**TEMPLATE STRUTTURA RELAZIONE**

# Descrizione del problema

## Analisi e specifica dei requisiti

Questa sezione deve contenere una descrizione informale dei requisiti (una rielaborazione del testo del progetto, aggiungendo eventuali chiarimenti e precisazioni, riportando solo i requisiti implementati) e possibilmente facendo una distinzione tra requisiti funzionali e requisiti non funzionali (per esempio, requisiti di prodotto, di processo, di usabilità, della GUI, di sicurezza, ecc.).

Il progetto in relazione tratta di un sistema di bike sharing.

Un utente generico ha la possibilità di accedere a tale sistema per avere una panoramica sulle varie possibilità di abbonamento e le biciclette utilizzabili.

Tale utente ha poi la possibilità di accedere ad una maschera di registrazione nella quale gli verranno richiesti un’email ed una password. Durante questa fase egli ha anche la possibilità di provare il suo status di studente, in modo tale da poter giovare del sistema in modo gratuito.

Tramite l’utilizzo di email e password l’utente avrà poi la possibilità di entrare nella propria area riservata nella quale potrà effettuare diverse azioni: attivare un abbonamento, registrare una carta di credito oppure classificarsi come studente provando il suo status di studente.

Una volta che l’utente registrato sceglie un’abbonamento viene subito stornato il costo totale, ciò implica che quindi l’utente abbia prima registrato una carta di credito valida. La validazione di tale carta di credito avverrà da un’istituto di credito esterno (TODO: che nel sistema simulerò).

Qualsiasi pagamento che avvenga all’interno del sistema verrà approvato da un istituto di credito esterno, nel momento nel quale viene accertato che la carta di credito registrata abbia la somma, oppure rigettato.

L’utente che ha attivato un abbonamento avrà la possibilità di noleggiare una bicicletta; esistono tuttavia diverse tipologie di biciclette le quali avranno una tariffazione diversa basata anche sul tempo di utilizzo:

SPECIFICA TEMPI UTILIZZO, costi e supplemento e penale.

Concluso il noleggio l’utente dovrà restituire la bicicletta. La verifica che la bicicletta sia stata riconsegnata avverrà digitando, sul totem, il codice univoco assegnato all’utente.

L’utente durante il noleggio oppure di una data bicicletta potrà comunicare dei danni ad essa, i quali verranno controllati poi dagli operatori comunali.

Qualora l’utente superasse il limite massimo di utilizzo della bicicletta, stabilito a 2 ore, per tre volte, gli verrà annullato l’abbonamento.

Tra un noleggio ed un altro da parte dello stesso utente dovranno intercorrere almeno 5 minuti.

L’operatore comunale è un altro utente del sistema con principalmente compiti di gestione. Tale gestione è divisa principalmente in due parti: gestione delle biciclette e gestione delle rastrelliere.

Per quanto riguarda la gestione delle biciclette avrà la possibilità di eliminare oppure aggiungere delle biciclette al sistema. Un’ulteriore azione attuabile da tale utente è la rilocazione di una o più biciclette su rastrelliere diverse. Qualora venissero comunicati dei danni da parte dell’utente, tale bicicletta verrà automaticamente posta in uno stato di manutenzione e non sarà utilizzabile fino al controllo da parte del personale comunale.

Per quanto riguarda la gestione delle rastrelliere l’utente comunale avrà la possibilità di eliminare e aggiungere delle rastrelliere.

Quest’ultimo utente (operatore comunale) può anche richiedere al sistema dei dati statistici per cercare di offrire il miglior servizio possibile.

## [Opzionale] Glossario

Definisce termini e acronimi utilizzati nel documento.

Sistema = software di bike sharing

!utente studente = not utente studente (tutti gli utenti tranne gli studenti)

# Progettazione del Sistema

## Diagramma dei casi d’uso

Per ogni caso d’uso ci deve essere almeno una descrizione testuale del caso d’uso. Per i casi d’uso principali (almeno per i casi d’uso base) deve esserci anche la descrizione di un possibile scenario, eventualmente con l’indicazione delle possibili varianti.

