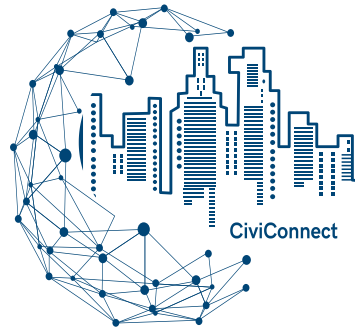




Laurea Triennale in informatica-Università di Salerno
Corso di *Ingegneria del Software* - Prof.ssa F. Ferrucci e Prof. F. Palomba



Test Plan

CiviConnect

Riferimento	C14_TP_ver.0.2
Versione	1.0
Data	
Destinatario	ANCI-DTD, Prof. Filomena Ferrucci, Prof. Fabio Palomba
Presentato da	ConnecTeam
Approvato da	Bacco Luigi, Scala Benedetto



Laurea Triennale in informatica-Università di Salerno
Corso di *Ingegneria del Software*- Prof.ssa F.Ferrucci e Prof. F.Palomba



Laurea Triennale in informatica-Università di Salerno
Corso di *Ingegneria del Software* - Prof.ssa F. Ferrucci e Prof. F. Palomba

Revision History

Data	Versione	Descrizione	Autori
25/11/2024	0.1	Primo Template	GSP
25/11/2024	0.2	Stesura capitolo 1,2,4,5,6,7,8	BS
28/11/2024	0.3	Stesura capitolo 3	LB
02/12/2024	0.4	Stesura RF_GU_1	MC
02/12/2024	0.5	Stesura Test RF_GU_2	GG
04/12/2024	0.6	Stesura Test RF_GSA_4	MB
10/12/2024	0.7	Prima stesura Test RF_GA_1	GSO
12/12/2024	0.8	Prima stesura Test RF_GSC_3	FF
12/12/2024	0.9	Stesura Test RF_GU_3	GSP
12/12/2024	0.10	Stesura RF_GSC_1	DA
13/12/2024	0.11	Stesura Test RF_GSA_3	GC
16/12/2024	1.0	Revisione Generale	Tutto il Team



Laurea Triennale in informatica-Università di Salerno
Corso di Ingegneria del Software - Prof.ssa F. Ferrucci e Prof. F. Palomba

Team Members

Nome	Acronimo	Contatto
Luigi Bacco	LB	l.bacco2@studenti.unisa.it
Benedetto Scala	BS	b.scala1@studenti.unisa.it
Generoso Sorridi	GSO	g.sorridi@studenti.unisa.it
Marco Brescia	MB	m.brescia16@studenti.unisa.it
Francesco Faiella	FF	f.faiella16@studenti.unisa.it
Gianluigi Citro	GC	g.citro37@studenti.unisa.it
Giuseppe Speranza	GSP	g.speranza15@studenti.unisa.it
Manuel Cieri	MC	m.cieri1@studenti.unisa.it
Domenico Auriemma	DA	d.auriemma4@studenti.unisa.it
Giuseppe Gambardella	GG	g.gambardella27@studenti.unisa.it



Sommario

Revision History	3
Team Members	4
Sommario	5
1. Introduzione	6
2. Panoramica del sistema.....	7
3. Relazioni con gli altri documenti.....	7
4. Features da testare	8
5. Pass/Fail criteria	8
6. Approccio al testing.....	9
6.1 Unit Testing	9
6.2 Functional Validation.....	10
7. Testing schedule.....	10
7.1 Pianificazione	10
7.2 Esecuzione dei Test	10
7.3 Report Finali	11
8. Sospensione e Ripristino.....	11
9. Test Cases.....	12



1. Introduzione

CiviConnect si propone di facilitare le interazioni tra cittadini e amministrazioni locali, al fine di migliorare la comunicazione e l'efficienza dei servizi pubblici, creando uno strumento che coinvolga attivamente la comunità.

