

Pearson Institute of Higher Education
Object-Oriented Systems Analysis
and Design
Semester 1 – Assignment

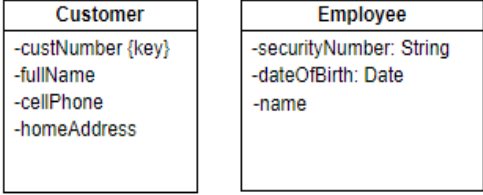

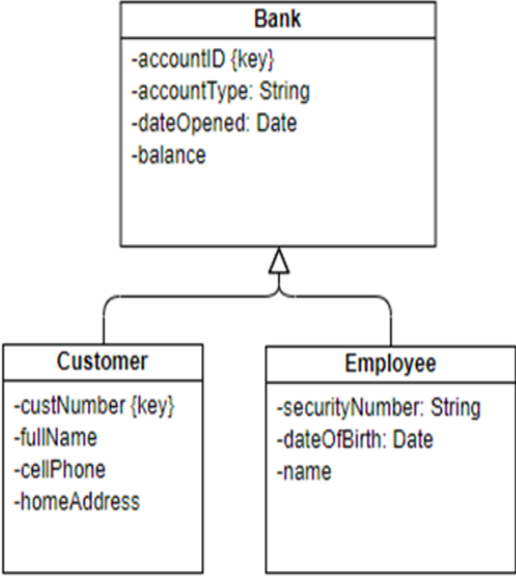
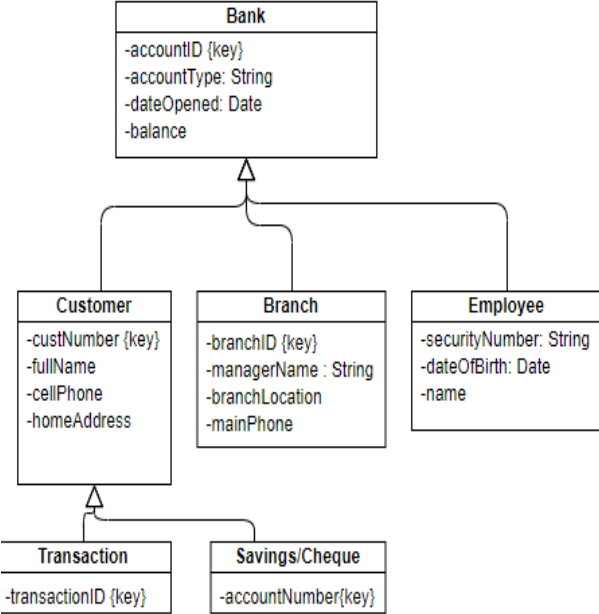
Nompumelelo Mtshatsheni, Student Number: BXMDLL7W9
5-8-2020

