

**TEMPLATE
PROJECT WORK**

Corso di Studio	INFORMATICA PER LE AZIENDE DIGITALI (L-31)
Dimensione dell'elaborato	Minimo 6.000 – Massimo 10.000 parole (<i>pari a circa Minimo 12 – Massimo 20 pagine</i>)
Formato del file da caricare in piattaforma	PDF
Nome e Cognome	Andrea Isceri
Numero di matricola	0312201297
Tema n. (Indicare il numero del tema scelto):	1
Titolo del tema (Indicare il titolo del tema scelto):	La digitalizzazione dell'impresa
Traccia del PW n. (Indicare il numero della traccia scelta):	8
Titolo della traccia (Indicare il titolo della traccia scelta):	Sviluppo di una dashboard in JavaScript per l'analisi delle prestazioni aziendali nel settore primario
Titolo dell'elaborato (Attribuire un titolo al proprio elaborato progettuale):	Dashboard in JavaScript: Analisi delle performance aziendali nel settore primario

PARTE PRIMA – DESCRIZIONE DEL PROCESSO

Utilizzo delle conoscenze e abilità derivate dal percorso di studio

(Descrivere quali conoscenze e abilità apprese durante il percorso di studio sono state utilizzate per la redazione dell'elaborato, facendo eventualmente riferimento agli insegnamenti che hanno contribuito a maturarle):

Il mio Project Work ha l'obiettivo di mettere in luce le abilità e le conoscenze che ho acquisito durante il corso di Informatica per le Aziende Digitali, focalizzandomi su aree importanti come "Tecnologie web", "Reti di calcolatori e sicurezza informatica", "Programmazione 2" e "Ingegneria", che sono fondamentali per affrontare e risolvere il tema trattato.

Durante il modulo di "Tecnologie web", ho imparato:

- Introduzione a web 2.0;
- Comunicazione su internet;
- Servizi e protocolli;
- Strutture OSI e TCP/IP;
- Dispositivi di rete;
- CISCO Packet Tracer - Server Web;
- Protocollo http;

- Laboratorio su http;
- Replit - Ambiente di sviluppo online;
- Cookie e attività pratiche;
- Architettura Client-Server.

Dal corso di "Reti di calcolatori e sicurezza informatica", ho appreso:

- Nozioni di computer in rete;
- Connessioni internet;
- Trasmissione di dati su internet;
- Efficienza delle reti di computer;
- Protocolli ISO/OSI e TCP/IP;
- Sicurezza informatica;
- E-mail sicura e uso di PGP;
- Protocolli SSL, TLS e HTTPS.

Nel modulo di "Programmazione 2", ho sviluppato abilità per:

- Scrivere programmi in linguaggi di programmazione orientati agli oggetti;
- Gestire e mantenere software;
- Analizzare dati e Big Data usando Python;
- Competenze in Python, inclusi funzioni, librerie statistiche e opzioni grafiche.

Il corso permette di utilizzare librerie per elaborare dati tradizionali e grandi volumi di dati.

Grazie al corso di "Ingegneria", ho acquisito:

- Principi dell'ingegneria software;
- Ciclo di vita del software e modelli;
- Gestione di progetti informatici;
- Introduzione a UML;
- Diagrammi d'uso, sequenza e macchine a stati.

Questo percorso aiuta a realizzare un progetto di dashboard interattiva pensato per il settore primario, affrontando le sfide nella gestione delle risorse naturali e nella pianificazione delle operazioni.

Utilizzando JavaScript e tecnologie web, si intende supportare le aziende di questo settore in contesti in continua evoluzione.

Il mio lavoro ha l'obiettivo di mettere in luce le abilità e le conoscenze che ho acquisito durante il corso di Informatica per le Aziende Digitali, focalizzandomi su aree importanti come "Tecnologie web", "Reti di calcolatori e sicurezza informatica", "Programmazione 2" e "Ingegneria", che sono fondamentali per affrontare e risolvere il tema trattato.

Fasi di lavoro e relativi tempi di implementazione per la predisposizione dell'elaborato

(Descrivere le attività svolte in corrispondenza di ciascuna fase di redazione dell'elaborato. Indicare il tempo dedicato alla realizzazione di ciascuna fase, le difficoltà incontrate e come sono state superate):

Le fasi del progetto si sono svolte come segue:

- Comprendere l'argomento e gli obiettivi necessari.
- Cercare online informazioni utili su siti come Google Scholar e PubMed.
- Iniziare a scrivere il documento.

