Reflection of Actor Based Design

TSA Airport Screening

Geoff Berl, Conor Wright, Kocsen Chung

Document Overview

This document is meant to describe a reflection on how Actor Based design worked and any sort of comments specific to any of our design changes. This was done in conjunction with the TSA Airport Screening assignment

Actor Based Design

Our team realized that actor based design has to be thought of differently than O-O design. The biggest hurdle in this mind set is to know and understand that Actors are not direclty mapped to objects, but rather key points where data has to be manipulated or detained.

Another mind set that we found out about is that an Actor listens and acts, it does not ask. An actor isn't in perpetual wait until a condition happens, it is already implicitly (and efficiently) waiting for messages. This leads on to the final thing that we had to keep in mind, messages. Messages had to be simple and efficient. They are the data that is being handled and manipulated by all these actors and this is the representation of critical data that only someone can act upon at the same time.

If there is one sentence that relates to Actor Based Design, it's 'delegate, delegate, delegate'.

It did help that the project is a very close to relate to actors and messages given the story behind it (airport screenings).

Changed to Our Design

There were several changes to our design that we thought would fit best to make the project follow actor based design guidelines.
- Passenger is not an actor
  - In O-O it's almost obvious to think of a passenger, as an actor but in this case actor is just data that is being handled from actor to actor.
- System is the death causer
  - Before we weren't sure who should tell everyone to stop. And we didn't want System to know about everyone.
  - We decided to propagate the poison pill (per actor based design) to kill everyone safely as a message.
- Document Check is an Actor
  - Previously document check wasn't an actor, but we decided it was in best interest

- Config Messages vs Constructors
  - At first we expected all messages to be in constructors.
  - Instead we initial messages to configure references to the system.