**Introduction**: Today’s modern business world relies on gathering and utilizing information in order to serve consumers with better services. As consumer preferences and requirements are constantly changing over time, efficient utilization of information can provide important insight and analytics to find new and innovative business ideas which can help them to survive in the competitive market. Good news is this modern era of science and technology makes things a bit convenient. Storing, retrieving and analyzing information is easier than ever due to the contribution of Information Technology (IT) in the business market.

IT systems can provide automation, insights and analytics which can accelerate the business processes [1]. But it is essential to use and implement IT systems in business with proper plan, design and engineering. Otherwise, it may cause a huge harm to the business. Because although IT systems are used to enhance and improve customer satisfaction, a poor implementation with inappropriate planning can cause poor services which will eventually dissatisfy the customers. And it will affect the business badly.

In this report, a case study is considered based on a failed implementation of IT system in a billion dollar business context and its consequences. The report identifies and critically analyzes the major flaws and misjudgments of the IT infrastructure that was implemented, also proposed an alternative plan in order to avoid the loss that caused by the failure of the IT system.

**Discussion**: In this section, first a brief description of the background of the case study is presented. Then a critical analysis on the situation based on the IT infrastructure and engineering flaws is discussed. Because in order to analyze the IT system, first we need to understand and identify the business processes and customer requirements which were the main focus to implement the new system and re-engineer the previous one. An alternative plan is proposed in the following sub-section to identify what could have improved the situation and avoided the loss along with justifications of the proposed ideas.

* Background of the case study and critical analysis: The case study is based on an organization named Child Support Agency (CSA) which was backed by the Department of Work and Pensions in Great Britain. The case study also mentioned how CSA intended to re-engineer their information system in order to simplify their previous system with new and advanced methods. An American company, Electronic Data Systems (EDA) was awarded the contract to provide and implement their new IT system which was named as CS2 system. But due to poor planning and implementation, this system was criticized by the customers and UK government soon after it was ready and open to use. It was proved by several audits that the system spent way over the budget, got delayed by two years to start and had critical issues with the services which eventually caused dissatisfaction among the customers and a great loss to the company. And as a result of continuous dissatisfaction and loss during the following years, in 2006 CSA was shut down and replaced by Child Maintenance and Enforcement Commission (C-MEC). The critiques claimed that the poor management, implementation and service provided by CS2 system were the main reasons for the poor performance and shutting down the organization.

CSA was formed in 1993 with aims to make sure the support from a non-resident parent towards the financial cost of raising their child; this situation may occur after a divorce or child was born to a single mother. CSA provided all necessary documents and end-to-end services from collecting the applications to payment distribution [2]. Typically, the process starts by receiving and assessing an application from a parent for child support or maintenance. Following the assessments, CSA identifies and locates non-resident parents and confirming paternity with mutual agreements of both parties by calculating the payments and accepting other terms. After all the arrangements CSA used to ensure the payment was done successfully. The agency was also responsible to chase missing payments and collect debts, and pursue parents who did not pay the required support [3].

All these business processes were expected to be done by the new CS2 system. CS2 was expected to receive applications from the clients directly, facilitate them by tracking their applications and providing other information like calculating the payments, account information for collection and disbursement of the payments, agreement policy and terms etc. The system was designed to provide simplified new rules for calculation of the payment amounts. But CSA had no internal technical staff to assure the quality and standards of the design and implementation of CS2 by EDS [2] [3].

This analysis is presented by identifying and pointing out the major engineering and restructuring flaws of CS2 system that eventually led to failure.

* Insufficient requirement analysis: The government wanted new rule or formula to calculate the liability for child maintenance and migrate the whole large and complex information system to the new CS2 system. The first major flaw was misunderstanding and misjudgment of these requirements by EDA. A badly designed system based on insufficient requirement analysis cannot replace such a huge and complex information system. They could not determine the current state of the system and estimate the outcome of migrating and implementing before start developing the new system. And as a result, the project ran seriously over budget and trailed behind the estimated deadline. To catch up the backlogs and cover the losses, they rushed to make progress but the end result was not satisfactory and the product was not up to the mark. After the deployment, the newly introduced simplified formula miscalculated the payments and as a result some parents received little or no maintenance, whilst others were unfairly pursued and faced legal enforcement action against incorrect charges. Loss and customer complaints began to raise due to this fact.
* Rigid design: After the deployment, CSA and EDA constantly worked to repair and align their system to the customer’s requirements. But they had failed to modify or extend the system according to the requirements. This also occurred due to less research on customer requirement on the first place. Also the design was not flexible that it could not be modified anymore and it was conflicting with other features all the time. Developers didn’t think about the flexibility that the system might have to accommodate in the future according to the customers. As a matter of fact, customers were getting unhappier with the services.

