

Use Case Model  
for the  
So Long Sucker Simulation

Submitted by

Mr. Steven W. Braeger  
COP 4331

March 23, 2009

# Table of Contents

<b><u>Title</u></b>	<b><u>Page</u></b>
<a href="#"><u>1.0 System Summary.....</u></a>	<a href="#"><u>1</u></a>
<a href="#"><u>1.1 Document Scope.....</u></a>	<a href="#"><u>1</u></a>
<a href="#"><u>1.2 Motivation and Business Case.....</u></a>	<a href="#"><u>1</u></a>
<a href="#"><u>1.3 Concept of Operation.....</u></a>	<a href="#"><u>2</u></a>
<a href="#"><u>1.4 So Long Sucker Architecture and Interfaces.....</u></a>	<a href="#"><u>5</u></a>
<a href="#"><u>1.4.1 Physical Design and Actor Interfaces.....</u></a>	<a href="#"><u>5</u></a>
<a href="#"><u>1.4.2 Activity States.....</u></a>	<a href="#"><u>6</u></a>
<a href="#"><u>2.0 Use Case Specifications.....</u></a>	<a href="#"><u>9</u></a>
<a href="#"><u>2.1 Decide Next Player.....</u></a>	<a href="#"><u>9</u></a>
<a href="#"><u>2.2 Transfer Prisoner.....</u></a>	<a href="#"><u>10</u></a>
 <a href="#"><u>3.0 Glossary</u></a>	<a href="#"><u>17</u></a>
<a href="#"><u>3.1 References</u></a>	<a href="#"><u>17</u></a>

## Table of Figures

<b><u>Title</u></b>	<b><u>Page</u></b>
Figure 1.3 Use Case Diagram for So Long Sucker Simulation.....	2
Figure 1.4-1. So Long Sucker Simulation Physical Design.....	5
Figure 1.4-2 Play Mechanisms.....	6
Figure 1.4-3 Activity Transitions Controlling the game.....	7



## 1.0 System Summary

### 1.1 Document Scope

This document describes the functional requirements (use cases) and design constraints (non-functional requirements) of the game “So Long Sucker”, as viewed from the perspective of a potential simulation system So Long Sucker Simulation. Additionally, the document also describes these requirements of the game “So Long Sucker” to someone prepared to use the system, the Player. Its purpose is to formally capture and present an operational view of the game “So Long Sucker” and to enumerate the game requirements as understood by the developer, Nash Simulations, Inc.

