

2014-12-12

Linköpings Universitet
Tekniska högskolan
ITN



Projektrapport

InterHelios

Program:

Civilingenjör i Medieteknik

Kurs:

TNM040 – Kommunikation och användargränssnitt

Gruppmedlemmar:

Adam Alsegård (adaal265)

Matthias Berg (matbe790)

Daniel Böök (danbo324)

Jonas Kinnvall (jonki910)

Benjamin Wiberg (benwi631)

Innehållsförteckning

| | |
|----------------------------------|----|
| 1. INTRODUKTION | 2 |
| 2. BESKRIVNING AV PROJEKTUPPGIFT | 2 |
| 2.1 Om spelet | 2 |
| 3. METOD OCH ARBETSFÖRDELNING | 3 |
| 3.1 Utvecklingsverktyg | 3 |
| 3.2 Tidsåtgång | 4 |
| 3.3 Arbetsfördelning | 4 |
| 3.4 Användartest | 4 |
| 3.5 Databas | 5 |
| 4. RESULTAT | 5 |
| 4.1 Tutorial | 5 |
| 4.2 Main menu | 6 |
| 4.3 Settings | 7 |
| 4.4 Upgrades | 7 |
| 4.5 Game | 8 |
| 5. DISKUSSION | 10 |
| 6. SLUTSATS | 11 |
| BILAGOR | |

1. Introduktion

Varje dag stöter vi på olika typer av gränssnitt, oftast utan att reflektera över deras design eller hur de ska användas. De gånger vi reagerar på själva gränssnittet är det oftast för att designen är dålig och att det ej är användarvänligt. Bra design syns oftast inte medan vi stör oss på och uppmärksammar dålig design.

Under utvecklingen av ett gränssnitt är det viktigt att kommunicera med användare för att kunna anpassa designen till sin målgrupp så bra som möjligt. Gränssnittet behöver utvärderas för att kunna vidareutvecklas, gärna med ett antal iterationer för bästa resultat. Den här rapporten sammanfattar och utvärderar projektgruppens arbete med sitt gränssnitt så här långt.

2. Beskrivning av projektuppgift

Syftet med projektet var att ta fram ett väldesignat gränssnitt för en vald målgrupp. Huvudmålet var att tillämpa Norman's designprinciper¹ i utvecklingen av gränssnittet samtidigt som gruppen fick möjlighet att lära sig mer om programmering genom ett lite mer avancerat programmeringsprojekt.

Projektgruppen valde att utveckla ett spel för mobila Android-enheter. Målgruppen bestod av människor i alla åldrar med visst spelintresse och spelvana. En vision för gränssnittet som tidigt togs fram var att

- applikationen ska kunna användas på alla olika Android-enheter
- spelet ska kunna spelas var som helst, i alla miljöer och situationer
- spelet ska ha flera menyer med olika alternativ med implementerad funktionalitet
- spelet ska ge en lättillgänglig och rolig spelupplevelse
- spelet ska ha lättförstådda gränssnitt och kontroller överlag

2.1 Om spelet

Spelets genre valdes till en "top-down space shooter". Gruppen skrev upp ett antal mätbara grundmål för spelet som skulle vara klara vid projektets slut:

- Minst 70 % av användarna skulle ifrån startmenyn vilja starta spelet direkt och klara av dess kontroller utan att behöva titta i instruktionerna.
- Användare utanför målgruppen ska enkelt kunna läsa sig till hur spelet fungerar.
- Alltid ha väl synliga knappar (minst 1x1cm).
- Gå att använda på alla mobila Android-enheter.
- Fungera att spela var som helst.
- Minst en färdig spelbar nivå eller "bana".
- Spelprogression i form av uppgraderingar till spelarens skepp eller liknande som uppmuntrar till fortsatt spelande.

¹ <http://www.csun.edu/science/courses/671/bibliography/preece.html>

I mån av tid skulle fler banor, musik, fiende- och vapenvariation samt andra funktioner som förhöjer spelupplevelsen implementeras.

3. Metod och arbetsfördelning

I början av projektet diskuterade projektgruppen olika tänkbara spelidéer samt idéernas styrkor och svagheter. Spelidén som slutligen valdes utmärkte sig eftersom den var ett redan känt koncept och gav goda utvecklingsmöjligheter utan att kräva för komplexa system.

