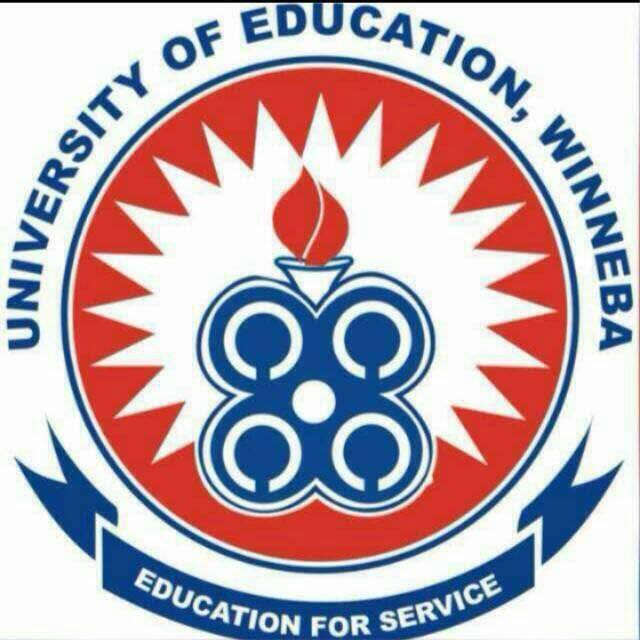
**UNIVERSITY OF EDUCATION WINNEBA, KUMASI CAMPUS**

**COLLEGE OF TECHNOLOGY EDUTAION, KUMASI**

****

**ASSIGNMENT PHASE 1**

**COURSE TITLE: INTRODUCTION TO SOFTWARE ENGINEERING**

**COURSE CODE: ITC 352**

**LECTURER: Dr TWUM FRIMPONG**

**Topic:**

Giving reasons for your answer based on the type of system being developed, suggest the most appropriate generic software process model that might be used as a basis for managing the development of the following systems:

**Group List**

|  |  |
| --- | --- |
| **Name** | **Index number** |
| **Samuel Gyamfi Adabor** | **5171040031** |
| **Appiah Ernest** | **5171040042** |
| **Birago Priscilla Dua** | **5171040036** |

1. **A system to control anti-lock braking in a car**

This is a security basic framework so requires a ton of direct front investigation before execution. This absolutely needs an arrangement driven way to deal with advancement with the prerequisites deliberately examined. A Waterfall model in this manner the most fitting way for this system particularly suited to the development of systems that have stringent safety, reliability, or security requirements. The formal approach simplifies the production of a safety or security case. This demonstrates to customers or regulators that the system actually meets its safety or security requirements.

1. **A virtual reality system to support software maintenance.**

This is a system where the requirement will change and there will be an extensive user interface components. Incremental development with, perhaps, some UI prototyping is the most appropriate model. An agile process may be used. This is where the necessities will change and there will be a broad UI segment. Gradual Advancement is the most suitable

1. **A university accounting system that replaces an existing manual paper based system.**

This is a system which the requirements are fairly known since is to replace an existing manual paper based system, and so no new fundamental specification activities will be stated. Rather the system will be developed in series of versions with each version adding functionalities to the old versions until a newly improved system is developed. Perhaps, incremental model would be appropriate.

1. **A travel planning system that helps users plan journeys with the lowest environment impact.**

System with a complex user interface but which must be stable and reliable. An incremental development approach is the most appropriate as the system requirements will change as real user experience with the system is gained. A steady improvement approach is the most proper as the framework necessities will change as genuine client involvement in the picked up.

1. **The ATM system**

Business programming frameworks normally programming concentrated, and regularly being changes when business objectives or procedures are changed. So incremental development is better. This type of systems usually involve many hardware components which are not easy to change. Usually safety critical which needed be built based on well planned incremental process.

6. **MTN mobile money system**

Business software systems usually complex, software intensive, and frequently being changes when business goals or processes are changed. So incremental development is better since necessities will change as certifiable customer contribution in the structure is gotten.

7. **A university accounting system that replaces an existing system**

This is a system whose requirements are fairly well-know and which will be used in an environment in conjunction with lots of other systems. This is a framework whose necessities are genuinely notable and which will be utilized in a situation related to loads of different frameworks. Consequently, a reuse-based methodology is probably going to be proper for this.