

Crazy Night Out

Game Design Treatment



Copyright © Sonic Boom, Inc.
Proprietary and Confidential

Table of Contents

1. Overview	3
2. Overall Game Structure.....	4
2.1. Play Cycles.....	4
2.2. Game Length.....	4
2.3. Turn Structure.....	4
2.3.1. Scenario 1: Daily Cycle	5
2.3.2. Scenario 1: Weekend-Centric Cycle	6
2.3.3. Work Week-Centric Cycle	7
3. Turns.....	8
3.1. Turn Types.....	8
3.1.1. Major Event Turn.....	8
3.1.2. Story Turn.....	9
3.1.3. Activities Turn.....	10
3.1.4. Minigame Turn.....	10
3.1.5. Turn Concept Example: Ethan and Abbie.....	11
3.1.6. Turn Concept Example: Sue	12
4. Character Traits	15
5. Episode Scripts	17
5.1. Script Schedule	17
5.2. Story Lines.....	18
5.3. Milestones.....	18
5.4. Script Logic.....	19
6. Other Major Game Features	20
6.1. People	20
6.2. Journal	20
6.3. My Character.....	20
6.4. Popularity	20
6.5. Replayability	20
7. Minigames	21
7.1. Overview	21
7.1.1. Minigame Selector.....	21
7.2. Minigames	21
7.2.1. Beach Volleyball.....	21
7.2.2. Racquetball	22
7.2.3. Matchmaking	22
7.2.4. Spin the Bottle.....	22
7.2.5. Hot Tub Wrestling	23
7.2.6. Rock God	24
7.2.7. Rave All Night.....	24
7.2.8. Dream Date.....	25
7.2.9. Make the Scene	25
7.2.10. Music Match.....	26
8. Notes.....	28

1. Overview

Crazy Night Out will be a story adventure game in the format established by games like Surviving High School and Surviving Hollywood. Players take the roll of a character and progress through a branching story. Choices the players make and how they develop their characters determine how the story progresses and ends. There are several different outcomes.

The actual title and game theme for Crazy Night Out is being researched and will be chosen later. It's expected, however, that the game will include relationship and dating themes, music, dancing, and other fun stuff for young adults, and these themes are used throughout this design document.

Crazy Night Out features:

- A game structure in which a player guides the life of a character over a period of time like 3-4 months. Numerous turns occur in the game across the simulated weeks and days, which create a sense of the passage of time and allow the story lines to build and resolve in a satisfying manner.
- Episodes. The game will have one episode on launch, but additional episodes will be added over time.
- Multiple types of turns, which all for the story to be told, major events to happen, minigames to be played, and other activities to occur.
- Character traits, which feed into the story lines. Each trait measures the ability of the character in some manner. For example, happiness would measure the character's overall happiness while smarts would measure how well the character is doing in college.
- Episode scripts. The stories of an episode unfold in story lines. Each episode starts with a single line but story lines then branches out and merge based on the player's decisions and traits.
- Popularity. The game can track how popularity the players find certain items like dance clubs are in an episode. The results of popularity are feed back into the game, giving it a subtle reality-show element.
- A fun selection of minigames themed to the game's overall topic. All episodes would use these minigames.



2. Overall Game Structure

2.1. Play Cycles

The game is played in a series of cycles based on the concept of **weeks**.

Crazy Night Out: Game Structure Overview

Preparation Player set up, episode choice, gender choice, etc.
1st Cycle "Weekly" cycle of 4-7 days, each day with about 2-4 turns
2nd Cycle "Weekly" cycle of 4-7 days, each day with about 2-4 turns
3rd Cycle "Weekly" cycle of 4-7 days, each day with about 2-4 turns
More Cycles ...
Last Cycle "Weekly" cycle of 4-7 days, each day with about 2-4 turns
Outcome Results of the episode

2.2. Game Length

Depending upon the subject chosen for an episode, a week can consist of 4-7 **intervals** (an interval is often a day). The game engine will be designed so that it can handle episodes with different numbers of weeks in them.

The number of weeks in an episode needs to be chosen to give a satisfying depth to an episode, allowing the story to unfold and resolve. However, an episode needs to avoid excessive length, which could bore players and would require extra content to be created.

# of weeks per episode	# weeks x 4 intervals x average 4 turns/interval	# weeks x 6 intervals x average 2.5 turns/interval	# weeks x 7 intervals x average 2 turns/interval
10	160 turns	150 turns	140 turns
13 (3 months)	208 turns	195 turns	182 turns
17 (4 months)	272 turns	255 turns	238 turns
26 (6 months)	416 turns	390 turns	364 turns

The cells in green are those with a good number of **turns** in the game. Fewer than about 160 seem insufficient while more than about 240 seem to be too many. (For comparison, Surviving High School seems to have about 150-175 turns in a game.)

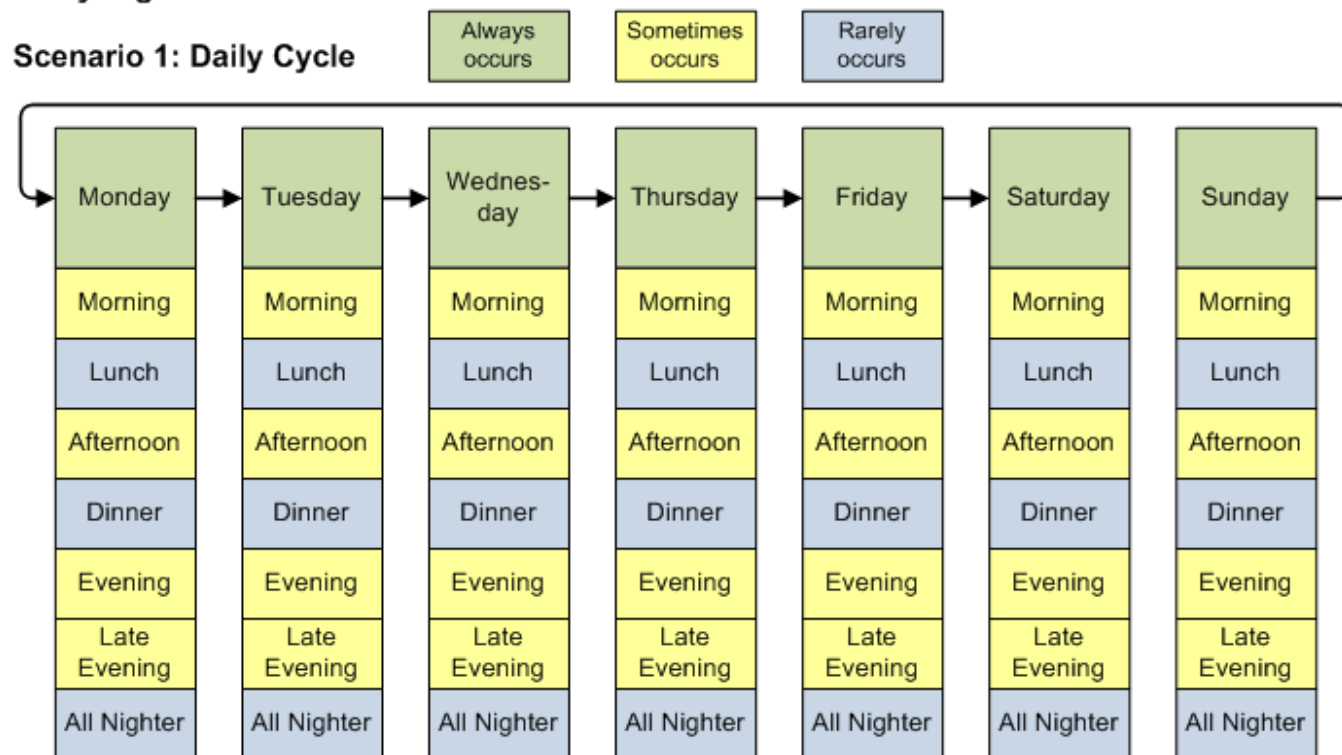
2.3. Turn Structure

The number of intervals in a week depends upon the subject to be chosen for an episode. Each interval is broken up into a number of possible turns. Only a few turns will actually occur in an interval, as determined by the episode's **script**. The variable number of turns per interval and variable times at which turns can occur in an interval helps create a sense of time passing and a feeling that everything isn't precisely scheduled.

