



UNIVERSITÀ
DI TRENTO

Dipartimento di Ingegneria e Scienza dell'Informazione

Corso di Laurea in
Informatica

ELABORATO FINALE

VOLLEYVISIONAI: WEB APP PER
L'ANALISI VIDEO NELLA PALLAVOLO

Supervisore

Luca Turchet

Laureando

Andrea Richichi

Co-supervisore

Maurizio Napolitano

Anno accademico 2023/2024

Indice

Abstract	3
1 Introduzione	4
2 Stato dell'Arte	5
2.1 Sport 4.0	5
2.2 Analisi delle Performance	5
2.2.1 Esempi	6
2.3 Competitors	6
2.3.1 Video Analysis	7
2.3.2 Hybrid Systems	8
2.3.3 Volleyball Specific	9
3 Analisi e progettazione	10
3.1 Perché una Web App?	10
3.2 Analisi dei Requisiti	11
3.2.1 Requisiti Funzionali	11
3.2.2 Requisiti Non Funzionali	12
3.3 Progettazione	12
3.3.1 Personas e Scenari	12
3.3.2 Struttura e Interfaccia	13
3.3.3 Server	13
4 Sviluppo e implementazione	14
4.1 Tecnologie utilizzate	14
4.1.1 Node.js	14
4.1.2 React.js	14
4.1.3 Figma	14
4.1.4 Uvicorn	15
4.1.5 FastAPI	15
4.1.6 YOLOv9	15
4.1.7 OpenCV	15
4.2 Implementazione	16
4.2.1 Architettura	16
4.3 UI e Funzionamento Web App	17
4.3.1 Menu iniziale	17
4.3.2 Analisi Manuale	17
4.3.3 Analisi AI	20
5 Evaluation	21
5.1 Usability Testing	21
5.2 Feedback e Miglioramenti	22

6 Conclusioni	23
6.1 Raggiungimento degli obiettivi	23
6.1.1 Funzionalità della Web App	23
6.2 Sviluppi Futuri	23
6.3 Formazione	24
6.4 Conclusioni Personalali	24
Bibliografia	24

Abstract

L'elaborato presenta lo sviluppo e l'implementazione di *VolleyVisionAI*, una Web App progettata per l'analisi video nello sport della pallavolo. L'obiettivo principale è quello di colmare il vuoto di mercato tra gli strumenti professionali, spesso inaccessibili, e l'esigenza di un'applicazione intuitiva e semplice rivolta ad un'utenza amatoriale.

VolleyVisionAI presenta due modalità di funzionamento: la modalità manuale, che permette agli utenti di creare, importare ed esportare progetti di analisi video, effettuare *Event Tagging* e generare videoclip personalizzati; e la modalità AI, che utilizza modelli di *Computer Vision* per tracciare la palla e riconoscere i giocatori in campo. Queste funzionalità sono state implementate sviluppando un'architettura che garantisce alte prestazioni e scalabilità, con *React.js* per il front-end e *FastAPI* per il back-end.

Il processo di sviluppo ha compreso l'analisi dei requisiti, la creazione di un prototipo iniziale con *Figma*, e lo sviluppo tecnico di interfaccia utente in *JavaScript* e funzionalità back-end in *Python*. Queste ultime comprendono l'analisi tramite AI per la quale è stato integrato il modello di riconoscimento *YOLOv9* e la libreria *OpenCV*.

I test di usabilità effettuati hanno confermato l'efficacia dell'applicazione e l'intuitività della sua interfaccia. Inoltre i feedback ricevuti hanno permesso di migliorare ulteriormente la *user-experience* e offrire spunti per sviluppi futuri dell'app. La versione beta di *VolleyVisionAI* è attualmente operativa e ha raggiunto pienamente gli obiettivi prefissati. I risultati ottenuti offrono nuove opportunità per il futuro, tra cui l'aggiunta di nuove funzionalità nella modalità AI e l'estensione dell'accessibilità attraverso un sistema di *Deployment*.

1 Introduzione

L'obiettivo di questo progetto è sviluppare una Web App per l'analisi video nel volley dedicata ad allenatori e atleti. L'idea nasce in seguito ad alcune ricerche di mercato che hanno evidenziato l'assenza di soluzioni *user-friendly* gratuite, per l'analisi video nello sport della pallavolo, adatte ad utenti amatoriali. L'applicazione, rispetto ad altre soluzioni già presenti sul mercato, ha infatti lo scopo principale di avere una *User Interface* estremamente semplice ed intuitiva, offrendo allo stesso tempo un riuso gratuito. Questo permette di garantire una *User Experience* di alto livello, qualunque siano le conoscenze tecniche dell'utente finale.

Le competenze accademiche e lavorative acquisite durante il percorso di laurea Triennale in Informatica a Trento, sono risultate fondamentali in fase di progettazione e sviluppo. Il corso online tenuto da EyeStudios Academy [5] su *Figma* ha fornito l'esperienza necessaria per realizzare un prototipo interattivo dell'app, velocizzando così il processo di sviluppo.

L'esperienza di tirocinio presso la *Fondazione Bruno Kessler* ha inoltre contribuito a consolidare la conoscenza della libreria *React* e del linguaggio *JavaScript*, ampiamente utilizzati nello lo sviluppo della Web App.

Il corso di Ingegneria del Software ha aiutato a definire, in fase di analisi e progettazione, le funzionalità dell'app individuando requisiti funzionali e non funzionali.

Il resto dell'elaborato prevede questi capitoli:

- **Stato dell'Arte:** dove viene introdotto il concetto di *Sport4.0* con particolare attenzione al mondo della pallavolo e presentate le tecnologie e le soluzioni attualmente presenti sul mercato.
- **Analisi e Progettazione:** presenta le fasi di analisi e progettazione della Web App con le definizioni dell'interfaccia e requisiti.
- **Implementazione e Sviluppo:** descrive lo sviluppo pratico dell'applicazione mediante la rappresentazione dell'architettura di sistema e le tecnologie utilizzate.
- **Evaluation:** qui vengono riportati i risultati relativi ad un test di usabilità effettuato da parte degli utenti. Sono, inoltre, presentati i loro feedback ed i miglioramenti apportati di conseguenza.
- **Conclusioni:** riassume le caratteristiche dell'app, ripercorrendo gli obiettivi raggiunti e conclude valutando possibili sviluppi futuri e riflessioni personali.

La tesi descrive i dettagli del progetto, iniziando dall'analisi del contesto e delle esigenze degli utenti, continuando con la progettazione e lo sviluppo della Web App, fino alla valutazione dei risultati ottenuti. Il tutto è debitamente illustrato, quando necessario, rendendo l'elaborato maggiormente comprensibile.

2 Stato dell'Arte

Questo capitolo esamina come il mondo dello sport stia attraversando una forte evoluzione, segnata dall'integrazione di nuove tecnologie. Di seguito viene introdotto, in particolare, il concetto di *Sport 4.0*, necessario per capire il contesto dell'elaborato.

2.1 Sport 4.0

Questo nuovo paradigma, denominato *Sport 4.0* [9], si ispira al concetto di Industria 4.0, con l'obiettivo di rivoluzionare le attività sportive sotto tutti i punti di vista. In particolare, le pratiche tradizionali vengono integrate con tecnologie quali *Internet of Things*, *Artificial Intelligence*, *Machine Learning*, *Augmented Reality* e *Virtual Reality*.

Lo scopo è migliorare le prestazioni degli atleti, ottimizzare le strategie di gioco e coinvolgere i tifosi in nuove esperienze.

Le principali aree di applicazione sono infatti:

- **Analisi delle Performance:** analisi delle prestazioni degli atleti come singoli e come squadra, con lo scopo di ottimizzare e migliorare le loro performance in gara.
- **Coinvolgimento dei Fan:** che ha l'obiettivo di offrire esperienze immersive e coinvolgenti ai tifosi, tramite l'integrazione di realtà aumentata e virtuale.
- **Infrastrutture Smart:** introduce il concetto di *Internet of Things*, utilizzato per la gestione Smart delle strutture sportive. Con lo scopo di renderle più sostenibili da un punto di vista energetico e di aumentare la sicurezza dei tifosi.
- **Salute e Benessere:** si interessa di monitorare, tramite l'utilizzo di dispositivi specifici che raccolgono dati biometrici, le metriche di salute degli atleti permettendo quindi di controllare gli sportivi in maniera più approfondita e personalizzata.

In sintesi, *Sport 4.0* rappresenta una grande evoluzione per il mondo sportivo. Integrando tecnologie innovative che ottimizzano ogni aspetto delle attività, dall'allenamento alla competizione, fino all'esperienza dei tifosi.

2.2 Analisi delle Performance

Questo è uno degli ambiti più rivoluzionati da *Sport 4.0*, grazie all'integrazione di software e hardware che hanno permesso un approccio *Data-Driven* nell'analisi degli atleti. L'implementazione e l'utilizzo di Software di analisi video, come alcuni competitor successivamente presentati, consentono di estrarre da registrazioni di allenamenti e competizioni informazioni e statistiche dettagliate.

