Checkpoint #3 Notes:

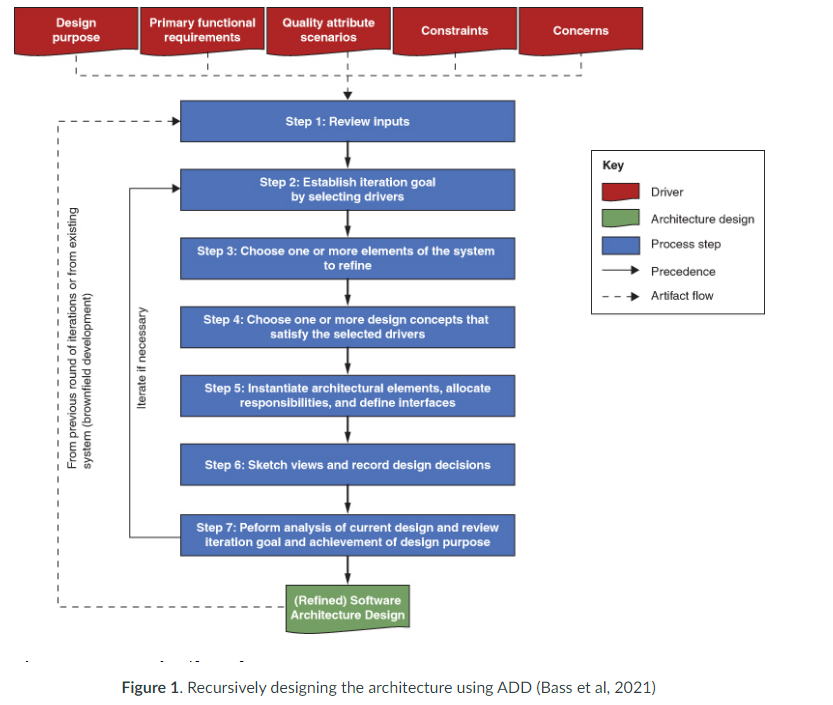
At this stage your team has gathered all the architectural drivers for your system. Now following the ADD process, your team will design the architecture of your system showing clearly:

* How design concepts are used to address the design objectives of the architecture drivers.
* How responsibilities are allocated to the components created as a part of the recursive. decomposition process. In particular, show the element interaction design for primary use cases.
* How tradeoffs if any are handled in the design.

**Rubric**:

|  |  |
| --- | --- |
| How design concepts are used to address the design concerns of the architecture drivers.  The ADD process works by looking at each architectural driver and making a decision on which design concept to use on which part of the system to address the concerns related to that driver. This results in adding new components and architectural capabilities. | **4 pts**  **Full Marks**  How design concepts are used to address the design concerns of the architecture drivers IS clear. |
| How functionality is allocated to the components created as a part of the recursive decomposition process  It should be clear applying which design concepts gives rise to which module /component, and what architectural and functional responsibilities does that module handle. In particular, show the element interaction design for primary use cases. | **4 pts**  **Full Marks**  How functionality is allocated to the components created as a part of the recursive decomposition process IS clear. |
| How tradeoffs are handled in the design  Usually when we select some tactics to implement quality A; we may be hurting quality B due to this selection. Then we need to add more tactics for quality B to reduce the negative impact. Your tradeoff analysis should identify such conflicts and describe how they are resolved. | **2 pts**  **Full Marks**  Discussion on how tradeoffs are handled in the design IS clear. |

-----



### **Architects use the following steps when designing an architecture using the ADD method:**

**Step 1: Review inputs**

This step ensures that we have a good grasp on the problem we are trying to solve. This means, we understand the drivers that would be significant in designing the architecture of a system. Recall the discussion from our previous lesson on different categories of drivers that include design purpose, primary functionality, quality attributes, constraints, and architectural concerns.

