

Dokumentaatio Island Game projektille

Samu Reilimo, student number: 245948, samu.reilimo@student.tut.fi

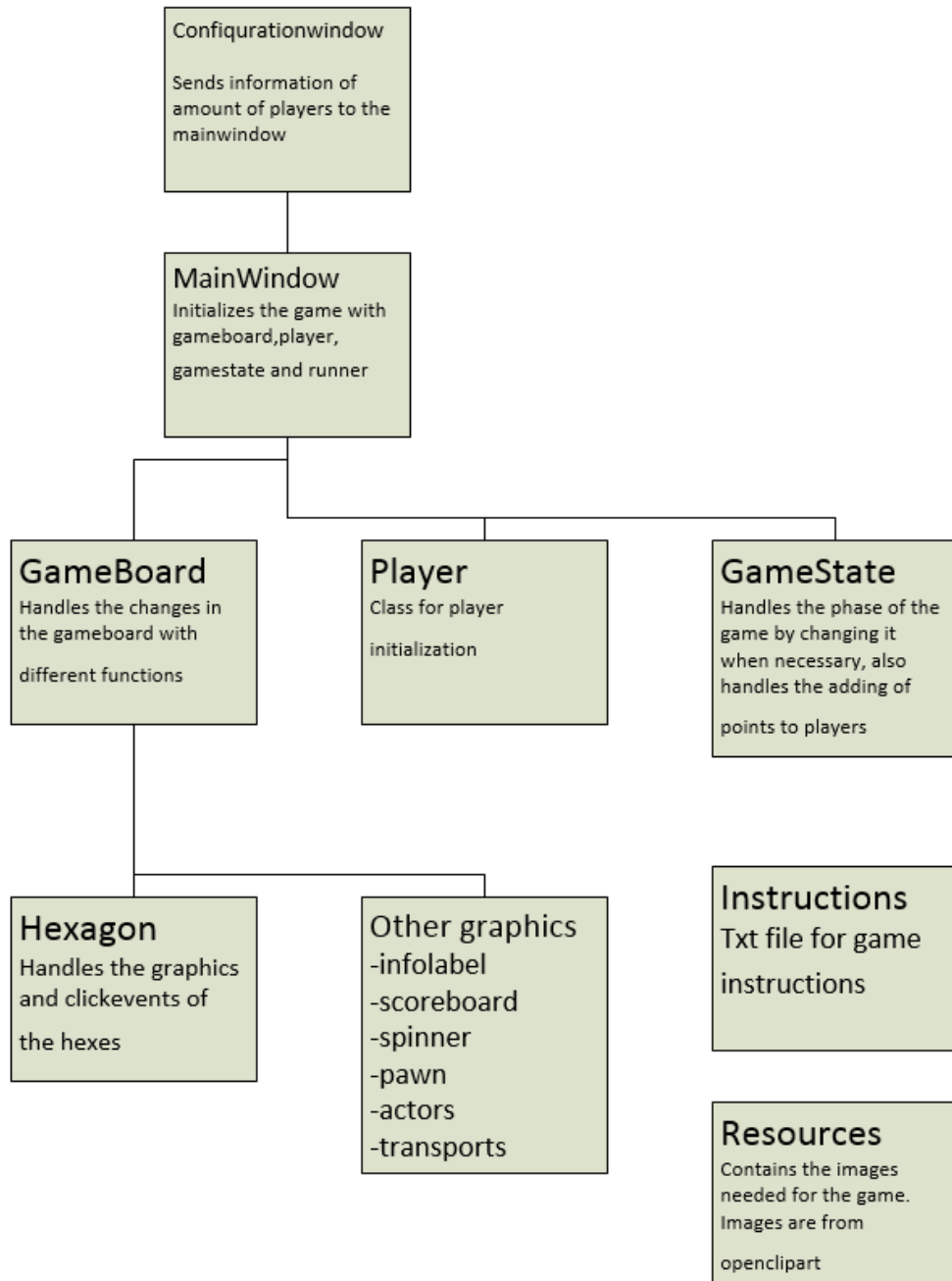
Henri Pulkkinen, student number: 258141, [henri.pulkkinen@student.tut.fi](mailto:henri.pulkkinen@student.tut.fi)

Island Game

Ohjelmoinnin tekniikat - IslandGame project

## 1. Ohjelman rakenne:

Tässä karkeasti ohjelman rakenne. Tärkeimmät luokat eriteltynä ja muut ei niin toiminnalliset, eli lähinnä grafiikoiden luokat jaoteltu erikseen.



