A BEAR'S JOURNEY

BY

GUIL, JACOB, and JORDAN

Introduction

A Bear's Journey invites you to play as a bear that woke prematurely out of its hibernation. Rather than being angry at the world for waking it up, the bear instead will go on a peaceful midnight stroll through the woods. It is up to you as the player to reach the end of the level and pick up tasty berries along the way. Sit back and get your jump on in *A Bear's Journey*!

Table of Contents

Statement of Purpose

A Bear's Journey is simple and has most of your favorite elements of games you used to play as a kid. Collectables, Platforming, and a cute character. You are a bear, you have just woken from hibernation, and most of all, you are hungry. The player must explore the once flourishing land which has now become frozen.You must look for the scare food that has survived the harsh winter. Once the player has found food, they may proceed to the next level. Look high, and look low, food comes scarce in these lands.

Accomplishments

The entire project was built in the Java language within the Eclipse IDE by team members Jacob Davis, Jordan Skerda, and Guil Poelsma. The planning and design were informed by Scrum practices. For extra knowledge and guidance on 2-D platformers in Java we referred to Youtube user's Matt The Potato's Java programming guide. We also used Github.com to share project versions amongst group members. To further manage our project timeline we used project management tool ProjectLibre.
All this resulted in a responsive, and fun 2-D platformer which we proudly call "A Bear's Journey".
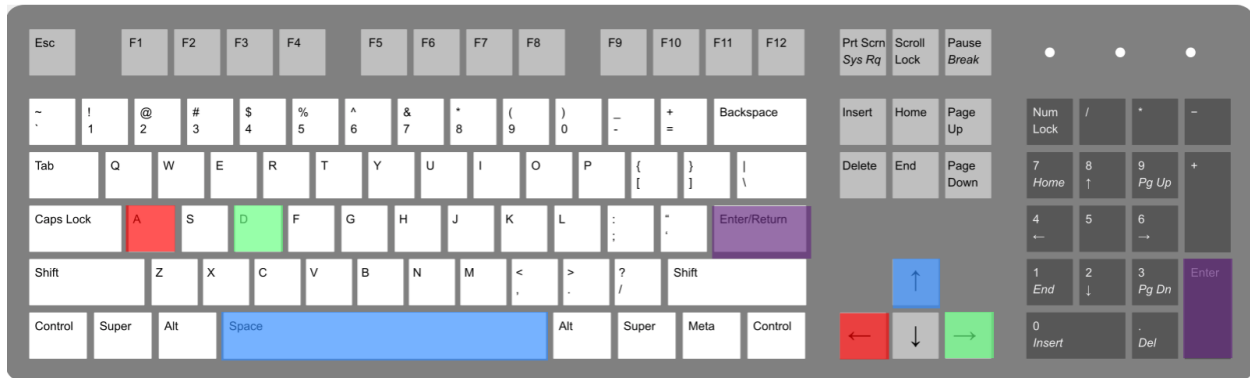
Controls

Figure 1.0: Controls

Our control scheme is simple. We allow the players to choose how they move with either A/D, or arrow keys. Don't have a mouse? No problem! You can use the enter and arrow keys to make selections in the menu.

