

Theo yêu cầu của khách hàng, trong một năm qua, chúng tôi đã dịch qua 16 môn học, 34 cuốn sách, 43 bài báo, 5 sổ tay (chưa tính các tài liệu từ năm 2010 trở về trước) Xem ở đây

**DỊCH VỤ  
DỊCH  
TIẾNG  
ANH  
CHUYÊN  
NGÀNH  
NHANH  
NHẤT VÀ  
CHÍNH  
XÁC  
NHẤT**

Chỉ sau một lần liên lạc, việc dịch được tiến hành

Giá cả: có thể giảm đến 10 nghìn/1 trang

Chất lượng: Tao dựng niềm tin cho khách hàng bằng công nghệ 1. Bạn thấy được toàn bộ bản dịch; 2. Bạn đánh giá chất lượng. 3. Bạn quyết định thanh toán.

Tài liệu này được dịch sang tiếng việt bởi:

**[www.mientayvn.com](http://www.mientayvn.com)**

Tìm bản gốc tại thư mục này (copy link và dán hoặc nhấn Ctrl+Click):

<https://drive.google.com/folderview?id=0B4rAPqlxIMRDSFE2RXQ2N3FtdDA&usp=sharing>

Liên hệ để mua:

[thanhlam1910\\_2006@yahoo.com](mailto:thanhlam1910_2006@yahoo.com) hoặc [frbwrthes@gmail.com](mailto:frbwrthes@gmail.com) hoặc số 0168 8557 403 (gặp Lâm)

Giá tiền: 1 nghìn /trang đơn (trang không chia cột); 500 VND/trang song ngữ

Dịch tài liệu của bạn: [http://www.mientayvn.com/dich\\_tiang\\_anh\\_chuyen\\_nghanh.html](http://www.mientayvn.com/dich_tiang_anh_chuyen_nghanh.html)

## Making Your First Game: A Walkthrough for Game Developers

Steven Lambert on Feb 19th 2013 with

10 Comments

This entry is part 1 of 6 in the Making Your First Game Session

When you get stuck in a video game, you might head to GameFAQs to find a walkthrough to help you get through it. When you get stuck making your first video game, what can you do? That's where this article comes in: think of this as a walkthrough for developing your first game.

### My First Game

Before we get talking about the video game development walkthrough, I want to tell you about my first game. It was a simple text based adventure game inspired by a MUD called Divine Blood:

Making\_Your\_First\_Game\_divine\_blood

It was first programmed as just a simple room using the TI-83+ Calculator, and later made into a full-fledged town using Java in my last year of high school. I never finished that game (it didn't go

Thiết kế game đầu tiên cho riêng bạn: Hướng dẫn cho người lập trình game

Steven Lambert ngày 19 Tháng Hai năm 2013

10 Bình luận

Bài này là một trong 6 bài về chủ đề Thiết kế game đầu tiên cho riêng bạn

Khi bạn gặp khó khăn trong video game , bạn có thể đến mục Những Câu Hỏi Thường gặp về Game để tìm hướng dẫn cách khắc phục vấn đề đó. Khi bạn gặp khó khăn trong việc thiết kế game đầu tiên của bạn, bạn sẽ làm gì? Bài viết này sẽ giúp bạn giải quyết khó khăn đó: hãy xem bài viết như một hướng dẫn thiết kế game đầu tiên của bạn.

Game đầu tiên của tôi

Trước khi trình bày hướng dẫn tạo video game , tôi muốn kể với bạn về game đầu tiên do tôi tạo ra. Đó là một trò chơi phiêu lưu dạng text được nghĩ ra từ một MUD được gọi là Dòng Máu Linh Thiêng:

[REDACTED]

[REDACTED]

beyond the first town) and I don't even think I have the code any more.

Unfortunately, my experience isn't uncommon for game developers. Many developers start a game but never finish it. So how can you increase your chances of finishing your first game? Enter the video game development walkthrough.

### The Four Levels of Game Development

If developing a game is a game in itself, then it consists of four levels: Plan, Prototype, Develop, and Release.

Just like in a video game, each level is more difficult than the last. Each level also has its own encounters (the problems and difficulties you are likely to come up against) and its own end level boss (which signifies when you are ready to move on to the next level).

This walkthrough will guide you through the process of successfully creating your first game. It will incorporate resources and quotes from other game professionals, as well as insights about how implementing the walkthrough for my own first game would have helped me to finish it.