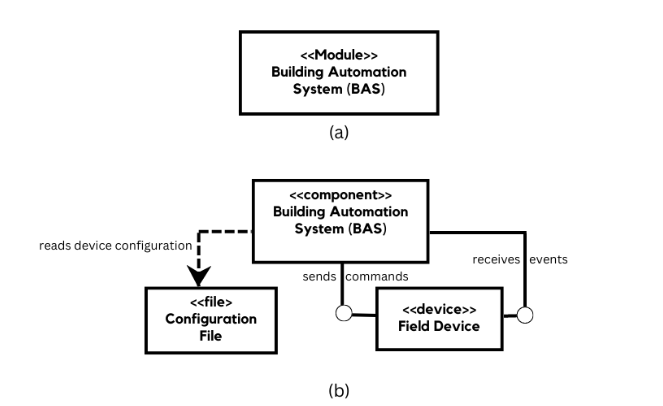
* + 1. **Design purpose:** The effort expended on designing the architecture of a system will depend on the purpose for which it is undertaken. Going through a proof-of-concept exercise or producing enough architecture for cost and effort estimation in response to a request for proposal will require much less effort than designing one for full scale development of a system.
    2. **Primary functionality:** These include the functions within the system that are business critical and have significant quality attribute concerns associated with them.
    3. **Quality attributes:** Also referred to as non-functional requirements, these define qualities (captured using quality attribute scenarios) a module must exhibit during development (e.g., maintainability or testability) or when performing its functions (e.g., performance, security, or reliability).
    4. **Constraints:** These limit the ability of an architect to make certain decisions when designing a module. For example, an architect may be asked to use a particular technology or a particular legacy component.
    5. **Architectural concerns:** Architects from their experience are aware of certain architectural concerns such as logging, exception handling, security, etc. that are significant and need to be designed for in most systems.

**Step 2: Establish iteration goal by selecting drivers**

Rather than addressing all drivers at once, we divide and conquer by selecting a subset of drivers to work on during each iteration.

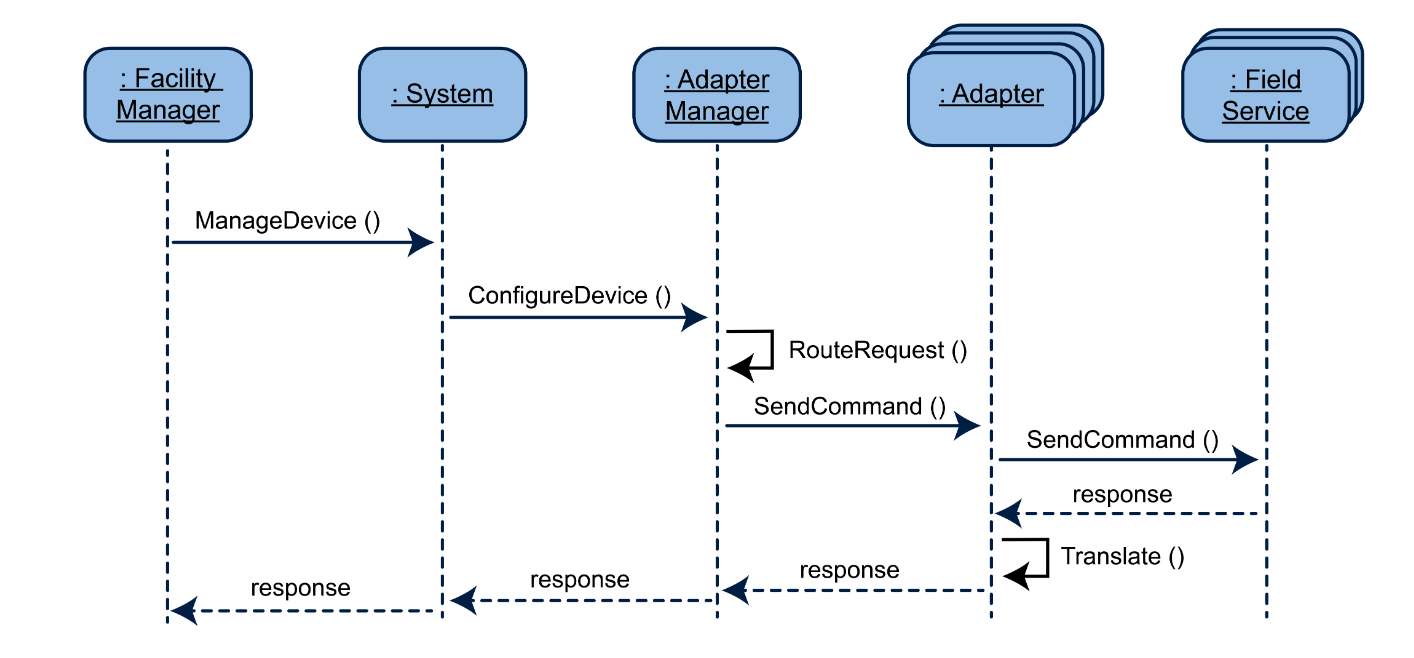
**Step 3: Choose one or more elements of the system to refine**

