International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 03 | March - 2023 Impact Factor: 7.185 ISSN: 2582-3930

EDUCATIONAL RESOURCE PLANNING AS SAAS IN CLOUD APPLICATION

U. Chindiyababy[1], M. Vennilla[2], P. Sandhiya[3], B. Sanjeev[4], N. Sivakumar[5]

[1]U. Chindiyababy Information Technology & SRM Valliammai Engineering College
[2]M. Vennilla Information Technology & SRM Valliammai Engineering College
[3]P. Sandhiya Information Technology & SRM Valliammai Engineering College
[4]B. Sanjeev Information Technology & SRM Valliammai Engineering College
[5]N. Sivakumar Information Technology & SRM Valliammai Engineering College

---***---

Abstract - An institution or university may utilise educational resource planning as a software application to store, manage, and interpret data. It is applied to reduce work's time complexity. Five key modules make up our educational resource planning, and each module has specific functions. We have an analytics module that is integrated with all the other modules and displays progress based on analysis, as well as an admission module, student database module, service desk module, inventory management module, and module for admissions. We utilise PostgreSQL, a free and open-source relational database management system (RDBMS) that places a strong emphasis on extensibility and SQL compliance, for our database. In order to prevent unauthorised users from accessing, reading, or changing any personal information, PostgreSQL features SSL security that protects sensitive data, such as photographs, documents, and financial information, from capture or change as two systems send and receive it. A private cloud platform called pythonanywhere.com is where we offer our software as SAAS (Software as a Service). We construct our software using the Python-based Django framework. Django is a framework for building web applications. Model View Template (MVT) is the design pattern on which it is based. Due to its capability for quick evolution, the Django is quite demanding.

***Key Words*:** Educational Resource Planning, PostgreSQL, Secured Socket Layer, Software-as-a-Service, Model View Template.

1.INTRODUCTION

1.1 EDUCATIONAL PLANNING

Planning comprises determining the organization's goals and objectives as well as developing strategies for accomplishing them. In plainer terms, this means that planning permeates every man's economic, political, and social activities. Everyone in the world is aware of and engages in planning. All human action is the result of thought, or what is known as strategic planning. Planning is the systematic, deliberate, and forward-looking process of choosing one's future course of action in any area of human effort (Undie, Ekere & Adah, 2011). Every planning process is often done with the goal of maximising the use of the resources at hand to achieve a set of established objectives. The process of creating a set of decisions concerning educational enterprises in a way that the goals and objectives of education will be appropriately achieved in the future with the available resources is known as educational planning. How to best use the existing human and material resources to accomplish educational objectives is the subject of educational planning. It is an ongoing process that is concerned with where to go and the best way to get there. The following are some justifications for why educational planning is essential in every school setting. It offers a regional setting within which the educational system can create and implement plans. It provides a standard against which actual performance can be evaluated and assessed. It is essential for assisting with error avoidance or spotting overlooked chances. It makes it possible for educational managers to more clearly comprehend what they want to accomplish, how to accomplish it, and when it may be accomplished. These arguments support the claim that educational planning entails the process of determining educational needs, the course that education should take, and the logistics of putting plans into action. To emphasise even further, creating a thorough plan won't ensure success; only its efficient execution can. For the achievement of educational goals at all levels of education, it is therefore required to plan education, put plans into action, and evaluate the planning process. The goal of educational planning is to make sure that a number of tasks leading to the attainment of predetermined goals for educational development are carried out in a systematic manner. The use of rational and systematic analysis of the past or present educational production function with the goal of outlining potential reforms that would increase the effectiveness and efficiency of education production is another key component of educational planning.

1.2 EDUCATIONAL REQUIREMENT

Planning for education is anticipating future educational needs and making the necessary preparations. In order to ensure that the planning process is successful, numerous elements must be taken into account. Planning for education involves developing policies, establishing goals, conducting feasibility studies, and essentially forecasting the future. This makes educational planning the appropriate reaction to the broader tendency towards or planning for a nation's overall development. To establish projections for future educational progress, including estimations of the human, physical, and financial resources required to fulfil objectives, educational planning therefore entails gathering, evaluating, and sharing data. The Implementation Committee of the on Education's definition of educational planning is an ongoing process of collecting and evaluating data with the goal of informing decision-makers on how successfully the educational system is carrying out its objectives, and in particular, how the cost effectiveness of educational programmes and individual items can be increased.

