

# Personal Project Report

## - Mobile Application Lab 2023 -

Student ID : 2018312280

Student Name : 이상수

Application Name
Yu-Gi-Oh fan game
Application Explanation
<ul style="list-style-type: none"><li>- What your application?</li><li>- App usage scenario</li></ul>
유희왕 게임 어플리케이션 1. 유희왕 카드 검색 2. 유희왕 카드로 덱을 구성 (아직 제작중인 기능) : 3. 블루투스를 활용해 주변 사람과 1:1 매치
Activity Explanation
<ul style="list-style-type: none"><li>- What activities is in your application?</li><li>- Role of each activity</li><li>- Design of each activity in aspect of UI/UX</li></ul>
1. Dictionary Activity: 유저가 영어이름을 사용해 원하는 카드를 open api와의 인터넷 연결로 검색해 확인할 수 있습니다. 2. editdeck activity: 유저가 게임에서 사용할 덱을 구축할 수 있습니다. 덱은 어플 종료 후에도 데이터가 유지되며, 원하는 경우에 저장할 수 있습니다. 덱의 매수 제한은 30장, 최소 20장이 게임에 사용 가능하기에 30장을 넘으면 덱에 카드 추가가 불가능하고, 20장 밑이면 이에 대해 경고도 전달됩니다. 같은 이름의 카드는 최대 3장까지 덱에 투입 가능하기에 그 이상 투입하면 경고와 함께 투입이 불가능 합니다. 카드들의 자세한 정보는 사진 이미지 칸 옆에 이름 같은 칸을 누르면 확인 가능합니다. 카드 이미지를 누르면 덱에 투입하거나, 덱에서 제외가 가능합니다. 3. MainActivity: 카드 검색, 덱 구성, 블루투스 연결, 게임 시작 등으로 이어주는 어플 시작 시 나타나는 메인 화면입니다.  (아직 제작중인 기능) : 4. InGameActivity : 상대 유저와 대결하는 기능이 있습니다. Draw, Standby, Main, Battle, End 페이지가 존재합니다. 구현 상태에 따라 시작 후 자동으로 턴이 넘어 갈 수 있습니다. (Draw, standby, end 페이지는 이 타이밍에 발동하는 효과가 존재하는 카드가 구현되지 않았으면 아직 유저가 컨트롤할 이유

가 없기에 자동으로 넘어가질 수 있습니다.) 이외의 Main, Battle페이지는 어떤 카드들만 구현 되더라도 높은 확률로 유저가 작업할 일이 존재하기에 유저가 가운데 턴 버튼을 누르면 넘어 가지도록 구현될 것입니다.

또한 드로우, 공격, 일반/특수 소환/ 세트 기능이 구현되어야 하며, 표시형식은 일반 마법 카드, 지속 마법 카드, 필드 마법 카드만 특별한 일이 없을 경우 세트 한 턴에도 사용 가능하며 몬스터, 속공마법, 함정카드는 소환, 세트한 턴 형식 변경이 불가능 합니다. 표시형식 변경은 몬스터의 경우 특별한 일이 없으면 1몬스터당 1턴에 1번만 가능합니다.

게임 시작 시 패 매수는 4장, 라이프는 4000, 덱 셔플, 다양한 카드들의 각각의 고유한 효과의 발동이나 지속, 1턴에 1번 몬스터 일반소환 또는 세트 제한, 필드 마법이 존재하는 상태에서 필드 마법을 발동하면 이미 존재하는 필드 마법은 묘지로, 필드에 몬스터존, 마법 존이 부족하면 필드에 카드 소환이나 세트 발동이 불가능. 0라이프 포인트 계산, 승리와 패배조건 (라이프가 0 또는 덱에 카드가 0장인 상태에서 드로우 실행시),

end 페이지시 패 매수가 6장을 넘을 경우 넘는 만큼 패 버리기,

묘지나 제외 존 확인 기능,

필드, 묘지, 제외 존, 패의 카드에 대한 자세한 정보를 확인하는 기능,

효과의 대상을 선택하는 기능,

공격의 대상을 선택하는 기능,

몬스터의 파괴, 카드의 사용시 묘지로 보내질 카드를 묘지로 보내는 기능

그리고 상대와 블루투스를 기반으로 소통하며 서로 하는 행동에 따라 서로 상대에게 신호를 주어 상대의 필드와 상태를 업데이트 하는 기능 등이 있습니다.

대부분의 카드들의 기능은 메인페이지에 사용이 가능하고, 배틀페이지에만 공격이 가능합니다. 그리고 어떠한 일이 발생해서 필드나 묘지, 패, 제외존에 변화가 생기면 그 때 단순히 카드의 내용을 확인하거나 묘지, 제외존의 카드들을 확인하는 등의 기능은 비활성화 시켰습니다. 이는 어떠한 업데이트가 일어날지 카드들과 상황들에 따라 매우 다양한 경우의 수가 존재하기에 그 때 실행중이던 다른 활동들을 비활성화 시키므로써 사건들의 일관성을 유지시키기 위한 목적 입니다. 그러지 않으면 묘지에 카드가 보내졌는데 유저가 확인하는 목록에는 묘지에 있어야 할 카드가 없다 거나 하는 경우의 수가 생길 수 있습니다.

