



Riktlinjer för utveckling av tillgängliga mobilgränssnitt

Senast uppdaterad 2012-04-30

Funka Nu.

Innehåll

Inledning.....	3
Om riktlinjerna.....	3
Riktlinjer för utveckling av tillgängliga mobilgränssnitt.....	4
Val av lösning.....	4
Konstruktion.....	4
Layout och design.....	6
Interaktion.....	8
Innehåll.....	10
Användarinställningar.....	10
Funka Nu AB.....	11
Bolagsfakta.....	11
Kontaktinformation.....	11

Inledning

De här riktlinjerna för utvecklingen av tillgängliga mobilgränssnitt är framtagna av Funka Nu AB i ett projekt finansierat av internetsfonden .SE.

Funkas metodik är utvecklad i nära samarbete med handikapprörelsen och allt vi rekommenderar är testat i verkligheten. Vår verksamhet bygger på de internationella riktlinjerna för tillgänglighet, Web Content Accessibility Guidelines 2.0 (WCAG 2.0). Funkas långa erfarenhet av tillgänglighetsarbete och tester med användare med olika behov och förutsättningar, med och utan hjälpmedel, visar att WCAG 2.0 inte räcker. Vi har därför själva utarbetat testkriterier för punkter som kompletterar det internationella regelverk som offentlig sektor genom EU-beslut ska följa.

2009 fick Funka i uppdrag av W3C att genomföra en auktoriserad översättning av WCAG 2.0 till svenska. Denna översättning kommer att ligga till grund för det arbete med vidareutveckling av Vägledningen 24-timmarswebben som e-delegationen ansvarar för.

- [Web Content Accessibility Guidelines 2.0 \(WCAG 2.0\)](#)
- [Den auktoriserade svenska översättningen av WCAG 2.0](#)
- [World Wide Web Consortium \(W3C\)](#)
- [Web Accessibility Initiative \(WAI\)](#)

Läs mer om Funka under rubriken "[Funka Nu AB](#)" i slutet av detta dokument.

Om riktlinjerna

Allt fler använder pekskärmar och traditionella mobila enheter. De flesta har förutsättningar att fungera väl för användare med funktionsnedsättning, med eller utan behov av hjälpmedel. Därmed ökar kraven på att de som utvecklar appar och mobilgränssnitt har kunskap om tillgänglighet och de olika behov som användarna kan ha.

De internationella riktlinjerna för tillgänglighet, WCAG 2.0, som svensk offentlig sektor ska följa och många andra stora aktörer, exempelvis banker, följer frivilligt saknar utvecklingsprinciper för mobila gränssnitt.

I arbetet med att ta fram riktlinjerna har vi inventerat befintliga riktlinjer och undersökningar om tillgänglighet i mobilgränssnitt. Vi har också genomfört en enkätundersökning bland användare av smartphones för att ta reda på vad som utgör problem och vad som fungerar bra. Men framförallt har vi intervjuat och genomfört praktiska tester med användare med olika typer av funktionsnedsättningar.

De här riktlinjerna är öppna och fria att användas av alla. Vi tar gärna emot feedback och förslag för vidareutveckling, förtydliganden och ändringar.

Riktlinjer för utveckling av tillgängliga mobilgränssnitt

Val av lösning

1. Se till att din grundwebbplats fungerar på mobila enheter

Den behöver inte fungera optimalt, men den ska vara så enkel som möjligt och all funktionalitet ska så långt praktiskt möjligt gå att hantera via ett mobilgränssnitt. Exempel på vanliga problem här är menyer som kräver att muspekaren vilar över ett menyalternativ för att fälla ut en undermeny. En bra strategi är "mobile first", alltså att utformningen av gränssnittet utgår ifrån visningen i en mobil.

2. Tvinga inte användaren att använda en mobilversion men erbjud gärna en sådan om grundwebbplatsens sidor är stora eller funktionerna komplexa

Många användare föredrar en mobilversion av webbplatser. Det gäller inte minst om webbplatsen är omfattande och innehåller många grafiska objekt. Det kan därför finnas skäl till att erbjuda en mobil version, men tvinga inte användaren att använda mobilversionen. Erbjud länkar mellan de olika versionerna och kom ihåg användarens val.

