KARELIA-AMMATTIKORKEAKOULU

Tietojenkäsittelyn koulutusohjelma

Janne Romppanen 1101096

VISUAALINEN NOVELLI

Opinnäytetyö

Marraskuu 2014

|  |  |  |
| --- | --- | --- |
| Karelia_musta_jpg_SU | **OPINNÄYTETYÖ**  **Elokuu 2014**  **Kaikki koulutukset**  Tikkarinne 9  80220 JOENSUU  Keskuksen puhelinnumero | |
| Tekijä(t)  Opinnäytetyöryhmä ja Karelia-amk | | |
| Nimeke  Opinnäytetyön ohje  Toimeksiantaja  PKAMK ja Karelia-amk | | |
| Tiivistelmä  Tässä opinnäytetyön ohjeessa annetaan yleisohjeita opinnäytetyön tekemiseen Karelia-ammattikorkeakoulussa (Karelia-amk). Opinnäytetyön ohje esittelee opinnäytetyön prosessia Karelia-amk:ssa ja antaa ohjeet kirjallisen raportin laatimiseen sekä opinnäytetyöprosessissa ja raportissa esiin tuleviin eettisiin kysymyksiin.  Opinnäytetyön prosessissa esitellään tutkimustyyppisen ja toiminnallisen opinnäytetyön lähtökohtia, rakennetta sekä erityispiirteitä. Kirjallisen raportin osalta käydään läpi opinnäytetyön kirjoittamisen prosessia, kielenhuollon asioita, tekstidokumentin asetteluohjeita, viittauskäytäntöjä sekä lähdeluettelon laatimiseen liittyviä seikkoja.  Ammattikorkeakoulun opinnäytetyötä koskeva ohjeistus muuttuu ajassa. Kukin opiskelija ja opiskelijaryhmä noudattavat opinnäytetyöprosessin alussa voimassa olevaa ohjetta. Tämän ohjeen lisäksi opiskelijan tulee perehtyä opinnäytetyön muuhun ohjeistukseen PAKKI-opiskelija­portaalissa, opinnäytetyön laadintaa koskevaan kirjallisuuteen sekä oman työnsä kannalta tärkeään menetelmäkirjallisuuteen. Ohjeisiin tehdään tarvittaessa päivityksiä ja päivityskuukausi ja -vuosi ilmoitetaan ohjeen kansilehdessä. | | |
| Kieli  suomi | | Sivuja 36  Liitteet 12  Liitesivumäärä |
| Asiasanat  opinnäytetyö, opinnäytetyön prosessi, kirjallinen raportti, opinnäytetyön eettiset kysymykset | | |

|  |  |  |
| --- | --- | --- |
| Karelia_musta_jpg_EN | **THESIS**  **August 2014**  **All Degree Programmes**  Tikkarinne 9  80220 JOENSUU  FINLAND  Telephone number of the centre | |
| Author (s)  Thesis coordination group and KUAS | | |
| Title  Thesis Instructions  Commissioned by  NKUAS and KUAS | | |
| Abstract  In this thesis instructions document general guidelines are given for writing thesis in Karelia University of Applied Sciences (KUAS). The thesis goes through the thesis process in KUAS, gives instructions in writing the written report as well as informs about ethical questions in thesis and report writing processes.  In the thesis process part of the document prerequisites, structure and special features of writing both a research and project based thesis are presented. Furthermore, the written report part covers thesis writing process, language planning, documentation styles, citations, and references.  The thesis instructions change in time. Each student and student group follow the specific instructions that are valid when starting the thesis process. In addition to these instructions, the student is expected to be familiar with other instructions in PAKKI web portal, literature that covers thesis writing and specific method literature that is of importance for each student´s own thesis. The instructions are updated when necessary and the month and the year of the updated version are given on the first page of the instructions. | | |
| Language  Finnish | | Pages 36  Appendices 12  Pages of Appendices |
| Keywords  thesis, thesis process, written report, ethical questions in thesis | | |

**Sisältö**

1 Kehitystyökalujen tietoperusta 3

1.1 Johdanto 3

1.2 Kehitystyökalujen tietoperusta 4

1.3 Aktiiviset työkalut 5

1.3.1 Ren’py 5

1.3.2 Belle 7

1.3.3 WebStory Engine 8

1.3.4 Visual Novel Toolkit 11

1.4 Ei-aktiiviset työkalut 13

1.4.1 Novelty 13

1.4.2 JS-Vine 15

1.5 Japaninkielisiä kehitysympäristöjä 16

1.5.1 NScripter / ONScripter / ONScripter-EN / PONScripter 16

1.5.2 KiriKiri ( KiriKiri2/KAG3 ) 17

1.6 Lopuksi 17

2 Visuaalisten novellien tietoperusta 18

2.1 Kerronnalliset elementit 18

2.1.1 ADV 18

2.1.2 NVL 19

2.1.3 Taistelu 20

2.1.4 Inventaario 22

2.1.5 Hahmon kehitys 23

2.1.6 Kartta 23

2.2 Tarina 24

2.3 Tekniset elementit 30

2.3.1 Tallennus 30

2.3.2 Lataus 30

3 Prototyyppi 30

3.1 Ren’py 31

3.1.1 Työkalun käyttö 31

Yleiset 31

Kuva 31

Ääni 32

Valintarakenne 33

Debug 33

3.2 WebStory Engine 34

3.3 Visual Novel Toolkit 38

3.4 Novelty 39

Liitteet

Liite 1 Opinnäytetyön sisältö ja osaamistavoitteet

Liite 2 Opinnäytetyön eteneminen

Liite 3 Opinnäytetyösuunnitelman malli

Liite 4 Tutkimustyyppisen ja toiminnallisen opinnäytetyön rakenne

Liite 5 Esimerkit tekstiin sijoitettavista kuviosta, taulukosta ja kuvasta

Liite 6 Suomenkielisen tiivistelmän tekstin asettelu

Liite 7 Vieraskielisen tiivistelmän tekstin asettelu

Liite 8 Tarkistusohje opinnäytetyöraportin viimeistelyyn

Liite 9 Nimiölehden tekstin asettelu

Liite 10 Esimerkki lyhenneluettelosta

Liite 11 Kansi- ja selkätarran ulkoasu

Liite 12 Tekniikan koulutusohjelmien numeroviittausjärjestelmä

Kehitystyökalujen tietoperusta

Johdanto

Tämä harjoitusaine tutkii, mitä erilaisia kehitysympäristöjä on tarjolla visuaalisten novellien kehittämiseen. Visuaaliset novellit ovat digitaalisessa muodossa olevia kuvitettuja novelleja, joissa on usein myös ääntä. Yleensä käyttäjä voi vaikuttaa novellin juonen kulkuun tekemällä valintoja dialogissa. Visuaalisten novellien kehitykseen erikoistuneet kehitysympäristöt tarjoavat työkalut dialogin, kuvan ja äänen yhdistämiseen sekä juonen rakentamiseen. Kuva ja ääni kehitetään yleensä jollain ulkopuolisella työkalulla, dialogi ja juonirakenne kehitysympäristöllä. Jotkin kehitysympäristöt tukevat myös videoita ja animaatioita.

Harjoitusaineessa esitellään myös pelkkiä moottoreita, joissa ei ole varsinaista kehitysympäristöä. Näissä tapauksissa novelli rakennetaan moottorin omalla skriptauskielellä käyttämällä tekstieditoria. Idea on kuitenkin sama eli yhdistetään eri elementit yhdeksi julkaisuksi.

Käytän tekstissäni yhteisiä termejä tietyille asioille. Näitä termejä ovat työkalut ja julkaisu. Termiä työkalut käytän tekstissä, kun viittaan yleisesti kehitysympäristöihin ja moottoreihin. Termiä julkaisu käytän, kun viittaan työkalulla tehtyyn julkaisuun.

Visuaaliset novellit ovat erityisen suosittuja Japanissa. Siellä ne jaotellaan kerronnallisiin ja toiminnallisiin novelleihin. Kerronnallisissa novelleissa pääpaino on tarinalla ja tekstiä näytetään yleensä useita rivejä kerrallaan. Varsinaista vuorovaikutusta novellin ja lukijan välillä on vähän. Toiminnallisissa novelleissa pääpaino on vuorovaikutuksessa ja tekstiä näytetään yleensä yksi rivi kerrallaan. Toiminnalliset novellit ovat yleensä seikkailupelejä, jotka voivat sisältää roolipelielementtejä. Japanissa vuonna 2006 julkaistuista PC-peleistä melkein 70 prosenttia oli toiminnallisia novelleja (Wikipedia 2014a).

Tämän tutkimuksen tarkoituksena on luoda yhtenäinen ja kattava raportti kaikista tällä hetkellä julkaistuista työkaluista, joissa on tuki Englannin kielelle. Raportin pohjalta voidaan siten valita sopivin kehitysympäristö omaan käyttötarkoitukseen. Tässä tutkimuksessa ei käsitellä keskeneräisiä työkaluja.

Kehitystyökalujen tietoperusta

Olen jaotellut työkalut kolmeen osioon: aktiiviset, ei-aktiiviset ja japaninkieliset. Erottelu aktiivisiin ja ei-aktiivisiin on tehty, jotta lukija on tietoinen valintaa tehdessään, onko työkalulla aktiivista tukea tai kehitysyhteisöä, onko päivityksiä ja kehitystä odotettavissa vai onko työkalu jäänyt ns. oman onnensa nojaan mutta on vielä kuitenkin täysin soveltuva sille annettuun tehtävään. japaninkieliset työkalut on myös eroteltu omaksi luvukseen. Vaikkei japaninkielisistä kehitysympäristöistä ole suoranaista hyötyä kieltä osaamattomalle, haluan kuitenkin mainita pari suosituinta kehitysympäristöä Japanissa olevan suuren fanipohjan takia. Vähäisen englanninkielisen tiedon takia olen jättänyt pois niiden tarkemmat ominaisuudet.

Kaikista työkaluista on esitelty tärkeimmät ominaisuudet omissa taulukoissaan. Keskeisimpiä omaisuuksia ovat seuraavat seikat: onko kyseessä kokonaisvaltainen kehitysympäristö vai pelkkä moottori, millä ohjelmointikielellä moottori on kehitetty, mitä ohjelmointikieltä ja skriptikieltä työkalut tukevat visuaalisen novellin kehityksessä, mitä käyttöjärjestelmiä ne tukevat ja mille alustoille visuaalisen novellin voi julkaista.

Lisäksi tarkastelen sitä, mihin käyttötarkoituksiin kyseinen työkalu soveltuu. Esittelen myös työkalun kehittäjän sekä millä lisenssillä kyseinen työkalu on julkaistu. Lopuksi vertailen työkaluja keskenään ja kerron oman näkemykseni parhaiten eri tarkoituksiin soveltuvista työkaluista.

Aktiiviset työkalut

Tässä luvussa on esitelty tällä hetkellä aktiivisessa kehityksessä olevat työkalut. Olen määritellyt työkalun aktiiviseksi, jos kehittäjä sitä edelleen aktiivisesti kehittää tai jos kyseessä on avoimen lähdekoodin työkalu, jolla on aktiivinen kehitysyhteisö.

Ren**’**py

Ren’py Tom "PyTom" Rothamelin kehittämä kehitysympäristö visuaalisten novellien kehittämiseen. Ensimmäinen versio julkaistiin 24.8.2004. Taulukossa 1 esitetty versionumero 6.18.1 on julkaistu 22.9.2014. Pitkä kehityskaari näkyy mm. kehitysympäristöjen tehtyjen julkaisujen määrässä, joita on yli 700 kappaletta. (Wikipedia 2014b.)

Ren’pyn päätyökalu on Ren’py Launcher. Se on graafinen projektien hallintatyökalu, jolla voi luoda uusia projekteja, hallita ja muokata jo olemassa olevia sekä suorittaa niitä. (Ren’py 2014a.) Uusi projekti sisältää tarvittavat skripititiedostot julkaisun asetusten ja sisällön määrittelyyn. Ren’py Launcher sisältää myös mm. kääntäjän (compiler) koodin kääntämiseen, käännöstyökalun tekstisisällön kääntämiseen eri kielille sekä julkaisutyökalun taulukossa 1 määritellyille julkaisualustoille.

Ren’pyyn on kehitetty oma Pythoniin pohjautuva skriptikieli, joka vaikuttaa pohjautuvan Python-skriptikieleen (Ren’py 2014b). Ren’pyssä on myös tuki Pythonille, jolla voidaan kehittää moottorin ulkopuolisia toiminnallisuuksia. Ren’pyyn on kehitetty myös näyttökieli (screen language). Näyttökielellä määritellään julkaisun käyttöliittymät kuten päävalikko, tallennusvalikko tai puhevalikko. (Ren’py 2014c.)