## Descrizione degli scenari

|  |  |
| --- | --- |
| **Nome** | Visualizzazione sistema |
| **Scopo** | Dare una panoramica del sistema all’utente appena visualizza l’applicativo |
| **Attore/i** | Utente |
| **Pre-condizioni** | Attore accede alla pagina principale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore accede alla pagina principale del sistema  2. L’attore visualizza la panoramica del sistema |
| **Alternativa/e** |  |
| **Post-condizioni** | L’attore visualizza la panoramica del sistema |

|  |  |
| --- | --- |
| **Nome** | Registrazione |
| **Scopo** | Dare la possibilità di registrarsi al sistema |
| **Attore/i** | Utente |
| **Pre-condizioni** | L’utente non si deve essere mai registrato con l’email e la password che intende usare |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore accede alla maschera di registrazione  2. L’attore compila correttamente tutti i campi richiesti  3. L’attore sceglie una password  4. La password viene accettata dal sistema  5. L’attore è registrato |
| **Alternativa/e** | 2.a. L’attore non compila correttamente i campi  2.b. L’attore deve ricompilare i campi richiesti.  4.a. La password inserita non viene accettata dal sistema  4.b. L’utente deve scegliere una nuova password. |
| **Post-condizioni** | L’attore è registrato al sistema |

|  |  |
| --- | --- |
| **Nome** | Prova status studente |
| **Relations** | Extension of: Registrazione  Extension of: Accesso area riservata |
| **Scopo** | Dar prova dello status di studente |
| **Attore/i** | Utente studente e sistema universitario |
| **Pre-condizioni** | L’utente deve essere uno studente e quindi possedere e registrarsi con una mail universitaria |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si registra come utente studente  2. L’attore entra nella maschera per provare il suo status di studente  2. L’attore inserisce l’email universitaria  3. Il sistema richiede la conferma all’istituto universitario  4. Il sistema universitario approva l’email |
| **Alternativa/e** | 1.a. L’attore accede all’area riservata  continua dal punto 2 |
| **Post-condizioni** | L’utente è un utente studente |

|  |  |
| --- | --- |
| **Nome** | Registrazione carta di credito |
| **Relations** | Extension of: Accesso area riservata |
| **Scopo** | Registrare una carta di credito dalla quale verranno stornati i costi dell’abbonamento e noleggio delle biciclette |
| **Attore/i** | Utente registrato e Istituto di credito |
| **Pre-condizioni** | L’utente deve essere registrato e possedere una carta di credito valida |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’utente accede all’area riservata  2. L’utente inserisce correttamente la carta di credito con tutti i campi richiesti.  3. I dati vengono inviati dal sistema all’istituto di credito  4. L’istituto di credito accerta la validità della carta di credito |
| **Alternativa/e** | 2.a. L’utente non inserisce correttamente tutti i campi richiesti della carta di credito  2.b. Il sistema richiede all’utente di reinserire correttamente i dati della carta di credito  3.a. L’istituto di credito non accerta la validità della carta di credito  3.b. L’utente torna al punto 1 |
| **Post-condizioni** | La carta di credito dell’utente viene registrata nel sistema |

|  |  |
| --- | --- |
| **Nome** | Registrazione abbonamento |
| **Relations** | Extension of: Accesso area riservata  Include: Pagamento |
| **Scopo** | Registrare un nuovo abbonamento oppure rinnovare uno scaduto |
| **Attore/i** | Utente registrato |
| **Pre-condizioni** | L’utente deve essere registrato ed aver registrato correttamente una carta di credito. Inoltre deve essere disponibile su quest’ultima il denaro pari all’ammontare del costo dell’abbonamento scelto. |
| **Trigger** | Pagamento {Condition:!Utente studente} |
| **Descrizione sequenza eventi** | 1. L’attore accede all’area riservata  2. L’attore sceglie un’abbonamento  3. L’attore inserisce la propria password  4. Il sistema valuta la correttezza della password  5. Il sistema invia la richiesta di pagamento  6. Il pagamento viene effettuato  7. Il sistema mostra all’attore il codice univoco col quale ritirare una bicicletta |
| **Alternativa/e** | 4.a. Il sistema valuta la password non corretta  4.b. ritorna al punto 3  6.a. Il pagamento non viene effettuato  6.b. L’utente deve scegliere un’altra carta di credito registrata per il pagamento |
| **Post-condizioni** | L’utente registrato ha un abbonamento valido con un codice univoco utile per ritirare una bicicletta |

