

УНИВЕРЗИТЕТ У БЕОГРАДУ
ФАКУЛТЕТ ОРГАНИЗАЦИОНИХ НАУКА

ЗАВРШНИ РАД

**Развој софтверског система за део
пословања цвећаре применом ASP.NET
Core оквира**

Ментор

**Др Саша Лазаревић,
Редовни професор**

Студент

Татјана Баиловић, 0021/2018

Београд, 2022. године

Садржај

1.	УВОД.....	11
2.	ПРЕГЛЕД КОРИШЋЕНИХ ТЕХНОЛОГИЈА.....	12
2.1	<i>.NET</i> платформа.....	12
2.2	<i>.NET Core</i>	13
2.3	<i>ASP.NET Core</i>	13
2.3.1	<i>ASP.NET Core MVC</i> архитектура.....	14
2.4	<i>ENTITY FRAMEWORK CORE</i>	15
2.4.1	<i>Entity Framework Core</i> и <i>LINQ</i> унуту.....	17
2.4.2	<i>Entity Framework Core</i> миграције.....	17
2.5	Софтверски патерни.....	18
2.5.1	<i>DEPENDENCY INJECTION PATTERN</i>	18
2.5.3	<i>SINGLETON PATTERN</i>	18
2.6	<i>JavaScript</i>	19
2.6.1	<i>JQuery</i>	19
2.6.2	<i>AJAX</i>	20
2.7	<i>Bootstrap</i>	20
3	СТУДИЈСКИ ПРИМЕР.....	21
4.	КОРИСНИЧКИ ЗАХТЕВИ.....	23
3.1	Вербални Опис.....	23
3.2	Опис захтева помоћу модела случаја коришћења.....	24
	<i>СК1: Случај коришћења – Приказивање детаља цвета</i>	26
	<i>СК2: Случај коришћења – Приказивање детаља цветног аранжмана</i>	27
	<i>СК3 : Случај коришћења – Претраживање наруџбина</i>	28
	<i>СК4 : Случај коришћења – Пријављивање на систем</i>	29
	<i>СК5 : Случај коришћења – Измена наруџбине</i>	30
	<i>СК6 : Случај коришћења – Креирање цветног аранжмана</i>	31
	<i>СК7 : Случај коришћења – Креирање корисничког налога</i>	32
	<i>СК8 : Случај коришћења – Брисање наруџбине</i>	33
	<i>СК9 : Случај коришћења – Измена корисничког налога</i>	34
	<i>СК10 : Случај коришћења – Креирање цвета</i>	35
	<i>СК11 : Случај коришћења – Креирање паковања</i>	36

<i>СК12 : Случај коришћења – Одјава са система.....</i>	37
<i>СК13 : Случај коришћења – Креирање наруџбине</i>	38
5. АНАЛИЗА.....	39
5.1 Понашање софтверског система – Системски дијаграми секвенци	39
<i>СК1: Случај коришћења – Приказивање детаља цвета</i>	40
<i>СК2: Случај коришћења – Приказивање детаља цветног аранжмана.....</i>	42
<i>СК3 : Случај коришћења – Претраживање наруџбина.....</i>	44
<i>СК4 : Случај коришћења – Пријављивање на систем</i>	46
<i>СК5 : Случај коришћења – Измена наруџбине.....</i>	48
<i>СК6 : Случај коришћења – Креирање цветног аранжмана.....</i>	50
<i>СК7 : Случај коришћења – Креирање корисничког налога.....</i>	52
<i>СК8 : Случај коришћења – Брисање наруџбине.....</i>	54
<i>СК9 : Случај коришћења – Измена корисничког налога</i>	56
<i>СК10 : Случај коришћења – Креирање цвета</i>	58
<i>СК11 : Случај коришћења – Креирање паковања</i>	60
<i>СК12 : Случај коришћења – Одјава са система.....</i>	62
<i>СК13 : Случај коришћења – Креирање наруџбине</i>	64
5.2 Закључак на основу дијаграма секвенци случаја коришћења	66
5.3 Понашање софтверског система – Дефинисање уговора о системским операцијама	67
<i>Уговор УГ1: ПрикажиЦвет</i>	67
<i>Уговор УГ2: ПрикажиНаруџбине</i>	67
<i>Уговор УГ3: ПрикажиЦветниАранжман</i>	67
<i>Уговор УГ4: ВратиЦвеће</i>	67
<i>Уговор УГ5: ВратиЦветнеАранжмане</i>	68
<i>Уговор УГ6: ВратиНаруџбине</i>	68
<i>Уговор УГ7: ВратиПаковања</i>	68
<i>Уговор УГ8: ВратиКупца</i>	68
<i>Уговор УГ9: ВратиПроизводе.....</i>	68
<i>Уговор УГ10: ВратиПродајнаМеста.....</i>	68
<i>Уговор УГ11: ИтзмениНаруџбину.....</i>	69
<i>Уговор УГ12: ИтзмениКупца.....</i>	69
<i>Уговор УГ13: КреирајЦветниАранжман</i>	69

<i>Уговор УГ14: КреирајКупца</i>	69
<i>Уговор УГ15: КреирајЦвет</i>	69
<i>Уговор УГ16: КреирајПаковање</i>	70
<i>Уговор УГ17: КреирајНаруџбину</i>	70
<i>Уговор УГ18: ОбришиНаруџбину</i>	70
<i>Уговор УГ19: Пријава</i>	70
<i>Уговор УГ20: Одјава</i>	70
6. ФАЗА ПРОЈЕКТОВАЊА	71
6.1 Структура софтверског система – Концептуални доменски модел	71
<i>Пројектовање складишта података</i>	72
<i>Пројектовање екранских форми</i>	77
<i>СК1: Случај коришћења – Приказивање детаља цвета</i>	78
<i>СК2: Случај коришћења – Приказивање детаља цветног аранжмана</i>	80
<i>СК3 : Случај коришћења – Претраживање наруџбина</i>	82
<i>СК4 : Случај коришћења – Пријављивање на систем</i>	84
<i>СК5 : Случај коришћења – Измена наруџбине</i>	87
<i>СК6 : Случај коришћења – Креирање цветног аранжмана</i>	90
<i>СК7 : Случај коришћења – Креирање корисничког налога</i>	93
<i>СК8 : Случај коришћења – Брисање наруџбине</i>	96
<i>СК9 : Случај коришћења – Измена корисничког налога</i>	98
<i>СК10 : Случај коришћења – Креирање цвета</i>	101
<i>СК11 : Случај коришћења – Креирање паковања</i>	103
<i>СК12 : Случај коришћења – Одјава са система</i>	105
<i>СК13 : Случај коришћења – Креирање наруџбине</i>	107
<i>Пројектовање апликационе логике</i>	109
<i>Уговор УГ1: ПрикажиЦвет</i>	110
<i>Уговор УГ2: ПрикажиНаруџбине</i>	111
<i>Уговор УГ3: ПрикажиЦветниАранжман</i>	111
<i>Уговор УГ4: ВратиЦвеће</i>	112
<i>Уговор УГ5: ВратиЦветнеАранжмане</i>	112
<i>Уговор УГ6: ВратиНаруџбине</i>	113
<i>Уговор УГ7: ВратиПаковања</i>	113
<i>Уговор УГ8: ВратиКупца</i>	114

<i>Уговор УГ9: ВратиПроизводе.....</i>	114
<i>Уговор УГ10: ВратиПродајнаМеста.....</i>	115
<i>Уговор УГ11: ИтзмениНаруџбину.....</i>	115
<i>Уговор УГ12: ИтзмениКупца.....</i>	116
<i>Уговор УГ13: КреирајЦветниАранжман.....</i>	116
<i>Уговор УГ14: КреирајКупца.....</i>	117
<i>Уговор УГ15: КреирајЦвет.....</i>	117
<i>Уговор УГ16: КреирајПаковање.....</i>	118
<i>Уговор УГ17: КреирајНаруџбину.....</i>	118
<i>Уговор УГ18: ОбришиНаруџбину.....</i>	119
<i>Уговор УГ19: Пријава.....</i>	119
7. ИМПЛЕМЕНТАЦИЈА.....	125
7.1 Имплементација екранских форми.....	125
<i>СК1: Случај коришћења – Приказивање детаља цвета.....</i>	126
<i>СК2: Случај коришћења – Приказивање детаља цветног аранжмана.....</i>	128
<i>СК3 : Случај коришћења – Претраживање наруџбина.....</i>	130
<i>СК4 : Случај коришћења – Пријављивање на систем.....</i>	132
<i>СК5 : Случај коришћења – Измена наруџбине.....</i>	135
<i>СК6 : Случај коришћења – Креирање цветног аранжмана.....</i>	138
<i>СК7 : Случај коришћења – Креирање корисничког налога.....</i>	141
<i>СК8 : Случај коришћења – Брисање наруџбине.....</i>	144
<i>СК9 : Случај коришћења – Измена корисничког налога.....</i>	146
<i>СК10 : Случај коришћења – Креирање цвета.....</i>	149
<i>СК11 : Случај коришћења – Креирање паковања.....</i>	151
<i>СК12 : Случај коришћења – Одјава са система.....</i>	153
<i>СК13 : Случај коришћења – Креирање наруџбине.....</i>	155
7.1 Структура софтверског решења.....	158
7.2 Имплементација апликационе логике.....	160
<i>7.2.1 Имплементација комуникације са корисницима.....</i>	160
<i>7.2.2 Имплементација пословне логике.....</i>	162
<i>Уговор УГ1: ПрикажиЦвет.....</i>	162
<i>Уговор УГ2: ПрикажиНаруџбине.....</i>	163
<i>Уговор УГ3: ПрикажиЦветниАранжман.....</i>	163

<i>Уговор УГ4: ВратиЦвеће</i>	163
<i>Уговор УГ5: ВратиЦветнеАранжмане</i>	163
<i>Уговор УГ6: ВратиНаруџбине</i>	164
<i>Уговор УГ7: ВратиПаковања</i>	164
<i>Уговор УГ8: ВратиКупца</i>	164
<i>Уговор УГ9: ВратиПроизводе</i>	164
<i>Уговор УГ10: ВратиПродајнаМеста</i>	165
<i>Уговор УГ11: ИтзмениНаруџбину</i>	165
<i>Уговор УГ12: ИтзмениКупца</i>	165
<i>Уговор УГ13: КреирајЦветниАранжман</i>	166
<i>Уговор УГ14: КреирајКупца</i>	166
<i>Уговор УГ15: КреирајЦвет</i>	166
<i>Уговор УГ16: КреирајПаковање</i>	167
<i>Уговор УГ17: КреирајНаруџбину</i>	167
<i>Уговор УГ18: ОбришиНаруџбину</i>	167
<i>Уговор УГ19: Пријава</i>	168
<i>Уговор УГ20: Одјава</i>	168
7.3 Имплементација слоја приступа подацима	169
7.3.2 Слој приступа подацима.....	170
7.5 Имплементација презентационог слоја	172
7.5 Имплементација складишта података.....	174
8. ТЕСТИРАЊЕ	178
9. ЗАКЉУЧАК.....	179
10. ЛИТЕРАТУРА	180

Списак слика

Слика 1 Приказ NET. Платформе [6]	12
Слика 2 Употреба језика C# и .NET платформе за изградњу веб сајтова [7]	13
Слика 3 Model–view–controller архитектура [8]	14
Слика 4 Различити приступи употребе Entity Framework Core-а [9].....	16
Слика 5 Случајеви коришћења на страни купца	24
Слика 6 Случајеви коришћења на страни Администратора	25
Слика 7 СК1 - основни сценарио, дијаграми секвенци	40

Слика 8 СК1 - алтернативни сценарио, дијаграми секвенци	41
Слика 9 СК2 - основни сценарио, дијаграми секвенци	42
Слика 10 СК2 - алтернативни сценарио, дијаграми секвенци	43
Слика 11 СК3 - основни сценарио, дијаграми секвенци	44
Слика 12 СК3 - алтернативни сценарио, дијаграми секвенци	45
Слика 13 СК4 - основни сценарио, дијаграми секвенци	46
Слика 14 СК4 - алтернативни сценарио, дијаграми секвенци	47
Слика 15 СК5 - основни сценарио, дијаграми секвенци	48
Слика 16 СК5 - алтернативни сценарио, дијаграми секвенци	49
Слика 17 СК6 - основни сценарио, дијаграми секвенци	50
Слика 18 СК6 - алтернативни сценарио, дијаграми секвенци	51
Слика 19 СК7 - основни сценарио, дијаграми секвенци	52
Слика 20 СК7 - алтернативни сценарио, дијаграми секвенци	53
Слика 21 СК8 - основни сценарио, дијаграми секвенци	54
Слика 22 СК8 - алтернативни сценарио, дијаграми секвенци	55
Слика 23 СК9 - основни сценарио, дијаграми секвенци	56
Слика 24 СК9- алтернативни сценарио, дијаграми секвенци.....	57
Слика 25 СК10 - основни сценарио, дијаграми секвенци	58
Слика 26 СК10 - алтернативни сценарио, дијаграми секвенци	59
Слика 27 СК11 - основни сценарио, дијаграми секвенци	60
Слика 28 СК11 - алтернативни сценарио, дијаграми секвенци	61
Слика 29 СК12 - основни сценарио, дијаграми секвенци	62
Слика 30 СК12 - алтернативни сценарио, дијаграми секвенци	63
Слика 31 СК13 - основни сценарио, дијаграми секвенци	64
Слика 32 СК13 - алтернативни сценарио, дијаграми секвенци.....	65
Слика 33 Табела продајно место.....	73
Слика 34 Табела купац	73
Слика 35 Табела декорација	73
Слика 36 Табела ставка	74
Слика 37 Табела производ.....	74
Слика 38 Табела особа	74
Слика 39 Табела наруџбина	75
Слика 40 Табела цветни аранжман	75
Слика 41 Табела паковање	75
Слика 42 Табела цвет.....	76
Слика 43 Табела администратор	76
Слика 44 Пројектовање екранских форми	77
Слика 45 Листа цветова	78
Слика 46 Детаљи цвета	79
Слика 47 Грешка у проналажењу цвета.....	79
Слика 48 Листа цветних аранжмана	80
Слика 49 Детаљи цветног аранжмана	81
Слика 50 Грешка проналажења цветног аранжмана	81
Слика 51 Регистровани администратор.....	82

Слика 52 Претрага наруџбина	82
Слика 53 Приказ наруџбина	83
Слика 54 Поручка о наруџбинама купца	83
Слика 55 Форма за пријаву	84
Слика 56 Попуњена форма за пријаву	85
Слика 57 Приказ профила купца	85
Слика 58 Приказ поручке да корисник не постоји.....	86
Слика 59 Листа наруџбина одабраног купца	87
Слика 60 Приказ измене наруџбине	88
Слика 61 Приказ измењене наруџбине.....	88
Слика 62 Приказ грешке при измени наруџбине	89
Слика 63 Форма за креирање цветног аранжмана	90
Слика 64 Попуњена форма за креирање цветног аранжмана	91
Слика 65 Приказ креираног цветног аранжмана.....	91
Слика 66 Испис грешке.....	92
Слика 67 Форма за регистрацију	93
Слика 68 Попуњена форма за регистрацију.....	94
Слика 69 Празна форма за пријаву	94
Слика 70 Поручка грешке приликом пријаве.....	95
Слика 71 Форма за рад са наруџбином	96
Слика 72 Поручка о успешно обрисаној наруџбини	96
Слика 73 Поручка о неуспешно обрисаној наруџбини	97
Слика 74 Форма за измену података купца	98
Слика 75 Измењени подаци о купцу.....	99
Слика 76 Профил купца	99
Слика 77 Грешка приликом измене купца	100
Слика 78 Форма за креирање цвета	101
Слика 79 Попуњена форма за креирање цвета	101
Слика 80 Профил администратора	102
Слика 81 Испис грешке приликом креирања цвета	102
Слика 82 Форма за креирање паковања	103
Слика 83 Попуњена форма з креирање паковања	103
Слика 84 Профил администратора	104
Слика 85 Поручка грешке приликом креирања паковања	104
Слика 86 Улогован купац.....	105
Слика 87 Страна за пријаву	106
Слика 88 Испис грешке.....	106
Слика 89 Форма за креирање наруџбине	107
Слика 90 Попуњена форма за креирање наруџбине	108
Слика 91 Креирана наруџбина	108
Слика 92 Грешка пиликом креирања наруџбине	108
Слика 93 Животни циклус једног захтева [10].....	109
Слика 94 УГ1 пројектовање.....	110
Слика 95 УГ2 пројектовање.....	111

Слика 96 УГ3 пројектовање.....	111
Слика 97 УГ4 пројектовање.....	112
Слика 98 УГ5 пројектовање.....	112
Слика 99 УГ6 пројектовање.....	113
Слика 100 УГ7 пројектовање.....	113
Слика 101 УГ8 пројектовање.....	114
Слика 102 УГ9 пројектовање.....	114
Слика 103 УГ10 пројектовање.....	115
Слика 104 УГ11 пројектовање.....	115
Слика 105 УГ12 пројектовање.....	116
Слика 106 УГ13 пројектовање.....	116
Слика 107 УГ14 пројектовање.....	117
Слика 108 УГ15 пројектовање.....	117
Слика 109 УГ16 пројектовање.....	118
Слика 110 УГ17 пројектовање.....	118
Слика 111 УГ18 пројектовање.....	119
Слика 112 УГ19 пројектовање.....	119
Слика 113 Пројектовање слоја приступа подацима (Први део).....	120
Слика 114 Пројектовање слоја приступа подацима (Други део).....	121
Слика 115 Архитектура софтверског система (Први део).....	122
Слика 116 Архитектура софтверског система (Други део).....	123
Слика 117 Архитектура софтверског система (Трећи део).....	124
Слика 118 Листа цветова.....	126
Слика 119 Детаљи цвета.....	127
Слика 120 Грешка у проналажењу цвета.....	127
Слика 121 Листа цветних аранжмана.....	128
Слика 122 Детаљи цветног аранжмана.....	129
Слика 123 Грешка проналажења цветног аранжмана.....	129
Слика 124 Регистровани администратор.....	130
Слика 125 Претрага наруџбина.....	130
Слика 126 Приказ наруџбина.....	131
Слика 127 Порука о наруџбинама купца.....	131
Слика 128 Форма за пријаву.....	132
Слика 129 Попуњена форма за пријаву.....	133
Слика 130 Приказ профила купца.....	133
Слика 131 Приказ поруке да корисник не постоји.....	134
Слика 132 Листа наруџбина одабраног купца.....	135
Слика 133 Приказ измене наруџбине.....	136
Слика 134 Приказ измењене наруџбине.....	136
Слика 135 Приказ грешке при измени наруџбине.....	137
Слика 136 Форма за креирање цветног аранжмана.....	138
Слика 137 Попуњена форма за креирање цветног аранжмана.....	139
Слика 138 Приказ креираног цветног аранжмана.....	139
Слика 139 Испис грешке.....	140

Слика 140 Форма за регистрацију	141
Слика 141 Попуњена форма за регистрацију	142
Слика 142 Празна форма за пријаву	142
Слика 143 Порука грешке приликом пријаве.....	143
Слика 144 Форма за рад са наруџбином	144
Слика 145 Порука о успешно обрисаној наруџбини	144
Слика 146 Порука о неуспешно обрисаној наруџбини	145
Слика 147 Форма за измену података купца	146
Слика 148 Измењени подаци о купцу	147
Слика 149 Профил купца	147
Слика 150 Грешка приликом измене купца	148
Слика 151 Форма за креирање цвета	149
Слика 152 Попуњена форма за креирање цвета	149
Слика 153 Профил администратора	150
Слика 154 Испис грешке.....	150
Слика 155 Форма за креирање паковања	151
Слика 156 Попуњена форма з креирање паковања	151
Слика 157 Профил администратора	152
Слика 158 Порука грешке приликом креирања паковања	152
Слика 159 Улогован купац.....	153
Слика 160 Страна за пријаву.....	154
Слика 161 Испис грешке.....	154
Слика 162 Форма за креирање наруџбине	155
Слика 163 Попуњена форма за креирање наруџбине	156
Слика 164 Креирана наруџбина	156
Слика 165 Грешка пиликом креирања наруџбине	157
<i>Слика 166 Имплементација класе Домен</i>	<i>158</i>
<i>Слика 167 Имплементација Слоја приступа подацима.....</i>	<i>158</i>
<i>Слика 168 Имплементација Контролера и модела</i>	<i>159</i>
<i>Слика 169 Имплементација погледа</i>	<i>159</i>
Слика 170 Табела Ставка	174
Слика 171 Табела Производ	174
Слика 172 Табела ПродајноМесто	174
Слика 173 Табела Паковање	175
Слика 174 Табела Наружбина	175
Слика 175 Табела Купац.....	175
Слика 176 Табела Декорација	175
Слика 177 Табела ЦветниАранжман	176
Слика 178 Табела Цвет	176
Слика 179 Табела Администратор.....	176
Слика 180 Табела Особа.....	177

1. УВОД

Microsoft ASP.NET Core је део историје Microsoft технологија чији је циљ изградња веб сајтова и веб сервиса. Све је почело још 1996. године са Active Server Pages (ASP) који представља први покушај стварања платформе за динамичко извршење кода веб сајта на страни сервера.

ASP.NET Core је нова верзија радног оквира ASP.NET, настала због акумулације великог броја функција у програмском склопу на коме је грађен ASP.NET и његове строге везаности са Microsoft-овим сервером Internet Information Services (IIS). Иако ASP.NET Core и даље подржава коришћење IIS сервера као веб сервера постоје и боље опције. Данас се ASP.NET Core апликација може развити и покренути међуплатформски на Windows, macOS и Linux систему. Такође, Microsoft је креирао и међуплатформски веб сервер сјајних перформанси под називом Kestrel [2].

ASP.NET Core подразумева интегрисање имплементација .NET Framework технологија, попут Web API, MVC и SignalR, са новијим технологијама, као што су Razor Pages, gRPC и Blazor, што пружа могућност међуплатформског извршавања [2].

Софтверски систем представљен у овом раду је веб апликација развијена за део пословања цвећаре, која својим регистрованим и пријављеним корисницима нуди могућност једноставног креирања цветних аранжама, такође и наручивање тих цветних аранжмана, али и засебног цвећа, на жељену адресу. Исто тако, апликација ће помоћи и самим компанијама да једноставније представе своју понуду потрошачима, пруже им могућност персонализованих производа, као и лакшу манипулацију самим процесом наручивања и доставе производа. Приликом реализације имплементирани су Repository и Unit of Work патерни.

Наредно поглавље нас детаљније упознаје са коришћеним технологијама за развој веб апликације, као што су .NET технологије, JavaScript-ом као програмским језиком који је коришћен за програмирање функционалности изгледа сајта, Bootstrap-ом, радним оквиром који је коришћен за стилизовање компоненти сајта. Од трећег поглавља почиње описивање фаза Ларманове методе, а то су: фаза прикупљања захтева, фаза анализе, фаза пројектовања, фаза имплементације и фаза тестирања. На крају су представљени закључак и коришћена литература.

2. ПРЕГЛЕД КОРИШЋЕНИХ ТЕХНОЛОГИЈА

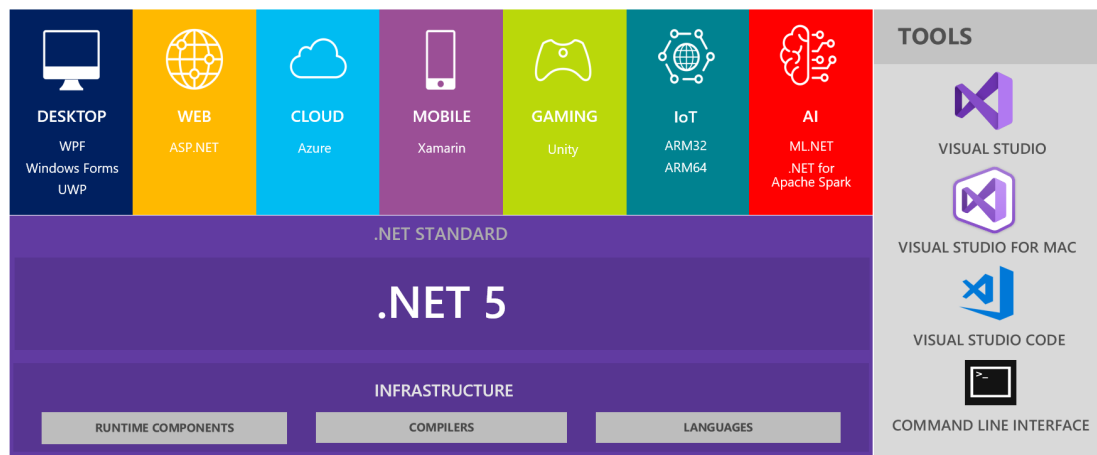
2.1 .NET платформа

.NET је платформа отвореног кода намењена програмерима за креирање великог броја различитих апликација употребом различитих језика и библиотека [2].

Компоненте .NET-а су:

- *Компајлери језика* - Претварају изворни код написан језиком C#, F# и Visual Basic у код посредничког језика. Од верзије C# 6 Microsoft користи преписани компајлер отвореног кода, Roslyn.
- *Common Language Runtime (CoreCLR)* - компјлира ИЛ код сачуван у програмским склоповима у инструкције процесора рачунара и извршава код унутар окружења који управља ресурсима, као што су програмске нити и меморија
- *Base Class Libraries (BCL CoreFX)* - Унапред грађени програмски склопови типова упакованих и дистрибуираних помоћу NuGet пакета за извршавање уобичајених задатака током изградње апликација.

.NET – A unified platform



Слика 1 Приказ NET. Платформе [6]

2.2 .NET Core

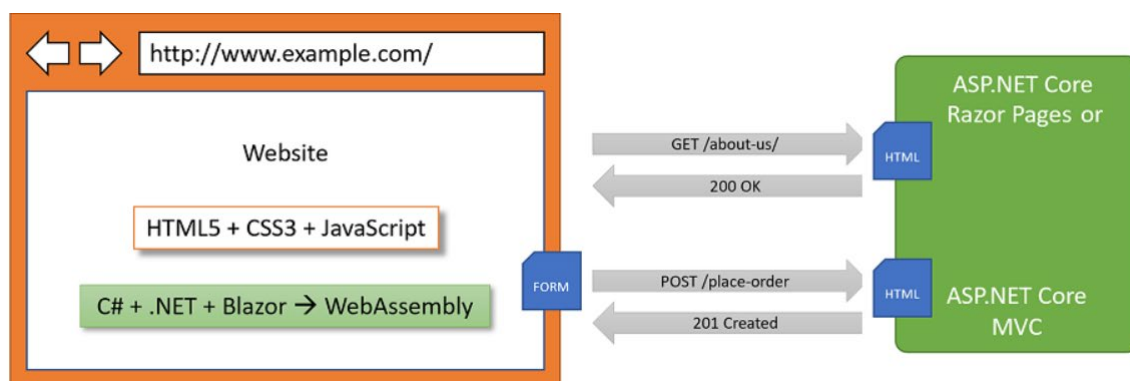
Тежња Microsoft-а да уклони зависност .NET платформе од Windows оперативног система проистекла је из све више присутних међуплатформских захтева и развоја модерних мобилних и cloud платформи. Током процеса тог прилагођавања из .NET Framework-а уклоњени су главни делови који више нису сматрани основним. Тако добијен, сасвим нови производ, назван .NET Core, омогућио је међуплатформску имплементацију компоненте CLR, познатију као CoreCLR, као и модернију библиотеку класа названу CoreFX.

2.3 ASP.NET Core

Као што је већ речено, ASP.NET Core радни оквир користи се за изградњу веб сајтова. Веб сајт чини више веб страница које могу бити статички учитане из фајл система или их динамички генерише технологија на страни сервера, као што је ASP.NET Core [2].

ASP.NET Core пружа могућност различитих технологија за израду веб сајтова:

- ASP.NET Core Razor Pages и Razor библиотека класа нам пружа динамичко генерисање HTML-а за једноставне веб сајтове.
- ASP.NET Core MVC представља имплементацију патерна Model-View-Controller (MVC).
- Blazor омогућава изградњу компоненте корисничког интерфејса коришћењем језика C# и .NET платформе, уместо коришћења UI радног оквира заснованог на JavaScript-у.

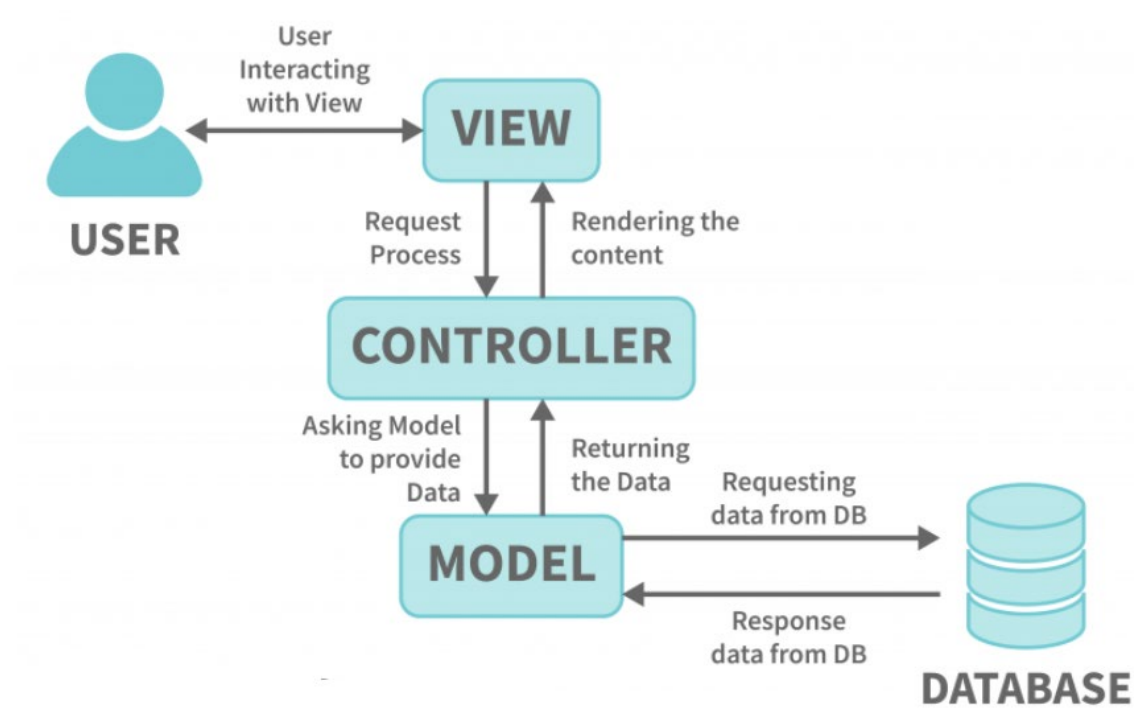


Слика 2 Употреба језика C# и .NET платформе за изградњу веб сајтова [7]

2.3.1 ASP.NET Core MVC архитектура

ASP.NET Core Razor Pages је погодан за једноставне сајтове, али у случају сложенијих веб сајтова препоручује се формалнија структура како би управљање том сложености било могуће. За такве потребне користан је патерн Model-View-Controller (MVC) који апликацију раздваја у три главне групе компоненти [2]:

- **Модели** (енг. *Models*) - класе које представљају ентитете података и моделе приказа употребљене у веб сајту.
- **Прикази** (енг. *Views*) - Разор фајлови, односно .cshtml фајлови који рендерују податке у моделе приказа у HTML веб страницама.
- **Контролери** (енг. *Controllers*) - класе које извршавају код када стигне HTTP захтев на веб сервер. Методе контролера обично креирају модел приказа који може да садржи моделе ентитета и прослеђује га у приказ за генерисање HTTP одговора за слање назад у веб претраживач или други клијент.



Слика 3 Model-view-controller архитектура [8]

Подразумевана структура директоријума AS.NET Core MVC пројекта садржи [2]:

- *Controllers* - Овај директоријум садржи C# класе које имају методе које реузимају модел и прослеђују га у приказ.

- *Models* - Овај директоријум садржи C# класе које представљају све податке које је контролер сакупио и прослодеио у приказ.
- *Properties* - Овај директоријум садржи конфигурациони фајл за IIS или IIS Express на Windows систему за покретање веб сајта у току развоја, чији је назив launchSettings.json. Овај фајл се користи само на локалној развојној машини и не распоређује се у веб сајт у продукцији.
- *Views* - Овај директоријум садржи .cshtml Razor фајлове који комбинују HTML и C# код за динамичко генерисање HTML одговора.
- *wwwroot* - Овај директоријум садржи статички садржај употребљен у веб сајту, као што су CSS за стилове, JavaScript библиотеке, слике, PDF документи,...
- *appsettings.json* - овај фајл садржи подешавања која веб сајт може да учита у време извршавања, као што је знаковни низ конекције базе података за ASP.NET Identity систем и нивои евидентирања.
- *Program.cs* - овај фајл дефинише скривену класу Program која садржи тачку уноса Main.
- *Startup.cs* - овај фајл помоћу свог метода ConfigureServices(IserviceCollection services) пружа могућност дефинисања сервиса који се могу инјектовати у конструкторе објеката помоћу Dependency Injection (DI) патерна, један од таквих сервиса јесте, на пример, и ASP.NET Core Identity за аутентикацију. Поред овог метода класа Startup има и метод Configure(IApplicationBuilder app, IWebHostEnvironment env) којим дефинише ASP.NET Core pipeline задужен за обраду улазних HTTP захтева.

ASP.NET Core апликације су апликације засноване на концепту Generic Host-а, зато можемо приметити да се у класи Program фајла Program.cs поред Main методе, налази и метод CreateHostBuilder који позивањем методе UseStartup<Startup>() омогућава да Host сервиси и дефиниција ASP.NET Core pipeline-а буду преузети из метода класе Startup.cs пројекта.

2.4 ENTITY FRAMEWORK CORE

Object-Relation Mapping (ORM) алати преводе (мапирају) релационе табеле базе података у колекцију објеката у меморији, док поља у табели представљају одговарајућа својства објекта. У конфигурацији мапирања дефинисане су опције за мапирање између типова базе података и типова објектно-оријентисаног језика. Entity Framework Core нуди три начина на које се може дефинисати конфигурација:

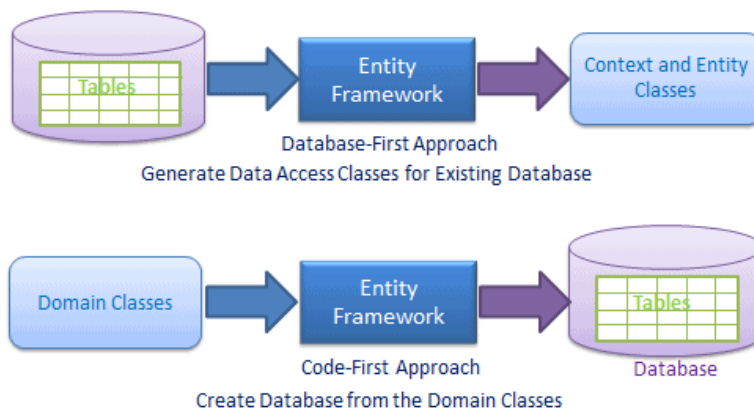
- Анотације података (атрибути својства)
- Конвенције именовања
- Fluent API - може се употребити за спецификовање било које конфигурационе опције, док се претходна два могу употребити за мање подскупове конфигурационих опција

Због разлика у управљању трансакцијама, типовима базе података и другим нестандардизованим функцијама, сваки ORM прилагођава се специфичном типу базе података помоћу адаптера одређеног за дату базу података, под називом провајдери или конектори. Entity Framework Core има провајдере за већину доступних база података.

Цела база података (или само њен део) представљена је класом у кеш меморији која садржи својство за сваку колекцију мапирану у табели базе података. Операције се најпре извршавају у инстанци класе у кеш меморији, а тек након тога се та инстанца синхронизује са базом податка. Такву класу кеша у меморији код Entity Framework Core-а називамо DbContext, а она садржи конфигурацију мапирања. Тачније, класа кеша у меморији специфична за апликацију добија се наслеђивњем класе DbContext и додавањем инстанце у све мапиране колекције и потребне конфигурационе информације. Entity Framework Core прослеђује све промене извршене на инстанци DbContext у базу података позивом методе SaveChanges(). Промене прослеђене у базу података током операције синхронизације се извршавају у једној трансакцији.

Након што су дефинисани сви ентитети потребно је дефинисати колекције у меморији, које ће заправо представљати саме табеле у бази података. То се може урадити веома лако додавањем својства колекције DbSet<T> у класу DbContext за сваки ентитет T.

Информације конфигурације мапирања које се не могу спецификовати у дефиницијама самих ентитета морају бити додате у OnModelCreating() метод DbContext-а. Свака информација везана за ентитет T почиње са builder.Entity<T>() и наставља се позивом методе која спецификује одређено ограничење. Додатни позиви спецификују додатна својства ограничења [3].



Слика 4 Различити приступи употребе Entity Framework Core-а [9]

2.4.1 Entity Framework Core u LINQ ynumu

LINQ упити креирају IEnumerable инстанце, колекције чији елементи нису израчунати када је инстанца IEnumerable креирана на крају упита, већ покушајем да се преузму елементи колекције из инстанце. Таква појава назива се одложено извршење и функционише тако што LINQ упити који се покрећу из мапиране колекције класе DbContext креирају специфичан подтип инстанце IEnumerable под називом IQueryable, који садржи све потребне информације за слање упита у базу података, али се стварни SQL креира и извршава тек приликом преузимања првог елемента подтипа IQueryable. Обично се Entity Framework Core упит завршава операцијом ToList() или ToArray() која трансформише IQueryable у листу низова, што доводи до извршавања упита у бази података. Операцију SingleOrDefault() извршавамо уколико очекујемо један или ниједан елемент [3].

2.4.2 Entity Framework Core миграције

Ради генерисања физичке базе података и креирања снимака структуре базе података неопходних за комуникацију Entity Framework Core-а са базом података, потребно је инсталирати дизајнерске алатке Entity Framework Core-а као NuGet пакете. Постоје две опције:

- Алатке које функционишу у било којој Windows конзоли
- Алатке које су специфичне за Visual Studio Package Manager Console-у

Дизајнерске алатке се користе у процедури ажурирања, када се модификује DbContext класа или дефиниција неког ентитета потребно је покренути дизајнерске алатке да би Entity Framework Core детектовао и обрадио све настале промене. Дизајнерске алатке ажурирају снимак структуре базе података и генеришу нову миграцију, која представља фајл који садржи све инструкције потребне за модификовање физичке базе података да би се рефлектовале све извршене промене [3].

Командом Add-Migration *nazivMigracije* у Package Manager Console-и креирамо нову миграцију. У случају да желимо да поништимо миграцију унећемо команду Remove-Migration. Након извршене прве миграције у пројекту се појављује нови директоријум под називом Migrations. Свака миграција има метод Up I Down, где Up подразумева миграцију, а Down њено поништавање. Миграције се примењују у бази тек када се изврши команда Update-Database [3].

2.5 Софтверски патерни

2.5.1 DEPENDENCY INJECTION PATTERN

Употреба Dependency Injection патерна сматра се добрим начином имплементирања Dependency Inversion принципа, такође, још једна добра страна примене овог патерна јесте што намеће свакој имплементацији да прати све остале SOLID принципе [3].

Уместо креирања интерфејса објеката од којих је компонента зависна, потребно је само дефинисање њихове зависности, декларисање њихових интерфејса и омогућавање пријема објеката инјектирањем.

.NET Core садржи скуп библиотека за решавање овог проблема у именском простору Microsoft.Extensions.DependencyInjection. Овај Inversion of Control (IoC) контејнер је одговоран за креирање и излагање објеката који су затражени.

Постоје три опције за животни век инјектиране компоненте:

- Transient - Објекти су креирани сваки пут када су затражени
- Scoped - Креира се објекат за сваки опсег дефинисан у апликацији
- Singleton - Сваки објекат има исти животни век примене, па је један објекат употребљен више пута да би служио свим захтевима за дати тип

2.5.3 SINGLETON PATTERN

Један од најчешће коришћених патерна у свакој апликацији. Овај патерн користимо када желимо да у целој апликацији имамо имплементирану само једну инстанцу неког објекта. Singleton патерн пружа боље решење од глобалних променљивих.

Класа је одговорна за креирање и испоруку једног објекта који ће апликација користити. Singleton класа креира једну инстанцу, а да би то било могуће потребно је да се објекат који је креиран статички испоручује у статичком својству или методу [3].

Важно је напоменути да је због имплементације инјектирања зависности у .NET-у 5 употреба Singleton патерна постала мање уобичајена јер се може подесити инјектирање зависности за руковање Singleton објектима.

2.6 JavaScript

JavaScript је скрипти, платформски неутралан језик, који омогућава динамичко ажурирање садржаја. Углавном се извршава на клијентској страни и служи за креирање интерактивних страница. Једна од његових најзначајнијих предности је могућност функционисања на свим познатијим веб претраживачима, као што су Mozilla, Chrome, Firefox и остали.

Када је JavaScript креиран, у почетку је имао друго име, LiveScript, међутим Java је у то време била веома популарна, па је одлучено да би позиционирање новог језика као „млађег брата“ Jave помогло. Али како је еволуирао, JavaScript је постао потпуно независан језик са сопственом спецификацијом званом ECMAScript, и сада уопште нема везе са Javom [11].

JavaScript у претраживачу може да уради све што се односи на манипулацију веб страницом, интеракцију са корисником и веб сервером. На пример, JavaScript у прегледачу може [11]:

- Додати нови HTML на страницу, променити постојећи садржај, променити стилове.
- Реаговати на радње корисника, покренути акцију кликом миша, померањем показивача, притиском тастера.
- Послати захтеве преко мреже на удаљене сервере, преузимати и постављати датотеке (тзв. AJAX и COMET технологије).
- Набавити и поставити колачиће, поставити питања посетиоцу, приказати поруке.
- Запамтити податке на страни клијента (“local storage”).

2.6.1 JQuery

JQuery је једна од најпопуларнијих JavaScript библиотека отвореног кода (енг. open source). JQuery се користи за преузимање уобичајених задатака који захтевају много редова JavaScript кода и уједини их у различите методе, које се затим могу позвати једном линијом кода. JQuery је брза, сажета, мала и богата функцијама библиотека која је подржана је од стране свих претраживача. Најчешће се користи за манипулацију HTML DOM-а, поједностављује руковање догађајима, CSS анимацијама и слањем Ajax позива. Символ \$ се користи за приступ JQuery библиотеци, проналажење елемената и предузимање акција. [12]

2.6.2 AJAX

AJAX је скраћеница за асинхрони JavaScript и XML. Представља употребу XMLHttpRequest за комуникацију са серверима. Може да шаље и прима информације у различитим форматима, укључујући JSON, XML, HTML и текстуалне датотеке. Најпривлачнија карактеристика AJAX-а је његова асинхрона природа, што значи да може да комуницира са сервером, размењује податке и ажурира страницу без потребе за освежавањем странице [13].

Две главне карактеристике AJAX-а омогућавају вам да урадите следеће [13]:

- Направите захтеве серверу без поновног учитавања странице
- Примате и радите са подацима са сервера

2.7 Bootstrap

Bootstrap је бесплатни front-end радни оквир (енг. framework) отвореног кода (енг. open source) за креирање веб страница и веб апликација. Првобитно је дизајниран да омогући респонзивни развој веб сајтова намењених мобилним уређајима [14].

Bootstrap оквир је изграђен на Hypertext Markup Language (HTML), cascading style sheets (CSS) и JavaScript-у. Веб програмери који користе Bootstrap могу да праве веб странице много брже без трошења времена на бриге о основним командама и функцијама [14].

3 СТУДИЈСКИ ПРИМЕР

Ово поглавље биће посвећено детаљном објашњењу израде студијског примера, чији је развој заснован на Лармановој методи развоја софтверског система.