Luvun lopussa olevassa taulukossa 1, esitetystä Genret-listasta nähdään, että Ren’pyllä voidaan tehdä monen tyylisiä julkaisuja. Julkaisut kattavat mm. simulaatiot, vuoropohjaisen taistelun ja roolipelit (Ren’py 2014d).

Ren’pyn lähdekoodi on pääosin MIT-lisenssin alaista (Ren’py 2014e). Se on avoimen lähdekoodin ohjelmisto ja sen käyttö on ilmaista kaupalliseen ja ei-kaupalliseen käyttöön. (Ren’py 2014f).

Yhteenvetona voin todeta, että Ren’py on avoimen lähdekoodin kehitysympäristö monelle käyttöjärjestelmälle. Siinä on tehokkaat skriptaustyökalut ja siitä löytyy kattava valikoima työkaluja visuaalisissa novelleissa yleisesti käytettävien ominaisuuksien tekoon. Tiedostotuki kattaa yleisimmät tyypit mukaan lukien videotiedostot. Lisäksi Ren’py on dokumentoitu hyvin sekä sillä on aktiivinen kehitysyhteisö. (Ren’py 2014f.) Ren’pyllä on pitkä kehityskaari ja sillä on tehty yli 700 julkaisua useille kielille (Ren’py 2014g). Ren’Py onkin mielestäni erittäin suositeltava vaihtoehto paikallisten (standalone) julkaisujen työkaluksi.

Taulukko 1. Ren’pyn ominaisuuksia. (Ren’py 2014f; Ren’py 2014e.)

|  |  |
| --- | --- |
| Kehitysympäristö | |
| Versionumero | 6.18.1 |
| Julkaisupäivämäärä | 22.9.2014 |
| Pääominaisuudet | Pythonilla toteutettu pelimoottori  Python skriptikielellä määriteltävät novellin tekniset elementit  Ren’pyn omalla skriptikielellä määriteltävät novellin tarinalliset elementit  Ren’py Launcher. Graafinen työkalu projektien luomiseen ja hallintaan |
| Formaatti | itsenäinen tiedosto |
| Käyttökohteet | visuaaliset novellit  kineettiset novellit (Kinetic novels)  roolipelit  simulaatiot  sekalaiset |
| Kehitysalusta | Windows XP+ (x86)  Mac OS X 10.6+ (x86\_64)  Linux (x86, x86\_64)  Sekondäärinen tuki:  Android (2.0+, OpenGL ES 2.0 tuella) |
| Julkaisualusta | sama kuin kehitysalusta |

Belle

Belle on Carlos Paisin kehittämä kehitysympäristö visuaalisten novellien kehittämiseen (Carlos 2013a). Ensimmäinen versio (0.1a) editorista ja moottorista on julkaistu 15.11.2012 (Carlos 2013b; Carlos 2013c). Luvun lopussa olevassa taulukossa 2, esitetty versionumero 0.4.2:sta ei ole virallisesti ilmoitettua julkaisupäivämäärää. Viimeisin tiedostojen muokkauspäivämäärä on päivätty 12.9.2014. Lisäksi Bellen GitHub-säiliöön (repository) tehty viimeisin muokkaus on tehty 6.10.2014 (tarkistettu 8.10.2014). Tästä voimme todeta, että Belle on aktiivisessa kehityksessä.

Bellen kehityksessä on otettu esimerkkiä Noveltystä, joka on samantyyppinen kehitysympäristö (Belle 2014a). Novelty on esitelty luvussa 3 ”Ei-aktiiviset työkalut”. Belle on kehitysympäristö. Sen päätyökalu on WYSIWYG (What You See Is What You Get) editori jossa kehitystyö tapahtuu. Editorilla voi lisätä graafiset elementit kuten painikkeet (buttons) ja dialogi-ikkunat. (Belle 2014b.) Belle ei tue mitään skriptikieltä vaan toiminnot tehdään editorin toimintotyökalulla (actions). Esim. dialogille ja äänentoistamiselle on omat toimintonsa. (Belle 2014c.) Kehittäjän sivuilla ei ole tietoa Bellellä kehitetyistä visuaalisista novelleista. Koska Belle on vasta Alpha versiossa, voidaan olettaa, ettei sillä ole vielä kehitetty kaupallisia novelleja.

Belle on julkaistu GNU GPLv3-lisenssillä. Se on avoimen lähdekoodin ohjelmisto ja sen käyttö on ilmaista kaupalliseen ja ei-kaupalliseen käyttöön. (Belle 2014a.)

Tiedostotuki kattaa yleisimmät tyypit mukaan lukien videotiedostot. Se soveltuu web-sovellusten kehitykseen visuaalisella editorilla. Siinä ei ole skriptaustyökalua vaan ohjelmointi hoidetaan editorilla täysin graafisesti. Vaikka Belle onkin vasta Alpha-versiossa siitä löytyvät työkalut yleisimpien visuaalisen novellin ominaisuuksien kehitykseen. Belleä kehitetään aktiivisesti ja dokumentaation taso on ainakin perustason kattavaa. Belleltä kuitenkin puuttuu vielä aktiivinen kehitysyhteisö eikä kotisivuilta löydy vielä forumia, joten omaan kehitystyöhön voi olla hankalaa saada apua. Mielestäni Belle sopii hyvin web-sovellusten kehitykseen, kun ei ole tarvetta kustomoiduille ominaisuuksille skriptikieltä käyttäen. Kannattaa kuitenkin ottaa huomioon kehitysyhteisön puute ja siitä mahdollisesti aiheutuvat ongelmat.

Taulukko 2. Bellen ominaisuuksia. (Belle 2014a.)

|  |  |
| --- | --- |
| Kehitysympäristö | |
| Versionumero | 0.4.2. Alpha (Windows ja Mac OS) |
| Muokkauspäivämäärä | 12.9.2014 |
| Pääominaisuudet | C++:lla ja QT4:llä toteutettu WYSIWYG-editori  HTML 5:llä, CSS:llä ja JavaScriptillä toteutettu pelimoottori |
| Formaatti | Web-sovellus |
| Käyttökohteet | visuaaliset novellit |
| Kehitysalusta | Windows XP+  GNU / Linux  Mac OS X (10.4+) |
| Julkaisualusta | verkkoselain online- tai offline-tilassa |

WebStory Engine

WebStory Engine on Jonathan Steinbeckin kehittämä moottori visuaalisten novellien kehittämiseen (GitHub 2014b). Ensimmäisen version julkaisupäivämäärä ei ole selvillä, mutta ensimmäinen lisäys (commit) WebStory Enginen GitHub-tilille (repository) on tehty 26.6.2012. (Steinbeck 2012a). Luvun lopussa olevassa taulukossa 3, esitetty versionumero on julkaistu jo melkein vuosi sitten. Toisaalta viimeisin lisäys GitHub-tilille on tehty 3.9.2014 (Steinbeck 2014b). Lisäksi viimeisin viesti moottorin kotisivujen forumille on lisätty 30.9.2014. (iiYO 2014). Lisäksi kotisivuilta löytyy pohdintaa seuraavan sukupolven WebStory Engine-moottorin ominaisuuksista. Siinä on pohdittu nykyisen moottorin huonoja puolia ja esitetty kehittämisideoita. (WebStory Engine 2014a; WebStory Engine 2014b.) Kaikki nämä seikat puhuvat sen puolesta, että moottori on edelleen aktiivisessa kehityksessä ja sillä on vahva suuntautuminen tulevaisuuteen.

WebStory Enginen moottori on kirjoitettu JavaScript-kielellä. Moottori käyttää XML-kielellä määriteltyjä elementtejä joista varsinainen julkaisu koostuu. Elementtien avulla määritellään julkaisun asetukset ja sisältö yhteen XML-tiedostoon, joka linkitetään HTML-indeksisivulle. Varsinainen julkaisu näytetään verkkoselaimessa indeksisivulla. XML-tiedosto on jaoteltu kolmeen osaan: asetukset (settings), resurssit (assets) ja kohtaukset (scenes). Asetukset sisältävät novellin teknisiä määrittelyjä kuten leveyden ja korkeuden sekä näppäinmäärittelyt. Resurssit sisältävät käytössä olevien sisällöllisten resurssien (esim. kuva- ja äänitiedostot) määrittelyt kuten tiedostopolun resurssiin ja resurssin nimen. Kohtaukset käyttävät resursseja sekä niihin määritellään varsinainen leipäteksti. Kohtauksia voi olla useita ja ne erotellaan omiksi kohtauselementeiksi. Kohtauksiin myös määritellään ehdot, kuinka tarina etenee. (WebStory Engine 2014c; WebStory Engine 2014d.)

WebStory Enginen kotisivuilla ei ole tarkemmin kerrottu moottorin ominaisuuksista. Sivuilla olevassa oppaassa on esimerkiksi kohta Animations, mutta siihen ei ole lisätty vielä sisältöä (WebStory Engine 2014e). Lukijalle jää vain oletus, että moottori ehkä pystyy käsittelemään animaatioita jollain määrittämättömällä tavalla. Tämä on mielestäni suuri puute ja kertoo työkalun keskeneräisyydestä.

WebStory Engine on julkaistu BSD 3-Clause-lisenssillä (GitHub 2014c). Salliva lisenssityyppi mahdollistaa moottorin lähdekoodin muokkaamisen sekä uusien toiminnallisuuksien lisäämisen että myös uusien XML-elementtien luomisen.

WebStory Enginen kotisivuilla on lista julkaistuista julkaisuista (WebStory Engine 2014e). Listassa on vasta yhdeksän julkaisua mukaan lukien WebStory Enginen tutoriaali. Tämä osoittaa kyseessä olevan vielä nuori moottori, joka ei ole vielä herättänyt julkaisujen kehittäjien huomiota laajemmin. Nuoren iän lisäksi ainakin graafisen editorin puute voi olla vieraannuttava seikka sellaisille kehittäjille jotka suosivat graafista kehitystä tekstieditorilla kirjoittamisen sijaan. Kotisivujen mukaan editori on kehittäjien tulevaisuuden suunnitelmissa (WebStory Engine 2014f).

Yhteenvetona voin todeta, että WebStory Engine on avoimen lähdekoodin moottori usealle eri käyttöjärjestelmälle. Tiedostotukea ei ole mainittu kotisivuilla. Työkalu soveltuu web-sovellusten kehitykseen tekstieditorilla työkalun omalla XML-merkintäkielellä. Kehittäjä voi myös kehittää moottorin kieltä ja lisätä omia elementtejään siihen. Kehitystyyli soveltuu ohjelmoinnista pitäville kehittäjille. WebStory Engineä kehitetään aktiivisesti ja kehittäjillä on myös selkeitä tulevaisuuden suunnitelmia. Kotisivuilta löytyy oma foorumi, tosin se ei ole kovin aktiivinen. Visuaalisten novellien kehitystyöhön suuntautuneella Lemma Soft Forumilla, WebStory Enginellä on oma sijansa yhteisön keskuudessa. (Lemma Soft Forums 2014.) Työkalun dokumentointi on vielä vajaata. Esimerkiksi kaikista työkalun ominaisuuksista ei ole tietoa. Mielestäni WebStory Engine sopii kehittäjälle, joka haluaa kirjoittaa koodia moottorin antamassa kehyksessä ja ohjelmoida spesifejä ominaisuuksia juuri kyseistä julkaisua varten. Tukea on saatavilla ainakin Lemma Soft Forumilta.

Taulukko 3. WebStory Enginen ominaisuuksia. (WebStory Engine 2014f; WebStory Engine 2014g)

|  |  |
| --- | --- |
| Moottori | |
| Versionumero | 0.3.7 |
| Julkaisupäivämäärä | 1.11.2013 |
| Pääominaisuudet | HTML:llä, CSS:llä, XML:llä ja JavaScriptillä toteutettu pelimoottori  oma XML-pohjainen merkkauskieli novellien sisällön tuottamiseen |
| Formaatti | Web-sovellus |
| Käyttökohteet | visuaaliset novellit  esitelmät  interaktiiviset tutoriaalit |
| Kehitysalusta | kaikki HTML, CSS, XML ja JavaScript tuen omaavat käyttöjärjestelmät |
| Julkaisualusta | verkkoselain online- tai offline-tilassa |

Visual Novel Toolkit

Visual Novel Toolkit on Japanilaisen Sol-Tribe-yhtiön kehittämä kehitysympäristö Unity-pelimoottorille. Visual Novel Toolkitistä on kaksi eri versiota: ilmainen ja maksullinen. Ilmainen on rajoitettu versio maksullisesta. Maksullisen hinta on tällä hetkellä 30 dollaria Unityn Asset Storessa. (Asset Store 2014a; Asset Store 2014b.) Sol-Triben kotisivuilla ensimmäinen ilmoitettu julkaisupäivämäärä 4.11.2013 on annettu versiolle 1.003 (Sol-Tribe 2014a). Taulukossa 4 ilmoitettu maksullisen version versionumero 1.1.1 on julkaistu 11.3.2014. Tästä voidaan tehdä oletus, että Visual Novel Toolkit on ollut kehityksessä noin vuoden verran. Tulevasta kehityksestä ei ole englanninkielistä tietoa saatavilla.

