



UNIVERSITÀ
DI TRENTO

Dipartimento di Ingegneria
e Scienza dell'Informazione

Ingegneria del software

REPORT FINALE

MOUNTAIN WONDERS
Gruppo G24

Anno accademico 2023/2024

Indice

1	Scopo del documento	3
2	Organizzazione del gruppo	4
3	Suddivisione delle task	5
4	Monte ore finale	6
5	Criticità	6
6	Autovalutazione finale	6
7	Commento sui seminari	7
7.1	Seminario Blue Tensor	7
7.2	Simulazione Kanban	7
7.3	Seminario IBM	7
7.4	Seminario META	7
7.5	Seminario U-Hopper	8
7.6	Seminario Fusco - OpenSource	8
7.7	Seminario Microsoft	8
7.8	Seminario Molinari	9
7.9	Seminario Marsiglia	9
7.10	Seminario APSS	9

1 Scopo del documento

Il presente documento riporta un report finale riguardo l'organizzazione del gruppo G24 durante lo sviluppo del progetto richiesto.

Verrà di conseguenza presentata in primo luogo la suddivisione delle task per poi proseguire con la tabella oraria ad indicazione del tempo utilizzato per ciascun compito.

Verranno infine elencate eventuali problemi/criticità riscontrate durante lo sviluppo e le soluzioni adottate dal team. Dopo l'analisi del progetto svolto, verrà quindi presentata una sezione di autovalutazione del lavoro.

Infine sarà presente una sezione con dei brevi commenti sui vari seminari svolti durante l'anno.

2 Organizzazione del gruppo

L'idea iniziale per questo progetto era quella di dividerci il lavoro in modo piuttosto equilibrato tra i vari membri del team.

Durante la prima parte del progetto, la modalità di lavoro è stata principalmente individuale, mentre poi nell'ultima parte c'è stato un maggior lavoro di team.

Infatti nei primi deliverable del progetto e nella relativa creazione diagrammi necessari, a ciascun membro del team è stata assegnata una sezione del documento. Così facendo è stato possibile proseguire il lavoro in parallelo, evitando quindi eventuali perdite di tempo nel caso in cui tutti e 3 i membri del gruppo avessero lavorato alla stessa sezione.

Ogni volta che un membro terminava il lavoro a lui assegnato, passava ad aiutare gli altri componenti del gruppo.

Anche nella seconda parte del progetto, durante lo sviluppo, c'è stata una prima fase di lavoro individuale per la creazione delle pagine lato frontend e delle api necessarie.

Arrivati poi alla fase di testing, si è rivelato molto utile il confronto tra i membri del team, che ha permesso di identificare e coprire tutti i casi di errori possibili ritornati dall'applicazione.

3 Suddivisione delle task

Componente team	D1	D2
Giorgia Saccon	Group leader, frontend developer, backend developer, documentazione, analyst requisiti	<p>Si è occupata di gestire il gruppo, assegnando ad ogni componente le task e i ruoli che doveva ricoprire.</p> <p>Ha avuto il ruolo principale nello sviluppo di tre delle pagine più importanti del sito web (pagina rifugi, pagina montagne e homepage), effettuando il collegamento tra backend e frontend. Ha inoltre contribuito allo sviluppo delle API e effettuato testing dell'UI dell'applicazione. Ha contribuito alla scrittura di D1, D2 e D3.</p> <p>Ha avuto un ruolo principale nella scrittura del D4, scrivendo gran parte del documento.</p> <p>Ha contribuito alla scrittura del D5.</p>
Alessio Amiri	Frontend developer, backend developer, documentazione, analyst requisiti	<p>Ha avuto il ruolo principale nello sviluppo delle API, ha sviluppato le pagine frontend di login e registrazione e contribuito a alcune funzionalità presenti nella pagina dei rifugi e nella pagina montagne.</p> <p>Ha contribuito a collegare front end e backend.</p> <p>Ha scritto e effettuato tutti i test delle API.</p> <p>Ha contribuito alla scrittura di D1, D2 e D3 e in piccola parte al D4.</p> <p>Ha contribuito alla scrittura del D5.</p>
Davide Paolazzi	Backend, documentazione, swagger	<p>Ha contribuito in parte allo sviluppo delle API, ha avuto il ruolo principale nella documentazione tramite swagger, ha collegato swagger al sito web rendendolo disponibile nella pagina /api-docs.</p> <p>Ha contribuito alla scrittura di D1, D2 e D3 e in minima parte al D4.</p>

4 Monte ore finale

Componente del gruppo	D1	D2	D3	D4	D5	Totale
Giorgia Saccon	13	28	21	54	4	121
Alessio Amiri	13	27	18	59	4	122
Davide Paolazzi	10	24	4	27	0	66
Totale	36	79	43	140	8	309

5 Criticità

Nella prima parte del progetto non abbiamo riscontrato nessuna difficoltà tecnica o pratica.

