

# Användbarhetsutvärdering av en logistik-simulator

Usability evaluation of a logistical-simulator



LUNDS TEKNISKA  
HÖGSKOLA  
Lunds universitet

Examensarbete:  
Renas Reda

© Copyright Renas Reda

LTH Ingenjörshögskolan vid Campus Helsingborg  
Lunds universitet  
Box 882  
251 08 Helsingborg

LTH School of Engineering at Campus Helsingborg  
Lund University  
Box 882  
SE-251 08 Helsingborg  
Sweden

Tryckt i Sverige  
Media-Tryck  
Biblioteksdirektionen  
Lunds Universitet  
Lund 2003

## **Abstract**

The purpose of this thesis is to explore what rules are to be followed when developing software and to make a usability test of a prototype, that a project group CELIT has developed for a large Swedish furniture company X. This usability test can then be used by the project group to make the prototype better. The prototype is a logistical-simulator.

I started of with studying literature to learn more about usability testing. After I decided which usability method I was going to use, I wrote down all the questions I thought would be interesting to ask about. When I had deleted a lot of questions that where either not relevant or to similar I was left with twenty three questions.

The usability test was conducted in a way that I first explained for the ten persons that where considered to be representative of the evaluation what the game was about and why it had been developed. After that I asked them to answer a questionnaire while and after they had played a game. If I didn't think their answer was thorough enough, I reformulated the question orally and took notes of their answer.

The conclusion of the usability test is that the users thought the idea behind the game was good but that the game was too difficult to play.

## **Sammanfattning**

Syftet med detta arbete är att utforska vilka regler som gäller vid utvecklandet av en mjukvara och att göra användbarhetstest av en prototyp som är tillverkad av en projektgrupp CELIT för ett stort svenskt möbelföretag X. Detta användbarhetstest kan sedan användas som underlag av projektgruppen för att förbättra prototypen. Prototypen är ett spel som ska visa X's anställda hur företagets varuflöde ser ut.

Jag började med att studera litteratur för att lära mig mer om användbarhetsutvärdering. Efter att jag hade bestämt mig för vilken utvärderingsmetod jag skulle använda började jag skriva ner alla frågor jag kunde tänka mig vara intressanta för studien. Efter att jag strykt många frågor som antingen varit irrelevanta eller för lika återstod tillslut 23 frågor.

Användbarhetstestet gick till så att jag först förklarade för tio personer som ansågs represantiva för utvärderingen vad spelet handlade om och varför det har blivit utvecklat. Sen fick de svara på enkätfrågorna medan och efter att de spelat en omgång. Om jag ansåg att deras svar inte var tydligt nog omformulerade jag frågan muntligt och antecknade deras svar.

Slutsatsen av användbarhetstestet var att användarna tyckte att idén bakom spelet var bra men att spelet var för svårt att bemästra.

## **Förord**

Först och främst skulle jag vilja tacka min examinator, Joakim Eriksson på institutionen för Design Vetenskap på Lunds Tekniska Högskola, för hans stöd och hjälp under arbetets gång.

Jag skulle vilja tacka Anders Lundblad för hans vägledning och för att han svarat på mina många frågor.

Jag skulle även vilja tacka de trevliga personerna som ställt upp i intervjuerna. Detta arbete skulle inte kunna ha skrivits utan den informationen som de gett mig.



# Innehållsförteckning

<b>1</b>	<b>INTRODUKTION</b> .....	<b>7</b>
<b>2</b>	<b>SYFTE OCH BEGRÄNSNINGAR</b> .....	<b>8</b>
2.1	SYFTE.....	8
2.2	BEGRENSNINGAR.....	8
<b>3</b>	<b>ARBETSGÅNG</b> .....	<b>9</b>
<b>4</b>	<b>MÄNNISKA-DATORINTERAKTION</b> .....	<b>10</b>
4.1	MJUKVARA.....	10
4.2	ANVÄNDARGRÄNSSNITT.....	10
4.3	ANVÄNDBARHET.....	11
4.4	DESIGNRIKTLINJER.....	13
<b>5</b>	<b>ANVÄNDBARHETSTESTER</b> .....	<b>15</b>
5.1	ATT VÄLJA TESTMETOD.....	15
5.2	FORMATIVA OCH SUMMATIVA UTVÄRDERINGAR.....	15
5.3	EXEMPEL PÅ METODER FÖR ANVÄNDBARHETSMÄTNING.....	15
5.3.1	<i>Heuristic Evaluation</i> .....	15
5.3.2	<i>Cognitive walkthrough</i> .....	16
5.3.3	<i>Enkäter</i> .....	16
5.3.4	<i>Checklistor</i> .....	16
<b>6</b>	<b>MÖTESANTECKNINGAR</b> .....	<b>18</b>
<b>7</b>	<b>METOD VID UTVÄRDERING</b> .....	<b>20</b>
7.1	TESTGRUPP.....	20
7.2	ANVÄNDARENKÄT.....	20
7.3	ANVÄNDBARHETSTESTET.....	20
<b>8</b>	<b>RESULTAT AV ANVÄNDARENKÄTEN</b> .....	<b>21</b>
8.1	DATORKUNSKAPER.....	21
8.2	SPELINSTRUKTIONER.....	22
8.3	FÖRVÄNTNINGAR EFTER SPELINSTRUKTIONERNA.....	23
8.4	SPELETS UTSEENDE.....	24
8.5	SPELETS KOMPLICERINGSGRAD.....	25
8.6	ALLMÄNT OM SPELET.....	26
8.7	LOGISTIK FÖRSTÅELEN.....	27
8.8	OBJEKTENS PLACERING.....	28
8.9	OBJEKTENS FÄRGER.....	29
<b>9</b>	<b>SLUTSATS OCH DISKUSSION</b> .....	<b>30</b>
<b>10</b>	<b>REFERENSLISTA</b> .....	<b>32</b>
	<b>BILAGA 1</b> .....	<b>33</b>

## 1 Introduktion

Inom mjukvaruvärlden ökar hela tiden intresset för användbarhet samtidigt som förståelsen av att mjukvara måste vara lätt att använda också ökar kontinuerligt. Hur ofta har man inte hört "Vad dålig jag är på att använda datorer" eller "Jag måste ha tryckt på fel knapp"? Generellt brukar användarna skylla på sig själva när de gör fel eller inte förstår hur dem borde göra. För ett mjukvaruföretag är detta väldigt allvarligt. Tiden då datorer endast blev använda av ett fåtal människor med väldigt hög kompetens är sedan långt gången, datorer används nu av miljontals människor. Mjukvaruföretagen måste fråga sig själva varför folk trycker på fel knappar och hur det kan förhindras. Det är nuförtiden oftast inte personen det är fel på när något går fel utan själva mjukvaran.

