

Code Challenge Notice, Instructions & Rules

Government Contact: DJ.Griffiths@gov.bc.ca

This notice is dated December 12, 2023 at 9 a.m. (the "Notice Date").

Congratulations - you are a Shortlisted Candidate eligible to participate in the Code Challenge.

Rules and Instructions

Please be advised of the following rules and instructions:

1. These code challenge rules and instructions apply only to Shortlisted Proponents and are part of the Requests for Proposal (RFP).
2. Shortlisted Proponents will have at least two (2) Days from the Notice Date to complete the code challenge. The deadline to complete the code challenge in accordance with these rules is **9:00 a.m. Pacific Time on Thursday, December 14th, 2023** (the "Deadline").
3. The Shortlisted Proponent's code challenge submission Deliverable (defined below) must be received by the province (as provided for by these instructions) and be deposited and located in the applicable Repository before the Deadline, failing which such submission will not be eligible for evaluation and the associated Shortlisted Proponent Proposal will receive no further consideration and such Shortlisted Proponent will be eliminated from the RFP competition.
4. Only the Proponent Resources that were put forward in a Shortlisted Proponent's RFP Proposal are eligible to participate in the Code Challenge.
5. The Shortlisted Proponent Resources will be sent invites via GitHub to join this private repository.
6. As of the Notice Date, the code challenge issue has been created in this private repository, under the "BCDevExchange-CodeChallenge" organization.

7. Shortlisted Proponents may direct clarifying questions by creating an issue in the applicable GitHub Repository. Any such questions must be received by the Government Contact **before 2:00 p.m. Pacific time on Wednesday, December 13^h 2023** (the "Code Challenge Questions Deadline").
8. The Province reserves the right to amend this notice before or after the Closing Time, including changing the Deadline or the Code Challenge Questions Deadline upon notice to all Shortlisted Candidates.
9. The Shortlisted Proponent must complete all the following tasks and the Deliverable and as such they must be deposited and received in the applicable Repository by the Province in the form specified by this notice before the Deadline:
 - a. Complete all code changes required to complete the code challenge (the "Deliverable"); and
 - b. Attach an Apache License 2.0 to Deliverable.
10. The rules and instructions set forth in this notice are in addition to any rules, terms and conditions set forth elsewhere in the RFP.

Deliverable

Introduction

This code challenge asks you to build an online multi-user crossword puzzle, suitable for a small team working together, as a team-building exercise. The basic experience for a single user would be similar to that with any online crossword puzzle, but enhanced with the capability for multiple, identified users to contribute answers.

A feature of common crossword puzzle sites is that only one user can contribute to solving an instance of the puzzle. We would like you to develop a solution where members of a team can each study the puzzle, and contribute answers. As each team member contributes a correct answer, the answer is stored and displayed so that other team members can see it.

Here is an example puzzle grid and clues which may be used for the code challenge. These are free of copyright restrictions.

W	A	I	L	I	N	G								
H		M		C		O								
I	N	B	R	E	E	D								
T		I		F		H								
E	M	B	A	R	G	O								
N		E		E		O								
S	U	S	P	E	N	D								
Across														
1 Complaining bitterly at the wall in the holy city														
3 Concerning in a state of repose can result in birth abnormalities														
5 Prohibition drinking establishment between typographical measure and board game														
7 Hang American caught by disbursement														
Down														
1 Effect of winter on landscape of pale eastern province														
3 Backing large truck containing protective clothing quenches thirst														
5 Navigable in winter, but immigration agency is absent														
7 Service animal up at your stamping grounds, achieving divinity														

High Level Requirements

1. Ability to manage a list of authorized users
2. Ability to display an empty, partially-completed or completed crossword grid
3. Ability to receive a suggested answer, check it, and add it to the grid if it's correct
4. Ability to upload a starting (empty) grid and (hidden) answers, encoded in a text file using a markup convention of the proponent's choice

Technical Requirements

The solution should demonstrate modern best practices, ability to build web applications using modern technology and design approaches.

The solution should be deployed on a publicly accessible website, even behind a password if you'd like (e.g. OpenShift Online, AWS, Heroku). It needs to be available for a week after the day of your submission. Be sure to document how you built and deployed the app.

In addition, your solution must be deployable to a Docker container platform. A single-line CLI command is preferred.

Build and deployment configurations, along with **clear** instructions on how to build and deploy the solution, should be provided, along with a sample puzzle file.

Evaluators will be testing the ability to build and deploy your solution locally with a few, well-documented, commands.

User Stories

All of the following user stories must be completed. They may be completed in any order.

Assumptions

There are 3 personas for this application: Anonymous, Regular, and Administrator

- Anonymous Users may see a welcome page, including the capability to identify themselves using an email address (but no password). Once they have identified themselves, if their email address is in the Administrator's list of Regular users, they may proceed to the
- Regular users should have access to the puzzle, in which they can see the crossword grid and the clues, and they may suggest answers.
- Administrators should have the capabilities of Regular users, plus the ability to upload a grid, clues and correct answers, and the ability to maintain user enrollments.

#1 Sign into the application as an Administrator or Regular user

Given that I'm the holder of the email address which is in the list of enrolled users' email addresses

And I'm in the Landing/Sign-in page

When I supply my email address

Then I am taken to the main puzzle page, where I can view the crossword grid, clues, and any correct answers that have already been supplied. If I am the Administrator (as indicated on the list of email addresses, I am also offered the capability to upload a new grid along with clues and solutions.

Note: There is no requirement to authenticate Regular Users or Administrators in this code challenge.

#2 Offer a candidate answer

Given that I'm a regular user or an administrator

And I am viewing the grid and clues and correct answers that have already been supplied

When I select a clue and enter a candidate answer

Then if my answer is correct:

- It is retained in the grid; and
- The fact that I solved the clue is recorded and displayed alongside the clue.

If my answer is incorrect, I am so informed and the grid is not updated.

#3 Upload a new Puzzle

Given that I'm an Administrator

And I am on the puzzle view

When I click on the "Upload Puzzle" button, and select a file

Then the file containing the puzzle, clues and solutions is uploaded,

And the puzzle view displays an emptied grid and the clues.

#4 Administer the User List

Given that I'm an Administrator

And I am on the puzzle view

When I click on the "Manage Users" button

Then the list of users' email addresses is displayed, along with an indication of which ones are Administrators

And I may edit this list.

And I may not remove the last Administrator

Submission

1. Include all artifacts that are required to build and deploy the solution including code, Dockerfiles, other configuration files, etc.
2. Include all artifacts that would typically be included in the definition of done including demonstration of testing (full test suites are not required, but only a demonstration of how your team would test your solution).
3. Include a README file with instructions for building and deploying your solution. If these instructions are omitted or unclear, points may be deducted.
4. Submit artifacts by way of a pull request to the private repository. All artifacts must be committed before **9:00 a.m. Pacific Time on Thursday, December 14, 2023** (the "Deadline").
5. Attach an Apache License 2.0 to your pull request.

Evaluation Criteria

Criteria	Max Points
Architecture and Technical Design	10
Code Quality and Maintainability	20
Build and Deploy Instructions	5
User Stories – Requirements Satisfied	30
Total	65