3. En eventuell mobilversion av webbplatsen ska så långt möjligt ge användaren tillgång till samma information och tjänster som den vanliga webbplatsen om det inte är en uttalad mobilversion av en specifik avgränsad tjänst/funktion

Om du erbjuder en mobil version av webbplatsen ska den gå att använda för att göra och läsa samma saker som på den vanliga webbplatsen, men i ett enklare format där exempelvis stora mängder information och tillval göms i första läget i stället för att alltid ligga synligt. Ett undantag kan vara om det är en mobilversion av en tjänst/funktion, exempelvis en biljettbokning hos ett reseföretag. Då ska mobilversionen ses som ett enklare alternativ till denna tjänst/funktion.

4. Skapa applikationer för klart avgränsade funktioner som en användare kan behöva tillgång till ofta

Applikationer (appar) fungerar i första hand bra för klart avgränsade uppgifter, till exempel för att visa aktuell trafikinformation/störningar. Applikationen bör i första hand vara skapad för väl avgränsade uppgifter som en användargrupp kan tänkas behöva utföra/nå ofta.

Konstruktion

5. Följ WCAG 2.0 utom där de här riktlinjerna motsäger WCAG 2.0

WCAG 2.0 appliceras i första hand på webbgränssnitt men flera delar av riktlinjerna kan också appliceras för applikationer, exempelvis framgångskriterier som handlar om uppmärkning av icke-textobjekt och framgångskriterier om kontraster.

6. Vid skapande av applikationer för specifika enheter ska eventuella design- och tillgänglighetsriktlinjer följas så länge dessa inte motsäger de här riktlinjerna

Om det finns specifika riktlinjer för utveckling av applikationer för den valda enheten/operativsystemet ska dessa följas. Exempelvis ska Apples riktlinjer följas vid utvecklingen av applikationer för iPhone.

7. Om du utvecklar en applikation för en specifik plattform ska denna stödja de för plattformen karakteristiska egenskaperna

Till exempel ska det så långt möjligt gå att zooma in/ut via att knipa/dra isär två fingrar. Många användare är vana vid detta sätt att interagera på pekskärmar och det är därför viktigt att erbjuda möjligheten om det är praktiskt möjligt. Tänk dock på att det också måste finnas möjlighet att göra den typen av inställningar med ett finger, så för just exemplet med knipande rörelser med två fingrar bör gränssnittet också erbjuda ett alternativ för att zooma, exempelvis två knappar för att zooma in och ut. På Androidtelefoner finns oftast fysiska knappar som ska fungera på ett för användaren invariant sätt, medan telefoner med iOS i stället använder mjukvaruknappar längst ner på skärmen.

8. Märk upp grafiska element, ikoner och knappar med deras motiv eller funktion

På webbsidor ska meningsbärande bilder ha en alt-text, på samma sätt ska bilder och knappar i applikationer förses med en beskrivning. Exakt hur det görs varierar i de olika operativsystemen.

9. Varje formulärobjekt ska antingen ha en ledtext eller en beskrivning

Formulärobjekt, exempelvis textfält, radioknappar och kryssrutor, ska så långt möjligt ha en synlig ledtext som kopplas till formulärfältet på ett korrekt sätt. På webbsidor görs detta med elementet label. Om en ledtext inte kan ge hela instruktionen kan formulärobjektets title-text användas för att ge användaren information. Dock måste instruktionen alltid även finnas visuellt synlig på sidan, eftersom användare utan skärmläsare i de flesta mobila enheter inte kan få fram en title-text.

10. Använd inte ramar i webbgränssnitt

Ramar (frames) eller inline-ramar (iframes) fungerar dåligt i många mobila enheter och ska därför undvikas. Om en iframe måste användas för en tjänst bör du så långt möjligt också erbjuda användaren en länk för att visa funktionen i ett eget fönster utan ramar. Man kan till exempel tänka sig en integration av en karta i en iframe på den vanliga webbsidan, men samtidigt ge användaren en länk för att visa kartan i ett eget fönster utan ramar.