Jag finner användbarhet mycket intressant eftersom det rör kommunikationen mellan männikan och datorer. Det är roligt att observera vilka val människor gör när de vill utföra specifika kommandon. Jag anser det vara viktigt för företag att använda sig av fler människor när de testar sina prototyper för att kunna ändra och förbättra dem.

Genom Lunds Tekniska Högskola fick jag en möjlighet att delta i ett projekt där en logistik-simulator skulle bli utvecklad till ett stort svenskt möbelföretag som kommer att betecknas med X i detta arbetet. Detta spel ska simulera för deras anställda hur X varuflöde fungerar. Projektet drevs av ett antal aktörer där även Institutionen för Designvetenskaper på Lunds Tekniska Högskola var med. Jag fick i uppdrag att göra användningstest av prototypen som kan användas av projektgruppen CELIT för att förbättra den.

## **2 Syfte och begränsningar**

### **2.1 Syfte**

Syftet med arbetet är följande:

- Utforska vilka regler som bör följas vid utvecklandet av en mjukvara.
- Att göra användbarhetstest som kan användas som underlag hos CELIT för att förbättra prototypen.

### **2.2 Begränsningar**

Illustration och beskrivning av prototypen är utelämnat på grund av sekretess.

### **3 Arbetsgång**

För att lära mig mer om användbarhetsutvärdering har jag använt mig av flera metoder, dels litteraturstudier som vägledning för att få en större generell kunskap om hur en mjukvara ska skrivas och vad utvärdering är. Internet har jag använt när jag haft småfrågor som litteraturen ej kunnat besvara mig. Som utvärderingsmetod har jag använt mig av intervjuer och enkäter. Utvärderingen har gått till så att jag bett 10 personer testa prototypen genom att spela en omgång och sedan svara på enkätfrågor och delta i intervjuer. Under spelomgångarna har jag antecknat misstag som jag observerat genom videobandsinspelning och observationsanteckningar. Därefter har jag sammanställt resultaten av användbarhetstesten (se kap 8).

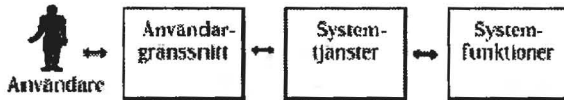
Relevant information om hur prototypen har tagits fram är nedskrivna med förklaringar om vad som diskuterats och beslutats i de olika mötena samt var mötena har hållits (se kap 6, tabell 1).

Avslutningsvis har jag fört en diskussion och skrivit ner mina tankar om testernas gång och resultat (se kap 9).

## 4 Människa-Datorinteraktion

### 4.1 Mjukvara

Mjukvara är en produkt som innefattar ett programsystem. Support och användardokumentation såsom manualer ingår också i begreppet.



Figur 1: Mjukvarans uppbyggnad

Löwgren (1993) väljer att se mjukvara som en arkitektur bestående av fyra nivåer; användare, användargränssnitt, systemtjänster och systemfunktioner (se Figur 1).

Användarens kontakt med systemet sker genom användargränssnittet.

Användargränssnittet visar användaren de tjänster som systemet har att erbjuda, exempelvis tjänsten 'spara' erbjuder användaren att spara sitt arbete på lämpligt utrymme. Systemtjänsterna erbjuder användaren systemfunktionerna, det vill säga vad systemet kan göra. Systemtjänsterna ska vara anpassade till användarens behov samt stödja användaren i sin uppgift. Samma systemtjänst kan vara kopplad till flera systemfunktioner (Löwgren, 1993). Exempelvis kan tjänsten 'spara' innebära att rätt dokument öppnas, kollar ledigt utrymme, skapar ett utrymme i minnet, skriva det som finns i minnet till disk etc.

### 4.2 Användargränssnitt

Ett användargränssnitt (fortsättningsvis enbart benämnt gränssnitt i de fall där ingen risk för feltolkning kan föreligga) är det som först möter användaren vid kontakt med exempelvis en datorapplikation. Det är genom gränssnittet som användaren interagerar med programmets underliggande funktioner och kan ta del av programmets tjänster (Löwgren, 1993). Utformningen är således betydelsefull för hur väl användaren kan utföra sina uppgifter. Exempel på en del av ett gränssnitt kan vara knappar som erbjuder användaren att spara ett dokument, eller en inmatningsruta som erbjuder användaren att skriva in ett sökord för att sedan utföra en sökning i en databas. Ett gränssnitt ska vara enkelt att använda eftersom gränssnittet kommer att påverka, och i hög grad avgöra, hur användarens attityd mot systemet faller sig samt om systemet anses dugligt eller ej för att lösa användarens uppgift. Det är av underordnad betydelse hur "bra" systemet är om inte stor vikt läggs vid gränssnittets utformning, eftersom användarna i motsatt fall inte kan utnyttja systemet optimalt (Löwgren, 1993). Ett gränssnitt kan inte ses som

något självständigt, skilt från mjukvaran eller systemet i sin helhet för det är just genom gränssnittet som användaren får kontakt med systemet (Löwgren, 1993). Ibland används begreppet grafiskt användargränssnitt och innefattar då den del av användargränssnittet som representeras grafiskt, vanligtvis på skärm, i form av färg, bilder, former och textdispositioner etc.

### 4.3 Användbarhet

Löwgren (1993) ser användbarhet som summan av relevans, effektivitet, attityd och lärbarhet. Löwgren väljer att kalla definitionen för REAL-modellen. Nedan följer en presentation av de fyra elementen i begreppet användbarhet.

**Relevans**, hur relevant systemet är för användaren (Löwgren, 1993). Stödjer inte systemet användarens arbete kan inte heller systemet ses som relevant för användaren. Ett systems gränssnitt ska erbjuda användaren de tjänster som är relevanta för att utföra användarens arbete.

**Effektivitet**, hur effektivt eller lätt det är för användaren att utföra önskad uppgift samt hur lätt det är att upptäcka fel för att sedan kunna åtgärda dem (Löwgren, 1993). Ett effektivt arbete kräver att systemets gränssnitt stödjer en snabb och korrekt informationsbearbetning. Ett användbart gränssnitt minskar således användarens mentala belastning, användaren behöver inte lägga extra energi och tankeverksamhet på att hantera gränssnittet utan kan fokusera på själva uppgiften.

**Attityd**, användarens subjektiva attityd mot systemet (Löwgren, 1993).

Användarens attityd är huruvida användaren tror att systemet kan tillgodose användarens behov i arbetet. Om användarna har en positiv attityd mot systemet ökar detta användbarheten enligt Löwgrems definition. Ett sätt som kan bidra till en positiv attityd hos användarna är att låta användarna bli delaktiga i utvecklingen och utformningen av produkten.

