

TeamDank



CARL
THE CRASHER

Spooks

nightlife

FOOOFEUO

CARL
THE CRASHER

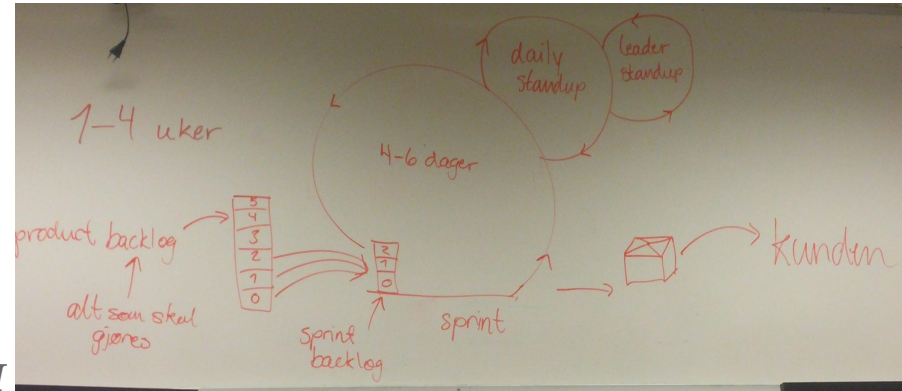
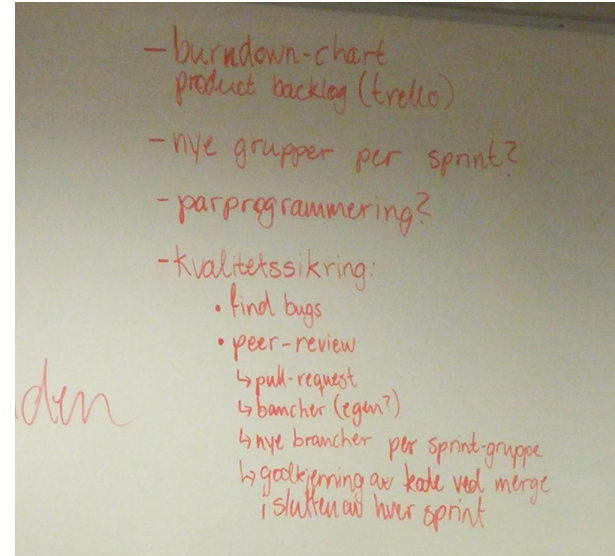
FOOD FEUD

Organisering

- Ingen “tydelig” leder
 - Kommunikasjonsansvarlig
- God kommunikasjonsflyt i gruppen
 - Slack
- Undergrupper med forskjellige ansvarsområder
 - Kodegruppe
 - Grafikkgruppe
 - Legal team
- GitLab Flow
 - Feature branch
 - Merge request
 - Pipelines
 - Kvalitetsstyring
 - Versjonskontroll
- SonarQube
 - Prosjektadministrasjon
 - Kvalitetsstyring
- Endringer fra oblig 3
 - SCRUM - modifisert

Sprint 1

- Diskuterte utviklingsmetoder
 - SCRUM
 - KANBAN
- Inndeling i par / 3 og 3
- GUI, generelle klasser, grafikk
- Bruke alle verktøy
 - Vanskeligere enn forventet
- Laget tester



Sprint 2

- Brukte ikke Trello riktig
 - Burndown Chart
- Git Issues
 - SCRUM-lignende boards
- Planning Poker
 - Tidkrevende
- Fordele oppgaver etter kompetanse

<input type="checkbox"/>	Bygninger og annet ved veien #38 · opened 2 weeks ago by Peter Johansen	sprint2	Sprint2	To Do
<input type="checkbox"/>	Rydd opp CarGame #37 · opened 2 weeks ago by Peter Johansen	sprint2	libGDX	To Do
<input type="checkbox"/>	Special render for overridden layers #36 · opened 2 weeks ago by Peter Johansen	sprint2	libGDX	To Do
<input type="checkbox"/>	weather data oppdatere previouspulltime #35 · opened 2 weeks ago by Markus Johan Ragnhildstveit	sprint2		
<input type="checkbox"/>	implement inventory classes #34 · opened 2 weeks ago by Ole Magnus Lie	sprint2	Backlog	Doing
<input type="checkbox"/>	Fix GameScreen test #33 · opened 3 weeks ago by Peter Johansen	sprint2	Testing	
<input type="checkbox"/>	Clear screen on render #32 · opened 3 weeks ago by Peter Johansen	sprint2	libGDX	To Do

Sprint 3 og 4

- Organisering på plass
- Ordentlig arbeidsflyt
- Oppgaver fordelt på annen måte
 - Git Issues
 - Assignee - velge issue selv
- Store endringer
 - Sluttet med testing



















