

FakeNews Checker
Problem Statement
Versione 1.0



FakeNews
Checker

Data: 13/10/2025

Progetto: Nome Progetto	Versione: X.Y
Documento: Titolo Documento	Data: GG/MM/AAAA

Coordinatore del progetto:

Nome	Matricola
Swami Falasca	0512119323

Partecipanti:

Nome	Matricola
Federica De Simone	0512120472
Giuseppe Tirelli	0512120367
Swami Falasca	0512119323

Scritto da:	Swami Falasca, Federica De Simone, Giuseppe Tirelli
--------------------	---

Revision History

Data	Versione	Descrizione	Autore
13/10/2025	1.0	Prima versione del Problem Statement	Swami Falasca, Federica De Simone, Giuseppe Tirelli

INDICE

1. Problem Statement.	4
1.1 Problem Domain	5
1.1.1 Obiettivi	5
2. Scenarios	5
3. Functional Requirements	9
4. Nonfunctional Requirements	9
5. Target Enviroment	9
6. Deliverable & Deadlines	10

1. PROBLEM STATEMENT

PURPOSE

Lo scopo di questo documento è stabilire una comprensione condivisa tra il team di progetto e il cliente in merito al problema affrontato dal sistema FakeNews Checker. Nell'era digitale, la disinformazione si diffonde rapidamente attraverso i canali online, plasmando l'opinione pubblica e spesso portando a incomprensioni o manipolazioni.

FakeNews Checker mira a contrastare questo fenomeno fornendo una piattaforma web accessibile che consente agli utenti di leggere notizie, segnalare articoli sospetti e confrontarli con fonti verificate e affidabili.

Questo documento descrive il contesto attuale del problema, delinea le funzionalità richieste, specifica l'ambiente operativo e definisce i risultati del progetto, le scadenze e i criteri di accettazione.

AUDIENCE

FakeNews Checker è destinato a:

- Il *cliente*: l'organizzazione che richiede lo sviluppo della piattaforma anti-disinformazione.
- Gli *utenti finali*:
 - Utente non registrato: consulta le notizie pubblicate
 - Utente registrato: consulta le notizie, segnala contenuti sospetti, invia articoli per la verifica e gestisce il profilo
- Gli *amministratori di sistema*:
 - Amministratore Tecnico: gestisce la piattaforma a livello tecnico
 - Amministratore delle Verifiche: esamina le segnalazioni e approva o rimuove notizie

1.1 PROBLEM DOMAIN

L'ascesa dei media digitali e delle piattaforme social ha aumentato drasticamente la velocità e la portata della diffusione delle informazioni. Sfortunatamente, questo ha anche facilitato la diffusione di fake news, contenuti falsi o fuorvianti progettati per manipolare l'opinione pubblica o generare profitto attraverso il coinvolgimento. Molti utenti non hanno gli strumenti o le conoscenze per verificare se una notizia è attendibile. L'assenza di sistemi centralizzati e di facile utilizzo per la verifica delle informazioni online contribuisce alla diffusione della disinformazione e alla confusione pubblica.

1.1.1 OBIETTIVI

Gli obiettivi della piattaforma sono:

- Fornire agli utenti una fonte affidabile di notizie verificate,
- Consentire la segnalazione e la verifica di contenuti sospetti,
- Promuovere la consapevolezza digitale e il pensiero critico tra i lettori.

2. SCENARIOS

- Scenario 1:
 - Consultazione delle notizie (Utente non registrato)
- Attori: Utente non registrato, Sistema (FakeNews Checker)
- Precondizioni:
 - L'utente dispone di un dispositivo con browser e connessione internet.
 - Esiste almeno un articolo pubblicato nella piattaforma.
- Flusso principale:
 - L'utente apre il browser e naviga nella homepage di FakeNews Checker.
 - Il sistema presenta una lista di notizie; ogni notizia mostra il titolo, un riassunto, un'immagine (se presente) e uno stato (es. "Verificata", "In verifica", "Non attendibile").
 - L'utente scorre la lista e clicca sul titolo di interesse.
 - Il sistema apre la pagina dell'articolo, mostrando il testo completo, la fonte originale (link), la data e l'autore (se disponibile).

- L'utente legge l'articolo; nella pagina trova un pulsante "Segnala come sospetta" (disabilitato per utenti non registrati).
 - L'utente decide di cercare altre notizie simili usando la barra di ricerca.
 - Se l'utente desidera partecipare (segnalare o inviare articoli), clicca su "Registrati / Accedi". Il sistema mostra il form di registrazione o di login.
 - L'utente può chiudere la sessione di lettura in qualsiasi momento.
- Flussi alternativi / eccezioni:
 - Se l'utente seleziona "Registrati", il sistema richiede e-mail e password.
- Condizioni di uscita
 - L'utente ha letto gli articoli e ha eventualmente scelto di registrarsi o no.
 - Non viene effettuata nessuna segnalazione se l'utente non è registrato.
- Scenario 2:
 - Segnalazione di notizie sospette (Utente registrato)
- Attori: Utente registrato, Sistema, Amministratore delle Verifiche
- Precondizioni:
 - L'utente è registrato e autenticato.
 - L'articolo da segnalare è raggiungibile.
- Flusso principale:
 - L'utente effettua il login e accede alla pagina dell'articolo che ritiene sospetto.
 - Clicca su "Segnala"; si apre il form di segnalazione con campi obbligatori: titolo, breve descrizione della motivazione, URL di una fonte affidabile.
 - L'utente conferma e invia la segnalazione.
 - Nella propria area personale visualizza la lista delle segnalazioni inviate e lo stato attuale di ciascuna.
- Flussi alternativi / eccezioni:
 - Se l'URL inserito non è valido, il sistema mostra un errore e suggerisce di copiare l'URL corretto.

