Shake and Chat

Software Design Description

|  |  |
| --- | --- |
| Project Name | Shake and Chat |
| Starting Date | 2017/12/4 |
| Developer | 李旭恩、陳立誠、綦家志、高睦修、王瀚磊、王本奕 |

**版次變更記錄**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版次 | 變更項目 | 變更日期 | 負責人 | 審核人 |
| 0.1 | System Model/System Architecture, Process Design, Class Diagram, Data Design | 2017/12/4 | All | All |
| 0.2 | Data Structure Modify, Introduction Modify | 2017/12/5 | All | All |
| 0.3 | Modify NearbyList UI | 2017/12/5 | All | All |
| 0.4 | Modify Class Diagram | 2017/12/12 | All | All |
| 0.5 | Add feature in project, modify all. | 201712/19 | 綦家志 | All |
| 0.6 | Modify architecture and front-end class diagram | 2018/01/08 | 王本奕 | All |
| 0.7 | Add backend class diagram and sequence diagram | 2018/01/08 | 陳立誠 | All |

目錄

1. 系統模型與架構(System Model/System Architecture) 4

2. 流程設計(Process Design) 5

3. 使用者介面設計(User Interface Design) 6

4. 類別圖設計(Class Diagram) 14

5. 循序圖設計(Sequence Diagram) 17

6. 資料細部設計(Data Design) 17

ShakeActivity 17

MessageHandler 17

GeoHandler 18

GeoInfo 18

EventHandler 18

EventQueue 18

Event 19

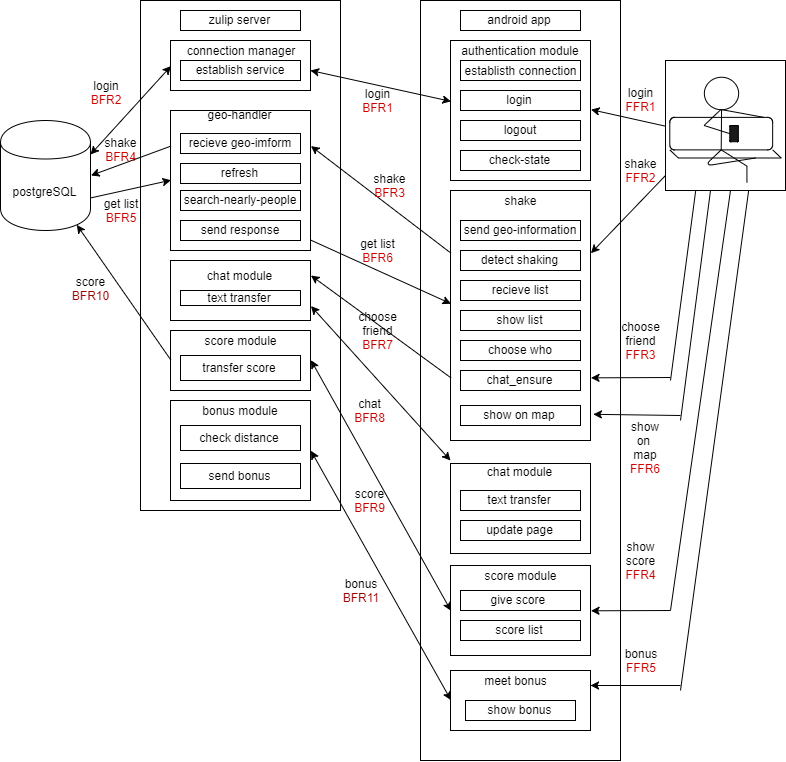
ConnectionManager 19

AuthenticationManager 19

1. 系統模型與架構(System Model/System Architecture)

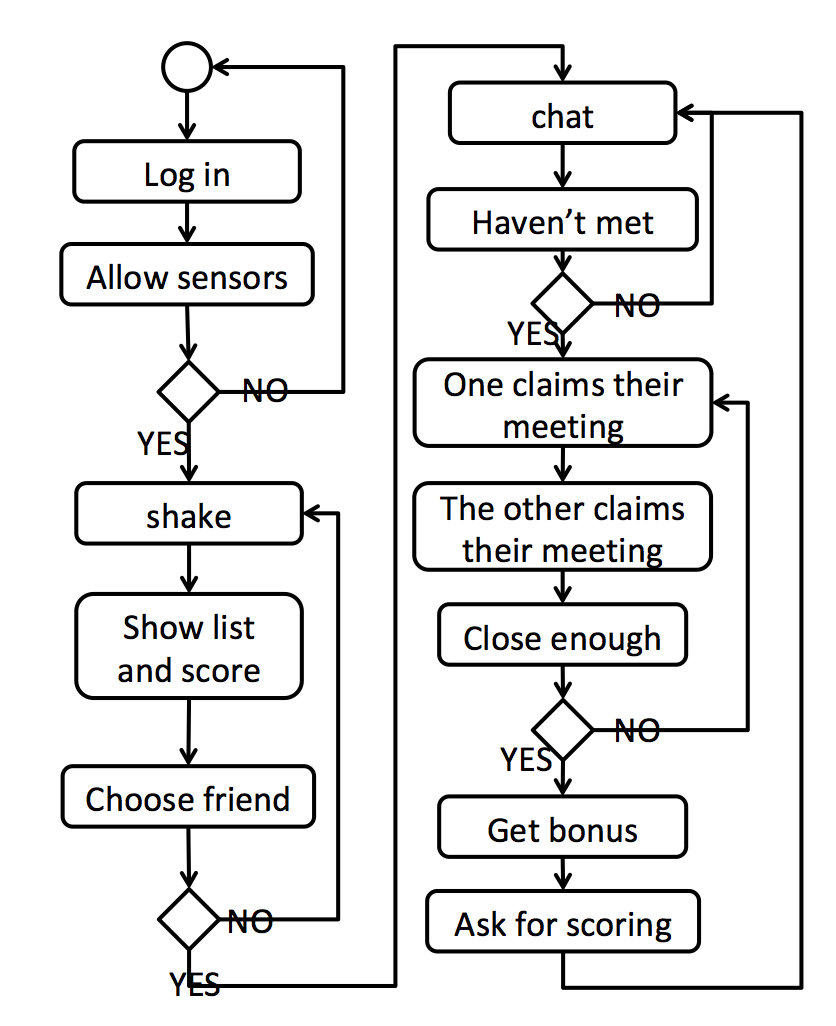
隨著資訊科技的進步，現代人的交友範圍從現實逐漸虛擬化，利用網路作為交友平台的軟體如雨後春筍般出現。我們團隊決定反其道而行，雖然利用網路，但卻希冀能利用網路，使人們能踏出螢幕，重新建立人與人之間的實際互動。