Il documento di Test Plan ha l'obiettivo di descrivere e analizzare le attività di testing per la piattaforma **CiviConnect**, garantendo che ogni funzionalità operi in maniera corretta e soddisfi i requisiti utente.

Al suo interno vengono delineate le strategie di testing adottate, le funzionalità da verificare e gli strumenti scelti per la rilevazione e la gestione degli errori, con lo scopo di presentare al cliente finale una piattaforma stabile, affidabile e priva di malfunzionamenti.

Sono state pianificate attività di testing per le seguenti aree funzionali:

- Gestione Utente
- Gestione Admin
- Gestione Segnalazione Cittadino
- Gestione Segnalazione Comune
- Analisi Dati



2. Panoramica del sistema

Il sistema proposto per CiviConnect adotta un'architettura **Three-Tier**, con una chiara separazione tra i diversi strati funzionali: Presentation Layer, Service Layer e Data Layer:

- **Presentation Layer:** sviluppato con Flutter.
- **Service Layer:** implementato in Dart con gestione delle richieste tramite API Firebase.
- **Data Layer:** basato su Cloud Firestore per la persistenza dei dati.

Strategia di Testing

Il testing del sistema sarà focalizzato esclusivamente sull'**Unit Testing**, condotto attraverso la tecnica di **Category Partition** applicata ai casi d'uso definiti nel RAD.

L'obiettivo è verificare il corretto funzionamento delle singole unità di codice all'interno dei diversi strati dell'architettura, garantendo che i componenti rispettino i requisiti funzionali specificati.

3. Relazioni con gli altri documenti

I documenti a cui si fa riferimento sono:

- Requirements Analysis Document
- System Design Document

Il seguente documento elenca i casi di Test che saranno eseguiti, i cui risultati faranno riferimento ai seguenti documenti:

- [Test Case Specification](#)
- [Test Incident Report](#)
- [Test Summary Report](#)



4. Features da testare

Di seguito la lista delle features di cui si effettuerà il testing per le varie gestioni:

- Gestione Utente
- UC_GU_1: Login
- UC_GU_2: Logout
- UC_GU_3: Registrazione Cittadino
- Gestione Admin
- UC_GA_1: Generazione Credenziali Amministratore Comunale
- Gestione Segnalazioni Cittadino
- UC_GSC_1: Invio Segnalazione
- UC_GSC_3: Filtro Cittadino
- Gestione Segnalazione Amministratore Comunale
- UC_GSA_3: Modifica Segnalazione
- UC_GSA_4: Assegnazione Priorità

Le funzionalità di cui non si andrà ad effettuare le attività di testing riguardano requisiti funzionali di bassa o media priorità; sono inoltre escluse le funzionalità che non prevedono input manuale da parte dell'utente - ad esempio attività riguardanti esclusivamente visualizzazioni di dati.

5. Pass/Fail criteria

Le attività di testing sono mirate ad identificare la presenza di faults (errori) all'interno del sistema, per effettuarne un successivo intervento di eliminazione.

L'esito di un test case è valutato mediante un oracolo, inteso come il risultato atteso della sua esecuzione, basandosi sui requisiti.

Un **test** ha **successo (pass)** se, dato un input al sistema, l'output ottenuto è **diverso** dall'output atteso dall'oracolo.

Un **test fallisce (fail)** se, dato un input al sistema, l'output ottenuto è **uguale** all'output atteso dall'oracolo.



Il testing sarà considerato valido se e solo se tutti i test di unità sugli otto use case identificati avranno successo.

6. Approccio al testing

L'attività di testing per il sistema **CiviConnect** sarà focalizzata esclusivamente sul **testing di unità**. L'obiettivo principale è garantire la correttezza delle unità funzionali individuate nei casi d'uso attraverso un approccio strutturato e metodico, nel rispetto delle risorse e dei vincoli definiti dal progetto.

6.1 Unit Testing

Il testing di unità sarà condotto utilizzando la tecnica di **Category Partition** per i casi d'uso descritti, con l'obiettivo di identificare tutte le combinazioni significative di input e condizioni. Verranno testati gli otto casi d'uso individuati, suddivisi per i requisiti funzionali.

