Company Merger

Infrastructure Integration Project

Deliverable 4: Data Modeling and Starting Design

IS 436

Muhammad Hamza - Project Manager

[ham15@umbc.edu](mailto:ham15@umbc.edu)

443-889-8146

Siril Pattammady - Systems Developer

[psiril1@umbc.edu](mailto:psiril1@umbc.edu)

301-323-3245

Josh Johnson - Systems Analyst

[jjohn3@umbc.edu](mailto:jjohn3@umbc.edu)

240-786-8420

Khadija Shafiq - Business Analyst

[kshafiq1@umbc.edu](mailto:kshafiq1@umbc.edu)

410-508-5849

Jay Patel - Network Engineer

[jpa2@umbc.edu](mailto:jpa2@umbc.edu)

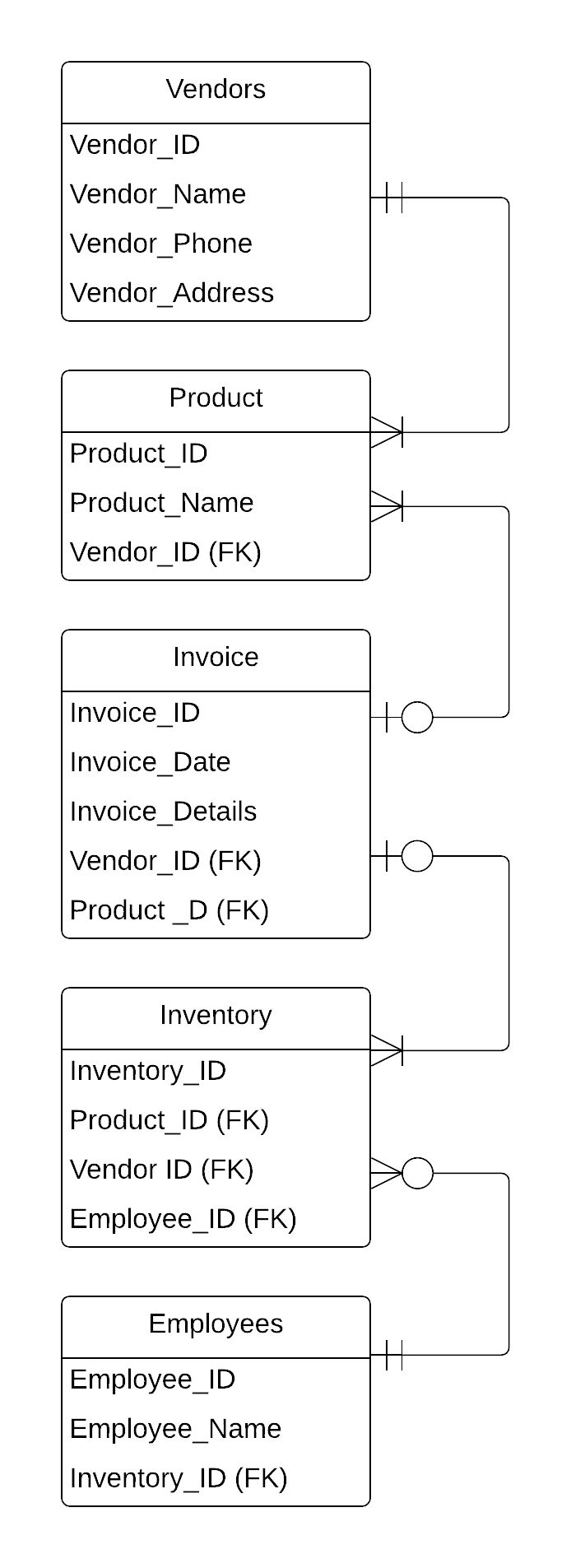
410-428-8465

Shaikha Al Shamsi - Requirement Analyst

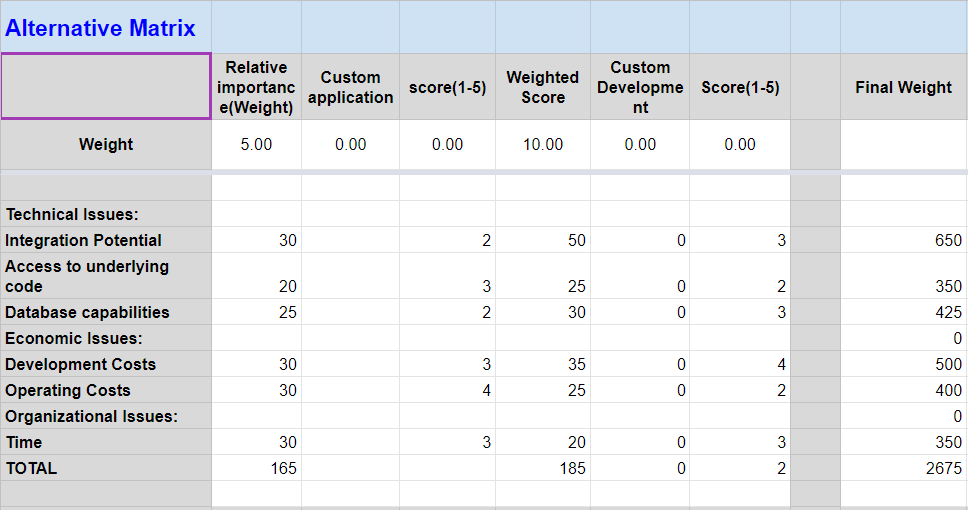
[shaikha1@umbc.edu](mailto:shaikha1@umbc.edu)

443-515-9765

1) Develop a data model by **drawing an Entity-Relationship diagram** using Visio, Lucid Charts (or any drawing tool) The model will be accompanied by a textual description of entities and relationships. The diagrams should closely follow the notation. Make sure that the model is in the third normal form.



2) **Develop an alternative matrix** for your project. You must also **provide a description of each of the alternatives that you have chosen.** Each member of the team will rate the three alternatives independently. **Include each individual matrix with the deliverable.** The team will then come together on a consensus, creating a **team matrix**. Include this team matrix with the deliverable. **Provide a narrative for the justification about which alternative was picked by your team**



[Simplified Decision Matrix](https://docs.google.com/spreadsheets/d/1RIsw7rVS3FSKij5um7IvmS5o4KHSx8qknUTeyqOrXaA/edit?usp=sharing)4

3) **Design the architecture of the system by to developing a matrix** . In its rows, this matrix should clearly list the **non-functional requirements of your project under four main categories: Operational, Performance, Security, and Cultural/Political** requirements. In its columns, the matrix should include the **architectural options (i.e. server-based, client-based, thin-client server, thick client server).** If a particular architecture is good for any requirements put a check mark in the corresponding matrix cell. This matrix should be accompanied by a narrative that talks about why particular architectures are a good fit for particular non-functional requirements. Based on this matrix, **make a decision of the system architecture and explain the justification. Mention the trade-offs and the reasoning behind your decision. After that, develop a Hardware and Software Specification. Note that your specifications can include different server and client configurations.