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

## Level 1: Plan

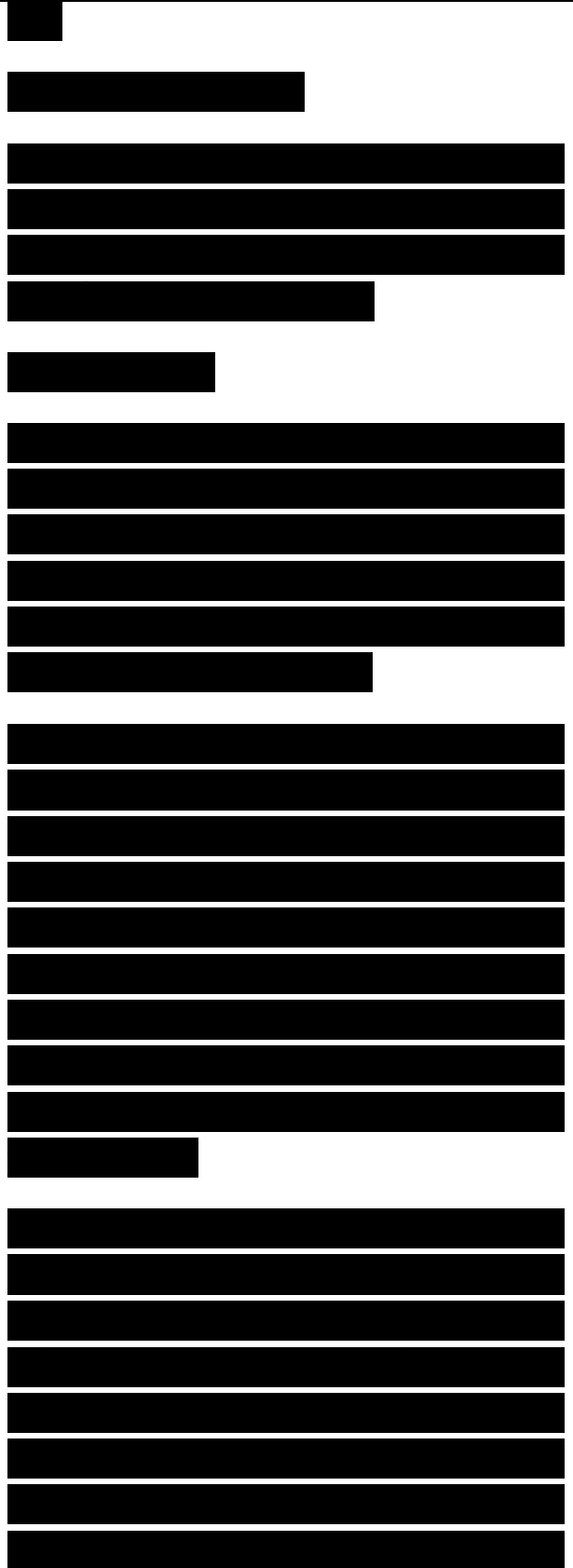
Remember, the first 90% of your game takes 90% of the time; the last 10% takes the remaining 90% of the time. Plan accordingly.

– Ian Schreiber

The first level, and arguably the most important level, is planning – planning out every aspect of the game before you even begin coding. The goal of this level is to consider everything about your game up front so that you don't have to improvise later.

When you first start making your game, the best thing to do is write down all the details. Think about everything you want in your game and then write it down. Usually this is pretty easy at first as your mind races with lots of neat ideas that you will want to incorporate into your game. However, as time goes by this level becomes increasingly more difficult to complete.

As you jot down your ideas, also jot down the details of these ideas: how they will work, how they will affect the player experience, and how they will affect the gameplay. You don't want to leave any big holes in your design that will cause you to struggle later.



It becomes more difficult to finish a game as the problems you encounter increases. Thus, the secret to finishing a game is to deal with problems in small amounts and without letting them pile up all at once.

Game development is about the trade-offs — great games don't have to do everything, they just have to a limited number of things well.

– Patrick Wyatt

Because of this, it is extremely important that you do not skip this level and head straight into development (Level 3). If you did, you would encounter hard problems that would cause you to quickly become demotivated and stop. If you've played the Halo series, just think about how hard the game would have been if you encountered a Hunter pair during the first level of your first play through. You would have been hard pressed to continue playing if you kept dying over and over again to these massive obstacles.

This is why the main encounter for this level is a desire to stop planning and begin coding. Again, it is important that you plan out as much as you can before you begin coding. For you to have the greatest chance of finishing your first game you must resist this urge and stick to Level 1.

