CIS641F2017 **Peer Project Review Worksheet** **Schymik**

"Our goal is to remove defects as close to their insertion point as possible"

PROJECT TEAM BEING REVIEWED: GROUP 9				DATE: 11/24/17
REVIEWER: KATE SKOCELAS				
Category for Review *(rate the work presented based on the criteria stated below)*	Needs Improvement	Criteria Achieved	Mastered	Comments *(if you rated any category less than "Mastered", you need to explain why and be specific with your explanations so that the authors can correct the deficiencies based on your comments)*
Overall Project Requirements				**Word isn't letting me add X's to the top three line items, so I have written the selected box at the top of the comment for each**
• Did the models adequately define the scope of the project? Listing what is and what is not included so that no confusion applies when reviewing the models?				[Mastered] The scope of the project is articulated well and consistently throughout the paper.
• Did the models, in concert with the JBGE analysis, accurately represent the problem space?				[Criteria Achieved] The scope of the models is consistent and most of the problem space is represented. However, several features of the problem space are omitted, including mandatory checkout sessions for pilots flying a model for the first time and instructor endorsements when required for aircraft models with heightened restrictions. Also, payment is included in the problem space and models (exp: use-case description for "Calculate Total Time"), but the process for handling it is not explained.
• Did the document overall properly present the model to the client and to developers?				[Mastered] The document is formatted, indexed, and linked well. The written portions are articulate and helpful in explaining how the system works and what the models mean. It is put together in a professional manner.
JBGE Analysis				
• JBGE Process Did the team demonstrate an understanding of the JBGE concept and process? Did they apply the agile values and principles properly?			**X**	The JBGE rationale was clearly thought through and explained well.
• Quality of the JBGE analysis. Is the JBGE analysis sufficient/appropriate? (In other words: did their JBGE analysis result in information important to developers or other stakeholders being lost?)		**X**		More detailed information on how the use cases function is necessary for developers to be able to understand this system. How a user navigates through the system is explained well, but after reading the report, it was still unclear to me how the use cases triggered by the user actually operate. I am a beginner, however, so this may not be the case for more experienced developers.

Functional Models				
• Use Case Diagram drawn properly? Is the diagram "syntax" correct? - are proper symbols used to define actors, use cases and the relationships between and among the two? Are the labels adequate and proper? (use cases should be verb phrases so that combining actor and use case, a logical noun-verb phrase can be stated eg: Customer selects product to purchase)		X		Overall, the use case diagram looks good. Use cases like "Schedule" should be verb phrases so that it is clear what they do. Based on their descriptions, I think some of the includes/extends relationships may have been mixed up with one another and/or have arrows pointing the wrong direction. For example, "Create Reservation" extends the "Schedule" use case, and "Schedule" extends the "Reserve Aircraft" use case. Therefore, "Create Reservation" is an extension of "Reserve Aircraft," which does not match the way it is described in other models. It is likely that "Reserve Aircraft" was intended to extend "Schedule" instead of the other way around.
• Use Case Diagram complete? Is scope properly identified? Does the diagram include or cover all of the problem space?		X		The use case diagram is almost complete. It is missing a few functions described in the problem space, such as creating and maintaining a pilot's account. The issues of required checkout sessions and instructor endorsements are also not addressed. If those functions are outside the scope of the new system, there should be an explanation of why that decision was made.
• Use Case Descriptions properly formatted? (don't be overly detailed in your requirement here – did they name the use cases, provide for traceability, state initial and post conditions, and properly list normal, sub-, and exceptional flows appropriately?)		X		The overall format for the use case descriptions looks good, and they provide traceability. The individual steps should be written in SVDPI form when possible. For example, in the "Change Reservation" use case, step 3 is "Sends confirmation back to system." Who sends the confirmation? The relationships section should also include relationships with other use cases.
• Use Case Descriptions complete? Have they followed their JBGE analysis/plan? Did they cover all of the use cases with descriptions where needed?	X			The steps in the use case descriptions give a brief overview of the operation of the system, which is good for management. More detail is needed, however, for developers to be able to understand how each use case is performed by the new system. For example, in the Schedule use case, step 3 is "Pilot finds available time of instructor pilot and books." How is that completed? Does the system only display the available times, or does it show the entire calendar of every instructor to the pilot and let them hunt for an opening? How does the system "book" an instructor?
• Activity Diagrams properly drawn? Has proper syntax been followed? Perfect is not the goal but complete information must be properly communicated.			X	The syntax of the activity diagrams looks great. The only minor issue is that a few of the decision criteria are unclear (exp: "delete" on Manage Reservations) and/or missing their brackets.