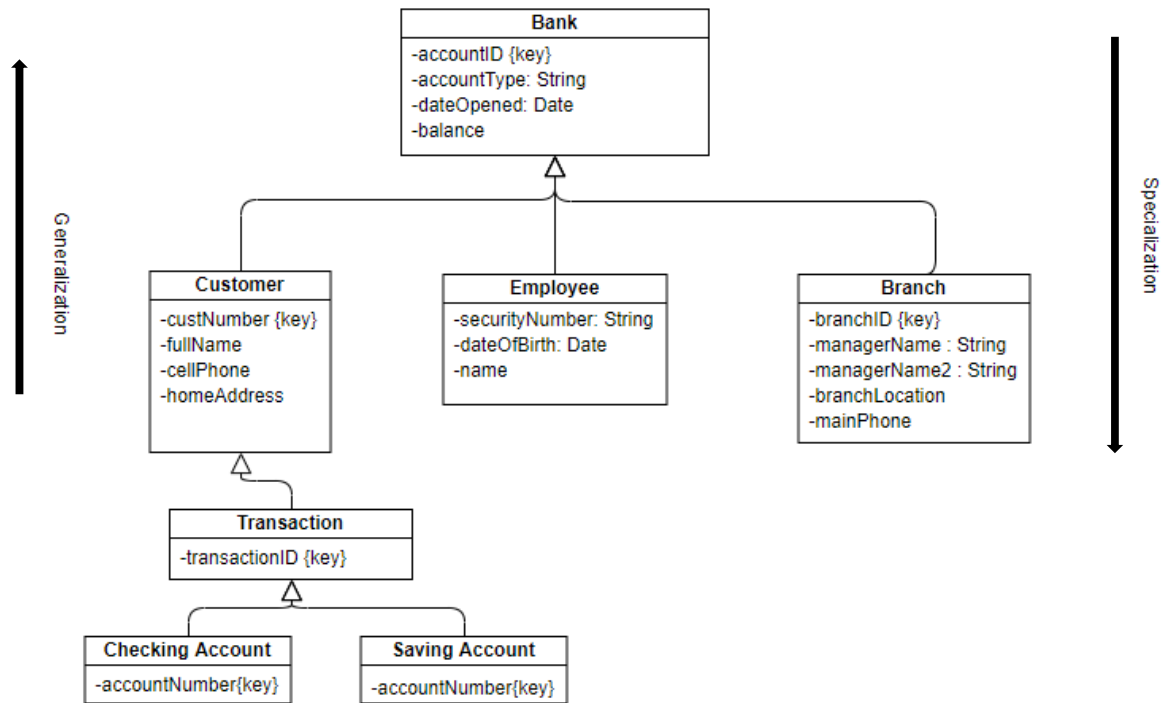
Table of Contents

Question 1 – Scenario	2
Question 2 – Class Diagrams, Use-case diagrams & activity diagram	6
Bibliography	9

Question 1 – Scenario

1.1

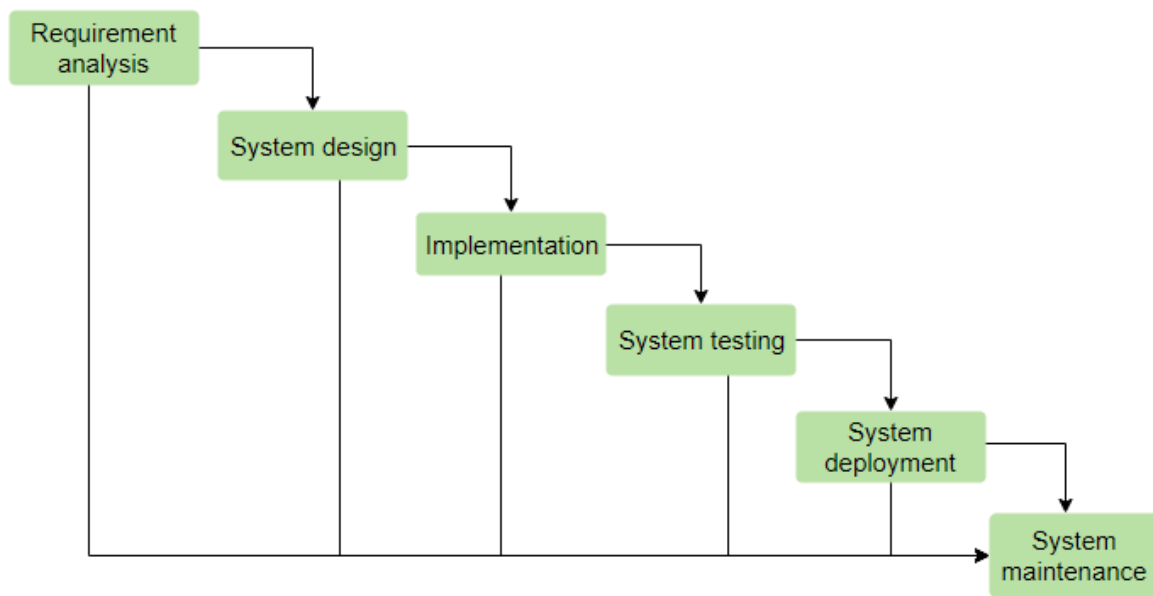
<p>Class</p> <ul style="list-style-type: none"> It signifies the gathering of objects. Built with set of functions that will describe the actions of objects (Weisfeld, 2013) 	<p>Object</p> <ul style="list-style-type: none"> It recognizes and separates objects from the system or program (Weisfeld, 2013) Are modelled corresponding to the physical presence of certain things 
<p>Subclass</p> <ul style="list-style-type: none"> It is a child class from the extension of the superclass 	<p>Superclass</p> <ul style="list-style-type: none"> More general class in generalization or specialization (Weisfeld, 2013) 
<p>Generalization</p> <ul style="list-style-type: none"> Occurs when all the classes have been merged to form a class in an elevated hierarchy (Weisfeld, 2013) 	<p>Specialization</p> <ul style="list-style-type: none"> It is known as the opposite procedure of generalization; the recognizing aspects of clustered objects will be established for more focused classes from existing classes (Weisfeld, 2013)



1.2 A waterfall lifecycle it is simpler to utilize and comprehend. Each phase within the waterfall needs to be finalized ahead of moving to the following stage only then you can commence with the development although it does not allow an intersecting with the phases. (Valacich, 2014)

