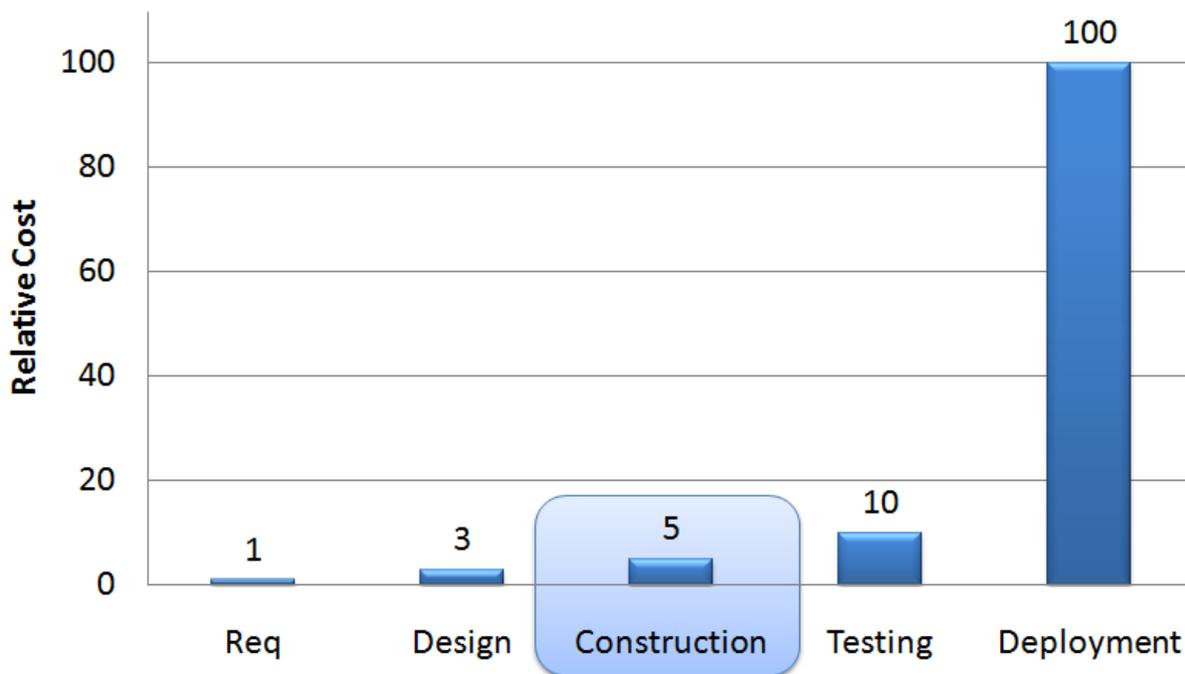


# Code Review Tool

protium

## Introduction

Code review is a systematic examination of source code with an intention to find and fix bugs. Code reviews are primarily done during development phase to improve the quality of the software. The cost of fixing bugs after release to customer is nearly ten times the cost of the fixing it during development. Finding and fixing bugs early in the software development phase can lead to large cost savings.



Relative cost of fixing bugs

Code reviews are done by team members or by an automated source code static analysis tool. Automated static analysis tools can catch simple bugs based on predefined rules and are quite limited in terms of type of rules that can be defined.

Peer code reviews done by team members are more useful in that they can identify much more impactful bugs, like implementation, multi-threading, boundary conditions, algorithmic and logic bugs, which can have higher impact on the software product's quality and user acceptance.

Code review tool allows team members to review the code collaboratively in an easy and efficient manner. It provides all the benefits of formal code inspections and requires considerably less effort and time when compared with manual formal code inspections.

Generally code reviews are done in a collaborative fashion. Code review tools facilitate the code review process by helping author and reviewers to review code effortlessly in a collaborative fashion.

Apart from early identification and removal of defects, code reviews help in increasing shared understanding of code base across team members. This makes team members more agile to work across multiple features of a project.

Code reviews can be an effective mentoring tool for new team members. Code reviews helps in shared learning for team members as they participate in code reviews.

Code reviews fall into two main categories:

1. Formal code reviews
2. Lightweight code reviews

**Formal code reviews** involves a careful and detailed process with multiple participants and multiple phases. Formal code reviews are the traditional method of review, in which reviewers attend a series of meetings and review code line by line, usually using printed copies of the material. Formal inspections are extremely thorough and have been proven effective at finding defects in the code under review. Manual formal code reviews are quite laborious and generally require common time from all the participants of the review. Multiple participants are needed for review to play various different roles like Author, Reader, Tester, Moderator & Scribe etc.

