**Phase II Source Project Report for Kingdomino: Team Domino Dynasty**

University of Maryland Global Campus

UMGC CMSC 495 6380 Capstone in Computer Science (2238)

Alan Anderson, William Feighner, Michael Wood Jr., Ibadet Mijit, Jenna Seipel, & Joseph Lewis

Professor Shanna Nevarez

26 September 2023



KINGDOMINO

PHASE II

PROJECT REPORT

# Table of Contents

[Summary 1](#_Toc145696223)

[Timeline & Schedule 1](#_Toc145696224)

[Problems Encountered 2](#_Toc145696225)

[Task List / Schedule 2](#_Toc145696226)

[Github Learning Curve 3](#_Toc145696227)

[Different Java Version / IDE / System Configurations 3](#_Toc145696228)

[Re-evaluation of Decisions 4](#_Toc145696229)

[Using Java 4](#_Toc145696230)

[Building a Text-Based Foundation First 4](#_Toc145696231)

[2D vs. 3D 4](#_Toc145696232)

# Summary

Phase II is where we planned to implement a 2D layer over the text-based foundation that we created in Phase I. We assumed that doing it this way would save us time because all of the game logic is completed, and now we only need to convey it to the user on screen. Ultimately, we feel that we are having a successful Phase II and believe that this will make our Phase III even easier. The report below will cover our milestones for Phase II, timeline status and schedule, problems encountered, and re-evaluations of decisions we have made thus far.

# Timeline & Schedule

In Phase II, our team completed all 44 group and individual tasks. Below is a list of each of those tasks and the assignees. In particular, for the group tasks, those designated as “everyone” were done by everyone, whereas those designated as either “Framework Team” or “UI / UX Team” were done by the members of those teams. Individual tasks were completed by the assignee as shown.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **KINGDOMINO: PHASE 2 TASKS (30 AUG TO 24 SEP)** | | | | | |
| **LAYING THE 2D GUI FOUNDATION OVER THE FRAMEWORK** | | | | | |
| **#** | **Task** | | | **Assignee** | **Deadline** |
| **I -** | **Create and submit project design** | | | Alan | 12-Sep |
| **1.a** |  | Complete draft project design | | Alan | 5-Sep |
| **1.b** |  | Gather team input and incorporate feedback | | EVERYONE | 6-Sep |
| **1.c** |  | Submit final draft to message board | | Alan | 7-Sep |
| **1.d** |  | Receive professor's feedback and resubmit | | Alan | 9-Sep |
| **1.e** |  | Submit project design to assignment folder | | EVERYONE | 12-Sep |
|  | | | | | |
| **2 -** | **Create and submit test plan and user guide** | | | Jenna | 12-Sep |
| **2.a** |  | **Test Plan** | |  | 12-Sep |
| **2.a.1** |  |  | Complete draft test plan | Jenna | 5-Sep |
| **2.a.2** |  |  | Gather team input and incorporate feedback | EVERYONE | 6-Sep |
| **2.a.3** |  |  | Submit final draft to message board Week 3 | Alan | 7-Sep |
| **2.a.4** |  |  | Receive professor's feedback and resubmit | Alan | 9-Sep |
| **2.a.5** |  |  | Submit project design to assignment folder | EVERYONE | 12-Sep |
| **2.b** |  | **User Guide** | |  | 12-Sep |
| **2.b.1** |  |  | Complete draft user guide | Jenna | 5-Sep |
| **2.b.2** |  |  | Gather team input and incorporate feedback | EVERYONE | 6-Sep |
| **2.b.3** |  |  | Submit final draft to message board Week 3 | Alan | 7-Sep |
| **2.b.4** |  |  | Receive professor's feedback and resubmit | Alan | 9-Sep |
| **2.b.5** |  |  | Submit project design to assignment folder | EVERYONE | 12-Sep |
|  | | | | | |
| **3 -** | **Complete text-based framework** | | | Framework team | 30-Aug |
| **3.a** |  | Clean up and comment throughout code for a polished final frame work | | Framework team | 30-Aug |
| **3.b** |  | Create classes for data structures that can translate into the GUI version | | Framework team | 30-Aug |
|  | | | | | |
| **4 -** | **Begin 2D game development** | | | Framework team | 15-Sep |
| **4.a** |  | Create of Java classes that will overlay on top of text-based framework to create 2D final product | | Framework team | 15-Sep |
| **4.b** |  | Incorporate UI team products and ideas into 2D implementation | | Framework team | 15-Sep |
| **4.c** |  | Implement sound assets | | Framework team | 15-Sep |
| **4.d** |  | Implement graphics assets | | Framework team | 15-Sep |
| **4.e** |  | Ensure UI displays correctly | | Framework team | 15-Sep |
| **4.f** |  | Correct scoring to implement tie-breaker rules | | Framework team | 15-Sep |
|  | | | | | |
| **5 -** | **Begin UI / UX development** | | | UI / UX team | 26-Aug |
| **5.a** |  | Create player number selection UI | | UI / UX team | 26-Aug |
| **5.b** |  | Create player name input UI | | UI / UX team | 26-Aug |
| **5.c** |  | Create gameplay UI | | UI / UX team | 26-Aug |
| **5.d** |  | Create final score screen UI | | UI / UX team | 26-Aug |
| **5.e** |  | Create buttons | | UI / UX team | 26-Aug |
| **5.f** |  | Provide suggestions for UI implementations to Framework Team | | UI / UX team | 26-Aug |
|  | | | | | |
| **6 -** | **Complete sound creation** | | | Jenna | 15-Sep |
| **6.a** |  | Create opening title screen music | | Jenna | 15-Sep |
| **6.b** |  | Create sound that fires when an option is selected | | Jenna | 15-Sep |
| **6.c** |  | Create sound for placing domino on game board | | Jenna | 15-Sep |
| **6.d** |  | Create a sound for when a domino is moved over the game board | | Jenna | 15-Sep |
| **6.e** |  | Create a sound for when a domino is discarded | | Jenna | 15-Sep |
| **6.f** |  | Create music that plays when game ends | | Jenna | 15-Sep |
| **6.e** |  | Create other sound resources as required | | Jenna | 15-Sep |
|  | | | | | |
| **7 -** | **Complete graphics creation** | | | Michael | 15-Sep |
| **7.a** |  | Create button images for domino actions (rotate clockwise, move up, etc.) | | Michael | 15-Sep |
| **7.b** |  | Create other images as required | | Michael | 15-Sep |
|  | | | | | |
| **8 -** | **Submit Phase 1 source** | | | Alan | 20-Sep |
| **8.a** |  | Draft response to Week 5 Discussion post | | Alan | 14-Sep |
| **8.b** |  | Gather team input and incorporate feedback | | EVERYONE | 16-Sep |
| **8.c** |  | Submit final draft to message board | | Alan | 18-Sep |
| **8.d** |  | Submit Phase 1 Source to project folder | | EVERYONE | 20-Sep |