Le attività lavorative suddivise per giorni:

1. Esaminare il compito per organizzare il lavoro. (5 giorni).
2. Pianificare e definire gli obiettivi principali. (4 giorni).
3. Cercare materiali utili su internet. (5 giorni).
4. Scrivere la prima bozza del progetto. (15 giorni).
5. Controllare e completare il lavoro. (2 giorni).
6. Revisione finale del documento per correggere eventuali errori. (1 giorno).

Risorse e strumenti impiegati

(Descrivere quali risorse - bibliografia, banche dati, ecc. - e strumenti - software, modelli teorici, ecc. - sono stati individuati ed utilizzati per la redazione dell'elaborato. Descrivere, inoltre, i motivi che hanno orientato la scelta delle risorse e degli strumenti, la modalità di individuazione e reperimento delle risorse e degli strumenti, le eventuali difficoltà affrontate nell'individuazione e nell'utilizzo di risorse e strumenti ed il modo in cui sono state superate):

Le risorse utilizzate per la realizzazione del mio progetto sono:

- ResearchGate;
- Pubmed;
- EBSCO;
- Scribbr;
- Google Scholar;
- Academia.edu.

La bibliografia utilizzata per la stesura del progetto è la seguente:

- Cagnazzo, F. (2018). *Progettazione e sviluppo di una dashboard per l'analisi della qualità dell'aria nel contesto urbano. Caso di studio: la città di Milano* (Doctoral dissertation, Politecnico di Torino).
- Rolando, M. (2017). *Studio e sviluppo di un cruscotto per il monitoraggio dell'inquinamento urbano = Design and development of a dashboard for*

- monitoring urban pollution* (Doctoral dissertation, Politecnico di Torino).
- Murtezan, E. Sviluppo di API REST in ambito IoT e di una webapp per il controllo qualità.
 - Faienza, A. Progettazione ed Implementazione di una Dashboard di Location Intelligence in Ambito Retail.
 - GATTO, M. Progettazione e sviluppo di un'interfaccia grafica per un sistema di raccomandazione di istanze di processi aziendali.
 - BRACEROS, J. V. Analisi, Progettazione e Implementazione di Nuove Funzionalità in un'applicazione Web per la Schedulazione e Rendicontazione delle Attività dei Team di Sviluppo Aziendale.
 - D'Amilo, S. (2021). *Progetto e implementazione di un'architettura innovativa per la gestione e la visualizzazione dei dati IoT di una filiera produttiva= Design and implementation of an innovative architecture for the IoT data management and visualization of a production chain* (Doctoral dissertation, Politecnico di Torino).
 - Segalini, S. (2020). *Progetto e implementazione di un cruscotto interattivo a supporto dell'analisi dinamica di una supply chain= Design and implementation of an interactive dashboard in support of the dynamic analysis of a supply chain* (Doctoral dissertation, Politecnico di Torino).
 - DOLZAN, M. Una web app per la gestione di dati industriali.
 - Raffa, G. (2019). *Progettazione e sviluppo di un'architettura scalabile per il calcolo dei Key Performance Indicators (KPI)= Design and development of a scalable architecture to compute Key Performance Indicators (KPI)* (Doctoral dissertation, Politecnico di Torino).
 - Sartore, U. Reingegnerizzazione di processi aziendali e sviluppo delle relative applicazioni di supporto mediante Google Apps.
 - Andreeva, J., Dzhunov, I., Karavakis, E., Kokoszkiewicz, L., Nowotka, M., Saiz, P., & Tuckett, D. (2012, December). Designing and developing portable large-scale JavaScript web applications within the Experiment Dashboard framework. In *Journal of Physics: Conference Series* (Vol. 396, No. 5, p. 052069). IOP Publishing.
 - Dabbas, E. (2021). *Interactive Dashboards and Data Apps with Plotly and Dash: Harness the power of a fully-fledged frontend web framework in Python—no JavaScript required*. Packt Publishing Ltd.
 - Le, D. (2019). Developing a User Management Dashboard with Fullstack Javascript.
 - Gaspar, H. M. (2017). JavaScript applied to maritime design and engineering. In *Proceedings of the 16th Conference on Computer and IT Applications in the Maritime Industries* (pp. 253-269).
 - Morton-Owens, E. G., & Hanson, K. L. (2012). Trends at a glance: A management dashboard of library statistics. *Information Technology and Libraries*, 31(3), 36-51.
 - Pangemanan, T. G. V., Lumenta, A. S., & Rindengan, Y. D. (2023). Desain Dashboard Keuangan dan Informasi di GMIM Wilayah Manado Barat Daya: Financial and Information Dashboard Design in GMIM Southwest Manado Region. *Jurnal Teknik Informatika*, 18(4), 143-152.
 - Christian, C., & Voutama, A. (2024). Implementasi Aplikasi Antrian

