

Boston Bruins Developer Applicant Coding Challenge

Boston Bruins Hockey Analytics Department

December 2021

Intro

The purpose of this coding challenge is to give you the opportunity to demonstrate your skills as a programmer and a creative problem solver. The challenge is designed to be open ended and there is no “correct” way to do it. We encourage you to get creative!

You have two days to complete this challenge from the time you download it. Please do not exceed 10 hours of work.

Task

The prompt: Your task is to design a “player card” for evaluating NHL players. The specifics of the design are totally up to you. We have provided you with a simple backend written in Flask/Python with some useful endpoints that mirror the NHL API. You should create your own frontend that communicates with this backend via http requests. The website you design should be able to render a “player card” for anyone in the NHL (and it would be nice if it also allowed for navigation between player cards).

Judging Criteria: We will run your project and interact with it in a browser, so please make sure your code is running properly when you submit it. Your submission will be judged based on the following criteria:

- The quality and design of your UI
- The quality, readability, scalability, and organization of your code
- The depth of the functionality and features in your design
- Your creativity

Project: We’ve supplied you with a project directory, which includes /frontend and /backend subdirectories. Please create a frontend project (using any frontend framework) in the /frontend directory. The /backend directory contains a bare-bones Flask API, which includes a couple of endpoints that use the NHL’s publicly available API to fetch data on players, teams, and games. *We encourage you to edit or extend this API* to fit the needs of your project. You are allowed to use any other data sources as well (APIs, scraped webpages, etc.), just please cite them. See **README.md** for additional details.

When we evaluate your submission, we will run your frontend and your backend on separate development servers (the backend is set up to allow Cross-Origin requests).

Tips: The player card you design might include tools for investigating...

- High level player info
- A breakdown of a player’s key statistics
- Graphical tools for player evaluation

These are just suggestions, remember that solution will be judged on their creativity!

Deliverables:

- Please return your codebase with frontend and backend
- A brief writeup detailing:
 - o Instructions for starting your code (e.g. “install this... npm run start...”)
 - o An explanation of the features of your project, especially those that might not be immediately obvious
 - o Next steps: how would you supplement or change your submission given more time?

Details:

- **Timing:** Up to 10 hours
- **Contact:** cweaver@bostonbruins.com (Don't hesitate to reach out with any questions!)
- **Frontend Frameworks:** We highly recommend using a modern frontend framework such as React/Angular/Vue
- **Backend:** We've provided you with a bare-bones API written in Python/Flask to get you started. We encourage you to edit/extend this to fit your needs.
- **External Code:** You may use any external packages or libraries you like, just make sure you cite them
- **Data sources:** You may use any relevant publicly available data, just make sure to cite your sources.

Submission

When you're finished, please upload your project somewhere (Google Drive, Dropbox, Github, Bitbucket...) and send us a link. (Our corporate attachment filter won't allow you to send us your code directly).