On a greenfield project, the module to start with is usually the whole system. After the system has gone through the initial decomposition in the first iteration, the focus in subsequent iterations will shift to one or more of its elements that need to be decomposed to satisfy the subset of drivers selected for that iteration. In a brownfield project, the starting point in the existing system will be one or more of its elements that should be the focus of decomposition to satisfy the chosen subset of drivers. EXAMPLE:



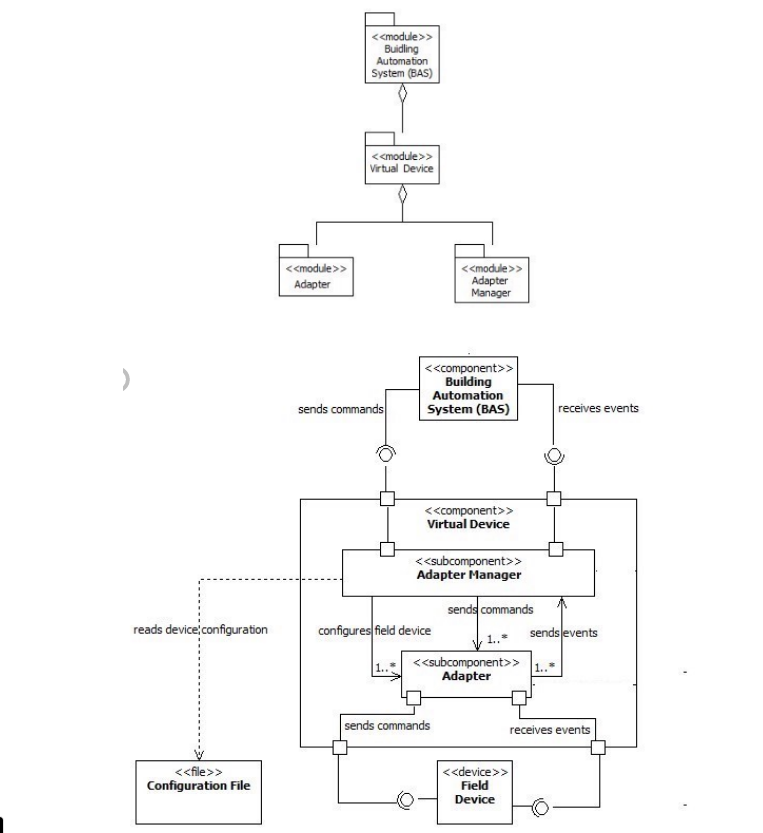
**Step 4: Choose one or more design concepts that satisfy the selected drivers**

Choose from a collection of design concepts (such as reference architectures, patterns, tactics or externally developed components discussed in the last lesson) that can be used to address the subset of architectural drivers for this iteration. EXAMPLE:



**Step 5: Instantiate architectural elements, allocate responsibilities and define interfaces**

Decompose the chosen elements of the system into child elements by applying the chosen design concepts, allocate functionality to the child elements, and define their interfaces. EXAMPLE:



**Step 6: Sketch views and record design decisions**

Lightweight design fragments, that are a byproduct of step five, are captured and design decisions that went into creating that design fragment are also recorded. An element catalog is also created that enumerates all the elements in the design fragment along with their responsibilities.

**Step 7: Perform analysis of current design and review iteration goal and achievement of design purpose**

At this stage you have created a partial design that achieves the goal established for the current iteration. Making sure that this is the case is important and can be achieved through a review potentially involving someone other than yourself. An independent analysis will be unbiased because someone else would not share the same assumptions and will offer a different perspective on the provided solution. We will discuss architecture analysis technique in a forthcoming lesson.

***Repeat steps 2 through 7 above as necessary to address the remaining drivers on the architecture design backlog.***

Brian: I think the goal for this week is to reiterate the “Roadmaps guide to your design” for this week’s checkpoint 3. I think we should recreate the tables, drawings, etc. From this section. Seems fairly straightforward. The only complicated part could be illustrating. Maybe we can use powerpoint to show this, then copy/paste the image into our word doc when we submit it.

