

9. Prototipiranje

KAKO razvijemo PROTOTIP rešitve in zakaj je le-ta koristen?

Faze razvoja IS



KAJ je potrebno narediti? Zahteve IS se pregledajo in strukturirajo.

KAKO bomo to naredili? Zahteve se zapišejo v formalni obliki dokumentacije (diagrami, dokumenti) in pripravi se načrt izvedbe.

Programiranje IS.

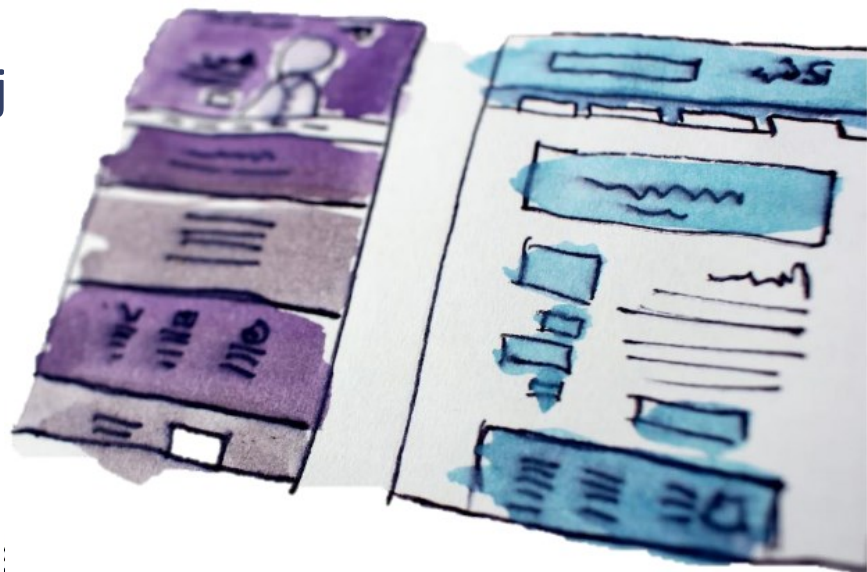
Ali obstajajo **napake** in KAJ ne dela prav?

Prenos IS v realno uporabo. Pravimo, da gre IS **v produkcijo** – za uporabo končnih uporabnikov.

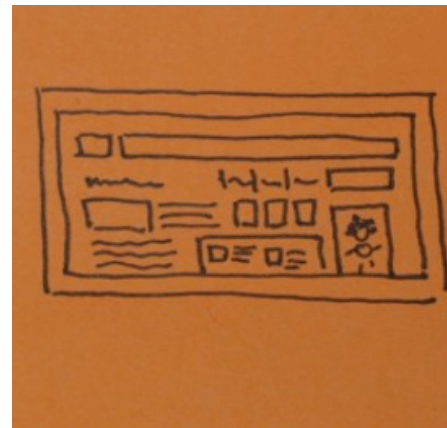
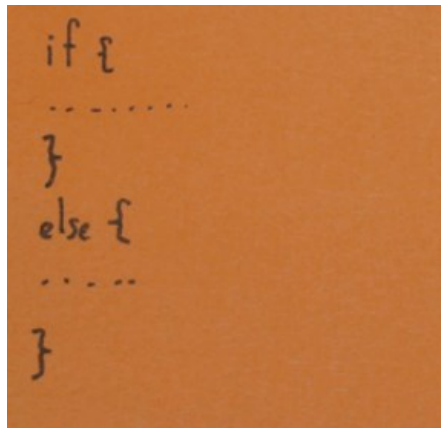
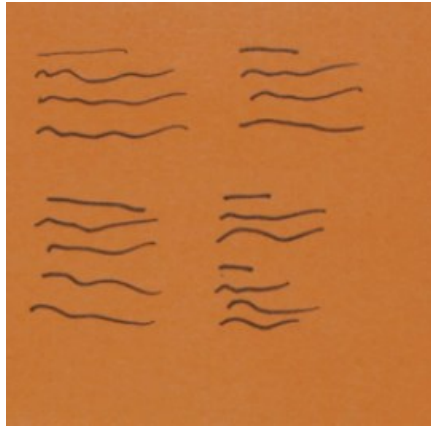
Popravki in dopolnitve IS. Odpravljanje napak in nadgradnja / dodajanje novih funkcionalnosti.

Kaj je prototip?

- Prototip je **delovni model** bodoče rešitve – v našem primeru bodočega IS
 - Še vedno smo pri **modeliranju IS**
- Prototipi so bistveno bolj kompaktni/zgoščeni od besedila
 - Ni potrebno veliko časa oz. truda, da jih razumemo
 - Poenostavijo razvoj
 - Zmanjšajo čas, potreben za razvoj
- Prototipiranje se uporablja na mnogih področjih
 - Arhitektura, inženirstvo, avtomobilska industrija, umetnost, **programska oprema**



O čem govorimo?

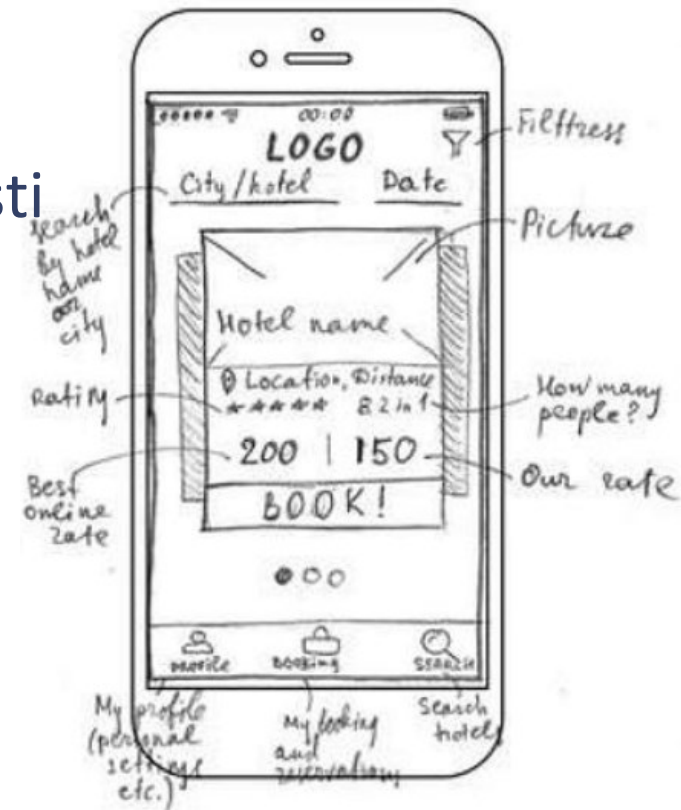


Zakaj prototipi?

- Bolje **spoznamo** načrtovan izdelek
 - Načrtovalci dobijo vpogled v obseg in zahtevnost rešitve
- Pridobimo **povratne informacije**
 - Bodisi od identificiranih strank ali od naročnika
- Pridobimo informacije o **možnih problemih** pri izdelavi (implementaciji rešitve)
 - Od naših razvijalcev
- Pridobimo možnost **spreminjanja oz. prilagajanja** načrtovanega izdelka pred samo izdelavo
 - Kar zmanjša možnost razvoja neustrezne rešitve

Vrste prototipov – skelet (*wireframe*)

- **Skelet** je prikaz razmestitve elementov našega IS
- Pomaga nam definirati informacijsko arhitekturo (izmenjavo informacij v aplikaciji)
 - Lažje definiramo postavitev, če vemo katere informacije bodo posredovane in pridobljene od uporabnika
- Pomaga določiti potek funkcionalnosti (zgodbo, menjavo prikazov)
- Naredi ga lahko kdorkoli (tehniško znanje razvoja ni nujno potrebno)
- Primeren je za validacijo rešitve pri naročniku in potencialni stranki
- Ne vsebuje preveč vizualnih elementov, ki bi bili moteči



Vrste prototipov – model (*mock-up*)

- **Model** služi kot osnova za oblikovanje naše rešitve
- Pomaga nam pri optimizaciji uporabniškega vmesnika in uporabniške izkušnje
- Naredijo ga oblikovalci
 - Medtem ko se skelet ne osredotoča na dizajn, pri modelu poskrbimo za lepo oblikovanje
- So primerni za validacijo uporabniške izkušnje in vmesnika pri potencialnih strankah

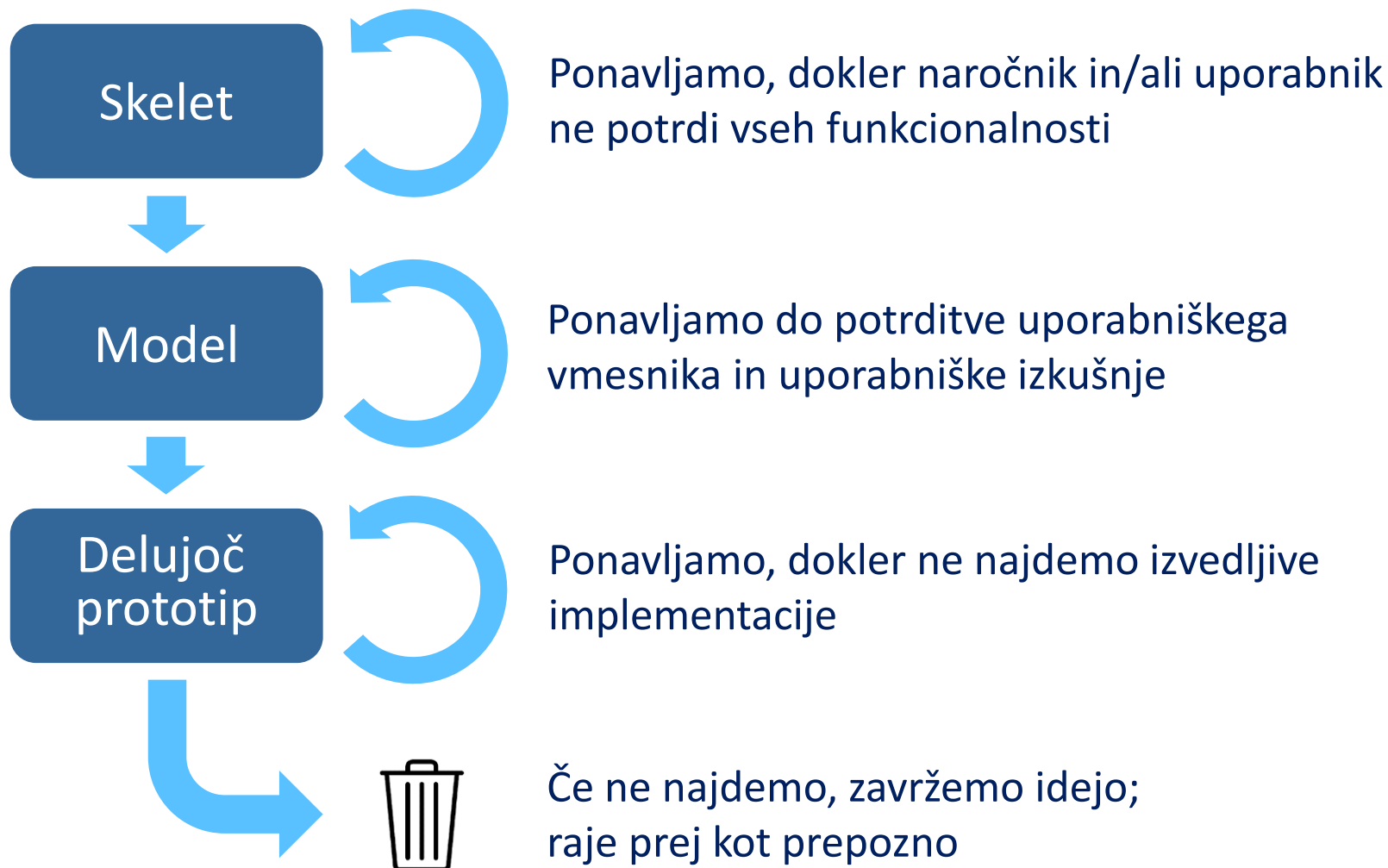


Vrste prototipov – delujoč prototip (*working prototype*)