這款在zulip上透過搖一搖手機配對附近之陌生人並使之交流的功能將支援定位系統，讓使用者可以為自己發出「訊號」找尋在一定範圍內一樣在線上活動的使用者，搜尋的範圍可以彈性設定，訊號廣播的時間也很自由，因應安全性上的疑慮，亦可過濾是否要搜尋完全不認識的陌生人，或是只需要搜尋在線的好友名單。專案強調以地點、持續時間為主軸，來進行朋友的搜尋建議。並期望可利用此系統，使線上交友不只是在虛擬平台，更達成人與人之間的真心交流。此外，專案內容主要包含APP的設計、訊息傳遞設計、資料庫演算法設計、後端管理介面等項目。



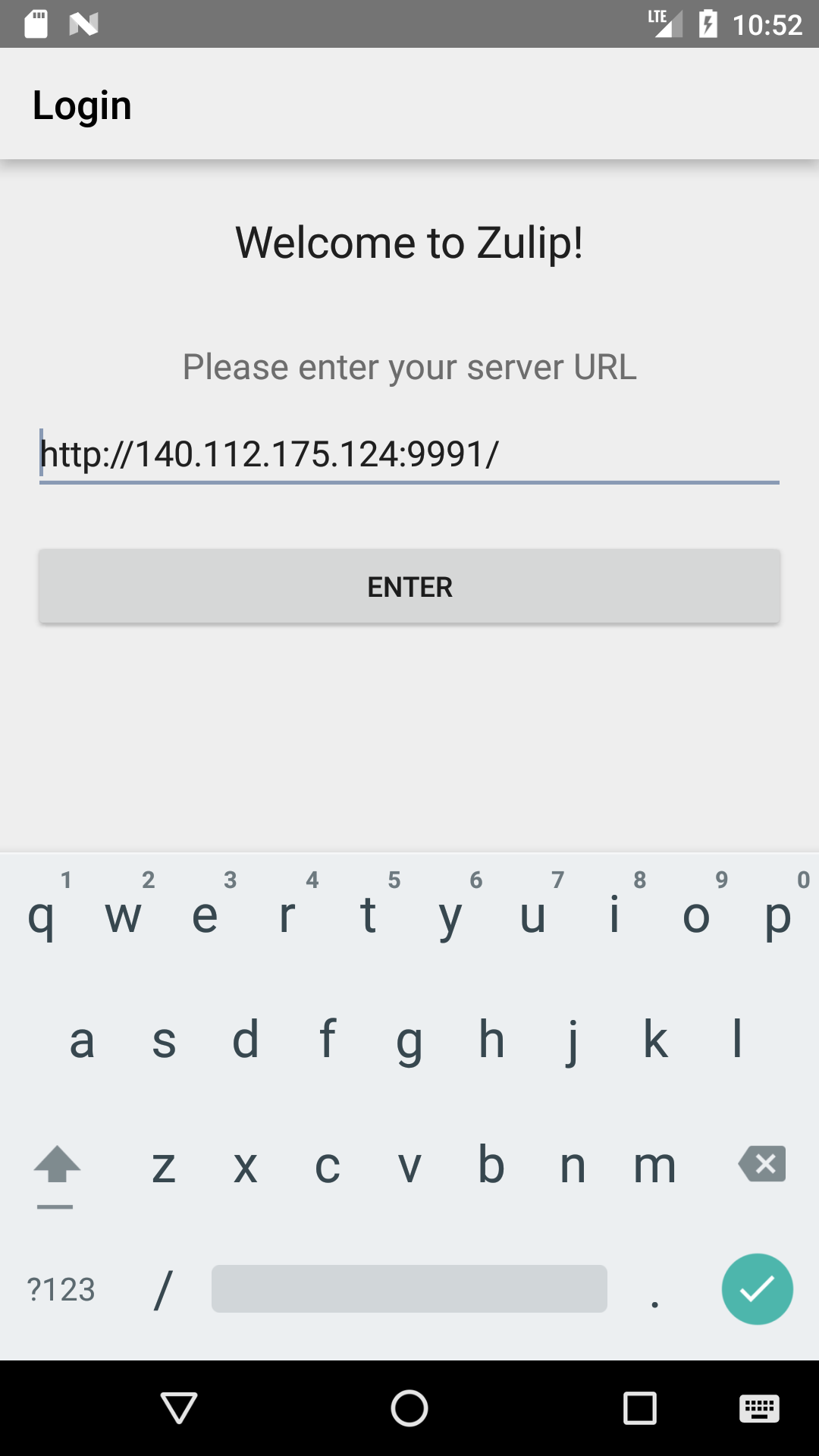
**Fig 1. 系統架構圖 (System Architecture)**

1. 流程設計(Process Design)
2. **Shake and Chat Algorithm**

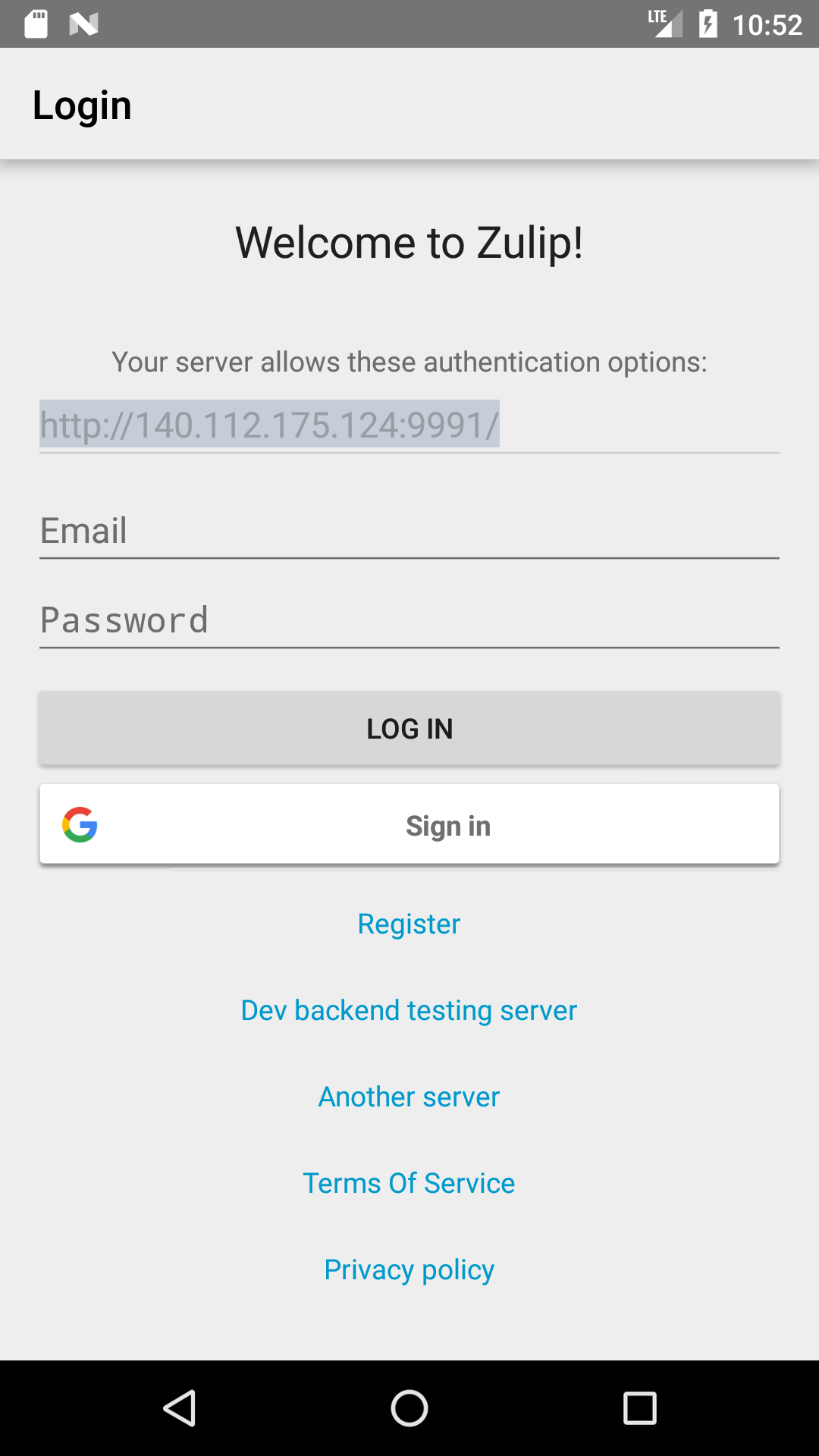


1. 使用者介面設計(User Interface Design)

