**Scialpi - Funzionamento generale + requisiti**

Salve, noi siamo i CodeBusters e ci occupiamo del progetto HDViz.

***(click)***

questo tratta di una web app che si propone l’idea di visualizzare dati a molte dimensioni. In un tipico utilizzo del programma vediamo l'utente che deve poter caricare dei dati nel sistema, e potrà decidere se farlo attraverso l’invio di un file CSV o tramite interrogazioni ad un database.

Successivamente potrà scegliere uno tra i vari algoritmi per la riduzione dimensionale dei dati o per il calcolo delle distanze, ed eventuali parametri associati ad esso. L’utente deve poi poter decidere quali siano le dimensioni da visualizzare tra quelle già esistenti originariamente e quelle ottenute dalla sintesi di più dimensioni.

L’utente arrivato a questo punto può decidere una delle cinque diverse visualizzazioni di dati offerte, e ne può personalizzare la visualizzazione a seconda delle proprie preferenze.

Non essendo presente un sistema di autenticazione, l’utente, per interrompere una sessione di lavoro e ripristinarla in un secondo momento, ha la possibilità di generare e scaricare un file in formato JSON che conterrà i dati e le preferenze di visualizzazione.

***(click)***

Ci eravamo lasciati alla scorsa revisione con un prodotto che rendeva l’idea ma che sicuramente non era ben definito. Durante questo periodo è stata introdotta un'architettura più chiara, nuove visualizzazioni e funzionalità, ovvero il caricamento dei dati da database e 4 nuove tipologie di grafico.

Tutti i requisiti obbligatori richiesti sono già stati implementati e questo ci permette di dedicare più tempo agli aspetti qualitativi del prodotto e ai requisiti facoltativi rimanenti in modo da soddisfare in pieno le esigenze del proponente. Alcuni requisiti dovranno essere testati completamente, ma ci sentiamo soddisfatti di quanto è stato fatto fino ad ora

***(click)***

**PB**

L’architettura che abbiamo implementato ha avuto origine da una product baseline che ha subito diverse modifiche durante questo periodo, in seguito ad alcuni problemi rilevati durante il colloquio con il docente Cardin.

Un primo problema riguardava il pattern architetturale implementato. Abbiamo deciso di applicare il pattern MVVM e, basandoci sulle tecnologie individuate durante la fase della TB, di apportare alcune modifiche allo schema usuale. Questa scelta però ha portato ad un’architettura difficilmente mantenibile e che inevitabilmente avrebbe introdotto difetti con il proseguire delle attività di codifica.

Un secondo problema riguardava invece la gestione dello STRATEGY pattern che abbiamo applicato al processo di riduzione dimensionale.

Il prof. Cardin ha deciso di sospendere per qualche giorno l’avanzamento delle attività, in modo da sanare i problemi sollevati durante il colloquio.

***(click)***

Nei giorni a seguire abbiamo collaborato intensivamente per individuare delle possibili soluzioni, creando diverse demo da condividere tra di noi e per comprendere meglio la fattibilità nell’applicarle al prodotto.

Questa pausa ci ha permesso di capire come sfruttare al meglio le tecnologie che abbiamo utilizzato e a perseguire le best practices. Abbiamo sicuramente capito gli errori effettuati e tratto beneficio da questo imprevisto, introducendo sostanziali modifiche basate sul single responsibility principle. Anche il docente Cardin ha apprezzato i cambiamenti introdotti e al secondo colloquio ha deciso di approvare le scelte architetturali.

***(click)***

**Safdari Verifica**

Per quanto riguarda la parte di verifica del prodotto, ci siamo trovati molto bene con il sistema di continuous integration stabilito prima di iniziare con la codifica. Attraverso l’utilizzo di Jest e React Testing Library, abbiamo prodotto una suite di test che viene eseguita in automatico da GitHub ad ogni commit.

Come visibile dal grafico in basso a sinistra, la percentuale di code coverage che abbiamo raggiunto è positiva e rispetta la soglia limite che ci eravamo imposti. Tuttavia confidiamo di aggiungerne ancora per verificare appieno il corretto funzionamento di tutte le funzionalità.

***(click)***

**DEMO**

* guida introduttiva
* caricamento dati default (focus su alert)
* riduzione dimensionale (focus su errore nome già utilizzato)
* SPM
* caricamento dati db -> penguins ridotto (focus su alert fallito)
* matrice distanze
* AM, FF
* caricamento dati da file -> demoHM
* HM
* caricamento dati da file -> swiss roll
* PLMA

***(click)***

**Problemi/Cambiamenti affrontati**

**Slide 1)** Durante il periodo RP-RQ abbiamo quindi raggiunto buoni risultati, ma abbiamo anche incontrato alcune problematiche che hanno rallentato lo sviluppo, sia dal punto di vista della stesura della documentazione, sia per la codifica del prodotto.

