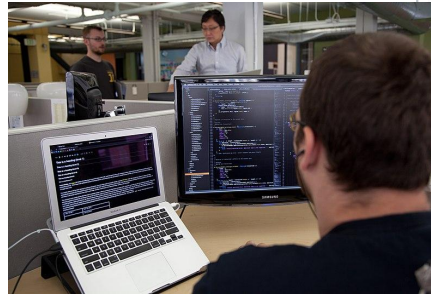
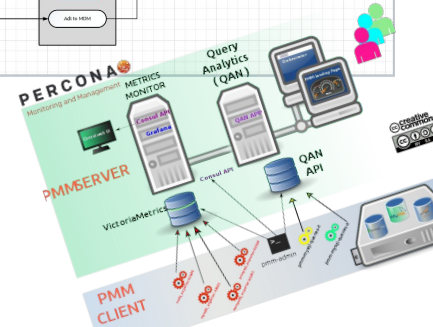
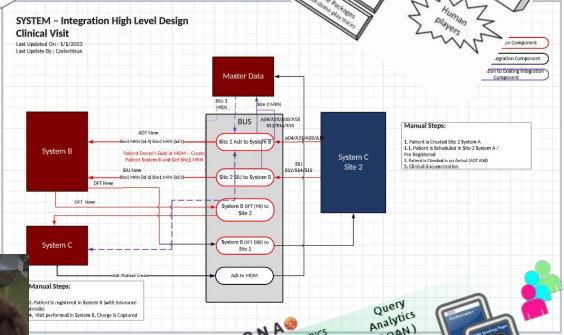
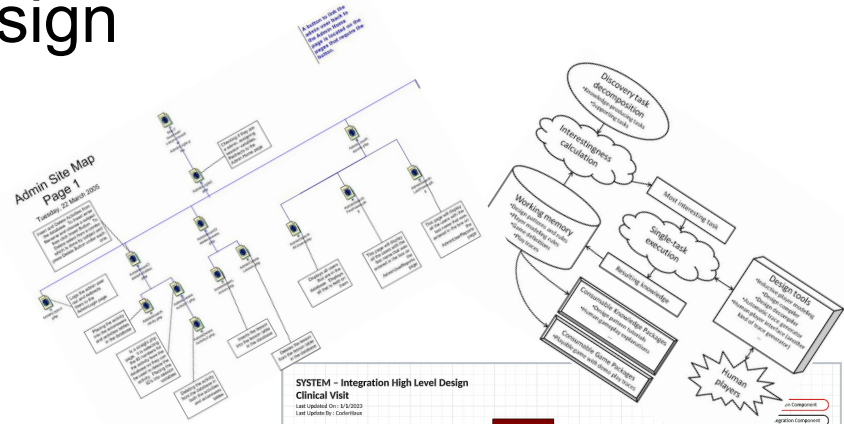
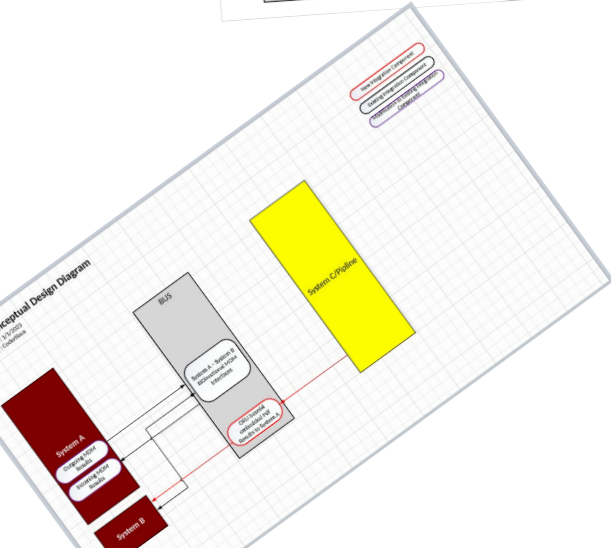
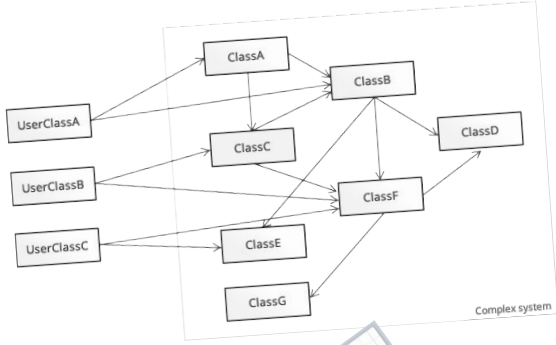