L'utilizzo di hardware, invece, come sensori biometrici e dispositivi indossabili, consente di monitorare molti più parametri, dalla precisione dei movimenti alla resistenza fisica. Di conseguenza, si ottimizzano le performance atletiche e si minimizza il rischio di infortuni.

2.2.1 Esempi

Dispositivi Indossabili

Dal 2016 ad oggi, a dominare i primi tre posti delle classifiche di tendenze fitness, sono stati i Wearable. Così definiti, sono dei dispositivi indossabili come smartwatch e fitness tracker, che permettono di rilevare dati relativi alla salute e all'attività fisica. Pensati inizialmente per un pubblico di professionisti nello sport, nel tempo si sono estesi anche verso una clientela più generica con la passione del fitness. Questa larga diffusione sul mercato è stata resa possibile dalle interfacce estremamente *User-friendly*, che hanno reso i Wearable un must-have per chiunque voglia monitorare la propria attività fisica e il proprio benessere.

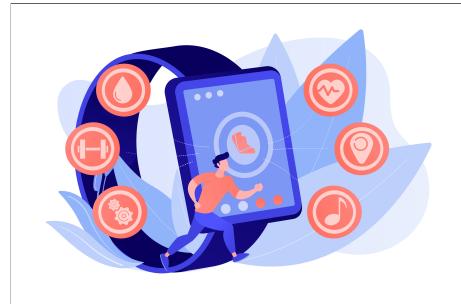


Figura 2.1: Wearable technology

Sistemi di Analisi Biomeccanica

Gli sport di squadra si distinguono per un elevato tasso di tecnica e tattica. La pallavolo, nello specifico, è uno sport dove la tecnica rappresenta una caratteristica discriminante nelle capacità di un giocatore. Per esempio, la realizzazione del fondamentale della schiacciata, richiede il coordinamento totale dell'atleta, partendo dalla sequenza dei passi (rincorsa), al salto, fino ad arrivare al movimento di braccia e colpo sulla palla con perfetta coordinazione spazio/temporale ed equilibrio. La biomeccanica si occupa proprio di analizzare le forze e l'efficacia dei singoli movimenti. Tramite algoritmi che utilizzano telecamere e sensori, elaborando anche lo scanner 3D dell'atleta, si può analizzare a 360 gradi ogni singola azione. Lo scopo di queste analisi è quello di aiutare staff ed allenatori nella creazione di allenamenti mirati alla correzione dei difetti rilevati.

Software di Analisi Video

Alcuni Club calcistici inglesi, come Leicester FC e Southampton FC, utilizzano largamente software di analisi video. Questi applicativi permettono di analizzare le prestazioni dei giocatori per raggiungere risultati precisi. Vengono utilizzati, ad esempio, in fase di scouting. L'analisi costante di giocatori dilettanti delle squadre giovanili, infatti, fornisce alle società una banca dati di informazioni che possono aiutare il processo della ricerca di nuovi talenti. Allo stesso modo, vengono analizzati anche i giocatori che le società danno in prestito. Anche qui lo scopo è di seguirli a 360 gradi per tutto il periodo contrattuale.



Figura 2.2: Video analysis

2.3 Competitors

Dopo aver approfondito il concetto di Sport 4.0 ed in particolare le tecnologie legate all'analisi delle performance, segue una prima analisi di mercato. Questo passaggio identifica lo stato dell'arte, e comprende al meglio le preferenze degli utenti. Grazie a queste informazioni, è possibile delineare con maggiore precisione le direzioni di sviluppo della Web App, assicurando che soddisfi le esigenze degli utenti e che si differenzi dalle soluzioni attualmente disponibili.

I competitor sono divisi in due categorie principali: chi si concentra esclusivamente sulla *video analysis* e chi la combina con l'uso di hardware integrativi, denominati *Hybrid Systems*.

Di seguito vengono illustrate le due tipologie, analizzando alcune soluzioni per ognuna. Da qui sono inoltre presentati due prodotti verticali allo sport della pallavolo. Questo aiuterà a definire meglio, in seguito, le caratteristiche del progetto.

2.3.1 Video Analysis

Le applicazioni e i software di analisi video rappresentano uno strumento essenziale per lo studio delle performance sportive. Questi strumenti permettono di fornire dashboard interattive a staff ed allenatori per svolgere analisi specifiche. Inoltre, i sistemi più avanzati sono integrati con algoritmi di *Computer Vision*, che permettono di raccogliere informazioni automaticamente, semplificandone ulteriormente l'uso. Di seguito vengono presentati tre competitor che si concentrano esclusivamente sulla video analysis.

Dartfish

Dartfish [3] è una soluzione che permette l'automazione di alcuni processi di analisi utilizzando modelli di *Computer Vision*. Tramite algoritmi di *pose estimation* e in alcuni casi di *object tracking*, il software estrae informazioni dai video forniti. I dati raccolti vengono poi elaborati per fornire statistiche e grafiche utili all'utente finale. *Dartfish* propone 3 soluzioni così suddivise: *game analysis*, *motion analysis* e *HealthCare*. Questi prodotti, anche se con funzionalità differenti tra loro, hanno lo scopo comune di aiutare il lavoro degli analisti sportivi. *Dartfish* non è indicato per l'utilizzo nell'ambito della pallavolo, non fornisce infatti modelli specifici per questo sport.

LongoMatch

LongoMatch [7] - nato come progetto open source poi diventato proprietario - si concentra sull'analisi di eventi nei video. Infatti è utilizzato principalmente per la sua funzionalità di *tagging*. Questa feature consente agli utenti di annotare azioni e momenti d'interesse durante la fase di analisi. L'interfaccia dell'app è personalizzabile dal singolo utente, permettendo agli analisti sportivi di creare *dashboard* secondo le loro necessità. Un'altra delle caratteristiche specifiche di *LongoMatch* è definita come "Drawing" (illustrata in figura 2.3b) che permette di disegnare direttamente sui frame video, utile quando si vogliono sviluppare schemi di gioco. *LongoMatch* risulta così essere un software versatile, ma allo stesso tempo complesso per utenti non esperti. Infatti, *LongoMatch*, si ritrova ad essere una soluzione non sufficientemente *user-friendly* per un utente amatoriale, ma allo stesso tempo con funzionalità troppo limitate per un team professionale.

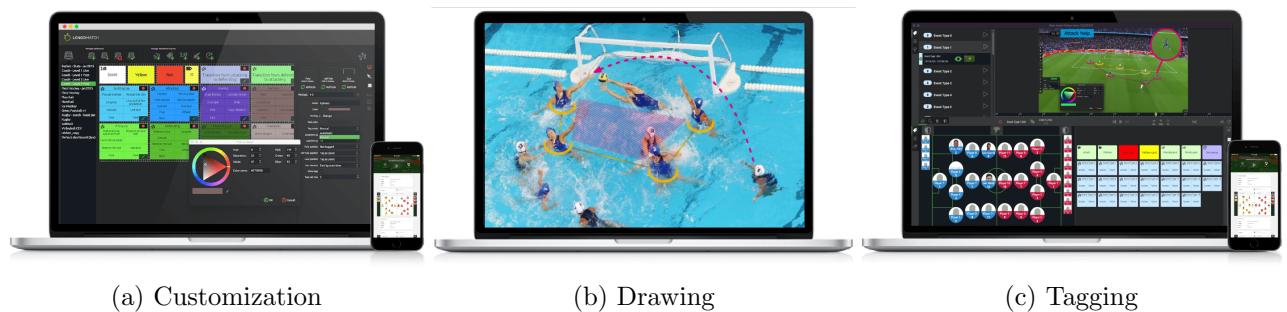


Figura 2.3: LongoMatch Functionalities

NAC Sport

NAC Sport [8] è un app con caratteristiche molto simili a quelle di *LongoMatch*. Ciò che la distingue, è la possibilità di effettuare analisi in *realtime*, fornendo all'utente feedback istantanei. Inoltre supporta il riconoscimento tridimensionale di palla e giocatori, per avere un panorama a 360 gradi di un determinato evento. Come *LongoMatch*, implementa funzionalità di tagging eventi tramite dashboard o shortcut da tastiera preimpostati. Implementa inoltre grafiche interattive, come ad esempio *heatmaps*, per fornire ulteriori dettagli all'utente. Economicamente, *NAC Sport* risulta essere più costoso rispetto ad altre alternative, proprio perchè è stato progettato per soddisfare le esigenze di un target professionale. Questi aspetti rendono *NAC Sport* uno software completo, ma non adatto ad un bacino di utenti amatoriali.