A / ←	Move Left
D / →	Move Right
Space / ↑	Jump
Enter	Make Selection on Menu

Figure 1.1: Key for Control Scheme
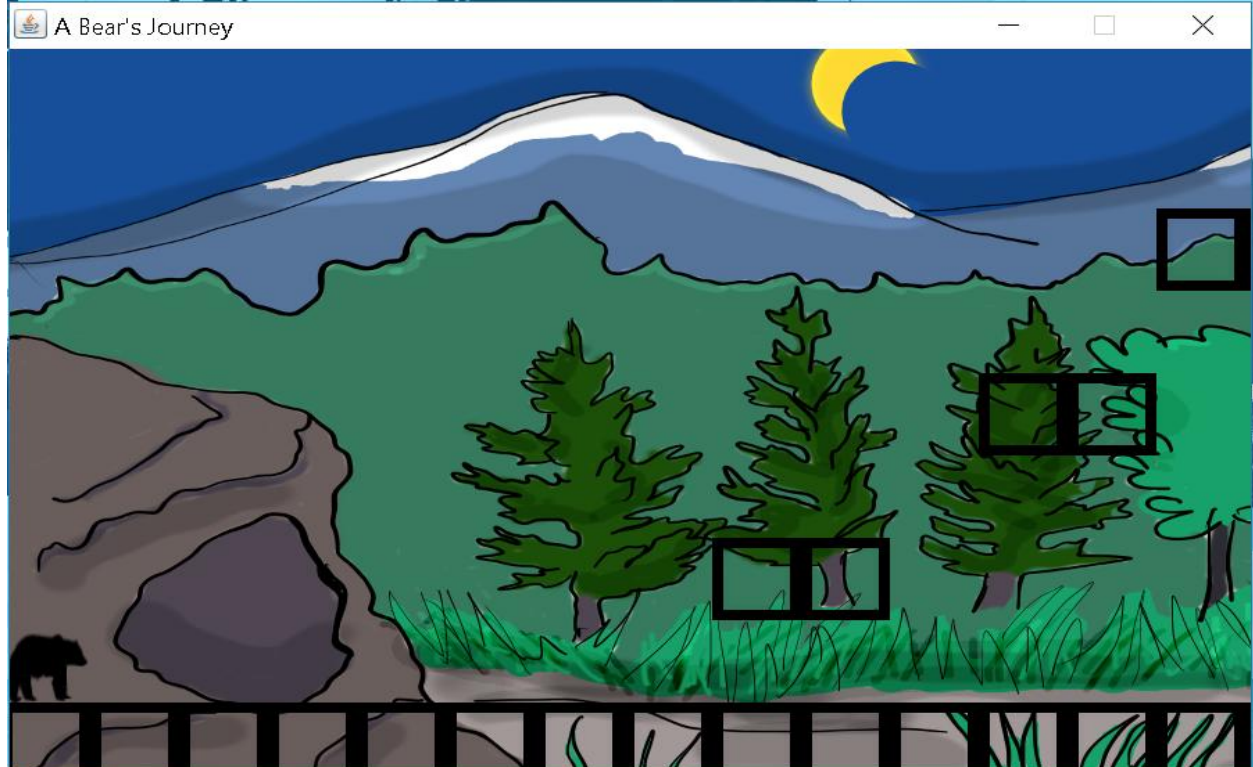
Screen Shots