Pencucian Mobil Berbasis Web Menggunakan Php, Javascript, Html, Css Dan Uml. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(2), 2243-2248.

- Brown, J., Pirrung, M., & McCue, L. A. (2017). FQC Dashboard integrates FastQC results into a web-based, interactive, and extensible FASTQ quality control tool. *Bioinformatics*, 33(19), 3137-3139.
- Opach, T., & Rød, J. K. (2018). Developing a dashboard visualizing compensation data on damages caused by extreme events.

Ragioni per le quali sono state scelte le seguenti fonti bibliografiche:

Le fonti sono state scelte per la loro rilevanza e importanza per essere adeguate all'argomento, identificate tramite un'accurata ricerca su internet.

Problemi riscontrati e strategie adottate:

La difficoltà principale è stata posizionare correttamente le fonti nel contesto appropriato.

Questo problema è stato superato grazie a un'analisi approfondita e alla comprensione dei materiali, permettendo di collocare le fonti in modo corretto.

PARTE SECONDA – PREDISPOSIZIONE DELL'ELABORATO

Obiettivi del progetto

(Descrivere gli obiettivi raggiunti dall'elaborato, indicando in che modo esso risponde a quanto richiesto dalla traccia):

Gli obiettivi chiave del mio Project Work sono:

- L'unione e l'uso di varie fonti di dati per creare una dashboard interattiva, che possa elaborare e mostrare informazioni in modo chiaro e utile per i dirigenti.
- L'utilizzo di tecnologie moderne per il frontend, come React o Vue.js, con lo scopo di realizzare un'interfaccia utente efficace e reattiva, capace di mostrare dati in tempo reale e permettere interazioni su misura in base ai bisogni degli utenti.
- L'uso di principi di visualizzazione dei dati e dell'esperienza dell'utente, per assicurarsi che la dashboard sia facile da usare e da capire anche per chi non possiede competenze tecniche.

Contestualizzazione

(Descrivere il contesto teorico e quello applicativo dell'elaborato realizzato):

Una dashboard o anche detta cruscotto digitale serve come un pannello principale per mostrare le informazioni fondamentali sul funzionamento di un software.

La capacità di un cruscotto di comunicare efficacemente dipende in gran parte dalla qualità del suo design. Deve presentare i dati in modo chiaro e semplice per aiutare nella comprensione dei KPI (Indicatori Chiave di Prestazione) del sistema.

Le informazioni sono di solito mostrate tramite tabelle, grafici e vari widget, privilegiando numeri e immagini piuttosto che testo.

Vue.js, comunemente conosciuto come Vue, è un framework JavaScript open-source

progettato con il modello Model-View-ViewModel per creare interfacce utente e applicazioni single-page. Il modello architetturale MVVM separa l'applicazione in tre componenti distinte. Il Model consente l'accesso ai dati, la View presenta l'interfaccia dell'applicazione, mentre il ViewModel agisce da intermediario tra i due: i dati vengono ricevuti, elaborati e poi inviati alla View.

Vue.js, che si integra di base con Laravel, ha un'architettura incrementale che si concentra sul rendering dichiarativo e sulla composizione di componenti.

Sono disponibili funzionalità avanzate necessarie per applicazioni complesse, come il routing, la gestione dello stato e strumenti di compilazione, grazie a librerie e pacchetti ufficiali di supporto.

I componenti rappresentano una delle parti più interessanti di Vue.

Permettono di creare applicazioni di medie e grandi dimensioni, organizzando il codice in moduli chiari e riutilizzabili. Ogni componente include una porzione di HTML, arricchita da logica applicativa e, se necessario, da regole CSS specifiche.