2.3.2 Hybrid Systems

Gli *Hybrid Systems*, come dice la sigla stessa, combinano software di analisi video e tecnologie esterne. Alcuni esempi di hardware integrativi sono i dispositivi wearable, i sensori e le telecamere specializzate. Questi sistemi risultano estremamente efficienti perché suddivisi in due fasi. Inizialmente, l'utilizzo di sensori permette di monitorare molti parametri che, successivamente, vengono utilizzati ed integrati nella fase di analisi video per avere risultati più precisi. Di seguito sono presentate ed illustrate 2 soluzioni leader nel settore.

Hudl

Hudl [6] è una piattaforma di analisi video sportiva, utilizzata da oltre *8 milioni* di utenti registrati e *230.000 squadre* in più di *40 sport*, con *16 milioni* di download dell'app.

Hudl offre una vasta gamma di prodotti innovativi per rispondere alle esigenze del mondo dello *Sport 4.0*. I principali sono: *Sportscode*, *Volleymetrics*, *Assist*, *WIMU* e *Hudl Focus*.

Proprio quest'ultimo è il servizio di punta dell'azienda, che integra l'analisi video con l'utilizzo di telecamere progettate da Hudl stessa. In particolare l'azienda ha sviluppato software dedicati che permettono alle camere di seguire le azioni live in maniera automatica, offrendo una soluzione completamente *hands-free*. Hudl Focus utilizza queste camere intelligenti per registrare ogni momento del gioco senza interventi manuali. Questo servizio è particolarmente utilizzato negli sport in cui il campo di gioco è ampio, ad esempio calcio o football americano.

Catapult

Catapult [2] è una piattaforma leader nel monitoraggio e nell'analisi sportiva, utilizzati da oltre *4.200 squadre* in più di *40 sport* in tutto il mondo. Uno dei loro prodotti di punta è *Vector Core*.

Questa soluzione fornisce pettorine con tecnologie GPS/GNSS per tracciare i movimenti degli atleti, fornendo dati in *realtime* ma anche post-sessione. Il prodotto ha riscosso grande successo perchè facile da configurare e offre un'ampia gamma di metriche per l'analisi delle prestazioni fisiche.

Le caratteristiche principali di Vector Core includono:

- *Tracciamento integrato indoor e outdoor*
- *Tecnologia GPS/GNSS a 10 Hz*
- *Analisi inerziale del movimento*
- *Compatibilità con monitoraggio della frequenza cardiaca*
- *Reportistica dettagliata con oltre 200 parametri live e 1.800 parametri post-sessione*



Figura 2.4: Hudl Focus



Figura 2.5: Catapult Vector Core

2.3.3 Volleyball Specific

Alcune aziende si sono specializzate nel fornire soluzioni verticali per specifici sport, come la pallavolo. Di seguito analizziamo due esempi rilevanti.

DataVolley 4.0

DataVolley 4.0 [4] è uno dei prodotti di riferimento nella pallavolo internazionale ed è sviluppato dall'azienda Data Project. Il software è progettato per la raccolta e analisi di dati di partite ed allenamenti. Il suo punto di forza risiede nell'interfaccia per la raccolta dei dati (molto simile alla stenografia) e nelle successive funzioni che permettono di generare report e statistiche relative alle squadre e ad ogni singolo giocatore associando anche il video. L'applicazione è focalizzata in particolare alla raccolta dei dati delle performance e non all'analisi dei singoli fondamentali (e quindi la tattica). Inoltre richiede un po' di addestramento per imparare ad utilizzarlo, per cui utenti amatoriali finiscono per non farne uso anche se esistono versioni economiche.

BallTime

BallTime [1] è una startup NewYorkese che sfrutta modelli di Computer Vision per l'analisi video. In particolare i modelli di ML riescono a:

- Riconoscere e distinguere i singoli giocatori
- Tracciare la palla
- Analizzare pattern di gioco
- Riconoscere azioni e segnare il risultato

La startup, nata a Settembre 2022, fornisce una web app completa, ma ancora in costante sviluppo. Infatti questa soluzione sta introducendo le funzionalità di volta in volta guidata dalla domanda. L'approccio innovativo e soprattutto *user-friendly*, il prezzo contenuto e l'attività di marketing che l'azienda sta sviluppando, rendono *BallTime* una soluzione competitiva dove le alternative dovranno puntare su funzioni diverse o strategie di creazione di comunità.

3 Analisi e progettazione

Analizzando i vari competitor descritti nel precedente capitolo, si può notare che la maggior parte delle soluzioni presenti sul mercato sono rivolte ad una clientela qualificata. Questo probabilmente per avvicinarsi ad un utenza con più disponibilità economica. Anche soluzioni come *LongoMatch*, che offrono funzionalità più semplici, non sono progettate per un utilizzo amatoriale.

In seguito a queste analisi, ho deciso di sviluppare un'applicazione orientata ad un unico sport che fosse estremamente *user-friendly*, con un'interfaccia intuitiva e con un set di funzionalità relativamente limitato. Ho deciso di focalizzarmi sulla pallavolo, uno sport che ho praticato per quasi 10 anni. Le caratteristiche di questo sport lo rendono ideale per l'analisi video, in quanto vi sono pochi giocatori e il campo ha dimensioni ridotte. Questo permette di avere una visione chiara di tutte le dinamiche di gioco, rendendole facilmente analizzabili.

La soluzione proposta è caratterizzata da un'interfaccia utente estremamente intuitiva, ideata per essere facilmente utilizzabile da utenti amatoriali. Inoltre, seguendo l'esempio della start-up *BallTime*, oltre alle comuni funzionalità di analisi, ho implementato alcune funzionalità AI offrendo quindi due modalità di uso: Manuale e AI.

Le due modalità si distinguono in questo modo:

- **Manuale:** Questa modalità permette di creare, importare ed esportare progetti. In particolare sono implementate le funzionalità di:
 - *Event Tagging*: che offre la possibilità di salvare eventi d'interesse durante l'analisi video. Sono caratterizzati dall'utilizzo di shortcut personalizzabili secondo atleta, azione e comando rapido.
 - *Drawing Tools*: attraverso cui è possibile annotare ed illustrare tattiche o azioni di gioco direttamente sul frame video.
 - *Videoclip Generator*: per generare highlights in maniera completamente automatica, partendo dagli eventi taggati.
- **AI:** Questa modalità implementa l'utilizzo di modelli di Computer Vision per l'analisi automatica dei video. In particolare sono implementate le funzionalità di:
 - *Ball Trajectory*: visualizzazione della traiettoria della palla durante il gioco. Disegnando una scia gialla sul video.
 - *Player Recognition*: per identificare automaticamente i giocatori in campo, assegnando ad ognuno un id e un colore di bounding box diverso.

I risultati dell'analisi di mercato hanno inoltre evidenziato che i prodotti si suddividono in due categorie: *Desktop App* e *Web App*. Al fine di favorire il riuso su più scenari ho deciso di creare una Web App che, oltre a poter essere disponibile su un server, può essere eseguita su un computer in locale permettendo quindi scenari di servizio in cloud o di riuso su una propria macchina dedicata senza necessità di creazione di utenze

3.1 Perché una Web App?

La decisione di sviluppare una Web App invece che una desktop app è basata su diversi fattori.

Una prima analisi ha delineato come l'utilizzo di siti web risulti più semplice per un target di utenza amatoriale. Questo perchè non richiede l'installazione secondo sistemi operativi specifici e può essere eseguita su un qualsiasi browser.

Questo rende l'accesso all'app semplice ed immediato, creando un'esperienza più *user-friendly* con meno problematiche sul fronte della configurazione. Inoltre, offrendo potenzialmente il servizio in

cloud, diventa molto più facile la gestione degli aggiornamenti in quanto saranno sempre centralizzati, rendendo l'esperienza utente più semplice e lontana da noiose installazioni manuali necessarie a garantire sempre l'ultima versione.

Un ulteriore vantaggio viene dalla funzionalità AI. Infatti, l'uso di modelli di Computer Vision richiedono risorse computazionali che non sempre sono disponibili a chiunque, sviluppando l'applicazione con interfaccia web è possibile fare uso di tecniche di *cloud computing* in cui è possibile delegare i processi computazionalmente intensi a server remoti. Questo garantisce una *user-experience* il più fluida e performante possibile.

Pertanto, seguendo la volontà di creare un prodotto semplice e con basse problematiche agli utenti finali, l'approccio web è risultato quello vincente, dato che il bacino di utenza a cui si rivolge è quello di utenti amatoriali con poche disponibilità di tempo e di risorse.

3.2 Analisi dei Requisiti

La Web App ha lo scopo di rivolgersi ad un target non professionale, quali atleti ed allenatori amatoriali. Identificandosi in questa utenza, sono state studiate le soluzioni già presenti e le funzionalità da loro proposte. In particolare, sono state prese in considerazione le feature di *LongoMatch* e *BallTime*, successivamente rivisitate e adattate per una *user-experience* orientata a rendere il tutto più facile. In fase di analisi è stato fondamentale definire quelli che sono i requisiti funzionali e non funzionali del sistema.

