

# **Università degli Studi di Salerno**

**Corso di Ingegneria del Software**

## **MeerKat** **Requisiti e casi d'uso** **Versione 0.1.3**



Data: 25/10/2024

Progetto: MeerkKat	Versione: 0.1.3
Documento: Requisiti e casi d'uso	Data: 25/10/2024

**Coordinatore del progetto:**

Nome	Matricola

**Partecipanti:**

Nome	Matricola
Stefano Nicolò Zito	0512117457
Gabriel Tabasco	0512116905
Giuseppe Ballacchino	0512116596
Francesco Giuseppe Trotta	0512116788

<b>Scritto da:</b>	Stefano Nicolò Zito
--------------------	---------------------

**Revision History**

Data	Versione	Descrizione	Autore
19/10/2024	0.0	Scrittura indice.	Stefano Nicolò Zito
22/10/2024	0.1	Rifacimento indice, aggiunta requisiti e scenari, modifiche scenari, scrittura casi d'uso 5,6,9	Stefano Nicolò Zito
23/10/2024	0.1.1	Aggiunti UC7 ed UC8 e sistemazione elenchi puntati	Gabriel Tabasco
25/10/2024	0.1.2	Aggiunti UC1 ed UC4 con relativi diagrammi	Giuseppe Ballachino
25/10/2024	0.1.3	Final Revision	Stefano Nicolò Zito, Giuseppe Ballachino, Gabriel Tabasco, Francesco Giuseppe Trotta

# Indice

1.	Introduzione.....	4
1.1.	Purpose of the System.....	4
1.2.	References .....	4
2.	Current System .....	4
3.	Sistema Proposto .....	5
3.1.	Overview .....	5
3.2.	Functional Requirements .....	5
3.3.	Non Functional Requirements.....	7
3.3.1.	Usability .....	7
3.3.2.	Reliability.....	7
3.3.3.	Performance .....	7
3.3.4.	Packaging .....	7
3.3.5.	Legal.....	7
3.4.	System Models .....	8
3.4.1.	Scenari.....	8
<u>3.4.1.1.</u>	Scenario registrazione utente .....	8
<u>3.4.1.2.</u>	Scenario Creazione di un team.....	8
<u>3.4.1.3.</u>	Scenario Creazione di Task.....	9
<u>3.4.1.4.</u>	Scenario: Svolgimento di task.....	9
<u>3.4.1.5.</u>	Scenario: messaggio di gruppo al team.....	10
<u>3.4.1.6.</u>	Scenario: Messaggio individuale tra utenti .....	10
3.4.2.	Use Case Model .....	11
<u>3.4.2.1.</u>	Use Case Diagram.....	11
<u>3.4.2.2.</u>	UC1: Registrazione .....	12
<u>3.4.2.3.</u>	UC2: Create Team.....	12
<u>3.4.2.4.</u>	UC3: Modifica info di un team .....	14
<u>3.4.2.5.</u>	UC4: AddTask .....	15
<u>3.4.2.6.</u>	UC5: CreateTaskList.....	16
<u>3.4.2.7.</u>	UC6: MooveTaskInList.....	17
<u>3.4.2.8.</u>	UC7: TaskDelivery.....	18
<u>3.4.2.9.</u>	UC8: StartTimer.....	19
<u>3.4.2.10.</u>	UC9: SendMessage .....	20

## **1. Introduzione**

### ***1.1. Purpose of the System***

Lo scopo del sistema MeerKat è di realizzare un luogo unico dove le aziende possano organizzare progetti e gruppi di lavoro, permettendo nella stessa applicazione di organizzare e creare Task, comunicare con gli altri membri del progetto in modo ben organizzato e ricevere dati riguardanti come il team sta lavorando e come procede il progetto in tempo reale.

### ***1.2. References***

- MeerKat Problem Statement

## **2. Current System**

Alla situazione corrente non esiste un sistema che permetta di eseguire funzioni di gestione del team, gestione dei task e comunicazione testuale tutto insieme, questo richiede agli utenti di svolgere ognuna di queste attività separatamente, richiedendo quindi agli utenti di registrarsi su molti sistemi software sia desktop che web based, risultato in un maggior uso di tempo dall'utente, soprattutto se una singola attività svolta va riportata su sistemi diversi non integrati fra di loro.

### 3. Sistema Proposto

#### 3.1. Overview

Questa piattaforma integra tutte le funzionalità necessarie per la gestione dei progetti e del lavoro di gruppo in un'unica applicazione, ottimizzata per semplificare il lavoro quotidiano. Con MeerKat, i manager possono facilmente creare gruppi di lavoro, assegnare membri a specifici team promuovendo così un ambiente di lavoro ben strutturato.

#### 3.2. Functional Requirements

Gestione dei progetti

- Un manager può creare un nuovo progetto specificando nome, descrizione ed eventuale data di scadenza.
- Il manager può modificare i dettagli del progetto in qualsiasi momento.
- Il manager può visualizzare un elenco di tutti i progetti attivi.

