

# **Devoir 2 – SEG 3502**



**SEG 3502- Architecture et conception logiciel**

**Université d'Ottawa**

**Professeur : Stephane Some**

**Noms et numéros des étudiants :**

Gbegbe Decaho Jacques 300094197  
Nono Wandji Jordan Steve 300277480

Date de soumission: 20 October 2024

## Devoir 2

Remarque : ce devoir vaut 2/3 de la note des devoirs (le devoir 1 vaut 1/3).

Ce devoir peut être soumis en équipe de 2 maximum. Une seule soumission par équipe. Assurez-vous d'identifier tous les auteurs sur la page de couverture de la soumission en fournissant les noms et numéros des étudiants.

Toute soumission dont vous n'êtes pas l'auteur constitue une violation de l'intégrité académique. Cela inclut les solutions fournies par tout tiers, y compris celles générées par les systèmes d'intelligence artificielle.

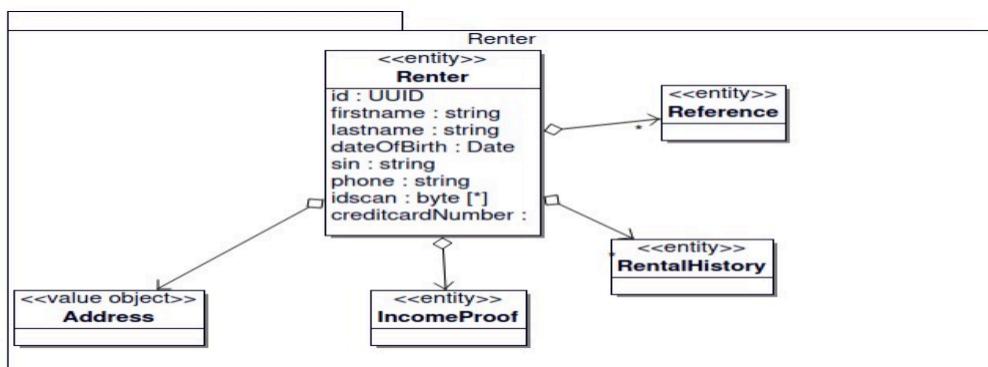
Tous les diagrammes doivent être réalisés à l'aide d'outils de modélisation.

### Problème

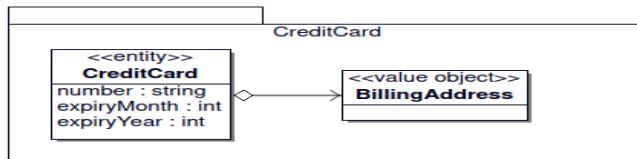
Le travail demandé dans ce devoir s'inscrit dans la continuité du devoir précédent sur le Système d'Application de Location en Ligne/Online Rental Application System (SALO/ORAS).

Les sous-domaines suivants ont été identifiés pour l'application.

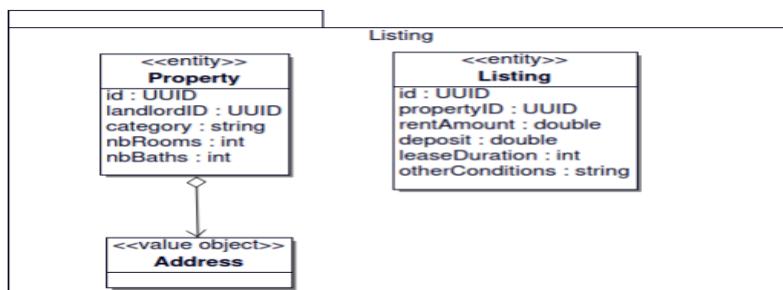
- Domaine Renter



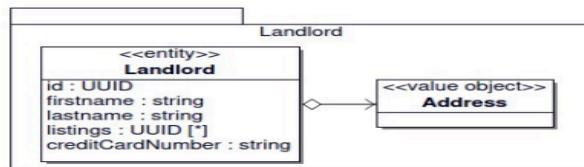
- Domaine Credit Card



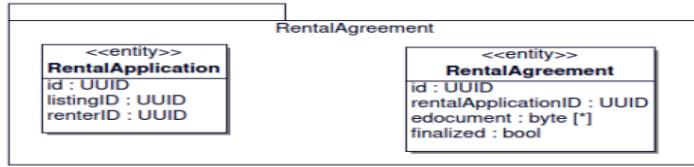
- Domaine Listing



- Domaine Landlord



- Domaine Rental Agreement



Voici les contrats de commandes d'applications identifiés à partir du cas d'utilisation *Create Rental Agreement*.

