

UNIVERSITA' DEGLI STUDI DI NAPOLI FEDERICO II
FACOLTÀ DI INFORMATICA



Ingegneria del Software 2022/2023

Ratatouille23

Docenti:

Prof. Sergio Di Martino

Prof. Francesco Cutugno

Prof. Luigi Lucio Libero Starace



Gruppo INGSW2223_N_04

No.	Nome e Cognome	Matricola
1	Luigi Vessella	N86003354
2	Matteo Marino	N86003963
3	Biagio Speranza	N86002964



Indice

1	Descrizione del progetto	5
2	Documento dei requisiti software	6
2.1	Individuazione target d'utenti	6
2.2	Requisiti Funzionali	6
2.2.1	Admin	6
2.2.2	Cameriere	9
2.2.3	Cucina	9
2.2.4	Supervisore	10
2.2.5	Tutti	10
2.3	Requisiti Non Funzionali	12
2.4	Requisiti di Dominio	12
2.5	Modellazione dei casi d'uso	13
2.5.1	Use-Case Diagram	13
2.5.2	Tabelle di Cockburn	20
2.6	Mock-up dell'applicazione	27
2.6.1	Homepage dell'applicazione	27
2.6.2	Schermata di accesso nel sistema	29
2.6.3	Schermata di registrazione nel sistema	30
2.6.4	Schermata home per gli admin	31
2.6.5	Schermata di registrazione di un nuovo ristorante	32
2.6.6	Schermata di modifica di un ristorante	33
2.6.7	Schermata di registrazione di un nuovo dipendente	34
2.6.8	Schermata di gestione del menù	35
2.6.9	Schermata home per i supervisori	36
2.6.10	Schermata di conferma di logout	37
2.6.11	Schermata home per i camerieri	38
2.6.12	Schermata di visualizzazione dello stato degli ordini	39
2.6.13	Schermata di visualizzazione del profilo	40
2.6.14	Schermata di cambio password per gli admin	41
2.6.15	Schermata di cambio email per gli admin	42
2.6.16	Schermata di cambio password per i dipendenti al primo accesso	43
2.6.17	Schermata di creazione degli avvisi	44
2.6.18	Schermata di aggiunta piatti al menù	45
2.6.19	Schermata di aggiunta ordini	47
2.6.20	Schermata di visualizzazione delle statistiche	48
2.6.21	Schermata di visualizzazione delle notifiche	49
2.7	Valutazione dell'usabilità	50
2.8	Individuazione target d'utenti	51
2.9	Glossario	52
2.10	Class diagram di analisi o dominio	53
2.10.1	Class diagram Login	53
2.10.2	Class diagram Ristorante	54



2.10.3	Class diagram dipendenti	55
2.10.4	Class diagram Menu	56
2.10.5	Class diagram Avvisi	57
2.10.6	Class diagram Ordini	58
2.10.7	Class diagram statistiche	58
2.11	Sequence di analisi	59
2.12	Statechart	61
2.12.1	Aggiungi ristorante	61
2.12.2	Aggiungi piatto	62
2.12.3	Prendi ordine	63
2.12.4	Visualizza avvisi	64
3	Documento di design	65
3.1	Architettura del sistema	65
3.2	Introduzione	65
3.2.1	Client	65
3.2.2	Server	65
3.3	Sequence diagram di design	67
3.3.1	Funzionalità: "Aggiungi Ristorante"	67
3.3.2	Funzionalità: "Aggiungi portata"	67
3.3.3	Funzionalità: "Prendi ordinazione"	68
3.3.4	Funzionalità: "Visualizza avvisi"	68
3.4	Class diagram di design	69
3.4.1	Login	69
3.4.2	Gestione piatti	70
3.4.3	Gestione Avvisi	70



1 Descrizione del progetto

Ratatouille23 è un sistema finalizzato al supporto alla gestione e all'operatività di attività di ristorazione. Il sistema consiste in un'applicazione performante e affidabile, attraverso cui gli utenti possono fruire delle funzionalità del sistema in modo intuitivo, rapido e piacevole. La nostra visione della richiesta prevede lo sviluppo di un'applicazione mobile su sistema operativo Android che offrirà agli utilizzatori i seguenti servizi:

- *I proprietari (o amministratori) potranno creare account per i dipendenti*
- *I proprietari saranno in grado di gestire uno o più ristoranti*
- *Si avrà la possibilità di visualizzare e/o modificare il menu*
- *I dipendenti (camerieri) saranno in grado di prendere e inoltrare le ordinazioni in cucina*
- *Gli addetti alla cucina potranno avvisare i camerieri nel momento in cui è pronta un'ordinazione*
- *Tutti potranno visionare lo storico delle ordinazioni con i dettagli*

Ovviamente, il tipo di funzionalità messo a disposizione dall'applicativo sarà cambiato dinamicamente a seconda di chi si logga. Gli amministratori e/o supervisori saranno inoltre dotati di tablet con a bordo l'OS di Google Android per una migliore fruizione della loro dashboard.

I dipendenti quali camerieri, operatori di cucina, capisala, saranno dotati di smartphone aziendali sempre con OS Android correttamente configurati per un'ottimale fruizione dell'applicazione.

Tutti i dispositivi dovranno essere in grado di accedere a internet, preferibilmente per tutta la durata del servizio. Il funzionamento del server è invece garantito servizi allo stato dell'arte quali Microsoft Azure.



2 Documento dei requisiti software

2.1 Individuazione target d'utenti

Durante la progettazione di un nuovo software è fondamentale definire qual'è il target di utenti a cui riferirsi. Da una prima analisi dei casi d'uso è possibile individuare per la nostra applicazione quattro diverse utenze:

- Admin (o proprietario del ristorante)
- Supervisore
- Addetto alla cucina
- Addetti alla sala (principalmente i camerieri)

In seguito verranno studiate le suddette figure e verrà approfondito lo studio del target d'utenti.

2.2 Requisiti Funzionali

Vengono qui presentati i requisiti funzionali dell'applicativo, ossia quei servizi che l'app deve offrire agli utenti:

2.2.1 Admin

ID	Admin_1
Nome	Registrazione account amministratore
Descrizione	Il sistema permette ad un amministratore non registrato di registrarsi alla piattaforma utilizzando: <i>nome</i> , <i>cognome</i> , <i>email</i> , <i>codice fiscale</i> , <i>P.IVA</i> e <i>password</i>

ID	Admin_2
Nome	Modifica account amministratore
Descrizione	Il sistema permette ad un amministratore loggato di modificare i campi del proprio account

ID	Admin_3
Nome	Registrazione dei dipendenti
Descrizione	Il sistema permette ad un amministratore di creare utenze per i dipendenti non registrati del ristorante, specificandone <i>nome</i> , <i>cognome</i> , <i>email</i> e <i>ruolo</i>



ID	Admin_4
Nome	Modificare/Eliminare account dipendenti
Descrizione	Il sistema permette ad un amministratore loggato di modificare gli account dei propri dipendenti.

ID	Admin_5
Nome	Aggiungere personale della cucina
Descrizione	Il sistema permette ad un amministratore loggato di aggiungere al proprio ristorante il personale della cucina.

ID	Admin_6
Nome	Aggiunta dei Ristoranti
Descrizione	Il sistema permette ad un amministratore di poter aggiungere le proprie attività di ristorazione (CAMPI DA DEFINIRE).

ID	Admin_7
Nome	Modifica/Eliminazione dei Ristoranti
Descrizione	Il sistema permette ad un amministratore di poter modificare ed eliminare le proprie attività di ristorazione del sistema.

ID	Admin_8
Nome	Modifica dati dei Dipendenti
Descrizione	Il sistema permette ad un amministratore loggato di poter cambiare i dati personali dei dipendenti (nome, cognome, email, luogo).



ID	Admin_9
Nome	Aggiungere/Modificare elementi nel menù
Descrizione	Il sistema permette ad un amministratore o supervisore dell'attività di ristorazione di aggiungere/modificare elementi nel menù dell'attività. Ogni elemento dovrà avere i seguenti campi: Nome, Costo, Descrizione, Elenco di Allergeni, Categoria/e

ID	Admin_10
Nome	Modifica dati personali
Descrizione	Il sistema permette ad un amministratore loggato di poter cambiare i propri dati personali.

ID	Admin_12
Nome	Tradurre il menu
Descrizione	Il sistema permette ad un Amministratore di poter tradurre gli elementi del proprio menù in un'altra lingua

ID	Admin_13
Nome	Visualizza statistiche personale della cucina
Descrizione	Il sistema permette ad un Amministratore di visualizzare, grazie anche all'ausilio di grafici interattivi, informazioni sull'operato degli addetti alla cucina

ID	Admin_14
Nome	Modifica menu
Descrizione	Il sistema permette ad un Amministratore di poter aggiornare (aggiungere, modificare ed eliminare elementi) il menu del ristorante



2.2.2 Cameriere

ID	Waiter_1
Nome	Prendere le ordinazioni
Descrizione	Il sistema permette ai camerieri di prendere ordinazioni ai tavoli, inoltrandole alla cucina.

ID	Waiter_2
Nome	Gestione delle ordinazioni
Descrizione	Il sistema permette ai camerieri di prendere ordinazioni ai tavoli, inoltrandole alla cucina.

ID	Waiter_3
Nome	Evasione degli ordini
Descrizione	Il sistema permette a un Cameriere di marcare i singoli elementi di un ordine come conclusi, aggiornando gli addetti in cucina

ID	Waiter_4
Nome	Sollecitare la cucina
Descrizione	Il sistema permette a un Cameriere di sollecitare la cucina nel caso in cui un ordine sia da troppo tempo in preparazione

2.2.3 Cucina

ID	Kitchen_1
Nome	Marcare gli ordini pronti
Descrizione	Il sistema permette alla cucina di marcare gli ordini pronti alla "consegna ", specificando lo chef che l'ha preparato

ID	Kitchen_2
Nome	Sollecitare i camerieri
Descrizione	Il sistema permette alla cucina di notificare i camerieri nel caso in cui un ordine sia pronto alla consegna da troppo tempo



2.2.4 Supervisore

ID	Hypervisor_1
Nome	Avvisare il personale
Descrizione	Il sistema permette ad un supervisore ed un amministratore di inviare degli avvisi al personale

ID	Hypervisor_2
Nome	Visualizzare stato ordini per tavolo
Descrizione	Il sistema permette ad un Supervisore ed un Cameriere di visualizzare gli stati delle ordinazioni per ogni tavolo

ID	Hypervisor_3
Nome	Visualizzare ordini in arrivo
Descrizione	Il sistema permette ad un Supervisore e alla Cucina di visualizzare l'elenco di ordini in arrivo

ID	Hypervisor_4
Nome	Visualizzare ordini in uscita
Descrizione	Il sistema permette ad un Supervisore e alla Cucina di visualizzare l'elenco di ordini pronti all'uscita

ID	Hypervisor_5
Nome	Visualizzare storico ordini
Descrizione	Il sistema permette ad un Supervisore e alla Cucina di visualizzare lo storico degli ordini

2.2.5 Tutti

ID	All_1
Nome	Recupero/Cambio Password
Descrizione	Il sistema permette a tutti gli utenti registrati di poter recuperare la password e di poterla modificare



ID	All_2
Nome	Login
Descrizione	Il sistema permette a tutti gli utenti registrati di poter effettuare il login con Username e Password all'interno della piattaforma

ID	All_3
Nome	Visualizzare un avviso o sollecitazione
Descrizione	Il sistema permette a tutti gli utenti registrati di poter visualizzare gli avvisi e le sollecitazioni ricevuti, marcandoli come visualizzati



2.3 Requisiti Non Funzionali

Vengono qui elencati i requisiti non funzionali dell'applicativo:

ID	Unfunctional_1
Nome	Policy first password
Descrizione	Il sistema richiede al primo accesso di un dipendente il cambio della password provvisoria in una password personale.

ID	Unfunctional_2
Nome	Verifica esistenza della mail
Descrizione	Al fine di evitare registrazioni con e-mail fittizie, il sistema richiede l'autenticazione della mail mediante codice di verifica per poter procedere con il completamento della registrazione

ID	Unfunctional_3
Nome	Password Strength
Descrizione	Al fine di evitare la creazione di password poco sicure, il sistema impone all'utente di utilizzare una password di almeno 8 caratteri che contenga numeri e caratteri speciali.

ID	Unfunctional_4
Nome	Unique Commercial Code
Descrizione	Una P.IVA appartiene ad un solo amministratore.

2.4 Requisiti di Dominio

ID	Domain_1
Nome	GDPR
Descrizione	Il sistema deve essere conforme alla normativa GDPR (Regolamento Generale sulla Protezione dei Dati), per il trattamento dei dati personali e riguardante la privacy dell'utente



2.5 Modellazione dei casi d'uso

In questa sezione vengono riportati i Use-case diagram relativi al sistema e le tabelle di Cockburn e i Sequence diagram relativi alle funzionalità *Aggiungi ristorante*, *Aggiungi piatto*, *Prendi ordinazione*, *Visualizza avvisi*.

2.5.1 Use-Case Diagram

Qui viene riportato lo Use Case Diagram relativo all'applicazione, che per rendere la lettura più semplificata è stato diviso in più sezioni.