Approccio:

- **Category Partition:**

I casi di test saranno derivati dalla combinazione di:

- Input validi e invalidi.
- Condizioni di stato specifiche per il caso d'uso.
- Scenari descritti nei flussi principali e alternativi.

- **Strumento di Testing:**

Sarà utilizzato il framework test di Dart per scrivere e documentare i test. Ogni caso di test sarà implementato come una funzione Dart all'interno delle classi corrispondenti.

- Esecuzione e Revisione:
- I test saranno sviluppati in parallelo con il codice per garantire la copertura continua delle funzionalità.
- Ogni test sarà successivamente revisionato da un membro del team diverso dal suo autore, per garantire qualità e correttezza.
- Obiettivo dell'Unit Testing



- Validare la correttezza delle unità software.
- Rilevare errori nel comportamento delle unità, isolandoli in modo rapido.
- Garantire la conformità delle unità rispetto ai requisiti funzionali derivati dagli use case.

6.2 Functional Validation

L'unit testing sarà utilizzato anche come metodo per validare indirettamente i requisiti funzionali attraverso i casi d'uso testati. Ogni caso di test rappresenterà un sottoinsieme delle funzionalità richieste dal sistema, permettendo una validazione completa e affidabile del comportamento previsto.

7. Testing schedule

Le attività di testing per **CiviConnect** seguiranno un piano definito per garantire un processo strutturato e allineato alle fasi dello sviluppo del sistema.

7.1 Pianificazione

Le attività di testing inizieranno **subito dopo la fase di design** necessaria per la pianificazione e proseguiranno in parallelo con lo sviluppo del codice.

I casi di test saranno definiti contestualmente allo sviluppo delle unità funzionali, per garantire una copertura tempestiva e una maggiore qualità del codice.

7.2 Esecuzione dei Test

I test verranno eseguiti in due fasi principali:

- **Durante l'implementazione:** per individuare tempestivamente eventuali errori e garantire la conformità delle unità funzionali.
- **Dopo il completamento dello sviluppo:** per rieseguire tutti i test e produrre report finali che certifichino la correttezza delle unità sviluppate.

In caso di modifiche o aggiornamenti al codice, i test saranno rieseguiti per validare il mantenimento delle funzionalità previste.



I test saranno integrati in una pipeline di CI/CD (Continuous Integration/Continuous Deployment) con GitHub Actions, che garantirà:

- **Build automatiche:** il codice sarà costruito e verificato automaticamente ad ogni modifica.
- **Esecuzione automatica dei test:** tutti i casi di test saranno eseguiti durante ogni ciclo di build.
- **Controllo Checkstyle:** verrà applicato un controllo automatico dello stile del codice per garantirne la conformità ai principi di leggibilità e manutenibilità.

7.3 Report Finali

Alla conclusione dell'esecuzione dei test, saranno prodotti report finali che documenteranno:

- La copertura dei casi d'uso testati.
- Gli esiti di ciascun caso di test.
- Gli eventuali problemi riscontrati e risolti.

Per ulteriori dettagli sullo schedule del progetto, si rimanda ai documenti di management relativi alla pianificazione.

8. Sospensione e Ripristino

In questa sezione verranno specificati i criteri di sospensione del test e le attività di test che dovranno essere ripetute quando si riprende il test.

Criteri di sospensione

Il testing non verrà sospeso fino alla sua terminazione, anche in caso di rilevazione di una failure. Il testing potrà essere momentaneamente sospeso nel caso venga restituito, al momento dell'esecuzione, un errore nella definizione di uno dei test stessi.

