



# NYSSEOSU

Dokumentaatio

## TIIVISTELMÄ

Ohjelman toimintaa ja suunnittelua selventävä dokumentaatio

Juho Kuusisto, Teemu Pöytäniemi

TIE-02400

# 1 SISÄLLYS

---

2	Pelin idea ja käyttöohjeet.....	2
2.1	Pelin Idea .....	2
2.2	Käyttöohjeet .....	2
2.2.1	Rynnäkkökivääri.....	2
2.2.2	Maamiina.....	2
2.2.3	Laser.....	2
3	Pelin Toteutus.....	3
3.1	Ohjelman toiminta.....	3
3.2	Luokkien vastuujaako .....	3
3.2.1	Kaupunki .....	3
3.2.2	Peliikkuna .....	3
3.2.3	PeliAlue .....	3
3.2.4	Drooni .....	3
3.2.5	Ammus (Sekä siitä periytyt luokat) .....	4
3.2.6	Nyssekuva .....	4
3.2.7	LopetusDialog .....	4
3.2.8	Tilasto .....	4
3.3	Lisäominaisuudet.....	4
3.4	Sopimussuunnittelun käyttö .....	4

## 2 PELIN IDEA JA KÄYTTÖOHJEET

---

### 2.1 PELIN IDEA

Pelissä pelaaja liihoittelee Tampereen yllä mahtavalla droonillansa. Hän vaanii kaupungin kaduilla ajavia nyssejä, ja pyrkii ammusvarastonsa rajoissa aiheuttamaan mahdollisimman paljon tuhoa. Droni on varusteltu rynnäkkökiväärillä, maamiinoilla ja laserilla. Aseiden tarkempi toiminta esitellään myöhemmin.

Pelissä ei ole aikarajaa, vaan tarkoitus on nerokkaan sijoittautumisen avulla tuhota mahdollisimman monta nysseä (ja niissä olevia matkustajia). Yhdestä tuhotusta nyssestä saa 50 pistettä, ja jokaisesta kyydissä olleesta matkustajasta yhden lisäpisteen.

### 2.2 KÄYTTÖOHJEET

Jottei pelaamista rajoittaisi puutteellinen tietämys droonin kontrolleista, kerrottakoon ne tässä.

Pelin alkaessa pelaajan silmien eteen avataan droonin taktinen taistelupaneeli, tuttavallisemmin käyttöliittymä. Dronia voi ohjata kolmella metodilla; käyttäen taktisen taistelupaneelin ohjauspainikkeita, valiten droonin ja tämän jälkeen nuolinäppäimillä tai klikkaamalla kartalta haluttua määränpäättä. Käytettäessä viimeistä metodia droonin ajotietokone laskee optimaalisen reitin siirtymistä varten ja suorittaa siirtymän.

Pahojen nyssejen vallan kitkemiseksi droonissa on tehokas aseistus. Pelaaja valitsee haluamansa aseenn droonin taktisen taistelupaneelin oikeasta laidasta löytyvästä The Ultimate Weapon Armory asevalikosta. Aseen laukaisu tapahtuu joko "Ammu" painikkeella tai välilyönnillä. Pelaaja voi halutessansa ampua useita laukauksia nopeasti sarjassa joko rävakällä ranneliikkeellä tai pitämällä välilyöntiä pohjassa. Asevalikossa näytetään myös reaaliajassa jäljellä olevien ammusten määrä. Alla on esitelty aseiden tyypillinen toiminta.

#### 2.2.1 Rynnäkkökivääri

Rynnäkkökivääri on tehoton, rajatulla kantamalla toimiva yleiskäyttöinen ase, joka kykenee tuhoamaan yhden nysnen. Rynnäkkökiväärin ammuksen nopeus on melko hidas, joten osuminen vaatii tarkkuutta.

#### 2.2.2 Maamiina

Maamiina on nimensä mukaisesti miina. Ammuttaessa miina jätetään droonin alle maahan. Miina räjähtää, kun nysse ajaa sen ylitse. Räjähtäessään miina tuhoaa kaikki lähellä olevat nysset.

#### 2.2.3 Laser

Pelin tehokkain ase on laser. Laserin kantama on koko pelialueen mittainen ja laser etenee valonnopeudella – käytännössä välittömästi. Laser tuhoaa kaikki reitillensä tulevat nysset.

## 3 PELIN TOTEUTUS

---

### 3.1 OHJELMAN TOIMINTA

Ohjelma luo käynnistyessään pelin tarvitsemat oliot. KurssinPuolen kutsuessa luoPeli() funktiota luodaan uusi KaupunkiRP:n toteuttava Kaupunki luokan olio. Kaupunki oliota luotaessa luodaan Peliikkuna, Tilasto ja Drooni. Vastaavasti Peliikkuna luo rakentajassaan PeliAlue olion, johon varsinainen peli sijoittuu.

Kun KurssinPuolen logiikka lisää kaupunkiin pysäkkejä, luodaan pysäkkejä kartalla esittävät oliot ja talletetaan pysäkit ja niitä vastaavat kuvat kaupunkiolion private osaan.

Nysset luodaan myös KurssinPuolen kaupunkioliolle lähettämällä käskyllä. Nyssejä varten on toteutettu oma luokkansa, Nyssekuva. Nyssepoinntterit ja Nyssekuvapointterit talletetaan unordered\_map:iin kaupunkiolion private osaan. Kaupunkiolion tehtävä on hallita kaupungissa olevia toimijoita.

Pelin pelaaminen, eli droonilla liikkuminen ja ampuminen tapahtuvat ilman välikäsiä. Peliikkuna välittää ohjauskomennot droonille ja drooni niiden mukaan liikuttaa itseänsä kartalla. Ammuttaessa drooni luo uuden ammusolion (Rynnäkkökivääri, Maamiina tai Laser) ja lisää sen PeliAlueelle.

Ammus on itsenäinen olio pelissä. Ammuksella on oma ajastin jonka avulla ammus liikkuu sulavasti PeliAlueella ja tarkistaa onko osuma tapahtunut tasaisin väliajoin. Osuman tapahtuessa ammus ilmoittaa kohteelle osumasta ja tuhoaa itsensä. Kohde ilmoittaa mahdollisesta tuhoutumisestaan Kaupungille.

Kaupunki ilmoittelee Tilastolle pelin tapahtumista ja pyytää tilastolta päivityksiä reaaliaikasta tilastointia varten.

Drooni lähettää signaalin kun sen ammukset ovat lopussa, ja näin ollen Kaupunki saa tiedon ja ajoittaa pelin lopetuksen.

Syvällisemmän ymmärryksen saavuttamiseksi suosittelemme tämän asiakirjan liitteenä olevan Doxygen dokumentaation lukemista ja lähdekoodin pohdiskelua. Vastaamme myös mielellämme pelin toimintaa koskeviin kysymyksiin sähköpostitse.

### 3.2 LUOKKIEN VASTUUNJAKO

#### 3.2.1 Kaupunki

Ns. "pääjehu", pitää kirjaa pelissä olevista pysäkeistä, nysseistä, matkustajista ja muista toimijoista. Omistaa Peliikkunan ja Tilaston. Päättää pelin kun droonin ammukset loppuvat.

#### 3.2.2 Peliikkuna

Pelin pääikkuna, tarjoaa käyttöliittymän hallintaan liittyvän toiminnallisuuden. Luo droonin ja sijoittaa sen PeliAlueelle. Luo PeliAlueen.

#### 3.2.3 PeliAlue

QGraphicsScene:stä periytetty luokka. Omistaa pelissä olevat graafiset, QGraphicsItem:sta tai QGraphicsObject:sta periytetyt oliot, kuten Dronin, Nyssekuvat ja pysäkkikuvat. Näin koska Qt:n dokumentaatio käskee.

#### 3.2.4 Drooni

Taistelualus, pelaajan kontrolloima. Luo ammukset, ilmoittelee ammustensa loppumisesta.

### 3.2.5 Ammus (Sekä siitä periytyvät luokat)

Ammus on abstrakti kantaluokka Rynnäkökivääri:lle, Maamiina:lle ja Laser:lle. Pelin ammuksat, hoitavat osuimien selvittämisen.

### 3.2.6 Nyssekuva

Nyssejen grafiikat. Tehtävänä tuoda nysset näkyviin ja olla ammusten maalitauluina.

### 3.2.7 LopetusDialog

Pelin päättyessä esitettävä dialogi. Kertoo lopullisen tuloksen, tarvittaessa kysyy nimimerkin ja pyytää Tilastoa tallettamaan tuloksen Top10 listalle.

### 3.2.8 Tilasto

Pitää tilastoa pelin tapahtumista. Toteuttaa TilastoRP:n ja hoitaa Top10 listan lukemisen ja kirjoittamisen.

