#securemanufactureui

CSC 591, Spring 2020

Team:

#	Name	Unity Id
1	Tushar Himmat Dahibhate	tdahibh
2	Yang-Kai Chou	ychou3
3	Rajshree Jain	rjain27
4	Satanik Ray	sray7
5	Ankit Arvind Tiwari	atiwari4

Clients:

Dr. Hal Aldridge, Secmation
Dr. Binil Starly, CAMAL

Long Term Goals:

Secmation is a company that provides engineering, technology, and tools to add information security in new and existing products. The main focus of our project is to design a User Interface for monitoring tasks, providing updates and capturing the security aspects related to the manufacturing process in a user-friendly manner. As a part of this project, we also collaborate with the Center for Additive Manufacturing and Logistics(CAMAL) from the Industrial and Systems Engineering department at NCSU. The objective of the project is to make security, scheduling jobs etc as user-friendly as possible and helping the operators to use the machines without any struggle.

Challenges:

Consistent Design
We will need to provide a consistent UI to avoid breaking the User experience. The UI will be serving different devices and machines connected to the SAMM. We will need to make sure that there are no inconsistent elements in the UI for different devices. The operators might find themselves in an unfamiliar territory due to inconsistent designs.

Handling Access control and escalations

The SAMM operator will be operating on different security levels. A security level incorporates different actions that an operator is permitted to perform. For example, a machine operator is not permitted to change any of the machine settings. The UI must not expose any options which allow the machine operator to do so. On the other hand, a floor manager must have access to such options as he will have a security clearance to make any desired changes to the configurations of the machine.

Handling errors and alerts

Since we are incorporating security aspects related to the machines, we also need to handle the various alerts and errors which would require operator intervention. We need to design the UI in such a way that the operator can handle the errors without having to consult the instruction manuals or seeking help from the IT department. The errors and the error handling routines must be simple enough for the operator to understand.

Maintaining a paper trail

Since there might be some jobs operating on a higher level of security policies, the UI will have to make sure that only authorized personnel can carry out the tasks related to that particular job. This includes authenticating operators, maintaining a record of all the activities carried out by the operators and, maintaining a checklist of all the prerequisites carried out for that particular job. The UI must also be able to maintain a record of all the completed tasks, and the details of all the machine supplies currently present.

Maintenance activities

The machines often need to be serviced and calibrated periodically. These tasks are performed by the vendors of the machine or other third-party companies. The UI must alert the operator whenever a particular machine is due to be serviced. Some records about the personnel responsible for the maintenance and calibration activities must also be stored for accountability. The UI must also be able to display the current supplies present in the machines. This prior information will help the operator to perform the tasks efficiently.

Avoiding alert fatigue

Displaying many alerts can numb and desensitize the operators. The operators might choose to bypass all the security measures entirely. This can lead to vulnerabilities in case of jobs with high-level security policies. We must present the alerts and errors in such a clear and readable manner by abstracting all the security-related jargon. We must also offer an actionable checklist to handle all the alerts.

Incorporating user fatigue

The User experience will need to be designed in such a way that it does not exacerbate user fatigue during late-night shifts. One of the goals of the project is to maximize the efficiency of the operator. As a result, we will need to take into account the user fatigue due to late-night shifts and design a UI which is simple and intuitive but not too monotonous.

Easy onboarding

The UI needs to be designed from the operator's perspective as they will be the end-users. This will require incorporating the suggestions given by the operators on how they want the whole experience to be. This will further lead to a smooth onboarding process. A complex user interface will take a longer time for the onboarding of the operators.

Designing easy and intuitive workflows

The workflows must be intuitive and easy for the operators to understand. Haphazardly placing the UI components will cause a lot of confusion and may lead to decision fatigue.

Providing feedback and building trust

The UI must be able to provide suitable feedback to the operator whenever a particular task is complete. Acknowledgments and status messages will help the operator in understanding what is happening and will build a certain level of trust in the application.

