

# Projecten bij het lagenmodel in C# met Visual Studio

---

HA-2271-03

Informaticabeheer  
Bert Cauwenberg en Lieven Pauwels  
Werkgroep Handel  
2018





# Projecten bij het lagenmodel in C# met Visual Studio

---

HA-2271-03

Informaticabeheer  
Bert Cauwenberg en Lieven Pauwels  
Werkgroep Handel  
2018



---

# Inhoud

<b>1</b>	<b>Inleiding.....</b>	<b>5</b>
1.1	Wat houdt het lagenmodel in? .....	5
1.2	Terminologie en afspraken.....	6
1.3	Grotere projecten ontwikkelen .....	7
<b>2</b>	<b>Herbruikbaarheid structureel toepassen.....</b>	<b>9</b>
2.1	Bestaande klassen bij de voorbeelden .....	9
2.2	Het Spel-concept volgens OOP .....	10
<b>3</b>	<b>Two-layer projecten .....</b>	<b>15</b>
3.1	RekeningSaldo .....	17
3.1.0	Projectomschrijving .....	17
3.1.1	Probleemstelling .....	17
3.1.2	Probleemanalyse.....	17
3.1.3	Structuur van de solution .....	18
3.1.4	Presentatie.....	18
3.1.5	Klassen in business laag .....	20
3.1.6	Extra uitleg .....	21
3.1.7	Uitbreidingen .....	23
3.2	VerfCalculator .....	25
3.2.0	Projectomschrijving .....	25
3.2.1	Probleemstelling .....	26
3.2.2	Probleemanalyse.....	27
3.2.3	Structuur van de solution .....	28
3.2.4	Presentatie.....	29
3.2.5	Klassen in business laag .....	30
3.2.6	Extra uitleg .....	33
3.2.7	Uitbreidingen .....	38
3.2.8	Minimalisatie van de hoeveelheid verf .....	40
3.3	Unit Conversion .....	45
3.3.0	Projectomschrijving .....	45
3.3.1	Probleemstelling .....	45
3.3.2	Probleemanalyse.....	46
3.3.3	Structuur van de solution .....	46
3.3.4	Presentatie.....	47
3.3.5	Klassen in business laag .....	48
3.3.6	Extra uitleg .....	49
3.3.7	Extra: Foutafhandeling .....	51
3.3.8	Uitbreiding die gebruik maakt van overerving .....	52
3.3.9	Extra: Webpagina als UI .....	54
3.4	VolleybalCompetition .....	55
3.4.0	Projectomschrijving .....	55
3.4.1	Probleemstelling .....	55
3.4.2	Probleemanalyse.....	56
3.4.3	Structuur van de solution .....	56
3.4.4	Presentatie.....	57
3.4.5	Klassen in business laag .....	58
3.4.6	Extra uitleg .....	58
3.4.7	Aanpassingen aan presentatie (laag).....	63
3.5	VoetbalRanking.....	65
3.5.0	Projectomschrijving .....	65
3.5.1	Probleemstelling .....	65

3.5.2	Probleemanalyse .....	66
3.5.3	Structuur van de solution.....	66
3.5.4	Presentatie .....	67
3.5.5	Klassen in business laag.....	68
3.5.6	Extra uitleg.....	69
3.5.7	Uitbreidingen: Meerdere competities .....	73
3.6	AlphabeticalAntics .....	75
3.6.0	Projectomschrijving .....	75
3.6.1	Probleemstelling .....	75
3.6.2	Probleemanalyse .....	76
3.6.3	Structuur van de solution.....	77
3.6.4	Presentatie .....	77
3.6.5	Klassen in business laag.....	79
3.6.6	Extra uitleg.....	80
3.6.7	Extra oefening .....	82
3.7	Cryptography .....	83
3.7.0	Projectomschrijving .....	83
3.7.1	Probleemstelling .....	83
3.7.2	Probleemanalyse .....	84
3.7.3	Structuur van de solution.....	85
3.7.4	Presentatie .....	85
3.7.5	Klassen in business laag.....	86
3.7.6	Extra uitleg.....	87
3.7.7	Uitbreidingen .....	89
<b>4</b>	<b>Toepassingen met databanken.....</b>	<b>93</b>
4.1	BucketList .....	95
4.1.0	Projectomschrijving .....	95
4.1.1	Probleemstelling .....	95
4.1.2	Probleemanalyse .....	95
4.1.3	Structuur van de solution.....	96
4.1.4	Presentatie .....	96
4.1.5	Klassen in business en persistence laag.....	97
4.1.6	Extra uitleg.....	98
4.1.7	Hoe de connection string initialiseren .....	100
4.2	SplitTheBill .....	101
4.2.0	Projectomschrijving .....	102
4.2.1	Probleemstelling .....	103
4.2.2	Probleemanalyse .....	104
4.2.2.1	Gebruikersinteractie .....	104
4.2.2.2	Databank .....	104
4.2.2.3	Testgegevens in databank .....	105
4.2.3	Structuur van de solution.....	106
4.2.4	Presentatie .....	106
4.2.4.1	StartForm.....	106
4.2.4.2	Vrienden, onkosten en afrekening tonen .....	107
4.2.4.3	Vrienden en onkosten toevoegen .....	108
4.2.4.4	MDI gebruiken .....	109
4.2.5	De vriendenfunctionaliteit implementeren .....	110
4.2.5.1	De klasse Vriend.....	110
4.2.5.2	De klasse VriendRepository .....	110
4.2.5.3	De klasse VriendMapper.....	111
4.2.5.4	De klasse Controller (Persistence) .....	111
4.2.5.5	De klasse Controller (Business) .....	111
4.2.5.6	Aangaande de presentatie.....	112
4.2.6	De onkostenfunctionaliteit implementeren .....	113
4.2.6.1	De klasse Onkost .....	113
4.2.6.2	De klasse OnkostRepository .....	113
4.2.6.3	De klasse OnkostMapper .....	114

4.2.6.4	De klasse VriendOnkostMapper .....	114
4.2.6.5	De klasse Controller (Persistence) - aanvulling .....	114
4.2.6.6	De klasse Controller (Business) - aanvulling .....	115
4.2.6.7	Aangaande de presentatie .....	115
4.2.7	Het afrekeningoverzicht implementeren .....	116
4.2.7.0	Bijkomende probleemanalyse .....	116
4.2.7.1	De klasse OnkostOverzicht .....	116
4.2.7.2	De klasse Vriend - aanvulling .....	118
4.2.7.3	De klasse VriendRepository - aanvulling .....	118
4.2.7.4	De klasse OnkostRepository - aanvulling .....	119
4.2.7.5	De klasse Controller (Business) - aanvulling .....	119
4.2.7.6	Aangaande de presentatie .....	119
4.2.8	Uitbreiding: de eindafrekening uitrekenen .....	120
4.2.8.1	Probleemstelling .....	120
4.2.8.2	Probleemanalyse .....	120
4.2.8.3	De klasse AfrekeningInvoerItem .....	121
4.2.8.4	De klasse AfrekeningInvoerItemComparer .....	121
4.2.8.5	De klasse EindAfrekening .....	121
4.2.8.6	De klasse AfrekeningUitvoerItem .....	122
4.2.8.7	De klasse AfrekeningUitvoerItemComparer .....	122
4.2.8.8	Aangaande de presentatie .....	122
4.2.9	Uitbreiding: mailing .....	123

