



UPORABNIŠKO USMERJENO NAČRTOVANJE

- Zgodnji prototipi lahko odkrijejo probleme uporabnosti
- Slapovni model razvoja programske opreme
- Iterativno načrtovanje
- Iterativno načrtovanje, napačni pristop
- Spiralni model
- Iterativno načrtovanje uporabniških vmesnikov s spiralnim modelom
- Uporabniško usmerjeno načrtovanje
- Načrtuj
- Implementiraj
- Natančnost prototipov je večdimenzionalna
- Vrednoti
- Iterativno načrtovanje uporabniških vmesnikov in navodila
- Dodatni materiali

Zgodnji prototipi lahko odkrijejo probleme uporabnosti



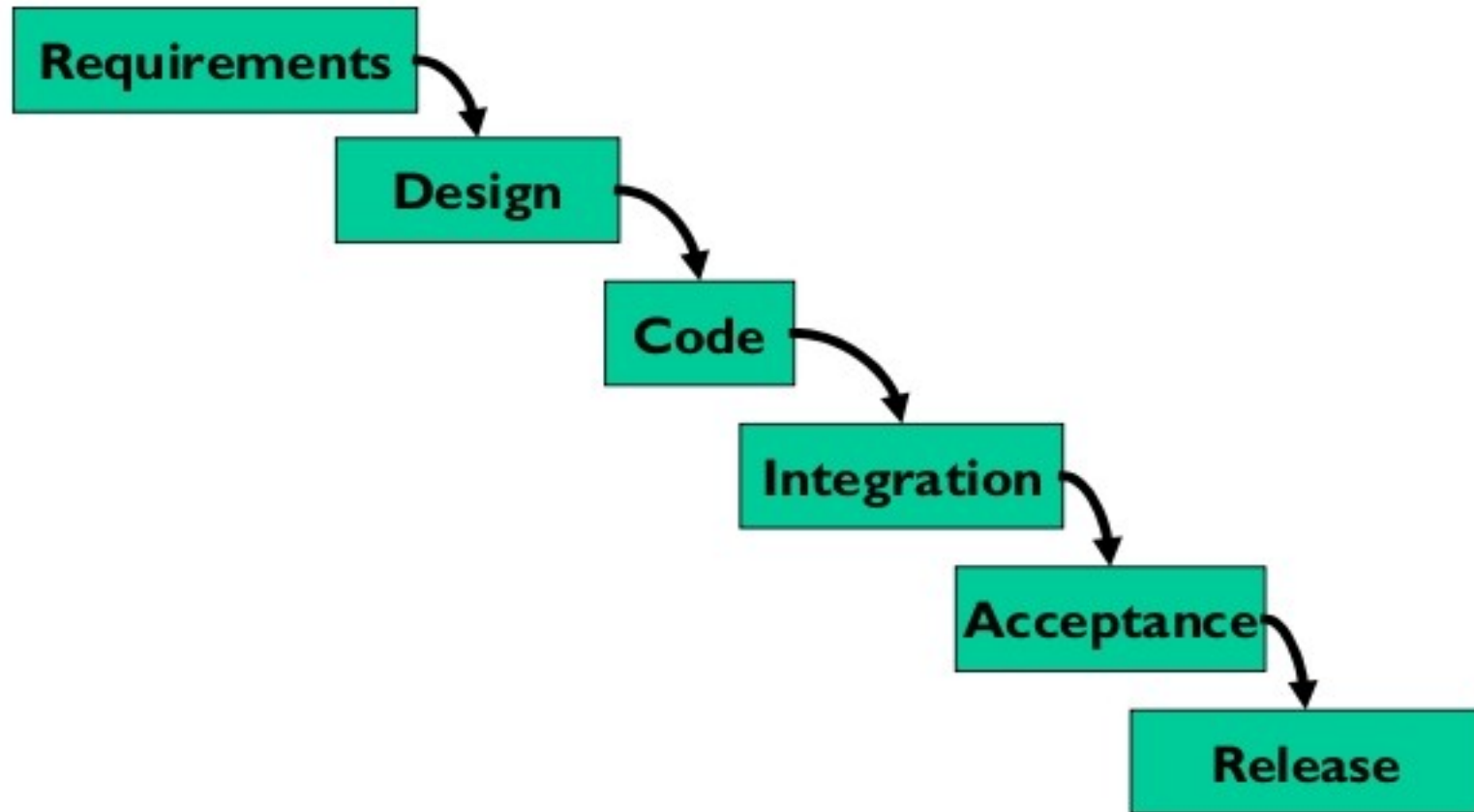
- **Uporabnosti ni**
- **Konsistentnost:** drsnik – predhodne izkušnje, druge aplikacije, ga sploh uporabiti?, kako najdem vzorec drugič?, pregled vsega
- **Pomagljivost:** drsnik - ponuja zvezno drsenje ne pa diskretno izbiro → zato pomoč
- **Navodila načrtovanja niso upoštevana:**
 - tekst ("patch" - programski popravek, popravek za dosego uporabosti na koncu načrtovanja):
 - OKAY – kje je?,
 - Poravnavanje teksta!
 - Kako tiskam?
- **Bližnjic ni.**

