**INITIAL NOTES**

**AS IS (solo se non cambia dal TO BE scrivi, altrimenti inutile)**

**TO BE**

FOOD sets up WebApp and employees for delivery.

R agrees with FOOD: FOOD collects a fee for delivery + 10%15% share of R bill

After agreement, R uploads on FOOD website its offer

Customer accesses FOOD website and selects from list R (then orders [address, credit card]); manager of R collects order from FOOD website (passes it to Kitchen and prepares packages).

Employee of FOOD collects packages and delivers to Customer (using FOOD App); then Customer can rate service on FOOD website and add a tip for Employee

**1.ORGANIZATIONAL MODEL**

FOOD

Employee (delivery)

R (Restaurant)

Manager

Kitchen

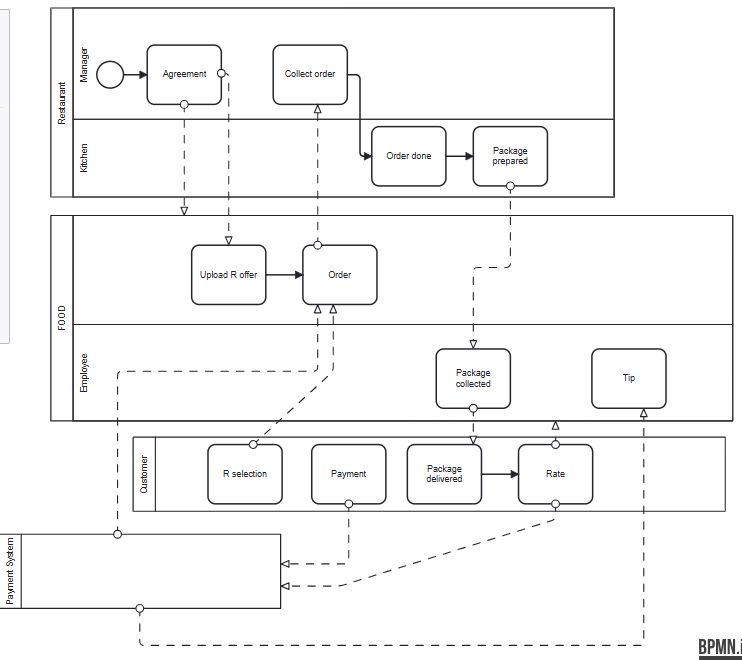
Customer

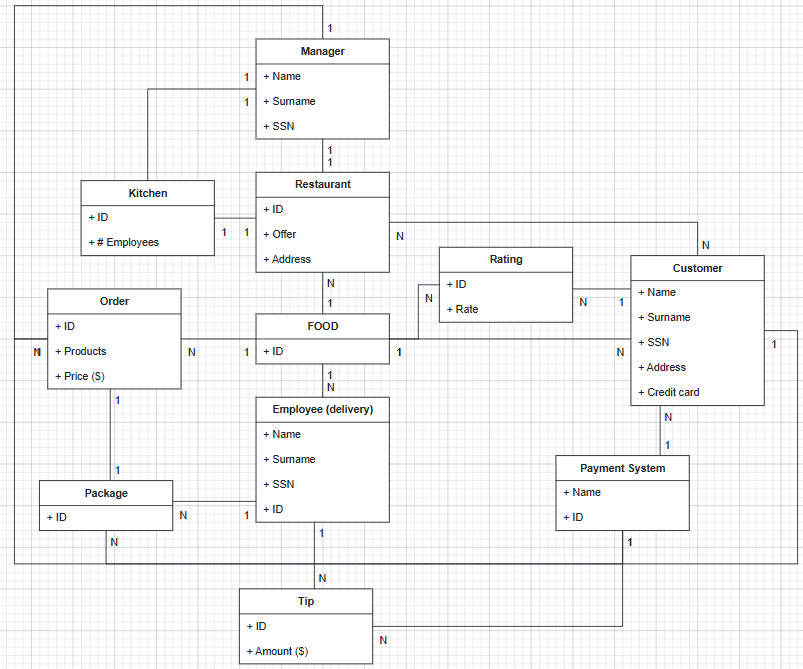
Payment system (because of credit card)

