Модерно Банково Управление

(ABM)

Инфраструктурен модел

Версия <1.2>

История

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Версия** | **Описание** | **Автор** |
| 05.01.2016 | 1.0 | Създаване на документа | Мартин Абрашев |
| 07.01.2016 | 1.1 | Промяна на информацията и добавяне на диаграмите | Мартин Абрашев |
| 09.01.2016 | 1.2 | Обновяване на диаграмите | Мартин Абрашев |

Съдържание

1. Представяне 4

1.1 Цел 4

1.2 Обхват 4

1.3 Дефиниции, акроними и абревиатури 4

1.4 Връзки 4

2. Среда за разработка 4

2.1 Хардуер 5

2.2 Софтуер 5

2.2.1 Среда за разработка е Eclipse. 5

2.2.2 Система за контрол на кода 5

2.2.3 Maven 5

2.2.4 JBoss application server 5

2.3 Мрежово местоположение 5

2.4 Профили и пароли 5

3. Управление на проекта 5

3.1 Хардуер 5

3.2 Софтуер 6

3.3 Мрежово местоположение 7

3.4 Профили и пароли 7

4. Продукционна среда 8

4.1 Хардуер 8

4.2 Софтуер 8

4.3 Мрежово местоположение 8

4.4 Профили и пароли 8

5. Среда за тестване 9

5.1 Хардуер 9

5.2 Софтуер 9

Модел на инфраструктурата

# Въведение

## Цел

Целта на документа е да опише използваните по време на разработка хардуер и софтуер. Разглежда компютрите, на които ще се разработва системата, и софтуерните инструменти, с помощта на които ще се изпълнява разработката и комуникацията в екипа. Предмет на този документ е също и продукционната среда, за която също ще бъдат описани машините, инфраструктурата на комуникацията, сърварите и софтуера, който ще бъде инсталиран.

## Обхват

Обхвата на документа включва инфраструктурата на:

Средата за разработка – машините и софтуерните инструменти необходими при разработването  
Управление на проекта – машините и софтуерните инструменти необходими за управление на проекта

Продукционната среда – машините и софтуера инсталиран на системата на която е публикуван готовия продукт.

## Дефиниции, акроними и абревиатури

Дефиниции, акроними и абревиатури могат да бъдат намерени в следния документ: ABM-4-I1-Glossary.doc

## Препратки

Rational Unified Process Version 2003

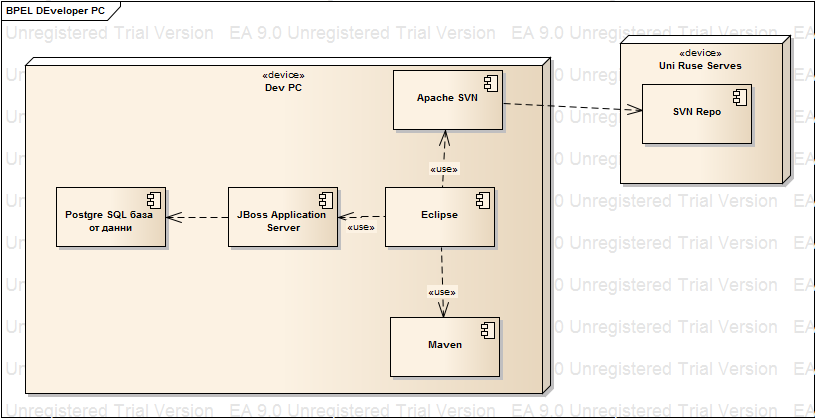
<http://bg.wikipedia.org/wiki/JIRA>

# Изисквания към инфраструктурния модел

# Среда за разработка

В този раздел, ще бъдат описани средата, различните инструменти и сървъри използвани за разработка.

Всички инструменти използвани за разработката са с отворен код.



Фигура1 - Инфраструктура на средата за разработка

## Хардуер

Хардуера използван в процеса на разработката са персоналните компютри на всеки член на екипа и ресурсите на предоставените от университета сървъри, на които са инсталирани средите за управление на разработката – jBoss, SVN, Maven.