Abbiamo riscontrato alcuni problemi durante lo sviluppo del D4 a causa dell'ottimizzazione delle richieste al database fatte da Next, in particolare al fetching dei componenti (<https://stackoverflow.com/questions/76356803/data-not-updating-when-deployed-nextjs13-app-on-vercel-despite-using-cache-no>). La nostra soluzione è stata

Siamo riusciti però a trovare una soluzione più creativa, in quanto abbiamo fuso la nostra funzione "get" dei rifugi con la funzione per ottenere i rifugi di una determinata montagna.

Passando infatti il parametro 0, la funzione controllerà il numero della versione del documento, che è 0, e otterrà in modo dinamico i dati, evitando i caching.

Abbiamo inoltre avuto alcuni problemi a causa di un membro del gruppo che sfortunatamente si è ammalato durante lo sviluppo delle API e non ha potuto contribuire al meglio ai documenti D4 e D5.

6 Autovalutazione finale

Inizialmente, il gruppo ha collaborato con impegno e costanza per raggiungere l'obiettivo di sviluppo di un'applicazione web ottimale e funzionante.

Nella seconda parte della progettazione, a causa di alcune problematiche, non è stato possibile procedere come quanto fatto inizialmente. Nonostante i vari problemi, siamo riusciti ad ultimare il progetto, ritenendoci molto soddisfatti del lavoro svolto e soprattutto di quanto appreso durante lo sviluppo della web app.

Componente del gruppo	Autovalutazione
Giorgia Saccon	30
Alessio Amiri	30
Davide Paolazzi	24

7 Commento sui seminari

7.1 Seminario Blue Tensor

Concetto base in questo seminario è stata la metodologia Agile.

Il nostro gruppo concorda con il fatto che spesso questa metodologia risulti fondamentale per il raggiungimento degli obiettivi richiesti dal cliente perchè, come detto al seminario, permette di mostrare subito al cliente una prima versione del progetto finale, rendendo partecipe il cliente di ciò che viene fatto.

Un punto molto favorevole che è molto piaciuto al gruppo è il fatto che viene consentito al cliente di visionare passo passo il procedimento ed eventualmente modificare (prima che progetto sia finito) alcuni elementi, facilitando quindi il lavoro degli sviluppatori.

Questa tecnologia è stata sfruttata anche per lo sviluppo del nostro progetto, infatti alla consegna del progetto il risultato è una prima versione del prodotto, ottenuta dopo il primo sprint. Per consegnare l'applicazione ben fatta, servirebbero infatti una serie di altri sprint.

7.2 Simulazione Kanban

Un seminario interessante e tipologicamente diverso dai precedenti è stato quello dedicato alla simulazione kanban, ci ha permesso di sperimentare sulla nostra pelle l'efficientamento del lavoro rispetto ad altri metodi di lavoro, grazie ad una migliore organizzazione, divisione e collaborazione nell'esecuzione delle varie tasks.

E' stato possibile notare l'applicazione pratica di questo metodo anche all'interno del nostro team di lavoro, durante lo sviluppo di questo progetto: siamo riusciti a suddividere i lavori su diverse task, assegnandone una ad ogni membro del gruppo; una volta finite le proprie task ciascun membro ha aiutato il resto del team facendo del proprio meglio.

7.3 Seminario IBM

In questo seminario è stato mostrato come l'azienda IBM possa fornire diversi servizi per l'ingegneria del software che permettono di avere dei software di qualità superiore. Molto importante è stata anche la presentazione di sistemi per proteggere le applicazioni da eventuali attacchi informatici. Questi sistemi potrebbero essere inglobati all'interno della nostra applicazione qualora fossero possibili più sprint per lo sviluppo, in modo da evitare eventuali attacchi DDos.

7.4 Seminario META

Durante il seminario, è stata data l'opportunità di esplorare l'esperienza che si ha nel lavorare per META, concentrandosi in particolare sul loro approccio all'ingegneria del software. È sorprendente notare che Meta adotta un modello agile, ma ciò che risulta più

affascinante è la totale assenza di restrizioni sugli strumenti e i linguaggi da impiegare. Si nota inoltre come META offre molta libertà quando si parla di lavoro, processo che permette di massimizzare il lavoro lasciando la possibilità ai propri dipendenti di organizzarsi al meglio.

