



Live It Rich

Chef de Projet: Angélique Houdelier

Charles Fouco, Cédric Hulin, Calypso Petit, Léo Souquet



Contexte

- ◆ SI pour une chaîne d'hôtels de luxe avec:
 - ◆ Restaurants,
 - ◆ Bars,
 - ◆ Spa,
 - ◆ Golf,
 - ◆ Salles de conférence.
- ◆ Projet d'ACSI du S3: Octobre 2009 à Janvier 2009.
- ◆ Extension en projet tuteuré pour phase d'implémentation.



Cahier des charges

- ◆ Livré le 13 Octobre 2009, en début de projet.
- ◆ Trois axes:
 - ◆ Exigences académiques,
 - ◆ Exigences de gestion de projet,
 - ◆ Exigences fonctionnelles applicatives.



Éxigences académiques

- ◆ Approche d'analyse évènementielle. [Castellani et al., 1979]
- ◆ Méthode de conception classique par l'analyse de la dépendance fonctionnelle. [Cavarero, 1979]
- ◆ Représentation conceptuelle classique selon un modèle type E / A. [Chen, 1976]
- ◆ Méthode de conception objet utilisant l'algorithme de composition. [Cavarero et Lecat, 2000]
- ◆ Représentation conceptuelle objet selon UML ou un sous-ensemble approprié. [Jacobson;Booch;Rumbaugh, 1998] [Cavarero et Lecat, 2000]



Éxigences de projet

- ◆ Projet exclusivement mené en langue Anglaise.
- ◆ Un chef de projet désigné par ses pairs, unique responsable:
 - ◆ de la planification
 - ◆ de la communication avec la maîtrise d'ouvrage
- ◆ Utilisation d'un outil full-web de gestion de projet: *Zoho*
- ◆ Planification du projet selon la méthode *PERT* [Pocok et al., 1957]



Gestion des livrables

- ◆ **Livrables finaux:** processus en entonnoir de validation
Analyse → Conception → Implémentation
- ◆ **Livrables projet:**
 - ◆ Rapport d'activité hebdomadaire au chef de projet pour chaque membre ("internal reports", Thursdays COB)
 - ◆ Rapport d'activité hebdomadaire au client, transmis par le chef de projet ("external report", Fridays COB)

REPORTS



e It Rich Internal report

Week:

X

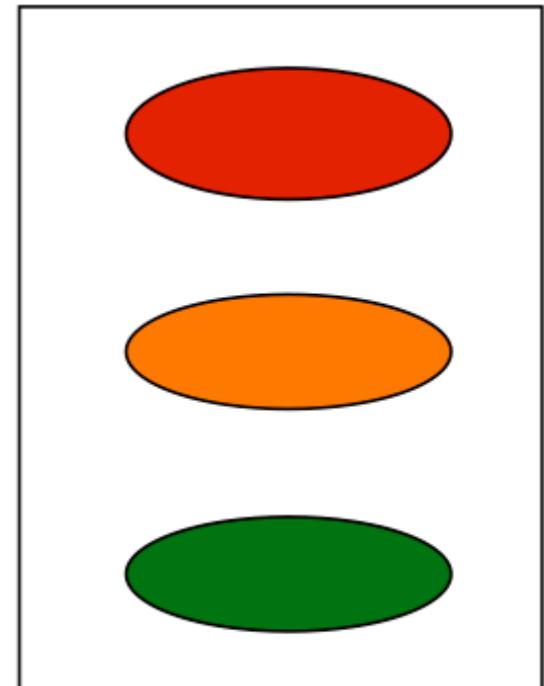
Approved by:

Angélique Houdelier, Project Manager

Date:

xx/xx/yyyy

Project status



Con



Rappel

- ◆ Analyse évènementielle:
- ◆ Diagramme de contexte (*optionnel*),
- ◆ Étude exhaustive des évènements externes au SI,
- ◆ Formalisation des actions élémentaires qui en découlent,
- ◆ Dictionnaire de données,
- ◆ Prototype non fonctionnel de la GUI.
- ◆ **Où sont formellement les besoins utilisateurs ?!**





Démarche d'analyse

- ◆ Sujet peu complexe au sens algorithmique ...
- ◆ Mais délicat pour l'intégration de l'ensemble des règles "métiers"
- ◆ Volonté d'aller "*plus loin*" que l'analyse évènementielle:
prise en compte des *requirements* [Colland; Pastor; Cavarero, 2008]
- ◆ Problème: Comment les rendre actionnables ?



