

Quizbowl software design documents

April 29, 2014

Contents

CONTRIBUTORS	v
1 COMPONENTS OF A QUIZBOWL SOFTWARE SUITE	1
2 QUIZBOWL OBJECT MODEL	11
3 TOURNAMENT DIRECTOR USER STORIES	15

Contributors

Please add your name to this file by submitting a pull request.

- Ophir Lifshitz (HFTF)
- Jonah Greenthal (JONAHGREENTHAL)
- Arnav Sastry (ARKNAVE)
- Balachandar ‘Raj’ Kesavan (raj or RAJKSVN)
- Ramapriya Rangaraju (SEPHIROTHRR)
- Gregory Gauthier (RJAGUAR3)
- Jeffrey Hill (JEFFREYMHILL)
- Jim Puls (PULS)
- Alejandro Lopez-Lago (ALOPEZLAGO)
- Harry White (HDWHITE)

Chapter 1

Components of a quizbowl software suite

Introduction

Occasionally, people have suggested organizing a public dialogue about “software in quizbowl” to be scheduled after the mob of national tournaments. To prepare for such a discussion, I thought it would be valuable to get everyone on the same page about the state of software in quizbowl — not just for “software people,” but to help “non-software people” understand how the full scope of quizbowl software works together. It would also be beneficial to the discussion to standardize the terminology used for the different components of the suite.

This document intends to describe the components of a hypothetical quizbowl software suite and the relationships between those components. It will review which components already exist independently, and establish which components are still needed and how they will interface with the already existing ones.

Contents

The diagram below illustrates the flow of data among the components of a complete quizbowl software suite. Click on a component to jump to that section.

components.tikz

1 Tournament database

Description of what a “tournament database” is.

- Stores data on all schools, organizations, people, etc., involved in quizbowl.
 - Generate contact lists for hosts to send out tournament announcements and find willing staffers in the area.
 - Limit public access to sensitive data.
- For a host to announce a tournament, link to their TOURNAMENT DIRECTOR public main page/registration form.
- Teams (and staffers) can search for nearby tournaments.
 - Filter out tournaments that are on the same question set or on the same date as tournaments the team has already played/registered for.
 - Subscribe for notifications when nearby tournaments are announced.

1.1 Existing

HSQB’s tournament database PHP and MySQL.

2 Submission manager

Description of what a “submission manager” is.

- Publish packet due dates and fee structure.
- Automatically list the received packets.
- Enforce who is required to submit packets using TOURNAMENT DATABASE.
- Receive feedback on submitted questions (only after set is clear).
- Security.

For all other packet editing functionality not unique to packet submissions, see TOURNAMENT WRITER. Note that most of the items in this component are tightly integrated with the TOURNAMENT DIRECTOR.

2.1 Existing

None yet exist.

3 Tournament director

Description of what a “tournament director” is.

- Registration (including player rosters)
- Finances (including invoices, possibly e-payment)

- Brackets + seeding
 - Easy rescheduling in case of teams dropping at the last minute
- Publish updates (list of registered teams, building address, tournament date and time, parking amenities, fee structure, contact details)
- Track moderators (room assignments)
- Track equipment (buzzer and laptop assignments)

3.1 Existing

Open-source niche software

“Tournament Director” Node and CouchDB.

4 Tournament writer

Description of what a “tournament writer” is.

- Assign various roles (writer, subject editor, head editor, proofreader, etc.)
- Assign questions to writers (send follow-up emails near due date)
- Comment on questions
- Tag questions
- “Set writer strengths”
- Suggest/claim answers
- Adjust distribution
- *Feng shui* packet randomization (anti-clustering/dithering vs. truly random)
- See overall status/progress chart/timeline
- Security
- Potential “playtesting infrastructure”

4.1 Existing

Closed-source niche software

QEMS I know nothing about this. (See Cody’s screenshot of the MAIN PAGE)

Ginseng I know nothing about this.

Open-source niche software

QED An 8-year-old Perl script.

Garuda A 2-year-old PHP site.

Generic software**Google Docs**

5 Packet reader

Description of what a “packet reader” is.

- Displays a (parsed) packet one question at a time.
- Interfaces with MATCH SCOREKEEPER to:
 - Show the current score
 - Provide buttons to record game events
- Receives information about each match from the TOURNAMENT DIRECTOR, including team and player names
 - Could potentially verify that none of the players wrote a question in the current packet
- Protests (immediate adjudication if resolved in a quicker game room before the slower game rooms get to the question)
- Communication about lunchtime, when and where possible tiebreakers will be held, malfunctioning equipment (buzzers, laptops, timers), other news
- Passwords and progressive unlocking (to minimize the risk of reading the wrong packet)

5.1 Existing

Closed-source niche software**Abacus****Non-software solutions****Printed packets**

6 Match scorekeeper

Description of what a “match scorekeeper” is.