There are three main scenarios for the number of intervals in a week, which are covered in the following sections.

2.3.1. Scenario 1: Daily Cycle

Crazy Night Out Turn Structure

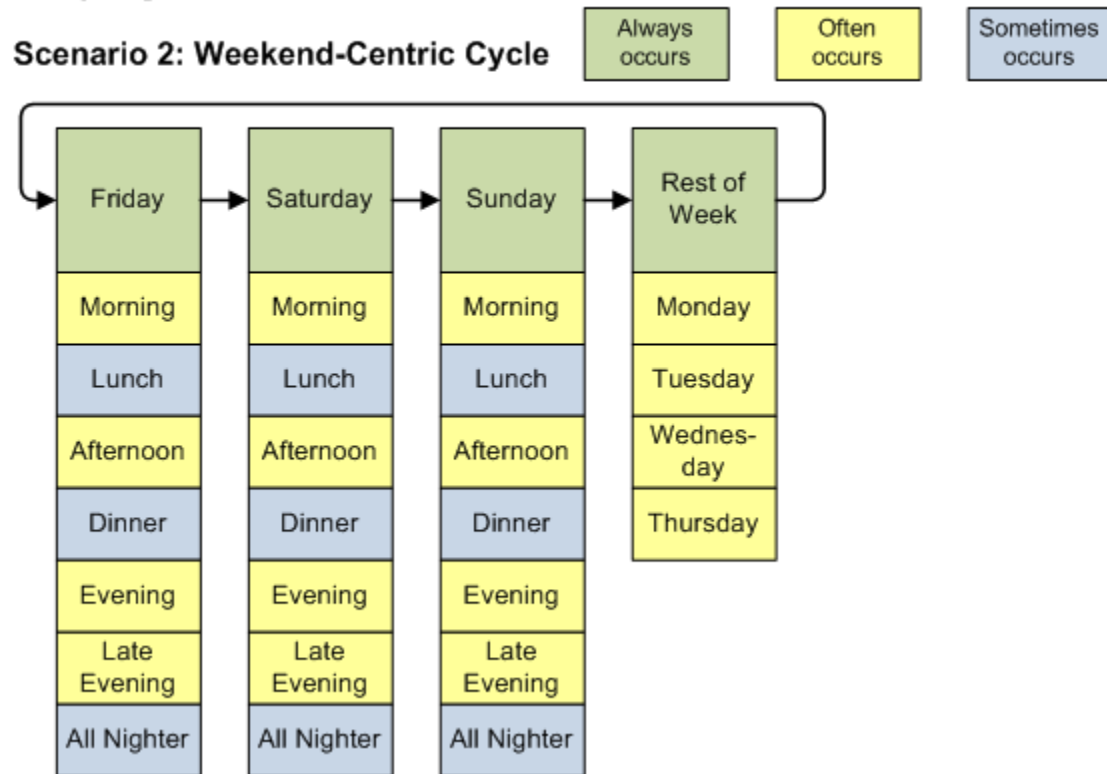


The daily cycle has turns seven intervals in a week, with each one being a day. This would be appropriate for episodes in which the action is spread out over the entire week, with neither work days nor week ends particularly favored.

Every day, something occurs, with two turns per day being the average. Most of the action should occur in the morning, afternoon, evening, or late evening turns. Lunch and dinner turns should be used for special dining events (e.g., a family picnic event or a big dinner date). All-nighter turns should occur when the player makes a special decision to stay up all night. For example, in a late evening turn, the player is having a great time at a dance club with his or her date. The course of the turn goes so well that they're having tremendous fun, and the player gets the choice of dancing till dawn.

2.3.2. Scenario 1: Weekend-Centric Cycle

Crazy Night Out Turn Structure

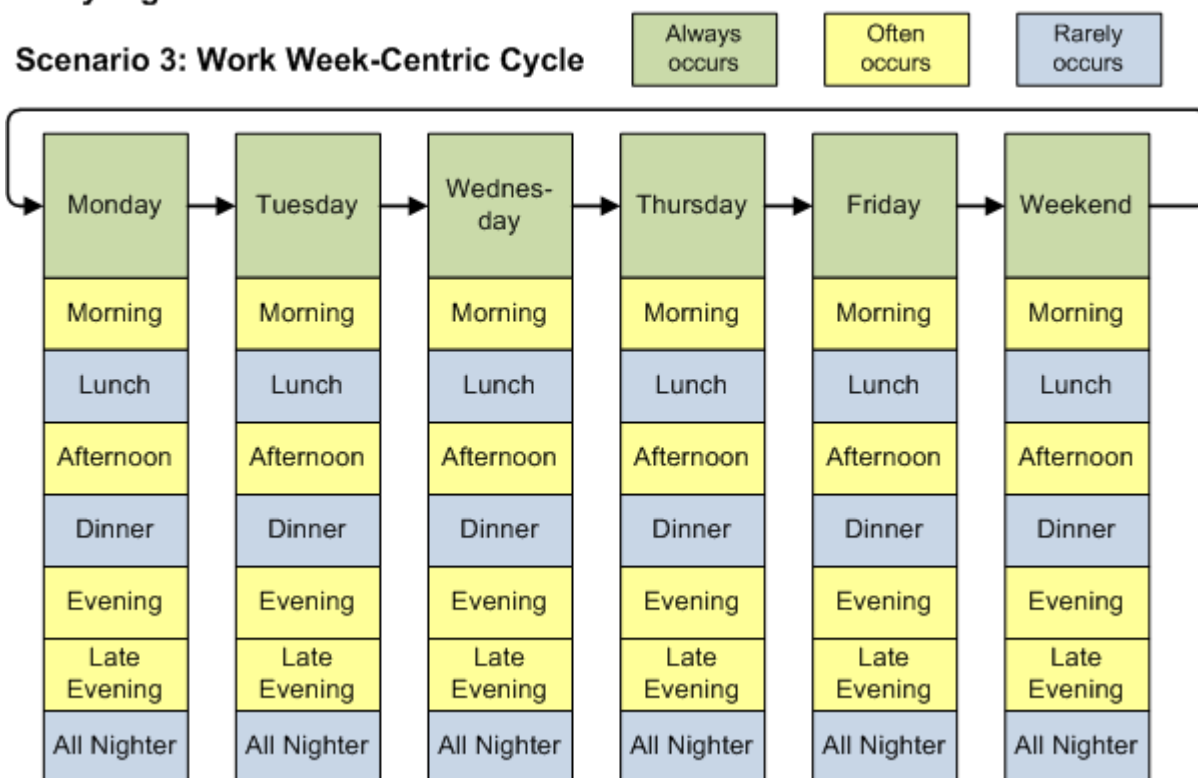


This cycle has turns four intervals in a week, featuring Friday, Saturday, Sunday, and the rest of the week. This would be appropriate for episodes in which the action is concentrated on weekends and the work week is deemphasized. For example, this would be suited for an episode following the life of an aspiring rock band member.

