

Progetto Trips&Travel

Analisi e Progettazione del Software
Appello A.A.2023/2024

Gruppo: *iCasiD'Uso*

xxxxxx xxxxxx – mat. xxxxxx

xxxxxx xxxxxx – mat. xxxxxx

xxxxxx xxxxxx – mat. xxxxxx

xxxxxx xxxxxx – mat. xxxxxx



INDICE

Disciplina dei Requisiti

Diagramma dei Casi D’Uso	3
Casi d’uso in formato dettagliato	3
System Sequence Diagram – SSD	7
Contratti	10

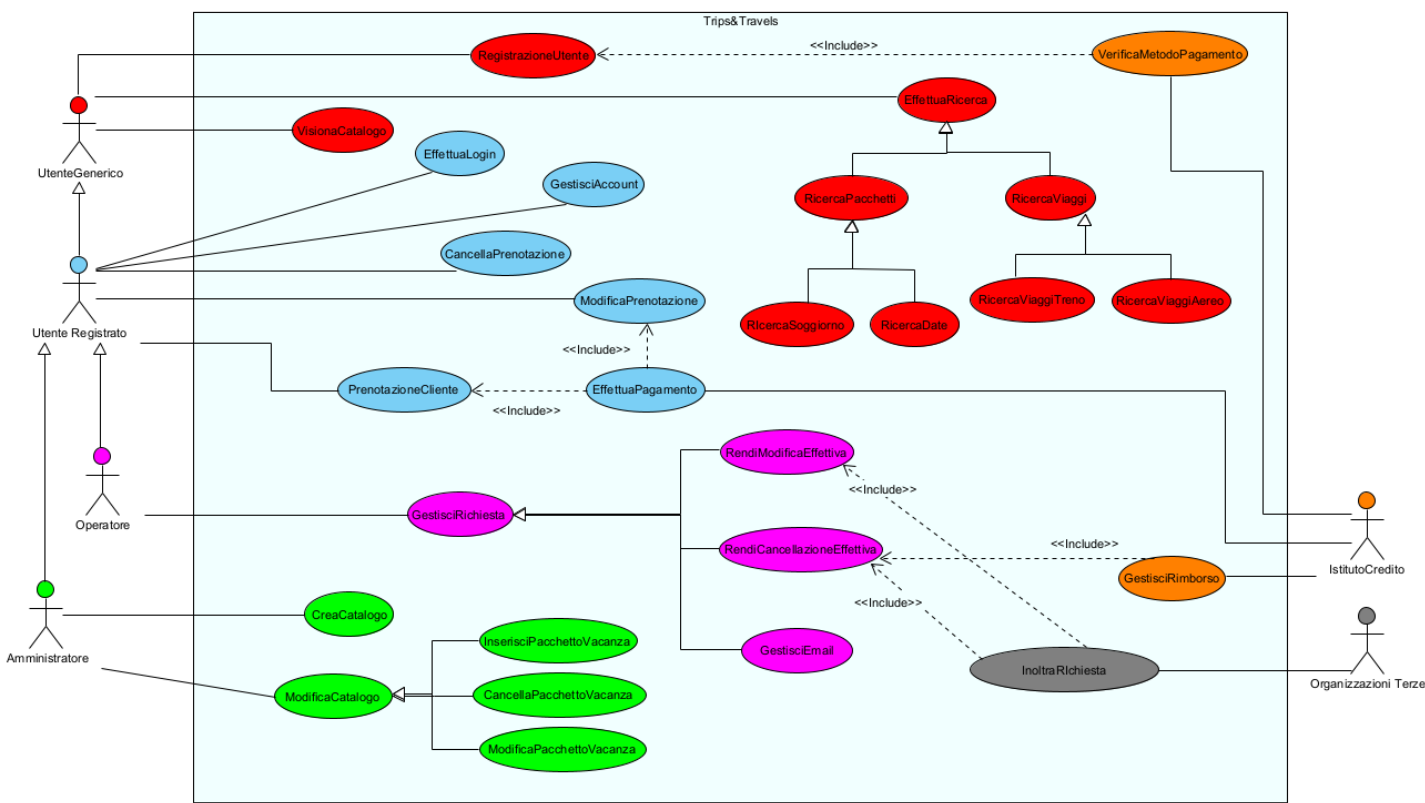
Modellazione del Business

Modello di Dominio	6
--------------------------	---

Progettazione

Diagrammi di attività	11
Sequence Diagram – SD	13
Architettura Logica	16
Diagramma Classi Software	16
Pattern applicati	18
Diagrammi di macchine a stati	18

Diagramma dei Casi D'Uso:



Casi D'uso in formato dettagliato:

CASO D'USO - PrenotazioneCliente	
Portata	Sistema Trips&Travels
Livello	Obiettivo Utente
Attore Primario	Utente Registrato
Parti interessate e interessi	<p>Utente Registrato: vuole effettuare una prenotazione di un pacchetto vacanza o di un viaggio in treno o aereo</p> <p>Organizzazioni Terze: riceve la richiesta in modo automatico dal sistema e la elabora</p> <p>Istituto di Credito: è l'intermediario del pagamento</p>
Pre-condizioni	<ul style="list-style-type: none">L'utente deve essere registrato con tutti i datiL'utente deve avere fatto il login al suo account
Post-condizioni	<ul style="list-style-type: none">L'utente ha prenotato il prodotto richiesto, ha pagato la caparra e ha ricevuto la confermaL'organizzazione terza ha ricevuto la richiesta e la elabora
Scenario principale	<p>1.L'utente cerca il prodotto che desidera</p> <p>2.L'utente seleziona "prenota prodotto"</p> <p>3.L'utente inserisce tutti i dati nel form</p> <p>4.L'utente effettua il pagamento della caparra indicata</p> <p>5.L'istituto di credito approva il pagamento</p> <p>6.L'utente riceve conferma e l'email di riepilogo</p> <p>7.Il sistema invia alle organizzazioni terze le specifiche del prodotto prenotato</p>
Estensioni	<p>Inserimenti mancanti nel form:</p> <p>3a. Il form non è stato compilato nel modo corretto</p> <p>4a. Il sistema richiede di correggere i campi mancanti o scorretti</p> <p>Quando sono corretti procedere con i passi 4,5 e 6 dello scenario principale</p> <p>Pagamento mancante:</p> <p>4b. Il pagamento della caparra non va a buon fine</p> <p>5b. Il sistema chiede all'utente di ritentare il pagamento</p> <p>Quando è andato a buon fine procedere con i passi 5 e 6 dello scenario principale</p>
Requisiti speciali	Il sistema invia la conferma entro 30 secondi
Elenco delle varianti tecnologiche e dei dati	-
Frequenze di ripetizione	Quasi ininterrotta
Punti estensione	Inclusion point "Effettua pagamento" al passo 2 dello scenario principale

CASO D’USO - InserisciPacchettoVacanza	
Portata	Sistema Trips&Travels
Livello	Obiettivo Utente
Attore Primario	Amministratore
Parti interessate e interessi	Amministratore: vuole inserire un pacchetto vacanza nel catalogo
Pre-condizioni	<ul style="list-style-type: none">L’amministratore ha effettuato il login al sistema con successo
Post-condizioni	<ul style="list-style-type: none">Il sistema ha registrato con successo l’inserimento di un pacchetto vacanza effettuato dall’amministratore
Scenario principale	<ol style="list-style-type: none">L’amministratore seleziona l’operazione di inserimento di un pacchetto vacanzaL’amministratore compila il form con i dati relativi al pacchetto (destinazione, alloggio, viaggio, date, prezzo)L’amministratore salva le scelte effettuateIl pacchetto vacanza viene inserito nel catalogo come disponibile e visibile agli utenti
Estensioni	Inserimenti mancanti nel form: 4a. Il form non è stato compilato nel modo corretto 5a. Il sistema richiede di correggere i campi mancanti o scorretti 6a. L’amministratore corregge i dati errati 7a. L’amministratore salva le scelte effettuate Quando i dati inseriti sono corretti procedere con il passo 5 dello scenario principale
Requisiti speciali	Il sistema inserisce il pacchetto vacanza nel catalogo entro 30 secondi
Elenco delle varianti tecnologiche e dei dati	-
Frequenze di ripetizione	Quasi ininterrotta
Punti estensione	-

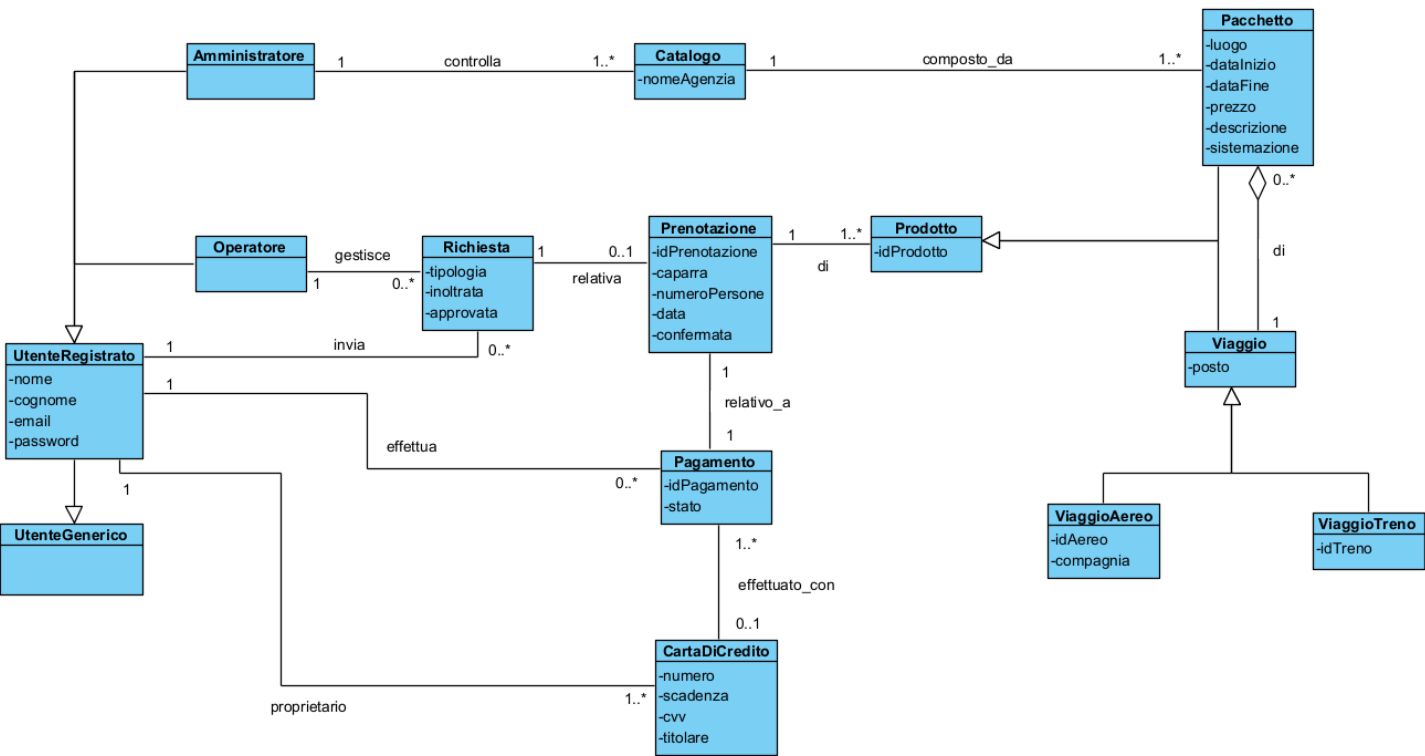
CASO D’USO - ModificaPacchettoVacanza	
Portata	Sistema Trips&Travels
Livello	Obiettivo Utente
Attore Primario	Amministratore
Parti interessate e interessi	Amministratore: vuole modificare un pacchetto vacanza nel catalogo
Pre-condizioni	<ul style="list-style-type: none">L’amministratore ha effettuato il login al sistema con successo
Post-condizioni	<ul style="list-style-type: none">Il sistema ha registrato con successo la modifica di un pacchetto vacanza effettuata dall’amministratore
Scenario principale	<ol style="list-style-type: none">L’amministratore seleziona l’operazione di modifica di un pacchetto vacanzaL’amministratore compila il form con i dati da aggiornare relativi al pacchettoL’amministratore salva le scelte effettuateIl pacchetto vacanza viene aggiornato nel catalogo
Estensioni	Inserimenti mancanti nel form: 4a. Il form non è stato compilato nel modo corretto 5a. Il sistema richiede di correggere i campi mancanti o scorretti 6a. L’amministratore salva le scelte effettuate Quando i dati sono corretti procedere con il passo 5 nello scenario principale
Requisiti speciali	Il sistema aggiorna i dati del pacchetto vacanza nel catalogo entro 30 secondi
Elenco delle varianti tecnologiche e dei dati	-
Frequenze di ripetizione	Quasi ininterrotta
Punti estensione	-

