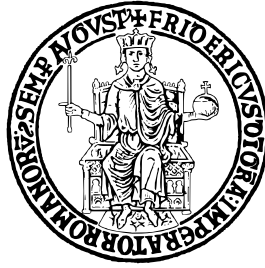


UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II



SCUOLA POLITECNICA E DELLE SCIENZE DI BASE

DIPARTIMENTO DI INGEGNERIA ELETTRICA E TECNOLOGIE DELL'INFORMAZIONE

CORSO DI LAUREA MAGISTRALE IN INFORMATICA

INGEGNERIA DEL SOFTWARE

RATATOUILLE23

Professori

S. D. MARTINO

F. CUTUGNO

L. L. L. STARACE

M. GRAZIOSO

Candidato

Paolo CAMMARDELLA

N86/3043

Anno Accademico 2022–2023

Indice

0	Glossario	2
0.1	Definizioni	2
0.2	Acronimi	4
1	Introduzione	5
1.1	Cosa contiene questa documentazione	5
1.2	Tecnologie utilizzate	5
2	Analisi dei requisiti	6
2.1	Requisiti del software	6
2.1.1	Requisiti funzionali	6
2.2	Requisiti non funzionali	6
2.3	Requisiti di dominio	7
2.4	Modellazione dei casi d'uso	7
2.4.1	Use-case generale	7
2.4.2	Creazione piatto	8
2.4.3	Eliminazione piatto	8
2.4.4	Creazione ordinazione	9
2.4.5	Creazione notifica	10
2.5	Tabelle cockburn	11
2.5.1	Creazione ordinazione	11
2.5.2	Eliminazione piatto	12
2.5.3	Creazione piatto	13
2.5.4	Creazione notifica	14
2.6	Mock-Up	15
2.6.1	Creazione ordinazione	16
2.6.2	Eliminazione piatto	19
2.6.3	Creazione piatto	21
2.6.4	Creazione notifica	23
2.7	Individuazione del target a priori	25
2.7.1	Personas	25
2.8	Valutazione dell'usabilità a priori	29
2.8.1	Tecnica utilizzata	31
3	Specifica dei casi d'uso	32
3.1	Classi, oggetti e relazioni di analisi	32
3.2	Class diagram di analisi - Gestione menù	33
3.3	Class diagram - Creazione utenza	34
3.3.1	Class diagram - Accesso alla piattaforma	34
3.4	Class diagram - Gestione tavoli	35
3.5	Class diagram - Gestione notifiche	36

0 Glossario

Qui sono racchiuse tutte le definizioni e gli acronimi che sono stati utilizzati nella documentazione.

0.1 Definizioni

Scalabilità si riferisce alla capacità di un sistema di aumentare o diminuire le risorse a seconda delle necessità.

Usabilità l'usabilità di un prodotto è il grado con cui esso può essere usato da specificati utenti per raggiungere specificati obiettivi con efficacia, efficienza e soddisfazione in uno specificato contesto d'uso.

Mock-up Un mock-up, è una realizzazione a scopo illustrativo o meramente espositivo di un oggetto o un sistema, senza le complete funzioni dell'originale. Un mockup può rappresentare la totalità o solo una parte dell'originale di riferimento (già esistente o in fase di progetto), essere in scala reale oppure variata.

Class diagram I class diagram (diagrammi delle classi, in italiano) sono uno dei tipi di diagrammi che possono comparire in un modello UML.

In termini generali, consentono di descrivere tipi di entità, con le loro caratteristiche e le eventuali relazioni fra questi tipi. Gli strumenti concettuali utilizzati sono il concetto di classe del paradigma object-oriented e altri correlati.

Sequenc diagram Un Sequence Diagram (Diagramma di sequenza, in italiano) è un diagramma previsto dall'UML utilizzato per descrivere uno scenario.

Scenario Uno scenario è una determinata sequenza di azioni in cui tutte le scelte sono state già effettuate; in pratica nel diagramma non compaiono scelte, né flussi alternativi.

State chart diagram Uno state chart (diagramma di stato, in italiano) è un tipo di diagramma usato in informatica per descrivere il comportamento dei sistemi, il quale viene analizzato e rappresentato tramite una serie di eventi che potrebbero accadere per ciascuno stato. Per poter essere realizzato, il sistema deve essere composto da un numero finito di stati.

Activity diagram Un activity diagram (diagramma di attività in italiano) è un tipo di diagramma che permette di descrivere un processo attraverso dei grafi in cui i nodi rappresentano le attività e gli archi l'ordine con cui vengono eseguite.

Persona Una persona in UX design è un identikit vero e proprio di un utente (fittizio) ideale, che viene utilizzato per identificare una fascia d'utenti, i quali esprimono le loro esigenze, comportamenti ed interessi.

White-Box Modalità di testing in cui un metodo viene testato in base al codice che contiene.

Black-Box Modalità di testing in cui un'unità viene testata in base ai requisiti.

Framework È un'architettura logica di supporto sul quale un software può essere progettato e realizzato.

Back-end Rappresenta la parte che permette l'effettivo funzionamento di queste interazioni.

Front-end Rappresenta la parte che interagisce con l'utente finale.

Android Sistema operativo realizzato da Google.

Server Programma che ha il compito di offrire servizi ai client richiedenti.

Tabella Cockburn Tabelle utilizzate per la rappresentazione dei casi d'uso.

Use case Rappresenta una funzione o servizio offerto dal sistema ad uno o più attori.

Docker È un popolare software libero progettato per eseguire processi informatici in ambienti isolabili, minimali e facilmente distribuibili chiamati container Linux, con l'obiettivo di semplificare i processi di deployment di applicazioni software.

0.2 Acronimi

AWS Amazon Web Services è un'azienda statunitense di proprietà del gruppo Amazon, che fornisce servizi di cloud computing su un'omonima piattaforma *on demand*.

EC2 Uno dei servizi di cloud offerti da AWS.

API Un API (Application Program Interface), in un programma informatico, in italiano "interfaccia di programmazione dell'applicazione", si indica un insieme di procedure atte a risolvere uno specifico problema di comunicazione tra diversi computer, tra diversi software, tra diversi componenti di software. Spesso tale termine designa le librerie software di un linguaggio di programmazione.

HTTP Sta per HyperText Transfer Protocol, è un protocollo a livello applicativo usato come principale sistema per la trasmissione di informazioni via web.

ISO/IEC ISO e IEC cooperano strettamente per creare uno standard internazionale sviluppato specificatamente per la gestione dei servizi IT.

GDPR General Data Protection Regulation, è un regolamento, sviluppato dalla comunità europea, che disciplina il trattamento dei dati personali da parte di aziende.

MVP Model View Presenter è un design pattern, principalmente utilizzato per software Android.

UI User Interface è il termine inglese usato per indicare l'interfaccia grafica.

DTO Data Transfer Object è un design pattern per trasferire dati da una parte back-end a quella front-end (ove necessario).

SECT È una specifica modalità di testing. Consiste nell'effettuare i test tramite prodotto cartesiano dei parametri.

WECT È una specifica modalità di testing. Consiste nell'effettuare il minimo numero di test case che ricoprono tutte le classi d'equivalenza valide e uno per quelle non valide.

1 Introduzione

Ratatouille23™ è un software sviluppato e progettato dal gruppo INGSW2223_N_46 per conto della società SoftEngUniNA™.

Gli sviluppatori si prendono carico della verifica dei moduli software necessari al corretto funzionamento di tale sistema. Ratatouille23™ è un software che offre il supporto alla gestione di attività di ristorazione.

Il sistema consiste in un'applicazione performante e affidabile, attraverso la quale gli utenti possono fruire delle funzionalità del sistema in modo intuitivo, rapido e piacevole ed una parte back-end che garantisce la comunicazione veloce ed affidabile tra i client.