1.3 SERVICE DESK

IT service desks serve as hubs of communication where users may ask for assistance and receive IT support. They are also intended to meet more general business needs and improve user tech experience at work. The user community, which is often comprised of the organization's employees and other stakeholders, and other service management are communicated with through IT service desks. They aid with problem management, keeping track of third party contacts, logging modification requests, and managing software licences. The terms "help desk" and "service desk" are sometimes used interchangeably by corporations. Although both products have certain similarities, IT help desks and IT service desks serve different purposes and have different objectives inside a company. Help desks are more tactical and assist in resolving the needs, problems, and incidents of end users. A help desk is typically reactive in nature and is designed to offer prompt, effective problem-solving as it occurs. They may be integrated into bigger service desk operations or run independently. offering fundamental event and service request management while using tracking for incoming incidents. offering assistance at several levels, increasing problems as necessary. offering end consumers simple self-service choices. Service desks serve as a central point of contact for IT assistance, but they also perform more strategic tasks. Instead of concentrating solely on meeting user demands, they typically address broader company needs.

1.4 INVENTORY MANAGEMENT SYSTEM

The products or materials in inventory are those that a company plans to resell to customers for a profit. Tracking inventory from producers to warehouses and from these locations to a point of sale is inventory management, a crucial component of the supply chain. Having the right products at the right location at the right time is the aim of inventory management. For this, inventory visibility is necessary in order to know how much to order, when to order it, and where to put the stock. The fundamental steps in inventory management are as follows: Ready-to-sell items are purchased, delivered to the warehouse, or put directly on the sales floor. stockpiling goods, Inventory is kept until it is required. Materials or goods are moved around your fulfilment network until they are prepared for shipping. The quantity of goods available for purchase is managed by profit from inventory. Orders are filled by pulling finished goods. Customers receive product shipments. Inventory for multichannel order fulfilment operations is frequently dispersed throughout the supply chain. Knowing what inventory you have and where it is situated is known as inventory visibility. To ensure that client orders are fulfilled, shorten shipping turnaround times, and decrease stockouts, oversells, and markdowns, businesses require an accurate view of their inventory. The most valuable asset of a business may be its inventory. Where all the components of the supply chain come together is in inventory management. Inventory carries its own risks, such as the expense of storing and insuring it as well as the possibility of deterioration, theft, and damage. Businesses with intricate manufacturing and supply chains must strike the correct balance between having too much inventory on hand and not enough. Companies assign a number to each item they sell using barcode inventory management systems. The supplier, the size and weight of the item, as well as changeable information like the quantity in stock, can all be linked to the number. In order to track things and give comprehensive product information, RFID inventory management systems wirelessly broadcast a product's identify in the form of a distinctive serial number. The RFID-based warehouse management system may boost productivity, increase inventory visibility, and assure quick delivery and receipt recording. Inventory tracking, among other essentials of good inventory management Know where each item of inventory is in the supply chain.

1.5 STUDENT DATABASE MANAGEMENT

Software called Student Management System is beneficial to both school administration and students. All tasks are completed manually under the current system. It is expensive and takes a lot of time. Our student management system manages a range of activities. It would be quite profitable to develop this application. For the development of the previously existing system, the company doesn't need to spend a lot of money. The only thing that can be done is to set up a development environment under strong supervision. By doing this, we can maximise the usability of the relevant resources. Even after the progress, the organisation won't be in a position to increase its investment. The system is therefore commercially viable.

2. LITERATURE SURVEY

2.1. S. Mansmann and M. H. Scholl, "Decision Support System for Managing Educational Capacity Utilization," in IEEE Transactions on Education, vol. 50, no. 2, pp. 143-150, May 2007, doi: 10.1109/TE.2007.893175..