- Problem v dialogu bi bil odkrit s papirnatisim (preprostim) prototipom

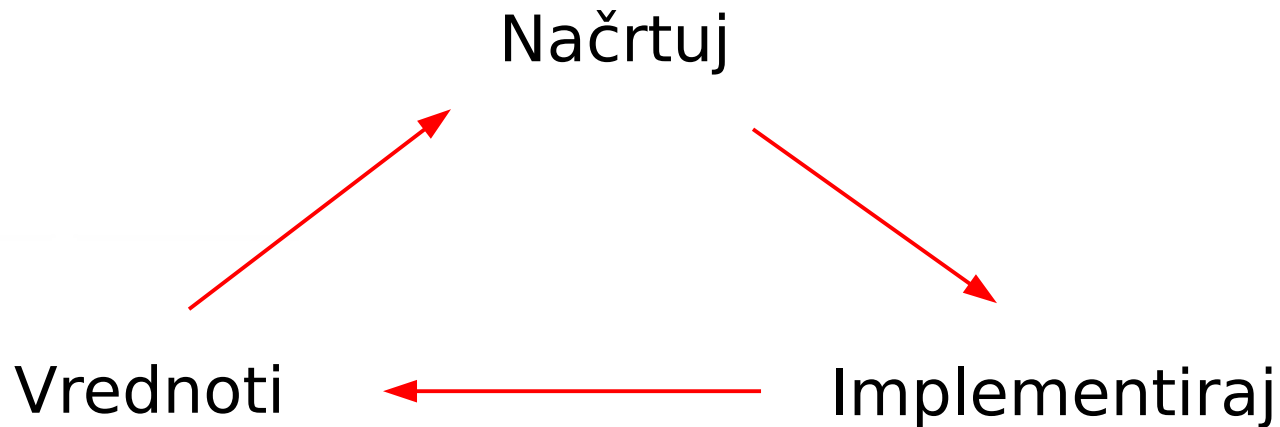
(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Slapovni model razvoja programske opreme

- Najprej misli, nato kodiraj



Iterativno načrtovanje



- Zmanjšuje vsebovano tveganje in omogoča njegovo odpravo pri načrtovanju UV
- Bo ena iteracija s končnim testiranjem dovolj?
- **Ne. Ponavljaj!**

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

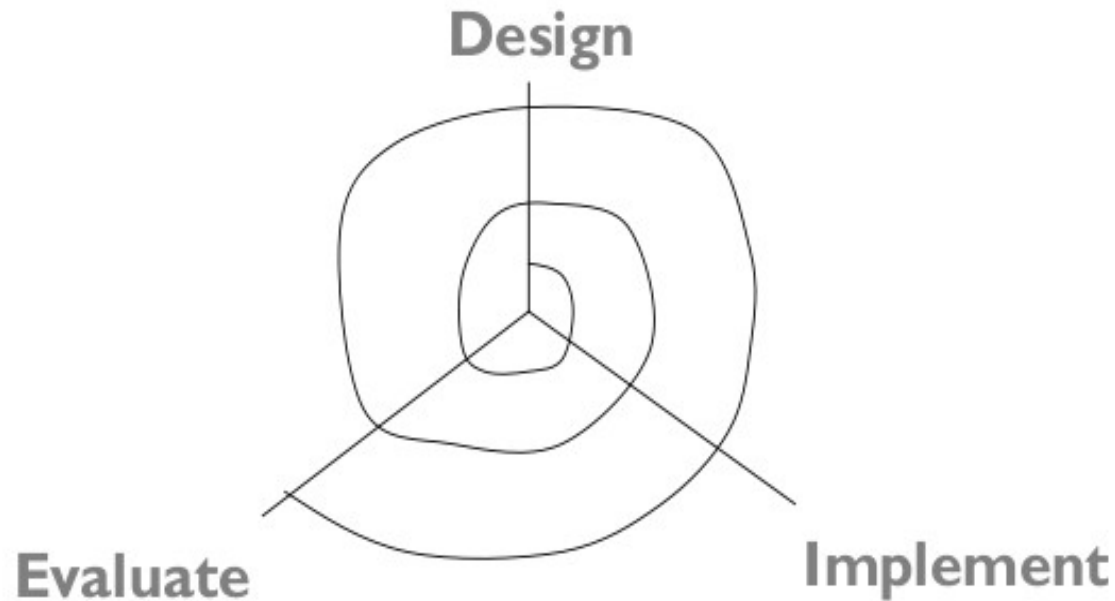
Iterativno načrtovanje, napačni pristop

- **Vsaka iteracija vključuje tudi izdajo programske opreme**
 - Vrednotenje (pritožbe) vodi v načrtovanje naslednje verzije
- **Končni uporabniki so tisti, ki vrednotijo uporabnost**
 - Ne bodo kupili naslednje verzije

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Spiralni model

Načrtuj, Implementiraj, Vrednoti, ...



