

Developer Candidate Assignment

IMPORTANT: You have 5 working days to finish the three problems. Problem #1 is **MANDATORY**. Problems #2 and #3 are optional.

Please focus on timeliness and completeness - if you have to choose between delivering 100% of the first solution + 0% of the second, and 50%-50%, please choose the first option.

The 5 working days should also cover for things like setting up an IDE such as Visual Studio, in which you can write and run C# code.

How to submit:

1. Add your work to a private git repository on bitbucket.org
2. Grant read access to user 'evision-recruit' to your repository
3. We will review your work and contact you in maximum 1 week

PROBLEM #1

Given the following interface:

```
public interface IAccountService
{
    double GetAccountAmount(int accountId);
}
```

...and the following class that depends on this interface:

```
public class AccountInfo
{
    private readonly int _accountId;
    private readonly IAccountService _accountService;

    public AccountInfo(int accountId, IAccountService accountService)
    {
        _accountId = accountId;
        _accountService = accountService;
    }

    public double Amount { get; private set; }

    public void RefreshAmount()
    {
        Amount = _accountService.GetAccountAmount(_accountId);
    }
}
```

REQUIRED: Write a unit test that asserts the behaviour of `RefreshAmount()` method.

PROBLEM #2

It has been determined that `IAccountService.GetAccountAmount()` is a potentially slow and unreliable remote network call and that it should be made asynchronous. Note that `AccountInfo.RefreshAmount()` *may* be invoked multiple times concurrently. Adjust `IAccountService` and / or `AccountInfo` and your tests accordingly.

PROBLEM #3

Write a build script. Consider compilation, test execution and producing a nuget package.

v5