**Game Hardware Exam Notes**

**Graphical user interface, text, application

Description automatically generated**



**Design Process**

* The design process is a process based on identifying users needs and iterating various times to solve said needs.
* The process consists of the following Steps
  + **1. Need Finding**
  + **2. Ideation**
  + **3. Product Concept**
  + **4. Prototyping**
  + **5. Development, Commercialization & Maintenance**
* When designing we must consider the user, “who are we designing for?”

**Identifying a Need and Attempting to create a Solution**

* In real life, there are problems that people encounter for almost anything. People have had some clever solutions to these problems which addressed users needs.
* **For example:** You need to collect credits or attain an achievement that requires a monotonous task. What do you do?
  + Instances where people have use elastic bands and wrapped them around the analog sticks to make the character or something move in the game, so they are not idling (Hatching eggs in Pokémon)

**Use Cases**

* A use case is a written description of how users will perform tasks. It outlines from a user’s point of view, a system’s behaviour as it responds to a request. Each use case is represented as a sequence of simple steps, beginning with a user’s goal and ending when that goal is fulfilled

**Elements**

* **Actor –** anyone or anything that performs a behaviour (who is using the system)
* **Stakeholder –** Someone or something with vested interests in the behavior of the system under discussion (SUD)
* **Primary Actor –** Stakeholder who initiates an interaction with the system to achieve a goal
* **Preconditions –** What must be true or happen before and after the use case runs
* **Triggers –** This is the event that causes the use case to be initiated
* **Main Success Scenarios –** [Basic Flow] – Use case in which nothing goes wrong
* **Alternative Paths –** [Alternative Flow] – These paths are a variation on the main theme. These expectations are what happen when things go wrong at the system level

**Elements Use cases DO NOT Include**

* Implementation-Specific Language
* Details about the user interfaces or screens

**Use Case Example – Nintendo Labo Fishing Rod**

* Main Use: Reel and Aim to catch a fish using motion controllers
* First identify who: **Player** holds a game controller and uses arm movements combined with wrist movements and button pressing to catch a fish using a virtual fish rod
* Then:
  + **The Player** waits for the fish to **latch onto the hook of the fishing rod**
  + **The Player** reels the fish by **turning the reel to bring the line back to the surface. Raising the fishing rod will also help to pull the line up from the ocean**
* Use cases range from **simple, middleweight and heavyweight**, each with increasing elements of the Use case.
  + **Example:**  A simple use case will contain the use case, actor(s) and the basic flow of the use case

**What is the Goal of Use Cases?**

* Fill a certain Gap in the market
* Eliminate a dissatisfaction with an existing product
  + Reduce Cost
  + Increase reliability/performance
  + Improve its user friendliness
  + Change its appearance
  + Improve aesthetic appeal
* Product to fill a market gap: Powerful portable gaming consoles (Nintendo Switch, Steamdeck)

**Design Process**

**Defining the Problem**

* Objectives/Goals
* Definition of Technical terms
* Features:
  + Required (Must-have) features
  + Desired (nice-to-have) features
* Constraints:
  + Hard (must not have)
  + Soft (Better not to have)
* Present state
* Criteria to evaluate the design

**Identifying the problem and Users**

* Who are my users?
  + **External Users:** Ones that purchase a product or service of a company
  + **Internal users:** Different groups involved in the design process within the company (management, manufacturing, sales, technical service, or your instructors!)
* What does the user want?
  + **How can I provide it?**
* Keep in mind that the person who buys is not necessarily the person who uses the product (e.g, school utensils, parents buy, children use)

**Obtaining Information from Users**

* Observation
* Interviews/surveys
* Focus groups

**Observation for Game Hardware**

* Video recording
* Note taking
* Focus group
* Usability settings

**Example:** Controller handling

* How would you hold a Gamecube controller?