## Софтуер

### Среда за разработка е Eclipse.

Всеки член на екипа разполага с дистрибуция на Eclipse Kepler, чрез която извършва разработката на зададените му задачи. Към Eclipse Kepler са инсталирани следните plug-in-и с цел улесняване на работата:

* Subclipse – Apache Subversion (виж точка 2.2.2) клиент в Eclipse;
* m2е – Plug-in за използването на Maven (виж точка 2.2.3) в Eclipse;
* JBoss – инструменти за улесняване на работата с Application server JBoss (виж точка 2.2.4).

### Система за контрол на кода

Използва се главно за поддържане на настоящи и минали версии на файлове за изходен код, уеб страници и документи.

За използването на хранилището за версии, всички членове на екипа разполагат с Apache Subversion (SVN) plug-in инсталиран на средата за разработка инсталирана на персоналните им компютри. При написването на нова функционалност потребителят качва написаното от него на сървъра, давайки възможност няколко човека от екипа да работят едновременно по проекта. Системата позволява връщане към предишни версии, както и маркирането на стабилни версии (tags) и разделянето на функционалности (branches). Екипът работи в сървърната папка trunk.

При запазване на нова версия на файл, SVN дава възможност за описване на качените промени, което е полезно за управлението на разработката и връщането на проекта към позната стабилно работеща версия.

### Maven

Maven е инструмент за автоматизирано билдване на проект. Използват се XML файлове за описание на зависимостите между отделните външни модули и компоненти, както и начина по който се билдват.

Maven проектите се конфигурират посредством Project Object Model, който се намира във файл с име: pom.xml

### JBoss application server

Приложният сървър (Application Server, AS) е софтуерен компонент, който предлага генерализиран подход за реализация на технология клиент-сървър, без да взима в предвид какви са точните функции на сървърното приложение.

JBoss поддръжка прозрачна връзка с база от данни от една страна и връзка с уеб клиент от друга. JBoss с идентична конфигурация е инсталиран на компютрите на членовете на екипа.

## Мрежово местоположение

Ще бъдат предоставени от университета, но на този етап са още неизвестни.

## Профили и пароли

SVN системата работи с акаунти, които всеки член на екипа е създал индивидуално и предоставил на мениджъра на екипа, за да ги добави към проекта.

