

Progettazione di un sistema che gestisca via web  
delle Domande di trasferimento

Gruppo 3

Riccardo Italia (PM)  
Matteo Busanelli (Librarian)  
Marco Carlesi  
Erica Gazzani

12 gennaio 2004

# Indice

<b>Glossario</b>	<b>1</b>
<b>1 Casi d'uso</b>	<b>3</b>
1.1 Scenario 1: Richiesta di trasferimenti . . . . .	3
1.2 Scenario 2: Valutazione delle richieste . . . . .	4
1.3 Scenario 3: Aggiornamento del sistema da parte del Ccds . . . . .	4
1.4 Scenario 4: Il Ccds modifica le regole della domanda di trasferimento . . . . .	5
<b>2 Descrizione dei Diagrammi dei Casi d'uso</b>	<b>7</b>
2.1 Attore Studente . . . . .	7
2.2 Attore Ccds . . . . .	7
2.2.1 Ccds: Valuta Domanda . . . . .	7
2.2.2 Ccds: Crea Carriera . . . . .	8
2.2.3 Ccds: Modifica Regole . . . . .	8
<b>3 Diagrammi delle classi</b>	<b>11</b>
3.1 Diagramma delle classi concettuale . . . . .	11
3.2 Diagramma delle classi di specifica . . . . .	13
3.2.1 Descrizione delle classi . . . . .	13
3.2.2 Design patterns . . . . .	15
<b>4 Diagrammi di sequenza</b>	<b>17</b>
4.1 Valuta domanda . . . . .	17
4.2 Modifica regole . . . . .	18
4.3 Ccds . . . . .	19
4.4 Studente . . . . .	20
<b>5 Diagrammi delle attività</b>	<b>23</b>
<b>6 Pianificazione delle Attività</b>	<b>27</b>
6.1 RUP (Rational Unified Process) . . . . .	27
6.1.1 Riassunto delle attività svolte da ciascuna risorsa . . . . .	28
6.1.2 Diagramma di Gantt . . . . .	28
6.1.3 Workflow . . . . .	29
<b>Appendice A</b>	<b>31</b>
<b>Appendice B</b>	<b>35</b>

**Bibliografia**

**43**

# Elenco delle figure

2.1	Diagramma dei Casi D'uso . . . . .	9
3.1	Diagramma delle classi concettuale . . . . .	12
3.2	Diagramma delle classi di specifica . . . . .	16
4.1	Diagramma di sequenza - Valuta domanda . . . . .	18
4.2	Diagramma di sequenza - Modifica regole . . . . .	19
4.3	Diagramma di sequenza - Ccads . . . . .	20
4.4	Diagramma di sequenza - Studente . . . . .	21
5.1	Diagramma delle attività generale . . . . .	24
5.2	Diagramma delle attività del Ccads . . . . .	25
5.3	Diagramma delle attività dello Studente . . . . .	26
6.1	Diagramma distribuzione tempi per attività e risorsa . . . . .	28
6.2	Diagramma di Gantt . . . . .	29

# Glossario

Per prima cosa forniamo una breve descrizione dei principali termini utilizzati nella fase di progettazione ed analisi che sono derivati da un'attenta analisi effettuata sui requisiti iniziali e gli scenari individuati insieme agli altri gruppi

## Elenco termini

- **Carriera:**  
Indica la storia di uno studente quindi: gli esami sostenuti, gli esami che deve sostenere ed i relativi voti.
- **Crediti:**  
Un valore che indica lo sforzo, in quantità di tempo, che uno studente deve utilizzare per poter superare l'esame.
- **Consiglio del corso di studi:**  
Gruppo di professori di un corso di studi. É un organo che ha diversi compiti (come decisioni riguardanti i piani di studi, i tirocini, le convenzioni, etc.) tra i quali quello di valutare le domande di trasferimento e decidere quelle che devono essere accolte (atto che comporta la creazione della carriera per quel determinato studente) e quelle che devono essere respinte. Il Consiglio del corso di studi è anche l'unico utente che ha il permesso di modificare le regole relative alle domande di trasferimento.  
*Sinonimi:* CCdS.
- **Esame:**  
Prova conclusiva di un insegnamento.
- **Regola:**  
Metodo utilizzato per controllare i crediti consentiti all'interno di una domanda di trasferimento.
- **Sistema:**  
Insieme di servizi, accessibili via Web, progettati per gli utenti, al fine di gestire le domande di trasferimento.
- **Identificazione:**  
Metodo con cui gli utenti possono essere riconosciuti dal sistema.
- **Studente:**  
Persona iscritta ad un'università e che vuole effettuare il trasferimento.

## ELENCO DELLE FIGURE

---

- **Utenti:**  
Studenti o Consiglio del corso di studi.
- **Domanda di trasferimento:**  
Modulo con cui lo studente richiede il trasferimento da una un'università (nazionale) a quella di Bologna. Nella domanda lo studente indica anche gli esami che vuole siano convertiti.

# Capitolo 1

## Casi d'uso

### 1.1 Scenario 1: Richiesta di trasferimenti

- Obbiettivi:

Lo studente compila la richiesta di trasferimento che viene inserita nell'elenco delle richieste da valutare

- Descrizione:

- a. Lo studente si collega al sito.
- b. Lo studente si identifica.
- c. Il sistema verifica l'identificazione dello studente.
- d. Il sistema verifica che lo studente non abbia già compilato la domanda.
- e. Lo studente compila il modulo per la domanda.
- f. Il sistema controlla che la domanda sia stata compilata correttamente.
- g. Il sistema calcola i crediti inseriti dallo studente.

- Alternativa 1: Lo studente non è riconosciuto

Al passo c il sistema non riconosce lo studente.  
Si deve permettere allo studente di riprovare.

- Alternativa 2: Lo studente ha già compilato la domanda.

Al passo d il sistema verifica che lo studente ha già compilato la domanda.  
Non gli si permette di compilarla di nuovo.

- Alternativa 3: Lo studente non ha compilato correttamente la domanda.

Al passo f il sistema si accorge che lo studente non ha compilato tutti i campi correttamente.  
Si deve permettere allo studente di compilare i campi mancanti o non corretti.

- Alternativa 4: Lo studente ha più di 194 crediti

Al passo g il sistema farà notare al Ccds che questo studente ha più crediti di quelli consentiti

- Priorità:  
Alta
- Ipotesi:  
Il sistema deve avere un elenco di domande di trasferimento da valutare.
- Precondizioni:  
Lo studente deve essere iscritto all'università.
- Postcondizioni:  
La domanda è compilata correttamente.

## 1.2 Scenario 2: Valutazione delle richieste

- Obbiettivi:  
  
Il consiglio di corso di studi si collega per prelevare le domande e valutarle.
- Descrizione:
  - a. Il Ccads si collega al sistema e si identifica.
  - b. Il sistema verifica l'identificazione del Ccads.
  - c. Il Ccads accede all'elenco delle richieste.
  - d. Esamina la richiesta dello studente.
  - e. Elabora la nuova carriera dello studente
- Alternativa 1: Il Ccads non è riconosciuto.  
Al passo b il sistema non riconosce il Ccads.  
Si deve permettere al Ccads di riprovare.
- Priorità:  
Alta
- Ipotesi:  
Il sistema possiede un elenco delle domande di trasferimento.
- Precondizioni:  
Solo il Ccads può esaminare l'elenco delle domande
- Postcondizioni:  
Il Ccads valuta le domande di trasferimento.

## 1.3 Scenario 3: Aggiornamento del sistema da parte del Ccads

- Obbiettivi:  
  
Il Ccads si collega al sistema aggiornandolo con la nuova carriera dello studente. Il sistema in seguito calcola i crediti riconosciuti allo studente informandolo sull'anno in cui sarà iscritto e sugli esami che potrebbe dover integrare. Infine avvisa lo studente che la sua domanda è stata esaminata.