Requirements

- ◆ Un modèle à cinq types a été construit:
- ◆ T1 = **Time and synchronisation**
- ◆ T2 = **Network and physical space**
- ◆ T3 = **Usage and safety of human ressources**
- ◆ T4 = **Storage**
- ◆ T5 = **Links between concepts**



Notre modèle

- ◆ Concepts simples et précis.
- ◆ Permettent aisément de **classifier les besoins** du CDC et des interviews avec le client.
- ◆ Aboutissent à un formalisme aidant à:
 - ◆ La **découverte** des évènements ($T1, T2$ et $T3$)
 - ◆ L'**analyse** des dépendances fonctionnelles ($T5$)
 - ◆ L'**intégration** de ces contraintes en conception, dans les modèles fonctionnels ($T1, T2, T3$ et $T4$)



Notre démarche d'analyse

- ◆ **Identification des domaines du système**
- ◆ Pour chaque domaine:
 - ◆ Formalisation des requirements
 - ◆ Diagramme de contexte
 - ◆ Évènements / Actions
 - ◆ En parallèle: dictionnaire de données et GUI (*non fonctionnelle*)



Le “Quoi?” puis le “Comment?”

- ◆ La conception se devrait d'être moins problématique.
- ◆ Un choix clair entre le **classique** ou l'**objet**.
- ◆ Des **méthodes** pour chaque paradigme: LAPAGE, algorithme de composition.
- ◆ Pourtant, un état de fait: un système créé de zéro en 2010 a tout intérêt à:
 - ◆ Être implémenté avec un LOO (*bibliothèques de classes*)
→ *Donc conçu en objet*
 - ◆ Faire reposer son “cœur métier” dans un SGBDR (*performances, SQL*)
→ *Donc conçu en classique !*



Les faits

- ◆ Le choix des SGBDR est **sans appel**:
Top 10 aux Benchmarks C du *Transaction Performance Council*
 - Entre 1,3 et 7,6 millions de transactions par minute (*Oracle, DB2*).
- ◆ Mais aussi: principe de la "*locality of reference*" [Denning; Schwartz, 1972]
 - Faire exécuter des traitements au plus proche de leur données sous-jacentes.
- ◆ De même, rapidité de construction d'un formulaire à quatre boutons et six listes déroulantes (*ex. environnement Microsoft Windows*):
 - en C AINSI = 1/2 à 1 journée homme
 - en Java ou .Net = 5 à 10 min



Nos choix

- ◆ **Respecter** les faits et études précédemment cités, selon **trois principes**
- ◆ **1^{er} principe:** Le “coeur métier” sera conçu en classique, avec une méthode permettant:
 - la normalisation des données selon les règles de [Codd, 1970] [Codd, 1975]
 - la transformation de ses modèles en structures relationnelles.
- ◆ **2^{ème} principe:** Tout traitement “métier” sera conceptualisé en classique, autour du “cœur métier” selon [Denning; Schwartz, 1972].
Corolaire: L'utilisation des concepts objets pour ces traitements sera évitée.
- ◆ **3^{ème} principe:** Tout élément d'interface graphique et de communication inter-système sera conçu en objet (*pour une implémentation idoine avec bibliothèques*).



Exigences fonctionnelles

- ◆ Le système doit gérer de manière informatique ces sept domaines:
 - ◆ Les chambres (*jusqu'à 50 par hôtel*),
 - ◆ Les bars (*3 par hôtel*),
 - ◆ Les restaurants (*un "Grill" et un "Gastronomique"*),
 - ◆ Le court de golf (*9 trous*),
 - ◆ Le spa,
 - ◆ La ou les salles de conférences,
 - ◆ La gestion administrative de l'hôtel, particulièrement son organisation et les droits du personnel.

Analyse: Chambres

◆ Requirements:

Trois de *temps et synchronisation*: “Customers must check out of room by 11:30 pm”

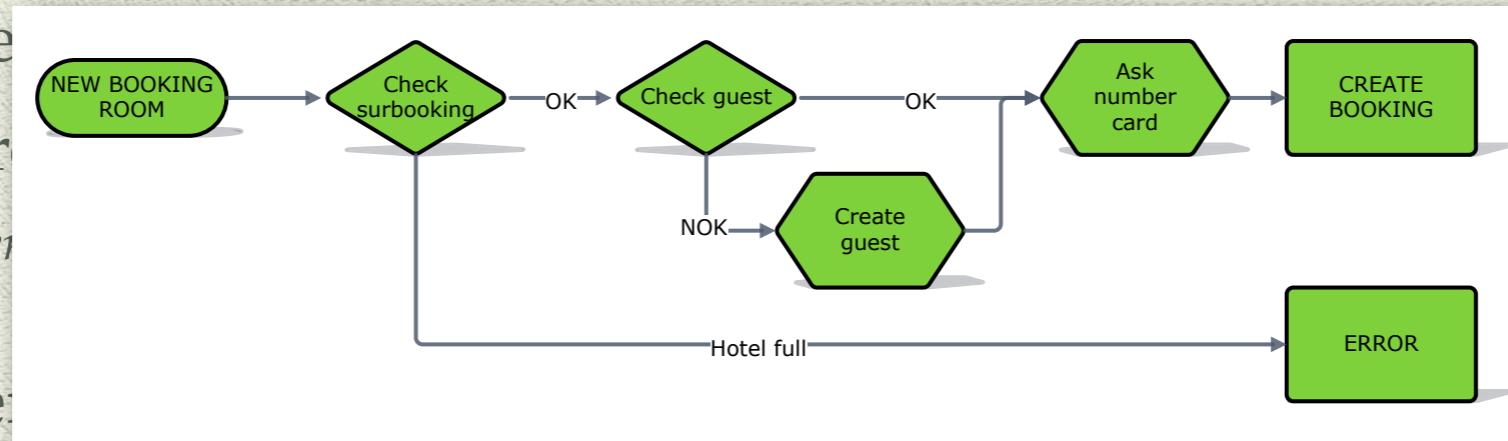
Trois de *réseau et espace physique*: “Safes are network-connectable.”

