Week 8

Boundaries

1. What is a boundary in a software system? Give an example in the context of Clean Architecture.
   - Software architecture is the art of drawing lines that I call boundaries. Those boundaries separate software elements from one another, and restrict those on one side from knowing about those on the other. Some of those lines are drawn very early in a project's life even before any code is written. Others are drawn much later. Those that are drawn early are drawn for the purposes of deferring decisions for as long as possible, and of keeping those decisions from polluting the core business logic.
   - You draw lines between things that matter and things that don't. The GUI doesn't matter to the business rules, so there should be a line between them. The database doesn't matter to the GUI, so there should be a line between them. The database doesn't matter to the business rules, so there should be a line between them.
2. What were some of the boundaries from the Warehouse example? The boundary between a QR code parser and the controller that inputs the QR image; the boundary between the presenter that gives item data and the UI that displays it.
3. What are two boundaries in your course project?
   i. The boundary between the MessageManager and Controller
   ii. The boundary between the UserRepo and Repo
4. Briefly, what are MVP, MVC, and MVVM?
   MVP:
   i. The user interacts with the View, but the View immediately asks the Presenter to take over
   ii. The Presenter manipulates the model
   iii. The Presenter updates the View
   MVC:
   i. The user sees the View
   ii. The user interacts with the View, which immediately asks the Controller to take over
   iii. The Controller manipulates the Model
   iv. The Model updates the View
   MVVM:
   i. The user interacts with the View
   ii. The View updates information to the ViewModel
   iii. The ViewModel represents the state of the View - the View is very thin, and is bound to the ViewModel
   iv. The ViewModel passes control to the Model
   v. The Model updates the ViewModel , which is observed by the View

Embedded Ethics Module II

5. What is the explanation for why we should accommodate people with accessibility needs?
   From a moral perspective, we should maximize happiness and ensure everyone can possess the minimum defined capabilities necessary for a dignified life; it is also a legal requirement.

6. Briefly, what are Utilitarianism and the Capability Approach?
   - Utilitarianism: to maximize happiness and minimize pain
   - The Capability Approach: to ensure every possess the minimum capabilities necessary for a dignified life

7. Give an example of a legal requirement related to accessibility.
   Accessibility for Ontarians with Disabilities Act

8. Why might following Clean Architecture make it easier for software developers to implement accessibility features?
   It is easier to change each component we can mainly focus on changing the frontend components.