

**OikoNaos**  
**Problem Statement**  
**Versione 2.3**



Data: 13/10/2025

Progetto: OikoNaos	Versione: 2.3
Documento: Problem Statement	Data: 13/10/2025

### Coordinatore del progetto:

Nome	Matricola
Luigi Potestà	0512120076

### Partecipanti:

Nome	Matricola
Luigi Potestà	0512120076
Maria Antonietta Ruggiero	0512121126
Giulia Buonafine	0512119695
Mario Sinopoli	0512119113

<b>Scritto da:</b>	Giulia Buonafine
--------------------	------------------

## Revision History

Data	Versione	Descrizione	Autore
26/09/2025	1.0	<b>Proposta di Progetto:</b> Formazione del gruppo, inserimento dei partecipanti e prima bozza della descrizione del progetto	Giulia Buonafine
29/09/2025	1.1	<b>Proposta di Progetto:</b> Stesura del titolo di progetto, prime specifiche del progetto	Mario Sinopoli
30/09/2025	1.2	<b>Proposta di Progetto:</b> Inserimento del logo e stesura dei ruoli	Luigi Potestà
01/10/2025	1.3	<b>Proposta di Progetto:</b> Decisione e stesura ultime specifiche di progetto	Maria Antonietta Ruggiero
09/10/2025	2.0	<b>Problem Statement:</b> Scrittura dominio del problema, definizione audience e scopo del sistema	Mario Sinopoli
10/10/2025	2.1	<b>Problem Statement:</b> Definizione dominio del problema e requisiti funzionali	Luigi Potestà
10/10/2025	2.2	<b>Problem Statement:</b> Realizzazione scenari e inserimento scadenze	Giulia Buonafine
13/10/2025	2.3	<b>Problem Statement:</b> Inserimento requisiti non funzionali e ridefinizione ambiente di sviluppo	Maria Antonietta Buonafine

Indice

1. Purpose ..... 4

2. Audience ..... 4

3. Outline ..... 4

    3.1 Problem domain ..... 4

    3.2 Scenarios ..... 4

    3.3 Functional Requirements ..... 6

    3.4 Nonfunctional requirements ..... 7

    3.5 Target Enviroment ..... 8

    3.6 Deliverable & deadlines..... 8

## 1. Purpose

OikoNaos è una piattaforma volta alla gestione e organizzazione di una comunità di Co-housing a cura di un'Azienda Coordinatrice. La piattaforma offre diversi servizi utili per la comunità gestiti da diverse figure: Supervisionatore di ticket critici e Monitoratori di spese, turni, eventi e risorse.

L'azienda è proprietaria dell'intera struttura, comprendente sia gli appartamenti che gli spazi comuni. I coinquilini della co-housing pagano una quota trimestrale, gestibile online tramite il proprio profilo personale, che include tutti i servizi della comunità, l'uso degli spazi condivisi e le bollette. Ogni coinquilino, munito di credenziali apposite, ha la possibilità di monitorare le attività attraverso una bacheca e un calendario per la prenotazione di spazi comuni, di chiedere assistenza, gestire risorse condivise e spese comuni.

## 2. Audience

- Cliente:
  - Azienda Coordinatrice: organizzazione esterna che coordina le comunità di co housing e richiede la piattaforma.
- Utenti finali:
  - Coinquilini: utenti che utilizzano la piattaforma per gestire gli spazi comuni, i beni e le richieste di assistenza.
- Project Management e Analisti di Sistema: Gruppo OikoNaos di sviluppo, si occupa di raccogliere le esigenze, progettare e implementare.
  - Componenti: Luigi Potestà, Maria Antonietta Ruggiero, Giulia Buonafine, Mario Sinopoli.

## 3. Outline

### 3.1 Problem domain

OikoNaos migliora la qualità e la gestione delle co-housing concretizzando in un'unica piattaforma tutte le attività della comunità. Il calendario condiviso, con la possibilità di aggiungere prenotazioni, riduce i conflitti su spazi e risorse comuni; il sistema di invio di ticket traccia le richieste di manutenzione attraverso campi quali priorità, stato per tempi di risposta più rapidi; la bacheca eventi invece favorisce comunicazioni chiare. La gestione delle spese offre ripartizioni sicure tra iresidenti con storici e resoconti trasparenti, garantendo così la fiducia reciproca. La bacheca con gli annunci garantisce maggiore partecipazione alle iniziative che coinvolgono la comunità. Tutto questo permette un migliore coordinamento e senso di comunità, con benefici garantiti da tempi di risoluzione brevi, equità nelle spese e serenità dei residenti.