1.1 Cosa contiene questa documentazione

Questa documentazione contiene tutte le informazioni necessarie a comprendere il funzionamento del software Ratatouille23™

- Documento dei Requisiti Software.
- Documento di Design del sistema.
- Definizione di un piano di testing e valutazione sul campo dell'usabilità.

1.2 Tecnologie utilizzate

Lo sviluppo del software Ratatouille23™ è avvenuto tramite:

- **Client:**
 - Android con linguaggio object oriented (Java).
 - Retrofit: richieste HTTP lato android.
- **Server:**
 - Framework Java Spring boot.
 - RestTemplate.
- **Database:** PostgreSQL.
- **Testing:** Suite JUnit.

Il tutto è affiancato tecnologie allo stato dell'arte quali *Docker* per separare in *container* back-end e database, e servizi di *Cloud Computing* come *AWS*, al fine di massimizzare la scalabilità del sistema in vista di un possibile repentino aumento del numero degli utenti nelle fasi iniziali di rilascio al pubblico.

2 Analisi dei requisiti

In questa sezione andremo a descrivere il modello funzionale del software partendo da un'analisi dei casi d'uso assegnati per l'applicativo, e proseguendo poi con tabelle cockburn e mock-up.

2.1 Requisiti del software

Lo sviluppo di Ratatouille23™ in parte è stato scandito dalla presenza di requisiti imposti dal cliente per il corretto funzionamento del software, in parte da requisiti tecnologici.

2.1.1 Requisiti funzionali

Il sistema offre le seguenti funzionalità:

1. La possibilità da parte di un amministratore di poter creare utenze per i propri dipendenti (e.g. addetti alla sala, addetti alla cucina, supervisori) con nome utente scelto dall'amministratore ed una password di default.
2. La possibilità da parte di un amministratore (o un supervisore) di personalizzare il menù dell'attività di ristorazione. In particolare, la possibilità di riordinare il menù, creare e/o eliminare elementi dal menù, caratterizzandoli tramite:
 - *nome*.
 - *descrizione*.
 - *elenco di allergeni comuni*.
 - *prezzo*.

In fase di creazione di un determinato piatto, è disponibile, utilizzando l'apposito tasto per la ricerca, l'autocompletamento di alcuni prodotti (e.g.: bibite o preconfezionati).

3. La possibilità da parte di un addetto alla sala di poter registrare ordinazioni indicando l'identificativo del tavolo e gli elementi del menù (già presenti) desiderati.
4. Un supervisore o un amministratore può inserire nel sistema degli avvisi (chiamati notifiche), che possono essere visualizzati da tutti i dipendenti. Ciascun dipendente può poi, qualora lo ritenesse necessario, marcare un avviso come "visualizzato" nascondendolo.

2.2 Requisiti non funzionali

Le modalità secondo le quali saranno offerte le funzionalità sopracitate sono le seguenti:

- **Usabilità** L'applicazione è stata sottoposta a una ricerca nel dettaglio di "User Personas".
- **Scalabilità** Scalabilità Il sistema deve avere un back-end in cloud scalabile per adattarsi a frequenze di accesso elevate.
- **Password policy security** Le password dell'utente verranno salvate in cloud con password crittografata.
- **Utilizzo di single-activity, multi-fragment** L'applicazione utilizza per fluidità e facile gestione il pattern di single-activity e multi-fragment in modo da dare precisi scopi alle activity che gestiscono fragment comuni.

2.3 Requisiti di dominio

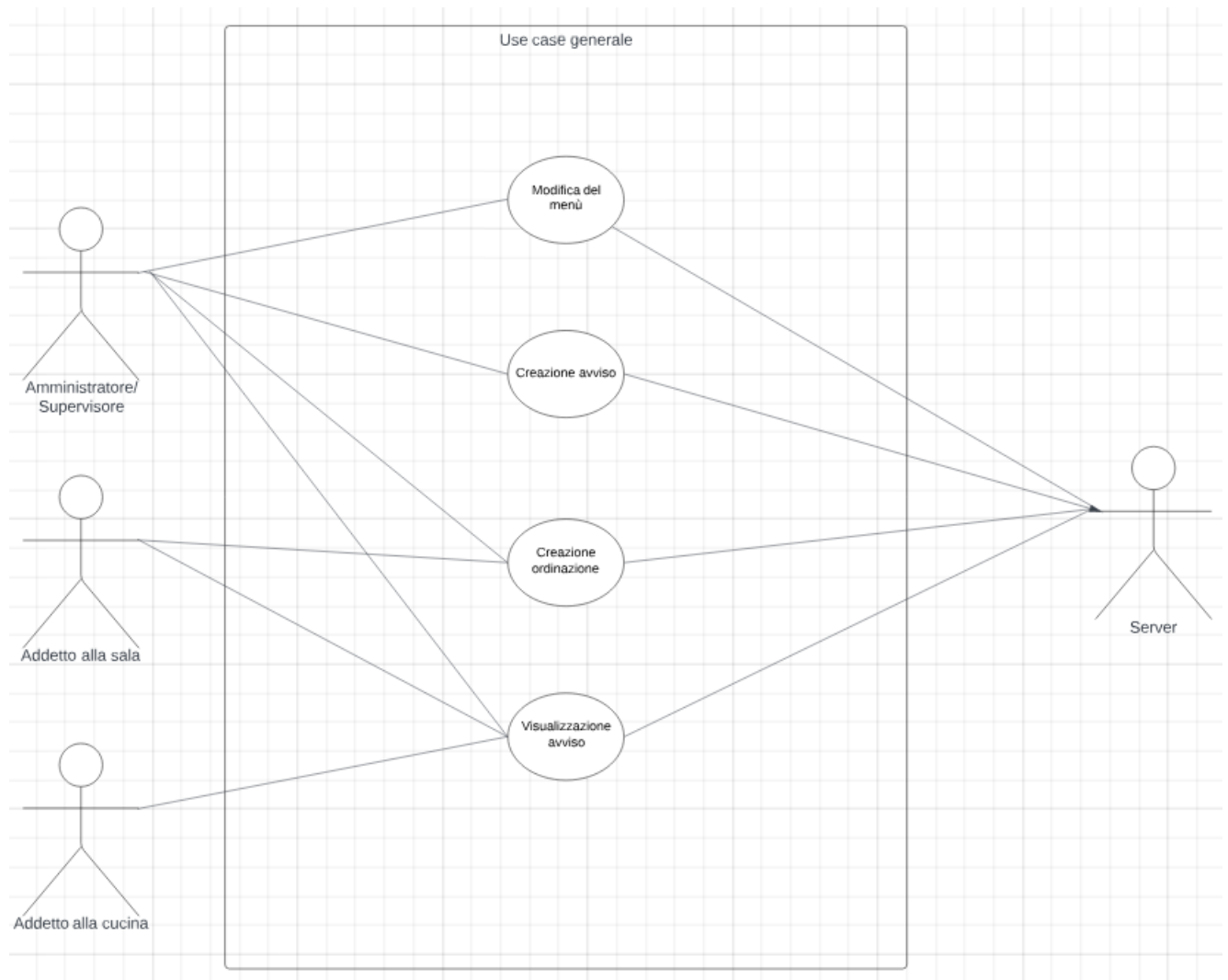
- **ISO/IEC** Il sistema deve essere conforme alle direttive *ISO/IEC* del trattamento dei dati privati su servizi di hosting in cloud.
- **GDPR** Il sistema segue le direttive europee sulla *GDPR* e quelle delle policy della privacy.

2.4 Modellazione dei casi d'uso

Di seguito sono descritti in dettaglio gli use-case scelti dal team di sviluppatori, al fine di dare un'idea concreta e chiara sul funzionamento del software sviluppato.