- **Več iteracij** v procesu načrtovanja
- **Radialna dimenzija** spiralnega modela **ustreza ceni**, zgodnje iteracije naj bodo poceni

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Iterativno načrtovanje uporabniških vmesnikov s spiralnim modelom

- Uporabljaljaj *poceni* prototipe v zgodnjih iteracijah
 - **Vzporedno načrtovanje** je izvedljivo: zgradi in testiraj več prototipov z namenom preizkusa alternativnih načrtovanj, lažje spremeniti ali zavreči
- Kasnejše iteracije so lahko dražje
 - ni več nevarnosti za bistvene pomanjkljivosti UV

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Uporabniško usmerjeno načrtovanje

- **Iterativno načrtovanje s spiralnim modelom**
- **Najprej se osredotoči na uporabnike in naloge**
 - Analiza uporabnikov in njihovih nalog
 - Vključevanje uporabnikov za vrednotenje, konzultacije in včasih kot načrtovalce
- **Stalno vrednoti**
 - Uporabniki naj so vključeni v vsako iteracijo

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Načrtuj

- **Upoštevaj principe in navodila načrtovanja**
 - So heuristike, ne trda pravila
 - Lahko so nejasna ali kontradiktorna
(uporabnik naj ima poln nadzor ↔ izogibaj se napakam)

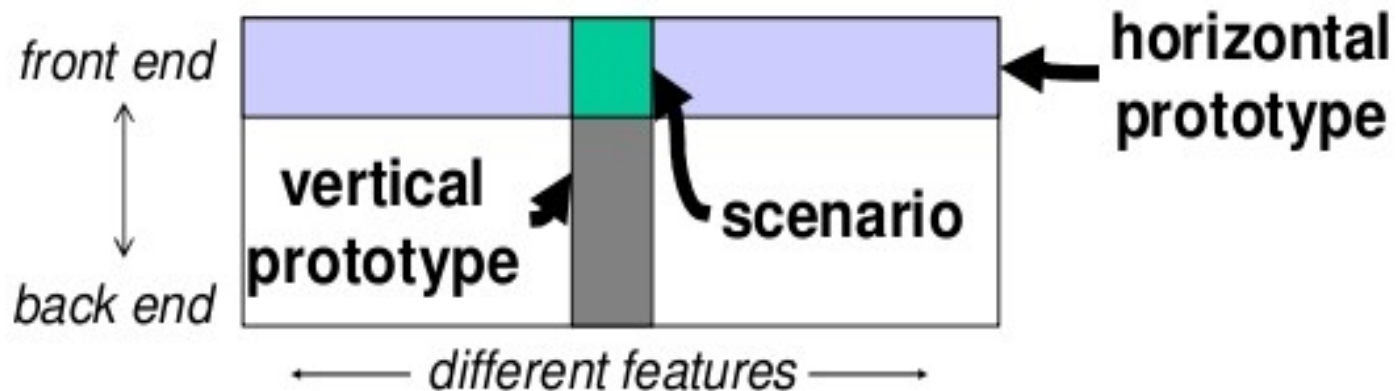
(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Implementiraj

- **Uporabljaljaj prototipe**
 - *Preprosti, (poceni), papirnati, (skice), prototipi*
 - *Računalniški prototipi*
- **Kaj je natančnost prototipa?**
 - Nizka natančnost (ni detajlov, cenejše, drugačne tehnike interakcije)
 - Visoka natančnost (zelo podobno končnemu izdelku)
- **Tehnike implementacije UV**
 - Uporabljaljaj razvojna orodja
 - Uporabljaljaj generatorje vmesnikov

Natančnost prototipov je večdimenzionalna

- **Natančnost v smislu look & feel:** izgled, občutek pri delu
- **Natančnost po širini:** odstotek realiziranih funkcionalnosti (npr., število opcij vrstičnega menuja)
- **Natančnost po globini:** stopnja, globina, implementacije realiziranih funkcionalnosti
 - Omejeno število izbir (npr: dvostranskega tiskanja ni; omejeni, isti, odzivi)



