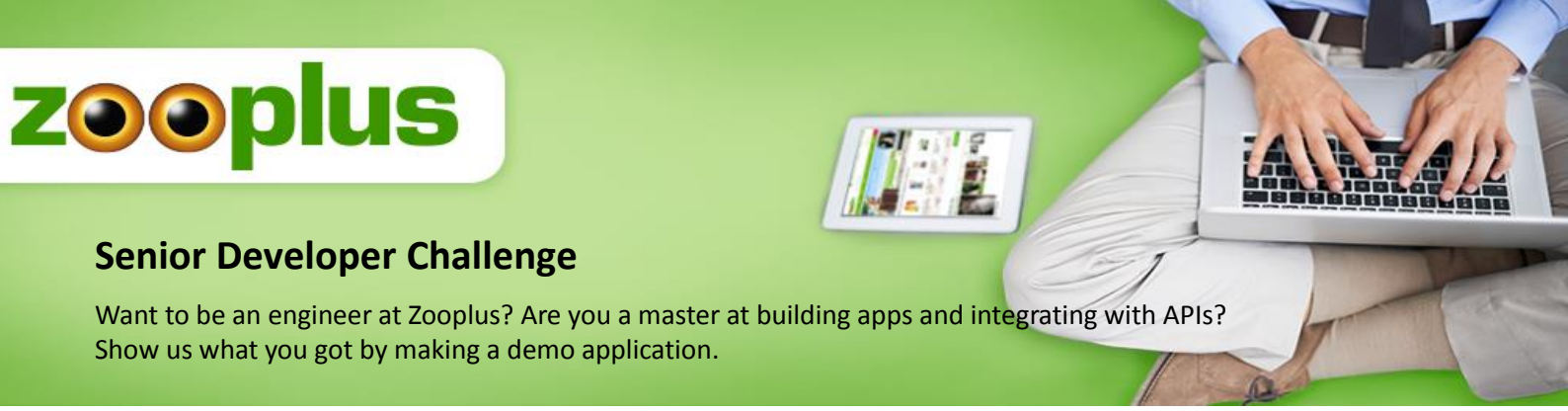

Senior Developer Challenge

Want to be an engineer at Zooplus? Are you a master at building apps and integrating with APIs? Show us what you got by making a demo application.

We suggest you invest as much time as you need to complete the challenge and cover as much as you can of a software delivery lifecycle. If you come up short, that's OK! This is not all or nothing.

Here's what to do:

Develop a protected currency converter application using a public currency converter API (https://openexchangerates.org/signup/free, https://currencylayer.com/documentation, etc...). The application should provide a login/registration screen and a main screen to query historical or current exchange rates. After the successful login the application should show the last 10 queries and their results on the main screen as reminder. Some main currencies (EUR, USD, GBP, NZD, AUD, JPY, HUF, etc...) are enough.

- Create a Java web app. Store your source code in your Github account and provide documentation on how we can build executable war/jar.
- Make your app available via Internet (hint: Heroku, Digital Ocean, Google App Engine or CloudFoundry) and share with us the link.
- You could use in-memory (h2, etc.), cloud database providers (http://mongolab.com, https://www.mongosoup.de/en/index.html) or some RDBMS provider with MySQL hosting for persistence.
- A reliable, working code is a must! We should see a login screen, a registration screen and the main screen with logout feature. Any simple web page with a basic GUI will suffice, i.e. three fields with currencies and the date and a simple list of past entries. Show an application based on some Spring stuff and JPA persistence to make us smile.
- BONUS POINTS:
  - Use Spring MVC for the registration and login forms. The registration should require a valid email address (according to the RFC), a reasonable date of birth, as well as a postal address with street, zip code, city, and a country selectable from a list. **Mandatory if you want to be considered for the Shop dev. team**
  - Implement acceptance tests with a BDD framework like Cucumber or JBehave
  - Provide an automated build and test run on a continuous integration server (it could be hosted on https://travis-ci.org/, https://shippable.com, https://circleci.com/ or something similar, but you can also setup your own Jenkins instance)
  - Implement monitoring and management interface (JMX, REST, etc..)
  - Cache external request with configurable TTL
  - Any stuff that ensures the application can integrate very well with the external components, checks and protects the quality, automates things could be a nice extra.

Feel free to ask questions during the process by emailing developer_challenge@zooplus.com . We'll able to help you quickly if you follow this subject line: (Name) – Integration challenge. Thank you and good luck!