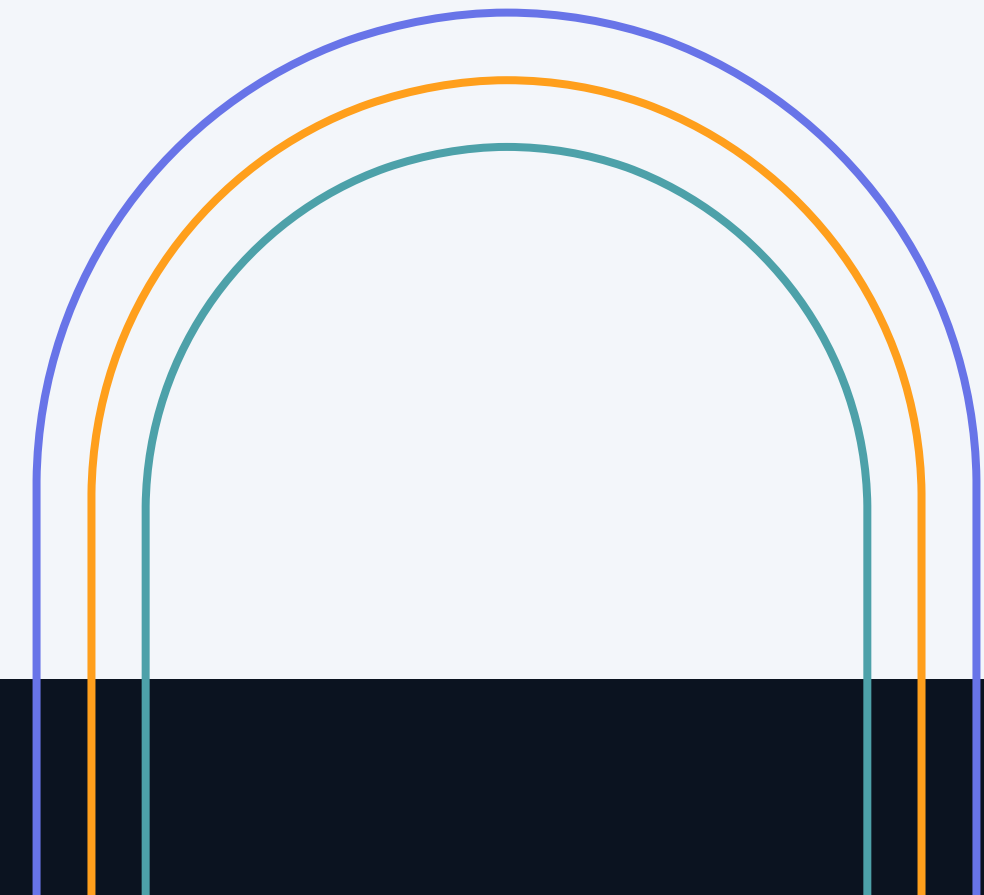




Лекция 6. Системы отслеживания ошибок

- Система отслеживания ошибок
- Баг
- Этимология
- Что такое Agile?
- Отчёты об ошибках для улучшения приложения
- Состав информации о дефекте
- Жизненный цикл дефекта
- Примеры систем отслеживания ошибок



Система отслеживания ошибок. Баг



Система отслеживания ошибок (англ. bug tracking system)

прикладная программа, разработанная с целью помочь разработчикам программного обеспечения (программистам, тестировщикам и прочим) учитывать и контролировать ошибки (баги), найденные в программах, пожелания пользователей, а также следить за процессом устранения этих ошибок и выполнении или невыполнении пожеланий.

Баг

В программировании баг (англ. **bug** — жук) — жаргонное слово, обычно обозначающее ошибку в программе или системе, которая выдаёт неожиданный или неправильный результат.

Большинство багов возникают из-за ошибок, сделанных разработчиками программы в её исходном коде, либо в её дизайне. Также некоторые баги возникают из-за некорректной работы компилятора, синтезирующего некорректный код. Программу, которая содержит большое число багов и/или баги, серьёзно ограничивающие её функциональность, называют нестабильной (англ. unstable, buggy).

Термин «баг» обычно употребляется в отношении ошибок, проявляющих себя на стадии работы программы. *Отчет о проблеме* - отчет, содержащий информацию о баге. *Крэш репорт* (англ. crash report) - отчет о критической проблеме (англ. crash), вызывающей аварийное завершение программы.

«Баги» могут быть обнаружены:

- в процессе тестирования программы (тестировщиками);
- в процессе отладки программы (разработчиками);
- сторонними пользователями приложения.




Этимология

По легенде, 9 сентября 1947 года учёные Гарвардского университета, тестировавшие вычислительную машину Mark II Aiken Relay Calculator, нашли мотылька, застрявшего между контактами электромеханического реле и Грейс Хоппер произнесла этот термин.

Извлечённое насекомое было вклеено скотчем в технический дневник, с сопроводительной надписью: «First actual case of bug being found» (англ. «первый случай обнаружения бага»).

Этот забавный факт положил начало использованию слова «debugging» в значении «отладка программы».



На самом деле слово «bug» в современном значении употреблялось задолго до этого. Так, в течение Второй мировой войны словом «bugs» назывались проблемы с радарной электроникой.

