



# UNIVERSITÀ DI TRENTO

Dipartimento di Ingegneria e Scienza dell'Informazione  
2022/2023

## YINCO

### Analisi dei requisiti

Doc. Name	D1 YINCO Analisi Requisiti	Doc. Number	Rev 0.6
Description	Documento di analisi dei requisiti funzionali, non funzionali, front-end e back-end		

# INDICE

1. Scopo del documento .....	3
2. Obiettivi del progetto .....	3
3. Requisiti .....	4
3.1 Requisiti funzionali .....	4
3.2 Requisiti non funzionali .....	5
4. Front-End .....	7
5. Back-End .....	12

# 1. Scopo del documento

Il presente documento riporta l'analisi dei requisiti di sistema del progetto "Yinco", in linguaggio naturale.

Il suo scopo è quello di:

- presentare gli obiettivi del progetto;
- definire i requisiti funzionali e non funzionali;
- presentare i requisiti di Front-End;
- presentare i requisiti di Back-End.

# 2. Obiettivi del progetto

Il progetto ha come fine la realizzazione di una **chatbox** impersonificata da una mascotte, la quale interagisce con gli utenti e, dopo che gli è stata chiesta un'informazione, ritorna un link ad una pagina contenente i dati riguardanti la richiesta effettuata.

Nello specifico, la web-app permetterà:

- a) a un **utente autenticato**, ovvero in possesso di credenziali dell'università di Trento, di porre alla chatbox una domanda, a cui quest'ultima risponderà con un link che condurrà l'utente ad una pagina contenente informazioni inerenti all'argomento richiesto.

In caso di mancata presenza di informazioni adeguate, verrà ritornato un messaggio di errore. ([fig. 4.4](#))

- b) a un **utente anonimo**, quindi non ancora in possesso di credenziali UniTn, di utilizzare la chatbox solo per i seguenti argomenti:

- immatricolazione;
- tasse;
- borse di studio;
- posti e alloggio;
- corsi di studio.

In caso di ricerca di informazioni che esulano da questi argomenti, la chatbox manderà una notifica di errore per mancata autorizzazione.

- c) di mandare all'utente, grazie al collegamento con esse3, delle **email** per le seguenti casistiche:

- scadenza iscrizioni esami;
- scadenza tasse;

Le seguenti mail verranno inviate una settimana prima della scadenza.

- d) all'utente autenticato di cercare il nome di un **docente**: se tale nome è presente nel database, la chatbox ritornerà il link alla pagina del docente stesso.

## 3. Requisiti

Nei seguenti paragrafi affronteremo i vari requisiti funzionali e non funzionali che Yinco dovrà soddisfare, consci del fatto che il sistema deve fare una distinzione tra due tipi di utenti:

- utente anonimo, ovvero non in possesso di credenziali UniTn;
- utente autenticato, ovvero in possesso di credenziali UniTn.

### 3.1 Requisiti Funzionali

#### **UTENTE ANONIMO**

##### **RF1: ACCESSIBILITA' AL SISTEMA**

Il sistema deve garantire ad un utente anonimo l'accesso alle sue funzioni solamente riguardo per gli argomenti descritti nell'[obiettivo b.](#)

##### **RF2: CAMBIO PRIVILEGI**

Il sistema deve garantire all'utente anonimo la possibilità di diventare un utente autenticato in qualsiasi momento, accedendo al sistema con le sue credenziali UniTn tramite un bottone di login, posto sulla barra di navigazione.

#### **UTENTE AUTENTICATO**

##### **RF3: INTERAZIONE COL SISTEMA**

Il sistema deve garantire all'utente un modo per interagire col sistema affinché l'utente possa fare a questi delle domande. In particolare il sistema avrà una sezione chatbox nella quale si troverà un box di testo grazie al quale si potrà inserire la domanda.

##### **RF4: RICERCA**

Il sistema, in base alla domanda dell'utente, effettuerà una ricerca utilizzando le parole chiave della domanda, ovvero solo l'argomento principale.

##### **RF5: ERRORE**

Se il sistema non trova nulla di pertinente all'informazione richiesta in base alla ricerca spiegata nel [RF4](#), deve ritornare un messaggio di errore all'utente attraverso la chat.

##### **RF6: RISPOSTA**

Il sistema deve ritornare all'utente l'informazione richiesta tramite un messaggio, che apparirà nella chat, ed un link, che ricondurrà alla pagina contenente le informazioni inerenti a ciò che l'utente ha richiesto.