**Lightweight code reviews** typically requires less overhead than formal code inspections, though it can be equally effective when done properly. Lightweight reviews are often conducted as part of the normal development process. Following are the different types of lightweight code reviews:

Review Type	Details
Over-the-shoulder	Reviewer(s) looks over the author's shoulder as the latter walks through the code
Email pass-around	Source code management system emails code to reviewers automatically after or before the check-in is made. Emails are used for discussion threads.

Pair programming	Two developers review the code together as they develop the code together at the same workstation
Tool-assisted code review	Authors and reviewers use specialized tools designed for peer code review

Code Review Tool supports both formal and lightweight code review processes so you can choose the one which works best for your team.

## Code Review Tool Features

Code Review Tool allows developers to do "tool-assisted" code reviews in an easy and efficient manner.

## Easy review management

Code review tool supports both "**post-commit**" and "**pre-commit**" changes for reviews. Post-commit reviews are reviews made up of changes which have already been committed to source control manager. Pre-commit reviews are reviews made up of changes which are present in author's workspace and are not yet committed to source control manager.

Dashboard - Windows Internet Explorer  
<https://codereviewtool.com/DashBoard>  
 Welcome demo [ Log Off ]  
 Home New Review Search Depots Account Feedback About Help  
**Code Review Tool** beta

**Existing Reviews:**

- [Assigned Reviews \(0\)](#)
- [Created Reviews \(4\)](#)
- [All Reviews \(5\)](#)

**Recent Reviews:**  
 [Select All] [Clear All] [Show Details] [Edit] [Delete] [Start] [Cancel] [Close] [Review] [Decline] [Complete]

Title	Status	Author	Threads	Created On	Modified On
<a href="#">'Changes: 'quest-perforce:7752''</a>	In Review	demo	0	8/13/2010 11:50:49 AM	8/13/2010 12:02:50 PM
<a href="#">'Changes: 'quest-perforce:7750''</a>	Completed	demo	0	8/13/2010 11:50:59 AM	8/13/2010 12:02:23 PM
<a href="#">'Changes: 'quest-perforce:7728''</a>	Closed   Completed	demo	0	8/13/2010 11:51:47 AM	8/13/2010 12:01:43 PM
<a href="#">'Changes: 'quest-perforce:7735''</a>	Cancelled	demo	0	8/13/2010 11:51:11 AM	8/13/2010 12:01:25 PM
<a href="#">'Changes: 'public-perforce:7660''</a>	In Review	demo	0	8/13/2010 11:51:41 AM	8/13/2010 12:00:02 PM

Saturday, August 21, 2010 12:38:33 PM  
 Done Internet | Protected Mode: On 100%

Code Review Tool Dashboard

Authors can easily choose relevant changelists and create a review. An author can create and manage multiple reviews simultaneously. An author can invite multiple reviewers for a review. A reviewer can participate in multiple reviews simultaneously. The overall state of each review is automatically managed as each participating reviewer completes his review. Email notifications are sent to author and reviewer(s) when review state changes to help them track reviews efficiently using their email inbox.

Dashboard allows users to easily manage all the reviews they started as authors or as reviewers participating in reviews started by others.

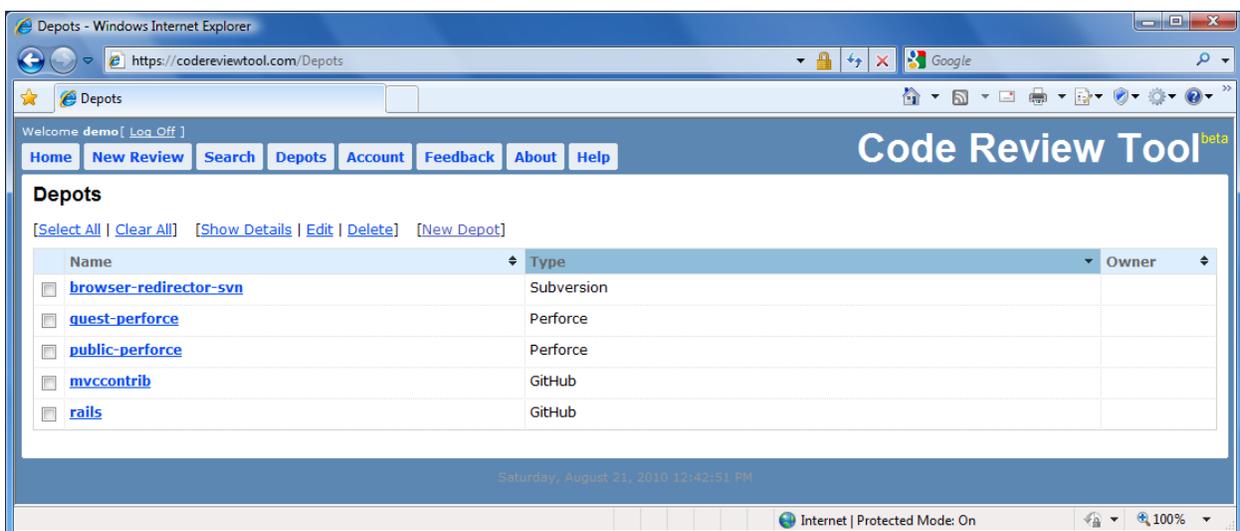
A review can contain one or more changelists (or changesets) from single or multiple different source control managers. An existing review can be updated with newer versions of source files to support iterative review process.