<input type="checkbox"/>	juksekode vær	#194 · opened about 23 hours ago by Markus Johan Ragnhildstveit	sprint4	CarGame	Doing
<input type="checkbox"/>	rapport organisering	#193 · opened a day ago by Håvar Eggereide	sprint4	To Do	6 of 6 tasks completed
<input type="checkbox"/>	prosjekt skal kunne ta dm av player	#191 · opened a day ago by Kenneth Apeland	sprint4	Doing	FoodFeud
<input type="checkbox"/>	Skjult knapp foodfeud	#190 · opened a day ago by Håvar Eggereide	sprint4	Done	FoodFeud
<input type="checkbox"/>	neste person når "eg" dør	#189 · opened a day ago by Kenneth Apeland	sprint4	Done	FoodFeud
<input type="checkbox"/>	Oppdatere spillmanual	#188 · opened a day ago by Markus Johan Ragnhildstveit	sprint4	Doing	
<input type="checkbox"/>	Forenkle setupScreen og fikse font	#187 · opened a day ago by Kenneth Apeland	sprint4	Done	FoodFeud

Prosesser

- Før hver sprint:
 - Lage issue i Git
 - Sette en assignee
 - Opprette og løse issue i egen branch
 - Skrive tester til egen kode
 - Merge request
 - Annen person går gjennom koden, og merger med master

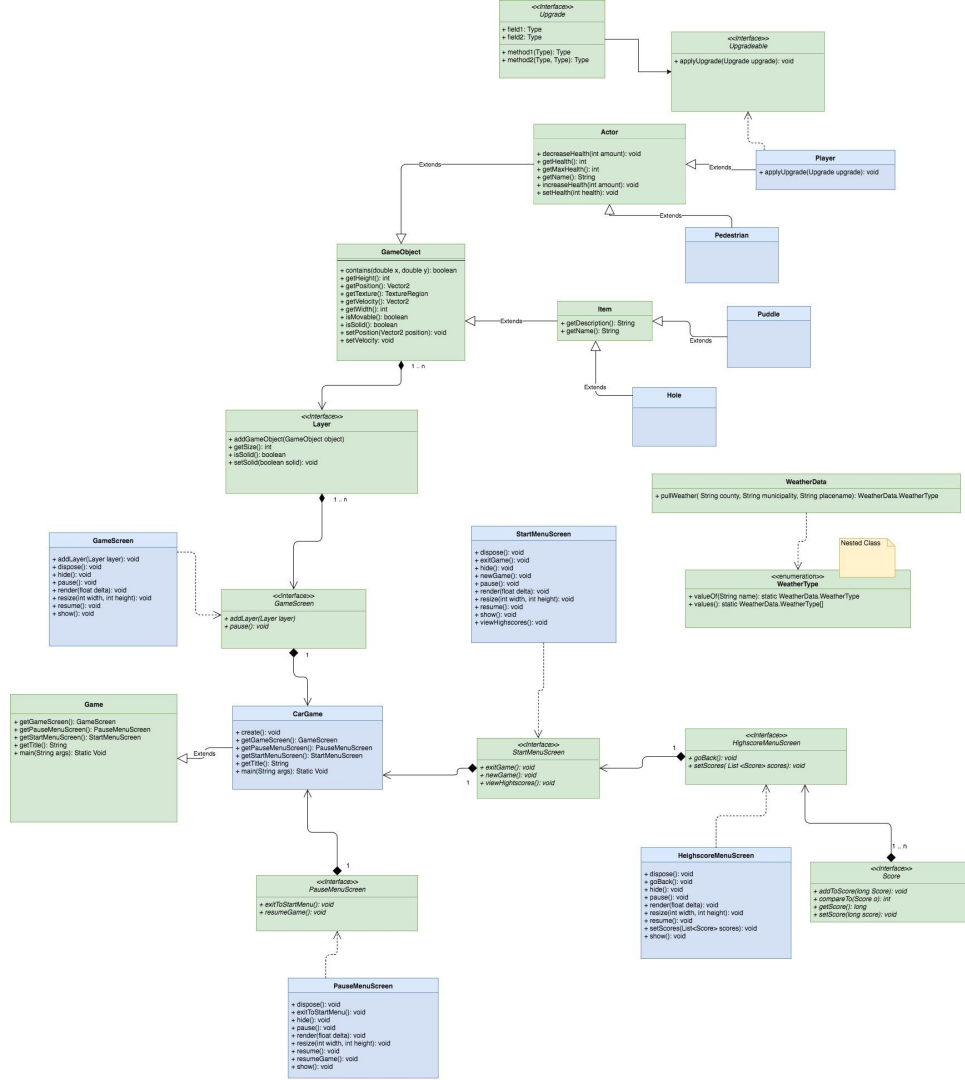
Gitlab

- Vise gitlab
- Pipeline
- Sonarqube?
- Pipeline knelte i slutten
 - (både build og sonarqube)