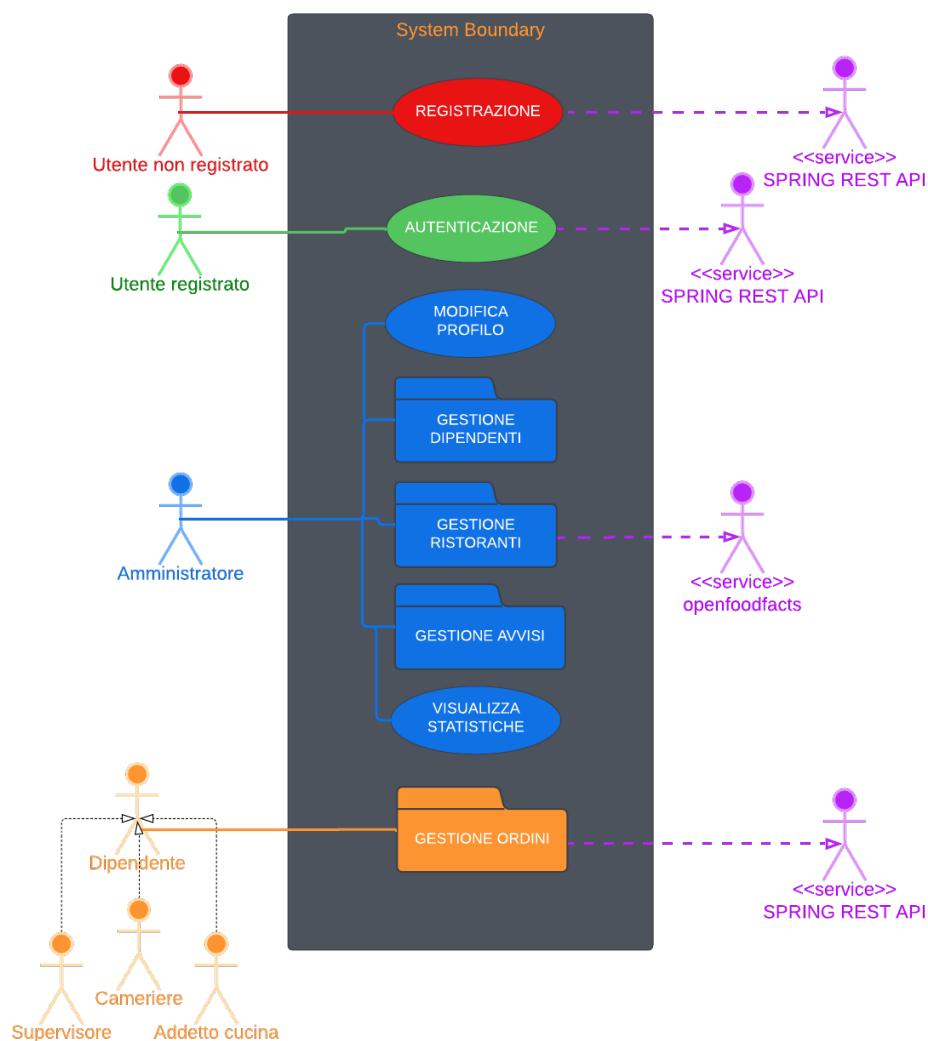


Figura 2.1: Use Case dell'applicazione

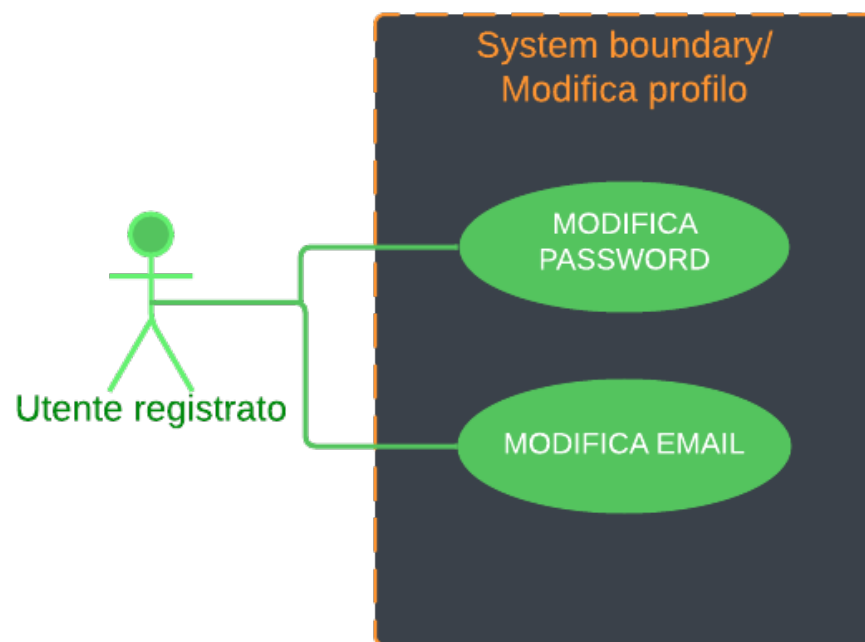


Figura 2.2: Use Case relativo alla modifica del profilo

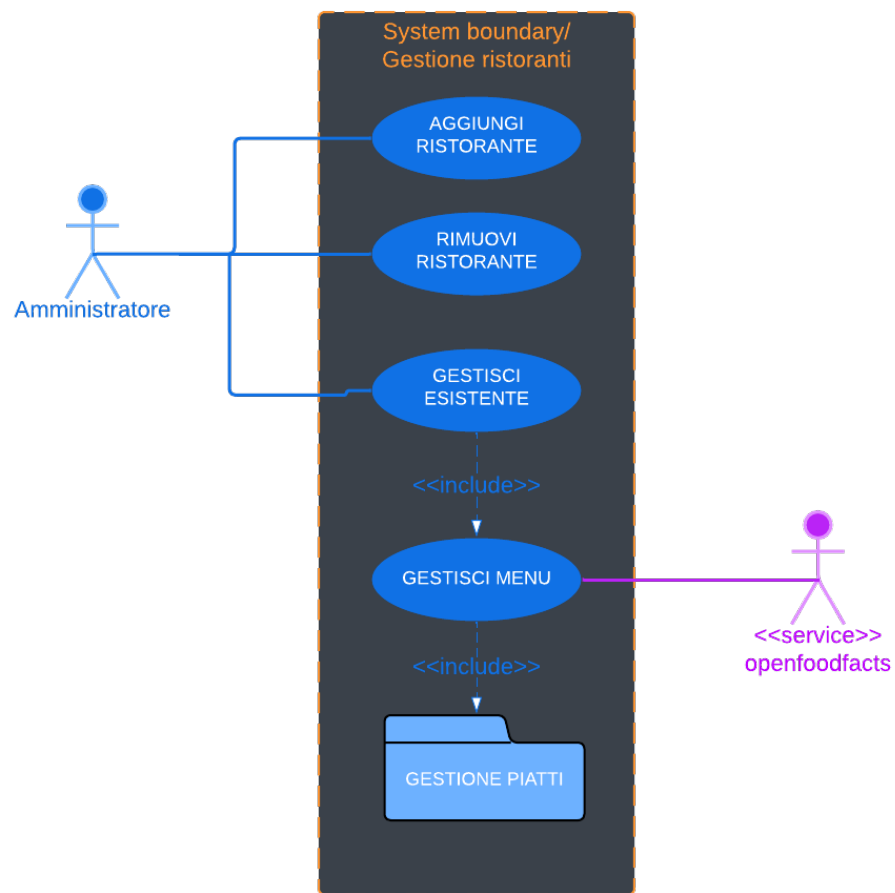


Figura 2.3: Use Case relativo alla gestione dei ristoranti

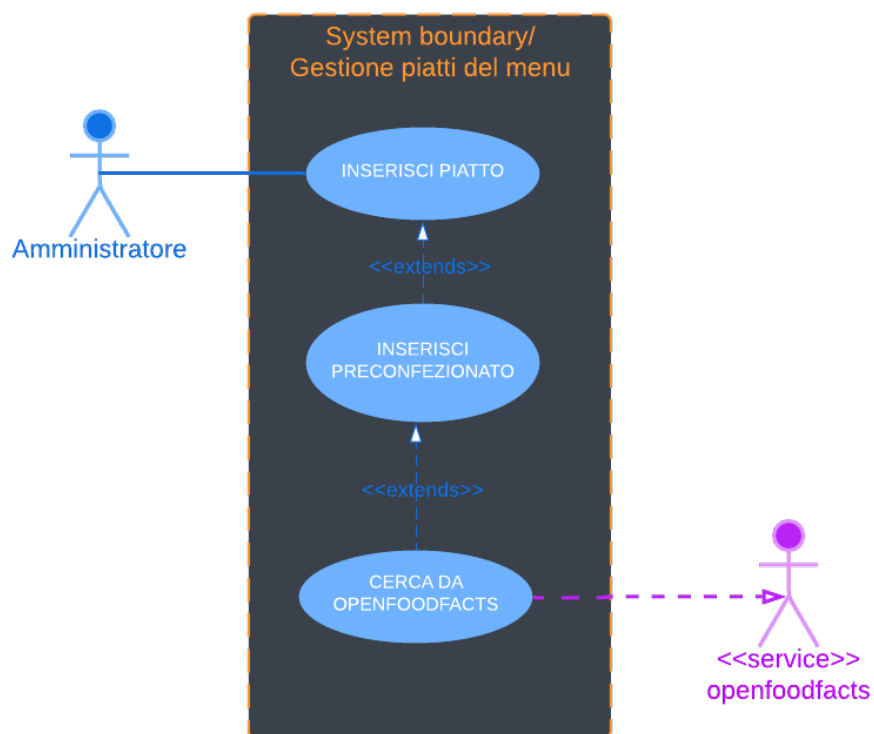


Figura 2.4: Use Case relativo alla gestione dei piatti del menù

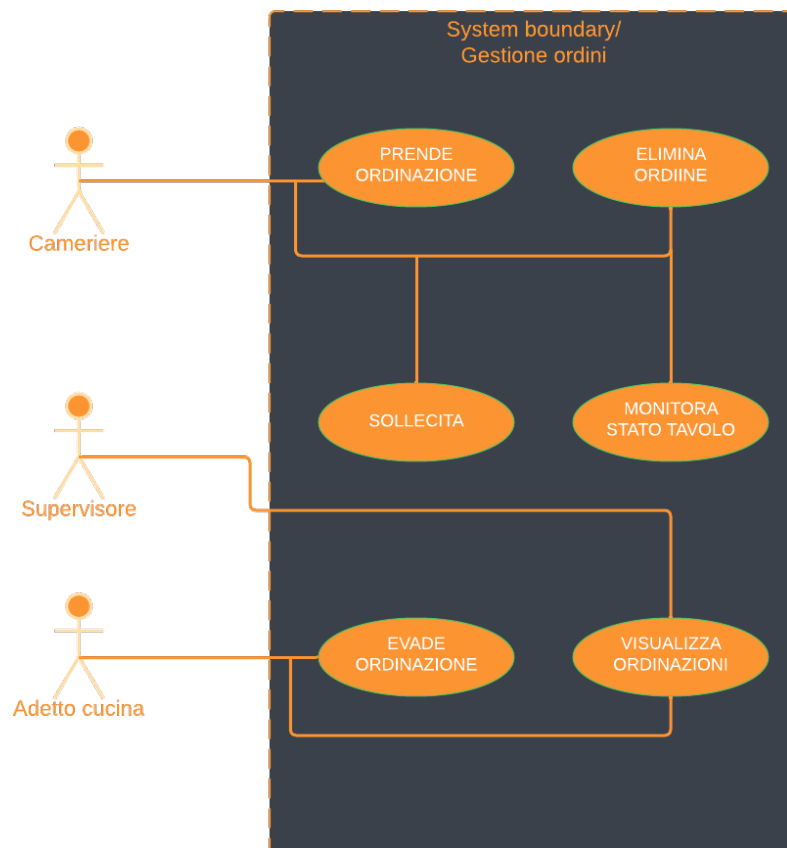


Figura 2.5: Use Case relativo alla gestione degli ordini

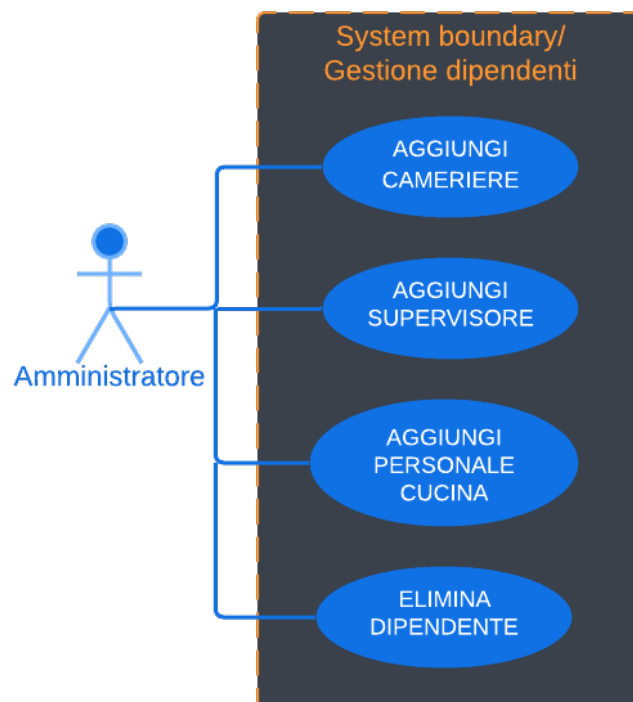


Figura 2.6: Use Case relativo alla gestione dei dipendenti

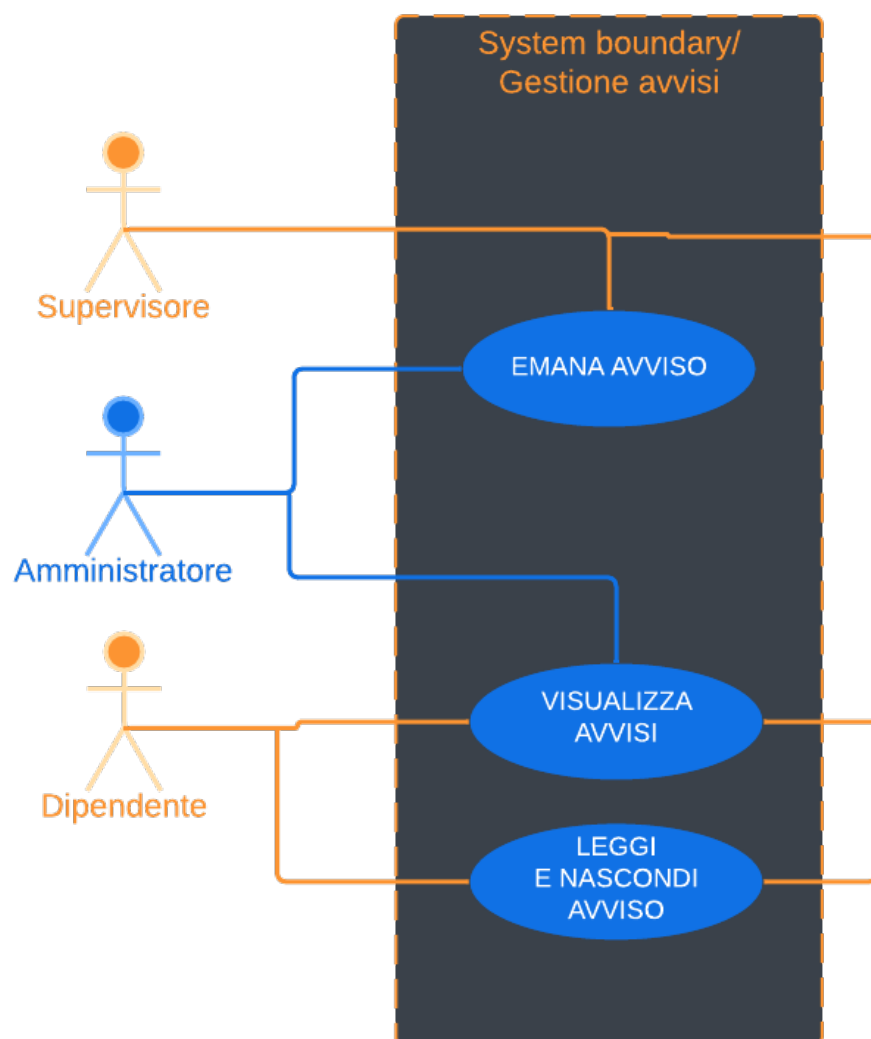


Figura 2.7: Use Case relativo alla gestione degli avvisi



2.5.2 Tabelle di Cockburn

Vengono qui riportate le tabelle di cockburn dei casi d'uso *Aggiungi ristorante*, *Aggiungi piatto*, *Prendi ordinazione*, *Visualizza avvisi*.

