

# **ARMS DEMONSTRATION**

---

## **Aircraft Repair Model Simulation**

December 11, 2000

### **TEAM 9: GOLDRUSH**

Nick Capron

Dave Stone

Carina Lansing

Russell Swannack

Kyle Klicker

Miguel Vilchez

Jonathan McPherson

Kelly Yearout

Andy Reynolds

## **Agenda**

---

- 1. Traceability Approach**
- 2. Test Cases (6)**
- 3. Test Case Demonstration**
- 4. Open Issues**
- 5. Current Status**
- 6. Programmer Comments**
- 7. Action Items**

## Traceability Approach

---

- Categorize/number requirements
- Identify testing method for each requirement
- Develop Requirements Traceability Matrix (RTM)
- Link DFD modules to RTM requirement categories
- Test modules against assigned requirements

## Traceability Approach Cont'd

---

Req. ID System Level	Req. ID Sub-system Level	DFD Identifier(s)	Module Name(s)	Verification Method	Tested
A01.00		GUI		D	Pass
	A01.01	GUI		D	Pass
	A01.02	GUI		D	Pass
	A01.03	GUI		D	Pass
	A01.04	GUI, 1.2	Evaluate # of Bays	D	Pass
	A01.05	GUI, 1.3	Eval. Random Seed	D	Pass
	A01.06	GUI		D	Pass

## Demo Test Cases

---

### Case 1:

- Event: Running stored Queuing Scenario #1.
- Stimulus: User selects QS1, enters a valid number of service stations, and presses start.
- Activity: ARMS runs to completion using the logic defined by QS1.
- Response: Results are displayed to output GUI.
- Effect: Verifying the description and logic associated with QS1.

## Demo Test Cases

---

### Case 2:

- Event: Running stored Queuing Scenario #2.
- Stimulus: User selects QS2, enters a valid number of service stations, and presses start.
- Activity: ARMS runs to completion using the logic defined by QS2.
- Response: Results are displayed to output GUI.
- Effect: Verifying the description and logic associated with QS2.

## Demo Test Cases

---

### Case 3:

- Event: Running stored Queuing Scenario #3.
- Stimulus: User selects QS3, enters valid numbers of service stations, and presses start.
- Activity: ARMS runs to completion using the logic defined by QS3.
- Response: Results are displayed to output GUI.
- Effect: Verifying the description and logic associated with QS3.

## Demo Test Cases

---

### Case 4:

- Event: Running stored Queuing Scenario #3.
- Stimulus: User selects QS3, enters invalid numbers of service stations, and presses start.
- Activity: ARMS prompts user for re-entry until valid parameters are entered.
- Response: Error message is displayed to GUI.
- Effect: Verifying the error checking of input parameters.

## Demo Test Cases

---

### Case 5:

- Event: Running stored Queuing Scenario #2.
- Stimulus: User presses Pause button on progress GUI, presses button to Delete an aircraft from the queue, and presses Resume.
- Activity: ARMS deletes that aircraft from the selected queue and then runs to completion.
- Response: Progress GUI shows that the aircraft has been removed from the queue.
- Effect: Verifying the Delete function.

## Demo Test Cases

---

### Case 6:

- Event: Running stored Queuing Scenario #1.
- Stimulus: User presses Pause button on progress GUI, presses button to Insert an event to the event list, and presses Resume.
- Activity: ARMS prompts the user for a description of the event, inserts the event to the event list, and then runs to completion.
- Response: Progress GUI shows that the event has been inserted into the event list.
- Effect: Verifying the Insert function.

## **Extensibility**

---

- Object Oriented language
- Logic based on Queue Scenario Object
- Focused on providing more Queuing Scenarios
- Queue Scenario: Service Stations, Aircraft and Queue Info
- Only handful of input files to modify to add another scenario
- “Create New Queuing Scenario” button - future enhancement-series of modification forms

## **Open Issues**

---

- Veracity of output data
- Testing time in general
- Error trapping

## **Current Status**

---

### **Accomplishments:**

- All deliverables completed

### **Plans:**

- Study for final

### **Problems:**

- Not enough time to perfect the product

## **Programmer Comments**

---

- Understanding SimLib function calls
- How interface SimLib to Java
- GUI appearance and functionality

## **Action Items**

---