11. Hjälp användaren vid inmatning genom att anpassa det virtuella tangentbordet efter vad som ska matas in

I webbaserade gränssnitt kan detta åstadkommas genom att använda html5 för att markera olika typer av inmatningsfält, exempelvis telefonnummer, e-post, eller text.

12. Minimera användningen av script på klientsidan

Mobila enheter har ofta sämre prestanda än vanliga datorer och användningen av många script kan orsaka problem. Kräv inte mer av enheten än vad som är nödvändigt.

13. Genomför praktiska tester av lösningen

Oavsett hur väl man följer riktlinjerna vid utvecklingen av en ny applikation eller

ett nytt webbgränssnitt som ska fungera för mobila enheter är komplexiteten större. Det skapar tillgänglighetsproblem som kan vara svåra att förutse. Man ska därför alltid testa lösningen praktiskt med personer som inte varit med vid utvecklingen. Man bör alltid inkludera testpersoner med funktionsnedsättningar i användartester och låta tillgänglighetsexperter tolka resultatet.

Layout och design

14. Positionera viktiga saker högre upp och mindre viktiga saker längre ner i scrollande vyer

Eftersom skärmarna ofta är mindre på mobila enheter bör den viktigaste informationen positioneras högt upp för att se till att den syns utan att det krävs scroll. Men tänk också på att det är svårt att nå att klicka på objekt allra överst på skärmen. Viktig interaktion bör alltså inte ligga allra överst.

15. Gruppera delar som hör ihop

Detta gäller generellt men blir extra viktigt att hantera vid utveckling med Responsive design. På många webbplatser ligger relaterade länkar till höger vilket ofta funkar när det visas på en stor skärm. Om mindre skärmar får högerspalten att åka ner under innehållspalten kan det däremot bli mycket scroll för att hitta relaterade länkar till en del av innehållet. Då bör sidan så långt möjligt flöda om på ett sätt att den relaterade informationen positioneras direkt efter det område det är relaterat till i stället för att allt relaterat material placeras tillsammans längst ner.

16. Sträva efter att skapa en ren design och minimera antalet "onödiga" objekt

Ett stort problem är sidor som innehåller många objekt som inte användaren upplever som intressanta/viktiga för användningen. Webbplatser som utformats för visning på vanliga datorer med skärmar med hög upplösning inbjuder till en design med många objekt och områden. När sådana sidor visas på en liten skärm skapas stora problem för användaren eftersom sidan tar lång tid att ladda och kräver mycket scroll. Om en sida exempelvis har en högerspalt med reklam bör denna flödas om på en mindre skärm och läggas sist, om den inte går att ta bort helt.

17. Sträva efter att göra sidhuvudet litet

I mobila enheter är det ofta problem med mycket scroll. Genom att minimera sidhuvudet kan du minska problemet samtidigt som menyer och innehåll kan göras synligt direkt när sidan laddas.

18. Skapa stora klickytor

Eftersom enheternas skärmstorlek, dpi och upplösning varierar går det inte att ange ett exakt mått. Det är också skillnad på en webbsida och en applikation, men sträva efter att klickytan åtminstone ska vara brödtextens radhöjd åt ena hållet och brödtextens radhöjd * 3 åt andra hållet. En ikon i en app bör ha en bredd och höjd på minst 9 millimeter.

19. Lägg inte ofta använda knappar ute vid höger-/vänsterkanten om de inte upptar minst en tredjedel av skärmbredden

Viktiga knappar bör i första hand placeras centralt och relativt långt ner på skärmen eftersom det är svårt att trycka på knappar ute vid kanten för

användare som bara använder en hand eller som måste balansera mobilen på ett knä för att kunna använda den. Det gäller exempelvis användare med nedsatt motorisk förmåga som kan ha svårt att hålla i telefonen.