Every interval, something occurs, with four turns per interval being the average.

2.3.3. Work Week-Centric Cycle

Crazy Night Out Turn Structure



This cycle has turns six intervals in a week, featuring the work week. This would be appropriate for episodes in which the action is concentrated on the work week and the weekend is deemphasized. For example, this would be suited for an episode that concentrates on an office worker.

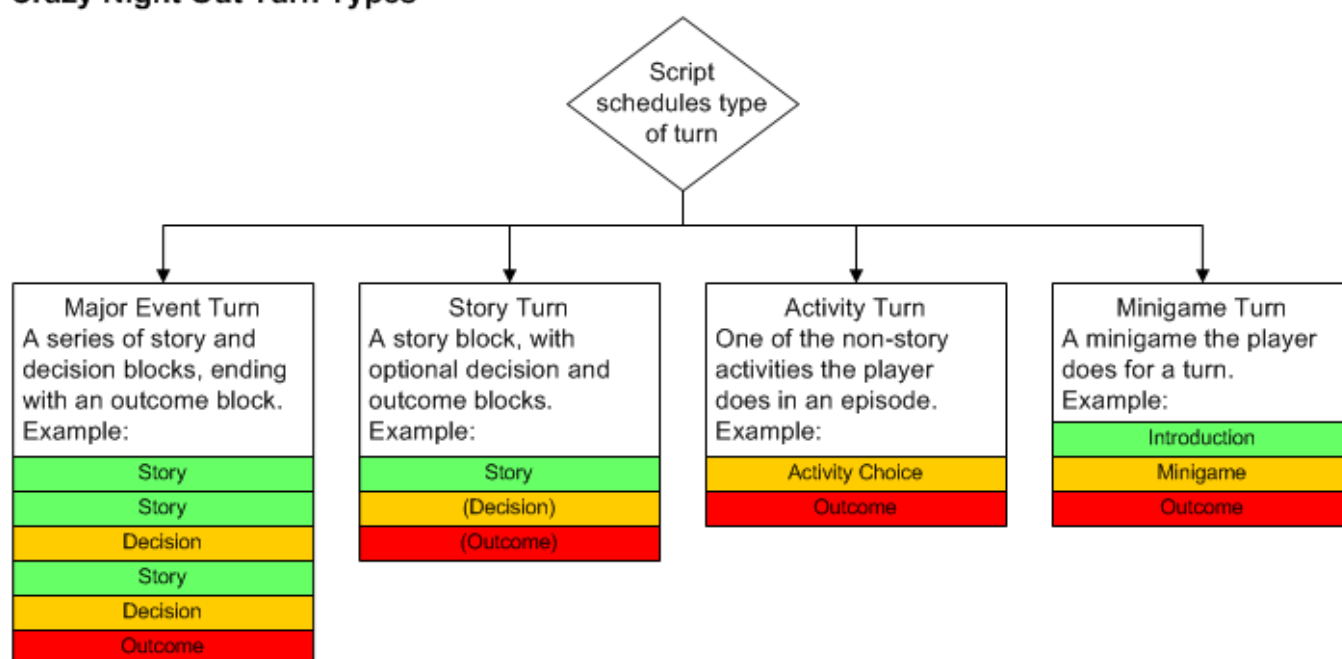
Every interval, something occurs, with 2.5 turns per interval being the average.

3. Turns

All action in the game occurs in the turns. There are four types of turns that can occur: **major event turns**, **story turns**, **activities turns**, and **minigame turns**. The episode's script schedules the turns and their types.

3.1. Turn Types

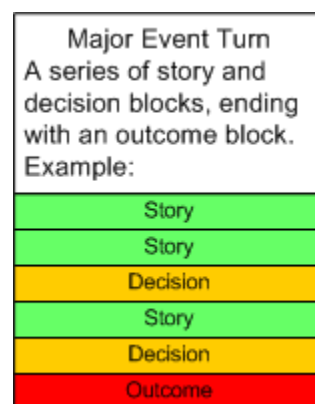
Crazy Night Out Turn Types



3.1.1. Major Event Turn

This type of turn occurs when the script schedules a major event for the episode. The storyline of the episode is significantly advanced during the time. The turn is organized in a set of blocks (the script sets the number of blocks in such a turn):

- **Story Blocks.** A story block tells a segment of the story. It can have graphics and text. The player does not make game decisions in this type of block. There should be multiple story blocks in a major event, interspersed with decision blocks.
- **Decision Blocks.** A decision block poses a set of choices for the player to make. It can have graphics, text, and a menu of two or more choices. The choice the player makes in the decision determines how the story branches and how the player's **traits** change. There should be at least one decision block in a major event and likely several.
- **Outcome Block.** The event ends with an outcome block, which tells the player of any significant results of the event, such as changes to traits or to the story. It can have graphics and text.



The results of decisions in major event turns can have random elements in them. For example, a player may make a decision that affects a trait. The script can specify that the trait is affected by a

fixed amount or it can specify a range by which the trait is affected. In the later case, the range can be affected by the current level of the trait. (For example, it may be harder to increase a trait that is already high level as opposed to one that's medium level.)

The final turn of an episode is always a major even turn, with the outcome of the episode revealed in the outcome block.

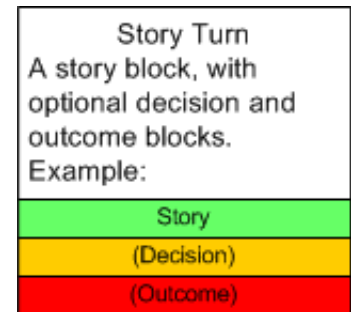
See the two concept examples below for major event turns.

3.1.2. Story Turn

This type of turn occurs when the script schedules a turn that advances the episode's story in some way. It is similar to a major event turn, except not as long or involved.

The turn is organized as one or more blocks (per the script):

- **Story Block.** A story block tells a segment of the story. It can have graphics and text. The player does not make game decisions in this type of block. There should typically be one story block (although the turn could have a decision block instead).
- **Decision Block (Optional).** A decision block poses a set of choices for the player to make. It can have graphics, text, and a menu of two or more choices. The choice the player makes in the decision determines how the story branches and how the player's traits change. A decision block is optional; the story may not need one this turn.
- **Outcome Block (Optional).** If the turn has a decision block, it can also have an outcome block, which tells the player of any significant results of the turn, such as changes to traits or to the story. It can have graphics and text. An outcome block is optional and is only required if any significant result occurs in the turn.



The results of decisions in story turns can have random elements in them. For example, a player may make a decision that affects a trait. The script can specify that the trait is affected by a fixed amount or it can specify a range by which the trait is affected. In the later case, the range can be affected by the current level of the trait. (For example, it may be harder to increase a trait that is already high level as opposed to one that's medium level.)

