|  |  |  |
| --- | --- | --- |
| Use Case Name : Registration | ID : 1 | Importance Level : High |
| Primary Actor : New User | Use case Type : Detail , Essential | |
| Stakeholders and Interests:  new user: wants to make a registration | | |
| Brief Description:  this Use Case show how to sign up in Application | | |
| Trigger: User Send His phone number to Server by pressing Submit button  Type : External | | |
| Relationships :  Association: New User  Include:  Extend:  Generalization: | | |
| Normal Flow of Events:   1. the user Enter his phone number and submit it 2. the phone number will be send to server and check there 3. since his a new User so will be asked to enter his information 4. the user info will be send for server to be stored | | |
| SubFlows:  S-1: Check phone number by Server:  1- server will check database for finding a match with sent  phone number  2- send “NewUser” Or “OldUser” as response  S-2: Store User Info:  1- Server will generate a unique token  2- Server will Store user information and his token in Database  3- Server send user’s token as response | | |
| Alternate/Exceptional Flows:  S-1 -> (1) : if server found a match, registration page will not be shown and login process will  start | | |

|  |  |  |
| --- | --- | --- |
| Use Case Name : Profile manager | ID : 2 | Importance Level : Medium |
| Primary Actor : Logged in User | Use case Type : Detail , Essential | |
| Stakeholders and Interests:  logged in user: wants to edit his/her profile for more information and add profile picture for better appearance | | |
| Brief Description:  this Use Case show how user can edit his/her profile | | |
| Trigger: user edit desired fields and save changes  Type : external | | |
| Relationships :  Association:  Include:  Extend:  Generalization: | | |
| Normal Flow of Events:   1. the user will click on Profile managing Button 2. last saved fields will be shown 3. User can edit them and save fields | | |
| SubFlows:  S-1: Showing last fields by Server:  1- server will check database for finding a match fields for this profile  2- send last fields to user  S-2: Store User Info:  1- after clicking “save changes” by user  2- Server will Store user information | | |
| Alternate/Exceptional Flows: | | |

|  |  |  |
| --- | --- | --- |
| Use Case Name : Ewallet | ID : 3 | Importance Level : High |
| Primary Actor : Logged in User | Use case Type : Detail , Essential | |
| Stakeholders and Interests:  logged in user: wants to manage Ewallet | | |
| Brief Description:  this Use Case show how user can | | |
| Trigger: User Send His phone number to Server while its not exist in Database  Type : | | |
| Relationships :  Association:  Include:  Extend:  Generalization: | | |
| Normal Flow of Events:   1. the user Enter his phone number and submit it 2. the phone number will be send to server and check there 3. since his a new User so will be asked to enter his information 4. the user info will be send for server to be stored | | |
| SubFlows:  S-1: Check phone number by Server:  1- server will check database for finding a match with sent  phone number  2- send “NewUser” Or “OldUser” as response  S-2: Store User Info:  1- Server will generate a unique token  2- Server will Store user information and his token in Database  3- Server send user’s token as response | | |
| Alternate/Exceptional Flows:  S-1 -> (1) : if server found a match, registration page will not be shown and login process will  start | | |

|  |  |  |
| --- | --- | --- |
| Use Case Name : Help Center | ID : 4 | Importance Level : High |
| Primary Actor : Old User | Use case Type : Detail , Essential | |
| Stakeholders and Interests:  old user: wants to ask a question | | |
| Brief Description:  this use case shows how to solve user’s problem | | |
| Trigger: user click on Help Center Button  Type : external | | |
| Relationships :  Association: old user  Include:  Extend:  Generalization: | | |
| Normal Flow of Events:   1. the user will click on Help Center Button 2. List of his tickets will be shown 3. List of FAQ will be shown | | |
| SubFlows:  S-1: Add new Ticket:  1- user click on “ new ticket”  2- user will write his question in text area  3- user will click on send button  4- question,user’s token will be send to server  5- show suitable message to user  S-2: Answer to old ticket:  1- user open his ticket  2- press “Send Answer” button  3- user will write his question in text area  4- user will click on “send” button  5- answer and user’s token will be send to server  6- show suitable message to user  7- if ticket was closed:  status of ticket will be change to “Open”  S-3: Change Status:  1- user open his ticket  2- press “close ticket” button  3- ticket will be closed | | |
| Alternate/Exceptional Flows:  S-2 -> : if ticket was closed by Help center, user won't be able to send answer | | |

