

# Design of a tactical turn-based game for mobile devices

### Ricardo André Machado da Silva

Thesis to obtain the Master of Science Degree in

### Information Systems and Computer Engineering

Supervisors: Prof. Carlos António Roque Martinho, Prof. Pedro Alexandre Simões dos Santos

### **Examination Committee**

Chairperson: Prof. Ernesto José Marques Morgado Supervisor: Prof. Carlos António Roque Martinho Member of the Committee: Prof. Mário Rui Fonseca dos Santos Gomes

October 2016

### Acknowledgments

- I would like to thank everyone that supported me through my 24 years of existence.
- To my family, for being present from the start.
- To my friends, for allowing me to decompress during hard times.
- To my mother, for providing me with everything I ever needed.
- To my girlfriend, for putting up with my constant tantrums and breakdowns.
- To my professors, for clearing all my doubts and guiding me to the right direction.

### Abstract

With the mobile games market growing by the day, more and more gamers are shifting to mobile platforms. Most of the games, however, are very simplistic, which leads to non interesting gameplay for veteran players, or are ports of preexisting games, which leads to poor interfaces and user experience. This project focuses on the second problem, trying to understand what the best user interface/user experience is in tactical turn based games for mobile platforms.

### **Keywords**

Tactical, Turn-based, Mobile devices, Strategy, Mobile game, User experience, User interface

### Resumo

Com o mercado dos jogos móveis a crescer a um ritmo elevado, mais e mais jogadores estão a aderir a plataformas móveis. No entanto, a maior parte dos jogos ou são bastante simplistas, o que levam a jogabilidade pouco interessante para jogadores experientes, ou são *ports* de jogos pré-existentes noutras plataformas, o que leva a más interfaces e experiências de utilizador. Este projecto foca-se no segundo problema, tentando perceber qual a melhor interface/experiência para os utilizadores num jogo táctico por turnos para plataformas móveis.

### **Palavras Chave**

Táctico, Turnos, Dispositivos móveis, Estratégia, Jogo móvel, Experiência para o utilizador, Interface para o utilizador

## **Table of Contents**

Ac	knov	vledgm	ents			
Abstract						
Resumo						
Та	ble o	f Conte	ents			
Li	st of I	Figures	six			
Li	st of <sup>.</sup>	Tables	xi			
Li	stof	Acrony	ms			
1	Intro	oductio	n			
	1.1	Motiva	tion			
	1.2	Object	ives			
	1.3	Outline	9			
2	Rela	ated wo	rk			
	2.1	Tactics	s vs strategy			
	2.2	Turn b	ased types			
		2.2.1	Classic turn based			
		2.2.2	Active Time Battle (ATB)			
		2.2.3	Simultaneous execution			
	2.3	Game	examples			
	2.4	Interes	sting mechanics			
		2.4.1	Use space			
		2.4.2	Variable terrain			
		2.4.3	Dynamic environment			
		2.4.4	Directional facing			
		2.4.5	Several controllable characters			
		2.4.6	Specialize the characters			
		2.4.7	Variable distance			
		2.4.8	Resource management			
		2.4.9	Multiple attack options			
		2.4.10	Support multiple objectives			
		2.4.11	Delayed attacks/Automatic actions			
		2.4.12	Character revival			

		2.4.13	$Conclusion \ . \ . \ . \ . \ . \ . \ . \ . \ . \ $				 	15
	2.5	Mobile	constraints				 	15
		2.5.1	Hardware capability				 	16
		2.5.2	Display size				 	16
		2.5.3	Control schemes				 	17
		2.5.4	Interface design				 	17
		2.5.5	Environmental constraints				 	17
3	Prop	bosed s	solution				 	19
	3.1	Backst	tory				 	19
	3.2	Game	mechanics				 	19
	3.3	The m	ap				 	20
	3.4	Units .					 	21
	3.5	Unit m	ovement				 	
	3.6	Comba	at				 	23
	3.7	Items					 	23
	3.8	Availal	ole actions				 	24
		3.8.1	Special actions				 	25
	3.9	A turn					 	25
	3.10	Progre	ession				 	
	3.11	Unique	e selling points				 	27
	3.12	Replay	vability/Player retention				 	
	3.13	Discus	sion				 	29
	3.14	Techno	ology used				 	29
4	Deve	eloped	work				 	31
	4.1	New s	olution				 	31
		4.1.1	Control mode				 	33
		4.1.2	Basic range mode				 	34
		4.1.3	Auto attack mode				 	35
5	Use	r tests					 	37
	5.1	Setup					 	
	5.2	Result	s				 	
	5.3	Discus	sion				 	46
6	Con	clusior	1				 	49
	6.1	Future	work				 	50
Bibliography							53	
Ap	penc	dix A (	Gameplay summary tables				 	<b>A-1</b>
Aŗ	penc	lix Bl	Jser questionnaire				 	<b>B</b> -1

## **List of Figures**

2.1	Different phases of a game: (a) Strategic phase, (b) Tactical phase	3
2.2	Different strategies in a Real Time Strategy (RTS): (a) Rushing, (b) Turtling	4
2.3	A classic Turn-based (TB) example.	5
2.4	An ATB example.	6
2.5	An example of XCOM's combat.	7
2.6	An example of Fire Emblem's combat.	8
2.7	A screenshot from an Advance Wars game where different terrain types are visible	10
2.8	Fog of War in Fire Emblem	10
2.9	An unit in XCOM taking cover	11
2.10	A list of the playable characters in Fire Emblem	12
2.11	Different class abilities on XCOM: (a) Sniper class, (b) Assault class	13
2.12	A "meld canister" in XCOM, a secondary objective present in some of the missions. $\ . \ .$	14
4.1	An image where the vulnerability indicator is visible. If the airplane moves to the gray	
	squares it will be in attack range from at least one enemy unit.	33
4.2	Input method 1	34
4.3	Fire Emblem's movement and range indicator	34
4.4	Input method 2	35
4.5	Input method 3	36
5.1	Control methods preference	38
5.2	Summary of the differences between the different modes using Wilcoxon's signed-rank	
	test	39
5.3	Bar graph showing the user distribution amongst different play modes	40
5.4	Box-plots depicting the evaluation users gave to the different play modes	40
5.5	Box-plots depicting the evaluation users gave to the reach and quickness questions $\ . \ .$	42
5.6	Summary of the statistically difference between experienced and novice players $\ldots$ .	44
5.7	Box-plots depicting different evaluation depending on the users experience with tactical	
	games	44
5.8	Comparison between mode 1(a) and mode 2(b). In (a), the user knows that the tank	
	can hit the soldier after moving, while in (b) the users has to count the number of spaces	
	to move	47
5.9	Auto attack mode attack example.	48

## **List of Tables**

A.1	Different attributes and what they influence	A-1
A.2	Types of spaces and their interaction with movement and combat	A-1
A.3	Different specializations and their characteristics	A-1
A.4	Available special actions unlocked via level-up	A-1
A.5	Consumable items	A-2
A.6	List of available equipments	A-3

## **List of Acronyms**

FF Final Fantasy
ATB Active Time Battle
RTS Real Time Strategy
<b>UP</b> Upgrade Points
exp experience
Ran Range
Eva Evasion
Acc Accuracy
Def Defence
Spd Speed
Atk Attack
UI User interface
HP Hit points
AP Action points
TB Turn-based
UI/UX user interface/user experience
MOJO MOntra de JOgos
AoE area of effect

## **1** Introduction

Even though there are more games being developed for the mobile market, the same cannot be said so much about tactical Turn-based (TB) games. While there are a few of them (even like XCOM and Final Fantasy tactics, that are games that both the public and press views as being very good) they're not really made *for* mobile devices and so they don't have the constraints in mind (Section 2.5), are just ports of pre existing games.

This is a case of games that are good on consoles/computers not fitting the target audience, being a game far too complex for the system in question.

While there are handheld consoles on the market, that on the surface face the same challenges as mobile games, that is not the fact, as handheld consoles have both a different audience (comprised more of hardcore/regular than casual gamers) as well as different control schemes, since the consoles have some form of physical input, unlike mobile devices.

The objective of this project is, therefore, to create a game that can suit the mobile market well, both in terms of interface design as well as game mechanics that suit the target audience, with a greater focus on the interface/user experience.

#### 1.1 Motivation

The mobile market, particularly in the videogames area, presents some unique challenges when developing for it. One of the main problems that developers are faced with is the user interface/user experience (UI/UX) design. This is because mobile devices have much smaller screen area than desktop ones. This can be seen from the fact that, in 2015, the most popular mobile physical screen size was 360x640 [1] while for desktop devices it was 1366x768 [2].

The screen size is also a challenge in handheld consoles, such as the Nintendo DS, Playstation Vita, etc, but the user interaction is solved from the fact that consoles, even though have less screen space, have some traditional input in the form of buttons and/or joysticks.

This is the main distinction from mobile devices, where the user has access to less inputs (with mouse and keyboard the user has access to at least 2 mouse buttons, dragging motion, hovering information and about 100 keyboard keys while with a mobile device the user can only tap and swipe) which creates an interesting challenge, as the developers have both less physical space to display information as well as less ways to capture the user input information.

#### 1.2 Objectives

The objective of this project is to study what is the best way to develop UI/UX for a tactical turn based game for mobile devices. In order to do so, a simple working prototype will be developed. The prototype will have some simple tactical TB mechanics and will be presented with three different control methods to the users.

The objective of this method is to present users with a very simple game, so that the focus can be on the UI/UX, while having a tactical TB background to test it in, using common actions present in the genre, such as selecting and controlling units, etc.

So, the project will be divided in two parts. The first one is deciding which mechanics to implement in order to make a meaningful game, without over saturating it, and thus make it confusing for the users. In order to do so, a study will be conducted on both core concepts (like difference between *strategy* and *tactics*, which is a common misconception) and interesting mechanics used throughout multiple games on the genre. This will allow for an educated decision on which mechanics to use and which to filter, thus creating an interesting, yet simple game.

The second part, after having the game decided, is to create an interface that can suit both the game and the mobile platform.

#### 1.3 Outline

In order to complete the objectives outlined in the previous sections, a study will have to be conducted on several areas:

To create an interesting game, we will look at multiple traditional mechanics from TB and tactical games, which will be done from Sections 2.1 to 2.4.

Then, since the game is supposed to appeal to mobile users, which have their own characteristics, an analysis on the mobile environment will be conducted in Section 2.5.

After that analysis is made, a proposed solution will be presented in Chapter 3. The implementation of the solution will then be described in Chapter 4, along with the problems that arose during it.

Finally, an analysis of the developed solution will be presented during Section 5, with some conclusions and proposals for future work in Chapter 6.

## 2 Related work

This chapter will start by defining some concepts core to the project. In section 2.1, the common misconception between tactics and strategy will be analyzed. Then, in section 2.2, multiple types of turn based mechanics will be described.

After defining those concepts, an analysis on multiple state of the art examples will be conducted during section 2.3, with a list of interesting mechanics derived from those games will be laid out on section 2.4.

Finally, the mobile constraints will be analysed in section 2.5.

#### 2.1 Tactics vs strategy

The first thing that was necessary to clearly define was the difference between strategy and tactics, because those are terms commonly used interchangeable, even though they have different meanings, both when used within game contexts as outside of them.

In a general way, and in its origin, strategy refers to high level decisions that are taken in *preparation* to a battle or encounter, while tactics refer to the decisions that are taken *during* said battle to try and execute the defined strategy[3]. In figure 2.1 we see both phases in action in the same game: in 5.8(a), the player needs to chose what countries to attack, what countries to be friends with, make trades, etc (strategic phase, because they are permanent decisions that impact the game as a whole) while in 5.8(b) the player has to employ tactics (like what units to use, etc) to be able to execute the strategy.



(a)

(b)

Figure 2.1: Different phases of a game: (a) Strategic phase, (b) Tactical phase.

For example, in a Real Time Strategy (RTS), two players may opt for two different strategies to win a battle: one may try to rush his opponent, while the other one may try to store more resources, research more technology and defend its base so that it can dominate during the late game with more advanced troops, commonly referred as "turtling" (see figure 2.2 for an example. In 2.2(a) it is possible

to see a lot of units swarming an enemy base while in 2.2(b) a lot of defensive buildings are placed to try and repel an enemy attack). This is their strategy for the battle. The tactics that each player will use would be something like:

- What buildings to build;
- In what order;
- What troops to train;
- Which and how many resources to store;
- When to start to attack the enemy;
- Where to move the available troops and how to organize them in the battlefield;
- etc.

Being that all the decisions taken have the objective of accomplishing the strategy previously selected.



(a)

(b)

Figure 2.2: Different strategies in a RTS: (a) Rushing, (b) Turtling.

