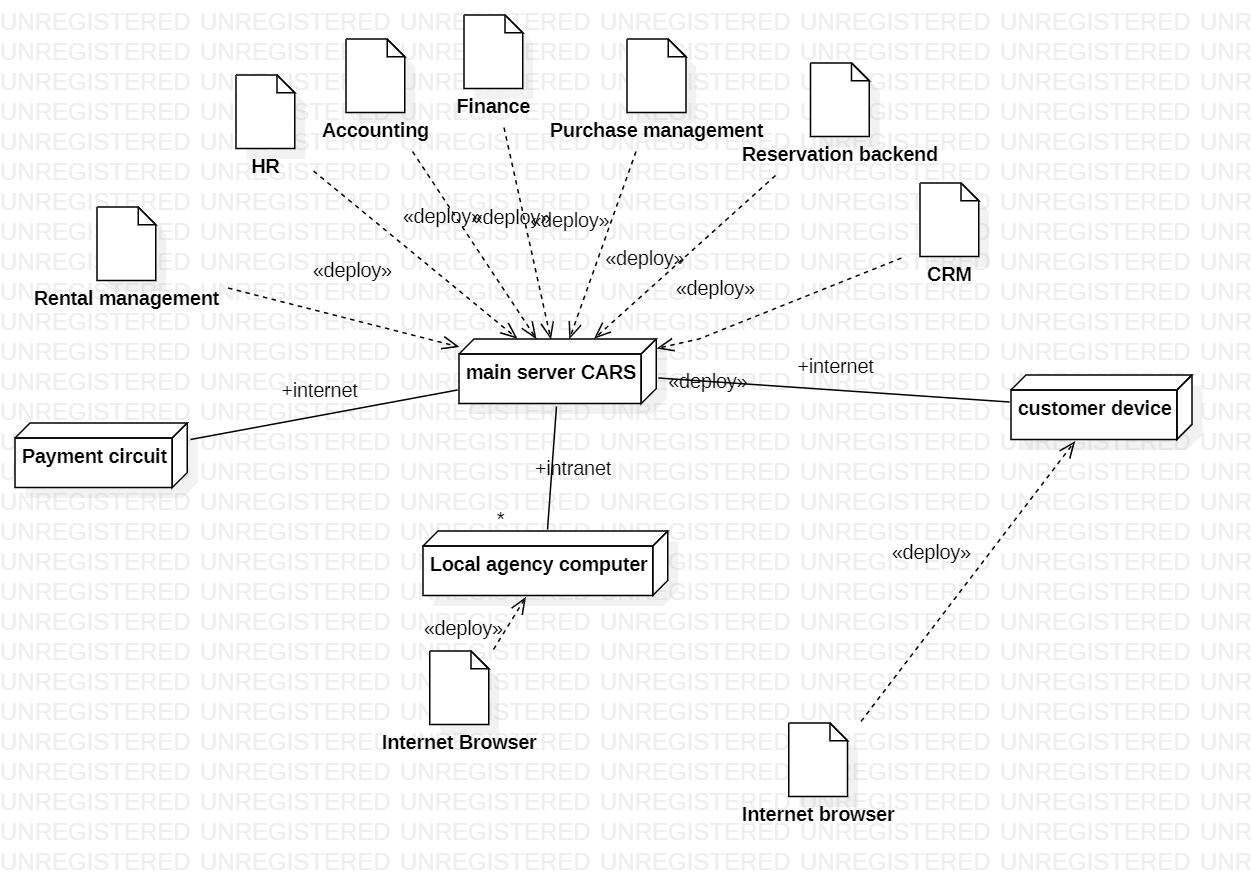
## 5 Technical model

Application portfolio

(each artifact in deployment diagram is an application in the application portfolio)

Rental management, HR, Accounting, Finance, Purchase management, Reservation, CRM

Deployment diagram



# B – TO BE

## 1 Organizational model

Car rental company

Accounting

Finance

Human resources

IT area (manage web site and all IT services)

Legal department

Sales and marketing (will implement reservations)

Reservation office

Customer assistance (accidents, ..)

Car management (“manufacturing”)

**Car remote control (ADDED)**

Local agency (repeated many times, in each city, airport, railway station..)