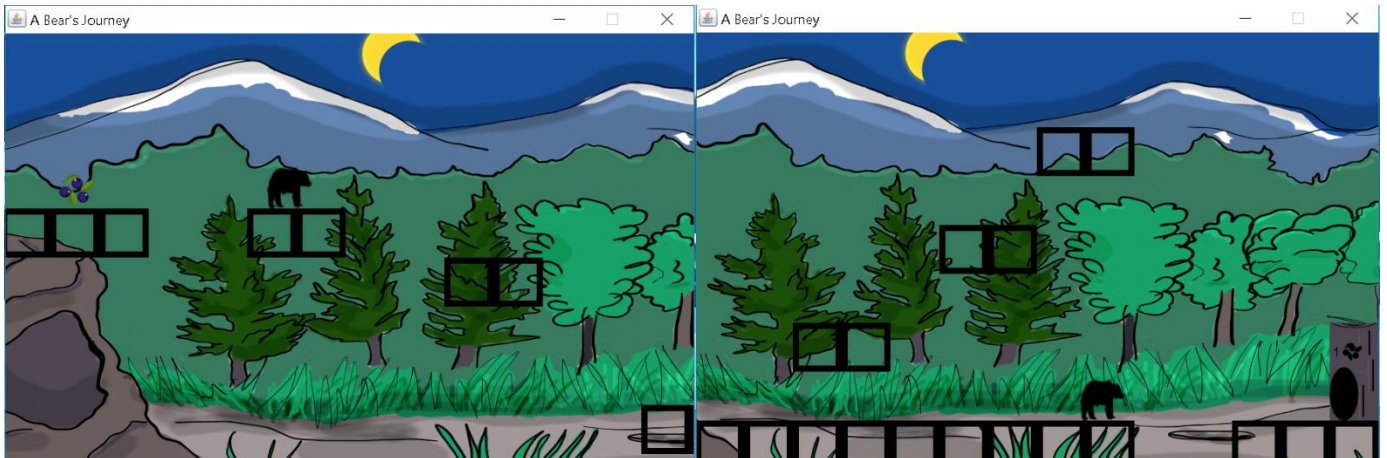Figure 1.2: Game Menu

Figure 1.3: Starting Position Level 1

Figure 1.4: Level 1 Berry Pick up and Level 1 Exit

Figure 1.5: Starting Position Level 2

Unique Parts of the Game

Art

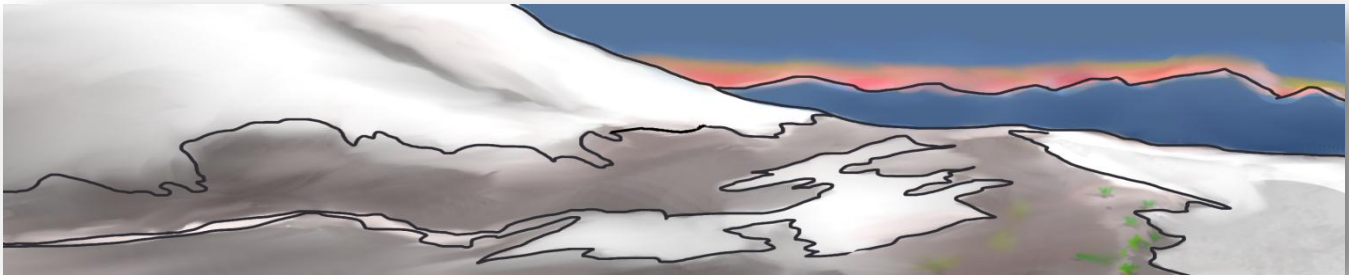Figure 1.6: Level 1 Background by Nicole Schumacher

Figure 1.7: Level 2 Background by Nicole Schumacher

Our game art is completely designed using Affinity Photo and Nicole Schumacher's artistic talent to create the background using Gimp. We use a unique art style that not only compliments one another, but adds to each others' beauty and atmosphere.

Level Design/Creation

Figure 1.8: Level image

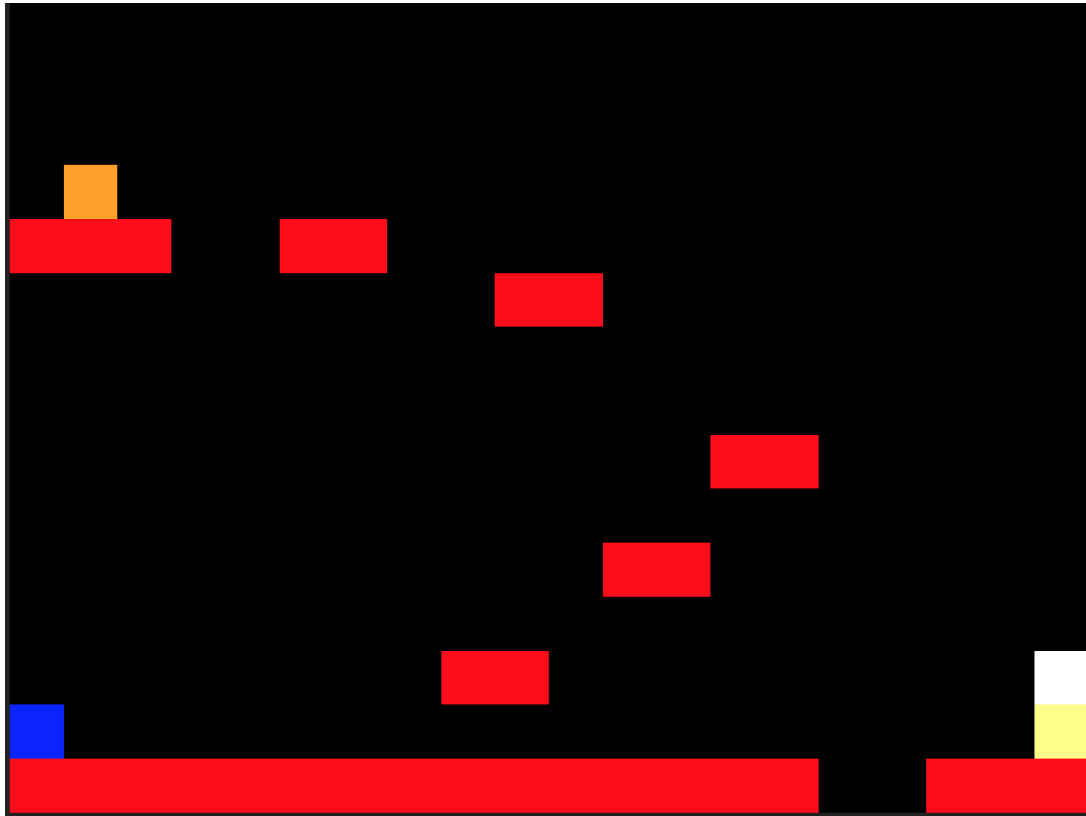Our levels are designed and also built in Affinity Photo. Our game uses a unique technique where each colored block in the .png is rendered as either a solid block (red), a collectable/pick up block (orange), the bear (player) spawn block (blue), level exit blocks (white/yellow/pink), and our misc. blocks (lime green, pastel blue). Our game pulls the hex code and the properties are given to the texture.
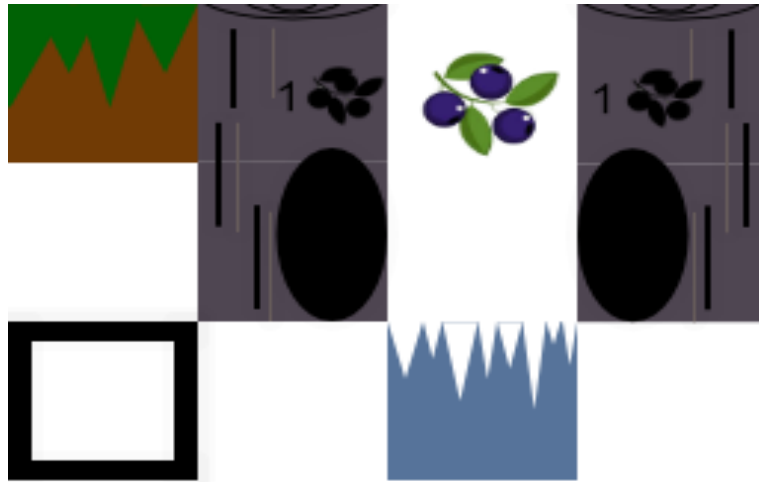
Sprite Sheet

