

**CAPSTONE PROJECT REPORT**

**Report 3 – System Requirement Specification**

**Table of Contents**

[**III. Software Requirement Specification**](#_5qz5zmocojv) **9**

[1. Overall Description](#_1t3h5sf) 9

[1.1 Product Overview](#_2s8eyo1) 9

[1.2 Business Rules](#_4l1mx3s146j8) 9

[Table 1. Business Rules](#_32hioqz) 9

[2. User Requirements](#_1ksv4uv) 12

[2.1 Overview](#_3fwokq0) 12

[2.1.1. Use Case Diagram](#_1egqt2p) 12

[Figure 1. Use case diagram](#_odc9jc) 13

[2.1.2. System Actors](#_1pgrrkc) 13

[Table 2. System actors](#_49gfa85) 13

[2.1.3. Use Cases List](#_38czs75) 13

[Table 3. Use Cases List](#_44sinio) 13

[2.2. Use case specification](#_1nia2ey) 16

[2.2.1 UC01 – Create manager](#_47hxl2r) 16

[Figure 2. <Use case> Create manager](#_2eclud0) 16

[Table 4. UC01 - Create manager](#_147n2zr) 16

[2.2.2 UC02 – Get list manager](#_xhenwjud7npo) 18

[Figure 3. <Use case> Get list manager](#_2eclud0) 18

[Table 5. UC02 - Get list manager](#_147n2zr) 18

[2.2.3 UC03 – Enable/ disable manager’s account](#_mfkourc064hd) 19

[Figure 4. <Use case> Enable/disable manager’s account](#_q4z2l97l987l) 19

[Table 6. UC03 - Enable/disable manager’s account](#_imvotduei0u6) 19

[2.2.4 UC04 – View user information](#_f85gofriddd7) 20

[Figure 5. <Use case> View user information](#_37rc1to39eqf) 20

[Table 7. UC04 - View user information](#_5a33shkm3nfs) 20

[2.2.5 UC05 – Ban/Unban user](#_5a3mk965h0uy) 21

[Figure 6. <Use case> Ban/unban user](#_5scorzq7dfem) 21

[Table 8. UC05 - Ban/unban user](#_w514ws7kxp0w) 21

[2.2.6 UC06 – Create country’s tag](#_ltqshgeuk3hg) 22

[Figure 7. <Use case> Create country’s tag](#_hpk0yc44mzbd) 22

[Table 9. UC06 - Create country’s tag](#_mwhdbdoxv49) 22

[2.2.7 UC07 – Get list country’s tag](#_iyyx4z4fn9ei) 24

[Figure 8. <Use case> Get list country’s tag](#_agashee4kjzo) 24

[Table 10. UC07 - Get list country’s tag](#_kdd51uyl0gzv) 24

[2.2.8 UC08 – Create cooking method](#_8bdweus9dh5) 25

[Figure 9. <Use case> Create cooking method](#_y1tox71f37zm) 25

[Table 11. UC08 - Create cooking method](#_egs0l5n0mg05) 25

[2.2.9 UC09 – Get list cooking method](#_bwpk9g499e63) 26

[Figure 10. <Use case> Get list cooking method](#_q511hr1imr45) 26

[Table 12. UC09 - Get list cooking method](#_p0vgq4xhlmb6) 26

[2.2.10 UC10 – Disable cooking method](#_z2avnl3drcho) 28

[Figure 11. <Use case> Disable cooking method](#_mguux452521r) 28

[Table 13. UC10 - Disable cooking method](#_69x6famlmi9x) 28

[2.2.11 UC11 – Create ingredient](#_jkmd6aufwsvr) 29

[Figure 12. <Use case> Create ingredient](#_3ubu8wocpwdb) 29

[Table 14. UC11 - Create ingredient](#_m4eb14fkq3q2) 29

[2.2.12 UC12 – Get list ingredient](#_ay81ooccs16k) 30

[Figure 13. <Use case> Get list ingredient](#_fkhci3xrxvh4) 30

[Table 15. UC12 - Get list ingredient](#_7jerien67atj) 30

[2.2.13 UC13 – Update ingredient](#_g2nhwutq4vh8) 32

[Figure 14. <Use case>Update Ingredient](#_10g5z6wtxgu6) 32

[Table 16. UC13 - Update ingredient](#_9en00rjabqqf) 32

[2.2.14 UC14 – Disable ingredient](#_4ba0x9qiomga) 33

[Figure 15. <Use case> Disable ingredient](#_822indyrib85) 33

[Table 17. UC14 - Disable ingredient](#_l4x25nrskhog) 33

[2.2.15 UC15 – Create recipe](#_wt4ve3hd4gxv) 34

[Figure 16. <Use case> Create recipe](#_u9ip6c9r4m2b) 34

[Table 17. UC15 - Create recipe](#_881hp0ltlp6n) 34

[2.2.16 UC16 – Get list recipe](#_l189hb1yxm4t) 36

[Figure 17. <Use case> Get list recipe](#_1grgtl89vf99) 36

[Table 19. UC16 - Get list recipe](#_p882da9q159) 36

[2.2.17 UC17 – Update recipe](#_7lneg49do2ro) 37

[Figure 18. <Use case>Update recipe](#_6uv4u56p9dvg) 37

[Table 20. UC17 - Update recipe](#_30vgj145ruh2) 37

[2.2.18 UC18 – Disable recipe](#_3hj4ll6hd2eq) 39

[Figure 19. <Use case> Disable recipe](#_f8cbog1aghoy) 39

[Table 20. UC18 - Disable recipe](#_t4ng0f3vgsnv) 39

[2.2.19 UC19 – Create categories of recipe](#_1gab1os3bp1h) 40

[Figure 20. <Use case> Create categories of recipe](#_f0fl69g4zdh2) 40

[Table 22. UC19 - Create categories of recipe](#_ylnnh7a1n3g9) 40

[2.2.20 UC20 – Get list categories of recipe](#_1epxkw31dpxr) 41

[Figure 21. <Use case> Get list categories of recipe](#_k67wfolz8rg1) 41

[Table 23. UC20 - Get list categories of recipe](#_375bhecglc1n) 41

[2.2.21 UC21 – Update categories of recipe](#_ppxlx3akdxtw) 42

[Figure 22. <Use case>Update categories of recipe](#_2ap1njfx2n5r) 43

[Table 24. UC21 - Update categories of recipe](#_wz4ctxns363p) 43

[2.2.22 UC22 – Disable categories of recipe](#_dy2zaoeabm9z) 44

[Figure 23. <Use case> Disable categories of recipe](#_qfb7ba4ysjjz) 44

[Table 25. UC22 - Disable categories of recipe](#_7bkg5v7t9lpn) 44

[2.2.23 UC23 – Create categories of ingredient](#_bltiske4a788) 45

[Figure 24. <Use case> Create categories of ingredient](#_5zw18kp1ad19) 45

[Table 26. UC23 - Create categories of ingredient](#_eh3afh8t7ymv) 45

[2.2.24 UC24 – Get list categories of ingredient](#_2i5tsj1koajn) 46

[Figure 25. <Use case> Get list categories of ingredient](#_2uopknos174p) 46

[Table 27. UC24 - Get list categories of ingredient](#_eqa1meursw6p) 46

[2.2.25 UC25 – Update categories of ingredient](#_m1uxjn1odqrf) 48

[Figure 26. <Use case>Update categories of ingredient](#_dbhrbc77ry5o) 48

[Table 28. UC25 - Update categories of ingredient](#_q5loya3z1rhc) 48

[2.2.26 UC26 – Disable categories of ingredient](#_6iuos5z6repn) 49

[Figure 27. <Use case> Disable categories of ingredient](#_3zpvw4lg5i1a) 49

[Table 29. UC26 - Disable categories of ingredient](#_mw9efto9fgx0) 49

[2.2.27 UC27 – Create notification](#_gi0b0vygs04z) 50

[Figure 28. <Use case> Create notification](#_31lnfl3od2j5) 50

[Table 30. UC27 - Create notification](#_4lcuvox3tu0j) 50

[2.2.28 UC28 – Get list notification](#_7mye36r5lyxd) 52

[Figure 29. <Use case> Get list notification](#_227kaoo4jlom) 52

[Table 31. UC28 - Get list notification](#_urpptwrupnj2) 52

[2.2.29 UC29 – Update notification](#_1gjcrpdsbqdj) 53

[Figure 30. <Use case>Update notification](#_kvvd6ts7h2hk) 53

[Table 32. UC29 - Update notification](#_d6axkpmo3zrk) 53

[2.2.30 UC30 – Disable notification](#_hzlb7q6cr7xl) 55

[Figure 31. <Use case> Disable notification](#_r1i6twwq3jky) 55

[Table 33. UC30 - Disable notification](#_kb5ha9xyfzhn) 55

[2.2.31 UC31 – Get list post](#_2uiunqiv2f7c) 56

[Figure 32. <Use case> Get list post](#_f655l76qb65t) 56

[Table 34. UC31 - Get list post](#_phef8013jbi6) 56

[2.2.32 UC32 – Approve post](#_pjaf6385pnhb) 57

[Figure 33. <Use case> Approve post](#_kj69ugwwv7jm) 57

[Table 35. UC32 - Approve post](#_s3s6h5fs0m7j) 57

[2.2.33 UC33 – Disable post](#_rmnu3jzfo4qq) 58

[Figure 34. <Use case> Disable post](#_x07n51ce4si7) 58

[Table 36. UC33 - Disable post](#_ejvauehpvvvg) 58

[2.2.34 UC34 – Get list report](#_z4074c9vzt1g) 60

[Figure 35. <Use case> Get list report](#_aisg3vcuumhp) 60

[Table 37. UC34 - Get list report](#_amkie9m3a1du) 60

[2.2.35 UC35 – Get list log](#_ny0szlx3q78) 61

[Figure 36. <Use case> Get list log](#_8fb7xb5pwkj1) 61

[Table 38. UC35 - Get list log](#_1pq4bnbiw2dv) 61

[2.2.36 UC36 – View user information](#_961701jujm7m) 62

[Figure 37. <Use case> View user information](#_xkk89zz658o2) 62

[Table 39. UC036 - View user information](#_8gnwp78j86ie) 62

[2.2.37 UC37 – Ban/Unban user](#_w5e542lsxiy9) 63

[Figure 38. <Use case> Ban/unban user](#_8psmdauwpbai) 63

[Table 40. U037 - Ban/unban user](#_gpgjc2xhgsw0) 63

[2.2.38 UC38 – Create categories of recipe](#_65rxj8646lq5) 64

[Figure 39. <Use case> Create categories of recipe](#_v3vc9fsx5570) 64

[Table 41. UC38 - Create categories of recipe](#_ld9ntc4s6rin) 64

[2.2.39 UC39 – Get list categories of recipe](#_pr5p2t9sjw35) 66

[Figure 40. <Use case> Get list categories of recipe](#_wda9gsd4wnrq) 66

[Table 42. UC39 - Get list categories of recipe](#_d815zmeuq9jh) 66

[2.2.40 UC40 – Update categories of recipe](#_xzmasohu16jp) 67

[Figure 41. <Use case>Update categories of recipe](#_wedhxkrzawn9) 67

[Table 43. UC40 - Update categories of recipe](#_fqir6zqhuz6p) 67

[2.2.41 UC41 – Disable categories of recipe](#_sdf0uq4z1cia) 69

[Figure 42. <Use case> Disable categories of recipe](#_er2nqzlt7ujh) 69

[Table 44. UC41 - Disable categories of recipe](#_uvx9w3fzcx2r) 69

[2.2.42 UC42 – Create categories of ingredient](#_a2y939k2jjsg) 70

[Figure 43. <Use case> Create categories of ingredient](#_fy0rj4puggab) 70

[Table 45. UC42 - Create categories of ingredient](#_pejvnpgwwva0) 70

[2.2.43 UC43 – Get list categories of ingredient](#_vr5v5wcedd4i) 71

[Figure 44. <Use case> Get list categories of ingredient](#_e6ok9tipk785) 71

[Table 46. UC43 - Get list categories of ingredient](#_lk15ohz96ndb) 71

[2.2.44 UC44 – Update categories of ingredient](#_ntyrasfffswm) 73

[Figure 45. <Use case>Update categories of ingredient](#_wy9xsj2v9m0q) 73

[Table 47. UC44 - Update categories of ingredient](#_jkdij2b9yfhg) 73

[2.2.45 UC45 – Disable categories of ingredient](#_aix9zsahtdcm) 74

[Figure 46. <Use case> Disable categories of ingredient](#_xa4gj4dt2r6h) 74

[Table 48. UC45 - Disable categories of ingredient](#_mdpmxjuxzrxt) 74

[2.2.46 UC46 – Get list post](#_5dbtkf1bks99) 75

[Figure 47. <Use case> Get list post](#_izv9p6qlzurr) 75

[Table 49. UC46 - Get list post](#_suivlxzelagm) 75

[2.2.47 UC47 – Approve post](#_pom0waylv7ho) 77

[Figure 48. <Use case> Approve post](#_jm5r5csq0yde) 77

[Table 50. UC47 - Approve post](#_cdfw4mhcby36) 77

[2.2.48 UC48 – Disable post](#_5v4azypx6j1z) 78

[Figure 49. <Use case> Disable post](#_32oktybtpmc) 78

[Table 51. UC48 - Disable post](#_wzawyvg3m5lj) 78

[2.2.49 UC49 – Get list report](#_s2bh7q1jrgij) 79

[Figure 50. <Use case> Get list report](#_jsoxwqmg6npo) 79

[Table 52. UC49 - Get list report](#_4zsodthv9a36) 79

[2.2.50 UC50 – View profile](#_4a8axxvbb7yv) 80

[Figure 51. <Use case> View profile](#_6tpg37msio39) 80

[Table 53. UC50 - View profile](#_l69apc6a445s) 80

[2.2.51 UC51 – Update profile](#_ack5v5pdcydm) 81

[Figure 52. <Use case> Update profile](#_64q47k5aabtb) 81

[Table 54. UC51 - Update profile](#_kn9g4jifubsm) 81

[2.2.52 UC52 – Change password](#_qqm59vucgdiw) 83

[Figure 53. <Use case> Change password](#_qiu9y0bsluc5) 83

[Table 55. UC52 - Change password](#_36t8jltb54s5) 83

[2.2.53 UC53 – Create Ingredient](#_n6c1onqro704) 84

[Figure 54. <Use case> Create ingredient](#_pzkyfzbqjfxg) 84

[Table 56. UC53 - Create ingredient](#_gb7fms6fiu1w) 84

[2.2.54 UC54 – Update Ingredient](#_c6054sl1t8g) 85

[Figure 55. <Use case> Update Ingredient](#_qhh1luoi0pxz) 85

[Table 57. UC54 - Update Ingredient](#_a60ke17gvqit) 85

[2.2.55 UC55 – Delete Ingredient](#_sopddjt4a1b9) 87

[Figure 56. <Use case> Delete Ingredient](#_nywzpp7zzcao) 87

[Table 58. UC55 - Delete Ingredient](#_t7oj09881v9w) 87

[2.2.56 UC56 – View Ingredient](#_tb95jm3moqlx) 88

[Figure 57. <Use case> View Ingredient](#_1ska9pcqq8x3) 88

[Table 59. UC56 - View Ingredient](#_36wtvi7y0mew) 88

[2.2.57 UC57 – Search recipe](#_muyolcdbhqu) 89

[Figure 58. <Use case> Search recipe](#_e4qf6rmzqqp) 89

[Table 60. UC57 - Search recipe](#_er1qz5nfzcxs) 89

[2.2.58 UC58 – View recipes](#_rkcyge5qw0mp) 90

[Figure 59. <Use case> View recipes](#_h19bo4z8gkiy) 90

[Table 61. UC58 - View recipes](#_78firxdao7m3) 90

[2.2.59 UC59 – Rating recipe](#_n5fhpvnzq7x1) 91

[Figure 60. <Use case> Rating recipe](#_n8f5o09av65x) 91

[Table 61. UC59 - Rating recipe](#_r4fx3twbc1fh) 91

[2.2.60 UC60 – Save recipe](#_3gylm9x4ptrm) 92

[Figure 61. <Use case> Save recipe](#_43nuxdjakvyo) 92

[Table 62. UC60 - Save recipe](#_wiar22r6qiz) 92

[2.2.61 UC61 – Report recipe](#_1ifpg2c5ed2e) 93

[Figure 62. <Use case> Report recipe](#_62hsme37bzcp) 93

[Table 63. UC61 - Report recipe](#_nf578qbn58q1) 93

[2.2.62 UC62 – React recipe](#_cv5aedcmk4qt) 94

[Figure 63. <Use case> React recipe](#_mtyiht4pxtxm) 94

[Table 64. UC62 - React recipe](#_q7qqpkg8hzb0) 94

[2.2.63 UC63 – Create cooking plan](#_gkifwojxiebr) 95

[Figure 64. <Use case> Create cooking plan](#_um1dpf91gebk) 95

[Table 66. UC63 - Create cooking plan](#_spfmj5xxoxer) 95

[2.2.64 UC64 – Update cooking plan](#_1mb999i43wl0) 97

[Figure 65. <Use case> Update cooking plan](#_1bc8hupnc38r) 97

[Table 67. UC64 - Update cooking plan](#_ybm8ckylu772) 97

[2.2.65 UC65 – Delete cooking plan](#_rv7j81g9jb44) 98

[Figure 66. <Use case> Delete cooking plan](#_yrgk7ic76p9q) 98

[Table 68. UC65 - Delete cooking plan](#_9bp94c7twfas) 98

[2.2.66 UC66 – View cooking plan](#_nhz2iqr9cm3c) 99

[Figure 67. <Use case> View cooking plan](#_2kzcots1jrd0) 99

[Table 69. UC66 - View cooking plan](#_2xb5u7ifeebn) 99

[2.2.67 UC67 – Create post](#_obgy5hhu9b6f) 100

[Figure 68. <Use case> Create post](#_btbs8psrag94) 100

[Table 70. UC67 - Create post](#_jyjxjrrkz9fh) 100

[2.2.68 UC68 – Update post](#_pni5i9dx6q1y) 102

[Figure 69. <Use case> Update post](#_hhodgg3wfbwd) 102

[Table 71. UC68 - Update post](#_dmj9mqrgps8w) 102

[2.2.69 UC69 – Delete post](#_ybut9hyyq03o) 103

[Figure 71. <Use case> Delete post](#_b2k5aeke6gnv) 103

[Table 73 . UC69 - Delete post](#_xm42b0vgeyrc) 103

[2.2.70 UC70 – View post](#_lqto8nyg2j1t) 104

[Figure 71. <Use case> View post](#_azp50pdxrqjs) 104

[Table 73. UC70 - View post](#_w1k85rt85pyj) 104

[2.2.71 UC71 – React post](#_ls7v8ms2otox) 106

[Figure 72. <Use case> React post](#_4xeqquat0rxo) 106

[Table 74. UC71 - React post](#_ihldvdjyvszz) 106

[2.2.72 UC72 – Comment post](#_3wtb6bfyyj4j) 107

[Figure 73. <Use case> Comment post](#_8pafurac27sf) 107

[Table 75. UC72 - Comment post](#_ythee4xsiyej) 107

[2.2.73 UC73 – Report post](#_jrq6r8xs80ix) 108

[Figure 74. <Use case> Report post](#_jitqhv3rthkf) 108

[Table 76. UC73 - Report post](#_wknjsfgpdcci) 108

[2.2.74 UC74 – Save post](#_ammsqqiqqe3) 109

[Figure 75. <Use case> Save post](#_wfnxh0atbnpa) 109

[Table 77. UC74 - Save post](#_btcktfeef78m) 109

[2.2.75 UC75 – Report user](#_1r57dtn8lzu) 110

[Figure 76. <Use case> Report user](#_2y5g7murye9t) 110

[Table 78. UC75 - Report user](#_1kb0b59ut1zh) 110

[2.2.76 UC76 – Login](#_ldeupqdkv0y1) 112

[Figure 62. <Use case> Login](#_pt4yxfni9dpp) 112

[Table 63. UC76 - Login](#_ywa2kalj96vq) 112

[2.2.77 UC77 – Login Google](#_f9yupojit2nm) 113

[Figure 78. <Use case> Login Google](#_cfa6uh5f6920) 113

[Table 80. UC77 - Login Google](#_x5ed4ypeq7kk) 113

[2.2.78 UC78 – Register](#_kuduhi3chta2) 114

[Figure 79. <Use case> Register](#_4ftt8cla5vn4) 114

[Table 81. UC78 - Register](#_d4575hsgw44b) 114

[2.2.79 UC79 – Forgot password](#_23p7mjeg8jhf) 116

[Figure 79. <Use case> Forgot password](#_q2hykuvhl52x) 116

[Table 81. UC79 - Forgot password](#_99m7pleewxsf) 116

[2.2.80 UC80 – Send mail](#_1x9g7t8rn6yl) 117

[Figure 81. <Use case> Send mail](#_c10c1ibn435a) 117

[Table 83. UC80 - Send mail](#_2m1qitzhlxey) 117

[2.2.81 UC81 – Push notification](#_fhl96xyovg8y) 118

[Figure 82. <Use case> Push notification](#_3jlfz28ilhuf) 118

[Table 84. UC81 - Push notification](#_skh5dnjwt7p2) 118

[2.2.82 UC82 – Recipe suggestions for users](#_tyol9ftmy9zl) 119

[Figure 83. <Use case> Recipe suggestions for users](#_6hcwf1gjie6p) 119

[Table 85. UC82 - Recipe suggestions for users](#_6edl2dnufneo) 119

[2.2.83 UC83 – Arrange posts for managers to approve](#_8irfw2gzf74m) 120

[Figure 84. <Use case> Arrange posts for managers to approve](#_v3lob7ns3t2d) 120

[Table 86. UC83 - Arrange posts for managers to approve](#_rf6ot8jkdzdx) 120

[2.2.84 UC84 – Weekly delete temporal data](#_e9wvcehvq5t3) 121

[Figure 85. <Use case> Weekly delete temporal data](#_diao7obns062) 121

[Table 87. UC84 - Weekly delete temporal data](#_nq94p9e7pm2e) 121

[3. Functional Requirements](#_q6vdz1csv8s2) 122

[3.1 System Functional Overview](#_3gxvt7e) 122

[3.1.1. Screen Flow](#_1w363f7) 122

[Figure 86. Admin screen flow](#_tw23shplfj45) 123

[Figure 87. Manager screen flow](#_5sxi7cy62x65) 124

[Figure 88. User screen flow](#_upf7dh8y0gr) 125

[3.1.2. Screen Authorization](#_93481oxrmq) 126

[Table 88. Fooma Web Application screen authorization](#_ctkgsfleylld) 126

[Table 89. Fooma Mobile Application screen authorization](#_n5goa844s3w9) 127

[3.1.3. Non-Screen Functions](#_2xhct4mu9sal) 129

[Table 88. Non-screen authorization](#_h0ivoserf7op) 129

[3.1.4. Entity Relationship Diagram](#_l42hpzj7esa2) 129

[Figure 89. Entity Relationship diagram](#_4zh0v98lpkml) 129

[4. Non-Functional Requirements](#_2hzdiz78wl6j) 130

[4.1 External Interfaces](#_ts64ni) 130

[a. User Interfaces](#_mnkr6j7a5zt0) 130

[b. Software Interfaces](#_207lw27794aw) 130

[c. Communications Interfaces](#_hxa20pphzmoj) 130

[4.2 Quality Attributes](#_cycjkk11splz) 130

[a. Usability](#_t2ueoaehg0v9) 130

[b. Reliability](#_36n63amhrpfh) 130

[c. Performance](#_ksv7gd2wjax2) 130

[d. Dependability](#_6q0oe4en4b7f) 130

[d1. Security](#_sw02a5yn5lxu) 130

[d2. Safety](#_xu7k5qjkhd15) 130

[e. Supportability](#_lx6l5t8vb0op) 130

[f. Design Constraints](#_hmnwrapng3de) 131

[g. Support Documents](#_voi3vdpphloo) 131

[5. Other Requirements](#_fcw6kcn46dc) 131

[5.1 Messages List](#_vdliptxf1yo9) 131

[Table 89. Message list](#_sfub50tvjgfn) 131

[6. Algorithms](#_4i1bb8e) 131

# III. Software Requirement Specification

## 1. Overall Description

### 1.1 Product Overview

Fooma Food Management Application is an application that helps users manage the food in their refrigerator and also suggests recipes based on those ingredients. Users can schedule food cooking and share their recipes and results with the Fooma's community.

On Fooma's web platform, the admin can manage users, and managers, and post recipes and ingredients of the system. Managers can browse user's posts and recipes to post to the Fooma community.