Un de

Quatr

coded and ide

◆ Évèneme



headquarters' office”

service kit) in a room are

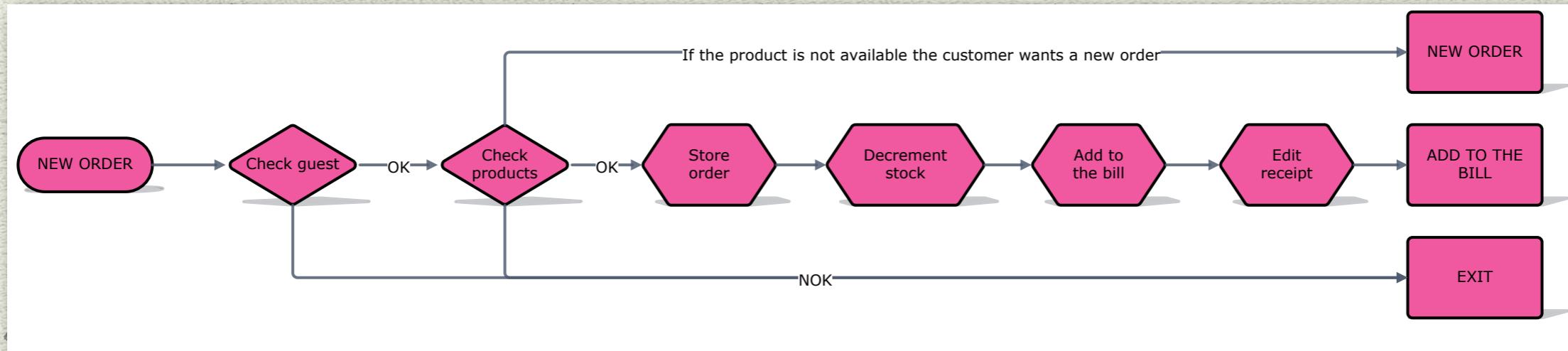
◆ Dictionnaire de données: 38 attributs

◆ Écrans:

Analyse: Bars

◆ Requirements:

Deux de réseau et espace physique: “There is always three bars in the hôtel: one in the lobby, one in the golf house, one in the spa.”



◆ Dictionnaire de données: 5 attributs (*sans compter les produits vendus*)

◆ Écrans:

Analyse: Restaurants

◆ Requirements:

Neuf de *temps et synchronisation*: “If the customer doesn’t come (no show) we release the reserved table. see SYNC-022 for the no-show period”

