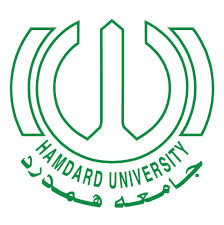
Hamdard University

Department of Computing

Final Year Project



**SpinMaster**

**FYP-035/FL24**

**Software Design Specifications**

Submitted by

**Syed Abraash Ul Haq**

BSAI/F21/2269

**Mubashir Naveed**

BSAI/F21/2433

Supervisor

**Mr. Amir Hussain**

Co Supervisor

**Mr. Waqas Pasha**

**Fall 2021**

**Document Sign off Sheet**

### **Document Information**

|  |  |
| --- | --- |
| **Project Title** | SpinMaster |
| **Project Code** | FYP-035/FL24 |
| **Document Name** | Software Requirements Specifications |
| **Document Version** | 1.0 |
| **Document Identifier** | FYP-035/FL24-SRS |
| **Document Status** | Final |
| **Author(s)** | Abraash Syed Ul Haq & Mubashir Naveed |
| **Approver(s)** | Mr. Amir Hussain |
| **Issue Date** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Signature** | **Date** |
| Syed Abraash Ul Haq | Team Lead |  |  |
| Mubashir Naveed | Team Member 1 |  |  |
| Mr. Amir Hussain | Supervisor |  |  |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
|  | 1.0 |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Definition of Terms, Acronyms, and Abbreviations**

|  |  |
| --- | --- |
| **Term** | **Description** |
| **AI Model** | A computational model built using machine learning techniques to solve specific tasks like shot classification, movement analysis, and rule detection. |
| **AI** | **Artificial Intelligence**: A branch of computer science that focuses on creating machines capable of intelligent behavior, such as learning and problem-solving. |
| **API** | **Application Programming Interface**: A set of rules that allows different software applications to communicate with each other. |
| **AWS** | **Amazon Web Services**: A comprehensive cloud computing platform provided by Amazon, offering storage, computing power, and machine learning tools. |
| **CV** | **Computer Vision**: A field of computer science that enables machines to interpret and understand visual information from the world, such as images and videos. |
| **GPU** | **Graphics Processing Unit**: A hardware component specialized in performing complex calculations required for tasks like rendering graphics and processing data in parallel. |
| **IDE** | **Integrated Development Environment**: A software application that provides tools for writing, testing, and debugging code in one interface. |
| **ML** | **Machine Learning**: A subset of AI that involves algorithms and statistical models that allow systems to improve their performance on tasks through experience. |
| **RACI** | **Responsible, Accountable, Consulted, and Informed**: A matrix that defines the roles and responsibilities of team members for project tasks. |
| **SQL** | **Structured Query Language**: A programming language used for managing and querying data in relational databases. |
| **UI** | **User Interface**: The interface through which users interact with the system, consisting of the visual elements and controls. |
| **Verification and Validation (V&V)** | A methodology used to ensure that a system meets its requirements and performs its intended functions effectively. |

Table of Contents

**1. Introduction**

1.1 Purpose of Document

1.2 Intended Audience

1.3 Document Convention

1.4 Project Overview

**2. Overall System Description**

2.1 Project Background

2.2 Problem Statement

2.3 Project Scope

2.4 Not in Scope

2.5 Project Objectives

2.6 Stakeholders & Affected Groups

2.7 Operating Environment

2.8 System Constraints

2.9 Assumptions & Dependencies

**3. External Interface Requirements**

3.1 Hardware Interfaces

3.2 Software Interfaces

3.3 Communications Interfaces

**4. System Functions / Functional Requirements**

4.1 System Functions

4.2 Use Cases

4.3 Description of Use Cases

**5. Non-Functional Requirements**

5.1 Performance Requirements

5.2 Safety Requirements

5.3 Security Requirements

5.4 Reliability Requirements

5.5 Usability Requirements

5.6 Supportability Requirements

5.7 User Documentation

**6. References**

# Introduction

In the dynamic world of table tennis, precision, strategy, and adaptability are crucial for success. Leveraging advanced computing techniques, this project aims to revolutionize how players analyze and improve their game. By utilizing computer vision (CV) and machine learning (ML), the system will track player movements, ball trajectories, and shot accuracies from game recordings.

