Use Case List

|  |  |  |
| --- | --- | --- |
| ID | Primary Actor | Use Case Title |
| 0 | Renter | Leshen successfully searches for Parking Spot |
| 1 | Space owner, Renter | Haggling |
| 2 | Space owner, Renter | Payment |
| 3 | Renter | Renter profile creation |
| 4 | Space owner | Space owner profile creation |
| 5 | Space owner and Renter | Reviews and comments for profiles |
| 6 | Space owner  and Renter | Advanced scheduling/reserving |

Use Case Template

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 0 | | |
| Use Case Name: | Leshen successfully searches for Parking Spot | | |
| Created By: | Group 3 | Last Updated By: | Ji Eun Yang |
| Date Created: | 2/16/16 | Date Last Updated: | 2/16/16 |

|  |  |
| --- | --- |
| Actors: | Space owner, Renter |
| Description: | Leshen wants to park his car for a baseball game. He has difficulty finding public parking and looks to iPark. He successfully parks nearby in someone’s private space he rented. After his game he gets his car and leaves. |
| Trigger: | There is an excessive high demand for parking in a localized area. |
| Preconditions: | 1. Space owners have already registered rentable spaces nearby.  2. All actors are authenticated.  3. Leshen is already in his car looking for parking. |
| Postconditions: | 1. Leshen finds and sends a request to an available. |
| Normal Flow: | 0.0 Location services provide Leshen’s location  0.1 App finds available spaces near the location  0.2 App sends Leshen a list of nearby options, ranked by price and distance  0.3 Leshen chooses one and requests it |
| Alternative Flows: | 0.0 ibid  0.1 ibid  0.2 There are no available spaces nearby |
| Exceptions: | 0.0.E.0 Unavailable space is listed as available space.  0.0.E.1 Owner accepts 2 requests at conflicting times.  0.0.E.2 DoubleBookedChecker() checks if the location and time are double booked.  0.0.E.3 One person is removed from the spot  0.0.E.4 An apology message is sent to the removed person telling them they’ve been bumped and also blaming the spot owner  0.0.E.5 They’re given a try again option and prioritized in some way  0.0.E.6 The owner is told off. Perhaps penalized. |
| Includes: | DoubleBookedChecker() |
| Priority: | 1. Web portal 2. User authentication 3. Database |
| Frequency of Use: | 3-5 times over a ten minute span. Probably once or so a week. |
| Business Rules: | None |
| Special Requirements: | Must update states of available parking quickly. |
| Assumptions: | 1. It’s not easy for users to find available public parking spots  by driving around that area.  2. There are people who would like to rent out their private parking spaces.  3. Users want a way to rent parking spots. |
| Notes and Issues: | None |

**Use Case Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1 | | |
| Use Case Name: | Haggling | | |
| Created By: | Group 3 | Last Updated By: | Nisa Gurung |
| Date Created: | 3/23/2016 | Date Last Updated: | 3/23/2016 |

|  |  |
| --- | --- |
| Actors: | Space owner, Renter |
| Description: | After the renter finds a place, he or she can contact the space owner to negotiate price. |
| Trigger: | Space owner always wants to maximize profit but renter wants to save money. |
| Preconditions: | 1. The renter has found an available price.  2. The renter thinks the listed price is too high. |
| Postconditions: | 1. The renter and the space owner reach an agreement |
| Normal Flow: | 1.0 The renter receives information on the available parking spaces through the app  1.1 The renter choose one of them and contacts the space owner through email or phone depending on what contact information the owner has provided |
| Alternative Flows: | 1.0 There are no available spaces nearby and thus, there is no possibility of negotiation or the renter is satisfied with the price |
| Exceptions: | None |
| Includes: | Information from the space owner’s profile |
| Priority: | 1. Web portal  2. Database |
| Frequency of Use: | 1-2 times over a ten minute span. |
| Business Rules: | None |
| Special Requirements: | None |
| Assumptions: | 1.Renters want to spend the extra time in negotiation.  2.Space owners and renters are “nice” and willing to negotiate. |
| Notes and Issues: | None |