In order to make decisions in the field of academic planning, a lot of data from many systems must be thoroughly analysed. Deans and department chairmen must make sure that new specialities and programmes are effectively supported in light of the numerous new technology application areas emerging from the fields of electrical engineering, computer engineering, and computer science. The distribution of teaching resources to appropriately support the university's instructional framework is an issue of academic workload management (faculties, degrees, courses, admission policies, teaching workload, etc.). In this study, a technique for determining educational capacity, planning how it will be distributed, and utilising it is presented. This methodology is implemented as a decision support system that enables simulation and evaluation of various ideas and scenarios. The system incorporates data input from pertinent sources into a self-contained data warehouse. By highlighting important elements and dependencies in the data, a graphical client front-end makes sure that the output is presented to the decision-makers in an appropriate manner. By using the system as a "on-the-fly" decision-support tool, policymakers can significantly speed up planning processes, gain a deeper understanding of the facts and the methods behind them, and ultimately improve academic administration.

2.2. J.E. Tate, T. J. Overbye, J. Sebestik and G. C. Reese, "Interactive Lessons for PreUniversity Power Education," in IEEE Transactions on Power Systems, vol. 23, no. 3, pp. 824-830, Aug. 2008, doi: 10.1109/TPWRS.2008.920734

The continual need to train its future workers is a major issue for the electric power sector. Although attending college is an important step in this process, research has shown that many bright students are not aware of potential employment in the power industry. In middle and even high school, many students lose interest in math and science. In order to help meet these demands, this article offers lesson plans and related applets that were created through collaboration between electric power researchers and education experts. To involve pre-university students in the power field, two courses have been created thus far. The first unit, Power and Energy in the Household, presents numerous sample loads to show the effects of operating various appliances while also serving as an introduction to the concepts of power and energy. Energy Star appliances, incandescent lighting, and compact fluorescent lighting are all included, showing that special attention is being paid to environmental concerns.

2.3. M. A. Layek, T. Chung and E. Huh, "Adaptive Desktop Delivery Scheme for Provisioning Quality of Experience in Cloud Desktop as a Service," in The Computer Journal, vol. 59, no. 2, pp. 260-274, Feb. 2016, doi: 10.1093/comjnl/bxv116.

Desktop as a Service (DaaS) enables customers to link their on-desk computers, laptops, and other mobile devices with their virtual remote desktops. It is crucial for service providers to understand how users and customers view their system in order to give the greatest possible user experience while still achieving their business goals. This paper discusses desktop delivery solutions for DaaS that are QoE aware. We first establish one-to-one exponential relationship models between quality of service (QoS) and various objective qualities in desktop delivery, where screen sizes in pixels unit are taken as the QoS, based on our prior studies.. Later, these models are brought together to create an integrated quality of experience (QoE) model, which can calculate the overall QoE based on specified QoS parameters. We suggest an adaptive desktop delivery strategy for DaaS using the QoE model that was derived.

2.4. K.Djemame, D. Armstrong, J. Guitart and M. Macias, "A Risk Assessment Framework for Cloud Computing," in IEEE Transactions on Cloud Computing, vol. 4, no. 3, pp. 265-278, 1 July-Sept. 2016, doi: 10.1109/TCC.2014.2344653.

To maximise the quality of service (QoS) they provide and reduce the number of SLA violations, cloud service providers need well-balanced infrastructures. These providers offer access to their resources through formal service level agreements (SLAs). This article focuses on one particular use of risk assessment in cloud computing: techniques inside a framework that cloud service providers and service users can use to evaluate risk throughout service setup and operation. It outlines the many service lifecycle stages where risk assessment occurs as well as the relevant risk models that have been created and put into practise. The effect of risk on architectural elements is also described, with a focus on holistic management assistance during service operation. The experimental evaluation of the implementation has demonstrated the effectiveness of the risk assessor, which is already included in a cloud computing toolbox.

