#### Technisch Ontwerp

Project : Oil’s Well

Bedrijf : MBO Utrecht



|  |  |  |  |
| --- | --- | --- | --- |
| **Plaats, datum:**  **Versie:** | Hilversum, 06-10-2015  0.1 | | |
| **Opgesteld door:** | Projectgroep Oil’s Well | | |
|  | **[Studentnaam]** | **[Studentnummer]** | **[Mailadres]** |
|  | Gerben Tesselaar | 305931 | Gemme123@hotmail.com |

#### Inhoudsopgave Technisch ontwerp

[1. Achtergrond 2](#_Toc422425165)

[2. Apparatuur 3](#_Toc422425166)

[3. Software 5](#_Toc422425167)

[4. Database (website) 6](#_Toc422425168)

[5. Klassen 7](#_Toc422425169)

[6. Sequencediagram (movement) 9](#_Toc422425170)

**Versie beheer:**

|  |  |
| --- | --- |
| **Versie** | **Wijzigingen** |
| **V0.1** | **Toegevoegd: Achtergrond, Apparatuur, Software, database (website)** |
|  |  |
|  |  |
|  |  |

# 1. Achtergrond

Het project heet “Oil’s Well”, dit is tevens ook de naam van het spel wat wordt nagemaakt in dit project. De opdracht gevende organisatie is in dit geval MBO Utrecht. De opdrachtgever is M. Odijk. De opdrachtnemer ben ik als leerling Gerben Tesselaar. Ik ben tevens ook verantwoordelijk voor de uitvoering van dit project.

De bedoeling in het spel is om zoveel mogelijk punten te behalen, dit doe je door de snoepjes op te pakken. Je zit vast aan een lijn en kan niet naar beneden lopen daarvoor in de plaats trekt de lijn je terug naar je vorige positie. Er zijn “power” snoepjes aanwezig die je kracht geven om de monsters op te eten, hiervoor krijg je ook punten.

