



Ingegneria del Software 2016/2017

Piano di Progetto

Applicazione "Agorà"

v. 2.0 - 28 febbraio 2017

Gruppo: **NoPanic**
Antonio Emanuele Cinà 854866
Feliks Hibraj 854342
Paula Manzano 984308
Federico Marcuzzi 853770
Elia Lo Monaco 826197
Lorenzo Veronese 852058

Indice

1	Introduzione	1
1.1	Overview del Progetto	1
1.2	Deliverables del Progetto	1
1.3	Evoluzione del Progetto	1
1.4	Materiale di Riferimento	2
1.5	Definizioni e Abbreviazioni	2
2	Organizzazione del Progetto	4
2.1	Modello del Processo	4
2.2	Struttura Organizzativa	4
2.3	Interfacce Organizzative	4
2.4	Responsabilità di Progetto	4
3	Descrizione dei Processi Gestionali	6
3.1	Obbiettivi e Priorità	6
3.2	Assunzioni, Dipendenze, Vincoli	6
3.3	Gestione dei Rischi	6
3.4	Meccanismi di Monitoraggio e Controllo	7
3.5	Pianificazione dello staff	7
4	Descrizione dei Processi Tecnici	8
4.1	Metodi, Strumenti e Tecniche	8
4.1.1	Sistemi di calcolo	8
4.1.2	Struttura del team	8
4.2	Documentazione del Software	8
4.3	Funzionalità di supporto al progetto	9
4.3.1	Pianificazione della qualità	9
4.3.2	Pianificazione di gestione delle configurazioni	9
5	Pianificazione del Lavoro, delle Risorse e del Budget	10
5.1	Work Breakdown Structure	10
5.2	Dipendenze	11
5.3	Risorse Necessarie	15
5.4	Allocazione del Budget e delle Risorse	15
5.5	Pianificazione	15

1 Introduzione

1.1 Overview del Progetto

Al fine di valorizzare gli open data resi disponibili a livello nazionale si propone lo sviluppo di un'applicazione per sistema operativo Android che possa promuovere i luoghi e gli eventi culturali presenti nel territorio.

Applicazione Nella vista principale l'applicazione proposta permette agli utenti di visualizzare gli eventi o i luoghi della cultura vicini alla sua posizione o ad una posizione specificata mediante una barra di ricerca, con la possibilità di filtrare i risultati per tipologia o data.

Ogni luogo o evento può essere selezionato mostrando la sua scheda informativa con descrizione, un'immagine, sito web, telefono, costo dei biglietti e posizione. Dalla scheda informativa è anche possibile avviare l'applicazione *Maps* di *Google* per ottenere indicazioni stradali.

Una sezione preferiti dà la possibilità all'utente di salvare le informazioni degli eventi o luoghi che lo interessano, per la consultazione offline delle informazioni disponibili e per l'impostazione di notifiche.

Dati Utilizzati L'applicazione utilizza gli open data forniti dal *Ministero dei beni e delle attività culturali e del turismo* attraverso il servizio web *REST* della banca dati *DBUnico* che permette l'accesso ad informazioni aggiornate che riguardano **luoghi della cultura, eventi e comunicati**.

1.2 Deliverables del Progetto

Il progetto porterà alla produzione di diversi documenti e oggetti con le seguenti scadenze:

- ◇ (21/10/2016) Piano di Progetto
- ◇ (11/11/2016) Documento di analisi e specifica
- ◇ (10/12/2016) Documento di Progettazione
- ◇ (20/12/2016) Piano di testing
- ◇ (28/02/2017) Versione Finale dell'Applicazione

Ogni Scadenza si riferisce alle ore 23.59 del giorno indicato.

Durante la fase di sviluppo verranno realizzati diversi prototipi che permetteranno al gruppo di ottenere feedback da parte del committente e di un gruppo di utenti finali. Questi non avranno scadenze precise e definite a priori, ma se ne prevede un rilascio mensile.