Figure 1.9: Texture Spritesheet

Our game uses a sprite sheet for textures and character design to easily store and access the correct image needed and you read in the correct texture using a coordinate system. For example, the huckleberry would be the first row, 3rd column (1,3).

Gantt Chart

|	Ⓐ	Name	Duration	Start	Finish	Predecessors
1		⊟Pre-Production	20 days	9/15/17 8:00 AM	10/12/17 5:00 PM	
2		Project Proposal	6 days	9/15/17 8:00 AM	9/22/17 5:00 PM	
3	▣	UML Activites Diagram	4 days	9/25/17 8:00 AM	9/28/17 5:00 PM	2
4	▣	Create Scrum Paperwo	6 days	10/2/17 8:00 AM	10/9/17 5:00 PM	3
5	▣	Create Gantt Chart	3 days	10/10/17 8:00 AM	10/12/17 5:00 PM	4
6	▣	⊟Production	31 days	10/13/17 8:00 AM	11/28/17 5:00 PM	
7	▣	Create JFrame	6 days	10/13/17 8:00 AM	10/20/17 5:00 PM	5
8	▣	Create Simple Charact	3 days	10/23/17 8:00 AM	10/25/17 5:00 PM	7
9	▣	Confirm Character mo	7 days	10/26/17 8:00 AM	11/3/17 5:00 PM	8
10	▣	Create jumping/falling	8 days	11/6/17 8:00 AM	11/15/17 5:00 PM	9
11	▣	Add in platforms-check	3 days	11/16/17 8:00 AM	11/20/17 5:00 PM	10
12	▣	Create level obstacles	2 days	11/21/17 8:00 AM	11/22/17 5:00 PM	11
13	▣	Death Notification	2 days	11/27/17 8:00 AM	11/28/17 5:00 PM	12
14	▣	⊟Post-Production	13 days	11/23/17 8:00 AM	12/13/17 5:00 PM	
15	▣	Add details like art	1 day	11/23/17 8:00 AM	11/27/17 5:00 PM	
16	▣	Check for bugs	4 days	11/29/17 8:00 AM	12/4/17 5:00 PM	6
17	▣	Review game for issue	7 days	12/5/17 8:00 AM	12/13/17 5:00 PM	16