passed	#2018 by 	🔗 master → 1e9a5473 Merge branch '219-change-si...	🟢🟢🟢🟢	🕒 00:02:12 📅 7 minutes ago
passed	#2017 by 	🔗 219-change-size-... → 43860e66 fjernet static	🟢🟢🟢	🕒 00:02:47 📅 10 minutes ago
passed	#2016 by 	🔗 master → ead62883 mindre tid	🟢🟢🟢🟢	🕒 00:05:01 📅 10 minutes ago
failed	#2015 by 	🔗 master → 7b2632ca manual	🟢🔴🔴🔴	🕒 00:04:20 📅 13 minutes ago
canceled	#2014 by 	🔗 master → 48edd7c8 Merge branch '215-lyder-til-fo...	🔴🔴🔴	🕒 00:04:20 📅 19 minutes ago
canceled	#2013 by 	🔗 master → 30d8aaab Merge branch '177-copyright'...	🔴🔴🔴	🕒 00:00:40 📅 18 minutes ago
canceled	#2012 by 	🔗 master → 4bc42d0a Diagram	🔴🔴🔴	🕒 00:00:32 📅 19 minutes ago
failed	#2011 by 	🔗 219-change-size-... → efce4c93 ballistics?	🟢🟢🔴	🕒 00:00:32 📅 16 minutes ago
failed	#2009 by 	🔗 188-oppdaterer-sp... → 6cb70552 endringer	🟢🔴	🕒 00:00:32 📅 1 minute ago
canceled	#2008 by 	🔗 186-foodfeud-rel... → dc7fc4c8 Merge branch 'master' of http...	🔴🔴🔴	🕒 00:00:32 📅 19 minutes ago
canceled	#2007 by 	🔗 184-carlthecrash... → 58eb2289 Merge branch 'master' into 18...	🔴🔴🔴	🕒 00:01:33 📅 20 minutes ago
canceled	#2006 by 	🔗 master → 82015399 Merge branch '188-oppdaterer...	🔴🔴🔴	🕒 00:02:59 📅 19 minutes ago
canceled	#2005 by 	🔗 master → dc7fc4c8 Merge branch 'master' of http...	🔴🔴🔴	🕒 00:01:59 📅 21 minutes ago
passed	#2003 by 	🔗 215-lyder-t11-fo... → 21116e75 Merge branch 'master' into '2...	🟢🟢🟢	🕒 00:05:52 📅 19 minutes ago
failed	#2002 by 	🔗 215-lyder-t11-fo... → 99ccdfae fjernet debug	🔴🔴🔴	🕒 00:00:10 📅 25 minutes ago
passed	#2001 by 	🔗 177-copyright → f47a6df1 Copyright	🟢🟢🟢	🕒 00:06:11 📅 19 minutes ago
canceled	#2000 by 	🔗 215-lyder-t11-fo... → 43bf2b57 musikk	🟢🔴	🕒 00:02:09 📅 24 minutes ago
failed	#1999 by 	🔗 master → bdf03ca8 Merge branch '224-ikke-skyte...	🟢🔴🔴	🕒 00:04:05 📅 23 minutes ago
passed	#1997 by	🔗 177-copyright → 4789d513 Merge branch '223-more-than...	🟢🟢🟢	🕒 00:03:28 📅 25 minutes ago