Ларманова метода се састоји из следећих фаза:

1. Прикупљање захтева
2. Фаза анализе
3. Фаза пројектовања
4. Фаза имплементације
5. Фаза тестирања

Фаза прикупљања захтева – У овој фази неопходно је прикупити што више информација везаних за пројекат, ради бољег разумевања потреба које софтверски систем треба да задовољи. Неидентификација или лоше дефинисање корисничких потреба може довести до великих проблема у даљем развоју софтверског система. На основу уочених функционалности које захтева наш систем дефинишемо UML модел случаја коришћења. UML модел случајева коришћења састоји се од скупа случаја коришћења, актора и веза између актора и случаја коришћења. Случај коришћења је садржан од скупа сценарија, међу којима је један основни, а остали алтернативни. Сценарио се састји од више акција које представљају комуникацију актера и система [1].

Акције које систем може извршити су:

- СО – Систем извршава системску операцију
- ИА – Излазни аргументи који представљају резултат системске операције

Акције које актор може извршити су:

- АПУСО – Актор припрема улаз за системску операцију
- АПСО – Актор позива системску операцију
- АНСО – Актро извршава несистемску операцију

Фаза анализе - У фази анализе се, из претходно добијеног UML модела случајева коришћења, одређује структура и понашање софтверског система. Најкоришћенији модели помоћу којих се описује структура система су проширени модел објекти везе, релациони модел и концептуални модел. Понашање софтверског система описује се помоћу системских операција и секвенцих дијаграма, односно дијаграма сарадње. Овако добијени модели података и понашања система представљају пословну логику софтверског система. Дијаграм секвенци за сваки

сценарио случаја коришћења приказује интеракцију актора и система у одређеном редоследу. Резултат анализе секвенцих дијаграма су системске операције које треба пројектовати у коду. Уговор се креира за сваку системску операцију, а он се састоји из секција [1]:

- Операција – назив операције и њени улазни и излазни аргументи
- Веза са СК - имена случајева коришћења у којима се позива ова системска операција
- Предуслов – пре извршења системске операције морају бити испињени одређени предуслови
- Постуслови – после извршења системске операције у систему морају бити задовољени одређени постуслови

Фаза пројектовања – У фази пројектовања се описује понашање и физичка структура софтверског система. Пројектовање архитектуре представља трослојну архитектуру која обухвата: пројектовање корисничког интерфејса, апликационе логике и складишта података [1].

Фаза имплементације – У фази имплементације се креирају имплементационе компоненте које имплементирају у некој технологији [1].

Фаза тестирања – У фази тестирања се тестирају све компоненте које су креиране у фази имплементације. За сваку имплементациону компоненту се прави тест случајева коришћења, тест процедуре и тест компоненте [1].

4. КОРИСНИЧКИ ЗАХТЕВИ

Прва фаза у развоју софтверског система Лармановом методом је прикупљање захтева. Захтеви представљају својства и услове које софтверски систем мора да задовољи. Постоје различити типови захтева које систем мора да задовољи и они се категоризују као функционални и нефункционални захтеви. Функционални захтеви дефинишу захтеване функције система, док нефункционални захтеви дефинишу остале захтеве [1].

3.1 Вербални Опис

Софтверски систем за део пословања цвећаре представља систем преко којег администратори могу представити своју понуду, водити евиденцију о потрошачима и њиховим наруџбинама и додати своје нове понуде. У систему постоје две улоге: купац и администратор.

На почетној страници администратори могу видети своје информације, као и дугме за додавање нових цветова и нових паковања, такође и дугме помоћу кога могу да претраже своје купце по корисничком имену, а затим и управљати њиховим наруџбинама. Они могу мењати статус наруџбине или је обрисати кад та наруџбина постане застарела и више не буде потребе да се информације о њој памте.

На почетној страници купац такође има увид у тренутне информације које се о њему складиште и по потреби их може изменити. Купцу се пружа могућност да сам креира цветни аранжман, изабере које ће све цвеће бити у њему, у којој количини и у каквом паковању. Купац има детаљан приказ свих цветова и паковања, као и њихових цена. Купац може креирати и наруџбину са избором да ли она треба бити достављена на жељену адресу или лично преузета у неком продајном објекту, такође купац поред лично креираних цветних аранжмана може поручити и засебно само цветове. Такође, он има и детаљан преглед свих својих цветних аранжмана, наруџбина и продајних места.

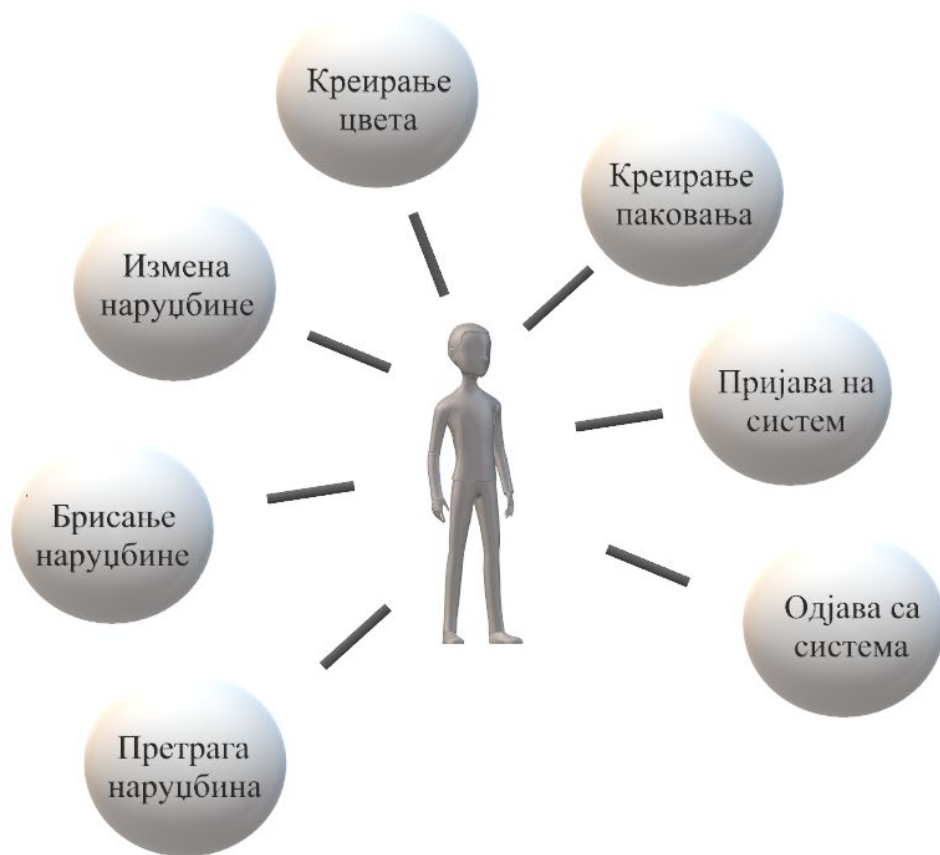
3.2 Опис захтева помоћу модела случаја коришћења

Издавају се следећи случајеви коришћења:

1. Приказивање детаља цвета
2. Приказивање детаља цветног аранжмана
3. Претраживање наруџбина
4. Пријављивање на систем
5. Измена наруџбине
6. Креирање цветног аранжмана
7. Креирање корисничког налога
8. Брисање наруџбне
9. Измена корисничког налога
10. Креирање цвета
11. Креирање паковања
12. Одјава са система
13. Креирање наруџбине



Слика 5 Случајеви коришћења на страни купца



Слика 6 Случајеви коришћења на страни Администратора

СК1: Случај коришћења – Приказивање детаља цвета

Назив СК

Приказивање детаља цвета

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Купцу је приказана листа цвећа.

Основни сценарио СК

1. Купац **бира** цвет коју жели да му се прикаже. (АПУСО)
2. Купац **позива** систем да му прикаже изабрани цвет. (АПСО)
3. Систем **тражи** цвет по задатој вредности. (СО)
4. Систем **приказује** кориснику цвет. (ИА)

Алтернативна сценарија

4.1. Уколико систем не може да пронађе цвет, купцу се приказује порука : „Десила се грешка“ (ИА)

СК2: Случај коришћења – Приказивање детаља цветног аранжмана

Назив СК

Приказивање детаља цветног аранжмана

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Купцу је приказана листа цветних аранжмана.

Основни сценарио СК

1. Купац **бира** цветни аранжман коју жели да му се прикаже. (АПУСО)
2. Купац **позива** систем да му прикаже изабрани цветни аранжман. (АПСО)
3. Систем **тражи** цветни аранжман по задатој вредности. (СО)
4. Систем **приказује** кориснику цветни аранжман. (ИА)

Алтернативна сценарија

- 4.1. Уколико систем не може да пронађе цветни аранжман, кориснику се приказује порука : „ Десила се грешка“ (ИА)

СКЗ : Случај коришћења – Претраживање наруџбина

Назив СК

Претраживање наруџбина

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Администратор је регистрован на свој налог.

Основни сценарио СК

1. Администратор **уноси** критеријум претраживања наруџбина. (АПУСО)
2. Администратор **контролише** да ли је коректно унео корисничко име за претрагу. (АНСО)
3. Администратор **позива** систем да претражи наруџбине по корисничком имену. (АПСО)
4. Систем **претражује** наруџбине. (СО)
5. Систем **приказује** администратору наруџбине. (ИА)

Алтернативна сценарија

4.1. Уколико систем не може да претражи наруџбине, администратору се приказује порука: „Тражени купац нема наруџбине“ (ИА)

СК4 : Случај коришћења – Пријављивање на систем

Назив СК

Пријављивање на систем

Актери СК

Администратор/купац

Учесници СК

Администратор/купац и систем

Предуслов: Систем је укључен и приказује форму за пријављивање

Основни сценарио СК

1. Администратор/купац **уноси** податке за пријављивање. (АПУСО)
2. Администратор/купац **контролише** да ли је коректно унео корисничко име и лозинку (АНСО)
3. Администратор/купац **позива** систем да га пријави. (АПСО)
4. Систем **проверава** да ли су унети подаци у реду. (СО)
5. Систем **приказује** администратору/купцу доступне опције. (ИА)

Алтернативна сценарија

5.1 Уколико систем не може да верификује администратора/купца он приказује поруку: „Корисник не постоји” (ИА)

СК5 : Случај коришћења – Измена наруџбине

Назив СК

Измена наруџбине

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Учитане су наруџбине и листа опција за измену статуса наруџбине. Систем приказује форму за рад са наруџбином.

Основни сценарио СК

1. Администратор **уноси** критеријум по ком претражује наруџбине. (АПУСО)
2. Администратор **позива** систем да пронађе наруџбине на основу задате вредности. (АПСО)
3. Систем **тражи** наруџбине по задатом критеријуму и учитава податке о њима. (СО)
4. Систем **приказује** администратору пронађене наруџбине. (ИА)
5. Администратор **бира** наруџбину коју жели да измени. (АПУСО)
6. Администратор **мења** податке о наруџбини (АПУСО)
7. Администратор **контролише** да ли је коректно унео податке о наруџбини. (АНСО)
8. Администратор **позива** систем да запамти податке о наруџбини. (АПСО)
9. Систем **памти** податке о наруџбини. (СО)
10. Систем **приказује** администратору забележену измену. (ИА)

Алтернативна сценарија

4.1 Уколико систем не може да пронађе наруџбине он приказује администратору поруку „Десила се грешка” (ИА)

10.1 Уколико систем не може да запамти податке о наруџбини он приказује администратору поруку „Десила се грешка” (ИА)

СК6 : Случај коришћења – Креирање цветног аранжмана

Назив СК

Креирање цветног аранжмана

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитна је листа цветова и паковања. Систем приказује форму за рад са цветним аранжманом.

Основни сценарио СК

1. Купац **уноси** податке о цветном аранжману. (АПУСО)
2. Купац **контролише** да ли је коректно унео податке о цветном аранжману. (АНСО)
3. Купац **позива** систем да креира нови цветни аранжман са задатим подацима. (АПСО)
4. Систем **креира** цветни аранжман са задатим подацима. (СО)
5. Систем **приказује** купцу забележени цветни аранжман. (ИА)

Алтернативна сценарија

- 6.1 Уколико систем не може да креира цветни аранжман он приказује купцу поруку „Десила се грешка” (ИА)

СК7 : Случај коришћења – Креирање корисничког налога

Назив СК

Креирање корисничког налога

Актери СК

Посетилац

Учесници СК

Посетилац и систем

Предуслов: Систем је укључен. Систем приказује форму за креирање корисничког налога

Основни сценарио СК

1. Посетилац **уноси** податке о новом купцу. (АПУСО)
2. Посетилац **контролише** да ли је коректно унео податке о купцу. (АНСО)
3. Посетилац **позива** систем да креира новог купца са задатим подацима. (АПСО)
4. Систем **креира** новог купца. (СО)
5. Систем **приказује** посетиоцу поруку: „Успешно креиран налог“. (ИА)

Алтернативна сценарија

5.1 Уколико систем не може да креира купца он приказује посетиоцу поруку „Десила се грешка“ (ИА)

СК8 : Случај коришћења – Брисање наруџбине

Назив СК

Брисање наруџбине

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са наруџбином

Основни сценарио СК

1. Администратор **уноси** критеријум по ком претражује наруџбине. (АПУСО)
2. Администратор **позива** систем да пронађе наруџбине на основу задате вредности. (АПСО)
3. Систем **тражи** наруџбине по задатом критеријуму и учитава податке о њима. (СО)
4. Систем **приказује** администратору пронађене наруџбине. (ИА)
5. Администратор **бира** наруџбину коју жели да обрише. (АПУСО)
6. Администратор **позива** систем да обрише податке о наруџбини. (АПСО)
7. Систем **брише** податке о наруџбини. (СО)
8. Систем **приказује** администратору поруку: „Успешно је обрисана наруџбина.” (ИА)

Алтернативна сценарија

4.1 Уколико систем не може да пронађе наруџбине он приказује администратору поруку „Десила се грешка” (ИА)

8.1 Уколико систем не може да обрише податке о наруџбини он приказује администратору поруку „Десила се грешка” (ИА)

СК9 : Случај коришћења – Измена корисничког налога

Назив СК

Измена корисничког налога

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитани су подаци о купцу. Систем приказује форму за рад са профилем купца.

Основни сценарио СК

1. Систем **тражи** податке о купцу. (СО)
2. Систем **приказује** купцу пронађене податке. (ИА)
3. Купац **мења** податке. (АПУСО)
4. Купац **контролише** да ли је коректно унео податке о купцу. (АНСО)
5. Купац **позива** систем да запамти податке. (АПСО)
6. Систем **памти** податке о купцу. (СО)
7. Систем **приказује** посетиоцу поруку: „ Успешно сте изменили податке “. (ИА)

Алтернативна сценарија

2.1 Уколико систем не може да нађе податке о купцу он приказује купцу поруку: „ Десила се грешка”.

7.1 Уколико систем не може да запамти податке о купцу он приказује купцу поруку „ Десила се грешка” (ИА)

СК10 : Случај коришћења – Креирање цвета

Назив СК

Креирање цвета

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са цветом.

Основни сценарио СК

1. Администратор **уноси** податке о цвету. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке о цвету. (АНСО)
3. Администратор **позива** систем да креира нови цвет са задатим подацима. (АПСО)
4. Систем **креира** цвет са задатим подацима. (СО)
5. Систем **приказује** администратору поруку: „Успешно сте креирали цвет“. (ИА)

Алтернативна сценарија

- 5.1 Уколико систем не може да креира цвет он приказује администратору поруку „Десила се грешка” (ИА)

СК11 : Случај коришћења – Креирање паковања

Назив СК

Креирање паковања

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улоган под својом шифром. Систем приказује форму за рад са паковањем.

Основни сценарио СК

1. Администратор **уноси** податке о паковању. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке о паковању. (АНСО)
3. Администратор **позива** систем да креира ново паковње са задатим подацима. (АПСО)
4. Систем **креира** паковање са задатим подацима. (СО)
5. Систем **приказује** администратору поруку: „Успешно сте креирали паковање“. (ИА)

Алтернативна сценарија

5.1 Уколико систем не може да креира паковање он приказује администратору поруку „Десила се грешка“ (ИА)

СК12 : Случај коришћења – Одјава са система

Назив СК

Одјава са система

Актери СК

Администратор/купац

Учесници СК

Администратор/купац и систем

Предуслов: Систем је укључен и администратор/купац је улогован под својом шифром.

Основни сценарио СК

1. Администратор/купац позива систем да га одјави. (АПСО)
2. Систем одјављује администратора/купца са система и приказује почетну страну. (ИА)

Алтернативна сценарија

- 2.1 Уколико систем не може да одјави администратора/купца са система: „Десила се грешка”. (ИА)

СК13 : Случај коришћења – Креирање наруџбине

Назив СК

Креирање наруџбине

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитана је листа производа и продајних места. Систем приказује форму за рад са наруџбином.

Основни сценарио СК

1. Купац **уноси** податке за нову наруџбину. (АПУСО)
2. Купац **контролише** да ли је коректно унео податке о наруџбини. (АНСО)
3. Купац **позива** систем да креира нову наруџбину са задатим подацима. (АПСО)
4. Систем **креира** наруџбину са задатим подацима. (СО)
5. Систем **приказује** купцу забележену наруџбину. (ИА)

Алтернативна сценарија

5.1 Уколико систем не може да креира наруџбину он приказује купцу поруку „Десила се грешка” (ИА)

5. АНАЛИЗА

Фаза анализе описује логичку структуру и понашање софтверског система, тј. пословну логику софтверског система. Понашање се дефинише кроз дијаграме секвенци и уговоре о системским операцијама, а структуру дефинишемо кроз концептуални и релациони модел [1].

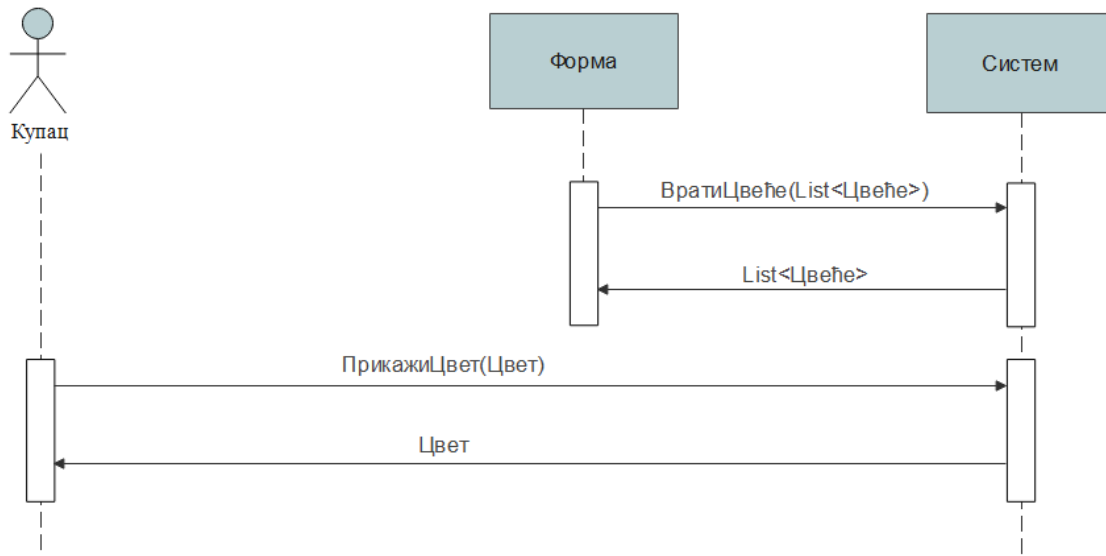
5.1 Понашање софтверског система – Системски дијаграми секвенци

UML дијаграмом секвенци се описује понашање система. За сваки случај коришћења, настао у фази прикупљања захтева, се приказује по један дијаграм секвенци. Он приказује догађаје у одређеном редоследу, који успостављају интеракцију између актора и софтверског система [1].

СК1: Случај коришћења – Приказивање детаља цвета

Основни сценарио СК

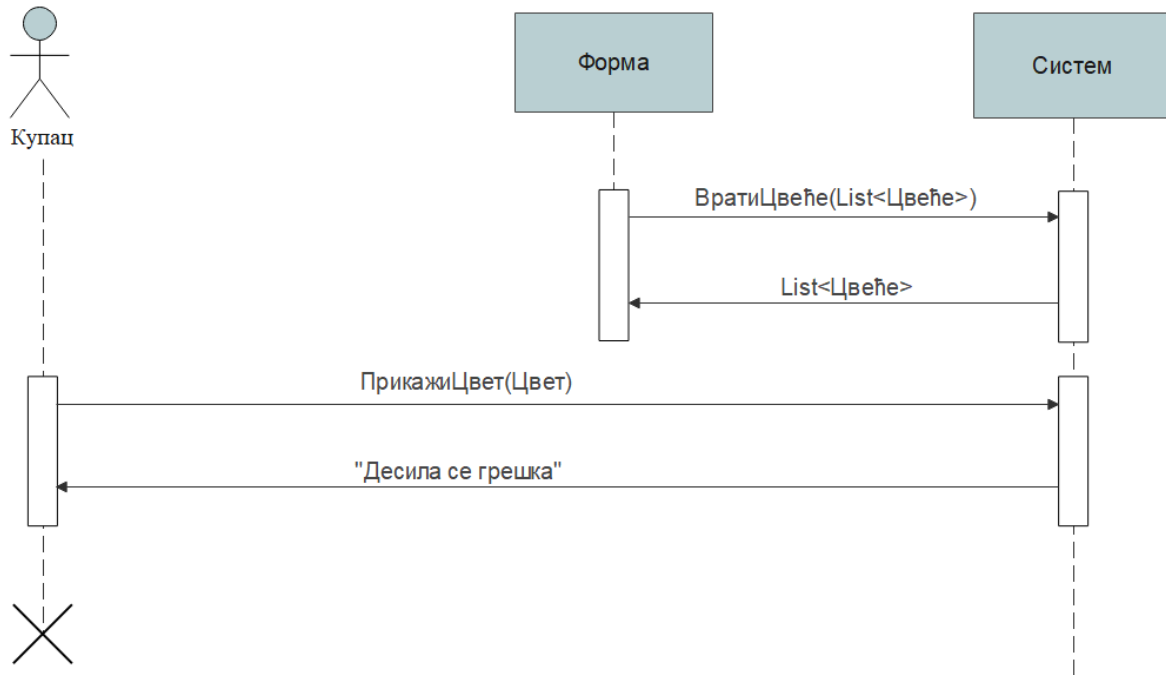
1. Купац **бира** цвет коју жели да му се прикаже. (АПУСО)
2. Купац **позива** систем да му прикаже изабрани цвет. (АПСО)
3. Систем **тражи** цвет по задатој вредности. (СО)
4. Систем **приказује** кориснику цвет. (ИА)



Слика 7 СК1 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

4.1. Уколико систем не може да пронађе цвет, купцу се приказује порука : „Десила се грешка“ (ИА)



Слика 8 СК1 - алтернативни сценарио, дијаграми секвенци

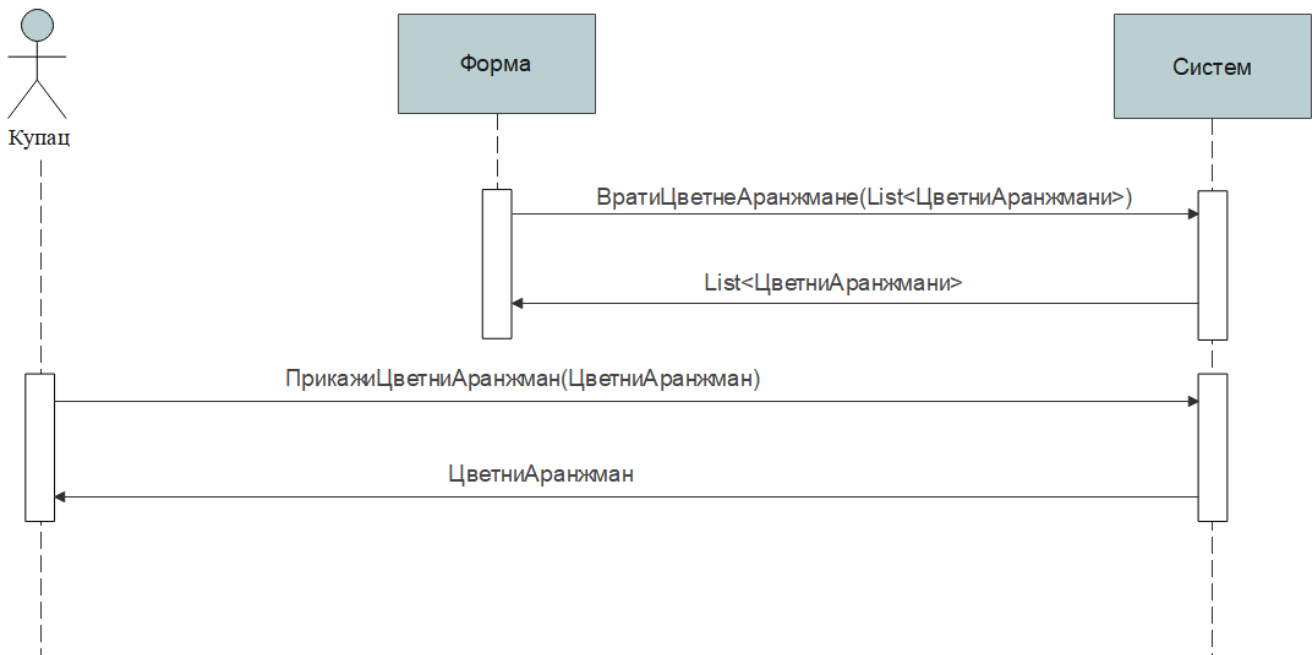
Са наведених дијаграма секвенци уочавају се две системске операције:

1. Сигнал ПрикажиЦвет(Цвет)
2. Сигнал ВратиЦвеће(List)

СК2: Случај коришћења – Приказивање детаља цветног аранжмана

Основни сценарио СК

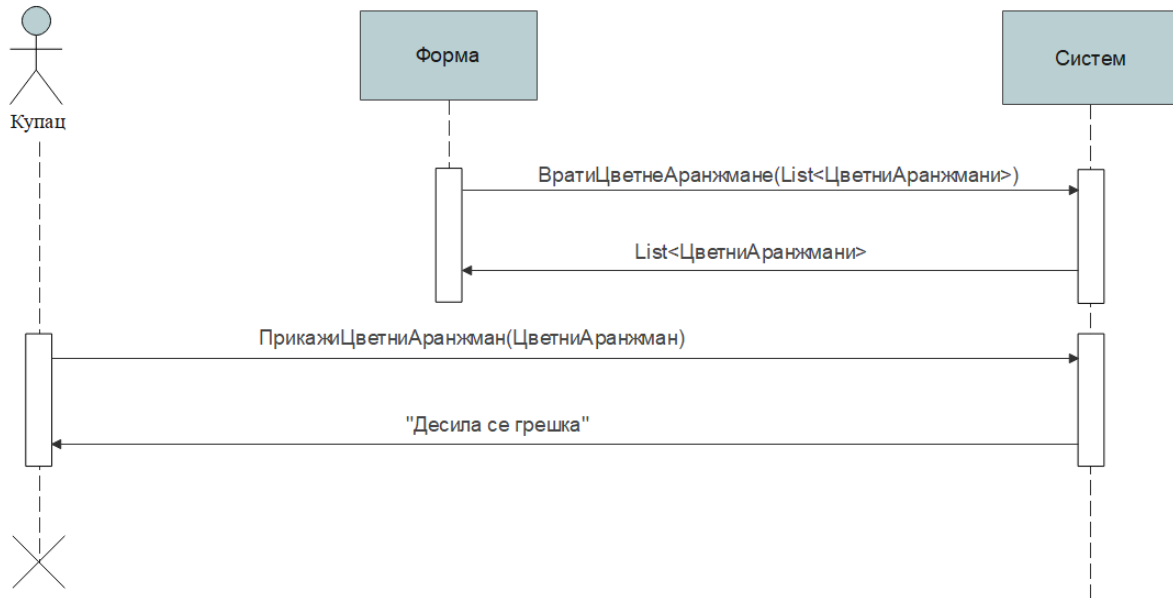
1. Купац **бира** цветни аранжман коју жели да му се прикаже. (АПУСО)
2. Купац **позива** систем да му прикаже изабрани цветни аранжман. (АПСО)
3. Систем **тражи** цветни аранжман по задатој вредности. (СО)
4. Систем **приказује** кориснику цветни аранжман. (ИА)



Слика 9 СК2 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

4.1. Уколико систем не може да пронађе цветни аранжман, кориснику се приказује порука : „ Десила се грешка“ (ИА)



Слика 10 СК2 - алтернативни сценарио, дијаграми секвенци

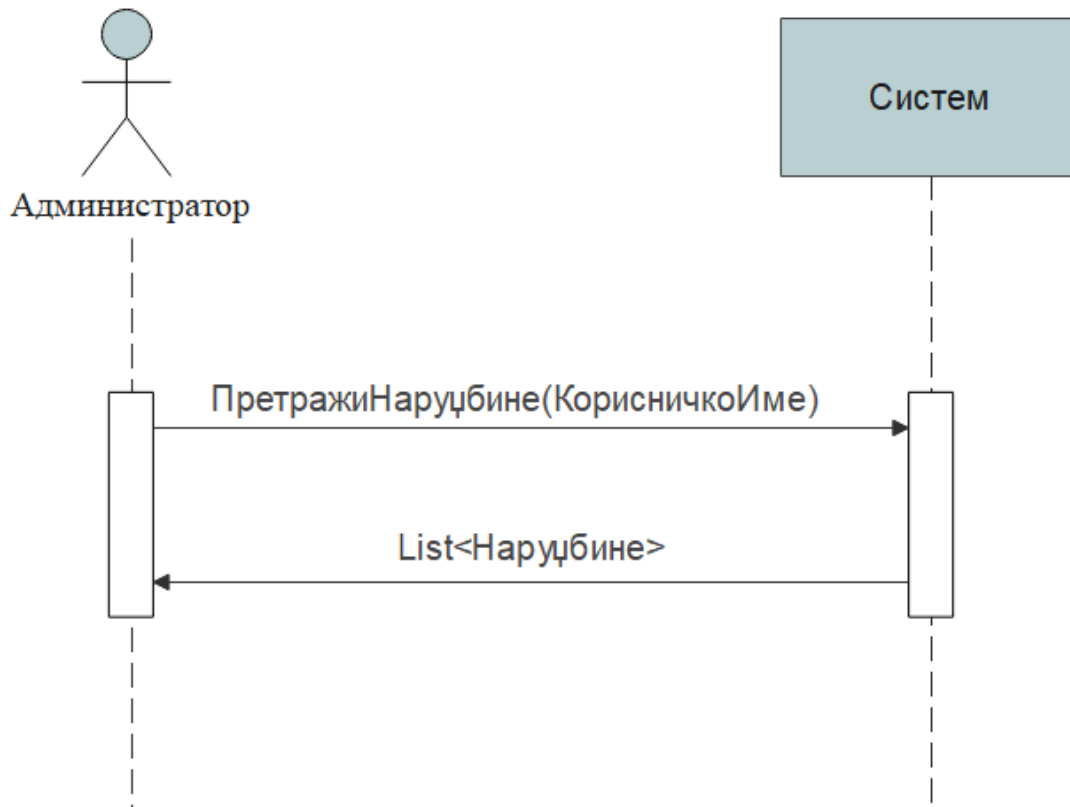
Са наведених дијаграма секвенци уочавају се две системске операције:

1. Сигнал ПрикажиЦветниАранжман(ЦветниАранжман)
2. Сигнал ВратиЦветнеАранжмане(List)

СКЗ : Случај коришћења – Претраживање наруџбина

Основни сценарио СК

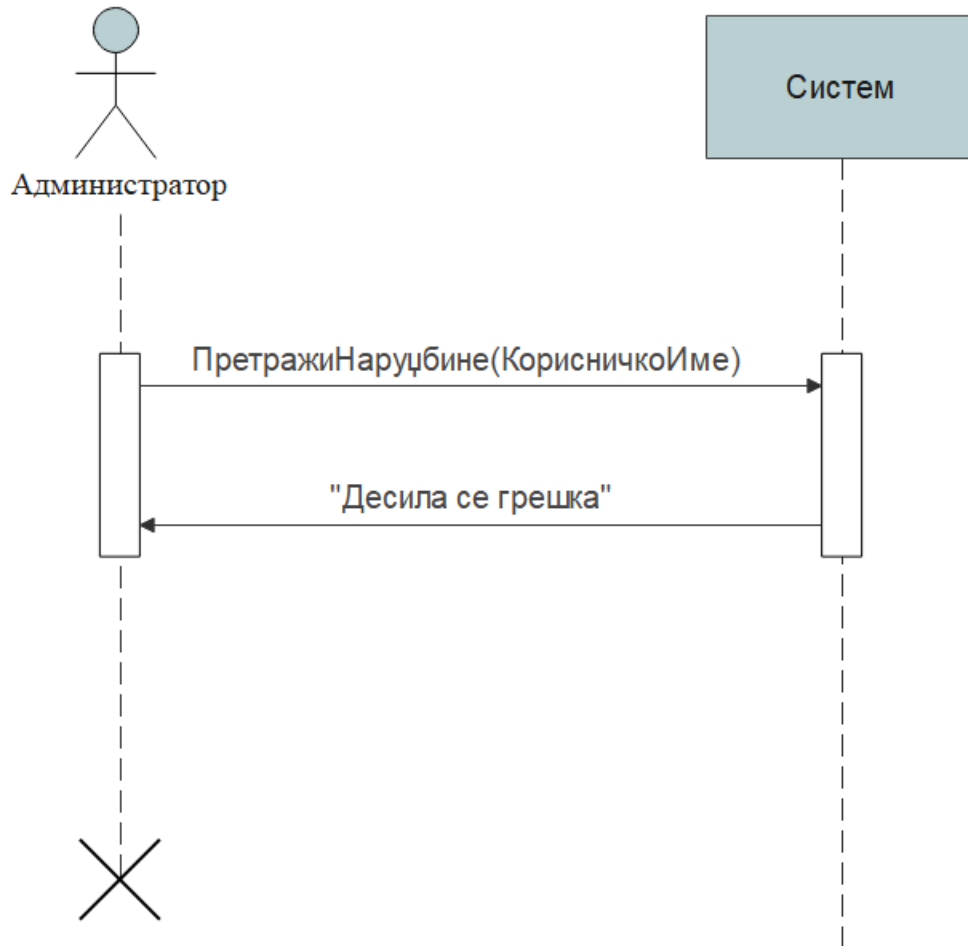
1. Администратор **уноси** критеријум претраживања наруџбина. (АПУСО)
2. Администратор **контролише** да ли је коректно унео корисничко име за претрагу. (АНСО)
3. Администратор **позива** систем да претражи наруџбине по корисничком имену. (АПСО)
4. Систем **претражује** наруџбине. (СО)
5. Систем **приказује** администратору наруџбине. (ИА)



Слика 11 СКЗ - основни сценарио, дијаграми секвенци

Алтернативна сценарија

5.1. Уколико систем не може да претражи наруџбине, администратору се приказује порука: „Десила се грешка“ (ИА)



Слика 12 СКЗ - алтернативни сценарио, дијаграми секвенци

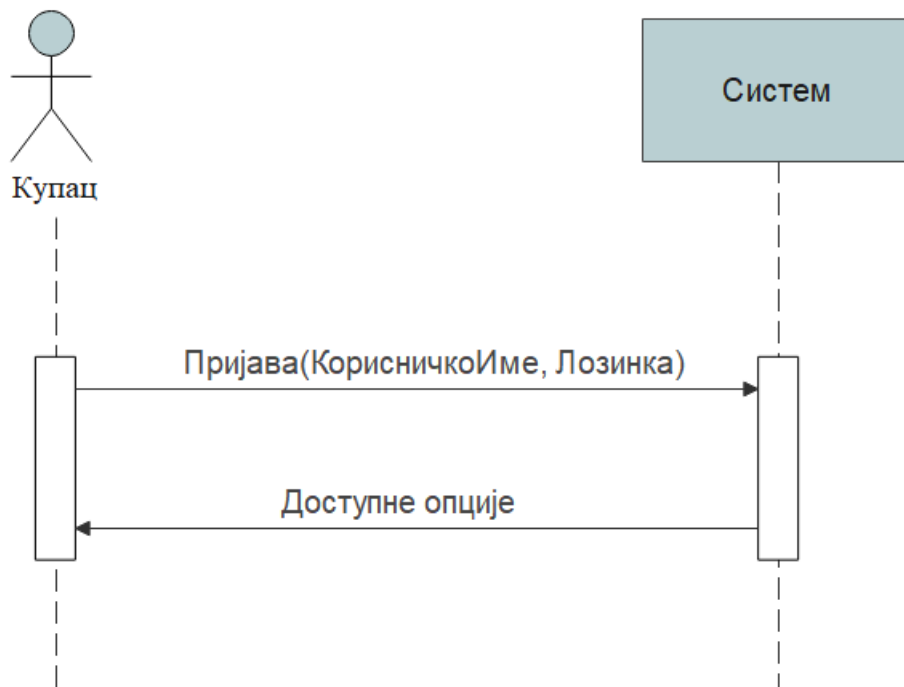
Са наведених дијаграма секвенци уочава се системска операција:

1. Сигнал ПрикажиНаруџбине(КорисничкоИме)

СК4 : Случај коришћења – Пријављивање на систем

Основни сценарио СК

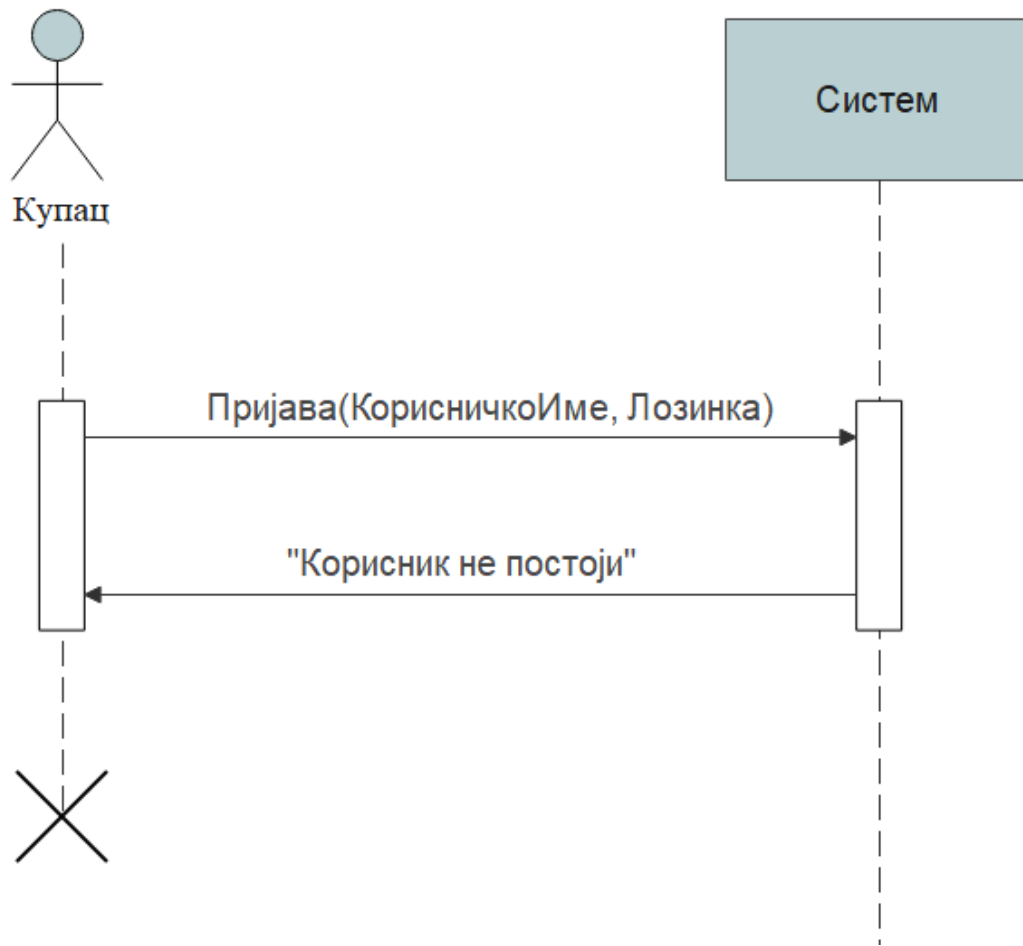
1. Администратор/купац **уноси** податке за пријављивање. (АПУСО)
2. Администратор/купац **контролише** да ли је коректно унео корисничко име и лозинку (АНСО)
3. Администратор/купац **позива** систем да га пријави. (АПСО)
4. Систем **проверава** да ли су унети подаци у реду. (СО)
5. Систем **приказује** администратору/купцу доступне опције. (ИА)



Слика 13 СК4 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

5.1 Уколико систем не може да верификује администратора/купца он приказује поруку: „Корисник не постоји“ (ИА)



Слика 14 СК4 - алтернативни сценарио, дијаграми секвенци

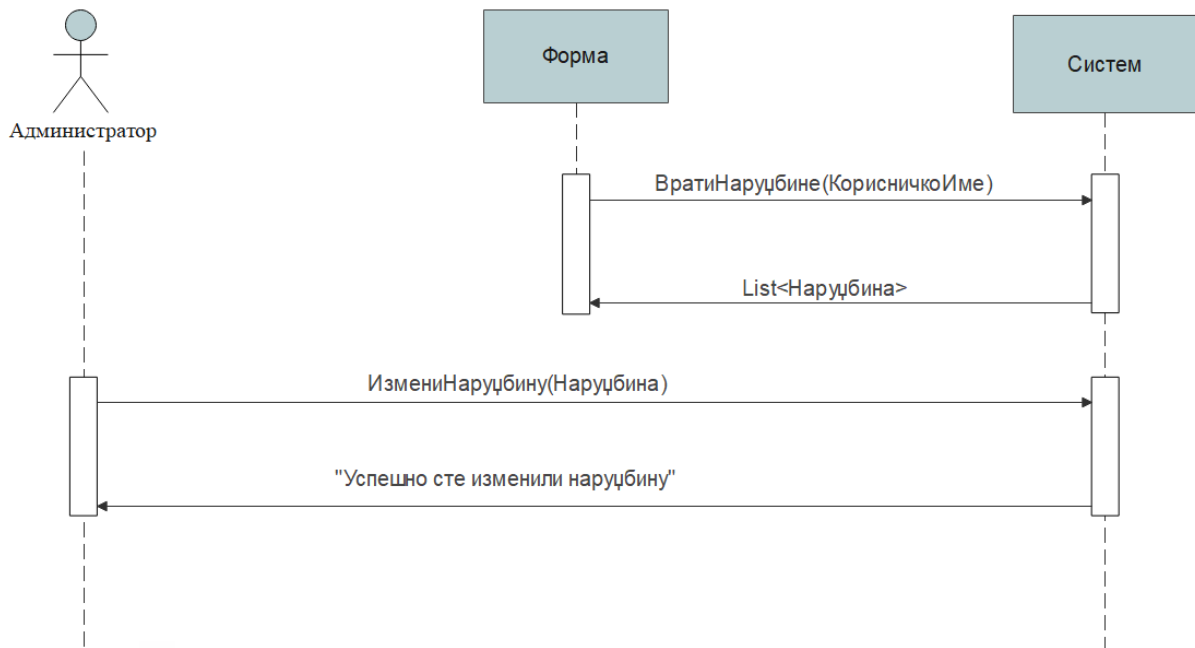
Са наведених дијаграма секвенци уочава се системска операција:

1. Сигнал Пријава(КорисничкоИме,Лозинка)