Luvun lopussa olevassa taulukossa 4, kohdassa Pääominaisuudet mainittu editori on Visual Novel Toolkitin päätyökalu. Sitä käytetään Unityn GameView-tilassa. Editorilla voidaan luoda novellin rakenne ja dialogit sekä ehdot tarinan etenemiselle. Editorissa on myös oma äänieditori valmiiden äänitiedostojen editoimiseen. Ohjelmointi on korvattu edellä mainituilla ehdoilla, joten ohjelmointitaitoja ei tarvita. Visual Novel Toolkit tukee kuitenkin KiriKiri-skriptauskieltä, joka on KiriKiri-kehitysympäristön oma skriptauskieli (Sol-tribe 2014b). KiriKiri on esitelty luvussa 5 Japaninkielisiä kehitysympäristöjä.

Visual Novel Toolkitillä voidaan julkaista itsenäisiä (standalone) julkaisuja Unityn tukemille alustoille sekä web-sovelluksia jotka toimivat verkkoselaimessa Unity Web Playerillä. Unity tukee tällä hetkellä kuuttatoista eri alustaa sisältäen kaikki yleisimmät alustat. (Unity 2014a.) Tähän mennessä julkaistuista novelleista ei löydy tietoa. Visual Novel Toolkit käyttää Unityn lisenssiä. Tämä tarkoittaa, että lähdekoodi on suljettua mutta julkaisujen julkaisu on ilmaista lisenssin sallimissa rajoissa (Unity 2014b).

Yhteenvetona voin todeta, että Visual Novel Toolkit on suljetun lähdekoodin työkalu usealle eri käyttöjärjestelmälle. Unityn tarjoaman laajan laitetuen ansiosta Visual Novel Toolkit soveltuu paikallisten (standalone) sekä web-sovellusten julkaisuun. Myös tiedostotuki on Unityn myötä varsin kattava (Unity 2014c). Visual Novel Toolkit ei tue skriptikieliä ja kaikki ohjelmointi tehdään graafisesti editorin työkalulla. Se sopii mielestäni kehittäjille jotka käyttävät paljon Unityä kehitystyössään. Näin he voivat pysyä tutussa ympäristössä myös kehittäessä visuaalisia novelleja. Toisaalta jos kehittäjä ei käytä Unityä aiheuttaa sen käyttöönotto lisätyötä kehittäjälle, jolloin muut kehitysympäristöt voivat olla parempi vaihtoehto. Tämä on ainoa maksullinen kehitysympäristö jonka käyttöönotossa tulee ottaa huomioon myös Unityn lisenssointi. Unitystä on saatavilla ilmainen ja maksullinen versio. Ilmaisessa versiossa on rajoituksia julkaisujen suhteen. Rajoitukset ovat nähtävillä liitteessä Unity 2014b. Dokumentaationa toimii kehittäjän kotisivuilta löytyvät 74-sivuinen käyttöopas sekä äänieditori-opas (Sol-tribe 2014c). Oppaat ovat mielestäni ainakin Unityn tuntevalle tarpeeksi kattavat. Kehittäjäyhteisö löytyy Unityn forumilta jossa on useita aiheeseen liittyviä keskusteluja (Unity 2014).

Taulukko 4. Visual Novel Toolkitin ominaisuuksia. (Asset Store 2014.)

|  |  |
| --- | --- |
| Kehitysympäristö | |
| Versionumero (ilmainen) | 1.004b |
| Julkaisupäivämäärä (ilmainen) | 23.1.2014 |
| Versionumero (maksullinen) | 1.1.1 |
| Julkaisupäivämäärä (maksullinen) | 11.3.2014 |
| Pääominaisuudet | WYSIWYG-editori Unityn GameView-tilassa  KiriKiri-skriptauskieli |
| Formaatti | itsenäinen tiedosto ja web-sovellus |
| Käyttökohteet | visuaaliset novellit  dialogimekanismi Unity-julkaisuun |
| Kehitysalusta | Unityn tukemat alustat |
| Julkaisualusta | Unityn tukemat alustat |

Ei-aktiiviset työkalut

Tässä osiossa on mainitut kehitysympäristöt ovat täysin toimivia mutta niiden kehitys ja ylläpito eivät ole enää aktiivista. Työkalua voidaan kuitenkin kehittää vielä epäsäännöllisesti ja kehitysyhteisö voi olla yhä olemassa.

Novelty

Novelty on Erik Benerdalin kehittämä kehitysympäristö visuaalisten novellien kehittämiseen. Kehitystyö alkoi sivuprojektina vuonna 2007. Ensimmäinen alpha-versio ilmestyi heinäkuussa 2008. (Benerdal 2014a.) Luvun lopussa olevassa taulukossa 5, esitetty versio 0.8.9 on julkaistu 14.12.2010. Viimeisin päivitys Noveltyn kotisivuilla kehittäjän toimesta on forumilla vuodelta 2012. Pääasiallinen uutisointi näyttääkin siirtyneen kehittäjän henkilökohtaiselle Twitter-tilille. (Benerdal 2014b.) Kotisivujen forumilla on viestejä tältä vuodelta ja muutamia kuluvalta kuulta (lokakuu), joten jonkin asteista aktiivisuutta kehitysympäristön parissa on edelleen. Aktiivisuus on kuitenkin sen verran vähäistä ja viimeisestä julkaisusta on jo niin pitkä aika, etten luokittele Noveltyä aktiiviseksi työkaluksi.

Noveltyssä on poikkeava suunnitteluratkaisu muihin työkaluihin verrattuna: se on suunniteltu olio-ohjelmointimallin mukaiseksi. Tämä tarkoittaa, että kaikki julkaisussa käytettävät resurssit määritellään olioiksi NoveltyML-merkintäkielellä. (Novelty 2014a.) Noveltyn päätyökalu on WYSIWYG-editori jossa kehitystyö tapahtuu. Editorilla hallinnoidaan olioita sekä kehitetään tarinalle dialogi ja ehdot. Ehdot (actions) toimivat graafisina ohjelmointikomponentteina jotka määrittävät tarinan toiminnallisuuden ja etenemislogiikan. Ehtojen lisäksi voidaan luoda omia toiminnallisuuksia AngleScript-skriptikieltä käyttäen. (Novelty 2014b; Novelty 2014c.)

Noveltyllä tehtyjen novellien määrästä ei ole virallista lukua mutta Noveltyn kotisivujen forumilla on oma keskustelualue jossa voi esitellä Noveltylla kehitettyjä novelleja. Keskustelualueella on yli 120 aihetta. (Novelty 2014d.) Novelty on suljetun lähdekoodin ohjelmisto mutta sen käyttö on ilmaista (Novelty 2014e).

Yhteenvetona voin todeta, että Novelty on suljetun lähdekoodin kehitysympäristö Microsoft Windows -käyttöjärjestelmälle. Se soveltuu itsenäisten (standalone) julkaisujen kehitykseen. Tiedostotukea ei ole ilmoitettu kotisivuilla. Noveltyssä on tuki AngelScript-skriptikielellä sekä editorissa on työkalu graafiselle ohjelmoinnille, joten molemmat ohjelmointimuodot on katettu. Novelty on dokumentoitu kattavasti ja kotisivujen forumin julkaisukeskustelusta päätellen sillä on tehty yli sata julkaisua. Viimeisen version julkaisusta on kulunut jo kohta neljä vuotta eikä uusista julkaisuista ole ilmoitettu. Lisäksi lähdekoodi on suljettua, joten ulkopuoliset tahot eivät voi kehittää sitä. Kotisivujen forumilla on viestejä tältä vuodelta mutta vai muutama tältä kuulta (lokakuu). Tukea kehitystyöhön uskoisin löytyvän helposti kotisivujen forumilta ja Lemme Soft Forumilta. Novelty on mielestäni vaihtoehto Ren’pylle silloin kun halutaan ohjelmoida vain graafisesti. Muissa tapauksissa kuten avoimessa lähdekoodissa, käyttöjärjestelmien kattavuudessa ja työkalun kehittäjän sekä kehittäjän yhteisön aktiivisuudessa Ren’py on parempi vaihtoehto.

Taulukko 5. Noveltyn ominaisuuksia. (Novelty 2014a; Novelty 2014c; Novelty 2014d.)

|  |  |
| --- | --- |
| Kehitysympäristö | |
| Versionumero | Beta 0.8.9. |
| Julkaisupäivämäärä | 14.12.2010 |
| Pääominaisuudet | Olio-ohjelmointimallin mukainen toteutus  NoveltyML-merkintäkieli resurssien määrittelyyn olio-ohjelmointimallin mukaiseksi olioksi  WYSIWYG-editori  graafinen ohjelmointirajapinta  tuki AngleScript-skriptauskielelle  C++:lla ja Javalla toteutettu pelimoottori |
| Formaatti | itsenäinen tiedosto |
| Käyttökohteet | visuaaliset novellit |
| Kehitysalusta | Windows XP / Vista / 7  vaatii DirectX 9.0c API:n |
| Julkaisualusta | sama kuin kehitysalusta |

JS-Vine

JS-Vine on J. David Eisenbergin kehittämä moottori visuaalisten novellien ja esitysten kehittämiseen (Eisenberg 2014a). Hän kehitti moottorin esitelläkseen työtään verkossa visuaalisen novellin muodossa (Eisenberg 2014b). Eisenbergin GitHub-tilin JS-Vine-haaraan tehdyt lisäykset alkoivat kesäkuussa 2011. Viimeisin lisäys on marraskuulta 2011. (Eisenberg 2014c.) Tämän jälkeen kehitystyö näyttää päättyneen.

JS-Vinen moottorina toimii yksittäinen JavaScript-tiedosto (Eisenberg 2014d). Varsinainen julkaisu kehitetään HTML-tiedostoon, joka sijoitetaan samaan kansioon moottorin kanssa. Resursseina toimivat kuva- ja äänitiedostot voidaan lisätä omiin kansioihinsa. HTML-tiedostolle voidaan antaa tyylimuotoiluja CSS-tyyliohjeilla. Tiedostoon lisätään sisältö moottorissa määriteltyjen JavaScript-funktioiden avulla. Moottori tarjoaa funktiot kuvien, tekstikenttien ja valikoiden näyttämiseen, dialogin lisäämiseen sekä äänien toistamiseen. Valikoiden avulla saadaan ohjattua tarinan kulkua. (Eisenberg 2014e.) Tämä antaa valmiin kehyksen julkaisun kehitykseen. Koska moottori on avointa lähdekoodia, voidaan sitä jatkokehittää vapaasti. Lisäksi HTML-tiedostoon voidaan määritellä myös moottorin ulkopuolista toiminnallisuutta JavaScript-skriptikielen avulla. Valmis julkaisu lisätään palvelimelle, jolloin sitä voidaan käyttää verkkoselaimella.

JS-Vine on julkaistu LGPL-lisenssillä (Eisenberg 2014a). Lisenssi tarjoaa käyttäjälle lähdekoodin vapaan muokkaamisen ja hyödyntämisen omassa julkaisussa (Wikipedia 2014c).

Yhteenvetona voin todeta, että JS-Vine on avoimen lähdekoodin moottori monelle käyttöjärjestelmälle. Se on hyvin pelkistetty työkalu jolla voi koostaa yhteen ääntä, kuvaa ja dialogia sekä käyttäjän valintojen mukaan etenevän dialogipuun. Tiedostotuesta ei ole kerrottu kotisivuilla. JS-Vine on dokumentoitu kattavasti mutta minkäänlaista kehitysyhteisöä ei näytä olevan eikä kotisivuilla ole forumia eikä Lemma Soft Forumilta löytynyt keskusteluja JS-Vineen liittyen. JS-Vinellä tehdyistä julkaisuista ei ole tietoa kotisivuilla. Olettaisin ettei sillä ole tehty ainakaan kaupallisia julkaisuja. JS-Vine soveltuu mielestäni yksinkertaisten visuaalisten novellien sekä esitelmien kehitykseen web-julkaisuiksi. Jos työkalulla halutaan tehdä web-julkaisu kattavilla ominaisuuksilla, kannattaa mieluummin valita Belle tai WebStory Engine.

Japaninkielisiä kehitysympäristöjä

Tässä osiossa on mainittu muutama suosittu, täysin tai osittain japaninkielinen kehitysympäristö. Olen jättänyt yksityiskohtaiset tiedot pois englanninkielisen tiedon puutteen takia.

NScripter / ONScripter / ONScripter-EN / PONScripter

NScripter on Naoki Takahashin kehittämä moottori visuaalisten novellien kehittämiseen Microsoft Windowsilla. NScripter oli suosituimmillaan yksi käytetyimmistä moottoreista Japanissa. Kehitys on ollut suurelta osin pysähdyksissä vuodesta 2003, versiosta 2.59. NScripter on suljetun lähdekoodin ohjelmisto mutta ilmainen kaupalliseen ja ei-kaupalliseen käyttöön. Lisenssi ei ole selvillä. (Wikipedia 2014d.)