|  |  |
| --- | --- |
| **Nome** | Accesso area riservata |
| **Relations** | Include: Autenticazione |
| **Scopo** | Accedere all’area riservata per gestire gli abbonamenti, inserire una carta di credito oppure per provare lo status di studente |
| **Attore/i** | Utente registrato |
| **Pre-condizioni** | L’utente deve essere registrato. |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’utente effettua l’autenticazione  2. L’utente visualizza la propria area riservata con gli abbonamenti e le carte di credito |
| **Alternativa/e** | - |
| **Post-condizioni** | - |

|  |  |
| --- | --- |
| **Nome** | Autenticazione utente |
| **Scopo** | Permettere all’utente registrato di autenticarsi ed entrare nella propria area riservata |
| **Attore/i** | Utente registrato |
| **Pre-condizioni** | L’utente deve essere registrato |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’utente entra nella maschera di autenticazione come utente  2. L’utente inserisce email e password correttamente  3. Il sistema controlla la validità dell’account  4. La validità è accertata |
| **Alternativa/e** | 2.a. L’utente non inserisce in modo corretto username e email  2.b. Il sistema richiede di inserire username ed email  4.a. Il sistema non accerta la validità dell’account  4.b. rinizia dal punto 1. |
| **Post-condizioni** | L’utente è autenticato |

|  |  |
| --- | --- |
| **Nome** | Pagamento |
| **Scopo** | Interfacciarsi con l’istituto di credito per attuare una richiesta di pagamento |
| **Attore/i** | Istituto di credito |
| **Pre-condizioni** | L’utente deve aver registrato una carta di credito valida e deve avere abbastanza denaro per la richiesta fatta |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. Il sistema invia all’istituto di credito le informazioni della carta di credito dell’utente ed il costo dell’abbonamento  2. L’stituto di credito accerta la correttezza della carta di credito  3. L’istituto di credito invia i soldi al sistema |
| **Alternativa/e** | 1.a. Il sistema invia all’istituto di credito le informazioni della carta di credito dell’utente ed il costo del noleggio  2.a. L’istituto di credito non accerta la correttezza della carta di credito  2.b. Fallimento dell’istanza di pagamento |
| **Post-condizioni** | Il sistema riceve il pagamento dall’istituto di credito |

|  |  |
| --- | --- |
| **Nome** | Noleggio bicicletta |
| **Scopo** | Permettere all’utente di noleggiare una bicicletta |
| **Attore/i** | Utente abbonato  Totem rastrelliera |
| **Pre-condizioni** | L’utente deve avere un abbonamento attivo nel momento in cui vuole noleggiare una bicicletta |
| **Trigger** | Pagamento {Condition:!Utente studente} |
| **Descrizione sequenza eventi** | 1. L’attore arriva alla rastrelliera  2. L’attore inserisce nel totem il proprio codice utente e la password  3. Il totem controlla la validità del codice utente e la password  4. Il totem valida il codice utente e la password  5. L’attore sceglie il tipo di bici che vuole utilizzare tra quelle disponibili e non in manutenzione  6. Il totem richiede alla rastrelliera di sbloccare la bicicletta prescelta  7. La rastrelliera indica che ha sbloccato la bicicletta correttamente  8. Il totem indica il numero di posteggio della bicicletta assegnata all’attore  9. L’attore preleva la bicicletta |
| **Alternativa/e** | 4.a. Il totem non valida il codice utente e la password  4.b. Il totem richiede nuovamente il codice utente e la password  4.c. Continua dal punto 5  7.a. La rastrelliera indica che è impossibilitata a sbloccare il tipo di bicicletta prescelta. Ciò indica un’errore hardware, la rastrelliera è impossibilitata a sbloccare qualsiasi bicicletta di questo tipo.  7.b. Il totem mostra un messaggio di errore all’utente  7.c. Il totem richiede all’utente di scegliere un nuovo tipo di bicicletta |
| **Post-condizioni** | L’attore può utilizzare la bicicletta |

