

WhatYaNeed - Server-Side Procedural Design

1. User Registration

```
PROCEDURE RegisterUser
  DISPLAY registration form
  INPUT name, email, password, confirm_password, selected_role

  IF any field is empty THEN
    DISPLAY "Please complete all fields"
    EXIT PROCEDURE
  END IF

  IF password ≠ confirm_password THEN
    DISPLAY "Passwords do not match"
    EXIT PROCEDURE
  END IF

  IF email already exists in Users_Table THEN
    DISPLAY "Email already registered"
    EXIT PROCEDURE
  END IF

  INSERT INTO Users_Table (name, email, password, role)
  DISPLAY "Account created successfully!"
  REDIRECT to login page
END PROCEDURE
```

2. User Login

```
PROCEDURE UserLogin
  INPUT email, password, selected_role

  FETCH user_record FROM Users_Table WHERE email = input_email

  IF no record found THEN
    DISPLAY "Email not found"
    EXIT PROCEDURE
  END IF

  IF password ≠ user_record.password THEN
    DISPLAY "Incorrect password"
    EXIT PROCEDURE
  END IF
```

```

IF selected_role ≠ user_record.role THEN
    DISPLAY "Incorrect role"
    EXIT PROCEDURE
END IF

SET CurrentUser = user_record
DISPLAY "Welcome, " + user_record.name
REDIRECT to Dashboard based on role
END PROCEDURE

```

3. Create Request

```

PROCEDURE SubmitRequest
    DISPLAY request form
    INPUT title, description, category, urgency, address

    IF any required field is empty THEN
        DISPLAY "All fields are required"
        EXIT PROCEDURE
    END IF

    INSERT INTO Requests_Table (requester_id, title, description, category, urgency, location,
status)
    DISPLAY "Request submitted successfully"
    RESET form
END PROCEDURE

```

4. View Requests

```

PROCEDURE ViewRequests
    FETCH all requests WHERE status = 'open'
    DISPLAY requests in cards

    IF user clicks 'Offer Help' THEN
        CALL OfferHelp(request_id)
    END IF
END PROCEDURE

```

5. Offer Help

```

PROCEDURE OfferHelp(request_id)
    INSERT INTO Help_Offers (volunteer_id, request_id, status)
    UPDATE Requests_Table SET status = 'help_offered' WHERE request_id = request_id
    CREATE notification for requester: "Someone offered to help!"
    DISPLAY "Offer submitted successfully"
END PROCEDURE

```

6. Dashboard

```
PROCEDURE DisplayDashboard(user_role)
  IF user_role = 'requester' THEN
    SHOW total requests, completed requests, notifications
  ELSE IF user_role = 'volunteer' THEN
    SHOW tasks offered, tasks completed
  ELSE IF user_role = 'admin' THEN
    SHOW flagged posts, active users, pending approvals
  END IF
END PROCEDURE
```

7. Notification System

```
PROCEDURE SendNotification(recipient_id, message)
  INSERT INTO Notifications_Table (recipient_id, message)
  DISPLAY "Notification sent"
END PROCEDURE
```

```
PROCEDURE ViewNotifications(user_id)
  FETCH all notifications WHERE recipient_id = user_id
  DISPLAY each notification in list
END PROCEDURE
```

8. Admin Panel

```
PROCEDURE AdminDashboard
  FETCH flagged requests, active users, pending approvals

  IF admin clicks 'Approve' THEN
    UPDATE Requests_Table SET status = 'open'
  ELSE IF admin clicks 'Deny' THEN
    UPDATE Requests_Table SET status = 'closed'
  ELSE IF admin clicks 'Flag' THEN
    MARK request as flagged
  END IF
END PROCEDURE
```