### 1.2 Business Rules

###### Table 1. Business Rules

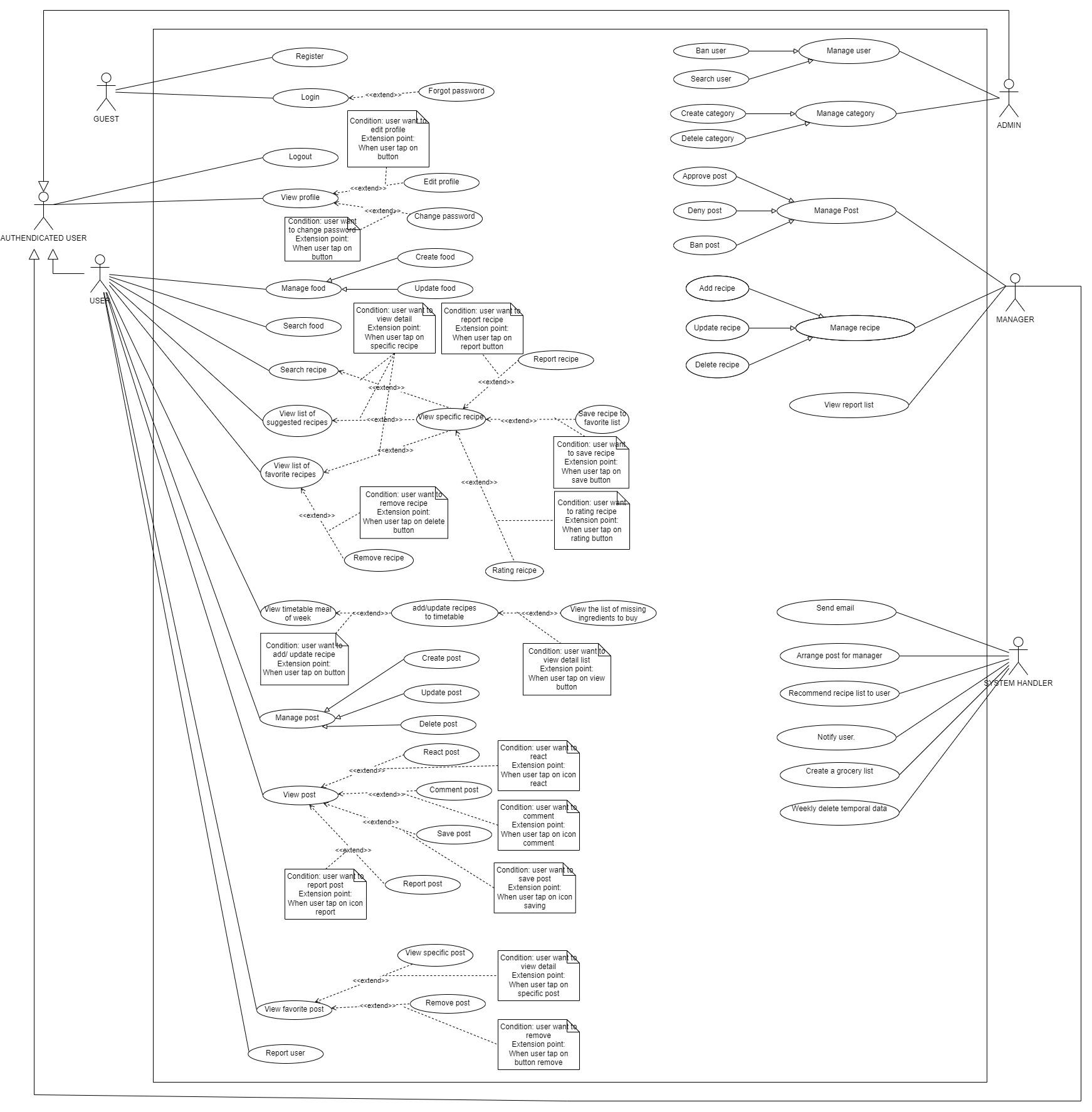
| **ID** | **Rule Definition** |
| --- | --- |
| BR-0001 | Users must log in to be able to perform the functions of the mobile application, web application. |
| BR-0002 | An authentication token for administrator and manager lasts for 60 minutes and for users lasts for 7 days. |
| BR-0003 | Expired token cannot be used to do any action with the system including login. |
| BR-0004 | Accounts with ban status (being ban by administrator) cannot login into the system. |
| BR-0006 | Users who have forgotten their password need to access the email of the account to receive the password recovery code. |
| BR-0007 | Users need real email to create new accounts. |
| BR-0008 | After users created an account, the email containing the activation code would be sent by the system, users need to type the activation code in the mobile application to activate their account. |
| BR-0009 | Accounts with roles Admin and Manager are allowed to log in to the web application while users to the mobile application. |
| BR-1001 | Admin can create a new manager account for Fooma Web application. |
| BR-1002 | Admin can manage, adjust manager, mobile user status from active to disable, and also set the status to active. |
| BR-1003 | Admin can manage users, and managers with all status(active, disabled) |
| BR-1004 | Admin can only view but cannot change the personal information of users and managers. |
| BR-1005 | The ingredient category is the category of the system ingredients, and also the type of the user's ingredients. |
| BR-1006 | The ingredient category can only be created, edited and removed by the admin. When the admin wants to create a new ingredient category, the admin needs to enter the name of the ingredient category, the default status is active and the Id will be automatically initialized. |
| BR-1007 | The ingredient category cannot have the same name. |
| BR-1008 | Recipes created by the admin will have the role "Admin". |
| BR-1009 | When the admin creates a recipe, it needs an ingredient list, origins(Vietnam, China,...), cooking method(fry,...), recipe categories(Asian food, Dinner,...), calories, servers, recipe images, cooking times(minutes). Each recipe will have one or more steps, each step includes content and illustrations. The default status will be active and the Id will be automatically initialized. |
| BR-1010 | The ingredients of the recipe match the ingredients of the system (same Id, name, and unit). |
| BR-1011 | Recipes can have multiple recipe categories. |
| BR-1012 | Recipes posted by admin can get user ratings, comments, and reactions. |
| BR-1013 | Admin can manage the Recipe country’s tag(origins). |
| BR-1014 | Admin can manage Recipe cooking methods. |
| BR-2001 | Only managers can add and modify system ingredients. |
| BR-2001 | To create a new ingredient the Fooma system needs the full name of the ingredient, the category of the ingredient, the image, and the unit, the Id of the system ingredient will be automatically initialized. |
| BR-2002 | The system ingredient name cannot be duplicated and cannot be blank. |
| BR-2003 | Each system ingredient can only have one category of ingredient. |
| BR-2004 | The system's ingredients are used to add recipes created by the admin. |
| BR-2005 | The manager can see all posts posted (by the manager and mobile user app) with full status(active, pending, denied, disabled). |
| BR-2006 | When the manager creates a Post, the manager needs to enter the Title, Content, and post images, the default Status for posts created by the manager will be active and will be directly posted to the Fooma community. |
| BR-2007 | The manager does not change the post content of the mobile app user. |
| BR-2008 | Manager can remove and restore posts of mobile app users. |
| BR-2009 | The manager can see the list of pending posts with the status "pending". |
| BR-2010 | For posts with "pending" status, the manager can choose to "accept" or "deny" to change the status of the post, allowing the article to be posted to the Fooma community. |
| BR-2011 | The manager can see all recipes (Recipe Post) posted by mobile app users with full status (active, pending, denied, disabled). |
| BR-2012 | The manager does not change the recipe content of the mobile app user. |
| BR-2013 | Managers can remove and restore recipes of mobile app users. |
| BR-2014 | The manager can see the list of recipes that are waiting for approval by the mobile app user with the status of "pending". |
| BR-2015 | For recipes with pending status, the manager can choose to accept or deny to change the status of the article, allowing the article to be posted to the Fooma community. |
| BR-2016 | The manager can view the list of reported users. |
| BR-2017 | The list of reported users will include reports with Title, Content, and Id, the username of the reporting user, and the Id, the username of the reported user. |
| BR-2018 | The manager can view the list of reported posts. |
| BR-2019 | The list of reported posts will include reports with Title, Content, and Id, the username of the reporting user, Id, and Title of the reported post. |
| BR-2020 | The manager can view the list of reported recipes. |
| BR-2021 | The reported recipe list will include reports with the Title, Content, and Id, the username of the reporting user, and the Id and Name of the reported recipe. |
| BR-3001 | The user can see a list of all ingredients in his refrigerator. Ingredients are sorted by remaining days in ascending order. |
| BR-3002 | Users can see the status of the ingredient(good: remainDate > 5, about to expire: 0 < remainDate <=5 , expired: remainDate = 0). |
| BR-3003 | User can search ingredients by name, category, and condition. |
| BR-3004 | When a user creates a new userIngredient, need system ingredientId(for ingredient exist in the system database, the ingredientDBId is matched with system ingredient called other), ingredientName, expiredDate, quantity, unit(kg, g,...), Id will be automatically generated and default status is "active" |
| BR-3005 | Users can view details, edit and remove userIngredient. |
| BR-3006 | When the user clicks to view detailed ingredients, recipes with that ingredient as the main ingredient will be displayed to suggest to the user. |
| BR-3007 | User can create a post, the user needs to enter Title, Content, and post images, Default Status for posts created by user will be "pending", role is "USER". |
| BR-3008 | User can edit, and remove their recipe. |
| BR-3009 | User can see a list of "pending", "active", "denied" posts divided into separate tabs. |
| BR-3010 | Users can react, and comment on their own posts and those of others displayed on the Fooma community. |
| BR-3011 | When a user creates a recipe, it needs an ingredient list, origins(Vietnam, China,...), cooking method(fry,...), recipe categories(Asian food, Dinner,...), |
| BR-3012 | calories, servers, recipe image, cooking times(minutes). Each recipe will have one or more steps, each step includes content and illustrations. The default status will be "pending" and the Id is automatically initialized. |
| BR-3013 | Users can edit, and remove their recipes. |
| BR-3014 | Users can view a list of "pending", "active", "denied" recipes divided into separate tabs. |
| BR-3015 | Users can react, and comment on their own recipe and that of others who appear on the Fooma community, only recipes with the role "Admin" can rate. |
| BR-3016 | When the user enters the recipe he wants to cook in the meal timetable, the system will compare the ingredients of the refrigerator with the recipe ingredients, thereby creating a list of ingredients to buy. |

## 

## 2. User Requirements

### 2.1 Overview

#### 2.1.1. Use Case Diagram



###### 

###### Figure 1. Use case diagram

#### 2.1.2. System Actors

###### Table 2. System actors

| **#** | **Actor** | **Description** |
| --- | --- | --- |
| 1 | Unauthenticated User | A user who has not logged in to Fooma Food Management Application. |
| 2 | Admin | A user who has logged in to Fooma Food Management Application with the role “ADMIN” |
| 3 | Manager | A user who has logged in to Fooma Food Management Application with the role “MANAGER” |
| 4 | Mobile User | A user who has logged in to Fooma Food Management Application with the role “USER” |
| 5 | System Handler | Fooma Food Management Application system. |

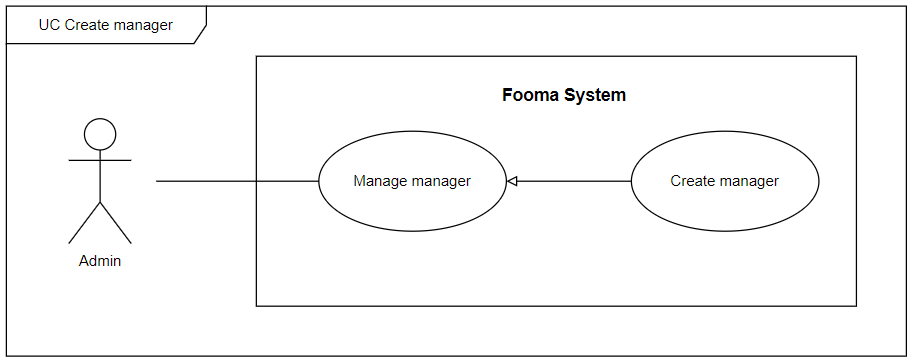
#### 2.1.3. Use Cases List

###### Table 3. Use Cases List

| **ID** | **Use Case** | **Primary Actors** | **Secondary Actors** |
| --- | --- | --- | --- |
| UC01 | Create Manager | Admin |  |
| UC02 | Get list Manager | Admin |  |
| UC03 | Enable/disable Manager's account | Admin |  |
| UC04 | View user infomation | Admin |  |
| UC05 | Ban/ unban user | Admin |  |
| UC06 | Create country's tag | Admin |  |
| UC07 | Get list country's tag | Admin |  |
| UC08 | Create cooking method | Admin |  |
| UC09 | Get list cooking method | Admin |  |
| UC10 | Disable cooking method | Admin |  |
| UC11 | Create ingredients | Admin |  |
| UC12 | Get list ingredient | Admin |  |
| UC13 | Update ingredient | Admin |  |
| UC14 | Disable ingredient | Admin |  |
| UC15 | Create recipe | Admin |  |
| UC16 | Get list recipe | Admin |  |
| UC17 | Update recipe | Admin |  |
| UC18 | Disable recipe | Admin |  |
| UC19 | Create categories of reicpe | Admin |  |
| UC20 | Get list categories of reicpe | Admin |  |
| UC21 | Update categories of reicpe | Admin |  |
| UC22 | Disable categories of reicpe | Admin |  |
| UC23 | Create categories of ingredient | Admin |  |
| UC24 | Get list categories of ingredient | Admin |  |
| UC25 | Update categories of ingredient | Admin |  |
| UC26 | Disable categories of ingredient | Admin |  |
| UC27 | Create notification | Admin |  |
| UC28 | Get list notification | Admin |  |
| UC29 | Update notification | Admin |  |
| UC30 | Disable notification | Admin |  |
| UC31 | Get list post | Admin |  |
| UC32 | Approve post | Admin |  |
| UC33 | Diable post | Admin |  |
| UC34 | Get list report | Admin |  |
| UC35 | Get list log | Admin |  |
| UC36 | View user infomation | Manager |  |
| UC37 | Ban/ unban user | Manager |  |
| UC38 | Create categories of reicpe | Manager |  |
| UC39 | Get list categories of reicpe | Manager |  |
| UC40 | Update categories of reicpe | Manager |  |
| UC41 | Disable categories of reicpe | Manager |  |
| UC42 | Create categories of ingredient | Manager |  |
| UC43 | Get list categories of ingredient | Manager |  |
| UC44 | Update categories of ingredient | Manager |  |
| UC45 | Disable categories of ingredient | Manager |  |
| UC46 | Get list post | Manager |  |
| UC47 | Approve post | Manager |  |
| UC48 | Diable post | Manager |  |
| UC49 | Get list report | Manager |  |
| UC50 | View profile | Mobile User |  |
| UC51 | Update profile | Mobile User |  |
| UC52 | Change password | Mobile User |  |
| UC53 | Create ingredients | Mobile User |  |
| UC54 | Update ingredients | Mobile User |  |
| UC55 | Delete ingredients | Mobile User |  |
| UC56 | View ingredients | Mobile User |  |
| UC57 | Search recipes | Mobile User |  |
| UC58 | View recipes | Mobile User |  |
| UC59 | Rating recipes | Mobile User |  |
| UC60 | Save recipes | Mobile User |  |
| UC61 | Report recipes | Mobile User |  |
| UC62 | React posts | Mobile User |  |
| UC63 | Create cooking plans | Mobile User |  |
| UC64 | Update cooking plans | Mobile User |  |
| UC65 | Delete cooking plans | Mobile User |  |
| UC66 | View cooking plans | Mobile User |  |
| UC67 | Create posts | Mobile User |  |
| UC68 | Update posts | Mobile User |  |
| UC69 | Delete posts | Mobile User |  |
| UC70 | View posts | Mobile User |  |
| UC71 | React posts | Mobile User |  |
| UC72 | Comment posts | Mobile User |  |
| UC73 | Report posts | Mobile User |  |
| UC74 | Save posts | Mobile User |  |
| UC75 | Report user | Mobile User |  |
| UC76 | Login | Unauthenticated User |  |
| UC77 | Login with google | Unauthenticated User |  |
| UC78 | Register | Unauthenticated User |  |
| UC79 | Forgot password | Unauthenticated User |  |
| UC80 | Send mail | System Handler |  |
| UC81 | Push notification | System Handler |  |
| UC82 | Recipe suggestions for users | System Handler |  |
| UC83 | Arrange posts for managers to approve | System Handler |  |
| UC84 | Weekly delete temporal data | System Handler |  |

### 2.2. Use case specification

#### 2.2.1 UC01 – Create manager

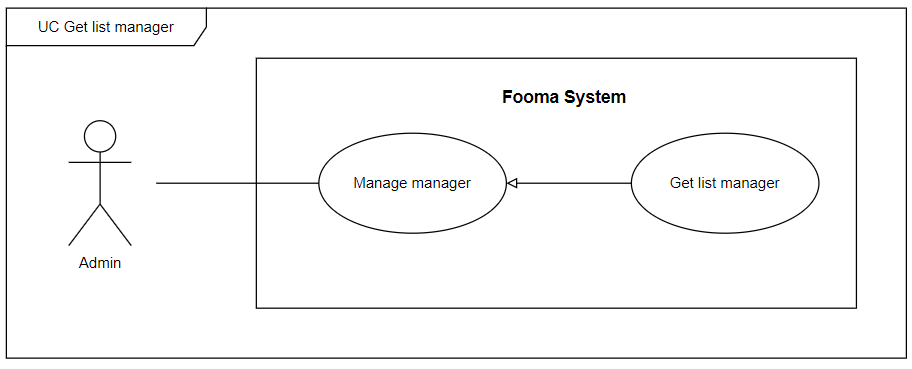


###### Figure 2. <Use case> Create manager

###### Table 4. UC01 - Create manager

| ID and Name: | **UC-01 Create manager** | | |
| --- | --- | --- | --- |
| Created By: | BaoHg |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin needs to create a new manager. | | |
| Trigger: | Admin needs to try to create a new manager. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage manager page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return a “Create Success” message. An email with an auto-generated password would be sent to the registered email(the manager needs to change the password in the first login). * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **1.0 Create Manager**   1. Admin Click on the “Create Manager” button on the Manage manager page. 2. Fooma Web Application shows create manager form:    * Email: text input    * Name: text input    * Phone number: text input    * Birthday: date input    * Image URL: file input    * Bio: text area    * Create Button 3. Admin fill all items of creating manager form and click the “Create” button. 4. The Create manager form will be closed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **1.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again (MSG0105).   **1.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields (MSG0104).   **1.0.E3 Admin inputs existed email**   1. The system shows an existing email message (MSG0107). | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1001 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.2 UC02 – Get list manager

****

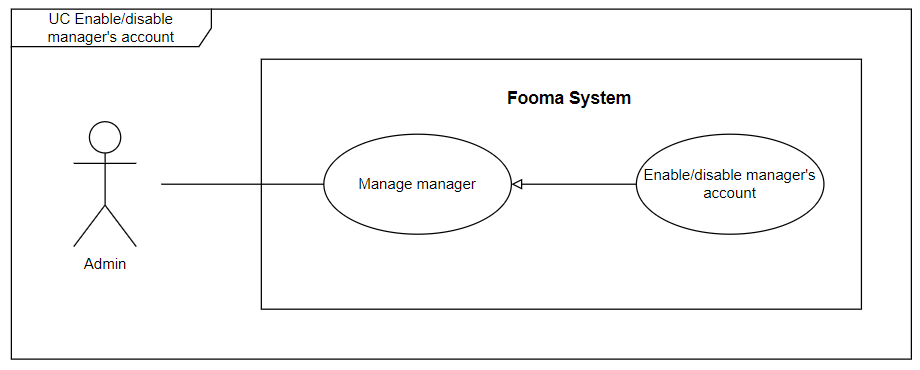
###### Figure 3. <Use case> Get list manager

###### Table 5. UC02 - Get list manager

| ID and Name: | **UC-01 Create manager** | | |
| --- | --- | --- | --- |
| Created By: | BaoHg |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can view the list of managers with a search filter. | | |
| Trigger: | Admin clicks on Manage manager pages or fills some search filter and click Search on the Manage manager pages to get a list of managers. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage manager page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show the manager list matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **2.0 View List Manager**   1. Admin Click on the Manage manager pages or fill on the search box, applied on the search filter on the Manage manager page. 2. Fooma Web Application shows a list of managers, each row would display:  * Avatar: image * Email: text * Name: text * Birthdate: date * Phone number: text * Bio: text * Status: text * Action:   + View detail   + Edit   + Enable/Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **2.0.E1 Manager list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1002, BR-1003, BR-1004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.3 UC03 – Enable/ disable manager’s account

****

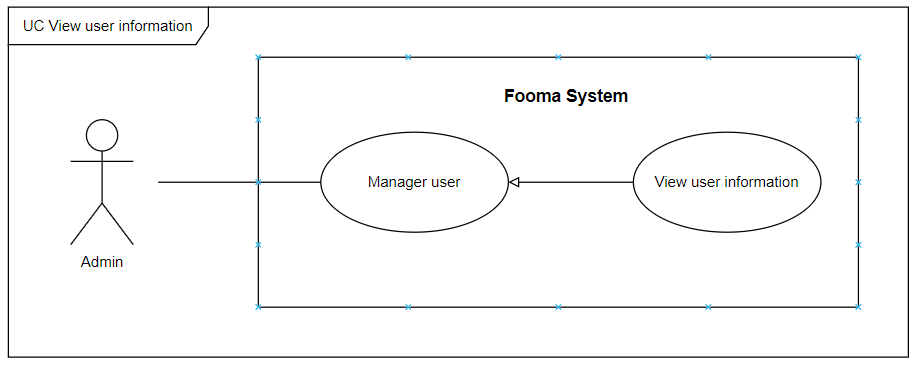
###### Figure 4. <Use case> Enable/disable manager’s account

###### Table 6. UC03 - Enable/disable manager’s account

| ID and Name: | **UC-03 Enable/disable manager’s account** | | |
| --- | --- | --- | --- |
| Created By: | BaoHg |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can enable, or disable manager's accounts. | | |
| Trigger: | On the manager list, the admin clicks on Enable or Disable button based on the manager’s status. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage manager page.  PRE-3. Required manager account displayed in the manager list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list manager view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **3.0 Enable/Disable manager’s account**   1. Admin gets the list of managers. 2. On the specific manager in the manager list, the admin clicks on Enable/Disable button based on the manager’s current status. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve enable/disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **3.0.E1 Manager no longer exist**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1002 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.4 UC04 – View user information

****

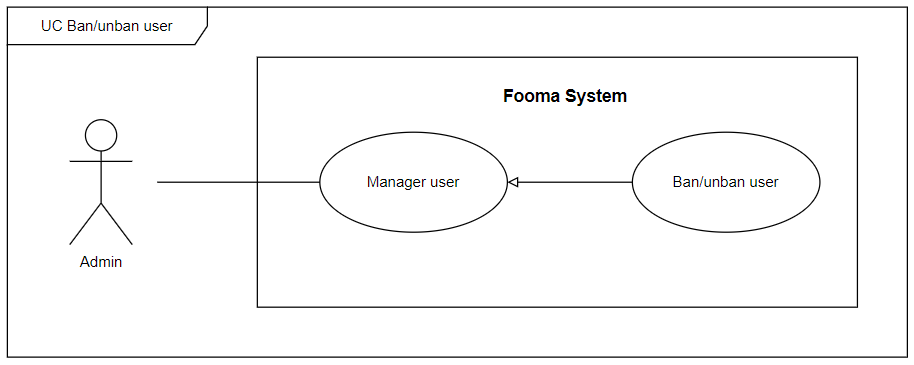
###### Figure 5. <Use case> View user information

###### Table 7. UC04 - View user information

| ID and Name: | **UC-04 View user information** | | |
| --- | --- | --- | --- |
| Created By: | BaoHg |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can view detailed information of a user account. | | |
| Trigger: | On the user list, at the specific user account row, the admin click on the View detail button. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage user page.  PRE-3. Required user account displayed in the user list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show details of the user. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **4.0 View user’s information**   1. Admin gets a list of users. 2. On the specific user in the user list, the admin clicks on the Details button. 3. Fooma Web Application redirects to view details of the user. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **4.0.E1 User no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1002, BR-1004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.5 UC05 – Ban/Unban user

****

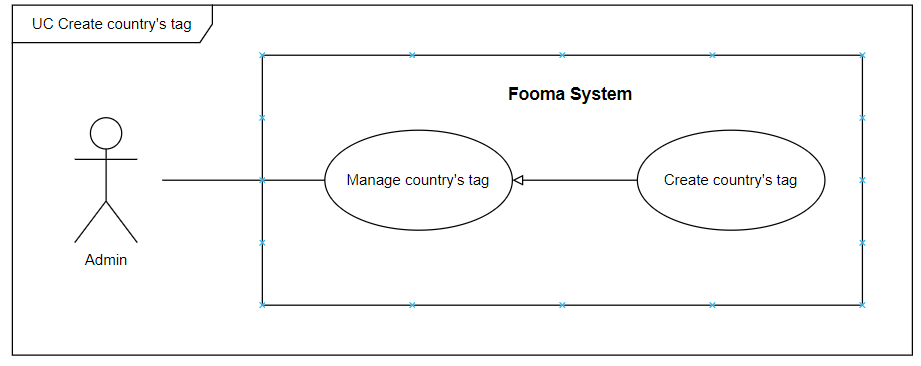
###### Figure 6. <Use case> Ban/unban user

###### Table 8. UC05 - Ban/unban user

| ID and Name: | **UC-05 Ban/unban user** | | |
| --- | --- | --- | --- |
| Created By: | BaoHg |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can ban/unban user. | | |
| Trigger: | On the user list, at the specific user account row, the admin clicks on the ban/unban button based on the user's current status. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage user page.  PRE-3. Required user account displayed in the user list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list user view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **5.0 Enable/Disable manager’s account**   1. Admin gets a list of users. 2. On the specific manager in the user list, the admin clicks on the Ban/unban button based on the user's current status. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve ban/unban action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **5.0.E1 Manager no longer exist**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0003 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.6 UC06 – Create country’s tag

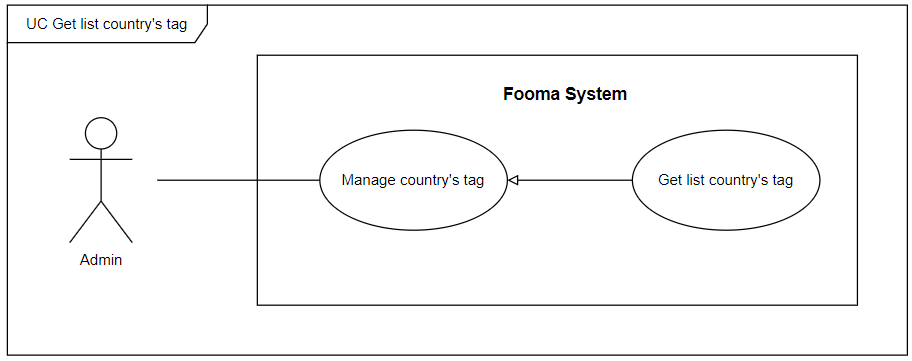


###### Figure 7. <Use case> Create country’s tag

###### Table 9. UC06 - Create country’s tag

| ID and Name: | **UC-06 Create country’s tag** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin needs to create a new country’s tag. | | |
| Trigger: | Admin tries to create a country's tag. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage country’s tag page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Create country’s tag Success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **6.0 Create country’s tag**   1. Admin Click on the “Create Country’s tag” button on the Manage country’s tag page. 2. Fooma Web Application shows create country’s tag form:    * Name: text input    * Create Button 3. Admin fill all items of creating a country's tag form and click the “Create” button. 4. The Create country's tag form will be closed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **6.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again (MSG0105).   **6.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields (MSG0104).   **6.0.E3 Admin inputs existed country’s tag**   1. The system shows an existing country’s tag message. | | |
| Priority: | Medium | | |
| Frequency of Use: | Medium | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1013 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.7 UC07 – Get list country’s tag

****

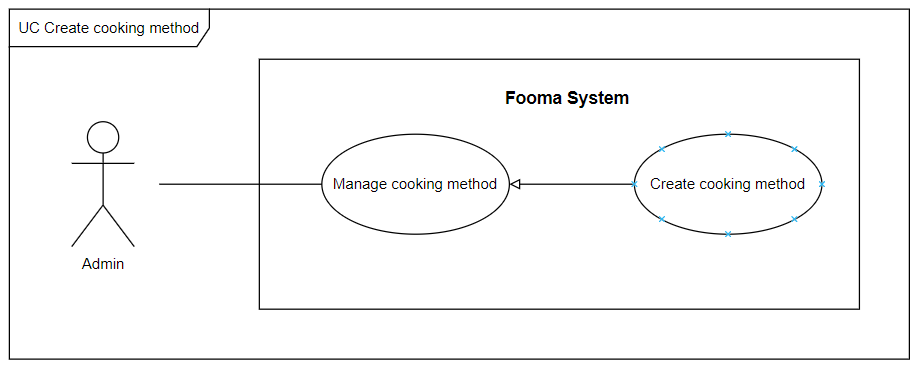
###### Figure 8. <Use case> Get list country’s tag

###### Table 10. UC07 - Get list country’s tag

| ID and Name: | **UC-07 Get list country’s tag** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can view a list of country’s tag with search filter. | | |
| Trigger: | Admin clicks on Manage country’s tag page or fills some search filter and click Search on the Manage country’s tag pages to get list country’s tag. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage country’s tag page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show the list country’s tag matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **7.0 View List Manager**   1. Admin Click on Manage country’s tag page or fill on the search box, applied on the search filter on the Manage country’s tag page. 2. Fooma Web Application shows a list of country’s tags, each row would display:  * Id: text * Name: text * Create Date: text * Status: text | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **7.0.E1 Country’s tag list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1013 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.8 UC08 – Create cooking method