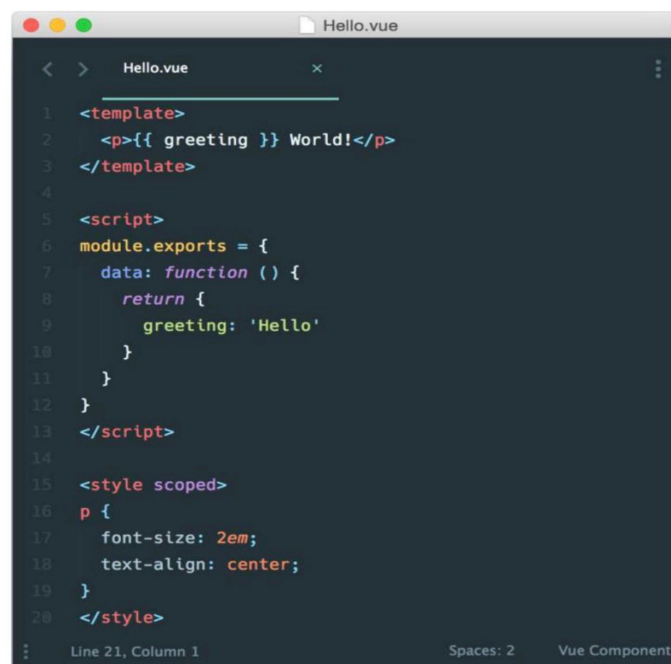
I componenti si dividono in:

- Struttura HTML: il disegno del componente.
- Stili CSS: che stabiliscono come verrà visualizzato il markup.
- Codice JavaScript: per gestire dati e funzionalità.

Questi elementi possono essere separati in file diversi o uniti in un unico file con estensione.vue.

Un esempio di componente si trova nel file HelloWorld.vue, mostrato nella Figura 1.

Il codice è diviso in `<template>` per il codice HTML, `<script>` per quello JavaScript e `<style>` per il CSS.



```
< > Hello.vue x
1 <template>
2   <p>{{ greeting }} World!</p>
3 </template>
4
5 <script>
6 module.exports = {
7   data: function () {
8     return {
9       greeting: 'Hello'
10    }
11  }
12 }
13 </script>
14
15 <style scoped>
16 p {
17   font-size: 2em;
18   text-align: center;
19 }
20 </style>
⋮ Line 21, Column 1 Spaces: 2 Vue Component
```

Figura 1 Struttura componente Vue

Una volta che si sviluppa un componente, è necessario registrarlo con Vue.

Questo processo può essere eseguito in due maniere:

- Registrazione globale dei componenti (Figura 2): all'interno del file `resources/js/app.js`. I componenti globali possono essere utilizzati ovunque nell'app Vue.
- Registrazione locale dei componenti (Figura 3): questi sono riservati all'uso specifico nel componente principale in cui vengono incorporati.

```
Vue.component('mio-componente', {  
  [...] //elenco opzioni  
});
```

Figura 2. Registrazione mondiale di un elemento.

```
new Vue({  
  [...],  
  components: {  
    'mio-componente': {  
      [...] //elenco opzioni  
    }  
  }  
});
```

Figura 3. Registrazione locale di un elemento.

Gli sviluppatori di Vue forniscono delle indicazioni considerate come pratiche migliori, che aiutano a realizzare componenti non solo funzionali nel breve periodo, ma anche versatili per diverse situazioni.

Questo semplifica la costruzione di interfacce standard di alta qualità e favorisce la diffusione di componenti open-source.

Secondo Vue, esistono tre modalità per interagire con componenti realizzati da altri:

- Props: utilizzate per inviare dati al componente;
- Eventi: consentono al componente di influenzare l'ambiente esterno;
- Slot: permettono all'ambiente esterno di cambiare il contenuto del componente.

Se impiegati nel modo giusto, questi strumenti amplificano le potenzialità dei componenti, contribuendo all'implementazione richiesta.

I componenti possono essere registrati sia a livello globale che locale, e la loro inizializzazione offre varie opzioni.

Nel progetto è adottato il modello Single File Component, che richiede file con

estensione.vue, organizzati come mostrato in Figura 1.