Ogni documento, se necessario, verrà revisionato ed eventualmente modificato.

1.3 Evoluzione del Progetto

Questo paragrafo descrive le modifiche applicabili al progetto, durante le fasi di progettazione e sviluppo.

Eventuali evoluzioni future saranno riportate in una successiva versione del *Piano di Progetto*

1.4 Materiale di Riferimento

1. Slide del corso di ingegneria e indicazioni del docente presenti al sito web http://blogs.unive.it/groups/software_engineering_2012/
2. Libro *Android 5 Programming by Example* di Kyle Mew, fornito dal docente
3. Documentazione API Android (<https://developer.android.com/reference/>)
4. Manuale *Impara L^AT_EX!* di Marc Baudoin
5. Guida *L^AT_EX Wikibooks* (<https://en.wikibooks.org/wiki/LaTeX>)
6. Manuale Servizi REST DBUnico 2.0 (<http://www.beniculturali.it/mibac/multimedia/MiBAC/minisiti/DB2/document/DBUnico20%20Servizi%20Rest%20rev%201.5.pdf>)
7. Specifica Schema DBUnico 2.0 (<http://www.beniculturali.it/mibac/xsd/MibacSchema-1.2.xsd>)

1.5 Definizioni e Abbreviazioni

Di seguito sono elencati i termini e le abbreviazioni utilizzate in questo documento:

Android	Sistema operativo per dispositivi mobili sviluppato da Google. Si farà riferimento alle versioni a partire dalla 4.1.
Play Store	Piattaforma di distribuzione per applicazioni per sistema Android gestita da Google.
Android Studio	Ambiente di sviluppo integrato (IDE) ufficiale per lo sviluppo per la piattaforma Android.
REST	(<i>Representational State Transfer</i>) Tipo di architettura software per i sistemi distribuiti che utilizza le tecnologie e i protocolli del WorldWide Web.
Open Data	Dati liberamente accessibili a tutti le cui eventuali restrizioni sono l'obbligo di citare la fonte o di mantenere la banca dati sempre aperta.
MiBACT	<i>Ministero dei beni e delle attività culturali e del turismo</i>
DBUnico	Banca dati destinata contenente informazioni del MiBACT: luoghi della cultura, eventi, comunicati stampa, appalti, Enti.
L^AT_EX	Linguaggio di Markup usato per la preparazione di testi basato sul programma di composizione tipografica T _E X.
Git	Software di controllo versione distribuito utilizzabile da interfaccia a riga di comando.
API	(<i>Application Programming Interface</i>) Insieme di procedure disponibili al programmatore, di solito raggruppate a formare un set di strumenti specifici per l'espletamento di un determinato compito all'interno di un certo programma.
UI	<i>User Interface</i>
UX	<i>User Experience</i>

Diagramma di Gantt Strumento di supporto alla gestione dei progetti, costruito da un asse orizzontale - a rappresentazione dell'arco temporale totale del progetto, suddiviso in fasi incrementali - e da un asse verticale - a rappresentazione delle mansioni o attività che costituiscono il progetto.

Diagramma di Pert Metodo statistico di determinazione dei tempi delle attività di progetto, costituito da un grafo della rete di azioni che portano al raggiungimento degli obiettivi di progetto.

2 Organizzazione del Progetto

2.1 Modello del Processo

Per la realizzazione del progetto è stato scelto il *Modello Evolutivo*, che consente, mediante la realizzazione di una serie di prototipi, l'evoluzione, a partire da una specifica di massima, verso il sistema finale.