Étape 3 : Créer une demande de location

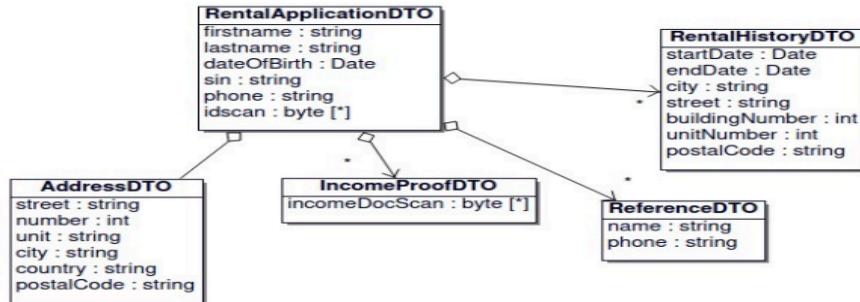
**Command:** createRentalApplication(rentalApplicationInfo : RentalApplicationDTO, plistId, renterId)

**Preconditions:**

- there is an existing PropertyListing *plist* ided by *plistId*
- there is an existing Renter *renter* ided by *renterId*

**Postconditions:**

- *renter* properties (personal information, references, rental history, proof of income, address) were updated based on rentalApplicationInfo (**modification of attributes**)
- a new RentalApplication *rap* was created (**creation of instance**)
- *rap* was associated to *renter* (**creation of association**)
- *rap* was associated to *plist* (**creation of association**)



Étape 7 : Générer un contrat de location

**Command:** generateRentalAgreement(rentalApplicationId)

**Preconditions :**

- there is an existing open RentalApplication *rentap* ided by *rentalApplicationId*

**Postconditions:**

- a new RentalAgreement *rag* was created (**creation of instance**)
- *rag* was associated to *rentap* (**creation of association**)
- a e-document was attached to *rag* (**modification of attribute**)

Étape 12 : Spécifier la carte de crédit du locataire

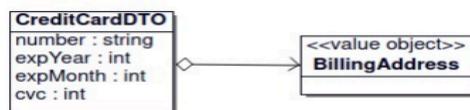
**Command:** setRenterCreditCard(renterId, creditCardInfo : CreditCardDTO)

**Preconditions :**

- there is an existing Renter *renter* ided by *renterId*

**Postconditions :**

- a new CreditCard *cc* created (**creation of instance**)
- *cc* was initialized from *creditCardInfo* (**modification of attributes**)
- *cc* was associated to *renter* (**creation of association**)



Étape 17 : Finaliser le contrat de location

**Command:** finalizeRentalAgreement(rentalAgreementId)

**Preconditions :**

- there is an existing RentalAgreement *rentagr* ided by *rentalAgreementId*
- *rentagr* was marked as completed (**modification of attribute**)

