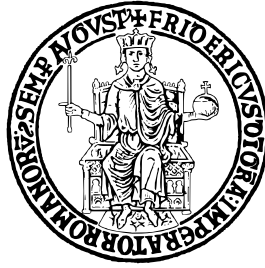


UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II



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

<b>1</b>	<b>Introduzione</b>	<b>2</b>
1.1	Cosa contiene questa documentazione . . . . .	2
1.2	Tecnologie utilizzate . . . . .	2
<b>2</b>	<b>Modello funzionale</b>	<b>3</b>
2.1	Requisiti del software . . . . .	3
2.1.1	Requisiti funzionali . . . . .	3
2.2	Requisiti non funzionali . . . . .	3
2.3	Requisiti di dominio . . . . .	4
2.4	Descrizione use-case . . . . .	4
2.4.1	Use-case generale . . . . .	4
2.4.2	Creazione piatto . . . . .	5
2.4.3	Eliminazione piatto . . . . .	5
2.4.4	Creazione ordinazione . . . . .	6
2.4.5	Creazione notifica . . . . .	7
2.5	Tabelle cockburn . . . . .	8
2.5.1	Creazione Ordinazione . . . . .	8
2.5.2	Eliminazione piatto . . . . .	9
2.5.3	Creazione piatto . . . . .	9
2.5.4	Creazione notifica . . . . .	10
2.6	Mock-Up . . . . .	11
2.6.1	Creazione piatto . . . . .	11
2.6.2	Eliminazione piatto . . . . .	11
2.6.3	Creazione ordinazione . . . . .	11
2.6.4	Creazione notifica . . . . .	11

# 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 Modello funzionale

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" e monitoraggio tramite servizi official Google.inc.
- **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 un controllo regex avanzato (vedesi testing) e sicurezza su crittografia SHA-256.
- **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

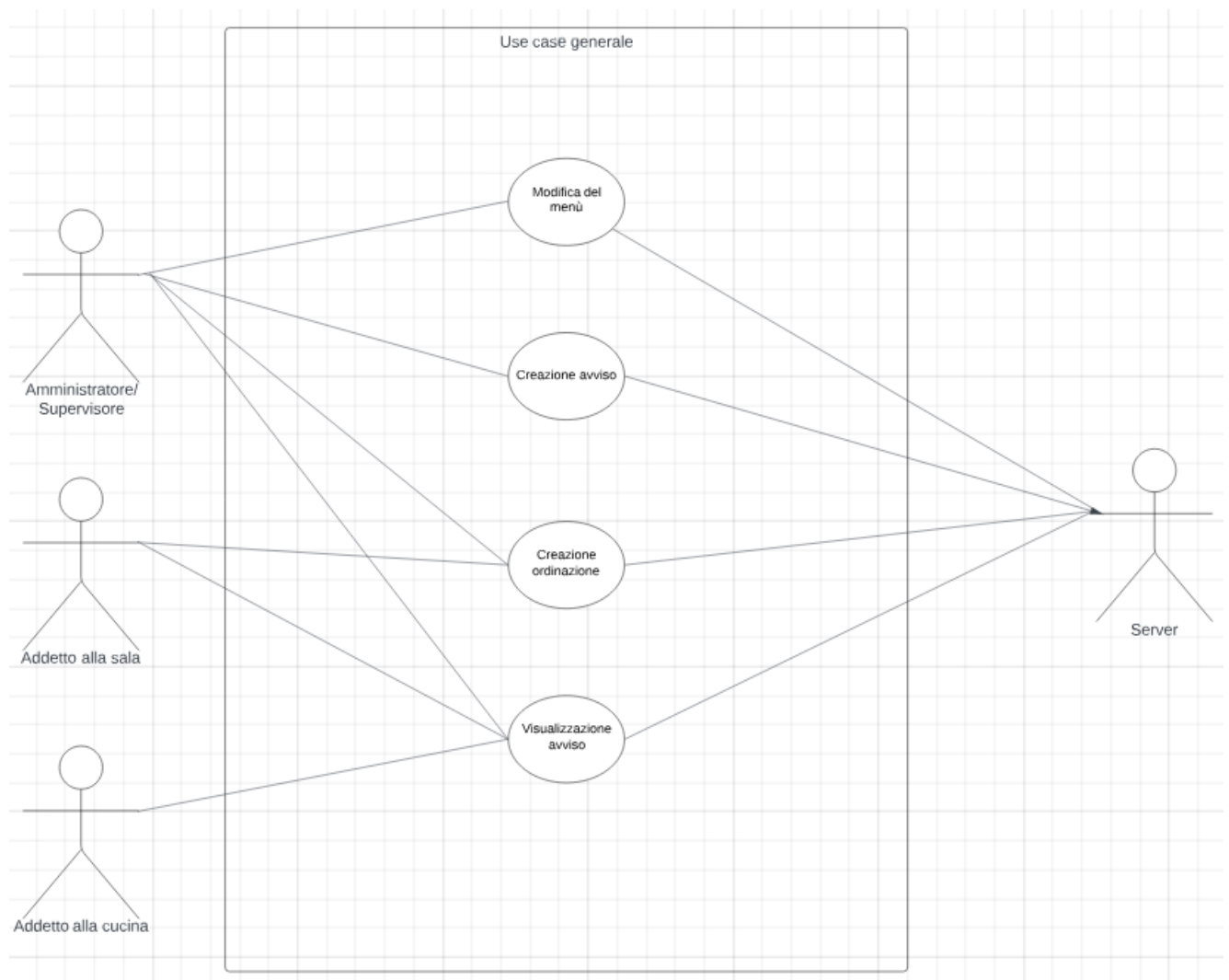
- **ISO/IEC** Il sistema deve essere conforme alle direttive *ISO/IEC* del trattamento dei dati privati su servizi di hosting in cloud.
- **GDPR** Il sistema segue le direttive europee sulla *GDPR* e quelle delle policy della privacy.