#### 1.4. SCENARIO 4: IL CCDS MODIFICA LE REGOLE DELLA DOMANDA DI TRASFERIMENTO

---

- Descrizione:
  - a. Il Ccds si collega al sistema e si identifica.
  - b. Il sistema verifica l'identificazione del Ccds.
  - c. Il Ccds aggiorna il sistema con la nuova carriera dello studente.
  - d. Il sistema calcola i crediti dello studente.
  - e. Il sistema calcola a quale anno iscrivere lo studente e che esami dovrà eventualmente integrare.
  - f. Il sistema avvisa lo studente che è stata esaminata la sua domanda.
- Alternativa 1: Il Ccds non è riconosciuto.  
Al passo b il sistema non riconosce il Ccds.  
Si deve permettere al Ccds di riprovare.
- Priorità:  
Media
- Ipotesi:  
Il Ccds deve aver valutato le domande di trasferimento.
- Precondizioni:  
Il Ccds ha creato la nuova carriera dello studente.
- Postcondizioni:  
Il sistema ha calcolato i crediti dello studente e lo avvisa che la sua domanda è stata valutata.

#### 1.4 Scenario 4: Il Ccds modifica le regole della domanda di trasferimento

- Obbiettivi:

Il Ccds si collega al sistema modificando le specifiche della domanda di trasferimento.
- Descrizione:
  - a. Il Ccds si collega al sistema e si identifica.
  - b. Il sistema verifica l'identificazione del Ccds.
  - c. Il Ccds apporta delle modifiche alle specifiche della domanda di trasferimento.
- Alternativa 1: Il Ccds non è riconosciuto.  
Al passo b il sistema non riconosce il Ccds.  
Si deve permettere al Ccds di riprovare.
- Priorità:  
Media
- Ipotesi:  
Devono essere cambiate alcune specifiche per la domanda di trasferimento.

## CAPITOLO 1. CASI D'USO

---

- Precondizioni:  
Il Ccds ha pronte tutte le modifiche da apportare alla domanda.
- Postcondizioni:  
Il sistema è aggiornato con la nuova domanda di trasferimento.

## Capitolo 2

# Descrizione dei Diagrammi dei Casi d'uso

In questo diagramma si vuole mostrare come e quali siano le interazioni fra i due principali attori utenti (Studente e Ccads) e il sistema. Per mezzo poi di relazioni di inclusione fra casi d'uso si vuole dare una migliore e più chiara organizzazione alle azioni che i due attori possono intraprendere nei confronti del sistema.[vedi figura 2.1]

### 2.1 Attore Studente

Lo studente essenzialmente potrà solamente interagire col sistema per *compilare la domanda* e restare poi in attesa di una risposta (via email) quando il Ccads l'avrà approvata. Per compilare la domanda però egli deve preventivamente *Accedere* al sistema in modo tale da essere riconosciuto da quest'ultimo. Ecco allora emergere una chiara relazione di **inclusione** (include) fra il caso d'uso *Accedi* e quello *Compila Domanda*.

### 2.2 Attore Ccads

Il Ccads invece può interagire col sistema in molteplici modi previo sempre un momento di accesso al sistema per il riconoscimento. Ne segue che ogni caso d'uso inerente al Ccads *include* l'*Accedi* come punto di partenza per ogni possibile interazione col sistema. Le azioni principali che il Ccads compirà per mezzo del servizio progettato dovranno essere essenzialmente 3:

1. *Valuta Domanda*
2. *Crea Carriera*
3. *Modifica Regole*

#### 2.2.1 Ccads: Valuta Domanda

*Valuta Domanda* (1) *include* necessariamente un sotto caso che abbiamo individuato essere *Consulta Elenco* delle domande. In questa fase il Ccads, previo

Accesso al sistema e riconoscimento, dovrà richiedere per esempio la stampa di tutte le domande di trasferimento che il sistema ha accumulato nell'*Elenco Domande*.

### 2.2.2 Ccads: Crea Carriera

Altra prerogativa del Consiglio di corso di studi quella di *Creare la Carriera* (2) degli studenti successivamente alla valutazione delle domande in sede di consiglio. Ancora una volta il Ccads dovrà preventivamente *Accedere* e farsi riconoscere dal sistema.

*Crea Carriera* include necessariamente un altro caso d'uso concernente il *Calcolo Anno* a cui lo studente verrà iscritto. Il caso denominato *Informa Studente*, invece, rappresenta il momento in cui il *sistema* avvertirà lo studente tramite email ed include a sua volta il caso *Crea Carriera* in quanto può avvenire solo dopo quest'ultimo.

Si noti che queste ultime due azioni (*Calcola Anno* e *Informa Studente*) non prevedono interazione dell'attore Ccads e sono esclusivamente di competenza del sistema che le effettuerà in modo automatico.

### 2.2.3 Ccads: Modifica Regole

Infine, l'ultima operazione che può essere effettuata una tantum dal Ccads quella di *Modifica Regole* (3). In questo caso, dopo che il Ccads ha effettuato l'accesso al servizio, richiamerà quelle che sono le *regole* usate per il calcolo dei crediti e l'assegnazione dell'anno di iscrizione per modificarne i parametri.

## 2.2. ATTORE CCDS

---

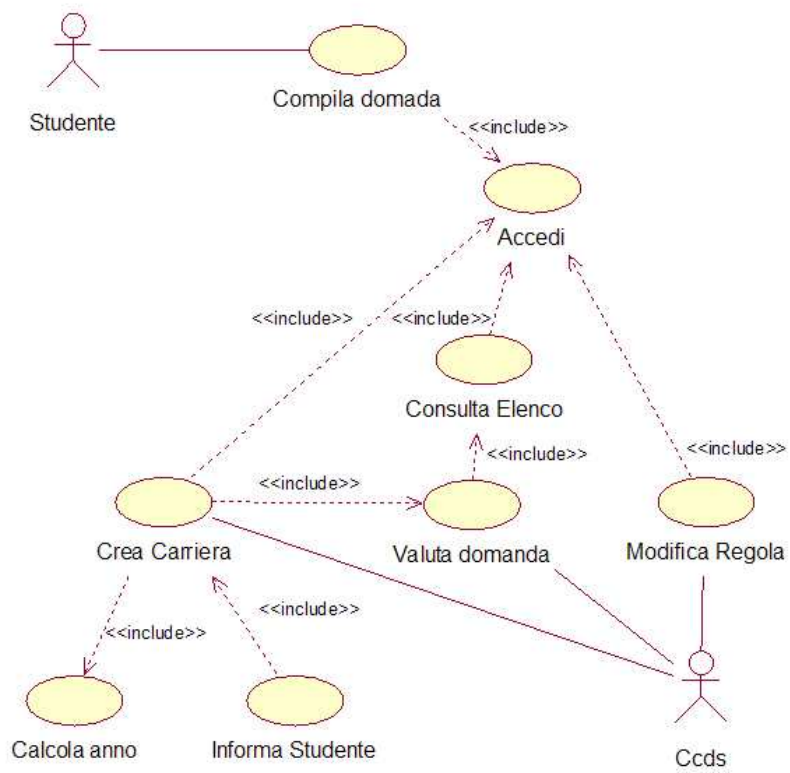


Figura 2.1: Diagramma dei Casi D'uso



## Capitolo 3

# Diagrammi delle classi

Partendo dal diagramma delle classi concettuale si è cercato di individuare le classi principali. Ampliando ed integrando il diagramma delle classi concettuale con l'inserimento degli attributi e le operazioni delle classi e con l'inserimento di design pattern, si è giunti alla costruzione del diagramma delle classi di specifica.[1]

### 3.1 Diagramma delle classi concettuale

Nel diagramma delle classi concettuale [vedi figura 3.1] sono presenti le seguenti classi: Sistema, Utente, Studente, Ccds, Domanda, Regola. Queste classi definiscono la struttura base del servizio che si vuole progettare.