СК5 : Случај коришћења – Измена наруџбине

Основни сценарио СК

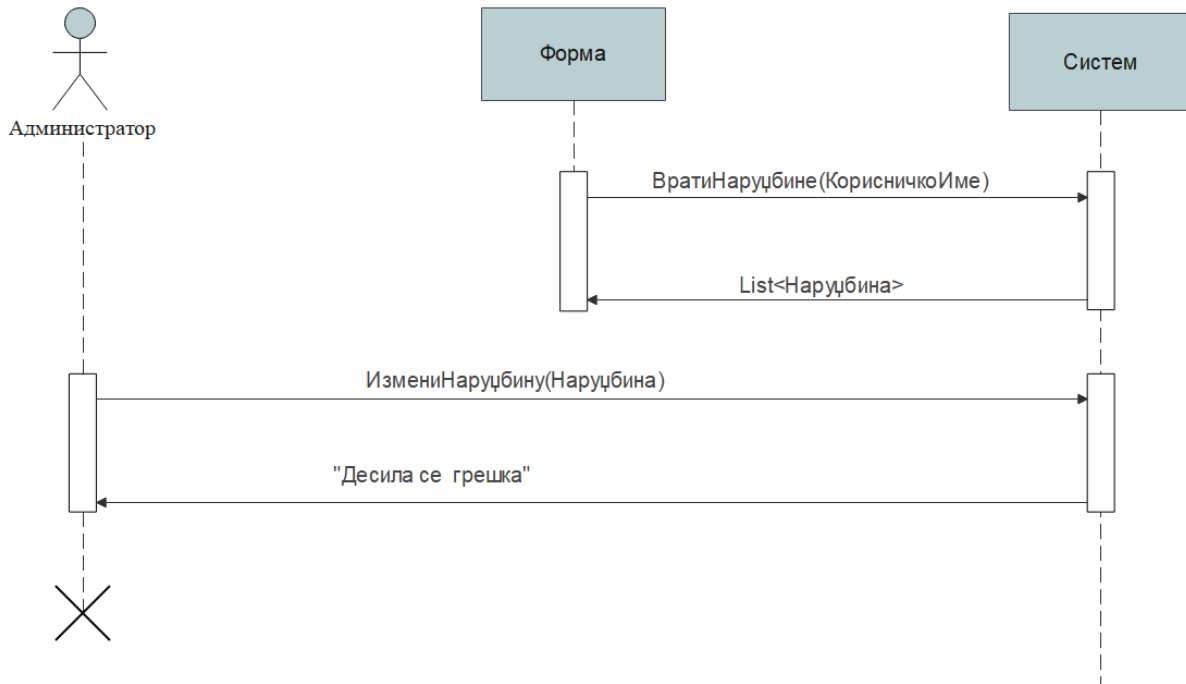
1. Администратор **уноси** критеријум по ком претражује наруџбине. (АПУСО)
2. Администратор **позива** систем да пронађе наруџбине на основу задате вредности. (АПСО)
3. Систем **тражи** наруџбине по задатом критеријуму и учитава податке о њима. (СО)
4. Систем **приказује** администратору пронађене наруџбине. (ИА)
5. Администратор **бира** наруџбину коју жели да измени. (АПУСО)
6. Администратор **мења** податке о наруџбини (АПУСО)
7. Администратор **контролише** да ли је коректно унео податке о наруџбини. (АНСО)
8. Администратор **позива** систем да запамти податке о наруџбини. (АПСО)
9. Систем **памти** податке о наруџбини. (СО)
10. Систем **приказује** администратору забележену измену. (ИА)



Слика 15 СК5 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

10.1 Уколико систем не може да запамти податке о наруџбини он приказује администратору поруку „ Десила се грешка” (ИА)



Слика 16 СК5 - алтернативни сценарио, дијаграми секвенци

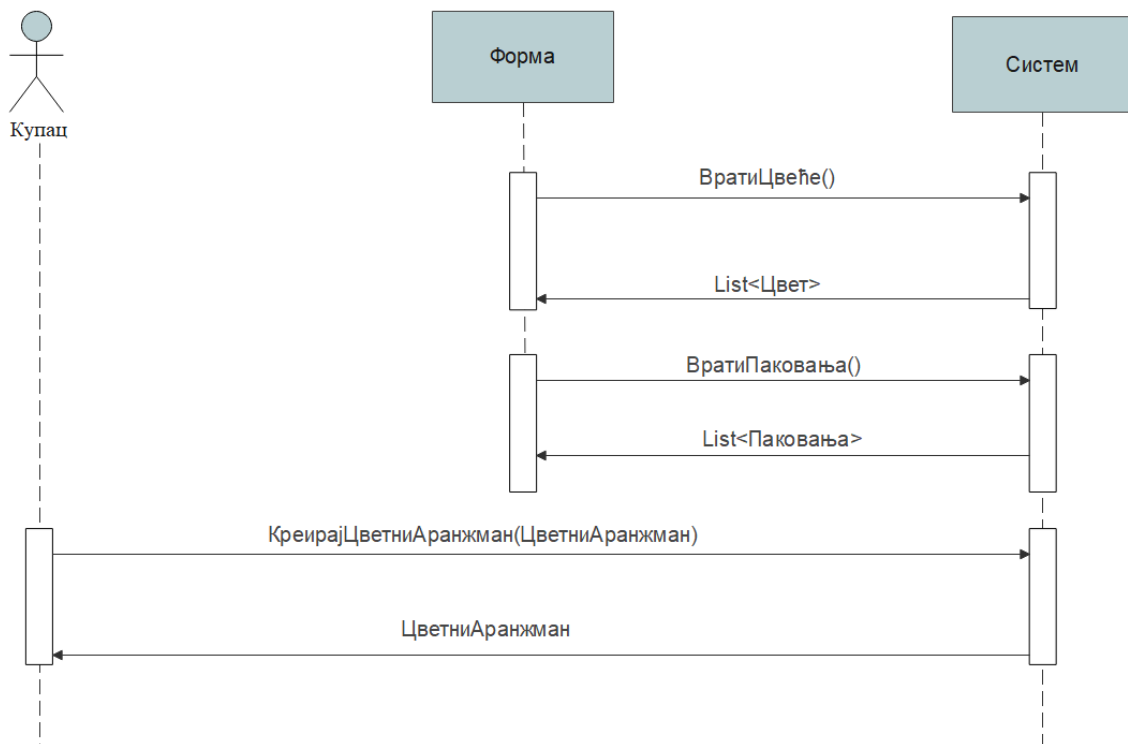
Са наведених дијаграма секвенци уочавају се две системске операције:

1. Сигнал ВратиНаруџбине(КорисничкоИме)
2. Сигнал ИзмениНаруџбину(Наруџбина)

СК6 : Случај коришћења – Креирање цветног аранжмана

Основни сценарио СК

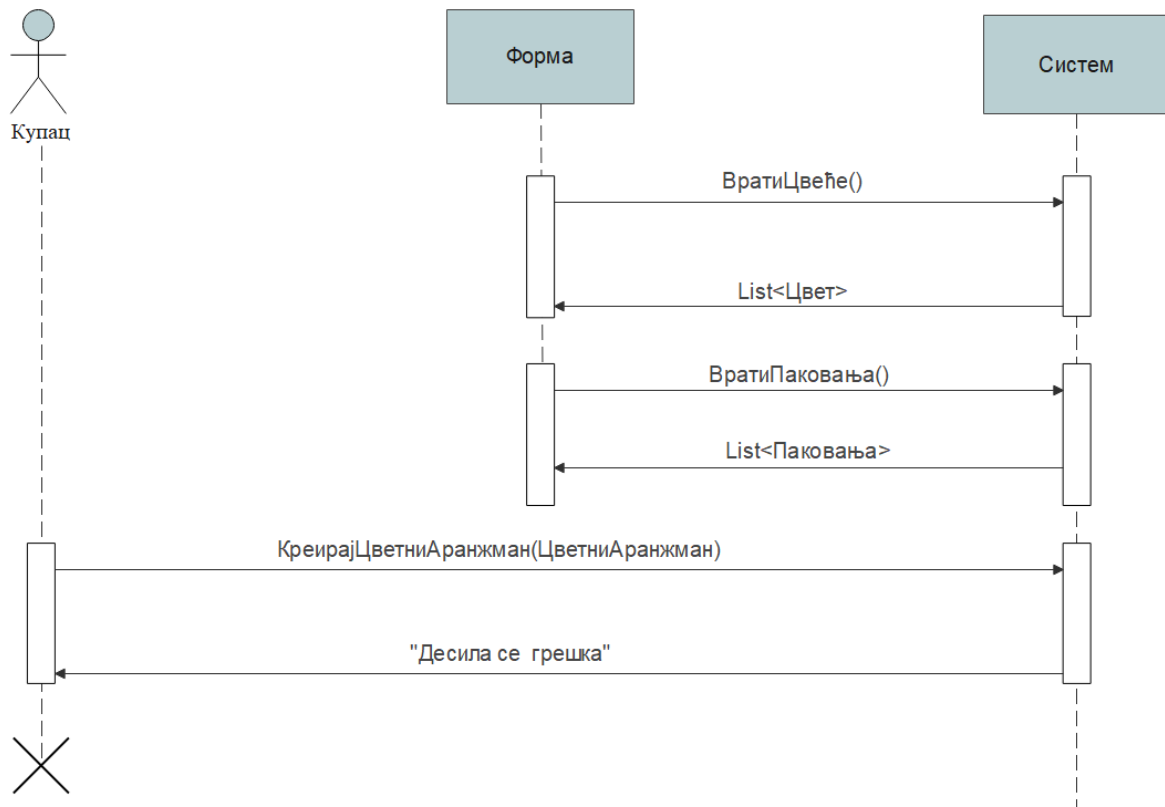
- 1 Купац **уноси** податке о цветном аранжману. (АПУСО)
- 2 Купац **контролише** да ли је коректно унео податке о цветном аранжману. (АНСО)
- 3 Купац **позива** систем да креира нови цветни аранжман са задатим подацима. (АПСО)
- 4 Систем **креира** цветни аранжман са задатим подацима. (СО)
- 5 Систем **приказује** купцу забележени цветни аранжман. (ИА)



Слика 17 СК6 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

5.1 Уколико систем не може да креира цветни аранжман он приказује купцу поруку „Десила се грешка” (ИА)



Слика 18 СКБ - алтернативни сценарио, дијаграми секвенци

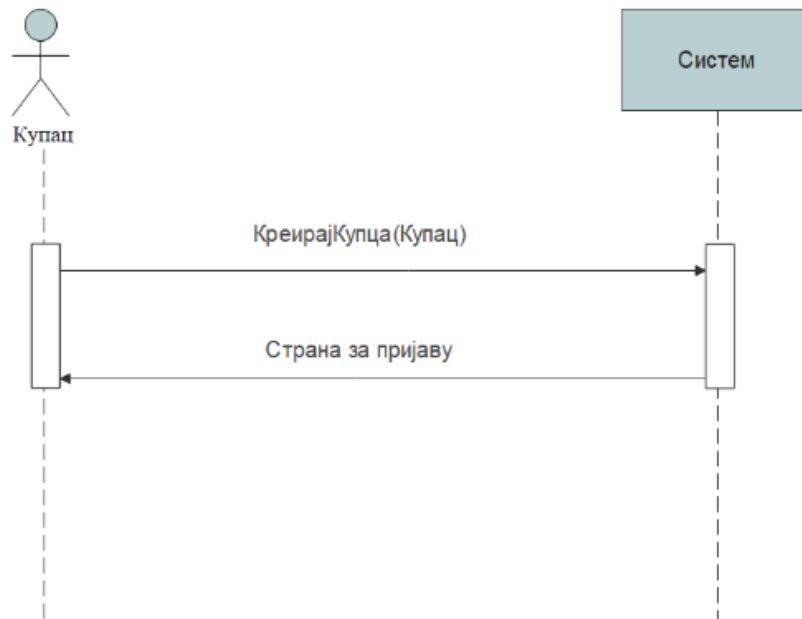
Са наведених дијаграма секвенци учествују се три системске операције:

1. Сигнал ВратиЦвеће()
2. Сигнал ВратиПаковања()
3. Сигнал КреирајЦветниАранжман(ЦветниАранжман)

СК7 : Случај коришћења – Креирање корисничког налога

Основни сценарио СК

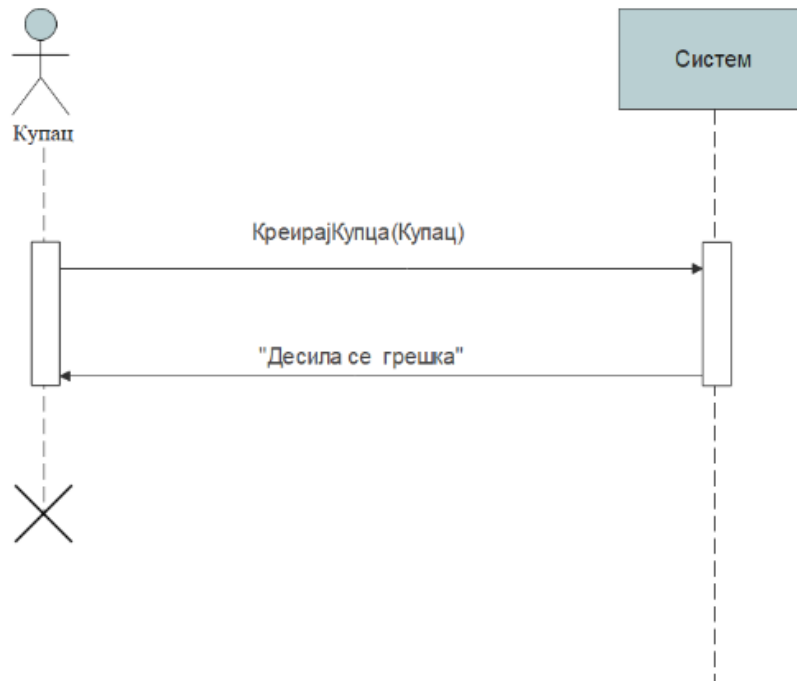
1. Посетилац **уноси** податке о новом купцу. (АПУСО)
2. Посетилац **контролише** да ли је коректно унео податке о купцу. (АНСО)
3. Посетилац **позива** систем да креира новог купца са задатим подацима. (АПСО)
4. Систем **креира** новог купца. (СО)
5. Систем **приказује** посетиоцу страну за пријаву. (ИА)



Слика 19 СК7 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

6.1 Уколико систем не може да креира купца он приказује посетиоцу поруку „ Десила се грешка” (ИА)



Слика 20 СК7 - алтернативни сценарио, дијаграми секвенци

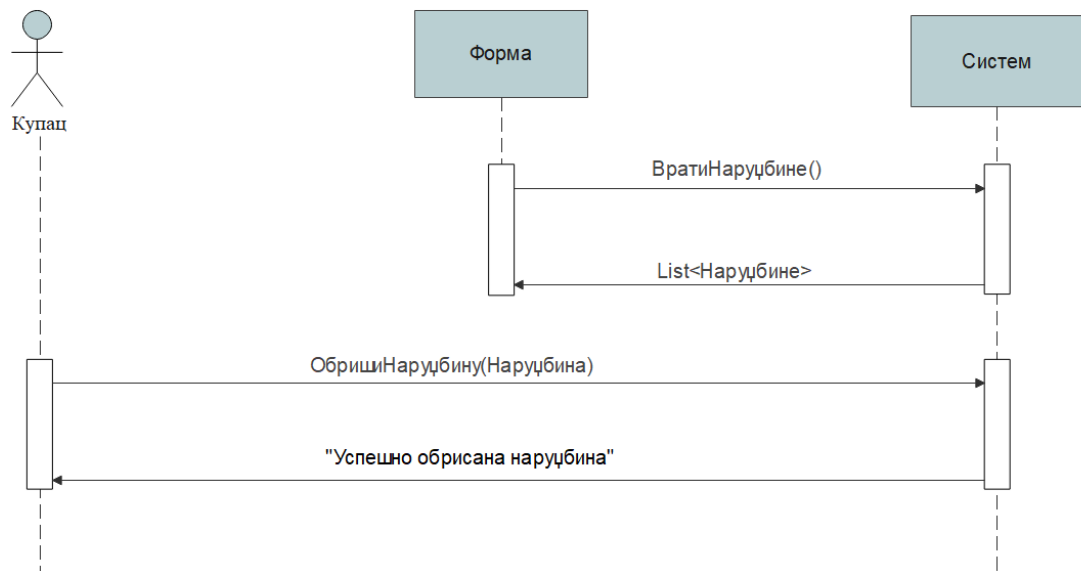
Са наведених дијаграма секвенци уочава се системска операција:

1. Сигнал КреирајКупца(Купац)

СК8 : Случај коришћења – Брисање наруџбине

Основни сценарио СК

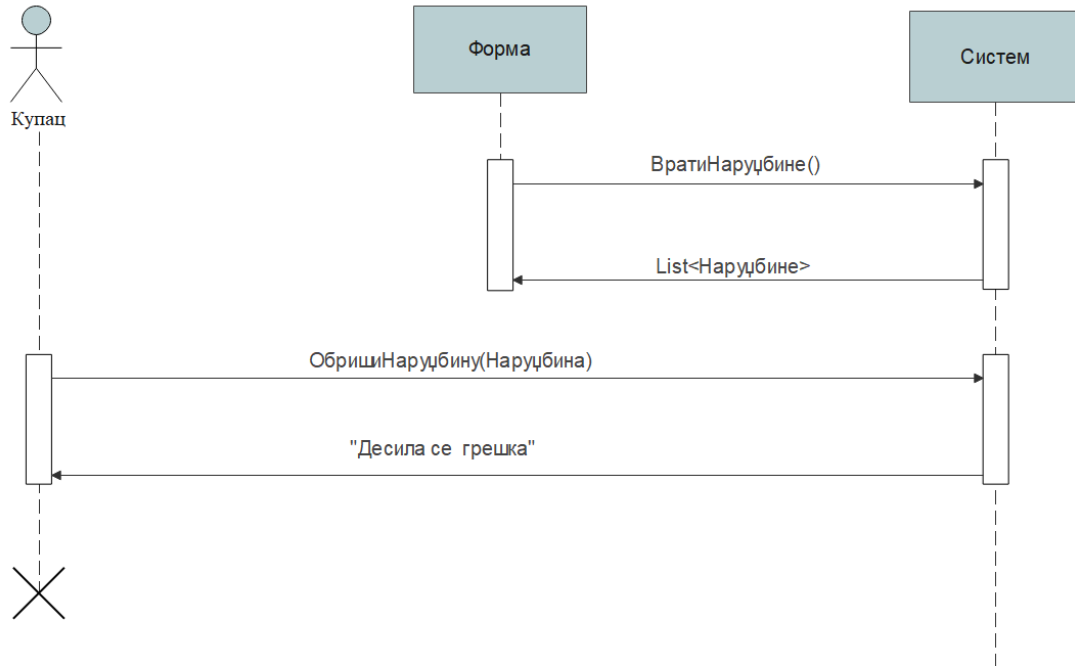
1. Администратор **уноси** критеријум по ком претражује наруџбине. (АПУСО)
2. Администратор **позива** систем да пронађе наруџбине на основу задате вредности. (АПСО)
3. Систем **тражи** наруџбине по задатом критеријуму и учитава податке о њима. (СО)
4. Систем **приказује** администратору пронађене наруџбине. (ИА)
5. Администратор **бира** наруџбину коју жели да обрише. (АПУСО)
6. Администратор **позива** систем да обрише податке о наруџбини. (АПСО)
7. Систем **брише** податке о наруџбини. (СО)
8. Систем **приказује** администратору поруку: „Успешно је обрисана наруџбина.” (ИА)



Слика 21 СК8 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

9. Уколико систем не може да обрише податке о наруџбини он приказује администратору поруку „ Десила се грешка” (ИА)



Слика 22 СК8 - алтернативни сценарио, дијаграми секвенци

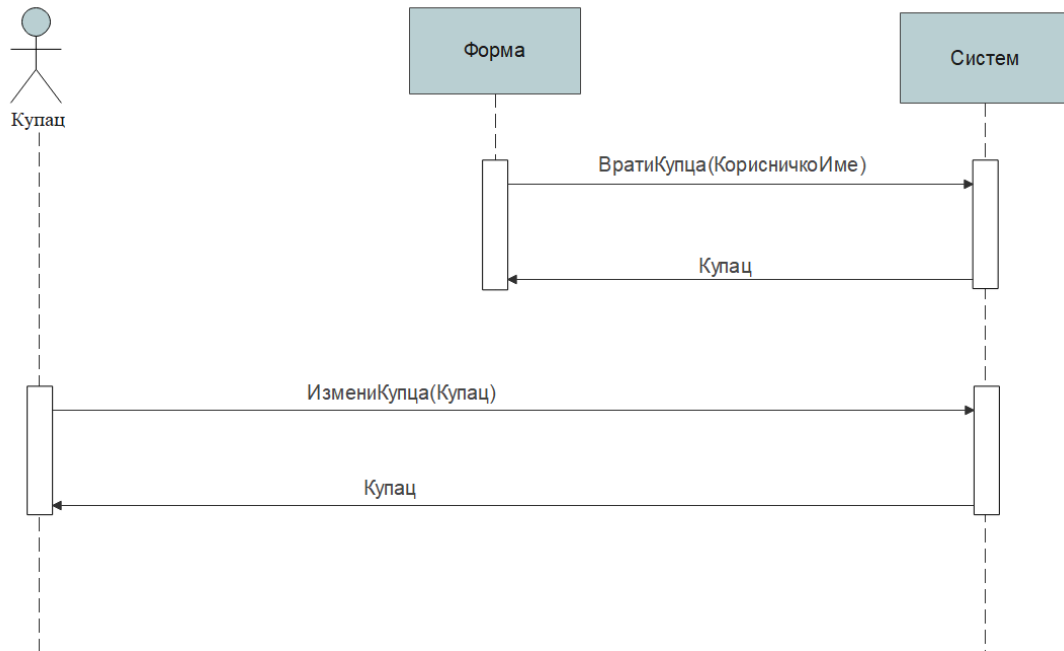
Са наведених дијаграма секвенци уочавају се две системске операције:

1. Сигнал ВратиНаруџбине(КорисничкоИме)
2. Сигнал ОбришиНаруџбину(Наруџбина)

СК9 : Случај коришћења – Измена корисничког налога

Основни сценарио СК

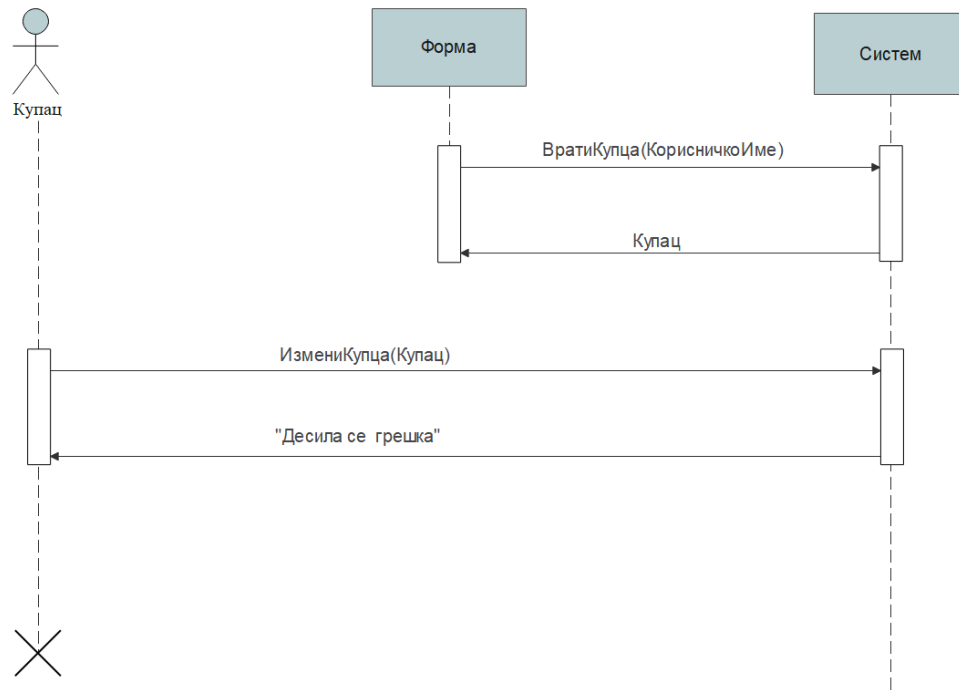
1. Систем **тражи** податке о купцу. (СО)
2. Систем **приказује** купцу пронађене податке. (ИА)
3. Купац **мења** податке. (АПУСО)
4. Купац **контролише** да ли је коректно унео податке о купцу. (АНСО)
5. Купац **позива** систем да запамти податке. (АПСО)
6. Систем **памти** податке о купцу. (СО)
7. Систем **приказује** посетиоцу поруку: „ Успешно сте изменили податке “. (ИА)



Слика 23 СК9 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

7.1 Уколико систем не може да запамти податке о купцу он приказује купцу поруку „Десила се грешка” (ИА)



Слика 24 СК9- алтернативни сценарио, дијаграми секвенци

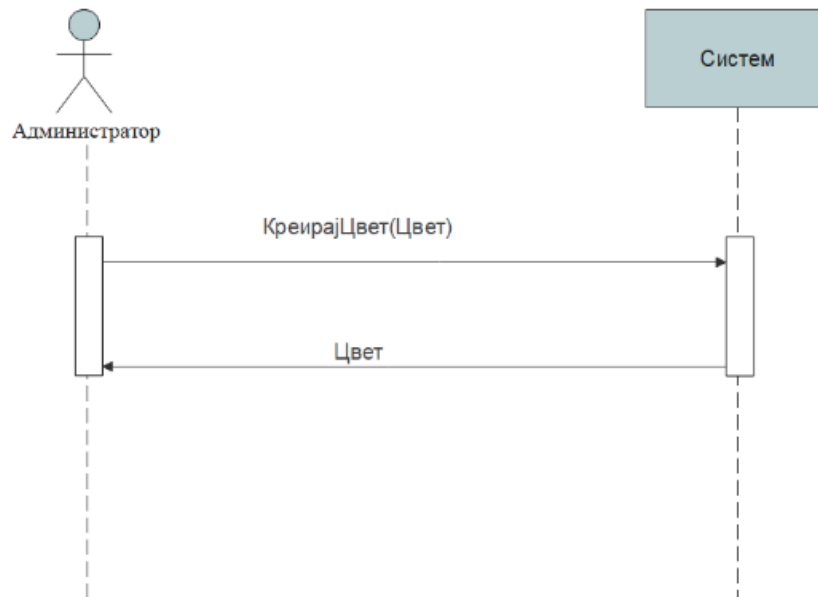
Са наведених дијаграма секвенци уочавају се две системске операције:

1. Сигнал ВратиКупца(КорисничкоИме)
2. Сигнал ИзмениКупца(Купац)

СК10 : Случај коришћења – Креирање цвета

Основни сценарио СК

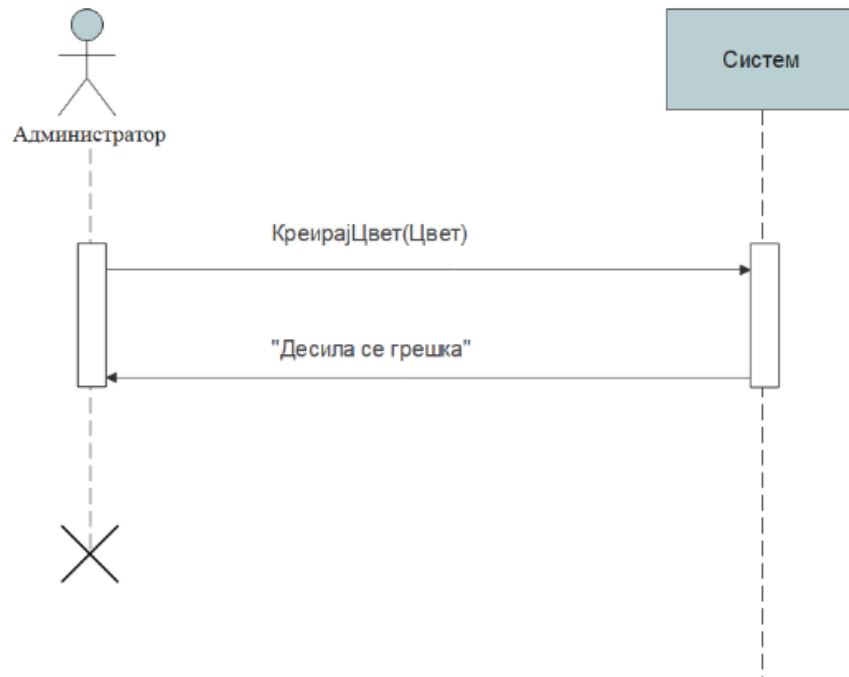
1. Администратор **уноси** податке о цвету. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке о цвету. (АНСО)
3. Администратор **позива** систем да креира нови цвет са задатим подацима. (АПСО)
4. Систем **креира** цвет са задатим подацима. (СО)
5. Систем **приказује** администратору поруку: „Успешно сте креирали цвет“. (ИА)



Слика 25 СК10 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

5.1 Уколико систем не може да креира цвет он приказује администратору поруку „Десила се грешка” (ИА)



Слика 26 СК10 - алтернативни сценарио, дијаграми секвенци

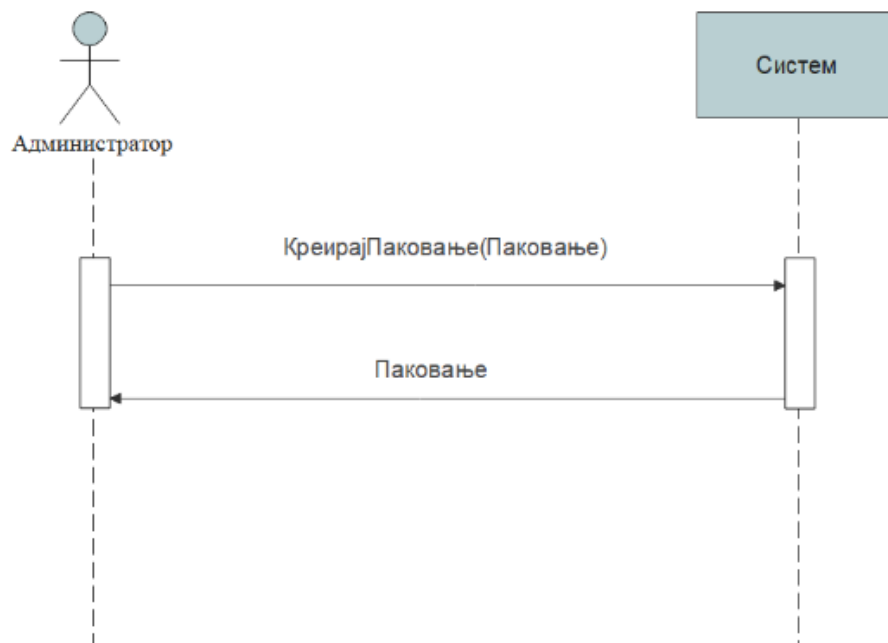
Са наведених дијаграма секвенци уочава се системска операција:

5.1 Сигнал КреирајЦвет(Цвет)

СК11 : Случај коришћења – Креирање паковања

Основни сценарио СК

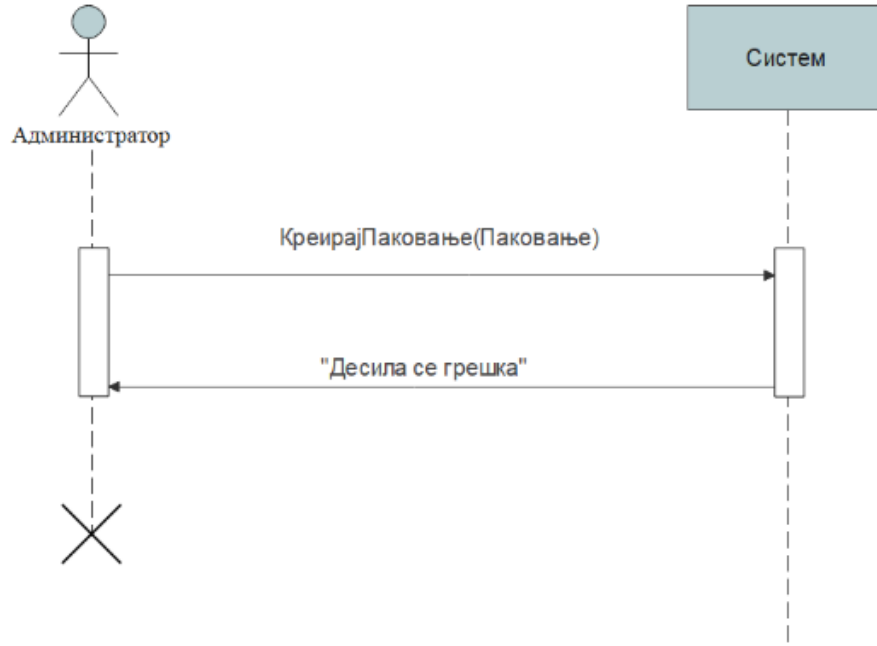
1. Администратор **уноси** податке о паковању. (АПУСО)
2. Администратор **контролише** да ли је коректно унео податке о паковању. (АНСО)
3. Администратор **позива** систем да креира ново паковње са задатим подацима. (АПСО)
4. Систем **креира** паковање са задатим подацима. (СО)
5. Систем **приказује** администратору поруку: „Успешно сте креирали паковање“. (ИА)



Слика 27 СК11 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

- 5.1 Уколико систем не може да креира паковање он приказује администратору поруку „Десила се грешка“ (ИА)



Слика 28 СК11 - алтернативни сценарио, дијаграми секвенци

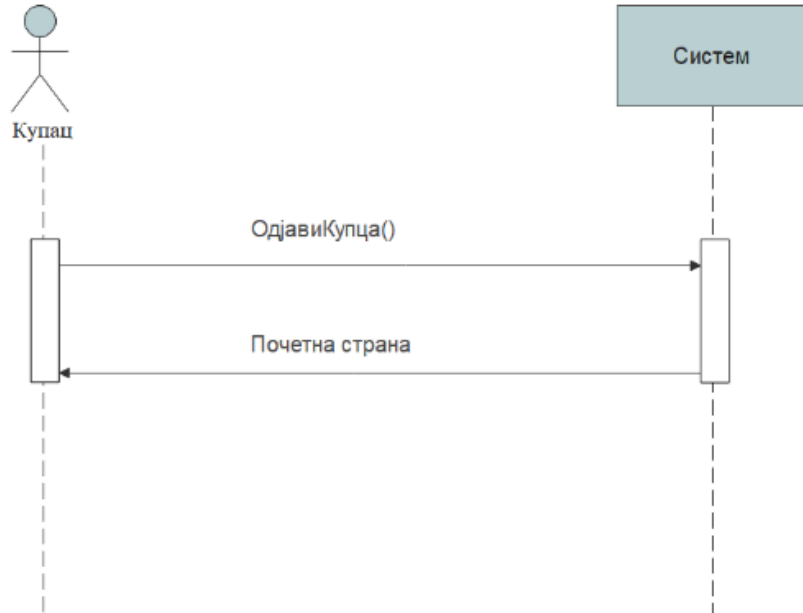
Са наведених дијаграма секвенци уочава се системска операција:

1. Сигнал КреирајПаковање(Паковање)

СК12 : Случај коришћења – Одјава са система

Основни сценарио СК

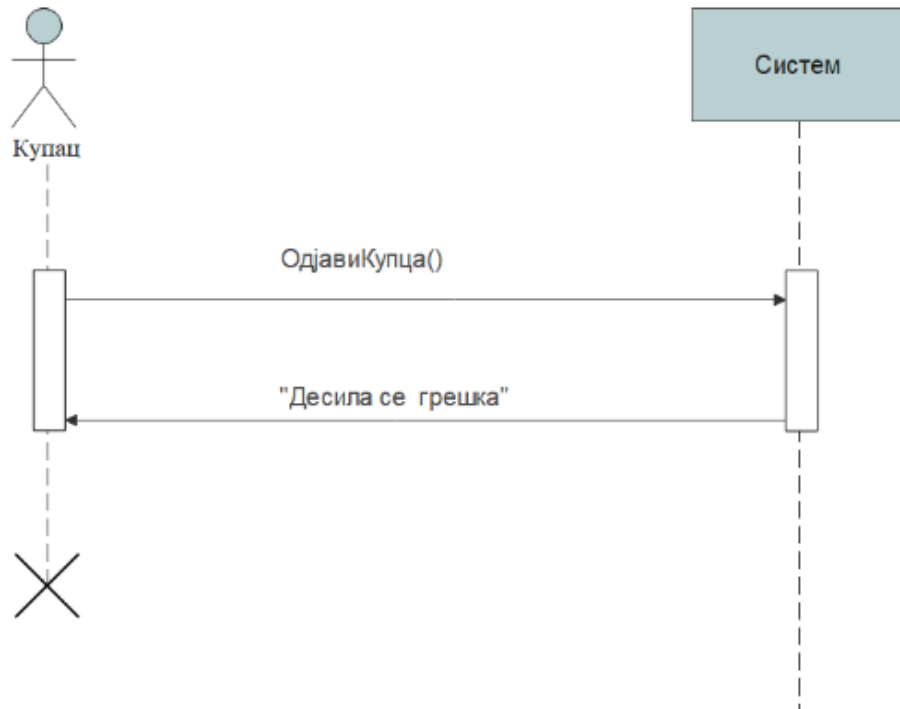
1. Администратор/купац позива систем да га одјави. (АПСО)
2. Систем одјављује администратора/купца са система и приказује почетну страну. (ИА)



Слика 29 СК12 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

2.1 Уколико систем не може да одјави администратора/купца са система: „Десила се грешка”. (ИА)



Слика 30 СК12 - алтернативни сценарио, дијаграми секвенци

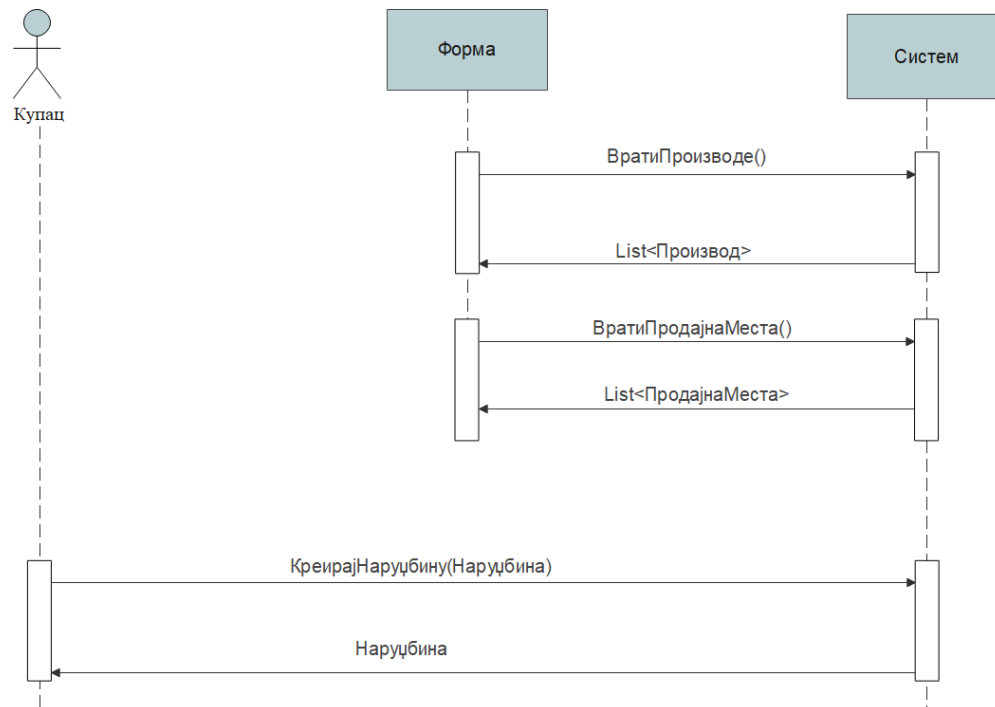
Са наведених дијаграма секвенци уочава се системска операција:

1. Сигнал ОдјавиКупца()

СК13 : Случај коришћења – Креирање наруџбине

Основни сценарио СК

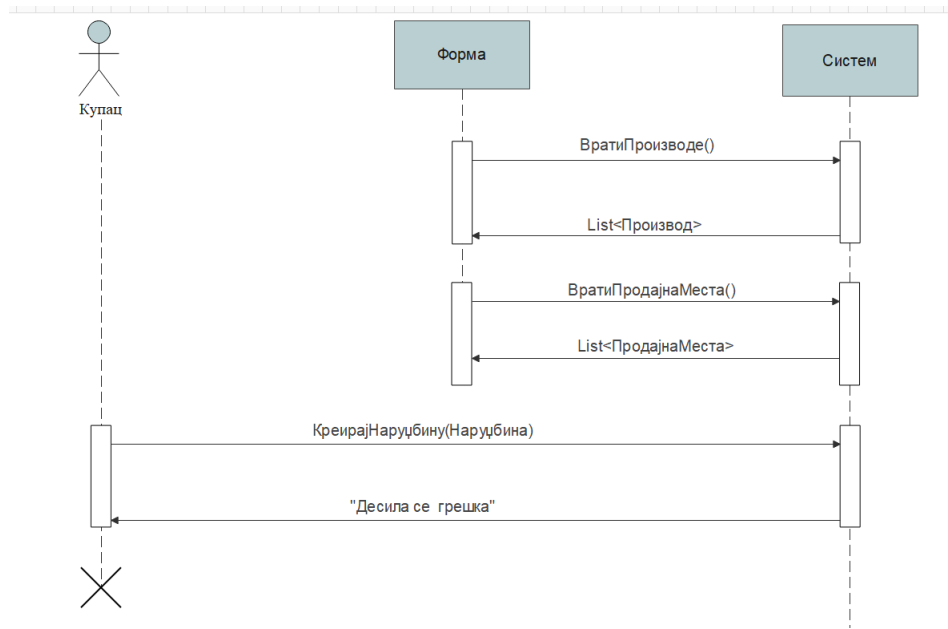
- 1 Купац **уноси** податке за нову наруџбину. (АПУСО)
- 2 Купац **контролише** да ли је коректно унео податке о наруџбини. (АНСО)
- 3 Купац **позива** систем да креира нову наруџбину са задатим подацима. (АПСО)
- 4 Систем **креира** наруџбину са задатим подацима. (СО)
- 5 Систем **приказује** купцу забележену наруџбину. (ИА)



Слика 31 СК13 - основни сценарио, дијаграми секвенци

Алтернативна сценарија

5.1 Уколико систем не може да креира нарудбину он приказује купцу поруку „Десила се грешка” (ИА)



Слика 32 СК13 - алтернативни сценарио, дијаграми секвенци

Са наведених дијаграма секвенци уочавају се три системске операције:

1. Сигнал ВратиПроизводе()
2. Сигнал ВратиПродајнаМеста()
3. Сигнал КреирајНарудбину (Нарудбина)

5.2 Закључак на основу дијаграма секвенци случаја коришћења

Као резултат анализе сценарија добијено је укупно 20 системских операција које треба пројектовати:

1. Сигнал ПрикажиЦвет(Цвет)
2. Сигнал ПрикажиНаруџбине(КорисничкоИме)
3. Сигнал ПрикажиЦветниАранжман(ЦветниАранжман)
4. Сигнал ВратиЦвеће(List<Цвет>)
5. Сигнал ВратиЦветнеАранжмане(List<ЦветниАранжман>)
6. Сигнал ВратиНаруџбине(КорисничкоИме)
7. Сигнал ВратиПаковања(List<Паковање>)
8. Сигнал ВратиКупца(КорисничкоИме)
9. Сигнал ВратиПроизводе(List<Производ>)
10. Сигнал ВратиПродајнаМеста(List<ПродајноМесто>)
11. Сигнал ИзмениНаруџбину(Наруџбина)
12. Сигнал ИзмениКупца(Купац)
13. Сигнал КреирајЦветниАранжман(ЦветниАранжман)
14. Сигнал КреирајКупца(Купац)
15. Сигнал КреирајЦвет(Цвет)
16. Сигнал КреирајПаковање(Паковање)
17. Сигнал КреирајНаруџбину(Наруџбина)
18. Сигнал ОбришиНаруџбину(Наруџбина)
19. Сигнал Пријава(КорисничкоИме,Лозинка)
20. Сигнал ОдјавиКупца()

5.3 Понашање софтверског система – Дефинисање уговора о системским операцијама

Уговори се праве за системске операције и они описују њено понашање. Уговори описују шта операција треба да ради, без објашњења како ће то да ради. Један уговор је везан за једну системску операцију [1].