Figure 2.0: Breakdown of Tasks
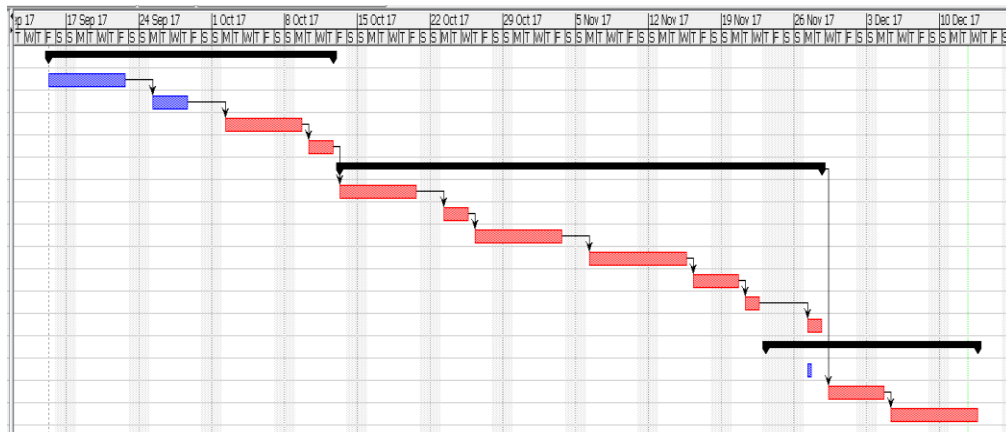
Figure 2.1: Gantt Chart Timeline

Our Gantt Chart was created using ProjectLibre, a project management tool. The red tasks are on the critical path, which means they must be completed in the alloted time or it will delay the finish date.

Scrum

Vision Statement:

This was our very first vision

"Create a 2D side scrolling platformer game. This game will be tooled and themed around a bear waking from hibernation and going on a journey through the seasons. What sets this game apart, gameplay-wise, is that our bear can switch between two modes of movement dubbed: "OnFours" and "OnHind". Players will need to use both modes to navigate through environmental challenges."
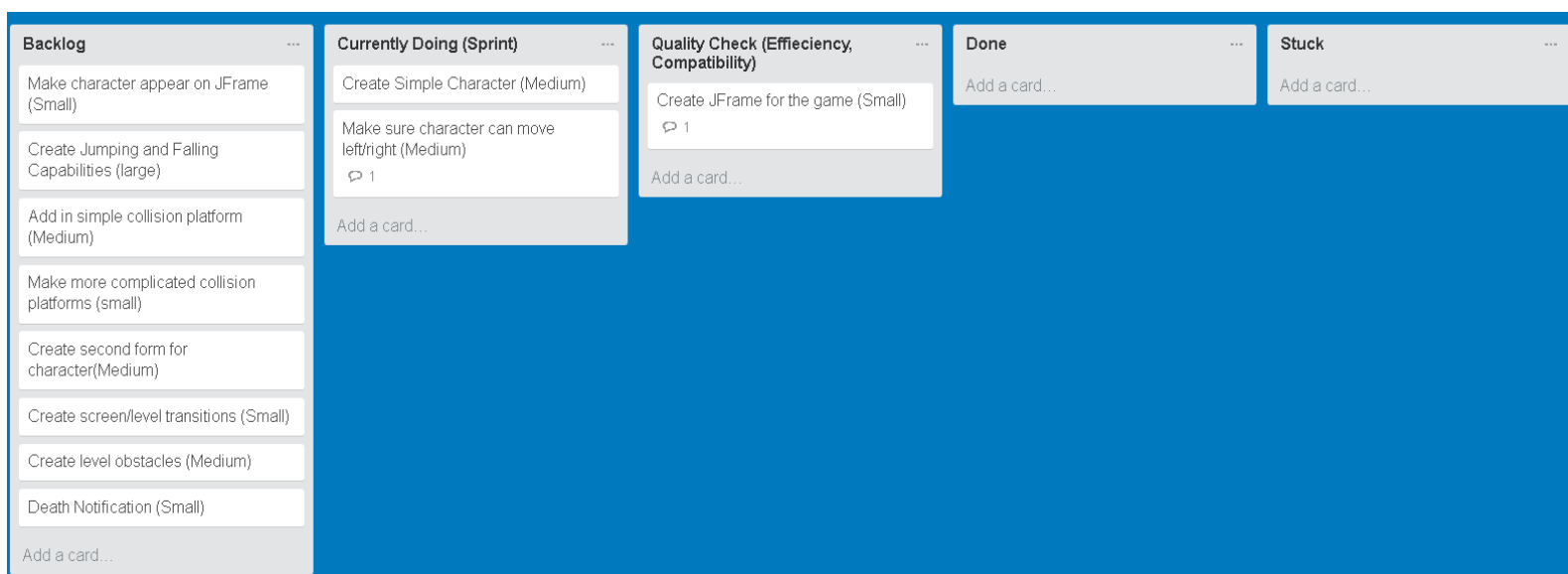
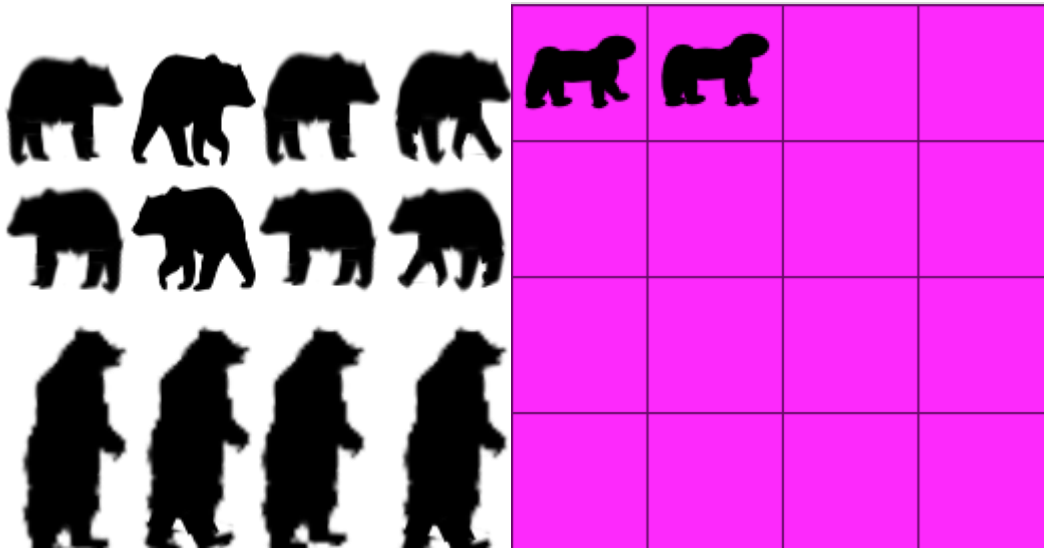| Backlog	Currently Doing (Sprint)	Quality Check (Effieciency, Compatibility)	Done	Stuck
Make character appear on JFrame (Small)	Create Simple Character (Medium)	Create JFrame for the game (Small) 💬 1	Add a card…	Add a card…
Create Jumping and Falling Capabilities (large)	Make sure character can move left/right (Medium) 💬 1	Add a card…		
Add in simple collision platform (Medium)	Add a card…			
Make more complicated collision platforms (small)				
Create second form for character(Medium)				
Create screen/level transitions (Small)				
Create level obstacles (Medium)				
Death Notification (Small)				
Add a card…				