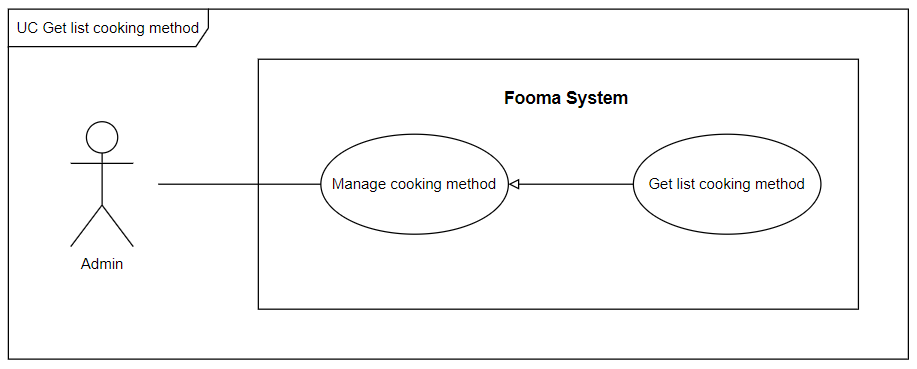
###### Figure 9. <Use case> Create cooking method

###### Table 11. UC08 - Create cooking method

| ID and Name: | **UC-08 Create cooking method** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin needs to create a new cooking method. | | |
| Trigger: | Admin tries to create a cooking method. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage cooking method page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Create cooking method Success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **8.0 Create cooking method**   1. Admin Click on the “Create cooking method” button on the Manage country’s tag page. 2. Fooma Web Application shows create cooking method form:    1. Name: text input    2. Create Button 3. Admin fill all items of the create cooking method form and click the “Create” button. 4. The Create cooking method form will be closed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **8.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **8.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **8.0.E3 Admin inputs existed country’s tag**   1. The system shows an existing country’s tag message. | | |
| Priority: | Medium | | |
| Frequency of Use: | Medium | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1014 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.9 UC09 – Get list cooking method

****

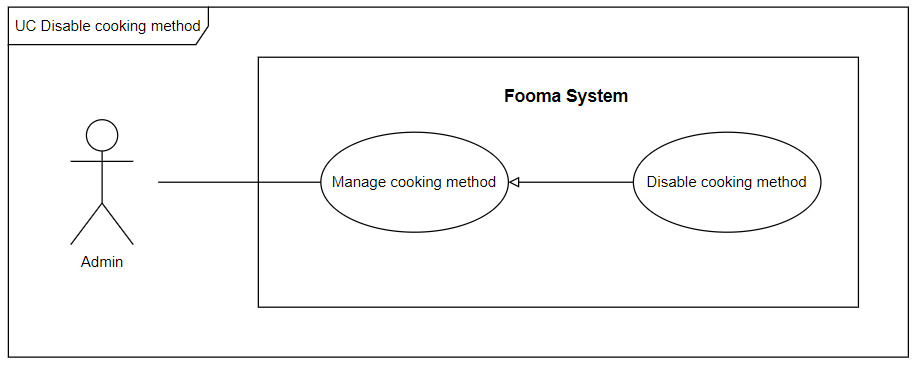
###### Figure 10. <Use case> Get list cooking method

###### Table 12. UC09 - Get list cooking method

| ID and Name: | **UC-09 Get list cooking method** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can view a list of cooking methods with a search filter. | | |
| Trigger: | Admin clicks on the Manage cooking method page or fills some search filter and clicks Search on the Manage cooking method page to get a list of cooking methods. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at manage cooking method page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show the list country’s tag matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **9.0 Get list cooking method**   1. Admin Click on the Manage cooking method page or fill on the search box, applied on the search filter on the Manage cooking method page. 2. Fooma Web Application shows a list of cooking methods, each row would display:  * Id: text * Name: text * Create Date: date * Status: text | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **9.0.E1 Cooking method list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1014 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.10 UC10 – Disable cooking method

****

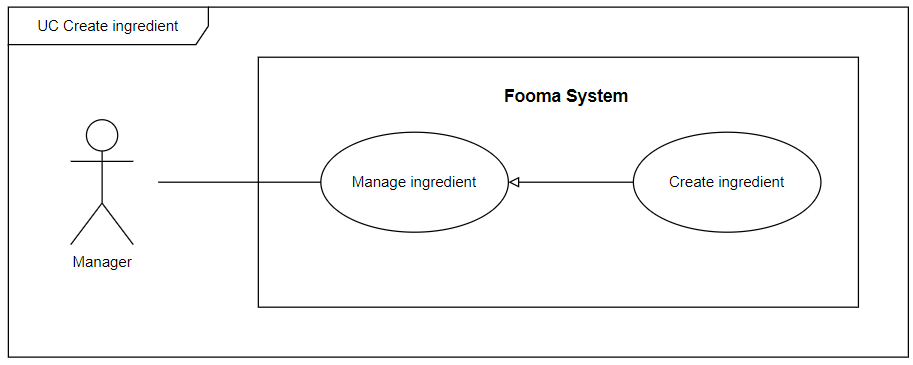
###### Figure 11. <Use case> Disable cooking method

###### Table 13. UC10 - Disable cooking method

| ID and Name: | **UC-10 Disable cooking method** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can disable cooking methods. | | |
| Trigger: | On the cooking method list, at the specific cooking method row, admin clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with admin role.  PRE-2. Admin at manage user page.  PRE-3. Required cooking method is displayed in the cooking method list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show success message and update list cooking method view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **10.0 Disable cooking method**   1. Admin gets a list of cooking methods. 2. On the specific manager in the cooking method list, admin click on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **10.0.E1 Cooking method no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0003, BR-1014 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.11 UC11 – Create ingredient



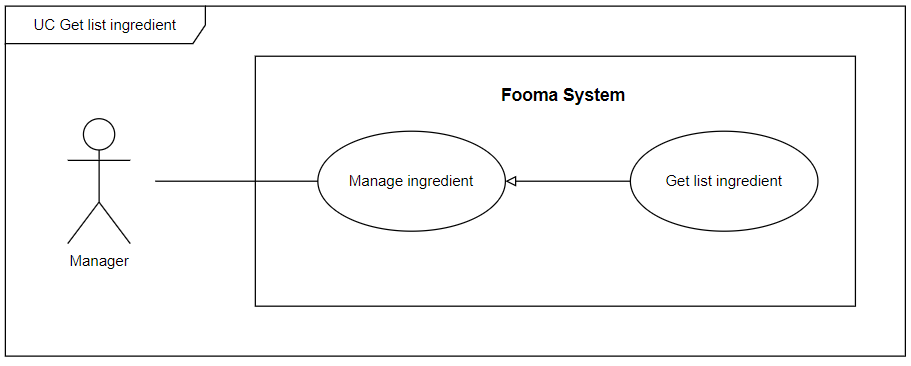
###### Figure 12. <Use case> Create ingredient

###### Table 14. UC11 - Create ingredient

| ID and Name: | **UC-11 Create ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager need to create new ingredients. | | |
| Trigger: | Manager tries to create ingredients. | | |
| Preconditions: | PRE-1. User has logged in with the manager role.  PRE-2. Manager is at the manage ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return “Create ingredient success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **11.0 Create ingredient**   1. Manager Click on the “Create ingredient” button at the Manage ingredient page. 2. Fooma Web Application show create ingredient form:    1. Ingredient Name: text input    2. Unit: text input    3. Category: dropdown    4. Image: file input    5. Create Button 3. The manager fills all items on the ingredient form and clicks the “Create” button. 4. The Create cooking method form will be closed. | | |
| Alternative Flows: | **11.0 Manager selects the “Close” button**   1. The manager selects the “Close” button. 2. Fooma Web Application closes the created ingredient form and returns to the ingredient list view. | | |
| Exceptions: | **11.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again (MSG0105).   **11.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields (MSG0104).   **11.0.E3 Manager inputs existed ingredient**   1. The system shows an existing ingredient message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-2001, BR-2002, BR-2003, BR-2004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.12 UC12 – Get list ingredient

****

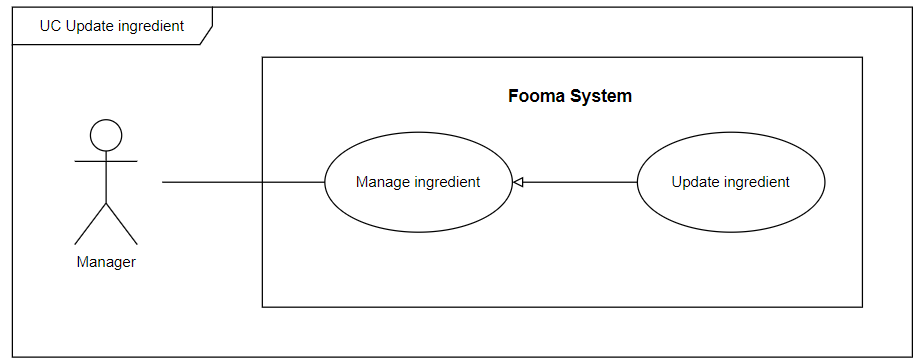
###### Figure 13. <Use case> Get list ingredient

###### Table 15. UC12 - Get list ingredient

| ID and Name: | **UC-12 Get list ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | The manager can view the list of ingredients with a search filter. | | |
| Trigger: | The manager clicks on the Manage ingredient page or fill some search filter and click Search on the Manage ingredient page to get a list of ingredient. | | |
| Preconditions: | PRE-1. User has logged in with the manager role.  PRE-2.Manger at manage ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a list of ingredients matched with the search filter which is sent by the manager. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **12.0 Get list ingredient**   1. The manager clicks on the Manage ingredient page or fills on the search box, and applied the search filter on the Manage ingredient page. 2. Fooma Web Application shows a list of ingredient, each row would display:  * Name: text * Create Date: text * Category name: text * Unit: text * Status: text * Action:   + Update   + Remove | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **12.0.E1 Ingredient list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 BR-2001, BR-2002, BR-2003, BR-2004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.13 UC13 – Update ingredient

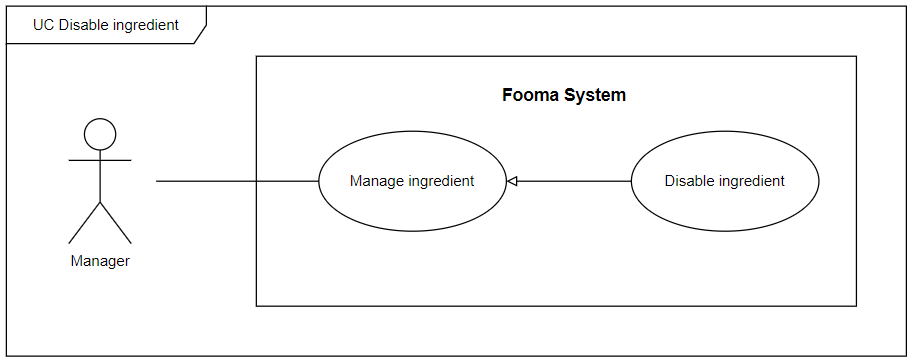


###### Figure 14. <Use case>Update Ingredient

###### Table 16. UC13 - Update ingredient

| ID and Name: | **UC-13 Update ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager needs to update ingredients. | | |
| Trigger: | Admin tries to update ingredients. | | |
| Preconditions: | PRE-1. User has logged in with the manager role.  PRE-2. Manager at manage ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return an “Update ingredient success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **13.0 Update ingredient**   1. Admin Click on the “Update” button at a specific ingredient in the ingredient list. 2. Fooma Web Application show update ingredient form:    1. Ingredient Name: text input    2. Unit: text input    3. Category: dropdown    4. Image: file input    5. Update Button 3. Manager fills all items of the update ingredient form and clicks the “Update” button. 4. The Update ingredient form will be closed and the ingredient list will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **13.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **13.0.E2 Manager inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **13.0.E3 Manager inputs existed ingredient name**   1. The system shows an existed ingredient name message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-2001, BR-2002, BR-2003, BR-2004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.14 UC14 – Disable ingredient

****

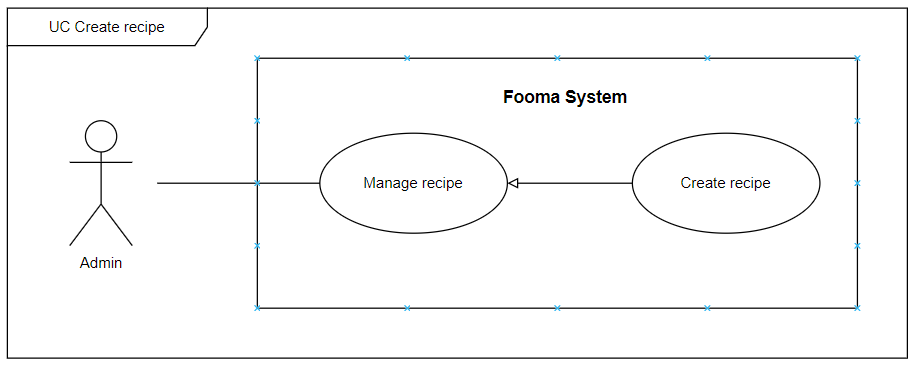
###### Figure 15. <Use case> Disable ingredient

###### Table 17. UC14 - Disable ingredient

| ID and Name: | **UC-14 Disable ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager can disable ingredient. | | |
| Trigger: | On the ingredient list, at the specific user account row, the admin clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the manager role.  PRE-2. Admin at manage user page.  PRE-3. The required ingredient is displayed in the ingredient list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list ingredient view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **14.0 Disable ingredient**   1. The manager gets a list of ingredients. 2. On the specific ingredient in the ingredient list, the Manager click on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **14.0.E1 Ingredient no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0003 BR-2001, BR-2002, BR-2003, BR-2004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.15 UC15 – Create recipe

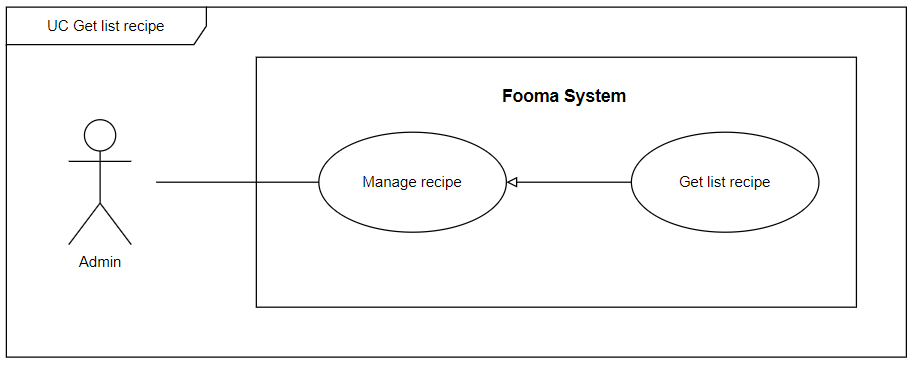


###### Figure 16. <Use case> Create recipe

###### Table 17. UC15 - Create recipe

| ID and Name: | **UC-15 Create recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin needs to create a new recipe. | | |
| Trigger: | Admin tries to create a new recipe. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at the manage recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return “Create recipe success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **15.0 Create recipe**   1. admin Click on the “Create Recipe” button on the Manage recipe page. 2. Fooma Web Application shows create recipe form:    1. Recipe Name: text input    2. Description: text input    3. Origin: dropdown    4. Cooking method: dropdown    5. Category: dropdown    6. Hashtag: text input    7. Image: file input    8. Recipe ingredient:       1. Ingredient name: text input       2. Quantity: number input       3. Unit: text input    9. Recipe method:       1. Content: text input       2. Image: file input    10. Create Button 3. admin fills all items of the recipe form and clicks the “Create” button. 4. The Create recipe form will be closed. | | |
| Alternative Flows: | **15.0 Recipe selects the “Close” button**   1. admin selects the “Close” button. 2. Fooma Web Application closes the create recipe form and returns to the recipe list view. | | |
| Exceptions: | **15.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **15.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **15.0.E3 Admin inputs existed recipe**   1. The system shows an existing recipe message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1008, BR-1009, BR-1010, BR-1011, BR-1012, BR-1013, BR-1014 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.16 UC16 – Get list recipe

****

###### Figure 17. <Use case> Get list recipe

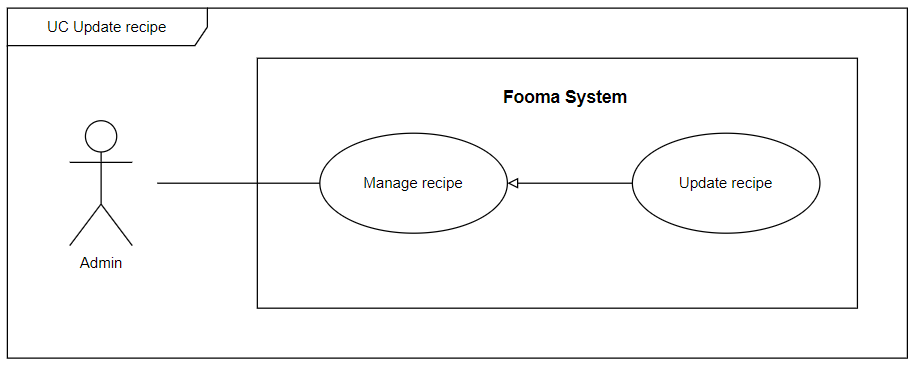
###### Table 19. UC16 - Get list recipe

| ID and Name: | **UC-16 Get list recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | admin |  |  |
| Description: | admin can view the list of recipes with a search filter. | | |
| Trigger: | admin click on Manage recipe page or fill some search filter and click Search on the Manage recipe page to Get list recipe. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Manger at manage recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a list of recipes matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **16.0 Get list recipe**   1. admin clicks on the Manage recipe page or fills on the search box, and applied the search filter on the Manage recipe page. 2. Fooma Web Application shows a list of a recipe, each row would display  * Title: text * User name: text * Published date: date * Hashtag: text * Cooking method: text * Categories: text * Action:   + Update   + Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **16.0.E1 recipe list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 BR-2001, BR-2002, BR-2003, BR-2004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 

#### 2.2.17 UC17 – Update recipe

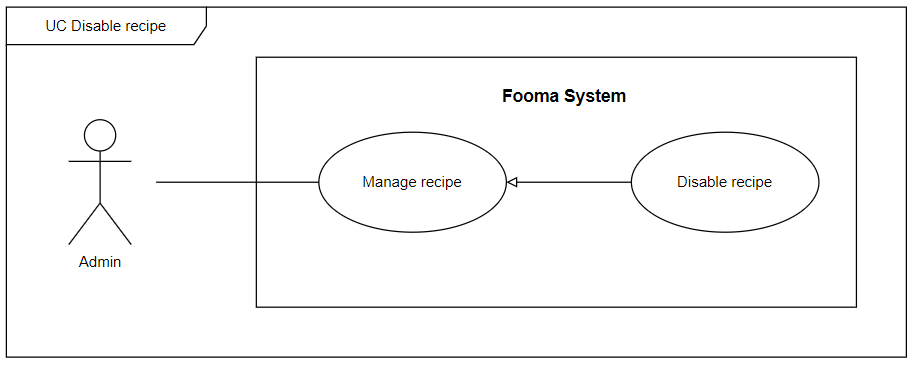


###### Figure 18. <Use case>Update recipe

###### Table 20. UC17 - Update recipe

| ID and Name: | **UC-17 Update recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | admin needs to update the recipe. | | |
| Trigger: | Admin tries to update the recipe. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Admin at manage recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return an “Update recipe success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **17.0 Update recipe**   1. Admin Click on the “Update” button at a specific recipe in the recipe list. 2. Fooma Web Application show update recipe form:    1. Recipe Name: text input    2. Description: text input    3. Origin: dropdown    4. Cooking method: dropdown    5. Category: dropdown    6. Hashtag: text input    7. Image: file input    8. Recipe ingredient:       1. Ingredient name: text input       2. Quantity: number input       3. Unit: text input    9. Recipe method:       1. Content: text input       2. Image: file input    10. Update Button 3. Admin fills all items of the update recipe form and clicks the “Update” button. 4. The Update recipe form will be closed and the recipe list will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **17.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **17.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **17.0.E3 Admin inputs existed recipe name**   1. The system shows an existed recipe name message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-2001, BR-2002, BR-2003, BR-2004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.18 UC18 – Disable recipe

****

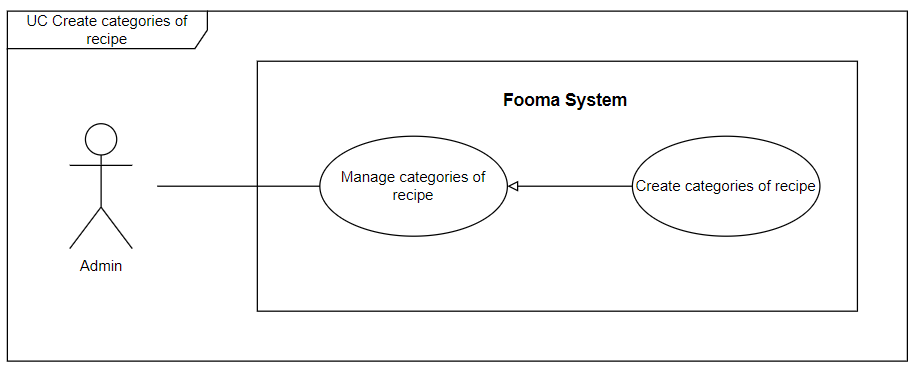
###### Figure 19. <Use case> Disable recipe

###### Table 20. UC18 - Disable recipe

| ID and Name: | **UC-18 Disable recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can disable recipe. | | |
| Trigger: | On the recipe list, at the specific user account row, the admin clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the Admin role.  PRE-2. Admin at manage user page.  PRE-3. The required recipe is displayed in the recipe list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list recipe view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **18.0 Disable recipe**   1. Admin gets a list of recipes. 2. On the specific recipe in the recipe list, the admin click on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **18.0.E1 recipe no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0003 BR-2001, BR-2002, BR-2003, BR-2004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.19 UC19 – Create categories of recipe



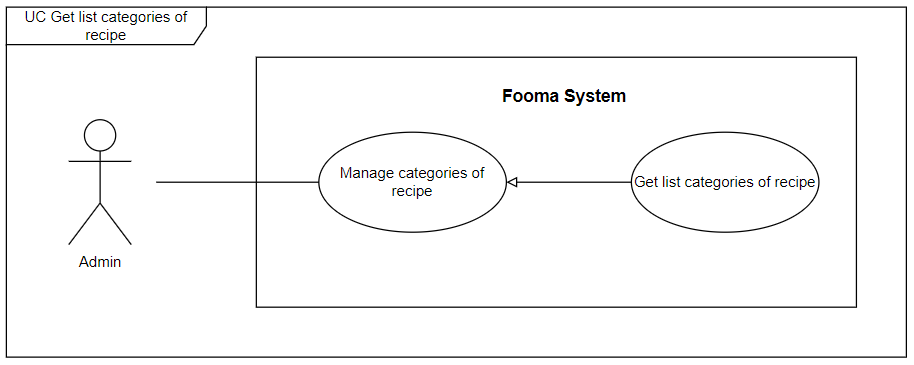
###### Figure 20. <Use case> Create categories of recipe

###### Table 22. UC19 - Create categories of recipe

| ID and Name: | **UC-19 Create categories of recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin need to create new categories of recipe. | | |
| Trigger: | Admin tries to create new categories of recipe. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at the manage categories of recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Create categories of recipe success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **19.0 Create categories of recipe**   1. Admin Click on the “Create categories of recipe” button on the Manage categories of recipe page. 2. Fooma Web Application shows create categories of recipe form:    1. Categories of recipe Name: text input    2. Create Button 3. Admin fills all items of the categories of recipe form and clicks the “Create” button. 4. The Create categories of recipe form will be closed. | | |
| Alternative Flows: | **19.0 categories of recipe selects the “Close” button**   1. Admin selects the “Close” button. 2. Fooma Web Application closes the create categories of recipe form and returns to the categories of recipe list view. | | |
| Exceptions: | **19.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **19.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **19.0.E3 Admin inputs existed categories of recipe**   1. The system shows existing categories of recipe messages. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.20 UC20 – Get list categories of recipe

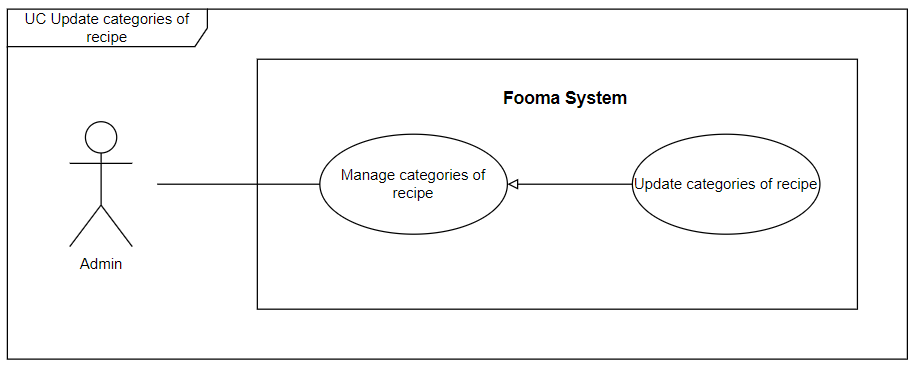
****

###### Figure 21. <Use case> Get list categories of recipe

###### Table 23. UC20 - Get list categories of recipe

| ID and Name: | **UC-20 Get list categories of recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | admin |  |  |
| Description: | admin can view the list of categories of recipes with a search filter. | | |
| Trigger: | admin click on Manage categories of recipe page or fill some search filter and click Search on the Manage categories of recipe page to Get list categories of recipe. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Manger at manage categories of recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show list categories of recipes matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **20.0 Get list categories of recipe**   1. admin click on Manage categories of recipe page or fill on the search box, applied on the search filter on the Manage categories of recipe page. 2. Fooma Web Application shows a list of categories of recipe, each row would display:  * Id: text * Categories of recipe Name: text * Action:   + Update   + Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **20.0.E1 categories of recipe list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.21 UC21 – Update categories of recipe

