

# VEDLEGG 1 KRAVSPESIFIKASJON

## INNHALDSFORTEGNELSE

- Forord..... 2
- 1 Systembeskrivelse ..... 2
- 2 Mål for systemet ..... 3
- 3 Funksjonelle krav..... 4
- 4 Ikke-funksjonelle krav..... 5
- 5 Use-case diagram ..... 6
- 6 Rammekrav..... 6

## FORORD

Hensikten med denne kravspesifikasjonen er å beskrive bakgrunnen og formålet med applikasjonen, hvilke behov som finnes og hvilke tekniske rammer prosjektet skal gjennomføres innenfor.

Kravspesifikasjonen blir regnet som et internt styringsdokument mellom arbeidsgiver og gruppen, og fungerer som en slags avtale. Kravspesifikasjonen er beregnet på alle interessenter i prosjektet, og er skrevet deretter med tanke på faguttrykk og liknende. Prosjektet med dets funksjonelle og ikke-funksjonelle krav vil bli nøye undersøkt og definert og datamodeller vil bli utarbeidet.

## 1 SYSTEMBESKRIVELSE

Prosjektet går ut på å lage en webapplikasjon hvor lærere kan utforme quizer etter undervisning og fag. Det kreves at hver quiz skal kunne inneholde minst 30 oppgaver og at hver oppgave skal kunne ha opptil tre svaralternativer. Det må i tillegg være mulig å legge inn verkslogoer i applikasjonen. Applikasjonen kan skal bestå av følgende oppgavetyper: videosnutt med spørsmål, bilde med spørsmål, lyd med spørsmål og kun spørsmål (kan være en lengre tekst etterfulgt av et spørsmål).. De riktige svarene må kunne markeres som riktige av forfatteren. Oppgavene skal nummereres fortløpende fra 1 og oppover.

Et annet krav til applikasjonen er at det skal være mulig å bruke quizene på forskjellige måter, dvs. med ulikt antall deltagere. Det skal være mulig å benytte seg av quizen som én deltager, flere deltagere og i grupper. I tillegg skal det være mulig å skrive ut quizen på papir. Elevene svarer da individuelt, eller gruppevis. Hver elev, eller gruppe, får et skjema med oppgaver for avkryssing.

Det skal være en administratordel hvor lærerne kan utforme quizen. Elevene kommer inn på den aktuelle quizen ved å skrive inn en kode og så registrere navnet sitt. Hvis det er en quiz for flere deltagere må elevene vente på at læreren starter quizen, ellers kan eleven starte når den selv vil.

Webapplikasjonen skal la lærere opprette quizer/prøver, som elevene kan ta ved å gjøre følgende:

- Gå inn på applikasjonen og taste inn en quiz id (kode)
- Gå inn på quizen sin unike URL

Elevene kan melde seg på en quiz enten individuelt, eller som en gruppe. Ved tidsbegrensing, kan lærer selv starte quizen når alle har meldt seg på. Lærer vil kunne se måloppnåelse for elevene/gruppene inne på en egen side.

I tillegg til quizene læreren selv lager, vil det være quizer til noen av oppdragsgiverens bokverk som er laget på forhånd.

Hovedmålet med arbeidet vårt er å tilby Cappelen Damm en god applikasjon som skal fungere som et digitalt lærerverktøy for lærere i grunnskolen. Applikasjonen skal gjøre det mulig for lærere å opprette quizer relatert til fag, inneholde et bibliotek med quizer og lærere skal kunne vurdere elevenes måloppnåelse ved å holde quizene.

I tillegg har vi disse målene:

- Løsningen skal være en webapplikasjon, altså en nettside som kan leses i en nettleser.
- Lage en komplett løsning som fungerer, bestående av back-end og front-end.
- Kildekoden skal kunne vedlikeholdes og videreutvikles av andre utviklere uten store utfordringer.
- Løsningen skal være frittstående slik at den lett kan implementeres på nettstedene til CDU.
- Løsningen skal ha en administratordel, hvor lærerne kan opprette, endre, eller slette quizer.
- Løsningen skal koble brukere til quizen ved hjelp av innskriving av en kode/ID.
- Løsningen skal inneholde et bibliotek av quizer, knyttet opp mot CDU sine bokverk.

