|  |  |
| --- | --- |
| the taxi - app  Piattaforma per gestire le prenotazioni dei tassisti della zona | Progetto  Abbiamo deciso di realizzare un sito per prenotare delle corse in Taxi  Giada, Abdullah e Daniel  I3a |

**Sommario**

[1 Introduzione al progetto: 2](#_Toc134298469)

[2 Protocollo di realizzazione 2](#_Toc134298470)

[2.1 Sviluppo del progetto: 2](#_Toc134298471)

[2.2 Test e controllo di qualità: 3](#_Toc134298472)

[2.3 Approvazione e consegna: 3](#_Toc134298473)

[3 Elenco delle funzionalità 3](#_Toc134298474)

[3.1 Prenotare una corsa (clientiUC\_maps\_availability\_indicazioni.js file) 4](#_Toc134298475)

[3.2 Tassista, accettare o declinare una corsa(taxiUC\_disponibilita\_accettaCorsa\_prenotazioniFiltrate.js file) 4](#_Toc134298476)

[4 Progettazione del lavoro (Bizzarri) 5](#_Toc134298477)

[5 Considerazioni e conclusioni 5](#_Toc134298478)

# Introduzione al progetto:

Durante il terzo anno alla CPT come Sviluppatori di applicazioni, ci è stato chiesto di scegliere un gruppo e sviluppare un sito internet con delle funzionalità complesse, come integrare delle API e altri processi che non avevamo ancora avuto modo di sperimentare in classe.

Lavorare in gruppo su un progetto è sempre un'esperienza entusiasmante, soprattutto quando si collabora con compagni di classe con cui solitamente non si ha l'occasione di lavorare. Nel corso degli ultimi mesi, abbiamo avuto l'opportunità di sviluppare una piattaforma utilizzando Node.js, che ci ha permesso di creare una soluzione per riservare corse in taxi (lato utente) e di implementare anche la parte del tassista.

Sviluppare questo progetto ci ha fornito un'occasione per imparare nuove competenze e conoscenze tecniche, collaborare e condividere idee con i nostri compagni di gruppo. Ci siamo impegnati al massimo e abbiamo lavorato sodo per assicurarci che la nostra piattaforma fosse funzionale e completa.

Nella seguente relazione, descriveremo l'esperienza che abbiamo vissuto lavorando su questo progetto, evidenziando i punti di forza e le sfide incontrate, e concludendo con le nostre considerazioni finali.

# Protocollo di realizzazione

## Sviluppo del progetto:

**Definizione**: sviluppare la soluzione o il prodotto, seguendo la pianificazione stabilita e utilizzando le competenze del gruppo.

Durante le prime settimane di questo progetto, non avendo mai collaborato prima, abbiamo dovuto capire che genere di competenze erano presenti nel gruppo ma soprattutto, abbiamo definito chi sarebbe stato responsabile di quale parte.

Abdullah avendo già da due anni l’idea di questa piattaforma, ha deciso di prendersi la responsabilità insieme a Daniel del back-end, mentre io (Giada) che ho più esperienza con il front-end mi sono focalizzata sulla grafica e la documentazione.

Nelle le settimane successive abbiamo sviluppato la base dell’applicazione, come le grafiche, il numero di pagine e le loro funzionalità (ne parliamo in dettaglio nel prossimo punto).

Dopo alcuni giorni che Daniel a Abdullah stavano lavorando sulla logica, si sono imbattuti in vari problemi di sovrascrittura e lettura del JSON, inizialmente avevano scelto Ajax (è una tecnologia web che consente di aggiornare in modo asincrono e dinamico una pagina web senza la necessità di ricaricare completamente la pagina stessa) ma visto l’inesperienza con i JSON, hanno preferito convertire il progetto in Node.js con cui abbiamo avuto modo di sperimentarlo a scuola nel corso di quest’anno.

## Test e controllo di qualità:

**Definizione**: testare la soluzione o il prodotto per assicurarsi che soddisfi i requisiti del cliente e che sia privo di errori. Effettuare controlli di qualità su ogni aspetto del prodotto.

Durante il processo di sviluppo del nostro progetto, abbiamo effettuato numerosi test e controlli per garantire che la soluzione finale soddisfacesse tutti i requisiti posti dai docenti e che fosse privo di errori.

Per effettuare i test, abbiamo usato una tecnica semplice ma che si è rivelata efficacie, ogni volta che uno di noi implementava qualcosa sulla pagina, automaticamente testavamo tutte le componenti della pagina per confermare il loro buon funzionamento. Ogni volta che si rilevava un errore o una falla nel codice, abbiamo comunicato immediatamente al responsabile della pagina per apportare le necessarie modifiche e correggere gli errori.

Ogni volta che veniva segnalato un errore o un problema, il responsabile interveniva prontamente per risolvere il problema e assicurare che la soluzione fosse sempre all'altezza delle aspettative del cliente.

In sintesi, il processo di testing e controllo di qualità è stato un passaggio cruciale, che ci ha permesso di garantire che il codice finale soddisfacesse tutti i requisiti dei docenti e che fosse priva di errori. Grazie a questi test e controlli di qualità.

## Approvazione e consegna:

**Definizione**: ottenere l'approvazione del cliente e consegnare il prodotto o la soluzione.

Valutazione e miglioramento: effettuare una valutazione del progetto e identificare eventuali aree di miglioramento per progetti futuri.

Dopo settimane di lavoro intenso, siamo entusiasti di poter presentarvi il prodotto finale. Siamo molto soddisfatti del lavoro svolto e speriamo che il nostro codice sia all'altezza delle aspettative dei docenti.