Gestione del team:

- Il manager può assegnare e rimuovere dipendenti a un progetto.
- Il sistema mostra un elenco aggiornato dei membri del team per ogni progetto.

Gestione dei task:

- Il manager può creare task all'interno di un progetto, specificando titolo, descrizione e scadenza.
- I task possono essere assegnati a uno o più membri del team.
- I task sono raccolti in liste.
- I membri del team possono aggiornare lo stato dei task, notificandone il completamento.
- Il sistema fornisce una vista d'insieme dei task per progetto, mostrando il loro stato di avanzamento.

Monitoraggio dell'avanzamento:

- Il sistema genera report sull'avanzamento del progetto, mostrando la percentuale di completamento dei task.
- Il manager può visualizzare grafici e statistiche sull'andamento del progetto nel tempo.
- I membri del team possono visualizzare grafici e statistiche sull'andamento del progetto e le proprie prestazioni.

Time tracking:

- Il sistema è in grado di tracciare il tempo speso dai dipendenti a lavorare, in

generale e su specifiche task.

- I dati di time tracking sono aggregati e visualizzabili dal manager per ogni dipendente e per ogni progetto.

### ***3.3. Non Functional Requirements***

#### **3.3.1. Usability**

- Il sistema implementa un robusto sistema di autenticazione e autorizzazione.
- I dati dei progetti sono protetti e accessibili solo agli utenti autorizzati.
- Il sistema ha un uptime del 99.9% durante le ore lavorative.
- In caso di crash, il sistema può essere ripristinato entro 15 minuti senza perdita di dati.

#### **3.3.2. Reliability**

- Il sistema risponde alle richieste degli utenti entro 2 secondi in condizioni normali di carico.
- Il sistema può gestire almeno 100 utenti concorrenti senza degradazione delle prestazioni.

#### **3.3.3. Performance**

- Il sistema risponde alle richieste degli utenti entro 2 secondi

#### **3.3.4. Packaging**

- L'applicazione viene distribuita con un installer .net standard e/o una versione portable

#### **3.3.5. Legal**

- Il sistema deve rispettare le normative base sulla privacy
- Tutte le librerie utilizzate devono avere licenze compatibili con uso open source

### **3.4. System Models**

#### **3.4.1. Scenari**

##### **3.4.1.1. Scenario registrazione utente**

A Mario Rossi è stato detto dall'azienda per cui lavora che avrà bisogno di usare MeerKat per organizzare i suoi progetti. Dopo aver installato l'applicazione, Mario si ritrova nella schermata di creazione del account, dove gli viene chiesto di inserire il proprio nome e cognome, e-mail, data di nascita, password, dopo aver inserito i dati, l'operazione è completata e il sistema invierà a Mario un'e-mail di conferma, mentre lui può già iniziare a usare l'applicazione.

Mario il giorno seguente apre la sua casella di posta elettronica e usa il link di conferma inviatogli per confermare la sua mail finendo così la creazione del suo account.

##### **3.4.1.2. Scenario Creazione di un team**

Mario Rossi lavora per un'azienda di produzione di sistemi software e deve creare il team per partire con un nuovo progetto di lavoro da remoto. Per prima cosa esegue l'accesso all'applicazione MeerKat inserendo nella pagina d'accesso la sua e-mail aziendale e la sua password, dopo aver fatto ciò, si ritrova nella schermata d'elenco del team, dove può visualizzare un elenco organizzato in una griglia dei team di cui fa parte, ognuno con un titolo e un'immagine, qui sceglie l'opzione per creare un nuovo team e inserisce al interno del form che si è aperto davanti a lui: un nome e una breve descrizione, l'immagine del team e la attuale data di consegna del progetto, inseriti questi dati gli si apre un nuovo form da cui ha la possibilità di invitare i suoi collaboratori nel team, attraverso una funzione di ricerca tramite e-mail o un link di invito associato al progetto, Mario sceglie la prima opzione, e digita le email dei suoi 4 collaboratori nella barra di ricerca, trovandoli e selezionando l'opzione posta affianco ai nomi per inviare un invito al Team. A questo punto Mario viene portato alla homepage del suo nuovo team, dove attraverso un menu laterale potrà selezionare se accedere alla schermata delle statistiche di lavoro, quella per scrivere messaggi di gruppo o individuali ai suoi collaboratori o a quella per l'elenco e per l'organizzazione dei task.

Filippo Bianchi, è uno dei colleghi di Mario che sono stati scelti per collaborare allo stesso progetto software menzionato prima, Filippo apre l'applicazione di MeerKat e accede con e-mail aziendale e password, come Mario si ritrova nella schermata con l'elenco dei team di cui fa parte, da cui accede a un menu laterale per visualizzare gli inviti a un nuovo team, vede la notifica del team creato da Mario e accetta l'invito, venendo portato nella schermata principale del team.