Для понимания того, как команда работает над ошибками, рассмотрим методологию разработки Agile (читается «аджайл»), так как системы отслеживания ошибок используют те же принципы.

Что такое Agile?



Здесь:

- *Список задач продукта (product backlog) — список задач, которые нужно выполнить;*
- *Список задач спринта (sprint backlog) — задачи, которые будут выполнены в ближайшей итерации;*
- *Спринт (sprint) — итерация, по ходу которой (после планирования и до окончания), проходят ежедневные встречи команды, где обсуждается процесс выполнения задач;*
- *обновление системы.*

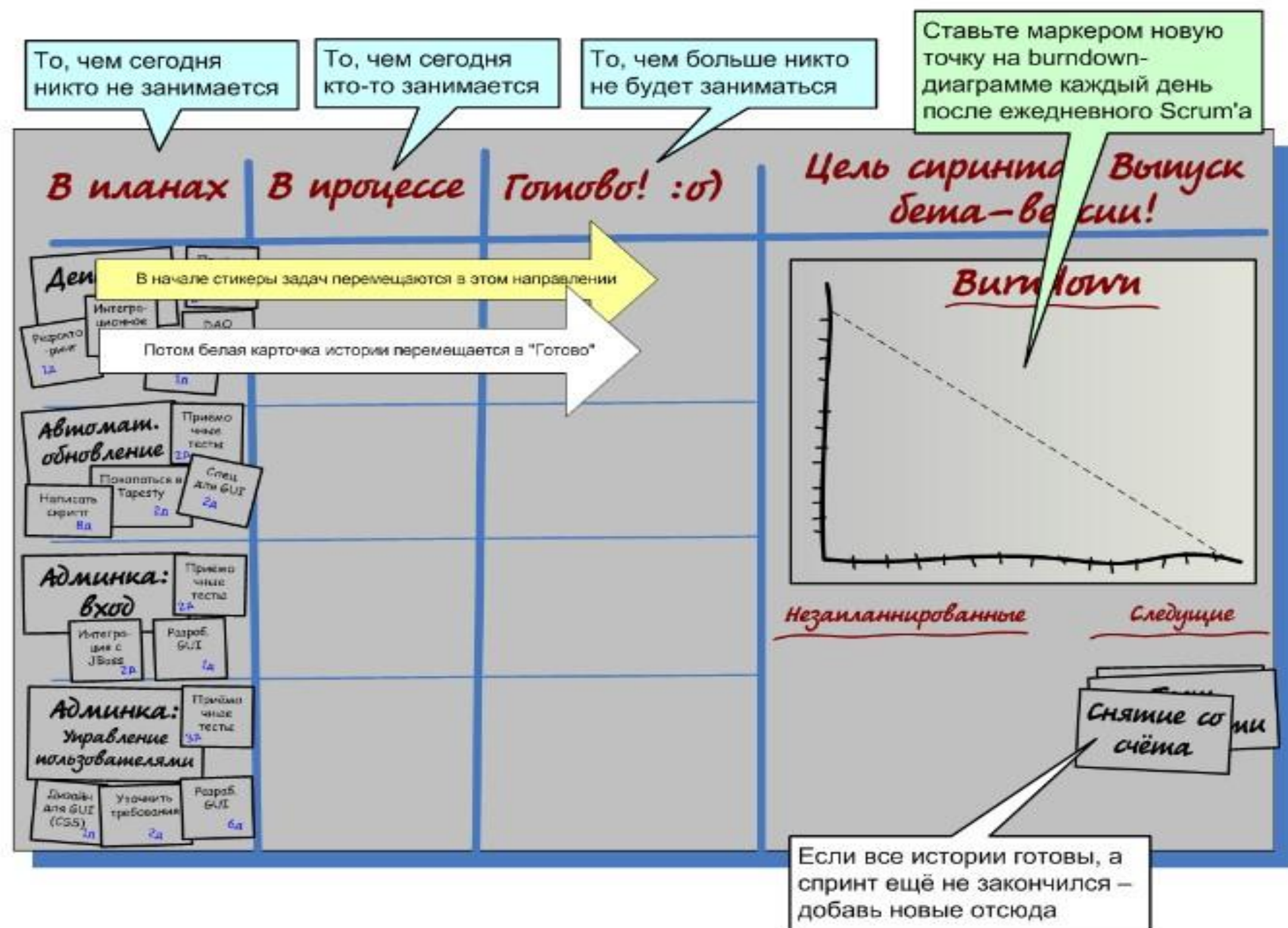
Agile-методы - это методы разработки программного обеспечения, ориентированные на разработку по итерациям.

Существует четыре «завета» гибкой методологии:

- люди и взаимодействие важнее процессов и инструментов (Individuals and interactions over processes and tools);
- работающий продукт важнее исчерпывающей документации (Working software over comprehensive documentation);
- сотрудничество с заказчиком важнее согласования условий контракта (Customer collaboration over contract negotiation);
- готовность к изменениям важнее следования первоначальному плану (Responding to change over following a plan).

Что такое Agile?





Для контроля выполнения задач в Agile используется доска, по которой можно отслеживать процесс выполнения задач.