## 3.3 LISÄOMINAISUUDET

Pelissämme on toteutettuna seuraavat työohjeessa listatut lisäominaisuudet:

- Minimaalinen ruudunpäivitys
  - Toteutettu asettamalla Viewport päivittymään vain QGraphicsItemien boundingRect alueelta muutosten tapahtuessa. ks. Peliikkuna luokan rakentaja.
- Vieritettävä kartta
  - Käytetään pelilogiikan tarjoamaa isoakarttaa
- Droonin tasainen liike
  - Tähän erityisesti panostettu! Drooni kiihdyttää vauhdin tasaiseen liikkeeseen, ohjaus joko nuolilla tai peli-ikkunan painikkeilla. ks. Drooni luokan alaLiikkua ja alaKaantya slotit.
  - LISÄKSI: **Droonin nerokas sulava liikkuminen**
    - Drooni liikkuu kartalta klikattuun sijaintiin kääntyen ja liikkuen eteenpäin vain tarvittavan määrän. ks. Drooni luokan liikuSulavasti() metodi
- Pelin tilan seuranta
  - Reaaliaikainen tilasto esillä peli-ikkunassa. Pitää kirjaa koko pelissä olevista asioista. ks. Tilasto luokan paivitaTilasto(), kutsutaan Kaupungin toimijaLiikkunut metodissa. Näin tilasto on aina ajantasalla.
- Top10-lista
  - Näytetään peli-ikkunassa. Mikäli Top10 listan tiedostoa ei löydy, luo ohjelma sellaisen automaattisesta ajokansioon (etsii myös sieltä automaattisesti). ks. Tilasto luokan lue\_tiedosto() ja kirjoitaTiedostoon()
- Laaja asevalikoima
  - Kolme toisistaan reilusti poikkeavaa asetta, kullakin ainutlaatuinen osumalaskenta. ks. Ammus, Rynnäkökivääri, Maamiina ja Laser luokat

## 3.4 SOPIMUSSUUNNITTELUN KÄYTTÖ

Työssä on hyödynnetty sopimussuunnittelua joissakin määrin. Luokat joissa on hyödynnetty sopimussuunnittelua on dokumentoitu Doxygen dokumentaationa. Doxygen dokumentaation pdf version löydät tämän dokumentin liitteenä (Liite 1).

## 4 LIITTEET

---

## 4.1 DOXYGEN DOKUMENTAATIO

NysseOsu

Generated by Doxygen 1.8.5

Mon Apr 25 2016 03:02:18





# Contents

<b>1</b>	<b>Hierarchical Index</b>	<b>1</b>
1.1	Class Hierarchy	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List	5
<b>4</b>	<b>Class Documentation</b>	<b>7</b>
4.1	AlyMatkustaja Class Reference	7
4.2	Ammus Class Reference	7
4.2.1	Constructor & Destructor Documentation	8
4.2.1.1	Ammus	8
4.2.2	Member Function Documentation	9
4.2.2.1	alustaAmmus	9
4.2.2.2	asetAmmuskuva	9
4.2.2.3	etene	9
4.2.2.4	osuma	10
4.2.2.5	suuntaPiste	10
4.2.2.6	tarkistaOsuma	10
4.2.2.7	tuhoaAmmus	11
4.2.2.8	ulkoOsuma	11
4.3	Drooni Class Reference	11
4.3.1	Constructor & Destructor Documentation	13
4.3.1.1	Drooni	13
4.3.1.2	~Drooni	13
4.3.2	Member Function Documentation	13
4.3.2.1	alaKaantyaMyota	13
4.3.2.2	alaKaantyaVasta	13
4.3.2.3	alaLiikkuaEteen	14
4.3.2.4	alaLiikkuaTaakse	14

4.3.2.5	ammu	14
4.3.2.6	ammuksiaJaljella	14
4.3.2.7	drooniAmpunut	15
4.3.2.8	drooniLiikkunut	15
4.3.2.9	keyPressEvent	15
4.3.2.10	keyReleaseEvent	15
4.3.2.11	liikuSulavasti	15
4.3.2.12	lopetaEteenLiikkuminen	16
4.3.2.13	lopetaKaannosMyota	16
4.3.2.14	lopetaKaannosVasta	16
4.3.2.15	lopetaTaakseLiikkuminen	16
4.3.2.16	valitseLaser	17
4.3.2.17	valitseMiina	17
4.3.2.18	valitseRynkky	17
4.4	Kaupunki Class Reference	17
4.4.1	Member Function Documentation	18
4.4.1.1	ajastaLopetus	18
4.5	Laser Class Reference	18
4.5.1	Constructor & Destructor Documentation	19
4.5.1.1	Laser	19
4.5.1.2	~Laser	19
4.5.2	Member Function Documentation	19
4.5.2.1	asetAmmuskuva	19
4.5.2.2	tarkistaOsuma	20
4.6	LopetusDialog Class Reference	20
4.7	Miina Class Reference	21
4.7.1	Constructor & Destructor Documentation	21
4.7.1.1	Miina	21
4.7.1.2	~Miina	22
4.7.2	Member Function Documentation	22
4.7.2.1	alustaAmmus	22
4.7.2.2	asetAmmuskuva	22
4.7.2.3	osuma	23
4.7.2.4	tarkistaOsuma	23
4.8	Nyssekuva Class Reference	23
4.8.1	Constructor & Destructor Documentation	24
4.8.1.1	Nyssekuva	24
4.8.1.2	~Nyssekuva	24
4.8.2	Member Function Documentation	24
4.8.2.1	kerroKaupunki	24

4.8.2.2	osuma	25
4.8.2.3	tuhottu	25
4.8.2.4	tuhoudu	25
4.9	PeliAlue Class Reference	25
4.9.1	Member Function Documentation	26
4.9.1.1	lisaaDrooni	26
4.10	PeliIkkuna Class Reference	26
4.10.1	Constructor & Destructor Documentation	27
4.10.1.1	PeliIkkuna	27
4.10.1.2	~PeliIkkuna	27
4.10.2	Member Function Documentation	27
4.10.2.1	asetaNayttam	27
4.10.2.2	asetaNayttam	28
4.10.2.3	lisaa_nysse	28
4.10.2.4	lisaa_pysakki	28
4.10.2.5	lisaaDrooni	29
4.10.2.6	naytaToplista	29
4.10.2.7	PaivitaTulokset	29
4.11	Rynnakkokivaari Class Reference	30
4.11.1	Constructor & Destructor Documentation	30
4.11.1.1	Rynnakkokivaari	30
4.11.1.2	~Rynnakkokivaari	31
4.11.2	Member Function Documentation	31
4.11.2.1	asetaNayttam	31
4.12	Tilasto Class Reference	31
4.12.1	Member Function Documentation	32
4.12.1.1	kirjoitaTiedostoon	32
4.12.1.2	lue_tiedosto	32
4.12.1.3	paivita_tilasto	33
4.12.1.4	selvitaListasijoitus	33
<b>5</b>	<b>File Documentation</b>	<b>35</b>
5.1	ammus.hh File Reference	35
5.1.1	Detailed Description	35
5.2	drooni.hh File Reference	35
5.2.1	Detailed Description	36
5.3	kaupunki.hh File Reference	36
5.3.1	Detailed Description	36
5.4	laser.hh File Reference	36
5.4.1	Detailed Description	36

5.5	miina.hh File Reference . . . . .	37
5.5.1	Detailed Description . . . . .	37
5.6	nyssekuva.hh File Reference . . . . .	37
5.6.1	Detailed Description . . . . .	37
5.7	pelialue.hh File Reference . . . . .	37
5.7.1	Detailed Description . . . . .	38
5.8	pelikkuna.hh File Reference . . . . .	38
5.8.1	Detailed Description . . . . .	38
5.9	rynnakkokivaari.hh File Reference . . . . .	38
5.9.1	Detailed Description . . . . .	38

<b>Index</b>	<b>39</b>
--------------	-----------

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

KaupunkiRP	
Kaupunki . . . . .	17
Matkustaja	
AlyMatkustaja . . . . .	7
QDialog	
LopetusDialog . . . . .	20
QGraphicsItem	
Nyssekuva . . . . .	23
QGraphicsObject	
Ammus . . . . .	7
Laser . . . . .	18
Miina . . . . .	21
Rynnakkokivaari . . . . .	30
Drooni . . . . .	11
QGraphicsScene	
PeliAlue . . . . .	25
QObject	
Kaupunki . . . . .	17
QWidget	
Peliikkuna . . . . .	26
TilastoRP	
Tilasto . . . . .	31
ToimijaRP	
Drooni . . . . .	11



## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">AlyMatkustaja</a>	7
<a href="#">Ammus</a>	7
<a href="#">Drooni</a>	11
<a href="#">Kaupunki</a>	17
<a href="#">Laser</a>	18
<a href="#">LopetusDialog</a>	20
<a href="#">Miina</a>	21
<a href="#">Nyssekuva</a>	23
<a href="#">PeliAlue</a>	25
<a href="#">PeliIkkuna</a>	26
<a href="#">Rynnakkokivaari</a>	30
<a href="#">Tilasto</a>	31





## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

<b>alymatkustaja.hh</b>	??
<b>ammus.hh</b>	
Kantaluokka kaikille pelissä käytettäville ammuksille. Tarjoaa perustoiminnoille toteutukset. Muistin loppuessa heittää std::bad_alloc ellei poikkeustakuu ole nothrow	35
<b>drooni.hh</b>	
<b>Drooni</b> luokka toteuttaa pelissä käytettävän droonin. Periytetty QGraphicsObject ja ToimijaRP luokista	35
<b>kaupunki.hh</b>	
<b>Kaupunki</b> luokka on KaupunkiRP:stä periytetty. <b>Kaupunki</b> hallinnoi pelin tapahtumia	36
<b>laser.hh</b>	
<b>Ammus</b> luokasta periytetty luokka, määrittelee laserin toiminnan	36
<b>lopetusdialog.hh</b>	??
<b>miina.hh</b>	
<b>Ammus</b> luokasta periytetty luokka, määrittelee miinan toiminnan. Räjähdyksessä käytetyt kuvat <a href="http://wrathgames.com/blog">http://wrathgames.com/blog</a> WrathGames Studio. Muistin loppuessa voidaan heittää aina std::bad_alloc	37
<b>nyssekuva.hh</b>	
<b>Nyssekuva</b> tarjoaa graafisen ilmeen pelin nysseille. Periytetty QGraphicsItemista	37
<b>pelialue.hh</b>	
<b>PeliAlue</b> on periytetty QGraphicsScenestä pelin tarkoituksiin sopivaksi	37
<b>pelikkuna.hh</b>	
<b>Peliikkuna</b> on pelin "pääikkunan" tarjoava luokka	38
<b>rynnakkokivaari.hh</b>	
<b>Ammus</b> luokasta periytetty luokka, määrittelee rynnäkkökiväärin ammuksen	38
<b>tilasto.hh</b>	??

