Last Updated: 11/10/2020

MET CS682 TERM PROJECT PART 1

REQUIREMENTS

Sneha

The purpose of this exercise is to get practice of specifying requirements. In practice, requirements are uncovered in an iterative process. This is your start. Go as far as your imagination allows, remaining aware of the fact that there is a long way to go in specifying an entire system.

Please leave the headings and the gray text unchanged except for the hints section which should not be included in your solution. Observe the page limitations; however, you may include as many appendices as you wish. All appendices should be referred to in the main text. Include your last name in the file name of the assignment. (i.e. SmithM\_CS682ProjectPart1.docx)

Some remarkable corporate work changes are taking place. For example, “[Why REI Is Selling Its Brand-New Headquarters: The retail giant is planning a remote future for main office employees](https://www.outsideonline.com/2416265/rei-selling-bellevue-headquarters-remote-work).” You are to specify a system called *RemoteAssist* that assists employees in their remote work.

* You are setting the requirements for *RemoteAssist*. Consider a specific scenario of your choice (i.e. a scientific lab, doctor’s office, fitness studio, or any other scenario which had to switch to remote work).
* Form your own vision and scope, focusing on features which assist employees working remotely. Your eight sections should be consistent as a single solution.
* *RemoteAssist* is an enterprise application—accessible to all employees, though possibly at differing levels of access.
* Your solution should focus on software-intensive aspects.
* Like all sensible IT systems, *RemoteAssist* will not reinvent the wheel, for example we are not recreating a video-conferencing platform like Zoom however your system can integrate with one. We all know what Zoom meetings look like; instead, think anew about what’s really wanted from remote work assuming that perhaps the entire company, with thousands of employees, works remotely all the time.

The last section on this template lists numerous hints.

# Assumptions/Scope (optional)

In no more than a few sentences outline your scope for *RemoteAssist*. Provide a scenario and what your project will focus on. A bulleted list is fine.

The scope of the project is to help FBI agents connect to the forensic team in real-time to receive instant assistance to advance a crime scene investigation. The system will be for enterprise users (100 to 500 users).

1. **Timeline**: The project should be achievable within 5 months with 8 developers and 3 test engineers. The project will utilize agile methodologies to complete the project in the given time.
2. **Cost**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Project Manager cost | Project team cost | Hardware cost | Software cost | Total Estimate |
| $50,000 | $220,000 | $20,000 | $40,000 | $330,000 |

1. **Benefits**: The business can venture into new opportunities or new industries. The company may increase its financial resources.

# Overview/Mission Statement

One paragraph, about four sentences.

*RemoteAssist* is an intelligent application for users working in a federal law enforcement agency. The applicationshall allow FBI agents to connect with the forensics team to receive remote assistance in progressing a crime scene via preliminary investigation. The agents can place a call and utilize the camera and video options when the recipient answers the call. *RemoteAssist* integrates with advanced image processing systems to collect and share the digital forensic evidence virtually to obtain analysis in real-time.

# User Stories

Provide two (2) user stories from two different types of actors (system users) in the format provided below. The two user stories should come from two different actor’s point of view. Replace all of the “<…>” parts.

## First User Story

As an investigator, I want *RemoteAssist* to allow me to contact user(s) in other departments to share what I see at a crime scene so that I can accelerate the investigation process.

*OR*

As an investigator, I want *RemoteAssist* to:

* allow me to contact user(s) in other departments.
* have a camera with special functionalities to capture the forensic evidence at crime scenes.
* allow me to write on image(s) or video while on call and share the image/video with the recipient.

so that I can accelerate the investigation process which will eventually lead to catching the suspect soon.

## Second User Story

As a forensic expert, I want *RemoteAssist* to allow me to remotely assist an investigator during an on-going investigation so I can pull crucial evidence into the view and recreate the crime scene to deduce what may or may not have happened.

*OR*

As a forensic expert, I want *RemoteAssist* to:

* allow me to receive audio/ video calls from the investigation unit.
* analyze evidence by performing DNA tests.

so that the investigation can move forward identifying a suspect or a victim through the evidence.