En användarundersökning var tänkt att genomföras i ett tidigt skede av projektet, med handritade bilder på gruppens gränssnittsidéer, för att få en bra start i rätt riktning. Detta övergavs dock eftersom gruppen ansåg att det behövdes en bättre bild av vad som faktiskt var möjligt att realisera innan en undersökning genomfördes. Det dröjde därför till senare i projektet innan applikationen testades på användare. Detta medförde dock att användarnas åsikter då direkt kunde implementeras i applikationen.

3.1 Utvecklingsverktyg

Gruppen valde att utveckla applikationen till Android eftersom det är den vanligast förekommande enheten samt att det är gratis att utveckla till Android. Gruppmedlemmarna hade även tidigare erfarenhet av programspråket Java som används till Android.

Googles dedikerade Android-IDE, Android Studio², valdes som utvecklingsmiljö till applikationen eftersom det var den utvecklingsmiljö som använts under laborationerna i kursen. Android Studio har verktyg och hjälpmedel för att designa utformning av grafiska gränssnitt samt stöd för emulering av Android-enheter. Vidare lånades två Android-plattor av modellen Google Nexus 7 för att testa applikationen på. De datorerna som klarade av att testköra spelet på plattorna var två av gruppmedlemmarnas privata MacBooks, de tre andra gruppmedlemmarna använde sig av Android Studios enhetsemulator för att testa applikationen under utvecklingsprocessen.

För att hålla ordning på versionshanteringen användes verktyget Git, där projektets kod lagrades med hjälp av tjänsten GitHub³. Detta gjorde att gruppen kunde arbeta på olika håll och sedan på ett enkelt sätt lägga ihop koden utan större besvär. AndEngine⁴ användes som spelmotor, en lättanvänd open source-motor skriven i Java.

² Beta version 0.8.14 användes.

³ <https://github.com/about>

⁴ <http://www.andengine.org/blog/>

3.2 Tidsåtgång

Projektgruppen enades i början av projektet om en preliminär tidsplan (se *Tabell 1*) för hela projektarbetet med syfte att ge gruppmedlemmarna en bättre överblick av tiden som fanns till förfogande.

Tabell 1 - tidigt tidsupplägg

| Vecka | 43 - 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 |
|----------|-------------------------|-----------|------|------|-----------------------|---------------------------|-------------------------------|----------|
| Att göra | Läsa på om att spelkoda | Förstudie | Koda | Koda | Första versionen klar | Utvärdera och färdigställ | Förbered redovisning, rapport | Redovisa |

För att hålla tidsschemat lades en veckoplanering upp. Varje måndag lunch ägnades åt ett projektmöte där föregående veckas jobb sammanfattades samt kommande veckas arbete planerades. Här specificerades det vem som skulle jobba med vad. Minst en eftermiddag per vecka skulle gruppen sedan sitta och jobba gemensamt. Schemalagda tillfällen i datorsalar utnyttjades men gruppen arbetade även gemensamt utanför schemalagd tid nästan varje vecka. Enskilt arbete har även genomförts annorstädes fortlöpande under projektets förlopp.

3.3 Arbetsfördelning

Då gruppmedlemmarna hade olika mycket erfarenhet av programmeringsspråket Java sedan tidigare var medlemmarnas startsträckor olika långa för att komma igång. Arbetet delades sedan upp så att några av gruppmedlemmarna direkt började arbetet med spelets menyer och deras design. De övriga påbörjade implementationen av själva spelet.

När projektet fortskred blev det tydligare vilka gruppmedlemmar som jobbade med vad. En gruppmedlem fick huvudansvaret för designen av knappar, bilder och menyer. En medlem hade huvudansvaret för funktionaliteten av spelaren och fiender medan en annan letade passande musik och ljudeffekter. De två resterande parprogrammerade och löste blandade uppgifter.

3.4 Användartest

Två användartester genomfördes. Det första testet (se Bilaga 1) genomfördes så fort applikationen hade grundläggande funktionalitet och hade i syfte att undersöka gränssnittet, om det levde upp till gruppens mål och i vilken riktning menyerna och spelet borde utvecklas åt.

Användartest 2 (se Bilaga 2) hade som syfte att se ifall användarna generellt upplevde en förbättring från första testet eller om de fortfarande upplevde samma sorts brister, samt återigen få tips om utvecklingsmöjligheter. Vissa av deltagarna var med på båda testerna.

Båda användartesterna genomfördes som öppna undersökningar där användaren fick testa applikationen utan några instruktioner och uppmuntrades till att "tänka högt".

