

ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA
CAMPUS DI CESENA

DIPARTIMENTO DI INFORMATICA – SCIENZA E INGEGNERIA
Corso di Laurea in Ingegneria e Scienze Informatiche

**FITNESS TO DRIVE NELL'ERA DIGITALE:
UN'ANALISI BASATA SUL DIGITAL TWIN
DEL GUIDATORE**

Elaborato in

APPLICAZIONI E SERVIZI WEB

Relatore

Dott. ROBERTO GIRAU

Presentata da

MARIA MENGOZZI

Corelatori

Dott. MIRRI SILVIA

**II Sessione di Laurea
Anno Accademico 2022 – 2023**

Indice

Introduzione	vii
1 Background e stato dell'arte	1
1.1 Fitness-to-Drive	1
1.2 Studio dei guidatori	4
1.3 Digital twin	6
1.3.1 Definizione	7
1.3.2 Caratteristiche	7
1.3.3 Applicazioni	10
Human digital twin	11
1.3.4 Vantaggi e svantaggi	16
2 Il progetto	18
2.1 Obiettivi del progetto	18
2.2 Personalizzazione del Fitness-to-Drive	19
2.2.1 Requisiti strategici della proposta	21
Cd e Vd - distrazione cognitiva e visuale	24
E - emozioni e arousal	26
PersonalFactor - età e frequenza di guida	29
2.3 Human Digital Twin	34
2.4 Design del progetto	36
2.4.1 Human Digital Twin	38
2.4.2 Stereotipi	40
Classi di stereotipo	40
Moduli di stereotipo	42

2.4.3	Interazioni	43
3	Implementazione	48
3.1	Tecnologie utilizzate	48
3.2	Dettagli implementativi	51
	Gestione ad eventi	52
	Gestione degli stereotipi	52
	Adapters per la comunicazione: HTTP, MQTT e AMQP	55
3.3	Comunicazione tra Digital Twin: considerazioni	59
4	Valutazione sperimentale	62
4.1	Valutazione della precisione del FtD	62
4.1.1	Descrizione dei test effettuati	63
4.1.2	Test	63
	Valutazione prestazioni del FtD versione v0	64
	Valutazione prestazioni del FtD versione v1	65
4.1.3	Considerazioni	71
4.2	Valutazione della latenza	73
4.2.1	Descrizione del test	73
4.2.2	Risultati e considerazioni	75
4.2.3	Latenza di set-up	76
4.3	Utilizzo delle risorse	77
4.3.1	HDT	78
4.3.2	Microservizio FtD	80
	Conclusioni	82
	Ringraziamenti	85

Elenco delle figure

1.1	Componenti dei digital twin e processi ad alto livello [47]	9
1.2	Tecnologie fondamentali dei digital twin [39]	10
2.2	Funzione di Fitness-to-Drive secondo [31, 7].	22
2.3	Nuova funzione di Fitness-to-Drive.	23
2.4	Variazione del FtD in funzione della distrazione cognitiva e del rispettivo parametro w	26
2.5	Variazione del FtD in funzione delle emozioni al variare di w_1 , w_2 e w_3	28
2.6	Rappresentazione della funzione rappresentante il peso dell'età.	30
2.7	Variazione del FtD in funzione dell'età del guidatore	31
2.8	Variazione del FtD in funzione della frequenza di guida	33
2.9	Variazione del FtD in funzione dell'esperienza di guida	34
2.10	Modello di human digital twin proposto.	36
2.11	Architettura esagonale (fonte [26]).	37
2.12	Diagramma delle classi dello HDT.	40
2.13	Diagramma delle classi del modulo di stereotipo ftd.	43
2.14	Diagramma delle classi del microservizio ftdMS incaricato del calcolo del FtD.	44
2.15	Diagramma delle interazioni generico.	45
2.16	Diagramma delle interazioni per il caso d'uso FtD personalizzato.	46
3.1	Rappresentazione del pattern Strategy relativo agli adapter.	57
4.1	Velocità tenuta dal guidatore durante il test.	64
4.2	FtD calcolato con la formula precedente	65

4.3	Grafici della simulazione con un guidatore di 22 anni con esperienza, ma frequenza di guida bassa.	67
4.4	Grafici della simulazione con un guidatore di 22 anni con poca esperienza, ma frequenza di guida media.	68
4.5	Grafici della simulazione con un guidatore di 45 anni con esperienza, ma frequenza di guida bassa.	70
4.6	Grafici della simulazione con un guidatore di 45 anni con poca esperienza, ma frequenza di guida media.	71
4.7	Comparazione FtD senza notifica.	72
4.8	Comparazione FtD con notifica.	73
4.9	Utilizzo della CPU dello HDT a diverse frequenze di invio dei dati.	78
4.10	Utilizzo della RAM dello HDT a diverse frequenze di invio dei dati.	79
4.11	Utilizzo della CPU del microservizio a diverse frequenze di invio dei dati.	80
4.12	Utilizzo della RAM del microservizio a diverse frequenze di invio dei dati.	81

Elenco delle tabelle

4.1	Latenze registrate nei test con e senza personalizzazione.	76
4.2	Latenze registrate per l'avvio e configurazione dello HDT e del modulo ftd.	77

Introduzione

Oggigiorno l'industria automobilistica sta affrontando una delle sfide più importanti per il genere umano, ossia incrementare la sicurezza stradale, garantendo al tempo stesso una esperienza di guida confortevole. I comportamenti pericolosi alla guida possono contribuire in modo significativo agli incidenti stradali. Tuttavia, al momento manca un sistema standard di monitoraggio e allarme in tempo reale per la guida, che possa affrontare ed attenuare efficacemente questi pericoli stradali. Di conseguenza, i conducenti potrebbero non ricevere un feedback tempestivo o avvertimenti riguardo alle loro pratiche di guida rischiose, aumentando la probabilità di potenziali incidenti e mettendo a rischio la sicurezza stradale. Implementare sistemi avanzati e affidabili di monitoraggio della guida in tempo reale è essenziale per identificare tempestivamente comportamenti rischiosi e fornire ai conducenti allarmi immediati per promuovere abitudini di guida più sicure e ridurre la frequenza degli incidenti e dei sinistri. Una delle soluzioni più promettenti per sopperire a tale mancanza è lo sviluppo di digital twin, in particolare, visto il contesto in esame, una replica virtuale del conducente e dell'autoveicolo, che sia in grado di simulare, analizzare e prevedere il comportamento e le prestazioni del conducente. L'adozione di tecnologie come il digital twin può, infatti, offrire vantaggi significativi nel campo della sicurezza stradale. Attraverso la raccolta di dati provenienti da sensori installati sul veicolo e sul corpo del guidatore, è possibile monitorare parametri fisiologici [34], prestazioni di guida e interazioni con l'ambiente stradale in tempo reale [10]. Utilizzando poi algoritmi e modelli avanzati, il digital twin può analizzare questi dati per valutare le abitudini di guida, identificare comportamenti rischiosi e prevedere situazioni potenzialmente pericolose.

Nel contesto del progetto europeo NextPerception, è stato sviluppato un modulo di calcolo per valutare in modo oggettivo l'idoneità alla guida del conducente [31, 7]. Questo progetto si proponeva di sviluppare la prossima generazione di sensori di percezione e di migliorare il modello di intelligenza distribuita, creando soluzioni adattabili, sicure, affidabili e dinamiche per la rilevazione precoce e il monitoraggio umano. Tali soluzioni verranno applicate nel settore sanitario, del benessere e dei veicoli, compresi automobili, autoarticolati e autotreni. Durante la realizzazione di tale progetto sono stati effettuati test con simulatori di guida e utenti reali, raccogliendo preziosi dati sulle performance di guida in relazione a vari fattori fisiologici e ambientali.

L'obiettivo del seguente elaborato di tesi è quello di riuscire a sviluppare uno Human Digital Twin in modo tale da impiegarlo per calcolare un indice di idoneità alla guida che sia personalizzato sulla base delle informazioni conosciute relative al guidatore. In particolare, l'indice che si intende calcolare dovrà tenere conto sia di fattori oggettivi come età, genere e condizioni mediche, sia di come la persona reagisce effettivamente agli stimoli ed agli eventi durante la guida. La formula, necessaria per calcolare tale punteggio, secondo quanto emerso dagli studi effettuati, sarà quindi basata sia sui dati provenienti da sensori installati sul veicolo e sul corpo del guidatore, sia sulla base dei dati personali reperibili attraverso il digital twin. Il punteggio di idoneità verrà calcolato dinamicamente, aggiornandosi costantemente sulla base delle più recenti performance di guida e dello stato della persona.

Per fare ciò si prevede quindi di realizzare:

- una rappresentazione digitale del guidatore, al fine di monitorare lo stato e l'evoluzione della persona nel tempo;
- un modulo di calcolo del Fitness-to-Drive personalizzato, il cui scopo consiste nell'individuare un indice che rifletta l'idoneità del conducente a continuare la guida durante il movimento del veicolo, tenendo conto delle caratteristiche e delle risposte specifiche del guidatore.

In sintesi, questa tesi si propone di utilizzare il digital twin per personalizzare l'indice di idoneità alla guida, portando così avanti il lavoro iniziato con

il progetto NextPerception.

Fatta questa premessa, questo volume di tesi verrà strutturato come segue.

Nel **primo capitolo** viene illustrata una panoramica sullo stato dell'arte della ricerca relativa ai due principali temi affrontati nella tesi: i digital twin, per la rappresentazione digitale del guidatore, e l'idoneità alla guida dei soggetti sulla base di fattori oggettivi e non, per il calcolo personalizzato del Fitness-to-Drive. Si andrà quindi a illustrare cos'è il Fitness-to-Drive e quanto emerso dalla sua applicazione e verrà inoltre riportato quali sono i soggetti la cui idoneità di guida risulta più facilmente compromessa dal punto di vista statistico. Infine si andrà a definire cos'è un Digital Twin, il loro sviluppo e il loro potenziale per simulare e ottimizzare le prestazioni dei guidatori.

Nel **secondo capitolo** verrà fornita una panoramica generale degli obiettivi del progetto. Successivamente, verrà presentata la nuova proposta architettuale elaborata sulla base delle scoperte degli studi discussi nel capitolo 1 e dai dati emersi durante la fase di test con utenti reali nell'ambito del progetto NextPerception. In particolare, verrà descritta la nuova logica di calcolo del Fitness-to-Drive, evidenziando i fattori considerati e il diverso peso ad essi attribuito rispetto all'approccio tradizionale. Infine, verrà illustrato il design dello Human Digital Twin che si intende sviluppare, il quale rappresenterà la base per lo sviluppo del sistema, focalizzandosi sull'utilizzo di questo strumento per personalizzare il Fitness-to-Drive e adattarlo al contesto di guida.

Nel **terzo capitolo** verranno descritti nel dettaglio gli aspetti implementativi del progetto volti a concretizzare l'approccio architettuale definito nel capitolo precedente (capitolo 2). In particolare, verranno inizialmente illustrate le tecnologie utilizzate per lo sviluppo del sistema. A seguire verranno illustrati, mediante esempi pratici, gli aspetti di implementazione ritenuti più significativi, evidenziando al contempo le scelte effettuate per la realizzazione del sistema.

Nel **quarto capitolo** verranno esaminati i risultati ottenuti dall'applicazio-

ne del sistema realizzato ed effettuate alcune considerazioni su quanto emerso. In particolare, si analizzerà la precisione di tali dati e i tempi di elaborazione, mettendoli a confronto con i dati prodotti durante le simulazioni del progetto NextPerception, precursore del seguente elaborato di tesi. Verranno inoltre analizzati i tempi di set-up e l'utilizzo di risorse durante l'esecuzione. L'obiettivo è quindi quello di valutare l'efficacia delle soluzioni e delle strategie implementate nel contesto del progetto, nonché comprendere se queste rispondano in modo adeguato agli scopi prefissati.

Capitolo 1

Background e stato dell'arte

Lo scopo del seguente capitolo è di fornire una panoramica sullo stato dell'arte della ricerca relativa ai due principali temi affrontati nel seguente elaborato di tesi: i digital twin e l'idoneità alla guida dei soggetti, sulla base di fattori oggettivi e non.

Più in dettaglio, verranno discusse le più recenti ricerche su:

- Il Fitness-to-Drive di una persona mentre si trova alla guida.
- I fattori personali, clinici e situazionali che compromettono l'idoneità alla guida in gruppi specifici di popolazione, in particolare quelli che statisticamente risultano più a rischio di incidenti.
- I modelli digital twin, il loro sviluppo e le loro potenziali applicazioni alle persone e alla guida per migliorare le prestazioni.

1.1 Fitness-to-Drive

Il Fitness-to-Drive, o idoneità di guida, si riferisce alla capacità di guidare in sicurezza senza problemi causati da capacità fisiche, lesioni, salute medica o mentale. Generalmente l'analisi relativa alla capacità di guida viene determinata al momento del rilascio o rinnovo della licenza a seguito di valutazioni mediche e prove su strada o simulatori. Durante questa verifica usualmente

si procede ad addestrare e utilizzare un modello di regressione se l'obiettivo è individuare il valore di idoneità come una misura continua di prestazione. Al contrario, se si desidera semplicemente determinare se un individuo è idoneo o meno, viene utilizzato un modello di classificazione. Questi modelli sono fondamentali per prendere decisioni informate basate sulle caratteristiche e gli attributi degli individui. Oltre ai modelli appena citati, per la valutazione off-road del conducente sono stati effettuati diversi studi come ad esempio [38] e [9] i quali hanno trattato il tema di stabilire l'idoneità alla guida per valutare se rilasciare o meno la licenza di guida; tali studi però, oltre ad essere svolti off-road, o comunque considerando al massimo dei test di guida, valutavano l'idoneità solo per soggetti le cui capacità fisiche o mentali erano determinate a priori, come ad esempio per persone affette da lesioni cerebrali, Alzheimer, disturbi cognitivi, normodotate o con disabilità degli arti inferiori.

Un'analisi differente è stata condotta da Dronseyko [13], il quale ha proposto un coefficiente di pericolo di guida come metodo di valutazione del comportamento del generico conducente nel traffico stradale. Nel suo studio, Dronseyko, prende in considerazione fattori quali la velocità e la distanza laterale e longitudinale rispetto a dei valori di riferimento considerati sicuri, per valutare il comportamento del conducente. Un altro esempio di valutazione durante la guida è stata eseguita da ToledoT. et al.[44], il quale ha descritto il potenziale dei sistemi di registrazione dei dati a bordo dell'auto (IVDR)¹ nell'essere utilizzati in varie applicazioni commerciali e di ricerca, come strumenti per monitorare e fornire feedback ai conducenti sul loro comportamento su strada. L'utilizzo del sistema di DriveDiagnostics utilizzato ha permesso di identificare vari tipi di manovre che si verificano nelle misurazioni grezze; sulla base queste informazioni sono stati calcolati diversi indici di rischio, i quali indicano la sicurezza complessiva del viaggio. Il sistema di DriverDiagnostics impiegato si basa su: l'identificazione del guidatore attraverso un sistema di chiavi magnetiche, utilizzato poi per mantenere delle statistiche di rischio relative al

¹Per IVDR si intende "In-Vehicle Data Recorder", consiste in un dispositivo elettronico installato all'interno di un veicolo che registra e memorizza dati relativi alla guida e al comportamento del conducente.

conducente, e i dati del veicolo quali l'accelerazione laterale e longitudinale, la velocità e la posizione GPS. Questi ultimi dati sono, come detto, confrontati con dati standard al fine di calcolare ulteriori indici di rischio e successivamente mantenuti come statistiche di rischio relative al veicolo e al guidatore. I conducenti ricevono, sulla base dell'indice, un feedback attraverso vari rapporti di sintesi, messaggi di testo in tempo reale o un display all'interno del veicolo. I risultati dell'esperimento condotto, hanno mostrato che l'utilizzo di sistemi IVDR ha ridotto in modo ampio e statisticamente significativo il tasso di incidenti.

Qiang et. al. [23] ha proposto un sistema real-time non intrusivo, per il monitoraggio e la predizione della stanchezza del conducente durante la guida. Il sistema elaborato utilizza una telecamera per rilevare le caratteristiche facciali del conducente, come la chiusura degli occhi, la frequenza delle palpebre e l'inclinazione della testa. Questi dati vengono, secondo il modello proposto, elaborati da un algoritmo di apprendimento automatico che analizza i segnali elettrici del cervello e la frequenza cardiaca per determinare il livello di stanchezza del conducente. I risultati dello studio hanno dimostrato l'efficacia del sistema nel rilevare la stanchezza del conducente in tempo reale, con un'accuratezza del 90%. Altri studi, come quello di Yi-Cheng et. al.[45], hanno esaminato la relazione tra comportamento di guida e flusso del traffico o consumo di carburante; tuttavia, tali studi non hanno preso in considerazione le caratteristiche del tempo di guida, le richieste di guida degli utenti o come l'utente reagisce agli stimoli prodotti. In particolare, Yi-Cheng et. al.[45], ha proposto un sistema di assistenza alla guida che integra le informazioni dinamiche del veicolo, ottenute dal On-Board Diagnostics (OBD-II), e le condizioni del traffico in tempo reale, al fine di ridurre il rischio di incidenti stradali e migliorare la sicurezza del conducente attraverso l'invio di feedback all'utente ed eventualmente prendendo il controllo del veicolo. I test prodotti, sebbene effettuati solo in simulazione, hanno dimostrato come il sistema sia effettivamente in grado di adempiere al proprio scopo riducendo quindi il rischio di incidenti stradali.

Sulla stessa linea, ossia della valutazione on-road del conducente, nel volu-

me di tesi *Studio del Fitness-to-Drive: sensing delle attività di guida e monitoraggio del driver*[31], è stata introdotta una prima versione di una formula per individuare un indice che rappresenti l'idoneità di continuare o meno l'atto della guida da parte del conducente nel momento in cui il veicolo è in movimento. Tale indice era stato calcolato a partire dai dati provenienti dai sensori relativi alla distrazione visuale, cognitiva, la velocità del veicolo e delle emozioni provate dal conducente: come emozioni sono state prese in considerazione le emozioni di base più comuni nella quotidianità ossia le 6 emozioni primarie, definite tali perchè comuni e uguali per tutti in tutte le culture, individuate da Paul Ekman [14] (rabbia, felicità, sorpresa, disgusto, paura e tristezza) a cui è stato aggiunto lo stato emotivo neutrale, per un totale di 7 emozioni. Successivamente, a seguito del completamento da parte dell'università di Padova del modulo necessario per misurare l'arousal del guidatore, la formula è stata elaborata aggiungendo questo ulteriore parametro come fattore di attivazione delle emozioni. Infatti, l'arousal ha la funzione di indicare il grado di eccitazione di un individuo in relazione a ciò che si sta provando. Per approfondire l'argomento si consiglia di consultare il volume di tesi *Effetti dell'arousal sul Fitness-to-Drive: algoritmi e simulazione*[7] o il relativo articolo *Arousal effects on Fitness-to-Drive assessment: algorithms and experiments*[6] a cui la sottoscritta ha preso parte. Il suddetto calcolo del Fitness-to-Drive prevede: una penalizzazione maggiore in caso di distrazione cognitiva, seguita poi da quella visuale, entrambe poste dipendenti dalla velocità del veicolo, ed infine dalle emozioni, a cui sono state assegnati dei pesi decrescenti in base agli effetti che hanno sul compito di guida e successivamente stabilita la loro attivazione o meno sulla base del valore di arousal. Nel capitolo 2 verrà illustrata l'analisi dei dati ottenuti a seguito dei test con gli utenti.

1.2 Studio dei guidatori

Diverse condizioni cliniche e fattori demografici possono influenzare l'idoneità alla guida delle persone. Lo scopo di questa sezione è individuare i gruppi più vulnerabili e discutere le problematiche relative ad essi.

Sempre più frequentemente quando si parla di idoneità alla guida ci si concentra sugli aspetti puramente clinici delle persone dimenticando che come tali sono soggette a fattori psicologici, cognitivi ed emozionali, oltre che dalle proprie esperienze passate e abitudini consolidate. In particolare, in passato sono stati effettuati numerosi studi relativi ai disturbi cognitivi o alle impossibilità motorie di soggetti anziani; in questo caso, invece, si vuole porre l'attenzione sugli aspetti demografici e situazionali che possono alterare l'attitudine alla guida. È infatti opportuno ricordare che la guida non è solo questione di salute fisica, ma anche di benessere psicologico e cognitivo. Ad esempio, lo stress e l'ansia possono influire negativamente sulla capacità di guidare in modo sicuro.

Come già emerso dall'elaborato di tesi triennale [31] esistono diversi fattori che possono compromettere, in modo variabile, l'idoneità di guida. In particolare, come già sottolineato, un determinante chiave dell'idoneità alla guida è rappresentato dall'età del conducente: vi è una maggior fiducia negli adulti al volante rispetto ai giovani, infatti, il numero di violazioni che un giovane o neopatentato deve compiere perché una di esse si trasformi in un incidente è più basso rispetto a quello di un adulto esperto. È ancora più basso per gli anziani: anche una sola violazione può trasformarsi in un incidente, ma in compenso ne commettono pochissime. I giovani e gli anziani risultano quindi le fasce d'età più vulnerabili al volante [27, 28].

Secondo lo studio condotto da Ray Fuller[17], il comportamento del guidatore è influenzato da fattori sia interni che esterni, in particolare quelli interni riguardano l'età, il genere, l'esperienza di guida, lo stato mentale e fisico, la personalità e le abilità cognitive. I fattori esterni includono invece le condizioni atmosferiche, le condizioni del traffico, le caratteristiche dell'ambiente di guida, il design del veicolo, la presenza di altri utenti della strada e le leggi e le normative di guida. Sebbene non suddivisi in fattori esterni ed interni, anche Summala[21] identifica come elementi fondamentali per determinare l'idoneità di guida, l'età, l'esperienza di guida e la comfort-zone del conducente. Secondo lo studio, l'esperienza porta i guidatori ad acquisire competenze, strategie ed abitudini che migliorano la loro idoneità in termini di padronanza delle abilità, capacità decisionale, ansia ridotta e comfort, mentre per quanto riguarda la

comfort-zone, riferita al guidare in presenza di affetti o meno, afferma che i conducenti sono propensi ad essere più cauti in presenza di qualcuno a cui sono legati, rispetto a quando sono in presenza di estranei.

Studi come [46, 16, 3, 30] hanno esaminato come l'esperienza di guida e l'inattività influenzino le abilità dei guidatori nel mitigare i pericoli. In particolare, in[46], a seguito dei test condotti, è stato rilevato che i guidatori più esperti (3-5 anni dalla patente) ottenevano i punteggi più alti, indicando migliori abilità di mitigazione dei pericoli. I guidatori meno esperti (<1 anno) ottenevano invece i punteggi più bassi. Inoltre, periodi prolungati di inattività alla guida erano associati a peggiori prestazioni nei test, specialmente nei primi 2-3 anni dal conseguimento della patente. I risultati suggeriscono quindi che l'esperienza di guida migliora in modo significativo le abilità di mitigazione dei pericoli da parte dei conducenti, ma queste abilità possono deteriorarsi rapidamente con l'inattività. [30] ha approfondito come il genere del guidatore e la media dei chilometri annui incida sui sinistri commessi sottolineando come chilometraggio medio annuo risulta un indicatore livello di esperienza di guida. L'autore ritiene infatti che i conducenti più esperti siano più abili nell'attività di guida e abbiano maggiori probabilità di evitare incidenti, il che porta a un tasso di incidenti inferiore.

1.3 Digital twin

Il concetto di “*digital twin*”(DT) sta sempre più guadagnando popolarità negli ultimi anni, soprattutto nel campo dell'ingegneria e del design. Un “digital twin”, ad alto livello, non è altro che una replica virtuale di un oggetto o di un sistema fisico, creata attraverso l'uso di tecnologie avanzate come sensori, software di simulazione e analisi dati. Questo modello virtuale può essere utilizzato per monitorare e analizzare le prestazioni dell'oggetto o del sistema fisico, simulare diversi scenari e persino prevederne il comportamento futuro.

In questa sezione della tesi, esploreremo in profondità il concetto di “digital twin”, esaminandone le origini, le applicazioni e i potenziali benefici. In particolare, ci concentreremo sull'uso della tecnologia “digital twin” per forni-

re supporto decisionale, anticipare rischi e ottimizzare il benessere individuale delle persone.

1.3.1 Definizione

Il concetto di “digital twin” è stato concepito, sebbene con un nome diverso, per la prima volta da Michael Grieves e presentato nel 2003 all’università del Michigan[32]. Negli anni diverse sono le definizioni che sono state date di questo concetto, di seguito ne vengono riportate alcune, il cui fattore comune, come si può notare, è rappresentato dall’idea che un digital twin sia una replica digitale di un sistema fisico, utilizzata per analizzare, simulare e prevedere il comportamento del sistema in tempo reale.

- *“Macchine (fisiche e/o virtuali) o modelli basati su computer che simulano, emulano, riflettono o creano un ”gemellaggio” con la vita di un’entità fisica, che può essere un oggetto, un processo, un essere umano o una caratteristica correlata all’essere umano.”*[8].
- *“Una rappresentazione digitale di un prodotto unico attivo o di un sistema di prodotto-servizio unico, che comprende le sue caratteristiche, proprietà, condizioni e comportamenti selezionati attraverso modelli, informazioni e dati all’interno di una singola fase o anche attraverso più fasi del ciclo di vita.”*[42].
- *“Un digital twin è un’istanza virtuale di un sistema fisico (gemello) che viene continuamente aggiornata con i dati sulle prestazioni, la manutenzione e lo stato di salute di quest’ultimo durante l’intero ciclo di vita del sistema fisico.”*[29].