## 2.4 Descrizione use-case

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.

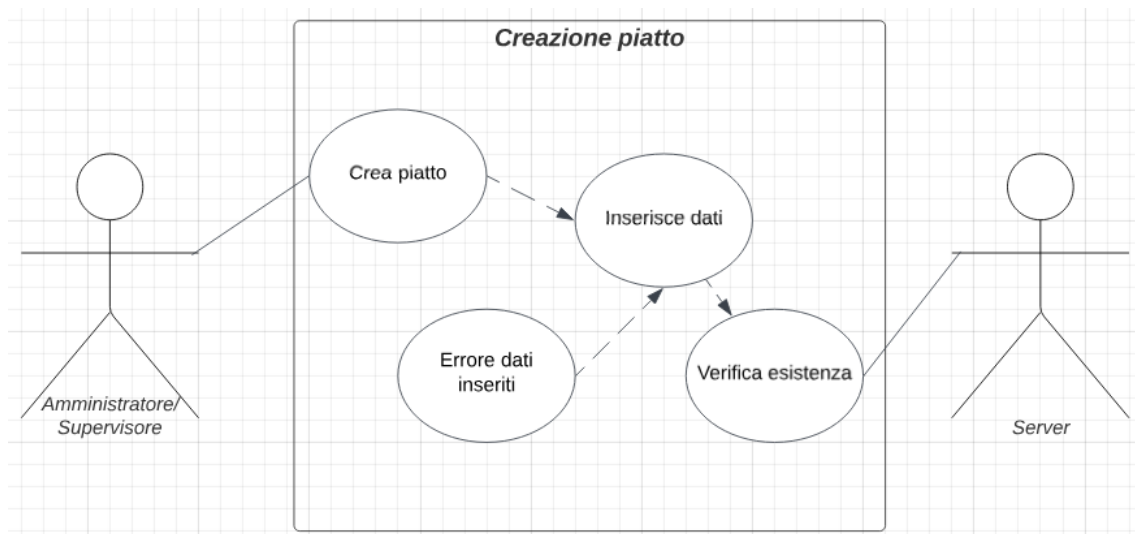
### 2.4.1 Use-case generale

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.



