Модерно Банково Управление

(Advanced Bank Management)

План за осигуряване на качеството

Версия 1.1

История на промените

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Версия** | **Описание** | **Автор** |
| 1.12.2015 г. | 1.0 | Създаване на документа | Б. Дечев |
| 2.12.2015 г. | 1.1 | Довеждане до завършен вид | К. Гецов |
|  |  |  |  |
|  |  |  |  |

Съдържание

[1. Въведение 4](#_Toc442617764)

[1.1 Цел 4](#_Toc442617765)

[1.2 Дефиниции, акроними и абревиатури 4](#_Toc442617766)

[1.3 Препратки 4](#_Toc442617767)

[1.4 Обзор 4](#_Toc442617768)

[2. Целеви показатели за качество 4](#_Toc442617769)

[3. Управление 5](#_Toc442617770)

[3.1 Организация 5](#_Toc442617771)

[3.2 Задачи и отговорности 6](#_Toc442617772)

[4. Документация 6](#_Toc442617773)

[5. Стандарти и насоки 6](#_Toc442617774)

[6. Метрики 6](#_Toc442617775)

[7. План за преглед и проверка 8](#_Toc442617776)

[7.1 Предаване и приемане на итерация 8](#_Toc442617777)

[7.2 Преглед на отчетната документация 11](#_Toc442617778)

[8. Оценка и тест 11](#_Toc442617779)

[9. Рискове, разрешаване на проблеми и коригиращи действия 11](#_Toc442617780)

[10. Инструменти, техники и методологии 11](#_Toc442617781)

[11. Управление на конфигурацията 11](#_Toc442617782)

[12. Записки по качеството 11](#_Toc442617783)

План за осигуряване на качеството

# Въведение

Планът за осигуряване на качеството е документ, който дава ясна представа за това как трябва да се разработи даден продукт, за да сме сигурни в неговото качеството. Той съдържа план за преглед и одит, и референцията на редица други документи, разработени по време на фазата на Inception. Той се поддържа в рамките на целия проект.

## Цел

Целта на този документи е гарантирането на коректно свършената работа, ползване на времевия ресурс и поставяне на отговарящи на уменията на екипа задачи.

## Дефиниции, акроними и абревиатури

Дефинираните акроними и абревиатури могат да бъдат намерени в следния документ:

* ABM-I1-Glossary;

## Препратки

Този документ включва препратки към следните документи:

* ABM-E1-Software Requirements Specifications
* ABM-I1-Glossary
* ABM-I1-Software Development Plan

## Обзор

До края на този документ е включена следната информация:

* Показатели за качество.
* Спецификация с отговорностите и задачите в екипа.
* Необходима документация за проекта.
* Стандарти и насоки за реализация на проекта.
* Метрики за гарантиране на качество.
* План за преглед на проверка.
* Инструменти техники и методологии.
* Записки по качеството.

# Целеви показатели за качество

Основните показатели, които засягат качеството на разработвания продукт са описани в документа „ABM-E1-Software Requirements Specifications ” (спецификация на софтуерните изисквания) – това са следните раздели и съответно целевите показатели за качество в тях:

* „Използваемост”:
  + Уеб модула на системата трябва да работи без проблемно с повечето модерни браузери.
* „Надеждност”:
  + Системата трябва да е достъпна по всяко време. Веднъж пусната в експлоатация, системата трябва да работи 24/7/365.
  + Системата трябва да предлага всички модерни и доказани методи за осигуряване на сигурността на потребителите си.
  + За намаляване на шанса системата да остане без достъп до интернет, поради аварии и други причини, системата е подсигурена от 3 интернет доставчика.
  + Време за възстановяване на системата в случай на установяване на повреда, системата поддържа възстановяване до най-много един час.
  + Информацията в системата се архивира автоматично всеки ден в 00:00 (GMT).
* „Изпълнение и поддръжка”:
  + Системата трябва да поддържа едновременна работа на около 500 000 потребители. (клиенти и служители общо).
  + За писане на програмния код ще се използва Java Code Conventions
  + За графично представяне на моделите ще се изисква UML стандарта
  + За представянето на бизнес процесите е се изисква BPMN 2 стандарта