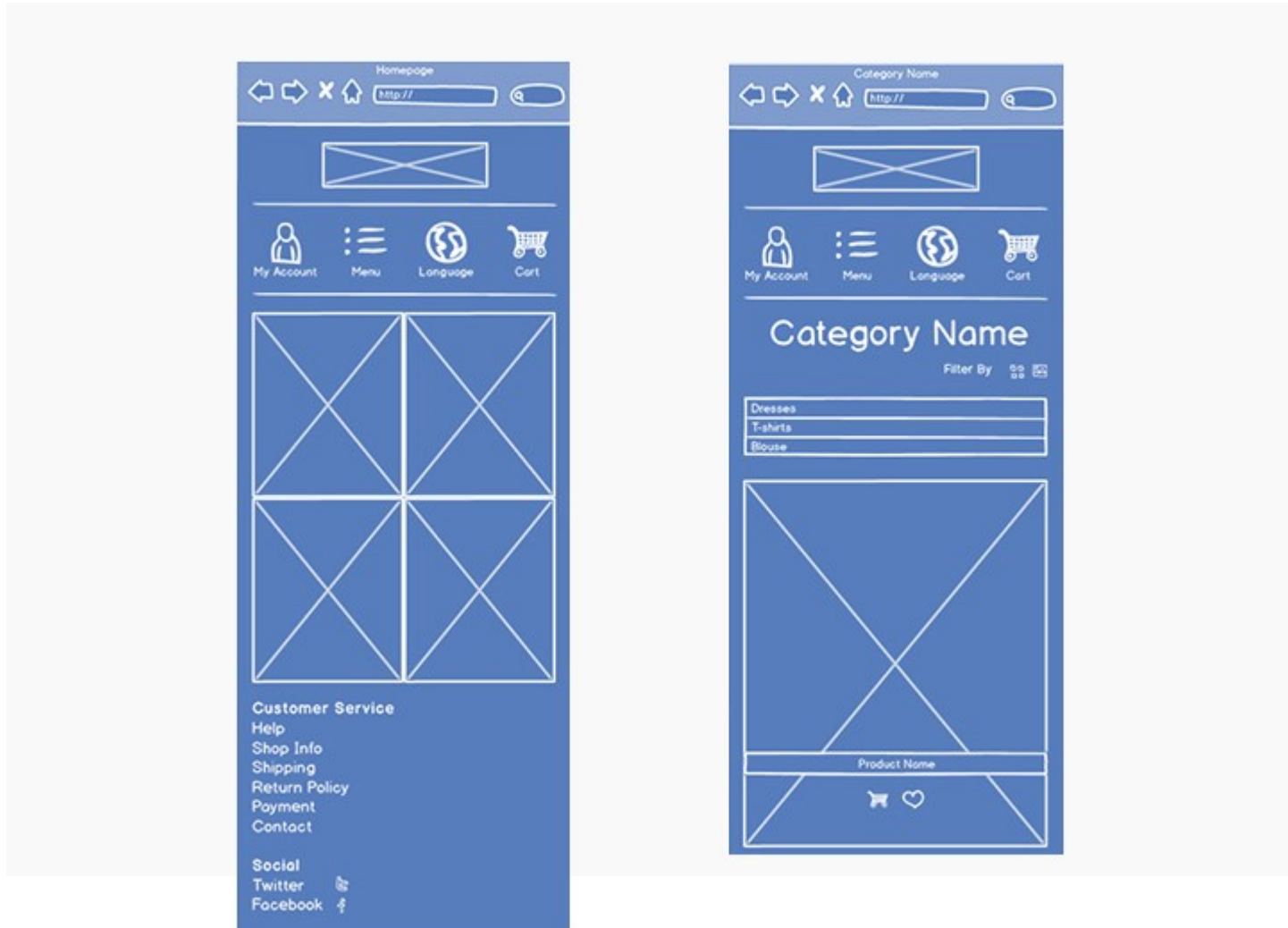
- **Delujoči prototipi** so na hitro in grobo implementirane rešitve, ki služijo potrditvi možnosti izvedbe rešitve
 - Funkcionalnosti v ozadju (ki se ne vidijo) dejansko niso implementirane
- Za kreacijo takih prototipov se porabi nekoliko več časa, in jih morajo razviti razvijalci
- Se praviloma ne uporabijo za osnovo pri implementaciji dejanske rešitve
- So primerni za demonstracijo potencialnim strankam
 - Npr. prvi iPhone na predstavitvi dejansko ni deloval