|  |  |
| --- | --- |
| **Nome** | Addebito penale |
| **Relations** | Include: Pagamento  Extension of: Noleggio bicicletta |
| **Relations** | Extension of: Noleggio bicicletta |
| **Scopo** | Addebitare una penale all’attore abbonato che non restituisce la bicicletta entro le 24 ore dall’ultimo noleggio |
| **Attore/i** | Utente abbonato |
| **Pre-condizioni** | L’utente deve avere un abbonamento valido, aver noleggiato la bicicletta ed avere registrato una carta di credito valida |
| **Trigger** | Pagamento |
| **Descrizione sequenza eventi** | 1. L’attore noleggia una bicicletta  2. L’attore non restituisce la bicicletta entro le 24 ore successive  3. Il sistema gli addebita la penale |
| **Alternativa/e** | - |
| **Post-condizioni** | Addebito penale sulla carta di credito dell’attore |

|  |  |
| --- | --- |
| **Nome** | Pagamento supplemento |
| **Relations** | Include: Pagamento  Extension of: Noleggio bicicletta |
| **Scopo** | Addebitare all’attore il supplemento per l’utilizzo prolungato della bicicletta noleggiata |
| **Attore/i** | Utente abbonato |
| **Pre-condizioni** | L’attore deve avere un abbonamento valido, aver noleggiato una bicicletta ed avere una carta di credito valida. |
| **Trigger** | Pagamento {Condition: !utente studente} |
| **Descrizione sequenza eventi** | 1. L’attore noleggia una bicicletta  2. Il sistema calcola che il noleggio va oltre l’utilizzo gratuito  3. Il sistema addebita il supplemento sulla carta di credito dell’attore |
| **Alternativa/e** | - |
| **Post-condizioni** | Addebito supplemento sulla carta di credito dell’autore |

|  |  |
| --- | --- |
| **Nome** | Restituzione bicicletta |
| **Scopo** | Permettere all’attore di restituire una bicicletta noleggiata |
| **Attore/i** | Utente abbonato  Totem rastrelliera |
| **Pre-condizioni** | L’attore deve avere noleggiato una bicicletta e non deve ancora averla restituita |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore noleggia la bicicletta  2. L’attore conclude il noleggio  3. L’attore trova una rastrelliera libera  4. L’attore pone la bicicletta |
| **Alternativa/e** | 3.a. L’attore trova una rastrelliera occupata  3.b. L’attore cerca un’altra rastrelliera libera |
| **Post-condizioni** | L’attore ha riposto la bicicletta nella rastrelliera |

|  |  |
| --- | --- |
| **Nome** | Annullamento abbonamento |
| **Relations** | Extension of: Restituzione bicicletta |
| **Scopo** | Annullare l’abbonamento all’attore che supera il limite massimo di 2 ore per tre volte |
| **Attore/i** | Utente abbonato |
| **Pre-condizioni** | L’utente deve essere in possesso di un abbonamento valido, aver noleggiato una bicicletta per almeno 3 volte. |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore noleggia una bicicletta  2. L’attore restituisce la bicicletta passate 2 ore dal noleggio  3. Il sistema calcola il numero di volte nel quale non ha rispettato tale vincolo  4. Il sistema trova che l’attore è recidivo (>2)  5. Il sistema revoca l’abbonamento all’attore |
| **Alternativa/e** | - |
| **Post-condizioni** | Il sistema revoca l’abbonamento all’attore |

|  |  |
| --- | --- |
| **Nome** | Verifica riconsegna |
| **Relations** | Extension of: Restituzione bicicletta |
| **Scopo** | Dare la possibilità all’attore di verificare la corretta restituzione della bicicletta |
| **Attore/i** | Utente abbonato  Totem rastrelliera |
| **Pre-condizioni** | L’attore deve aver noleggiato una bicicletta |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore noleggia una bicicletta  2. L’attore conclude il noleggio  3. L’attore trova una rastrelliera libera  4. L’attore pone la bicicletta nella rastrelliera libera  5. L’attore digita sul totem della rastrelliera il codice utente e la password  6. Il sistema controlla la validità del codice utente e della password  7. Il sistema valuta corretti codice utente e password  8. Il sistema controlla la corretta restituzione  9. Il totem visualizza un messaggio di corretta restituzione |
| **Alternativa/e** | 7.a. Il sistema non valuta corretti codice utente e password  7.b. Il totem richiede di reinserire codice utente e password  7.c. torna al punto 5  6.a. Il sistema verifica la non corretta restituzione  6.b. Il totem visualizza un messaggio di non corretta restituzione e richiede di porre la bicicletta nel posto adatto  6.c. ritorna al punto 3 |
| **Post-condizioni** | L’utente ha verificato la corretta restituzione della bicicletta noleggiata |