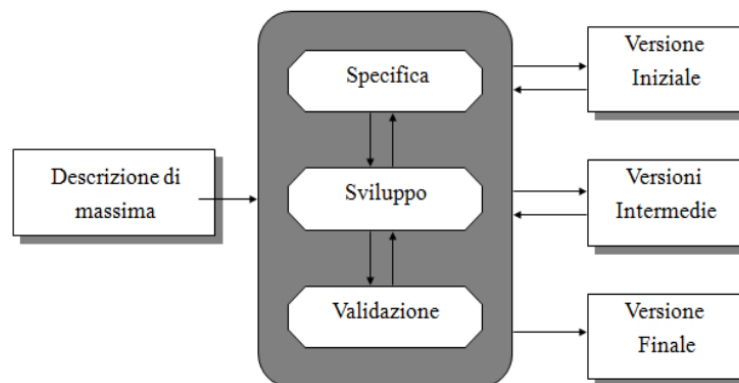


Figura 1: Modello Evolutivo

Sarà così possibile per il gruppo, testare e discutere, con committente e utenti, le varie sezioni realizzate e apportare modifiche in base al feedback ricevuto. Sarà anche più facile la scelta tra le varie alternative implementative utilizzando il test dei prototipi come strumento di valutazione.

2.2 Struttura Organizzativa

Team

Il team è formato da 6 studenti iscritti al corso di Ingegneria del Software

Tipologia di Team

Il gruppo è organizzato secondo la tipologia Democratica Decentralizzata. Non vi è quindi un leader permanente, e l'organizzazione del lavoro avviene grazie al consenso di tutti i componenti. Sarà così più facile il monitoraggio e la risoluzione degli eventuali problemi.

2.3 Interfacce Organizzative

Il Gruppo avrà relazioni con le seguenti figure:

- ◊ Il *Professor Agostino Cortesi (Committente)*, per ottenere feedback e consigli sulle eventuali correzioni o cambiamenti
- ◊ Gli *Utenti*: colleghi e amici che possano fornire feedback durante le fasi di progettazione e sviluppo, e gli utenti finali dell'applicazione
- ◊ Gli altri *Gruppi*: i diversi gruppi potranno comunicare tra loro, confrontando il lavoro svolto, per favorire uno scambio di idee e conoscenze.

2.4 Responsabilità di Progetto

Come specificato nella sezione 2.2 è stata scelta una struttura democratica decentralizzata, quindi ogni membro del gruppo collaborerà alle singole funzioni. Si è scelto comunque di assegnare ad

ogni funzione uno o più responsabili, con il ruolo di monitorare e assicurare la qualità del lavoro. Le funzioni identificate sono:

- ◇ Stesura e revisione della documentazione: Elia Lo Monaco, Lorenzo Veronese
- ◇ Design dell'Interfaccia Utente: Feliks Hibraj, Paula Manzano
- ◇ Scrittura, Revisione e Test del Codice: Antonio Emanuele Cinà, Federico Marcuzzi
- ◇ Gestione delle Relazioni con committente, fornitore e utenti: Feliks Hibraj

All'inizio di ogni fase di sviluppo o progettazione si procederà alla suddivisione dei compiti in base alle competenze dei vari membri del gruppo.

3 Descrizione dei Processi Gestionali

3.1 Obiettivi e Priorità

Gli obiettivi che ci prefiggiamo sono:

- ◊ Sviluppare un'app funzionante e *user friendly* (priorità Alta)
- ◊ Rispettare i tempi di consegna e i derivable (priorità Alta)
- ◊ Creare una struttura organizzativa interna al nostro team tale da rendere efficiente il coordinamento nelle varie attività da svolgere (priorità Media)

3.2 Assunzioni, Dipendenze, Vincoli

Il team assume che gli OpenData forniti dal MiBACT siano sempre liberamente disponibili e soprattutto costantemente aggiornati, per un corretto sviluppo e funzionamento dell'applicazione. I vincoli principali sono rappresentati dalle scadenze fissate al punto 1.2.