(1) serverUI



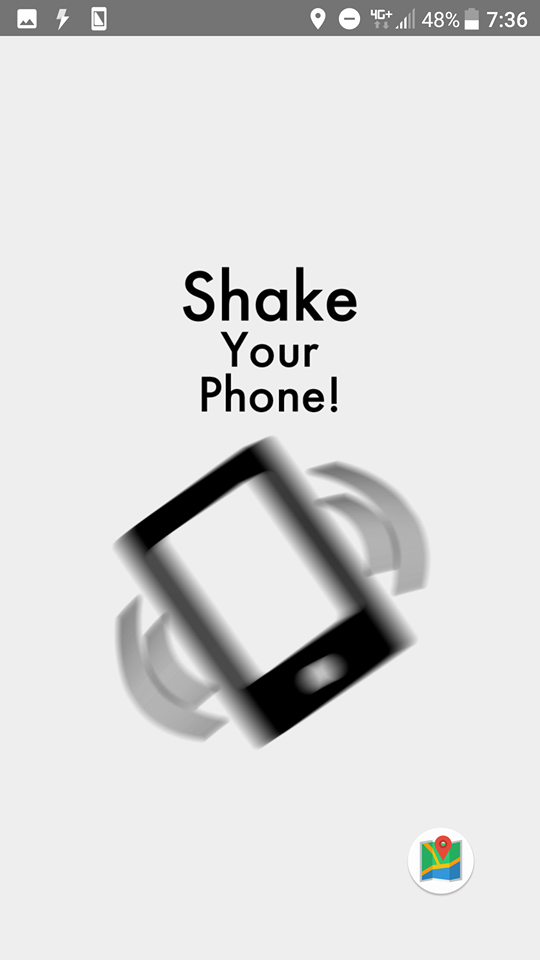
**(2) LoginUI**

****

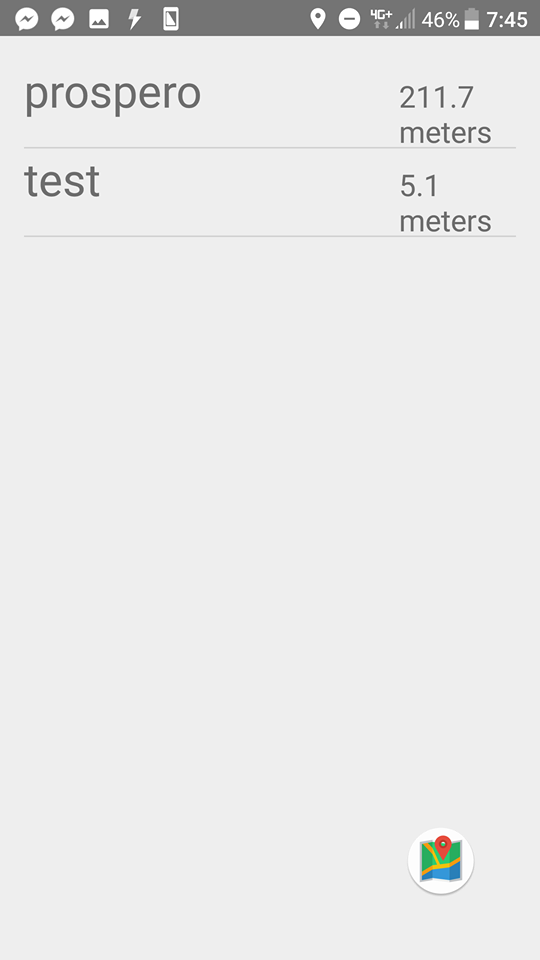
**(3)chatUI**



**(4)shakeUI**