Criteri di ripristino

Il testing verrà ripreso dopo aver risolto i fault individuati



9. Test Cases

RF_GU_1: Login Utente - (Autore MC)

Parametro: email	
Nome Categoria	Scelte per Categoria
Lunghezza [LN]	1. Lunghezza > 255 OR Lunghezza = 0 [Error] 2. Lunghezza <= 255 && Lunghezza > 0 (campo richiesto) [Property_LN_OK]
Correttezza [CO]	1. Formattazione con caratteri speciali non ammessi [Error] 2. Formattazione con elementi mancanti [Error] 3. Formattazione: [testo.+] + [@] + [testo] + [.] + [testo] [Property_CO_OK]
Validità [VA]	1. E-mail non a sistema [Error] 2. E-mail correttamente registrata [Property_VA_OK]
Parametro: password	
Nome Categoria	Scelte per Categoria
Lunghezza [LG]	1. Lunghezza < 8 OR Lunghezza > 255 [Error] 2. Lunghezza >= 8 && Lunghezza <= 255 [Property_LG_OK]
Correttezza [CZ]	1. Almeno 1 lettera mancante tra: Maiuscola, minuscola, carattere speciale, numero [Error] 2. Almeno 1 lettera ognuna tra: Maiuscola, minuscola, carattere speciale, numero [Property_CZ_OK]



Validità [VD]	<ol style="list-style-type: none">1. Password (hash) non a sistema [Error]2. Password corretta [Property_VD_OK]
---------------	--

Note:

- Context: parametro usato per reindirizzamento non validazione, non è un input utente;
- Alcuni controlli sfruttano il plugin: form_builder_validators.

Test Frames:

Test Case ID	Test Frame	Esito
TC_1.0_1	LN1	Errato: Lunghezza e-mail non valida
TC_1.0_2	LN2, CO1	Errato: Formato non valido
TC_1.0_3	LN2, CO2	Errato: Elementi mancanti
TC_1.0_4	LN2, CO3, VA2, LG1	Errato: Lunghezza password non valida
TC_1.0_5	LN2, CO3, VA2, LG2, CZ1	Errato: manca carattere richiesto
TC_1.0_6	LN2, CO3, VA1	Errato: E-mail non valida
TC_1.0_7	LN2, CO3, VA2, LG2, CZ2, VD1	Errato: Password non valida
TC_1.0_8	LN2, CO3, VA2, LG2, CZ2, VD2	Corretto

Note:

- A livello user verrà indicato e-mail/password errata per motivi di sicurezza.
- L'inversione di testing VA1/VA2 è dovuta al fatto che la validità è verificata solo dopo la corretta formattazione degli input.



RF_GU_2: Logout

Parametro: user	
Nome Categoria	Scelte per Categoria
Autenticato [AT]	1. Non è autenticato [Error]
	2. È autenticato [Property_AT_OK]

Test Frames:

Test Case ID	Test Frame	Esito
TC_2.0_1	AT1	Errato: L'utente non è autenticato
TC_2.0_2	AT2	Corretto



RF_GU_3: Registrazione Cittadino (Autore - GSP)

Parametro: email	
Nome Categoria	Scelte per Categoria
Lunghezza [LN]	1. Lunghezza > 255 OR Lunghezza = 0 [Error] 2. Lunghezza <= 255 && Lunghezza > 0 (campo richiesto) [Property_LN_OK]
Correttezza [CO]	1. Formattazione con caratteri speciali non ammessi [Error] 2. Formattazione con elementi mancanti [Error] 3. Formattazione: [testo.+] + [@] + [testo] + [.] + [testo] [Property_CO_OK]
Validità [VA]	1. E-mail a sistema [Error] 2. E-mail correttamente registrata [Property_VA_OK]
Parametro: password	
Nome Categoria	Scelte per Categoria
Lunghezza [LG]	1. Lunghezza < 8 OR Lunghezza > 255 [Error] 2. Lunghezza >= 8 && Lunghezza <= 255 [Property_LG_OK]
Correttezza [CZ]	1. Almeno 1 lettera mancante tra: Maiuscola, minuscola, carattere speciale, numero [Error] 2. Almeno 1 lettera ognuna tra: Maiuscola, minuscola, carattere speciale, numero [Property_CZ_OK]
Parametro: Nome	
Nome Categoria	Scelte per Categoria
Lunghezza [LH]	1. Lunghezza > 255 OR Lunghezza = 0 [Error] 2. Lunghezza <= 255 && Lunghezza > 0 (campo richiesto) [Property_LH_OK]
Correttezza [CR]	1. Formattazione con caratteri speciali e numerici non ammessi [Error] 2. Solo lettere ammesse [Property_CR_OK]