Examples of story turns:

Example 1:

Story block advances story.

Decision block calls for player decision. Player's choice has significant outcome.

Outcome block wraps up turn.

Example 2:

Story block advances story.

Decision block calls for player decision. Player's choice has no significant outcome for turn.

Example 3:

Story block advances story. No decision required.

Example 4:

Decision block calls for player decision. Player's choice has significant outcome.

Outcome block wraps up turn.

Example 5:

Decision block calls for player decision. Player's choice has no significant outcome for turn.

3.1.3. Activities Turn

This type of turn occurs when the script schedules a standard activity. The story does not advance in any significant. Instead, the player chooses an activity to do, which can change the player's traits. The turn is organized in two blocks:

- **Activity Choice Block.** An activity choice block lists a set of activities the player can do. It can have graphics, text, and a menu of two or more choices. The choice the player makes results in one of the player's traits changing slightly.
- **Outcome Block.** The activity ends with an outcome block, which tells the player of the results of the activity. It can have graphics and text.

Activities are not minigames; the player simply makes a choice and finds out what happens. The activities and their effects are designed to be appropriate for the episode.

The results of activities have random elements in them. For example, a player chooses an activity that affects a certain trait. There is by which the trait is affected. The range can be affected by the current level of the trait. (For example, it may be harder to increase a trait that is already high level as opposed to one that's medium level.)

The purpose of activities is to give the player the ability to choose which traits to work on. It is also a way to have short turns that do not require much if any custom scripting.

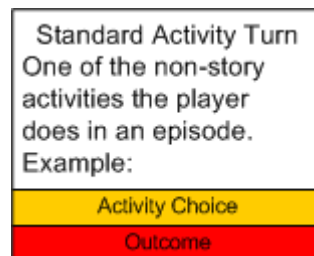
Example:

Player is a college student and an aspiring rock band member. The script assigns a "school day type 1" activity for a turn:

Activity Choice block. Choice are:

- Go to class (effect: +smarts)
- Skip class and practice music (effect: +music, -smarts)
- Skip class and rest (effect: +health, -smarts)

Outcome block lists how any trait has been affected by the choice.



3.1.4. Minigame Turn

There will be a set minigames designed to be appropriate for the game's overall game. Each episode can use these minigames, and the episode's script schedules minigame turns for the play of these games.

The turn is organized in three blocks:

- **Introduction Block.** This block introduces the minigame (setting it in context with the ongoing story) and explains how to play it (if necessary).
- **Minigame Block.** The player plays the minigame
- **Outcome Block.** The outcome of the minigame is announced.

A minigame can affect:

- The ongoing story. Doing well in the game can lead to a good story branch while doing poorly can lead to a different branch. For example, winning a soccer-styled minigame may have the player meet a potential date.
- The player's traits. For example, doing well in a Guitar Hero-like minigame may increase the player's music trait.

See the second concept example below for a minigame turn.



3.1.5. Turn Concept Example: Ethan and Abbie

Background



You're Ethan, a pretty happy guy who's been working out a lot. You've been doing a lot of clubbing, but you recently broke up with Ava. You guess you did something real stupid—maybe you shouldn't have hit on Ava's kid sister at the family party. Now Ava is gone and your friends think you're not so cool anymore.

Also, you find your job boring and have been cutting work, so you're a bit short of money right now.

Ethan	
Happy	😊😊😊
Buff	💪😊💪😊💪😊💪😊💪😊
Love	💔
Cool	😎😎
Career	💵

Saturday Afternoon [Major Event Turn]

At the Gym

You go down to the gym to work out. You see Abbie there—she comes here a lot but you haven't bothered with her while you were going with Ava.

Do you:

- Check out Abbie
- Work on buffing up
- See who else is here

Abbie

You check out Abbie. She looks hot! She sees you checking her out and comes up to talk to you.

Do you:

- Tell her she's and ask if she wants to go make out.
- Excuse yourself and go hit the showers
- Ask how she likes the gym
- Show off on the elliptical trainer

Chatting

You and Abbie chat. She works out at the gym to keep in shape to go clubbing—she loves to dance. Hey, you love clubbing, too! You two seem to have some interests together. Maybe you'd like to ask her out tonight to a dance club. Zanzibar is the coolest today, but it'll cost a lot. Club dSeel doesn't have top bands, but it'd be a lot cheaper.

Do you:

- Ask Abbie to go to Club dSeel



Ask Abbie to go to Zanzibar

Go to work to earn \$\$\$ - you can ask Abbie out another time
Ask Abbie if she'd like to go dancing at your place

Going Dancing!

You ask Abbie if she'd like to go to Zanzibar with you. Yes!!! It's her favorite club. You can see she's impressed with your cool style. You'll meet there this evening.

You didn't get much of a workout today, but who cares?

You get:

+



-



Ethan	
Happy	  
Buff	   
Love	
Cool	  
Career	

3.1.6. Turn Concept Example: Sue

Background

You're Sue, an on-the-go college girl. You get pretty good grades—not the best but you can do well when you try. You sing lead for the Flawed Haloes, a local alt rock band that plays gigs in the area. The band's been doing well. Sarah, who plays the keyboard, thinks a talent agent may check you out soon. Maybe, but she's only been wrong about this the last three times.

You're under a lot of stress right now, as your parents want you to quit the band and focus on academics. "We're paying a lot to send you to this fancy university," your dad says, "and money doesn't grow on trees, you know."

You hang out with the guys when you have time but you're not going with anyone special right now.

Sue	
Happy	
Smarts	  
Love	



Cool	
Music	

Friday Morning [Major Event Turn]

Rise and Shine

Tonight's the big night. The Haloes are playing at Zanzibar for the first time—that club always has good bands and pays well, too.

Do you:

Practice your music

Go to classes

Do yoga—you need to find inner peace

Call home and ask for money

Climate Change 301

You've had your first class, but it's been hard to concentrate. Climate Change is always interesting, though, and you take it with Sarah. You hardly hear the professor. Sarah talks to you after class. "Sue, we really need to rehearse for tonight. The band is going to cut classes this afternoon and practice."

You say, "But Sarah I've got a big Statistical Analysis test this afternoon. I can't make it."

"You would if you cared for the band," says Sarah. She's such a boss.

Do you:

Skip the test and go rehearse

Take the test—the band will be fine

Get mad at Sarah and quit the band

Cutting Class

You decide to skip the test and go rehearse. Always more stress. Hope there'll be a makeup!

Friday Afternoon [Minigame Turn]

Rehearsal

[The player views instructions for a Guitar Hero-like minigame.]

Let's Rock!

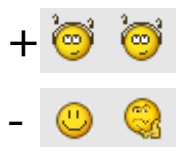
[The player then plays a music minigame and does excellent.]

Awesome

This was your best rehearsal ever. Sarah was awesome with her new moves she'd been perfecting. You are RED-D!

Sarah says, "My brother's friend's uncle has a garage studio. He'll be at Zanzibar tonight. If he likes what he hears, he'll help up cut a demo track."

You get:



Sue	
Happy	
Smarts	 
Love	
Cool	  
Music	     

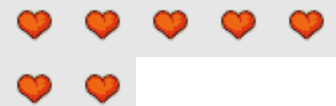
4. Character Traits

The player character in each episode has a set of **traits**. Each trait measures the ability of the character in some manner. For example, happiness would measure the character's overall happiness while smarts would measure how well the character is doing in college. Traits do not have to be mental or personal; for example money could measure how much money the character has, while morale could measure the overall morale of an organization the character is in. ("Traits" is used as a collective design term in this document. In the game, the players will just see the character's traits listed directly, without the term itself needing to be used.)