- Record all types of events:
 - Buzz = (Word × Player × Point value)
 - Bonus part = (Team × Point value)
 - Throw out tossup or bonus
 - Substitution = (Player in × Player out)

- Protest
 - Tiebreaker
 - Timeout
- Undo/redo

Could potentially interface with SMART BUZZER SYSTEM to track buzzer races.

6.1 Existing

Closed-source niche software

SQBS

Taft

Open-source niche software

QBSQL

Generic software

Excel

Non-software solutions

Pen and paper

7 Statistics viewer

Description of what a “statistics viewer” is.

- Post stats immediately
 - TOURNAMENT DIRECTOR can observe progress of each game room
- Instant results after each part of the tournament for easy rebracketing
- Instant corrections (during or after tournament)
- Summarize data by:
 - Player
 - Team
 - Tossup, with a histogram of buzz locations
 - Bonus
 - Category
- Queryable API (to easily create rankings or school/team/player reports)
- Could support custom aggregation functions

7.1 Existing

HSQB's SQBS hosting

NAQT's stats database

8 Question repository

Description of what a “question repository” is.

- Notify all teams that played the set after it has been posted (interface with TOURNAMENT DATABASE)
- All questions tagged by subject, difficulty, author
- Incorporate data of all times (tournaments or practice) a question was played
- Report errors
- Permalinking

8.1 Existing

Unparsed questions

HSQB's packet archive

Parsed questions

<https://github.com/neotenic/packets>

9 Search engines

Description of what a “search engine” is.

9.1 Existing

Closed-source service

ACFDB

Open-ish-source service

QBDB

10 Practice tools

Description of what a “practice tool” is.

- Multiplayer and single player

10.1 Existing**Open-source niche software****Protobowl****TriviaBot**

A Potential components

This appendix lists components that, while outside of the scope of a hypothetical quizbowl software suite, still interface with the same data used by the suite.

A.1 Smart buzzer system

Description of what a “smart buzzer system” is.

- It interfaces with the PACKET READER for player names, and interfaces with the MATCH SCOREKEEPER for output.
- Record who got the question completely automatically.
- Keep track of buzzer races.
- Automatically recognize the player and start the 5-second timer.

A.2 Quizbowl-playing robots

Description of what a “quizbowl-playing robot” is.

Chapter 2

Quizbowl object model

Introduction

Text here

Contents

The class diagram below illustrates the objects in a complete quizbowl software suite. Click on a class to jump to that section.

classes.tikz

1 Games

2 Rounds

3 Locations

4 Brackets

5 Stages

6 Divisions

7 Tournaments

8 Events

Polymorphic; see MATCH SCOREKEEPER in the components document.

9 Questions

Polymorphic: Question = Tossup of Words | Bonus of Parts.

10 Packets

11 Sets

12 Formats

13 Players

14 Teams

15 Schools

16 Buzzers

17 Moderators

Chapter 3

Tournament director user stories

1 Primary stories

1.1 Tournament director

- I want to create a tournament
- I want to cancel a tournament
- I want to create a tournament announcement
- I want to change the date/time of a tournament
- I want to change the location of a tournament
- I want to set the fee structure of a tournament
- I want to publicize my tournament to nearby teams
- I want to know how many teams have signed up
- I want to know how many buzzers and moderators there are
- I want to assign discounts for buzzers and moderators
- I want to assign buzzers and moderators to rooms
- I want to send invoices to teams that have registered
- I want to have brackets and schedules generated for me

1.2 Team contact

- I want to register my team for a tournament
- I want to cancel my team's registration
- I want to register a buzzer or moderator for a tournament
- I want to request an invoice
- I want to set/update my team's roster

1.3 All

- I want to communicate with the Tournament director
- I want to see when and where a tournament is
- I want to see what other teams are attending a tournament
- I want to see which players are playing at a tournament
- I want to volunteer as a moderator

2 Packet submission stories

2.1 Tournament director

- I want to accept/confirm or reject a submitted packet
- I want to assign monetary penalties for a packet

2.2 Team contact

- I want to submit/resubmit a packet
- I want to withdraw a submitted packet

2.3 All

- I want to see who submitted a packet and when it was submitted

3 Legend

3.1 Users

Administrator Runs the service

Tournament director Runs the tournament

Team contact Repsonsible for tournament registration

Participant A team member not responsible for registration

Anonymous An individual not related to the tournament

All All of the user types

3.2 Priorities

In order of most important to least important:

- Must-have
- Important
- Moderate
- Reach
- Not supported