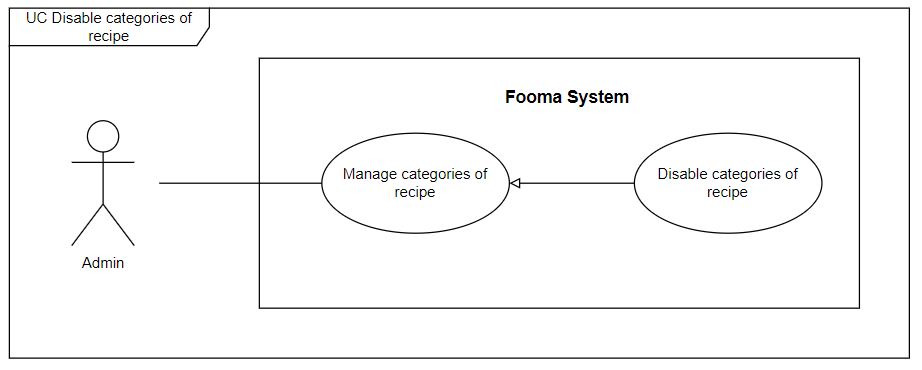


###### Figure 22. <Use case>Update categories of recipe

###### Table 24. UC21 - Update categories of recipe

| ID and Name: | **UC-21 Update categories of recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin needs to update categories of recipe. | | |
| Trigger: | Admin tries to update categories of recipe. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Admin at manage categories of recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Update categories of recipe success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **21.0 Update categories of recipe**   1. Admin Click on the “Update” button at specific categories of recipe in the categories of recipe list. 2. Fooma Web Application show update categories of recipe form:    1. categories of recipe Name: text input    2. Update Button 3. Admin fills all items of the update categories of recipe form and clicks the “Update” button. 4. The Update categories of recipe form will be closed and the categories of recipe list will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **21.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **21.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **21.0.E3 Admin inputs existed categories of recipe name**   1. The system shows an existed categories of recipe name messages. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.22 UC22 – Disable categories of recipe

****

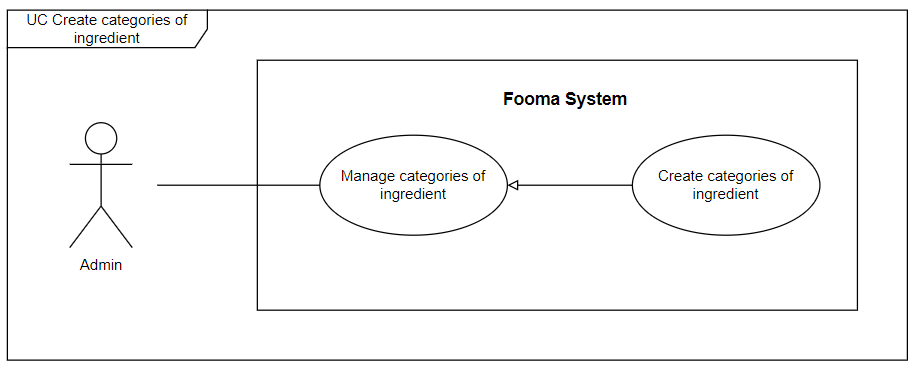
###### Figure 23. <Use case> Disable categories of recipe

###### Table 25. UC22 - Disable categories of recipe

| ID and Name: | **UC-22 Disable categories of recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can disable categories of recipe. | | |
| Trigger: | On the categories of recipe list, at the specific user account row, the admin clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the Admin role.  PRE-2. Admin at manage user page.  PRE-3. Required categories of recipe is displayed in the categories of recipe list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list categories of recipe view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **22.0 Disable categories of recipe**   1. Admin gets a list of categories of recipe. 2. On the specific categories of recipe in the categories of recipe list, the admin clicks on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **22.0.E1 categories of recipe no longer exist**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.23 UC23 – Create categories of ingredient



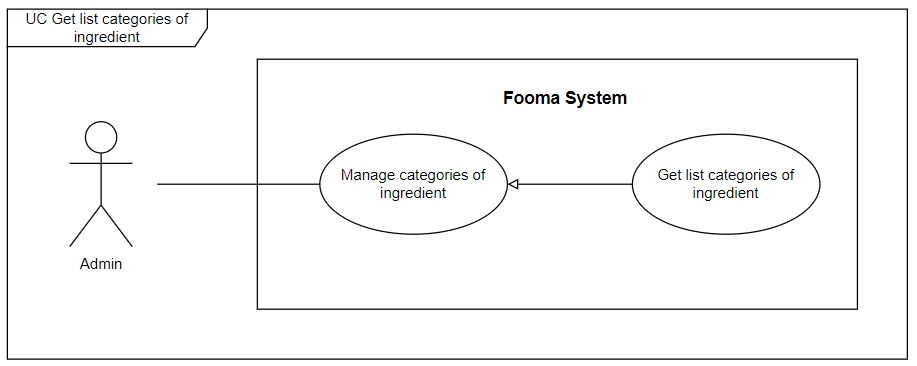
###### Figure 24. <Use case> Create categories of ingredient

###### Table 26. UC23 - Create categories of ingredient

| ID and Name: | **UC-23 Create categories of ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin need to create new categories of ingredient. | | |
| Trigger: | Admin tries to create new categories of ingredient. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at the manage categories of ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Create categories of ingredient success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **23.0 Create categories of ingredient**   1. Admin Click on the “Create categories of ingredient” button on the Manage categories of ingredient page. 2. Fooma Web Application shows create categories of ingredient form:    1. categories of ingredient Name: text input    2. Create Button 3. Admin fills all items of the categories of ingredient form and clicks the “Create” button. 4. The Create categories of ingredient form will be closed. | | |
| Alternative Flows: | **23.0 categories of ingredient selects the “Close” button**   1. Admin selects the “Close” button. 2. Fooma Web Application closes the create categories of ingredient form and returns to the categories of ingredient list view. | | |
| Exceptions: | **23.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **23.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **23.0.E3 Admin inputs existed categories of ingredient**   1. The system shows existing categories of ingredient messages. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.24 UC24 – Get list categories of ingredient

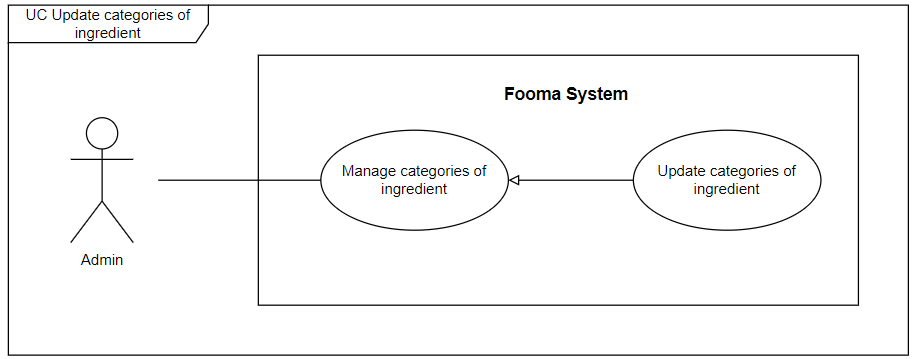
****

###### Figure 25. <Use case> Get list categories of ingredient

###### Table 27. UC24 - Get list categories of ingredient

| ID and Name: | **UC-24 Get list categories of ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | admin |  |  |
| Description: | admin can view the list of categories of ingredients with a search filter. | | |
| Trigger: | admin click on Manage categories of ingredient page or fill some search filter and click Search on the Manage categories of ingredient page to Get list categories of ingredient. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Manger at manage categories of ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show list categories of ingredients matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **24.0 Get list categories of ingredient**   1. admin click on Manage categories of ingredient page or fill on the search box, applied on the search filter on the Manage categories of ingredient page. 2. Fooma Web Application show a list of categories of ingredient, each row of would display:  * Id: text * categories of ingredient Name: text * Action:   + Update   + Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **24.0.E1 categories of ingredient list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.25 UC25 – Update categories of ingredient

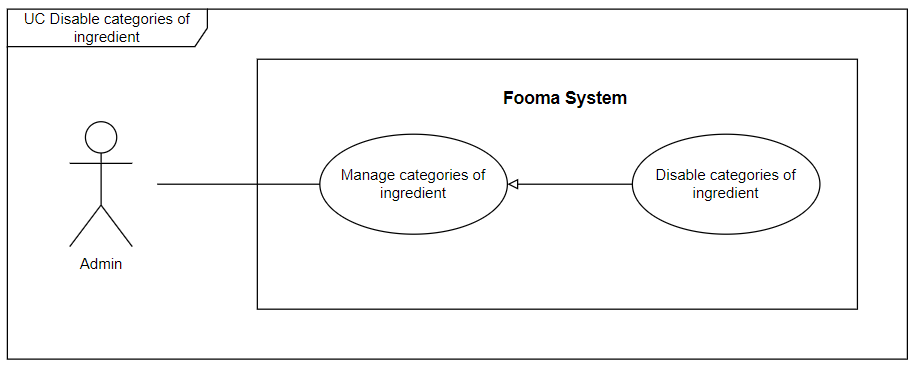


###### Figure 26. <Use case>Update categories of ingredient

###### Table 28. UC25 - Update categories of ingredient

| ID and Name: | **UC-25 Update categories of ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin needs to update categories of ingredient. | | |
| Trigger: | Admin tries to update categories of ingredient. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Admin at manage categories of ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Update categories of ingredient success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **25.0 Update categories of ingredient**   1. Admin Click on the “Update” button at a specific category of ingredient in the categories of ingredient list. 2. Fooma Web Application show update categories of ingredient form:    1. categories of ingredient Name: text input    2. Update Button 3. Admin fills all items of the update categories of ingredient form and clicks the “Update” button. 4. The Update categories of ingredient form will be closed and the categories of ingredient list will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **25.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **25.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **25.0.E3 Admin inputs existed categories of ingredient name**   1. System shows a existed categories of ingredient name message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.26 UC26 – Disable categories of ingredient

****

###### Figure 27. <Use case> Disable categories of ingredient

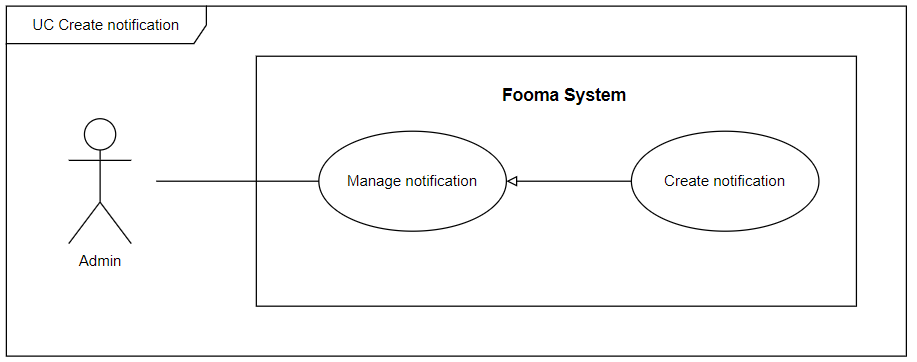
###### Table 29. UC26 - Disable categories of ingredient

| ID and Name: | **UC-26 Disable categories of ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can disable categories of ingredient. | | |
| Trigger: | On the categories of ingredient list, at the specific user account row, admin clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the Admin role.  PRE-2. Admin at manage user page.  PRE-3. Required categories of ingredient is displayed in the categories of ingredient list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show success message and update list categories of ingredient view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **26.0 Disable categories of ingredient**   1. Admin gets a list of categories of ingredient. 2. On the specific categories of ingredient in the categories of ingredient list, the admin click on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **26.0.E1 categories of ingredient no longer exist**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 

#### 2.2.27 UC27 – Create notification



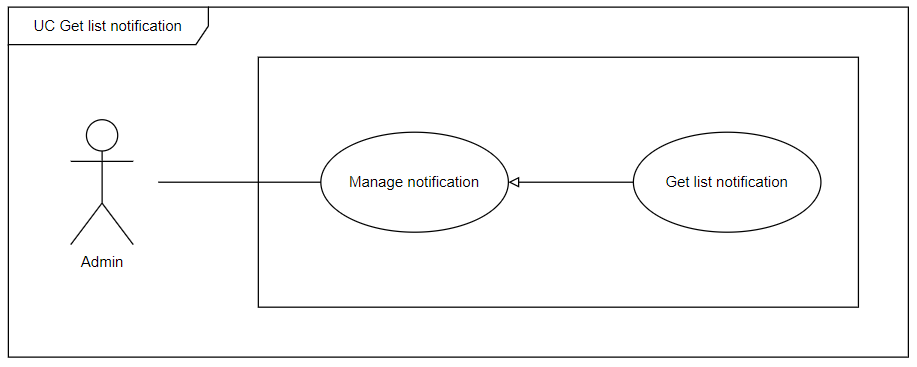
###### Figure 28. <Use case> Create notification

###### Table 30. UC27 - Create notification

| ID and Name: | **UC-27 Create notification** | | |
| --- | --- | --- | --- |
| Created By: |  |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin needs to create a new notification. | | |
| Trigger: | Admin tries to create a new notification. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2. Admin at the manage notification page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return “Create notification success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **27.0 Create notification**   1. Admin Click on the “Create notification” button on the Manage notification page. 2. Fooma Web Application shows create notification form:    * notification content: text input    * userId: text input    * Create Button 3. Admin fills all items of the notification form and clicks the “Create” button. 4. The Create notification form will be closed. | | |
| Alternative Flows: | **27.0 notification selects the “Close” button**   1. Admin selects the “Close” button. 2. Fooma Web Application closes the create notification form and returns to the notification list view. | | |
| Exceptions: | **27.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **27.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **27.0.E3 Admin inputs existed notification**   1. The system shows an existing notification message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.28 UC28 – Get list notification

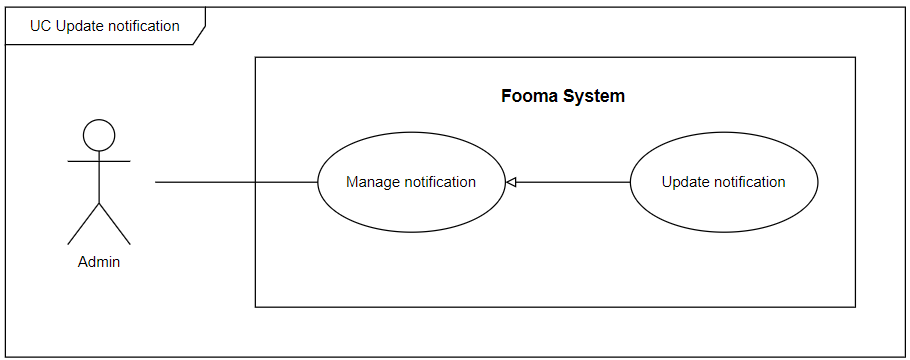
****

###### Figure 29. <Use case> Get list notification

###### Table 31. UC28 - Get list notification

| ID and Name: | **UC-28 Get list notification** | | |
| --- | --- | --- | --- |
| Created By: |  |  |  |
| Primary Actor: | admin |  |  |
| Description: | admin can view the list of notifications with a search filter. | | |
| Trigger: | admin click on the Manage notification page or fill some search filter and click Search on the Manage notification page to Get list notification. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Manger at manage notification page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show list notifications matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **28.0 Get list notification**   1. admin click on the Manage notification page or fill in the search box, and applied the search filter on the Manage notification page. 2. Fooma Web Application shows a list of notifications, each row would display:  * Id: text * notification content: text * Action:   + Update   + Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **28.0.E1 notification list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.29 UC29 – Update notification

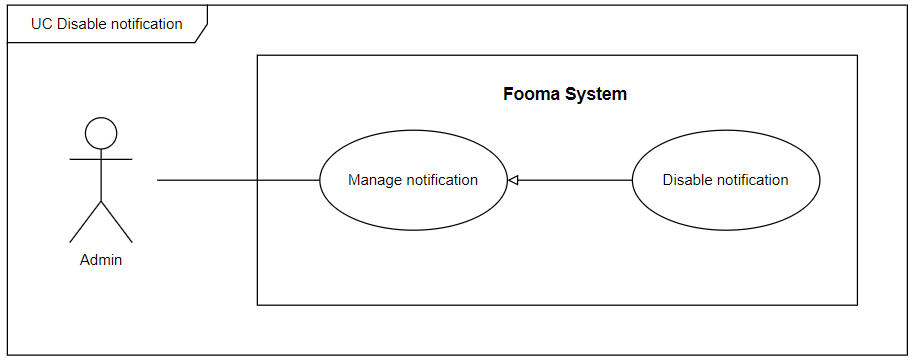


###### Figure 30. <Use case>Update notification

###### Table 32. UC29 - Update notification

| ID and Name: | **UC-29 Update notification** | | |
| --- | --- | --- | --- |
| Created By: |  |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin needs to update notification. | | |
| Trigger: | Admin tries to update notification. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Admin at manage notification page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return an “Update notification success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **29.0 Update notification**   1. Admin Click on the “Update” button at a specific notification in the notification list. 2. Fooma Web Application show update notification form:    1. notification content: text input    2. Update Button 3. Admin fills all items of the update notification form and clicks the “Update” button. 4. The Update notification form will be closed and the notification list will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **29.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **29.0.E2 Admin inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **29.0.E3 Admin inputs existed notification name**   1. The system shows an existed notification name message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.30 UC30 – Disable notification

****

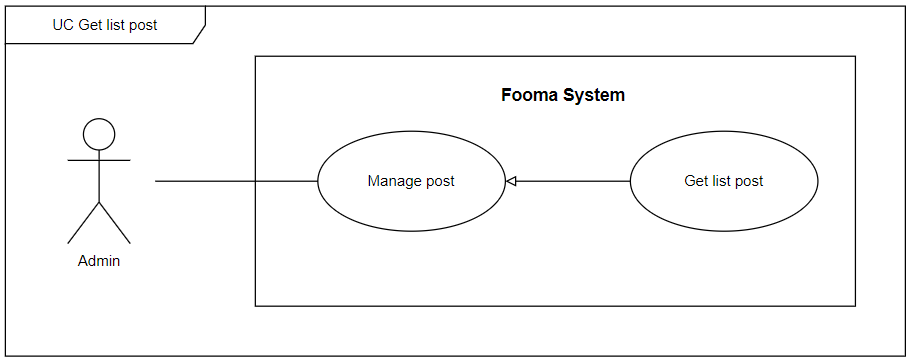
###### Figure 31. <Use case> Disable notification

###### Table 33. UC30 - Disable notification

| ID and Name: | **UC-30 Disable notification** | | |
| --- | --- | --- | --- |
| Created By: |  |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can disable the notification. | | |
| Trigger: | On the notification list, at the specific user account row, the admin clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the Admin role.  PRE-2. Admin at manage user page.  PRE-3. Required notification is displayed in the notification list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list notification view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **30.0 Disable notification**   1. Admin gets a list of notifications. 2. On the specific notification in the notification list, the admin click on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **30.0.E1 notification no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.31 UC31 – Get list post

****

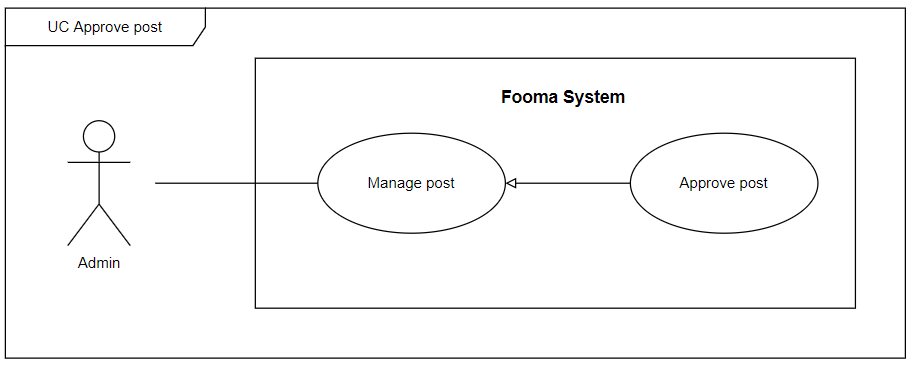
###### Figure 32. <Use case> Get list post

###### Table 34. UC31 - Get list post

| ID and Name: | **UC-31 Get list post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can view the list of posts with a search filter. | | |
| Trigger: | Admin click on Manage post page or fill some search filter and click Search on the Manage post page to Get list post. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Manger at manage post page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a list of posts matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **31.0 Get list post**   1. admin click on Manage post page or fill on the search box, applied on the search filter on the Manage post page. 2. Fooma Web Application shows a list of posts, each row of would display:  * Title: text * User name: text * Published date: date * Hashtag: text * Total comment: number * Total react: number * Action:   + Update   + Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **31.0.E1 post list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.32 UC32 – Approve post

****

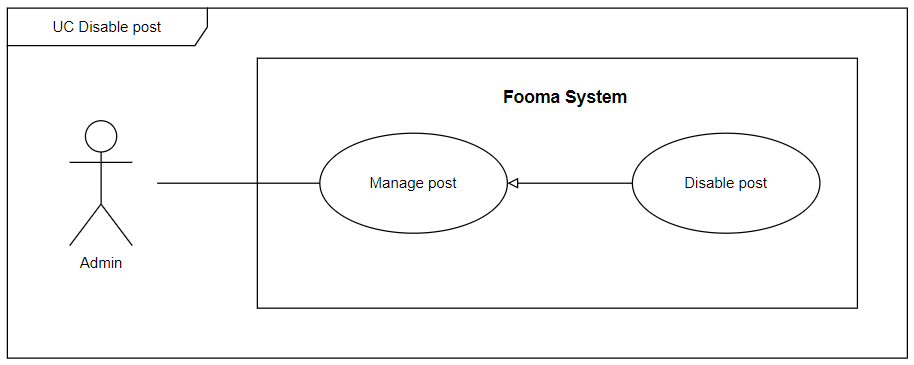
###### Figure 33. <Use case> Approve post

###### Table 35. UC32 - Approve post

| ID and Name: | **UC-32 Approve post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can Approve post. | | |
| Trigger: | On the post list, at the specific user account row, the admin clicks on Approve button. | | |
| Preconditions: | PRE-1. User has logged in with the Admin role.  PRE-2. Admin at manage user page.  PRE-3. Required post is displayed in the post list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list post view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **32.0 Approve post**   1. Admin gets a list of posts. 2. On the specific post in the post list, the admin click on Approve button. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve Approve action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **32.0.E1 post no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.33 UC33 – Disable post

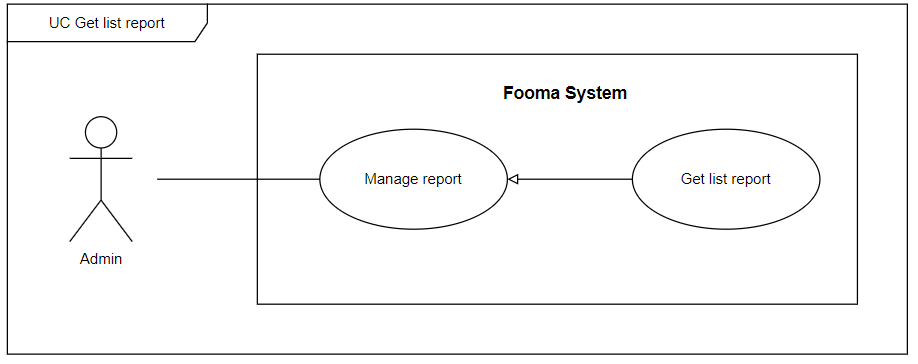
****

###### Figure 34. <Use case> Disable post

###### Table 36. UC33 - Disable post

| ID and Name: | **UC-33 Disable post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can disable post. | | |
| Trigger: | On the post list, at the specific user account row, the admin clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the Admin role.  PRE-2. Admin at manage user page.  PRE-3. Required post is displayed in the post list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list post view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **33.0 Disable post**   1. Admin gets a list of posts. 2. On the specific post in the post list, the admin click on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Admin click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **33.0.E1 post no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.34 UC34 – Get list report

****

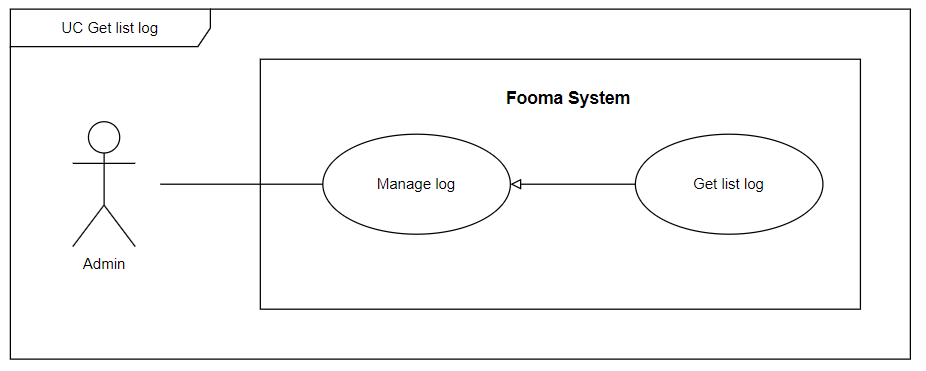
###### Figure 35. <Use case> Get list report

###### Table 37. UC34 - Get list report

| ID and Name: | **UC-34 Get list report** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can view the list of reports with a search filter. | | |
| Trigger: | Admin clicks on Manage report page or fill some search filter and click Search on the Manage report page to Get list report. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Manger at manage report page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show list reports matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **34.0 Get list report**   1. admin click on the Manage report page or fill on the search box, applied on the search filter on the Manage report page. 2. Fooma Web Application shows a list of the report, each row would display:  * Title: text * Content: text * User name: text * Create date: date | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **34.0.E1 report list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.35 UC35 – Get list log

****

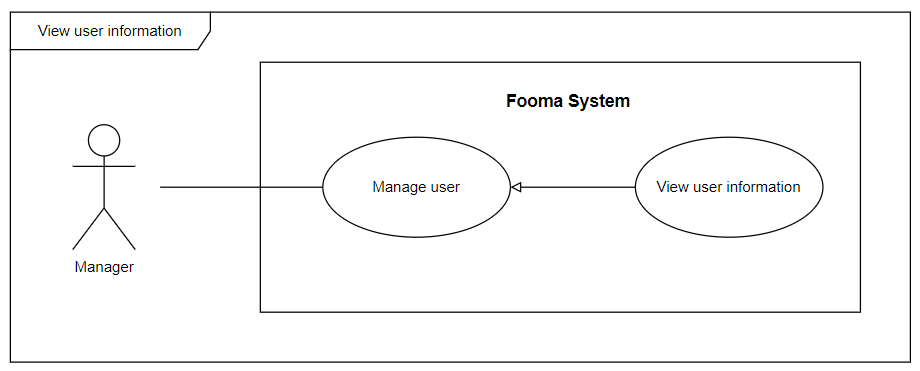
###### Figure 36. <Use case> Get list log

###### Table 38. UC35 - Get list log

| ID and Name: | **UC-35 Get list log** | | |
| --- | --- | --- | --- |
| Created By: |  |  |  |
| Primary Actor: | Admin |  |  |
| Description: | Admin can view the list of logs with a search filter. | | |
| Trigger: | Admin clicks on the Manage log page or fills some search filter and clicks Search on the Manage log page to Get list log. | | |
| Preconditions: | PRE-1. User has logged in with the admin role.  PRE-2.Manger at manage log page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show list logs matched with the search filter which is sent by the admin. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **35.0 Get list log**   1. admin click on the Manage log page or fill on the search box, applied on the search filter on the Manage log page. 2. Fooma Web Application shows a list of log, each row would display:  * Title: text * Content: text * User name: text * Create date: date | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **35.0.E1 log list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.36 UC36 – View user information