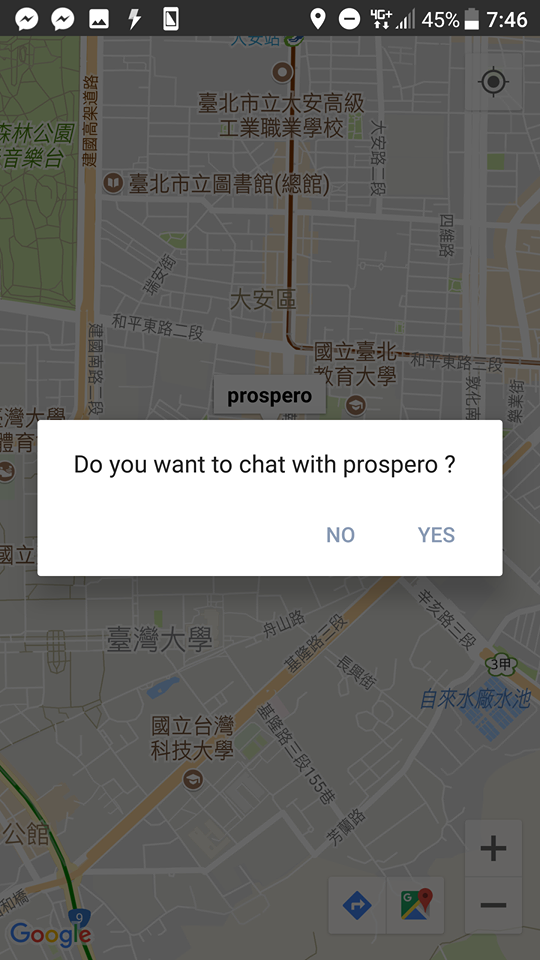
**(5)nearbyUI**





**(6)mapUI**





**(7)bonus&commentUI**



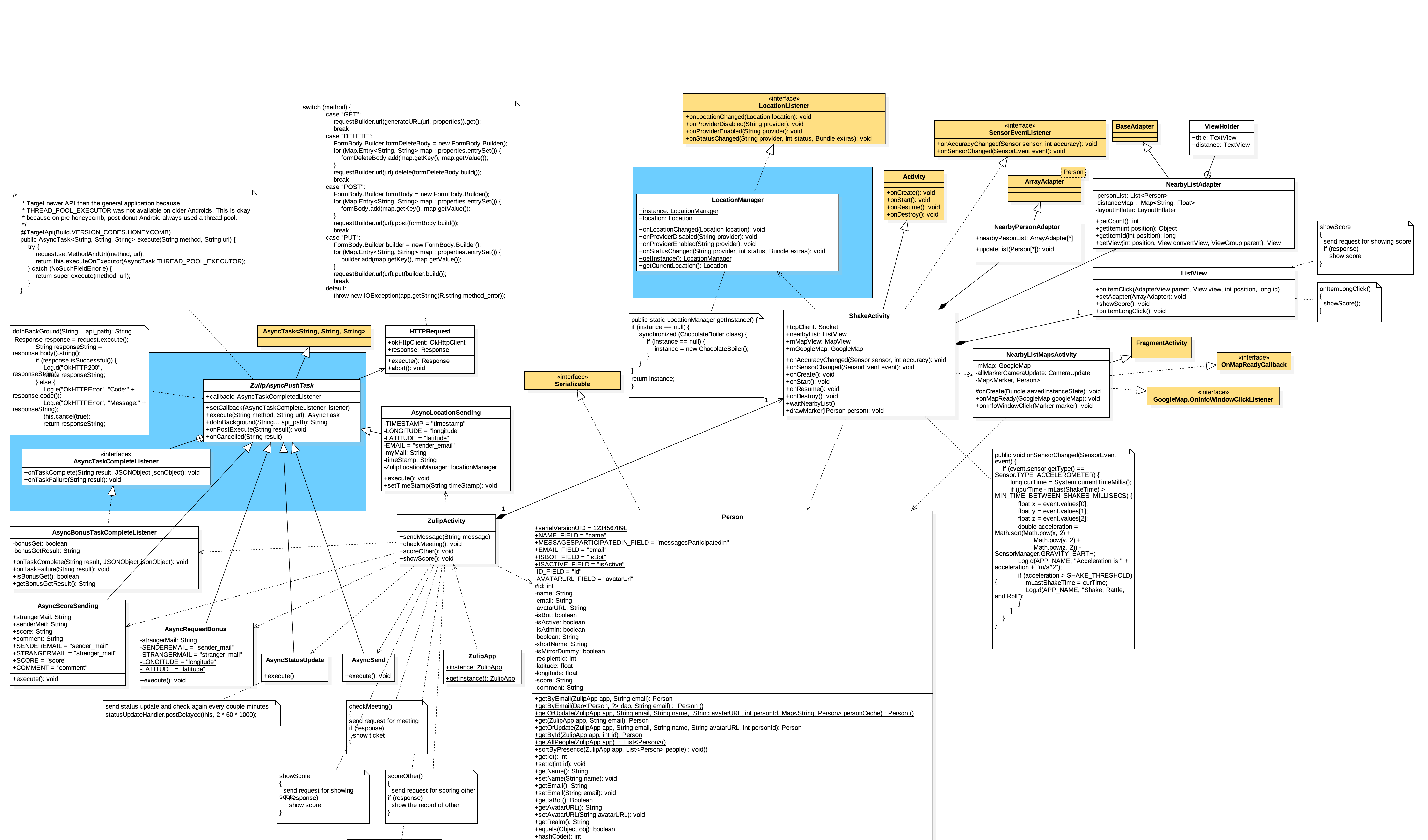
1. 類別圖設計(Class Diagram)