Le direttive in Vue.js sono attributi HTML particolari, contraddistinti dal prefisso v-, che aggiungono comportamenti alla visualizzazione.

Molte direttive necessitano di argomenti quando il nome non spiega il loro uso, e aggiungono ulteriori attributi.

Ad esempio, v-bind ha bisogno del nome dell'attributo HTML da collegare.

Gli attributi seguono la sintassi:

v-{nomedirettiva}:{nomeargomento}.

Un altro esempio è v-on, che collega una funzione agli eventi creati dall'utente.

In questo caso, l'attributo indica l'evento, come v-on:click per associare una funzione al clic.

È possibile anche modificare le direttive mediante i modificatori, attributi specifici che alterano il comportamento.

I modificatori seguono la stessa sintassi: *v-{nomedirettiva}:{nomeargomento}*, e vengono scritti dopo gli attributi, se entrambi sono presenti.

Ecco un elenco breve delle direttive principali di Vue:

- v-bind: per collegare i dati dell'oggetto data ai tag HTML che si aggiornano dinamicamente.
- v-for: mostra una lista di elementi di un array.
- v-if, v-else, v-else-if: controllano la visualizzazione di tag o blocchi di codice in base a una condizione.
- v-show: funziona in modo simile a v-if, ma il tag rimane nel DOM, non visibile a causa della proprietà CSS "display:none".
- v-on: attiva un metodo del componente dopo un evento specifico, come con v-on: click="miaFunzione", eseguito quando si clicca.
- v-model: utilizzato nei form o nei componenti, stabilisce un collegamento bidirezionale tra i dati e il template.

La direttiva v-model consente di cambiare il valore delle proprietà dell'oggetto in base all'input dell'utente, modificando anche l'output visualizzato.

Descrizione dei principali aspetti progettuali

(Sviluppare l'elaborato richiesto dalla traccia prescelta):

Il mio project work si dividerà in due fasi progettuali:

- 1. Che cos'è una Dashboard?**
- 2. Descrizione del contesto in cui opera l'azienda;**
- 3. Dashboard con risultati dei dati random.**

1. Il termine inglese dashboard si traduce in italiano come "pannello di controllo" o "cruscotto". Nel campo dell'informatica, le dashboard sono rappresentazioni grafiche destinate agli utenti, composte da vari elementi visivi utili per visualizzare dati o amministrare sistemi. In altre parole, offrono una panoramica di informazioni e grafici che indicano le prestazioni, ad esempio di

un'azienda. Nel settore tecnologico, diversi sviluppatori di software utilizzano la parola dashboard.

Alcuni esempi famosi includono:

- Dashboard su Mac OS X: gestisce le impostazioni del sistema operativo e dispone i widget.
- Google Dashboard: mostra i dati legati a un profilo Google.
- Dashboard in CMS come WordPress: pagina iniziale che aiuta nella gestione dei siti web.
- Le imprese utilizzano spesso le dashboard per il monitoraggio.

Il loro scopo è mostrare in modo chiaro indicatori chiave e KPI di vari tipi. Queste informazioni aiutano a prendere decisioni aziendali.

Ecco alcune applicazioni pratiche:

- Monitoraggio di dati finanziari importanti (come il flusso di cassa).
- Confronto tra valori pianificati e reali (come margini di profitto e fatturato).
- Rappresentazione degli indicatori di magazzino (come tempi di spedizione e copertura).
- Controllo della produzione (ad esempio, consumo di energia ed efficienza delle macchine).

Le dashboard sono create da sistemi avanzati come ERP, Business Intelligence e nuove metodologie di archiviazione dati legate ai Big Data.

Un grande vantaggio delle dashboard moderne è che gli utenti possono personalizzare le loro interfacce.

È possibile utilizzare vari elementi grafici come semafori, tachimetri e grafici.

2. L'azienda GreenHarvest, fondata da Luca Rossi nel 2010, ha le sue radici in una tradizione agricola familiare che risale a tre generazioni. La passione per la terra e l'innovazione ha spinto Luca a trasformare l'azienda in un leader nel settore primario, incorporando tecnologie avanzate per ottimizzare la produzione e ridurre l'impatto ambientale.

