# Recollections of a Novice Javascript Programmer

by Brian Schick. In submission for the Game Engineer position at Weeby.co

<https://github.com/bmschick/BubbleGum.git>

## Summary

I did manage to get far further along in the GameClosure project than I had anticipated. And yet, I was unable to complete the game in seven days. This was my first HTML5 / javascript game I started from scratch. I worried that I would get lost among the documentation just after ‘Hello World’, and I did run into problems installing the devkit and in running some of the examples. Much of my time was spent reading documentation on the GameClosure devkit, reading javascript documentation, and working through bugs introduced by transferring code from the examples into my growing bubble shooter game. It was challenging, and ultimately rewarding.

After eighteen hours of work over seven days, I had a bubble gun game skeleton that ran on the iPhone emulator. It had a menu screen, a gameplay screen, and a scoreboard screen, linked together by buttons. The gameplay screen had a hexagonal grid for the bubbles, a bubble gun, and a score board. And I could launch bubbles into the grid by clicking on the grid. I had placeholder art. But I had in no way a game. Gameplay was missing. The scoreboard file was missing. The art was raw placeholder art. And there was no sound at all. And I had a lot of documentation.

I scheduled my time, and stuck to the schedule up to putting the devkit to use. I designed my game. I linked my game design to the specific ui components available in the kit. And I recorded my progress. The journaling and designing helped guide my development.

This is my second solo attempt to make an app for a mobile. With the GameClosure devkit, I got a lot farther, to the point where I am confident I could finish and polish my game. Next time I attempt to create a mobile game, I’ll reach for the devkit first.

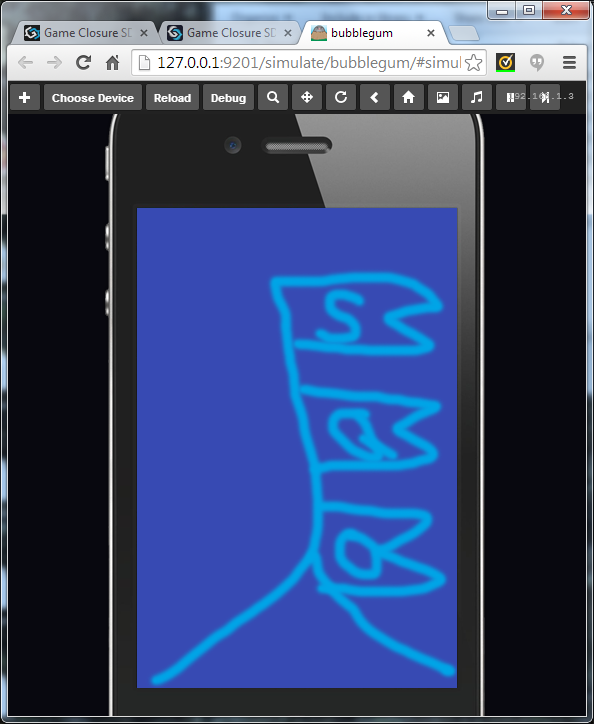
## The Devkit

GameClosure’s devkit allowed me to get a lot further into mobile game programming than any of my earlier attempts. The Quick Start Guides helped me get set up rapidly. I had the examples up and running in a day. I did run into the common problem of a broken install for basil that occurs if the git bash for windows is not run as administrator. But I found the fix among the forums, and managed to get ‘Hello World’ working. The Quick Start Hello World guide walked me through Chrome’s useful javascript console. I made extensive use of the console exploring my run-time app. But at the bottom of the ‘Hello World’ guide, the ‘basil install examples’ failed. It seems there were file changes in the devkit/addons/examples directory that needed to be committed before the install succeeded. Then I had all the examples to play with and examine. The Game Awaits walkthrough was fairly straightforward and informative, although the code installed by basil has been slightly refactored since authoring the document.