Endringer i design

- Endret alt i designmodellen
 - Minileveranse etter hver sprint
- LibGDX
- Scene2D
- Box2D



```

classDiagram
    class Player {
        +Player()
        +applyUpgrade(Upgrade upgrade) : void
    }
    class Actor {
        +Actor()
        +Actor(int mh, int ch, java.lang.String n)
        +Actor(int h, java.lang.String n)
        +getHealth() : int
        +decreaseHealth(int amount) : void
        +getMaxHealth() : int
        +getName() : java.lang.String
        +increaseHealth(int amount) : void
        +setHealth(int health) : void
    }
    class Pedestrian {
        +Pedestrian()
    }
    class Puddle {
        +Puddle(java.lang.String n, java.lang.String d)
    }
    class Hole {
        +Hole(java.lang.String n, java.lang.String d)
    }
    class Item {
        +Item(java.lang.String n, java.lang.String d)
        +getDescription() : java.lang.String
        +getName() : java.lang.String
    }
    class GameObject {
        +GameObject()
        +GameObject(float x, float y)
        +GameObject(float x, float y)
        +GameObject(com.badlogic.gdx.graphics.g2d.TextureRegion tRegion)
        +contains(double x, double y) : boolean
        +getHeight() : int
        +getWidth() : int
        +getPosition() : com.badlogic.gdx.math.Vector2
        +getTexture() : com.badlogic.gdx.graphics.g2d.TextureRegion
        +getVelocity() : com.badlogic.gdx.math.Vector2
        +isMarkedForRemoval() : boolean
        +isMovable() : boolean
        +isSolid() : boolean
        +markForRemoval() : void
        +render(float delta) : void
        +setScale(float s) : void
        +setTexture(com.badlogic.gdx.graphics.g2d.TextureRegion texture texture) : void
    }
    class CarGame {
        +CarGame()
        +create() : void
        +getGameScreen() : GameScreen
        +getPauseMenuScreen() : PauseMenuScreen
        +getStartMenuScreen() : double
        +getSpriteBatch() : com.badlogic.gdx.graphics.g2d.SpriteBatch
        +getTitle() : java.lang.String
        +main(java.lang.String[] args) : static void
    }
    class WeatherData {
        +WeatherData()
        +pullWeather(java.lang.String county, java.lang.String municipality, java.lang.String placeName) : WeatherData WeatherType
    }
    class WeatherType {
        +WeatherType()
        +CLOUD
        +RAIN
        +SNOW
        +SUN
        +valueOf(java.lang.String name) : static WeatherData WeatherType
        +values() : static WeatherData WeatherType
    }
    class GameScreen {
        +GameScreen(Game game)
        +dispose() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +show() : void
        +update(float delta) : void
    }
    class Game {
        +Game()
        +getGameScreen() : abstract GameScreen
        +getPauseMenuScreen() : abstract PauseMenuScreen
        +getStartMenuScreen() : abstract StartMenuScreen
        +getSpriteBatch() : com.badlogic.gdx.graphics.g2d.SpriteBatch
        +getTitle() : abstract java.lang.String
    }
    class StartMenuScreen {
        +StartMenuScreen(CarGame game)
        +exitGame() : void
        +newGame() : void
        +viewHighScores() : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +resume() : void
        +show() : void
        +ting(float velocityX, float velocityY, int button) : boolean
        +pan(float x, float y, float deltaX, float deltaY) : boolean
        +panStop(float x, float y, int pointer, int button) : boolean
        +pinch(com.badlogic.gdx.math.Vector2 initialPointer, com.badlogic.gdx.math.Vector2 pointer1, com.badlogic.gdx.math.Vector2 pointer2) : void
        +pinchStop() : void
        +tap(float x, float y, int count, int button) : void
        +touchDown(float x, float y, int pointer, int button) : void
        +zoom(float initialDistance, float distance) : void
    }
    class PauseMenuScreen {
        +PauseMenuScreen()
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +void resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Upgradeable {
        +getInventory() : Inventory
    }
    class Upgrade {
        +getDescription() : String
        +getLevel() : int
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function.Consumer<GameObject> consumer) : void
        +getSize() : int
        +setSolid(boolean solid) : void
        +isSolid() : boolean
        +removeMarkedGameObjects() : void
    }
    class StartMenuScreen {
        +StartMenuScreen()
        +exitGame() : void
        +newGame() : void
        +viewHighScores() : void
    }
    class PauseMenuScreen {
        +PauseMenuScreen()
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +void resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Upgradeable {
        +getInventory() : Inventory
    }
    class Upgrade {
        +getDescription() : String
        +getLevel() : int
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function.Consumer<GameObject> consumer) : void
        +getSize() : int
        +setSolid(boolean solid) : void
        +isSolid() : boolean
        +removeMarkedGameObjects() : void
    }
    class StartMenuScreen {
        +StartMenuScreen()
        +exitGame() : void
        +newGame() : void
        +viewHighScores() : void
    }
    class PauseMenuScreen {
        +PauseMenuScreen()
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +void resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Upgradeable {
        +getInventory() : Inventory
    }
    class Upgrade {
        +getDescription() : String
        +getLevel() : int
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function.Consumer<GameObject> consumer) : void
        +getSize() : int
        +setSolid(boolean solid) : void
        +isSolid() : boolean
        +removeMarkedGameObjects() : void
    }
    class StartMenuScreen {
        +StartMenuScreen()
        +exitGame() : void
        +newGame() : void
        +viewHighScores() : void
    }
    class PauseMenuScreen {
        +PauseMenuScreen()
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +void resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Upgradeable {
        +getInventory() : Inventory
    }
    class Upgrade {
        +getDescription() : String
        +getLevel() : int
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function.Consumer<GameObject> consumer) : void
        +getSize() : int
        +setSolid(boolean solid) : void
        +isSolid() : boolean
        +removeMarkedGameObjects() : void
    }
    class StartMenuScreen {
        +StartMenuScreen()
        +exitGame() : void
        +newGame() : void
        +viewHighScores() : void
    }
    class PauseMenuScreen {
        +PauseMenuScreen()
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +void resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Upgradeable {
        +getInventory() : Inventory
    }
    class Upgrade {
        +getDescription() : String
        +getLevel() : int
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function.Consumer<GameObject> consumer) : void
        +getSize() : int
        +setSolid(boolean solid) : void
        +isSolid() : boolean
        +removeMarkedGameObjects() : void
    }
    class StartMenuScreen {
        +StartMenuScreen()
        +exitGame() : void
        +newGame() : void
        +viewHighScores() : void
    }
    class PauseMenuScreen {
        +PauseMenuScreen()
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +void resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Upgradeable {
        +getInventory() : Inventory
    }
    class Upgrade {
        +getDescription() : String
        +getLevel() : int
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function.Consumer<GameObject> consumer) : void
        +getSize() : int
        +setSolid(boolean solid) : void
        +isSolid() : boolean
        +removeMarkedGameObjects() : void
    }
    class StartMenuScreen {
        +StartMenuScreen()
        +exitGame() : void
        +newGame() : void
        +viewHighScores() : void
    }
    class PauseMenuScreen {
        +PauseMenuScreen()
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +void resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Upgradeable {
        +getInventory() : Inventory
    }
    class Upgrade {
        +getDescription() : String
        +getLevel() : int
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function.Consumer<GameObject> consumer) : void
        +getSize() : int
        +setSolid(boolean solid) : void
        +isSolid() : boolean
        +removeMarkedGameObjects() : void
    }
    class StartMenuScreen {
        +StartMenuScreen()
        +exitGame() : void
        +newGame() : void
        +viewHighScores() : void
    }
    class PauseMenuScreen {
        +PauseMenuScreen()
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +void resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Upgradeable {
        +getInventory() : Inventory
    }
    class Upgrade {
        +getDescription() : String
        +getLevel() : int
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function
```