1.3.2 Caratteristiche

Al fine di poter realizzare un digital twin è necessaria l’esistenza di tre componenti[47, 33](Figura 1.1):

1. una **realtà fisica**, che consiste nel mondo reale in cui si trova l'oggetto fisico o il sistema che viene replicato e modellato;
2. una **rappresentazione virtuale**, è il digital twin vero e proprio, rappresenta una replica digitale che utilizza dati in tempo reale per simulare e prevedere il comportamento del sistema stesso;
3. **interconnessioni**, che permettono lo scambio di informazioni tra la realtà fisica e quella virtuale. Lo scambio può essere sia mono-direzionale, che bi-direzionale: nel primo caso generalmente le informazioni vengono inviate dalla realtà fisica a quella virtuale (*physical-to-virtual*), questo abilita il processo mediante il quale i dati raccolti dalla realtà fisica vengono utilizzati per aggiornare gli stati mantenuti nella rappresentazione virtuale. Nel secondo caso possiamo avere sia lo scambio *physical-to-virtual*, appena descritto, che uno scambio *virtual-to-physical*. Quest'ultimo consente di realizzare nel sistema fisico le intuizioni e le decisioni generate attraverso la rappresentazione virtuale, sia attraverso azioni che provocano un cambiamento negli stati del sistema fisico, sia attraverso azioni che raccolgono ulteriori informazioni dal sistema fisico per aggiornare ulteriormente la rappresentazione virtuale.

In particolare, è la comunicazione bi-direzionale che differenzia un DT dai simulatori tradizionali[2]. Un DT si connette al mondo fisico e segue rappresentazioni dettagliate e precise attraverso modelli personalizzati (ad esempio, implementando la logica di un dispositivo e i suoi parametri come il tempo, la posizione, la localizzazione, i processi, le funzioni, le forme geometriche, ecc..). Al contrario, un simulatore convenzionale non integra modelli specializzati che danno una rappresentazione dettagliata delle particolari caratteristiche del mondo fisico e stabiliscono interfacce bidirezionali tra gli spazi.

Secondo quanto riportato nel survey prodotto da Minerva et al.[33] un digital twin è caratterizzato da un aggiornamento continuo, in tal modo risulta essere sempre sincronizzato con la realtà fisica e la sua evoluzione. Il DT

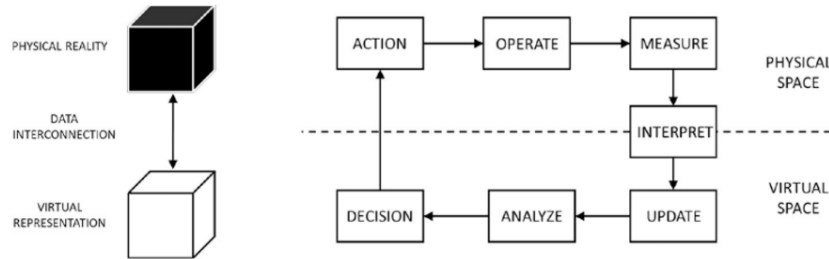


Figura 1.1: Componenti dei digital twin e processi ad alto livello [47]

fornisce, inoltre, un supporto decisionale basato sui dati. Infatti, grazie alla sua struttura e realizzazione è possibile effettuare nelle diverse fasi della realtà fisica che rappresenta una sua simulazione al fine di predire e studiare il comportamento del sistema in particolari situazioni o ambienti, permettendo così di prendere decisioni più informate e tempestive, per migliorare le prestazioni del sistema.

Diversi studi concordano inoltre, dal punto di vista implementativo, su alcune caratteristiche comuni: ogni digital twin e la rispettiva controparte reale devono essere univocamente identificati. L'identità del DT può essere vista come un'aggregazione più complessa di informazioni, infatti comprende il proprio identificativo, quello della controparte fisica, e le informazioni necessarie per localizzare ciascuna delle repliche ad esso associate, in modo da rappresentare la realtà in modo specifico e nel giusto contesto[33].

Ogni DT dovrebbe essere il più verosimile possibile alla realtà che rappresenta, senza però essere al contempo altrettanto complesso. Per garantire la rappresentatività della realtà che si intende analizzare, è spesso necessario semplificarla attraverso un processo di astrazione. Questo processo permette di mantenere le caratteristiche essenziali del sistema o del fenomeno analizzato, eliminando al contempo i dettagli superflui che potrebbero compromettere la validità e l'affidabilità dei risultati dell'analisi. Per fare ciò è richiesto l'utilizzo e supporto di tecnologie innovative, come i sensori, le simulazioni, il cloud computing e l'intelligenza artificiale [39] (figura 1.2).

La rappresentatività appena citata viene misurata rispetto tre principali

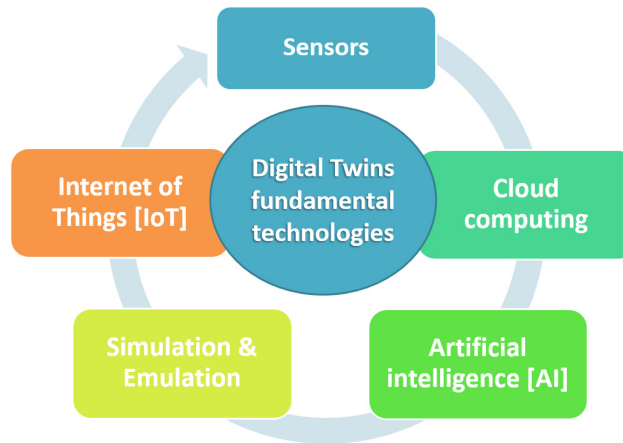


Figura 1.2: Tecnologie fondamentali dei digital twin [39]

parametri[33]:

- **Similarità**, indica quanto il DT riproduce bene la realtà, il suo stato e caratteristiche;
- **Casualità**, indica la probabilità che il DT differisca dalla realtà;
- **Contestualizzazione**, si riferisce al fatto che tutte le caratteristiche e i dati rilevanti disponibili sono necessari e sufficienti per rappresentare la realtà nello specifico spazio virtuale in esame.

In generale modellare un digital twin risulta un compito complesso che richiede diversi studi e accortezze. Tutti gli studi condotti a riguardo concordano in ogni caso che le nuove tecnologie aiuteranno notevolmente i designer in tale compito.

1.3.3 Applicazioni

La tecnologia dei digital twin presenta un'ampia gamma di applicazioni. È penetrata in molti domini tecnologici e si prevede che lo farà in molti altri campi con il progredire della tecnologia; pertanto, il concetto di DT è diventato sempre più importante per diversi settori aziendali, come ad esempio quello manifatturiero, per simulare e ottimizzare i processi di produzione e migliorare

la qualità dei prodotti; aerospaziale, per monitorare e ottimizzare le prestazioni dei velivoli, ridurre i costi di manutenzione e migliorare la sicurezza dei voli; dell'healthcare, per simulare e ottimizzare i processi di cura, migliorare la personalizzazione delle terapie e ridurre i rischi per i pazienti; smart city, per monitorare e gestire le infrastrutture urbane, migliorare la sicurezza e la qualità della vita dei cittadini; automotive, per monitorare e ottimizzare le prestazioni dei veicoli, migliorare la sicurezza stradale e sviluppare veicoli autonomi.

Human digital twin

Come detto l'applicazione delle tecnologie dei DT sono svariate, in particolare in questo elaborato di tesi si prende in considerazione il digital twin associato ad una persona (*Human digital twin*(HDT)) al fine di sfruttare le informazioni raccolte sul soggetto e poter calcolare un indice d'idoneità alla guida che sia personalizzato sulla base di esse. In particolare l'HDT consiste nella replicazione del singolo corpo umano, nella sua interezza o in una sua sottoparte, nello spazio virtuale, riflettendo contemporaneamente il suo stato fisico sia psicologicamente che fisiologicamente in tempo reale [35].

Secondo quanto riportato nell'articolo prodotto da Wei Shengli [41] per essere considerato tale, un human digital twin dovrebbe avere le seguenti 10 caratteristiche:

1. Ogni HDT deve avere un indice univoco, che permette di identificarlo e di accedervi in modo sicuro attraverso un account;
2. Ogni HDT nasce in contemporanea al soggetto a cui è associato. La sua configurazione dovrà essere effettuata in modo opportuno da un addetto specializzato. Una volta creato, lo HDT evolverà con il soggetto. È inoltre presente un concetto di evoluzione, pertanto i figli della persona avranno a loro volta un HDT associato il quale eredita alcune caratteristiche dai digital twin dei genitori, così come avviene nel mondo reale.

3. Ogni persona reale e il corrispondente Human Digital Twin cambieranno sincronicamente. Tutti i cambiamenti della persona stessa (chiamati fattori interni, ad esempio la crescita in altezza), le influenze su di essa (chiamate input esterni, come l'infortunio) e le sue risposte (chiamate output, come la stimolazione) verranno trasmessi allo spazio virtuale, e l'Human Digital Twin cambierà di conseguenza.
4. Tutti i dati raccolti verranno inviati, previo consenso del soggetto, allo spazio virtuale e l'HDT aggiornerà i dati di conseguenza.
5. Alcuni attributi della persona reale, come età e altezza, aumenteranno in modo naturale e graduale, e allo stesso tempo, secondo la legge della crescita umana, anche l'Human Digital Twin "crescerà", il che può essere corretto dalle informazioni provenienti dal suo corrispettivo reale, ottenute attraverso la sensoristica.
6. All'ambiente virtuale vengono inviati sia i dati relativi alla persona reale che quelli dell'ambiente in cui è immerso, creando così anche una rappresentazione virtuale dell'ambiente. In questo caso è importante mantenere la coerenza tra i dati dell'ambiente virtuale e i dati in funzione di questi di tutti gli HDT presenti in esso. A supporto di questo è possibile utilizzare tecnologie come la *Block Chain*.
7. Sulla base dei dati raccolti per la persona reale possono essere messe in campo tecnologie quali l'intelligenza artificiale, per effettuare analisi, simulazioni e previsioni, così da anticipare la possibile risposta della persona ai possibili cambiamenti a cui può andare incontro.
8. Ogni soggetto o le persone autorizzate possono accedere al sistema del proprio Human Digital Twin per visualizzare il contenuto.
9. Per mantenere la privacy, possono essere utilizzate tecnologie di sicurezza, pertanto ad ogni HDT è necessario associare un account protetto a cui potrà accedere solo la persona reale e il personale addetto, eventualmente quest'ultimo con limitazioni.

10. Garantire una sicurezza di accesso tenendo in considerazione le possibili vulnerabilità e cercare di porvi rimedio.

Un human digital twin può essere impiegato per monitorare lo stato di salute e il benessere di una persona, fornire un supporto personalizzato per la salute e il benessere e per monitorare e migliorare la sicurezza delle persone, ad esempio prevenendo situazioni di pericolo, come ad esempio un incidente alla guida, o in caso di emergenza, l'HDT può inviare automaticamente una chiamata di emergenza ai servizi di emergenza preposti [37].

In letteratura è possibile trovare diverse applicazioni dello HDT, in particolare, molte applicazioni sono relative all'ambito dell'healthcare e quello industriale, mentre in presenza minore troviamo quelli legati al settore dello sport e del trasporto. Relativamente a quest'ultimo, e connesso all'obiettivo del seguente elaborato di tesi, abbiamo due studi che prendono in considerazione il "guidatore", inteso in questo caso in senso lato, ossia considerando il generico conducente di un veicolo. Il primo [11] è relativo ad un progetto sponsorizzato dall'US Air Force, il quale mira a creare un "twin" di un pilota di aerei utilizzando una varietà di dati provenienti da sensori wearable e processati mediante intelligenza artificiale. Questi dati vengono applicati al modello muscolo-scheletrico del pilota, al modello antropometrico e al modello fisiologico per produrre risultati come: addestramenti personalizzati per migliorare le prestazioni del pilota, prevenzione degli infortuni, previsioni fisiologiche e ottimizzazione dell'ergonomia della cabina e dell'abbigliamento.

Il secondo studio [22] invece si concentra sul concetto di Driver Digital Twin (DDT), una particolare forma di human digital twin in cui viene creata una replica virtuale del conducente in grado di simulare il comportamento di guida umano e prevedere le intenzioni del conducente. Il DDT è creato combinando tra loro la modellazione basata sui dati, l'apprendimento automatico e la modellazione basata sulla fisica. Lo studio, in particolare, si sofferma su fatto che la digitalizzazione del conducente deve essere in grado di modellare le abitudini, i tratti di personalità e i modelli decisionali, che sono caratteristiche cruciali che definiscono un conducente. In questo modo, sfruttando il DDT e il

monitoraggio dello stato fisiologico sarà possibile prevenire potenziali incidenti causati da distrazione o improvvisa malattia, nonché migliorare la sicurezza alla guida. In tale ambito quindi il DDT dovrebbe includere un ampio spettro di informazioni, che vanno dalla comprensione a livello generale dello stile di guida, fino ai dettagli più precisi riguardanti il comportamento e l'attenzione del conducente. Infine gli autori discutono anche del potenziale del DDT per supportare lo sviluppo di veicoli autonomi, simulando il comportamento di guida umano e addestrando veicoli autonomi a rispondere in modo appropriato a un'ampia gamma di scenari di guida.

Sebbene non esplicitamente definito digital twin, anche il Virtual User (VU) fa parte dell'insieme degli Human Digital Twin, infatti nella definizione di un VU è prevista: la presenza del soggetto fisico, ossia la persona a cui viene associato il VU, la sua rappresentazione virtuale, il VU vero e proprio, ed infine la presenza di interconnessioni che permettono lo scambio di informazioni. Come si può notare corrispondono alle caratteristiche elencate precedentemente necessarie per la definizione di un DT. In particolare in [19] e [18], il VU viene definito come: “la fidata controparte digitale della persona che rappresenta, il cui scopo consiste nel facilitare l'integrazione del mondo tecnologico digitale riducendo l'onere di eseguire i processi di impostazione, configurazione e aggiornamento delle applicazioni in cui è coinvolta e sfruttando al meglio le potenzialità rilevanti, tenendo sempre conto della qualità di vita della persona”. Sempre secondo [19] e [18] il VU deve implementare alcune funzionalità, tra le più rilevanti troviamo: lo *User context* e lo *User profile*. Il primo risulta fondamentale per la gestione di tutte le applicazioni che considerano la posizione e più in generale il contesto in cui è presente una persona per poter operare, come ad esempio una smart house o una macchina; mentre il secondo fornisce informazioni importanti sulle caratteristiche del soggetto, permettendo così la personalizzazione delle applicazioni, tra cui quelle che utilizzano il contesto appena citato. Da questo ne deriva che la rappresentazione del VU, per come pensata dall'autore, presenta per l'utente un profilo, il quale descrive le principali caratteristiche della persona. Il profilo di una persona è stato altamente studiato in letteratura [43, 5, 25, 36, 15] e tutti concordano

sul fatto che sia un qualcosa di altamente dinamico, infatti basta pensare a come cambiano nel tempo gli hobby o interessi di una persona ad esempio durante l'adolescenza potrebbero riguardare uno sport, mentre in età adulta si concentrano più su eventi culturali. Poichè risulta di fondamentale importanza nella realizzazione di un HDT definire il profilo di una persona, alcuni studi, tra i quali [25, 36] si sono soffermati sull'individuare un modo standard per poterlo rappresentare, anche attraverso una ontologia; infatti sebbene in passato altri studi ([5, 43, 1]) hanno preso in considerazione ed elaborato il profilo di un utente, tale profilo non era standard e condivisibile, in quanto era solo application-specific o comunque creato appositamente per un particolare dominio. Secondo [25, 36, 15] un generico profilo di una persona dovrebbe includere almeno le seguenti categorie:

- **dati personali**, contenente le informazioni base dell'utente, come ad esempio i dati anagrafici;
- **interessi**, come ad esempio hobby (sport, cucina, automobilismo...) espressi sotto forma di keyword;
- **preferenze**, come ad esempio relativamente alle lingue, ai colori o un particolare modello di auto o moto;
- **professioni**;
- **educazione**;

Il profilo dovrà essere popolato sia in modo manuale dall'utente, in particolare al momento dell'inizializzazione, che in modo automatico a seguito di meccanismi studiati ad hoc, i quali sulla base delle attività e risposte dell'utente ne apprendono e modificano il profilo. Infatti, come detto precedentemente, il profilo della persona è altamente dinamico, per tale motivo [15] ha proposto di aggiornarlo mediante un meccanismo che integra tra loro il data mining e l'utilizzo di ontologie; più precisamente il processo di data mining viene utilizzato per andare ad associare gli attributi appresi ai concetti ontologici. I concetti vengono poi confrontati al fine di individuare degli elementi comuni

tra gli attributi in modo tale poi da aggiungere il segmento ontologico al profilo dell'utente. Come sottolineato dallo stesso autore, l'approccio impiegato non elimina completamente l'incertezza della previsione delle preferenze implicite dell'utente, ma permette di migliorare significativamente la precisione dell'agente in un numero considerevole di casi.

Come affermato in [20, 40], il profilo dell'utente, e quindi del VU/HDT, può essere sostituito o elaborato a partire dagli stereotipi. Uno stereotipo, infatti, contiene le caratteristiche tipiche di un gruppo di utenti in un particolare dominio applicativo insieme ad un insieme di condizioni di attivazione, che rendono possibile identificare gli utenti appartenenti a questo gruppo; un esempio potrebbe essere la classificazione di una persona come atletica perché se osservata si può notare un individuo energico, propenso all'allegria, fortemente interessato ad attività all'aria aperta e, in modo ossessivo, a regimi dietetici e di allenamento. L'utilizzo di questo approccio per la definizione del profilo vede però una pre-definizione delle classi di stereotipo ed il rischio di una eccessiva generalizzazione della persona, infatti se non opportunamente configurati potrebbero rientrare in delle classi di stereotipo persone che sono completamente l'opposto, ad esempio non è possibile definire la classe "anziano" solo sulla base di caratteristiche della personalità o le attività svolte, ma sarà necessario tenere presente anche l'età del soggetto per avere una classificazione più precisa, anche se comunque questa non risulterà mai vera per tutti i soggetti; l'obiettivo e la difficoltà maggiore è quindi quella di creare delle classi che riducano al minimo gli errori di classificazione. Pertanto queste inferenze devono essere trattate come impostazioni predefinite, che possono essere ignorate da osservazioni specifiche.

1.3.4 Vantaggi e svantaggi

I digital twin presentano vantaggi significativi in termini di efficienza, produttività, qualità, sicurezza e supporto decisionale, ma anche numerosi svantaggi come costi elevati, complessità, rischi per la sicurezza e la privacy e limitazioni tecnologiche. La scelta di utilizzare o meno i digital twin dipende quindi

dalla specifica situazione e dalle esigenze dell'organizzazione o dell'individuo coinvolto.

Per quanto riguarda l'ambito di applicazione dei DT nel seguente elaborato, i principali vantaggi riguardano la possibilità di riuscire ad adattare l'indice generico rappresentante il Fitness-to-Drive sulla base delle caratteristiche dell'utente, ma d'altra parte, come accennato, bisogna tenere in considerazione il fatto che la creazione di un DT di una persona solleva questioni etiche e di privacy, e in molti paesi potrebbe essere illegale senza il consenso esplicito della persona coinvolta.

Capitolo 2

Il progetto

Il seguente capitolo fornirà una panoramica generale degli obiettivi del progetto. Successivamente, sarà presentata la nuova formula del Fitness-to-Drive, basata sulle scoperte degli studi precedenti e sui test condotti con gli utenti, con particolare attenzione all'analisi degli elementi costituenti. Infine, sarà illustrato il design dello Human Digital Twin che si intende sviluppare, focalizzandosi sull'utilizzo di questo strumento per personalizzare il Fitness-to-Drive e adattarlo al contesto di guida.

2.1 Obiettivi del progetto

Il presente volume di tesi si concentra su un progetto articolato in due obiettivi: la personalizzazione del Fitness-To-Drive e la sua integrazione nello stato dello Human Digital Twin.

Il primo obiettivo riguarda il calcolo del Fitness-to-Drive personalizzato, mirando a individuare un indice che rifletta l'idoneità del conducente a continuare la guida durante il movimento del veicolo, tenendo conto delle caratteristiche e delle risposte specifiche del guidatore. Tale valutazione viene effettuata tramite il monitoraggio del conducente utilizzando apposite tecnologie sensoriali poste all'interno dell'abitacolo, i feedback forniti dall'utente stesso e le sue peculiarità.

Il secondo obiettivo si propone di introdurre e implementare un nuovo concetto di Human Digital Twin che consideri il profilo dell'utente rappresentato, consentendo di sfruttarlo attraverso stereotipi classificati per personalizzare le applicazioni e i dispositivi con cui l'utente interagisce.

Entrambi gli obiettivi sono fondati sulle evidenze scientifiche presenti in letteratura, le quali sono state dettagliate nel capitolo precedente.

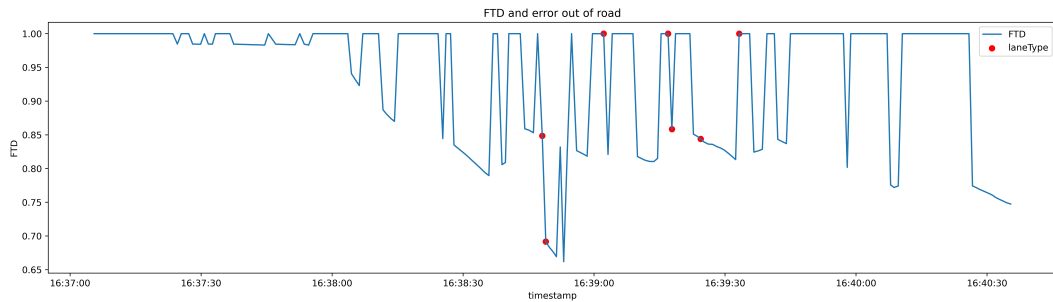
2.2 Personalizzazione del Fitness-to-Drive

Come discusso nel capitolo precedente, nel volume di tesi [31] era stata presentata una prima proposta di Fitness-to-Drive (FtD), che successivamente è stata integrata con il valore di arousal in [7]. La formula così elaborata è stata poi impiegata all'interno del progetto europeo NextPerception, il cui obiettivo è di innalzare il livello di qualità e della tecnica di utilizzo di tecnologie di rilevamento della percezione, migliorando le loro caratteristiche, per consentire una valutazione più accurata del comportamento umano e dei parametri fisiologici. Al fine di valutare il raggiungimento degli obiettivi prefissati ed in particolar modo il funzionamento complessivo del sistema prodotto, sono stati condotti dei test su un campione di utenti composti da uomini e donne di diverse età ed in possesso di una licenza di guida. I test sono stati condotti grazie all'utilizzo di un simulatore messo a disposizione dall'azienda Re:lab e sul quale sono stati installati i moduli relativi alla distrazione visuale, cognitiva, al rilevamento delle emozioni e dei valori di arousal ed infine il modulo incaricato di calcolare il Fitness-to-Drive del conducente. Durante una simulazione di guida, ai partecipanti è stato chiesto di svolgere cinque diverse attività, ciascuna associata a una relativa fase di test, ognuna della durata di 3 minuti:

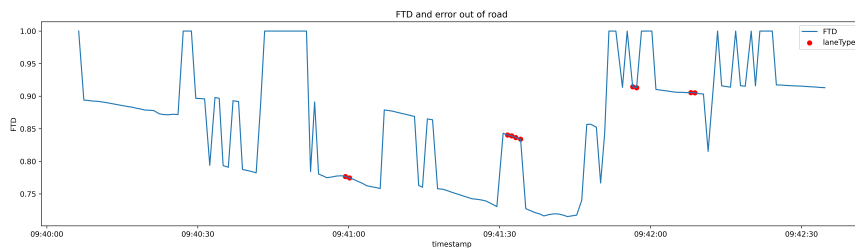
1. *Guida libera*, il cui scopo è creare una linea di base rispetto alla quale confrontare i dati raccolti nelle altre fasi.
2. *Car following*, durante la quale il partecipante era incaricato di seguire un veicolo posto davanti a sé; ciò consente di misurare lo stato del partecipante rispetto ad una semplice attività.

3. *Car following con musica*, si richiede al conducente di seguire il veicolo che lo precede mentre è in corso un sottofondo musicale, il cui obiettivo è indurre un particolare tipo di emozione, in particolare positive o negative.
4. *Car Following e attività di memoria*, si richiede al conducente di seguire il veicolo che lo precede mentre gli vengono poste delle domande, il cui obiettivo è indurre una distrazione cognitiva nel conducente.
5. *Car Following e distrazione visuale*, viene richiesto al conducente di seguire il veicolo che lo precede e al contempo svolgere delle attività che lo inducono a distogliere lo sguardo dalla strada, quindi una distrazione visuale.

Come si può osservare nelle figure 2.1a, 2.1b i valori di Fitness-to-Drive in concomitanza di un errore di guida risultano o essere elevati per poi decadere bruscamente in seguito all'evento di errore, oppure già bassi in concomitanza di questo; dove come evento di errore è stato considerato l'uscita, anche parziale, dell'auto dalla carreggiata.



(a) Dati ottenuti su un test con emozioni negative



(b) Dati ottenuti su un test con emozioni positive

Inoltre, osservando i valori rilevati è possibile notare che:

- per la distrazione cognitiva e visuale è possibile confermare quanto affermato da precedenti studi([4, 24]), ossia che risultano tra loro in parte distinte e in parte congiunte;
- la distrazione cognitiva è correlata positivamente alla velocità e all'emozione di rabbia, mentre quella visuale è più correlata alla felicità;
- l'evento di errore è correlato negativamente alla velocità, alle emozioni positive e alle due tipologie di distrazione, ciò significa che l'utente accortosi dell'errore tende a rallentare e prestare nuovamente attenzione al task di guida, perciò in concomitanza dell'errore stradale il Fitness-to-Drive tende ad essere più elevato. Mentre risulta correlato positivamente alle emozioni negative, quali rabbia, paura e noia, e all'arousal.

Osservando i risultati è possibile anche notare che il dato prodotto dal Fitness-to-Drive risulta spesso troppo generico, pertanto si ritiene opportuno implementare dei meccanismi utili a calcolare l'idoneità di guida sulla base di ogni specifico guidatore, considerando quindi le sue caratteristiche e le reazioni alle distrazioni ed emozioni a cui è sottoposto.