Parametro: Cognome	
Nome Categoria	Scelte per Categoria
Lunghezza [LE]	1. Lunghezza = 0 OR Lunghezza > 255 [Error] 2. Lunghezza > 0 && Lunghezza <= 255 [Property_LE_OK]
Correttezza [CE]	1. Formattazione con caratteri speciali e numerici non ammessi [Error] 2. Solo lettere ammesse [Property_CE_OK]
Parametro: Città	
Nome Categoria	Scelte per Categoria
Lunghezza [LUN]	1. Lunghezza > 255 OR Lunghezza = 0 [Error] 2. Lunghezza <= 255 && Lunghezza > 0 (campo richiesto) [Property_LUN_OK]
Correttezza [COZ]	1. Formattazione con caratteri speciali e numerici non ammessi [Error] 2. Solo lettere ammesse [Property_COZ_OK]
Validità [VAL]	1. Città non presente [Error] 2. Città presente [Property_VAL_OK]
Parametro: CAP	
Nome Categoria	Scelte per Categoria
Lunghezza [LGZ]	1. Lunghezza != 5 [Error] 2. Lunghezza = 5 [Property_LGZ_OK]
Correttezza [CZZ]	1. Non contiene solo numeri [Error] 2. Contiene solo numeri [Property_CZZ_OK]
Validità [VDD]	1. Non associato ad una città [Error] 2. Associato ad una città [Property_VDD_OK]
Parametro: Via	
Nome Categoria	Scelte per Categoria
Lunghezza [LHZ]	1. Lunghezza > 255 OR Lunghezza = 0 [Error] 2. Lunghezza <= 255 && Lunghezza > 0 (campo richiesto) [Property_LHZ_OK]
Parametro: Civico	



Nome Categoria	Scelte per Categoria
Lunghezza [LGA]	1. Lunghezza = 0 OR Lunghezza > 10 [Error] 2. Lunghezza > 0 && Lunghezza <= 10 [Property_LN_OK]
Correttezza [CZD]	1. Non contiene solo numeri [Error] 2. Contiene solo numeri [Property_CZD_OK]

Test Frames:

Test Case ID	Test Frame	Esito
TC_3.0_1	LN1	Errato: Lunghezza e-mail non valida
TC_3.0_2	LN2, CO1	Errato: Formato non valido
TC_3.0_3	LN2, CO2	Errato: Elementi mancanti
TC_3.0_4	LN2, CO3, VA2, LG1	Errato: Lunghezza password non valida
TC_3.0_5	LN2, CO3, VA2, LG2, CZ1	Errato: manca carattere richiesto
TC_3.0_6	LN2, CO3, VA1	Errato: E-mail non valida
TC_3.0_7	LN2, CO3, VA2, LG2, CZ2, LH1	Errato: Lunghezza nome non valida
TC_3.0_8	LN2, CO3, VA2, LG2, CZ2, LH2, CR1	Errato: Formattazione non valida
TC_3.0_9	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE1	Errato: Lunghezza cognome non valida
TC_3.0_10	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE1	Errato: Formattazione non valida
TC_3.0_11	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN1	Errato: Lunghezza città non valida
TC_3.0_12	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ1	Errato: Formattazione città errata
TC_3.0_13	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ2, VAL1	Errato: Città non valida
TC_3.0_14	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ2, VAL2, LGZ1	Errato: Lunghezza CAP errato
TC_3.0_15	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ2, VAL2, LGZ2, CZZ1	Errato: Formato CAP non valido
TC_3.0_16	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ2, VAL2, LGZ2, CZZ2, VDD1	Errato: CAP non valido



