Project Plan Final Version

Team: May1639

A Web-Based Platform for Java Specifications

CprE/SE 491
Fall, 2015

Advisor & Client: Dr. Hridesh Rajan

Project Members: Arik Coats (Web Master, Key Concept Holder)
Evan Dye (Team Leader)
Robert Kloster (Communications Leader)

Table of Contents

Problem Statement

In the world of software development, specifications inform both humans and computers alike how software is intended to behave, and are highly important to the verification process. Checkable, understandable, useful formal specifications can significantly help minimize the cost of developing software which is secure, assured, and reliable. However, such specifications are not widely available.

Interest in formal specifications, or more precisely behavioral interface specifications, lies primarily in their capability to verify the functionality of software in addition to guaranteeing the behavior of an API. Outside of academia, concern often falls on the observable functionality of code over the explicit guarantees offered by formal specifications. There is little motivation for developers to take the time to write formal specifications for their code, due to the time and difficulty involved. As a result, within an academic setting or any environment that requires strict verification of software, there is a noticeable lack of such specifications for commonly available libraries, and available specifications are not necessarily easily accessible.

Deliverables

Our team is working in conjunction with team May1620 to develop a solution proposed by Rajan et al. This solution proposes the use of data mining to retrieve formal specifications from sites such as Stack Overflow and consolidate them on a web platform.

The primary deliverable for team May1639 is the project's web platform. This platform is a forum that will contain specifications and source code for common Java libraries. It will also facilitate the discussion, creation and refinement of new and existing specifications. Any new specifications may be approved or modified by users of the platform. While serving as an independent platform, the forum will also provide relevant references to the Stack Overflow site to aid in specification creation and editing. The platform will provide an identical API to that of the Stack Exchange network of Q&A sites to allow integration with May1620 and their accessing of Stack Overflow. More detailed functional requirements are outlined below.

A second deliverable is a MySQL database of our design that contains all essential information for supported Java Libraries. This information includes specifications, source code and components of individual classes and methods. This database will be used by the forum to construct the elements of the library, class, and method views, and can be queried by the forum's API.

System Requirements

Requirements

Dr. Rajan has indicated that he will define more requirements for the second semester of development. The requirements already specified, including those to receive primary developmental focus during the first semester, are listed below.

Functional Requirements

- The system will display and traverse a library/package/class/method hierarchy for supported libraries.
- The system will allow for upvoting/favoriting of specifications that are considered correct.
- The system will facilitate creating/editing specifications with manual input.
- The system will allow users to create and contribute to discussions.
- The system will display relevant discussions for currently selected classes/methods.
- The system will display source code for selected classes and methods.
- The system will support additional features as needed to mimic desirable functionality of a Stack Exchange site, with emphasis on Stack Overflow.

Non-Functional Requirements

- The system shall implement the same API calls as Stack Exchange, allowing for back-end programming such as that under development by team May1620 to access information.
- The system shall be extensible, allowing for the addition of new features both during and after development.
- The system shall be maintainable, allowing the project to be appropriately managed in the event control is transferred to other parties.

Currently, the noted NFRs do not have any measurable goals to be met, such as “must be down for no more than 5 minutes” or “must be able to support at least 25 users in an hour”. Our goal is to make a proof-of-concept web platform that is intended to be passed on to future developers. Specific, measurable NFRs detailing requirements such as connection speeds, minimum down time or maximum concurrent users are not entirely in our control and are not in the domain of this project.

Possible Solutions

Our group discussed three possible means of constructing our web-based Q&A discussion platform.

- Create a Q&A platform using an existing internet forums framework
    - There are a variety of available frameworks for the purpose of creating internet forums, including Q&A discussion sites.
    - Notable options include NodeBB and MyBB.
- Construct a new platform from scratch
    - Using a language/framework such as ASP.NET to create our own platform with our necessary functionalily.
- Construct a Stack Exchange site dedicated to formal specifications
    - With permission, it is possible to create a Stack Exchange site dedicated to the discussion of topics distinct enough to separate them from already existing Stack Exchange sites.

