# INF2C SOFTWARE ENGINEERING 2019-20 COURSEWORK 1

Capturing requirements for a federalised bike rental system

- HUACHENG SONG s1826390
- XUDONG ZHANG s1817972

## Description:

- producing a requirements document for a simplified computer system for managing a bike rental system.

## Q3.1 Stakeholders:

(1) Core stakeholders:

- Locals: The local citizens who want to use bikes to travel around different sights in Scotland and they can easily to use the rental system to find the suitable kind of bike.
- Tourists: Visitors can use the system to have a better knowing of the city and they can easily meet their best needs.
- Bike shops: The providers of rental bikes. The rental system will utilize them to offer customers wide variety of types of bikes. Shops might gain benefits (daily rental money) from system users.
- Bike producers: Bike producers would make more variety of types of bikes (e.g. road bikes, mountain bikes, hybrid bikes, ebikes, etc.) in order to meet the needs of the bike shops.
- Delivery drivers: Delivery drivers would delivery bikes back to the original providers after users finished their renting.

(2) Additional stakeholders:

- Scottish tourism board: The proposer of the rental system. This system is willing to boost tourism and protect the environment due to allowing customers to cycle instead of drive.
- Investors: Investors could gain profits from the rental system which they invested.
- Government: Government would support the system to encourage and help the development of the city. Also, the system may elevate the cycling awareness of public which is the aim that the government would like to see.
- Local police: Police would increase the manpower to maintain the traffic safety in order to prevent the bad traffic consequence caused by the increasing bikes.
- Public transportation: Public transportation might lose some amount of proportion of customers due to the launch of system.
- Local media: Media might report the news of the system.

## Q3.2 System state:

- User state:

1. Personal information (first name, surname, address, post code, phone number).
2. State of deposit (available or hold by provider).
3. Statuses of renting Bike (not renting or keep renting).
4. Information of provider (if user is keep renting, system should record the information of bike provider).
5. order number (unique for each order).

6. Total renting price.
7. Booking date.
7. Required dates (the date that the bike should be returned).

- Provider state:

1. Personal information (first name, surname, address, post code, phone number).
2. Bikes information (e.g. road bike, mountain bike, hybrid bike, ebike, etc).
3. deposit rate (different providers may have different deposit rates).
4. State of deposit (registers the received deposit and updates the status of the bikes
).
5. Statuses of each renting Bikes (available or renting by users).
6. Register with other providers (whether they have other partners).

- Bikes state:

1. Information (types etc.).
2. statuses (available, booking, tracking).
3. renting price. (Different kinds of Bike might have different renting prices).
4. If the bikes is booking, system should record the required date (Date of bikes should be returned).

- Delivery drivers state:

1. Personal information.
1. statuses (available, tracking).
2. State of deposit (registers the received deposit and updates the status of the bikes
).

## Q3.3 Use cases:

### 1. Get quote:

- Primary actor: User
- Summary: Users get and read quotes from the providers after providers provide different quotes on the system.
- Precondition: Users need to provide information on their rental needs.
- Trigger: Users successfully get quotes from providers.
- Guarantees:
  i) Success guarantees: All users would get the different kinds of quotes from providers.
  ii) Failure guarantees: Users did not get preferred quotes.
- Main Success Scenario:
  1. Providers need to register.
  2. Once registered, the provider is able to register new bike types onto the system by providing each type’ s name (e.g. road bike, mountain bike, hybrid bike, ebike, etc), and the full replacement value for bikes of this type (for determining deposit amounts).
  3. Users successfully get quotes.
- Extension:

1. If no quotes are available, system will suggest other quotes.

## 2. Book quote:

- Primary actor: User
- Secondary actor: Provider
- Summary: Users choose and book a preferred bike after looking through the quotes provided by the providers and pay after choosing the quote.
- Precondition: Users need to get quote from providers.
- Trigger: Users successfully book their preferred bike and successfully pay the rent of their preferred bike.
- Guarantees:
  i) Success guarantees: All users would book the preferred bike and successfully pay the deposit and the rental fee without any problems.
  ii) Failure guarantees: After the system find other quotes, there are still none quotes exist. Hence the users do not book any bikes or fail to pay.
  iii) Minimal guarantees: Users find a bike and successfully pay the deposit and the rental fee.
- Main Success Scenario:
  1. Users upload their rental requirements on system.
  2. System show enough quotes from different providers for user.
  3. Users choose one of suited quote.
  4. Users upload their personal information.
  5. Users decide the way of bike collection.
  6. System jumps to payment program when user book the bike successfully.
  7. User pay the rent and deposit online.
  8. System generate a confirmation about rental bike, including the order number, order summary, deposit, total price, delivery and return information.
  9. System stop providing any quotes about booked bike on the required dates.
- Extension:
  1. If no quotes are available, system will suggest other quotes.
  2. Users can pick up from bike shop or deliver.
  3. We assume that all users will pay deposit on time.