Уговор УГ1: ПрикажиЦвет

Операција: ПрикажиЦвет(Цвет)

Веза са СК: СК1

Предуслов: /

Постуслов: /

Уговор УГ2: ПрикажиНаруџбине

Операција: ПрикажиНаруџбине(КорисничкоИме)

Веза са СК: СК3

Предуслов: /

Постуслов: /

Уговор УГ3: ПрикажиЦветниАранжман

Операција: ПрикажиЦветниАранжман(ЦветниАранжман)

Веза са СК: СК2

Предуслов: /

Постуслов: /

Уговор УГ4: ВратиЦвеће

Операција: ВратиЦвеће(List<Цвет>)

Веза са СК: СК1, СК6

Предуслов: /

Постуслов: /

Уговор УГ5: ВратиЦветнеАранжмане

Операција: ВратиЦветнеАранжмане(List<ЦветниАранжман>)

Веза са СК: СК2

Предуслов: /

Постуслов: /

Уговор УГ6: ВратиНаруџбине

Операција: ВратиНаруџбине(КорисничкоИме)

Веза са СК: СК8, СК5

Предуслов: /

Постуслов: /

Уговор УГ7: ВратиПаковања

Операција: ВратиПаковања(List<Паковање>)

Веза са СК: СК6

Предуслов: /

Постуслов: /

Уговор УГ8: ВратиКупца

Операција: ВратиКупца(КорисничкоИме)

Веза са СК: СК9

Предуслов: /

Постуслов: /

Уговор УГ9: ВратиПроизводе

Операција: ВратиПроизводе(List<Производ>)

Веза са СК: СК13

Предуслов: /

Постуслов: /

Уговор УГ10: ВратиПродајнаМеста

Операција: ВратиПродајнаМеста(List<ПродајноМесто>)

Веза са СК: СК13

Предуслов: /

Постуслов: /

Уговор УГ11: ИзмениНаруџбину

Операција: ИзмениНаруџбину(Наруџбина)

Веза са СК: СК5

Предуслов: Вредносна и структурна ограничења над објектом Наруџбина морају бити задовољена

Постуслов: Подаци о наруџбини су ажурирани

Уговор УГ12: ИзмениКупца

Операција: ИзмениКупца(Купац)

Веза са СК: СК9

Предуслов: Вредносна и структурна ограничења над објектом Купац морају бити задовољена

Постуслов: Подаци о купцу су ажурирани

Уговор УГ13: КреирајЦветниАранжман

Операција: КреирајЦветниАранжман(ЦветниАранжман)

Веза са СК: СК6

Предуслов: Вредносна и структурна ограничења над објектом ЦветниАранжман морају бити задовољена

Постуслов: Подаци о цветном аранжману су креирани

Уговор УГ14: КреирајКупца

Операција: КреирајКупца(Купац)

Веза са СК: СК7

Предуслов: Вредносна и структурна ограничења над објектом Купац морају бити задовољена

Постуслов: Подаци о купцу су креирани

Уговор УГ15: КреирајЦвет

Операција: КреирајЦвет(Цвет)

Веза са СК: СК10

Предуслов: Вредносна и структурна ограничења над објектом Цвет морају бити задовољена

Постуслов: Подаци о цвету су креирани

Уговор УГ16: КреирајПаковање

Операција: КреирајПаковање(Паковање)

Веза са СК: СК11

Предуслов: Вредносна и структурна ограничења над објектом Паковање морају бити задовољена

Постуслов: Подаци о паковању су креирани

Уговор УГ17: КреирајНаруџбину

Операција: КреирајНаруџбину(Наруџбина)

Веза са СК: СК13

Предуслов: Вредносна и структурна ограничења над објектом Наруџбина морају бити задовољена

Постуслов: Подаци о наруџбини су креирани

Уговор УГ18: ОбришиНаруџбину

Операција: ОбришиНаруџбину(Наруџбина)

Веза са СК: СК8

Предуслов: Структурна ограничења над објектом Наруџбина морају бити задовољена

Постуслов: Подаци о наруџбини су обрисани

Уговор УГ19: Пријава

Операција: Пријава(КорисничкоИме,Лозинка)

Веза са СК: СК4

Предуслов: Морају бити задовољена структурна и вредносна ограничења над објектом Купац/Администратор

Постуслов: Купац/Администратор је пријављен на систем.

Уговор УГ20: Одјава

Операција: Одјава()

Веза са СК: СК12

Предуслов: Морају бити задовољена структурна и вредносна ограничења над објектом Купац/Администратор

Постуслов: Купац/Администратор је одјављен са система.

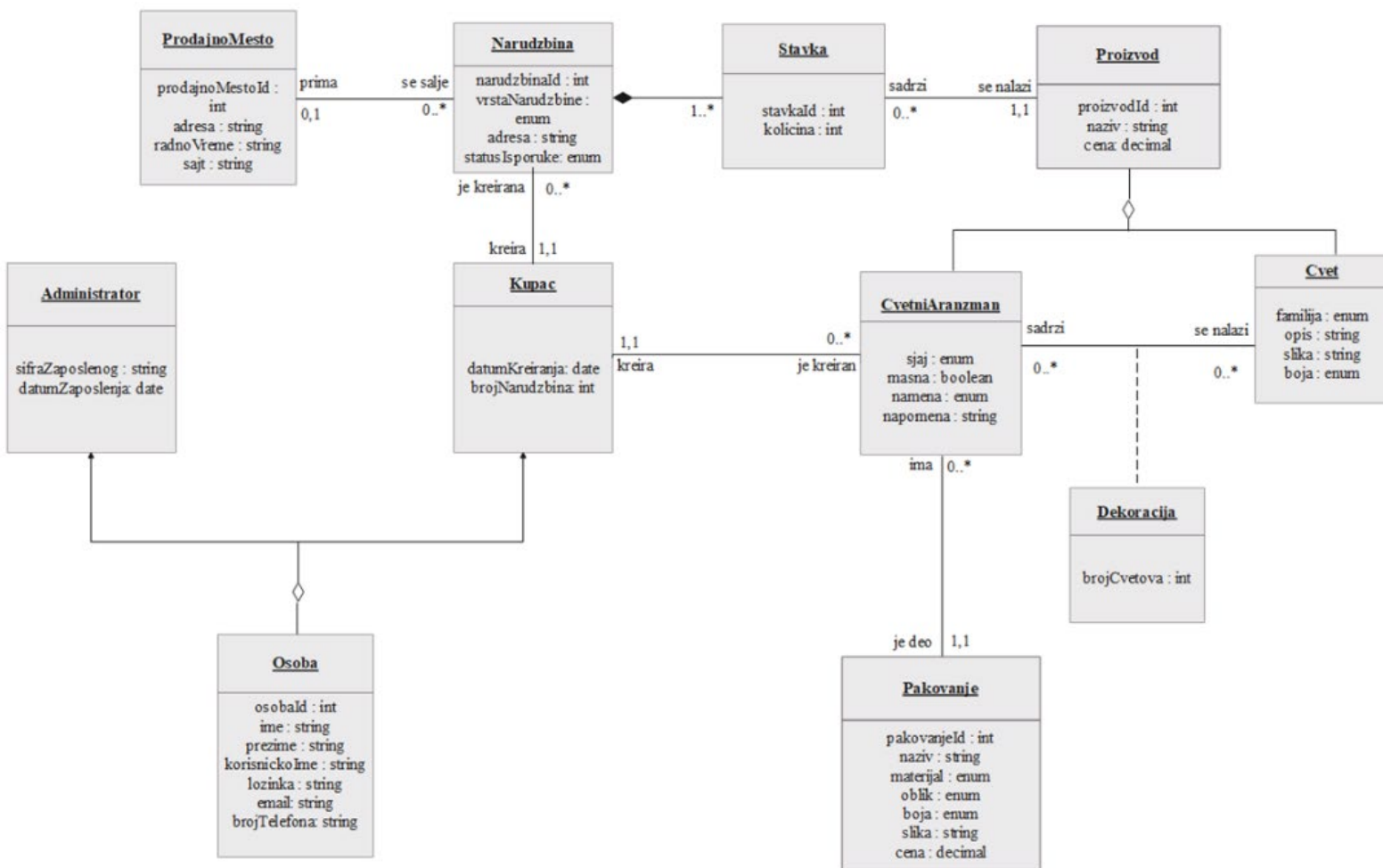
6. ФАЗА ПРОЈЕКТОВАЊА

Фаза пројектовања се израђује на основу сакупљених информација у фази планирања. Фаза пројектовања описује физичку структуру и понашање софтверског система. Пројектовање архитектуре софтверског система обухвата пројектовање три слоја [1]:

- пројектовање корисничког интерфејса
- пројектовање апликационе логике
- пројектовање складишта података

6.1 Структура софтверског система – Концептуални доменски модел

Концептуални модел описује концептуалне класе домена проблема. Концептуални модел садржи концептуалне класе (доменске објекте) и асоцијације између концептуалних класа. Често се за концептуалне моделе каже да су то доменски модели или модели објектне анализе [1]. На наредној слици приказан је концептуални модел:



Slika 1 Концептуални модел

Пројектовање складишта података

Претходно приказан концептуални модел користи се за пројектовање релационог модела, који ће касније бити употребљен и за пројектовање саме базе. Свака табела базе података заправо је једна релација релационог модела, а сваки ред табеле представља својства једног ентитета или везу међу њима.

Администратор(ОсобаИд, ДатумЗапослења,ШифраЗапосленог)

ЦветниАранжман(ПроизводИд, Сјај, Машна, Намена, Напомена, ПаковањеИд, КупацИд)

Цвет(ПроизводИд, Опис, Фамилија, Слика, Боја)

Особа(ОсобаИд, Име, Презиме, Емаил, Лозинка, КорисничкоИме, БројТелефона)

Нарџбина(НарџбинаИд, ВрстаНарџбине, Адреса, СтатусИспоруке, ПродајноМестоИд, КупацИд)

Купац(ОсобаИд, БројНарџбина, ДатумКреирања)

Декорација(ЦветПроизводИд, ЦветниАранжманПроизводИд, БројЦветога)

Ставка(СтавкаИд, НарџбинаИд, ПроизводИд, Количина)

Производ(ПроизводИд, Назив, Цена)

ПродајноМесто(ПродајноМестоИд, Адреса, РадноВреме, Сајт)

Паковање(ПаковањеИд, Назив, Материјал, Облик, Боја, Цена, Слика)

Табела продајноМесто		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT / UPDATE CASCADE Наруџбина DELETE RESTRICTED Наруџбина
	ПродајноМестоИд	Integer	not null			
	Адреса	String	not null			
	РадноВреме	String	not null			
	Сајт	String	not null			

Слика 33 Табела продајно место

Табела купац		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT RESTRICTED Особа UPDATE CASCADE ЦветниАранжман, Наруџбина, RESTRICTED Особа DELETE RESTRICTED ЦветниАранжман, Наруџбина
	ОсобаИд	Integer	not null			
	БројНаруџбина	Integer	not null			
	ДатумКреирања	Date	not null			

Слика 34 Табела купац

Табела декорација		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT RESTRICTED Цвет, ЦветниАранжман UPDATE RESTRICTED Цвет, ЦветниАранжман DELETE /
	ЦветПроизводИд	Integer	not null			
	ЦветниАранжманПроизводИд	Integer	not null			
	БројЦветова	Integer	not null			

Слика 35 Табела декорација

Табела ставка		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT RESTRICTED Производ, Нарудбина UPDATE RESTRICTED Производ, Нарудбина DELETE /
	СтавкаИд	Integer	not null			
	НарудбинаИд	Integer	not null			
	Количина	Float	not null			
	ПроизводИд	Integer	not null			

Слика 36 Табела ставка

Табела производ		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT / UPDATE CASCADES Ставка, Цвет, ЦветниАранжман DELETE RESTRICTED Цвет, ЦветниАранжман
	ПроизводИд	Integer	not null			
	Назив	Integer	not null			
	Цена	Float	not null			

Слика 37 Табела производ

Табела особа		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Вредности атрибута једне табеле	Вредности атрибута више табела	INSERT / UPDATE CASCADES Купац, Администратор DELETE RESTRICTED Купац, Администратор
	ОсобаИд	Integer	not null			
	Име	String	not null			
	Презиме	String	not null			
	КорисничкоИме	String	not null			
	Лозинка	String	not null			
	Емаил	String	not null			
БројТелефона	String	not null				

Слика 38 Табела особа

Табела наруџбина		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT RESTRICTED ПродајноМесто, Купац UPDATE CASCADE Ставка RESTRICTED Купац, ПродајноМесто DELETE RESTRICTED Ставка
	НаруџбинаИд	Integer	not null			
	ВрстаНаруџбине	Integer	not null			
	Адреса	String	not null			
	СтатусИспоруке	Integer	not null			
	ПродајноМестоИд	Integer	not null			
	КупацИд	Integer	not null			

Слика 39 Табела наруџбина

Табела цветниАранжман		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT RESTRICTED Паковање, Купац, Производ UPDATE CASCADE Декорација RESTRICTED Паковање, Купац, Производ DELETE RESTRICTED Декорација
	ПроизводИд	Integer	not null			
	Сјај	Integer	not null			
	Машна	Bit	not null			
	Напомена	String	not null			
	Намена	Integer	not null			
	КупацИд	Integer	not null			
	ПаковањеИд	Integer	not null			

Слика 40 Табела цветни аранжман

Табела паковање		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Вредности атрибута једне табеле	Вредности атрибута више табела	INSERT / UPDATE CASCADES ЦветниАранжман DELETE RESTRICTED ЦветниАранжман
	ПаковањеИд	Integer	not null			
	Материјал	Integer	not null			
	Облик	Integer	not null			
	Цена	Float	not null			
	Назив	String	not null			
	Боја	Integer	not null			
	Слика	String	not null			

Слика 41 Табела паковање

Табела цвет		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT RESTRICTED Производ UPDATE CASCADE Декорација RESTRICTED Производ DELETE RESTRICTED Декорација
	ПроизводИд	Integer	not null			
	Опис	String	not null			
	Фамилија	Integer	not null			
	Слика	String	not null			
	Боја	Integer	not null			

Слика 42 Табела цвет

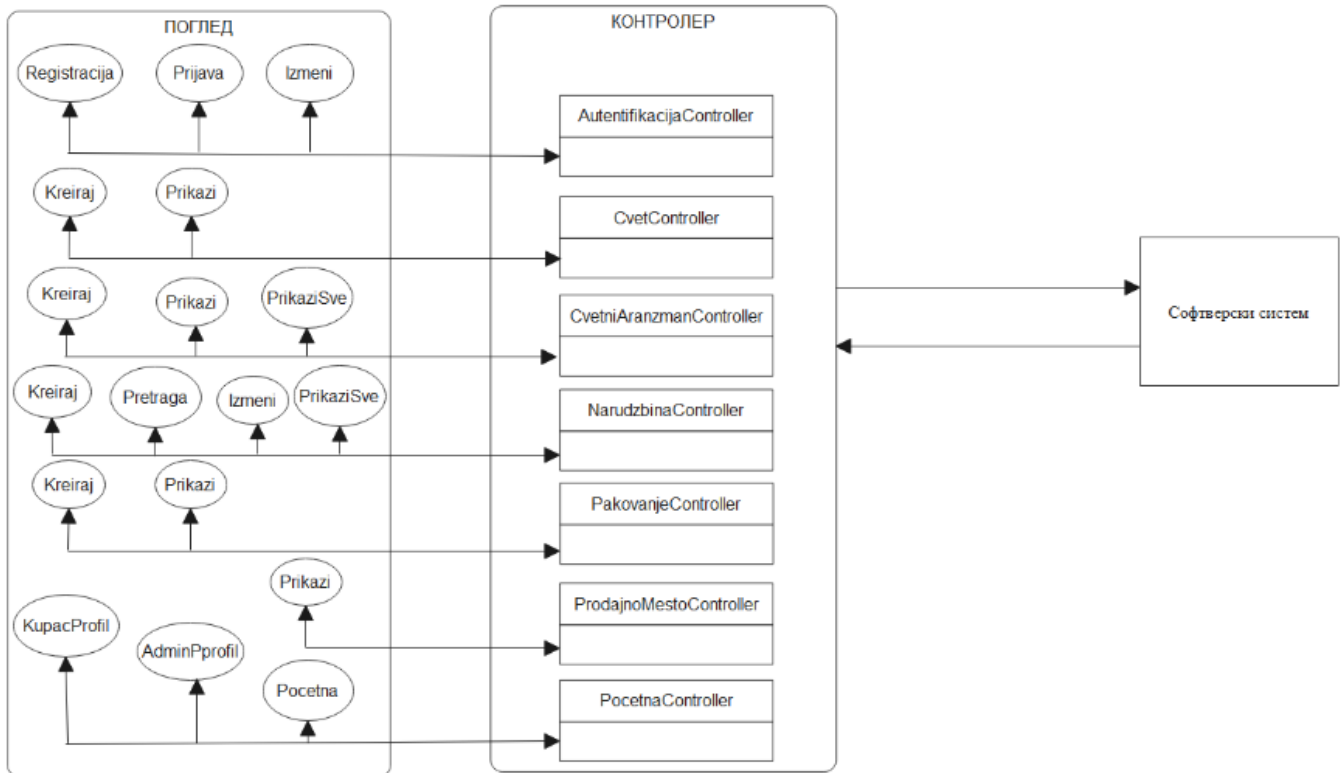
Табела администратор		Просто вредносно ограничење		Сложено вредносно ограничење		Структурно ограничење
Атрибути	Назив Атрибута	Тип атрибута	Вредности атрибута	Међузависности атрибута једне табеле	Међузависности атрибута више табела	INSERT RESTRICTED Особа UPDATE RESTRICTED Особа DELETE /
	ОсобаИд	Integer	not null			
	ШифраЗапосленог	String	not null			
	ДатумЗапослења	Date	not null			

Слика 43 Табела администратор

Пројектовање екранских форми

Кориснички интерфејс представља реализацију улаза и/или излаза софтверског система. Кориснички интерфејс се састоји од [1]:

1. Екранске форме која је одговорна да прихвата податке које уноси актор, прихвата догађаје које прави актор, позива контролера графичког интерфејса, прослеђујући му прихваћене податке, приказује податке које је добио од контролера графичког интерфејса [1].
2. Контролера корисничког интерфејса који је одговоран да прихвати податке које шаље екранска форма, конвертује податке у објекат који представља улазни аргумент СО која ће бити позвана, шаље захтев за извршење СО до апликационог сервера, прихвата објекат софтверског система који настаје као резултат извршења СО и конвертује објекат у податке графичких елемената [1].



Слика 44 Пројектовање екранских форми

СК1: Случај коришћења – Приказивање детаља цвета

Назив СК

Приказивање детаља цвета

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Купцу је приказана листа цвећа.

Odaberite cveće:

Naziv	Boja	Cena		
Ruza	Crvena	200.00	Detalji	Dodaj
Orhideja	Ljubicasta	150.00	Detalji	Dodaj
Muskatla	Crvena	120.00	Detalji	Dodaj

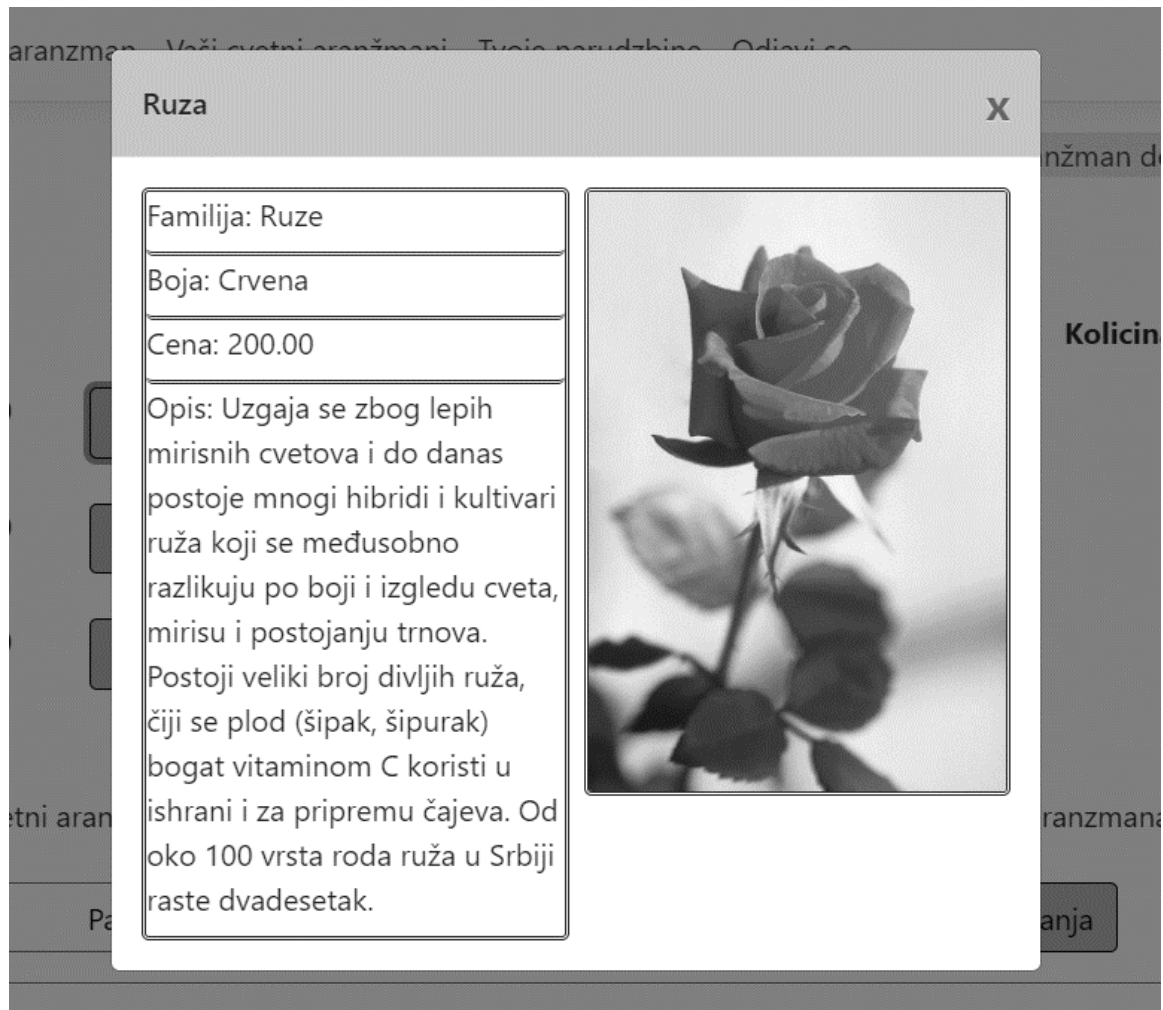
Слика 45 Листа цветова

Основни сценарио СК

1. Купац **бира** цвет коју жели да му се прикаже. (АПУСО)
2. Купац **позива** систем да му прикаже изабрани цвет. (АПСО)

Опис акције: Купац кликом на дугме „Detalji“ позива системску операцију ПрикажиЦвет(Цвет)

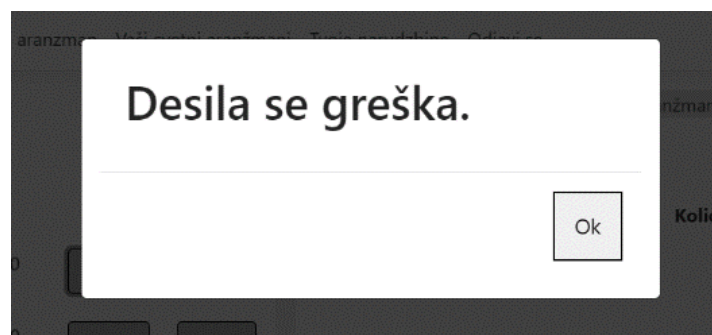
3. Систем **тражи** цвет по задатој вредности. (СО)
4. Систем **приказује** кориснику цвет. (ИА)



Слика 46 Детаљи цвeта

Алтернативна сценарија

4.1. Уколико систем не може да пронађе цвeт, купцу се приказује порука : „Десила се грешка“ (ИА)



Слика 47 Грешка у проналажењу цвeта

СК2: Случај коришћења – Приказивање детаља цветног аранжмана

Назив СК

Приказивање детаља цветног аранжмана

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Купцу је приказана листа цветних аранжмана.

Naziv	Vrsta	Cena	Detalji	Dodaj
Rodjendanski poklon	Cvetni aranzman	1640.00	Detalji	Dodaj
Ljubicica	Cvet	140.00	Detalji	Dodaj

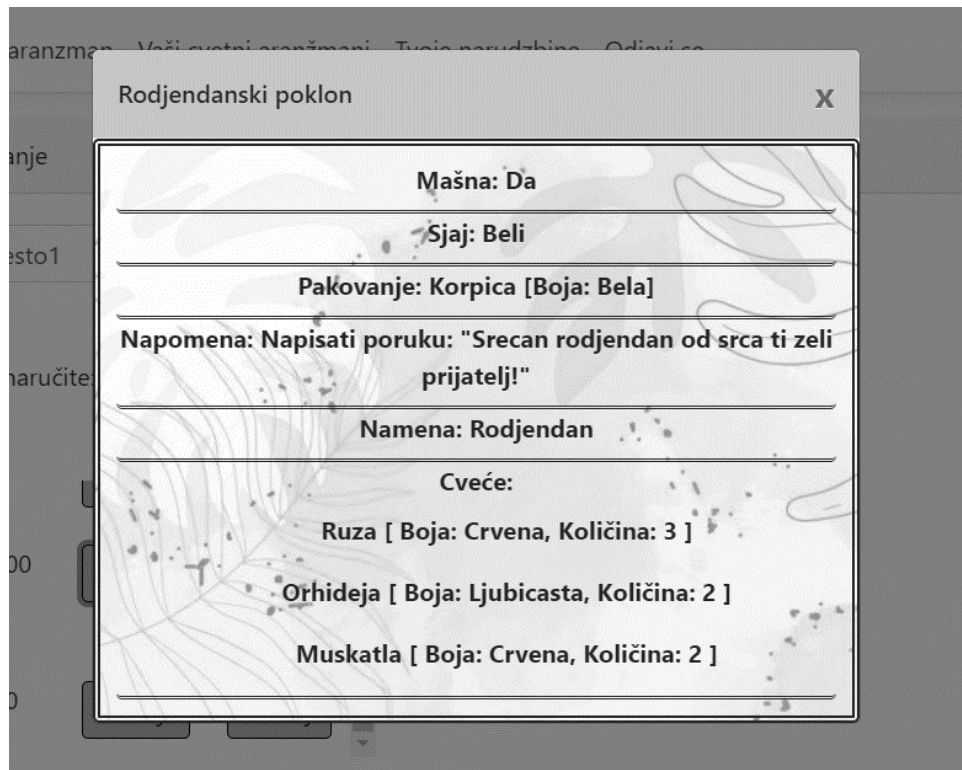
Слика 48 Листа цветних аранжмана

Основни сценарио СК

1. Купац **бира** цветни аранжман коју жели да му се прикаже. (АПУСО)
2. Купац **позива** систем да му прикаже избрани цветни аранжман. (АПСО)

Опис акције: Купац кликом на дугме „Detalji“ позива системску операцију ПрикажиЦветниАранжман(ЦветниАранжман)

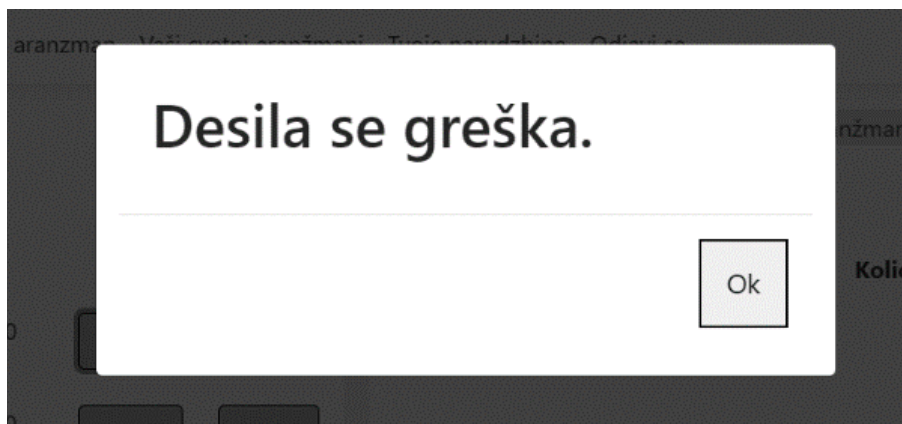
3. Систем **тражи** цветни аранжман по задатој вредности. (СО)
4. Систем **приказује** кориснику цветни аранжман. (ИА)



Слика 49 Детаљи цветног аранжмана

Алтернативна сценарија

4.1. Уколико систем не може да пронађе цветни аранжман, кориснику се приказује порука : „ Десила се грешка“ (ИА)



Слика 50 Грешка проналажења цветног аранжмана

СКЗ : Случај коришћења – Претраживање наруџбина

Назив СК

Претраживање наруџбина

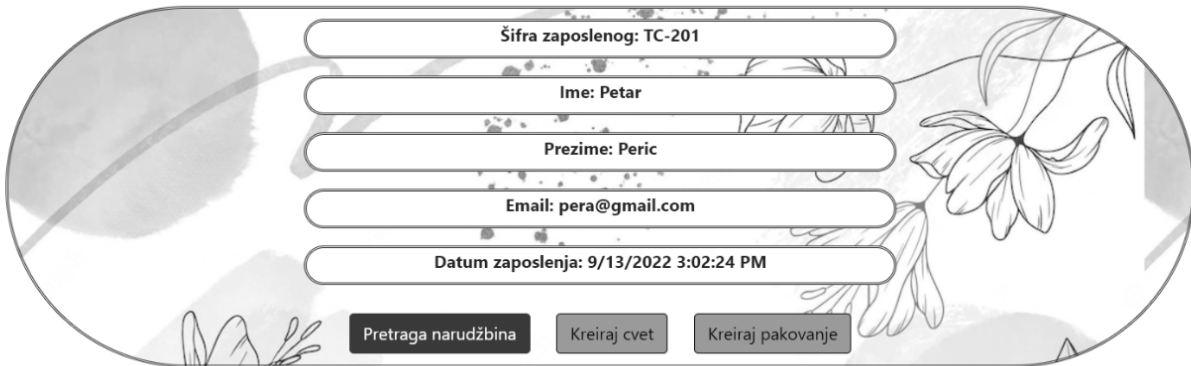
Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Администратор је регистрован на свој налог.



Слика 51 Регистровани администратор

Основни сценарио СК

1. Администратор **уноси** критеријум претраживања наруџбина. (АПУСО)



Слика 52 Претрага наруџбина

2. Администратор **контролише** да ли је коректно унео корисничко име за претрагу. (АНСО)

3. Администратор **позива** систем да претражи наруџбине по корисничком имену. (АПСО)

Опис акције: Администратор кликом на дугме „Pretraži“ позива системску операцију ПрикажиНаруџбине(List<Наруџбина>)

4. Систем **претражује** наруџбине. (СО)
5. Систем **приказује** администратору наруџбине. (ИА)



Слика 53 Приказ наруџбина

Алтернативна сценарија

- 4.1. Уколико систем не може да претражи наруџбине, администратору се приказује порука: „Купац нема наруџбине“ (ИА)

Купац нема наруџбине.

Слика 54 Порука о наруџбинама купца

СК4 : Случај коришћења – Пријављивање на систем

Назив СК

Пријављивање на систем

Актери СК

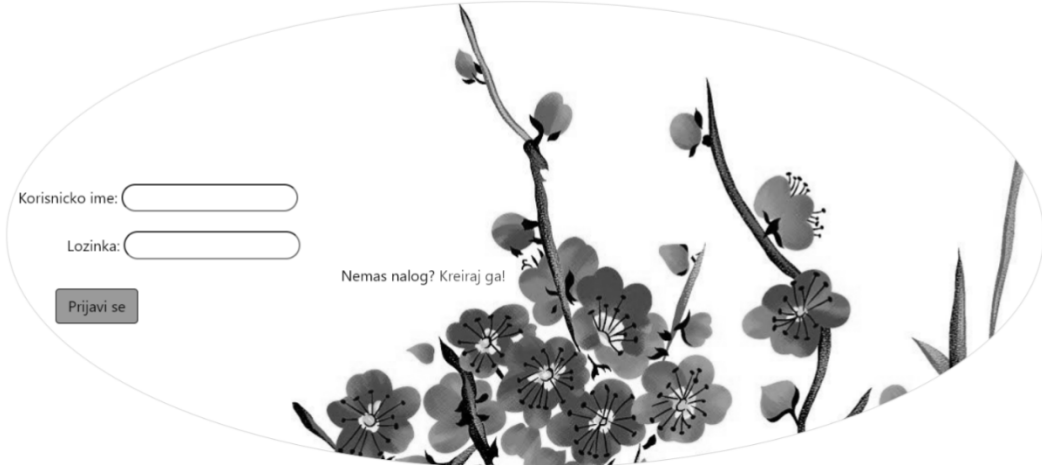
Администратор/купац

Учесници СК

Администратор/купац и систем

Предуслов: Систем је укључен и приказује форму за пријављивање

Lycoris Radiata Prijava



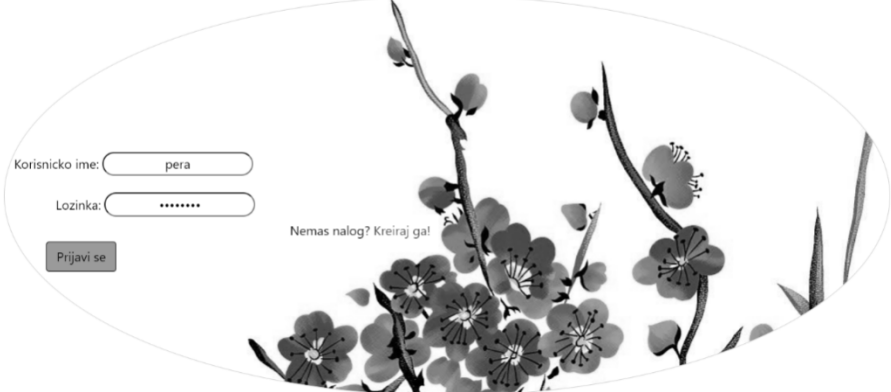
The screenshot shows a login interface within an oval frame. On the left, there are two input fields: 'Korisnicko ime:' and 'Lozinka:'. Below the password field is a 'Prijavi se' button. To the right of the fields, the text 'Nemas nalog? Kreiraj ga!' is displayed. The background of the oval frame features a grayscale illustration of a plant with several flowers and buds.

Слика 55 Форма за пријаву

Основни сценарио СК

1. Администратор/купац **уноси** податке за пријављивање. (АПУСО)

Lycoris Radiata Prijava



Korisnicko ime:

Lozinka:

[Nemas nalog? Kreiraj ga!](#)

Слика 56 Попуњена форма за пријаву

- Администратор/купац **контролише** да ли је коректно унео корисничко име и лозинку (АНСО)
- Администратор/купац **позива** систем да га пријави. (АПСО)

Опис акције: Администратор/купац кликом на дугме „Prijavi se“ позива системску операцију Пријава(КорисничкоИме, Лозинка)

- Систем **проверава** да ли су унети подаци у реду. (СО)
- Систем **приказује** администратору/купцу доступне опције. (ИА)

Lycoris Radiata [Kreiraj cvetni aranžman](#) [Vaši cvetni aranžmani](#) [Tvoje narudzbine](#) [Odjavi se](#) Naruci



Ime: Tatjana

Prezime: Bailovic

Korisnicko ime: taca

Email: bailovict@gmail.com

Broj telefona: 12345678

Слика 57 Приказ профила купца

Алтернативна сценарија

5.1 Уколико систем не може да верификује администратора/купца он приказује поруку: „Корисник не постоји” (ИА)



Слика 58 Приказ поруке да корисник не постоји

СК5 : Случај коришћења – Измена наруџбине

Назив СК

Измена наруџбине

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Учитане су наруџбине и листа опција за измену статуса наруџбине. Систем приказује форму за рад са наруџбином.



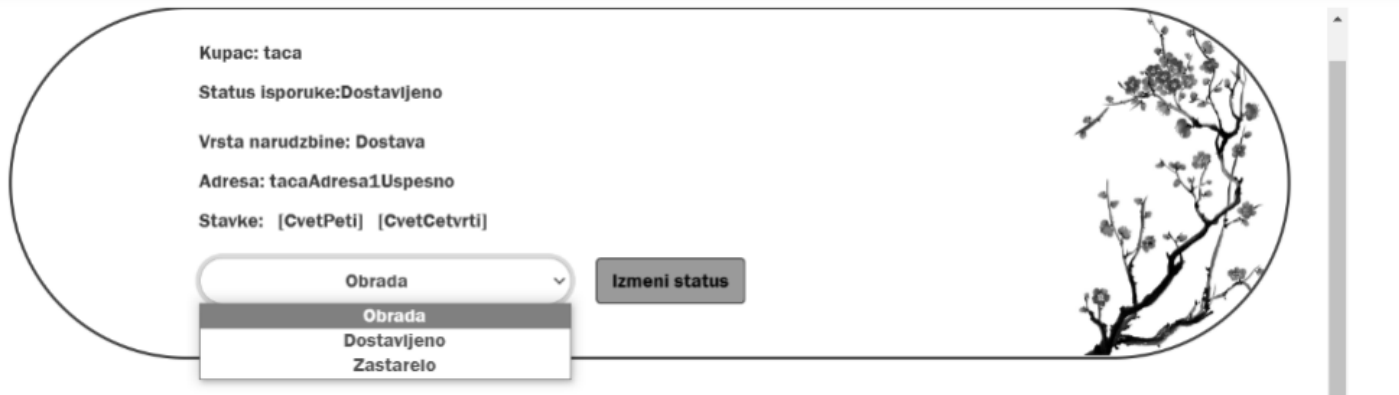
The screenshot shows a web interface for managing orders. It features two rounded rectangular cards, each representing an order. The top card shows an order with status 'Dostavljeno' (Delivered) and a 'Dostava' (Delivery) type. The bottom card shows an order with status 'Zastarelo' (Expired) and a 'Preuzimanje' (Pickup) type. Both cards include a dropdown menu for 'Obrada' (Processing) and buttons for 'Izmeni status' (Change status) and 'Izbrisi' (Delete). The interface is decorated with a floral illustration on the right side of each card.

Order ID	Kupac	Status isporuke	Vrsta narudzbine	Adresa	Stavke	Buttons
1	taca	Dostavljeno	Dostava	tacaAdresa1Uspesno	[CvetPeti] [CvetCetvrti]	Obrada, Izmeni status
2	taca	Zastarelo	Preuzimanje	Mesto1	[Orhideja] [Rodjendanski poklon]	Obrada, Izmeni status, Izbrisi

Слика 59 Листа наруџбина одабраног купца

Основни сценарио СК

1. Администратор **бира** наруџбину коју жели да измени. (АПУСО)
2. Администратор **мења** податке о наруџбини (АПУСО)



Слика 60 Приказ измене наруџбине

3. Администратор **контролише** да ли је коректно унео податке о наруџбини. (АНСО)
4. Администратор **позива** систем да запамти податке о наруџбини. (АПСО)

Опис акције: Администратор кликом на дугме „Izmeni status“ позива системску операцију ИзмениНаруџбину(Наруџбина)

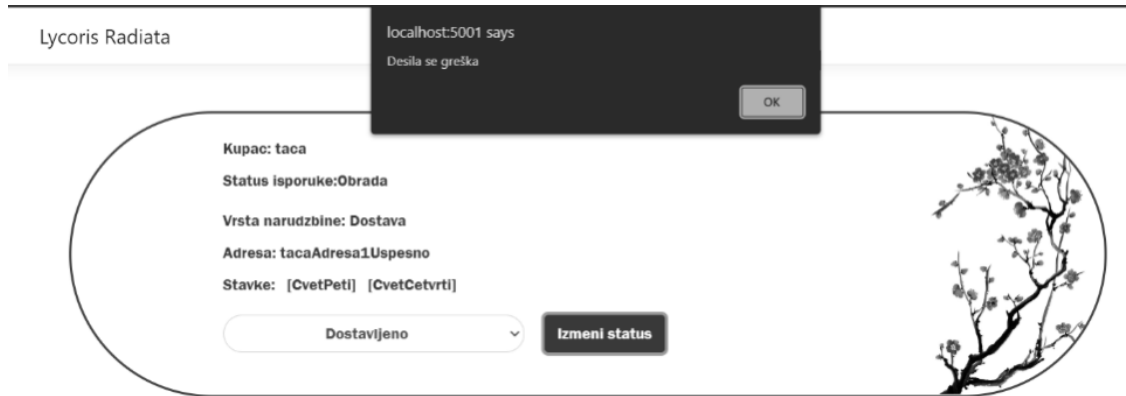
5. Систем **памти** податке о наруџбини. (СО)
6. Систем **приказује** администратору забележену измену. (ИА)



Слика 61 Приказ измењене наруџбине

Алтернативна сценарија

6.1 Уколико систем не може да запамти податке о наруџбини он приказује администратору поруку „Десила се грешка” (ИА)



Слика 62 Приказ грешке при измени наруџбине

СК6 : Случај коришћења – Креирање цветног аранжмана

Назив СК

Креирање цветног аранжмана

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитна је листа цветова и паковања. Систем приказује форму за рад са цветним аранжманом.

Lycoris Radiata Kreiraj цветни aranžman Vaši цветни aranžmani Tvoje narudzbine Odjavi se Naruci

Naziv: Ne možete kreirati цветни aranžman dok ne unesete njegov naziv.