Trois

stage the communication of

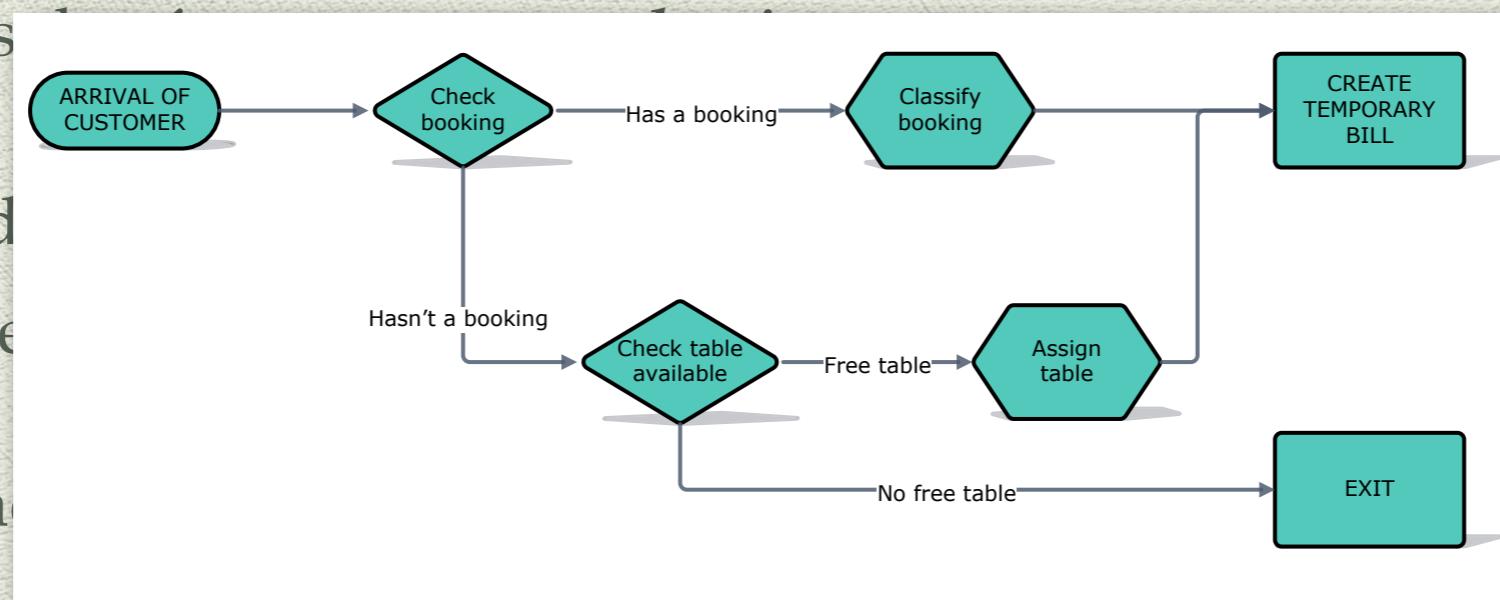
Un d

restaurants’ tables”

Onze

unique table number”

◆ Évènements:



◆ Dictionnaire de données: 26 attributs (*sans compter les produits vendus*)

◆ Écrans:

Analyse: Golf

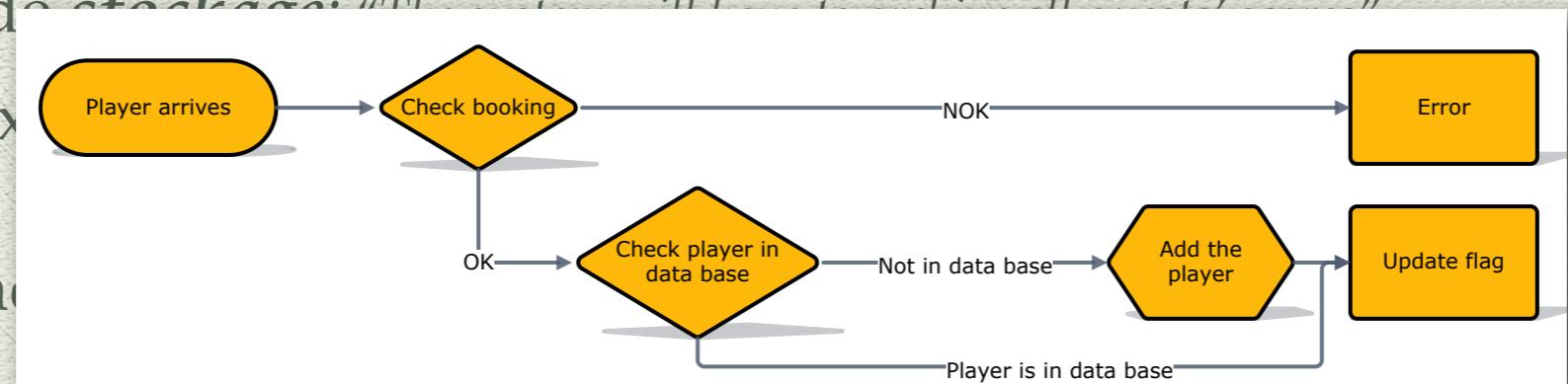
◆ Requirements:

Un de *temps et synchronisation*: “All greens need to be booked as they can hold only a certain number of guests at the same time.”

Un de *stockage*: “The system will keep a list of all the players.”

Deux

◆ Évènements:



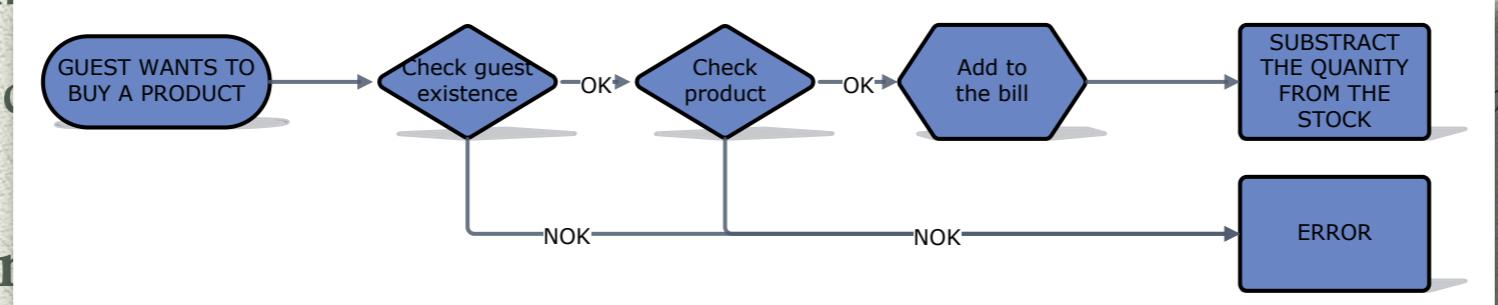
◆ Dictionnaire de données: 8 attributs

◆ Écrans:

Analyse: Spa

◆ Requirements:

Deux cas de figure : "Guest wants to buy a product" et "Guest wants to increase its purchase price"



◆ Évènements:

◆ Dictionnaire de données: 36 attributs

◆ Écrans:



Analyse: Conférences

- ◆ À savoir: le système ne gère pas l'organisation interne de la conférence
- ◆ Requirements:
Un liens entre concepts: "With regards to a booking of a conference room, the payment is addressed to organise ..."
- ◆ Évènements/Actions: aucun en particulier, les évènements concernent la réservation de chambres particulières (les salles) ainsi que les autres produits de l'hôtel
- ◆ Dictionnaire de données: 15 attributs

Analyse: Organisation générale

◆ Requirements:

Vingt de *temps et synchronisation*: "If there is a possible extension of stay but the current customer's room was booked by another guest, the system will offer a new room."

Un de *sécurité et utilisation des ressources humaines*: "The restaurant, the spa and the golf have an employee in charge of dealing with the bookings."

Quatre de *réseau et espace physique*: "When a guest calls the receptionist, records are displayed on the receptionist's PC number"

Un de *stockage*: "The system will store any produced bill less than 10 years old."

Vingt de *liens entre concepts*: "They are usually 4 rooms (standard, double, junior suite and with balcony, junior suite and suite) charged according rates

```

graph LR
    A([BOOKING SERVICE MODIFICATION]) --> B{Booking existence}
    B --> C{Hourly demand}
    C --> D{Check free place}
    D -- OK --> E[Modification the receipt]
    E --> F[SEND THE RECEIPT]
    E --> G[EXIT]
    D -- Cancel booking --> H[CANCEL BOOKING]
    D -- Cancel modification --> I[CANCEL MODIFICATION]
    D -- Booking non-existent --> J[NEW BOOKING]
    
```

◆ Évènements/Actions: 37 évènements, 189 actions

◆ Dictionnaire de données: 105 attributs

◆ Écrans:

Analyse: Gestion du personnel

◆ Requirements:

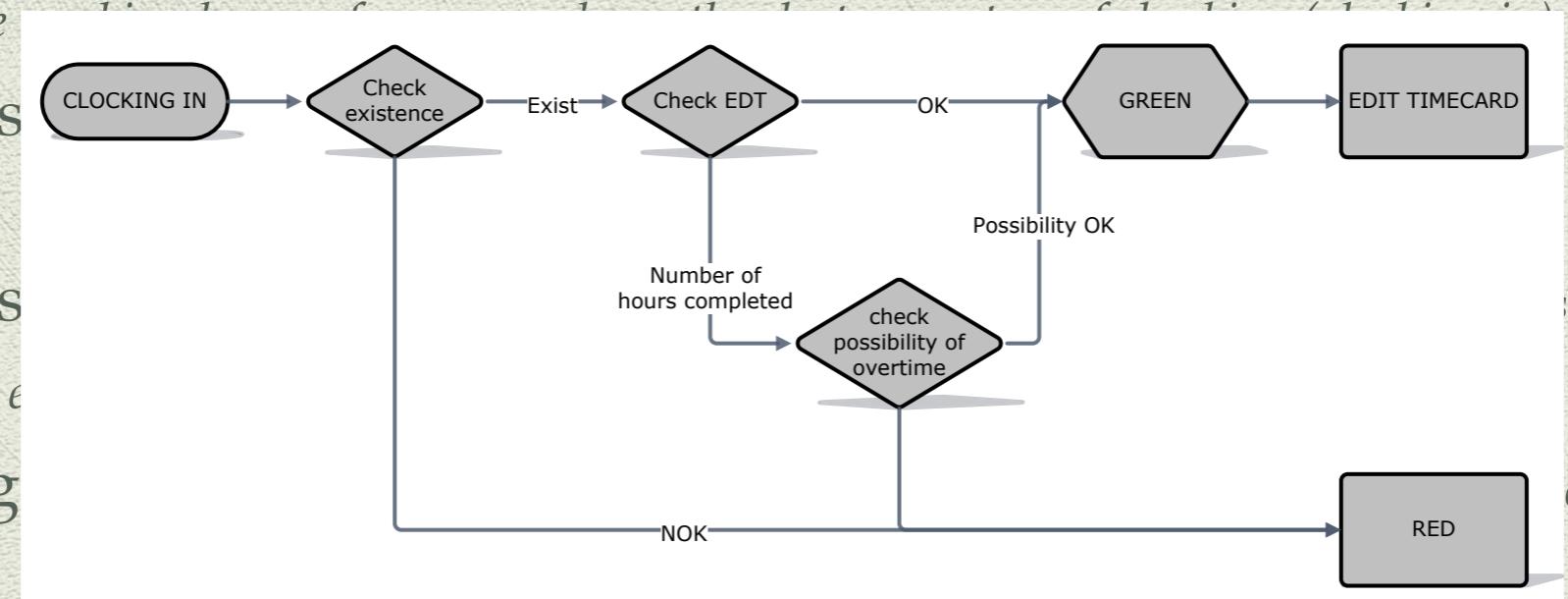
Six de temps et synchronisation: “The staff of the swimming pool will have to open of the service at 8 am and will have to close the service at 10 pm.”