La classe Sistema controllerà gli accessi al servizio di trasferimento e si occuperà di informare gli studenti che la loro domanda di trasferimento è stata valutata.

La classe Utente identifica gli utenti che possono accedere al servizio di trasferimento, nel nostro caso gli utenti si suddividono in studenti e Ccds.

La classe Studente contiene le informazioni relative allo studente che compila una domanda di trasferimento.

La classe Ccds compie diverse azioni nel nostro servizio: valuta le domande di trasferimento, aggiorna il sistema memorizzando i dati degli studenti che richiedono il trasferimento e talvolta può decidere di modificare le regole di trasferimento.

La classe Domanda rappresenta una singola domanda di trasferimento compilata da uno studente.

Infine la classe Regola contiene i vincoli che una domanda di trasferimento deve rispettare.

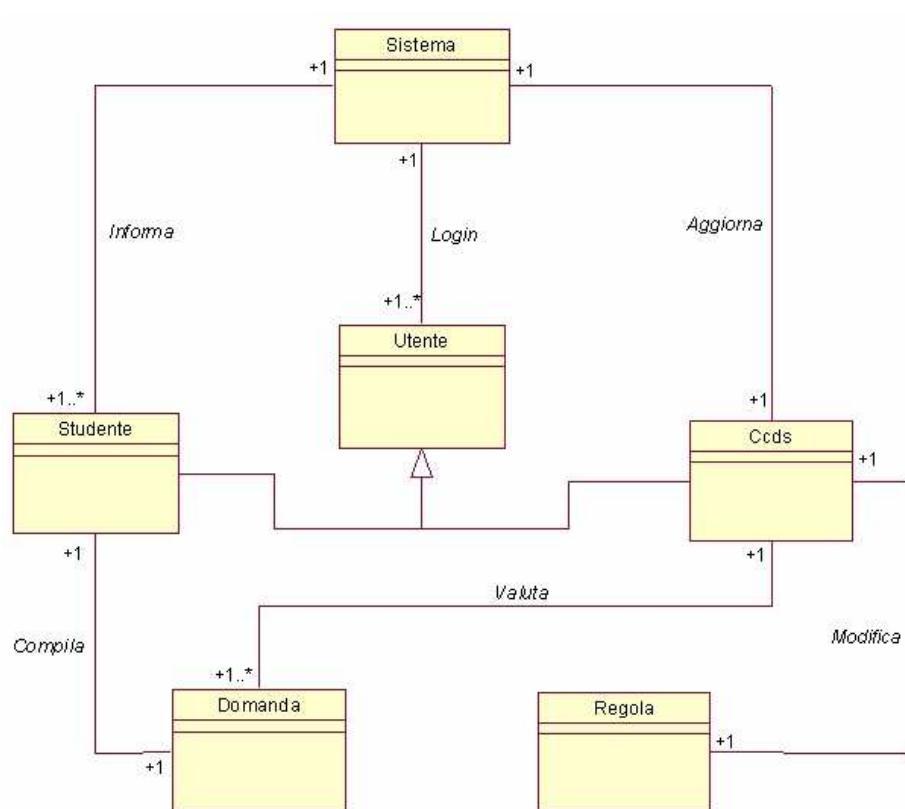


Figura 3.1: Diagramma delle classi concettuale



## 3.2. DIAGRAMMA DELLE CLASSI DI SPECIFICA

---

### 3.2 Diagramma delle classi di specifica

Partendo dal diagramma delle classi concettuale sono state introdotte nuove classi, gli attributi e le operazioni di ciascuna classe arrivando alla creazione del diagramma delle classi di specifica.[vedi figura 3.2]

#### 3.2.1 Descrizione delle classi

##### **Sistema**

###### Descrizione:

Questa classe è la più importante del servizio di trasferimento, rappresenta il punto di ingresso per le operazioni compiute al suo interno.

###### Operazioni:

*accedi(username:string,pwd:string):boolean* - Questa operazione permette ad un utente di accedere ai servizi offerti, prende in input due stringhe (username e pwd) e restituisce true se l'utente viene riconosciuto dal sistema.

*informaStudente(studente:Studente,uid:unsigned long)* - Attraverso questa operazione il sistema informa lo studente, tramite e-mail, che la domanda di trasferimento è stata valutata dal Ccds.

##### **Utente**

###### Descrizione:

La classe Utente viene definita dagli attributi username e pwd vengono utilizzati dalla classe Sistema per identificare l'utente che intende accedere al servizio offerto.

###### Operazioni:

Questa classe non possiede metodi.

##### **Studente**

###### Descrizione:

La classe studente viene definita dagli attributi che seguono: uid (numero che identifica univocamente uno studente, ad esempio matricola), nome, cognome, email, carriera, domanda.

###### Operazioni:

*compilaDomanda(domanda:Domanda,uid:unsigned long):Domanda* - Attraverso questa operazione lo studente compila la domanda di trasferimento, prende in input un nuovo oggetto di tipo Domanda e l'uid dello studente e restituisce la domanda compilata.

##### **Ccds**

###### Descrizione:

La classe Ccds possiede gli attributi: membriccds (indica i nomi dei membri del Ccds), data\_ultima\_riunione, data\_prossima\_riunione.

###### Operazioni:

*valutaDomanda(domanda:Domanda,uid:unsigned long):ListaEsami* - Questo metodo prende una domanda di trasferimento e l'uid dello studente e restituisce la lista degli esami che gli vengono riconosciuti dopo il trasferimento.

*creaCarriera(lista\_esami\_convalidati:ListaEsami,uid:unsigned long):Carriera* - Attraverso questo metodo il Ccds crea la nuova carriera dello studente, nella quale sono indicati gli esami riconosciuti dopo il trasferimento.

### **ElencoDomande**

Descrizione:

La classe ElencoDomande possiede l'attributo `n_domande` che indica quante domande sono pronte per essere valutate dal Ccda.

Operazioni:

*inserisciDomanda(domanda:Domanda,uid:unsigned long)* - Inserisce una domanda di trasferimento nell'elenco delle domande di trasferimento.

*prelevaDomanda(domanda:Domanda,uid:unsigned long)* - Preleva una domanda di trasferimento dall'elenco delle domande di trasferimento.

### **Domanda**

Descrizione:

La classe Domanda è caratterizzata dagli attributi: `uid`, `data`, `lista_esami_sostenuti`, `lista_esami_corrispondenti`, `crediti_eccesso`. Questa classe rappresenta la domanda di trasferimento che lo studente compila e che il sistema inserisce nell'elenco delle domande da valutare.

Operazioni:

*calcolaCrediti(lista\_esami\_sostenuti:ListaEsami,regola:Regola):int* - L'operazione `calcolaCrediti()` restituisce, in base alle regole di trasferimento, il numero di crediti in possesso dello studente che compila la domanda.

### **ListaEsami**

Descrizione:

Nella classe ListaEsami sono mantenuti gli esami corrispondenti ad uno studente. Questa classe è caratterizzata dagli attributi: `n_esami`, `tot_crediti`, `crediti_inf`, `crediti_mat`, `crediti_fis`, `crediti_liberi`.

Operazioni:

*inserisciEsame(esame:Esame,codice\_corso:int)* - Inserisce un esame nella lista degli esami.

*cancellaEsame(esame:Esame,codice\_corso:int)* - Cancella un esame dalla lista degli esami.

### **Esame**

Descrizione:

La classe Esame contiene tutte le informazioni relative ad un esame. Essa contiene gli attributi: `codice_corso`, `nome`, `data`, `area`, `crediti`, `voto`.

