Optimizing Deployment Speed and Reliability with DevOps

PHASE 5- DEPLOYMENT PHASE

College Name: Dr. Sri Sri Sri Shivakumara Mahaswamy College of Engineering, Bengaluru

Group Members:

- **Name: Binod Khatri Chatri**
  CAN ID Number: CAN_35805323
- **Name: Ashwini J**
  CAN ID Number: CAN_35777960
- **Name: Kavana M**
  CAN ID Number: CAN_35778518
- **Name: Shashi kumar M**
  CAN ID Number: CAN_35805208

Deployment:

Automate the deployment of your full-stack app using CI/CD and Docker. Ensure each code push leads to a smooth, production-ready deployment.

1. Choose a Deployment Platform

Some common choices:

- Heroku – simple setup for containerized apps
- Render / Railway – free-tier options for quick deployments
- AWS EC2 / Lightsail – more control, more configuration
- VPS (like DigitalOcean) – total control (requires setup)

2. Prepare App for Production

Frontend (React):

- Already handled via Dockerfile (Nginx serves build/)

Backend (Express):

- Production-ready: uses Dockerfile
- Expose port 5000

3. Set Up Heroku for Docker Deployments

Step 1: Install Heroku CLI (if needed):

npm install -g heroku

Step 2: Log in to Heroku:

heroku login

Step 3: Create a Heroku app with container support:

heroku create your-app-name --stack=container

Step 4: Add Heroku credentials to GitHub:

- Go to Settings → Secrets → Actions
- Add the following secrets:
    - HEROKU_API_KEY: Your Heroku API key (get it from Account Settings)
    - HEROKU_EMAIL: Your Heroku account email
    - HEROKU_APP_NAME: The app name you used above

4. Update GitHub Actions for Deployment

name: CI/CD Pipeline

on:

  push:

```
    branches: [ main ]
jobs:
  build-test-deploy:
    runs-on: ubuntu-latest
    steps:
    - name: Checkout code
      uses: actions/checkout@v3
    - name: Set up Node.js
      uses: actions/setup-node@v3
      with:
        node-version: '18'
    - name: Install client dependencies
      run: |
        cd client
        npm install
    - name: Run client tests
      run: |
        cd client
        npm test

    - name: Install server dependencies
      run: |
        cd server
```

```
      npm install

  - name: Run server tests
    run: |
      cd server
      npm test
  - name: Deploy to Heroku
    env:
      HEROKU_API_KEY: ${{ secrets.HEROKU_API_KEY }}
      HEROKU_APP_NAME: ${{ secrets.HEROKU_APP_NAME }}
      HEROKU_EMAIL: ${{ secrets.HEROKU_EMAIL }}
    run: |
      echo "$HEROKU_API_KEY" | docker login --username=_ --password-stdin registry.heroku.com
      docker build -t registry.heroku.com/$HEROKU_APP_NAME/web .
      docker push registry.heroku.com/$HEROKU_APP_NAME/web
      heroku container:release web --app $HEROKU_APP_NAME
```

5. Verify Your Deployment

Once your GitHub Action finishes:

- Visit https://your-app-name.herokuapp.com
- You should see your deployed full-stack app running