But Whack-the-mole did not demonstrate every feature I wanted for my game. And so I turned to the less friendly Examples. I needed to input text for my high scores. The Examples have a TextEditView, but the Launch Demo button just shows a large black frame. I could copy the example code over my Hello-World to get some of the examples to run, but the TextEdit did not transfer the text from the TextEditView to the TextView, and I could not squash that bug. I wish the Example write-ups had the same detailed and story-like manner as the Quick-Start guides.

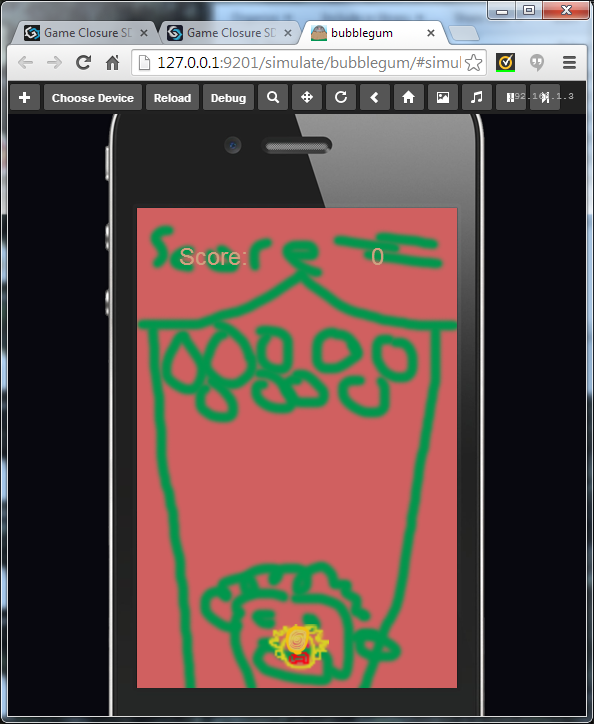
## My Game

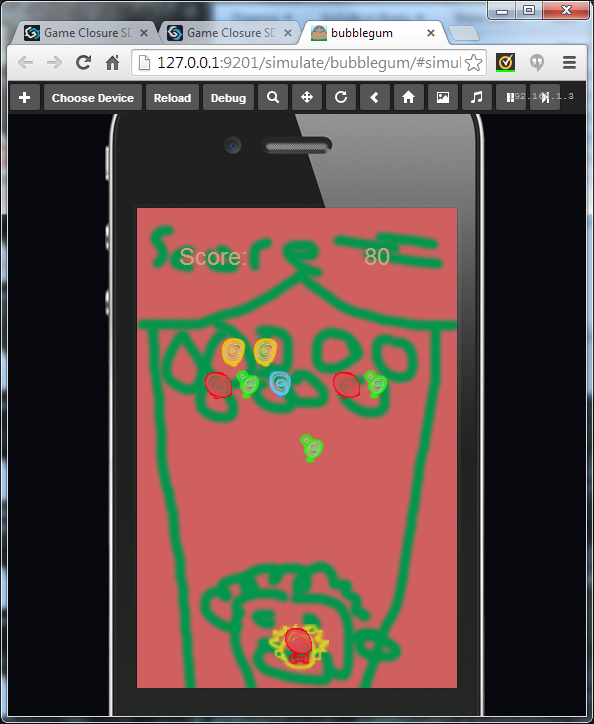
The menu screen

My game starts with a view of the top of the big top tent at the circus. There are three flags: ‘Start’, ‘Quit’, and ‘Scores’. Clicking on a flag brings the next view.