Following are the main functionalities of the system and system architecture:

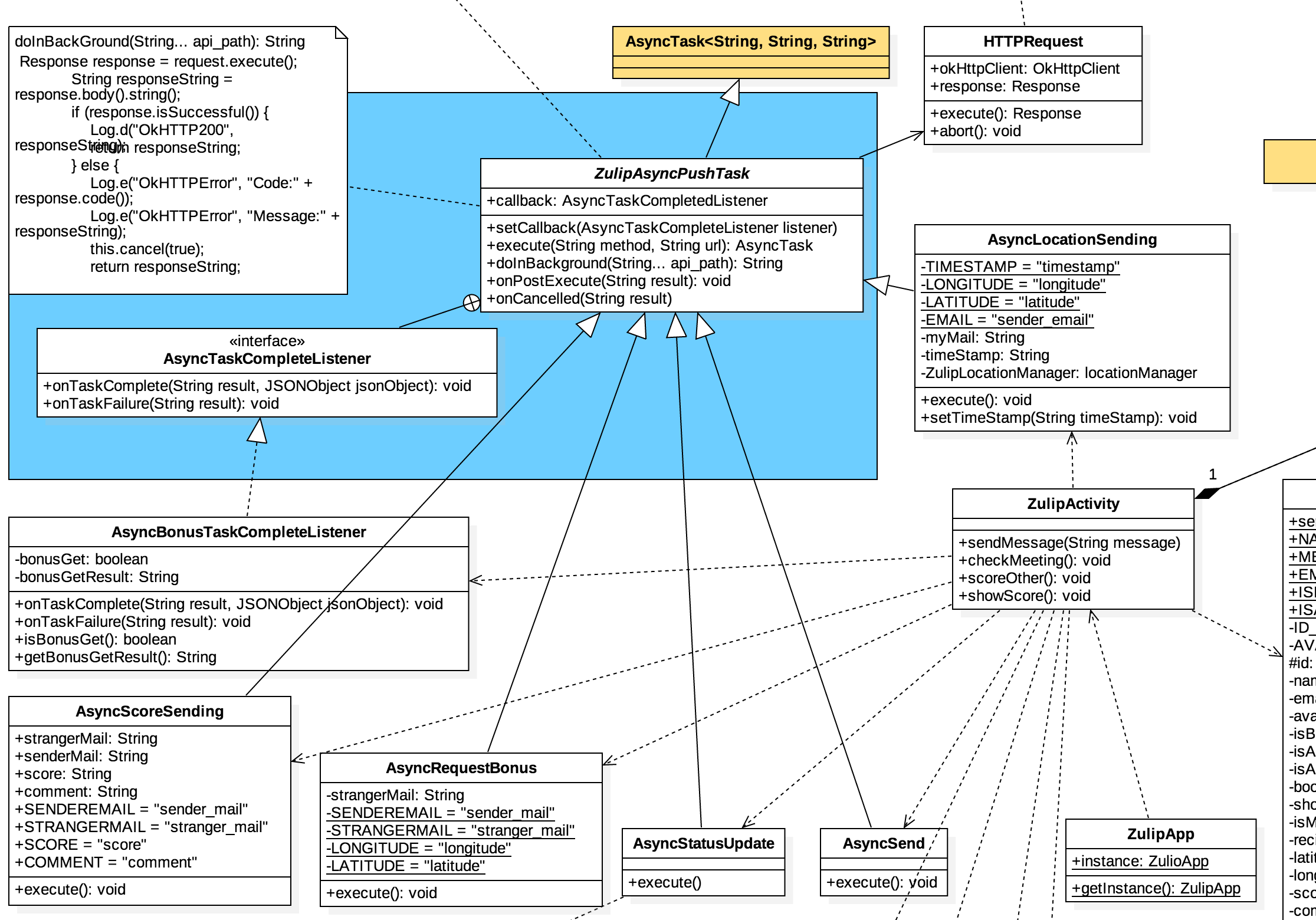
|  |  |
| --- | --- |
| Module | Descriptions |
| Authentication Modulet | Provides login and check user’s state functionality. |
| Shake Module | Provides detecting users motion to send geo-information, showing nearbylist, ensuring chat. |
| Chat Module | Provides message transferring and updating pages. |
| Connection Manager Module | Provides front-end and back-end service establishing. |
| Geo-handler Module | Provides searching nearby people, receiving geo-information, sending response, and refreshing in server. |
| Meet Bonus Module | Give bonus to people who meet each other. |
| Score Module | Score the people after getting bonus. Show the score on the nearby-list. |