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

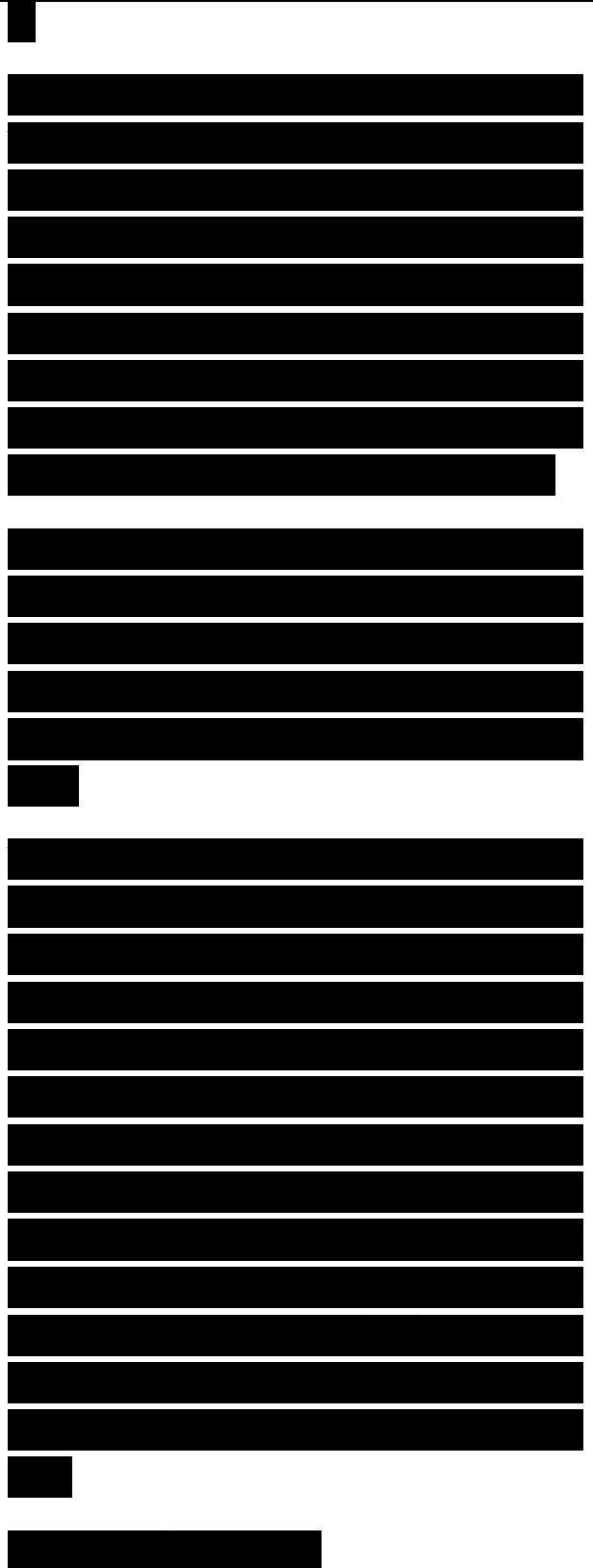
[REDACTED]

The boss fight for this level is questions. If you review all your notes and you can't think of any more questions, then you are ready to move on to the next level. If, however, you read over your notes and you still have a couple of questions about how things will work, you should probably work those out before moving on.

There are times when you can't answer a question because it deals more with balancing issues rather than functional issues. In this case you are also probably ready to move on to the next level.

If I had planned out my text based adventure game properly, I would have mapped out the entire game, decided what all the items the player could purchase would be and where they could purchase them from, designed all the monsters and dungeons, and even figured out which loot would be dropped where and when. That's a lot of detail that should be planned out before coding! For this reason, I don't recommend making a text adventure game as your first game. Start small.

Related Posts  
[The CAKE Design Process](#)



Game Design Concepts

Level 2: Prototype

Design is a process, not an outcome.

– Christopher Simmons

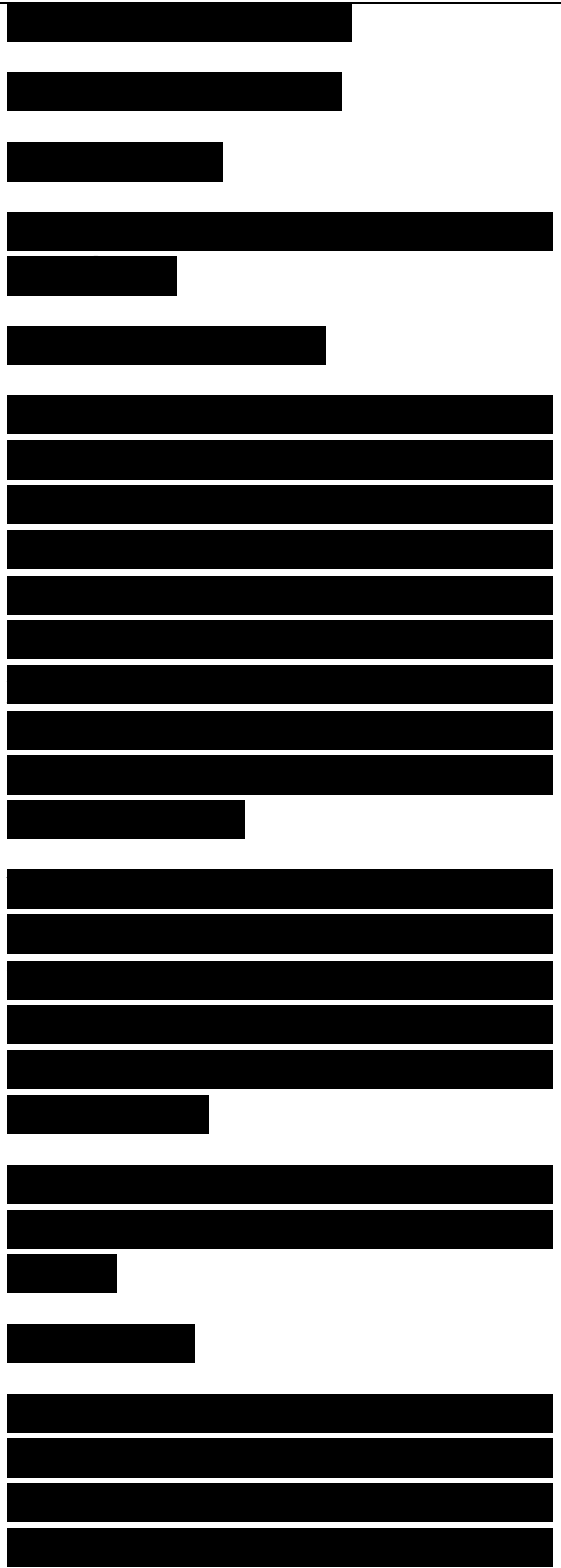
The second level of game development is prototyping. In this level, you test out the core mechanics and features of the game to work out any balancing issues. You want to flesh out your ideas quickly by using low fidelity prototyping so that you can know before you start coding what will and will not work.