**Lärbarhet**, hur lätt det är att lära sig använda systemet första gången samt hur lätt det är att sedan komma ihåg vad man lärt sig (Löwgren, 1993). Att lära sig använda ett system kräver till en början stor uppmärksamhet och belastar användaren mentalt. Efter övning kräver hanteringen av systemet allt mindre medveten uppmärksamhet och blir mer och mer automatisk. Att snabbt lära sig använda ett system och sedan komma ihåg vad man har lärt sig ger användaren större möjlighet att fokusera på själva arbetetsuppgifterna än att lära sig hantera systemet. Att investera i lärbarhet lönar sig då system med hög lärbarhet bidrar till att användarna behöver mindre utbildning för att hantera systemet, vilket medför en lägre kostnad för företaget, samt mindre mental belastning för användaren. Ett annat sätt är att utnyttja redan befintliga resurser såsom användarnas erfarenheter. Genom att bygga in användarnas erfarenheter i det nya gränssnittet/systemet behöver inte användarna

lära om i samma utsträckning som för ett helt nytt arbetssätt vilket minskar användarens medvetna informations-bearbetning. Vikten av att utforma system och användargränssnitt med hög användbarhet får mer och mer uppmärksamhet. Genom att investera i användbarhet sparas både ekonomiska och personella resurser i form av ett effektivare arbete.

Med effektivt menas här ett snabbt och korrekt informationsinhämtande. Användbarhet specificeras alltid i relation till en specifik användargrupp, vilket innebär att det finns ingen universellt bästa gränssnittsutförande. Varje användargrupp har olika bakgrund, arbetsuppgifter, krav etc. En klar fördel med REAL är att samtliga element går att mäta. Måtten kan exempelvis användas i användartester där mått på användbarhet eftersöks. Löwgren & Stolterman (1998) tar upp fyra mått för att mäta just användbarhet som kommer från den experimentella forskningen. Nedan kopplas måtten ihop med Löwgrens fyra element i REAL.

- Relevans kan mätas i hur flexibel systemets utformning är, mätt andel försökspersoner ur en heterogen grupp som lyckades genomföra de uppsatta testuppgifterna.
- Effektivitet kan mätas i hur väl försökspersonerna presterar vid lösande av testuppgifter, mätt i andel lösta uppgifter, hur lång tid det tog, eller hur många fel som begicks.
- Attityd kan mätas i vad försökspersonerna tycker om systemet, mätt i subjektiva skattningar av exempelvis systemets hjälpsamhet och effektivitet.
- Lärbarhet kan mätas i hur lätt systemet är att lära sig, uttryckt i andel lösta uppgifter, prestationstider och antal fel över tiden, hur väl försökspersonerna minns det de lärt sig, hur ofta de måste använda hjälpfunktioner eller fråga om råd.

Den traditionella synen på användbarhet innebär att om rätt funktioner, för användarens uppgifter, finns hos produkten eller systemet så är den användbar. Löwgren och Stolterman (1998) framhåller kontextuell design som en reaktion på den traditionella synen på användbarhet. Kontextuell design, liksom Löwgrens REAL, väger bland annat in användarens subjektiva och personliga upplevelse av gränssnittet. Man menar att utan hänsyn taget till användningssituationen (kontexten) kan inte ett gränssnitt uppfylla kravet på användbarhet. Användarens första intryck av gränssnittet kan vara direkt avgörande om produkten kommer att användas. Om användaren får en negativ reaktion kan det resultera i att användaren inte litar på att produkten kan utföra den önskvärda uppgiften. Likaså kan ett gränssnitt som skapar en omedelbar positiv inställning hos användaren vara

avgörande för användarens fortsatta inställning till, och användning av systemet (Löwgren & Stolterman, 1998).

#### 4.4 Designriktlinjer

Hur kan man då designa för att uppnå användbarhet i ett system? Donald Norman (1990) ger ett antal riktlinjer som hjälper till att skapa användbara system. Först bör man bestämma för vem man designar systemet eftersom: "det inte finns någonting som en medelperson". Det är viktigt att inse att personer med olika bakgrund, kunskaper och motiv kommer att handla på olika sätt. Det är en stor utmaning att göra en design som passar alla. Först när man bestämt vem det är man ska designa för kan man göra antaganden om vad man kan förvänta sig av användarna och tillämpa riktlinjer för design.

Normans riktlinjer lyder:

1. Använd både kunskap i världen och kunskap i huvudet
2. Förenkla strukturen på uppgifter
3. Gör saker synliga
4. Gör mappningen rätt
5. Utnyttja kraften i begränsningar, både naturliga och artificiella
6. Designa för fel
7. Om inget annat fungerar, standardisera

Den första riktlinjen betyder i stora drag att man inte ska tvinga sina användare att komma ihåg information i huvudet som de inte behöver komma ihåg. Eftersom igenkänning är lättare än hågkomst underlättar det för användare att till exempel få välja mellan olika val i en lista istället för att få skriva in en exakt ordalydelse. Då utnyttjar man kunskap i världen istället för att helt förlita sig på kunskap i huvudet på användaren. Riktlinje nummer två gör gällande att man ska hjälpa användaren genom att göra svåra uppgifter lätta att utföra. Norman tar upp fyra olika sätt att göra detta. Den första är att göra uppgiften liknande (om man ändrar sätt att utföra uppgiften på) men tillhandahålla mentala hjälpmedel. Mentala hjälpmedel kan till exempel vara minnesanteckningar för att avlasta användarens korttidsminne. Det andra sättet är att använda teknologi för att göra saker synliga som annars inte är tillgängliga. Normans egna exempel på detta är instrumentpanelen i en bil som ger information om motorns tillstånd. Som nummer tre föreslår Norman att man ska



automatisera men låta uppgiften vara den samma. Slutligen kan man ändra sättet man utför en uppgift på. Detta kan vara motiverat när ett sätt att utföra en uppgift är onödigt komplext eller när man helt enkelt kommer på ett enklare sätt att utföra en uppgift på. Genom att göra saker synliga kan användaren få en överblick över vad som är möjligt att göra. Synligheten kan också innebära att feedback ges för att visa användaren att deras handlingar ger något resultat. I det här fallet behöver synlighet inte tolkas som visuellt utan kan också innebära klickljud vid musklickningar och säkert även andra former av feedback. Poängen är att förmedla något till användaren. Till att börja med kan en förklaring av termen mappning (egen översättning från eng. mapping) vara på sin plats. Mappning är relationen mellan kontroller, till exempel knapparna på en fjärrkontroll och resultatet av knapptryckningarna. Med naturlig mappning avser Norman att förhållandet är uppenbart eller självklart för en betraktare. Det är ett tillstånd man bör sträva efter enligt Normans fjärde riktlinje.