#### API Usage

- From which point your application connect with Open API(or your own server)?
- Why you used it?

Dictionary 기능에 사용됩니다.

유저가 특정 카드에 대해 정보가 궁금한데, 게임에 아직 구현 되어있지 않거나 간단하게 확인 하고 싶을 때 바로 검색하여 그 정보를 확인할 수 있게 해줍니다. 당연히 확인만 가능하지 저 장은 할 수 없는 것이 저장을 해도 그 카드의 기능이 구현되지 못하기에 사용 불가능하여 저 장은 의미가 없어 카드의 정보 확인용으로만 사용됩니다.

### Implementation Challenge

- From where you took time while implementing your application?
- What is your special-care-point?  
(code optimization, user-friendly UI, ... / specify code file&line or activity name)

일단 외부 파일에 텍과 게임상 존재하는 카드들의 데이터를 저장해 두어야 했기에 이에 대해 알아보고 외부 파일과 소통하는 기능을 만들어야 했던 부분이 좀 힘들었습니다.

이 앱의 특별히 힘든 일은 위에 activity 설명에 보시면 알겠지만 유희왕의 방대한 규칙들, 카드들의 수많은 효과, 그리고 상대 유저와의 소통을 통한 서로 간의 필드와 상태에 대한 일관성 유지가 힘들었습니다. 유저간의 소통은 제작중이지만 특히 저 수많은 룰들을 문제없게 적용하는데 많은 어려움이 따르고 시간도 많이 걸려서 카드들의 기능 제작은 뒤로 미루어질 수밖에 없었습니다.

또한 game thread를 직접 만들어서 게임을 제작해본 것이 처음이라 UI를 써야하는지 Gameview를 써야하는지 판단을 제대로 못했는데, 많은 작업을 이미 해버린 지금 다시 생각해 보면 game thread를 제대로 활용하지 못한 것 같습니다.

묘지에서 작업이 일어나고, 필드에서 작업이 일어나고, 손에서 작업이 일어나고, 그 작업이 효과의 발동인지 그냥 확인인지 공격인지 이러한 부분들을 gamethread에서 처리를 해야 하는지 ingameactivity에서 처리를 해야 하는지 앱을 만들어본 경험 없이 그때그때 기능을 만들다 보니 앱의 기능 구현 자체가 매번 부담으로 다가온 것 같습니다.

### Application Design

- Overall design of your android application
- It can be design pattern, code separation, or so on.

일단 대부분의 기능들은 각각 다른 activity에서 작동되게 해주었는데, 특별한 점은 GameThread를 나누어서 여기에서 인게임상 활동이 처리되게 하였습니다. 특히 정말 다양한 작업이 일어날 수 있는 게임 이이기에 한 번에 다양한 작업이 겹치거나 하여 일관성이 문제가 되거나 데이터에 손실이 생기는 것을 방지하기가 중요합니다. (예를 들어 공격을 실행하는 도중에 일반 소환을 한다든가, 상대 턴에 상대가 활동 중인데 내가 할 수 없는 활동을 시행한다든가 하는 문제가 발생 할 수 있습니다.)

### Server Design (Optional)

- (Write only you used your own server like AWS) Overall design of your server
- It can be design pattern, code separation, or so on.

- You can freely add the row in the table above.

게임에 사용된 데이터들과 이미지들의 출처가 있습니다.

[1] KONAMI. 유희왕 OFFICIAL CARD GAME CARD DATABASE. 2020

[https://www.db.yugioh-card.com/yugiohdb/?request\\_locale=ko](https://www.db.yugioh-card.com/yugiohdb/?request_locale=ko)

[2] github users. Yu-Gi-Oh! API by YGOPRODeck. 2023

<https://ygoprodeck.com/api-guide/>

[3] Kenney. OpenGameArt.org. February 24m 2014.

<https://opengameart.org/content/ui-pack>

[4] wallpaper safari . Void Backgrounds. Sep 16, 2019

<https://wallpapersafari.com/w/c1iT9a>

[5] Rush Duel. Silver-Cards news. January, 01 2022.

<https://silver-cards.de/news/rushduel/>

[6] Space Star Wallpapers Mobile. WallpaperCave. 2023

<https://wallpapercave.com/space-star-wallpaper-mobile>

[7] Yu-Gi-Oh! Phone Wallpapers. WallpaperAccess. 2023

<https://wallpaperaccess.com/yu-gi-oh-phone>

#### 참고 코드

[1] Coil: url을 활용한 이미지 다운

<https://coil-kt.github.io/coil/>

[2] Create Android Game using Kotlin

<https://www.tutorialkart.com/kotlin-android/get-started-with-android-game-development/>