|  |  |  |
| --- | --- | --- |
| Use Case Name : Smart Routing | ID : 5 | Importance Level : Medium |
| Primary Actor : New User | Use case Type : Detail , Essential | |
| Stakeholders and Interests:  new user: wants to make a registration | | |
| Brief Description:  this Use Case show how to sign up in Application | | |
| Trigger: User Send His phone number to Server while its not exist in Database  Type : | | |
| Relationships :  Association:  Include:  Extend:  Generalization: | | |
| Normal Flow of Events:   1. the user Enter his phone number and submit it 2. the phone number will be send to server and check there 3. since his a new User so will be asked to enter his information 4. the user info will be send for server to be stored | | |
| SubFlows:  S-1: Check phone number by Server:  1- server will check database for finding a match with sent  phone number  2- send “NewUser” Or “OldUser” as response  S-2: Store User Info:  1- Server will generate a unique token  2- Server will Store user information and his token in Database  3- Server send user’s token as response | | |
| Alternate/Exceptional Flows:  S-1 -> (1) : if server found a match, registration page will not be shown and login process will  start | | |

|  |  |  |
| --- | --- | --- |
| Use Case Name : Traveling History | ID : 6 | Importance Level : Medium |
| Primary Actor : New User | Use case Type : Detail , Essential | |
| Stakeholders and Interests:  new user: wants to make a registration | | |
| Brief Description:  this Use Case show how to sign up in Application | | |
| Trigger: User Send His phone number to Server while its not exist in Database  Type : | | |
| Relationships :  Association:  Include:  Extend:  Generalization: | | |
| Normal Flow of Events:   1. the user Enter his phone number and submit it 2. the phone number will be send to server and check there 3. since his a new User so will be asked to enter his information 4. the user info will be send for server to be stored | | |
| SubFlows:  S-1: Check phone number by Server:  1- server will check database for finding a match with sent  phone number  2- send “NewUser” Or “OldUser” as response  S-2: Store User Info:  1- Server will generate a unique token  2- Server will Store user information and his token in Database  3- Server send user’s token as response | | |
| Alternate/Exceptional Flows:  S-1 -> (1) : if server found a match, registration page will not be shown and login process will  start | | |

|  |  |  |
| --- | --- | --- |
| Use Case Name : Score Page | ID :7 | Importance Level : Low |
| Primary Actor : Old User | Use case Type : Detail , Essential | |
| Stakeholders and Interests:  old user: wants to manage his score | | |
| Brief Description:  this Use Case lets user to manage his/her score | | |
| Trigger: User click on score button  Type : external | | |
| Relationships :  Association: old user, profile,e wallet  Include:  Extend:  Generalization: | | |
| Normal Flow of Events:   1. the score of user will be shown as soon as user open score page 2. the unique invitation link will be show 3. user will redirect to ewallet page if click on “use score” | | |
| SubFlows:  S-1: Share invitation link:  1-user click on “invite friends” button  2- sharing page will be drop up  3- user can send his invitation link after choosing way of sharing  S-2: Use score:  1- user will click on his score  2- user will redirect to E-Wallet page | | |
| Alternate/Exceptional Flows: Non | | |

|  |  |  |
| --- | --- | --- |
| Use Case Name : Routing | ID : 8 | Importance Level : Medium |
| Primary Actor : New User | Use case Type : Detail , Essential | |
| Stakeholders and Interests:  new user: wants to make a registration | | |
| Brief Description:  this Use Case show how to sign up in Application | | |
| Trigger: User Send His phone number to Server while its not exist in Database  Type : | | |
| Relationships :  Association:  Include:  Extend:  Generalization: | | |
| Normal Flow of Events:   1. the user Enter his phone number and submit it 2. the phone number will be send to server and check there 3. since his a new User so will be asked to enter his information 4. the user info will be send for server to be stored | | |
| SubFlows:  S-1: Check phone number by Server:  1- server will check database for finding a match with sent  phone number  2- send “NewUser” Or “OldUser” as response  S-2: Store User Info:  1- Server will generate a unique token  2- Server will Store user information and his token in Database  3- Server send user’s token as response | | |
| Alternate/Exceptional Flows:  S-1 -> (1) : if server found a match, registration page will not be shown and login process will  start | | |