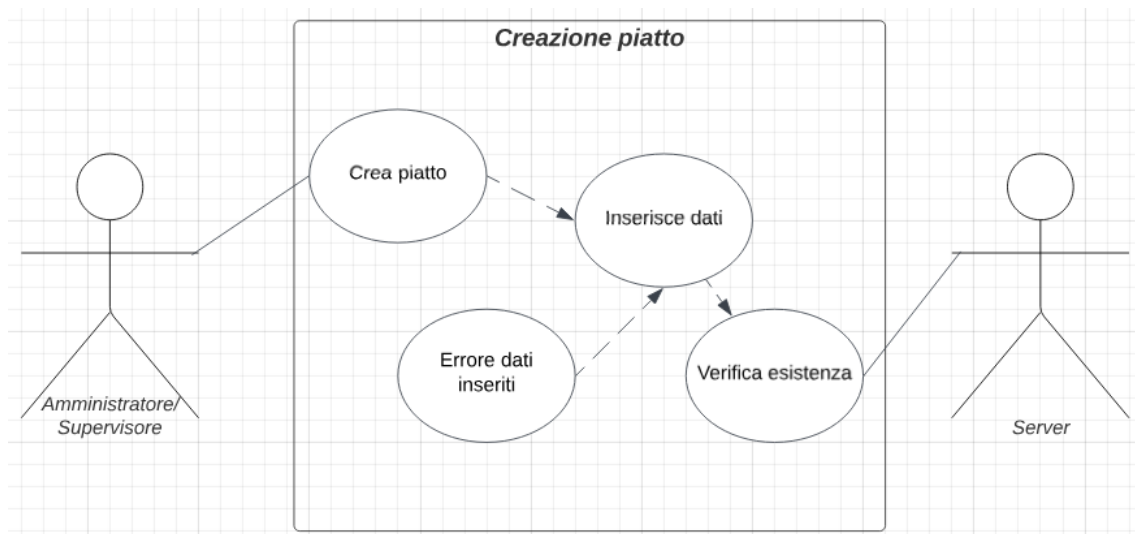
2.4.1 Use-case generale

Questo use-case è nato dopo aver attentamente valutato le richieste da parte degli stakeholders, infatti tale use-case contiene tutte le funzionalità richieste, mostrate in maniera semplificate solo ed esclusivamente per dare un'idea generale delle possibilità offerte dal software.



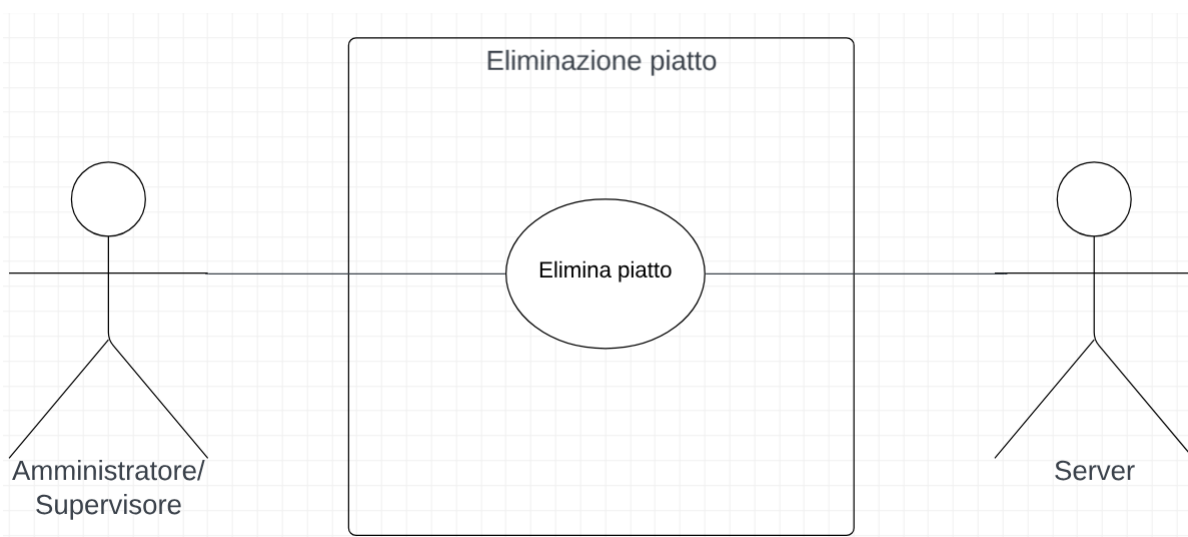
2.4.2 Creazione piatto

La creazione del piatto è una delle funzionalità più elaborate, in quanto, non basterà essere un amministratore/supervisore e inserire i dati (nome, descrizione, allergeni, categoria e prezzo) necessari alla creazione di un piatto. Verrà infatti controllata l'esistenza del piatto e, nel caso di esito positivo il piatto non verrà creato, altrimenti verrà aggiunto alla categoria selezionata.



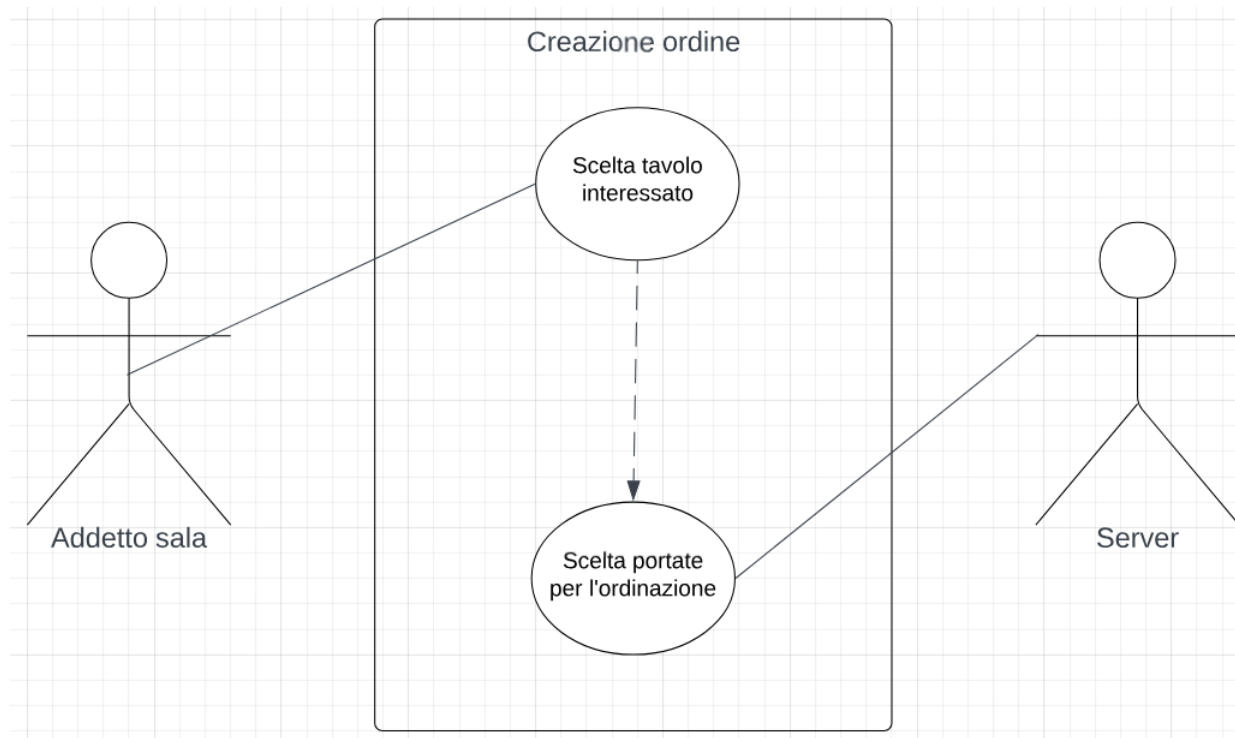
2.4.3 Eliminazione piatto

L'eliminazione del piatto è tutto sommato una funzionalità semplice ed autoesplicativa. Viene semplicemente scelto, da un amministratore/supervisore, il piatto che si desidera eliminare, una volta eliminato ci sarà una richiesta di conferma di eliminazione.



