



TeachMePoker

Grupp 18

Verifiering och valideringsdokument

V. 3.0

2017-05-25

Dokumenthistorik

Datum	Version	Beskrivning	Författare
170410	0.1	Lagt till <i>Användbarhetstestning: Semistrukturerad Intervju</i> samt <i>Referens</i> till detta.	Vedrana Zeba
170410	0.1.1	Lagt till <i>Testfall kravbaserad systemtestning, Spårningsmatris</i> klar.	Lykke Levin
170410	0.1.2	Lagt till <i>Riktlinjer för dokument</i> , lagt till <i>checklistor från granskningsmöten</i> samt lagt till checklistor från dokumentgranskning under <i>Bilagor</i> . Lagt till <i>granskningsprotokoll</i> DG.PP.1.	Lykke Levin
170410	0.1.3	Riktlinjer för kodgranskning och checklista för kodgranskning	Amin Harirchian
170410	0.1.4	Granskningsprotokoll	Amin Harirchian
170411	0.2	Text för <i>processmodell</i> under <i>Dokumentgranskning</i> tillagd. Lagt till <i>granskningsprotokoll</i> DG.KD.1 och DG.K.1	Lykke Levin
170411	0.2.1	Användbarhetstestning: Think-Aloud-Protocol text skriven.	Rikard Almgren
170412	0.3	<i>Testprocess</i> färdigskriven	Lykke Levin Vedrana Zeba Rikard Almgren
170412	0.3.1	<i>Kravbaserad systemtestning</i> tillagd	Lykke Levin Vedrana Zeba Max Frennessen
170412	0.3.2	Uppdaterat <i>ordlistan</i>	Amin Harirchian
170412	0.3.3	Uppdaterat <i>testfall, spårningsmatris</i> och <i>testrapport</i>	Rikard Almgren Max Frennessen
170412	0.3.4	Uppdaterat <i>referenser, innehållsförteckning</i> , lagt till <i>referens</i> .	Lykke Levin
170413	1.0	Klar för inlämning sprint2	Lykke Levin
170428	1.1	<i>Analysrapport: Semistrukturerad Intervju S.I.1</i> tillagd. Bilaga för <i>Semistrukturerad intervju #1</i> och <i>Transkript: Semistrukturerad intervju (S.I.1 #1)</i> tillagda.	Vedrana Zeba
170501	1.2	<i>Gjort ändringar enligt kritik från retrospektmöte.</i>	Rikard Almgren
170502	1.3	<i>Uppdaterat riktlinjer för kod</i>	Amin Harirchian
170502	1.3.1	Uppdatering av <i>riktlinjer för designdokument</i>	Lykke Levin
170503	1.3.2	Lägger till testfall för kravbaserad testning	Max Frennessen
170504	1.4	<i>Innehållsförteckning</i> uppdaterad	Lykke Levin
170504	2.0	Klar för inlämning sprint3.	Lykke Levin
170511	2.1	<i>Uppdaterat kodgranskning, ordlistan</i>	Amin Harirchian

170513	2.1	<i>Ändra på Whitebox beskrivning och prioritering</i>	Max Frennessen
170513	2.1	<i>Ändra och fixa till mallar för testfall</i>	Max Frennessen
170515	2.2	Brutit ur alla stora <i>bilder</i> och <i>DG-checklistor</i> och lagt till i Bifogat-mappen. Kontrollerat <i>referensformatering</i> .	Lykke Levin
170518	2.3	Ändrat i <i>Användbarhetstestning: Semistrukturerad Intervju</i> samt lagt till bilaga för <i>Semistrukturerad intervju #2</i> .	Vedrana Zeba
170519	2.4	Lagt till Kodgranskning rapport (KD.2)	Amin Harirchian
170521	2.5	Reviderat <i>Dokumentgranskningsdelen</i> enligt kommentarer från <i>DG.2</i> .	Lykke Levin
170521	2.5.1	Lagt till testfall TF.26 och TF.27 (spårningsmatris och testrapport)	Lykke Levin
170525	2.6	Lagt till användbarhetstestning TAP	Rikard Almgren
170525	2.6.1	Skrivit <i>Testrapport Användbarhetstestning - Semistrukturerad Intervju, TR.SI.AB.2</i>	Vedrana Zeba
170525	2.6.2	<i>Länkat</i> alla bifogade dokument, lagt till <i>VoVgranskningsrapporten</i> .	Lykke Levin
170525	2.6.3	Lagt till ID för testade krav i testfallen.	Amin Harirchian, Vedrana Zeba.
170525	3.0	Klar för slutgiltig inlämning, sprint4.	Lykke Levin

Innehåll

Dokumenthistorik	2
Verifiering och valideringsdokument	6
Syfte	6
Ordlista	6
Referenser	8
Testprocess	9
Granskning	10
Kodgranskning	10
Dokumentgranskning	15
Testning	18
Kravbaserad systemtestning	18
White box-testning: Path Coverage	18
Användbarhetstestning: Semistrukturerad Intervju	19
Användbarhetstestning: Think-Aloud-Protocol	19
Testfall kravbaserad systemtestning	20
Systemtestning av funktionella Must-prioriterade krav	20
Spårningsmatris	36
Spårningsmatris över funktionella Must-prioriterade krav	36
Spårningsmatris över kvalitativa krav	36
Granskningsprotokoll	37
Granskningsprotokoll dokumentgranskning DG.PP.1	37
Granskningsprotokoll dokumentgranskning DG.KD.1	37
Granskningsprotokoll dokumentgranskning DG.K.1	37
Granskningsprotokoll dokumentgranskning DG.VoV.1	37
Granskningsprotokoll kodgranskning KG.1	37
Granskningsprotokoll kodgranskning KG.2	38
Testrapporter	39
Testrapport Kravbaserad systemtestning TR.KS	39
Testrapport White-box testning TR.WB	40
Testrapport Användbarhetstestning - Think-Aloud-Protocol, TR.TAP.AB.	45
Testrapport Användbarhetstestning - Semistrukturerad Intervju, TR.SI.AB.1	48
Testrapport Användbarhetstestning - Semistrukturerad Intervju, TR.SI.AB.2	52
Bilagor	54
Think-Aloud-Protocol Session 1-5 (scenario)	54
Semistrukturerad intervju #1 (intervjufrågor)	55

Transkript: Semistrukturerad intervju (S.I.1 #1)	57
Semistrukturerad intervju #2 (intervjufrågor)	60

Verifiering och valideringsdokument

Syfte

Detta dokument innehåller en beskrivning av projektets testprocess samt beskriver hur granskningar och testning utförs. Dokumentet fungerar även som ett bokförings- och referensdokument för de testfall och test som har utförts. Testfall som tas upp i dokumentet har direkt koppling till projektets kravdokument.

Vidare innehåller dokumentet ett flertal granskningsprotokoll från dokument- och kodgranskningar samt test- och analysrapporter.

Ordlista

AI (Artificial Intelligence): En datorstyrd spelare som med hjälp av olika beräkningar fattar beslut och fungerar som en spelare.

All in: Användaren/AI går in med alla sina pengar/medel.

Array: En array är ett sammanhängande antal element av samma typ som representeras med ett variabelnamn.

Black-box testning: Testning av program där testaren inte har tillgång till någon del av programkoden för det som ska testas [1].

Call: Användaren/AI synar den högsta satsningen och är därmed delaktig i rundan [2].

Check: Användaren/AI säger han är med men lägger inte in en summa [3].

Fold: Användaren/AI lägger sig för den spelomgången [4].

Färg: I en kortlek finns där fyra olika färger som ett kort kan ha. Färgerna är Hjärter, Spader, Klöver och Ruter.

Hand-klassen: Den klass som räknar ut vad spelaren har och ger råd till spelaren beroende på kortens styrka.

Hålkort: Hålkort är de personliga kort som en spelare tilldelas i ett floppspel så som Texas Hold'em. Dessa hålkort är privata och får endast användas av de som de tilldelats till [5].

If-sats: En sats i programmering där koden körs om ett eller flera villkor uppfylls.

Javadoc: Javadoc är ett verktyg för att skapa API-dokumentation i HTML-format från Doc kommentarer i källkoden.

Klassdiagram: En karta över kodfiler som används i projektet.

Konstruktor: En konstruktor är en slags metod som körs när ett objekt skapas.

Loop: När datorn kör samma programkod om och om igen.

Pott: De pengar/medel som är placerade i mitten på bordet och som spelarna försöker vinna.

Paket: En sorteringsmetod för kodfiler. Kodfiler som har en koppling läggs i samma paket.

Path coverage: Testmetod inom white-box testning [6].

Raise: Användaren/AI höjer vad de andra måste lägga in för att vara med på denna rundan [7].

Sträng (eng. String): Sträng är en datatyp som handlar om text.

Stubb: Kort program eller del av program som representerar ett större program eller en procedur [8].

Switch-sats: Switch-satser används när programmet behöver kunna göra val, baserat på olika fall.

Texas Hold'em Poker: En variation av kortspelet poker i vilken varje spelare får två egna kort och delar en gemensam kortpool på fem kort som ligger öppna på bordet [9].

UTF (eng. Unicode transformation format): Sätt att representera tecken som finns i Unicode på ett sätt som passar datorer. Det finns tre varianter av UTF, nämligen UTF-8, UTF-16 och UTF-32 [10].

Variabel: En variabel är något som kan ändras, har en viss typ. Variabler kan vara lokala eller globala.

White-box testning: Testning av program där testaren har tillgång till all programkod för det som testas [11].

Whitespace: Oanvända rad i koden.

Referenser

- [1] Lotsson, A., "Black box testing", IT-ord, (2017). [Online]. Tillgänglig: <https://it-ord.idg.se/?s=black-box+testning> [2017-04-12].
- [2] "Call", Pokernews, (2017). [Online]. Tillgänglig: <https://se.pokernews.com/pokertermer/call.htm> [2017-04-12].
- [3] "Check", Pokernews, (2017). [Online]. Tillgänglig: <https://se.pokernews.com/pokertermer/check.htm> [2017-04-12].
- [4] "Fold", Pokernews, (2017). [Online]. Tillgänglig: <https://se.pokernews.com/pokertermer/fold.htm> [2017-04-12].
- [5] "Hålkort", Pokernews, (2017). [Online]. Tillgänglig: <https://se.pokernews.com/pokertermer/halkort.htm> [2017-04-12].
- [6] Shilpa, C. Roy., "How to perform White Box testing -Explained with a simple example", (2016). [Online]. Tillgänglig: <http://www.softwaretestinghelp.com/white-box-testing-techniques-with-example/> [2017-04-11].
- [7] "Raise", Pokernews, (2017). [Online]. Tillgänglig: <https://se.pokernews.com/pokertermer/raise.htm> [2017-04-12].
- [8] Lotsson, A., "Stubb", IT-ord, (2017). [Online]. Tillgänglig: <https://it-ord.idg.se/ord/stubb/> [2017-04-12].
- [9] Merriam-Webster online dictionary, "Texas Hold'em Poker", (2017). [Online]. Tillgänglig: <https://www.merriam-webster.com/dictionary/Texas%20Hold'em> [2017-04-05].
- [10] Lotsson, A., "UTF", IT-ord, (2017). [Online]. Tillgänglig: <https://it-ord.idg.se/ord/utf/> [2017-04-12].
- [11] Lotsson, A., "White box testing", IT-ord, (2017). [Online]. Tillgänglig: <https://it-ord.idg.se/ord/white-box-testing/> [2017-03-24].
- [12] Erickson, C., Jorgensen, P., "Object-oriented integration testing", *Communications of the ACM*, vol. 37, no. 9, september 1994. [Online]. Tillgänglig: <http://dl.acm.org.proxy.mah.se/citation.cfm?doid=182987.182989> [2017-04-12].
- [13] Tsui, F. F., Karam, O. & Bernal, B., *Essentials of software engineering*. Burlington, MA : Jones & Bartlett Learning, 2013.
- [14] Jones, C., "Software quality in 2012: A Survey of the State of the Art", (2012). [Online]. Tillgänglig: <http://sqgne.org/presentations/2012-13/Jones-Sep-2012.pdf> [2017-04-12].
- [15] Eklund, S., *Arbeta i projekt: individen, gruppen, ledaren*, 4:7 uppl. Lund: Studentlitteratur, 2011, s. 128.
- [16] Bennet, S., Mcrobb, S. & Farmer, R., *Object-Oriented Systems Analysis and Design using UML*, 4:e uppl, Maidenhead, Berkshire: McGraw-Hill Higher Education, 2010.
- [17] Qualitative Research Guidelines Project, "Semi-structured Interviews", (2017). [Online]. Tillgänglig: <http://www.qualres.org/HomeSemi-3629.html> [2017-04-10].
- [18] Lewis, C. H., *Using the "Thinking Aloud" Method In Cognitive Interface Design*. Yorktown Heights, NY: IBM T.J. Watson Research Center, 1982.