Our group has settled on the first option of using the MyBB internet forum framework to create our web-based Q&A discussion platform.

Solution Assessment

There are several reasons that make the first solution the best option. While the second option was brought up, it was immediately considered too involved. It would be too time consuming to learn a framework and build a web-platform to meet the project requirements from scratch. The third option is one that was seriously considered. However, using the Stack Exchange framework may not give the freedom to completely customize our web platform as we may like. So, using a forums framework was determined to be the best option, and we chose MyBB, because it was, and still is, an open-source, free-to-download forum framework with an active and supportive community of developers.

Validation & Acceptance Testing

Proposed requirements tests (working list, high-level abstractions of types of tests):

- Make several calls to the web platform in the same manner as one would make to the Stack Exchange API.
- Traverse the library-->package-->class-->method hierarchy to locate a specific method by finding a specific class or by searching for a particular library.
- Create, then upvote and/or favorite a question to the web platform.
- Create a set of specifications for some code, then edit it, then delete it.
- Create a new discussion, then contribute to it.
- Examine several discussion pages and check if the related discussions are relevant.
- Select a class/method in a commonly used API and confirm that the source code is correctly displayed.

Interface/System Description

System Design

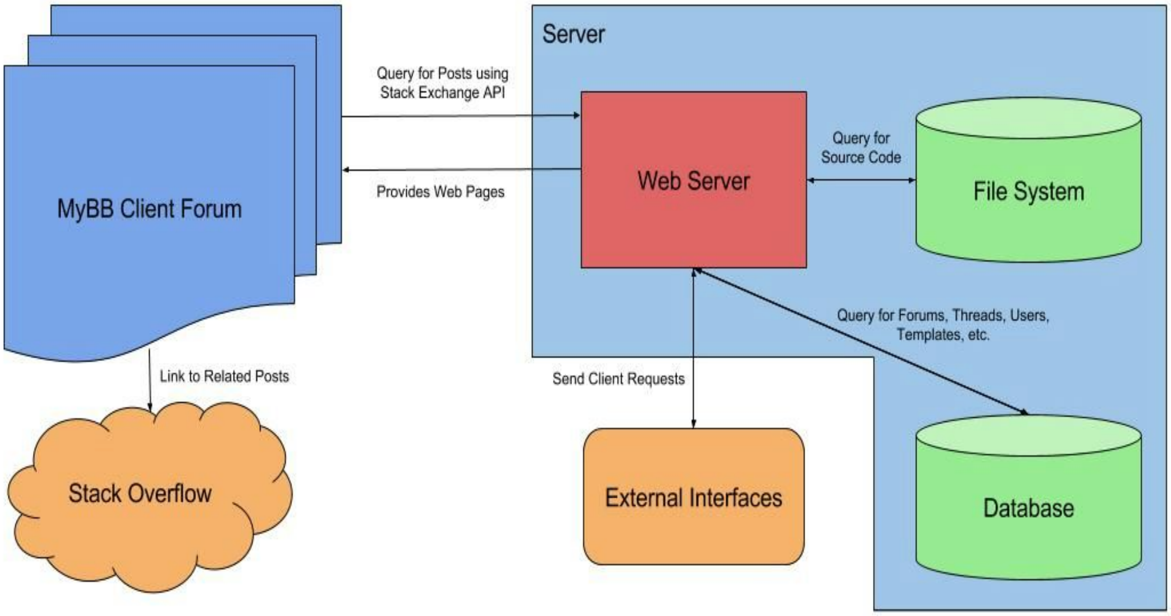

Figure 1: A high-level diagram of the overall structure of the project forum.

The components of the project and their purposes are as follows:

- MyBB Client Forum
    - The front end display of the project, in the form of a question and answer discussion forum created using the MyBB forum software.
- Web Server
    - The intermediate module responsible for assembling the client forum using the information in the project file system and in the project database. In addition, the web server is responsible for performing all interactions with external modules.
- Database
    - A MySQL database that stores information regarding forums, posts, users, and website templates.
- File System
    - The module which stores all resources overseeing the construction and management of the forum. These resources include all PHP files, images, source code files, and specifications.
- Stack Overflow
    - The Stack Exchange website concerning programming. The MyBB client forum provides links to related posts on this site.
- External Interfaces
    - External sources may query the project forums for posts using an identical REST API to that provided by Stack Exchange.

