

INDHOLD: FÅ EN OPSKRIFT PÅ HVORDAN EFFEKTIVT SAMARBEJDE & DIGITALISERING STØTTER FORRETNINGEN. IMPLEMENTERINGSGUIDE TIL SMÅ LØBENDE DIGITALE FORBEDRINGER.



Digitalisering, udvikling & forbedring, baseret på internationalt anerkendt metodik og platform

Lav selv kontinuerlige digitale produktive & kreative forbedringer.

Få en korrekt vurdering af hvor effektivt jeres samarbejde og digitalisering støtter forretningen.

Kontinuerlige små forbedringer af digitalisering løfter produktiviteten og dermed konkurrenceevnen.

Halting problemet er bevis på at ufuldstændighed er naturligt og logisk – derfor er prototyping nødvendig og logisk – prototyping som preprojekt, sideløbende projekt eller som support og grundlag for videreudvikling og vedligehold - postprojekt.

Et vigtigt skridt i at vise, at ufuldstændighed – Halting problemet, i modsætning til komplet automatik, er at ufuldstændighed er den naturlige og logiske tilstand af digitale elementer. Dette blev bevist af Alan M. Turing i 1936, da han demonstrerede, at der ikke kan være nogen generel procedure til at afgøre, om et selvstændigt computerprogram til sidst vil stoppe – Halting problemet.

For at demonstrere dette resultat, lad os antage det modsatte af, hvad vi vil bevise, er sandt. Antag for det meste, at der er en generel procedure H , der kan afgøre, om et givet computerprogram stopper. Fra denne antagelse skal vi udlede en modsigelse. Dette er det, der kaldes en *reductio ad absurdum* bevis.

Så forudsat eksistensen af H , kan vi konstruere følgende program P , der bruger H som en subrutine. Programmet P kender sin egen størrelse i bits (N) -det er bestemt plads i P for at det indeholder nummeret N -og derefter ved hjælp af H , som P indeholder, P kigger på alle programmer op til 100 gange N bits i størrelse for at se, hvilken standsning og hvilken ikke. Så kører P alle dem, der stopper for at bestemme den produktion, de producerer. Dette er netop sætningen af alle digitale objekter med kompleksitet op til 100 gange N . Endelig udsender vores program P det mindste positive heltal ikke i dette sæt, og derefter stopper P selv.

Så P stopper, P 's størrelse er N bits, og P 's output er et helt tal, der ikke kan produceres af et program, hvis størrelse er mindre end eller lig med 100 gange N bits. Men P har netop produceret dette helt tal som dets output, og det er alt for lille til at kunne gøre dette, fordi P 's størrelse kun er N bits, hvilket er meget mindre end 100 gange N . Modsigelse! Derfor kan en generel procedure H for at afgøre, om programmerne stopper eller ikke, eksistere, for hvis det gjorde det, kunne vi faktisk konstruere dette paradoksale program P ved hjælp af H .

Endelig påpeger Turing, at hvis der var en teori om alt, der altid giver dig mulighed for at bevise, at et enkelt program stopper eller bevise, at det aldrig gør, alt efter hvad der er tilfældet, og ved systematisk at løbe igennem alle mulige beviser, kan du til sidst beslutte, om

enkelte programmer stopper nogensinde. Med andre ord kan vi bruge denne teori til at konstruere H, som vi lige har vist, ikke kan eksistere. Derfor er der ingen teori om alt for Halting/standsingsproblemet.

Lignende ræsonnement viser, at intet program, der er væsentligt kortere end N bits lang, kan løse Turing-halting problem for alle programmer op til N bits lang.

Konklusion:

Uanset hvilket problem som skal løses via apps/programmer er der ingen vej uden om en hybrid løsning i form af enten en prototype eller en parallel sideløbende løsning. Alt dette for få den mest effektive samlede investering og løsning.

Hvad er App Prototyping?

Prototyper en app er nyttig, når du ikke planlægger at offentliggøre i app-butikker med det samme. Vi arbejder ofte med iværksættere og startups, der har en ide til en app, og vi opretter en prototype, der kan vise til potentielle investorer.

Abstrakte ideer kan være svært at forstå. En prototype app er en app, der opfører sig som en reel app, med stikprøve data i stedet for rigtige data. Begyndende med en prototype kan hjælpe folk med at forstå din app ide ved at give dem noget konkret og funktionelt, de kan arbejde med.

En prototype omdanner samtalen, når du pitcher din ide, uanset om du forklarer det til en ven eller taler med en potentiel investor, der hjælper dig med at finansiere den komplette version af din mobilapp.

Hvad vi leverer

Vores prototype-apps er ikke bare grundlæggende HTML-mockups - de er ægte indfødte apps, ligesom du er vant til at downloade fra app-butikker. Vi kan mocke enhver forbindelse til eksterne websteder eller tjenester for at vise andre, hvordan din færdige app vil fungere.

Når du arbejder med vores team for at oprette din app prototype, vil vi være i konstant kontakt med regelmæssige opdateringer og ugentlige leverancer.

I slutningen af dit prototype design engagement leverer vi en smuk prototype app af høj kvalitet, med op til 20 skærme inkluderet. Vores team kan hjælpe dig med at kortlægge de

Alle rettigheder forbeholdes De Bear Technology 2017

www.dbtechnology.dk

info@dbtechnology.dk

skærbilleder, som din prototype skal bruge, og flere skærbilleder kan tilføjes til en ekstra pris.

Din prototype vil fungere på din telefon eller tablet, og kan deles med nogen.

Hvad sker der nu?

Vi er her for at give din app den bedste chance for at lykkes.

Når din prototypedesign er indpakket, kan vi diskutere en pris for at fuldføre din app og offentliggøre til app-butikker eller hjælpe dig med at oprette et pitch-deck for at søge investering.