## FUNKSJONALITETSKRAV

---

- Det skal kunne opprettes quizer med én deltager, flere deltagere og/eller grupper.
- Svarskjema skal kunne skrives ut på papir.
- Applikasjonen skal gjøre det mulig å lage følgende oppgavetyper: videosnutt + spørsmål, bilde + spørsmål, lyd + spørsmål og bare spørsmål.
- Hver quiz skal ha en egen ID.
- Hvert spørsmål skal kunne ubegrenset antall svaralternativer.
- Hver quiz skal kunne ha verkslogo for hvert av Cappelen sine bokverker.
- Elev skal kunne melde seg på en quiz ved å taste inn quizens kode og skrive inn navnet sitt.

## TEKNOLOGIER

---

- AngularJS
- NodeJS
- ExpressJS
- HTML
- CSS (Bootstrap)
- MS SQL / SQL Server

## RAMMEBETINGELSER

---

- Bruke JavaScript i både front-end og back-end.
- Bruke HTML, CSS (Bootstrap) og AngularJS til å utforme front-end.
- Bruke NodeJS og Express JS til å utforme back-end.

- Bruke Adobe Brackets og Visual Studio Code IDE for å utvikle systemet
- GitHub for versjonshåndtering og backup.
- Bruke smidig metodikk for prosjekthåndtering ved hjelp av Scrum.

### 3 FUNKSJONELLE KRAV

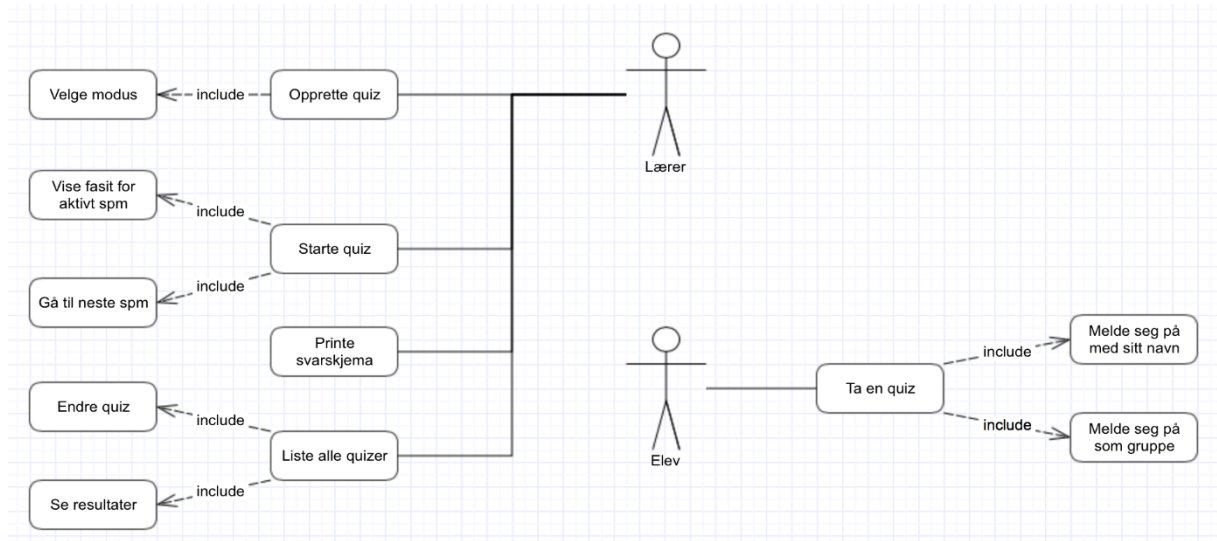
Krav	Forklaring	Viktighetsgrad
<b>Opprette quiz</b>	Skal ha muligheten til å opprette opptil flere quizer	Absolutt krav
<b>Lenke til quiz</b>	Hver quiz skal ha sin egen URL som man kan lenke til	Absolutt krav
<b>Opptil 30 spørsmål</b>	Hver quiz skal kunne ha opptil 30 spørsmål	Absolutt krav
<b>Opptil 3 svaralternativer</b>	Hvert spørsmål skal kunne ha opptil 3 svaralternativer	Absolutt krav
<b>Adgangskode</b>	Elev som går inn på en nettside vil ha mulighet til å skrive inn en kode for quizen. Alternativ til å gå direkte inn på en aktuelle URL'en	Høy
<b>Oppgavetyper</b>	- Videosnutt + spørsmål - Bilde + spørsmål - Lyd + spørsmål - Spørsmål	Høy
<b>Logo</b>	Applikasjonen skal ha logo for Cappelen Damm	Høy
<b>Verkslogo</b>	Hver quiz skal kunne ha verkslogo for lærebok	Høy
<b>Avspilling av spørsmål</b>	Spørsmålsteksten skal kunne spilles av som lyd f.eks. via google translate	Middels
<b>Endre rekkefølge på spørsmålene</b>	Lærer burde kunne endre rekkefølgen på spørsmålene	Middels
<b>Animasjoner</b>	Flyt og brukeropplevelse kan forbedres med animasjoner	Middels
<b>Hjelp</b>	En hjelpedialog skal vises ved mouse-hover på knapper o.l.	Middels
<b>Karakterskala</b>	Lærer kan velge på sin egen side hvilken prosentandel som gir hvilken karakter. Karakteren vil da dukke opp ved siden av hver elev på resultatsiden.	Lav