2.4.4 Creazione ordinazione

La creazione di un'ordinazione è un requisito fondamentale per la corretta gestione di un'attività di ristorazione che sia essa una tavola calda, un Autogrill o un classico ristorante. Per prima cosa è necessario che l'utente registrato che prova a creare un'ordinazione è un "addetto alla sala". Infatti nel caso in cui l'utente registrato non appartiene a questa categoria, non gli sarà permesso a priori la creazione di un'ordinazione. Per creare un'ordinazione è necessario scegliere un tavolo (tramite l'identificativo associato), e selezionare i piatti che si vogliono aggiungere all'ordinazione (sarà possibile registrare più ordinazioni per lo stesso tavolo in momenti separati).



2.4.5 Creazione notifica

La creazione di un avviso (notifica), è un aspetto fondamentale del software in quanto permette all'amministratore/supervisore di comunicare informazioni importanti indistintamente ad ogni utente. Infatti gli unici utenti con permessi per creare un avviso (notifica) sono proprio l'amministratore e i supervisori.



2.5 Tabelle cockburn

Ci è stato richiesto di presentare, tra i casi d'uso assegnati, quattro casi specifici a scelta descrivendoli tramite **tabelle di Cockburn**.

Cosa sono? Le tabelle di Cockburn (create da *Alistair Cockburn*, dal quale prendono il nome) sono un formalismo di rappresentazione dei casi d'uso.

Sono la rappresentazione di un main scenario (nel nostro primo caso la creazione di un'ordinazione) nel quale uno o più attori interagiscono tra loro attraverso l'invocazione di trigger e descrivendo (in formato tabellare) gli eventi.

Per garantire una certa coerenza al committente e una comprensione a tutto tondo abbiamo scelto gli stessi casi esplorati nella presentazione dei casi d'uso.

2.5.1 Creazione ordinazione

Use Case #1	Creazione ordinazione		
Goal in Context	Si vuole creare una nuova ordinazione		
Preconditions	L'utente deve essere autenticato correttamente ed in M2		
Success End Conditions	L'ordinazione è stata creata		
Failed End Conditions	L'ordinazione non è stata creata		
Primary Actor	Utente autenticato		
Trigger	L'utente autenticato preme il tasto "crea ordinazione"		
Description	Step	Utente autenticato	Sistema
	1	Preme "Aggiungi ordinazione"	
	2		Mostra M3
	3	Seleziona i piatti da inserire	
	4	Conferma l'ordine	
	5		Chiude M3
	6		Mostra messaggio di successo
Extensions#1: Categoria o piatto non validi	Step	Utente autenticato	Sistema
	6.1		Mostra messaggio d'errore
Extensions#2: Impossibile connettersi	Step	Utente autenticato	Sistema
	6.2		Mostra messaggio "Errore di connessione"

2.5.2 Eliminazione piatto

Use Case #2	Eliminazione piatto		
Goal in Context	L'utente vuole eliminare un piatto dal menù		
Preconditions	L'utente deve essere autenticato correttamente ed in M4		
Success End Conditions	Il piatto è stato eliminato		
Failed End Conditions	Il piatto non è stato eliminato		
Primary Actor	Utente autenticato		
Trigger	L'utente autenticato preme il tasto "elimina" sul piatto desiderato		
Description	Step	Utente autenticato	Sistema
	1		Mostra M5
	2	Clicca "ELIMINA"	
	3		chiude M5
	4		Mostra messaggio di successo
Extensions#1: L'utente clicca "INDIETRO"	Step	Utente autenticato	Sistema
	2.1	Clicca "INDIETRO"	
	3.1		Chiude M5
Extensions#2: Impossibile connettersi	Step	Utente autenticato	Sistema
	4.2		Mostra messaggio "Errore di connessione"

2.5.3 Creazione piatto

Use Case #3	Creazione piatto		
Goal in Context	L'utente vuole creare un piatto		
Preconditions	L'utente deve essere autenticato correttamente ed in M6		
Success End Conditions	Il piatto è stato creato		
Failed End Conditions	Il piatto non è stato creato		
Primary Actor	Utente autenticato		
Trigger	L'utente autenticato preme il tasto "crea piatto" in M6		
Description	Step	Utente autenticato	Sistema
	1	Clicca "crea piatto"	
	2		chiude M7
	3	Riempie i campi	
	4	Conferma la creazione	
	5		Chiude M7
	6		Mostra messaggio di successo
Extensions#1: campi non validi	Step	Utente autenticato	Sistema
	5.1		Svuota i campi inseriti
	6.1		Mostra messaggio d'errore
Extensions#2: Uno o più campi vuoti	Step	Utente autenticato	Sistema
	5.2		Svuota i campi inseriti
	6.2		Mostra messaggio d'errore
Extensions#3: Impossibile connettersi	Step	Utente autenticato	Sistema
	5.3		Mostra messaggio "Errore di connessione"

2.5.4 Creazione notifica

Use Case #4	Creazione notifica		
Goal in Context	L'utente vuole creare una notifica		
Preconditions	L'utente deve essere autenticato correttamente ed in M8		
Success End Conditions	La notifica è stata creata		
Failed End Conditions	La notifica non è stata creata		
Primary Actor	Utente autenticato		
Trigger	L'utente autenticato preme il tasto "crea notifica" in M8		
Description	Step	Utente autenticato	Sistema
	1	Clicca "crea notifica"	
	2		chiude M9
	3	Riempie i campi	
	4	Conferma la creazione	
	5		Chiude M9
	6		Mostra messaggio di successo
Extensions#1: Uno o più campi vuoti	Step	Utente autenticato	Sistema
	5.2		Svuota i campi inseriti
	5.1		Mostra messaggio d'errore
Extensions#2: Impossibile connettersi	Step	Utente autenticato	Sistema
	5.2		Mostra messaggio "Errore di connessione"

2.6 Mock-Up

Un mock-up , è una realizzazione a scopo illustrativo o meramente espositivo, di un oggetto o un sistema, senza le complete funzioni dell'originale; un mockup può rappresentare la totalità o solo una parte dell'originale di riferimento (già esistente o in fase di progetto), essere in scala reale oppure variata.



La versione del mock-up qui sopra rappresenta le schermate principali dell'applicazione. Nella realizzazione di quest'applicazione c'è stato uno studio del colore e dell'usabilità, con una palette creata appositamente per non stancare gli occhi e rendere il tutto più "vicino" al cibo possibile. Di seguito andremo a visualizzare i mock-up dei metodi scelti dal team.

2.6.1 Creazione ordinazione

Questa schermata rappresenta l'elenco dei tavoli già presenti nel database, quando verrà selezionato un tavolo premendoci sopra, il sistema aprirà **M2**.



M1: pagina dei tavoli

Una volta aperta **M2** ci verrà presentato l'elenco delle portate del tavolo selezionato (nel nostro caso, il tavolo 1). Quando l'utente cliccherà il pulsante "Aggiungi ordinazione" il sistema aprirà **M3**.



M2: Contenuto del tavolo n°1.

Ora che ci troviamo in **M3** potremo usare i due spinner (categoria e piatto) per selezionare il piatto che desideriamo aggiungere all'ordinazione e quando saremo soddisfatti potremo confermarlo.

TAVOLI

 Tavolo numero: 1

NUOVA ORDINAZIONE

Categoria:

PRIMI

Piatto:

Pasta al sugo

SECONDI	Aragosta	1211321€	
SECONDI	Aragosta	1211321€	
SECONDI	Aragosta	1211321€	
SECONDI	Aragosta	1211321€	
SECONDI	Aragosta	1211321€	
SECONDI	Aragosta	1211321€	
SECONDI	Aragosta	1211321€	

CONFERMA ORDINAZIONE

M3: Nuova ordinazione per il tavolo n°1.

