SCI 250 PROGRAMMING PROJECT

BY Ms Vasdinus Ngemu

vngemu@seku.ac.ke,0711230883

{Advisory Corner: Keep COVID 19 and HIV and AIDs at bay. Road safety should be your individual responsibility , Avoid stress and take care of your digital figure print internet never forgets}

Purpose

- To provide expose learners to practical software solution development process

Expected Learning Outcomes:

- By the end of the unit the learner should be able to:
  - Identify a programmable problem
  - Identify the requirements for a software solution to a programmable problem
  - Design a software solution
  - Develop software solution
  - Test a software solution
  - Document a software solution

Assessment

- The continuous assessment test will carry a total of 30%, and will be examined as follows:
  - CAT One will account for 15% *(problem identification upto design)*
  - CAT Two will account for 15%(*first system presentation)*
- The University final Examination will account for 70%*(final system presentation plus documentation)*

Course Description

- Students will be required to work in groups; Each group will be expected to develop an Information Technology solution to a practical problem; Each group will be expected to deliver an executable software and documentation.

TOPIC 1: PROBLEM IDENTIFICATION

- In groups of 5students, you are required to select a programmable problem to solve.
- Projects can be categorized based on the project's impetus, time window, and general priority.
- The impetus for a project is often to respond to a problem, an opportunity, or a directive.
- Regardless of the reason behind the software development, the software should be data driven.
- This means the software should have both front and backend.

Categorizing IT Projects

- **Problems** are undesirable situations that prevent an organization from achieving its goals. These problems can be current or anticipated e.g users of an information system may be having trouble logging on to the system or getting information in a timely manner because the system has reached its capacity. The company could initiate a project to enhance the current system by adding more access lines or upgrading the hardware with a faster processor, more memory, or more storage space.
- **Opportunities** are chances to improve the organization e.g a project to deliver a new product.
- **Directives** are new requirements imposed by management, government or some external influence.
- Organizations select projects for any of these reasons. It is often easier to get approval and funding for projects that address problems or directives because the organization must respond to these categories to avoid hurting their business.

Examples

- Some examples of IT programmable projects include the following:
- Hospital Management information system
- Smartphone application
- A company develops a driverless car.
- A school examination Management system
- A supermarket management system
- Cafeteria management system
- A social media platform
- An organization marketing website
- A government group develops a system to track child immunizations.
- Air and water pollutant monitoring system
-

OBJECTIVES

- Based on the title chosen the group will be required to formulate the objectives of the program/software. These objectives should be a reflection of what the program will do i.e the functions of the program. The objectives must be SMART, meaning:
- Specific
- Measurable
- Attainable
- Realistic
- Time bound

Examples of SMART objectives

- The system should be able to:
- Authenticate users
- Capture and store user data
- Display daily sales(assuming it's a sales management system)
- The team is expected also to prepare a roadmap to be followed during the development for proper management of time. The project shall be carried out in 13week. The team is require to prepare a gnat chart to show when the work shall be done.

Expected output

- The output of this topic shall be:
- A list of group members(reg no., name)
- Title of the project
- Objectives to be achieved.
- Gantt chart
- Once you have chosen your title, written the objectives and prepared a Gantt chart you shall be required to seek approval from your lecturer before you proceed to topic two by submitting a document that contain the topic outputs as indicated above.

TOPIC 2: ROLES AND RESPONSIBILITIES

- Now that you have identified your idea as a group, it's important to also identify the role of every member in the group.
- Each member will take leadership role in the activity they will be assigned.
- However each member will be required to fully participate in each stage for learning purposes.

Roles and Responsibilities

- There are several roles that you can identify in software development.
- For the purposes of this unit each member should be assigned a role but each member will be part of each team. The responsibility of ensuring that the role is done resides on the person assigned the role.
- **Business Owner**
- The business owner makes decisions for the company and is responsible for identifying new possible businesses and partnerships.
- Another responsibility of the business owner is to negotiate the terms of the partnership or establish the prices of a new product.
- Also, the viability of the new project itself is something that is usually run by the business owner. This also includes, for example, checking if the solution is legal and whether the company will face any judicial problems in the future because it violated some patent or committed some illegal market approach.
- It is important that the business owner keeps track of those metrics in order to improve the product and respond to possible market changes.
- Business owners work closely with product managers in order to come up with a concept that can be further implemented by a technical team.

Product Manager

- The product manager can be seen as the orchestrator between the business owners and the development teams. **Common tasks of product managers include:**
- prioritization of tasks,
- supporting non-technical teams such as marketing and content, and gathering feedback, among other tasks
- The product manager role is important since they are responsible for translating ideas and concepts into products or features for the company.
- A product manager needs to possess several skills in order to successfully fulfill the role.
- Soft skills to enable him/her deal with different individuals
- Technical skills
- Preparing the roadmap of a product and making sure the product meets the users' needs.