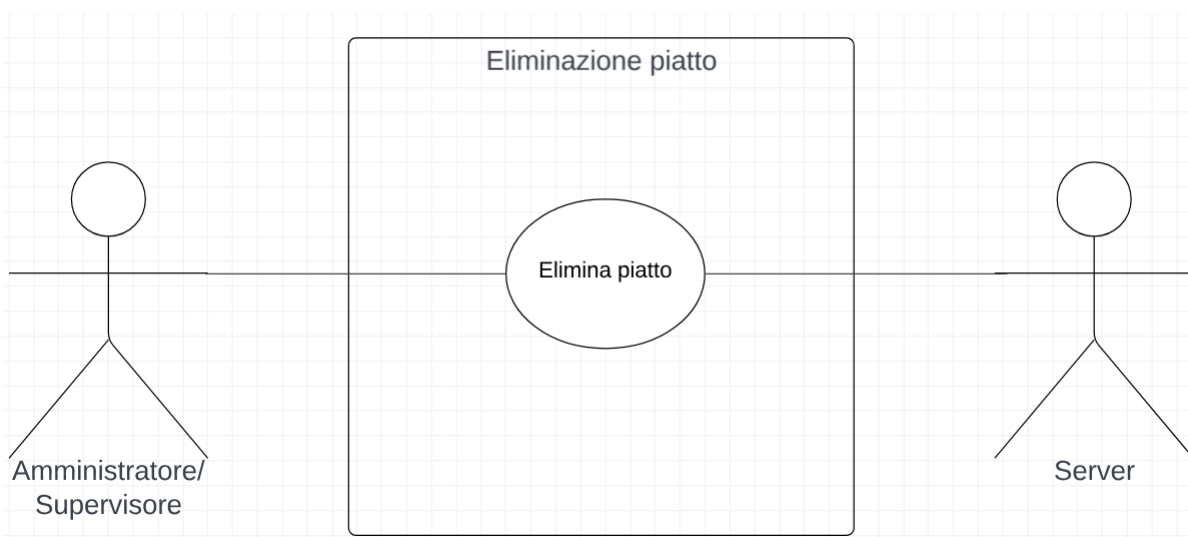
### 2.4.2 Creazione piatto

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.