It is used before the development of any software project. It demonstrates the software progress procedure in a linear form. This will show that the progress of every development starts only if the preceding of each phase is completed. (Valacich, 2014)

The diagram below indicates the interpretation of the various phases found in a waterfall model



(Tutorialspoint, 2020)

- First, the requirement analysis. It is the probability needs of the system to be established are secured within this stage, in addition to recorded in a constraint design file (Valacich, 2014)
- Second, the system design. The requisite requirements after the first stage are examined and analyzed within this stage as well as the system design is arranged. It is extremely helpful in identifying the hardware and the system obligations as well as supports in distinguishing the whole-system structural design (Valacich, 2014)
- Third, implementation. All the contributions from the system design, the system is initially built-in minor programs called components, which stay incorporated for the next stage. Every single component is created and examined for its functionality (Valacich, 2014)
- Fourth, system testing. Each of the components established in the third phase, they are combined into a system following a scanning for each component
- Fifth, the deployment of the system. Hence, once all the testing is done, the result is installed for the client to choose workspace and it can also be distributed into the marketplace (Valacich, 2014)
- Sixth, maintenance. There will certainly be problems which will arise for the consumer workspace environment, fixing problems will need to occur regularly. This final stage is to ensure that the system works perfectly and deliver the best version for clients (Valacich, 2014)

1.3

- The assessment period occurs extremely too late in the developmental procedure (Lvivity, 2020)
- The client's important response cannot be incorporated with the current development stage as well as the opportunity to familiarize themselves with the system further, not until the final product is completed (Lvivity, 2020)
- The lower flexibility of the entire system makes it difficult to alter modifications while creating (Tutorialspoint, 2020)
- Hence, a waterfall lifecycle is impossible to revert to a previous phase and fix something, once a phase is done with its errors, it is likely to threaten the whole system (Lvivity, 2020)