# Functional Requirements

In about three quarters of single-space page (using 12-point type), specify key functional requirements to various types of users. These requirements define the scope you have chosen. On average this works out to be about 12-16 requirements.

1. *RemoteAssist* shall allow users to login with their departmental credentials.
2. *RemoteAssist* shall allow the Investigator to contact other user(s) in real-time by chat, audio, or video.
3. *RemoteAssist* shall display a message “Call disconnected,” if the recipient does not answer the call.
4. *RemoteAssist* shall have a camera that will incorporate most basic settings of modern cameras.

* Panorama mode
* 3600 view
* Live Focus and Selective focus
* Slow Motion
* Hyper-lapse
* Self-timer and more

1. *RemoteAssist* shall have a camera feature with advanced image processing methods. For example:

* Capture and restore fingerprint patterns.
* Provide 3D images of shoe/ footprints.
* Describe hair strands accurately by indicating the color of the strand.
* Provide filters to set out unique wavelengths which projects substances from their environments. (The substances can be blood, water, alcohol, chemicals, etc.)

1. *RemoteAssist* shall allow both the investigator and the forensics expert toadd or remove annotations on an image or a video during a call with a recipient. For example:

* Circle out wounds on a victim’s body.
* Point out objects at a crime scene using pictograms like arrows.

1. *RemoteAssist* shall allow the forensics expert to zoom in and out of the video during the call.
2. *RemoteAssist* shall allow both the investigator and the forensics expert to record a call.
3. *RemoteAssist* shall allow both the investigator and the forensics expert to send messages, images, documents, or videos in the chat.
4. *RemoteAssist* shall allow both the investigator and the forensics expert to download and screenshot images or videos.
5. *RemoteAssist* shall save the downloaded images and videos and the call recordings in Files.
6. *RemoteAssist* shall allow both the investigator and the forensics expert to add additional people (up to 5 extra) to the on-going call.
7. *RemoteAssist* shall have a News feature which will display departmental news and announcements.
8. *RemoteAssist* shall have Settings as a sub-menu which will allow the users to:

* change profile picture.
* set ringtone.
* set volume.
* start/ reset timer.

1. *RemoteAssist* shall have a feature called Journal which will serve as an internaltracker. The investigators can use this feature to:

* log hours spent on a case.
* note down suspects.
* make a note of visitors (victim’s relatives, friends, neighbors, etc.) at a crime scene.
* register witnesses’ details (name, contact info, etc.) and their testimony.

1. *RemoteAssist* will have an Agenda feature which can be used to schedule and log appointments.
2. *RemoteAssist* shall have a feature called Emergency. The investigator can click on this feature to alert the police department to inform of any danger.
3. *RemoteAssist* shall not log out unless prompted by the users.

# Use Cases

Specify two detailed use cases in the tables below which you consider important and complex to the functionality of *RemoteAssist*, showing actors, preconditions, actor actions, and system responses. Each of these use cases should have approximately 5 to 6 steps, with each step potentially having an actor and/or system component.

