Progetto:

**iSpesa**

Titolo del documento:

**Report Finale**

Informazioni Documento

|  |  |  |  |
| --- | --- | --- | --- |
| Nome Documento | *iSpesa\_Report\_Finale\_D5* | Numero  Documento | D5 |
| Descrizione | Report Finale del progetto: organizzazione del lavoro, ruoli, tempo complessivo e di ciascun membro dedicato al progetto, criticità, autovalutazione | | |

Indice

[Scopo del documento 3](#_Toc1)

[1. Approcci all’Ingegneria del Software 4](#_Toc2)

[2. Organizzazione del Lavoro 5](#_Toc3)

[3. Ruoli e Attività 6](#_Toc4)

[4. Carico e distribuzione del lavoro 6](#_Toc5)

[5. Criticità 7](#_Toc6)

[6. Autovalutazione 7](#_Toc7)

# Scopo del documento

Il seguente documento è un report finale che testimonia come il nostro team ha lavorato allo sviluppo del progetto.

Dopo aver descritto gli approcci all’Ingegneria del Software, verrà descritta l’organizzazione e la suddivisione del lavoro tra i vari membri del team, dopodiché mostreremo le ore di lavoro complessive ed individuali per i vari deliverables.

Successivamente elencheremo le eventuali criticità e problematiche riscontrate durante il corso del progetto per finire poi con una breve autovalutazione del nostro lavoro.

# 1. Approcci all’Ingegneria del Software

## 1.1 BlueTensor

Durante il seminario BlueTensor ci è stato mostrato un approcio allo sviluppo di programmi IA composto da un misto di sistema a cascata e agile. In questo approccio il progetot viene diviso in cinque parti: studio di fattibilità preliminare, una parte di progettazione software e hardware con identificazione dei requisiti funzionali e non, una parte di sviluppo e test, il rilascio finale. Le prima tre parti, per via della loro natura intrinseca, prevedono un approccio a cascata che ha come risultato la realizzazione di una progetto per l’architettura del programma ed anche due set di dati completi, uno di test ed uno di addestramento. Successivamente, i processi di sviluppo, addestramento e test seguono un modello agile in cui ogni parte del programma viene divisa in moduli che implementano funzionalità base. Progressivamente ogni modulo viene completato, integrato con gli altri e rilasciato per mostrare l’andamento dello sviluppo al cleinte ed interfacciarsi con lo stesso. Dopo questo sviluppo continuo, infine, si ha il rilascio vero e proprio del software che non sompre rappresenta la fine dello sviluppo, poiché possono essere necessari aggiornamenti o integrazioni in futuro. A mio avviso questo approccio è molto pragmatito e permette di sfruttare le migliori qualità sia del metodo agile sia del metodo a cascata. Non noto criticità poiché reputo che la parte di sviluppo a cascata sia necessaria, siccome si tratta di processi che sono sequenziali uno all’altro, mentre la parte di sviluppo agile sia il modo migliore per sviluppare il software interfacciandosi allo stesso tempo col cliente.

# 2. Organizzazione del Lavoro

Il lavoro è stato organizzato mediante una suddivisione per ruoli, assegnati in base alle competenze individuali. L’idea di partenza è stata quella di ripartire equamente in base alle competenze di ciascun individuo il carico di lavoro senza sovraccaricare alcun membro. Sin dall’inizio abbiamo diviso il lavoro in parti a cui ognuno di noi avrebbe singolarmente lavorato per poi unirle. Nel far ciò abbiamo stabilito dei ruoli inizialmente informali che poi si sono solidificati all’interno dell’organizzazione.

Nel far ciò abbiamo sia lavorato in gruppo sia singolarmente, il tutto in base alle necessità del Deliverables da elaborale. E’ essenziale il fatto che ogni qual volta che si dovesse completare un Deliverable, unendo i vari pezzi ci siamo incontrati per discutere del lavoro fatto e fare una revisione generale del lavoro del gruppo. Ciò ha permesso di trovare numero problemi e di migliorare così il tutto.