3.3 Gestione dei Rischi

Rischio	Categoria	Probabilità	Impatto	Azione
Incomprensioni che frenano il lavoro in gruppo (1)	Progetto	Bassa	Alto	Monitorare
Gravi problemi di salute di qualcuno dei membri del gruppo (2)	Processo	Bassa	Medio	-
Orari molto diversi fra i membri del gruppo, che rendano difficile monitorare il progetto tutti insieme (3)	Progetto	Medio Alta	Medio Basso	Mitigare
Errata valutazione dei tempi di sviluppo (4)	Progetto	Media	Medio	Mitigare
Fallimenti nel software e/o hardware utilizzati per lo sviluppo (5)	Progetto Processo	Medio Bassa	Medio	-
Complessità dell'applicazione maggiore di quella prevista (6)	Progetto Prodotto	Media	Alto	Monitorare

Tabella 1: Rischi di Progetto

Impatto	1		6		
	2	5	4		
				3	

Probabilità

Tabella 2: Classificazione dei Rischi

3.4 Meccanismi di Monitoraggio e Controllo

Sono previsti dei checkpoint intermedi durante le varie fasi del progetto, al fine di testare la bontà delle scelte implementative fatte dal singolo ed eventualmente discutere in gruppo delle soluzioni migliorative, qualora talune decisioni complicassero il lavoro di un altro membro del team, impegnato nello sviluppo di un'altra funzionalità dell'applicazione.

Questi "momenti" di controllo oltre a rendere più coordinato il lavoro fra i vari componenti della squadra, limitando quanto più possibile situazioni spiacevoli in cui bisogna rimettere mano pesantemente a tutto il codice, serviranno a ciascuno per avere una visione d'insieme del progetto che va prendendo forma.

Ogni sezione avrà un suo referente e per agevolare lo scambio di informazioni all'interno del team verrà usato principalmente Telegram, mentre per la elaborazione della documentazione necessaria per il progetto e per lo sviluppo del codice dell'applicazione stessa verrà usato il sistema di controllo versione Git.

3.5 Pianificazione dello staff

Gli skill necessari sono i seguenti:

- ◊ Conoscenza di Android Studio e delle Api di Android, oltre al linguaggio Java
- ◊ Buone conoscenze di Project Management
- ◊ Capacità di elaborazione scritta e stesura della documentazione secondo le linee guida indicate dal professore

L'assegnazione dei vari compiti riguardanti il progetto terrà conto delle pregresse conoscenze di ciascun componente del gruppo; in maniera tale da valorizzare le specificità del singolo, favorendo le naturali inclinazioni, senza però perdere di vista l'obiettivo principale di una generale maturazione da parte di tutti degli skill richiesti per l'esecuzione del progetto.

4 Descrizione dei Processi Tecnici

4.1 Metodi, Strumenti e Tecniche

4.1.1 Sistemi di calcolo

Software

Per realizzare l'App il team utilizzerà AndroidStudio, aggiornato alla versione 2.2.

Il linguaggio di programmazione sarà Java, la piattaforma target Android e verrà usato Git come sistema di versioning, utilizzando una repository remota installata su un server gestito da un componente del gruppo.

L^AT_EX verrà utilizzato per redigere i documenti.

La parte di sviluppo del codice sarà svolta su personal computer con diversi sistemi operativi (Ubuntu 16.04, OSX 10.10, Arch Linux, Windows 10).

I diagrammi di Gantt e Pert sono stati realizzati utilizzando il software *GanttProject*.

Hardware

Dispositivi android posseduti:

- ◊ Nexus 5X con Android Nougat 7.0
- ◊ Nexus 5 con Android Marshmallow 6.0
- ◊ Nexus 7 con Android KitKat 4.4

Dispositivi emulati:

- ◊ Nexus 5 con Android 4.1
- ◊ Nexus 5 con Android 5.0
- ◊ Nexus 5 con Android 6.0

4.1.2 Struttura del team

Essendo l'organizzazione di tipo Democratica Decentralizzata, il lavoro per l'avanzamento del progetto sarà uniformemente distribuito tra i membri del gruppo secondo le responsabilità stabilite all'inizio di ogni attività.

Verranno utilizzati sistemi di messaggistica e incontri per discutere le principali caratteristiche del sistema e l'utilizzo di un sistema come Git permetterà di lavorare contemporaneamente tenendo traccia delle modifiche.