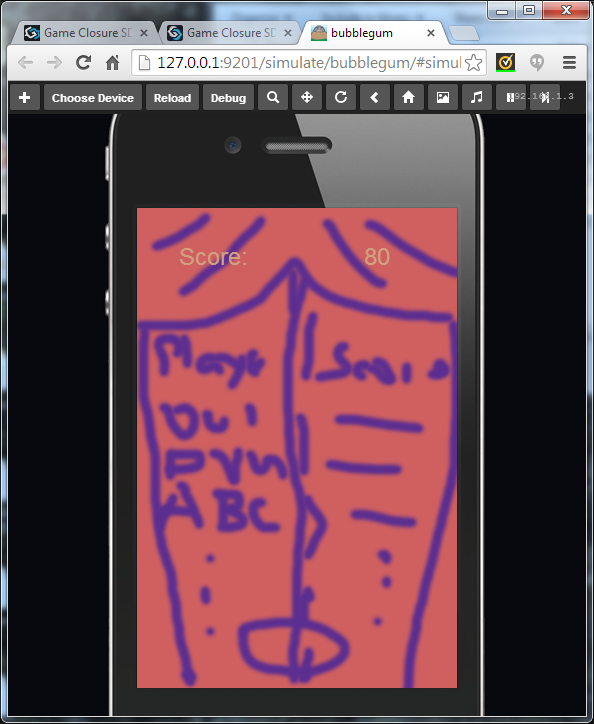
I got placeholder art working, and working buttons near the flags. A fairly straightforward application of Whack-that-mole’s titlescreen.js.

The game screen

The gamescreen shows the inside of the big top. There is a scary clown at the bottom chewing bubble gum. Eventually the clown blows a bubble, and is ready to spit his bubble onto the ceiling of the big top.

When finished, the player will touch the screen in the middle of the tent. A cross-hair will appear where they touch, and a line stretching from the clown to the first available space for a bubble on the ceiling.

I made a hexagonal grid for the ceiling bubbles out of a gridview and an array of imageviews. The bubble is animated and flies up to a clicked cell. And then the clown reloads another bubble in one of four colors. Each time a cell is bubbled, the score increases by 10. This was similar to the checkerboard of moles in Whack-that-mole.

There’s a secret button in the lower left corner of the game view, that ends the game and moves on to the score view.

The Scoreview

The score view shos the outside of the big top, after the curtains have closed. At the top is the score of the just finished game, and down the curtains are the top 10 high scores. If the player managed to fit into a high score, she’d be able to enter her initials to record her score for posterity.

I transferred the score from the gameview to the scoreview. The scoreview also has a button bottom center to move to the main menu when pressed.

# Schedule

X: finished. O: not finished.

I had a good start with my schedule, and kept to it for three days. Once faced with the steep technical learning curve around Friday, my schedule kept slipping.

* Today – Design method/admin goals.
  + X by 1 pm. First step – brainstorm admin goals.
  + X by 1:30 pm. Put hourly times on the goals.
* Today – X 1:30-2 – Admin work for DeVry.
* Today – X Prior commitment. 2-4.
* Today – 6/8X 4 – 7. X Download, install, run engine. X Run a tutorial/example. O Look at code. O Make a small program that runs. At http://docs.gameclosure.com/guide/hello-world.html.
* Today O 8 – 9 – Prior commitment.
* Today – X by 9:30. Choose game genre. Leaning toward bubble gun one. Sounds easy.
* Today – X 9:30-10. Adjust Schedule.
* Thur 1-7 –O Prior commitment
* Thur 1-7 – X Prior commitment.
* Thur 11-1 – X Look at code O Make a small program that runs. At http://docs.gameclosure.com/guide/hello-world.html.
* Thur –X Design my game. Keep it simple!!!
  + 8-9:30 X Brainstorm game ideas / design.
  + 8:30 – 10 X Check ideas. Is it too complicated? If so, simplify.
  + Author Game Design Document.
* Thur. – O Design my program.
  + 10-11:30 O Brainstorm design. Keep it simple!!!!!
  + 10:30 – 12 O Test ideas. (How?) Is it too complicated? If so, simplify.
  + Author Technical Design Document.
* Thur. – 9:30-10 OAdjust schedule
* Fri. 10-10:30 X Draw views on paper
* 10:30-11:30 O Brainstorm design. Keep it simple!!!!!
* Fri. 10:30 – 12 O Test ideas. (How?) Is it too complicated? If so, simplify.
* Fri. 12-2 O List api needs. List documentation.
* Fri. Author Technical Design Document.
* Fri. – 12-4:30 O Gather art and sound assets.
* Fri – 4:30-5 O Adjust schedule.
* Sat – 7-7:15. X Adjust schedule.
* Sat 7:15-8 O Brainstorm GameView play. Keep it simple!!!!!
* Sat. 8-8:30 O List api needs. List documentation.
* Sat. 8:30 – 12 O Test api ideas. By running api examples in the Hello-World demo.
* Sat 12-12:30 Adjust Schedule
* Sun. – 12-8 O Develop my game.
* Sun. 8-10 O Gather art and sound assets.
* Sun – 10-10:30 O Adjust schedule.
* Mon – 10-3 – Senior Project in class
* Mon. – 3 – 9:30. Debug and play test my game. (Find people?)
* Mon. 9:30-10 Adjust schedule. (Really? Panic time!)
* Tue – 10 – 10. Last minute work.
* Tue – 10 – 10. Write recollection.