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Non-Functional Requirements** | **Server-Based** | **Client-Server Based** | **Thin Client-Server** |
| **Operational Requirements** |  |  |  |
| **Systems Integration Requirements** | **✔** | **✔** | **✔** |
| **Portability Requirements** |  | **✔** | **✔** |
| **Maintainability Requirements** | **✔** |  |  |
| **Performance Requirements** |  |  |  |
| **Speed Requirements** |  | **✔** | **✔** |
| **Capacity Requirements** |  | **✔** | **✔** |
| **Availability/Reliability Requirements** | **✔** | **✔** | **✔** |
| **Security Requirements** |  |  |  |
| **High System Value** | **✔** | **✔** | **✔** |
| **Access Control Requirements** | **✔** | **✔** |  |
| **Encryption/Authentication Requirements** |  | **✔** | **✔** |
| **Virus Control Requirements** | **✔** |  |  |
| **Cultural/Political Requirements** |  |  |  |
| **Multilingual Requirements** |  | **✔** | **✔** |
| **Customization Requirements** |  | **✔** | **✔** |
| **Making Unstated Norms Explicit** |  | **✔** |  |
| **Legal Requirements** | **✔** | **✔** | **✔** |

**Client-Server Architecture is most compatible with our design.**

**Server Based:**

Includes no additional client used only terminal and server computer. Using a virtual desktop infrastructure provides benefits with security requirements as it handles storage and logic fairly well. However there are downsides in terms or performance and cultural/political requirements. There are disadvantages in our integration since clients are needed to make changes whether its based on DB management or client requests.

**Client-Server Based:**

A balanced process between client device and server devices. This implementation seems to work the best with our architecture since clients are responsible for logic and decision making while the server is responsible for data access/storage. In our architecture our clients make requests based on our database storage which is stored using server (to be cloud based). This model satisfies most of the requirements from security since it is highly reliable, performance, operational, and a common cultural/political requirements.

