

## **Opis możliwości programu do zarządzania wymaganiami IBM Rational RequisitePro.**

RequisitePro jest narzędziem wspomagającym zarządzanie wymaganiami. Za pomocą tego programu członkowie zespołu projektowego mogą wszechstronnie zarządzać wymaganiami, kontaktować się i współpracować ze sobą podczas realizacji projektu, zmniejszając w ten sposób ryzyko jego niepowodzenia.

Zdefiniowane w projekcie wymagania są przechowywane w bazie danych (użytkownik ma do wyboru następujące serwery baz danych: MsAccess, Oracle lub MsSQL). Jednakże RequisitePro umożliwia również współpracę z MSWord dzięki czemu nowe wymagania w prosty sposób mogą być dodawane do bazy wymagań z poziomu dokumentu zawierającego zbiór wymagań dotyczących danego projektu. Wprowadzane do dokumentów zmiany automatycznie są przekazywane do bazy danych – zawiera więc ona zawsze aktualny zestaw wymagań. Jak widać zarządzanie wymaganiami może odbywać się zarówno z poziomu interfejsu RequisitePro jak również z poziomu udostępnianych przez RequisitePro dokumentów.

### **Projekt.**

W celu rozpoczęcia pracy użytkownik tworzy projekt. Nowy projekt może zostać stworzony na podstawie istniejących szablonów. RequisitePro udostępnia następujące szablony projektów:

- *Use-Case Template* (Szablon ten oparty jest na metodologii przypadków użycia. Idealnie nadaje się do implementacji *Rational Unified Process* i jest używany do integracji zdefiniowanych w Rational Rose przypadków użycia z projektem wymagań RequisitePro),
- *Traditional Template* (Szablon ten oparty jest na tradycyjnym podejściu do specyfikacji wymagań podczas tworzenia oprogramowania ),
- *Composite Template* (Łączy oba powyższe szablony ).

Użytkownik ma również możliwość stworzenia własnego szablonu projektu lub rozpoczęcia pracy z pustym projektem.

Informacje zawarte w projekcie są podzielone na tzw. pakiety (*packages*), co pomaga w ich grupowaniu. W skład każdego pakietu mogą wchodzić następujące elementy:

- *documents* ( dokumenty ),
- *requirements* ( wymagania ),
- *views* ( widoki ).

### **Dokumenty.**

RequisitePro umożliwia współpracę z MSWord w celu dokumentacji wymagań. Po utworzeniu projektu na podstawie jednego z istniejących szablonów, dostępnych jest kilka rodzajów gotowych dokumentów:

- *vision* (dokument zawierający ogólny opis właściwości tworzonego systemu: główne charakterystyki, kluczowe wymagania stawiane przez zleceniodawcę, główne udostępniane serwisy itp.),

- *glossary* (słownik zawierający definicję wszystkich użytych w projekcie terminów),
- *use-case specification* (zawierający opis zdefiniowanych w projekcie przypadków użycia – wymagań funkcjonalnych),
- *supplementary specification* (zawierający wymagania, które nie znalazły się w specyfikacji przypadków użycia – są to wymagania нефункционалне),
- *test plan* (zawierający informacje na temat planów testów – jakie komponenty systemu będą testowane i w jaki sposób),

Ponadto użytkownik może tworzyć własne typy dokumentów.

Korzystając z dokumentów użytkownik ma możliwość dodawania lub usuwania wymagań jak również zmiany ich właściwości (opis, atrybuty itp.). Zmiany wprowadzone w dokumencie od razu propagują do bazy danych co zapewnia spójność informacji a jednocześnie użytkownik może zdecydować czy woli zarządzać wymaganiami z poziomu dokumentów czy interfejsu RequisitePro.

## **Wymagania.**

RequisitePro umożliwia organizację wymagań na podstawie ich typów (grupowanie podobnych wymagań w projekcie). Typ wymagania definiuje informacje z nim związane. Do przykładowych typów należą wymagania określające cechy projektowanego systemu (*feature requirements*), wymagania funkcjonalne (*use-case requirements*), wymagania нефункционалне (*supplemental specification requirements*). Ponadto użytkownik ma możliwość tworzenia własnych typów wymagań.

Wymagania są dodawane do poszczególnych pakietów projektu w zależności od ich typów. Poza zwykłym dodawaniem wymagań możliwe jest również tworzenie ich hierarchii (definiowanie między wymaganiami relacji rodzic – dziecko). Tworzenie hierarchii wymagań pozwala na podział ogólnego wymagania na kilka bardziej szczegółowych, a co za tym idzie na jego dokładniejszy opis.

Z wymaganiami związane są również ich atrybuty. Każde nowo tworzone wymaganie posiada kilka przypisanych domyślnych atrybutów. Użytkownik ma możliwość usuwania, dodawania lub modyfikacji wartości już istniejących atrybutów. Przykładowymi atrybutami wymagań mogą być: status, priorytet, ryzyko poprawnej realizacji, koszt implementacji, planowana data zakończenia.

