Capitolato scelto 9: Vimar  
  
Punti a favore:

1. **Dominio Applicativo Concreto e Socialmente Utile:** Il contesto delle "residenze protette" fornisce uno scopo nobile e tangibile. Sviluppare un sistema che può migliorare la sicurezza e il benessere degli anziani è un forte motivatore e rende il progetto molto più significativo rispetto a un dominio astratto.
2. Tecnologie Moderne e Ricercate

3.  Supporto Aziendale Eccezionale e Materiale Fornito: Vimar fornisce un supporto strutturato con incontri bisettimanali/settimanali (SAL) e, aspetto cruciale, fornisce un kit hardware fisico per testare con dispositivi reali. Questo riduce enormemente il gap tra teoria e pratica e permette di affrontare problematiche reali di integrazione IoT.

4.  Supporto Aziendale Eccezionale e Materiale Fornito: Vimar fornisce un supporto strutturato con incontri bisettimanali/settimanali (SAL) e, aspetto cruciale, fornisce un kit hardware fisico per testare con dispositivi reali. Questo riduce enormemente il gap tra teoria e pratica e permette di affrontare problematiche reali di integrazione IoT.

Punti contro:

* 1.      Elevata Complessità e Ampio Scope: La completezza del progetto è anche la sua principale sfida. Il team deve essere bravo a gestire la complessità e a priorizzare le funzionalità obbligatorie, per non perdersi negli optional. Una buona pianificazione iniziale è fondamentale.
* 2.      Elevata Complessità e Ampio Scope: La completezza del progetto è anche la sua principale sfida. Il team deve essere bravo a gestire la complessità e a priorizzare le funzionalità obbligatorie, per non perdersi negli optional. Una buona pianificazione iniziale è fondamentale.

Capitolato 1  
  
Punti a favore:

* Requisiti Ben Definiti: I requisiti obbligatori e opzionali sono elencati in modo chiaro e strutturato. Questo riduce l'ambiguità e fornisce un'ottima checklist per la pianificazione e la verifica finale del progetto.
* Caso Studio Concreto: La presenza di un caso studio specifico (la macchina da caffè connessa via Wi-Fi) fornisce un dominio applicativo tangibile per testare le funzionalità, evitando di lavorare in modo troppo astratto.
* Supporto Aziendale Strutturato: Bluewind si impegna esplicitamente per un supporto "ibrido" (online e in presenza), con incontri periodici. Questo è un enorme vantaggio, poiché fornisce un canale diretto per chiarire dubbi, ottenere feedback e allinearsi con le aspettative dell'azienda.

Punti contro:

* 1.      Complessità del Dominio: Il dominio normativo (EN 18031, direttiva RED) è intrinsecamente complesso. Comprendere appieno la logica dei "Decision Tree" e le interdipendenze tra i requisiti richiederà uno sforzo iniziale significativo di analisi e studio.

Capitolato 2  
  
Punti a favore:

* Tema Innovativo e ad Alto Potenziale: L'uso di un'architettura multi-agente per l'analisi automatizzata del codice è estremamente attuale e all'avanguardia.
* Dominio Concreto e Utile: La piattaforma risolve problemi reali di qualità del codice, sicurezza e manutenzione dei repository.
* Sessione di mentoring sulle tecnologie che verranno utilizzate

Punti contro:

* 1.      Alta Complessità Concettuale: L'architettura multi-agente è concettualmente avanzata. Progettare un sistema dove agenti specializzati comunicano efficacemente attraverso un orchestratore richiede una solida comprensione di pattern complessi.
* 2.      Requisiti di Testing Stringenti: La richiesta del 70% di test coverage (obbligatorio) è apprezzabile professionalmente ma può essere impegnativa da raggiungere in un progetto universitario, specialmente per componenti AI.
* 3.   Non sembra esserci flessibilità nella scelta delle tecnologie da utilizzare

Capitolato 3

Punti a favore:

* Dominio Specializzato e di Alto Valore: La conservazione digitale è un campo di nicchia ma cruciale, specialmente in ambito legale e amministrativo.
* Problema Concreto e Ben Definito: L'esigenza di accedere a documenti conservati digitalmente in modalità offline è un requisito reale per molti professionisti. Il progetto risolve un problema tangibile.