Designers

- Designers are important .
- They are responsible for creating the interfaces users will interact with, but also bringing some sort of identity to a product or even a company to make it consistent among all platforms.
- They are somehow the bridge that connects the users to the technology a product exposes.
- Understanding how users interact with the products and taking that into account in designing them is a responsibility assigned to design teams.
- When talking about design, two major terms often appear: *UI* and *UX*.

Cont:

- **user interface design**
- UI focuses mostly on the appearance, branding, and consistency.
- **User experience**.
- UX focuses mostly on the interaction the user has with a product (experience), what and when something can be clicked, feedback to the user when some action is finished, etc

Backend developer

- The backend is the entity of a software product that is responsible for receiving requests from the client applications and handling them by running on dedicated servers typically hosted on cloud services or server providers. Amazon web services, Google Cloud platform, and Microsoft Azure Cloud computing etc.
- Backend teams focus on exposing operations, so the frontend applications can retrieve, store, modify, and delete data entities of an application.
- In some cases, frontend and backend implementations will start in parallel, since there are always tasks that can be done without the backend, such as the implementation of the screens.
- Optimization is a big topic when designing backend endpoints, how to fetch data, how to organize data, making them in a way so that they can easily cope with future changes, how to integrate with external services, caches, databases, etc.
- Another key concept that the backend has to take care of is authentication and authorization. Public versus private endpoints, user roles (what users can access or not), revocation of access, etc. must be carefully defined by product managers, as they are part of the product concept in general.
- If the application exposes some sort of online shop or subscription-based services, the backend is also responsible for handling the payments with the Payment Service Providers or, also common nowadays, receiving receipts from the mobile stores, validating those receipts, and acting accordingly.

Frontend developer

- The frontend application of a product is the one that is visible to the end-user.
- Frontend refer to a web application running on a browser in web programming.(for web project)
- Any application that exposes a graphic interface or even command line interface can be considered a frontend application.
- A frontend application is a piece of software that runs on the client side; this includes not only web applications but also mobile and, more recently, TV applications.
- For example in web programming a frontend is a web applications running on browsers using *JavaScript, HTML, and CSS*.

Cont:

- Frontend can be divided into two main modules:
  - **Representation**
  - Representation is what the user sees, the interface, how the elements are rendered, and how to interact with them.
  - **Logic**
  - The logic is everything else that makes it an application, such as fetching data, transforming them to present it to the user, and handling requests, states, validation of data input, etc.
  - Depending on the size of the organization the people doing frontend also do backend.
  - People with the ability to do front and backend are called **full-stack engineers** or **full-stack programmers.** I
  - n summary frontend teams are responsible to implement the interfaces and the logic of the application that interacts with the users.

Quality Assurance (QA)

- The quality assurance (QA) department is responsible for making sure everything that gets to the end user meets the requirements and is working properly.
- Several types of tests will be written, such as:
  - **unit tests-**where components in general will be tested on an isolated manner;
  - **integration tests**-where several components are tested to work together; and also,
  - **functional tests**-to check if the requirements are met given the acceptance criteria.
  - **Regression testing -**is basically making sure that a set of changes (the new version) is not damaging the system as a whole.

Cont:

- **System testing -**is also the responsibility of Quality Assurance teams;
  - it is verifying the product works as expected on all target environments. This can be operating systems, in cases where software gets installed on premises, or, for example checking if a web application can work with all the specified browsers.
- There are also other types of testing, but they are not done by Quality Assurance teams but rather by development and operations teams (DevOps). These include:
  - **stress testing**—making sure the system performs during heavy loads and anticipating how the system behaves if those occur;
  - **performance testing**- to check if the system is performing as expected and answering requests within accepted time frame.
- The product manager is responsible for deciding what to with the issues picked by the quality assurance team. Usually two possible outcomes can originate from that: either the feature gets rejected, and it's iterated from the beginning, passing by all the necessary stages; or the feature gets released as it is, but further development or optimizations will be considered in the next versions.
- To perform testing it is recommended that the team must prepare test data and document the test results.

DevOps (development + operations)

- DevOps teams are responsible for all the operational aspects of the development and infrastructure.
- This means they are responsible for building the continuous integration and continuous delivery pipelines, managing the servers, performing migrations, and doing the actual deployments. This is quite a technical role.
- The teams are also responsible for making sure the current infrastructure can handle the expected load and that it still performs under it by not entering in denial of service.
- The output of this topic is a list of team members with assigned responsibilities.