(geo structure)

**~~Office (CANCELED)~~**

Parking site

Maintenance (cleaning, small repairs) (full maintenance is outsourced to external

workshops)

Purchase office (for cars)

Customer

Payment circuit (credit card , bank circuit)

(yellow = specific to the part actually described in case study. Non yellow, assumptions about part of CARS not described in case study)

In the new organization the office part of the local agency is canceled, since damage check and car key delivery are also avoided. Car damage is managed through insurance (see above), keys are not used anymore, the cars are open/closed via remote control. As a consequence the related infrastructure has to be put in place (Car remote control OU). Car remote control works for all local agencies.

Since the organization structure changes, besides process and technology, this is a third order change

## 2 Process

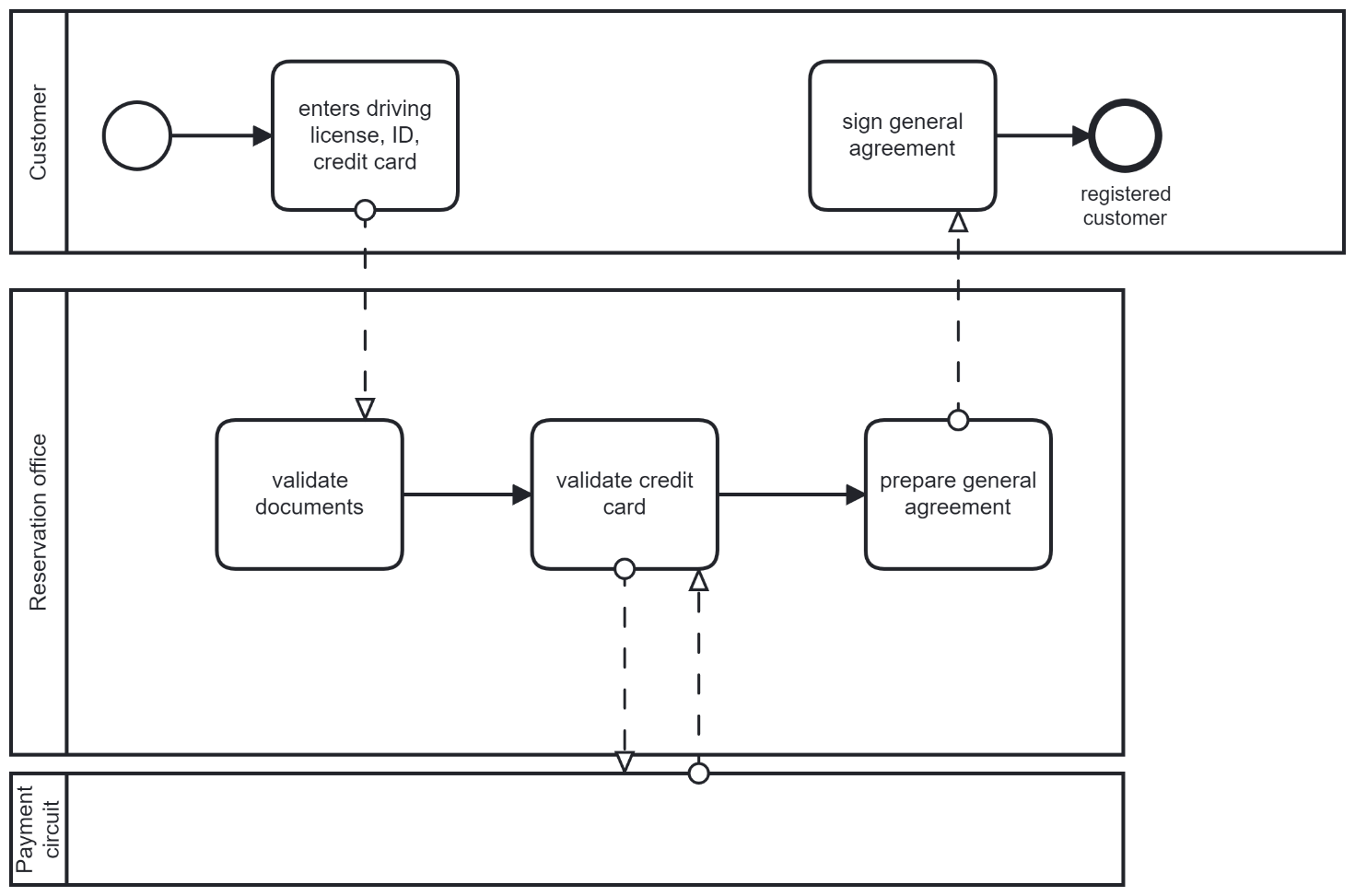
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Process name | In | out | Description | OU involved |
| Onboarding |  | Registered customer | Customer defines and account with CARS, enters driving license, ID, credit card. CARS validates everything, creates account for customer, defines fees, collects signature on agreement | Reservation office, customer, payment circuit |
| Reservation |  | Reservation complete | Customer enters his account, selects location, dates, type of car | Reservation office, customer |
| Checkout |  | Rental started | Allocate car, open car | Customer, Car remote control, |
| Checkin |  | Rental ended | Return car, close it, define payment | Customer, Car remote control  payment circuit |
|  |  |  |  |  |

We need to introduce a new process, onboarding, where a customer defines an account and inserts all her relevant documents (ID, license, credit card), so to avoid this part in the checkout process.