Odaberite цвеће: Odabrano цвеће:

Naziv	Boja	Cena			Naziv	Boja	Kolicina
CvetTreci	Zuta	300.00	<input type="button" value="Detalji"/>	<input type="button" value="Dodaj"/>			
CvetCetvrti	Plava	400.00	<input type="button" value="Detalji"/>	<input type="button" value="Dodaj"/>			
CvetPeti	Ljubicasta	500.00	<input type="button" value="Detalji"/>	<input type="button" value="Dodaj"/>			

Čekiraj polje ako ukoliko da tvoj цветни aranžman ima mašnu: Izaberi sjaj: Namena цветног aranžmana:

Izaberi pakovanje:

Napomena (opciono polje):

Слика 63 Форма за креирање цветног аранжмана

Основни сценарио СК

1. Купац уноси податке о цветном аранжману. (АПУСО)

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se **Naruci**

Naziv:

Odaberite cveće:

Naziv	Boja	Cena	
Ruza	Crvena	200.00	<input type="button" value="Detalji"/>
Orhideja	Ljubicasta	150.00	<input type="button" value="Detalji"/>
Muskatla	Crvena	120.00	<input type="button" value="Detalji"/>

Odabrano cveće:

Naziv	Boja	Kolicina	
Orhideja	Ljubicasta	<input type="text" value="2"/>	<input type="button" value="Ukloni"/>
Muskatla	Crvena	<input type="text" value="2"/>	<input type="button" value="Ukloni"/>
Ruza	Crvena	<input type="text" value="3"/>	<input type="button" value="Ukloni"/>

Čekiraj polje ako ukoliko da tvoj cvetni aranžman ima mašnu: Izaberi sjaj: Namena cvetnog aranžmana:

Izaberi pakovanje:

Napomena (opciono polje):

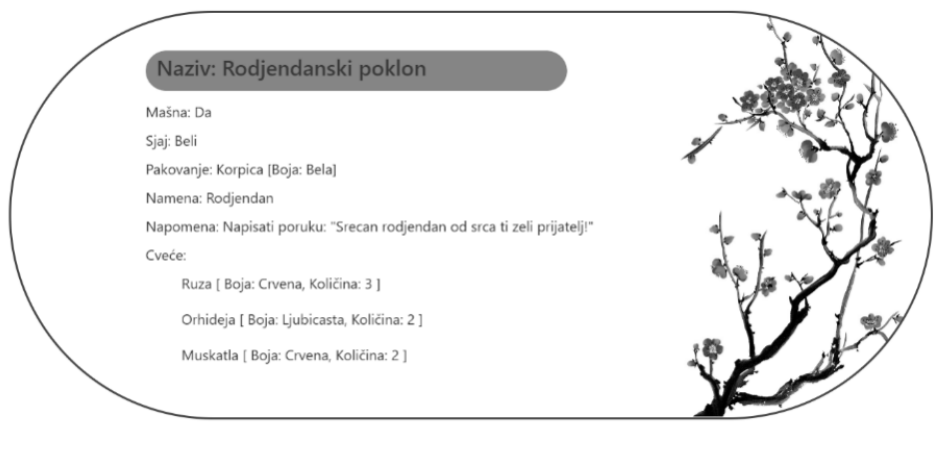
© 2022 - LycorisRadiata - Privacy

Слика 64 Попуњена форма за креирање цветног аранжмана

2. Купац **контролише** да ли је коректно унео податке о цветном аранжману. (АНСО)
3. Купац **позива** систем да креира нови цветни аранжман са задатим подацима. (АПСО)

Опис акције: Купац кликом на дугме „Креирај“ позива системску операцију КреирајЦветниАранжман(ЦветниАранжман)

4. Систем **креира** цветни аранжман са задатим подацима. (СО)
5. Систем **приказује** купцу забележени цветни аранжман. (ИА)



Слика 65 Приказ креираног цветног аранжмана

Алтернативна сценарија

6.1 Уколико систем не може да креира цветни аранжман он приказује купцу поруку „Десила се грешка” (ИА)

Desila se greška.



Слика 66 Испис грешке

СК7 : Случај коришћења – Креирање корисничког налога

Назив СК

Креирање корисничког налога

Актери СК

Посетилац

Учесници СК

Посетилац и систем

Предуслов: Систем је укључен. Систем приказује форму за креирање корисничког налога

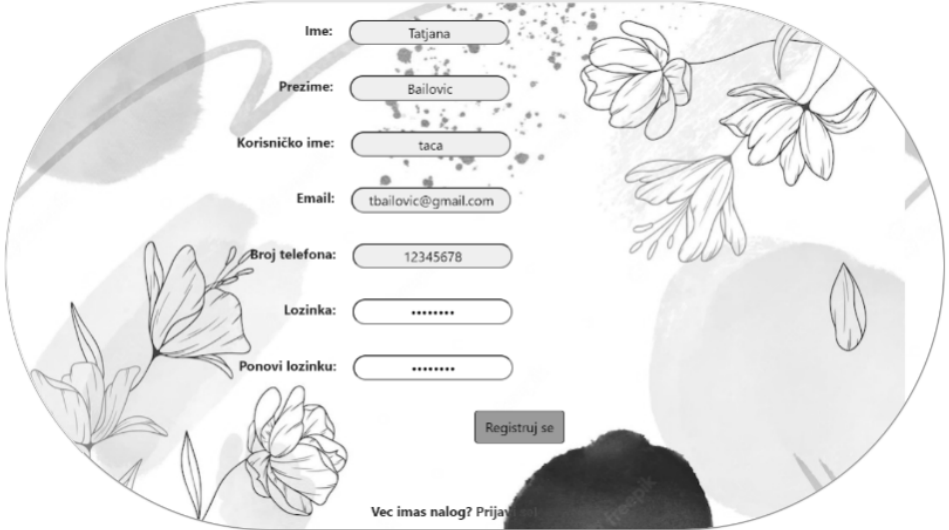
The image shows a registration form within an oval frame decorated with floral illustrations. The form contains the following fields and elements:

- Ime:
- Prezime:
- Korisničko ime:
- Email:
- Broj telefona:
- Lozinka:
- Ponovi lozinku:
- Registruj se button
- Vec imas налог? Prijavi se link

Слика 67 Форма за регистрацију

Основни сценарио СК

1. Посетилац **уноси** податке о новом купцу. (АПУСО)



Ime:

Prezime:

Korisničko ime:

Email:

Broj telefona:

Lozinka:

Ponovi lozinku:

Vec imas nalog? Prijavi se

Слика 68 Попуњена форма за регистрацију

2. Посетилац **контролише** да ли је коректно унео податке о купцу. (АНСО)
3. Посетилац **позива** систем да креира новог купца са задатим подацима. (АПСО)

Опис акције: Посетилац кликом на дугме „Registruj se“ позива системску операцију КреирајКупца(Купац)

4. Систем **креира** новог купца. (СО)
5. Систем **приказује** посетиоцу поруку страницу за пријаву. (ИА)



Korisničko ime:

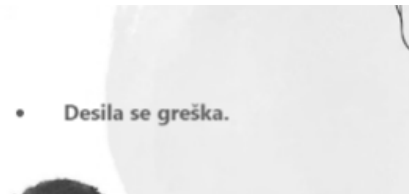
Lozinka:

Nemas nalog? Kreiraj ga!

Слика 69 Празна форма за пријаву

Алтернативна сценарија

5.1 Уколико систем не може да креира купца он приказује посетиоцу поруку „Десила се грешка” (ИА)



Слика 70 Порука грешке приликом пријаве

СК8 : Случај коришћења – Брисање наруџбине

Назив СК

Брисање наруџбине

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са наруџбином



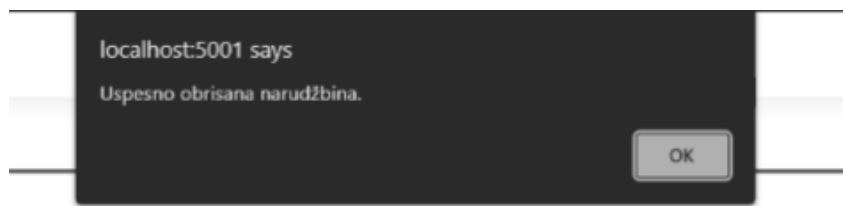
Слика 71 Форма за рад са наруџбином

Основни сценарио СК

1. Администратор **бира** наруџбину коју жели да обрише. (АПУСО)
2. Администратор **позива** систем да обрише податке о наруџбини. (АПСО)

Опис акције: Администратор кликом на дугме „Izbriši“ позива системску операцију ОбришиНаруџбину(Наруџбина)

3. Систем **брише** податке о наруџбини. (СО)
4. Систем **приказује** администратору поруку: „Успешно обрисана наруџбина.“ (ИА)



Слика 72 Порука о успешно обрисаној наруџбини

Алтернативна сценарија

10.1 Уколико систем не може да обрише податке о наруџбини он приказује администратору поруку „Десила се грешка” (ИА)



Слика 73 Порука о неуспешно обрисаној наруџбини

СК9 : Случај коришћења – Измена корисничког налога

Назив СК

Измена корисничког налога

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитани су подаци о купцу. Систем приказује форму за рад са профилем купца.

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se

Ime:

Prezime:

Korisnicko ime:

Email:

Broj telefona:

Stara lozinka:

Nova lozinka:

Слика 74 Форма за измену података купца

Основни сценарио СК

- 1 Систем **тражи** податке о купцу. (СО)
- 2 Систем **приказује** купцу пронађене податке. (ИА)
- 3 Купац **мења** податке. (АПУСО)

Ime: TatjanaZMENA

Prezime: BailovicZMENA

Korisnicko ime: tacalZMENA

Email: bailovict@gmail.com

Broj telefona: 123456789

Stara lozinka:

Nova lozinka:

Sacuvaj promene

Слика 75 Измењени подаци о купцу

- 4 Купац **контролише** да ли је коректно унео податке о купцу. (АНСО)
- 5 Купац **позива** систем да запамти податке. (АПСО)

Опис акције: Купац кликом на дугме „Сачувај промене“ позива системску операцију ИзмениКупца(Купац)

- 6 Систем **памти** податке о купцу. (СО)
- 7 Систем **приказује** купцу измењене податке. (ИА)

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se

Naruci

Ime: TatjanaZMENA

Prezime: BailovicZMENA

Korisnicko ime: tacalZMENA

Email: bailovict@gmail.com

Broj telefona: 12345678

Izmeni

Слика 76 Профил купца

Алтернативна сценарија

7.1 Уколико систем не може да запамти податке о купцу он приказује купцу поруку „Десила се грешка” (ИА)

Desila se greška.



Слика 77 Грешка приликом измене купца

СК10 : Случај коришћења – Креирање цвета

Назив СК

Креирање цвета

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са цветом.



Слика 78 Форма за креирање цвета

Основни сценарио СК

1. Администратор уноси податке о цвету. (АПУСО)

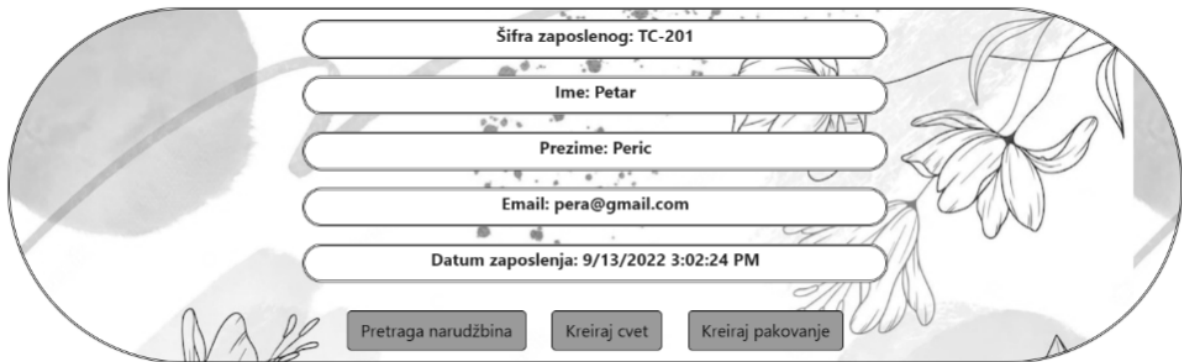


Слика 79 Попуњена форма за креирање цвета

2. Администратор **контролише** да ли је коректно унео податке о цвету. (АНСО)
3. Администратор **позива** систем да креира нови цвет са задатим подацима. (АПСО)

Опис акције: Администратор кликом на дугме „Креирај“ позива системску операцију КреирајЦвет(Цвет)

4. Систем **креира** цвет са задатим подацима. (СО)
5. Систем **приказује** администратору почетну страну. (ИА)



Слика 80 Профил администратора

Алтернативна сценарија

- 6.1 Уколико систем не може да креира цвет он приказује администратору поруку „Десила се грешка” (ИА)

Desila se greška.



Слика 81 Испис грешке приликом креирања цвета

СК11 : Случај коришћења – Креирање паковања

Назив СК

Креирање паковања

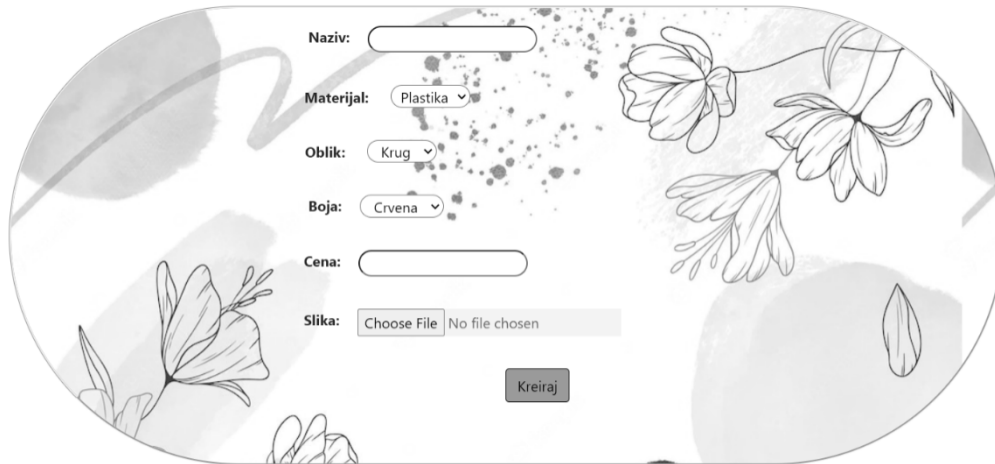
Актери СК

Администратор

Учесници СК

Администратор и систем

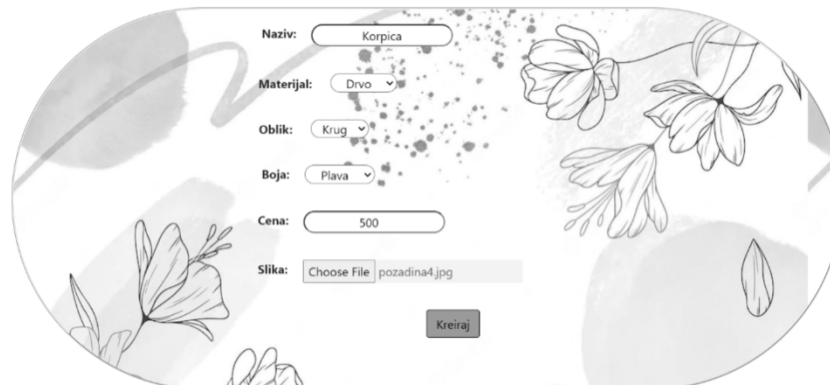
Предуслов: Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са паковањем.



Слика 82 Форма за креирање паковања

Основни сценарио СК

1. Администратор уноси податке о паковању. (АПУСО)



Слика 83 Попуњена форма за креирање паковања

2. Администратор **контролише** да ли је коректно унео податке о паковању. (АНСО)
3. Администратор **позива** систем да креира ново паковање са задатим подацима. (АПСО)

Опис акције: Администратор кликом на дугме „Креирај“ позива системску операцију КреирајПаковање(Паковање)

4. Систем **креира** паковање са задатим подацима. (СО)
5. Систем **приказује** администратору почетну страну. (ИА)



Слика 84 Профил администратора

Алтернативна сценарија

5.1 Уколико систем не може да креира паковање он приказује администратору поруку „Десила се грешка” (ИА)

Desila se greška.



Слика 85 Порука грешке приликом креирања паковања

СК12 : Случај коришћења – Одјава са система

Назив СК

Одјава са система

Актери СК

Администратор/купац

Учесници СК

Администратор/купац и систем

Предуслов: Систем је укључен и администратор/купац је улогован под својом шифром.

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se

Ime: Tatjana

Prezime: Bailovic

Korisničko ime: taca

Email: bailovict@gmail.com

Broj telefona: 0648578649

Слика 86 Улогован купац

Основни сценарио СК

1. Администратор/купац позива систем да га одјави. (АПСО)

Опис акције: Администратор/купац кликом на дугме „Odjavi se“ позива системску операцију Одјави(Особа)

2. Систем одјављује администратора/купца са система и приказује почетну страну. (ИА)



Слика 87 Страна за пријаву

Алтернативна сценарија

2.1 Уколико систем не може да одјави администратора/купца са система: „Десила се грешка”. (ИА)

Desila se greška.



Слика 88 Испис грешке

СК13 : *Случај коришћења – Креирање наруџбине*

Назив СК

Креирање наруџбине

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитна је листа производа и продајних места. Систем приказује форму за рад са наруџбином.

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se **Naruci**

Tip narudzbine: Lično preuzimanje

Adresa preuzimanja: Mesto1 **Prodajna mesta**

Odaberite proizvode koje želite da naručite: :

Naziv	Vrsta	Cena		
Orhideja	Cvet	150.00	Detalji	Dodaj
Muskatla	Cvet	120.00	Detalji	Dodaj
Rodjendanski poklon	Cvetni aranžman	1640.00	Detalji	Dodaj

Odabrani proizvodi:

Naziv	Kolicina
-------	----------

Naruci

Слика 89 Форма за креирање наруџбине

Основни сценарио СК

- 1 Купац **уноси** податке за нову наруџбину. (АПУСО)

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se Naruci

Tip narudzbine: Lično preuzimanje

Adresa preuzimanja: Mesto1 Prodajna mesta

Odaberite proizvode koje želite da naručite :

Naziv	Vrsta	Cena	
Orhideja	Cvet	150.00	Detalji
Muskatla	Cvet	120.00	Detalji Dodaj
Rodjendanski poklon	Cvetni aranžman	1640.00	Detalji

Odabrani proizvodi:

Naziv	Kolicina	
Orhideja	<input type="text" value="3"/>	Ukloni
Rodjendanski poklon	<input type="text" value="1"/>	Ukloni

Naruci

Слика 90 Попуњена форма за креирање наруџбине

- 2 Купац **контролише** да ли је коректно унео податке о наруџбини. (АНСО)
- 3 Купац **позива** систем да креира нову наруџбину са задатим подацима. (АПСО)

Опис акције: Купац кликом на дугме „Наруџи“ позива системску операцију КреирајНаруџбину(Наруџбина)

- 4 Систем **креира** наруџбину са задатим подацима. (СО)
- 5 Систем **приказује** купцу забележену наруџбину. (ИА)

Tip narudžbine: Lično preuzimanje

Adresa preuzimanja: Mesto1

Status narudžbine: Obrada

Stavke:

- Orhideja [Količina: 3]
- Rodjendanski poklon [Količina: 1]

Слика 91 Креирана наруџбина

Алтернативна сценарија

- 3.2 Уколико систем не може да креира наруџбину он приказује купцу поруку „Десила се грешка” (ИА)

Desila se greška.

Ok

Слика 92 Грешка пиликом креирања наруџбине

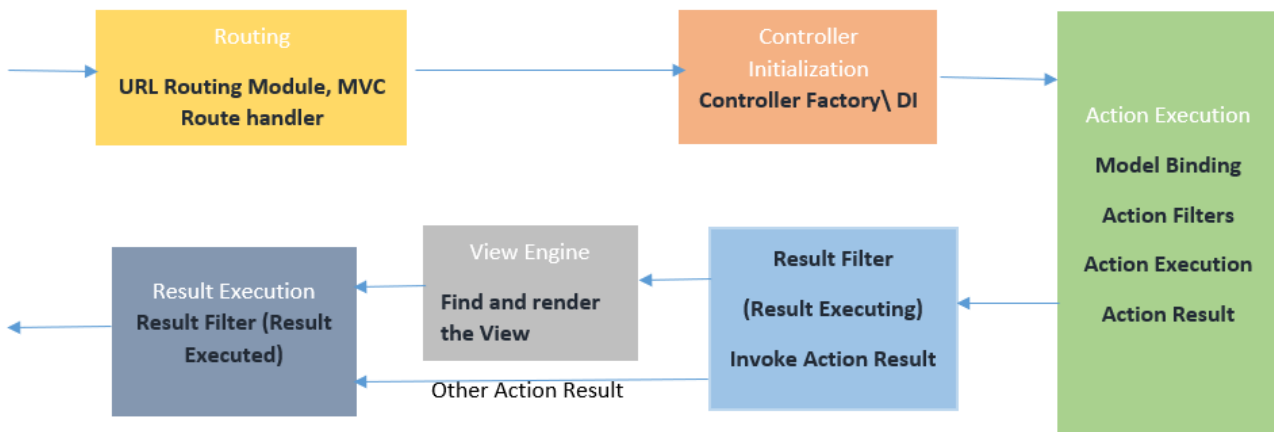
Пројектовање апликационе логике

Пројектовање апликационе логике садржи:

- Пројектовање контролера апликационе логике
- Пројектовање пословне логике
- Пројектовање слоја приступа подацима

Пројектовање контролера апликационе логике

Захтеви доспели са погледа шаљу се контролеру који их прихвата, а затим позива одговарајуће операције слоја приступа подацима. Добијене податке из резултата контролер попуњава у одговарајући модел и прослеђује заједно са погледом.



Слика 93 Животни циклус једног захтева [10]

Пројектовање пословне логике

Пословна логика је описана доменским класама и системским операцијама. За сваки од уговора системских операција пројектује се концептуално решење [1].

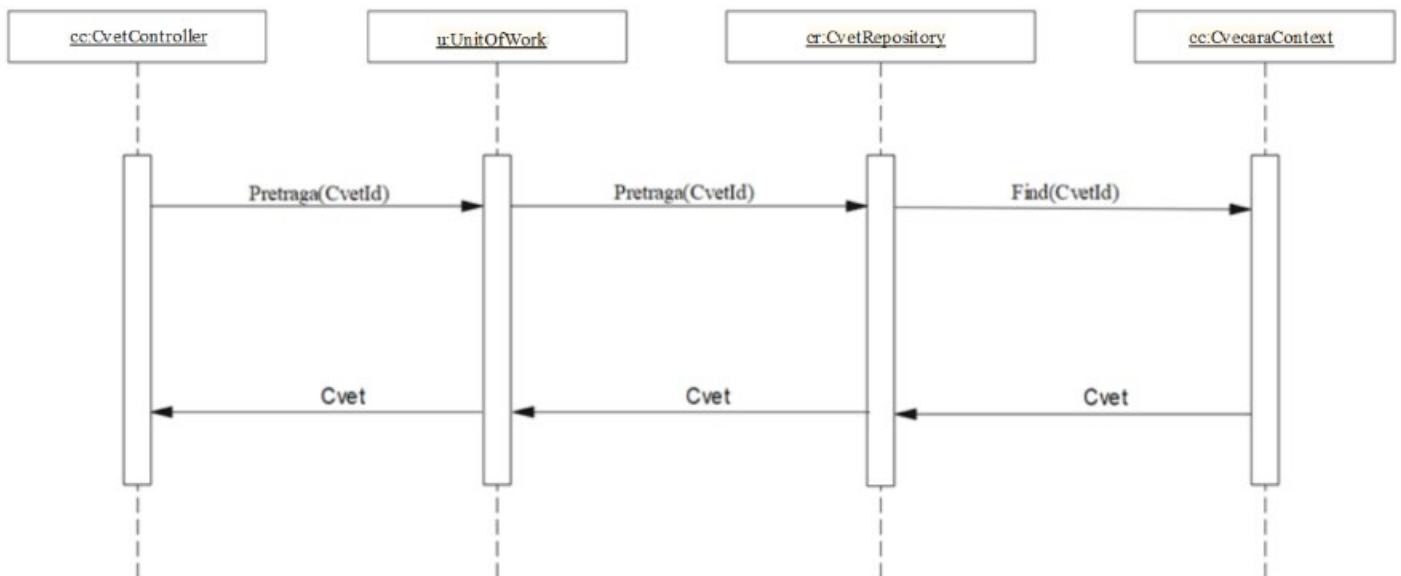
Уговор УГ1: ПрикажиЦвет

Операција: ПрикажиЦвет(Цвет)

Веа са СК: СК1

Предуслов: /

Постуслов: /



Слика 94 УГ1 пројектовање

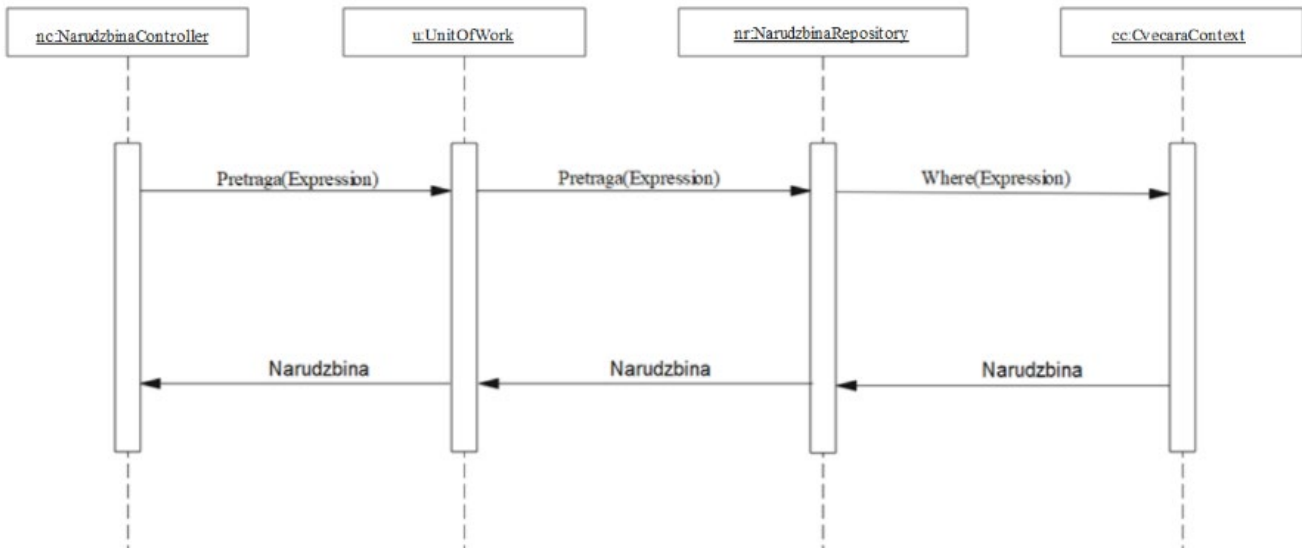
Уговор УГ2: ПрикажиНаруџбине

Операција: ПрикажиНаруџбине(КорисничкоИме)

Веа са СК: СК3

Предуслов: /

Постуслов: /



Слика 95 УГ2 пројектовање

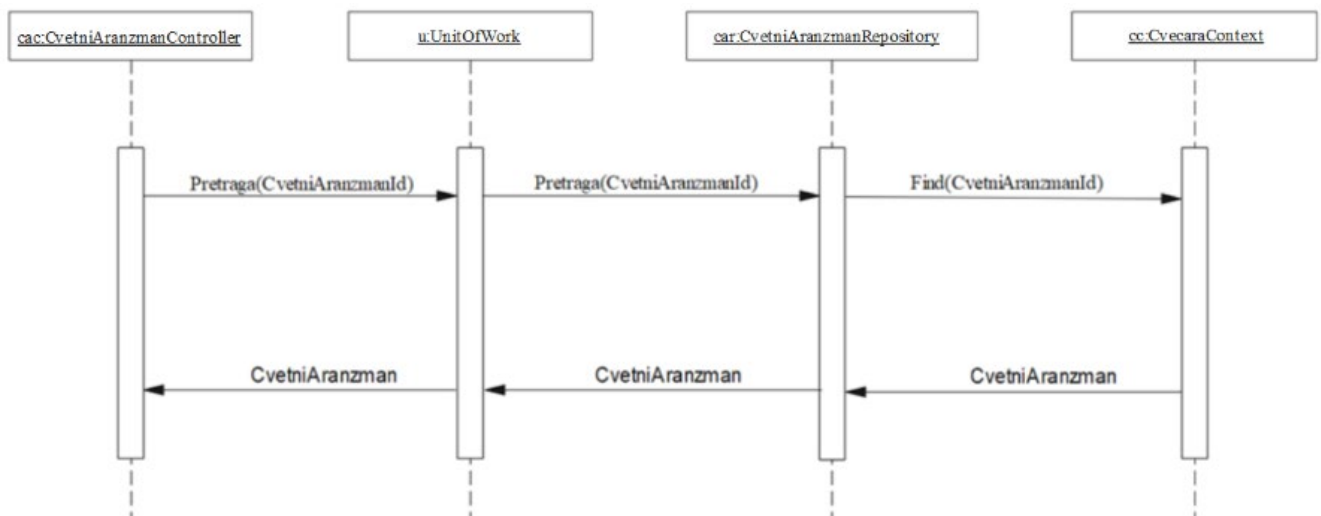
Уговор УГ3: ПрикажиЦветниАранжман

Операција: ПрикажиЦветниАранжман(ЦветниАранжман)

Веа са СК: СК2

Предуслов: /

Постуслов: /



Слика 96 УГ3 пројектовање

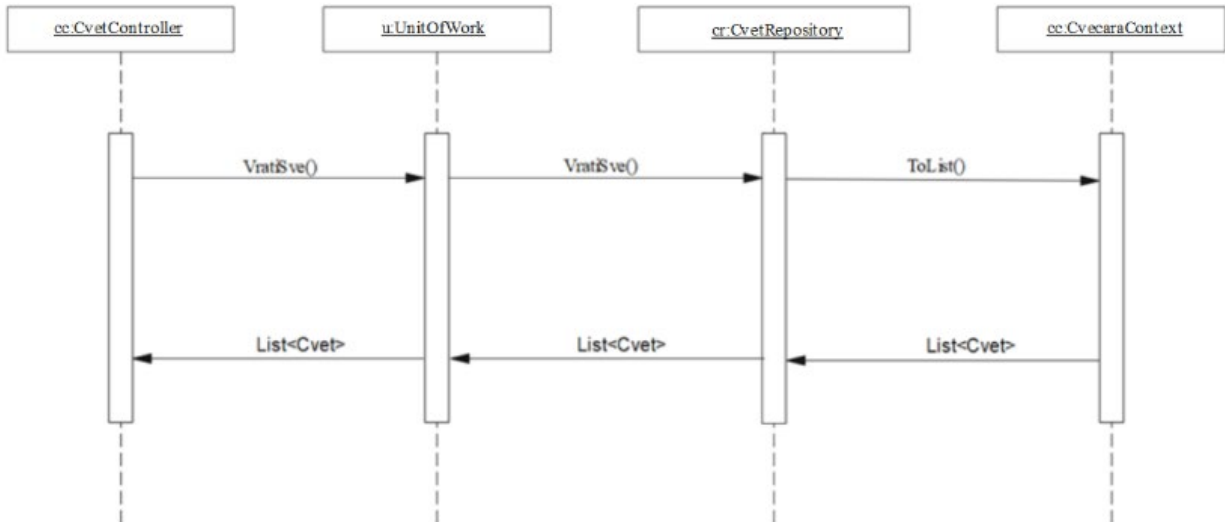
Уговор УГ4: ВратиЦвеће

Операција: ВратиЦвеће(List<Цвет>)

Вега са СК: СК1, СК6

Предуслов: /

Постуслов: /



Слика 97 УГ4 пројектовање

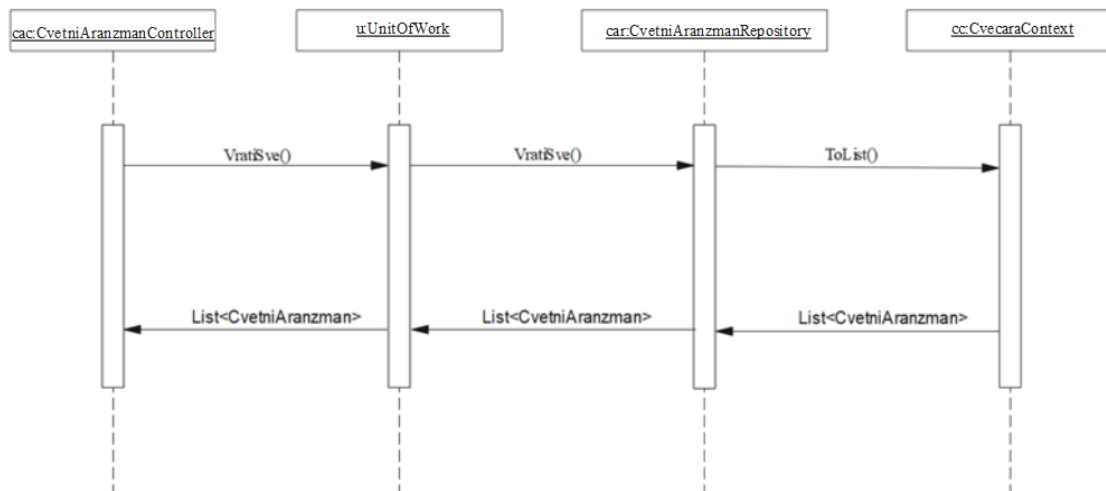
Уговор УГ5: ВратиЦветнеАранжмане

Операција: ВратиЦветнеАранжмане(List<ЦветниАранжман>)

Вега са СК: СК2

Предуслов: /

Постуслов: /



Слика 98 УГ5 пројектовање

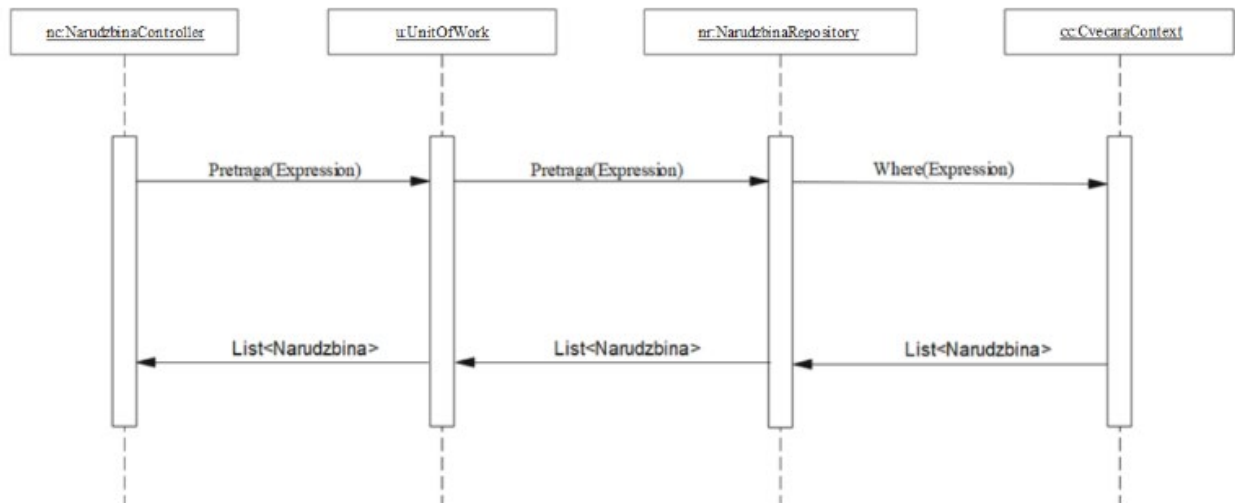
Уговор УГ6: ВратиНаруџбине

Операција: ВратиНаруџбине(КорисничкоИме)

Вега са СК: СК8, СК5

Предуслов: /

Постуслов: /



Слика 99 УГ6 пројектовање

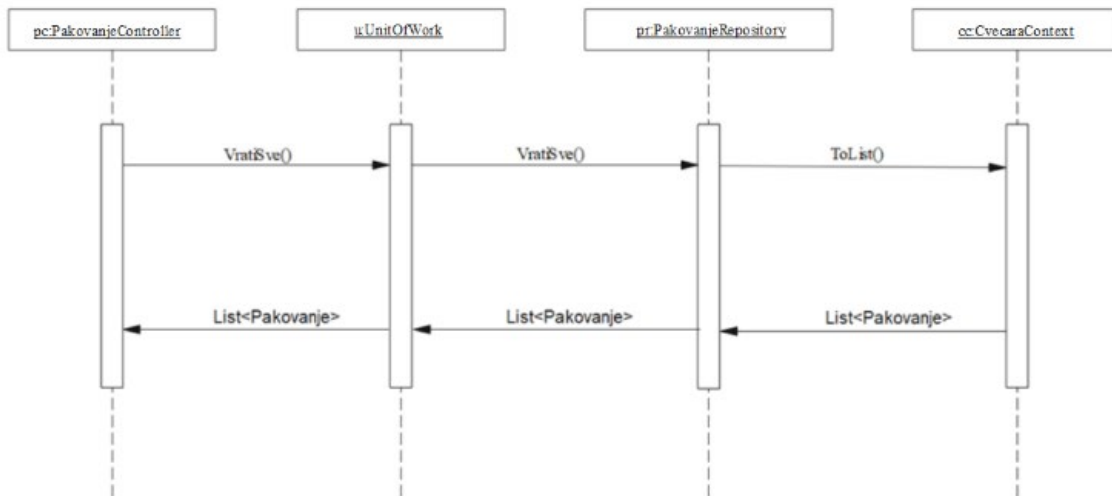
Уговор УГ7: ВратиПаковања

Операција: ВратиПаковања(List<Паковања>)

Вега са СК: СК6

Предуслов: /

Постуслов: /



Слика 100 УГ7 пројектовање

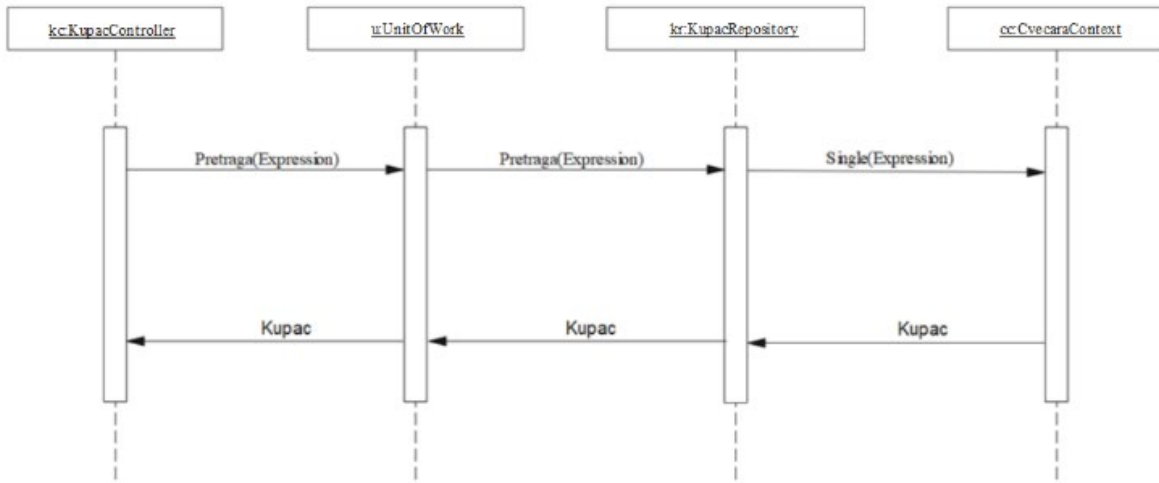
Уговор УГ8: ВратиКупца

Операција: ВратиКупца(КорисничкоИме)

Веа са СК: СК9

Предуслов: /

Постуслов: /



Слика 101 УГ8 пројектовање

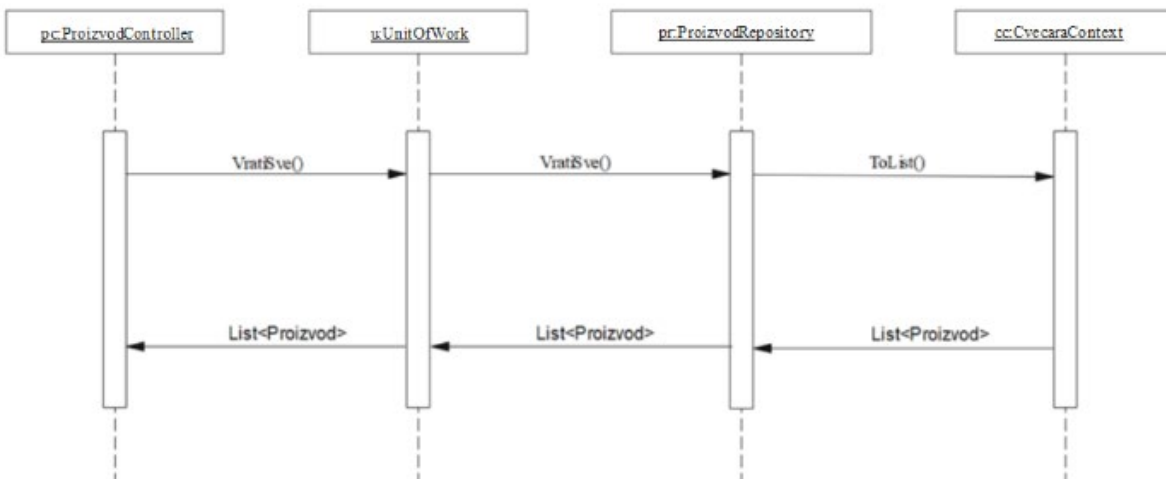
Уговор УГ9: ВратиПроизводе

Операција: ВратиПроизводе(List<Производ>)

Веа са СК: СК13

Предуслов: /

Постуслов: /



Слика 102 УГ9 пројектовање

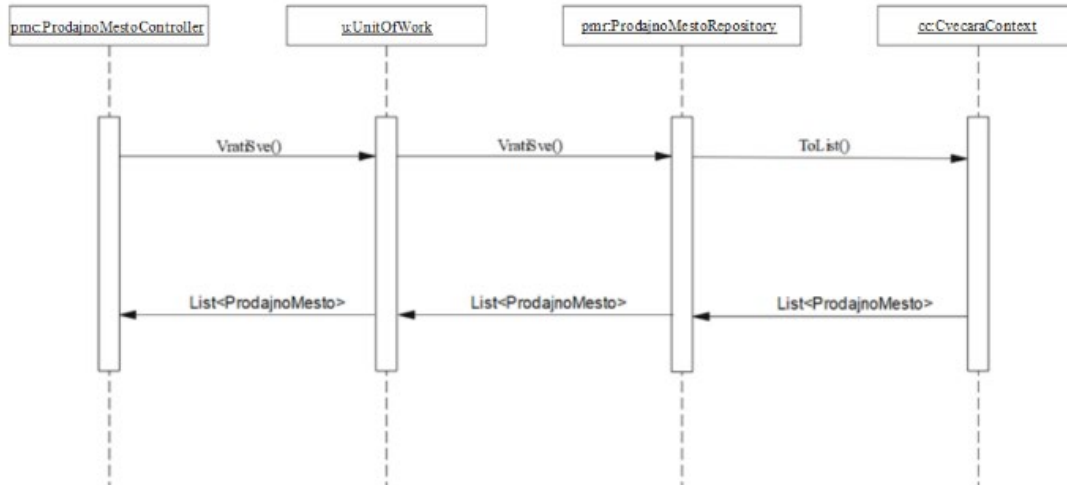
Уговор УГ10: ВратиПродајнаМеста

Операција: ВратиПродајнаМеста(List<ПродајноМесто>)

Веа са СК: СК13

Предуслов: /

Постуслов: /



Слика 103 УГ10 пројектовање

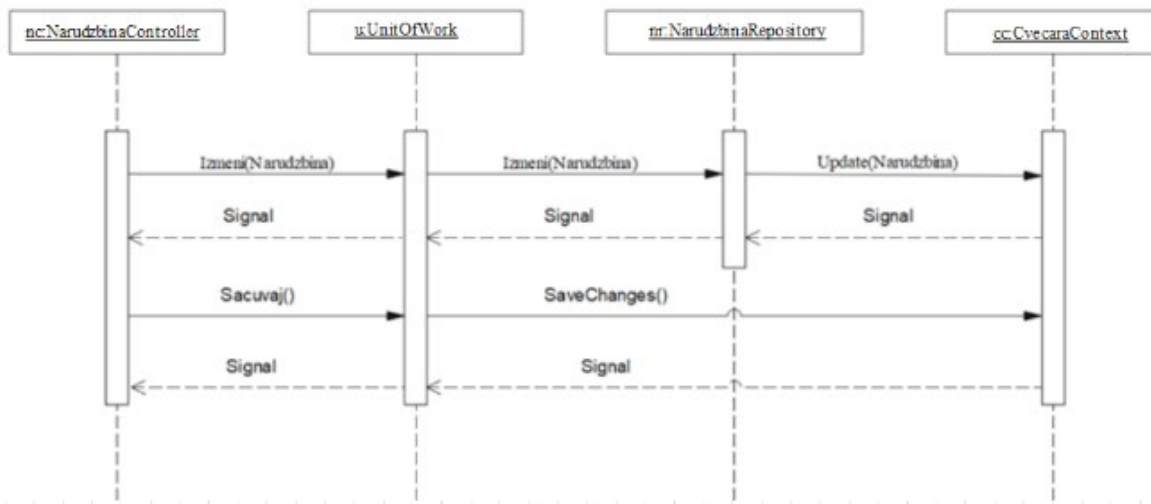
Уговор УГ11: ИтзмениНаруцбину

Операција: ИзмениНаруцбину(Наруцбина)