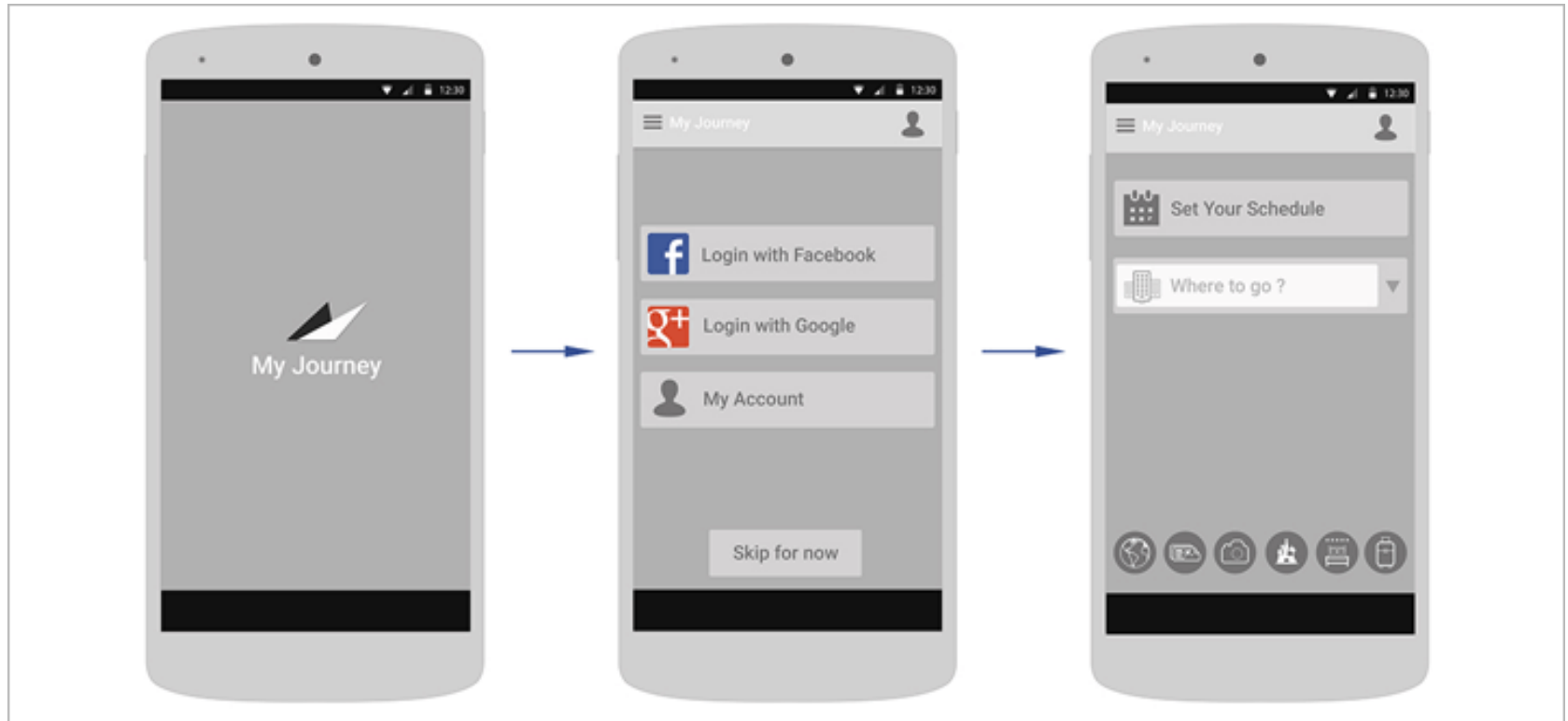
Potek prototipiranja



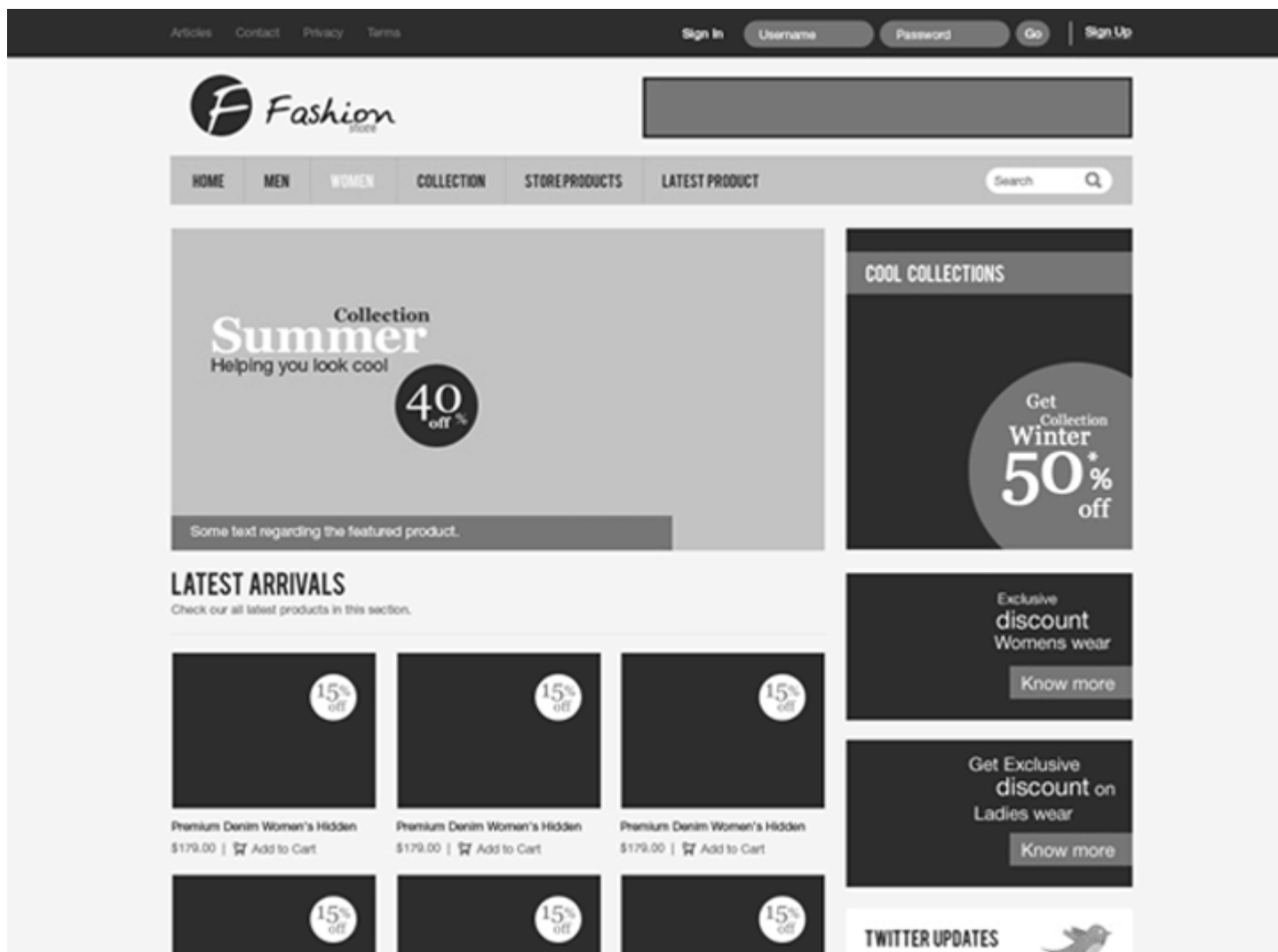
Primer: mobilna aplikacija e-poslovanja



Primer: mobilna rešitev za potovanja



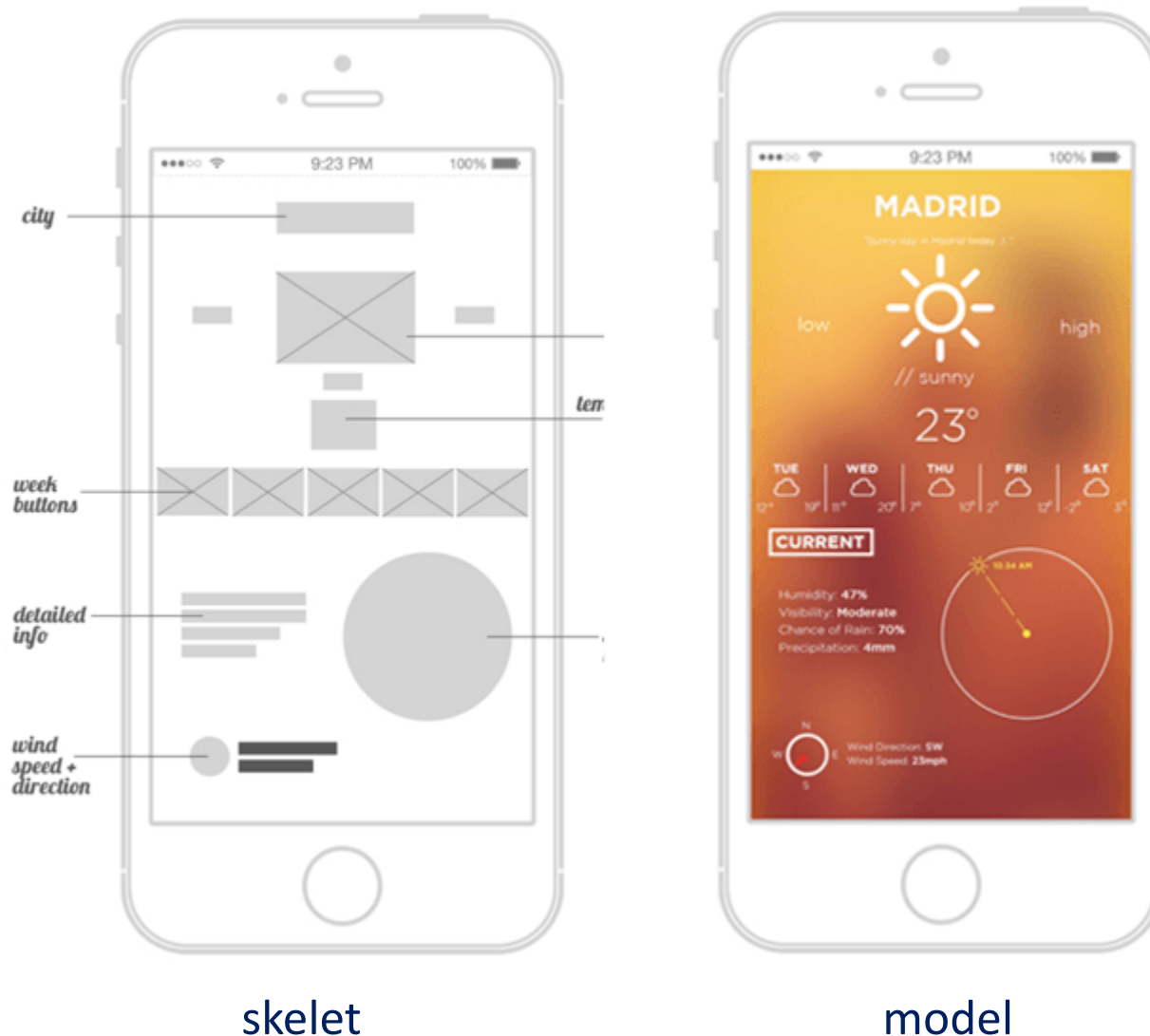
Primer: spletna trgovina



Primer: mobilna aplikacija za skejterje



Primer: vremenska mobilna aplikacija



Pogosta napaka: preobilje informacij



slaba rešitev

| PARKING SCHEDULE | | | |
|--------------------|----------|----------|----------|
| | M-F | SAT | SUN |
| 7am | (P) FREE | (P) FREE | (P) FREE |
| 8am | (R) | (P) 1 HR | |
| 8 ³⁰ am | (R) | (R) | |
| 4pm | (P) 1 HR | (P) 1 HR | |
| 7pm | (P) FREE | (P) FREE | |

dobra rešitev

Pogosta napaka: neinformativnost



BEGINNER USER EXPERIENCE COURSES



Gestalt Psychology and Web Design: The Ultimate Guide

100 %
BOOKED

CLOSED FOR ENROLLMENT



GO TO COURSE

STARTS IN
0 10 47 45
days hrs mins secs

98 %
BOOKED

VIEW COURSE



Information Visualization

STARTS IN
1 10 47 45
days hrs mins secs

95 %
BOOKED

VIEW COURSE



The Psychology of Online Sales: The Beginner's Guide

STARTS IN
2 10 47 45
days hrs mins secs

91 %
BOOKED

VIEW COURSE

Agilno prototipiranje

Kako povečamo možnosti, da bo naša rešitev med uporabniki naletela na dober odziv

Agilno prototipiranje

- **Agilno prototipiranje** temelji na kratkih ciklih načrtovanja in razvoja prototipa
 - Običajno vodi v agilni razvoj rešitev
- 1. Napredujemo **hitro skozi faze** prototipiranja
- 2. Hitro **pridobimo povratne informacije** (feedback) naročnikov, strank in naše razvijalske ekipe
- 3. Hitro **prilagajamo** načrte in prototipe
- 4. **Po potrebi zavržemo** narejeno delo, če se izkaže, da ni pozitivnega odziva
 - Prej zavržemo neustrezne rešitve, hitreje napredujemo!

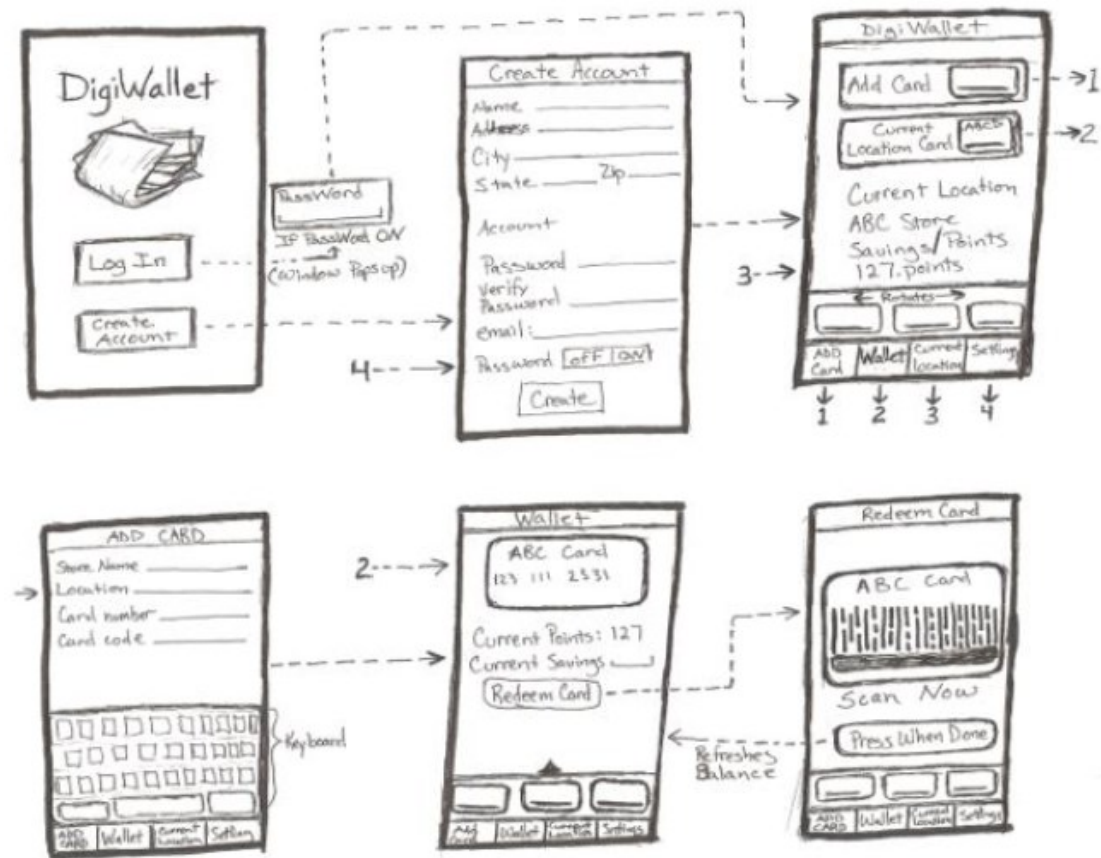
Dobre prakse

- ❑ Skeleti naj bodo enostavni in brez odvečnih gradnikov, informacij in oblikovanja
- ❑ Pri modelu naj bo jasno razvidna pot zgodbe ene funkcionalnosti
- ❑ Jasno naj bo razvidna akcija za nadaljevanja poti



