**CHAPTER TWO**

**LITERATURE REVIEW**

**2.1 Introduction**

The literature review for this project on the design and implementation of an Online Assignment Submission Platform includes a comprehensive examination of existing research and advancements in the field of educational technology systems. It focuses on the current trends, problems, and benefits associated with academic assignment management and highlights the possibilities of employing modern technology to enhance assignments submission and grading procedures.

**2.2 Review Of Related work**

Rahman et al. (2018) developed an online assignment submission platform utilizing PHP 7.2 and MySQL 5.7 to streamline academic workflows at Malaysian universities. The system, built with HTML5, CSS3, JavaScript frontend and Apache 2.4 server infrastructure, incorporates PHPMailer for notifications and TCPDF for document generation. The system's core operations include automated submission tracking, plagiarism detection, and a robust feedback mechanism. Students interact through a dashboard where they can view assignment details, submit work in multiple formats (PDF, DOC, DOCX), and track submission status. Lecturers manage assignments through an administrative interface, enabling them to set deadlines, provide rubric-based assessments, and generate performance analytics. The platform features real-time notification systems for both submission confirmations and feedback availability. However, limitations include occasional server timeout issues during peak submission periods and restricted file size uploads, affecting multimedia assignment submissions. The system also lacks integration with popular learning management systems, creating additional work for faculty managing multiple platforms.

Wilson & Thompson (2019) analyzed an online assignment submission platform designed specifically for creative arts education. The system architecture comprises Node.js 12.0 with Express.js for backend processing, MongoDB 4.2 for database operations, and Vue.js 2.6 for frontend interactions. It leverages Amazon S3 for cloud storage, FFmpeg for media processing, and Sharp.js for image handling. The system supports high-resolution image uploads, video submissions, and interactive portfolio creation. Students engage through a gallery-style interface where they can organize multimedia submissions and receive feedback on specific elements of their work. Lecturers utilize annotation tools for precise feedback on visual elements and can conduct virtual critiques through integrated video conferencing. The platform includes version control features allowing students to track project evolution. Key limitations include compatibility issues with certain file formats, storage constraints for large multimedia files, and occasional rendering problems with complex visual content. The system also lacks advanced collaboration features for group projects.

Chen & Liu (2020) presented an AI-enhanced online assignment submission platform focused on programming courses. The system utilizes Python 3.8 with Django 3.1 for backend operations, PostgreSQL 12 for database management, and React.js 16.8 for the frontend interface. The platform integrates TensorFlow 2.3 and scikit-learn 0.23 for AI components, implements an Abstract Syntax Tree parser for code analysis, and uses Git for version control. The system employs automated code testing and intelligent feedback generation through machine learning algorithms. Students submit code through a specialized IDE-like interface that provides real-time syntax checking and preliminary feedback. Lecturers can create test cases, set automatic grading parameters, and override AI-generated assessments when necessary. The platform specifically excels in detecting common programming errors and suggesting improvements, while maintaining detailed submission logs for academic integrity. However, the system faces challenges with complex programming assignments requiring multiple file dependencies, and its AI feedback sometimes lacks the nuanced understanding that human graders provide. Additionally, high server processing requirements during simultaneous code testing sessions can lead to performance bottlenecks.

Patel & Johnson (2021) introduced a blockchain-based online assignment submission platform focusing on submission integrity and authentication. The system implements Ethereum Smart Contracts written in Solidity 0.8.0, utilizes Go 1.16 for backend operations, CouchDB 3.1 for database management, and Angular 12 for frontend interface. It incorporates OAuth 2.0 with JWT for authentication and IPFS for file storage. The system implements smart contracts for assignment submission verification and maintains an immutable record of submission timestamps and feedback history. Students interact through a secure portal requiring two-factor authentication, where they can submit work and receive timestamped confirmation receipts. Lecturers manage assignments through a specialized interface that tracks submission modifications and provides tools for detecting unauthorized collaborations. While the platform excels in security and verification, it faces challenges with large file handling due to blockchain storage limitations and experiences slower processing times compared to traditional systems. The technical complexity also creates a steeper learning curve for users unfamiliar with blockchain technology.

Williams (2021) introduced an online assignment submission platform incorporating artificial intelligence for automated grading using Django REST framework 3.12 for the API backend, TensorFlow 2.5 for machine learning operations, and React Native for cross-platform mobile access. The system employs PostgreSQL for structured data and MongoDB for handling unstructured data, with Redis for caching and session management. Students submit assignments through a mobile-responsive interface that supports offline mode with automatic synchronization. The platform features AI-powered instant feedback generation, plagiarism detection using natural language processing, and automated grading suggestions for objective questions. Lecturers can override AI-generated grades, customize grading rubrics, and track student performance trends through interactive dashboards powered by D3.js. However, the system faces limitations in accurately grading subjective responses and requires significant computational resources for AI operations.