# Technical Design

### Views

#### GameView Layout

* background view - bigtop
* divided vertically into three regions
  + Top 320 by 96 (top 1/5), score – 2 textviews
    - ‘Score:’ – textview, centered on left half.
    - numerical score – textview, centered on right half.
      * Adjusts to score!
  + Bottom 320 by 96 (bottom 1/5), gun animation
    - *Clown* 32 by 32. imageview - centered bottom of screen
    - *Hand* 32 by 32 – imageview – 64 right of clown (ie 32 space between)
  + Middle 320 by 288 (middle 3/5), bubble zone.
    - Middle 256 by 288, gridview *bubble collector*
      * 9rows by 16columns. Arranged in a hexagonal grid as such:
        + odd rows -> even columns merged with odd columns from column 0. All now 32 by 32. Row of 8 cells
        + even rows -> odd columns merged with even columns from column 1. Columns 0 and 17 are 16 by 32. Rest are 32 by 32. Row of 7 full cells, and 2 half cell spacers on ends.

### Descriptions

#### GamveView:Bubble collector

* 256 by 288, divided into hexagonal cells.

## API needs

O – not tested. X – tested passed. **B** – tested, blocked.

* Buttons – X Whack-that-mole Titlescreen.
* Timed events – O Whack-that-mole Gamescreen (tell moles to peek and then hide)
* Text input – **B** TextEditView – **Blocked** texteditview example
* Text input – O TextPromptView
* View Rotation –
* Collision Detection –
* Text display – X TextView – Whack-that-mole Gamescreen.
* Animation –
* View transitions – X – stackview – Whack-that-mole Application.js
* Sound

# Game Design

## Genre:

Bubble Gun.

## Platform

Android. iPhone.

## GamePlay Flow

Start: Enter at MainMenuView

### MainMenu View:

* Top of midway.
  + Flag with ‘Start’ [button]
  + Flag with ‘Quit’ [button]
  + (Flag with HighScores [button]
* Start: onClick(tap)
  + transition to play view
* Quit: onClick(tap)
  + Quit app. How?
* (HighScores: onClick
  + transition to GameOverView, with score of N/A)

### Play View:

* Inside open midway
* ClownFace(gun) (scary cartoon clown face) in middle bottom. Clown hand to right (white cartoon glove). Colored bubble gum in hand. (Little [wrapped] colored says ‘Gum’ on wrapper.
* Timedown.
* hand gum to mouth. animation (unwrap?)
* timedown
* chew animation. color gum in mouth. (Spittle dribble. etc.)
* timedown
* blow bubble animation. colored bubble (transparent if possible)
* look up animation. Clown face and bubble up.
* aim:
  + mouse over (drag finger over)
  + see cross hair at finger
  + see line from mouth to cross hair
  + see ghost bubble at intersect point
  + (extra: clown head tracks aim)
* fire:
  + click (tap) to fire.
  + Clown(gun) spits(shoots) bubble
  + Clown not move until bubble stops.
  + Line freeze / blink. No longer tracks mouse(finger drag)
  + Bubble animate: fly along line
  + Fly to collision.
  + Collision detect (stops at blocked hexagonal grid from ceiling of midway, or hanging bubble.)
  + Snaps to nearest available cell once blocked.
* Check for pop:
  + If stopped bubble creates 3 or more (could be up to 7) same colored bubble group touching, all pop, emptying cells.
  + If orphaned cells (no bubble trace to ceiling), all orphaned cells (individually) drop animation (this could be a performance bottleneck)
* Check for collapse
  + If no pop, if bubble trace ground to ceiling, (also bottleneck?)
    - Animate midway collapse
    - Animate fear/angry clown face
    - Show ‘game over’
    - transition to GameOverView with score of score