Testprocess

Projektets testprocess utgår ifrån en iterativ modell i enlighet med de agila utvecklingsprocesserna projektet använder som utgångspunkt. Kravbaserad testning utförs löpande under sprinter efterhand som krav implementeras. När två kompletta moduler som ska arbeta tillsammans är färdiga utförs även integrationstester. Regressionstestning utförs vid varje integrationstest [12]. Testning genomförs av enskilda individer vid modultestning samt delar av gruppen eller gruppen i helhet vid integrationstestning. De individuella fördjupningarna, *dokumentgranskning*, *kodgranskning*, *white-box testning* och *användbarhetstestning*, genomförs av gruppen med ledning av den som har övergripande ansvar. De fördjupningar som kan knytas an till krav i kravdokumentet kommer att genomföras med åtanke på kravens prioritering. När *användbarhetstestning* sker så består testgruppen av opartiska deltagare utanför projektgruppen. Iteration av de individuella fördjupningarna sker minst en gång, när större delen av kraven är implementerade.

Testfall ska skrivas för varje systemtest och white-box test som utförs och dessa ska antecknas i detta dokument. Dokumentation över testfallen innehåller förberedelser inför testning, teststeg och förväntade resultat. Varje test har ett unikt identifikationsnummer. Kravbaserade testfall ska härledas till krav i kravdokumentet och dokumenteras i korrekt spårningsmatris i detta dokument. Alla tester som utförs ska dokumenteras i en testrapport som kan härledas till ett testfall. Testrapporter dokumenteras efter vilken typ av test som utförs. I testrapporten antecknas datum då testerna utfördes, vilket testfall som testats, vilken version av kod eller dokument som blivit testat och vem i gruppen som utfört testet. Testrapporter kontrolleras och uppvisas vid gruppmöten. Vid avvikande resultat är det den som är ansvarig som delegerar vem som ska korrigera felen.

Granskning

Kodgranskning

Metoden som har valts för kodgranskningen är av typen informell inspektion (möte). Hela gruppen ska delta i mötet och varje person blir tilldelad en roll under mötet. Eftersom gruppen får checklistan i början av mötet modifieras metoden från en formell inspektion [13] till en informell inspektion. Roller som moderator, läsare och kodförfattare delas ut mellan gruppmedlemmar [13]. Gruppen har planerat att ett möte på viss tid. Eftersom gruppmedlemmarna har jobbat med produkten från början, hoppar gruppen över steget. De som ska granska koden ska förbereda sig inför mötet genom att kolla igenom all kod som ska granskas. Moderatoren ska skriva någon form av checklista som ska hjälpa att hitta fel snabbare. Under mötet ska gruppen granska och testa kod tillsammans och följa checklistan och riktlinjer för kod och hitta fel. Mötet ska vara i högst 2 timmar och alla kommentarer ska dokumenteras (granskningsprotokoll). Sedan ska författaren få granskningsprotokollet och fixa alla problem. Till sist ska moderatoren dubbelkolla alla de granskade kodfilerna som författaren har korrigerat [13].

Kodgranskningen ska göras i två omgångar. Den första kodgranskningen ska göras under sprint 2 och den andra kodgranskningen ska göras under sprint 4.

Gruppen har ritat ett klassdiagram för projektet. Klassdiagrammet innehåller klasser/kodfiler som är tillgängliga och diagrammet visar relationer mellan klasser i den här applikationen. Gruppen ska granska klasser i en prioriteringsordning av vilka klasser som har störst påverkan på projektet. Gruppen ska granska först de klasserna som har mest relation med varandra och klasserna som innehåller de viktiga funktioner och sedan ska övriga klasser granskas.

Riktlinjer för kod

1. Alla paketnamn (packages) ska börja med liten bokstav.
2. Alla klassnamn ska börja med stor bokstav.
3. Alla filnamn ska börja med liten bokstav.
4. Alla klasser ska tillhöra något specifikt paket.
5. Alla källkodsfiler ska ha UTF-8 [10] formatering.
6. Man ska ha en rads mellanrum (whitespace) mellan olika metoder.
7. När det gäller mellanrum i olika strängar (Strings) i koden ska de olika escape sequences som (\b, \t, \n, \f, \r, \", \' and \\) användas.
8. Alla metodnamn ska börja med liten bokstav.
9. Alla parametrar ska börja med liten bokstav.
10. Man ska undvika att separera långa parameternamn med binde-eller understreck och istället skriva längre parameternamn Stor bokstav som separator (till exempel: firstNumber).
11. Källkodsfiler ska följa den här ordningen:
 - 11.1. Paketdeklarationer
 - 11.2. Importdeklarationer
 - 11.3. Huvudklassen
12. Alla importanrop ska skrivas på en egen rad till exempel:

```
import java.x;
import java.y;
```
13. Alla klasser ska följa samma logiska ordning. Konstruktor först och sedan metoder.
14. Alla metoder ska följa den här formateringen:

```
void doThis() {
}
```
15. Alla If-satser ska följa den här formateringen:

```
if (...) {
}
```
16. Alla loopar ska följa den här formateringen:

```
for (...) {
}
```
17. Alla try-satser ska följa den här formateringen:

```
try {
    do something();
} catch (Exception e) {
}
```
18. Varje ny metod eller block ska ha 1 rads mellanrum.

19. Alla kodfiler ska vara kommenterade och det ska finnas Javadoc kommenterar.
20. För array-deklarationer ska man ha mellanrum mellan typ och variabel till exempel:
`List<String> list`
21. Formatering för arrayer ska vara `int[] array = new int[] {5, 6}` eller `int[] array = new int [2]`.
22. Alla oanvända importar eller variabler skall tas bort från koden.
23. Varje variabel ska deklareras för sig själv som till exempel:
`int a;`
`int b;`
`int c,d;` är inte tillåtet.
24. Man ska endast använda lokala variabler om det behövs och då ska den användas direkt efter deklarationen.
25. Switch-satser ska följa den här formateringen:
`switch (...) {`
`case 1:`
`case 2:`
`dosomething2();`
`case 3:`
`dosomething3();`
`break;`
`default:`
`dosomething(...);`
`}`
26. Alla static final variabler ska skrivas med `UPPER_CASE`.
27. Alla parametrar och variabelnamn ska börja med liten bokstav.
28. Man ska följa Javadoc standard för varje block (`@param`, `@return`, `@throws`, `@deprecated`).

Checklista för granskningsmöte**1. Grundläggande kodgranskningsfrågor:**

- Kan man förstå koden lätt?
- Är koden skriven enligt gruppens riktlinjer?
- Är samma kod upprepade mer än två gånger?
- Kan man enhetstesta / felsöka koden lätt för att hitta orsaken?
- Är denna funktion eller klass för stor? Om svaret är ja, har funktionen eller klassen för många ansvarsområden?

2. Detaljerade kodgranskningsfrågor:**Struktur**

- Har koden fullständig och korrekt genomförd design?
- Överensstämmer koden med alla relevanta kodstandarder?
- Är koden välstrukturerad, konsekvent i stil och konsekvent formaterad?
- Finns det några onödiga procedurer eller någon onåbar kod?
- Finns det några överblivna stubbar [8] eller testrutiner i koden?
- Kan någon kod ersättas av anrop till externa återanvändbara komponenter eller biblioteksfunktioner?
- Finns det några block av upprepade kod som kan kondenseras till ett enda förfarande?
- Används lagring effektivt?
- Finns det några moduler som är alltför komplicerade och bör omstruktureras?

Kodformatering

- Är alla källkodsfiler i UTF-8 [10] formatering?
- Börjar alla parametrar och variabelnamn med liten bokstav?
- Följer alla If-satser och loopar riktlinjerna för formatering?

Dokumentation

- Är koden tillräckligt dokumenterad med en enkel att underhålla kommenterad stil?
- Överensstämmer alla kommentarer med koden?
- Är alla kodfiler kommenterade?
- Finns det Javadoc kommentarer i alla kodfiler?

Variabler

- Är alla variabler korrekt definierade med meningsfulla, konsekventa och tydliga namn?
- Har alla tilldelade variabler rätt datatyp?
- Finns det några överflödiga eller oanvända variabler?

- Deklareras alla variabler var för sig?
- Börjar alla parametrar och variabelnamn med liten bokstav?

Loopar och if-satser

- Är alla loopar och if-satser, grenar och logiska konstruktioner fullständiga, korrekta och riktigt inkapslade?
- Är de vanligaste fallen testade först i IF-ELSE IF kedjor?
- Är loopars avslutningsförhållanden uppenbara och uppnås de alltid?
- Kan några if-satser som är inneslutna i loopar placeras utanför loopar?
- Har koden i loopen undvikit att manipulera indexvariabler eller använda denna vid utgång från loop?

Dokumentgranskning

Vald metod för dokumentgranskning är en informell inspektion med modifikationer för att passa projektets storlek och utvecklarnas arbetsmetod [13]. Planeringsstadiet inför granskningsmötet består av framtagning av dokument som ska användas under inspektionen, så som checklistor, mallar för felrapportering och rapportmall. Moderatoren skickar inte ut detta material innan inspektionen då genomgång och faktisk granskning ska göras tillsammans. Gruppen kommer inte att samlas innan inspektionen för en genomläsning av dokumenten.

Dessa två anpassningar av metoden görs då samtliga gruppmedlemmar är författare av de dokument som ska granskas. Alla gruppmedlemmar är alltså väl införstådda med dokumentens innehåll och struktur.

Checklistor delas ut i början av inspektionen tillsammans med de mallar som ska fyllas i. För varje dokument hålls ett två timmar långt möte. Under inspektionen fungerar en av gruppmedlemmarna som läsare av dokumentet samtidigt som övriga deltagare agerar som skrivare som söker och skriver ner fel de hittar. Efter varje avslutat stycke som läsaren läst upp, diskuteras stycket i grupp. Moderatoren har ansvar över att varje upptäckt fel förs in i felrapporten, även om skrivarna också antecknar. Detta för att täcka de fel som diskuteras fram under granskningen.

Då alla gruppmedlemmar är författare till samtliga dokument är det var gruppmedlems ansvar att korrigera eventuella felaktigheter som hittats i dokumenten. Moderatoren sammanställer de felrapporter som fyllts i under mötet och delegerar ut ansvarsområden till gruppmedlemmar. Dessa baseras på tidigare ansvarsområde under dokumentets utformning. När samtliga gruppmedlemmar rapporterat att det granskade dokumentet har reviderats efter de synpunkter som antecknats, läser moderatoren igenom förändringarna i dokumenten och publicerar en ny version.

De dokument som granskas prioriteras utifrån allvaret av konsekvens som utebliven granskning medför. Högst prioritering har kravdokumentet då tidig upptäckt av fel i kraven minimerar framtida kostnader för åtgärder [14]. Dokumentgranskning sker två gånger under projektet. Den första granskningen skedde under sprint 2 då projektplanen och kravdokumentet hade en första version. Det andra granskningstillfället kommer att ske under andra veckan av sprint 4. Prioriterat dokument kommer att vara verifiering- och valideringsdokumentet med tillhörande testrapporter av projektets krav.

Riktlinjer för dokument

Följande riktlinjer gäller för samtliga dokument.

Riktlinjer för layout, struktur och namngivning

Samtliga dokument som gruppen framställer ska skrivas i typsnitt Times New Roman. Brödtext skrivs med storlek 12, huvudrubriker skrivs i fetstil i storlek 16 och underrubriker skrivs i fetstil i storlek 14. Rubriker för att avdela stycken i brödtext skrivs i fetstil i storlek 12. Stycken delas in genom en blankrad. Brödtext skrivs direkt under rubrik eller underrubrik.

Vid namngivning av dokument följs strukturen *Dokumentnamn v.x.x* där Dokumentnamn är titel på dokument som angivits på försättsbladet följt av versionsnummer.

Versionshantering

Varje dokument ska ha ett uppdaterat versionsnummer i försättsbladet och sidhuvudet. För varje nytt datum ska dokumentet tilldelas ett nytt versionsnummer i form av ett heltal och ett decimaltal. Vid multipla ändringar under samma datum ska versionsnumret utökas till ett heltal och två decimaltal. När dokument har reviderats eller ska skickas in för granskning till sprintar ska dokumentet tilldelas ett nytt versionsnummer i form av ett heltal med ett decimaltal som ska vara 0. Nedan följer exempel på versionshantering.