4.2 Documentazione del Software

Documenti consegnati:

- ◊ Presentazione proposta di progetto v1.0 (13/10/2016)

Documenti redatti periodicamente:

- ◊ Piano di progetto (consegna prevista 21/10/2016)
- ◊ Documento di analisi e specifica (consegna prevista 11/11/2016)
- ◊ Documento di progettazione (consegna prevista 10/12/2016)
- ◊ Piano di testing (consegna prevista 20/12/2016)

Tutti i membri del gruppo si occupano della redazione e della revisione dei documenti, che verranno prodotti durante lo sviluppo del progetto.

4.3 Funzionalità di supporto al progetto

4.3.1 Pianificazione della qualità

- ◇ Rispettare i requisiti funzionali e non funzionali
- ◇ Aggiornare la documentazione
- ◇ Scrivere il codice il modo tale da permettere la riusabilità e una facile manutenzione

4.3.2 Pianificazione di gestione delle configurazioni

- ◇ Tenere traccia delle modifiche tramite il sistema Git per monitorare ed eventualmente intervenire in caso di errori

5 Pianificazione del Lavoro, delle Risorse e del Budget

5.1 Work Breakdown Structure

1. Piano di progetto
 - 1.1. Definizione ambito del progetto
 - 1.2. Definizione obiettivi del progetto
 - 1.3. Analisi delle risorse necessarie
 - 1.3.1. Verifica e raccolta Open Data
 - 1.3.2. Definizione le competenze necessarie
 - 1.4. Responsabilità del progetto
 - 1.4.1. Identificare responsabilità
 - 1.4.2. Suddivisione delle responsabilità tra i componenti del gruppo
 - 1.5. Analisi dei rischi
 - 1.5.1. Identificare i rischi
 - 1.5.2. Definizione probabilità ed impatto
 - 1.5.3. Definizione piano di gestione dei rischi
 - 1.6. Stima dei costi
 - 1.6.1. Definizione costi diretti
 - 1.6.2. Definizione costi indiretti
 - 1.7. Definizione piano di comunicazione
 - 1.7.1. Definizione modalità di comunicazione
 - 1.7.2. Definizione formato e contenuto dei documenti
 - 1.8. Definizione piano di testing
 - 1.9. Definizione piano della qualità
 - 1.9.1. Definizione attività di riesame, verifica, validazione
 - 1.9.2. Definizione strumenti di controllo
 - 1.10. Stesura documento del piano di progetto
2. Formazione del gruppo
 - 2.1. Formazione utilizzo tecnologie REST
 - 2.2. Formazione sviluppo applicazioni Android
3. Progettazione concettuale
 - 3.1. Analisi dei requisiti
 - 3.1.1. Analisi requisiti funzionali
 - 3.1.1.1. Analisi funzioni utente
 - 3.1.1.2. Analisi UI
 - 3.1.1.3. Analisi UX
 - 3.1.1.4. Analisi modalità di navigazione
 - 3.1.2. Analisi requisiti non funzionali
 - 3.1.2.1. Analisi sicurezza
 - 3.1.2.2. Analisi scalabilità
 - 3.1.2.3. Analisi livello di servizio richiesto

- 3.2. Analisi delle funzioni
- 3.3. Analisi dei dati
 - 3.3.1. Identificare le entità
 - 3.3.2. Identificare le relazioni
- 3.4. Predisporre prototipo
- 3.5. Stesura documento di specifica dei requisiti
- 4. Stesura documento di testing
- 5. Progettazione tecnica
 - 5.1. Verifica specifiche funzionali
 - 5.2. Definizione ambiente di sviluppo
 - 5.2.1. Scelta framework per lo sviluppo dell'applicazione
 - 5.2.2. Scelta tool di versioning per codice e documenti
 - 5.3. Definizione tool per lo sviluppo dell'interfaccia grafica
 - 5.4. Stesura documento di progettazione
- 6. Sviluppo
 - 6.1. Verifica progettazione tecnica
 - 6.2. Implementazione logica applicazione
 - 6.3. Implementazione interfaccia grafica applicazione
- 7. Testing
 - 7.1. Test funzionalità utente
 - 7.2. Test di sicurezza
 - 7.3. Test di scalabilità
- 8. Rilascio
 - 8.1. Effettuare il deploy in produzione dell'applicazione
 - 8.2. Registrazione account su Play Store
 - 8.3. Pubblicazione su Play Store dell' applicazione
- 9. Controllo del progetto
 - 9.1. Definizione politica di manutenzione
 - 9.2. Controllo della pianificazione
 - 9.3. Controllo dei costi
 - 9.4. Controllo della qualità
 - 9.5. Riunioni di coordinamento con il team