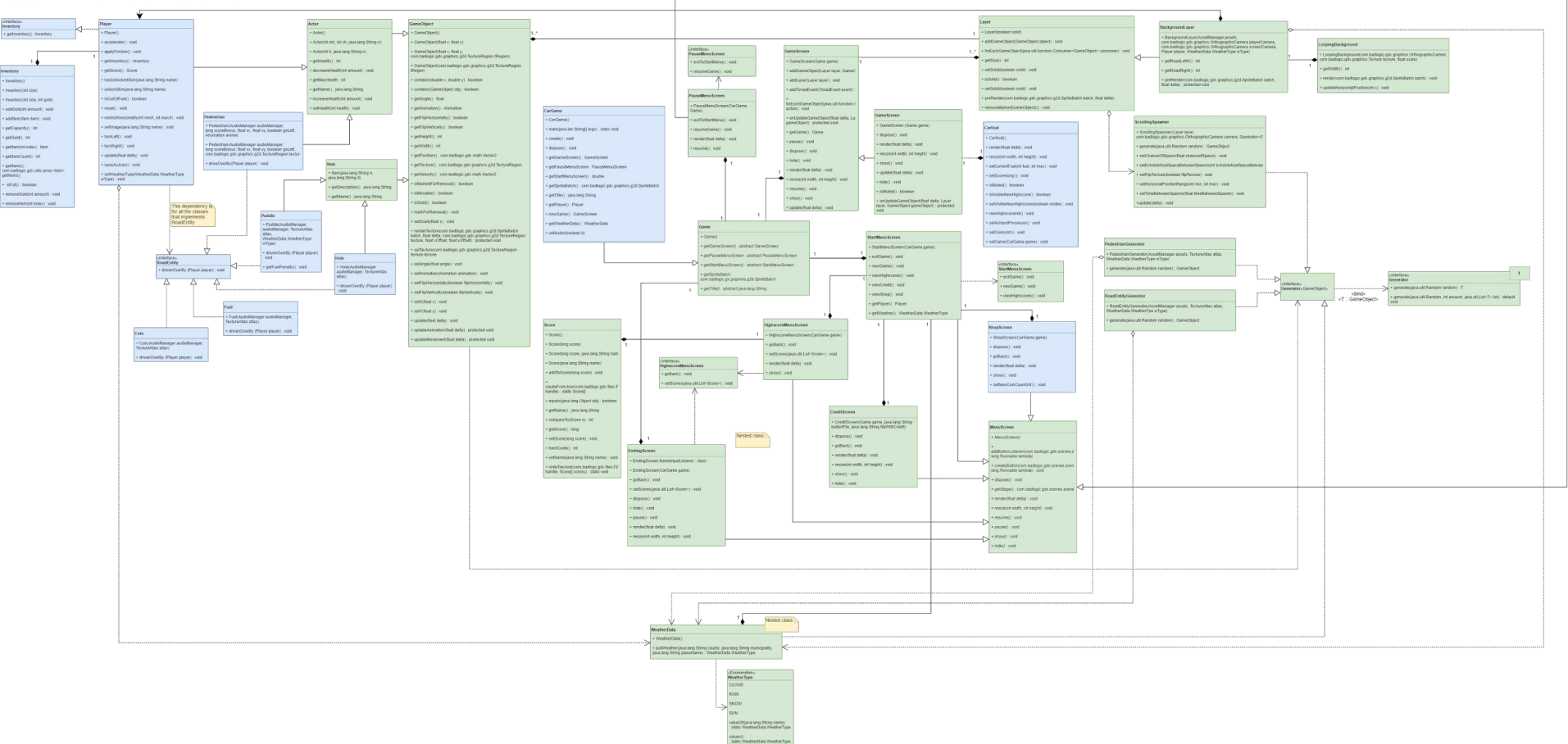
```

classDiagram
    class Player {
        +Player()
        +applyUpgrade(Upgrade upgrade) : void
    }
    class Actor {
        +Actor()
        +Actor(int mh, int ch, java.lang.String n)
        +Actor(int h, java.lang.String n)
        +getHealth() : int
        +decreaseHealth(int amount) : void
        +getMaxHealth() : int
        +getName() : java.lang.String
        +increaseHealth(int amount) : void
        +setHealth(int health) : void
    }
    class GameObject {
        +GameObject()
        +GameObject(float x, float y)
        +GameObject(float x, float y)
        +GameObject(com.badlogic.gdx.graphics.g2d.TextureRegion tRegion)
        +contains(double x, double y) : boolean
        +getWidth() : int
        +getHeight() : int
        +getWidthIn() : int
        +getPosition() : com.badlogic.gdx.math.Vector2
        +getTexture() : com.badlogic.gdx.graphics.g2d.TextureRegion
        +getVelocity() : com.badlogic.gdx.math.Vector2
        +isMarkedForRemoval() : boolean
        +isMovable() : boolean
        +isSolid() : boolean
        +markForRemoval() : void
        +setScale(float s) : void
        +setTexture(com.badlogic.gdx.graphics.g2d.TextureRegion texture) : void
        +getTexture() : java.lang.String
    }
    class Layer {
        +Layer(boolean solid)
        +addGameObject(GameObject object) : void
        +forEachGameObject(java.util.function.Consumer<GameObject> consumer) : void
        +getSize() : int
        +setSold(boolean sold) : void
        +isSold() : boolean
        +removeMarkedGameObjects() : void
    }
    class Background_Layer {
        +Background_layer(AssetManager.assets, com.badlogic.gdx.graphics.OrthographicCamera playerCamera, com.badlogic.gdx.graphics.OrthographicCamera screenCamera, Player player)