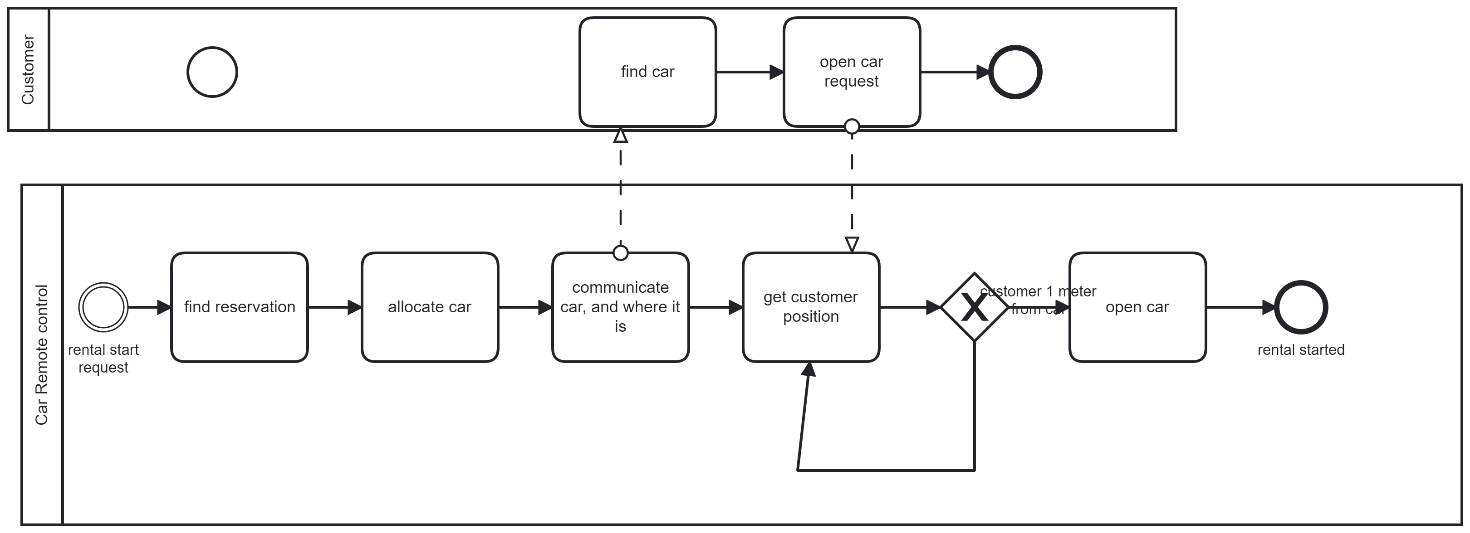
### Reservation

Same as in AS IS

### Onboarding



### Checkout



### Checkin

TBD

## 3 KPIS – TO BE

Onboarding

|  |  |  |
| --- | --- | --- |
| General |  |  |
| Efficiency |  |  |
| Service |  |  |
| Quality |  |  |

Reservation

|  |  |  |
| --- | --- | --- |
| General |  |  |
| Efficiency |  |  |
| Service |  |  |
| Quality |  |  |