3. SYSTEM ARCHITECTURE

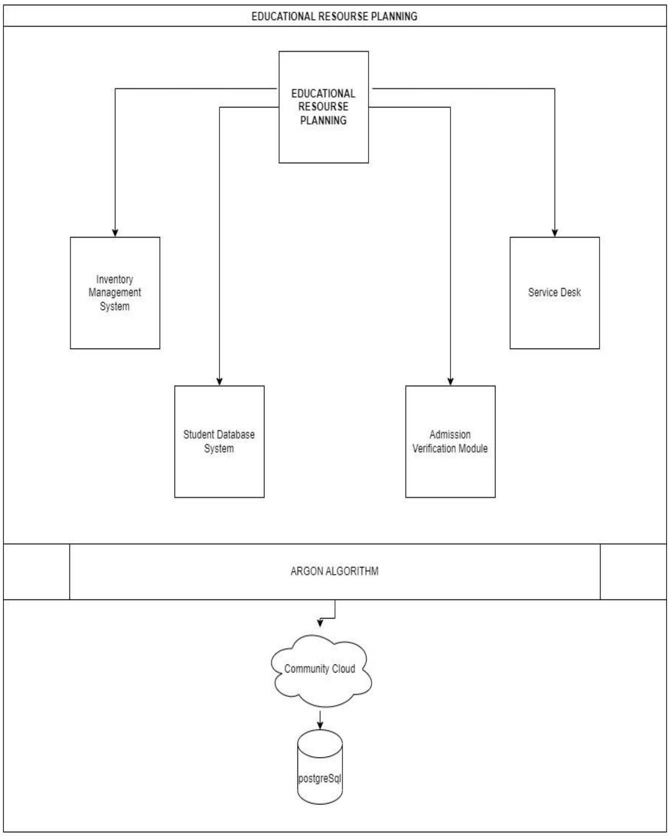

Fig-1: System Architecture of Educational Resource Planning

4. METHODOLOGY

4.1 ARGON

The memory-hard function Argon2 is hard. It has an efficient layout. The maximum memory fill rate and efficient usage of several processing units are its goals, while yet offering protection against trade-off assaults. Argon2 takes advantage of the cache and memory layout of current Intel and AMD processors and is designed for the x86 architecture. Argon2 features two supplementary variations (Argon2d and Argon2i) in addition to one major variant (Argon2id). Because Argon2d uses data-dependent memory access, it is safe from side-channel timing attacks and suited for usage with proof-of-work applications and cryptocurrencies. For password hashing and key generation based on passwords, Argon2i employs data-independent memory access. Due to trade-offs between time and memory, Argon2id performs as Argon2i for the first half of the first run through the memory and as Argon2d for the remaining portion, protecting against side channel attacks and saving money when using brute force. To defend against trade-off assaults, Argon2i does additional passes over the memory. G's internal permutation serves as the foundation for the compression function. It also makes use of a hash function H' with variable length. The main method of authentication on many web services is still passwords. Passwords are often kept in a database on a server in hashed form. As passwords typically have low entropy, attackers frequently obtain these databases and use dictionary attacks on them. Many techniques are employed by protocol designers to address these problems. A random salt value has been added to passwords since the late 1970s to prevent the identification of similar passwords among users and services. The number of times the hash function computations were invoked increased the cost of the password trial for the attacker. Moore's law caused hash function computations to get faster and faster. Password crackers moved to new architectures, where the amortised cost of a multiple-iterated hash function is substantially lower, in the meantime, such as FPGAs, multiple core GPUs, and dedicated ASIC modules. It was rapidly discovered that these new environments work well when the computation is almost memoryless but have trouble when the memory size is huge. In response, the defenders created memory-hard functions, which need a lot of memory to compute. Scrypt's password hashing algorithm is an example of this function. The use of password hashing algorithms is not limited to these. They can be used to derive keys from sources with low entropy. If a cryptocurrency's designer wants to discourage the usage of GPUs and ASICs for mining and encourage the use of ordinary PCs, memory-hard techniques are also welcome. It turned out to be a challenging task to create a memory-hard function. Many cryptographic problems that appear to require a lot of memory can really be solved with a time-memory tradeoff, as has been known since the early 1980s. where the opponent can work on quick hardware with little memory and trade memory for time. This applies to password-hashing techniques and indicates that, despite the higher expense, password crackers can still be deployed on specialised hardware. For instance, the scrypt function permits a straightforward tradeoff where the time-area product nearly always remains constant. Our initial proposal to PHC was Argon. It is a multipurpose hash function that is designed to be as resilient as possible to tradeoff attacks so that even a tiny memory reduction would result in large computational and time penalties. Any memory-intensive computation, including key derivation and password hashing, can be performed with argon.

