# REVENGETABLES

Harman Singh

Nicolas Malamug

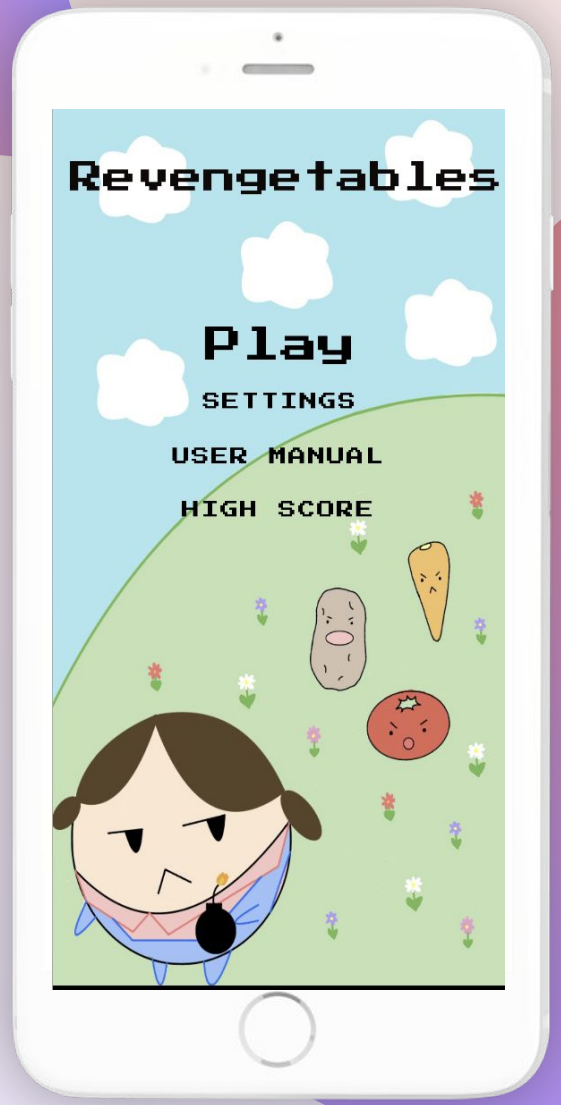
Bennett Taylor

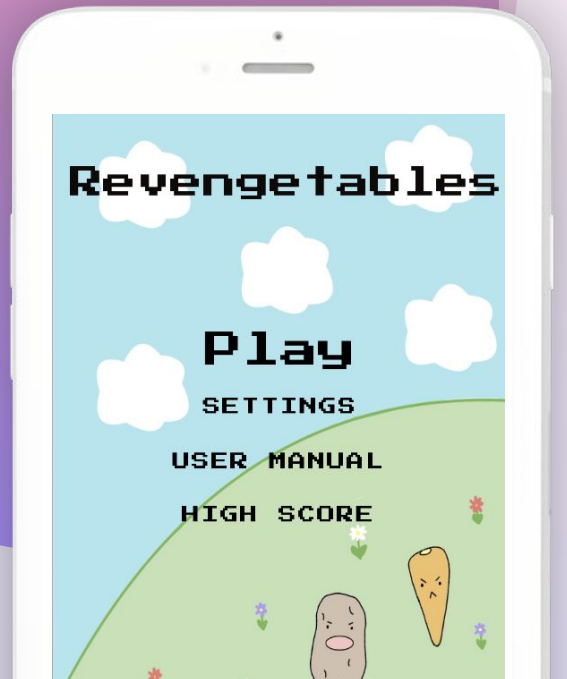
Joscelynn Palen

Maiko Lum

**COMING SOON**

Google Play



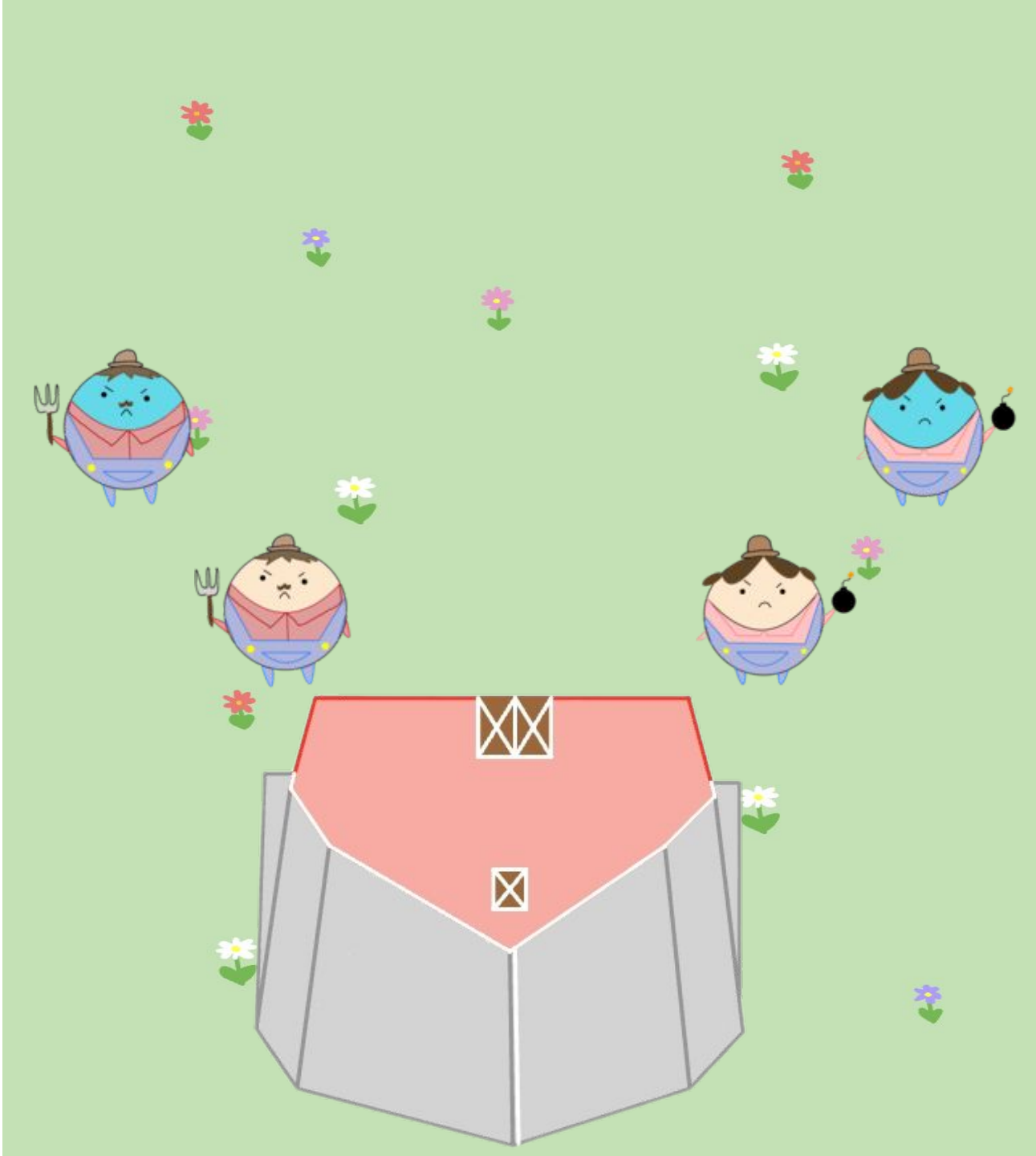
# 01

# THE INVASION...

Infectious vegetables have infiltrated the farm and it is your responsibility to protect the crops and the farm!

You will have a variety of farmers to choose from that have their own special abilities :

- the **EXPLOSIVE** farmer
  - when given a signal, you can blow away multiple vegetables via an explosion.
- the **DASHING** farmer
  - when given a signal, you can dash through a hoard of vegetables and sweep them off the screen.



# 02

# Requirements & Features

## Requirements

Moving graphical pieces.

- Farmer, Infected Vegetables

The ability of the user to control some of the moving pieces through input.

- Farmer responds to an input from the user ⇒ touching and dragging

Include a score that is changed based on the user satisfying requirements.

- Score is incremented based on
  a. Time elapsed
  b. Number and type of monsters killed

## Features

High-score list that persists when the app is closed and then reopened.

Allow the user to tweak the rules of the game being played.