**Use Case Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 2 | | |
| Use Case Name: | Payment | | |
| Created By: | Group 3 | Last Updated By: | Leshen Sui |
| Date Created: | 3/23/2016 | Date Last Updated: | 3/23/2016 |

|  |  |
| --- | --- |
| Actors: | Space owner, Renter |
| Description: | The renter makes a payment which the owner of the selected space receives. |
| Trigger: | Renters have to pay for the space they rent, and space owner need a way to receive money. |
| Preconditions: | 1. All actors are authenticated.  2. The renter has found a available parking space  3. The renter agrees the space owner’s conditions(price, time range, etc...) |
| Postconditions: | The space owner receives the payment and allows the renter to park for a period of time. |
| Normal Flow: | 2.0 The website pops up a window “Do you agree to pay \_\_\_$ for parking on \_\_\_\_\_(location) for \_\_\_\_\_\_\_\_\_\_(time range)”  2.1 The renter click “yes” and choose a payment method  2.2 The renter inputs related information  2.3 The renter completes payment and the space owner receives the money |
| Alternative Flows: | 2.0 The website pops up a window “Do you agree to pay \_\_\_$ for parking on \_\_\_\_\_(location) for \_\_\_\_\_\_\_\_\_\_(time range)”  2.1 The renter click “no”  2.2 The website pops up a window “Are you sure you want to decline the transaction?”  2.3 The renter clicks “yes” to cancel the order, or click “no” to go back to the transaction or contact the owner for any negotiation. |
| Exceptions: | 2.0.E.0 The payment method fails  2.0.E.1 The system detects the payment cannot be done  2.0.E.2 A message will be sent to the renter and asks  to use another payment method |
| Includes: | Haggling |
| Priority: | 1.Web portal  2.User authentication |
| Frequency of Use: | 3-5 times over a ten minute span. Probably once or so a week. |
| Business Rules: | None |
| Special Requirements: | Protect the renters’ payment information. Make sure they will not be disclosed. |
| Assumptions: | 1. The space owner linked a bank account to accept payment  2. The renter is willing to pay through the app |
| Notes and Issues: | None |

**Use Case Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 3 | | |
| Use Case Name: | Renter profile creation | | |
| Created By: | Group 3 | Last Updated By: | Nisa Gurung |
| Date Created: | 3/23/2016 | Date Last Updated: | 3/23/2016 |

|  |  |
| --- | --- |
| Actors: | Renter |
| Description: | A renter creates his profile on iPark |
| Trigger: | A renter wants to register an account so he/she can use iPark conveniently. |
| Preconditions: | 1. The renter wants to become a subscriber of iPark.  2. The renter is authenticated. |
| Postconditions: | The renter successfully create his/her account. |
| Normal Flow: | 3.0 The renter click the “sign up” button  3.1 The renter fills in his information following the instructions  provided by the app.  3.2 The renter complete his profile creation, the profile is stored in database. |
| Alternative Flows: | 3.0 The renter click the “sign up” button  3.1 The renter provides wrong information or misses some required information.  3.2 The app ask the renter to correct/refill those blank. |
| Exceptions: | 3.0.E.0 The renter did not provide a required information  3.0.E.1 The application will remind the user to fill out the required parts  3.0.E.2 The renter fills out the registration form correctly |
| Includes: | None |
| Priority: | 1. Database |
| Frequency of Use: | Just one time when the renters first sign up |
| Business Rules: | None |
| Special Requirements | None |
| Assumptions: | It’s not easy for users to find public parking spaces, so they want to use websites like iPark to rent spaces |
| Notes and Issues: | None |

**Use Case Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 4 | | |
| Use Case Name: | Renter profile creation | | |
| Created By: | Group 3 | Last Updated By: | Leshen Sui |
| Date Created: | 3/23/2016 | Date Last Updated: | 3/23/2016 |