20. Högerjustera inte knappar, funktioner eller grupper med knappar och funktioner om inte gruppen med knappar/funktioner sträcker sig över minst 75 % av skärmen i alla lägen

Användare som inte ser sidan använder pekfingret för att scanna av gränssnittet. Telefonen läser upp det användaren pekar på. Detta görs mest naturligt uppifrån övre vänstra hörnet och nedåt på sidan. Knappar som ligger långt ute till höger utan att det finns något annat på samma rad blir mycket svåra att upptäcka.

21. Orientera knappar och länkar på tydliga rader (horisontellt och vertikalt)

Det gör det lättare för användare som inte ser gränssnittet att hitta dem. Hittar användaren en knapp blir det lättare att hitta de andra knapparna också. Detta skapar också en tydligare visuell överblick för seende användare.

22. Ledtexter till inmatningsfält ska i första hand positioneras ovanför fältet

Undantag för kryssrutor och radioknappar där texten kan ligga till höger om knappen/rutan. Grupper med radioknappar och kryssrutor ska dock ha en överskrift inlagd som anger vad gruppen har för funktion. Denna ska positioneras ovanför gruppen med radioknappar/kryssrutor.

23. Radlängder ska anpassas efter skärmbredden men alltid hålla sig till max 70 tecken per rad inklusive mellanslag

Det ska så långt som möjligt inte krävas scroll i sidled för att läsa en rad i innehållet. Samtidigt bör inte radlängden bli så kort att enskilda ord måste delas upp på flera rader om det inte är en naturlig brytning, exempelvis "innehållsförteckning". Målsättningen bör vara radlängder som håller sig till 55-60 tecken inklusive mellanslag per rad.

24. Begränsa informationsmängden och antalet objekt som visas

För att underlätta för användare med små bildskärmar är det bra att begränsa mängden objekt och text som visas. Det betyder inte att du ska ta bort delar, men det kan underlätta för användaren om du döljer delar i exempelvis så kallade dragspelsfunktioner (du klickar på en rubrik för att fälla ut den underliggande informationen). Då får användaren en snabb överblick men också enkel tillgång till all eventuell information och funktionalitet. Ett annat sätt att dölja objekt är att lägga menyer och länkgrupper i utfällande menyer. Tänk dock på att funktionaliteten måste vara tydlig, det ska vara intuitivt för användaren att få fram dolda delar.

25. Använd kända ikoner

Uppfinn inte egna utseenden på vanliga ikoner, utan återanvänd utseenden som användaren har en chans att känna igen sedan tidigare.

26. Utforma klickbara objekt så att de ser klickbara ut

Utforma länkar så att de ser ut som länkar. Använd inte enbart färg för att visa att något är länkat. Länkarna blir då bland annat svåra att uppfatta i direkt

solljus. Gör knappar tredimensionella och återanvänd kända utformningar och placeringar av ikoner.

27. Använd höga kontraster

Många användare uppger att det är svårt att se det som visas på skärmen när mobilen används i direkt solljus. För att underlätta användandet är det viktigt att alltid sträva efter att ha bra kontraster. Brödtext och text till ikoner bör så långt som möjligt presenteras som svart text på vit bakgrund, eller tvärtom, om inte texten är stor eller kan zoomas. Text som kan zoomas eller som i utgångsläget är stor bör minst följa WCAG 2.0:s strängare punkt, 1.4.6.

28. Det ska vara möjligt att använda gränssnittet i både stående och liggande visning

Interaktion

29. Använd enkla navigationskoncept

När en webbsida visas på en vanlig skärm finns det ofta gott om plats att låta navigationen ta stora ytor i anspråk. Ett exempel på det är så kallade megamenyer som ofta visar 2-3 nivåer i menystrukturen samtidigt. I mobila enheter fungerar detta dåligt. Här behöver menyerna utformas så att de tar lite plats och har en visuellt tydlig uppställning. För en webbtjänst som ska fungera både på en dator och i en mobil enhet kan det då i vissa lägen finnas behov av att låta menyn visas på olika sätt beroende på skärmbredden.