2.2.1 Requisiti strategici della proposta

A seguito di quanto scaturito dai test appena descritti, per colmare le lacune presenti nel progetto NextPerception, si ritiene necessario, al momento del calcolo del Fitness-to-Drive, tenere conto di un fattore di personalizzazione. Pertanto è stato deciso di:

1. ri-definire la formula del Fitness-to-Drive;
2. variare i pesi assegnati ai singoli parametri della formula entro un range prestabilito;
3. tenere conto di alcune peculiarità del guidatore e di come queste influenzano la guida, in particolare: l'età e la frequenza ed esperienza di guida.

Di seguito viene presentata la nuova formula complessiva di Fitness-to-Drive (FtD) proposta, mentre nei paragrafi successivi verranno analizzati più in dettaglio i singoli componenti che la costituiscono.

In [31, 7] il FtD veniva rappresentato come una funzione lineare (figura 2.2) che dipendeva unicamente dal valore calcolato per la distrazione cognitiva, visuale, le emozioni e il livello di arousal.

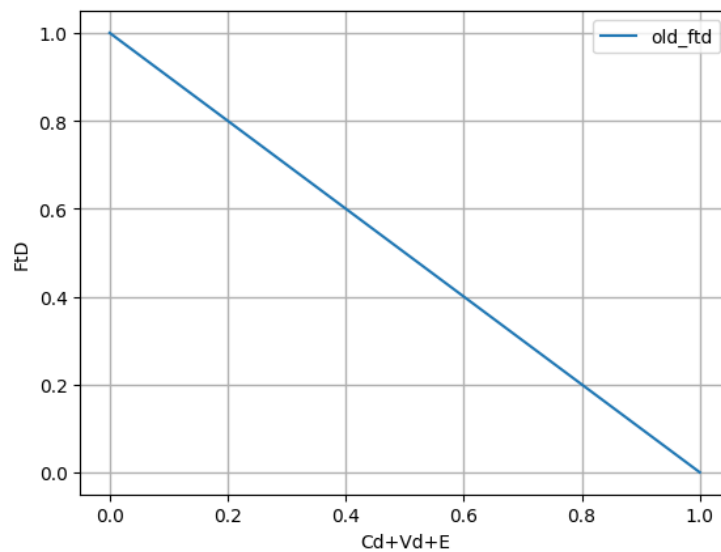


Figura 2.2: Funzione di Fitness-to-Drive secondo [31, 7].

La nuova formula prevede l'utilizzo di una funzione esponenziale con valori in $]0, 1]$:

$$FtD(x) = e^{-\frac{x}{1.2}} \quad (2.1)$$

Dove:

- x , rappresenta il valore ottenuto dalla somma di $Cd+Vd+E+personalFactor$, ossia del valore della distrazione cognitiva, visuale, emozioni e arousal e le caratteristiche del conducente. Tali valori, come detto verranno descritti nei prossimi paragrafi.
- 1.2, rappresenta un fattore di scala per l'input x , in quanto regola la "velocità" con cui il valore dell'esponenziale diminuisce o aumenta al

variare di x . Lo scopo risulta quindi quello di calibrare il comportamento della funzione e garantire che i risultati siano coerenti con i valori attesi. Il valore è stato scelto a seguito dell'analisi dei risultati dei test presentati in precedenza, in particolare dalla relazione tra valore di FtD e le uscite di strada.

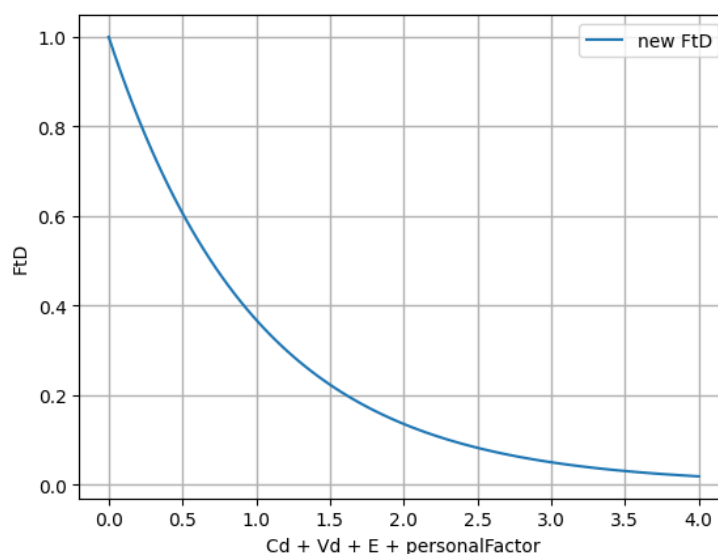


Figura 2.3: Nuova funzione di Fitness-to-Drive.

La formula appena illustrata, come si può osservare dalla figura 2.3, descrive un decadimento esponenziale, poiché il suo valore diminuisce rapidamente man mano che x aumenta in valore assoluto. Ciò rende la formula adatta per modellare fenomeni che diminuiscono in intensità o probabilità nel tempo o nello spazio, come nel caso dell'idoneità di guida in presenza di distrazioni prolungate e/o emozioni che la influenzano negativamente. La scelta di utilizzare questa funzione è motivata dalla necessità di penalizzare maggiormente i primi istanti in cui il guidatore inizia a manifestare un comportamento non idoneo all'attività di guida, in modo da notificarlo prontamente e prevenire tempestivamente eventuali situazioni di pericolo. Prima di procedere con l'analisi dei singoli elementi che influiscono sul calcolo del FtD, si vuole precisare che la formula presenta delle semplificazioni legate ai valori costanti impie-

gati, i quali a seguito di test più approfonditi potranno essere ulteriormente calibrati.

Cd e Vd - distrazione cognitiva e visuale

Il peso della distrazione (XD considerando X come “distrazione cognitiva” o “distrazione visuale”) sul FtD è rappresentato dalla formula:

$$XD_{i+1} = D_{i+1} * \frac{s_{i+1}}{th_s} * w_{i+1}^{Id-th_{di}} \quad (2.2)$$

Dove:

- D_{i+1} , indica se è presente (1) o meno (0) una distrazione all’istante $i + 1$;
- s_{i+1} , è la velocità del veicolo;
- th_s , è un parametro utilizzato per normalizzare la velocità;
- w_{i+1} , valuta la distrazione in base al numero di istanti in cui essa è presente. Di seguito viene illustrato come varia all’interno del range di valori possibili, infatti $w_{i+1} \in [0.95, 1.10]$, tale range di valori è stato definito a seguito di un attento studio dei test svolti in occasione del progetto NextPerception:

$$w_{i+1} = \begin{cases} w_i, & \text{se } D_i - 0.03 < D_{i+1} < D_i + 0.03. \\ w_i - 0.01, & \text{se } D_{i+1} < D_i. \\ w_i + 0.01, & \text{se } D_{i+1} > D_i. \end{cases} \quad (2.3)$$

- Id sono il numero di istanti consecutivi in cui è presente una distrazione, cognitiva o visuale;
- th_{di} è il numero di istanti in cui è ammissibile che l’utente sia distratto. Tale soglia sarà pari ad 1 per la distrazione cognitiva, mentre per la distrazione visuale sarà pari a 2.

Come si può notare la formula è rimasta pressoché invariata rispetto a quella proposta in [31, 7], l'unica differenza è rappresentata dal fattore w . Questa scelta è stata motivata dall'introduzione di un feedback dell'utente rispetto a quanto notificato all'istante precedente. L'obiettivo è di adeguare il peso della distrazione, sia cognitiva che visuale, in base alle azioni dell'utente in un determinato istante. Se l'utente era distratto e continua a esserlo, il fattore w viene aumentato, aumentando così l'impatto della distrazione sul FtD come se fosse un reward negativo, ossia una punizione. Al contrario, se il valore della distrazione è inferiore rispetto a quello dell'istante precedente, il fattore w viene diminuito, premiando il guidatore per la maggiore concentrazione. Questo approccio ci permette di calibrare la formula della distrazione in base alle caratteristiche specifiche del guidatore.

In figura 2.4 viene mostrato lo studio di variabilità del FtD in funzione di una delle due distrazioni e in particolare del valore di w ; pertanto data la formula del FtD proposta in precedenza viene fatto variare il valore della distrazione, ad esempio cognitiva, tenendo a valore costante nullo il risultato derivante dalla distrazione visuale, delle emozioni e dei fattori personali. Inoltre vengono considerati costanti i valori della distrazione, ponendola come sempre presente $D_{i+1} = 1$, la velocità del veicolo, $s_{i+1} = 50$, il numero di istanti consecutivi $Id = 2$ ossia uno in più rispetto alla soglia th_{di} considerata.

Come si può notare in figura 2.4, il valore della distrazione dipendente da w influisce notevolmente sul valore complessivo del FtD. In particolare, possiamo notare che con $w = 0.95$ il valore è 0.8204, mentre con $w = 1.10$ il valore è 0.7952. Questa variazione dei valori in base a w indica che tale parametro ha un impatto significativo sulla funzione del FtD, e modifica il rapporto tra il valore di partenza e il valore finale. Sempre osservando il grafico è possibile notare come il peso fa variare il valore del calcolo della distrazione, infatti da quando è più basso a quando è più elevato, l'impatto della distrazione aumenta solo di 0.035.

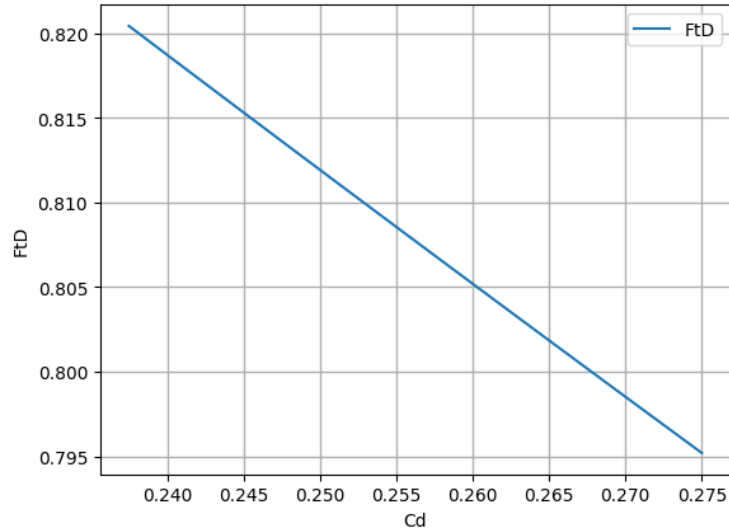


Figura 2.4: Variazione del FtD in funzione della distrazione cognitiva e del rispettivo parametro w

E - emozioni e arousal

Il peso delle emozioni, tenendo conto anche dell'arousal (E) sul FtD è rappresentato dalla formula:

$$E_{i+1} = ar_{i+1} * \frac{(a + h) * w1_{i+1} + (s_a + f) * w2_{i+1} + (d + s_u) * w3_{i+1} + n * 0}{w1_{i+1} + w2_{i+1} + w3_{i+1}} \quad (2.4)$$

Dove:

- ar_{i+1} , il valore di arousal rilevato nell'istante $i + 1$;
- a, h, s_a, f, d, s_u, n , rappresentano il valore rilevato all'istante $i + 1$ per le sei emozioni di Ekman e quella neutrale. Si riferiscono rispettivamente a: rabbia, felicità, tristezza, paura, disgusto, sorpresa e neutrale.
- $w1_{i+1}, w2_{i+1}, w3_{i+1}$, rappresentano i pesi assegnati alle emozioni all'istante $i + 1$. I pesi sono decrescenti con $w1_{i+1}$ il più elevato, $w2_{i+1}$ intermedio e $w3_{i+1}$ il più basso tra tutti, e sono tali che $w1_{i+1} + w2_{i+1} + w3_{i+1} \in [0.25, 0.75]$, tale range di valori è stato definito a seguito di un attento

studio dei test svolti in occasione del progetto NextPerception. Di seguito viene definito come varia wx , dove la x rappresenta uno dei valori 1, 2, 3 usati per identificare i pesi:

$$wx_{i+1} = \begin{cases} wx_i, & \text{se } E_i - 0.01 < E_{i+1} < E_i + 0.01. \\ \max(0, wx_i - 0.005), & \text{se } E_{i+1} < E_i. \\ wx_i + 0.005, & \text{se } E_{i+1} > E_i. \end{cases} \quad (2.5)$$

Come si può notare, come la precedente, la formula è rimasta pressoché invariata rispetto a quella proposta in [31, 7], l'unica differenza è rappresentata dai valori di $w1$, $w2$ e $w3$, i quali non risultano più costanti, ma variano in un range di valori. Anche in questo caso la scelta è stata motivata dall'introduzione di un feedback dell'utente rispetto a quanto notificato all'istante precedente. L'obiettivo è quindi quello di adeguare il peso delle emozioni in base alle azioni dell'utente in un determinato istante. Se l'utente provava emozioni negative per la guida con un livello di attivazione elevato e continua a provarle, i pesi wx vengono aumentati, aumentando così l'impatto delle emozioni e dell'arousal sul FtD, come se fosse un reward negativo cioè una punizione. Al contrario, se il valore delle emozioni è inferiore rispetto a quello dell'istante precedente, i pesi wx vengono diminuiti, premiando il guidatore per la presenza di uno stato emotivo più idoneo alla guida. Questo approccio ci permette di calibrare la formula delle emozioni in base alle caratteristiche specifiche del guidatore.

In figura 2.5 viene mostrato lo studio di variabilità del FtD in funzione delle emozioni, in particolare dei valori assunti da $w1$, $w2$ e $w3$; pertanto data la formula del FtD proposta in precedenza viene fatto variare il valore delle emozioni, tenendo a valore costante nullo il risultato derivante dalle distrazioni e dei fattori personali. Inoltre, vengono considerati costanti i valori dell'arousal, ponendolo come sempre presente e con il livello di attivazione più elevato $ar_{i+1} = 1$ e il valore rilevato per le singole emozioni, in particolare: $a = 0.144814$, $h = 0.050278925$, $f = 0.017982975$, $s_a = 0.0024303525$, $n = 0.747774$, $d = 0.0026372775$, $s_u = 0.034082575000000004$. Tali valori sono

stati estrapolati da una simulazione di test eseguita nell'ambito del progetto NextPerception.

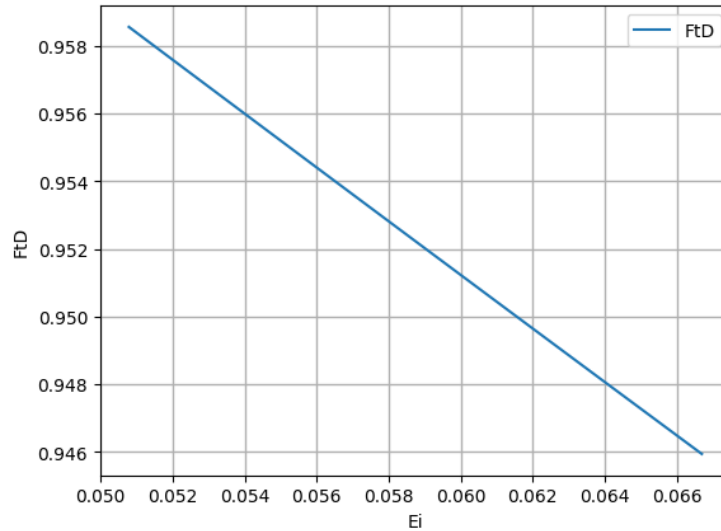


Figura 2.5: Variazione del FtD in funzione delle emozioni al variare di w_1 , w_2 e w_3 .

Come si può notare in figura 2.5, il valore delle emozioni dipendente da w_1 , w_2 e w_3 influisce solo in parte sul valore complessivo del FtD. In particolare, possiamo notare che con somma $w = 0.25$, quindi quando abbiamo il valore più basso per il peso delle emozioni, il valore del FtD è 0.9585, mentre con $w = 0.75$ il valore del FtD è pari 0.9459. Questa variazione dei valori in base a w indica che tale parametro ha un impatto non troppo elevato sulla funzione del FtD, e modifica il rapporto tra il valore di partenza e il valore finale. Sempre osservando il grafico è possibile notare come i pesi fanno variare il valore del calcolo delle emozioni, infatti da quando abbiamo la somma più bassa dei pesi a quando è più elevata, l'impatto delle emozioni aumenta solo di 0.016. I dati, inoltre, mostrano una bassa varianza e deviazione standard, il che suggerisce che i valori nel campione sono abbastanza vicini alla media e poco dispersi. Il range relativamente piccolo indica che la variazione tra il valore massimo e il valore minimo è contenuta. Questi risultati indicano che il campione di dati è omogeneo e poco variabile.

PersonalFactor - età e frequenza di guida

In base agli studi condotti e presentati nel capitolo precedente, si è scelto di considerare l'età del guidatore e la sua frequenza ed esperienza di guida come fattori personali rilevanti.

L'età influisce sulla guida in particolar modo nelle fasce relative ai giovani e agli anziani, mentre risulta ottimale per le età intermedie. Pertanto, abbiamo deciso di rappresentare l'età utilizzando una funzione gaussiana, e il peso relativo a questo fattore viene calcolato nel seguente modo:

$$A(age) = [(1 - 0.95)e^{-\frac{(median - age)^2}{(2std)^2}} + 0.95] - [(1 - 0.95)e^{-\frac{(age - median)^2}{(2std)^2}} + 0.95] \quad (2.6)$$

Dove:

- $(y_{max} - y_{min}) * y_{norm} + y_{min}$, con $y_{max} = 1$, $y_{min} = 0.95$ e $y_{norm} = e^{-\frac{(x - median)^2}{(2std)^2}}$, viene impiegato per riportare i valori della funzione per il calcolo dell'impatto dell'età nel range $[0.95, 1]$, definito sulla base degli studi condotti;
- *median*, rappresenta la mediana della gaussiana, e viene posta pari a 49, essendo questa l'età media tra quando è possibile ottenere una licenza di guida e l'età oltre al quale il rinnovo deve avvenire in modo più frequente in quanto relativo ad età a rischio[12];
- *std*, rappresenta la deviazione standard ed è posta pari a 10, il che significa che la maggior parte delle osservazioni si trovano entro 20 anni dall'età media (10 anni a sinistra e 10 anni a destra);
- *age*, rappresenta l'età del guidatore.

Il peso dell'età sul FtD è calcolato come la differenza tra il valore ottenuto nell'età ideale, rappresentata dalla mediana, e quello ottenuto nell'età del guidatore. Questa differenza è dovuta al fatto che l'età influenza la guida in modo simile a una gaussiana rovesciata (come mostrato nella Figura 2.6). Come accennato nel capitolo precedente, infatti, i giovani e gli anziani sono le

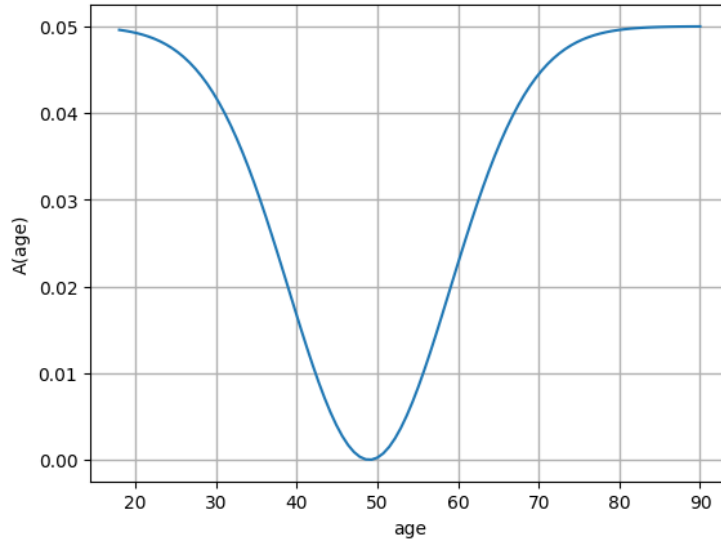


Figura 2.6: Rappresentazione della funzione rappresentante il peso dell'età.

fasce d'età più vulnerabili al volante [27], pertanto in corrispondenza di essi il valore del FtD dovrà risultare più basso, mentre in corrispondenza delle età ottimali dovrà essere più elevato.

In figura 2.7 viene mostrato lo studio di variabilità del FtD in funzione dell'età; pertanto data la formula del FtD proposta in precedenza viene fatto variare il valore dell'età nel range di riferimento ossia tra i 18 e i 90 anni, tenendo a valore costante nullo il risultato derivante dalle distrazioni e dalle emozioni. Inoltre vengono considerati costanti il valore della mediana e della deviazione standard, posti rispettivamente uguali a 49 e 10.

Come si può notare in figura 2.7, il valore dell'età influisce solo in parte sul valore complessivo del FtD. In particolare, possiamo notare come man mano che ci si allontana dall'età media di guida (49 anni) il peso aumenti, mentre in corrispondenza di essa risulta essere nullo. Sempre osservando il grafico è possibile notare come l'età fa variare il valore del calcolo del peso dell'età sul FtD, infatti da quando abbiamo l'età ideale a quando ci troviamo nelle fasce più a rischio, il peso aumenta solo di 0.04. I dati, inoltre, indicano una varianza e deviazione standard relativamente basse, suggerendo che i dati sono concentrati intorno alla media. Il range è relativamente piccolo, indicando una

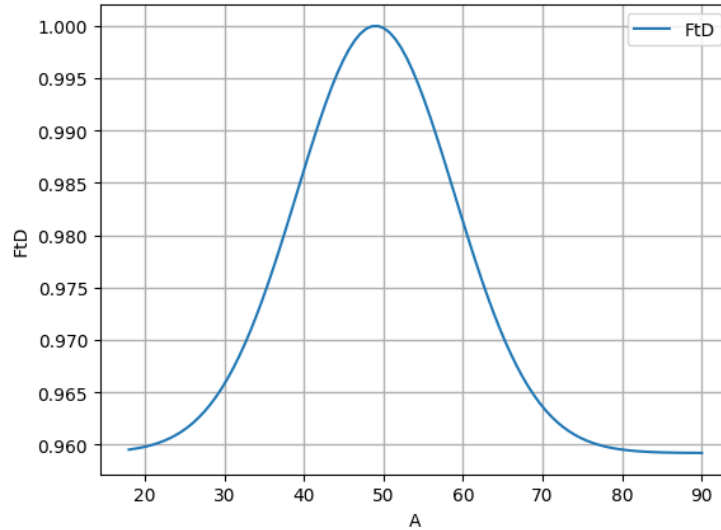


Figura 2.7: Variazione del FtD in funzione dell'età del guidatore

limitata variazione dei dati nel campione. Complessivamente, i dati mostrano una certa omogeneità e stabilità rispetto alla media.

Per quanto riguarda la frequenza ed esperienza di guida si è deciso di considerare, per il calcolo del suo peso sul FtD, i chilometri percorsi all'anno e gli anni di esperienza. In particolare per il chilometraggio sono state individuate delle soglie secondo le quali si può affermare che un guidatore ha una bassa ($km < 2000$), media ($2000 \leq km \leq 5000$) o alta ($km \geq 5000$) frequenza di guida, tenendo conto che secondo l'EEA¹ mediamente un utente percorre 11.000 - 13.000 km all'anno; mentre per gli anni di esperienza si è considerato quanto appreso dagli studi [46, 16], quindi è stata posta una soglia in corrispondenza dei 2 anni dal conseguimento della licenza di guida; al di sotto di tale soglia infatti si ritiene il guidatore ancora inesperto. A seguito delle considerazioni fatte, il peso relativo a questo fattore viene calcolato come segue:

$$F(km, ly) = exp(ly) * freq(km) \quad (2.7)$$

Dove:

¹European Environment Agency (EEA): Transport and environment report 2022 <https://www.eea.europa.eu/publications/transport-and-environment-report-2022/transport-and-environment-report/view>

- $exp(ly)$, rappresenta il valore calcolato per l'esperienza sulla base degli anni dal conseguimento della licenza di guida e la soglia definita in precedenza:

$$exp(ly) = \begin{cases} 2, & \text{se } ly < 3. \\ 1, & \text{altrimenti.} \end{cases} \quad (2.8)$$

- $freq(km)$, rappresenta il valore calcolato per la frequenza di guida sulla base dei chilometri percorsi all'anno e le soglie definite in precedenza:

$$freq(km) = \begin{cases} 0.1, & \text{se } km < 2000. \\ 0.05, & \text{se } 2000 \leq km < 5000. \\ 0, & \text{se } km \geq 5000 \end{cases} \quad (2.9)$$

Come si può notare il valore di $F(km, ly) \in [0, 0.2]$, tale range di valori è definito sulla base degli studi condotti. Tale valore deriva dal prodotto tra l'esperienza del guidatore e il peso assegnato alla frequenza di guida.

In figura 2.8 viene mostrato lo studio di variabilità del FtD in funzione della frequenza di guida, ossia dei chilometri percorsi annualmente; pertanto data la formula del FtD proposta in precedenza viene fatto variare il valore dei chilometri percorsi annualmente, tenendo a valore costante nullo il risultato derivante dalle distrazioni e dalle emozioni e a valore neutro, ossia pari ad 1, il peso derivante dall'esperienza, ipotizzando un numero di anni dal conseguimento della patente pari a 4.

Nell'analisi dei dati presentati in figura 2.8, si osserva che il valore della frequenza di guida ha solo una modesta influenza sul valore complessivo del FtD. Questo andamento a gradini è dovuto ai pesi prestabiliti associati alla frequenza di guida. Globalmente, si può notare che la frequenza di guida incide al massimo con un fattore di 0.08 sul FtD. I dati, inoltre, evidenziano una varianza e una deviazione standard relativamente basse, indicando una concentrazione dei dati intorno alla media. Il range dei dati è relativamente piccolo, suggerendo una limitata variazione dei valori all'interno del campione. Complessivamente, l'analisi dei dati indica un certo grado di omogeneità e stabilità rispetto alla media.

