**Comparative Analysis Report:**

|  |  |
| --- | --- |
| **Database Storage Options** | |
| **Relational Database** | A relational database stores the data in form of tables. The entities involved in the system are presented as tables and their attributes are presented as columns in the tables also called as a schema.  Advantages:   * SQL is a powerful query language for interactive information from the tables. * Easy to store and categorize the data. * Easy to understand * Data manipulation is very easy. * Record validation is performed by the database. * Ease of availability of product implementing this model.   Disadvantages:   * Complex data with different feature sets cannot be stored using this database as this is based on data having common characteristics. * Customisation of the data model is hard, because the model is optimised for data with regular structure (i.e. records). |
| **NOSQL / Document Oriented Database** | This database is useful when we have a variety of data. It does not require a predefined schema and uses collections and documents. NoSQL. Examples include MongoDB or Cloundant/CouchDB.  Advantages:   * Schema-less implementation * Highly scalable and useful for companies related to big data. * More useful in real time applications as it does not require a predefined schema.   Disadvantages:   * Record validation needs to be implemented within the application or by using additional libraries. * The capabilities of the query languages differ from product to product as there is no standard in the field. * Lack of knowledge of this type of storage model within the team. |
| **File Based System** | This is the system which uses computer’s storage devices to store the information either in textual format or graphical format using the file manager system.  Advantages:   * Can be located from a path. * Easy to customise as we can decide what to store in the file and these can either be records of the same structure or different records.   Disadvantages:   * Difficult to access files if large set of data is involved. * A lot of duplication is there due to decentralization approach which leads to waste of storage space. * Performance issues as this system is not too fast. * The set of built-in services and capabilities strongly varies from product to product, ranging from simple file access to more sophisticated operations. It might be hard to have available a query language that can be effectively used. * The model might not be able to provide backup and custom implementation is needed for this feature. |
| **Selected Option** | Relational Database |
| **Justification** | This database is particular advantageous because of the nature of the application we’re developing. In particular, we do not need to provide user-defined records and the application entities abide to a well-defined structure that is more effectively represented and manipulated trough a relational model.  Moreover, the team has a well-developed set of skills and expertise with relational database and this will boost the development activities and reduce time. |