- Condizioni di uscita:
 - Segnalazione registrata con successo e visibile al reporter nella sua area.
 - Notifica inviata all'Amministratore delle Verifiche.
 - Feedback immediato in pagina (es. "Segnalazione inviata").
- Scenario 3:
 - Verifica e aggiornamento dello stato di una notizia (Amministratore delle Verifiche)
- Attori: Amministratore delle Verifiche, Sistema, Utente registrato
- Precondizioni:
 - L'amministratore dispone delle credenziali e dell'accesso al sistema
 - Esiste almeno una segnalazione.
- Flusso principale:
 - L'Amministratore effettua il login.
 - Il sistema presenta la coda delle segnalazioni. Per ogni notizia mostra ID, titolo, URL e dati dell'utente.
 - La scheda mostra il contenuto della segnalazione.
 - L'Amministratore avvia l'attività di verifica: consulta fonti certificate (giornali ufficiali, siti di fact-checking) ed esamina le notizie.
 - Durante la verifica l'Amministratore può impostare lo stato.
 - Se la verifica produce prove sufficienti, l'articolo è "Verificato".
 - Il sistema assegna lo stato verde "Verificato" all'articolo.
 - Se l'articolo è "Non attendibile", l'Amministratore provvederà ad eliminare l'articolo.
 - Se le evidenze sono insufficienti, l'Amministratore può posticipare la decisione (stato "In verifica")
- Condizioni di uscita:
 - Ticket chiuso con esito definito (Verificato/ Non attendibile/In verifica).

- Scenario 4:
 - Gestione tecnica del sistema (Amministratore Tecnico)
- Attori: Amministratore Tecnico, Sistema, Utenti
 - L'Amministratore Tecnico ha accesso all'ambiente di sviluppo e produzione del sistema.
 - Sono attivi sistemi di database e monitoraggio dell'applicazione.
- Flusso principale:
 - L'Amministratore Tecnico accede alla dashboard locale per il monitoraggio del sistema.
 - L'Amministratore fa i suoi controlli per individuare errori o anomalie, come fallimenti di connessione o query non eseguite.
 - Applica eventuali aggiornamenti o correzioni al codice e li testa in un ambiente di prova locale.
 - Dopo i test, aggiorna la versione del sistema in produzione e verifica il corretto funzionamento della piattaforma.
 - Gestisce eventuali richieste di supporto tecnico (es. problemi di caricamento pagine o di accesso al database).
- Flussi alternativi /eccezioni:
 - In caso di malfunzionamento, il sistema mostra un messaggio di errore o l'applicazione non risponde.
 - L'Amministratore effettua i controlli per identificare la causa.
 - Riavvia i servizi necessari (server, database) e verifica che la piattaforma torni operativa.
- Condizioni di uscita:
 - Il sistema risulta operativo, con il database accessibile e le funzioni principali attive.
 - Gli eventuali incidenti tecnici sono risolti.
 - Gli altri amministratori e utenti possono riprendere l'utilizzo della piattaforma senza interruzioni.

3. FUNCTIONAL REQUIREMENTS

- Permettere agli utenti di consultare notizie pubblicate sulla piattaforma.
- Consentire agli utenti registrati di segnalare notizie sospette.
- Implementare una funzionalità per inviare articoli da sottoporre a verifica.
- Fornire agli amministratori strumenti per gestire le segnalazioni, approvare o rimuovere notizie inaffidabili e aggiornare il loro stato.
- Implementare la gestione dei profili utente (registrazione, autenticazione e modifica dati).
- Garantire un'interfaccia chiara e accessibile da diversi dispositivi (desktop, tablet, smartphone).

4. NONFUNCTIONAL REQUIREMENTS

- **Usabilità:** l'interfaccia deve essere semplice, intuitiva e accessibile anche agli utenti con competenze digitali limitate.
- **Affidabilità:** la piattaforma deve garantire l'integrità dei dati e il corretto funzionamento delle componenti di verifica.
- **Scalabilità:** il sistema deve essere in grado di gestire un numero crescente di utenti e segnalazioni.
- **Sicurezza:** i dati personali degli utenti devono essere protetti tramite autenticazione e crittografia.
- **Manutenibilità:** il codice deve essere strutturato in modo modulare, per facilitare aggiornamenti e correzioni future.

5. TARGET ENVIRONMENT

- **Tipo di sistema:** piattaforma web.
- **Ambiente di esecuzione:** browser web (Google Chrome, Mozilla Firefox, Safari, Edge).
- **Tecnologie previste:** linguaggi web standard (HTML, CSS, Java); database relazionale per la gestione dei dati.
- **Accesso:** tramite rete Internet, disponibile su dispositivi desktop e mobile.

6. DELIVERABLE & DEADLINES

DELIVERABLE	DESCRIZIONE	DATA DI CONSEGNA
Requisiti e casi d'uso	Identificazione dei requisiti funzionali e non funzionali e definizione dei principali casi d'uso.	28/10/25
Requirements Analysis Document	Analisi dettagliata dei requisiti del sistema	11/11/25
System Design Document	Documento di progettazione architeturale: struttura three-tier del sistema	25/11/25
Specifica delle interfacce dei moduli del sottosistema da implementare	Descrizione delle classi, dei metodi e delle interfacce relative al sottosistema.	16/12/25
Piano di test di sistema e specifica dei casi di test per il sottosistema da implementare	Piano di test del sistema e specifica dei casi di test.	16/12/25
Object Design Document	È il documento che descrive la progettazione di basso livello del sistema.	07/01/26
Documenti di esecuzione del test	Documenti che descrivono come è stato testato il software.	09/01/26
Consegna finale	Progetto finale: FakeNews Checker	11/01/26