Projektgruppen tog anteckningar på användarens tankar och vilka knappar hen tryckte på och i vilken ordning. I slutet av testet ställdes frågor för att ta reda på i vilken riktning användarna tyckte att applikationen borde fortsätta i.

Projektgruppen har inte använt sig av några konkreta personas eller "user stories" då dessa ansågs ta för mycket tid och fokus från programmeringen under ett så litet projekt som detta. Dock har ett liknande synsätt använts av gruppen när applikationen diskuterades från ett användarperspektiv.

3.5 Databas

För att lyckas få en känsla av progression samt att förbättra spelupplevelsen hade gruppen tidigt planer på att implementera en databas för att spara användarens alla val. Både Settings och Upgrades läser och skriver till databasen vilket ger spelaren en känsla av kontinuitet. Databasen sparas på enheten och behåller därmed värdena även när applikationen stängs av.

4. Resultat

Applikationen består av fem huvudsakliga vyer. Nedan följer en beskrivning av samtliga vyer, granskade ur ett användarvänlighetsperspektiv, samt motivering till de designval gruppen gjort.

4.1 Tutorial

Användaren möts vid första användningen av en Tutorial-vy (se Bild 1) som ger lite bakgrundshistoria och sedan förklarar hur menyerna och spelet går att använda. För den ointresserade så går det att stänga av Tutorial-vyn på valfritt blad genom att trycka på **skip**-knappen, som även är lite större än de andra knapparna för att dra blickarna mot sig. Tutorial-vyn visas inte automatiskt efter första användningen. Detta eftersom projektets mål var att ha en spelupplevelse som inte krävde massor av förklaringar. För den osäkre användaren går det att åter få upp Tutorial-vyn genom att klicka på en knapp med ett frågetecken på i huvudmenyn, även detta går hand i hand med gruppens grundmål.

Ordvalet **skip** indikerar att det finns fler sidor i Tutorial-vyn och fungerar därför som en signifier. Det är också den knappen som har bäst synlighet för att uppmuntra de spelsugna användarna att gå direkt till spelet. När sista bladet tryckts fram ändras texten till **close** för att indikera att det inte finns fler blad. Knapparna **next** och **back** syns endast när de går att

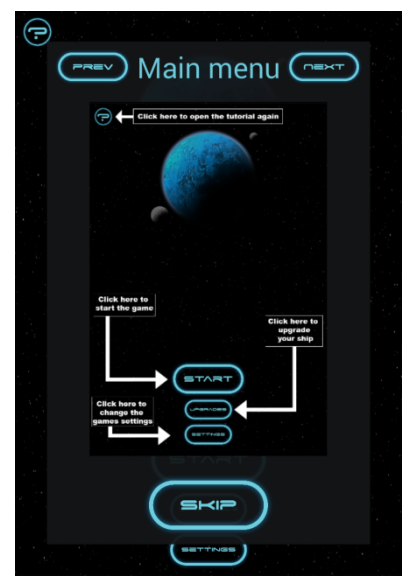


Bild 1: Tutorial-vyn, blad 2

använda, för att få bästa möjliga affordans. Vyn har **next** uppe till höger och **back** uppe till vänster, likt rådande konventioner, och placeringen är konsekvent genom alla blad.

När Tutorial-vyn är öppen så är knapparna i Main menu-vyn inaktiverade. Tutorial-vyn går även att stänga genom att trycka på frågetecknet igen då användartester visade att detta var önskvärt.

Möjlig vidareutveckling av vyn skulle vara att lägga till förklarande bilder för hur själva spelet fungerar.

4.2 Main menu

I denna vy finns tre knappar i mitten av skärmen samt en uppe i vänstra hörnet (se *Bild 2*). Ett av ursprungsmålen var att majoriteten av användarna skulle klicka på **start** det första de gjorde. Därför placerades **start** överst av knapparna samt gjordes lite större än de andra. Användartesterna visade att över 80 % tryckte på **start** det första de gjorde eftersom de genast ville börja spela. Detta tyder på bra synlighet, affordans och mappning. De tre knapparna har utrymme mellan varandra och är tillräckligt stora för att användaren inte skall råka trycka fel. I övrigt är vyn ganska ren vilket ger en lagom densitet och trevlig balans.