Tabella 1:

Use Case #1	Aggiunta ristorante		
Goal in Context	Un admin (proprietario di uno o più ristoranti) vuole aggiungere uno di questi nell'app Ratatuille.		
Precodition	Il proprietario deve essere registrato e loggato nell'app come amministratore		
Success End Condition	Il proprietario aggiunge correttamente il ristorante		
Failed End Condition	Il proprietario non riesce ad aggiungere il ristorante		
Primary Actor	Amministratore		
Trigger	Preme il pulsante <i>Aggiungi</i>		
Description	Step	UserAction	System
	1	L'amministratore preme sul tasto " <i>AGGIUNGI RISTORANTE</i> " sulla schermata <i>M04</i>	
	2		Mostra la schermata <i>M05</i>
	3	Compila i campi necessari per la registrazione del proprio ristorante e preme sul tasto " <i>SALVA</i> " per aggiungere il Ristorante	
	4		Ricarica la schermata <i>M04</i> aggiungendo nella lista dei ristoranti l'ultimo appena inserito
Extension # 1 L'amministratore non fa nulla e torna indietro	Step	UserAction	System
	4a		Torna alla schermata <i>M04</i>

Continued on next page



Tabella 1: (Continued)

Use Case #1	Aggiunta ristorante		
Extension #2 L'amministratore ha aggiunto un ristorante già presenta nella propria lista di ristoranti	Step	UserAction	System
	4b		Mostra uno dei messaggi di errore della schermata M0boh restando allo step 2 dello scenario principale
Extension #3 L'amministratore ha lasciato uno o più campi vuoti	Step	UserAction	System
	4c		Mostra uno o più dei messaggi di errore della schermata M05 restando allo step 2 dello scenario principale
Extension #4 L'amministratore ha aggiunto un nome troppo corto	Step	UserAction	System
	4d		Mostra il messaggio di errore di fianco al campo "Nome" della schermata M05 restando allo step 2 dello scenario principale
Extension #5 L'amministratore ha aggiunto un numero di coperti non valido	Step	UserAction	System
	4e		Mostra il messaggio di errore di fianco al campo "Numero di coperti" della schermata M05 restando allo step 2 dello scenario principale
Extension #6 L'amministratore ha aggiunto un indirizzo troppo corto	Step	UserAction	System
	4f		Mostra il messaggio di errore di fianco al campo "Indirizzo" della schermata M05 restando allo step 2 dello scenario principale

Continued on next page



Tabella 1: (Continued)

Use Case #1	Aggiunta ristorante		
Extension #7 L'amministratore ha aggiunto un numero di telefono non valido	Step	UserAction	System
	4g		Mostra il messaggio di errore di fianco al campo "Numero di telefono" della schermata M05 restando allo step 2 dello scenario principale
Subvariation #1 L'amministratore torna indietro completando solo parzialmente i campi	Step	UserAction	System
	1		Mostra la schermata M0boh
	2	Preme su "SI"	
	3		Torna alla schermata M04
Subvariation #2 L'amministratore inizialmente vuole tornare indietro completando solo parzialmente i campi ma cambia idea	Step	UserAction	System
	1		Mostra la schermata M0boh
	2	Preme su "NO"	
	3		Torna alla schermata M05 allo step 2 dello scenario principale



Tabella 2:

Use Case #2	Prendere un ordine		
Goal in Context	Un cameriere vuole prendere l'ordinazione di un tavolo		
Precodition	Il cameriere deve essere registrato e loggato nell'app e il ristorante deve avere un menu con dei piatti al suo interno		
Success End Condition	Il cameriere prendere correttamente un'ordinazione		
Failed End Condition	Il cameriere non prende l'ordinazione		
Primary Actor	Cameriere		
Trigger	Preme il pulsante <i>NUOVO ORDINE</i>		
Description	Step	UserAction	System
	1	Il cameriere preme sul pulsante " <i>NUOVO ORDINE</i> " sulla schermata <i>M11</i>	
	2		Mostra la schermata <i>M19</i> con i vari piatti del menu
	3	Il cameriere seleziona il numero del tavolo, inserisce le portate all'interno dell'ordine e preme il pulsante " <i>SALVA</i> "	
	4		Torna alla schermata <i>M11</i>
Extension # 1 Il cameriere non fa nulla e torna indietro	Step	UserAction	System
	4a		Torna alla schermata <i>M11</i>
Subvariation #1 Il cameriere torna indietro dopo aver aggiunto dei piatti all'ordine senza salvare	Step	UserAction	System
	1		Mostra la schermata <i>M0boh</i>
	2	Preme su " <i>SI</i> "	
	3		Torna alla schermata <i>M11</i>

Continued on next page



Tabella 2: (Continued)

Use Case #2	Prendere un ordine		
Subvariation #2 L'amministratore inizialmente vuole tornare indietro completando solo parzialmente i campi ma cambia idea	Step	UserAction	System
	1		Mostra la schermata M0boh
	2	Preme su "NO"	
	3		Torna alla schermata M11

Tabella 3:

Use Case #3	Aggiungi piatto al menu		
Goal in Context	Un amministratore vuole aggiungere un piatto al proprio menu		
Precodition	Il proprietario deve essere registrato e loggato nell'app come amministratore, deve aver aggiunto almeno un ristorante e deve averne creato un menu		
Success End Condition	L'amministratore aggiunge correttamente un piatto al suo menu		
Failed End Condition	L'amministratore non riesce ad aggiungere un piatto al suo ristorante		
Primary Actor	Amministratore		
Trigger	Preme il pulsante <i>Aggiungi prodotto</i>		
Description	Step	UserAction	System
	1	L'amministratore preme sul pulsante <i>Aggiungi prodotto</i> sulla schermata M08	
	2		Mostra la schermata M18
	3	L'amministratore compila il campi del piatto e preme sul pulsante <i>OK</i>	
	4		Torna alla schermata M08
Extension # 1 L'amministratore non fa nulla e torna indietro	Step	UserAction	System
	4a		Torna alla schermata M08

Continued on next page



Tabella 3: (Continued)

Use Case #3	Aggiungi piatto al menu		
Extension # 2 L'amministratore compila solo parzialmente i campi e preme sul pulsante <i>OK</i>	Step	UserAction	System
	4a		Mostra l'errore nella schermata <i>M0boh</i>
Subvariation #1 L'amministratore vuole aggiungere un prodotto preconfezionato al proprio menu	Step	UserAction	System
	1		Mostra la schermata <i>M18</i>
	2	L'amministratore scrive il nome (completo o parziale) di un prodotto e preme il pulsante <i>Cerca</i>	
	3		Mostra i prodotti corrispondenti alla ricerca
	4	L'amministratore seleziona il prodotto desiderato e preme il pulsante <i>OK</i>	
	5		Torna alla schermata <i>M08</i>



Tabella 4:

Use Case #4	Visualizza avvisi		
Goal in Context	Un dipendente vuole visualizzare un avviso		
Precondition	Il dipendente deve essere registrato e loggato nell'app		
Success End Condition	Il dipendente visualizza gli avvisi		
Failed End Condition	Il dipendente non riesce a visualizzare i propri avvisi		
Primary Actor	Dipendente		
Trigger	Preme il pulsante <i>Avvisi</i> nella propria Dashboard		
Description	Step	UserAction	System
	1	Il Dipendente preme sul pulsante <i>Avvisi</i> sulla schermata <i>M11</i> o <i>M09</i> o <i>M12</i>	
	2		Mostra la schermata <i>M21</i>
Extension # 1 Non ci sono avvisi da mostrare	Step	UserAction	System
	4a		Mostra la schermata <i>M21</i>
Subvariation #1 Il Dipendente vuole marcare un avviso come visualizzato	Step	UserAction	System
	1	Il Dipendente scorre col dito sull'avviso	
	2		Aggiorna la schermata <i>M21</i> eliminando l'avviso



2.6 Mock-up dell'applicazione

L'interfaccia utente ha il compito di “filtrare” la complessità, presentando all'utente un'immagine semplificata del prodotto, e congruente con i compiti che egli deve svolgere (Figura 15). Una buona interfaccia non solo nasconde la complessità interna del sistema, ma ne riduce la complessità funzionale, mettendo a disposizione dell'utente funzioni di più alto livello, in grado di effettuare compiti complessi con un grado di automatismo maggiore. (da "Facile da Usare") Qui vengono presentati i mockup relativi all'applicazione.

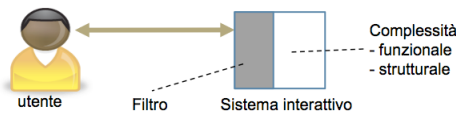


Figura 2.8: L'interfaccia Utente

2.6.1 Homepage dell'applicazione



Figura 2.9: **M01**: Homepage dell'applicazione

ID **M01**

Componenti:



Tipo	Nome	Funzione
Bottone	ACCEDI	Quando cliccato porta alla schermata <i>M02</i>
Bottone	REGISTRATI	Quando cliccato porta alla schermata <i>M03</i>



2.6.2 Schermata di accesso nel sistema

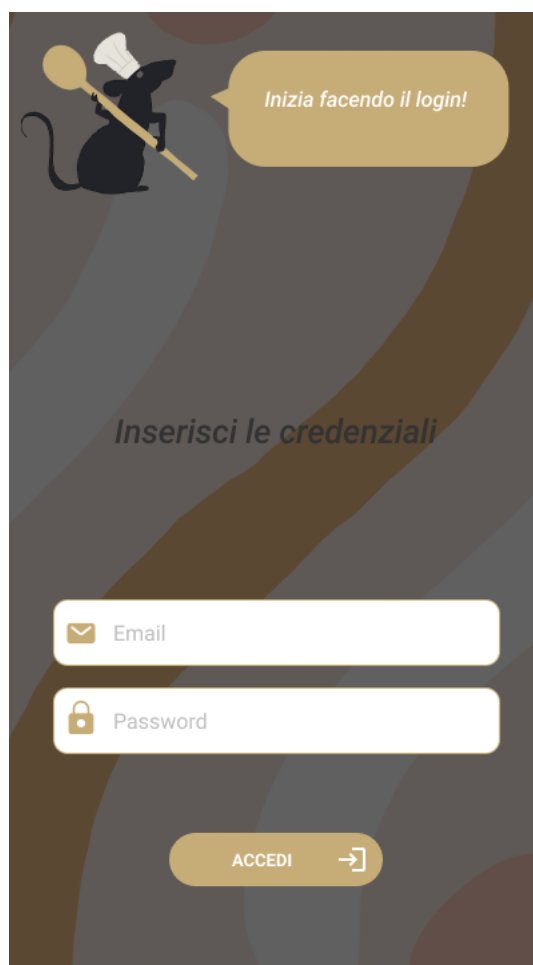


Figura 2.10: **M02**: Schermata di accesso nel sistema

ID **M02**

Componenti:

Tipo	Nome	Funzione
Edit Text	EMAIL	Permette l'inserimento dell'email dell'utente
Edit Text	PASSWORD	Permette l'inserimento della password dell'utente
Bottone	ENTRA	Quando cliccato porta alla schermata M04 se admin, M09 se supervisore, M11 se cameriere, M12 se account della cucina



2.6.3 Schermata di registrazione nel sistema

Figura 2.11: **M03**: Schermata di registrazione nel sistema

ID **M03**

Nota: la schermata di registrazione è valida solo per gli admin proprietari dei ristoranti, in quanto sono poi loro a registrare i dipendenti.

Componenti:

Tabella 5:

Tipo	Nome	Funzione
Edit Text	NOME	Permette di inserire il nome dell'admin
Edit Text	COGNOME	Permette di inserire il cognome dell'admin
Edit Text	PASSWORD	Permette di inserire una password per l'admin
Edit Text	EMAIL	Permette di inserire l'email dell'admin
Edit Text	CODICE FISCALE	Permette di inserire il codice fiscale dell'admin
Edit Text	P.IVA	Permette di inserire la partita IVA dell'admin
Bottone	REGISTRATI	Quando cliccato riporta alla schermata M01 per permettere l'accesso



2.6.4 Schermata home per gli admin



Figura 2.12: **M04**: Schermata home per gli admin

ID **M04**

Componenti:

Tabella 6:

Tipo	Nome	Funzione
ScrollView	I TUOI RISTORANTI	Visualizza ed eventualmente permette la modifica dei ristoranti registrati
Bottone	AGGIUNGI RISTORANTE	Quando cliccato porta alla schermata M05
Bottone	MODIFICA	Quando cliccato porta alla schermata M06
Bottone	PROFILO	Quando cliccato porta alla schermata M13
Bottone	ESCI	Quando cliccato mostra la schermata M10
Bottone	STATISTICHE	Quando cliccato porta alla schermata M20



2.6.5 Schermata di registrazione di un nuovo ristorante

Figura 2.13: **M05**: Schermata di registrazione di un nuovo ristorante

ID **M05**

Componenti:

Tipo	Nome	Funzione
Edit Text	NOME	Permette di inserire il nome del nuovo ristorante
Edit Text	LOCALITA'	Permette di inserire la località del nuovo ristorante
Edit Text	COPERTI	Permette di inserire il n° dei coperti del nuovo ristorante
Edit Text	NUMERO DI TELEFONO	Permette di inserire il contatto telefonico del ristorante
Bottone	SALVA	Quando cliccato salva il nuovo ristorante nel database e torna alla schermata M04



2.6.6 Schermata di modifica di un ristorante



Figura 2.14: **M06**: Schermata di modifica di un ristorante

ID **M06**

Componenti:

Tipo	Nome	Funzione
Bottone	AGGIUNGI DIPENDENTE	Quando cliccato porta alla schermata M07
Bottone	GESTISCI MENU	Quando cliccato porta alla schermata M08
ScrollView	IL TUO PERSONALE	Mostra il personale che lavora nel ristorante, dove se si tiene premuto il bottone <i>Elimina</i> viene eliminato l'account del dipendente scelto dal ristorante
Bottone	CREA AVVISO	Quando cliccato mostra la schermata M17



2.6.7 Schermata di registrazione di un nuovo dipendente

Figura 2.15: **M07**: Schermata di registrazione di un nuovo dipendente

ID **M07**

Componenti:

Tipo	Nome	Funzione
Edit Text	NOME	Permette di inserire il nome del nuovo dipendente
Edit Text	COGNOME	Permette di inserire il cognome del nuovo dipendente
Edit Text	EMAIL	Permette di inserire l'email del nuovo dipendente
Edit Text	CODICE FISCALE	Permette di inserire il codice fiscale del nuovo dipendente
Spinner	cameriere (default)	Permette di inserire il ruolo del nuovo dipendente
Bottone	REGISTRA	Quando cliccato, se tutti i dati sono corretti, riporta alla schermata M04 registrando il nuovo dipendente



2.6.8 Schermata di gestione del menù



Figura 2.16: **M08**: Schermata di gestione del menù

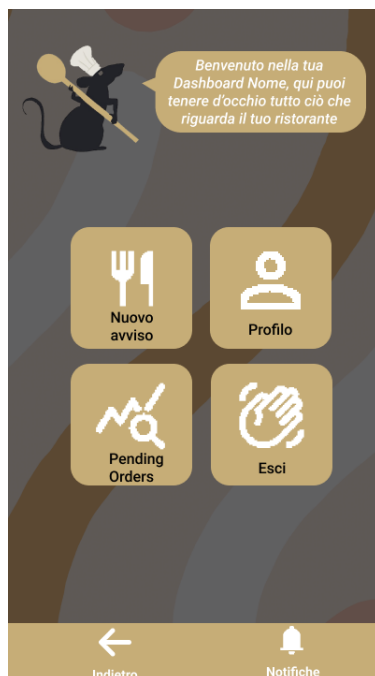
ID **M08**

Componenti:

Tipo	Nome	Funzione
Bottone	AGGIUNGI PRODOTTO	Quando cliccato porta alla schermata M18
Edit Text	SCARICA MENU	Permette di generare il menù e salvarlo sul dispositivo in formato PDF
Bottone	CREA MENU	Quando cliccato permette di creare un menu (se non è già stato creato)
Bottone	CANCELLA MENU	Permette di cancellare il menu del ristorante (se esistente)



2.6.9 Schermata home per i supervisor



M09: Schermata home dei supervisor

ID **M09**

Componenti:

Tipo	Nome	Funzione
Bottone	PROFILO	Quando cliccato porta alla schermata M13
Bottone	NUOVO AVVISO	Quando cliccato porta alla schermata M21
Bottone	PENDING ORDERS	Quando cliccato porta alla schermata M12
Bottone	ESCI	Quando cliccato porta alla schermata M10
Bottom Navigation Bar	BARRA NAVIGAZIONE	Permette di visualizzare le notifiche cliccando su <i>Notifiche</i> che porterà alla schermata M21 e di tornare indietro cliccando su <i>Indietro</i>



2.6.10 Schermata di conferma di logout



M10: Schermata di conferma logout

ID *M10*

Componenti:

Tipo	Nome	Funzione
Bottone	SI	Quando cliccato porta alla schermata <i>M02</i>
Bottone	NO	Quando cliccato resta nella schermata corrente



2.6.11 Schermata home per i camerieri



M11: Schermata home per i camerieri

ID **M11**

Componenti:

Tipo	Nome	Funzione
Bottone	NUOVO ORDINE	Quando cliccato porta alla schermata M19
Bottone	STATO ORDINE	Quando cliccato porta alla schermata M12
Bottom Navigation Bar	BARRA NAVIGAZIONE	Permette di visualizzare le notifiche cliccando su <i>Notifiche</i> che porterà alla schermata M21 e di tornare indietro cliccando su <i>Indietro</i>



2.6.12 Schermata di visualizzazione dello stato degli ordini



M12: Schermata home per i camerieri

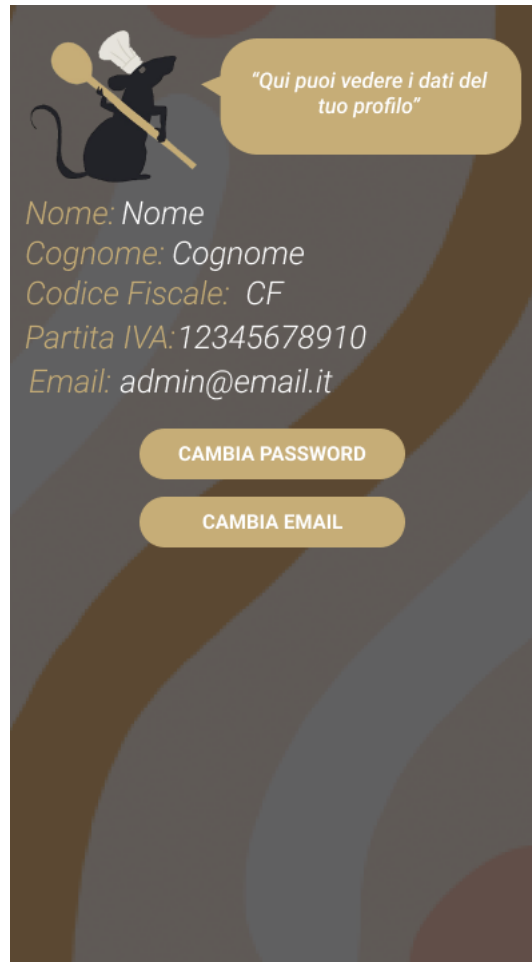
ID **M12**

Componenti:

Tipo	Nome	Funzione
RecyclerView	ORDINI	Permette di visualizzare la lista degli ordini, dove per i camerieri vi è la possibilità di sollecitare la cucina su un ordine e per la cucina di evaderlo eliminandolo
Bottom Navigation Bar	BARRA NAVIGAZIONE	Permette di visualizzare le notifiche cliccando su <i>Notifiche</i> che porterà alla schermata M21 e di tornare indietro cliccando su <i>Indietro</i>



2.6.13 Schermata di visualizzazione del profilo



M13: Schermata visualizzazione profilo

ID **M13**

Componenti:

Tipo	Nome	Funzione
Bottone	CAMBIA PASSWORD	Quando cliccato mostra la schermata M14
Bottone	CAMBIA EMAIL	Quando cliccato mostra la schermata M15



2.6.14 Schermata di cambio password per gli admin

Qui puoi vedere i dati del tuo profilo

Nome: Nome
Cognome: Cognome
Codice Fiscale: CF
Partita IVA: 12345678910
Email: admin@unina.it

Nuova Password

ANNULLA SALVA

M14: Schermata cambio password admin

ID **M14**

Componenti:

Tipo	Nome	Funzione
Edit Text	NUOVA PASSWORD	Permette di inserire la nuova password per l'account dell'admin
Bottone	ANNULLA	Quando cliccato torna alla schermata M13 senza effettuare modifiche
Bottone	SALVA	Quando cliccato torna alla schermata M13 modificando la password dell'account



2.6.15 Schermata di cambio email per gli admin

Qui puoi vedere i dati del tuo profilo

Nome: Nome
Cognome: Cognome
Codice Fiscale: CF
Partita IVA: 12345678910
Email: admin@unina.it

Nuova Email

ANNULLA SALVA

M15: Schermata cambio email admin

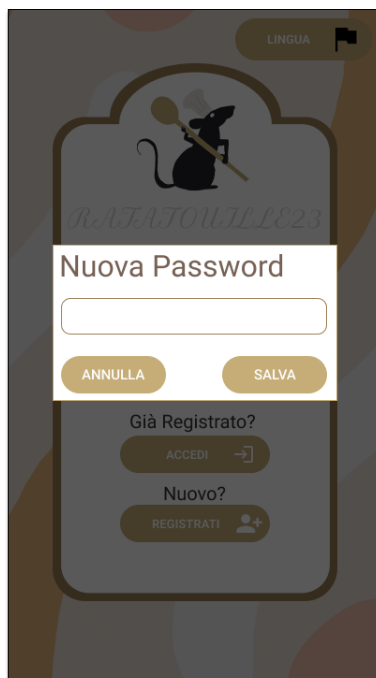
ID **M15**

Componenti:

Tipo	Nome	Funzione
Edit Text	NUOVA EMAIL	Permette di inserire la nuova email per l'account dell'admin
Bottone	ANNULLA	Quando cliccato torna alla schermata M13 senza effettuare modifiche
Bottone	SALVA	Quando cliccato torna alla schermata M13 modificando l'email dell'account



2.6.16 Schermata di cambio password per i dipendenti al primo accesso



M16: Schermata cambio password dipendenti

ID *M16*

Componenti:

Tipo	Nome	Funzione
Edit Text	NUOVA PASSWORD	Permette di inserire la nuova password per l'account del dipendente
Bottone	ANNULLA	Quando cliccato torna alla schermata <i>M02</i> senza effettuare modifiche
Bottone	SALVA	Quando cliccato porta alla schermata <i>M09</i> se è supervisore, <i>M11</i> se è cameriere, <i>M12</i> se è un account di cucina, modificando l'email dell'account



2.6.17 Schermata di creazione degli avvisi

Ristorante: Ristorante1

Aggiungi dipendente Gestisci Menu

INVIA

Il tuo personale (tieni premuto per eliminare)

cameriere	Elimina
cucina	Elimina
supervisore	Elimina

M17: Schermata creazione avvisi

ID *M17*

Componenti:

Tipo	Nome	Funzione
Edit Text	AVVISO	Permette di inserire il testo dell'avviso
Bottone	INVIA	Quando cliccato torna alla schermata <i>M06</i> inviando l'avviso



2.6.18 Schermata di aggiunta piatti al menù

Aggiungi i piatti al menu, oppure visualizzalo in PDF

Nome prodotto ... Cerca

Descrizione ...

Allergeni ...

Prezzo ...

Tipo Antipasto ▼ Tipo alimento Pesce ▼

Annulla Ok

Preconfezionato ☐

Scarica Menu

M18: Schermata aggiunta piatti al menù

ID **M18**

Componenti:

Tabella 7:

Tipo	Nome	Funzione
EditText	NOME PRODOTTO	Permette di inserire il titolo del prodotto da inserire
EditText	DESCRIZIONE	Permette di inserire la descrizione del prodotto da Inserire
EditText	ALLERGENI	Permette di inserire gli allergeni contenuti nel prodotto

Continued on next page



Tabella 7: (Continued)

Tipo	Nome	Funzione
EditText	PREZZO	Permette di inserire il prezzo del prodotto
Spinner	TIPO	Permette di inserire il tipo di portata (antipasto, primo, ...)
Spinner	TIPO DI ALIMENTO	Permette di inserire il tipo di alimento (carne, pesce, ...)
Bottone	CERCA	Quando cliccato, ricerca il prodotto scritto nel database di OpenFoodFacts
Bottone	OK	Quando cliccato, ritorna alla schermata M08 salvando nel menu il prodotto
Bottone	ANNULLA	Quando cliccate, ritorna alla schermata M08 senza salvare il prodotto
Selettore	PRECONFEZIONATO	Se selezionato, indica che l'oggetto è di tipo preconfezionato(bibita, ...)



2.6.19 Schermata di aggiunta ordini



M19: Schermata aggiunta ordini

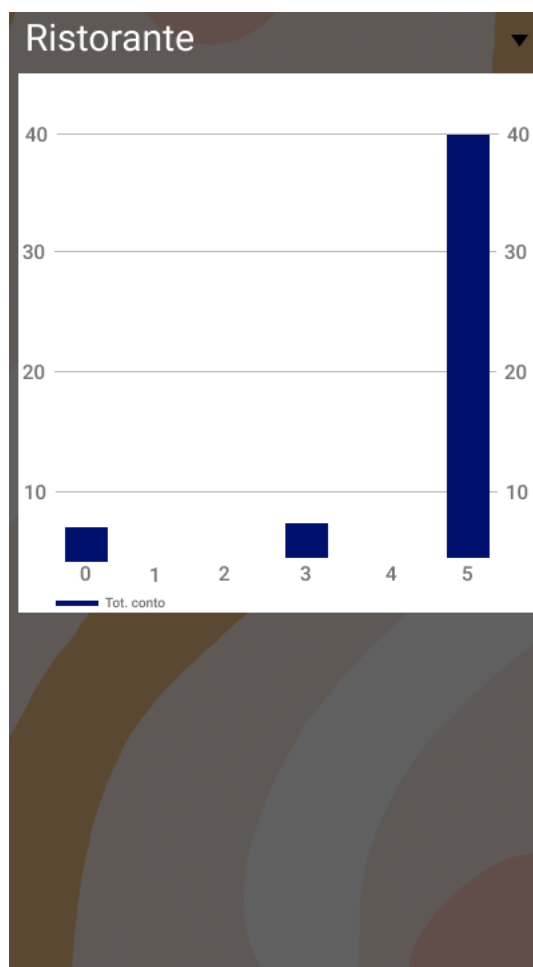
ID **M19**

Componenti:

Tipo	Nome	Funzione
ScrollView	PIATTI	Visualizza la lista dei piatti del menù del ristorante, dove ogni piatto si può cliccare per aggiungerlo all'ordine
Bottone	SALVA	Quando cliccato porta alla schermata M11 inviando l'ordine
Bottone	INDIETRO	Quando cliccato porta alla schermata M11 senza effettuare nessuna azione



2.6.20 Schermata di visualizzazione delle statistiche



M20: Schermata visualizzazione statistiche

ID *M20*

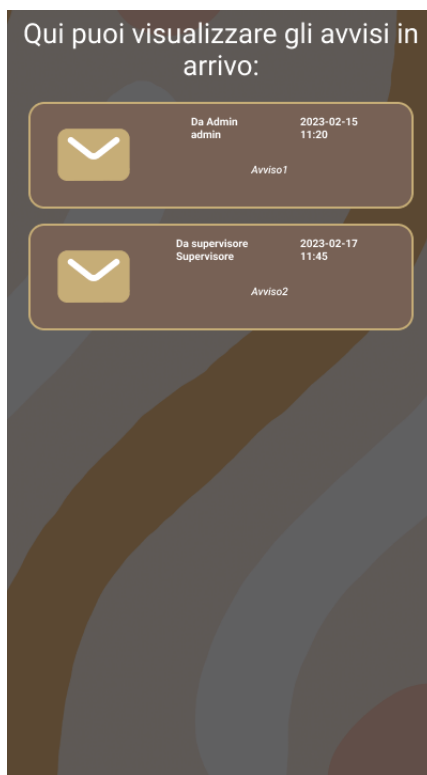
Componenti:

Tabella 8:

Tipo	Nome	Funzione
Spinner	RISTORANTE	Permette di selezionare il ristorante di cui si vogliono visualizzare le statistiche
ScrollView	GRAFICI STATISTICHE	Permette di visualizzare i grafici relativi alle statistiche di <i>Ristorante</i>



2.6.21 Schermata di visualizzazione delle notifiche



M21: Schermata visualizzazione notifiche

ID **M21**

Componenti:

Tabella 9:

Tipo	Nome	Funzione
ScrollView	NOTIFICHE	Permette di visualizzare le varie notifiche ricevute dal proprio ristorante, inviate dall'admin o dal supervisore, contenenti: <i>Nome, Data e ora, Messaggio</i> , che tramite swipe verso sinistra vengono segnati come letti ed eliminati dalla schermata



2.7 Valutazione dell'usabilità

Per la valutazione dell'usabilità del nostro applicativo a priori, cioè prima della fase di sviluppo vera e propria, abbiamo deciso di imporci come linee guida le euristiche di Nielsen. Ne sono 10, ma vorremmo richiamare l'attenzione su alcune di esse nello specifico:

- *Visibilità dello stato del sistema.* Il sistema presentava una discreta mancanza di feedback, che prevediamo di colmare con elementi quali Dialog, Toast, AlertDialog e SnackBar.
- *Prevenzione degli errori.* Il sistema reagisce in maniera controllata e predeterminata alle situazioni di errori che gli utenti possono causare. Nulla è lasciato al caso, ed è, nelle build provate dal team, gestita qualsiasi azione eseguibile dagli utenti.
- *Riconoscere piuttosto che ricordare.* La nostra app è dotata di sezioni ben distinte, interfacce dinamiche a seconda del tipo di utente che le usa, e icone e testi esplicativi dell'azione che si va a intraprendere.
- *Guida e documentazione.* E' presente un simpatico topolino (che richiama il logo dell'app) che consiglia tramite vignette e piccoli dialoghi le azioni che si possono intraprendere. Inoltre ci sono piccole note come campi obbligatori ecc.