This could be seen as "pre-battle" decisions, and in some instances it can be the case. For example, some games may have you build your empire (where you can decide what buildings to build, etc) but in other cases this decisions are done *during* the battle (when you decide if you want to build barracks to train troops, or if you want to build farms to gain more resources, etc. This being done during an encounter).

One easy way to distinguish between strategic from tactical decisions is to see if the decisions are permanent. As in, if what I build in one battle transposes into the next battle, then I'm being faced with strategic decisions. On the other way, if the decisions are atomic (i.e. only affect the current battle or encounter) I'm being faced with tactical decisions.

Being the objective of this project to develop a tactical game, it is now clear that it had to be a game focused on "low level" decisions, battle to battle, and not so much on decisions outside of the battles, that influence the game in a much more permanent way.

#### 2.2 Turn based types

In order to design a TB game, it was necessary to know what defines a TB game and what types of TB mechanics exist.

A TB game is defined for having its game flow divided into individual phases, where the game is paused, called turns. Depending on the TB variant used the turns progression may vary, but overall turns are used to give players more time to think about their actions and not focus so much on keeping track and controlling several things at once in real time [4].

#### 2.2.1 Classic turn based

In its classic form, a turn ends when one of the players executes the actions of all of the elements that he controls (figure 2.3). This allows for the player to have unlimited time to think about its actions, because the opponent only acts after the player acted. In multiplayer computer games, since the unlimited time system would be easily abusable, each player usually has a time limit to perform his turn (but it is usually more than enough so that the player can still think carefully about what he wants to do).



Figure 2.3: A classic TB example.

#### 2.2.2 Active Time Battle (ATB)

In this system, usually used in party based combats, each element of the party has an action bar that fills over time (in figure 2.4 we can see that each character has a bar with the action to be performed that fills up over time). When the bar is fully filled, that element executes the action that the player queued for him.



Figure 2.4: An ATB example.

This system also allows the player to have time to think about his actions, because the action bar takes a while to fill, but gives some sense of urgency, because if the bar gets completely full and that character doesn't have an attributed command, it will just stay still, wasting time and giving more opportunities to the enemy to act.

#### 2.2.3 Simultaneous execution

In this system, both players decide what they want to do at the same time, and when the turn advances, both players' inputs are executed simultaneously. Even if this looks like a minor variation, it leads to one very important (and major) difference in the way of thinking about the moves. While in the previous variations the player knows how the state of the system will be when his actions are executed, it can plan them with 100% confidence that his actions will occur how he ordered them to.

In simultaneous execution, however, the state of the system when the actions are executed is unknown to the player, so he has to plan them while trying to guess what his opponents will do and vice versa, which can lead to much more complex train of thoughts.

#### 2.3 Game examples

Before deciding what mechanics to implement in the game, it is important to evaluate other games within the same genre to see what works, what doesn't and what can be improved. The games covered next were chosen based on having simple mechanics that interact with each other in such a way that it creates a deep and rich tactical environment, where the players have multiple ways of approaching a specific problem. They are also games that the overall critic describes as being genre

defining, and so, seem like a good starting point.

Said mechanics will be broken down and discussed in more detail in section 2.4.

The first game that will be covered is XCOM, which is a game divided in two parts. One management (strategic) part, where the player must organize his base, build and research new technology, etc. with the objective of upgrading his troops into better ones.

The other part is the combat part, which is why this game was chosen. In XCOM, the player moves his units in a grid like map, and can both move and attack or sprint (move more squares in the grid, but preventing an attack in the same turn). Each unit also has some specific abilities, that depend on the class/weapons equipped, and for the most part, skills function like a regular attack, in the sense that the player can move and use an ability/attack or sprint and do nothing else. In figure 2.5 it is visible the radius of movement of the selected unit in the form of blue lines, as well as his special abilities in the bottom bar.



Figure 2.5: An example of XCOM's combat.

This game is an interesting example because there are a lot of skills and mechanics that occur during the enemies turn (most notably, the Overwatch skill [5]). This makes it so players have to think not only about their turns, but also about what the enemies will do and how their units will react.

Another good thing about XCOM is the way the skill tree is designed. Every soldier can be of one of 4 classes, each class having different skills that unlock as that unit levels up. One of the interesting facts about the skill tree is that in some cases the skill that the unit learns is mandatory, because it is class defining (a heavy unit *must* have some sort of skill that makes it tankier, or else it doesn't make much sense to be called "Heavy"). In most level ups, however, the player must chose between one of two skills. This is good because, if balanced properly, all skills should be equally desirable and if that happens one unit would have all the skills and be incredibly powerful. By forcing the player to chose between the two, it requires him to have several specialized units instead of just one powerful one [6]

The second example is the Fire Emblem series. It was mostly chosen because of it's grid based

combat, easily translated to mobile, and because it utilizes many techniques that can turn a tactics game much more interesting, like using space, having several controllable characters (each very specialized), different terrain types and dynamic environments. Each of these mechanics will be described in detail in section 2.4, but in figure 2.6 the different terrain types are visible, in the form of the squares with the trees in them, which impact how the combat plays out.



Figure 2.6: An example of Fire Emblem's combat.

However (and one of the reasons I picked this series as an example), one of the best things about the game is how the characters evolve and upgrade. It is a very natural and organic system, where characters *get better the more they do something*, and not by the player upgrading some attribute to certain character. What this means is that, if two characters are usually close to each other during fights, they'll get along better and perform better when put together. This can even evolve to romantic relationships outside of combat and some of those interactions leak into the storyline, with the close characters can joke with each other during some storyline sequences.

This is a good way of doing the progression, as the player sees their party grow up in a natural way that makes sense. If someone uses a character to constantly hit enemies from really far away it is expected that that characters eventually gets a really good accuracy out of practice. In usual games, if a player wants a character to have really good aim, he must level it and assign points to accuracy (or get gear that boosts accuracy, or something along those lines, but the player must, one way or the other, explicitly say "you have high accuracy"). In Fire Emblem this doesn't happen, and in my opinion (and of many others [7]) it is a good way of handling the matter as it is intuitive for the player.

#### 2.4 Interesting mechanics

In addition to the different types of TB, there exist several mechanics transverse to any kind of TB game. Those mechanics, if used correctly, can easily make a game's strategic space increase (but

one has to be careful as to not over saturate a game with mechanics, as that may make the game more complex than it is supposed to, leading to confusion amongst players).

All of the concepts presented in the next sections have the objective of making the game have the following qualities [8, 9]:

- Emergent complexity, meaning it creates complex gameplay out of a simple set of rules. This is opposed to having a game that's complex by having hundreds of pages of rules;
- Clarity, meaning the player has an immediate feedback and understanding of the result of its actions and decisions;
- Determinism, meaning that the game, even if it uses some sort of random, gives enough tactical tools and agency to the player, which results in proper strategies working the majority of the time.

#### 2.4.1 Use space

Having a spacial dimension in a game automatically creates a new complexity, without much of a burden to the player. That is because adding space enables the use of:

- Attack ranges: where different units can have different ranges, making the player think about where to position them;
- Fog of war: where some zones of the map can be opaque to the player, and he has to send units there to scout what's there;
- Blocking paths: where some zones of the map may be impassable to some or all of the units, creating choke points and ways for players to think about positioning as a tactical tool instead of just thinking about the units themselves.

This technique is more of a basis where many other mechanics can be implemented than a mechanic itself.

#### 2.4.2 Variable terrain

Terrain should be diversified, both in terms of positive/negative buffs on specific spaces (a character on a fortress should have more defense than an uncovered one, and walking through mud should be harder than through flat terrain) and in its design (a narrow and long bridge creates a nice choke-point, while a large plain creates space for spread out strategies).

If also coupled with classes specializations, more interesting dynamics can be created, by making certain classes have specific interactions with some terrain (maybe flying units can go over water but none of the others can).

However, this concept can be pushed even further by allowing the player itself to manipulate the terrain, by destroying/creating some paths (eliminating some passageway, creating choke points, etc) or by placing traps.

This would make it so that the player must think not only where to place his characters according to the terrain, but also how to change the terrain to place the characters where he wants them to be.

Games where the mechanic is used: Final Fantasy Tactics, Fire Emblem series and Advance Wars (different terrains have different movement and/or combat modifiers. See figure 2.7), XCOM series (some spaces can be poisoned/on fire, and height plays an important role on the combat).



Figure 2.7: A screenshot from an Advance Wars game where different terrain types are visible.

#### 2.4.3 Dynamic environment

This seemingly complex topic can actually be turned into a really simple one, but with great implications to the pace and strategic depth of a game, by having some hazardous zones and/or random item drops around the map.

If those zones aren't constant along the map it forces the player to move their units around, either to escape the dangerous zones or reach the "good zones" (where the items drop).



Figure 2.8: Fog of War in Fire Emblem

The Fire Emblem series in particular has a great implementation of this concept. Concerning the hazards, on some levels there is a fog of war that advances over time [10] and that can be seen on figure 2.8. In this fog, the player can't see the enemies' position. Only by advancing his characters (or

by using some special items/abilities) can the player scout the terrain ahead. This not only encourages the player to move but also touches the point discussed previously, "Resource management", because the scouting items are limited and the actions often use all of the player's turn.

Fire Emblem also has some "item drops" in the form of villages scattered across the battlefield [11]. These villages can be visited by the player (offering some rewards, varying from advices from the townspeople, to new weapons or even new characters) but can also be destroyed by the enemies if they reach them first. This heavily encourages the player to move fast to reach those places, because if the enemies go there first the reward will no longer be available, and while their locations don't vary nor are chosen randomly, it adds a lot of depth and decision making to each battle.

This feature also ties with the multiple objectives one, since visiting the villages is in no way necessary to proceeding with the story or winning the battles, but gives the player some nice optional objectives.

Games where the mechanic is used: Fire Emblem, XCOM (destructible cover).

#### 2.4.4 Directional facing

If the characters (both allied and foe) suffer more damage from the sides or the rear side, it forces the players to think where to turn their characters to, instead of just where to move them to.

This concept can be used both offensively and defensively, because the players must think where to face his characters so they don't take extra damage, but also how to place them so the enemy cant avoid taking extra damage from someone (pincer movement) [12].

Games where the mechanic is used: Final Fantasy Tactics (units hit from the back or the sides take more damage), XCOM (units can take cover, and that cover is directional as it only covers the units from certain angles, as seen in figure 2.9).



Figure 2.9: An unit in XCOM taking cover.

#### 2.4.5 Several controllable characters

The more characters a player has, the more it has to think about what to do with each one of them (an example can be seen in figure 2.10). Obviously, this creates more strategic diversity, because the number of actions a player can make increases exponentially with every new character available.



Figure 2.10: A list of the playable characters in Fire Emblem.

This, coupled with the space dimension, takes this concept even further, because the player has one more action to do with each character (where to place him).

#### 2.4.6 Specialize the characters

This one ties up nicely with the mechanic explained previously. Having a lot of generic characters isn't interesting (nor useful to the game's depth), because there isn't a reason to chose one character over another.

However this differentiation shouldn't just be a case of the characters having different stats, like damage or hit points, and should provide meaningful choices to the player in the form of special actions only available to certain classes. For example in Fire Emblem, some classes like the Bard are so specialized that they don't even have the "basic" actions available to them (like attack) and are used with an unique purpose (in the case of the Bard, it has an action that allows an ally to act twice in the same turn).

On the same note, enemies should also be varied and specialized, making the players think which of their characters' classes are good versus the enemies'.

Games where the mechanic is used: XCOM (figure 2.11), Fire Emblem, Final Fantasy tactics.



(a)

(b)

Figure 2.11: Different class abilities on XCOM: (a) Sniper class, (b) Assault class.

#### 2.4.7 Variable distance

This, simply put, means that the battles should have different initial states, with the characters in different distances from one another.

This makes the player think how many turns his ranged and melee characters will take to reach the enemy, instead of always being one turn away.

For example, if the player was ambushed, the battle could start with all the enemies really close to the player's units, which could pose a threat for more fragile units that usually try to stay far back. In another scenario, the player could be under attack from a long range artillery piece, forcing the player to look for safe spots all allong the way, etc.

Games where the mechanic is used: Fire Emblem series, Final Fantasy Tactics, Advance Wars, XCOM.

#### 2.4.8 Resource management

If a tactical game has some sort of resource management, it immediately creates some form of risk reward.

This resource could be anything, from mana necessary to cast spells, to number of charges on abilities/weapons, or even available Action points (AP) per turn, strong attacks taking more, but having a high chance to miss. There can also be consumable items that the player can use for some immediate benefit but are gone forever afterwards.

Having such a system would promote a risk/reward gameplay, not only because heavy attacks have a higher chance to miss, but because players can't move afterwards, since it probably used all its AP.

Games where the mechanic is used: Final Fantasy series (mana), XCOM series (cooldowns), Fire Emblem series (attack charges on weapons).