Gli incontri che abbiamo fatto sono sempre stati in presenza e molto frequenti. Abbiamo sfruttato il fatto di vivere nel medesimo appartamente per incontrarci ogni due giorni circa per parlare o in ogni caso in qualunque momento fosse necessario. Ciò ha permesso di avere aggiornamente costanti e frequenti sullo stato del lavoro di ciascuno di noi,

Per la strumentazione ci siamo affidati a Microsoft Word per la stesura dei Deliverables, VisualStudio Code per lo sviluppo del codice, PhPMyAdmin e MySQLWorkbench per il lavoro sul database del sito e GitHub per salvare i vari file e mantenere uno storico. Per ogni i vari diagrammi abbiamo utilizzato Draw.io che abbiamo preferito a LucidChart poiché sapevamo già usarlo, mentre per i mockup del Deliverable 1 abbiamo deciso di realizzarli direttamente in HTML, accellerando così lo sviluppo del front-end del sito web. Per segnare le ore abbiamo utilizzato dei file salvati sul git in cui semplicemente scrivevamo la data come nome del file e le ore di lavoro e l’attività svolta all’interno degli stessi.

# 3. Ruoli e Attività

Abbiamo riassunto nella seguente tabella la suddivisione dei ruoli all’interno del team.

|  |  |  |
| --- | --- | --- |
| Componente del team | Ruolo | Principali attività |
| Davide Colosimo | Capo progetto, analista requisiti funzionali e non con relative specifiche, architetto di sistema. | Il ruolo principale è stato la gestione del progetto e la coordinazione delle attività. Si è occupato dell’impaginazione dei documenti e della loro rifinitura. Ha contirbuito attivamente a tutti i deliverables e in maniare principale al D2 e al D3. |
| Gabriele Lanaro | Analista requisiti funzionali e non, progettista mockup, full stack developer. | Ha attivamente contribuito a tutti i deliverables, specialmente nella stesura dei mockup durante il D1 e nello sviluppo del front-end e del back-end del sito web durante il D4. |
| Ismael Checkri Belghiti | Analista requisiti funzionali e non, sviluppatore del database, architetto di sistema. | Ha contribuito attivamente a ciascun deliverables, con particolare attenziale ai D2 e D3, grazie ai diagrammi realizzati. Ha contribuito attivamente anche al D4 mediante la reazione dei database necessari per il sistema |

# 4. Carico e distribuzione del lavoro

La seguente tabella riporta la distribuzione delle ore di lavoro.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nome** | **D1** | **D2** | **D3** | **D4** | **D5** | **Totale** |
| Davide Colosimo | 16.15 | 24 | 23.30 | 2 | 2.30 | 68.15 |
| Gabriele Lanaro | 3 | 7.30 | 3 | 13 | 1 | 27.30 |
| Ismael Checkri Belghiti | 8 | 15.30 | 10 | 6.30 | 1 | 41 |
| **Totale** | 27.15 | 47 | 36.30 | 21.30 | 4.30 | **137.15** |

# 5. Criticità

Durante il corso di tutto il progetto abbiamo incontrato un unico grande problema, gli esempi dei vari deliverables. Essendo incompleti e poco chiari, più che indicazione ci hanno sviato dal percorso corretto a nostro avviso portandoci a rifare parte del lavoro in un tempo successivo. Tranne questo increscioso problema (che ha portanto indietro il progresso del progetto di molti giorni poichè abbiamo rifatto completamente D2 e D3), non reputiamo ci siano stati problemi durante il corso del progetto. Ad esclusione di ciò che è stato riferito prima, potremmo annoverare tra i problemi alcune incomprensioni sullo sviluppo del back-end del progetto, ma non riteniamo esse problemi per via della loro superfluità.

# 6. Autovalutazione

Nel complesso riteniamo di aver lavorato tutti molte ore al progetto, dimostrando sia quantità che qualità nel nostro lavoro.

La nostra autovalutazione è la seguente:

|  |  |
| --- | --- |
|  | **Voto** |
| Davide Colosimo | 30 |
| Gabriele Lanaro | 30 |
| Ismael Checkri Belghiti | cotone |