Operazioni:

Questa classe non possiede metodi.

### **Carriera**

Descrizione:

La classe Carriera contiene tutte le informazioni relative agli esami convalidati dopo il trasferimento ad uno studente. Questa classe possiede gli attributi: `uid`, `lista_esami_convalidati`, `data`, `anno_iscrizione`.

Operazioni:

*salvaCarriera()* - Con questo metodo viene memorizzata la nuova carriera dello studente nel sistema.

*calcolaAnno(lista\_esami\_convalidati:ListaEsami):int* - Questo metodo calcola a quale anno lo studente verrà iscritto dopo il trasferimento

## 3.2. DIAGRAMMA DELLE CLASSI DI SPECIFICA

---

### **Regola**

#### Descrizione:

La classe Regola contiene tutti i vincoli che servono al Ccds per valutare una domanda di trasferimento. Gli attributi di questa classe sono: `crediti_per_anno`, `n_max_crediti`, `crediti_inf`, `crediti_mat`, `crediti_fis`, `crediti_liberi`, `crediti_laurea`.

#### Operazioni:

*applicaRegole(domanda:Domada,uid:unsigned long)* - Attraverso questo metodo il Ccds applica le regole alla domanda di trasferimento dello studente.

### **3.2.2 Design patterns**

Analizzando il diagramma delle classi si è deciso di inserire il design pattern *Singleton*. Il *Singleton* viene utilizzato quando un'applicazione necessita di una, e una sola, istanza di un oggetto. Nel servizio progettato le classi Sistema, che controlla gli accessi e informa gli studenti, e la classe Ccds, che valuta le domande e modifica le regole di trasferimento, devono essere istanziate una sola volta e per questo motivo abbiamo adottato il *Singleton* per queste classi.[2]

CAPITOLO 3. DIAGRAMMI DELLE CLASSI

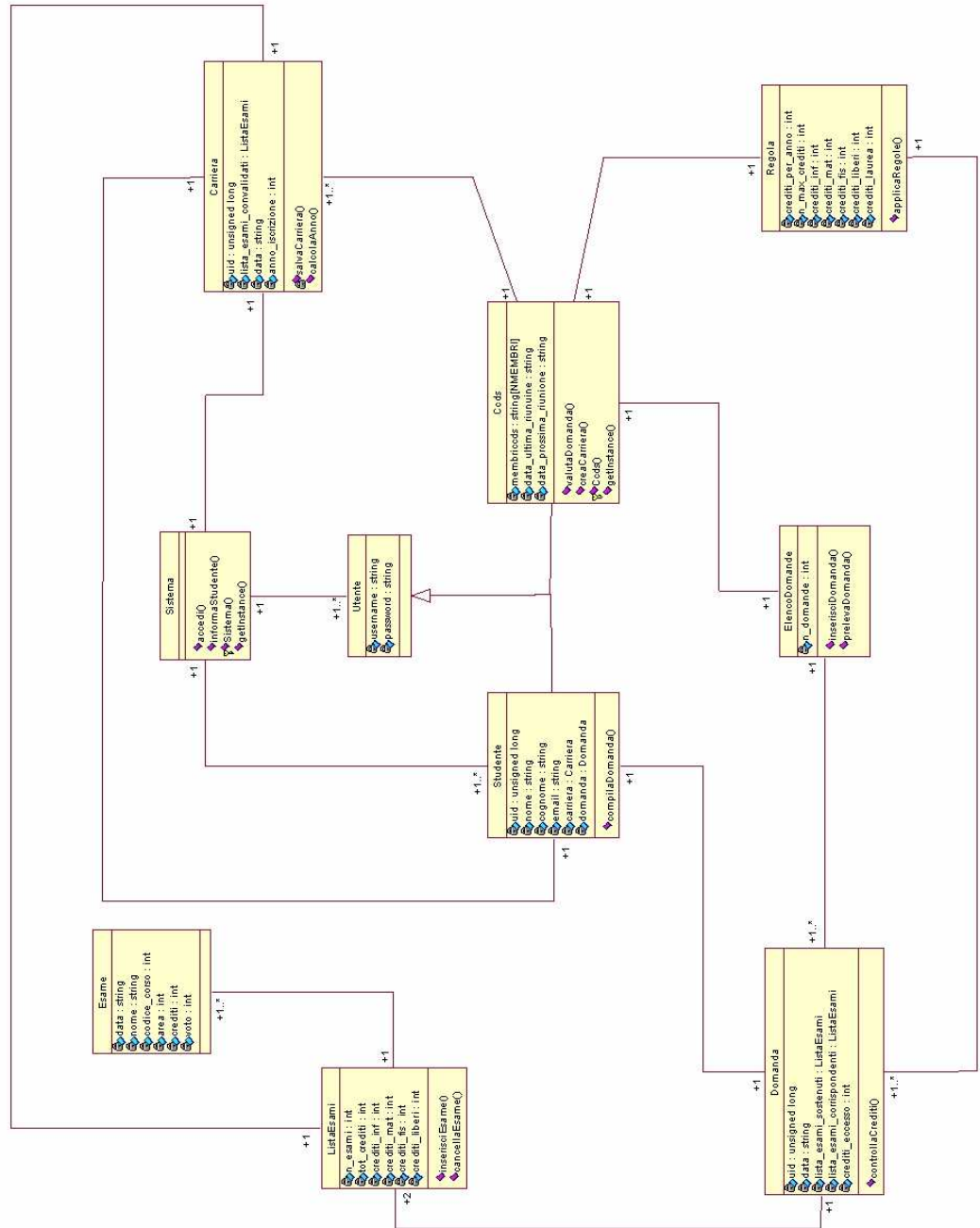


Figura 3.2: Diagramma delle classi di specifica

## Capitolo 4

# Diagrammi di sequenza

I diagrammi di sequenza descrivono in che modo gli attori e le classi del sistema progettato interagiscono tra loro.[3]

### 4.1 Valuta domanda

Nel diagramma [vedi figura 4.1] vengono mostrati come *Studente*, *Ccads* e *Sistema* interagiscono tra loro.

Lo *Studente* compila la domanda di trasferimento e la invia al *Sistema*. Il *Sistema* controlla che la domanda sia stata compilata correttamente ed in caso contrario avverte lo *Studente* che deve ricompilare la domanda. Se invece la domanda è stata compilata correttamente il *Sistema* controlla le regole di trasferimento ed inserisce la domanda nell'elenco delle domande di trasferimento da valutare. Il *Ccads* esamina le domande di trasferimento prelevandole dall'elenco delle domande di trasferimento da valutare. Dopo aver fatto valutare dal *Ccads* le domande di trasferimento il *Sistema* calcola i crediti posseduti dallo studente e determina a quale anno verrà iscritto. A questo punto la domanda viene eliminata dall'elenco delle domande di trasferimento. Infine il *Sistema* avvisa, tramite e-mail, lo studente che la sua domanda di trasferimento è stata valutata.