**Getting Familiar with your idea**

* Create a system architecture where inputs, outputs and possible systems processing the information based on the chosen use case are defined.
* System architecture is revised **after the Design thinking is done** to enhance with possible improvements
* When analyzing the controller/idea of choice, think of its usability, effectiveness, and room for improvement

**Gathering Info from Users**

* Interviews
* Focus Groups – **Small group discussion led to learn about a topic**
  + Lead by a facilitator
  + The discussion is planned to avoid any discomfort during the discussion
  + It promotes an open communication
  + Expect open-ended responses
* User Surveys
* Observation

**Design Thinking**

* Design thinking is a design methodology that provides a solution-based approach to solving problems

**How:**

* By understanding the human needs involved
* By re-framing the problem in human-centric ways
* By creating many ideas in brainstorming sessions and by adopting a hands on approach in prototyping and testing

**Design Thinking Steps**

* **Empathising:** Understanding the human needs involved
* **Defining:** Re-framing and defining the problem in human-centric ways
* **Ideating:** Creating many ideas in ideation sessions
* **Prototyping:** Adopting a hands-on approach in prototyping
* **Testing:** Developing a prototype/solution to the problem

**Problem Definition**

* Problem definition is what you base your project around that you create to address these needs
* Not defined from your own perspective
* Use experiences from others

Graphical user interface, text, application

Description automatically generated

**Electronic Fundamentals**

Diagram, schematic

Description automatically generated

**Current Flow**

* The battery symbol indicates that a difference of potential (also called voltage is being supplied to the circuit.
* Voltage causes the current to flow if there is a complete circuit present as shown in the figure
* The current flows in the opposite direction

**Formula**

* **V = I X R**
  + V: Voltage
  + I: Current
  + R: Resistance

**Examples:** What is the voltage for each combination of resistance and current values?

1. R = 20 ohms, I = 0.5 amperes

V = 0.5 \* 20

**V = 10**

1. R = 560ohms, I = 0.02 amperes

**V = 11.2**

1. R = 1000Ohm, I = 0.01 amperes

**V** = 0.01 \* 1000 -> V = 10

**Solve for Resistance**

V = I X R

**R = V/I**

1. V = 1 Volt, I = 1 ampere

R = 1/1

R =

1. **V = 2, I = 0.5**

R = 4 Ohms

1. V = 10, I = 3

R = 3.3333Ohms

**Solve for Current**

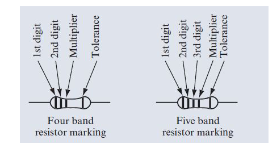
**I = V/R**

= 12 / 10ohms

= 1.2amperes

**Resistors**

* Resistors are used to control the current that flows through a portion of a circuit



Examples

Table

Description automatically generated

Resistor 2

Five Bands

1. Green: **5**
2. Red : **2**
3. Brown : **1**
4. Black 1x Multiplier
5. Red 2% Tolerance

521 X 1

= 521

Therefore, resistor 2 is equivalent to, 521 Ohms 2%

**Resistors in Series**

**Graphical user interface, text, application, chat or text message

Description automatically generated**

**Resistors in Parallel**

Graphical user interface

Description automatically generated with medium confidence

**Parallel Resistors Examples**

1. R1 = 1,000 ohms, R2 = 500ohms

Rt = 1/1000 + 1/500

Rt = 0.003

**Example**

* 6 volts is applied across a resistor, and 5mA of current is required to flow through the resistor. What value of resistance must be used and what power will it dispute

**Solve for resistance**

**R = V/I**

= 6/0.005 (5 mA)

**=1,200 Ohms (1.2 killaohms)**

**Solve for Power**

**P = V X I**

= 6 \* 0.005

= 0.03 watts (30 mW)

**The Voltage Divider**

**Electronic Components that Constitute a Game Controller**

* Variable Resistors
  + Change their resistance based in response to some physical input
* Sensors
* Push Buttons
* Wiring
* Actuators
* Thumbstick
* Push buttons