## Multiple source control managers

Code Review Tool supports multiple different types of source control managers. Currently the following source control managers are supported:

1. Perforce
2. Subversion
3. GitHub
4. Team Foundation Server

Support for new types of source control managers can be added easily on request.



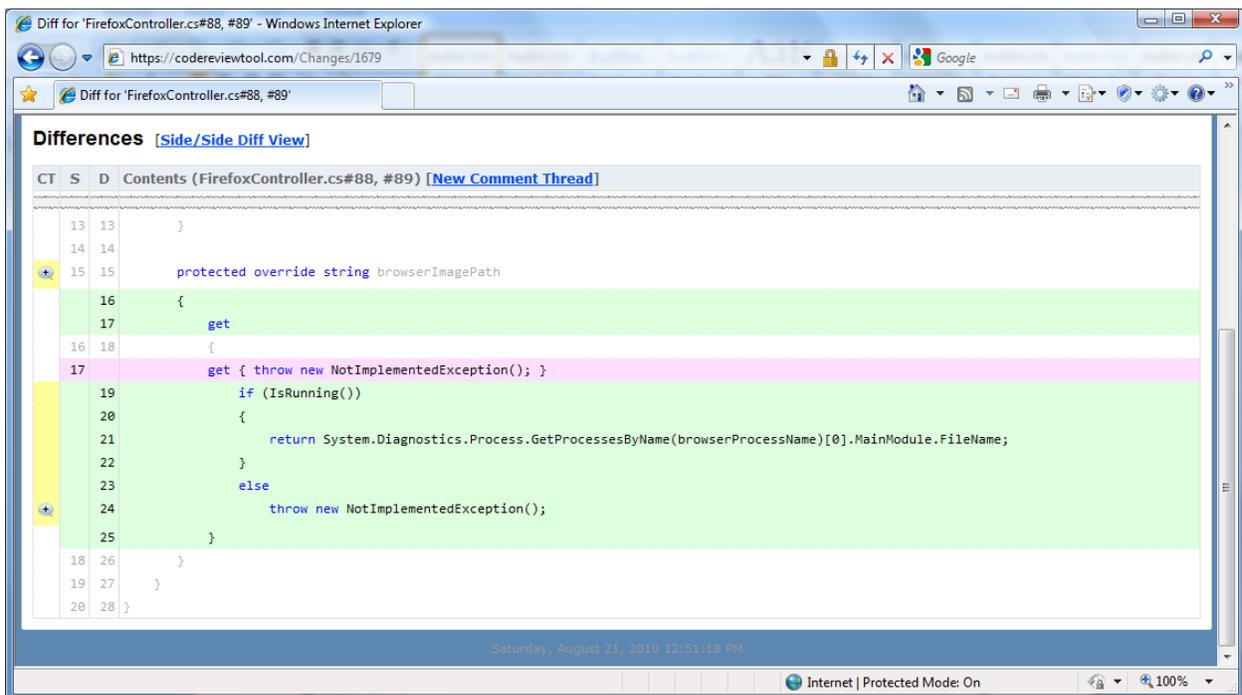
Depots management

**Depots** support the notion of a source code project under a source control manager. Depots allow the author to pick changelists from particular source control manager for the review. Each user can create and manage multiple depots for different types of source control managers.

