**Trond Vebjørn Omberg (193211)**

**Høgskolen i Østfold**

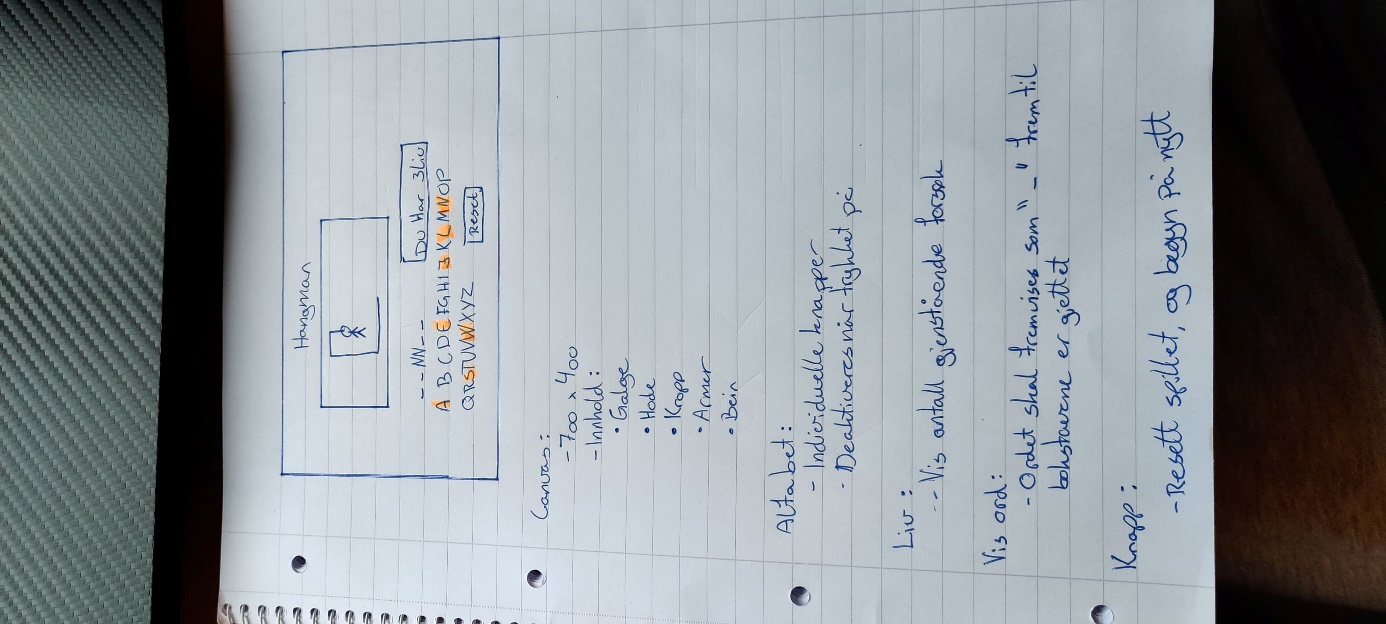
**Innføring i programmering Høst 2020**

**Arbeidsprosess – spillutvikling: Hangman**

**Forarbeid:**

Før jeg startet å kode, satte jeg meg ned og begynte å tegne nettsiden med penn og papir.

Et bilde som inneholder tekst

Automatisk generert beskrivelse

Når jeg hadde tegnet ned alle elementene jeg skulle ha med i spillet, kunne jeg begynne å skrive ned de variablene jeg trenger for å kode spillet, og kartlegge hvordan de er knyttet sammen.

**Kodingen:**

Jeg startet merd HTML elementer og CSS for å forme plattformen, og se bedre hvordan siden utvikler seg.

JavaScript-en kommer til å lage egne ul og li elementer. Disse plasseres i div-er

**JavaScript:**

Start med window.onload for å kjøre koden så snart siden lastes, og kjør resten av koden her.