If a game is fun to play at this level, you know it will be fun to play when it is finished. Remember that you will be playing this game a lot as you code it, so you'd better make sure it's fun!

No matter what your score is, you'll enjoy it, and want to play it again.

– Stan Jarocki

Even though this is Level 2, it is very common to prototype ideas as you are writing them down in the planning level. Now you are probably thinking: "hold on – you just told me that I shouldn't skip



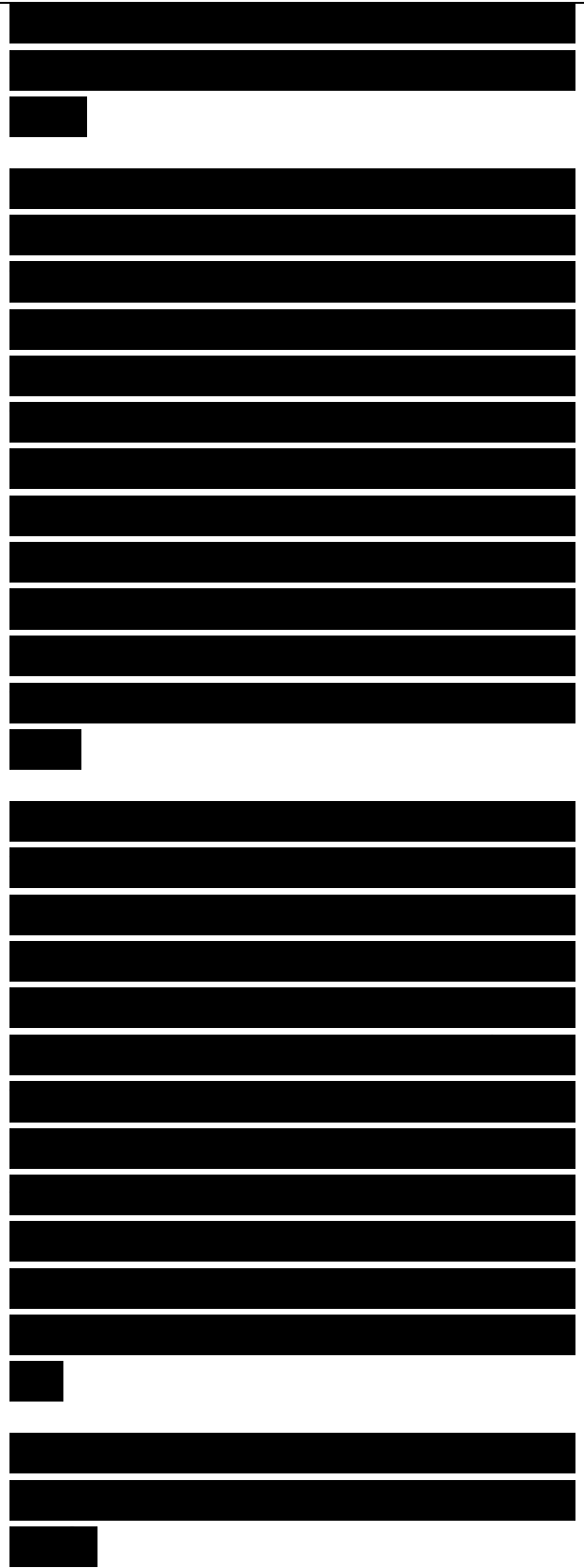
levels, but here you are telling me that I can skip to this level!”

Think of it like level skipping using the secret warp pipes in Super Mario Bros. The first time you play the game, you’re probably going to play each level in order. It isn’t until the second or third playthroughs that you use the warp pipes to finish the game faster. The same thing applies with Levels 1 and 2, except that going back and forth between planning and prototyping helps you flesh out your ideas faster.