This data-driven approach enables the identification of each player’s unique style, strengths, and weaknesses. Inspired by the concept of a comprehensive guide akin to the Pokémon Pokédex, this innovative tool will provide detailed profiles and strategic insights for the top 100 players globally, as well as personalized feedback for individual users.

The ultimate goal is to equip players with the knowledge and strategies needed to enhance their performance and outmaneuver their opponents, making table tennis training more intelligent and effective than ever before.

**1.1 Purpose of Document**

The purpose of this document is to provide a comprehensive Software Design Specification (SDS) for the SpinMaster Intelligent Table Tennis Analytics It outlines the design and architecture of the software, including its functionalities, system requirements, and design methodology.

**1.2 Intended Audience**

The intended audience for this document includes sports coaches, table tennis player software developers involved in sports analytics. It also targets potential collaborators, such as AI and CV experts, to contribute to the development process.

Additionally, this document serves as a guide for project team members, including designers, developers, and researchers, to understand the system’s objectives, requirements, and deliverables.

**1.3 Document Convention**

This document uses:

* **Font:** Arial and book antiqua, size 12 for text, size 14 for headings.
* **Notations:** UML diagrams for system design, ER diagrams for database design.

#### **1.4 Project Overview**

The system employs advanced technologies to analyze recorded video footage of table tennis matches. Key features include tracking player movements, classifying shot types, detecting match events, and visualizing performance metrics.

Unlike real-time systems, this project focuses on generating detailed reports post-match, offering in-depth insights and actionable recommendations.

#### 

#### 

#### 

#### 

# Overall System Description

## **2.1 Project Background**

Table tennis is a game of precision and strategy, requiring players to adapt quickly to dynamic scenarios. Traditional training methods often rely on subjective analysis, limiting opportunities for players to identify and address weaknesses effectively.

The SpinMaster project leverages cutting-edge computer vision (CV) and machine learning (ML) technologies to transform gameplay analysis, offering players, coaches, and analysts a data-driven approach to improve performance.

## **2.2 Problem Statement**

Existing tools for table tennis analysis lack the ability to provide granular insights into player movements, shot accuracy, and ball trajectories. Coaches and players need an intelligent system capable of delivering actionable feedback to improve training efficiency and match performance. The SpinMaster project addresses these gaps by providing a comprehensive analytics platform.

## **2.3 Project Scope**

* The project focuses on analyzing pre-recorded table tennis matches to identify player strengths, weaknesses, and strategies. Key features include:
* Tracking player movements and ball trajectories.
* Analyzing shot accuracy and gameplay patterns.
* Generating performance profiles for players.
* Providing actionable insights to enhance training methods.
* The system will not include real-time analysis in its initial phase but will emphasize scalability for future enhancements.

## **2.4 Not in Scope**

* Real-time match analysis or feedback.
* Integration with augmented reality (AR) systems.
* Predictive modeling of match outcomes.
* Hardware development for recording or tracking.

## **2.5 Project Objectives**

* To design a scalable and modular system for table tennis analysis using CV and ML.
* To provide detailed player profiles, including strengths, weaknesses, and actionable strategies.
* To enhance training methodologies by offering data-driven insights.
* To ensure fairness by detecting and analyzing unjust play styles.
* To build a user-friendly interface for seamless interaction and interpretation of data.

## **2.6 Stakeholders & Affected Groups**

1. Primary Stakeholders:

* Players: Benefit from detailed analytics to improve performance.
* Coaches: Use data insights to tailor training sessions.

1. Secondary Stakeholders:

* Sports Analysts: Leverage data for match commentary and reporting.
* Developers and Researchers: Enhance the system with community-driven improvements.

## **2.7 Operating Environment**

The system will operate on:

* Data Source: High-quality pre-recorded match videos meeting resolution and frame rate standards.
* Computing Environment: Cloud-based infrastructure with GPU acceleration for ML tasks.
* End-User Platform: A web-based interface accessible via browsers on desktops or mobile devices.