# Problems Encountered

Our team expected to face a larger problem set for Phase II as we shifted from a text-based board game to a UI-based one. However, we understood that by establishing a solid framework, we would be able to weather any potential issues. While we set out with a solid development plan and ideas on how we would like to implement Phase II of our project, we did encounter a few problems that changed our overall trajectory. Nevertheless, we still were still able to meet all of our Phase II task deadlines because of the hard work and flexibility of our team members. Below is a list of the problems we encountered while developing our project.

## Falling Behind Schedule

Admittedly, our team had not had a respectable appreciation for the amount of work it might take to deliver something like a digital board game, and thus we decided to expand our project development phase at the onset, shifting our Phase I start left and expanding all three phases throughout this new extended timeline. Nevertheless, for Phase II, we still found ourselves missing some of the internal task deadlines as we shifted energies and focuses while some areas found themselves lacking adequate support. Still, the team pulled through and ultimately finished all of the tasks required for Phase II of our project.

## Java Kind of Sucks for Games

We set out with lofty expectations for what our end product would be. Luckily, most of us watched the [Expectations vs. Reality](https://learn.umgc.edu/d2l/le/news/923430/3119878/view?ou=923430) video that was posted earlier in our course. Many of the things that we thought would be simple (image manipulation, playing sounds, stopping sounds, etc.) proved to be even more difficult than we could have imagined. Much of the time developing the 2D framework involved troubleshooting mundane issues that often took more time and research than they may have been worth to fix and include. Ultimately, we came to the conclusion that without installing additional, non-standard libraries, Java just sucks for games in general. We continue to stick with that hypothesis as we embark into Phase III.

## Managing Java Layouts

We envisioned that our game would implement various types of animations and screens that would carry the option menus and game play itself. While some of the Java layout types area easy to use, others (e.g., GridBagLayout) are more difficult. The difficulty level was further compounded by the fact that we often could not use just one layout manager, but a nesting of layout managers to get the desired look that we wanted. Furthermore, our decision to use an image array to display the game board had to be taken into account for layout managers that employed it due to how it needed to be displayed. If there was a mistake in the configuration of a layout manager at any level, it would throw off all other levels. We found ourselves needing to do a lot of research in order to achieve even small successes with our use of layout managers.

# Re-evaluation of Decisions

Although we succeeded in accomplishing our milestones for Phase II, there were a number of decisions we had to make and then re-evaluate before we could embark on Phase III. Below is a list of the decisions we have made thus far and a re-evaluation of each.

## The Framework is Immutable

We assumed that if we established an excellent framework of classes and methods that could run the entire game as a text-based version, then applying a 2D overlay / extension to those classes would be easy. At the same time, if we were to build off of the base framework, then we also could not change the base framework because this would have cascading effects across our 2D overlay. Therefore, we decided that we would not change any of the base framework; if adjustments needed to be made, we would create a child / extension of the base framework class and apply those adjustments there. For instance, if Player.java did not have a GUIGameBoard attribute, then we could create GUIPlayer.java, an extension of Player.java, that would contain a GUIGameBoard attribute. This proved to be successful for our team throughout Phase II.

## Developing a Cartoony Design Style

We went back and forth on several types of styles we wanted to present the game with. Should we choose something cartoony? Or should we go with something sleeker and more professional? One of our targets was ensuring that the UI did not look like a typical UI. We wanted to expound on the base Java UI and give the display a more enjoyable feel. Ultimately, we decided to lean into the cartoon style during the middle of Phase II. We knew that our group’s overall inexperience with this type of project preventing us from fully realizing that design approach, but we committed to it nonetheless. We figured it would be okay because Phase III would give us the time needed to “beautify” our design.

## Using a UI Creator vs. Hand-Coded UI

In the beginning, we attempted to incorporate UI that was made with a UI Creator. This could have ultimately saved a lot of time and heartache as working with Java UI layouts is precarious at best. However, we found that the UI created by an application does it in a way such that the underlying Java code is difficult to read by a human. Understanding that this may complicate our process of implementing these machine-generated UIs, we reverted to coding the UI manually. While this took a bit more time, we found that this made it easier to manipulate the UI and mold it to our specifications.

## How to Represent the Player’s Game Board

Another decision at the start of Phase II we had to lock in was how would we go about representing the player’s game board. While there were many ways to approach this problem, we reduced it down to two courses of action: the game board would either be a group of game pieces, or the game board would just be an array of images that represent the game pieces. We went with the latter because we thought that we could do more with images if we decided to add in animations for Phase III.