## 3. Update bike information:

- Primary actor: Providers
- Summary: In order to register the system, essential information is given by the provider. After registered the system, providers can change information of bikes such as bike states.
- Trigger: Providers successfully register the system and successfully change the information as they wish to.
- Main Success Scenario:
  providers upload their information on the database of system.
- Extension:
  1. Providers can also make partnership agreements with other providers.

2. Provider must set daily rental price for each type of bike included in its stock or it will be a failure.

## 4. Return bikes:

- Primary actor: User
- Secondary actor: Provider (Partner of provider)
- Summary: Users return bikes to original providers or their partner.
- Main Success Scenario:
  1. Users successfully return the bike back to original provider or their partner.
  2. Provider’ s partners will use a delivery driver to return the bikes to the original provider overnight so they are available for rental on the next day.

## 5. Return deposit:

- Primary actor: Provider
- Summary: Provider should return the deposit to the users after they returned the bike.
- Trigger: Provider successfully return the deposit to the users.
- Guarantees:
  i) Success guarantees: Provider successfully return the deposit to the users.
  ii) Failure guarantees: Provider do not return the deposit to the users.
- Extension:
  No extension, we assume that all providers will return deposit to users on time.

## 6. Register:

- Primary actor: Provider
- Secondary actor: Delivery drivers
- Summary: When a customer picks a bike order, the provider or delivery driver registers the received deposit and updates the status of the bikes.
- Main Success Scenario:
  1. When a customer picks a bike order, the provider or delivery driver registers the received deposit and updates the status of the bikes.

## 7. Record bike return to original provider

- Primary actor: Providers (partner)
- Secondary actor: Delivery driver
- Summary: A provider’ s partners will use a delivery driver to return the bikes to the original provider.
- Notes:
- 1. Partner will return the bikes to the original provider overnight so they are available for rental on the next day, and that the status of the bikes will be tracked throughout this process.

## 8. Return bikes:

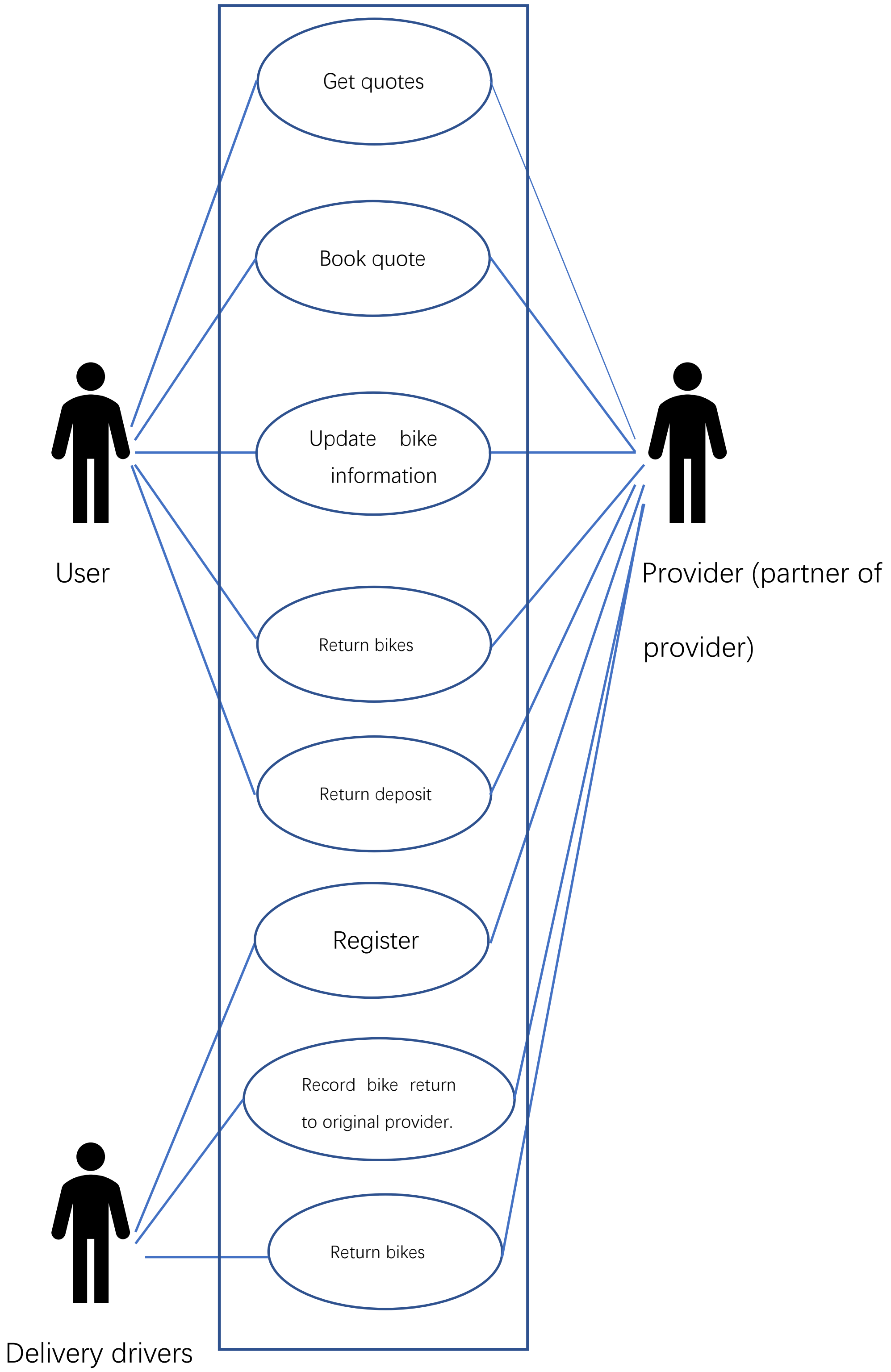
- Primary actor: User
- Secondary actor: Delivery driver