#### 2.4.9 Multiple attack options

This blends in nicely with the previous point, since different attack options could have different AP costs, differing not only on the strength/miss chance of the attack but also on having different effects

if they land. For example, maybe a regular attack could have an elemental modifier which increased the AP cost, but that dealt more damage to enemies with certain weaknesses.

This usually ties in with resource management, as stronger attacks commonly have a bigger drawback. In XCOM, attack with a high critical chance or area of effect (AoE) attacks usually are associated with higher miss chances and/or longer cooldowns. In Final Fantasy (FF), the player can unleash devastating attacks (for example in the form of summoning very powerful creatures to fight by his side) but such attack can only be used very rarely, as it takes several battles to recharge.

#### 2.4.10 Support multiple objectives

Multiple objectives, as in having multiple win conditions (defeat all the enemies, defeat the Boss, rescue someone, etc) already creates some nice strategic diversity and is pretty common place on most games.

However there is something else that can be done, and that is to give secondary, or optional objectives to the player. Those objectives can be something like "rescue a character in x or less turns to receive an extra reward", or maybe having some persistent stat on the players (like morale in XCOM [13]) that affects the characters' performance, and changes based on how the player handled the missions. Morale isn't necessary to completing the missions, but is a side objective in the sense that the player is trying to complete the missions objectives while also doing actions that boosts their character's morale.

Games where the mechanic is used: Fire Emblem (villages [11]), XCOM (meld canister [14]. See figure 2.12).



Figure 2.12: A "meld canister" in XCOM, a secondary objective present in some of the missions.

#### 2.4.11 Delayed attacks/Automatic actions

Delayed attacks are attacks (or actions) that occur even when it is not the player's turn to act.

This can be done in several ways. One of them is having a special action, like "stay on guard", that makes a character attack anything that comes in range in a certain radius and/or direction. This action could be different for each character. Maybe it makes more sense that the healer's action heals an ally if he gets below a certain Hit points (HP) threshold within its range.

Another way of doing delayed attacks is making it possible for the characters to counter attack when hit. However this can only be strategical and interesting if the characters can't retaliate under all circumstances. Melee units should only be able to attack at melee range and ranged units could only attack at a minimum range (but not melee). This way, if the player attacks a ranged unit with a melee one the ranged won't be able to respond to it. This creates another level of depth concerning character placement, since the player, when moving his units, has to take into consideration things like "will my character be able to be attacked without being able to counter attack?".

Games where the mechanic is used: Fire Emblem (characters attacked retaliate, even outside of their turns), some Final Fantasies (counter abilities [15], XCOM (overwatch, [5]).

#### 2.4.12 Character revival

In turn based games, usually character death is dealt with in one of two ways. Either when a character's HP reaches 0 it becomes inactive until the player can revive it later, or in some more extreme games, the character permanently dies and the player can't have it back.

Those approaches, however, don't give any real sense of urgency to the player. On the first case, the player can revive the character at any given time, and on the later one, when a character dies, it is gone for good, so there isn't anything the player can do at all.

A nice and simple way to make the decision making more interesting is by adding a "revive countdown" of sorts. With this approach, when a character's HP reaches 0, the player has a set number of turns to revive it, after which it dies permanently. This creates a new secondary objective (covered previously) because the player doesn't have to do it in order to complete the mission but it has a strong incentive to do so, while also giving the player an opportunity to recover from his mistakes.

Games where the mechanic is used: XCOM (characters that get critically injured have a set number of turns to be healed before they bleed out).

#### 2.4.13 Conclusion

Most of the games covered in Section 2.3 use at least one of these techniques to create a more interesting and rich gameplay. However, none uses all of them, which is probably for the best, since using everything would make the gameplay too confusing and with dozens of rules for the player to remember.

For those reasons, the game examples, coupled with the mechanics described previously, are a good starting point for a new game, having in consideration the mobile constraints, which will be described in the next section.

#### 2.5 Mobile constraints

Developing games for mobile platforms is really different than developing a game (even when it is the same game, which is the case in ports) for consoles or computers.

There are multiple reasons for that, which will be described in detail in the following sections, but they are:

- Hardware capability;
- Display size;
- Control scheme;
- Interface
- Environmental constraints.

For those reasons, developing a game for mobile should be made carefully and have the design decisions be made with the constraints in mind, and shouldn't be thought as just developing a game that happens to be played in a smaller screen.

#### 2.5.1 Hardware capability

With smartphones/tablets being such small devices, their processing power isn't their strongest point, and so, games have to have those factors in consideration, both through code optimization since the beginning of the development, and also by extensively testing in several devices with different hardware to see if the game runs without lag or fps drops. There is a plethora of services that provide testing in multiple devices and operating systems, to ensure that the game runs in as many devices and configurations as possible. In the Android's developer web page there are even a couple of requisites regarding the performance of the apps that go into the play store: "App loads quickly or provides onscreen feedback to the user (a progress indicator or similar cue) if the app takes longer than two seconds to load." and "With StrictMode enabled (see StrictMode Testing, below), no red flashes (performance warnings from StrictMode) are visible when exercising the app, including during game play, animations and UI transitions, and any other part of the app." [16] <sup>1</sup>.

Also, the space required to both download and install a game should be taken into consideration, since most of the time the game will be downloaded over cellular networks. This is even enforced, for example, by Apple's App Store guidelines for content put up in the app store ("Apps larger than 100MB in size will not download over cellular networks (this is automatically prohibited by the App Store" [17]).

#### 2.5.2 Display size

The mobile devices displays present two problems. First, in most cases, the resolution is fairly small, which limits the amount of content one can fit into the screen at a given time (which includes any menus and interface elements, in addition to the gameplay ones).

Secondly, the aspect ratio varies between different devices, ranging from 3:2 on the iPhone 4 to 17:10 on Samsung Galaxy Tab 7 and it is necessary to have them into account. This can have serious

<sup>&</sup>lt;sup>1</sup>Like Applause, for example, http://www.applause.com/mobile-app-testing

implications in the gameplay, for example on games that need the users to quickly react to an object that enters the scene, like runners. If two users are playing on different devices, one in a tablet (4:3 resolution) and another one on a phone (16:9 resolution), and if the game is poorly made, the phone user could have an advantage because he would be able to see farther then the one in the tablet.

The aspect ratio factor is also really important in a tactical turn based game, because if some users can see more of the battlefield than others, that creates a clear unfair advantage.

#### 2.5.3 Control schemes

Controls in mobile games present a great challenge, because it can be really tricky to create intuitive controls with only a few possible inputs (the only three things recognized by a touchscreen are touch begin, touch end and touch move, while with a mouse there is also one extra "action", the hover). It is possible to derive some basic actions with those inputs (a touch begin closely followed by a touch end is a tap, a touch begin followed by a touch move followed by a touch end is a swipe, etc) but it is still hard to create a simple and understandable control scheme with those basic "control primitives".

For example, Super Mario Bros had really simple controls: two directional buttons, one button to accelerate and one button to jump, but even that would be tricky to replicate in a touchscreen (Gamasutra has an article on touchscreen controls, where the author gives a nice step by step description on progressively more complex controls [18]).

However, in a turn based game, most of the controls are menu based, so that won't be as much of a problem.

#### 2.5.4 Interface design

As discussed previously, the devices' display size may be (and in most cases is) very small. This creates a huge problem in terms of User interface (UI) design, because it can compete with space that could be used for gameplay elements.

As with any UI, it should be made incrementally, but with some mobile related constraints (like resolution [19], number of buttons per screen, information per screen, etc [20])<sup>2</sup>.

Apart from those specifics, the UI should be created following the usual rules.

#### 2.5.5 Environmental constraints

Mobile games are played mostly while commuting and at home, in bed [21] and with an average playsession duration of 6.8 minutes [22]. Those two factor combined lead to two considerations:

Games, even if with a multiplayer component, should allow for offline play. This is because
if someone is playing outdoors, or while traveling, may not have a good internet connection,
and/or may need to stop their gameplay midway, so if the game is purely online it is losing on a
big percentage of potential players.

 $<sup>^2</sup> This$  is a website with examples of UI from several mobile games:  $\tt http://www.game-patterns.com/,$  accessed at Monday 21st November, 2016

- Games should allow short playsessions. This means that a game "match" (either a level of a puzzle game, or a battle of a tactical TB game) should be short, so that the player can play one entire match in a short period.
- Games should also be careful with the usage of sound, as many people play mobile games while traveling, so they may not have the sound enabled. This means that sound should be used as a support and not as a fundamental part of the game.

## **3 Proposed solution**

The game I propose is "Dawn of Revolution" and is a single player tactical TB game with a level based campaign and player vs player matches.

The game will be developed for the mobile market and targets both tactical games fans as well as newcomers that are trying to get into the genre. This will be achieved by fulfilling some design goals:

- 1. Simple base mechanics: so that everyone can understand the simple rules of the game, such has how to move, how to fight, etc;
- Emergent complexity: so that more experiences players can develop advanced strategies that derive from the simple base rules
- 3. Fast matches: being developed for the mobile market, and because of the constraints presented in section 2.5, it is necessary that the game allows for short playsessions.

#### 3.1 Backstory

The game is set in the future, where humans have developed highly advanced combat technology, and have access to heavy weaponry. Unfortunately, one man, Baron Rivendere, has been developing his secret army, in an attempt to be the most powerful man on Earth. He started his advance and turned everything into a giant wasteland, creeping with debris and destruction everywhere. The player will step into the shoes of General Sharr Brand, viewed by many as the greatest tactician alive.

Then, by completing several missions and building his own army, the player will try to claim all that Rivendere took for himself.

#### 3.2 Game mechanics

In order for the game to have any sort of strategic complexity, at the least it is necessary to use the space (Section 2.4), because so many other mechanics need this one to be implemented.

Several aspects of the game mechanics and progression will be detailed in the next sections, and will be:

- Grid map with simple movement rules (design goal 1): Section 3.3;
- Standard attributes per unit so that the game is easy to pickup (design goal 1): Section 3.4;
- Unit specialization so that more experienced players can customize their party to have a particular playstyle, or have different units to suit different needs (design goal 2): Section 3.4;
- Special actions, most notably carry, allow for complex turns and deep strategies (design goal 2): Sections 3.4 and 3.8.

• Secondary objectives that make the missions harder but give more rewards, so that the game can be balanced towards both new and experienced players, as well as accommodating different playsession lengths (design goals 1, 2 and 3): Section 3.10.

Appendix A has a few summarizing tables with several examples of units, terrains, items and special abilities, that try to include examples of initial, mid game and end game content.

It is important to note that all the values and units/items presented here are just listed as examples to better understand the progression and what sort of actions will be possible and so are subject to change when being implemented and tested.

#### 3.3 The map

Since the mobile devices have the constraints presented in Section 2.5, a grid like map with squared spaces seems to be the best solution, since it is easier for the player to click where he wants to. This layout is pretty standard across this type of game, and suits the mobile environment pretty well. Alternatives could be an "open map", where the player can send his units to wherever he wants them to go, pretty much like in a RTS, but with turn constraints. However, this could lead to the players misclicking the screen by a few pixels, which could be the difference between being in range for an attack or not.

Plus, it is easier to balance around the multiple resolutions that devices can have, because it can be easily defined how many spaces should be visible at any given time, and then, no matter what resolution the game is currently running in, all devices show exactly the same amount of information, making it fair for every player.

That said, the map will be composed of several squared spaces, where every space can be of a different type. Every type of space defines a set of special rules regarding unit movement and combat.

The different spaces (which rules will be defined in the next sections, with a summary table at the end) will be:

- Normal space;
- Walls;
- Water/hole;
- Mud/swamp;
- Forest;
- Shielded spaces.

There is also one particular space that can appear at random in any map, dropped crates. Crates don't appear when the level starts, but rather can be dropped at any time during any mission. Crates can contain either money or items (Section 3.7). Those crates belong either to the player or the enemy's faction, and any unit can reach and interact with them. If the unit belongs to the same faction
as the crate, it can open it and get whatever is inside, otherwise it can destroy it, preventing any other unit from getting its content.

# 3.4 Units

All of the units will have a set of attributes that define how they behave during the missions:

- HP: how much damage a unit can take before it is taken down. Can only be replenished by special items or abilities and does not reset to maximum in the beginning of a battle (meaning damage taken takes over from one battle to another).
- AP: points a unit has available to perform special actions. It replenishes over time and is set to maximum at the beginning of a battle;
- Speed (Spd): how many spaces a unit can move during one turn;
- Attack (Atk): how much damage the unit deals when it successfully hits an enemy;
- Defence (Def): how much damage the unit absorbs when it is hit by an enemy;
- Range (Ran): where a unit can attack (a Ran = 3 doesn't mean that that unit can attack at range 1, 2 and 3. It means it can *only attack* at range 3). This attribute can be one or more numbers and it is the only attribute that isn't specific to a unit, but to the weapon that unit is carrying. If it is more than one it means that the unit has several spaces where it can attack;
- Accuracy (Acc) and Evasion (Eva): if a unit being attacked has higher Eva than the attackers Acc, it has a chance to dodge the attack. If the attackers Acc is higher than the defender's Eva, then it has a chance to critically strike for double damage.