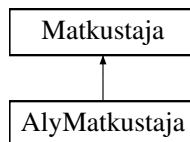


## Chapter 4

# Class Documentation

### 4.1 AlyMatkustaja Class Reference

Inheritance diagram for AlyMatkustaja:



#### Public Member Functions

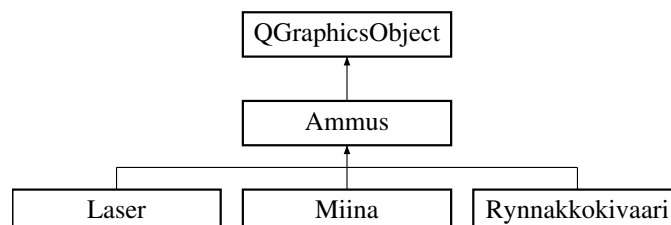
- virtual bool **nousetkoNysseen** (std::weak\_ptr< Nysse > nysse) const
- virtual bool **menetkoPysakille** (std::weak\_ptr< Rajapinta::PysakkiRP >) const

The documentation for this class was generated from the following files:

- alymatkustaja.hh
- alymatkustaja.cpp

### 4.2 Ammus Class Reference

Inheritance diagram for Ammus:



#### Public Slots

- void **tuhoaAmmus** ()  
*tuhoaAmmus tuhoaa ammuksen hallitusti, käytetään singleshot timerin kanssa viivästettyyn tuhoamiseen. Hyödyllinen esimerkiksi animoidun räjähdysten kanssa.*

## Public Member Functions

- [Ammus](#) ()  
*Tyhmä rakentaja ammukselle, ei tule käyttää.*
- [Ammus](#) (QGraphicsItem \*parent, qreal suunta)  
*Ammuksen "perus" rakentaja; ei ole hyvä käyttää sinänsä, mutta toteutettu malliksi aliluokkien rakentajaksi.*
- [~Ammus](#) ()  
*Ammus kantaluokan purkaja, hoitaa kantaluokan new'llä varaamien muuttujien tuhoamisen.*
- virtual void [etene](#) ()  
*etene liikuttaa ammusta ja tarkistaa mahdolliset osumat. Kantaman ylittyessä tuhoaa ammuksen.*
- virtual void [ulkoOsuma](#) ()  
*ulkoOsuma tuhoaa ammuksen. Käytetään kun ammus halutaan tuhota ulkopuolelta.*

## Protected Member Functions

- void **paint** (QPainter \*painter, const QStyleOptionGraphicsItem \*option, QWidget \*widget)
- virtual QRectF **boundingRect** () const
- virtual void [alustaAmmus](#) (QGraphicsItem \*parent, qreal suunta)  
*alustaAmmus Alustaa ammuksen toimintakuntoon. Käynnistää ammuksen ajastimen. Kutsutaan yleensä ammuksen rakentajassa.*
- virtual void [asettaAmmuskuva](#) ()=0  
*asettaAmmuskuva asettaa ammuksen ulkoasun. Toteutettava periytyyessä luokassa, asetettava ammuskuva on QImage Ammus::ammuskuva.*
- virtual bool [tarkistaOsuma](#) ()  
*tarkistaOsuma tarkistaa osuuko ammus nysseen ja osuttaessa ilmoittaa sekä ammukselle että nysseille osumasta.*
- virtual void [osuma](#) ()  
*osuma määrittelee ammuksen (ylimääräisen) käyttäytymisen osuessa kohteeseen. Oletustoteutus ei tee mitään.*
- QPointF [suuntaPiste](#) () const  
*suuntaPiste selvittää ammuksen etenemissuunnan.*

## Protected Attributes

- QImage **ammuskuva**
- bool **animoituOsuma\_** {false}
- int **nopeus** {1500}
- int **kantama** {300}
- qreal **suunta\_** {0}
- qreal **etenema**
- QTimer \* **animaatioajastin**
- bool **tuhoutumassa** {false}

## 4.2.1 Constructor & Destructor Documentation

### 4.2.1.1 Ammus::Ammus ( QGraphicsItem \* parent, qreal suunta )

Ammuksen "perus" rakentaja; ei ole hyvä käyttää sinänsä, mutta toteutettu malliksi aliluokkien rakentajaksi.

#### Parameters

---

<i>parent</i>	ammuksen vanhempi
<i>suunta</i>	Ammuksen lähtösuunta

**Precondition**

-

**Postcondition**

[Ammus](#) on luotu, alustettu ja liikkuu

**4.2.2 Member Function Documentation****4.2.2.1 void Ammus::alustaAmmus ( QGraphicsItem \* *parent*, qreal *suunta* )** [protected],[virtual]

alustaAmmus Alustaa ammuksen toimintakuntoon. Käynnistää ammuksen ajastimen. Kutsutaan yleensä ammuksen rakentajassa.

**Parameters**

<i>parent</i>	Ammuksen ampuja, ammuksen lähtöpaikaksi asetetaan ampujan paikka
<i>suunta</i>	Ammuksen lähtösuunta

**Precondition**

[Ammus](#) on alustamaton

**Postcondition**

[Ammus](#) on alustettu ja toimintakuntoinen. Poikkeustakuu: perus

Reimplemented in [Miina](#).

**4.2.2.2 void Ammus::asetAmmuskuva ( )** [protected],[pure virtual]

asetAmmuskuva asettaa ammuksen ulkoasun. Toteutettava periytyyessä luokassa, asetettava ammuskuva on QImage Ammus::ammuskuva.

**Precondition**

-

**Postcondition**

Ammus::ammuskuva on asetettu. Poikkeustakuu: perus

**Exceptions**

<i>PeliVirhe</i>	Ammuskuvan avaaminen epäonnistui.
------------------	-----------------------------------

Implemented in [Miina](#), [Laser](#), and [Rynnakkokivaari](#).

**4.2.2.3 void Ammus::etene ( )** [virtual]

etene liikuttaa ammusta ja tarkistaa mahdolliset osumat. Kantaman ylittyessä tuhoaa ammuksen.

**Precondition**

[Ammus](#) on alustettu

**Postcondition**

[Ammus](#) on liikkunut ja tarkastanut osumat. Poikkeustakuu: perus

**4.2.2.4 void Ammus::osuma ( ) [protected],[virtual]**

osuma määrittelee ammuksen (ylimääräisen) käyttäytymisen osuessa kohteeseen. Oletustoteutus ei tee mitään.

**Precondition**

-

**Postcondition**

Poikkeustakuu: perus

Reimplemented in [Miina](#).

**4.2.2.5 QPointF Ammus::suuntaPiste ( ) const [protected]**

suuntaPiste selvittää ammuksen etenemissuunnan.

**Precondition**

[Ammus](#) on alustettu

**Returns**

etäisyydellä 1 ammuksesta oleva QPointF ammuksen etenemissuunnassa

**Postcondition**

Poikkeustakuu: perus

**4.2.2.6 bool Ammus::tarkistaOsuma ( ) [protected],[virtual]**

tarkistaOsuma tarkistaa osuuko ammus nysseen ja osuttaessa ilmoittaa sekä ammukselle että nysseille osumasta.

**Precondition**

[Ammus](#) on alustettu

**Returns**

true mikäli ammus osui, muuten false

**Postcondition**

Poikkeustakuu: perus

Reimplemented in [Miina](#), and [Laser](#).

## 4.2.2.7 void Ammus::tuhoaAmmus ( ) [slot]

tuhoaAmmus tuhoaa ammuksen hallitusti, käytetään singleshoot timerin kanssa viivästettyyn tuhoamiseen. Hyödyllinen esimerkiksi animoidun räjähdysen kanssa.

## Precondition

[Ammus](#) on alustettu

## Postcondition

[Ammus](#) on tuhottu oikein. Poikkeustakuu: vahva

## 4.2.2.8 void Ammus::ulkoOsuma ( ) [virtual]

ulkoOsuma tuhoaa ammuksen. Käytetään kun ammus halutaan tuhota ulkopuolelta.

## Precondition

[Ammus](#) on alustettu

## Postcondition

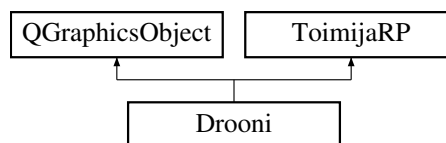
[Ammus](#) on tuhottu normaalisti tai ajoitettu tuhoutumaan normaalisti. Poikkeustakuu: perus.

The documentation for this class was generated from the following files:

- [ammus.hh](#)
- [ammus.cpp](#)

## 4.3 Drooni Class Reference

Inheritance diagram for Drooni:



## Public Slots

- void [alaLiikkuaEteen](#) ()  
*alaLiikkuaEteen* Laittaa droonin liikkumaan eteenpäin kunnes liike lopetetaan.
- void [alaLiikkuaTaakse](#) ()  
*alaLiikkuaTaakse* Laittaa droonin liikkumaan taaksepäin kunnes liike lopetetaan.
- void [alaKaantyaMyota](#) ()  
*alaKaantyaMyota* Laittaa droonin kääntymään myötäpäivään kunnes kääntyminen lopetetaan.
- void [alaKaantyaVasta](#) ()  
*alaKaantyaVasta* Laittaa droonin kääntymään vastapäivään kunnes kääntyminen lopetetaan.
- void [lopetaEteenLiikkuminen](#) ()  
*lopetaEteenLiikkuminen* Lopettaa droonin eteenpäin liikkumisen.