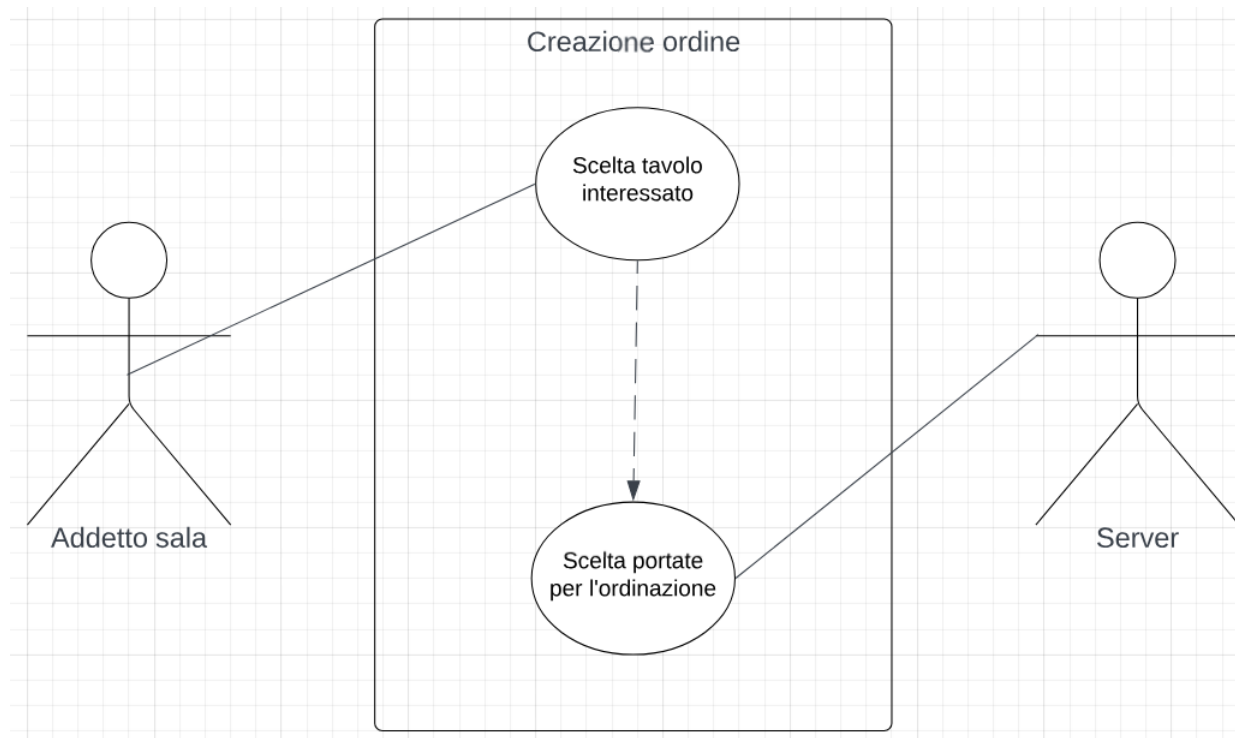
### 2.4.3 Eliminazione piatto

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.

La politica adotta la rappresentazione di un Main scenario nella 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		
<b>Goal in Context</b>	L'addetto alla sala vuole creare una nuova ordinazione per un tavolo		
<b>Preconditions</b>	L'addetto alla sala deve essersi autenticato correttamente		
<b>Success End Conditions</b>	La nuova ordinazione è stata creata		
<b>Failed End Conditions</b>	Le informazioni inserite non sono valide		
<b>Primary Actor</b>	Utente autenticato		
<b>Trigger</b>	Pressione del tasto "crea ordinazione"		
<b>Description</b>	<b>Step</b>	<b>Addetto sala</b>	<b>Server</b>
	1	Sceglie il tavolo	
	2	Sceglie i piatti da inserire	
	3	Conferma l'ordine	Verifica esistenza del tavolo
	4		Verifica esistenza dei piatti inseriti
	5		Registra ordine

### 2.5.2 Eliminazione piatto

<b>Use Case #2</b>	<b>Eliminazione piatto</b>		
<b>Goal in Context</b>	L'utente vuole eliminare un piatto dal menù		
<b>Preconditions</b>	L'utente è autenticato correttamente ed ha i permessi		
<b>Success End Conditions</b>	Il piatto è stato eliminato		
<b>Failed End Conditions</b>	Il piatto che ha provato ad eliminare non è valido		
<b>Primary Actor</b>	Amministratore/supervisore		
<b>Trigger</b>	Pressione tasto "elimina piatto"		
<b>Description</b>	<b>Step</b>	<b>Client</b>	<b>Server</b>
	1	Seleziona il piatto da eliminare	
	2	Apertura dialog di conferma	
	3	Conferma l'eliminazione	
	4		Elimina il piatto

### 2.5.3 Creazione piatto

<b>Use Case #3</b>	<b>Creazione piatto</b>		
<b>Goal in Context</b>	L'utente vuole creare un piatto da aggiungere al menù		
<b>Preconditions</b>	L'utente è collegato correttamente ed ha i permessi necessari		
<b>Success End Conditions</b>	Il piatto è stato creato		
<b>Failed End Conditions</b>	Le informazioni inserite sono già presenti o non valide		
<b>Primary Actor</b>	Amministratore/supervisore		
<b>Trigger</b>	Apertura schermata creazione piatto		
<b>Description</b>	<b>Step</b>	<b>Client</b>	<b>Server</b>
	1	Inserisce i dati richiesti per la creazione	
	2	Clicca tasto "crea piatto"	
	3		Verifica la possibilità della creazione del piatto
	4		Salva il piatto

**2.5.4 Creazione notifica**

<b>Use Case #4</b>	<b>Creazione notifica</b>		
<b>Goal in Context</b>	L'utente vuole creare una notifica da inviare ai dipendenti		
<b>Preconditions</b>	L'utente è autenticato correttamente ed ha i permessi		
<b>Success End Conditions</b>	La notifica è stata creata		
<b>Failed End Conditions</b>	La notifica con la stessa data è già presente		
<b>Primary Actor</b>	Amministratore/supervisore		
<b>Trigger</b>	Pressione tasto "crea notifica"		
<b>Description</b>	<b>Step</b>	<b>Client</b>	<b>Server</b>
	1	Inserisce titolo e testo della notifica	
	2	Preme il tasto per confermare la creazione	Verifica se la notifica non è già presente
	3		Salva la notifica

## 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 piatto

### 2.6.2 Eliminazione piatto

### 2.6.3 Creazione ordinazione

### 2.6.4 Creazione notifica