4.2 ARGON VARIANT

Moreover, Argon2 is a mode of operation over the compression function G with fixed input length and the hash function H with variable input length. The BLAKE2b function e P-byte zero string can be used with any function H, even though Argon2 can potentially be used with any function H as long as it allows outputs up to 64 bytes. Message string p, a password used in password hashing software. It needs to be no longer than 2^(32-1) bytes.nonce s, a salt used in password hashing software. It needs to be no longer than 2^(32-1) bytes. It is advised to use 16 bytes when hashing passwords. Every password should use a different salt.

4.3 ARGON2 OPERATION

Argon2 employs an internal hash function Hx(), with x being the output length in bytes, together with an internal compression function G with two 1024-byte inputs and a 1024-byte output. The BLAKE2b ([BLAKE2], Section 3.3) function, which accepts as inputs, is used in this instance to apply Hx() to the string A. d is the length of d in bytes, and ll is the padding of A to a multiple of 128 bytes. G's internal permutation serves as the foundation for the compression function. It also makes use of a hash function H' with variable length.For every I between (and including) 0 and (not including) p and for all j between (and including) 2 and (not including) q, compute B[i][j]. Slicewise computation MUST be used. (Section 3.4): Blocks from slice 0 are computed first for all lanes (in any order), followed by blocks from slice 1, etc. For Argon2d, Argon2i, and Argon2id, the block indices l and z are computed for each I j in a different way.If t passes is greater than 1, step 5 is repeated. For every I ranging from (and including) 0 to (not including) p and for all j ranging from (and including) 1 to (not including) q, we compute B[i][0] and B[i][j]. Blocks, however, are calculated differently because the previous value is XO. with the new, red The final block C is calculated as the XOR of the last column following t steps of iteration.

4.4 INDEXING

We further divide the memory matrix into SL vertical slices to support parallel block computing. A segment, which has a length of q/SL, is the point where a slice and a lane intersect. The same slice's segments can be computed in parallel without referencing different blocks. Each additional block may be referred. Argon uses a design methodology that has been successful in creating cryptographic hash functions. Once the memory is full, a series of identical rounds are performed on the internal state. The rounds alternate block permutations that act column-wise and produce diffusion with nonlinear transformations that operate row-wise and produce confusion. Internal diffusion is to be maximised and time memory tradeoffs are to be avoided while designing nonlinear transformations. The well-known RC4 state permutation is used to describe how the data-dependent permutation component works. Because Argon2 is so resistant to ranking tradeoff attacks, it is far more challenging to efficiently optimise on FPGAs. Although the incorporated RAM blocks in more current FPGAs, memory bandwidth is still a problem.