        +getRoadLeftX() : int
        +getRoadRight() : int
        +preRender(com.badlogic.gdx.graphics.g2d.SpriteBatch batch, float delta) : protected void
    }
    class GameScreen {
        +addGameObject(Layer layer, GameC)
        +addLayer(Layer layer) : void
        +getGame() : Game
        +pause() : void
        +dispose() : void
        +hide() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
        +update(float delta) : void
    }
    class PauseMenu_Screen {
        +PauseMenuScreen(CarGame game)
        +exitToStartMenu() : void
        +resumeGame() : void
        +dispose() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
    }
    class CarGame {
        +CarGame()
        +create() : void
        +dispose() : void
        +getGameScreen() : GameScreen
        +getPauseMenuScreen : PauseMenuScreen
        +getStartMenuScreen() : double
        +getSpriteBatch() : com.badlogic.gdx.graphics.g2d.SpriteBatch
        +getTitle() : java.lang.String
        +main(java.lang.String[] args) : static void
        +getPlayer() : Player
    }
    class WeatherData {
        +WeatherData()
        +pullWeather(java.lang.String county, java.lang.String municipality, java.lang.String placeName) : WeatherData.WeatherType
    }
    class NestedClass {
        +NestedClass()
    }
    class Enumerated_WeatherType {
        CLOUD
        RAIN
        SNOW
        SUN
        valueOf(java.lang.String name) : static WeatherData.WeatherType
        values() : static WeatherData.WeatherType
    }
    class Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class Interface_Score {
        +addScore(long score) : default void
