1. **Jak uživatel vnímá interaktivní systém?**

Jako uživatel – tedy vnímá jen to co vidí a to na co může kliknout. Neřeší v čem je to napsane a jaké patterny jsou použity apod. Vidi jen tu slupku designu.

1. **Co uživatel očekává od UI?**

Že bude intuitivní, rychlý, splní svou funkci, bez chyb, jednoznačný, zapamatovatelný

1. **Bude si uživatel stěžovat na využití UI?**

Nebude. Uživatele si nestěžují ani nechválí. Často si myslí, že je to jejich chyba.

1. **Rozdíl mezi SW inženýrem a HCI specialistou v oblasti vývoje UI?**

Sw inženýr přemýšlí spíš nad tím jaké patterny použít, jakou technologii. Hci chce se více soustředí na tu „lidskou stránku“ chce, aby systém byl jednoduchý, intuitivní apod. po stránce designu.

1. **Základy post-WIMP UI**

Wimp UI je jen 2d, objekty, drag and drop (windowsy).

Ale post-WIMP UI reraguje na zvuk, dotyk, pohyb. Může být 3d

1. **Jaké jsou omezení konceptu WIMP GUI?**

Nereaguje na pohyb, zvuk apod..

1. **Definuje zvukové rozhraní novou třídu UI?**

Ne, spadá pod post-WIMP UI

1. **Proč je MVC důležité pro vývoj UI?**

Protože rozděluje implementaci appek do patternu. Do 3 nezávislých modulů. A to umožňuje udržet velké aplikace přehledné a jednoduché na připadné vylepšování či údržbu

1. **Popiš ovládací prvek UI Window. Jaký je rozdíl oproti Page?**

Ve window je appka zobrazena, page je ve window a ve window může být několik pagů

1. **Co je interakční designový vzor (design pattern)? Vysvětlete na příkladech.**

Je designový pattern, který poskytuje základní a opakovatelné řešení pří problému při tvorbě designu rozhraní. Př: multi-level undo

1. **Co je deklarativní popis UI?**

To je kod v XAMLU – rozmístení a popis komoponent

Procedurální – jsou procedury v kodu např: c#

1. **Popiš MVC model. Jaké má omezení?**

Model – má v sobě backend, databazi, fungovani apod -> komunikuje s View

View – zobrazuje to co vidí uživatel – kliky posílá na controler

Controler – controler dostane něco že se ve view na něco klikne a zavola konkretní věc z modelu

Omezení – model by neměl měnit view, View a controler nejsou oddelene, uživatel pracuje s obouma. View ma část logiky v modelu

1. **Jaké jsou rozdíly mezi MVC a MVP?**

Model, view, presenter. Presenter je mezi view a modelem. Řídí komunikaci mezi nimi. Uživatel teď už komunikuje pouze s View, který bere input a zároveň vykresluje output. Presenter volá potřebné věci z modelu.

1. **Popiš observer pattern. Jak lze udělat Model nezávislý na View a Controller?**

Observer – má své observery. Jakmile se v něm něco změní – informuje posluchače. Viz mapa v střílečkách. Když hráč změní svoji pozici notifikuje ostatní že změnil pozici -> updatuje observery

1. **Popiš data binding pattern. Jak ho lze použít pro implementaci MVVM modelu?**

Model, viewmodel, view. To same jako presenter, jen komponenty z view mohou být nabindované na viewmodel. Když se změní jedna změní se druha(two-way binding)

1. **Jak lze dosáhnout content customization? Napiš kód pro vytvoření tlačítka s obrázkem.**

<Button>

<StackPanel>

<Image ….\>

<Label ….\>

</StackPanel>

</Button>

1. **Jak lze přizpůsobit komponenty pomocí Style?**

Stylem lze pomocí setterů měnit vlastnoti určité komponenty(button, circle apod..)

1. **Jak lze přizpůsobovat pomocí triggrů? Jaké triggry existují?**

Triggery jsou jako styly, ale spusti se až za určité podmínky.

