# Blood Bank Management System Architecture

## Introduction

The Blood Bank Management System is designed to manage blood donations, inventory, and transfusions. It allows users to register as donors, schedule donation appointments, manage blood inventory, and track blood transfusion records.

## Components

### 1. User Management

* **Description**: Handles user registration, authentication, and authorization.
* **Technologies**: Django, SQLite
* **Key Functions**:
  + User Registration
  + User Login/Logout
  + Role Management (Donor, Admin, Staff)

### 2. Donation Management

* **Description**: Manages blood donation appointments and records.
* **Technologies**: Django, SQLite
* **Key Functions**:
  + Schedule Donation
  + Record Donation
  + Donor History

### 3. Inventory Management

* **Description**: Keeps track of blood units in the inventory.
* **Technologies**: Django, SQLite
* **Key Functions**:
  + Add Blood Units
  + Update Inventory
  + View Inventory Levels

### 4. Transfusion Management

* **Description**: Manages blood transfusion requests and records.
* **Technologies**: Django, SQLite
* **Key Functions**:
  + Request Blood Units
  + Approve/Reject Requests
  + Record Transfusion Details

### 5. Notifications

* **Description**: Sends notifications to users about appointments, inventory alerts, etc.
* **Technologies**: Django, Celery
* **Key Functions**:
  + Email Notifications
  + SMS Notifications (if integrated)

## Data Flow

1. **User Registration**: Users register and their information is stored in the User Management component.
2. **Schedule Donation**: Donors schedule a donation, and the appointment details are stored in the Donation Management component.
3. **Donation Recording**: After a donation, the details are recorded and the blood units are added to the Inventory Management.
4. **Inventory Update**: The inventory is updated with new blood units and any changes due to transfusions.
5. **Transfusion Requests**: Hospitals or patients request blood units, which are processed by the Transfusion Management component.
6. **Notifications**: Users receive notifications about their appointments and any important updates.

## Technologies Used

* **Backend**: Django (Python)
* **Database**: SQLite
* **Frontend**: HTML, CSS, JavaScript
* **Notifications**: Celery (for asynchronous tasks), Email, SMS (if integrated)

## Diagram

Architecture Diagram

Architecture Diagram