CYAOG

**Choose Your Own Adventure Game**

A user-generated text adventure

# Project Statement

Choose Your Own Adventure Game, or CYOAG for short, is an idea Steven Kitzes came up with, based on the choose-your-own-adventure books he grew up with. The basic idea behind these books was that the reader would be presented with very short chapters for context; then get to pick from a handful of options leading to different page numbers, ultimately resulting in different paths through a story that always started the same but had many possible endings. Steven enjoyed reading, and was thrilled to discover books with variable outcomes.

Now, later in life, Steven has come to love writing as much as he loves reading. Having grown into a career in software development, and remembering his love of choose-your-own-adventure books, he decided it would be a great potential side project to build a website that combines all of these loves into one. The CYOAG is, thus, a web app that proposes to let users read and/or create content for a choose-your-own-adventure game online. This broad-strokes definition will be described in much greater detail throughout this document.

# Definitions

The following terms are used throughout this document and warrant some explanation:

|  |  |
| --- | --- |
| *account* | Unless otherwise specified, a CYOAG account. Not to be confused with a Facebook account or Google+ account, which in this document shall be referred to as such to prevent confusion. |
| *child* | Any *node*,reachable via a *path*, that follows immediately after the current *node* |
| *CYOAG* | Create Your Own Adventure Game, the name of this web app |
| *node* | A collection of all data representing a particular point in the story. This includes *snippet* and *ID* data for the current part of the story, the current *node*’s *parent*, and zero or more *paths*. |
| *node ID* | The unique ID of the current *node* |
| *node snippet* | The main story content at the current *node* |
| *parent* | The *node* immediately preceding the current *node*; in other words, the current *node* is reachable via a *path* from the *parent* |
| *path* | One option (usually of several) for story progression, leading away from the current *node* toward another, subsequent *node*. |
| *path ID* | The unique ID of the *node* to which a given path leads |
| *path snippet* | A short piece of text describing one of a reader’s potential choices for progressing the story past the current *node*. Note that a given *node*’s *path snippet* is displayed with that *node*’s *parent*, to provide an option for progression that leads from said *parent* to the current *node*. |
| *root node* | The very first *node* of the story, provided free of charge by Steven Kitzes, from which all *paths* and subsequent *nodes* follow |
| *trajectory* | A complete series of *paths* leading from the *root node* all the way to the current *node* |

# Basic Feature Families

Observing this application at a high level, the following basic features are proposed (with further details provided later in this document):

## Read the story

**Any user** visiting the site shall be able to read the story, making path selections to progress as far through the story as desired without needing an account.

## User accounts

**Any user** shall be able to make and log into a CYOAG account by associating either a Facebook account or Google+ account.

## Continuation

If a user was logged in while reading the story, it should be possible for that user to pick up where the story left off.

## Contribution

**Any user** who is logged in shall be able to create new paths, leading to new story nodes form existing nodes.

## Voting

**Any user** who is logged in shall be able to apply positive or negative reversible sentiment to any node to which they have navigated.

# User Stories

# Explicit Restrictions

There are certain user behaviors we do *not* want the user to be able to do. These behaviors are prevented or moderated by design.

## Edit

Edits to a node – including its node snippet and path snippet – shall be allowed only by that node’s original creator, and only if that node has zero child nodes.

## Delete

A node shall be possible to delete only by its creator, and only if it has no child node yet.

## Multiple Paths from a Single Node

Any given user shall be permitted to contribute only a single child to any node.

## Consecutive Paths

A user shall not be permitted to contribute a child node to a node that was created by that same user. In other words, all child nodes must be created by users that did not create the parent.

## Snippet Length Restrictions

A node snippet shall have a minimum length of 250 characters, and a maximum length of 1,000 characters. A path snippet shall have a minimum length of 3 characters, and a maximum length of 100 characters.

## Voting

A user shall be unable to vote (either up or down) for their own nodes.

# Architecture and Technology Stack

# Data Model