Jeg må opprette et alfabet som skal bli til individuelle knapper. Jeg velger å holde meg innenfor det engelske alfabetet da det verken er æ, ø eller å i løsningsordene.

Jeg oppretter en array som heter myAlphabet, og inneholder alle bokstavene fra a – å.

Videre må jeg opprette variabler som jeg skal ta med meg videre:

Variabel som skal ta imot en array med løsningsordene:

* wordsList;

Bokstaver som skal gjettes:

* GuessedLetter

Array som lagrer de gjettede bokstavene:

* GuessedLetterArray

Hvis ordet skulle hatt mellomrom, blir dette lagret her:

* EmptySpace

Så må jeg hente elementer som skal vise liv:

* showCurrentLives = document.getElementById(‘’myLives’’)

Jeg må opprette liste-elementene som skal bli klikkbart alfabet.

Buttons = function() {

- Hente området jeg vil ha knappene i

- Lage ul-liste som skal ta imot li-elementer

- For-loop for å gå igjennom hele myAlphabet {

* + Legge ved id til bokstavene
  + Opprette li – elementer
  + Gi listen id letter
  + Innholdet i listen er innholdet i myAlphabet
  + Hver knapp skal kalle på funksjonen checkLetter
  + Plassere listen i div-en button
  + Plassere bokstavene i listen

}

};

Fremvisningen av løsningsord / bokstavene:

displaySlectedWord = function {

* Hente elementet som holder løsningsordet
* Opprette ul for korrekte bokstaver
* For-loop for å telle lengden av det valgte ordet {
  + Sette attributter på ordet
  + Lage li- elementer til hver bokstav
  + Sette attributter på bokstavene

Hvis ordet inneholder mellomrom {

* + - Sette fremvisningen til ‘’-‘’
    - Gi en verdi på 1

} ellers {

* + - Sett fremvisningen til ‘’\_’’

}

* + Pushe gjettet bokstav til arrayen
  + Plassere ul der løsningsordet skal vises
  + Plassere li i ul

}

};

Jeg vil vise livene, eller en nedtelling på hvor mange forsøk man har før spillet er tapt. Jeg vil også gi beskjed dersom spillet er vunnet eller tapt.

updateLives = function() {

* Vis livene med konkatinering da antallet angis senere
* Hvis liv er mindre enn 1 {
  + Taper man spillet (Game Over)

}

* For-loop som sjekker om alle bokstavene er gjettet {
  + Hvis teller + mellomrom er den samme fremvisning som løsningsordet {
    - Vinner man spillet (Winner!)

}

}

};

På tide å hente canvas og tegne mannen og galgen. Galgen og mannen blir kodet slik at hver kroppsdel og galgen i seg selv er et eget element i en array, og blir plassert etterhvert som man mister liv. Galgen skal være påtegnet før man starter

Jeg lager en variabel som oppdaterer canvaset

updateCanvas = function() {

* Opprette en variabel som har lik verdi som antall liv
* Kalle på en array som har elementer som skal tegnes

}

Sette opp canvas:

Canvas = function() {

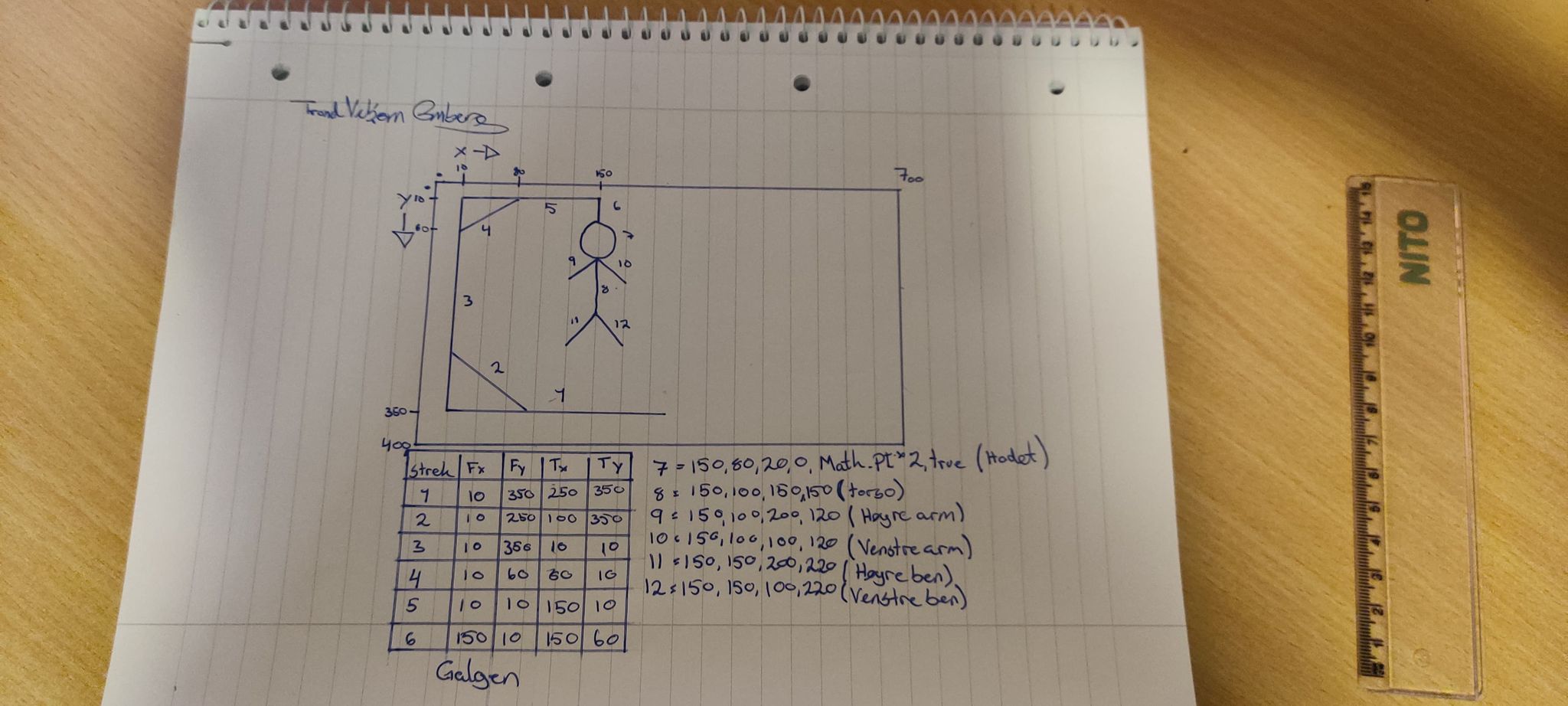
* Hente HTML-canvaset gjennom id
* Hente contexten av HTML-canvaset
* Starte tegningen ( beginPath(); )
* Sette tilpasninger til strekenes utseende.

}

Ved å starte tegningen i canvas, kan jeg opprette funksjoner på elementene som skal tegnes med moveTo (x-verdi, y-verdi)(start), lineTo (x-verdi, y-verdi)(slutt) og stroke (tegn/slå strek).

Neste steg er å tegne galgen og mannen. Canvaset har koordinater 0,0 – 700,400 da det er 700 pixler bredt og 400 pixler høyt. Dette er satt i HTML-strukturen.

Jeg må tegne de forskjellige delene som skal være med.



Galgen består av seks streker. Dette skrives inn i en funksjon som jeg kaller for ‘’gallow’’.

Gallow = function() {

* + moveTo 10, 350
  + lineTo 250, 350
  + moveTo 10, 250
  + lineTo 100, 350
  + moveTo 10, 350
  + lineTo 10, 10
  + moveTo 10, 6
  + LineTo 80, 10
  + moveTo 10, 10
  + lineTo 150, 10
  + moveTo 150, 10
  + lineTo 150, 60
  + stroke

}

Hodet er en sirkel som har en radius på 20 pixler. Den skal vises under siste strek fra galgen, men ikke krysse den. Galgen slutter med 60, så legger vi til 20, og får startpunkt 80. start-vinkel er 0, sluttvinkel er Pi\*r^2, retningen settes til true.