## **2.8 System Constraints**

SpinMaster is designed to offer a user-friendly experience with simple navigation and a clean, minimalist interface for easy video uploads and match analysis. The platform will be responsive, ensuring a consistent experience across devices. Accessibility features include screen reader support, keyboard navigation, and high-contrast text for better readability.

The system will optimize the processing of pre-recorded videos using cloud-based infrastructure and machine learning models to analyze data efficiently. It will comply with privacy regulations, encrypt sensitive data, and implement role-based access for security.

Scalability is ensured with cloud services and independent scaling of features like video analysis. User feedback will drive continuous improvements, with updates based on data and community engagement to refine the platform.

## **2.9** .**Assumptions & Dependencies**

The development and functionality of the Advanced Table Tennis Analytics System are based on several key assumptions:

**2.1.1 Video Quality and Consistency**

It is assumed that high-quality video footage with a stable frame rate will be available for analysis. Any variations in video resolution or frame rate could negatively impact the accuracy of player and ball detection. Consistent video quality is vital to ensure reliable results from the system.

**2.1.2 Court and Camera Calibration**

The system is expected to undergo an initial manual calibration to account for table dimensions, camera angles, and player positioning. This step is crucial for accurate spatial measurements and event detection throughout the match.

**2.1.3 Hardware Requirements**

The system’s hardware needs to meet specific performance criteria to handle intensive tasks such as video processing and machine learning operations. High-performance hardware, such as GPUs or dedicated computing infrastructure, may be necessary for processing large datasets efficiently.

**2.1.4 Machine Learning Model Training**

The system relies on a labeled dataset of table tennis shots and events for training and validating the machine learning models. The quality, diversity, and size of this dataset will directly influence the accuracy of the system’s predictions and analysis.

**2.1.5 Cloud Storage and Processing**

The project will use cloud services like AWS S3 for storing large video files and analytics data. A robust, scalable cloud infrastructure is essential for managing and accessing large volumes of data efficiently.

**2.1.6 Internet Connectivity**

# Stable internet access is crucial to ensure smooth upload of match footage and retrieval of analytics results. Any interruptions in connectivity could delay the analysis process and prevent timely access to the system’s outputs.

# 3. External Interface Requirements

**3.1 Hardware Interfaces**

**Data Input Devices:**

• Cameras: Minimum 1080p resolution, 60 FPS for accurate tracking.

• External Storage: For transferring recorded footage.

**Processing Devices:**

• Servers/GPUs: High computational power for video analysis and ML algorithms.

• Recommended Hardware: NVIDIA RTX 3060 or equivalent cloud GPU services

(AWS EC2, Google Cloud TPU).

**End-User Devices:**

• Desktop: 8GB RAM, Intel i5 or equivalent.

• Mobile: 4GB RAM, quad-core processor.

**Data Flow:**

• Input data to storage, then processing, and finally output to user interface.

• Performance: Accurate data capture, efficient processing, and minimal delay.

.

**3.2 Software Interfaces**

• OpenCV: Detects player movements and ball trajectories in video footage.

• PyTorch: Trains ML models to analyze gameplay and generate insights.

• Flask/Django: Web frameworks for building the user interface.

• Cloud Storage: Centralized storage for video files and processed data.

• NumPy/Pandas: For data preprocessing and analysis.

•

**3.3 Communications Interfaces**

• LAN: Transfers raw video data to local servers.

• Cloud: HTTPS for secure data transfer between components.

• APIs: RESTful APIs for system module interaction (e.g., retrieving player data).

• UI Communication: TCP/IP protocols for real-time feedback and player statistics.

