

Software Construction and Development

**ASSIGNMENT 03 & PROJECT REPORT**

***Name:*** *Iqra Zafar*

***Roll Number:*** *20I - 2401*

***Section:*** *S*

***Date of Submission:*** *9th  December 2022*

***Submitted To:*** *Sir Zaheer*

Contents

[1. Description of Project 3](#_Toc121520960)

[1.1. Purpose: 3](#_Toc121520961)

[1.2. Scope: 3](#_Toc121520962)

[2. Work Distribution 3](#_Toc121520963)

[3. Docker Files 4](#_Toc121520964)

[4. Docker Hub Links 5](#_Toc121520965)

[5. Deployment Steps 6](#_Toc121520966)

[6. Link to deployed module 7](#_Toc121520967)

# Description of Project

## 1.1. Purpose:

The Blood Bank Management System is a project that aims to simplify the complex and challenging process of donating blood to hospitals and various centers that may require blood in the case of emergencies. Further in-depth knowledge on this system will be provided below.

## 1.2. Scope:

The Blood Bank Management System aims to digitize the blood donation process by collaborating with Hospitals to move most of the stages of the donation process online and therefore, create an entirely new system of online record keeping and communication, which is designed from the ground up to be accurate, quick and robust when it comes to emergency situations as well as the day-to-day workings of a blood bank. Contracts will also be signed with local and foreign companies, especially fast-food chain restaurants, to create rewards and provide incentives for users to spend their generated points for cash rewards.

A comprehensive online database shall be implemented to store information of all individuals who use the application to donate blood, the access to which will be granted by the business app. With the help of an interactive user interface, the application aims to promote the use of the app for most of the general population. The scope of this system is towards helping individual blood banks, both private and government owned. This can later be expanded to both organ donation as well as other medical applications.

# Work Distribution

Wali Zaidi – Login, Sign-up, Main Landing Page, Update User Information

Shoaib Ahmed – View Appointments, Update Appointments

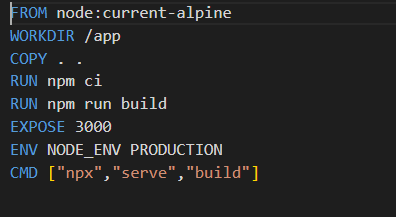
Sheheryar Younas – Generate a Request, Send Request to nearby Donors

Saifullah Ahmad – Check profile details, Check History

Iqra Zafar - Login, Sign-up, Sponsor-Panel, Update and delete it

# Docker Files

***Dockerfile of Backend***



***Dockerfile of frontend***

# ==== CONFIGURE =====

# Use a Node 16 base image

FROM node:16

# Set the working directory to /app inside the container

WORKDIR /app

# Copy app files

COPY . .

# ==== BUILD =====

# Install dependencies (npm ci makes sure the exact versions in the lockfile gets installed)

RUN npm ci

# Build the app

RUN npm run build

EXPOSE 3000

# ==== RUN =======

# Set the env to "production"

ENV NODE\_ENV PRODUCTION

# Expose the port on which the app will be running (3000 is the default that `serve` uses)

# Start the app

CMD [ "npx", "serve", "build" ]

#CMD [ "npm", "start" ]

# Docker Hub Links

**DockerHub links to Frontend image**

[**https://hub.docker.com/repository/docker/iqrazafar1234/frontendimagefinaldeploy**](https://hub.docker.com/repository/docker/iqrazafar1234/frontendimagefinaldeploy)

**DockerHub links to Backend image**

**https://hub.docker.com/repository/docker/iqrazafar1234/backendimage**

# Deployment Steps

1. Create an account on Amazon AWS.
2. Login in to the AWS Management Console.
3. Search for services and select EC2.
4. Select Launch Instance from the EC2 Dashboard.
5. Enter a name for the instance “IqraSCDProject”.
6. Select the Ubuntu Image under Quick Start.
7. Select t2.micro as Instance Type.
8. Create a new key pair for login from your own console and enter a key pair name and click on Create Key Pair.
9. Under Network Settings, click on Create New Security Group and add a Security Group Rule. Set the Inbound Security Group Rule’s Source Type to “Anywhere”
10. Leave the Configure Storage options as it is and click on Launch Instance.
11. Click on view All Instances.
12. Select the “IqraSCDProject” Instance and click on Connect and a console will open in the next tab.
13. Run commands:

* sudo apt install
* sudo apt update
* sudo apt install docker.io
* sudo apt install docker-compose

1. After running these commands, go to your docker hub registry and click on your frontend image repository. Copy the docker pull command. (Before building the frontend image, replace all the local host addresses with the aws ec2 instance’s Public IP and then push it to the registry)
2. Go to your backend repository and copy the docker pull command.
3. On the AWS Console run the command:

* sudo docker pull iqrazafar1234/frontendImagefinaldeploy
* sudo docker pull iqrazafar1234/backendImagefinaldeploy
* sudo docker images
* sudo docker run –d –p 3000:3000 iqrazafar1234/frontendImagefinaldeploy
* sudo docker run –d –p 3700:3700 iqrazafar1234/backendImagefinaldeploy

1. To check the containers running, run the command:

* sudo docker ps

1. Copy the Public ID and with the attach 3000 i.e. 52.194.192.70:3000
2. Enter the address in your browser and the react application will be displayed and responding to the http requests.

# Link to deployed module

52.194.192.70:3000