Roberts & Kumar (2022) developed an online assignment submission platform specifically for computer science education using Go 1.18 for the backend, CockroachDB for distributed SQL operations, and Angular 14 for the frontend interface. The system implements Docker containers for isolated code execution, Jenkins for continuous integration testing, and GraphQL for efficient data queries. Students interact with an integrated development environment that supports multiple programming languages, provides real-time syntax highlighting, and offers automated unit testing. The platform features intelligent code analysis tools that detect potential bugs, suggest optimizations, and identify coding style violations. Lecturers can create automated test cases, track student progress through git-style version control, and generate detailed performance reports. However, the system experiences performance bottlenecks during concurrent code compilation and faces challenges with complex project dependencies.

Anderson & Martinez (2022) created an online assignment submission platform specifically designed for mathematics courses using C# .NET Core 6.0 backend, MS SQL Server 2019 for database management, and Angular 13 for the frontend. The system incorporates MathJax for mathematical notation rendering and Computer Algebra System (CAS) integration through Wolfram Alpha API. Students can input mathematical equations through a LaTeX editor or a graphical equation builder, while receiving instant validation of mathematical expressions. Lecturers can create parametrized questions, generate multiple versions of the same assignment with different numerical values, and provide step-by-step feedback using an interactive mathematics editor. The platform's automatic grading system can evaluate symbolic mathematics and verify proof steps. Despite its sophisticated mathematical capabilities, the system struggles with complex mathematical diagrams and lacks support for certain advanced mathematical notations.

Kim & Park (2022) constructed an online assignment submission platform focused on language assessment using Python Flask 3.0 for the backend, MongoDB Atlas for cloud database management, and Next.js 13 for the frontend interface. The system incorporates OpenAI's GPT-3 API for language processing, AWS Transcribe for speech-to-text conversion, and WebRTC for real-time audio communication. Students engage through an intuitive interface that supports multi-modal submissions including text, audio recordings, and video presentations. The platform features automatic pronunciation assessment, grammar checking, and vocabulary analysis tools. Lecturers can create customized rubrics, provide audio feedback, and track student progress through detailed analytics dashboards powered by Grafana. The system automatically generates progress reports and identifies areas where students need additional support. However, it faces challenges with accent recognition accuracy in non-native English speakers and experiences occasional latency issues during peak usage periods.

Thompson & Garcia (2023) implemented an online assignment submission platform specializing in group project management using Ruby on Rails 7.0 for backend operations, PostgreSQL 14 for database management, and Vue.js 3 with Vuex for state management. The system utilizes Socket.io for real-time collaboration, Git for version control integration, and AWS S3 for file storage. Students collaborate through project workspaces featuring real-time document editing, task assignment boards, and progress tracking tools. The platform implements a unique contribution tracking algorithm that measures individual student participation through various metrics including code commits, document edits, and peer evaluations. Lecturers can monitor group dynamics, intervene in team conflicts, and assess both individual and group performances through comprehensive analytics. Despite its robust features, the system struggles with merge conflicts during simultaneous editing and faces challenges in handling large multimedia project files.

Yamamoto & Chen (2023) designed an online assignment submission platform focused on multimedia project submissions using Spring Boot 2.7 for the backend, Apache Cassandra for distributed database management, and Svelte for the frontend interface. The system incorporates FFmpeg for media processing, WebAssembly for browser-based video editing, and CloudFront CDN for content delivery. Students can submit various multimedia formats including videos, animations, and interactive presentations, with built-in compression and format conversion capabilities. The platform features a unique timeline-based feedback system where lecturers can provide time-stamped comments on video submissions and annotate specific frames. Real-time collaboration tools enable peer reviews and group critiques. Despite its innovative features, the system struggles with processing 4K video content and faces storage limitations for large media files.

Lee & Hassan (2023) created an online assignment submission platform emphasizing accessibility and inclusive design using ASP.NET Core 6.0 for the backend, SQL Server 2022 for database management, and React with TypeScript for the frontend. The system implements Web Content Accessibility Guidelines (WCAG) 2.1 standards, integrates screen reader compatibility, and provides multiple input methods for diverse user needs. Students can customize interface layouts, color schemes, and text sizes to match their preferences while submitting assignments in various formats. The platform includes text-to-speech capabilities for feedback reading, keyboard navigation shortcuts, and support for alternative input devices. Lecturers can create accessible content through built-in tools that check for accessibility compliance and suggest improvements. Despite its comprehensive accessibility features, the system faces challenges with complex mathematical notation rendering and multimedia content accessibility.

Zhang & Wang (2023) developed an online assignment submission platform leveraging microservices architecture using Spring Boot 2.6 for backend services, Redis for caching, PostgreSQL for primary storage, and React with TypeScript for the frontend interface. The system implements RabbitMQ for message queuing and Docker for containerization, enabling scalable deployment. Students interact with the platform through a responsive interface that supports real-time collaboration and automatic save functionality. The system's unique feature includes an integrated IDE supporting multiple programming languages, allowing students to write, test, and submit code directly within the platform. Lecturers benefit from automated plagiarism detection powered by JPlag algorithm integration and can provide inline code comments. Performance analytics generated through Elasticsearch and Kibana help identify common student mistakes and learning patterns. However, the system faces challenges with concurrent user load during peak submission times, and the microservices architecture introduces complexity in maintaining data consistency across services.

**2.3 programming Languages and Frameworks in Assignment Submission Platform**