Marco Perico 08/03/2024

FILM FOR YOU

Marco Perico 08/03/2024

Introduzione:

La passione per il cinema e l'entusiasmo per la tecnologia hanno ispirato Marco Perico nel suo progetto "Film For You". Creato con l'obiettivo di fornire agli utenti una piattaforma interattiva per esplorare e valutare film, questo sito web rappresenta un'opportunità unica per condividere opinioni, scoprire nuovi titoli e arricchire l'esperienza cinematografica di tutti gli appassionati.

La seguente relazione offre un'analisi dettagliata del progetto "Film For You", esplorando gli obiettivi, i concetti teorici applicati, l'implementazione delle funzionalità chiave come il CRUD, l'utilizzo di UML, la gestione dei rischi e le lezioni apprese durante lo sviluppo del progetto. Attraverso un'esplorazione approfondita di queste aree, si può apprezzare l'impegno di Marco nel creare un'esperienza utente coinvolgente e soddisfacente nel mondo del cinema online.

Link:

<http://pericomarco5ie.altervista.org/capolavoro/login/login.php>

pag 1

Marco Perico 08/03/2024

**Indice:**

1 Obiettivi del Progetto

2 Cenni Teorici

2.1 API

2.1.1 Interoperabilità

2.1.2 Riutilizzo del Codice

2.1.3 Modularità

2.1.4 Scalabilità

2.1.5 Sicurezza

2.2 REST

2.2.1 Risorse

2.2.2 URI

2.2.3 Metodi HTTP

2.2.4 Rappresentazioni

2.2.5 Statelessness

2.2.6 interfacce Uniformi

2.3 CRUD

2.3.1 Create

2.3.2 Read

2.3.3 Update

2.3.4 Delete

2.4 Matrice dei Rischi

2.4.1 Standardizzazione

2.4.2 Versatilità

2.4.3 Tipi di diagrammi

2.4.4 Comunicazione efficace

2.4.5 Supporto degli strumenti

2.5 matrice di rischi

2.6 Work Breakdown Structure (WBS)

2.7 Lezioni Apprese

2.8 linguaggi

2.8.1 JavaScript

2.8.2 PHP

2.8.3 HTML

2.8.4 CSS

2.8.5 SQL

2.9 strutture

2.9.1 Database relazionali

2.9.2 File JSON:

3 Implementazione del CRUD

3.1 Create

3.2 Read pag 2

Marco Perico 08/03/2024

3.3 Update

3.4 Delete

3.5 Miglioramento del Contenuto

4 Utilizzo di UML

5 Struttura della WBS

6 matrice dei rischi:

6.1 Mancanza di collaborazione

6.2 Abbandono da parte di un componente del gruppo

6.3 Problemi di salute

6.4 Malfunzionamento dei sistemi informatici

6.5 Difficoltà di utilizzo degli strumenti

6.6 Difficoltà nello sviluppo delle componenti del progetto

6.7 Mancanza di esperienza dell’ambito del progetto

6.8 Modifiche situazioni ambientali

6.9 Perdita dei dati

6.10 Rivalutazione del budget

7 materiale utilizzato

7. 1 linguaggi

7.1.1 JavaScript

7.1.2 PHP

7.1.3 HTML

7.1.4 CSS

7.1.5 SQL

7.2 strutture

7.2.1 Database relazionali

7.2.2 File JSON:

8 Lezioni Apprese

9 Conclusioni

1 Obiettivi:

* riuscire a completare il progetto
* riuscire a collegare la maggior parte delle materie
* esporre il progetto in modo tecnico e approfondito

2 Cenni teorici:

**2.1 Le API (Application Programming Interface)** rivestono un ruolo cruciale nella creazione di applicazioni moderne, consentendo la comunicazione e l'interazione tra servizi, applicazioni e componenti software diversi. Esse agevolano la cooperazione tra varie parti del sistema, garantendo la compatibilità e l'efficace scambio di informazioni.

