|  |
| --- |
|  |
| Dokumentaatio |
|  |
| Peli prototyyppi |

**Helka, mursu ja miikka**

10. huhtikuuta 2017

Tekijä: Hahkio Helka

# Sisältö:

## Tarina s.2

## Tasot s.2

## Rakenne s.3

### Luontilauseet s.6

Dokumentaatio

Peli prototyyppi

Halusimme tehdä seikkailupelin, mutta alkuperäinen idea siitä oli liian hankala tehdä meidän taidoilla, joten me päätimme yksinkertaistaa suunnitelmaa ja saimme sopivan suunnitelman peliä varten.

Työn jako päätettiin tehdä näin:

Helka: Vastuussa visuaalisesta puolesta

Miikka: Vastuussa koodaamisesta

Mursu: Vastuussa koodaamisesta

## Pelin tarina

Pelin päähenkilö on haaksirikkoutunut autiosaarelle ja ilman mitään tapaa päästä pois.

Saarella on luola ja se on täynnä aarteita. Päähenkilöllä ei ole muuta vaihtoehtoa kuin mennä luolaan ja löytää tapa päästä saarelta pois. Kun pelaaja kulkee luolassa alaspäin, hän löytää useita eri työkaluja ja esineitä mitä hän voi käyttää.

Luolassa kuhisee hämähäkkejä ja muita hirviöitä. Alussa Päähenkilöllä ei ole aseita ja voi vain juosta hirviöitä karkuun.

## Teko

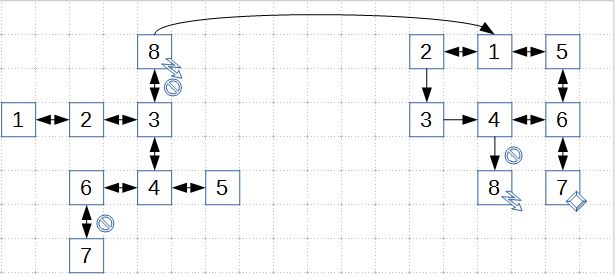
Aloitimme projektin tekemällä Ilkan opettamalla pelin pohjalla taulukkopelin, missä hahmo liikkuu ruudulta toiselle. Seuraavaksi teimme tasosuunnitelman ja aloitimme huoneiden suunnittelun. Samalla Helka piirsi hahmon ja muut kuvat peliä varten. Mursu koodasi peliä ja lisäsi huoneita pelattavaksi.

## Tasot

Yhdessä tasossa on 8 huonetta ja osa huoneiden ovista ovat kiinni ja niihin tarvitaan avain. Huoneissa on aarteita mitä pelaaja voi kerätä ja saada parhaan mahdollisen tuloksen. Osa huoneista on sellaisia, että niistä pääsee jatkamaan vain yhteen suuntaan ja samaan huoneeseen ei pääse takaisin

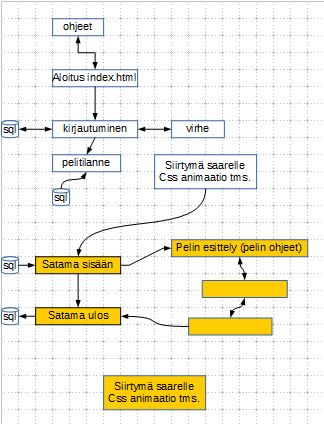
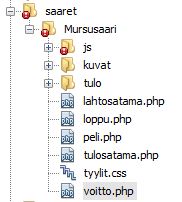
# Rakenne

