

# QuizTime

**Grupp 35**

**Projektplan**

**v. 1.3**

**27/03-20**

### Dokumenthistorik

-Hur dokumentet har förändrats?

-Författare → Den som skrev och justerade texten i dokumentet.

Datum	Version	Beskrivning av ändring	Författare
12/3-2020	1.0	Dokument skapas genom att skriva ner alla rubriker. Skriver förslag på syfte, utvecklingsprocess, planering & riskfaktor.	Marianne Mukanga
21/3-2020	1.1	Lade till en mening under <i>Syfte</i> och 4 ord under <i>ordlista</i> .	Marianne Mukanga
22/3-2020	1.2	Lade till text under <i>utvecklingsprocessen</i>	Marianne Mukanga
26/3-2020	1.3	Uppdatering av riskanalys	Martin Bergman, Marianne Mukanga, Erik Nielsen, Markus Gerdtsen
27/3-2020	1.4	Uppdatering av planering för sprint 1	Erik Nielsen Marianne Mukanga

[Fler rader läggs till efterhand som det behövs. Utifrån beskrivningen ska det gå att förstå vad personen gjorde för typ av ändring. Så bara en text som säger ”ändring” räcker inte. Texten behöver exempelvis säga ”Lagt till beskrivning av risk Sjukdom” eller ”Förtydligat beskrivning av produktbeskrivningen.”

Det kan finnas flera författare. Endast de som faktiskt är aktiva med att skriva texten listas som författare. Om man är två som sitter och jobbar tillsammans och en skriver men man hela tiden aktivt diskuterar det som skrivs så kan bägge personerna stå som författare. Om man har ett möte i hela gruppen där man diskuterar saker, en person tar anteckningar och skriver sedan rent detta i dokumentet så står endast denna person som författare.]

## Projektplan

### Syfte

Gruppen har utformat projektplanen för att ha något att utgå ifrån under arbetets gång. Dokumentet kommer att fungera som en vägledning för gruppen när det kommer till vilka mål medlemmarna har med projektet.

### Ordlista

*Svåra ord förklaras här.*

Användare - Person som använder applikationen

Scrum - Utvecklingsprocess

Kanban - Utvecklingsprocess

API - ett slags protokoll som används för att program (även kallat applikationer) enklare ska kunna snacka med varandra

[Ta upp förkortningar och uttryck som används i dokumentet och som kanske inte är självklara för en läsare som inte är insatt i projektet. Tänk också på att ta upp begrepp som skulle vara svåra att förstå för en läsare som skulle kunna vara en presumtiv kund eller referensperson i form av exempelvis slutanvändare. Ordlistan ska ordnas i bokstavsordning.]

### Referenser

*-IEEE*

<https://gs.statcounter.com/macOS-version-market-share/desktop/worldwide>

### Översikt av projekt

<Underrubrik>

### Mål

Vi vill uppnå ett färdigt quiz-program på datorn. Vi vill även att spelet ska vara estetiskt tilltalande och kul. Alla i gruppen vill bli bättre på att programmera. Vi vill också bli bättre på att arbeta i grupp och lära oss hur man arbetar agilt.

## Omfattning

Begränsade i tid, gruppens storlek. Kunskapsnivån - programmeringsspråk. Inga fysiska möten. För samtliga gruppmedlemmar är detta första projektet vi gör i denna utsträckningen. Gruppen har inga tidigare erfarenheter med programmerings kunskaperna som ska användas.

## Jag vill

Markus: Jag vill bli bättre på att dokumentera.

Erik: Bli bättre på att arbeta i grupp som ett team.

Martin: Jag vill bli bättre på att programmera. Målet är att ha ett färdigt quiz-program på datorn. Jag vill lära mig använda de agila utvecklingsprocesserna.

Marianne: Jag vill bli bättre på att programmera.

## Ambitionsnivå

Erik: Jag ser kursen som en kurs att klara av med precis lagom ansträngning och vill helst göra en enklare prototyp.

Martin: Kommer lägga ner ca 18/h i veckan. Ambitionen är såklart en färdig produkt.

Marianne: Jag vill lära mig om vad det innebär att vara projektledare. Jag är öppen för att i så stor utsträckning som möjligt satsa på att ha gjort en färdig produkt, men det är inget krav för mig.

Markus:

## Intressenter

### <Underrubrik>

*-Vilka är våra intressenter?*

*-OBS! Vi i projektet räknas också som intressenter. Gäller även lärare.*

*-Vilka användare är intressenter?*

Våra intressenter är oss själva, kurskamrater samt lärare. Det finns inga planer på att lansera en färdig produkt till företag. Om så skulle vara fallet så skulle våra potentiella användare vara personer som kan läsa på engelska och som har ett intresse för allmänbildning.