Datum	Version	Beskrivning
17-04-01	0.1	Ändringar
17-04-01	0.1.1	Andra ändringar
17-04-02	0.2	Andra ändringar
17-04-03	1.0	Klar för inlämning

Riktlinjer för specifika benämningar

Samtliga dokument ska använda ordet *användare* som beteckning för applikationens användare. Samtliga dokument ska använda ordet *AI-spelare* som beteckning för användarens motspelare (datorn). Inga dokument ska innehålla subjektiva formuleringar som *vi*, *vår*, *oss* och *våra*. Inga dokument ska innehålla förkortningar i någon form.

Riktlinjer för kravdokument

Krav ska skrivas till under den kategorin kravet tillhör. Huvudkategorierna är *Funktionella krav* och *Kvalitativa krav*. Är det nya kravet ett funktionellt krav ska detta delas in i en av de två underkategorierna *Boundary* eller *Logik* samt följande underkategori *Användarkrav* eller *Systemkrav*. Kravet ska sedan prioriteras efter MoSCoW-modellen [15]. De funktionella kraven identifieras med hjälp av bokstavsprefix för huvudkategori, underkategori 1, underkategori 2, prioritering samt löpnummer. Är det nya kravet ett kvalitativt krav ska detta delas in i en av de två underkategorierna *Begränsningar* eller *Användbarhet*. De kvalitativa kraven identifieras med hjälp av bokstavsprefix för huvudkategori, underkategori samt löpnummer. Varje nytt krav ska läggas till i slutet av tillhörande kategori. Nya krav och ändringar i tidigare tillagda krav ska dokumenteras i dokumenthistoriken och vara spårbara med hjälp av kravets id. För mer detaljer och exempel se kravdokumentet, avsnitt Funktionella krav och/eller Kvalitativa krav.

Riktlinjer för designdokument

Samtliga användningsfallsbeskrivningar som dokumenteras i designdokumentet ska följa samma struktur. Endast en aktivitet ska beskrivas i varje användningsfall. Alla beskrivningar

ska innehålla en beskrivning av *huvudflödet*, en del som beskriver *startvillkoren* och en del som beskriver *slutvillkoren* för användningsfallet.

Riktlinjer för verifiering och valideringsdokument

Testfall för kravbaserad testning ska ange vilka förberedelser som krävs för att utföra testfallet. Stegen i testfallet ska numreras och för varje testfall ska ett förväntat resultat beskrivas. Varje testfall har ett unikt identifikationsnummer som ska vara spårbart. Testfallen kategoriseras efter systemtestning av funktionella krav, och hur ifall de funktionella kraven är prioriterade som Must, Should, Could eller Would-krav, eller systemtestning av kvalitativa krav.

Checklista för granskningsmöte

Checklista för granskningsmöte gällande projektplan finns bifogat separat som pdf-fil ([Checklista_Granskning_Projektplan_v.1.0.pdf](#)).

Checklista för granskningsmöte gällande kravdokument finns bifogat separat som pdf-fil ([Checklista_Granskning_Kravdokument_v.1.0.pdf](#)).

Checklista för granskningsmöte gällande krav från kravdokument finns bifogat separat som pdf-fil ([Checklista_Krav_v.1.0.pdf](#)).

Mall för felrapport som användes under granskningsmöten finns bifogat separat som pdf-fil ([Felrapport_Mall_Dokument_v.1.0.pdf](#)).

Checklista för granskningsmöte gällande verifierings och valideringsdokument finns bifogat separat som pdf-fil ([Checklista_Granskning_VoVDokument_v.1.0](#)).

Testning

Kravbaserad systemtestning

Systemtestning ska utföras när systemet till större del är komplett. Systemtestning görs för att kontrollera så att alla komponenter fungerar som tänkt. Generellt utförs black-box testning under systemtestning. Detta genomförs typiskt i grupp. Testningen berör både funktionella och kvalitativa krav. Testningen kan ske enligt ett script eller vara utav en mer explorativ typ. Det finns många olika testmetoder att utgå ifrån under systemtestning [16].

Den kravbaserad systemtestning som ska göras under projektet kommer att testa de krav som är implementerade samt kontrollera om kraven är implementerade på ett korrekt sätt. De fall där ett krav ej har blivit implementerat på ett korrekt sätt ska kod relaterat till kravet skrivas om och omprövas. Ifall kravet inte ses som möjligt längre det vill säga de fall där tiden inte kommer räcka till för att korrigera felet. Dessa kommer då bortprioriteras för mer prioriterade krav. Gruppen kommer utgå från en scriptbaserad testning i enlighet med användningsfallsbeskrivningar som finns specificerade i designdokumentet.

White box-testning: Path Coverage

Path coverage är den valda metoden. Det path coverage kommer att vara är skapandet av scenarion som sedan testas emot koden. Dessa scenarion är gjorda för att testa en specifik del av koden och se ifall det motsvarar de förväntade resultatet man har.. Därefter testas dessa scenarion och det kontrolleras vilket resultat man får. Målet är att testa alla möjliga tänkbara alternativ minst en gång var och då få 100% täckning (coverage) över programmet. Vad som menas med täckning (coverage) är att man som utvecklare testar alla tänkbara scenarion och får på så sätt veta att inga helt oväntade resultat kan inträffa. Det man testar är bland annat att man skickar in specifika kort för att testa alla olika handstyrkor och se att rätt hand vinner. Man kontrollerar även att den hjälp spelaren får via Hand-Klassen är korrekt gällande de korten spelaren har. Utöver detta ser man även till att AI-spelaren gör beräkningar korrekt och genererar ett resultat som kan ses som lämpligt för de kortens styrka som AI-spelaren har enligt de pokerregler som finns [6].

Prioritering

Testningen ska komma att genomföras i överensstämmelse med kravens prioritering enligt kravspecifikationen. Det är då 'Must-kraven' som kommer prioriteras att testas och implementeras. Ifall alla 'Must' är implementerade och klara och mer tid finns kvar kommer resterande krav testas i följande ordning; 'Should', 'Could', 'Would'.

Användbarhetstestning: Semistrukturerad Intervju

Semistrukturerade intervjuer är en formell användbarhetstestningsmetod där intervjuaren ställer frågor till en respondent. Dessa frågor består dels utav förberedda frågor som intervjuaren förväntar sig få svar på, men också av eventuella oförberedda frågor som lämpar sig att undersöka under intervjuens gång. Metoden anses vara lämplig att använda när det finns få tillfällen till att intervjua samma respondent, samt när flera olika personer agerar rollen som intervjuare. Generellt utförs röstinspelningar under intervjun [17].

Två semistrukturerade intervjuer kommer att utföras. Den första intervjun kommer att användas för att hämta information om eventuella förväntningar som respondenterna har på applikationen. Den andra intervjun kommer fokusera på spelplanens design och utformning. Respondenterna kommer att bestå utav både personer som *kan* respektive *inte kan* spela kortspelet Texas Hold'em Poker. Frågorna planeras att utformas så att direkta ja- och nej svar undviks. Frågorna eftersträvas även att vara så neutrala som möjligt för att inte indirekt råka påverka respondentens svar.

Prioritering

Främst prioriteras frågor gällande förväntade funktioner som respondenten anser vara viktiga i applikationen. En del frågor kommer att beröra språk samt ordval som kommer att användas vid interaktion med applikationen. Frågor om design och utformning av spelplanen, samt om hur en tutorial bör utformas för att förtydliga spelplanen, kommer också att prioriteras.

Användbarhetstestning: Think-Aloud-Protocol

Think-Aloud-Protocol [18] är en informell metod för att testa användbarhet. När man gör en Think-Aloud-Protocol-undersökning så används ett föreskrivet scenario som ett antal deltagare följer. Sessionerna spelas in, fördelaktigen med både ljud och bild för att underlätta analys av materialet. Under tiden som de följer handlingsmallen för scenariot bes deltagaren att tänka högt. Det vill säga att de vid varje val eller handling talar om hur de hade tänkt sig att funktionen skulle fungera. Till exempel kan de säga vad de förväntar sig att en meny skulle innehålla eller vad de förväntar sig bör hända när de trycker på en knapp.

Versionen av Think-Aloud-Protocol som används för detta användbarhetstest är en anpassning av standardmetoden. Deltagarna kommer att använda en prototyp av gruppens programvara och kommer behöva informeras om detta. Om deltagaren fastnar eller tystnar under testet används neutrala uppmaningar för att driva fram fortsatta tankegångar. Fokus läggs på att undvika ledande uppmaningar och att försöka hålla dessa neutrala och tankeväckande. Videoinspelning används ej, men i andra aspekter liknar metoden originalet.

Prioritering

Vid utförandet prioriteras uppfattningen av applikationens funktion, estetik och om applikationen uppfyller sitt syfte. Då alla sessioner spelas in kommer all annan relevant information användas.

Testfall kravbaserad systemtestning

Systemtestning av funktionella Must-prioriterade krav

ID	TF1
ID Testat krav	F.L.S.M.1
Rubrik	AI-spelare ska få två kort och fatta ett beslut utifrån kortens värde
Förberedelser	Skriva testprogram som tilldelar AI-spelaren kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. AI-spelaren får två kort. 2. AI-spelaren blir ombedd av Controller att göra ett beslut. 3. AI-spelaren fattar ett beslut efter kortens värde och färg.
Förväntat resultat	AI-spelaren fattar ett beslut (call/check, fold, raise, all-in).

ID	TF2
ID Testat krav	F.L.S.M.1 & F.L.S.M.2
Rubrik	AI-spelare kan utföra beslut på alla rundor under spelets gång
Förberedelser	Skriva testprogram som tilldelar AI-Spelare kort och tar emot AI's beslut.
Teststeg	<ol style="list-style-type: none"> 1. AI-spelaren blir ombedd av kontrollern att göra ett beslut. 2. AI-spelaren kontrollerar sina tillgängliga kort (hålkort och gemensamma). 3. AI-spelaren fattar ett beslut efter kortens värde och färg.
Förväntat resultat	AI-spelaren fattar ett beslut (call/check, fold, raise, all-in).

ID	TF3
ID Testat krav	F.L.S.M.3
Rubrik	Starta spelomgång
Förberedelser	Skriv testprogram som skapar en controller och kallar på startmetod med fördefinierat antal AI-spelare.
Teststeg	1. Kör testprogram.
Förväntat resultat	Specificerat antal AI-spelare blir skapade, kortlek blir skapad och blandad, spelare och AI-spelare blir tilldelade kort. Spelomgång startar.

ID	TF4
ID Testat krav	F.L.S.M.8
Rubrik	AI-spelaren ska fatta ett beslut utifrån dess korts styrka
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåller och räknar ut dess styrka utifrån de poker regler som finns och med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelarens utvärdering av sin handstyrka har stigit eller sjunkit enligt de kort den blivit tilldelad

ID	TF5
ID Testat krav	F.L.S.M.1 & F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-Spelare ska fatta ett beslut utifrån korten Hjärter 2, Spader 4
Förberedelser	Skriva testprogram som tilldelar AI-Spelare kort och tar emot AI's beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "fold".

ID	TF6
ID Testat krav	F.L.S.M.1 & F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-användare ska fatta ett beslut utifrån korten Hjärter ess och Spader 6
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "call".

ID	TF7
ID Testat krav	F.L.S.M.1 & F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-användare ska fatta ett beslut utifrån korten Hjärter 6 och Spader 6
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "raise".

ID	TF8
ID Testat krav	F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-Spelare ska fatta ett beslut utifrån korten Hjärter 2, Spader 4, Spader 2, Klöver 2 och Ruter 3
Förberedelser	Skriva testprogram som tilldelar AI-Spelare kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "call".