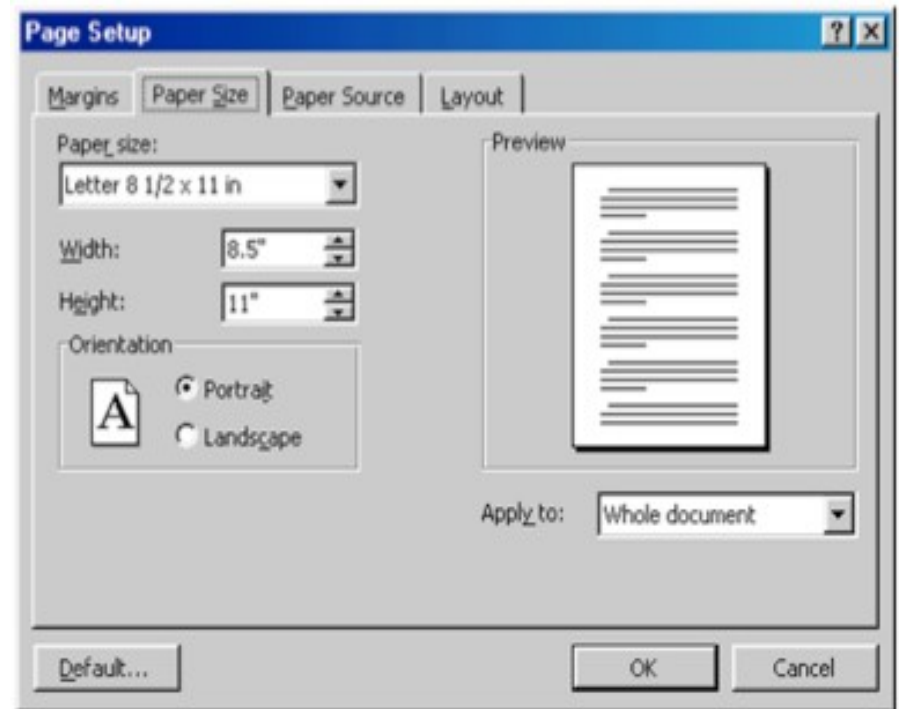
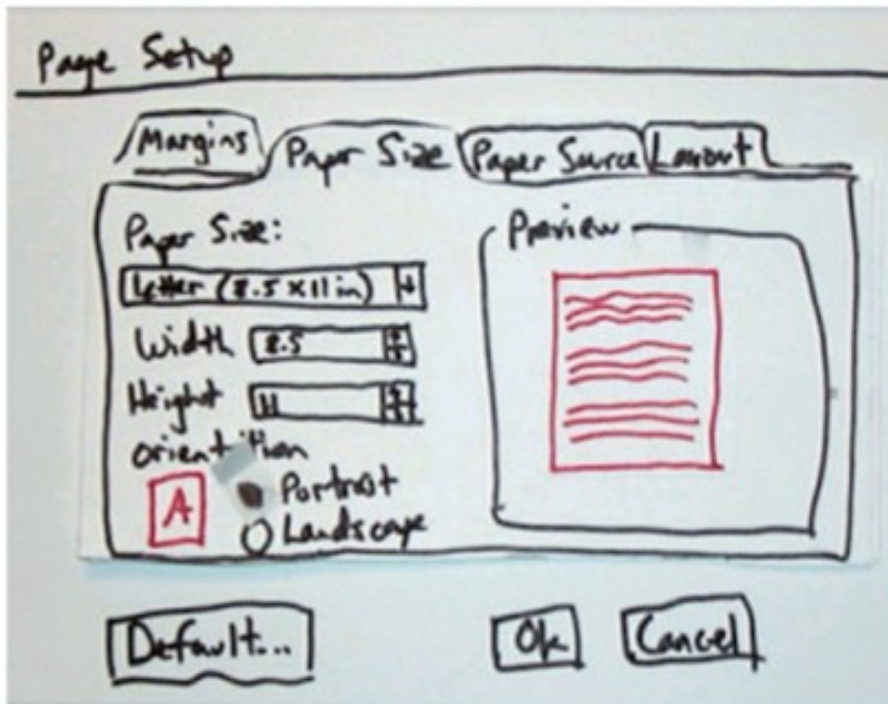
(Andrews, Human-Computer Interaction, 706.021 Mensch-Machine Kommunikation 3VU SS, Graz University of Technology)

(Nielsen, Usability Engineering, p. 94)