## Produktbeskrivning

### <Underrubrik>

*-Typ som projektförslaget men mer finslipat.*

Produkten ska vara en quiz applikation med flera kategorier. En användare ska skapa en profil och logga in med användarnamn och lösenord. Vid registrering väljs en bild.

Användaren kan starta spel genom att utmana en annan person i realtid. Kategori väljs, sedan börjar frågesport. Ett antal frågor med fyra svarsalternativ dyker upp med tidsbegränsning på 15 sekunder. Efter spelet är färdigt är vinnaren den med flest rätt och spelarnas resultat sparas i en databas, resultat visas i form av lista tillsammans med användarnamn och poäng, typ en

highscore lista. Användaren kan lägga till andra spelare i en sorts friendlist. Highscoren visar endast användarens highscore och andra spelare som har lagts till i friendlist. Valmöjligheten att möta en bot finns.

**Utvecklingsprocess** (Glöm ej att den här delen ska refereras.)

*-Hur tänker vi jobba - utvecklingsprocess? (Tänker Scrum & Kanban?)*

*-Hur ska vi testa produkten?*

*-Utvecklar vi produkten versionsvis eller parallellt (?)*

*-Vem ansvarar för vad och hur bestämmer vi det?*

*-Ska vi parprogrammera eller köra var för sig, eller kombinera detta?*

*-Ska vi ge varandra olika roller? Vem har vilken roll i så fall?*

[Referens:

[Kanban / scrumban \(10.00-12.00\)](https://www.agilealliance.org/glossary/kanban/#q=~(infinite~false~filters~(postType~( '~page '~post '~aa_book '~aa_event_session '~aa_experience_report '~aa_glossary '~aa_research_paper '~aa_video) ~tags~( '~kanban) ~searchTerm~ '~sort~false~sortDirection~'asc~page~1) ]</a></p></div><div data-bbox=)

### **Kanban**

Kanban är en metod för att organisera ett projekt. På så vis effektiviseras arbetsflödet så att arbetet går snabbare. Eftersom alla gruppmedlemmarna saknar erfarenhet i området så passar den här utvecklingsprocessen oss väl. Detta då Kanban anses vara lätt att införa i arbetet och för att det är väl anpassat efter små förändringar. [KÄLLA finns]

### **Scrum**

Kort om Scrum - mer ska skrivas här!

Kanban är enkelt att anpassa efter andra metoder/processmodeller, exempelvis Scrum. Detta då man i Kanban också kan ha en **scrummaster** och en **produktägare**. Skillnaden mot att enbart jobba med metoden Scrum blir att arbetsgruppen/utvecklarna har ett begränsat antal aktiviteter att jobba med samtidigt. Först när en aktivitet är färdig att gå vidare till utvecklingsfasen så kan man lägga till en ny aktivitet. Fördelen med detta arbetssätt blir för oss att hela processen tydliggörs och att varje steg blir tydligt för arbetslaget. Det är extra viktigt för oss nu när vi inte kan träffa varandra. Vi kommer ta hjälp av YouTrack där vi skapar en så kallad digital tavla med våra backlogs för sprintplanering. [KÄLLA finns]

Vanligtvis ska man hålla dagliga möten, så kallade synkroniseringsmöten och har samma principer som ett scrummöte. Skillnaden i vårt fall blir att vi tittar på sprintplaneringen på vår digitala tavla på YouTrack. I samband med synkroniseringsmötena hålls även reflektionsmöten för att se över om något krav behöver justeras eller läggas till. När man arbetar efter Kanban så tar man inte göra någon uppskattning på hur lång tid det kommer ta

att utföra en uppgift. Vi kommer att anpassa vår processmodell genom att plocka in delen från Scrum där vi vid varje sprint fördelar de utvalda delarna från vår backlog och bryter ner dessa i mindre bitar så att de går att utföra på antingen en dag eller en vecka. [KÄLLA finns]

### **Rollerna i gruppen:**

Scrummaster – *ungefär som en projektledare, en som leder gruppen* I vår arbetsgrupp har vi bestämt att vi är två personer som leder arbetsgruppen, där en person har det yttersta ansvaret medan den andra tar ett mindre ansvar. Att vara projektledare (tror jag) innefattar att man lägger upp en god sprintplanering. Detta innebär att projektledaren bör ha god koll på gruppens product backlog. Det förutsätter att alla kraven är färdigskrivna och tydligt prioriterade för att projektledaren även ska kunna välja/göra(?) en sprint backlog och utifrån det planera den framtida sprinten.