TC_3.0_17	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ2, VAL2, LGZ2, CZZ2, VDD2, LHZ1	Errato: Lunghezza VIA non valida
TC_3.0_18	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ2, VAL2, LGZ2, CZZ2, VDD2, LHZ2, LGA1	Errato: Lunghezza Civico non valida
TC_3.0_19	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ2, VAL2, LGZ2, CZZ2, VDD2, LHZ2, LGA2, CZD1	Errato: Civico non contiene solo numeri
TC_3.0_20	LN2, CO3, VA2, LG2, CZ2, LH2, CR2, LE2, CE2, LUN2, COZ2, VAL2, LGZ2, CZZ2, VDD2, LHZ2, LGA2, CZD2	Corretto



RF_GA_1: Generazione Credenziali (Autore - GSO)

Parametro: email Comune	
Nome Categoria	Scelte per Categoria
Correttezza [CO]	1. Formattazione con caratteri speciali non ammessi [Error] 2. Formattazione con elementi mancanti [Error] 3. Formattazione: [testo] + [@] + [testo] + [.] + [testo] [Property_CO_OK]
Parametro: password Admin	
Nome Categoria	Scelte per Categoria
Lunghezza [LG]	1. Lunghezza < 8 OR Lunghezza > 255 [Error] 2. Lunghezza >= 8 && Lunghezza <= 255 [Property_LG_OK]
Correttezza [CR]	1. Almeno 1 lettera mancante tra: Maiuscola, minuscola, carattere speciale, numero [Error] 2. Almeno 1 lettera ognuna tra: Maiuscola, minuscola, carattere speciale, numero [Property_CR_OK]
Validità [VD]	1. Password (hash) non a sistema [Error] 2. Password corretta [Property_VD_OK]
Parametro: email Comune generata	
Nome Categoria	Scelte per Categoria
Correttezza [CT]	1. Formattazione con caratteri speciali non ammessi [Error] 2. Formattazione con elementi mancanti [Error] 3. Formattazione: ["comune."] + [nome del comune] + ["@anci.gov"] [Property_CT_OK]
Parametro: password Comune generata	
Nome Categoria	Scelte per Categoria
Correttezza [CZ]	1. Almeno 1 lettera mancante tra: Maiuscola, minuscola, carattere speciale, numero [Error] 2. Almeno 1 lettera ognuna tra: Maiuscola, minuscola, carattere speciale, numero [Property_CZ_OK]
Parametro: Comune	



Nome Categoria	Scelte per Categoria
Validità [VA]	1. Comune già presente [Error] 2. Comune non presente [Property_VA_OK]

Test Case ID	Test Frame	Esito
TC_4.0_1	CO1	Errato: Formato non valido
TC_4.0_2	CO2	Errato: Elementi mancanti
TC_4.0_3	CO3, LG1	Errato: Lunghezza password non valida
TC_4.0_4	CO3, LG2, CR1	Errato: Formato non valido
TC_4.0_5	CO3, LG2, CR2, VD1	Errato: Password non valida
TC_4.0_6	CO3, LG2, CR2, VD2, CT1	Errato: Formato non valido
TC_4.0_7	CO3, LG2, CR2, VD2, CT2	Errato: Elementi mancanti
TC_4.0_8	CO3, LG2, CR2, VD2, CT3, CZ1	Errato: Formato non valido
TC_4.0_9	CO3, LG2, CR2, VD2, CT3, CZ2, VA1	Errato: Comune con credenziali già generate
TC_4.0.10	CO3, LG2, CR2, VD2, CT3, CZ2, VA2	Corretto



RF_GSC_1: Invio Segnalazione (Autore - Domenico Auriemma)