# 4. System Functions / Functional Requirements

## **4.1 System Functions**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ref # | Functions | Category | Attribute | Details & Boundary Constraints |
| R1 | Capture video footage of matches | Evident | Data Input | Video must be recorded at a minimum of 1080p resolution and 60 FPS for accurate analysis. |
| R2 | Process and analyze video footage | Hidden | Processing Speed | Analysis must be completed within 10 seconds per video frame for efficient real-time feedback. |
| R3 | Track player movements and ball trajectories | Hidden | Accuracy | Player and ball tracking should have an accuracy rate of 95% or higher, using OpenCV and machine learning models. |
| R4 | Provide analytics and statistics | Evident | User Interface | Interactive statistics (e.g., shot speed, ball spin) must appear on the user interface within 3 seconds of the analysis. |
| R5 | Store match data and analysis results | Hidden | Data Storage | Processed match data and analytics must be stored in cloud storage with high availability and must sync within 5 seconds of analysis completion. |
| R6 | Allow users to upload and view video footage | Frill | User Access | Uploading video files should complete within 3 minutes for up to 1 GB per file. Users can view results within 5 minutes after upload. |
| R7 | Provide feedback and recommendations based on player performance | Evident | User Feedback | Personalized feedback on gameplay performance must be provided within 5 seconds of analysis completion. |
| R8 | Ensure secure access to user data | Hidden | Security & Compliance | User data must be encrypted, and access must comply with GDPR or equivalent data privacy standards. |
| R9 | Enable real-time processing for multiple matches | Hidden | Concurrent Users | The system must support simultaneous video analysis of up to 500 matches with no delay greater than 3 seconds. |

**System Attributes/ Nonfunctional Requirements**

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Details and Boundary Constraints** | **Category** |
| **Match Video Recording** | When a match video is uploaded, the system will process and display basic match details (e.g., players, date) within 5 seconds. | Optional |
| **Concurrent User Load** | The platform must support a minimum of 1,000 users viewing match analytics simultaneously without noticeable degradation in performance. | Mandatory |
| **System Availability** | The platform must maintain an uptime of 99.9%, allowing less than 43 minutes of downtime per month. | Mandatory |
| **Data Backup** | Daily backups of match data and analytics must be performed, with a maximum recovery time of 15 minutes in case of failure. | Mandatory |
| **User Interface** | The user interface will be graphical, web-based, and responsive for both desktop and mobile environments. | Optional |
| **Accessibility** | The system must comply with WCAG 2.1 accessibility standards to support users with disabilities. | Mandatory |
| **Match Analysis Processing** | The system must process and display match analytics (e.g., ball speed, player movements) with a success rate of 99.5%. | Mandatory |
| **Error Recovery** | The system must log and recover from minor errors without user disruption, with downtime for critical errors capped at 15 minutes. | Mandatory |

## **4.2 Use Cases**

### 4.2.1 List of Actors

### **User (Player/Coach):**

* Uploads match videos for analysis.
* Interacts with match data and analytics for performance insights.
* Provides feedback and suggestions on the analytics results.
* Manages their profile, including match history and personal details.

**System (SpinMaster Table Tennis Analytics System):**

* Handles user authentication and session management.
* Processes and analyzes uploaded match videos using computer vision and machine learning.
* Provides real-time performance metrics, such as ball speed, player movement, and shot accuracy.
* Displays match insights, including player statistics, event highlights, and trends.
* Allows users to save and view past analytics results.

### **4.2.2 List of Use Cases**

**Upload Match Video**

* Users upload video footage of table tennis matches for analysis by the system.

**Log In**

* Enables users to access their account using registered credentials to view or upload new match data.

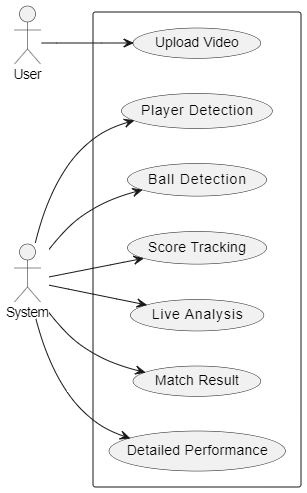
**View Match Analytics**

* Users can view performance metrics such as ball speed, player movement, shot accuracy, etc., for an uploaded match.

**Analyze Player Performance**

* The system generates insights on individual player performance, including shot success rates and positioning trends.