Доска может иметь много состояний, у каждой команды они называются по-своему, но основные из них три:

- задачи, которые еще не выполняются, но планируются на эту итерацию;
- задачи, которые сейчас разрабатываются;
- задачи, которые уже выполнены и будут выпущены в конце итерации.

Доска для контроля задач

Системы отслеживания ошибок в большинстве своём основываются на том же под-ходе – существует своеобразная доска (обычно в какой – то системе отслеживания багов), на которой появляются все «баги», которые нужно исправить. Их исправляют разработчики / тестировщики / аналитики, беря задачу в работу себе на текущий спринт. Обычно за спринт сотрудник успевает исправить несколько «багов».



Отчеты и состав информации



Отчёты об ошибках для улучшения приложения

Основная масса багов обычно отлаживается на этапе компиляции и тестирования программы. Однако некоторая часть ошибок всё же попадает в релиз и проявляется на компьютерах конечных пользователей в процессе эксплуатации ПО. Для повышения качества программного обеспечения пользуются специальными программами, цель которых — отловить ошибку в целевом приложении, собрать необходимую информацию об её симптомах и отправить отчёт по интернету к разработчикам данного ПО.

Например, в операционную систему Windows встроена утилита Dr. Watson, которая по умолчанию отлавливает ошибки в приложениях пользователя и отправляет отчёт на специальный сервер компании Microsoft. Также в качестве примера можно привести аналогичные библиотеки Breakpad и CrashRpt.

Состав информации о дефекте

Главный компонент системы отслеживания ошибок - база данных, содержащая сведения об обнаруженных дефектах. Эти сведения могут включать в себя:

- номер (идентификатор) дефекта;
- кто сообщил о дефекте;
- дата и время, когда был обнаружен дефект;
- версия продукта, в которой обнаружен дефект;
- серьёзность (критичность) дефекта и приоритет решения;
- описание шагов для выявления дефекта (воспроизведения неправильного поведения программы);
- кто ответственен за устранение дефекта;
- обсуждение возможных решений и их последствий;
- текущее состояние (статус) дефекта;
- версия продукта, в которой дефект исправлен.



Жизненный цикл дефекта

Как правило, система отслеживания ошибок использует тот или иной вариант «жизненного цикла» ошибки, стадия которого определяется текущим состоянием, или статусом, в котором находится ошибка

1

- Новый — дефект зарегистрирован тестировщиком

2

- Назначен — назначен ответственный за исправление дефекта

3

- Разрешён — дефект переходит обратно в сферу ответственности тестировщика. Как правило, сопровождается разрешением ошибки, например:
 - Исправлено (исправления включены в версию такую-то);
 - Дубль (повторяет дефект, уже находящийся в работе);
 - Не исправлено (работает в соответствии со спецификацией, имеет слишком низкий приоритет, исправление отложено до следующей версии и т.п.) ;
 - «У меня всё работает» (запрос дополнительной информации об условиях, в которых дефект проявляется) ;

4

- Далее тестировщик проводит проверку исправления, в зависимости от чего дефект либо снова переходит в статус “Назначен” (если он описан как исправленный, но не исправлен), либо в статус “Закрит”.