**2a.PROCESS TABLE (TO BE [+AS IS])**

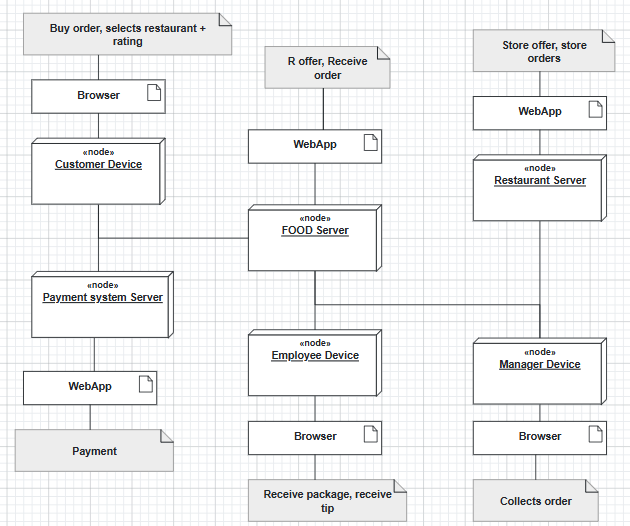
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NAME** | **INPUT** | **OUTPUT** | **DESCRIPTION** | **OU INVOLVED** |
| Agreement | Agreement | Agreement ok | R agrees with FOOD: FOOD collects a fee for delivery + 10%15% share of R bill | FOOD  R |
| Uploading offer | R’s offer | Uploading ok | After agreement, R uploads on FOOD website its offer | R  FOOD |
| Ordering | Customer order | Packages | Customer accesses FOOD website and selects from list R (then orders [address, credit card]); manager of R collects order from FOOD website (passes it to Kitchen and prepares packages). | Customer  FOOD  R (Manager + Kitchen)  Payment System |
| Delivery | Packages | Delivery done (+ eventual rating and tip) | Employee of FOOD collects packages and delivers to Customer (using FOOD App); then Customer can rate service on FOOD website and add a tip for Employee | Employee (delivery)  FOOD (App)  Customer  Payment System |

**2b.FUNCTIONAL MODEL (BPMN + UML class) of TO BE**





**3a.TECH MODEL (UML deployment) of TO BE**



**3b.BUSINESS RULE** = FOOD collects a fee for delivery + 10%15% share of R bill

**5.KPI** (considering these high-level business goals (or CSF): CSF1 increase customer satisfaction, CSF2 reduce the cost of the process for the restaurant R and company FOOD)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CSF**  **Name** | **KPI**  **Category** | **KPI**  **Name** | **KPI Description** | **Unit of measure** |
|  | **General** | N\_orders | Number of orders |  |
|  |  | N\_restaurants | Number of restaurants in FOOD DB |  |
| CSF2 | **Efficiency** | C\_order | Total orders’ cost/number of orders | Euro |
| CSF2 |  | C\_delivery | Total deliveries’ cost/number of deliveries | Euro |
| CSF1 | **Service** | LT\_order | Leading time from order begin to order prepared | t |
| CSF1 |  | LT\_delivery | Leading time from package retired to eventual tip + rating | T |
| CSF1 | **Quality** | Q | Non-conform orders/Total orders | % |
| CSF1 |  | Q\_Rating | Customer satisfaction = rating at the end of delivery | 1-5 |
| CSF1 |  | Q\_delivery | Order not delivered/total orders | % |