Som förklaring till riktlinje nummer fem skriver Norman: ”Använd begränsningar så att användaren känner att det bara finns en sak att göra”. Författaren använder det engelska ordet constraint för att benämna de begränsningar eller hinder (både upplevda och verkliga) man kan bygga in i ett system. Genom att hindra användare från att göra fel styr man in dem på rätt väg. Detta är en, i kombination med andra metoder, mycket kraftfull metod att styra användares handlingar. Den sjätte riktlinjen säger att man ska designa för fel. Med detta förklarar Norman att man ska anta att alla fel som kan göras kommer att göras. Genom att försöka förutse olika fel kan man i designen bygga in hjälp för att styra användaren i rätt riktning. Man ska stödja användarna och göra det lätt att ångra vid fel. Man bör också göra det svårt att göra fel som inte går att ångra. Slutligen finns en ofta använd utväg, att standardisera. Genom att alltid använda samma mönster att utföra handlingar på behöver användaren endast lära sig detta mönster en gång. Sedan kan de använda det på andra håll. Standardisering tillåter en ganska godtycklig design, om man bara är konsekvent så lär sig användarna ändå hur de ska handla och kan klara av sina uppgifter.

## 5 Användbarhetstester

Det finns många olika metoder att mäta användbarhet. Olika metoder är dock lämpade för att testa olika delar av användbarhetsbegreppet. Nedan finns en beskrivning av några saker man bör tänka på innan man väljer metod. Efter det följer två olika typer av tester och en kort beskrivning av några olika metoder för användbarhetsutvärdering.

### 5.1 Att välja testmetod

Det finns ett antal aspekter man bör tänka på innan man väljer testmetod. För det första bör man bestämma vad det är som avses mätas. Man kan svara användbarhet på denna fråga men mot bakgrund av ovan nämnda olikheter i definition av begreppet säger inte detta så mycket. Om man däremot avser att mäta på vilka sätt användare kan få/får problem med en produkt eller prototyp har man avsevärt förenklat uppgiften. En metod ger ett specifikt resultat, så man bör välja en metod som ger svar på en frågeställning som motsvarar vad man vill veta. För det andra bör det vägas in vilka resurser man har tillgängliga. En metod som *Heuristic evaluation* bl.a. beskriven i Nielsen (1993) och Wharton et al (1994) kräver tillgång till ett antal användbarhetsexperter men inga användare. Andra metoder kräver tillgång till försökspersoner. Har man inte tillgång till nödvändiga resurser bör man börja leta efter en annan metod som bättre passar tillgängliga medel.

### 5.2 Formativa och summativa utvärderingar

En uppdelning av olika typer av tester är när i tiden de utförs i en utvecklingsprocess. Löwgren (1993) nämner formative och summative evaluation (egen översättning till formativ och summativ utvärdering) som två typer. Formativa utvärderingar är metoder som kan tillämpas under utvecklingen av ett system för att få feedback på olika designuppslag. Detta har som mål att ge information som ska användas i en iterativ designprocess. Summativa utvärderingar används för att mäta hur ett redan existerande system är beskaffat. De formativa utvärderingarna bidrar mest till ökad användbarhet i ett system när de tillämpas många gånger eftersom man kan testa olika designförslag och därefter välja det som verkar vara bäst.

### 5.3 Exempel på metoder för användbarhetsmätning

#### 5.3.1 Heuristic Evaluation

Heuristic evaluation är en metod bl.a. beskriven i Nielsen (1989) och Wharton et al (1994) som ger ett antal riktlinjer som kan tillämpas när ett system utvärderas. De som testar bör ha kunskaper från användbarhetsområdet. Detta för att de då bättre kan förstå konsekvenserna av att bryta mot riktlinjer och kan avgöra när det är befogat eller inte. Det är dock inte en garanti att hitta alla problem i användbarhet bara för att man har en duktig utvärderare. Olika personer hittar olika problem. Det är därför lämpligt, enligt Nielsen, att mellan 3 och 5 utvärderare arbetar på samma gränssnitt för att kompensera för skillnader i fynd. Några exempel på riktlinjer är: enkel och naturlig dialog, minimera användarens minnesbelastning, var konsekvent och använd precisa och konstruktiva felmeddelanden. Genom att hitta problem kan de senare åtgärdas vid iterativ design.

### **5.3.2 Cognitive walkthrough**

Cognitive walkthrough är en metod som fokuserar på att utreda hur lätt ett system är att lära sig att använda. Metoden finns beskriven i Wharton et al (1994) och genomförs genom att designförslag i form av t.ex. en prototyp utvärderas av andra än utvecklarna själva. Utvärderarna bör ha kunskaper om användbarhet för att uppnå ett bra resultat. I stora drag genomförs en utvärdering genom att några uppgifter formuleras som ska lösas med hjälp av prototypen och utvärderarna tänker högt hur de tror att en användare skulle resonera när de löser uppgifterna. Om designen på gränssnittet inte på ett bra sätt stödjer uppgifterna kan man identifiera vart problemen finns och genom iterativ design åtgärda dem. Om man väljer att använda denna metod bör man komma ihåg att den endast syftar till att mäta hur lätt ett gränssnitt är att lära sig, inget mer.

### **5.3.3 Enkäter**

Enkäter är enligt Faulkner (2000) en bra metod att samla information på. Med dem kan man få mycket information som kan vara användbart. Nackdelen med enkäter är att de tar väldigt lång tid att producera. Enkäter kan antingen testaren själv få fylla i eller få hjälp av en som assisterar honom. Fördelen med att testaren inte själv fyller i enkäten är att man kan få längre svar och större kontroll på hur frågorna uppfattas. Nackdelen är att allting kommer ta mycket längre tid.

### **5.3.4 Checklistor**

Denna metod påminner om Heuristic evaluation men istället för övergripande riktlinjer använder man mer detaljerade beskrivningar av vad man ska leta efter. Löwgren (1993) skriver att det finns många generella listor men att det i vissa projekt utarbetas listor tidigt i utvecklingsprocessen för att senare kunna följa upp

dem. När man utvärderar med checklistor tillämpas en serie regler för olika delar av ett gränssnitt. Detta är egentligen inte en metod i sig utan mer en generell taktik för att identifiera användbarhetsproblem. Om checklistan som används, inte på ett bra sätt speglar vad man är ute efter kommer resultatet att bli därefter.

## 6 Mötesanteckningar

Tabell 1. Visar vad som diskuterades och beslutades under de olika mötena.

Möte	Datum	Plats	Diskussion om
1	11/9 2002	Campus i Helsingborg	spelets innebörd, hur det kan komma att se ut och i vilken plattform den skulle kunna göras.
2	30/10 2002	Campus i Helsingborg	vilket symbolspråk som bör användas och vem som är spelaren i spelet.
3	6/11 2002	Campus i Helsingborg	utformning av spelets utseende med studenter på Campus i Helsingborg.
4	8/11 2002	Campus i Helsingborg	olika idéer till hur spelet och scenarierna ska se ut, vilket språk spelet ska bli skrivet i (lingo eller java) och identifiering av de olika delarna i spelet (trading, agent, dc etc).
5	13/11 2002	C-its i Helsingborg	idéer och scenarier fortsätter i detta mötet medan java blir bestämt som skrivspråket.
6	22/11 2002	C-its i Helsingborg	utseendet är i stort sett avslutat förutom viss finputsning av smådetaljer
7	7/2 2003	C-its i Helsingborg	hur lång tid spelet ska ta att spela, hur graferna ska se ut och beslut om att jag ska testa prototypen.