pag 3

Marco Perico 08/03/2024

* **2.1.1 Interoperabilità**: Le API permettono a sistemi e tecnologie eterogenei di comunicare tra loro, valorizzando l'interoperabilità e la coesistenza anche tra linguaggi di programmazione e piattaforme differenti.
* **2.1.2 Riutilizzo del Codice**: Grazie alle API, gli sviluppatori possono accedere a funzionalità esistenti senza dover riscrivere il codice da zero. Questo comporta un notevole risparmio di tempo e risorse.
* **2.1.3 Modularità**: Le API consentono la creazione di applicazioni modulari, dove le varie parti del software interagiscono tra loro attraverso interfacce ben definite. Questo approccio semplifica lo sviluppo, il testing e la manutenzione del software
* **2.1.4 Scalabilità**: Utilizzando le API, le funzionalità del software possono essere distribuite su più server o servizi, consentendo di adeguare l'infrastruttura alle necessità di carico in modo dinamico.
* **2.1.5 Sicurezza**: Le API possono essere utilizzate per definire controlli di accesso e autenticazione, garantendo la protezione delle risorse e dei dati. Questo assicura che solo gli utenti autorizzati possano accedere alle funzionalità offerte dalle API.

**2.2 REST (Representational State Transfer)** è un'architettura software che si basa su principi chiari per la progettazione e l'implementazione di servizi web. I suoi concetti fondamentali includono:

* **2.2.1** **Risorse**: Qualsiasi entità identificabile tramite un URI è considerata una risorsa. Questo può essere un dato, un file o qualsiasi altra entità con un'identità univoca.
* **2.2.2 URI**: Gli URI sono identificatori uniformi delle risorse, utilizzati per accedere e manipolare le risorse tramite il web. Ogni risorsa dovrebbe avere un URI univoco per identificarla in modo univoco.
* **2.2.3 Metodi HTTP**: REST si basa sui metodi standard HTTP per definire le operazioni eseguibili sulle risorse, come GET, POST, PUT, PATCH e DELETE.
* **2.2.4** **Rappresentazioni:** Le risorse REST possono essere rappresentate in diversi formati, come JSON o XML, e scambiate tra client e server tramite richieste e risposte HTTP.
* **2.2.5 Statelessness**: REST è stateless, il che significa che ogni richiesta deve contenere tutte le informazioni necessarie per comprendere e soddisfare, semplificando la scalabilità e l'affidabilità dei servizi.

pag 4

Marco Perico 08/03/2024

* **2.2.6 Interfacce Uniformi**: REST promuove l'uso di interfacce uniformi per favorire l'interoperabilità tra client e server, come metodi HTTP standard, URI ben strutturati e rappresentazioni dei dati comuni.
* **2.3 Il CRUD (Create, Read, Update, Delete)** è un'operazione di base utilizzata nel contesto dei database e delle applicazioni software per gestire le entità o i dati. Ecco una spiegazione tecnica del CRUD:
* **2.3.1 Create (Creazione):** Questa operazione consiste nel creare nuovi record o entità all'interno di un sistema. In un database relazionale, ad esempio, potrebbe essere eseguita attraverso l'uso di un'istruzione SQL di tipo INSERT, che aggiunge una nuova riga a una tabella.
* **2.3.2 Read (Lettura)**: Questa operazione consiste nel recuperare i dati o le informazioni esistenti da un sistema. In un database, viene eseguita attraverso l'uso di un'istruzione di tipo SELECT, che estrae dati specifici da una tabella o un insieme di tabelle.
* **2.3.3 Update (Aggiornamento)**: Questa operazione consiste nell'aggiornare i dati o le informazioni esistenti all'interno di un sistema. In un database, può essere eseguita attraverso l'uso di un'istruzione di tipo UPDATE, che modifica i valori di una o più colonne in una tabella.
* **2.3.4 Delete (Eliminazione)**: Questa operazione consiste nell'eliminare i dati o le informazioni esistenti da un sistema. In un database, viene eseguita attraverso l'uso di un'istruzione di tipo DELETE, che rimuove una o più righe da una tabella.

In pratica, il CRUD rappresenta le operazioni di base che un'applicazione può eseguire sui dati, consentendo agli utenti di creare, leggere, aggiornare ed eliminare le informazioni all'interno del sistema. Queste operazioni sono fondamentali per la gestione dei dati in molte applicazioni software e sono spesso implementate attraverso interfacce utente intuitive e funzionalità di back end complesse.