#### **NOTICHE**

##### **RF7: NOTIFICHE VIA MAIL**

Al primo accesso dell'utente, il sistema deve chiedere a quest'ultimo se vuole ricevere email di promemoria riguardanti gli argomenti descritti nell'[obiettivo c.](#)

##### **RF8: CAMBIO PREFERENZA MAIL**

Il sistema deve garantire all'utente la possibilità di cambiare questa scelta tramite un'opzione disponibile nella pagina impostazioni. ([fig. 4.8](#))

### **RF9: INVIO MAIL**

Il sistema deve essere in grado di mandare email inerenti agli argomenti descritti nell'[obiettivo c](#) solo agli utenti che hanno aderito al sistema di invio mail ([RF7](#)).

### **ESSE3**

#### **RF10: COLLEGAMENTO CON ESSE3**

Il sistema dovrà interfacciarsi con Esse3 per accedere alle informazioni richieste negli obiettivi [b](#) e [c](#).

### **UNITN**

#### **RF11: COLLEGAMENTO CON UNITN**

Il sistema deve interfacciarsi con i sistemi dell'università di Trento per accedere alle informazioni richieste dall'[obiettivo d](#) e dal [RF2](#).

### **PAGINA SISTEMA**

#### **RF12: VERIFICA ACCESSO SITO**

Il sistema, all'apertura della pagina, deve verificare se l'utente ha già effettuato l'accesso in precedenza oppure no, in modo da capire se porre la domanda posta nel RF8.

#### **RF13: LOGIN AUTOMATICO**

Se il sistema si accorge che l'utente aveva già effettuato il login, deve effettuare autonomamente il login.

#### **RF14: CONTATTI**

Il sistema, attraverso una sezione apposita, deve mostrare la mappa che indica la sede dell'università, e i contatti dei fondatori di Yinco.

#### **RF15: LOGOUT**

Il sistema deve dare la possibilità all'utente autenticato di effettuare il logout, tornando ad essere un utente anonimo, in qualsiasi momento attraverso una funzione presente nella sezione impostazioni.

## **3.2 Requisiti Non Funzionali**

### **RNF1: PRESTAZIONI**

Il sistema deve ritornare all'utente una risposta in un tempo non superiore ai due secondi.

### **RNF2: COMPATIBILITÀ**

Il sistema deve essere compatibile con le versioni dei browser più usati (Firefox, Chrome, Edge, Safari) dalle versioni del 2021 in poi.

### **RNF3: PORTABILITÀ**

Il sistema deve poter funzionare tramite browser sia su Personal Computer che su dispositivi mobili, questo significa che il sito deve adattare responsivamente la propria interfaccia alle dimensioni di ogni dispositivo.

### **RNF4: SCALABILITÀ**

Il sistema deve garantire l'elaborazione di un numero crescente di utenti, di conseguenza fornire le prestazioni definite nel [RFN1](#) anche con un numero di utenti maggiore di 500.

#### **RNF5: FLESSIBILITÀ EVOLUTIVA**

All'interno del sistema deve risultare intuitivo introdurre elementi/componenti innovativi. In particolare, deve risultare facile lavorare sul codice di sviluppo e nel caso in cui un update del sistema provochi bug o errori, deve essere possibile ritornare alla versione precedente.

#### **RNF6: USABILITÀ**

Un utente deve essere in grado di usare tutte le funzioni fornite dal sistema in meno di 15 minuti.

#### **RNF7: TEMPO LIMITE PER IL LOGIN AUTOMATICO**

Il sistema, come detto nel [RF13](#), deve essere in grado di effettuare il login automaticamente solamente se l'utente accede nuovamente alla pagina in un tempo massimo di 10 minuti.

#### **RNF8: PRIVACY**

Il sistema deve rispettare le norme legali imposte dal [GDPR](#).

In particolare:

- Il sistema può elaborare solo i dati personali necessari al raggiungimento delle finalità per i quali sono trattati ([obiettivo c](#)).
- il trattamento dei dati da parte del sistema è limitato al solo scopo legittimo per il quale tali dati personali sono stati originariamente raccolti ([obiettivo c](#)).
- Il sistema può richiedere solo i dati personali strettamente assolutamente necessari a tale scopo([obiettivo c](#)).
- I dati personali degli utenti devono essere strettamente accurati e aggiornati, gli utenti hanno quindi il diritto di chiedere che i propri dati personali inesatti o incompleti vengano cancellati o rettificati.
- I responsabili del sistema devono eliminare i dati personali di un utente qualora non siano più necessari.