NScripter-moottoria on jatkokehitetty eri klooneiksi. Yksi suosituimmista on japanilaisen ohjelmoijan nimimerkillä Opagee kehittämä ONScripter. Se on avoimen lähdekoodin ohjelmisto monelle eri käyttöjärjestelmälle, pohjautuen SDL-kirjastoon. (Wikipedia 2014d.)

ONScripter-EN on englanninkielisen yhteisön oma kehityshaara OnScripteristä. Kehitystyö keskittyy englannin kielen tuen lisäämisen sekä sitä tukevien ominaisuuksien lisäämisen moottoriin. Moottoria on käytetty monien japaninkielisten visuaalisten novellien kääntämiseen englanninkielelle. (Wikipedia 2014d.)

PONScripter on oma kehityshaaransa ONScripter-EN-haarasta. Sen tarkoituksena on viedä kehitystä länsimaisempaan suuntaan lisäämällä mm. natiivituki UTF-8 merkistökoodaukselle sekä länsimaisille fonttityypeille. Tämän takia lähdekoodiin on tehty suuria muutoksia eikä PONScripter ole täysin yhteensopiva muiden versioiden kanssa. (Wikipedia 2014d.)

KiriKiri ( KiriKiri2/KAG3 )

KiriKiri on japanilaisen W.Deen kehittämä moottori visuaalisten novellien kehittämiseen. Sitä käytetään yleensä yhdessä KAG (KiriKiri Adventure Game System) -kehitysympäristön (framework) kanssa. Viimeisintä versiota kutsutaan nimellä KiriKiri2/KAG3. Työkalusta yleisesti puhuttaessa käytetään vain termiä KiriKiri. KiriKiriä pidetään yleisesti Nscripterin modernina korvaajana Japanissa. KiriKiri on julkaistu GNU-lisenssillä Microsoft Windowsille. Vaikkakin lähdekoodi on avointa, ei porttausta muille käyttöjärjestelmillä ole vielä tehty. (Wikipedia 2014e.)

KiriKiri-moottoriin kehitetty skriptikieli, nimeltään TJS, on olio-ohjelmointimallin mukainen ECMAScript-toteutus (implementation). KAG-kehitysympäristöön on kehitetty XML-tyyppinen merkintäkieli. Molempia voidaan laajentaa laajennusten avulla. (Wikipedia 2014e.)

KiriKirillä tehtyjä julkaisuja voidaan käyttää suoraan länsimaisilla käyttöjärjestelmillä, jos julkaisun merkistökoodaus on UNICODE-tyyppiä. Muuten käyttöjärjestelmän paikkatiedot (locale) täytyy asettaa Japaniin tai käyttää Microsoftin AppLocale-työkalua. (Wikipedia 2014e.)

Lopuksi

Työkalua valittaessa kannattaa huomioida seuraavat seikat: halutaanko julkaisusta tehdä paikallinen (standalone) vai web-sovellus, halutaanko työskennellä graafisella kehitysympäristöllä vai itse valitsemilla työkaluilla pelkkää moottoria käyttäen ja lopuksi, halutaanko kehitystyöhön täysin vapaat kädet alkaen lähdekoodin muokkaamismahdollisuudesta vai tehdäänkö kehitystyö ainoastaan työkalun tarjoamilla työkaluilla.

Omat suositukseni edellä mainituille seikoille ovat Ren’Py paikallisille (standalone) julkaisuille. Ren’Py siksi, että sillä on pisin kehityskaari kaikista englanninkielisistä työkaluista. Se tarjoaa laajan työkaluvalikoiman ja täysin vapaat kädet lähdekoodin muokkaamiseen ja omien toiminnallisuuksien lisäämiseen Python-skriptikielellä. Lisäksi se on aktiivisessa kehityksessä ja sillä on aktiivinen kehitysyhteisö.

Web-sovelluksen kehitykseen suosittelen joko Belleä tai WebStory Engineä riippuen siitä, halutaanko käyttää pelkästään graafista ympäristöä, joka ei vaadi lainkaan ohjelmointitaitoa vai pelkkää moottoria mieluisilla työkaluilla täysin vapaasti lähdekoodia muokaten ja omia toiminnallisuuksia kehittäen.

Visual Novel Toolkit on myös hyvä vaihtoehto, jos Unity on jo valmiiksi tuttu työkalu. Unity tarjoaa laajan tuen eri julkaisualustoille aina PC:stä konsoleihin ja mobiilipuolelle. Kannattaa kuitenkin muistaa, että Visual Novel Toolkitin täysi versio on maksullinen ja Unityn ilmaislisenssissä on rajoituksia kaupallisille julkaisuille.

Visuaalisten novellien tietoperusta

Tähän tulee tieto visuaalisissa novelleissa käytetyistä kerronnallisista tyyleistä sekä novelleissa käytetyistä kerronallisista elementeistä.

Kerronnalliset elementit

ADV

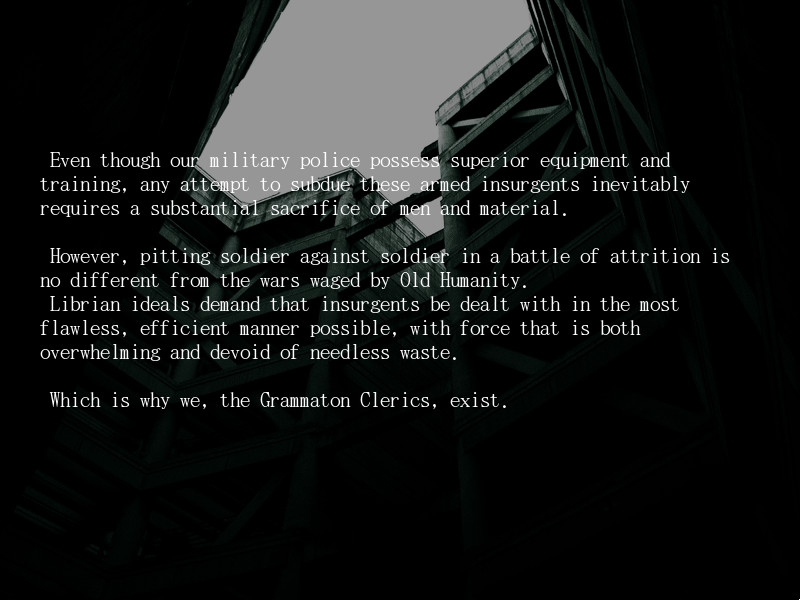
ADV (Adventure style game) on dialoginen tarinan kerrontatyyli, jossa tekstiä näytetään vähän kerrallaan, yleensä riveittäin, kuten kuvassa ” Kyokugen Dasshutsu ADV Zennin Shibou Desu” alla. Tekstirivi voidaan ”juoksuttaa” ruudulle kirjain kerrallaan. Pelaajan ollessa dialogin toinen osapuoli, hänen vastaus vaikuttaa tarinan kulkuun, eli pelaajan vaikutus tarinaan on suuri. Vastausaika voi olla määritelty. Eli pelaajan on valittava jokin vastauksista määrätyn ajan sisällä tai oletusvastaus, esim. vaikeneminen, tulee valituksi. Teksti on yleensä omassa laatikossaan ruudun alalaidassa kuvan alapuolella.

Kyokugen Dasshutsu ADV Zennin Shibou Desu (https://vndb.org/v7809.)



NVL

NVL (Novel style game) on monologinen tarinan kerrontatyyli, jossa tekstiä näytetään sivu kerrallaan. Tekstin on tarkoitus olla kuvailevampaa (verbose) ADV-tyyliin verrattuna. Vuorovaikutus pelaajan kanssa on vähäisempää ja satunnaisempaa; pelaajalta kysytään valintaa tarinan risteyskohdissa. Teksti on yleensä keskellä ruutua, koko ruudun pituudelta, kuten kuvassa ”Jouka no Monshou” alla. Taustakuva näkyy tekstilaatikon sivuilta ja ehkä läpikuultavasti alta.

Jouka no Monshou (https://vndb.org/v1390).

Kinetic novel on täysin monologinen tarinan kerrontatyyli, jossa pelaaja ei voi vaikuttaa tarinan kulkuun.

Lähteet:

AVG ja NVL: <http://lemmasoft.renai.us/forums/viewtopic.php?f=4&t=8064>

<http://tvtropes.org/pmwiki/pmwiki.php/SoYouWantTo/MakeAVisualNovel>

Kinetic novel: <http://en.wikipedia.org/wiki/Visual_novel>

Kustomoidut elementit



Taistelukohtaus Loren The Amazon Princess –visuaalisesta novellista. Novelli on tehty Ren’pyllä.

Lähde: <http://games.renpy.org/game/loren-amazon-princess>



Taistelukohtaus Planet Stronghold –visuaalisesta novellista. Novelli on tehty Ren’pyllä.

Lähde: <http://games.renpy.org/game/planetstronghold>



Taistelukohtaus Sunrider –visuaalisesta novellista. Novelli on tehty Ren’pyllä.

Lähde: <http://www.indiedb.com/games/sunrider/images>



Inventaario Loren The Amazon Princess–visuaalisesta novellista. Tavaroiden hallinnointiin on tehty useita ruutuja.

Lähde: <http://www.winterwolves.com/lorenamazonprincess.htm>



Inventaario Planet Stronghold-visuaalisesta novellista.



Hahmon kehitysikkuna Loren The Amazon Princess –visuaalisesta novellista.



Karttaikkuna Vlad the Impaler-visuaalisesta novellista.

Lähde: <http://store.steampowered.com/app/295850/>

Tarina

Pariskunta saapuu kylään. He majoittuvat majataloon. Yöllä heidän huoneeseen murtautuu ihmisiä. Syntyy taistelu.

LOPPUTULOKSET: 1. TAISTELE (2. PAKENE)

TAISTELE: Pelaaja puolustautuu hyökkääjiä vastaan. Ylivoima on kuitenkin liian suuri ja pelaaja häviää taistelun. Pelaaja herää metsästä mukiloituna, vaatteet repaleisena. Kylä näkyy alempana laaksossa. Ylempänä näkyy vuori ja ympärillä on metsää. Vaimo on kadonnut.

LOPPUTULOKSET: 1. PALAA KYLÄÄN 2. MENE VUORELLE 3. TUTKI METSÄÄ

PALAA KYLÄÄN: Ihmiset pitävät pelaajaa villinä ja hyökkäilevät tämän päälle.

LOPPUTULOKSET: 1. PAKENE METSÄÄN 2. ETSI MAJATALO

PAKENE METSÄÄN: 1. TUTKI METSÄÄ 2. TUTKI VUORTA

ETSI MAJATALO: Pelaaja onnistuu löytämään majatalon vältellen kylän asukkaita. Hän menee sisään ja kohtaa majatalon omistajan.

LOPPUTULOS: DIALOGI 5 (OPTIONAL) (jos ei kerkeä tehdä niin ETSI MAJATALO jätetään pois)

TUTKI METSÄÄ: Pelaaja löytää ihmisjoukon jotka ovat kerääntyneet suuren vanhan tammen ympärille. He näyttävät ihmisiltä mutta käyttäytyvät kuin eläimet. Ihmiset ovat alasti ja tanssivat, kisailevat, parittelevat tai vain oleskelevat. Joku huomaa pelaajan ja hälyttää muut kovalla huudolla. Kaikki alkavat lähestyä pelaajaa varuillaan häntä tarkastellen. Taistelu vaikuttaa turhalta ylivoiman edessä.

LOPPUTULOS: 1. PAKENE 2. TUTUSTU

TUTKI VUORTA: Pelaaja löytää linnan jossa kohtaa vanhan miehen.

LOPPUTULOS: DIALOGI 1 (ei painotusta koska ei vielä sitovia tapahtumia)

PAKENE: Katso kohtaus TAKAA-AJO

LOPPUTULOS: 1. PELAAJA PAKENEE 2. PELAAJA JÄÄ KIINNI

PELAAJA PAKENEE: Pelaaja päätyy vuorella olevalle linnalle. Hän kohtaa linnalla vanhan miehen.

LOPPUTULOS: DIALOGI 1 (painotus taistelulla koska pelaaja pitää metsän ihmisiä uhkaavina)

PELAAJA JÄÄ KIINNI: Ihmiset vievät pelaajan vuorella olevalle linnalle. Linnalla on vanha mies jota ihmiset selkeästi kunnioittavat suuresti.