**2.4 UML (Unified Modeling Language)** è un linguaggio di modellazione grafica utilizzato principalmente nel campo dell'ingegneria del software per descrivere, progettare e documentare sistemi software complessi. È uno standard industriale ampiamente utilizzato, sviluppato per offrire una metodologia comune per la visualizzazione e la comunicazione dei concetti legati allo sviluppo del software.

pag 5

Marco Perico 08/03/2024Ecco alcuni punti chiave su UML:

* **2.4.1 Standardizzazione**: UML fornisce una serie di simboli grafici e regole di notazione standardizzate per rappresentare concetti come classi, oggetti, relazioni, comportamenti e strutture di un sistema software. Questo standardizzazione favorisce la comprensione e la collaborazione tra gli sviluppatori.
* **2.4.2 Versatilità**: UML può essere utilizzato in diverse fasi del ciclo di vita dello sviluppo del software, dalle fasi di analisi e progettazione fino alla documentazione e al testing. È in grado di modellare sia aspetti concettuali e astratti, come i requisiti del sistema, che aspetti più tecnici e implementativi, come l'architettura del software e il design delle classi.
* **2.4.3 Tipi di diagrammi**: UML offre una vasta gamma di diagrammi, ognuno dei quali si concentra su aspetti specifici del sistema. Alcuni esempi includono il diagramma dei casi d'uso, il diagramma delle classi, il diagramma delle sequenze, il diagramma degli stati e molti altri. Ogni tipo di diagramma fornisce una prospettiva diversa sul sistema e può essere utilizzato per comunicare informazioni specifiche agli stakeholder.
* **2.4.4 Comunicazione efficace**: Uno dei principali vantaggi di UML è la sua capacità di facilitare la comunicazione tra membri del team, stakeholder e altre parti interessate. Utilizzando diagrammi UML, gli sviluppatori possono rappresentare in modo chiaro e conciso i concetti e le relazioni all'interno del sistema, riducendo il rischio di ambiguità e fraintendimenti.
* **2.4.5 Supporto degli strumenti**: Esistono numerosi strumenti software dedicati alla creazione e alla gestione dei diagrammi UML. Questi strumenti offrono funzionalità avanzate per la creazione, la modifica, la condivisione e la collaborazione sui diagrammi UML, semplificando il processo di modellazione e documentazione dei sistemi software.

**2.5 MATRICE DEI RISCHIç:** La matrice dei rischi è uno strumento essenziale per identificare, valutare e gestire i rischi all'interno di un progetto. Consiste in una tabella che elenca i rischi potenziali in una colonna e valuta la probabilità e l'impatto di ciascun rischio in altre colonne.Nella tabella, ogni rischio è valutato per probabilità e impatto. Il peso è il prodotto di probabilità e impatto, mentre la valutazione rappresenta l'importanza del rischio. L'azione indica quali azioni possono essere intraprese per mitigare o gestire il rischio. La valutazione pesata è la somma dei pesi delle azioni correlate. Questa valutazione pesata può aiutare a classificare e prioritizzare i rischi in base alla loro gravità complessiva. Assicurati di adattare la matrice dei rischi alle specifiche del tuo progetto e di aggiornarla regolarmente durante il suo svolgimento. pag 6

Marco Perico 08/03/2024**2.6 La WBS**, acronimo di Work Breakdown Structure, è una tecnica di gestione dei progetti utilizzata per scomporre un progetto complesso in componenti più piccole e gestibili, chiamate attività. La struttura della WBS si presenta come un diagramma gerarchico che mostra la suddivisione del lavoro necessario per completare un progetto. Ogni livello inferiore della struttura rappresenta una decomposizione più dettagliata del livello superiore, fino a raggiungere il livello delle attività specifiche che possono essere facilmente

assegnate, pianificate e monitorate. La WBS aiuta a definire chiaramente i compiti del progetto, a organizzare il lavoro in modo logico e a stabilire una base per la stima dei costi e delle risorse necessarie.