Data trigger

Event trigger

Property trigger

1. **Jak lze udělat vlastní seznam položek? (custom list of items)**

CustomControly se dají např do stack panelu.

(Tak prý využijeme trigery a setery pro interaktivitu a definujeme cil content templatu a jednotlive položky v item template)

1. **Jak lze využít ContentTemplate pro přizpůsobení**

Nevim..mrdat

Vytvorime custom componentu a vložíme ji do contenttemplate, která swapne strom

1. **Porovnej UserControl a CustomControl**.

CustomControl si udělame každý vlastní, lze stylizovat.

UserControl už existuje, nelze stylizovat.

1. **Jak se definují validační pravidla s Data source exception?**

Binding.validationRules->ExceptionValidationRule

1. **Jak se definují validační pravidla s Data error Interface?**

IDataErrorInfo -> DataErrorValidationRule

Asynchronní volání

1. **Jak se definují validační pravidla s vlastní validační třídou?**

Vytvoříme novou třídu dědíci z validationRule a přepíšeme metodu Validate.

1. **Jak lze zobrazit chybové stavy při použití ControlTemplate a Style?**

Objektu dame error template, který definujeme pomoci:

-Control template

-Style, trigery

1. **Jak implementovat navigaci v Okně? Kdy se používá Page?**

Jako v browser navigation. Odkazuje cestu na jednotlive pages

1. **Jaký je proud událostí mezi stránkami? (Pages)**

Navigation process-> loading -> init 2nd page -> unload 1st page -> navigated -> LoadCompleted -> 2nd page loaded

1. **Jak posílat a získávat data mezi stránkami?**

Forward pasing - navigate parametr, událost LoadCompleted

Returning data-  používá se PageFunction místo Page

1. **Popište rozdíly mezi Webovou mobilní aplikací a Nativní mobilní aplikací.**

Webova funguje na všech platformách – je to jakoby web. Má vyšší dosah, ale menší strávený čas.

Nativní – více využívá HW, pouze na jedné platformě. Nižší navstevnost než webova, ale vyšší stravený čas.

1. **Jak implementovat hybridní mobilní aplikaci? WebView vs Compiled approach.**

WebView – jako web a pak se zkompiluje do webu

Compiled – jeden kus kodu a zkompiluje se do verzi pro danou platformu

1. **Jaký je rozsah UI specifikace?**

Screenflow, popisky, obrazky, uživatele, scenaře, obsah, design iterakce, indikator vyzralosti, dekompozice, konkretni komponenty (button apod.)

1. **Definuj formativní vyhodnocování. Jak je začleněno do vývojového procesu?**

Mělo by být v každé fázi vývoje. Viz jestli se splnil učel jestli to funguje apod.

1. **Co by nemělo chybět v UI specifikaci, aby byla implementace více efektivní?**

Komponenty a tyhle sračky. Maturify of spcefication, vyhodnocovani, apod

1. **Co znamená indikátor vyzrálosti v UI specifikaci? (maturity indicator)**

Připraveno na přímou implementaci - většina relevantních situací, pořádné uživatelské testování (hodně účastníků a iterací), stabilní design, málo implementačních iterací

Implementace s konzultacemi - většina relevantních situací, ale jen zběžné uživatelské testování (málo účastníků i iterací), design se může měnit

Nutné vyhodnocení designu - nereflektuje většinu relevantních situací, zběžné testování, design se výrazně mění, implementace se často zahazují

1. **Co je accessibility API? Popiš architekturu a mapování MVC / MVVM modelu.**

Rozhrani pro lidi s postižením, aby mohli používat appky.

Kvuli omezenim mohou byt určitě View a controlery nepoužitelné.

1. **Definuj Accessibility RIA.**

Rich internet application

Webova aplikace, která požaduje pluginy adobe apod. (vyhoda pro hendikepované)

ARIA to samé ale jen nějaký vývojový standard

Hendikepovaných je ve společnosti cca 1%

1. **Popiš kroky k vytvoření ARIA aplikace.**

Nevim, vole mrdat.

Pristupnost RIA komponent – vytvoreni offline prototypu, zjednoduseni architektury pridani aria API apod.