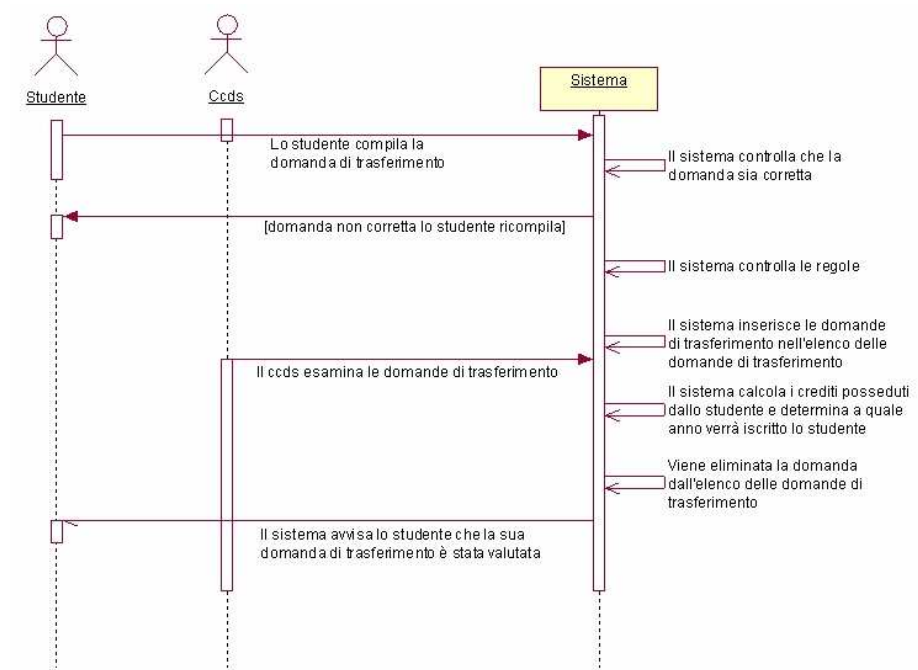


Figura 4.1: Diagramma di sequenza - Valuta domanda

## 4.2 Modifica regole

In questo diagramma [vedi figura 4.2] viene illustrato come il Ccads interagisce con il Sistema per modificare le regole di trasferimento.

Il Ccads accede al Sistema attraverso il metodo *accedi()*, il Sistema controlla che username e password corrispondano; nel caso siano errati invia un messaggio di errore al Ccads; nel caso invece che la convalida abbia successo il Ccads modifica le regole di trasferimento, il Sistema salva le nuove regole ed invia un messaggio al Ccads per confermare l'avvenuta modifica delle regole.

### 4.3. CCDS

---

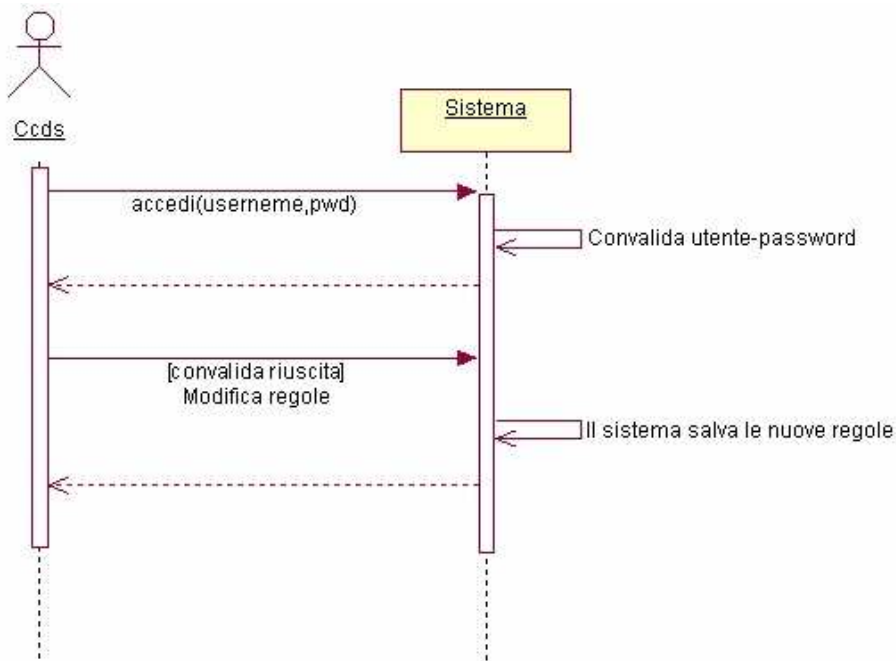


Figura 4.2: Diagramma di sequenza - Modifica regole

### 4.3 Ccdfs

In questo diagramma [vedi figura 4.3] viene illustrato come il *Ccdfs* interagisce con la classe *Sistema*, la classe *Ccdfs* e la classe *Carriera* per valutare una domanda di trasferimento e creare la nuova carriera dello studente.

Il *Ccdfs* attraverso il metodo `accedi(username,pwd)` si identifica al *Sistema*, in caso di esito negativo dell'identificazione viene inviato al *Ccdfs* un messaggio di errore. Nel caso, invece, in cui il *Ccdfs* sia stato identificato viene chiamato il metodo `valutaDomanda(domanda,uid):ListaEsami` della classe *Ccdfs*. Dopo aver valutato la domanda il *Ccdfs* crea la nuova carriera dello studente attraverso il metodo `creaCarriera(lista esami convalidati,uid):Carriera` sempre della classe *Ccdfs*. A questo punto la classe *Ccdfs* utilizza il metodo `calcolaAnno(lista esami convalidati):int` della classe *Carriera* per calcolare a quale anno verrà iscritto lo studente. Infine la classe *Carriera* invoca il metodo `informaStudiante(studiante,uid)` della classe *Sistema* per informare lo studente che la sua domanda di trasferimento è stata valutata.

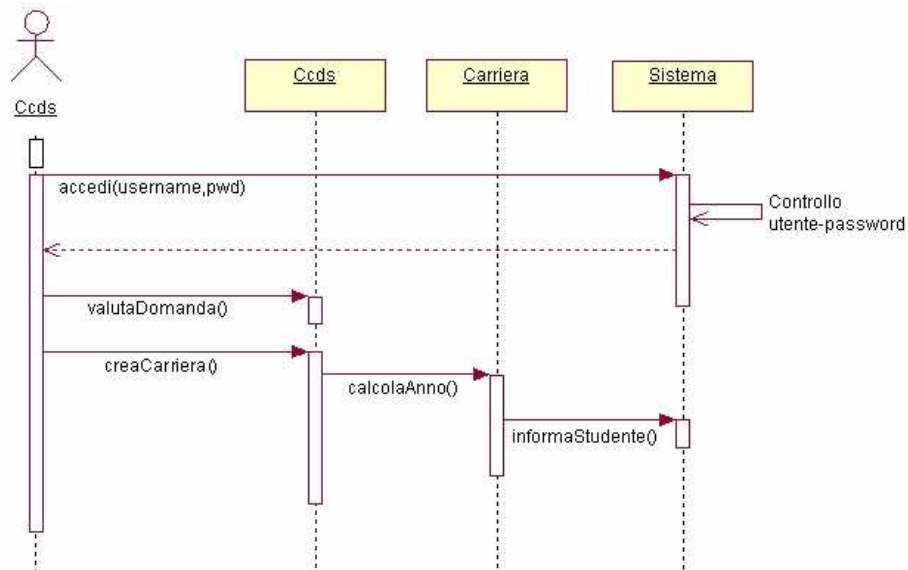


Figura 4.3: Diagramma di sequenza - CcDs

## 4.4 Studente

In questo diagramma [vedi figura 4.4] viene mostrato come lo Studente interagisce con la classe Sistema, la classe Studente, la classe Domanda, la classe Regola e la classe ElencoDomande per compilare la domanda di trasferimento che verrà inserita nell'elenco delle domande di trasferimento da valutare.

Lo Studente attraverso il metodo `accedi(username,pwd)` si identifica al Sistema, in caso di esito negativo dell'identificazione viene inviato allo Studente un messaggio di errore. In caso di identificazione avvenuta con successo lo Studente chiama il metodo `compilaDomanda(domanda,vid):Domanda` della classe Studente e compila la domanda di trasferimento. A questo punto la classe Studente invoca il metodo `calcolaCrediti(lista esami sostenuti,vid):int` della classe Domanda per conoscere i crediti in possesso dello Studente. La classe Domanda attraverso il metodo della classe Regola `applicaRegole(domanda,uid)` controlla che la domanda rispetti i vincoli imposti e utilizzando il metodo `inserisciDomanda(domanda,uid)` della classe ElencoDomande inserisce la domanda nell'elenco delle domande che il CcDs dovrà valutare.