Most of these attributes are pretty simple and are what would be expected from a tactical TB game and are so in order to fulfill the first design goal, of having simple rules and being easy to pickup. The only exception to that case is the Acc/Eva interaction. Usually units have two attributes, hit chance and critical chance, which are independent (sometimes hit chance isn't present and units always hit when in range) but conceptually it makes more sense that a unit that has a greater chance to hit also has a greater chance to crit.

Besides the attributes that all units have, they can be of one of three different general types:

- Infantry units: default units, jack of all trades, masters of none;
- Land vehicles: have greater Spd, but are more slowed down by movement modifiers (explained later);
- Air vehicles: the more mobile units, aren't slowed down by any modifiers and can cross almost all kinds of terrain (explained later) but aren't as suited for combat.

Both types of vehicle can carry one unit on foot with them and drop them somewhere else, not consuming the carried unit's turn.

Every type of unit can eventually develop one of 3 specializations (based on their original type) that increase some stats while decreasing others and can give access to new abilities. Table A.3 on appendix A has more detailed information on how the attributes change on a per class basis.

Infantry units can be specialized into:

- Assassins: Good to sneak past enemies and kill high priority targets;
- Snipers: Good to pick off targets from afar;
- Sappers: Good to defend specific positions by placing traps and scanning devices.

Land vehicles can be specialized into:

- · Hovercrafts: Good to carry troops across the battlefield;
- Artillery: Good at taking out clusters of enemies with area of effect attacks;
- Support tanks: Good at withstanding damage and providing cover to the allies.

Air vehicles can be specialized into:

- Carriers: Better at carrying units across the map than the Hovercrafts, but more frail;
- Disruptors: Good at impeding the movement of units by placing obstacles;
- Bombers: The only unit that can destroy terrain, good for shaping the battlefield to suit the player's needs.

## 3.5 Unit movement

Each space (unless otherwise noted) costs one movement to move *into*. This means that if a unit has only one movement left it can't move from a space that costs 1 into a space that costs 2, but can move from a space that costs 2 to a space that costs 1.

Flying units ignore cost increases in every special space (therefore only using 1 action to walk into *any* square).

Regarding the special spaces, the rules are summarized in Table A.2 of Appendix A.

When a unit is carrying another one, the movement rules that apply are the ones that affect the carrier unit, the carried one doesn't have relevance.

Friendly units may move over each other in order to reach their final destination, while enemy units must move around them. Whichever the case, no unit can stack with another unit. The exception to this are flying units, that ignore unit collision and can end in the same space as a grounded unit (but not on the same space of another flying unit).

## 3.6 Combat

When two units engage in combat, the one who declared it always strikes first, and if its range allows it, the attacked one has a chance to retaliate. This means that if a unit with Ran = 1 attacks an enemy with Ran = 1, they both hit each other, but if a unit with Ran = 1 attacks an enemy with Ran = 3, the enemy won't be able to retaliate, since its range doesn't allow for it.

The combat works in a sort of *rock, paper, scissor* style, where different units have an advantage over some and a disadvantage over others. In this case, it will work like so: infantry->airborne vehicles->grounded vehicles->infantry, where the arrow direction indicates that one unit has an advantage over the other (in this case, infantry is good vs airborne vehicles, etc).

When an unit hits another, the calculations are done as follows:

- The attacker's Acc is compared with the defender's Eva. Each point of Acc over Eva translates into one percentual point of critical chance (double damage). Every point of Eva over Acc translates into one percentual point of miss chance. These values will vary greatly amongst units, allowing for some units, if properly equipped, to almost have guaranteed critical chances against other units (or guaranteed misses vs others).
- The attacker's Atk is compared with the defender's Def. The difference between the two is the damage inflicted by the attacker. In case of a critical hit, this is the value used to calculate the damage.
- 3. If the attacking unit has a type advantage over the defender, an extra 25% damage is dealt. This stacks additively with critical hits, so a critical hit dealt by an unit with type advantage will effectively deal 250% damage (200% from the critical and 25% of 200%, which is 50%).
- 4. Damage is dealt to the defender. If at the end of it the defender is still alive, and if it can retaliate, the process is done but with the roles switched.

If a unit reaches 0HP it is considered critically wounded and unavailable to proceed with the mission. It leaves the battlefield, and needs 24h of real time to fully replenish it to full. One unit that reaches this state can only enter a new mission when it reaches full HP.

Regarding the special spaces, the rules are summarized in Table A.2 of appendix A.

## 3.7 Items

The game will feature items that come in two types: consumables and equipment. Consumables are items that units can carry into battle and once used have their effect and disappear from the player's inventory. They have both an effect and a range (if applicable) and are summarized in Table A.5 of Appendix A.

Each unit can carry a set amount of consumable items by default, depending on their type: infantry units can carry 1 consumable item, grounded vehicles can carry 2 and flying vehicles can carry 3. Stacks of items count as only one item, meaning that two medKits use one inventory slot, not two.

Equipments are items that are attached to the units before combat and can be either armor or weapons. Some equipments have level requirements, meaning that an unit can only use that particular piece if it has a similar or greater level than the requirement (character progression will be covered in 3.10). Armor affects the units' HP, Def, Eva and Spd (both positive and negative impacts are possible) and can also have some special effects (like increasing carry capacity, for example) and weapons set the units' range. Both armor and weapons are specific to each type of unit.

Weapons can be both melee or ranged, and can also provide modifiers to an unit's Atk and Acc and can also modify an unit's attack (like having splash damage, affecting multiple spaces at once, for example). Some specializations can only use certain kinds of weapons: Assassins can only use melee weapons, while Snipers and Artillery can only use ranged weapons. Every other class can use both kinds of weapons.

Both types of equipment can also give some special actions, like invisibility during a set amount of turns in case of armor, or long range area of effect attacks in case of weapons.

As mentioned previously, items can be acquired through dropped crates, and any unit can open the crate. If the item inside it is a consumable item and the unit doesn't have enough carry capacity, it must chose between keeping the item in the crate and dropping something from the inventory or keeping the item in the inventory. Either way, the crate will disappear after being opened. If the crate has money or equipment inside it, whatever is there goes to the player's inventory, and is usable after the mission is over.

## 3.8 Available actions

Any unit has a set of default actions they can execute: move, use item, trade, open chest and attack.

"Move" changes the unit position according to the rules already described in section 3.5.

"Use item" uses any consumable item that the unit currently possesses as long as the item requirements are met.

"Trade" trades at most one item from the unit being moved with at most one item from an adjacent unit.

"Open crates/Destroy crates" opens/destroys a crate (depending on the faction, section 3.3) adjacent to the unit.

"Attack" is available if there is at least one target within Ran of the current unit and engages both units in combat.

"Wait" is always available and makes the unit stand in the current space, not executing any other action.

Eventually the units will be able to learn special actions that consume AP to execute (that will be described later).

Vehicles also have the carry and drop actions available.

"Carry" is only usable when there is an unit on foot adjacent to the vehicle and if the vehicle isn't

carrying any unit at the moment, and effectively removes the unit on foot from the map until it is dropped.

"Drop" is only available when the vehicle is carrying someone with them and if there are any adjacent spaces where the unit can be legally dropped. If the carrier unit is destroyed while carrying another unit, that unit is dropped on the same space. If that space is illegal to that unit, it will be dropped in the closest legal one.

#### 3.8.1 Special actions

Some special actions are granted by leveling up (character progression will be covered in 3.10) and can be learned by all types of units, while others are specific to some. The actions that can be learned by everyone are:

- Overwatch: makes the unit attack the first enemy that comes within range during the next turn (the action is invisible to the enemy);
- Defend: makes the unit gain a massive boost to its defense during the next enemy turn. This
  action is visible to enemies.

There is one ability that can only be learned by infantry units, which is Camouflage, that makes the unit invisible until its next turn, while regaining some HP.

Conversely, there's also one ability that can only be learned by vehicles, which is Overdrive, that gives that unit an extra turn after the current one but loses the turn afterwards.

As mentioned in 3.7 some equipments also grant special abilities.

## 3.9 A turn

One turn is where a player gives orders to his units. There are two kinds of orders: final and non-final orders.

Final orders are the ones that, when issued, end that unit's turn and are: attack, use item, open/destroy chest, use special action and wait.

Non-final orders, on the other hand, are orders that allow units to carry on with their turns and execute other final and non-final orders. Non-final orders are: move, trade, carry and drop.

When an unit is dropped, its state is the same as when it was picked up, meaning that if that unit hadn't acted prior to being carried it can still act in the same turn.

Orders are executed as soon as they are ordered, meaning that the units move and attack when the players give the command, as opposed to the player giving all the orders and them being executed all at once. This may seem a minor concern but it is actually important. If the orders were executed just at the end of the turn, the player could order two of his units to attack one enemy, but the first one could have a lucky crit and immediately kill that enemy, effectively rendering the second unit's turn useless (or the same could happen but with an unlucky miss, making one enemy live longer than it should). The player can also give one order to a unit, then give orders to a different unit and finish the first unit's action. This mean the player can move unit A one square, then move and attack with unit B, then move unit A again, etc.

When the player assigns one final order per unit the turn passes to the enemy.

## 3.10 Progression

The game progression will be comprised of two distinct stages: the story/stages progression and the party/level up progression.

The stages progression will consist of a series of sequential levels, with 3 degrees of completion. One primary objective, which is the only one required to complete the level itself, and two secondary objectives. Completing the secondary objectives will give the player extra cash in order to develop their party further. The levels can be completed as many times as desired, but only the first completion of each objective will give its reward. In addition to it, if the player can complete all 3 objectives at the same time, it will get an extra reward.

The types of objectives (both primary and secondary)that can appear are the following:

- Annihilation: kill every enemy on the map;
- Conquest: have control of some key points in the map. Having control means having a unit on a spot for two entire turns without leaving that spot;
- Defend the VIP: don't let enemies kill a certain key character during x turns;
- Assassination: kill one particular enemy.

There are also some objectives that are only used as secondary:

- Rush: complete the primary objective within x turns;
- Treasure hunter: collect all treasures on the map;
- Give them a break: complete the mission without using troops of x type.

It is important to note that any damage taken by the units isn't healed upon completing a mission. The units recover HP over real time, being that at most one unit will take 24h to recover to full (the 24h cap is reached when a unit gets critically injured (reaches 0HP) during a mission). If a unit is simply wounded, but not critically injured, the player can also use consumable items (mediKits and repairKits) outside of the missions to heal the units, but it won't get any experience (exp) (covered later) in doing so. Any unit that was simply injured during a previous mission can start a new one with any amount of HP, but a critically injured unit must wait to be fully healed before doing so.

The party progression consists of the player leveling up their units by using them in combat as well as buying items at the shop to upgrade them.

The level up is based on an exp system. Whenever a player does anything with a particular unit, both the unit and the equipment that unit is wearing gain exp (with certain actions giving more than

others), and upon reaching a certain threshold, the unit levels up. The actions that give exp (by ascending order of points given) are:

- 1. Playing a mission with a unit;
- 2. Taking damage (armor gets extra exp from this one);
- 3. Dealing damage (weapons get extra exp from this one);
- 4. Assist another unit (healing, giving items, carrying);
- 5. Kill an enemy;
- 6. Win a mission;
- 7. Complete an objective (meaning that the unit that actively completes an objective gets extra exp.

When an unit levels up, it gains a few Upgrade Points (UP) that the player can then use. Each UP can be used to increase one attribute of that unit by one. Upon reaching certain thresholds, leveling up one attribute gets more expensive. For example, when an attribute is between level 1 and 10, it costs 1 UP to increase it. From level 11 to 20, it costs 2 UP, and so forth. These values are only to exemplify the concept and are subject to change upon testing.

UP can also be used to upgrade some special fields, namely special actions effectiveness and item effectiveness. Upgrading that makes the items replenish more HP and makes the special actions last longer/deal more damage/have their effect increased (it depends on each particular action).

When a weapon levels up, it gets an increase in effectiveness, either by giving extra bonuses or by having smaller penalties, but unlike with units development, the player doesn't chose how the weapon levels up.

Plus, the player can also go to the shop to buy both consumable items and equipment.

## 3.11 Unique selling points

Most mobile games are games played for about 5-15 minutes, while most strategy games, usually designed for consoles or computers, are games that require much greater play sessions (that can go up to several hours) in order to play effectively and fully engage on the game.

Plus, most strategy games for the mobile market are mostly ports of other games that come from consoles or computers, which sometimes lead to poor design decisions and poor optimization for the target market, like bad interfaces or too steep/long of a learning curve. The reality is that if a player wants to spend hundreds of hours with a complex game, mobile wouldn't be the platform or choice.