5.2 Dipendenze

<i>Attività</i>	<i>Codice</i>	<i>Durata</i>	<i>Dipendenze</i>
Definizione ambito del progetto	1.1	1 Giorno	-
Definizione obiettivi del progetto	1.2	2 Giorni	1.1
Verifica e raccolta Open Data	1.3.1	3 Giorni	1.2
Definire le competenze necessarie	1.3.2	2 Giorni	-
Identificare Responsabilità	1.4.1	1 Giorno	1.3.2
Analisi dei Rischi	1.5	3 Giorni	-
Stima dei costi	1.6	1 Giorno	1.5
Definire piano di comunicazione	1.7	1 Giorno	-
Definire piano della qualità	1.9	1 Giorno	-
Stesura documento del piano di progetto	1.10	7 Giorni	1.2, 1.3.1, 1.4.1, 1.5, 1.6
Formazione del gruppo	2	11 Giorni	1.3.2
Analisi dei requisiti	3.1	2 Giorni	-
Analisi delle funzioni	3.2	2 Giorni	-
Predisporre prototipo	3.4	6 Giorni	3.1, 3.2
Stesura documento di specifica dei requisiti	3.5	15 Giorni	3.1, 3.2
Stesura documento di testing	4	8 Giorni	3.5
Progettazione Tecnica	5	24 Giorni	3.5
Implementazione logica applicazione	6.2	15 Giorni	5
Implementazione front-end applicazione	6.3	20 Giorni	5
Test	7	13 Giorni	4
Rilascio	8	1 Giorno	6.2, 6.3, 7