|  |  |
| --- | --- |
| **Nome** | Comunicazione danni |
| **Relations** | Extension of: Restituzione bicicletta |
| **Scopo** | Dare la possibilità all’attore di comunicare danni alla bicicletta noleggiata, in modo da permettere una manutenzione |
| **Attore/i** | Utente abbonato |
| **Pre-condizioni** | L’attore deve possedere un abbonamento valido ed aver noleggiato una bicicletta |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore noleggia una bicicletta  2. L’attore si accorge di danni alla bicicletta  3. L’attore restituisce la bicicletta  4. L’attore digita sul totem codice utente e password  6. Il sistema controlla la validità del codice utente e della password  7. Il sistema valuta corretti codice utente e password  8. Il sistema controlla la corretta restituzione  9. Il totem visualizza un messaggio di corretta restituzione  10. L’attore seleziona il servizio relativo alla comunicazione danni  11. La rastrelliera rende indisponibile tale bicicletta  12. Il totem invia un messaggio di ringraziamento all’attore |
| **Alternativa/e** | 7.a. Il sistema non valuta corretti codice utente e password  7.b. Il totem richiede di reinserire codice utente e password  7.c. torna al punto 5 |
| **Post-condizioni** | La bicicletta viene resa indisponibile |

|  |  |
| --- | --- |
| **Nome** | Gestione biciclette |
| **Relations** | Generalization of:  - Aggiunta bicicletta  - Eliminazione bicicletta  - Rilocazione bicicletta  - Manutenzione bicicletta  Include: Autenticazione personale comunale |
| **Scopo** | Dare la possibilità all’attore di gestire le biciclette del sistema di bike sharing |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello di gestione delle biciclette dal quale potrà scegliere le azioni opportune. |
| **Alternativa/e** | - |
| **Post-condizioni** | L’attore entra nel pannello di gestione delle biciclette |

|  |  |
| --- | --- |
| **Nome** | Manutenzione bicicletta |
| **Relations** | Specialization of: Gestione biciclette |
| **Scopo** | Permettere all’attore di attuare una manutenzione ad una bicicletta |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale e deve esistere una bicicletta in manutenzione |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello di gestione delle biciclette  3. L’attore entra nel pannello di manutenzione  4. L’attore seleziona una bicicletta  5. L’attore sistema la bicicletta  6. L’attore indica al sistema la conclusione della manutenzione  7. Il sistema rende di nuovo disponibile la bicicletta |
| **Alternativa/e** | - |
| **Post-condizioni** | La bicicletta torna disponibile |

|  |  |
| --- | --- |
| **Nome** | Rilocazione bicicletta |
| **Relations** | Specialization of: Gestione biciclette |
| **Scopo** | Permettere all’attore di rilocare una bicicletta |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello di gestione delle biciclette  3. L’attore entra nel pannello di rilocazione bicicletta  4. L’attore seleziona una bicicletta da rilocare  5. L’attore seleziona la rastrelliera di destinazione  6. Il sistema controlla che la rastrelliera non sia piena  7. Il sistema conferma la disponibilità della rastrelliera  8. La bicicletta viene rilocata nella nuova rastrelliera |
| **Alternativa/e** | 7.a. Il sistema indica che la rastrelliera è piena  7.b. Il sistema chiede all’attore di scegliere una nuova rastrelliera  7.c. continua da punto 5 |
| **Post-condizioni** | La bicicletta viene rilocata in una nuova rastrelliera correttamente |