- Types of Farmer Movements
  a. the **DASHING** farmer
  b. the **EXPlOSIVE** farmer
- Difficulty
  a. Easy, Medium, Hard
- Music on/off

Integrated sound effects for

- Home Screen
- Lose Screen
- Button Clicks

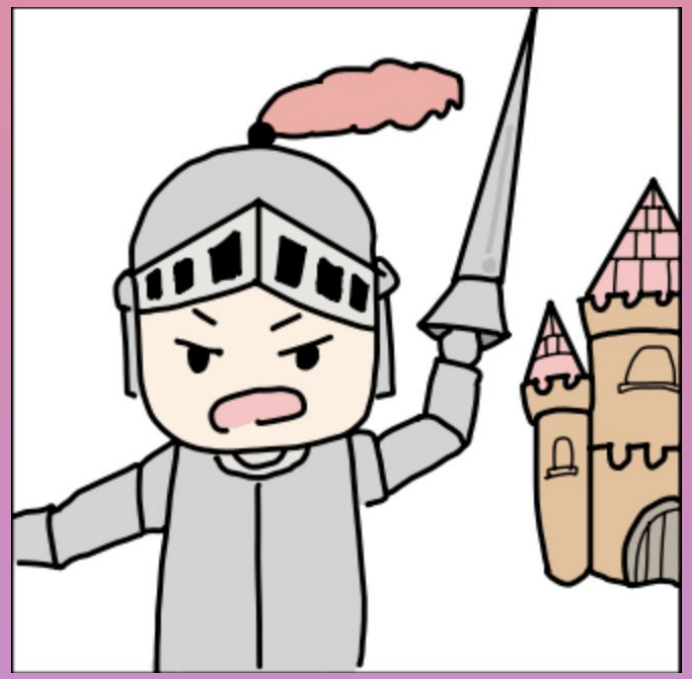
C++ Backend

# Design Development

We started with the concept of 'Knight's Labyrinth', where a brave knight defended a castle from attacking monsters.

However, we wanted to create a game that could appeal to a wider audience and decided to pivot towards a more universally loved game.

Thus, we developed a new concept that would provide an engaging and exciting experience for players of all ages and interests.

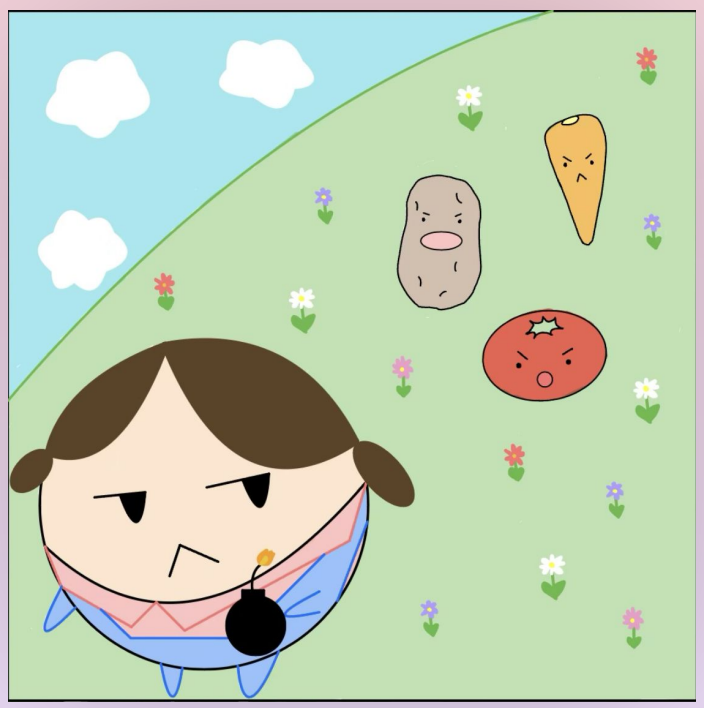


Original concept art for Knight’s Labyrinth

# 03

# Project Demos

# Logo



# LIVE DEMONSTRATIONS + GAMEPLAY

# 04

# Obstacles and Future Implementations

As we navigated through this project, we came across numerous issues:

- had multiple projects from other courses
  - caused roles to be switched around
  - had trouble keeping gitlab organized
- Minor things we didn't understand about android studio
  - Binding
  - Fragments

Although we were able to accomplish all of the deliverables, we fell short in a few categories:

- Sound effects were added, however, we did not incorporate any sound effects for the special ability of each farmer
- The C++ backend did nearly all calculations
  - we had a few functions in Backend that were more compatible with Java than C++
- The ScoreBoard also could have been greatly improved where we could have prompted the user for a name rather than call them players.

Remaining Deliverables:

- Unit Testing
- Proper Documentation
- Additional Features to improve user experience

Improvements:

- Use activities rather than fragments.
- Create separate classes instead of switch statements for each monster rather than combining them into one case system.
  - More efficient
  - Less chance of error
- Use Gitlab for issues and documentation

# Thank You

## Any Questions?