# Управление на проекта

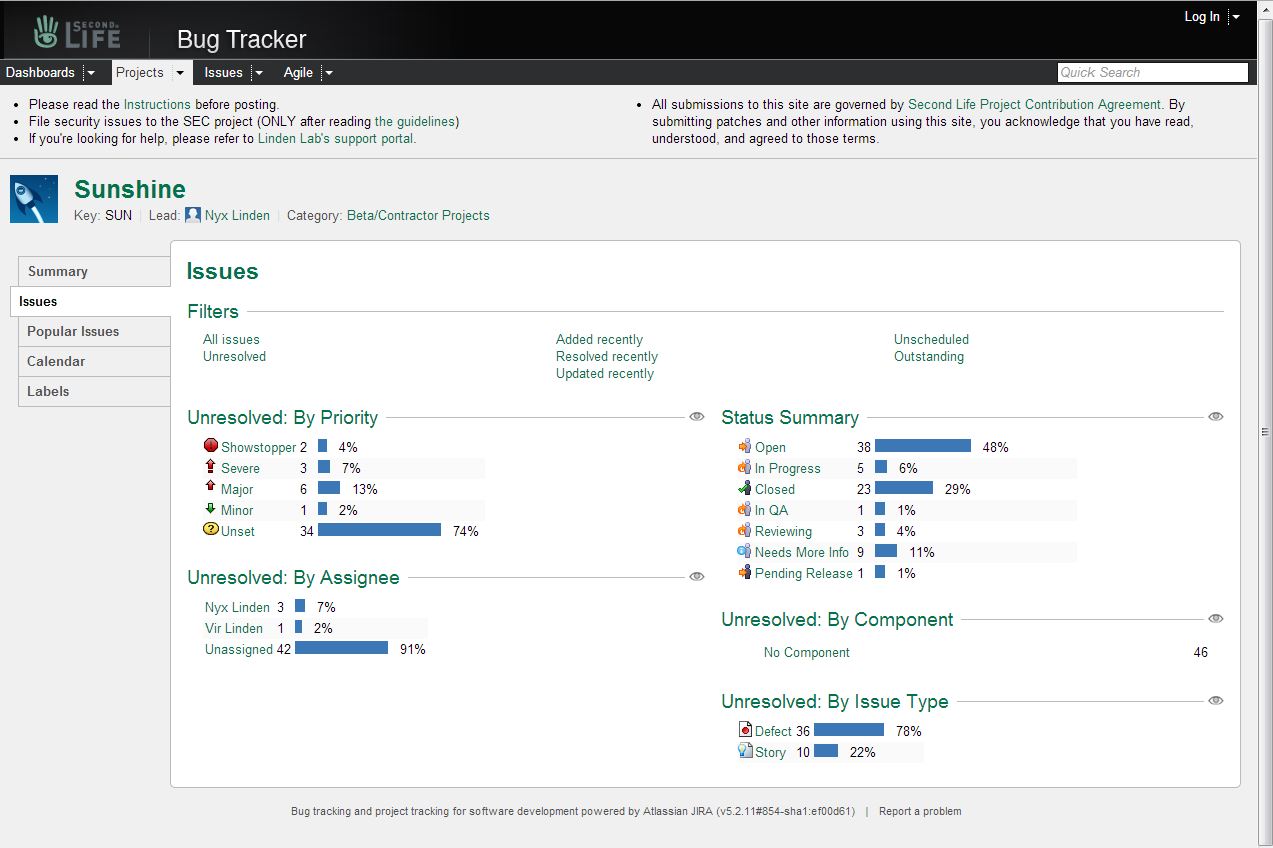
## Хардуер

Хардуерът ползван при анализа и дизайна, както и при разработката са персоналните компютри на всеки член на екипа. Инструментите, необходими за работа по време на фазите са инсталирани на всеки компютър.

## Софтуер

Софтуерът, който се използва за управление на проекта е онлайн свободната версия на системата JIRA.

JIRA представлява уеб-базирана система за проследяване на грешки (bug tracking), проблеми (issue tracking) и управление на разработката на софтуерни проекти. Управлението на технологичния процес (workflow) прави JIRA подходяща за управление и подобряване на процеса на работа в една организация или екип.



Фигура2 - Екран от публичен проект управляван в JIRA

Всеки член на екипа има собствен профил за JIRA, което е и минималното изискване за ползването на системата. Всеки член на екипа е отговорен да преглежда, обновява и решава различните задачи, които са възложени на него.