From the 3pm Zoom Meeting:

* We do not have to address all architectural drivers for this week’s checkpoint, however, must be addressed by the Final Turn in.
  + If a driver is not covered for this turn-in, we must specify which one was missed and that it would be covered by the Final submission
* Professor Kassab wants us to follow Steps 1-7 for submission
* Prefers UML diagrams
* If our own diagram is created (such as on powerpoint), we must have a legend describing the image/shapes/arrows/lines/etc.

Note: I am Referencing the “Novel Domain” from the Canvas Reading: If we want to use the mature domain, let me know, and I will stop editing!

To Fill out this Table, I referenced Checkpoint #2:

Step 1: Review Inputs:

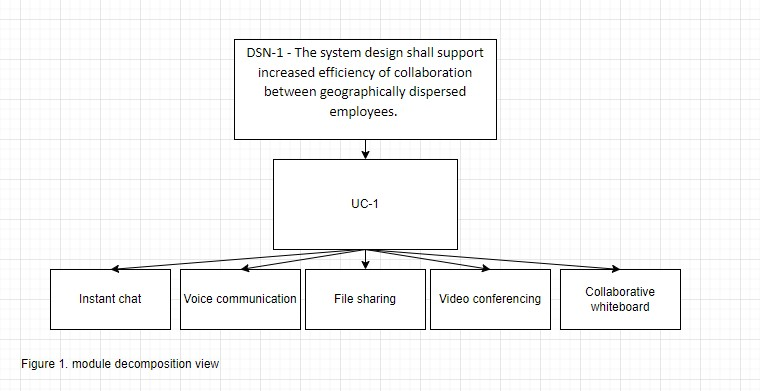
|  |  |  |
| --- | --- | --- |
| **Table 1: Architectural Drivers for Internet-based Collaborative Work Environment** | | |
| **ID** | **Architectural Drivers** | |
| **Design Purpose** | | |
| **DSN-1** | The system design shall support increased efficiency of collaboration between geographically dispersed employees | |
| **DSN-2** | The system design shall allow for the product to differentiate itself in the product in a competitive market | |
| **DSN-3** | The system shall reduce employee downtime and increase productivity | |
| **Primary Functional Requirements** | | |
| **UC-1** | The system shall support 5 different services including voice communication, video conferencing, instant chat, file sharing, and collaborative whiteboarding. | |
| **UC-2** | Real-time features such as voice, video, and whiteboarding shall have a response time of no more than 100ms. | |
| **UC-3** | The system must be profitable | |
| **Quality Attribute Requirements** | | |
| **QA-1** | Usability - A user attempts to access any of the 5 services. They are able to connect and use any of the services at any time. | (H, H) |
| **QA-2** | Performance - The system detects a change in connection strength or availability while a user is using a service. It will try to reconfigure the settings for an optimal connection within 10 microseconds or it will alert the user of possible network disruption. | (M, M) |
| **QA-3** | Performance - When a users’ data in any of the 5 services is sent to the server, it will be compressed and then decompressed once the client receives it. No packets will be dropped. | (L, M) |
| **QA-4** | Security - A user will sign into the product and the system will verify the credentials. Once authenticated, only features that are within the users’ role permission will be available. | (H, H) |
| **Constraints** | | |
| **CON-1** | The system shall be able to support a variety of hardware and operating system inputs and outputs | |
| **CON-2** | Any registered user must be over the age of 13 | |
| **Architectural Concerns** | | |
| **CRN-1** | The 5 system components shall be developed and run independently within the system | |
| **CRN-2** | The system shall be continuously and quickly deployable | |
| **CRN-3** | Metadata for the system shall be stored in a database | |