Experience Map:

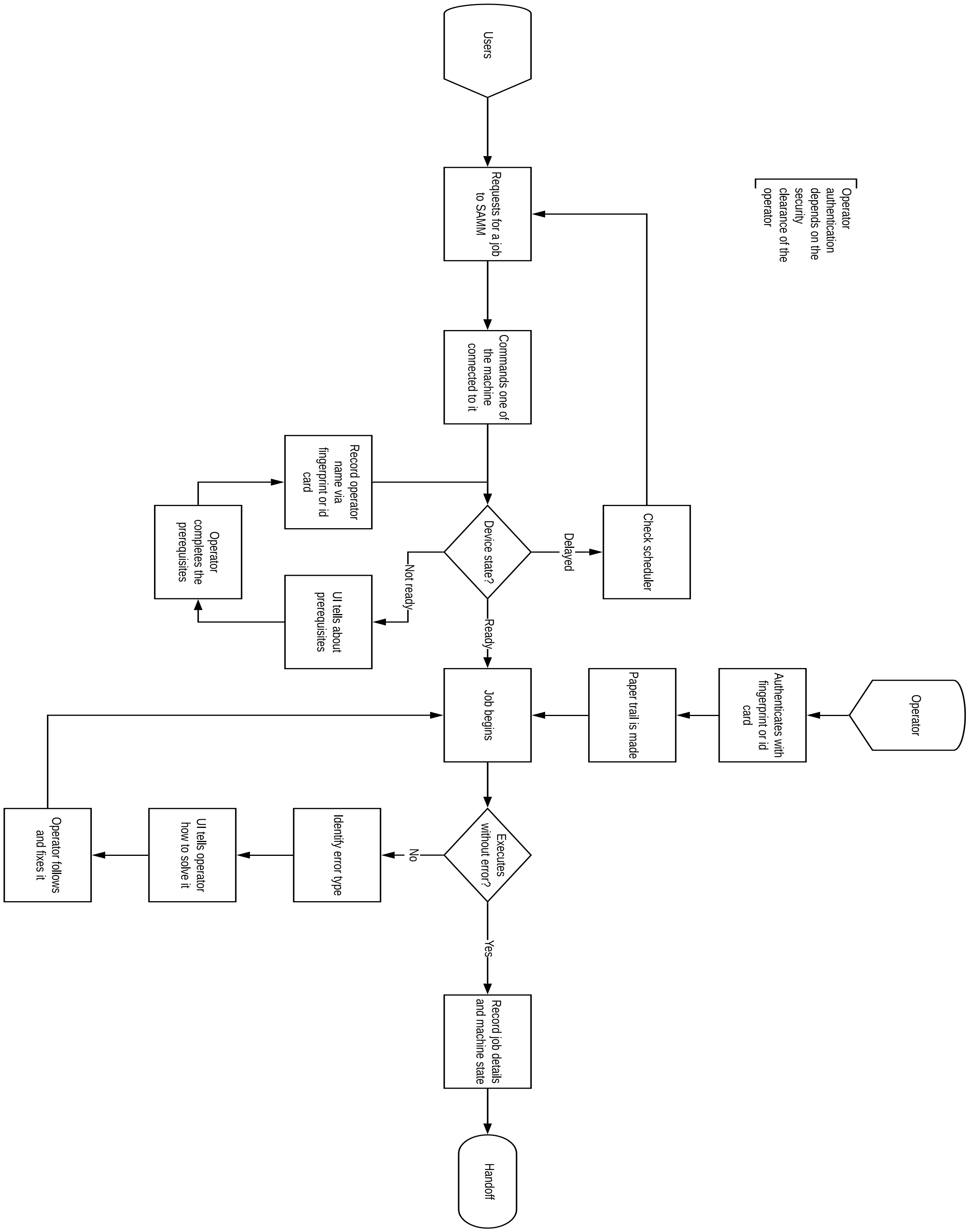

Users
Requests for a job to SAMM
Commands one of the machine connected to it
Device state?
Delayed
Check scheduler
Not ready
UI tells about prerequisites
Operator completes the prerequisites
Record operator name via fingerprint or id card
Ready
Job begins
Operator
Authenticates with fingerprint or id card
Paper trail is made
Operator authentication depends on the security clearance of the operator
Executes without error?
No
Identify error type
UI tells operator how to solve it
Operator follows and fixes it
Yes
Record job details and machine state
Handoff