Abbiamo riscontrato l'applicazione di queste libertà anche all'interno del corso, dove si ha avuto la possibilità di scegliere i linguaggi e l'organizzazione del lavoro che più si preferiva, nonostante ciò il nostro gruppo ha optato per l'utilizzo dei linguaggi visti a lezione.

7.5 Seminario U-Hopper

In questo seminario una buona parte è stata dedicata alla descrizione del processo e dei tool utilizzati per lo sviluppo in un progetto, che coincide con quanto fatto dal nostro team. Viene infatti descritta la fase di sprint durante lo sviluppo, che rappresenta proprio il lavoro svolto durante questo corso, tramite 3 passi principali: sviluppo del codice, testing e revisione del codice dal punto di vista pratico. Sono stati poi descritti i tool utilizzati dall'azienda: il team durante lo sviluppo del progetto sono stati creati due branch locali, uno per lo sviluppo delle api e uno per lo sviluppo dell'interfaccia grafica per poi unire il lavoro e testare il funzionamento dell'applicazione.

7.6 Seminario Fusco - OpenSource

Questo seminario si è incentrato principalmente sul concetto di open source, che indica un modello di sviluppo software in cui il codice sorgente è accessibile, modificabile e distribuibile liberamente.

Questo favorisce la collaborazione globale e la trasparenza, permettendo agli sviluppatori di contribuire e migliorare il software. Uno dei leader del settore è Red Hat, offrendo soluzioni basate su Linux e contribuendo attivamente a progetti come Fedora e CentOS. Red Hat dimostra il successo del modello open source nell'ambito aziendale.

7.7 Seminario Microsoft

In questo seminario sono stati trattati i test: parte fondamentale dello sviluppo di un'applicazione dal momento che permettono di verificare che siano rispettati i requisiti funzionali e non.

Questa è stata l'ultima parte effettuata dal team per lo sviluppo dell'applicazione per fare un controllo su quanto fatto e descritto nei documenti precedenti. Il processo di testing dell'applicazione e delle sue funzionalità si è dimostrato estremamente essenziale per lo sviluppo complessivo del sito.

Questo approccio ha consentito di individuare errori, vulnerabilità e problematiche di sistema che sarebbero stati difficili da rilevare attraverso una semplice analisi del codice.

7.8 Seminario Molinari

In questo seminario sono stati affrontati i sistemi legacy. Questi sistemi sono una sfida per l'ingegneria del software dal momento che sono basati su vecchie tecnologie e per essere convertiti in sistemi con nuove tecnologie hanno bisogno di un lungo processo di conversione. E' necessario quindi mantenerli aggiornati, ma per fare ciò serve personale con le adeguate conoscenze. L'ingegneria del software in questo settore risulta molto importante soprattutto nel caso di grandi progetti, in cui vedere il codice risulta molto dispendioso e quindi è necessario visionare la documentazione.

7.9 Seminario Marsiglia

In questo seminario è stato presentato gli step necessari per lo sviluppo di un software, dalla pianificazione al mantenimento. Questi step sono gli stessi che sono stati affrontati durante lo sviluppo del nostro progetto: dall'idea iniziale siamo passati all'analisi, allo sviluppo e al testing finale. Se fosse un'applicazione da mettere sul mercato, il nostro progetto richiederebbe ovviamente anche la parte di mantenimento necessaria per mantenere l'applicazione funzionante e soddisfare i clienti tramite la metodologia agile, già affrontata in altri seminari precedenti.

7.10 Seminario APSS

In questo seminario è stato presentato il sistema informativo dell'azienda sanitaria trentina e le tecnologie utilizzare per gestire la grande quantità di lavoro, rivedendo diverse tecnologie conosciute tramite seminari seguiti in precedenza. E' stato affrontato per esempio il tema della migrazione dei servizi sull' cloud. Infatti stanno migrando la maggior parte dei propri servizi su cloud, si trovano ancora a lavorare con sistemi legacy che vanno mantenuti e aggiornati e utilizzano strumenti di AI per migliorare le prestazioni sanitarie rivolte ai pazienti. Possiamo quindi affermare che questo seminario conclusivo ci ha confermato ancora una volta come tutte le tecnologie e i processi presentatici nel corso delle lezioni di ingegneria del software siano parte fondamentale per il sano funzionamento di qualunque azienda: dalla piccola start-up alla grande pubblica amministrazione.