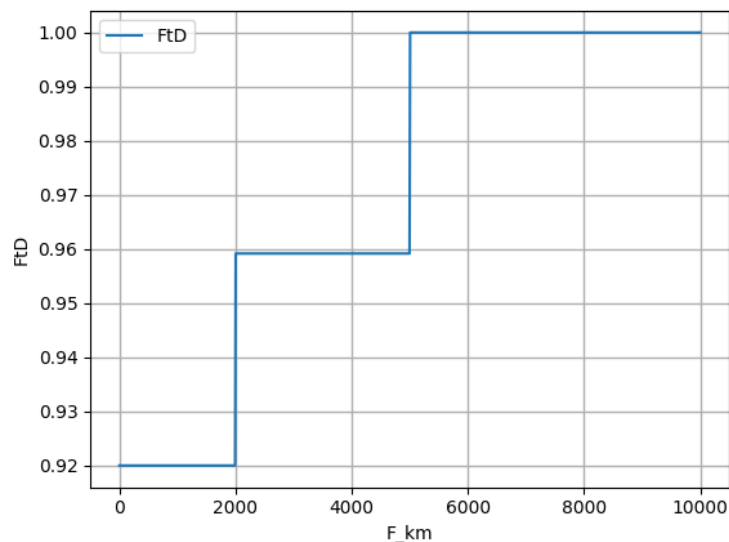


Figura 2.8: Variazione del FtD in funzione della frequenza di guida

In figura 2.9 viene mostrato lo studio di variabilità del FtD in funzione dell'esperienza, ossia degli anni dal conseguimento della licenza di guida; pertanto data la formula del FtD proposta in precedenza viene fatto variare il valore degli anni dal conseguimento della patente, tenendo a valore costante nullo il risultato derivante dalle distrazioni e dalle emozioni. Inoltre viene considerato come costante il valore relativo alla frequenza di guida, si ipotizza un chilometraggio percorso annualmente pari a 4000, pertanto il peso risultante applicando 2.9 risulta pari a 0.05.

Nell'analisi dei dati presentati in figura 2.9, si osserva che il valore dell'esperienza di guida ha solo una modesta influenza sul valore complessivo del FtD. Questo andamento a gradini è dovuto ai pesi prestabiliti associati all'esperienza di guida, tale andamento è smussato solo dal peso inferiore ad 1 legato alla frequenza di guida. Globalmente, si può notare che la frequenza di guida incide al massimo con un fattore di 0.04 sul FtD. Complessivamente, i dati mostrano una moderata dispersione attorno alla media, con un intervallo relativamente stretto tra il valore massimo e minimo. La varianza e la deviazione standard confermano questa osservazione, indicando una relativa omogeneità dei dati rispetto alla media.

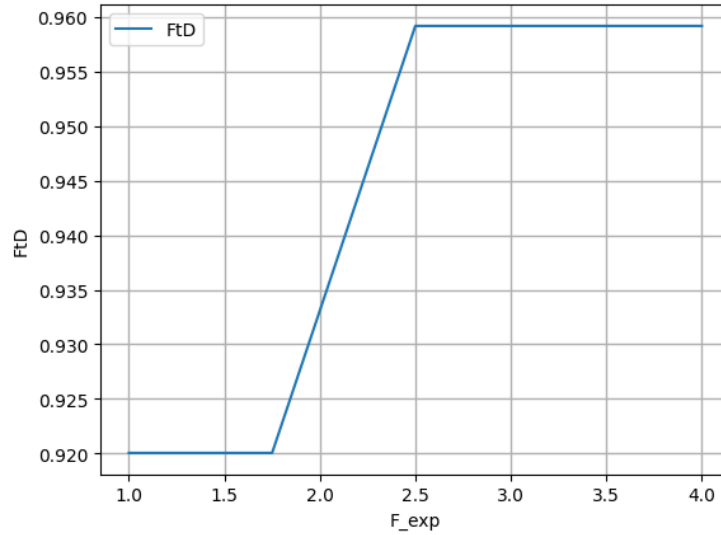


Figura 2.9: Variazione del FtD in funzione dell'esperienza di guida

2.3 Human Digital Twin

Nel capitolo precedente abbiamo introdotto il concetto di human digital twin (HDT) e abbiamo evidenziato come non esista un modello di architettura predefinito per rappresentarlo, a causa della sua natura estremamente personale e contestuale.

In questa sezione, ci concentreremo sulla definizione di un'architettura strutturata in grado di descrivere un HDT. A tale scopo, ci basiamo sul modello proposto in [19] e [18], che riguarda la rappresentazione di un virtual user (VU).

Secondo questo modello, ogni HDT è caratterizzato da alcune funzionalità, tra le più rilevanti troviamo: lo *User context* e lo *User profile*. Il primo risulta fondamentale per la gestione di tutte le applicazioni che considerano la posizione e più in generale il contesto in cui è presente una persona per poter operare; mentre il secondo fornisce informazioni importanti sulle caratteristiche del soggetto, permettendo così la personalizzazione delle applicazioni, tra cui quelle che utilizzano il contesto appena citato. Da questo ne deriva che la rappresentazione del VU, per come pensata dall'autore, presenta per l'utente

un profilo. Sulla base di quanto appena descritto si vuole proporre un modello HDT caratterizzato da un profilo statico e uno dinamico.

Il **profilo statico** rappresenta gli **stereotipi** pertanto, contiene le caratteristiche tipiche di un gruppo di utenti in un particolare dominio applicativo insieme ad un insieme di condizioni di attivazione, che rendono possibile identificare gli utenti appartenenti a questo gruppo. Per poter operare al meglio è stato deciso di suddividere gli stereotipi in classi e moduli.

Le **classi** rappresentano le diverse caratteristiche dell'utente, come ad esempio l'età e la frequenza ed esperienza, e influiscono su specifici dati in base al contesto di riferimento. Ad esempio, potremmo avere una classe relativa all'età della persona, all'interno della quale sono definiti contesti specifici, come la guida o la gestione di una *smart house*. Grazie all'età e al contesto specifico, la classe restituirà un valore che indica in che modo e in che misura l'età influisce su quel determinato contesto.

Per quanto riguarda i **moduli**, essi rappresentano le condizioni di attivazione che permettono di identificare gli utenti in un particolare dominio applicativo, ossia nel contesto. Come si può intuire quindi, anche tali condizioni sono legate al contesto e alle caratteristiche dell'utente. I moduli, pertanto, si occupano di combinare i dati provenienti dai sensori associati, elaborarli e combinarli con i dati ottenuti dalle classi di riferimento del contesto descritte in precedenza, al fine di ottenere lo stato del HDT in relazione al contesto. Sulla base dello stato ottenuto, e quindi l'identificazione dell'utente nel contesto, sarà poi possibile effettuare determinate azioni.

Il **profilo dinamico**, al contrario, ha lo scopo di aggiornare le classi individuate nel profilo statico; per farlo terrà conto, per ogni contesto della classe, sia dei dati raccolti dai moduli che sfruttano tale classe, che i dati provenienti da altre fonti. L'obiettivo che si vuole raggiungere consiste quindi nel riuscire a imparare dai dati elaborati in passato al fine di migliorare e calibrare il peso della classe nel contesto.

Il modello generale proposto è possibile osservarlo nella figura 2.10. Come si può notare, dato un contesto, i dati provenienti dai sensori vengono recepiti dal rispettivo modulo, questo li elabora sfruttando anche i dati ottenuti dalle

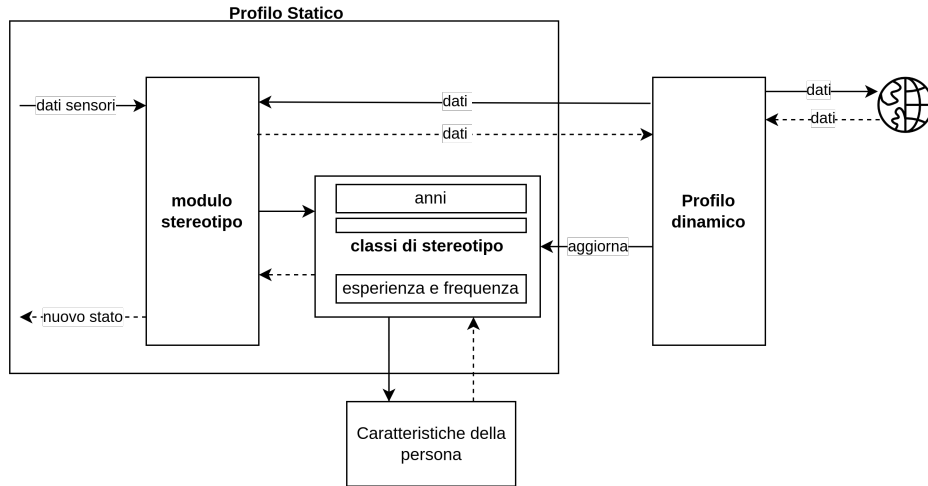


Figura 2.10: Modello di human digital twin proposto.

classi relative al contesto. Una volta terminata l'elaborazione restituiscono in output lo stato dell'utente in funzione del contesto. Analizzando invece la parte destra della figura 2.10 è possibile osservare il comportamento del profilo dinamico. Questo infatti, presi i dati dal modulo di uno stereotipo ed eventualmente aggiungendo alcune informazioni provenienti da altre fonti, elabora i dati, attraverso ad esempio degli algoritmi di *machine learning*, e ottenuto il risultato lo impiega per aggiornare le classi di stereotipo relative al contesto analizzato.

Attraverso questa architettura modulare e contestuale, sarà possibile rappresentare e gestire i molteplici aspetti che compongono l'identità digitale di un individuo.

2.4 Design del progetto

Sulla base di quanto appena descritto, viene di seguito illustrato il design ideato per assolvere agli obiettivi del seguente volume di tesi.

Il sistema che si vuole andare a realizzare si compone di due parti principali:

- lo human digital twin associato alla persona;

- una libreria che si occupa di gestire il profilo statico, e quindi gli stereotipi;

Per la realizzazione delle due componenti si è deciso di adottare un'architettura a microservizi; infatti l'impiego di tale architettura porta con sé diversi vantaggi, come: l'isolamento dagli errori a singoli componenti del sistema, una maggiore scalabilità, semplicità nel deployment etc.

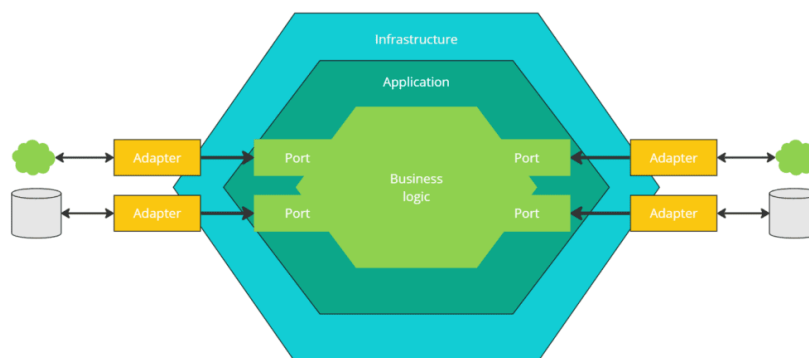


Figura 2.11: Architettura esagonale (fonte [26]).

Ogni microservizio è stato implementato seguendo una struttura architettonica esagonale, comunemente denominata "port and adapters" (figura 2.11), con l'obiettivo primario di ottimizzare l'autonomia e l'isolamento della logica di dominio, conformemente alle linee guida fornite dal paradigma del Domain-Driven Design (DDD). In tale contesto, il principio guida consiste nel separare la logica di dominio di tali servizi dal resto del sistema, in modo da renderla completamente indipendente dalle varie tecnologie e interfacce impiegate. Questa autonomia è conseguita attraverso l'attuazione del principio di inversione delle dipendenze, il quale stabilisce che un livello architettonico dovrebbe dipendere soltanto dai livelli inferiori e non da quelli esterni. Pertanto, la logica di dominio rappresenta il cuore centrale dell'architettura, rimanendo immune da qualsiasi forma di dipendenza, mentre il livello applicativo è l'unico autorizzato a farvi riferimento. Infine, il livello infrastrutturale è consentito ad avere dipendenze sia dal livello applicativo che da quello di dominio.

Ogni servizio sarà costituito da tre elementi:

- *domain*, rappresenta la logica di dominio;
- *port*, rappresentano le interfacce che definiscono come il mondo esterno può interagire con la logica di dominio all'interno di un'applicazione;
- *adapters*, svolgono il ruolo di collegamento tra le porte (input/output ports) e le componenti interne dell'applicazione, come la logica di dominio e la persistenza dei dati. Gli adapters agiscono come intermediari che traducono le richieste e le risposte provenienti dalle porte in formati comprensibili per le componenti interne e viceversa, consentendo di separare gli aspetti tecnici dalle funzionalità di business dell'applicazione.

Nelle sezioni successive verranno analizzate più in dettaglio tali componenti.

2.4.1 Human Digital Twin

Al fine di realizzare un HDT con le caratteristiche descritte nella sezione precedente è necessario in primo luogo andare a definire ciò che caratterizza una persona. A tal proposito sono stati individuati tre elementi, i quali si distinguono sulla base delle informazioni contenute e alla frequenza con le quali queste vengono aggiornate. Ogni persona, secondo il modello proposto è definita da:

- **dati generali**: i quali rappresentano le informazioni generali dell'utente, come: nome, cognome, data di nascita, indirizzo, data di conseguimento della licenza di guida ecc. Tali informazioni risultano essere quelle la cui frequenza di aggiornamento è più bassa, infatti bisogna tenere in considerazione che a parte alcuni dati quali da data di nascita, altri possono, se previsto dalla normativa, essere modificate; ad esempio in Italia è consentito cambiare cognome. All'interno di tali dati verrà compreso anche l'identificativo univoco associato al digital twin della persona.
- **caratteristiche**: rappresentano delle informazioni aggiuntive della persona e che ne definiscono una sua caratteristica, come ad esempio i chilometri percorsi annualmente, i decimi e le diottrie ecc. Tali dati risultano

essere aggiornati con una frequenza media, infatti i chilometri dovranno essere aggiornati secondo una finestra temporale di un anno, i decimi potrebbero diminuire nell'arco di mesi e così via.

- **stato attuale**: rappresenta lo stato dell'utente sulla base dei valori ottenuti dai sensori a cui esso è sottoposto. Tali dati risultano altamente dinamici e pertanto hanno una frequenza di aggiornamento elevata.

Oltre ai dati appena descritti, ogni HDT terrà traccia dei sensori ai quali è sottoposta la persona ad esso associato, e dello stato in cui questi riversano (ON/ OFF).

I dati della persona verranno poi utilizzati ed aggiornati attraverso gli stereotipi di cui dispone lo HDT, pertanto gli stereotipi, o più precisamente i moduli di stereotipo dovranno potere essere aggiunti, configurati, azionati e stoppati a seconda del contesto in cui si trova la persona.

Lo HDT, come detto, verrà realizzato seguendo un'architettura esagonale e la suddivisione esposta in precedenza:

- *domain*, composto da `PersonService`, che si occupa di gestire la vera e propria logica che viene implementata all'interno dello HTD, e l'entità `Person` che si occupa di rappresentare la persona secondo quanto appena descritto;
- *port*, definisco le interfacce per far comunicare la logica di dominio e l'esterno, sia in entrata che in uscita, in particolare la comunicazione avverrà con la base di dati, per effettuare chiamate ai servizi esterni, cioè gli stereotipi, e per servire le richieste degli utenti;
- *adapter*, implementano le porte andando ad aggiungere gli aspetti tecnici, come ad esempio la scelta della base di dati da utilizzare o della comunicazione con l'utente. Tali scelte verranno discusse all'interno del capitolo successivo (capitolo 3).

Nella figura 2.12 viene mostrato ad alto livello il diagramma delle classi dello HDT realizzato.

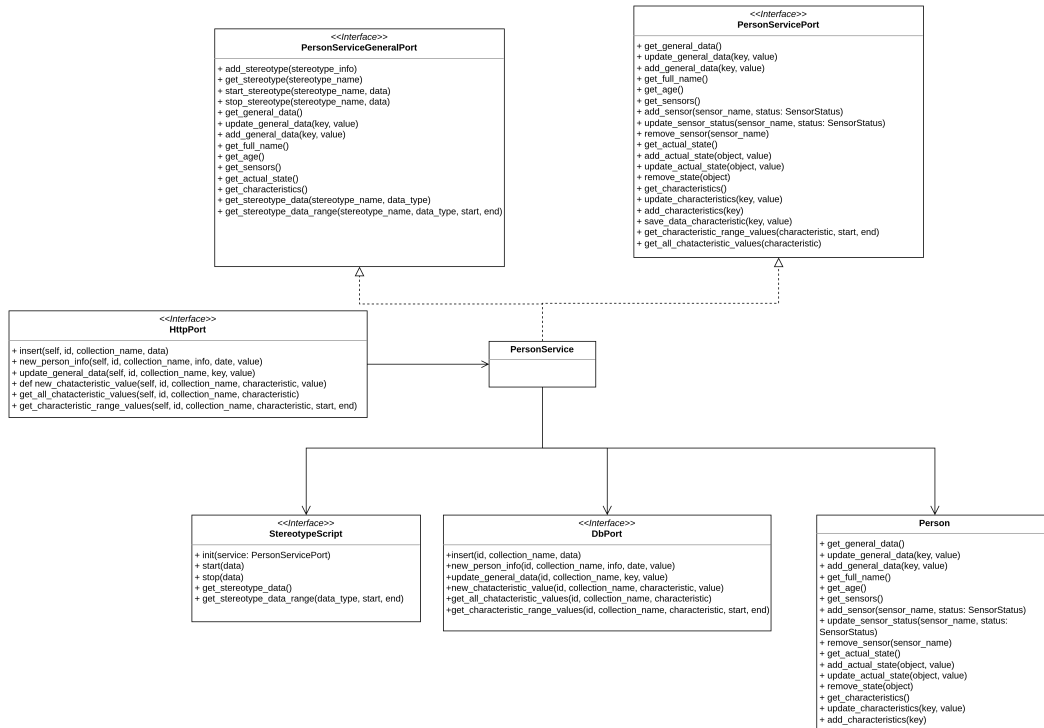


Figura 2.12: Diagramma delle classi dello HDT.

2.4.2 Stereotipi

Lo stereotipo, come detto precedentemente, si occupa di definire il profilo statico di una persona, in termini di classi e moduli. In questa sezione verrà descritto un esempio di realizzazione di uno stereotipo applicato all'ambito automobilistico, più precisamente al calcolo del Fitness-to-Drive personalizzato sulla base delle caratteristiche e delle risposte del guidatore. Di seguito vengono descritte le classi e i moduli realizzati.

Classi di stereotipo

Seguendo il modello di formula proposto nella sezione 2.2, emerge la possibilità di distinguere due categorie di stereotipi di rilievo: l'età del guidatore e la frequenza combinata all'esperienza di guida. Entrambe queste categorie vengono ponderatamente considerate tenendo presente il quadro di riferimento, che, in questo specifico contesto, si focalizza sul mondo automobilistico.

Le categorie menzionate svolgono un ruolo cruciale nell'attribuire un peso al valore che rappresentano. Nello specifico, la categoria relativa all'età valuta il contributo dell'età del conducente. D'altro canto, la categoria associata alla frequenza ed esperienza quantifica l'effetto sulla formula del numero di anni dal conseguimento della licenza di guida e i chilometri percorsi annualmente. Questa valutazione, in entrambi i casi, tiene conto delle peculiarità proprie al contesto di guida descritte nel capitolo precedente e consentendo di attribuire un peso appropriato alle diverse fasce d'età dei conducenti e agli anni trascorsi dal conseguimento della patente di guida e ai chilometri percorsi annualmente al fine di modellare in modo accurato l'impatto che questi hanno sulla sicurezza e l'idoneità di guida.

In accordo con l'approccio scelto per la modellazione del HDT, ciascuna classe di stereotipo restituisce un valore che indica come e in che misura quella specifica classe influisce in un determinato contesto. Ciò implica che ogni classe può essere impiegata in vari contesti. Ad esempio, la classe relativa all'età può essere utilizzata in contesti automobilistici, oftalmologici, di smart house, e così via. Pertanto, è stato deciso di strutturare ogni classe in modo che includa diverse definizioni specifiche per ciascun contesto. Considerando il caso d'uso trattato nella presente tesi, verranno definite due funzioni, ciascuna associata a una classe di stereotipo. Tali funzioni calcoleranno il peso delle caratteristiche dell'utente nell'ambito del contesto di guida, secondo le formule definite nella sezione 2.2:

- `drive_age_weight(age)`, che calcola il peso relativo all'età;
- `drive_experience_and_frequence_weight(licence_year, yearly_km)`, che calcola il peso connesso alla frequenza e all'esperienza di guida.

Poiché, come precedentemente accennato, il modello di HDT che si intende realizzare include un profilo dinamico che si occupa di aggiornare le classi di stereotipo in base ai dati storici relativi a un determinato contesto, ciascuna classe di stereotipo sarà dotata dei metodi necessari all'aggiornamento delle rispettive funzioni. Tali metodi saranno denominati `update_X(function)`, in

cui X rappresenta il contesto che si desidera aggiornare e `function` indica la nuova funzione da impiegare per il calcolo del peso della classe nel contesto X .

Moduli di stereotipo

I moduli di stereotipo rappresentano elementi fondamentali all'interno dell'architettura proposta. Ogni modulo svolge un ruolo specifico e dettagliato nel contesto che descrive, riunendo una serie di funzioni e logiche che concorrono a calcolare lo stato attuale dell'individuo. In altre parole, questi moduli si occupano di raccogliere dati dai sensori, elaborarli e aggiornare lo stato dell'utente relativamente al contesto. L'obiettivo principale dei moduli di stereotipo è garantire che ciascun calcolo e processo associato al contesto sia gestito in modo accurato e autonomo. Inoltre, ciascun modulo può fare affidamento su microservizi esterni per compiti computazionalmente intensi, che vengono delegati in modo efficiente e reattivo. Pertanto, l'adozione dell'architettura esagonale risulta cruciale per delineare chiaramente i ruoli e i confini funzionali dei moduli di stereotipo. Questo approccio, infatti, assicura che la logica di dominio sia completamente isolata da aspetti tecnologici, favorendo così la modularità e la manutenibilità dell'architettura. Infine, la strategia di utilizzare microservizi esterni contribuisce ulteriormente a garantire l'ottimizzazione dei calcoli e la scalabilità dell'intero sistema.

Considerando il caso d'uso trattato nella presente tesi, verrà definito un modulo, nominato **ftd**, il quale una volta aggiunto agli stereotipi dello HDT potrà essere configurato in modo tale da ricevere i dati provenienti dai sensori posti all'interno del veicolo e una volta raccolti delegare il calcolo vero e proprio al microservizio **ftdMS**. Quest'ultimo si occuperà di calcolare, secondo le formule viste in precedenza (sezione 2.2), il valore del Fitness-to-Drive personalizzato, storicizzando i dati e inviando il valore elaborato allo stereotipo cosicchè questo possa aggiornare lo stato attuale del guidatore.

Nella figura 2.13 viene presentato il diagramma delle classi che rappresenta il modulo di stereotipo denominato **ftd**. Questo diagramma offre una panoramica delle relazioni tra le classi all'interno del modulo, mettendo in

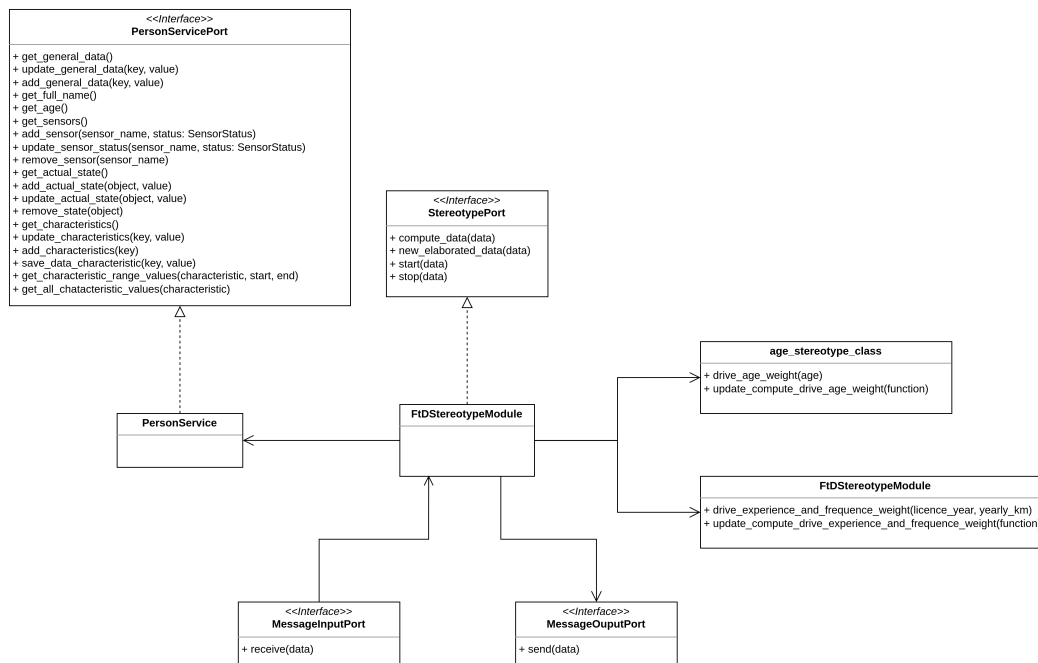


Figura 2.13: Diagramma delle classi del modulo di stereotipo ftd.

evidenza come ciascuna classe interagisce con le altre per implementare il calcolo dell'idoneità alla guida personalizzata. Nella figura 2.14 è invece possibile esaminare il diagramma delle classi relativo al microservizio incaricato dell'effettivo calcolo dell'idoneità di guida. Anche in questo caso, come già spiegato in precedenza, si è scelto di adottare l'architettura esagonale. Questa scelta conferisce al microservizio una struttura ben definita, che permette una chiara separazione tra la logica di dominio e gli aspetti di interfacciamento e adattamento dei dati.