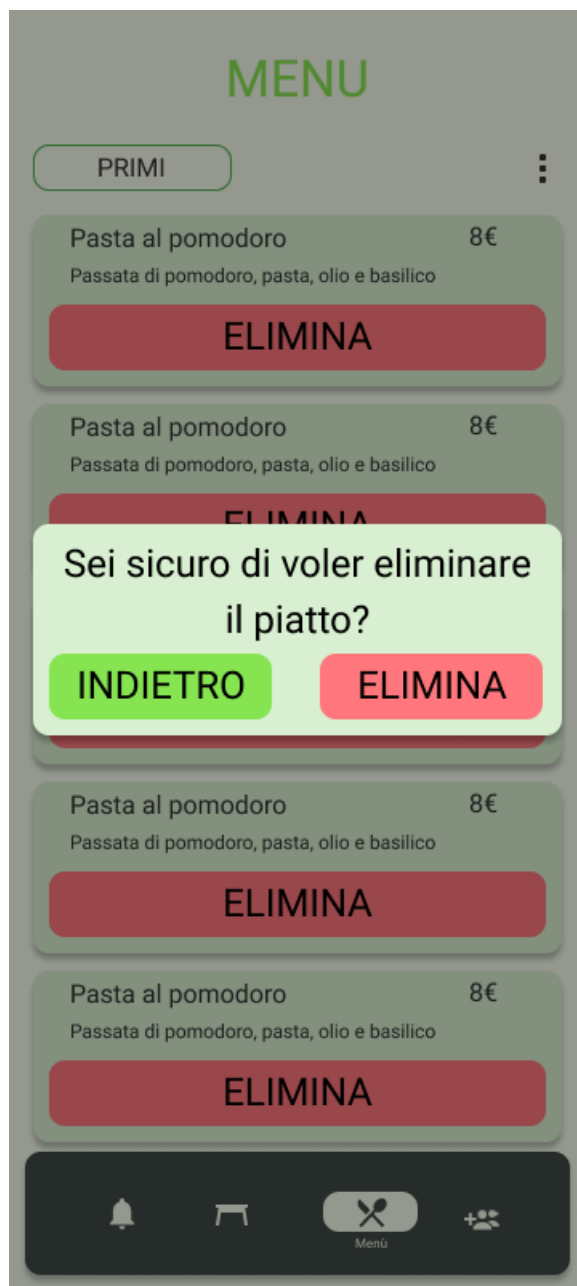
2.6.2 Eliminazione piatto

La schermata riportata qui sotto mostra il menù del ristorante con la possibilità di eliminare i piatti da esso. Qualora l'utente dovesse cliccare "ELIMINA" su uno dei piatti si aprirà **M5**.



M4: Menù ristorante con piatti da eliminare.

In questa schermata l'utente potrà confermare l'eliminazione del piatto oppure tornare indietro, annullando così l'eliminazione.



M5: Conferma eliminazione piatto.

2.6.3 Creazione piatto

La creazione del piatto è una delle funzionalità più importanti, infatti è quella che ci permette di popolare il menù con piatti creati da noi. Per creare un piatto, bisognerà che l'utente di trovi nella schermata del menù (M6).



M6: Schermata del menù.

In questa schermata, tramite i tre puntini in alto a destra, possiamo attivare la funzione di creazione di un piatto, che ci aprirà **M7**.

The screenshot shows a mobile application interface for a menu. At the top, the word 'MENU' is displayed in green. Below it, there is a button labeled 'PRIMI' and a vertical ellipsis menu icon. The main content area lists two items under the 'PRIMI' category: 'Pasta al pomodoro' for 8€, with a description 'Passata di pomodoro, pasta, olio e basilico'. Below this, there is a large green overlay titled 'NUOVO PIATTO' (New Dish). This overlay contains four input fields: 'Nome' (Name) with the value 'Pasta al sugo', 'Descrizione' (Description) with the value 'Pasta con passata di pomodoro.', 'Allergeni' (Allergens) with the value 'Glutine', and 'Prezzo' (Price) with the value '12.5'. At the bottom of the overlay, there is a 'Categoria:' label and a button labeled 'PRIMI'. A large green button at the very bottom of the overlay is labeled 'CREA PIATTO'.

M7: Schermata del menù.

Una volta in questa schermata non bisognerà fare altro che riempire i campi necessari e confermare la creazione.

2.6.4 Creazione notifica

La creazione di una notifica, permette ad un amministratore/supervisore, di mandare un messaggio a tutti i suoi dipendente, così da poter comunicare avvisi importanti indistintamente ad ogni dipendente qualora si voglia. Per farlo, l'utente deve trovarsi in **M8** (↓) e premere "CREA NOTIFICA".



M8: Schermata delle notifiche.

Una volta premuto "CREA NOTIFICA", il sistema mostrerà **M9** (↓), al cui interno ci saranno due campi richiesti (titolo e testo, la data verrà prelevata in automatico dal sistema), una volta compilati, l'utente può premere "CREA NOTIFICA".

NOTIFICHE

Lorem Ipsum dolor sit
17/12/1999

Lorem Ipsum dolor sit
17/12/1999

Lorem Ipsum dolor sit
17/12/1999

Lorem Ipsum dolor sit

—

NUOVA NOTIFICA

Titolo
Notifica 1

Testo
Questa è una nuova notifica

CREA NOTIFICA

M9: Schermata per la creazione di una notifica.

2.7 Individuazione del target a priori

Ratatouille23TM nasce con l'idea di creare un sistema efficiente e affidabile per la gestione di attività di ristorazione. Nello specifico poter permettere ad un responsabile di poter dividere dettagliatamente le mansioni di un dipendente con la possibilità di assumere dei collaboratori (e.g. supervisore). Il target che si può definire da una prima analisi delle funzionalità e casi d'uso, è il seguente:

- Camerieri e manager di ristorante.
- Cuochi, sous chef e aiuto cuoco.
- Proprietario e dirigente di ristorante.

Questa prima analisi può fornirci una visione sommaria delle personas.

2.7.1 Personas

Il focus principale per noi è stato quello di analizzare 2 aspetti che abbiamo ritenuto fondamentali per individuare le user personas così da poter sviluppare un software cucito su misura:

1. **Età:** la fascia d'età che abbiamo potuto notare essere più conforme all'utente ideale è quella dei 18-65 anni, in modo da comprendere uno studente ma anche un cuoco di fama mondiale con alle spalle numerosi anni di esperienza.
2. **Background:** per background s'intende la formazione dell'utente. Infatti un nostro utente può essere uno studente, che lavora il weekend per poter pagare l'affitto da fuorisede, così come può essere un ragazzo con formazione *sommelier* o una ragazza che ha conseguito un corso all'*ALMA*.

Questo ci permette di pianificare sviluppi futuri del nostro software. Potremmo infatti creare una sezione dedicata ai vini, facendo in modo che il sommelier consigli in base ai piatti ordinati, quali vini si accostano meglio.

Potremmo aggiungere una sezione con dei percorsi culinari, studiati in precedenza da un personale esperto e preparato. Così facendo potremo fidelizzare i nostri utenti dandogli completa libertà per la gestione della loro attività.

Lucia Zhang La prima user persona è stata utilizzata per raccogliere il target di utenti nella fascia compresa tra i 25-35 anni, che hanno un'alta competenza con la tecnologia e che sono già avviati, nel mondo del lavoro.



The user persona card for Lucia Zhang is presented in a light green, rounded rectangular layout. It is divided into several sections: a header with the name 'Lucia Zhang' and a circular profile picture of a woman; a table of personal details; a biography; sections for needs, disincentives, and values; and a personality section with three tags.