Levels 1 and 2 are interchangeable, as long as you don’t progress to Level 3 before you are ready. Maybe you want to test out your core mechanic before you spend more time on details. Perhaps you want to test out a new feature of your game to ensure that it will work as intended. These are all good reasons to want to prototype while you plan; just ensure that each prototype only focuses on one thing at a time, and that each time you prototype the gameplay it’s still fun.

Great game-play is a stream of interesting decisions the player must resolve.

– Bruce Shelley

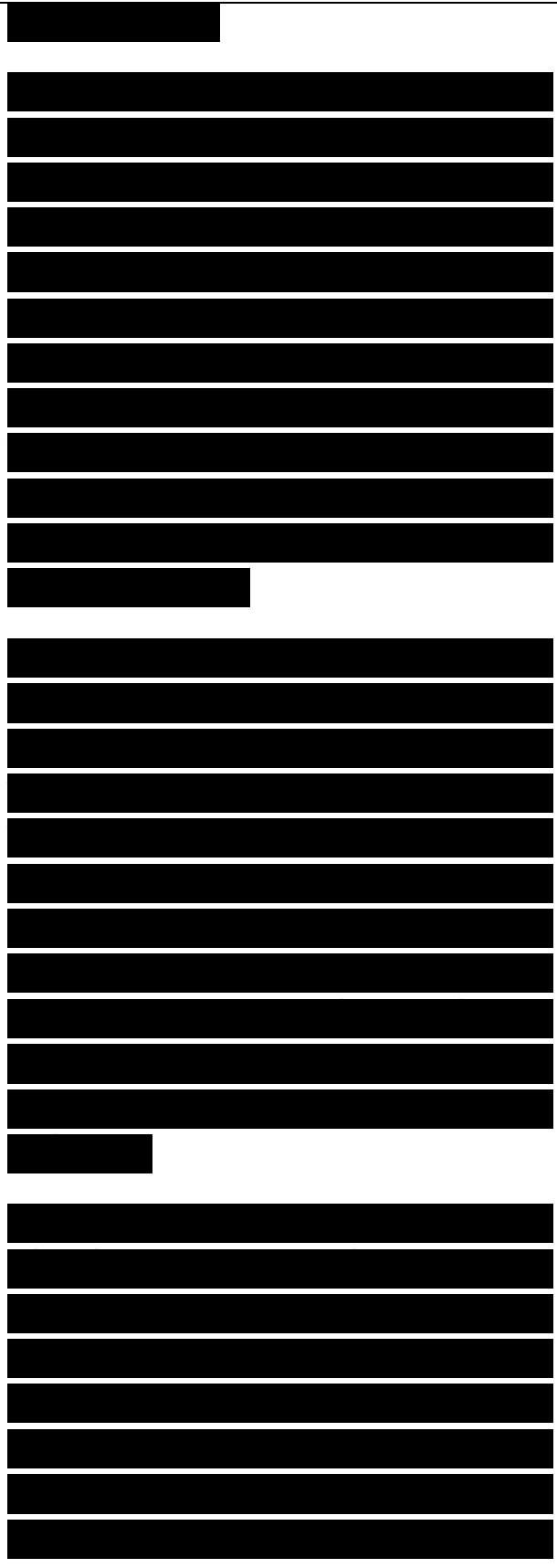




There are two encounters that you will most likely face in this level. The first is again the desire to just skip to the next level and start coding. The second encounter will be a desire to create higher fidelity prototypes. As good as high fidelity prototypes are, they are a dangerous path as they will increase your desire to start coding when you are not ready. To fight this battle, stick to quick rapid throwaway prototypes.

The boss for this level is completeness. When you practically have the entire game planned out and fleshed out using prototypes, you are ready to move on to the next level. You want to have a good idea of how the game will play out from start to finish so that you are not dealing with these problems while coding. Again, the less problems you have while coding, the more likely you are to finish your game.

Prototyping my first game would have involved balancing the game. This includes the combat system, how much items should be worth and how much bonus they should give the player, all the stats for both the player and the monsters, and even details such as story and quests. Again, there are a lot of details that go into making an RPG that



need to be worked out before coding.

## Related Posts

Tips for a One Man Gamedev Team

What to do Before Even Touching a Computer

Game Balance Concepts

Game Design Workshops

Level 3: Develop

Before you start the programming part, cut 90% of your intended features.

– Ian Schreiber

The third level, developing, is the most difficult level to complete. It is here that many developers lose their motivation and their desire to keep going. But fear not! With the first two levels successfully completed, you should be well on your way to finishing your game.

To begin this level, you will want to figure out what platform you want to release your game on. Will you release it for the Xbox, as a mobile app, or as a web game? After you have chosen your platform, choose a development engine or programming language that will work for this platform.