2.4.3 Interazioni

Nella seguente sezione verranno presentati due diagrammi delle interazioni, le quali si ritengono le più significative. Il primo diagramma (figura 2.15) illustra un caso astratto di interazioni che coinvolgono l'utilizzo di un modulo di stereotipo generico. Questo diagramma offre un quadro generale delle relazioni tra i componenti all'interno del contesto di utilizzo del modulo di stereotipo.

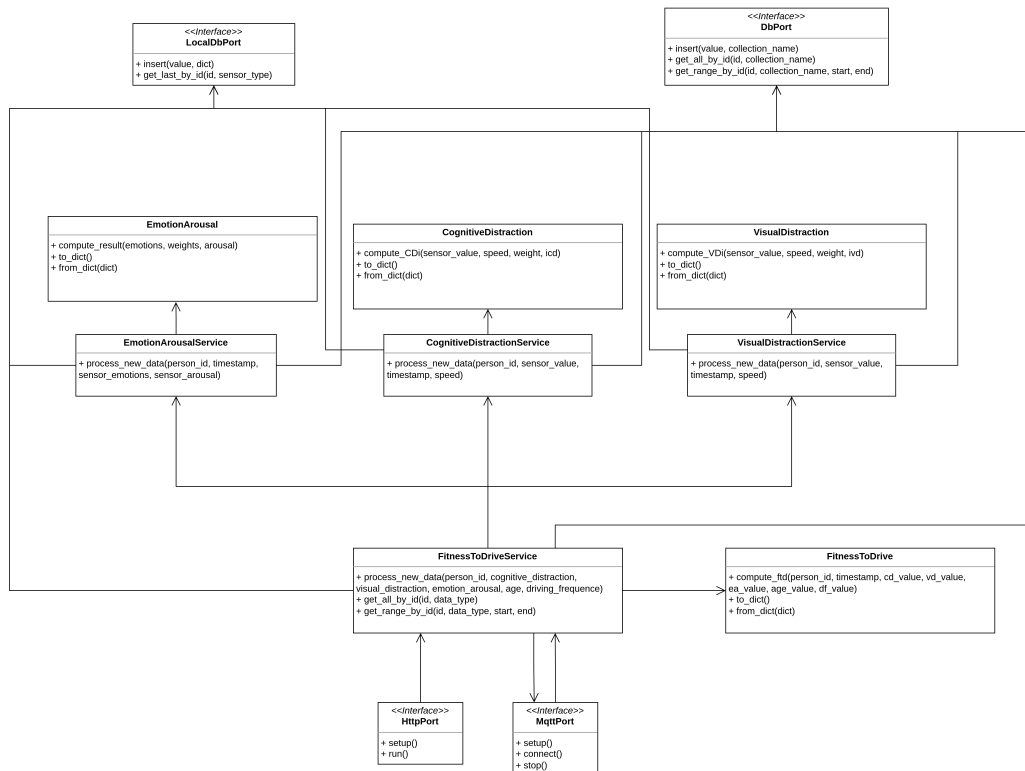


Figura 2.14: Diagramma delle classi del microservizio ftdMS incaricato del calcolo del FtD.

Il secondo diagramma (figura 2.16) invece, riguarda le interazioni specifiche legate al caso d'uso per il calcolo del Fitness-to-Drive personalizzato. Questo diagramma fornirà una rappresentazione chiara e dettagliata delle interazioni tra le diverse parti coinvolte nel processo di calcolo dell'idoneità alla guida su misura.

Nel diagramma delle interazioni astratto (figura 2.15) è possibile osservare il flusso di operazioni iniziate a seguito dell'installazione del modulo da parte di un operatore. Dopo l'installazione, il modulo è pronto per essere utilizzato; per fare ciò è necessario effettuare una configurazione iniziale, attraverso una route predefinita, in modo che possa ricevere i dati dai sensori di cui necessita per il suo funzionamento. Una volta completata la configurazione, iniziano le comunicazioni tra i sensori e il modulo stesso. Quando il modulo riceve un

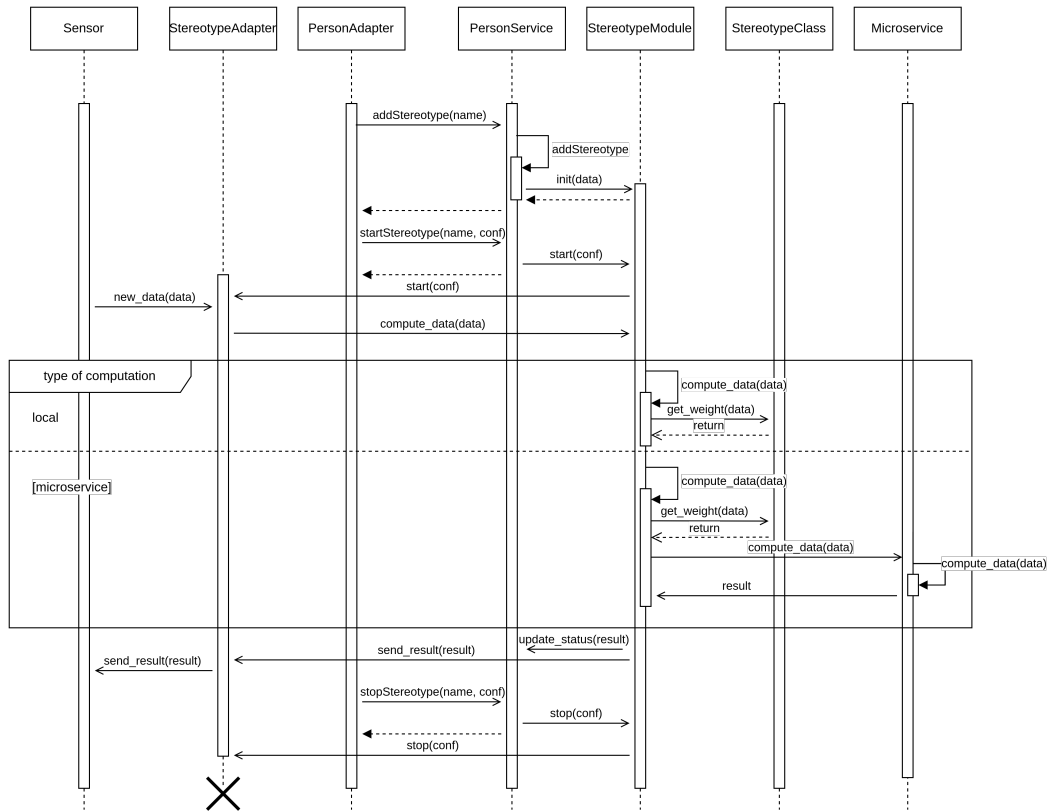


Figura 2.15: Diagramma delle interazioni generico.

dato dai sensori, si apre la possibilità di due percorsi: il modulo può decidere di comunicare con il microservizio ad esso associato, delegando così parte del calcolo, oppure può elaborare il dato in modo locale. Indipendentemente dal percorso scelto, una volta ottenuto il risultato complessivo del calcolo, il modulo si occupa di aggiornare lo stato della persona in relazione al contesto di applicazione specifico del modulo stesso. Quando richiesto dall'utente il modulo di stereotipo può essere stoppato; facendo ciò il modulo non sarà più in grado di ricevere i dati dai sensori.

In questo modo, il diagramma delle interazioni astratto offre una panoramica chiara dei passaggi coinvolti nell'utilizzo del modulo di stereotipo, dalla sua installazione alla configurazione iniziale, alla comunicazione con i sensori, alla decisione di calcolo e all'aggiornamento dello stato della persona.

Per quanto concerne le interazioni relative al caso d'uso per il calcolo del

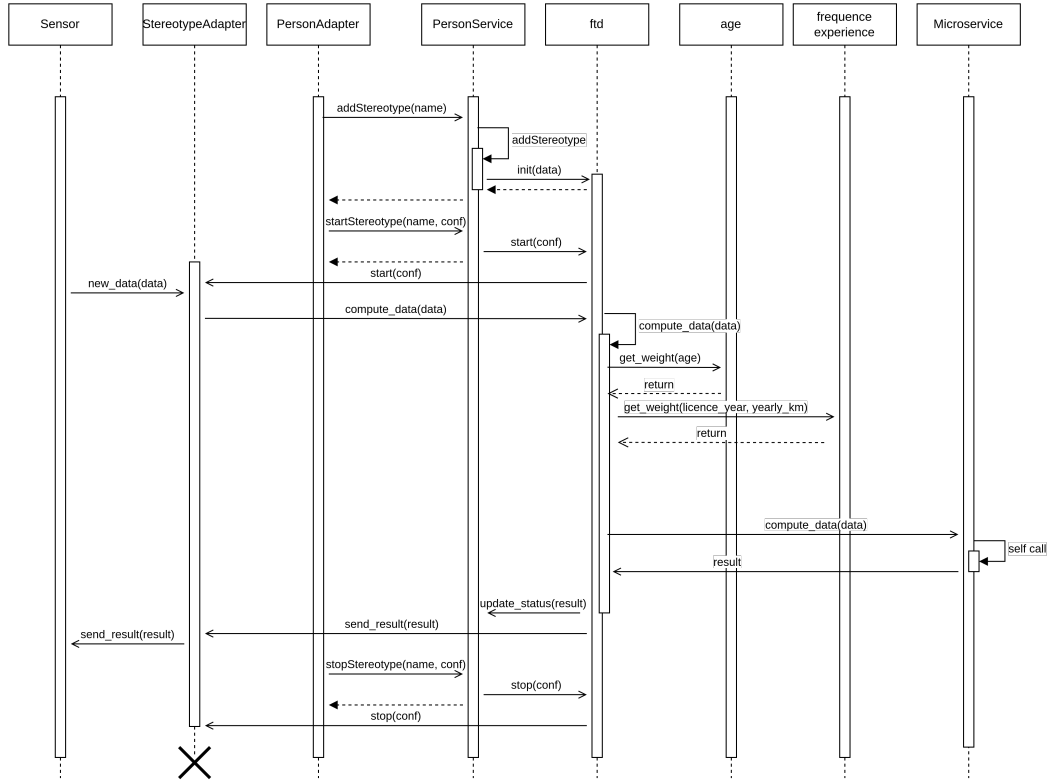


Figura 2.16: Diagramma delle interazioni per il caso d'uso FtD personalizzato.

Fitness-to-Drive personalizzato (figura 2.16), esse coinvolgono il modulo di stereotipo `ftd`, nonché le classi di stereotipo `age` e `frequency_and_experience`. Come anticipato, queste interazioni si svolgono nel contesto dell'uso pratico dello HDT. Inizialmente, il processo richiede l'installazione del modulo di stereotipo `ftd`, il quale viene aggiunto alla lista dei moduli disponibili per l'HDT della persona di riferimento. Successivamente, è necessaria la sua configurazione, al fine di iniziarne la comunicazione con l'autoveicolo. Dopo aver completato questi passaggi preliminari, durante la fase di guida dell'autoveicolo, quest'ultimo invia i dati rilevati dai sensori al modulo `ftd`. Il modulo, una volta ricevuti i dati e dopo aver eseguito i controlli necessari, procede a reperire i pesi relativi alle classi di stereotipo `age` e `frequency_and_experience`, basandosi sulle informazioni specifiche del guidatore. Una volta ottenuti tali pesi, il modulo li utilizza per inviare i dati raccolti al microservizio `ftdMS`, al

quale affida il compito di eseguire il calcolo effettivo dell'indice di idoneità. Una volta che il microservizio `ftdMS` ha completato il calcolo, il risultato viene restituito al modulo `ftd`. Quest'ultimo, a sua volta, si occupa di aggiornare lo stato dell'utente in base all'indice di idoneità calcolato e invia queste informazioni all'autoveicolo. In questo modo, il guidatore può essere tempestivamente informato sul suo stato di idoneità alla guida. Nel caso in cui il valore del Fitness-to-Drive risulti al di sotto di determinate soglie, l'autoveicolo, sulla base del valore dell'indice calcolato, attua le azioni correttive e di supporto necessarie per garantire la massima idoneità possibile e la sicurezza del guidatore. Terminata la guida il modulo `ftd` verrà disattivato. In tal caso, il modulo cessa di ricevere dati dai sensori e il processo di calcolo dell'idoneità verrà interrotto.

Capitolo 3

Implementazione

Il seguente capitolo ha lo scopo di motivare e dettagliare gli aspetti implementativi ritenuti rilevanti per una corretta comprensione del progetto. Inizialmente verranno descritte le tecnologie scelte per la realizzazione del sistema oggetto del seguente elaborato di tesi; successivamente verranno forniti alcuni esempi di implementazione atti a delucidare le scelte che si ritengono significative per la realizzazione del sistema illustrato nel capitolo 2.

3.1 Tecnologie utilizzate

Nel corso della realizzazione del progetto di tesi sono state impiegate diverse tecnologie, le quali sono state selezionate per soddisfare i requisiti specifici del progetto e hanno giocato un ruolo fondamentale nel raggiungimento degli obiettivi prefissati. In questa sezione verranno illustrate brevemente le tecnologie scelte per la realizzazione del sistema, descrivendo le caratteristiche principali e spiegando come sono state applicate nel contesto del progetto. In particolare verranno analizzate:

- Python
- Protocollo MQTT
- RabbitMQ

- FastAPI
- MongoDB
- Redis

Python Python rappresenta la base solida su cui è stato costruito l'intero sistema. Python è un linguaggio di programmazione ad alto livello, altamente versatile e noto per la sua semplicità e leggibilità del codice. La scelta di utilizzare Python è stata motivata da diversi fattori. In primo luogo è stata incentivata dall'obiettivo di mantenere continuità con il progetto realizzato nel volume di tesi *Studio del fitness-to-drive: sensing delle attività di guida e monitoraggio del driver*[31], in particolare con il modulo realizzato per il calcolo del FtD. In secondo luogo, la scelta per il suo impiego è stata guidata dalla volontà di approfondire la conoscenza di tale linguaggio, il quale sta diventando sempre più diffuso sia nell'ambito della ricerca che in quello lavorativo, anche grazie alle numerose librerie che mette a disposizione. Infine, la scelta è stata motivata anche dalla sua idoneità per eventuali future implementazioni di algoritmi di machine learning. Questi algoritmi possono contribuire alla realizzazione del profilo dinamico, ovvero l'aggiornamento delle classi di stereotipo in base ai dati raccolti nel tempo.

Protocollo MQTT MQTT è l'acronimo di Message Queuing Telemetry Transport e indica un protocollo di trasmissione dati TCP/IP basato su un modello di pubblicazione e sottoscrizione che opera attraverso un apposito message broker. MQTT è estremamente leggero, risultando quindi ideale per connettere dispositivi remoti con un ingombro di codice ridotto e una larghezza di banda di rete minima. Poiché MQTT offre una modalità asincrona di trasmissione dei messaggi, consentendo una comunicazione efficiente e affidabile tra i componenti, i messaggi MQTT, all'interno del progetto, sono stati utilizzati per realizzare la comunicazione tra il modulo di stereotipo e il microservizio a cui esso delega il calcolo vero e proprio del FtD, consentendo un flusso di informazioni in tempo reale.

RabbitMQ RabbitMQ è una piattaforma di messaggistica open-source che implementa il protocollo Advanced Message Queuing Protocol (AMQP). Questa tecnologia è stata scelta per gestire efficacemente il flusso di messaggi tra le varie componenti del sistema, garantendo l'affidabilità e l'ordine dei messaggi scambiati. La scelta del suo impiego è strettamente legata alle caratteristiche appena elencate. All'interno del progetto è stato utilizzato per consentire la comunicazione tra il veicolo, sul quale sono posti i sensori necessari a monitorare il guidatore e a calcolare la sua idoneità di guida, e il digital twin del guidatore, o più precisamente con il suo modulo relativo al calcolo del FtD, garantendo un flusso coerente dei dati all'interno del sistema.

FastAPI FastAPI è un moderno framework Web Python per la creazione di API veloci e scalabili basate su standard come JSON Schema e type hints. La scelta di utilizzare FastAPI come interfaccia di comunicazione con lo HDT e per reperire i dati dal microservizio incaricato del calcolo del FtD è giustificata dalle sue elevate prestazioni. FastAPI, infatti, è noto per essere uno dei framework Web più veloci disponibili, grazie alla sua capacità di gestire un grande volume di richieste in modo efficiente. Inoltre, offre un supporto integrato per la tipizzazione statica, il che significa che gli errori di tipo vengono individuati durante la fase di sviluppo invece che a runtime. Ciò migliora la qualità del codice e facilita la manutenzione dell'applicazione. Infine, la sintassi dichiarativa messa a disposizione dal framework, lo rende facile da usare e da comprendere. Questo riduce il tempo necessario per lo sviluppo, il testing e la manutenzione del codice, fornendo quindi una soluzione più efficiente e produttiva.

MongoDB MongoDB è un database non relazionale open source, in grado di elaborare dati strutturati, semi-strutturati e non strutturati. Si tratta di un database orientato ai documenti che sfrutta un linguaggio di query non strutturato. MongoDB, quindi, si allontana dalla struttura tradizionale basata su tabelle dei database relazionali in favore di documenti in stile JSON, rendendo l'integrazione di alcuni tipi di dati più facile e veloce. La scelta di utilizzo di

questo tipo di database è motivata da molteplici fattori. In primo luogo, la sua capacità di adattarsi a diversi modelli di dati conferisce una notevole flessibilità all'architettura del sistema. Questo è fondamentale quando ci si trova ad affrontare dati di natura eterogenea, comuni nei contesti moderni. Inoltre, MongoDB dimostra di mantenersi performante anche in presenza di grandi volumi di informazioni, garantendo tempi di accesso rapidi e un'esperienza utente reattiva.

Redis Redis è un sistema di archiviazione dei dati in memoria ad alte prestazioni, utilizzato per la memorizzazione di dati chiave-valore. Nel contesto del progetto, Redis è stato impiegato per la memorizzazione e la gestione di dati di supporto, come la memorizzazione nella *cache* dei risultati di calcoli o il mantenimento di informazioni di stato temporaneo, ciò ha permesso di migliorare notevolmente la velocità di accesso a informazioni frequentemente utilizzate. La scelta di utilizzare Redis è stata guidata da alcune sue caratteristiche, infatti questo *in-memory database* è particolarmente utile per le applicazioni che richiedono accesso veloce ai dati, come nel nostro caso, poiché memorizza i dati in memoria anziché su disco. Ciò significa che i dati possono essere recuperati quasi istantaneamente, senza la necessità di accedere al disco. Redis, inoltre, è anche altamente scalabile, consentendo di gestire grandi quantità di dati in modo efficiente.

3.2 Dettagli implementativi

Nella seguente sezione verranno analizzate più in dettaglio alcune scelte implementative effettuate al fine di realizzare il progetto oggetto del seguente elaborato di tesi. È opportuno sottolineare che, al fine di garantire l'integrità e l'efficacia del sistema sviluppato, è stato imprescindibile condurre un attento processo di *testing* su tutte le componenti coinvolte. A tal fine, sono stati eseguiti test sia manuali che automatici, con l'obiettivo di verificare il corretto funzionamento e l'aderenza alle specifiche previste.

Gestione ad eventi

La progettazione di sistemi complessi, quale la realizzazione dello HDT, richiede decisioni ponderate in termini architetturali. Come già descritto in precedenza nel capitolo 2, per quanto riguarda l'architettura generale del sistema, è stato deciso di adottarne una di tipo esagonale. Per quanto concerne, il punto di vista implementativo dell'architettura software, è stata effettuata una scelta basata sull'efficace gestione dei molteplici eventi generati dall'interazione dinamica tra l'HDT e potenziali utilizzatori come sensori, l'utente stesso o altri HDT. Tale scelta ha portato all'adozione di un'architettura basata su eventi, conosciuta anche come *event loop*. Tale approccio è stato selezionato in funzione delle caratteristiche e peculiarità ad esso legate. Innanzitutto, la gestione ad eventi consente un utilizzo efficiente delle risorse di sistema, evitando l'*overhead* associato alla creazione di *thread* o processi separati. Inoltre, questo modello di esecuzione consente di gestire in modo efficiente le attività asincrone e gli eventi in tempo reale, semplificando la comunicazione e la sincronizzazione tra le diverse parti del sistema. Infine, un ulteriore vantaggio derivante dall'uso degli *event loop* risiede nella loro efficienza in termini di consumo di risorse, il che li rende particolarmente adatti per scenari in cui numerosi eventi necessitano di essere gestiti in modo fluido e reattivo.

Gestione degli stereotipi

Gli stereotipi all'interno del progetto di tesi sono stati organizzati come una libreria chiamata **Stereotypes**. Tale scelta è stata motivata dalla possibilità di massimizzare la riusabilità del codice, condividere le implementazioni tra diversi progetti, agevolare la manutenibilità e consentire aggiornamenti agevoli. Inoltre, l'utilizzo di una libreria permette di stabilire uno standard per rappresentare il dominio trattato.

La libreria realizzata è strutturata in tre componenti: classi di stereotipi, moduli di stereotipi ed elementi generali. La prima parte contiene la definizione di tutte le classi di stereotipi, `age` e `frequence_and_experience`. La seconda parte contiene l'implementazione di tutti i moduli degli stereotipi,

come il modulo `ftd`, responsabile del calcolo dell'idoneità alla guida. Infine, gli elementi generali definiscono sia le interfacce comuni che possono essere utilizzate per creare un modulo di stereotipo, rappresentate dall'interfaccia `StereotypeScript`, o per accedere ai dati HDT, rappresentate dall'interfaccia `PersonServicePort`, sia per definire in modo univoco gli stati che i sensori che monitorano l'utente possono avere e che sono utilizzati dai moduli di stereotipo. L'interfaccia `StereotypeScript`, mostrata nel listato 3.1, riveste un ruolo fondamentale all'interno della libreria. Essa stabilisce i metodi necessari per configurare e utilizzare i moduli degli stereotipi, garantendo un'interazione coerente e strutturata con tali moduli.

```
def init(self, service: PersonServicePort)
def start(self, data)
def stop(self, data)
def get_stereotype_data()
def get_stereotype_data_range(data_type, start, end)
```

Listato 3.1: Interfaccia `StereotypeScript`

Per poter utilizzare uno stereotipo è necessario eseguire un processo di installazione specifico. Questa operazione può essere eseguita mediante un'apposita interfaccia API messa a disposizione dallo HDT stesso (listato 3.3). Alla *route* in questione vengono inviate le informazioni relative al modulo che si desidera installare. Dopo aver verificato che il modulo non sia già incluso tra quelli disponibili per lo HDT, viene eseguita la sua installazione. Durante questo processo, vengono considerati due possibili scenari relativi all'installazione del modulo. Nel primo scenario il modulo è già presente all'interno della libreria `Stereotypes`. La libreria in questione risulta di default già installata al momento dell'avvio dello HDT, ma le sue funzionalità non sono automaticamente importate all'interno dello HDT. In questa situazione, l'HDT esegue semplicemente l'importazione del modulo dalla libreria. Successivamente, viene creata un'istanza concreta della classe `StereotypeScript` per il modulo in questione, chiamata comunemente `Start.py`. Infine, l'HDT salva il modulo

tra quelli che potranno essere impiegati, attraverso un sistema chiave-valore, che ne permette successivamente il reperimento.

Nel secondo scenario, il modulo richiesto non è ancora presente all'interno della libreria, e pertanto sarà presente in una sua versione futura rispetto a quella attualmente installata. In tal caso, prima dell'importazione del modulo, si procede all'aggiornamento della libreria stessa, gestita da strumenti come *pip*¹ per Python. Effettuato l'aggiornamento il processo prosegue come descritto precedentemente: importazione del modulo, creazione dell'istanza `StereotypeScript` e il successivo salvataggio tra quelli che potranno essere utilizzati dallo HDT.

Gestione del profilo dinamico Nel capitolo 2 è stato accennato alla realizzazione di un profilo dinamico, il cui compito consiste nell'aggiornare le funzioni presenti all'interno delle classi di stereotipo a seguito delle analisi eseguite sui dati storici dei moduli che utilizzano tali classi per adempiere ai loro scopi. A tal proposito, sebbene non sia stato implementato l'algoritmo di analisi dei dati, sono state definite le interfacce e il sistema necessario al reperimento dei dati storici e all'aggiornamento delle funzioni.

Soffermandoci su quest'ultimo punto, nel listato 3.2 è possibile osservare come si è deciso di gestire l'aggiornamento delle funzioni delle classi di stereotipo. In particolare, si è scelto di gestire ciò attraverso una combinazione di variabili e funzioni, cosicchè i moduli che utilizzano quelle determinate classi non debbano essere aggiornati. Il meccanismo ideato prevede per ogni funzionalità messa a disposizione dalla classe quattro elementi:

- una variabile a cui viene assegnata la funzione che si occupa del calcolo relativo all'incidenza della classe di stereotipo sulla base dello specifico contesto (`_compute_drive_age_weight`);
- una funzione per consentire l'aggiornamento della variabile, assegnandogli una nuova funzione (`update_compute_drive_age_weight`);

¹pip: Python Package Index (PyPI) è un repository che consente di installare, aggiornare e gestire facilmente librerie e package di terze parti che estendono le funzionalità di Python


```
def _compute_age_value(age)-> float:
    ...
    return value

_compute_drive_age_weight: Callable[[int], float] =
    _compute_age_value

def drive_age_weight(age:int) -> float:
    return _compute_drive_age_weight(age)

def update_compute_drive_age_weight(function: Callable[[int],
float]):
    _compute_drive_age_weight = function
```

Listato 3.2: Gestione del profilo dinamico per la classe *age* nel contesto di guida

- una funzione di default da assegnare alla variabile (`_compute_age_value(age)`);
- una funzione che utilizzerà quella assegnata alla variabile e pertanto sarà quella effettivamente richiamata dai moduli di stereotipo (`drive_age_weight`).