Development team – *utvecklarna*

Utvecklarnas roll blir i detta fallet att utveckla produkten, vilket görs inkrementellt. [KÄLLA finns]

### **Planering**

*-Hur ska projektet fördelas över tid?*

*(-Hur ser gruppmedlemmarnas privata tid ut?)*

*-Hur ser veckorna ut?*

*-Glöm ej att lägga till marginal vid planering - ca 10%*

<Text>

[Den första delen är en övergripande text som beskriver förutsättningarna för planeringen så som antal timmar i budgeten, strategier som används har för att inte över- eller under-planera budgeten (exempelvis att planera med 10% oplanerad tid för att ha marginal för oplanerade saker). Här ska listas saker ni kan se att ni måste ta hänsyn till i er planering av aktiviteter i YouTrack senare. Detta kan vara saker som exempelvis

- att där är helgdagar vissa veckor
- tentor eller uppgifter på andra kurser som tar extra tid någon vecka
- någon ska vara bortrest

eller andra saker som gör att ni kanske planerar att jobba mindre dessa veckor och vilka veckor ska ni då planera att ta igen den här tiden på.]

### **Veckoplanering:**

Vecka 11 **Mål:** Delta i workshopen, föreläsningarna och labbarna. Diskutera projektplan och krav dokumentet. Definiera socialt kontrakt.

Kickoff 9/3, Workshop 10/3, Projektplan och kravdokument & YouTrack labb 11/3, Grupphandledning, Gruppdynamik & möte om socialt kontrakt 13/3

Vecka 12 **Mål:** Påbörja en grund med kodningen och presentera förslag på GUI. Bestämma en utvecklingsprocess. Fortsätta med kravdokument och fastställa kraven. Dela in individuella fördjupningar till projektets medlemmar där målet är att påbörja de individuella fördjupningarna. Komma igång med arbete med JavaFX. Projektets medlemmar kommer ta hänsyn till tentamen nästkommande vecka och kommer därför inte lägga ner lika mycket arbete på projektet som önskat.

Vecka 13 **Mål:** Påbörja diagram för projektets kod upplägg och att programmera en server-klient uppkoppling. Påbörja sökande av ett passande API och börja designa en prototyp. Se över projektplanen och krav dokumentet inför RS1. Påbörja arbetet med design dokumentet.

Sprint 2 v14-16. Till deadline för retrospektmöte 2 ska följande som minst ha uppnåtts:

- Det ska finnas en version av produkten med en delmängd av de specificerade kraven ur kravdokumentet implementerade.
- Det ska finnas ett designdokument i en första version med väsentlig funktionalitet och design av gränssnitt beskrivet.
- Det ska ha genomförts kravbaserade tester enligt testdokument för detta av gruppen själva samt av externa testare i samband med testdag för detta och dessa ska ha dokumenterats i minst två testrapporter.
- Minst en kodgranskning och en dokumentgranskning ska ha genomförts.
- För eventuella användbarhetsanalyser och/eller användbarhet/spel-tester ska metoder ha valts ut och finnas beskrivna samt vilka anpassningar som behövs för det aktuella projektet.

Sprint 3 v17-v19. Till deadline för retrospektmöte 3 ska följande som minst ha uppnåtts:

- Det finns en första version av produkten som är en helhet.
- Alla krav prioriterade som ”ska”/”must” är implementerade och testade enligt testdokument för kravbaserad testning och dessa tester ska ha dokumenterats.
- Ytterligare tester ska ha körts av annan grupp i samband med testdag och denna grupp ska ha lämnat testrapport på testerna som kördes.
- Eventuella användbarhetsanalyser och/eller användbarhet/spel-tester ska ha genomförts minst en gång.
- Minst två kodgranskningar och två dokumentgranskningar ska ha genomförts.

Sprint 4 v20-22. Till deadline för retrospektmöte 4 ska slutinlämning av projektet ske. Under sprint 4 ska projektet förfinas utifrån den version som fanns efter sprint 3. Som minst så ska följande ha uppnåtts:

- Problem som noterats tidigare ska ha åtgärdats och vissa av de krav som prioriterats som ”bör”/”should” ska ha implementerats.
- Regressionstester ska ha genomförts för förändringar som skett och ny implementation ska ha testats minst en gång.

- För eventuella användbarhetsanalyser ska en kompletterande rapport ha gjorts som adresserar hur problem som pekats ut i den första analysen åtgärdats.
- För eventuella användbarhet/spel-tester ska en andra omgång av tester genomförts för att undersöka om åtgärder som införts på rekommendation av de första testerna har haft önskad effekt.