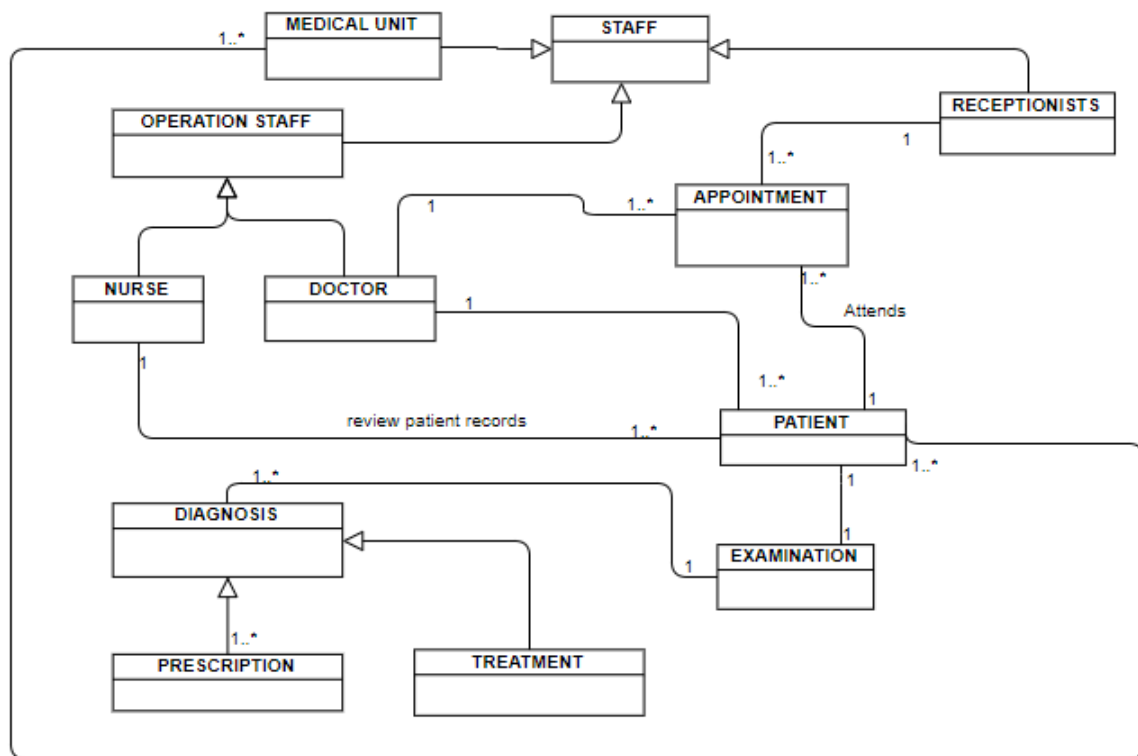
1.4

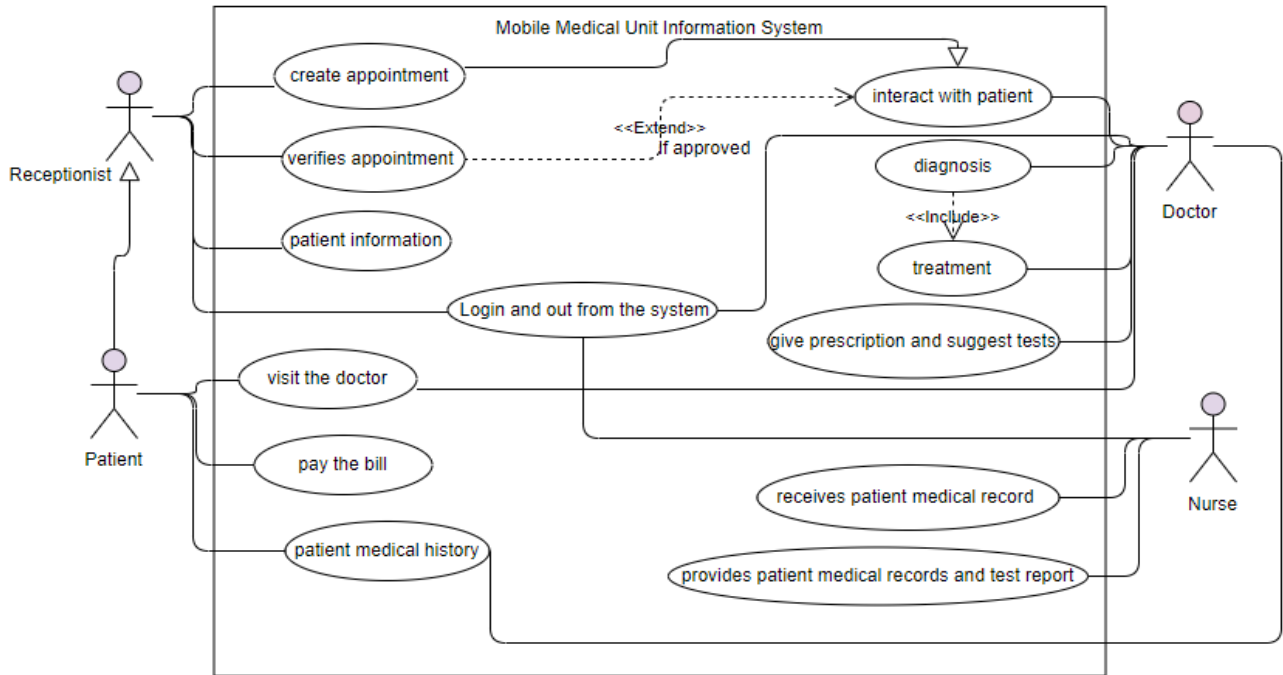
- The enlarged protection cost, uncertainty the foundation of the code which reused for the software system, its main units will not be available, this will undoubtedly increase the cost of maintenance. All the elements found inside the system will be incompatible once the system is changing (Software Engineering , 2020)
- With the lack of tool support, most software tools do not support expansion with recycling. It might be challenging or unfeasible to incorporate these devices with an element store system. Hence, the software process understood that all the necessary tools will not be able to intake the reuse into the current account. This is mainly valid for devices that assist implanted systems planning (Sommerville, (n.a))
- Being unable to design, not many software manufactures would rather modify mechanisms since they suppose enhance them. Although, this is comparatively to do with hope and relatively to do with the reality that composing a unique software is regarded to be more difficult rather than reprocessing other individuals' software programs (Software Engineering , 2020)

(Schmidt, 2020)