### Use Case Diagram



### Description of Use Cases

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Section | | Main | | | | |
| Name | | Analyze Match | | | | |
| Actors | | Player/Coach | | | | |
| Purpose | | Capture match data and generate performance analytics. | | | | |
| Description | | A player or coach uploads a video, and the system provides detailed match analytics. | | | | |
| Cross References | | Functions: R1.1, R1.2  Use case: The player or coach must have a recorded match video ready for analysis. | | | | |
| Pre-Conditions | | * The player or coach is registered and logged into the system. * The player or coach has uploaded a match video for analysis. | | | | |
| Successful post-conditions | | * The system processes the video and generates match analytics. * The player or coach receives a detailed performance report. | | | | |
| Failure post-conditions | | * Failed video processing leaves the match data uncaptured. * The player or coach is notified and prompted to retry or upload another video.   . | | | | |
|  | |  | | | | |
| Typical Course of Events | | | | | | |
| Actor Action | | | | | System Response | |
| 1 | Player or coach requests analytics. | | | | 1 | System generates and displays a detailed performance report. |
| 2 | Player or coach reviews the report. | | | | 2 | System stores the analytics for future reference or download. |
| 3 | Player or coach confirms the upload. | | | | 3 | System generates and displays a detailed performance report. |
| 4 | Player or coach uploads a match video. | | | | 4 | System stores the analytics for future reference or download. |
| Alternative Course | | | | | | |
| Step 3: | | | | Invalid video format or corrupted file uploaded. (Actor action) | | |
| Step 4: | | | | System indicates an error and prompts the user to re-upload a valid video file. (System response) | | |
|  | | | | | | |
| Section | | | | Video Upload and Processing | | |
| Typical Course of Events | | | | | | |
| Actor Action | | | | | System Response | |
| 1 | The user uploads a match video. | | | | 1 | System begins video processing to track player movements and ball trajectory. |
| 2 | System processes the video. | | | | 2 | Video is analyzed for key performance metrics (ball speed, player movements). |
| 3 | System generates analytics report. | | | | 3 | User receives detailed match analytics report with performance insights. |
| 4 | The user reviews the analytics. | | | | 4 | System displays the report on the user interface for easy viewing. |
| Alternative Courses | | | | | | |
| Step 1: | | | User uploads an unsupported video format or corrupted file. (Actor action) | | | |
| Step 2: | | | System prompts the user to upload a valid video format or retry the upload. (System response) | | | |

# 5. Non - Functional Requirements

## **5.1 Performance Requirements**

## This section defines how well the system or device should perform. For example, if it’s a table tennis training system, it might include the speed of ball tracking, accuracy in detecting spins, and real-time feedback latency. It could also involve the system’s responsiveness in a competitive environment.

## **5.2 Safety Requirements**

Safety measures that need to be implemented to ensure the system doesn’t harm users or surrounding equipment. For example, if there are mechanical or electrical components, you’d need to account for proper insulation, the handling of equipment to avoid injury, and any other safety features, such as automatic shutdown or protective covers.

## **5.3 Security Requirements**

If your system involves any data collection or online components, security features such as data encryption, user authentication, and secure data storage need to be mentioned. This might also involve protecting sensitive user data if applicable.

## **5.4 Reliability Requirements**

Describes how dependable the system is expected to be. For example, your system should be able to operate for a certain amount of hours without failure, or it should have fail-safe mechanisms in case of malfunction, like backup systems or error alerts.

## **5.5 Usability Requirements**

Defines how easy it should be for users to interact with the system. For a table tennis project, this could mean intuitive controls, a user-friendly interface, accessibility for people with disabilities, and straightforward setup.

## **5.6 Supportability Requirements**

Details how the system will be supported after deployment. For instance, if there are any maintenance tools, error diagnostics, remote support options, or software updates, these would be covered here.