That said, none of the mechanics in this game are completely unique, rather just adaptations from other games, but on the other hand, it is developed with a mobile market in mind from scratch, and so can provide an in depth game that strategy fans can enjoy while doing so in small play sessions.

This allows for every design decision to have in mind some additional constraints, like user retention and monetization, both of which can be very difficult to integrate after the game has been designed, and can often lead to a broken and unbalanced game.

However, there are a couple of mechanics that aren't commonly used in TB tactical games. One of them is the fact that each unit can use a multitude of different weapons, that can have different ranges, unlike what's usual, which is, each unit can carry weapons of one type and always has the same range. The other one is the delayed attack. Currently there is only one example (the Artillery specialization special skill) but later, if seen fit, it can be added to other actions.

# 3.12 Replayability/Player retention

This chapter will cover replayability in the sense of playing the game more than once, and not *beating* the game more than once. What this means is, what encourages the player to play the game for more than one or two play sessions, and not what will keep the player from playing the game after he beat it.

The first and most important one is having a constant sense of growth and that the difficulty ramps up as the player gets more experience. However, there will also be some other more "objective" measures in order to do so:

- Multiple levels of completion: since all missions have several levels of completion, the players
  have an incentive for retrying missions they already beat previously, because they can get extra
  rewards by doing so.
- Daily quests: every day, if the player logs in, he will be given a new daily quest, that can only be completed once and gives the player extra money to spend on the store. If a player does not complete one daily quest it crosses over to the next day. This gives an incentive for the player to log in at least once a day in order to maximize his in-game income.
- Unit HP regeneration: This is probably on of the more important techniques used to keep players engaged, and it is widely used in the current mobile games. The fact that the units don't replenish their HP instantly and need some time to recover (or need the player to buy some items, which he may not be willing to) makes it so the player sometimes is better waiting for a few hours/a day and only then come back to the game. This may seem contradictory, but it is better to have some forced exit points so the player leaves the game and comes back a few hours late, because it prevents frustration at the game [23, 24]. However, this waiting time only happens if the player fails to protect his units, instead of just being something that depletes whenever the player plays at all, which would be unfair.
- Push notifications: this can integrate with almost all the other methods already presented, and it serves as an attention grabber, that can let the players know when something important has happened while they weren't at the game. For example, whenever a unit reaches full HP or whenever a new daily quest is granted a notification can be sent.

## 3.13 Discussion

In this section I will present some mechanics that I purposely left out of the game, that could make sense put into it, and why.

The first, and probably most obvious one, is the Fire Emblem's progression system. I left it out because while it is a good system for a game where players spend long hours with their characters, that have a deep story and personality behind, in this game, where play sessions are expected to be much shorter (in the order of minutes) it could cause two problems: either the specialization and leveling of the characters would feel too rushed (if a player uses a character two times with a specific purpose and it immediately gets some sort of specialization in that direction, which could actually lead to the player not understanding how and why it happened, maybe because he didn't purposefully try to make that unit have a specific role) or would feel too slow and could make the player feel that its actions were meaningless (playing 10 mission, while in a Fire Emblem context is almost nothing more than an introduction, for this type of game it is interpreted as much longer and some sort of feedback is expected, almost on a mission by mission basis).

Another mechanic that could have been easily implemented and it is not very explored in tactics games is the notion of simultaneously executed turns (section 2.2). This wasn't implemented because it requires much more advanced thinking, because the player doesn't only need to think what he wants to do, but must think "what will the opponent do, and whit that assumption what do i want to do?". It may not seem like much, but the space of possibilities explodes, and can lead to a much more heavy and overly complex game, that may be completely fine in other contexts (like computer games) it isn't as suited for the mobile market. Plus, the game being vs an AI could make the player feel cheated, because he would have no guarantee that the AI didn't know what he ordered his units to do.

However, there are hints of this mechanic in the game, exemplified by the Artillery specialization special action. However, its use is completely optional, so that players that are looking for some extra depth can delve into it without sacrificing the overall simplicity of the game.

## 3.14 Technology used

As far as how the game was implemented, I chose Unity engine, and that is for a few reasons. First and most importantly, it has easy deployment for mobile devices, which is really important since the project was be focused on that target. It is also the engine with which I have some experience, having developed some projects, both in 2D as well as 3D. Another good point is the Unity store, that has a lot of content that can aid and speed up some processes. Lastly, Unity also has a larger userbase than any other game development platform, which can be proved useful if facing some hard resolution problems.

UDK was also considered as an alternative, but the lack of experience, coupled with the lack of 2D options and smaller userbase made me chose Unity as the development tool, even if UDK can be more powerful in some aspects, like graphical fidelity (that isn't at all the objective of this work) and having a profiler available, which can help debug and solve performance problems (Unity only has it

in the paid version, but nothing in the game makes it seem that there will be any issues).

# 4 Developed work

As described in Section 3, the idea was to develop a fully fledged game, from the combat system to the in-game shop. However, when implementing the main game loop (the combat itself) one thing became apparent: the UI/UX would be a much larger problem than initially anticipated.

Due to this fact, the project scope had to be reduced, so the project became a study on what the best way to implement the UI/UX on a mobile tactical turn based game is.

As far as mechanics go, the original idea was for the game to have several different intertwining mechanics, such as multiple classes with specific abilities, different terrain types that units can overcome in different manners, etc. However, early testing suggested that the public was not too receptive to such a system, and instead preferred something a little more simple, as an introduction to the genre. Therefore, the game evolved into an experience that would make it intuitive and user friendly for anyone who never played a tactical TB game.

This, coupled with the heavy focus on UI/UX made this project into a game focused only on introducing new players to simple concepts commonly used in TB games, such as positioning, area of influence of units and danger zones.

## 4.1 New solution

Having decided to reduce the scope of the project, a newer and simpler approach was required. The first thing to determine was "how much is too much" in terms of complexity. So, a very simple prototype (focused only on the mechanics, not the UI/UX) was developed, consisting of:

- 1 type of unit with only 4 attributes: HP, Ran, Atk and Spd;
- simple combat: whenever one unit is in range of another one and attacks, the attacked unit loses HP equal to the attacking unit Atk;
- 1 type of terrain, where each unit can move one square per Spd point.

Where each player had a sheet of paper with the unit's attributes and what they meant and the interaction consisted only of clicking on a unit to select it and a square/enemy unit to move/attack.

This proved to be much better in terms of the new player experience, as everyone was able to easily comprehend how all the mechanics worked together and seemed to enjoy the game more than before.

However, there was still some major complains regarding the UI/UX (mainly referring some difficulty understanding the "area of influence" of both allied and enemy units) as well as the lack of diversity in terms of units, saying the game was a bit stale.

The next iteration had more unit variety (4 different units, each one with different attributes), as well as different terrain types (as described previously: normal, difficult, impassable). This resembles the first developed prototype, the main difference being that all units had the same rules for movement and the combat was simplified.

This, once again, proved to be an improvement, but the different terrain types still seemed to cause some confusion next to the test subjects.

With this in mind, a final gameplay prototype was developed with the following characteristics:

- 4 units with the following attributes: HP, Ran, Atk and Spd;
- One terrain type: where one point in Spd equates to one square per turn;
- Simple combat: whenever one unit is in range of another one and attacks, the attacked unit loses HP equal to the attacking unit Atk;
- One game mode: the first player to destroy all the opponent's troops wins.

The 4 created units were infantry, tanks, airborne units and power suit units. Even though all the units have the same attributes I tried to make them feel different both in playability by varying the number of each troop as well as having attributes that can be easily relatable to the type of unit in question.

With that in mind, the units had the following characteristics:

- Infantry: low all around attributes, but high number of units;
- Tanks: incredibly high Ran and Atk, but very low Spd;
- Airborne units: incredibly high Spd;
- Powersuit units: above average attributes, but lower number of units.

The number of units are relevant to the user tests (described in more detail in section 5.1).

With the gameplay mainly finished, the major focus was the UI/UX. This was a big issue due to the lack of case studies that fit the genre/platform combination (tactical TB games for mobile). This, just like the gameplay, also required a lot of iteration but unlike the gameplay, didn't intend to develop the best UI/UX. Instead, the objective was to study different approaches and analyze the pros and cons of each one, concluding what would be a recommended approach.

There was also another consideration to keep in mind: how many different approaches to try and test. If too little, no relevant conclusion could be taken. If too many, the users could get bored testing them all at once (and testing them at once is necessary to reduce data corruption) and so the test results would not be conclusive.

With that in mind, I decided to test 3 different interfaces:

- Alternative 1 (Control mode): One resembling the interface present in computer games to serve as a baseline. Something that the users should be used to, only interacting with it in a different manner;
- Alternative 2 (Basic range mode): One trying to provide the information as simply as possible. This should lead to more ease of access to the game, since more information is present, but more frustration to experienced players, as they would need to click more than necessary. This should also be true for alternative 1;

 Alternative 3 (Auto attack mode): One trying to use as few taps as possible. This would be a somewhat more complex interface, because in order to reduce the number of click, less information would be presented.

However, all of the interfaces had some similarities. The similarities were intended and served the purpose to verify if one feature was rated higher or lower depending on the context it appeared in. The similarities were:

- Whenever one unit could still move on the map, gray squares appeared on spaces that the unit could move to, but would then be vulnerable to an attack from an enemy unit (Figure 4.1);
- There was a menu where players could toggle on or off the display of both movement and/or attack squares. This could be used to more easily plan a movement (toggling off the attack indicators, which could be distracting) or vice versa.



Figure 4.1: An image where the vulnerability indicator is visible. If the airplane moves to the gray squares it will be in attack range from at least one enemy unit.

### 4.1.1 Control mode

This interface had the following characteristics: upon clicking on a friendly unit, 3 sets of icons appeared. **Green squares** represented the spaces where that unit could move to, **gray squares** represented places where the unit could move but would be in attack range of an enemy and **red targets** represented the max range at which the unit could attack (max range in this case means "range that the unit can hit if it moves to it's fullest capability).

Red targets would also appear if the selected unit could attack an enemy unit from its current position (figure 4.2).



Figure 4.2: Input method 1.

To attack, the players had to move into attack range, then select the same unit again and target the enemy unit they wanted to attack.

This tried to emulate as much as possible the interface commonly used in games such as this one (in the figure 4.3 we can see an example, taken from a Fire Emblem game <sup>1</sup>. In this case, blue squares represent where the selected unit can move to while red squares represent it's range beyond that movement).



Figure 4.3: Fire Emblem's movement and range indicator.

### 4.1.2 Basic range mode

This interface had the following characteristics: upon clicking on a friendly unit, 3 sets of icons appeared. **Green squares** represented the spaces where that unit could move to, **gray squares** depicted places where the unit could move but would be in attack range of an enemy and **red targets** appeared on the spots that the unit could attack **from the current position**.

Red targets would also appear if the selected unit could attack an enemy unit from its current

<sup>&</sup>lt;sup>1</sup>Taken from Fire Emblem Awakening

position (figure 4.4).

To attack, the players had to move into range, then select the same unit again and target the enemy unit they wanted to attack.

While it requires as many clicks as the first approach, it is here to see if it's easier to understand the units' range with this presentation or the "max range" one.



Figure 4.4: Input method 2.

### 4.1.3 Auto attack mode

This interface had the following characteristics: upon clicking on a friendly unit, 3 sets of icons appeared. **Green squares** represented the spaces where that unit could move to, **gray squares** depicted places where the unit could move but would be in attack range of an enemy and **red targets** appeared over enemy units that the selected unit could attack either in its current position or in any of its possible moves.

This means that if the selected unit currently isn't in attack range of an enemy unit, but can move somewhere where it gets in attack range, a red target appears on that enemy. Then, by clicking on that target, the selected unit automatically moves into attack range and attacks (in the figure 4.5 we can see that the airplane unit is currently not in range of the infantry unit, but a target still appeared on that unit).

Alternatively, the players could also move the unit to a spot where the unit is in attack range of an enemy and manually target an enemy, just like in the previous case.

This tried to make it so experienced players (that already know the units and their attributes) could easily issue the desired commands. So, instead of:

- 1. Selecting a unit;
- 2. Moving it into the desired range;

- 3. Selecting the same unit;
- 4. Attacking an enemy.

Players could just:

- 1. Select an unit;
- 2. Attack an enemy.

This, however, could pose a couple of problems: since this mode it doesn't present the attack range as clearly, it could be difficult to understand all of the units. Also, it could give a sense of unpredictability, as the players don't know the path the unit will take in order to perform the attack.

This is the reverse of the previous one. Instead of making it easier for newer players, it tried to make the game faster for experienced players.



Figure 4.5: Input method 3.