Punti contro:

* 1.      **Complessità del Dominio Normativo**: La conservazione digitale è regolata da standard e normative complesse. Comprendere appieno il formato dei DIP e i requisiti di compliance richiederà uno sforzo iniziale significativo.
* 2.      **Sfide Tecniche per l'Offline**: Implementare ricerche efficienti e visualizzazioni di anteprima completamente offline, specialmente per grandi volumi di dati, presenta sfide non banali di performance e gestione della memoria.
* 3.      **Ambiguità Architetturale**: Non è completamente chiaro se l'applicazione debba essere una PWA, un'app desktop (Electron) o entrambe. Questa decisione avrà impatti significativi sull'architettura.
* 4.      **Scope Potenzialmente Ampio**: Le funzionalità opzionali come la ricerca semantica (con FAISS) e la verifica delle firme digitali sono progetti ambiziosi che potrebbero distrarre dallo sviluppo del MVP.
* 5.      **Testing su Grandi Volumi**: Garantire le performance con "grandi volumi" di dati potrebbe essere difficile da testare e validare in ambiente universitario.

Capitolato 4  
  
Punti a favore:

1. Impatto Sociale Elevatissimo: Il progetto ha uno scopo nobile e concretamente utile - prevenire e supportare vittime di violenza di genere. Sviluppare un'app che può potenzialmente salvare vite fornisce una motivazione etica molto forte.
2. Supporto Aziendale Eccezionale e Strutturato: Miriade offre un supporto completo:

o  Referenti specializzati per ogni area (tecnica, design, dominio sociale)

o  Formazione iniziale sulla tematica della violenza di genere

o  Strumenti professionali (Jira, Bitbucket)

o  Supporto multidisciplinare continuo

o  Possibilità di incontri in sede

3.  Requisiti Chiari e Dettagliati: Sia i requisiti funzionali che non funzionali sono ben specificati, con una chiara distinzione tra obbligatori e opzionali.

Punti contro:

1.  Alta Complessità Tecnica e Progettuale:

o L'integrazione di AI/ML per l'analisi comportamentale richiede competenze specialistiche

o L'architettura serverless su AWS con microservizi è complessa da gestire

o La sicurezza dei dati è critica e richiede implementazioni robuste

2.  Responsabilità e Sensibilità del Dominio: L'errore in un'app di questo tipo può avere conseguenze gravi. La progettazione deve essere impeccabile sotto il profilo della sicurezza e dell'affidabilità.

3.  Scope Molto Ampio: Le funzionalità previste sono numerose e ambiziose (rilevamento AI, allarmi silenziosi, diario criptato, moduli educativi, community). Il rischio di sovra-estendere il progetto è alto.

Capitolato 5: Nexum  
  
Punti a favore:

* 1.  Prodotto Reale e Integrazione con Piattaforma Esistente: NEXUM è una piattaforma HR già operativa. Sviluppare moduli che si integreranno in un prodotto commerciale fornisce un'esperienza di lavoro su codice legacy e integrazione con sistemi esistenti.
* 2. Utilizzare e sperimentare metodologie Agile in un team aziendale.
* 3. Tecnologie definite esaustivamente e  requisiti del progetto chiari, con due requisiti opzionali definiti.

Punti contro:

* 1.  NEXUM è una piattaforma già esistente e operativa, lo sviluppo di nuovi moduli comporta un livello di rischio elevato. L’integrazione in un prodotto reale, destinato a un mercato effettivo e a clienti concreti, implica vincoli progettuali e tecnologici particolarmente rigidi, dettati dalle esigenze e dagli standard aziendali. Questo può limitare la flessibilità del team e aumentare la complessità delle attività di sviluppo e validazione
* 2. Alta complessità riguardante le tecnologie da utilizzare con AWS, OCR (Optical Character Recognition), e integrazione dell’AI.

Capitolato 6: Second Brain  
  
Punti a favore:

1.      L'integrazione degli LLM in un'applicazione di produttività personale (note-taking) è un campo molto interessante. Lavorare su questo progetto fornisce un'esperienza diretta con una tecnologia all'avanguardia.

