



**Università degli Studi di Padova**

Laurea in Informatica

Corso di Ingegneria del Software

Anno Accademico 2023/2024



**Gruppo:** SWEet16

**Email:** [sweet16.unipd@gmail.com](mailto:sweet16.unipd@gmail.com)

# Verbale Esterno

## 7 Dicembre 2023

Redattori:	Alberto M.
Verificatori:	Alex S.
Amministratore:	Alberto C.
Destinatari:	T. Vardanega   R. Cardin
Versione:	1.0.0

# Registro delle modifiche

Versione	Data	Autore	Verificatore	Descrizione
1.0.0	2023/12/14	Alberto M.		Approvazione per rilascio
0.2.0	2023/12/13	Sig. Staffolani		Apposizione firma
0.1.0	2023/12/07	Alberto M.	Alex S.	Prima stesura del documento

# 1 Partecipanti

Di seguito i nomi dei partecipanti con le rispettive matricole:

Nome	Matricola
Alberto Cinel	1142833
Bilal El Moutaren	2053470
Alberto Michelazzo	2010007
Alex Scantamburlo	2042326
Iulius Signorelli	2012434

Ha inoltre partecipato il Sig. Alessandro Staffollani, rappresentante di *Imola Informatica*.

**Inizio incontro:** Ore 15:00

**Fine incontro:** Ore 15:45

## 2 Sintesi dell'incontro

In questo incontro avvenuto con *Imola Informatica* via Microsoft Teams, dopo una prima stesura dei vari casi d'uso, abbiamo illustrato i nostri dubbi riscontrati al sig. Staffolani. I nostri più grandi dubbi sono sulla prenotazione ed ordinazione condivisa, e su come l'amministratore di ristorante debba rapportarsi rispetto alle ordinazioni dei clienti precedentemente il loro arrivo al ristorante, in modo da creare un consuntivo<sup>G</sup> degli ingredienti necessari.

Di seguito le domande che sono state poste al rappresentante aziendale.

## 3 Domande e risposte

**Il sistema come deve gestire la situazione in cui non tutti gli utenti di una prenotazione hanno fatto il proprio ordine?**

Potete gestirlo come preferite voi, ad esempio si può decidere che in caso un utente (tra quelli invitati) ignori e non confermi la sua partecipazione allora potrà solo ordinare direttamente al ristorante.

Chi invece ha dato conferma della sua partecipazione viene aggiunto alla lista delle persone che possono ordinare anticipatamente.

**La nostra idea era di aggiungere un campo "mail degli amici da invitare" al momento della prenotazione, tuttavia questo genererebbe un problema in quanto non avremmo modo di poter validare se le email effettivamente esistono.**

**Può andare bene come idea?**

Potete eliminare in un primo momento questo campo dalla prenotazione e semplicemente generare un link per la condivisione della prenotazione, che poi potrà essere condiviso come ogni utente preferisce (WhatsApp<sup>G</sup>, Telegram, email, ecc.)

**Il ristorante può vedere il consuntivo degli ingredienti solo di un giorno o anche di un periodo più ampio? Noi cosa dobbiamo offrire come funzionalità?**

Ha senso poter selezionare un range di tempo più ampio (2 giorni, 1 settimana, ecc.), tuttavia il problema è che quasi sicuramente il ristorante non sarà al completo tutti i giorni, e quindi il ristorante potrebbe avere dei dati fuorvianti, riguardo agli ingredienti da acquistare, ad esempio tra 2 settimane.

Per porre rimedio a questo problema si può porre una data limite dopo la quale non è più possibile effettuare ordini o modificarli (ad esempio fino a mezzanotte del giorno prima della data di prenotazione).

E' possibile inoltre introdurre una sanzione in caso i clienti non si presentino al ristorante.

**Nel capitolato è esplicitato il fatto che il cliente possa rilasciare una recensione solo dopo aver pagato, quindi solo i clienti che hanno effettuato un pagamento possono scrivere delle recensioni?**

No, possono farlo tutti i clienti che hanno partecipato ad una prenotazione, si può ad esempio introdurre un campo all'interno della prenotazione chiamato "Stato pagamento" e solo una volta che tutte le piazze della prenotazione sono state effettivamente pagate allora tutti gli utenti facenti parte della prenotazione possono lasciare una recensione.

**Premettendo di aver deciso in un primo momento di utilizzare React per la parte di front-end e PHP come back-end, ci siamo poi accorti che sarebbe più utile andare ad utilizzare NodeJS per la parte di back-end, lei ha qualche consiglio da darci a riguardo?**

Era stato consigliato PHP come linguaggio per il back-end perché già conosciuto e utilizzato dai membri del gruppo, ad esempio nel corso di Tecnologie Web.

NodeJS (ed il framework Express<sup>G</sup>) sono molto utili perché nativamente scritti in Javascript, è inoltre molto facile fare deploy delle applicazioni ed utilizzarlo.

In più è molto ben documentato, con numerose librerie già implementate per funzionalità da noi utilizzabili (autenticazione, socket, ecc.).

Ci viene consigliato di utilizzare NestJS<sup>G</sup>, che offre più funzionalità nativamente disponibili "out of the box" rispetto ad Express, ad esempio creare in modo più facile la documentazione per le nostre API.

Tuttavia, che si usi PHP o NodeJS, l'obiettivo finale è lo stesso, ovvero andare a creare delle API REST che vadano a comunicare tramite un'interfaccia standard.

**Ha qualche novità riguardo ai seminari sulle tecnologie da utilizzare per il progetto?**

Ha chiesto a dei suoi colleghi quando e come sarà possibile effettuare dei seminari su Docker, ci farà sapere più avanti.

### **3.1 Problemi con docker**

Sono stati poi chiesti dei consigli di carattere tecnico riguardanti l'utilizzo di docker, con cui stiamo avendo dei problemi. Ci sono quindi state illustrate alcune soluzioni ai nostri problemi.

Firma del proponente



Alessandro Staffolani, *Imola Informatica*