**Potentiometers**

* **Potentiometer:** A component used for measuring the unknown voltage by comparing it with the known voltage
* **Rotary vs Slider:** Rotary potentiometers use a rotating knob to control the wiper leg while slider potentiometers use a slider
* **Knobs**
* **Mount**
* **Linear vs. Logarithmic**

**Sensors**

* An electrical sensor is a device that detects a physical parameter of interest (e.g. heat, light, sound) and converts it into electrical signal that can be measure and used by an electrical or electronic system

**Common Sensors**

* Pressure sensor
* Photo sensor
* Thermal Sensor
* Motion Sensor

**Motion Sensors**

* Internal measurement units
  + Each tool in an IMU is used to capture different data types:
  + **Accelerometer**: measures velocity and acceleration
  + **Gyroscope:** Measures rotation and rotational rate
  + **Magnetometer:** establishes cardinal direction (directional heading)

**Actuators**

* Electrical Actuator is an electromechanical device that converts electrical energy into mechanical energy
  + Motors
  + Speakers

**CAD**

**CAD Standards**

* A set of guidelines for the way computer-aided drafting (CAD), drawings should appear, to improve productivity and interchange of CAD documents between different offices and CAD programs, especially in architecture and engineering

Why are standards so important?

* To make sure no errors occur during its manufacturing and others can work with your designs!

**Technical Drawing**

* Detailed, accurate, and precise
* Use symbols and conventions to transmit technical information
* Fundamentals of technical drawing are required
* You only need **4 views** ( Front, Top, Side and Isometric)
* Isometric view **never contains dimensions**
* **Dimensions should not be grouped inside the technical drawings**

**Extension Lines**

* Extension lines should not cross dimension lines, and should avoid crossing other extension lines whenever possible
* When the location of the center of a feature is being dimensioned, the center line of the feature is used as an extension line
* When a point is being located by extension lines only, the extensions must pass through the point

Diagram

Description automatically generated

Diagram, engineering drawing

Description automatically generated

**Assemblies**

* An assembly is a collection of components that function as a single design
* Assemblies need components
  + Components keep the assembly together
  + Components also help transfer movement

**Showing Details**

* Make sure you have the proper scale
* Double Click on the view
* Change the scale

Under Create tab, press the detail view button. This button will allow you to drag over a certain area and provide a component of the 3D model with more detail

* To add detail focus, on areas that are difficult to understand

**Moreover**

* Assembly Drawings must have several views to show how parts fit together
  + Section Views to show how parts fit to eliminate hidden detail
  + Dimensions to indicate range of motion or overall size of assembly for reference purposes
  + Individual components identified with balloons and leader lines

**Bill of Materials**

* Make sure to include all items
* However, you only 3D model the ones you are creating
* Existing commercial models are only referenced and depicted by imported 3D models or placeholders.

**Ensure Numbering System flows correctly for Parts List**

* **Slides in the lecture for Bill of Materials are wrong**
  + Not every item has a specific material (From example: Arduino, other commercial components)
* **Bill of Materials should go in the same page as the exploded view**
* **Resize the sheet according to**

**Be Consistent**

* Be careful about the location of pieces within the assembly
* The ID number of the pieces must match their blueprint ID
* Identical pieces will share the same ID in the assembly
* Us numbers starting from 1
* Numbering is consecutive and in a logical order
* **PIECES ARE NUMBERED CLOCKWISE**

**Drawing Numbering – follow the item numbers on the exploded view**

* Assembly goes first – no dimensions
* Exploded View with bill of Materials – no dimensions
* Assembly starts at Part 0 (The origin)
* Every other part that WE create needs to be listed

**Tolerance**

* The total amount a dimension may vary and is the difference between the upper (maximum) and lower (minimum) limits
* Tolerances are used to control the amount of variation inherent in all manufactured parts. In particular, tolerances are assigned to mating parts in an assembly