### 1.2 Motivation and Business Case

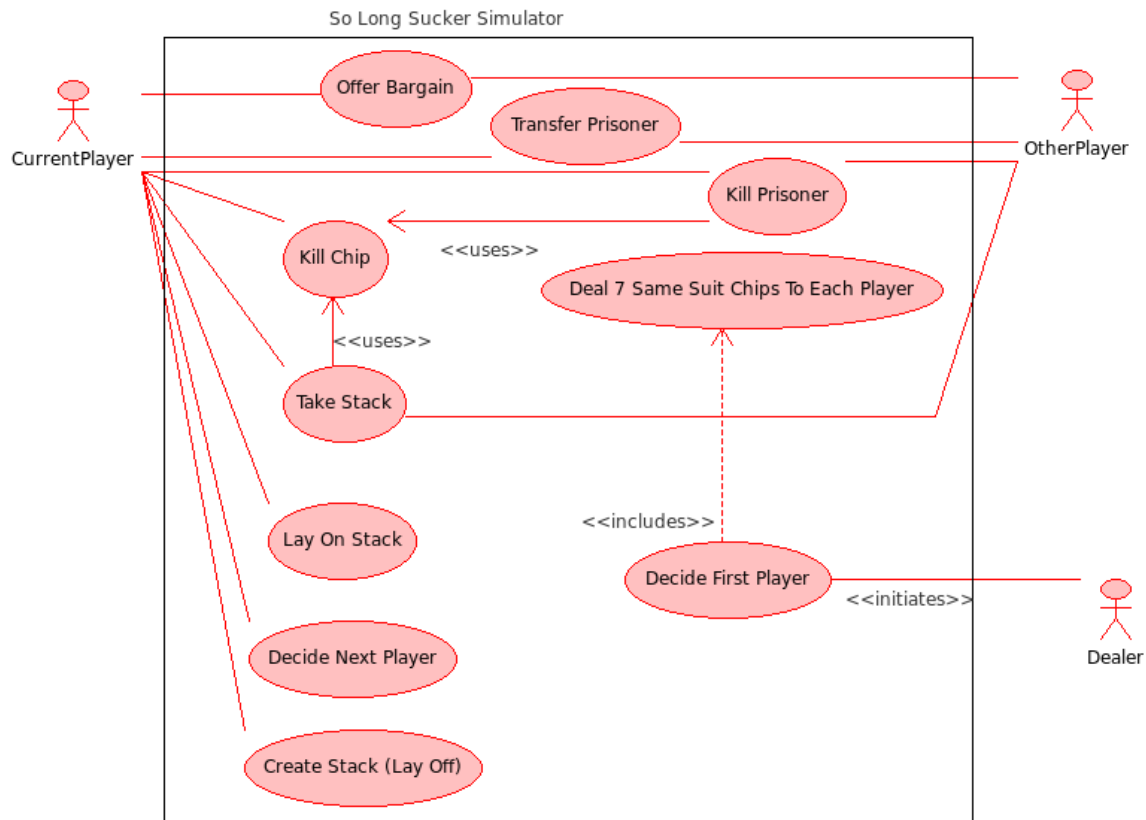
**Nash Simulations, Inc.**, hereafter referred to as “Nash Sims”, has observed a number of its most valuable employees become fed up with company practices regarding software development, and many of them have left for greener pastures at many of their competitors. Additionally, many of those whom Human Resources (HR) has convinced to stay, (even those not as valuable to the team) enjoy massively overweighted salaries and bonuses. In the face of decreased product quality and increasing labor costs, upper management tasked Research to investigate this problem. After performing a detailed internal investigation into the details of HR, Research employees discovered that 45%-65% of the employee defection cases could have been resolved with more effective and more competitive employment offers. In addition, 70% of the employees whom Upper Management identified as likely overcompensated were, in fact overcompensated. Of these, 10% were overcompensated due to payroll errors (with a fix to be suggested in document A4352), and 90% were overcompensated due to middle management fear of social rejection and lack of confidence to reject employee demands.

As a means of solving this problem, Research decided to come up with a way to better train HR staff to bargain with potential and current employees. Additionally, they have decided to lay the groundwork for future work training an artificial intelligence system to effectively bargain for resources with potential applicants automatically. Therefore, Researchers decided that a simulation of a simple game based on economics professor John Nash's thesis work with collective bargaining would be the best choice as a general framework to teach the tools required. The specific game chosen, called So Long Sucker, was chosen, because it is well known to the Research team, was created by John Nash himself, and is well-known for the bargaining behavior it teaches, as well as the ability to break and create bargains.

The remainder of this section and the next present a functional description and external design of the So Long Sucker Simulation, the proposed game designed to achieve the business goals recommended by **Nash Simulations** Research team.

### 1.3 Concept of Operation

Figure 1.3 presents a functional model of the envisioned So Long Sucker system. User-oriented functions (*use cases*) are identified by round bubbles appearing inside the box denoting the physical and logical boundary of the new system; All entities shown outside the system box represent *actors*, that is, independent agents that must interact with the system to accomplish all use cases defined by the diagram. As shown in the figure, there are three actors: the *OtherPlayers*, the *Dealer*, and the *CurrentPlayer*.



**Figure 1.3 Use Case Diagram for So Long Sucker Simulation**

**[Para.1]** The **Dealer** denotes the first player of the game. The goal of this actor is to prepare the So Long Sucker Simulation for the setup of the game. The Dealer does two actions, consecutively to setup and initiate the game, (1) first, he deals out 7 chips to each player in the game, and (2) chooses the next player to go in the game, which is the first player.

**[Para.2]** At any point in the game, with or without warning, or for any reason, any **OtherPlayer** may choose to trade, or execute any Prisoners in his possession at any time. If control is given to him, **OtherPlayer** becomes **CurrentPlayer**. Additionally, he may offer a bargain or deal to the **CurrentPlayer** or to any **OtherPlayer**.