30. Om du utvecklar en applikation för ett operativsystem eller en mobil enhet som kan ha styrknappar (exempelvis piltangenter och en ok-knapp) ska dessa gå att använda för att navigera i gränssnittet

Det här gäller idag exempelvis Android. Den fysiska tillbaka/backa-knappen ska alltid fungera.

31. Om du utvecklar ett gränssnitt som kan användas av enheter där du kan koppla in ett tangentbord ska gränssnittet så långt det är möjligt gå att styra med tangentbordet

32. Lägg in genvägar för att låta användaren hoppa mellan delar i innehållet i långa sidor

En genväg bör vara visuellt dold i utgångsläget men dyka upp när den får fokus vid tangentbordsnavigering.

33. Minimera textinmatning i gränssnittet

Textinmatning är både svårt och tidskrävande i mobila enheter och bör därför om möjligt undvikas. Ett sätt att undvika det är att erbjuda listor med val i stället för att kräva textinmatning samt att tillhandahålla "auto complete" (gränssnittet föreslår fraser när användaren börjat mata in text).

34. Om gränssnittet medger styrning med gestik bör detta implementeras

Gestik är ett sätt att styra en enhet genom att göra olika gester på skärmen med ett eller flera fingrar. Exempelvis kan användaren på iPad ofta bläddra mellan olika sidor genom att dra fingret över skärmen. I många gränssnitt kan man också zooma ut/in genom att föra två fingrar ifrån varandra eller mot varandra.

Gestiken ska vara intuitiv och konsekvent. Använd de gester användarna är vana vid.

35. Lägg inte in funktioner som enbart går att hantera via gestik, utan komplettera alltid med en knapp/länk

36. Gör det möjligt att styra gränssnittet med bara ett finger

Det kan finnas situationer där det inte går men så långt möjligt ska det gå att styra all funktionalitet med bara ett finger. Det kan kräva att knappar döljs och dyker upp när man rör ett visst område på skärmen eller klickar på en annan knapp.

37. Var konsekvent

Lägg till exempel knappar med en viss funktionalitet på samma ställe på skärmen och utforma dem konsekvent.

38. Använd inbyggda objekt som det är tänkt att de ska användas och som användaren förväntar sig att de ska användas

Exempelvis innehåller ofta operativsystemen inbyggda komponenter/widgets som en applikation kan använda i stället för att utveckla implementera egna komponenter med motsvarande funktionalitet. På enheter som har fysiska knappar ska dessa så långt möjligt stödjas av applikationen.

39. Ge feedback till användaren

När användaren gör en inmatning bör detta bekräftas både med ett ljud och med en kort vibration om enheten stödjer detta. Det bör dock också finnas möjlighet att stänga av feedbacken. Notera att inmatning inte nödvändigtvis måste vara text via ett tangentbord. Det kan vara ett röstkommando, ett taget foto, en gest eller en rörelse med mobilen. Feedback bör ges i de flesta fall men det kan finnas undantag då det skulle upplevas som störande med för mycket feedback (exempelvis ska inte en applikation som fungerar som en stegmätare ge feedback för varje registrerat steg).

40. Ge tydlig statusinformation till användaren

Många använder mobila enheter i stressade situationer. Det är då viktigt att användaren hela tiden får veta vad som händer, i synnerhet när användaren väntar på applikationen/webbsidan. Om exempelvis applikationen/webbsidan håller på att ladda data är det bra att visa hur långt laddningen kommit. Ge alltid tydlig statusinformation. Det är en fördel att erbjuda både visuell feedback och feedback med ljud.

41. Ge användaren tillräcklig tid och varna innan tidsgränser uppnås

Många använder mobilen på resande fot. Det är då vanligt att det uppstår avbrott i användningen på grund av yttre omständigheter. Detta kan till exempel vara att användaren använder mobilen i väntan på bussen. När bussen kommer uppstår ett avbrott i användningen när man går på bussen. Under den tiden får mobilen vila. Det är då viktigt att applikationen/tjänsten ger användaren tillräckligt med tid och varnar användaren när tiden håller på att löpa ut. Går det att lösa bör det också finnas möjlighet att förlänga tiden på ett enkelt sätt. Det vanligaste exemplet på tidsgränser är automatiska utloggningar.