Dobre prakse

- Različne funkcionalnosti razdelimo na več ločenih prikazov, da povečamo preglednost
- Pazimo, da prototip ne postane nepregleden



V službi uporabnikov (1/4)

- Ključnega pomena je, da našo rešitev **oblikujemo za uporabnika!**
 - Uporabimo obstoječe podatke – iz uporabniškega testiranja, sporočil strank, iskalnih dnevnikov, spletnih profilov, blogov, zbranih predlogov, ...
- Ko zberemo čim več podatkov, na njihovi osnovi kreiramo **“persone”**
 - Persona naj čim bolje predstavlja, kar je strankam všeč, česa ne marajo, njihovo ozadje in obnašanje



Toby – “Fashion Phone Upgrader”

“One year in phones is a long time”

Toby loves technology and has to be seen with the newest and coolest digital gadgets. His phone is not just about making calls; he loves using its wealth of features for everything he can: surfing the web, writing emails, social networking and using it as a personal organiser.

Because he gets bored quickly with his phones, Toby is always looking for the latest toy and pays attention to new releases. He frequently upgrades part way through his contract and is willing to pay the upgrade fee to get the best phone. To him, a contract is a mere inconvenience, but something he endures to get a bigger discount off his new phone

Behaviours

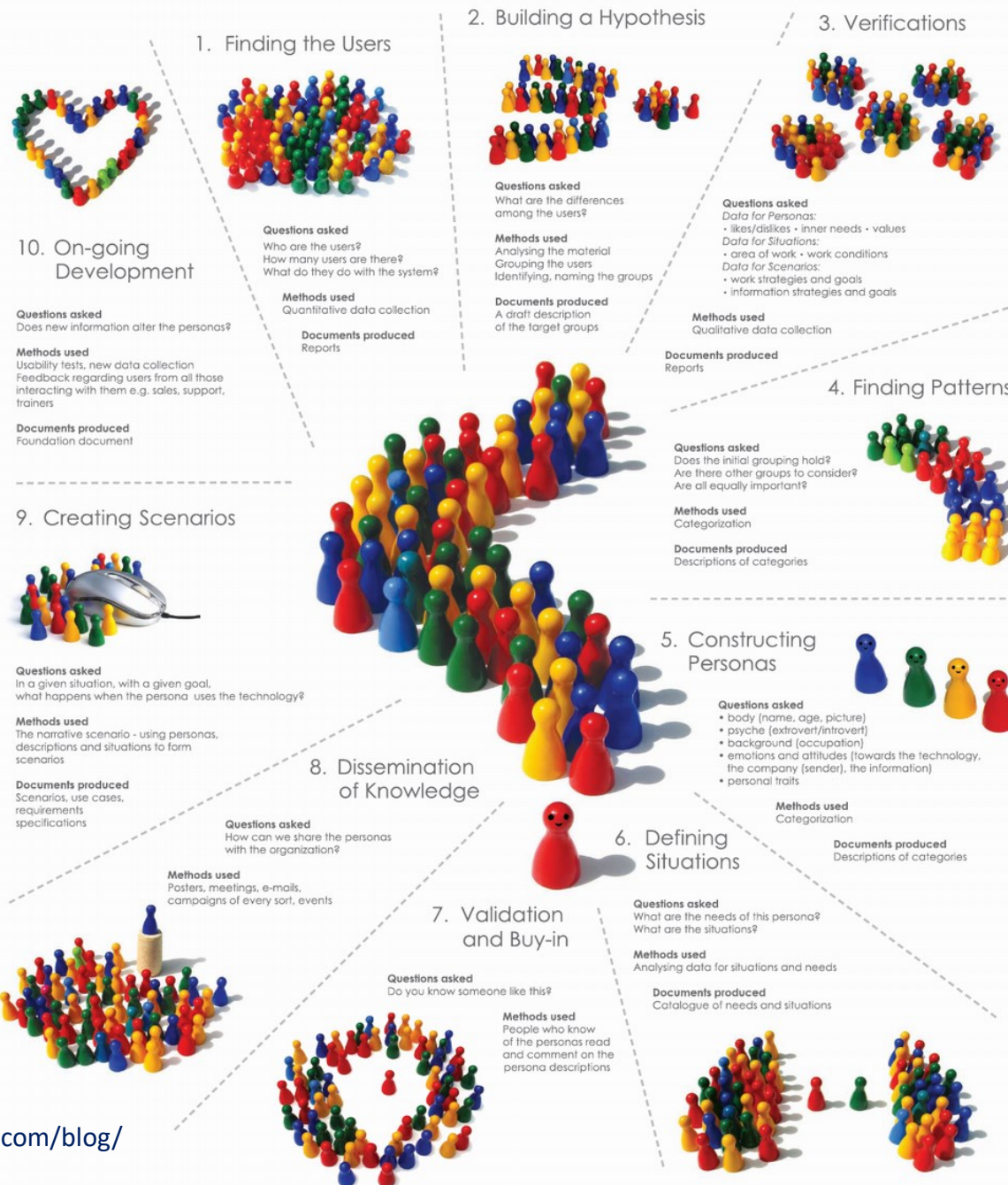
- Handset change reason: Want (green dot) / Need (grey dot)
- Phone perception: High tech toy (green dot) / It's a tool (grey dot)
- Handset discovery: Existing (green dot) / Churn (grey dot)
- Interest in new phone: Always looking (green dot) / Only when needed (grey dot)
- Priority in a handset: Features (green dot) / Price (grey dot)
- Phone life expectancy: A long time (green dot) / Not very long (grey dot)
- Would change provider: Yes, for the right phone (green dot) / Current provider is fine (grey dot)

Key Characteristics

- Age 20-35
- Is tech savvy
- Loves showing off his new phone to friends
- Would find a way to get out of his current contract for the latest phone
- Keeps up to date with the latest phones online
- Gets bored with phones quickly

Goals

- Have the latest, coolest phone
- Be up to date with the newest phones on the market
- Use as many features on his phone as possible



Vir: <https://www.mockplus.com/blog/post/ux-persona-examples>

TIM
THE FAMILY MAN

MARRIED MALE, 37 YEARS OLD
Atlanta, GA



PERSONALITY: BOISTEROUS, CREATIVE

"I'm a detail guy. I get cars. I spent years studying them as a teenager. However, now I look after 4 cars. It's too much."

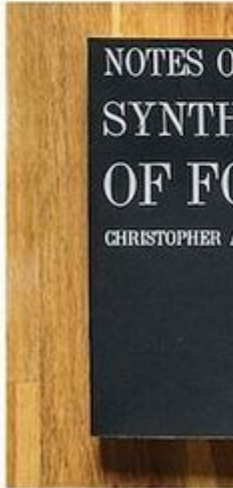
JOAN
THE SINGLE MOM

SINGLE FEMALE, 44 YEARS OLD
Chicago, IL



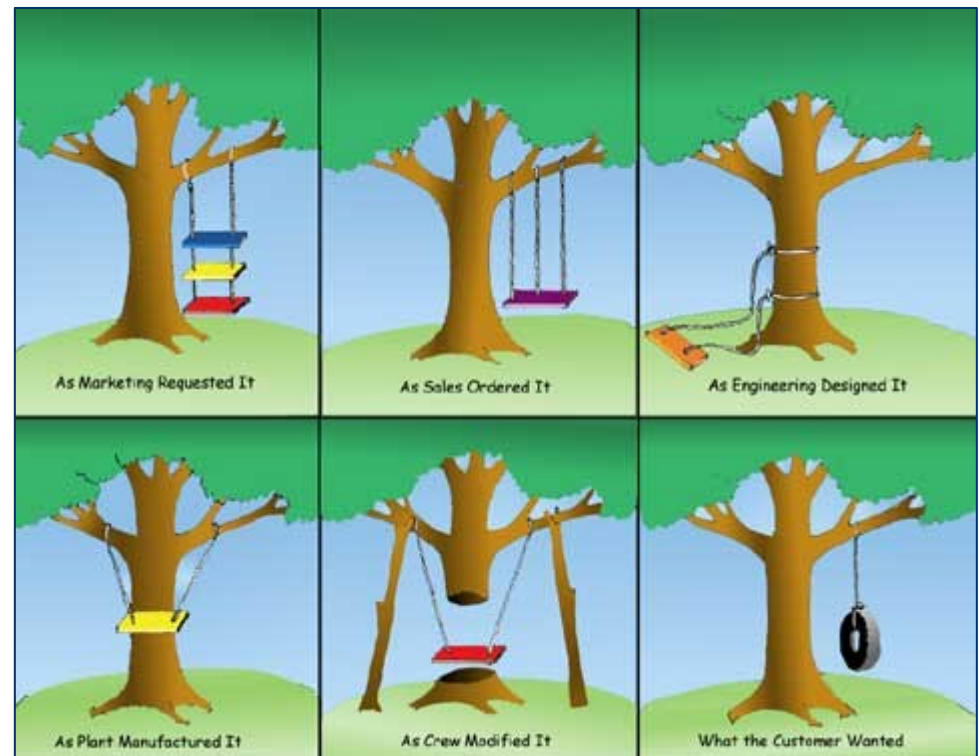
PERSONALITY: SENSE OF HUMOR, FAMILY-ORIENTED

*"m, so if I'm not working, I'm taking care
y clock, every minute counts and it's
nd."*



V službi uporabnikov (2/4)

- Spoznajmo, kaj si naše stranke dejansko želijo
 - Samo zato, ker smo sposobni razviti nekaj krasnega in lesketajočega, še ni treba nujno tega tudi storiti!