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Preprosti prototipi, skice

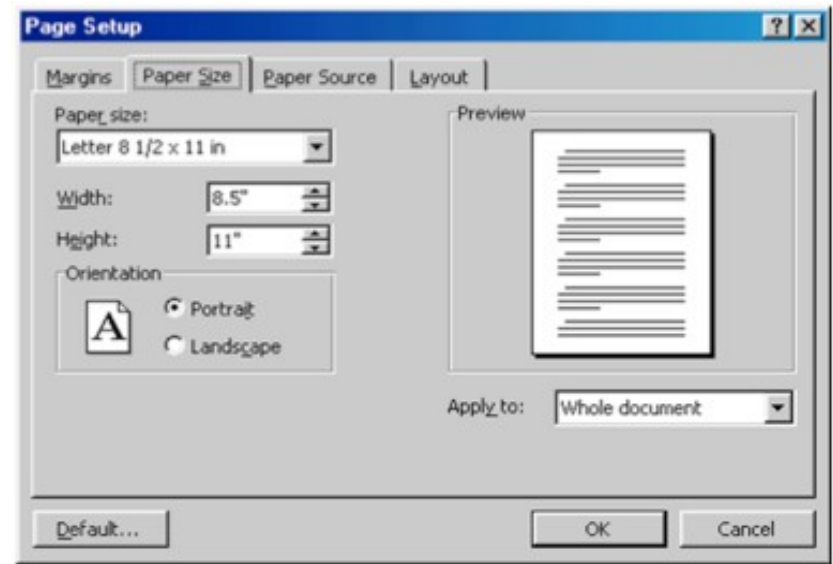
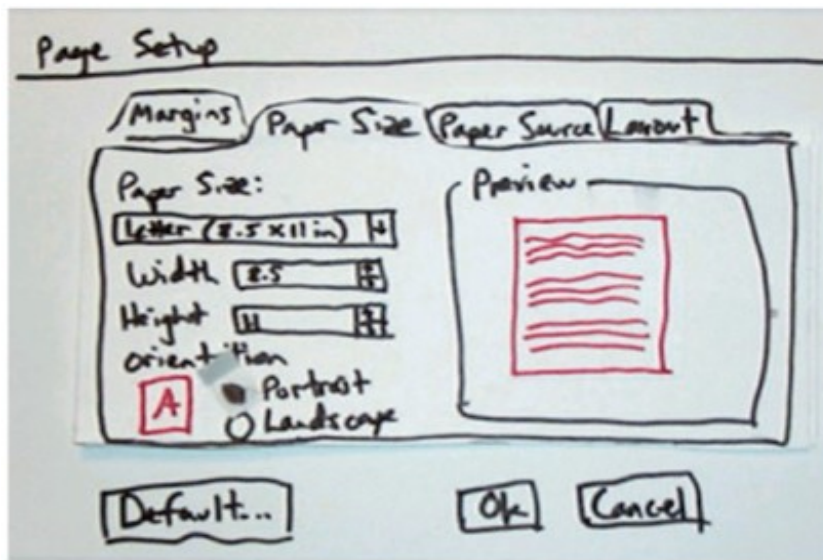
- **Nizka natančnost v smislu *look & feel***
 - *Look*: izgled prototipa, grafično načrtovanje, z roko narisano
 - *Feel*: kazenje in pisanje je drugačno po občutku od dela z miško in tipkovnico
- **Visoka natančnost po globini (človek simulira (opisuje) ozadje aplikacije)**



(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Računalniški prototipi

- Interaktivna simulacija programske opreme
- **Visoka natančnost v smislu *look & feel***
- **Nizka natančnost po globini**
 - Pri papirnatem prototipu človek simulira (opisuje) ozadje, pri računalniškem prototipu ne
 - Računalniški prototip je tipično horizontalen: pokriva večino značilnosti, ne pa nujno tudi ozadja



(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Vrednoti

- **Hevristično (ekspertno) vrednotenje**
 - Hevristike in poizkusi (preglej vmesnik izčrpno, primerjaj vmesnik s principi in navodili načrtovanja, odkrij probleme uporabnosti)
- **Testiranje uporabnikov, vrednotenje uporabnosti (empirična evaluacija)**
 - Opazovanje in evaluacija uporabnikov v smislu metrik uporabnosti (naučljivost, učinkovitost, varnost, zadovoljstvo)

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Iterativno načrtovanje uporabniških vmesnikov in navodila

N – Načrtuj, I – Implementiraj, V – Vrednoti

N: Razumevanje uporabnika in domene

N: Razumevanje nalog, dela, akcij, konceptov in željene aplikacije

N: Izbor naprav za interakcijo, vhodne naprave, izhodne naprave

N: Načrtovanje oken

N: Načrtovanje menujev

N: Izbor grafičnih gradnikov za interakcijo

N: Aranžiranje grafičnih gradnikov za interakcijo

N: Grafično načrtovanje (izbor teksta, barv, slik in animacije)

N: Izbor in načrtovanje ikon

N: Načrtovanje povratne informacije in interakcij (obvestila, zvok, dialogi)

I: Papirnati prototipi

I: Računalniški prototipi (orodja za načrtovanje: JavaSwing, GTK+, ...)

V: Hevristično vrednotenje

V: Testiranje uporabnikov (vrednotenje uporabnosti)

Dodatni materiali

- Implementiraj (zakaj papirnati (enostavni) prototipi?)
- Papirnati prototipi
- Papirnati prototipi (primeri)
- Kaj se lahko naučimo iz papirnega prototipa?
- Računalniški prototipi
- Kaj se lahko naučimo iz računalniškega prototipa?
- Tehnike računalniških prototipov
- Orodja za delo s sekvencami slik
- Generatorji vmesnikov
- Prednosti in slabosti generatorjev vmesnikov
- Kaj se lahko naučimo iz prototipov?