Forum Interface

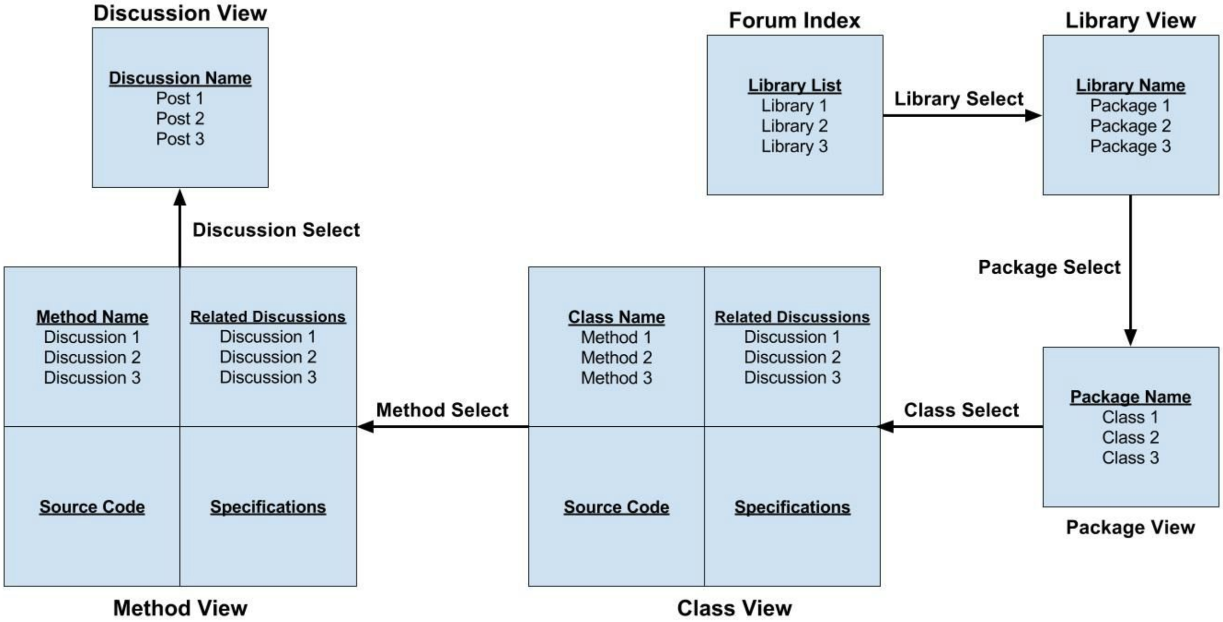

Figure 2: An abstract diagram of the eventual structure of the web platform interface system.

The major components of the forum interface for this project can be broken down as follows:

- Forum Index - The forum homepage.
    - Provides a list of all libraries supported by the forum. Selecting a library navigates to the Library View.
- Library View - Displays relevant information for the currently selected library.
    - Provides a list of all packages in the currently selected library. Selecting a package navigates to the Package View.
- Package View - Displays relevant information for the currently selected package.
    - Provides a list of all classes in the currently selected package. Selecting a class navigates to the Class View.
- Class View - Displays relevant information for the currently selected class.
    - Provides a list of all methods in the currently selected class. Selecting a method navigates to the Method View.
    - Related Discussions - Provides a list of links to Stack Overflow discussions relevant to the currently selected class.
    - Source Code - Displays the source code of this class.
    - Specifications - Displays the currently approved specifications for this class.
- Method View - Displays relevant information for the currently selected method.
    - Provides a list of all discussions concerning the currently selected method. Selecting a discussion navigates to the Discussion View.
    - Related Discussions - Provides a list of links to Stack Overflow discussions relevant to the currently selected method.
    - Source Code - Displays the source code of this method.
    - Specifications - Displays the currently approved specifications for this method.
- Discussion View - Displays the currently selected discussion.
    - Provides a list of all posts in this discussion.