#### 4.4. STUDENTE

---

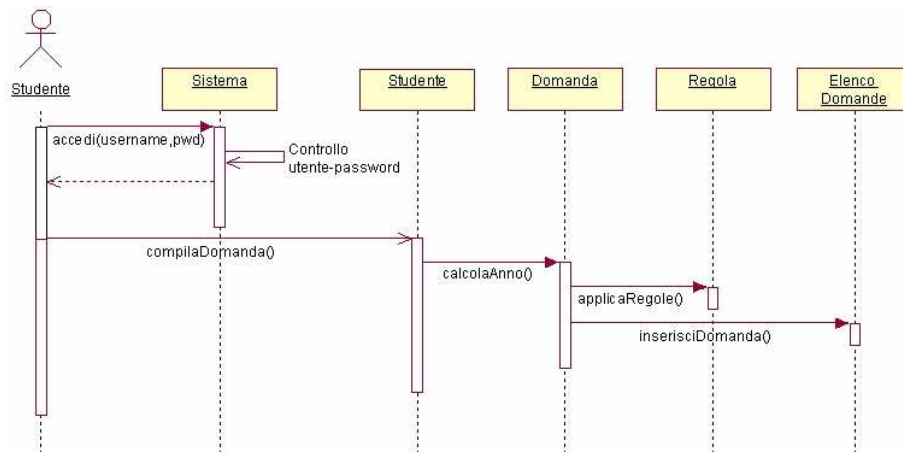


Figura 4.4: Diagramma di sequenza - Studente



## Capitolo 5

# Diagrammi delle attività

I diagrammi delle attività sono utilizzati per modellare il flusso fra diversi componenti di un sistema a oggetti. Con essi si possono quindi illustrare le funzionalità del sistema e come sono utilizzate.[4]

### Valutazione domanda di trasferimento

Nel diagramma che segue [vedi figura 5.1] vengono indicate quali sono le azioni da svolgere per valutare la domanda di trasferimento. L'utente accede al sistema tramite identificazione, a questo punto in base al tipo di utente, studente o Cc ds, le azioni che si verificano sono diverse. Nel caso iniziale sarà lo studente a collegarsi, compilerà la domanda che verrà controllata per valutare se è stata compilata correttamente. Se il controllo avrà esito positivo si prosegue controllando le regole, che riguardano principalmente il controllo dei crediti. La domanda a questo punto sarà inserita in un elenco delle domande pronte per essere valutate dal Consiglio del corso di studi. Il Cc ds, che rappresenta l'altra tipologia di utente, dopo essere stato riconosciuto valuta la domanda compilata dallo studente ne crea la carriera e infine il sistema provvederà ad avvisare lo studente sulla avvenuta valutazione.

### Comportamento del Cc ds

In questo diagramma [vedi figura 5.2] vengono spiegate le operazioni che può eseguire il Cc ds per gestire la domanda di trasferimento.

Per prima cosa il Cc ds si deve identificare, una volta riconosciuto può decidere che operazioni eseguire. Nel caso decida di valutare le domande di trasferimento compilate dagli studenti deve consultare l'elenco dove sono state precedentemente riposte, valutarle, il che significa valutare se gli esami che gli studenti richiedano vengano riconosciuti siano compatibili con quelli dell'università di Bologna. Terminata la fase di valutazione, il sistema provvede a calcolare i crediti relativi agli esami riconosciuti allo studente e in base a quelli decide a quale anno iscrivere lo studente. In seguito quindi il Cc ds crea la carriera dello studente e una volta terminata tutta la procedura lo studente viene informato tramite email che la sua domanda è stata valutata.