CASO D’USO - Cancellazione Pacchetto Vacanza	
Portata	Sistema Trips&Travels
Livello	Obiettivo Utente
Attore Primario	Amministratore
Parti interessate e interessi	Amministratore: vuole cancellare un pacchetto vacanza nel catalogo
Pre-condizioni	<ul style="list-style-type: none">L’amministratore ha effettuato il login al sistema con successoIl pacchetto vacanza da cancellare è presente e disponibile nel catalogo
Post-condizioni	<ul style="list-style-type: none">Il sistema ha registrato con successo la cancellazione di un pacchetto vacanza effettuata dall’amministratore
Scenario principale	<ol style="list-style-type: none">L’amministratore seleziona l’operazione di cancellazione di un pacchetto vacanzaIl sistema mostra un messaggio per chiedere di confermare la cancellazioneL’amministratore conferma la scelta di cancellazioneIl pacchetto vacanza viene cancellato dal catalogo
Estensioni	-
Requisiti speciali	Il sistema cancella il pacchetto vacanza nel catalogo entro 30 secondi
Elenco delle varianti tecnologiche e dei dati	-
Frequenze di ripetizione	Quasi ininterrotta
Punti estensione	-

CASO D’USO - Registrazione Utente	
Portata	Sistema Trips&Travels
Livello	Obiettivo Utente
Attore Primario	Utente Generico
Parti interessate e interessi	Utente Generico: intende registrarsi al sistema con il fine di gestire il proprio account e prenotazioni Istituto di Credito: verifica la validità del metodo di pagamento
Pre-condizioni	-
Post-condizioni	<ul style="list-style-type: none">L’utente, ora registrato, può gestire il proprio account, inserire e modificare richieste di prenotazioni
Scenario principale	<ol style="list-style-type: none">L’utente visualizza la finestra per la registrazioneL’utente inserisce i dati personali richiesti (nome, cognome, email, password, dati carta di credito)Il sistema accetta le credenziali inseriteL’istituto di credito verifica la validità del metodo di pagamento fornito dall’utenteL’utente, ora registrato, ottiene i suoi privilegi di gestione dell’account
Estensioni	credenzialiNonValide: 3a. Il sistema non accetta le credenziali poiché errate 4a. Il sistema richiede all’utente di reinserirle correttamente Quando i dati sono corretti procedere con il passo 3 dello scenario principale utenteGiàRegistrato: 3b. L’utente è già registrato 4b. Il sistema richiede all’utente di reinserire le credenziali Quando i dati sono corretti procedere con i passi 3, 4 e 5 dello scenario principale metodoPagamentoErrato: 5c. Il metodo di pagamento non è stato verificato 6c. Il sistema chiede nuovi dati per il metodo di pagamento Quando i dati sono corretti procedere con i passi 4 e 5 dello scenario principale
Requisiti speciali	Il sistema aggiunge l’utente registrato entro 30 secondi
Elenco delle varianti tecnologiche e dei dati	-
Frequenze di ripetizione	Quasi ininterrotta
Punti estensione	Inclusion point “Verifica Metodo Pagamento” al passo 4 dello scenario principale

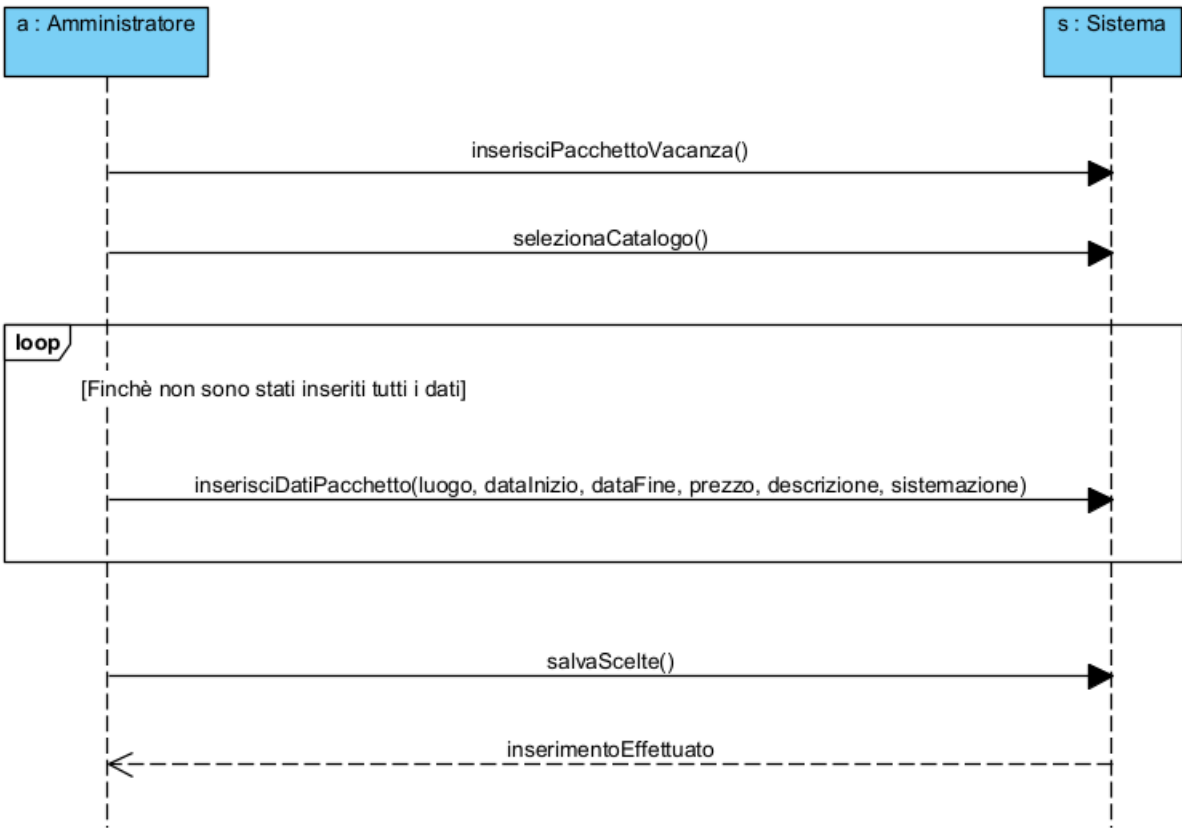
CASO D'USO - GestisciRichiesta	
Portata	Sistema Trips&Travels
Livello	Obiettivo Utente
Attore Primario	Operatore
Parti interessate e interessi	Operatore: deve gestire le richieste dell'utente Utente Generico: chiede informazioni relative a pacchetti o viaggi, oppure vuole modificare o cancellare la sua prenotazione Organizzazioni Terze: devono approvare la modifica/cancellazione del prodotto Istituto di Credito: deve gestire eventuali rimborsi
Pre-condizioni	<ul style="list-style-type: none">L'operatore deve aver fatto il loginL'utente deve aver inviato una richiesta informazioni oppure richiesto un'operazione di modifica/cancellazione su una prenotazione
Post-condizioni	<ul style="list-style-type: none">L'operatore ha soddisfatto la richiesta dell'utente
Scenario principale	1. L'operatore inoltra la modifica all'organizzazione terza 2. L'organizzazione verifica la fattibilità della richiesta 3a. Se la modifica è fattibile 3a.1. L'organizzazione conferma la fattibilità della richiesta 3b. Se la modifica non è fattibile 3b.1. L'operatore effettua la modifica del prodotto nell'account dell'utente
Estensioni	cancellazioneProdotto: 1b. L'operatore inoltra la richiesta di cancellazione all'organizzazione terza 2b. L'organizzazione verifica la fattibilità della richiesta 3b. L'operatore avvisa l'utente se la cancellazione è andata a buon fine e in caso procede con il rimborso richiestaViaEmail: 1d. L'operatore risponde in modo esaustivo alla mail che l'utente ha inviato 2d. La comunicazione continua finché la domanda dell'utente non viene soddisfatta
Requisiti speciali	L'Operatore deve rispondere alle richieste entro 2 giorni
Elenco delle varianti tecnologiche e dei dati	-
Frequenze di ripetizione	Quasi ininterrotta
Punti estensione	-

Modello di Dominio:

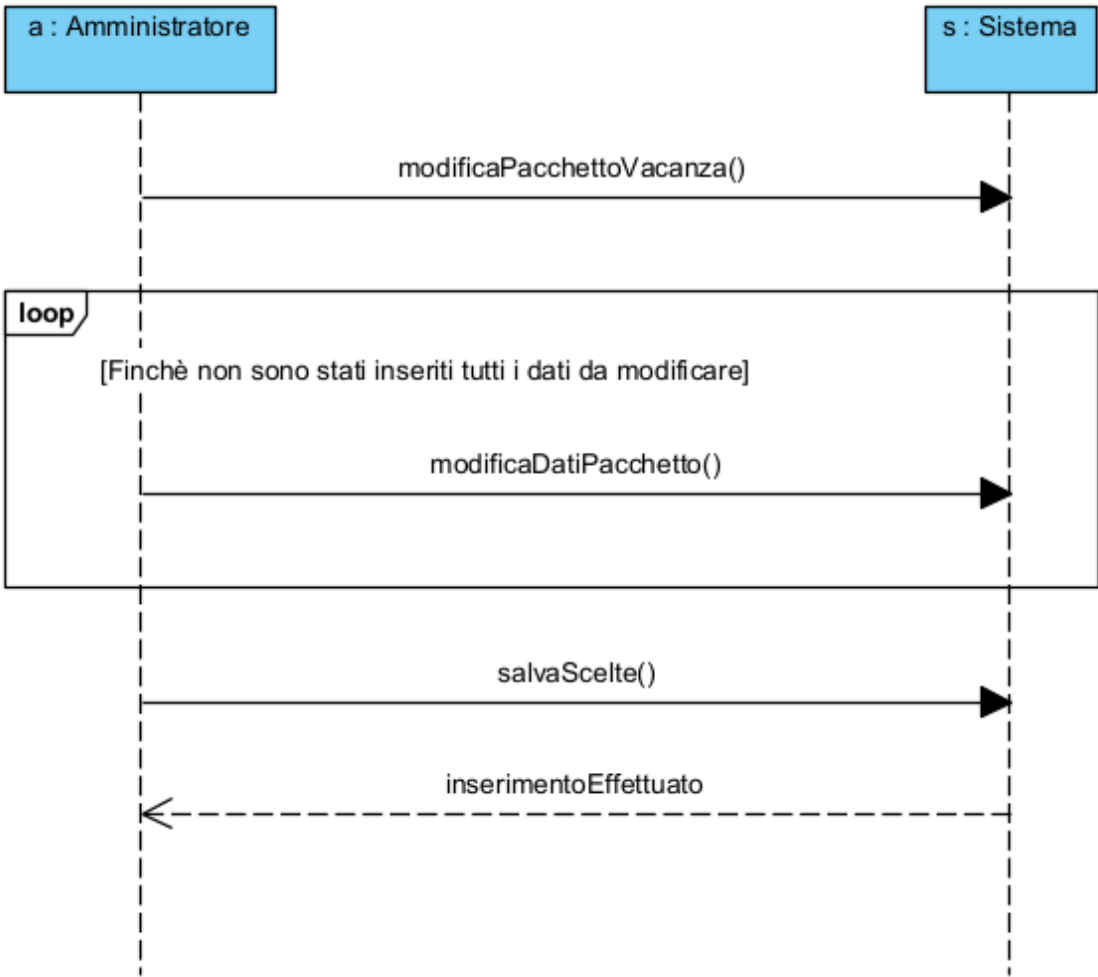


System Sequence Diagram - SSD:

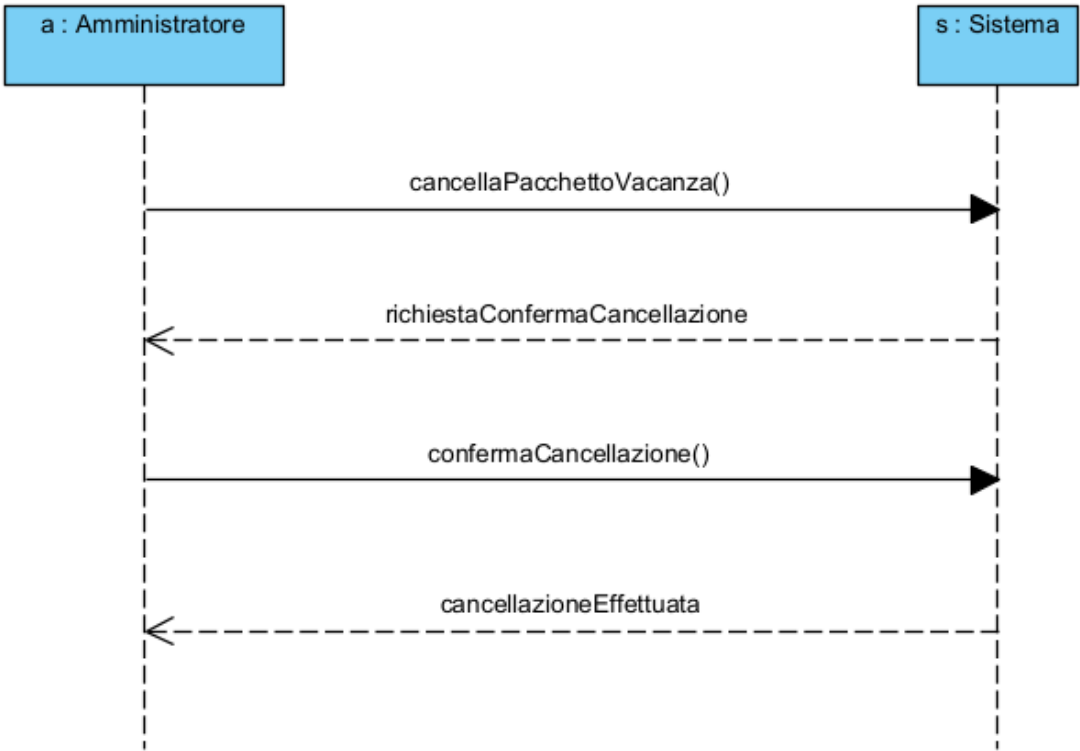
InserisciPacchettoVacanza:



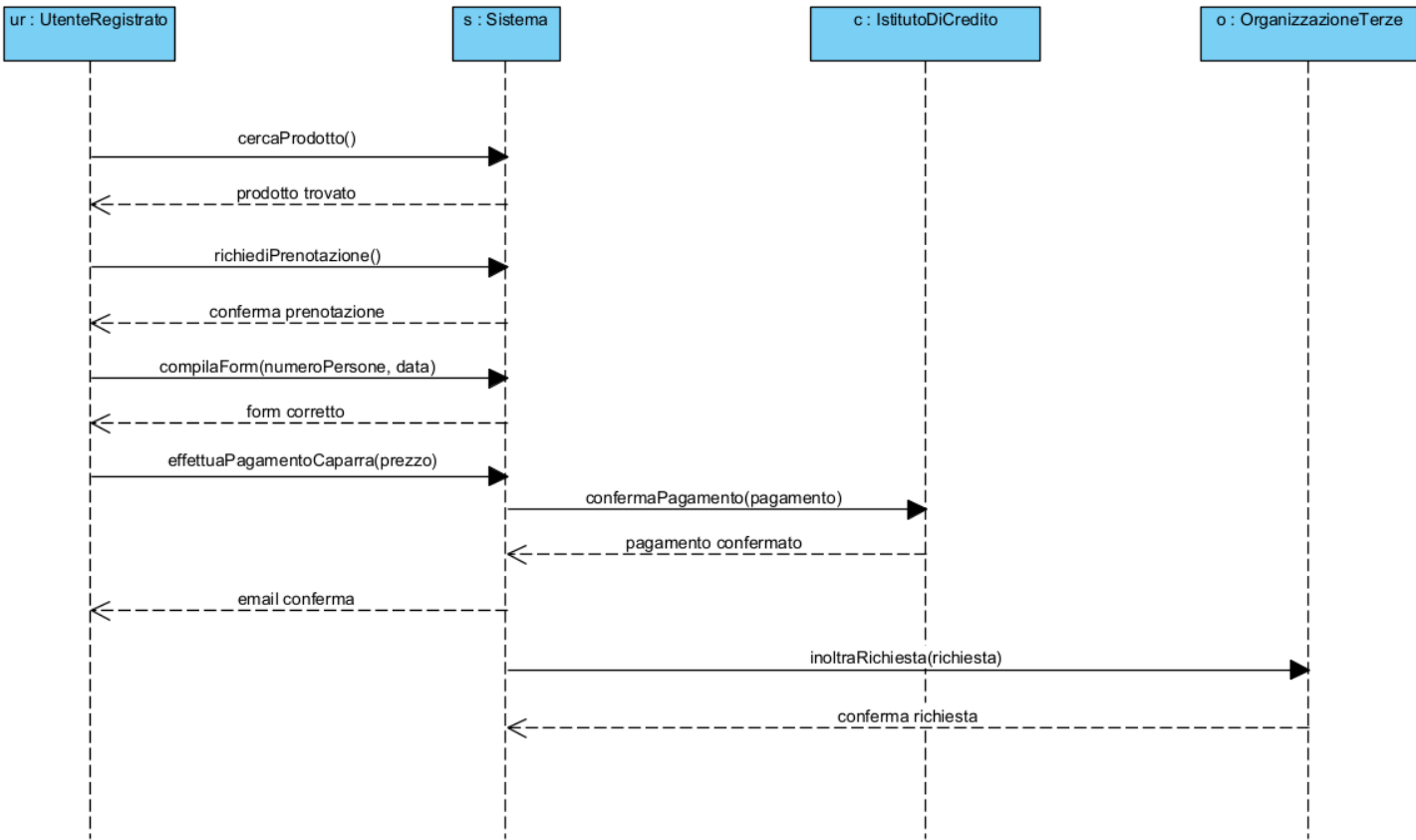
ModificaPacchettoVacanza:



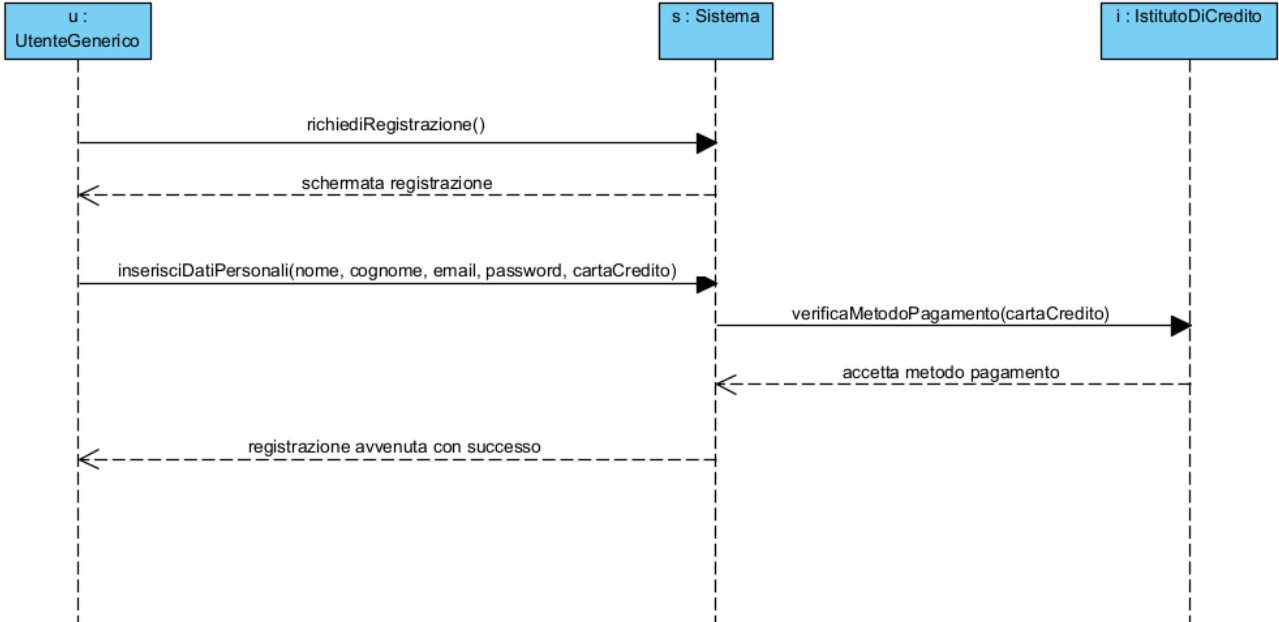
CancellaPacchettoVacanza:



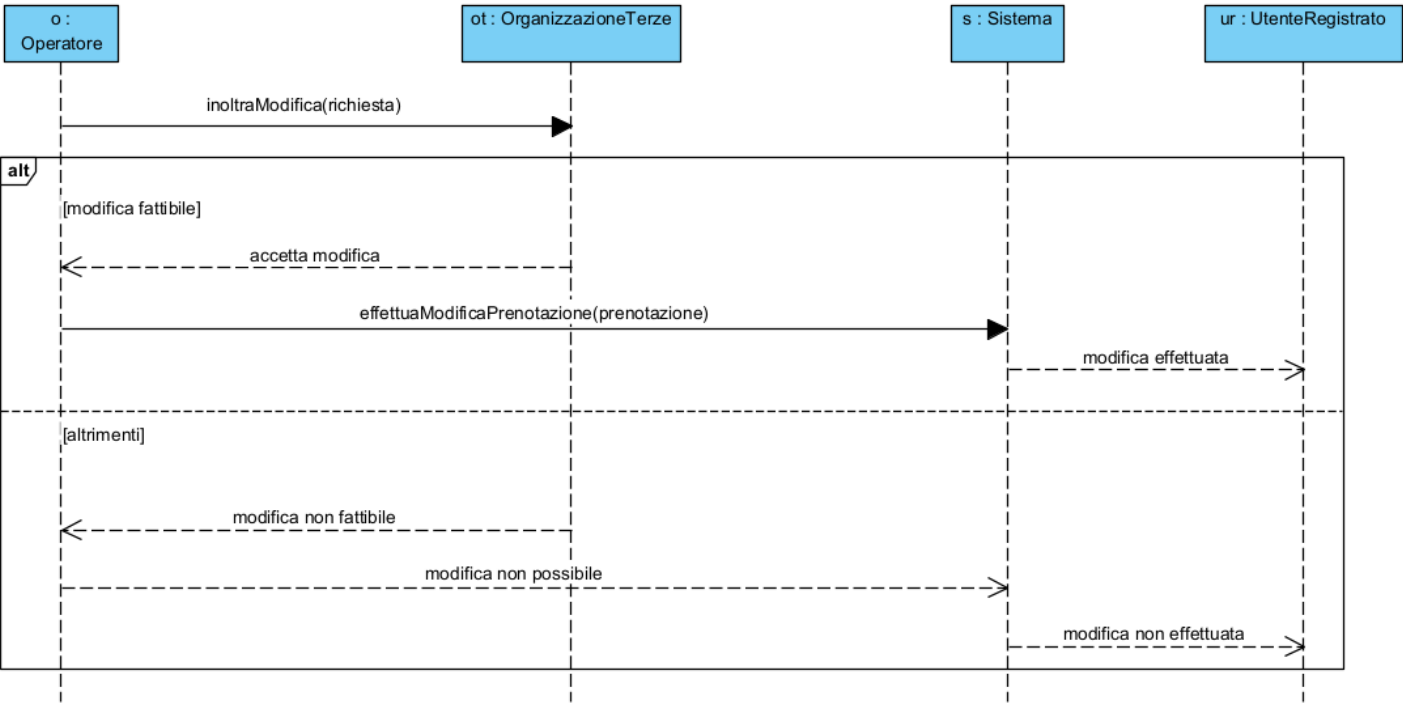
PrenotazioneCliente:



RegistrazioneUtente:



GestisciRichiesta:



Contratti:

CONTRATTO - RichiediPrenotazione	
Operazione	RichiediPrenotazione()
Riferimenti	Caso d’uso PrenotazioneCliente()
Pre-condizioni	<ul style="list-style-type: none">Si sta creando una nuova prenotazione
Post-condizioni	<ul style="list-style-type: none">È stata creata una nuova istanza pr di prenotazione (creazione istanza)L’istanza pr è stata associata a Prodotto (formazione di collegamento)

CONTRATTO - PagamentoCaparra	
Operazione	EffettuaPagamentoCaparra(prezzo)
Riferimenti	Caso d’uso PrenotazioneCliente()
Pre-condizioni	<ul style="list-style-type: none">Si sta pagando la caparra di una prenotazione
Post-condizioni	<ul style="list-style-type: none">È stata creata una nuova istanza p di pagamento (creazione istanza)Gli attributi di p vengono inizializzati (modifica attributi), in particolare stato = inLavorazioneL’istanza è stata associata a CartaDiCredito, a Prenotazione e a UtenteRegistrato (formazione di collegamento)

CONTRATTO - InserisciPacchettoVacanza	
Operazione	inserisciPacchettoVacanza()
Riferimenti	Caso d’uso InserisciPacchettoVacanza()
Pre-condizioni	<ul style="list-style-type: none">L’amministratore accede alla sezione di modifica dei dati del sistema
Post-condizioni	<ul style="list-style-type: none">È stata creata un’istanza p di pacchetto (creazione istanza)

CONTRATTO - InserisciDatiPacchetto	
Operazione	inserisciDatiPacchetto(luogo, dataInizio, dataFine, prezzo, descrizione, sistemazione)
Riferimenti	Caso d’uso InserisciPacchettoVacanza()
Pre-condizioni	<ul style="list-style-type: none">É presente un’istanza p di pacchettoÉ presente almeno un’istanza c di catalogoSono state associate le istanze c e p
Post-condizioni	<ul style="list-style-type: none">Sono stati inizializzati gli attributi dell’istanza p di pacchetto (modifica di attributi)

CONTRATTO - RichiediRegistrazione	
Operazione	richiediRegistrazione()
Riferimenti	Caso d’uso RegistrazioneUtente()
Pre-condizioni	<ul style="list-style-type: none">L’utente generico è collegato alla piattaforma
Post-condizioni	<ul style="list-style-type: none">È stata creata l’istanza ur di UtenteRegistrato (creazione istanza)