V službi uporabnikov (3/4)

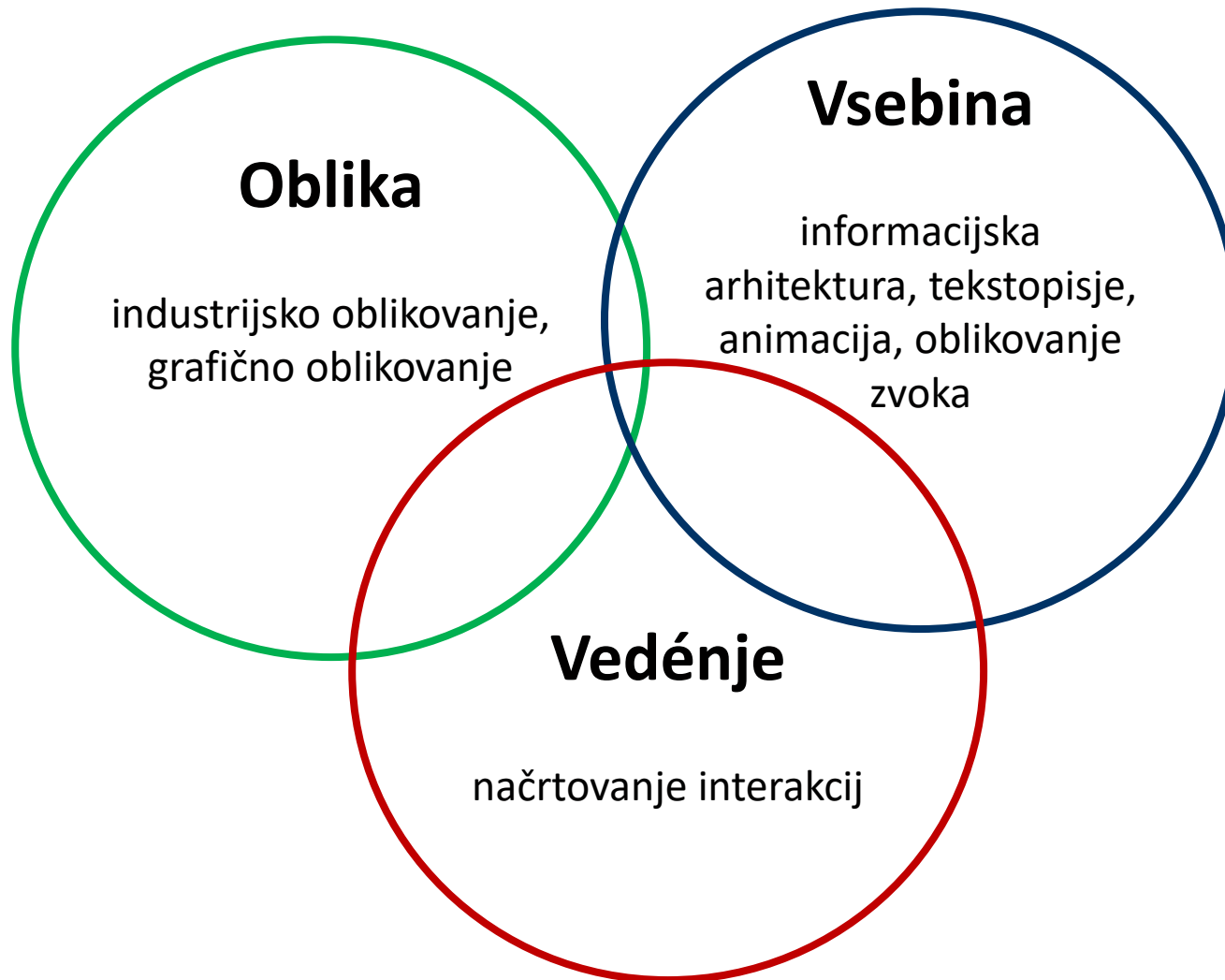
- Pomagajmo uporabnikom doseči želeno
 - Možganska nevihta (brainstorming) možnih scenarijev uporabe – kaj bi uporabniki radi dosegli?
 - Ko ustvarimo začetni načrt rešitve, ga preverimo s cilji uporabnikov
 - Posvetimo se ključnim personam; to so uporabniki, ki bodo našo rešitev največ uporabljali in s tem zelo verjetno ustvarili največ prometa – bodimo pozorni na njih!

V službi uporabnikov (4/4)

- Poenostavimo (predvsem začetne!) korake uporabe naše rešitve
 - Če bodo začetni koraki pri uporabi naše rešitve preveč zahtevni, se uporabniki enostavno ne bodo potrudili, da bi jih opravili!

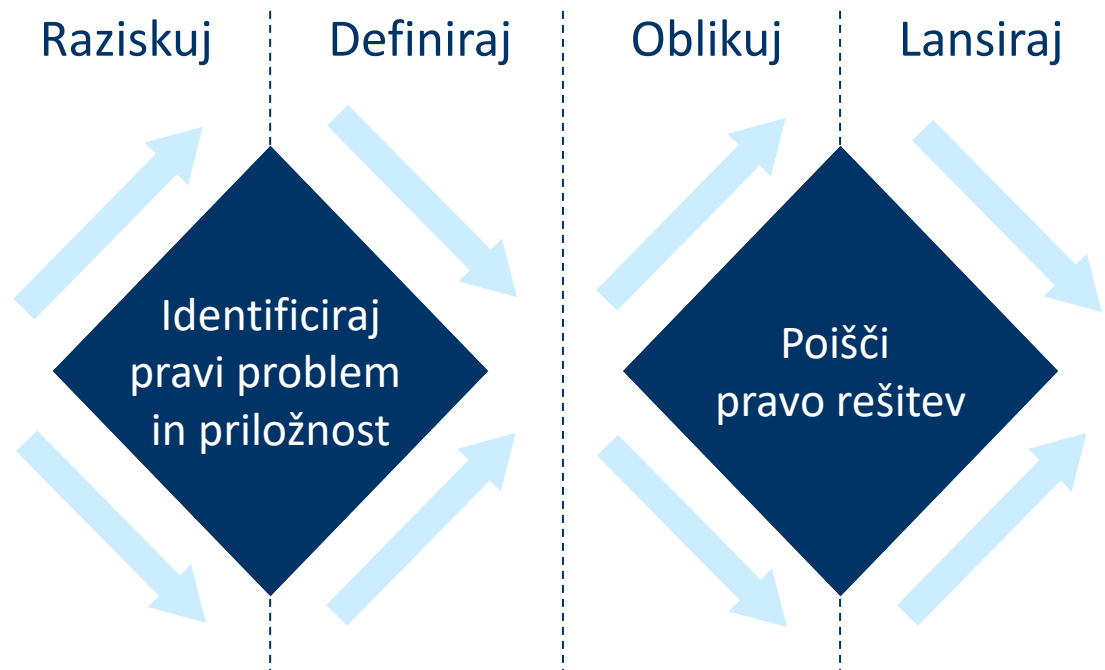


Uporabniška izkušnja



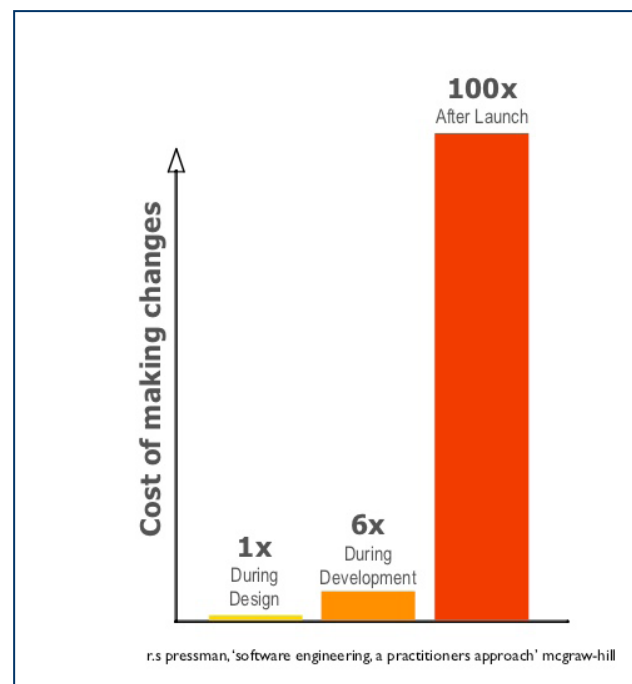
Uporabniška izkušnja

- Raziskuj / Definiraj / Oblikuj / Lansiraj
 - Iterativni proces s konstantnim vključevanjem uporabnikov v proces načrtovanja in evalvacije



Raziskuj

- Definiraj: uporabniki (primarni, sek., terc.)
 - Identificiraj: obnašanje in vzorci
 - Definirane: priložnosti, strategija, možne rešitve
-
- Raziskovanje → vpogled → razumevanje → boljša UX → zadovoljstvo!
 - Opazuj, komuniciraj, vprašaj, odkrivaj, razumi



Definiraj

- Od raziskovanja k definiciji
 - Od **ugotovitev** (težko je najti dobre kadre) do **priložnosti** (izboljšaj povezavo med študenti in podjetji) do **vprašanj** (kako bi lahko izboljšali...?) do **strategij** (predstaviti mladim poklice ter jih povezati s službami) in do **rešitev** (mobilni zaposlitveni portal)
- Oblikuj persone
- Scenariji in zgodborisi (storyboards)
- Odkrij zgodbe
- Karta izkušenj
- Funkcionalnosti
- Povezovanje med kanali/mediji
- Zahteve (uporabniške, poslovne, tehnološke)
- Vedenje in odzivanje sistema