• Activity Diagrams sufficient? Have they followed the JGBE analysis and completely covered what you see as all of the complexity requiring explanation with their set of Activity Diagrams?	X			The activity diagrams seem to represent a user interface instead of the steps involved in executing a use case. This can be valuable information, but I am still confused as to how the use cases function. That may not mean more activity diagrams, but the steps should be clearly spelled out somewhere. I had a difficult time understanding the activity diagrams. For example, on the "Manage Hobbs Time Entry" activity diagram, one guard condition is "Delete Record" – presumably meaning an actor selects "delete record" on the interface – which then causes the "Delete Info" activity to execute. Is the "Delete Info" activity the same as the "Delete Hobbs Record" use case on the use case diagram? On the use case diagram, only an employee can execute the "Delete Hobbs Record" use case, but the conditional loop above it on the activity diagram, "Manage Hobbs Time Entry," is only connected to Pilot. So who is selecting "delete record" (if that's what's happening for the guard condition)? Are there any limits on when a record can be deleted? This type of confusion happened in several places for me.
Structural Models				
• Do the CRC cards (if needed – JBGE – and presented) sufficiently identify and specify the classes? Do all CRC cards produced have a corresponding class on the class diagram?		X		The CRC cards all have a corresponding class on the class diagram. However, the responsibilities and attributes on the CRC cards don't match the operations and attributes on the class diagram.
• Balancing Are all classes, attributes, operations (methods), and relationships identified in the class diagram and CRC cards justified by the problem description and functional models?			X	The written text was helpful in explaining how the classes were structured and why.
• Class Diagram correctness Does the class diagram use proper "syntax" or notation? Are all symbols used in the diagram used correctly? And completely?			X	Almost perfect. A few minor details: Multiplicity symbols should be in a range when many, like one Aircraft can have 0…* Reservations associated with it. Also, since instructors and customers are both Users, and sometimes an instructor is needed to reserve a plane, should Reservation be able to have 1...2 Users associated with it?
• Object Diagram If used, is it used properly? Does it use proper notation? Does it convey useful information? Does it match (balance with) the class diagram?			X	
Behavioral Models				

• Communications Diagrams completeness Are there sufficient diagrams present to explain complexity of the system and properly communicate all of the necessary information to the developers? (per the JBGE analysis)	X			How the system functions from a user's perspective is explained well, but developers will need communication diagrams to understand how the system functions internally to build it.
• Communications Diagram correctness Do the diagrams use the proper notation? Do they balance with the class diagram and object diagram(s)? Do all messages have a corresponding relationship on the class diagram?	X			Communication diagrams depict the messages sent between actors, objects, and classes to complete a use case. The communication diagram given, however, shows the messages being sent to use cases instead of objects/classes. I cannot tell what process is being completed by the messages sent.
• Sequence Diagrams completeness Are there sufficient diagrams present to explain complexity of the system and properly communicate all of the necessary information to the developers? (per the JBGE analysis)		X		The sequence diagram was crafted well in Visio, but it is difficult for me to understand. For example, you say in the paragraph above it, "Remember, a pilot who is a student will need to schedule time with a pilot who has an instructor license." The sequence diagram, however, does not show any checks to the pilot object to get its license type. It also doesn't check instructor availability, which would be necessary for those students to know when they can reserve a plane and instructor. I do not understand what "showAircraft" means. I assumed that "ShowReservations" meant display them to the user, but "showAircraft" is only passed to the Reservation class. Since it is never passed to the pilot, does this mean that "ShowReservations" is also displaying plane availability and allowing the user to select a model? For me, more information is needed to understand what is happening in this process.
• Sequence Diagram correctness Do the diagrams use the proper notation? Do they balance with the class diagram and object diagram(s)? Do all messages have a corresponding relationship on the class diagram?		X		The sequence diagram notation is mainly correct. It's unclear whether the first column represents a pilot actor or pilot object. From the way the process is described in the sections and diagrams above, it seems like one of each is needed as part of the process.
• State transition diagrams completeness Are there sufficient diagrams present to explain complexity of the system and properly communicate all of the necessary information to the developers? (per the JBGE analysis)	X			I was not able to understand the state diagram given. The reservation class is complex enough that a state diagram representing it is necessary for understanding how it functions.
• State Transition Diagram correctness Do the diagrams use the proper notation? Do they properly and completely explain the management of the state of the class to which they correspond? Do they balance with the class diagram and object diagram(s)?	X			I cannot tell which class the diagram is depicting. The events seem to correspond with the Reservation class, but the states do not. For example, what would the "Manage Reservation" state be? It is also missing the initial and final state circles, so I don't know how to follow the flow of the diagram.

• CRUDE (or at least CRUD) analysis Does the analysis communicate useful information? Does it cover the class diagram completely? (if required by JBGE analysis)			X	Not required in JBGE analysis.
Strengths of the team's model presentation include: The document is well designed, organized and presented. The text accompanying the models is well written in a way that explains the models without going beyond "Just Barely Good Enough" standards. The hyperlinks and indexing made it easy to navigate, and the models were presented in a logical order. The structural models were the strongest section. The class organization was modeled and explained well.				
Suggestions for improving the team's model presentation include: The system is explained well to managers, but developers will require more details. I would suggest including more detailed steps in the use case descriptions and more behavioral models, with special attention to how the behavioral state machine and communication diagrams are constructed.				