Слайд 1.

Добрый день. Тема моей работы: «Разработка расширения для Jira для планирования и контроля работ».

На сегодняшний день система управления проектами Jira остается одной из наиболее популярных. Однако она не располагает средством для контроля большого потока создаваемых задач при выполнении работы, запланированной на некоторый длительный период, например, полгода. Поэтому возникла идея создания для этой цели расширения.

Слайд 2.

Была реализована часть плагина, позволяющая создавать планы, содержащие многоуровневую иерархию задач. Целью моей работы является расширение функциональности плагина, а именно разработка перечисленных на слайде функций. Данная функциональность позволит отслеживать возникающие ошибки плана, например, отчеты сотрудников в задачи вне плана, образование циклов в дереве, превышения плановых оценок и др., выявлять изменения, вызвавшие ошибки, …

Слайд 3.

… и упростить работу с задачами плана.

Кроме того, приложение должно поддерживать локализацию интерфейса на русском и английском языке.

Слайд 4.

На основе наличия необходимой функциональности был проанализирован рынок существующих решений. На слайде представлены сделанные в результате выводы. Подробный анализ отражен в разделе 2.1 моей работы.

Слайд 5.

Приложение построено на базе шаблона MVC, структура которого представлена на слайде.

Слайд 6.

Задача реализации работы со снимками деревьев задач делится на создание снимков вручную и автоматически. При этом, для экономии физического пространства должно выполняться автоматическое удаление снимков. На слайде схематически изображено, какие снимки должны оставаться, а именно ежедневные за последнюю неделю, еженедельные за последний месяц и ежемесячные за остальной период плана.

Слайд 7.

Здесь перечислены использованные для реализации средства.

Слайд 8.

Требования к аппаратному и программному обеспечению представлены на слайде.

Слайд 9.

Рассмотрим пользовательский интерфейс основных функций.

В режиме сравнения на странице расположены выпадающие списки для выбора снимков и одного или нескольких критериев сравнения. В первом списке по умолчанию выбран пункт «Текущее состояние», т.к. часто требуется сравнить текущее состояние дерева со снимком, не создавая для этого отдельный снимок. Во втором списке по умолчанию, устанавливается последний снимок. Сравнение запускается по нажатию на кнопку «Сравнить».

Результат выводится следующим образом: зеленым цветом отмечаются новые задачи, красным – удаленные. Оранжевым цветом отмечаются все изменения, при наведении выводится старое значение. Для каждой задачи выводятся иконки для обозначения наличия добавлений, удалений или изменений в ее поддереве..

Слайд 10.

Поиск и вывод последних измененных задач осуществляется при наведении на иконку часов.

Слайд 11.

На данном слайде показан пример использования JQL-функции поиска задач плана или его части, т.е. поддерева. В данном случае для функции были заданы название плана и ключ задачи. Предупреждающее сообщение информирует пользователя о том, что он не задал номер версии плана и поиск по умолчанию был выполнен для актуальной версии.

Слайд 12.

Для вывода ошибок плана сделана отдельная вкладка. Ошибки группируются по типам.

Слайд 13.

На следующем слайде показан образец электронного письма с ошибками.

Слайд 14.

На слайде представлена диаграмма пакетов.

* Пакет «entity» содержит интерфейсы моделей.
* Работа с базой данных реализована в пакете «dao».
* В пакете“service” реализована вся бизнес логика. Там находится сервис для работы со снимками.
* «job» - содержит классы, выполняющие проверку плана на наличие ошибок, рассылку уведомлений об ошибках и автоматическое создание и удаление снимков по расписанию;
* «treebuilder» - содержит классы для построения дерева задач и вспомогательные методы для работы с деревом;
* «planerrors» - содержит классы, представляющие собой ошибки плана;

Слайд 15.

На слайде изображена логическая модель данных снимков планов. Снимки принадлежат версиям планов и содержат набор задач. Задача снимка связана с родительской.

Слайд 16.

Рассмотрим блок-схему алгоритма автоматического создания и удаления снимков, запускаемого ежедневно. Для каждого актуального плана выполняется следующее:

* Создается снимок
* Определяется дата создания первого снимка
* Если прошло больше или неделя, то удаляется снимок недельной давности, кроме сделанных в определенный день недели
* Если прошло больше или месяц и текущий день недели совпадает с днем еженедельных снимков, то удаляется снимок, созданный 4 недели назад, если он не первый из еженедельных снимков своего месяца

Слайд 17.

В результате проделанной работы был разработан набор функций, позволяющих:

* Проверять план на наличие различных ошибок
* Автоматически корректировать оценку родительских задач при добавлении новых дочерних
* Создавать снимки дерева задач плана вручную
* Автоматически создавать и удалять снимки

Слайд 18.

* Сравнивать снимки между собой, а также с текущим состоянием плана
* Находить последние измененные задачи поддерева
* Перетаскивать задачи, изменяя связи между ними
* Иметь JQL-функцию поиска задач плана

Слайд 19.

Приложение внедрено в работу компании «РЕЛЭКС» и будет публиковаться на Atlassian Marketplace.