Веа са СК: СК5

Предуслов: Вредносна и структурна ограничења над објектом Наруцбина морају бити задовољена

Постуслов: Подаци о наруцбини су ажурирани



Слика 104 УГ11 пројектовање

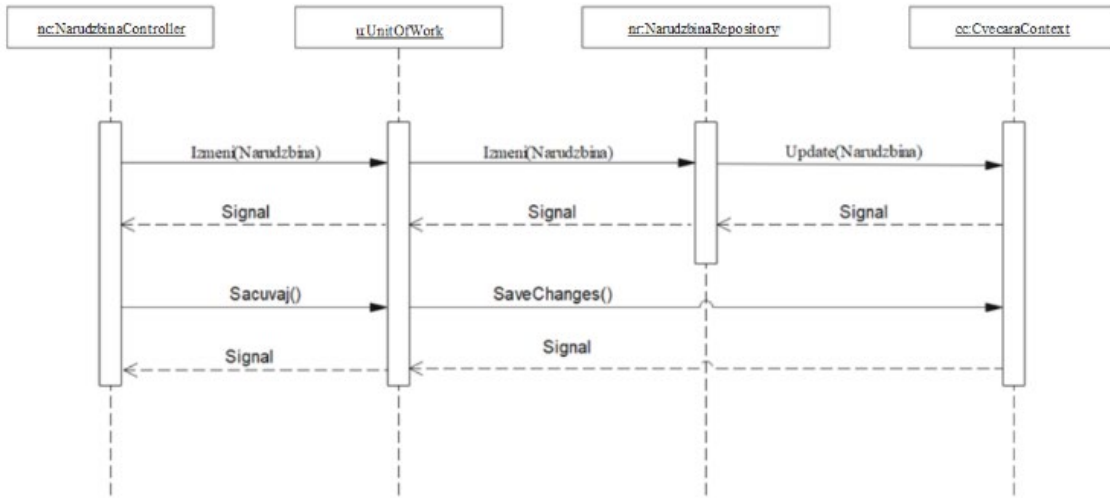
Уговор УГ12: ИзмениКупца

Операција: ИзмениКупца(Купац)

Веа са СК: СК9

Предуслов: Вредносна и структурна ограничења над објектом Купац морају бити задовољена

Постуслов: Подаци о купцу су ажурирани



Слика 105 УГ12 пројектовање

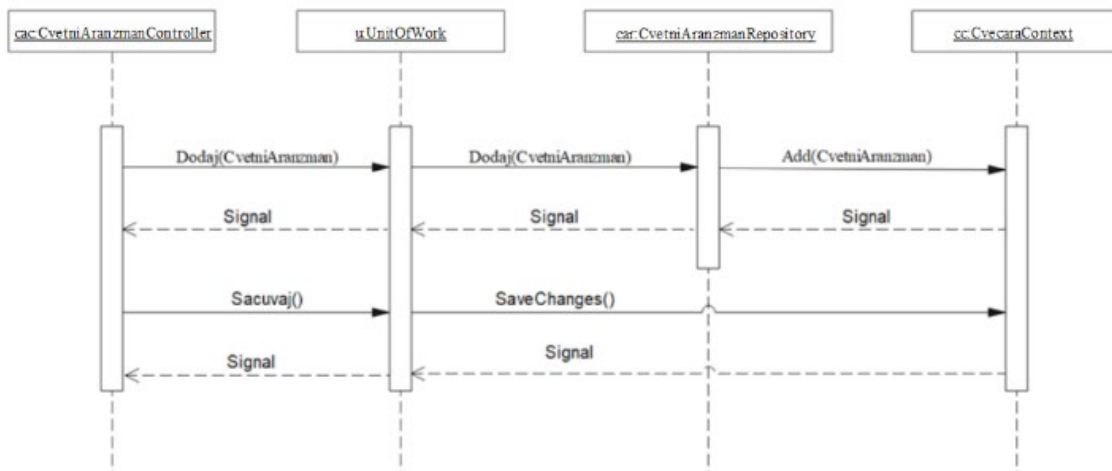
Уговор УГ13: КреирајЦветниАранжман

Операција: КреирајЦветниАранжман(ЦветниАранжман)

Веа са СК: СК6

Предуслов: Вредносна и структурна ограничења над објектом ЦветниАранжман морају бити задовољена

Постуслов: Подаци о цветном аранжману су креирани



Слика 106 УГ13 пројектовање

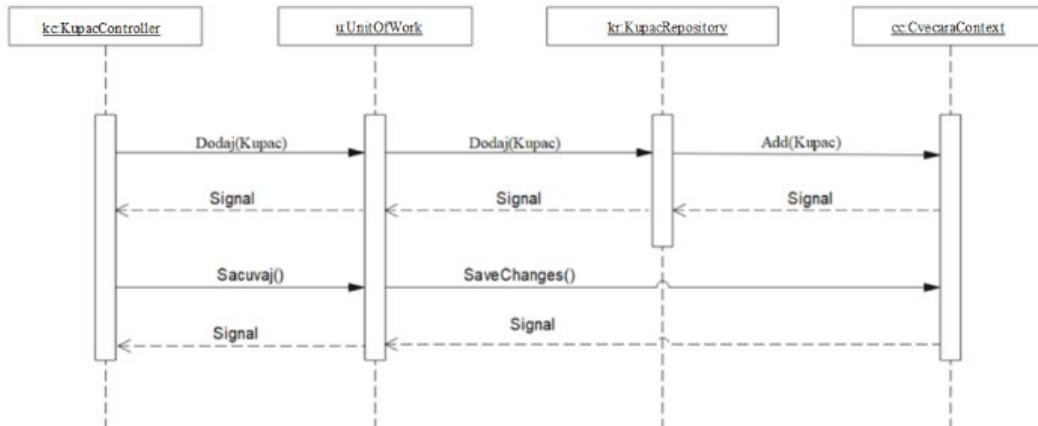
Уговор УГ14: КреирајКупца

Операција: КреирајКупца(Купац)

Веа са СК: СК7

Предуслов: Вредносна и структурна ограничења над објектом Купац морају бити задовољена

Постуслов: Подаци о купцу су креирани



Слика 107 УГ14 пројектовање

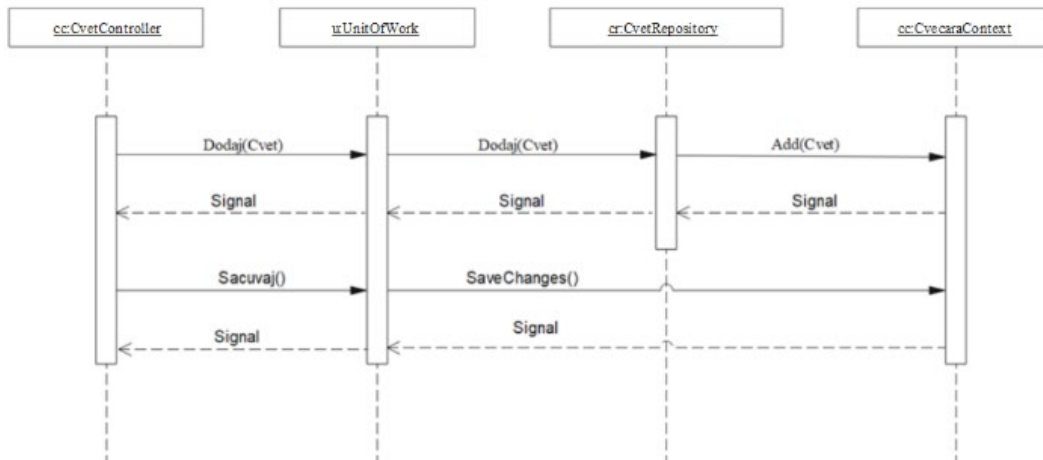
Уговор УГ15: КреирајЦвет

Операција: КреирајЦвет(Цвет)

Веа са СК: СК10

Предуслов: Вредносна и структурна ограничења над објектом Цвет морају бити задовољена

Постуслов: Подаци о цвету су креирани



Слика 108 УГ15 пројектовање

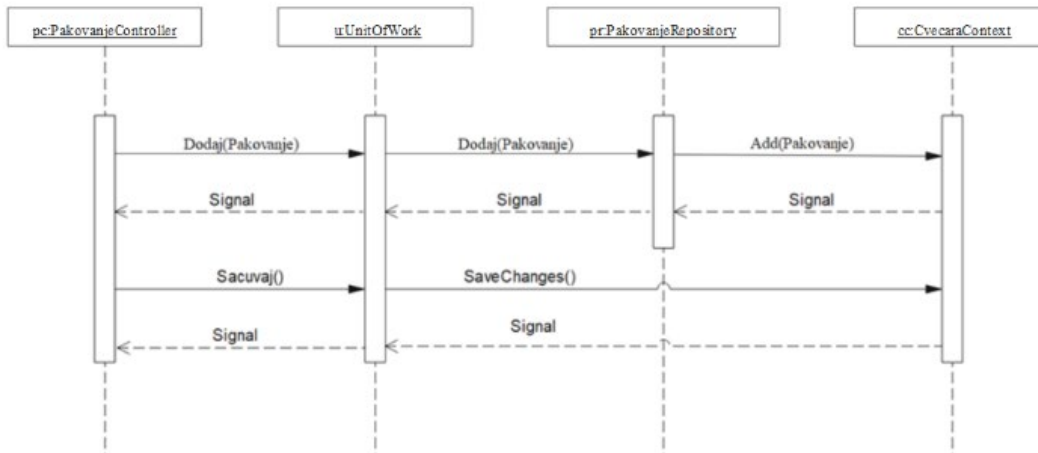
Уговор УГ16: КреирајПаковање

Операција: КреирајПаковање(Паковање)

Веза са СК: СК11

Предуслов: Вредносна и структурна ограничења над објектом Паковање морају бити задовољена

Постуслов: Подаци о паковању су креирани



Слика 109 УГ16 пројектовање

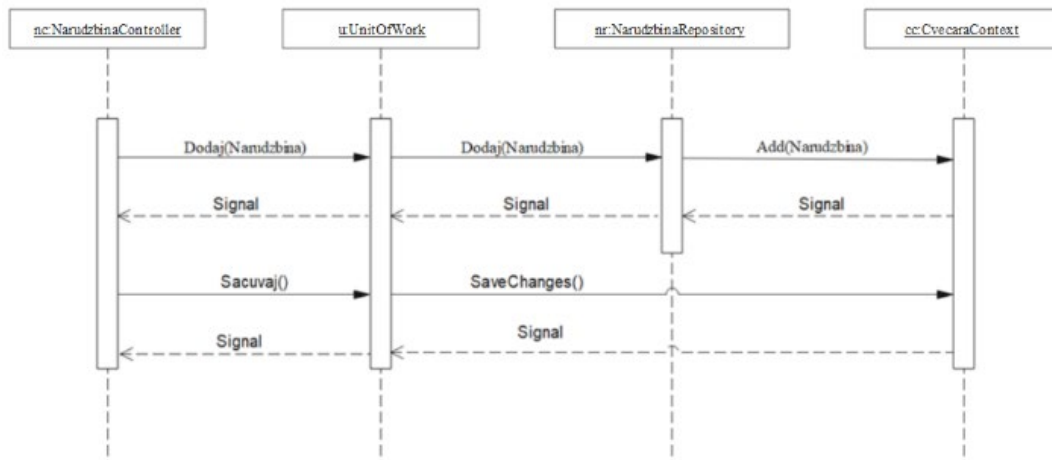
Уговор УГ17: КреирајНаруџбину

Операција: КреирајНаруџбину(Наруџбина)

Веза са СК: СК13

Предуслов: Вредносна и структурна ограничења над објектом Наруџбина морају бити задовољена

Постуслов: Подаци о наруџбини су креирани



Слика 110 УГ17 пројектовање

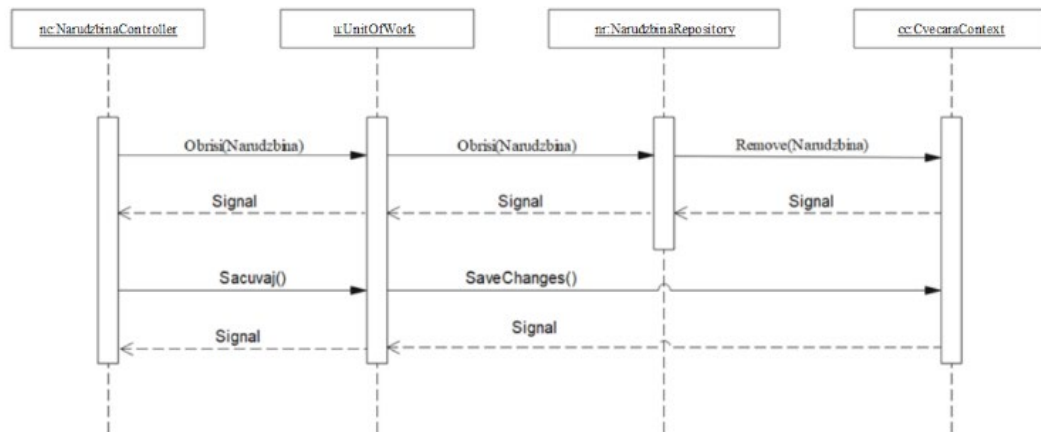
Уговор УГ18: ОбришиНаруџбину

Операција: ОбришиНаруџбину(Наруџбина)

Веа са СК: СК8

Предуслов: Структурна ограничења над објектом Наруџбина морају бити задовољена

Постуслов: Подаци о наруџбини су обрисани



Слика 111 УГ18 пројектовање

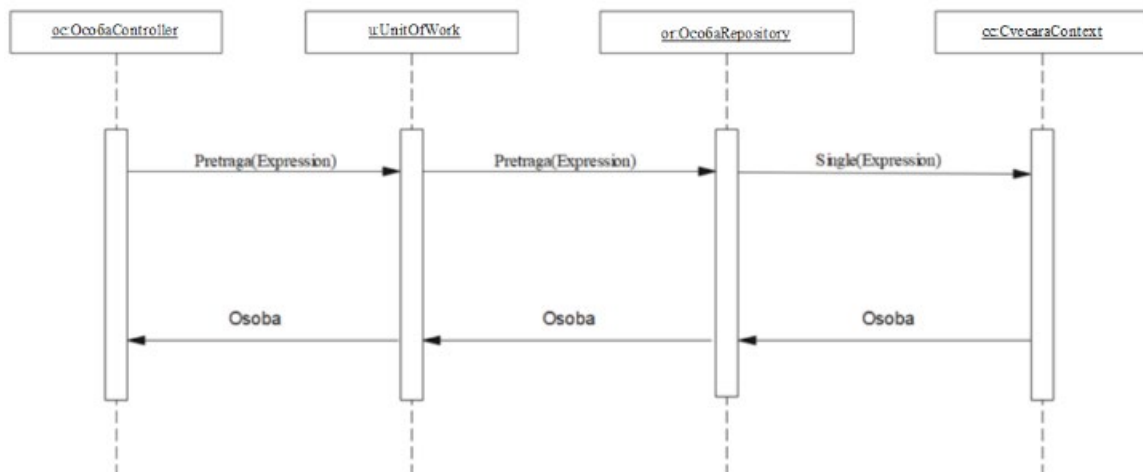
Уговор УГ19: Пријава

Операција: Пријава(КорисничкоИме,Лозинка)

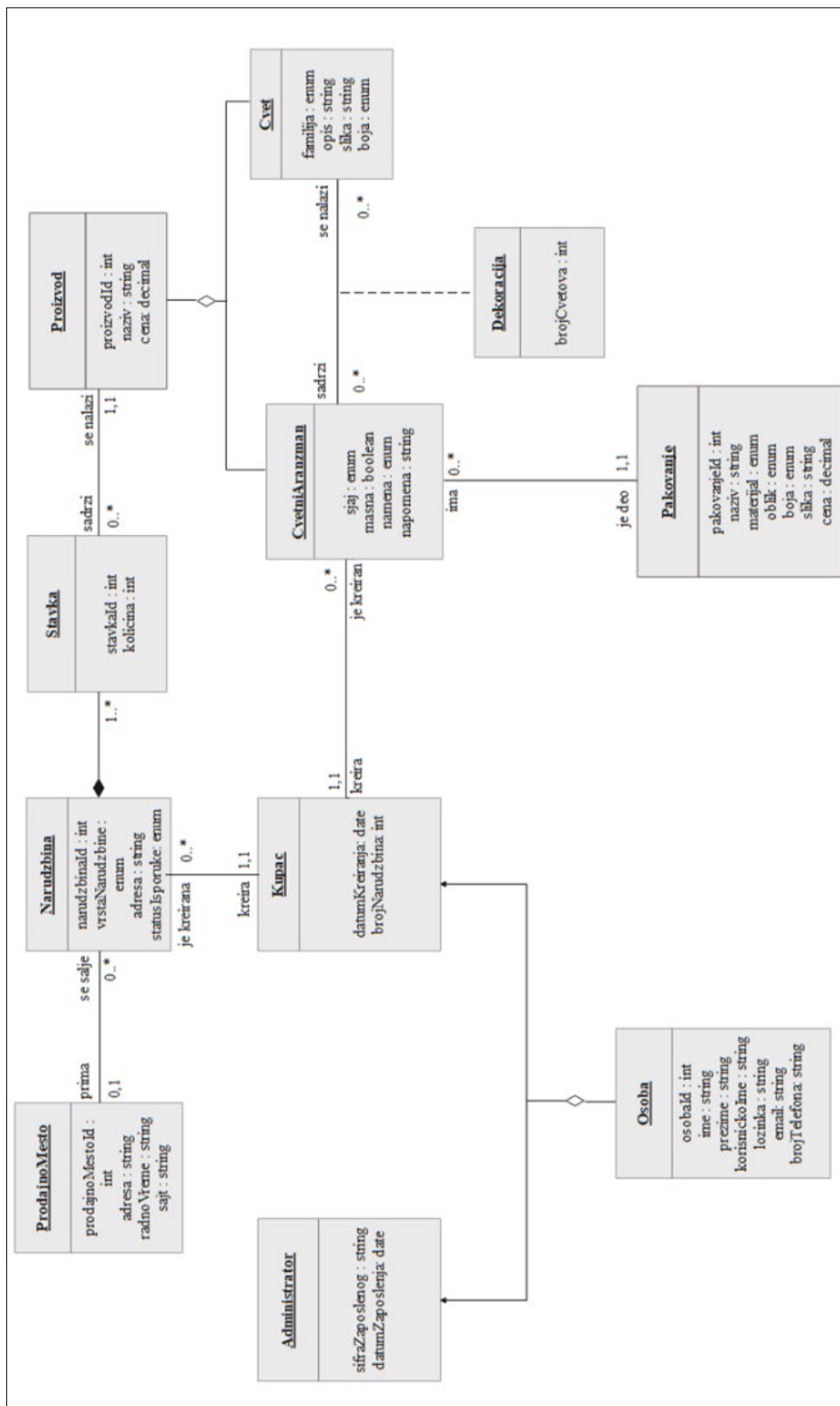
Веа са СК: СК4

Предуслов: Морају бити задовољена структурна и вредносна ограничења над објектом Купац/Администратор

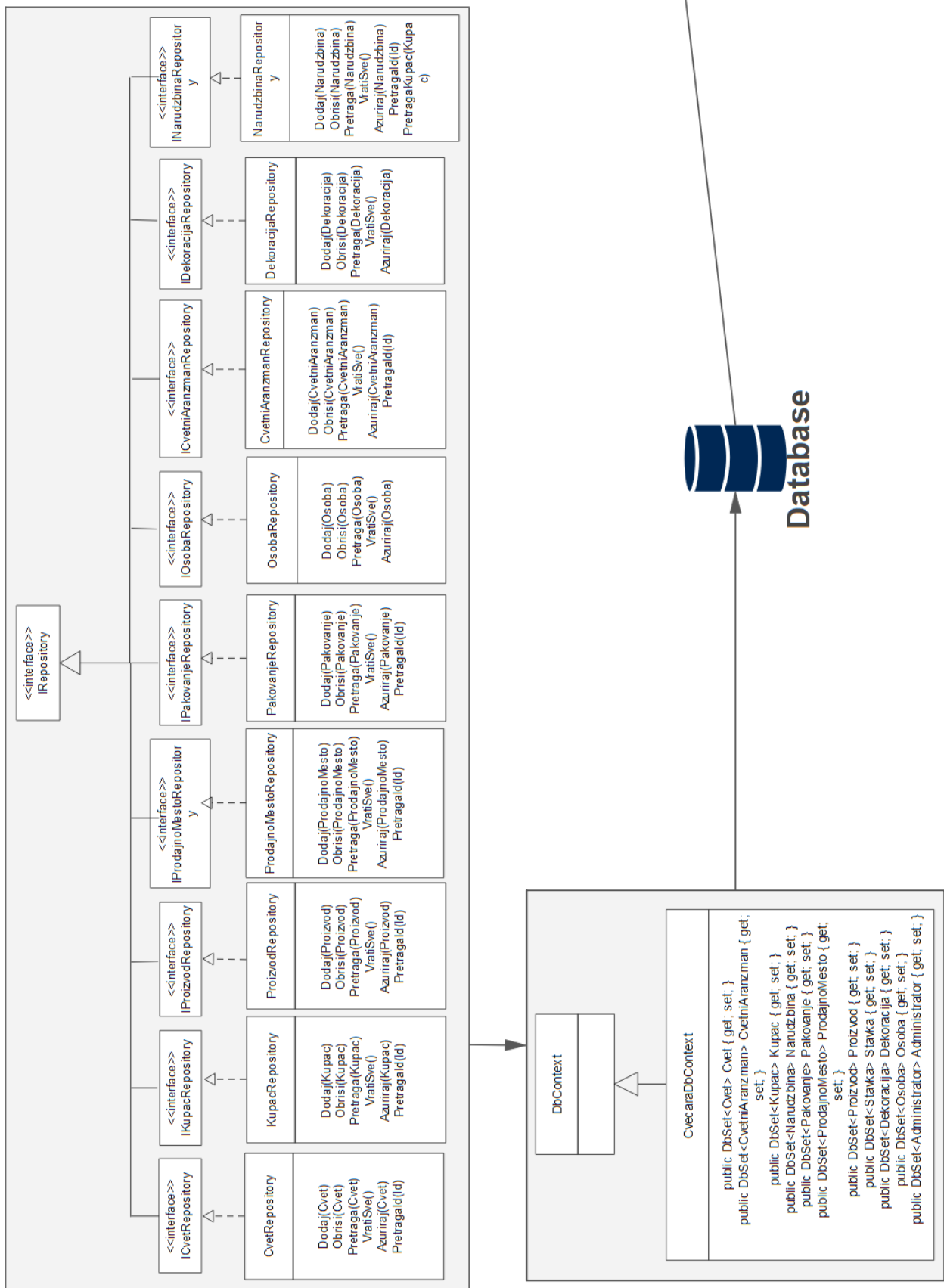
Постуслов: Купац/Администратор је пријављен на систем.



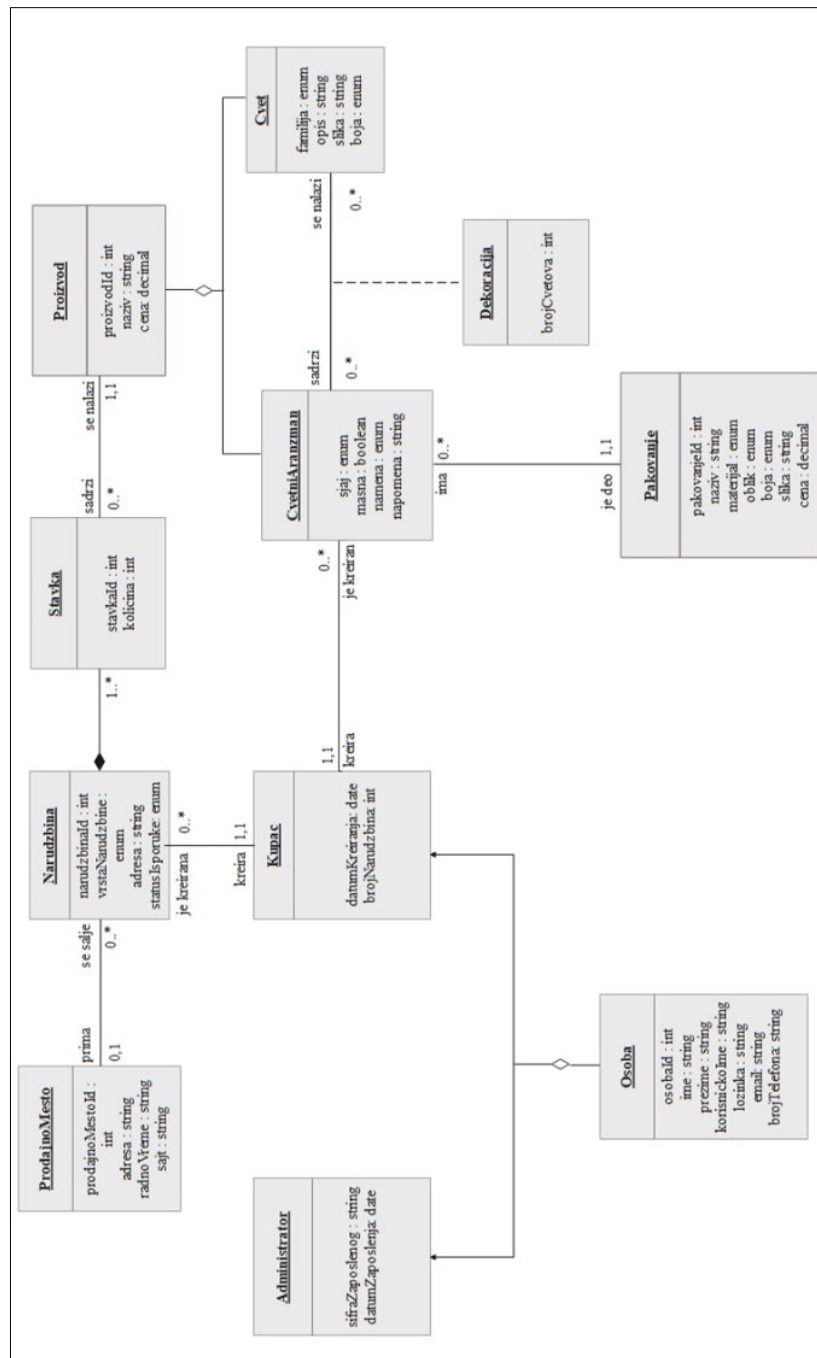
Слика 112 УГ19 пројектовање



Слика 113 Пројектовање слоја приступа подацима (Први део)



Слика 114 Пројектовање слоја приступа подацима (Други део)



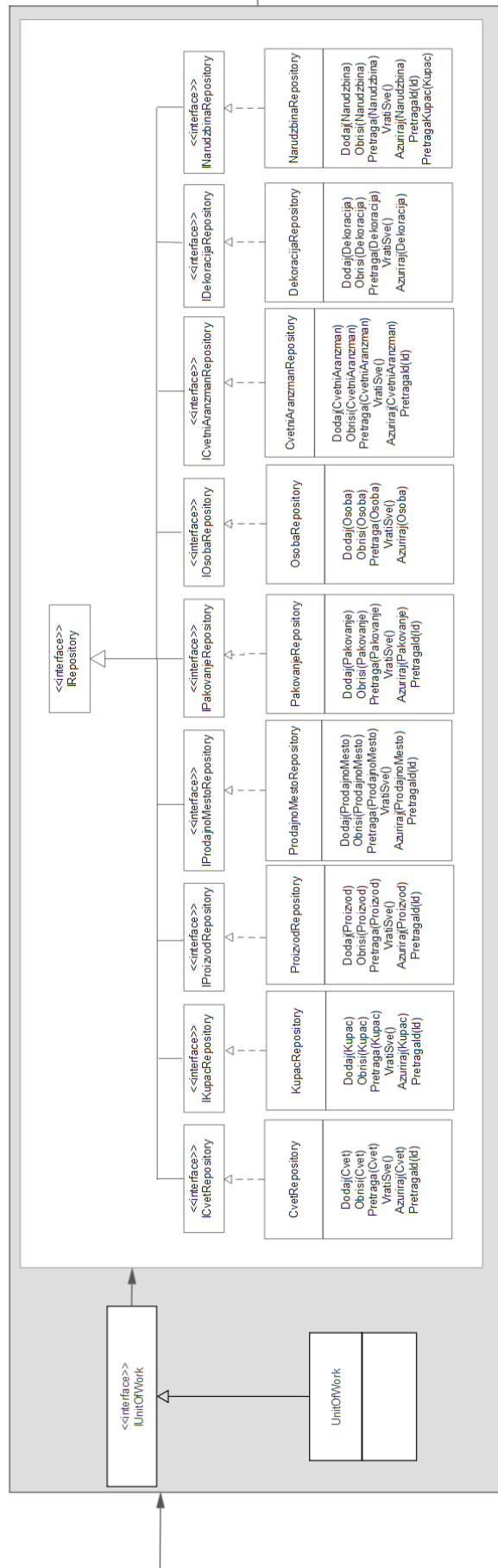
```

classDiagram
    class DbContext
    class CvetararDbContext {
        public DbSet<Cvet> Cvet { get; set; }
        public DbSet<CvetniAranzman> CvetniAranzman { get; set; }
        public DbSet<Kupac> Kupac { get; set; }
        public DbSet<Narudzbinas> Narudzbinas { get; set; }
        public DbSet<Pakovanje> Pakovanje { get; set; }
        public DbSet<ProdajnoMesto> ProdajnoMesto { get; set; }
        public DbSet<Proizvod> Proizvod { get; set; }
        public DbSet<Stavka> Stavka { get; set; }
        public DbSet<Dekoracija> Dekoracija { get; set; }
        public DbSet<Osoba> Osoba { get; set; }
        public DbSet<Administrator> Administrator { get; set; }
    }
    DbContext <|-- CvetararDbContext
  
```

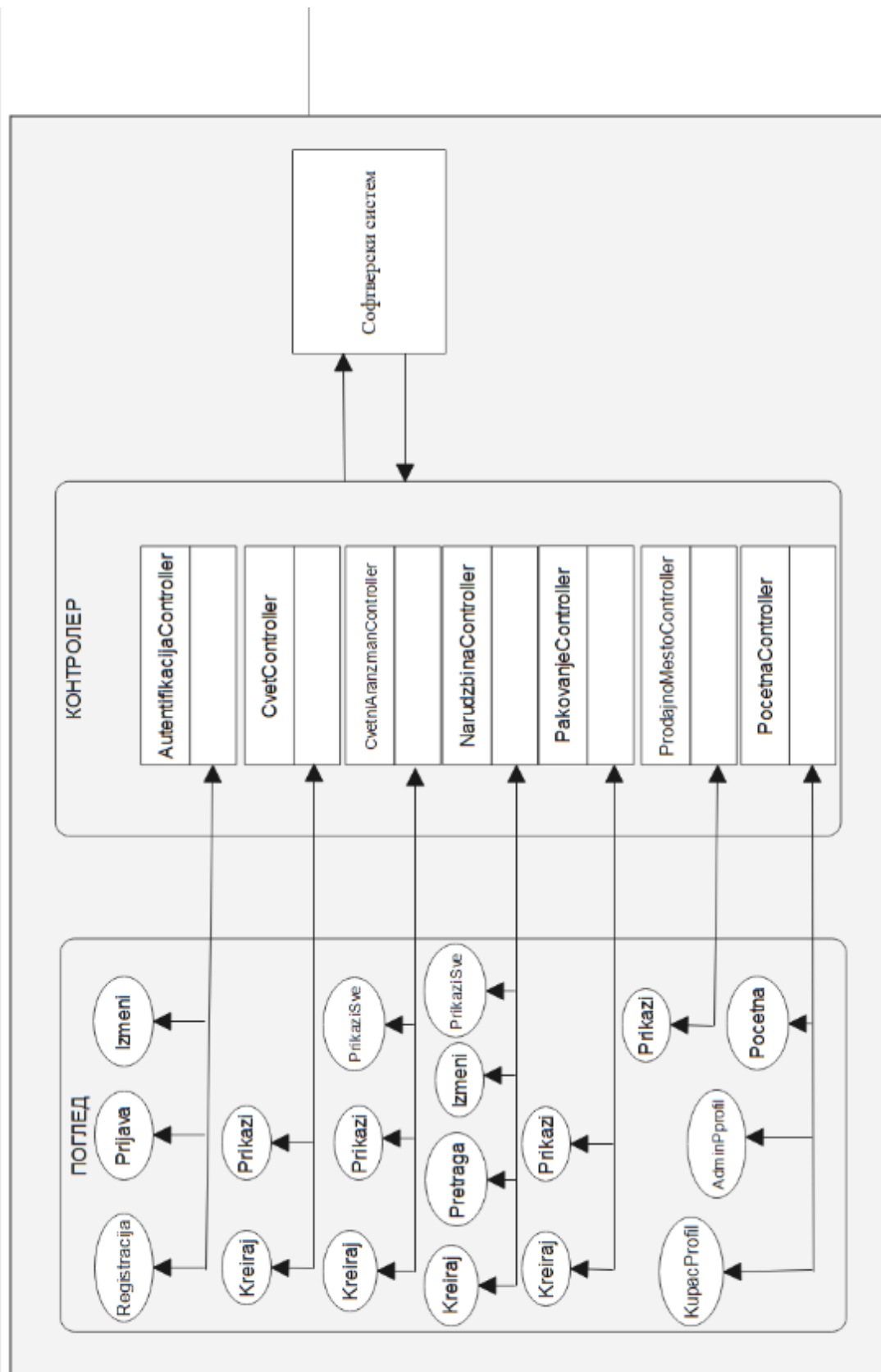


Database

Слика 115 Архитектура софтверског система (Први део)



Слика 116 Архитектура софтверског система (Други део)



Слика 117 Архитектура софтверског система (Трећи део)

7. ИМПЛЕМЕНТАЦИЈА

Софтверски систем је пројектован као веб апликација развијена је у програмском језику у C#, у .NET окружењу уз помоћ MVC оквира. Као развојно окружење коришћен је Microsoft Visual Studio 2019. Систем за управљање базом података је SQL Server, док је клијентска страна изграђена уз помоћ JavaScript, jQuery, AJAX и Bootstrap технологија.

7.1 Имплементација екранских форми

За сваки случај коришћења приказан је изглед екранских форми:

СК1: Случај коришћења – Приказивање детаља цвета

Назив СК

Приказивање детаља цвета

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Купцу је приказана листа цвећа.

Odaberite cveće:

Naziv	Boja	Cena		
Ruza	Crvena	200.00	Detalji	Dodaj
Orhideja	Ljubicasta	150.00	Detalji	Dodaj
Muskatla	Crvena	120.00	Detalji	Dodaj

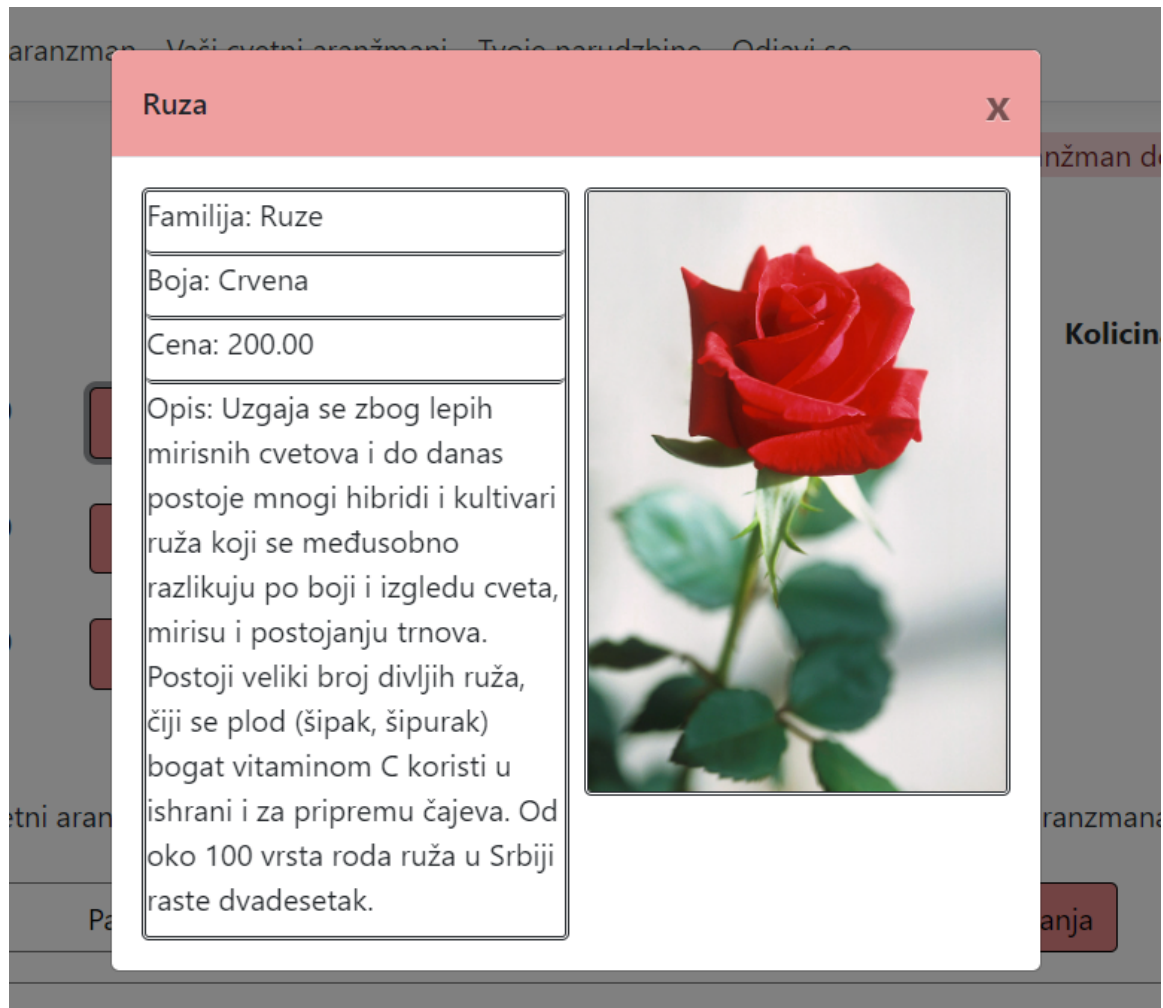
Слика 118 Листа цветова

Основни сценарио СК

1. Купац **бира** цвет коју жели да му се прикаже. (АПУСО)
2. Купац **позива** систем да му прикаже избрани цвет. (АПСО)

Опис акције: Купац кликом на дугме „Detalji“ позива системску операцију ПрикажиЦвет(Цвет)

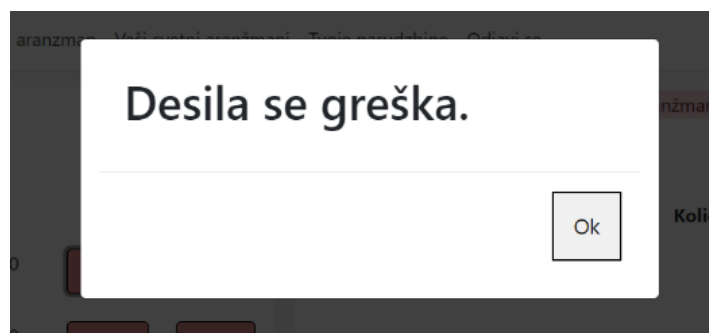
3. Систем **тражи** цвет по задатој вредности. (СО)
4. Систем **приказује** кориснику цвет. (ИА)



Слика 119 Детаљи цвета

Алтернативна сценарија

4.1. Уколико систем не може да пронађе цвет, купцу се приказује порука : „Десила се грешка“ (ИА)



Слика 120 Грешка у проналажењу цвета

СК2: Случај коришћења – Приказивање детаља цветног аранжмана

Назив СК

Приказивање детаља цветног аранжмана

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Купцу је приказана листа цветних аранжмана.

