Модерно Банково Управление

(Advanced Bank Management)

Спецификация на софтуерните изисквания

Версия <1.0>

История на промените

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Версия** | **Описание** | **Автор** |
| 30/11/2015 | 1.0 | Създаване на документа | <name> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Съдържание

1. Въведение 2

1.1 Цел 2

1.2 Scope 2

1.3 Дефиниции, акроними и абревиатури 2

1.4 Препратки 2

1.5 Overview 2

2. Requirements Management 2

2.1 Organization, Responsibilities, and Interfaces 2

2.2 Tools, Environment, and Infrastructure 2

3. The Requirements Management Program 2

3.1 Requirements Identification 2

3.2 Traceability 2

3.2.1 Criteria for <traceability item> 2

3.3 Attributes 2

3.3.1 Attributes for <traceability item> 2

3.4 Reports and Measures 2

3.5 Requirements Change Management 2

3.5.1 Change Request Processing and Approval 2

3.5.2 Change Control Board (CCB) 2

3.5.3 Project Baselines 2

3.6 Activities and Tasks 2

4. Milestones 2

5. Training and Resources 2

Requirements Management Plan

# Въведение

## Цел

Целта на този документ е да представи и опише подробно софтуерните изисквания за Банкова информационна система (БИС).

## Scope

БИС, банковата информационна система представлява съвкупност от софтуерни продукти управлявани от единно ядро. Нейната цел е осигуряването на сигурна и надеждна среда за работа с банковите активи, в и извън страната. Потребителите на системата ще разполагат с бърз достъп до нужната на клиентите информация, както и ще могат да правят и изпълняват заявки към лични и правителствени бази от данни. Системата позволява регистрирането и разделението на потребители в йерархия. Тази система също така разполага с гъвкава функционалност, която и позволява връзки с други системи.

## Дефиниции, акроними и абревиатури

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

## Препратки

[This subsection provides a complete list of all documents referenced elsewhere in the **Requirements Management Plan**. Identify each document by title, report number if applicable, date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.]

## Overview

[This subsection describes what the rest of the **Requirements Management Plan** contains and explains how the document is organized.]

# Requirements Management

## Organization, Responsibilities, and Interfaces

Отговорни за този документ са Калоян Гецов и Мартин Абрашев. Работата свършена в документа цели обособяване на събраната информация и спомагане създаването на макетен модел на софтуерния интерфейс. Тази работа ще се извърши от Калоян Гецов и Мартин Абрашев в качеството си на спецификатори на потребителски изисквания. Както и Адриан --- графичен дизайнер.

## Tools, Environment, and Infrastructure

[Describe the computing environment and software tools to be used in fulfilling the Requirements Management functions throughout the project or product lifecycle.

Describe the tools and procedures used to version control Requirements items generated throughout the project or product lifecycle.]

# The Requirements Management Program

## Requirements Identification

[Describe traceability items and define how they are to be named, marked, and numbered. (A traceability item is any project element that needs to be explicitly traced from another textual or model item in order to keep track of the dependencies between them. With respect to Rational Requisite Pro, this definition can be rephrased as: any project element represented within RequisitePro by an instance of a RequisitePro requirement type.)]

[For each type of requirement work product in your project, list the traceability items contained in it and briefly explain what it is used for. You may also wish to list the responsible role.]

|  |  |  |
| --- | --- | --- |
| **Work Product**  **(Document Type)** | **Traceability Item** | **Description** |
| Stakeholder Requests (STR) | Stakeholder Request (STRQ) | Key requests, including Change Requests, from stakeholders  *[If you use a Change Request Management tool, such as Rational ClearQuest, then stakeholder requests are often stored in that tool and not duplicated in the requirements management tool.]* |
| Vision (VIS) | Stakeholder Need (NEED) | Key stakeholder or user need |
| Vision (VIS) | Feature (FEAT) | Conditions or capabilities of this release of the system |
| Use-Case Model | Use Case (UC) | Use cases for this release, documented in Rational Rose |
| Supplementary Specification (SS) | Supplementary Requirement (SUPP) | Non-functional requirements that are not captured in the use-case model |

## Traceability

[Overview of traceability, for example, a traceability graph.]

### Criteria for <traceability item>

[For each traceability item you have identified, list any additional rules or guidelines that apply to traceability links. Describe any applicable constraints, such as “every approved feature must trace to one or more Use Cases or to one or more Supplementary Requirements”.]