## Milstolpar

*-Sprintar & och milstolpar inom sprintarna*

*-Den individuella fördjupningen (milstolpar där också?)*

[Milstolpar innebär att något mål ska ha uppnåtts och/eller att något ska levereras (exempelvis att en första release av kravspecifikationen ska vara klar eller att en viss release av produkten ska vara klar). Varje leverabel som ingår i en milstolpe ska listas. En milstolpe kan innehålla flera saker som ska vara avklarade för att milstolpen ska uppnås. Fler milstolpar än de definierade nedan behövs. Observera också att vissa milstolpar kan härledas ur kraven på sprintarna i projektguiden. Det sak finnas fler milstolpar än slutet på de fyra sprintarna.]

Milstolpe 1 <Namn på milstolpe>, vecka <nr>, <dag>/<månad>

<beskrivning av milstolpe, några meningar är tillräckligt>

Leverabler:

<beskrivning av leverabel>

<beskrivning av leverabel>

Milstolpe 2 <Namn på milstolpe>, vecka <nr>, <dag>/<månad>

<beskrivning av milstolpe, några meningar är tillräckligt>

Leverabler:

<beskrivning av leverabel>

<beskrivning av leverabel>

Risikanalys

<Text som beskriver syftet med risikanalysen, hur den genomförts samt beskrivning av olika kategorier som används. Sannolikhet och konsekvens kan anges som en siffra 1-x eller ord som låg, mellan, hög. Vad som avses med de olika kategorierna ska beskrivas så att man kan förstå nivån på dessa i relation till projektet. Det behövs fler risker än vad som visas nedan.>

## Risikanalys

R1- JavaFX

Beskrivning: Att programvaran som projektet använder ej fungerar efter förväntan.

Sannolikhet: Hög - Alla i gruppen har inte lyckats installera JavaFX och det upplevs vara svårt att använda.

Konsekvens: Låg - Det kommer inte påverka själva programmet i sig. Det kommer enbart påverka programmets design och användarens upplevelse.



- F1: Vara flexibla med val av programvaran
- H1 Handlingsplan: Byter programvara

#### R2 - Databas

Beskrivning: Att inte lyckas med att programmera en databas

Sannolikhet: Mellan

Konsekvens: Stor - Det kommer påverka gruppens möjligheter att skapa ett spel som sparar data. En stor och viktig del av spelet påverkas.

- F2 - Förebyggande: Använda andra alternativ istället för en databas om det finns
- H2 - Handlingsplan: Hitta en annan lösning än databas alternativt få mer hjälp av handledare eller dylikt.

#### R3 - Sjukdom

Beskrivning: Att en person i gruppen blir allvarligt sjuk

Sannolikhet: Låg

Konsekvens: Mellan - Antalet timmar som läggs ner på projektet minskar och risk för att hamna efter i tidsplanen och risk för minskat deltagande.

- F3. Ta hand om sig själv och följa myndigheternas riktlinjer om att hålla sig hemma.
- H3 Handlingsplan: Dela upp personens uppgifter mellan de friska medlemmarna, kommunicerar med personen som är sjuk och bestämmer när de kan återgå.

#### R4. Tekniska svårigheter

Beskrivning: Att datorerna gruppen använder sig av hemifrån havererar

Sannolikhet: Låg

Konsekvens: Tidsplanen och möten blir påverkade, hamnar efter i planeringen.

- F4. Bra underhåll av datorn, använda sig av antivirus program. Gå inte in på osäkra eller oseriösa webbsidor.
- H4. Handlingsplan: Spara kod på GitHub så att den finns som backup!

#### R5. Ny kunskap

Beskrivning: Att den nya kunskapen gruppen ska lära sig är för tidskrävande

Sannolikhet: Mellan

Konsekvens: Mellan

- F5. Skala ner på omfattningen av projektet. Dela med sig av den kunskapen man redan har.
- H5. Handlingsplan: Ta den hjälp som personen känner att den behöver från handledare.

#### R6. Handledare

Beskrivning: Att handledaren inte har tillräcklig eller förväntad kunskap.

Sannolikhet: Svår att definiera

Konsekvens: Medel

- F6. Informera handledaren om brist på undervisnings kvalite.
- H6. Handlingsplan: Hitta andra sätt att tillhandahålla kunskapen.

R7. Kunskapsnivå

Beskrivning: Att vissa arbetsuppgifter i projektet tar längre tid att utföra på grund av att alla i gruppen har olika kunskaper.

Sannolikhet: Hög

Konsekvens: Låg

- F7. Ha tålmod och hjälps åt. Vi göra detta för att lära oss. Var pedagogisk
- H7. Handlingsplan: Skulle det vara för stora kunskapsskillnader så kanske man får rotera ansvarsområdena och hjälpa till där man känner man kan hjälpa.

### **Riskdiagram**

[Infoga diagram som placerar riskerna i ett diagram som har sannolikhet på ena axeln och konsekvens på den andra. Diagrammet behöver inte vara komplicerat utan kan exempelvis skapas med en tabell.]