Samtliga knappar i applikationen ger omedelbar återkoppling då de lyser upp eller skiftar färg vid beröring (se *Bild 3*). Dessutom spelas ett ljud upp vid alla knapptryckningar, som en ytterligare återkoppling på att ett kommando har utfärdats. Det som spelas upp är korta ljudklipp som alla låter likadant förutom för **start**. För just **start** spelas ljudet av en flygplansmotor som startar, vilket även är en signifier på att spelet är i uppstart. Innan spelet kommer igång så visas en "loading screen" (se *Bild 4*) istället för en svart skärm för ytterligare återkoppling till spelaren att spelet håller på att laddas.

De tre huvudsakliga knapparna har placerats långt ner på skärmen för att vara lättåtkomliga för alla användare, oavsett enhet.

Frågetecknet, som leder till Tutorial-vyn, placerades däremot lite mer svårtillgängligt, men synligt. Androids inbyggda tillbakaknapp är inaktiverad i alla vyer för att användaren inte ska råka trycka på den av misstag. För att gå tillbaka till föregående vy behöver användaren trycka på **back**, vilket är konsekvent på alla vyer och ger därmed god mappning.

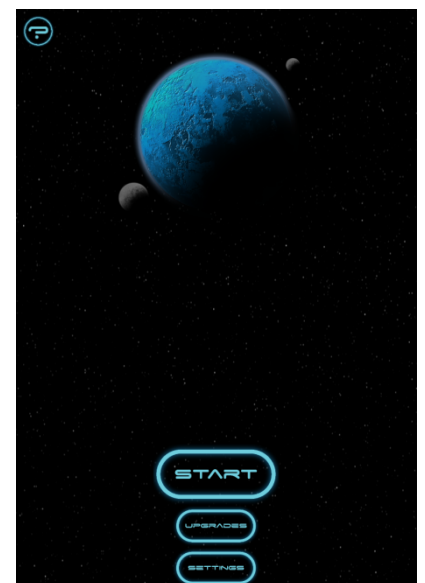


Bild 2: Main menu-vyn



Bild 3: Knapp med återkoppling



Bild 4: Loading screen

När huvudmenyn först blir tillgänglig börjas bakgrundsmusik spela. Musiken kommer att spelas oavbrutet oavsett om användaren skulle byta vy. Detta ger god kontinuitet igenom menyerna, användaren kommer ej störas på att samma musikslinga börjar spela från början vid varje vald aktivitet. Utöver kontinuiteten ger bakgrundsmusiken en mer dynamisk upplevelse för användaren. Samtliga vyer känns mer levande utan att skapa distraktioner på skärmen.

4.3 Settings

Användartesterna visar att den generella användaren genast förstår vad Settings-vyn (se *Bild 5*) har för funktion. Dels tack vare dess namn, och då på grund av gällande konvention, men även för en klar design. Placerade i mitten av vyn finns två "slidebars" som har tydlig affordans, de inbjuder användaren att inkrementellt välja ett värde med hjälp av pluppens placering. Deras synlighet är god då de förklarande orden **music** och **sound** står ovanför respektive slidebar. Återkopplingen ges här dels av att pluppens ring förstärks samt att markören flyttas men även av att ett ljud spelas efter ljudnivån har ändrats. Användaren förstår enkelt från denna återkoppling vilken ändring som skett. Eftersom ljud och musik är kopplade till databasen sparas dessutom användarens val. God densitet mellan elementen minimerar feltryck.

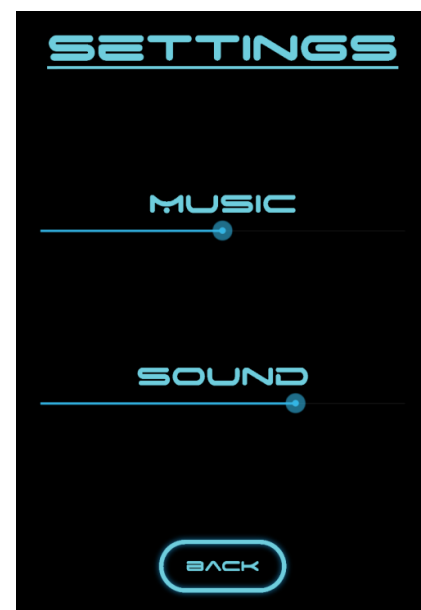


Bild 5: Settings-vyn

Möjlig vidareutveckling av vyn skulle vara att implementera funktioner så att användaren kan välja mellan att använda olika styrfunktioner, "flip screen-orientation" och kortkommando för "mute".