# 6. References

1. **Kaustubh Milind Kulkarni, Sucheth Shenoy. (2021).**  
   Table Tennis Stroke Recognition Using Two-Dimensional Human Pose Estimation.  
   Available at: [arXiv](https://arxiv.org/abs/2104.09907)  
   **Relevance:** This paper supports the use of pose estimation techniques for player movement analysis and stroke classification, aligning with the SpinMaster system's goal of tracking player movements and providing detailed analytics.
2. **Roman Voeikov, Nikolay Falaleev, Ruslan Baikulov. (2020).**  
   TTNet: Real-time Temporal and Spatial Video Analysis of Table Tennis.  
   Available at: [arXiv](https://arxiv.org/abs/2004.09927)  
   **Relevance:** Provides insights into using computer vision and machine learning for real-time tracking and analysis of ball trajectories, crucial for implementing SpinMaster's ball tracking module.
3. **Zhuo-yong Shi, Ye-tao Jia, Ke-xin Zhang, Ding-han Wang, Long-meng Ji, Yong Wu. (2023).**  
   Design of Recognition and Evaluation System for Table Tennis Players' Motor Skills Based on Artificial Intelligence.  
   Available at: [arXiv](https://arxiv.org/abs/2309.07141)  
   **Relevance:** Highlights the importance of motor skill evaluation using AI, directly supporting the design of SpinMaster's detailed player profiles and skill analysis functionalities.
4. **Kangnan Dong & Wei Qi Yan. (2024).**  
   Player Performance Analysis in Table Tennis Through Human Action Recognition.  
   Available at: [MDPI](https://www.mdpi.com/2073-431X/13/12/332)  
   **Relevance:** Focuses on using deep learning models for action recognition, reinforcing SpinMaster's performance analysis capabilities.
5. **Anum Zahra & Pierre-Etienne Martin (2021).**  
   Two Stream Network for Stroke Detection in Table Tennis.  
   Available at: [arXiv](https://arxiv.org/abs/2112.12073)  
   **Relevance:** Describes methods for stroke detection from video footage, which aligns with SpinMaster’s use case of analyzing gameplay patterns and shot accuracies.
6. **Kaustubh Milind Kulkarni, Rohan S Jamadagni, Jeffrey Aaron Paul, Sucheth Shenoy. (2023).**  
   Table Tennis Stroke Detection and Recognition Using Ball Trajectory Data.  
   Available at: [arXiv](https://arxiv.org/abs/2302.09657)  
   **Relevance:** Supports the analysis of ball trajectory data, a core feature of SpinMaster's ball tracking and gameplay analysis.
7. **Tianjian Zou, Jiangning Wei, Bo Yu, Xinzhu Qiu, Hao Zhang, Xu Du & Jun Liu. (2024).**  
   Fast Moving Table Tennis Ball Tracking Algorithm Based on Graph Theory.  
   Available at: [Nature Scientific Reports](https://www.nature.com/articles/s41598-024-80056-3)  
   **Relevance:** Details efficient tracking algorithms for high-speed ball movements, directly contributing to the ball tracking features in SpinMaster.
8. **Zhong-Kai Wei and Jieh-Ren Chang. (2024).**  
   Automatic Segmentation of Table Tennis Match Video Clips Based on Player Posture Recognition.  
   Available at: [Journal of Advances in Information Technology](https://www.jait.us/articles/2024/JAIT-V15N12-1374.pdf)  
   **Relevance:** Discusses automated video segmentation using posture recognition, relevant for SpinMaster's feature to extract actionable insights from match recordings.
9. **Guang Liang Yang, Minh Nguyen, Wei Qi Yan & Xue Jun Li. (2024).**  
   Foul Detection for Table Tennis Serves Using Deep Learning.  
   Available at: [MDPI Electronics](https://www.mdpi.com/2079-9292/14/1/27)  
   **Relevance:** Demonstrates how deep learning can be used for rule-based fault detection, aligning with SpinMaster's goal to identify faults like net touches or out-of-bounds shots.
10. **Yu Kit Foo, Xi Li & Rami Ghannam. (2024).**  
    Sensor Fusion and Pose Estimation with a Smart Tennis Ball.  
    Available at: [PubMed Central](https://pmc.ncbi.nlm.nih.gov/articles/PMC11359718/)  
    **Relevance:** Offers methods for integrating pose estimation and data collection, indirectly supporting SpinMaster's aim to utilize advanced pose tracking technologies.