LOPPUTULOS: DIALOGI 1 (painotus liittymisellä. pelaaja on alistunut kiinnijäämisestä ja lupaa liittyä jos autetaan)

TUTUSTU: Ihmiset kerääntyvät pelaajan ympärille ja tutkivat häntä koskettelemalla ja kommunikoimalla keskenään. (dialogi joka päättää ihmisten suhtautumisen pelaajaan?) He näyttävät luottavan pelaajaan. Ihmiset keräävät oksia ja tekevät suuren nuotion. He aloittavat villin tanssin ja laulun nuotion ympärillä. Pelaaja menee mukaan ja he tanssivat pitkälle yöhön. Pelaaja saavuttaa hurmostilan ja saa pedon vaistot. Hän vaistoaa vaimonsa ja alkaa seuraamaan vaistoa.

LOPPUTULOS: MENE KYLÄÄN PETONA

MENE KYLÄÄN PETONA: Pelaaja onnistuu välttelemään ihmisiä vainunsa avulla mutta hän herättää levottomuutta kotieläimissä. Eläimet alkavat elämöidä ja ihmiset alkavat käydä levottomiksi. Vainu johdattaa pelaajan majatalon takana olevalle vajalle. Pelaaja jyrää oven läpi vajaan hurmoshenkensä johdattamana.

LOPPUTULOS: MURTAUDU VAJAAN

MURTAUDU VAJAAN: Pelaaja löytää vaimon pahoinpideltynä. Murtautumisen äänet hälyttävät majatalon omistajan ja hänen kätyrinsä paikalle.

LOPPUTULOS: 1. TAISTELE 2. PAKENE

TAISTELE: Pedon nopeus ja raivo ajavat miehet pakosalle. Ihmisten levottomuus ja taistelun äänet hälyttävät paikalle lisää ihmisiä.

LOPPUTULOS: 1. JATKA TAISTELUA 2. PAKENE

PAKENE: (kaksi edellistä) Katso kohtaus TAKAA-AJO

LOPPUTULOS: 1. PELAAJA PAKENEE 2. PELAAJA JÄÄ KIINNI

PELAAJA PAKENEE: Pelaaja onnistuu pakenemaan metsään. Hän menee takaisin metsänihmisten luo, jotka johdattavat pelaajan linnalle vanhan miehen luo.

LOPPUTULOS: DIALOGI 1 (painotus liittymisellä koska pelaaja on vielä hurmostilassa)

PELAAJA JÄÄ KIINNI: pelaaja otetaan kiinni mutta vaimo pääsee pakenemaan.

LOPPUTULOS: PELAA VAIMOLLA

JATKA TAISTELUA: Pedon vaistoista huolimatta vastus on liian ylivoimainen ja ihmiset saavat lopulta kukistettua pelaajan. Pedon taltuttaminen kiinnittää ihmisten ja eläinten huomion, joten vaimo pääsee pakenemaan huomattuaan tilanteen epätoivoisuuden.

LOPPUTULOS: PELAA VAIMOLLA

PELAA VAIMOLLA: Nainen seuraa ihmisjoukkiota, jotka vievät pelaajan miehen luo joka vaikuttaa kylän päälliköltä. Naisen katsellessa, kujalle tulee vanha nainen joka huomaa vaimon. Vanha nainen säikähtää vaimoa.

LOPPUTULOS: 1. TAISTELE 2. PAKENE 3. RAUHOITTELE

TAISTELE: Nainen ryntää kohti vanhaa naista, vanha nainen alkaa kirkua ja yrittää paeta. Vanhan naisen kirkuminen hälyttää paikalle miehen vangitsijoita ja he ottavat vaimon kiinni. Miehet vievät naisen kylän päällikön luo miehen seuraksi.

LOPPUTULOS: ROVIO

ROVIO: Nainen ja mies poltetaan roviolla metsän ihmisinä (joita pidetään pahoina henkinä). GAME OVER

PAKENE: Nainen pakenee metsään.

LOPPUTULOS: Nainen pakenee metsään ja kohtaa metsän ihmiset. Metsän ihmiset yhdistävät tämän mieheen ja auttavat häntä. He vievät naisen linnalle vanhan miehen luo.

LOPPUTULOS DIALOGI 2 (painotus liittymisellä koska mies ystävystyi metsän ihmisten kanssa (auttavat jos molemmat liittyvät))

RAUHOITTELE: Vanha nainen hymyilee, menee talon ovelle, avaa sen ja viittoilee vaimoa sisään. Sisällä nainen hoitaa vaimon haavoja ja kertoo kylän ja metsän ihmisten historiaa. Vaimo on hädissään miehestä ja pyytää vanhan naisen apua. Vanha nainen neuvoo menemään metsään ja hakemaan apua vuorella olevasta linnasta, jossa asustelee vanha mies. Hän kuulemma voisi auttaa. Vanha nainen antaa vaimolle symbolin joka auttaa vanhan miehen suostuttelussa. Vanha nainen antaa kartan johon hän piirtää linnan sijainnin. Vaimo lähtee kiiruhtamaan kohti linnaa.

LOPPUTULOS: NAINEN LÄHTEE LINNALLE

NAINEN LÄHTEE LINNALLE: Nainen löytää linnan kartan avulla ja hän tapaa linnalla vanhan miehen.

LOPPUTULOS: DIALOGI 2 (painotus auttamisella koska naisella on vanhan naisen symboli)

DIALOGI 1:

LOPPUTULOS: 1. PELAAJAA AUTETAAN 2. PELAAJA LIITTYY METSÄN IHMISIIN 3. PELAAJA PAKENEE 4. PELAAJA TAISTELEE

DIALOGI 2:

Jos naisella on mukanaan vanhan naisen yrttiriipus, hänen on helpompi suostutella vanha mies avuksi. Vanhalla miehellä ja vanhalla naisella voi olla romanttinen historia ja kaipuu toista kohtaan.

LOPPUTULOS: 1. PELAAJAA AUTETAAN 2. PELAAJA LIITTYY METSÄN IHMISIIN 3. PELAAJA PAKENEE

PELAAJAA AUTETAAN: Vanha mies kutsuu metsän ihmiset kokoon ja yhdessä pelaajan kanssa he suuntaavat kylään pelastamaan vaimoa. Kylään päästyään kylän ihmiset kerääntyvät paikalle kylän päällikön kanssa. Tunnelma on erittäin kireä.

LOPPUTULOS: DIALOGI 3

PELAAJA LIITTYY METSÄN IHMISIIN: Vanha mies saa lumottua pelaajan ja pelaaja liittyy heihin.

LOPPUTULOS: PROTOSSA GAME OVER. Jatkoversiossa kontrolli siirtyy toiseen pelaajahahmoon joka voi yrittää etsiä ja pelastaa toisen.

PELAAJA PAKENEE: Pelaaja pakenee takaisin kylään. Epätoivoissaan, viimeisenä oljenkortenaan hän lupaa näyttää metsänihmisten sijainnin, jos puoliso vapautetaan. Kylän asukkaat suostuvat tähän. Pelaaja puolisoineen johdattaa kyläläiset metsän ihmisten luo.

LOPPUTULOS: DIALOGI 4

PELAAJA TAISTELEE: (OPTIONAL) (jää pois jos ei ehdi tekemään)

LOPPUTULOS: 1. PELAAJA VOITTAA 2. PELAAJA HÄVIÄÄ

PELAAJA VOITTAA: Vanha mies anoo armoa ja lupaa auttaa pelaajaa

LOPPUTULOS: 1. SUOSTU AVUN TARJONTAAN 2. SURMAA VANHA MIES

SUOSTU AVUN TARJONTAAN: Vanha mies kutsuu metsän ihmiset koolle ja he lähtevät pelaajan kanssa kylään. Kylässä kylän ihmiset kerääntyvät kylän päällikön kanssa ympärille. Tunnelma on kireä.

LOPPUTULOS: DIALOGI 3

SURMAA VANHA MIES: Pelaajan surmattua vanhan miehen metsän ihmiset hyökkäävät hänen kimppuunsa.

LOPPUTULOS: TAKAA-AJO

TAKAA-AJO: 1. PELAAJA PAKENEE 2. PELAAJA JÄÄ KIINNI

PELAAJA PAKENEE: Pelaaja pakenee kylään. Kylän ihmiset hyökkäävät hänen kimppuunsa. Metsän ihmiset hyökkäävät kylään pelaajan jäljessä. Syntyy yleinen kahakka josta tulee verilöyly.

LOPPUTULOS: Kaikki kuolee. GAME OVER

PELAAJA JÄÄ KIINNI: Pelaaja viedään vanhalle puulle ja sidotaan puuhun kiinni. He aloittavat jonkinlaisen villin rituaalin. Viimeisillä hengen vedoilla pelaaja tuntee kuinka vanhan miehen henki laskeutuu pelaajaan ja pelaajan tietoisuus kuolee. GAME OVER

DIALOGI 5:

LOPPUTULOS: 1. TOTUUS PALJASTUU

TOTUUS PALJASTUU: Pelaajalle paljastuu, että yölliset hyökkääjät olivat majatalon omistaja kätyreineen. Motiivina oli epäilys pariskunnan olevan metsän ihmisten kätyreitä ja tulleensa tekemään kyläläisille pahaa. Vaimo otettiin kiinni miesten himojen takia, siksi muille kyläläisille ei kerrottu asiasta. Mies oli tarkoitus jättää kuolemaan metsään.

LOPPUTULOS: TAISTELU

TAISTELU: Syntyy hurja taistelu joka hälyttää muita kyläläisiä paikalle. Yhdessä kyläläiset nujertavat miehen. Mies ja vaimo viedään kylän päällikön luo jossa majatalon omistaja syyttää heitä metsän ihmisten kätyreiksi. Koska mies ja vaimo ovat tuntemattomia ja majatalon omistaja luotettava yhteisön jäsen, mies ja nainen päätetään polttaa roviolla.

LOPPUTULOS: ROVIO

ROVIO: Nainen ja mies poltetaan roviolla metsän ihmisinä (joita pidetään pahoina henkinä). GAME OVER

Tekniset elementit

Mitä visuaalisille novelleille ominaisia teknisiä elementtejä on olemassa. Esim. skenen taustakuva, skenen hahmojen kuvat taustakuvan päällä, text box jossa skenen teksti, valinta dialogi, yms.

Kuva X. Lähde: http://www.renpy.org



Kuvassa X näkyvät kaikki visuaalisille novellille ominaiset elementit: taustakuva, pelihahmot, teksti-ikkuna jossa puhuvan hahmon nimi sekä pelimenu teksti-ikkunan alalaidassa. Teksti-ikkunan oikeassa laidassa oleva pelihahmo on ylimääräinen tässä tapauksessa, tällä puhuja on haluttu visualisoida tällä tavalla pelkän nimen sijaan. Peli kuvataan yleensä pelaajan näkökulmasta kuten kuvassa yllä.

Kuva X. Lähde: http://www.renpy.org



Kuvassa X näkyy myös visuaalisille novelleille tyypillinen elementti eli valintadialogi ruudun keskellä. Pelaaja valitsee listasta vastauksen teksti-ikkunassa esitettyyn kysymykseen. Tarina on harvoin lineaarinen vaan se on jaettu useisiin haaroihin. Kyseisen listan vastaukset vievät yleensä eri tarinahaaroihin, jotka mahdollisesti risteävät tarinan aikana. Tarina voi yleensä päättyä usealla tavalla, riippuen pelaajan valinnoista.

Yleensä visuaalisessa novellissa on jonkinlainen valikko, kuten kuvassa X. Menussa on yleensä toiminnot tallennukselle, lataukselle, resetoinnille, pelin lopettamiselle sekä pelivalikolla josta voi tehdä säätöjä peliasetuksiin. Tallennuksen lisäksi käytössä voi olla toimintoja tarinassa navigoimiseen: rollback tallentaa jokaisen kohtauksen jossa pelaaja on käynyt pelin aikana. Rollbackin avulla pelaaja voi palata taaksepäin juonipuussa (story branch) ja jatkaa uutteen suuntaan halutessaan. Fast-forward taas pikakelaa alusta viimeisimpään kohtaukseen jossa pelaaja on käynyt. Tätä käytetään silloin kun halutaan käydä läpi kaikki juonipuun kohtauksen ja näin selvittää tarina kokonaisuudessaan. Fast-forwardin avulla päästää tarinan haaraan jossa on vielä käymättömiä haaroja.

Prototyyppi

Tässä osiossa tarkistellaan prototyypin kehitystä valituilla työkaluilla.

Ren’py

Tällä työkalulla tehdään paikallinen (standalone) versio prototyypistä. Työkalu valittiin sen yleisyyden vuoksi. Se on yleisin länsimaissa käytetty visuaalisten novellien kehitykseen erikoistuneista työkaluista.

Dokumentaatio

Dokumentaatio ei kata kaikkea. Lainaus sivulta http://www.renpy.org/doc/html/ “We're in the process of rewriting Ren'Py's documentation. While what we have here is the most up-to-date documentation, it's also very incomplete.”

Viimeisin dokumentaatio löytyy osoitteesta http://www.renpy.org/doc/html/. Osa Ren’Pystä on dokumentoitu vain vanhempien versioiden osalta. Nämä dokumenaatiot löytyvät osoitteesta http://www.renpy.org/wiki/renpy/Wiki\_Home\_Page.