Figure 2.2: Scrum

Animations

Figure 2.3: New spritesheet vs. Old spritesheet

Our animation is also made from scratch in Affinity Photo. We use the same sprite sheet design as our textures and levels. This was our first crack at getting spritesheets to work correctly, and it went just about how one might expect when using new technology. We struggled. We started with a very rough sketch of a bear, and gradually added more detail until deciding on the final bear. Once we figured out how to use spritesheets and edit them efficiently, we were golden.

Testing:

One of our most important ways of testing was System.out.println() statements and JOptionPane statements at intervals where we thought things were going wrong. We also took advantage of the debugging system in Eclipse to troubleshoot. We went through many iterations of our game using these aforementioned steps, which eventually led to this final version where the player can proceed throughout the entire experience without any glitches stopping them.

Things We Learned:

We learned that having a detailed vision, and roadmap is instrumental to making such a project happen. We accomplished the majority of what we set out to do except for having a "Switch Bear" capability. Even though that didn't make it into our current version from our initial vision it did further emphasize that planning is hugely important. In the final month of development we agreed that such a task was a bit too ambitious for our time line and thus agreed to exchange the "Switch Bear" capability with a pickup system. Still, we are very excited to continue work on this project and add that capability further down the line.

Among the many coding challenges that we overcame the best victory was when we made pickups work the way we wanted. We added pickups as an incentive for players to explore the level before being able to proceed to the next. The "door" would only let players pass through when pickup == 1. Which is great, but the way the code worked was once the bear came in contact with the pickup tile it would continually increment number into ludicrous values. Thus we had that pickup tile get set to null once the bear came into contact with it, ensuring it disappeared and would not increment past the value we wanted. As simple as this solution sounds we tried a ton of different solutions until it "clicked" and we solved it.

The Jar file of the game can be found at: https://github.com/Gpoelsma/ABearsJourney