3.2.1 Requisiti Funzionali

La definizione dei Requisiti Funzionali è fondamentale in fase di analisi. Ci permette infatti di identificare le funzionalità chiave che il sistema deve garantire all'utente finale. Inoltre è utile per fare un brainstorming sulla direzione che il progetto dovrà prendere.

Di seguito sono elencati i requisiti funzionali del sistema, suddivisi per categorie:

- **Gestione Progetti**

- Gli utenti devono poter creare nuovi progetti, scegliendo video, giocatori e azioni.
- Gli utenti devono poter esportare progetti.
- Gli utenti devono poter importare progetti esistenti.

- **Modalità e Navigazione**

- Gli utenti devono poter scegliere tra due pagine. Una gestisce l'analisi manuale e l'altra le funzionalità AI.
- Gli utenti devono poter navigare tra le due modalità in qualsiasi momento.

- **Analisi Manuale**

- Gli utenti devono poter etichettare eventi di loro interesse durante l'analisi video.
- Gli eventi etichettati devono essere personalizzabili e la loro attivazione deve essere rapida.
- Gli eventi etichettati devono salvare dati su giocatore, azione, minutaggio e comando rapido associato.
- Gli eventi etichettati devono essere valutabili.
- Gli utenti devono poter disegnare sui frame video.
- Gli utenti devono poter scegliere di creare videoclip partendo degli eventi etichettati.

- **Analisi AI**

- Gli utenti devono poter scegliere tra due modalità di analisi.
- Gli utenti devono poter visualizzare i risultati di queste analisi.
- Gli utenti devono poter scaricare i risultati delle analisi AI.

3.2.2 Requisiti Non Funzionali

Lo scopo dei Requisiti Non Funzionali è quello di descrivere le caratteristiche del software che non sono definite in termini di funzionalità. Questi requisiti riguardano il funzionamento del sistema e i vincoli da rispettare per un prodotto di qualità.

Di seguito sono elencati i principali requisiti non funzionali del sistema, suddivisi in tre macroaree:

- **Prestazioni**

- L'app deve garantire l'elaborazione video in tempi brevi.
- L'app deve essere estremamente reattiva agli input dell'utente.

- **Usabilità**

- L'app deve garantire una *UX* fluida e dinamica.
- L'app deve avere una *UI* chiara ed intuitiva.

- **Compatibilità**

- L'app deve essere compatibile con i principali browser web.
- L'app deve adattarsi a diverse dimensioni di schermo.

3.3 Progettazione

La progettazione della Web App ha mantenuto il focus sull'esperienza utente. Proprio per questo motivo ho creato un *mockup* dell'interfaccia dell'app utilizzando Figma. Per avere una visione più approfondita di comportamenti e abitudini del target, ho definito due *personas* e due *scenari*. Durante lo sviluppo dell'applicazione ho progettato *UX* e *UI*, tenendo conto di questi preconcetti.

3.3.1 Personas e Scenari

Le *personas* sono personaggi inventati che rappresentano possibili utenti reali. Gli *scenari*, invece, descrivono dettagliatamente delle situazioni specifiche in cui l'app potrebbe essere utilizzata. In fase di progettazione e sviluppo del prototipo Figma, definire questi concetti mi ha aiutato a capire come il target potrebbe interagire con l'applicazione.

Personas

- **Marco**, 45 anni, vive a Trento ed è allenatore di una delle squadra di pallavolo Under-16 della provincia. Durante le partite è solito annotare con carta e penna le sue considerazioni rispetto alle prestazioni della squadra. Proprio a causa di questo però, gli capita spesso di perdere fasi di gioco importanti. Gli piacerebbe, quindi, poter registrare le partite, per poi fare le sue valutazioni a casa.
- **Gaia**, 22 anni, vive a Modena ed è una giocatrice di pallavolo in una squadra universitaria. Durante le partite è solita registrarsi, chiedendo poi a suo fratello, una volta a casa, di creare highlights delle sue migliori azioni. A Gaia da fastidio non poter essere autonoma, ma proprio non riesce a prendere praticità con le app di editing video e non trova nessuna altra soluzione.

Scenari

- **Scenario A:** Durante un time-out, l'allenatore raduna i giocatori attorno alla panchina. Mostra sul suo tablet il video dell'azione appena conclusa utilizzando VolleyVisionAI e con un pennino inizia a disegnare sul video, evidenziando movimenti e possibili tattiche. I giocatori hanno così istruzioni chiare su come procedere, tornando in campo più consapevoli e preparati.
- **Scenario B:** Prima di una gara importante, l'allenatore decide di convocare la squadra per sviluppare insieme un piano di gioco. Tramite l'utilizzo di VolleyVisionAI l'allenatore si è preparato il giorno prima dei video di alcune fasi di gioco della squadra avversaria. Successivamente

ha mostrato i videoclip ai giocatori per poi interrogarli su come avrebbero reagito in determinate situazioni. Questa sessione di confronto ha permesso di migliorare l'intesa di squadra e di prepararsi per la gara.

3.3.2 Struttura e Interfaccia

L'utilizzo di Figma per la creazione di un prototipo della *User Interface* ha permesso di capire come impostare la struttura della Web App. Il corso di *EyeStudios Academy* mi ha fornito le competenze tecniche necessarie per la creazione del mockup.

VolleyVisionAI è stato ideato per avere un menu principale con due opzioni, come già descritto nell'introduzione di questo capitolo. Entrambe le funzionalità redirigono l'utente ad una nuova pagina, di seguito descritte.

La prima funzionalità, *Manual*, è stata strutturata in modo da rendere chiari i tre componenti principali. Sulla destra troviamo il video player con sotto un box contenente due campi utili alla creazione degli shortcut. Sulla sinistra, invece, troviamo il contenitore di tutti gli eventi taggati fino a quel momento. Questa struttura offre una *User Interface* e *User Experience* semplice ed intuitiva.

La seconda funzionalità, *AI*, presenta invece un video player quasi a tutto schermo con sopra la descrizione di ciò che viene rappresentato. Questa interfaccia minimalista aiuta l'utente a concentrarsi unicamente sul risultato dell'analisi AI.

La Web App presenta come colore predominante l'azzurro, scelto poiché trasmette tranquillità e professionalità. Inoltre le sue sfumature rendono ben leggibili i vari componenti dell'interfaccia e, allo stesso tempo, non stancano l'occhio dell'utente. In fase di progettazione, ho notato come l'azzurro fosse ideale per rappresentare il connubio tra sport e tecnologia: evoca dinamismo ed energia, tipici dello sport, e trasmette al contempo freschezza e innovazione, caratteristiche della tecnologia. Così, l'azzurro bilancia intensità e modernità, rendendo l'interfaccia armoniosa ed efficace.

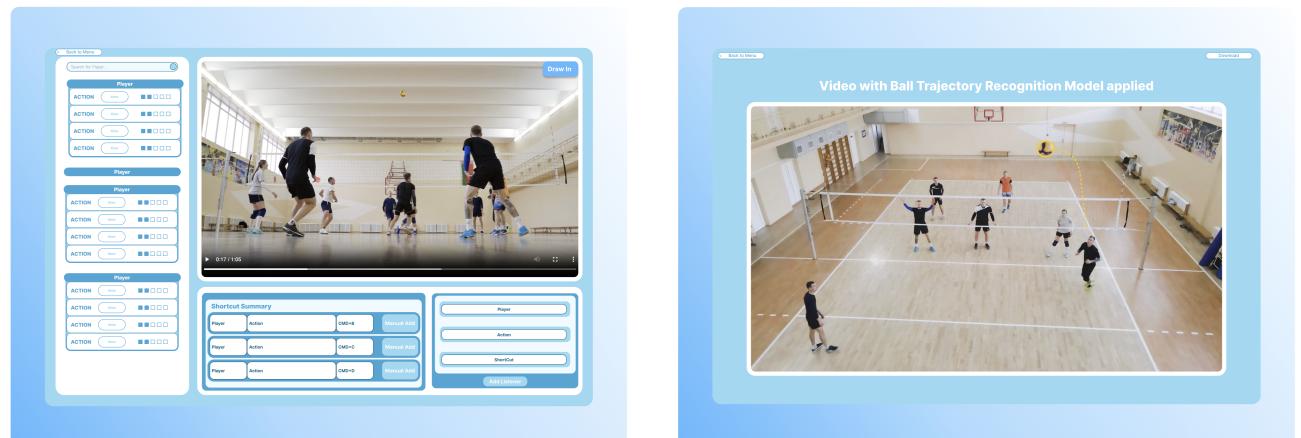


Figura 3.1: Figma Mockup

3.3.3 Server

Per garantire una manipolazione efficiente dei video e l'implementazione dei modelli di *Computer Vision*, è stato necessario sviluppare un server back-end robusto e affidabile. Questo server svolge un ruolo cruciale nella gestione delle operazioni computazionalmente intensive, liberando i dispositivi degli utenti dal carico di elaborazione e assicurando un'esperienza d'uso fluida e performante. Il Server è connesso direttamente al front-end della Web App tramite API, consentendo una comunicazione rapida e sicura tra l'interfaccia utente e il back-end. Questo permette di ricevere richieste, elaborare i dati dei video tramite i modelli di *Computer Vision* e restituire i risultati all'utente in tempo reale.