### 3.2 Scenarios

#### Autenticazione

Il coinquilino Matteo vuole effettuare il login sulla piattaforma web per poter visionare il proprio profilo personale. Matteo apre il browser e inserisce nella pagina di login la propria email [matteo.rossi@gmail.com](mailto:matteo.rossi@gmail.com), la password associata al suo account e il codice dato dall'Azienda Coordinatrice per identificare i residenti della comunità. Matteo quindi clicca sul pulsante "Accedi", il sistema verifica le credenziali e, se corrette, lo reindirizza alla home page personale. In caso di credenziali errate, il sistema mostra un messaggio di errore che invita a riprovare. L'utente ha inoltre la possibilità di recuperare la password tramite un link temporaneo inviato all'email

registrata, qualora non riuscisse ad accedere.

### **Prenotazione spazio comune**

Il coinquilino Luca vuole prenotare una postazione della sala studio. Le postazioni sono in totale 10 con tre fasce orarie disponibili per ciascuna: 8:00-12:00, 12:00-15:00, 15:00-18:00. Il coinquilino effettua il login nel sistema inserendo la propria email: [luca.g@gmail.com](mailto:luca.g@gmail.com) e la sua password 3050ciao!) accedendo così alla home page.

Dal menu a tendina seleziona il calendario. Luca successivamente seleziona la sala studio, la postazione 2, il giorno della prenotazione (11/10/2025) e la fascia oraria desiderata (12:00-15:00) per poi confermare la sua scelta. Il sistema ora presenterà tale postazione nell'orario e nel giorno specifico di colore rosso dunque non selezionabile.

### **Modifica prenotazione sala studio**

Il coinquilino Luca apre la sezione "Calendario prenotazioni". Scorrendo le sue prenotazioni attive, nota di aver prenotato erroneamente la sala studio per il giorno 15/10/2025 dalle 8:00 alle 12:00, ma desidera spostarla al pomeriggio. Il coinquilino seleziona quindi la prenotazione in questione e clicca sull'opzione "Modifica". Il sistema gli permette di aggiornare i dati scegliendo una nuova fascia oraria (15:00–18:00) per lo stesso giorno, verificando automaticamente che la postazione sia ancora disponibile in quell'orario.

Dopo aver confermato la modifica, il sistema aggiorna il calendario e mostra un messaggio di conferma: "*Prenotazione modificata con successo*". La nuova fascia oraria risulta ora occupata (colorata in rosso) e quella precedente torna disponibile per gli altri utenti.

### **Cancellazione prenotazione sala studio**

Il coinquilino Luca decide di non utilizzare più la sala studio che aveva prenotato per il giorno 18/10/2025 dalle 12:00 alle 15:00. Dopo aver effettuato l'accesso alla piattaforma OikoNaos, entra nella sezione "Calendario prenotazioni" e seleziona la prenotazione da annullare.

Clicca sull'opzione "Cancella prenotazione" e conferma l'operazione. Il sistema chiede una breve conferma per evitare cancellazioni accidentali e, dopo l'approvazione, rimuove la prenotazione dal calendario. A questo punto, la postazione precedentemente occupata torna disponibile per gli altri coinquilini, e Matteo riceve un messaggio di conferma dell'avvenuta cancellazione visualizzabile anche nella sezione "Storico prenotazioni".

### **Segnalazione guasto (ticket)**

La coinquilina Sara ha avuto problemi relativi al guasto della propria tv. Il sistema permette di segnalare disagi attraverso la creazione di ticket per manutenzioni esterne: di essi occorre specificare categoria, descrizione, foto. Sara effettua il log-in nel sistema inserendo la propria email: [sara09@gmail.com](mailto:sara09@gmail.com) e la sua password 8960sara!! accedendo così alla home page.

Dal menu a tendina seleziona la voce "Segnalazioni" poi la categoria del guasto ("Tv"), una breve descrizione del problema e opzionalmente una foto per poi confermare la sua scelta. Il sistema ora elaborerà la richiesta a cui assegna automaticamente una priorità e inoltra il ticket al supervisore specifico. Il sistema conferma poi la creazione del ticket e assegna ad esso un codice univoco, l'utente può monitorare lo stato di elaborazione nella propria pagina personale nella sezione "Ticket" in cui verranno ci saranno aggiornamenti automatici.

L'utente può inoltre scegliere di cancellare il ticket, purché esso sia ancora nello stato APERTO, se si sono verificati errori nella compilazione della richiesta o se la segnalazione non è più necessaria.

### Cancellazione ticket di segnalazione

Il coinquilino Paolo, dopo aver segnalato un guasto al frigorifero della cucina comune tramite la piattaforma, si accorge che il problema è stato risolto autonomamente da un altro residente. Decide quindi di annullare il ticket appena inviato.

