

Postmortem: Bane Games' Flick Buddies



By Alistair Doulin

[Brisbane-based Bane Games recounts the process of going indie, developing their first iOS title, Flick Buddies -- and learning important lessons about timing.]



After many years working together in mainstream game development, a small group of developers decided it was time to "go indie" and create their own company. [Bane Games](#), based in Brisbane Australia, grew out of this desire to work independently on fun, small games. Our first title, *Flick Buddies*, recently launched on the iOS App Store, and this is the story of its creation.



What Went Right

1. Rapid Development

With only four developers working part-time, we knew we had to set our sights on a realistic target to make sure we would produce our first game in a few months, rather than years as we were used to from our mainstream days.

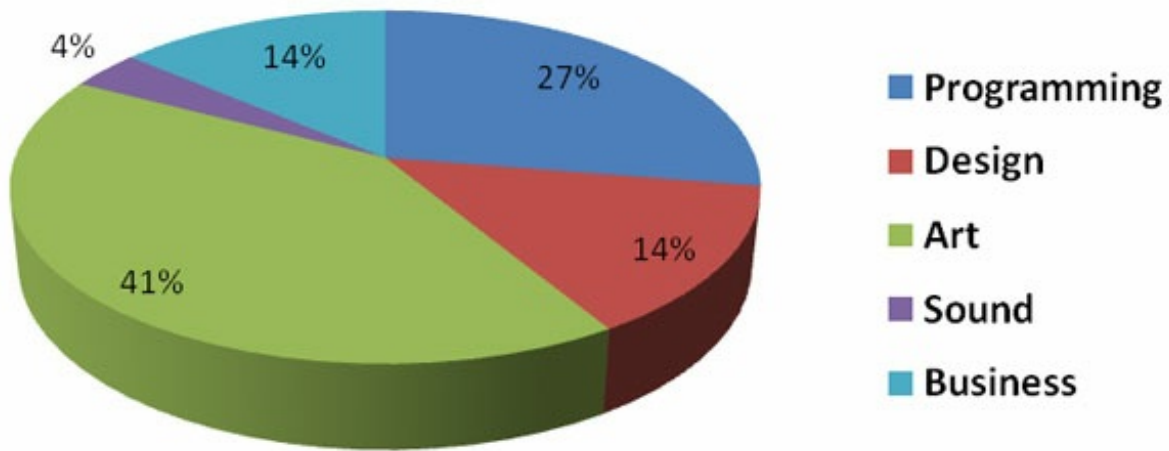
We reached our first playable build in the first few hours of development and had prototypes of all 12 levels complete within the first few days. Much of this rapid development was thanks to the use of the Unity game engine. This freed me (our sole programmer) up to work solely on gameplay rather than engine development. The editor also allowed our game designer Simon to rapidly prototype level ideas and tweak values without ever touching script.

This rapid development helped keep team morale high, allowed us to rapidly iterate over gameplay, and kept costs low. Below are the approximate hours spent on each area of development:

- Programming - ~200 hours
- Design - ~100 hours
- Art - ~300 hours
- Sound - 26 hours
- Business/Marketing - ~100 hours

We spent a lot of time iterating over the gameplay to keep it as fun and tight as possible. For example, we experimented with different game lengths before deciding 60 second games delivered just the right amount of fun to keep players wanting more. We also spent a lot of time on the flick mechanic to make sure it felt as real and tactile as possible.

Development Time Breakdown



2. The Team

All the members of the team had worked together previously on at least one shipped title. This meant that we all knew each other's strengths and weaknesses going into the project. We'd also all worked under the watchful eye of publishers before, which accentuated the freedom we were now experiencing as we set our own deadlines and had complete freedom in design.



From left to right: Shauno (art), Alistair (code), Simon (design), Mick (audio)

We created the smallest possible team capable of creating a high quality game in a small period of time. We had seen how much money a large team could suck out of a company, and we decided on four core developers. This

ended up being the perfect size, filling all the requirements of developing an iOS game. Any smaller would lead to holes in skills and leave the game looking unprofessional. Any larger and the fact we were working remotely would put too much strain on the team.

3. Release Early, Release Often

We subscribe to the belief that it's best to release our games as early as they are ready, so we can get feedback from our players to help steer future development.

Making small casual iOS games fits perfectly with this model. We decided at an early stage during development that we would leave all features that didn't fit with our core gameplay for a future update. This helped keep the development time down to three months of part-time work. It allowed us to stay focused and determine at an early stage whether the game was going to be successful.

The playable teams in *Flick Buddies* are made up of three members of a "family". This allows us to easily add new families with each update without having a major art overhead. The family theme includes three unique models and a single background.

As our families have completely different themes, we are able to take requests from our players as to what they'd like to see. This fits well with our desire to have lots of small updates to the game, constantly adding features requested by our players.