Each episode sets its own number of traits, their names, and their icon graphics. A character should have at least 3 traits and no more than about 10-12, with 5-7 perhaps being ideal. This allows the players to develop their characters' abilities over the course of an episode.

Each trait can range from 0 (minimum) to 10 (maximum) and can have fractional values (e.g., 5.75). In the game, traits are graphically displayed with icons in a character listing. There are icons for the 0 value, a half value, and a whole value. Multiple icons, in groups of five for easy recognition, are displayed for values above 1. Decimals are rounded off to the nearest .5 when calculation how many icons to display (e.g., 5.75 = six icons).

Examples:

Trait	0 Value	1/2 Value	1 Value	3.5 Displayed	7 Displayed
Love					
Happiness					

The values change over the course of an episode, as determined by the script and player actions. In general, a player should not be able to max out all the character's traits over the course of an episode. (Indeed, in some episodes, some traits are likely to be in inverse relationships with one another. For example, maximizing career success might be done at the expense of love, as the character is almost always at work.) This creates a dynamic tension in the game, as the player chooses which traits to increase.

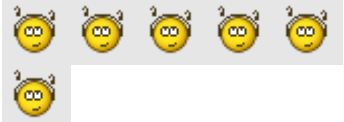
When traits change in a turn, at the end of the turn the player sees a summary of the changes and the character's current trait listing.

Example

You get:

$$\begin{array}{r}
 + \quad \begin{array}{c} \img alt="Smiling face with sunglasses icon" data-bbox="175 745 215 770"/> \img alt="Smiling face with sunglasses icon" data-bbox="215 745 255 770"/> \\
 - \quad \begin{array}{c} \img alt="Happy face icon" data-bbox="175 785 215 810"/> \img alt="Thinking face icon" data-bbox="215 785 255 810"/>
\end{array}
\end{array}$$

Sue	
Happy	
Smarts	 

Love	
Cool	
Music	

5. Episode Scripts

Each episode has its own script. A script has a **script schedule** and several **story lines**.

5.1. Script Schedule

The script schedule controls how the episode unfolds. Each turn in the episode is scheduled, as is the type of turn.

Crazy Night Out Script Schedule Turn & Turn Type Scheduling									
						Story Turn	Major Event Turn	Activity Turn	Minigame Turn
Week 1						Week 2...			
Monday	Tuesday	Wednesday	Thursday	Friday	Weekend	Monday	Tuesday	Wednesday	Thursday
Morning	Morning	Morning	Morning	Morning	Morning	Morning	Morning	Morning	Morning
Lunch	Lunch	Lunch	Lunch	Lunch	Lunch	Lunch	Lunch	Lunch	Lunch
Afternoon	Afternoon	Afternoon	Afternoon	Afternoon	Afternoon	Afternoon	Afternoon	Afternoon	Afternoon
Dinner	Dinner	Dinner	Dinner	Dinner	Dinner	Dinner	Dinner	Dinner	Dinner
Evening	Evening	Evening	Evening	Evening	Evening	Evening	Evening	Evening	Evening
Late Evening	Late Evening	Late Evening	Late Evening	Late Evening	Late Evening	Late Evening	Late Evening	Late Evening	Late Evening
All Nighter	All Nighter	All Nighter	All Nighter	All Nighter	All Nighter	All Nighter	All Nighter	All Nighter	All Nighter

The consequence of the chosen schedule is that when a major event is scheduled, every story line must have a major event for that turn. This is a worthwhile feature, as it establishes pacing for story lines.

Other types of schedules are possible, but this one is powerful yet simple to create content for. There are some major alternatives:

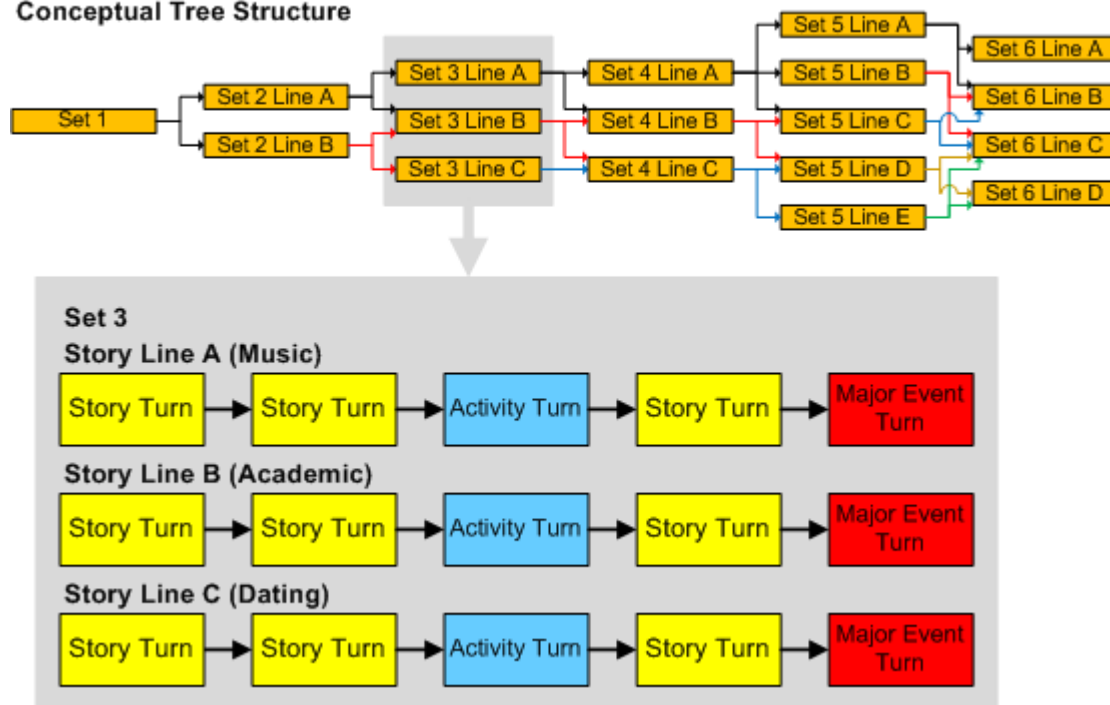
- 1) The turns are scheduled as above, but the type of turn depends on the story line in force. For example, Week 1 Tuesday's late evening turn may be a minigame turn in one story line but an activity turn in another. This, however, seems to make creating scripts much more complicated with little practical effect on game play.
- 2) Turns are scheduled based on the story line in force. For example, Week 1 Tuesday may have a late evening turn in one story line but not in another. This again seems to make creating scripts much more complicated with little practical effect on game play.
- 3) Both 1) and 2) could be combined, but this would make story creation extremely complex.

The episode content creator is responsible for scheduling when the turns occur and what type of turns they are. A CMS tool is planned to assist in episode creation.

5.2. Story Lines

Crazy Night Out Story Lines

Conceptual Tree Structure



The stories of an episode unfold in **story lines** that are organized in sets of turn and subsets of individual story lines. Each episode starts with a single story line but branches out based on the player's decisions and traits. For example, a player who plays a college student and concentrates on dating over school work will get more romance story lines and fewer academic ones.