#### **RNF9: LINGUA**

Il sistema deve permettere all'utente di poter cambiare la lingua della pagina da inglese a italiano e viceversa attraverso una funzione apposita nella sezione impostazioni.

## 4. Front-End

In questo paragrafo verranno riportati alcuni **mock-up** relativi alle schermate della web-app, sia per smartphone che per PC, con l'obiettivo di rappresentare come il sistema si interfacerà all'utente. Le pagine che presenteremo sono:

- Homepage
- Pagina di Login
- Pagina di dialogo
- Pagina con le informazioni richieste
- Contatti
- Impostazioni
- Menù versione mobile

### **HOMEPAGE**

Il sistema deve presentare all'utente, appena entrato nel sito, una pagina con una descrizione del progetto, e il percorso che ha portato alla sua creazione ([fig. 4.1](#)). Inoltre, premendo il tasto **"Inizia"**, il sistema invia l'utente alla pagina contenente la chatbox, la quale cambierà a seconda che l'utente sia autenticato oppure che sia un utente anonimo.



Fig. 4.1: Homepage di Yinco



Fig. 4.2: Homepage mobile



Fig. 4.3: Menù ad "hamburger" versione mobile

## **PAGINA DI LOGIN**

La pagina utilizza le API dell'università di Trento, perciò la schermata di login sarà quella standard utilizzata dall'API. Dopo aver effettuato il login, se sarà la prima volta che l'utente accede al sito, il sistema gli presenterà una pagina dove verrà data la possibilità di attivare le notifiche via email definite nel [RF7](#). Dopo aver selezionato la propria risposta, l'utente, cliccando il tasto "OK", verrà riportato alla pagina della chatbox ([fig 4.4](#)).

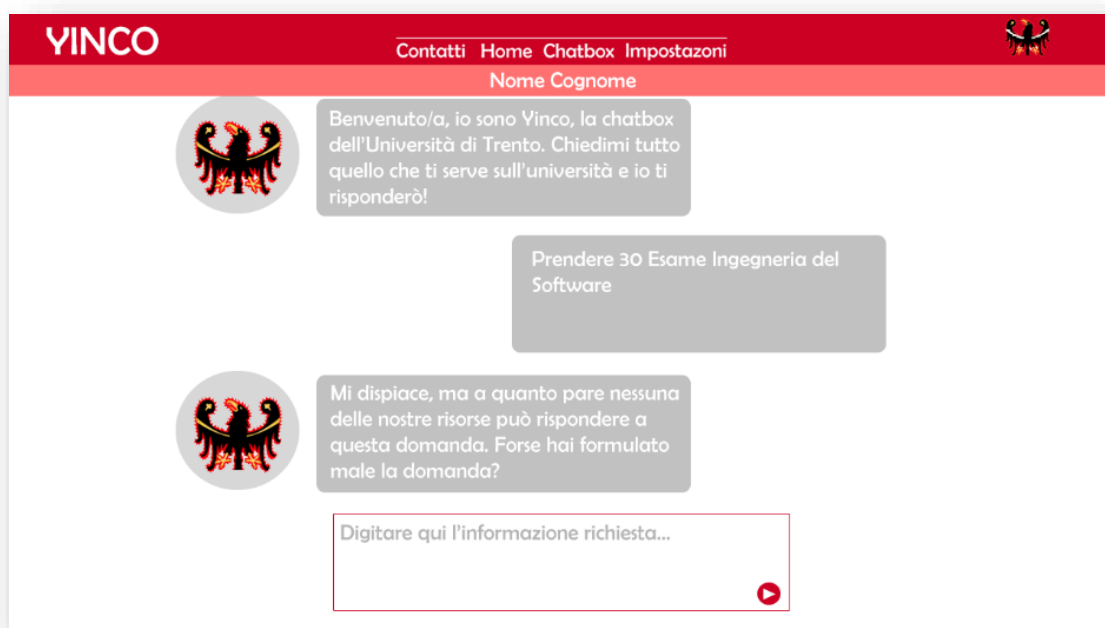
## **PAGINA DI DIALOGO**

Nella pagina di dialogo, la chatbox invia un messaggio di benvenuto all'utente, chiedendogli in cosa può essergli utile ([fig 4.4](#)). In basso l'utente avrà a disposizione una casella di testo dove scrivere la domanda di cui vuole avere risposta.