Parametro: descrizione Segnalazione	
Nome Categoria	Scelte per Categoria
Lunghezza [LN]	1. Lunghezza > 1023 OR Lunghezza = 0 [Error] 2. Lunghezza <= 1023 && Lunghezza > 0 (campo richiesto) [Property_LN_OK]
Correttezza [CR]	1. Inserite parole non adeguate [Error] 2. Inserite parole adeguate [Property_CR_OK]
Parametro: titolo Segnalazione	
Nome Categoria	Scelte per Categoria
Lunghezza [LG]	1. Lunghezza > 255 OR Lunghezza = 0 [Error] 2. Lunghezza <= 255 && Lunghezza > 0 (campo richiesto) [Property_LG_OK]
Correttezza [CZ]	1. Inserite parole non adeguate [Error] 2. Inserite parole adeguate [Property_CR_OK]
Parametro: categoria Segnalazione	
Nome Categoria	Scelte per Categoria
Correttezza [CO]	1. Testo diverso da: "Rifiuti" OR "Dissesto stradale" OR "Manutenzione" OR "Illuminazione" [Error] 2. Testo uguale a: "Rifiuti" OR "Dissesto stradale" OR "Manutenzione" OR "Illuminazione" [Property_CO_OK]
Parametro: indirizzo Segnalazione	
Nome Categoria	Scelte per Categoria
Validità [VA]	1. Coordinate in range non ammissibili [Error] 2. Indirizzo restituito incorrettamente [Error] 3. Categoria valida [Property_VA_OK]
Parametro: foto Segnalazione	
Nome Categoria	Scelte per Categoria
Validità [VL]	1. Estensione non ammessa [Error] 2. Foto valida [Property_VL_OK]



Note:

- La data di invio viene impostata in automatico dal sistema
- Lo status della segnalazione viene impostato come "In Verifica" in automatico dal sistema
- La priorità della segnalazione viene impostata come "Non impostata" in automatico dal sistema
- Alcuni controlli sfruttano il plugin: form_builder_validators.

Test Frames:

Test Case ID	Test Frame	Esito
TC_5.0_1	LN1	Errato: Lunghezza descrizione non valida
TC_5.0_2	LN2, CR1	Errato: Parole nella descrizione non adeguate
TC_5.0_3	LN2, CR2, LG1	Errato: Lunghezza titolo non valida
TC_5.0_4	LN2, CR2, LG2, CZ1	Errato: Parole nel titolo non adeguate
TC_5.0_5	LN2, CR2, LG2, CZ2, CO1	Errato: Categoria incorretta
TC_5.0_6	LN2, CR2, LG2, CZ2, CO2, VA1	Errato: Coordinate non ammissibili
TC_5.0_7	LN2, CR2, LG2, CZ2, CO2, VA2	Errato: Indirizzo ricavato restituito in modo non valido
TC_5.0_8	LN2, CR2, LG2, CZ2, CO2, VA3, VL1	Errato: Estensione foto non ammessa
TC_5.0_9	LN2, CR2, LG2, CZ2, CO2, VD3, VL2	Corretto



RF_GSC_3: Filtro Cittadino (Autore - Francesco Faiella)

Parametro: categoriaSegnalazioni	
Nome Categoria	Scelte per Categoria
Validità [VC]	<ol style="list-style-type: none">1. Categoria diversa da: "Rifiuti" OR "Dissesto stradale" OR "Manutenzione" OR "Illuminazione" [Error]2. Categoria uguale a: "Rifiuti" OR "Dissesto stradale" OR "Manutenzione" OR "Illuminazione" [Property_VA_OK]
Parametro: dateSegnalazioni	
Nome Categoria	Scelte per Categoria
Validità [VD]	<ol style="list-style-type: none">1. Data non valida (Es: Date future)[Error]2. Data valida [Property VD_OK]
Formattazione [FD]	<ol style="list-style-type: none">1. Formato data non valido [Error]2. Formato data valido: 10 dicembre 2024 alle ore 18:22:55 UTC+1 [Property_FD_OK]
Parametro: ricercaSegnalazioni	
Nome Categoria	Scelte per Categoria
Lunghezza [LR]	<ol style="list-style-type: none">1. Lunghezza > 255 [Error]2. Lunghezza=0 Lunghezza <=255 [Property_LR_OK]
Parametro: comuneSegnalazioni	
Nome Categoria	Scelte per Categoria
Lunghezza [LA]	<ol style="list-style-type: none">1. Lunghezza > 255 [Error]2. Lunghezza=0 Lunghezza <=255 [Property_LR_OK]
Parametro: prioritàSegnalazioni	
Nome Categoria	Scelte per Categoria
Validità [VP]	<ol style="list-style-type: none">1. Priorità diversa da: "Bassa" OR "Media" OR "Alta" [Error]