Peli on rakenteeltaan taulukkopeli ja se on tehty html5 ja javascript- koodilla

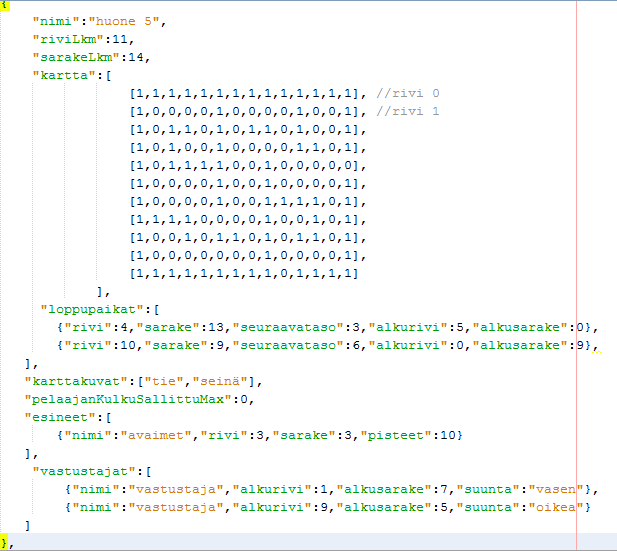


Taso 2

Taso 1



### Alla on taso 1 huone 6 koodina



**Tässä on tason rakenteen selitys:**

***”nimi”:”huone 5”,***

kertoo minkä niminen huone on.

***”riviLkm”:11,***

***”sarakeLkm”:14,***

kertoo kuinka monta riviä ja saraketta kartan taulukossa on.

*”kartta”:[*

*[1,1,1,1,1,1,1,1,1,1,1,1,1,1],*

*[1,0,0,0,0,1,0,0,0,1,0,0,0,1],*

*[1,0,1,1,0,1,0,1,0,1,0,1,0,1],*

*[0,0,1,0,0,0,0,1,0,0,0,1,0,1],*

*[1,1,1,1,0,1,0,1,0,1,1,1,1,1],*

*[1,0,0,0,0,1,1,0,0,1,0,0,0,0],*

*[1,1,1,1,0,1,0,0,0,1,0,1,1,1],*

*[1,0,0,1,0,1,0,1,1,1,0,0,0,1],*

*[1,0,1,1,0,1,0,1,0,1,0,1,0,1],*

*[1,0,0,0,0,1,0,0,0,0,0,1,0,1],*

*[1,1,1,1,1,1,1,1,1,1,1,1,1,1]*

*],*

Huoneen kartta taulukkomuodossa.

*”loppupaikat”: [*

*{”rivi”:4,”sarake”:13,”seuraavataso”:3,”alkusarake”:0}*

*{”rivi”:10,”sarake”:9,”seuraavataso”:6,”alkurivi”:0,”alkusarake”:9}*

Päättää minne pelaaja viedään, kun hän astuu tiettyyn kohtaan.

*”karttakuvat”:[”tie”,”seinä”],*

Tähän tulee kuvat , jotka esiintyy kartan tietyssä ruuduissa

*”pelaajanKulkuSallittuMax”:0,*

Kertoo minkä päällä ja läpi pelaaja voi kulkea

*”esineet”: [*

*{”nimi”:”avaimet”,”rivi”:3,”sarake”:3,”pisteet:10}*

*],*

Kaikki esineet huoneessa mitkä voi kerätä esim. tässä on avain.

*”vastustajat”:[*

*{”nimi”:”vastustaja”,”alkurivi”:1,”alkusarake”:7,”suunta”:”vasen”},*

*{”nimi”:”vastustaja”,alkurivi”:9,”alkusarake”:5,”suunta”:”oikea”}*

*]*

Päättää mitkä vastustajat huoneessa on ja missä ne ovat.

