

Nao Challenge 2024 Relazione Tecnica

[NAOorNever](#)

PER MAGGIORI INFORMAZIONI:

Visit our [linktree](#)

Visit our [website](#)

Visit our [repository](#)

Visit our [instagram page](#)

Contenuto:

1. [INTEGRAZIONE DI NORTON E DISCORD PER ORGANIZZAZIONE E COMUNICAZIONE](#)
 - 1.1. [Notion](#)
 - 1.2. [Discord](#)
2. [NAO COME CONSULENTE](#)
 - 2.1. [Comunicazione client/server](#)
 - 2.2. [Monitor](#)
3. [ARTIFICIAL INTELLIGENCE](#)
 - 3.1. [Database](#)
 - 3.2. [AI](#)
4. [SITO WEB](#)
 - 4.1. [Descrizione](#)
 - 4.2. [Contenuti](#)
5. [NAO Manager](#)
 - 5.1. [Backend](#)
 - 5.2. [Frontend](#)
6. [BRAND IDENTITY](#)
7. [TEAM](#)
 - 7.1. [Social & Public relation division](#)
 - 7.2. [Coding division](#)
 - 7.3. [Choreograph division](#)
 - 7.4. [Design division](#)

0. INTRODUZIONE → Esigenze dell'azienda

La SICIS, nostra azienda partner, fin da subito si è mostrata collaborativa e, attraverso una serie di colloqui, online e in presenza, ci ha spiegato l'esigenza principale dell'azienda, cioè trovare una figura con competenze ibride, sia commerciali che tecniche.

Grazie a questa analisi siamo stati in grado di individuare la soluzione migliore che abbiamo soddisfatto con gli strumenti tecnologici a nostra disposizione, basando l'intero progetto sul robot umanoide NAO.

La prima necessità era quella di implementare una vendita al dettaglio in maniera automatica, ma al contempo garantendo un dialogo organico come quello umano. Dunque abbiamo deciso di impiegare una ai al fine di garantire queste condizioni. Inoltre abbiamo posto soluzione anche alla necessità dell'azienda di essere più inclusiva. Questo fondamentale punto è stato reso possibile grazie ai sottotitoli del dialogo che vengono stampati a schermo, permettendo così per la prima volta ai non udenti di fare shopping in completa autonomia. Altro aspetto di rilievo per il progetto è stato quello di garantire un'ottima tutela ambientale tramite una app che comunica col nao gestendo un inventario. In questo modo i lavoratori di sicis possono adesso monitorare con agilità gli ordini e le rimanenze di magazzino, abbattendo il numero di spedizioni (decurtando quelle superflue), migliorando l'efficienza dell'azienda.

1. INTEGRAZIONE DI NORTON E DISCORD PER ORGANIZZAZIONE E COMUNICAZIONE

Per quanto concerne questo progetto abbiamo utilizzato due potenti strumenti digitali, *Notion* e *Discord*, per ottimizzare l'organizzazione e la comunicazione all'interno di un contesto lavorativo. L'integrazione di *Notion* e *Discord* ha dimostrato di essere un approccio vincente per migliorare l'organizzazione e la comunicazione all'interno del contesto lavorativo. L'efficienza operativa è stata potenziata attraverso la centralizzazione delle informazioni su *Notion* e la comunicazione istantanea e strutturata su *Discord*. L'implementazione di queste due piattaforme ha creato un ambiente collaborativo e sinergico, contribuendo al successo delle attività del team.

1.1. *Notion* come Hub organizzativo

Notion è stato adottato come principale piattaforma per la gestione delle informazioni, delle attività e della documentazione. La sua flessibilità consente la creazione di pagine personalizzate, database e bacheche condivise, facilitando la collaborazione e la tracciabilità delle attività. *Notion* è

stato utilizzato per la gestione completa del ciclo di vita dei progetti. Attraverso la creazione di database di progetto, il team è stato in grado di monitorare lo stato delle attività, assegnare compiti e collaborare in tempo reale. La possibilità di inserire commenti, allegare file e integrare calendari ha facilitato la comunicazione e la condivisione di informazioni cruciali.

1.2. *Discord come Piattaforma di Comunicazione*

Discord è stato implementato come strumento di comunicazione principale, offrendo una combinazione di messaggistica istantanea, chiamate vocali e video. La creazione di un *server Discord* dedicato ha consentito di mantenere una comunicazione fluida e organizzata.

Il server *Discord* è stato strutturato con canali dedicati a diversi aspetti del lavoro, come *web developer*, *nao programming*, *social media*, *general discussion* e *shared resources*. Ciò ha permesso ai membri del team di accedere rapidamente alle informazioni rilevanti e di partecipare alle conversazioni specifiche.