Adapters per la comunicazione: HTTP, MQTT e AMQP

All'interno del progetto, poichè progettato secondo un'architettura esagonale, ogni servizio, espone i propri adapters i quali possono essere utilizzati dagli altri componenti del sistema per stabilire la comunicazione con esso. In particolare i servizi individuati risultano tre: lo HDT, il modulo di stereotipo e il microservizio incaricato del calcolo. Di seguito, per ognuno di questi servizi verranno esaminati gli adapters che mettono a disposizione. Prima di proseguire con l'analisi è opportuno sottolineare che per quanto riguarda gli adapter HTTP realizzati, ognuno di essi presenta delle rotte, le quali sono state ideate seguendo le linee guida delle API REST, per cui per ogni risorsa manipolata dal servizio sono state definite rotte differenti, ognuna delle quali si occupa di

gestire, mediante l'apposito handler, una o più delle operazioni CRUD (Create, Read, Update, Delete) che possono essere effettuate su di essa.

```
def _user_routes(self, router: APIRouter):
    @router.get('/user/general_data', status_code=200)
    @router.post('/user/general_data', status_code=201)
    @router.put('/user/general_data', status_code=204)
    @router.get('/user/characteristics', status_code=200)
    @router.post('/user/characteristics', status_code=201)
    @router.put('/user/characteristics', status_code=204)
    @router.get('/user/sensors', status_code=200)
    @router.get('/user/state', status_code=200)

def _stereotypes_routes(self, router: APIRouter):
    @router.post('/stereotype/add', status_code=201)
    @router.post('/stereotype/start', status_code=204)
    @router.post('/stereotype/stop', status_code=204)
    @router.get('/stereotype/get_data', status_code=200)
    @router.get('/stereotype/get_data_range', status_code=200)
```

Listato 3.3: HDT routes

Per poter comunicare con la componente principale del sistema, rappresentata dallo HDT, si è scelto di utilizzare il protocollo HTTP. Osservando le rotte definite tramite l'adapter HTTP realizzato (listato 3.3) è possibile individuare le funzionalità che mette a disposizione; in particolare le rotte possono essere suddivise in due categorie: la prima è relativa alle informazioni dello HTD, per cui permettono di reperire, aggiornare ed inserire le informazioni riguardanti i dati generali, le caratteristiche, lo stato attuale e lo stato dei sensori a cui la persona è sottoposta. La seconda categoria è invece relativa agli stereotipi e ha lo scopo di permetterne l'installazione dei moduli, il rispettivo avvio e terminazione e il reperimento dei dati salvati da questi ultimi.

Il secondo servizio è rappresentato come detto dal modulo di stereotipo

realizzato, ossia il modulo `ftd`. Analizzando tale servizio è possibile notare la presenza di due adapters: il primo è rappresentato da quello consente la comunicazione attraverso il protocollo MQTT, mentre il secondo per quella tramite AMQP, più precisamente per consentire la comunicazione attraverso la piattaforma RabbitMQ. Al fine di gestire al meglio l'implementazione degli adapter è stato deciso di adottare il pattern **Strategy** pertanto ogni Adapter sarà composto da tre elementi (figura 3.1):

- **ClientAdapter**, incaricato del setup del protocollo utilizzato e della gestione del suo avvio e terminazione;
- **InputAdapter**, incaricato di gestire la ricezione dei messaggi attraverso il metodo `receive(data)`;
- **OutputAdapter**, incaricato di gestire l'invio dei messaggi attraverso il metodo `send(data)`;

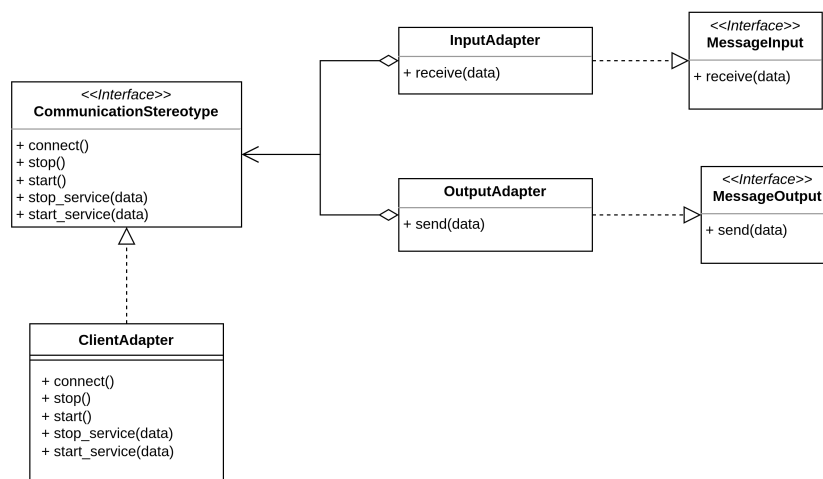


Figura 3.1: Rappresentazione del pattern Strategy relativo agli adapter.

`MQTTAdapter` ha il compito di gestire la comunicazione tra il modulo di stereotipo (`ftd`) e il microservizio `FtDMS`, incaricato del calcolo vero e proprio del `FtD`. In particolare, per permettere la comunicazione tra i due sono stati scelti appositi `topic` che sono stati poi sottoscritti da `publisher` e `subscribes`

al fine di inviare e ricevere i messaggi. Relativamente agli adapter MQTT, secondo il modello appena descritto, è possibile notare, oltre al modulo principale, due adapters distinti: `MQTTInputAdapter`, che si occupa di ricevere il nuovo valore di FtD calcolato dal microservizio e inviarlo al modulo preposto per la sua gestione, e `MQTTOutputAdapter`, il quale è incaricato di inviare i dati semi-elaborati al microservizio FtDMS al fine di delegargli il calcolo.

`RabbitMQAdapter` si occupa di gestire la comunicazione tra il modulo di stereotipo (`ftd`) e il veicolo su cui si trova la persona a cui è associato lo HDT detentore del modulo di stereotipo. In particolare, al fine di indirizzare i messaggi contenenti i dati rilevati dai sensori e comunicare il valore computato per il FtD è stato scelto di utilizzare un *exchange* di tipo `topic`, pertanto sono state individuate diverse *routing key* sui quali produttori e consumatori si mettono in ascolto ed inviano e ricevono i messaggi. Relativamente agli adapter `RabbitMQ`, come per quelli MQTT, oltre al modulo principale, è possibile notare due adapters distinti: `RabbitMQInputAdapter`, che si occupa di gestire i dati provenienti dai sensori e quindi in input al modulo di stereotipo, e `RabbitMQOutputAdapter`, il quale è incaricato di inviare al veicolo il valore del FtD calcolato.

L'ultimo servizio che espone degli adapter relativi alla comunicazione è rappresentato dal microservizio FtDMS. Tale servizio adotta due tipologie di adapters, il primo, come si può intuire da ciò che è stato appena descritto, è rappresentato dal `MQTTAdapter`, che ha lo scopo di ricevere i dati semi-elaborati che compongono il FtD e inviarli al modulo preposto per la loro elaborazione e storage, il secondo è rappresentato da un adapter HTTP. Quest'ultimo adapter, ha il compito di ricevere le richieste relative al reperimento dello storico dei dati elaborati dal microservizio; in particolare le rotte che mette a disposizione, come si può osservare dal listato 3.4, sono due ed entrambe si basano sull'identificativo dello HDT e il codice identificativo relativo al parametro per reperire i dati storici, che nel primo caso riguardano tutti i dati memorizzati, mentre nel secondo solo quelli relativi al range temporale richiesto.

```
@router.get('/all')
async def get_all_by_id(id, data_type)

@router.get('/range')
async def get_range_by_id(id, data_type, start, end)
```

Listato 3.4: ftdMS HTTP adapter routes

3.3 Comunicazione tra Digital Twin: considerazioni

In questa sezione verranno illustrate alcune considerazioni ed idee riguardanti la gestione della comunicazione tra digital twin, nonostante questa soluzione, non essendo il focus del progetto di tesi, non sia stata concretamente implementata. In particolare, ci soffermeremo sulla comunicazione tra il DT del veicolo e lo HDT.

Per instaurare una comunicazione adeguata, è fondamentale che i DT valutino l'identità e i permessi della controparte coinvolta. Per farlo è quindi necessario instaurare un processo di autenticazione ed autorizzazione. Si tenga presente che l'autenticazione consiste nell'identificare e confermare l'identità dichiarata da una fonte, mentre l'autorizzazione concede o limita l'accesso a un componente o funzionalità protetta in base ai privilegi di accesso della fonte. A tal fine, in fase di progettazione, è stato ipotizzato che il processo avvenga mediante l'utilizzo di NFC (Near Field Communication). In particolare, l'utente sfrutterà un dispositivo ad esso associato, come può essere uno smartphone, dotato di tale tecnologia, al fine di comunicare la sua presenza e i dati di base relativi al suo HDT al DT con il quale vuole interagire. Grazie a questo processo di identificazione, il DT ricevente avrà le informazioni necessarie per avviare la comunicazione con lo HDT. Per completare questa fase di *handshaking*, il DT dovrà comunicare allo HDT la sua configurazione attraverso un'apposita *route*, ossia quella il cui compito consiste nell'avviare il modulo di stereotipo. Fatto ciò sarà possibile avviare l'effettiva comunicazione, durante

la quale avverrà lo scambio di informazioni tra i due DT.

```
{
  "name": "ftd",
  "data": {
    "rabbitmq_config" : {
      "host": "localhost",
      "port": 5672,
      "username": "guest",
      "password": "guest",
      "queues": ["NP_UNITO_DCDC",
        "AITEK_EVENTS",
        "Emotions",
        "NP_UNIPR_AROUSAL",
        "RL_VehicleDynamics"],
      "exchange": "ftdStereotype",
      "routing_key": "car"
    },
    "module": {
      "km": 100
    }
  }
}
```

Listato 3.5: esempio di configurazione del modulo

Nell'ambito del caso d'uso trattato in questa tesi, si può identificare il seguente processo: il guidatore, utilizzando uno smartphone dotato di NFC, notifica la sua presenza all'interno del veicolo ed invia al DT di quest'ultimo le informazioni relative al proprio indirizzo IP e la *route* da impiegare per configurare ed avviare il modulo di stereotipo *ftd*. Ricevute queste informazioni il DT del veicolo invia la configurazione 3.5 attraverso l'apposita *route*. Una volta avviato il modulo ha inizio lo scambio di informazioni: il DT del veicolo invierà allo HDT i dati rilevati dai sensori, mentre lo HDT informerà il DT del

valore del FtD calcolato. Ricevuta questa informazione il veicolo mostrerà attraverso il *cockpit* il valore di FtD e, sulla base di determinate regole, metterà in atto le azioni correttive necessarie.

Per lo sviluppo dell'approccio appena descritto si ritiene opportuno delineare alcune minacce a cui si potrebbe andare incontro. Innanzitutto è necessario considerare la sicurezza e la privacy dei dati, in quanto è possibile incorrere nella manipolazione dei dati da parte di intrusi. Questi potrebbero alterare il comportamento del sistema, diffondere informazioni false o analizzare i dati, violando così la privacy, attraverso l'accesso non autorizzato ai dati. Pertanto è fondamentale mettere in atto adeguati sistemi di crittografia dei dati e monitoraggio delle attività sospette e aggiornamenti regolari dei software. Inoltre, è importante considerare le possibili perdite di connettività e adottare sistemi di ri-connessione nel caso in cui si verifichino interruzioni nella comunicazione.

Capitolo 4

Valutazione sperimentale

Il capitolo che segue rappresenta una tappa cruciale nel percorso di questa ricerca, in quanto si concentrerà sull'analisi e la validazione dei risultati ottenuti. L'obiettivo è valutare l'efficacia delle soluzioni e delle strategie implementate nel contesto del progetto, nonché comprendere se queste rispondano in modo adeguato agli scopi prefissati. Infine, saranno tratte conclusioni significative dalle valutazioni, evidenziando i punti di forza della soluzione proposta e suggerendo possibili sviluppi futuri.

4.1 Valutazione della precisione del FtD

Nel contesto del presente studio, l'accuratezza e l'affidabilità del calcolo del FtD rivestono un ruolo di fondamentale importanza. Questo capitolo si dedicherà alla valutazione dettagliata della precisione del FtD, con l'obiettivo di esaminare quanto i risultati ottenuti riflettano la condizione reale del guidatore. Inoltre, si analizzerà come l'utilizzo di un sistema di notifica integrato attraverso l'interfaccia uomo-macchina (HMI) all'interno del veicolo, quindi attraverso il cockpit, possa contribuire al miglioramento della sicurezza stradale.

4.1.1 Descrizione dei test effettuati

I test di validazione verranno effettuati a partire dai dati ottenuti durante la fase di test con utenti reali nell'ambito del progetto NextPerception (descritto nei capitoli 1 e 2). Le prove in questione hanno una durata di circa 15 minuti durante i quali vengono fatte sperimentare al guidatore alcune combinazioni di distrazione visuale e cognitiva, diverse emozioni positive e negative di cui si valuterà anche l'attivazione attraverso il valore di *arousal*. Della totalità del test è stato deciso di analizzarne una sottoparte della durata di due minuti, durante la quale il guidatore manifesta diversi comportamenti di distrazione e diverse emozioni. Al fine di procedere alla validazione dei risultati sono state poste le seguenti condizioni:

1. è stato ipotizzato un limite di velocità sul tratto percorso pari a 30 km/h;
2. per la valutazione del sistema in presenza di notifica del FtD si ipotizza che il guidatore venga avvertito attraverso una notifica visiva e uditiva qualora il valore del FtD risulti al di sotto della soglia prestabilita pari allo 0.75. A seguito della notifica il guidatore ritorna in uno stato di completa idoneità fino al momento in cui non si ripresenti nuovamente una situazione che faccia decrementare il FtD.

4.1.2 Test

Di seguito verranno illustrati i test eseguiti, suddivisi sulla base dei dati relativi al guidatore e sulla versione di sistema adottato. In particolare, **FtD v0** si riferisce al sistema impiegato all'interno del progetto NextPerception, descritto nei capitoli 1 e 2, e rappresenta la versione base per il calcolo del FtD; mentre **FtD v1** rappresenta il sistema oggetto del seguente studio e pertanto basato sul concetto di HDT e FtD personalizzato.

Nella figura 4.1 è possibile osservare la velocità tenuta dal guidatore durante la simulazione in assenza di notifica del proprio FtD. Come si può notare sono presenti degli istanti in cui il guidatore supera il limite di velocità, che per le condizioni descritte in precedenza è stato posto a 30 km/h.

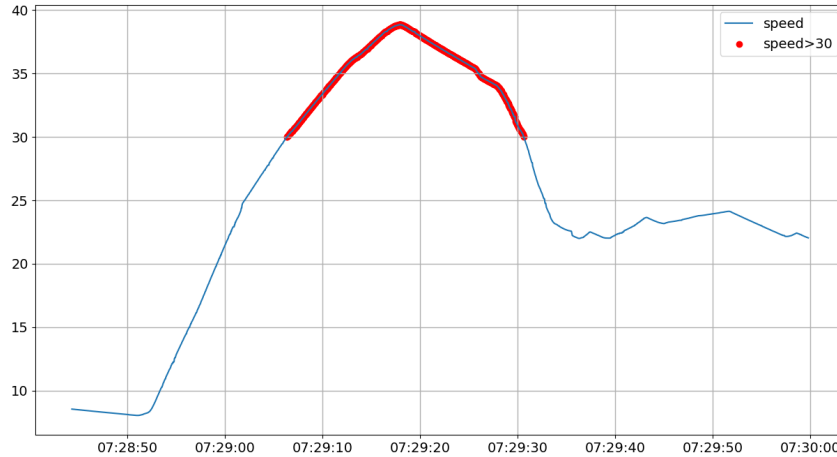


Figura 4.1: Velocità tenuta dal guidatore durante il test.

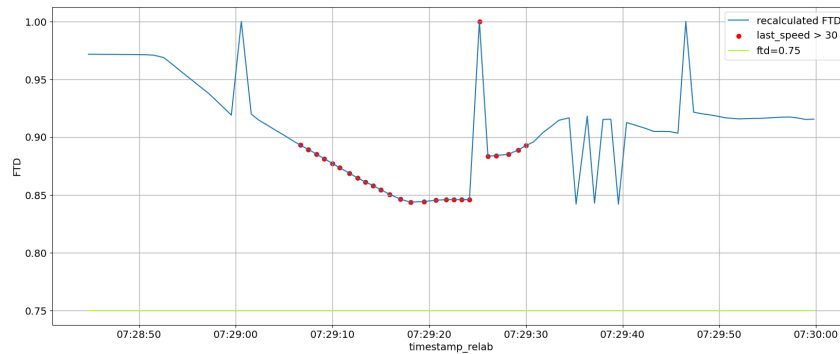
L'obiettivo dell'analisi che andremo a fare sarà quello di andare a verificare come varia il FtD del guidatore e come questo, se opportunamente notificato, riuscirebbe ad andare a limitare le situazioni di pericolo.

Valutazione prestazioni del FtD versione v0

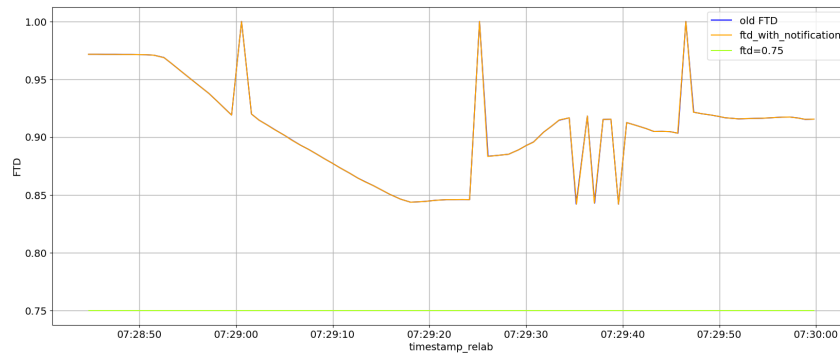
Come base di confronto è stato scelto di analizzare il FtD rilevato durante la simulazione tenuta nell'ambito del progetto NextPerception. Si ricorda che tale valore è stato ottenuto adottando la formula precedente, la quale non considerava i fattori personali del guidatore, pertanto indipendentemente dalle caratteristiche del guidatore, a parità di distrazioni, emozioni e velocità il valore rilevato di FtD sarà sempre il medesimo.

Nel grafico di figura 4.2a è possibile osservare il valore del FtD registrato durante il test ed in particolare il valore di quest'ultimo in corrispondenza degli istanti in cui è stata rilevata una velocità superiore al limite consentito (ipotesi 1). Dai dati è possibile notare come il FtD risulti generalmente abbastanza elevato, tanto che non risulta mai inferiore alla soglia prestabilita di per la notifica (ipotesi 2), questo implica che sfruttando una formula priva di personalizzazione il guidatore non sarebbe mai notificato di un comportamento errato e pertanto gli istanti in cui commette un'infrazione rimarrebbero

invariati, non andando a migliorare la sicurezza stradale.



(a) FtD tenuto durante la simulazione di guida.



(b) FtD tenuto durante la simulazione di guida con e senza notifica.

Figura 4.2: FtD calcolato con la formula precedente

Nella figura 4.2b viene mostrato come cambia il valore di FtD in presenza e assenza di notifica del valore dell'indice; come si può osservare i FtD rilevati risultano completamente identici, sia con che senza notifica. Questo è dovuto al fatto che avendo un comportamento lineare e non personalizzato, il FtD non risulta mai inferiore alla soglia definita in precedenza.

Valutazione prestazioni del FtD versione v1

Per l'adozione della nuova formula del FtD, quindi con personalizzazione, verranno considerate quattro casistiche:

1. guidatore di 22 anni con esperienza, ma frequenza di guida bassa;

2. guidatore di 22 anni con poca esperienza e frequenza di guida media;
3. guidatore di 45 anni con esperienza, ma frequenza di guida bassa;
4. guidatore di 45 anni con poca esperienza e frequenza di guida media;

Nei paragrafi successivi verranno descritti più approfonditamente i parametri utilizzati per il test e verrà fornita una considerazione su quanto emerso.

Guidatore di 22 anni con esperienza Il test viene effettuato su un uomo adulto con le seguenti caratteristiche:

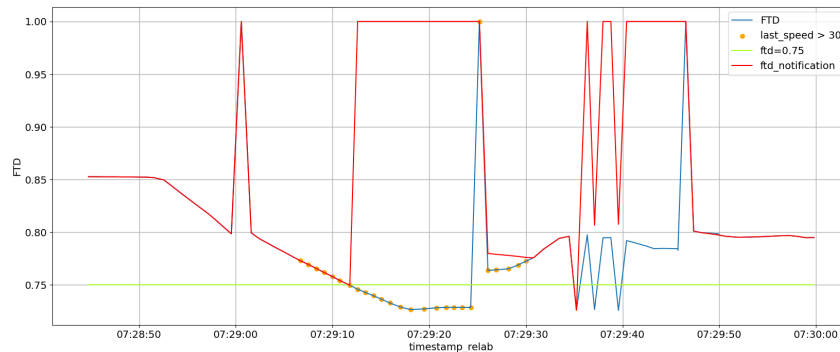
- nato il 2000-09-10 (22 anni);
- patente di guida conseguita il 2020-01-05 (3 anni);
- chilometri percorsi nell'ultimo anno pari a 0;

Nel grafico mostrato in figura 4.3a è possibile osservare un valore di FtD relativamente basso, ciò è dovuto soprattutto alla giovane età del guidatore, che fa sì che venga considerato più propenso a commettere degli errori durante la guida. Sempre dal grafico è inoltre possibile notare come il superamento del limite di velocità sia prevalentemente in corrispondenza di un valore di FtD basso; ciò suggerisce che l'aumento della velocità e il conseguente superamento dei limiti siano da ricondurre ad un comportamento non idoneo.

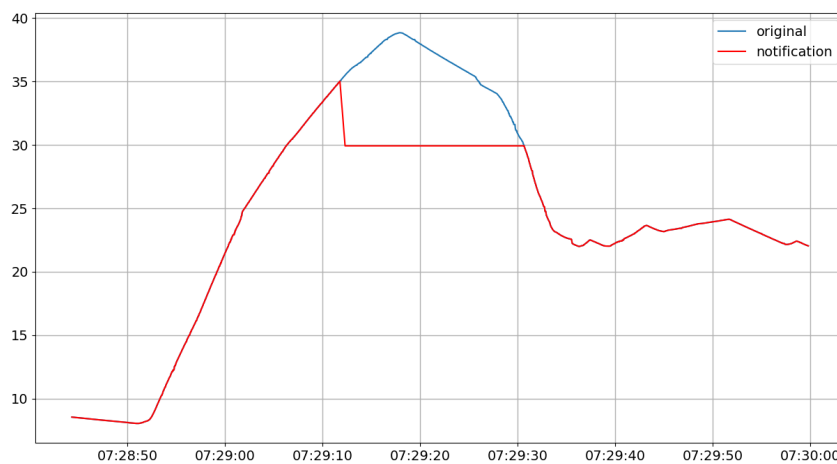
Attraverso la notifica di un valore di FtD al di sotto della soglia, figura 4.3a e 4.3b, è possibile ridurre gli istanti in cui il guidatore risulta non idoneo e conseguentemente vengono ridotti gli istanti in cui viene commessa un'infrazione. Questo fa sì che sia possibile andare ad incrementare la sicurezza sia per il guidatore che per tutti i soggetti terzi, come pedoni, passeggeri o altri conducenti.

Guidatore di 22 anni con poca esperienza Il test viene effettuato su un uomo adulto con le seguenti caratteristiche:

- nato il 2000-09-10 (22 anni);



(a) FtD tenuto durante la simulazione di guida con e senza notifica.



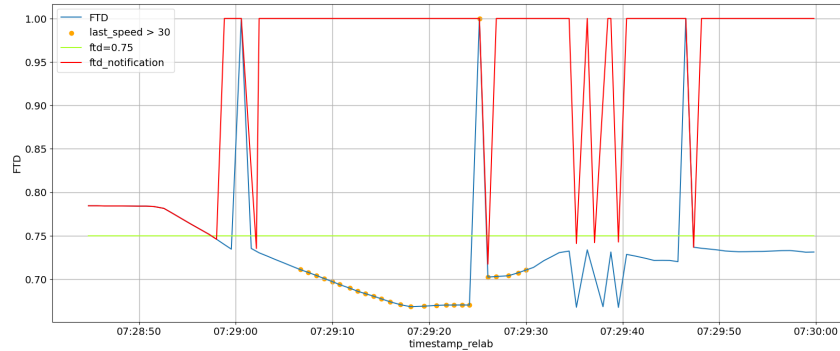
(b) Velocità tenuta durante la simulazione con la notifica di FtD.

Figura 4.3: Grafici della simulazione con un guidatore di 22 anni con esperienza, ma frequenza di guida bassa.

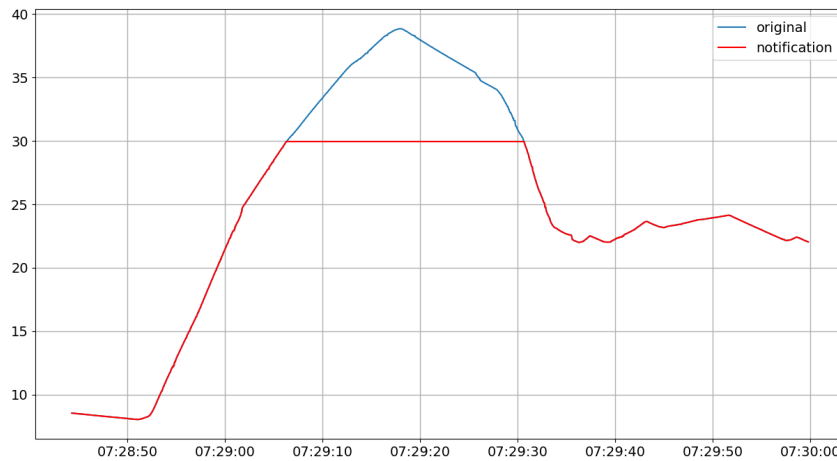
- patente di guida conseguita il 2021-01-05 (2 anni);
- chilometri percorsi nell'ultimo anno pari a 4500;

Nel grafico mostrato in figura 4.4a è possibile osservare un valore di FtD relativamente basso, ciò è dovuto soprattutto alla giovane età del guidatore e alla poca esperienza nel mondo automobilistico; ciò fa sì che venga considerato più propenso a commettere degli errori durante la guida. Sempre dal grafico è inoltre possibile notare come il superamento del limite di velocità sia prevalentemente in corrispondenza di un valore di FtD basso. Ciò suggerisce

che l'aumento della velocità e il conseguente superamento dei limiti siano da ricondurre ad un comportamento non idoneo.