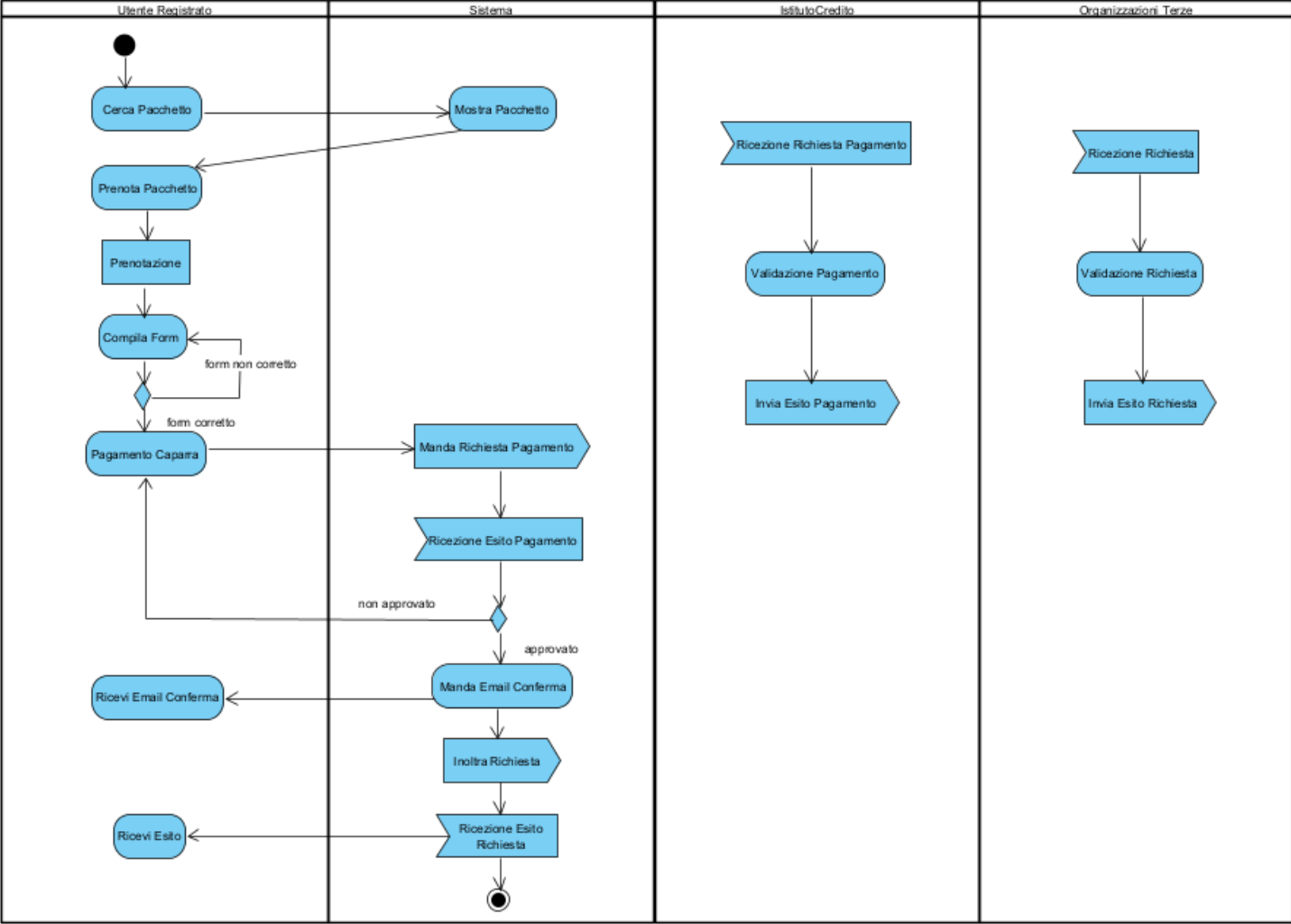
CONTRATTO - InserisciDatiPersonali	
Operazione	inserisciDatiPersonali()
Riferimenti	Caso d’uso RegistrazioneUtente()
Pre-condizioni	<ul style="list-style-type: none">È stata creata l’istanza ur di UtenteRegistrato (creazione istanza)
Post-condizioni	<ul style="list-style-type: none">Sono stati inizializzati gli attributi dell’istanza ur di UtenteRegistrato (modifica di attributi)È stata creata l’istanza c di CartaDiCredito (creazione di istanza)L’istanza ur di UtenteRegistrato è stata associata a CartaDiCredito (formazione di collegamento)

CONTRATTO - InoltraModifica	
Operazione	inoltraModifica()
Riferimenti	Caso d’uso GestisciRichiesta()
Pre-condizioni	<ul style="list-style-type: none">• É presente un’istanza r di richiesta
Post-condizioni	<ul style="list-style-type: none">• Vengono modificati gli attributi di r (modifica attributi), in particolare inoltrata = true

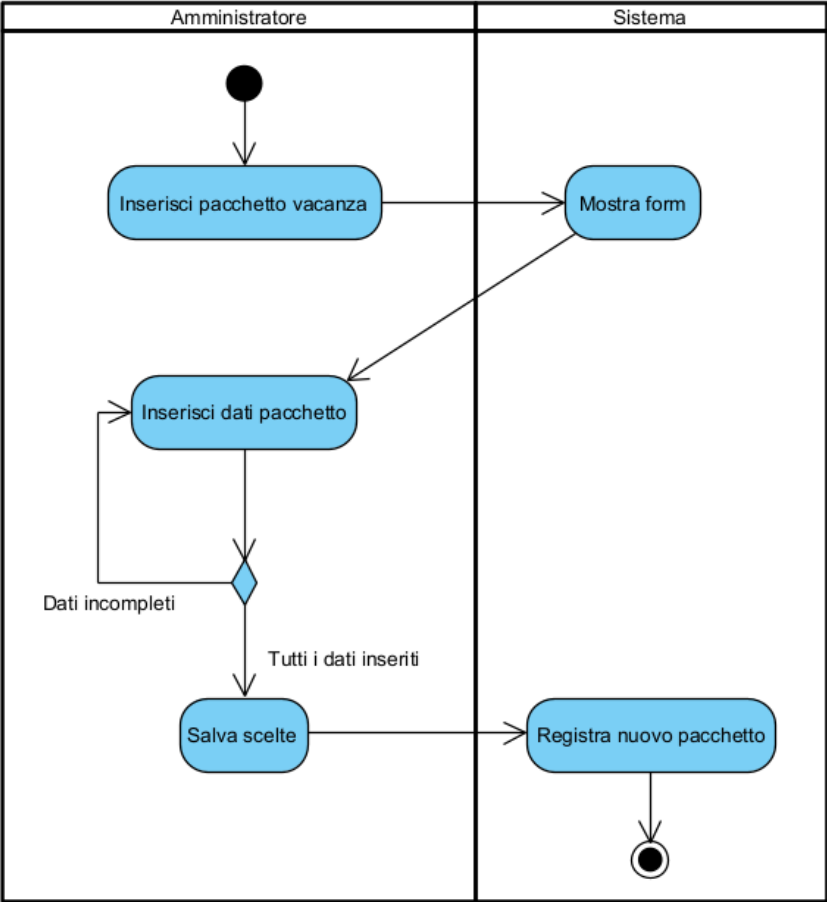
CONTRATTO - EffettuaModificaPrenotazione	
Operazione	effettuaModificaPrenotazione(nome, cognome, email, password, cartaCredito)
Riferimenti	Caso d’uso GestisciRichiesta()
Pre-condizioni	<ul style="list-style-type: none">• É in corso una richiesta di modifica• É presente un'istanza r di richiesta
Post-condizioni	<ul style="list-style-type: none">• Vengono modificati gli attributi di prenotazione pr (modifica di attributi)

Diagrammi di attività:

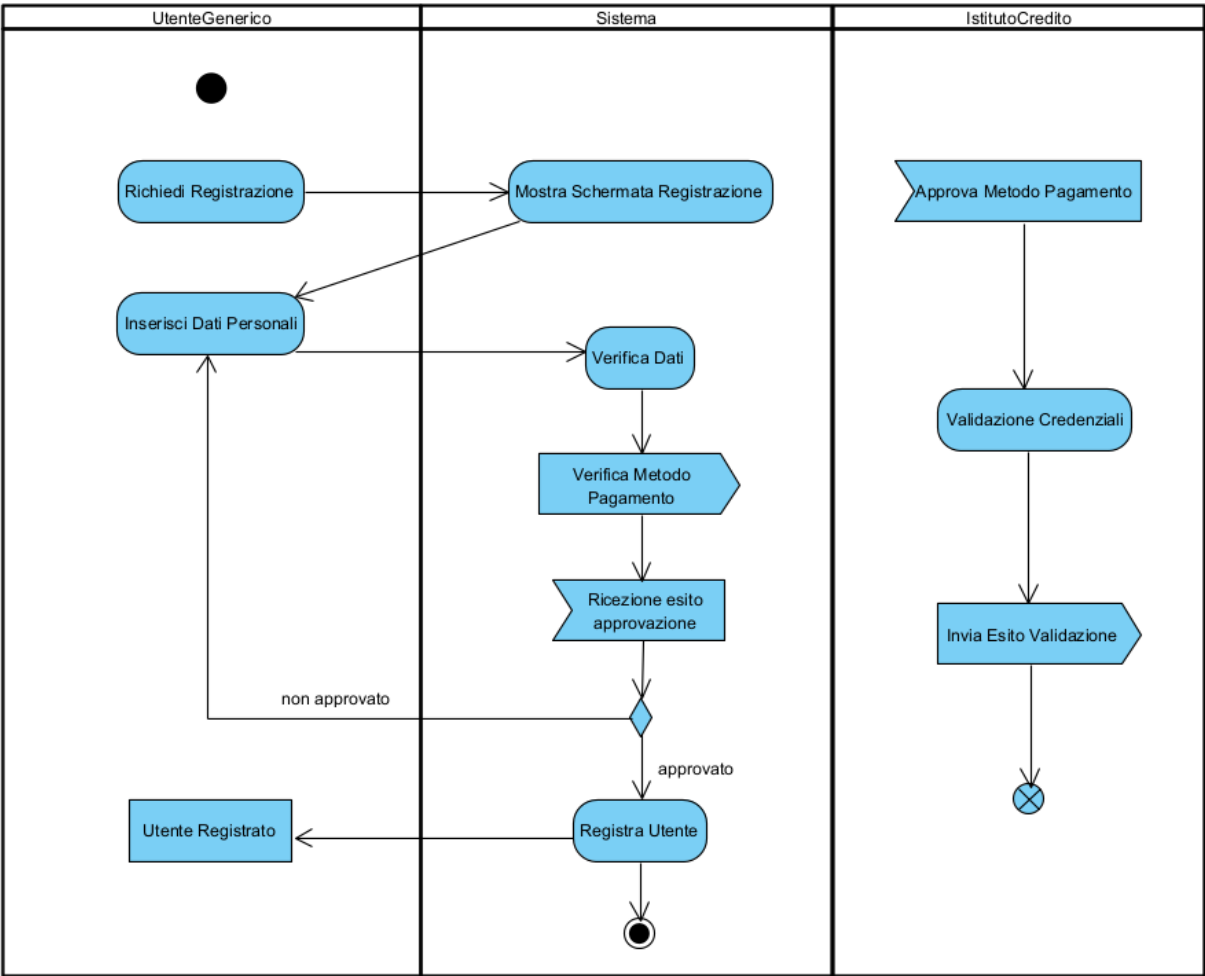
PrenotazionePacchettoVacanza:



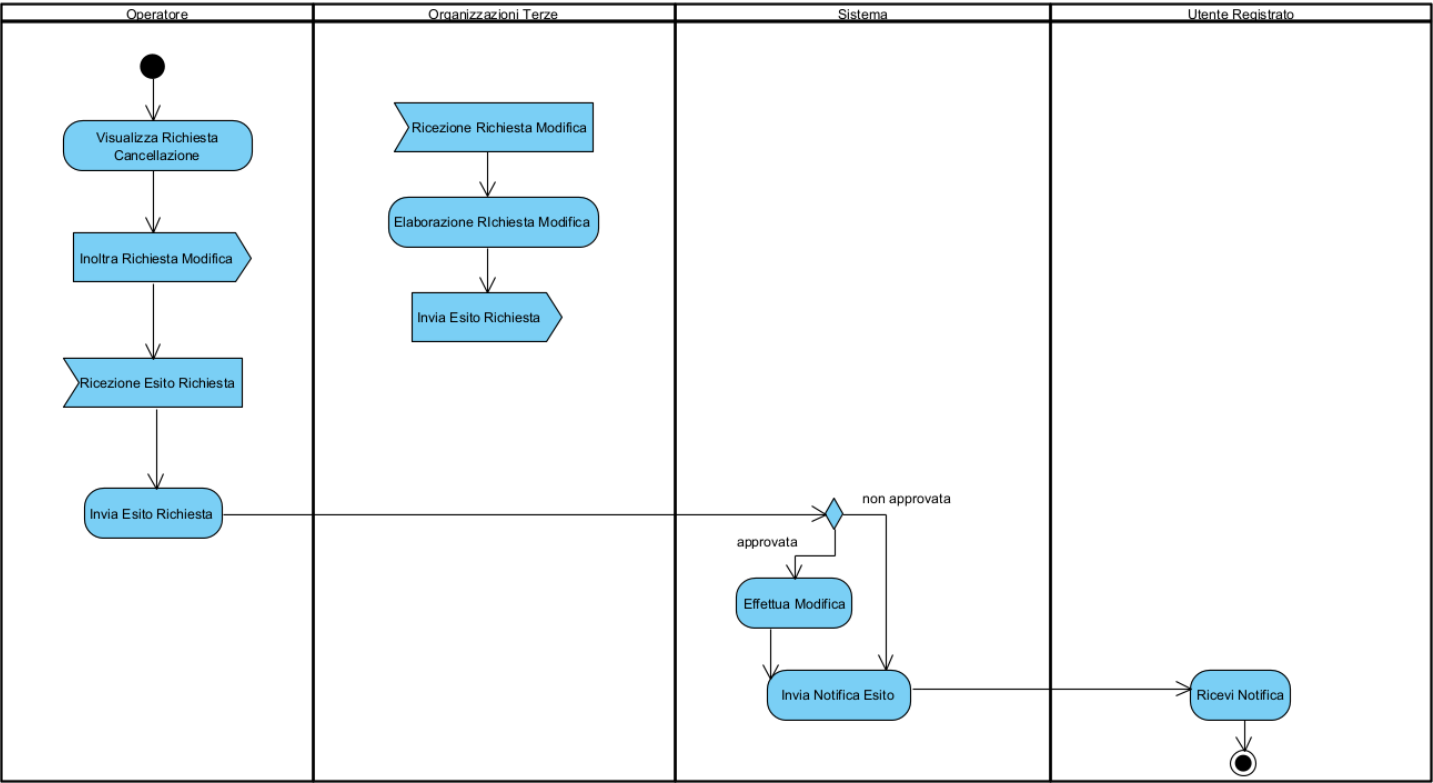
InserisciPacchettoVacanza:



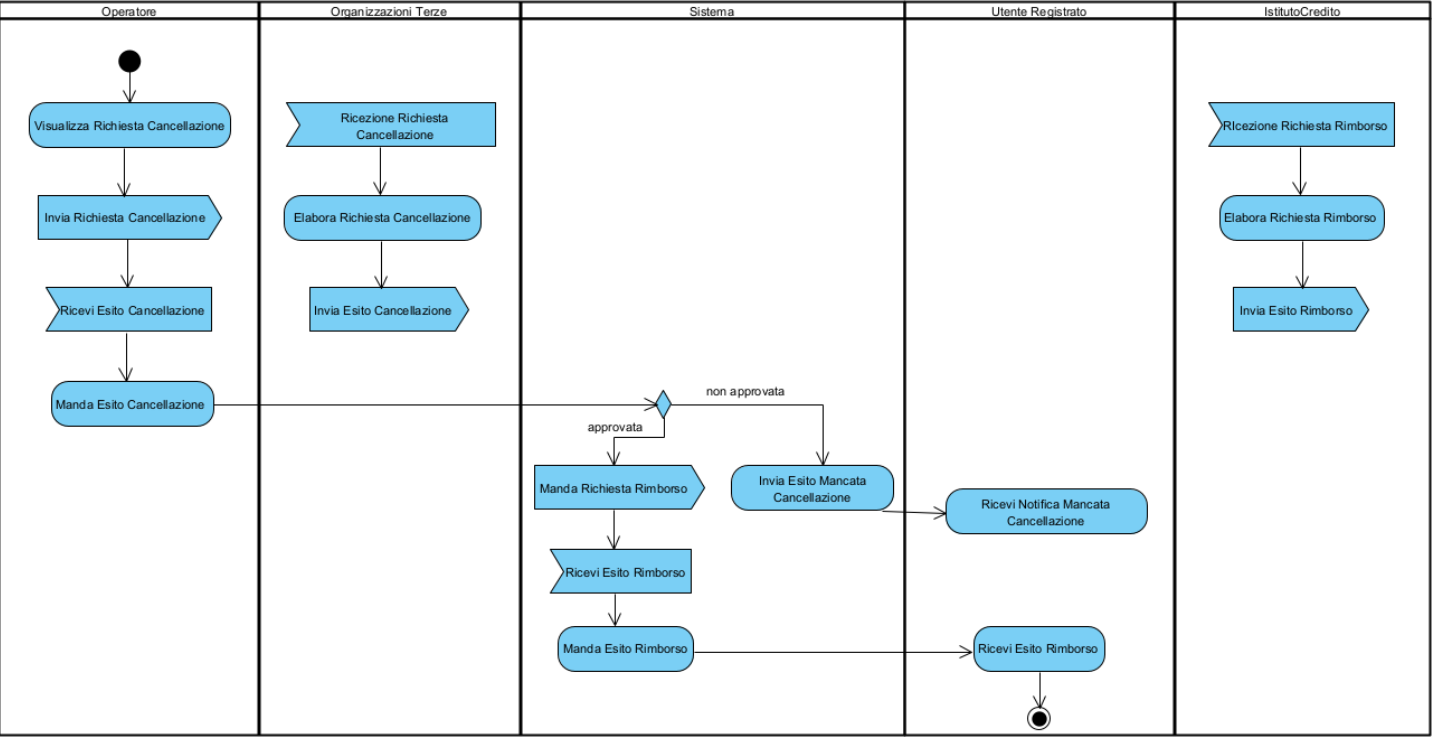
RegistrazioneUtente:



GestisciRichiestaModifica:

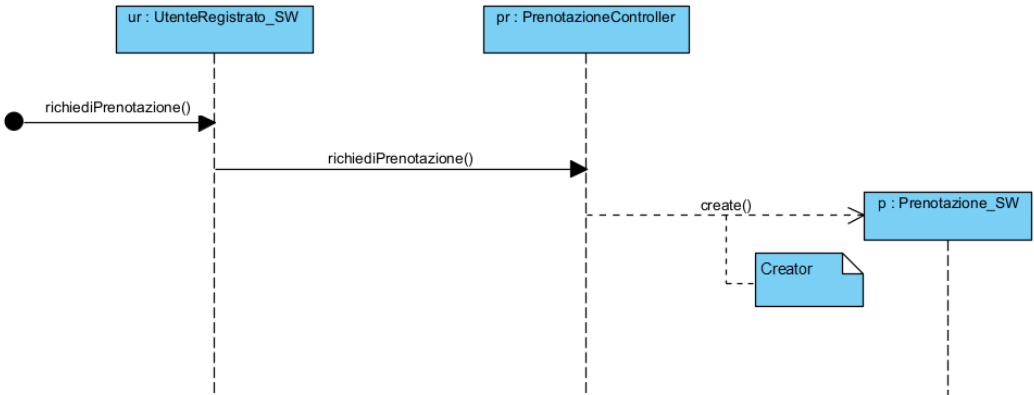


GestisciRichiestaCancellazione:

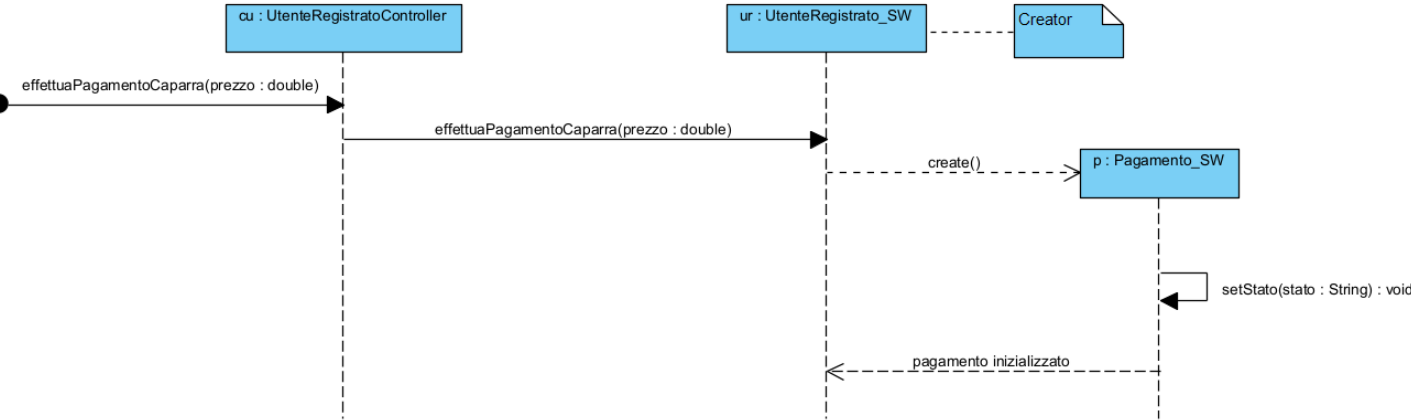


Sequence Diagram - SD:

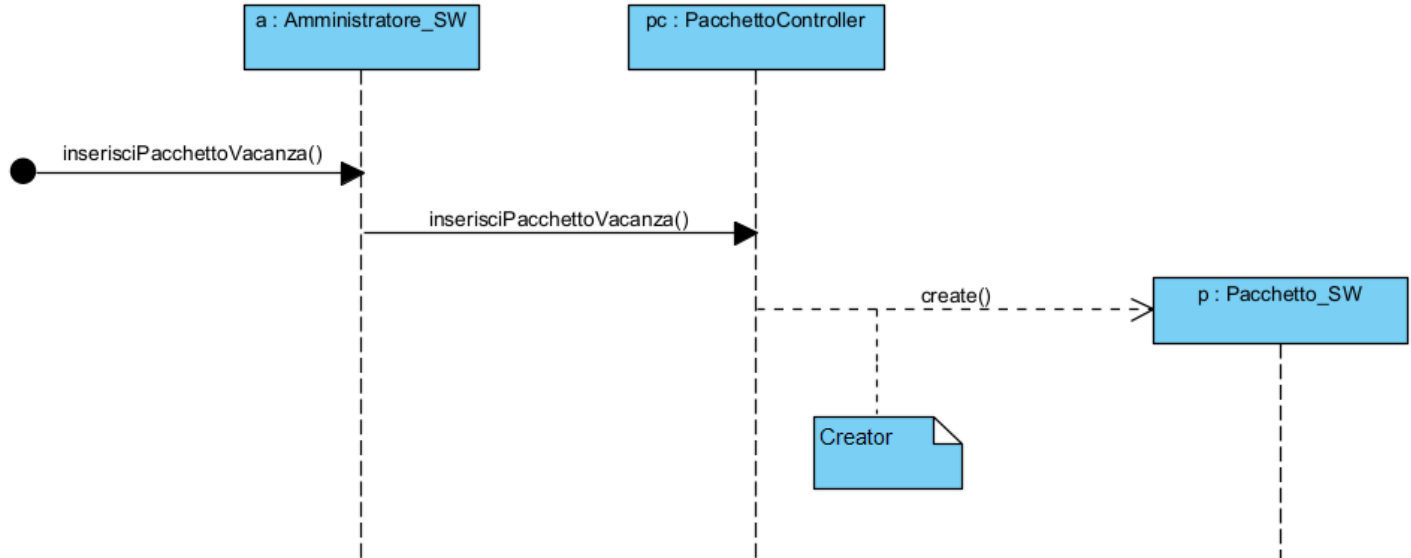
RichiediPrenotazione:



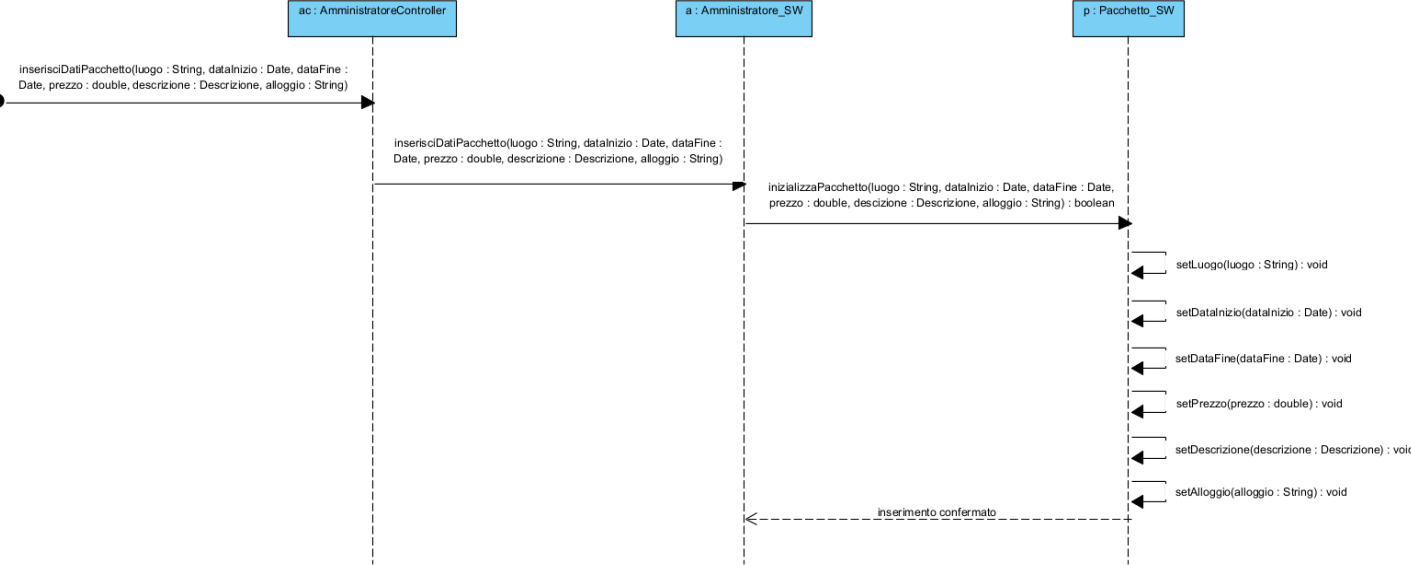
PagamentoCaparra:



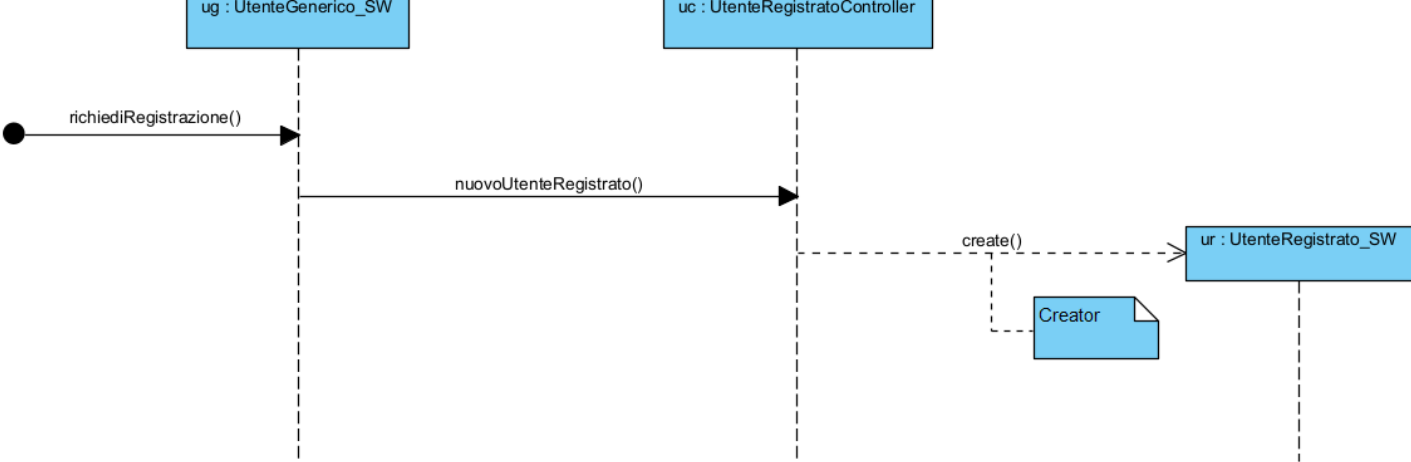
InserisciPacchettoVacanza:



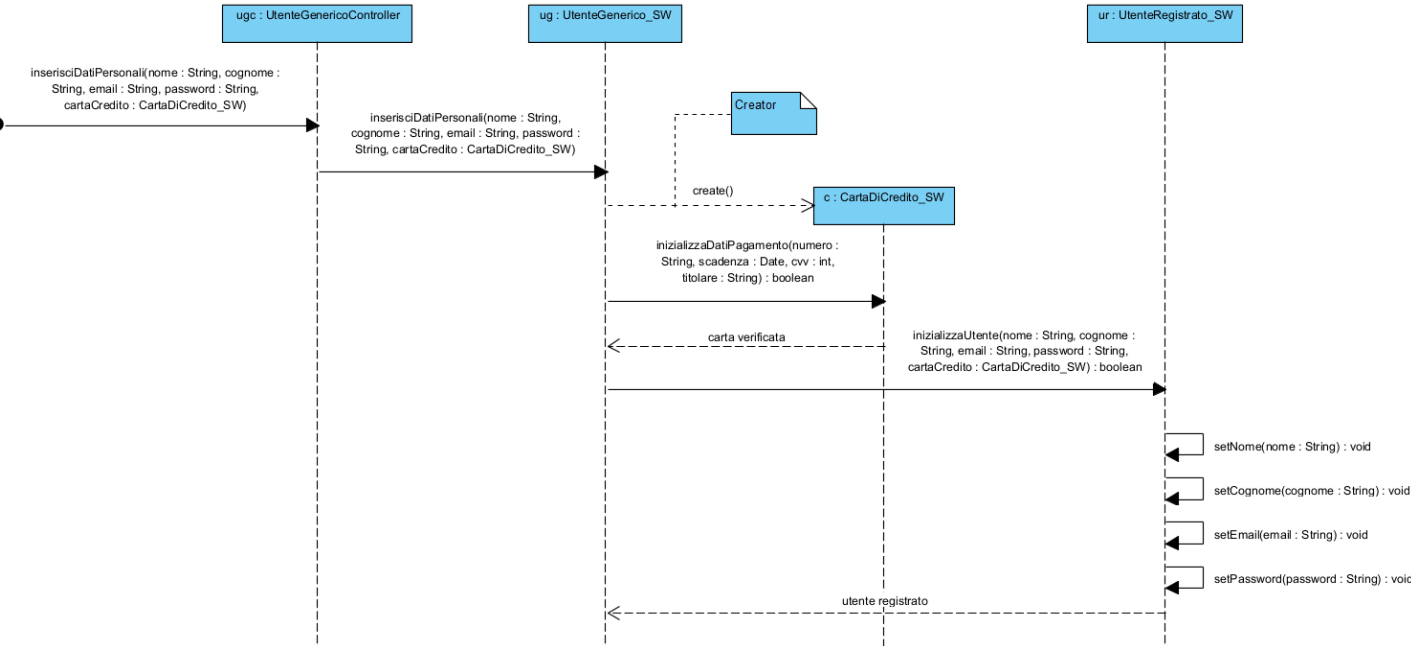
InserisciDatiPacchetto:



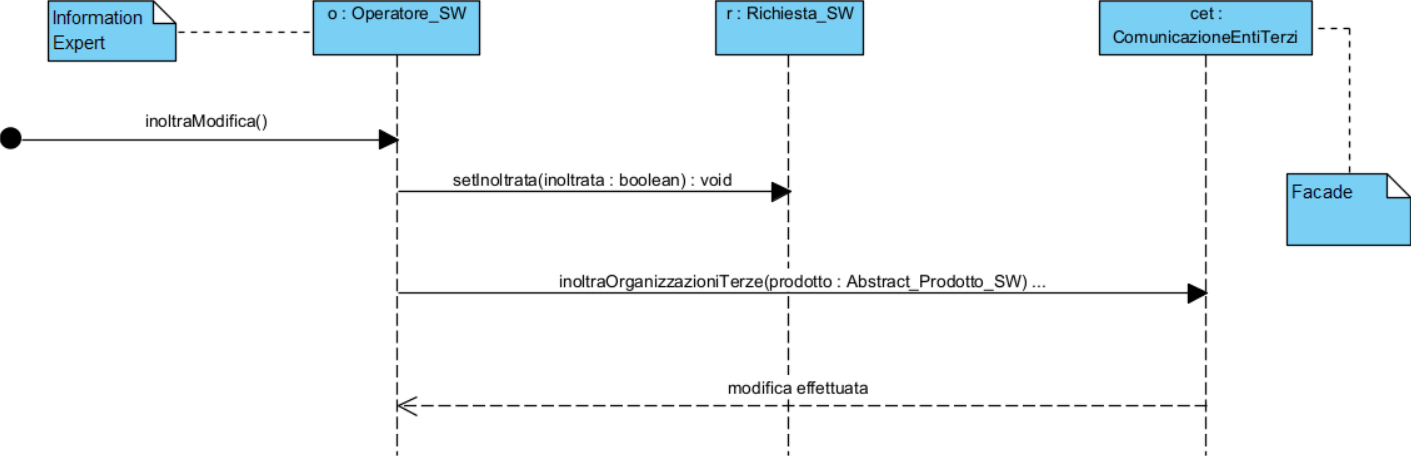
RichiediRegistrazione:



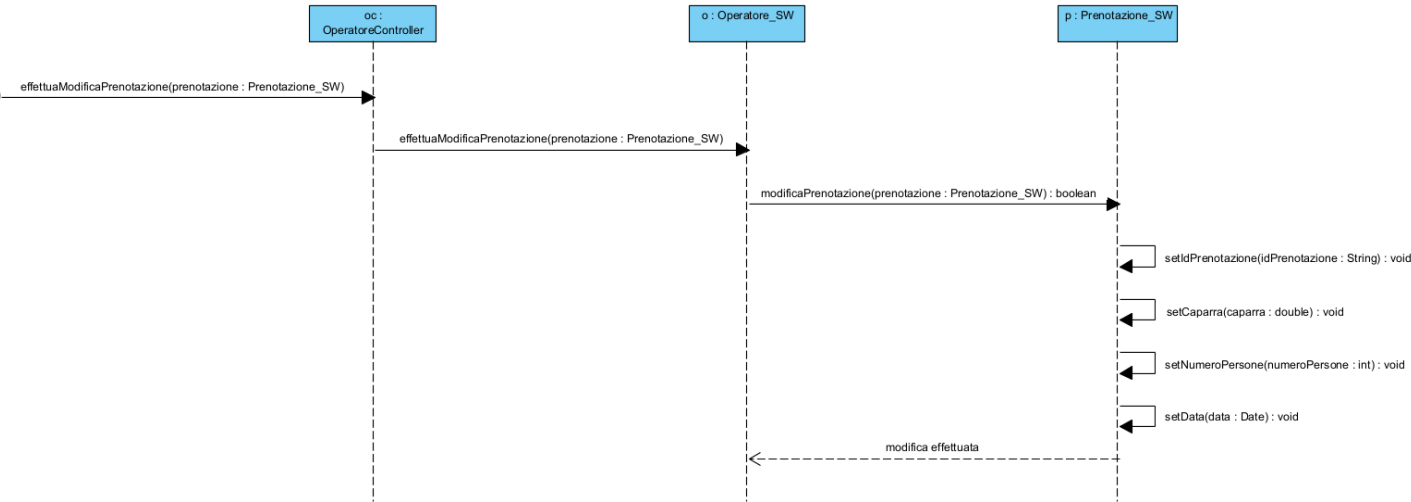
InserisciDatiPersonali:



InoltraModifica:



EffettuaModificaPrenotazione:



Architettura Logica:

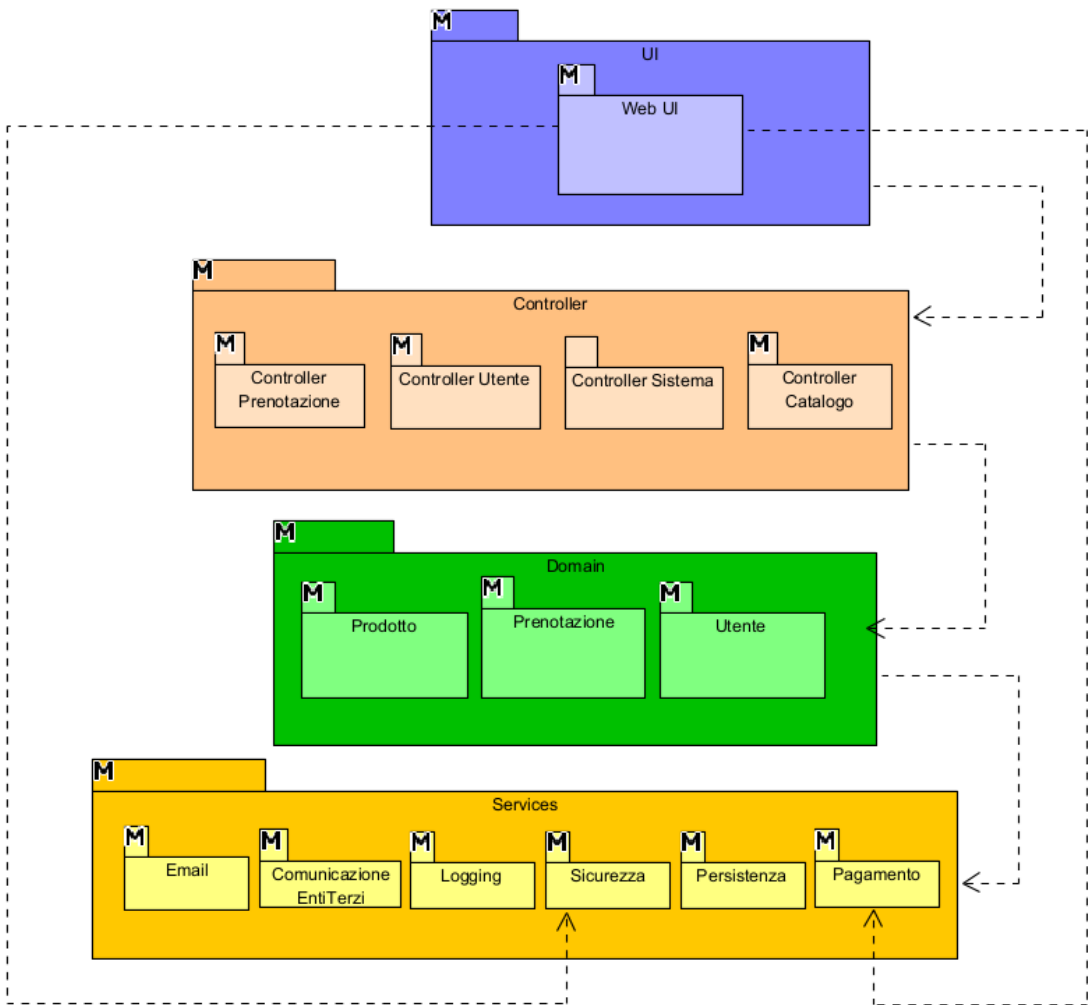
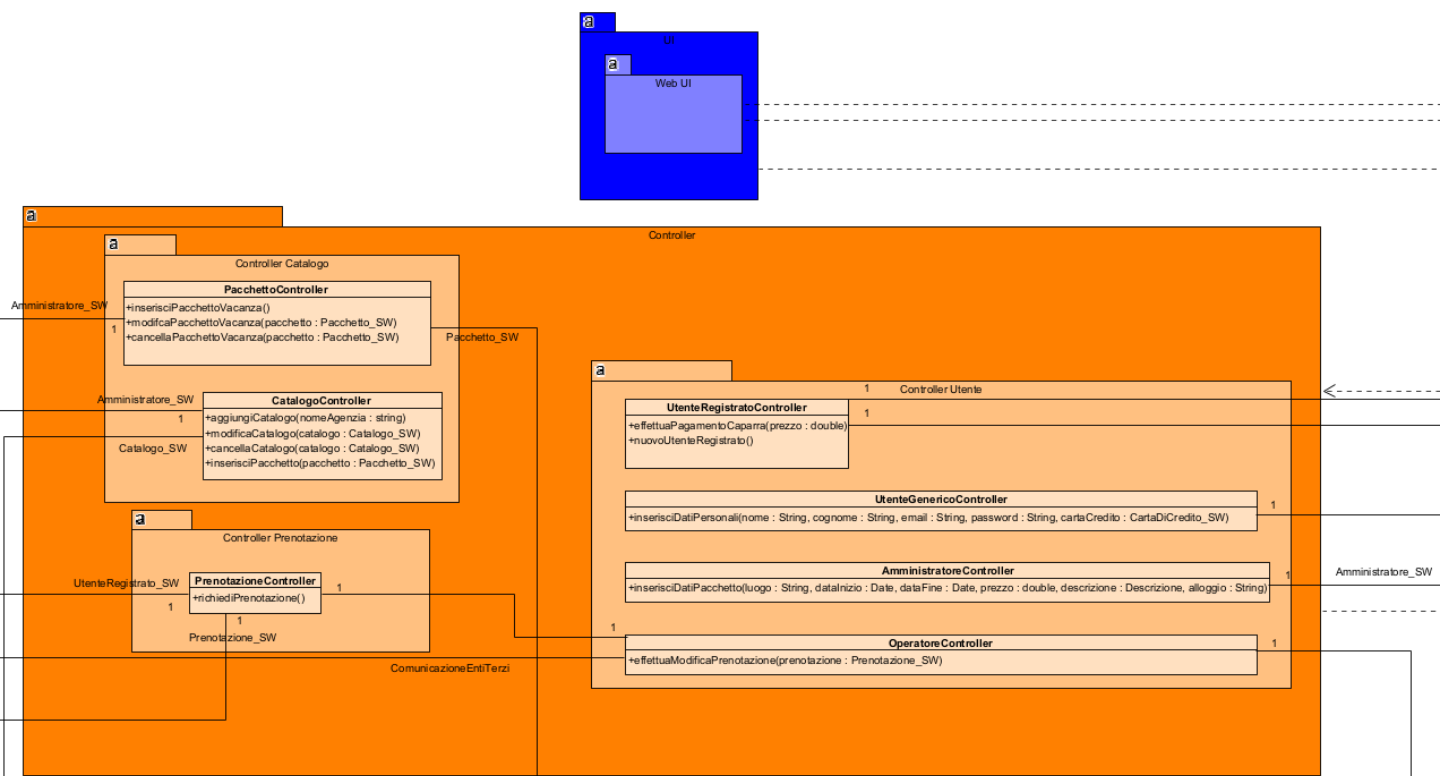