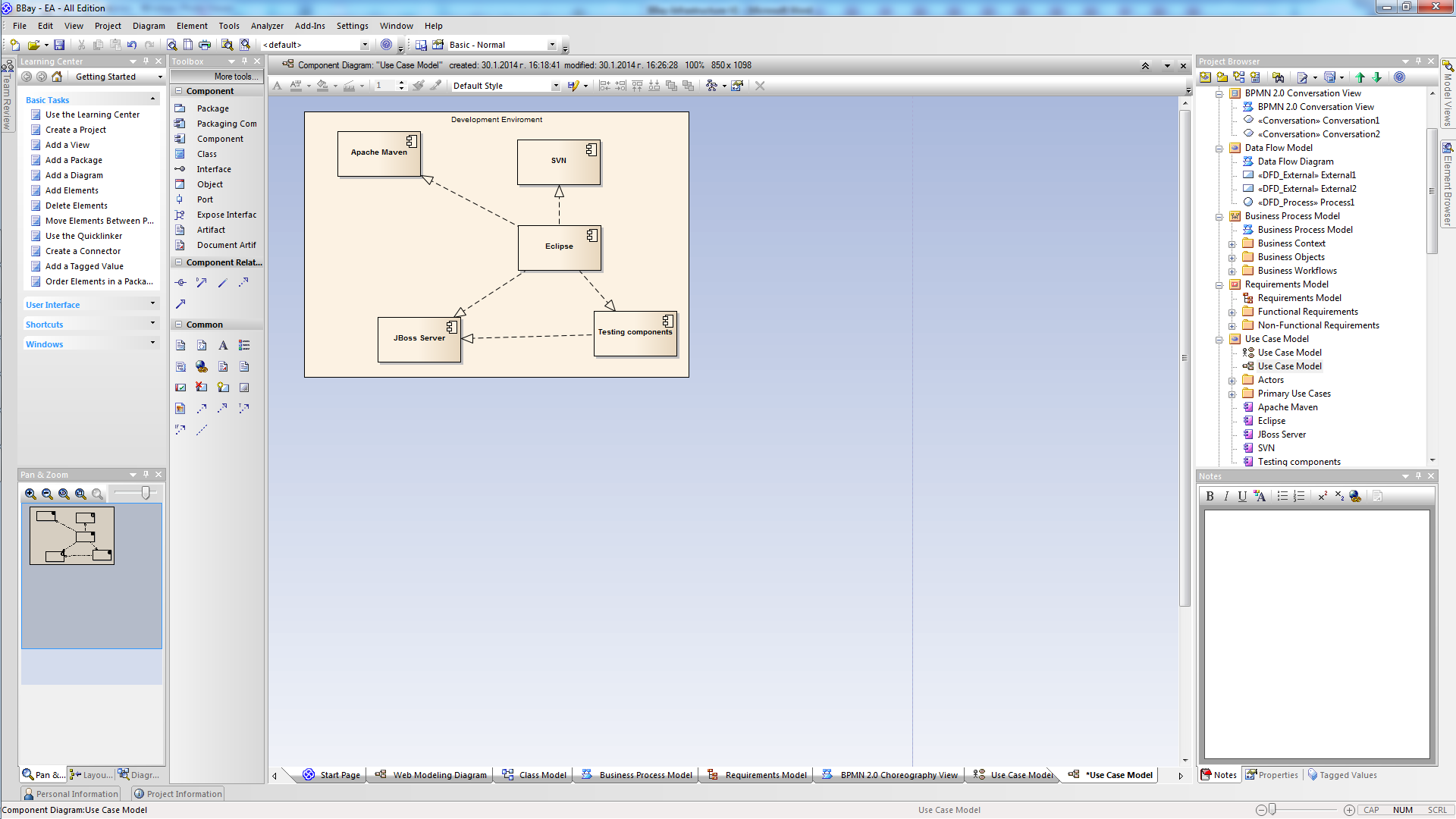
Различните модули, както и главните задачи се възлагат от мениджъра на проекта, за веки етап от жизнения цикъл на проекта. Също така само той може да затвори главна задача, когато всички нейни подзадачи са решени.

Останалите членове на екипа могат да си възлагат задачи един към друг, като както беше оказано по-горе те сами са отговорни за поддържането на задачите възложени към тях. Задачите се затварят след преглед от мениджъра на проекта.

Всяка задача има поне по един член на екипа, който отговоря за обновяването и редактирането и. За всяка задача се задава време, за което тя трябва да се приключи. Всеки който работи по дадена вписва време и свършена работа.

Софтуерът необходим и използван по време на дизайна и анализа на проекта е: MS Word, MS Excel, MS Project, както и средата за проектиране на диаграмите Enterprise Architect 7.5.

Enterprise Architect е инструмент на IBM за създаване на UML диаграми. Използва се при малко и големи проекти предимно по време на анализа и дизайна. Той е удобен за създаването на различните диаграми необходими за изграждането на един проект. Чрез него могат да се генерират готови документи, диаграми на база на други диаграми, а също така и готов код на различни езици. Инструментът бе предоставен от РУ “Ангел Кънчев “.