Tabella 3: Tabella delle Dipendenze

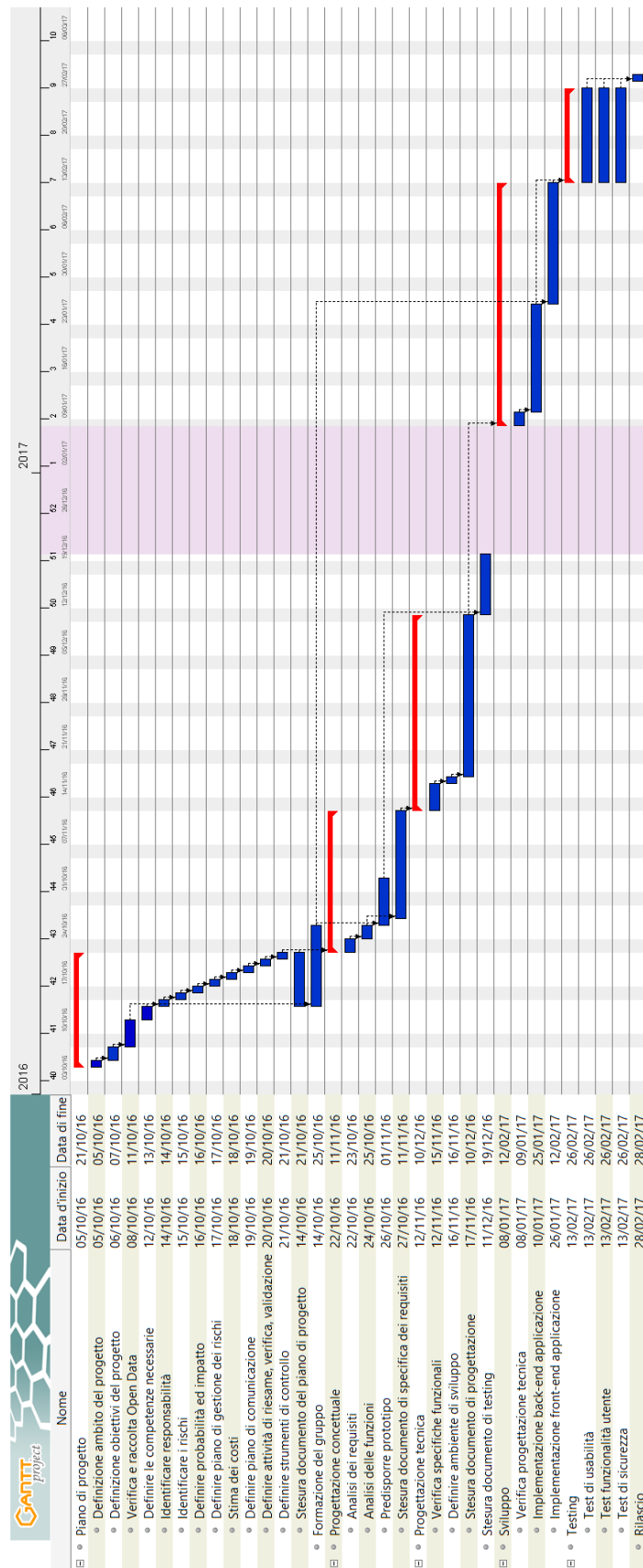


Figura 2: Diagramma di Gantt

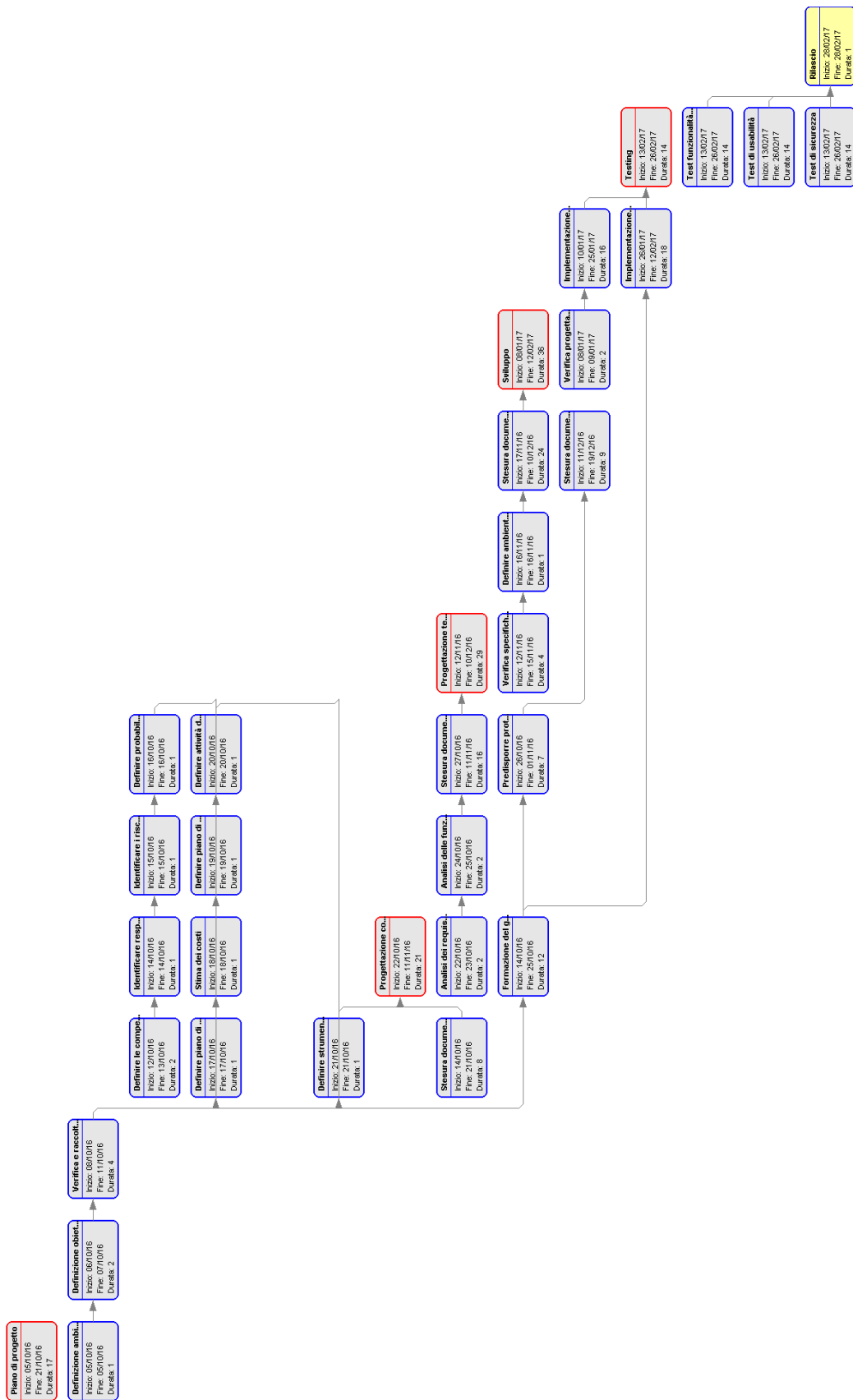


Figura 3: Diagramma di Pert

5.3 Risorse Necessarie

Personale

I sei componenti del gruppo iscritti al corso di Ingegneria del Software: Antonio Emanuele Cinà, Feliks Hibraj, Paula Manzano, Federico Marcuzzi, Elia Lo Monaco, Lorenzo Veronese.

Documentazione

La documentazione necessaria è elencata nel paragrafo 1.4, e le eventuali informazioni aggiuntive saranno ottenute tramite ricerca e approfondimento su Internet

Tecnologie e Hardware

L'hardware e le tecnologie utilizzate saranno quelle già in possesso dei membri del gruppo, in particolare quanto descritto nel paragrafo 4.1.1

Tempo

Si stima che in 139 giorni l'applicazione raggiunga la sua versione finale, come previsto dalle deadlines pianificate.

5.4 Allocazione del Budget e delle Risorse

La realizzazione dell'applicazione non richiede allocazione di budget per quanto riguarda gli strumenti utilizzati, poichè disponibili gratuitamente oppure già in possesso dei membri del gruppo.

Si stima che il numero di giorni necessari per il completamento dei task definiti al paragrafo 5.1 sia:

<i>Attività</i>	<i>Codice</i>	<i>Giorni</i>	<i>Giorni/Uomo</i>
Piano di Progetto	1	22	110
Formazione del Team	2	11	55
Progettazione Concettuale	3	25	125
Stesura Docuemnto di Testing	4	8	40
Progettazione Tecnica	5	24	120
Sviluppo	6	35	125
Test	7	13	65
Rilascio	8	1	5
Totale		139	645

Tabella 4: Tabella Allocazione Tempo

I giorni/uomo sono calcolati tenendo conto che alcuni componenti del gruppo, per impegni estranei al progetto non possano garantire un impegno esclusivo: si assume, quindi, una media di 5 su 6 persone che lavorano.

Assumendo che un giorno/uomo corrisponda a 2.5 ore lavorative, e che la retribuzione sia di 5 Euro/ora, il budget è fissato a 8062.5 Euro.

5.5 Pianificazione

La pianificazione del progetto seguirà le scadenze stabilite con il Professor Cortesi durante il corso, in particolare si seguirà la stesura dei documenti richiesti e lo sviluppo dell'applicazione rispettando le date descritte nel paragrafo 1.2.