## First Use Case

|  |  |  |
| --- | --- | --- |
| **Use case Name** | Investigator (Caller) Use Case | |
| **Actor:** | Investigator | |
| **Description:** | *RemoteAssist* shall allow the investigator to contact a forensic expert to share images and videos from a crime scene to receive assistance to accelerate the investigation. | |
| **Pre-condition:** | The investigator logs into the application on a smart display with departmental credentials. *RemoteAssist* shall open the home screen displaying the menu. | |
| **Step #** | **Actor** | **System** |
| **1** | The investigator clicks on Contacts and starts typing the contact’s name in the search bar. | *RemoteAssist* displays a contact list and starts searching a contact with the name being entered. The result is then returned to the user. |
| **2** | The investigator clicks on the Call icon. | *RemoteAssist* calls the selected contact. Displays a call screen when the recipient answers the call |
| **3** | The investigator turns on the camera and captures fingerprints at a crime scene. | *RemoteAssist* processes the fingerprints, restores any lost patterns, and displays the processed image to both the caller and the recipient. |
| **4** | The investigator selects video mode and adds annotations on the screen. | *RemoteAssist* switches to video mode and shows annotations being added to both the caller and the recipient. |
| **5** | The investigator records the call. | *RemoteAssist* starts recording |
| **6** | The investigator sends a message in the chat and uploads images. | *RemoteAssist* sends the message and images to the recipient and displays the same in the chat. |
| **7** | The investigator ends the call. | *RemoteAssist* saves the recording in Folders, disconnects from the recipient, and displays the Contacts screen. |
| **8** | The investigator goes to the home screen and selects Journals from the menu. | *RemoteAssist* shows all the entries made by the investigator. |
| **Alternate Courses:** | #2 The investigator’s call is not answered— display a message “Call Disconnected.” | |
| **Implementation Constraints:** | 1. *RemoteAssist* shall have a high-resolution camera of 50 megapixels or higher. 2. *RemoteAssist* shall not take more than 10 seconds to process the fingerprint patterns. | |

## Second Use Case

|  |  |  |
| --- | --- | --- |
| **Use case Name** | Forensic Expert (Recipient) Use Case | |
| **Actor:** | Forensic Expert | |
| **Description:** | *RemoteAssist* shall allow forensic expert(s) to answer calls to solve problems in real-time, pull crucial evidence into view, and support collective problem-solving. | |
| **Pre-condition:** | The forensic expert will always be logged in to the application on a smart display device. The device (system) will display the menu on the home screen. | |
| **Step #** | **Actor** | **System** |
| **1** | The forensic expert will answer a call. | *RemoteAssist* displays the call screen or the camera screen with image/ video depending on the caller’s action. |
| **2** | The forensic expert zooms on to an object in the video during the call. | *RemoteAssist* directs the focus on the selected object. |
| **3** | The forensic expert zooms out of the video and highlights a few areas in the scene. | *RemoteAssist* displays the full video with highlights. |
| **4** | The forensic expert takes a screenshot. | *RemoteAssist* captures a still image from the video, along with the highlights, and saves it to Folders. |
| **5** | The forensic expert clicks on a DNA test icon on the camera screen and selects the ‘Analyze Fingerprint’ option. | *RemoteAssist* navigates from the video screen to DNA test screen and starts analyzing fingerprints. |
| **6** | The forensic expert waits. | *RemoteAssist* returns the result to the forensic expert. |
| **7** | The forensic expert downloads the results. | *RemoteAssist* downloads the results and saves it to Folders. |
| **8** | The forensic expert decodes the results and messages the details in the chat. | *RemoteAssist* sends the message and displays the sent message. |
| **Alternate Courses:** | #6 If the DNA test does not return any data—display “Match not found” | |
| **Implementation Constraints:** | The highlights on the video will remain in place even if the investigator changes the focus on the camera. | |

Note: DNA test feature will only be available to the forensics team and not the investigators.

# State Transition Diagram

Develop a state transition diagram for a part or whole of *RemoteAssist*. Limit this to 5-7 states if possible. Explain what aspect of *RemoteAssist* the diagram applies to. Note that sub-states of one of these states will be added in question 6. You may use Visio, LucidChart, or another design tool of your choice (but please check with your facilitator in advance if not Visio or LucidChart).

\*\*Attached as a PDF (*Figure 3.1*)