4.4 Upgrades

I vyn Upgrades (se *Bild 6*) erbjuds användaren att uppgradera sitt skepp med hjälp av sina inspelade pengar. Det viktigaste för användaren att hålla koll på är därmed **cash**, som därför är isolerat från de andra och även fått en utmärkande bakgrund, vilket medför god synlighet. Detta även för att belysa att det inte har samma funktionalitet som de andra rubrikerna.

För god mappning har sammanhörande element placerats på hophängande rader, åtskilda av tydliga typografiska gränser. För att få bra affordans har köpknappen placerats i höjd med gällande uppgradering. Om köp är möjligt så är knappen blå och priset för att uppgradera en nivå står utskrivet, men om spelaren kommit till maxnivån eller inte har tillräckligt med pengar så byts knappens färg

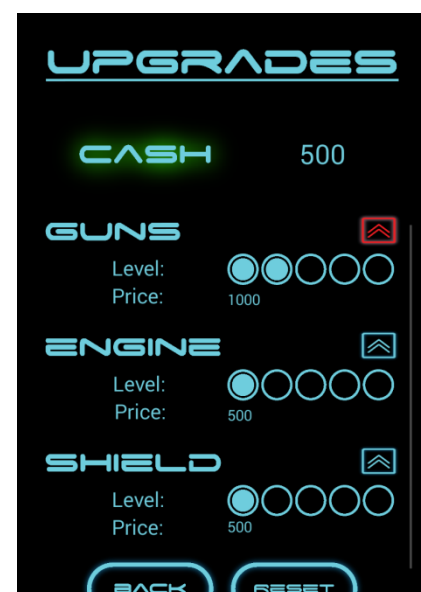


Bild 6: Upgrades-vyn

till rött för att indikera, med rådande färgkonvention, att köp ej längre är möjligt. Hur många nivåer det finns av varje uppgradering indikeras av antalet ringar, när spelaren uppgraderat så fylls ringarna i, vilket innebär både bra synlighet, tydlig återkoppling och god mappning. När spelaren kommit till maxnivån är alla ringar ifyllda och där priset tidigare fanns står det nu "Maxed out!" för ytterligare tydlighet.

Eftersom spelarens val även här, precis som i Settings-vyn, sparas i databasen ger det senare en återkoppling att de önskade uppgraderingarna faktiskt gjordes och implementerades. Dock först när spelaren börjar spela igen, vilket innebär att återkopplingen i detta fall är relativt långsam och kan uppfattas som mindre bra mappning. För att underlätta för användaren skulle en förklarande text kunna implementeras under varje uppgraderingsalternativ.

Beroende på skärmstorlek är det möjligt att användaren inte ser alla uppgraderingar. Därför är "scrollbaren" till höger alltid synlig, som en signifier att det finns fler val längre ner på skärmen. Köpet av uppgraderingarna fungerar konsekvent för alla olika typer av uppgraderingar. Vad de olika uppgraderingarna gör märks dock inte innan spelaren testat att spela igen. Användartesterna visar dock att majoriteten av användarna märkte skillnad direkt efter en uppgradering, som då också ofta förhöjde spelupplevelsen.

Reset-knappen nere i högra hörnet nollställer spelarens tidigare progression, alltså nollställs användarens **cash** och alla uppgraderingar sätts till nivå ett. Trycker användaren på **reset** får denne upp en **reset**-overlay. I **reset**-overlayen står **yes** till vänster och **no** till höger, eftersom västerlänningar läser från vänster till höger och därmed oftast ser yes först, vilket känns naturligt. Reset är en tillfällig funktion som är inlagd för att underlätta vid användartester och demonstrationer och kommer inte att vara tillgänglig i det slutgiltiga spelet.

4.5 Game

I Game-vyn (se *Bild 7*) presenteras användaren med ett rymdskepp som flyger över en rullande bakgrund. En tydlig textruta på övre delen av skärmen med texten "Touch anywhere to start" ger användaren information om vad som krävs för att påbörja spelet, och spelet startar först när spelaren rör vid skärmen. Då märker även spelaren hur skeppet styrs då det åker dit fingret pekar vilket innebär klar återkoppling och affordans. Skeppet skjuter sedan skott automatiskt och fiendeskepp kommer in på skärmen med slumpmässigt genererade mellanrum. Även fiendeskeppen skjuter skott automatiskt.