Möte 1 och 2 hölls för att CELIT skulle informeras om vad meningen med projektet var. Här pratades det om vad spelet skulle lära ut till användarna och gruppen spekulerade i olika idéer om hur spelet skulle kunna se ut.

Möte 3 hölls med studenter på Campus i Helsingborg. Projektledaren ville få lite idéer av studenterna om hur de tyckte att spelet skulle se ut. Först förklarade projektledaren vad spelet skulle handla om och sedan startades en diskussion om hur spelet borde se ut.

Under möte 4 kom två varianter fram som möjlig spelplan. Den första var att spelaren med en gång skulle se hur varuflödet såg ut medan i den andra var spelaren själv tvungen att bygga upp varuflödet. Två olika spelidéer kom också fram, att spelaren antingen var tvungen att göra insatser i alla varuflödets

huvuddelar eller själv fick bestämma vad den ville ändra. Den enda idén till hur spelet skulle startas var att en missnöjd kund saknar en viss vara på hyllan. Fler idéer eftersöktes.

Under möte 5 bestämdes vilka av de olika idéerna i möte 4 som skulle användas. Det beslutades att spelaren genast skulle se hur varuflödet ser ut. När användaren klickade på ett visst objekt skulle den kunna ändra vilka parametrar som helst. Spelaren skulle få meddelanden från varuflödet som angav vad som är dåligt och behöver förbättras. Poäng och tidsåtgång skulle också visas.

När möte 6 inleddes var utseendet i stort sett klart. Smådetaljer ändrades och ett par nya idéer kom till som tex att lastbilarna skulle röra sig mellan ändpunkterna i den frekvens som styrs via spelet och att besökarna skulle röra sig mot varuhuset beroende på hur bra försäljningen gick.

Möte 7 var det sista mötet som arrangerades innan prototypen skulle testas. Det diskuterades då om hur lång tid det skulle ta att spela en omgång och hur graferna skulle se ut. Det beslutades att jag skulle göra ett användbarhetstest av prototypen.

## **7 Metod vid utvärdering**

### **7.1 Testgrupp**

Jag valde ut tio personer som representativa för utvärderingen då de var av olika åldrar och hade varierande kunskaper inom spel och IT. Eftersom spelet är utvecklat för X's väldigt många anställda antog jag att deras kunskaper inom spel, IT och själva företaget varierar kraftigt.

### **7.2 Användarenkät**

Utvärderingen av gränssnittets användarvänlighet baserades på information i Kap 4 speciellt REAL-modellen (Löwgren, 1993).

För att konstruera användarenkäten började jag med att skriva ner alla frågor som jag ansåg vara viktiga, enligt information i Kap 4. Därefter strök jag frågor beroende på hur intressanta de var för studien samt frågor som var för lika varandra. Efter att ha diskuterat med min handledare valde jag att dela upp enkäten i 3 delar: Briefing, Test och De-briefing. Briefing var den första delen som testarna skulle svara på, frågor som återfinns här är bla namn och utbildning. I testdelen ska de inledande spelinstruktionerna bedömas. Sist i denna delen var att de ska tänka högt när de spelar en omgång. Sista delen av enkäten är De-briefing. Denna delen skulle testarna svara på efter att de spelat klart. Frågor som återfinns här är vad de tycker om spelets utseende och vad de allmänt tycker om spelet.

Enkäten som delades ut kan ses i Bilaga 1.

### **7.3 Användbarhetstestet**

Användbarhetstestet som tog mellan 30-40 minuter gick till så att jag först förklarade vad spelet gick ut på och varför det har blivit utvecklat. Därefter fick testpersonerna svara på ett par frågor under Briefing delen av enkäten. Sedan fick de gå igenom spelets bruksanvisningar och svara på 4 frågor under testing om vad de trodde och tyckte inledningsvis om spelet. Efter sista frågan under testing stod det att de ska tänka högt under spelets gång. Jag poängterade för dem ännu en gång hur viktig detta var för undersökningen. Jag både antecknade och filmade dem medan de spelade.

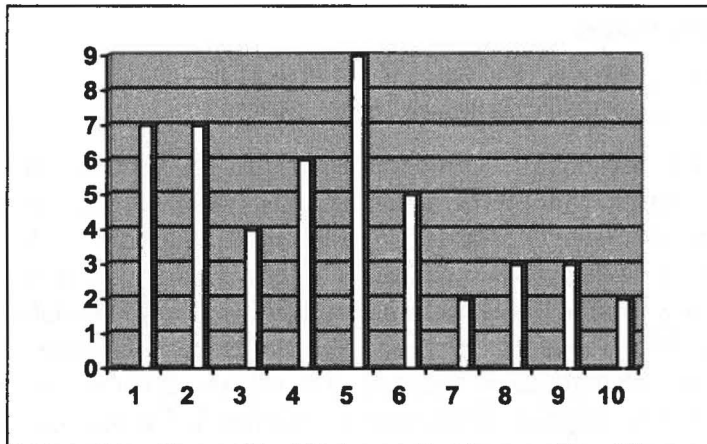
När de spelat klart fick de svara på de 14 sista frågorna under De-briefing. Då jag ansåg att jag inte hade fått tillräckligt bra svar på särskilda enkätfrågor, omformulerade jag de muntligt så de bättre skulle förstå vad jag var ute efter och sedan antecknade jag deras svar.

## 8 Resultat av användarenkäten

I detta kapitel har jag valt att visa resultat till frågor som jag anser vara viktiga.

### 8.1 Datorkunskaper

Datorkunskaperna hos de intervjuade varierar men det fanns dock ingen som inte hade använt datorer innan.