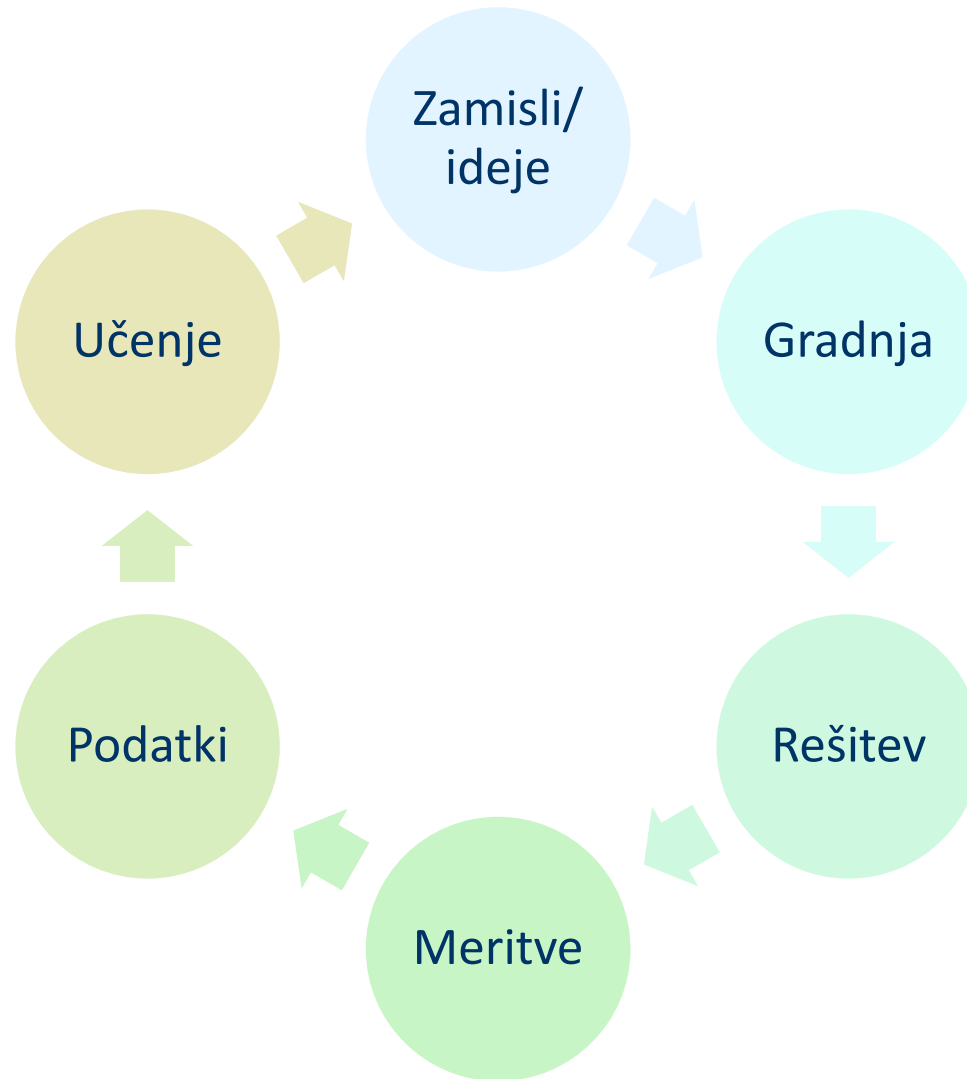
Oblikuj

- Konceptualni Lo-Fi prototipi
 - Uporabniški vmesnik (UI)
 - Vizualni jezik, stili, forme in vzorci
 - Hi-Fi prototipi
-
- Iterativni proces s konstantnim vključevanjem uporabnikov v proces načrtovanja in evaluacije!

Lansiraj

- Testiranje končnega **prototipa** z uporabniki
 - Testiranje z “naročniki”
 - Javna predstavitev prototipa
-
- Testiramo **pred**, **med** in **po** koncu načrtovanja in razvoja!

Neskončna zanka vitkih rešitev



Prototipni razvoj

Kako si pri razvoju IS pomagamo z delujočimi prototipi
(delnimi implementacijami sistema)

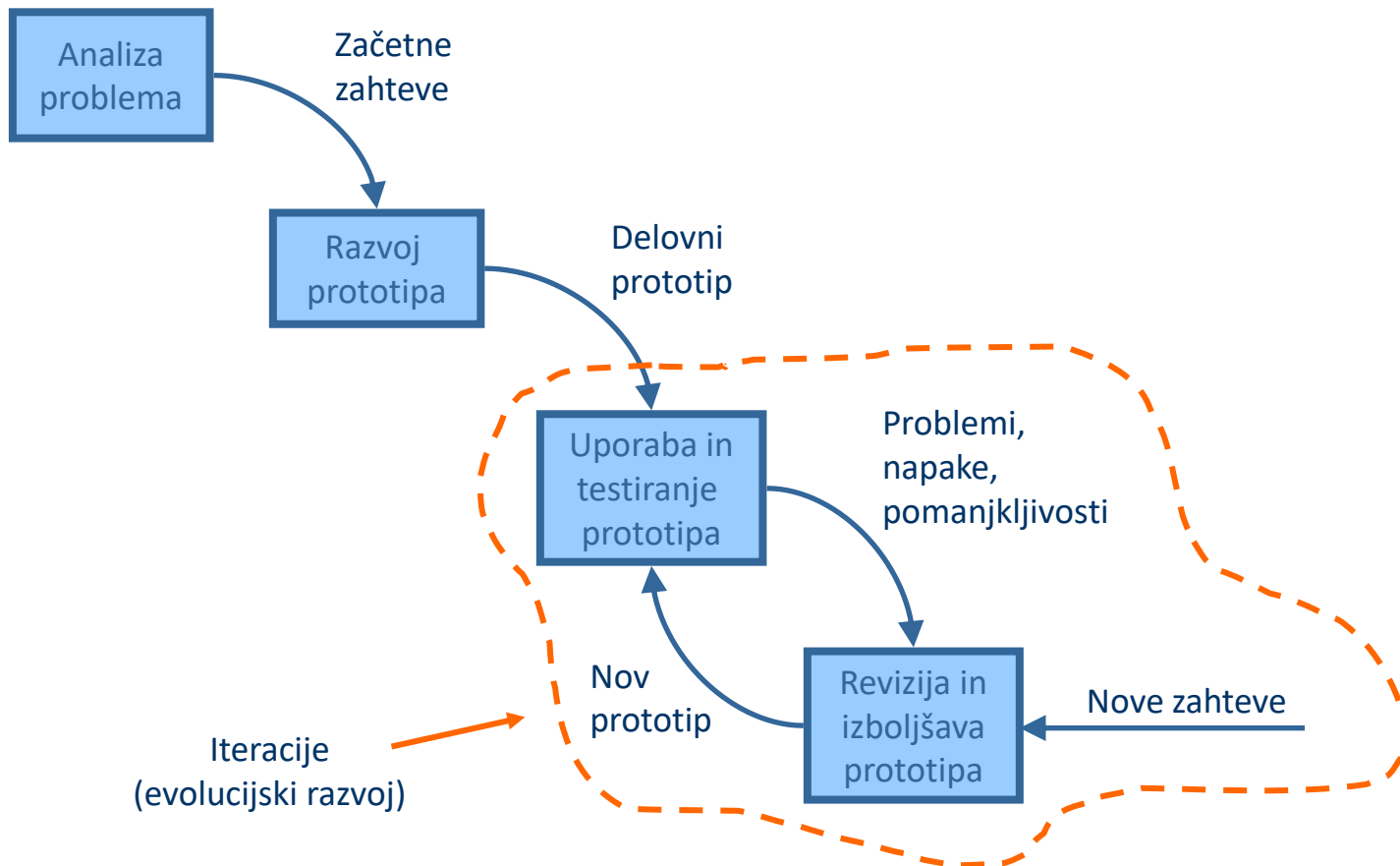
Prototipiranje

- Z agilnim prototipiranjem si predvsem pomagamo pri pridobivanju informacij za izoblikovanje uspešne ideje nove informacijske rešitve
- Prototipiranje uporabljamo še za pridobivanje drugih informacij, ki nam pomagajo zgraditi uspešen in učinkovit informacijski sistem
 - Namesto da razvijamo sistem, zgradimo model, katerega lahko hitro in enostavno prilagodimo uporabnikovim zahtevam
 - Te modele – prototipe predstavimo uporabnikom, ti jih ocenijo in v primeru, ko z njimi niso zadovoljni, lahko model hitro in enostavno modificiramo

Prototipiranje

- Prototipiranje je tehnika konstruiranja **delne implementacije** sistema, tako da se lahko uporabniki in razvijalci podrobneje seznani s problemom oz. njegovo rešitvijo
 - Temelji na izdelavi prototipov in njihovi postopni izboljšavi, dokler ne dosežemo zadovoljive kakovosti
 - Prototip označuje predhodno izdelane in navadno še nepopolne različice sistema ali dela sistema

Prototipni razvoj



Kdaj uporabiti prototipni razvoj? (1/2)

- Uporabniki ne vedo natančno, kaj želijo
- Uporabniki s težavo podajajo oz. izražajo svoje zahteve
- Sistem spremeni osnove poslovnih operacij
- Uporabniške vmesnike je potrebno prilagoditi končnim uporabnikom
- Funkcije, ki naj bi jih računalniško podprli, so zapletene in nejasne ter jih uporabniki poznajo oz. razumejo bolje kot analitik
- Zaslonske oblike in poročila je potrebno preveriti, da ugotovimo možne izboljšave ter zvišamo nivo uporabnosti

Kdaj uporabiti prototipni razvoj? (2/2)

- Če obstaja področje uporabnikovega dela, ki ga je možno izboljšati, je potrebno zagotoviti kreativno sodelovanje uporabnikov
- Uporabniki ne razumejo in opazijo vseh pridobitev in možnosti, ki jih nudi nastajajoči sistem
- Potrebno je raziskati relativno vrednost alternativnih rešitev
- Vpeljati želimo sistem, ki spremeni naravo dela uporabnikov, zato želimo ugotoviti njihov odziv

Vrste delujočih prototipov

□ Raziskovalni prototipi

- Ugotovimo, kaj uporabniki želijo od sistema, torej kaj naj bi grajeni IS zagotavljali; takšne prototipe nato zavržemo

□ Eksperimentalni prototipi

- Pripomoček za ugotavljanje izvedljivosti (realizacije); jih zavržemo

□ Razvojni prototipi

- Razvijamo postopoma in z namenom, da v določenem času postanejo sam sistem

Raziskovalni prototipi

- Raziskovalni prototip predstavimo potencialnim uporabnikom ali naročnikom z namenom, da skupaj z njimi:
 - preučimo **izvedljivost** želja (zahtev),
 - potrdimo potrebnost oz. **nujnost** posameznih funkcij,
 - odkrijemo **manjkajoče**, torej nepodane zahteve, in
 - raziščemo možnosti razvoja ustreznega uporabniškega vmesnika
- Prototipa ne gradimo, če lahko identificiramo vse zahteve in se potencialni uporabniki z njimi strinjajo

Eksperimentalni prototipi

- Hitre in robustne prototipe lahko gradimo v fazi načrtovanja (tako preliminarne kot podrobnega)
- Z njimi lahko **validiramo strukturo** programskega sistema, torej ugotovimo, ali zasnovana programska struktura zadovoljuje vse identificirane zahteve, oz. bo lahko zahteve zadovoljila
- Prav tako lahko **preverimo, ali je neka** izbrana **tehnologija**, orodje ali pristop **primerna** za razvoj končnega sistema
- Tudi pri načrtovanju velja podobno kot pri analizi → prototipa ne gradimo, če smo lahko dovolj učinkoviti tudi brez njega oz. obstaja splošno soglasje glede ustrezne alternative