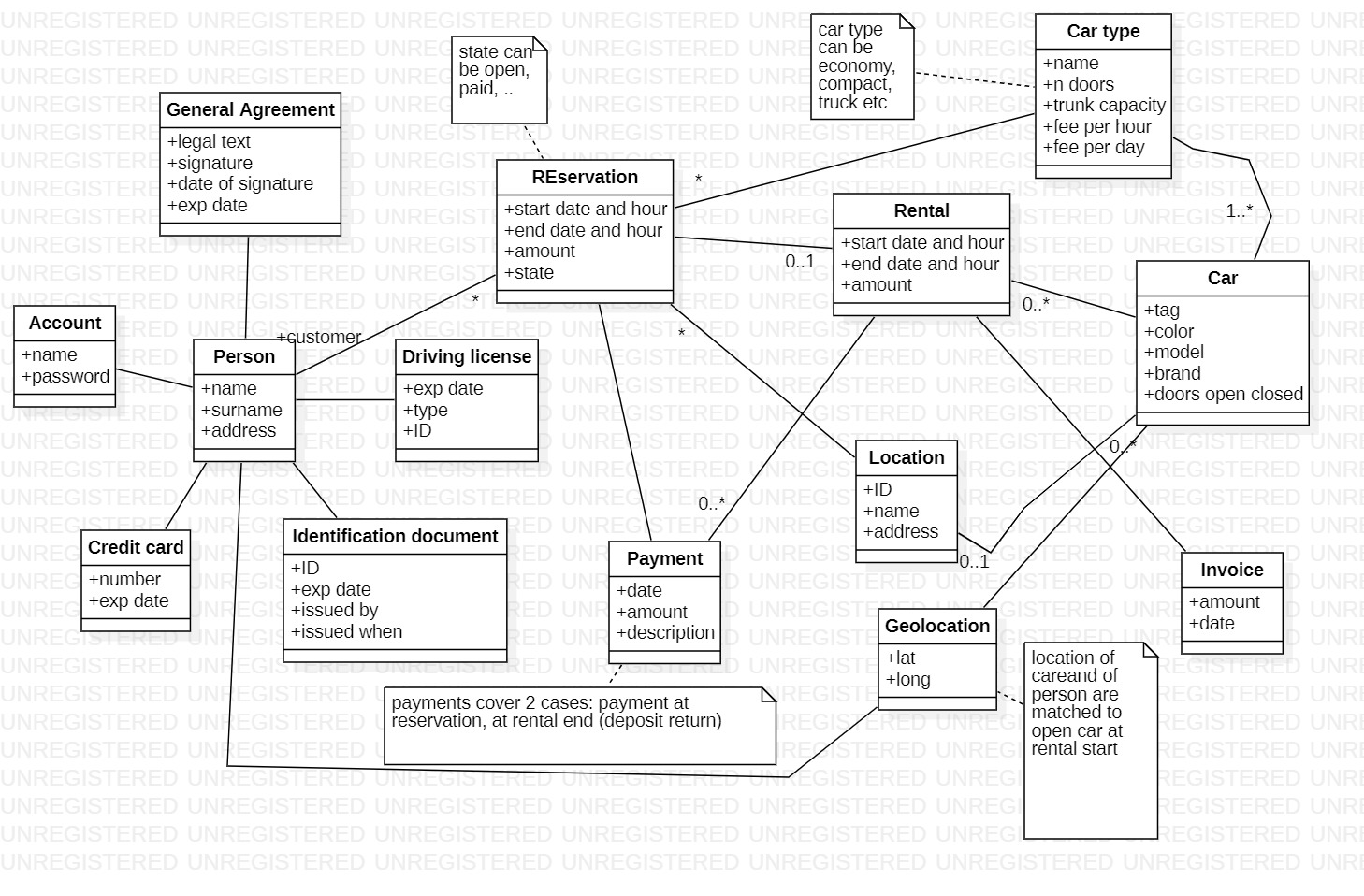
Checkout

|  |  |  |
| --- | --- | --- |
| General |  |  |
| Efficiency |  |  |
| Service |  |  |
| Quality |  |  |

Checkin

|  |  |  |
| --- | --- | --- |
| General |  |  |
| Efficiency |  |  |
| Service |  |  |
| Quality |  |  |