**2.7 Le lezioni apprese (lesson learned)** rappresentano un concetto fondamentale nella gestione dei progetti e consistono nell'analizzare e documentare le esperienze, sia positive che negative, fatte durante lo svolgimento di un progetto al fine di migliorare le prestazioni future.Ecco alcuni punti chiave relativi alle lezioni apprese:

1. Durante e alla fine di un progetto, è importante identificare le lezioni apprese attraverso l'analisi delle esperienze, dei successi e dei fallimenti.
2. Documentazione: Le lezioni apprese dovrebbero essere documentate in modo accurato e completo, includendo i dettagli del contesto, delle azioni intraprese e dei risultati ottenuti.
3. Condivisione: Le lezioni apprese devono essere condivise all'interno dell'organizzazione coinvolta nel progetto e, se possibile, anche con altre organizzazioni o team che potrebbero trarne beneficio.
4. Implementazione: È importante non solo identificare e documentare le lezioni apprese, ma anche implementare le azioni correttive o migliorative necessarie per integrare queste lezioni nei processi futuri
5. Cultura di apprendimento: Promuovere una cultura di apprendimento continuo all'interno dell'organizzazione, dove le lezioni apprese sono
6. valorizzate e utilizzate per migliorare le prestazioni future dei progetti.

**2.8 linguaggi**

**2.8.1 JavaScript:** Linguaggio di programmazione ampiamente utilizzato per aggiungere interattività e dinamicità alle pagine web. Grazie alla sua capacità di controllare il comportamento dei caratteri, può rendere le pagine più dinamiche e coinvolgenti per gli utenti. pag 7

Marco Perico 08/03/2024 **2.8.2 PHP**: Linguaggio di scripting lato server utilizzato principalmente

per sviluppare applicazioni web dinamiche e interattive. Con PHP, è

possibile interagire con il database e gestire le operazioni sul lato

server, come l'autenticazione degli utenti e la gestione dei dati.

**2.8.3 HTML**: Acronimo di HyperText Markup Language, è il linguaggio standard utilizzato per creare e strutturare le pagine web. HTML definisce il contenuto e la disposizione degli elementi all'interno di una pagina, consentendo agli sviluppatori di definire la struttura di base di un sito web.

**2.8.4 CSS**: Acronimo di Cascading Style Sheets, è un linguaggio utilizzato per definire lo stile e la formattazione delle pagine HTML. Con CSS, è possibile modificare l'aspetto visivo di un sito web, definendo colori, dimensioni, layout e altri attributi estetici.

**2.8.5 SQL**: Acronimo di Structured Query Language, è un linguaggio di programmazione utilizzato per interrogare e gestire i database relazionali. Con SQL, è possibile memorizzare, recuperare, aggiornare e cancellare dati all'interno di un database, consentendo agli sviluppatori di creare applicazioni web dinamiche e basate sui dati.

**2.9 Struttura e memorizzazione dei contenuti:**

**2.9.1 Database relazionali**: I database relazionali sono sistemi di gestione dei database che memorizzano i dati in tabelle correlate tra loro. Utilizzano il concetto di relazioni tra le tabelle per rappresentare e gestire i dati in modo strutturato e organizzato.

**2.9.2 File JSON:** JSON, acronimo di JavaScript Object Notation, è un formato leggero e facilmente leggibile per lo scambio di dati tra un server e un client web. È basato sulla sintassi di JavaScript ed è ampiamente utilizzato per memorizzare e trasmettere dati strutturati, come ad esempio le risposte alle richieste AJAX o i dati di configurazione delle applicazioni.

pag 8

Marco Perico 08/03/202

**3 CRUD**:

**3.1 Create :**

Nel nostro sito sarà possibile registrare un account, il quale verrà memorizzato in un database. Gli utenti avranno la possibilità di creare un account fornendo le informazioni richieste e accedendo così a tutte le funzionalità offerte dal sito. Le informazioni dell'account saranno gestite in modo sicuro e conforme alle normative sulla privacy.Inoltre, gli utenti potranno aggiungere valutazione sui film presenti nel database del sito. Questa funzionalità permetterà loro di condividere le proprie opinioni sui film visti, arricchendo così l'esperienza degli altri utenti del sito.