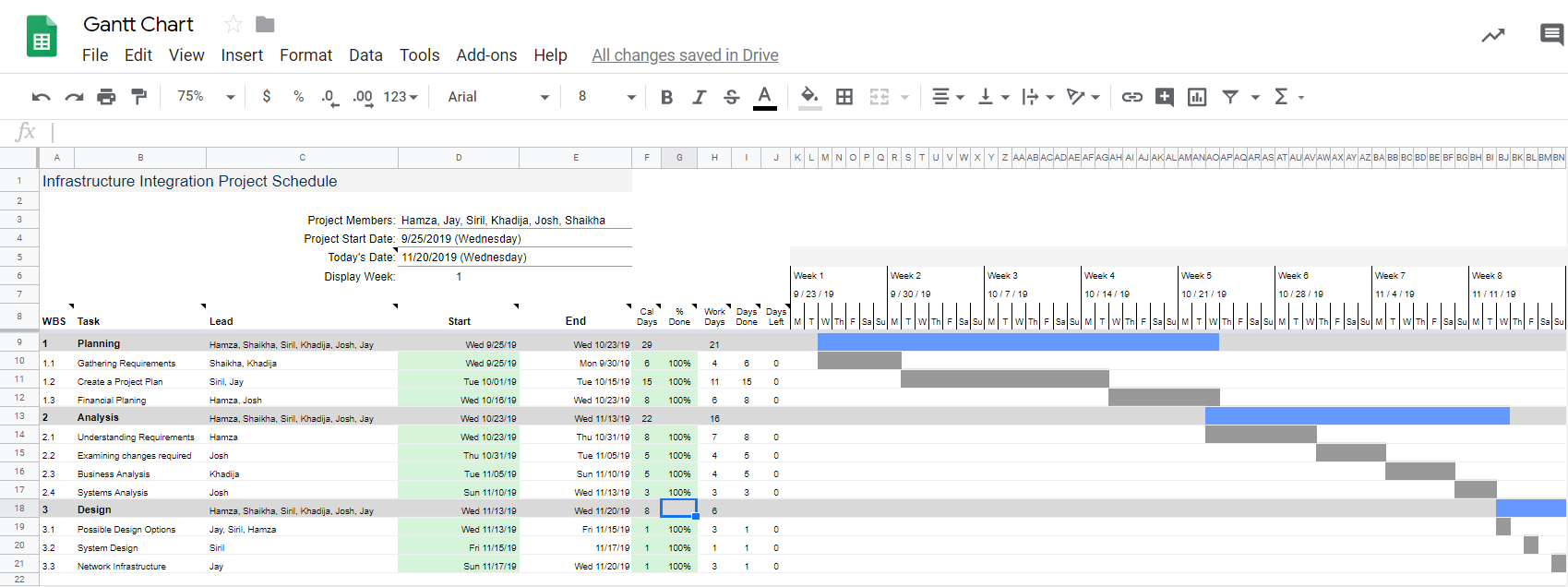
**Thin Client Server:**

Similar to client server but doesn’t cover all the aspects that a typical client-server based architecture would. Does not cover all the cultural/political requirements and security requirements, but satisfies most of the performance and operational requirements.