        +compareTo(Score o) : default int
        +getScore() : long
        +setScore(long score) : void
    }
    class EndingScreen {
        +EndingScreen(CarGame game)
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
        +setupButton(java.lang.String) : com.badlogic.gdx.scenes.scene2d.ui.ImageButton
    }
    class HighscoreMenuScreen {
        +HighscoreMenuScreen()
        +goBack() : void
        +setScores(java.util.List<Score>) : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +resume() : void
        +show() : void
        +setupButton(java.lang.String imageString) : com.badlogic.gdx.scenes.scene2d.ui.ImageButton
    }
    class StartMenuScreen {
        +StartMenuScreen(CarGame game)
        +exitGame() : void
        +newGame() : void
        +viewHighscores() : void
        +dispose() : void
        +hide() : void
        +pause() : void
        +render(float delta) : void
        +resize(int width, int height) : void
        +resume() : void
        +show() : void
        +getPlayer() : Player
        +setupButton(java.lang.String imageString) : com.badlogic.gdx.scenes.scene2d.ui.ImageButton
    }
    class CarHUD {
        +CarHUD()
        +render(float delta) : void
        +resize(int width, int height) : void
        +setCurrentFuel(int fuel, int max) : void
        +show() : void
        +setScore(long f) : void
    }
    class LoopingBackground {
        +LoopingBackground(com.badlogic.gdx.graphics.OrthographicCamera camera, com.badlogic.gdx.graphics.Texture texture, float scale)
        +getWidth() : int
        +render(com.badlogic.gdx.graphics.g2d.SpriteBatch batch) : void
        +updateHorizontalPosition(int x) : void
    }
    class Interface_StartMenuScreen {
        +exitGame() : void