### **3.4.1.3.      Scenario Creazione di Task**

Mario Rossi deve creare i primi task per il suo nuovo progetto, esegue gli stessi passi iniziali visti in 3.4.1.2, ma qui nella schermata di elenco dei progetti, sceglie il progetto a cui vuole accedere ritrovandosi nella pagina di homepage, dove sceglie tramite il menu di accedere alla pagina di organizzazione dei tasks. In questa schermata può vedere una serie di blocchi rappresentanti liste di task, ognuno di questi ha un titolo in cima, il nome della lista, e una serie di blocchetti più piccoli impilati uno sotto l'altro, questi sono i task, ognuno rappresentato da un nome e una data di scadenza, in questo momento sono presenti quattro liste, ognuna è una to-do list per uno dei collaboratori di Mario. La nuova Task è per Filippo Bianchi, Mario sceglie l'opzione per aggiungere un task alla to-do list Filippo e un form si apre, qui può dare un nome al task, scrivere una descrizione più in dettaglio e ovviamente inserire la data di scadenza del task, inseriti i dati necessari, Mario completa l'operazione e viene reindirizzato alla pagina di organizzazione dei task, dove può vedere che il nuovo task è stato aggiunto alla lista di Filippo. Dopo Aver creato il task in questo modo, Mario ci ripensa e decide che questo task, essendo un lavoro a lungo termine, dovrebbe stare in una lista differente, sceglie l'opzione per creare una nuova lista di task e si apre un nuovo form dove Mario può inserire il titolo della lista e scegliere una Task preesistente da inserire come prima task nella lista, Mario dà un nome alla Lista e sceglie la Task appena creata, dopo di che salva la sua scelta e torna alla pagina di gestione delle Task, dove la Task è stata spostata dalla to-do list di Filippo alla nuova lista.

### **3.4.1.4.      Scenario: Svolgimento di task**

Filippo Bianchi è il grafico del team di Mario Rossi e apre l'app di MeerKat seguendo i passi indicati in 3.4.1.3, fino al accesso alla schermata di gestione delle task, qui controlla la sua to-do list e vede che è stata aggiunta un task, quindi clicca sopra il blocco corrispondente, aprendo una pagina di info del task dove oltre al nome e alla data di scadenza, può vedere la descrizione scritta dal suo project manager, oltre a questo nella pagina è presente un campo che permette di caricare un file, eventuale prodotto da consegnare al team e uno per segnalare che il task è completato. Avendo capito la task assegnatogli, che consiste nel creare un'icona per l'applicazione che il suo team sta sviluppando, Filippo esce dalla pagina di focus del task e apre invece la pagina dedicata alle prestazioni, qui può vedere in alto una barra che rappresenta l'andatura del progetto rispetto alla consegna, in questo momento la barra si mostra non completamente riempita, in quanto sono presenti task non completate, oltre alla barra, vede anche dei grafici che presentano le sue performance rispetto ai task assegnatogli e al tempo speso a lavorare questa settimana e un timer che segna quanto tempo Filippo ha speso a lavoro quest'oggi, al momento impostato su 0 minuti, Filippo clicca sul timer avviandolo e compare un piccolo widget che può muovere attraverso lo schermo, mostrandogli il progresso del tempo e un bottone per la pausa, riduce MeerKat a icona e si mette a lavorare tenendo sott'occhio il timer.

Dopo qualche ora, Filippo ha finito il task, blocca quindi il timer e riapre MeerKat, si sposta nella pagina del task, carica il file dell'immagine che ha creato e notifica il sistema del completamento dell'incarico.