# **5 User tests**

With both the gameplay and all the required interfaces complete, user tests had to be performed. The objective was for the tests to be conducted with a minimum of 20 subjects in order to ensure some minimum consistency, but there was no upper bound in the number of subjects.

Each test was conducted with a different interface order, to prevent some result skewing due to one interface being played before/after another one.

## 5.1 Setup

The tests had a simple procedure: since the game didn't have any AI, it was played by 2 test subjects at a time in a versus match, played in 3 rounds. Each round was played with a different interface option, being that the order of the interfaces changed from one test to another. However, the ordering wasn't random nor tried to cover all the different possibilities. This is because covering all the possibilities would make it so very few people played one given option, which in turn would lead to a sample size not big enough to be conclusive.

So, three orders were chosen:

- Order A: Alternative 1, 2, 3
- Order B: Alternative 3, 1, 2
- Order C: Alternative 2, 3, 1

After playing a session, each subject received a questionnaire that was focused on 3 aspects (appendix B):

- The habits of each subject: this was focused on gaming and handheld devices usage and could be relevant to draw relations between those habits and the enjoyment/understanding of the game mechanics and interfaces;
- The playability of the game itself: while this wasn't the main focus of this project, it could be very
  relevant, because if the game itself is too boring/confusing, it could influence the understanding
  of the interfaces themselves. It had questions about perceived game balance (if one unit is
  much stronger than another, if the amount of units was too low or high) and if the difference
  between troops was clear (as detailed in Section 4.1 I tried to make each unit feel unique, with
  both strengths and weaknesses, so I tried to check if that difference was clear);
- The interfaces: this was the main focus of the project and so was the Section that was more detailed. It had questions about each of the interface options and also asked the subjects to order them by preference. Each question will be analyzed in more detail in Section 5.2.

The test was conducted in Taguspark, during a game related event (MOntra de JOgos (MOJO), which could lead to some skewing of the results, as 96% of the population had some prior experience

with videogames) and counted with the presence of 32 participants with 10 people playing each of the interface orders.

## 5.2 Results

In this section, we will try to analyze the user responses and try to figure out multiple things: **which** play mode was the preferred one, **why** it was the preferred one and also if the other modes had some features that could be taken and used to upgrade the best mode.

#### **Preferred mode**

The first thing to analyze is what the preferred mode was, as it will help when deciding which features to look at as good examples to follow. For that, a simple graph depicting how the users organized the different modes is present in figure 5.1. The graph shows the number of users that placed each mode in 1st, 2nd and 3rd place.

In it, we can see that the difference between modes is fairly well defined, with 23 users (76% of the whole test base) placing the basic range mode in 1st place, 16 users (53%) placing the auto attack mode in 2nd place and 20 users (67%) placing the control method in 3rd place. This shows pretty clearly that most users preferred the basic range mode.



Figure 5.1: Control methods preference

#### Breakind down each mode

To understand why, we need to analyze each mode individually. In order to do so, the users where asked 4 questions for each of the modes (where the users could answer on a scale from 1 to 5, where 1 was didn't understand at all and 5 was understood perfectly):

- Comprehension: the overall understanding of the mode;
- Quickness: how fast the users could execute their turns;

- Reach: how easy it was to understand the range of each unit;
- Vulnerable: how easy to see were the positions where a unit would be vulnerable to enemy attacks.

We performed a Friedman's test to check if the answers to the questions are statistically different between the different modes, which gave a positive result with p < 0.01.

Then, to understand exactly where the differences occur, we perform a Wilcoxon's signed-rank test between every pair of modes.

Following are the results obtained by said test, with a summarizing table on figure 5.2:

- The control mode is significantly different from the basic range mode with p < 0.05 for every question;
- The control mode is significantly different from the auto attack mode with p < 0.05 for all the questions, except for the vulnerability question, with p = 0.08;
- The basic range mode is significantly different from the auto attack mode only for the reach question with p < 0.05 and is not significantly different for all the other questions (p = 0.426 for the comprehension question, p = 0.083 for the quickness and p = 0.091 for the vulnerable). In this case we consider that the quickness isn't statistically different because while the p = 0.08 (just as in the control mode auto attack mode comparison) in this case the median value changes less than in the previous case.

	Comprehension	Quickness	Reach	Vulnerable
Control/Basic range	diff, p < 0.05	diff, p < 0.05	diff, p < 0.05	diff, p < 0.05
Control/Auto attack	diff, p < 0.05	diff, p < 0.05	diff, p < 0.05	diff, p = 0.08
Basic range/Auto attack	not diff, p = 0.4	not diff, p = 0.83	diff, p < 0.05	not diff, p = 0.09

Figure 5.2: Summary of the differences between the different modes using Wilcoxon's signed-rank test

From this analysis, we can conclude that, while some differences were still present between the basic range and auto attack mode (particularly concerning the units reach) the main differences were between the control method and the other two.

Now, to understand how big were the differences, we can look at a graph showing how many users gave which score to each question (figure 5.3) and a box-plot chart that shows the average value of each question (figure 5.4).





(b)



Figure 5.3: Bar graph showing the user distribution amongst different play modes.



Figure 5.4: Box-plots depicting the evaluation users gave to the different play modes.

The fact that the control mode was the worst one is once again present, and now we can see that it was the worst one in every individual category, as well as the worst one overall, so from now on all the tests will be conducted on the other two modes.

Regarding the other two modes, even though the global evaluation was fairly distinct (23/6 users placed the control mode in 1st/2nd place, while 5/16 users placed the auto attack mode in the same positions) that difference isn't as evident when comparing the separate categories.

Firstly, as we analyzed previously, most of the answers regarding this two modes aren't statistically different, so any conclusions drawn from this information is prone to be dismissed with extra testing. However, there are some relevant factors that can still be observed.

For example, even though the answers are pretty similar, we can see that, in average, the basic range mode has slightly better scores overall than the auto attack mode. This very slight difference could be the reason why users placed the auto attack mode in 1st place a higher number of times. One of the possible explanations could be that both of the options support the same concept, but the execution in one of them was perceived as superior to the other mode.

The reach question, however, is substantially different (evident both by the quartiles as well as the test conducted previously) for the worse. This, coupled with the fact that the quickness question has virtually the same answers, can indicate that the tradeoff between clarity and speed was not worthwhile.

Still, this conclusion cannot be drawn with certainty, as the objective of the quickness mode was to give **experienced** players an advantage. The problem with that is threefold:

- The number of users that had experience with tactical games was negligible: only 16% (5 out of 30) of the users had prior experience with tactical games;
- Short playsessions: comprised of 20 minutes and one playthrough of each mode, didn't give enough time to the users to get used to the game (which in theory should be the point at which the auto attack advantages became more apparent, as that was how it was designed);
- The order of the different modes: due to how the tests were designed, some users began with this mode, which could lead to very different responses depending on the order the modes were played. The different orders were used to mitigate this difference, but analyzing each order could provide some helpful insight.

To try and get a better answer to the problem, we can try to reduce the impact of the third point by, instead of looking at all the users' answers at once, we divide them by the order they played each mode.

#### Influence of the order of play

It is important to keep in mind that every conclusion in this subsection is very fragile, even in terms of statistical relevance, as each order only had 10 players play it. However, the analysis is still performed as it can give hints of possible correlations, which could be proved with further testing.

The first thing to check is if the differences in the responses given are now statistically different (both in regards to the quickness as well as the reach questions) and by running the same, Wilcoxon's signed-rank test, we observe that:

- Reach question:
  - When users played the auto attack mode last, p = 0.565, meaning the results are very similar.
  - When users played the auto attack mode first, p = 0.06, which is very close to statistic relevance.
- Quickness question:
  - When users played the auto attack mode last, p = 0.480, meaning the results are still similar.
  - When users played the auto attack mode first, p = 0.066, which is very close to statistic relevance.

This gives a good indicator that the order by which users tried different modes actually made an impact on how they viewed each of them. Once again, by looking at a boxplot of the responses, we can have a clearer view of how each mode was rated, when regarding the reach question (figure 5.5).



Figure 5.5: Box-plots depicting the evaluation users gave to the reach and quickness questions

Just to recall, it is important to remember that when users played the auto attack mode first, also played the basic range mode in last and vice versa.

With this in mind, and regarding the reach question, the results are the expected ones:

When playing the auto attack mode first, their understanding of the units' reach was fairly limited when compared to the basic range mode. Not only that, but the basic range quartiles are incredibly small, which indicates that all users (both experienced and novice ones) had a very similar experience.

However, when playing the auto attack mode last, even if the average values are pretty similar, there is an important difference to note, which is the fact that the quartiles of both modes are fairly larger. Regarding the basic range mode, one of the possible conclusions is that the inexperienced users had more trouble understanding the range of their units (since it was the first time they were playing the game) while more experienced players quickly understood them. Regarding the auto attack mode, once again, we can draw a conclusion based on the difference between experienced/novice players. In this case the upper quartile was the one that rose, meaning that the experienced players, when using this mode last, already had a pretty good idea of how the units worked, hence the rising in better scores

When we analyze the quickness question, contrary to the reach one, the results are far from the expected. If we remember, the objective of the auto attack mode was to give experienced players an advantage (not in terms of power, but in terms of input speed) when playing their turns. When looking at the results, however, it seems it had the opposite effect, with users playing the auto attack mode first rating the quickness higher than the basic range mode. But if we look at the question differently, a possible conclusion arises. What could've happened is that, by virtue of how the auto attack mode is designed, users playing it first didn't need to have an understanding of the units' range to play their turns, so they could play them faster. However, even if they were able to play the turns quickly, they didn't understand how the units worked, which is why when reaching the basic mode, had to take extra time when playing the turns.

Conversely, when playing the auto attack mode last, since users already understood how the units worked, they didn't feel a major difference in the speed of turns.

#### Influence of experience with tactical games

One test we can perform to see if the hypothesis posed previously is true (that having prior experience with tactical games influences the answers in a meaningful way) is check if there is a difference on the responses of users with said experience. This alternative poses a problem, the fact that only 5 out of 30 users (about 17%) had previous experience with tactical games. However, this being the only method that will allow to get an early test on the hypothesis, coupled with the fact that there is at least one experienced user in every order of play, made it the one chosen for the effect.

What we will do is check if there is a statistically significant difference between the answers to the 4 questions (comprehension, quickness, reach, vulnerability) of both the experienced and the non experienced group. What we conclude, after performing a Wilcoxon's signed-rank test (with a summarizing table in figure 5.6) is:

- For experienced players none of the answers presented a statistically relevant difference, with p = 0.102 for the comprehension, p = 0.312 for the quickness, p = 0.197 for the reach and p = 0.461 for the vulnerable questions.
- For non experienced players, there is a statistical difference between the quickness (p < 0.05) and reach (p < 0.05) questions, while the other two were statistically similar (p = 0.893 for the comprehension and p = 0.114 for the vulnerable).

	Comprehension	Quickness	Reach	Vulnerable
Experienced players	not diff, p = 0.102	not diff, p = 0.317	not diff, p = 0.197	not diff, p = 0.461
Non experienced players	not diff, p = 0.893	diff, p < 0.05	diff, p < 0.05	not diff, p = 0.114

Figure 5.6: Summary of the statistically difference between experienced and novice players

Now, to actually observe the difference, we can once look at box-plots that will give us some insight on what the users answered, available on figure 5.7.





The obtained values, unfortunately don't support the previous theories (either due to the fact of a very small sample leading to incorrect results or due to the theory being wrong in the first place) because as we can see the values follow very similar patterns independently of the users having experience or not. However we can observe some things:

• For experienced players, none of the measured parameters showed a statistically relevant difference: this can mean that experienced players have some adaptability, and the skills they got from previous experiences allowed them to understand all the modes with more ease.

- Values for the basic mode were higher on average: since the basic range mode follows very closely how information is presented in most games of the genre, users might have been familiar with it, rating it higher
- Smaller auto attack values on average: even though the Wilcoxon's test showed that there isn't a relevant difference in the values, they all are smaller by some extent. This could mean that experienced players, being used to a different UI/UX in the first place had a harder time understanding the auto attack one, as it is quite a bit different from the usual.

#### Conclusion

The first thing we did was try to organize all the options from best to worst, and while there was a clear worst (the control mode) the distinction between the other ones isn't as clear.

At first, looking at only the users "podium", the basic range mode seems to clearly be the best one. However, after further inspection, we see the difference isn't as large as it may appear at first, at least regarding the considered metrics. There could be other, non tested metrics (such as overall enjoyment, for example, which can have more weight in the users responses than the ones used) that would make the difference between the modes more apparent.

While the basic attack one is marginally better in most of the evaluated categories, the difference isn't always significant. However, since one of the questions *forced* the users to organize all the modes from better to worst, most of them chose the basic range as the best one, hence the major apparent difference.