- void [lopetaTaakseLiikkuminen](#) ()  
*lopetaTaakseLiikkuminen* Lopettaa droonin taaksepäin liikkumisen.
- void [lopetaKaannosMyota](#) ()  
*lopetaKaannosMyota* Lopettaa droonin kääntymisen myötäpäivään.
- void [lopetaKaannosVasta](#) ()  
*lopetaKaannosVasta* Lopettaa droonin kääntymisen vastapäivään.
- void [ammu](#) ()  
*ammu* luo uuden valitun asean ammuksen ja lisää sen samaan *QGraphicsScene* droonin kanssa.
- void [valitseRynkky](#) ()  
*valitseRynkky* valitsee rynnäkkökiväärin aktiiviseksi aseeksi.
- void [valitseMiina](#) ()  
*valitseMiina* valitsee maamiinan aktiiviseksi aseeksi.
- void [valitseLaser](#) ()  
*valitseLaser* valitsee laserin aktiiviseksi aseeksi.

## Signals

- void [drooniLiikkunut](#) (*QGraphicsItem* \*drooni)  
*drooniLiikkunut* signaali lähetetään aina kun drooni on liikkunut.
- void [drooniAmpunut](#) (short int ase, short int ammuksia\_jaljella)  
*drooniAmpunut* signaali lähetetään aina kun drooni on ampunut.
- void [ammuksetLopussa](#) ()  
*ammuksetLopussa* signaali lähetetään kun droonin ammukset ovat lopussa.

## Public Member Functions

- [Drooni](#) (*QGraphicsItem* \*parent=NULL)  
*Droonin* rakentaja, luo uuden drooni olion.
- virtual [~Drooni](#) ()  
*~Drooni* tuhoaa droonin.
- void [keyPressEvent](#) (*QKeyEvent* \*event)  
*keyPressEvent* vastaanottaa olion saaman näppäimistösyötteen. Muuntaa nuolinäppäimet ja välilyönnin droonin liikkeiksi. Ei yleensä tarvitse kutsua itse.
- void [keyReleaseEvent](#) (*QKeyEvent* \*event)  
*keyReleaseEvent* ks. *keyPressEvent*.
- short int [ammuksiaJaljella](#) (int ase) const  
*ammuksiaJaljella* Kertoo paljonko kysytyssä aseessa on ammuksia jäljellä
- virtual Rajapinta::Sijainti [annaSijainti](#) () const
- virtual void [liiku](#) (Rajapinta::Sijainti sij)
- virtual bool [onkoTuhottu](#) () const
- virtual void [tuhoa](#) ()
- void [liikuSulavasti](#) (Rajapinta::Sijainti sij)  
*liikuSulavasti* Laittaa droonin liikkumaan sulavasti kohti annettua sijaintia.

## Public Attributes

- *QTimer* \* [animaatioajastin](#)
- bool [liikkuuEteen](#)
- bool [liikkuuTaakse](#)
- bool [kaantyyVasemmalle](#)
- bool [kaantyyOikealle](#)

## Protected Member Functions

- void **paint** (QPainter \*painter, const QStyleOptionGraphicsItem \*option, QWidget \*widget)
- virtual QRectF **boundingRect** () const
- virtual QVariant **itemChange** (GraphicsItemChange change, const QVariant &value)

### 4.3.1 Constructor & Destructor Documentation

#### 4.3.1.1 Drooni::Drooni ( QGraphicsItem \* *parent* = NULL )

Droonin rakentaja, luo uuden drooni olion.

##### Parameters

<i>parent</i>	droonin vanhempi, mahdollista jättää määrittelemättä.
---------------	---

##### Precondition

-

##### Postcondition

[Drooni](#) on luotu ja valmis käytettäväksi. Poikkeustakuu: ??

#### 4.3.1.2 Drooni::~Drooni ( ) [virtual]

~Drooni tuhoaa droonin.

##### Postcondition

drooni on tuhottu.

### 4.3.2 Member Function Documentation

#### 4.3.2.1 void Drooni::alaKaantyaMyota ( ) [slot]

alaKaantyaMyota Laittaa droonin kääntymään myötöpäivään kunnes kääntyminen lopetetaan.

##### Precondition

-

##### Postcondition

kaantyyOikealle = true

#### 4.3.2.2 void Drooni::alaKaantyaVasta ( ) [slot]

alaKaantyaVasta Laittaa droonin kääntymään vastapäivään kunnes kääntyminen lopetetaan.

##### Precondition

-

##### Postcondition

kaantyyVasemmalle = true;

#### 4.3.2.3 void Drooni::alaLiikkuaEteen ( ) [slot]

alaLiikkuaEteen Laittaa droonin liikkumaan eteenpäin kunnes liike lopetetaan.

##### Precondition

-

##### Postcondition

liikkuuEteen = true

#### 4.3.2.4 void Drooni::alaLiikkuaTaakse ( ) [slot]

alaLiikkuaTaakse Laittaa droonin liikkumaan taaksepäin kunnes liike lopetetaan.

##### Precondition

-

##### Postcondition

liikkuuTaakse = true

#### 4.3.2.5 void Drooni::ammu ( ) [slot]

ammu luo uuden valitun asean ammuksen ja lisää sen samaan QGraphicsSceneen droonin kanssa.

##### Precondition

[Drooni](#) kuuluu QGraphicsSceneen ja ase on valituna.

##### Postcondition

Uusi ammus on luotu ja lisätty QGraphicsSceneen. Poikkeustakuu: vahva

#### 4.3.2.6 short Drooni::ammuksiaJaljella ( int ase ) const

ammuksiaJaljella Kertoo paljonko kysytyssä aseessa on ammuksia jäljellä

##### Parameters

<i>ase</i>	Aseen, jonka ammustiedot halutaan, numero. 1 = RK, 2 = Maamiina, 3 = <a href="#">Laser</a>
------------	--

##### Returns

Jäljellä olevien ammusten lukumäärä

##### Precondition

ase on välillä [0,2]

##### Postcondition

Poikkeustakuu: vahva

## Exceptions

<i>PeliVirhe</i>	mikäli aseeseen tietoja ei löydy
------------------	----------------------------------

**4.3.2.7 void Drooni::drooniAmpunut ( short int *ase*, short int *ammuksia\_jaljella* ) [signal]**

drooniAmpunut signaali lähetetään aina kun drooni on ampunut.

## Parameters

<i>ase</i>	Ampumiseen käytetty ase.
<i>ammuksia_jaljella</i>	Aseen jäljellä oleva ammusmäärä.

**4.3.2.8 void Drooni::drooniLiikkunut ( QGraphicsItem \* *drooni* ) [signal]**

drooniLiikkunut signaali lähetetään aina kun drooni on liikkunut.

## Parameters

<i>ottaa</i>	droonin parametrinsa.
--------------	-----------------------

**4.3.2.9 void Drooni::keyPressEvent ( QKeyEvent \* *event* )**

keyPressEvent vastaanottaa olion saaman näppäimistösyötteen. Muuntaa nuolinäppäimet ja välilyönnin droonin liikkeiksi. Ei yleensä tarvitse kutsua itse.

## Parameters

<i>event</i>	QKeyEvent* tyyppinen tapahtuma.
--------------	---------------------------------

## Precondition

-

## Postcondition

[Drooni](#) on reagoinnut painikkeeseen halutulla tavalla. Poikkeustakuu: ??

**4.3.2.10 void Drooni::keyReleaseEvent ( QKeyEvent \* *event* )**

keyReleaseEvent ks. keyPressEvent.

## Parameters

<i>event</i>	ks. keyPressEvent.
--------------	--------------------

**4.3.2.11 void Drooni::liikuSulavasti ( Rajapinta::Sijainti *sij* )**

liikuSulavasti Laittaa droonin liikkumaan sulavasti kohti annettua sijaintia.

## Parameters

<i>sij</i>	Määränpää sijainti
------------	--------------------

## Precondition

-

## Postcondition

[Drooni](#) liikkuu kohti määränpäää. Poikkeustakuu: perus

**4.3.2.12 void Drooni::lopetaEteenLiikkuminen ( ) [slot]**

lopetaEteenLiikkuminen Lopettaa droonin eteenpäin liikkumisen.

## Precondition

-

## Postcondition

liikkuuEteen = false

**4.3.2.13 void Drooni::lopetaKaannosMyota ( ) [slot]**

lopetaKaannosMyota Lopettaa droonin kääntymisen myötäpäivään.

## Precondition

-

## Postcondition

kaantyyOikealle = false

**4.3.2.14 void Drooni::lopetaKaannosVasta ( ) [slot]**

lopetaKaannosVasta Lopettaa droonin kääntymisen vastapäivään.

## Precondition

-

## Postcondition

kaantyyVasemmalle = false

**4.3.2.15 void Drooni::lopetaTaakseLiikkuminen ( ) [slot]**

lopetaTaakseLiikkuminen Lopettaa droonin taaksepäin liikkumisen.

## Precondition

-

## Postcondition

liikkuuTaakse = false

#### 4.3.2.16 void Drooni::valitseLaser ( ) [slot]

valitseLaser valitsee laserin aktiiviseksi aseeksi.