**[Para.3]** The **CurrentPlayer** may, like other players, initiate bargains, execute or trade prisoners at any time, for any reason. However, unlike other players, the **CurrentPlayer** must play a chip on the board, and he must choose between playing on a new stack **[Para.12]**, or playing on a stack that already exists**[Para.11]**. After he is done, he must select a new player based on the stack rules for player selection **[1,Para.13]**.

[Para.4] **Offer Bargain** is the action that either **CurrentPlayer** or any **OtherPlayer** can take, that allows an offer to be made that offers a deal of some kind, in exchange for goods. The offer can take any form whatsoever, and may be made at any time. [1]

[Para.5] **Transfer Prisoner** is another action that players may take. It specifies that a prisoner chip in the players hand may be given to any other player. This may be done at any time. [1]

[Para.6] **Kill Prisoner** is another action that any player may take. It specifies that a prisoner chip in the players hand may be transferred to the discard pile, through the **Kill Chip** action [Para 7]. This may be done at any time. [1]

[Para.7] **Kill Chip** is a subprocess of both **Take Stack** [Para.8] and **Kill Prisoner**[Para.6]. A chip is killed when it is put into the discard pile, and therefore forever removed from the game.

[Para.8] At any point in the game, if two chips of the same color are consecutively played on the top of the stack, the player of that color must execute the **Take Stack** action, and put all the chips, except for one, into his hand. The one that is left is immediately **Killed** [Para. 7]. The player in question now becomes a **CurrentPlayer**.

[Para.9] At the beginning of the game, the **Dealer Deals 7 same-suit chips** to each player. Every Player in the game receives seven chips in the same color.

[Para.10] At the beginning of the game, after the chips are dealt [Para. 9], the dealer randomly promotes an **OtherPlayer** to **CurrentPlayer** .

[Para.11] During his turn, a **CurrentPlayer** plays a chip on the stack or off of the stack. If he cannot play a chip, he is out of the game. If he **Plays on a Stack**, If there are two chips of the same color on the stack, then a player takes the stack, according to the rules in [Para. 8]. Otherwise, the **CurrentPlayer** picks a new **CurrentPlayer** according to [Para. 13].

[Para.12] During his turn, a **CurrentPlayer** plays a chip on the stack or off of the stack. If he cannot play a chip, he is out of the game. If he **Plays off a Stack**, then he creates a new pile, and picks a new **CurrentPlayer** according to the rules in [Para. 13]

[Para.13] **To get a new player**, the stack most recently played upon is investigated. The **CurrentPlayer** nominates a new player from the set of players NOT represented in the last stack played upon. If all players are represented, the last player in the stack who has been nominated last goes next. [1]

## 1.4 So Long Sucker Architecture and Interfaces

### 1.4.1 Physical Design and Actor Interfaces

Figure 1.4-1 illustrates the physical design of the proposed So Long Sucker Simulation.





**Figure 1.4-1. So Long Sucker Simulation Physical Design**

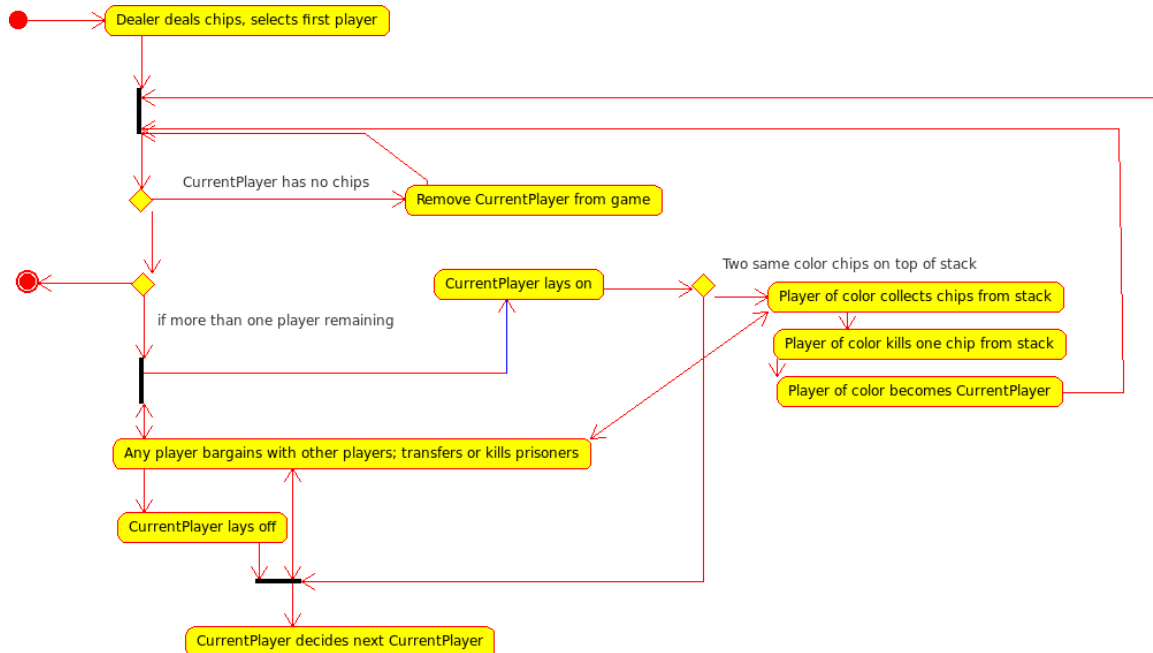
Figure 1.4-2 illustrates the primary play mechanisms. Represented is a close-up view of the play area, with several stacks, an example of a chip, and the current player “Playing On” to one of the stacks.