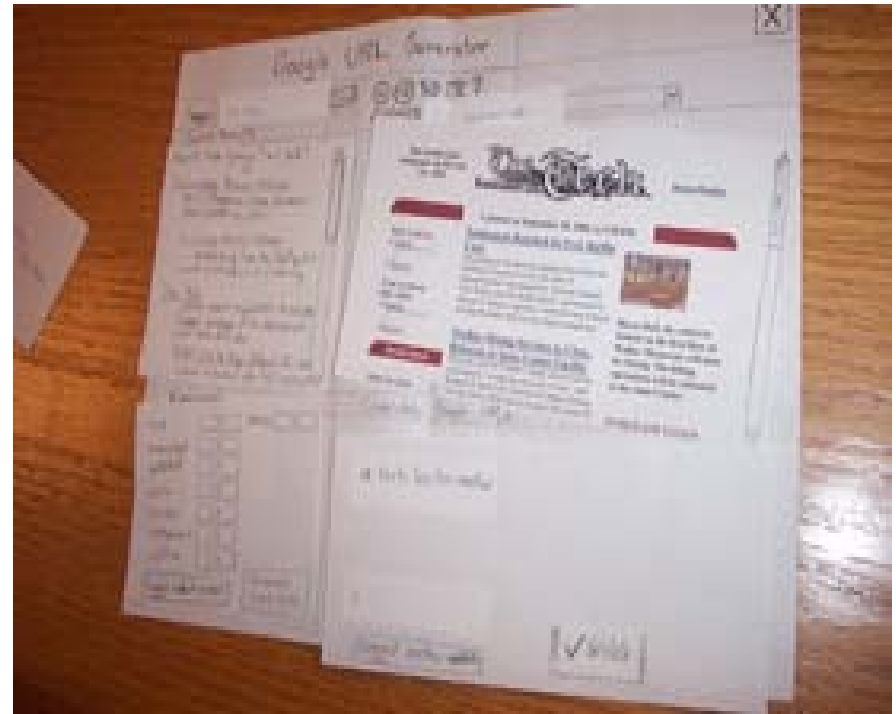
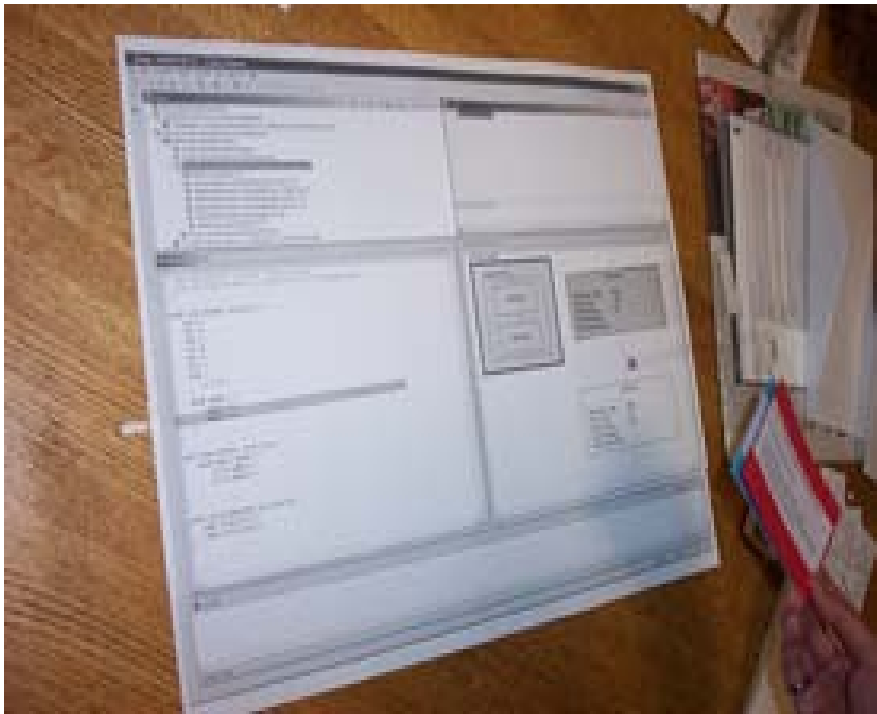
Implementiraj (zakaj papirnati (enostavni) prototipi?)

- **Hitrejši razvoj**
 - Skiciranje je hitrejšo od programiranja
- **Lažje spreminjati**
 - Ni investicije v kodiranje, ostane načrt drugo lahko odvržemo
 - Lažje spremembe med iteracijami, ali celo med samim testiranjem uporabnikov
- **Pozornost je usmerjena na “veliko sliko”**
 - Načrtovalci ne izgubljajo časa z drobnimi detajli
 - Uporabniki imajo bolj kreativne predloge
- **Neprogramerji so lahko v pomoč**
 - Predznanje ni potrebno

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

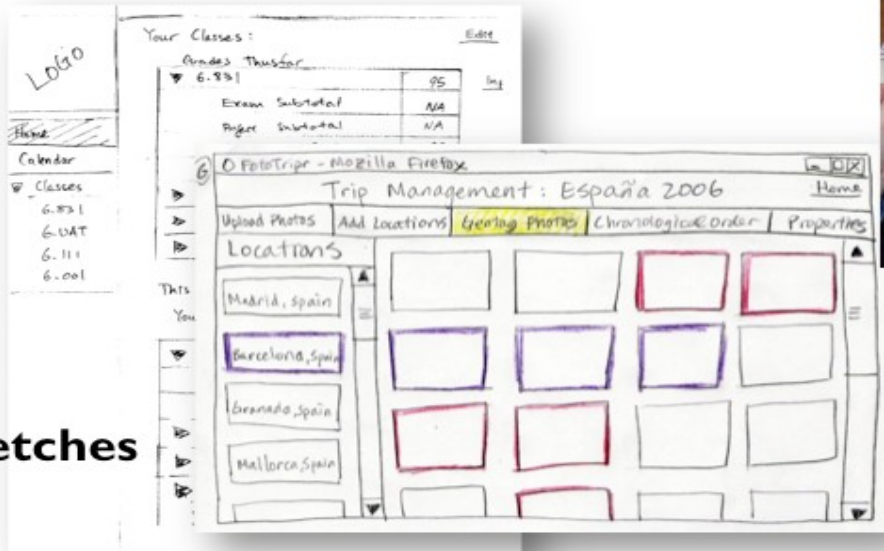
Papirnati prototipi (primera)