##### Precondition

-

##### Postcondition

[Laser](#) on valittuna. Poikkeustakuu: vahva

#### 4.3.2.17 void Drooni::valitseMiina ( ) [slot]

valitseMiina valitsee maamiinan aktiiviseksi aseeksi.

##### Precondition

-

##### Postcondition

Maamiina on valittuna. Poikkeustakuu: vahva

#### 4.3.2.18 void Drooni::valitseRynkky ( ) [slot]

valitseRynkky valitsee rynnäkkökiväärin aktiiviseksi aseeksi.

##### Precondition

-

##### Postcondition

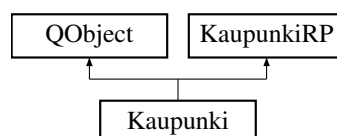
Rynnäkkökivääri on valittuna. Poikkeustakuu: vahva

The documentation for this class was generated from the following files:

- [drooni.hh](#)
- drooni.cpp

## 4.4 Kaupunki Class Reference

Inheritance diagram for Kaupunki:



### Public Slots

- void [ajastaLopetus](#) ()  
*aloitaLopetus Käynnistää ajastimen joka X ajan kuluttua päättää pelin. Kutsutaan kun droonin ammuksen loppuvat.*

## Public Member Functions

- virtual void **asetTausta** (QImage &perustaustakuva, QImage &isotaustakuva)
- virtual void **asetKello** (QTime kello)
- virtual void **lisaaPysakki** (std::shared\_ptr< Rajapinta::PysakkiRP > pysakki)
- virtual void **pelialkaa** ()
- virtual void **lisaaToimija** (std::shared\_ptr< Rajapinta::ToimijaRP > uusitoimija)
- virtual void **poistaToimija** (std::shared\_ptr< Rajapinta::ToimijaRP > toimija)
- virtual void **toimijaTuhottu** (std::shared\_ptr< Rajapinta::ToimijaRP > toimija)
- virtual bool **loytyykoToimija** (std::shared\_ptr< Rajapinta::ToimijaRP > toimija) const
- virtual void **toimijaLiikkunut** (std::shared\_ptr< Rajapinta::ToimijaRP > toimija)
- virtual std::vector  
     < std::shared\_ptr  
     < Rajapinta::ToimijaRP > > **annaToimijatLahella** (Rajapinta::Sijainti paikka) const
- virtual bool **peliloppunut** () const

### 4.4.1 Member Function Documentation

#### 4.4.1.1 void Kaupunki::ajastaLopetus ( ) [slot]

aloitaLopetus Käynnistää ajastimen joka X ajan kuluttua päättää pelin. Kutsutaan kun droonin ammukset loppuvat.

##### Precondition

Peli on käynnissä

##### Postcondition

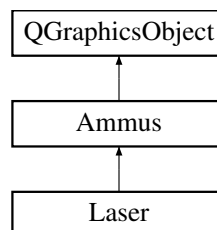
Peli on ajastettu loppumaan

The documentation for this class was generated from the following files:

- [kaupunki.hh](#)
- kaupunki.cpp

## 4.5 Laser Class Reference

Inheritance diagram for Laser:



## Public Member Functions

- [Laser](#) (QGraphicsItem \*parent, qreal suunta)  
*Laserin rakentaja.*
- [~Laser](#) ()  
*Laserin purkaja.*

## Protected Member Functions

- void **paint** (QPainter \*painter, const QStyleOptionGraphicsItem \*option, QWidget \*widget)
- virtual QRectF **boundingRect** () const override
- virtual void [asetAmmuskuva](#) ()  
*asetAmmuskuva* asettaa ammuksen ulkoasun. Toteutettava periytetyssä luokassa, asetettava ammuskuva on QImage Ammus::ammuskuva.
- virtual bool [tarkistaOsuma](#) ()  
*tarkistaOsuma* tarkistaa osuuko ammus nysseen ja osuttaessa ilmoittaa sekä ammukselle että nysseille osumasta.

## Additional Inherited Members

### 4.5.1 Constructor & Destructor Documentation

#### 4.5.1.1 Laser::Laser ( QGraphicsItem \* parent, qreal suunta )

Laserin rakentaja.

Parameters

<i>parent</i>	laserin ampunut toimija, yleensä drooni
<i>suunta</i>	suunta minne laser lähtee

Postcondition

[Laser](#) luotu onnistuneesti ja ajoitettu tuhoutumaan 200ms kuluttua. Poikkeustakuu: perus

#### 4.5.1.2 Laser::~~Laser ( )

Laserin purkaja.

Postcondition

[Laser](#) on tuhottu oikein

### 4.5.2 Member Function Documentation

#### 4.5.2.1 void Laser::asetAmmuskuva ( ) [protected],[virtual]

asetAmmuskuva asettaa ammuksen ulkoasun. Toteutettava periytetyssä luokassa, asetettava ammuskuva on QImage Ammus::ammuskuva.

Precondition

-

Postcondition

Ammus::ammuskuva on asetettu. Poikkeustakuu: perus



### Exceptions

<i>PeliVirhe</i>	Ammuskuvan avaaminen epäonnistui.
------------------	-----------------------------------

Implements [Ammus](#).

#### 4.5.2.2 `bool Laser::tarkistaOsuma ( )` [protected],[virtual]

tarkistaOsuma tarkistaa osuuko ammus nysseen ja osuttaessa ilmoittaa sekä ammukselle että nyssele osumasta.

### Precondition

[Ammus](#) on alustettu

### Returns

true mikäli ammus osui, muuten false

### Postcondition

Poikkeustakuu: perus

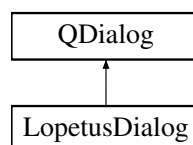
Reimplemented from [Ammus](#).

The documentation for this class was generated from the following files:

- [laser.hh](#)
- laser.cpp

## 4.6 LopetusDialog Class Reference

Inheritance diagram for LopetusDialog:



### Public Member Functions

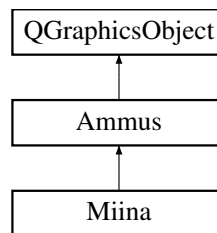
- **LopetusDialog** (QWidget \*parent=0)
- void **annaTilasto** (std::shared\_ptr< [Tilasto](#) > tilastoptr)
- void **tallennaJaLopeta** ()

The documentation for this class was generated from the following files:

- lopetusdialog.hh
- lopetusdialog.cpp

## 4.7 Miina Class Reference

Inheritance diagram for Miina:



### Public Member Functions

- [Miina](#) (QGraphicsItem \*parent)  
*Miinan rakentaja.*
- [~Miina](#) ()  
*Miinan purkaja.*
- int **getRajahdysVaihe** () const
- void **setRajahdysVaihe** (int rajahdysVaihe)

### Protected Member Functions

- virtual void [asetAmmuskuva](#) ()  
*asetAmmuskuva asettaa ammuksen ulkoasun. Toteutettava periytetyssä luokassa, asetettava ammuskuva on QImage Ammus::ammuskuva.*
- virtual void [alustaAmmus](#) (QGraphicsItem \*parent, qreal suunta)  
*alustaAmmus Alustaa ammuksen toimintakuntoon. Käynnistää ammuksen ajastimen. Kutsutaan yleensä ammuksen rakentajassa.*
- virtual void [osuma](#) ()  
*osuma määrittelee ammuksen (ylimääräisen) käyttäytymisen osuessa kohteeseen. Oletustoteutus ei tee mitään.*
- virtual bool [tarkistaOsuma](#) ()  
*tarkistaOsuma tarkistaa osuuko ammus nysseen ja osuttaessa ilmoittaa sekä ammukselle että nyssele osumasta.*

### Properties

- int **rVaihe**

### Additional Inherited Members

#### 4.7.1 Constructor & Destructor Documentation

##### 4.7.1.1 Miina::Miina ( QGraphicsItem \* parent )

Miinan rakentaja.

Parameters

---

<i>parent</i>	miinan jättäjä, yleensä drooni
---------------	--------------------------------

**Postcondition**

[Miina](#) asetettu ja alustettu. Poikkeustakuu: perus

**4.7.1.2 Miina::~Miina ( )**

Miinan purkaja.

**Postcondition**

miina on tuhottu oikein

**4.7.2 Member Function Documentation****4.7.2.1 void Miina::alustaAmmus ( QGraphicsItem \* *parent*, qreal *suunta* ) [protected],[virtual]**

alustaAmmus Alustaa ammuksen toimintakuntoon. Käynnistää ammuksen ajastimen. Kutsutaan yleensä ammuksen rakentajassa.

**Parameters**

<i>parent</i>	Ammuksen ampuja, ammuksen lähtöpaikaksi asetetaan ampujan paikka
<i>suunta</i>	Ammuksen lähtösuunta

**Precondition**

[Ammus](#) on alustamaton

**Postcondition**

[Ammus](#) on alustettu ja toimintakuntainen. Poikkeustakuu: perus

Reimplemented from [Ammus](#).

**4.7.2.2 void Miina::asetAmmuskuva ( ) [protected],[virtual]**

asetAmmuskuva asettaa ammuksen ulkoasun. Toteutettava periytyyssä luokassa, asetettava ammuskuva on QImage Ammus::ammuskuva.

**Precondition**

-

**Postcondition**

Ammus::ammuskuva on asetettu. Poikkeustakuu: perus

**Exceptions**

<i>PeliVirhe</i>	Ammuskuvan avaaminen epäonnistui.
------------------	-----------------------------------

Implements [Ammus](#).