4 Sviluppo e implementazione

Il prototipo realizzato con Figma, per quanto semplice, ha permesso di indirizzare da subito i passaggi dello sviluppo: realizzazione del codice del front-end progettato e successiva implementazione dello sviluppo del back-end.

Successivamente viene descritto l'effettivo funzionamento dell'app, con relative illustrazioni, prestando attenzione anche ad architettura e tecnologie utilizzate.

4.1 Tecnologie utilizzate

Di seguito sono presentate brevemente le varie tecnologie utilizzate per l'implementazione di Volley-VisionAI.

4.1.1 Node.js

Node.js è un runtime system in JavaScript orientato agli eventi e completamente Open-Source. Garantisce performance e prestazioni elevate, basandosi sul motore JavaScript V8 di Google Chrome. Node.js è particolarmente valorizzato per il suo accesso a *Node Package Manager*. NPM è un gestore di pacchetti predefinito per l'ambiente di runtime JavaScript Node.js, nonché il più grande ecosistema di librerie open source al mondo. Questo permette agli sviluppatori di utilizzare soluzioni già pronte condivise dalla community, accelerando le fasi di programmazione e garantendo sicurezza e solidità del codice.



Figura 4.1: Logo NodeJS

4.1.2 React.js

ReactJS è una libreria JavaScript, anche essa Open Source, sviluppata da Meta per la costruzione di interfacce utente, particolarmente utilizzato per complesse *User-Interface* e applicazioni a pagina singola. Questa libreria è largamente utilizzata ed apprezzata per il suo vasto ecosistema, che include strumenti come *React Developer Tools* e *Create React App*. Per lo sviluppo della mia app ho adottato proprio quest'ultimo che utilizza *Node.js* e *NPM*, permettendomi di avviare un progetto React in maniera semplice e veloce. Inoltre React è una libreria che utilizza una struttura basata su componenti, che permettono di creare interfacce riutilizzabili e modulabili.

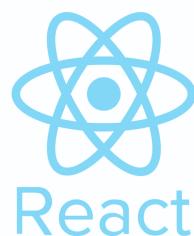


Figura 4.2: Logo ReactJS

4.1.3 Figma

Figma è uno strumento di design e grafica vettoriale utilizzato per la creazione di mockup e prototipi. Figma si distingue per essere un applicativo Web, ma che permette comunque soluzioni desktop e mobile. L'obiettivo di Figma è quello di permettere la creazione di design facendo particolare attenzione a *User Experience* e *User Interface*, permettendo inoltre la collaborazione *realtime* tra utenti. Figma è stato fondamentale in fase di analisi e progettazione, poiché mi ha permesso di avere un prototipo dell'app prima dell'implementazione effettiva.



Figura 4.3: Logo Figma

4.1.4 Uvicorn

Uvicorn è un server ASGI (Asynchronous Server Gateway Interface), ideato e progettato per eseguire applicazioni web asincrone in Python, in maniera leggera e veloce. La sua architettura asincrona lo rende ottimale per lavorare con framework moderni come *FastAPI* e *Starlette*, permettendogli di gestire un elevato numero di richieste simultaneamente senza problemi. La tecnologia HTTP/2 e WebSocket sono supportate da Uvicorn, rendendolo ideale per applicazioni realtime. Uno dei motivi principali per cui ho scelto di utilizzare un server meno conosciuto come questo è la sua compatibilità con *FastAPI*. Queste tecnologie combinate tra loro, mi permettono di avere un server Back-End veloce ma soprattutto scalabile.

4.1.5 FastAPI

FastAPI è un framework web progettato per creare API in Python. Il framework si basa su standard quali, OpenAPI e JSON Schema, che gli permettono di generare documentazione interattiva in maniera completamente automatica. Come descritto in precedenza, FastAPI utilizza *Uvicorn* come server ASGI a cui delega la gestione delle richieste. Questo permette di avere un'applicazione estremamente veloce, una caratteristica fondamentale per le applicazioni web moderne. Inoltre, ho optato per questa soluzione in quanto FastAPI ha una documentazione chiara e completa, che mi ha permesso di utilizzare il framework senza problemi.

4.1.6 YOLOv9

YOLOv9 è un modello di *object-detection* sviluppato da Ultralytics. Questa ultima versione della famiglia YOLO (You Only Look Once), è una delle soluzioni più avanzate presenti al momento sul mercato. YOLOv9 viene presentato in 5 varianti principali che si distinguono per la loro complessità. In particolare differiscono per la quantità di parametri di cui il modello dispone, ovvero, le varianti più complesse faranno rilevamenti più precisi a discapito di una maggiore richiesta computazionale. Nel contesto della mia Web App, avere la possibilità di scegliere quale tra le soluzioni fosse la più adatta al progetto mi ha fatto scegliere YOLOv9 come soluzione.

4.1.7 OpenCV

OpenCV, acronimo di Open Source Computer Vision, è una libreria per la visione artificiale e l'elaborazione delle immagini. È scritta in C++ ma ha anche altre interfacce, una di queste è in Python, che ho utilizzato nel mio progetto. Gli strumenti messi a disposizione da OpenCV vengono integrati con modelli, nel mio caso YOLOv9, utilizzando i loro risultati per elaborare specifiche grafiche video. La libreria è stata scelta per la sua documentazione e versatilità, ideale per uno sviluppatore che come me è alle prime armi con il mondo AI.



Figura 4.4: Logo Uvicorn



Figura 4.5: Logo FastAPI



Figura 4.6: Logo Ultralytics



Figura 4.7: Logo OpenCV

4.2 Implementazione

L'implementazione rispetta le premesse definite nelle fasi di analisi e progettazione, concentrandosi sul mantenere una *User Interface* pulita e un'architettura solida e scalabile. Attualmente la Web App non è ospitata pubblicamente (Deploy) in quanto ancora in fase di sviluppo. Lo sviluppo principale è stato lato Front-End implementando tutte le funzionalità definite nel capitolo 3. Per l'avvio ho utilizzato *create-react-app* necessario a creare l'ambiente di sviluppo. Questo configura in maniera autonoma, tramite l'utilizzo di *webpack-dev-server*, un server di sviluppo basato su *Node.js* che serve un interfaccia *React* sul browser locale offrendo così un ambiente caratterizzato dal *live reloading*, utile per vedere i cambiamenti apportati nel codice in *real time*. La componente *UnifiedContext.js* è stata utilizzata per centralizzare la gestione dello stato della Web App. In questo modo vengono salvati i dati inseriti dall'utente durante l'utilizzo dell'applicazione, come atleti, azioni, shortcut attivati ed eventi taggati. Nel Front-End sono implementate tutte le funzionalità ad eccezione di quelle che richiedono maggiori risorse computazionali (es la manipolazione di video o l'applicazione di modelli AI) in quanto delegate tramite API ad un server Back-End esterno. Questo approccio *Multi Server* è stato pensato per alleggerire il lavoro lato client e garantire una maggiore scalabilità in ottica degli sviluppi futuri dell'applicazione. Questa componente server Back-End è stata sviluppata in Python tramite il pacchetto Python *FastAPI* supportato a sua volta dal server ASGI *Uvicorn*. La scelta di questo Framework (vedi 4.1) ha permesso di creare API in maniera semplice. L'implementazione con FastAPI gestisce due funzionalità principali: la creazione di videoclip e l'applicazione di modelli AI. Nella prima viene utilizzato il modulo *moviepy* che permette l'*editing* di video. Mentre, per la seconda si è scelto il modello YOLOv9 nella variante YOLOv9c in quanto adatta alla complessità di object-detection che necessita la Web App. L'uso del modello è delegato alla libreria OpenCV con la sua implementazione Python, che permette la gestione grafica dei risultati. Per l'identificazione dei singoli atleti in campo è stato utilizzato l'algoritmo *SORT* (Simple Online and Realtime Tracking), che ha permesso di mantenere l'identità dei singoli per tutta la durata del video.

4.2.1 Architettura

L'architettura di sviluppo della Web App è rappresentata in figura 4.8.