Head = function() {

* Hente canvaset – Dette er for å kun få en sirkel (uten start-strek)
* Hente contexten – Dette er for å kun få en sirkel (uten start-strek)
* Starte tegningen ( beginPath(); )
* Slå arc(150, 80, 20, 0, Math.PI\*2, ture)
* stroke

}

Torso = function() {

* + moveTO 150, 100
  + lineTo 150, 150
  + stroke

}

rigthArm = function() {

* + moveTo 150, 100
  + lineTo 200, 120
  + stroke

}

leftArm = function() {

* + moveTo 150, 100
  + lineTo 100, 120
  + stroke

}

rightLeg = function() {

* + moveTo 150, 150
  + lineTo 200, 220
  + stroke

}

leftLeg = function() {

* + moveTo 150, 150
  + lineTo 100, 220
  + stroke

}

Lagre Kroppsdelene i en array, slik at strekene kommer når man mister ett liv

* drawBodypartsArray [
  + rightLeg
  + leftLeg
  + rightArm
  + leftArm
  + torso
  + head

]

Jeg vil ha en funksjon som oppdaterer canvas ettersom man mister liv

updateCanvas = function(){

* variabel som alltid har amme verdi som livene
* Kalle på arrayen med kroppsdelene, og sende med variabelen.

}

Knappene må sjekkes:

Jeg oppretter en variabel som sjekker om knappene er trykket, og hva som skjer med spillet ettersom knappene er rette eller gale.

checkLetter = function() {

listen blir trykket på = function() {

* + Gjettet bokstav = this.innerHTML
  + Sette på attributt calss og clicked
  + Fjerne funksjon til knapp

For-loop som teller verdiene i det valgte ordet {

Hvis indexene i det valgte ordet er lik noen av knappene {

* + - Skal den gjettede bokstaven lagres i guessedLetterArray
    - Teller legger til 1 poeng

}

}

Opprette en variabel som tar vekk 1 poeng (j)

Hvis j har en verdi på -1 {

* + Miste man ett poeng
  + Oppdater liv
  + Oppdater canvas

} ellers {

* + Oppdater canvas

}

}(listen blir trykket på - slutt)

}

Jeg vi ha en funksjon som starter selve spillet, velger ord, og plasserer elementer

playGame = function() {

* + Ordliste – array med alle ordene
  + Tilfeldig valgt ord (ordliste.floor(Math.random()\*ordliste.length))
  + Nullstille guessedLetterArray
  + Stille liv til 6
  + Nullstille teller
  + Nullstille mellomrom
  + Kalle på alfabet-knappene
  + Kalle på funksjonen som viser det valgte ordet
  + Kalle på funksjonen som oppdaterer canvas
  + Kalle på galgen som skal vises før spillet starter

}

* Kalle på playGame

Til slutt skal spilleren nullstille spillet ved å trykke på en knapp.

Kalle på knappen fra HTML-dokumentet = function() {

* Fjerne alt av lister
* Fjerne alt av innhold
* Fjerne innholdet i canvas
* Kalle på funksjonen playGame for å re-skape spillet

}

Jeg har gjennom prosjektet benyttet meg av følgende kilder:

* https://www.freecodecamp.org/news/master-the-art-of-looping-in-javascript-with-these-incredible-tricks-a5da1aa1d6c5/
* <https://codepen.io/cathydutton/pen/ldazc>
* https://www.w3schools.com/tags/canvas\_beginpath.asp
* Fagbok: Programmering i JavaScript (2016):
  + Arrayer s. 118 - 138
  + For-løkker s. 104, 122
  + Canvas s. 55 - 58

)