Lucia Zhang



AGE	27
EDUCATION	Tecniche di cucina ALMA
OCCUPATION	Cuoco
LOCATION	Napoli, it

Biografia

Vive a Napoli, ha conseguito un diploma alberghiero con specializzazione in cucina e successivamente ha conseguito un corso di cucina all'ALMA. Alta competenza della tecnologia, alta competenza in ambito culinario.

Necessità

- Poter cancellare le vecchie notifiche, rendendo l'applicazione più ordinata
- Applicazione veloce ed intuitiva

Disincentivi

- Le applicazioni che ho usato non mi permettevano di cancellare le notifiche, creando casino.

Personalità

Curiosa Diligente Cinica

Valori

- Poter deliziare i miei clienti con le mie creazioni.

Fig.1: User persona n.1.

Stefano Iamunno La seconda user persona è stata utilizzata per raccogliere il target di utenti nella fascia d'età compresa tra i 18-25 anni, che hanno un'alta competenza tecnologica ma che si sono appena approcciati al mondo del lavoro.

Stefano Iamunno



AGE

22

EDUCATION

Diploma alberghiero

OCCUPATION

Addetto alla sala

LOCATION

Bologna, IT

Personalità

Introverso

Ambizioso

Nerd

Biografia

Vive a Bologna, ha conseguito il diploma alberghiero circa 4 anni fa. Ha una grande dimestichezza con i dispositivi tecnologici.

Necessità

- Necessità di avere un app versatile e veloce nell'esecuzione.
- Non mi piace ricevere email, vorrei ricevere notifiche all'interno dell'app.

Disincentivi

- I software in circolazione sono lenti.
- Le app che ho usato sono sempre state trascurate nell'estetica.

Valori

- Poter prendere ordinazioni intuitivamente e in modo veloce.

Fig.2: User persona n.2.

Maria Colombo La terza user persona è stata utilizzata per raccogliere il target di utenti nella fascia d'età compresa tra i 35-65 anni, che hanno una bassa competenza tecnologica ma con una vasta esperienza nel mondo del lavoro. Sono utenti che con la loro esperienza lavorativa possono assumere pressochè qualsiasi ruolo all'interno di un'attività di ristorazione.

Maria Colombo



AGE	52
EDUCATION	Managment
OCCUPATION	Supervisore
LOCATION	Milano, IT

Biografia

Vive a Milano, ha 10 anni di esperienza nel settore manageriale. Ha una conoscenza sufficiente della tecnologia ma ottime competenze manageriali.

Necessità

- Poter parlare ai dipendenti anche senza essere presente in ufficio.
- Gestire il menù potendo creare nuovi piatti dando sfogo alla fantasia
- Assumere nuovi impiegati ed inserirli nel sistema.

Frustrations

- I software in circolazione sono "brutti" e poco intuitivi.
- I software che ho usato avevano poca libertà creativa.

Valori

- Poter gestire i dipendenti come se fossi "una direttrice d'orchestra".

Personalità

Estroversa

Creativa

Pragmatica

Fig.3: User persona n.3.

2.8 Valutazione dell'usabilità a priori

L'usabilità di un prodotto è il grado con cui esso può essere usato da specificati utenti per raggiungere specificati obiettivi con efficacia, efficienza e soddisfazione in uno specificato contesto d'uso.

Facendo riferimento allo standard **ISO 9241**(↑), abbiamo basato la valutazione dell'usabilità a priori principalmente su questa definizione in quanto semplice ed efficace, e sulle modalità a nostra disposizione.

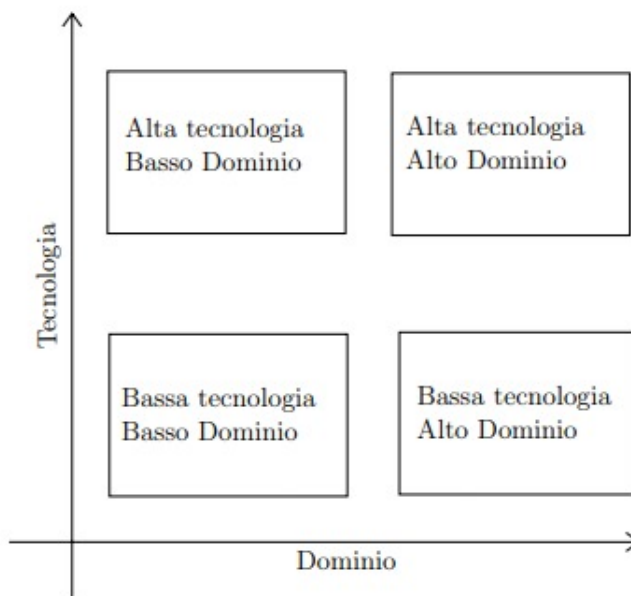
In particolare, per ottenere una valutazione dell'usabilità a priori oggettiva e corretta, abbiamo speso molto tempo nella fase di progettazione, della progettazione UI e delle funzionalità, in modo da poter riprodurre dei prototipi della nostra applicazione quanto più realistici possibile. Questo ci ha permesso di apportare poche ma essenziali modifiche alla versione finale, rendendo il tutto più robusto e affidabile.

Per realizzare i mock-up abbiamo usato Figma, quest'ultimo ci permette di creare dei mockup dinamici, simulando il funzionamento dell'applicazione, senza però tralasciare l'aspetto estetico, potendo così fornire una visione concreta della nostra applicaizione.

Le 8 regole d'oro I mock-up, e quindi l'applicazione poi, sono stati realizzati seguendo alcune delle 8 regole di **Shneiderman** con particolare attenzione a 4 delle 8 regole:

- **Riscontri informativi:** abbiamo aggiunto dei *pop-up*, schermate di caricamento e messaggi "toast", in modo da guidare l'utente durante tutta l'esperienza di utilizzo. In particolare ad ogni azione dell'utente viene notificata tramite messaggio "toast", se l'azione è andata a buon fine oppure c'è stato un errore. Abbiamo utilizzato la schermata di caricamento principalmente durante il login, così che se la richiesta al server dovesse richiedere più tempo, l'utente lo saprà e saprà quando la richiesta è terminata. Per quanto riguarda il *pop-up* è stato utilizzato nell'eliminazione del piatto. Infatti quando si proverà ad eliminare un piatto il sistema mostrerà un *pop-up* di conferma per evitare qualsiasi eliminazione fortuita.
- **Riduzione del carico di memoria a breve termine:** è stato garantito non richiedendo informazioni complesse all'utente. Inoltre l'utente non dovrà preoccuparsi di "portare" informazioni da una schermata all'altra.
- **Usabilità universale:** l'applicazione è stata sviluppata tenendo conto di tutte le fasce d'età infatti è estremamente semplice e intuitiva nel suo utilizzo. Le funzionalità sono state implementate nella maniera più semplice possibile, richiedendo meno informazioni possibili e nel caso in cui ci fosse la necessità di chiedere più informazioni, è stato fatto nel modo più graduale possibile.
- **Coerenza a tutti i costi:** le funzionalità sono state sviluppate in modo "simile", creazione notifiche e creazione piatto ad esempio, sono pressochè la stessa funzionalità, rendendo così l'applicazione coerente e di facile utilizzo.

Dopo la realizzazione dei prototipi, è stato effettuato un test di usabilità. Per garantire un risultato oggettivo, sono stati scelti attentamente dei valutatori che andranno utilizzati sui prototipi. Qui sotto è riportato un grafico che rappresenta 4 tipi di valutatori



Ricordandoci "la regola di **Nielsen**", abbiamo scelto 5 tester per l'applicazione. I 5 utenti mettono in evidenza la maggior parte dei problemi di usabilità (circa il 75%) più significativi (aggiungerne di più comporterebbe un aumento dei costi inutile). Gli utenti sono i seguenti:

- **Valutatore:** con bassa conoscenza tecnologica, bassa conoscenza del dominio (Fabiana E.).
- **Valutatori x2:** bassa conoscenza tecnologica, alta conoscenza del dominio (Antonio L., Roberto P.).
- **Valutatore:** con alta conoscenza tecnologica, bassa conoscenza del dominio (Corrado R.).
- **Valutatore:** con alta conoscenza tecnologica, alta conoscenza del dominio (Ciro C.).