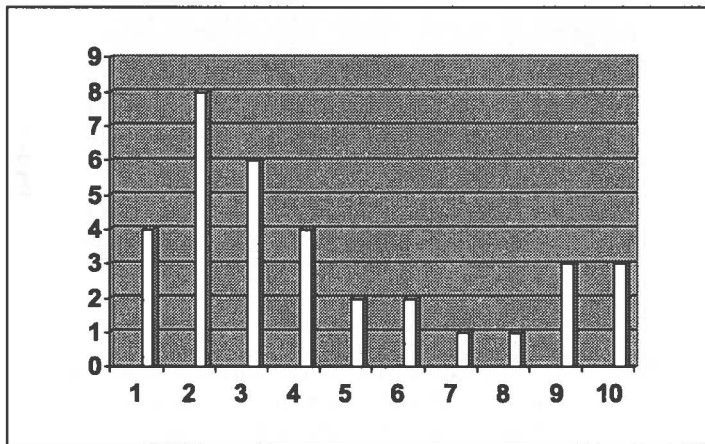


**Diagram 1:** *Fördelningen på hur bra datorvana de intervjuade har. Där 1 är mindre bra och 9 är mycket bra.*



## 8.2 Spelinstruktioner

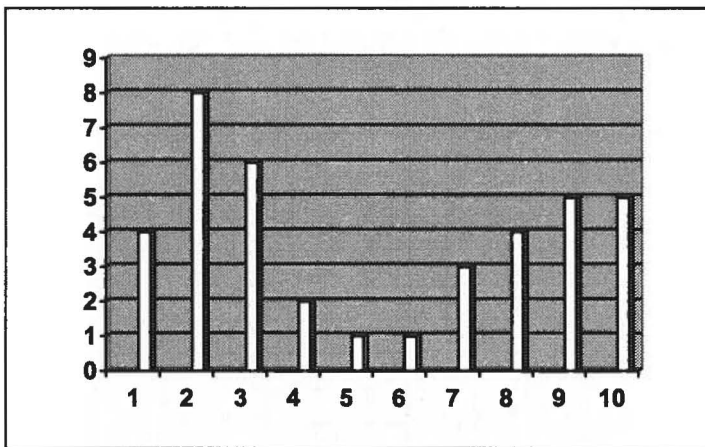
De flesta testpersonerna tyckte inte att spelinstruktionerna var bra för att det var svårt att förstå vilka rutor instruktionerna gällde och för att det fanns för mycket text i rutorna med för litet typsnitt. Något som många eftersökte var att det skulle finnas en bakåt funktion, så man skulle kunna bläddra tillbaka om man missat något. Det uppskattades inte heller att instruktionerna och spelplanen var blandade för att det var svårt att koncentrera sig på en viss punkt. En indikator på hur många instruktioner som man läst och hur många man hade kvar saknades. Somliga tyckte även att språket var för svårt med en massa termer som de inte förstod. Efter att ha spenderat en viss tid på att läsa instruktioner bläddrade de flesta förbi den sista delen pga att det helt enkelt fanns för mycket att läsa och att det var svårt att förstå.



**Diagram 2:** *Fördelningen på hur bra de intervjuade tycker spelinstruktionerna var. Där 1 är mindre bra och 9 är mycket bra.*

### 8.3 Förväntningar efter spelinstruktionerna

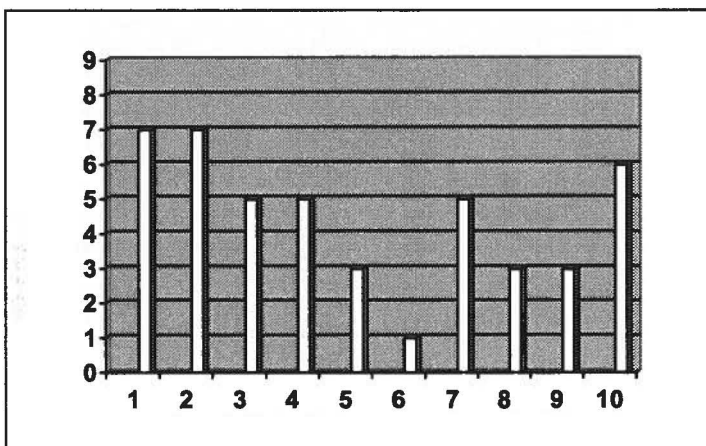
Det fanns delade meningar om hur roligt de intervjuade trodde spelet var efter de inledande spelinstruktionerna. En del var inte särskilt optimistiska eftersom de tyckte spelinstruktionerna var för långa. Ju mer var person läste och ju mer komplicerade de tyckte spelet verkade desto mindre roligt trodde de att det skulle bli. Eftersom en del inte förstod texten så bra, trodde de att spelet skulle vara för svårt för dem att bemästra.



**Diagram 3:** *Fördelningen på de intervjuades förväntningar efter att de läst spelets spelinstruktioner. Där 1 är mindre roligt och 9 är mycket roligt.*

#### 8.4 Spelets utseende

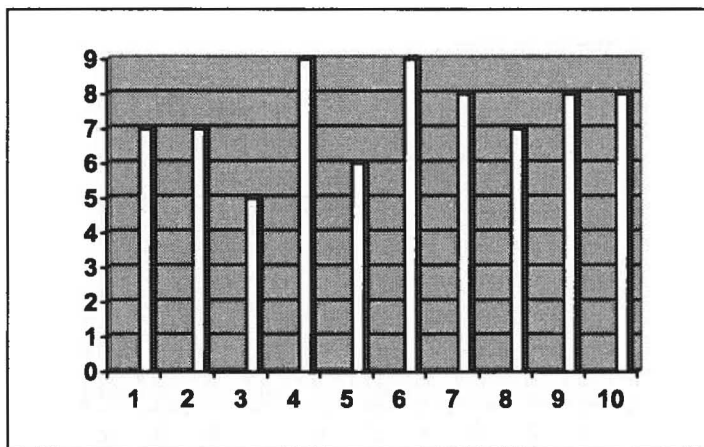
Utseendemässigt tyckte spelarna att spelet såg tillfredställande ut. Det fanns dock ett par frågetecken. Många förstod inte varför planen var uppdelad i tre olika färger, förklaringen var för diskret och uppfattades inte. Objekten tyckte de flesta var fint ritade. Objekt som Home och Store uppskattades extra mycket eftersom testpersonerna genast förstod vad de innebar. Många tyckte däremot att spelets utseende förlorade lite på att det fanns så många objekt nära varandra. Det tog rätt lång tid för dem flesta att uppfatta stolparna bredvid objekten



**Diagram 4:** *Fördelningen på hur fint de intervjuade tyckte spelets utseende är. Där 1 är mindre fint och 9 är mycket fint.*

## 8.5 Spelets kompliceringsgrad

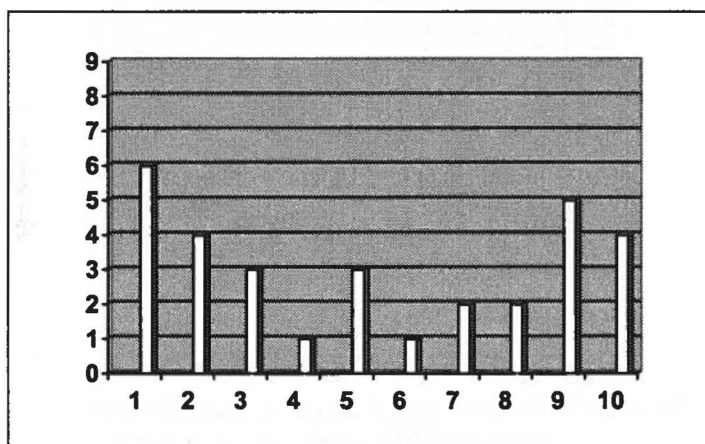
Nästan alla tyckte att spelet var för komplicerat. Spelarna visste inte vad de olika parametrarna innebar och hur de påverkade byggnaderna. Mot slutet av spelgången satt spelarna oftast bara och klickade och hoppades på det bästa. Flera hade svårt att inse om deras ingripanden hade förbättrat eller försämrat resultatet. Parametrarnas språk tyckte många var för svåra att förstå. En del berättade att de inte tyckte de var tillräckligt utbildade för att förstå alla termer.