## 4 Data



## 5 Technical model

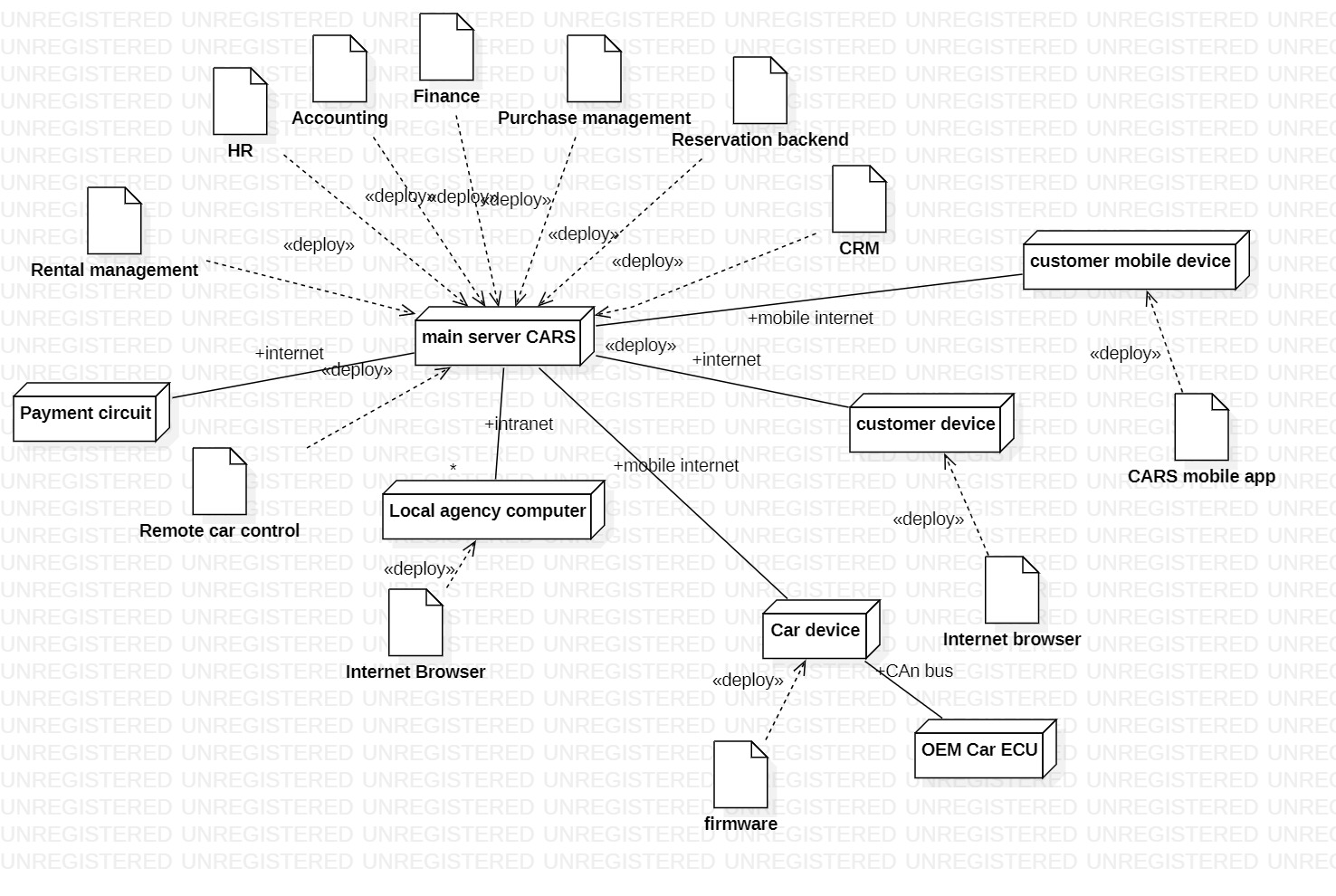
Application portfolio

(each artifact is an application in the application portfolio)

Rental management, HR, Accounting, Finance, Purchase management, Reservation, CRM ,

CARS mobile app, firmware, remote car control (yellow = added in TO BE) Remark also the hw device to be added in each car to control open/close doors, interacting via CAN BUS with existing electronics on board of car.

Deployment diagram



# Comparison (by process)

Reservation

No change

Onboarding

New, comparison not possible

Checkout

|  |  |  |
| --- | --- | --- |
|  | AS IS | TO BE |
| Volume |  | No change |
| Unit Cost  Unit cost = total cost / #rental started  Total cost: cost of employees in office + cost of employees in parking lot + cost of IS, cost of parking lot, cost of office and appliances in office , + cost of device on cars + cost or remote control system + cost of app | Green part of unit cost missing or reduced | Green part added, yellow part changes  Savings: personnel in office, personnel in parking lot, office rental, office appliances  New costs: device in cars, remote control system  The TOBE unit cost should be computed using the TCO (see later)  Unit\_cost\_tobe = TCO /5 /2  TCO is divided by 5 since it is computed on 5 years, while here unit cost is computed over 1 year.  TCO is further divided by 2 since the capex (green part) and also the savings impact two processes (not only checkout but also check in) |
| Service time | At least 10 min in office, 10 min in parking lot, without queue in line (pure response time) | Few minutes (1 min max to open the car when in front of it)  Consider somehow that customer must do onboarding initially. Onboarding is negligible for a customer who uses the rental company many times, is important if done once only. |
| Quality |  | A bit better, same defects as in ASIS, except wrong keys |