### GameOverView

* Show closed door of midway
* Show score (and leaderboard)
* Show ‘Done’ button.
* (Show leader board Score: Initials pairs.
* (If top 10,
  + highlight score and ‘…’ for initials
  + allow enter initials
    - if tap on initials, highlight all 3 initials, start with first initial
    - if initials highlighted, on keystroke, fill in initial, one per keystroke to 3.
    - if initials highlighted, on backspace, set last filled initial to ‘.’
  + on click (tap) Done button, save initials in top 10 slot.)
* On click (tap) Done button, transition to MainMenuView

## Scoring

* ColorChainScore: Each color bubble worth the same (N^2-N)\*10, where N is number of bubbles in popping color chain.
* If M orphaned bubbles dropped, add to score M \* ColorChainScore.
* Final formula (M+1)(N)(N-1): M-orphaned bubbles popped. N-color chain bubbles popped

## Balancing

* Increase min chain to 4?
* Final formula (M+10)(N)(N-1)\*10?

## Art Assets and Animation

* Clown Face
  + Chewing animation
  + LookUp
  + Fear/Angry
* Clown Hand
  + Unwrap Gum
  + Hold Gum (Thumb to forefinger)
* Clown Arm(s)
  + Open Midway animation
* Bubble Gum
  + Wrapped
  + Unwrapped
  + Chewed
* Bubble Gum Wrapper
* Bubble
  + Blow animation
  + Pop animation
  + (Collision animation)
* Midway top
* Flag(s)
* Midway sides
* Midway Curtain
  + Open midway animation
  + Close/Crash animation
* Midway ceiling

## Sound assets

* Midway background music (John Phillips Souza)
* Wet Chewing sound
* Blowing sound
* Wet Smooshing(Collision) sound
* Wrapper sound
* Wet Popping sound
* Crash sound

# Raw Notes

## Day 1

5/7 –How to start this?

Make a schedule-

* Test the engine’s capabilities.
* Design a game.
* Build the game.
* Write the recollection.

Schedule day 1. Behind with stat class and looking at demo code. A lot accomplished.

Time – 7 days.

Previously commited (out)

* Mon,Thu work
* Fri Sat play
* Sun work.

Game design prelim

* + Goal: Bubble gum game – Clown chewing bubblegum spits on ceiling for points.
  + Framework
  + Mechanics
    - Input
    - Gameplay
  + Graphics
  + Animation?
  + Sound?

### Questions?

* X Need to put hourly times on the goals.
* How to download and install game engine?
* Where’s your testing/learning time?
* Design – Graphics and Sound. Make or find.
* Design – Animation? A bubble gum game would be fun.
* I need breaks too. Breaks or no breaks?
* I need to keep time for my prior commitments.
* I need time to skim the HTML5/javascript games textbook.
* Need recollection time at end of each day to adjust schedule.

12:21 Running out of ideas/questions. Move on?

16:12 – Installation is apple/linux. There is a Windows instruction list…

msysgit? Will that overwrite my git? One easy way to find out…Looks familiar…I’m running 1.9.0 msysgit.0. Instructions want >1.7.10. How big a difference between the old and new? I think I have this already…Yes.

node.js… Node looks like a server?... Installing… 6MB. Good.

npm gives me an unsecure connection. Trust?... Weird. Went to the home page of the site. And now the link works fine. A github repository. js wrappers for command line commands for windows. How to install? Installed by command line ‘npm command’. What’s npm? How do I access it? Installed by node.js! Accessible from command line! Good!

Try ‘npm install windows’… Mostly working. One warning: Current engine node is v0.10.28. Wanted v0.6. Better versioin works.