|  |  |
| --- | --- |
| Actors: | Space owner |
| Description: | A space owner create his/her profile on iPark |
| Trigger: | A space owner wants to rent his parking space out when it is not currently in use to make some money. |
| Preconditions: | 1. The space owner wants to become a subscriber of iPark.  2. The space owner is authenticated. |
| Postconditions: | The space owner successfully creates his/her account. |
| Normal Flow: | 4.0 The space owner click the “sign up” button  4.1 The space owner fills in his information following the instructions provided by the app.  4.2 The space owner provides the information about his parking space(address, time available, price...)  4.3 The space owner complete his profile creation, the profile is stored in database. |
| Alternative Flows: | 4.0 The space owner click the “sign up” button  4.1 The space owner provides wrong information /misses some blank.  4.2 The app ask the space owner to correct/refill those blank. |
| Exceptions: | 4.0.E.0 The space owner proved fake information about himself/his parking space(e.g it’s a public space).  4.0.E.1 The space owner failed to pass the authentication of the app  4.0.E.2 The app asks the space owner to provide the correct information  4.0.E.3 If the space owner keeps providing fake/wrong information, he/she will be rejected to use the app. |
| Includes: | None |
| Priority: | 1. User authentication  2. Database |
| Frequency of Use: | Just once when the owner first signs up |
| Business Rules: | None |
| Special Requirements: | The private parking space is valid. |
| Assumptions: | The space owner wants to make money by renting his space out when he doesn’t use it. |
| Notes and Issues: | None |

**Use Case Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 5 | | |
| Use Case Name: | Reviews and comments | | |
| Created By: | Group 3 | Last Updated By: | Leshen Sui |
| Date Created: | 3/23/2016 | Date Last Updated: | 3/23/2016 |

|  |  |
| --- | --- |
| Actors: | Renters |
| Description: | The renter writes reviews and comments on the app to share his experience with others. |
| Trigger: | After the renter has used a parking space, he or she might want to comment on their experiences |
| Preconditions: | 1. The renter has had experience using a space of the owner he or she is writing review of |
| Postconditions: | 1. Other people can see the reviews and comments online. |
| Normal Flow: | 5.0 The user clicks the “review and comment” button to send a request for comment.  5.1 The server receives the request and pops up a window for the user to write.  5.2 The user completes his comment and clicks “submit”  5.3 The comment will be stored and shared. |
| Alternative Flows: | None |
| Exceptions: | None |
| Includes: | None |
| Priority: | 1. Web Portal  2. Server  3. Database |
| Frequency of Use: | Probably once or twice a week depending on how frequently they rent |
| Business Rules: | None |
| Special Requirements: | Comments must be polite, otherwise it will be shielded. |
| Assumptions: | The renters want to share their experience with other people. |
| Notes and Issues: | None |

**Use Case Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 6 | | |
| Use Case Name: | Advanced scheduling/reserving | | |
| Created By: | Group 3 | Last Updated By: | Leshen Sui |
| Date Created: | 3/23/2016 | Date Last Updated: | 3/23/2016 |

|  |  |
| --- | --- |
| Actors: | Space owner, Renter |
| Description: | The renter contacts the space owner in advance to book an available parking space, knowing that it’s hard to find a public parking space in his/her designated area. |
| Trigger: | There is an excessive high demand for parking in a localized area. |
| Preconditions: | 1. Space owners have already registered rentable spaces nearby.  2. All actors are authenticated.  3. The renter decides to book a parking space in advance. |
| Postconditions: | 1. The renter successfully books an available spot and drive there . |
| Normal Flow: | 6.0 The renter inputs a location online.  6.1 App finds available spaces near the location  6.2 App sends the renter a list of nearby options, ranked by price and distance  6.3 the renter chooses one and requests it, and communicate with the owner. |
| Alternative Flows: | 6.0 The renter inputs a location online.  6.1 App finds available spaces near the location  6.2 There are no available spaces nearby |
| Exceptions: | 6.0.E.0 Unavailable space is listed as available space.  6.0.E.1 Owner accepts 2 requests at conflicting times.  6.0.E.2 DoubleBookedChecker() checks if the location and time are double booked.  6.0.E.3 One person is removed from the spot  6.0.E.4 An apology message is sent to the removed person telling them they’ve been bumped and also blaming the spot owner  6.0.E.5 They’re given a try again option and prioritized in some way  6.0.E.6 The owner is told off. Perhaps penalized. |
| Includes: | DoubleBookedChecker() |
| Priority: | 1. Web Portal  2. User authentication  3. Database |
| Frequency of Use: | 3-5 times over a ten minute span. Probably once or so a week. |
| Business Rules: | None |
| Special Requirements: | Must update states of available parking quickly. |
| Assumptions: | 1. It’s not easy for users to find available public parking spots  by driving around that area.  2. There are people who would like to rent out their private parking spaces.  3. Users want a way to rent parking spots in advance to save time. |
| Notes and Issues: | None |