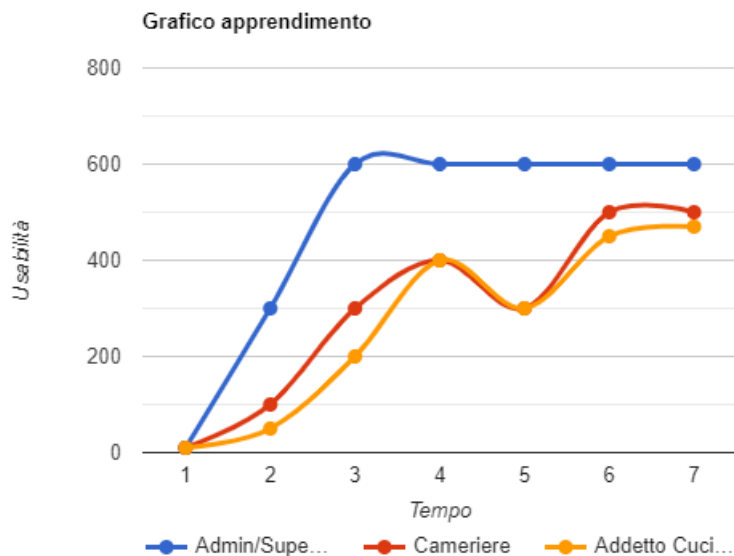


Figura 2.17: **Grafico:** Usabilità



2.8 Individuazione target d'utenti

La conoscenza dell'utente finale è di importanza fondamentale per chi progetta sistemi software di questo tipo. La grande diversità degli esseri umani fa sì che, anche considerando compiti e contesti d'uso simili, un oggetto potrebbe risultare usabile per un certo utente e del tutto inusabile per un altro.

Sicuramente, in base alle richieste del committente abbiamo subito individuato 4 principali categorie di utenti utilizzatori dell' app:

Admin: Amministratore e proprietario del ristorante. Una persona che deve avere tutto sotto controllo, può gestire le sue attività nonché i dipendenti che ne fanno parte, aggiungerne di nuova e talvolta, purtroppo, eliminarli.

Supervisore: Dopo l'amministratore, è nella "gerarchia" da noi definita, la seconda persona con più funzioni disponibili in-app. Anch'esso dispone di una dashboard completa sullo stile dell'admin.

Cameriere/Addetto sala: Senza la figura del cameriere un'attività di ristorazione non va avanti. Sappiamo quant'è importante fornire a questi ultimi un applicativo funzionale, facile da usare e da apprendere: per questo la sua interfaccia è ottimizzata per un palmare o smartphone compatto.

Addetto Cucina: Riceve tutti gli ordini dai camerieri e li inoltra alla cucina. Anche lui dispone di un'interfaccia semplice e dinamica che perfettamente si adatta al suo ruolo nell'attività.

Tutto ciò è reso possibile da uno sviluppo che va incontro alle esigenze dei diversi utenti. La nostra interfaccia riconosce il tipo di utente che logga e cambia in base alle sue esigenze.

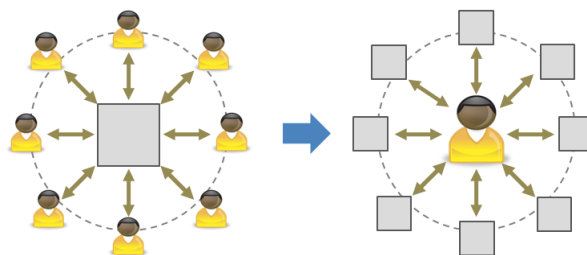


Figura: Da una visione centrata sul sistema a una visione centrata sull'utente



2.9 Glossario

In questa sezione vengono chiariti alcuni termini usati all'interno della documentazione, per rendere la lettura accessibile anche ai non esperti del settore.

- *Dialog*: Tipo di popup che il sistema Android mette a disposizione che mostra una finestra di dialog personalizzabile. Nella nostra app, mostra "Ok" e "Torna indietro".
- *Spring Boot*: Spring è uno strumento che permette a noi programmatori di usare il linguaggio di programmazione ad oggetti java per scrivere ottime app lato server.
- *JPA*: Le Java Persistence API, talvolta riferite come JPA, sono un framework per il linguaggio di programmazione Java che si occupa della gestione della persistenza dei dati di un DBMS relazionale
- *MVC*: Pattern Model-View-Controller, usato da Android Studio e da Spring Boot.



2.10 Class diagram di analisi o dominio

Class diagram a livello concettuale Studia i concetti propri del dominio sotto studio, senza preoccuparsi della loro successiva implementazione. È una sorta di Dizionario Visuale del dominio.

Class diagram di specifica implementativa Rappresenta la struttura implementativa, specificando come va sviluppato il sistema. È un raffinamento del precedente.

I Class diagram di analisi o di dominio ci forniscono un'indicazione iniziale sulle componenti principali del software a prescindere dall'architettura e dalle tecnologie che verranno poi scelte per implementarli. In questa fase infatti ancora di analisi, il team sta concretizzando i requisiti in qualcosa di pratico da dare in pasto ai programmatori, che ne seguiranno il modello per arrivare al prodotto finito.