Verrattuna muihin, tarkasteluni kohteena oleviin kehitysympäristöihin, Ren’Pyn dokumentaatiota voi mielestäni kutsua kattavaksi. Se kattaa kaikki Ren’Pyn osa-alueet kehitysympäristöstä skriptaukseen ja valmiin distribuution julkaisuun (build). Referenssi sisältää kuvauksen kaikista(?) toiminnoista, lyhyen kuvauksen toiminnosta, toiminnon tyypin ja mahdolliset parametrit. Referenssin sisältö on jakautunut uuden ja vanhan dokumentaation välille. Sama toiminto voi esiintyä molemmissa dokumentaatioissa, esim. Displayables. Referenssin heikkous on, ettei se sisällä esimerkkejä. On vain kerrottu mihin tarkoitukseen jokin toiminto on mutta ei ole esimerkkiä kuinka se toteutetaan. Tämä on tunnettu ongelma Ren’py yhteisössä ja monet uudet käyttäjät valittavat kuinka hankalaa Ren’Pyn skriptaukseen on päästä sisään vaikeaselkoisen dokumentaation takia, etenkin jos ei omaa koodauskokemusta. Tässä yksi ketju Lemmasoft forumilta: http://lemmasoft.renai.us/forums/viewtopic.php?f=4&t=17008

Käyttäjät julkaisevat myös tutoriaaleja. Niitä on koottu osoitteeseen http://renpyhandbook.tumblr.com/code-tutorials.

Kehitysyhteisö

* onko forumia
  + jos on, onko se aktiivinen
  + onko paljon postauksia

Ren’Pyllä ei ole omaa forumia mutta Lemmasoft forum toimii sen epävirallisena forumina. Siellä on oma osio Ren’Pylle joka sisältää yli 10 000 eri aihetta.

Tiedon löytäminen netistä

* löytyyko googlettamalla apua (löytääkö ylipäätään muuta kuin omat sivut)

Ren’Pyllä on oma tili mm. Facebookissa, Twitterissä ja DevianArtissa. Lisäksi löytyy mm. videotutoriaaleja Youtubesta.

Ympäristön vaatimukset (esim. http-palvelin, tekstieditori yms)

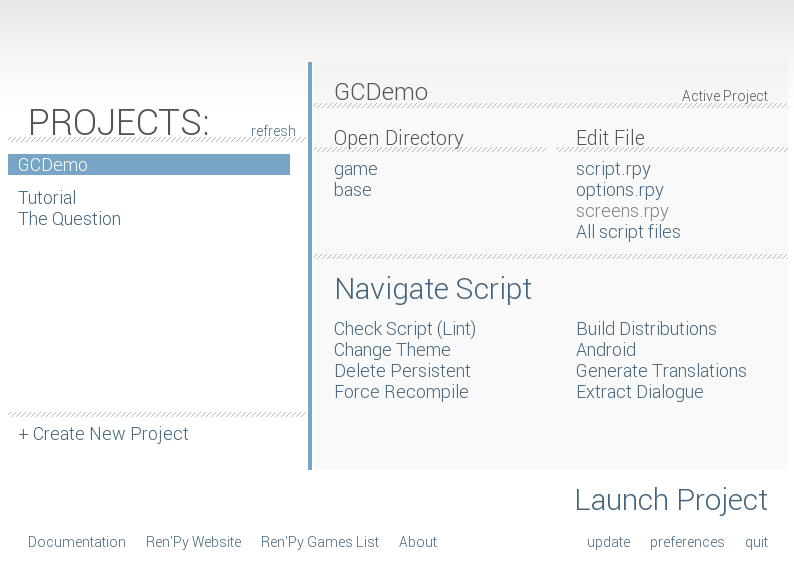
* riittääkö pelkkä ympäristö vai pitääkö viritellä kolmannen osapuolen ohjelmia
  + esim. webstory engine vaatii http-palvelimen jotta tuotosta voi testata

Ren’Pyssä ei ole omaa tekstieditoria. Sen asetuksista voi ladata ja asentaa Editra- tai JEdit-tekstieditorin. Lisäksi voi valita, että järjestelmän oletukseksi asetettu editori avautuu tai ettei avata mitään editoria. Ren’Pyn mukana tulee kaikki kirjastot joista se on riippuvainen, kuten Python. Eli Ren’Py huolehtii itse kolmannen osapuolten liittämisestä siihen. Käyttäjän ei tarvitse määritellä kuin käytettävä tekstieditori jonka liittämisestä Ren’Py myös huolehtii.

Toiminnot

* mitä toimintoja on tarjolla graafisessa ympäristössä
  + sisällön tuottamiseen
  + projektin hallintaan

Kuva X.



Kuvassa X on päänäkymä Ren’Py Launcherista. Launcherin vasemmassa laidassa otsikon ”Projects” alla on listattu kaikki Ren’Pyn juurikansiossa olevat projektit. Uuden projektin voi luoda listan alla olevasta ”+ Create New Project” -linkistä. ”Open Directory” otsikon alta saa avattua projektin game ja base kansiot. Game-kansio sisältää kaikki projektin assetit ja skriptitiedostot. Myös pelitallennukset tulevat tässä kansiossa olevaan ”saves”-kansioon. Base-kansio sisältää kaikki projektin tiedostot, mukaan lukien ”game”-kansion. ”Edit file”-otsikon alta saa avattua Ren’Pyn oletuksena olevat skriptitiedostot (scripts.rpy, screens.rpy ja options.rpy). ”All script files” avaa kaikki projektissa olevat skriptitiedostot. Tiedostot avataan Launcherin ”preferences”-ikkunassa määritellyllä tekstieditorilla (kuvailtu myöhemmin). ”Navigate script”-otsikon alta löytyvät Launcherin varsinaiset työkalut.

Assetit

Lisääminen, poistaminen yms.

Assetit kuten kuva- ja äänitiedostot tulevat projektin game-kansion alle. Ne voidaan laittaa omiin alikansiohin joihin viitattaessa kansio tulee mainita: kuvat/kuva.jpg

Kuva

Kaikki kuvat, niin taustakuvat kuin sen päällä olevat kuvat, määritellään image-elementillä. Eli kaikki kuvat määritellään samalla tavalla.

Kohtaus määritellään scene-elementillä jolle määritellään taustakuva: scene <kuvan nimi>. Kohtaukseen voi lisätä hahmoja (Character) tai muita kuvia show-komennolla: show <kuvan nimi>. Kohtauksen vaihtuessa kuvat häviää. Kuvia voi myös piilottaa kohtauksen aikana hide-komennolla: hide <kuvan nimi>.

Kohtauksesta toiseen voidaan vaihtaa transition-efektiä käyttäen, niitä ovat mm. fade ja dissolve. Transition efektejä käytetään with-komennolla scene-elementin yhteydessä: scene kohtaus with fade aloittaa kohtauksen ”kohtaus”, häivitys-efektin kanssa. Efektejä voidaan käyttää kuville samaan tapaan show-komennon yhteydessä: show kuva with dissolve. Käsky voidaan myös ketjuttaa, eli antaa sama efekti useammalle kuvalle yhdellä käskyllä:   
scene kohtaus  
show kuva  
with dissolve  
Näyttää kohtauksen kohtaus ja kuva kuvan, molemmat dissolve-efektiä käyttäen.

at-komennolla voidaan määritellä mihin kohti ruutua kuva näytetään: show kuva at top näyttää kuvan ruudun yläreunassa.

Ääni

Musiikkia soitetaan play-komennolla jota seuraa heti music-määrite, jonka jälkeen tulee tiedoston nimi: play music ”musiikkitiedosto.ogg”. play-komennolle voidaan antaa parametrejä kuten fadein(kesto), fadeout(kesto) tai loop. Playlist voidaan tehdä samalla komennolla mutta listaa käyttämällä: play music [”eka biisi.ogg”,”toka biisi.ogg”]

Ääniefektit soitetaan muuten samalla tavalla mutta music-käskyn sijaan käytetään sound-määritelmää.  
Ren’py suositellee Oggvorbis (ogg) äänitiedostojen käyttöä.

Video

* videota ei ole protossa tällä hetkellä

Skriptaus

Init-lohkon sisään tulee alustukset. Alustaa voidaan muuttumaton tieto kuten kuvat tai henkilöt (Character).

menu-elementtiä käytetään kun halutaan ohjata tarinaa pelaajan päätösten pohjalta.

menu:  
 ”lause”:  
 jump labelin nimi

menu-elementti aloittaa lohkon, ”lause” näytetään pelaajalle. Jos pelaaja valitsee lauseen, hypätään labeliin nimi.

Pelaajan aiempia valintoja voidaan tallentaa muuttujiin (variables) ja tarkastella niitä tarvittaessa if-lauseilla. Muuttujat on esiteltävä ennen labeliä jossa tarkastelu tapahtuu, aikaisintaan start-labelissä.

Kuten Pythonissa on tapana. Lohkoilla ei ole aaltosulkuja vaan sisennykset määrittelevät lohkon pituuden. Lohkon vaativien elementtien sisältö täytyy siis olla sisennettynä oikealla tasolla. Sisennykseksi tulkitaan neljä perättäistä välilyöntiä. Ren’py ei tue tabulaattoria mutta editoriin voidaan määritellä makro tabulaattorille joka tulostaa neljä välilyöntiä.  
Näitä elementtejä ovat mm. if-ehtolause ja toistolauseet. Lohko alkaa elementin määrittelyn jälkeen tulevalla kaksoispisteellä. Esim. if luku > 6:

Luotettavuus

* tuliko ongelmia ympäristön kanssa kehitystyön aikana

Debuggaus

Debugger-tilan voi avata näppäinkomennolla shift+O. Se on tekstipohjainen tila jossa voi antaa komentoja kuten hypätä (jump) suoraan johonkin labeliin tai seurata (watch) jonkin muutujan arvoa. Debugger-tilan voi sulkea mutta kaikki seuratut kohteet ja niiden arvot jäävät näkyviin.

Käyttökokemus

* omakohtainen fiilis ympäristöstä

WebStory Engine

Tähän osioon tulee sama rakenne kuin osiossa 3.1. Ren’Py.

jotain tekstiä.

Muistiinpanot:

Ei perinteistä if-ehtolausetta käytössä. xml-elementille voi antaa attribuutin ifvar, ifvalue ja ifnot. Näiden avulla voidaan tarkastella onko annetulla muuttujalla haluttua arvoa.

Esim:  
<line s=”n” ifvar=”muuttujanNimi” ifvalue=”haluttuArvo”>  
 näytä tekstiä  
</line>

Esimerkki tulostaa rivin ”näytä tekstiä” jos muuttujan ”muuttujanNimi” arvo on ”haluttuArvo”.

Tämä on ihan toimiva ratkaisu silloin kun tarkastus tehdään yhdelle elementille mutta työläs usemman elementin tarkastelussa. Tämän ongelman voi todennäköisesti poistaa antamalla if-attribuutit ensimmäiselle elementille josta siirrytään omaan sceneen jos arvo toteutuu. Loput elementit sijaitsevat tässä scenessä jolloin niille ei tarvitse antaa if-attribuuttia.

show ja hide asset on ihan perseestä: show asset ei korvaa sillä hetkellä näkyvissä olevaa assettia ja se piirretään saman z-indeksin omaavan kuvapakan alimmaiseksi. Eli edellinen assetti pitää piilottaa hide-komennolla. Hide vaatii aina assetin nimen. Nimeä on välillä tuskallista selvittää jos assetteja näytellään ja piilotellaan flow-controllin avulla ts. pelaajan päätösten mukaisesti. Ja koska pelaajan päätös laukasee heti napin attributtina olevan goto-käskyn, ei piilotusta voi tehdä lähde scenessa vaan kohde scenessä. Jos kohdesceneen tullaan monesta eri lähteestä pitää tehdä useita piilotuksia joista vain käytetään yhtä sillä hetkellä tai sitten pitää tehdä if-rakenne javascriptillä.

Kuvat piirtyvät selainikkunaan oudosti. Ensin kuvasta piirtyy mustalle taustalle ja sen jälkeen vanhan kuvan päälle (jos kyseessä suurempi kuva).

Ei debuggausta. XML-tiedoston voi validoida mutta jos logiikka kusee niin ei voi kuin etsiä ongelma koodin seasta

Valintapainikkeet ovat kiinteän mittaisia. Pitkät teksti eivät näy. Jos napeista tekee pitkän tekstin pituisia niin lyhyet tekstit näyttävät hölmöiltä. Ts. automaagisesti skaalautuvat napit ois hyvä (niin kuin Ren’pyssä on).

Mitä tästä opimme: KÄYTÄ VAIN HYVIN DOKUMENTOITUJA TYÖKALUJA / YMPÄRISTÖJÄ JOILLA ON LAAJA KÄYTTÄJÄKUNTA = PALJON FORUM POSTAUKSIA ONGELMA- JA KÄYTTÖTILANTEISTA.