The story lines can be visualized as a branching and merging tree structure. A key concept is that some story lines merge over time. The goal is to keep the amount of content within reasonable bounds while being able to tell several different, interesting stories with a range of endings.

Story lines are organized in sets of turns. Per the script schedule (see above), each story line in a set will have the same number of turn and same types of turns. The episode content creator is responsible for designing and creating the story content for the turns in the sets. A CMS tool is planned to assist in episode creation.

A set must have at least one turn and can have several. The episode content creator determines the number of turns in a set. There is no requirement that a set start or end with a particular type of turn (although for story telling purposes some will likely end in major event turns.)

5.3. Milestones

The story lines will have various **milestones**, which are significant happenings in the story. Milestones are revealed in the story lines as they occur and are recorded in the player's episode **journal**. Milestones are used to help the story lines branch and merge.

Example: Ethan, played by the player, has been interested in Abbie and has been flirting with her for some time. Abbie's liking for Ethan has been increasing and she finally agrees to date him. The milestone **Ethan and Abbie are dating** occurs.

5.4. Script Logic

The episode content creator is responsible for designing how story lines transition (branch and merge) between sets. Whenever there is more than once choice for a transition, there must be a way for the creator to specify the conditions for the transition. Two factors can affect transitions:

- Milestones. A transition may depend upon one or more milestones have occurred.
- Traits. A transition may depend upon the character's traits being in some specified range.

For example, a story line may transition to three stories lines:

- 1 Ethan and Abbie are dating (milestone) and Ethan's career trait is equal to or greater than 7. (Ethan has a lot of money and the triggered story line will see Ethan taking Abbie to Aruba for the weekend.)
- 2 Ethan and Abbie are dating (milestone) and Ethan's career trait is less than 7. (Ethan and Abbie go skating in Central Park.)
- 3 Neither of the above is triggered. (Ethan plays videogames all weekend.)

A simple scripting language should be created to allow the episode content creator to specify the transitions. We should also explore how difficult it would be to have automatic error checking, so that major errors are caught. (For example, the creator created a transition in which one line was triggered if Ethan and Abbie are dating and Ethan's career is 7+ and another line was triggered if Ethan and Abbie are not dating. The transition does not handle the possibility that Ethan and Abbie are dating but Ethan's career is less than 7.)

6. Other Major Game Features

6.1. People

Anytime during play except in minigames, the player can use the **people** feature to see profiles of the various significant characters the player character has met in the game. This allows the player to check out characters of interest for romance, career advancement, etc. There should be a large, interesting graphic for each character in their profile.

As the relationship of the player character with the other characters change, the characters' profiles change accordingly (a writing task of the episode content creator).

We might consider calling this feature My Spacebook or something that suggests a social networking site.

The player can also use this feature as a memory aid. For example, if the player hasn't played for a few weeks for some reason, checking out people will let the player review the current relationships.

6.2. Journal

As milestones occur in an episode, they are recorded in the player's **journal**. The player can use this feature anytime during play except in minigames as a memory aid.

6.3. My Character

Anytime during play except in minigames, the player can use the **my character** feature to see the player character's set of traits.

6.4. Popularity

The game can track the **popularity** of certain items in an episode. While there are no direct interactions between players, popularity can give a subtle reality-show element to the game.

For items being tracked, the game aggregates the player choices to see how popular they are. The results are used when other players subsequently play the game. For example, a player may have a choice of three clubs to go to on a date. A very popular club is expensive, a less popular one is not as expensive, and the unpopular one is cheap. However, the player's happiness or the satisfaction of the player's date depends on going to the popular one.

The game establishes default values for these items (such as one more popular or all three starting equal) but as players actually choose them their rankings can change.

To handle this, items that are tracked by popularity need to be templated, so that the episode content creator can write text about them like "<MostPopClub> is hot! Let's go there, not to <LeastPopClub>.", which might resolve to "Zanzibar is hot! Let's go there, not to Sonoma."

6.5. Replayability

Any episode can be replayed. Player will be encouraged to replay, as different choices will result in different episode outcomes.

7. Minigames

7.1. Overview

Crazy Night Out will have a set of minigames that are used in common across all episodes. Since the title and theme of Crazy Night Out are still under consideration, this document presents some minigame concepts with dating, relationship, music, and fun sports themes that may be appropriate to the final game. These will need to be reviewed when the title and theme of Crazy Night Out is finalized.

All minigames will be fairly simple and quick to play. Some will be extremely simple while others will be more interesting.

Each game will have a scoring system. At the end of the game, the player's score affects the play of the episode. The episode content creator will have a simple scripting language that allows scores to change **milestones** or the values of **traits** (like in script logic as discussed above). For example, if Sue does well in a music minigame, her music trait is increased. If she does just OK, there's no change. A poor performance reduces her music trait.

7.1.1. Minigame Selector

We should seriously consider having a richer set of minigames for higher end phones than for lower end phones. Given user expectation for good phones and given the likely competition Crazy Night Out will face, it seems inappropriate to have the lowest-common denominator phones determine the minigame lineup for all phones. This is especially the case if Crazy Night Out will be developed for low-memory Java phones.

To this end, Crazy Night Out should have two sets of minigames: minigames for all phones (driven by the lowest-common denominator phones) and minigames for higher end phones. When episode content creators schedule a higher-end minigame in a turn, they also specify the lower-end minigame that will be used instead on lower-end phones.

If this scheme is adopted, it will be decided during development which category the minigames are assigned to.

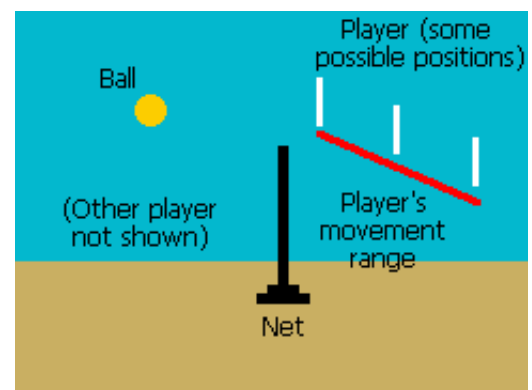
7.2. Minigames

7.2.1. Beach Volleyball

The beach theme is established with the background graphics for the minigame. The game is not a volleyball simulation but just a very simple game with a volleyball look. The court is divided in half by the net and the volleyballer is on one side. (Whether we show an actual volleyball character or just a symbol of some sort is TBD). The player uses the left and right keys to move in those directions.

The player must move the volleyballer to intercept the volleyball when it comes over the net. Success means the volleyball is hit back; failure means the volleyball lands and the other team scores a point. If the volleyballer intercepts a volleyball very near the net, a "Spike!" graphics appears and the ball is spiked.

A spike always scores a point. If the player hits back the ball without spiking it, there's a cumulative 1/3 chance of scoring a point (1st return 1/3 chance, 2nd return 2/3 chance, 3rd return



automatically scores). The opponent always returns the ball until the player scores a point or misses.

The team that scores 5 points first wins.