No installation or configuration is needed on the client machines for post-commit reviews from a new depot. Similarly no installation or configuration is needed on the client machines for post-commit reviews from a new type of depot.

## Discussion threads over source code

Code Review tool allows reviewers to select a range of source code lines and start discussions to identify potential issues with the code. The discussion threads, also called comment threads, from a particular reviewer are visible to other participants once the reviewer completes a review.



Discussion threads in overlaid view

Discussion threads can be in “**open**” or “**closed**” state. Open discussion threads point to potential issues in code. A discussion thread is closed by the creator when the issue has been taken care of.

Author of the review generally takes care of “open” discussion threads by

1. fixing the bugs in the code and updating the review
2. clarifying the issue to the reviewer by replying to the discussion thread

- opening a tracking bug in a bug database and listing the details of the bug in the reply to the discussion thread

There can be multiple discussion threads per review and per changed file in review. Discussion threads can be started over single or multiple lines of source code being reviewed. The participants of a review can manage comment threads in the following different views:

View Type	Details
<b>Overlaid view</b>	All comment threads belonging to a changed file are overlaid over source code diff
<b>Detailed view</b>	Comment thread details are opened in a separate page with snippet of relevant source code diff and discussion details
<b>Filtered view</b>	All comment threads belonging to either a review or a changelist or a changed file are listed in a table with relevant details

As the review contents are edited, the existing comment threads are automatically migrated from old version to new versions of the source files. This feature significantly reduces the effort needed for managing comment threads as source code in review gets updated in multiple iterations of a review.

The screenshot shows the Code Review Tool interface in a Windows Internet Explorer browser. The page title is "Comment Threads for Review 'Changes: 'browser-redirector-svn:89'' - Windows Internet Explorer". The address bar shows the URL: <https://codereviewtool.com/Reviews/307/CommentThreads/Open>. The page content includes a navigation menu with links for Home, New Review, Search, Depots, Account, Feedback, About, and Help. The main heading is "Code Review Tool beta". Below this, the "Review Comment Threads" section displays "Review Details" for a review titled "'Changes: 'browser-redirector-svn:89''". The review details include: Author: demo; Reviewers: reviewer (Status: 'In Review', Threads: 2); Overall Status: 'In Review'; Reviewer Status: 'In Review' Complete; Description: Change 'browser-redirector-svn:89': Added support for the Firefox controller to open things in existing firefox instances.; Created On: 8/21/2010 12:49:23 PM; Modified On: 8/21/2010 12:53:19 PM; Threads: Open (2). Below the review details, the "Comment Threads" section shows a table of threads with columns for Thread, Change, Location, Status, Author, Created On, and Modified On. Two threads are listed: "Why are we throwing" (Change: FirefoxController.cs#88, #89, Location: (18), (19-24), Status: Open, Author: reviewer, Created On: 8/21/2010 12:53:19 PM, Modified On: 8/21/2010 12:53:19 PM) and "Why is this method" (Change: FirefoxController.cs#88, #89, Location: (15), Status: Open, Author: reviewer, Created On: 8/21/2010 12:52:32 PM, Modified On: 8/21/2010 12:52:32 PM). The page footer shows the date and time: "Saturday, August 21, 2010 12:59:58 PM".