Diagramma Classi Software:

N.B.: Per il diagramma completo in HD vedere l'allegato «Diagramma_Classi_SW»

Package UI e Package Controller:




```
classDiagram
    class ViaggioAereo_SW {
        -idAereo : String
        -compagnia : String
        +getIdAereo() : String
        +setIdAereo(idAereo : String) : void
        +getCompagnia() : String
        +setCompagnia(compagnia : String) : void
    }
    class ViaggioTreno_SW {
        -idTreno : String
        +getIdTreno() : String
        +setIdTreno(idTreno : String) : void
    }
    class Abstract_Viaggio_SW {
        -posto : String
        -partenza : String
        -destinazione : String
        +geIPosto() : String
        +setPosto(posto : String) : void
        +geIPartenza() : String
        +setPartenza(partenza : String) : void
        +geIDestinazione() : String
        +setDestinazione(destinazione : String) : void
    }
    class Prodotto {
        -luogo : String
        -dataInizio : Date
        -dataFine : Date
        -prezzo : double
        -descrizione : Descrizione
        -alloggio : String
        +create()
        +getLuogo() : String
        +setLuogo(luogo : String) : void
        +getDataInizio() : Date
        +setDataInizio(dataInizio : Date) : void
        +getDataFine() : Date
        +setDataFine(dataFine : Date) : void
        +getPrezzo() : double
        +setPrezzo(prezzo : double) : void
        +getDescription() : Descrizione
        +setDescription(descrizione : Descrizione) : void
        +getAlloggio() : String
        +setAlloggio(alloggio : String) : void
        +inizializzaPacchetto(luogo : String, dataInizio : Date, dataFine : Date, prezzo : double, descrizione : Descrizione, alloggio : String) : boolean
        +modificaPacchettoVacanza(pacchetto : Pacchetto_SW)
        +cancellaPacchettoVacanza(pacchetto : Pacchetto_SW)
    }
    class Pacchetto_SW {
    }
    class Catalogo_SW {
        -nomeAgenzia : String
        +create(nomeAgenzia : string)
        +getNomeAgenzia() : String
        +setNomeAgenzia(nomeAgenzia : String) : void
        +inserisciPacchettoVacanza(pacchetto : Pacchetto_SW) : boolean
        +modificaCatalogo(catalogo : Catalogo_SW)
        +cancellaCatalogo(catalogo : Catalogo_SW)
    }
    class CatalogoController {
    }
    class Amministratore_SW {
        +inserisciPacchettoVacanza()
        +inserisciDatiPacchetto(luogo : String, dataInizio : Date, dataFine : Date, prezzo : double, descrizione : Descrizione, alloggio : String)
    }
    class AmministratoreController {
    }
    class Operatore_SW {
    }
    class OperatoreController {
    }

    ViaggioAereo_SW <|-- Abstract_Viaggio_SW
    ViaggioTreno_SW <|-- Abstract_Viaggio_SW
    Abstract_Viaggio_SW <|-- Abstract_Prodotto_SW
    Prodotto <|-- Pacchetto_SW
    Abstract_Prodotto_SW <|-- Prodotto
    CatalogoController --> Abstract_Viaggio_SW
    CatalogoController --> Abstract_Prodotto_SW
    Catalogo_SW --> CatalogoController
    Catalogo_SW --> CatalogoController : 1..*
    CatalogoController --> Abstract_Viaggio_SW : 1
    CatalogoController --> Abstract_Prodotto_SW : 1..*
    Catalogo_SW --> Amministratore_SW : 1
    Amministratore_SW --> AmministratoreController : 1
    AmministratoreController --> Operatore_SW : 1
    Operatore_SW --> OperatoreController : 1
```

```

classDiagram
    class PacchettoControllore {
        +InserisciDataPacchetto(uogo : String, dataFine : Date, prezzo : double, descrizione : Descrizione, alloggio : String)
    }
    class UtenteRegistrato_SW {
        -nome : String
        -cognome : String
        -email : String
        -password : String
        +richiediPrenotazione()
        +effettuaPagamentoCaparra(prezzo : double) : boolean
        +create()
        +inizializzaUtente(nome : String, cognome : String, email : String, password : String, cartaCredito : CartaDiCredito_SW) : boolean
        +getNome() : String
        +setName(nome : String) : void
        +getCognome() : String
        +setCognome(cognome : String) : void
        +getEmail() : String
        +setEmail(email : String) : void
        +getPassword() : String
        +setPassword(password : String) : void
        +compilaForm(Form form) : boolean
        +inviaRichiesta(Richiesta_SW richiesta) : void
    }
    class Operatore_SW {
        +inoltraModifica() : boolean
        +effettuaModificaPrenotazione(prenotazione : Prenotazione_SW)
        +rispondi>Email()
    }
    class UtenteGenerico_SW {
        +richiediRegistrazione()
        +inserisciDatPersonali(nome : String, cognome : String, email : String, password : String, cartaCredito : CartaDiCredito_SW)
        +cercaProdotto(idProdotto : String) : Abstract_Prodotto_SW
    }
    class Prenotazione_SW {
        -idPrenotazione : String
        -caparra : double
        -numeroPersone : int
        -data : Date
        -confermata : boolean
        +create()
        +getIdPrenotazione() : String
        +setIdPrenotazione(idPrenotazione : String) : void
        +getCaparra() : double
        +setCaparra(caparra : double) : void
        +getNumeroPersone() : int
        +setNumeroPersone(numeroPersone : int) : void
        +getData() : Date
        +setData(data : Date) : void
        +getConfermata() : boolean
        +setConfermata(confermata : boolean) : void
        +modificaPrenotazione(prenotazione : Prenotazione_SW) : bo...
    }
    class Richiesta_SW {
        -tipologia : TipologiaRichiesta
        -inoltra : boolean
        -approvata : boolean
        +getTipologia() : TipologiaRichiesta
        +setTipologia(tipologia : TipologiaRichiesta) : void
        +getInoltra() : boolean
        +setInoltra(inoltra : boolean) : void
        +getApprovata() : boolean
        +setApprovata(approvata : boolean) : void
    }
    class Form {
        -numeroPersone : int
        -data : Date
        -informazioniAggiuntive : String
        +getNumeroPersone() : int
        +setNumeroPersone(numeroPersone : int) : void
        +getData() : Date
        +setData(data : Date) : void
        +getInformazioniAggiuntive() : String
        +setInformazioniAggiuntive(informazioniAggiuntive : String) : void
    }
    class TipologiaRichiesta {
        <<enumeration>>
        cancellazione
        modifica
        email
    }
    class UtenteObserver {
        <<interface>>
        +update()
    }
    PacchettoControllore --> UtenteRegistrato_SW : 1
    UtenteRegistrato_SW --> Operatore_SW : 1
    UtenteRegistrato_SW --> UtenteGenerico_SW : 1
    UtenteRegistrato_SW ..> UtenteObserver : <<use>>
    UtenteRegistrato_SW --> Prenotazione_SW : 1
    UtenteRegistrato_SW --> Richiesta_SW : 0..*
    UtenteRegistrato_SW --> Form : 1
    Prenotazione_SW --> Richiesta_SW : 0..*
    Prenotazione_SW --> Form : 1
    Richiesta_SW --> Operatore_SW : 0..*
    Richiesta_SW --> Form : 1
    
```

The diagram illustrates the 'Servizi' package structure. It includes several classes and interfaces:

- ComunicazioneEntiTerzi**: A class with methods `+inoltraOrganizzazioniTerze(prodotto : Abstract_Prodotto_SW) : boolean`, `+inoltraIstitutoDiCredito(cartacredito : CartaDiCredito_SW) : boolean`, and `+verificaOasiPersonali(utenteRegistrato_SW utenteReg) : boolean`.
- Sicurezza**: A class with a single association to **CartaDiCredito_SW**.
- Pagamento**: A class with associations to **UtenteRegistrato_SW** (multiplicity 0..*), **Prenotazione_SW** (multiplicity 1), and **CartaDiCredito_SW** (multiplicity 1..*). It contains methods `+create()`, `+getStato() : String`, and `+setStato(stato : String) : void`.
- CartaDiCredito_SW**: A class with attributes `-numero : String`, `-scadenza : Date`, `-cvv : int`, and `-titolare : String`. It includes methods `+create()`, `+getNumero() : String`, `+setNumero(numero : String) : void`, `+getScadenza() : Date`, `+setScadenza(scadenza : Date) : void`, `+getCvv() : int`, `+setCvv(cvv : int) : void`, `+getTitolare() : String`, `+setTitolare(titolare : String) : void`, and `+inizializzaDat(Pagamento(numero : String, scadenza : Date, cvv : int, titolare : String) : boolean`.
- Email**: An interface `<<interfaccia>> ISubscriber` with methods `+attach(utenteObserver u)`, `+detach(utenteObserver u)`, and `+notify()`. It has a generalization relationship with the **NotificaEmail** class. The **NotificaEmail** class has attributes `-email : Email` and `-subscriber : utenteObserver[]`, and methods `+getOggetto() : String`, `+setOggetto(oggetto : String) : void`, and `+inviaNotificaEmail(email : Email)`.