Regarding the different categories, which evaluated how the users understood each mode, how quick they could perform their turns, how well they understood the units' range and how well they could check for vulnerable positions, a first analysis showed that the major differences where between the control mode and the other two, while the basic range and auto attack modes were fairly similar, with 3 of the parameters not even being statistically different.

What this does is further prove that while the difference between the control mode and the other two is very large, the difference between the basic range and the auto attack is much smaller, which allowed us to remove one of the control modes from the analysis, simplifying both the analysis itself as well as its representation .

However, further analysis can still be conducted in order to try and refine the differences between the modes. The first and most obvious thing to try is separate the users, grouping them by the order in which they played the different modes, since that could provide valuable insight.

When doing said division, it became apparent that the order in which users played the games had a major influence on the results. For this case, only the quickness and reach answers were analyzed, as they were the only ones that showed any statistical or close to statistical difference, and only the cases where the users played the basic range/auto attack in first/last and vice versa were looked at.

There were major differences observed in both questions' case, being that one corresponded to the expected case and the other one was the opposite of what was expected.

The expected outcome was regarding the reach. When looking at how the order influenced that answer, it became clear that when users played the auto attack mode first, their understanding of the units' range was greatly reduced, which makes sense since the way it is designed makes it harder to understand the range itself, trying to compensate in input speed when users were familiar with the game and units.

On the contrary, when looking at the quickness, the answers didn't correspond to that hypothesis. In the responses, we saw that users that played the auto attack mode **first** (so, not having any prior experience with the game) thought that the auto attack mode had more impact in the speed of their turns than the users that used that mode in **last**. This goes against the expected result because the objective of the mode was to give experienced users that advantage, not novice ones.

However, one potential explanation for that is the fact that when users played the auto attack mode first, since they didn't really understand how units worked, when playing the basic range mode, had to take more time to get familiar with the units, which lead to the difference in perception.

Finally, further testing was conducted to check if users with prior experience with tactical games gave significantly different responses than the ones without experience. What we observed in that case was that, while experienced users had more adaptability (giving roughly the same responses when comparing the basic range and auto attack modes) their answers weren't very different from the users without experience. However, this final part should be taken with care, as the distribution between experienced and non experienced users was not close to even, with the experienced population representing only 17% of the total test base.

## 5.3 Discussion

From the results obtained we can see that while both the basic range and auto attack modes have good features, none of them are perfect by themselves and could use some improvement. Fortunately, most pros of the alternatives aren't mutually exclusive, and as such can be combined into one better alternative.

The pros and cons of each alternative, as indicated by the user tests, were as follows:

- Alternative 2 (basic range) pros:
  - It was the easier one for non experienced players to understand units' range, as it displayed exactly where each unit could attack.
- Alternative 2 cons:
  - It was the one more prone to human error, as the players had to manually place the units in attack range. This also happens in alternative 1, but in that one, as soon as one unit was selected, it was clear which enemies would be in range after the movement (see figure 5.8 for a comparison of the two).



Figure 5.8: Comparison between mode 1(a) and mode 2(b). In (a), the user knows that the tank can hit the soldier after moving, while in (b) the users has to count the number of spaces to move.

- Alternative 3 (auto attack) pros:
  - If the players already understood how the units worked, they were able to perform the desired actions much more efficiently and faster;
  - It was the only alternative that wasn't prone to human error when trying to attack an enemy unit (in the other alternatives, as the players had to manually move their units into attack range and then attack, it would sometimes lead to them moving the unit into a wrong space, not being able to attack afterwards).
- Alternative 3 cons:
  - If the players didn't already understand how the units worked, with this mode it wasn't clear what the units' range were. This is because the mode only displayed the enemy units suitable for attack (a way to solve this will be discussed in section 6);
  - When a player selected an enemy to attack that required his unit to move in order to get in range, if there was more than one possible square that the unit could move to, it moved into one of the safest spots automatically instead of asking the player where he wanted to move it to (in figure 5.9 we can see the multiple places the plane could move to to attack. In this case it would move to the blue cross instead of asking the user).



Figure 5.9: Auto attack mode attack example.

One of the most common requests, present both in the questionnaire as well as asked orally during the tests, was to merge the basic range with the auto attack modes. This way, the range presented would be in relation to the unit, just like in the basic range mode, but units would also be able to automatically move and attack if such an option was available, just like in the auto attack mode.

This suggestion is perfectly in line with what the test results show, since it combines the two preferred methods, giving the users easy to read information from one side as well as providing the ability to speed up the turns with automatic attacks is the users so desire.

Another common user suggestion was to be able to click on enemy units in order to see their ranges.

Both this suggestions will be covered in more detail in Section 6.1.

# 6 Conclusion

This project tried to do a study on what the best UI/UX is for mobile devices, being that most current games on the genre are either pc ports or don't have the best user interaction. Having a better understanding of the subject is important because it would lead to better mobile games, a market commonly criticized for being composed mostly of time waster games without any real depth. However, this depth is often coupled with more user options and operations, leading to an increasing amount of information required on screen at a given time.

In order to do that, the approach taken was the development of three different options, each with a different objective, for users to try, later performing user tests, where each user would play the different options. The options were:

- Control mode (mode 1): trying to emulate how tactical games present information, with the attack range laid out after the movement range;
- Basic range mode (mode 2): displays the attack range in regards to the unit, not the maximum movement range. This tried to check if the users thought the information was clearer when presented like this or the usual way;
- Auto attack mode (mode 3): makes the units able to attack automatically if they can move in attack range of an enemy and hit it in the same turn. This tried to give experienced users (ones who understood the game properly) a way to speed up their turns, requiring only 1 click to attack instead of 2 (very relevant in mobile, preventing misclicks).

Since every user played the 3 game modes instead of playing only one, and to prevent the results being skewed by the order in which users played the modes, three orders were selected and each users played one of those. The user tests had several different objectives:

- 1. Figure out if one alternative was clearly better than the others: this would give a good first idea of what features to look at as good practice;
- 2. Understand what made the alternatives good or bad: this is a more thorough approach than the first one, looking at several different parameters instead of just the overall rating of each mode;
- Check if previous experience with tactical games influences how the users react to the different modes.

The first objective was somewhat achieved, with the control mode being the clear worst one. However, while the basic range mode was the preferred one by the majority of users, the difference between it and the auto attack mode, regarding the chosen metrics was small enough to warrant further analysis.

This analysis crossed over with the second objective, as the best way to distinguish between the alternatives is look at what makes them good and/or bad. All the alternatives, apart from being rated

in terms of global preference, were also divided in categories: overall **comprehension**, **quickness** of the turns, understanding of the **reach** of units and understanding of **vulnerable** positions (spaces where a unit would be in range of attack from an enemy troop).

An early analysis of the different parameters showed no meaningful difference between the basic range and auto attack modes, except in the reach category, where the basic range was clearly better than the auto attack one. This pointed to the fact that maybe the tradeoff between clarity and turn speed was not worthwhile. However, this conclusion cannot be drawn just from this fact, because the auto attack method was meant to give experienced users (experienced both in the game, with a good understanding of how the units worked and what their ranges were, as well as experienced with tactical games in general) an advantage. Because of it, and the fact that the order of play was not the same for all users, it was important to subdivide the users both by the ones that played the auto attack mode first and last as well as by users with prior experience with tactical games.

With the first division, it became clear that the auto attack mode had the opposite effect it was supposed to. This is because the users without any experience with the game, when playing the auto attack mode first, used it as a crutch to play their turns (because they didn't have to play the units in specific places) which sped up their game, but made it so they didn't understand how the units actually worked.

Regarding the division between experienced and non experienced users, the answers weren't as clear, mainly because the sample size was much smaller, consisting only of 17% of the users. However, what we can observe is that prior experience doesn't influence the responses, which, once again, goes against the objective of the auto attack mode.

These conclusions, however, don't mean that the auto attack mode is useless or harmful to the experience, rather meaning that it serves a different purpose from the initial one, and with some improvements can become even more powerful, discussed in section 6.1.

### 6.1 Future work

The work developed in this project, even though serving as a decent ground for other projects to be built upon, still is in a very early stage, and could use some changes and improvements. Some of these, even if looking only small quality of life improvements, could mean major changes in the way users perceive the game (evidently subject to further testing).

One addition that should be done to the game as a whole is to add the option for users to click on enemy units and seeing their range (the way it would be presented depends on how the developed decides to implement range in the first place, but the option should exist nonetheless). This was suggested by some users but another, more important sign it is a significant feature is the fact that while playing, many users after moving their units, were attacked by their opponent, later saying they didn't have conscience that that was a vulnerable position.

Regarding the auto attack mode, one change that could be made (even though it wasn't suggested and there wasn't evidence during the tests that it impacted the experience negatively) is allowing the users to select where they want their units to move when attacking a target, unlike how it is currently done, where a position is automatically selected. This would defeat the purpose of this mode altogether, because it would then require the same number of clicks to perform the same actions, it is only mentioned purely from a theoretical point of view, as further testing could reveal some interesting conclusions.

Another change that could be done, and was even proposed by some users, would be to merge the basic range mode and the auto attack mode. This way, the range would be presented just like in the basic range mode, allowing for manual control of the attacks, while simultaneously giving the option to automatically attack any unit in movement+attack range.

# Bibliography

- P. Piejko, "Most popular smartphone screen resolutions in 2015 https://deviceatlas.com/blog/most-popular-smartphone-screen-resolutions-2015, accessed at Monday 21<sup>st</sup> November, 2016."
- [2] "Browser display statistics http://www.w3schools.com/browsers/browsers\_display.asp, accessed at Monday 21<sup>st</sup> November, 2016."
- [3] A. Solo, "Strategy vs. tactics in video games a small essay http://www.spacesector.com/blog/2011/12/strategy-vs-tactics-in-video-games-asmall-essay/, accessed at Monday 21<sup>st</sup> November, 2016."
- [4] "Turns, rounds and time-keeping systems in games http://en.wikipedia.org/wiki/Turns,\_rounds\_and\_time-keeping\_systems\_in\_games# Turn-based, accessed at Monday 21<sup>st</sup> November, 2016."
- [5] "Xcom overwatch system

http://xcom.wikia.com/wiki/Overwatch, accessed at Monday 21st November, 2016."

- [6] taw, "Xcom: Optimal character build or how to design skill trees http://t-a-w.blogspot.pt/2013/12/XCOM-optimal-character-build-or-how-to.html, accessed at Monday 21<sup>st</sup> November, 2016."
- [7] S. Totilo, "Fire emblem: Awakening: The kotaku review http://kotaku.com/5980657/fire-emblem-awakening-the-kotaku-review, accessed at Monday 21<sup>st</sup> November, 2016."
- [8] C. Stern, "12 ways to improve turn-based rpg combat systems http://sinisterdesign.net/12-ways-to-improve-turn-based-rpg-combat-systems/, accessed at Monday 21<sup>st</sup> November, 2016."
- [9] C. Stern, "6 more ways to improve turn-based rpg combat systems http://sinisterdesign.net/6-more-ways-to-improve-turn-based-rpg-combat-systems/, accessed at Monday 21<sup>st</sup> November, 2016."
- [10] "Fire emblem fog of war system http://fireemblem.wikia.com/wiki/Fog\_of\_War, accessed at Monday 21<sup>st</sup> November, 2016."

[11] "Fire emblem villages system

http://fireemblem.wikia.com/wiki/Village, accessed at Monday 21<sup>st</sup> November, 2016."

- [12] "The complete book of military science, abridged http://web.archive.org/web/20020113200048/http://www.28thmasscob.org/reconac. PDF, accessed at Monday 21<sup>st</sup> November, 2016."
- [13] "Xcom morale system

http://www.ufopaedia.org/index.php?title=Agents\_Stats\_%28Apocalypse%29#Morale. 2FBravery, accessed at Monday 21<sup>st</sup> November, 2016."

- [14] "Xcom meld http://XCOM.wikia.com/wiki/Meld, accessed at Monday 21<sup>st</sup> November, 2016."
- [15] "Final fantasy counter ability http://finalfantasy.wikia.com/wiki/Counter\_%28Ability%29, accessed at Monday 21<sup>st</sup> November, 2016."
- [16] "Android's core app quality http://developer.android.com/distribute/essentials/quality/core.html, accessed at Monday 21<sup>st</sup> November, 2016."
- [17] "App store guidelines

https://developer.apple.com/app-store/review/guidelines/#terms-conditions, accessed at Monday 21<sup>st</sup> November, 2016."