Oltre a valutare la domanda il Cc ds può anche modificarne le regole. Questo è

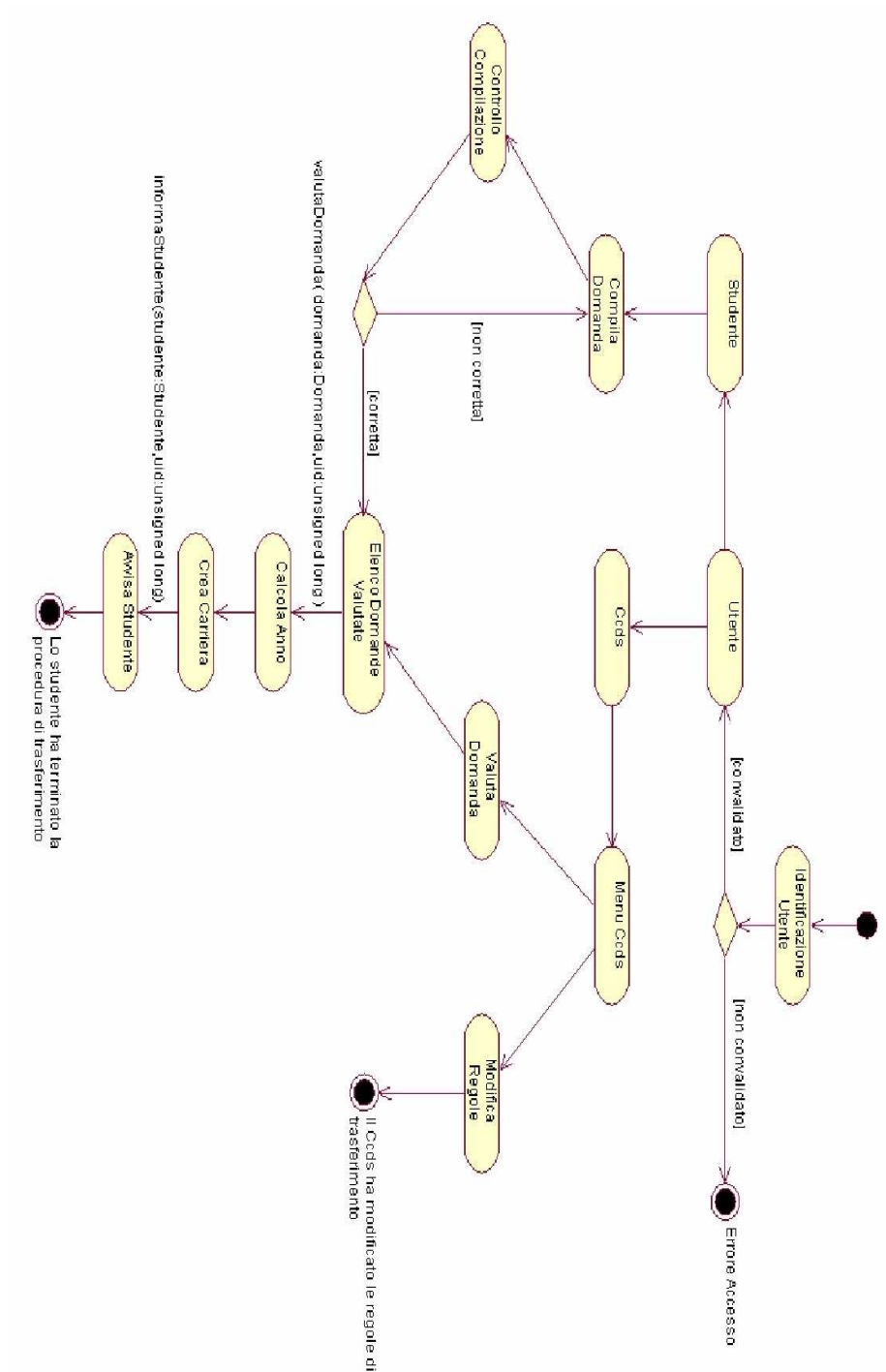


Figura 5.1: Diagramma delle attività generale

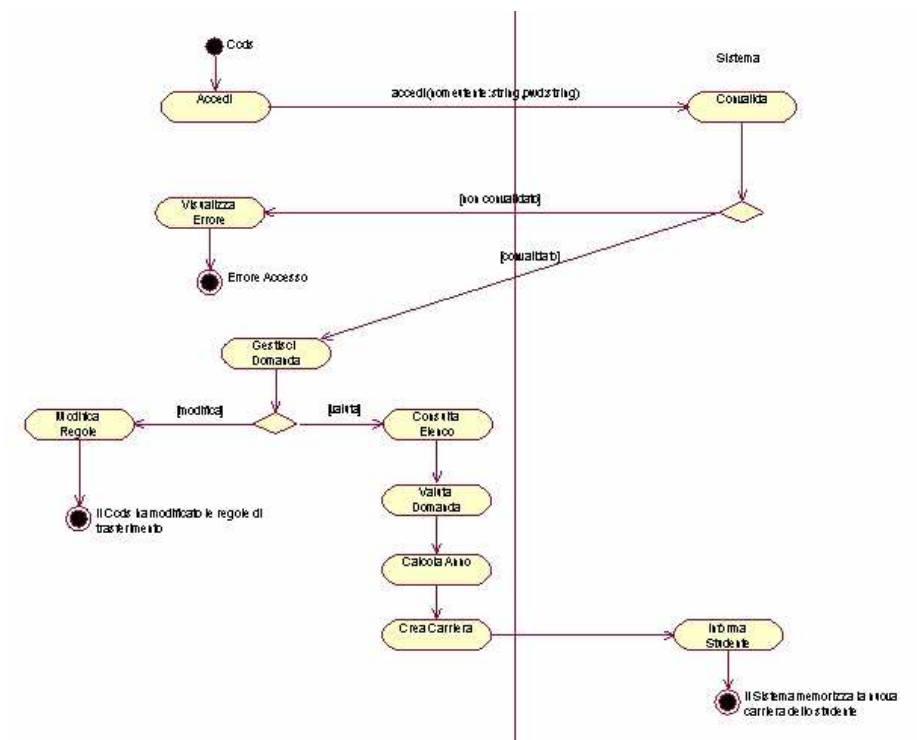


Figura 5.2: Diagramma delle attività del Ccds

un compito che si verificherà poco frequentemente in quanto può verificarsi solo se le regole per valutazione delle domande vengono cambiate, ad esempio in seguito a una nuova riforma il numero dei crediti totali consentiti viene cambiato. Per eseguire tale operazione quindi il Ccds si collega al sistema e cambia le regole per la valutazione, ad esempio come precedentemente specificato cambiando, il numero dei crediti.

## Comportamento dello studente

Di seguito [vedi figura 5.3] sono riportate le azioni svolte dallo studente che decide di trasferirsi all'università di Bologna.

Per prima cosa lo studente si collegherà al sistema. (Il sistema può già riconoscere lo studente perchè lo studente per potersi trasferire deve aver pagato le tasse, risultando quindi iscritto all'università di Bologna ed essendo quindi in possesso di un PIN fornitogli dall'università per accedere ai vari servizi offerti). Una volta collegato compilerà la domanda che verrà controllata per valutare se è stata compilata correttamente ad esempio se tutti i campi sono stati riempiti. Se il controllo ha esito negativo verrà chiesto allo studente di compilare la domanda in tutti i suoi campi, se invece il controllo ha esito positivo si prosegue controllando le regole, che riguardano principalmente il controllo dei crediti. Precisamente verranno calcolati i crediti per ogni settore (Mat, Fis, Liberi e

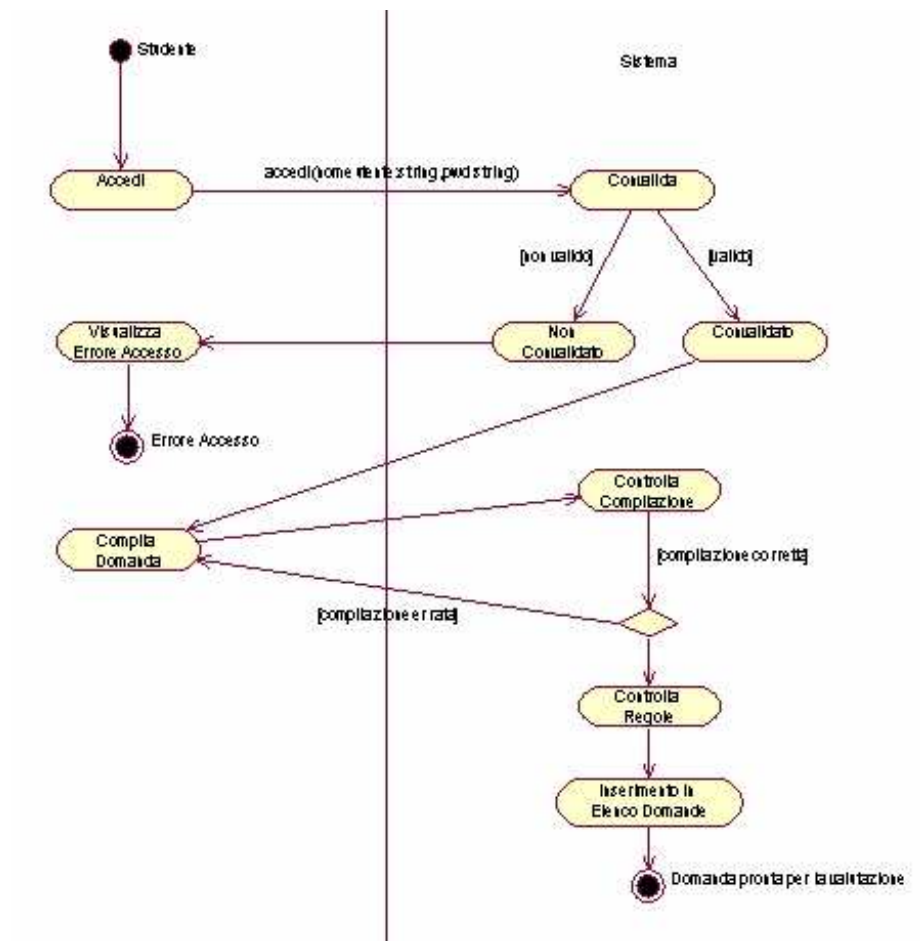


Figura 5.3: Diagramma delle attività dello Studente

Tirocinio) e totali che lo studente sostiene di possedere e chiede gli vengano riconosciuti e verranno confrontati con quelli che la facoltà richiede. La domanda verrà inserita nell'elenco delle domande pronte per essere valutate dal Consiglio del corso di studi e lo studente rimarrà in attesa della valutazione.

## Capitolo 6

# Pianificazione delle Attività

La progettazione di un'applicazione generalmente attraversa determinate fasi ed è soggetta a regole che, se rispettate, determinano un aumento della qualità sia del risultato finale sia del processo di creazione dell'applicazione stessa. Nei successivi paragrafi verrà mostrato come sono state pianificate le attività e come sono state sfruttate le risorse a disposizione.

### 6.1 RUP (Rational Unified Process)

Per poter creare un'applicazione che fornisca un servizio Web riguardante i trasferimenti universitari abbiamo utilizzato come modello generale il RUP (Rational Unified Process) il quale definisce una serie di regole che i partecipanti al progetto devono seguire per tutte le fasi che caratterizzano il ciclo di vita del progetto.