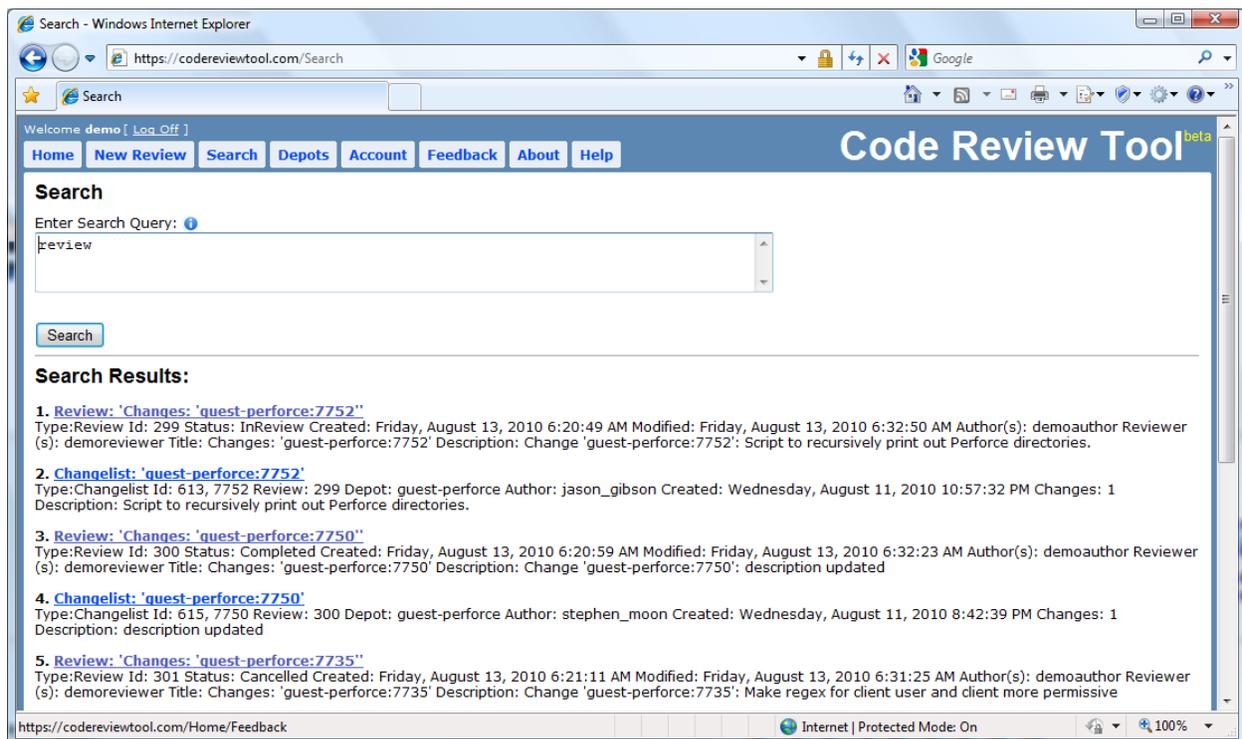
Discussion threads, of a particular review, in filtered view

# Integrated search

Search is a key feature to quickly locate an old review, comment thread or source code which the user wants. The integrated search features allows searching across contents of following types of objects:

1. Reviews
2. Discussion Threads
3. Changelists
4. Source code

The search also allows filtering of results based on particular type of object listed above. The search results contain links to the objects for users to quickly navigate to the object in question.

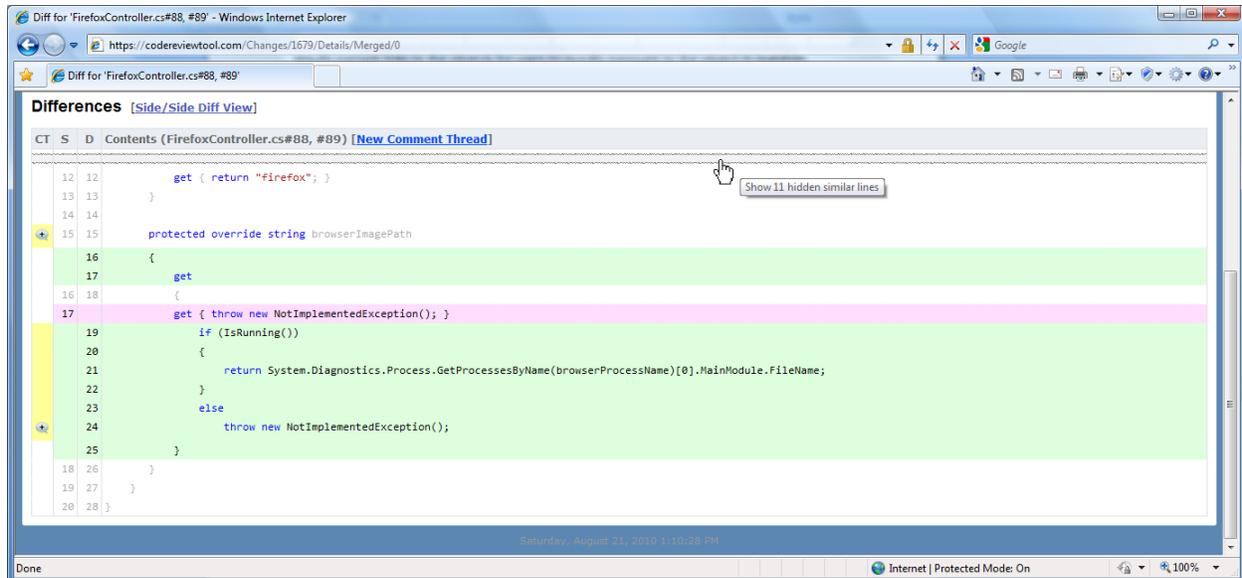


Integrated search results

# Minimalistic user interface

Code Review Tool uses minimalistic UI design so that developers can do code reviews quickly and efficiently. The UI has rich keyboard shortcut bindings for developers to easily invoke all key functionality using keyboard.