È importante che gli utenti scelti (al di fuori del gruppo di progetto, in modo da non avere risultati falsati) abbiano caratteristiche diverse, scelti con cura in modo da rappresentare la più ampia fascia d'utenti (per quanto possibile).

2.8.1 Tecnica utilizzata

Per lo svolgimento dei test è stata utilizzata la tecnica del *Mago di Oz*, che consiste nel realizzare un prototipo interattivo, in cui le risposte, se possibile all'insaputa dell'utente, da parte di un essere umano che operi "dietro le quinte". nella maniera più fedele possibile.

Compiti assegnati Ai nostri utenti sono stati assegnati i seguenti 4 compiti da svolgere:

- **Compito 1:** Cambio password.
- **Compito 2:** Creazione piatto.
- **Compito 3:** Cancellazione piatto.
- **Compito 4:** Visualizzazione notifica.

Nella seguente tabella è possibile visualizzare i risultati dei test:

	COMPITO 1	COMPITO 2	COMPITO 3	COMPITO 4
Fabiana E.	S	S	S	P
Antonio L.	S	F	S	S
Roberto P.	S	S	S	S
Corrado R.	S	P	P	S
Ciro C.	S	P	S	S

Table 1: F = fallimento = 0; P = successo parziale = 0.5; S = successo = 1

Avendo così un totale di $17/20 = (15 + (4 * 0.5))$ compiti eseguiti per una percentuale di 85%.

Feedback degli utenti i nostri tester ci hanno evidenziato come:

- L'inserimento del prezzo era lento e incline ad errori, abbiamo sostituito così l'inserimento a mo' di orario con un semplice EditText.
- La visualizzazione delle notifiche era controintuitiva a primo impatto, infatti non tutti sono riusciti subito a trovare il tasto "visualizza", adesso con un semplice tocco della notifica appare il tasto "visualizza".

P.S. in principio erano stati assegnati dei punteggi anche al tempo impiegato, abbiamo però notato che il nostro gruppo di tester si aggirava più o meno sullo stesso tempo impiegato, abbiamo quindi deciso fosse inutile (nel nostro caso) lasciare quei punteggi.

3 Specifica dei casi d'uso

In questa sezione si analizzeranno le classi e come sono relazionate tra loro, daremo un'occhiata ai sequence diagram (in particolare quelli riguardanti due funzionalità dell'applicazione) e agli statechart relativi all'interfaccia grafica. Per fare ciò utilizzeremo:

1. Class diagram.
2. Sequence diagram.
3. State chart

3.1 Classi, oggetti e relazioni di analisi

In questa sezione utilizzeremo i *class diagram* per farci un'idea della relazione tra le classi, prima di tutto diamo uno sguardo alla struttura generale che abbiamo creato in fase di analisi:

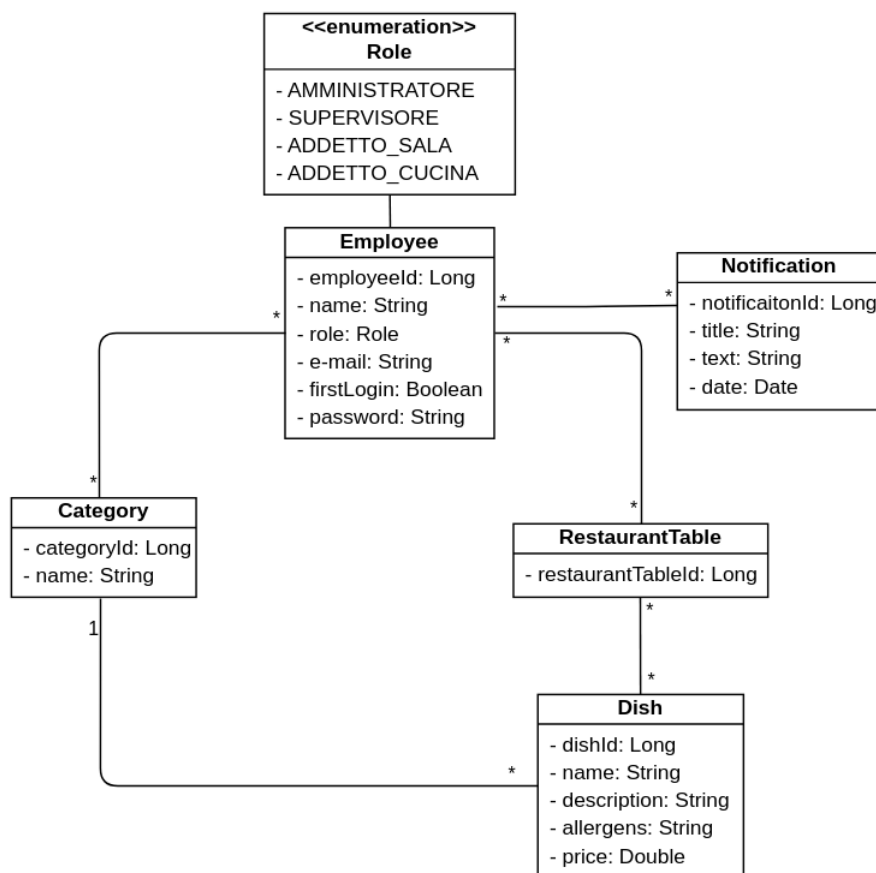


Fig.4: Class diagram delle entità

D'ora in poi invece, i *class diagram* mostrati, saranno quelli delle funzionalità che sono state assegnate al team di sviluppo.

3.2 Class diagram di analisi - Gestione menù

In questo class diagram viene analizzato il punto 3 della traccia ovvero la gestione del menù.

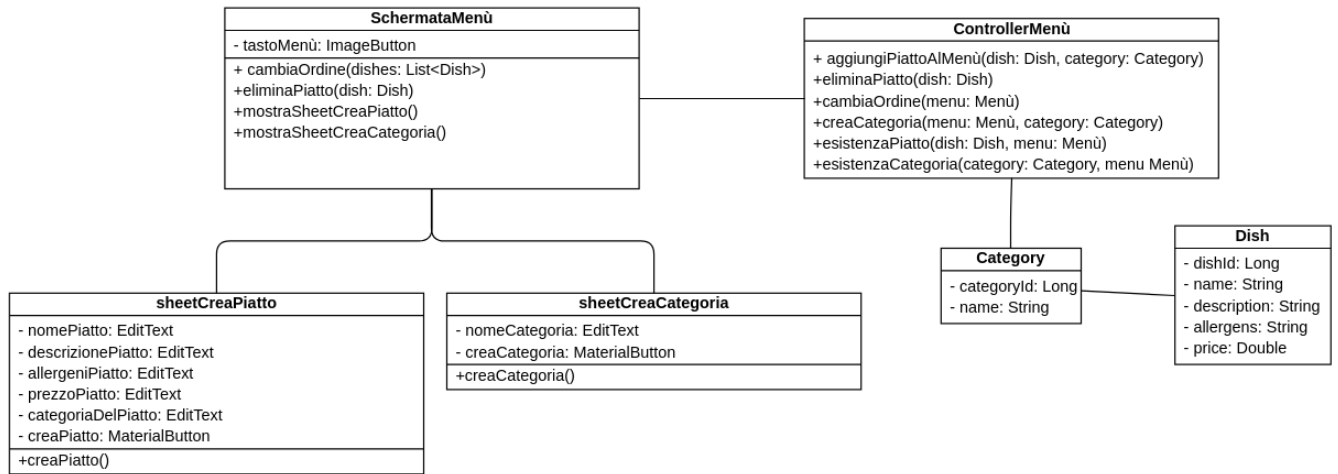


Fig.5: Class diagram della gestione del menù

3.3 Class diagram - Creazione utenza

Nel seguente class diagram verrà analizzato il punto 1, quello nel quale si richiede la creazione di un utente e l'eventuale cambio password nel caso in cui si tratti del primo login, il cambio password verrà però affrontato in un *class diagram* successivo, in modo da poter comprendere anche l'accesso alla piattaforma.