Razvojna metoda

- Pri tej metodi **prototip razvijemo v končni sistem**
- Prototip mora imeti kvalitete končnega produkta, kot so zanesljivost ter enostavno vzdrževanje
- Vse to povzroča počasnejši razvoj prototipa v primerjavi z metodo zavračanja (ko razvijamo prototip le zato, da pridobimo informacije)
- Če se odločimo za ta pristop, je naravna pot seveda ta, da sledimo običajnemu življenjskemu ciklu razvoja
- Proces ponavljamo tako dolgo, dokler ne dosežemo zadovoljive stopnje prototipa, katerega razvijemo v končni sistem
 - Tak način nam zagotavlja razvoj potrebne dokumentacije in potreben pregled nad sistemom

Sistem delnega prototipiranja

- Pogovorni prototip – dialog končnega sistema z uporabnikom
- Vnos podatkov – ugotoviti hitrost in točnost vnašanja podatkov
- Sistem sporočanja – preveriti sporočila in način njihovega podajanja
- Podatkovni sistem – prototip podatkovne baze lahko predstavimo kot nekaj zapisov
- Računanje in logika – ko so izračuni ali logika programskega sistema zelo zapleteni
- Integracija aplikacij – preverimo uporabnost, da se izognemo slabemu povezovanju z drugimi aplikacijami
- Koncept – pri velikih sistemih; včasih se pojavi vprašanje koncepta sistema

Orodja za prototipiranje (1/2)

- Orodje naj bi:
 - bilo interaktivno
 - bilo enostavno za uporabo
 - omogočalo enostavno in hitro spreminjanje
 - omogočalo hitro gradnjo prototipov (tudi delnih)
 - spodbujalo postopno dograjevanje
 - podpiralo ustrezne strukture podatkovnih baz
 - vsebovalo vnaprej pripravljene komponente za splošne oz. pogosto uporabljane funkcije

Orodja za prototipiranje (2/2)

- Orodje naj bi vključevalo:
 - zmogljiv zaslonski oblikovalnik
 - možnost povezovanja zaslonov kot odgovora na dialog
 - zmogljiv generator poročil
 - generator kode
 - primeren sistem za upravljanje podatkovne baze
 - integrirani podatkovni slovar
 - možnost ekstrakcije podatkov iz obstoječih zbirk oz. podatkovnih baz
 - prenos podatkov v prototipno bazo podatkov (ali on-line dostop)

Potek prototipiranja

- Določanje področja in cilja
 - Določitev, kdo bo testiral prototip
 - Določitev, kdo bo zgradil prototip
 - Izbira orodja za prototipiranje
 - Skrb za podatkovno bazo
 - Kreiranje začetnega prototipa
 - Razvijanje prototipa s končnimi uporabniki
 - Dograjevanje ali dodajanje prototipu
 - Testiranje prototipa z uporabniki
 - Vodenje pogovorov z uporabniki ali delavnice
 - Določitev, katere lastnosti spremeniti oz. dodati
 - Oblikovanje možnih in koristnih izboljšav
 - Določitev roka za naslednjo verzijo
 - Določitev korakov, ki sledijo prototipiranju
 - Zagotovitev uporabnikovih zahtev
- ponavljamo za
vsako verzijo
prototipa

Razlike med prototipom in končnim sistemom

- V prototipu ni nujno zagotoviti nekaterih lastnosti, ki jih mora imeti končni sistem:
 - sposobnost okrevanja po napaki
 - varovanje podatkov
 - sposobnost revizije
 - enostavnost vzdrževanja
 - večuporabniško delovanje
 - ustrezni odzivni časi
 - obvladovanje obsežnih baz podatkov
 - delovanje v mreži
 - delovanje na različni strojni opremi
 - dokumentacija

Prednosti prototipiranja

- Uporabniki vidijo, kaj za njih gradimo ter podajo kritične pripombe
- Manjše tveganje, da zgradimo neustrezen sistem
- Spodbujanje uporabnikov h konstruktivnemu sodelovanju
- Za uporabnika lažje razumljivo kot specifikacije na papirju
- Hitreje izdelamo prototip kot specifikacije
- Izločimo pristranskost, saj nakažemo pridobitve novega načina dela
- Napake lahko odkrijemo pred cenovno zahtevnimi fazami
- Navdušenost in večja motiviranost tako uporabnikov kot razvijalcev
- Preizkusimo lahko več inačic in se odločimo za najboljšo

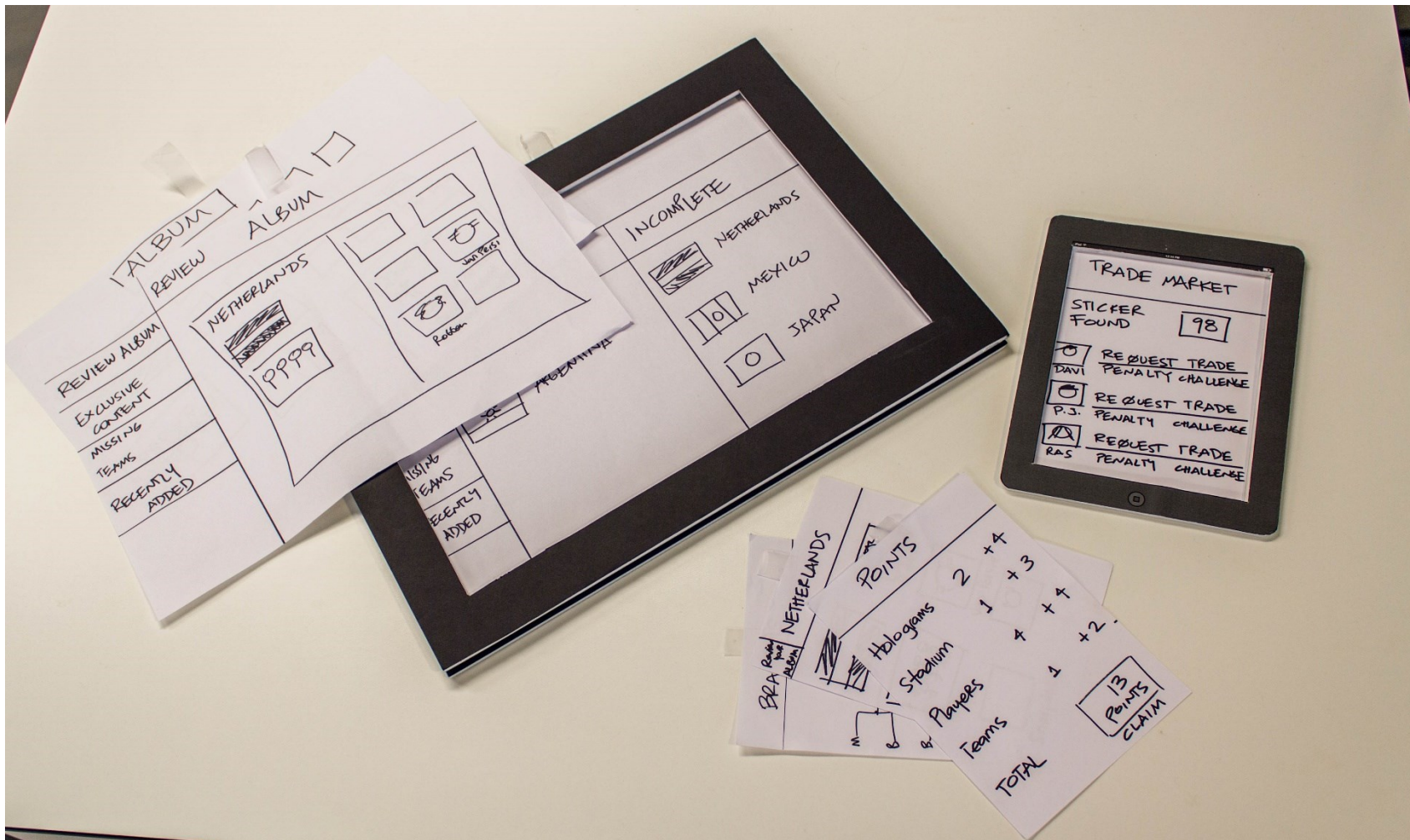
Slabosti prototipiranja

- ❑ Površna analiza in načrtovanje
 - ❑ Pogosto se zgodi, da iz prototipa nastane površno razvit končni izdelek
- ❑ Previsoke zahteve uporabnikov
 - ❑ Ko uporabniki vidijo prototip, si želijo vse več dodatnih funkcionalnosti
- ❑ Problem razlikovanja prototipa in sistema
 - ❑ Problem je prisoten tako pri uporabnikih kot tudi pri razvijalcih
- ❑ Površnost uporabnikov pri pregledu
 - ❑ Ker je razvoj iterativen, so uporabniki pri pregledih izboljšanih različic površni, ker se jim zdi, da so to že enkrat pregledali