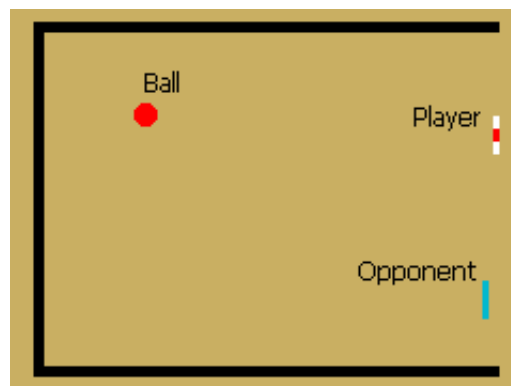
7.2.2. Racquetball

Again the theme is established with the background graphics. The court resembles a racquetball court and the player's racquetball character is on one side. (Whether we show an actual character or just a racquet symbol is TBD). The player uses the up and down keys to move in those directions.

The player must move to intercept the racquetball when it is the player's turn to hit it. (The ball changes color or otherwise indicates when it is the player's turn to hit.) Success means the ball is hit back; failure means the ball is missed and the other character scores a point. If the player intercepts a ball in a small "sweet spot", an "Ace!" graphics appears and the ball is hit so that the opponent cannot return it.

An ace always scores a point. If the player hits back the ball without aching it, there's a cumulative 1/3 chance of scoring a point (1st return 1/3 chance, 2nd return 2/3 chance, 3rd return automatically scores). The opponent always returns the ball until the player scores a point or misses.

The character that scores 5 points first wins.



7.2.3. Matchmaking



This is a Concentration-like puzzle minigame with a dating (matchmaking) theme. There's a grid of tiles (say 4x4). The player chooses two tiles in turn. As tiles are chosen, their pictures are revealed. If the two tiles match, they are removed. If they don't match, their pictures are hidden again. The player has a short amount of time, like 30 seconds, to match as many tiles as possible. The number of matches made is used for scoring.

The graphics show the matchmaking theme. Rather than having pairs of identical items as in Concentration, each tile in a pair is half of a couple. Clothing styles, colors, etc., are used so that each couple is distinctive from the other couples.

7.2.4. Spin the Bottle

The player character plays spin the bottle with a bunch of friends. They all decide to play spin the bottle. Someone brought their dog, who plays, too.

There are six faces around in a circle with a bottle in the center of the circle, five people and one friendly-looking dog. If the player is currently involved (dating or whatnot) with another character in the episode being played, that character's face is included. Each face is rated with a + # (+1, +2, +3, +4, +5) for how good looking it is; the dog is -5. The player's love interest, if present, is assigned the +5 slot.

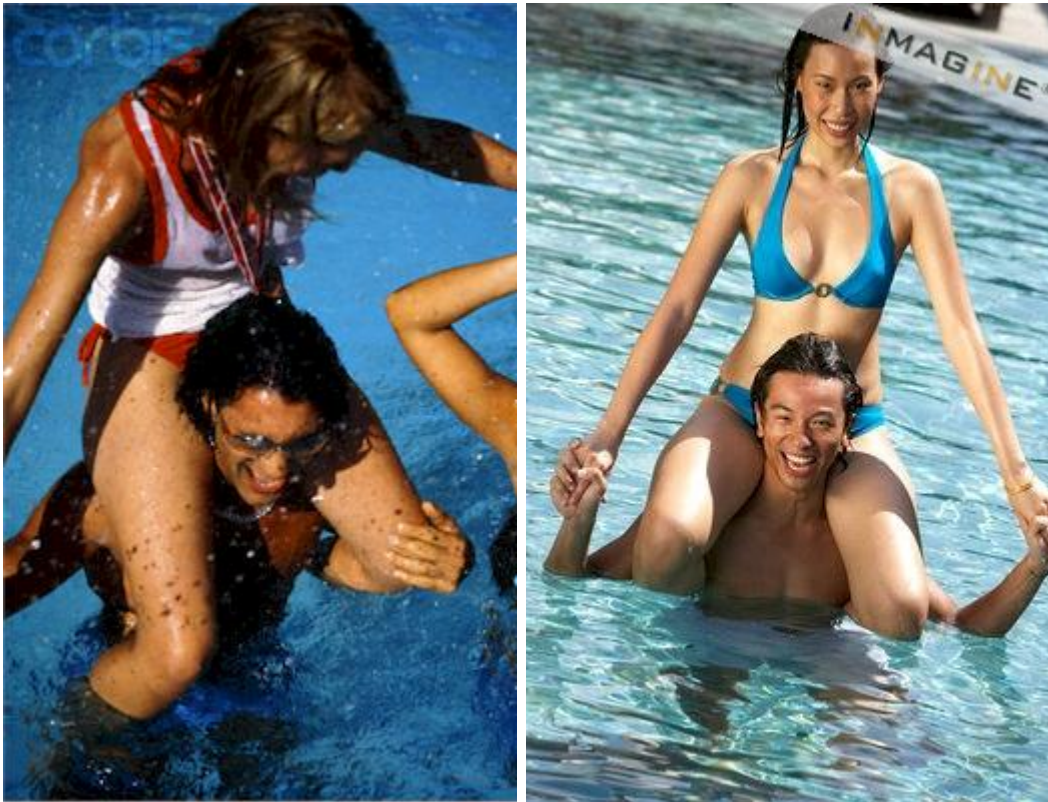


The player pushes OK to spin the bottle. The bottle spins quickly at first and then slows down over time. The player pushes OK again at any time to stop the bottle. The player kisses whoever's face the bottle stops at. (We probably won't show actually kissing but maybe the lips on the chosen face pucker up and a kissing sound plays—or a slurping sound for the dog.)

Then, the player spins the bottle again. Does the player stop the bottle quickly, when it is hard to select a face, or does the player wait for the bottle to slow down and pick the high scoring faces?

The minigame is played for 30 seconds. The goal is to rack up a high kissing score by the end of the time.

7.2.5. Hot Tub Wrestling



Two couples are staging a hot tub water wrestling match. For each couple, the girl is piggyback on the guy's shoulders. A couple wins if they unseat the other couple's girl and dunks her in the water.

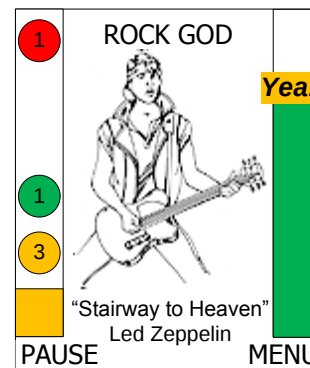
The player controls one couple and a simple AI controls the other. The player presses OK to start the bout. The right and left keys moves the player's couple right and left. The up key has the girl try to pull the other girl off. The down key has the girl try to push the other girl off. The combination of each couple's moves determines what happens.

The player wins the minigame by winning two bouts out of three.

7.2.6. Rock God

The image in the center is the player's character. Game play occurs on the stage screen (see above concept diagram), with these main elements:

- Avatar in the center. Avatar is animated and moves as music plays.
- Notes drop on the left. The "notes" or "chords" to be played drop down in the column on the left. When a note reaches the bottom square, the player needs to press the key at that time. Pressing sooner or later misplays the music, distorting it. Press nearly on time plays it well; pressing perfectly plays it the best.
- Audience meter on the right. This has a bar that pulses up and down, showing the audience's reaction to the player's music playing. Text and sound effects also highlight the audience's reaction, ranging from screams of adoration to boos.