2. NAO COME CONSULENTE

Il robot umanoide *NAO* è stato potenziato con un sistema di intelligenza artificiale, consentendogli di apprendere e interagire in modo più sofisticato con i clienti. L'intelligenza artificiale è in grado di analizzare i dati dei clienti, comprendere i loro bisogni e adattare le interazioni del robot in tempo reale. Durante la conversazione con il cliente, il robot *NAO* non è statico ma interagisce con il cliente attraverso micro movimenti per rendere la customer experience più gradevole. I movimenti riguardano saluti, inchini, movimenti nei momenti di attesa e indicazioni a schermo quando sono presenti prodotti da mostrare al cliente.

2.1. *Comunicazione client/server*

La comunicazione avviene in modo sincrono tramite client e server.

Il server apre un socket su una porta e aspetta una connessione, mentre il client sul *nao* si connette al socket e registra l'audio della conversazione.

Dopo un lasso di tempo preimpostato la registrazione finisce (con notifica sonora) e il *nao* manda il file audio al server.

Il server riceve tramite un buffer il file audio registrato dal *nao*, lo passa a *whisper* che trascrive la domanda del cliente, per poi passare il dialogo al modello *llama* che genera una risposta.

La risposta viene mandata al *nao* che per ricevere riapre una connessione al socket.

Il nao sempre tramite buffer riceve un json contenente 4 istruzioni: content (ciò che deve dire), move (l'azione che deve fare), stop (che gli dà il comando di fermarsi) e reset (che fa resettare il nao allo stato iniziale).

Il nao esegue poi le azioni in modo tale da poter muoversi mentre parla tramite parallelismo. Apre una connessione proxy locale all'api contenente le funzioni in c++ e manda il messaggio alla funzione ALTextToSpeech, che gli permette di parlare. In fine nel caso dovesse fermarsi si spegne, oppure se gli è stato ordinato si resetta. Altrimenti fa ripartire il loop di conversazione fino a quando non riceve uno di questi due comandi (basati sulla conversazione col cliente, e sulla funzionalità di acquisto di un prodotto, che lo fa resettare).

Il codice utilizzato è visionabile nel repository.

2.2. Monitor

Il monitor consiste in una pagina web che si connette al server con un websocket, il server invia i prodotti che il monitor deve mostrare e ciò che il NAO sta dicendo al cliente in modo da fornire contemporaneamente informazioni sui prodotti e sottotitoli per garantire accessibilità anche per chi ha problemi di udito.

3. ARTIFICIAL INTELLIGENCE

3.1. Database

Il database è stato realizzato tramite la piattaforma di google fogli di calcolo. Nel database sono stati inseriti i seguenti parametri dei prodotti:

1. nome
2. materiale
3. colore
4. sottocategoria, categoria, macrocategoria
5. peso
6. prezzo
7. dimensioni
8. foto prodotto
9. foto disegno tecnico
10. link sito
11. descrizione per la vendita
12. descrizione tecnica
13. varianti prodotto
14. proprietà

Questi parametri sono stati raccolti dal database in formato pdf fornito dall'azienda e dal sito ufficiale dell'azienda.

Questo database è fondamentale per il training per la AI che prende da questo le informazioni necessarie per il dialogo, questo argomento è più nel dettaglio analizzato nel capitolo dedicato alla AI.

3.2. AI

Un programma python interfaccia il NAO con il modello di ai Cerbero 4k (licenza MIT). Il nao si collega tramite socket al server e invia una registrazione della domanda posta dal cliente, dopo aver salvato il file il server lo trasforma in testo attraverso openai whisper (licenza MIT) e cerca di capire cosa sta cercando il cliente attraverso un prompt specifico al modello di ai hostato da un server llamafire (Licenza Apache). Dopo aver identificato la richiesta del cliente un prompt adatto alla domanda viene caricato nel contesto del modello e in seguito il programma invia direttamente le domande in formato di testo al modello. Quando il cliente decide di acquistare un prodotto il programma python rileva l'acquisto e resetta il contesto del modello. Durante l'interazione le risposte date dal modello vengono esaminate e eventuali prodotti nominati sono mostrati su uno schermo. L'interfacciamento con il modello é eseguito attraverso la API chatcompletion di openai.

Il codice utilizzato è visionabile nel repository.

4. SITO WEB

4.1. Descrizione

Il sito web presente all'indirizzo www.naoornever.it funge da presentazione per la squadra, riprendendo vari elementi di design della Brand Identity del team e il suo tipico colore azzurro insieme a descrivere il progetto. È realizzato con HTML, CSS e Javascript.