#### **3.4.1.5.      Scenario: messaggio di gruppo al team**

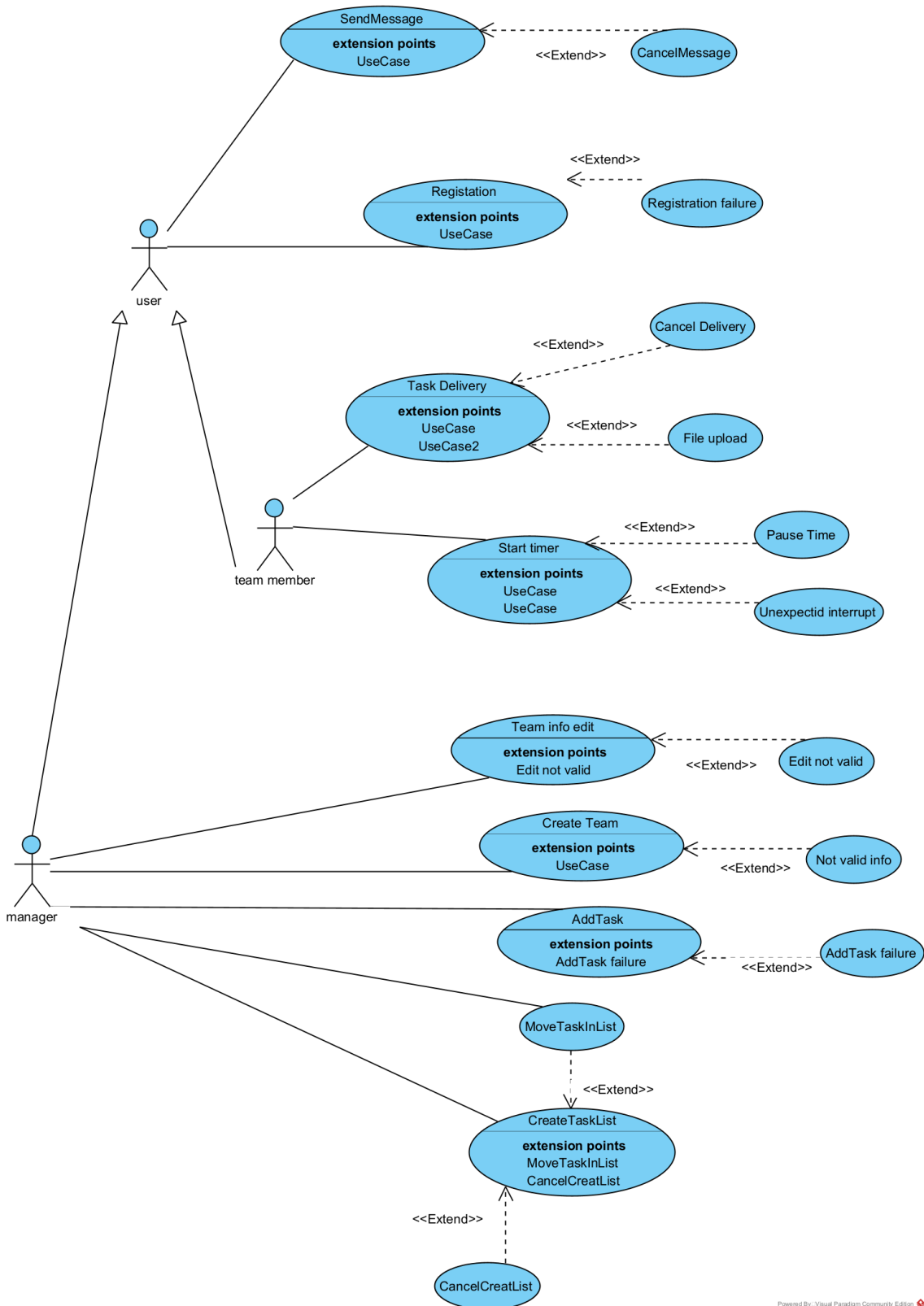
Mario Rossi deve scrivere un messaggio al suo team per avvertire i membri di un meeting che avverrà nella prossima settimana, segue i passi descritti in 3.4.1.2 e accede a MeerKat, spostandosi poi nella schermata home del team in questione e da lì nella schermata dedicata alla chat, qui un menù posto sul lato della pagina gli permette di scegliere se mandare un messaggio di gruppo o uno individuale agli altri membri, sceglie l'opzione di gruppo e nella pagina compare un elenco di topic di conversazione, simile a quelli di un forum, Mario sceglie di creare un nuovo topic e gli viene chiesto di inserire il nome, lo chiama "Meeting 20-10-2024", il topic viene creato e aperto, portandolo a una schermata simile a molte chat testuali con una barra a fondo pagina per vedere il messaggio che sta scrivendo e uno spazio per vedere i messaggi inviati in precedenza, al momento vuoto, scrive il messaggio e lo invia, vedendolo comparire nella sezione dei messaggi inviati, vicino al bordo destro della pagina.

#### **3.4.1.6.      Scenario: Messaggio individuale tra utenti**

Filippo Bianchi deve chiedere a Mario Rossi dei chiarimenti rispetto a uno degli ambiti discussi nell'ultimo meeting, segue quindi i passi descritti in 3.4.1.2 per l'accesso quelli descritti in 3.4.1.5 per arrivare al menu delle chat dove, sceglie dal menu l'opzione per mandare un messaggio individuale a Mario e sul lato destro dello schermo appare una chat simile a quella descritta sempre in 3.4.1.5, Filippo scrive il suo messaggio e preme invio, tiene d'occhio il messaggio per un po' e vede lo stato indicato sotto di questo passare da inviato e visualizzato. Da lì a poco nello spazio dei messaggi compare la risposta di Mario distinta dal colore e dal fatto che è vicina al bordo sinistro e non destro della schermata.

