**DB Performance and Security considerations**

An important part of creating a database is the choice between security and performance as if you wish to have a very secure database because you are handling sensitive data that must be protected then your performance will suffer as a result due to the extra checks that must be performed when querying the database. In a similar respect should you choose high performance because you are running queries on a large amount of records and need the results in a reasonable time then you will be limited in the sorts of security measures you may undertake as they will inevitably have a negative impact on performance.

In the database produced as a requirement for the project measures were taken in order to provide a balance in the performance and security characteristics, in terms of performance B Tree indexing was used which as Richardson, R (2015) shows is optimized for systems handling large amounts of data in order to cope with the many thousands of graduate records that must be stored and inevitably queried as well as giving a detailed description of several database features. This type of index keeps data in a tree structure which as Hu *et al* (2014) shows is very useful for speeding up read and writes on data as well as proposing a new type of index which is in part based on B Tree indexing that is designed to reduce the amount of writes on the data.

Another performance characteristic of the database is the design itself as many of the columns in the database are set to allow null values which means the constraints on those columns are quite lax causing less checks on data coming into the database so more data is able to be inserted as well as making the process quite easier for application developers creating the front end of the program where the data in this case will be coming from therefore this improves performance as well as usability.

Measures were also taken to ensure the security of the database as a whole, this is important because even if your application is quite secure the database is still vulnerable due to malicious individuals bypassing the application layer and attacking the database which if it is not secured against this type of activity will result in damages or at worst a complete loss of data. In order to prevent this the database is secured with a password so that only those who need to know for application development purposes have access to the database limiting potential threats against the database.

Another measure that was put in place for the project was that several administrators and normal staff members will require access to the application which intern will be able to make changes to database records whether that be update, insert or delete, as such all of these users details including their username and password which is required to log into the application are stored in a table on the database. Administrators have the ability to view a list of the users who currently have access to the application and make changes to said list whether that be to add a new user thereby giving them application and in so doing database access rights, they also can edit users details such as changing their password for them or even make them an administrator themselves, lastly administrators can delete users should that become necessary for example malicious users may try to gain access to the database by using a user that is no longer being used by an active staff member which is why deleting old users is a necessity.

However despite the several performance and security measures that have already been put in place there is still more that could be done to improve upon these important database considerations. One such measure would be to use a completely different kind of database using Not only structured query language(NoSQL) that favours performance and scalability as backed up by the work of Lee, K (2013) who details what NoSQL databases are, what it has to offer in this case high performance and scalability and its disadvantages such as how new they are meaning they are not as well tested as traditional relational databases.

Another measure that could have been put in place would be the use of views, this is essentially a stored query that would allow a user to see the information that they need to do their job without compromising data security by giving them access to a table that includes sensitive data such as a customer's credit card details or in this case another staff members salary or password, this measure therefore would have a positive effect on the databases overall security.

**References:**

Hu, W, Li, G, Ni, J, Sun, D, & Tan, K 2014, B'p Tree : A Predictive B+ -Tree for Reducing Writes on Phase Change Memory, IEEE Transactions On Knowledge & Data Engineering, 26, 10, pp.2368-2381, Academic Search Complete, EBSCOhost, viewed 1 May 2015.

RICHARDSON, R 2015, 'Disambiguating Databases', Communications Of The ACM, 58, 1, pp. 54-61, Academic Search Complete, EBSCOhost, viewed 2 May 2015.

Lee, K, Tang, W, & Choi, K 2013, 'Alternatives to relational database: Comparison of NoSQL and XML approaches for clinical data storage', Computer Methods & Programs In Biomedicine, 110, 1, pp. 99-109, Academic Search Complete, EBSCOhost, viewed 2 May 2015.