**4.7.2.3** `void Miina::osuma ( ) [protected],[virtual]`

osuma määrittelee ammuksen (ylimääräisen) käyttäytymisen osuessa kohteeseen. Oletustoteutus ei tee mitään.

**Precondition**

-

**Postcondition**

Poikkeustakuu: perus

Reimplemented from [Ammus](#).

**4.7.2.4** `bool Miina::tarkistaOsuma ( ) [protected],[virtual]`

tarkistaOsuma tarkistaa osuuko ammus nysseen ja osuttaessa ilmoittaa sekä ammukselle että nyssele osumasta.

**Precondition**

[Ammus](#) on alustettu

**Returns**

true mikäli ammus osui, muuten false

**Postcondition**

Poikkeustakuu: perus

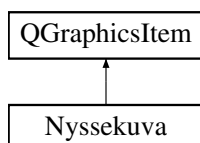
Reimplemented from [Ammus](#).

The documentation for this class was generated from the following files:

- [miina.hh](#)
- [miina.cpp](#)

## 4.8 Nyssekuva Class Reference

Inheritance diagram for Nyssekuva:



## Public Member Functions

- [Nyssekuva](#) (std::shared\_ptr< Rajapinta::ToimijaRP > nysse)  
*Nyssekuva* Luokan rakentaja.
- [~Nyssekuva](#) ()  
*Luokan purkaja.*
- void [osuma](#) ()  
*osuma Kutsutaan kun nysse on ottanut osuman ammuksesta. Ilmoittaa pelilogiikan nyssele tuhoutumisesta (Tuhoaa toimijan ja kertoo kaupungille)*
- void [kerroKaupunki](#) (Rajapinta::KaupunkiRP \*kaupunki)  
*kerroKaupunki kertoo Nyssekuvalle missä kaupungissa se on.*
- void [tuhoutu](#) ()  
*tuhoutu Vaihtaa nyssekuvan ulkoasu tuhoutuneeseen nysseen.*
- bool [tuhottu](#) ()  
*tuhottu Kertoo onko Nyssekuvan ulkoasu tuhoutunut nysse.*

## Protected Member Functions

- void **paint** (QPainter \*painter, const QStyleOptionGraphicsItem \*option, QWidget \*widget)
- virtual QRectF **boundingRect** () const

### 4.8.1 Constructor & Destructor Documentation

#### 4.8.1.1 Nyssekuva::Nyssekuva ( std::shared\_ptr< Rajapinta::ToimijaRP > nysse )

[Nyssekuva](#) Luokan rakentaja.

Parameters

<i>nysse</i>	osoitin kuvastettavaan nysseen
--------------	--------------------------------

#### 4.8.1.2 Nyssekuva::~~Nyssekuva ( )

*Luokan purkaja.*

Postcondition

[Nyssekuva](#) on tuhottu oikein

### 4.8.2 Member Function Documentation

#### 4.8.2.1 void Nyssekuva::kerroKaupunki ( Rajapinta::KaupunkiRP \* kaupunki )

*kerroKaupunki kertoo Nyssekuvalle missä kaupungissa se on.*

Parameters

<i>kaupunki</i>	KaupunkiRP:n toteuttava kaupunkiolio
-----------------	--------------------------------------

Postcondition

*kaupunki\_ asetettu. Poikkeustakuu: nothrow*

#### 4.8.2.2 void Nyssekuva::osuma ( )

osuma Kutsutaan kun nysse on ottanut osuman ammuksesta. Ilmoittaa pelilogiikan nyssele tuhoutumisesta (Tuhoaa toimijan ja kertoo kaupungille)

##### Precondition

[Nyssekuva](#) on ottanut osuman

##### Postcondition

Nysse on tuhattu. Poikkeustakuu: perus

#### 4.8.2.3 bool Nyssekuva::tuhottu ( )

tuhottu Kertoo onko Nyssekuvan ulkoasu tuhoutunut nysse.

##### Returns

Boolean arvo

##### Postcondition

Poikkeustakuu: nothrow

#### 4.8.2.4 void Nyssekuva::tuhoutu ( )

tuhoutu Vaihtaa nyssekuvan ulkoasun tuhoutuneeseen nysseen.

##### Precondition

-

##### Postcondition

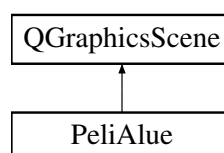
Ulkoasu on tuhoutunut nysse. Poikkeustakuu: nothrow

The documentation for this class was generated from the following files:

- [nyssekuva.hh](#)
- [nyssekuva.cpp](#)

## 4.9 PeliAlue Class Reference

Inheritance diagram for PeliAlue:



## Public Member Functions

- void **lisaaDrooni** ([Drooni](#) \*drooni)  
*lisaaDrooni lisää droonin pelialueelle.*

## Protected Member Functions

- void **mousePressEvent** (QGraphicsSceneMouseEvent \*mouseEvent)

### 4.9.1 Member Function Documentation

#### 4.9.1.1 void PeliAlue::lisaaDrooni ( [Drooni](#) \* *drooni* )

lisaaDrooni lisää droonin pelialueelle.

##### Parameters

<i>drooni</i>	Osoitin lisättävään drooniin.
---------------	-------------------------------

##### Precondition

Pelialueelle ei ole vielä lisätty droonia.

##### Postcondition

[Drooni](#) on lisätty pelialueelle. Poikkeustakuu: perus

##### Exceptions

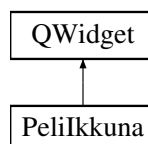
<i>PeliVirhe</i>	mikäli pelialueella on jo drooni tai droonin lisäys epäonnistui.
------------------	--

The documentation for this class was generated from the following files:

- [pelialue.hh](#)
- pelialue.cpp

## 4.10 Peliikkuna Class Reference

Inheritance diagram for Peliikkuna:



## Public Slots

- void **liikutaViewport** (QGraphicsItem \*kohde)
- void **paivitaViewport** ()
- void **paivitaAmmukset** (int ase, int ammuksia)

## Public Member Functions

- [Peliikkuna](#) (QWidget \*parent=0)  
*Peliikkuna luo uuden peli-ikkunan.*
- [~Peliikkuna](#) ()  
*Peliikkunan purkaja.*
- void [asetaNKartta](#) (QImage &kartta)  
*asetaNKartta Asettaa pelialueen pohjakartan.*
- QGraphicsPixmapItem \* [lisaa\\_pysakki](#) (std::shared\_ptr< Rajapinta::PysakkiRP > pysakkiptr)  
*lisaa\_pysakki Lisää pysäkin kartalle*
- [Nyssekuva](#) \* [lisaa\\_nysse](#) (std::shared\_ptr< Rajapinta::ToimijaRP > toimijaptr)  
*lisaa\_nysse Lisää nysse kartalle*
- void [lisaaDrooni](#) ([Drooni](#) \*drooni)  
*lisaaDrooni Lisää droonin kartalle.*
- void [PaivitaTulokset](#) (std::vector< int > tulokset)  
*PaivitaTulokset Päivittää peliruudulla näytettävät tilastotiedot.*
- void [naytaToplista](#) (std::vector< QString > huipputulokset)  
*naytaToplista Näyttää peliruudulla tämänhetkisen top10 tilanteen.*
- void [asetaNStatus](#) (QString status)  
*asetaNStatus Asettaa peliruudun statustekstin.*

### 4.10.1 Constructor & Destructor Documentation

#### 4.10.1.1 Peliikkuna::Peliikkuna ( QWidget \* parent = 0 ) [explicit]

[Peliikkuna](#) luo uuden peli-ikkunan.

##### Parameters

<i>parent</i>	peli-ikkunan vanhempi. Tässä ohjelmassa aina määrittämätön.
---------------	---

##### Postcondition

Ohjelmassa ei ole vielä peli-ikkuna oliota.

#### 4.10.1.2 Peliikkuna::~~Peliikkuna ( )

Peliikkunan purkaja.

##### Postcondition

[Peliikkuna](#) on poistettu hallitusti.

### 4.10.2 Member Function Documentation

#### 4.10.2.1 void Peliikkuna::asetaNKartta ( QImage & kartta )

asetaNKartta Asettaa pelialueen pohjakartan.



## Parameters

<i>kartta</i>	Tampereen keskustan kartta
---------------	----------------------------

## Precondition

Karttaa ei ole vielä asetettu. Kartta on pelilogiikan tarjoama isokartta.

## Postcondition

Pohjakartta on asetettu. Poikkeustakuu: perus

4.10.2.2 void Peliikkuna::asetatus ( QString *status* )

asetatus Asettaa peliruudun statustekstin.

## Parameters

<i>status</i>	Haluttu statusteksti
---------------	----------------------

## Precondition

-

## Postcondition

Statusteksti on asetettu. Poikkeustakuu: perus

4.10.2.3 Nyssekuva \* Peliikkuna::lisaa\_nysse ( std::shared\_ptr< Rajapinta::ToimijaRP > *toimijaptr* )

lisaa\_nysse Lisää nysse kartalle

## Parameters

<i>toimijaptr</i>	Osoitin nysseolioon joka toteuttaa ToimijaRP:n.
-------------------	---

## Returns

Osoitin [Nyssekuva](#) olioon.

## Precondition

toimijaptr on nysse ja löytyy kaupungista.

