CO3095 / CO7095 Coursework 1.

This coursework component accounts for 15% of your module mark.

System

Your software development company has been approached by an entrepreneur who has had an idea for a new web-application for short-term house rentals. She wants to call her website "AirBedAndBreakfast". The website will enable people to register and advertise their homes, so that users can rent them for short periods of time. The system should include the following:

1. A registration facility for users and home-owners.
2. Home owners should be able to upload details and pictures of their property onto a central database.
3. The central database should keep track of when a property is available (i.e. when users book a property, the availability dates should be updated accordingly).
4. Whilst browsing, users should be able to keep track of their favourites (stored locally).
5. Once a user has decided upon a property, the website should prompt them to ask them for the dates they require (ensuring that only valid dates are entered). It should also ask them to write a short message introducing themselves to the owner.
6. Once this has been submitted, the owner should be notified and, upon their logging in, should be presented with a pop-up message displaying the details of the request made by the user. They should then either confirm or reject the user's request.
7. The system should also maintain a ratings-system, whereby home owners and users are given a rating. This rating is updated after the stay is complete – the user rates the home owner and vice versa.
8. The system should be reasonably reliable. A degree of down-time is tolerable, but only very occasionally.
9. The system should be capable of dealing with a reasonably high load of users.
10. As a start-up, it is important that the source code should be reasonably flexible; new requirements are bound to be frequently added.

Task

1. Predict the cost of each requirement:
   - For each requirement (functional and non-functional), provide a guess at how many lines of code its implementation might required. Provide this number in thousands of lines of code (KLOC). So 450 lines of code = 0.45 KLOC.
   - Use historical data to estimate how much effort will be required to implement each requirement (using your estimates for KLOC). The data is the NASA data set that was demonstrated in lectures:

    `https://terapromise.csc.ncsu.edu/repo/effort/cocomo/cocomo2/nasa93-dem/nasa93-dem.arff`
    Specifically, within this data you can concentrate on the relationship between the effort and the kloc column. Compute a Linear Regression model on this data to calculate the relationship between KLOC and effort. This should give you a cost per KLOC, and an initial cost (if any).

    Refer to the list of essential R functions included in the appendix to this assignment.

  - You are aware that activities such as testing will become increasingly expensive with the size of the software. Using the historical data, select different values of b and gauge visually (by graphing the resulting curve) which value of b (if any) produces a better approximation than the linear model alone.

2. Produce a PERT chart:

   - For each requirement, use the model you generated above to calculate the "normal" predicted effort for each of the requirements.
   - Produce a table, add the normal estimates you've calculated, and then also make up your own pessimistic and optimal estimates. Use these to calculate the expected effort for each activity. Make sure that every row is numbered to provide an ID for the requirement in question.
   - Draw out a dependency network. For this part of the exercise, there are a multitude of different possible dependency networks, depending on the assumptions you make about the underlying implementation. In your submissions, these dependencies have to be justified. Therefore, aside from the dependency network, include a brief justification on a per-activity basis that explains its dependencies on previous activities.

     Within this network, calculate the earliest start time, the latest start time, the earliest finish time, the latest finish time, and the slack time.
   - List the nodes on the critical path.

3. Discuss the possible problems (if any) that could arise when interpreting the resulting PERT chart. What potential problems might arise?

Submission Instructions

Produce a PDF containing the following parts:

1. The use of Linear Regression to predict the cost for each requirement. Provide a sufficiently detailed description of how you applied Linear Regression, how you estimated the scale factor, what the various components of the inferred model are, and how you have interpreted these. Include also a table, listing each requirement, its predicted KLOC size, and its predicted cost according to the model you inferred.

2. The table with the cost estimates.

3. A network diagram, with nodes annotated to include the various required figures (earliest start, latest start, etc.).

Assessment

The three parts are weighted as follows:

- The calculation of the Linear Regression, the accompanying description, and interpretation. [30%]
- The process of producing and justifying the PERT chart. [40%]
- The discussion of how to interpret the final PERT chart. [30%]

Appendix: R Commands

```
library(ggplot2)
library(foreign)

nasa <- read.arff("nasa93.arff")

#Set the font size to a large font to make the plots more readable
theme_set(theme_gray(base_size = 18))

#Visualise the KLOC versus effort as a scatter plot
qplot(data=nasa,x=equivphyskloc,y=act_effort,size=I(3))

#Add a line representing the linear regression line to the data
qplot(data=nasa,x=equivphyskloc,y=act_effort,size=I(3))+ geom_smooth(method = "lm", se = FALSE)

#Calculate the linear model
fit <- lm(nasa$act_effort˜nasa$equivphyskloc)

#Print out the coefficients of the linear model
summary(fit)

#Make your own function, called EffortEstimation, assuming x is the LOC we supply,
#and that we hard code the cost per LOC as 12.
EffortEstimation <- function(x) {12 * (x)}

#Plot the function we have produced
ggplot(data.frame(x=c(0,250000)), aes(x=x)) + stat_function(fun = EffortEstimation)
```