Checkin

TBD

OVERALL comparison

|  |  |
| --- | --- |
| Service time (checkout, checkin) | Much better in TO BE  (possibly from 10-30 min to 1 min) |
| Quality | Similar, may be little improvement in TO BE |
| Unit cost (checkout, checkin) | If break even happens fast, then Unit cost decreases. It depends greatly on NC, NO (see TCO computation and analysis) |

# TCO (project point of view)

The TCO analysis considers the project point of view (not the single process point of view)

In this case we consider as project all activities to transition from AS IS to TO BE

Assumptions: the app is developed internally (no rented from external provider) (MAKE )

The devices to be mounted on cars are bought and not rented (BUY)

The cars are kept 5 years

Employees in office are not laid off, but office is closed and employees relocated to

other tasks

**Cost**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cost |  | Dir/indir | Fix /var | Opex / capex |
| Development of mobile app (client + server) | M\_dev | dir | Fix (does not depend on number of copies ) | capex |
| Maintenance of mobile app (client + server) | M\_maint | Dir | fix | opex |
| Mobile app deployment | M\_dep | dir | fix | opex |
| Training of employees previously in office | T | Indir | Var (in multiple of 20 employees, assuming classes of 20 employees) | opex |
| Development of back end – remote car control | RC\_dev | Dir | fix | capex |
| Remote control deployment | RC\_dep | dir | fix | capex |
| Maintenance of back end – remote car control | RC\_maint | dir | fix | opex |
| RC training cost | RC training | dir | Fix (assuming only one system admin is training) | opex |
| Device on car, selection cost | DC\_sel | dir | fix | opex |
| Device on car, cost of device + installation cost on car | DC | dir | var | capex |
| Maintenance of device | DC\_maint | dir | Var (for maintenance on each device on car)  + fixed part (new firmware and wireless update of firmware on cars) | opex |
| Operation cost (network bandwith, cpu, memory, electricity, conditioning) of mobile app, Remote control backend | OP = M\_op + RC\_op | dir | fix | opex |
| Dismount device from car (dismissal cost) | DC\_dismount | dir | var | opex |

Savings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Saving |  | Dir/indir | Fix /var | Opex / capex |
| Office cost  (rental , bills, copy machines) | O | Dir | Fix | Opex |
| Employees in office | E | Dir | Fix (long term contracts)  + var (short term contracts) | Opex |
| Key related cost (copies in case of loss) | K | dir | Var (ex 1% of rentals has a key loss) | opex |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Year | 1 | 2 | 3 | 4 | 5 |
| Cost |  | M\_dev | M\_maint | M\_maint | M\_maint | M\_maint |
|  |  | M\_dep |  |  |  |  |
|  |  | T |  |  |  |  |
|  |  | RC\_dev  RC\_dep  RC\_training | RC\_maint | RC\_maint | RC\_maint | RC\_maint |
|  |  | OP | OP | OP | OP | OP |
|  |  | DC\_sel |  |  |  |  |
|  |  | DC | DC\_maint | DC\_maint | DC\_maint | DC\_maint  DC\_dismount |
| Savings |  | O,E,K | O,E,K | O,E,K | O,E,K | O,E,K |
|  |  |  |  |  |  |  |

The table above should then be filled with actual cost figures.

Assuming M\_dev = 100Keuro

RC\_dev = 200Keuro

DC = 100 euro

NC= number of cars per location

NO = number of locations (or local agencies for rental)

O = 2k per month, 24k per year per location

E = 2k per month, 24k per year per location

Y = year

And, at least initially, not considering maintenance costs and operational costs, we have

|  |  |
| --- | --- |
| Costs | 100k + 300k + DC\*NC\*NO |
| Savings | (24k+24k) \* NO \* Y |

Break even happens when costs = savings

400k + DC \*(NC\*NO) = 48k \* NO \* Y

For a large rental company (NO = 100, NC= 30)

400k + 100\*(3000) = 48k \*100 \* Y

Y = 700k / 4.8 M

Break even happens very quickly

For a small rental company (NO =10, NC = 30)

400k + 100\*300 = 48k \*10 \* Y

Y = 430k / 480k

Break even happens at least after one year. But in this second case maintenance and operation costs should be considered for a more precise analysis

In general two extreme scenarios are possible

* Fast break even (1,2 years). In this case the change has no counter indications on the money side
* Slow (or no) break even (4 years or more). In this case the change must have other motivations beyond cost

# Comparison (by process, addendum)