Dopo aver effettuato il login con le proprie credenziali, Matteo accede alla sezione “Ticket” del proprio profilo personale, dove è presente l’elenco delle richieste aperte. Seleziona il ticket relativo al guasto del frigorifero e clicca sull’opzione “Cancella ticket”.

Il sistema verifica che la segnalazione non sia ancora stata presa in carico dal supervisore o dagli addetti alla manutenzione; constatata la disponibilità all’annullamento, procede con l’eliminazione del ticket. Una volta completata l’operazione, il sistema mostra un messaggio di conferma: “Il ticket è stato cancellato con successo”, e la richiesta scompare dall’elenco dei ticket attivi. Se invece il ticket fosse già stato preso in carico, il sistema mostrerebbe un messaggio informativo: “Impossibile annullare il ticket: la richiesta è già in lavorazione”.

## 3.3 Functional Requirements

### 1. Autenticazione

- **RF01. Registrazione:** Il sistema deve consentire la registrazione di un nuovo coinquilino della co-housing. Durante la registrazione, l’utente deve inserire i propri dati personali (nome, cognome, email, password e eventuali informazioni di contatto) e un codice identificativo speciale fornito dall’amministrazione della co-housing.  
Il sistema deve verificare la validità del codice prima di completare la registrazione
- **RF02. Login:** Il sistema deve consentire all’utente registrato di accedere alla piattaforma inserendo le proprie credenziali (email e password). In caso di credenziali errate, deve essere mostrato un messaggio di errore.
- **RF03. Logout:** Il sistema deve permettere all’utente autenticato di terminare la propria sessione in modo sicuro, reindirizzandolo alla pagina di login.
- **RF04. Recupero password:** Il sistema deve consentire agli utenti di recuperare la propria password tramite email, generando un link temporaneo di reimpostazione.
- **RF05. Gestione profilo:** Il sistema deve permettere all’utente di visualizzare e modificare i propri dati personali (nome, email, foto profilo, password), garantendo la persistenza delle modifiche.

### 2. Calendario Prenotazioni

- **RF06. Creazione prenotazione:** Il sistema deve consentire all’utente di creare una nuova prenotazione selezionando data, ora e risorsa disponibile, registrando i dati nel database.
- **RF07. Modifica prenotazione:** Il sistema deve permettere all’utente di modificare una prenotazione esistente, aggiornando le informazioni (data, ora o risorsa) nel rispetto delle regole di disponibilità.
- **RF08. Cancellazione prenotazione:** Il sistema deve consentire all’utente di cancellare una prenotazione precedentemente creata, rendendo di nuovo disponibile la risorsa associata.

### 3. Ticket di Manutenzione

- **RF09. Apertura ticket:** Il sistema deve permettere agli utenti di segnalare un guasto o una richiesta di manutenzione compilando un modulo con descrizione, priorità e foto opzionale.
- **RF10. Cancellazione ticket:** Il sistema deve consentire all’utente di annullare un ticket aperto, purché non sia già stato preso in carico dagli addetti alla manutenzione.
- **RF11. Visualizzazione cronologia ticket:** Il sistema deve fornire all’utente un elenco dei ticket aperti, in lavorazione e chiusi, con informazioni su data, stato e eventuali risposte ricevute. Sull’elenco sono applicabili dei filtri sulla base dello stato e della del periodo di

apertura dei ticket.

#### 4. Bacheca eventi e annunci

- **RF12. Visualizzazione della bacheca:** Il sistema deve consentire a tutti gli utenti autenticati di visualizzare la bacheca degli eventi e degli annunci pubblicati dall'amministrazione.
- **RF13. Iscrizione agli eventi:** Il sistema deve permettere all'utente di iscriversi a un evento disponibile, registrando la partecipazione nel database e aggiornando il numero dei posti disponibili.
- **RF14. Visualizzazione mappa:** Il sistema deve fornire una mappa della struttura per guidare i coinquilini verso i vari ambienti.

#### 5. Gestione risorse condivise

- **RF15. Richiesta risorsa:** Il sistema deve permettere all'utente di richiedere l'utilizzo di una risorsa condivisa (es. sala comune, veicolo, attrezzatura), verificandone la disponibilità nel periodo specificato dall'utente.
- **RF16. Presa visione delle regole d'uso:** Prima di confermare la richiesta di una risorsa, il sistema deve mostrare all'utente le regole di utilizzo, richiedendone l'accettazione esplicita. In caso di infrazione è prevista una penale da pagare.
- **RF17. Visualizzazione cronologia risorse condivise:** Il sistema deve consentire all'utente di consultare lo storico delle risorse utilizzate, incluse data di eventuale presa e restituzione, orari e stato delle richieste. Nella cronologia è possibile filtrare le risorse attraverso le voci "restituite" e "in uso".