        +newGame() : void
        +viewHighscores() : void
    }

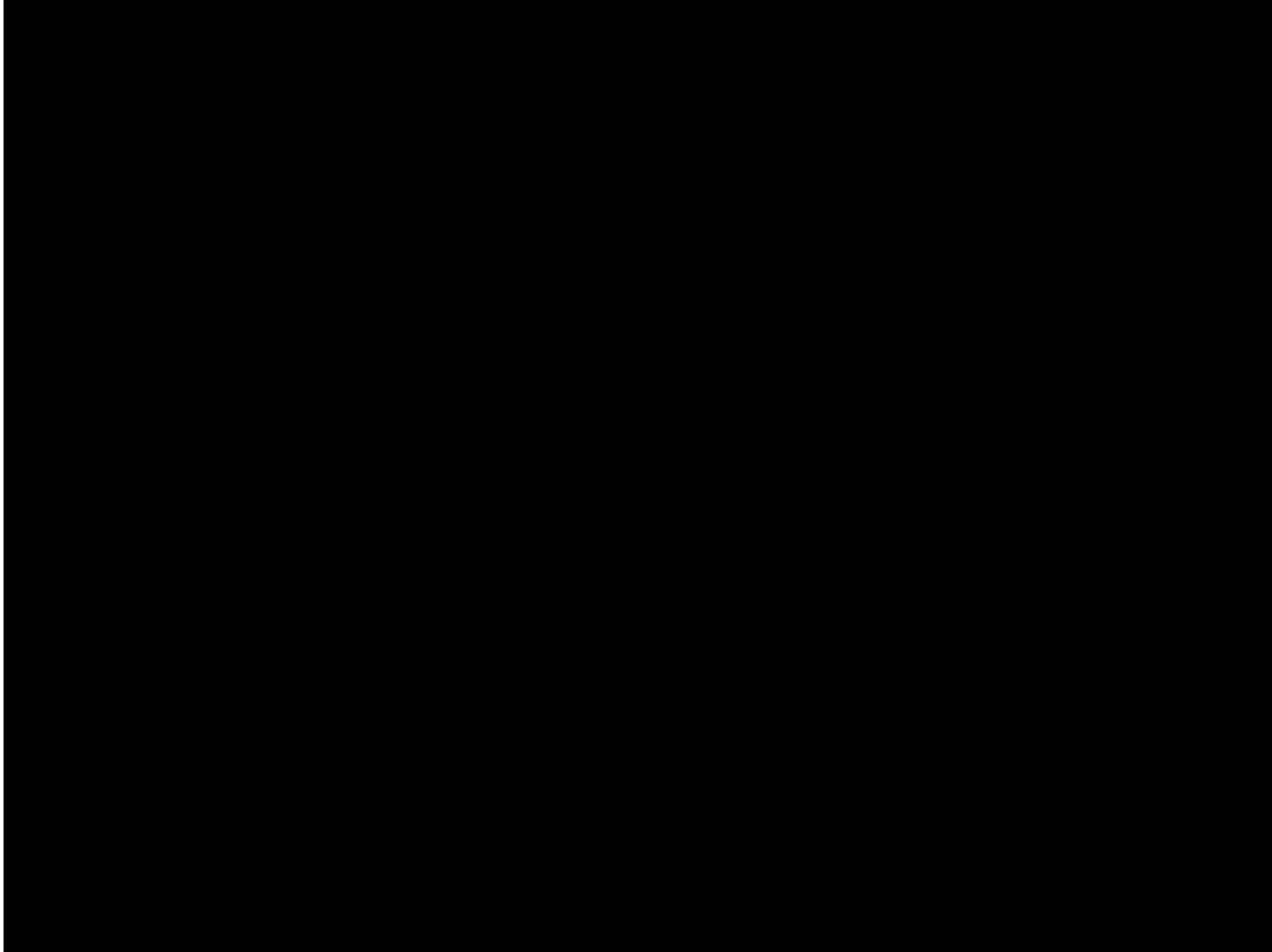
    Player --> Actor
    Actor --> GameObject
    GameObject --> Layer
    Layer --> Background_Layer
    GameScreen --> PauseMenu_Screen
    GameScreen --> Layer
    PauseMenu_Screen --> GameScreen
    CarGame --> GameScreen
    CarGame --> PauseMenu_Screen
    CarGame --> StartMenuScreen
    CarGame --> Score
    CarGame --> EndingScreen
    CarGame --> HighscoreMenuScreen
    WeatherData ..> Enumerated_WeatherType
    Enumerated_WeatherType --|> Interface_Score
    Score --|> Interface_Score
    EndingScreen --|> Interface_EndingScreen
    HighscoreMenuScreen --|> Interface_HighscoreMenuScreen
    StartMenuScreen --|> Interface_StartMenuScreen
    CarHUD --|> Interface_CarHUD
    LoopingBackground --|> Interface_LoopingBackground

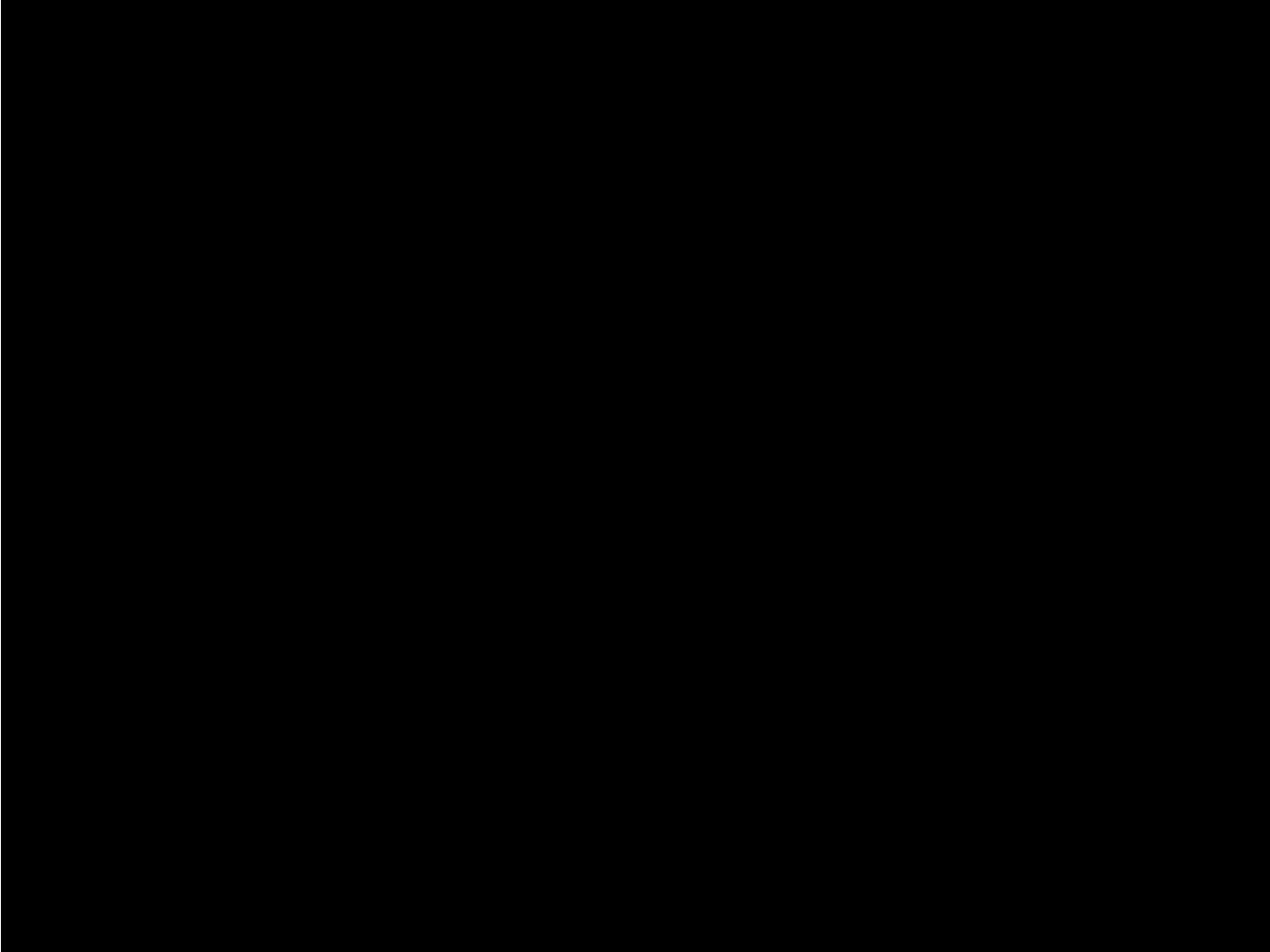
```

Design diagram for Carl the Crasher (CarGame) ((Final) Version 5, Sprint 4, 10.05.2017)



[illegible]





Videreutvikling

- Kjernekode
 - Software product line
 - Harvest common code
- Carl the Crasher:
 - Flere baner
 - Mer i butikken
- FoodFeud:
 - Flere baner
 - Flere matvarer
 - Flere personer
- Implementere Spooks og Nightlife

Refleksjon

- Resultater oppnådd
- Milestones skulle vært mer presis definert
- Læringsutbytte
 - Jobbe i større gruppe
 - Ta ansvar
 - Holde korte frister

sprint2

100% complete

29 Issues · 0 Merge Requests

(Expired) Apr 19, 2017–Apr 26, 2017

Edit

Close Milestone

Delete

sprint3

100% complete

46 Issues · 2 Merge Requests

(Expired) Apr 26, 2017–May 3, 2017

Edit

Close Milestone

Delete

sprint4

84% complete

77 Issues · 6 Merge Requests

May 3, 2017–May 10, 2017

Edit

Close Milestone

Delete

Milestones i Git