

# Relatório de Cobertura de Testes

Para a realização de testes unitários da aplicação proposta foi utilizado o xUnit e Mocks. Com auxílio da extensão *Fine Code Covernage* para Visual Studio, foi realizado mapeamento de cada uma das classes quanto a cobertura de testes em seus métodos. Conforme os testes eram sendo implementados, os gráficos e percentuais eram atualizados.

## Uso de Mock

```
1 referência
public class DadosRepositorioTeste
{
    private DadosRepositorio _dadosRepositorio;

    private Mock<IRegistrosRepositorio> _mockRegistrosRepositorio;

    0 referências
    public DadosRepositorioTeste()
    {
        _mockRegistrosRepositorio = new Mock<IRegistrosRepositorio>();

        _dadosRepositorio = new DadosRepositorio(_mockRegistrosRepositorio.Object);
    }
}
```

## Exemplo de Testes Desenvolvidos

```
[Fact]
0 referências
public void AdicionarNovaListaAtivos_RetornandoListaComMesmaQtdDeItensInformada()
{
    _mockRegistrosRepositorio.Setup(x => x.GerarCodigoLetrasNumerosAleatorio()).Returns("AAA1234");
    var qtdItensLista = _dadosRepositorio.AdicionarNovaListaAtivos(5);
    Assert.Equal(5, qtdItensLista.Count());
}

[Fact]
0 referências
public void AdicionarNovaListaAtivos_NaoInformandoQtdDeItens()
{
    var exception = Assert.Throws<Exception>(() => _dadosRepositorio.AdicionarNovaListaAtivos(0));
    Assert.Equal("Nenhuma quantidade informada!", exception.Message);
}
```

## Resultado Final

▼ Name	▼ Covered	▼ Uncovered	▼ Coverable	▼ Total		▲ Line coverage
— SimulacaoBolsaValores.Testes	144	0	144	324	100%	<div></div>
SimulacaoBolsaValores.Testes.Controllers.AtivoControllerTeste	12	0	12	39	100%	<div></div>
SimulacaoBolsaValores.Testes.Services.DadosRepositorioTeste	33	0	33	75	100%	<div></div>
SimulacaoBolsaValores.Testes.Services.RegistrosRepositorioTeste	32	0	32	76	100%	<div></div>
SimulacaoBolsaValores.Testes.ViewModel.InicioViewModelTeste	67	0	67	134	100%	<div></div>
— SimulacaoBolsaValores	215	26	241	527	89.2%	<div></div>
AppExemploMVVM.Services.CommandBase	5	0	5	27	100%	<div></div>
SimulacaoBolsaValores.DataContext.DadosRepositorio	68	0	68	97	100%	<div></div>
SimulacaoBolsaValores.DataContext.RegistrosRepositorio	36	0	36	64	100%	<div></div>
SimulacaoBolsaValores.Model.Entities.AtivoED	17	0	17	30	100%	<div></div>
SimulacaoBolsaValores.ViewModels.InicioViewModel	82	15	97	267	84.5%	<div></div>
SimulacaoBolsaValores.Services.AtivoController	7	11	18	42	38.8%	<div></div>

Conforme mostra o gráfico, 89,2% foi coberto com teste unitários. 10,8% são métodos das classes AtivoController e IniciaViewModel na qual não finalizei suas implementações.