4.2. Contenuti

Nel sito è presente un link al video realizzato dal team, caricato su Youtube. Presente è inoltre la descrizione del progetto svolto, insieme a cenni dei componenti che formano l'elaborato svolto: cenni al NAO Monitor, cenni all'Intelligenza Artificiale utilizzata per aiutare nelle sue scelte l'acquirente, come collegamenti ai social del Team o lo Sponsor principale del progetto.

5. NAO Manager

5.1. Backend

Per la gestione del database degli ordini e dell'inventario viene utilizzato flask che implementa una rest api, e sqlalchemy, che permette di interagire con database sql indipendentemente dall'engine, che in questo caso è sqlite.

Vengono create due tabelle che permettono di popolare il database sql: Ordini e Inventario.

Al primo avvio c'è anche la possibilità di importare un database in formato csv dentro all'sql.

Il server (hostato sulla porta 5000) implementa diverse pagine con metodi diversi:

/get (GET): senza argomenti da tutti gli elementi, con arg ?nome=nome da solo un elemento

/add (POST): dati in post tutti gli argomenti aggiunge un elemento all'inventario (data: "arg1=test&arg2=test...")

/remove (GET): dato un nome come argomento lo rimuove dall'inventario

/sell (GET): dato un nome come argomento lo aggiunge agli ordini in attesa

/control (GET): elenca tutti gli ordini in attesa di conferma

/control (POST): argomenti: id e conferma (True or False), se conferma è True conferma l'ordine e rimuove l'item dall'inventario, se è False annulla l'ordine, mantenendolo però nel log degli ordini.

Nel caso il sito mostri un errore 404, vuol dire che l'id dell'ordine o il nome del prodotto non sono stati trovati.

5.2. Frontend

L'inventario dell'azienda collegato al NAO e gli ordini da lui effettuati sono visualizzati in una single-page Web App realizzata con Vue.js, che interagisce con il Backend descritto nel punto precedente.

Grazie a questa interfaccia è quindi possibile quindi gestire l'inventario dei prodotti vendibili dal NAO, oltre che visualizzare gli ordini da lui conclusi.

Da quest'ultima interfaccia, Vendite NAO, si sarà infatti inoltre avvisati quando rimangono pochi oggetti in magazzino del prodotto appena venduto, permettendo quindi una più sostenibile gestione delle risorse.

Nell'interfaccia di Inventario è invece possibile aggiungere nuovi tipi di prodotti oltre che ad aggiornare la disponibilità in magazzino di quest'ultimo, oltre ad una visione tabellare dell'intero magazzino.

6. BRAND IDENTITY

Il nome NAO or Never prende ispirazione dal motto inglese "Now or never" ("Adesso o mai più") e si basa su un simpatico gioco di parole in cui la parola "Now" viene sostituita con il nome del robot umanoide NAO, dal ruolo fondamentale nel nostro progetto. Questa frase riflette lo spirito audace della squadra, la quale ha dato il massimo al fine di cavalcare l'onda dell'innovazione e rinnovare l'ambito del retail con le tecnologie disponibili oggi, senza farsi però sommergere dai costanti miglioramenti ad esse apportati.

Il logo della squadra è costituito da un NAO contornato da mosaici, che ricordano la collaborazione con l'azienda SICIS. È stata realizzata una versione più piccola del logo, contenente esclusivamente la testa del NAO e qualche tessera di mosaico, in modo da poterla implementare in dimensioni ridotte quando più opportuno (ad esempio nella barra principale del software Monitor, collegato all'intelligenza artificiale assistente nell'acquisto). Questo rende il brand riconoscibile anche senza la necessità di usare grandi porzioni di spazio per firmarsi tramite l'uso del logo.

Importante è il significato assunto dalla lettera "v" di "Never" all'interno delle scritte rappresentative del brand: essa è composta da due fulmini incrociati tra loro, a simboleggiare l'elettricità, essenziale per l'innovazione tecnologica, ma anche la forte volontà della squadra.



7. TEAM

Il team di NAO or Never si è diviso in 3 sezioni principali: social & public relation division, coding division, choreograph division and design division, ognuna delle quali si è occupata di un ambito specifico del progetto, ma tutti hanno collaborato in modo sinergico, richiedendo feedback e partecipando a votazioni per assicurare che ogni decisione fosse presa democraticamente e condivisa dall'intero gruppo.

Inoltre ci teniamo a ringraziare Ferrari Marco e Valpiani Giampaolo, i nostri due docenti-coach, che ci hanno guidati ed aiutati nella realizzazione del progetto.

7.1. *Social & Public relation division*

- Alberani Giacomo
- Zaccaria Ragni

7.2. *Coding division*

- Guido Betti
- Federico Santucci
- Christian Fiore
- Riccardo Deserti

7.3. *Choreograph division*

- Michele Missiroli
- Luca Tramontani

7.4. *Design division*

- Martina Baldini