### 3.4.2. Use Case Model

#### 3.4.2.1. Use Case Diagram



### **3.4.2.2. UC1: Registrazione**

#### **Attori coinvolti**

- utente

#### **Entry condition**

L'utente si trova sulla home page con la schermata di registrazione

#### **Flusso di eventi:**

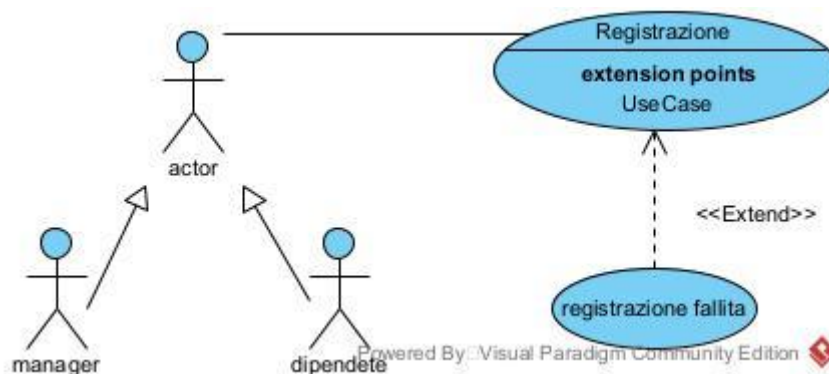
1. L'utente inserisce il proprio nome, cognome, la propria e-mail, data di nascita e una password, che va ripetuta due volte
2. L'utente invia i dati nel sistema
3. Il sistema controlla i dati, il controllo ha esito positivo
4. Il sistema invia una mail di conferma
5. L'utente accetta la conferma tramite il link inviato nella mail

#### **Exit condition**

L'utente è registrato e si trova nella home page

#### **Flussi di eventi/ Eccezioni:**

Se al punto 3 il sistema rileva credenziali non corrette, il sistema mostrerà un messaggio di errore "nome, cognome... non corretto" e ripresenterà la schermata di registrazione (UC1.1 Registrazione fallita).



### **3.4.2.3. UC2: Create Team**

#### **Attori coinvolti**

- Manager

#### **Entry Condition**

Il Manager è nella schermata di visualizzazione dei gruppi.

**Flusso di eventi:**

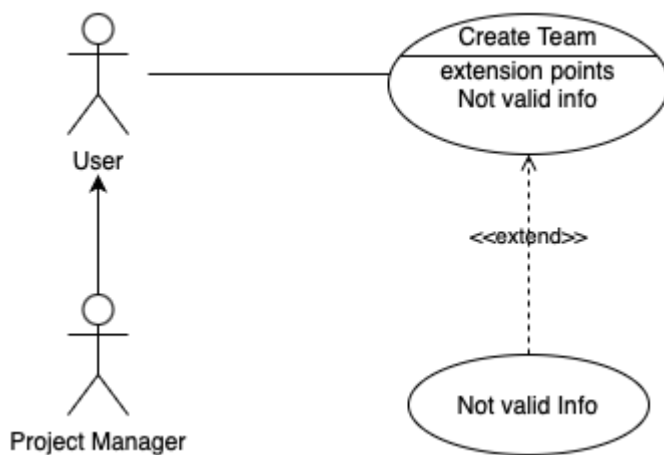
1. Il Manager seleziona l'opzione di creazione di un nuovo gruppo.
2. Il Manager inserisce i dettagli del gruppo, il nome, una descrizione, la data di consegna del progetto e un'immagine
- 2.1 Il Manager conferma la creazione del gruppo.
- 2.2 Il Manager salva e crea il nuovo gruppo.
- 2.3 Il sistema conferma che il gruppo è stato creato con successo.

**Exit Condition:**

Un nuovo gruppo è stato creato e registrato nel sistema.

**Flussi alternativi / Eccezioni:**

**UC2.1: Not valid info**, Se al punto 2.1, il Manager inserisce informazioni già esistenti o non completa i campi obbligatori, il sistema mostrerà un messaggio di errore e chiederà di inserire delle informazioni valide.



#### **3.4.2.4. UC3: Modifica info di un team**

##### **Attore:**

- Manager

**Entry Condition:** Il Manager è nella schermata di visualizzazione dei gruppi.

##### **Flusso di eventi:**

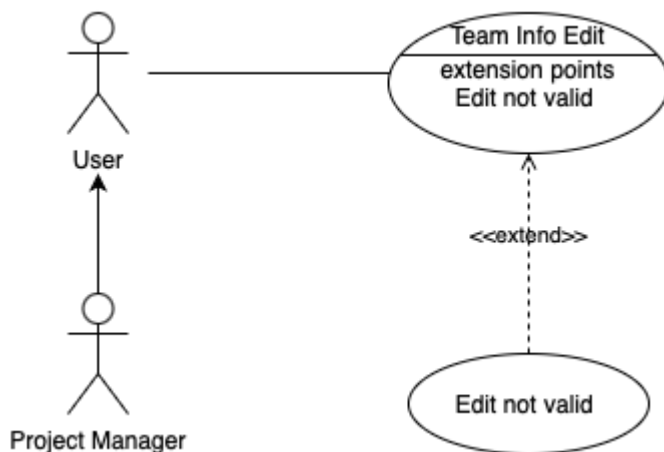
1. Il Manager seleziona il gruppo da modificare.
2. Il sistema visualizza le attuali informazioni del gruppo, nome, descrizione, data di consegna del progetto e l'immagine.
  - 2.1 Il Manager modifica le informazioni del gruppo desiderate.
  - 2.2 Il Manager conferma le modifiche.
  - 2.3 Il sistema salva le modifiche e aggiorna le informazioni del gruppo.
  - 2.4 Il sistema conferma che le informazioni sono state aggiornate con successo.