ID	TF9
ID Testat krav	F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-Spelare ska fatta ett beslut utifrån korten Hjärter 2, Spader 2, Ruter 2, Klöver 4 och Ruter 3
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåll och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "raise"

ID	TF10
ID Testat krav	F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-spelaren ska fatta ett beslut utifrån korten Hjärter 2, Hjärter 3, Hjärter 4, Hjärter 5 och Hjärter 6
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåll och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "raise"

ID	TF11
ID Testat krav	F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-spelaren ska fatta ett beslut utifrån korten Hjärter 2, Hjärter 4, Hjärter 6, Hjärter 8 och Hjärter 10
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI-spelaren gör uträkning på de korten den erhåll och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "raise"

ID	TF12
ID Testat krav	F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-spelaren ska fatta ett beslut utifrån korten Hjärter 2, Spader 2, Klöver 2, Hjärter 4 och Spader 4
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI-spelaren gör uträkning på de korten den erhåll och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "raise"

ID	TF13
ID Testat krav	F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-spelaren ska fatta ett beslut utifrån korten Hjärter 2, Spader 2, Klöver 2, Ruter 2 och Hjärter 4
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåll och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "raise"

ID	TF14
ID Testat krav	F.L.S.M.2 & F.L.S.M.8
Rubrik	AI-spelaren ska fatta ett beslut utifrån korten Hjärter 2, Hjärter 3, Hjärter 4, Hjärter 5 och Hjärter 6
Förberedelser	Skriva testprogram som tilldelar AI kort och tar emot AI-spelarens beslut.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela AI-spelaren kort. 2. AI gör uträkning på de korten den erhåll och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	AI-spelaren bör ge resultatet "raise"

ID	TF15
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 2 och Spader 4
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper - "Ingenting, tyvärr..."</p> <p>Advice - "Denna hand kanske inte är den bästa att spela på..."</p> <p>pwrBar - 1</p> <p>toHighlight []</p>

ID	TF16
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter ess och Spader 6
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg

	och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper - "Ingenting, tyvärr..."</p> <p>Advice - "Du har ett högt 'HIGH CARD'. Om det är billigt så kan du prova och se."</p> <p>-pwrBar - 2</p> <p>-toHighlight []</p>

ID	TF17
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 6 och Spader 6
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper - "'ONE-PAIR' i 6:or"</p> <p>Advice - 'ONE-PAIR' på första-handen är en stark hand! Så kör på!"</p> <p>pwrBar - 4</p> <p>toHighligh - [6,H,6,S]</p>

ID	TF18
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 2, Spader 4, Spader 2 Klöver 4 och Ruter 3
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper - " 'TWO PAIRS' i 2:or och 4:or"</p> <p>Advice - " 'TWO PAIRS' är en bra hand, kör på."</p> <p>pwrBar - 4</p> <p>toHighlight - [2,H, 4,S, 2,S, 4,C]</p>

ID	TF19
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 2, Spader 2, Ruter 2, Klöver 4 och Ruter 3
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort.

Teststeg	2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper -” 'THREE OF A KIND' i 2:or”</p> <p>Advice - “'THREE OF A KIND' är en väldigt stark hand. Kör på! Fundera även på att höja!”</p> <p>pwrBar - 4 toHighlight - [2,H, 2,S, 2,C]</p>

ID	TF20
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 2, Hjärter 3, Hjärter 4, Hjärter 5 och Spader 6
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper - “En 'STRAIGHT'!! Du har 5/5.”</p> <p>Advice -” En 'STRAIGHT' är en riktigt bra hand. Kör på! Fundera även på att höja!”</p> <p>pwrBar - 4 toHighlight - [2,H, 3,H, 4,H, 5,H, 6,D]</p>

ID	TF21
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 2, Hjärter 4, Hjärter 6, Hjärter 8 och Hjärter 10
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper -” En 'FLUSH' i hjärter!! Du har 5/5!!”</p> <p>Advice - “Du har en 'FLUSH'! Kör på, din hand är svår att slå!”</p> <p>pwrBar - 4</p> <p>toHighlight - [2,H, 4,H, 6,H, 8,H, 10,H]</p>

ID	TF22
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 2, Spader 4, Klöver 2, Hjärter 4 och Spader 2

Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper - "FULL HOUSE' med 2:or och 4:or!!"</p> <p>Advice -" Det är inte mycket som slår denna hand! Höja är rekommenderat!"</p> <p>pwrBar - 4</p> <p>toHighlight - [2,H, 4,H, 2,C, 2,S, 4,S]</p>

ID	TF23
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 2, Spader 2, Klöver 2, Ruter 2 och Hjärter 4
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
	<p>Helper - "'FOUR OF A KIND' i 2:or."</p> <p>Advice - "Inte mycket om slår denna hand.</p>

Förväntat resultat	<p>Kör på och fundera på att höja!.”</p> <p>-pwrBar - 4</p> <p>-toHighligh [2H, 2S, 2C, 2D]</p>
--------------------	---

ID	TF24
ID Testat krav	F.B.A.M.2 & F.L.S.M.9 & F.L.S.M.10
Rubrik	Hand-klassen ska räkna ut vad spelaren har och vilket råd den ska ge spelaren för korten Hjärter 2, Hjärter 3, Hjärter 4, Hjärter 5 och Hjärter 6
Förberedelser	Skriva testprogram som tilldelar Hand-klassen kort och tar emot Hand-klassens beslut och råd.
Teststeg	<ol style="list-style-type: none"> 1. Tilldela Hand-klassen kort. 2. Hand-klassen gör uträkning på de korten den erhåller och räknar ut dess styrka med hjälp av kortens färg och värde, och därefter ger ett resultat.
Förväntat resultat	<p>Helper -” 'STRAIGHT FLUSH' i färgen hjärter och i stegen FIXA TEMP!”</p> <p>Advice -” 'STRAIGHT FLUSH' är en grym hand! Kör på och höj!”</p> <p>-pwrBar - 4</p> <p>-toHighligh [2H, 3H, 4H, 5H, 6H]</p>

ID	TF25
ID Testat krav	IF.B.1
	Programmet är skrivet i Java

Rubrik	
Förberedelser	Titta på koden.
Teststeg	<ol style="list-style-type: none"> 1. Kontrollera klasser. 2. Kör klasserna i en Java kompilator.
Förväntat resultat	Koden går att skriva ut i en Java kompilator.

ID	TF26
ID Testat krav	IF.B.4
Rubrik	Programmets grafiska miljö är JavaFX
Förberedelser	Titta på koden.
Teststeg	<ol style="list-style-type: none"> 1. Kontrollera att Main-klassen ärver Application. 2. Kör programmet och kontrollera så att FXML-dokumentet laddas korrekt samt resterande kodfiler som använder JavaFX körs.
Förväntat resultat	Programmet körs med hjälp av FXML-dokument utan problem.

ID	TF27
ID Testat krav	F.B.A.M.1 & F.B.S.M.1-A.M.1
Rubrik	Visa användaren kort
Förberedelser	Kör applikationen. Välj 'Ladda spel' i

	huvudmenyn, skriv in användarnamn, tryck 'Starta spel'. Välj 'Ja' på frågan om användaren är redo att spela poker.
Teststeg	1. Titta på skärmen.
Förväntat resultat	Två kort syns.

Spårningsmatris

Matriserna är utformade efter de olika kategorierna för kraven (funktionella och kvalitativa). I de fall där kraven är prioriterade efter MoSCoW-modellen [15] är även matriserna detta. Ett X i matrisen indikerar att kravet testas av testfallet.

Spårningsmatris över funktionella Must-prioriterade krav

ID-Test fall	F.L.S. M.1	F.L.S. M.2	F.L.S. M.3	F.L.S. M.8	F.B.A. M.1	F.B.A. M.2	F.L.S. M.9	F.L.S. M.10	F.B.S.M.1 -A.M.1
TF1	X								
TF2	X	X							
TF3			X						
TF4				X					
TF5	X	X		X					
TF6	X	X		X					
TF7	X	X		X					
TF8		X		X					
TF9		X		X					
TF10		X		X					
TF11		X		X					
TF12		X		X					
TF13		X		X					
TF14		X		X					
TF15						X	X	X	
TF16						X	X	X	
TF17						X	X	X	
TF18						X	X	X	
TF19						X	X	X	
TF20						X	X	X	
TF21						X	X	X	
TF22						X	X	X	
TF23						X	X	X	
TF24						X	X	X	
TF27					X				X

Spårningsmatris över kvalitativa krav

ID-Testfall	IF.B.1	IF.B.4				
TF25	X					
TF26		X				

Granskningsprotokoll

Granskningsprotokoll dokumentgranskning DG.PP.1

Granskningsprotokoll från dokumentgranskningen av projektplan finns bifogat separat som en pdf-fil ([DG.PP.1.Rapport_Projektplan_v.1.0.pdf](#)).

Granskningsprotokoll dokumentgranskning DG.KD.1

Granskningsprotokoll från dokumentgranskningen av kravdokument finns bifogat separat som en pdf-fil ([DG.KD.1.Rapport_Kravdokument_v.1.0.pdf](#)).

Granskningsprotokoll dokumentgranskning DG.K.1

Granskningsprotokoll från dokumentgranskningen av krav finns bifogat separat som en pdf-fil ([DG.K.1.Rapport_Krav_v.1.0.pdf](#)).

Granskningsprotokoll dokumentgranskning DG.VoV.1

Granskningsprotokoll från dokumentgranskningen av verifiering och valideringsdokumentet finns bifogat som en pdf-fil ([DG.VoV.1.Rapport_VoVdokument_v.1.0.pdf](#)).

Granskningsprotokoll kodgranskning KG.1

Rapport från kodgranskningsmöte						
Granskad artefakt:	Kod	Titel av paket: aiClass/hand/deck/filehandler/controller		Datum (åå-mm-dd):	17-04-11	Tidpunkt: 10:30-12:30
Närvarande:	Max	Lykke	Vedrana	Rikard	Amin	
Roll:	Author/Scribe	Scribe/Recorder	Scribe	Scribe	Moderator	
Avvikelser						
Påträffat fel:	Klass och kod rad:	Typ:	Allvar:	Ansvarig:	Rekommendationer:	
Svenska kommentarer	Interface AICalculations	Språk	2	Max	Svenska eller engelska?	
Liten bokstav klassnamn Hand	Hand	Språk	3	Max	Klassens namn ska skrivas med stor bokstav	
Svensk hjälp	Hand	Språk	5	Max	Hjälpstexten ska vara på svenska	
White space fel	Alla	Struktur	1	Max	Dubbelkolla all white space i alla klasser. 2 rader white space efter metoder.	
Stor bokstav variabelnamn Card	hand	Struktur	2	Max	Ändra till liten bokstav Card	
Mellanrum mellan import	hand	Struktur	1	Max	Ta bort mellanrum mellan importer	
Ej javadoc	AI	Dokumentation	3	Max	Lägg till Javadoc (klass och metoder)	
Ej javadoc	TurnOne	Dokumentation	3	Max	Lägg till Javadoc (klass och metoder)	
Ej javadoc	TurnTwo	Dokumentation	3	Max	Lägg till Javadoc (klass och metoder)	
Ej javadoc	TurnThree	Dokumentation	3	Max	Lägg till Javadoc (klass och metoder)	
Ej javadoc	TurnFour	Dokumentation	3	Max	Lägg till Javadoc (klass och metoder)	
Ej javadoc	Hand	Dokumentation	3	Max	Lägg till Javadoc (klass och metoder)	
Ej javadoc	Calc	Dokumentation	3	Max	Lägg till Javadoc (klass och metoder)	
Ej javadoc	AICalculations	Dokumentation	3	Max	Lägg till Javadoc (klass och metoder)	
Variabler ej deklarerad för sig själv	TurnTwo, rad 350-351	variabler	4	Max	Separera variabeldeklarerationer	
Klasstorlek stor	Calc	Grundläggande	3	Max	Lägg help-metoden i egen klass	
Formatering saknas	Alla	Kod formatering	3	Max	Formatering behövs	
Bortkommenterad kod, stubbar	Alla	Struktur	4	Max	All testkod ska bort, byta namn på det som faktiskt används som är gammal te	
Upprepade metoder	TurnOne	Struktur	3	Max	Metoder upprepas i olika klasser, gård det att flytta ut koden som sedan kallas p	
Upprepade metoder	TurnTwo	Struktur	3	Max	Metoder upprepas i olika klasser, gård det att flytta ut koden som sedan kallas p	
Upprepade metoder	TurnThree	Struktur	3	Max	Metoder upprepas i olika klasser, gård det att flytta ut koden som sedan kallas p	
Upprepade metoder	TurnFour	Struktur	3	Max	Metoder upprepas i olika klasser, gård det att flytta ut koden som sedan kallas p	
Kommentarer fattas	Alla	Dokumentation	4	Max	Kommentera metoder så det är tydligt vad syftet är med metoderna	
Liten bokstav i variabelnamn	TurnOne	Variabler	3	Max	Ändra aiPotChance, stort P i pot.	
Otydligt variabelnamn	Alla	Variabler	3	Max	Ändra 'diff' till 'difference' för ökad tydlighet	
Gulningar finns	Alla kontrollera	Variabler	4	Max	Kontrollera oanvända variabler efter test	
White space fel	Card	Struktur	3	Lykke	Dubbelkolla all white space. 2 rader white space efter metoder.	
Filehandler paket stor bokstav fel	FileHandler	Struktur	3	Amin	Filehandler paket ska döpas om - litet f i början	
White space fel	FileHandler, rad 19-20	Struktur	1	Amin	lägg till white space	
White space fel	CardValue	Struktur	3	Vedrana	Lägg till white space	
Ej javadoc	CardValue	Dokumentation	3	Vedrana	Lägg till Javadoc (klass och metoder och author)	
White space fel	Deck	Struktur	1	Vedrana	Lägg till white space	
Ej javadoc	Deck	Dokumentation	3	Vedrana	Lägg till Javadoc (klass och metoder och author)	
White space fel	Suit	Struktur	1	Vedrana	Lägg till white space	
Ej javadoc	Suit	Dokumentation	3	Vedrana	Lägg till Javadoc (klass och metoder och author)	
White space fel	SPController, rad 6	Struktur	1	Rikard	White space överflöd	
White space fel	SPController	Struktur	1	Rikard	Lägg till white space, kontrollera hela klassen	
Ej javadoc	SPController	Dokumentation	3	Rikard	Lägg till Javadoc (klass och metoder och author)	