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

[Redacted]

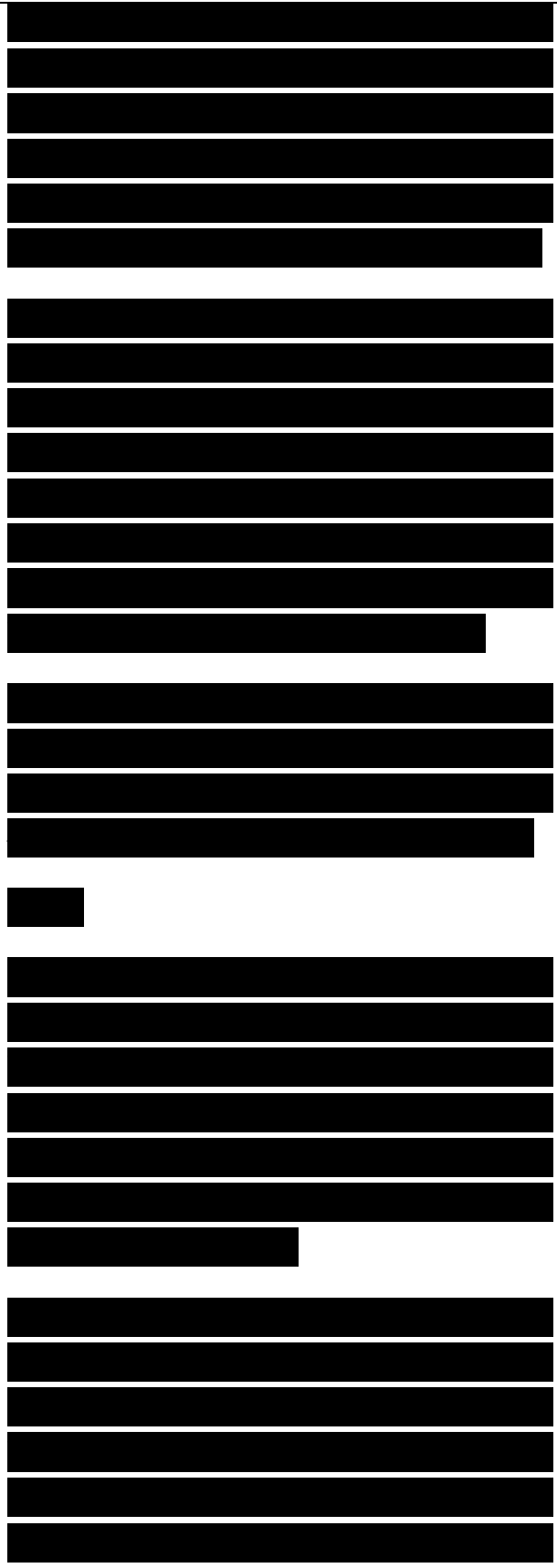
This can also be done in the reverse order if you are comfortable with a development engine or programming language. Your platform will then be determined by the capabilities of your chosen engine or language.

To help finish this level, look for “power-ups” like free code libraries, free game assets, and free online resources. Don’t try to reinvent the wheel; instead, use what is freely available to help you finish this difficult level.

‘I can’t draw’ — use clipart. ‘I can’t make sound’ — google midi files. ‘I can’t code well’ — no-one cares, you don’t need to.  
– Rob

The main encounter for this level will be problems and discouragement. There will be times where you encounter a problem to which you don’t know how to solve. You will also become discouraged the more you work on your game as it doesn’t live up to your high expectations.

To fight this encounter you must press on! If you have to, take a few days off from your game when you get stuck. This will allow your mind to clear and fresh new ideas to enter. Usually just sleeping on the problem helps you to find a solution the next morning. Don’t work



yourself to death; pace yourself so that you don't become too stressed or discouraged.