# Управление

## Организация

|  |  |
| --- | --- |
| Роля | Отговорности |
| Мениджър на проекта(PM) | Разпределя ресурсите, определя приоритетите на участниците в екипа. Координира комуникацията с клиента и потребителите. Също така установява набор от практики които подсигуряват цялостен и качествен процес на разработка на проекта. |
| Системен аналитик | Открива и координира моделирането на всички потребителски случаи в системните изисквания |
| Софтуерен архитект | Води и координира техническите дейности в проекта. Установява цялостна структура на архитектурата на продукта: определя технологиите, които ще се използват за цялото решение на системата. |
| Разработчик | Отговаря за разработването и тестването на компонентите на системата в съответствие с определените стандарти и конвекции за писане на код. Също така отговорен за разработването и тестването на тест компонентите и съответните подсистеми. |
| Тест дизайнер | Отговаря за планирането, разработването моделирането, изпълнението и на тестовете. Също така за разработването на тест план и тест модел. Определя кои тестове трябва да са ръчни и кои автоматични и създава автоматичните. |
| Тестер | Отговорен е за изпълнението на тестовете, включително подготовката на тестовете и стартирането на тестовете, оценка на теста и възможността за възстановяване от грешки, регистрирането на дефекти и записване на тестови резултати. |

## Задачи и отговорности

Този раздел съдържа задачите и отговорности свързани с осигуряването на качеството на продукта, както и детайлно описание на тези задачите. Описани са и ролите, които са отговорни за изпълнението на задачите в зависимост от тяхната специфика.

* Съвместни ревизии – реализират се съвместно от членове на екипа, изпълняващ проекта – екип QA, и член(ове) от екипа на възложителите;
* Одити на процеса – реализира се от един или повече одитори, които не са част от екипа по разработката, които реализират т.н. независима проверка на даден софтуерен продукт, софтуерен процес или набор от процеси за да определят дали съответства със спецификацията на изискванията, стандартите или други зададени критерии. (Тези одитори ще играят роля екип QA – Борислав и Калоян);
* Ревизии на процеса – членове на екипа на изпълнителите на проекта ще осъществяват редовни ревизии на процесите за разработка на продукта за да се откриват навреме несъответствия, проблеми или да се правят определени подобрения. Тези ревизии също ще се извършват главно от екип QA – Борислав и Калоян;
* Клиентски одити – одити, извъшвани от членове на екипа на възложителите на проекта;

# Документация

Този раздел съдържа списък на документите, които трябва да се създадат по време на проекта и които отговарят за това, качеството на продукта да удовлетворява зададените изисквания:

* График за създаване на документи – създава се в началото на фаза Планиране с цел точно и ясно разпределение на задачите от гледна точка на необходимата документация за проекта;
* Речник – създава се в началото на фаза Планиране и се работи по него до края на проекта;
* Главен план за тестване – създава се във фаза Планиране или началото на фаза Детайлизиране;
* Планове на итерациите – създават се във фаза Планиране и преди началото на всяка следваща фаза/итерация;
* Спецификация на софтуерните изисквания – функционалните и нефункционалните изисквания се определят във фаза Планиране, а спецификацията на изискванията се разработва във фаза Детайлизиране;
* Софтуерна архитектура – създава се във фаза Детайлизиране;
* Потребителска документация (за инсталиране, администриране, поддръжка) – създава се в последната фаза Изграждане;

# Стандарти и насоки

За разработването на различни части от проекта ще се използват стандарти и насоки като:

* Javadoc – конвенция на Oracle Corporation за генериране на API документация в HTML формат от Java сорс код;
* UML – стандартизиран език с общо приложение за моделиране в областта на софтуерното инженерство. Включва набор от графични техники за създаване на диаграми в обектно-ориентираните софтуерни системи.
* BPMN – графично представяне на бизнес процеси в модела на бизнес процесите;

# Метрики

Тази секция си поставя за цел намиране и коригиране на грешки в системата и оптимизиране свършената работа те се разделят на следните:

* Процент на изпълнени тестови случаи: Носи информация в проценти за Общ. Бр Изпълнени тестови случаи.

Получава се по следната формула:

**Процент на изпълнени** = (Изпълнени/всички написани тестови случаи)\*100

**Трябва да надвишава 80%;**

* Процент на неизпълнени тестови случаи: Аналогично на предната формула заменяне изпълнените тестови случай с не изпълнените.
* Процент на преминати тестови случаи: Дава ни информация в проценти за преминатите тестови случаи.