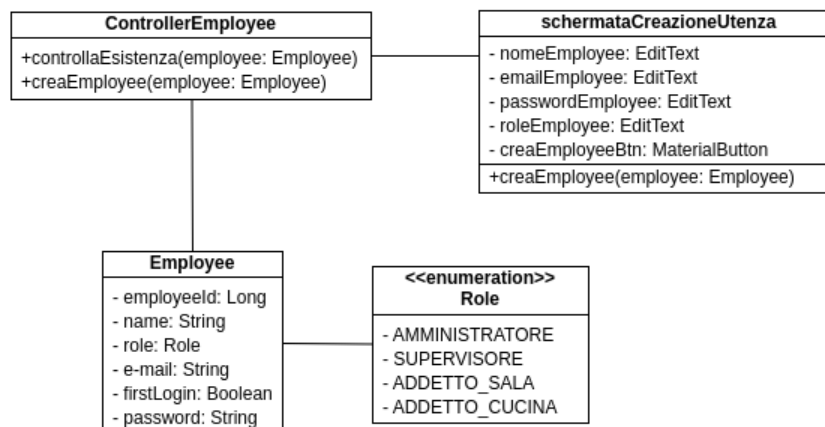


Fig.6: Class diagram per la creazione utenze

3.3.1 Class diagram - Accesso alla piattaforma

Adesso, come detto in precedenza, analizzeremo l'accesso (con conseguente cambio password) alla piattaforma.

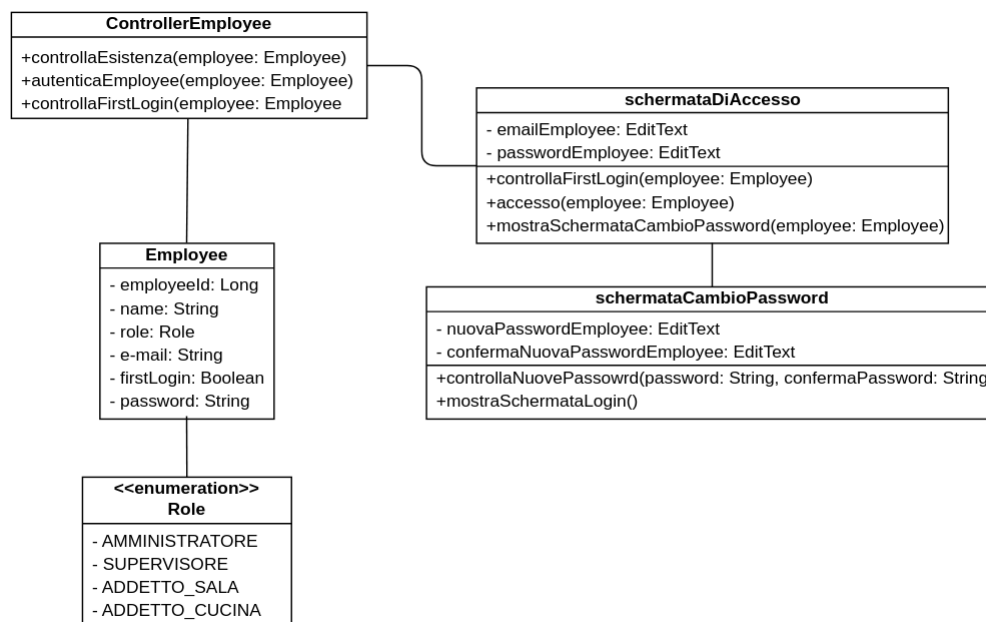


Fig.7: Class diagram dell'accesso alla piattaforma

3.4 Class diagram - Gestione tavoli

Il seguente *class diagram* analizza il punto 6, quindi creazione ordinazioni indicando l'identificativo del tavolo e gli elementi del menù da aggiungere al tavolo.

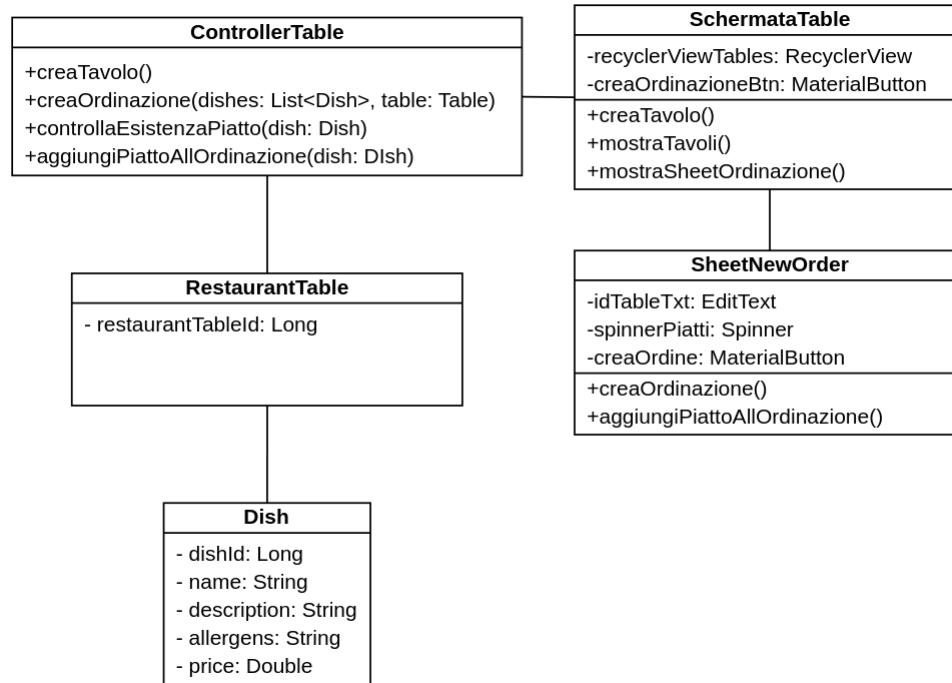


Fig.8: Class diagram della gestione dei tavoli

3.5 Class diagram - Gestione notifiche

Il seguente *class diagram* analizza il punto 13, quindi creazione di avvisi (notifiche), al quale verrà aggiunta la visualizzazione delle notifiche.

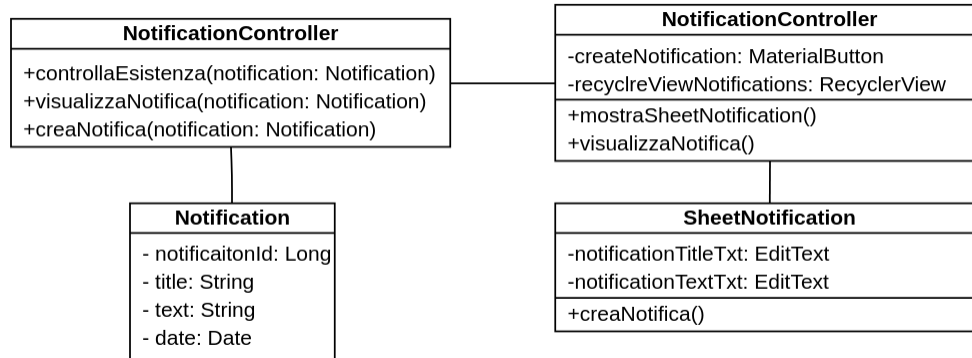


Fig.9: Class diagram della gestione delle notifiche