****

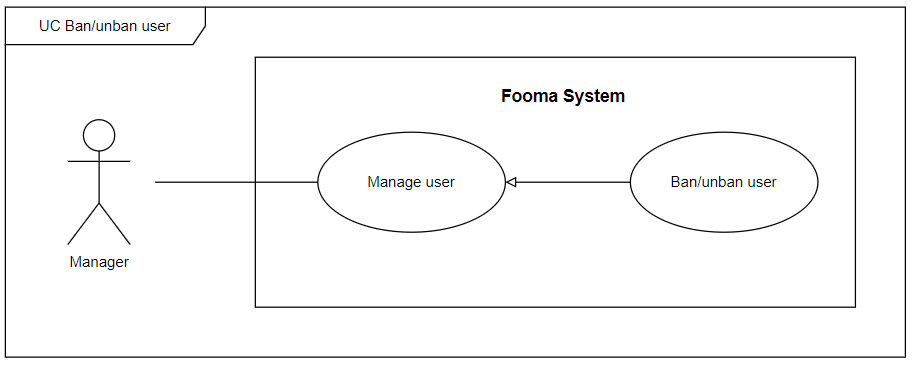
###### Figure 37. <Use case> View user information

###### Table 39. UC036 - View user information

| ID and Name: | **UC-36 View user information** | | |
| --- | --- | --- | --- |
| Created By: | Baohg |  |  |
| Primary Actor: | Manager |  |  |
| Description: | The manager can view detailed information of a user account. | | |
| Trigger: | On the user list, at the specific user account row, the Manager clicks on the View detail button. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2. Manager at manage user page.  PRE-3. Required user account displayed in the user list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show details of the user. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **36.0 Enable/Disable manager’s account**   1. The manager gets a list of users. 2. On the specific user in the user list, the Manager clicks on the Details button. 3. Fooma Web Application redirects to view details of the user. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **36.0.E1 User no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004, BR-1002, BR-1004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.37 UC37 – Ban/Unban user

****

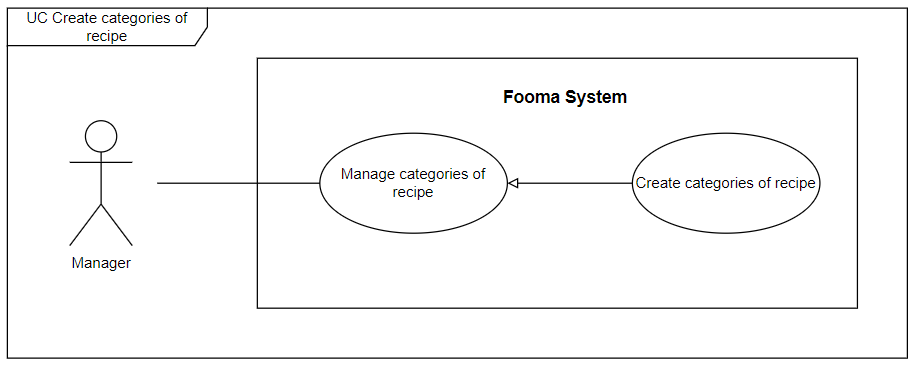
###### Figure 38. <Use case> Ban/unban user

###### Table 40. U037 - Ban/unban user

| ID and Name: | **UC-037 Ban/unban user** | | |
| --- | --- | --- | --- |
| Created By: | Baohg |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager can ban/unban user. | | |
| Trigger: | On the user list, at the specific user account row, the Manager clicks on the ban/unban button based on the user's current status. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2. Manager at manage user page.  PRE-3. Required user account displayed in the user list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list user view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **37.0 Enable/Disable manager’s account**   1. The manager gets a list of users. 2. On the specific manager in the user list, the Manager clicks on the Ban/unban button based on the user's current status. 3. Fooma Web Application will show a confirmation message. 4. Manager click Confirm to approve ban/unban action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **37.0.E1 Manager no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.38 UC38 – Create categories of recipe



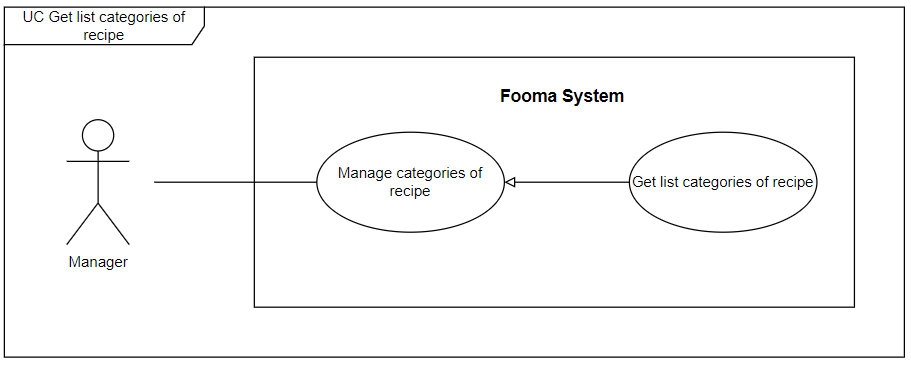
###### Figure 39. <Use case> Create categories of recipe

###### Table 41. UC38 - Create categories of recipe

| ID and Name: | **UC-38 Create categories of recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager need to create new categories of recipe. | | |
| Trigger: | Manager tries to create new categories of the recipe. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2. Manager at the manage categories of recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Create categories of recipe success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **38.0 Create categories of recipe**   1. Manager Click on the “Create categories of recipe” button on the Manage categories of recipe page. 2. Fooma Web Application shows create categories of recipe form:    1. Categories of recipe Name: text input    2. Create Button 3. The manager fills all items of the categories of recipe form and clicks the “Create” button. 4. The Create categories of recipe form will be closed. | | |
| Alternative Flows: | **38.0 categories of recipe selects the “Close” button**   1. The manager selects the “Close” button. 2. Fooma Web Application closes the create categories of recipe form and returns to the categories of recipe list view. | | |
| Exceptions: | **38.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **38.0.E2 Manager inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **38.0.E3 Manager inputs existed categories of recipe**   1. The system shows existing categories of recipe messages. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.39 UC39 – Get list categories of recipe

****

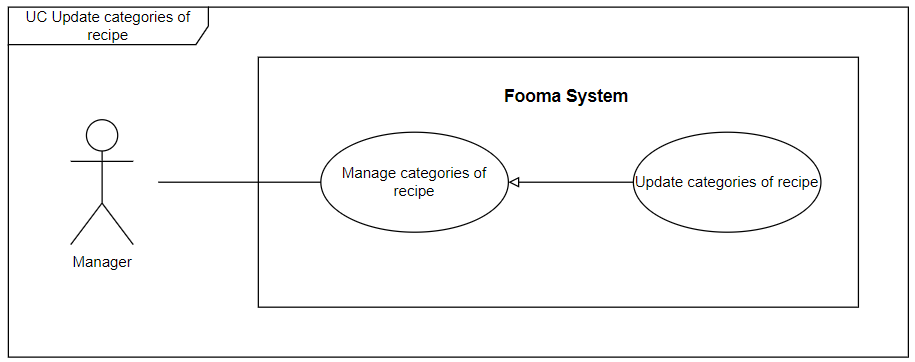
###### Figure 40. <Use case> Get list categories of recipe

###### Table 42. UC39 - Get list categories of recipe

| ID and Name: | **UC-39 Get list categories of recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | The Manager can view the list of categories of recipes with a search filter. | | |
| Trigger: | Manager clicks on Manage categories of recipe page or fill some search filter and click Search on the Manage categories of recipe page to Get list categories of recipe. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2.Manger at manage categories of recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show list categories of recipes matched with the search filter which is sent by the Manager. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **39.0 Get list categories of recipe**   1. Manager clicks on Manage categories of recipe page or fill on the search box, applied on the search filter on the Manage categories of recipe page. 2. Fooma Web Application shows a list of categories of recipe, each row would display:  * Id: text * Categories of recipe Name: text * Action:   + Update   + Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **39.0.E1 categories of recipe list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.40 UC40 – Update categories of recipe



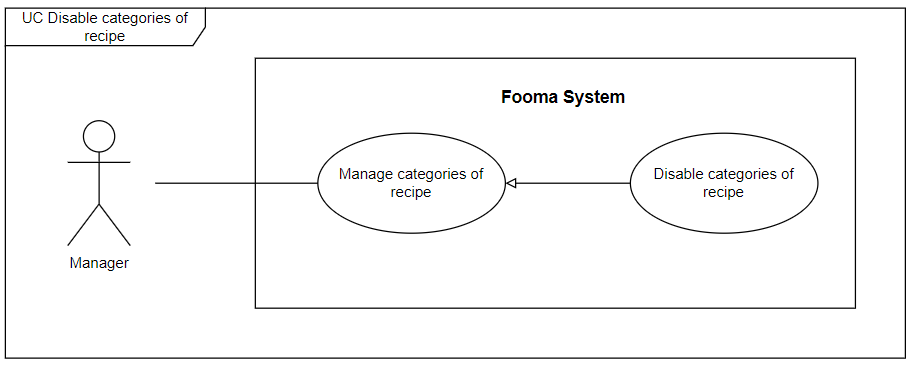
###### Figure 41. <Use case>Update categories of recipe

###### Table 43. UC40 - Update categories of recipe

| ID and Name: | **UC-40 Update categories of recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager needs to update categories of recipe. | | |
| Trigger: | The manager tries to update categories of recipe. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2.Manager at manage categories of recipe page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Update categories of recipe success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **40.0 Update categories of recipe**   1. Manager Click on the “Update” button at specific categories of recipe in the categories of recipe list. 2. Fooma Web Application show update categories of recipe form:    1. categories of recipe Name: text input    2. Update Button 3. The manager fills all items of the update categories of recipe form and clicks the “Update” button. 4. The Update categories of recipe form will be closed and the categories of recipe list will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **40.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **40.0.E2 Manager inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **40.0.E3 Manager inputs existed categories of recipe name**   1. The system shows an existed categories of recipe name messages. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.41 UC41 – Disable categories of recipe

****

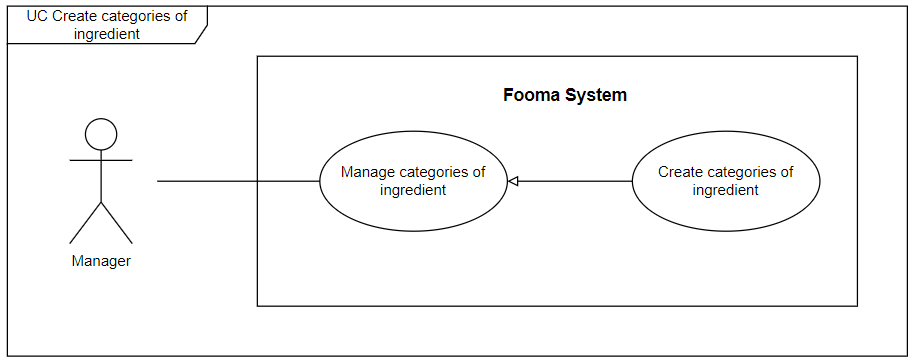
###### Figure 42. <Use case> Disable categories of recipe

###### Table 44. UC41 - Disable categories of recipe

| ID and Name: | **UC-41 Disable categories of recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager can disable categories of recipe. | | |
| Trigger: | On the categories of recipe list, at the specific user account row, the Manager clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2. Manager at manage user page.  PRE-3. Required categories of the recipe are displayed in the categories of recipe list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list categories of recipe view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **41.0 Disable categories of recipe**   1. The manager gets a list of categories of the recipe. 2. On the specific categories of recipe in the categories of recipe list, the Manager clicks on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Manager click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **41.0.E1 categories of recipe no longer exist**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.42 UC42 – Create categories of ingredient



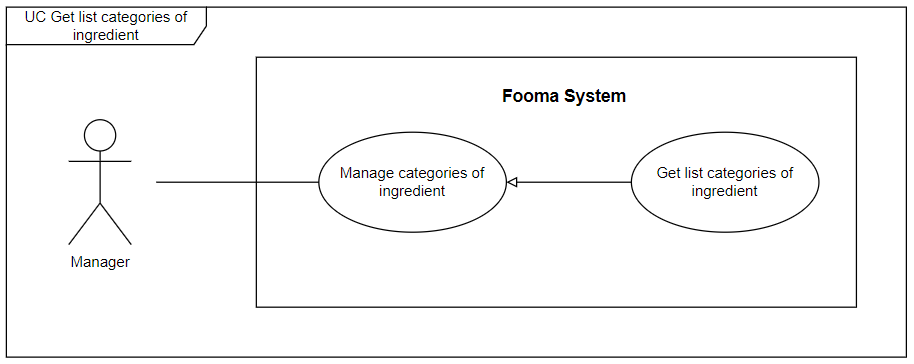
###### Figure 43. <Use case> Create categories of ingredient

###### Table 45. UC42 - Create categories of ingredient

| ID and Name: | **UC-42 Create categories of ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | The manager needs to create new categories of ingredient. | | |
| Trigger: | The manager tries to create new categories of ingredient. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2. Manager at the manage categories of ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Create categories of ingredient success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **42.0 Create categories of ingredient**   1. Manager Click on the “Create categories of ingredient” button on the Manage categories of ingredient page. 2. Fooma Web Application shows create categories of ingredient form:    1. categories of ingredient Name: text input    2. Create Button 3. The manager fills all items of the categories of ingredient form and clicks the “Create” button. 4. The Create categories of ingredient form will be closed. | | |
| Alternative Flows: | **42.0 categories of ingredient selects the “Close” button**   1. The manager selects the “Close” button. 2. Fooma Web Application closes the create categories of ingredient form and returns to the categories of ingredient list view. | | |
| Exceptions: | **42.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **42.0.E2 Manager inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **42.0.E3 Manager inputs existed categories of ingredient**   1. The system shows existing categories of ingredient messages. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.43 UC43 – Get list categories of ingredient

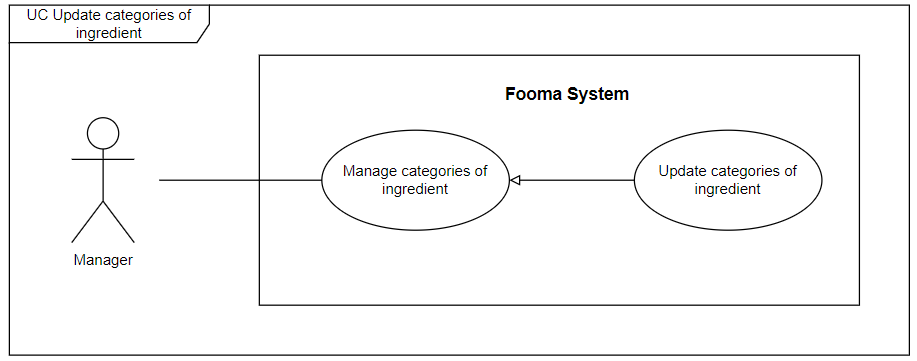
****

###### Figure 44. <Use case> Get list categories of ingredient

###### Table 46. UC43 - Get list categories of ingredient

| ID and Name: | **UC-43 Get list categories of ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | The manager can view the list of categories of ingredients with a search filter. | | |
| Trigger: | Manager clicks on Manage categories of ingredient page or fill some search filter and click Search on the Manage categories of ingredient page to Get list categories of ingredient. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2.Manger at manage categories of ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show list categories of ingredients matched with the search filter which is sent by the Manager. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **43.0 Get list categories of ingredient**   1. The manager clicks on the Manage categories of ingredient page or fill on the search box, applied on the search filter on the Manage categories of ingredient page. 2. Fooma Web Application shows a list of categories of ingredient, each row would display:  * Id: text * categories of ingredient Name: text * Action:   + Update   + Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **43.0.E1 categories of ingredient list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.44 UC44 – Update categories of ingredient



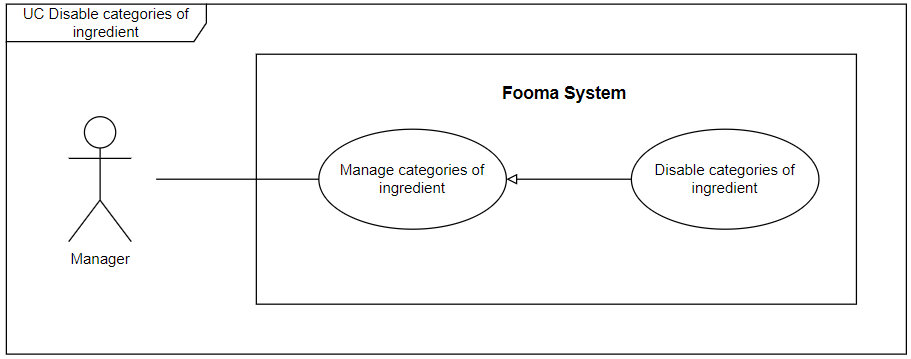
###### Figure 45. <Use case>Update categories of ingredient

###### Table 47. UC44 - Update categories of ingredient

| ID and Name: | **UC-44 Update categories of ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | The Manager needs to update categories of ingredient. | | |
| Trigger: | Manager tries to update categories of ingredient. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2.Manager at manage categories of ingredient page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web Application will return the “Update categories of ingredient success” message. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **44.0 Update categories of ingredient**   1. Manager Click on the “Update” button at a specific categories of ingredient in the categories of ingredient list. 2. Fooma Web Application show update categories of ingredient form:    1. categories of ingredient Name: text input    2. Update Button 3. The Manager fills all items of the update categories of ingredient form and clicks the “Update” button. 4. The Update categories of ingredient form will be closed and the categories of ingredient list will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **44.E1 Missing of required fields**   1. The System shows a message asking the user to fill the required fields again.   **44.0.E2 Manager inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields.   **44.0.E3 Manager inputs existed categories of ingredient name**   1. The system shows an existed categories of ingredient name messages. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.45 UC45 – Disable categories of ingredient

****

###### Figure 46. <Use case> Disable categories of ingredient

###### Table 48. UC45 - Disable categories of ingredient

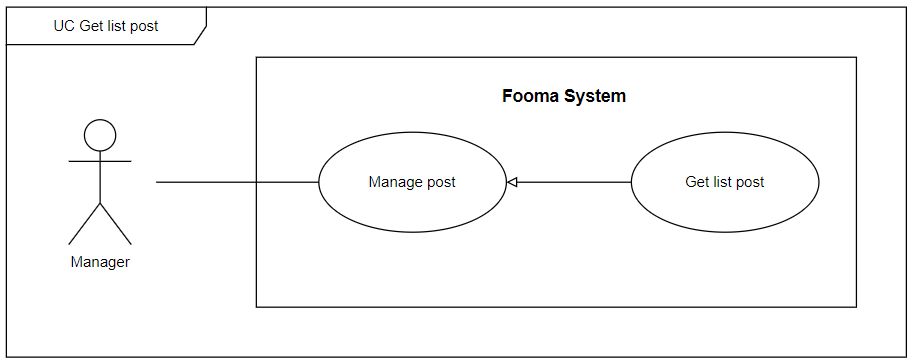
| ID and Name: | **UC-45 Disable categories of ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager can disable categories of ingredient. | | |
| Trigger: | On the categories of ingredient list, at the specific user account row, the Manager clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2. Manager at manage user page.  PRE-3. Required categories of ingredient are displayed in the categories of ingredient list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list categories of ingredient view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **45.0 Disable categories of ingredient**   1. The manager gets a list of categories of ingredient. 2. On the specific categories of ingredient in the categories of ingredient list, the Manager click on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Manager click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **45.0.E1 categories of ingredient no longer exist**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 

#### 

#### 2.2.46 UC46 – Get list post

****

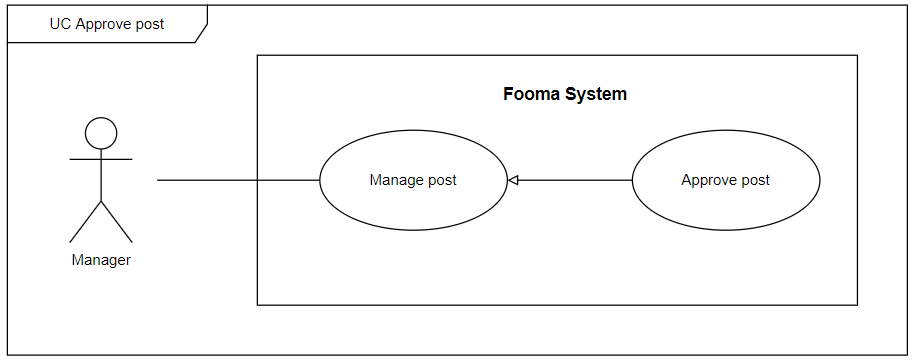
###### Figure 47. <Use case> Get list post

###### Table 49. UC46 - Get list post

| ID and Name: | **UC-46 Get list post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Manager |  |  |
| Description: | The manager can view the list of posts with a search filter. | | |
| Trigger: | The manager clicks on the Manage post page or fills some search filter and clicks Search on the Manage post page to Get list post. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2.Manger at manage post page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a list of posts matched with the search filter which is sent by the Manager. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **46.0 Get list post**   1. The manager clicks on the Manage post page or fills in the search box, and applied the search filter on the Manage post page. 2. Fooma Web Application shows a list of posts, each row would display:  * Title: text * User name: text * Published date: date * Hashtag: text * Total comment: number * Total react: number * Action:   + Update   + Disable | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **46.0.E1 post list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.47 UC47 – Approve post

****

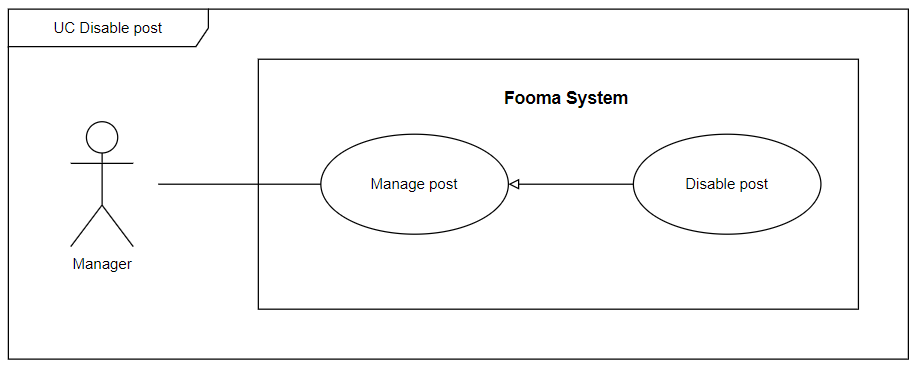
###### Figure 48. <Use case> Approve post

###### Table 50. UC47 - Approve post

| ID and Name: | **UC-47 Approve post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager can Approve post. | | |
| Trigger: | On the post list, at the specific user account row, the Manager clicks on Approve button. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2. Manager at manage user page.  PRE-3. Required post is displayed in the post list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list post view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **47.0 Approve post**   1. The manager gets a list of posts. 2. On the specific post in the post list, the Manager clicks on Approve button. 3. Fooma Web Application will show a confirmation message. 4. Manager click Confirm to approve Approve action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **47.0.E1 post no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.48 UC48 – Disable post

****

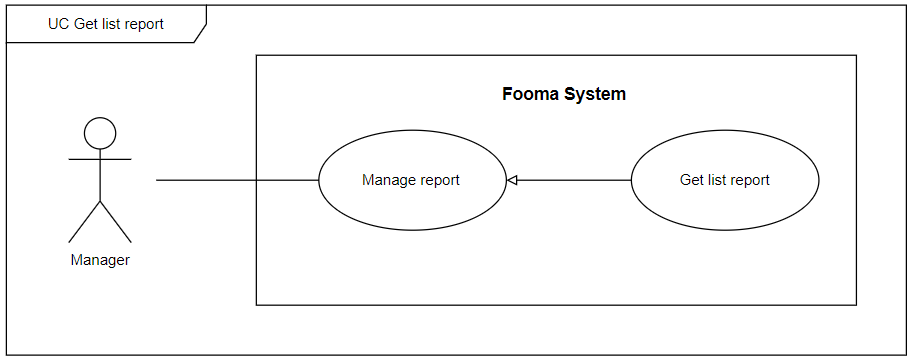
###### Figure 49. <Use case> Disable post

###### Table 51. UC48 - Disable post

| ID and Name: | **UC-48 Disable post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Manager |  |  |
| Description: | Manager can disable post. | | |
| Trigger: | On the post list, at the specific user account row, the Manager clicks on Disable button. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2. Manager at manage user page.  PRE-3. Required post is displayed in the post list. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show a success message and update the list post view. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **48.0 Disable post**   1. Manager gets a list of post. 2. On the specific post in the post list, the Manager click on Disable button. 3. Fooma Web Application will show a confirmation message. 4. Manager click Confirm to approve disable action. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **48.0.E1 post no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.49 UC49 – Get list report

****

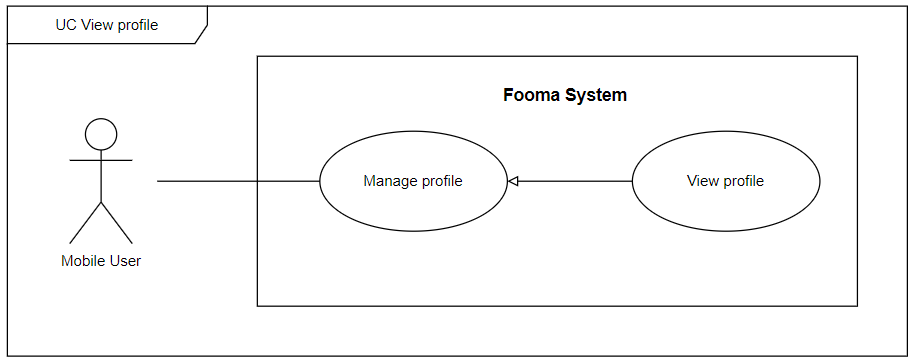
###### Figure 50. <Use case> Get list report

###### Table 52. UC49 - Get list report

| ID and Name: | **UC-49 Get list report** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Manager |  |  |
| Description: | The manager can view the list of reports with a search filter. | | |
| Trigger: | The manager clicks on the Manage report page or fills some search filter and clicks Search on the Manage report page to Get a list report. | | |
| Preconditions: | PRE-1. User has logged in with the Manager role.  PRE-2.Manger at manage report page. | | |
| Post-conditions: | POST-1:   * Success: Fooma Web application will show list reports matched with the search filter which is sent by the Manager. * Fail: Fooma Web Application shows an error message. | | |
| Normal Flow: | **49.0 Get list report**   1. The manager clicks on the Manage report page or fills on the search box, and applied the search filter on the Manage report page. 2. Fooma Web Application shows a list of the report, each row would display:  * Title: text * Content: text * User name: text * Create date: date | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **49.0.E1 report list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.50 UC50 – View profile

****

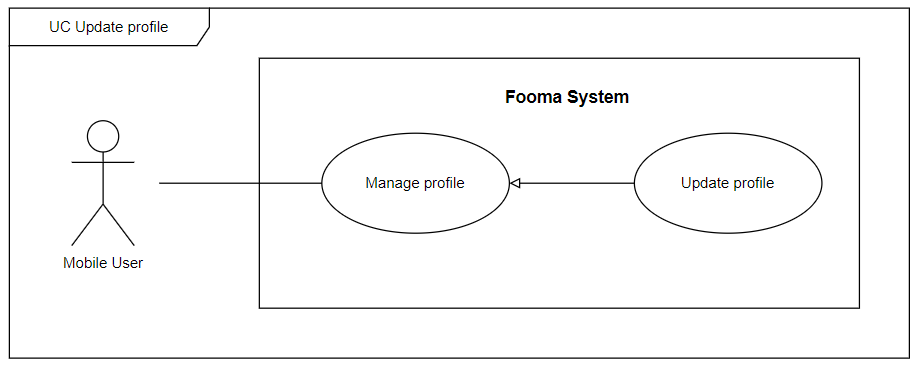
###### Figure 51. <Use case> View profile