4. Art Style

One of the most common comments we receive from players is that they love the art style. Shauno, our artist, has a unique style to his art that he has not been able to fully express in games he's developed for the mainstream industry in the past. The team wanted to give him complete artistic freedom to develop the characters he wanted as well as his unique style. The small, fat little characters worked perfectly for a physics-based game where characters bounce around the level.



This also helps with sales, as both the game icon on the App Store and the initial screenshots have a real "punch" to them that makes them stand out from other games. We tried to get our in-game 3D assets to have a similar look, with thick outlines matching the 2D illustrations. For performance reasons we had to move from 2 outlines (one black, one white) back to a single black outline; however, we're still happy with the result.

5. Playtesting

I've [previously written](#) about my thoughts on playtesting and why I think it's so important. We made sure to begin our playtesting as soon as possible and continued throughout development.

Developing an iOS title suits playtesting perfectly, as you can take your game to the testers. Everywhere I went I would have a build on my iPhone or iPad, and anyone I spoke to that seemed interested was able to quickly play the game.

Both our ability activation and "join game" functionality was changed regularly after watching people fumbling with our original implementations. Without this early feedback we may have released a game that only a small percentage of people would figure out how to start a match.

The best way to get the most out of playtesting was to hand the phone over, show them the icon, and let the player fend for themselves. They would be asking questions about how the game works and what they had to do, but we would force ourselves not to say anything and let them figure it out. This could be quite painful at times; however, it was invaluable for finding issues we completely missed as we were experienced with the game.

My rule of thumb is to let them fumble through until they give up. I then make a note of the issue, explain it to them and let them continue.

What Went Wrong

1. Unforeseen Circumstances

Our initial estimate was to create the game in two months part-time, with our first update coming in late December with a Christmas theme. After just one month of development, our artist fell sick, was admitted to hospital and was out of action for almost an entire month.

This couldn't have come at a worse time as the gameplay was nearing completion and we were moving into a production period where the critical path was almost exclusively in art. This stalled the project and put us almost an entire month behind schedule.

Thankfully our artist has completely recovered -- however, it showed the one weakness in our small team mentality. With only one person handling each "department" of the game, we are in the position where if any one person is sick the game development slows considerably.

More recently, the floods which swept through Australia have affected most members of the team as we worked towards our first update. I was cut off from the world and without power for three days. While this has only put us a few days behind schedule, it's another event that we simply couldn't predict as we created our initial estimates of development time.



[Click for large image.](#)

2. Bad Release Timing

The App Store is a crowded space and it is difficult to get noticed among the thousands of applications being released all the time. Due to our delayed development, we ended up releasing on the 15th of December during the busiest time in game releases. We were among some of the biggest games released on the iOS platform to date, which meant we were easily drowned out by the bigger games.

Unfortunately there wasn't much we could do about this, and we decided it was better to release before the holiday period -- otherwise we would have to wait another month until we started receiving money from the game to help fund future updates and new games.

This was made worse by the release date issue we experienced. When submitting the game to Apple, a developer can set the release date for the game. We set this for the 15th December to coincide with our release party at the [Mana Bar](#). The game was accepted by Apple on the 10th December and sat ready to be released for five days. Unfortunately, upon release, our "release date" was set as the 10th. This meant that we didn't show up in the new release section at the start of the list as we should have. This hurt our initial sales and made it difficult for us to rise in the charts.

3. Core Gameplay Confusion

Overall *Flick Buddies* has received good reviews, with many rating us 70 percent or above. Unfortunately, a number of negative reviews surfaced claiming the game was too simple and lacked depth.

After some investigation, we found that the reviewers had completely missed the special abilities in the game. This reduced the game to simple flicking and waiting for the next buddy to respawn, rather than the intended gameplay of flicking and timing special abilities to help reach the goal.

It's easy for us to get angry at the fact the reviewers didn't fully understand the game before writing their review, however it comes back to a game design flaw at our end. We were not explicit enough in explaining how the special abilities work, and therefore on a brief playthrough they can be overlooked.

Unfortunately, this didn't come out during playtests, as we found nearly everyone understood the visual cues when playing. This could be due to the fact the playtesters had heard us describing the abilities, or saw us using them when playing against us. We have plans to make the abilities more obvious in our update, however this is too late for the couple of initial negative reviews.

4. Piracy

We were completely surprised to find *Flick Buddies* available on a pirate website within 48 hours of its release. I had heard rumors of people asking for promo codes to acquire the game for free and then cracking it. Unfortunately, one of the promo codes we gave out to a seemingly legitimate user ended up being used to crack the game and release it for free on the internet.