2.10.1 Class diagram Login

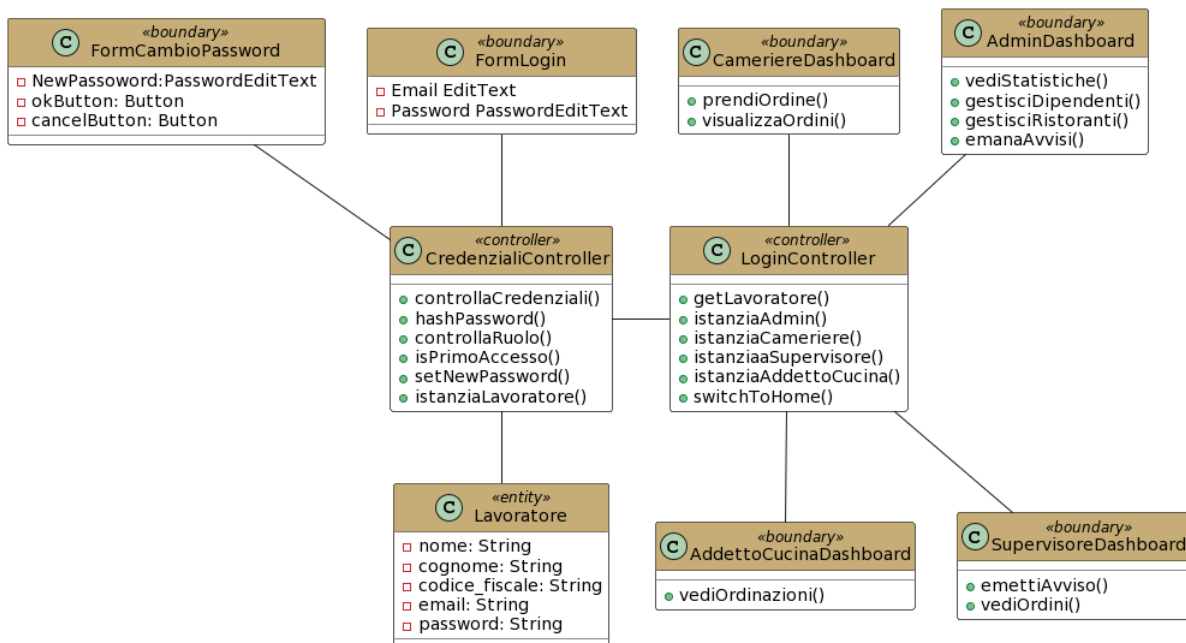


Figura 2.18: C01: Class diagram Login



2.10.2 Class diagram Ristorante

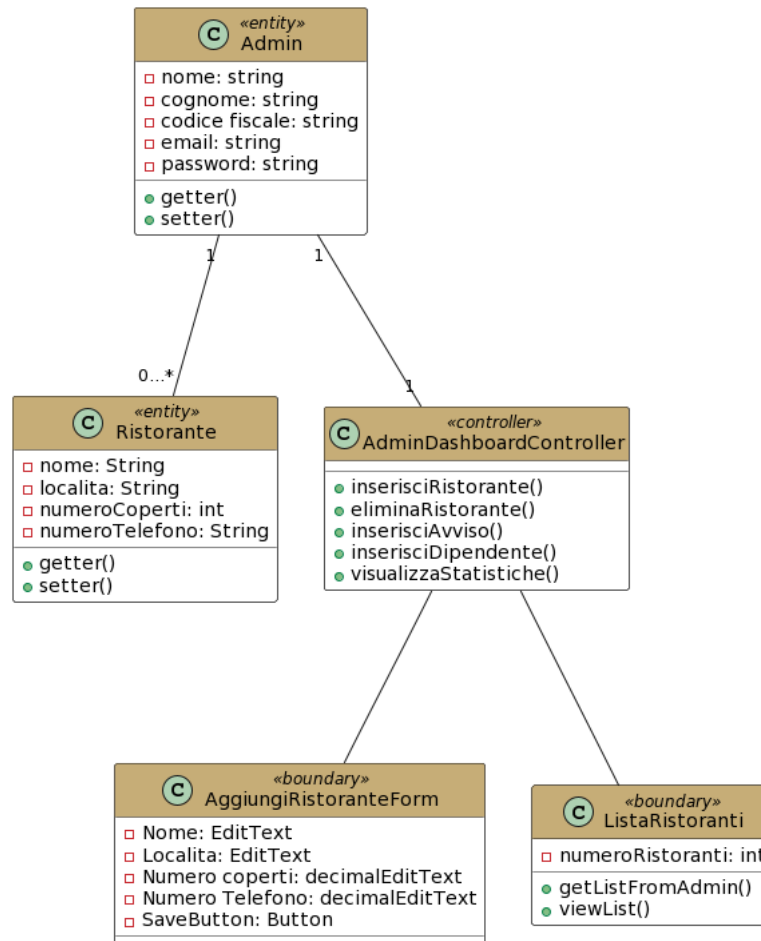


Figura 2.19: **C02**: Class diagram Gestione ristorante



2.10.3 Class diagram dipendenti

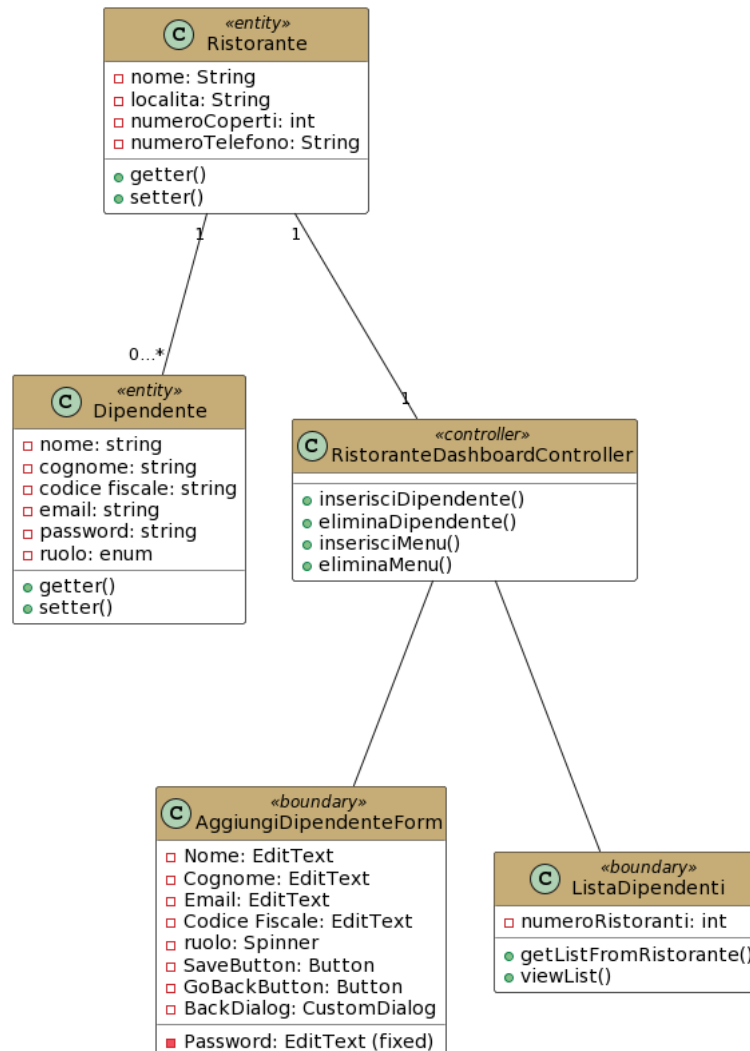


Figura 2.20: C03: Class diagram gestione dipendenti



2.10.4 Class diagram Menu

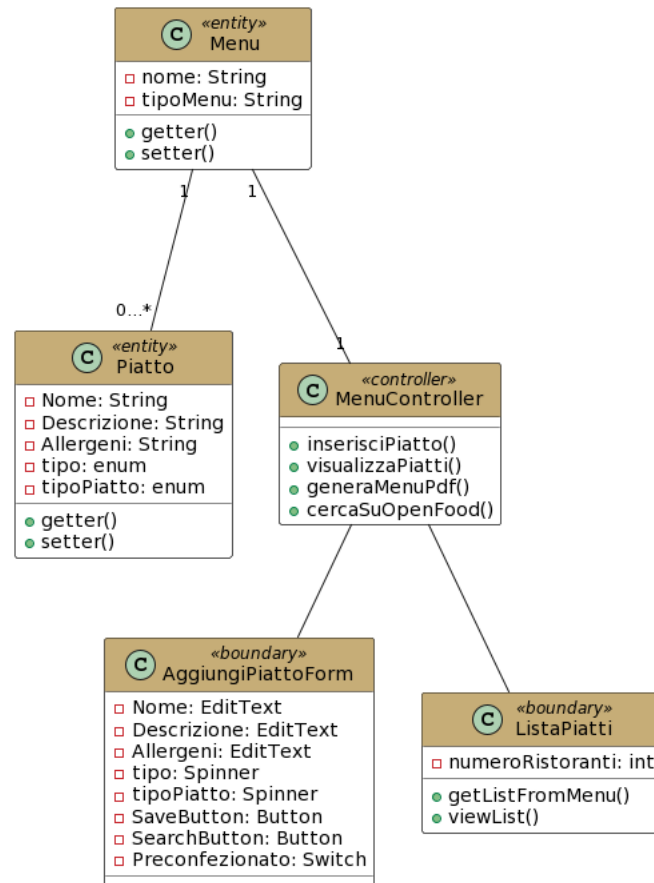


Figura 2.21: C04: Class diagram gestione piatti



2.10.5 Class diagram Avvisi

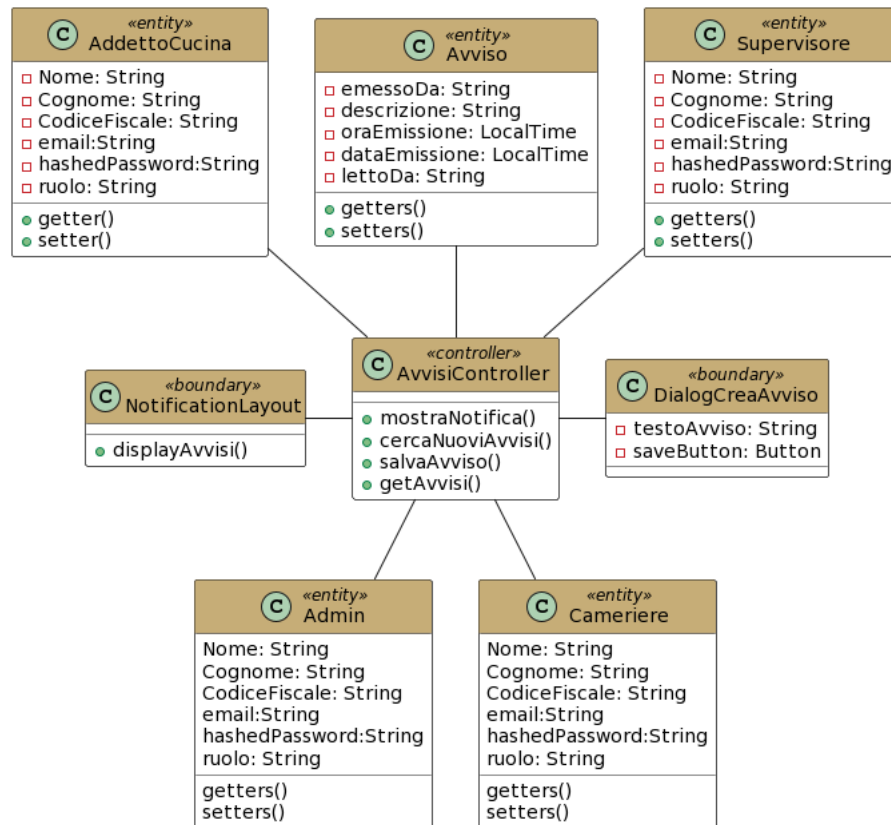


Figura 2.22: C05: Class diagram gestione avvisi



2.10.6 Class diagram Ordini

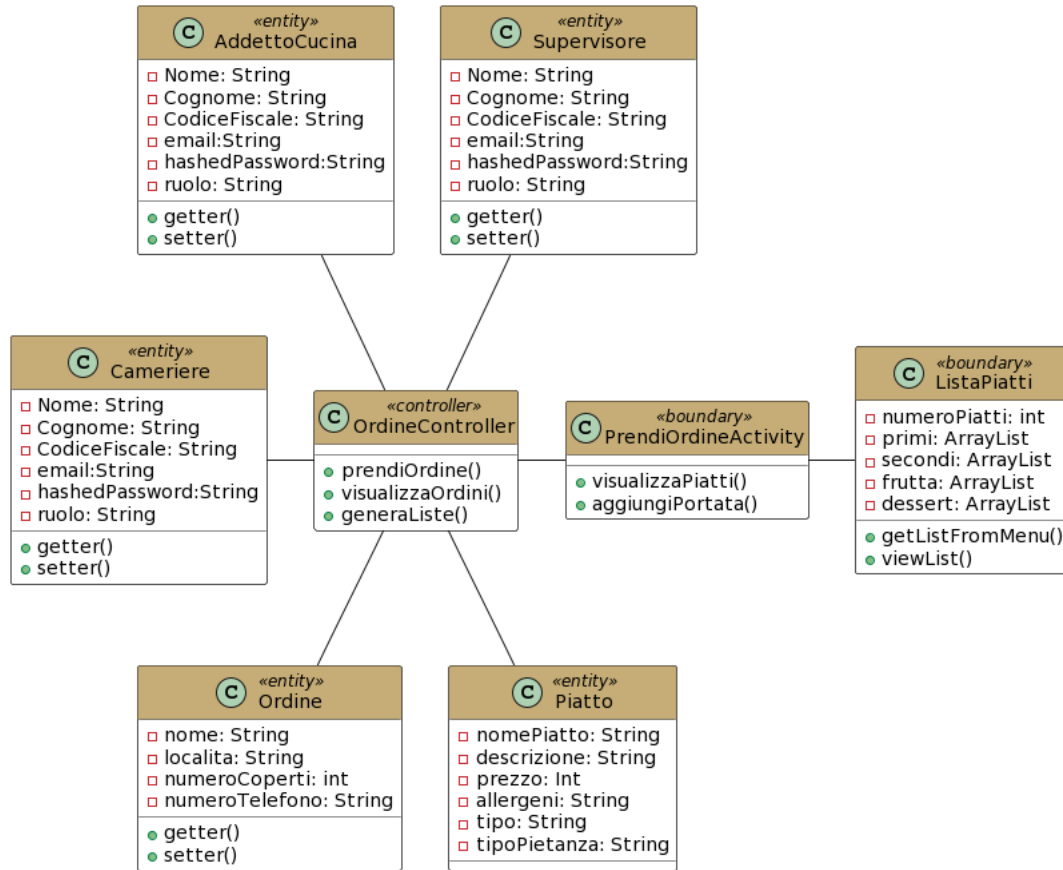


Figura 2.23: C06: Class diagram gestione ordini

2.10.7 Class diagram statistiche

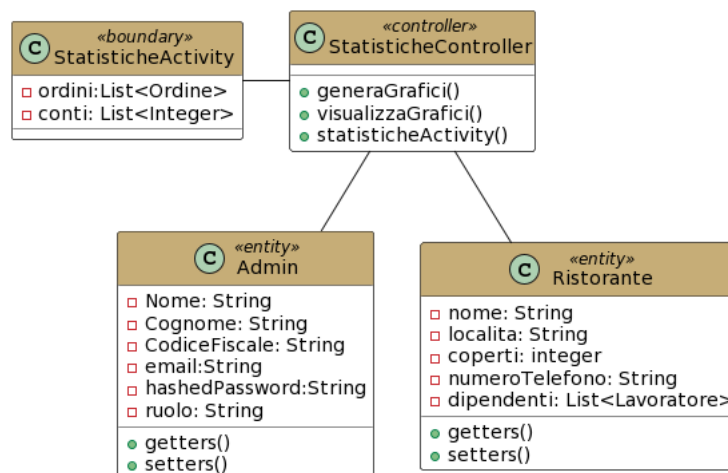


Figura 2.24: C07: Class diagram gestione statistiche



2.11 Sequence di analisi

Abbiamo scelto di modellare (in fase di analisi dei requisiti) con i sequence diagram i seguenti casi d'uso:

- *Aggiunta menu del ristorante*
- *Prendere ordinazione*

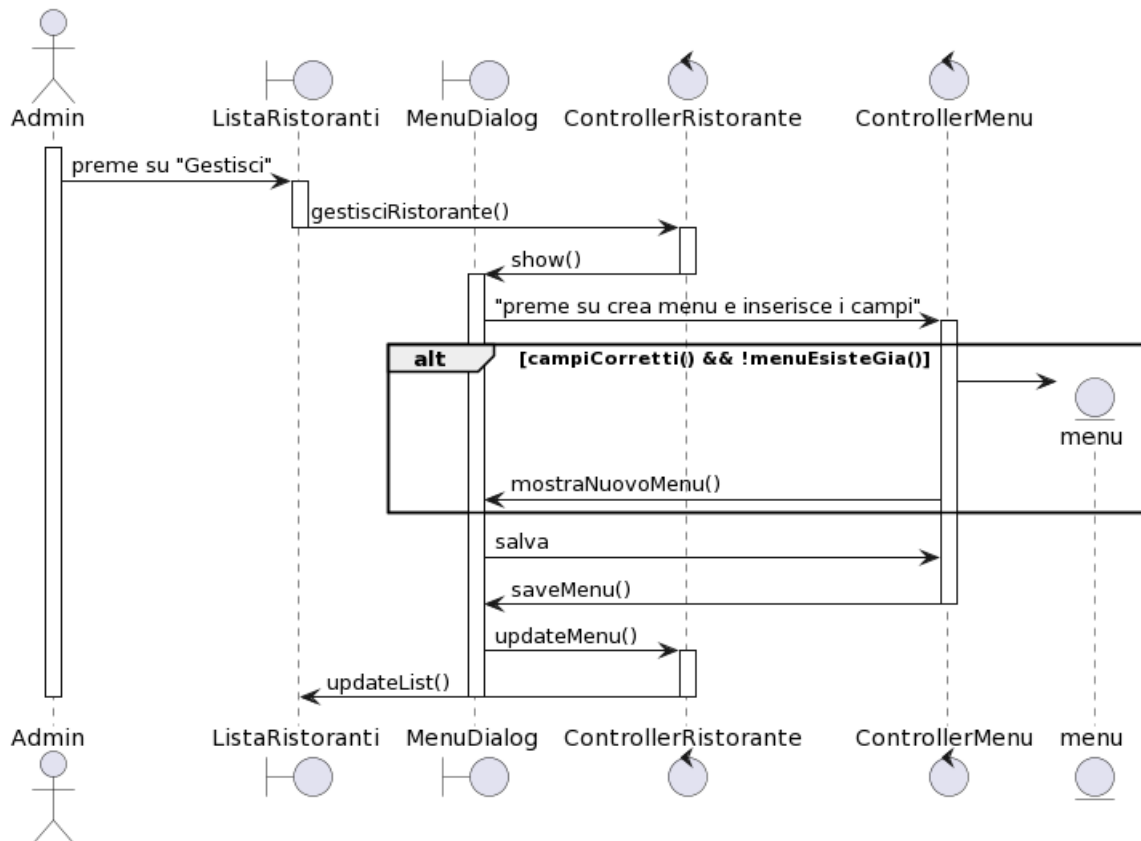


Figura 2.25: **Sequence:** Aggiungi menu

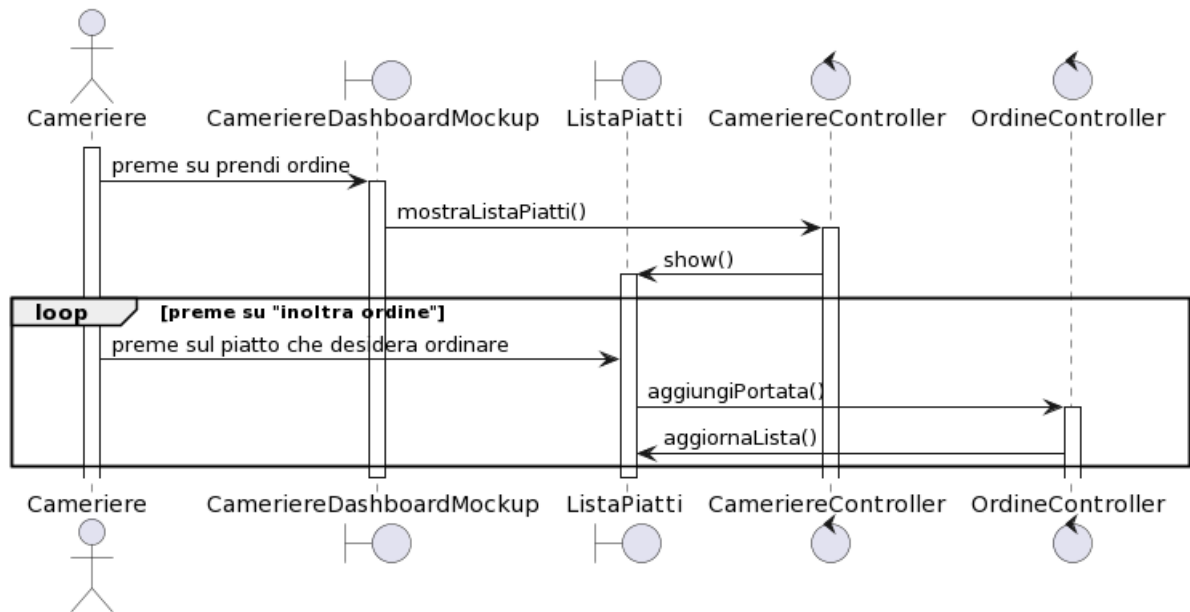


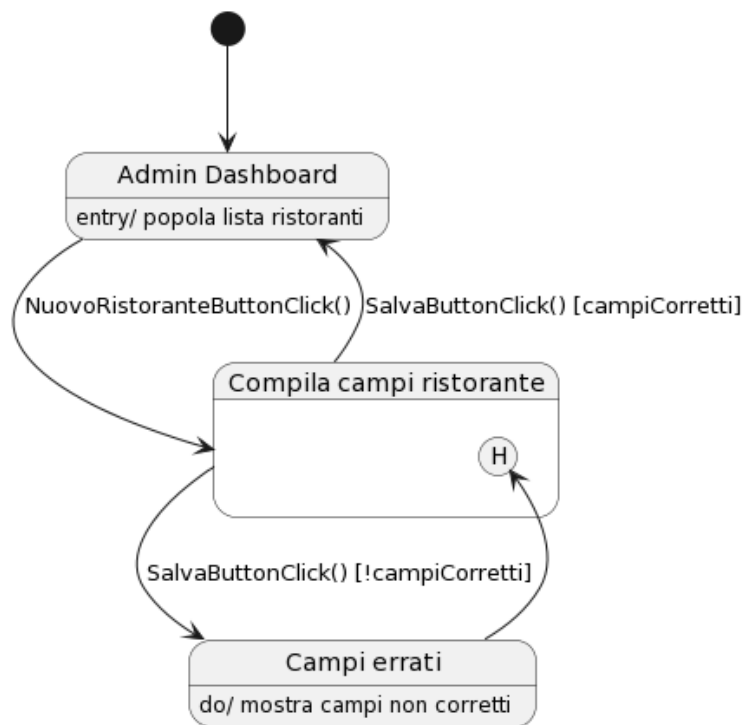
Figura 2.26: **Sequence:** Prendi ordine



2.12 Statechart

Gli statechart sono uno strumento di modellazione grafica, utile per rappresentare il funzionamento di un sistema. Sono caratterizzati da stati e transizioni, che rappresentano il passaggio tra stati diversi. Sono inoltre molto importanti poichè forniscono un chiaro mezzo di costruzione di prototipi statici: con pochi semplici simboli è possibile modellare l'intero funzionamento di un'interfaccia grafica. Useremo questo potente strumento per rappresentare i seguenti casi d'uso: **Aggiungi ristorante**, **Aggiungi piatto**, **Prendi ordine** e **Visualizza avvisi**.