Формулата за намирането им е следната:

**Процент на преминати** = (Успешно преминати/Изпълнени тестови случаи)\*100

**Трябва да надвишава 90%;**

* Бр. неуспешно преминати тестови случаи: Ползвайки същата формула но чрез замяна на преминали с непреминали ще получим броя на неуспешните тестови случаи.
* Процент на блокирани тестови случаи: Служи за определяне на процента на блокираните тестови случаи. Блокирани са тези тестови случаи до които не сме могли да достигнем поради грешка допусната в по ранен етап или неспособност да ги тестваме с наличните ни инструменти и технологии.

Формула:

**Процент блокирани** = (бр.блокирани/всички изпълнени)\*100

**Не трябва да надвишава 10%;**

* Честота на дефекти: Показва коефициент за намиране на дефекти при определено количество тестван код.

Формула:

**Честота** = Открити дефекти/Честота на дефекти.

* Ефикасност при премахване на дефекти: Тази метрика определя ефикасността на системата. Тя се простира до Alpha и Beta етапите на системата.

Формула:

**Ефикасност** = ( Development&QA testing / [ Development&QA testing + Alpha&Beta testing ])\*100

**Трябва да надвишава 80%;**

* Пропуснати дефекти: Тази метрика ни позволява да проверим качеството на QA тестовете. Съпоставя дефектите намерени от потребители в Alpha&Beta тестовете спрямо тези намерени от QA тестерите.

Формула:

**Процент пропуснати тестове** = (Alpha&Beta tests/QA tests)\*100;

**Ниско качество на тестване се счета надвишаването на 10%;**

* Нива на дефекти: Тук се разглеждат всички дефекти и се разделят на категории, за оптимизиране последователността от решаването им.

Критично ниво:

Процент критични дефекти = (бр. критични / всички дефекти) \* 100

Високо ниво:

Процент сериозни дефекти = (бр. сериозни / всички дефекти) \* 100

Средно ниво:

Процент средни дефекти = (бр. средни / всички дефекти) \* 100

Ниско ниво:

Процент малки дефекти = (бр. малки / всички дефекти) \* 100

|  |  |
| --- | --- |
| Номер | Метрика |
| 1 | Процент на изпълнени случаи. |
| 2 | Средно аритметично на направени тестове за едно изискване. |
| 3 | Общ брой на тестови случай направени за всички изисквания. |
| 4 | Общ брой на изпълнени тестови случай. |
| 5 | Общ брой на преминати тестови случай. |
| 6 | Общ брой на пропаднали тестови случай. |
| 7 | Общ брой на блокирани тестови случай. |
| 8 | Общ брой на не направени тестови случай. |
| 9 | Общ брой на открити дефекти. |
| 10 | Общ брой на критични дефекти. |
| 11 | Бр Големи дефекти. |
| 12 | Бр средни дефекти. |
| 13 | Бр ниски дефекти. |

# План за преглед и проверка

## Предаване и приемане на итерация

След всяка итерация е предвидено да се проведе среща между екипа по разработката и на поне един представител на възложителя. На тази срещи членовете на екипа разработващи системата представят официално чрез презентации извършените дейности по време на итерацията, както и необходимите завършени документи, които трябва да бъдат разгледани и приети от възложителя.

На срещите за предаване и приемане на итерация представителите на възложителя трябва да се запознаят с всички предавани документи. След това, те трябва да решат дали предадените артефакти имат нужното качество. Ако това не е така, се предвижда представителя да върне бележки и той да се уговори с представящите екипа хора, за насрочването на нова среща за предаване на итерацията. Новоназначената среща трябва да бъде в максимално кратък срок. В случат, когато представителят на възложителя приеме предадените му артефакти, то се приема, че те отговарят на необходимите критерии за качество и итерацията се приема за предадена.