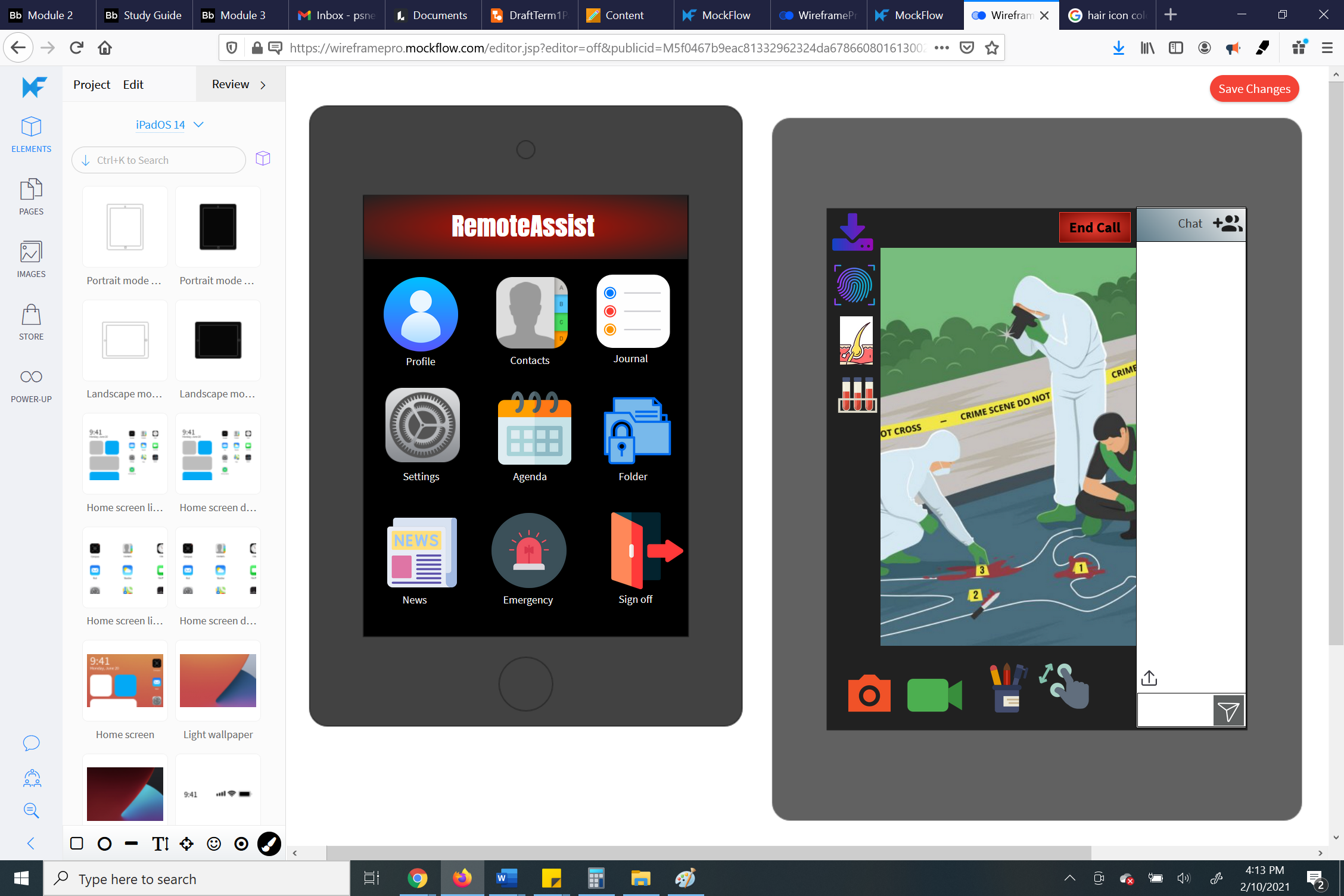
# Sub-States

Expand one of the states in the previous question into sub-states consistent with at least one of your detailed use cases. Limit this expansion to 2-4 sub-states. Show the transitions that affect these sub-states. You may combine parts 5 and 6 together into a single diagram for part 5 as long as it is clear.

\*\*Attached as a PDF (*Figure 3.2*)

# GUI Sketch

Create a GUI sketch of one important/complex screen for *RemoteAssist*. You can use LucidChart, Visio, <https://www.mockflow.com/>, or another design tool of your choice (please check with your facilitator in advance unless you plan to simply insert screenshots).



*RemoteAssist* will display the menu screen when it is launched, after signing-in.

\*\*The GUI mock-up for the main functionality is attached as a PDF.

# Non-Functional Requirements

Specify what you consider the two most important non-functional requirements. Describe your choices briefly and explain why you consider these most important.