###### Table 53. UC50 - View profile

| ID and Name: | **UC-50 View profile** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can view detailed of their profile. | | |
| Trigger: | On the main screen Mobile user click on the profile icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at User profile screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will show details of the user. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **50.0 View profile**   1. Mobile user at the main screen. 2. Mobile user clicks on the profile icon. 3. Fooma Mobile Application redirects to view details of their profile. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **50.0.E1 User no longer exists**   1. The system will show an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.51 UC51 – Update profile

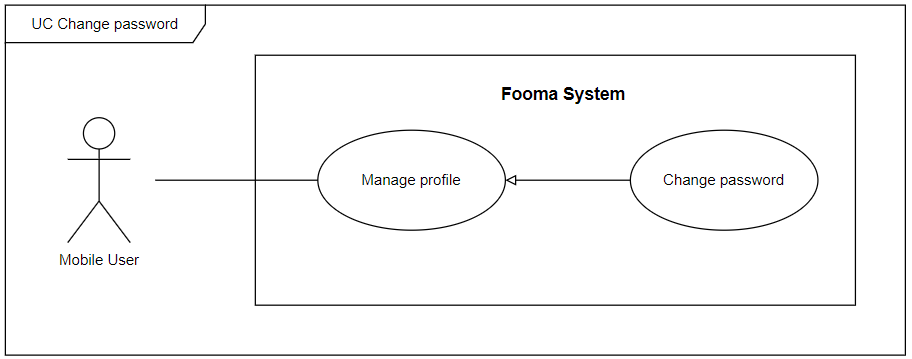
****

###### Figure 52. <Use case> Update profile

###### Table 54. UC51 - Update profile

| ID and Name: | **UC-51 Update profile** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can edit their profile. | | |
| Trigger: | On the Profile screen Mobile user click on “Chỉnh sửa hồ sơ” button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at User profile screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will update the user profile. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **51.0 Update profile**   1. Mobile user at profile screen. 2. Mobile user click on “Chỉnh sửa hồ sơ” button. 3. Fooma Mobile Application redirects to edit user profile screen with some fields:    1. Image: file input    2. Name: text input    3. bio: text input    4. Update button | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **51.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **51.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.52 UC52 – Change password

****

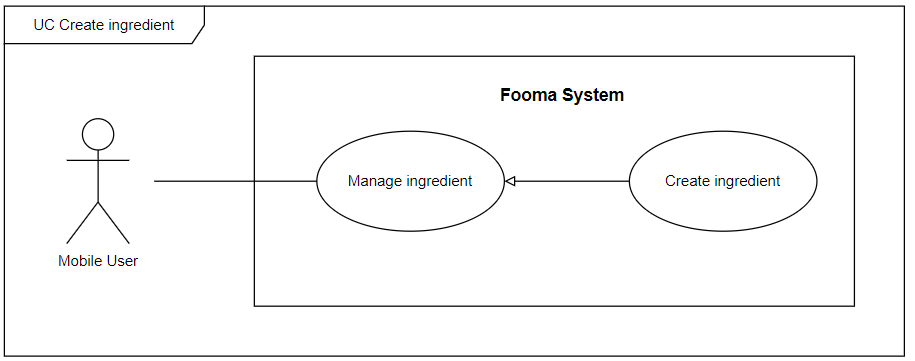
###### Figure 53. <Use case> Change password

###### Table 55. UC52 - Change password

| ID and Name: | **UC-52 Change password** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can edit their profile. | | |
| Trigger: | On the Login screen, the user clicks on the “Quên mật khẩu” button. | | |
| Preconditions: | PRE-1.  Mobile user is at the Login screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will update the user password. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **52.0 Change password**   1. Mobile user at the login screen. 2. Mobile user click on “Quên mật khẩu” button. 3. Fooma Mobile Application redirects to an Input email screen with field:    1. Email: text input 4. Fooma system will send code to the user’s email. 5. Fooma mobile application will direct to Input code screen. 6. User type code then Fooma mobile application will direct to Change password screen with some fields:    1. New password: password input    2. Re-enter new password: password input    3. Change the password button. 7. After Mobile user clicks change password button, the new password of the user would be set. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **52.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **52.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.53 UC53 – Create Ingredient

****

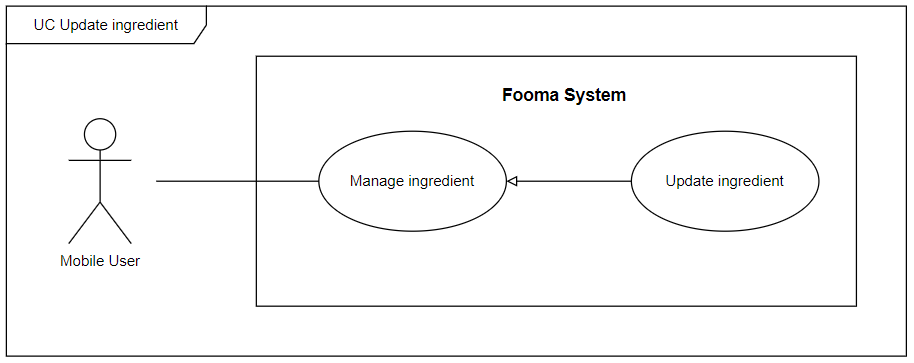
###### Figure 54. <Use case> Create ingredient

###### Table 56. UC53 - Create ingredient

| ID and Name: | **UC-53 Create ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile users can add a new ingredient. | | |
| Trigger: | On the Manage ingredient screen, the Mobile user clicks on Create ingredient button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at Manage ingredient screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will add a new ingredient to the user’s ingredient list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **53.0 Create ingredient**   1. Mobile user at Manage ingredient screen. 2. Mobile User click on Create button. 3. Fooma Mobile Application redirects to Create ingredient screen with some fields:    1. Name: text input    2. Quantity: number input    3. Unit: text input    4. Expired Date: date input    5. Create button 4. Mobile user filled all fields and click Create button, the new ingredient will be added to the user’s ingredient list. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **53.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **53.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.54 UC54 – Update Ingredient

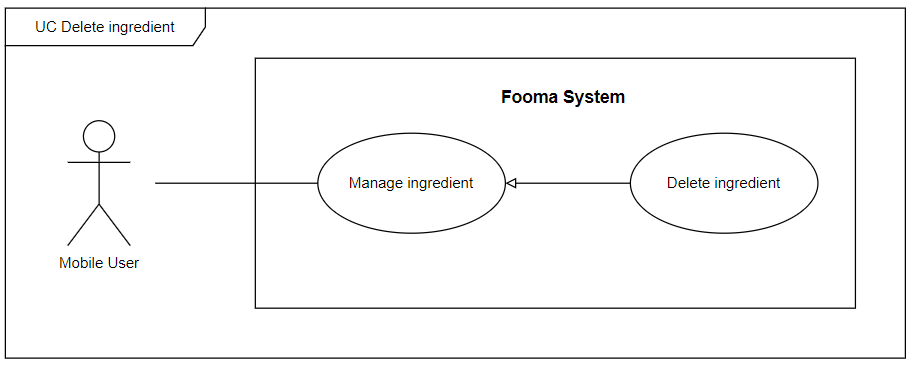
****

###### Figure 55. <Use case> Update Ingredient

###### Table 57. UC54 - Update Ingredient

| ID and Name: | **UC-54 Update ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can update ingredient. | | |
| Trigger: | On the Manage ingredient screen, at the ingredient list, the Mobile user clicks on the Update Ingredient button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at Manage ingredient screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will update ingredients and refresh the user’s ingredient list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **54.0 Update ingredient**   1. Mobile user at Manage ingredient screen. 2. On user ingredient list, at a specific ingredient, Mobile User clicks on the Update button. 3. Fooma Mobile Application redirects to Update Ingredient screen with some fields:    1. Name: text input    2. Quantity: number input    3. Unit: text input    4. Update button 4. Mobile user filled all fields and click the Update button, ingredients would be updated and the user ingredient list will be refreshed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **54.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **54.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.55 UC55 – Delete Ingredient

****

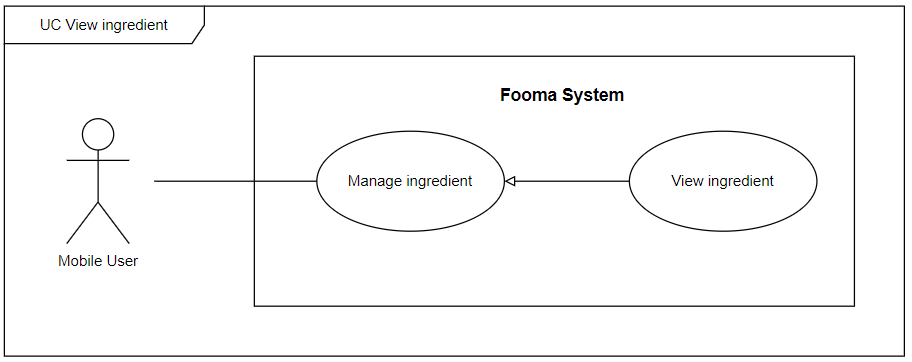
###### Figure 56. <Use case> Delete Ingredient

###### Table 58. UC55 - Delete Ingredient

| ID and Name: | **UC-55 Delete ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can delete ingredient. | | |
| Trigger: | On the Manage ingredient screen, at the ingredient list, the Mobile user swipe to left and click on the Delete Ingredient button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at Manage ingredient screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will Delete Ingredient and refresh the user’s ingredient list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **55.0 Delete ingredient**   1. Mobile user at Manage ingredient screen. 2. On the user ingredient list, at a specific ingredient, Mobile User swipes to left and clicks on the Delete button. 3. Mobile user filled all field and click Delete button, the ingredient would be updated and the user ingredient list will be refreshed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **55.0.E1 Ingredient no longer exists**   1. The system shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.56 UC56 – View Ingredient

****

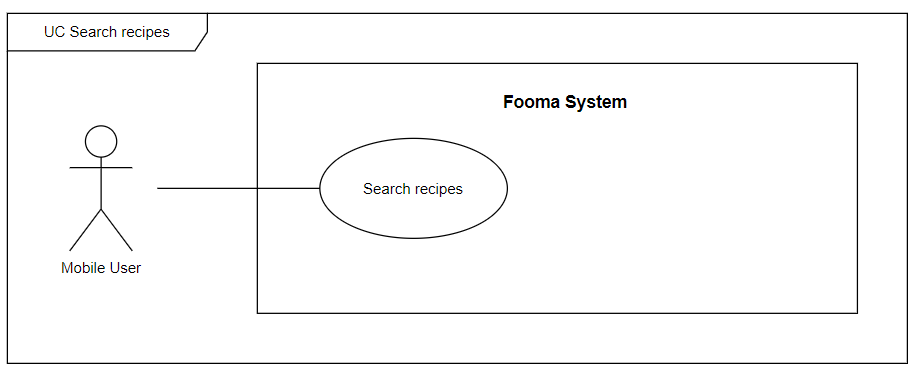
###### Figure 57. <Use case> View Ingredient

###### Table 59. UC56 - View Ingredient

| ID and Name: | **UC-56 View ingredient** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can view ingredient list. | | |
| Trigger: | On the Main screen, the user swipe to the right, and the sidebar will be shown, the user clicks on Manage ingredient. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will show the user’s ingredient list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **56.0 View ingredient**   1. On the Main screen, the user swipe to the right, and the sidebar will be shown, the user clicks on Manage ingredient. 2. User ingredient list will be shown. | | |
| Alternative Flows: | **56.0 View ingredient**   1. On the manage ingredient screen, the user typed the search box or applied some search filter, and the result ingredient list which are match will search filters will be shown. | | |
| Exceptions: | **56.0.E1 Ingredient list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.57 UC57 – Search recipe

****

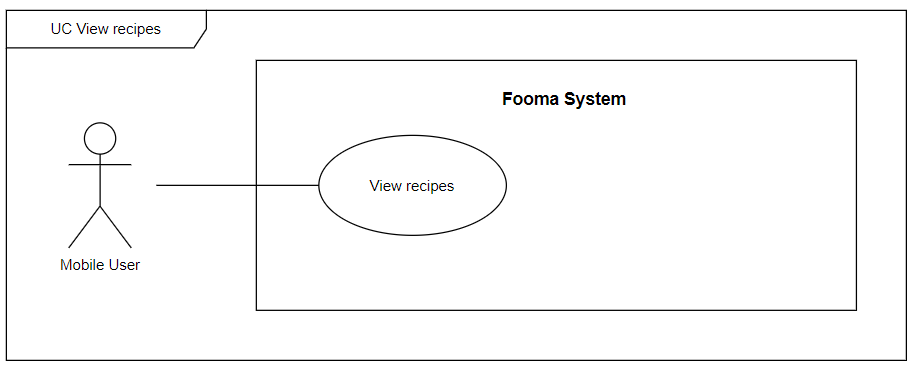
###### Figure 58. <Use case> Search recipe

###### Table 60. UC57 - Search recipe

| ID and Name: | **UC-57 Search recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can Search recipe. | | |
| Trigger: | On the Main screen, the Mobile user applied search filters. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will return the recipe list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **57.0 Search recipe**   1. On the Main screen, the Mobile user clicks the recipe icon. 2. Mobile users applied some search filters. 3. List recipes that are matched with the search filter will be shown. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **57.0.E1 Recipe list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.58 UC58 – View recipes

****

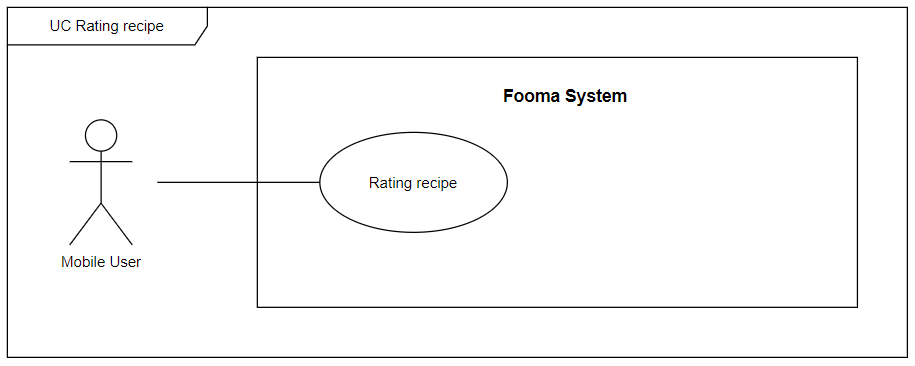
###### Figure 59. <Use case> View recipes

###### Table 61. UC58 - View recipes

| ID and Name: | **UC-58 View recipes** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can View recipes. | | |
| Trigger: | Mobile user clicks on Dashboard icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will return the recipe list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **58.0 View recipes**   1. Mobile users click on the Dashboard icon. 2. List recipes will be shown. | | |
| Alternative Flows: | **58.0 View recipes**   1. Mobile users click on the Dashboard icon. 2. Mobile users applied some search filters. 3. List recipes that are matched with the search filter will be shown. | | |
| Exceptions: | **58.0.E1 Recipe list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.59 UC59 – Rating recipe

****

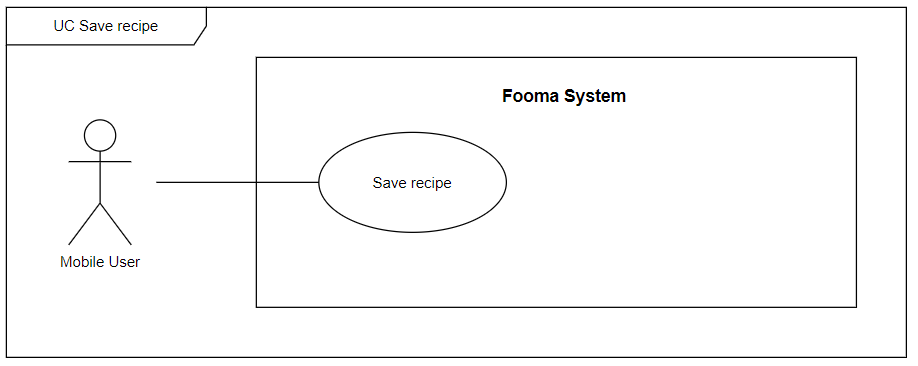
###### Figure 60. <Use case> Rating recipe

###### Table 61. UC59 - Rating recipe

| ID and Name: | **UC-59 Rating recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile users can Rating recipe. | | |
| Trigger: | On the recipe list, on the specific recipe, the Mobile user clicks on the Rating icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will add ratings and return the recipe list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **59.0 Rating recipe**   1. On the recipe list, on the specific recipe, Mobile users click on the Rating icon. 2. A rating recipe popup will be shown:    1. Content: text input    2. Rating: number input    3. Rating button 3. Mobile users click on rating and the user’s rating will be added. 4. Rating recipe popup will close. 5. List repice will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **59.0.E1 Recipe no longer exists**   1. The system shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.60 UC60 – Save recipe

****

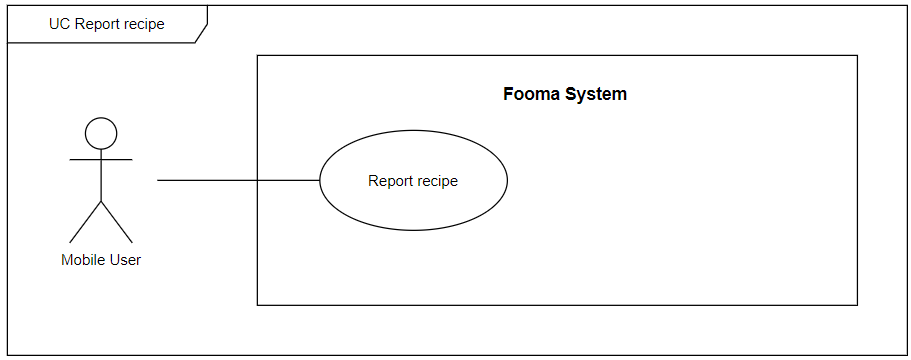
###### Figure 61. <Use case> Save recipe

###### Table 62. UC60 - Save recipe

| ID and Name: | **UC-60 Save recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can Save recipe. | | |
| Trigger: | On the recipe list, on the specific recipe, the Mobile user clicks on the Save icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will save recipes and return the recipe list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **60.0 Save recipe**   1. On the recipe list, on the specific recipe, the Mobile user clicks on the Save icon. 2. Recipe will be added to the user’s saved recipe list 3. List recipe will be updated view (add saved icon). | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **60.0.E1 Recipe no longer exists**   1. The system shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.61 UC61 – Report recipe

****

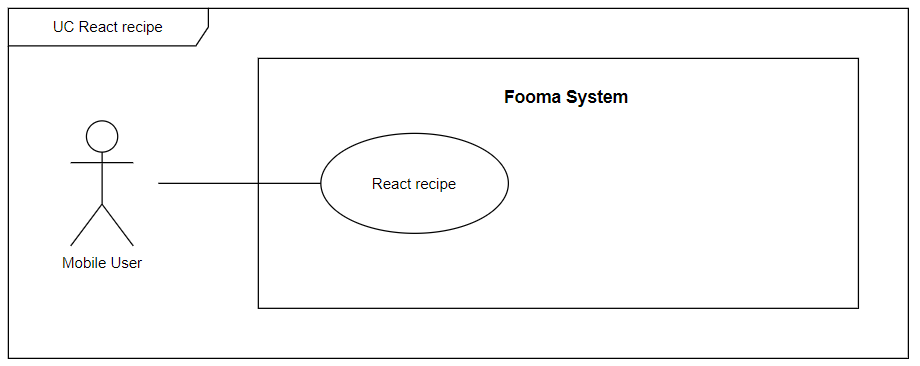
###### Figure 62. <Use case> Report recipe

###### Table 63. UC61 - Report recipe

| ID and Name: | **UC-61 Report recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can Report recipe. | | |
| Trigger: | On the recipe list, on the specific recipe, the Mobile user clicks on the Report icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will add the user’s report and return the recipe list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **61.0 Report recipe**   1. On the recipe list, on the specific recipe, the Mobile user clicks on the Report icon. 2. A Report recipe popup will be shown:    1. Title: text input    2. Content: text input    3. Report button 3. After the Mobile user clicks on the Report button, the report will be submitted. 4. The report recipe popup will be closed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.62 UC62 – React recipe

****

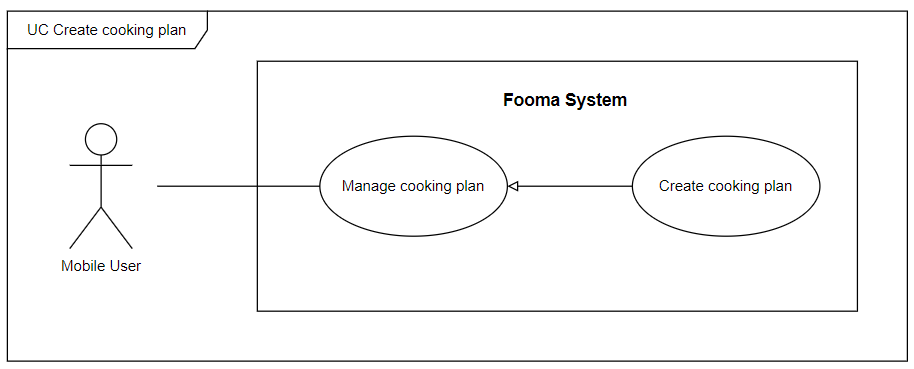
###### Figure 63. <Use case> React recipe

###### Table 64. UC62 - React recipe

| ID and Name: | **UC-62 React recipe** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can React recipe. | | |
| Trigger: | On the recipe list, on the specific recipe, the Mobile user clicks on React icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will React recipe. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **62.0 React recipe**   1. On the recipe list, on the specific recipe, the Mobile user clicks on the Save icon. 2. List recipe will be updated view (add reacted icon). | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **62.0.E1 Recipe no longer exists**   1. The system shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.63 UC63 – Create cooking plan

****

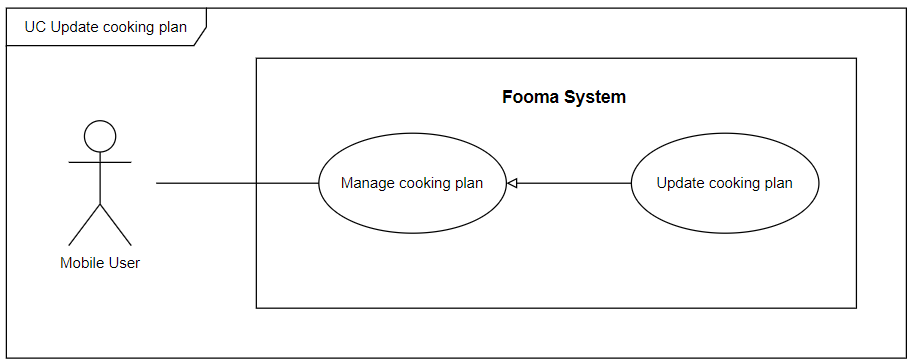
###### Figure 64. <Use case> Create cooking plan

###### Table 66. UC63 - Create cooking plan

| ID and Name: | **UC-63 Create cooking plan** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can add the new cooking plan. | | |
| Trigger: | On the Manage cooking plan screen show a calendar, the Mobile user clicks on a specific day and clicks on Create cooking plan button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at Manage cooking plan screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will add a new cooking plan to the user’s cooking plan list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **63.0 Create cooking plan**   1. Mobile user at Manage cooking plan screen. 2. Mobile User clicks on Create button. 3. Fooma Mobile Application redirects to Create cooking plan screen with some fields:    1. Name: text input    2. Quantity: number input    3. Unit: text input    4. Create button 4. Mobile users filled all fields and click Create button, the new cooking plan will be added to user’s cooking plan list. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **63.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **63.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.64 UC64 – Update cooking plan

****

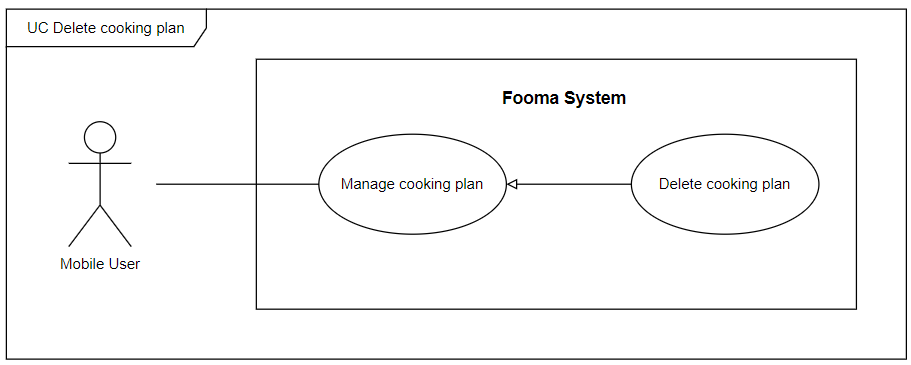
###### Figure 65. <Use case> Update cooking plan

###### Table 67. UC64 - Update cooking plan

| ID and Name: | **UC-64 Update cooking plan** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can update cooking plan. | | |
| Trigger: | On the Manage cooking plan screen, at the cooking plan list, the Mobile user clicks on the Update cooking plan button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at Manage cooking plan screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will update the cooking plan and refresh the user’s cooking plan list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **64.0 Update cooking plan**   1. Mobile user at Manage cooking plan screen. 2. On the user cooking plan list, at the specific cooking plan, Mobile User clicks on the Update button. 3. Fooma Mobile Application redirects to Update cooking plan screen with some fields:    1. Name: text input    2. Quantity: number input    3. Unit: text input    4. Update button 4. Mobile user filled all fields and click the Update button, the cooking plan would be updated and user cooking plan list will be refreshed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **64.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **64.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.65 UC65 – Delete cooking plan

****

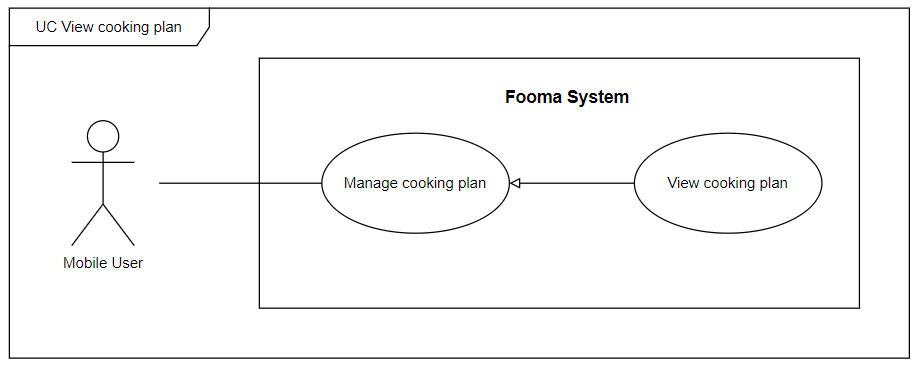
###### Figure 66. <Use case> Delete cooking plan