1. Front-end Class Diagram:

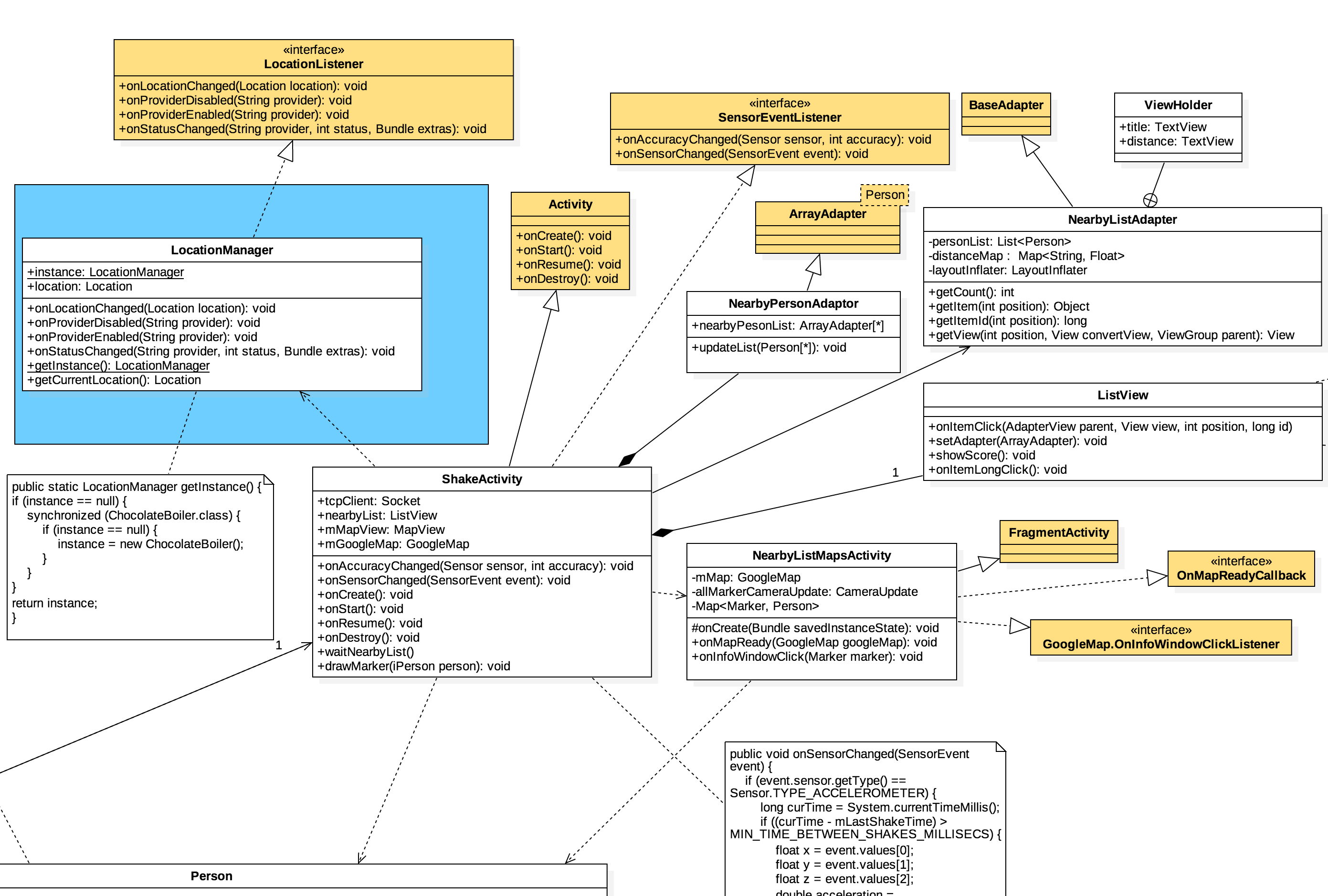
全圖



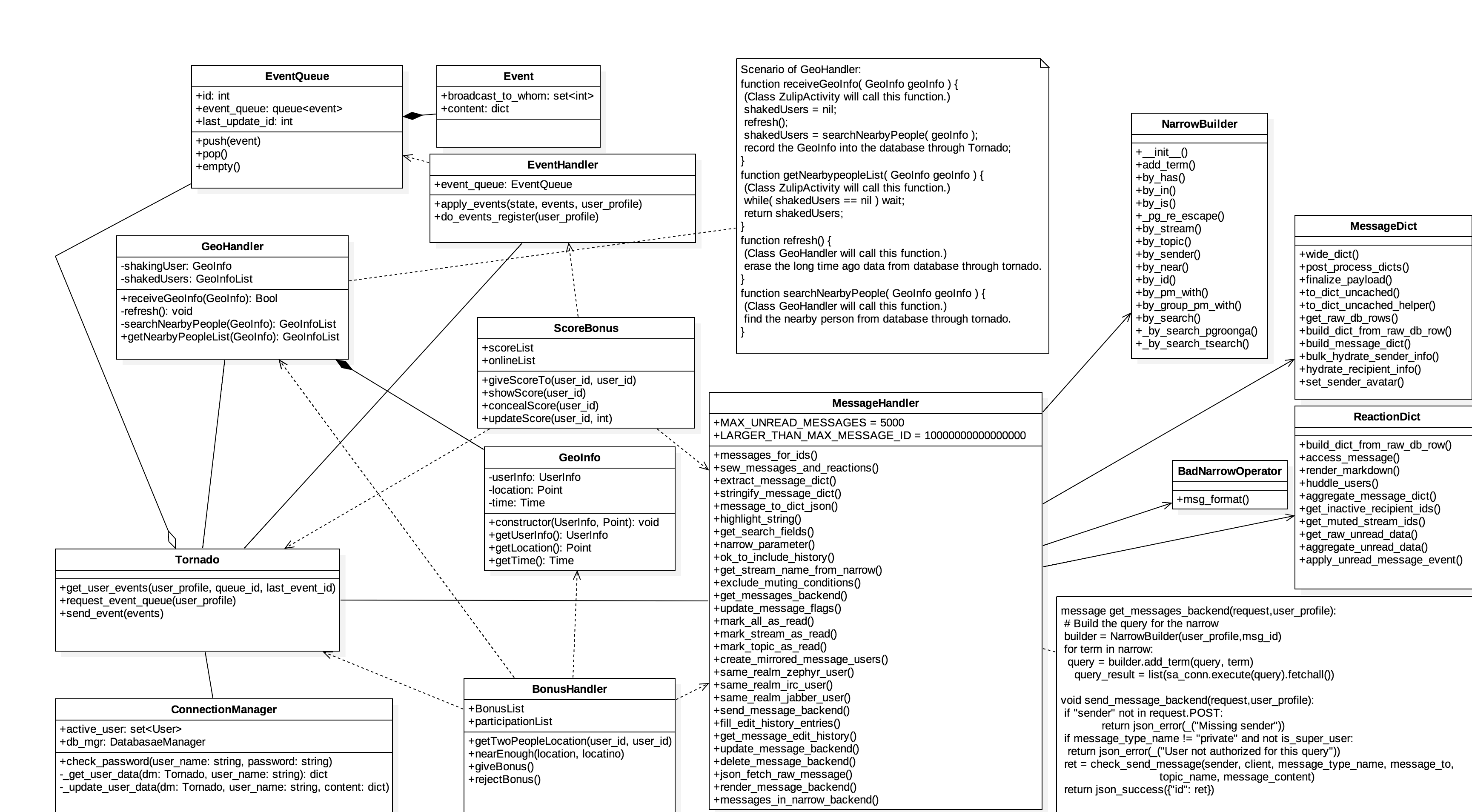
左側放大

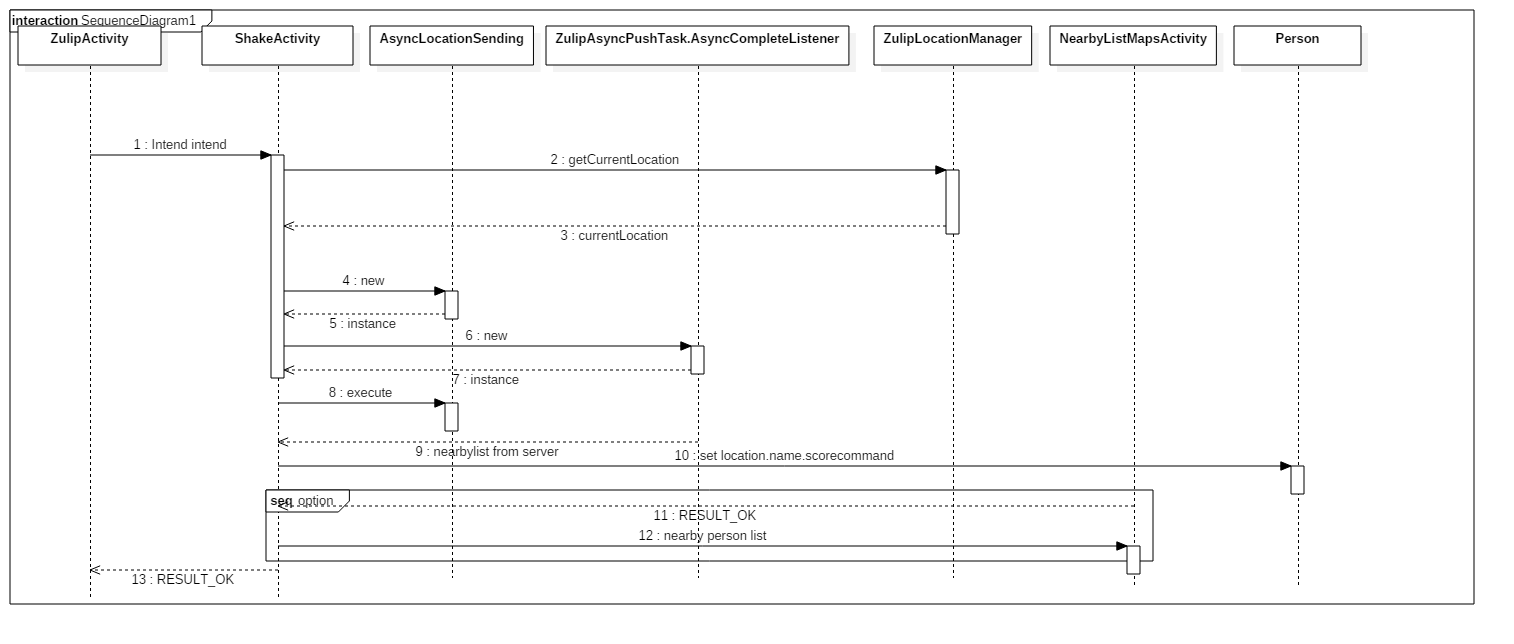


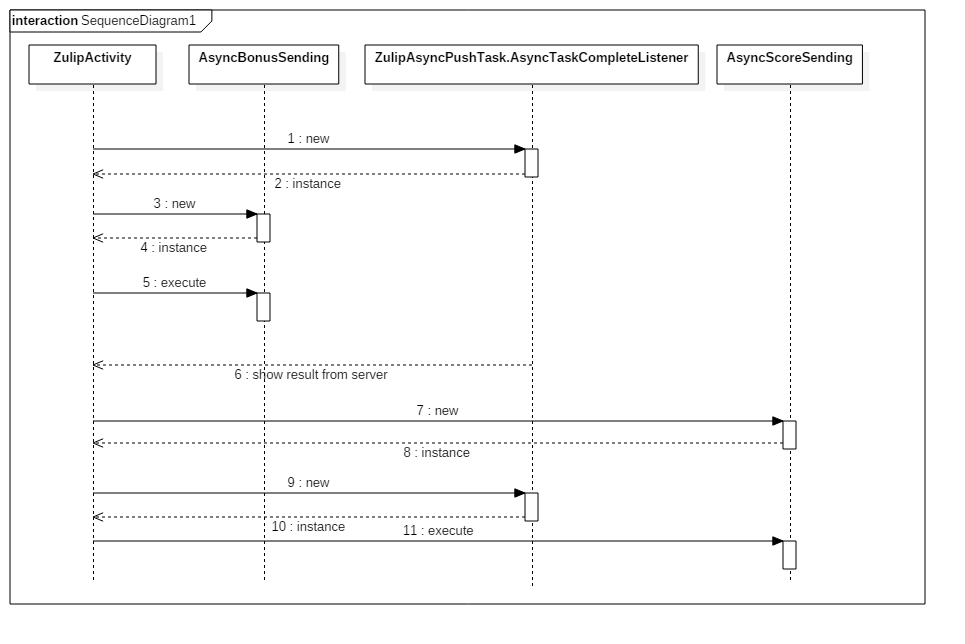
右側放大



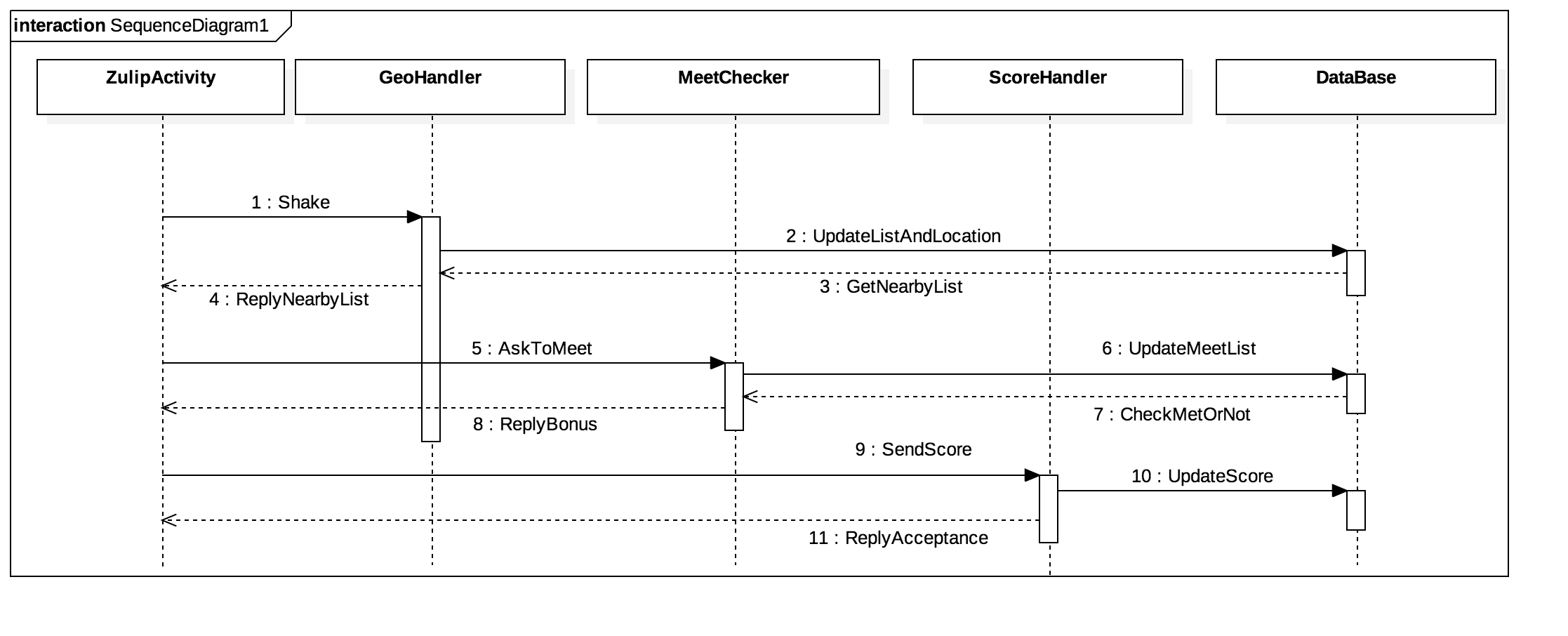
1. Back-end Class Diagram:



1. 循序圖設計(Sequence Diagram)
2. **Shake**
3. **Bonus and score**



1. **Backend**

****

1. 資料細部設計(Data Design)

ShakeActivity

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **tcpClient** | **Socket** |  |
| **nearbyList** | **ListView** |  |

MessageHandler

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **MAX\_UNREAD\_MESSAGES** | **int** | **5000** |
| **LARGER\_THAN\_MAX\_MESSAGE\_ID** | **Int** | **100000000000** |

GeoHandler

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **shakingUser** | **GeoInfo** |  |
| **shakedUsers** | **GeoInfoList** |  |
| **Dg\_mgr** | **DatabaseManager** |  |

GeoHandler

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **shakingUser** | **GeoInfo** |  |
| **shakedUsers** | **GeoInfoList** |  |
| **Dg\_mgr** | **DatabaseManager** |  |

GeoInfo

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **userInfo** | **UserInfo** |  |
| **Location** | **Point** |  |
| **time** | **Time** |  |

EventHandler

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **eventQueue** | **EventQueue** |  |

EventQueue

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **id** | **int** |  |
| **event\_queue** | **queue<event>** |  |
| **last\_update\_id** | **int** |  |

Event

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **broadcast\_to\_whom** | **set<int>** |  |
| **content** | **dict** |  |

ConnectionManager

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **activeUser** | **set<User>** |  |
| **db\_mgr** | **DatabaseManager** |  |

AuthenticationManager

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Type** | **Note** |
| **user** | **UserInfo** |  |