##### **Exit Condition:**

Le informazioni del gruppo sono state aggiornate nel sistema.

##### **Flussi alternativi / Eccezioni:**

**UC3.1: Edit not valid**, Se al punto 2.1 le modifiche non sono valide, il sistema notificherà l'errore al Manager.



### 3.4.2.5. UC4: AddTask

#### Attori coinvolti

- Manager.

#### Entry condition

Il manager accede alla pagina di organizzazione dei task

#### Flusso di eventi:

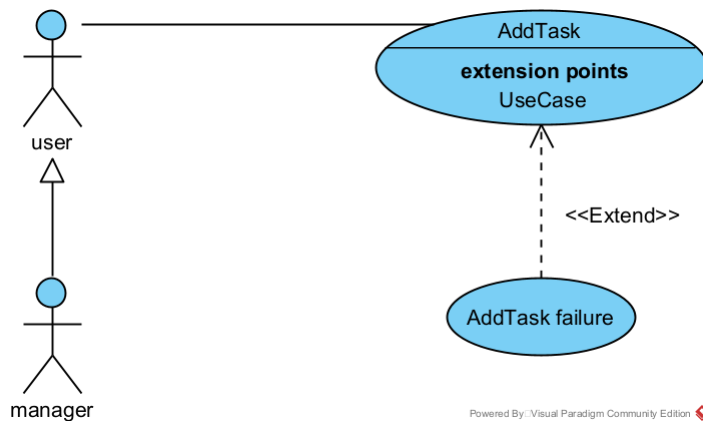
1. sceglie l'opzione di aggiungere un task alla lista
2. Il sistema
3. Il manager inserisce nome, descrizione e data di scadenza del task.
4. Il manager conferma l'inserimento
5. Il sistema controlla il nome e la data di scadenza, il controllo ha esito positivo
6. Il sistema salva i dati

#### Exit condition

Il task è salvato e il manager viene riportato alla pagina di organizzazione dove può vedere il nuovo task aggiunto

#### Flussi Alternativi / Eccezioni

Se al punto 3 il nome è già stato inserito o la data di scadenza è incoerente con le precedenti manda un messaggio di errore (UC4.1 AddTask fallita)



### **3.4.2.6. UC5: CreateTaskList**

#### **Attori coinvolti**

- Manager

#### **entry condition**

questo use case inizia quando il Manager sta guardando la schermata di organizzazione delle Task.

#### **Flusso di eventi**

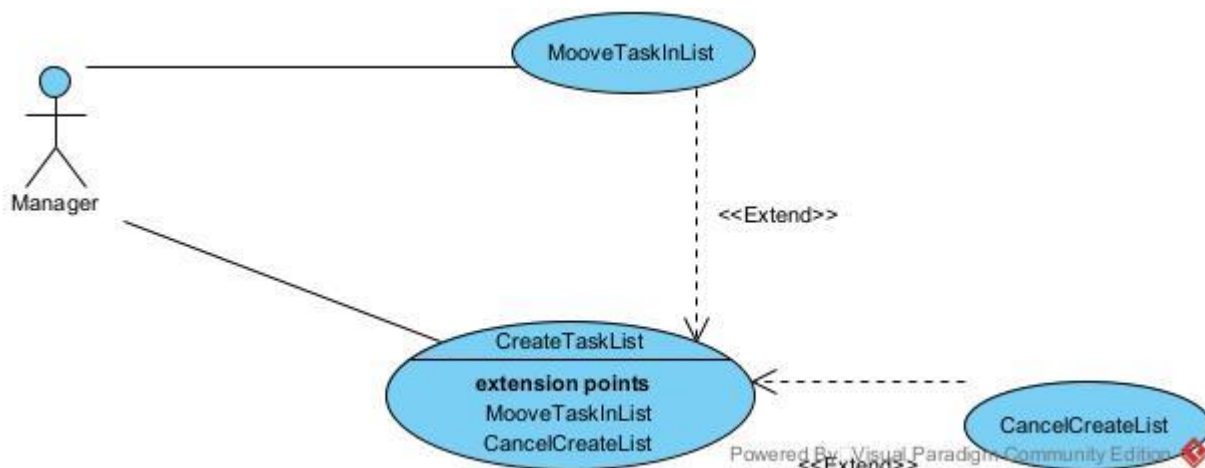
- 1) IL Manager seleziona la funzionalità di creazione di una nuova lista
- 2) il sistema porta il Manager in una pagina dove può inserire i dettagli della lista
- 3) il Manager inserisce il nome della lista di Task
- 4) il Manager sceglie una delle Task già esistenti da spostare nella lista (Opzionale)
- 5) Il Manager salva i dati inseriti
- 6) Il Sistema crea la nuova Lista di Task con i dati assegnatogli e la rende visibile nella pagina gestione Task
- 7) Il Manager esce dalla pagina di creazione Lista Task