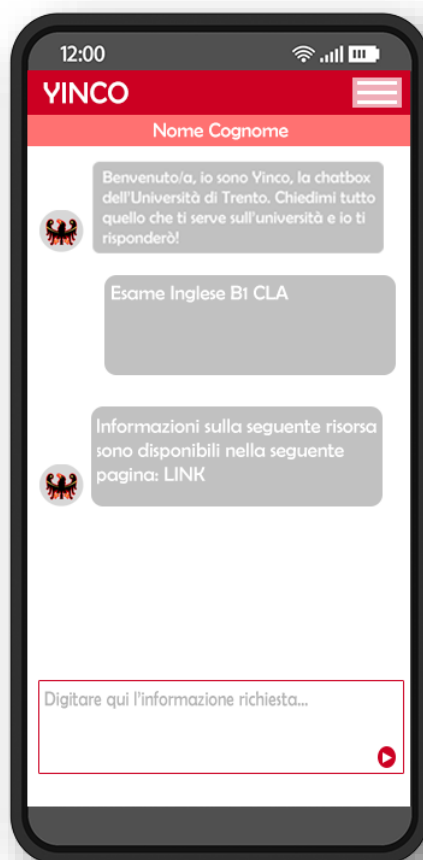
Una volta digitata la richiesta, l'utente, premendo invio, la invierà al sistema, che la visualizzerà a schermo. Il sistema, dopo aver cercato nel database l'informazione richiesta, invierà un link ad una pagina interna contenente le informazioni richieste



(fig. 4.5) oppure un messaggio di errore (fig. 4.4). Nel caso in cui l'informazione cercata è il nome di un docente, la chatbox riporterà il link alla sua pagina UniTN.



**Fig. 4.4:**  
Messaggio di  
errore



**Fig. 4.5:** Ricerca corretta e  
messaggio chatbox

## **PAGINA RISULTATO**

La pagina che la chatbox invia all'utente si presenta con un titolo (che può anche essere diverso da ciò che l'utente ha digitato) e con un testo sottostante che deve essere inerente all'argomento cercato. ([fig. 4.6](#))

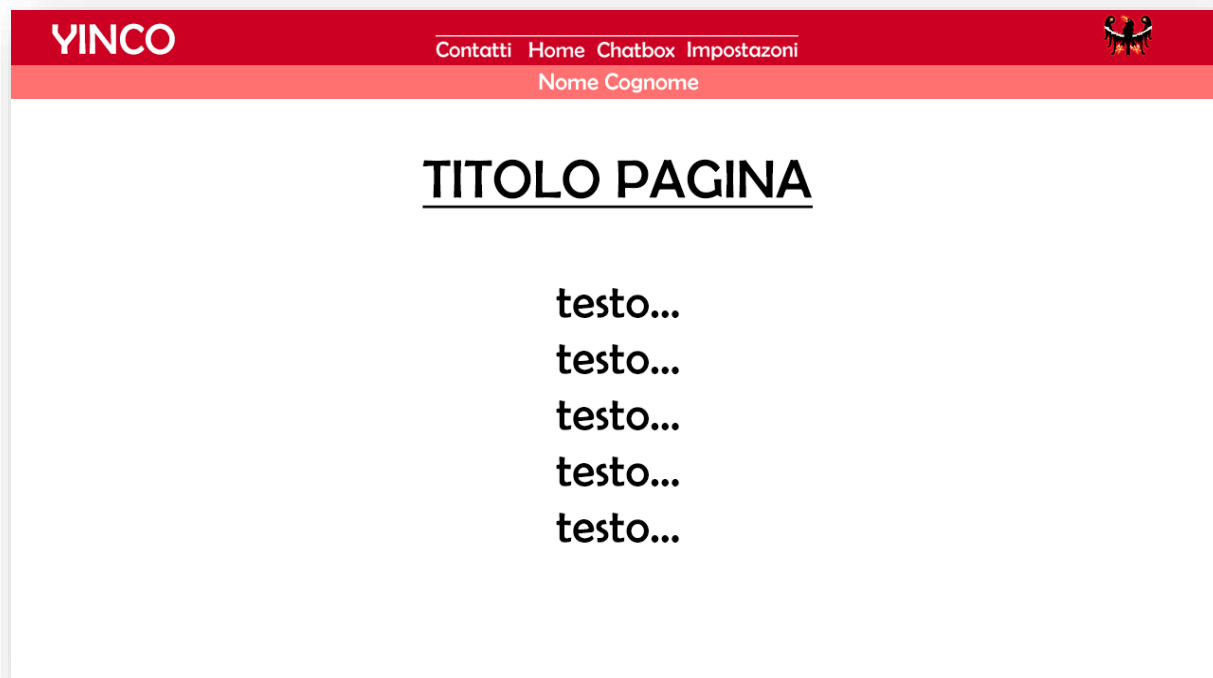


Fig. 4.6: Esempio di pagina che la chatbox invia come risposta

## **CONTATTI**


Questa pagina contiene una mappa che fa riferimento all'università di Trento come sede dell'azienda e i contatti dei responsabili del progetto. ([fig 4.7](#))