### **Questions**

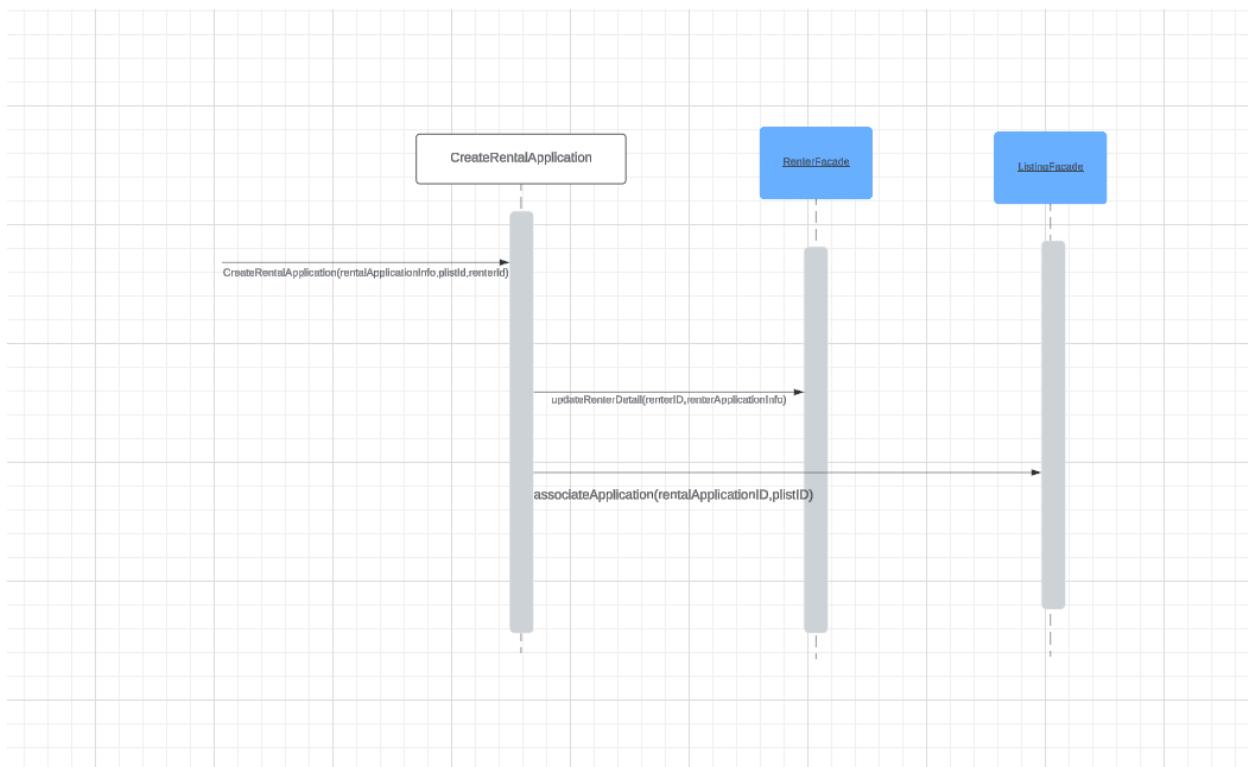
- i. Fournissez des Réalisations de Cas d'Utilisations de niveau 1 pour les commandes d'applications: `createRentalApplication`, `generateRentalAgreement`, `setRenterCreditCard` and `finalizeRentalAgreement`
- ii. Fournissez des Réalisations de Cas d'Utilisations aux niveau 2 pour toutes les responsabilités identifiées dans les réalisations de cas d'utilisation de niveau 1 créées en i.

Modéliser vos Réalisations de Cas d'Utilisations en utilisant des diagrammes de Séquences UML.