###### Table 68. UC65 - Delete cooking plan

| ID and Name: | **UC-65 Delete cooking plan** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can delete the cooking plan. | | |
| Trigger: | On the Manage cooking plan screen, at the cooking plan list, the Mobile user swipe to left and click on the Delete cooking plan button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at Manage cooking plan screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will Delete the cooking plan and refresh the user’s cooking plan list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **65.0 Delete cooking plan**   1. Mobile user at Manage cooking plan screen. 2. On the user cooking plan list, at the specific cooking plan, Mobile User swipes to left and clicks on the Delete button. 3. Mobile user filled all fields and click the Delete button, the cooking plan would be updated and the user cooking plan list will be refreshed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **65.0.E1 cooking plan no longer exist**   1. The system shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.66 UC66 – View cooking plan

****

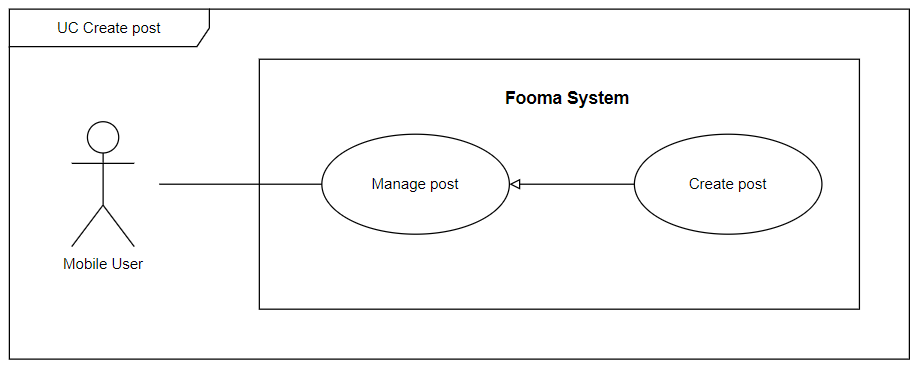
###### Figure 67. <Use case> View cooking plan

###### Table 69. UC66 - View cooking plan

| ID and Name: | **UC-66 View cooking plan** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can view cooking plan list. | | |
| Trigger: | On the Main screen, the user swipe to the right, and the sidebar will be shown, the user clicks on Manage cooking plan. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will show the user’s cooking plan list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **66.0 View cooking plan**   1. On the Main screen, the user swipe to the right, and the sidebar will be shown, the user clicks on Manage cooking plan. 2. User cooking plan list will be shown. | | |
| Alternative Flows: | **66.0 View cooking plan**   1. On the manage cooking plan screen, the user typed a search box or applied some search filter, and the result cooking plan list which are match will search filters will be shown. | | |
| Exceptions: | **66.0.E1 cooking list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.67 UC67 – Create post

****

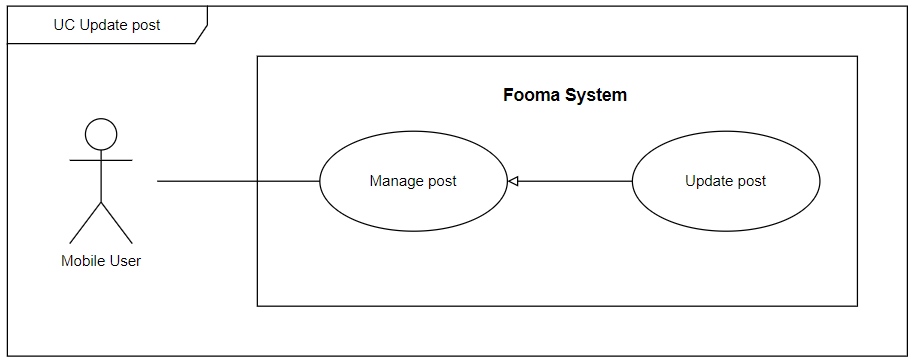
###### Figure 68. <Use case> Create post

###### Table 70. UC67 - Create post

| ID and Name: | **UC-67 Create post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile users can add a new post. | | |
| Trigger: | On the Post screen, the Mobile user clicks on a plus icon on the right downside, specific day, and clicks on Create post button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at the Manage post screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will add a new post to the user’s post list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **67.0 Create post**   1. Mobile user at Manage post screen. 2. Mobile User clicks on Create button. 3. Fooma Mobile Application redirects to Create post screen with some fields:    1. Name: text input    2. Quantity: number input    3. Unit: text input    4. Create button 4. Mobile user filled all fields and click Create button, a new post will be added to the user’s post list. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **67.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **67.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.68 UC68 – Update post

****

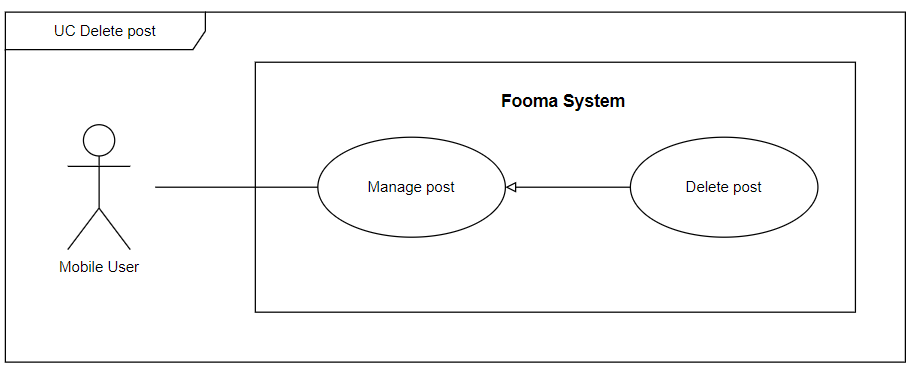
###### Figure 69. <Use case> Update post

###### Table 71. UC68 - Update post

| ID and Name: | **UC-68 Update post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can update post. | | |
| Trigger: | On the Manage post screen, at the post list, the Mobile user clicks on the Update post button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at the Manage post screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will update posts and refresh the user’s post lists. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **68.0 Update post**   1. Mobile user at Manage post screen. 2. At the user post list, at the specific post, Mobile User clicks on the Update button. 3. Fooma Mobile Application redirects to Update post screen with some fields:    1. Name: text input    2. Quantity: number input    3. Unit: text input    4. Update button 4. Mobile user filled all fields and click the Update button, the post would be updated and the user post list will be refreshed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **68.0.E1 Missing of required fields**   1. The system shows a message asking the user to fill the required fields again.   **68.0.E2 User inputs at least 1 invalid field**   1. The system shows a message asking the user to correct the fields. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.69 UC69 – Delete post

****

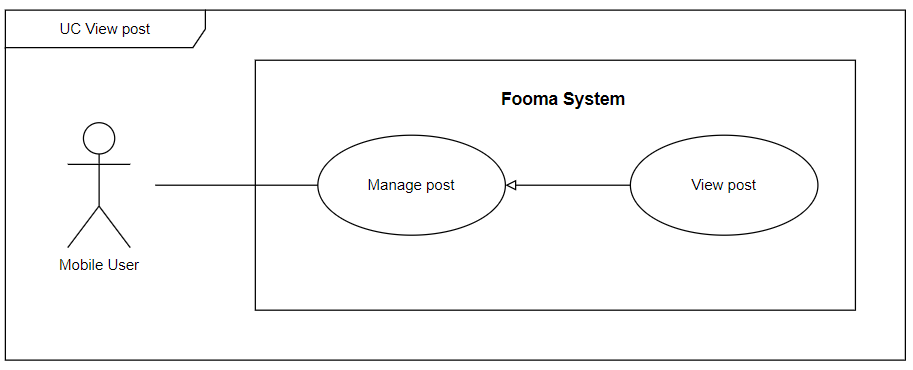
###### Figure 71. <Use case> Delete post

###### Table 73 . UC69 - Delete post

| ID and Name: | **UC-69 Delete post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can delete the post. | | |
| Trigger: | On the Manage post screen, at the post list, the Mobile user swipe to left and click on the Delete post button. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role.  PRE-2. Mobile user is at the Manage post screen. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will Delete post and refresh the user’s post list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **69.0 Delete post**   1. Mobile user at Manage post screen. 2. On the user post list, at the specific post, Mobile User swipes to left and clicks on the Delete button. 3. Mobile user filled all fields and click the Delete button, the post would be updated and the user post list will be refreshed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **69.0.E1 post no longer exists**   1. The System shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.70 UC70 – View post

****

###### Figure 71. <Use case> View post

###### Table 73. UC70 - View post

| ID and Name: | **UC-70 View post** | | |
| --- | --- | --- | --- |
| Created By: | Truongnhh |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can view post list. | | |
| Trigger: | On the Main screen, the user swipe to the right, and the sidebar will be shown, the user clicks on Manage post. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will show the user’s post list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **70.0 View post**   1. On the Main screen, the user swipe to the right, and the sidebar will be shown, the user clicks on Manage post. 2. The user post list will be shown. | | |
| Alternative Flows: | **70.0 View post**   1. On the manage post screen, the user typed a search box or applied some search filter, and the result post list which are match with search filters will be shown. | | |
| Exceptions: | **70.0.E1 cooking list is empty**   1. The system shows an empty message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

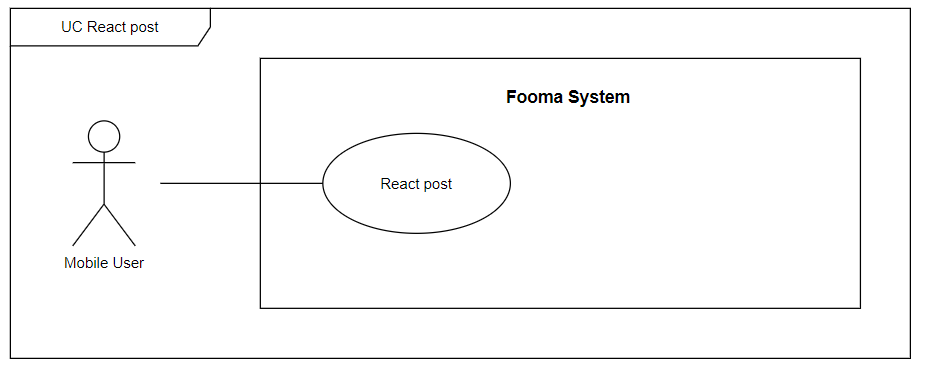
#### 

#### 

#### 

#### 

#### 2.2.71 UC71 – React post

****

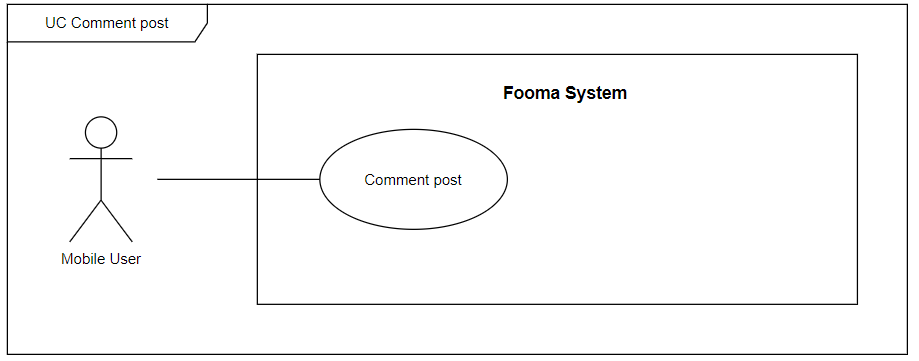
###### Figure 72. <Use case> React post

###### Table 74. UC71 - React post

| ID and Name: | **UC-71 React post** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can React post. | | |
| Trigger: | On the post list, on the specific post, the Mobile user clicks on React icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will React post. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **71.0 React post**   1. On the post list, on the specific post, the Mobile user clicks on the React icon. 2. List post will be updated view (add reacted icon). | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **71.0.E1 post no longer exists**   1. The system shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.72 UC72 – Comment post

****

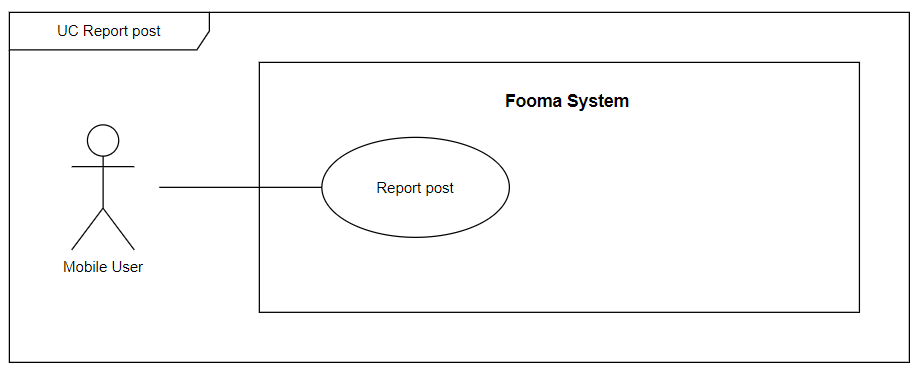
###### Figure 73. <Use case> Comment post

###### Table 75. UC72 - Comment post

| ID and Name: | **UC-72 Comment post** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile users can Comment on posts. | | |
| Trigger: | On the post list, on the specific post, the Mobile user clicks on the Comment icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will add Comments and return the post list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **72.0 Comment post**   1. On the post list, on the specific post, Mobile users click on the Comment icon. 2. A Comment post popup will be shown:    1. Comment: number input    2. Comment button 3. Mobile users click on Comment and the user’s Comment will be added. 4. Comment post popup will close. 5. List-post will be updated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **72.0.E1 Post no longer exists**   1. The system shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.73 UC73 – Report post

****

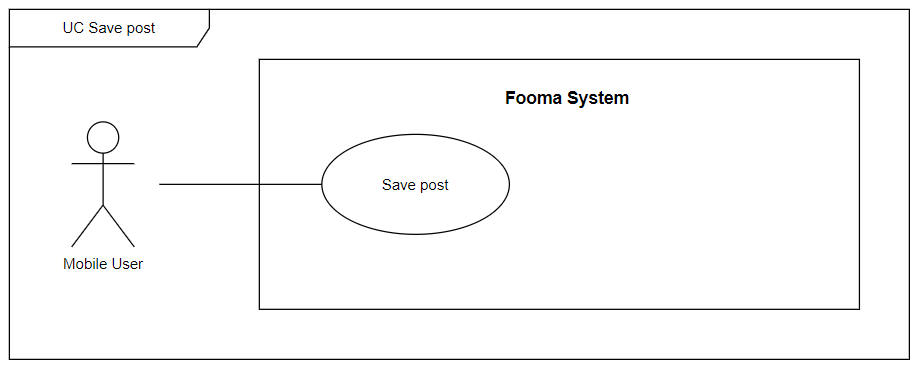
###### Figure 74. <Use case> Report post

###### Table 76. UC73 - Report post

| ID and Name: | **UC-73 Report post** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can Report post. | | |
| Trigger: | On the post list, on the specific post, the Mobile user clicks on the Report icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will add the user’s report and return the post list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **73.0 Report post**   1. On the post list, on the specific post, the Mobile user clicks on the Report icon. 2. A Report post popup will be shown:    1. Title: text input    2. Content: text input    3. Report button 3. After the Mobile user clicks on the Report button, the report will be submitted. 4. The report post popup will be closed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.74 UC74 – Save post

****

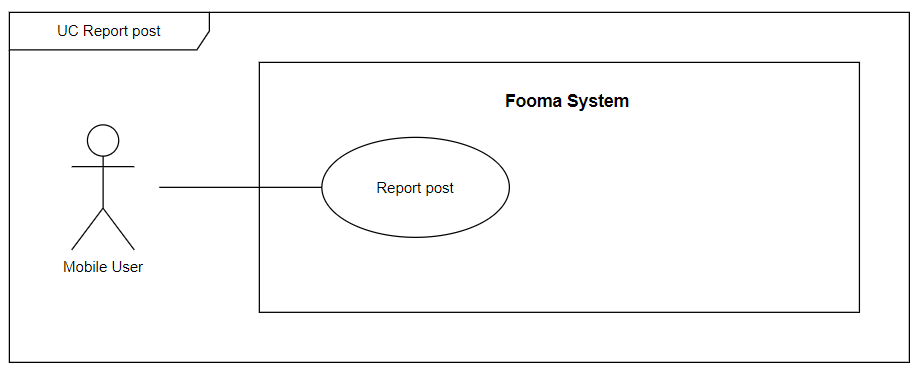
###### Figure 75. <Use case> Save post

###### Table 77. UC74 - Save post

| ID and Name: | **UC-74 Save post** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can Save post. | | |
| Trigger: | On the post list, on the specific post, the Mobile user clicks on the Save icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| Post-conditions: | POST-1:   * Success: Fooma Mobile Application will save posts and return the post list. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **74.0 Save post**   1. On the post list, on the specific post, the Mobile user clicks on the Save icon. 2. the post will be added to the user’s saved post list 3. List-post will be updated view (add saved icon). | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **74.0.E1 post no longer exists**   1. The system shows an error message. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.75 UC75 – Report user

****

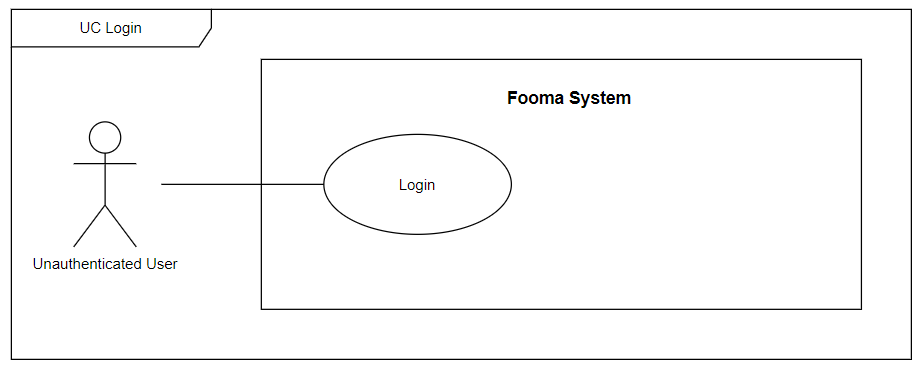
###### Figure 76. <Use case> Report user

###### Table 78. UC75 - Report user

| ID and Name: | **UC-75 Report user** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Mobile user |  |  |
| Description: | Mobile user can Report user. | | |
| Trigger: | On the user profile screen, the Mobile user clicks on the Report icon. | | |
| Preconditions: | PRE-1. User has logged in with the Mobile user role. | | |
| user-conditions: | POST-1:   * Success: Fooma Mobile Application will add the user’s report. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **75.0 Report user**   1. On the user profile screen, the Mobile user clicks on the Report icon. 2. A Report user popup will be shown:    1. Title: text input    2. Content: text input    3. Report button 3. After the Mobile user clicks on the Report button, the report will be submitted. 4. The report user popup will be closed. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.76 UC76 – Login

****

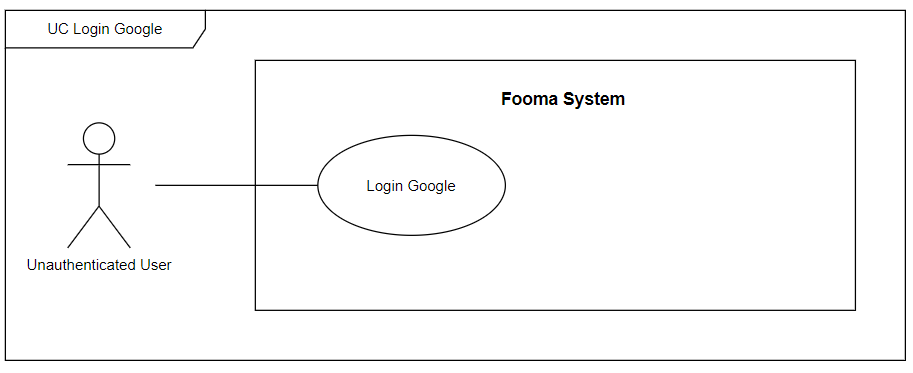
###### Figure 62. <Use case> Login

###### Table 63. UC76 - Login

| ID and Name: | **UC-76 Login** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Unauthenticated user |  |  |
| Description: | Unauthenticated user can Log in. | | |
| Trigger: | On the user Login screen, the Unauthenticated user clicks on the Login button. | | |
| Preconditions: | PRE-1. An unauthenticated user has opened Fooma web application or Fooma mobile application. | | |
| user-conditions: | POST-1:   * Success: The user will be logged into the Fooma web application or Fooma mobile application. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **76.0 Login**   1. On the user Login screen, a Login form will be shown:    1. Email: text input    2. Password: text input    3. Login button 2. After the Unauthenticated user fills the Email, and password field and clicks on the Login button, the system will check the submitted data and do login action and return the Bearer token if a user is valid. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.77 UC77 – Login Google

****

###### Figure 78. <Use case> Login Google

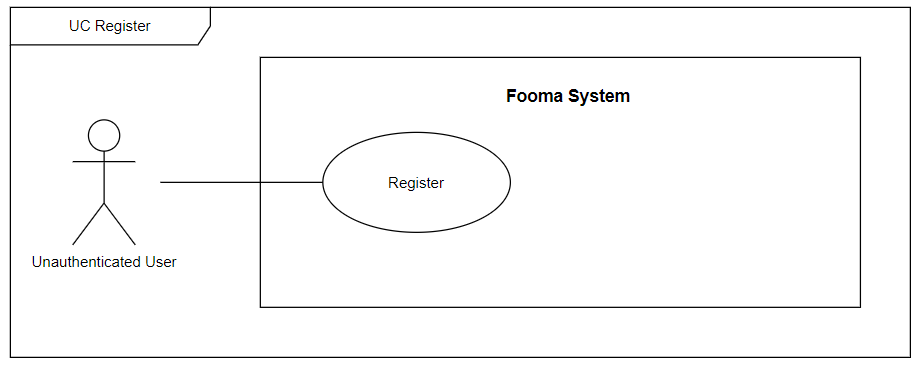
###### Table 80. UC77 - Login Google

| ID and Name: | **UC-77 Login** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Unauthenticated user |  |  |
| Description: | Unauthenticated user can Log in. | | |
| Trigger: | On the user Login screen, the Unauthenticated user clicks on the Login Google button. | | |
| Preconditions: | PRE-1. An unauthenticated user has opened Fooma mobile application. | | |
| user-conditions: | POST-1:   * Success: The user will be logged into the Fooma mobile application. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **77.0 Login**   1. On the user Login screen, unauthenticated users click on the Login Google button. 2. The system will check the submitted data and do login google action and return the Bearer token if a user is valid. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

## 

#### 2.2.78 UC78 – Register

****

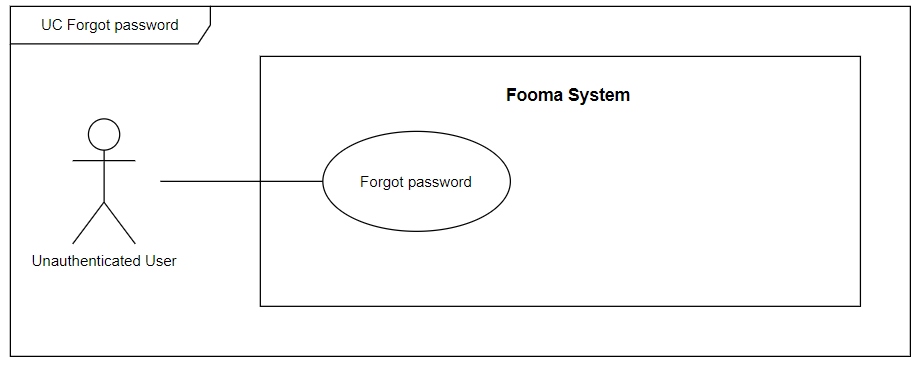
###### Figure 79. <Use case> Register

###### Table 81. UC78 - Register

| ID and Name: | **UC-78 Register** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Unauthenticated user |  |  |
| Description: | Unauthenticated user can register. | | |
| Trigger: | On the user Register screen, the Unauthenticated user clicks on the Register button. | | |
| Preconditions: | PRE-1. An unauthenticated user has opened Fooma mobile application. | | |
| user-conditions: | POST-1:   * Success: The user account will be added to the Fooma Food management application. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **78.0 Register**   1. On the user Register screen, a Register form will be shown:    1. Email: text input    2. Name: text input    3. Password: password input    4. Re-enter password: password input    5. Image: file input    6. Bio: text input    7. Register button 2. After the Unauthenticated user fills all the required fields and clicks on the Register button, the system will check the submitted data and add the user’s account to Fooma Food Management Application system. 3. If register success then an email containing the activation code to the registered email. 4. user took the activation code and then log in, and the activation screen will be shown. 5. user type activation code and the account will be fully activated. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.79 UC79 – Forgot password

****

###### Figure 79. <Use case> Forgot password

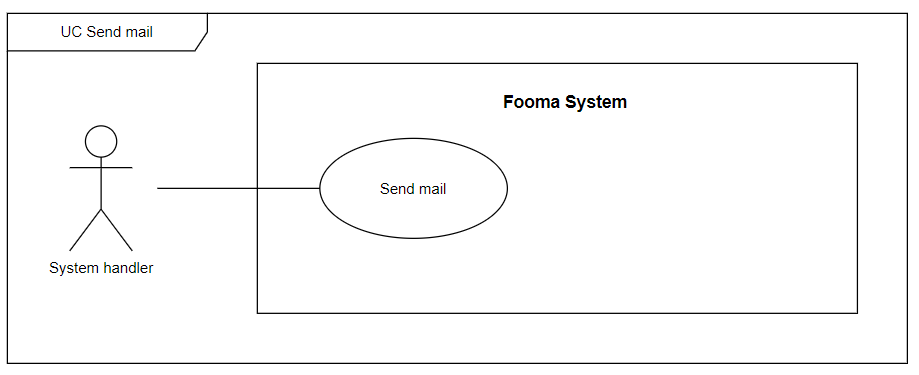
###### Table 81. UC79 - Forgot password

| ID and Name: | **UC-79 Forgot password** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | Unauthenticated user |  |  |
| Description: | Unauthenticated user can Forgot password. | | |
| Trigger: | On the user Login screen, the Unauthenticated user clicks on the Forgot password button. | | |
| Preconditions: | PRE-1. An unauthenticated user has opened Fooma mobile application. | | |
| user-conditions: | POST-1:   * Success: The user account’s password will be updated. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **79.0 Forgot password**   1. On the user Forgot password screen, a Forgot password form will be shown:    1. Email: text input    2. Send code button 2. After the Unauthenticated user fills the email and clicks on Send code button. 3. The system will send the code to the user’s email and the mobile app will be redirected to the Type code screen. 4. The user took the activation code and then typed in the Type code screen. 5. if the code is valid, the mobile application will be redirected to Enter new password screen. 6. The user enters the new password and clicks on the Change password button, new password will be updated to the user’s account. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