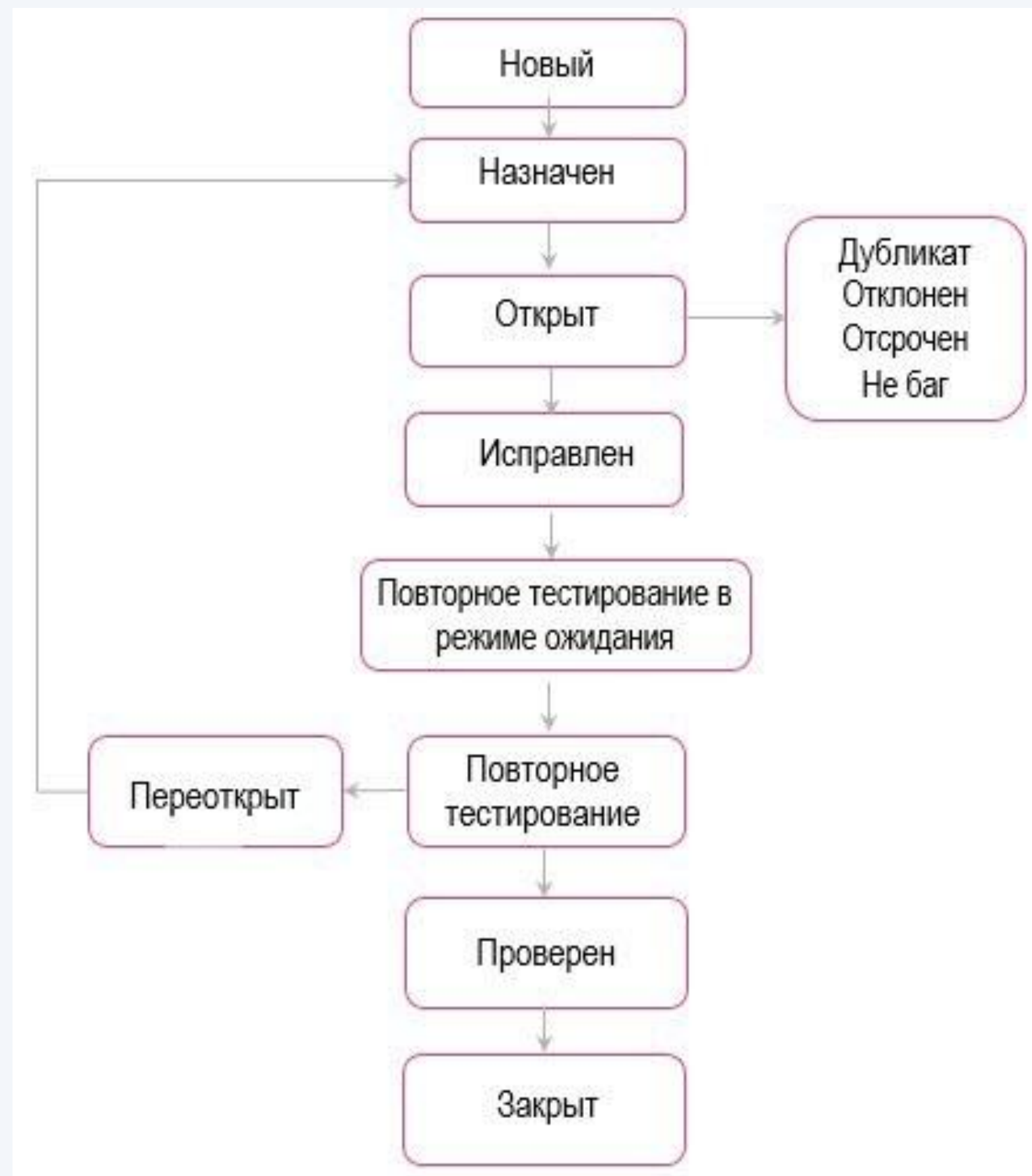
5

- Открыт повторно — дефект вновь найден в другой версии.

«Типичный жизненный цикл дефекта»

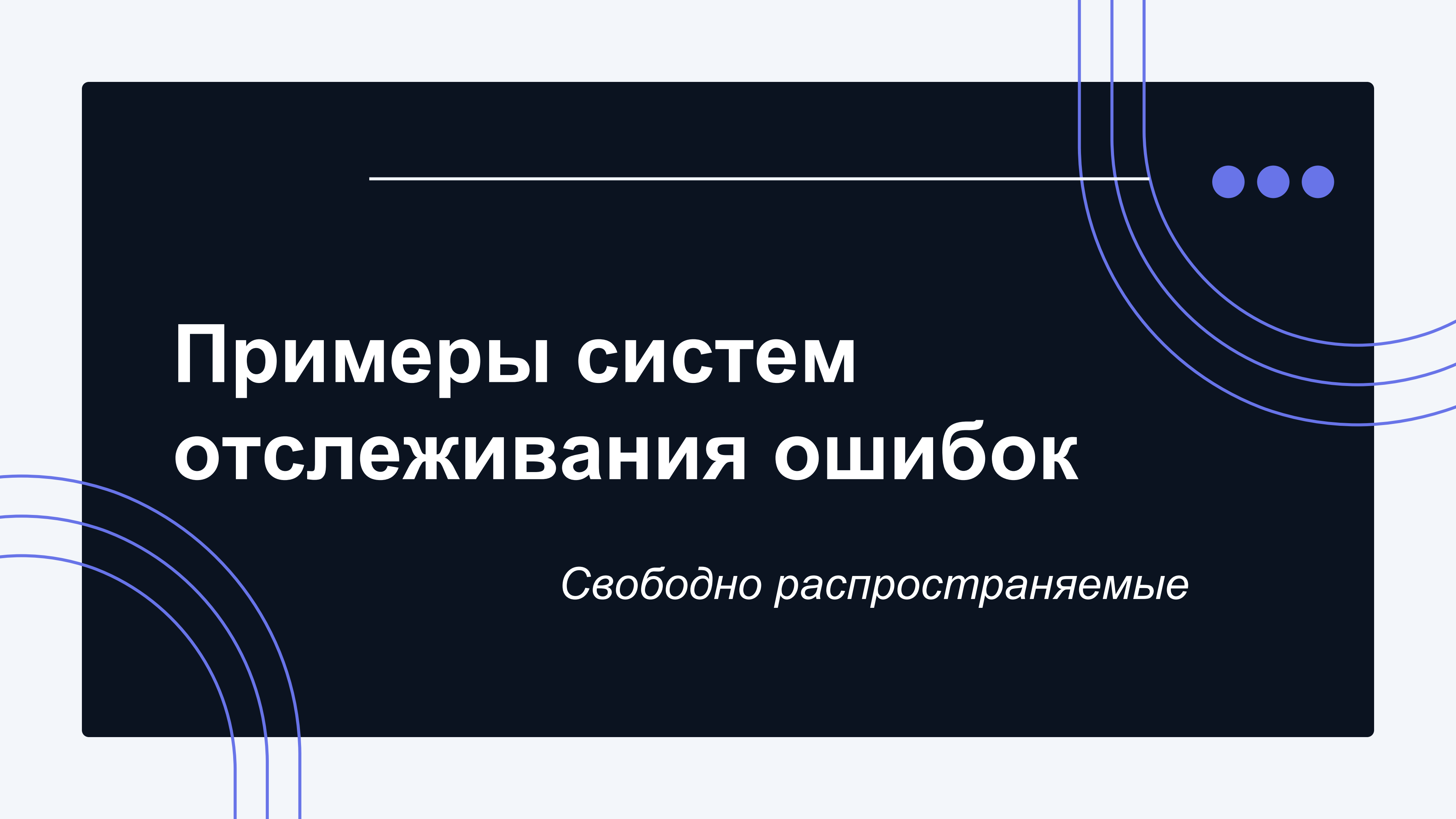
Система может предоставлять администратору возможность настроить, какие пользователи могут просматривать и редактировать ошибки в зависимости от их состояния, переводить их в другое состояние или удалять.

