**Possible solutions to the database of customers:**

* An ArrayList driven by a class so that we could easily search for each patient, and we wouldn’t be limited in terms of space.
* A binary search tree where the first right and left roots divide the two different types of patients, then it will store them normally by using the patient id.
* A hash table holding linked lists so that the adding and searching of patients would be efficient, and the linked list to give us more space to work with.
* A .txt file that will store all the existing patients (with all the relevant information). When a patient is going to be registered it will search if he already exists in the document.

**Solution to the differing priorities within the patients**

* After some consideration we have concluded that the best way to implement the waiting mechanic that the patients are to be submitted to is a single priority queue that modifies the priority of each patient according to their waiting time. The reason of this is that single queue would mean that priority patients are to be attended as any other patient, leading to an increased risk upon their lives, and while two queues would be a certainly fitting solution, it still has the problem of patients being attended only by their access time, and further risk patients would still be left behind less severe patients. As such, a single priority queue would solve this problem to a reasonable degree; because the priority queue works in around a priority system, higher priority patients would be naturally higher in the list than non-priority patients, also, to avoid normal patients staying indefinitely in queue, a point of priority may be added each time a patient is called.