4.5 ARGON2ID

64-bit multiplications are integrated with the modular additions in GB. The only modification from the original BLAKE2b design is multiplication. Due to parallelism and pipelining, this decision allows for nearly the same running time on CPUs while increasing the circuit depth and hence the running time of ASIC implementations. Argon2d is designed to work best in situations where the adversary does not have regular access to system memory or the CPU; as a result, they are unable to use timing information to launch side-channel attacks or recover the password more quickly using garbage collection. These configurations are more typical for cryptocurrency mining and backend servers. We recommend the following settings for practise. Mining a cryptocurrency takes 0.1 seconds on a 2 GHz CPU with a single core, an Argon2d, two lanes, and 250 MB of RAM. Argon2id is designed to operate in more realistic environments where the enemy may gain access to the same computer, employ its CPU, or launch cold-boot attacks. We advise using the settings listed below. On a 2 GHz CPU with 4 cores, an Argon2id with 8 lanes, and 4 GiB of RAM, backend server authentication takes 0.5 seconds. Hard drive encryption key derivation takes 3 seconds on an Argon2id with 4 lanes and 6 GiB of RAM running at a 2 GHz CPU utilising 2 cores. 0.5 seconds are needed for frontend server authentication on a 2 GHz CPU with 2 Argon2id cores, 4 lanes, and 1 GiB of RAM. To choose the type and characteristics for practical use of argon2, we advise following the steps below. Use Argon2id with t=1 iteration, p=4 lanes, m=2^(21) (2 GiB of RAM), 128-bit salt, and 256-bit tag size if a uniformly safe solution that is not specialised to your application or hardware is acceptable. The initial suggestion is this. If significantly less memory is available, Argon2id with t=3 iterations, p=4 lanes, m=2^(16) (64 MiB of RAM), 128-bit salt, and 256-bit tag size is a uniformly secure alternative. The second suggested choice is this one. If not, start by choosing the type y. Choose Argon2id if you are unsure about the differences between the types or if you believe side-channel attacks to be a real issue. Calculate the largest amount of RAM that each call can support, then convert that amount to the parameter m. Calculate the maximum time (in seconds) that each call can tolerate. The salt length can be chosen. All applications require a length of 128 bits, but this can be lowered to 64 bits in cases of space limitations. Choose the length of the tag. For the majority of applications, including key derivation, a length of 128 bits is adequate. Choose longer tags if longer keys are required. Use the memory-wiping option in the library call if side-channel assaults are a real concern or if you're unsure. Execute the type Y, memory m, and p lane scheme with a different t passes count. Determine the highest t such that the running time doesn't go above the allowable amount of time. If it even does for t = 1, then decrease m in line with that. We offer test vectors with comprehensive results (tags). We also offer a few temporary variables for developers' convenience, specifically the first and last memory blocks of each iteration.

4.6 RAFT ALGORITHM

The consensus algorithm Raft is made to be simple to comprehend. Performance and fault tolerance are on par with Paxos. The distinction is that it neatly handles all crucial components required for functional systems and is divided into reasonably isolated subproblems. We anticipate that Raft will open up consensus to a larger audience, and that this larger audience will be able to create a range of consensus-based systems of greater calibre than those that are currently accessible. A basic issue with fault-tolerant distributed systems is consensus. Several servers must concur on values for there to be consensus. The choice individuals make regarding a value is irrevocable once they have made it. While the majority of their servers are online, typical consensus methods advance; for instance, a cluster of five servers can keep running even if two of them fail. If additional servers go down, they stop growing. Sensus generally occurs when replicating state machines are used, which is a common strategy for creating fault-tolerant systems. State machines and logs are present on every server. The component that we wish to make fault-tolerant, similar to a hash table, is the state machine. Even if a small percentage of the servers in the cluster fail, clients will still think they are communicating with a single, dependable state machine. Commands from each state machine's log are used as input. The log in our hash table example might contain instructions like set x to 3. The commands recorded in the servers' logs are agreed upon using a consensus technique. The consensus algorithm must make sure that no state machine will ever apply a different nth command if any state machine applies set x to 3 as the nth command. As a result, each state machine executes the same set of commands, yielding the same set of outcomes and states in the process. A node can remain in any of the three states mentioned above under normal circumstances. The client can only be communicated with by a leader; All requests are forwarded to the leader node from the follower node. A candidate for leadership may solicit votes. A follower exclusively reacts to the leader or the candidate(s). The Raft algorithm breaks time into tiny periods of arbitrary duration in order to maintain these server status(es). A term number, which increases monotonically, is used to identify each term. Every node keeps track of this term number, which is handed along during node-to-node talks. Every term begins with an election to choose the new chief executive. To gain the majority, the candidates ask other server nodes (followers) for their votes. If enough people support the candidate, they elect him or her to lead for the remainder of the term. Split votes occur when a majority is not reached, in which case there is no leader at the end of the term. Thus, there can only be one leader every term.