# 2. Apparatuur

**Computer**

**Computertype** Op ACPI x64-gebaseerde PC (Mobile)

**Besturingssysteem** Microsoft Windows 8

**OS Service Pack** -

**Internet Explorer** 9.11.9600.17631

**DirectX** DirectX 11.0

**Computernaam** JOHNDOE-NOTE

**Naam gebruiker** g

**Aanmeld Domein** JOHNDOE-NOTE

**Moederbord**

**CPU Type** Mobile DualCore Intel Core i5, 2600 MHz

**Moederbordnaam** Asus X75A Series Notebook

**Moederbord Chipset** Intel Panther Point HM76, Intel Ivy Bridge

**Systeemgeheugen** 5581 MB

**BIOS Type** Onbekend

**Scherm**

**Video kaart** Intel(R) HD Graphics 4000 (2560 MB)

**Video kaart** Intel(R) HD Graphics 4000 (2560 MB)

**Video kaart** Intel(R) HD Graphics 4000 (2560 MB)

**3D Acceleratie** Intel HD Graphics 4000

**Monitor** Acer AL1914 [19" LCD] (ETL2908101)

**Multimedia**

**Geluidskaart** Intel Panther Point HDMI @ Intel Panther Point PCH - High Definition Audio Controller [C-1]

**Geluidskaart** Realtek ALC269 @ Intel Panther Point PCH - High Definition Audio Controller [C-1]

**Opslag**

**IDE controller** Intel(R) 7 Series Chipset Family SATA AHCI Controller

**Opslag controller** Controlefunctie voor opslagruimten van Microsoft

**Harde schijf** HGST HTS541075A9E680 (750 GB, 5400 RPM, SATA-III)

**Optische Station** DTSOFT Virtual CdRom Device

**Optische Station** MATSHITA DVD-RAM UJ8E1

**Partities**

**C: (NTFS)** 279.0 GB (91.1 GB vrij)

**D: (NTFS)** 398.1 GB (69.5 GB vrij)

**Geheugencapaciteit** 677.1 GB (160.6 GB vrij)

**Invoerapparaten**

**Toetsenbord** HID-toetsenbordapparaat

**Toetsenbord** PC/AT-uitgebreid PS/2-toetsenbord (101/102 toetsen)

**Muis** ASUS Touchpad

**Netwerk**

**Netwerkkaart** 802.11n draadloze LAN-kaart

**Netwerkkaart** LogMeIn Hamachi Virtual Ethernet Adapter (25.112.209.197)

**Netwerkkaart** Microsoft Hosted Network Virtual Adapter #2

**Netwerkkaart** Microsoft Wi-Fi Direct Virtual Adapter

**Netwerkkaart** Qualcomm Atheros AR8161/8165 PCI-E Gigabit Ethernet Controller

**Netwerkkaart** TAP-Win32 Adapter V9 (Tunngle)

**Netwerkkaart** VirtualBox Host-Only Ethernet Adapter (169.254.225.41)

**Externe apparaten**

**USB2 controller** Intel Panther Point PCH - USB 2.0 EHCI Controller #1 [C-1]

**USB2 controller** Intel Panther Point PCH - USB 2.0 EHCI Controller #2 [C-1]

**USB3 controller** Intel Panther Point PCH - USB 3.0 xHCI Controller [C-1]

**USB-apparaat** Generic USB Hub

**USB-apparaat** Generic USB Hub

**USB-apparaat** Samengesteld USB-apparaat

**USB-apparaat** Samengesteld USB-apparaat

**USB-apparaat** USB2.0 HD UVC WebCam

**USB-apparaat** USB-invoerapparaat

**USB-apparaat** USB-invoerapparaat

**Batterij** Accu die voldoet aan Microsoft ACPI-besturingsmethode

**DMI**

**DMI Systeem Fabrikant** ASUSTeK COMPUTER INC.

**DMI Systeem Product** X75A

**DMI systeem Versie**  1.0

**DMI Systeem Serienummer** E4N0CX791298177

**DMI System UUID** 11000000-00000000-000010C3-7B66CACA

**DMI Moederbordfabrikant** ASUSTeK COMPUTER INC.

**DMI Moederbordproduct** X75A

**DMI Moederbordversie** 1.0

**DMI Moederbord Serienummer** BSN12345678901234567

**DMI chassisfabrikant** ASUSTeK COMPUTER INC.

**DMI chassisversie** 1.0

**DMI chassisserienummer**  E4N0CX791298177

**DMI chassistype** Notebook

# 3. Software

**Programmma** Mozilla Firefox 38.0.5 (x86 nl)

**Versie** 38.0.5

**Geïnstalleerde** **grootte** Onbekend

**GUID** Mozilla Firefox 38.0.5 (x86 nl)

**Publisher** Mozilla

**Programmma** WampServer

**Versie** 2.5

**Geïnstalleerde** **grootte** Onbekend

**GUID** WampServer 2\_is1

**Publisher** Hervé Leclerc (HeL)

**Programmma** Test Mail Server Tool

**Versie** 2.54

**Geïnstalleerde** **grootte** Onbekend

**GUID** {5781A356-8BC3-4AD1-8214-DDD0CCA85B2A}

**Publisher** Toolheap

**Programmma** Java 8 Update 45 (64-bit)

**Versie** 8.0.450

**Geïnstalleerde** **grootte** Onbekend

**GUID** {26A24AE4-039D-4CA4-87B4-2F86418045F0}

**Publisher** Oracle Corporation

**Programmma** Java SE Development Kit 8 Update 45 (64-bit)

**Versie** 8.0.450.15

**Geïnstalleerde** **grootte** Onbekend

**GUID** {64A3A4F4-B792-11D6-A78A-00B0D0180450}

**Publisher** Oracle Corporation

**Programmma** Eclipse

**Versie** Luna Service Release 1a (4.4.1)

**Geïnstalleerde** **grootte** Onbekend

**GUID** -

**Publisher** Eclipse

# 4. Database (website)

**Structuur**

**Database naam:** oilswell

**Tabel:** users

**Veld:** id int(11) AUTO\_INCREMENT

**Veld:** username varchar(45) -

**Veld:**  password varchar(45) -

**Veld:**  email varchar(70) -

**Veld:**  userrole enum('administrator', 'user') -

**Veld:** activated tinyint(1) -

**Inserts**

19 admin 5f4dcc3b5aa765d61d8327deb882cf99 admin@mail.com administrator 1

20 henk 5f4dcc3b5aa765d61d8327deb882cf99 henk@mail.com user 1

21 peter 5f4dcc3b5aa765d61d8327deb882cf99 peter@mail.com user 0

22 jan 5f4dcc3b5aa765d61d8327deb882cf99 jan@mail.com user 0

# 5. Klassen

**Johndoe.game**

**Config.java**

Geen methodes

**OilsWell.java**

@Override

Public void create()

@Override

Public void render()

**Johndoe.game.entities.monster**

**Monster.java**

Public Rectangle getCollisionBox()

Public void setCollisionBox(int x, int y, int width, int height)

Public Monster(AssetManager manager, SpriteBatch batch, int id)

Public void draw()

Public void update(float delta)

Public void dispose()

Public void delete()

Private void checkCollision()

**Johndoe.game.entities.player**

**ObjectActions.java**

Private static void removeTiles(String used\_by, TiledMap map, RectangleMapObject rectangleObject)

Public static void destroyPowerCandy(TiledMap map, RectangleMapObject rectangleObject, Player player)

Public static void destroyCandy1(TiledMap map, RectangleMapObject rectangleObject, Player player)

Public static void destroyCandy2(TiledMap map, RectangleMapObject rectangleObject, Player player)