# Architecture and architecture design



Why?

What?

How?

# Why?

What business questions are we answering?

What are the business requirements?

What is the budget?

How do we know we've met the goal? (metrics)

How will users use the system? (user stories)

# What?

How will we accomplish the goals of the project?

What does our HLD, High Level Design, look like?

How does the data move through the system? Flow Diagrams

What is the state of the system, at any point in time? State Diagrams

# How?

What technologies will we use?

How are the technologies interconnected?

How are user workflows affected?

What are the specifics of each technology?

OnPrem vs Cloud

VM vs Bare Metal vs Containerized

# What is Enterprise Architecture?

More than drawing pretty pictures!

Documentation is a large part of the work.

Helps to understand

What are the operational workflows?

What needs aren't being met?

What's really being asked for?

How do we (IT) deliver?

# Key questions in getting to WHY

What is the purpose of the project?

Isn't always obvious!

Use the 5 whys!

What are the main objectives?

Drives the metrics for the project.

Helps us know when we're done done.

Let's us know, post project, if it's still meeting a need.

# Key questions in getting to WHY Cont.

Who are the users of the system?

Includes “downstream” users.

Includes “downstream” system owners!

What does the UX feel like?

Need to be aware of organizational requirements, or guidelines.

Understand requirements for things like: Colors, Fonts, Logos



# Key questions in getting to WHY Cont.

Two important questions:

What is the budget?

What is the timeline?

Expectations need to be inline with reality!

What are the pain points with the current work?

How will a new system impact current work and technology?

What are the future growth expectations, related to the implementation?

# Key questions in getting to WHY Cont.

What are the ROI metrics we're trying to hit?

Who are the key decision makers?

Who approves the design?

Who authorizes changes to the design?

When do you utilize these decision makers?

Check back often, to validate your assumptions!

# Key questions in getting to WHAT

Start to build detail into the plan.

Bridge the gap between need and technology.

Analytics

What types of reports are needed?

What is batch, what is real time?

# Key questions in getting to WHAT cont.

## Business Continuity

How the business operates, when systems are down.

Availability (from big three, Availability, Scalability, Security)

High Availability solutions mitigate throughput issues or short term outages.

Disaster Recovery solutions mitigate site outages.

Paper process solutions mitigate similarly for DR, do NOT use technology solution at all.

Going out of business, shutting down temporarily CAN be a proper solution!

Technology solutions can be EXPENSIVE, COMPLEX, and INCREASE time to deliver.

# Key questions in getting to WHAT cont.

What are the integration needs?

Specifically high throughput like IOT or medical devices.