Situata nella fertile pianura padana, GreenHarvest si dedica alla coltivazione di grano, mais e ortaggi, mirati a soddisfare le esigenze di un mercato in espansione che privilegia prodotti di qualità e sostenibili. Circa il 60% delle sue terre è destinato a cereali, il 25% agli ortaggi, mentre il restante viene utilizzato per sperimentazioni innovative con coltivazioni meno tradizionali.

Il cuore dell'azienda è il sistema di monitoraggio ambientale, che Luca ha introdotto dopo essersi ispirato ai metodi di precision farming utilizzati in paesi nordici. Utilizzando algoritmi per generare dati casuali sulle condizioni climatiche, GreenHarvest monitora variabili cruciali come temperatura, umidità e piovosità. Questi dati sono confrontati con dati di produzione: quantità di raccolto e tempi di

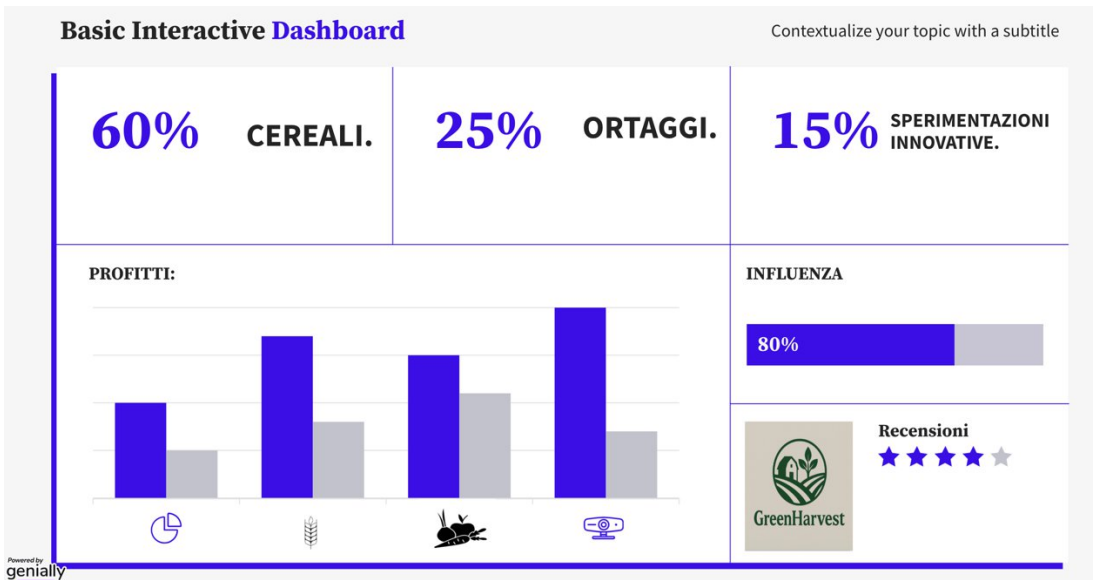
crescita specifici per ogni coltura.

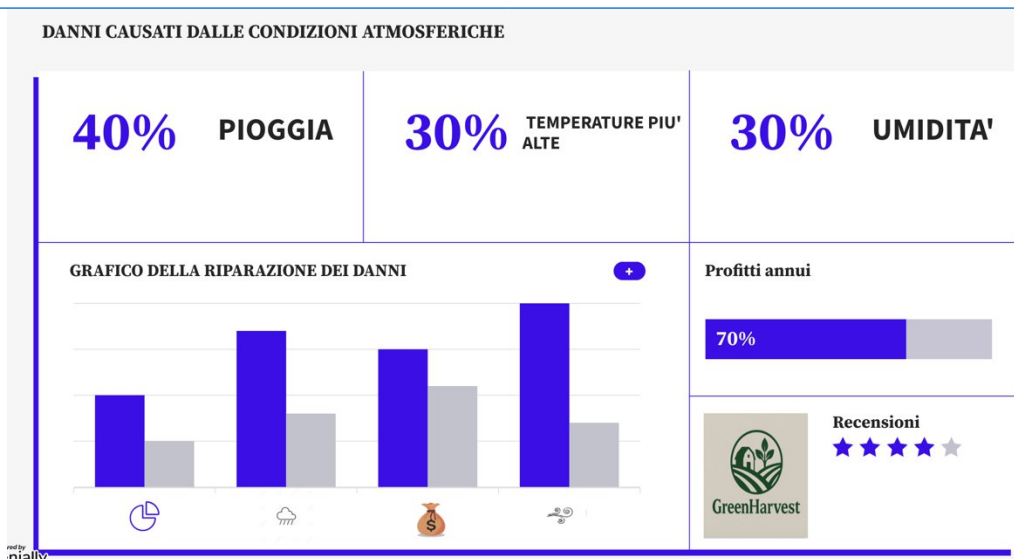
L'obiettivo è adattare le tecniche agricole in tempo reale, garantendo decisioni informate con un tasso di successo del 85% nel miglioramento del rendimento rispetto agli anni precedenti. L'azienda si vanta di essere in grado di incrementare il raccolto di grano del 20% durante le stagioni più favorevoli, grazie alle modifiche strategiche determinate dalle simulazioni dei dati ambientali.