Фигура3 - Екран от работното поле на Enterprise Architect

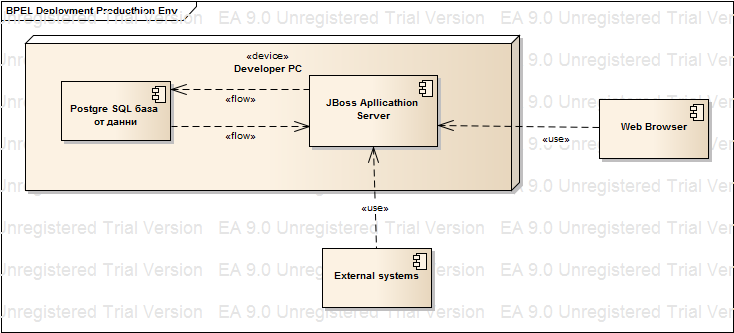
## Мрежово местоположение

Проекта в JIRA се намира на този адрес - http://puma.ami.uni-ruse.bg:8080/browse/EB

## Профили и пароли

Профили се създават от мениджъра на проекта след изпращане на данни или самостоятелно, когато е възможно. Тъй като се използва и университетската мрежа има случаи, в които е необходима връзка с мрежовите администратори на университета за създаване на специфични профили с различни права.

# Продукционна среда



Фигура4 - Инфраструктура на продукционната среда

## Хардуер

Хардуера използван за продукционната среда са ресурсите на предоставените от университета сървъри, на които са инсталирани Postgre SQL, JBoss Application server.

## Софтуер

Използваният софтуер се състой от подробно описаните в раздел Среда за разработка, JBoss Application server, WSO2 ESB приложенията от външните системи и Web Browser-ите инсталирани на потребителските машини.

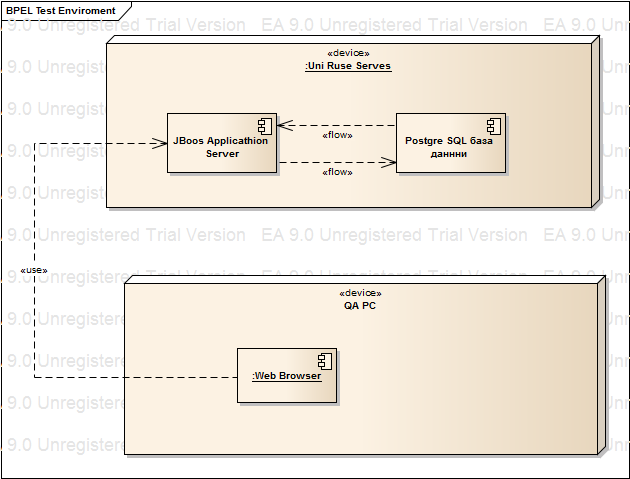
## Мрежово местоположение

Неизвестно на този етап.

## Профили и пароли

Ще бъдат създадени подходящи профили за администриране и подсигурени роли за всички категории на достъп до системата.

# Среда за тестване



Фигура5- Инфраструктура на тестовата среда

## Хардуер

Хардуера използван в процеса на тестване са персоналните компютри на всеки член на екипа и ресурсите на предоставените от университета сървъри, на които са инсталирани средите за управление на разработката

## Софтуер

Софтуерът, за тестване на системата са следните инструменти :

JUnit е прост модел на, който се пишат повторяеми тестове. Тя е пример за архитектурата xUnit за единица тестване модели.

Selenium е преносим модел за тестване на софтуер за уеб приложения. Selenium предоставя инструмент за запис / възпроизвеждане за авторство тестове без учене на език тест скриптове (Selenium IDE). Той също така предвижда тест домейн-конкретен език (Selenese), за да напишете тестове в редица популярни програмни езици, включително Java, C #, Groovy, Perl, PHP, Python и Ruby.

Изборът на тези инструменти е мотивиран от сервизно ориентираната архитектура, която ще бъде използвана за разработката на информационната система.

Инструментът е безплатен и ще бъде инсталиран на персоналните компютри на екипа по QA.