2.      Ampio Spazio per la Progettazione e la Ricerca:

Il capitolato non specifica esattamente come implementare le funzionalità, ma si concentra sul cosa. Questo lascia al team un'enorme libertà nelle scelte architetturali, nella progettazione dell'interfaccia utente (UI/UX) e, soprattutto, nell'ingegnerizzazione dei prompt per l'LLM, che è il cuore della "ricerca" del progetto.

3.      Zucchetti si offre esplicitamente di supportare il team nelle parti più complesse (es. configurazione API LLM) e metterà a disposizione gli LLM stessi. Questo riduce un potenziale grande ostacolo.

Punti contro:

 1.  Complessità e Ampiezza del Dominio:

Integrare un LLM in un'applicazione è un compito non banale. La gestione degli stati, degli errori delle API, dei costi (se si usano modelli cloud) e della latenza può essere complessa e richiedere molto tempo per la sola fase di prototipazione.

2.  Dipendenza da Tecnologie Esterne (LLM):

L'intera applicazione dipende dalla stabilità, disponibilità e costi del servizio LLM scelto. Un cambiamento nelle API o problemi di disponibilità del servizio potrebbero bloccare lo sviluppo o il funzionamento dell'app.

3. Le tecnologie non risultano chiaramente definite e i requisiti di progetto sono troppo ampi. Inoltre, l’assenza di specifiche vincolanti e di requisiti opzionali chiaramente delineati comporta un’eccessiva libertà decisionale, che può generare incertezza nella pianificazione e aumentare il rischio di deviazioni dagli obiettivi progettuali.

4. L’azienda propone lo sviluppo di un’applicazione web basata su HTML per la parte front-end. Tuttavia, il team di progetto ritiene che una soluzione fondata esclusivamente su HTML non sia tecnicamente adeguata, soprattutto considerando la disponibilità di framework moderni e più performanti, quali *Angular* o *React*, che garantirebbero maggiore scalabilità, manutenibilità e qualità dell’esperienza utente

Capitolato 7  
  
Punti a favore:

1.      Il progetto affronta una sfida attuale e molto concreta nel campo IoT: acquisire, gestire e rendere fruibili dati da sensori in modo sicuro, scalabile e multi-tenant. Questo fornisce un'esperienza su problematiche industriali reali, non solo accademiche.

Punti contro:

1.  Progettare e implementare un'architettura distribuita, sicura e multi-tenant è una sfida tecnica notevole. La gestione della concorrenza, della consistenza dei dati, della comunicazione asincrona e della tolleranza ai guasti richiede competenze solide e un'attenta progettazione.

2.  I requisiti sono molti e articolati (dalle API on-demand e streaming, alla UI, al monitoring, alla sicurezza). C'è il concreto rischio di dover sacrificare la profondità di alcune funzionalità per coprirne la quantità entro i tempi del progetto.

3.  Testare un'architettura distribuita è intrinsecamente difficile. Configurare ambienti di test end-to-end, simulare guasti e testare la scalabilità richiederà sforzi notevoli e una buona pianificazione.

Capitolato 8  
  
Punti a favore:

* 1.  L'integrazione di NLP, Computer Vision e Speech-to-Text in un'unica pipeline è un campo di ricerca e sviluppo estremamente attuale e complesso. Offre un'esperienza di apprendimento su tecnologie AI d'avanguardia.
* 2.  Viene fornito un ricco elenco di tecnologie suggerite per ogni componente (LLM, OCR, Speech-to-Text, UI), ma con la libertà di scegliere alternative. Questo permette al team di adattare lo stack tecnologico alle proprie competenze.

Punti contro:

* 1.  Integrare e far collaborare modelli di AI diversi (LLM, Vision, Audio) in una pipeline coerente è una sfida tecnica notevolissima.
* 2.  Il team dovrebbe padroneggiare o imparare rapidamente una vasta gamma di tecnologie AI e di framework, ciascuna con le sue complessità.
* 3.  Il pre-processing di dati multimodali (pulizia del testo, elaborazione di immagini, trascrizione audio) è un compito laborioso e critico. La qualità dell'output finale dipende fortemente da questa fase.
* 4.  I modelli di AI, specialmente gli LLM, possono essere imprevedibili. Validare l'accuratezza e l'affidabilità del sistema in scenari reali, dove un errore può significare un ordine sbagliato, è una sfida significativa.