

PROJECT SPECIFICATION

Design A Game

API Architecture

CRITERIA	MEETS SPECIFICATIONS
Is the project architected as a <code>Web Service API</code> using <code>Google App Engine</code> ?	Project is architected as a <code>Web Service API</code> using <code>Google App Engine</code> .

API Implementation

CRITERIA	MEETS SPECIFICATIONS
Is a new type of game implemented with additional features?	A new type of game is implemented with additional game logic or features (such as 2-player games). The new game is not a copy of Guess a Number, such as Guess a Date. If it is a guessing game like Hangman, additional features are included (partial reveal of the solution over time).
Are illegal moves handled gracefully?	"Illegal" moves are handled gracefully by the <code>API</code> . For example, if implementing <code>Tic-Tac-Toe</code> , if a <code>User</code> tries to play a square that has already been filled - the server will respond with an error message explaining that the move is illegal. There should be no 'Internal Server Errors' so long as <code>User</code> input is otherwise properly formed.

Resource Containers

CRITERIA	MEETS SPECIFICATIONS
Do <code>endpoints</code> make use of appropriate <code>Resource Containers</code> ?	All <code>endpoints</code> make use of sensible <code>Resource Containers</code> .

New Endpoints Created

CRITERIA	MEETS SPECIFICATIONS
Is <code>get_user_games</code> endpoint is implemented properly?	<code>get_user_games</code> is implemented as specified by Task 3 in the project description.
Is <code>cancel_game</code> endpoint is implemented properly?	<code>cancel_game</code> is implemented as specified by Task 3 in the project description.
Is <code>get_high_scores</code> endpoint is implemented properly?	<code>get_high_scores</code> is implemented as specified by Task 3 in the project description.
Is <code>get_user_rankings</code> endpoint is implemented properly?	<code>get_user_rankings</code> is implemented as specified by Task 3 in the project description.

Is <code>get_game_history</code> endpoint is implemented properly?	<code>get_game_history</code> is implemented as specified by Task 3 in the project description.
--	---

Appropriate use of HTTP Methods

CRITERIA	MEETS SPECIFICATIONS
Do additional <code>endpoints</code> make use of appropriate <code>HTTP methods</code> ?	Additional <code>endpoints</code> make use of appropriate <code>HTTP methods</code> . Meaning <code>GET</code> only reads, and <code>Post</code> writes to <code>Datastore</code> .

Task Queues

CRITERIA	MEETS SPECIFICATIONS
Does the email reminder <code>cronjob</code> handler only notify <code>Users</code> needing a reminder?	The email reminder <code>cronjob</code> handler is modified so that only <code>Users</code> 'needing' a reminder (actual logic up to the student) is modified.

Code Readability

CRITERIA	MEETS SPECIFICATIONS

Are comments present and do they effectively explain longer code procedures?	Comments are present and effectively explain longer code procedures.
--	--

Documentation

CRITERIA	MEETS SPECIFICATIONS
Is the new game documented in a <code>README.md</code> file?	The new game is documented in a <code>README.md</code> file, with explanation of the rules and score-keeping.
Is the <code>API</code> documented in a <code>README.md</code> file?	The <code>API</code> is documented in a <code>README.md</code> file so that users can understand how to use the <code>API</code> without reading the source code.
Did the student reflect on their design decisions in a text file?	The student has meaningfully reflected on their design decisions and recorded their reflections in a text file (preferably named <code>Design.txt</code>).

Suggestions to Make Your Project Stand Out!

- Design a front end for your game.
- Implement `OAuth` so that a user would have to sign in to play the game(you can play around with how you want to decide the permissions a user would have in your game).

[Student FAQ](#)