Fig 1: Experience Map

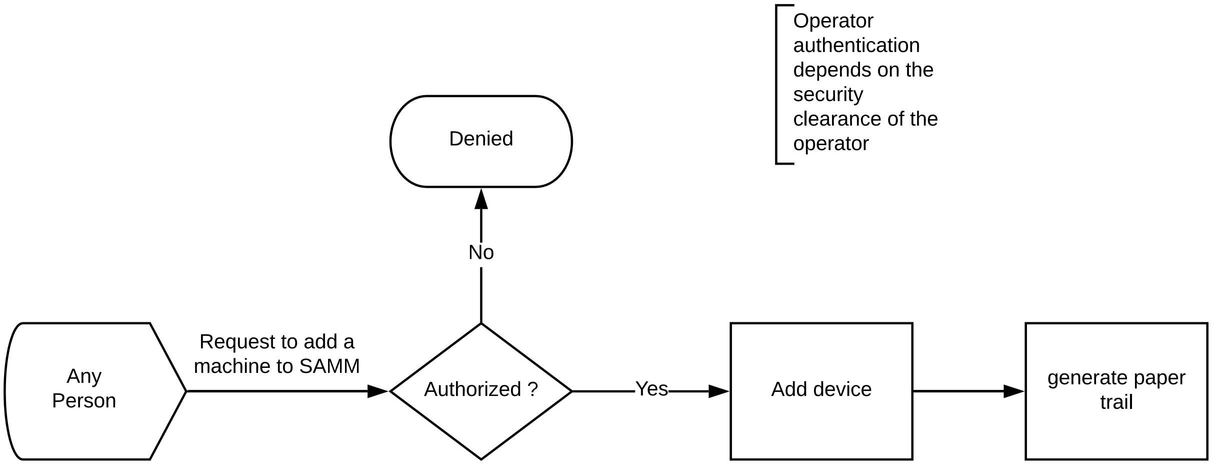

Operator authentication depends on the security clearance of the operator
Denied
No
Any Person
Request to add a machine to SAMM
Authorized ?
Yes
Add device
generate paper trail

Fig 2: Adding a machine to the SAMM

Expert Notes:

We had a meeting with Hal Aldridge and he elucidated us on the project that Secmation is making, in collaboration with the Department of Industrial and Systems Engineering(ISE). Secmation has created SAMM which will be a controller for a subset of total machines in a factory. SAMM controls what jobs these machines do and the operators (in the proximity of the SAMM) are responsible for checking if a machine is ready for taking that job or to troubleshoot (if some error arises).

He told us that since a user can connect to such SAMMs via a direct network or via a cloud, it is imperative to provide operators and the SAMMs with tools to prevent security leaks and attacks to the machines. We went over various use cases and we understood that operators of those machines are not equipped to deal with such security attacks. It will take a lot of re-education and training to make them qualified to deal with such a system, which at this point, raises questions about the requirement of SAMM at all. So, we are to make a UI that maps technical jargon and terminologies of system security to the terminologies which the operators are used to and at the same time, hiding details that are not relevant to their tasks. This can help Secmation to maximize the utility of their product- SAMM.

We also met with Dr. Binil Starly of the ISE department. He went through the procedure which operators follow for working on machines. We learned that for plastic printing, they use GrabCAD as an interface and for metal printing, they use a dedicated interface on the machine. We understood that most of the process currently is verified by humans and there are little automated security checks that are in place. He pointed out the scale of the operation under him and it seems that till the machines are not connected to customers via the cloud, and hence these checks are, for now, a safe way.

Goal: Make a UI for SAMM which aids operators of a factory to handle security attacks without a need for a long training period. It should help operators to handle errors, schedule jobs, schedule updates and aid them to troubleshoot through errors and security attacks. It should also create a paper trail that will be proof of the quality and security of the product and the machine.