**3.2 Read :**

Il nostro sito utilizza le API di DMDB (Database dei Film) per fornire agli utenti un accesso rapido e intuitivo alle informazioni sui film. Attraverso una casella di testo, gli utenti possono cercare i file desiderati per titolo. Il sistema quindi restituirà una lista di film corrispondenti al criterio di ricerca. Cliccando su uno specifico film, gli utenti potranno visualizzare una scheda dettagliata contenente informazioni complete sul film, comprese le valutazione degli utenti.

**3.3 Update:**

Gli utenti avranno la possibilità di modificare i propri dati di account, consentendo loro di mantenere aggiornate le informazioni personali, come ad esempio l'indirizzo email o la password.Inoltre, potranno anche aggiornare le valutazione precedentemente inserite, consentendo loro di riflettere eventuali cambiamenti di opinione o di correggere eventuali errori.

**3.4 Delete:**

Gli utenti avranno il controllo completo sui propri dati e potranno scegliere di cancellare il proprio account in qualsiasi momento. Questo processo garantirà la rimozione completa dei dati personali dal database del sito. Inoltre, gli utenti avranno la possibilità di cancellare le valutazione precedentemente inserite, offrendo loro un controllo completo sulla propria esperienza sul sito.

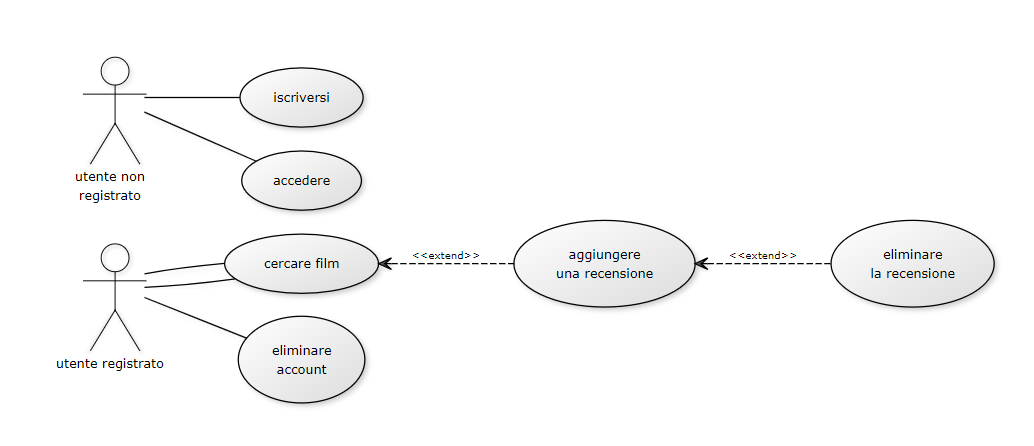
**3.5 Miglioramento del Contenuto:**

Per migliorare l'esperienza degli utenti, il sito sarà costantemente aggiornato con nuove funzionalità e contenuti. Questo potrebbe includere l'implementazione di sistemi di raccomandazione per suggerire film in base alle preferenze degli utenti, l'integrazione di social media per condividere valutazione e interagire con altri utenti, e l'espansione del database dei film per includere una vasta gamma di titoli. pag 9

Marco Perico 08/03/2024

L'obiettivo principale è quello di offrire agli utenti un'esperienza coinvolgente e soddisfacente, fornendo loro gli strumenti necessari per esplorare il mondo del cinema e condividere le proprie opinioni con una community appassionata.

