Dear Candidate,

This implementation should be done in C# in an object-oriented manner, and it should follow best practices for C# development, including method and variable naming and error handling. The project contains an existing web service reference that points to a WCF web service, and it includes a working app.config file. The web service can also be accessed directly at <http://40.71.250.33/InterviewService/BankService.svc>

Please implement the following features:

* Retrieve all branches from state code "MO.”
* Write to an xml file the details for all branches located in Springfield
* Validate the routing numbers for each Springfield branch using the following algorithm:
  + Routing numbers should be exactly 9 digits
  + Multiply the first digit by 3, the second digit by 7 and the third digit by 1.
  + Then, multiply the fourth digit by 3, the fifth digit by 7 and the sixth digit by 1.
  + Then, multiply the seventh digit by 3, the eighth digit by 7 and the ninth digit by 1.
  + Add up all of the products. If the sum is evenly divisible by 10 with no remainders, the routing number is valid.
* Write to an xml file the details for all branches that fail validation
* Write to an xml file the details for all branches with a ChangedDate greater than January 1, 2017

The code and the output files should be zipped, encrypted (in order to pass through the corporate firewall), and sent [SVC.FE.Leads@duckcreek.com](mailto:SVC.FE.Leads@duckcreek.com) and [gary.w.burnett@duckcreek.com](mailto:gary.w.burnett@duckcreek.com) If you encounter difficulties in sending the file, try sending only the .cs and .xml files.

A portion of the interview will involve a code review of your project.

The project is available here:

<http://40.71.250.33/InterviewService/DuckCreek.BranchManagerApp-Candidate.zip>

Implementation should take one to two hours.

Thank you,

The Duck Creek Field Engineering Team