Generalmente in un RUP è possibile trovare:

- Un'organizzazione del piano di progetto, suddivisa per fasi sequenziali
- L'indicazione delle attività da svolgere e la loro sequenza
- I componenti che determinano le risorse del progetto ed i loro ruoli all'interno di esso
- Una documentazione sul lavoro effettuato e su ciò che si è prodotto

Per dare solo un'idea di come è stato affrontato il progetto e senza entrare nei particolari, ricordiamo che il ciclo di vita di un RUP viene diviso nelle seguenti quattro fasi che devono rispettare un ordine temporale sequenziale:

- Concezione
- Elaborazione
- Costruzione
- Transizione

mentre i workflow principali sono :

- Requirement (nel nostro progetto è stata chiamata Requisiti)

## CAPITOLO 6. PIANIFICAZIONE DELLE ATTIVITÀ

- Analysis and Designe (nel nostro progetto stata chiamata Analisi e Progettazione)
- Implementation
- Testing

seguiti da altri tre di supporto

- Deployment
- Environment
- Project Management Environement

Poichè il progetto relativo al corso di Ingegneria del Software dei trasferimenti non include le fasi di *costruzione* e di *transizione* (che verranno svolte in un secondo tempo) che si occupano di fatto della parte implementativa di testing e di conclusione vera e propria del progetto con la conseguente produzione di un'applicazione funzionante che soddisfi tutti i requisiti iniziali.

Di conseguenza anche le attività considerate riguardano esclusivamente la definizione dei requisiti e l'analisi/progettazione del sistema.

I grafici che verranno mostrati successivamente hanno lo scopo di fare vedere, in maniera molto efficace, come sono stati suddivise le varie attivit tra le risorse a disposizione. (Le risorse considerate sono formate esclusivamente dai componenti del gruppo).

### 6.1.1 Riassunto delle attività svolte da ciascuna risorsa

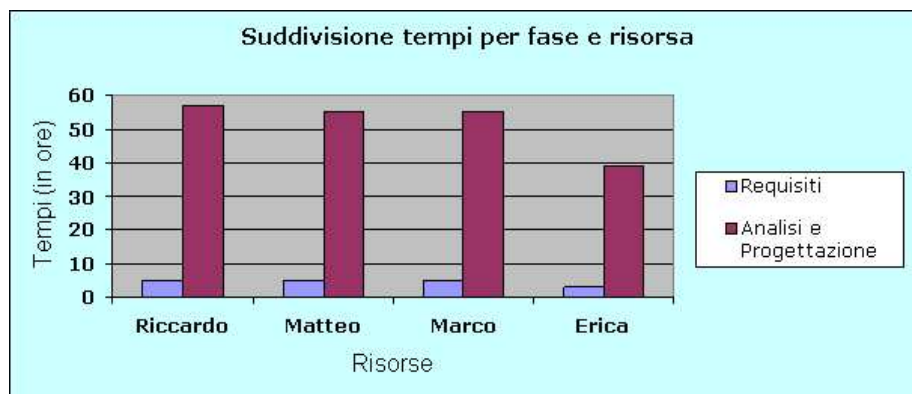


Figura 6.1: Diagramma distribuzione tempi per attività e risorsa

### 6.1.2 Diagramma di Gantt

Del seguente diagramma di Gantt [vedi figura 6.2] è stato fatto un uso improprio perchè, generalmente, esso viene utilizzato per effettuare una progettazione preventiva



## 6.1. RUP (RATIONAL UNIFIED PROCESS)

che precede l'inizio vero e proprio delle attività, mentre, nel nostro caso, viene mostrato a posteriori come sono state suddivise tutte le attività durante il progetto, facendo riferimento ai diari personali compilati dai vari componenti. La decisione di inserire anche questo diagramma è motivata dal fatto che la sua presenza mostra, con maggiore chiarezza, come si sia sviluppato il progetto durante le varie fasi e soprattutto quanto sforzo abbia richiesto alle risorse interessate.

Nelle ordinate del grafico vengono indicate le varie attività suddivise nei due blocchi principali individuati precedentemente (*Requisit e progettazione*), mentre sulle ascisse compare il tempo che indica la successione temporale delle varie attività. Ogni stato di avanzamento di un'attività indica anche la risorsa che ha eseguito quel determinato compito.

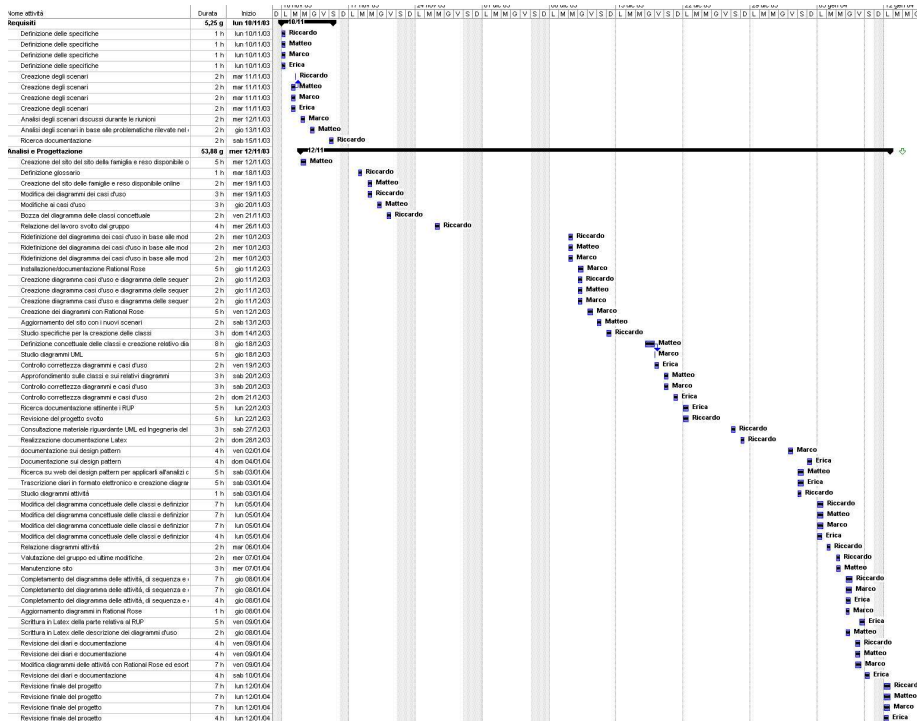


Figura 6.2: Diagramma di Gantt

### 6.1.3 Workflow

In questo paragrafo verranno indicate le attività in ordine temporale:

- Definizione dei requisiti
  - Analisi del problema
  - Individuazione degli scenari
- Analisi e progettazione

## CAPITOLO 6. PIANIFICAZIONE DELLE ATTIVITÀ

---

- Creazione di un glossario con i termini principali
- Creazione dei diagrammi UML (diagramma dei casi d'uso, delle classi, delle attività e di sequenza)
- Creazione della documentazione

Durante la prima fase è stato creato un sito online sia per le famiglie che per il singolo gruppo con lo scopo di mantenere aggiornato lo stato di avanzamento del progetto.

Altre operazioni che si sono eseguite durante il corso di tutto il progetto sono state: la manutenzione del sito e l'aggiornamento del diario di gruppo e di quelli personali.

# Appendice A

Diari personali

d

## 6.1. RUP (RATIONAL UNIFIED PROCESS)

---

d

d

# Appendice B

Diari di gruppo

# Bibliografia

- [1] Martin Fowler. *UML Distilled*. Addison-Wesley, 2000.
- [2] Vince Huston. Design pattern. <http://home.earthlink.net/huston2/dp/patterns.html>, 2003.
- [3] Ernesto Damiani and Mauro Madravio. *UML Pratico con elementi di ingegneria del software*. Addison-Wesley, 2003.
- [4] Jason T. Roff. *Fondamenti di UML*. McGraw-Hill, 2003.