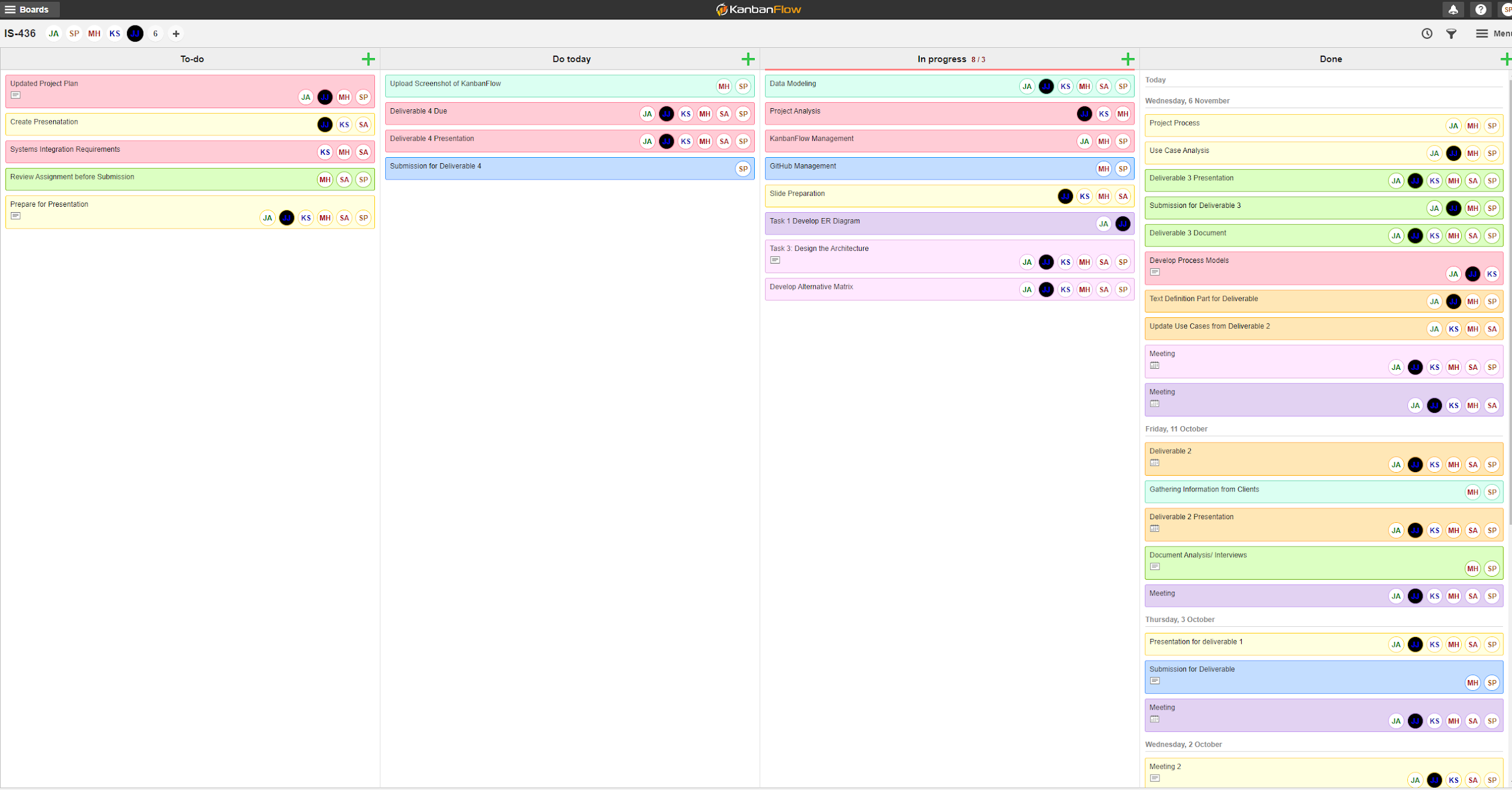
**Hardware/Software Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Standard Client** | **Standard Web Server** | **Standard Application Server** | **Standard Database Server** |
| **Operation Systems** | Windows 10 | Linux | Linux | Linux |
| **Special Softwares** | - Adobe Acrobat Reader/DC  -Microsoft Office  - Antivirus  - Cloud Client | Apache | Java | SAP  QAD |
| **Hardware** | HP Laptops  Intel Core i7 Processor 1TB HDD | 1TB Disk Drive | 1TB Disk Drive | 1-TB Disk Drive  RAID |
| **Network** | Always-on-  Broadband, preferred | Fast Ethernet: 100Mbps LAN | Fast Ethernet: 100Mbps LAN | Fast Ethernet: 100Mbps LAN |

4) Include your updated project plan.



5) On Kanban board assign the requirements to your team members



6) System Integration .Please follow the instructions provided on https://userpages.umbc.edu/~ss12/IS436/content/groupproject/group.html under system integration section

SHARED DOC:

Hey Everyone! It’s my understanding your system is moving from local servers to cloud? And issuing new laptops?

With that, we could use a system to assist with existing electronic medical record systems, EMRs, to be backed up to the cloud to manage patient’s records. Also, if you’ve gone as far as finding a way to upgrade laptops so they manage and communicate with current systems and the new cloud system that could be of interest to your group. Just tossing ideas…

With this, medical records need to be HIPPA certified. Microsoft sharepoint and sharefile which are cloud based solutions which are easily implemented in new laptops, is a good way to look at system integration. Sharepoint and Sharefile work alongside OneDrive to assist in a cloud based environment.

I use OneDrive and SharePoint with work all the time to manage medical records, so yes, it is secure and HIPAA-compliant. How shall we have out systems communicate with this common interest? Have any idea how he wants us to present this integration?

Perhaps our system can trigger your system to pull medical data from a local server and transition onto the cloud?

Perhaps instead of local storage, typically you would be using network drives, to make that switch to a cloud based data storage option would be ideal.

Maybe a document control solution (document management solution) for inhouse records that need to be put into a cloud environment.

Sounds perfect to me.

* Using the system to assist with existing electronic medical record systems, EMRs, to be backed up to the cloud to manage patient records
* Document control solution (document management solution) for inhouse records to be put in a cloud environment
* Using network drives, to make that switch to a cloud based data storage instead of using local storage
* Common data format