- Summary: Users return bikes to original providers or their partner.
- Main Success Scenario:

  1. Users successfully return the bike back to original provider or their partner.

  2. provider’ s partners will use a delivery driver to return the bikes to the original provider overnight so they are available for rental on the next day.

## Q3.4 Use case diagram:



## Q3.5 Non-functional requirements:

- Security:
  1. Verification: This system should check the identity of the user and the provider.
  2. Protection: The system should able to resist the attacks from any computer virus, in order to protect the personal information and payment process.
  3. Authorization: All transaction should be authorized and confirmed.
- Performance:
  1. Response time: Any operation of user should be responded within 1 second.
  2. System dependability: The system should not stop working unless user loses the signal of the internet. If that happens, the user should be informed.
  3. Showing information directly: User could look through quotes directly without any interference information.
- Availability:

  1. Updateability: This system could update when other requirements occur.

  2. Resource required: This system should be efficient and use as less memory as possible, so that this system could be download on any devices.

  3. Navigating: this system could provide the map of the bike location ,in order to helping user quickly find rent bike.
- Usability:
  1. Simple and convenient operation: The user who use this system for the first time should get familiar with each step clearly and quickly.
  2. Different version: the system should provide different language versions for user who come from different countries.
- Platform Compatibility:
  1. Screen resolution: This system should fit different screen size of wide range of devices from computer to smartphone.
  2. Operating system compatibility: This system could download on any operating system, like Microsoft or MacOS.
- Data Retention:
  1. Information reservation: all of information about users, providers and their bikes will upload and reserve in the system database. When user want to change the any detail, the database would be renewed immediately.
- Reliability:
  1. Payment service: The system should accept most mainstream card types.
  2. Easy recovery: When the internet disconnects, the detail that user did not save would upload automatically. After reconnecting, user could use or change previous information.
- Safety:
  1. This system will not affect the use of other software.
  2. It does not cause any bugs to the devices.

## Q3.6 Ambiguities and subtleties:

- Specific payment method:
  The system should specify some payment method to let users to pay rent.
- Return deposit method:
  The system should specify a return method to let providers to return deposit.
- Searching quotes:
  The system should provide some algorithms in order to run keyword searching.

## Q3.7 Self-assessment:

- Q 3.1 Identify stakeholders 15%/15%

  • Identify core stakeholders of the system 5%/5%
  • Identify additional stakeholders 5%/5%
  • Describe how the system affects each stakeholder 5%/5%

- Q 3.2 Describe system state 10%/10%

  • Include state essential to the operation of the system 5%/5%
  • Include additional state mentioned in the description 5%/5%

- Q 3.3 Describe use cases 40%/40%

  • Identify use cases 10%/10%
  • Describe use cases using the appropriate templates 30%/30%

- Q 3.4 Use case diagram 15%/15%

  • Correctly use UML use case notation 5%/5%
  • Include key actors and use cases 5%/5%
  • Identify connections between actors and use cases 5%/5%

- Q 3.5 Describe non-functional requirements 10%/10%

  • Identify non-functional requirements within the context of the system 7%/7%
  • Provide means for assessing non-functional requirements 3%/3%

- Q 3.6 Ambiguities and subtleties 5%/5%

  • Identify some ambiguities in system description 3%/2%
  • Discuss potential options for resolution of ambiguities 2%/2%