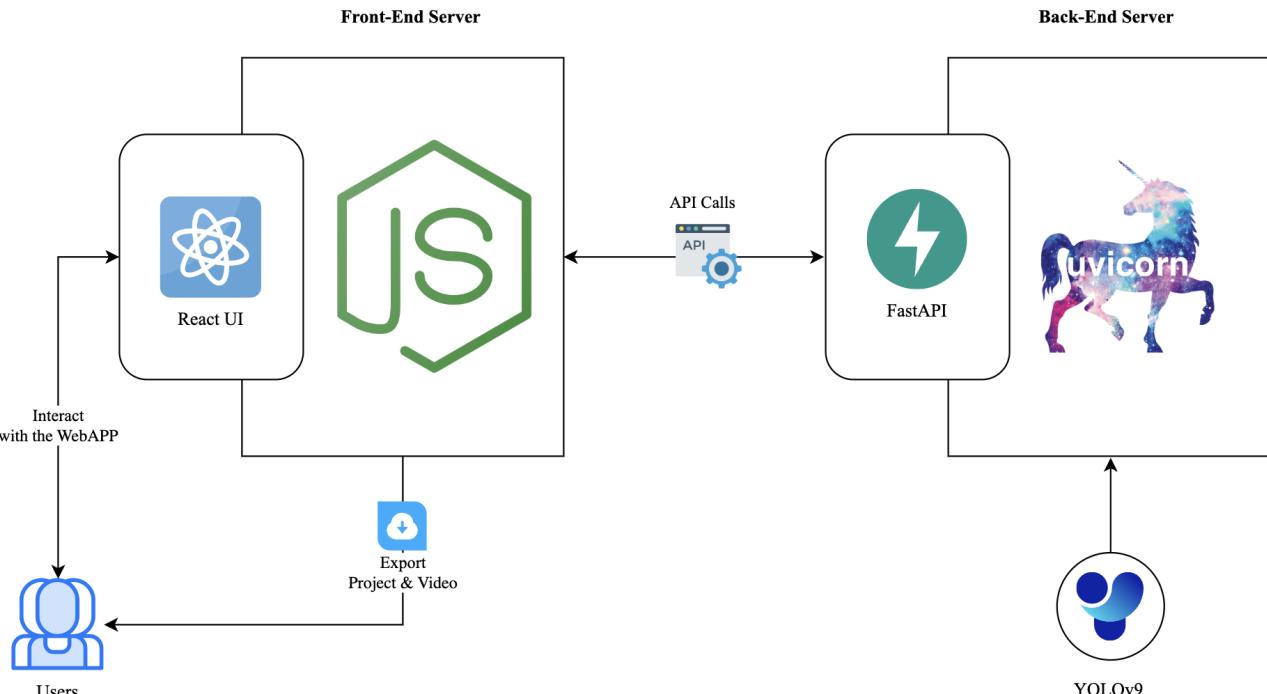


Figura 4.8: Architettura Web App

4.3 UI e Funzionamento Web App

Nel seguente paragrafo vengono presentate le funzionalità dell'app mantenendo un ordine logico di navigazione.

4.3.1 Menu iniziale

Il menu iniziale è la prima schermata a cui si accede lanciata la Web App. Questo ha la funzione di guidare l'utente attraverso le modalità base dell'applicazione. Il menu presenta il logo della Web App con le due principali opzioni d'uso: *Manual* ed *AI*.

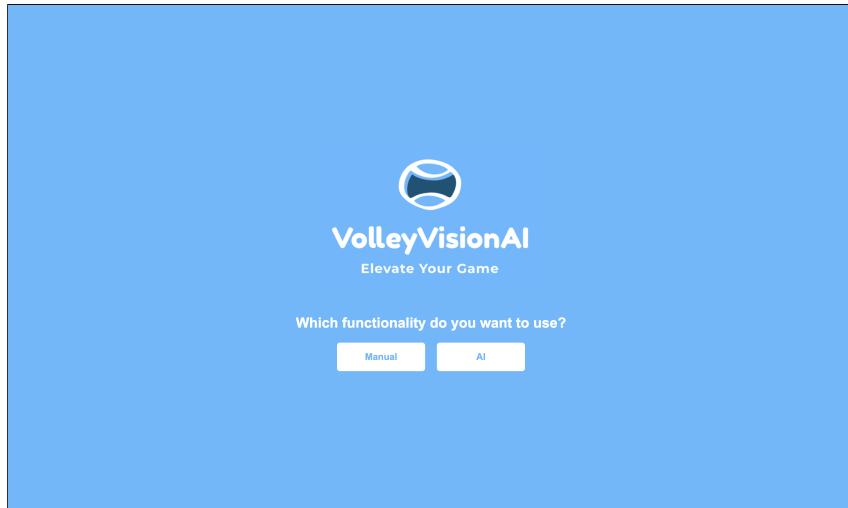


Figura 4.9: Menu iniziale

Ciascuna di queste opzioni porta ad una schermata con le relative funzionalità.

- **Manual:** una volta effettuata questa scelta, l'utente dovrà decidere se creare un nuovo progetto o caricare uno già esistente. Nel primo caso l'applicazione chiederà di inserire i nomi degli atleti e le azioni che vuole raccogliere. Essendo la Web App verticale per lo sport della pallavolo, sono già presenti i fondamentali principali come battuta, attacco, difesa, alzata e muro. Questo avviene per facilitare la configurazione, viene comunque permesso di aggiungere ulteriori etichette. Nel caso dell'importazione di un progetto già esistente, la risposta sarà di individuare sul *file-system* il file di un progetto precedentemente esportato. Effettuato questo setup di istruzione, l'applicazione prosegue con la schermata di *Analisi Manuale*.
- **AI:** questa modalità richiede di caricare il video da analizzare, specificando il tipo di riconoscimento desiderato, *Ball Tracking* o *Players Recognition*. Dato che questa operazione è delegata al back-end, l'UI mostrerà un messaggio di attesa finché il video non è stato elaborato, per poi passare alla schermata di *Analisi AI*.

In ciascuna delle due opzioni è permesso, in qualsiasi momento, di tornare al menu iniziale attraverso il bottone dedicato.

4.3.2 Analisi Manuale

La funzionalità di *Analisi Manuale* consente all'utente di eseguire operazioni manuali come il *tagging* degli eventi, il disegno su video e la creazione di videoclip. La schermata è caratterizzata da quattro elementi principali che permettono all'utente di svolgere le funzionalità descritte in precedenza. Inoltre l'interfaccia usa colori in maniera intelligente per differenziare i vari elementi migliorando la *User Experience* dell'utente finale.

Seguendo una logica di navigazione, le principali funzionalità vengono descritte di seguito.

Add Shortcut

posto in basso a destra, al di sotto del *VideoPlayer*, si trova un contenitore suddiviso in due componenti di cui l'*Add Shortcut* è quello sulla destra. Questo permette di creare nuovi shortcut personalizzati che servono durante la fase di *Event Tagging*. In particolare è composto da tre menu a tendina che permettono di selezionare l'atleta, l'azione e il comando rapido (shortcut) che si vuole creare. La configurazione dei parametri viene confermata al click sulla voce "Add Shortcut" sottostante. Naturalmente ogni comando rapido può essere assegnato ad un solo tipo di evento. Se ne viene inserito uno già presente allora l'applicazione rimuove il precedente e lo assegna alla nuova funzionalità scelta.

Shortcut Summary

Creando un qualsiasi shortcut, l'interfaccia presenta, all'interno dello stesso contenitore di *Add Shortcut* ma sulla sua sinistra, il pannello *Shortcut Summary*. La funzione di questo componente permette di tenere traccia degli shortcut creati, rendendoli visualizzabili in una lista che per ognuno ne definisce giocatore, azione e comando rapido. Per migliorare ulteriormente la *User Experience*, ogni shortcut è affiancato da un bottone "Manual Add" che permette ad utenti che non sanno utilizzare gli shortcut da tastiera, di poter comunque utilizzare questa funzionalità dell'app.

Tagged Events

Sulla sezione sinistra dello schermo troviamo uno dei componenti fondamentali della Web App. Una volta che uno shortcut viene attivato, sia tramite comando rapido da tastiera che tramite il bottone "Manual Add" nel componente *Shortcut Summary*, viene visualizzato qui dentro. Il componente è composto da una Search Bar che permette di cercare determinati eventi, selezionando anche se filtrarli in base al giocatore o all'azione. Nell'area sottostante sono invece elencati tutti gli eventi taggati in una lista suddivisa in gruppi. Gli eventi vengono raggruppati in base alla scelta dell'utente, può infatti decidere di visualizzarli in gruppi in base all'azione o in base al giocatore grazie



Figura 4.10: Add Shortcut



Figura 4.11: Enabled Shortcut



Figura 4.12: Tagged Events

al bottone che cambia tra "View by Action" e "View by Player". L'utente può cambiare questo tipo di visualizzazione in ogni momento. Ad ogni evento taggato corrispondono inoltre altre 2 variabili: il minutaggio e la valutazione. Il primo rappresenta il momento in cui è avvenuto l'evento taggato, dando la possibilità all'utente di cliccarlo per spostare il video a quel preciso istante. Il secondo invece è un componente composto da 5 stelle, inizialmente chiare, che sono utilizzabili dall'utente per dare valutazioni agli eventi taggati. È sempre disponibile, nel caso di uno shortcut attivato per errore, la possibilità di cancellarlo tramite un pulsante X presente accanto ad ognuno.

