Functional Requirements:

1. **Donor Registration:**
   - The system should allow donors to register by providing their National ID, Name, City, and Email.
   - The system should validate and store the donor's information in the database.
2. **Donation Acceptance:**
   - The system should check if the time since the last donation is more than 3 months.
   - The system should check if the blood virus test result is negative.
   - If both conditions are met, the system should accept the donation and add it to the blood stock.
3. **Donation Rejection Notification:**
   - If the donation is rejected, the system should notify the donor via email about the reasons for rejection.
4. **Add Donation to Blood Stock:**
   - When a donation is accepted, the system should add it to the blood stock with details such as Blood group, Blood Bank City, and Blood Expiration Date.
5. **Hospital Blood Request Management:**
   - The system should allow hospitals to request blood quantity based on blood type, location (City), and patient status (Immediate, Urgent, Normal).
   - The system should manage and prioritize hospital requests based on the least distance difference between the donation city and the hospital city.
   - The system should respond to requests if the required blood quantity is available in the stock.
6. **Withdraw Blood Quantity:**
   - When a hospital request is approved, the system should withdraw the required blood quantity from the stock and update the stock accordingly.

Non-Functional Requirements:

1. **Performance:**
   - The system should be able to handle a large number of donor registrations, donation acceptance/rejections, and hospital requests efficiently.
   - Response times for donor registration and donation acceptance/rejections should be minimal.
2. **Reliability:**
   - The system should be reliable and available 24/7, as blood donation and hospital requests can occur at any time.
   - Data integrity should be maintained, ensuring accurate storage and retrieval of donor and blood stock information.
3. **Security:**
   - The system should have robust security measures in place to protect donor information and ensure confidentiality.
   - Access controls should be implemented to restrict unauthorized access to sensitive data.
4. **Scalability:**
   - The system should be scalable to accommodate future growth in the number of donors, blood stock, and hospital requests.
   - It should be able to handle increased workload without compromising performance.
5. **Usability:**
   - The user interface should be intuitive and user-friendly for donors, hospitals, and system administrators.
   - Proper documentation and user guides should be provided for easy navigation and understanding of system functionalities.
6. **Compatibility:**
   - The system should be compatible with different web browsers and devices to ensure accessibility for users.
   - It should also be compatible with various database management systems (DBMS) for data storage and retrieval.

These requirements ensure that the blood bank management system functions effectively, securely, and reliably to meet the needs of donors, hospitals, and system administrators.