## 

#### 2.2.80 UC80 – Send mail

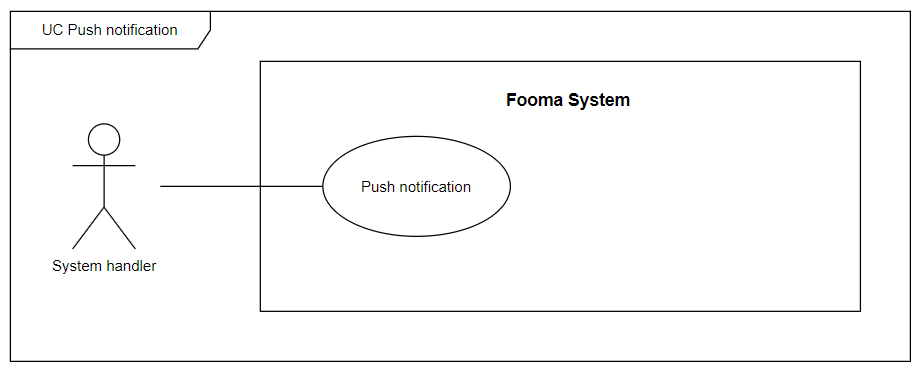
****

###### Figure 81. <Use case> Send mail

###### Table 83. UC80 - Send mail

| ID and Name: | **UC-80 Send mail** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | System handler |  |  |
| Description: | System can Send mail. | | |
| Trigger: | User does register or forgot password action. | | |
| Preconditions: | PRE-1. User does register or forgot password action. | | |
| user-conditions: | POST-1:   * Success: The email will be sent to the user’s email. * Fail: Fooma Mobile Application shows an error message. | | |
| Normal Flow: | **80.0 Send mail**   1. At the register screen, the user clicks on the Register button. 2. An email will be sent to the user’s email. | | |
| Alternative Flows: | **80.01Send mail**   1. At the Forgot password screen, the user clicks on the Send code button. 2. An email will be sent to the user’s email. | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 2.2.81 UC81 – Push notification

****

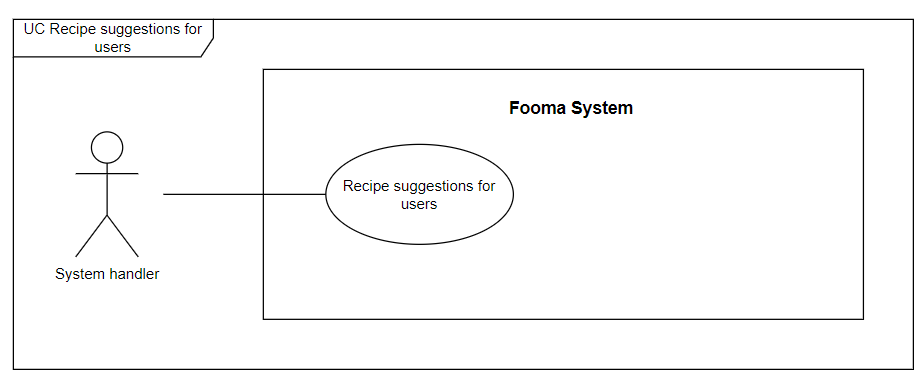
###### Figure 82. <Use case> Push notification

###### Table 84. UC81 - Push notification

| ID and Name: | **UC-81 Push notification** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | System handler |  |  |
| Description: | System can send Push notification. | | |
| Trigger: | Admin sends the password to a group of users or user ingredients are about to expire. | | |
| Preconditions: | PRE-1. Admin sends the password to a group of users.  PRE-2. User ingredients are about to expire. | | |
| user-conditions: | POST-1:   * Success: Push notification will be sent to users. * Fail: Fooma System shows an error message. | | |
| Normal Flow: | **81.0 Push notification**   1. Admin sends the password to a group of users. 2. A Push notification will be sent to the users. | | |
| Alternative Flows: | **81.1 Push notification**   1. User ingredients are about to expire. 2. A Push notification will be sent to the users. | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.82 UC82 – Recipe suggestions for users

****

###### Figure 83. <Use case> Recipe suggestions for users

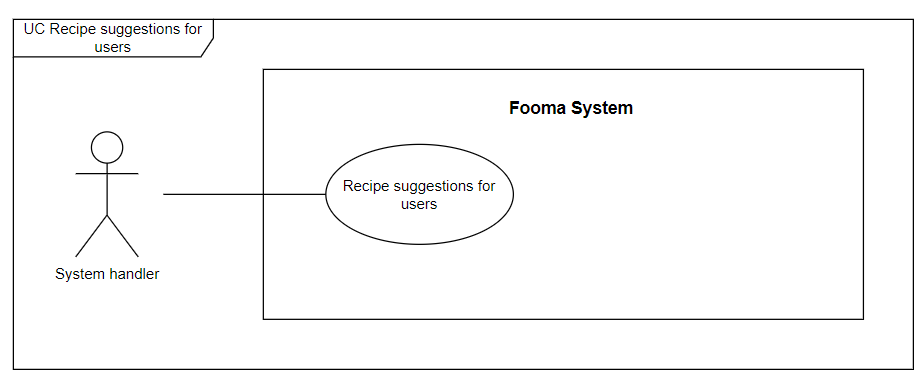
###### Table 85. UC82 - Recipe suggestions for users

| ID and Name: | **UC-82 Recipe suggestions for users** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | System handler |  |  |
| Description: | System suggests a list of recipes to the user. | | |
| Trigger: | The system suggests a list of recipes to the user. | | |
| Preconditions: | PRE-1. The User logged in to Fooma mobile application.  PRE-2. The User’s ingredient list isn’t empty. | | |
| user-conditions: | POST-1:   * Success: The system suggests a list of recipes to the user. * Fail: Fooma System shows an error message. | | |
| Normal Flow: | **82.0 Recipe suggestions for users**   1. The user clicks on the detail of the user’s ingredient in the user’s ingredient list. 2. The system suggests a list of recipes to the user. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 

#### 2.2.83 UC83 – Arrange posts for managers to approve

****

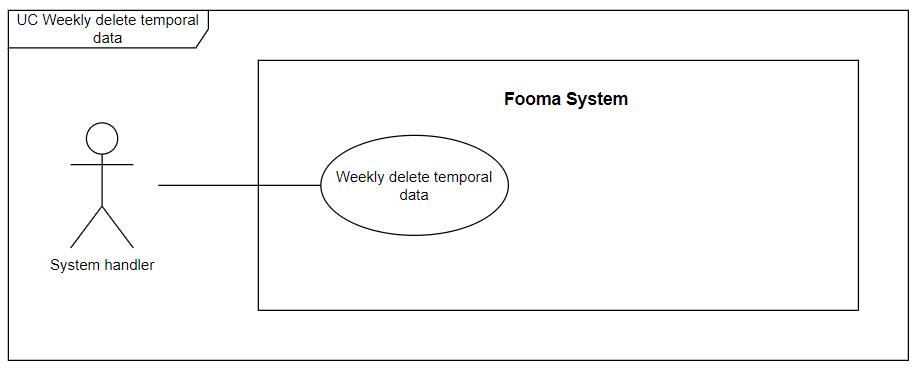
###### Figure 84. <Use case> Arrange posts for managers to approve

###### Table 86. UC83 - Arrange posts for managers to approve

| ID and Name: | **UC-83 Arrange posts for managers to approve** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | System handler |  |  |
| Description: | System arrange posts for managers to approve. | | |
| Trigger: | The system arranges posts for managers to approve. | | |
| Preconditions: | PRE-1. A New post created by the user is submitted to Fooma System. | | |
| user-conditions: | POST-1:   * Success: Post will be arranged for a specific manager. * Fail: Fooma System shows an error message. | | |
| Normal Flow: | **83.0 Arrange posts for managers to approve**   1. A New post created by the user is submitted to Fooma System. 2. Post will be arranged for a specific manager. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

#### 2.2.84 UC84 – Weekly delete temporal data

****

###### Figure 85. <Use case> Weekly delete temporal data

###### Table 87. UC84 - Weekly delete temporal data

| ID and Name: | **UC-84 Weekly delete temporal data** | | |
| --- | --- | --- | --- |
| Created By: | Hientp |  |  |
| Primary Actor: | System handler |  |  |
| Description: | System Weekly delete temporal data. | | |
| Trigger: | The System Weekly delete temporal data. | | |
| Preconditions: | N/A | | |
| user-conditions: | POST-1:   * Success: The System Weekly delete temporal data. * Fail: Fooma System shows an error message. | | |
| Normal Flow: | **84.0 Weekly delete temporal data**   1. The System Weekly delete temporal data. | | |
| Alternative Flows: | N/A | | |
| Exceptions: |  | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-0001, BR-0002, BR-0003, BR-0004 | | |
| Other Information: | N/A | | |
| Assumptions: | N/A | | |

#### 

## 

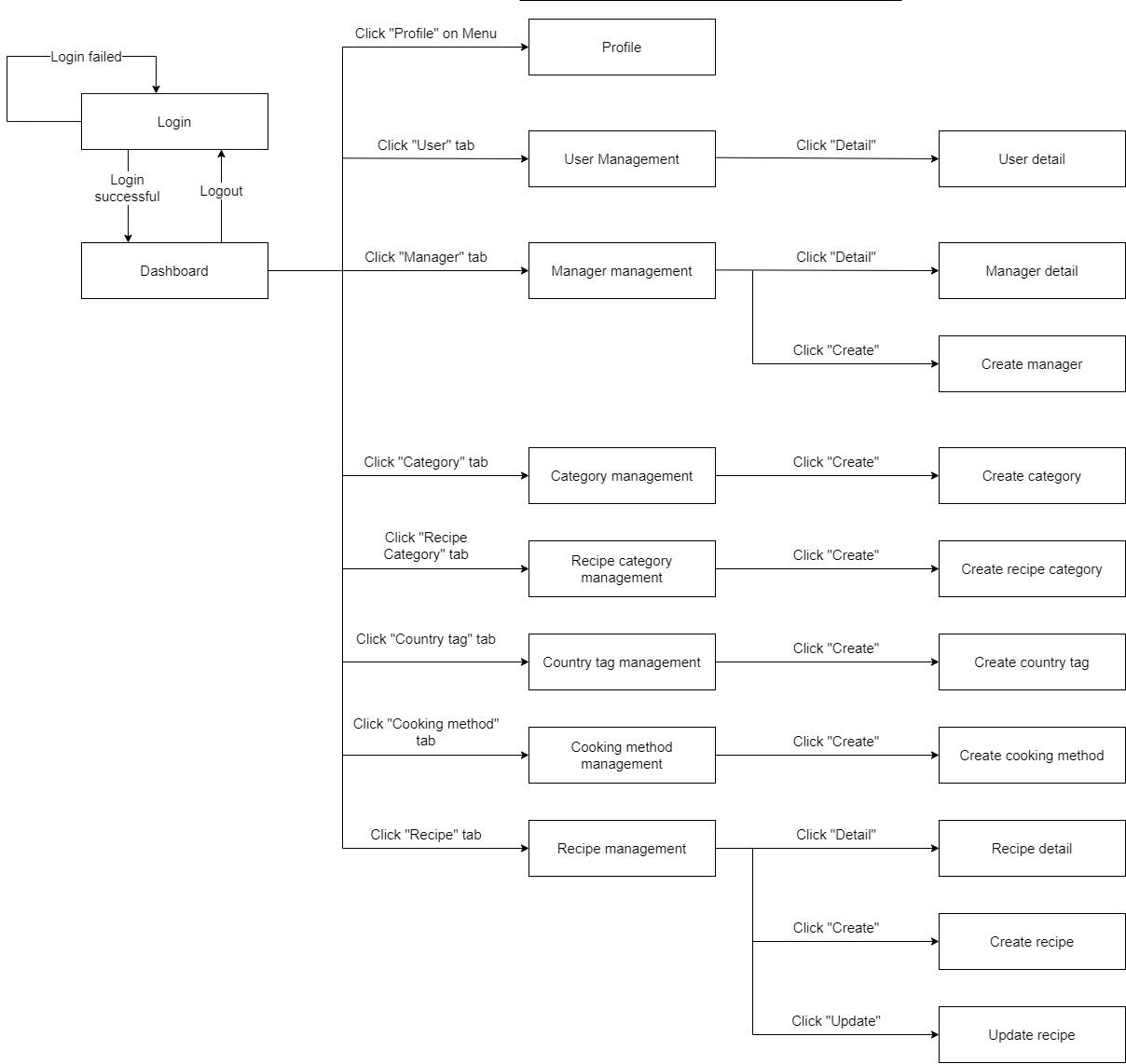
## 

## 3. Functional Requirements

### 3.1 System Functional Overview

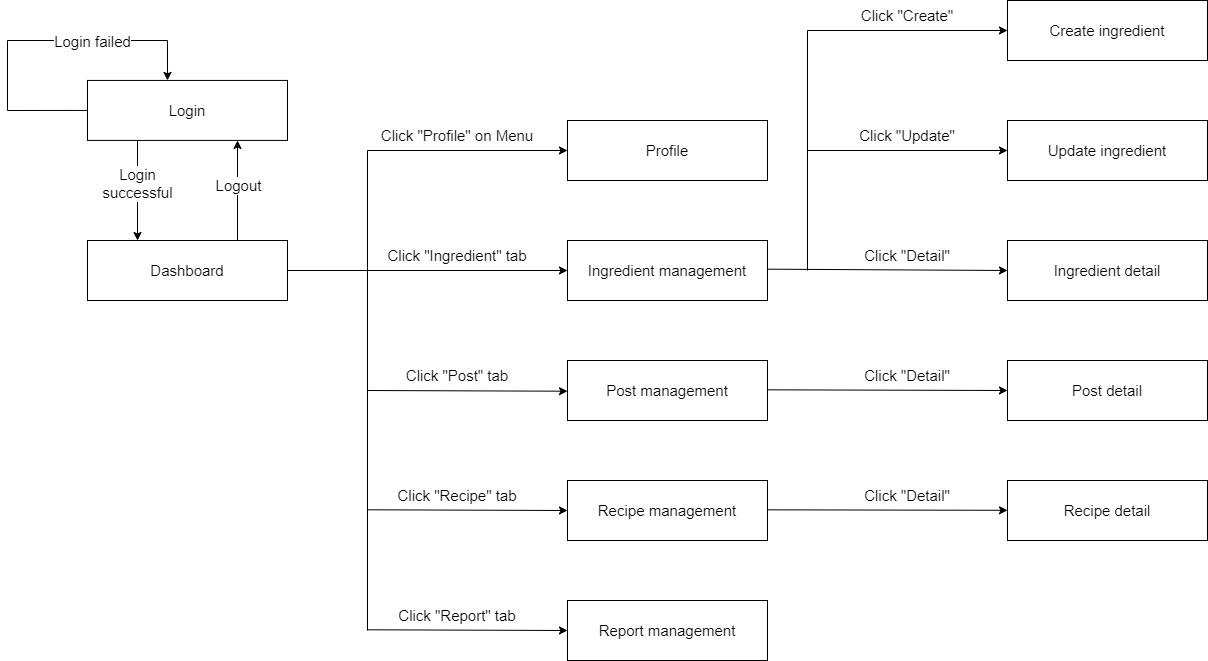
#### 3.1.1. Screen Flow

* **Admin**

****

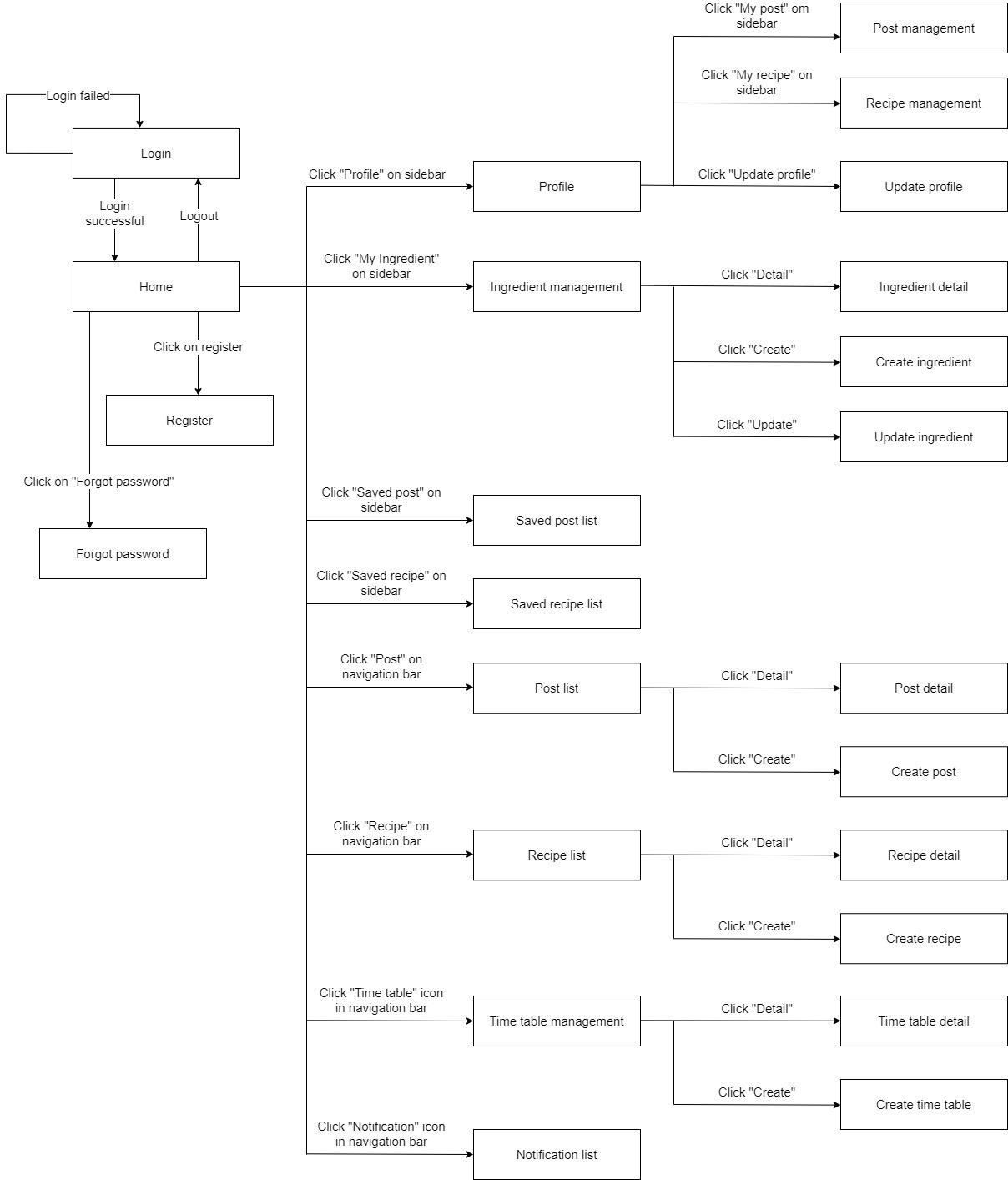
###### Figure 86. Admin screen flow

* **Manager**

****

###### Figure 87. Manager screen flow

* **Mobile User**



###### Figure 88. User screen flow

#### 

#### 

#### 

#### 3.1.2. Screen Authorization

###### Table 88. Fooma Web Application screen authorization

| **Screen / Popup / Modal** | **Unauthenticated User** | **Admin** | **Manager** |
| --- | --- | --- | --- |
| Login | X |  |  |
| Dashboard |  | X | X |
| Profile |  | X | X |
| User management |  | X |  |
| User detail |  | X |  |
| Manager management |  | X |  |
| Manager detail |  | X |  |
| Create Manager |  | X |  |
| Category management |  | X |  |
| Create category |  | X |  |
| Recipe category management |  | X |  |
| Create recipe category |  | X |  |
| Country tag management |  | X |  |
| Create country tag |  | X |  |
| Cooking method management |  | X |  |
| Create cooking method |  | X |  |
| Recipe management |  | X |  |
| Recipe detail |  | X |  |
| Create recipe |  | X |  |
| Update recipe |  | X |  |
| Ingredient management |  |  | X |
| Ingredient detail |  |  | X |
| Create ingredient |  |  | X |
| Update ingredient |  |  | X |
| Post management |  |  | X |
| Post detail |  |  | X |
| Recipe management |  |  | X |
| Recipe detail |  |  | X |
| Report management |  |  | X |

###### Table 89. Fooma Mobile Application screen authorization

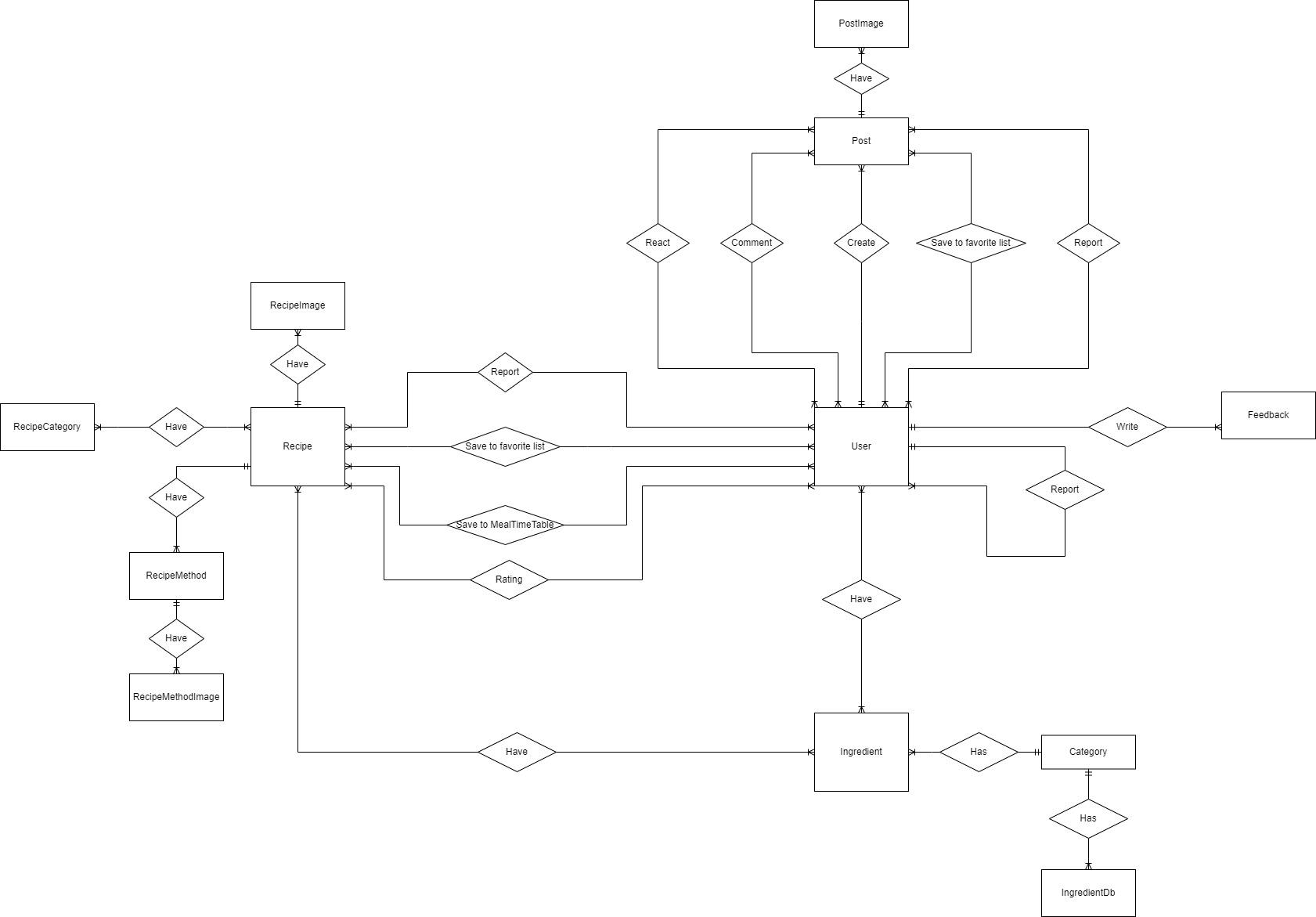
| **Screen / Popup / Modal** | **Unauthenticated User** | **Mobile User** |
| --- | --- | --- |
| Login | X |  |
| Register | X |  |
| Forgot password | X |  |
| Home |  | X |
| Profile |  | X |
| Update profile |  | X |
| Post management |  | X |
| Recipe management |  | X |
| Ingredient management |  | X |
| Ingredient detail |  | X |
| Create ingredient |  | X |
| Update ingredient |  | X |
| Saved post list |  | X |
| Saved recipe list |  | X |
| Post list |  | X |
| Post detail |  | X |
| Create post |  | X |
| Recipe list |  | X |
| Recipe detail |  | X |
| Create recipe |  | X |
| Time table management |  | X |
| Time table detail |  | X |
| Create time table |  | X |
| Notification list |  | X |

#### 3.1.3. Non-Screen Functions

###### Table 88. Non-screen authorization

| **#** | **System Function** | **Description** |
| --- | --- | --- |
| 1 | Send emails | Fooma food-management application sends email containing the activation code needed for active and forgot password action to the user. |
| 2 | Push notification | Foodma food management application sent notifications to users when their ingredient(s) are about to expire or the administration needs to send a notification to a group of users. |

#### 3.1.4. Entity Relationship Diagram



###### Figure 89. Entity Relationship diagram

## 4. Non-Functional Requirements

### 4.1 External Interfaces

#### a. User Interfaces

UI-1: The user interface should be clear, consistent, and easy to use.

UI-2: The label of the input field and error message must be meaningful(both on web applications and mobile applications).

UI-3: Optimized for both web platform and mobile platform.

#### b. Software Interfaces

SI-1: Fooma Food Management Application supports any browser that supports HTML5 and CSS3.

SI-2: For mobile applications, support Android version 6.0 (Marshmallow) or higher.

SI-3: For system handlers, it requires Windows 10 (Professional edition).

#### c. Communications Interfaces

CI-1: Fooma food-management application sent notifications to users when their ingredient(s) are about to expire or the administration needs to send a notification to a group of users.

### 4.2 Quality Attributes

#### a. Usability

* The user interface is designed to be easy to use.

#### b. Reliability

#### c. Performance

* The average response time is less than 3s.

#### d. Dependability

##### **d1. Security**

* JWT is used for authentication and authorization.
* All input and data are validated.

##### **d2. Safety**

#### e. Supportability

* Should apply the RESTful Web API standard.
* Should follow the coding convention.

#### f. Design Constraints

* The mobile application is built using Flutter framework and Dart.
* Web application is built from Angular.
* Back-end server is built using Asp.NET core and Entity Framework based on Repository and UnitOfWork patterns.

#### g. Support Documents

* Installation instructions should be detailed and easy to understand.
* Every configuration setting should be included.

## 5. Other Requirements

### 5.1 Messages List

###### Table 89. Message list

| **#** | **Message code** | **Message type** | **Context** | **Content** |
| --- | --- | --- | --- | --- |
| 1 | MSG0001 | Error message | ? | ? |

## 6. Algorithms