Dix de sécurité et utilisation des ressources humaines: “The system will count (record) the number of hours worked by each employee and calculate the salary according to the number of hours worked and the rate of pay assigned to each employee.”

Trois
bedrooms”

Trois
and former e

Ving



◆ Évènements/Actions: 10 évènements, 59 actions

◆ Dictionnaire de données: 55 attributs

◆ Écrans:



La conception

- ◆ Le dossier d'analyse fut approuvé le 13/12/2009 par la maîtrise d'ouvrage.
- ◆ Le processus de conception s'est alors déroulé en:
 - ◆ Création du modèle statique avec l'algorithme LAPAGE, aidé des requirements T5 → production d'un schéma Entité / Association
 - ◆ Réalisation des graphes de fonctionnement de chaque évènement,
(modifiant si nécessaire le modèle de données 100% BCNF d'origine)
 - ◆ Ecriture des algorithmes formels pour la réponse aux évènements d'interface graphique.

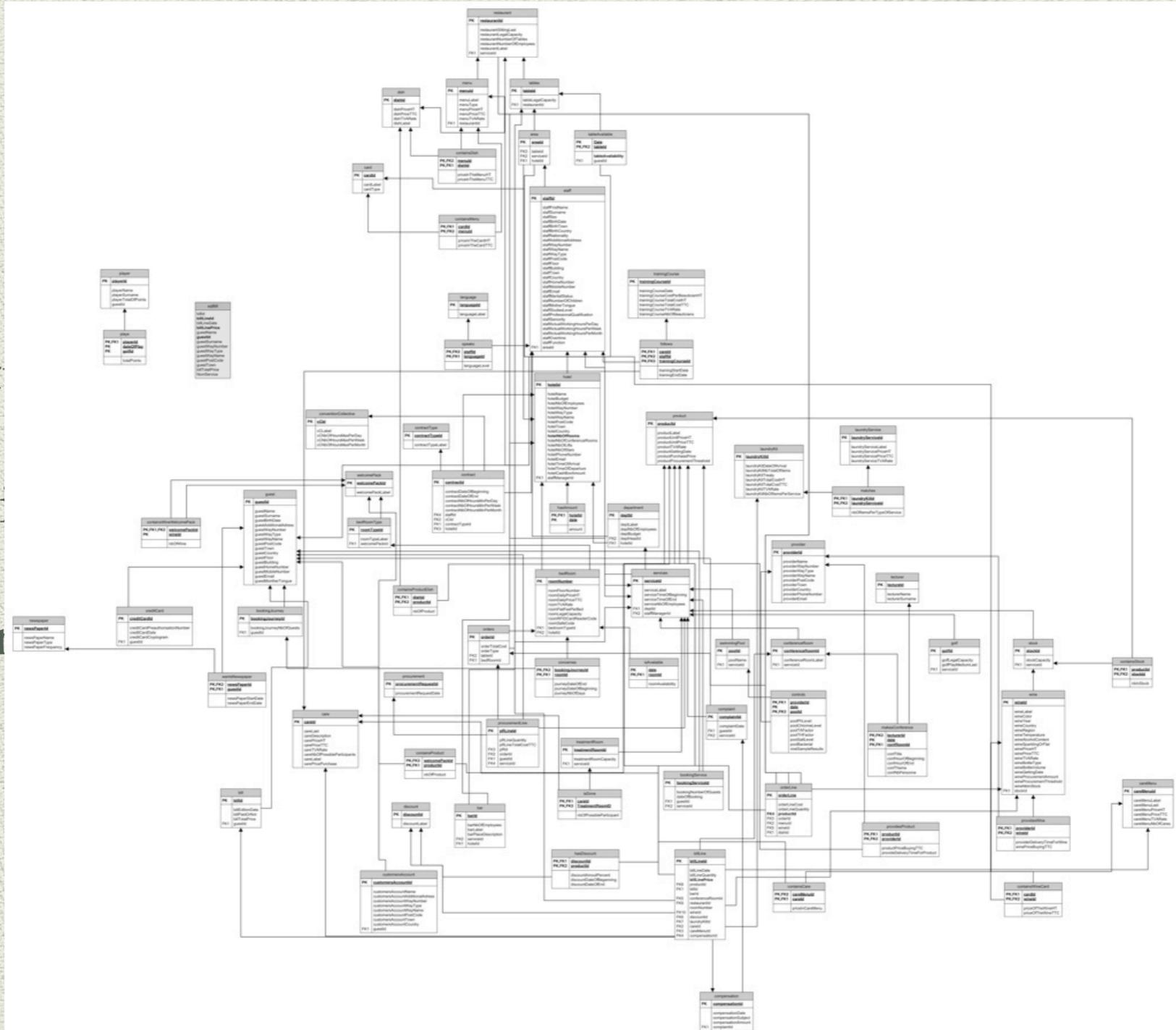
Modèle statique



◆ 338 At

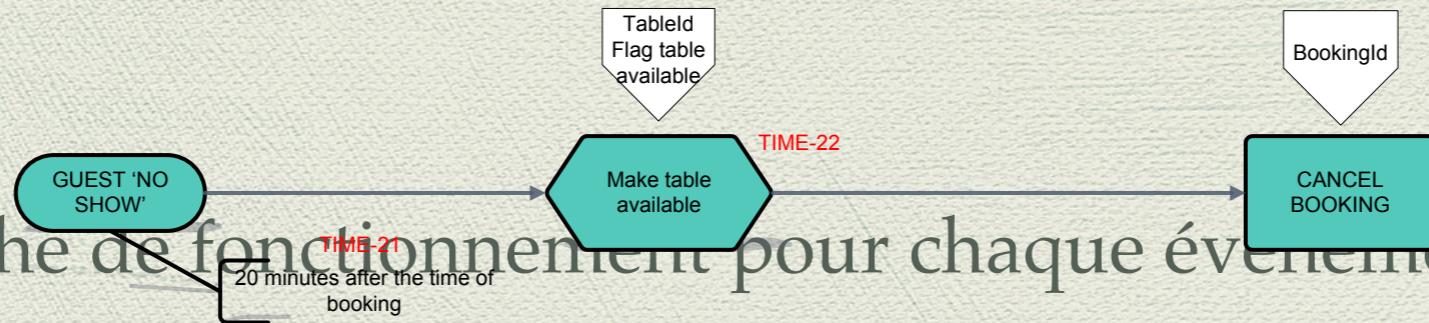
72 Ent

◆ 98 Idee

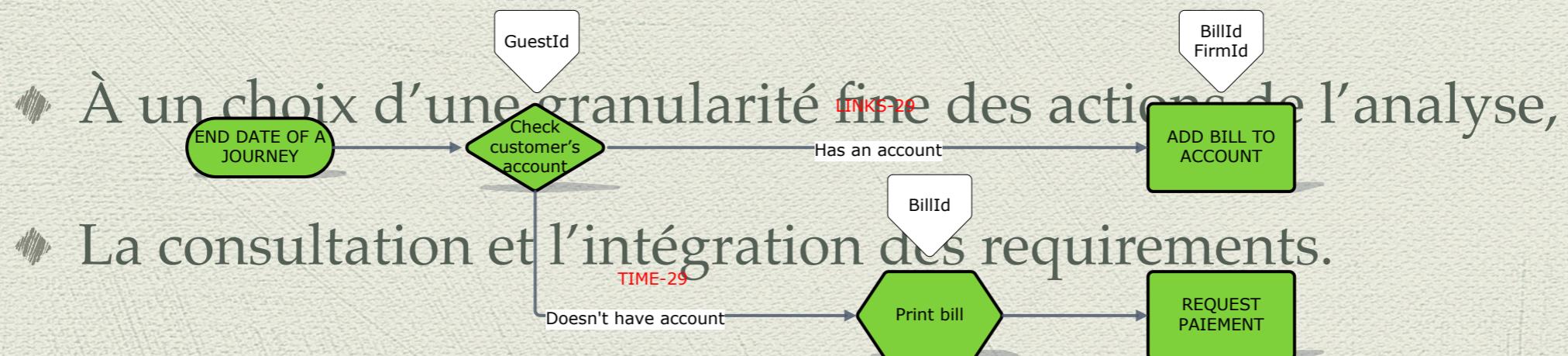


Modèle fonctionnel

- ◆ Un graphe de fonctionnement pour chaque événement externe.



- ◆ Réalisation facilitée grâce:





GUI: Diagramme de classes

- ◆ 44 Classes.
- ◆ Aucune ne contient d'algorithme issu de la “logique métier”
→ fan-in, fan-out limités.
- ◆ Elles ne font que répondre aux évènements souris et clavier de l'utilisateur.
- ◆ Deux classes particulières:
 - ◆ la DBConnector: permet le tunnel de connexion avec le monde relationnel,
 - ◆ la CrystalReport: permet d'intégrer le moteur SAP Crystal Report SDK pour la génération d'états.

