

Ali Sytsma

Professor Arias

Software Development 1

7 April 2017

## **Abstract**

The program is a text-based adventure game set in the Medieval Ages. Its protagonist is a young farm girl who is sent to court by her uncle to gain influence or money to save her family's farm. I focused heavily on user choices, and what they decide throughout the game will determine what happens. The game changes depending on their relationships with other characters, how they decide to interact with other characters, and what they decide to do at decision points.

## **Introduction**

The motivation for this project is to create a fun, detailed, choice-based text game. Giving the player options and different outcomes creates a more interactive experience than a linear game would. In this paper, I will discuss the detailed system description, the requirements, the literature survey, and the user manual.

## **Detailed System Description**

The program was designed so that all of the chapters reference the Global class. This way, when a variable is altered in one chapter, it is altered for the future ones as well. For example, if your relationship with Laith is 5 by the end of chapter 1, it will be stored in the Global class so that your relationship will remain at 5 for the beginning of chapter 2. The program uses many switches and if statements for decisions, as well as while loops to re-prompt the user if their input is invalid.

Global
- marcella: int - florence: int

<ul style="list-style-type: none"> <li>- jaime: int</li> <li>- lucas: int</li> <li>- laith: int</li> <li>- pack: int</li> <li>- steal: int</li> <li>- invite: int</li> <li>- dance: int</li> <li>- court: int</li> <li>- blame: int</li> <li>- sneak: int</li> </ul>
<ul style="list-style-type: none"> <li>+ enter()</li> <li>+ getName(): String</li> <li>+ getMarcella(): int</li> <li>+ getFlorence(): int</li> <li>+ getJaime(): int</li> <li>+ getLucas(): int</li> <li>+ getLaith(): int</li> <li>+ getPack(): int</li> <li>+ getSteal(): int</li> <li>+ getInvite(): int</li> <li>+ getDance(): int</li> <li>+ getCourt(): int</li> <li>+ getBlame(): int</li> <li>+ getSneak(): int</li> <li>+ setName(name: String)</li> <li>+ setMarcella(marcella: int)</li> <li>+ setFlorence(florence: int)</li> <li>+ setJaime(jaime: int)</li> <li>+ setLucas(lucas: int)</li> <li>+ setLaith(laith: int)</li> <li>+ setPack(pack: int)</li> <li>+ setSteal(steal: int)</li> <li>+ setInvite(invite: int)</li> <li>+ setDance(dance: int)</li> <li>+ setCourt(court: int)</li> <li>+ setBlame(blame: int)</li> <li>+ setSneak(sneak: int):</li> <li>+ getHighestRelationship(): String</li> </ul>

Chapter 1
<ul style="list-style-type: none"> <li>+ hay: boolean</li> <li>+ pick: boolean</li> <li>+ water: boolean</li> <li>+ chores: String</li> </ul>

Chapter 2
<ul style="list-style-type: none"> <li>+ response1: int</li> <li>+ response2: int</li> <li>+ response3: int</li> <li>+ choose: boolean</li> </ul>

Chapter 3
-----------

```
+ response: int
+ gotoroom: boolean
+ room: String
+ open: boolean
+ door: String
+ response2: int
+ choose: boolean
+ choice: int
+ bring: String
```

```
+ interact: int
+ talk: String
+ response4: int
+ response5: int
+ response6: int
+ choose2: boolean
+ take: String
```

```
+ open: boolean
+ door: String
+ go: boolean
+ room: String
+ response1: int
+ response2: int
+ response3: int
+ response4: int
+ response5: int
+ open2: boolean
+ letter: String
+ open3: boolean
+ door: String
+ response7: int
+ listen: boolean
```

## Requirements

As the main purpose of the game is to be choice-based and player directed, the game must have multiple outcomes and provide a difference experience based on how the user plays. For example, having a negative attitude will result in characters not liking the player, and being pleasant will result in more approval. It is apparent in the way the characters talk to the player how they feel about them. Another requirement is that, upon ending the game, a file will be made with all of their stats and their ending so that players can compare amongst themselves.

## Literature Survey

There are many text-based games out there. However, I was only able to find one that dealt with a similar theme, which was “Affairs of the Court.”

## User Manual

The user interacts with the system by typing. For dialogue, they enter the corresponding number to the line they want their character to say. For other situations, they type in commands, such as “take hairbrush” or “go to room.”

## Conclusion

The goal of the program is to provide the user with a fun, interactive, and choice-driven experience. Hopefully, because their choices shape the game, there will be replay value and players will enjoy experiencing different outcomes.