Naziv	Vrsta	Cena	Detalji	Dodaj
Rodjendanski poklon	Cvetni aranzman	1640.00	Detalji	Dodaj
Ljubicica	Cvet	140.00	Detalji	Dodaj

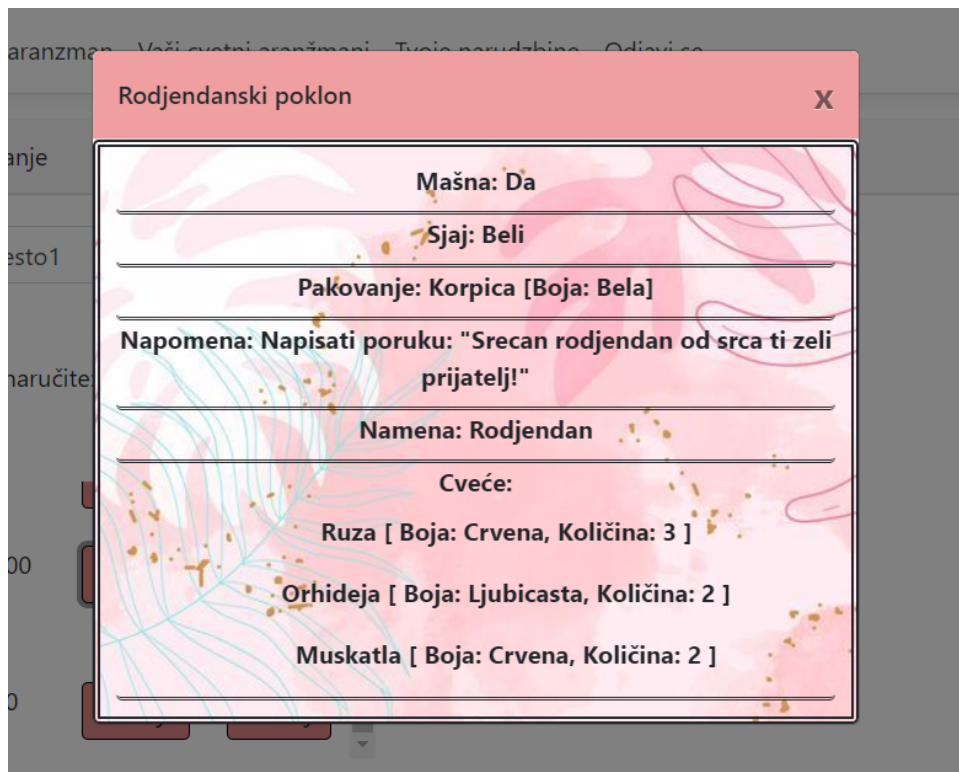
Слика 121 Листа цветних аранжмана

Основни сценарио СК

1. Купац **бира** цветни аранжман коју жели да му се прикаже. (АПУСО)
2. Купац **позива** систем да му прикаже избрани цветни аранжман. (АПСО)

Опис акције: Купац кликом на дугме „Detalji“ позива системску операцију ПрикажиЦветниАранжман(ЦветниАранжман)

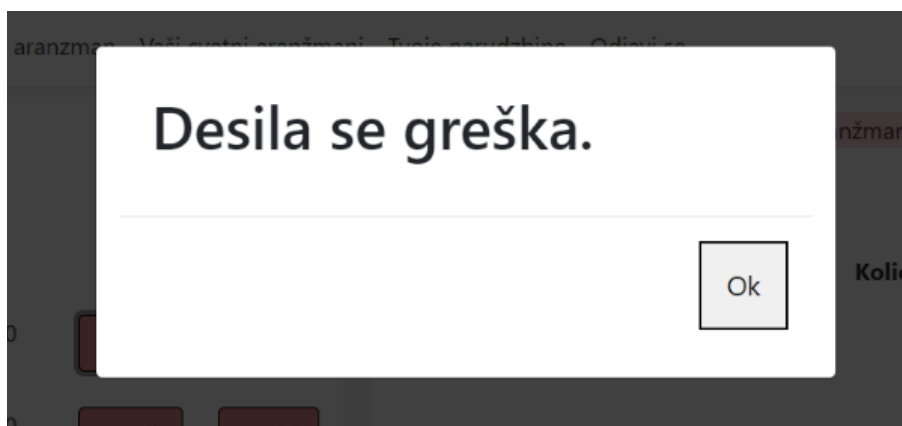
3. Систем **тражи** цветни аранжман по задатој вредности. (СО)
4. Систем **приказује** кориснику цветни аранжман. (ИА)



Слика 122 Детаљи цветног аранжмана

Алтернативна сценарија

4.1. Уколико систем не може да пронађе цветни аранжман, кориснику се приказује порука : „ Десила се грешка“ (ИА)



Слика 123 Грешка проналажења цветног аранжмана

СКЗ : Случај коришћења – Претраживање наруџбина

Назив СК

Претраживање наруџбина

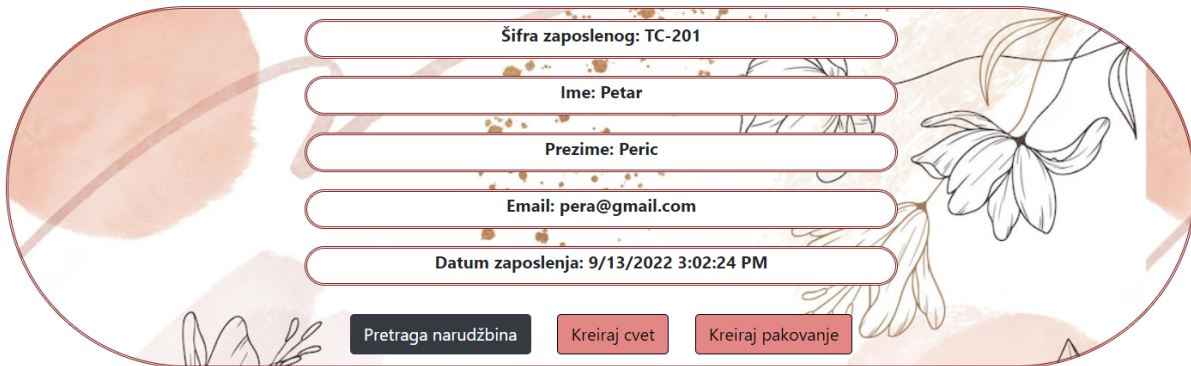
Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Администратор је регистрован на свој налог.



Šifra zaposlenog: TC-201

Ime: Petar

Prezime: Peric

Email: pera@gmail.com

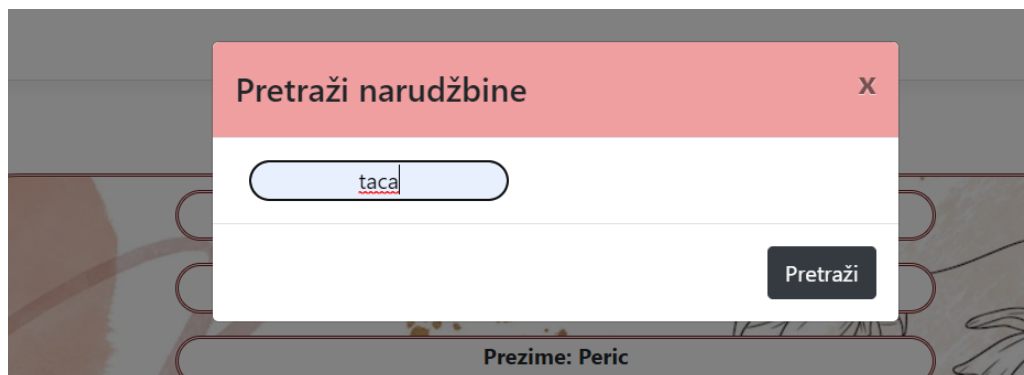
Datum zaposlenja: 9/13/2022 3:02:24 PM

Pretraga narudžbina Kreiraj cvet Kreiraj pakovanje

Слика 124 Регистровани администратор

Основни сценарио СК

1. Администратор **уноси** критеријум претраживања наруџбина. (АПУСО)



Pretraži narudžbine

taca

Pretraži

Prezime: Peric

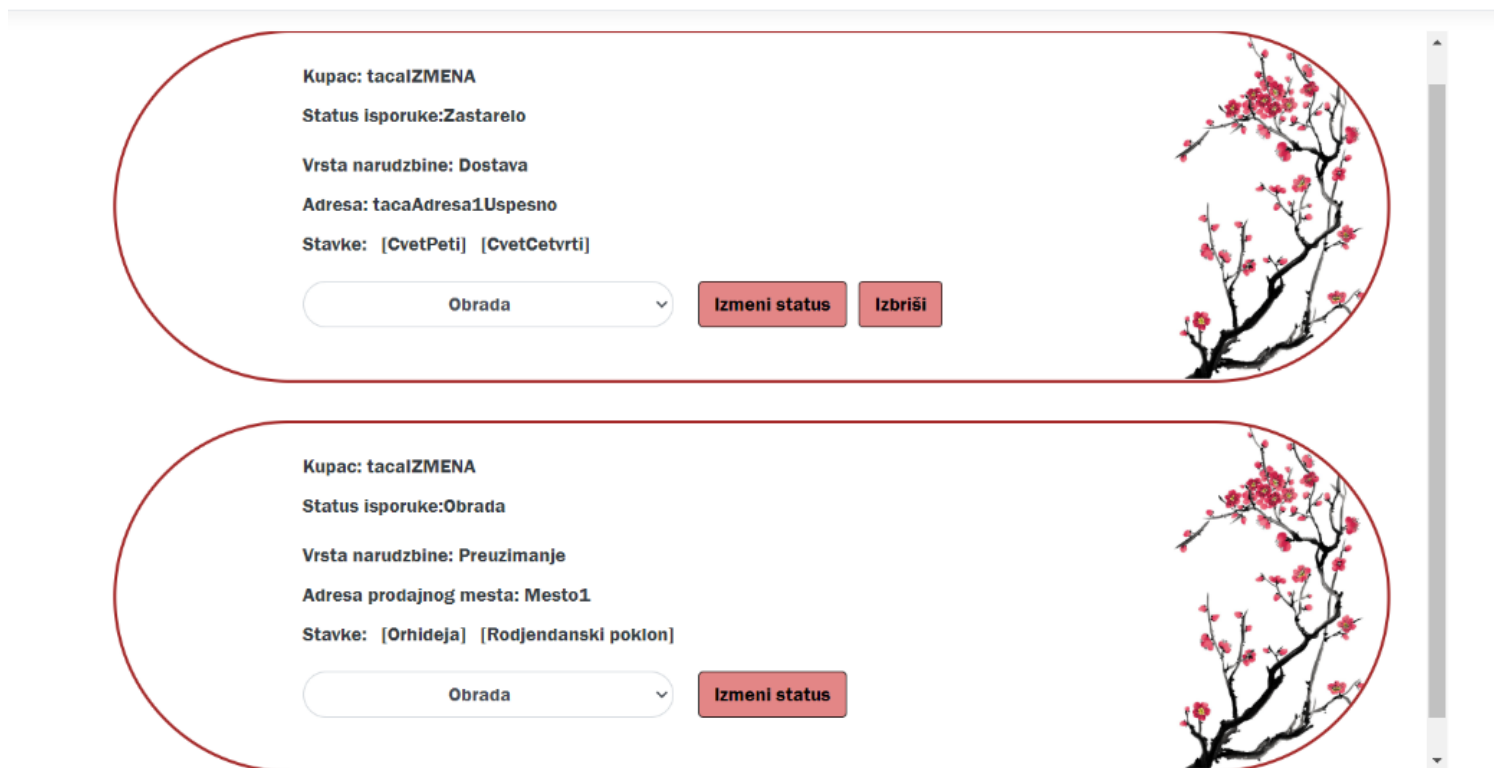
Слика 125 Претрага наруџбина

2. Администратор **контролише** да ли је коректно унео корисничко име за претрагу. (АНСО)

- Администратор **позива** систем да претражи наруџбине по корисничком имену. (АПСО)

Опис акције: Администратор кликом на дугме „Pretraži“ позива системску операцију ПрикажиНаруџбине(List<Наруџбина>)

- Систем **претражује** наруџбине. (СО)
- Систем **приказује** администратору наруџбине. (ИА)



Слика 126 Приказ наруџбина

Алтернативна сценарија

- Уколико систем не може да претражи наруџбине, администратору се приказује порука: „Купац нема наруџбине“ (ИА)

Купац нема наруџбине.

Слика 127 Порука о наруџбинама купца

СК4 : Случај коришћења – Пријављивање на систем

Назив СК

Пријављивање на систем

Актери СК

Администратор/купац

Учесници СК

Администратор/купац и систем

Предуслов: Систем је укључен и приказује форму за пријављивање

Lycoris Radiata

Prijava



Слика 128 Форма за пријаву

Основни сценарио СК

1. Администратор/купац **уноси** податке за пријављивање. (АПУСО)

Lycoris Radiata Prijava

Korisnicko ime:

Lozinka:

[Nemas nalog? Kreiraj ga!](#)

Слика 129 Попуњена форма за пријаву

- Администратор/купац **контролише** да ли је коректно унео корисничко име и лозинку (АНСО)
- Администратор/купац **позива** систем да га пријави. (АПСО)

Опис акције: Администратор/купац кликом на дугме „Prijavi se“ позива системску операцију Пријава(КорисничкоИме, Лозинка)

- Систем **проверава** да ли су унети подаци у реду. (СО)
- Систем **приказује** администратору/купцу доступне опције. (ИА)

Lycoris Radiata [Kreiraj cvetni aranžman](#) [Vaši cvetni aranžmani](#) [Tvoje narudzbine](#) [Odjavi se](#)

Ime: Tatjana

Prezime: Bailovic

Korisnicko ime: taca

Email: bailovict@gmail.com

Broj telefona: 12345678

Слика 130 Приказ профила купца

Алтернативна сценарија

5.1 Уколико систем не може да верификује администратора/купца он приказује поруку: „Корисник не постоји” (ИА)

- 
- Korisnik ne postoji.

Слика 131 Приказ поруке да корисник не постоји

СК5 : Случај коришћења – Измена наруџбине

Назив СК

Измена наруџбине

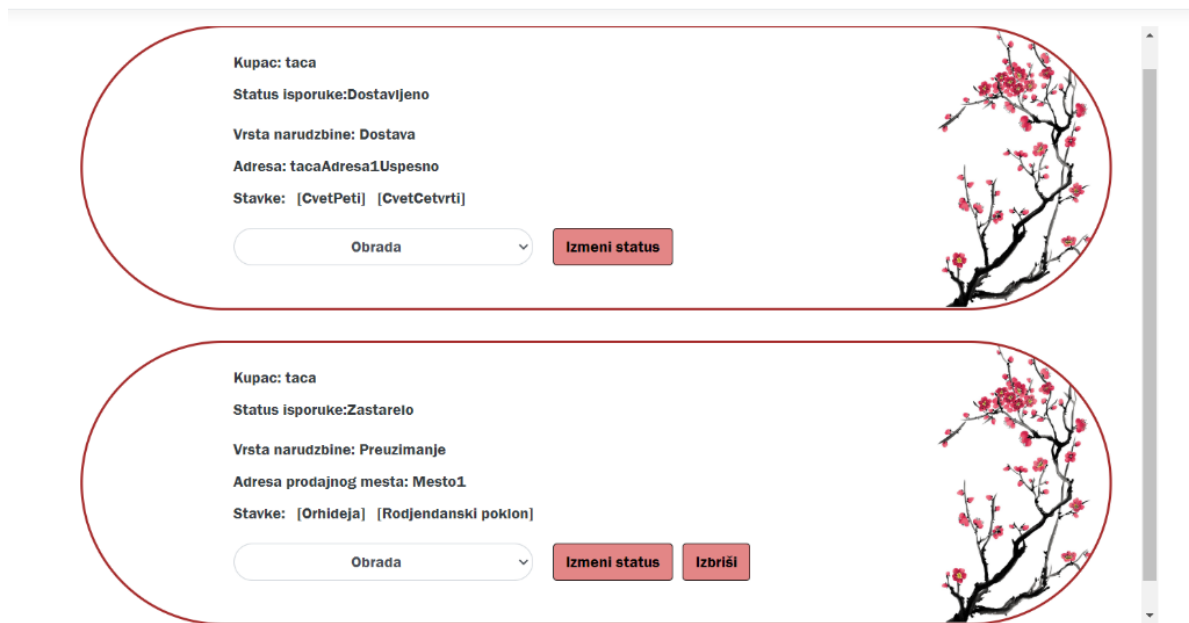
Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Учитане су наруџбине и листа опција за измену статуса наруџбине. Систем приказује форму за рад са наруџбином.



The screenshot displays two order cards in a list. Each card is enclosed in a rounded rectangle with a red border and contains the following information:

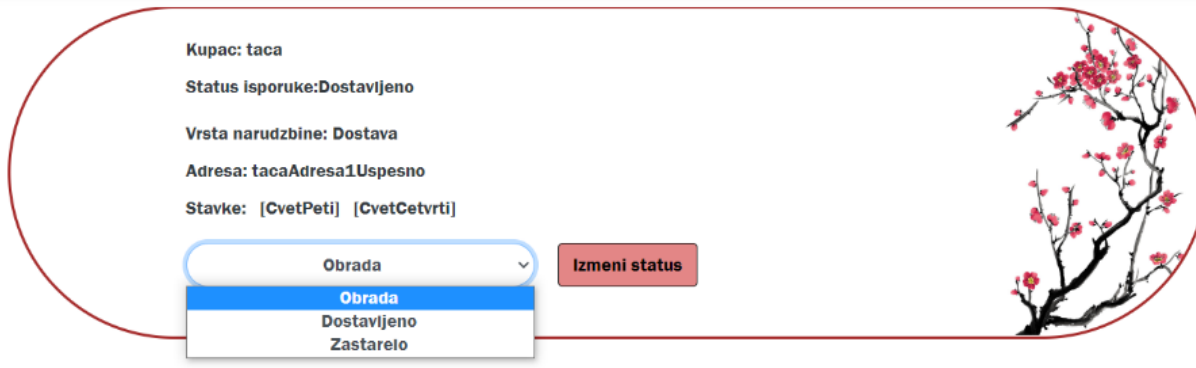
- Order 1:**
 - Kupac: taca
 - Status isporuke: Dostavljeno
 - Vrsta narudzbine: Dostava
 - Adresa: tacaAdresa1Uspesno
 - Stavke: [CvetPeti] [CvetCetvrti]
 - Buttons: Obrada (dropdown), Izmeni status
- Order 2:**
 - Kupac: taca
 - Status isporuke: Zastarelo
 - Vrsta narudzbine: Preuzimanje
 - Adresa prodajnog mesta: Mesto1
 - Stavke: [Orhideja] [Rodjendanski poklon]
 - Buttons: Obrada (dropdown), Izmeni status, Izbrisi

Each card also features a decorative illustration of a flowering branch on the right side.

Слика 132 Листа наруџбина одабраног купца

Основни сценарио СК

1. Администратор **бира** наруџбину коју жели да измени. (АПУСО)
2. Администратор **мења** податке о наруџбини (АПУСО)

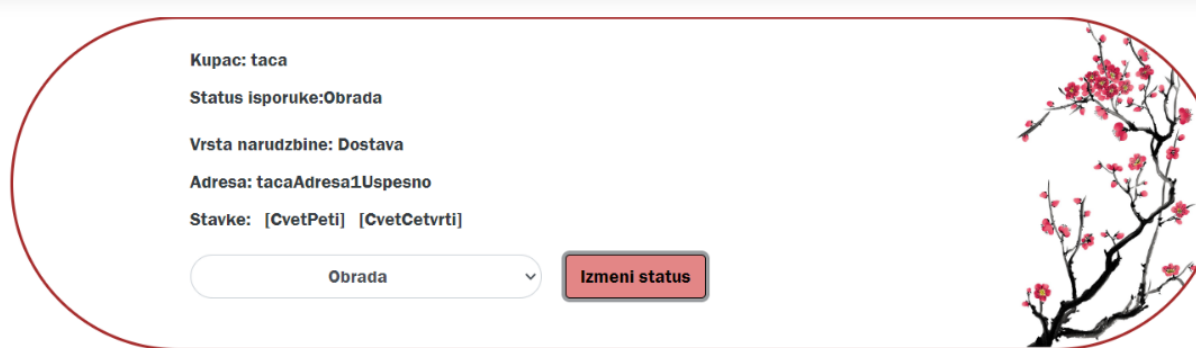


Слика 133 Приказ измене наруџбине

3. Администратор **контролише** да ли је коректно унео податке о наруџбини. (АНСО)
4. Администратор **позива** систем да запамти податке о наруџбини. (АПСО)

Опис акције: Администратор кликом на дугме „Izmeni status“ позива системску операцију ИзмениНаруџбину(Наруџбина)

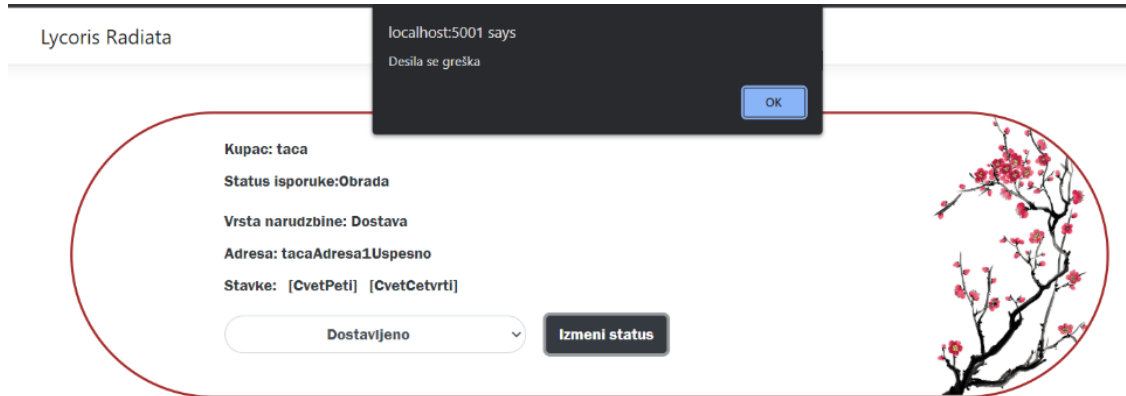
5. Систем **памти** податке о наруџбини. (СО)
6. Систем **приказује** администратору забележену измену. (ИА)



Слика 134 Приказ измењене наруџбине

Алтернативна сценарија

6.1 Уколико систем не може да запамти податке о наруџбини он приказује администратору поруку „Десила се грешка” (ИА)



Слика 135 Приказ грешке при измени наруџбине

СК6 : Случај коришћења – Креирање цветног аранжмана

Назив СК

Креирање цветног аранжмана

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитна је листа цветова и паковања. Систем приказује форму за рад са цветним аранжманом.

Lycoris Radiata Kreiraj цветни aranžman Vaši цветни aranžmani Tvoje narudzbine Odjavi se Naruci

Naziv: Ne možete kreirati цветни aranžman dok ne unesete njegov naziv.

Odaberite цвеће: Odabrano цвеће:

Naziv	Boja	Cena			Naziv	Boja	Kolicina
CvetTreci	Zuta	300.00	<input type="button" value="Detalji"/>	<input type="button" value="Dodaj"/>			
CvetCetvrti	Plava	400.00	<input type="button" value="Detalji"/>	<input type="button" value="Dodaj"/>			
CvetPeti	Ljubicasta	500.00	<input type="button" value="Detalji"/>	<input type="button" value="Dodaj"/>			

Čekiraj polje ako ukoliko da tvoj цветни aranžman ima mašnu: Izaberi sjaj: Namena цветног aranžmana:

Izaberi pakovanje: Katalog pakovanja

Napomena (opciono polje):

Слика 136 Форма за креирање цветног аранжмана

Основни сценарио СК

1. Купац **уноси** податке о цветном аранжману. (АПУСО)

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se Naruci

Naziv:

Odaberite cveće:

Naziv	Boja	Cena	
Ruza	Crvena	200.00	<input type="button" value="Detalji"/>
Orhideja	Ljubicasta	150.00	<input type="button" value="Detalji"/>
Muskatla	Crvena	120.00	<input type="button" value="Detalji"/>

Odabrano cveće:

Naziv	Boja	Kolicina	
Orhideja	Ljubicasta	<input type="text" value="2"/>	<input type="button" value="Ukloni"/>
Muskatla	Crvena	<input type="text" value="2"/>	<input type="button" value="Ukloni"/>
Ruza	Crvena	<input type="text" value="3"/>	<input type="button" value="Ukloni"/>

Čekiraj polje ako ukoliko da tvoj cvetni aranžman ima mašnu: Izaberi sjaj: Namena cvetnog aranžmana:

Izaberi pakovanje:

Napomena (opciono polje):

© 2022 - LycorisRadiata - Privacy

Слика 137 Попуњена форма за креирање цветног аранжмана

2. Купац **контролише** да ли је коректно унео податке о цветном аранжману. (АНСО)
3. Купац **позива** систем да креира нови цветни аранжман са задатим подацима. (АПСО)

Опис акције: Купац кликом на дугме „Креирај“ позива системску операцију КреирајЦветниАранжман(ЦветниАранжман)

4. Систем **креира** цветни аранжман са задатим подацима. (СО)
5. Систем **приказује** купцу забележени цветни аранжман. (ИА)



Слика 138 Приказ креираног цветног аранжмана

Алтернативна сценарија

5.1 Уколико систем не може да креира цветни аранжман он приказује купцу поруку „Десила се грешка” (ИА)

Desila se greška.



Слика 139 Испис грешке

СК7 : Случај коришћења – Креирање корисничког налога

Назив СК

Креирање корисничког налога

Актери СК

Посетилац

Учесници СК

Посетилац и систем

Предуслов: Систем је укључен. Систем приказује форму за креирање корисничког налога

Ime:

Prezime:

Korisničko ime:

Email:

Broj telefona:

Lozinka:

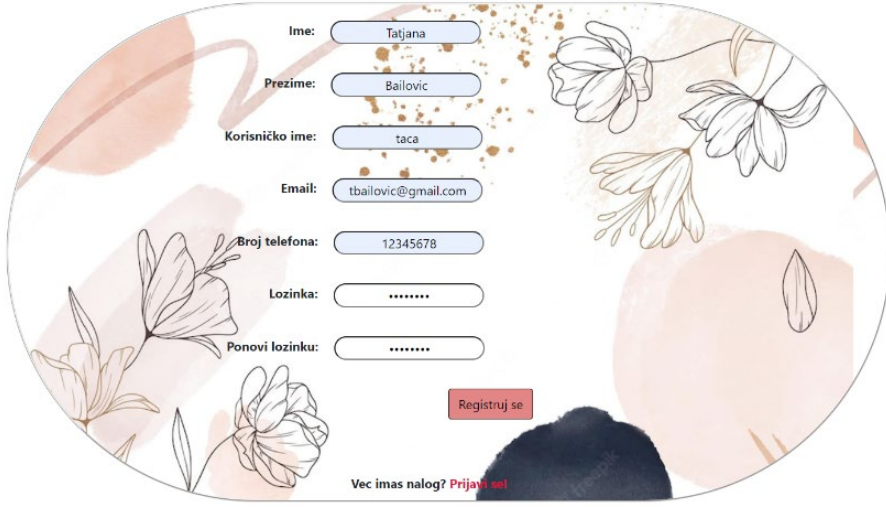
Ponovi lozinku:

[Vec imas nalog? Prijavi se!](#)

Слика 140 Форма за регистрацију

Основни сценарио СК

1. Посетилац **уноси** податке о новом купцу. (АПУСО)



Ime:

Prezime:

Korisnicko ime:

Email:

Broj telefona:

Lozinka:

Ponovi lozinku:

Vec imas nalog? [Prijavi se!](#)

Слика 141 Попуњена форма за регистрацију

2. Посетилац **контролише** да ли је коректно унео податке о купцу. (АНСО)
3. Посетилац **позива** систем да креира новог купца са задатим подацима. (АПСО)

Опис акције: Посетилац кликом на дугме „Registruj se“ позива системску операцију КреирајКупца(Купац)

4. Систем **креира** новог купца. (СО)
5. Систем **приказује** посетиоцу поруку страницу за пријаву. (ИА)



Korisnicko ime:

Lozinka:

Nemas nalog? [Kreiraj ga!](#)

Слика 142 Празна форма за пријаву

Алтернативна сценарија

5.1 Уколико систем не може да креира купца он приказује посетиоцу поруку „Десила се грешка” (ИА)

A screenshot of a web page showing an error message. The message is a red bullet point followed by the text "Desila se greška." in red. The background is a light beige color.

- **Desila se greška.**

Слика 143 Порука грешке приликом пријаве

СК8 : Случај коришћења – Брисање наруџбине

Назив СК

Брисање наруџбне

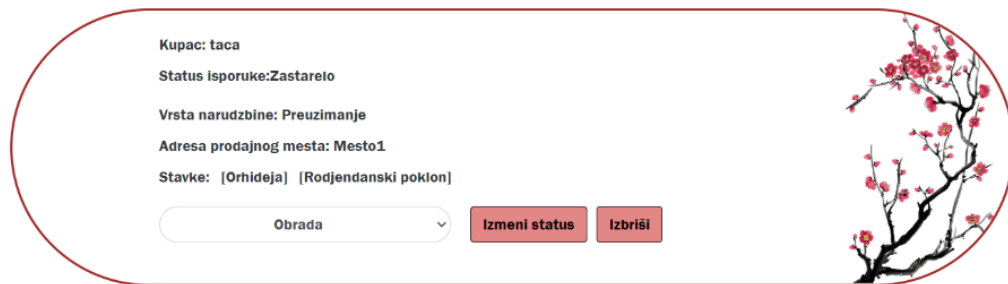
Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са наруџбином



The screenshot shows a web form for managing orders. The form contains the following text:

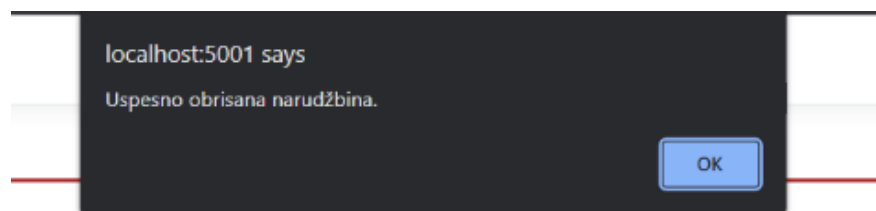
- Kupac: taca
- Status isporuke: Zastarelo
- Vrsta narudzbine: Preuzimanje
- Adresa prodajnog mesta: Mesto1
- Stavke: [Orhideja] [Rodjendanski poklon]

At the bottom of the form, there is a dropdown menu with the value "Obrada" and two buttons: "Izmeni status" and "Izbrisi". To the right of the form is a decorative image of a branch with red flowers.

Слика 144 Форма за рад са наруџбином

Основни сценарио СК

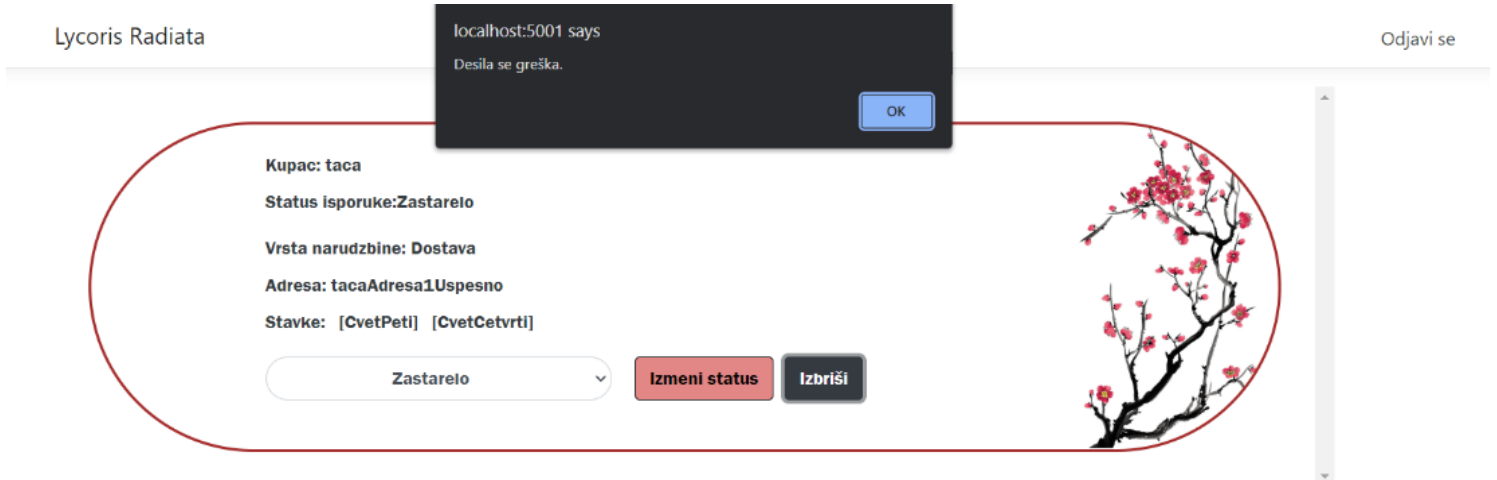
1. Администратор **бира** наруџбину коју жели да обрише. (АПУСО)
2. Администратор **позива** систем да обрише податке о наруџбини. (АПСО)
3. Опис акције: Администратор кликом на дугме „Izbrisi“ позива системску операцију ОбришиНаруџбину(Наруџбина)
4. Систем **брише** податке о наруџбини. (СО)
5. Систем **приказује** администратору поруку: „Успешно обрисана наруџбина.“ (ИА)



Слика 145 Порука о успешно обрисаној наруџбини

Алтернативна сценарија

5.1 Уколико систем не може да обрише податке о наруџбини он приказује администратору поруку „Десила се грешка” (ИА)



Слика 146 Порука о неуспешно обрисаној наруџбини

СК9 : *Случај коришћења – Измена корисничког налога*

Назив СК

Измена корисничког налога

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитани су подаци о купцу. Систем приказује форму за рад са профилем купца.

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se Naruci

Ime: Tatjana
Prezime: Bailovic
Korisnicko ime: taca
Email: bailovic@gmail.com
Broj telefona: 123456789
Stara lozinka:
Nova lozinka:
Sacuvaj promene

Слика 147 Форма за измену података купца

Основни сценарио СК

1. Систем **тражи** податке о купцу. (СО)
2. Систем **приказује** купцу пронађене податке. (ИА)
3. Купац **мења** податке. (АПУСО)

Ime:
 Prezime:
 Korisnicko ime:
 Email:
 Broj telefona:
 Stara lozinka:
 Nova lozinka:

Слика 148 Измењени подаци о купцу

4. Купац **контролише** да ли је коректно унео податке о купцу. (АНСО)
5. Купац **позива** систем да запамти податке. (АПСО)

Опис акције: Купац кликом на дугме „Сачувај промене“ позива системску операцију ИзмениКупца(Купац)

6. Систем **памти** податке о купцу. (СО)
7. Систем **приказује** купцу измењене податке. (ИА)

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se

Ime: TatjanaZMENA
 Prezime: BailoviciZMENA
 Korisnicko ime: tacaZMENA
 Email: bailovict@gmail.com
 Broj telefona: 12345678

Слика 149 Профил купца

Алтернативна сценарија

7.1 Уколико систем не може да запамти податке о купцу он приказује купцу поруку „Десила се грешка” (ИА)

Desila se greška.



Слика 150 Грешка приликом измене купца

СК10 : Случај коришћења – Креирање цвета

Назив СК

Креирање цвета

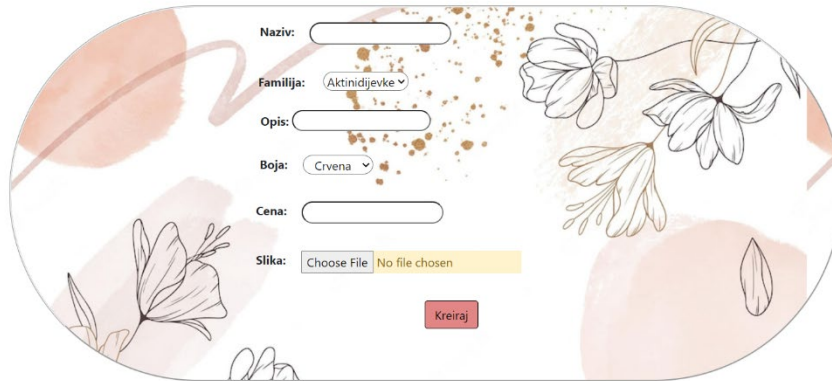
Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са цветом.



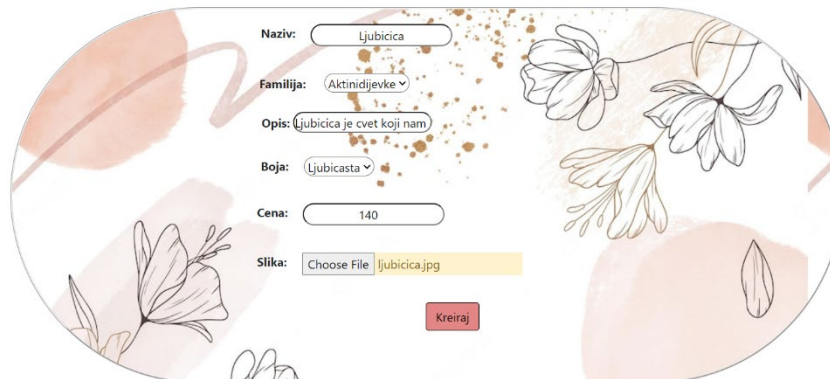
The screenshot shows a web form titled 'Create Flower' (Креирање цвета) set against a decorative background of flowers and leaves. The form fields are as follows:

- Naziv:** An empty text input field.
- Familija:** A dropdown menu with 'Aktinidijevke' selected.
- Opis:** An empty text input field.
- Boja:** A dropdown menu with 'Crvena' selected.
- Cena:** An empty text input field.
- Slika:** A file selection area with a 'Choose File' button and the text 'No file chosen'.
- Creiraj:** A red button at the bottom right of the form.

Слика 151 Форма за креирање цвета

Основни сценарио СК

1. Администратор уноси податке о цвету. (АПУСО)



The screenshot shows the same 'Create Flower' form, but now with data entered into the fields:

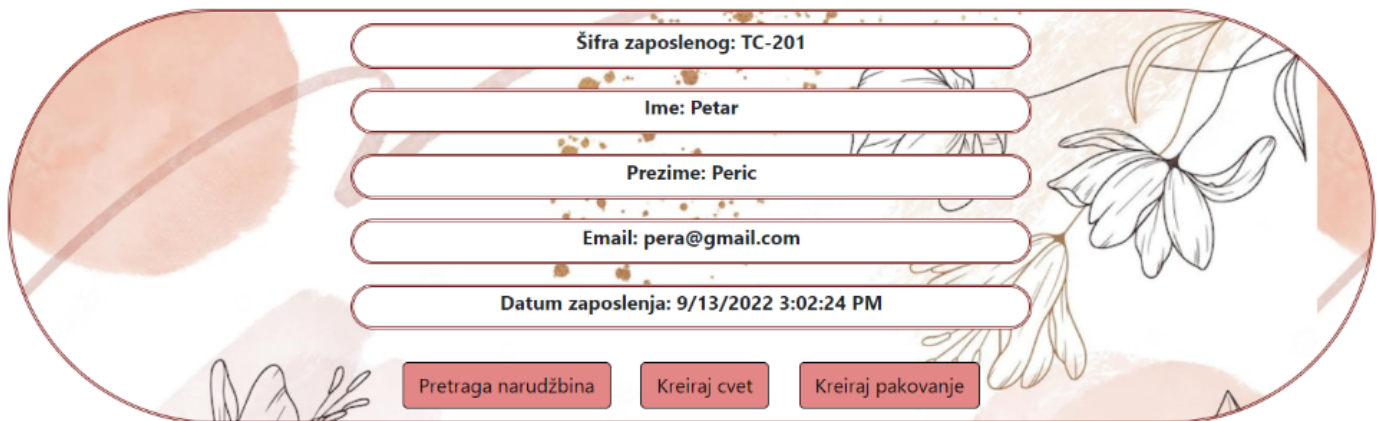
- Naziv:** 'Ljubicica' is entered in the text input field.
- Familija:** 'Aktinidijevke' is selected in the dropdown menu.
- Opis:** 'Ljubicica je cvet koji nam' is entered in the text input field.
- Boja:** 'Ljubicasta' is selected in the dropdown menu.
- Cena:** '140' is entered in the text input field.
- Slika:** The file selection area shows 'Ljubicica.jpg' as the chosen file.
- Creiraj:** The red button remains at the bottom right.

Слика 152 Попуњена форма за креирање цвета

2. Администратор **контролише** да ли је коректно унео податке о цвету. (АНСО)
3. Администратор **позива** систем да креира нови цвет са задатим подацима. (АПСО)

Опис акције: Администратор кликом на дугме „Креирај“ позива системску операцију КреирајЦвет(Цвет)

4. Систем **креира** цвет са задатим подацима. (СО)
5. Систем **приказује** администратору почетну страну. (ИА)



The image shows a user profile form with the following fields and buttons:

- Šifra zaposlenog: TC-201
- Ime: Petar
- Prezime: Peric
- Email: pera@gmail.com
- Datum zaposlenja: 9/13/2022 3:02:24 PM
- Buttons: Pretraga narudžbina, Kreiraj cvet, Kreiraj pakovanje

Слика 153 Профил администратора

Алтернативна сценарија

5.1 Уколико систем не може да креира цвет он приказује администратору поруку „Десила се грешка” (ИА)

Desila se greška.



Слика 154 Испис грешке

СК11 : Случај коришћења – Креирање паковања

Назив СК

Креирање паковања

Актери СК

Администратор

Учесници СК

Администратор и систем

Предуслов: Систем је укључен и администратор је улогован под својом шифром. Систем приказује форму за рад са паковањем.



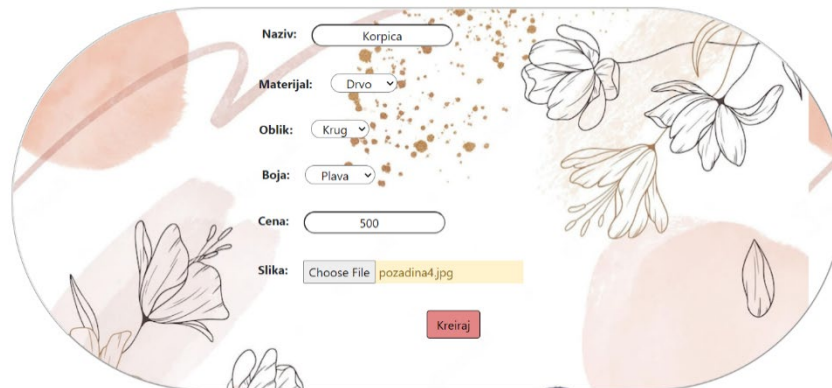
The screenshot shows a web form for creating a package. The form is set against a decorative background of watercolor-style flowers and leaves. The fields are as follows:

- Naziv:** An empty text input field.
- Materijal:** A dropdown menu with "Plastika" selected.
- Oblik:** A dropdown menu with "Krug" selected.
- Boja:** A dropdown menu with "Crvena" selected.
- Cena:** An empty text input field.
- Slika:** A file selection area with a "Choose File" button and the text "No file chosen".
- Kreiraj:** A red button at the bottom right of the form.

Слика 155 Форма за креирање паковања

Основни сценарио СК

1. Администратор уноси податке о паковању. (АПУСО)



The screenshot shows the same web form as in the previous image, but with the following data entered:

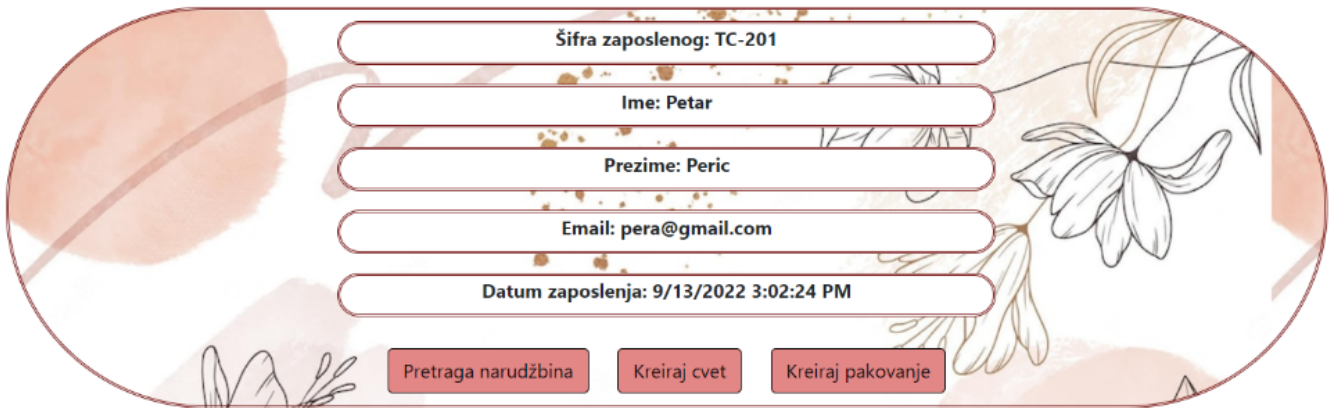
- Naziv:** "Korpica"
- Materijal:** "Drvo"
- Oblik:** "Krug"
- Boja:** "Plava"
- Cena:** "500"
- Slika:** "Choose File" button and "pozadina4.jpg" text.
- Kreiraj:** A red button at the bottom right of the form.

Слика 156 Попуњена форма з креирање паковања

- Администратор **контролише** да ли је коректно унео податке о паковању. (АНСО)
- Администратор **позива** систем да креира ново паковње са задатим подацима. (АПСО)

Опис акције: Администратор кликом на дугме „Креирај“ позива системску операцију КреирајПаковање(Паковање)

- Систем **креира** паковање са задатим подацима. (СО)
- Систем **приказује** администратору почетну страну. (ИА)



Слика 157 Профил администратора

Алтернативна сценарија

5.1 Уколико систем не може да креира паковање он приказује администратору поруку „Десила се грешка” (ИА)

Desila se greška.