В корпоративной среде, система отслеживания ошибок может использоваться для получения отчётов, показывающих продуктивность программистов при исправлении ошибок. Однако, часто такой подход не даёт достаточно точных результатов, из-за того что разные ошибки имеют различную степень серьёзности и сложности. При этом серьёзность проблемы не имеет прямого отношения к сложности устранения ошибки.



Жизненный цикл дефекта





Примеры систем отслеживания ошибок

Свободно распространяемые

The Bug Genie - это свободное программное обеспечение для управления проектами и отслеживания задач, с веб-интерфейсом. Основная функциональность: это отслежива-ие задач и ошибок, управление проектами, встроенная wiki и интеграция с основными системами контроля версий, такими как Git, Mercurial и SVN.

Главные возможности The Bug Genie включают в себя:

- отслеживание задач;
- управление проектами;
- встроенная wiki;
- интеграция с системами контроля версий;
- настраиваемый рабочий процесс;
- изменяемый интерфейс;
- поддержка RSS;
- консольные инструменты;
- обработка входящей почты и отправка исходящей.

Требования для установки The Bug Genie:

- веб-сервер с возможностью переписывания URL(rewriting);
- PHP версии 5.3.0 или выше;
- MySQL 5.0 или старше или PostgreSQL 8.2 или старше;
- Расширения PHP: gmp, gd (для статистики) и imap (для исходящей почты);



The Bug Genie

Bugzilla — свободная система отслеживания ошибок с веб-интерфейсом.


В 1998 году Bugzilla была выпущена как открытое программное обеспечение компанией Netscape. В настоящее время система разрабатывается «Mozilla Foundation». С одной стороны, Bugzilla довольно проста, с другой стороны, там есть всё, что нужно для ведения ошибок в типичном проекте.

По функциональности Bugzilla сейчас отстаёт от многих современных багтрекеров. Разработчики считают, что одна из причин этого — выбор Perl в качестве языка реализации Bugzilla, рассматривается возможность переписать её на каком-нибудь другом языке программирования.

Для работы Bugzilla требуются:

- веб-сервер с поддержкой CGI (рекомендуется Apache);
- поддержка языка Perl база данных MySQL, PostgreSQL, или Oracle.





Bugzilla

Bugzilla Version 2.19.1+

Bugzilla Bug 305134 **Description** : Remove FeedView from Firefox 1.5 Last modified: 2005-08-28 01:41 PDT

[Search page](#) [Enter new bug](#)

Bug# : 305134 alias:

Product: Firefox OS: All

Component: RSS Discovery and Preview Version: unspecified

Status: RESOLVED Priority: -

Resolution: FIXED Severity: normal

Assigned To: Ben Goodger (use ben at mozilla dot org for email) Target Milestone: -

QA Contact: nobody@mozilla.org

URL:

Summary: Remove FeedView from Firefox 1.5

Status Whiteboard:

Keywords: fixed1.8

Reporter: Ben Goodger (use ben at mozilla dot org for email) <bugs@bengoodger.com>

Add CC:

CC: alex@spamcop.net axel@pike.org bugs.mano@sent.com bugtrap@psychoticwol bugzilla@dougweb.org ☐ Remove selected C

Flags: (Help!) mtschrep: blocking1.8b4rc bugs: blocking1.8b4 blocking1.9a1 blocking-aviary1.0.7 blocking-aviary2.0 testcase

Attachment	Type	Created	Size	Flags	Actions
------------	------	---------	------	-------	---------

Bugzilla

eTraxis — это бесплатная веб-система для отслеживания записей, распространяемая под лицензией GPL v2. Наиболее популярное использование системы — контроль ошибок.

Базовые возможности:

- гибкая настройка жизненного цикла записей (ошибок, улучшений, любых запросов);
- управление правами групп — вплоть до отдельного поля; также есть авторизация через LDAP;
- зависимости между записями;
- сохранение полной истории работы с записями — запоминается любое изменение в любом поле;
- большое количество локализаций;
- нотификации/подписки/напоминания.

New record

Project: eTraxis

Template: Defect report

Subject: [required]

Priority: [required] Medium

Found in version:

Description:

DESCRIPTION

Critical - must resolve in the specified milestone
High - strongly want to resolve in the specified milestone
Medium - normal priority
Low - might slip to later milestone

[b]What steps will reproduce the problem?[/b]
1.
2.
3.

[b]What is the expected output? What do you see instead?[/b]

All fields marked as required should be filled in.
You can insert link to another record by specifying "rec#" and its number (e.g. "rec#305").

