

## Progress Report

Project Name	VMS – Virtual Meeting System
Index No	130614N
Report No	02
Reporting period	2016-04-03 to 2016-05-22
<b>Section 1: Activities and Progress</b>	
<b>1.1 Implementation/ development progress</b> As per the schedule, by 25th of May 2016, development of core functionality, excluding “Responsibility Assignment System” should be completed. But, as per the current progress, development of all the core functionalities (including “Responsibility Assignment System”) has been completed. The initial designing of the system were completed as per the schedule and implementation of the system commenced based on the design. The core functionality of 3 Subsystems, Server, Facilitator and Presenter has been completed. The systems are capable of <b>establishing and managing connection</b> with each other. <b>Pairing mechanism</b> between subsystems are also implemented. <b>Connection termination</b> mechanisms are also implemented. However, automatic connection terminations based on time-out situations are still being implemented. All the subsystems are capable of streaming content between each other. The internally developed <b>livestream</b> component is used for this purpose. Currently, the system is capable of <b>capturing, streaming and re-synthesising</b> both screen and voice using livestream component. The <b>Presenter Selection Logic</b> is fully implemented. Its relevant control mechanisms are also in place and it ensures only the required data is streamed over the network link. The <b>share permission request</b> mechanism is also implemented. All the functionality of “ <b>Responsibility Assignment System</b> ” are implemented in all the Subsystems. The Facilitator Subsystems are capable of viewing and modifying the shared list of Assigned Tasks. Each presenter subsystem is provided with assigned tasks relevant to it. The User Interface of Server Subsystem has been implemented. Currently, the user interfaces of Facilitator Subsystem and Presenter Subsystem are being designed. The project has mostly progressed according to the schedule except for the considerable advancement gained by completion of “Responsibility Assignment System” a week in advance. Thus, currently the project is in ahead of its schedule.	
<b>1.2 Testing progress</b> The system was tested in parallel to development. Separate testing prototypes were developed to test the internally developed livestream component. Logging mechanisms were also utilised to test the system. JUnit unit testing framework was used to cover the code relevant to core functionality. Considering performance testing, the system parameters to be satisfied were identified during the preparation of <b>SRS</b> document. These parameters, categorised as “ <b>Reliability</b> ”, “ <b>Usability</b> ” and “ <b>Performance and Security</b> ” would provide guidance in preparation of QA and System Testing Plans. The testing plan would be developed in the Week starting from 23/05/2016.	

### 1.3 Deviations from proposed architecture and design

There are no significant deviations from the initial design except for the introduction of **Adjourned** State to the Virtual Meeting.

Initial design introduced 4 states to the Virtual Meeting, **Not Ready**, **Waiting for the First Facilitator**, **Waiting for the Second Facilitator** and **Meeting Online**. In this initial design, there was no **adjourned** state since it was planned to span the same **Virtual Meeting Session** across multiple meeting. Thus, following the completion of a Virtual Meeting, the two Facilitator Subsystems would disconnect from the Server and the Server would switch to **Waiting for First Facilitator** State. However, this design was modified due to following reasons.

- The use of Same Virtual Meeting session for multiple meetings can cause security vulnerabilities since the previous facilitators who participated the meeting has session keys. Thus, re-initiating a Virtual Meeting Session following the completion of a virtual meeting is recommended. Thus, it is required to explicitly identify conclusion of a meeting and thus adjourn state is required.
- It is required to have a unique shared tasks list in "Responsibility Assignment System" for each meeting. Thus, it is required to identify the ending of a meeting explicitly in order to clear the shared tasks list.

Currently, this adjourn state is implemented into the system. Now, any facilitator may explicitly request the meeting to be adjourned. This would switch the meeting to the adjourned state. The facilitators would identify that meeting has switched to the adjourned state and then they would disconnect from the Server. When all the facilitators connected to the Server are disconnected, the server destroys the previous Virtual Meeting Session and re-starts a new session.

## Section 2: Risks, Issues and Challenges

Currently, most of the risks has been surpassed since majority of implementation is already completed. The preliminary testing has shown that system is meeting its non-functional requirements as per the current optimisations. However, the following risk are identified in future development.

- The complexity of user interface required for Facilitator Subsystem is high. This complexity may reduce the quality of the user interface. As a remedy, it is decided to design mock-up interfaces for Facilitator Subsystem prior to implementing final User Interface.
- The system is identified to be within network bandwidth constraints during the testing undertaken in development stage. However, the dynamics of a VPN over public internet/WAN is not considered in these tests. However, a significant margin of safety is available to modify system parameters in order to accommodate for these dynamics.

## Section 3: Next Steps

- **Automatic Connection Termination** – Automatic connection termination system should be implemented in-order to disconnect the systems that stop responding. The server and facilitator subsystems should receive a periodic signal from facilitator and presenter subsystems indicating their presence. If such a signal is not received for a significant time period, they should be considered as disconnected. Furthermore, the Facilitator and Presenter subsystems should be able to reach Server and Facilitator subsystems. In an event where continues such attempts fail to reach endpoint, the connection should be considered as terminated. The implementation of automatic connection termination would be an extension of existing disconnection procedure.
- **Finishing User Interfaces** – The User Interfaces of Facilitator Subsystem and Presenter Subsystem should be designed and implemented.
- **Attachment of Persistent Storage** – Currently, the interfaces required to feed information between subsystems and persistent storage are present. However, logic regarding serialisation and storage of these information is not implemented. The information stored in persistent storage includes Paired Device Details and System Configurations.
- **Testing** – Testing plans should be developed and tests should be undertaken on all 3 subsystems. These tests should not only cover the logic but should also cover the performance characteristics of the system.