## 1. Pelin säännöt (englanniksi) :

This is a game where the main goal is to save the three pawns given to the player at the beginning of the game, where player gets points for each saved pawn. The game begins with each player having 3 pawns randomly initialized on the beach of the island. Once all the pawns have either been saved or died to the dangers lurking deep in the sea, the game ends and the player with the most points scored is declared winner. Minimum of two players are required to play and maximum of six players can play at a time. Saving a pawn is done by getting the pawn to one of the coralhexes in the edge of the gameboard. By doing this the player is granted 1 point. So the maximum points per player can be 3 should the player be able to save all of his/her pawns. Each turn consists three parts. First a player moves his/her pawns. Then the player sinks one tile of the gameboard. Lastly the player spins a wheel, which points which animal moves and how much, giving the player ability to move an animal of that type. The 3rd phase is skipped if there isn't such animal to be found yet. Then next player starts his/her own turn. Each turn player has a 3 moves which can be used to move pawns. If player's pawn is in water swimming, moving that specific pawn will cost all 3 moves as swimming is tough. However there are transport methods that might help the player. Unfortunately there are also monsters lurking in the deep waters that might be harmful or even fatal to the player. These transports and monsters are following:

Boat: a boat can fit three pawns. The player with the most pawns in the boat can move the boat. In case of a tie, any player with pawns in the boat can move the boat. A pawn can be moved to boats on the neighbouring sea tile.

Dolphin: helps a swimmer. A swimmer with the dolphin can be moved three moves in a turn. A dolphin can help only one swimmer at a time

Shark: eats swimmers. The swimmers on the tile are removed from the game

Kraken: destroys the transports. Passengers are dropped into the sea.

Sea monster: eats swimmers and transports.

Vortex: destroys everything in the neighbouring tiles. Pawns, animals and boats on the tiles are removed from the game.

Movement is done by first doubleclicking the pawn or actor the player wants to move and then doubleclicking the target hex where the player wants to move to.

## 2. Tärkeimpien luokkien vastuujako

Tärkeimmät luokat ovat GameBoard, GameState, Player ja Hexagon. Hexagon käsittelee klikkaukset ja välittää niistä tiedon muille luokille. GameState luokka pitää tiedon pelin tilasta ja vie sen moottorin avulla muille luokille. Player alustaa pelaajat ja näiden liikemäärät. GameBoard hoitaa asiat jotka pelilaudalla tapahtuu vektoreiden ja mappien avulla, esim kun nappulaa siirretään laudalla.

### 3. Ohjelman toiminta lyhyesti

Ohjelma toimii siten, että aluksi konfiguraatioikkuna aukeaa, josta välitetään eteenpäin tiedot pelaavien pelaajien määrästä. Tämän jälkeen GameBoard, GameState ja Playerit, sekä muut pienemmät toiminnallisuudet, kuten infolabelit alustetaan mainwindowissa. Tämän jälkeen ohjelma toimii lähinnä GameBoardissa ja hexagon luokassa, jossa käsitellään pelilaudalla tapahtuvat liikkeet ja klikkaukset. Pelin lopputulemaa eli tilannetta jossa kaikkien pelaajien nappulat ovat loppu tarkastellaan GameBoard luokassa, ja näin peli päättyy silloin, kun kaikkien pelaajien nappulat ovat loppuneet ja voittajaksi julistetaan pelaaja, joka on kerännyt eniten pisteitä.

### 4. Sovittu ja toteutunut työnjako

Työnjako alustavasti sovittiin niin, että toinen toteuttaa GameBoardin funktiot ja toinen Playerin ja GameStaten luokat. Tarkoitus oli tästä eteenpäin jatkaa niin, että kun syntyy tarve uudelle luokalle, se toteutetaan sen henkilön toimesta, joka tätä luokkaa tarvitsi. Toteutunut työnjako kuitenkin poikkesi paljolti aluksi sovitusta, sillä usein ongelmiin törmätessä tarvitsimme molempien tekijöiden ajatusta ongelman ratkaisemiseksi. Työtä usein tehtiinkin yhdessä, siten että kommunikointiin jatkuvasti siitä mitä toinen tekee.

### 5. tiedossa olevat ongelmat ja puutteet

Ongelma 1: Pelin alustuksessa nappulat menevät visuaalisesti päällekkäin, mutta pelin edetessä ne kuitenkin toimivat oikein. Jotain siis alustuksessa on virheellistä, jota emme ehtineet nyt korjata.

Ongelma 2: Tilanne, jossa actori voi jäädä ”lammikkoon” jumiin ei ole käsitelty. Tämä on kuitenkin hyvin marginaalinen tilanne pelin etenemisen kannalta, eikä tähän tilanteeseen järkevällä pelaamisella ole tarkoitus pyrkiä.

Ongelma 3: Mikäli moottorin antama spinnerin result ei vielä ole näkyvissä pelilaudalla, ei spinnerin animaatio käynnisty ollenkaan. Ei vaikuta siis mitenkään pelin kulkuun, mutta pelaajalle saattaa jäädä epäselkeäksi miksi spinnerin visuaalinen ”pyörähdys” ei aina tapahdu.

Ongelma 4: Vortex tuhoaa kaiken ympäriltä olevista hexeistä, hexien tyypistä riippumatta, vaikka alustavissa pelisäännöissä puhuttiin vain meriruuduissa. Toiminto tulee kuitenkin kurssin puolelta tällaisena ja päätimme toteuttaa sen niin, että vortex käännettäessä tuhoaa kaiken ympäriltään.