* saako käyttää mitä tahansa kuvia/ääntä/videota?
* miksi tarviin järj.valvojaa?

http://untamed.wild-refuge.net/rmxpresources.php?characters

**Tornipuolustus**

**Yleiskuvaus**

Projektin aiheena on pikkupeliformaatti (tower defence), jossa erilaiset viholliset yrittävät päästä ennalta määritettyä reittiä pitkin maaliin. Pelaajan tavoitteena on estää vihollisten liike asettamalla reitin läheisyyteen torneja, jotka ampuvat tai muuten häiritsevät vihollisten liikettä.

Pelin tarkoituksena on kerätä pisteitä ja edetä tasoissa eteenpäin. Pelaaja saa pisteitä tuhoamistaan vihollisista ja etenee seuraavalle tasolle, kun tason jokainen vihollinen on joko tuhottu, tai päässyt karkuun, eli maaliin. Kun kymmenen vihollista on päässyt karkuun pelaajan torneista huolimatta, peli päättyy.

Pelin vaikeusaste muuttuu luonnollisesti pelin edetessä. Vihollisten määrä ja voimakkuus kasvavat, mutta pelaajalle aukeaa samalla uusia ja tehokkaampia torneja. Pelaajan tulee kuitenkin taktikoida puolustuksensa tarkoin, sillä torneja voi rakentaa vain rajallisen määrän. Pelaaja tarvitsee tornien rakentamiseen rahaa, jota hän saa tuhoamalla vihollisia. Vaikka pelin loppupuolella pelaaja on saanut paljon rahaa, ja kyennyt siten rakentamaan paljon torneja, häneltä voi myös loppua tila tai tornien kantama kesken, sillä torneja ei voi rakentaa päällekkäin.

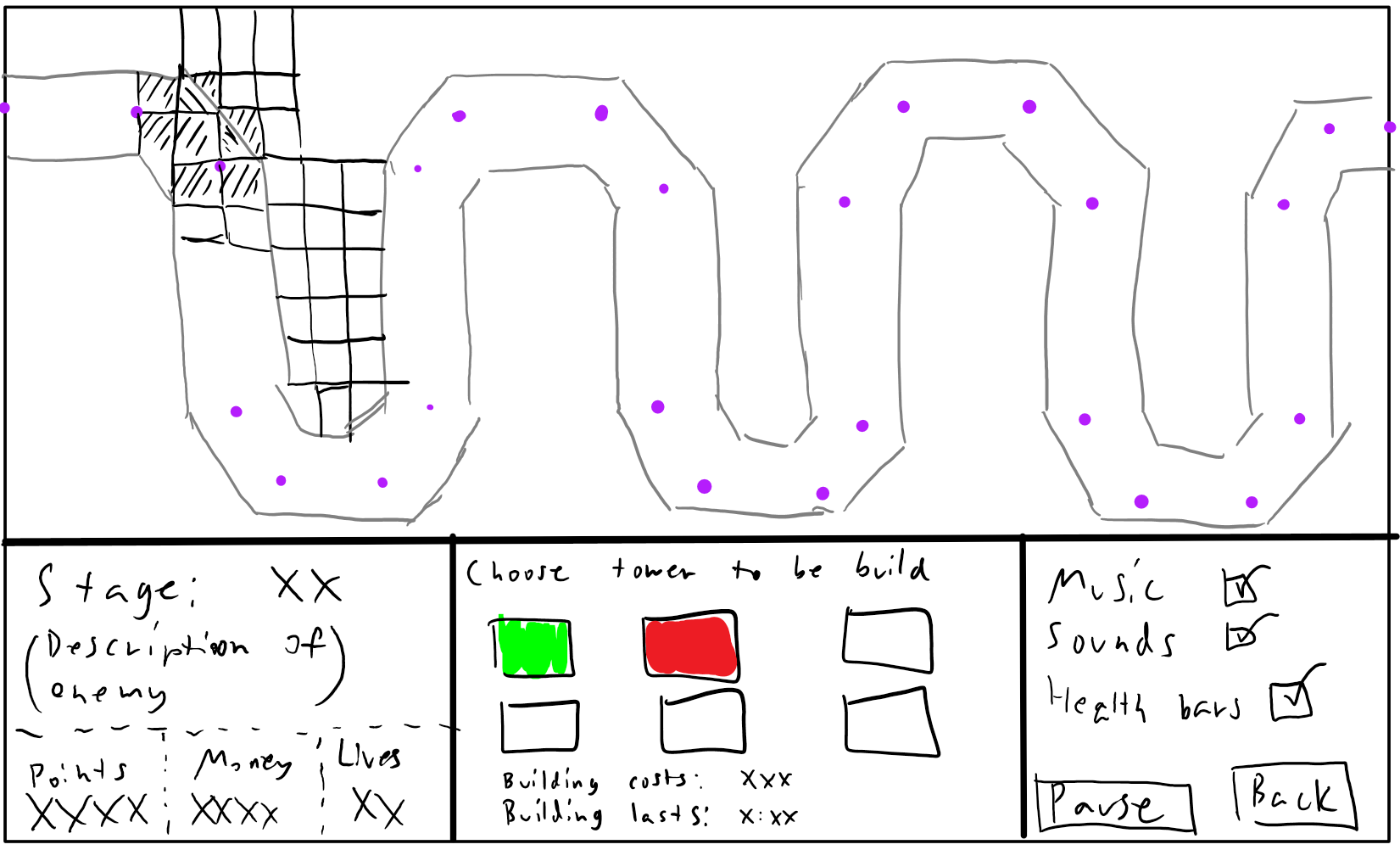
Pelaaja voi valita kolmesta eri vaihtoehdosta, ”Easy”, ”Medium” ja ”Hard”. Kentän vaikeusaste tulee suoraan siitä, että kenttä on lyhyempi, eli pelaajalla on vähemmän aikaa torjua viholliset.