4 UML:

5 WBS:

1. Progetto "Film For You"

1.1 Analisi e Pianificazione

1.1.1 Definizione degli Obiettivi del Progetto

1.1.2 Analisi dei Requisiti

1.1.3 Identificazione delle Risorse Necessarie

1.1.4 Pianificazione Temporale e delle Attività

1.2 Sviluppo del Sistema

1.2.1 Progettazione dell'Architettura del Sistema

1.2.2 Implementazione delle Funzionalità Principali

1.2.3 Integrazione con le API di DMDB (Database dei Film)

1.2.4 Realizzazione dell'Interfaccia Utente

1.3 Testing e Validazione

1.3.1 Creazione dei Test di Unità

1.3.2 Test dell'Integrazione del Sistema

1.3.3 Verifica dei Requisiti Funzionali e Non Funzionali

1.3.4 Risoluzione dei Bug e Ottimizzazione delle Prestazioni

1.4 Documentazione

1.4.1 Stesura della Documentazione Tecnica

1.4.2 Creazione di Manuali Utente

1.4.3 Preparazione di Materiali di Supporto per la Presentazione

pag 10

Marco Perico 08/03/2024

1.5 Implementazione e Lancio

1.5.1 Preparazione dell'Ambiente di Produzione

1.5.2 Training degli Utenti Finali

1.5.3 Lancio Ufficiale del Prodotto

1.6 Monitoraggio e Manutenzione

1.6.1 Monitoraggio delle Prestazioni del Sistema

1.6.2 Gestione delle Richieste di Supporto e Manutenzione

1.6.3 Aggiornamenti e Miglioramenti Continui del Software

**6 MATRICE DEI RISCHI:**

**6.1 Mancanza di collaborazione**: Potrebbe derivare da difficoltà comunicative o dalla carenza di interesse ed impegno da parte di un membro del gruppo. Una possibile strategia per risolvere questo problema potrebbe essere il richiamo del soggetto inadempiente.

**6.2 Abbandono da parte di un componente del gruppo**: In caso di un membro che si ritiri dal progetto, è importante attuare una ripartizione del carico di lavoro tra i membri rimasti per mitigare lo sconvolgimento della pianificazione.

**6.3 Problemi di salute**: Eventuali problemi di salute possono essere gestiti attraverso una ripartizione del lavoro tra i membri rimasti, garantendo che il lavoro continui senza intoppi.

**6.4 Malfunzionamento dei sistemi informatici**: Durante la fase di realizzazione del progetto, il malfunzionamento dei sistemi informatici potrebbe presentare problemi. In questo caso, una soluzione potrebbe essere quella di utilizzare le macchine presenti nei laboratori dell’università se il computer personale non funziona correttamente.

**6.5 Difficoltà di utilizzo degli strumenti**: Le difficoltà nell'utilizzo degli strumenti possono essere superate colmando le lacune conoscitive attraverso lo studio e l'approfondimento dei concetti mancanti da parte dei membri del gruppo.

**6.6 Difficoltà nello sviluppo delle componenti del progetto**: Nel caso di difficoltà nello sviluppo delle componenti del progetto, è fondamentale cercare soluzioni e ricorrere all'aiuto degli altri membri del gruppo per superare gli ostacoli.

pag 11

Marco Perico 08/03/2024

**6.7 Mancanza di esperienza dell’ambito del progetto**: Se nessun membro del team di sviluppo ha esperienza nell'ambito del progetto, potrebbe

essere necessario investire tempo nella formazione e nell'apprendimento delle competenze necessarie per affrontare le sfide.

**6.8 Modifiche situazioni ambientali**: Le modifiche nelle situazioni ambientali, come la modifica di leggi o regolamenti, possono influenzare il progetto. È importante essere flessibili e adattarsi alle nuove circostanze.

**6.9**. **Perdita dei dati**: Per prevenire la perdita di dati, è essenziale effettuare regolari backup dei dati prodotti durante la realizzazione del progetto.

**6.10 Rivalutazione del budget**: La rivalutazione del budget è un evento che potrebbe impattare il progetto, pertanto è necessario monitorare attentamente le risorse finanziarie e adattare la pianificazione di conseguenza.

pag 12

Marco Perico 08/03/2024

|  | rischio | probabilità | impatto | azione |
| --- | --- | --- | --- | --- |
| 1 | mancanza di collaboratori | media | medio | richiamo del  componente |
| 2 | abbandono di un componente del gruppo | media  bassa | molto alta | riorganizzazione  del lavoro |
| 3 | problemi di salute | bassa | basso | riorganizzazione  del lavoro |
| 4 | Malfunzionamento dei sistemi informatici | Molto  basso | Basso | Sostituzione/  riparazione |
| 5 | Difficoltà di utilizzo degli strumenti di sviluppo | basso | basso | studio |
| 6 | Difficoltà nello sviluppo dei componenti di progetto | medio | medio | ricerca una soluzione |
| 7 | Mancanza di esperienza dell’ambito del progetto | molto  basso | molto  alto | consulenza |
| 8 | modifiche situazioni ambientale | molto  basso | medio alto | studio normativo |
| 9 | Perdita di dati | Basso | molto  alto | backup  frequenti |
| 10 | rivalutazione del budget | molto basso | molto  alto | rivalutazione obiettivi |