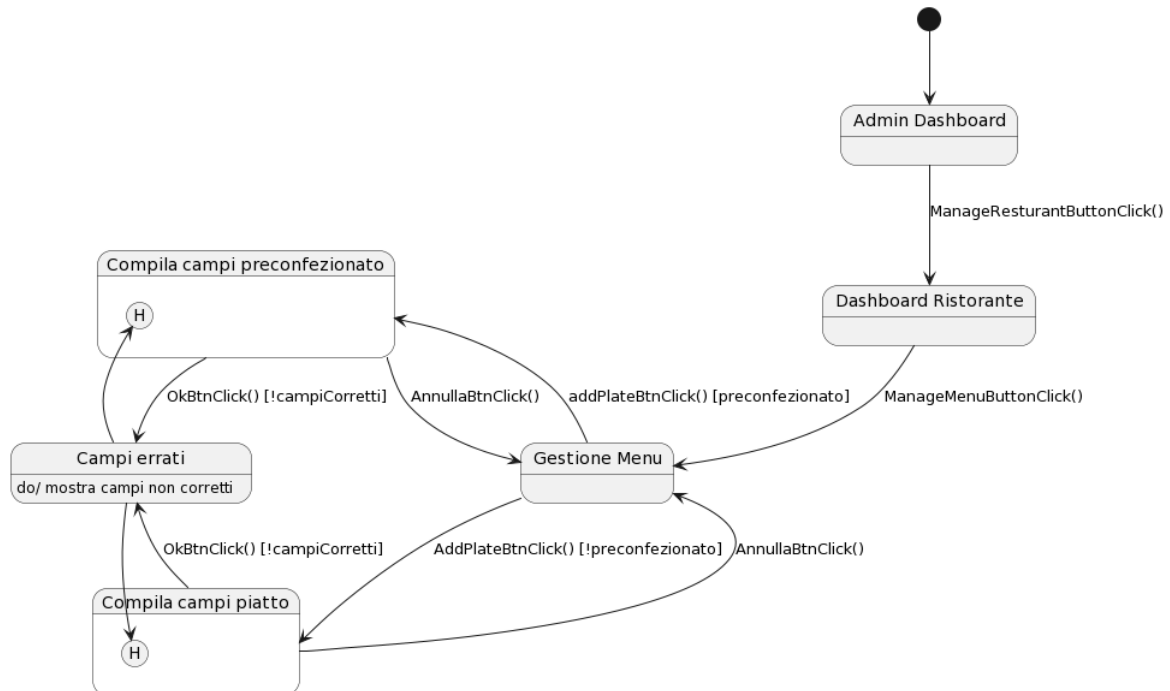
2.12.1 Aggiungi ristorante



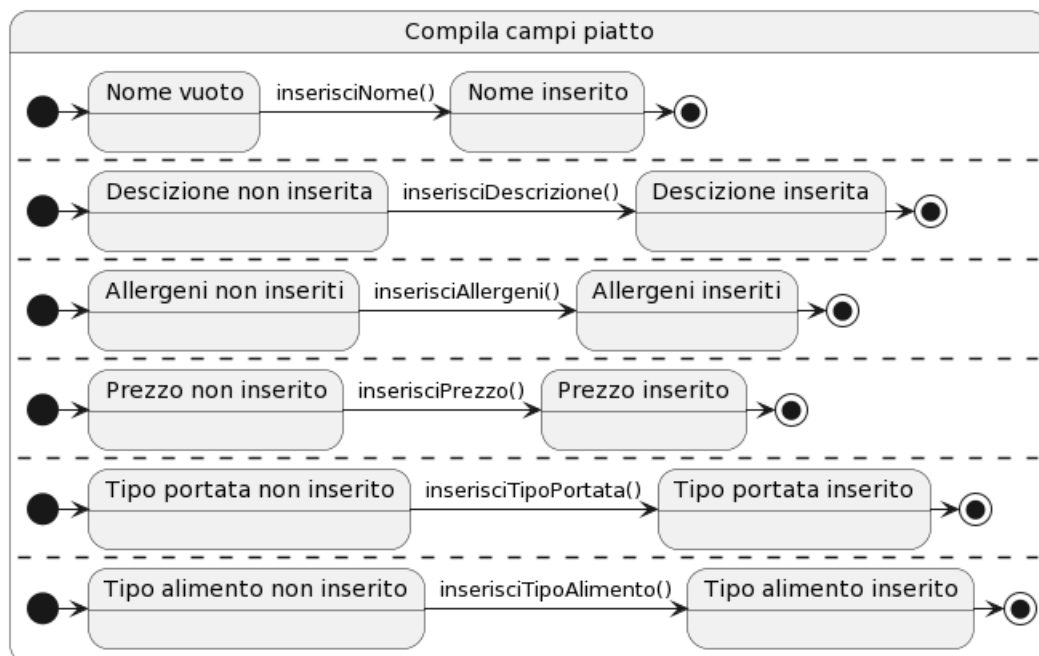
Statechart : aggiunta ristorante



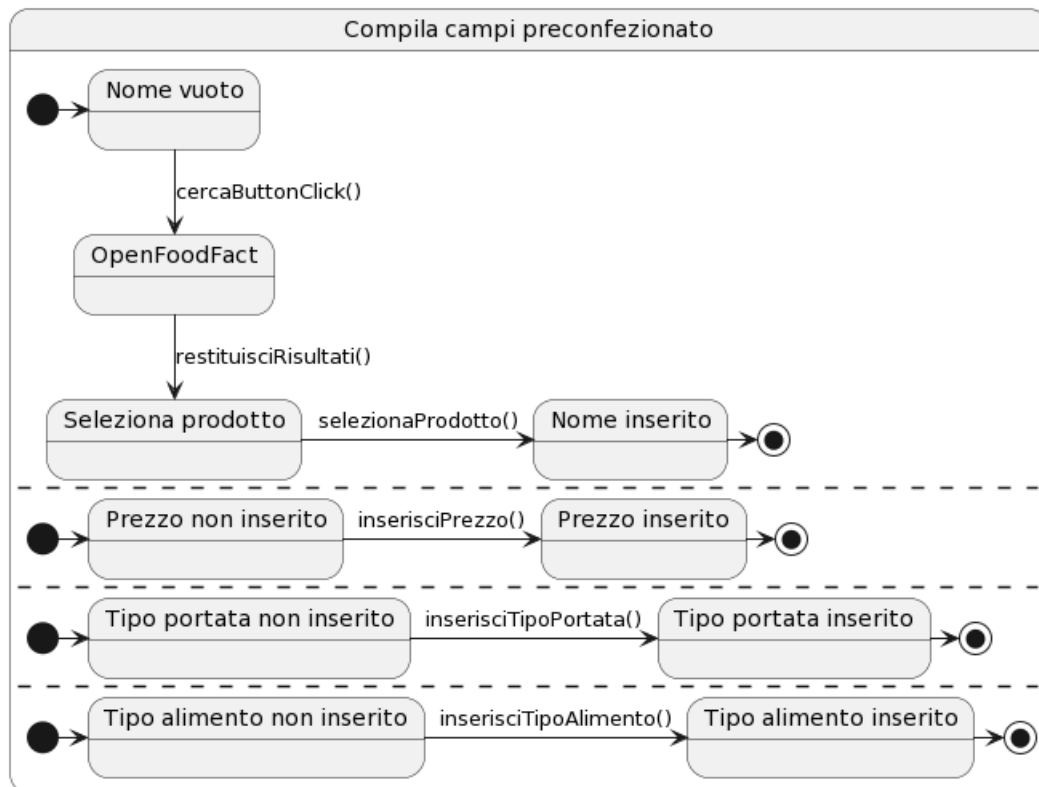
2.12.2 Aggiungi piatto



Statechart : aggiunta piatto

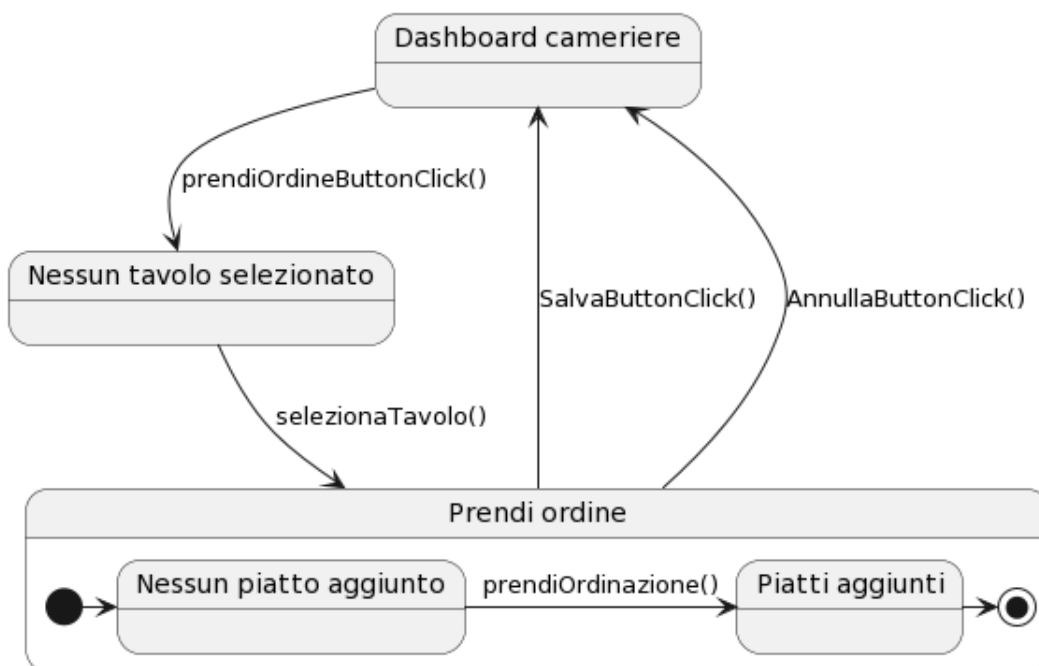


Statechart : compilazione campi piatto non preconfezionato



Statechart : compilazione campi piatto preconfezionato

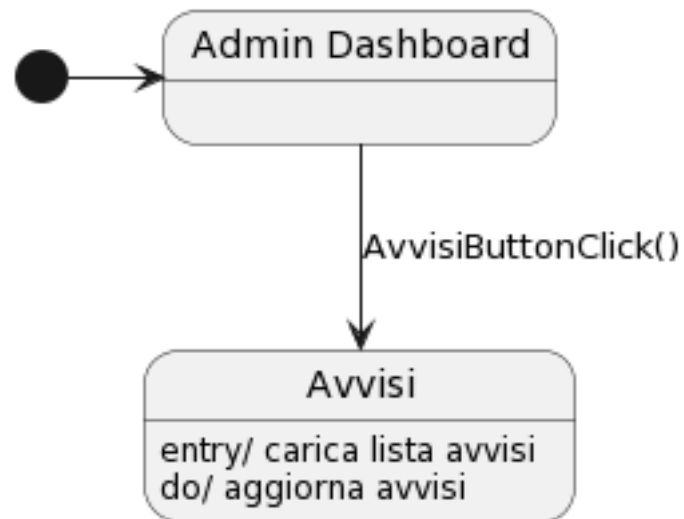
2.12.3 Prendi ordine



Statechart : Prendi ordinazione



2.12.4 Visualizza avvisi



Statechart : visualizzazione avvisi



3 Documento di design

3.1 Architettura del sistema

3.2 Introduzione

Alla luce delle valutazioni effettuate durante l'analisi dei requisiti, abbiamo trovato conveniente strutturare la nostra app seguendo un tipo di architettura client-server. I client, nel nostro caso, sono quasi sempre dei dispositivi portatili dotati di sistema operativo android (con la possibilità volendo di utilizzare emulatori da pc desktop) mentre il server è stato realizzato con il framework di sviluppo Spring Boot.

3.2.1 Client

Come già detto, il client è composto da un'applicazione android distribuita o tramite apk o tramite play store (rispetta tutti i requisiti per essere pubblicata) che mira ad offrire un'interfaccia utente semplice e pulita per l'accesso agli endpoint forniti dal nostro server Spring-Boot. Come ogni app progetto nativo Android, il nostro client adotta un modello di design MVC (Model-View-Controller) e in contemporanea ad un pattern Singleton che tiene traccia dell'utente loggato e si tiene aggiornato quasi in tempo reale con i dati forniti dal server.

3.2.2 Server

Il back-end della nostra applicazione è stato realizzato con la tecnologia offerta da Spring Boot, un potente framework di sviluppo che sfrutta Java. L'architettura che prevede è anch'essa a 3 livelli e prevede i seguenti componenti:

- **Model:** L'insieme di classi che rappresentano le entità della nostra applicazione. Le stesse sono riportate sul client, e, cosa più importante, spring mappa le classi con l'annotation "@entity" 1:1 con le tabelle nel db relazionale.
- **Repository:** Sfruttando il framework JPA (Java Persistence API) riusciamo a gestire persistenza e consistenza dei dati nel db in maniera quasi automatizzata. Spring ci fornisce un gran supporto in questo ambito.
- **Controller:** Qui ci va tutta la logica di controllo del server. Sono queste le classi che espongono gli end-point Rest ai client e, comunicando con le repository, aggiornano i model e le tabelle nel db allo stesso tempo.