**Предавани артефакти:**

*Итерация на планиране I1:*

* **План за разработка на софтуерния продукт** – предаване на документ „План за разработка на софтуерния продукт”. На този етап документът е разработен до голяма степен, но поради неговата важност и обхват, той бива допълван и в следващите фази и итерации на проекта.
* **План за управление на качеството** – предаване на документ „План за управление на качеството“ . На този етап документът е разработен, но поради големия си обхват, той бива допълван и в следващите фази и итерации на проекта.
* **Списък на рисковете** – към планът за разработка се предвижда да се предаде и списък с рисковете, които са били идентифицирани по време на итерацията. В рамките на жизнения цикъл на проекта е възможно този документ да бъде редактиран, за да може той да отговаря на ново откритите или вече елиминирани рискове.
* **Детайлен план за итерация на Е1** – мениджърът на проекта в края на итерацията на планиране, трябва да предаде и детайлен план за следващата итерация. В него той трябва да представи пред възложителя планираните дейности и ресурси за итерацията на планиране Е1.
* **Визия** – документирането на виждането на участниците в проекта относно продукта(основни нужди и характеристики) се прави в документа „Визия”. В рамките на тази итерация се представят основните изисквания, характеристики и ограничения на проекта.
* **Първа версия на Спецификация на софтуерните изисквания** – в края на итерацията се представя и работен вариант на документа „Спецификация на софтуерните изисквания”, който трябва да бъде в напреднал стадии на разработка. В него трябва да се идентифицират основните системни функционалности.
* **Конвенции за писане на код** – На този етап се предава и документа „Конвенции за писане на код“ в неговия краен вариант. В него се определят основните методологии и правила за писане на код.
* **Речник** – Речникът също се предава на този етап, след което се обновява и поддържа от целия екип.

*Итерация на детайлизиране Е1:*

* **Стабилна версия на Спецификация на софтуерните изисквания** – стабилна версия на документа „Спецификация на софтуерните изисквания”.
* **Детайлен план за итерация на Е2** – мениджърът на проекта в края на първата итерация на детайлизиране, трябва да предаде и детайлен план за следващата итерация. В него той трябва да представи пред възложителя планираните дейности и ресурси за итерацията на планиране Е2.
* **Първа версия на Главен план за тестване** – този документ бива разработен в рамките на итерацията за детайлизиране и представя идеи за управление на тестването. Той има ключова роля за постигането на високо качество на разработвания софтуерен продукт.
* **Модел на инфраструктурата** – предаване на документът „Модел на инфраструктура” е отговорност на софтуерния архитект..
* **Първа версия на Тестов модел** – тестовия модел се представя и поддържа от тест мениджъра в разработващия екип.
* **Първа версия на Бизнес модел** – отговорността на този документ е на бизнес аналитика.

*Итерация на детайлизиране Е2:*

* **Софтуерна архитектура** – отговорен за изработката и поддържането й е на софтуерния архитект
* **Първа версия на Модел на потребителски случаи** – отговорността за документа се пада на бизнес аналитика
* **Стабилна версия на Бизнес модела** – отговорността на документа се пада на бизнес аналитика.
* **Детайлен план за итерация Е3** – мениджърът на проекта в края на втората итерация на детайлизиране трябва да предаде и детайлен план за следващата итерация. В него той трябва да представи пред възложителя планираните дейности и ресурси за итерацията на планиране Е3.
* **Прототип** – прототипа бива представен пред възложителя от разработчиците взели участие за неговото изграждане.

*Итерация на детайлизиране Е3:*

* **Стабилна версия на Модел на потребителски случаи** – отговорността за документа се пада на бизнес аналитика
* **Инструменти** – този документ определя инструментите, които ще се използват за подпомагане в разработката на проекта.
* **Детайлен план за итерация C1** – мениджърът на проекта в края на втората итерация на детайлизиране трябва да предаде и детайлен план за следващата итерация. В него той трябва да представи пред възложителя планираните дейности и ресурси за итерацията на планиране C1.
* **Дизайн модел** – код дизайнера трябва да представи този документ.
* **Стабилна версия на Тестов модел** – тестовия модел се представя и поддържа от тест мениджъра в разработващия екип.
* **Стабилна версия на Главен план за тестване** – този документ бива разработен в рамките на итерацията за детайлизиране и представя идеи за управление на тестването. Той има ключова роля за постигането на високо качество на разработвания софтуерен продукт.

*Итерация на разработка C1:*

* **Детайлен план за итерация C2** – Мениджърът на проекта в края на първата итерацията на детайлизиране, трябва да предаде и детайлен план за следващата итерация. В него той трябва да представи пред възложителя планираните дейности и ресурси за итерацията на планиране C2.
* **Представяне на разработена Бизнес Функционална Група 1 (БФГ1)** - отговорността за представянето на разработените модули пада върху разработчиците.
* **Резултати от първични тестове на разработените модули от БФГ1 -** отговорността пада за резултатите пада върху ролите на тест мениджър и тестер.
* **Резултати от функционални тестове** – тест мениджърът и тестерът представят резултатите.