**Diagram 5:** *Fördelningen på hur komplicerat de intervjuade tycker att spelet är. Där 1 är mindre komplicerat och 9 är mycket komplicerat.*

## 8.6 Allmänt om spelet

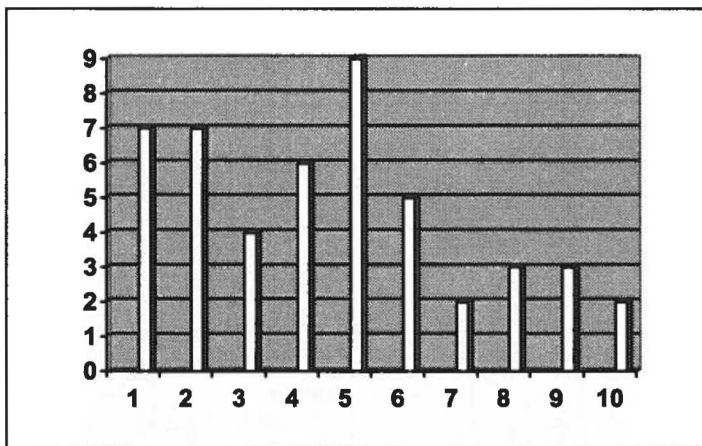
Många av de intervjuade tyckte inte om spelet eftersom det var svårt att förstå vad som krävdes av dem. En del av spelarna förklarade att de blev smått chokade till en början då de såg att det fanns så mycket på spelplanen som de inte förstod sig på. Lastbilarna som körde fram och tillbaka tyckte många gjorde att de hade svårt att fokusera sig på objektet som var markerat, samt att vissa också fann telefonledningarna mellan byggnaderna störande. Många tyckte att de kände sig förvirrade efter ett tag då de bara fick nya meddelanden och allting började röra sig ännu snabbare. Spelet blev också lidande av att tre byggnader blev beskrivna av förkortningar som folk inte visste vad de stod för.



**Diagram 6:** *Fördelningen på vad de intervjuade allmänt tyckte om spelet. Där 1 är mindre bra och 9 är mycket bra.*

## 8.7 Logistik förståelsen

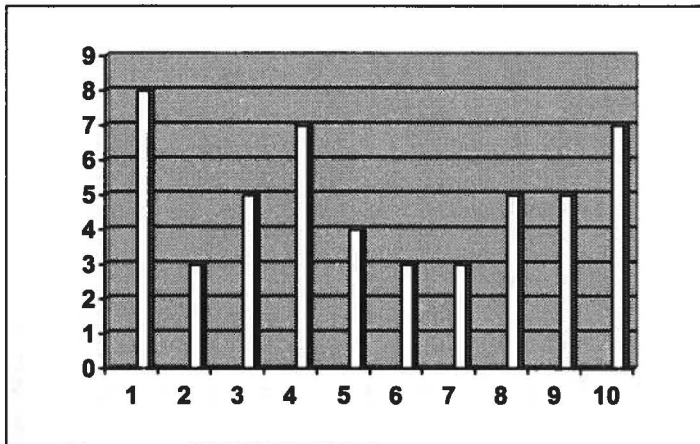
Alla gillade idén bakom spelet eftersom spelarna fick en inblick av hur verksamheten fungerar och hur varorna cirkulerar. De insåg av spelets gång var i kedjan de befinner sig. Många visste inte vad jordgloben högst uppe i det högra hörnet var, vilket gjorde att den högra delen av spelplanen blev lidande. Att lastbilarna åkte fram och tillbaka gjorde så att många inte kunde studera hela varuflödet grundligt och därför tyckte de blev avbrutna hela tiden.



**Diagram 7:** *Fördelningen på de intervjuades förståelse på hur varuflödet fungerar efter de har spelat. Där 1 är mindre bra och 9 är mycket bra.*

## 8.8 Objektens placering

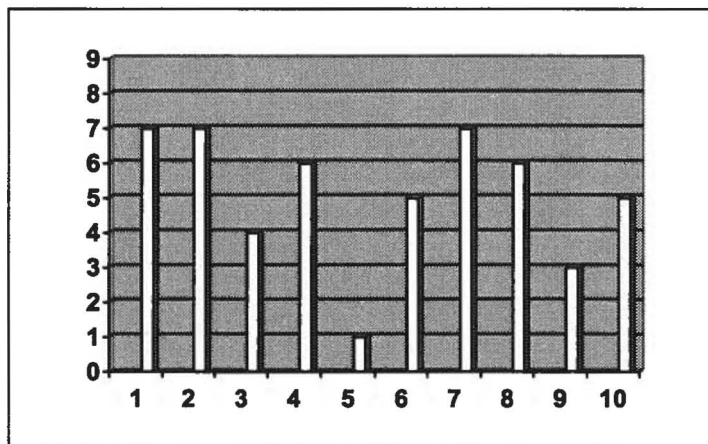
Placeringarna av objekten Home och Store tyckte alla var bra. Range missade många till en början på grund av placering och färg. Trade uppfattade de flesta som en bild och inte ett objekt. DC's placering förklarade för testpersonerna att den var en viktig del av varuflödet.



**Diagram 8:** *Fördelningen på hur bra de intervjuade tyckte spelets objekt var placerade. Där 1 är mindre roligt och 9 är mycket roligt.*

## 8.9 Objektens färger

Byggnadernas blå färg uppskattades av alla medan fabrikernas och lastbilarnas röda färg blev uppfattad som extrem. Många önskade att både fabriken och lastbilarna hade haft en mer diskret färg. Jordglobens färg tyckte spelarna var fin fastän de inte förstod jordglobens innebörd.



**Diagram 9:** *Fördelningen på hur bra de intervjuade tyckte objektens färger var. Där 1 är mindre bra och 9 är mycket bra.*



## 9 Slutsats och diskussion

Att utveckla en mjukvara med hög användbarhet är inte lätt, oberoende av vad mjukvaran är till för. Varje domän har sina egna användbarhetskrav som man måste ha i åtanke, men som först måste bli hittade. Vid utvecklandet av en mjukvara som är till för en heterogen grupp dvs en grupp där personerna har olika bakgrunder och kunskaper, måste många saker beaktas. Spelet måste vara lätt att spela, så pass lätt att personen som spelar inte är tvungen att komma ihåg information som den inte behöver. Svåra uppgifter måste vara lätta att utföra och liknande inställningar måste kunna göras på lika sätt. Det är viktigt att användaren med en gång förstår vad den ska göra och hur den ska göra det. När användaren sedan gör något måste den få en respons att spelet har uppfattat det och ett svar på om utförandet var bra eller dåligt.