Finish the full install and get to Play with Examples before break.

JDK. 8u5 is the latest. What do I have?... Download and find out… Windows x64… I have 7u51. 0.7Gb on my C:.. At least I get some installation breaks. Compiling!!! Everything is slow. Is it still installing? 24%... Done.

Set the PATH. Already done? How to check? Can’t find Java? Added to path, but cannot see with ‘set’.

Downloading devkit. Done. Installing with install.sh… Lots of demos downloading… npm downloading. Errors.

5:49 pm Moving ahead despite errors…Nope! basil does not run. Fix it!!!

18:36 **Blocking errors!!!!** sdk dir missing after install. Debugging:

* Try download and install again. Oh. Need a git bash terminal. Was that the error? Same errors.
* In repository? No. Probably created by install.sh.
* In Readme.md in repository, shows what’s supposed to be in the sdk after creation. Have links to the library files. Set the links myself? How?
* Reading a error list on github\gameclosure\issues. <https://github.com/gameclosure/devkit/issues/18> (accessed 5/7/2014) Try running GitBash as admin… Works!!!

19:39. Yep, that did it. Running demos… I am at break time. How much time left? None. I’m 39 minutes over schedule. Did not get to make a small app for myself.

How do I shut the basil server off? Killed something from taskmanager that did the trick.

### First install try errors:

**Fixed** with <https://github.com/gameclosure/devkit/issues/18>. (accessed 5/7/2014).

## Day 2

Schedule day 2, worse. Behind with technical. Ok with demos and game design. Game design fits demos. Good.

11:23 (1 hour behind already) Settings icon on Google Chrome? Where? Chrome developer tools from Google? Bundled with Google Chrome. Found at Customize -> Tools -> Javascript Console.

11:57 In the Whack that Mole walkthrough.

12:08 How do I author a project?

12:25 The titlescreen.js for the Whack that mole is different from the one in the walkthrough! Am I looking at the right one? This is the one in the devkit/projects. Same thing, just refactored from the one in the walkthrough.

Hello-world is under devkit, but Whack-that-mole is under projects. Did I put hello world in the wrong place?

12:39 Getting kinda lost. I’d like an IDE to manage the projects. basil? There was a manifest file…

12:44 I need more time for the walkthrough.

Some panic.

1:07 At the MoleHill.js in <http://docs.gameclosure.com/guide/game-walkthrough.html>

22:26 Played hookie for 3 hrs. Now am about a ½ day behind schedule. Ok? Design game brainstorm.

22:39.. Brainstorming. Fun. Should be simple enough to manage to develop.

### Questions

* How does animation work? Transitions between images? Instant? Fade between? How?
* Multiple(>2) images in animation? How work?
* How get alpha? (See whack a mole)
* How get game loop? Do I need one? Flying transition? (See alien shooter bullets)
* How get animation loop? (See Flappy bird?)
* Am I putting off the steep technical learning curve through design?
* Technical needs a feature schedule. How to tackle?
* What features to tackle first?
* How steep is the technical learning curve?

## Day 3

### Notes

* Off schedule!
* 20:02 Got the views exchanging via buttons throwing events. GC.app catches and handles all events.
* 20:43 How do I input text?
* 20:43 Examples don’t work on my machine. Try putting the text edit view example javascript into your hello world.
* 21:25 That worked. But I don’t know if the subscribe(“change”,… is working.

### Questions

* How do I find the position of a button ui.View? Aren’t they transparent? There’s a view tree in Chrome.
* How do I make a new project? basil init? Where was that? Yep. basil init projname.
* Where’s the man pages? basil init –help is not very verbose.
* What’s the screen resolution? I want to make a fake image to display with three buttons for the skeleton. It’s 321 by 480, according to the Whack-that-Mole image.
* Trying a simple button interface. Doesn’t load. Why?
* How to pass the observing object to a command?

## Day 4

* 18:54 Off schedule. Reschedule.
* 20:26 Good start on gameview design. Identifying game needs with api. Everything is there. Not done.