5. OUTPUT

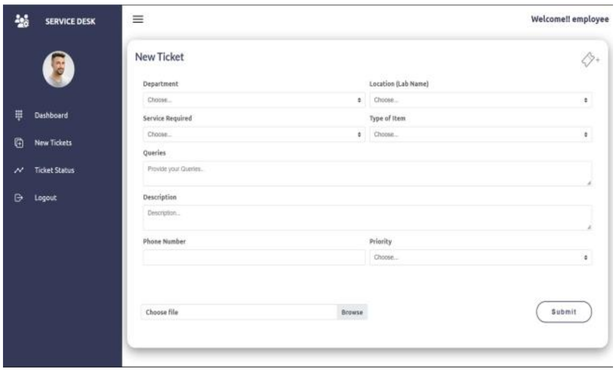

Fig-2: Employee Portal

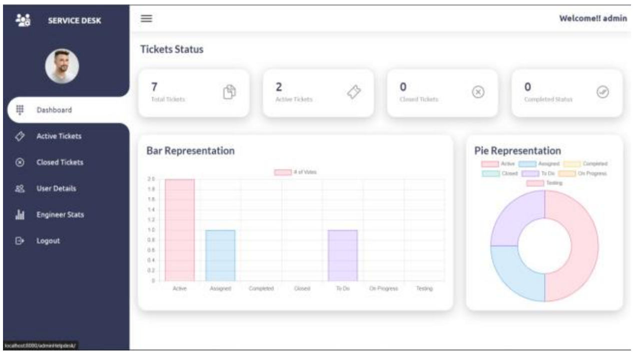

Fig-3: Admin Portal

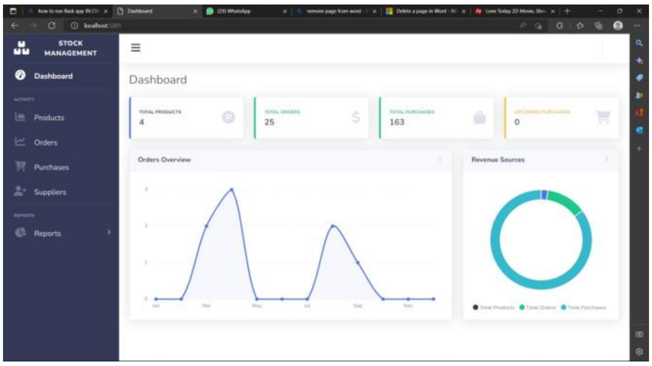

Fig-3: Inventory-Dashboard

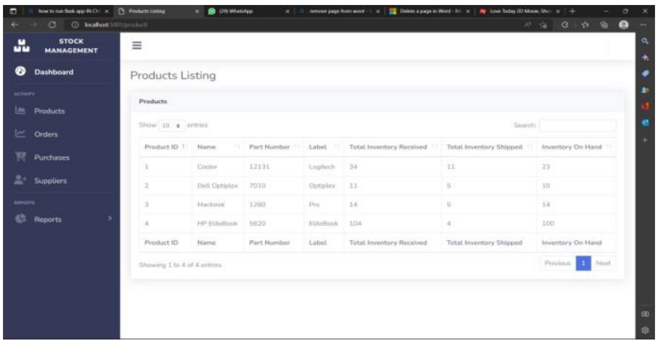

Fig-4: Inventory-Product Listing

6.CONCLUSION

The routine process of managing and setting up efficient educational systems is known as educational management. Every person participating in the system is made aware of what must be done and how it must be done. Every worker performing a job must also be aware of the signals that indicate whether or not the task assigned to them has been successfully done. As a result, educational management has a broad range of activities and tries to integrate society and education. establishing, creating, and maintaining the fundamental educational infrastructure, including schools, colleges, universities, libraries, museums, hostels, etc. upkeep of academic records evaluating student performance and keeping an eye on student behaviour Keeping track of supply inventories for items like office supplies, books, hardware, digital equipment, building infrastructure, lab equipment, etc. scheduling classes, training, exhibits, seminars, and presentations, among other things ensuring order, coordination of work across all departments, and cooperation with educational authorities Create the course materials for lectures, tests, internal assessments, final exams, etc. managing resources, spending, social services, and the general wellbeing of every system user. You may have figured out by now that educational management is a continuous process with a wide range of tasks and responsibilities. Therefore, a balanced mix of constancy and dynamism is the foundation of effective educational management. While educational managers must be constant in their leadership, management, and learning efforts, they must also be dynamic to make sure that all of the activities are correctly coordinated and conducted in parallel. We intend to share the most up-to-date knowledge with education stakeholders worldwide by conducting a thorough review of the fundamentals of educational management.

ACKNOWLEDGEMENT

We are grateful to our internal guide Ms.U.Chindiyababy, M.E (Ph.D.)., Assistant Professor (O.G) for her invaluable guidance and encouragement, this project would not have been successful.

REFERENCES