Слика 158 Порука грешке приликом креирања паковања

СК12 : Случај коришћења – Одјава са система

Назив СК

Одјава са система

Актери СК

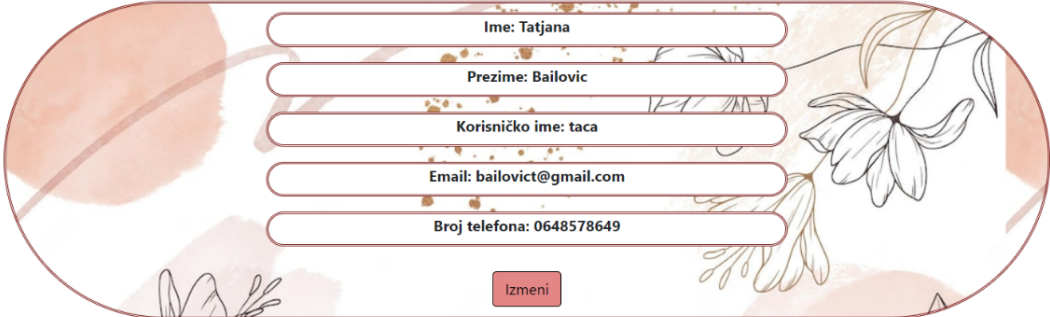
Администратор/купац

Учесници СК

Администратор/купац и систем

Предуслов: Систем је укључен и администратор/купац је улогован под својом шифром.

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se Naruci



Ime: Tatjana
Prezime: Bailovic
Korisničko ime: taca
Email: bailovict@gmail.com
Broj telefona: 0648578649
Izmeni

Слика 159 Улогован купац

Основни сценарио СК

1. Администратор/купац позива систем да га одјави. (АПСО)

Опис акције: Администратор/купац кликом на дугме „Odjavi se“ позива системску операцију Одјави(Особа)

2. Систем одјављује администратора/купца са система и приказује почетну страну. (ИА)



Korisnicko ime:

Lozinka:

Prijavi se

Nemas nalog? Kreiraj ga!

Слика 160 Страна за пријаву

Алтернативна сценарија

2.1 Уколико систем не може да одјави администратора/купца са система: „Десила се грешка”. (ИА)

Desila se greška.



Слика 161 Испис грешке

СК13 : *Случај коришћења – Креирање наруџбине*

Назив СК

Креирање наруџбине

Актери СК

Купац

Учесници СК

Купац и систем

Предуслов: Систем је укључен и купац је улогован под својом шифром. Учитна је листа производа и продајних места. Систем приказује форму за рад са наруџбином.

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se **Naruci**

Tip narudzbine: Lično preuzimanje

Adresa preuzimanja: Mesto1 **Prodajna mesta**

Odaberite proizvode koje želite da naručite:

Naziv	Vrsta	Cena		
Orhideja	Cvet	150.00	Detalji	Dodaj
Muskatla	Cvet	120.00	Detalji	Dodaj
Rodjendanski poklon	Cvetni aranžman	1640.00	Detalji	Dodaj

Odabrani proizvodi:

Naziv	Kolicina
-------	----------

Naruči

Слика 162 *Форма за креирање наруџбине*

Основни сценарио СК

1. Купац уноси податке за нову наруџбину. (АПУСО)

Lycoris Radiata Kreiraj cvetni aranžman Vaši cvetni aranžmani Tvoje narudzbine Odjavi se Naruci

Tip narudzbine: Lično preuzimanje

Adresa preuzimanja: Mesto1 Prodajna mesta

Odaberite proizvode koje želite da naručite :

Naziv	Vrsta	Cena	
Orhideja	Cvet	150.00	Detalji
Muskatla	Cvet	120.00	Detalji Dodaj
Rodjendanski poklon	Cvetni aranžman	1640.00	Detalji

Odabrani proizvodi:

Naziv	Kolicina	
Orhideja	3	Ukloni
Rodjendanski poklon	1	Ukloni

Naruci

Слика 163 Попуњена форма за креирање наруџбине

2. Купац **контролише** да ли је коректно унео податке о наруџбини. (АНСО)
3. Купац **позива** систем да креира нову наруџбину са задатим подацима. (АПСО)

Опис акције: Купац кликом на дугме „Наруџи“ позива системску операцију КреирајНаруџбину(Наруџбина)

4. Систем **креира** наруџбину са задатим подацима. (СО)
5. Систем **приказује** купцу забележену наруџбину. (ИА)


Tip narudzbine: Lično preuzimanje

Adresa preuzimanja: Mesto1

Status narudzbine: Obrada

Stavke:

- Orhideja [Količina: 3]
- Rodjendanski poklon [Količina: 1]



Слика 164 Креирана наруџбина

Алтернативна сценарија

5.1 Уколико систем не може да креира наруџбину он приказује купцу поруку „Десила се грешка” (ИА)

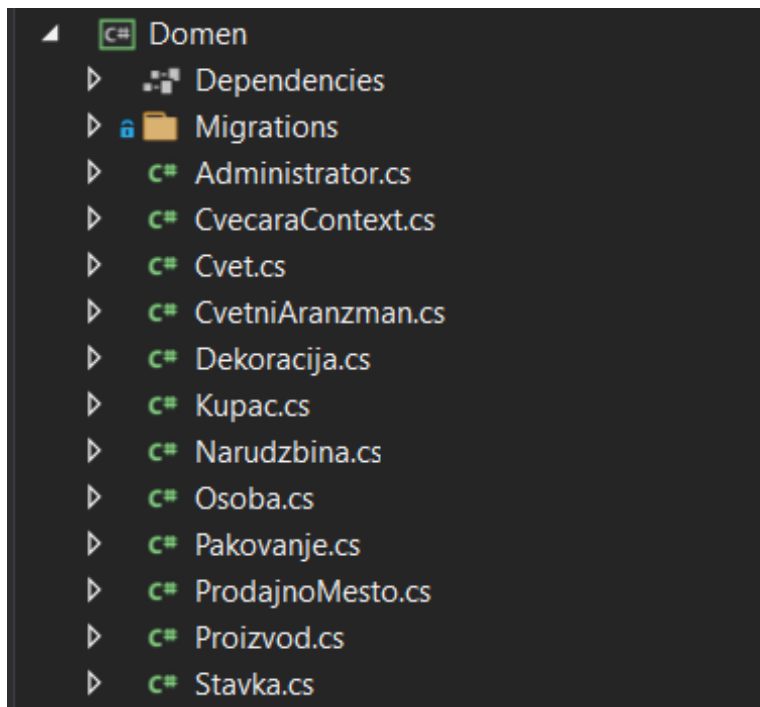
Desila se greška.



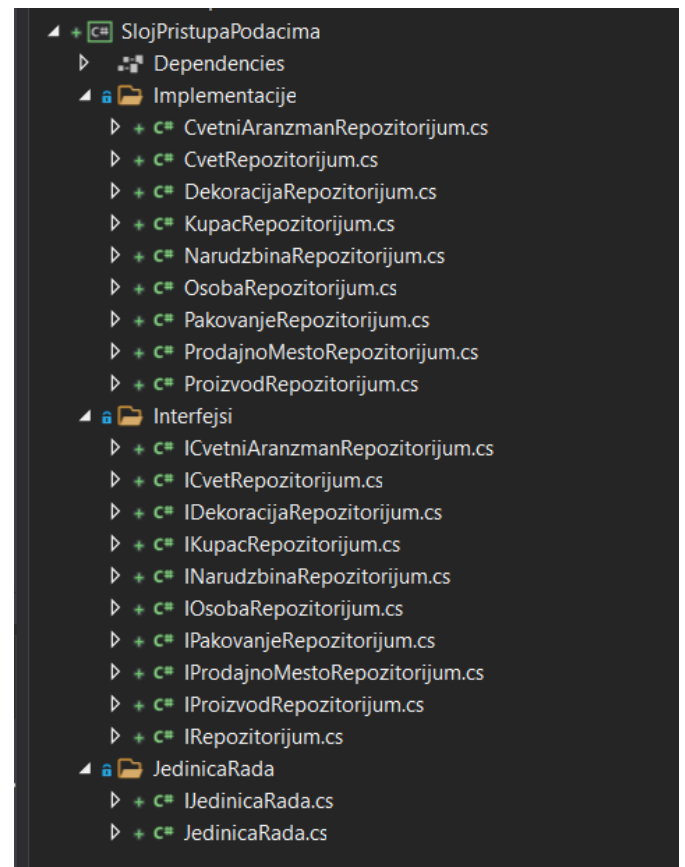
Слика 165 Грешка пиликом креирања наруџбине

7.1 Структура софтверског решења

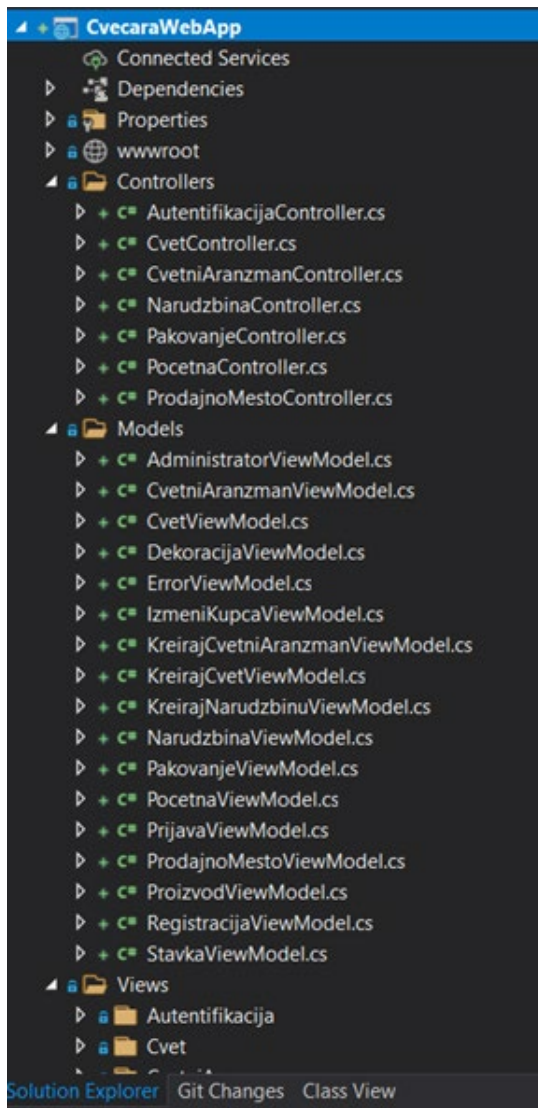
Приказ имплементираних класа:



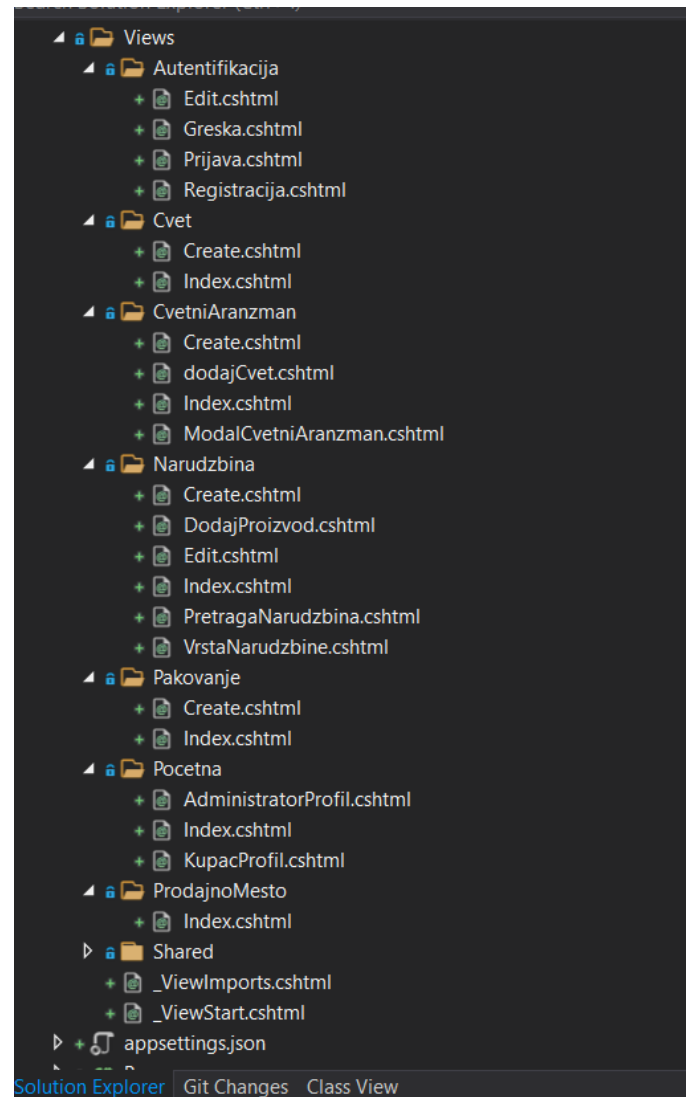
Слика 166 Имплементација класе Домен



Слика 167 Имплементација Слоја приступа подацима



Слика 168 Имплементација Контролера и модела



Слика 169 Имплементација погледа

7.2 *Имплементација апликационе логике*

У наставку је описана имплементација апликационе логике, која прима захтев од корисничког интерфејса, обрађује га користећи податке из складишта података и враћа одговор до корисника.

7.2.1 Имплементација комуникације са корисницима

Као што је већ објашњено у другом поглављу, свака апликација, па и ова, почиње своје извршавање у Main методи која се налази у класи Program. Main метода позива CreateHostBuilder методу, која позива две методе - CreateDefaultBuilder(), која подешава основне концепте апликације и ConfigureWebHostDefaults() која обезбеђује конфигурацију специфичну за апликацију, помоћу Startup класе. Имплементација Program класе приказана је у наставку.

```
public class Program
{
    public static void Main(string[] args)
    {
        CreateHostBuilder(args).Build().Run();
    }

    public static IHostBuilder CreateHostBuilder(string[] args) =>
        Host.CreateDefaultBuilder(args)
            .ConfigureWebHostDefaults(webBuilder =>
            {
                webBuilder.UseStartup<Startup>();
            });
}
```

Изглед Startup класе:

```
public class Startup
{
    public Startup(IConfiguration configuration)
    {
        Configuration = configuration;
    }

    public IConfiguration Configuration { get; }

    // This method gets called by the runtime. Use this method to add
    services to the container.
    public void ConfigureServices(IServiceCollection services)
    {
        services.AddControllersWithViews();
        services.AddScoped<IJedinicaRada, JedinicaRada>();
    }
}
```



```

        services.AddDbContext<CvecaraContext>(options => {
options.UseSqlServer(Configuration.GetConnectionString("baza")); });
        services.AddIdentity<Osoba,
IdentityRole<int>>().AddEntityFrameworkStores<CvecaraContext>();

services.AddAuthentication(CookieAuthenticationDefaults.AuthenticationScheme).
AddCookie();

        services.ConfigureApplicationCookie(options => {
            options.LoginPath = "/Autentifikacija/Prijava";
            options.AccessDeniedPath = "/Pocetna/Index";
            options.ExpireTimeSpan = TimeSpan.FromMinutes(5);
            options.SlidingExpiration = true;
        });
    }

    // This method gets called by the runtime. Use this method to
    // configure the HTTP request pipeline.
    public void Configure(IApplicationBuilder app, IWebHostEnvironment
env)
    {
        if (env.IsDevelopment())
        {
            app.UseDeveloperExceptionPage();
        }
        else
        {
            app.UseExceptionHandler("/Home/Error");
            // The default HSTS value is 30 days. You may want to change
            // this for production scenarios, see https://aka.ms/aspnetcore-hsts.
            app.UseHsts();
        }
        app.UseHttpsRedirection();
        app.UseStaticFiles();

        app.UseRouting();
        app.UseAuthentication();
        app.UseAuthorization();

        app.UseEndpoints(endpoints =>
        {
            endpoints.MapControllerRoute(
                name: "default",
                pattern: "{controller=Pocetna}/{action=Index}/{id?}");
        });
    }
}

```

7.2.2 Имплементација пословне логике

Имплементација доменских класа

Имплементација доменских класа се врши на основу дефинисаних концептуалних класа у доменском моделу и њихових релација, пишу се одговарајуће класе у програмском језику C#. Свака класа се састоји из приватних атрибута и јавних метода које омогућавају њихов приступ и измену. Доменске класе се налазе у оквиру библиотеке *Domen*. У наставку је приказан пример доменске класе *Narudzbina*:

```
public class Narudzbina
{
    public int NarudzbinaId { get; set; }
    public VrstaNarudzbine VrstaNarudzbine { get; set; }
    public string Adresa { get; set; }
    public StatusIsporuke StatusIsporuke { get; set; }
    public ProdajnoMesto ProdajnoMesto { get; set; }
    public int? ProdajnoMestoId { get; set; }
    public Kupac Kupac { get; set; }
    public int KupacId { get; set; }
    public List<Stavka> Stavke { get; set; }
}
```

Имплементација системских операција

Системске операције имплементиране су у оквиру одговарајућих *Controller* класа.

Уговор УГ1: ПрикажиЦвет

Операција: ПрикажиЦвет(Цвет)

Веза са СК: СК1

Предуслов: /

Постуслов: /

```
public Cvet Pretraga(int id) => jedinicaRada.CvetRepozitorijum.PretragaId(id);
```

Уговор УГ2: ПрикажиНаруџбине

Операција: ПрикажиНаруџбине(КорисничкоИме)

Вежа са СК: СК3

Предуслов: /

Постуслов: /

```
public List<Narudzina> Pretraga() =>
jedinicaRada.NarudzinaRepozitorijum.Pretraga(n => n.KupacId ==
Int32.Parse(HttpContext.User.Claims.ElementAt(0).Value));
```

Уговор УГ3: ПрикажиЦветниАранжман

Операција: ПрикажиЦветниАранжман(ЦветниАранжман)

Вежа са СК: СК2

Предуслов: /

Постуслов: /

```
public CvetniAranzman Pretraga(int id) =>
jedinicaRada.CvetniAranzmanRepozitorijum.PretragaId(id);
```

Уговор УГ4: ВратиЦвеће

Операција: ВратиЦвеће(List<Цвет>)

Вежа са СК: СК1, СК6

Предуслов: /

Постуслов: /

```
public List<Cvet> VratiSve() => jedinicaRada.CvetRepozitorijum.VratiSve();
```

Уговор УГ5: ВратиЦветнеАранжмане

Операција: ВратиЦветнеАранжмане(List<ЦветниАранжман>)

Вежа са СК: СК2

Предуслов: /

Постуслов: /

```
public List< CvetniAranzman > VratiSve() => jedinicaRada. CvetniAranzman
Repozitorijum.VratiSve();
```

Уговор УГ6: ВратиНаруџбине

Операција: ВратиНаруџбине(КорисничкоИме, List<Наруџбина>)

Веа са СК: СК8, СК5

Предуслов: /

Постуслов: /

```
public List<Narudzbin> VratiSve(string KorisnickoIme) =>
jedinicaRada.NarudzbinRepozitorijum.PretragaKupac(n =>
n.Kupac.UserName.ToLower().Trim() == KorisnickoIme.ToLower().Trim());
```

Уговор УГ7: ВратиПаковања

Операција: ВратиПаковања(List<Паковање>)

Веа са СК: СК6

Предуслов: /

Постуслов: /

```
public List<Pakovanje> VratiPakovanja() =>
jedinicaRada.PakovanjeRepozitorijum.VratiSve();
```

Уговор УГ8: ВратиКупца

Операција: ВратиКупца(КорисничкоИме, Лозинка)

Веа са СК: СК9

Предуслов: /

Постуслов: /

```
public Kupac Pretraga(string korisnickoIme, string lozinka) =>
jedinicaRada.KupacRepozitorijum.PretragaKorisnickoImeLozinka(korisnickoIme,
lozinka);
```

Уговор УГ9: ВратиПроизводе

Операција: ВратиПроизводе(List<Производ>)

Веа са СК: СК13

Предуслов: /

Постуслов: /

```
public List<Proizvod> VratiSve() => jedinicaRada.ProizvodRepozitorijum.VratiSve();
```

Уговор УГ10: ВратиПродајнаМеста

Операција: ВратиПродајнаМеста(List<ПродајноМесто>)

Веа са СК: СК13

Предуслов: /

Постуслов: /

```
public List<ProdajnoMesto> VratiPrMesta() =>
jedinicaRada.ProdajnoMestoRepozitorijum.VratiSve().ToList();
```

Уговор УГ11: ИтзмениНаруџбину

Операција: ИзмениНаруџбину(Наруџбина)

Веа са СК: СК5

Предуслов: Вредносна и структурна ограничења над објектом Наруџбина морају бити задовољена

Постуслов: Подаци о наруџбини су ажурирани

```
public string Izmeni(int id, int statusIsporuke)
{
    Narudzbina narudzbina = new Narudzbina();
    narudzbina = jedinicaRada.NarudzbinaRepozitorijum.PretragaId(id);
    narudzbina.StatusIsporuke = (Domen.StatusIsporuke)statusIsporuke;
    jedinicaRada.Sacuvaj();
    return narudzbina.StatusIsporuke.ToString();
}
```

Уговор УГ12: ИтзмениКупца

Операција: ИзмениКупца(Купац)

Веа са СК: СК9

Предуслов: Вредносна и структурна ограничења над објектом Купац морају бити задовољена

Постуслов: Подаци о купцу су ажурирани

```
public async Task<IdentityResult> Azuriraj(Osoba kupac)
{
    IdentityResult rezultat = await manager.UpdateAsync(kupac);
    return rezultat;
}
```

Уговор УГ13: КреирајЦветниАранжман

Операција: КреирајЦветниАранжман(ЦветниАранжман)

Веа са СК: СК6

Предуслов: Вредносна и структурна ограничења над објектом ЦветниАранжман морају бити задовољена

Постуслов: Подаци о цветном аранжману су креирани

```
public void Kreiraj(CvetniAranzman cvetniAranzman)
{
    jedinicaRada.CvetniAranzmanRepozitorijum.Dodaj(cvetniAranzman);
    jedinicaRada.Sacuvaj();
}
```

Уговор УГ14: КреирајКупца

Операција: КреирајКупца(Купац)

Веа са СК: СК7

Предуслов: Вредносна и структурна ограничења над објектом Купац морају бити задовољена

Постуслов: Подаци о купцу су креирани

```
public async Task<IdentityResult> Kreiraj(Kupac kupac, string lozinka)
{
    IdentityResult rezultat = await manager.CreateAsync(kupac, lozinka);
    return rezultat;
}
```

Уговор УГ15: КреирајЦвет

Операција: КреирајЦвет(Цвет)

Веа са СК: СК10

Предуслов: Вредносна и структурна ограничења над објектом Цвет морају бити задовољена

Постуслов: Подаци о цвету су креирани

```
public void Kreiraj(Cvet cvet)
{
    jedinicaRada.CvetRepozitorijum.Dodaj(cvet);
    jedinicaRada.Sacuvaj();
}
```

Уговор УГ16: КреирајПаковање

Операција: КреирајПаковање(Паковање)

Веза са СК: СК11

Предуслов: Вредносна и структурна ограничења над објектом Паковање морају бити задовољена

Постуслов: Подаци о паковању су креирани

```
public void Kreiraj(Pakovanje pakovanje)
{
    jedinicaRada.PakovanjeRepozitorijum.Dodaj(pakovanje);
    jedinicaRada.Sacuvaj();
}
```

Уговор УГ17: КреирајНаруџбину

Операција: КреирајНаруџбину(Наруџбина)

Веза са СК: СК13

Предуслов: Вредносна и структурна ограничења над објектом Наруџбина морају бити задовољена

Постуслов: Подаци о наруџбини су креирани

```
public void Kreiraj(Narudzina narudzina)
{
    jedinicaRada.KupacRepozitorijum.PretragaId(narudzina.KupacId).BrojNarudzina++;
    jedinicaRada.NarudzinaRepozitorijum.Dodaj(narudzina);
    jedinicaRada.Sacuvaj();
}
```

Уговор УГ18: ОбришиНаруџбину

Операција: ОбришиНаруџбину(Наруџбина)

Веза са СК: СК8

Предуслов: Структурна ограничења над објектом Наруџбина морају бити задовољена

Постуслов: Подаци о наруџбини су обрисани

```
public void Ozbrisi(int id)
{
    jedinicaRada.NarudzinaRepozitorijum.BrisiId(id);
    jedinicaRada.Sacuvaj();
}
```

Уговор УГ19: Пријава

Операција: Пријава(КорисничкоИме,Лозинка)

Вега са СК: СК4

Предуслов: Морају бити задовољена структурна и вредносна ограничења над објектом Купац/Администратор

Постуслов: Купац/Администратор је пријављен на система.

```
public async Task<SignInResult> Prijava(string korisnickoIme, string lozinka)
{
    var rezultat = await signIn.PasswordSignInAsync(korisnickoIme,
                                                    lozinka, false, false);
    return rezultat;
}
```

Уговор УГ20: Одјава

Операција: Одјава()

Вега са СК: СК2

Предуслов: Морају бити задовољена структурна и вредносна ограничења над објектом Купац/Администратор

Постуслов: Купац/Администратор је одјављен са система.

```
public async Task<IActionResult> Odjava()
{
    await signIn.SignOutAsync();
    return RedirectToAction("Index", "Pocetna");
}
```


7.3 Имплементација слоја приступа подацима

Слој приступа подацима имплементиран је помоћу Entity Framework Core-а и применом Unit of Work и Repository патерна.

7.3.1 Класа Context

```
public class CvecaraContext : IdentityDbContext<Osoba, IdentityRole<int>, int>
{
    public CvecaraContext([NotNull] DbContextOptions options) :
base(options)
    {
    }

    public DbSet<Cvet> Cvet { get; set; }
    public DbSet<CvetniAranzman> CvetniAranzman { get; set; }
    public DbSet<Kupac> Kupac { get; set; }
    public DbSet<Narudzbina> Narudzbina { get; set; }
    public DbSet<Pakovanje> Pakovanje { get; set; }
    public DbSet<ProdajnoMesto> ProdajnoMesto { get; set; }
    public DbSet<Proizvod> Proizvod { get; set; }
    public DbSet<Stavka> Stavka { get; set; }
    public DbSet<Dekoracija> Dekoracija { get; set; }
    public DbSet<Osoba> Osoba { get; set; }
    public DbSet<Administrator> Administrator { get; set; }
    protected override void OnConfiguring(DbContextOptionsBuilder
optionsBuilder)
    {
        optionsBuilder.LogTo(Console.WriteLine);
        optionsBuilder.EnableSensitiveDataLogging(true);
    }
    protected override void OnModelCreating(ModelBuilder modelBuilder)
    {
        modelBuilder.Entity<Narudzbina>().HasOne(n =>
n.Kupac).WithMany().onDelete(DeleteBehavior.NoAction);
        modelBuilder.Entity<Dekoracija>().HasKey(d => new {
d.CvetProizvodId, d.CvetniAranzmanProizvodId });
        modelBuilder.Entity<Dekoracija>().HasOne(d =>
d.Cvet).WithMany().HasForeignKey(d =>
d.CvetProizvodId).onDelete(DeleteBehavior.NoAction);
        modelBuilder.Entity<Dekoracija>().HasOne(d =>
d.CvetniAranzman).WithMany(cv => cv.Dekoracije).HasForeignKey(d =>
d.CvetniAranzmanProizvodId).onDelete(DeleteBehavior.NoAction);

modelBuilder.Entity<Cvet>().HasBaseType<Proizvod>().ToTable("Cvet");
```

```

modelBuilder.Entity<CvetniAranzman>().HasBaseType<Proizvod>().ToTable("CvetniA
ranzman");

modelBuilder.Entity<Kupac>().HasBaseType<Osoba>().ToTable("Kupac");

modelBuilder.Entity<Administrator>().HasBaseType<Osoba>().ToTable("Administrat
or");

        modelBuilder.Entity<Narudzbina>().OwnsMany(n =>
n.Stavke).WithOwner(s => s.Narudzbina);

                modelBuilder.Entity<CvetniAranzman>().HasOne(ca =>
ca.Pakovanje).WithMany().onDelete(DeleteBehavior.NoAction);
                modelBuilder.Entity<Narudzbina>().HasOne(n =>
n.ProdajnoMesto).WithMany().IsRequired(false).onDelete(DeleteBehavior.NoAction
);
                modelBuilder.Entity<CvetniAranzman>().HasOne(ca =>
ca.Kupac).WithMany().onDelete(DeleteBehavior.NoAction);

                base.OnModelCreating(modelBuilder);
        }
}

```

7.3.2 Sloj pristupa podacima

Unit of work

```

public interface IJedinicaRada
{
    public ICvetniAranzmanRepozitorijum CvetniAranzmanRepozitorijum {
get;set;}
    public IKupacRepozitorijum KupacRepozitorijum { get; set; }
    public IPakovanjeRepozitorijum PakovanjeRepozitorijum { get; set; }
    public ICvetRepozitorijum CvetRepozitorijum { get; set; }
    public IProdajnoMestoRepozitorijum ProdajnoMestoRepozitorijum { get;
set;}
    public IDEkoracijaRepozitorijum DekoracijaRepozitorijum { get; set; }
    public IProizvodRepozitorijum ProizvodRepozitorijum { get; set; }
    public INarudzbinaRepozitorijum NarudzbinaRepozitorijum { get; set; }
    public IOsobaRepozitorijum OsobaRepozitorijum { get; set; }
    public void Sacuvaj();
}

public class JedinicaRada : IJedinicaRada
{
    private readonly CvecaraContext context;
}

```

```

public JedinicaRada(CvecaraContext context)
{
    this.context = context;
    CvetniAranzmanRepozitorijum = new
CvetniAranzmanRepozitorijum(context);
    CvetRepozitorijum = new CvetRepozitorijum(context);
    KupacRepozitorijum = new KupacRepozitorijum(context);
    ProdajnoMestoRepozitorijum = new
ProdajnoMestoRepozitorijum(context);
    CvetRepozitorijum = new CvetRepozitorijum(context);
    PakovanjeRepozitorijum = new PakovanjeRepozitorijum(context);
    DekoracijaRepozitorijum = new DekoracijaRepozitorijum(context);
    ProizvodRepozitorijum = new ProizvodRepozitorijum(context);
    NarudzbinaRepozitorijum = new NarudzbinaRepozitorijum(context);
    OsobaRepozitorijum = new OsobaRepozitorijum(context);
}
public ICvetniAranzmanRepozitorijum CvetniAranzmanRepozitorijum { get;
set; }
public IKupacRepozitorijum KupacRepozitorijum { get; set; }
public IPakovanjeRepozitorijum PakovanjeRepozitorijum { get; set; }
public ICvetRepozitorijum CvetRepozitorijum { get; set; }
public IProdajnoMestoRepozitorijum ProdajnoMestoRepozitorijum { get;
set; }
public IDEkoracijaRepozitorijum DekoracijaRepozitorijum { get; set; }
public IProizvodRepozitorijum ProizvodRepozitorijum { get; set; }
public INarudzbinaRepozitorijum NarudzbinaRepozitorijum { get; set; }
public IOSobaRepozitorijum OsobaRepozitorijum { get; set; }

public void Sacuvaj()
{
    context.SaveChanges();
}
}

```

IRepository

```

public interface IRepozitorijum<TEntitet> where TEntitet : class
{
    public void Dodaj(TEntitet entitet );
    public void Brisi(TEntitet entitet);
    public List<TEntitet> Pretraga(Expression<Func<TEntitet, bool>>
uslov);
    public List<TEntitet> VratiSve();
    public void Azuriraj(TEntitet entitet);
}

```

7.5 Имплементација презентационог слоја

Пример модела:

```
public class RegistracijaViewModel
{
    [Required(ErrorMessage = "**")]
    public string Ime { get; set; }
    [Required(ErrorMessage = "**")]
    public string Prezime { get; set; }
    [Required(ErrorMessage = "**")]
    public string KorisnickoIme { get; set; }
    [Required(ErrorMessage = "**")]
    [Phone(ErrorMessage = "Samo brojevi")]
    public string BrojTelefona { get; set; }
    [Required(ErrorMessage = "**")]
    [MinLength(7, ErrorMessage = "Minimum 7 karaktera")]
    public string Lozinka { get; set; }
    [Required(ErrorMessage = "**")]
    [Compare("Lozinka", ErrorMessage = "Greška")]
    public string LozinkaProvera { get; set; }
    [Required(ErrorMessage = "**")]
    [EmailAddress(ErrorMessage = "**")]
    public string Email { get; set; }
}
```

Пример контролера:

```
public class CvetController : Controller
{
    private readonly IJedinicaRada jedinicaRada;
    private readonly IWebHostEnvironment webHost;

    public CvetController(IJedinicaRada jedinicaRada, IWebHostEnvironment
webHost)
    {
        this.jedinicaRada = jedinicaRada;
        this.webHost = webHost;
    }
    [Authorize(Roles = "Kupac")]
    public IActionResult Index(int id)
    {
        Cvet cvet = new Cvet();
        try
        {
            cvet = jedinicaRada.CvetRepozitorijum.PretragaId(id);
        }
        catch
        {
            return RedirectToAction("Greska", "Autentifikacija");
        }
    }
}
```

```

    CvetViewModel cvetView = new CvetViewModel()
    {
        Boja = cvet.Boja,
        Cena = cvet.Cena,
        Familija = cvet.Familija,
        Naziv = cvet.Naziv,
        Opis = cvet.Opis,
        Slika = cvet.Slika
    };
    return PartialView(cvetView);
}
[HttpGet]
[Authorize(Roles = "Administrator")]
public IActionResult Create()
{
    return View();
}
[HttpPost]
[Authorize(Roles = "Administrator")]
public IActionResult Create(CvetViewModel model)
{
    if (!ModelState.IsValid)
    {
        return Create();
    }

    string wwwRootPath = webHost.WebRootPath;
    string fileName =
Path.GetFileNameWithoutExtension(model.SlikaFile.FileName);
    string extension = Path.GetExtension(model.SlikaFile.FileName);
    model.Slika = fileName = fileName +
DateTime.Now.ToString("yymmssfff") + extension;
    string path = Path.Combine(wwwRootPath + "/slike/", fileName);
    using (var fileStream = new FileStream(path, FileMode.Create))
    {
        model.SlikaFile.CopyTo(fileStream);
    }

    Cvet cvet = new Cvet()
    {
        Boja = model.Boja,
        Opis = model.Opis,
        Naziv = model.Naziv,
        Cena = model.Cena,
        Familija = model.Familija,
        Slika = model.Slika
    };
    try
    {
        jedinicarada.CvetRepozitorijum.Dodaj(cvet);
        jedinicarada.Sacuvaj();
    }
    catch

```

```

{
    return RedirectToAction("Greska", "Autentifikacija");
}

return RedirectToAction("AdministratorProfil", "Pocetna");
}}

```

7.5 Имплементација складишта података

На основу софтверских класа структуре пројектовали смо табеле (складишта података) релационог система за управљање базом података [1].

	Name	Data Type	Allow Nulls	Default
PK	Stavkald	int	<input type="checkbox"/>	
	Narudzbinald	int	<input type="checkbox"/>	
	ProizvodId	int	<input type="checkbox"/>	
	Kolicina	int	<input type="checkbox"/>	
			<input type="checkbox"/>	

Слика 170 Табела Ставка

	Name	Data Type	Allow Nulls	Default
PK	ProizvodId	int	<input type="checkbox"/>	
	Naziv	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	Cena	decimal(18,2)	<input type="checkbox"/>	
			<input type="checkbox"/>	

Слика 171 Табела Производ

	Name	Data Type	Allow Nulls	Default
PK	ProdajnoMestold	int	<input type="checkbox"/>	
	Adresa	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	RadnoVreme	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	Sajt	nvarchar(MAX)	<input checked="" type="checkbox"/>	
			<input type="checkbox"/>	

Слика 172 Табела ПродајноМесто

	Name	Data Type	Allow Nulls	Default
PK	Pakovanjeld	int	<input type="checkbox"/>	
	Naziv	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	Materijal	int	<input type="checkbox"/>	
	Oblik	int	<input type="checkbox"/>	
	Boja	int	<input type="checkbox"/>	
	Cena	decimal(18,2)	<input type="checkbox"/>	
	Slika	nvarchar(MAX)	<input checked="" type="checkbox"/>	
			<input type="checkbox"/>	

Слика 173 Табела Паковање

	Name	Data Type	Allow Nulls	Default
PK	Narudzbinald	int	<input type="checkbox"/>	
	VrstaNarudzbine	int	<input type="checkbox"/>	
	Adresa	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	StatusIsporuke	int	<input type="checkbox"/>	
	ProdajnoMestold	int	<input checked="" type="checkbox"/>	
	KupaclId	int	<input type="checkbox"/>	
			<input type="checkbox"/>	

Слика 174 Табела Наруџбина

	Name	Data Type	Allow Nulls	Default
PK	Id	int	<input type="checkbox"/>	
	BrojNarudzbina	int	<input type="checkbox"/>	((0))
	DatumKreiranjaNaloga	datetime2(7)	<input type="checkbox"/>	('0001-01-01T00:00:00.0000000')
			<input type="checkbox"/>	

Слика 175 Табела Купац

	Name	Data Type	Allow Nulls	Default
PK	CvetProizvodId	int	<input type="checkbox"/>	
PK	CvetniAranzmanProizvodId	int	<input type="checkbox"/>	
	BrojCvetova	int	<input type="checkbox"/>	
			<input type="checkbox"/>	

Слика 176 Табела Декорација

	Name	Data Type	Allow Nulls	Default
PK	ProizvodId	int	<input type="checkbox"/>	
	Sjaj	int	<input type="checkbox"/>	
	Masna	bit	<input type="checkbox"/>	
	Namena	int	<input type="checkbox"/>	
	Napomena	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	Pakovanjeld	int	<input type="checkbox"/>	
	KupaclId	int	<input type="checkbox"/>	
			<input type="checkbox"/>	


Слика 177 Табела ЦветниАранжман

	Name	Data Type	Allow Nulls	Default
PK	ProizvodId	int	<input type="checkbox"/>	
	Opis	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	Familija	int	<input type="checkbox"/>	
	Slika	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	Boja	int	<input type="checkbox"/>	
			<input type="checkbox"/>	

Слика 178 Табела Цвет

	Name	Data Type	Allow Nulls	Default
PK	Id	int	<input type="checkbox"/>	
	DatumZaposlenja	datetime2(7)	<input type="checkbox"/>	
	SifraZaposlenog	nvarchar(MAX)	<input checked="" type="checkbox"/>	
			<input type="checkbox"/>	

Слика 179 Табела Администратор

	Name	Data Type	Allow Nulls	Default
	Id	int	<input type="checkbox"/>	
	Ime	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	Prezime	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	UserName	nvarchar(256)	<input checked="" type="checkbox"/>	
	NormalizedUserName	nvarchar(256)	<input checked="" type="checkbox"/>	
	Email	nvarchar(256)	<input checked="" type="checkbox"/>	
	NormalizedEmail	nvarchar(256)	<input checked="" type="checkbox"/>	
	EmailConfirmed	bit	<input type="checkbox"/>	
	PasswordHash	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	SecurityStamp	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	ConcurrencyStamp	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	PhoneNumber	nvarchar(MAX)	<input checked="" type="checkbox"/>	
	PhoneNumberConfirmed	bit	<input type="checkbox"/>	
	TwoFactorEnabled	bit	<input type="checkbox"/>	
	LockoutEnd	datetimeoffset(7)	<input checked="" type="checkbox"/>	
	LockoutEnabled	bit	<input type="checkbox"/>	
	AccessFailedCount	int	<input type="checkbox"/>	
			<input type="checkbox"/>	

Слика 180 Табела Особа

8. ТЕСТИРАЊЕ

Извршено је ручно тестирање апликације. Њеним покретањем и уносом неодговарајућих и неисправних података, вршене су провере исправности имплементираних валидација. Такође, уношени су и исправни подаци, како би се тестирали сви приказани случајеви коришћења и утврдили њихови успешни исходи. Сви недостаци откривени приликом тестирања су успешно исправљени

9. ЗАКЉУЧАК

У овом раду приказана је целовита слика процеса изградње једне веб апликације коришћењем ASP.NET Core оквира. Након упознавања са .NET платформом и њеним компонентама, радним оквиром ASP.NET Core, Entity Framework Core-ом, MVC патерном, JavaScript-ом и Bootstrap-ом, објашњене су фазе развоја софтверског система применом упрошћене Ларманове методе на студијском примеру апликације за део пословања цвећаре. Апликација имплементира MVC патерн због брзог и једноставног програмирања, као и велике флексибилности. Ради постизања што бољег корисничког искуства за дизајн су употребљени Bootstrap, који садржи велики број готових стилова, што такође омогућава олакшице приликом програмирања, али и JavaScript заједно са својом библиотеком jQuery. Захваљујући објектно-релационом маперу, Entity Framework Core-у, комуникација са базом била је крајње поједностављена, што значајно подиже продуктивност приликом програмирања.

Функционалности које би било добро додати у дањем развоју апликације су: понуда декорисања ентеријера за различите догађаје и прославе, могућност онлине плаћања, вођење евиденције о лојалности корисника кроз праћење броја, количине и цене наруџбина због додавања могућности персонализованих попушта, боља заштита система. Обзиром колико је фаза тестирања значајна, наредни корак свакако би био израда UNIT тестова. Такође, уместо чистог JavaScript-а, пожељна би била употреба неког JavaScript радног оквира, попут React-а, Vue.js-а, Node.js-а и слично.

За одабрне технологије одлучила сам се мотивисана градивом и стеченим знањем на предмету Напредне .NET технологије. У сврху израде ове веб апликације искористила сам и знање стечено на предмету Пројектовање софтвера, као и на многим другим предметима на Факултету организационих наука, где планирам да наставим своје даље формално образовање и усавршавање.

10. ЛИТЕРАТУРА

- [1] Dr Vlajic, S. (2020). Projektovanje softvera (skripta), FON, Beograd, 2020
- [2] Mark, P. (2022.). C# 10 i .NET 6 Moderno međuplatformsko programiranje
- [3] Gabriel and Francesco (2021). C# 9 i .NET 5 arhitektura softvera
- [4] Entity Framework Core. (2021, мај 25). Преузето са <https://docs.microsoft.com/en-us/ef/core/> [Последњи приступ 20.9.2022]
- [5] Smith, S. (2022, јун 27). Overview of ASP.NET Core MVC. Преузето са <https://learn.microsoft.com/en-us/aspnet/core/mvc/overview?view=aspnetcore-5.0> [Последњи приступ 20.9.2022]
- [6] Rich, L. (2019, мај 6). Introducing .NET 5. Преузето са <https://devblogs.microsoft.com/dotnet/introducing-net-5/> [Последњи приступ 20.9.2022]
- [7] C# 10 and .NET 6 – Modern Cross-Platform Development. (2022, септембар 12). Преузето са <https://www.amazon.com/10-NET-Cross-Platform-Development-websites/dp/1801077363> [Последњи приступ 20.9.2022]
- [8] MVC Architecture – Detailed Explanation. (2022, мај 30). Преузето са <https://www.interviewbit.com/blog/mvc-architecture/> [Последњи приступ 20.9.2022]
- [9] Entity Framework Core. (n.d.). Преузето са <https://www.educba.com/entity-framework-core/> [Последњи приступ 20.9.2022]
- [10] ASP.NET MVC Life Cycle. (2022, март 22). Преузето са <https://www.geeksforgeeks.org/asp-net-mvc-life-cycle/> [Последњи приступ 20.9.2022]
- [11] An Introduction to JavaScript. (2022, август 8). Преузето са <https://javascript.info/intro> [Последњи приступ 20.9.2022]
- [12] Pandya, J. (2020, новембар 4). What Is jQuery Used For? Преузето са <https://learningjquery.com/2020/11/what-is-jquery-used-for> [Последњи приступ 20.9.2022]

[13] What is Ajax? (n.d.). Преузето са https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX/Getting_Started [Последњи приступ 20.9.2022]

[14] Andrew, Z. (n.d.). Преузето са <https://www.techtarget.com/whatis/definition/bootstrap> [Последњи приступ 20.9.2022]