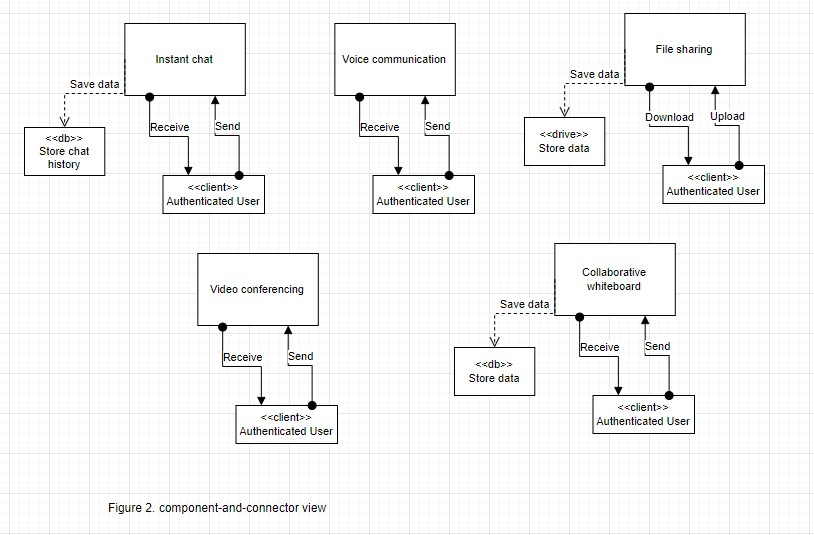
**Step 2: Establish Iteration Goal by Selecting Drivers**

* DSN-1 - The system design shall support increased efficiency of collaboration between geographically dispersed employees.

**Step 3: Choose one or more elements of the system to refine**

* We will be refining the whole system for this iteration.





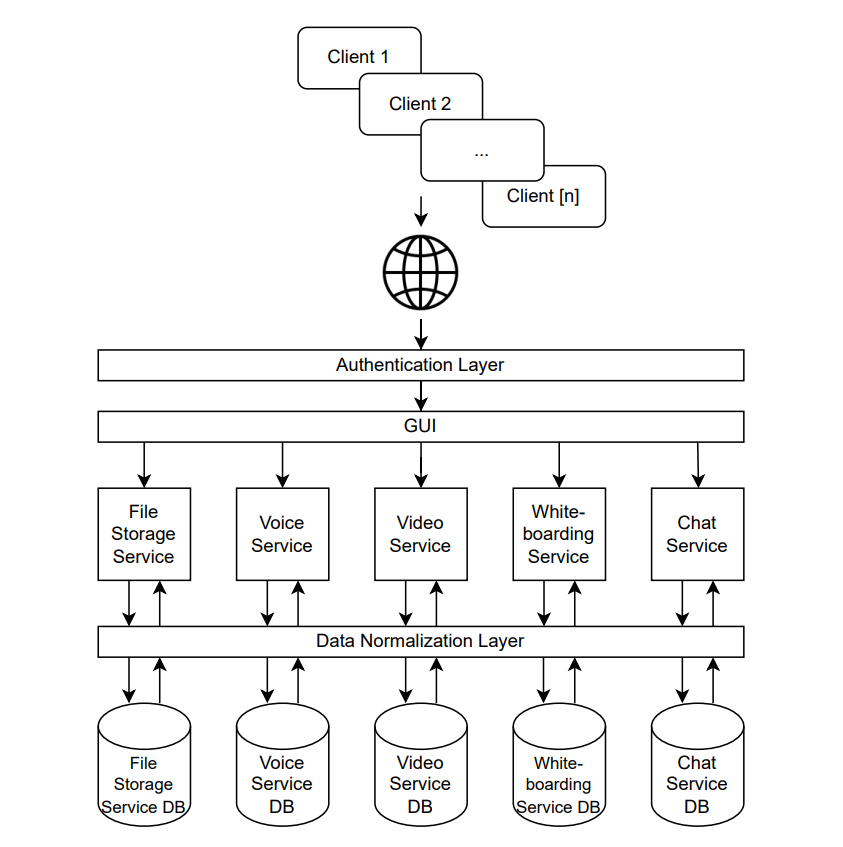
**Step 4: Choose One or More Design Concepts That Satisfy the Selected Drivers**

* UC-1 - The system shall support 5 different services including voice communication, video conferencing, instant chat, file sharing, and collaborative whiteboarding.

**Step 5: Instantiate Architectural Elements, Allocate Responsibilities, and Define Interfaces**

**Step 6: Sketch views and record design decisions**

**Step 7: Perform Analysis of Current Design and Review Iteration Goal and Achievement of Design Purpose**



---

How much storage do we need?

* Stream video
* Store recorded sessions
* Store chat
* Store files

How do the data storage communicate

Who is going to support it?

* Utilize AWS for existing architecture

Front end: web page to access the services

Session management service

Payment authorization layer