Fig. 4.7: Sezione contatti

## **IMPOSTAZIONI**

La pagina contiene le impostazioni inerenti alle notifiche via mail, che possono essere modificate a piacimento dall'utente (scegliendo tra sì e no), e un bottone che permette di effettuare il logout dal sito ([fig. 4.8](#)).



**YINCO** Contatti Home Chatbox Impostazioni Nome Cognome

Vuoi ricevere notifiche mail riguardanti la scadenza del pagamento delle tasse e delle iscrizioni agli esami? **Sì** No

Vuoi effettuare il logout? **LOGOUT**

Fig. 4.8: Sezione impostazioni

## 5. Back-End

Nel seguente paragrafo verranno riportati i sistemi esterni con cui l'applicazione dovrà interfacciarsi per poter funzionare; quest'ultimi sono:

- **UniTN API**, la quale è necessaria per usufruire del login tramite credenziali universitarie, e per accedere alla lista dei docenti, con tutte le loro informazioni;
- **ESSE3 API**, per poter accedere alle scadenze riguardanti tutti gli argomenti citati nell'obiettivo c;
- **Google Maps API**, al fine di mostrare una mappa nella sezione contatti per indicare la sede dell'azienda e dell'università dove è stato sviluppato il sistema;
- **Google Gmail API**, per poter inviare all'utente email riguardanti gli argomenti discussi nell'obiettivo c;
- Un **database** locale, per poter immagazzinare le informazioni che verranno successivamente cercate dagli utenti e per poter conservare le preferenze circa l'invio di mail.

La rappresentazione schematica sottostante mostra come il sistema deve interfacciarsi con le diverse API e con i database.

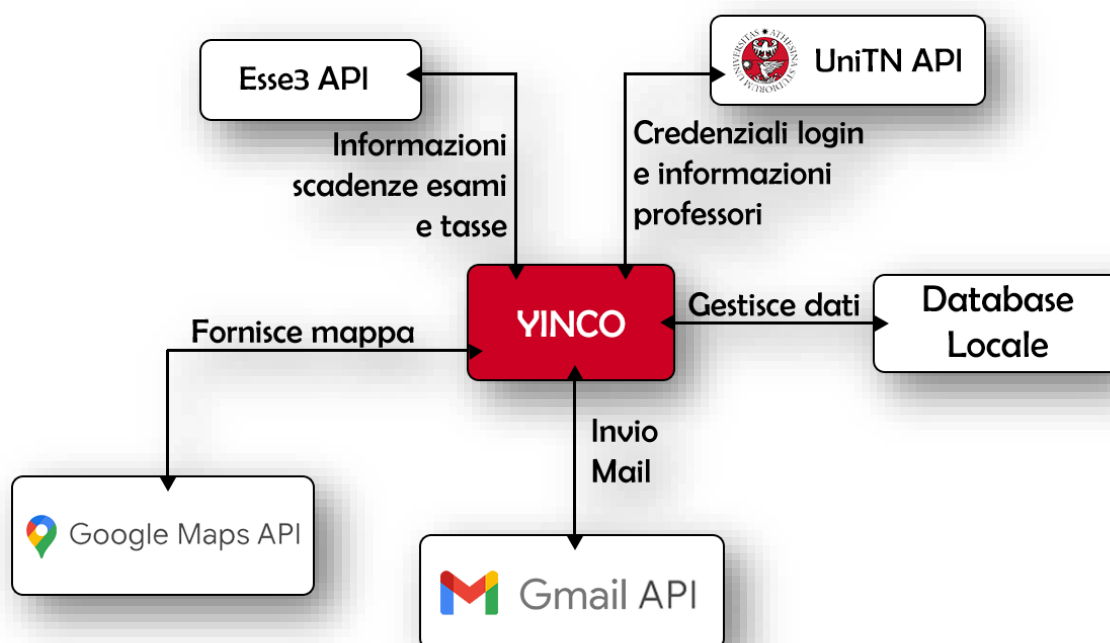


Fig. 5.1: Schema delle relazioni tra il sistema e le API esterne