42. Hjälp användaren att undvika fel och att korrigera eventuella fel

Detta är extra viktigt i mobila enheter där det är lätt att klicka på fel knapp. Tekniker för att hjälpa användaren undvika att göra fel är till exempel "auto complete" och sökförslag. Om fel ändå uppstår ska det tydligt meddelas till användaren både högt upp på sidan och där felet uppstått. Ge också så långt möjligt förslag på lösningar.

Innehåll

43. Använd bara bilder om det verkligen hjälper användaren

Bilder är bra för att förmedla information men i mobila enheter fungerar de ofta mindre bra. Dels eftersom de blir små och dels eftersom de tar längre tid att ladda. Bilder ska därför användas enbart när de verkligen hjälper användaren. Dekorationsbilder bör minimeras och så långt möjligt placeras i css-koden.

44. Använd korta men beskrivande rubriker för att strukturera informationen

Om det finns möjlighet att ange vad som är rubriker i koden ska detta göras. För webbsidor innebär det html-elementen H1-H6. En grundregel är också att hålla rubrikerna relativt korta, men de får inte bli så korta att de inte på ett begripligt sätt för användaren sammanfattar vad man kan läsa om under rubriken.

45. Undvik förkortningar

Även om mobila enheter med en mindre skärm kan locka till att använda förkortningar bör detta undvikas. Förkortningar av organisationsnamn och liknande kan användas om de förklaras första gången de används.

Användarinställningar

46. Se till att det går att zooma gränssnittet

I en applikation måste en förstoringfunktion byggas in medan en webbplats måste gå att zooma på ett för webbläsaren naturligt sätt.

47. Överväg att tillhandahålla en inställning för att invertera färger

Om applikationen exempelvis har mörk text på ljus botten bör användaren få möjlighet att ställa om så att det istället är ljus text på mörk botten, och visa versa.

48. Överväg att tillhandahålla en inställning för att ändra typsnitt

Funka Nu AB

Funka startade som ett projekt inom handikapprörelsen. I dag är Funka marknadsledande inom området tillgänglighet med 80 procent av Sveriges myndigheter som kunder. Sedan år 2000 är vi ett privatägt bolag och vår nära relation till handikapprörelsen innebär en unik kvalitetskontroll. Vi har kontor i Stockholm och Oslo och kunder över hela världen.

Funkas arbete sätter normer för utveckling och analys samt utformar krav vad gäller tillgänglighet. Vi finns med i de flesta internationella arbetsgrupper av betydelse och genomför regelbundet egna undersökningar. Funka driver därmed tillgänglighetsfrågan aktivt både i norden och EU.

Funkas konsulter har varit med och tagit fram Handikappombudsmannens riktlinjer för en tillgänglig statsförvaltning och Vervas Vägledning för 24-timmarswebben. I vägledningen har vi inarbetat Funkas metodik i avsnittet om tillgänglighet. Funka har dessutom drivit fram rekommendationer om snabbkommandon och ikoner.

Funka är EPiServer Solution Partner, Microsoft Partner och Adobe Certified Training Provider. Vi står bakom [Funkaportalen](#), en mötesplats för funktionshinderfrågor.

Bolagsfakta

Styrelsens säte: Stockholm

Styrelsen består av: Jan Friedman (ordförande), Mats Wennberg, Lennart Engelhardt, Johan Jakobsson och Shahzad Rana.

Antal anställda: 22

Omsättning 2010: 17 MSEK

Kontaktinformation

Funka Nu AB
Döbelngatan 21
111 40 Stockholm

Tel: 08-555 770 60

Fax: 08-55 11 32 36

E-post: kontakt@funkanu.se

Webb: www.funkanu.se

Funka Nu.