Game-vyn består utöver tidigare nämnda element av en **paus**-knapp



Bild 7: Game-vyn innan spelstart

uppe till vänster, en indikator på kvarvarande **hälsa** som följer spelaren, en indikator på inspelade **pengar** samt en **missil**-knapp nere till vänster på skärmen.

Paus-knappen pausar omedelbart spelet vid beröring och visar en overlay med tre nya knappar: **resume**, **restart** och **quit** (se Bild 8). Knapparna är centrerade runt mitten av skärmen och har alla samma design. Funktionaliteten i pausmenyn är vanligt förekommande inom spelvärlden och användartesterna visade på att alla användare förstod funktionaliteten vilket visade på god mappning och synlighet. Majoriteten av användarna tryckte instinktivt på paus när de skulle avsluta spelet eller behövde en paus, exempelvis när de behövde tänka för att svara på en fråga.

Vidareutveckling av pausmenyn skulle vara att implementera **settings**-alternativ och visa progressionen av spelet i menyn. I spelet får användaren omedelbar återkoppling när ett skepp går sönder genom att det uppstår en explosion och ett effektljud spelas upp (se Bild 9). Detta gäller även när spelaren själv dör. Då visas även en **game over**-overlay med ungefär samma funktionalitet som **paus**-overlay förutom att "Game Over" står stort och tydligt högst upp samt att spelaren inte kan återuppta spelet. Ytterligare saker gruppen tänkt implementera i denna overlay är att visa hur det gick i spelomgången gällande intjänade pengar och hur länge spelaren överlevt. En direktlänk till **upgrades** kan även vara användbart för att uppmuntra spelaren att använda sina inspelade pengar.

För en roligare spelupplevelse implementerades efter första användartestet en **missil**-knapp som skjuter iväg en missil som gör mer skada än de automatiska skotten. För att skapa balans i spelet måste spelaren vänta några sekunder för att få avfira missilen på nytt. Ursprungligen var tanken att denna korta tid skulle visas som en nedräkning och börja ticka direkt vid spelets start. Användartest 2 visade att detta enbart skapade förvirring hos användaren kring vad som sker när tiden blir noll. Detta tog onödig fokus från spelupplevelsen och beslut togs att låta spelaren ha möjlighet att avfira sin missil från början. För att vara säkra på att nedräkningen inte skulle skapa förvirring byttes denna ut mot en cirkelanimation kring knappen. Direkt efter knapptrycket försvinner cirkeln och en linje börjar växa fram med en cirkelrörelse i det område där cirkeln nyss var (se Bild 10). När linjen återigen blir en cirkeln återfår knappen dess färg vilket är en tydlig signifier att knappen också återfått dess



Bild 8: Paus-overlay



Bild 9: Game-vyn med explosion av fiende

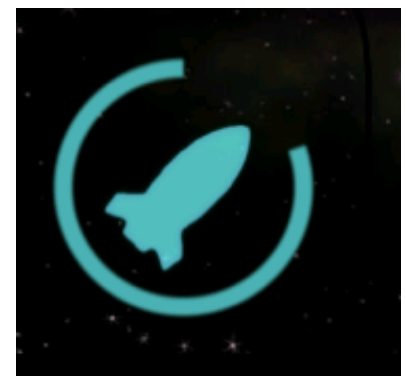


Bild 10: Laddande missil-knapp

funktionalitet. Detta ger god affordans och omedelbar återkoppling samt är ett av alla sätt som uppgraderingarnas effekt visar sig på, då tiden för att åter kunna använda knappen beror på vilken uppgraderingsnivå spelaren befinner sig på.

Missil-knappen är taktiskt placerad för att inte vara i vägen men heller inte oåtkomlig för spelaren. God mappning gör att spelaren enkelt och utan att behöva tänka kan använda knappen. Eftersom Game-vyn i övrigt är fri från knappar samt att Androids egna bakåtknapp har inaktiverats finns ingen chans att användaren av misstag skulle göra något icke-önskat.

Något som projektgruppen tänkt lägga till är omedelbar återkoppling när skeppen blir träffade, även när de inte dör, exempelvis genom att de blinkar till kort. Även snyggare design på skott och "cash" samt att spelet blir svårare med tiden är saker projektgruppen har som vision att implementera.

5. Diskussion

Utifrån ovan redovisat material har stora delar av de projektmål som ursprungligen sattes genomförts. Under hela skapandet av gränssnittet har projektgruppen haft fokus på användbarheten. Det är generellt svårt att objektivt utvärdera sitt gränssnitt på egen hand. Därför utfördes användartester vid två separata tillfällen under projektets gång.