**6.COMPARISON AS-IS vs TO-BE using KPI**

|  |  |  |
| --- | --- | --- |
| **KPI** | **AS IS** | **TO BE** |
| N\_orders | Number of orders | More orders thanks to FOOD website |
| N\_restaurants |  | More restaurants available for the customer thanks to FOOD website |
| C\_order | Total orders’ cost/number of orders | More orders thanks to FOOD website, so less C\_order |
| C\_delivery | Total deliveries’ cost/number of deliveries | More deliveries thanks to FOOD website, so less C\_delivery; there is a specific FOOD employee to perform deliveries, so there are more people to prepare the order in restaurant |
| LT\_order | Leading time from order begin to order prepared | There is a specific FOOD employee to perform deliveries, so there are more people to prepare the order in restaurant, so less LT\_order; order are requested through phone, so less because of website |
| LT\_delivery |  | There is a specific FOOD employee to perform deliveries, so less LT\_delivery |
| Q | Non-conform orders/Total orders | Less, thanks to FOOD management |
| Q\_Rating |  | Customer can rate the order at the end |
| Q\_delivery | Order not delivered/total orders | There is a specific FOOD employee to perform deliveries, so less Q\_delivery |

**7.SOFTWARE FUNCTIONS TO BE**

|  |  |
| --- | --- |
| **PROCESS/ACTIVITY** | **SW FUNCTION(S) NEEDED** |
| Agreement | Agreement request (R Server)  Agreement done (FOOD Server) |
| Uploading offer | Send offer (R Server)  Receive offer and uploads it (FOOD Server) |
| Ordering | Access FOOD website (Customer Device)  Payment request (Customer Device)  Payment done (Payment System Server)  Collect order (Manager Device) |
| Delivery | Start delivery (Employee Device)  Receive order + send rating (Customer Device)  Receive rating (FOOD Server)  Send tip (Customer Device)  Tip done (PS Server)  Tip received (FOOD Server) |

**8.PROS & CONS of implementing TO BE**

|  |  |  |
| --- | --- | --- |
|  | **PROS** | **CONS** |
| FOOD | Now used, so work more, more salary | Cost of infrastructure |
| Employee (delivery) | Now works, so receive a salary to perform delivery |  |
| Restaurant | Now works more, thanks to more order received through FOOD website; now all employees can prepare the order because there is a FOOD employee to perform delivery | Not control on orders |
| Manager | Now works more thanks to more orders (as above) | Not control on orders and on infrastructure |
| Customer | Can access to more restaurants, can order through website and can rate |  |
| Payment System | Works more because all payments/tips pass through it | Cost of infrastructure |

**9.TCO**

|  |  |  |
| --- | --- | --- |
| **PHASE** | **COSTS** | **CAPEX or OPEX** |
| Construction  Selection | Developing WebApp  Developing infrastructure | CAPEX |
| Deployment | Installing WebApp  Training employees | Capex |
| Operation | Electricity  Internet  Send/receive data (also payments) | Opex |
| Maintenance | Device maintenance (Manager, Employee)  Server maintenance (FOOD, R, Payment System)  WebApp bug fixes | Opex |
| Dismissal | Dismiss  Data migration | Opex |

**10.ROI**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Year/Cost or Saving** | **Year 1** | **Year 2** | **Year 3** | **Year 4** | **Year 5** |
| **Cost** | Construction, Selection, Deployment | Operation, Maintenance | Operation, Maintenance | Operation, Maintenance | Operation, Maintenance |
| **Saving** | Less error on orders  More efficiency (more orders per year) | Less error on orders  More efficiency (more orders per year) | Less error on orders  More efficiency (more orders per year) | Less error on orders  More efficiency (more orders per year) | Less error on orders  More efficiency (more orders per year) |

**11.Outsourcing**

A manufacturing company has an engineering department. Here engineers use a program like Autocad (produced by AutoDesk) for doing technical drawings. The engineers use PCs owned and maintained by the company. The technical drawings are stored on a cloud storage service like Dropbox. Frame this case in terms of the outsourcing dimensions.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Object | **Activity/Service** | **Unicity** | **Location** |  |
| Autocad | Application | Shared | On site | Outsourced |
| PCs | IT infrastructure | Shared | On site | Insourced |
| Dropbox | Application/IT infrastructure (front-end + server) | Shared | Off site | Outsourced |

**DOMANDE:**

9)

10)

11)

12)

13)