- IBM Eclipse, web browser

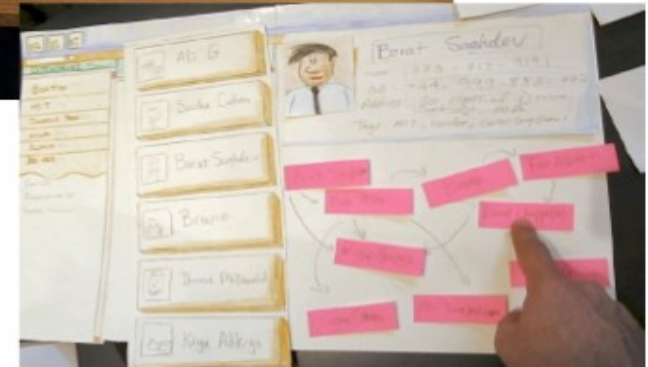


(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Papirnati prototipi (primeri)

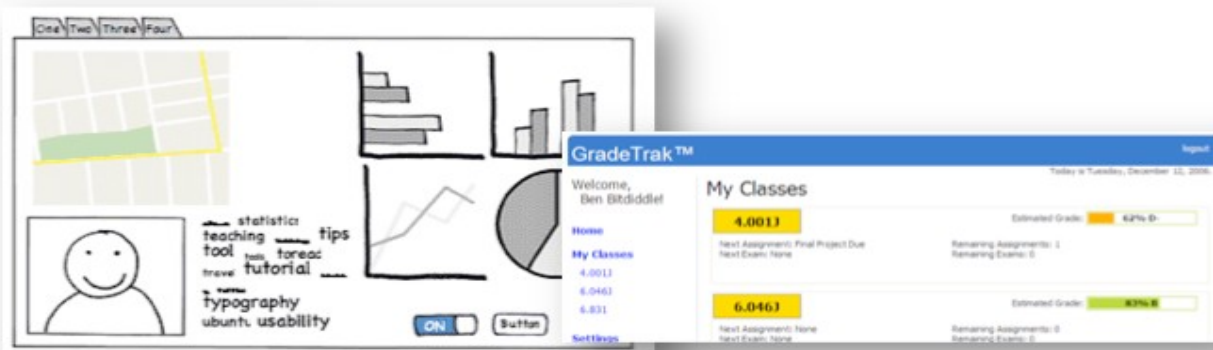


sketches



paper prototypes

computer mockups



(Miller, Course 6.831 UI Design and Implementation, Spring 2018, MIT EECS)

Kaj se lahko naučimo iz papirnatega prototipa?

- **Konceptualni model**
 - Ali ga uporabnik razume?
- **Funkcionalnosti**
 - Vmesnik naredi kar je potrebno? Manjkajo značilnosti?
- **Tok navigacije in nalog**
 - Se uporabniki znajdejo?
 - Je dovolj informacije za uporabnika za uspešno delo?
- **Terminologije**
 - Uporabniki razumejo oznake?
- **Vsebine za zaslon**
 - Kaj mora biti prikazano?

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Kaj se lahko naučimo iz računalniškega prototipa?

- Vse tisto kar iz papirnatega prototipa in:
 - **Izgleda na zaslonu** (Ali je vmesnik čist, prevelik, kompliciran, zmeden? Lahko uporabnik najde pomembne elemente?)
 - **Barv, pisav (font), ikon, drugih elementov** (Ali so dobro izbrani?)
 - **Vprašanj učinkovitosti** (Implikacij Fitts-ovega zakona. So osnovni gradniki dovolj veliki? So preblizu skupaj? So sezname pod drsniki predolgi?)
 - **Odzivnih časov**
 - **Dinamike** (uporabniki *manj* razmišljajo in *več* preiskujejo)
 - **Interaktivne povratne informacije** (Ali uporabnik opazi in se odzove na obvestila o spremembi statusa, spremembah kurzorja? ... Majhne spremembe – slepota sprememb)