GreenHarvest non è solo un esempio di efficacia nel settore agricolo, ma rappresenta anche un affresco di come le tradizioni e l'innovazione possano coesistere. Luca Rossi continua a perseguire la sua visione: una fusione armoniosa tra tecnologia e amore per la terra, garantendo che l'eredità della famiglia Rossi prosperi in un mondo che cambia rapidamente.

Durante gli ultimi anni, GreenHarvest ha dovuto adattarsi alle mutevoli condizioni meteorologiche della pianura padana. L'azienda ha osservato un aumento del 30% della temperatura estiva, con le piogge primaverili che sono diventate del 40% più intense, seppur sporadiche con il 30% di umidità in più. Questi cambiamenti climatici impongono una sfida costante per Luca Rossi e la sua squadra, che devono adattare le pratiche agricole in risposta a questi estremi. La variabilità della temperatura e l'incertezza delle piogge mettono in evidenza l'importanza del sistema di monitoraggio ambientale di GreenHarvest, consentendo all'azienda di ottimizzare l'irrigazione e di prevenire danni causati da eventi meteorologici imprevisti. Questo approccio proattivo non solo protegge le colture ma garantisce anche che l'azienda mantenga un'elevata produttività.

3. Dashboard con risultati dei dati random.





La traccia del mio Project Work chiedeva di creare una Dashboard Interattiva con dei dati random del contesto in cui opera l'azienda, di seguito il link:

<https://view.genially.com/6865c3dcf2767d869c6a0882/interactive-content-basic-interactive-dashboard>

Campi di applicazione

(Descrivere gli ambiti di applicazione dell'elaborato progettuale e i vantaggi derivanti della sua applicazione):

I settori che ho esplorato nel mio progetto includono:

Agricoltura di precisione: In questo settore, si utilizzano informazioni e analisi del tempo per prevedere le condizioni difficili, migliorando così i tempi per piantare e raccogliere.

Gestione delle risorse: Comporta una gestione efficace delle risorse tramite approcci sostenibili che siano anche vantaggiosi dal punto di vista economico, utilizzando strumenti come cruscotti per supportare le decisioni.

Adattamento delle colture: Si fonda sull'adeguamento ai cambiamenti climatici per migliorare la salute del suolo e aumentare la produzione dei raccolti.

Vantaggi:

- Promuovere la sostenibilità ambientale.
- Possibilità di diventare un modello per altre attività locali.
- Maggiore adozione di tecnologie innovative nel processo agricolo.
- Supporto da parte dell'azienda alla crescita economica regionale, creando nuove opportunità di lavoro per la comunità locale.

Valutazione dei risultati

(Descrivere le potenzialità e i limiti ai quali i risultati dell'elaborato sono potenzialmente esposti):

Le capacità del mio progetto mostrano come l'agricoltura di precisione possa migliorare la sostenibilità e l'efficienza nelle operazioni.

L'analisi svolta su una fattoria nella Pianura Padana serve come esempio pratico per mostrare come l'uso di nuove tecnologie abbia aumentato del 30% le entrate e abbia migliorato del 30% la precisione nelle pratiche agricole.

Di conseguenza, mettere in risalto le applicazioni tecnologiche può favorire la ricerca e lo sviluppo nel settore agricolo fino al 40%.

Le possibili limitazioni includono:

- Studio limitato a un solo caso.
- Variazioni costanti delle condizioni climatiche.
- Difficoltà di trasferire il progetto ad altre aree d'Italia.