Podczas realizacji projektu część wymagań, bez względu na to jak dokładnie zostały wcześniej zdefiniowane, ulega zmianie. RequisitePro udostępnia użytkownikowi narzędzia do śledzenia tych zmian. Użytkownik ma możliwość zdefiniowania relacji zależności między dwoma wymaganiami (dodatkowo wymagania te mogą należeć do dwóch różnych projektów). W momencie zmiany nazwy wymagania, jego opisu lub któregoś z atrybutów, relacja zdefiniowana między wymaganiami zostaje uznana za „podejrzaną” (*suspect*). W gestii użytkownika pozostaje sprawdzenie czy wprowadzone do wymagania zmiany nie naruszają poprawności relacji między nimi (czy zmiany wprowadzone w jednym wymaganiu nie wymuszają wprowadzenia zmian w innym wymaganiu). Ponadto RequisitePro zbiera historię zmian wprowadzonych do wymagań. W ten sposób można śledzić kto, kiedy i w jaki sposób dokonał modyfikacji wymagania.

## Widoki.

Jak wspomniano wcześniej zarządzanie wymaganiami może odbywać się z poziomu dokumentów jak i z poziomu interfejsu RequisitePro. Zarządzanie z poziomu interfejsu użytkownika odbywa się za pomocą widoków. RequisitePro definiuje 3 rodzaje widoków dostępne dla użytkownika:

- *attribute matrix* (macierz atrybutów zawierająca wszystkie wymagania określonego typu wraz z ich atrybutami; za jej pośrednictwem możliwe jest dodawanie, usuwanie i modyfikacja wymagań lub ich atrybutów),
- *traceability matrix* (macierz śledzenia ilustrująca relacje pomiędzy wymaganiami tych samych lub różnych typów, przy jej pomocy można dodawać, usuwać i modyfikować zależności między wymaganiami, możliwe jest również znajdowanie i analiza relacji „podejrzanych” ),
- *traceability tree* (drzewo śledzenia pełni tę samą rolę co macierz śledzenia, z tym że wymagania i relacje pomiędzy nimi są zilustrowane w postaci drzewa; ponadto drzewo śledzenia umożliwia również wyświetlanie relacji pomiędzy wymaganiami należącymi do różnych projektów).

Podczas pracy z widokami możliwe jest sortowanie lub filtrowanie danych zawartych w widokach. Przykładami zastosowań filtrów może być wyświetlenie w macierzy atrybutów jedynie wymagań o priorytecie „High” lub wyszukanie wszystkich „podejrzanych” relacji. Przykład sortowania – wyświetlenie wymagań posortowanych według wysokości priorytetu.

## Dodatkowe funkcje.

RequisitePro udostępnia kilka dodatkowych funkcji. Ich znajomość nie jest konieczna do skutecznego zarządzania wymaganiami ale, szczególnie przy pracy z większymi projektami, może okazać się przydatna.

Jedną z ważniejszych funkcji jest możliwość integracji programu RequisitePro z Rational Rose, dzięki czemu modelując system za pomocą Rose możemy korzystać ze zdefiniowanej w RequisitePro bazy wymagań. Możliwa jest również współpraca w drugą stronę tj. z każdym z wymagań można powiązać element modelowanego systemu z Rose.

### Możliwości integracji:

- ze zdefiniowanym elementem typu *use case* można powiązać wymaganie zdefiniowane uprzednio w bazie wymagań programu RequisitePro,
- ze zdefiniowanym elementem typu *use case* można powiązać nie istniejące jeszcze w bazie wymagania – czyli utworzyć nowe wymagania z poziomu programu Rose – zostanie ono automatycznie dodane do bazy wymagań RequisitePro,
- z poziomu Rose można dokonywać modyfikacji wartości atrybutów powiązanych wymagań – zmiany automatycznie są propagowane do bazy danych RequisitePro,
- ze zdefiniowanym elementem typu *use case* można również powiązać jeden z dokumentów znajdujących się w bazie projektu RequisitePro,
- z wybranym z poziomu RequisitePro wymaganiem można powiązać jeden z istniejących w modelu systemu element typu *use case*,

### Ograniczenia integracji:

- wymagania mogą być powiązane jedynie z elementami typu *use case*,
- z jednym elementem typu *use case* może zostać powiązane tylko jedno wymaganie,
- nazwa elementy *use case* musi być taka sama jak opis dotyczący wymagania, które chcemy z nim powiązać,

### Inne funkcje:

Dyskusje (*discussions*) są czymś w rodzaju listy dyskusyjnej i służą do wspomagania pracy grupowej. Do każdego z wymagań uczestnicy projektu mogą dołączać swoje spostrzeżenia lub zapytania i oczekiwać na odpowiedzi lub wskazówki. Dzięki temu możliwa jest komunikacja pomiędzy członkami zespołu projektowego bez konieczności bezpośredniego kontaktu.

Metryki (*metrics*) umożliwiają przekazywanie do RequisitePro zapytań dotyczących projektu i tworzenie (w programie MsExcell) różnego rodzaju wykresów i tabel ilustrujących odpowiedzi na te zapytania. Przykładem może być wykres będący odpowiedzią na pytanie: „Ile w projekcie zdefiniowano wymagań o poszczególnych priorytetach?”.

RequisitePro posiada również mechanizmy umożliwiające zarządzanie projektem i jego bezpieczeństwem. Istnienie możliwość zabezpieczenia projektu hasłem, jego archiwizacji, stworzenia kont i grup użytkowników i przypisania im odpowiednich praw do przeglądania, usuwania, dodawania i modyfikacji składowych projektu.

Jeżeli użytkownik posiada dodatkowo zainstalowany program Rational SoDa istnieje możliwość generacji z poziomu RequisitePro różnego rodzaju raportów dotyczących zdefiniowanych w projekcie wymagań.

Pełen opis możliwości programu RequisitePro znajduje się w Rational Requisite Pro User's Guide..