Pristupnost RIA aplikace – ARIA heuristika

Testovani aplikačni pristupnosti – musí testovat handicapovaní uživatelé

1. **Co by mělo být testováno při testování UI?**

Chybovost, zapamatovatelnost, jednoduchost, intuitivita, čitelnost, přehlednost, použitelnost, odolnost proti chybám

1. **Jaké jsou metody tvorby testovacích skriptů na UI?**

-Programem

-nahravani – testovaci kroky se nahraji automatizacnim kodem a pak se provedou na appce

GUI ripping – zvoli se začátek a konec a vygeneruji se všechny mozne scenare mezitím

1. **Jak lze simulovat lidský faktor při testování UI?**

MONKEY testing – random kliky – není zde lidsky faktor

Lepší opička – zařazeny časové limity, aby to působilo víc lidsky

1. **Co je Usability testing?**

Test použitelnosti. V podstatě to co se má testovat v UI. Takže testujeme zda uživatel dělá hodně chyb, jak je s ní uživatel schopen pracovat po seznamení/bez seznámení. Za jak dlouho je ji schopen znovu používat když ji dlouhou dobu nepoužíval a chce začít znovu. Uživatelská spokojenost – zda ho to uspokojí, když splnil úkol.(dle průzkumů ženám vyhovuje lehké zavibrování)

1. **Jaké jsou evaluační metody s / bez uživatele?**

Bez uzivatele – kognitivni pruchod, heuristicke vyhodnocovani(rizeno mnoha experty)

S uživatelem – v laborce/v jeho prostreti

1. **Co je Data mocking? Co je Wizard-of-Oz?**

Wizard of oz – systém zatím neexistuje – nahrazen clovekem co ho simuluje

Data mocking – data která neexistuji, ale potrebujeme je – dame primo do kodu

1. **Klasifikace herního UI + popis každého typu**

Ty vole kamo mě picne

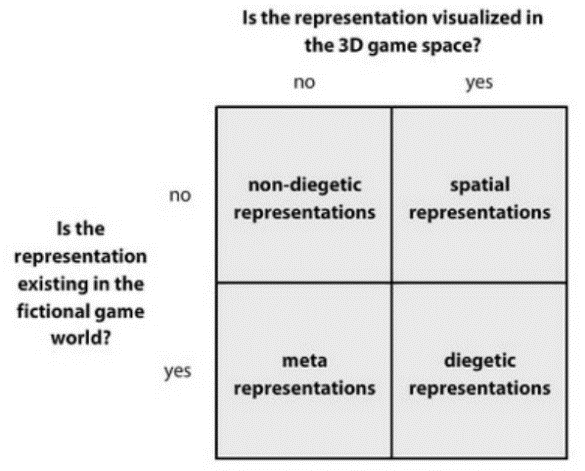
Takže tu máme

Ten o kterym herni postava neví : viz ping v csku, killy

Ten o kterym herni postava ví- sama ho pouziva – mapa v far cry

Ten o kterym herni postava nevi a jsou aby hracovi pomoholi: pr prihravka ve fotbale

Ten které jsou soucasti pribehu, ale ne herního světa. Pr krvava obrazovka apod



1. **Proč stále existuje více druhů herních enginů a ne jen jeden?**

Protože je strašně moc typů her a jeden by nevyhovoval všem

1. **Co je rozhraní HID a jaké jsou druhy?**

Human interface devices

Neboli zarizeni, kterým může uživatel ovládat hru

Př: ps ovladač, klavesnice

1. **Jaké jsou druhy věrnosti interakce?**

Realisticka – existuje v realnem svete

Nerealisticka – není jako v realnem svete, je lehce lepší

Magicka – totalni chaos, letani teleportace

1. **Jaké jsou referenční rámce ve VR?**

Ještě jednou? Nerozumim otázce…a taky ji nevim xD

Jo aha už vim. Je to proto aby systém VR mohl sledovat a menit prostředí podle toho co děla a jak se hýbe uživatel.

world reference frame, real-world reference frame, referenční rámec těla, ruky, hlavy, očí