Re-engineering an IT system often useful but with a complex and large system it is too risky to apply that all at once. The agency should have decided to restructure by dividing the whole system into separate small projects and develop them in a modular fashion. By this, the design could have been more flexible and could be modified according to the customer’s need. Also it would be easy to find out the problems and fix them. A detailed proposal of a modularized concept to re-engineer the system is discussed in the next sub-section.

* An Alternative Proposal: The main idea of the proposed plan is based on micro-service architecture [4] like dividing and distributing the whole system into a few sub-modules in order to migrate a large and complex information system. These modules are independent to each other and have separate responsibilities. The proposal is to migrate the system module by module, meaning implement and migrate one module and then another. The possible modules with functionalities are discussed below.
* Current system analysis process: First of all, all the functionalities of the current system should be analyzed. By aligning them with the requirements, new blueprint should be drawn with all the necessities. The blueprint should identify all the key requirements and efficient implementation plans.
* The application process: The first module that can be migrated in to a new system is the application process. The purpose of this module is only to receive the application from the clients, track and review them, give feedback to the clients and arrange initial agreements with both parties. This module requires only a platform to submit and a database to store the applications. And this platform is the basic building block of the system which will cost reasonable amount of money.
* Customer requirement collection process: The platform that is developed in the previous module can be used to collect customer’s requirements and align them with the system implementation blueprint if necessary. Also these requirements should be constantly review and analyzed in order to enhance the quality of other modules.
* Calculate payment process: After gathering customer requirements then the calculating module should safely implemented with simplified and better rules so that everyone can be benefited. Analyzing the customer feedback is mandatory after replacing the old rules. And it will be wise to proceed to the next module after majority of customers are happy with the implementation so far.
* Collecting and providing payments process: After that, the system should integrate accurate information of collecting and providing payments. The platform should introduce secured e-Commerce transactions. Security and privacy are the main concern of this module as dealing with money is a sensitive process. The system should notify the all stakeholders timely with the information.
* Assessment and maintenance process: After implementing all the modules, this module has to be implemented to assess and monitor all kind of activities in the system, also to assure the performance of the system. From customer review to the payments each and every actions should be monitored and controlled by the governing authority and they should be responsible to make any changes to the existing system in order to enhance the quality of services.
* Evaluation: To evaluate and justify our proposed system a few points are briefly described below.
* Structured and efficient plan: This is a structured plan that ensures quality in every level. Each module will have individual timeline and budget, so that no misuse of time and money can take place. The computational and processing cost will also be reduced as the modules are performing in different platforms.
* Constant review of customer’s requirements: Customer’s satisfaction should be the main concern because the main goal was to enhance the quality of system with better IT system. It will be ensured in this system by constantly evaluating them in order to maximize the satisfaction.
* Easy to modify and reengineer: It will be easy to integrate, modify and reengineer the modules as they are independent to each other.
* Easy Maintenance: Modular architecture is easy to maintain. The issues are easy to be identified and fixed with limited cost and time.
* Reduced development cost: Development with a structured plan always can save cost.

**Conclusion**: Large and complex information system should migrate the system gradually because a huge amount of valuable data can be stored and loss of a single amount can cause a massive blow to the company. That’s why a proper structured and modularized plan should be taken and implemented in order to perform a successful migration. The main problem of CS2 was the implementation plan. They thought it as one big problem with only one solution. Rather the big problem could have been broken into several smaller problems. Then the solutions could also be smaller and easier. And accumulating those solutions would give a better and efficient solution to the big problem.

**References**:

[1] A. Vera-Baquero, R. Colomo-Palacios and O. Molloy, "Business Process Analytics Using a Big Data

Approach," in IT Professional, vol. 15, no. 6, pp. 29-35, Nov.-Dec. 2013, doi: 10.1109/MITP.2013.60.

[2] “History Of The CSA,” [Online]. Available: <https://www.nacsa.co.uk/history-of-the-csa>

[3] House of Commons Work and Pensions Committee, 2004, “The Performance of the Child Support

Agency,” [Online]. Available: <https://publications.parliament.uk/pa/cm200405/cmselect/cmworpen/44/44i.pdf>

[4] N. Alshuqayran, N. Ali and R. Evans, "Towards Micro Service Architecture Recovery: An Empirical

Study," 2018 IEEE International Conference on Software Architecture (ICSA), Seattle, WA, 2018,

pp. 47-4709, doi: 10.1109/ICSA.2018.00014.