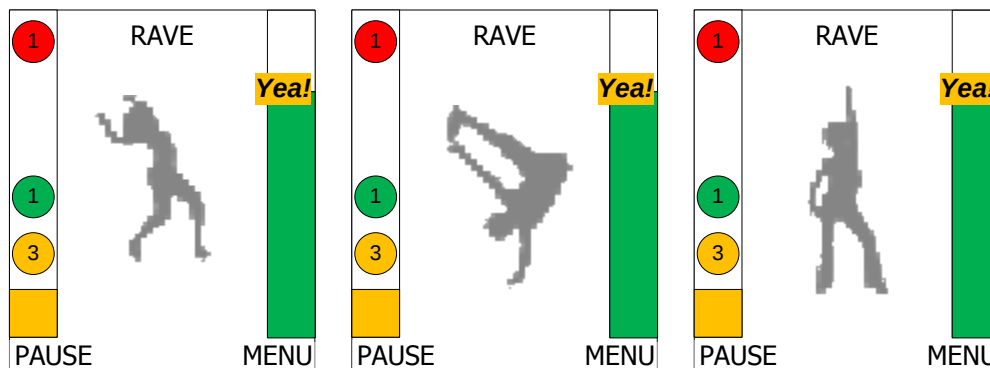
There are four levels of difficulties in the minigame, corresponding to the number of keys in play:

- Easy: Only the 1, 2, 3 keys are used for the notes.
- Medium: The 1, 2, 3, 4, 5, 6, keys are used.
- Hard: The 1, 2, 3, 4, 5, 6, 7, 8, 9 keys are used.
- Very Hard: The 1, 2, 3, 4, 5, 6, 7, 8, 9, *, 0, # keys are used.

The episode content creator assigns the level of difficulty of the minigame when Rock God is scheduled as a minigame turn.

The audience's reaction to the performance determines the player's score in the minigame.

7.2.7. Rave All Night



We reskin Rock God so that it is a dancing minigame rather than a music playing minigame.

7.2.8. Dream Date



The player has to put together their dream date. Several images are broken up into thirds vertically. For each third (upper, middle, lower), one random image is shown at the start of the game. The player starts with the top image having the focus. Pressing OK scrolls the images quickly; pressing OK again selects the image on screen, freezes it, and move the focus down to the next third. The player continues until they have built their dream date. One of the images is a joke image.

The player scores by how well they build a coherent image: 0 for all three thirds being random, 5 points for having two thirds matching, and 10 points for have all thirds matching unless they build the joke image, which scores -5 points.

7.2.9. Make the Scene



The player is given a theme, like "elegant dinner" or "beach party" or "mountain biking" and is then shown a series of objects. The player has a limited time to choose whether the object is

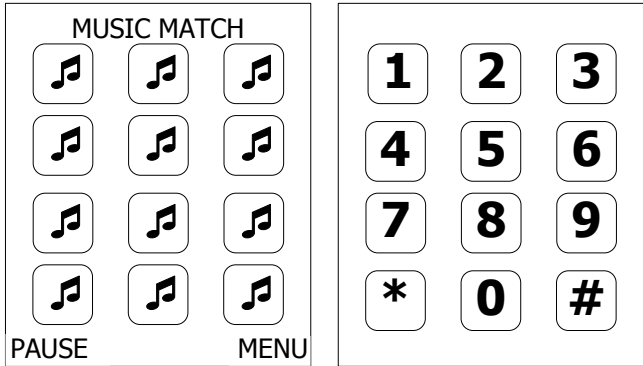
appropriate or not for the theme. At the end of the minigame, a scene is shown using the objects she chose, which could be hilarious like a beach umbrella at an elegant dinner. The player scores points by choosing well.

7.2.10. Music Match

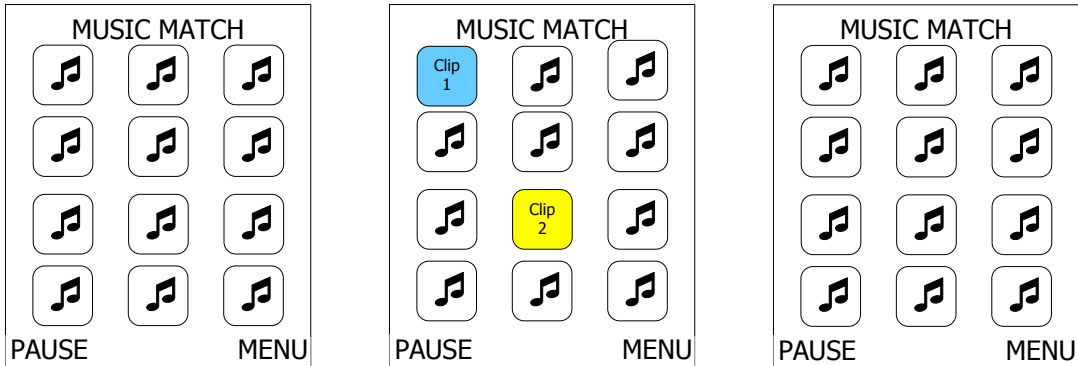
This is another Concentration-style game: choose an item, chose another item, if the two match they are cleared from the game. If they don’t match, keep trying. The goal is to clear all items within a set amount of time.

Instead of the player matching pictures, they match short music clips. For example a 3x4 grid would have six different clips, each appearing twice. Selecting a tile would play the audio. It would also show part of a scrambled picture. When the grid is cleared, the picture unscrambles as an extra visual award. For example, a game could have six clips from an artist, with the picture being the artist’s picture.

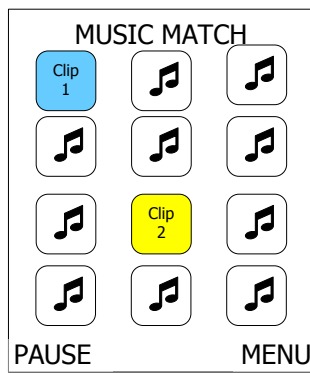
The number of matches made is used for scoring.
For d-pad phones, the UI maps one key to each tile, to allow for easy, rapid play.



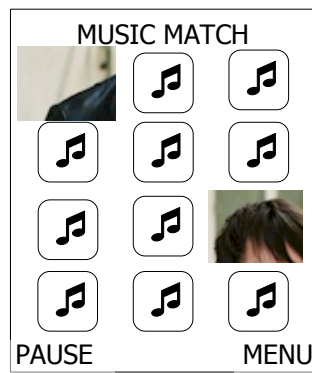
Here’s a short play sequence. As a bonus, clearing tiles would reveal a hidden, music-related picture:



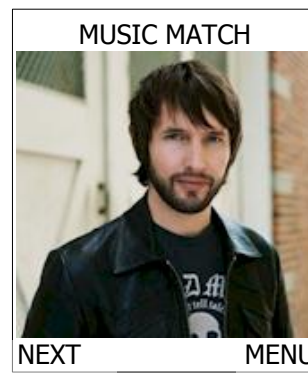
1. Start
2. 1 and 8 tiles chosen
3. Oops, different clips, reset



4. 1 and 9 tiles chosen



5. They match and clear!



6. Keep at it until all are cleared!

8. Notes

The following items will be designed or specified later in the development process of Crazy Night Out:

- **Content Management System.** The game will need a content management system that handles all aspects of each episode: script schedule, story lines, turn content, graphics, etc.
- **UI and Functional Specification.** This will be done once the design, title, subject of the game are green lighted.