Challenges:

- Create a UI which operators would actually like.
- The operators training period should be short.
- To incorporate the effect of time while using the UI. For example, activities during a night time shift require extra "spoon-feeding" for the drowsy people.
- Incorporate "Industry 4.0" standards into present-day companies and factories.
- Abstract security-related jargon from the UI and to present them as a black box.
- Make security part of an operator's daily routine seamlessly.

Previous attempts:

Dr. Hal Aldridge told us that no literature survey or attempts have been taken by the company for this project, he told us to look up bad UI or bad security practices that operators have to deal with. One of the prominent articles that came up explained how Hawaii got a missile alert on all their smartphones which were a UX fault. Another article[1] explained how people are more motivated to follow security when the threat is in terms of what they see in their everyday life. It gave an example of an error message on old Windows. The error message was, "This program has performed an illegal operation and will be shut down." The word "illegal" triggered non-technical people more strongly than telling them what was actually wrong.

Dr. Binil Starly told us about "3yourmind", which is a company that, along with other things, is also into Agile Manufacturing Execution System or MES. This software is similar to what we are trying to make but it is not clear if it also handles the security aspect.

Accuracy of maps and suggestions from experts:

Dr. Binil Starly suggested that since most of the security checks are now being handled by humans, it will be better if the UI can help operators in doing it correctly. For example, machines have a maintenance period and a cooldown period. UI can help them in ensuring these processes are executed timely. This will help in ensuring that the product made is of the highest quality. Also, not all operators are the same. Most of them have different security clearance and freedom while using the machine. The maps and the UI should perfectly represent that.

Elijah Morgan from Secmation suggested that the map of when a new device is added is too simplistic.

For example, the following process will provide a better picture:

1. Operator authenticates
2. Receives instructions from UI
3. Performs actions
4. The system validates (i.e. connect test, system self-test)
5. If failed send instructions to the operator
6. If succeed information is sent to IT (so they have a record of the new device coming online)

In our current map, the user sends a request to the SAMM directly and he points that may not be the case always. The SAMM needs to provide a job by a scheduler and hence the request may go to a scheduler first and then the scheduler chooses which job to go to which SAMM. Also, the operator should be given a hint by the UI about the device he is controlling with the SAMM.

He also felt that our usage of the word "user" is vague and we should restrict it to customers or clients.

Problems/Opportunities:

Easy learning experience and User Interface:

- How will we make the User Interface friendly enough for the Operators to understand the security jargon easily?
- What measures should we take to make the User Interface easy to use, understand and learn and even handle tasks irrespective of the time of the day?
- What kind of User Interface should we make for the inexperienced Operators to learn the tasks and machine usage as quickly as possible?
- SAMM will be connected to different machines. How to ensure a consistent interface across such machines?
- As there might be different machines and different jobs associated with those machines, how do we ensure that the operator knows which machine he is operating on?
- How do we provide the acknowledgments and status messages to the operator whenever a particular machine is done with its job?