# Luontilauseet

**-- MySQL Workbench Forward Engineering**

**SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;**

**SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;**

**SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='TRADITIONAL,ALLOW\_INVALID\_DATES';**

**-- -----------------------------------------------------**

**-- Schema saaristo**

**-- -----------------------------------------------------**

**DROP SCHEMA IF EXISTS `saaristo` ;**

**-- -----------------------------------------------------**

**-- Schema saaristo**

**-- -----------------------------------------------------**

**CREATE SCHEMA IF NOT EXISTS `saaristo` DEFAULT CHARACTER SET utf8 ;**

**USE `saaristo` ;**

**-- -----------------------------------------------------**

**-- Table `saaristo`.`Pelaaja`**

**-- -----------------------------------------------------**

**DROP TABLE IF EXISTS `saaristo`.`Pelaaja` ;**

**CREATE TABLE IF NOT EXISTS `saaristo`.`Pelaaja` (**

**`nimimerkki` VARCHAR(45) NOT NULL,**

**`salasana` VARCHAR(45) NOT NULL,**

**`rahamaara` INT NOT NULL,**

**PRIMARY KEY (`nimimerkki`))**

**ENGINE = InnoDB;**

**-- -----------------------------------------------------**

**-- Table `saaristo`.`Saari`**

**-- -----------------------------------------------------**

**DROP TABLE IF EXISTS `saaristo`.`Saari` ;**

**CREATE TABLE IF NOT EXISTS `saaristo`.`Saari` (**

**`saariID` INT NOT NULL AUTO\_INCREMENT,**

**`saarenNimi` VARCHAR(255) NOT NULL,**

**`saarenURL` VARCHAR(255) NOT NULL,**

**PRIMARY KEY (`saariID`))**

**ENGINE = InnoDB;**

**-- -----------------------------------------------------**

**-- Table `saaristo`.`Pelaaja\_vieraillut\_Saari`**

**-- -----------------------------------------------------**

**DROP TABLE IF EXISTS `saaristo`.`Pelaaja\_vieraillut\_Saari` ;**

**CREATE TABLE IF NOT EXISTS `saaristo`.`Pelaaja\_vieraillut\_Saari` (**

**`Pelaaja\_nimimerkki` VARCHAR(45) NOT NULL,**

**`Saari\_saariID` INT NOT NULL,**

**PRIMARY KEY (`Pelaaja\_nimimerkki`, `Saari\_saariID`),**

**INDEX `fk\_Pelaaja\_has\_Saari\_Saari1\_idx` (`Saari\_saariID` ASC),**

**INDEX `fk\_Pelaaja\_has\_Saari\_Pelaaja\_idx` (`Pelaaja\_nimimerkki` ASC),**

**CONSTRAINT `fk\_Pelaaja\_has\_Saari\_Pelaaja`**

**FOREIGN KEY (`Pelaaja\_nimimerkki`)**

**REFERENCES `saaristo`.`Pelaaja` (`nimimerkki`)**

**ON DELETE NO ACTION**

**ON UPDATE NO ACTION,**

**CONSTRAINT `fk\_Pelaaja\_has\_Saari\_Saari1`**

**FOREIGN KEY (`Saari\_saariID`)**

**REFERENCES `saaristo`.`Saari` (`saariID`)**

**ON DELETE NO ACTION**

**ON UPDATE NO ACTION)**

**ENGINE = InnoDB;**

**-- -----------------------------------------------------**

**-- Table `saaristo`.`Saari\_reitti\_Saari`**

**-- -----------------------------------------------------**

**DROP TABLE IF EXISTS `saaristo`.`Saari\_reitti\_Saari` ;**

**CREATE TABLE IF NOT EXISTS `saaristo`.`Saari\_reitti\_Saari` (**

**`Saari\_saarestaID` INT NOT NULL,**

**`Saari\_saareenID` INT NOT NULL,**

**`hinta` INT NOT NULL,**

**PRIMARY KEY (`Saari\_saarestaID`, `Saari\_saareenID`),**

**INDEX `fk\_Saari\_has\_Saari\_Saari2\_idx` (`Saari\_saareenID` ASC),**

**INDEX `fk\_Saari\_has\_Saari\_Saari1\_idx` (`Saari\_saarestaID` ASC),**

**CONSTRAINT `fk\_Saari\_has\_Saari\_Saari1`**

**FOREIGN KEY (`Saari\_saarestaID`)**

**REFERENCES `saaristo`.`Saari` (`saariID`)**

**ON DELETE NO ACTION**

**ON UPDATE NO ACTION,**

**CONSTRAINT `fk\_Saari\_has\_Saari\_Saari2`**

**FOREIGN KEY (`Saari\_saareenID`)**

**REFERENCES `saaristo`.`Saari` (`saariID`)**

**ON DELETE NO ACTION**

**ON UPDATE NO ACTION)**

**ENGINE = InnoDB;**

**SET SQL\_MODE=@OLD\_SQL\_MODE;**

**SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;**

**SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;**