<b>Nummerering</b>	Oppgave skal nummereres	Høy
<b>Moduser</b>	Lærer kan velge en modus på quizen (en eller flere personer)	Høy
<b>Enkeltperson-modus</b>	Enkeltperson uten tidsbegrensing	Høy
<b>Felles-modus</b>	Flere enkeltpersoner kan ta quizen med tidsbegrensing. Quiz startes av lærer når alle har registrert seg.	Høy
<b>Gruppe-modus</b>	En gruppe kan ta quizen med tidsbegrensing. Quiz startes av lærer når alle har registrert seg.	Høy
<b>Papir-modus</b>	Lærer starter quiz uten tidsbegrensing. Svarskjema kan enkelt printes ut, og elevene gjør	Høy
<b>Vise riktig svar</b>	På hvert spørsmål kan lærer trykke på en knapp som viser riktig svar	Høy
<b>Registrere seg</b>	Lærer skal kunne registrere en brukerkonto	Absolutt krav
<b>Påmelding av quiz</b>	Elev skal kunne melde seg på en quiz via. URL og taste inn sitt navn	Absolutt krav

#### 4 IKKE-FUNKSJONELLE KRAV

Krav	Forklaring	Viktighetsgrad
<b>SPA</b>	Applikasjonen skal være en SPA (Single-page application)	Absolutt krav
<b>Responsiv</b>	Applikasjonen skal være responsiv ovenfor forskjellige skjermstørrelser	Absolutt krav
<b>Ytelse</b>	Applikasjonen skal ha god flyt og ytelse	Middels

## 5 USE-CASE DIAGRAM



### 5.1 Use-case diagram over applikasjonen

Use-case diagrammet viser applikasjonens funksjonalitet og samspillet mellom systemet og omgivelsene (brukere, andre systemer og komponenter). Use-case diagrammet er en modellering av de funksjonelle kravene representert over. Diagrammet viser hvordan en lærer og en elev kan bruke applikasjonen.

## 6 RAMMEKRAV

### SIKKERHETSKRAV

- Sikring av inndata mot SQL injection
- All sensitiv data skal sikres
- Applikasjonen skal ikke lagre unødvendig sensitiv informasjon
- Scripting

### TEKNISKE KRAV

- Applikasjonen skal utvikles i Angular JS, Node og Express JS, HTML, CSS (Bootstrap)
- Applikasjonen skal støtte Google Chrome, Firefox, Safari, Edge og Internet Explorer 9 eller høyere.
- MVC (model-view-controller) skal benyttes for å skille logikk fra grensesnitt.
- Applikasjonen skal være skalerbar, dvs. det skal være enkelt å legge til ny funksjonalitet i fremtiden.
- Koden skal være godt strukturert, og navn på metoder, variabler mm. skal være selvforklarende.
- Koden skal supplementers med kommentarer som tydelig beskriver hva som foregår.

## DESIGN KRAV

---

- Knapper skal ha tilstrekkelig størrelse for å veie opp for unøyaktigheter
- Fargekontraster skal være tydelige.
- Plassering av knapper skal ligge naturlig i forhold til hverandre.

## KRAV TIL DATALAGRING

---

- Databasen skal være bygd opp med krav om normalisering

## KRAV TIL UNIVERSELL UTFORMING

---

- Applikasjonen skal være lettleselig og knappene skal være godt synlig.
- Applikasjonen skal ikke inneholde unødvendig gjentakelser.
- Farger som er brukt i applikasjonen skal være leselig for fargeblinde, svaksynte og folk med andre funksjonsnedsettelse