#### 6. Gestione Spese

- **RF18. Pagamento tassa trimestrale:** Il sistema deve consentire all'utente di visualizzare gli importi dovuti e di effettuare il pagamento della tassa trimestrale tramite un metodo di pagamento integrato (es. carta o bonifico), aggiornando automaticamente lo stato del pagamento.

### 3.4 Nonfunctional requirements

#### Usabilità

- **RNF01.** Il sistema deve consentire a un nuovo utente di completare la registrazione e accedere alle funzionalità principali, senza consultare documentazione esterna.
- **RNF02. Coerenza dell'interfaccia:** Tutte le schermate dell'applicazione devono utilizzare stili grafici, colori e terminologia coerenti per garantire un'esperienza d'uso uniforme.

#### Affidabilità (Dependability)

- **RNF03. Disponibilità del sistema:** Il sistema deve garantire una disponibilità operativa minima del 99% durante il periodo di attività previsto.
- **RNF04. Recupero in caso di errore:** In caso di interruzione imprevista, il sistema deve essere in grado di ripristinare lo stato precedente, recuperando lo stato precedente entro 30 secondi dal riavvio, senza perdita di dati utente.
- **RNF05. Robustezza ai guasti:** Il sistema deve gestire gli input non validi senza interrompere l'esecuzione e mostrando un messaggio d'errore.

#### Prestazioni

- **RNF06. Tempo di risposta:** Il tempo medio di risposta a una richiesta utente non deve superare i 2 secondi in condizioni di carico normale.
- **RNF07. Carico massimo supportato:** Il sistema deve poter gestire fino a 500 utenti simultanei senza degrado significativo delle prestazioni.
- **RNF08. Accuratezza dei dati:** I risultati delle operazioni di elaborazione devono garantire un

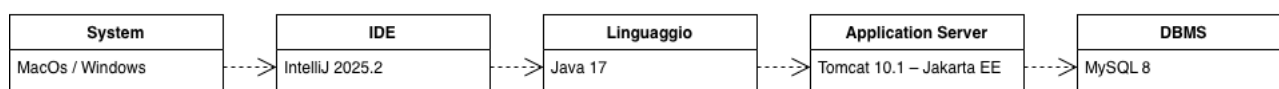
livello di accuratezza del 99,9%.

- **RNF09. Disponibilità del servizio:** Il servizio deve essere accessibile 24 ore su 24, 7 giorni su 7, salvo periodi di manutenzione pianificata.

#### Supportabilità

- **RNF10. Documentazione del codice:** Il codice sorgente deve essere commentato e documentato in conformità agli standard di sviluppo adottati. Ogni funzione deve avere almeno un commento descrittivo e un'intestazione con autore e data di modifica.
- **RNF11. Configurabilità:** Le principali impostazioni del sistema (es. parametri di connessione, lingua, limiti di sessione) devono poter essere modificate senza intervento sul codice sorgente.
- **RNF12. Internazionalizzazione:** Il sistema deve supportare la traduzione dei testi in almeno due lingue tramite file di risorse esterni.
- **RNF13. Portabilità:** L'applicazione deve poter essere eseguita su diversi ambienti server e browser (almeno Chrome, Firefox, Edge, Safari) senza modifiche al codice.

### 3.5 Target Enviroment



### 3.6 Deliverable & deadlines

<b>Dichiarazione del problema</b>	Documento che descrive il dominio del problema, gli stakeholder, gli obiettivi e i confini del sistema.	14 ottobre 2025
<b>Requisiti e casi d'uso</b>	Identificazione dei requisiti funzionali e non funzionali, definizione degli attori e dei principali casi d'uso.	28 ottobre 2025
<b>Documento di analisi dei requisiti (RAD)</b>	Analisi approfondita dei requisiti, modellazione UML, definizione delle relazioni tra casi d'uso e classi.	11 novembre 2025
<b>Documento di progettazione del sistema (SDD)</b>	Progettazione dell'architettura del sistema e dei principali moduli software.	25 novembre 2025
<b>Specifiche dell'interfaccia</b>	Descrizione tecnica delle interfacce dei moduli del sottosistema da implementare (parte del documento di progettazione degli oggetti).	16 dicembre 2025
<b>Piano di test e specifica del caso di test</b>	Definizione dei test di sistema e dei casi di test per il sottosistema implementato.	16 dicembre 2025



<b>Implementazione e integrazione</b>	Sviluppo, codifica e integrazione dei moduli software in base alla progettazione.	22 dicembre 2025
<b>Consegna e valutazione finale</b>	Consegna del sistema completo, documentazione tecnica	11 gennaio 2026