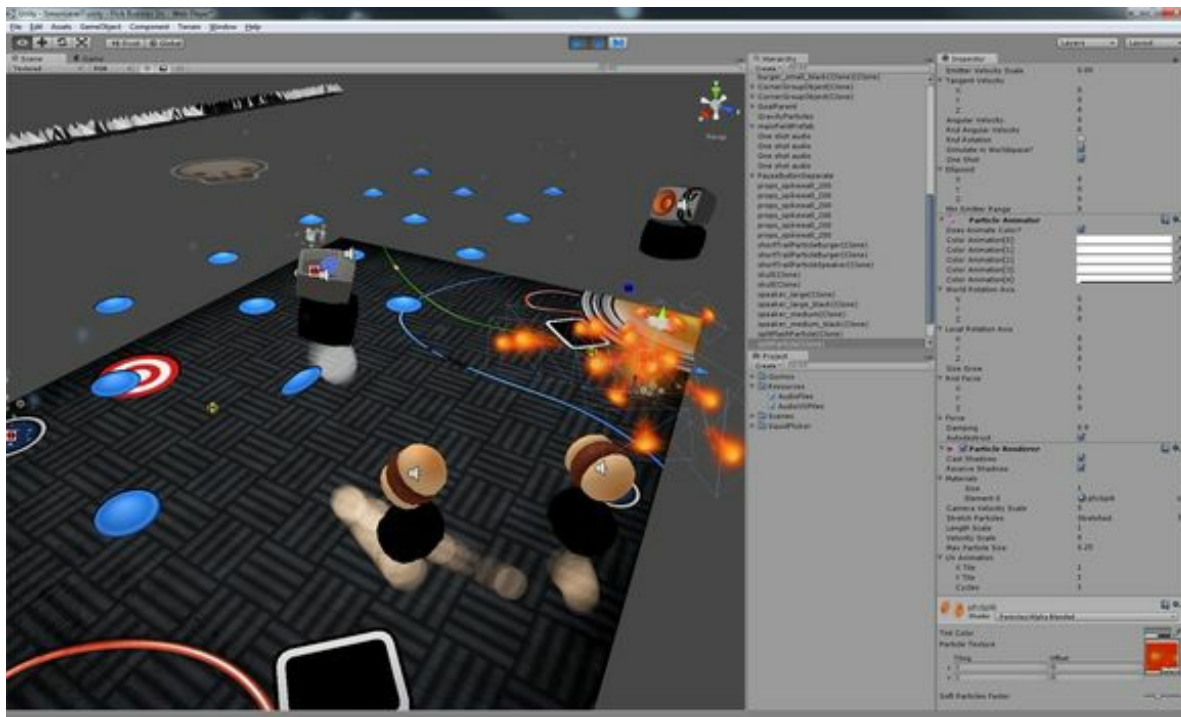
While I don't think this would have hurt our sales too badly, it was still quite disappointing, particularly with the game coming in at a 99 cent price point. We have learnt our lesson and plan to be more scrupulous with whom we share our promo codes with in the future.

5. Last Minute Problems

The final negative event is only small, but caused so many grey hairs. We distributed our press releases to media outlets around the world. This seemingly simple process took countless hours to get right. We were trying to organize an HTML email to go out with embedded pictures and video of the game. Everything that could go wrong seemed to, as we prepared the email and spent far too long going back and forth getting it right. We ended up sending the press release out and all was okay; however it ended up taking a total of almost six hours from when we started on it to release.

This fits well with most of the marketing/press material we created. My rule of thumb now is to allow at least a full

week of time to prepare marketing material for all the sites around the internet. This ranged from various image sizes to custom press releases targeting different sites. While we had prepared for this to take some time we were still surprised by just how long it all took to complete.



[Click for large image.](#)

We also had a similar issue from a technology standpoint. We only completed some of our assets right at the last minute, which seemed to push us over a memory footprint size restriction. After months of development with no crashes, we suddenly found the app was crashing regularly on most devices. There was a mad rush to find out the cause of the problems and we ended up submitting *Flick Buddies* to Apple a few days after we'd originally scheduled. We fixed all the problems, however it was fairly daunting having more issues in the last 24 hours than we did in the entire three month development process.

Behind The Scenes Video

Shauno, our artist, has put together behind the scenes video showing the process of creating one of our characters. You can check the video out on YouTube below:



What's Next?

We're currently working on the first update to *Flick Buddies* which will include many of the requests we've received from our players. We're adding a new game mode, family, set of levels and OpenFeint support. I've also started prototyping other games in the *Flick Buddies* franchise. This update also includes a web version of *Flick Buddies* intended for Kongregate's Unity competition.

We're also working on completely new games. We're looking at something completely different that is a little less casual. Our current two prototypes are for a new take on a platformer and strategy game.

Game Data

Developer: Bane Games

Partners: SQUIDTANK and IndieNoizes

Platform: iOS (iPad and iPhone/iPod Touch)

Release Date: 15th December 2010

Development Time: 3 months part-time

Number of developers: 4 – 1 programmer, 1 designer, 1 artist, 1 sound engineer

Technology: Unity, Adobe Photoshop and Adobe After Effects

Number of Files: 3,408

Lines of C# Code: 5,549

[Return to the full version of this article](#)

Copyright © 2015 UBM Tech, All rights reserved