(a) FtD tenuto durante la simulazione di guida con e senza notifica.



(b) Velocità tenuta durante la simulazione con la notifica di FtD.

Figura 4.4: Grafici della simulazione con un guidatore di 22 anni con poca esperienza, ma frequenza di guida media.

Attraverso la notifica di un valore di FtD al di sotto della soglia, figura 4.4a e 4.4b, è possibile ridurre notevolmente gli istanti in cui il guidatore risulta non idoneo e conseguentemente vengono ridotti gli istanti in cui viene commessa un'infrazione. Questo fa sì che sia possibile, anche in questo caso, andare ad incrementare la sicurezza sia per il guidatore che per tutti i soggetti terzi.

Guidatore di 45 anni con esperienza Il test viene effettuato su un uomo adulto con le seguenti caratteristiche:

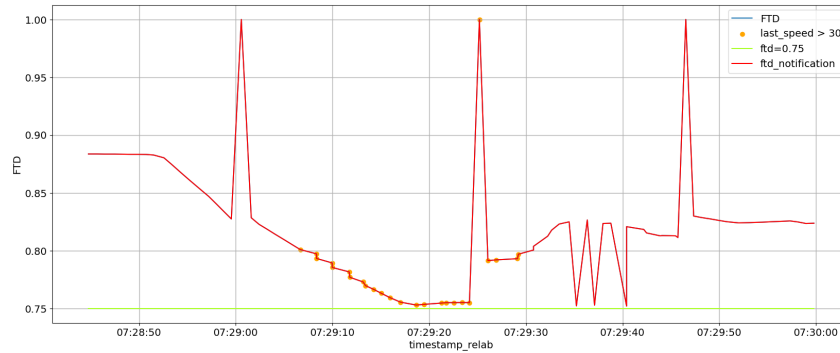
- nato il 1978-09-10 (45 anni);
- patente di guida conseguita il 2016-01-05 (8 anni);
- chilometri percorsi nell'ultimo anno pari a 0;

Nel grafico mostrato in figura 4.5a è possibile osservare un valore di FtD piuttosto elevato se confrontato con i precedenti, ciò è dovuto soprattutto al fatto che il guidatore risulta avere un'età il cui valore si trovi vicino a quella ritenuta ideale per la guida, ossia per la quale si ritiene che il guidatore sia più cauto e attento. Inoltre, il conducente risulta avere un'esperienza tale, secondo gli studi condotti, da avere dei tempi di reazione ad eventuali pericoli molto bassi. Sempre dal grafico è, per di più, possibile notare come il superamento del limite di velocità sia prevalentemente in corrispondenza di un valore di FtD più basso rispetto a quello tenuto negli istanti precedenti, ciò suggerisce che l'aumento della velocità e il conseguente superamento dei limiti siano da ricondurre ad un comportamento non idoneo.

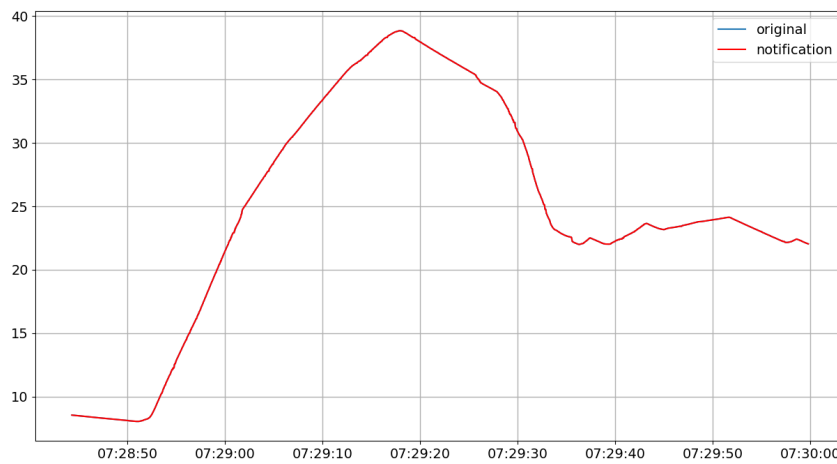
Come si può notare nelle figure 4.5a e 4.5b, in questo caso la notifica non viene mai registrata, in quanto il valore di FtD, sebbene venga commessa un'infrazione, non risulta mai al di sotto della soglia prestabilita. Questo è dovuto al fatto che, come detto precedentemente, il guidatore ha un'età e un'esperienza tali che gli viene permesso di commettere dei comportamenti non pienamente idonei al contesto di guida, in quanto risulta più propenso ad una guida attenta e a dei tempi di reazione agli errori molto bassi.

Guidatore di 45 anni con poca esperienza Il test viene effettuato su un uomo adulto con le seguenti caratteristiche:

- nato il 1978-09-10 (45 anni);
- patente di guida conseguita il 2020-01-05 (2 anni);
- chilometri percorsi nell'ultimo anno pari a 4500;



(a) FtD tenuto durante la simulazione di guida con e senza notifica.



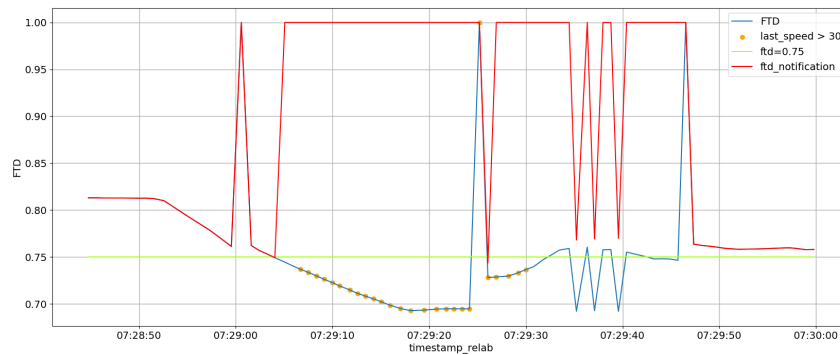
(b) Velocità tenuta durante la simulazione con la notifica di FtD.

Figura 4.5: Grafici della simulazione con un guidatore di 45 anni con esperienza, ma frequenza di guida bassa.

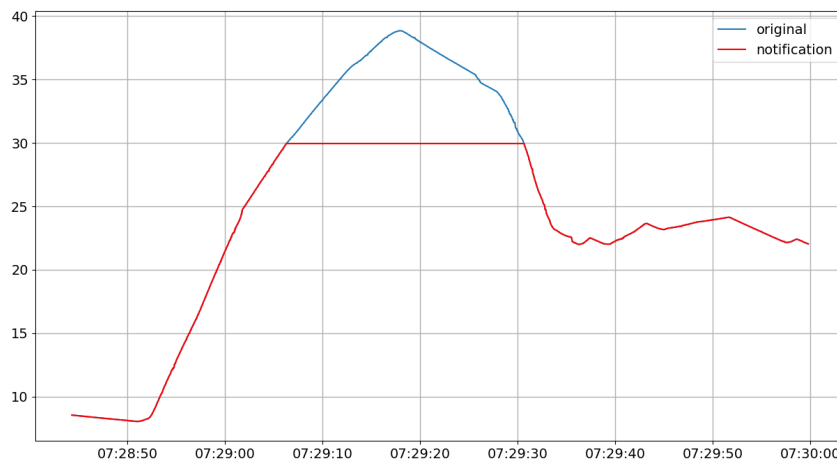
Nel grafico mostrato in figura 4.6a è possibile osservare un valore di FtD relativamente basso, ciò è dovuto soprattutto alla poca esperienza del guidatore nel mondo automobilistico, che fa sì che venga considerato più propenso a commettere degli errori durante la guida, sebbene la sua età risulti vicina a quella ottimale. Sempre dal grafico è inoltre possibile notare come il superamento del limite di velocità sia prevalentemente in corrispondenza di un valore di FtD basso, ciò suggerisce che l'aumento della velocità e il conseguente superamento dei limiti siano da ricondurre ad un comportamento non idoneo.

Attraverso la notifica di un valore di FtD al di sotto della soglia, figura 4.6a

e 4.6b, è possibile ridurre notevolmente gli istanti in cui il guidatore risulta non idoneo e conseguentemente vengono ridotti gli istanti in cui viene commessa un'infrazione. Questo fa sì che sia possibile, anche in questo caso, andare ad incrementare la sicurezza sia per il guidatore che per tutti i soggetti terzi.



(a) FtD tenuto durante la simulazione di guida con e senza notifica.



(b) Velocità tenuta durante la simulazione con la notifica di FtD.

Figura 4.6: Grafici della simulazione con un guidatore di 45 anni con poca esperienza, ma frequenza di guida media.

4.1.3 Considerazioni

Nel grafico mostrato nella figura 4.7 vengono messi a confronto i valori di FtD in assenza di notifica, rilevati nei test descritti nei paragrafi precedenti.

Come ci si aspetta il valore di FtD varia in relazione ai fattori personali che caratterizzano il guidatore, in particolare la sua età e la frequenza ed esperienza di guida. Ciò è dovuto al fatto che guidatori giovani e anziani o con una frequenza e/o esperienza bassa sono più propensi a commettere errori che possono risultare in una considerevole diminuzione della sicurezza stradale; viceversa persone la cui età è vicina a quella ottimale o che risultano avere una elevata esperienza o frequenza di guida risulteranno avere un valore di FtD che viene penalizzato in modo minore. Per quanto riguarda invece il confronto tra il FtD con e senza personalizzazione, si può osservare che generalmente quello senza personalizzazione, e quindi sviluppato con la formula precedente, risulta essere molto più elevato, ciò fa sì che non si riesca ad andare a correggere prontamente i comportamenti che fanno sì che l' idoneità del guidatore diminuisca.

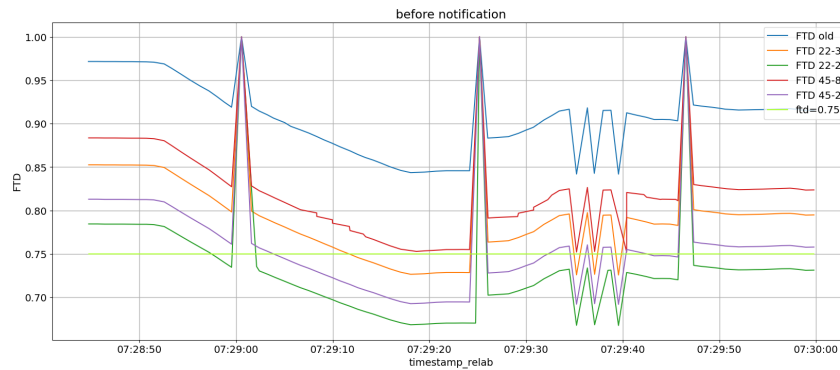


Figura 4.7: Comparazione FtD senza notifica.

Nel grafico mostrato nella figura 4.8, viene invece mostrato, come l'utilizzo del FtD in combinazione di una notifica possa far aumentare la sicurezza stradale a seguito di una maggiore idoneità del guidatore. Anche in questo caso, come per il precedente grafico è possibile osservare alcune differenze del valore di FtD rilevato in funzione dei fattori di personalizzazione, il che fa sì che persone con caratteristiche differenti vengano notificate in istanti temporali diversi; ad esempio, a parità di età, si osservino i valori relativi al guidatore di 22 anni: una persona inesperta verrà notificata prima rispetto ad un'altra la cui esperienza risulta essere maggiore.

I test appena descritti sono stati svolti sulla base dei dati registrati durante le simulazioni e facendo variare questi sulla base delle diverse caratteristiche che può assumere un guidatore; si rimanda a degli studi futuri una validazione più completa consistente di test effettuati con diversi guidatori sia in modalità *on-road* che *off-road*.

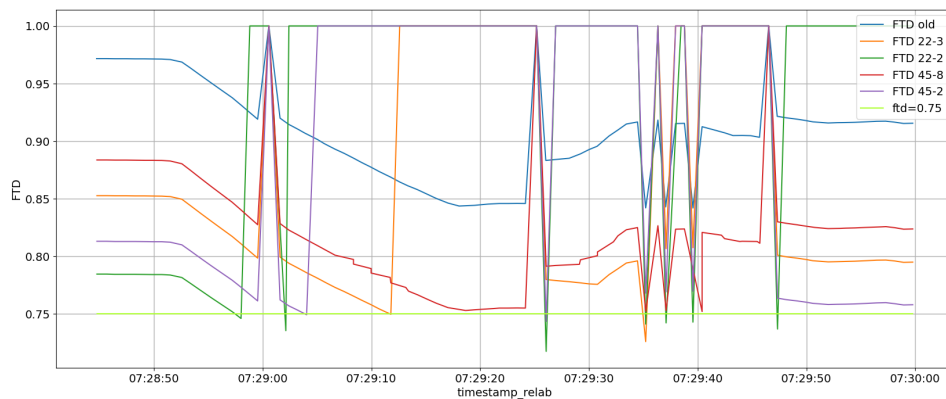


Figura 4.8: Comparazione FtD con notifica.

4.2 Valutazione della latenza

La latenza è un parametro critico nella valutazione delle prestazioni di un sistema. Rappresenta il ritardo temporale tra l'inizio di un'azione e la sua risposta effettiva, e può influenzare significativamente l'esperienza dell'utente, la qualità del servizio e l'efficacia complessiva del sistema. Nella seguente sezione verrà presa in esame la latenza che caratterizza il sistema sviluppato, e verrà messa a confronto con quella del sistema prodotto nell'ambito del progetto NextPerception.

4.2.1 Descrizione del test

Il test sulla latenza che è stato eseguito prevede due esecuzioni: una usufruendo del modulo per il calcolo del FtD sviluppato nell'ambito del progetto NextPerception e una mediante l'esecuzione del sistema oggetto del seguente

progetto di tesi. Entrambe le esecuzioni avvengono in un ambiente creato appositamente attraverso l'utilizzo di Docker ¹. Per la prima esecuzione, ossia quella che usufruisce del modulo sviluppato per il progetto NextPerception, è stata impiegata una sola immagine, chiamata NP, la quale racchiude il codice relativo al modulo di calcolo sviluppato. Per testare il sistema oggetto del seguente progetto di tesi sono state invece create due immagini: `ftdms-fdt_ms`, contenente il codice relativo al microservizio incaricato del calcolo del FtD, e `humandigitaltwin-hdt`, la quale racchiude il codice relativo allo HDT sviluppato. In particolare le due immagini realizzate permettono di definire i rispettivi container così caratterizzati:

- container `humandigitaltwin`, il quale si compone di tre servizi tra loro connessi attraverso una rete interna:
 - `hdt`, che rappresenta lo HDT realizzato;
 - `hdt_rabbitmq`, che rappresenta il servizio associato all'immagine `rabbitMQ`, necessario per la comunicazione con l'autoveicolo;
 - `hdt_mongodb`, che rappresenta il servizio associato all'immagine `mongodb`, necessario per lo storage dei dati.
- container `ftdms`, il quale si compone di tre servizi tra loro connessi attraverso una rete interna:
 - `ftd_ms`, che rappresenta il microservizio realizzato per il calcolo del FtD;
 - `ftd_redis`, che rappresenta il servizio associato all'immagine `redis`, necessario per lo storage locale dei dati, ossia l'in-memory database;
 - `ftd_ms_mongodb`, che rappresenta il servizio associato all'immagine `mongodb`, necessario per lo storage dei dati.

La latenza delle due esecuzioni sono state calcolate tenendo in considerazione il tempo intercorso tra l'invio dell'ultimo dato utile per il calcolo del FtD,

¹Docker è una piattaforma software open source utilizzata per creare, distribuire e gestire contenitori di applicazioni virtualizzati su un sistema operativo (OS) comune

rappresentato dalla distrazione cognitiva, in quanto, secondo le specifiche del sistema, è il dato la cui frequenza di elaborazione risulta essere più bassa, e la ricezione del valore di FtD da parte del veicolo. Questo permette di tenere in considerazione oltre che al tempo di elaborazione del risultato, anche dei tempi necessari alla comunicazione, ossia dei tempi per lo scambio di messaggi MQTT e/o AMPQ presenti sia tra veicolo e sistema che tra i servizi impiegati internamente dal sistema.

4.2.2 Risultati e considerazioni

Nella tabella 4.1 sono riportate le latenze medie registrate nei due scenari illustrati precedentemente. In particolare, il primo valore è riferito alla latenza rilevata durante l'utilizzo del modulo FtD sviluppato nell'elaborato di tesi [31, 7], quindi in assenza di personalizzazione dell'indice, mentre il secondo è riferito alla latenza registrata durante l'utilizzo del modulo oggetto del seguente studio di tesi e pertanto con personalizzazione.

Come si può notare dai dati la latenza nel secondo caso risulta essere maggiore rispetto al primo caso; le latenze registrate tra i due sistemi riflettono la differenza fondamentale tra un'architettura di base e una più complessa. Nel primo sistema, ossia nel caso del modulo di FtD senza personalizzazione, tutti i calcoli vengono eseguiti localmente al modulo in oggetto. Ciò implica che la latenza è strettamente legata alla latenza dovuta alla comunicazione attraverso il protocollo MQTT. Utilizzando il nuovo sistema sviluppato, al contrario, si aggiunge un'ulteriore complessità, che fa sì che i tempi necessari alla computazione siano leggermente maggiori. Infatti, al fine di poter calcolare il FtD in modo personalizzato si è scelto di adottare uno HDT, la cui architettura risulta essere più complessa, tanto che il modulo di stereotipo `ftd` per il calcolo del FtD sfrutta un microservizio esterno. Perciò la maggiore latenza individuata durante l'utilizzo del FtD con personalizzazione, può essere attribuita all'interazione tra componenti distribuiti, incluso l'uso congiunto di MQTT e RabbitMQ.

Tipo Ftd	latenza media in secondi
FtD non personalizzato	0.17850
FtD personalizzato	0.63210

Tabella 4.1: Latenze registrate nei test con e senza personalizzazione.

In conclusione, nonostante il sistema comprendente la personalizzazione presenti una leggera latenza maggiore rispetto al primo, la scelta di adottare questa architettura più complessa risulta ancora accettabile. Il tempo di latenza, seppur leggermente più elevato (0.63210 secondi), si mantiene comunque a un livello relativamente basso, garantendo un compromesso ottimale tra la precisione dei calcoli e l'efficacia dell'architettura. La maggiore scalabilità, flessibilità, modularità e l'isolamento dei guasti, offerte dal secondo sistema, grazie all'utilizzo di microservizi e al supporto combinato di MQTT e RabbitMQ, giustificano appieno la piccola riduzione dell'efficienza, offrendo una soluzione che soddisfa le esigenze del progetto senza compromettere in modo significativo le prestazioni complessive.

4.2.3 Latenza di set-up

Un ulteriore fattore da considerare durante la valutazione del sistema realizzato è la latenza di set-up, ossia di avvio e configurazione. Al fine di esaminare tale valore è stato scelto di dockerizzare il sistema, come descritto nella precedente sezione 4.2.1, e tenere in considerazione la latenza relativa all'avvio e configurazione dello HDT, l'installazione di un modulo - nel caso in esame rappresentato dal `ftd` - la sua configurazione e il suo avvio. In tabella 4.2 è possibile osservare le tre latenze registrate. In particolare, la prima si riferisce al caso sia necessario scaricare ed installare tutte le immagini docker di cui necessita il servizio. La seconda al caso non sia necessario installare le immagini docker di supporto. Mentre il terzo ed ultimo valore considera la latenza nel caso in cui il sistema sia già preinstallato, per cui comprende solo l'avvio dello HDT e la sua configurazione, secondo quanto descritto in precedenza.

Tipo Ftd	latenza
Senza immagini docker	1m 32s
Con immagini docker	1m 9s
Preinstallato	30s

Tabella 4.2: Latenze registrate per l'avvio e configurazione dello HDT e del modulo `ftd`.

Come si può notare le latenze rilevate evidenziano una notevole efficienza. In particolare, il tempo di avvio dimezza qualora lo HDT sia già presente sul dispositivo. Questi risultati sono giustificati dalla relativa semplicità di avvio e configurazione del sistema, nonostante la sua complessità interna. Pertanto, è essenziale sottolineare che questa latenza è attualmente considerata accettabile per le necessità del nostro sistema. Tale efficienza, infatti, ci consente di raggiungere gli obiettivi di avvio e configurazione senza compromettere le prestazioni complessive.

4.3 Utilizzo delle risorse

La valutazione dell'utilizzo delle risorse durante l'esecuzione del sistema rappresenta un aspetto fondamentale per comprendere le prestazioni, la stabilità e l'efficienza del sistema realizzato. Infatti, attraverso tale monitoraggio è possibile individuare eventuali punti di sovraccarico o di inefficiente utilizzo delle risorse.

Di seguito verranno in particolare analizzati l'utilizzo della CPU e della memoria RAM da parte dello HDT e da parte del microservizio incaricato del calcolo del FtD. Come per i precedenti test relativi alle latenze è stato scelto di dockerizzare il sistema, come descritto nella sezione 4.2.1, e di valutare l'utilizzo della CPU e della RAM rispetto ai container realizzati.

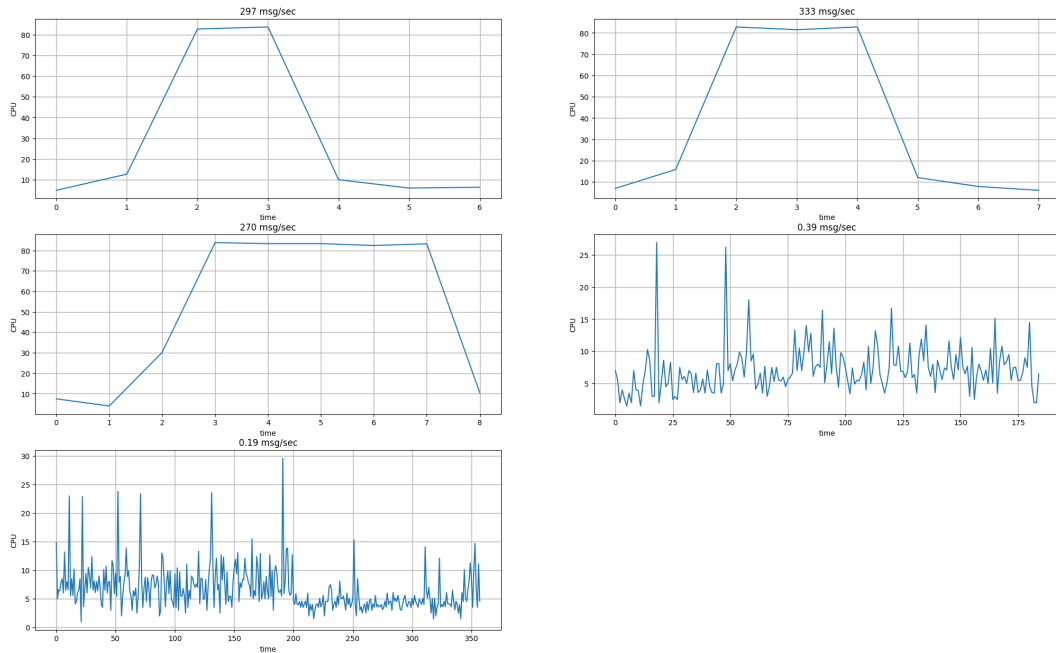


Figura 4.9: Utilizzo della CPU dello HDT a diverse frequenze di invio dei dati.

4.3.1 HDT

La figura 4.9 illustra le variazioni nell'utilizzo della CPU in funzione delle diverse frequenze con la quale vengono inviati i dati necessari al calcolo del FtD. Come si può osservare, l'utilizzo della CPU tende a stabilizzarsi intorno al 5-10%. Questo fenomeno è da attribuire al fatto che lo HDT realizzato ha un carico di lavoro costante dovuto al fatto che si tratta di un'applicazione che lavora in *real-time* o *simil real-time*. In generale, è possibile notare che il sistema risponde in modo adeguato alla richiesta delle risorse. Ad esempio, quando la frequenza è più elevata, e quindi sono necessarie più risorse, l'utilizzo di CPU aumenta fino ad un 83% di utilizzo, il quale rimane comunque al di sotto del limite di saturazione. Al contrario, quando la frequenza di invio è bassa, l'utilizzo della CPU rimane sostanzialmente invariato rispetto alla situazione in cui i dati non vengono inviati (5-10%), con l'eccezione di picchi in cui l'elaborazione richiede temporaneamente più risorse.