Video Player

In alto a destra, sopra al contenitore di *Shortcut Summary* e *Add Shortcut*, è collocato il componente principale di Volley-VisionAI, ovvero il *Video Player*. Questa sezione mostra all'utente il video caricato permettendo di interagire con esso come un normale *Player Video*, consentendo quindi di controllare la riproduzione, mettere in pausa, avanzare o tornare indietro nel video, e regolare il volume. Inoltre qui viene implementata la funzionalità di *Drawing* che permette di disegnare sul video. In particolare, tramite il bottone "Draw In" l'utente può attivare la funzionalità di disegno che implementa una dashboard molto semplice. Quest'ultima permette all'utente di scegliere se usare penna per disegnare o gomma per cancellare. Inoltre è possibile anche scegliere quello che è il colore del tratto, così da aiutare l'utente nel diversificare le sue annotazioni. Ogni volta che l'utente riclicca il pulsante "Draw In" la dashboard e le annotazioni scompariranno, rendendo il video pulito in caso di future annotazioni.

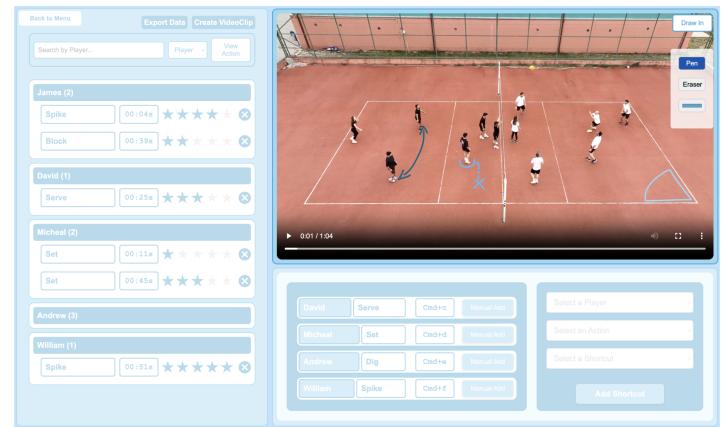


Figura 4.13: Video Player

Export & Create Videoclip

Nell'area sovrastante al componente *Tagged Events*, precisamente sopra alla search bar, sono collocati due pulsanti di seguito descritti:

- **Export Project:** consente di esportare il progetto, permettendo all'utente di salvarlo in locale e riutilizzarlo in futuro. In realtà questo bottone non è l'unico che permette di esportare il progetto, infatti se l'utente decide di tornare al menu principale gli verrà richiesto se vuole esportare il progetto prima di abbandonare la pagina. Il progetto viene esportato scaricando sul dispositivo dell'utente un file JSON contenente lo stato dell'app fino a quel momento.
- **Create Videoclip:** consente di creare videoclip in maniera automatica elaborando gli eventi taggati in precedenza. In particolare viene richiesto all'utente di selezionare la durata media delle clip che andranno a comporre il video. Questa scelta viene presentata come una barra scorrevole con estremi 1s e 15s, così da lasciare all'utente margine di scelta. Successivamente l'utente cliccando il bottone "Create Clip" viene messo in attesa finché il Back-End non ha elaborato il video. Ogni singola clip è caratterizzata dal testo "Giocatore - Azione" di quell'evento taggato. Una volta terminato il processo, la Web App scarica automaticamente il video risultante sul dispositivo dell'utente.

Come scritto in precedenza, infine, è presente il pulsante "Back to Menu" in alto a sinistra che permette all'utente di tornare al menu principale in qualsiasi momento.

4.3.3 Analisi AI

L'Analisi AI, seppur ancora in stato embrionale, rappresenta la funzionalità più innovativa dell'app.

Inizialmente una volta scelta questa modalità viene richiesto all'utente di caricare sull'app il video che vuole analizzare e il tipo di analisi AI che vuole applicare. Una volta inseriti i dati richiesti, tramite il bottone "Apply AI", l'utente viene messo in una schermata d'attesa caratterizzata da una scritta centrale "Applying Model", in modo tale da far capire all'utente che VolleyVisionAI sta elaborando il risultato. Una volta che l'output video è stato generato, l'utente viene portato all'effettiva pagina di Analisi AI. Questa pagina differisce in base al tipo di analisi scelta dall'utente, in particolare se ha scelto di riconoscere la traiettoria della palla o identificare i singoli giocatori.

Ball Trajectory

Questa funzionalità utilizza il modello YOLOv9c per riconoscere la palla. Vengono impostati in particolare 2 parametri: *confidence* e *object-type*. Per questo tipo di analisi impostiamo la confidenza a 0.6 e il tipo di oggetto a "32". Queste specifiche servono rispettivamente per evitare falsi positivi e per specificare al modello che deve riconoscere una palla. Successivamente i risultati di Yolo vengono utilizzati da OpenCV per rappresentare graficamente la traiettoria. In particolare, vengono presi gli ultimi 30 frame della palla e per ognuno di essi viene disegnato un punto giallo. Questo insieme di punti creano visivamente la traiettoria della palla.



Figura 4.14: Ball Trajectory

Players Identification

Questa funzionalità permette all'utente di avere un riconoscimento grafico dei giocatori presenti nel video caricato. In particolare vengono utilizzate delle *Bounding Boxes*, ovvero delle aree colorate che delimitano i giocatori. Ognuna di esse è caratterizzata da un colore diverso e da un identificativo numerico. Il modello YOLOv9c permette di identificare i singoli player assegnando "0" all'*object-type*. Successivamente i risultati vengono elaborati da un algoritmo SORT, definito anche nel paragrafo 4.2, che permette di identificare i giocatori utilizzando le loro coordinate, mantenendo così la loro identità.



Figura 4.15: Player Identification

L'interfaccia rimane invariata per entrambe le analisi. Presentando una *User Interface* minimalista composta da un player video, una semplice didascalia che indica il tipo di analisi, il bottone per tornare al menu principale e quello per scaricare il video analizzato.

5 Evaluation

Nella valutazione della Web App sono stati sviluppati alcuni *usability testing* chiedendo ad un allenatore di pallavolo femminile e ad un giocatore di pallavolo (ovvero due categorie di utenti a cui mi rivolgo) di completare dei task. Da qui si è passati dall'osservazione all'analisi del comportamento all'interno dell'applicazione. Dopo questa prima fase, ho chiesto agli utenti di descrivere quali, secondo la loro prospettiva, sono i punti di forza e di debolezza, chiedendogli quali migliorie apportare. Il risultato delle interviste ha permesso di integrare quanto rilevato dall'analisi con i feedback degli utenti.

5.1 Usability Testing

Questo l'elenco delle operazioni che è stato chiesto di svolgere:

- Accedere alla Web App nella funzionalità *Manual*.
- Creare un nuovo progetto, inserendo i dati richiesti a loro scelta.
- Abilitare *Shortcut* a loro piacimento.
- Testare la funzionalità di *Event Tagging*.
- Testare la funzionalità di *Drawing*.
- Creare il videoclip degli eventi taggati.
- Esportare il progetto.
- Ritornare al menu principale.
- Utilizzare la funzionalità *AI* per la traiettoria della palla.

Durante i test di usabilità della Web App gli utenti hanno definito l'applicazione intuitiva e di facile utilizzo. Gli utenti hanno particolarmente apprezzato la possibilità di assegnare valutazioni specifiche a ciascuna azione e di creare videoclip personalizzati. Tuttavia, sono emerse alcune problematiche che successivamente sono state implementate al fine di migliorare ulteriormente l'esperienza d'uso. In particolare:

- **Eliminazione dei *Tagged Events*:** inizialmente, la Web App non consentiva agli utenti di eliminare uno *Tagged Event* attivato erroneamente, rendendo l'interfaccia meno flessibile. Per risolvere questo problema, è stato aggiunto un pulsante "X" accanto a ogni shortcut attivato, permettendo così di eliminarlo facilmente.
- **Pulsante Manuale per fare *Event Tagging*:** dall'analisi del comportamento sull'uso dell'interfaccia è emersa la difficoltà nell'attivare gli *Shortcut* tramite i comandi rapidi da tastiera. Per semplificare ulteriormente l'utilizzo di questa funzione è stato così aggiunto un pulsante "Manual Add" accanto ad ogni *Shortcut* abilitato in modo da consentire di taggare manualmente un evento semplicemente al clic sul relativo pulsante, senza dover ricorrere per forza a combinazioni di tasti.
- **Feedback della Web App:** sempre dall'analisi del comportamento di riuso dell'applicazione è emerso che, durante la creazione videoclip e applicazione del modello AI, gli utenti percepivano una pagina statica e poco reattiva. Da qui la decisione di fornire un feedback visivo chiaro attraverso l'implementazione di un effetto di sfocatura della pagina con colori chiari, accompagnato da

un'icona dinamica al centro dello schermo indicando "Creating Videoclip" o "Applying Model". Questa modifica ha reso maggiormente evidente lo stato dell'operazione in corso, migliorando la percezione di reattività dell'applicazione.