*Итерация на разработка C2:*

* **Детайлен план за итерация C3** – Мениджърът на проекта в края на първата итерацията на детайлизиране, трябва да предаде и детайлен план за следващата итерация. В него той трябва да представи пред възложителя планираните дейности и ресурси за итерацията на планиране C3.
* **Представяне на разработена Бизнес Функционална Група 2 (БФГ2)** - отговорността за представянето на разработените модули пада върху разработчиците.
* **Резултати от първични тестове на разработените модули от БФГ2 -** отговорността пада за резултатите пада върху ролите на тест мениджър и тестер.
* **Резултати от функционални тестове** – тест мениджърът и тестерът представят резултатите.

*Итерация на разработка C3:*

* **Детайлен план за итерация C4** – Мениджърът на проекта в края на първата итерацията на детайлизиране, трябва да предаде и детайлен план за следващата итерация. В него той трябва да представи пред възложителя планираните дейности и ресурси за итерацията на планиране C4.
* **Представяне на разработена Бизнес Функционална Група 3 (БФГ3)** - отговорността за представянето на разработените модули пада върху разработчиците.
* **Резултати от първични тестове на разработените модули от БФГ3 -** отговорността пада за резултатите пада върху ролите на тест мениджър и тестер.
* **Резултати от функционални тестове** – тест мениджърът и тестерът представят резултатите.

*Итерация на разработка C4:*

* **Детайлен план за итерация T1** – Мениджърът на проекта в края на първата итерацията на детайлизиране, трябва да предаде и детайлен план за следващата итерация. В него той трябва да представи пред възложителя планираните дейности и ресурси за итерацията на планиране T1.
* **Представяне на разработена Бизнес Функционална Група 4 (БФГ4)** - отговорността за представянето на разработените модули пада върху разработчиците.
* **Резултати от първични тестове на разработените модули от БФГ4 -** отговорността пада за резултатите пада върху ролите на тест мениджър и тестер.
* **Резултати от функционални тестове** – тест мениджърът и тестерът представят резултатите.

*Итерация на внедряване Т1:*

* **Резултати от функционални интегрални тестове -** отговорността пада за резултатите пада върху ролите на тест мениджър и тестер.
* **Резултати от провеждането на тестове по приемане системата**

## Преглед на отчетната документация

**Преди един** документ **да достигне до стабилно състояние той трябва да бъде проверен**. Проверките **се извършват с цел повишаване** качеството на документация**та**. Всяка проверка трябва да се извърши от член на екипа, различен от поставения отговорник за документа.

# Оценка и тест

Оценките и тестовете ще бъдат разработени на следващите итерации и ще се опишат в документите:

* ABM-I1-Software Development Plan
* ABM-C1-Test Results

# Рискове, разрешаване на проблеми и коригиращи действия

При възникване на проблем по време на проекта екипът на изпълнителите анализира възникналия проблем и взема единно решение за разрешаването. При необходимост е възможно екипът на изпълнителите да се свърже с екипа от възложители за консултация за коректното отстраняване на възникналия проблем.

По-подробно описание на управлението на рисковете е представено в документ „ABM\_Risk-Management-Plan“.

# Инструменти, техники и методологии

RUP (Rational Unified Process) – стандарт за разработване на големи софтуерни проекти.

JIRA – среда, съдържаща множество инструменти, които спомагат за управлението на разработката на софтуерния проект и проследяване на грешки.

ECLIPSE – среда за разработване на софтуер.

GIT – среда за паралелна работа по документи и програми, синхронизираща работата на всички членове на екипа.

# Управление на конфигурацията

При необходимост за промяна или определяне на график от дейности, възложени отговорности, нужни ресурси за изпълнението на проекта, през целия му жизнен цикъл, член от екипа на изпълнителите в лицето (Б. Дечев) ще се консултира с екипа от възложители.

# Записки по качеството

Свършената работа и времето което е отнело за свършването и ще могат да бъдат видени в JIRA. В отсъствието на JIRA поради административни неизправности ще се ползва GIT.