OK Cancel



eTraxis

MantisBT (Mantis bug tracking system) - свободно распространяемая система отслеживания ошибок в программных продуктах. Обеспечивает взаимодействие разработчиков с пользователями. Позволяет пользователям заводить сообщения об ошибках и отслеживать дальнейший процесс работы над ними со стороны разработчиков.

Система имеет гибкие возможности конфигурирования, что позволяет настраивать её не только для работы над программными продуктами, но и в качестве системы учёта за-явок для Helpdesk.

Mantis Helpdesk доступен для пользователей по разным подпискам и предоставляет поддержку с пользователями через email, при этом автоматически создавая задачи в вашем баг-трекере.

Система является веб-приложением, поэтому не требует для работы специального ПО и работает через веб-браузер.

Минусы:

- Через веб-интерфейс нельзя произвести существенные изменения настроек. Необходимо настраивать в конфигурации;
- Через интерфейс можно редактировать возможность перехода между статусами, но не список статусов;
- Изменить (добавить, удалить) имеющиеся поля в фильтре, окнах создания и про-смotra бага можно только редактируя код;
- Редактирование набора полей в списке багов возможно только в коде. Но данные операции с кодом достаточно просты и не требуют глубоких знаний программирования на PHP.

Плюсы:

- Бесплатность;
 - Код на PHP свободно модифицируем;
 - Настраиваемые пользователем поля;
 - Скорость работы;
 - Большое количество плагинов, расширяющих функциональность.
- Возможность работать сразу, почти без настройки;
 - Цветовая индикация по статусу бага;
 - Удобные фильтры;
 - Уведомления по e-mail;



3: jamesguru (Guru99 - reporter)2014-11-18 01:23 EST

[My View](#) | [View Issues](#) | [Report Issue](#) | [Change Log](#) | [Roadmap](#) | [Wiki](#) | [IRC Chat](#) | [Repositories](#) | [My Account](#) | [Logout](#)

Enter Report Details

*Category

security

Reproducibility

always

Severity

crash

Priority

immediate

Select Profile

Or Fill In

Platform

JAVA

OS

Windows

OS Version

7

Product Version

1.1.0

*Summary

Due to security reasons, part of your code are blocked:

*Description

Unable to import any-type of library,due to security reason

Steps To Reproduce

Library should import and the content related to library should execute

Additional Information

Upload File (Maximum size: 2,097k)

Browse...

SecurityBug.png

View Status

☒ public ☐ private

Report Stay

☒ check to report more issues

* required

Submit Report

Системные требования:

- веб-сервер (например Apache, IIS и др.);
- поддержка языка PHP;
- база данных (например, MySQL);



MantisBT

Trac — инструмент управления проектами и отслеживания ошибок в программном обеспечении. Trac является открытым программным обеспечением, разработанным и поддерживаемым компанией Edgewall Software.

Trac использует минималистичный веб-интерфейс, основанный на технологии Wiki, и позволяет организовать перекрёстные гиперссылки между базой данных зарегистрирован-ных ошибок, системой управления версиями и вики-страницами. Это даёт возможность использовать Trac в том числе и как веб-интерфейс для доступа к системе контроля вер-сий Subversion, а также, через плагины, к Mercurial, Git, Bazaar и другим.

Поддерживаются базы данных SQLite, PostgreSQL, MySQL и MariaDB. Trac написан на языке программирования Python и в настоящее время распространяется по модифицированной лицензии BSD. В качестве системы HTML-шаблонов веб-интерфейса Trac до версии 0.11 использовал ClearSilver. Новые версии, начиная с 0.11, используют разработанную в Edgewall систему шаблонов Genshi, при этом совместимость с плагинами, использующими ClearSilver, будет оставлена еще в течение нескольких версий.



Create New Ticket

Properties

Summary: test

Reporter:

Description:

B I A

= bug description =

= bug analysis =

= fix recommendation =

Type: defect

Milestone: milestone2

Version:

Cc:

Owner:

Priority: major

Component: component1

Keywords:

Duetime:

☐ I have files to attach to this ticket

Preview

Create ticket

My Template

Create

Delete

Trac

EmForge - это интегрированное решение с открытым исходным кодом, основанное на J2EE, для управления процессом разработки программного обеспечения.

Главное преимущество EmForge — это управление задачами с использованием моделей бизнес-процессов. Любая порожденная задача — экземпляр бизнес-процесса в системе, со всеми необходимыми атрибутами: активностями, уведомлениями, ролями участников и т.д. В проекте есть готовая библиотека процессов, так же возможна разработка процессов под собственные нужды.

Возможности:

- Наличие Wiki-движка. Страницы проекта, описания процессов, комментарии к исполняемым задачам — для всего используется wiki
- Интеграция с репозиториями: EmForge способен взаимодействовать с различными типами репозиториями. В первую очередь, SVN и CVS.
- Интеграция со средствами разработки: реализовано взаимодействие с Eclipse MyLyn и MS Visual Studio.
- Планирование проектов — механизм milestones, , позволяет планировать этапы и отслеживать состояние исполнения этапа разработки.
- Механизм создания отчетов: всевозможные отчеты с использованием Jasper Reports позволят быстро получить статистическую информацию о состоянии проектов



EmForge

Redmine — открытое серверное веб-приложение для управления проектами и отслеживания ошибок. Redmine написан на Ruby и представляет собой приложение на основе широко известного веб-фреймворка Ruby on Rails. Распространяется согласно GNU GPL.