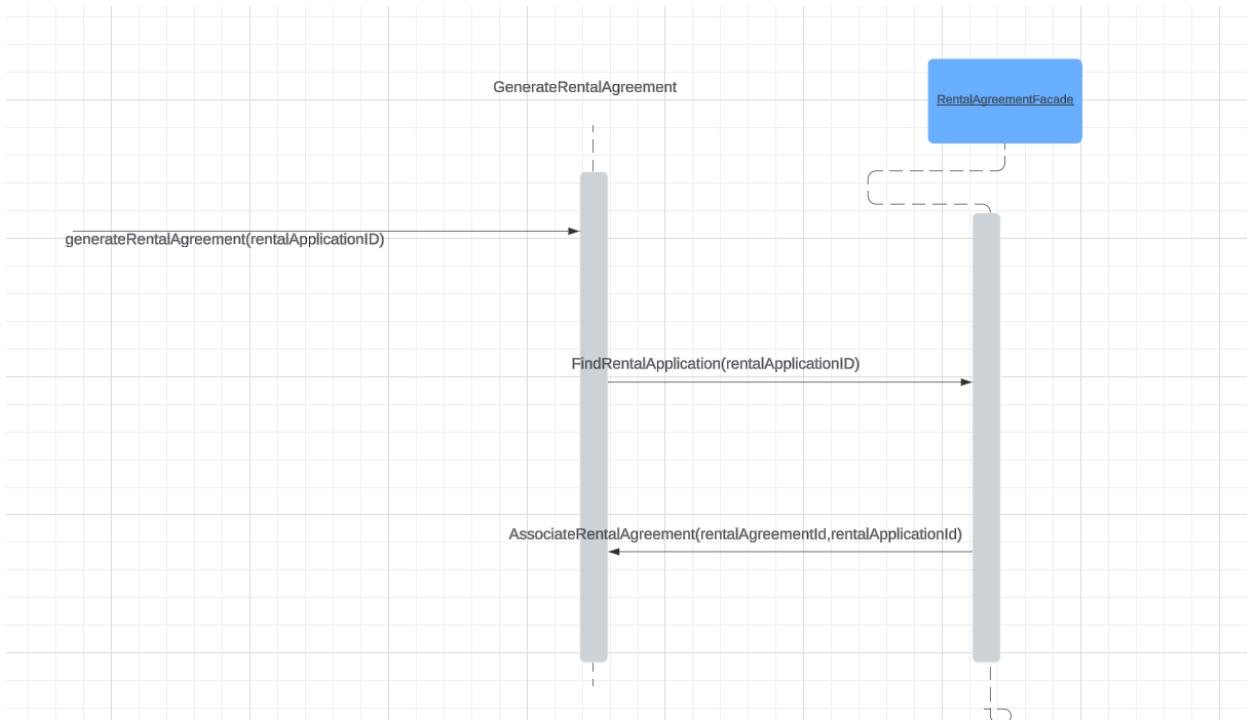
### **Résolutions des Questions**

#### NIVEAUX 1:

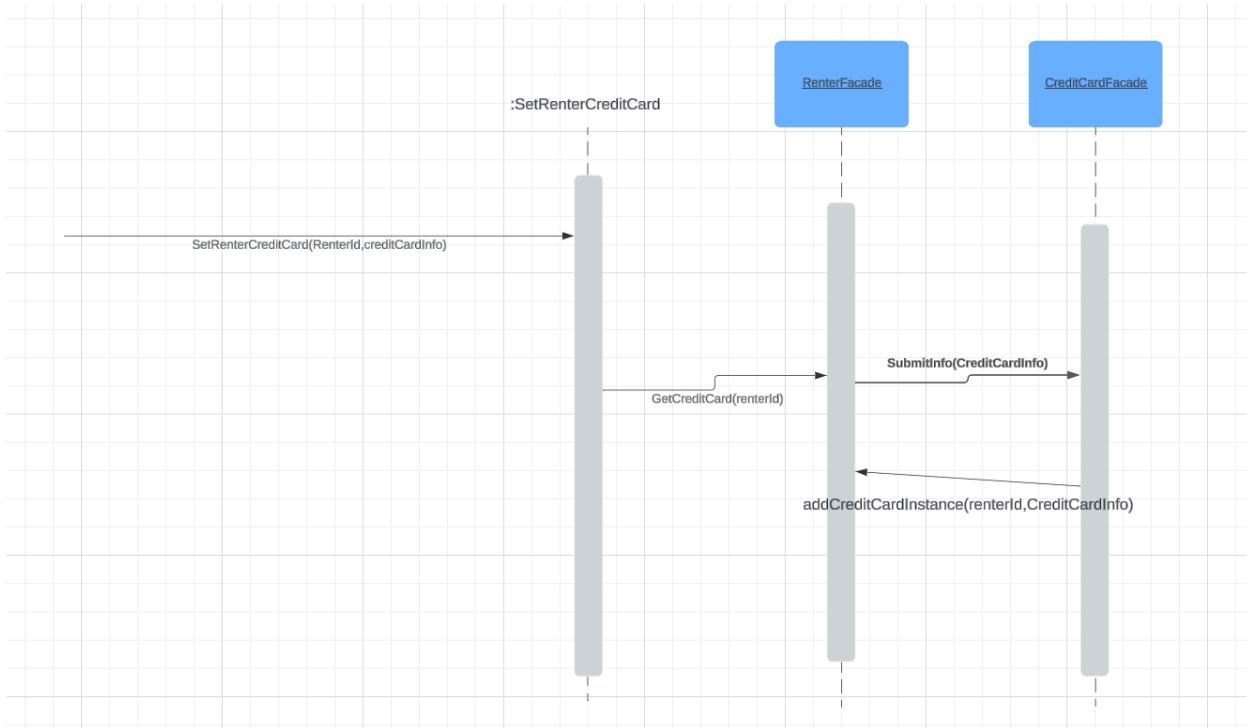
##### °CreateRentalApplication:



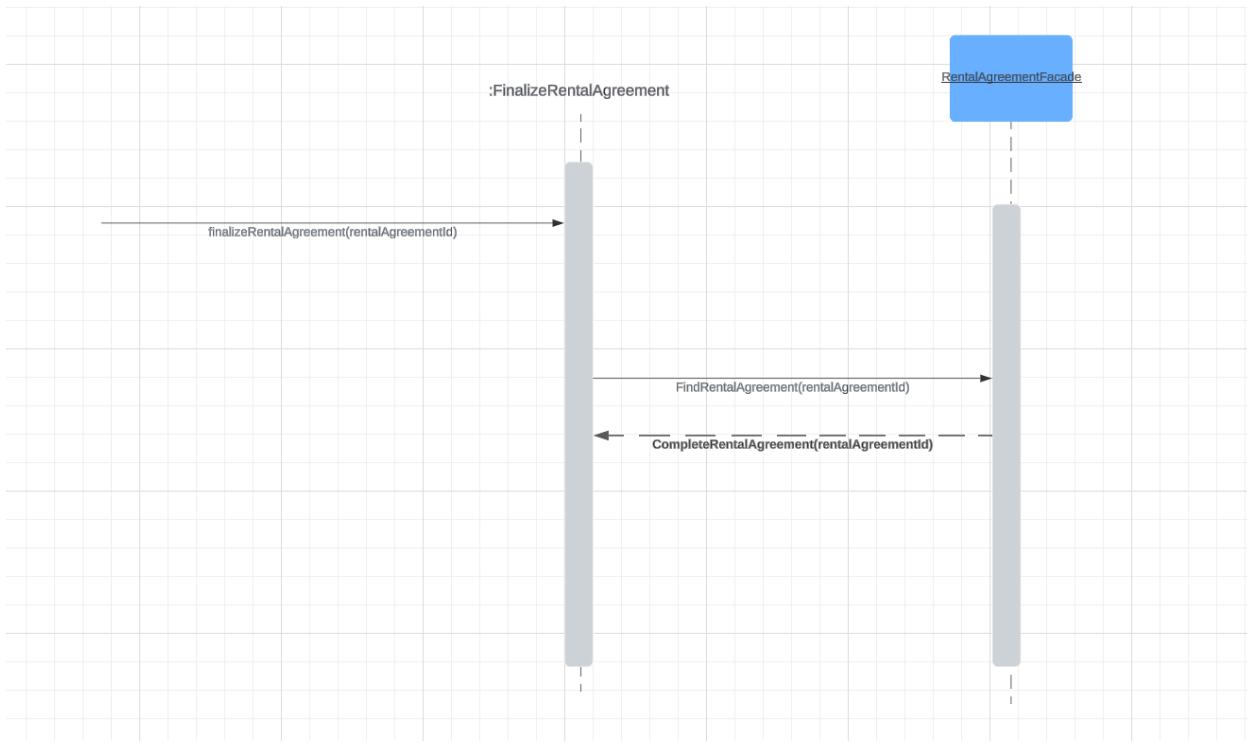
##### °generateRentalAgreement:



### °SetRenterCreditCard

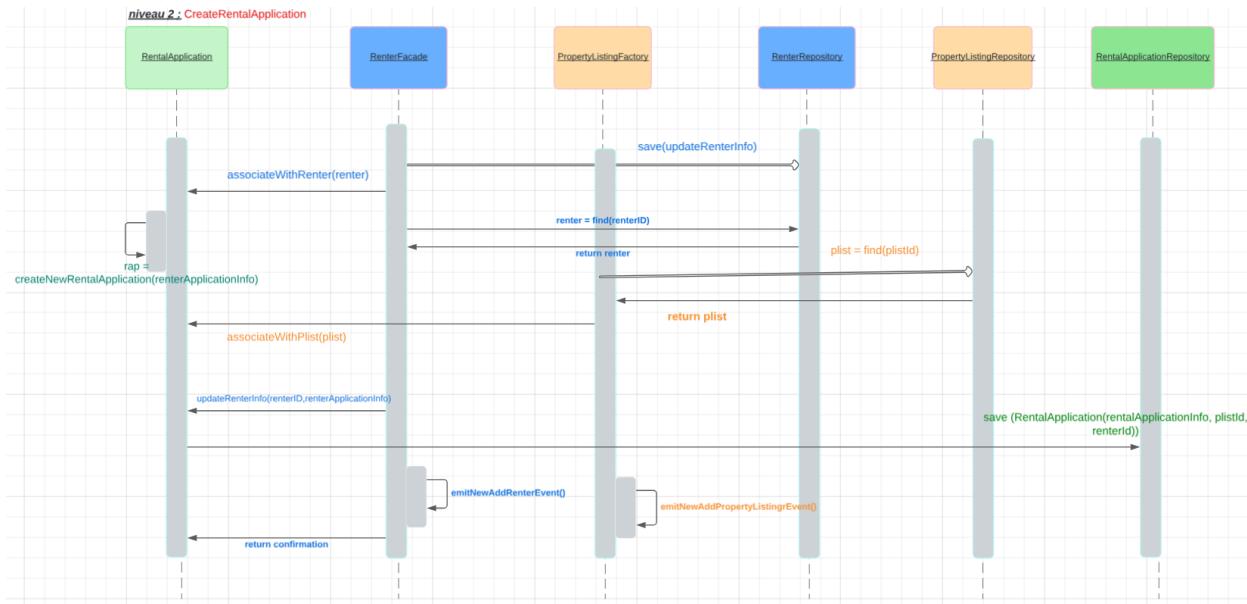


### °finalizeRentalAgreement

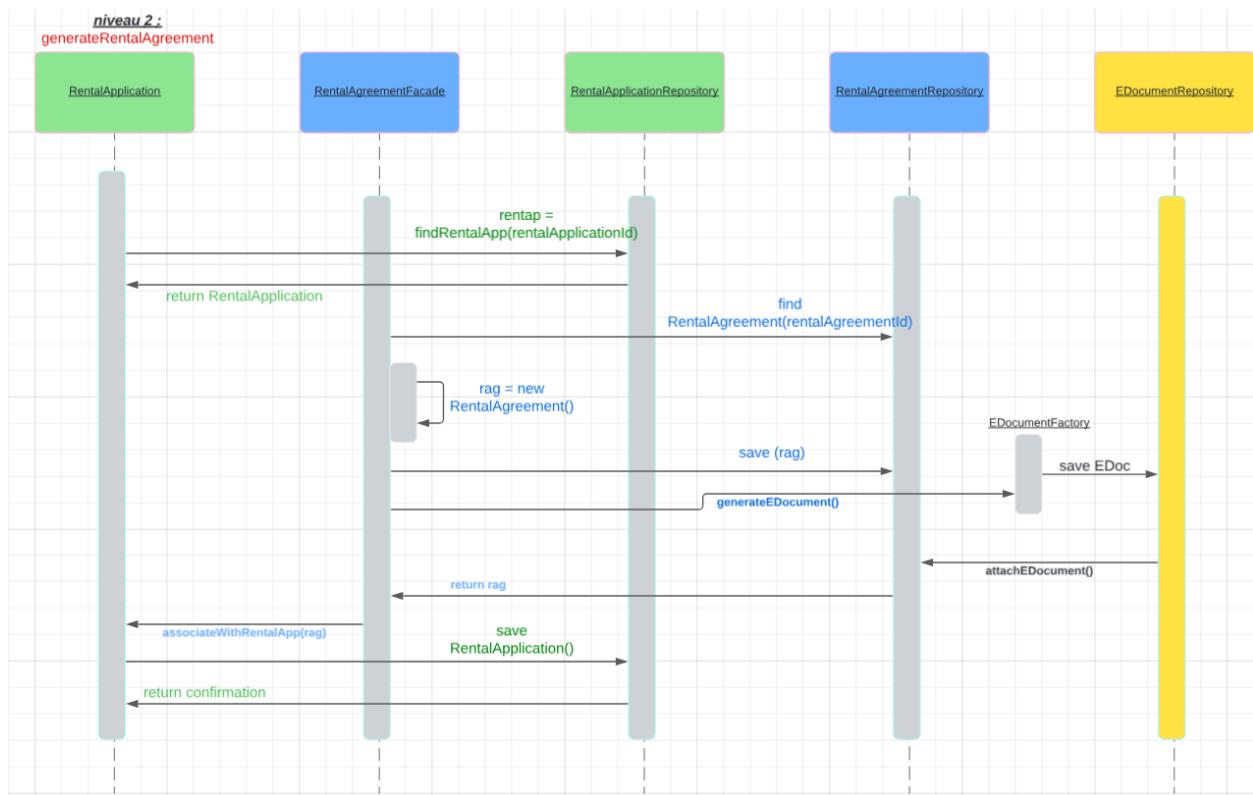


## NIVEAUX 2 :

### °CreateRentalApplication:

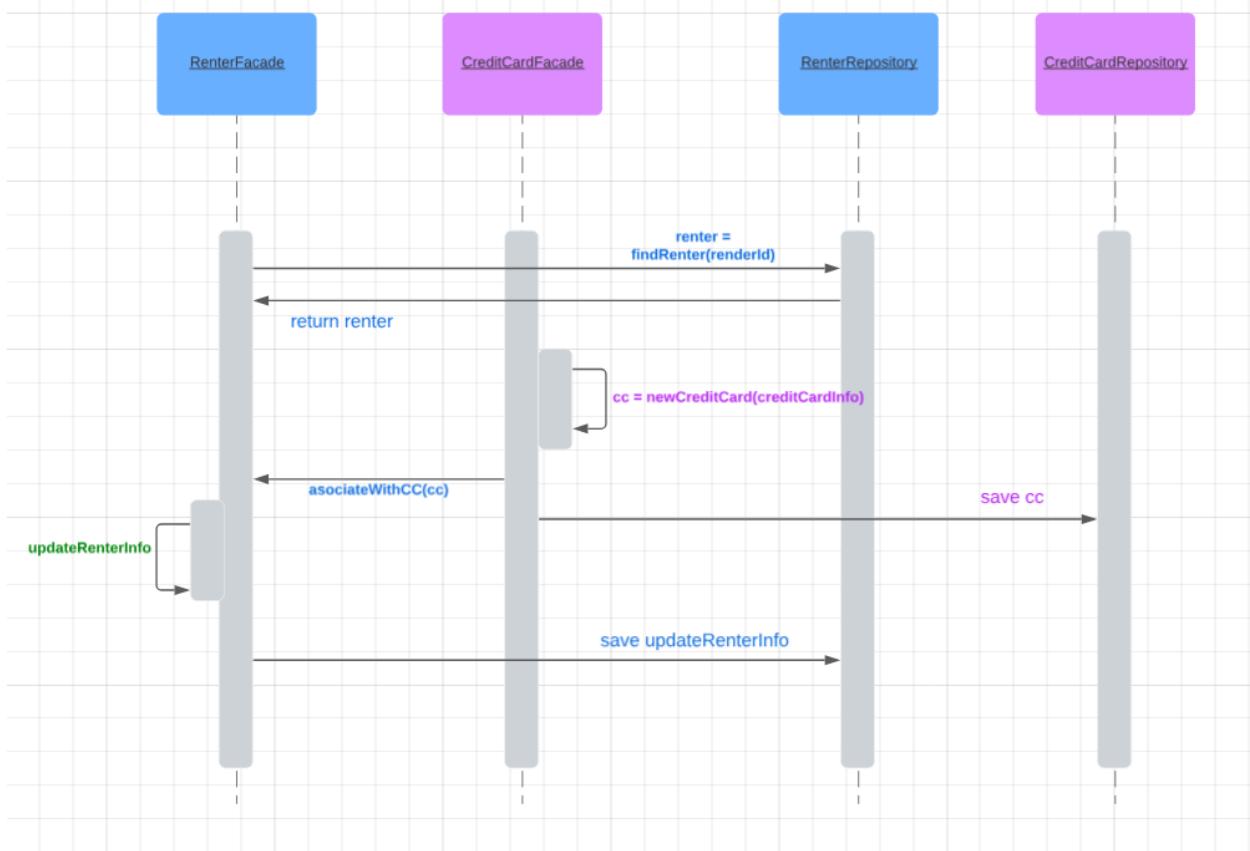


### °generateRentalAgreement:



°SetRenterCreditCard :

niveau 2 : SetRenterCreditCard



°finalizeRentalAgreement :

*niveau 2 :*  
finalizeRentalAgreement



rentagr =  
findRentalAgreement(rentalAgreementId)

return rentagr

markAsCompleted()

save(rentagr)