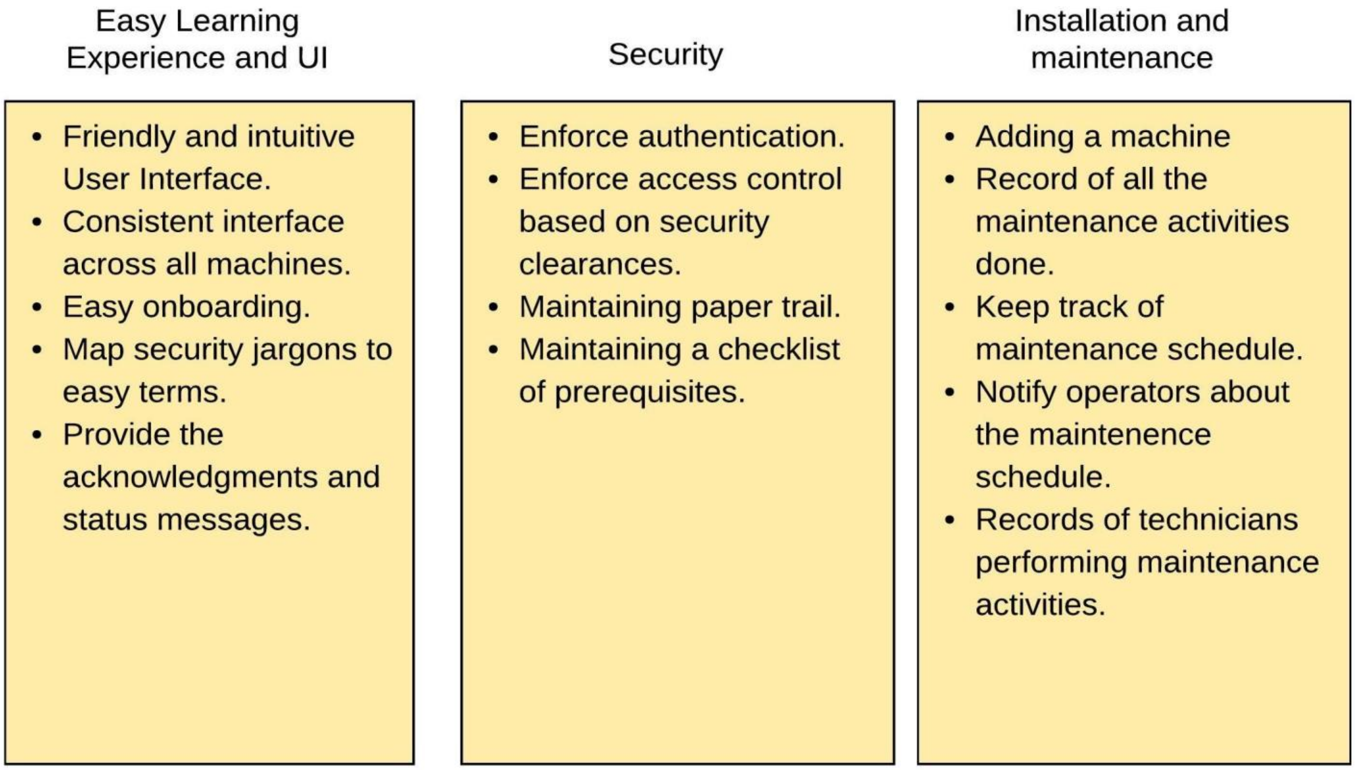
Security related:

- How will we make the User Interface to respond to different levels of Operators in order to ensure access control?
- How will we authenticate the Operators for tasks of different security levels?
- How will we make the understanding of security concerns easy for the Operators?

- How should we plan to enhance the ease of the maintenance tasks and regular security checks without delay in product delivery?
- Different operators will have access to the SAMM. How do we maintain a paper trail to ensure accountability?
- How do we ensure that all the prerequisites are followed diligently and not bypassed?

Installation and maintenance

- How do we add a machine to the SAMM?
- How do we keep track of all the maintenance activities done?
- How to keep a track of when the machines need to be calibrated and how to alert the operator regarding the same?
- How to keep a track of external entities interacting with the machines for the installation and maintenance activities?



Easy Learning Experience and UI

- Friendly and intuitive User Interface.
- Consistent interface across all machines.
- Easy onboarding.
- Map security jargons to easy terms.
- Provide the acknowledgments and status messages.

Security

- Enforce authentication.
- Enforce access control based on security clearances.
- Maintaining paper trail.
- Maintaining a checklist of prerequisites.

Installation and maintenance

- Adding a machine
- Record of all the maintenance activities done.
- Keep track of maintenance schedule.
- Notify operators about the maintenence schedule.
- Records of technicians performing maintenance activities.

Target:

Our team will try to focus on user interface design. Matching the mechanism to the terms which the operators are familiar with, will improve the user experience. We also need to deal with the error exceptions and make it easy to understand for the operators.

Reference:

[1] https://hackernoon.com/cyber-security-requires-an-important-ingredient-strong-ux-d0727a0c076

[2] https://www.secmation.com/about-us