In addition to the user interface described above, we are intending to implement the same REST API calls as Stack Exchange, in order to facilitate a more streamlined integration process with third-party systems.

Work Breakdown Structure

Project Schedule

- Completed for CprE/SE 491 by December, 2015
    - Display and traverse a library/package/class/method hierarchy for supported libraries.
    - Display relevant source code and discussions for currently selected classes/methods.
    - Implement at least partial support for the Stack Exchange API to facilitate later connections to a another module in development by team May1620.
    - Implement a prototype “Related Discussions” feature that will link to relevant discussions on Stack Overflow.
    - Create a database of source code information to support the forum’s pages and API calls.
    - Complete any other requirements necessary for the implementation of the interface outlined in Section 4.
- Completed for CprE/SE 492 by May, 2016
    - Improve and refine the method for determining “Related Discussions.”
    - Finish implementing support for the Stack Exchange API.
    - Implement any other listed requirements not completed for December, 2015.
    - Implement any additional requirements requested by Dr. Rajan for the second semester.
    - Allow for integration with a back-end in development by team May1620.

Risk & Feasibility Assessment

Many aspects of this project have already been proven feasible by work others have done. Creating a Q&A website should not be too difficult, as these types of websites are common, and there are many existing frameworks dedicated to their development. Some aspects that may prove challenging will be implementing a way for users to add, remove and update specifications through our website and automatically generating the forum hierarchies for each supported library. Another potential issue is the security of our platform, since, as our API is to be made public, there is a greater risk of attack. Furthermore, the forum’s security may become a problem as our site has already been visited by two spammers so far. Our team has some learning to do, but overall, this project seems very feasible.

Cost Estimate

There are no expected costs for this project at the moment. As a predominantly software-based project, there are no costs due to hardware. Currently, the plan is to select a freely available forum software as a development platform. We wish to avoid purchasing any development software for this project. There will be no costs due to labor. Overall, the project is presently expected to cost: $0.00.

Previous/Existing Work

There have been similar efforts to increase the availability of formal behavioral specifications, most of which worked through predominantly technical solutions. One focused on analyzing programs by running significantly large numbers of test cases and processing the results to detect patterns and invariants. This approach, however, can only provide, at best, an estimate at how a program behaves, since, in order to learn of how the program behaves universally through this method, we would need test cases for every conceivable possibility in which the program can be called. Such a number of test cases is regrettably infeasible to run, if not impossible to completely generate. Another approach that's been attempted was a program that analyzes another program's code statically. The fruit of such efforts, however, are specifications that apply to that program - and that program alone. These kinds of specifications aren't open enough to easily allow for maintenance or evolution. Still other efforts have focused on tracking the analysis of the APIs of the program they're specifying. By understanding what methods are called, when they are called, and in what order, patterns can arise that can be useful in the generation of specifications. Regrettably, such attempts do not utilize detailed analysis of the program itself enough to directly create specifications.

Conclusion

The direct result of our project will be a web platform to host formal specifications for commonly used Java libraries. In addition, the platform will be of a Q&A format allowing human users to validate and edit existing or computer generated specifications as well as propose and manually create new specifications. The platform is planned for use past the departure of the developing senior design students, and development will proceed with the intention of passing the work on to the community for future upkeep and extension.

Currently, there are no such services that meet the needs outlined above, so in addition to providing a working platform with the described functionality, it will also serve as a proof of concept that such platforms are feasible, ideally prompting further development in the area of formal specifications.

Figures, Tables, and References

Figures

References

1. Rajan, Hridesh, Tien N. Nguyen, Gary T. Leavens, and Robert Dyer. "Inferring Behavioral Specifications from Large-scale Repositories by Leveraging Collective Intelligence." “37th International Conference on Software Engineering: NIER Track” May 2015, ICSE’15, Florence, Italy.