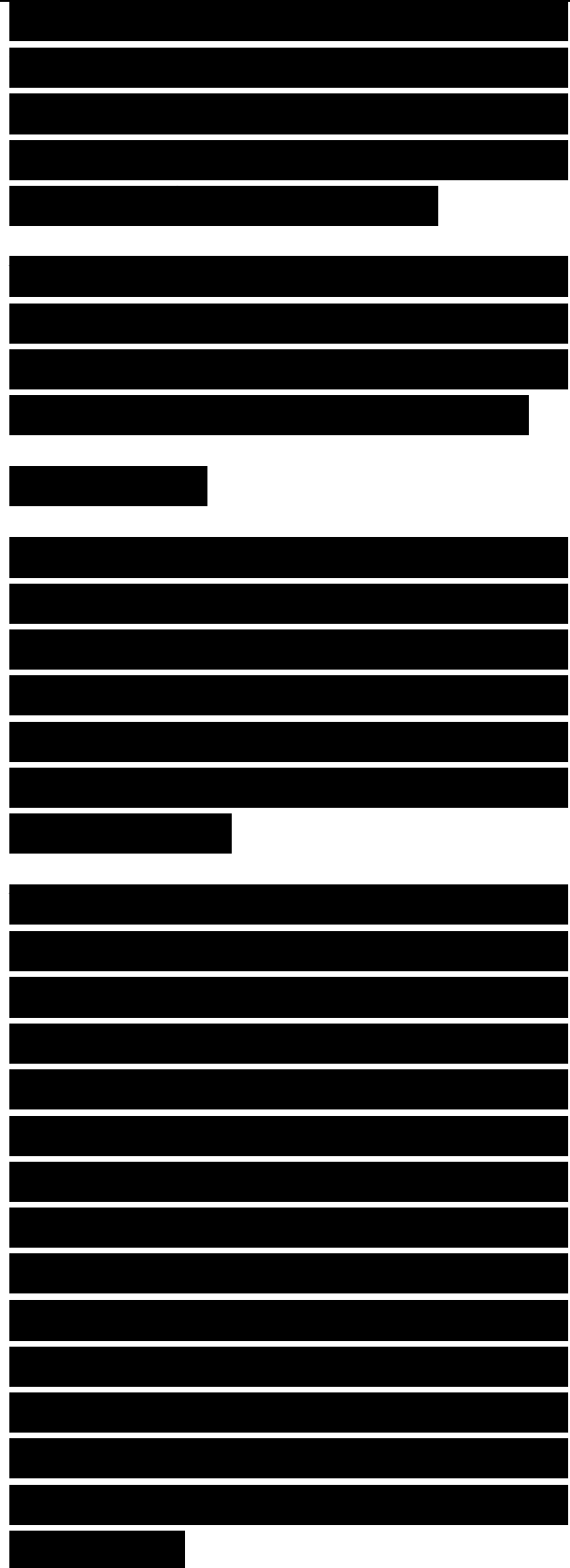
If you are trying to solve a really hard problem, stop. Solve a simpler problem the accomplishes the same things.

– Daniel Cook

The boss for this level is a game. The game doesn't have to be perfect nor fully finished, but it should be something that you could release and would want people to play. Once you have that done, you are nearly finished with your game.

If I had planned and prototyped before I started coding, I could easily have done more than just a single town. I would have know exactly what to code and how to code it beforehand, making it really easy to get more things completed quicker. I wouldn't have struggled as I tried to figure out the layout to my town, trying to decide where to put the shops and quests, and so on. As it was, I tried to do all of this while simultaneously trying to learn how to code and make a game. Work within your limits.

Related Posts



## How to Succeed at Making One Game a Month

### Level 4: Release

Never give up.

– Alice Taylor

The final level of game development is release. This level is not nearly as hard as Level 3, but it is still challenging. For this level you need to get people to play your game and to give you feedback. Based on this feedback (taken with a grain of salt), make changes that you feel will make the game more fun.

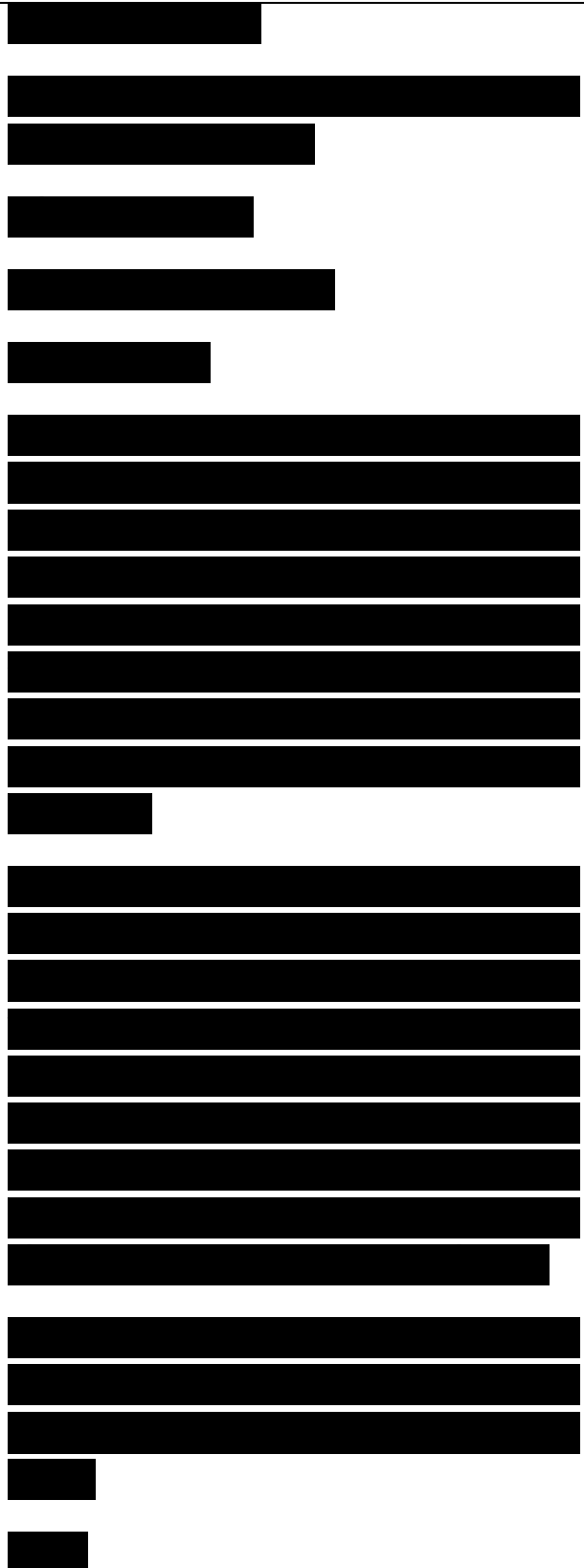
The main encounter for this level is dealing with criticism. There will be people that just don't like your game. This is fine. Don't let those people tear you down just because they feel the need to voice their opinion. What you will need to do is learn to focus on the people that do like your game and encourage them to keep playing, asking for their help to make it better.