Although the Code Review Tool is a web application it follows most of the principles for Web 2.0 applications and looks and feels like a regular desktop application. For e.g. pages are refreshed partially when needed rather than doing a full refresh.



Overlaid diff view with summary differences and syntax highlighting

Since source code diff view is a primary area where most of the review users spent a lot of time, special attention has been paid to make the diff view as easy as possible and flexible. Some of these features are:

Feature	Details
<b>Overlaid diff view</b>	The contents of two versions of the file are overlaid with similar lines appearing only once and added and removed lines appearing separately with different background colors. This view is useful for viewing simple changes between two versions of a file.
<b>Side by side diff view</b>	The two version of the file are shown side by side with removed and added lines appearing in different colors in source and destination version respectively. This view is useful for viewing complex changes between two versions of a file.
<b>Syntax highlighting</b>	The code syntax is highlighted, based on the keywords of the programming language, for easy viewing and understanding of the changes made. All popular programming languages are supported for syntax highlighting.
<b>Summary differences view</b>	Large regions of similar lines are collapsed into a single line summary to easily focus on the differences between the two versions of a file. If needed these summarized sections can be expanded by clicking the summary lines. Summary difference view is supported in both overlaid and side by side difference views.

# Benefits

1. **Easy & efficient** collaborative peer code reviews
2. Supports both **formal and informal** code review processes
3. Identifies, on average, **60% impactful bugs** during development phase
4. Reduces total product development cost by **finding bugs early** that can be fixed before deployment
5. Makes **team members more agile** because of better shared understanding of code base
6. Acts as **effective mentoring tool** for new team members
7. Provides effective review management
  - a. Integrates and works seamlessly with **multiple different source control managers** like **Subversion, Perforce, GitHub & Team Foundation Server**
  - b. Automatic review state management
  - c. **Email notifications** when review state changes with link back to details on relevant review
  - d. **Dashboard** provides quick access to all the reviews
  - e. **No installation or configuration** needed on client machines
8. Provides **easy and efficient management of discussion threads** on source code
  - a. **Easy and intuitive** way of adding comment threads over spanning over single or multiple lines of source code
  - b. **Multiple views**, like overlaid, detailed and filtered view, of comment threads to easily relate discussion threads with source code location and manage them
  - c. **Automatic migration** of comment threads as the source files get updated over the course of time in an iterative review
  - d. **Search across comment threads** to quickly identify relevant comment thread
  - e. Individual comment thread **status tracking**
  - f. Comment threads **grouping by status**
  - g. **Bulk operations**, like changing status & deleting, for comment threads
  - h. Reviewer comments are published to other members of the review only after reviewer completes his review **allowing reviewer to retract his comments** if needed before completing his review
9. **Minimalistic UI design** allows users to do easy and efficient reviews
  - a. **Keyboard bindings** allow developers to quickly complete repeated tasks using his keyboard
  - b. **Different types of diff views**, like overlaid & side by side view, allows the users to easily understand what changed.

- c. **Summary diff view** collapses similar regions of code, which can be expanded on demand, to focus on source code differences quickly and efficiently
  - d. **Syntax highlighting** improves readability of the source code and helps in easy understanding of the differences
10. **Integrated search** allows users to quickly search and locate the object of interest
- a. Integrated **search** works **across Reviews, Comment Threads** and **Source Code**
  - b. Search **results list link to objects** for easy navigation to details
  - c. Searches can be **filtered** based on particular object type like Reviews or Comment Threads or Source Code
11. Available both as **subscription service**, at <https://codereviewtool.com>, or as **software product**, at <http://codereviewtool.com>, which can be deployed inside your company privately.

## Contact

For more information on how Code Review Tool can help you improve your software quality, please contact Protium Software:

**Email:** [info@protiumsoftware.com](mailto:info@protiumsoftware.com)

**Phone:** +91 **9790511082**

**Web:** <http://codereviewtool.com>  
<https://codereviewtool.com>  
<http://protiumsoftware.com>

We will be more than happy to do a detailed demonstration of Code Review Tool at your office premises, just send us a demo request at [info@protiumsoftware.com](mailto:info@protiumsoftware.com)