Det första användartestet visade att stora delar av applikationens användargränssnitt var lättförståeligt och intuitivt. Förhoppningen att minst 70 % av alla användare direkt skulle starta spelet uppfylldes. Den simplistiska designen i Main menu-vyn bjuder in användaren att direkt börja spela. Eftersom det förutspåddes att detta skulle vara den erfarna spelarens första beslut var detta ett logiskt och lämpligt implementationsbeslut.

Däremot visade sig somliga funktionaliteter kräva förbättring. Exempelvis implementerades Tutorial-vyn för att förenkla för de osäkra användarna. Detta var även ett mål som fanns från början men där själva implementationen var ett dilemma. I de fall där spelaren tillhörde målgruppen upplevdes spelets kontroller vara intuitiva, denne behövde inte fundera över hur skeppet skulle styras. Det skedde på ett spontant och naturligt sätt, mycket tack vare rådande konventioner som finns för den här sortens spelgenre.

Då gruppen kände viss tvekan om hur skeppets avfyrningsmekanism skulle vara gav testerna dessutom ett tydligt svar i att gruppen borde implementera både automatiska skott som gjorde liten skada samt ett kraftfullare skott som behövde avfyras.

Eftersom några användare påpekade att det inte var uppenbart hur mycket pengar spelaren hade i Upgrades-vyn insåg gruppen att **cash** på något sätt måste sticka ut bättre. Således implementerades en lysande bakgrund för just **cash** och avståndet till de andra knapparna ökades.

Ett annat utsatt mål projektgruppen hade var att spelet skulle gå att spela var som helst. Detta nåddes genom att välja bort att behöva ha ständig internetuppkoppling. Som applikationen är just nu behövs internet endast när den ska installeras eller uppdateras.

Andra mål som nåddes var att alla knappar skulle vara väl synliga och klickbara samt att ha någon form av progression i form av uppgraderingar. Detta blir extra tydligt tack vare implementationen och användningen utav databasen. Uppgraderingar gjorda på skeppet, total mängd pengar och övriga användarinställningar är alla saker som sparas på databasen.

De mål som inte riktigt nåddes var att applikationen skulle fungera lika bra på alla Android-enheter och att ha minst en färdig bana. För att en applikation ska fungera på olika skärmstorlekar med varierande upplösning och bildförhållande krävs flera uppsättningar av alla grafiska element (bilder i olika upplösningar). Det skulle vara fullt möjligt att göra grafiken för olika upplösningar samt att skriva om koden så spelet blir brukbart på alla typer av enheter men i nuläget är designen specificerad till storleken för Nexus 7-plattorna.

Under projektets gång har visionen att ha flera olika banor ändrats till att ha ett spel som går ut på att överleva så länge som möjligt och att spelet istället gradvis blir svårare. Därför har inget naturligt slut på spelet implementerats utan tanken är istället att i framtiden implementera svårare fiender och kanske andra typer av uppgraderingar.

6. Slutsats

Gruppen är nöjd med resultatet även fast inte alla utsatta mål nåddes. I början av projektet kändes det som gruppen tagit sig vatten över huvudet eftersom det var mycket att sätta sig in i kring spelprogrammeringen innan några resultat visade sig. Nu när slutet av projektet närmar sig känner dock gruppen att spelet kommit en bra bit på vägen och att gruppen nu har kunskapen för att kunna fortsätta utveckla spelet.

I stort tycker projektgruppen att tidsplaneringen hölls bra och alla medlemmar har jobbat på och varit med om att föra projektet framåt. Gruppen gick in med inställningen att alla skulle lägga ner så mycket tid som möjligt. Samtliga medlemmar kongruerar att detta även syns nu när stora delar av applikationen är färdigställda.

Användartest 1 (27/11)

Användartest med öppna svar. Deltagarna fick testa applikationen utan instruktioner och uppmuntrades att tänka högt medan projektgruppen tog anteckningar på vad de gjorde och vad de sa.

Versionen deltagarna fick testa hade ett skepp som följde fingrets rörelser och fiender som genererades (se *Bild 1*). Spelaren kunde inte dö, ta skada eller skjuta. Den rullande bakgrunden, musik, **paus**-overlay, **game over**-overlay och Tutorial-vyn var inte implementerade.