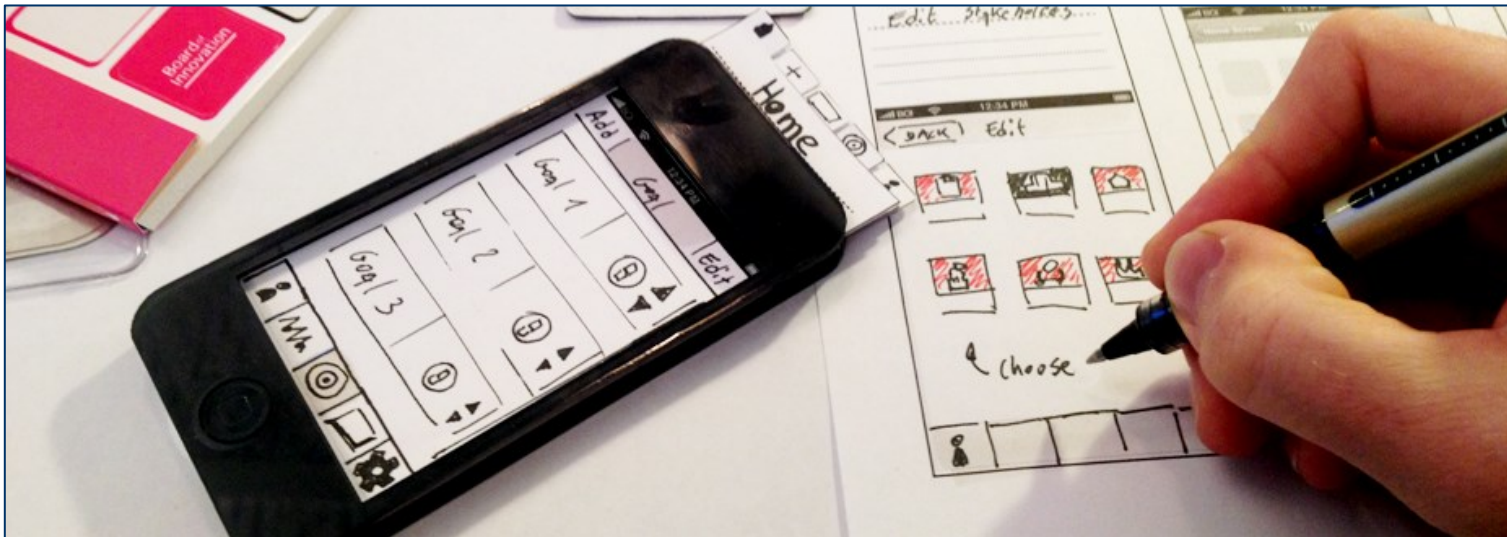
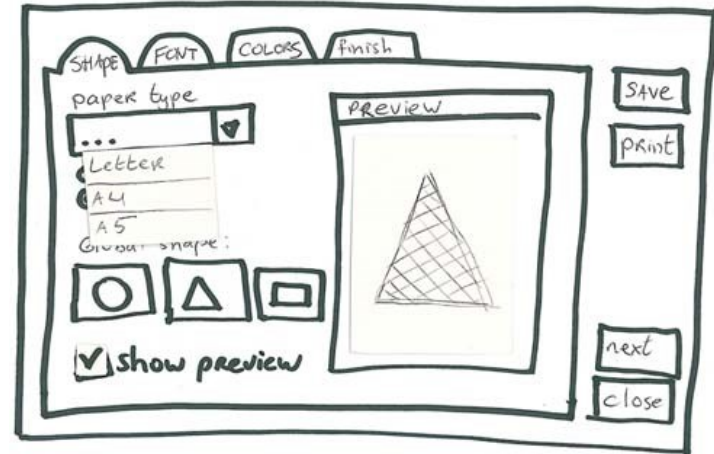
Prototipiranje na papirju

Najbolj "enostavno" je lahko pogosto najboljšo

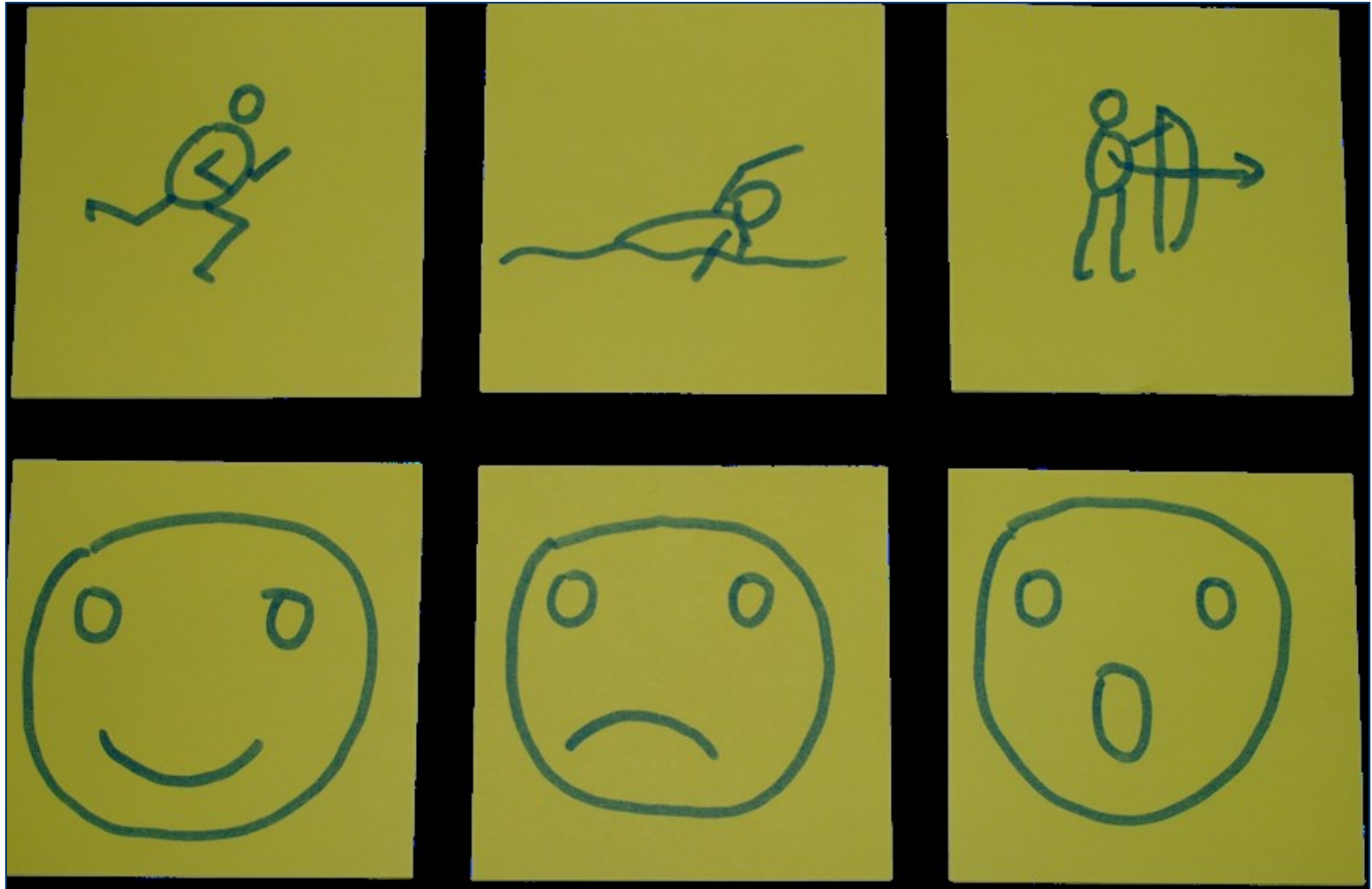
Zakaj na papirju?



Je enostavno!



Pomembno je le razumevanje!



Delo v skupini



Prednosti papirnega prototipiranja

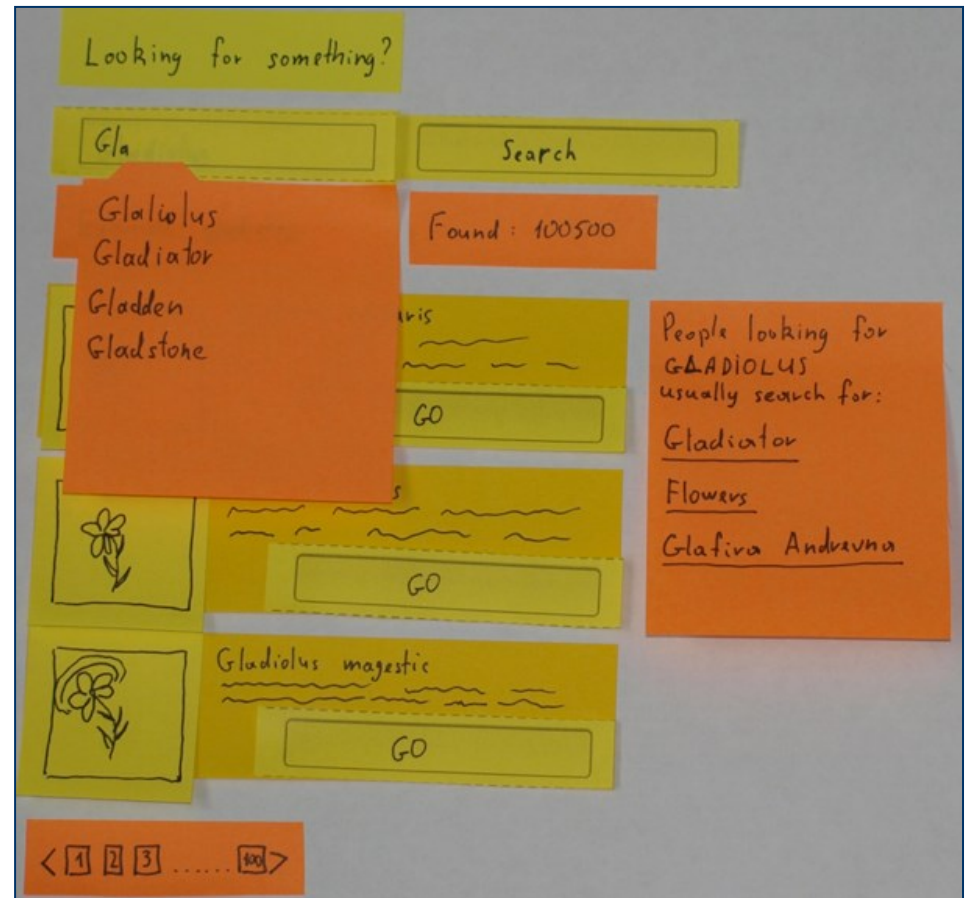
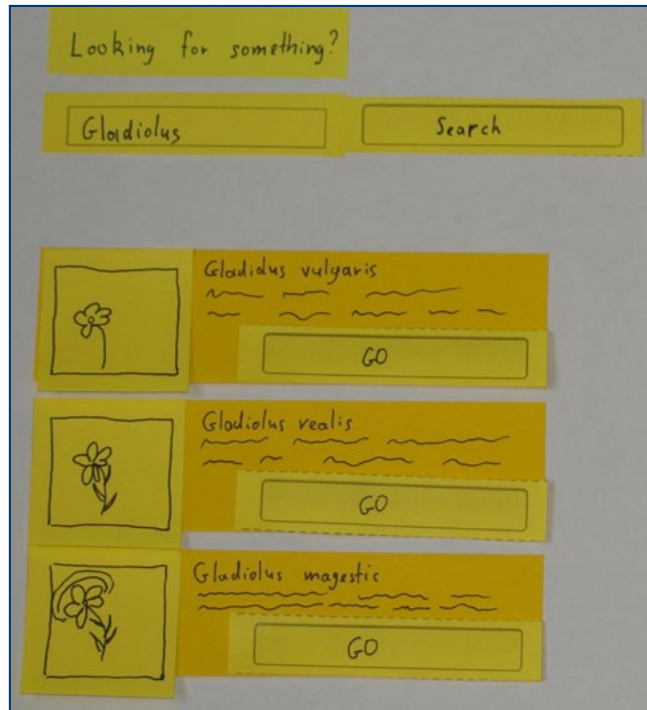
- Papirno prototipiranje je običajno najhitrejše
- Pomaga sprejemati odločitve
- Neodvisno od tehničnega znanja
- Potrebujemo:
 - Papir, transparentno folijo, samolepilne listke, škarje, lepilni trak, pisala, markerje, šablone, ...



Pripomočki



Primer 1



Primer 2

