***PG3302 Software Design Eksamen***

Da vi fikk oppgaven så fikk vi raskt satt opp hvordan vi skulle fullføre oppgaven som ble gitt. Det første vi gjorde var å se på uml til løsningen slik at det skulle bli lettere å kode siden vi hadde en struktur å se på. Vi brukte planen for emnet for å få et overblikk over hva vi måtte ha med for å få en god nok løsning ved innlevering. Det første vi gjorde var å få til et møte over discord for å planlegge hvordan vi skulle løse oppgaven og noterte oss hva vi måtte ha med i koden vår. Etter møtet så begynte vi å gå gjennom forelesningen og oppgaver gitt av foreleser for å ha best mulig kunnskap og overblikk over oppgaven. Deretter hadde vi et møte hvor vi la en strategi som gjorde at vi kunne disponere tiden best mulig.

Vi valgte først å disponere tiden vår på oppgaven slikt at en skrev grunnkoden for å få et overblikk over hvordan koden oppførte seg og deretter var alle med å legge til nødvendig kode for å få programmet til å oppføre seg slik løsningen krever. Tross situasjonen vi befinner oss i henhold til pandemien har vi klart å holde en god kommunikasjon på Snapchat, Discord og Messenger. Vi har brukt GitHub for å dele kode mellom oss og det har fungert veldig bra. Når vi først hadde begynt på koden så kom vi over noen feil og mangler, men dette ble løst fort ved at alle på gruppen hjalp hverandre med å løse feil.

Dette er vårt førsteutkast av vårt use case diagram som vi lagde før vi begynte på med kodingen for å få et bilde av hvordan programmet skulle se ut.

Når vi først begynte kodingen så begynte som sagt tidligere en på gruppen å legge til nødvendig kode for å ha et program å kode etter. Uten noe input og en kode som ikke ga så mye mening. Deretter begynte den seriøse programmeringen hvor vi først la til egne klasser som har en direkte lenke til Program.cs. Vi møtte noen problemer på veien ved små kodefeil, men dette ble raskt løst. Siden vårt første utkast i UML til løsningen ikke var helt gunstig måtte vi legge til flere faktorer for å få en UML som passet best til vårt program.

Når hardkodingen var over, måtte vi finne ut hvordan vi kunne gjøre noen endringer for å svare best mulig på oppgaven. Det vi gjorde for å klare dette var å ha jevnlige møte over discord for å hjelpe hverandre mens vi kodet. Og vi kom fram til en god løsning som samsvarte med oppgaveteksten. Deretter begynte jobben med å legge til fancy features som vi syntes var utrolig underholdende og lærerikt. Etter vi var ferdig lagde vi klassediagram og sekvensdiagram for å se om vi manglet noe og for å gi de som ikke har vært med på koden et godt overblikk om hvordan koden fungerer.

Følgende ting fra pensum har vi tatt med i leveringen:

* UML
  + Use Case Diagram
  + Class Diagram
  + Sequence Diagram
* Solid Prinsipper
  + Single-responsibility principle (SRP)
* Decorator pattern
* Dependency inversion principle (DIP)
* Design Patterns
* Multithreading
* Testing

Noen spesielle utfordringer vi hadde var at vi ikke hadde tilgang til å møtes som en gruppe på grunn av restriksjoner satt av kommunen, men vi gjorde vårt beste med bruk av sosiale medier. I koden så hadde vi noen spesielle utfordringer ved at det å definere hvor mange spillere man kunne ha, og genering av spillere som trekker og kaster kort uten at programmet krasjer, men dette klarte vi å fikse til slutt. En stor utfordring vi hadde var at vi ikke kunne bruke dynamisk variabelt navn i koden og fant egentlig ikke en relevant løsning til koden for å ha muligheten til å bruke dynamiske variabelnavn.

En metode vi har kalt GameLoop() er vi spesielt fornøyd med fordi den har lite kode og fungerer optimalt til vår løsning. Så vi føler den er godt programmert siden den krever lite kode på grunn av hvordan vi har kodet den. Noen andre ting vi er fornøyde med er hvordan vi klarte å bruke threading i koden vår og hvordan vi klarte å vise at vi behersker dette og vise hvordan vi viste at tester kan kjøres i vårt program.

Noen fancy features vi har med er …

Kjente bugs i programmet er følgende

**Kilder:**

Vi har ikke brukt noe materiell fra nettet, men det vi har brukt er forelesningene for å få en forståelse for hvordan vi skal løse eksamensoppgaven. Det har vært ren inspirasjon siden koden vår er 100% egenprodusert. Men kilder vi har brukt er listet følgende.

**Kildehenvisning:**

* Per Lauvås, (2020) forelesninger fra emnet “Software Design”
* UML diagrammer <https://www.uml-diagrams.org/use-case-reference.html>
* <https://en.wikipedia.org/wiki/Unified_Modeling_Language>
* <http://uml.org/>
* Tips til koding <https://stackoverflow.com/>

