TU Dublin, Tallaght Campus

Department of Computing

M.Sc. in Dev-Ops

2021/2022

Continuous Software Deployment

CA1 Project

Due: 13/12/2021

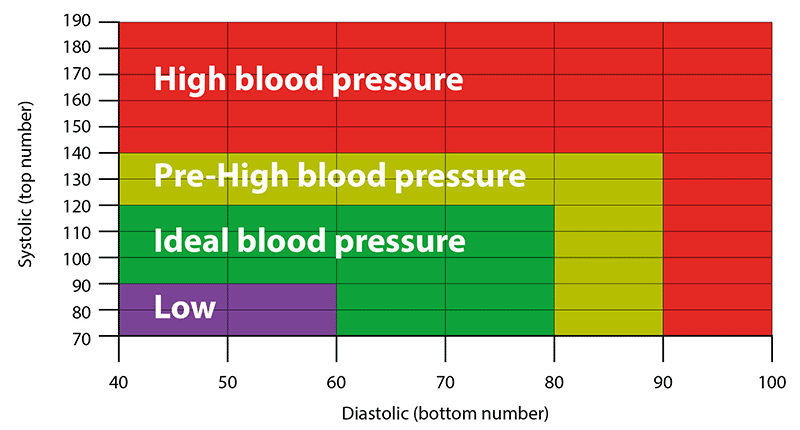
Value: 65%

Lecturer: Gary Clynch

**CI/CD Pipeline for Blood Pressure Category Calculator**

Blood pressure is measured as ‘systolic pressure’ over ‘diastolic pressure’, the unit of measure being mmHG. For example, a reading of **100 / 80** indicates a blood pressure with a systolic pressure of 100 over a diastolic pressure of 80. For the purpose of this application, systolic pressure can range from 70 to 190, and diastolic from 40 to 100. Systolic pressure is always higher than diastolic pressure.

A blood pressure reading can be used to categorise the blood pressure of the patient as low blood pressure, ideal blood pressure, pre-high blood pressure, or high blood pressure. This figure below shows how the category can be calculated:



In the above figure consider the lower limits to be inclusive in the category e.g. a systolic value of 140 or greater indicates high blood pressure.

Source code for a simple Blood Pressure Category calculator implemented in ASP.Net Core Razor pages can be seen in this public Git repo <https://github.com/gclynch/bp>

The code is incomplete with respect to the logic to calculate the blood pressure category, complete this code. Add telemetry tracking to the application.

You are required to design and build a CI/CD pipeline for the application using tools and frameworks of your choice.

As regards the CI part of the pipeline:

1. Include some unit testing (aim for code coverage of at least 80%)
2. Include some BDD type testing
3. Include code analysis/metrics
4. Include security features (e.g. checking for dependencies with vulnerabilities etc.)

In the CD part of the pipeline:

1. Implement a release management strategy (e.g. QA, staging etc.) and deployment strategy (e.g. blue/green or canary etc.)
2. Include some E2E testing
3. Includes some performance testing
4. Includes some security features (e.g. pen testing)
5. Implement continuous telemetry monitoring if appropriate
6. Implement authorisation gates if appropriate

For the whole CI/CD pipeline include some security tasks. Any bugs or code quality issues should be addressed and the code re-factored.

Add one new feature (with appropriate tests) in the application. The new feature should be max 30 lines of code. Use a branching strategy like the Git Feature Branch workflow.

**Deliverables**:

1. Video demo of pipeline running / description of pipeline (max 15 minutes)
2. Report describing the:
   1. CI/CD pipeline (including tests etc.), design philosophy
   2. The new feature (e.g. a user story)