Granskningsprotokoll kodgranskning KG.2

Rapport från kodgransningsmöte						
Granskad artefakt:	kod	Titel av paket: gui/controller		Datum (åå-mm-dd): 17-05-18	Tidpunkt: 12:00-14:00	
Närvarande:	Max	Lykke	Vedrana	Rikard	Amin	
Roll:	Scribe	Scribe/Author	Scribe/Author	Recorder/Author	Moderator/Author	
Avvikelser						
Påträffat fel:	Klass och kod rad:	Typ:	Allvar:	Ansvarig:	Rekommendationer:	
white space fel	SpController, lite överallt	struktur	1	Rikard	Dubblikolla all white space i alla klasser. 2 rader white space efter metoder.	
Kommentarer fattas	SpController, lite överallt	Struktur	3	Rikard	JavaDoc finns men inte förklarande kommentarer.	
Finns två oimplementerade variabler	private int[] potSplits och private boolean	Struktur	2	Rikard	Kontrollera och/eller ta bort.	
Ej fullständig	PotSplitt fattas	Struktur	4	Rikard	PotSplitt ska finnas	
Syso i kod	Spcontroller, Rad 315	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 316	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 317	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 328	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 329	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 330	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 331	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 332	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 333	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 343	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 352	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Rad 376	Struktur	1	Rikard	Ta bort testSysosar.	
Syso i kod	Spcontroller, Hela klassen	Struktur	1	Rikard	Ta bort testSysosar.	
Run-metoden stor	public void run()	Struktur	1	Rikard	Kan metoden delas upp?	
IF-ELSEIF kontroll	run-metoden, if (currentPlayer == 0)	Loopar och if-satsar	1	Rikard	Player kontrolleras först, ska AI (där fallen sker mer ofta) kontrolleras först i if-satsen	
Whitespace felaktig	GameController, Rad 140	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
Whitespace felaktig	GameController, Rad 152	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
Whitespace felaktig	GameController, Rad 267	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
Whitespace felaktig	GameController, Rad 330	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
Whitespace felaktig	GameController, Rad 425	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
Whitespace felaktig	GameController, Rad 550	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
Whitespace felaktig	GameController, Rad 820	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
Whitespace felaktig	GameController, Rad 1033	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
Kontrollera mellanrum mellan metodnamn	GameController	Riktlinjer	2	Amin	Fixa det enligt riktlinjer.	
JavaDoc fel paramnamn	GameController, setLabelUIAIBara	Riktlinjer	3	Vedrana	Parameter står namn, ska vara action.	
Stor bokstav variabelnamn	GameController, private AnchorP	Riktlinjer	2	Amin	Variabelnamn liten bokstav	
JavaDoc läggs till på alla metoder	GameController, alla metoder som är public	Riktlinjer	4	Författare till metoden	Skriv javadoc!	
Syso i kod	GameController, överallt	Struktur	3	Författare till metoden	Ta bort onödiga sysos	
Ej konsekventa variabelnamn (label)	GameController, överst i kodfilen	Variabler	2	Amin	Gör enhetlig	
Ej använd variabel	GameController, klassvariabler, ca	Variabler	2	Amin	Ta bort cardOne.	

Testrapporter

Testrapport Kravbaserad systemtestning TR.KS

Id testfall	datum	Kod/dokument version	Utfört av	Resultat
TF1	17-04-12	AI v.1.0 Deck v.1.0 Card v.1.0	Max Frennessen	AI tar emot kort och gör en uträkning som följd av tilldelade kort. Uträkning kan ses som <u>ok</u> .
TF2	17-04-12	Controller v.0.5 AI v.1.0 Deck v.1.0 Card v.1.0	Max Frennessen	AI fattar bestämmer ett beslut mellan (call/check, fold, raise, all-in). Och skickar det till controller. <u>Konflikt</u> mellan controller och AI då controller söker efter "call" och AI ger "bet".
TF3	17-04-12	Controller v.0.5 AI v.1.0 Player v.0.5 Deck v.1.0 Card v.1.0	Rikard Almgren	Spelomgången startar och första rundan spelas.
TF4	17-04-12	AI v.1.0 Deck v.1.0 Card v.1.0	Max Frennessen	AI ändrar och anpassar sig korrekt till de korten den får in. <u>Balansering</u> behövs.
TF25	17-04-12	Alla källkodsfiler.	Max Frennessen	Koden är skriven i Java.
TF26	17-04-21	Alla källkodsfiler i paketet gui samt FXML-filerna placerade i resources.	Lykke Levin	Programmets grafiska miljö är JavaFX.
TF27	17-04-21	GameController GameState.fxml	Lykke Levin	GUI:t visar användarens kort.

Testrapport White-box testning TR.WB**AI-testning**

Följande testfall nedan utnyttjar följande filer, rader och testkod. Samtliga är i version 1.5. Ifall resultat skiljer sig från det förväntade. Är detta markerat med rött och sist under alla testfall beskrivs det kortfattat om vad som gick fel.

Fil	Rader	Teskod	Version
Ai.java AiCalculation TurnOne.java TurnTwo.java TestAi.java	39-86 26-267 33-178 34-182 38-109	TestAi.java	1.5

Testfall	Kortstyrka	Korten	Förväntat	Resultat
TF5	High card - low	2H,4S	fold	Call - var bluff(15% chans) Andra försök - fold
TF6	High card - high	14H, 6S	call	call
TF7	Par	6h,6s	call	call
TF8	Two pair	2H, 4S, 2S, 4C, 3D	call	call
TF9	Three of a kind	2H, 2S , 2D,4C, 3D	raise	raise
TF10	Straight	2H, 3H, 4H,5H, 6S	raise	raise
TF11	Flush	2H, 4H, 6H, 8H, 10H	raise	call
TF12	Full house	2H, 2S, 2C,4H,4S	raise	raise
TF13	Four of a kind	2H, 2S, 2C, 2D,4H	raise	raise
TF14	Straight flush	2H, 3H, 4H, 5H, 6H	raise	raise

Problematik:

TF11 - Handen är så pass bra att en raise bör vara resultatet.

TF12 - Handen ses inte som en full house utan får ett resultat om att det är en triss bara.

Hand-klass testning

Följande testfall nedan utnyttjar följande filer, rader och testkod. Samtliga är i version 1.5. Ifall resultat skiljer sig från det förväntade. Är detta markerat med rött och sist under alla testfall beskrivs det kortfattat om vad som gick fel.

Fil	Rader	Teskod	Version
Hand.java HandCalculation.java TestHand.java	26-79 57-786 26-75	TestHand.java	1.5

Testfall	Kortstyrka	Korten	Förväntat	Resultat
TF15	High card - low	2H,4S	Helper - "Ingenting, tyvärr..." Advice - "Denna hand kanske inte är den bästa att spela på..." -pwrBar - 1 -toHighlight []	Helper - "Ingenting, tyvärr..." Advice - "Du har bara 'HIGH CARD'. Om det är billigt så kan du prova och se." -pwrBar - 2 -toHighlight[]
TF16	High card - high	14H, 6S	Helper - "Ingenting, tyvärr..." Advice - "Du har ett högt 'HIGH CARD'. Om det är billigt så kan du prova och se." -pwrBar - 2 -toHighlight []	Helper - "Ingenting, tyvärr..." Advice - "Du har ett högt 'HIGH CARD'. Om det är billigt så kan du prova och se." -pwrBar - 2 -toHighlight []
TF17	Par	6h,6s	Helper - "'ONE-PAIR' i 6:or" Advice - "'ONE-PAIR' på första-handen är en stark hand! Så kör på!" 	Helper - "'ONE-PAIR' i 6:or" Advice - "'ONE-PAIR' på första-handen är en stark hand! Så kör på!"

			pwrBar - 4 toHighlight - [6,H,6,S]	pwrBar - 4 toHighlight - [6,H,6,S]
TF18	Two pair	2H, 4S, 2S, 4C, 3D	<p>Helper - " 'TWO PAIRS' i 2:or och 4:or"</p> <p>Advice - " 'TWO PAIRS' är en bra hand, kör på."</p> <p>pwrBar - 4 toHighlight - [2,H, 4,S, 2,S, 4,C]</p>	<p>Helper - " 'TWO PAIRS' i 2:or och 4:or"</p> <p>Advice - " 'TWO PAIRS' är en bra hand, kör på."</p> <p>pwrBar - 4 toHighlight - [2,H, 4,S, 2,S, 4,C]</p>
TF19	Three of a kind	2H, 2S , 2D,4C, 3D	<p>Helper - " 'THREE OF A KIND' i 2:or"</p> <p>Advice - " 'THREE OF A KIND' är en väldigt stark hand. Kör på! Fundera även på att höja!"</p> <p>pwrBar - 4 toHighlight - [2,H, 2,S, 2,C]</p>	<p>Helper - " 'THREE OF A KIND' i 2:or"</p> <p>Advice - " 'THREE OF A KIND' är en väldigt stark hand. Kör på! Fundera även på att höja!"</p> <p>pwrBar - 4 toHighlight - [2,H, 2,S, 2,C]</p>
TF20	Straight	2H, 3H, 4H,5H, 6S	<p>Helper - "En 'STRAIGHT'!! Du har 5/5."</p> <p>Advice - "En 'STRAIGHT' är en riktigt bra hand. Kör på! Fundera även på att höja!"</p> <p>pwrBar - 4 toHighlight - [2,H, 3,H, 4,H, 5,H, 6,D]</p>	<p>Helper - "En 'STRAIGHT'!! Du har 5/5."</p> <p>Advice - "En 'STRAIGHT' är en riktigt bra hand. Kör på! Fundera även på att höja!"</p> <p>pwrBar - 4 toHighlight - [2,H, 3,H, 4,H, 5,H, 6,D, 2,H, 3,H, 4,H, 5,H, 6,D]</p>

TF21	Flush	2H, 4H, 6H, 8H, 10H	<p>Helper -” En 'FLUSH' i hjärter!! Du har 5/5!!”</p> <p>Advice - “Du har en 'FLUSH'! Kör på, din hand är svår att slå!”</p> <p>pwrBar - 4 toHighlight - [2,H, 4,H, 6,H, 8,H, 10,H]</p>	<p>Helper -” En 'FLUSH' i hjärter!! Du har 5/5!!”</p> <p>Advice - “Du har en 'FLUSH'! Kör på, din hand är svår att slå!”</p> <p>pwrBar - 4 toHighlight - [2,H, 4,H, 6,H, 8,H, 10,H]</p>
TF22	Full house	2H, 4S, 2C,4H, 2S	<p>Helper - “'FULL HOUSE' med 2:or och 4:or!!”</p> <p>Advice -” Det är inte mycket som slår denna hand! Höja är rekommenderat!”</p> <p>pwrBar - 4 toHighlight - [2,H, 4,H, 2,C, 2,S, 4,S]</p>	<p>Helper - “'FULL HOUSE' med 2:or och 4:or!!”</p> <p>Advice -” Det är inte mycket som slår denna hand! Höja är rekommenderat!”</p> <p>pwrBar - 4 toHighlight - [2,H, 4,H, 2,C, 2,S, 4,S]</p>
TF23	Four of a kind	2H, 2S, 2C, 2D,4H	<p>Helper - " 'FOUR OF A KIND' i 2:or."</p> <p>Advice - “Inte mycket som slår denna hand. Kör på och fundera på att höja!.”</p> <p>-pwrBar - 4 -toHighlight [2H, 2S, 2C, 2D]</p>	<p>Helper - “'Ingenting, tyvärr..’.”</p> <p>Advice - “Du har bara 'HIGH CARD'. Om det är billigt så kan du prova och se.”</p> <p>pwrBar - 4 toHighlight - [2,S, 2,H]</p>
TF24	Straight flush	2H, 3H, 4H, 5H, 6H	<p>Helper -” 'STRAIGHT FLUSH' i färgen hjärter och i stegen 2-6”</p> <p>Advice -” 'STRAIGHT FLUSH' är en grym hand! Kör på och höj!”</p> <p>-pwrBar - 4</p>	<p>Helper -” 'STRAIGHT FLUSH' i färgenhjärter och i stegen FIXA TEMP!”</p> <p>Advice - “'STRAIGHT FLUSH' är en grym hand! Kör på och höj!”</p> <p>pwrBar - 4 toHighlight - [2,H, 3,H,</p>