## Attributes

### Attributes for <traceability item>

[For each traceability item you have identified, list what attributes you will be using and briefly explain what they mean. For example, the following attributes might be specified for a traceability item of “feature”.]

Status

[Set after negotiation and review by the project management team. Tracks progress during definition of the project baseline.]

|  |  |
| --- | --- |
| Proposed | [Used to describe features that are under discussion but have not yet been reviewed and accepted by the "official channel", such as a working group consisting of representatives from the project team, product management, and user or customer community.] |
| Approved | [Capabilities that are deemed useful and feasible, and have been approved for implementation by the official channel.] |
| Rejected | [Rejected by the official channel.] |
| Incorporated | [Features incorporated into the product baseline at a specific point in time.] |

****Benefit****

[Set by Marketing, the product manager or the business analyst. All requirements are not created equal. Ranking requirements by their relative benefit to the end user opens a dialogue with customers, analysts, and members of the development team. Used in managing scope and determining development priority.]

|  |  |
| --- | --- |
| Critical | [Essential features. Failure to implement means the system will not meet customer needs. All critical features must be implemented in the release or the schedule will slip.] |
| Important | [Features important to the effectiveness and efficiency of the system for most applications. The functionality cannot be easily provided in some other way. Lack of inclusion of an important feature may affect customer or user satisfaction, or even revenue, but release will not be delayed due to lack of any important feature.] |
| Useful | [Features that are useful in less typical applications or for which reasonably efficient workarounds can be achieved will be used less frequently. No significant revenue or customer satisfaction impact can be expected if such an item is not included in a release.] |

****Effort****

[Set by the development team. Because some features require more time and resources than others, estimating the number of team or person-weeks, lines of code required or function points, for example, is the best way to gauge complexity and set expectations of what can and cannot be accomplished in a given time frame. Used in managing scope and determining development priority.]

****Risk****

[Set by the development team and based on the probability the project will experience undesirable events, such as cost overruns, schedule delays, or even cancellations. Most project managers find categorizing risks as high, medium, and low to be sufficient, although finer gradations are possible. Risk can often be assessed indirectly by measuring the uncertainty (range) of the projects’ team’s estimated schedule.]

****Stability****

[Set by the analyst and development team, this is based on the probability that the feature will change or the team’s understanding of the feature will change. Used to help establish development priorities and determine those items for which additional elicitation is the appropriate next action.]

****Target Release****

[Records the intended product version in which the feature will first appear. This field can be used to allocate features from a **Vision** document into a particular baseline release. When combined with the status field, your team can propose, record, and discuss various features of the release without committing them to development. Only features whose Status is set to Incorporated and whose Target Release is defined will be implemented. When scope management occurs, the Target Release Version Number can be increased so the item will remain in the **Vision** document, but will be scheduled for a later release.]

****Assigned to****

[In many projects, features will be assigned to "feature teams" responsible for further elicitation, writing the software requirements and implementation. This simple pull-down list will help everyone on the project team to better understand responsibilities.]

****Reason****

[This text field is used to track the source of the requested feature. Requirements exist for specific reasons. This field records an explanation or a reference to an explanation. For example, the reference might be to a page and line number of a product requirement specification or to a minute marker on a video of an important customer interview.]

## Reports and Measures

[Describe the content, format, and purpose of the requested reports or measures.]

## Requirements Change Management

### Change Request Processing and Approval

[Describe the process by which problems and changes are submitted, reviewed, and dispositioned. This should include the process for negotiating requirements changes with customers, and any contractual processes, activities, and constraints.]

### Change Control Board (CCB)

[Describe the membership and procedures for processing change requests and approvals to be followed by the CCB.]

### Project Baselines

[Baselines provide an official standard on which subsequent work is based and to which only authorized changes are made.

Describe at what points during the project or product lifecycle baselines are to be established. The most common baselines would be at the end of the Inception, Elaboration, Construction, and Transition phases. Baselines could also be generated at the end of iterations within the various phases or even more frequently.

Describe who authorizes a baseline and what goes into it.]

## Activities and Tasks

[Describe the activities and tasks that apply to managing requirements.

Describe review tasks, including review objectives, responsibilities, timing, and procedures.]

# Milestones

[Identify the internal and customer milestones related to the Requirements Management effort. This section should include details on when the Requirements Management Plan itself is to be updated.]

# Training and Resources

[Describe the software tools, personnel, and training required to implement the specified Requirements Management activities.]