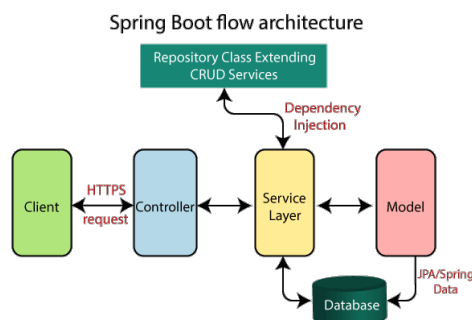


Figura: Spring-Boot



Microsoft Azure

Microsoft Azure è la piattaforma cloud pubblica di Microsoft, che offre servizi di cloud computing. Tra i vari piani che mette a disposizione, abbiamo usufruito di quello "gratuito" che ci permette di eseguire una macchina virtuale linux e un database relazionale, che nel nostro caso è stato PostgreSQL. La configurazione della macchina virtuale non ha richiesto particolari abilità: tramite connessione ssh abbiamo configurato il progetto Spring Boot e avviandolo mettiamo a disposizione gli end-point raggiungibili poi dai client. La potenza computazionale in questa fase di sviluppo e di utilizzo limitato dell'applicativo potrebbe risultare esile, ma il vantaggio di piattaforme quali Azure è possibile in qualsiasi momento aumentare le risorse dedicate a db e/o macchina virtuale, aumentando di fatto la scalabilità della nostra applicazione.

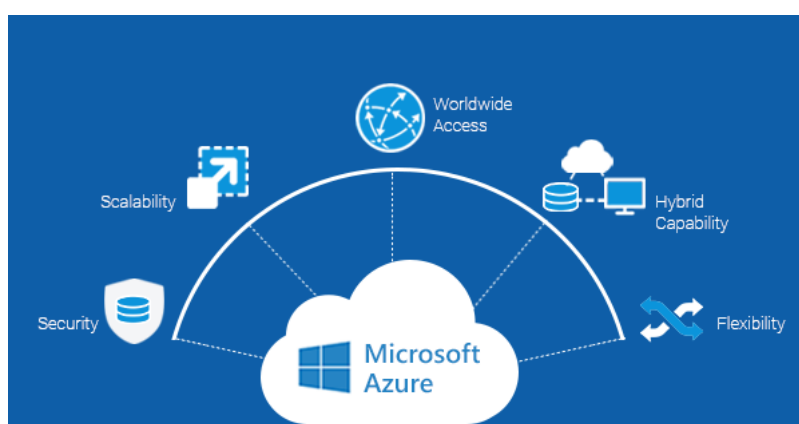


Figura: Microsoft Azure



3.3 Sequence diagram di design

3.3.1 Funzionalità: "Aggiungi Ristorante"

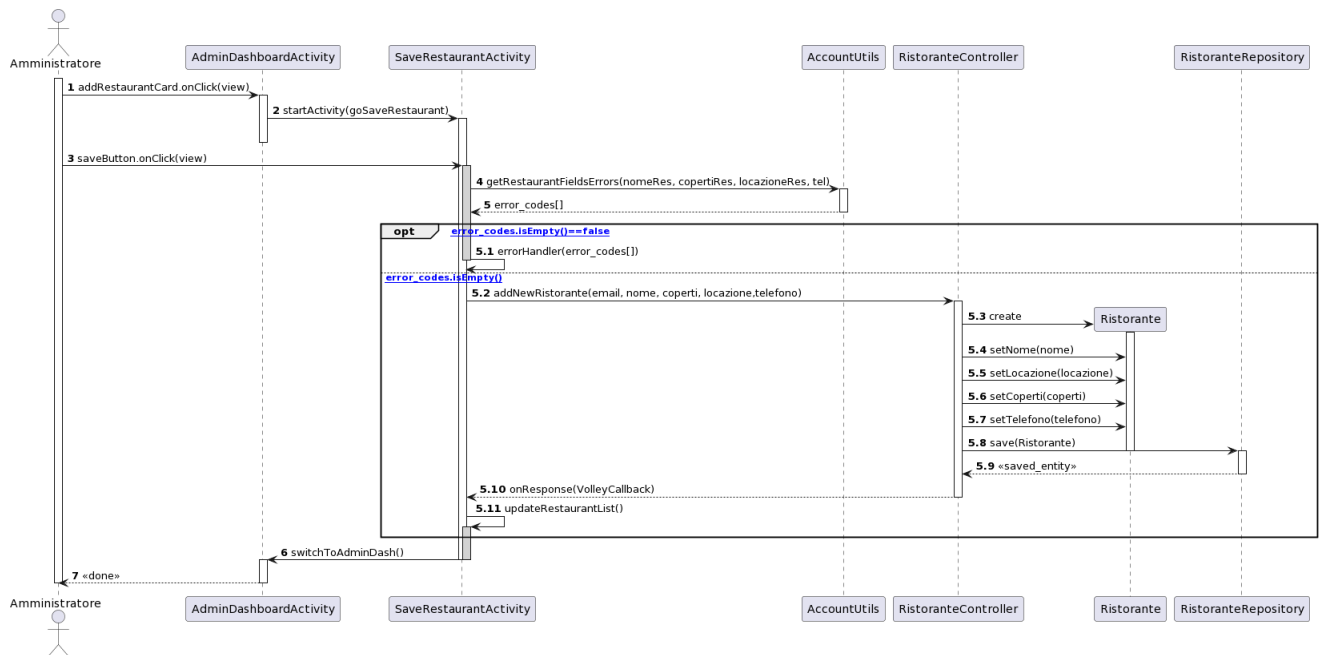


Figura 3.1: Sequence: Salva ristorante

3.3.2 Funzionalità: "Aggiungi portata"

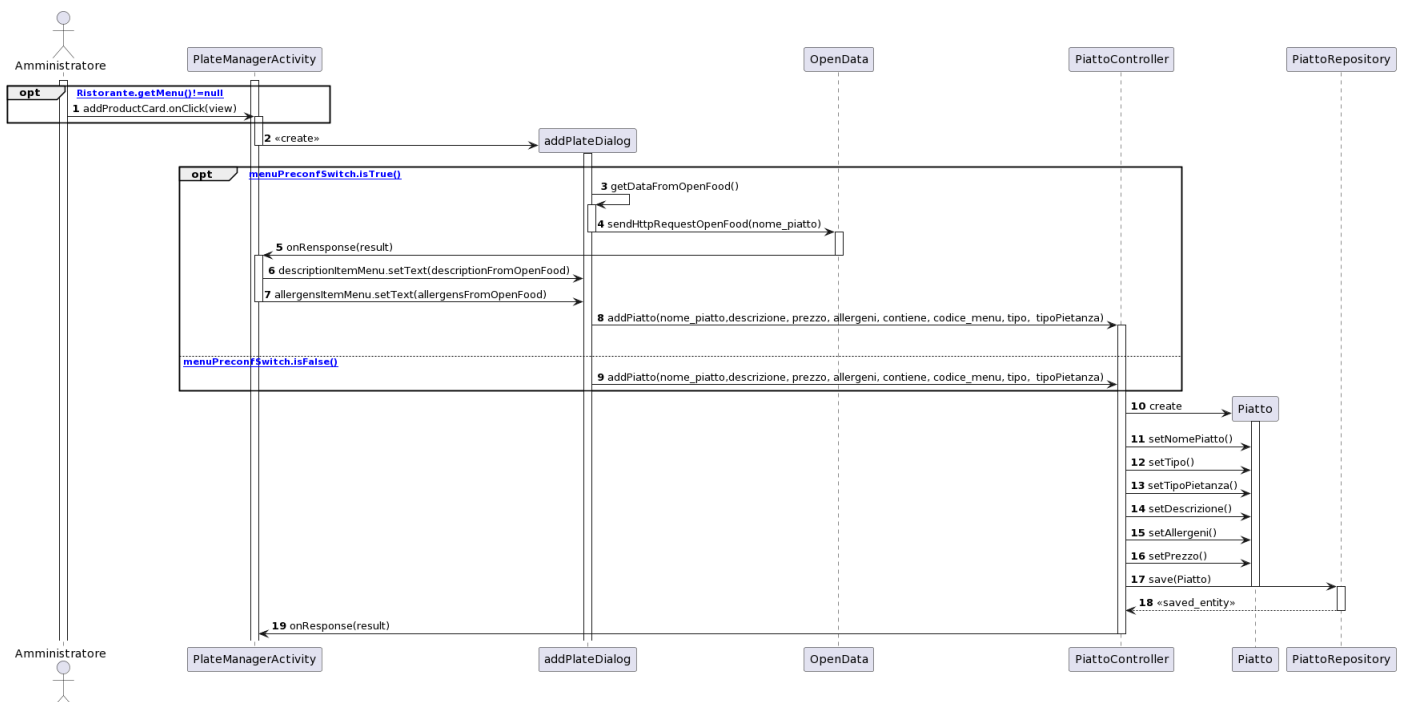


Figura 3.2: Sequence: Aggiunti portata



3.3.3 Funzionalità: "Prendi ordinazione"

3.3.4 Funzionalità: "Visualizza avvisi"

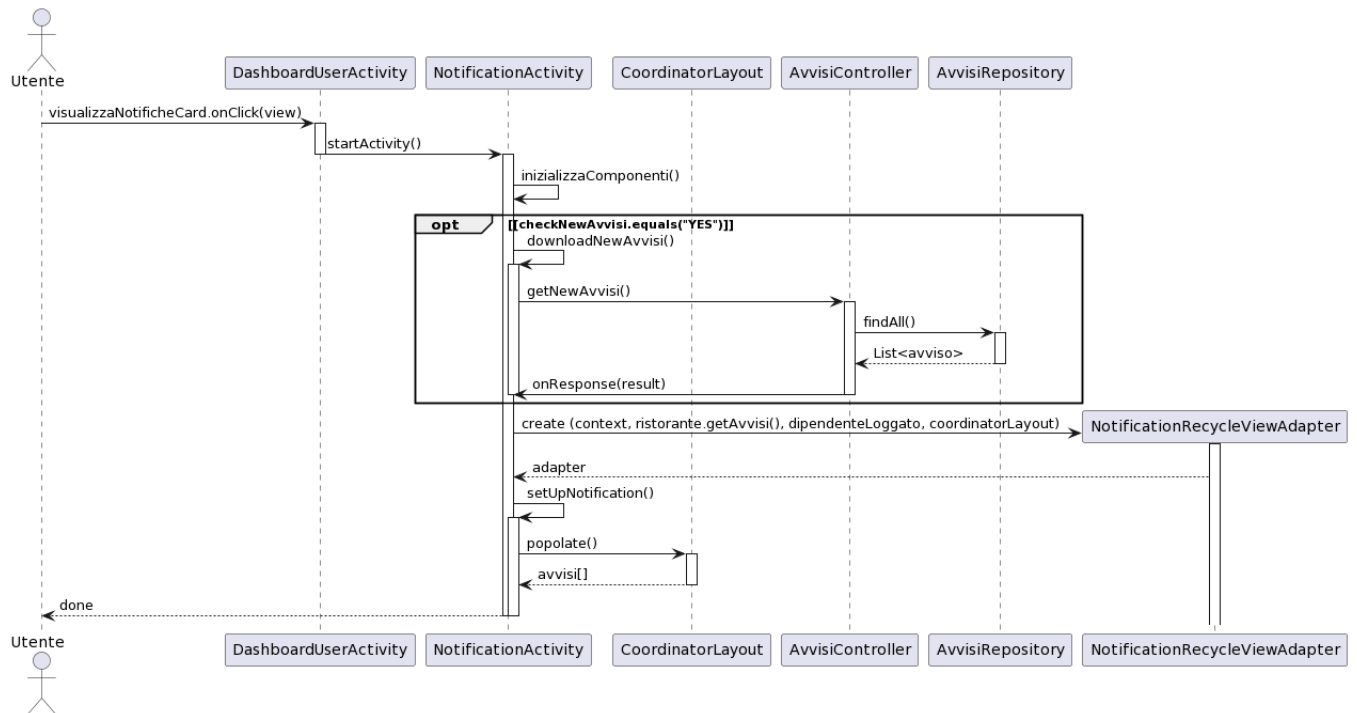


Figura 3.3: Sequence: Visualizza avvisi

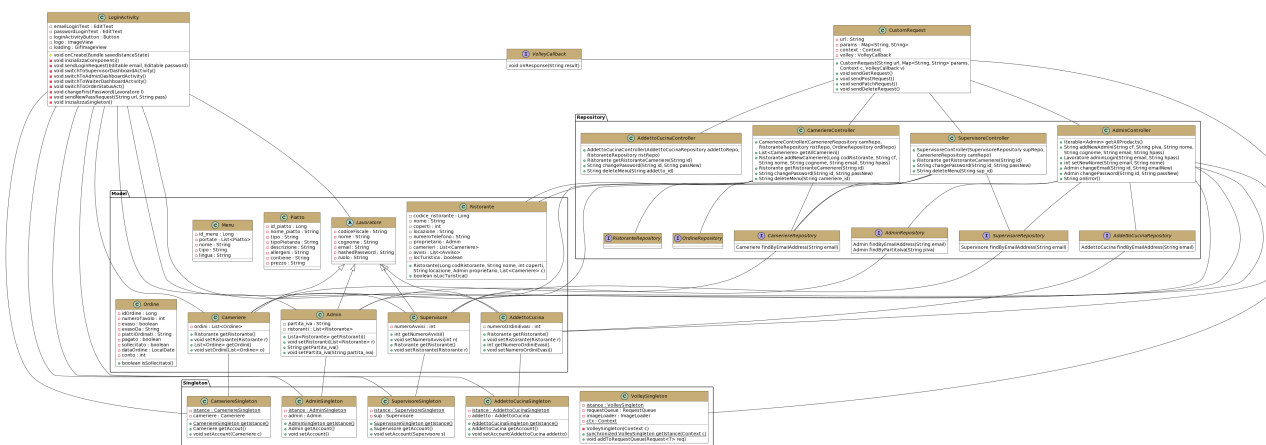


3.4 Class diagram di design

Verranno di seguito riportati i class diagram di design, che ora rispecchiano l'architettura del sistema e tutte le funzionalità nella loro interezza.

Nota: Tutti i costruttori banali (senza argomenti) e i getter e setter sono stati omessi per rendere più leggibili i diagrammi

3.4.1 Login



CD01: Class diagram Login



3.4.3 Gestione Avvisi