## Postcondition

Nysse on lisätty onnistuneesti. Poikkeustakuu: perus

4.10.2.4 QGraphicsPixmapItem \* Peliikkuna::lisaa\_pysakki ( std::shared\_ptr< Rajapinta::PysakkiRP > *pysakkiptr* )

lisaa\_pysakki Lisää pysäkin kartalle

## Parameters

<i>pysakkipt</i>	Osoitin PysakkiRP:n toteuttavaan pysäkkiolioon
------------------	--

## Returns

Palauttaa osoittimen pysäkkikuva oloon.

## Precondition

Pysäkkiä ei ole vielä piirretty

## Postcondition

Pysäkki on lisätty onnistuneesti. Poikkeustakuu: perus

## 4.10.2.5 void Pelilikkuna::lisaaDrooni ( Drooni \* drooni )

lisaaDrooni Lisää droonin kartalle.

## Parameters

<i>drooni</i>	Pelissä käytettävä drooniolio
---------------	-------------------------------

## Precondition

Droonia ei ole vielä lisätty

## Postcondition

[Drooni](#) on lisätty peliin. Poikkeustakuu: perus

## 4.10.2.6 void Pelilikkuna::naytaToplista ( std::vector&lt; QString &gt; huipputulokset )

naytaToplista Näyttää peliruudulla tämänhetkisen top10 tilanteen.

## Parameters

<i>huipputulokset</i>	Vektori top10 tilastosta.
-----------------------	---------------------------

## Precondition

Top10 listaa ei ole vielä esitetty

## Postcondition

Top10 lista näkyy peliruudulla. Poikkeustakuu: perus

## 4.10.2.7 void Pelilikkuna::PaivitaTulokset ( std::vector&lt; int &gt; tulokset )

PaivitaTulokset Päivittää peliruudulla näytettävät tilastotiedot.

## Parameters

<i>tulokset</i>	Vektori jossa päivitettävät tilastotiedot.
-----------------	--

## Precondition

-

## Postcondition

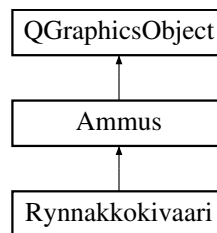
Tulokset on päivitetty. Poikkeustakuu: perus

The documentation for this class was generated from the following files:

- [peliikkuna.hh](#)
- [peliikkuna.cpp](#)

## 4.11 Rynnakkokivaari Class Reference

Inheritance diagram for Rynnakkokivaari:



### Public Member Functions

- [Rynnakkokivaari](#) (QGraphicsItem \*parent, qreal suunta)  
*Rynnakkokivaarin rakentaja.*
- [~Rynnakkokivaari](#) ()  
*Ammuksen purkaja, tuhoaa new'llä luodun QTimerin.*

### Protected Member Functions

- virtual void [asettaAmmuskuva](#) ()  
*asettaAmmuskuva asettaa ammuksen ulkoasun. Toteutettava periytetyssä luokassa, asetettava ammuskuva on QImage Ammus::ammuskuva.*

### Additional Inherited Members

#### 4.11.1 Constructor & Destructor Documentation

##### 4.11.1.1 Rynnakkokivaari::Rynnakkokivaari ( QGraphicsItem \* parent, qreal suunta )

Rynnakkokivaarin rakentaja.

## Parameters

<i>parent</i>	ammuksen ampuja
<i>suunta</i>	ammuksen lähtösuunta

## Postcondition

Rynnäkökiväärin ammus on alustettu ja toiminnassa. Poikkeustakuu: perus

## 4.11.1.2 Rynnakkokivaari::~~Rynnakkokivaari ( )

Ammuksen purkaja, tuhoaa new'llä luodun QTimerin.

## Postcondition

ammus on tuhottu oikein.

## 4.11.2 Member Function Documentation

## 4.11.2.1 void Rynnakkokivaari::asetAmmuskuva ( ) [protected],[virtual]

asetAmmuskuva asettaa ammuksen ulkoasun. Toteutettava periytetyssä luokassa, asetettava ammuskuva on QImage Ammus::ammuskuva.

## Precondition

-

## Postcondition

Ammus::ammuskuva on asetettu. Poikkeustakuu: perus

## Exceptions

<i>PeliVirhe</i>	Ammuskuvan avaaminen epäonnistui.
------------------	-----------------------------------

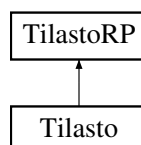
Implements [Ammus](#).

The documentation for this class was generated from the following files:

- [rynnakkokivaari.hh](#)
- [rynnakkokivaari.cpp](#)

## 4.12 Tilasto Class Reference

Inheritance diagram for Tilasto:



## Public Member Functions

- virtual int **annaPisteet** () const
- virtual void **matkustajiaKuoli** (int lkm)
- virtual void **lisaaMatkustajia** (int lkm)
- virtual void **nysseTuhoutui** ()
- virtual void **uusiNysse** ()
- virtual void **nyssePoistui** ()
- std::vector< int > **paivita\_tilasto** () const  
*paivita\_tilasto palauttaa kaupungille pistetaulua varten tarvittavat luvut.*
- std::vector< QString > **lue\_tiedosto** (QString)  
*lue\_tiedosto lukee tekstitiedoston, johon on tallennettu parhaiden pelaajien tiedot*
- void **kirjoitaTiedostoon** (int, QString) const  
*kirjoitaTiedostoon lisää pelaajan tuloksen top-listalle, jos on aihetta*
- int **selvitaListasijoitus** (int uusiTulos) const  
*selvitaListasijoitus Kertoo sijoituksen top 10 listalla annetuilla pisteillä.*

### 4.12.1 Member Function Documentation

#### 4.12.1.1 void Tilasto::kirjoitaTiedostoon ( int tulos, QString nimi ) const

kirjoitaTiedostoon lisää pelaajan tuloksen top-listalle, jos on aihetta

##### Parameters

<i>pelaajan</i>	tulos ja nimi
-----------------	---------------

##### Precondition

tiedosto johon kirjoitetaan on olemassa

##### Postcondition

poikkeustakuu: vahva

#### 4.12.1.2 std::vector< QString > Tilasto::lue\_tiedosto ( QString tiedosto )

lue\_tiedosto lukee tekstitiedoston, johon on tallennettu parhaiden pelaajien tiedot

##### Returns

vektorin, jossa alkioina on tiedoston rivi tallennettuna QStringiin

##### Precondition

tiedoston rivit on muotoa " nimi tulos "

##### Postcondition

poikkeustakuu: vahva

## Parameters

<i>tiedoston</i>	hakupolku
------------------	-----------

4.12.1.3 `std::vector< int > Tilasto::paivita_tilasto ( ) const`

paivita\_tilasto palauttaa kaupungille pistetaulua varten tarvittavat luvut.

## Returns

vektori, jossa on 5 tilaston private muuttujaa

## Precondition

olio rakennettu

## Postcondition

poikkeustakuu: vahva

4.12.1.4 `int Tilasto::selvitaListasijoitus ( int uusiTulos ) const`

selvitaListasijoitus Kertoo sijoituksen top 10 listalla annetuilla pisteillä.

## Parameters

<i>uusiTulos</i>	Pisteet joidenka sijoitus halutaan.
------------------	-------------------------------------

## Returns

Sijoitus top10 listalla, 11 jos ei top10 listalla

## Precondition

-

## Postcondition

Poikkeustakuu: vahva

The documentation for this class was generated from the following files:

- tilasto.hh
- tilasto.cpp



## Chapter 5

# File Documentation

### 5.1 ammus.hh File Reference

Kantaluokka kaikille pelissä käytettäville ammuksille. Tarjoaa perustoiminnoille toteutukset. Muistin loppuessa heittää `std::bad_alloc` ellei poikkeustakuu ole `nothrow`.

```
#include <QGraphicsObject>
#include <QTimer>
#include <QPointF>
```

#### Classes

- class [Ammus](#)

#### 5.1.1 Detailed Description

Kantaluokka kaikille pelissä käytettäville ammuksille. Tarjoaa perustoiminnoille toteutukset. Muistin loppuessa heittää `std::bad_alloc` ellei poikkeustakuu ole `nothrow`.

### 5.2 drooni.hh File Reference

[Drooni](#) luokka toteuttaa pelissä käytettävän droonin. Periytetty `QGraphicsObject` ja `ToimijaRP` luokista.

```
#include "toimijarp.hh"
#include "sijainti.hh"
#include "rynnakkokivaari.hh"
#include <QGraphicsObject>
#include <QImage>
#include <QPointF>
#include <QTimer>
```

#### Classes

- class [Drooni](#)



### 5.2.1 Detailed Description

[Drooni](#) luokka toteuttaa pelissä käytettävän droonin. Periytetty QGraphicsObject ja ToimijaRP luokista.

## 5.3 kaupunki.hh File Reference

[Kaupunki](#) luokka on KaupunkiRP:stä periytetty. [Kaupunki](#) hallinnoi pelin tapahtumia.

```
#include "kaupunkirp.hh"
#include "pysakkirp.hh"
#include "toimijarp.hh"
#include "peliikkuna.hh"
#include "drooni.hh"
#include "tilasto.hh"
#include "nyssekuva.hh"
#include <memory>
#include <vector>
#include <unordered_map>
#include <QImage>
#include <QTime>
#include <QGraphicsObject>
#include <QObject>
```

### Classes

- class [Kaupunki](#)

### 5.3.1 Detailed Description

[Kaupunki](#) luokka on KaupunkiRP:stä periytetty. [Kaupunki](#) hallinnoi pelin tapahtumia.

## 5.4 laser.hh File Reference

[Ammus](#) luokasta periytetty luokka, määrittelee laserin toiminnan.

```
#include "ammus.hh"
#include <QGraphicsObject>
#include <QRectF>
```

### Classes

- class [Laser](#)

### 5.4.1 Detailed Description

[Ammus](#) luokasta periytetty luokka, määrittelee laserin toiminnan.