			-toHighlight [2H, 3H, 4H, 5H, 6H]	4,H, 5,H, 6,H, 2,H, 3,H, 4,H, 5,H, 6,H]
--	--	--	-----------------------------------	---

Problematik:

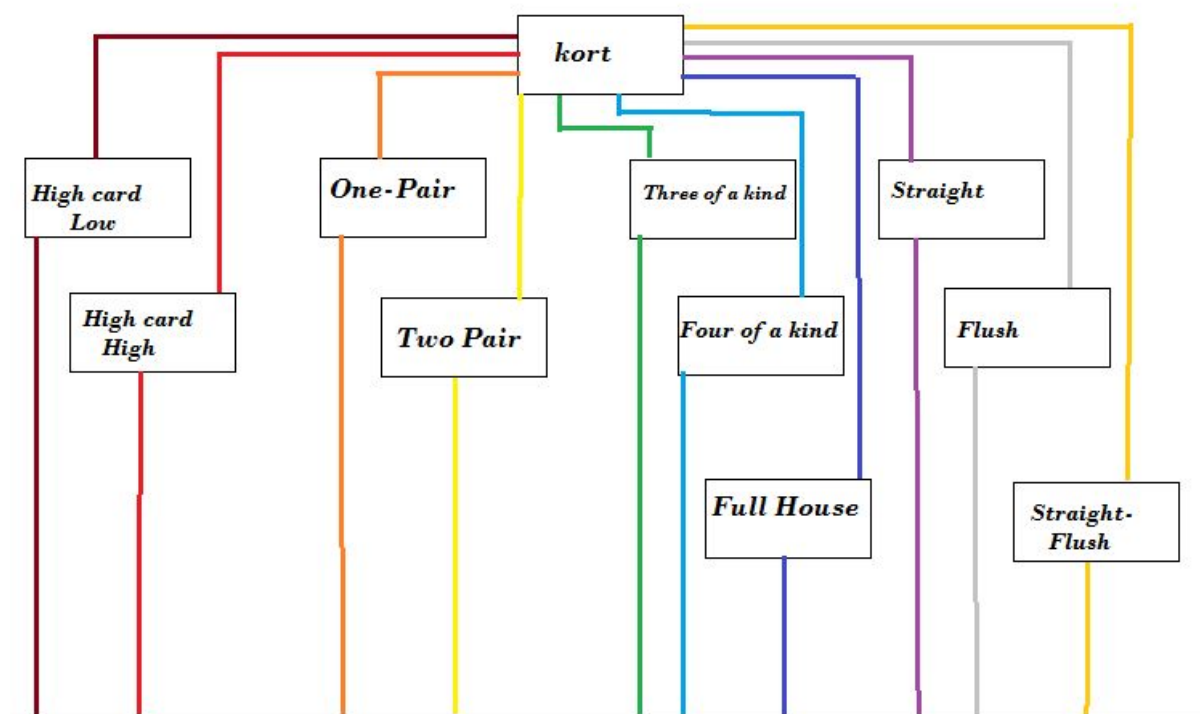
TF20: De kort som ska till toHighlight har blivit dubblerade.

TF22: Visas endast som en 'Full House' om man har ett av varje kort valör på hand. Annars visas endast triss.

TF23: Koden ser det som en dålig hand. Visar inte ens att man har 'Four of a kind'.

TF24: "färgenhjärter" sitter ihop, ska vara mellanrum mellan orden. Kort i toHighlight är dubblerade. "FIXA TEMP!" ska ersättas med vad man har i stegen ex 2-6.

Visualisering på hur path coverage har gått tillväga.



Alla val har blivit gjorda och 100% coverage har blivit uppnådd.

Testrapport Användarbarhetstestning - Think-Aloud-Protocol, TR.TAP.AB.

Scenariot som låg som bas för sessionerna finns bifogat under bilagor.

Fem Think-Aloud-Protocol sessioner utfördes och det deltagarna delade med sig av under tiden de testade sig igenom applikationen är vad denna testrapport baseras på.

Information som framkom under sessionerna:

Endast 2 av de 5 deltagarna tyckte sig känna till Texas Hold'em poker, något som visade sig bli ett smärre problem senare.

Då applikationen vid testningstillfället saknade ett par av huvudfunktionerna, primärt tutorial samt regelboken så blev tyvärr resultaten som förväntat.

2 utav 5 deltagare förväntade sig att spelet skulle börja direkt vid tryck på "Nytt spel", men de andra 3 trodde att en inställningsmeny skulle dyka upp. Ingen tyckte att inställningsmenyn var onödig utan samtliga ansåg att den var rimligt omfattande.

Samtliga deltagare förväntade sig att man skulle få upp en meny för att ladda spel vid klick på "Ladda Spel", tre antog att det inte hände något för att man inte ännu hade spelat något, medan två tyckte att det borde bli ett felmeddelande om det inte finns något att ladda.

Samtliga deltagare förväntade sig att en tutorial skulle dyka upp när tutorial-checkboxen var iklickad. Samtliga deltagare också tyckte att det var missvisande både att marknadsföra applikationen som ett program för att lära ut poker när det inte fanns någon tutorial, samt att valet för tutorial fanns när det inte gjorde något.

Väl inne i spelet så visade det sig snabbt att avsaknaden av tutorial och regelbok var ett problem för de som inte var insatta i pokertermer. De 3 deltagare som inte kände till reglerna och pokertermerna testade sig fram med knapparna.

Call:

Alla deltagare gissade korrekt att man var delaktig i rundan när man valde call, men endast 2 förstod att man betalade för att vara kvar i rundan. Att översätta till svenska pokertermer hjälpte inte.

Check:

3 deltagare trodde att man precis som call fortsatte vara med i rundan, men en av de tre förstod inte varför knappen bytte namn.

De återstående två deltagarna trodde att man kunde se AI-spelarnas kort genom att trycka på check. De tyckte att det hade varit användbart i en applikation för att lära sig poker.

Fold:

Tre deltagare översatte ordet till svenska och sa att man vek sig och inte var kvar i rundan. De tyckte dock inte att det var tydligt att man inte stannade kvar i rundan då korten fortsatte delas ut och resultatet blev upplöst.

Två deltagare trodde att man var kvar i rundan, men hade ingen specifik anledning för varför de trodde det.

Raise och raisebar:

Fyra deltagare trodde korrekt att raise betydde att man höjde mängden pengar som man spelade för, dock så var avsaknaden på en synlig bordspott ett problem som samtliga deltagare tyckte var otydligt.

En deltagare trodde att man kunde få bättre kort genom att betala med raise.

Regel, ljud och sparknappar:

Endast en deltagare uppmärksammade att knapparna fanns, denna påpekade att regel-knappen inte gjorde något, men trodde att den skulle ta upp en inställningsmeny. Samma deltagare trodde att ljud-knappen skulle sätta igång bakgrundsmusiken igen och att spar-knappen skulle spara spelet.

De andra 4 deltagarna gissade korrekt vad meningen med knapparna var när de blev visade att de fanns där. Dock tyckte de att saker som inte var färdiga inte bör synas för en användare.

Meny:

Återigen så uppmärksammade bara en deltagare att det fanns en vanlig meny, dock inte samma deltagare som upptäckte regel, ljud och spar-knappen.

Denna deltagare, samt de andra deltagarna efter visning, antog korrekt vad alla val förutom hjälp gjorde eller var ämnade till. Samtliga deltagare hoppades på att hjälpen som annars saknades i applikationen skulle dyka upp vid val av hjälpmenyn. Återigen så upprepades att ”Om den inte gör något så ska den inte vara där”.

Generellt spelrunda:

Samtliga deltagare påpekade att AI-spelare lyste upp när det var deras tur men påpekade också att det inte var tydligt när det var spelarens egna tur, även om två av deltagarna la märke till att ens knappar kom tillbaka först när det var ens tur.

Samtliga deltagare fann hjälprutan positiv och tyckte att den var trevlig och hjälpsam. Högra hjälpfältet med kortrankningen fick inte mycket uppmärksamhet. En deltagare tyckte inte om att behöva scrolla. Ingen deltagare upptäckte att det gick att fälla in dessa rutor. Ingen deltagare förväntade sig att det skulle gå att göra.

Analys av resultat:

Som tidigare sagt, då applikationen vid testningstillfället saknade ett par av huvudfunktionerna, primärt tutorial samt regelboken så blev tyvärr resultaten som förväntat med ett par undantag.

Större delen av applikationen uppfattades som otydlig och dåligt förklarad av de som inte hade tidigare kunskap, men samtliga deltagare tyckte att applikationen såg bra ut och att om man fick knapparna och spelrundan förklarade så hade deras generella intryck varit mycket bättre.

Call, Check, Fold, Raise, samt menyknappar:

Valet av engelska pokertermer verkade vara ett problem men i uppföljningen till testningen så svarade majoriteten av deltagarna att de inte visste kunde de motsvarande svenska termerna. Problemet var snarare att de inte visste hur spelet fungerade regelmässigt och att en förklaring saknades helt från applikationen.

Slutsatser:

Följande slutsatser har dragits:

- spelet var icke-intuitivt i upplägget och brister i den lärande delen av applikationen gjorde spelet svårare att förstå än nödvändigt.
- rätt ordning av funktionsimplementering är viktig vid användbarhetstestning. Alla bakomliggande funktioner kanske inte behövdes men den förklarande delen hade troligen gett bättre resultat.
- användandet av en ofärdig hi-fi prototyp kan ha försvårat det för deltagare att förstå olika steg. En lo-fi prototyp med en snabb tutorial hade troligtvis gett bättre resultat.
- en snabb förklaring hade hjälpt de utan vana att lära sig.
- Think-aloud-protocol utan förklarande steg i en ofärdig applikation var inte givande. En annan metod borde ha valts.

Testrapport Användarbarhetstestning - Semistrukturerad Intervju, TR.SI.AB.1

Frågeformuläret som intervjun utgick från finns bifogat under bilagor. En av röstinspelningarna (*S.I.1 #1*) har dessutom transkriberats och även denna finns att hitta under bilagor (*Semistrukturerad intervju #1 (intervjufrågor)*).

Totalt intervjuades 16 personer. Utav dessa spelades fyra intervjuer in under intervjuens gång. Under resterande intervjuer valde man att anteckna intervju svar istället. De antecknade intervjuerna finns dock också inspelade, fast då återberättade av intervjuaren (utifrån anteckningarna). Dessa inspelningar har tillsammans med anteckningarna legat som grund till nedanstående analys.

Information som framkom under intervjuerna

Utav 16 respondenter så var det 7 personer som ansåg sig veta hur man spelar Texas Hold'em Poker. 5 personer svarade att man visste på ett ungefär eller benämnde sig själva som "nybörjare". 3 personer svarade att de inte visste hur man spelar Texas Hold'em Poker. En person svarade att denne endast kände till vanlig poker, inte Texas Hold'em.

Av dem som kunde poker så hade nästintill samtliga lärt sig kortspelet genom att spela det med vänner eller familj. Endast en person hade lärt sig spelet via nätet. En annan person hade läst en bok om poker och lärt sig spelet på så sätt. De som inte kunde spela poker hade aldrig heller någon gång försökt lära sig spela kortspelet.

Vid frågan om hur lång tid (uppskattningsvis) det tog innan de (nu) pokerkunniga respondenterna (då) kände sig bekväma i spelets regler, svarade de att de förstod reglerna efter "1-2 sessioner", "3-5 spelomgångar" eller "en spelkväll". Ett flertal underströk att själva grunden till hur spelet går till inte var särskilt svårt att lära sig. Däremot tar det lång tid innan man faktiskt blir duktig på poker och behärskar en bra spelstrategi.

Endast två av respondenterna hade försökt lära ut spelet till någon annan. Den ena tyckte att det generellt hade gått bra och hade inte heller stött på några direkta svårigheter med att förklara vissa spelmoment. Däremot hade det tagit lång tid. Den andra respondenten pekade ut själva bettingrundorna som svårförklarade, och att det tog tid innan den andra personen förstod när man skulle call:a gentemot raise:a.

En av intervjufrågorna var baserad på ord-association. Syftet med frågan var att kontrollera om respondenterna främst kopplade ordet "poker" till engelska termer eller svenska termer. Endast två av totalt 16 respondenter nämnde *endast* svenska ord. Resterande respondenter nämnde alla engelska ord, främst gällande specifika pokertermer som exempelvis "raise", "royal straight flush", "dealer", och "river".