	2. Priorità valida ("Bassa" OR "Media" OR "Alta") [Property_VP_OK]
Parametro: statoSegnalazioni	
Nome Categoria	Scelte per Categoria
Validità [VS]	1. Stato diverso da: "In Verifica" OR "Accettata" OR "In lavorazione" OR "Completata" OR "Scartata" [Error] 2. Stato deve essere valido ("In Verifica" OR "Accettata" OR "In lavorazione" OR "Completata" OR "Scartata") [Property_VS_OK]

Test Frames:

Test Case ID	Test Frame	Esito
TC_6.0_1	VC1	Errore: Categoria non valida
TC_6.0_2	VC2 FD1	Errore: Formato data non valido
TC_6.0_3	VC2 FD2 VD1	Errore: Data non valida
TC_6.0_4	VC2 FD2 VD2 VP1	Errore: Priorità non valida
TC_6.0_5	VC2 FD2 VS2 VP2 VS1	Errore: Stato non valido
TC_6.0_6	VC2 FD2 VS2 VP2 VS2	Corretto
TC_6.0_7	LR1	Errore: Lunghezza ricerca non corretta
TC_6.0_8	LR2	Corretto
TC_6.0_9	LA1	Errore: Lunghezza Comune non corretta
TC_6.0_10	LA2	Corretto
TC_6.0_11	VC2 FD2 VS2 VP2 VS2	Corretto



RD_GSA_3: Modifica Stato Segnalazione (Autore - Gianluigi Citro)

Parametro: statoSegnalazione	
Nome Categoria	Scelte per Categoria
Correttezza [CO]	1. Stato diverso da: "In Verifica", "Accettata", "In lavorazione", "Completata", "Scartata" → [Error] 2. Stato valido tra i consentiti → [Property_CO_OK]
Validità [VA]	1. Da Rifiutata o Completata a qualsiasi altro stato → [Error] 3. Da qualsiasi stato non In Verifica a Rifiutata → [Error] 2. Verso stato precedente (es. da Completata a In lavorazione) → [Error] 4. Non rispetta ordine Consentito → [Error] 5. Transizione consentita: - Da In Verifica a Rifiutata → [Property_VA_OK] - Da In Verifica a Accettata → [Property_VA_OK] - Da Accettata a In lavorazione → [Property_VA_OK] - Da In lavorazione a Completata → [Property_VA_OK]

Test Frames:

Test Case ID	Test Frame	Esito
TC_7.0_1	CO1	Errato: Stato non valido.
TC_7.0_2	CO2, VD1	Errato: Transizione da stato Rifiutata o completata non consentita.
TC_7.0_3	C02, VD2	Errato: Transizione verso stato Rifiutata da uno stato non In Verifica non consentita.
TC_7.0_4	C02, VD3	Errato: Transizione verso stato precedente.
TC_7.0_5	C02, VD4	Errato: Transizione non rispetta ordine consentiti.
TC_7.0_6	C02, VD5	Corretto



RF_GSA_4: Assegnazione Priorità (Autore - Marco Brescia)

Parametro: priorità	
Nome Categoria	Scelte per Categoria
Correttezza [CO]	1. Testo diverso da: “Bassa” OR “Media” OR “Alta” [Error] 2. Testo uguale a: “Bassa” OR “Media” OR “Alta” [Property_CO_OK]

Test Frames:

Test Case ID	Test Frame	Esito
TC_8.0_1	CO1	Errato: testo errato.
TC_8.0_2	CO2	Corretto