Week 2 Homework Problems

Exercise 3.2:

Individuals and interactions take precedence over tools and processes. By letting the development team know what others are doing and utilizing individual skills, formal communication and process assurance overheads are eliminated. This allows the team to concentrate on creating functional software.

Working software is preferred over thorough documentation. It takes no time to create, review, or manage documentation. The creation and testing of code take up time. Agile developers receive feedback from customers directly throughout the development process rather than researching and discussing what should be in a system contract.

Exercise 3.3:

Advantages:

They simulate actual events that frequently occur so that the system can accommodate the most typical user actions. The stories are simple for users to comprehend and evaluate. They stand for functional increments; telling a tale gives the user something of value.

The drawbacks:

Due to its casual nature, it may be challenging to identify any incompleteness. They put more emphasis on functional than non-functional needs. When using stories, it is impossible to represent system needs that cut across multiple domains, such as performance and reliability. Architectural design is challenging since it's unclear how the system architecture and the user stories relate to one another.

Exercise 4.2:

Can a consumer purchase many tickets for the same location at once, or does each ticket need to be purchased separately? If a mistake is made, may clients cancel a request? What should happen if a bad card is entered into the system? What occurs if users attempt to insert their card before choosing a destination? If a user wants to purchase a second ticket for a different location, do they have to press the start button once more? Should the system offer tickets to all potential destinations, or should it only sell tickets between the station where the machine is located and direct connections?

Exercise 4.4:

No more than five minutes should be spent with the system down. After a system failure, the recovery period shouldn't last longer than two minutes. There shouldn't be more than 20 minutes of total system downtime.

Within 0.5 seconds of a consumer pressing a button on the device, the display ought should change. After receiving credit card validation, no more than 10 seconds should pass before a ticket is issued. Customers should be informed that credit card validation is underway via a status message on the display. This lets the consumer know that validation, which could take some time, is still being done and that the system hasn't completely failed. For ticket issue requests, a failure rate of 1 in 10,000 is considered to be acceptable.