5.2 Feedback e Miglioramenti

Al termine dei test, sono stati raccolti feedback da questo campione di utenti riguardo a miglioramenti da implementare. I feedback degli allenatori hanno rispecchiato quanto individuato in fase di testing: la necessità di semplificare ulteriormente alcune componenti della Web App sopra descritte. Inoltre, sono emerse richieste aggiuntive da parte di entrambi gli utenti:

- **Analisi AI:** Sia l'allenatore che l'atleta hanno espresso l'esigenza di funzionalità *AI* più avanzate. L'allenatore ha suggerito l'introduzione di un sistema di riconoscimento automatico delle azioni per evitare la fase di *Event Tagging* manuale. L'atleta ha proposto invece un analisi *AI* più verticale ai singoli giocatori. In particolare, ha richiesto di poter visualizzare tutti i momenti in cui un determinato giocatore è protagonista di un'azione. Entrambe le proposte sono state accolte e faranno parte degli sviluppi futuri di VolleyVisionAI.
- **Descrizione Clip Video:** L'atleta ha richiesto di poter avere nel videoclip generato una piccola descrizione per ogni clip, così da rendere ulteriormente più chiari i vari momenti del video. Questo miglioramento è stato implementato e ora il *VideoClip* generato presenta un Box in alto a destra con la descrizione "Atleta - Azione" relativa al momento visualizzato.
- **Raggruppamento degli Shortcut:** In origine, la Web App raggruppava gli eventi taggati esclusivamente per atleta e presentava una barra di ricerca per ricercarne di specifici. Il pallavolista ha suggerito di aggiungere la possibilità di filtrare i *Tagged Event* anche per azione, offrendo una visualizzazione più chiara e ordinata. Questa richiesta è stata implementata con l'aggiunta di un pulsante che consente di alternare tra "View by Action" e "View by Player" nella barra di ricerca, permettendo così una visualizzazione adattabile secondo le preferenze dell'utente.

In conclusione, i test di usabilità hanno confermato che VolleyVisionAI è intuitiva e di facile utilizzo, ma hanno anche evidenziato alcune aree di miglioramento. Grazie ai feedback degli utenti, sono state implementate diverse modifiche per rendere l'interfaccia più flessibile e reattiva, migliorando così l'esperienza d'uso complessiva. Inoltre, sono state raccolte richieste di nuove funzionalità che verranno implementate in futuro per arricchire ulteriormente l'offerta della Web App.

6 Conclusioni

In questo capitolo si espone l'analisi degli obiettivi raggiunti e le funzionalità di VolleyVisionAI che ho sviluppato e di come il percorso formativo seguito da studente ha influenzato il processo di progettazione e realizzazione, analizzando lo stato attuale del progetto e concludendo con riflessioni personali sul lavoro svolto e sui risultati ottenuti.

6.1 Raggiungimento degli obiettivi

VolleyVisionAI ha raggiunto gli obiettivi prefissati con successo, completando tutte le fasi necessarie alla realizzazione della Web App. In particolare, dopo un'attenta analisi di mercato, ho definito in fase di progettazione i requisiti chiave da implementare. Successivamente ho creato un prototipo dell'interfaccia con Figma, per poi arrivare alla fase di sviluppo vera e propria. Attualmente l'app è funzionante nella sua versione Beta reperibile alla repository GitHub di FBK come VolleyVisionAI [10].

6.1.1 Funzionalità della Web App

Come descritto nel capitolo 3, le funzionalità sviluppate permettono all'utente di:

- creare, importare ed esportare progetti di analisi video personali.
- creare *shortcut* personalizzati specificando atleta, azione e comando rapido da tastiera.
- etichettare eventi specifici durante l'analisi video utilizzando gli *shortcut* creati in precedenza.
- raggruppare gli eventi etichettati per giocatore o azione.
- filtrare gli eventi etichettati tramite una barra di ricerca.
- assegnare valutazioni a ciascun evento etichettato.
- disegnare e fare annotazioni direttamente sul video tramite una *dashboard* dedicata.
- richiedere all'app di generare un Videoclip contenente le azioni etichettate in precedenza.
- utilizzare l'analisi AI per visualizzare la traiettoria della palla durante le fasi di gioco.
- utilizzare l'analisi AI per identificare automaticamente i singoli giocatori in campo.

6.2 Sviluppi Futuri

VolleyVisionAI è ancora in una fase Beta e sono numerose le aree di sviluppo che potenzialmente possono rendere l'app ancora più completa e funzionale. In particolare, alcuni possibili sviluppi futuri sono descritti dalle seguenti macro-categorie:

- *Gestione utenti*: Sarà necessario implementare un sistema di registrazione e autenticazione per consentire agli utenti di avere il proprio account personale sulla piattaforma. Questo permetterà una gestione dei progetti molto più flessibile e dinamica.
- *Modelli Computer Vision*: Come specificato nel nome della Web App, la direzione principale è l'implementazione di ulteriori funzionalità AI. In particolare, lo sviluppo di un modello di Computer Vision per il riconoscimento automatico delle azioni, evitando così il processo di *Event Tagging* manuale.

- *Open Source*: In accordo con il tutor aziendale Maurizio Napolitano, è stata presa in considerazione la possibilità di rendere il progetto VolleyVisionAI Open Source. Questo permetterebbe di velocizzare la crescita della Web App e, allo stesso tempo, diffonderla il più possibile.
- *Deployment*: Per rendere l'app accessibile a molti più utenti, sarà necessario implementare un sistema di deployment. L'idea è quella di utilizzare inizialmente servizi di cloud hosting e in particolare l'utilizzo di *Vercel* per il Front-End e *Heroku* per il Back-End.

Durante la fase di progettazione tecnica è stata definita un'architettura estremamente scalabile composta da due server: Front-End e Back-End. Anche se il deploy verrà effettuato utilizzando servizi di cloud hosting, a cui verrà delegata in gran parte la gestione delle risorse, le tecnologie utilizzate sono state scelte per garantire un'ottima scalabilità. In particolare *React* e *FastAPI* permettono di creare applicazioni modulari e facilmente estendibili, garantendo un'ottima manutenibilità del codice.

6.3 Formazione

Per la creazione dell'app VolleyVisionAI, lo studio delle tecnologie utilizzate in fase di sviluppo è stato individuale. Il corso universitario di *Ingegneria del Software* mi ha aiutato in fase di analisi e progettazione a definire i requisiti funzionali e non funzionali della Web App. Inoltre, il corso di EyeStudios Academy su *Figma* mi ha permesso di implementare un prototipo interattivo dell'interfaccia utente prima di passare alla fase di sviluppo tecnico. Grazie all'esperienza di tirocinio formativo presso la *Fondazione Bruno Kessler* ho potuto approfondire tutte le fasi dello sviluppo web. In particolare, ho acquisito competenze nell'utilizzo di *React* e *JavaScript*, che sono state fondamentali per la realizzazione della Web App. Anche il corso di *Introduction to Machine Learning* tenuto dalla professoressa Elisa Ricci mi ha aiutato nel comprendere i concetti principali di questo settore, che poi ho utilizzato durante lo sviluppo della Web App.

6.4 Conclusioni Personalì

Il lavoro svolto per realizzare VolleyVisionAI mi ha fornito un'occasione preziosa per applicare e approfondire le conoscenze acquisite durante il percorso universitario. Lo sviluppo di un progetto pratico sarà sicuramente vantaggioso per i futuri impegni lavorativi e rappresenterà un valore aggiunto al proprio portfolio personale. Questo progetto, insieme all'esperienza di tirocinio, ha consolidato la mia passione per lo sviluppo web e per l'intelligenza artificiale, rendendomi consapevole del percorso formativo che intendo intraprendere in futuro.

Bibliografia

- [1] Balltime - volleyball specific. <https://www.balltime.com/>. Ultimo accesso 25/08/2024.
- [2] Catapult - hybrid system. <https://www.catapult.com/>. Ultimo accesso 25/08/2024.
- [3] Dartfish - video analysis software. <https://www.dartfish.com/>. Ultimo accesso 25/08/2024.
- [4] Data-volley 4.0 - volleyball specific. <https://www.dataproject.com/EU/it/Volleyball>. Ultimo accesso 25/08/2024.
- [5] Eyestudios academy - figma. <https://eyestudios.it/corso-figma/>. Ultimo accesso 29/08/2024.
- [6] Hudl - hybrid system. <https://www.hudl.com/>. Ultimo accesso 25/08/2024.
- [7] Longomatch - video analysis software. <https://longomatch.com/en/>. Ultimo accesso 25/08/2024.
- [8] Nacsport - video analysis software. <https://www.nacsport.com>. Ultimo accesso 25/08/2024.
- [9] Sport 4.0. <https://sportthinking.it/2017/11/22/marketing-4-0-e-industry-4-0-nasce-lo-sport-4-0/>. Ultimo accesso 26/08/2024.
- [10] Volleyvisionai - github repository. <https://github.com/andrearichichi/VolleyVisionAI>. Ultimo accesso 28/08/2024.