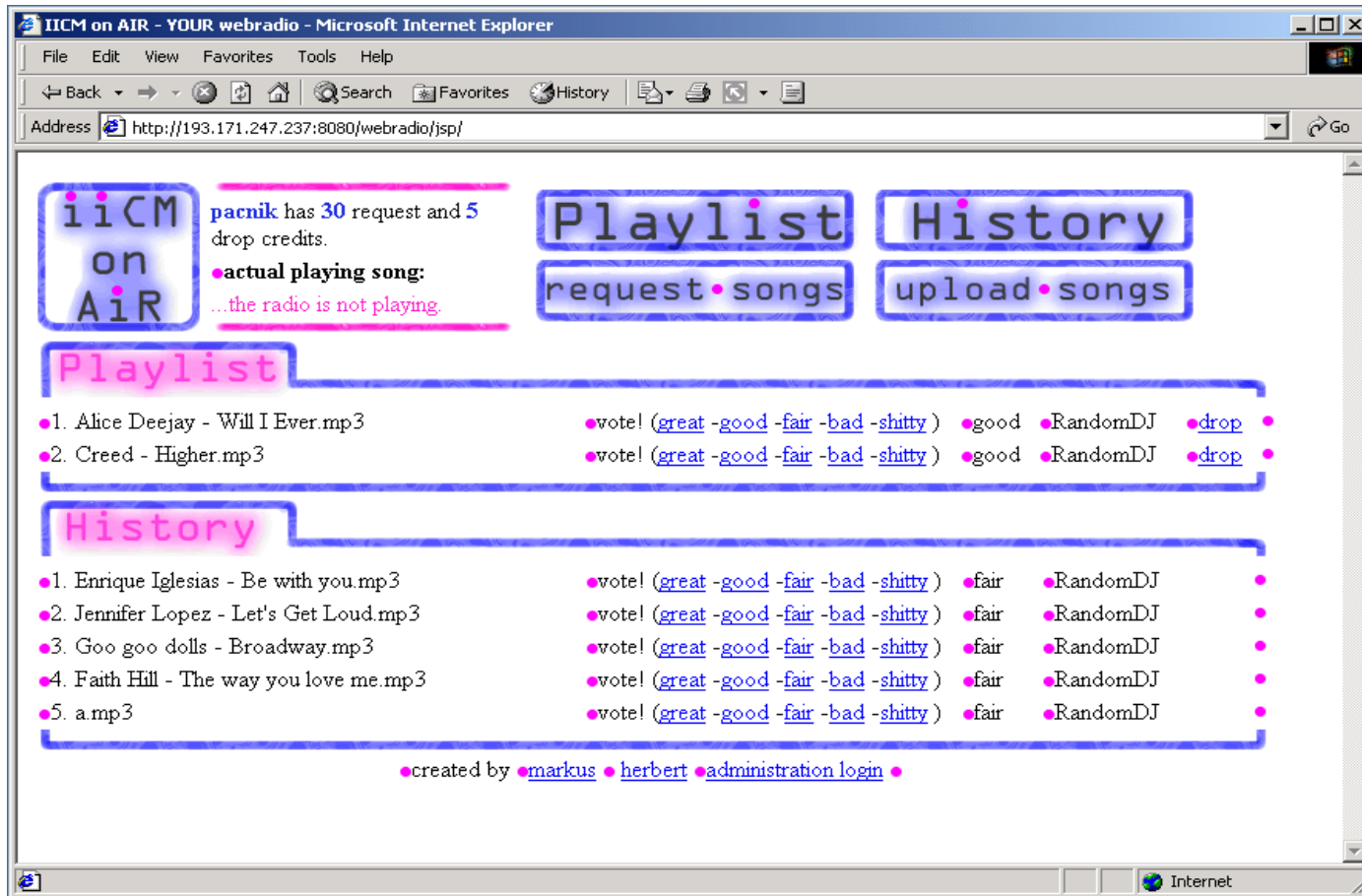
(Miller, Course 6.831 UI Design and Implementation, MIT EECS)

Tehnike računalniških prototipov

- Sekvence slik vmesnika
- Sekvence povezanih slik vmesnika
- Generatorji vmesnikov
 - Omogočajo gradnjo oken v katerih se nahajajo resnične podobe dane palete podob

Orodja za delo s sekvencami slik

- Delovni prototip za “on-line” radijsko postajo



(Andrews, Human-Computer Interaction, 706.021 Mensch-Machine Kommunikation 3VU SS, Graz University of Technology)

Generatorji vmesnikov

- Generatorji vmesnikov v Javi
 - Sun NetBeans
 - Eclipse Visual Editor
 - Borland Jbuilder
- Drugi generatorji vmesnikov
 - xdesigner (Motif)
 - Glade Interface Designer (GTK+)
 - Qt Designer (Qt)
 - Visual Basic, .NET Windows Forms
 - Mac Interface Builder
- Napotek: Ne dovoli, da generator sam izvrši aranžiranje gradnikov. Najprej uporabi absolutno pozicioniranje.

Prednosti in slabosti generatorjev vmesnikov

- Prednosti
 - Dejanski osnovni gradniki, ne le njihove slike
 - Vnaprej zgrajene sestavljene podobe
- Slabosti
 - Omejen nabor standardnih podob
 - “Bogati” grafični vmesniki zahtevajo mnogo kode

(Miller, Course 6.831 UI Design and Implementation, MIT EECS)