Vidare frågades respondenterna om förväntningarna på applikationen rent språkmässigt; skulle applikationen vara på svenska och engelska? Fanns det tillfällen då det ansågs vara

okej att blanda de två språken? En person var obrydd om vilket språk som valdes, men tyckte att man skulle vara konsekvent med språket rakt igenom hela applikationen. En annan person (troligtvis den respondent som överlag hade störst pokervana) ansåg att applikationen borde vara på engelska och såg inte behovet av svenska överhuvudtaget. Tre personer ville ha applikationen på endast svenska. Övriga 11 respondenter var alla positivt inställda till att blanda språken, där engelskan skulle komma att användas till traditionella pokertermer och svenskan skulle ligga till grund för själva undervisningen (hjälpuror med text samt applikationen i övrigt). En utav dessa berättade att hans vän som jobbar på kasinot här i Malmö sagt att de engelska pokertermerna brukas även där. Många påpekar också att poker är ett internationellt kortspel och tycker därför att de engelska pokertermerna väger tyngre än de svenska.

Respondenterna ombads försöka föreställa sig applikationen och sedan berätta på vilket sätt de ville motta hjälp från applikationen. Många tyckte att det var viktigt att tydligt informera användaren om vad denne har i handen samt visa om användaren faktiskt har en handkombination. Tips om kommande möjliga handkombinationer, utifrån nuvarande hand, föreslogs också. Ett flertal ville se någon form av sannolikhetskala på att få en viss kortkombination eller möjligtvis någon form av mätare som visade den nuvarande handens styrka. Hjälpuror med vägledande text ansågs nödvändiga. Hover-funktion med extra informationstext föreslogs vid svårförklarade funktioner. En tutorial innan spelets början ansågs vara bra. Många var positivt inställda till att ha regler och information om handrankningar tillgängligt.

Senare ombads respondenterna komma med egna förslag på olika funktioner som de hade velat ha i applikationen. Tre personer tyckte att man borde kunna backa tillbaka ett steg i spelet, för att kunna återvända till tillfällena då man missförstått något. Fyra personer föreslog att det skulle gå att stänga av hjälpfunktionerna, för att kunna testa om man faktiskt hade lärt sig spela utan hjälp. En person föreslog möjligheten till en *“snabbrunda”*, där användaren skulle slippa invänta sin tur (AI-spelarnas drag skulle alltså inte animeras för att verka realistiska). En annan person föreslog möjlighet till turneringsspel för att göra spelet roligare, men också för att lära användaren att spela strategiskt. Samma person ansåg att information om olika typer av spelstilar (exempelvis att spela konservativt gentemot att spela aggressivt) och vad dessa innebär hade varit bra. Någon önskade att AI-spelarna skulle ha olika personligheter/spelstilar.

9 respondenter tyckte att en spara-funktion hade varit bra. Resterande tyckte att det var meningslöst.

13 respondenter gillade idén om att som användare kunna välja antal AI-spelare att spela mot. De övriga tre såg ingen poäng med det (dessa tre hade antingen ingen erfarenhet av poker eller benämnde sig som *“nybörjare”*).

6 respondenter ville kunna ändra pottstorleken vid spelets början. 10 tyckte antingen att det inte spelade roll för deras del eller att det var en onödig funktion.

Vad det gällde ljud ansåg en klar majoritet att effektljud var önskvärda. Ett flertal av dessa påpekade också att effektljud skulle kunna fungera som en bra typ av feedback för nybörjare. När det kom till bakgrundsmusik var samtliga rörande överens om att det skulle undvikas, och om det fanns så skulle det gå att stänga av. Överlag ville majoriteten ha en funktion för att slå på/stänga av ljud, oavsett om det gällde effektljud eller bakgrundsmusik.

Analys av resultat

De valda respondenterna tillhörde ett ganska brett spektrum gällande pokervana. Man hade kunnat anse att fler av respondenterna borde tillhört gruppen som inte kunde spela poker eller åtminstone ansåg sig vara *“nybörjare”*, med tanke på att applikationen har dessa som tänkt målgrupp. Det går dock inte att bortse från faktumet att de med pokervana har ett helt annat perspektiv än de utan, och att deras feedback är minst lika viktig samt går att tillämpa på applikationen. Under intervjuerna framgick det många gånger att de utan pokervana inte hade samma konsekvenstänk när de besvarade vissa frågor. De hade svårt att förstå syftet med vissa funktioner, och i vilka avseenden dessa hade kunnat vara bra.

De flesta hade lärt sig spela kortspelet tillsammans med närstående. Detta antyder att de flesta lär sig spelet samtidigt som de spelar, vilket applikationen också kommer att försöka återspegla.

Få hade försökt lära ut spelet till någon annan, och därav är det svårt att hitta något mönster i vad som tycks vara svårast att lära ut.

Alla förutom en respondent associerade poker med engelska ord. Majoriteten var positivt inställda till att blanda språk i applikationen.

Applikationen skulle enligt respondenterna bland annat kunna ge användaren hjälp i form av att visa sannolikheten av att få olika kortkombinationer. Detta skulle dock kräva ganska avancerade matematiska algoritmer för att få till ett korrekt svar.

Vissa ansåg att möjligheten att kunna backa tillbaka ett steg mitt i en spelrunda hade varit bra i undervisningssyfte. Detta hade dock kunna medföra selfusk. Andra tyckte att det var viktigt att kunna stänga av hjälpen, för att kontrollera sitt kunnande. Ett intressant perspektiv, med tanke på att syftet är att ge användaren så mycket hjälp och förståelse som möjligt - ibland kanske man lär sig bäst först när hjälpen tas bort?

Spara-funktion verkar önskvärt, trots att spelet i första hand är ett spel skapat i undervisningssyfte, och har därav inte samma behov av att spara utveckling så som i de flesta andra spel.

Majoriteten vill kunna påverka antalet AI-spelare. Att justera antalet AI-spelare påverkar spelet i sig, då det är skillnad på att spela en mot en jämfört med att spela ett 5-mannabord. Att endast ge en variant kan således ge en missvisande bild av kortspelet.

De flesta ansåg att möjligheten till att ändra pottstorleken vid spelets början var en onödig funktion. De som däremot var positivt inställda till funktionen förstod att den skulle kunna komma att påverka spelets längd.

Effektljud är önskvärt och kan i vissa fall fungera som feedback. Att kunna slå på/av ljud anses också viktigt. Bakgrundsmusik tycks mest vara en störande moment.

Slutsatser

Följande slutsatser har dragits:

- ett brett spektrum gällande pokervana hos respondenterna är positivt
- de flesta lär sig poker under tiden som de spelar det
- många associerar poker med engelska termer
- sannolikhet på kortkombinationer är en förväntning hos vissa respondenter, men kan vara svårt att faktiskt implementera korrekt
- att kunna stänga av hjälp-funktioner kan faktiskt fungera som en typ av hjälp i sig
- spara-funktion är önskvärt även i denna typ av spel
- att kunna ändra antalet AI-spelare (motspelare) är önskvärt
- att kunna ändra pottstorleken anses vara en onödig funktion
- effektljud är bra! Bakgrundsmusik: nej tack.

Testrapport Användarbarhetstestning - Semistrukturerad Intervju, TR.SI.AB.2

Frågeformuläret som intervjun utgick från finns bifogat under bilagor (*Semistrukturerad intervju #2 (intervjufrågor)*).

Totalt intervjuades 3 personer. Samtliga intervjuer har spelats in. Dessa inspelningar har tillsammans med anteckningar legat som grund till nedanstående analys.

Information som framkom under intervjuerna

Respondenterna informerades om att en tutorial snart skulle implementeras i applikationen. Efter att blivit navigerade fram till själva spelplanen tillfrågades respondenterna vad de ansåg borde lyftas fram i en tutorial. Samtliga respondenter svarade att direkta pokertermer borde förklaras, såsom *“Raise”, “Check”, “Call”, “Fold”, “Dealer”, “Small Blind”* och *“Big Blind”*. En respondent påpekade att det var viktigt att belysa att reglaget (*som reglerar satsning*) tillhör raise-funktionen. Två respondenter ville att highlighting-funktionen av kort (*sker när man har en handkombination*) skulle förklaras. En respondent tyckte att namnet på rundorna borde förklaras. Två av respondenterna ville att regelboksknappen skulle framhåvas i tutorialn; den tredje respondenten tyckte att det var onödigt. En respondent ville att användarens personliga kort skulle förtydligas. Andra saker som uppkom under intervjun var att turordningen borde förklaras, att hjälprutan och handrankningsrutan borde framhåvas, samt att handrankningens hierarkiska form borde tydliggöras.

Vidare tillfrågades respondenterna om den valda färgpaletten på applikationen. De ombads att beskriva designen och vilken känsla den medförde. Samtliga respondenter uttryckte att designen upplevdes väldigt poker-aktig. Alla tyckte dessutom att det var bra att applikationen följde ett konventionellt poker-färgtema. En av respondenterna tyckte dock att det också hade varit kul färgskalan hade istället varit icke-konventionell, och att det kanske hade *“lockat fler”*. En av respondenterna sa att det fanns bra och tydliga färgkontraster i gränssnittet överlag. Färgpaletten ansågs inte vara för skrikig. En respondent uttryckte att stilen på hjälprutan och handrankningsrutan inte riktigt passade in med spelplanen i övrigt.

Slutligen tillfrågades respondenterna vad de ansåg om placeringarna av funktionerna i gränssnittet och om de hade velat ändra något. En respondent föreslog att det hade kunnat skapas mer utrymme för knapparna (*Check, Call, Fold, Raise*) genom att minska ner användarens userbar (*innehållandes användarens namn, pengar samt senaste drag*). En annan respondent beskrev att texten med rundornas namn upplevdes som en logotyp på grund av placeringen, och att denna text istället borde placeras någonstans längs nedre långsidan (*där all annan information är*). En respondent tyckte istället att det möjligtvis var för mycket information på samma ställe, och att denna borde spridas ut mer.

Analys av resultat

Majoriteten tyckte att just poker-terminologi borde förklaras under en tutorial. Det var också överlag viktigt att informera användaren om de hjälpfunktioner som användaren kan ta del av (*hjälpfunktionen, highlighting av kort, handrankningsrutan och regelboken*).

Den valda färgpaletten tycks inte en autentisk pokerkänsla. Detta var också positivt enligt alla respondenter.

Placeringarna av funktioner finns det delade meningar om; en respondent ville ha all information placerad på samma ställe, medan en annan istället tyckte att informationen borde spridas ut mer och att det kändes tomt på vissa delar av gränssnittet.

Slutsatser

Följande slutsatser har dragits:

- Det är viktigt att förklara pokertermer i en tutorial (*applikationen ska ju trots allt lära användaren spela poker*).
- Den valda färgpaletten uppskattas.
- Om vidare användbarhetstestning skulle genomföras så hade det varit en bra idé att presentera olika varianter av gränssnittet (*rent placeringsmässigt av funktioner*) på flera respondenter, för att se vilken som är mest omtyckt.

Bilagor

Think-Aloud-Protocol Session 1-5 (scenario)

Tanken med det här är att ni ska säga vad ni tänker och tycker högt under tiden som ni följer ett scenario som är förberett.

Exempel: Klicka med musen

Säg vad ni tror ska hända när ni klickar med musen.

Säg vad som hände när ni klickar med musen.

Säg vad ni ser och vad ni tycker om gränssnittet.

Under tiden så kommer det inte några förklaringar utan förhoppningen är att ni konstant uttalar vad ni tänker och förväntar er. Icke-verbala uppmaningar kan förekomma om ni blir tysta.

Scenario början:

Huvudmeny:

Tryck på "Ladda spel"

Tryck på "Nytt spel"

Tryck på "Starta spel"

Välj inställningar

Tryck på "Starta spel"

Välj "Call/Check, Fold eller Raise"x4

Titta runt, klicka på vad du än tror ska kunna ge någon effekt eller göra något.

Semistrukturerad intervju #1 (intervjufrågor)

Intro: Välkomna er respondent, och berätta följande:

- Detta är en semistrukturerad intervju, vilket innebär att det finns färdiga frågor som vi önskar att respondenten besvarar, men att det också finns utrymme för avvikelser från frågeformuläret. All information som kan vara av intresse är välkommet.
- Denna intervju är tänkt att spelas in (röstinspelning); bekräfta med respondenten att detta är okej.
- Förklara att målet med intervjun är att vi som projektgrupp ska få en klarare bild av vad potentiella slutanvändare har för förväntningar på denna typ av produkt.
- Berätta om vårt projektförslag! Glöm inte att klargöra följande:
 - ☐ Produkten är en spelapplikation (Texas Hold'em Poker).
 - ☐ Applikationen är främst tänkt till att lära användaren hur man spelar Texas Hold'em Poker, under spelpartiernas gång.
 - ☐ Användaren kommer att spela mot bottar.
 - ☐ Användaren kommer spela med låtsasvaluta.

Börja spela in!