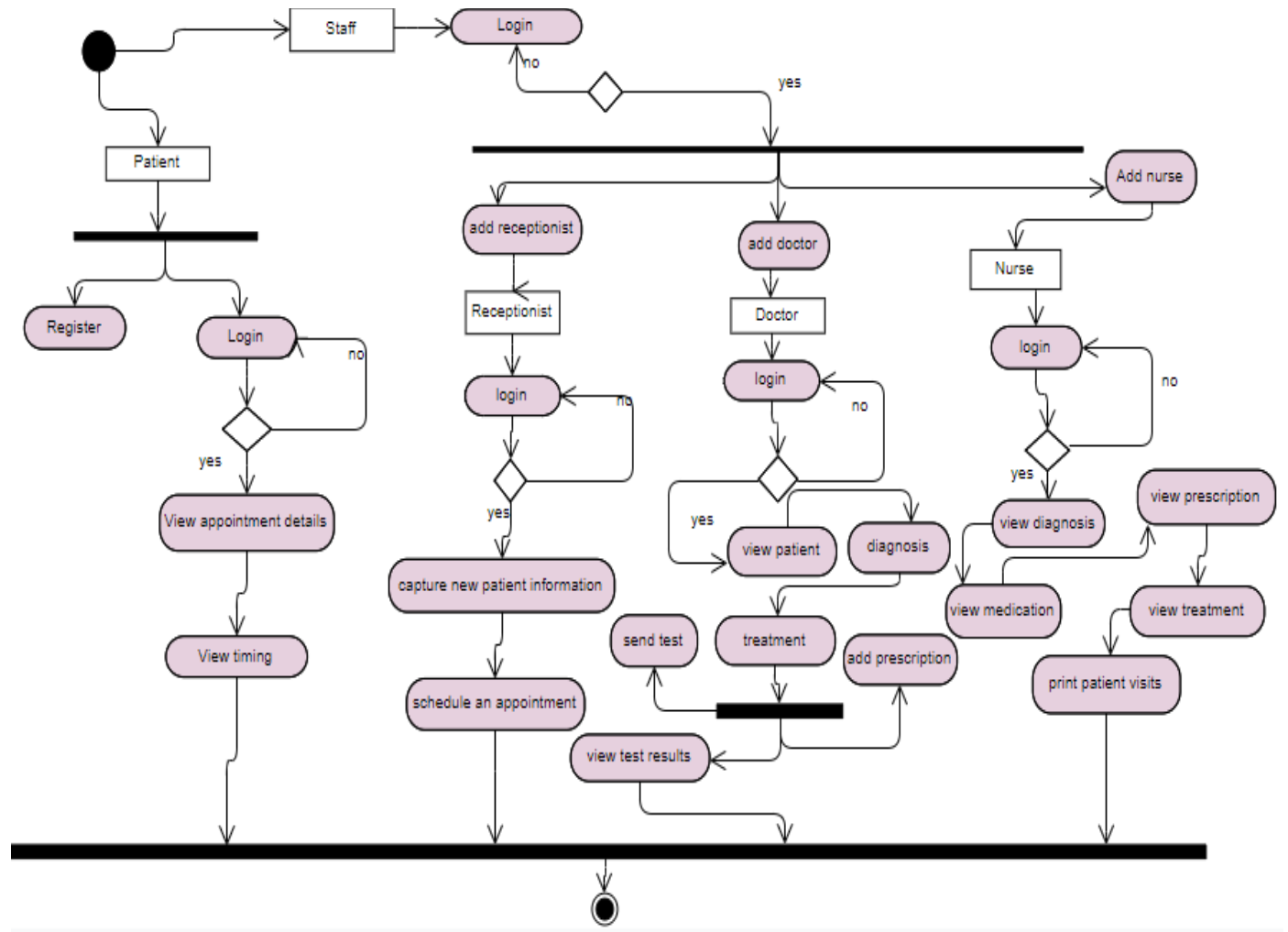
Question 2 – Class Diagrams, Use-case diagrams & activity diagram

2.1





2.3



Bibliography

- Software Engineering* . (2020, April 10). Retrieved from Software Reuse:
<https://cs.ccsu.edu/~stan/classes/CS530/Notes18/15-SoftwareReuse.html>
- Lvivity. (2020, April 10). *Lvivity - Waterfall Methodology: Advantages, Disadvantages And When to Use It?* Retrieved from <https://lvivity.com/waterfall-model>
- Schmidt, D. (. (2020, April 10). *Why Software Reuse has Failed and How to Make It Work for You*. Retrieved from <http://www.dre.vanderbilt.edu/~schmidt/reuse-lessons.html>
- Sommerville, 2. ((n.a)). *Software Engineering*, 10 ed., Chapter 15. In *Advanced Software Engineering* (p. (n.a)).
- Tutorialspoint. (2020, April 10). *Tutorialspoint - SDLC - Waterfall Model*. Retrieved from https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm
- Valacich, J. &. (2014). *Essentials of System Analysis & Design*. 5th Edition. In V. e. al. England: Pearson Education Limited.
- Weisfeld, M. (2013). *The Object-Oriented Thought Porcess, Fourth Edition* . United States: Pearson Education .