Функциональные возможности:

- ведение нескольких проектов;
- гибкая система доступа, основанная на ролях;
- система отслеживания ошибок;
- вики для каждого проекта;
- ведение новостей проекта, документов и управление файлами;
- форумы для каждого проекта;
- учёт временных затрат;
- лёгкая интеграция с репозиториями (SVN, CVS, Git, Mercurial, Bazaar и Darcs);
- создание записей об ошибках на основе полученных писем;
- многоязыковой интерфейс (в том числе русский);
- оповещение об изменениях с помощью RSS-потоков и электронной почты;
- настраиваемые произвольные поля для инцидентов, временных затрат, проектов и пользователей;
- возможность самостоятельной регистрации новых пользователей;
- поддержка СУБД MySQL, PostgreSQL, SQLite, Oracle

Новая задача

Трекер * Bug

Тема * Test

Описание

B I U S C H1 H2 H3

Test

Статус * New

Приоритет * Normal

Назначена Andreas Bagattini

Версия Test

Файлы

Выбрать файлы

Файл не выбран (Максимальный размер: 100 КБ)

Начата 2018-11-24

Дата выполнения 2018-11-30

Оценка времени 15 час(а,ов)

Готовность 50 %

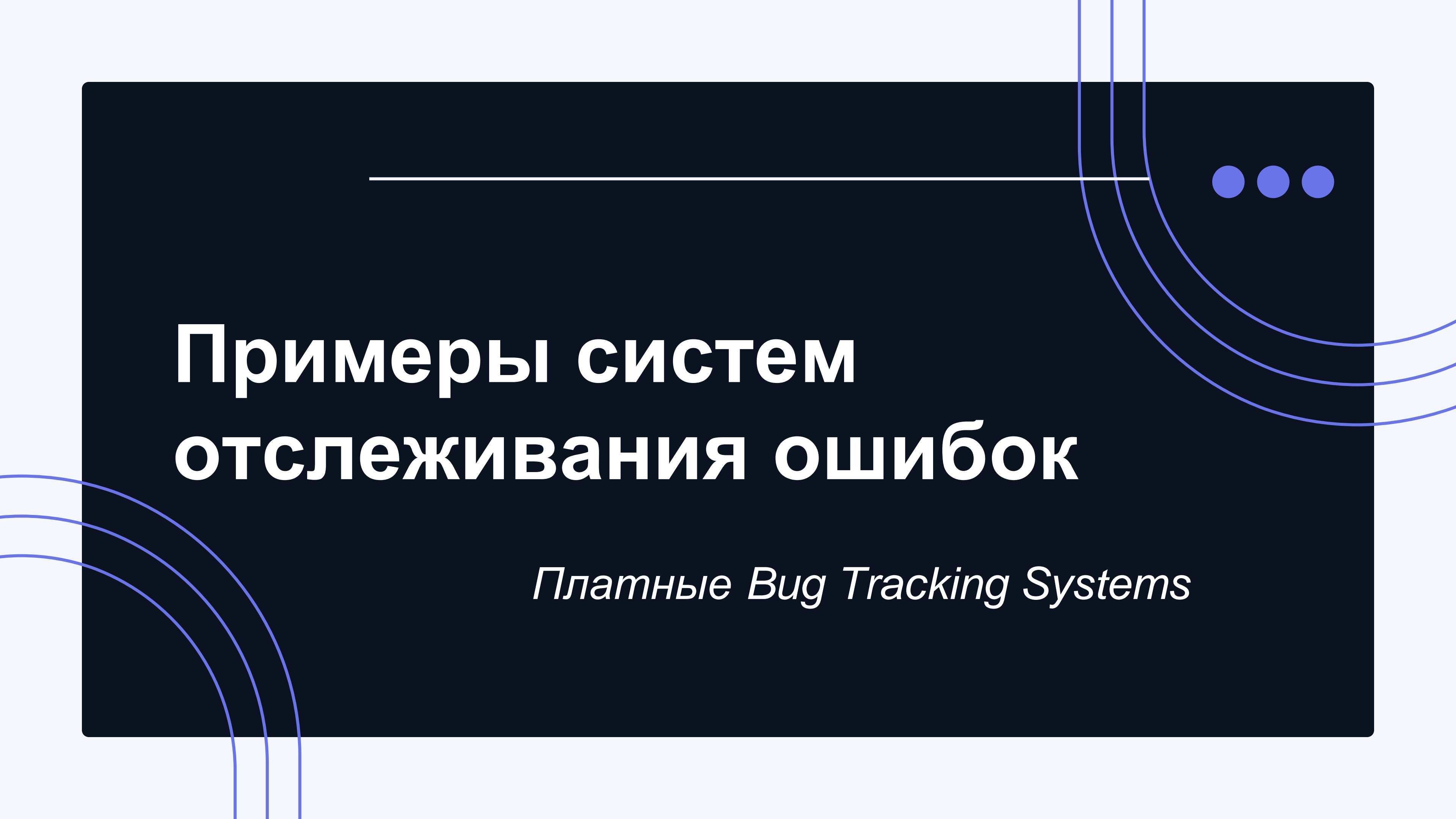
Создать

Создать и продолжить

Предпросмотр



Redmine



Примеры систем отслеживания ошибок

Платные Bug Tracking Systems

Atlassian JIRA — система отслеживания ошибок, предназначена для организации общения с пользователями, хотя в некоторых случаях систему можно использовать для управления проектами. Разработана компанией Atlassian Software Systems. Имеет веб-интерфейс. Название системы (JIRA) было получено путём модификации названия конкурирующего продукта - Bugzilla. JIRA создавалась в качестве замены Bugzilla и во-многом повторяет архитектуру Bugzilla. Система позволяет работать с несколькими проектами. Для каждого из проектов создаёт и ведёт схемы безопасности и схемы оповещения.