1. A. Althunibat, B. M. Al-mahadeen, F. Altarawneh and F. A. Al-Qarem, "The Acceptance of using Enterprise Resource Planning (ERP) System in Higher Education: A Case Study of Jordanian Universities," 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), 2019, pp. 315-318, doi: 10.1109/JEEIT.2019.8717451.

2. F. Wang, "Application of multidimensional association rule techniques in manufacturing resource planning system," 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery, 2010, pp. 1433-1437, doi: 10.1109/FSKD.2010.5569197.

3. H. Zhang and Y. Chen, "Research on Design and Compilation of "The Tourist Atlas of Shanxi Province"," 2008 International Workshop on Education Technology and Training & 2008 International Workshop on Geoscience and Remote Sensing, 2008, pp. 263-266, doi: 10.1109/ETTandGRS.2008.269.

4. I. A. Bairagi, A. Sharma, B. K. Rana and A. Singh, "UNO: A Web Application using Django", 2021 3rd International

Conference on Advances in Computing, Communication Control and Networking (ICAC3N), 2021, pp. 1371-1374, doi: 10.1109/ICAC3N53548.2021.9725577.

5. "IEEE Colloquium on 'Resource Scheduling for Large Scale Planning Systems' (Digest No.1993/144)," IEEE Colloquium on Resource Scheduling for Large Scale Planning Systems, 1993, pp. 0_1-.H. Supriyono, M. R. Fitriyan and Muamaroh, "Developing a QR Code-based Library Management System with Case Study of Private School in Surakarta City Indonesia", 2018 Third International Conference on Informatics and Computing (ICIC), 2018, pp. 1-6, doi: 10.1109/IAC.2018.8780424.

6. L. C. Morse, "Using management concepts in planning for distance education," FIE'99 Frontiers in Education. 29th Annual Frontiers in Education Conference. Designing the Future of Science and Engineering Education. Conference Proceedings (IEEE Cat. No.99CH37011, 1999, pp. 12B2/10-12B2/13 vol.2, doi: 10.1109/FIE.1999.841574.
7. M. Gupta and S. Jetty, "Library in Everyone's Pocket With reference to Bundelkhand University App", 2018 5th International Symposium on Emerging trends and Technologies in Libraries and Information Services (ETTLIS), 2018, pp. 79-82, doi: 10.1109/ETTLIS.2018.8485206.

8. V. Pantovic, M. Savkovic and D. Starcevic, "Work in progress - Virtual Business School and Enterprise Resource Planning system integration in Energorpojekt Group," 2008 38th Annual Frontiers in Education Conference, 2008, pp. T4B23T4B-24, doi: 10.1109/FIE.2008.4720590.M. Wan, H. Yao and X. Yan, "Generation of malicious web page samples based on GAN", 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2020, pp. 864-869, doi: 10.1109/TrustCom50675.2020.00116.