Analizzando l'utilizzo della RAM, è possibile notare (come mostrato nella

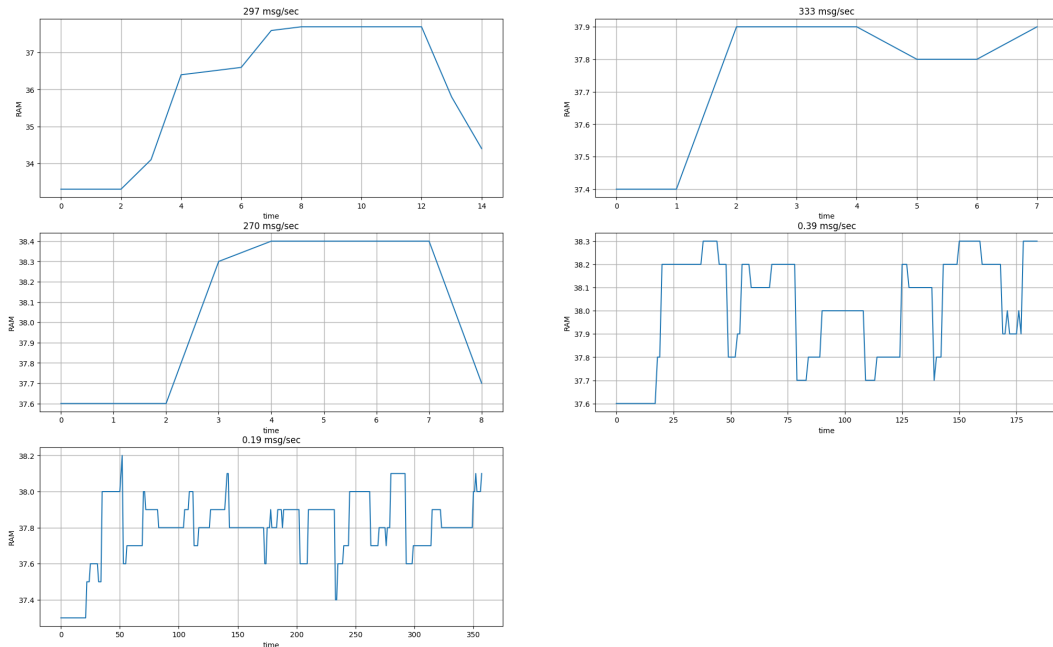


Figura 4.10: Utilizzo della RAM dello HDT a diverse frequenze di invio dei dati.

figura 4.10) che il suo valore percentuale non risulta eccessivamente elevato; tende piuttosto a stabilizzarsi intorno al 36-37%, con picchi occasionali fino al 38%. Questo fenomeno è da attribuire al fatto che il sistema realizzato svolge per lo più compiti CPU-bound, il che significa che la CPU è il componente chiave coinvolto nell'elaborazione delle attività del sistema, mentre l'utilizzo della RAM rimane relativamente stabile. Infatti, se confrontati quando vi è una frequenza maggiore di invio dei messaggi, è possibile notare come una frequenza maggiore faccia aumentare l'utilizzo di CPU, mentre l'utilizzo di RAM rimane pressochè invariato.

Valori bassi di utilizzo della RAM rappresentano un indicatore positivo di un sistema informatico ben ottimizzato e in salute. Tale situazione benefica, infatti, indica che il sistema ha ancora una buona quantità di memoria disponibile per gestire picchi improvvisi di utilizzo.

4.3.2 Microservizio FtD

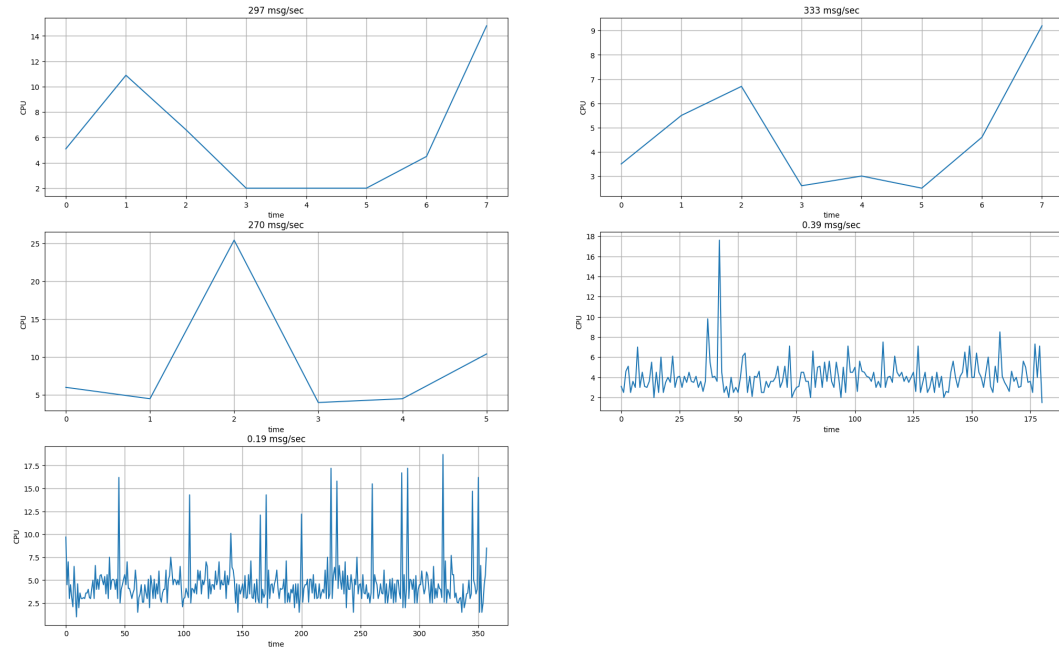


Figura 4.11: Utilizzo della CPU del microservizio a diverse frequenze di invio dei dati.

La figura 4.11 illustra le variazioni nell'utilizzo della CPU in funzione delle diverse frequenze con la quale vengono inviati i dati necessari al calcolo dell'indice di idoneità. Come si può osservare, l'utilizzo della CPU tende a stabilizzarsi intorno al 2-10%. Questo fenomeno è da attribuire al fatto che il microservizio realizzato ha un carico di lavoro costante dovuto al fatto che si tratta di un'applicazione che lavora in *real-time* o *simil real-time*. In generale, è possibile notare che il sistema risponde in modo adeguato alla richiesta delle risorse. Ad esempio, quando la frequenza è più elevata, e quindi sono necessarie più risorse, l'utilizzo di CPU aumenta fino ad un 25% di utilizzo, il quale rimane comunque ben al di sotto del limite di saturazione. Al contrario, quando la frequenza di invio è bassa, l'utilizzo della CPU rimane sostanzialmente invariato rispetto alla situazione in cui i dati non vengono inviati (2-10%), con l'eccezione di picchi in cui l'elaborazione richiede temporaneamente più risorse.

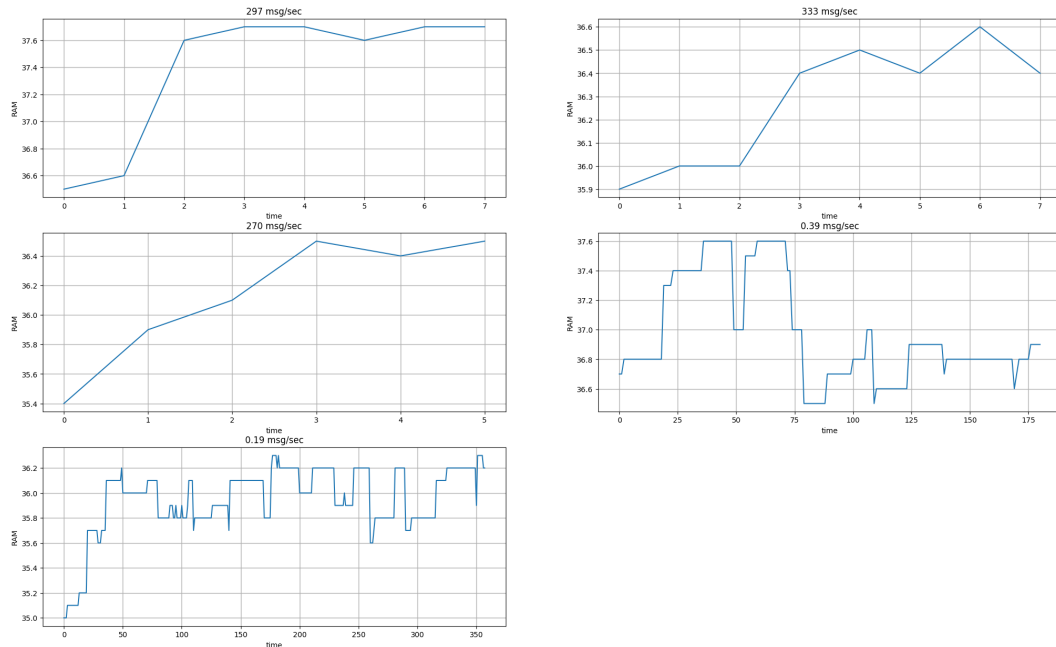


Figura 4.12: Utilizzo della RAM del microservizio a diverse frequenze di invio dei dati.

Analizzando l'utilizzo della RAM, è possibile notare (come mostrato nella figura 4.12) che il suo valore percentuale non risulta eccessivamente elevato; tende piuttosto a stabilizzarsi intorno al 36-37%, con picchi occasionali fino al 38%. Come si può osservare, il microservizio ha valori di utilizzo della RAM molto maggiori rispetto a quelli della CPU. Questo è dovuto al fatto che durante la progettazione ed implementazione è stato deciso di adottare un database di tipo *in-memory*, al fine di aumentare le prestazioni delle operazioni di lettura e scrittura dei dati, riducendo i tempi di accesso ai dati e migliorando la reattività delle applicazioni. Nonostante questo divario, i valori registrati per l'utilizzo RAM risultano comunque bassi. Ciò, come nel caso precedente, rappresenta un indicatore positivo di un sistema informatico ben ottimizzato e in salute. Tale situazione benefica, infatti, indica che il sistema ha ancora una buona quantità di memoria disponibile per gestire picchi improvvisi di utilizzo.

Conclusioni

Il presente elaborato di tesi si è posto due obiettivi principali: individuare una formula per il calcolo dell'idoneità di guida che fosse personalizzata sulla base delle caratteristiche del guidatore, e la definizione e realizzazione di uno Human Digital Twin, che consideri il profilo dell'utente rappresentato, consentendo di sfruttarlo attraverso stereotipi classificati per personalizzare le applicazioni e i dispositivi con cui l'utente interagisce.

In particolare sono stati svolti degli studi preliminari al fine di individuare e analizzare i parametri che consentissero di personalizzare il FtD e di definire il concetto di Human Digital Twin. A tal proposito è emerso che l'età, la frequenza di guida e l'esperienza rappresentano dei parametri fondamentali da tenere in considerazione quando si parla di idoneità di guida. Questo in quanto conducenti di giovane età o anziani risultano essere più propensi a commettere errori, mentre guidatori e con una bassa frequenza di guida o poca esperienza risultano avere un tempo di reazione maggiore al verificarsi di possibili situazioni di pericolo. Per quanto riguarda la definizione dello HDT sono emersi alcuni aspetti fondamentali per la sua definizione. In particolare, in tale ambito la definizione dello *User context* e dello *User profile*, sono stati ritenuti elementi chiave. Il primo in quanto risulta indispensabile per la gestione di tutte le applicazioni che considerano la posizione e più in generale il contesto in cui è presente una persona per poter operare; mentre il secondo fornisce informazioni importanti sulle caratteristiche del soggetto, permettendo così la personalizzazione delle applicazioni, tra cui quelle che utilizzano il contesto appena citato.

Alla luce di quanto emerso ci siamo spostati sulla definizione e sullo studio

di una nuova formula per il calcolo personalizzato dell'indice di idoneità di guida e sulla definizione del design e architettura dello HDT.

La formula in questione tiene conto, come nel caso della prima versione senza personalizzazione realizzata per il progetto NextPerception, delle distrazioni, visuali e cognitive, delle emozioni e del rispettivo valore di attivazione, ossia l'arousal; a questo è stato aggiunto il contributo del peso derivante dalla valutazione dell'età del guidatore, modellato attraverso una funzione gaussiana invertita, e al peso derivante dalla valutazione della frequenza di guida e dell'esperienza del guidatore.

Lo HDT ideato ha permesso di esplorare un nuovo concetto di digital twin che tenesse conto dello *User context* e dello *User profile* attraverso la definizione di classi e moduli di stereotipo. In particolare, le classi hanno permesso di rappresentare le diverse caratteristiche dell'utente, mentre i moduli di stereotipo le condizioni di attivazione che permettono di identificare gli utenti in un particolare dominio applicativo, ossia nel contesto.

Definita la nuova formula e realizzato il sistema si è provveduto ad effettuare una valutazione di quanto prodotto attraverso il confronto con il sistema implementato per il progetto NextPerception. Sono stati quindi valutate le metriche di precisione del calcolo, la latenza e l'utilizzo delle risorse. Alla luce di quanto emerso da tali valutazioni si può considerare il sistema realizzato, sebbene leggermente più lento, migliore del precedente, in quanto riesce, a fronte di opportune interazioni con il guidatore, ad andare a correggere prontamente i comportamenti, quali la presenza di distrazioni o emozioni eccessivamente positive o negative, che fanno sì che l'idoneità del guidatore diminuisca.

Nonostante i risultati positivi ottenuti con il sistema realizzato, è importante evidenziare alcuni limiti e problematiche che emergono dall'analisi. In particolare, per quanto riguarda i test eseguiti relativamente alla validazione della formula per il FtD, si ritiene importante precisare che questi sono stati effettuati, sebbene a partire dai dati provenienti da test reali, attraverso un ambiente virtuale, apportando modifiche ai dati senza quindi coinvolgere un simulatore di guida o utenti fisici. Per quanto riguarda invece l'adozione di uno HDT o più in generale di un digital twin come controparte virtuale di un sog-

getto umano, questo come detto solleva importanti considerazioni, soprattutto relative alla privacy, sicurezza e a questioni etiche e di consenso. Dal punto di vista della privacy della sicurezza, è importante considerare che la creazione e l'uso dei DT richiedono la raccolta e l'archiviazione di una vasta quantità di dati personali e sensibili, che se non gestiti in modo adeguato presentano il rischio che possano essere usate impropriamente o violate, ledendo la riservatezza dell'individuo. Dal punto di vista etico e di consenso, invece, bisogna considerare che la profilazione avanzata e le previsioni del comportamento possono limitare l'autonomia e la libertà di scelta della persona modellata. Inoltre, basando la modellazione del DT sui dati storici si solleva il rischio di promuovere e consolidare pregiudizi e stereotipi penalizzando individui con comportamenti non standard andando a dare origine a decisioni discriminatorie.

A fronte di ciò, in futuro, a seguito di maggiori test da effettuare sia on-road che off-road con utenti fisici, sarà possibile perfezionare ulteriormente la formula per il calcolo dell'indice di idoneità di guida denominato "Fitness-to-Drive". Inoltre, grazie all'utilizzo del linguaggio Python, sarà possibile definire gli algoritmi di machine learning atti a realizzare il profilo dinamico. Il cui scopo consiste nell'aggiornare le classi individuate nel profilo statico, tenendo conto, per ogni contesto della classe, sia dei dati raccolti dai moduli che sfruttano tale classe che i dati provenienti da altre fonti. Così facendo sarà possibile riuscire ad imparare dai dati elaborati in passato al fine di migliorare e calibrare il peso della classe nel contesto. Infine, grazie all'architettura ideata per lo HDT sarà possibile, implementando il modulo di autenticazione tra digital twin creare una social digital twin network, al fine di promuovere la collaborazione e alla condivisione di informazioni tra sistemi di digital twin. Utilizzando la definizione dello HDT e la libreria degli stereotipi, sarà anche possibile aggiungere nuovi moduli che, mediante l'impiego delle informazioni personali dell'utente associato allo HDT, favoriscano una collaborazione tra uomo e macchina al fine di migliorare lo stile di vita umano in termini di sicurezza, salute e benessere.

Ringraziamenti

Vorrei dedicare questo spazio a chi, con dedizione e pazienza, ha contribuito alla realizzazione di questo progetto.

Un ringraziamento particolare va al mio relatore Roberto Girau che mi ha offerto l'opportunità di contribuire a questo progetto, molto interessante e stimolante sia per l'ambito di applicazione che per le tecnologie utilizzate, e mi ha seguita, con la sua infinita disponibilità, in ogni step della realizzazione del progetto.

Non posso non menzionare la mia famiglia che da sempre mi sostengono nella realizzazione dei miei progetti, appoggiando le mie scelte e incoraggiandomi a dare sempre il massimo. Grazie per il vostro amore, sostegno e la costante fiducia in me. Il vostro appoggio incondizionato è stato il motore che mi ha spinto avanti in ogni fase di questo percorso.

Grazie a tutti i miei colleghi di corso, per avermi sempre incoraggiato fin dall'inizio del percorso universitario e supportato nei momenti in cui credevo di non farcela. Con loro ho condiviso momenti di crescita professionale e personale. Le nostre discussioni e la collaborazione sono stati fonte di ispirazione e arricchimento.

Un ringraziamento sentito va ai miei amici, quelli di lunga data e quelli che ho incontrato lungo il cammino. Le risate condivise, le chiacchierate motivanti e i momenti di svago sono stati una vera linfa per affrontare le sfide che ho incontrato lungo il percorso.

In conclusione, voglio esprimere la mia profonda gratitudine a tutti voi che avete contribuito, in modo grande o piccolo, a questo progetto. Questa tesi rappresenta un traguardo importante, ma è solo una tappa nel mio percorso

di crescita. Grazie per aver fatto parte di questo viaggio e per averlo reso significativo.

Bibliografia

- [1] Ahmed Al-Nazer, Tarek Helmy, and Mohammed Al-Mulhem. User's profile ontology-based semantic framework for personalized food and nutrition recommendation. *Procedia Computer Science*, 32:101–108, 2014. The 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014), the 4th International Conference on Sustainable Energy Information Technology (SEIT-2014).
- [2] Cristina Alcaraz and Javier Lopez. Digital twin: A comprehensive survey of security threats. *IEEE Communications Surveys & Tutorials*, 24(3):1475–1503, 2022.
- [3] R. Alfonsi, A. Ammari, and D. S. Usami. Lack of driving experience. European Road Safety Decision Support System, developed by the H2020 project SafetyCube, 2018.
- [4] H. Almahasneh. Deep in thought while driving: An eeg study on drivers' cognitive distraction. <https://doi.org/10.1016/j.trf.2014.08.001>, 2014.
- [5] Giuseppe Amato and Umberto Straccia. User profile modeling and applications to digital libraries. In Serge Abiteboul and Anne-Marie Vercoustre, editors, *Research and Advanced Technology for Digital Libraries*, pages 184–197, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.
- [6] Dr. Manuel Andruccioli, Dr. Maria Mengozzi, Dr. Roberta Presta, Prof. Silvia Mirri, and Dr. Roberto Girau. Arousal effects on fitness-

- to-drive assessment: algorithms and experiments. In *Atti della conferenza CCNC 2023*.
- [7] Manuel Andruccioli. *Effetti dell'arousal sul fitness to drive: algoritmi e simulazione*. PhD thesis.
- [8] Barbara Rita Barricelli, Elena Casiraghi, and Daniela Fogli. A survey on digital twin: Definitions, characteristics, applications, and design implications. *IEEE Access*, 7:167653–167671, 2019.
- [9] Wiebo Brouwer and Frederic Withaar. Fitness to drive after traumatic brain injury. *Neuropsychological Rehabilitation*, July 1:177–193, 10 2010.
- [10] Ximing Chen, Eunsuk Kang, Shinichi Shiraishi, Victor Preciado, and Jiang Zhihao. Digital behavioral twins for safe connected cars. pages 144–153, 10 2018.
- [11] Zhiqing Cheng. Human digital twin with applications. *Proceedings of the 7th International Digital Human Modeling Symposium (DHM 2022) and Iowa Virtual Human Summit 2022 -*, 2022.
- [12] Ministro delle infrastrutture e dei trasporti. Art. 126. durata e conferma della validità della patente di guida. *Codice della strada/TITOLO IV - GUIDA DEI VEICOLI E CONDUZIONE DEGLI ANIMALI*, Anno.
- [13] Vitaliy Dronseyko, Aleksandra Pakhomova, Elena Shalagina, and Maxim Pletnev. Driving danger coefficient as a method of evaluating the driver's behavior in road traffic. *Transportation Research Procedia*, 36:129–134, 2018. System and digital technologies for ensuring traffic safety.
- [14] P. EKMAN. Universal facial expressions in emotion. *Studia Psychologica*, 15(2):140, 1973. Ultimo aggiornamento - 2013-02-24.
- [15] Victoria Eyharabide and Analía Amandi. Ontology-based user profile learning. *Applied Intelligence - APIN*, 36, 06 2012.

-
- [16] Danish Farooq and Janos Juhasz. Statistical evaluation of risky driver behavior factors that influence road safety based on drivers age and driving experience in budapest and islamabad. *European Transport Trasporti Europei*, n.2:1–18, 12 2020.
- [17] Ray Fuller. Towards a general theory of driver behaviour. *Accident Analysis & Prevention*, 37(3):461–472, 2005.
- [18] Roberto Girau, Raimondo Cossu, Massimo Farina, Virginia Pilloni, and Luigi Atzori. Virtual user in the iot: Definition, technologies and experiments. *Sensors*, 19(20), 2019.
- [19] Roberto Girau, Virginia Pilloni, and Luigi Atzori. The virtual user: The holistic manager of our iot applications. In *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, pages 149–154, 2018.
- [20] Daniela Godoy and Analía Amandi. User profiling in personal information agents: A survey. *The Knowledge Engineering Review*, 20:329–361, 12 2005.
- [21] Summala Heikki. *Towards Understanding Motivational and Emotional Factors in Driver Behaviour: Comfort Through Satisficing*, pages 189–207. Springer London, London, 2007.
- [22] Zhongxu Hu, Shanhe Lou, Yang Xing, Xiao Wang, Dongpu Cao, and Chen Lv. Review and perspectives on driver digital twin and its enabling technologies for intelligent vehicles. *IEEE Transactions on Intelligent Vehicles*, 7(3):417–440, 2022.
- [23] Qiang Ji, Zhiwei Zhu, and P. Lan. Real-time nonintrusive monitoring and prediction of driver fatigue. *IEEE Transactions on Vehicular Technology*, 53(4):1052–1068, 2004.
- [24] David B. Kaber. Driver performance effects of simultaneous visual and cognitive distraction and adaptation behavior. <https://doi.org/10.1016/j.trf.2012.05.004>, 2012.

-
- [25] Akrivi Katifori, Maria Golemati, Costas Vassilakis, George Lepouras, and Constantin Halatsis. Creating an ontology for the user profile: Method and applications. pages 407–412, 01 2007.
- [26] Marcin Kwiatkowski. Hexagonal architecture (ports and adapters): Pros and cons, 2023.
- [27] David T. Levy. Youth and traffic safety: The effects of driving age, experience, and education. *Accident Analysis & Prevention*, 22(4):327–334, 1990.
- [28] Craig Lyon, Dan Mayhew, Marie-Axelle Granié, Robyn Robertson, Ward Vanlaar, Heather Woods-Fry, Chloé Thevenet, Gerald Furian, and Aggelos Soteropoulos. Age and road safety performance: Focusing on elderly and young drivers. *IATSS Research*, 44(3):212–219, 2020.
- [29] Azad M. Madni, Carla C. Madni, and Scott D. Lucero. Leveraging digital twin technology in model-based systems engineering. *Systems*, 7(1), 2019.
- [30] Dawn L. Massie, Paul E. Green, and Kenneth L. Campbell. Crash involvement rates by driver gender and the role of average annual mileage. *Accident Analysis & Prevention*, 29(5):675–685, 1997.
- [31] Maria Mengozzi. *Studio del fitness-to-drive: sensing delle attività di guida e monitoraggio del driver*. PhD thesis.
- [32] Grieves Michael. Product lifecycle management: Driving the next generation of lean thinking. 01 2005.
- [33] Roberto Minerva, Gyu Myoung Lee, and Noël Crespi. Digital twin in the iot context: A survey on technical features, scenarios, and architectural models. *Proceedings of the IEEE*, 108(10):1785–1824, 2020.
- [34] Gianfranco E. Modoni and Marco Sacco. A human digital-twin-based framework driving human centricity towards industry 5.0. *Sensors*, 23(13), 2023.

-
- [35] Samuel D. Okegbile, Jun Cai, Changyan Yi, and Dusit Niyato. Human digital twin for personalized healthcare: Vision, architecture and future directions. *IEEE Network*, pages 1–7, 2022.
- [36] Sara Ouafthouh, Ahmed Zellou, and Ali Idri. User profile model: A user dimension based classification. In *2015 10th International Conference on Intelligent Systems: Theories and Applications (SITA)*, pages 1–5, 2015.
- [37] Heribert Pascual, Xavi Masip Bruin, Albert Alonso, and Judit Cerdà. A systematic review on human modeling: Digging into human digital twin implementations, 2023.
- [38] Dafne Piersma, Anselm Fuermaier, Dick Waard, Ragnhild Davidse, Jolieke de Groot-Mesken, Michelle Doumen, Rutger Bredewoud, René Claesen, Afina Lemstra, Annemiek Vermeeren, Rudolf Ponds, Frans Verhey, Wiebo Brouwer, and Oliver Tucha. Prediction of fitness to drive in patients with alzheimer’s dementia. *PLOS ONE*, 11:e0149566, 02 2016.
- [39] Dimitrios Piromalis and Antreas Kantaros. Digital twins in the automotive industry: The road toward physical-digital convergence. *Applied System Innovation*, 5(4), 2022.
- [40] Elaine Rich. User modeling via stereotypes. *Cognitive Science*, 3(4):329–354, 1979.
- [41] Benjamin Schleich, Nabil Anwer, Luc Mathieu, and Sandro Wartzack. Shaping the digital twin for design and production engineering. *CIRP Annals*, 66(1):141–144, 2017.
- [42] Rainer Stark and Thomas Damerou. *Digital Twin*. 05 2019.
- [43] Kazunari Sugiyama, Kenji Hatano, and Masatoshi Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings of the 13th International Conference on World Wide Web, WWW '04*, page 675–684, New York, NY, USA, 2004. Association for Computing Machinery.

- [44] Tomer Toledo, Oren Musicant, and Tsippy Lotan. In-vehicle data recorders for monitoring and feedback on drivers' behavior. *Transportation Research Part C: Emerging Technologies*, 16(3):320–331, 2008. Emerging Commercial Technologies.
- [45] Yi-Cheng Tsai, Wei-Hsun Lee, and Chien-Ming Chou. A safety driving assistance system by integrating in-vehicle dynamics and real-time traffic information. In *2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST)*, pages 416–421, 2017.
- [46] Dwi Phalita Upahita, Yiik Diew Wong, and Kit Meng Lum. Effect of driving experience and driving inactivity on young driver's hazard mitigation skills. *Transportation Research Part F: Traffic Psychology and Behaviour*, 59:286–297, 2018.
- [47] Eric VanDerHorn and Sankaran Mahadevan. Digital twin: Generalization, characterization and implementation. *Decision Support Systems*, 145:113524, 2021.