Jag får inget intryck av att spelet har blivit utvecklat till en heterogen användargrupp. Spelet är för svårt att spela för en person utan antingen bra domänkunskap eller bra kunskaper inom företagsekonomi. Många termer används som många människor inte förstår. En hjälpfunktion innuti spelet är ett måste för de som inte kan eller kommer ihåg från de inledande instruktionerna, vad termerna betyder.

Spelinstruktionerna var svåra att läsa. Det fanns ingen bakåt funktion att använda om man tyckte det fanns något som man missat och ingen indikator på hur mycket man hade kvar att läsa. Språket tyckte en del var för komplicerat. Det var svårt att se vilka rutor som instruktionerna gällde samtidigt som att texten var för liten.

Spelet ser bra ut. Att planen var uppdelad i tre olika färger förstörde mer än vad det hjälpte då förklaringen knappt syntes. Objekten var fint ritade, speciellt omtyckta var Home och Store vilka genast förstods. Synd var dock att så många objekt låg så nära varandra, önskvärt var att spelplanen skulle vara lite större.

Det var svårt att spela en omgång. Det första de flesta tänkte var "vad ska jag göra nu?" Meddelande var svåra att se samtidigt som många var intetsägande. Det fanns för många grafer som var komplicerade och svåra att tyda. Från att man läst spelinstruktionerna var det svårt att komma ihåg objektens namn och deras förkortningar. Någon hjälp fanns dessutom inte att få. De olika parametrarnas innebörd var svåra att förstå. De spelande visste inte hur de skulle åtgärda de fel som de fått meddelande på. De visste ofta inte heller om ändringarna som de gjorde var bra eller dåliga för resultatet. Ibland fick de inte en direkt respons av en ändring, vilket gjorde dem ännu mer förvirrade. Att en ändring i ett objekt kunde påverka ett annat objekt observerades först av många mot slutet av omgången.

Placeringarna av objekten tyckte de flesta var bra. Färgen på fabrikena och lastbilarna var inte uppskattad då den var röd. Röd färg symboliserar världen över fara vilket inte är passande för fabriker och lastbilar. Färgerna var för övrigt fina

och objekten fint ritade. Synd var det också att många stora objekt rörde sig på spelplanen hela tiden, detta gjorde att ögonen automatiskt följde dem och många fick svårt för att koncentrera sig på det viktiga.

Spelet är till för att X's anställda ska förstå sig på hur företagets varuflöde fungerar och ser ut. Spelarna tyckte att idéen bakom spelet var bra då det klart och tydligt visade hur varuflödet ser ut. Synd var det dock att vissa objekt hade förkortningar som endast blivit förklarade i de inledande spelinstruktionerna vilket nästan alla glömt när spelet börjar. Detta gjorde att de fortfarande hade frågor kvar om varuflödet efter en spelomgång.

Jag hoppas slutligen att detta arbetet hjälper till att höja intresset av användbarhet hos utvecklarna av prototypen och att anmärkningarna hjälper för att göra detta och framtida spel bättre.

## 10 Referenslista

Faulkner, Kristine, (2000), Usability Engineering, Grassroots Series.

Löwgren, Jonas, (1993), Human-Computer Interaction, Studentlitteratur.

Löwgren, J. & Stolterman, E. (1998) Design av informationsteknik – materialet utan egenskaper. Lund: Studentlitteratur.

Nielsen, Jakob, (1993), Usability Engineering, Academic Press.

Norman, Donald, (1988), The design of Everyday Things, Doubleday.

Wharton, C & Rieman, J & Lewis, C & Polson, P. (1994). The cognitive walkthrough method: a practitioner's guide. i Nielsen, J & Mack, R. L. (red.) Usability inspection methods. USA: John Wiley & Sons, Inc.

## Bilaga 1

### Briefing

- 1) Vad är ditt namn?
- 2) Hur gammal är du?
- 3) Är du kvinna eller man?
- 4) Vad har du för bakgrund? (utbildning och arbetslivserfarenhet)

### Instruktioner

På nedanstående frågor ska du svara på en skala från 1 till 9 där 1 är sämst och 9 bäst.

- 5) Hur bra datorkunskap har du?

○ ○ ○ ○ ○ ○ ○ ○ ○ ○  
1 2 3 4 5 6 7 8 9

Mindre bra

Mycket bra

- 6) Hur stor erfarenhet har du av att spela online-spel?

○ ○ ○ ○ ○ ○ ○ ○ ○ ○  
1 2 3 4 5 6 7 8 9

Mindre stor

Mycket stor

### Test

7) Hur bra tycker du att de inledande spelinstruktionerna är?

1 2 3 4 5 6 7 8 9

Mindre bra

Mycket bra

8) Tycker du efter de inledande spelinstruktionerna att spelet verkar roligt?

1 2 3 4 5 6 7 8 9

Mindre roligt

Mycket roligt

9) Tänk nu högt när du spelar en omgång.

## De-briefing

10) Vad tycker du om spelets utseende?

1 2 3 4 5 6 7 8 9

Mindre fint

Mycket fint

11) Tycker du att placeringen av objekten på skärmen är bra?

1 2 3 4 5 6 7 8 9

Mindre bra

Mycket bra

12) Vad tycker du allmänt om spelet?

1 2 3 4 5 6 7 8 9

Mindre bra

Mycket bra

13) Skulle du i framtiden vilja spela spelet igen? (Varför?)

14) Skulle du rekommendera en kollega att spela spelet?

15) Hur lätt tycker du det är att förstå hur spelet går till?

1 2 3 4 5 6 7 8 9

Mindre lätt

Mycket lätt

16) Vad tycker du om objektens färg?

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

1 2 3 4 5 6 7 8 9

Mindre bra

Mycket bra

17) Hur komplicerat tycker du spelet är?

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

1 2 3 4 5 6 7 8 9

Mindre komplicerat

Mycket komplicerat

18) Vilket objekt på skärmen tittade du mest på?

19) Hur mycket bättre tycker du att du förstår dig på Ikeas verksamhet?

○ ○ ○ ○ ○ ○ ○ ○ ○ ○

1 2 3 4 5 6 7 8 9

Inget alls

Mycket bättre

20) Vad tycker du var speciellt bra i spelet?

21) Vad tycker du var mindre bra i spelet?

22) Vad skulle du vilja ändra på?

23) Övrigt?