1. Vet du hur man spelar Texas Hold'em Poker?
 - 1.1. Om **NEJ**, har du någon gång försökt lära dig hur man spelar? Varför gick detta inte bra?
 - 1.2. Om **JA**,
 - 1.2.1. Hur lärde du dig spela?
 - 1.2.1.1. Om **VIA HEMSIDOR**, vilka?
 - 1.2.2. Hur lång tid (uppskattningsvis) tog det dig att känna dig bekväm i spelets regler?
 - 1.2.3. Har du någon gång försökt lära ut spelet till någon annan? Hur har det gått i så fall? Fanns det något som var extra svårt att förklara?
2. Vilka ord förknippar du med Poker (rabbla upp dem som du kommer på)?
 - 2.1. Om **FLERTAL ORD ÄR PÅ ENGELSKA**, påpeka detta, och fråga respondenten om de tänker sig att applikationen borde vara på engelska.
 - 2.2. Om **FLERTAL ORD ÄR PÅ SVENSKA**, påpeka detta, och fråga respondenten om de tänker sig att applikationen borde vara på svenska.
3. Vilket språk bör vara dominant i applikationen (engelska/svenska)? Finns det tillfällen där du vill blanda språk; t.ex. ha pokertermer som *"call"* & *"raise"* på engelska, men hjälpinstruktioner på svenska?
4. Försök att föreställa dig applikationen; på vilket sätt skulle du vilja motta hjälp av applikationen?
5. Kommer du på någon viss funktionalitet som denna applikation borde ha enligt dig?
6. Ser du något värde i att ha en *spara*-funktion i denna typ av applikation?
7. Ser du något värde i att som användare kunna välja *antal bottar* att spela mot?

8. Ser du något värde i att som användare kunna bestämma *pottstorleken* vid spelets början?
9. Vad anser du om ljud i denna typ av applikation? Bakgrundsmusik? Effektljud?
Ena/båda/ingetdera?

Transkript: Semistrukturerad intervju (S.I.1 #1)

[Intervjuare]: Vet du hur man spelar Texas Hold'em Poker?

[Respondent]: Jag har faktiskt spelat en helt del när jag var yngre, online-poker och sånt... Men det var så länge sen...så jag kommer typ inte ihåg, heh!

[Intervjuare]: Så du känner dig osäker? Ska vi ta...

[Respondent]: Aa. Jag vet att man får två kort till en början. Och sen får man väl...och så läggs det ut tre kort. Och så får man välja om man vill betta eehm... stanna kvar eller lägga sig.

[Intervjuare]: Ja. Jag skriver ner ett ja så länge, du verkar ändå ha lite koll. Kommer du ihåg hur du lärde dig spela?

[Respondent]: Jag tror jag lärde mig tillsammans med mina vänner, mina barndomsvänner, för vi spelade varje vecka. Fysiskt då.

[Intervjuare]: Så i person helt enkelt? Inte via hemsidor?

[Respondent]: Nä.

[Intervjuare]: Hur lång tid tog det dig att känna sig bekväm i spelets regler..? Eller du kanske fortfarande inte är helt hundra på dem? Du kanske var hundra på dem då?

[Respondent]: Aa..eeh. Jag tror... Det är ju ett ganska simpelt spel, så sätt. Regelmässigt och vad det går ut på. Så jag tror de..nä, men det tog ganska.... Det gick ganska fort. Men sen, det andra...

[Intervjuare]: ...beslutsfattandet tar längre tid..?

[Respondent]: Aa, det gör det ju, absolut.

[Intervjuare]: Men, regelvis, en-två sessioner kanske..?

[Respondent]: Aa, det skulle jag tippa på.

[Intervjuare]: Har du någonsin försökt lära ut spelet till någon annan? Och i så fall, hur har det gått? Är det något som är extra svårt att försöka förklara för den här personen?

[Respondent]: Nej, jag tror aldrig jag har försökt lära ut det till någon annan....eehm. Möjligtvis någon gång för längesen i så fall, men jag kommer inte ihåg.

[Intervjuare]: Nä men det är lugnt, absolut. Ehm. Lite word-association. Om jag säger poker, vad får du upp i huvudet? Rabbla på så många ord som du kan komma på.

[Respondent]: Ehm. "PENGAR", hehe. Och – "solglasögon"! "Bluffa". Eehm...ja, det var väl det, jag vet inte, hah!

[Intervjuare]: Några termer..?

[Respondent]: "Fold". "Call". Ehm. Vet inte vad mer.

[Intervjuare]: Tog stopp där..? Då kör vi på det! Du använder en blandning av svenska och engelska ord, så jag skriver det. Om vi nu ska göra det här spelet, tycker du att det primärt ska vara på engelska eller på svenska?

[Respondent]: Bästa hade väl vart att kunna välja, mellan båda. Men ja, jag föredrar ju den alltså... engelska pokertermerna föredrar jag ju hellre än "syna" och "lägga sig" och hela det här.

[Intervjuare]: Så, du hade känt att det var helt okej att ha hjälpsektioner på svenska, men termer och sånt borde vara på engelska..?

[Respondent]: Ja, absolut, det tycker jag! Det känns som ändå poker är väldigt internationellt på något sätt, så mm. Man ska lära sig...

[Intervjuare]: Så det är bättre att säga "Call" än att säga "Syna" till exempel?

[Respondent]: Ja. Och lär man sig poker så kanske man har som mål att man vill spela på nätet, eller lite så... Då vill man inte sitta där och inte ha någon koll på vad dem olika orden betyder kanske, för att man lärt sig helt på svenska.

[Intervjuare]: Om du skulle föreställa dig en applikation som bokstavligen är till för att hjälpa

dig lära dig poker, finns det någon specifik typ av hjälp som du kanske hade velat ha?

[Respondent]: Jag vet inte, kanske typ såhär sannolikhet och sånt, alltså, någon form av...man kan få lite hjälp med det.

[Intervjuare]: Typ, handstyrka och liknande?

[Respondent]: Aa, precis.

[Intervjuare]: Eehm, jag vet inte om du har spelat online, men du vet, om man har handrankningen, den listan man brukar ha, med royal straight flush...och så high card längst ner. Hade du velat se en sådan? Eller ska det vara i en hjälpmeny..?

[Respondent]: De skulle väl vara schysst om man hade kunnat, ja men, få upp den hjälpen om man vill ha den. Och ta bort den om man inte vill ha den.

[Intervjuare]: Något annat som du kan tänka dig..?

[Respondent]: Ja, alltså, man kanske skulle kunna på tips om lika typer av sätt att spela på, för...det är ju väldigt mycket, man kan ju vara konservativ och bara slänga allt tills man får bra kort liksom. Eller du kan vara en person som är lurig och som man inte vet vart man har. Och vad dem olika spelstilarna kan innebära liksom, ja men, bluffar du så är det ju mer risk att folk synar dig liksom. Då kan du ju få jävligt mycket mer betalt för dina bra händer. Och andra sidan, om du slänger alla din kort hela tiden tills du får ess-ess eller ess-kung så är det ingen som kommer att syna dig när du...

[Intervjuare]: ...det finns ju en risk att du åker upp på blind:sen också om du sitter och håller dig för länge.

[Respondent]: Aa precis. Jo, men lite sånt kanske. Olika typer av spelstilar liksom.

[Intervjuare]: Jag ska ställa en lite ledande fråga här; om vi hade haft t.ex. en hover-funktion, så har du sett att man har en markör för "small blind" och "big blind" ibland; om man hade hov:rat över dem och fått upp en liten snabb inforuta, hade det varit intressant..?

[Respondent]: Ja, det tror jag ändå. Låter vettigt. Nu är jag ju inte helt grön på poker liksom, men jag kan tänka mig att om man är helt ny så vill man ändå veta så mycket som möjligt. Hellre för mycket information än för lite.

[Intervjuare]: Är det någon funktionalitet som du hade velat ha i den här applikationen?

[Respondent]: Svår fråga. Jag spelar ju aldrig mobilspel och sånt, så jag har ju lite dålig koll på vad jag själv skulle vilja ha.

[Intervjuare]: Ska jag ge ett par exempel? Fönsterstorlek..? Ljud, på/av?

[Respondent]: Jag är sån som typ avskyr reklam och typ, jobbiga grejer. Störande moment liksom. Sen förstår jag att man måste tjäna pengar på någonting. Jag är sån, jag köper hellre en app för en dollar eller två om den är bra, än att köra på att få se massa filmer och klipp var femte minut. Alla är olika.

[Intervjuare]: Antal AI-spelare som du spelar emot - hade det varit intressant att ändra? För att se dynamiken..?

[Respondent]: Absolut. Skillnaden mellan fem-personersbord och heads up är ju gigantiskt liksom.

[Intervjuare]: Samma sak med lika pott-storlekar..? Alltså, en startpott på ett bord. Också varit intressant..?

[Respondent]: Jaa. Och så kanske...möjlighet att spela någon form av turnering kanske.

[Intervjuare]: Ja, så att bord leder in till ett nytt bord och så vidare..?

[Respondent]: Ja men precis, att det är hundra AI-spelare från början..

[Intervjuare]: ...som sitter och spelar på egna bord tills en kommer ikapp?

[Respondent]: Ja, det är en annan form av tänk när man spelar turnering, i alla fall kommer jag själv ihåg det när jag var med i större turneringar på kasinot och sånt. Att, då tog man det ju väldigt lugnt i början och så lät man folk spela ut sig. Så det hade ju vart...

[Intervjuare]: Absolut en kul idé, det har vi inte tänkt på. Ser du något värde i att ha en

spar-funktion i en sådan här applikation?

[Respondent]: Ja men det tror jag, absolut. Alltid bra att kunna spara eller pausa, eller vad det nu är.

[Intervjuare]: När det gäller ljud, vilken typ av ljud hade du kunnat tänka dig ha? Snackar vi bakgrundsmusik eller effektljud eller det enda men inte det andra och så vidare..?

[Respondent]: Jaa...det är alltid schysst med lite så, ljudeffekter, i alla fall som någon form av respons som när man trycker på knappar, och kanske att det rasslar till när man bettar och sånt. Men inte för mycket, jag har aldrig varit ett fan av en 30-sekunders musikloop som går om och om igen, jävla Candy Crush eller vad det nu kan vara, men det finns väl folk som gillar det också. Jag personligen gillar det inte. Och det är skönt att kunna stänga av det om man vill.

[Intervjuare]: Så om bakgrundsmusik så ska det hellre vara ett längre musikstycke istället för något som kort-loopar?

[Respondent]: Ja, eller kanske nått mer ambient. Att det blir mer någon sorts matta som ligger i bakgrunden. Jag vet inte, när det blir för mycket melodier så sätter det sig på hjärnan.

[Intervjuare]: Bör man ha både en slide:r och av/på?

[Respondent]: Ja. Eller, slide:er och slide:er, du kan ju ändå trycka upp och ner på den här.

[Intervjuare]: Men om man bara hade kört av och på, hade du velat ha den tillgänglig alltid? Alltså, nere i hörnan, ha en liten sån..

[Respondent]: ..ja, det hade ju räckt.

[Intervjuare]: Sen så var det faktiskt inte så mycket mer för mig att fråga enligt den här, är det något annat du själv tänker på som du hade velat se i en sådan här eller liknande?

[Respondent]: Det hade ju vart kul att kunna spela mot andra människor, men ja, det är ju lite mer avancerat kanske. Någon grej som kanske hade varit schysst är att, om vi säger att man bara är intresserad av sina egna händer liksom, att kunna snabbspola runt, alltså...

[Intervjuare]: ...att AI-spelaren inte har någon delay?

[Respondent]: Ja men precis. Om du har lagt din runda liksom, nä jag stannar eller jag bettar, så kanske det vanligtvis tar lite för att göra det lite mer realistiskt liksom, så att det inte direkt bara är tillbaka till dig igen. Men du kan ha den möjligheten att bara trycka på en knapp och vara tillbaka till dig igen och så kan du bara se vad som har hänt.

[Intervjuare]: När man väl har börjat komma in i det så kanske det är drygt att sitta och vänta hela tiden..?

[Respondent]: Ja jag kan tänka mig det, att man kan tappa tålamodet lite grann.

[Intervjuare]: Ja absolut.

[Respondent]: Annars så...

[Intervjuare]: Ja, jag tror faktiskt inte jag har så mycket mer, så tack så mycket.

Semistrukturerad intervju #2 (intervjufrågor)

1. En tutorial ska strax implementeras i denna applikation. Det är meningen att en helt ny användare genom denna tutorial ska få vägledning och information om funktioner som finns tillgängliga, innan spelet kör igång. Vad anser du att tutorial:n bör fokusera på att lyfta fram på spelplanen?
2. Vad anser du om den valda färgpaletten som används i gränssnittet? Inger designen någon viss känsla? Hur skulle du beskriva designen?
3. Vad anser du om placeringarna av funktioner i gränssnittet? Hade du ändrat något?