**Figure 1.4-2 Play Mechanisms**

## 1.4.2 Activity States

The behavior of the So Long Sucker Simulation is determined by the state of several independent components, working together



**Figure 1.4-3 Activity Transitions Controlling the game.**

Figure 1.4-3 describes the events and conditions that determine the execution and completion of the game play.

## 2.0 Use Case Specifications

This section gives the detailed specifications for a subset of the use cases described above. In the Use Case Model, use case specifications should document the interaction scenarios between the *CurrentPlayer*, the *OtherPlayers*, the *Dealer*, and So Long Sucker Simulation, they should not describe the interaction between internal components of the So Long Sucker Simulation. Pre-conditions and Post-conditions should be expressed in terms of interaction events with one or more actors and/or the system states defined in Section 1.4. Only in the Analysis and Design Models should use case specifications document internal system behavior.

### 2.1 Decide Next Player

ID: UC1

**Purpose**

The purpose of the Decide Next Player use case is to choose the next player to play based on the choices of the CurrentPlayer and to move the game forward in a fair manner, allowing everyone to play.

**Pre-conditions**

1. The CurrentPlayer must first play a chip on or off.
2. There must be players remaining in the game.

**Scenario (Connect)**

1. The CurrentPlayer examines the stack most recently played upon.
2. Based on the rules in [1], the CurrentPlayer chooses the next player.

**Post-condition**

A new turn begins, and CurrentPlayer has been changed into the next player to go, (if that player is not the old CurrentPlayer, the OtherPlayer is promoted into a CurrentPlayer)

## **2.2 Transfer Prisoner**

**ID: UC2****Purpose**

The purpose of the Transfer Prisoner use case is to transfer one prisoner chip from one players hand to another.

**Pre-conditions**

The player initiating the transfer has at least one chip of any color except his own in his hand.

There is at least one other player in the game.

**Scenario**

1. The player hands the chip to the hand of another player

**Post Condition**

The chip is now in the hand of another player.

### 3.0 Glossary

**Dealer:** This refers to the actor who starts a game with the So Long Sucker Simulator and fulfills no other role.

**CurrentPlayer:** This refers to the actor whos turn it currently is to play.

**OtherPlayer:** This refers to any actor who is playing the game but is not the CurrentPlayer

**Stack:** This refers to any pile of chips in the playing area.

**Laying Off:** This refers to the action of creating a new stack by placing a chip within the playing area, but not on any currently existing stack.

**Laying On:** This refers to the action of placing a chip on a particular stack.

**Prisoner:** This refers to any chip that is held by a player other than the original owner.

**Kill:** This refers to any chip that has been forced out of play.

**Bargain:** This refers to any agreement made among players during the game.

### 4.1References

[1] [http://en.wikipedia.org/wiki/So\\_Long\\_Sucker](http://en.wikipedia.org/wiki/So_Long_Sucker)

[2] <http://www.boardgamegeek.com/boardgame/8304> (Images)

[3] Workman03, Use Case Model for Coffee Maker II