#### **Exit condition**

questo caso d'uso termina quando la lista è creata con i dati desiderati e il Manager torna alla schermata di gestione delle Task.

#### **Flussi Alternativi/ Eccezioni**

- Durante il punto 5) il Manager può decidere di uscire, senza salvare i dati, in questo caso il punto 6) viene saltato e si va direttamente al 7) (UC5.1 CancelCreateList)
- Se il punto 4 è avvenuto, durante il punto 6) viene eseguito UC6.





### **3.4.2.7. UC6: MooveTaskInList**

#### **Attori Coinvolti**

- Manager

#### **entry condition**

questo use case inizia quando il Manager sta guardando la schermata di organizzazione delle Task.

#### **Flusso di eventi**

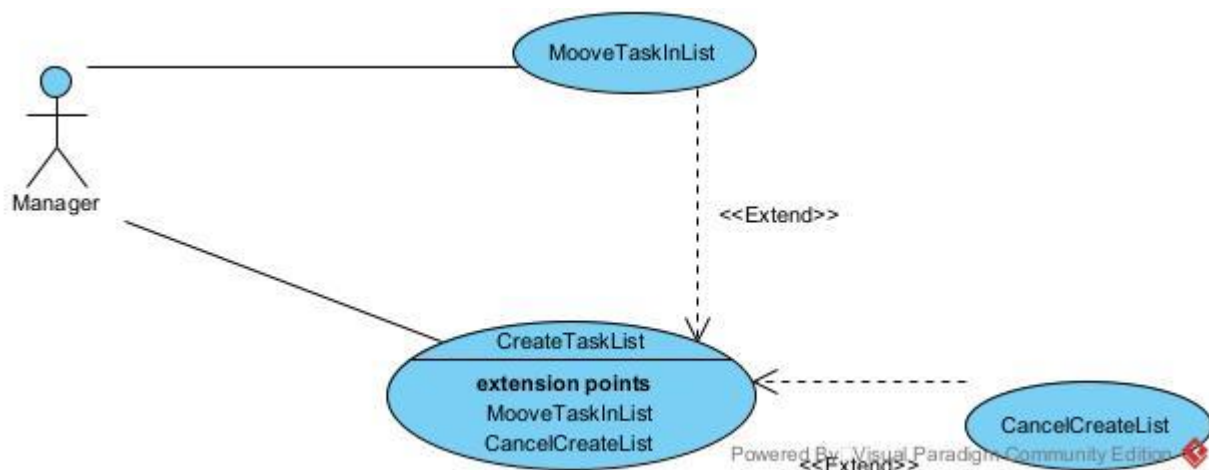
- 1) Il Manager seleziona la Task che desidera inserire in una lista e la posiziona nella lista adeguata dall'interfaccia grafica
- 2) il sistema inserisce effettivamente il task nella Lista desiderata e la elimina dalla lista che la conteneva in precedenza

#### **Exit condition**

questo caso d'uso termina quando la Task viene inserita nella Lista desiderata dal Manager.

#### **Flussi Alternativi/ Eccezioni**

- Quando questo Use Case e seguito come parte di UC 5, il punto 4) di UC 5 prende il posto del punto 1) di questo Use Case.