pag 13

Marco Perico 08/03/2024

**7 MATERIALE UTILIZZATO:**

**7.1 Linguaggi utilizzati:**

**7.1.1 JavaScript**: Utilizzato per il controllo dei caratteri e per aggiungere interattività alle pagine web.

**7.1.2 PHP**: Utilizzato per interagire con il database e gestire le operazioni lato server.

**7.1.3 HTML**: Utilizzato per la struttura delle pagine web, definendo il contenuto e la disposizione degli elementi.

**7.1.4 CSS**: Utilizzato per lo stile e la formattazione delle pagine HTML, permettendo di definire l'aspetto visivo dei contenuti.

**7.1.5 SQL**: Utilizzato per interrogare e gestire i database relazionali, permettendo di memorizzare e recuperare dati in modo efficiente.

7.2 **Struttura e memorizzazione dei contenuti**:

**7.2.1** **Database relazionali**: Utilizzati per memorizzare dati strutturati in tabelle

**7.2.2** **File JSON**: Utilizzati per memorizzare dati in un formato leggero e facilmente leggibile

**8 LESSON LEARNED:**

Le lezioni apprese durante questo processo saranno preziose risorse per affrontare sfide future e migliorare continuamente le proprie competenze. Marco ha sperimentato l'importanza di strumenti come Git per gestire le versioni del progetto e creare copie di backup, imparando questa lezione a caro prezzo dopo aver perso la versione 1.0 del progetto. Questo duro colpo ha sottolineato l'importanza di una solida strategia di versioneamento e backup fin dall'inizio di un progetto.Inoltre, ha acquisito una consapevolezza più profonda sull'efficacia della gestione del tempo. Con un progetto di grandi dimensioni, Marco ha imparato a pianificare attentamente le attività, stabilire priorità e lavorare in modo efficiente per rispettare le scadenze. L'esperienza ha dimostrato quanto sia cruciale una gestione efficace del tempo per garantire il successo e la qualità del lavoro svolto. pag 14

Marco Perico 08/03/2024

Un'altra importante lezione emersa è stata l'importanza del Work Breakdown Structure (WBS). Inizialmente potrebbe sembrare solo una formalità, ma Marco ha scoperto quanto fosse uno strumento utile per scomporre il progetto

**9 CONCLUSIONE:**

Il progetto "Film For You" rappresenta un'importante esperienza di sviluppo per Marco Perico, evidenziando la sua competenza nell'applicare concetti teorici fondamentali come le API, il CRUD, il REST, l'UML e la gestione dei rischi. Attraverso un'analisi dettagliata, è stato possibile esplorare gli obiettivi del progetto, i concetti teorici applicati e le lezioni apprese durante lo sviluppo. L'implementazione del CRUD ha permesso agli utenti di interagire in modo completo con il sistema, consentendo loro di creare, leggere, aggiornare ed eliminare informazioni sul sito. L'utilizzo di UML ha facilitato la progettazione e la comprensione dell'architettura del sistema, mentre la struttura della WBS ha organizzato efficacemente il lavoro necessario per completare il progetto. La matrice dei rischi ha identificato potenziali problematiche e ha fornito azioni correttive per gestirle in modo efficace, mentre le lezioni apprese hanno contribuito a migliorare le pratiche di gestione dei progetti future di Marco. L'uso di linguaggi di programmazione e strutture dati specifiche ha arricchito ulteriormente l'esperienza di sviluppo, fornendo una base solida per la crescita e il successo futuro. In conclusione, il progetto "Film For You" rappresenta un risultato significativo per Marco Perico, dimostrando la sua capacità di sviluppare e gestire progetti complessi nel campo della tecnologia. Le lezioni apprese durante questo processo saranno preziose risorse per affrontare sfide future e migliorare continuamente le proprie competenze.

pag 15