GUI: Algorithmes

◆ Utilisation d'opérations algorithmiques

◆ Variables, Constantes

◆ Primitives

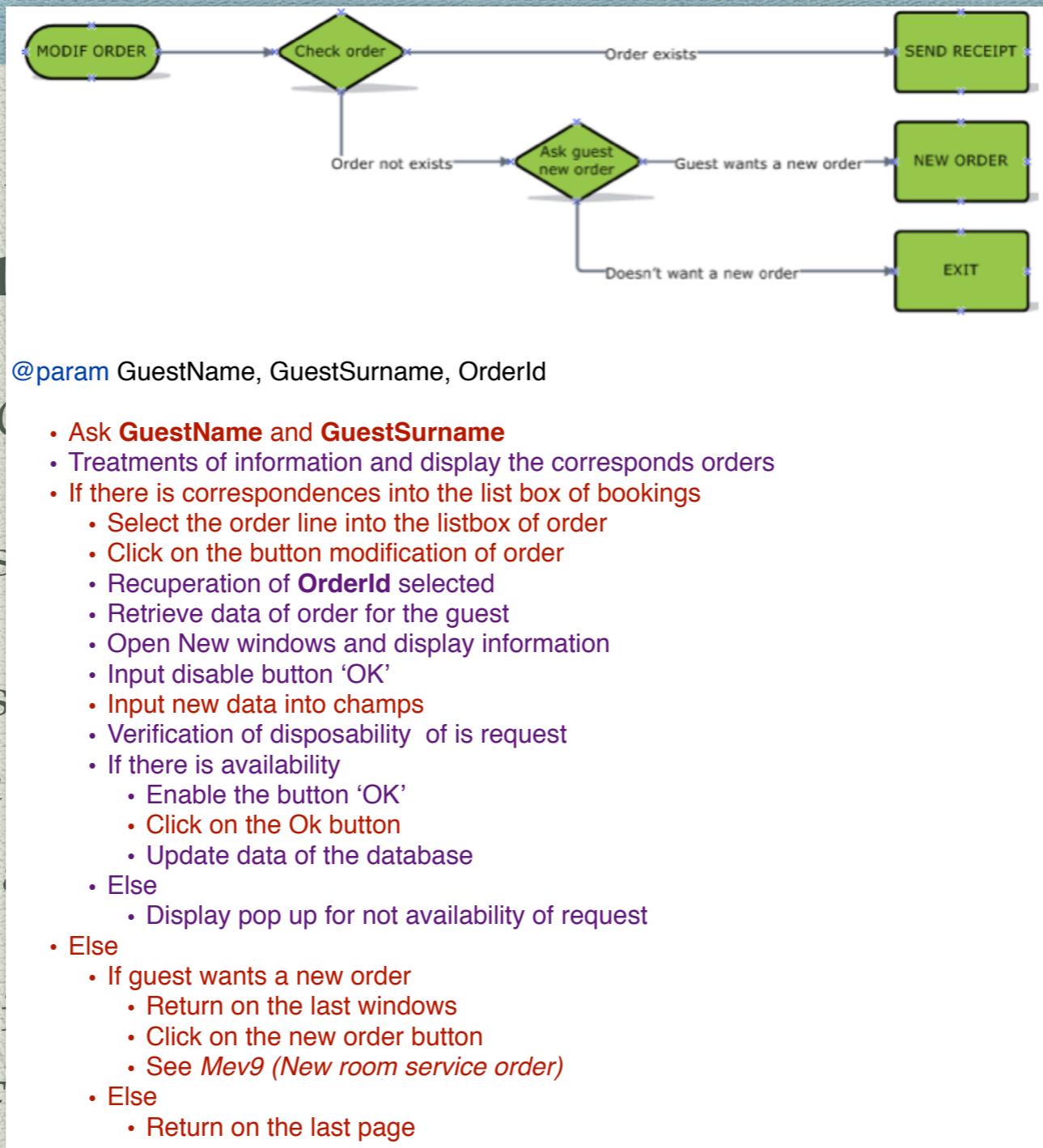
◆ Structures

If ... Then

Select (<view>)

◆ Structure de boucles

Boucles For





Implémentation: SGDBR

- ◆ Le modèle E/A obtenu s'est transformé, **selon les règles usuelles**, en:
 - ◆ 72 tables,
 - ◆ 439 attributs (*avec la redondance des clés étrangères*),
 - ◆ 98 clés primaires,
 - ◆ 101 clés étrangères.
- ◆ SGBDR retenu: SQL Server 2008.
- ◆ 4 tables ont été dé-normalisées pour **satisfaire les traitements**.
- ◆ Des index furent placés sur certains attributs pour lesquels les traitements provoquaient des "*full table scan*".



Implémentation: logique métier

- ◆ Tous les graphes de fonctionnement ont été transformés en:
 - ◆ Requêtes SQL,
 - ◆ Procédures stockées et/ou trigger si nécessaire (T-SQL).
- ◆ Comme prévu, aucun élément de la logique métier dans le code objet.



Implémentation: GUI

- ◆ Utilisation du Framework Microsoft™ .Net® 3.5
- ◆ IDE Visual Studio 2008
- ◆ Langage Visual Basic .Net
- ◆ Framework OLE DB .Net pour la connection à la base SQL Server
- ◆ Framework SAP Crystal Report pour la génération d'états



Conclusion

- ◆ Un projet globalement finalisé
- ◆ Approfondissement de connaissances dans :
 - ◆ la programmation
 - ◆ les bases de données
 - ◆ le reporting
- ◆ Améliorations possibles



Merci pour votre attention

Des questions ?