1. *RemoteAssist* shall not take more than 5 seconds to capture an image.
2. *RemoteAssist* shall have a high-resolution camera of 50 megapixels or higher.
3. End-to-end conversations shall be secured by enforcing Transport Layer Security (TLS) encryption to prevent snooping and tampering of evidence. The content in the application shall be secured through Advanced Encryption Standard (AES) encryption.

# Appendix

This section is optional, place any notes here on how you have used the references or any other notes you would like to make.

# References

Show that you used a wide variety of resources by listing them below and clearly indicating in the body above where you used. Make sure to use proper referencing in your paper. We suggest using APA format, but other formats are fine as long as it clearly distinguishes your work from work of others in your response—be mindful of plagiarism rules.

[1] Wen, Che-Yen, and Chiu-Chung Yu. “Fingerprint pattern restoration by digital image processing techniques.” *Journal of forensic sciences* vol. 48,5 (2003): 973-84.

[2] SWGIT. 1 Oct. 2015, www.crime-scene-investigator.net/swgit-section5.pdf.

Icons and images for GUI are taken from the following websites:

<https://www.flaticon.com/>

<https://www.freepik.com/popular-icons>

<https://freeicons.io/>

<https://www.trfm.com.au/articles/morwell-forensic-hub/>

[https://mockflow.com/](https://mockflow.com/app/#Wireframe)

# Evaluation



**Please do not include Hints section in your solution.**

# Hints

## Overall Assignment Notes

* As usual, the notes are a primary source for explanations and examples; we also encourage you to do outside reading and research to gain additional perspective. In the real world, you would be interviewing people to carry this out but that is not possible in a class setting.
* If you run out of space (after careful editing), you may provide an additional explanation in the appendices section to support your design decisions or to clarify your choices.
* As in the real world, the quality of your work is subjected to judgments, not mathematical exactness. Help your facilitator make favorable judgments by being as clear and specific as possible. “Clarity” is one of the criteria.
* Feel free to include notes for your facilitator (that are not actually part of your paper). A good way to do this is to insert, for example, “Note 3 to facilitator” and list the notes in the appendix.

## Notes to Selected Grading Criteria

**Clarity**: Part of requirements analysis is to understand what your customer wants and needs (and, as we discussed, this is not easy)—so be sure that you understand the problem. Your response will be clearer as a result. Are your requirement statements clear enough so that someone could create a design from them? Are your names (e.g., of states) expressive enough so the reader can understand them? Explain throughout. **Consistency (an aspect of clarity)**: Make sure that your entire solution is consistent. For example, did you introduce some functionality in the use case that you didn't mention in your functional requirement? Then go back to part 3 and make sure to revise it.

**Relevance**: Review the scenario and make sure to stick to it. If you need to expand on them a bit that’s fine; provide a brief introduction with your assumptions, but make sure to stay within the initial framework of the scenario. For example - it's best to describe your approach and/or scope for the diagrams.

## Mission Statement/Overview

**Technical Soundness:**

* A high level overview. Logically, one would write this first, and then the other parts would flow from it. However, our minds do not necessarily operate very logically. To stimulate your thinking about the mission, you may first want to think about a user story, or even just an actor.
* Avoid details—they come later.

**Clarity**

* Do not underestimate the time required to write a clear overview that is both short enough to be readable, and yet long enough to convey what the system does, structured appropriately.
* Come back to this section at the end, to make sure it is consistent with the rest of your solution.

## User Stories

* First identify a user of your system. For example, an executive who wants to schedule a highly secure meeting consisting of a steering groups and a few invitees permitted to participate for set time intervals.
* User stories are short, simple descriptions of a feature, told from the perspective of the person who desires the new capability, usually a user or customer of the system.
* This should not be overly detailed; the key point is to consider how a specific actor interacts with the system. We provide a template for you in the assignment.
* Additional information on User Stories is available on page 112 of the textbook.

