3-Sprints Initial Planning

Streaming event data compliance checking in Python

Jingjing Huo	376323
Sabya Shaikh	384606
Zheqi Lyu	378653

First Sprint: (42% of all functionality - 54h/ 64h)

1. Database manipulate-- 8h
2. Complete Client -- 8h
3. Function for building automata -- 8h
4. Server: Multi-threads handling -- 12h
5. Client logging -- 8h
6. Client exception -- 8h
7. Documentation -- 6h (each member 2h)

Second Sprint: (58% in the second sprint - 73h/ 112h)

1. Remaining Task for Sprint1 - 12h
2. Server: Function for compliance checking with exception raising or handling -- 17h
3. Generate Pdf with exception raising or handling -- 17h
4. Server logging with exception raising or handling -- 8h
5. Server exception handling(remaining) -- 8h
6. Documentation -- 9h (each member 3h)

Third Sprint: (48h/ 48h)

1. Fix Bug -- 20h
2. Refactor --12h
3. Test(Especially Time) and integrate --10h
4. Documents -- 6h (each member 2h)

First Sprint Planning Details

Tasks	Details	Time	Assignee
1.1 Database manipulate	Description: Develop the database tools, so that we can directly use this functions to CRUD data in the database. Create Automata and AlertLog object, which will be used to store the informations in memory.		Zheqi Lyu
	Implementation: 1. Reference constraint, cascade delete. 2. CRUD demo. 3. Create Automata and AlertLog object.	5h	
	Testing: 1. Tables could be automatically created. 2. Automata(alertlog) could record the right information, when the next connection(alert) comes. 3. CRUD	2.5h	
	Documentation	1h	
1.2 Complete Client	Description: Develop a user-friendly interaction, and the user can do step by step as the tips from the console. And the user can interrupt the system, while the server is doing compliance check.		Zheqi Lyu
	Implementation: 1. User-friendly interaction 2. Interrupt-function	5h	
	Testing: 1. Invalid input parameter 2. Invalid manipulate	2.5h	
	Documentation	1h	

1.3 Function for building automata	Description: This function will work on windowMemory for particular event, and calculate the connections of that event for different prefix-automata. For updating particular “connections” a lock is needed to guarantee that the same connection can not be updated by different threads at the same time.		Sabya Shaikh
	Implementation: 1. For each event received and for corresponding all window size create a source node and sink node 2. Using this source node and sink node and create automata 3. Store this data into Connection entity and Nodes entity.	5h	
	Testing: 1. Create and store the data into automata entity for different window sizes -1,2,3,4 2. Node table - update the number of connections	2.5h	
	Documentation	1h	
1.4 Server: Multi-threads handling with	Description: In this part multi-threads for building automata should be implemented, that is for each event we create a new thread and start it. When the event is the first event in one case, a lock for that case will be created, then all threads for this case will check this lock, if the previous event in the same case have released the lock, if not, it must waiting. This mechanism can guarantee that the events in the same case should be executed sequentially.		Jingjing Huo
	Implementation: 1. Whole structure 2. Multi-Threads Demo	9h	

	Testing: 1. If the events in the same case should be executed sequentially in terms of using the “windowMemory” 2. Time, how long it will take for 300 events?	2.5h	
	Documentation 1. Test results 2. How to test	1h	
1.5 Client logging	Description: In this section, logging every activity that happens on the client side shall be logged. There are different activities that will have different formats. For each format a function shall be created to log. For example for a particular event or for other activities like reading file etc. an error, info message, or users’ input everything shall be logged.		Sabya Shaikh
	Implementation: 1. Create Logging object 2. Format the logging messages 3. Write into logging file 4. Adding logging into every function at client side	5h	
	Testing: 1. Actions to be taken if a wrong format given in the message. (not able to parse message) 2. Data being logged into correct file	2.5h	
	Documentation: Logging functions with their description	1h	
1.6 Client exception	Description: In this part all the exception should be considered and solve in a appropriate way		Jingjing Huo
	Implementation:	5h	

	1. Create Exception object 2. Exception handling		
	Testing	2.5h	
	Documentation	1h	