Public static void destroyCandy3(TiledMap map, RectangleMapObject rectangleObject, Player player)

**Player.java**

Public Animation getCurrent\_animation()

Public void setCurrent\_animation(Animation current\_animation)

Public Animation getAnimation(String Animation)

Public Vector2 getPos()

Public void setPos(Vector2 position)

Public int getFacing()

Public void setFacing()

Public float getHeight()

Public float getWidth()

Public Boolean isMovingLeft()

Public Boolean isMovingRight()

Public Boolean isMovingUp()

Public Boolean isMovingDown()

Public float getSpeed()

Public void setSpeed()

Public void setLoop()

Public Rectangle getCollisionBox()

Public ArrayList<Sprite> getRopes()

Public Boolean hasPower()

Public void takePower()

Public void givePower()

Public int setScore()

Public int getLife()

Public void setLife()

Public Player(AssetManager manager, SpriteBatch batch)  
Public void draw()

Public void update(float delta)

Public void dispose()

Public void respawn()

Private void saveOldPosition()

Private void drawOldPosition()

Private void updatePosition(float delta)

Public void setLeftMove(Boolean bool)

Public void setRightMove(Boolean bool)

Public void setUpMove(Boolean bool)

Public void setDownMove(Boolean bool)

Private void checkCollision()

**PlayerInput.java**

Public PlayerInput(Player player)

@Override

Public Boolean keyDown(int keycode)

@Override

Public Boolean keyUp(int keycode)

@Override

Public Boolean keyTyped(char character)

**Johndoe.game.font**

**BmapFont.java**

Public static void init()

Public static void dispose()

**Johndoe.game.listeners**

**LabelListener.java**

Public LabelListener(Actor actor)

@Override

Public void enter(InputEvent event, float x, float y, int pointer, Actor fromActor)

@Override

Public void exit(InputEvent event, float x, float y, int pointer, Actor toActor)

@Override

Public void clicked(InputEvent event, float x, float y)

**Johndoe.game.renderers**

**TextureMapObjectAndTileRenderer.java**

public TextureMapObjectAndTileRenderer(TiledMap map)

public TextureMapObjectAndTileRenderer(TiledMap map, Batch batch)

public TextureMapObjectAndTileRenderer(TiledMap map, float unitScale)

public TextureMapObjectAndTileRenderer(TiledMap map, float unitScale, Batch batch)

@Override

Public void renderObject(MapObject object)

@Override

public void renderObjects(MapLayer layer)

**johndoe.game.screens**

**EndGame.java**

public EndGame(int score)

@Override

public void show()

@Override

public void render(float delta)

@Override

public void resize(int width, int height)

@Override

public void pause()

@Override

public void resume()

@Override

public void hide()

@SuppressWarnings("static-access")

@Override

public void dispose()

public static void save()

**Highscores.java**

@Override

public void show()

@Override

public void render(float delta)

@Override

public void resize(int width, int height)

@Override

public void pause()

@Override

public void resume()

@Override

public void hide()

@Override

public void dispose()

**Loading.java**

@Override

public void show()

@Override

public void render(float delta

@Override

public void resize(int width, int height)

@Override

public void pause()

@Override

public void resume()

@Override

public void hide()

@Override

public void dispose()

**MainMenu.java**

public MainMenu(boolean init)

@Override

public void show()

@Override

public void render(float delta)

@Override

public void resize(int width, int height)

@Override

public void pause()

@Override

public void resume()

@Override

public void hide()

@SuppressWarnings("static-access")

@Override

public void dispose()

private void setupSound()

private void setupStage()

@SuppressWarnings("static-access")

private void setupTween()

private void addLabelListeners()

public static void stopMusic()

**Play.java**

public Play(AssetManager manager)

@Override

public void show()

@Override

public void render(float delta)

@Override

public void resize(int width, int height)

@Override

public void pause()

@Override

public void resume()

@Override

public void hide()

@Override

public void dispose()

**Splash.java**

@Override

public void show()

@Override

public void render(float delta)

@Override

public void resize(int width, int height)

@Override

public void pause()

@Override

public void resume()

@Override

public void hide()

@Override

public void dispose()

**johndoe.game.tween**

**ActorAccessor.java**

@Override

public int getValues(Actor target, int tweenType, float[] returnValues)

@Override

public void setValues(Actor target, int tweenType, float[] newValues)

**SpriteAccessor.java**

@Override

public int getValues(Sprite target, int tweenType, float[] returnValues)

@Override

public void setValues(Sprite target, int tweenType, float[] newValues)

**johndoe.game.utils**

**PowerTimer.jave**

public PowerTimer(Player player)

@Override

public void run()

**SaveFileInteract.java**

public static void saveScore(FileHandle fileHandle, String object)

public static ArrayList<String> getScores(FileHandle fileHandle)

# 6. Sequencediagram (movement)