Teksti ei rivity valintanapissa vaan leveyden ylimenevä osa leikataan pois. Yritin lisätä rivityksen valintanapin lähekoodin Stackoverflow:sta löytämäni vastauksen perusteella: <http://stackoverflow.com/questions/13048499/text-in-newline-in-html-input-type-button>

lisäsin WebStoryEngine.js tiedoston rivin 8486 jälkeen seuraavat rivit:  
currentButton.setAttribute("value", "Carriage&#13;&#10;return&#13;&#10;separators");

currentButton.setAttribute("style", "text-align:center;");  
mutta niillä ei ollut mitään vaikutusta napin tekniseen tai visuaaliseen käytökseen.

Käänsin audiotiedostot selainten vaatimiin mp3 ja ogg tyyppeihin medio.io verkkosivuston työkalulla: <http://media.io/>

Sain border-kuvat aseteltua ruudun reunoihin käyttämällä moottorin omaa, kuville tarkoitettua imagepack-elementtiä, sillä on attribuutit x- ja y-koordinaateille.

Kuvat näytetään kuten muutkin kuvat ”set asset” ja ”show asset” komennoilla. Skriptin tekeminen ei onnistu. Tarkoitus on käyttää Javascriptin onmouseover eventtiä jokaiselle border-kuvaelementille. Eventin tapahtuessa tarkistetaan kulunut aika ja palautetaan true jos event tapahtui määritellyn aikarajan sisällä tai false jos aikaraja ylittyi. Tämä tehdään loopin sisällä jossa arvotaan kolme näytettävää borderia neljästä eli yksi kuvaruudun reuna ei saa kuvaa, hiiri on tarkoitus siirtää kuvattomalle alueelle (jossa on siis kuvaelementti jonka visibility css-arvo on hidden). True arvo vie loopin seuraavalle kierrokselle jos kierroksia on vielä jäljellä. Jos kierrokset ovat täynnä mennään loopistä ulos ja palautetaan onnistuneen scenen nimi johon game flow siirtyy (annetaan parametrina javascript funtiolle). False arvolla palautetaan epäonnistuneen skenen nimi (annetaan myös parametrina).

Tämän funktion toteutus ei siis onnistu koska en saa minkäänlaista Javascript- funktiota toimimaan en edes yksinkertaista onmouseover-esimerkkiä w3school sivustolta: <http://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_onmouseover_dom>

Tämä voi johtua siitä ettei imagepack-elementillä ole onmouseover-eventtiä. Imagepack-elementti on Javascriptillä määritelty XML-elementti. Sillä ei ole varsinaista XML-schemaa vaan määrittely on tehty puhtaasti Javascriptillä. onmouseover-event on taas HTML-elementille määritelty event. Käsittääkseni XML-elementeille ei ole mitään ”default määrittelyjä” kuten HTML-elementeillä, vaan kaikki määrittelyt löytyvät XML-schemasta. Toisin sanoen jos määrittelyä ei löydy schemasta ei elementillä sitä ole. Mutta koska imagepack-elementillä ei schemaa ole, en tiedä miten määrittely pitäisi tehdä. Ainoa vaihtoehto voi olla kyseinen Javascript-tiedosto (WebStoryEngine.js) jossa on imagepack-elementin määrittely ja XML-porttaus (XML-porttaus on tehty JavaScripitin module patternilla), mutta jälleen kerran en tiedä kuinka edes aloittaa. Käsittääkseni Javascriptillä ei ole muita eventtejä kuin kyseiset HTML-eventit.  
Eli yhteenvetona tuntuu ettei Javascript-eventtiä saa toimimaan XML-elementissä ilman schemaa (jota ei ole). Eli eventtiä ei saa käyttöön kuin ehkä jollain Javascript-virityksellä mistä minulla ei ole mitään käsitystä. Hiiren ja kuvaelementin tarkkailu on välttämätöntä skriptin toteutuksessa (onko hiiri kuvaelementin päällä). En keksi muuta tapaa (ilman kohtuuttomia virityksiä) tämän toteutukseen kuin eventit ja se ei toimi kuten edellä mainitsin, joten jätän skriptin tekemättä.  
Huomiona vielä, että yritin suorittaa skriptiä moottorin line-elementin tagien sisällä, perustuen seuraavaan moottorin kehittäjän forum vastaukseen: <https://iiyo.org/f/discussion/55/html-asset> Kehittäjä sanoo, että line-elementille voi antaa Javascript-koodia, esimerkki:

// imagepack esittely

<imagepack name="borderTop" x="0px" y="0px">

<image name="imgBorderTop" src="assets/images/borders/top2.png" />

</imagepack>

// textbox esittely

<textbox cssid="tbChase" name="tb\_chase" width="780px" height="100px" x="10px" y="490px" namebox="no" speed="0">

<nameTemplate>{name}</nameTemplate>

</textbox>

// character esittely

<character name="c" textbox="tb\_chase">

<displayName></displayName>

</character>

// line jossa käytetään määriteltyä characteriä joka käyttää määriteltyä textboxia (määritelty näkymättöksi default.css)

// tarkoitus siirtää kuvaa onmouseover-eventillä antamalla arvo imagepackin x-attribuutille (kokeilin myös muita kuten em. w3school esimerkkiä)

<line s="c">

<script>

document.getElementById("borderTop").onmouseover = function() { moveImage() };

function moveImage()

{

document.getElementById("borderTop").x = "100px";

}

</script>

</line>

## Visual Novel Toolkit

Tähän osioon tulee perustelut miksi hylkäsin tämän työkalun ja vaihdoin Noveltyyn.

Muistiinpanot:

Kehittäjät eivät ole vastanneet forumviesteihin vuoden 2014 maaliskuun 18.päivän jälkeen (http://forum.unity3d.com/threads/visual-novel-toolkit.217122 )

Teksti ei rivity automaattisesti text boxin sisään vaan piirtyy sen yli.  
Rivin vaihto on määritelty ylimenevälle osalle scriptitiedostossa VinoTextBox.cs rivillä 105. Se ei kuitenkaan toimi jostain syystä. Yksi syy voi olla, pituus luetaan counter-muuttujaan jota tarkastellaan wrapEvery-muuttujaan. Oletettavasti wrapEvery on rivin maksimipituus. Tämä saattaa olla pidempi mitä varsinainen tekstialueen visuaalinen kuva on. Eli mitta ei ylity vaikka se meneekin kuvan yli. Maksimimittaa ei voi tarkastella koska se sisältyy VinoToolkit-kirjastoon enkä löydä kyseistä kirjastoa mistään. Oletettavasti siitä ei ole lähdekoodia saatavilla.  
Tämä johtaa siihen, että rivinvaihto on tehtävä käsin. Tämä taas johtaa ongelmiin eri resoluutioilla.  
Visual Novel Toolkit (vino) lisää oman valikon Unityn GameObject-valikkoon. Valikosta saa lisäyttyä TextBoxin ja sitä tarkastellessa Inspectorissa siellä voi antaa arvon wrapEvery muuttajalle samoin löytyy Wrap Text boolean, johon laittaessa rastin teksti alkaa rivittymään wrapEvey muuttujaan määritellyllä rivin pituudella.

User manual (pdf) on vanhentunut. Sellaisia valikoita joita kuvakaappauksissa näytetään ei enää ole. Esim. ViNo-valikon pitäisi olla Window-valikon vieressä päävalikkona. Ellei se ole vain maksulisessa versiossa?

Yritin liittää kahta sceneä yhteen user manualin ohjeiden mukaisesti. Loin ensin scene1den, lisäsin siihen dialogin. Teksti näkyi oikein kun testasin. Lisäsin dialogiin LoadScene-objektin. Loin toisen scenen ViNoToolbarin scenario-välilehdellä. Lisäsin myös sille dialogin ja raahasin scenen resources-kohtaan. Tuhosin luodon scene hierarkia-ikkunasta. Lisäsin scene2:den scene1:den LoadScene-komponenttiin. Ohjeen mukaan nyt kahden scenen pitäisi olla linkitetty mutta kun painoin playta ei näkynyt kuin musta ruutu. Poistin elementtejä takaperin edellä mainitusta järjestyksestä ja testasin skeneä joka askeleen jälkeen mutta ruutu pysyi mustana. Toistin koko homman muutamasti mutta aina sama laulu. Teksti lakkaa näkymästä kun lisään toisen skenen. Ei auta vaikkaa poista ruksin kohdasta ”Play at start” tai jos poistan kaikki lapsiobjektit.

Ongelmaan ei löydy apua netistä.

Suosittelen Unity-lisäriksi mieluummin LDC (localiced dialogs & cutscenes) pakettia. Sillä onnistuu mm. visuaalisten novellien teko: <http://forum.unity3d.com/threads/ldc-localized-dialogs-cutscenes-powerful-and-intuitive-dialog-system-with-tons-of-features.157128/>

Lopetan tämän kanssa pelleilyn koska tuote on selkeästi susi. Eikä kehittäjätkään tunnu enää kehittävän sitä, ainakaan eivät ole enää vastailleet foorumille.

Vaihdan Noveltyyn koska sillä onnistuu myös visuaalisen novellin kehittäminen puhtaasti graafisesti (ilman koodausta). Siinä on myös tuki skriptikielellä (AngelScrit) joten lisäpalikan teko myös siihen onnistuu.

Mielenkiintoinen vaihtoehto on Belle. Se on täysin graafinen työkalu eikä siinä käsittääkseni ole skriptaustukea. Se tukee myös linuxia ja maciä toisin kuin Novelty. Kehittäjän viimeisin versiojulkaisu on 2014 syyskuulta toisin kuin Noveltyn vuodelta 2010. Toisaalta Bellellä ei ole forumia ja tuki on vielä olematonta toisin kuin Noveltyllä on omalla forumilla jo tuhansia viestejä.

## Novelty

Tähän osioon tulee sama rakenne kuin osiossa 3.1. Ren’Py.

Muistiinpanot:

Fontti pitää olla bitmap-muodossa (.fnt). Valmiit fontit ovat suuria joten yritin importata uuden pienemmän. Aina tuli virheilmoitus ”Unable to load font”. Forumilla oli viestejä samasta virheestä mutta ketkä yrittivät auttaa eivät onnistuneet saamaan virhettä aikaan eli heillä se toimi moitteetta. Huomasin kokeilujen jälkeen, että virhe johtuu liian pitkästä tiedostopolusta. Nyt ei tule virhettä mutta fontti ei silti ilmesty asset-hakemistoon, johon sen pitäisi ilmestyä. Tiedostot ilmestyvät kuitenkin kohdekansioon eli ongelma on saada näkyviin ne työkalun asset-näkymään.

Kerran fonttien importtaus onnistui, tein kaiken niin kuin aiemmin mutta fontti ilmestyä asset kirjastoon. Vähän ajan päästä Novelty kaatui. Tein uuden fontin Noveltyn mukana tulevalla fonttieditorilla mutta sen importtaus ei onnitunut vaan tulee sama virheilmoitus. Kun tein uuden fontin Littera nettityökalulla importtaus onnistuu, tosin tulee nimeämisvirhe.

kuvien feidaus tosi helppo tehdä action windowissa, oikeastaan tarvitsee vain valita feidaako sisään vai ulos, muut arvot ovat yleensä hyvät jo oletuksena.

tekstilaatikon ja kuvan sijainnin ja koon saa asetettua helposti property inspectorissa.

hahmoon liittyvä dialogi pitää esittää hahmon kautta. Hahmo on kuva jolle määritellään nimi ja häneen liittyvän dialogin väri. Ts. ilman kuvaa ei myöskään dialogia voi värittää. Tämä on ongelma kun ei käytetä hahmojen kuvia.

äänien importtaus ja lisäys sceneen todella helppoa ja nopeaa.

valintamenut ovat kiinteitä grafiikoita eli eivät skaalaudu sisällön mukaan.

uuden graafisen assetin teko on helppoa Desingerillä. Esim. uuden menun tekeminen muokkaamalla vanhaa, muuttaen sen kokoa vie vain hetken.

Ympäristön tuki skriptaukselle on aika heikko. Tekstieditori on pelkkä notepad tyylinen, ei värikoodausta tai mitään. Rivinvaihto vie aina rivin alkuun mikä on ärsyttävää. Skriptille ei ole minkäänlaista virheidentarkastusta eli ympäristö ei ilmoita jos koodissa on virheitä vaan kaikki selvitystyö jää käyttäjän harteille.

Ei ole omaa forumia. Kehittäjän sähköpostiosoite on sivuilla josta apua voi kysyä tai Lemmesoft forumilta mutta siellä on vain yksi Visual Novelty thread. Työkalun käyttäjiä kuitenkin forumilla on.

