Software Engineering, Group coursework worth 20% of your final mark

Rami Bahsoon, School of Computer Science, University of Birmingham, UK

Nandy Millan, IT Innovation Manager, Architecture, Security and Innovation Team, University of Birmingham, UK. Email – n.millan@bham.ac.uk

Submission is due by December 6th 2019 3:00 pm – Firm Deadline

In collaboration with the Architecture, Security and Innovation Unit of the University IT Services, we are seeking innovative suggestions for interactive applications of your choice that aim at tackling mental health and/or raising awareness about mental health. The scope is open and can relate to any of the mental health issues that your group is interested to investigate and design a software solution for. Among the many mental health issues include but not limited to depression, low self-esteem, anxiety and panic attacks, sleep disorder, addiction, suicidal thoughts, mood fluctuation etc. Your application may target one or more audience (e.g. mental health in children, school pupils, University students, ageing population, addicts, homeless etc.). You software solution has to deal with one chosen issue of interest (e.g., an app to manage depression; a virtual reality environment for positive thinking; a hologram boot for confidential counselling; an intelligent app for predicting mood fluctuation etc). Your software solution may also aim at educating and engaging with the concerned individuals and/or public. Your application may wish to use AI, Big data and/or emerging technologies (e.g., virtual and augmented reality, gesture controlled interactions, holograms, and/or mobile apps among the others- as seen fit) for your chosen system. Please note that the University has been investing in Augmented and Virtual Reality technologies (see that the Collaborative engineering lab) and you may wish to feature these in your solution.

We are open for *creative, novel ideas and useful applications of societal change and positive health impact.* We hope that your innovative solution will help us showcase the University of Birmingham as a cutting-edge institution in the use of data science and emerging technologies. We also hope to demonstrate the creativity and ability of our software engineering students to conceptualise

innovative digital solutions and to demonstrate their technical skills in systematically engineering data science solutions, covering requirements, distributed architecture and design.

- Describe your proposed system in 600 words (or less), <u>stating your own assumptions on the scope of the system. Be precise, concise, and creative! Note - your scope should be reasonable and interesting enough to be handled by group members.</u> **(1 point)**
- State the functional and non-functional requirements of your system. Use sensible phrasing and grouping. **(2 points)**
- Use UML to document the analysis and design of your system to a professional level.

Figure 1. UML Diagrams
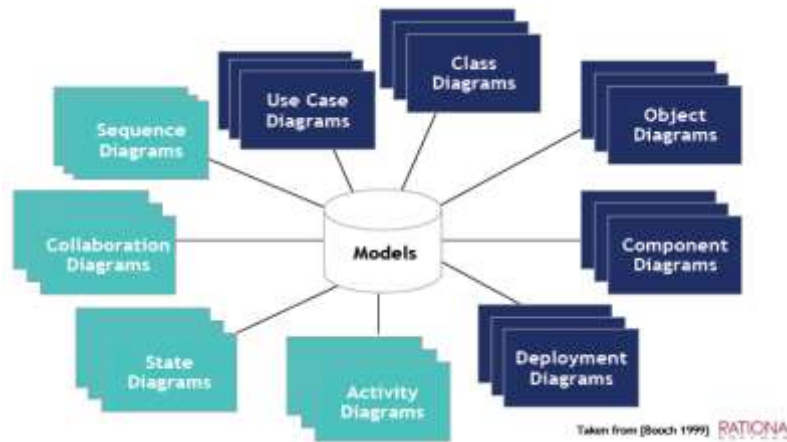
- Provide a comprehensive Use Case diagram for your system clearly indicating the actors involved. Make sure you use stereotypes such as <<extend>> and <<include>>. **(2 points)**
- Choose one non-trivial use case from your Use Case Diagram. Document the corresponding actors. Provide meaningful documentation for the chosen Use Cases using pre/post conditions, flow of events and scenarios. **(2 points)**
- Provide an Activity Diagram for a non-trivial scenario of interest. Make sure you use forks, 4 swim lanes, and conditions. **(2 points)**
- Perform Class Analysis: **(2 points)**
    - Perform noun-verb analysis over your specification: Identify the potential classes and potential operations for these classes using what discussed in Lectures. Derive your CRC (i.e., Class, Responsibility, and collaborators)

- Produce a First-Cut Class diagram to combine the consolidated results of using the above two techniques – i.e., noun-verb analysis and responsibility-driven analysis.
- Detail your Class Diagram. This should provide details on attributes, operations, relationships, visibility, multiplicity, etc) (refer to the class notes).

– Provide an Object Diagram for a non-trivial scenario of interest. **(1 point)**
– Provide a Sequence Diagram for a non-trivial scenario of interest. Make sure you use guards, envelopes, etc. State any assumption you make. **(1 point)**
– Provide a State Diagram for a non-trivial scenario of interest. Make sure you use guards, envelopes, etc. State any assumption you make. **(1 point)**

You need to adopt a sensible architecture. Your group should work on evaluating two possible architecture choices. Among the numerous possible tradeoffs, you should consider the security, performance, scalability, availability, safety etc. once you evaluate your architecture choices and decisions. You are also encouraged to think of other tradeoffs.

– Provide Component Diagrams for the two candidate architectures. Provide One Deployment Diagram for your chosen solution **(2 points)**

- Identify the potential components. You may organise them into categories as discussed in class. (Infrastructure, services, UI etc.),
- Specify their interfaces,
- Compose the components,
- Model ports (if any) and subcomponents (if any).
- Appraise your design using guidelines/heuristics to componentization.

– Your team should now make use of the Architectural Tradeoffs Analysis Method (ATAM) (http://www.sei.cmu.edu/reports/00tr004.pdf) to evaluate and compare the two architecture choices, which you have described above. Your group should simulate the ATAM evaluation process and provide justification for the architectural decisions and their tradeoffs with respect to scalability, availability, security, performance etc. Report on the Utility Tree step of ATAM only (**2 points).**

- Provide a sketch of the menu designs to porotype the main features of your system **(1 point).**
- **Consistency and Maturity**. Follow a systematic design for all the above questions. Be consistent in using the naming across all your diagrams and their supporting documentations. Strive for a professional analysis and presentation. Make your own judgment for what to be modeled, documented, and presented **(1 point).**

Submission requirements
- The report should be typed and presented in A4 paper format. The covering page should include your group number, ID numbers, and e-mails of all members in your group.
- Your diagrams should be produced using open source UML packages or Visio (installed in all labs).
- Please, do work effectively towards meeting the requirements of this coursework. Brief Snapshots of this exercise may appear in your examination.
- Please submit by **Dec 6th/2019 3:00 p.m.** both a hard- and soft - copies of your documentation. E-mail your softcopy to **softeng.bham@gmail.com with your group in the subject head and all group members in the cc/body of your e-mail.**
- **Please DO NOT use my e-mail,** as this will make it difficult to trace all submissions. Deposit your hardcopy in the designated Software Engineering safe (by the reception).
- Demonstrators of this module will provide courtesy group feedback.

<div align="center">Enjoyable Coursework and Good Luck!</div>