Основные детали:

- понятный пользовательский интерфейс;
- хорошая расширяемость;
- управление ошибками, проблемами, задачами.

Плюсы:

- Высокий уровень безопасности;
- Возможность отслеживать проблемы проекта и следить за ходом исполнения каждой задачи,

а также контролировать выполнение поставленных задач любым исполнителем в любой промежуток времени;

- Возможность составить план работы на день, неделю, месяц, а так же планировать своё на решение рабочих задач;
- Отличная система для управления задачами в мультиязычном сегменте;
- Наличие API и множества плагинов, по этому возможности расширений не ограничены.

Существует более 100 готовых бесплатных расширений, есть возможность написания собственных;

- Есть возможность настройки для любой сферы деятельности проекта или всей компании.

Минусы:

- Высокая цена;
- Сложность в настройке и обслуживании.



Summary*

Driver reads array elements less than two incorrectly.

Priority

Major - P3

?

Component/s

API ×

Implementation ×

Start typing to get a list of possible matches or press down to select.

Affects Version/s

3.3.1 ×

Start typing to get a list of possible matches or press down to select.

Environment

Linux Fedora 28 64-bit,
Processor: Intel® Core™ i5-7300HQ CPU @ 2.50GHz × 4.
C driver 1.11.0(debug) also this problem was replicated with version 1.13.0(debug) (manually installed)

For example operating system, software platform and/or hardware specifications (include as appropriate for the issue).

MongoDB Version

v3.6.8

Topology

ReplicaSet

Description

Style ▾

B

I

U

A ▾

↗ ▾

🔗 ▾

☰ ☷

☹ ▾

+ ▾

⌵

Hello, Dear developers!

Driver reads array elements less than two incorrectly.

1. I have this code (which selects array of TEST_CASES from collection):

```
auto collectionName = CollectionCnst::VARIANT;
auto collection = db.collection(collectionName);
auto doc = bsoncxx::v_noabi::builder::basic::document{};
doc = bsoncxx::v_noabi::builder::basic::document{};
doc.append(kvp(ID, bsoncxx::oid(ourVariantId)));
auto cursor = collection.find_one({doc});
if (cursor->view().length() == 0) {
    // not found
    throw std::exception();
}
// select testCase's
auto testCases = cursor->view()(FieldCnst::TEST_CASES).get_array();
return testCases.value;
```

2. I want to iterate through it (foreach loop):

```
for (auto &testCase: testCases) {

    // do smth

}
```

3. If I have 1 element in collection, loop don't invokes no one time. If I have two elements, it invokes 1 time. If I have three or more array elements, it invokes exactly through all elements.

Attachment

Drop files to attach, or browse.

Backwards Compatibility

None

☒ Fully Compatible

Minor Change

Major Change

If this change could break old code

Labels

Bug ×

Begin typing to find and create labels or press down to select a suggested label.

Story Points

Measurement of complexity and/or size of a requirement.

Epic Link

Jira

Bontq — веб-приложение для управления проектами и отслеживания ошибок. Bontq написан на PHP и Java, основным его отличием от конкурентов является кросс-платформенный клиент, который может делать скриншоты и записывать видео для со-ставления визуальных отчётов об ошибках.

Возможности:

- Отслеживание ошибок в ПО;
- Управление проектами и задачами;
- Захват скриншотов и видео через Java клиент;
- Интеграция с Google Docs;
- Импорт данных из Basecamp и FogBugz;



Bontq

YouTrack— коммерческая система отслеживания ошибок, программное обеспечение для управления проектами, разработанное компанией JetBrains. YouTrack поддерживает поисковые запросы, автодополнение, манипуляцию с наборами задач, настройку набора атрибутов задачи, создание пользовательских рабочих процессов и реализует подход, основанный на преимущественном использовании клавиатуры.

YouTrack интегрируется с системами управления версиями, такими как Subversion, CVS, Visual SourceSafe, Mercurial, Git и Perforce с помощью TeamCity. Также имеется возможность интеграции с сервисом GitHub.

Есть возможность интеграции с TeamCity, IntelliJ IDEA, TestLink, TestRail, поддержка аутентификации пользователей с помощью Google Account, Yahoo!, OpenID и LDAP.

YouTrack может импортировать задачи, созданные в других системах отслеживания ошибок: JIRA, FogBugz, Bugzilla, MantisBT, Trac, CSV-файлов и багтрекера Google Code. YouTrack предоставляет библиотеку, написанную на языке Python для импорта задач, созданных в иных системах отслеживания ошибок.

YouTrack



Спасибо!

Жду ваших вопросов...