Kääntäjä puuttuu. Kun ohjelman suorittaa, Novelty vain ilmoittaa, että jossakin kooditiedostossa on virhe ja se jätetään suorittamatta. Jos kooditiedostoja on useita, virheellinen kooditiedosto löytyy vain kokeilemalla. Sama pätee virheen etsimiseen itse tiedostosta.

Projektista ei saa poistettua sinne importattuja skriptitiedostoja. Kun tiedoston importtaa Asset librarystä tulee ilmoitus, että kyseinen asset on lisätty projektiin mutta se ei tulle minnekään näkyviin. Noveltyssä on scene-lista jossa näkyy kaikki skenessä olevet assetit mutta jostain syystä skriptiassetit eivät ilmesty siihen. Tämä on ongelma jos importatussa skriptissä on virhe, silloin tulee aina virheilmoitus ohjelmaa suoritettaessa ja tämä vaikeuttaa virheiden jäljitystä entisestään uusien skriptien kohdalla. Ratkaisu ongelmaan tuntuu olevan uuden projektin luonti ja kaikkien elementtien kopiointi siihen, yksi kerrallaan.

Noveltyssä ei tunnu olevan funktiota ajan mittaamiseen. AngelScript käyttää Windowsin (http://msdn.microsoft.com/en-us/library/windows/desktop/dd757629%28v=vs.85%29.aspx) oman kirjaston timeGetTime-funktiota mutta koska Novelty on suljetun lähdekoodin ohjelmisto en voi tarkistaa onko timeGeTime:n tarvitsemat kirjastot käytössä. Funktio ei ainakaan toimi suoraan, eikä headereiden ( Windows.h Mmsystem.h ) includauskaan auta.

Toteutin aikatoiminnon Noveltyn Wait-actionilla. Jokaisella border kuvaelementillä on OnMouseEnter event script joka laittaa lipun päälle jos hiiri tulee kuvaelementin alueelle. Tein branching virityksen Noveltyn Branch-actionilla. Event lipun true arvo vie success skeneen ja false vie false skeneen.   
Tein aluksi lipun kytkemisen true arvoon siten että jokaisella border kuvaelementillä on oma lippunsa joka kytketään päälle jos kyseinen kuvaelementti on hiiren kohde-elementti eli hiiri on vietävä kyseisen elementin päälle. Eli jos vasen border kuvaelementti on kohde-elementti, sen lippu kytkeytyy päälle ja kun hiiri viedään elementin alueelle event script ajetaan. Skriptissä katsotaan onko elementtin lippu päällä jos on niin event lippu kytketään päälle. Näin teoriassa jos vasen elementti on kohde-elementti, event lippu ei mene päälle jos hiiri viedään oikean elementin päälle. kaikki kohde elementtien liput asetaan false tilaan heti skriptin alussa joka. Kohde elementin lippu asetaan true tilaan kun näytettävät border kuvaelementit ovat arvottu, näin varmistetaan, että vain yksi kohde-lippu on päällä kerrallaan.  
Kuitenkin kun testasin toteutusta niin event lippu meni päälle vaikka hiiri oli ei-kohde-elementin päällä. Koska Noveltyssä ei ole debuggeriä en voi varmaksi todeta, että vain halutun elementin kohdelippu on päällä ja meneekö event lippu päälle väärästä paikakasta vai onko kyse jostain muusta.  
Mutta skripti suoretaan uudelleen vain jos event lipun arvo on true ja näyttökertoja on vähemmän kuin annettu arvo, tämä on määritelty em. branch-actionin ehtoina.  
Muutin toteutusta niin, että tein oman skriptitiedoston jokaiselle kuvaelementille. Skriptissä kytketään vain event-lippu päälle OnMouseEnter-eventillä. Tämä muutoksen takia jouduin tekemään oman branch-actionin jokaiselle eventille. Nyt true arvo vie success skeneen ja false arvo siirtyy seuraavaan branch-actioniin. Jokainen branch siis tarkistaa onko tietyn eventin lippu true.  
Kun testasin toteutusta niin lopputulos oli sama kuin edellisellä. Tämä kielisi, että ongelma on jossain muualla. Mutta ilman debuggausta ongelmaa on todella vaikeaa jäljittää. Annan asian olla tässä vaiheessa.

Bugi Noveltyssä: branching ja goto arvot menivät sekaisin ilman syytä joissakin skeneissä. Eli branch vei väärään skeneen true tai false arvolla.

Action-windowisa lukee silloin eri arvo mikä property inspectorissa on. Inspectrin arvoa käytetään.

Vertailu / keskustelu

* tähän varsinainen vertailu, ellei sitä tehdä samantien kehitysympäristöjen osioissa.
* myös jotain yleistä keskustelua kohdasta ko. kehitysympäristöistä.

Kiinnostavia kehitteillä olevia kehitysympäristöjä

* tänne potenttiaalisia kehitysympäristöjä jotka eivät olleet vielä tarpeeksi valmiita tätä kirjoittaessa

Embeddable Visual Novel Engine, EVEN:

http://lemmasoft.renai.us/forums/viewtopic.php?f=57&t=25377

LDC (Localized Dialogs & Cutscenes):

http://forum.unity3d.com/threads/ldc-localized-dialogs-cutscenes-powerful-and-intuitive-dialog-system-with-tons-of-features.157128/

**Lähteet**

Asset Store. 2014a. Visual Novel Toolkit Free. Unity. https://www.assetstore.unity3d.com/en/#!/content/9416. 10.10.2014.

Asset Store. 2014b. Visual Novel Toolkit. Unity. https://www.assetstore.unity3d.com/en/#!/content/9270. 10.10.2014.

Belle. 2014a. What is Belle? http://www.nongnu.org/belle/about.html. 25.9.2014.

Belle. 2014b. Overview. http://www.nongnu.org/belle/docs/overview.html. 8.10.2014.

Belle. 2014c. Actions. http://www.nongnu.org/belle/docs/actions.html. 8.10.2014.

Benerdal, E. 2014a. About Novelty. http://www.visualnovelty.com/docs/novelty\_intro.html. 13.10.2014.

Benerdal, E. 2014b. [Twitter: @NoveltyDev](http://www.visualnovelty.com/forum/viewtopic.php?f=7&t=292). http://www.visualnovelty.com/forum/viewtopic.php?f=7&t=292. 13.10.2014.

Carlos, P. 2013a. CREDITS.txt. GitHub https://github.com/fr33mind/Belle/commit/e709d75e20548174f73f71fbc5eb35759613b528. 25.9.2014.

Carlos, P. 2013b. editor/ChangeLog.txt. GitHub https://github.com/fr33mind/Belle/commit/e709d75e20548174f73f71fbc5eb35759613b528. 25.9.2014.

Carlos, P. 2013c. engine/ChangeLog.txt. GitHub https://github.com/fr33mind/Belle/commit/e709d75e20548174f73f71fbc5eb35759613b528. 25.9.2014.

Eisenberg, J. 2014a. JS-ViNE: A JavaScript Visual Novel Engine. langintro.com. http://langintro.com/js-vine/. 1.10.2014.

Eisenberg, J. 2014b. jdeisenberg. GitHub. https://github.com/jdeisenberg. 1.10.2014.

Eisenberg, J. 2014c. js-vine. GitHub. https://github.com/jdeisenberg/js-vine/commits/master. 14.10.2014.

Eisenberg, J. 2014d. js-vine.js. http://langintro.com/js-vine/js-vine.js. 14.10.2014.

Eisenberg, J. 2014e. JavaScript Visual Novel Engine. http://langintro.com/js-vine/docs/index.html. 14.10.2014.

iiYO. 2014. WebStory Engine Support. iiYO. https://iiyo.org/f/categories/webstory-engine-support. 9.10.2024.

Lemma Soft Forums. 2014. [WebStory Engine - A Visual Novel Engine for the Web](http://lemmasoft.renai.us/forums/viewtopic.php?f=57&t=16722&hilit=webstory+engine). http://lemmasoft.renai.us/forums/viewtopic.php?f=57&t=16722&hilit=webstory+engine. 16.10.2014.

Novelty. 2014a. About Novelty. http://www.visualnovelty.com/docs/novelty\_intro.html. 13.10.2014.

Novelty. 2014b. Using the Novelty editor. http://www.visualnovelty.com/docs/nov\_overview.html. 13.10.2014.

Novelty. 2014c. Novelty scripting reference. http://www.visualnovelty.com/docs/script\_index.html. 13.10.2014.

Novelty. 2014d. Expo. http://www.visualnovelty.com/forum/viewforum.php?f=8&sid=10fb7a5a99f7909e135866f6b5d198c9. 13.10.2014.

Novelty. 2014e. Download Novelty. http://www.visualnovelty.com/download.html. 13.10.2014.

Ren’Py. 2014a. The Ren’Py Launcher. http://www.renpy.org/doc/html/quickstart.html#the-ren-py-launcher. 15.10.2014.

Ren’Py. 2014b. Language Basics. http://www.renpy.org/doc/html/language\_basics.html. 16.10.2014.

Ren’Py. 2014c. Screens and Screen Language. http://www.renpy.org/doc/html/screens.html. 16.10.2014.

Ren’Py. 2014d. License. http://www.renpy.org/doc/html/license.html. 1.10.2014.

Ren’Py. 2014e. Ren’Py Games List. http://games.renpy.org. 16.10.2014.

Ren’Py. 2014f. Why Ren’Py? http://www.renpy.org/why.html. 24.9.2014.

Ren’Py. 2014g. What is Ren’Py. http://www.renpy.org. 16.10.2014.

Sol-tribe. 2014a. Release notes. Sol-tribe. http://www.sol-tribe.net/vino/vino\_j.html. 10.10.2014.

Sol-tribe 2014b. User Guide. Sol-tribe. http://www.sol-tribe.net/vino/pdf/UserGuide%28English%29.pdf. 13.10.2014.

Sol-tribe 2014c. Download. http://www.sol-tribe.net/vino/vino\_j.html. 16.10.2014.

Steinbeck, J. 2012a. First commit. GitHub. https://github.com/jsteinbeck/WebStory-Engine/commit/b6467f7b6218f27e4a894013ee379398d027abfa. 9.10.2014.

Steinbeck, J. 2014b. Merge pull request [#21](https://github.com/iiyo/WebStory-Engine/pull/21) [from lolbot-iichan/develop](https://github.com/jsteinbeck/WebStory-Engine/commit/fda45f4f5708d2f51b5e17caf125ebd7937b54a3). GitHub. https://github.com/jsteinbeck/WebStory-Engine/commit/fda45f4f5708d2f51b5e17caf125ebd7937b54a3. 9.10.2014.

Unity. 2014a. Effortlessly unleash your game on the world’s hottest platforms. Unity. http://unity3d.com/unity/multiplatform. 13.10.2014.

Unity. 2014b. Unity Software License Agreement 4.x. Unity. http://unity3d.com/legal/eula. 13.10.2014.

Unity. 2014c. The market-leading import pipeline. http://unity3d.com/unity/workflow/asset-workflow. 16.10.2014.

Unity. 2014d. SEARCH. http://unity3d.com/search?gq=sol-tribe. 16.10.2014.

Wikipedia. 2014a. Visual Novel. Wikipedia. http://en.wikipedia.org/wiki/Visual\_novel. 24.9.2014.

Wikipedia. 2014b. Ren’py. Wikipedia. http://en.wikipedia.org/wiki/Ren%27Py. 24.9.2014.

Wikipedia. 2014c. GNU Lesser General Public License. Wikipedia. http://en.wikipedia.org/wiki/GNU\_Lesser\_General\_Public\_License. 14.10.2014.

Wikipedia. 2014d. NScripter. Wikipedia. http://en.wikipedia.org/wiki/NScripter. 19.10.2014.

Wikipedia.2014e. KiriKiri. Wikipedia. http://en.wikipedia.org/wiki/KiriKiri. 19.10.2014.

WebStory Engine. 2014a. WSE Next Generation. http://webstoryengine.org/wse-next. 9.10.2014.

WebStory Engine. 2014b. Ideas For Features And Improvements. http://webstoryengine.org/ideas. 9.10.2014.

WebStory Engine. 2014c. WebStory Language Reference. http://webstoryengine.org/language. 9.10.2014.

WebStory Engine. 2014d. game.xml. http://webstoryengine.org/webstories/introduction/game.xml. 9.10.2014.

WebStory Engine. 2014e. Visual Novels Made With WebStory Engine. http://webstoryengine.org/showcase:visual-novels. 9.10.2014.

WebStory Engine. 2014f. WebStory Engine. http://webstoryengine.org/start. 25.9.2014.

WebStory Engine 2014g. [WebStory Engine - A Visual Novel Engine for the Web](http://lemmasoft.renai.us/forums/viewtopic.php?f=57&t=16722&sid=4e140f6e79fd3a19d04e5f86ccee618f). Lemma Soft. http://lemmasoft.renai.us/forums/viewtopic.php?f=57&t=16722. 7.10.2014.