# Introduction

Alpenta is a technology company that uses some of the latest middleware solutions to enable our customers to trade in real time. We are building a stable and scalable trading platform. The latest technology tends to be Open Source with a different support model. This is presents both a challenge and a competitive opportunity for Alpenta and our customers.

# Purpose

In order to be productive, we need to recruit a technical team who bring skills in some core areas as well as being capable of developing skills in new areas where there is little traditional support. There is plenty of great information and advice out on the web, but there is also a lot of out of date and misleading advice. This exercise is designed to asses your familiarity with some core technologies as well as to see how you are able to deal with some newer technologies that you probably have not been exposed to. We need to build things, test things and talk to things. And hopefully have some fun at the same time.

This exercise will provide some high level tasks but not detailed instructions. The how is all out there in the web - Google is your friend. We want to see how you pick up new information. There are tips along the way. You shouldn’t need to spend any money on software. Or hardware.

There are no time limits imposed.

# Pre-Requisites

A PC/laptop running your favourite flavour of Linux or OS X. 64-bit is going to make life easier. A free Github account.

# Instructions

1. Create a Red Hat Linux Virtual Machine(VM). RHEL is what we use but any Red Hat flavour will do. We tend to use free VM software but can work with anything.
2. Install what you consider the latest stable version of Fuse-ESB inside your VM. This should be started automatically when your VM boots up. [**Hint** - JBoss Developer site]
3. If you do not already have an Eclipse compatible IDE then download the free Fuse IDE.
4. Create a Maven project that will work with Camel and Fuse.
5. Create a public repository on Github. **Tip** - Push your project regularly. Release early and frequently to your server will help you.
6. Develop a Camel Route that does the following:
   1. Picks up files when placed in an input directory.
   2. Sends the files to our ftp server.
   3. Sends a ticket to our Zendesk ticketing system via the Zendesk REST API when the files have been uploaded. The ticket should contain the names of the files and their md5 checksums. You may want to test against your own free evaluation Zendesk instance, it takes 5 minutes to set one up.
7. When your VM boots up it should automatically deploy the latest version of your code from Github.
8. Don’t forget your tests!

# Deliverables

1. Your public github repository and any documentation or instructions you might have which should be on the README.md or on the Github repository Wiki. The repository should not contain any security credentials. It’s public. You can ftp security credentials file to our ftp server.
2. Your VM image of your working system. Ftp this to our ftp server.
3. Some comments on how the exercise could be improved. Write this in your wiki on github.

# Help

Of course we will help but bear in mind the idea is to see how you cope. If you get stuck or are confused by something the get in touch. You can obtain help by emailing [assessment@alpenta.com](mailto:assessment@alpenta.com).

# Start

When you are ready to start then send a ticket to the Help email address with the URL of your public Github repository. We will then send you the ftp login details and our Zendesk API key. Have fun.