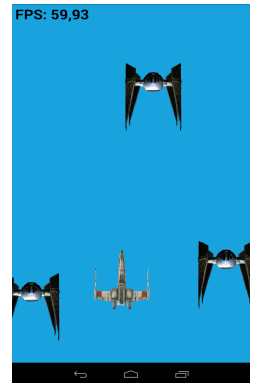


Bild 1: Tidig version av Game-vyn

Antal deltagare: 10 st

Del 1: Förstår deltagaren det nuvarande gränssnittet och hur de ska använda det?

Del 2: Hur tycker deltagaren applikationen bör utvecklas?

Frågor till Del 2:

- Om du ska skjuta/pausa, var bör den knappen finnas?
- Vill du att skeppet ska skjuta hela tiden eller ska man behöva trycka för skott/missil?
- Vad skulle du vilja ha med i **paus**-menyn?
- Saknar du någon information/funktion?
- Förstod du hur du skulle göra i Upgrades/Settings?

Kommentarer från Del 1:

Main Menu

- De flesta (8/10) går direkt på **start**-knappen.
- "Jag gillar att man inte behöver tänka. Bra med större **start**-knapp. Man lär sig spelet allt eftersom."

Game

- Inte populärt att skeppet har olika hastigheter beroende på om man trycker eller drar. Bör vara konstant!
- När de sitter med en platta så använder de två händer istället för en, vilket innebär att det är lättare att trycka än svepa på skärmen.
- Vissa störde sig på att skeppet "vickade".

Upgrades/Settings

- En deltagare tyckte "Köp"-knappen såg ut som en "fäll upp"-knapp men resten förstod direkt vad allt betydde, i båda menyerna.
- En deltagare ville lägga till en "flip screen-orientation" i Settings-vyn så man kunde välja mellan höger- eller vänsterhänt.

Paus

Förslag på saker att ha med:

- Settings med ljud
- Remaining Health
- Total Cash
- Progress
- Resume
- Restart
- Quit
- Tutorial

Undersökning av åsikter:

Hur vill du skjuta?

- Genom att trycka på en knapp: 3
- Endast automatiska skott: 2
- Både och: 3

Sammanfattning av kommentarer:

De flesta var positiva till mer grundläggande automatiska skott men att man skjuter missiler/uppgraderingar själv. Nästan alla tycker skjutknappen bör vara nere till vänster.

Var bör pausknappen sitta?

- Uppe till vänster: 4
- Uppe till höger: 2
- Nere till höger: 1

Allmänna tankar:

- Alla undrar hur man skjuter.
- Alla vill få poäng och mer cash.
- Vissa vill ha någon Tutorial första gången man startar spelet, andra tycker man lär sig allt eftersom.
- En ville spontant styra med gyro, en annan hade föredragit en "joystick", resten trivdes med att man både kunde trycka och dra.
- Många tryckte spontant på skeppet för att den skulle skjuta, men de flesta resonerade fram att det berodde på att den inte började skjuta automatisk samt att ingen knapp var synlig. Annars hade de föredragit de alternativen.
- De flesta tänker sig Credits på huvudmenyns frågetecken och att förklaringarna till spelet kommer i Game-vyn, antingen som Tutorial eller i pausmenyn. Vyn bör dock gå att stänga genom att trycka på frågetecknet igen.
- Några efterfrågade en lista med Highscore.
- Någon föreslog att ha "3, 2, 1, GO!" innan spelet startar/återupptas.

Användartest 2 (5/12)

Användartest med öppna svar. Deltagarna fick testa applikationen utan instruktioner och uppmuntrades att tänka högt medan projektgruppen tog anteckningar på vad de gjorde och vad de sa.

Versionen deltagarna fick testa var Test release 0.2. Saker som inte var med: Musik, Game over-overlay, missil-nedräkningen började vid start, Tutorial.

Antal deltagare: 8 st

Allmänna åsikter:

- **Cash** borde stå ut mer från de andra texterna i Upgrades. Siffrorna borde synas bättre också.
- Bra att **start** är större än de andra knapparna i huvudmenyn. (7/8 trycker direkt på **start**)
- Vill få feedback när skeppet (eller fienderna) blir träffat. Ex blinkar till.
- Oklart vad nedräkningen är. Många tror att det är så länge man ska överleva för att klara banan.
- Highscore borde vara hur lång tid man överlever.
- Bugg med inspelade pengar i Game-vyn: Om en nolla läggs till så försvinner den ur bild.
- Bugg: Skeppet dör av ett skott och applikationen kraschar när Shield är på jämna nivåer.