|  |  |
| --- | --- |
| **Nome** | Eliminazione bicicletta |
| **Relations** | Specialization of: Gestione biciclette |
| **Scopo** | Permettere all’attore di eliminare una bicicletta dal sistema |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello di gestione delle biciclette  3. L’attore entra nel pannello di eliminazione delle biciclette  4. L’attore seleziona una bicicletta da eliminare  5. Il sistema provvede ad eliminare la bicicletta  6. Il sistema conferma la corretta eliminazione della bicicletta |
| **Alternativa/e** | - |
| **Post-condizioni** | La bicicletta viene eliminata dal sistema |

|  |  |
| --- | --- |
| **Nome** | Aggiunta bicicletta |
| **Relations** | Specialization of: Gestione biciclette |
| **Scopo** | Permettere all’attore di aggiungere una nuova bicicletta al sistema |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello di gestione delle biciclette  3. L’attore entra nel pannello di aggiunta biciclette  4. L’attore inserisce i dati relativi la bicicletta richiesti dal sistema  5. Il sistema convalida i dati  6. Il sistema conferma la corretta aggiunta della bicicletta |
| **Alternativa/e** | 5.a. Il sistema non conferma i dati  5.b. Il sistema richiede all’attore di reinserire i dati in modo corretto  5.c. continua da punto 4 |
| **Post-condizioni** | La bicicletta viene aggiunta al sistema |

|  |  |
| --- | --- |
| **Nome** | Gestione rastrelliere |
| **Relations** | Generalization of:  - Eliminazione rastrelliera  - Aggiunta rastrelliera  Include: Autenticazione personale comunale |
| **Scopo** | Permettere all’attore di gestire le rastrelliere del sistema |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello di gestione delle rastrelliere dal quale potrà scegliere le azioni opportune. |
| **Alternativa/e** | - |
| **Post-condizioni** | L’attore entra nel pannello di gestione delle rastrelliere |

|  |  |
| --- | --- |
| **Nome** | Eliminazione rastrelliera |
| **Relations** | Specialization of: Gestione rastrelliere |
| **Scopo** | Permettere all’attore di eliminare le rastrelliere dal sistema |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale e la rastrelliera deve essere vuota |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello di gestione delle rastrelliere  3. L’attore entra nel pannello di eliminazione rastrelliera  4. L’attore seleziona una rastrelliera da eliminare  5. Il sistema controlla che la rastrelliera sia vuota  6. Il sistema conferma che la rastrelliera selezionata può essere eliminata  7. Il sistema elimina la rastrelliera  8. Il sistema invia un messaggio all’attore di corretta eliminazione della rastrelliera |
| **Alternativa/e** | 6.a. Il sistema trova delle biciclette legate alla rastrelliera  6.b. Il sistema indica all’attore che è necessario che la rastrelliera sia prima vuota per essere eliminata |
| **Post-condizioni** | La rastrelliera viene eliminata dal sistema |

|  |  |
| --- | --- |
| **Nome** | Aggiunta rastrelliera |
| **Relations** | Specialization of: Gestione rastrelliere |
| **Scopo** | Permettere all’attore di aggiungere una rastrelliera al sistema |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello di gestione delle rastrelliere  3. L’attore entra nel pannello di aggiunta rastrelliera  4. L’attore inserisce i dati richiesti dal sistema riguardanti la nuova rastrelliera  5. Il sistema controlla la correttezza dei dati  6. Il sistema conferma la correttezza dei dati immessi  7. Il sistema conferma l’aggiunta della rastrelliera al sistema |
| **Alternativa/e** | 6.a. Il sistema indica che i dati non sono stati immessi correttamente  6.b. Il sistema chiede il reinserimento dei dati all’attore  6.c. continua da punto 4 |
| **Post-condizioni** | La rastrelliere viene aggiunta al sistema |

|  |  |
| --- | --- |
| **Nome** | Produzione dati statistici |
| **Relations** | Include: Autenticazione personale comunale |
| **Scopo** | Permettere all’attore di avere dei dati statistici da parte del sistema |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore deve essere autenticato come personale comunale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore si autentica come personale comunale  2. L’attore entra nel pannello dei dati statistici  3. L’attore naviga tra le varie possibilità di viste dei dati |
| **Alternativa/e** | - |
| **Post-condizioni** | L’attore visualizza diverse viste di dati statistici |