**Käyttöliittymän luonnos**

Ohjelmaa, eli tässä tapauksessa peliä ohjataan hiirellä. Näppäimistöä käytetään ainoastaan pelaajan nimen syöttämiseen ennätystuloksia varten. Pelin aloitusvalikosta siirrytään peliin tai ennätyksiin klikkaamalla haluttua kohtaa. Pelissä pelaaja asettaa torneja klikkaamalla kyseisen tornin kuvaketta ja tämän jälkeen klikkaamalla haluttua kohtaa pelikentällä. Sekä pelistä, että ennätystuloksista voidaan poistua aloitusvalikkoon ”Palaa”-näppäimellä. Ohjelmaan lisätään kuitenkin mahdollisuus hyödyntää näppäimistöä, esimerkiksi ”ESC” asetetaan palaa komennoksi, ”p” taukotilaksi ja ”m” äänen hiljentämiseksi.

Torneja voidaan asettaa koko pelikentälle, pois lukien reitti

Pelin päättyessä ohjelma piirtää pelikentän päälle tekstin ”Game Over” ja pistemäärän. Lisäksi ohjelma piirtää näppäimen ”Highscores”, jolla pääsee tarkastelemaan, riittivätkö pisteet ennätyslistalle.



Miten ohjelma kommunikoi käyttäjän kanssa? Missä muodossa ja mistä ohjelma saa syötteensä, ja miten/mihin se tulostaa viestinsä? Mitä komentoja ohjelma tuntee ja miten ne annetaan? Jos ohjelman käyttöliittymä on graafinen, liittäkää mukaan jonkinmoinen kuva tärkeimmistä ikkunoista tai ainakin kuvailkaa sanallisesti, mitä ohjelmaruudulla näkyy. Kaikkia pikku yksityiskohtia ei tule tässä selittää.

**Tiedostot ja tiedostoformaatit**

Viholliset ladataan peliin kuvina

Suurin osa projektiaiheista vaatii tietojen tallentamista ja lataamista tiedostoista, ja usein myös pidemmät tekstit, kuvat jne. joita käyttöliittymissä tarvitaan on parempi säilyttää tiedostoissa kuin kirjoittaa kiinteästi koodiin. Selostakaa tässä osiossa millaisia tiedostoja ohjelmanne käsittelee, jos mitään. Esim. ovatko ne tekstitiedostoja vai binääritiedostoja, **ja miten tieto on niissä esitetty**? Kuvatkaa jo nyt tiedostoformaatit joita ohjelmanne käyttää. Vaikka formaatti voisi muuttua myöhemmin, sen pohtiminen auttaa välttämään mahdollisia ongelmia projektin toteutusvaiheessa.

**Järjestelmätestaussuunnitelma**

Projektin testausta järjestelmän tasolla voi ryhtyä suunnittelemaan saman tien kun toteutuksen ääriviivat on pohdittu. Pohjautuen täydentämäänne yleiskuvaukseen, käyttöliittymän kuvaukseen ja mahdollisiin tiedostoihin, kuvatkaa yleisluontoisesti, kuinka lopullisen ohjelman toimintaa kannattaisi testata. Menemättä liikaa toteutuksen yksityiskohtiin, pohtikaa mitä asioita ohjelman tulee pystyä tekemään ja millä tavoin näitä voisi testata. Mitä asiota ohjelman ei tule sallia? Kuinka tätä voisi testata?

Esimerkki testattavasta ominaisuudesta on tekstinkäsittelyohjelman find-komento. Komennon tulisi löytää kaikki annetun tekstin esiintymät kursorin nykyisestä paikasta eteenpäin. Toimintoa tulisi lopullisessa ohjelmassa testata ainakin tapauksilla joissa kursorin nykyisen paikan jälkeen tulee useampia esiintymiä, yksi esiintymä tai ei yhtään. Erikoistapauksina kannattaa kokeilla etsittävän asian sijainti koko tekstin lopussa tai juuri kursorin kohdalla. Myös kursoria edeltäviä tapauksia tulee kokeilla.

Koska ohjelma ei ole vielä täysin muotoutunut, ei tässä kannata liioitella. Testitapauksia kannattaa suunnitella vain sen verran, että ohjelman keskeisimmät toiminnot tulee ainakin ajateltua läpi. Kun on pohtinut mitä ohjelmaltaan vaatii ei toteuttaessa tule ehkä niin paljon yllätyksiä.