Riguardo a quest’ultima i problemi si sono incentrati nella realizzazione delle varie visualizzazioni. Queste infatti inizialmente sono state codificate utilizzando elementi HTML **svg,** ma una volta completate e quindi testate con dataset sempre più pesanti, si è notato che la loro renderizzazione richiedeva troppo tempo, causando lag e rallentamenti in tutta l’app.

Si è così deciso di migrare ai **canvas**, ossia delle vere e proprie tavole da disegno in HTML in cui è possibile in generale disegnarci e quindi anche costruirci dei grafici, in modo molto più ottimizzato.

Questa migrazione ha richiesto alcuni giorni di lavoro e ha reso la gestione degli eventi più complessa, ma dati i notevoli vantaggi dal punto di vista prestazionale abbiamo comunque deciso di perseguire questo cambiamento per tutte le visualizzazioni che forniamo.

***(click)***

**Slide 2)** Per quanto riguarda la documentazione c’è stata una ristrutturazione sia della pianificazione che del consuntivo nel Piano di Progetto, che ha comportato l’utilizzo di ulteriori ore di lavoro.

Abbiamo infatti eliminato l’incremento 9, le quali attività sono già state completate (durante l’incremento 6), in meno ore rispetto a quelle preventivate. Abbiamo quindi aggiunto le ore di lavoro guadagnate ai successivi incrementi per avere più tempo da dedicare all’implementazione delle ultime funzionalità e alla realizzazione dei test, così da prepare al meglio il prodotto per la fase di validazione e collaudo .

***(click)***

**CONSUNTIVO - Pirolo**

Come si può notare dal consuntivo del periodo appena trascorso, sono risultate necessarie meno ore totali rispetto a quelle preventivate.

Più precisamente nel **3° incremento** la quantità di codice prodotto non è stata significativa e di conseguenza non sono stati necessari molti interventi da parte dei verificatori.

Nel **4° incremento**, invece, sono state riscontrate varie difficoltà nel capire come suddividere la logica delle componenti grafiche e nell’integrarle l’architettura scelta, causando un aumento delle ore da programmatore; nel **5° e nel 6° incremento** però questa figura ne ha usate di meno, sia per l'esperienza maturata con le tecnologie di sviluppo sia perché, come spiegato in precedenza, l'implementazione degli aiuti all’utente e la guida introduttiva hanno richiesto molto meno tempo di quello che pensavamo.

Infine, visto il lavoro aggiuntivo dovuto all’esito della product baseline, nell’**ultimo incremento** sono state utilizzate tutte le ore pianificate.

Sommando questi scostamenti siamo riusciti a portare il numero totale di ore lavorative da 714 a 708 e di conseguenza il costo preventivato è diminuito, con un risparmio di circa 100 euro.

***(click)***

Come ultimo resoconto di questo periodo possiamo sicuramente menzionare gli incontri che abbiamo tenuto con il nostro proponente, che è stato contattato per mostrare lo stato del prodotto e per concordare i test di accettazione, fondamentali nel momento del collaudo.

Il dott. Piccoli si è mostrato molto entusiasta del lavoro che abbiamo svolto e delle librerie che abbiamo individuato ed integrato nel prodotto, e questo ha instaurato un clima di positività e d’impegno nel continuare in questo modo.

***(click)***

**PREVISIONI FUTURE**

Il nostro obiettivo ora è quello di presentare il prodotto completo e **collaudato il 17 maggio**. Avendo già superato tutti i requisiti obbligatori riteniamo che tale scadenza sia sufficiente per poter dedicarci a migliorare significativamente la qualità del software (tenendo conto anche degli impegni di tutti i componenti del gruppo che, nel mese di maggio, incrementano in relazione all’avvicinamento della sessione d’esami e di un nostro componente che inizierà il tirocinio).

***(click)***

**PROGRAMMAZIONE FUTURA**

Nel periodo in corso stiamo **sviluppando l’incremento VIII** che prevede di inserire la funzionalità di **esportazione e importazione** di una sessione di lavoro, salvando i dati necessari in un file JSON. Il tempo in eccesso verrà dedicato per realizzare ulteriori requisiti concordati con il proponente, come l’inserimento di UMAP, un nuovo algoritmo di riduzione dimensionale, requisito ottenuto nell’ultimo colloquio con il dottor Piccoli. (Se tale periodo non dovesse essere sufficiente per sviluppare i requisiti opzionali correttamente eviteremo di introdurre modifiche nell’applicazione. Pensiamo sia meglio avere meno funzionalità ma testate e di qualità piuttosto di aggiungerne frettolosamente di nuove con il rischio che il funzionamento del prodotto stesso possa essere inficiato.) Successivamente avvieremo la fase di **validazione e collaudo** in cui verranno incrementati i documenti, nel caso lo dovessimo ritenere necessario, e soprattutto verrà incrementata la suite di test con l’obiettivo di perseguire un giusto livello di qualità.

Con questo è tutto