## 5.5 miina.hh File Reference

**Ammus** luokasta periytetty luokka, määrittelee miinan toiminnan. Räjähdyksessä käytetyt kuvat <http://wrathgames.com/blog> WrathGames Studio. Muistin loppuessa voidaan heittää aina std::bad\_alloc.

```
#include "ammus.hh"
#include <QGraphicsObject>
#include <QPointF>
#include <unordered_map>
#include <QImage>
#include <QPropertyAnimation>
```

### Classes

- class [Miina](#)

#### 5.5.1 Detailed Description

**Ammus** luokasta periytetty luokka, määrittelee miinan toiminnan. Räjähdyksessä käytetyt kuvat <http://wrathgames.com/blog> WrathGames Studio. Muistin loppuessa voidaan heittää aina std::bad\_alloc.

## 5.6 nyssekuva.hh File Reference

**Nyssekuva** tarjoaa graafisen ilmeen pelin nysseille. Periytetty QGraphicsItemista.

```
#include "kaupunkirp.hh"
#include <toimijarp.hh>
#include <QGraphicsPixmapItem>
#include <QPixmap>
#include <QGraphicsItem>
#include <memory>
```

### Classes

- class [Nyssekuva](#)

#### 5.6.1 Detailed Description

**Nyssekuva** tarjoaa graafisen ilmeen pelin nysseille. Periytetty QGraphicsItemista.

## 5.7 pelialue.hh File Reference

**PeliAlue** on periytetty QGraphicsScenestä pelin tarkoituksiin sopivaksi.

```
#include "drooni.hh"
#include <QGraphicsScene>
#include <QGraphicsSceneMouseEvent>
```

## Classes

- class [PeliAlue](#)

### 5.7.1 Detailed Description

[PeliAlue](#) on periytetty QGraphicsScenestä pelin tarkoituksiin sopivaksi.

## 5.8 peliikkuna.hh File Reference

[Peliikkuna](#) on pelin "pääikkunan" tarjoava luokka.

```
#include "pysakkirp.hh"
#include "nyssekuva.hh"
#include "pelialue.hh"
#include "drooni.hh"
#include <QWidget>
#include <QImage>
#include <QGraphicsScene>
#include <QGraphicsItem>
#include <QGraphicsPixmapItem>
#include <QPointF>
#include <QString>
#include <QTimer>
#include <vector>
```

## Classes

- class [Peliikkuna](#)

### 5.8.1 Detailed Description

[Peliikkuna](#) on pelin "pääikkunan" tarjoava luokka.

## 5.9 rynnakkokivaari.hh File Reference

[Ammus](#) luokasta periytetty luokka, määrittelee rynnäkkökiväärin ammuksen.

```
#include "ammus.hh"
#include <QGraphicsObject>
#include <QPointF>
```

## Classes

- class [Rynnakkokivaari](#)

### 5.9.1 Detailed Description

[Ammus](#) luokasta periytetty luokka, määrittelee rynnäkkökiväärin ammuksen.

# Index

~Drooni  
Drooni, [13](#)  
~Laser  
Laser, [19](#)  
~Miina  
Miina, [22](#)  
~Nyssekuva  
Nyssekuva, [24](#)  
~Pelilikkuna  
Pelilikkuna, [27](#)  
~Rynnakkokivaari  
Rynnakkokivaari, [31](#)  
  
ajastaLopetus  
Kaupunki, [18](#)  
alaKaantyaMyota  
Drooni, [13](#)  
alaKaantyaVasta  
Drooni, [13](#)  
alaLiikkuaEteen  
Drooni, [13](#)  
alaLiikkuaTaakse  
Drooni, [14](#)  
alustaAmmus  
Ammus, [9](#)  
Miina, [22](#)  
AlyMatkustaja, [7](#)  
ammu  
Drooni, [14](#)  
ammuksiaJaljella  
Drooni, [14](#)  
Ammus, [7](#)  
alustaAmmus, [9](#)  
Ammus, [8](#)  
asetaAmmuskuva, [9](#)  
etene, [9](#)  
osuma, [10](#)  
suuntaPiste, [10](#)  
tarkistaOsuma, [10](#)  
tuhooAmmus, [10](#)  
ulkoOsuma, [11](#)  
ammus.hh, [35](#)  
asetaAmmuskuva  
Ammus, [9](#)  
Laser, [19](#)  
Miina, [22](#)  
Rynnakkokivaari, [31](#)  
asetaKartta  
Pelilikkuna, [27](#)  
asetaStatus

Pelilikkuna, [28](#)  
  
Drooni, [11](#)  
~Drooni, [13](#)  
alaKaantyaMyota, [13](#)  
alaKaantyaVasta, [13](#)  
alaLiikkuaEteen, [13](#)  
alaLiikkuaTaakse, [14](#)  
ammu, [14](#)  
ammuksiaJaljella, [14](#)  
Drooni, [13](#)  
drooniAmpunut, [15](#)  
drooniLiikkunut, [15](#)  
keyPressEvent, [15](#)  
keyReleaseEvent, [15](#)  
liikuSulavasti, [15](#)  
lopetaEteenLiikkuminen, [16](#)  
lopetaKaannosMyota, [16](#)  
lopetaKaannosVasta, [16](#)  
lopetaTaakseLiikkuminen, [16](#)  
valitseLaser, [16](#)  
valitseMiina, [17](#)  
valitseRynkky, [17](#)  
drooni.hh, [35](#)  
drooniAmpunut  
Drooni, [15](#)  
drooniLiikkunut  
Drooni, [15](#)  
  
etene  
Ammus, [9](#)  
  
Kaupunki, [17](#)  
ajastaLopetus, [18](#)  
kaupunki.hh, [36](#)  
kerroKaupunki  
Nyssekuva, [24](#)  
keyPressEvent  
Drooni, [15](#)  
keyReleaseEvent  
Drooni, [15](#)  
kirjoitaTiedostoon  
Tilasto, [32](#)  
  
Laser, [18](#)  
~Laser, [19](#)  
asetaAmmuskuva, [19](#)  
Laser, [19](#)  
tarkistaOsuma, [20](#)  
laser.hh, [36](#)

- liikuSulavasti
  - Drooni, [15](#)
- lisaa\_nysse
  - Peliikkuna, [28](#)
- lisaa\_pysakki
  - Peliikkuna, [28](#)
- lisaaDrooni
  - PeliAlue, [26](#)
  - Peliikkuna, [29](#)
- lopetaEteenLiikkuminen
  - Drooni, [16](#)
- lopetaKaannosMyota
  - Drooni, [16](#)
- lopetaKaannosVasta
  - Drooni, [16](#)
- lopetaTaakseliikkuminen
  - Drooni, [16](#)
- LopetusDialog, [20](#)
- lue\_tiedosto
  - Tilasto, [32](#)
- Miina, [21](#)
  - ~Miina, [22](#)
  - alustaAmmus, [22](#)
  - asetAmmuskuva, [22](#)
  - Miina, [21](#)
  - osuma, [23](#)
  - tarkistaOsuma, [23](#)
- miina.hh, [37](#)
- naytaToplista
  - Peliikkuna, [29](#)
- Nyssekuva, [23](#)
  - ~Nyssekuva, [24](#)
  - kerroKaupunki, [24](#)
  - Nyssekuva, [24](#)
  - osuma, [24](#)
  - tuhottu, [25](#)
  - tuhoudu, [25](#)
- nyssekuva.hh, [37](#)
- osuma
  - Ammus, [10](#)
  - Miina, [23](#)
  - Nyssekuva, [24](#)
- paivita\_tilasto
  - Tilasto, [33](#)
- PaivitaTulokset
  - Peliikkuna, [29](#)
- PeliAlue, [25](#)
  - lisaaDrooni, [26](#)
- Peliikkuna, [26](#)
  - ~Peliikkuna, [27](#)
  - asetAmmuskuva, [27](#)
  - asetStatus, [28](#)
  - lisaa\_nysse, [28](#)
  - lisaa\_pysakki, [28](#)
  - lisaaDrooni, [29](#)
  - naytaToplista, [29](#)
  - PaivitaTulokset, [29](#)
  - Peliikkuna, [27](#)
  - Peliikkuna, [27](#)
  - pelialue.hh, [37](#)
  - pelikkuna.hh, [38](#)
  - Rynnakkokivaari, [30](#)
    - ~Rynnakkokivaari, [31](#)
    - asetAmmuskuva, [31](#)
    - Rynnakkokivaari, [30](#)
  - rynnakkokivaari.hh, [38](#)
  - selvitaListasijoitus
    - Tilasto, [33](#)
  - suuntaPiste
    - Ammus, [10](#)
  - tarkistaOsuma
    - Ammus, [10](#)
    - Laser, [20](#)
    - Miina, [23](#)
  - Tilasto, [31](#)
    - kirjoitaTiedostoon, [32](#)
    - lue\_tiedosto, [32](#)
    - paivita\_tilasto, [33](#)
    - selvitaListasijoitus, [33](#)
  - tuhoaAmmus
    - Ammus, [10](#)
  - tuhottu
    - Nyssekuva, [25](#)
  - tuhoudu
    - Nyssekuva, [25](#)
  - ulkoOsuma
    - Ammus, [11](#)
  - valitseLaser
    - Drooni, [16](#)
  - valitseMiina
    - Drooni, [17](#)
  - valitseRynkky
    - Drooni, [17](#)