|  |  |
| --- | --- |
| **Nome** | Autenticazione personale comunale |
| **Scopo** | Permettere all’attore di autenticarsi come personale comunale per avere dei permessi privilegiati e specifici sul sistema |
| **Attore/i** | Personale comunale |
| **Pre-condizioni** | L’attore che si vuole autenticare come personale comunale deve essere stato aggiunto preventivamente al sistema da parte di un admin del personale comunale |
| **Trigger** | - |
| **Descrizione sequenza eventi** | 1. L’attore entra nella maschera di autenticazione come personale comunale  2. L’attore inserisce i dati richiesti per l’autenticazione correttamente  3. Il sistema controlla la validità dell’account  4. La validità viene accertata |
| **Alternativa/e** | 2.a. L’attore non inserisce correttamente i dati richiesti per l’autenticazione  2.b. Il sistema richiede all’attore di reinserire i dati  2.c. continua dal punto 2  4.a. Il sistema non accerta la validità dell’account  4.b. rinizia dal punto 1. |
| **Post-condizioni** | L’utente è autenticato |

## Diagramma delle classi (modello di progetto)

In questo stadio ha senso che si riporti e si discuta il diagramma delle classi di progetto. Il diagramma delle classi di programma verrà presentato e discusso nella sezione relativa all’implementazione.

Si ricordi che comunque il diagramma riportato in questa sezione:

1. Deve essere tracciabile con tutte le alte viste.
2. Deve essere tracciabile con il codice nel caso si decida di riportare un unico diagramma delle classi (quindi solo quello di programma).
3. Verrà valutato guardando se i principi chiave della progettazione OO siano stati considerati (classi altamente coese, poco accoppiamento tra classi, uso di gerarchia tra classi, ecc.) e considerando le associazioni tra classi.

I metodi get e set non sono stati riportati perché considerati impliciti. (l’ha detto in lab di scriverlo).

## Diagrammi di sequenza

## Diagrammi delle attività

## Macchine di stato

La notazione utilizzata per esprimere le macchine si stato deve essere usata in modo corretto, ad esempio i *call event* devono essere tracciabili con i messaggi dei casi d’uso e i metodi delle classi. Lo stesso si applica per tutte azioni che sono chiamate di metodi.

## Diagramma dei componenti

# Implementazione del sistema

## Diagramma delle classi (modello di programma)

In questa sezione va riportato il diagramma delle classi di programma.

### Discussione dei Design Pattern utilizzati

I design pattern utilizzati non vanno semplicemente elencati, ma anche discussi: va giustificato il loro utilizzo ed eventualmente va specificato quali modifiche del diagramma delle classi abbia implicato il loro utilizzo.

## Gestione dei dati persistenti

In questa sezione va descritta la struttura del database utilizzato per gestire la persistenza dei dati e il DBMS utilizzato.

## Descrizione dell’Interfaccia Grafica

Vanno riportate alcune schermate dell’interfaccia grafica implementata.

Nel caso si sia fatta una validazione dell’input, i controlli effettuati e le loro finalità vanno spiegati in questa sezione.

## Diagramma di deployment

## Specifica e verifica dei vincoli

Vanno elencati i vincoli OCL facendo riferimento al diagramma delle classi definitivo.

Per almeno un piccolo sottoinsieme di vincoli OCL è opportuno mostrare come almeno siano stati mappati in JML.

Visto che l’utilizzo delle librerie per l’implementazione dell’interfaccia grafica richiedono una versione di Java superiore a quella che supporta JML, non è obbligatorio ma opzionale riportare una o più schermate che mostrino il risultato in caso di violazioni di vincoli.

## Descrizione del testing

Vanno documentati i test driver in JUnit e, dove possibile, va specificato il criterio di copertura utilizzato.

## Note per l’installazione e l’utilizzo

In questa sezione vanno indicati l’ambiente di sviluppo e di compilazione.

Vanno specificate le eventuali credenziali necessarie per l’accesso al database e eventuali account di test già presenti nel sistema.