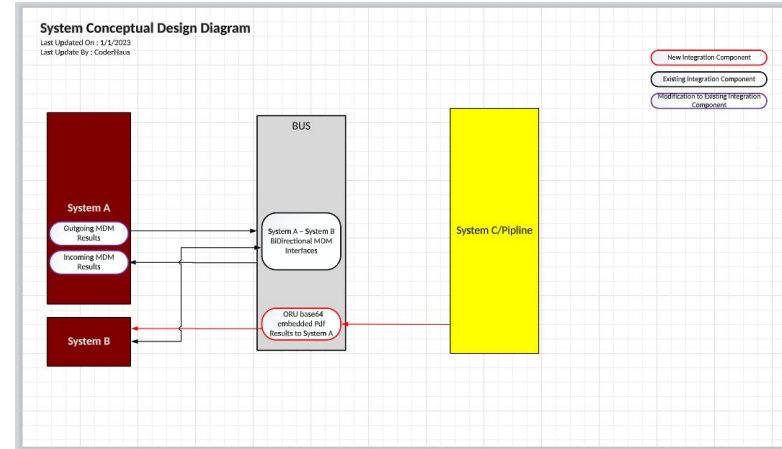
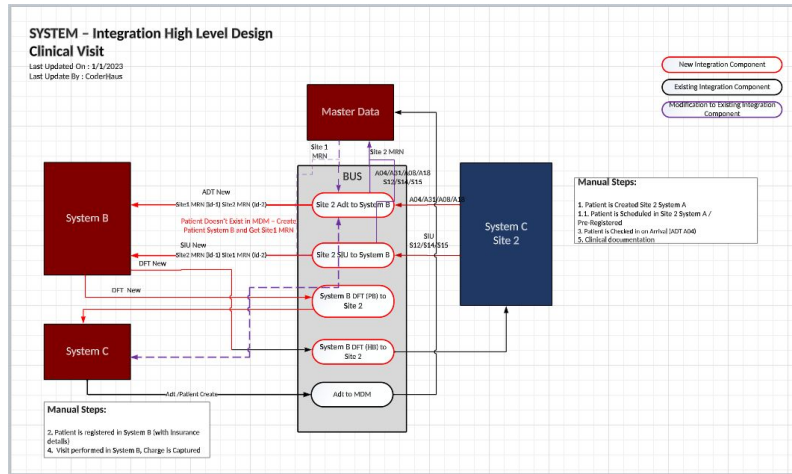
Requires more care in architecting a solution that can handle throughput.

What are the data ingestion needs?

How is the data processed? Is it taken in and spit out without modification?

Does the data need to hit in real time? Can it queue or batch?

# Get to the details!



# Key questions in getting to How

What are the specific:

Systems

Users

Workflows

Impacted by the project?

This helps to determine EXACTLY what work has to be done.

# Key questions in getting to How cont.

Determining the App Footprint.

What systems are going to be impacted?

This can be through changes in integration, build, workflows, hardware or other.

Informs whether you need to buy, build, or have something already.



# Key questions in getting to How cont.

Hardware purchasing decisions start to unfold.

This could include datacenter or user hardware.

What does each component impacted DO in this system?

What interactions, specifically, are taking place?

# Key questions in getting to How cont.

What are the functional requirements?

What functions does a system need to provide?

How do those functions meet the goals of the project?

What workflows or use cases are supported by the systems?

# Key questions in getting to How cont.

What are the non-functional requirements?

Specifically, what response time, latency, throughput needs are there?

What are the scalability, availability, security needs?

This all informs compute/storage needs.

Integration!!!

What do we need to interoperate with?

# Key questions in getting to How cont.

What are our data requirements?

What are our storage needs?

What types of data are stored?

How much data storage is needed? What are the growth patterns?

How long will data be stored for?

Are there regulatory needs for data storage?

How do we protect our, and our user's, data?

# Key questions in getting to How cont.

We begin to understand, at deeper and deeper levels, what our needs are.

Where will this be hosted? OnPrem, Cloud

What are the network bandwidth needs?

What is our availability stance?

What type of application are we hosting? Web, Mobile, Desktop, Backend

What is our server platform? VM, Containers, bare metal, serverless?

What types of software? Web, Database, App, API, other?

What are the specific storage needs? Fast vs Slow

# Key questions in getting to How cont.

Post live, how do we support the solution?

How do support folks access the servers?

How do users access the application?

What are the security requirements?

Microsegmentation, VIP, Firewalls, SEIM, B2B or B2C

How do we retire and archive data, and when?

What other work, now or in the future, will affect this project?