- [18] T. Rogers, "Let's talk about touching: Making great touchscreen controls http://www.gamasutra.com/view/feature/187126/lets\_talk\_about\_touching\_making\_ .php, accessed at Monday 21<sup>st</sup> November, 2016."
- [19] D. de Geus, "The 8-step guide to interface design for iphone games http://www.paladinstudios.com/2012/04/23/the-8-step-guide-to-interface-designfor-iphone-games/, accessed at Monday 21<sup>st</sup> November, 2016."
- [20] D. de Geus, "Momonga monday: The new retina interface http://www.paladinstudios.com/2012/05/07/momonga-monday-the-new-retinainterface/, accessed at Monday 21<sup>st</sup> November, 2016."
- [21] "Home is where the mobile game is: 96% play games in the house, 53% in bed http://techcrunch.com/2012/03/22/home-is-where-the-mobile-game-is-96-playgames-in-the-house-53-in-bed/, accessed at Monday 21<sup>st</sup> November, 2016."
- [22] "Average mobile app session length as of august 2014, by category (in minutes) http://www.statista.com/statistics/202485/average-ipad-app-session-length-byapp-categories/, accessed at Monday 21<sup>st</sup> November, 2016."

- [23] M. Katkoff, "Supercell's spooky pop and the six rules of a hit puzzle game http://www.deconstructoroffun.com/2015/01/supercells-spooky-pop-and-six-rulesof.html, accessed at Monday 21<sup>st</sup> November, 2016."
- [24] J. Julkunen, "3 things to know about session-length restriction when designing a free2play game http://www.gamerefinery.com/3-things-to-know-about-session-length-restrictionwhen-designing-a-free2play-game/, accessed at Monday 21<sup>st</sup> November, 2016."
# A Gameplay summary tables

Att	Effect
HP	Hit points, when it reaches 0 the unit is taken down
AP	Points available to perform special actions
Spd	How many spaces the unit can move during one turn
Atk	How much damage the unit deals
Def	How much damage the unit absorbs when it is hit
Ran	How far the unit can attack
Acc	The greater it is, the more likely the unit is to succeed an attack
Eva	The greater it is, the more likely the unit is to dodge an attack

Table A.1: Different attributes and what they influence

Terrain	Cost	Combat	Units	
Normal	1	normal	All	
Walls	na	normal	None	
Water/holes	1	normal	Flying	
Mud/swamp	2	- Eva	All	
Forest	2	+ Eva	ΔΙΙ	
101631	2	- Acc		
Shielded spaces	2	++ Def	All	

Table A.2: Types of spaces and their interaction with movement and combat

Base class	Specialization	Bonus	Penalties	Special actions
	Assassins	Acc, Spd	HP, Def, Eva	Sneak
Infantry	Snipers	Ran, Acc	HP, Def, Eva	Steady shot
	Sappers	Def	Spd, Atk	Contact mines, Detection device
	Hovercrafts	na	Def, Eva	Hover, Extra carry
Ground vehicles	Artillery	Atk	Def, Eva	Siege
	Support tanks	HP, Def	Spd, Atk	Shield drop
	Carriers	Spd	HP, Def, Eva	na
Flying vehicles	Disruptore	Dof*	Spd*, can't carry units*	Wall drop
	Disruptors	Dei	can't attack	(*only in effect when the skill is active)
	Bombers	Def	Spd	Bomb drop

**Table A.3:** Different specializations and their characteristics

Special action	Unit	Effect	Level req
Defense	All	Get extra Def for one turn	5
Overwatch	All	Attack the first unit that comes within range	10
Camouflage	Infantry	Get invisible for one turn and get HP regeneration	15
Overdrive	Vehicles	Get an extra turn	15

Table A.4: Available special actions unlocked via level-up

Item	Effect	Range	
MedKit	Heal infantry unit	Self or 1	
RepairKit	Heal vehicle	Self or 1	
Adrenaline Shot	Give extra turn to infantry	Self or 1	
Overcharger	Give extra turn to vehicle	Self or 1	
Portable Shield	Drop a 3x3 shielded zone	On the unit's position	

Table A.5: Consumable items

lype		Att modifier	Special effect	Special action	Level req
Infantry	Melee weapon	Atk: +1	na	na	na
try	Weapon: Ran = 2, 3	na	na	na	na
<u>e</u>	Weapon: Ran = 3, 4	na	na	na	na
ntry	Armor	Def: +1	na	na	na
icle	Armor	Def: +1, Spd: -1	na	na	na
ntry	Weapon: Ran = 2	Atk: -1	Area of effect in a 1x3 area perpen-	Create a 1x3 (per- pendicular to the	5
			dicular to the attack	direction fired) area	
			direction	of flames. Units in	
				it take 1 damage	
				per turn	
cles	Weapon: Ran = 3	Atk: -3	na	Deal damage	5
				equal to the unit's	
				attack to all units	
				(not just enemies)	
				in a 3x1 line.	
htry	Armor	Def: -1	Allows the unit to	na	10
			move over enemy		
			units.		
cles	Armor	na	Allows the unit to	na	10
			carry one extra		
			consumable item		

Table A.6: List of available equipments

# **B** User questionnaire

# Análise a um jogo táctico por turnos

\* Required

#### 1. formato \*

Mark only one oval.

$\bigcirc$	formato 1
$\bigcirc$	formato 2
$\bigcirc$	formato 3

## Análise a um jogo táctico por turnos

Ao responderes ao inquérito autorizas que os dados sejam guardados e processados estatisticamente de uma forma anónima.

2. Sexo \*

Mark only one oval.

$\bigcirc$	Masculino
$\bigcirc$	Feminino

# 3. Idade? \*

Mark only one oval.



# Hábitos

 Costumas utilizar dispositivos móveis tácteis (smartphone, tablet)? \* Mark only one oval.

C	Sim	

Não

5. Se respondeste sim à pergunta anterior: Para que costumas utilizar os dispositívos tácteis? (selecciona as 3 opções mais relevantes)

Check all that apply.

Jogar
 Ver filmes/séries
 Ouvir música
 Trabalhar (edição de documentos, etc)
 Redes sociais
 Email
 Other:

 Costumas jogar jogos de vídeo (consola, computador, mobile, etc)? \* Mark only one oval.

$\bigcirc$	Sim
$\bigcirc$	Não

#### Se respondeste sim à pergunta anterior: Que género de jogos costumas jogar (selecciona as 5 opções mais relevantes) Check all that apply.

WRPG (Skyrim, Mass Effect, etc)
Puzzle (Portal, Talos Principle, etc)
Plataformas (Sonic, Super Mario, etc)
Corridas
ARPG (Diablo, Dark Souls, etc)
Hack and Slash (Devil May Cry, Darksiders, etc)
MOBA (League of Legends, DOTA, etc)
FPS (Call of Duty, Battlefield, etc)
Mobile (Candy Crush, Angry Birds, etc)
Desporto (Fifa, NBA, etc)
Fighters (Street Fighter, Mortal Kombat, etc)
JRPG (Final Fantasy, Disgaea, etc)
Jogos tácticos (X-Com, Advance Wars, etc)
RTS (Starcraft, Age of Empires, etc)
Simuladores (Sims, SimCity, etc)
Other.

# Jogabilidade

8. Avalia a dificuldade que tiveste em compreender o objectivo do jogo *Mark only one oval.* 

			1	2	3	4	5	
	Muita dificulda	ade	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$	Nenhuma dificuldade
9.	Achaste a div Mark only one	ersid	lade de	tropas	adequa	da? *		
	Sim							
	Não, a	chei d	que devi	am hav	er meno	s		
	Não, a	chei d	que devi	am hav	er mais			
	Other:							
10.	Achaste a qua Não o número Mark only one	antid de tr oval	<b>ade de t</b> opas dife	erentes	a <b>dequa</b> , mas o	<b>da? *</b> número	de tropa	as totais
	Sim							
	Não, a	chei d	que devi	am hav	er meno	s		
	Não, a	chei d	que devi	am hav	er mais			
	Other:							
11.	Conseguiste	comp	preende	r as vai	ntagens	e desv	antager	is de cada tipo de tropa

11. Conseguiste compreender as vantagens e desvantagens de cada tipo de tropa? \* Mark only one oval per row.

	1 - Não compreendi	2	3	4	5 - Compreendi perfeitamente
Infantaria	$\bigcirc$	$\bigcirc$	$\Box$	$\supset$	$\bigcirc$
Tanque	$\bigcirc$	$\bigcirc$	$\bigcirc$		$\bigcirc$
Avião	$\bigcirc$	$\bigcirc$	$\Box$		$\bigcirc$
Unidade de fato	$\bigcirc$	$\bigcirc$	$\Box$	$\supset$	$\bigcirc$

#### 12. Achaste que todos os tipos de tropas estavam equilibrados? \*

Ou seja, achas que o facto de algumas tropas serem mais fortes era devidamente compensado pelo facto de existirem em menor número Mark only one oval.



13. Organiza os vários tipos de tropas da mais forte para a mais fraca \*

Mark only one oval per row.

1º lugar (a tropa mais forte) 2º lugar 3º lugar 4º lugar (a tropa mais fraca)

Infantaria	$\bigcirc$	$\bigcirc$ $\bigcirc$	
Tanque	$\bigcirc$	$\bigcirc$ $\bigcirc$	$\bigcirc$
Avião	$\bigcirc$	$\bigcirc$ $\bigcirc$	$\bigcirc$
Unidade de fato	$\bigcirc$	$\bigcirc$ $\bigcirc$	$\bigcirc$

#### 14. Quão limitativo foi para ti o facto de o alcance das tropas ser apenas a uma distância específica

Por exemplo, o facto do alcance dos tanques ser apenas a 3 quadrados de distância e não a 1, 2 e 3 quadrados de distância *Mark only one oval.* 

	1	2	3	4	5	
Muito limitativo	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$	Nada limitativo

#### 15. Deixa aqui algumas sugestões que tenhas relativamente ao equilíbrio das tropas

Por exemplo, se achaste que as tropas estavam desiquilibradas diz o porquê e o que farias para diminuir esse problema


Skip to question 16.

## Interface

 Organiza os 3 modos de controlo que te foram apresentados por ordem de preferência \* Mark only one oval per row.

1º lugar (gostei mai	s) 2º lugar	3º lugar	(gostei menos)	)
----------------------	-------------	----------	----------------	---

Primeiro modo	$\bigcirc$	$\bigcirc$	$\bigcirc$	
Segundo modo	$\bigcirc$	$\bigcirc$	$\bigcirc$	
Terceiro modo	$\bigcirc$	$\bigcirc$	$\bigcirc$	

#### 17. Avalia o primeiro modo segundo os vários critérios \*

Posições vulneráveis são posições onde a tua unidade ficaria susceptível de ser atacada por um inimigo

Mark only one oval per row.

	1 - Muito mau	2	3	4	5 - Muito bom
Facilidade de compreensão	$\bigcirc$	$\bigcirc$	$\Box$	$\supset$	$\bigcirc$
Rapidez da introdução dos comandos	$\bigcirc$	$\bigcirc$	$\square$	$\supset$	$\bigcirc$
Facilidade em perceber o alcance de cada unidade	$\bigcirc$	$\bigcirc$	$\square$	$\supset$	$\bigcirc$
Facilidade em perceber posições vulneráveis ao mover as unidades	$\bigcirc$	$\bigcirc$	$\square$	$\supset$	$\bigcirc$

#### 18. Avalia o segundo modo segundo os vários critérios \*

Posições vulneráveis são posições onde a tua unidade ficaria susceptível de ser atacada por um inimigo

Mark only one oval per row.

	1 - Muito mau	2	3	4	5 - Muito bom
Facilidade de compreensão	$\bigcirc$	$\bigcirc$	$\square$	$\bigcirc$	$\bigcirc$
Rapidez da introdução dos comandos	$\bigcirc$	$\bigcirc$	$\supset$	$\bigcirc$	$\bigcirc$
Facilidade em perceber o alcance de cada unidade	$\bigcirc$	$\bigcirc$	$\supset$	$\bigcirc$	$\bigcirc$
Facilidade em perceber posições vulneráveis ao mover as unidades	$\bigcirc$	$\bigcirc$	$\supset$	$\bigcirc$	$\bigcirc$

#### 19. Avalia o terceiro modo segundo os vários critérios \*

Posições vulneráveis são posições onde a tua unidade ficaria susceptível de ser atacada por um inimigo

Mark only one oval per row.

	1 - Muito mau	2	3	4	5 - Muito bom
Facilidade de compreensão	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$
Rapidez da introdução dos comandos	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$
Facilidade em perceber o alcance de cada unidade	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$
Facilidade em perceber posições vulneráveis ao mover as unidades	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$	$\bigcirc$

#### 20. Utilizaste o menu de "filtros"?\*

As opções "toggle attack" e "toggle movement" Mark only one oval.

C	$\supset$	Sim
C	$\supset$	Não

21. Se respondeste sim à pergunta anterior, quão útil achaste que foi o menu de filtros? Mark only one oval.



## 22. Deixa aqui alguma sugestão que tenhas acerca da interface do jogo que tenhas

Podes deixar críticas relativas às opções que te foram dadas ou, em alternativa, sugerir novas opções