The point of making games isn't to see who is better at making games, it's to make what you want to make.

– Rob

My first finished game.

My first finished game.



The final boss for game development is a game that you are proud of. It doesn't have to be any good (and, honestly, as it's your first game it probably won't be), but you made it! You made a complete game from start to finish, which is not something a lot of people can say.

Every game developer will have a 'bad' game. If not yet, at some point. All of us. Things happen.  
– Brenda Romero

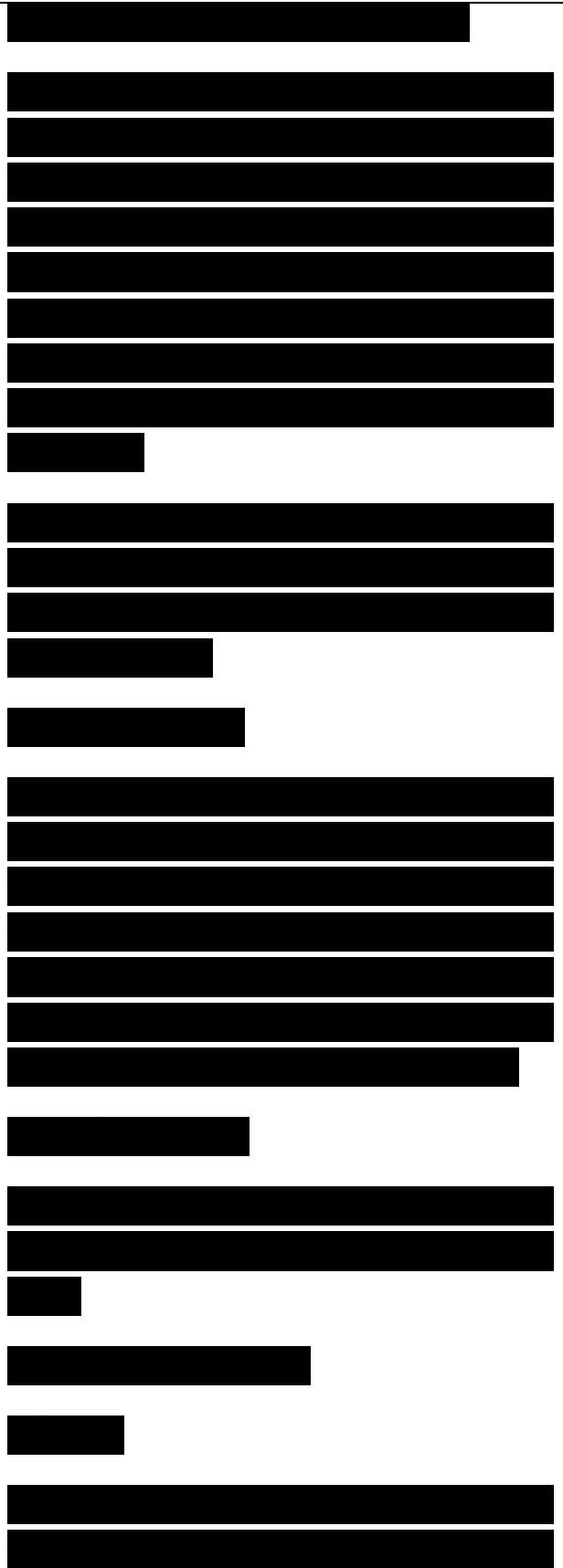
I may not have finished my first attempted game, but I am still proud of it to this day. It taught me a lot about game development and I was doing what I loved. Remember, success is paved by a road of failures. Don't give up!

#### Related Posts

What They Don't Tell You About Being a Game Developer  
Finishing a Game

#### Conclusion

Making your first game is an exciting and fulfilling experience. And yet, it is



very common for game developers to not make it through their first game, abandoning it for another game they also may never finish.

By following these four levels of game development, you help increase your chances of finishing your first game by knowing what the road ahead looks like and knowing how to deal with them. Now go out there and make your game!

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

Qua bốn mức phát triển game, chúng tôi đã giúp bạn tăng khả năng hoàn thành game đầu tiên qua việc nhận thức được viễn cảnh phía trước và biết cách giải quyết chúng. Bây giờ hãy tiến hành thiết kế game của bạn! **checked**