Desideriamo ringraziare il nostro docente Ghiggi per averci dato l'opportunità di lavorare su un progetto diverso da quelli proposti in classe. Ci ha dato la possibilità di mettere in pratica le conoscenze acquisite durante il corso e di sviluppare un applicativo interessante

Siamo felici di poter condividere questo progetto e di ricevere i vostri feedback sul nostro lavoro.

# Elenco delle funzionalità

Abbiamo deciso di andare nel dettaglio di due file in particolare che ci hanno causato varie difficolta nel corso del progetto, il file per una nuova corsa e il file per rendere visibili le richieste ai tassisti.

Teniamo inoltre a specificare che abbiamo deciso di usare il framework bootstrap per il css.

## Prenotare una corsa (clientiUC\_maps\_availability\_indicazioni.js file)

La funzione iniziale **`initMap()`** è la funzione di inizializzazione dell'API di Google Maps, che crea una mappa e un geocodificatore. Inoltre, l'API viene configurata con un oggetto “autocomplete” che consente all'utente di cercare località in modo interattivo.

La funzione **`calcolaIndicazioni()`** viene chiamata quando l'utente cerca un'indicazione stradale tra due località. Utilizza l'oggetto `DirectionsService` dell'API di Google Maps per recuperare le indicazioni e visualizzarle sulla mappa. Inoltre, la funzione calcola la distanza tra le due località e il prezzo stimato del viaggio in taxi in base alla distanza. Queste informazioni vengono quindi visualizzate all'utente sulla pagina

La funzione **`availabilityCheck()`** viene chiamata quando l'utente fa clic sul pulsante "Controlla disponibilità". Recupera i dati degli utenti dal file `data.json`, filtra solo i tassisti che sono attualmente disponibili e li visualizza all'utente sulla pagina in modo che possa scegliere un tassista. Se non ci sono tassisti disponibili, il pulsante di prenotazione verrà disattivato.

La funzione **`prenota()`** viene chiamata quando l'utente fa clic sul pulsante "Prenota Corsa" dopo aver verificato la disponibilità del tassista. La funzione richiede all'utente di inserire il metodo di pagamento e confermare il prezzo del viaggio; quindi, crea un oggetto di prenotazione che viene quindi inviato al server per la registrazione. La funzione convertirà l'oggetto di prenotazione in una stringa JSON prima di inviarlo al server.

## Tassista, accettare o declinare una corsa(taxiUC\_disponibilita\_accettaCorsa\_prenotazioniFiltrate.js file)

Questo codice consente ai tassisti di accettare o rifiutare le richieste di viaggio dai clienti. Utilizza dei dati JSON per archiviare informazioni sui conducenti e sulle richieste di viaggio.

Quando la pagina web si carica, recupera le informazioni del conducente dalla memoria locale utilizzando *JSON.parse(localStorage.getItem('userData'))* e quindi visualizza il nome e lo stato del conducente sulla pagina. Se il conducente è disponibile, la pagina mostra un elenco di richieste di viaggio disponibili e il conducente può fare clic su un pulsante per accettare un viaggio.

Le richieste di viaggio sono archiviate in un file JSON, che il codice recupera utilizzando *fetch('/prenotazioni.json')*. Filtra quindi le richieste per mostrare solo quelle disponibili. Per ogni richiesta, trova le informazioni del cliente dal *file data.json* utilizzando il loro ID e visualizza il nome del cliente e i dettagli del viaggio in una sezione.

Il codice disabilita anche l'opzione di accettare nuove richieste di viaggio se il conducente ha già un viaggio attivo o se il suo stato è impostato su non disponibile. Quando lo stato del conducente viene modificato, il codice aggiorna la memoria locale con il nuovo stato.

# Progettazione del lavoro (Bizzarri)

Il nostro docente Bizzari ha chiesto al nostro gruppo di fornire un powerpoint dettagliato che contenga tutte le informazioni riguardanti il progetto. In particolare, ci ha richiesto di includere un diagramma di Gantt che mostri le attività da svolgere nel tempo e la loro durata, una to-do list che elenchi tutti i compiti da svolgere in ordine di priorità, e un preventivo che stimi i costi necessari per completare il progetto.

In vista della consegna, il nostro gruppo si è impegnato nella stesura di questo documento, che conterrà tutte le informazioni richieste dal docente. Inoltre, il giorno della consegna forniremo anche il powerpoint dettagliato, che presenterà il nostro progetto in modo chiaro e conciso, evidenziando i punti salienti e le funzionalità principali della nostra piattaforma. Siamo molto determinati a consegnare un lavoro di alta qualità e a dimostrare al docente la nostra dedizione e il nostro impegno nel portare a termine questo progetto.

# Considerazioni e conclusioni

In conclusione, lavorare su questo progetto è stato un'esperienza unica e stimolante. Abbiamo imparato molto riguardo la creazione di una piattaforma funzionale utilizzando node.js, collaborare con compagni di classe con cui solitamente non si lavora, e come gestire un progetto in modo efficace e organizzato.

Il nostro gruppo ha affrontato diverse sfide durante lo sviluppo della piattaforma, ma siamo riusciti a superarle grazie alla nostra determinazione e al nostro impegno. Siamo molto orgogliosi del risultato finale e crediamo che la piattaforma che abbiamo creato sia un ottimo risultato.

Una delle difficoltà maggiori che abbiamo incontrato è sicuramente stato la sovrascrittura del JSON che ci ha causato di ricominciare il progetto che era ormai quasi alla fine. Fortuna ho avuto un gruppo motivato che ha comunicato bene sin dall’inizio, dopo qualche ora Abdu era già riuscito a trovare una soluzione.

Infine, ci sentiamo di consigliare a chiunque di provare almeno una volta a lavorare in gruppo su un progetto simile, in quanto è un'esperienza che ti consente di apprendere e di crescere sia a livello personale che professionale.