### **3.4.2.8. UC7: TaskDelivery**

#### **Attori Coinvolti**

- Team Member

#### **Entry condition**

Il team member è autenticato e si trova nella pagina di dettaglio del task

#### **Flusso di eventi**

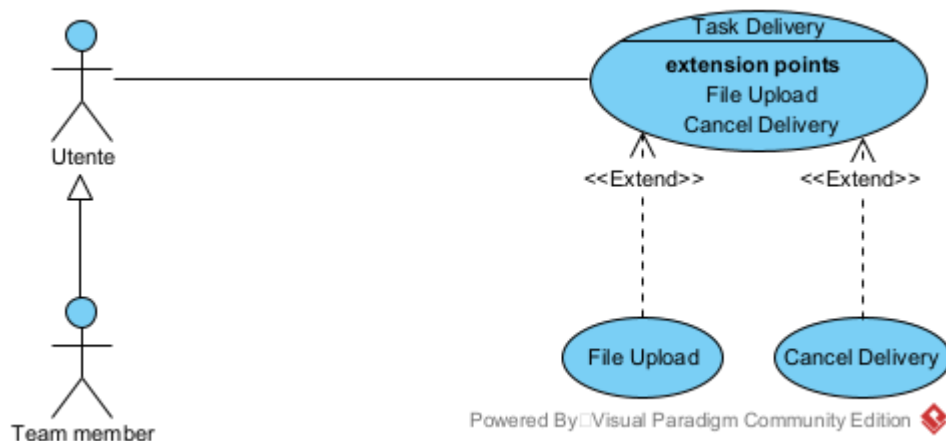
- 1) L'utente visualizza i dettagli del task
- 2) Caricamento eventuale di file allegati (opzionale)
- 3) L'utente seleziona l'opzione per segnalare il completamento del task
- 4) Il sistema chiede conferma del completamento
- 5) L'utente conferma il completamento
- 6) Il sistema aggiorna lo stato del task e le statistiche di progetto

#### **Exit condition**

Il task è stato marcato come completato.

#### **Flussi Alternativi/ Eccezioni**

- Se al punto 2) si sceglie di caricare dei file allegati viene eseguito UC 7.1 (UC 7.1 File Upload)
- Se al punto 5) l'utente annulla la conferma, il sistema mantiene lo stato corrente del task (UC 7.2 CancelDelivery)



### **3.4.2.9. UC8: StartTimer**

#### **Attori Coinvolti**

- Team Member

#### **Entry condition**

Il team member è autenticato e si trova nella pagina delle prestazioni

#### **Flusso di eventi**

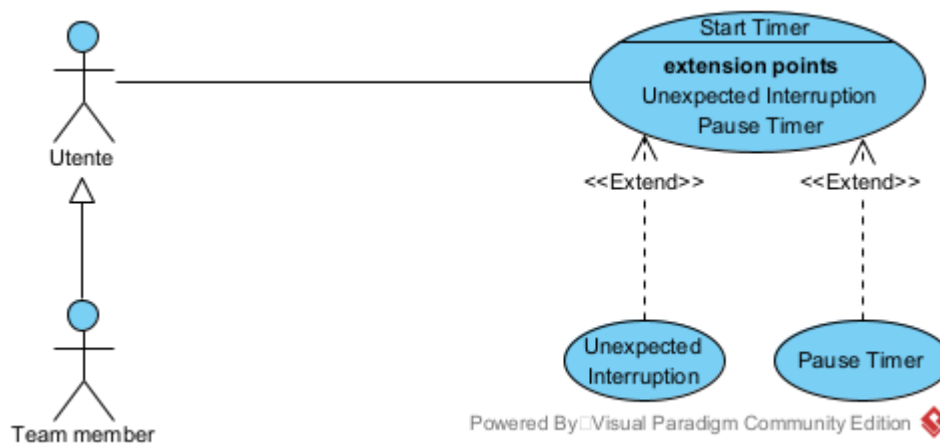
- 1) L'utente seleziona il pulsante di avvio del timer nella pagina delle prestazioni
- 2) Il sistema avvia il conteggio del tempo
- 3) Il sistema mostra il tempo trascorso aggiornandolo in tempo reale
- 4) L'utente può selezionare il pulsante di pausa quando necessario
- 5) Il sistema interrompe il conteggio del tempo mantenendo il totale

#### **Exit condition**

Il timer è attivo e sta tracciando il tempo di lavoro.

#### **Flussi Alternativi/ Eccezioni**

- Se al punto 4 l'utente mette in pausa, il sistema sospende il conteggio fino alla ripresa (UC 8.1 PauseTimer)
- Se durante il tracking si verifica una disconnessione, il sistema salva il tempo registrato fino a quel momento (UC 8.2 UnexpectedInterruption)



### **3.4.2.10. UC9: SendMessage**

#### **Attori Coinvolti**

- Utente (Manager o Team member)

entry condition

questo use case inizia quando l'Utente sta scorrendo l'elenco di chat room del progetto.

#### **Flusso di eventi**

- 1) L'utente sceglie la chat room, di gruppo o individuale, in cui deve inviare un messaggio
- 2) il sistema apre la schermata relativa alla chat e rende visibile i messaggi inviati in precedenza dagli Utenti
- 3) L'utente inserisce il corpo del suo messaggio e preme il pulsante di invio
- 4) il sistema invia il messaggio agli altri membri della chat e li notifica del suo arrivo

#### **Exit condition**

questo caso d'uso termina quando il messaggio è stato inviato con successo dal sistema.

#### **Flussi Alternativi/ Eccezioni**

- Al punto 3) l'utente può smettere di scrivere e cambiare pagina senza premere il pulsante di invio, in quel caso il testo del messaggio viene eliminato e il caso d'uso termina. (UC9.1 CancellMessage)

#### **Requisiti Speciali**

- Il sistema deve inviare il Messaggio con successo in un tempo di 10 secondi dal comando dell'utente