## Functional Requirements

**Clarity**

* You do not have to go into the finest details of the requirements, but make sure that you describe the major functionality. Enumerate and describe your functions systematically.
* Look to organize requirements with headers and sub-headers, for example by actor and or functional area.
* If a functional requirement sounds a little bit non-functional, explain your reasoning as to why you decided to keep it functional in a note (as sometimes these are borderline).

**Technical Soundness:**

* Make sure to understand the difference between functional & non-functional requirements.
  + Functional: Reasons for the system and what it’s meant to do.
  + Non-functional: Needed or wanted but not the reasons for the system.

**Toughness and Coverage:**

* You may need to come back to this after you complete the assignment to see what you missed, for example, are there key features you outlined in the GUI, state transition or use case which were not covered here?

**Relevance:**

* You may want to do some outside research to see relevant examples of how functional requirements are defined for systems. You can include your findings in the Appendices section

## Use Cases

**Technical Soundness:**

* Complete the use case template we give you in the assignment.
* Additional information is available in the tabular-narrative forms of Figure 4-13 on page 143 of the textbook and in the “Use Cases” section of the Module 3 notes.
* The use case needs to show appropriate sequence (actor/system)
* Clear understanding in difference between constraints and pre-conditions. The use case itself should follow one path as best as possible.
* Avoid branching in use cases, if possible—use only if necessary.

**Toughness and Coverage:**

* Complete the use case template we give you in the assignment. This is where we ask you to describe the use case, who uses it, what might be the pre-conditions, and alternate steps.
* Research can be applied by looking at similar systems (i.e. here is what I found and how it relates to my design). This makes your solution real and is something that analysts need to do to understand what technology will be used to implement this.

**Clarity**

* Consider consistency and relevance to the scenario.

## State Transition Diagrams

**Technical Soundness:**

* A good place to start is to review the “State Machine and State Transition Diagrams,” “Components of State Transition Diagram,” and “State Transition Diagram—An example” sections in this week’s lecture notes.
* Consider your use case as a way to start thinking about state transition, ‘what the system does’ are the states of your system, and ‘what the actor does’ could be the events that trigger the transitions, and then look to functional requirements and user stories to add detail.
* Avoid focusing too much on use cases themselves, you want to capture the overall main steps of the system.
* Make sure all transitions are labeled with events. If a state transitions into a state (i.e. search completed) but is not significant to be it’s own state, use it as an event within guard conditions.
* Please see pages 221-227 in the text on Behavioral State Machines – however, please note that this requires Object Oriented approach which we will look at next week. For now, review the mechanics of the state machine as it applies to this assignment.

**Toughness and Coverage:**

* Make sure all key functionality is covered (use cases, user stories, what was discussed in the system overview)

**Clarity**

* Make sure to show what the composite state(s) (which contain sub-states) are.
* Are diagrams clear to read (i.e. no overlapping lines, no non-polished designs).
* Diagram should be consistent with requirements (i.e. functional, use cases)

## GUI Sketch

**Relevance:**

* Decide what is the most important, potentially complex screen, consider what it should contain and draw a rough mockup sketch.

**Technical Soundness:**

* See the “GUI Mockup Example” section in this week’s module for examples.
* Chapter 10 in the text covers material on human-computer interaction layer design.
* GUI is fairly straight forward; many students have fun with this. As mentioned previously, Visio and Lucidchart have a wireframe template, but you can also check out balsamiq.com, wireframepro.com or mockflow.com

**Clarity**

* Provide “Sticky Notes” to describe functionality which may not be obvious.

**Non-Functional Requirements**

**Technical Soundness:**

* Make sure to understand the difference between functional & non-functional requirements.
* This is the "How" the system is implemented (i.e. quality requirements, constraints). Use references to support your choices. Think about what is most important and why; use of references should help here.
* The “Non-Functional Requirements” section of this week’s notes provide examples.
* Please see secondary readings for Week 3 in the textbook for additional examples of non-functional requirements.

**Toughness and Coverage:**

* You may want to do some outside research to see relevant examples of how non-functional requirements are defined for systems. You can include your findings in the Appendices section.

**Relevance:**

* Review your entire solution after completing it—you will uncover additional considerations. Check for consistency.