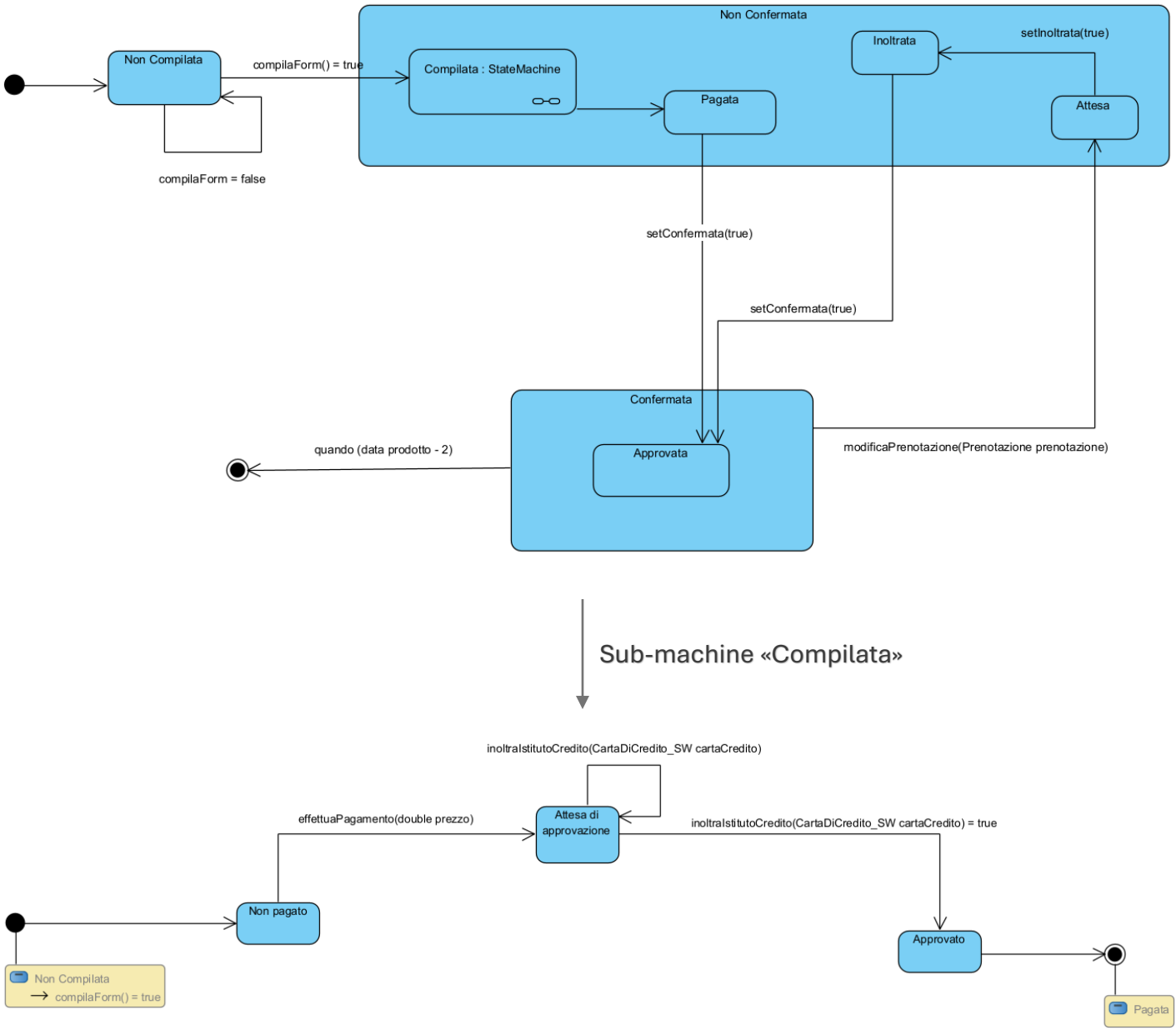
Associations are shown with solid lines and multiplicities. Generalization is indicated by a hollow triangle arrow pointing from **NotificaEmail** to **ISubscriber**.

Pattern applicati:

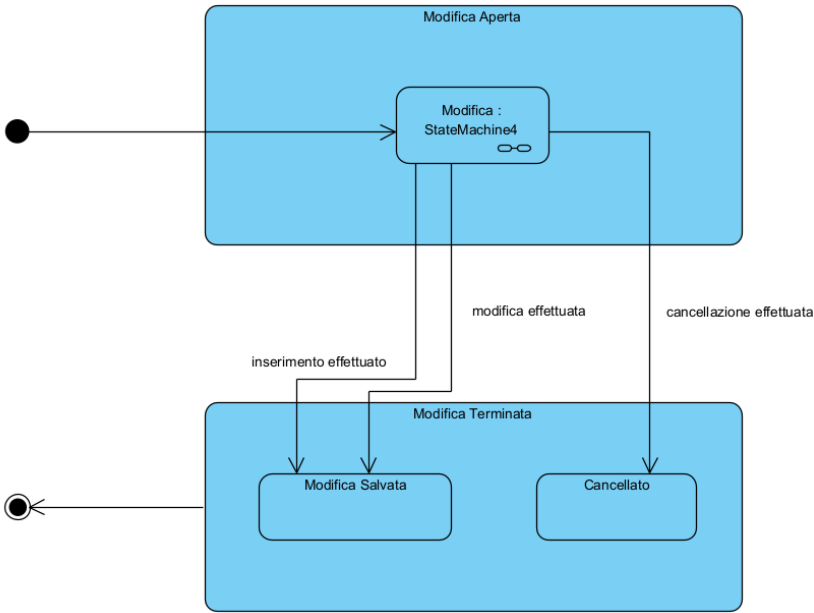
PATTERN	CLASSI COINVOLTE	UTILIZZO
Creator	Amministratore → Pacchetto PrenotazioneController → Prenotazione UtenteRegistrato → Pagamento PacchettoController → Pacchetto	La prima classe è responsabile della creazione della seconda
Controller	PacchettoController CatalogoController PrenotazioneController UtenteGenericoController UtenteRegistratoController AmministratoreController OperatoreController	Sono classi che fungono da intermediari per le classi «principali», ricevono e coordinano le operazioni di sistema. Un pattern correlato è pure fabrication , in quanto sostiene low coupling e high cohesion
Pure Fabrication	ComunicazioneEntiTerzi	Classi artificiali che riducono l'accoppiamento e aumentano la coesione tra classi nel sistema
Information Expert	Operatore OperatoreController	Si occupano di inoltrare le richieste in quanto ne conoscono il destinatario
Observer	Utente NotificaEmail	Tramite le interfacce si crea un meccanismo publisher-subscriber

Diagrammi di macchine a stati:

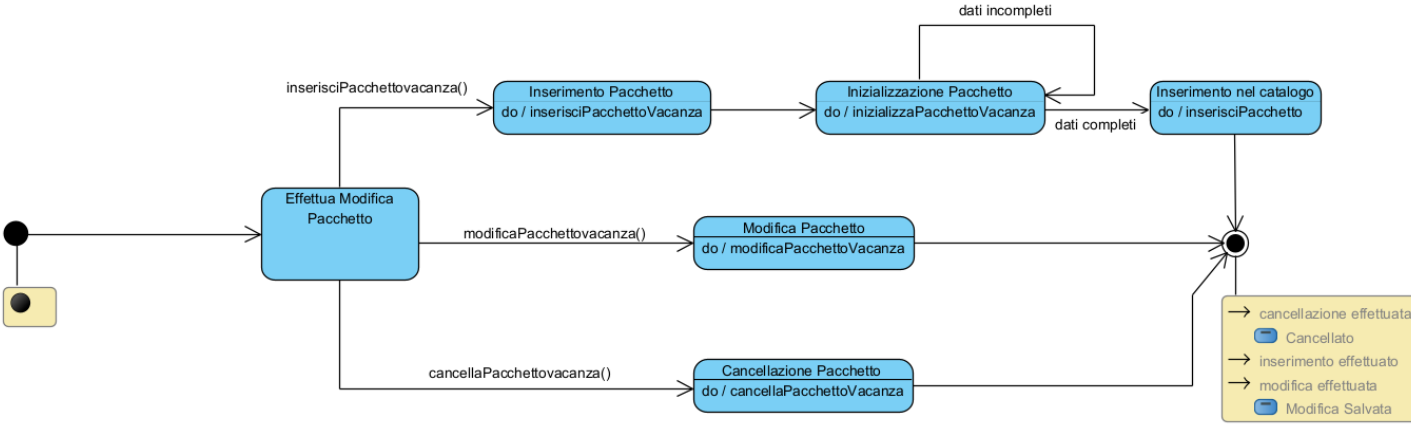
Prenotazione:



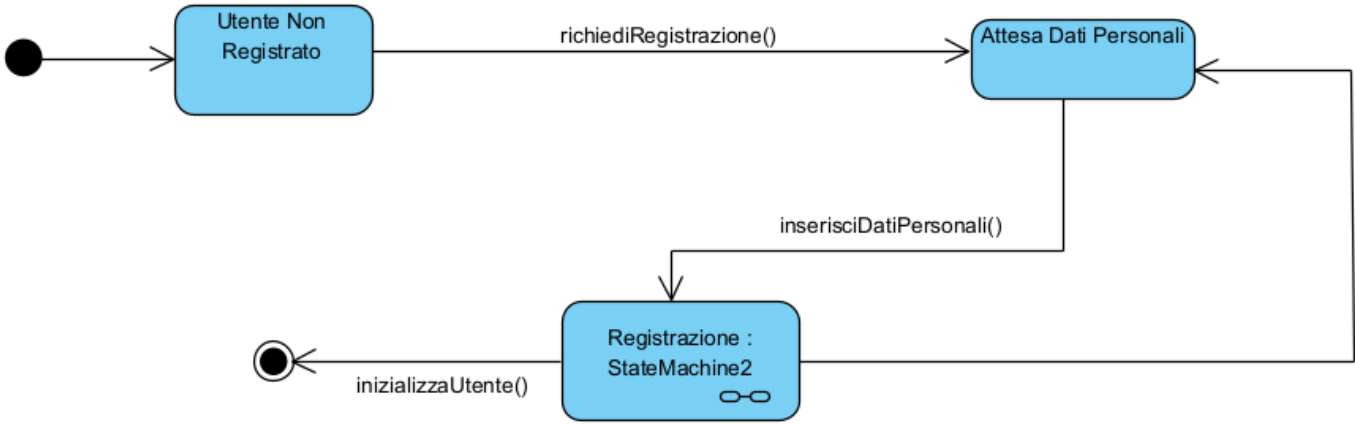
Pacchetto:



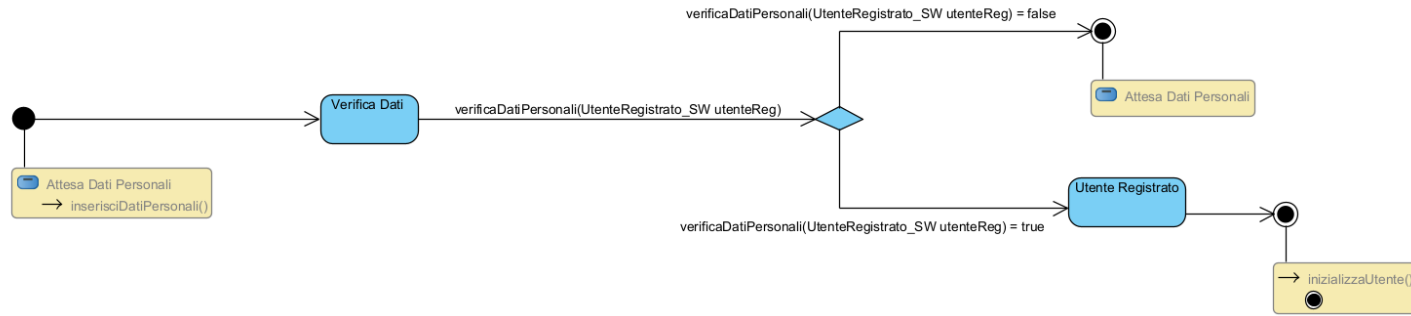
Sub-machine «Modifica»



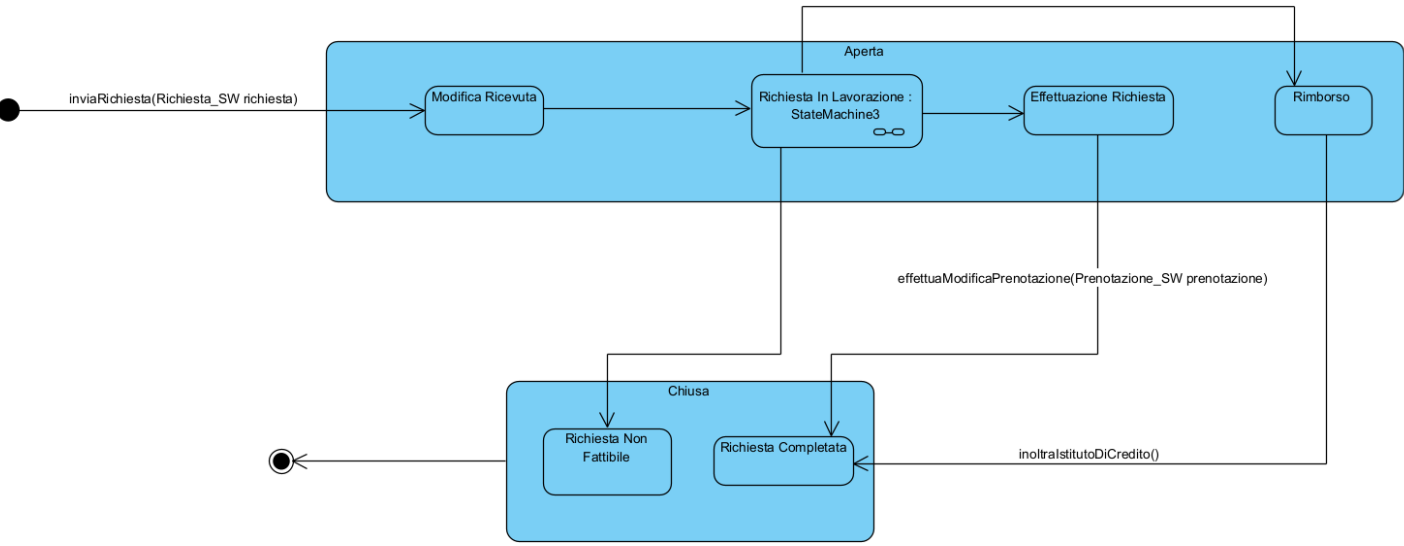
Utente - registrazione:



Sub-machine «Registrazione»



Richiesta utente:



Sub-machine «Richiesta in lavorazione»

