

Metodyka

Historia wersji:

Wersja	Data	Autor	Uwagi
1	13.03.2017	Kamila Wysocka	Utworzenie dokumentu
2	01.05.2017	Sebastian Borecki	Dopisanie podziału ról i harmonogramu spotkań

1. Skład sekcji i słownik aliasów

- Franciszek Bednorz - FranioBed
- Sebastian Borecki - Sebik
- Łukasz Diekemper - dixiluk
- Marek Mrowiec - merk
- Zuzanna Musiał - zuzanna
- Kamila Wysocka - kwysocka
- Alicja Zorzycka - Lunatyk

2. Środowisko programistyczne

Unity 5.6.0f3

3. System kontroli wersji - Git

Link do repozytorium: <https://github.com/FranioBed/ManicDragonInsurrection>

Przed commitem upewniamy się, że projekt się kompiluje!

Commity powinny odpowiadać zadaniom na Trello.

4. System śledzenia zadań i błędów - Trello

Link do tablicy: <https://trello.com/b/wcn3h8p9/manic-dragon-insurrection>

Edytować tablicę mogą tylko członkowie dodani do grupy przez **adres e-mail** przez administratora tablicy - **Kamile Wysocką**.

Flow zadania na trello:

- Zadania tworzymy na tablicy *New*
- Programista na początku pracy nad zadaniem przypisuje się do niego (jeśli jeszcze nie jest przypisany) i przeciąga je na listę *In progress*
- Po skończeniu pracy i wypchnięciu commita z wykonanym zadaniem na repozytorium, programista przeciąga zadanie na listę *Ready to test*, wybiera innego członka zespołu do roli testera dla tego zadania i przypisuje go do zadania
- Tester sprawdza czy zadanie zostało wykonane. Jeśli tak - przeciąga zadanie na tablicę *Closed*, jeśli znajdzie jakiś błąd - przeciąga zadanie na tablicę *Replay* i dodaje komentarz opisujący problem
- Podczas pracy nad zadaniem z tablicy *Replay* programista przeciąga je na listę *In progress* i zadanie przechodzi cały flow tak samo jak pochodzące z tablicy *New*

5. Dokumentacja

Link do katalogu na google drive:

<https://drive.google.com/drive/u/0/folders/0B3rywWvIvoTmZzZPOHFQb3pnVkU>

Każdy dokument powinien mieć wpisany na górze numer wersji, datę edycji i autora edycji. Ponadto, w repozytorium znajdzie się osobny katalog, do którego należy wrzucać dokumenty, w celu ich wygodnego wersjonowania.

6. Podział ról

Osoba	Rola	Odpowiedzialności
Sebastian Borecki	Kierownik projektu	Konsultacja konceptu gry z resztą zespołu. Osoba decyzyjna w przypadku konfliktu. Moderacja tablicy na Trello i pracy streamu.
	Designer	Tworzenie całościowej koncepcji gry.
	Programista	Implementacja generacji lochów na mapie
	Architekt	Projektowanie struktury projektu Unity oraz odpowiedzialności klas. Utworzenie szkieletu projektu.
Łukasz Diekemper	Programista	Implementacja rozkładu lochów w obrębie danego etapu.
Franciszek Bednorz	Programista	Implementacja systemu walki i sztucznej inteligencji.
	Designer	Tworzenie całościowej koncepcji gry.
Alicja Zorzycka	Programistka	Implementacja UI i Menu.
	Designer	Zaprojektowanie systemu UI oraz ekwipunku
Marek Mrowiec	Programista	Implementacja systemu ekwipunku
Kamila Wysocka	Programistka	Implementacja systemu walki i sztucznej inteligencji
	Designer	Zaprojektowanie systemu walki i AI.
	Architekt	Projektowanie struktury projektu Unity oraz odpowiedzialności klas.
		Tworzenie dokumentów oraz raportowanie postępu prac

Zuzanna Musiał	Art Designer	Tworzenie assetów graficznych do projektu. Projektowanie GUI, płytek z których będzie składał się poziom, postaci. Design efektów cząsteczkowych.
----------------	--------------	---

7. Tryb pracy

Standardowy tryb pracy polega na zajmowaniu się zadaniami przydzielonymi na tablicy na Trello. Osoby implementujące daną funkcjonalność mogą zamieszczać komentarze pod zadaniem oraz konsultować bieżące kwestie z innymi członkami zespołu za pośrednictwem komunikatora Discord.

Osoby w zespole powinny wykazać inicjatywę. Główne zadania przydzielane są przez kierownika projektu, ale jeżeli dana osoba widzi potrzebę utworzenia jakiejś funkcjonalności sama zakłada zadanie i kontaktuje się z odpowiednimi osobami. W przypadku wątpliwości można zawsze się dopytać lub skonsultować to z dokumentem wizji gry.

Commity dokonujemy do brancha Master, chyba że funkcjonalność dotyczy konkretnego modułu i będzie trwała trochę czasu. Zalecane wtedy jest użycie brancha. Decyzja gdzie commitować zależy od programisty. Wymagana jest kompilacja projektu na branchu Master. Poboczne branche nie muszą się kompilować.

Od maja organizujemy spotkania na grupie Discord w poniedziałki o 19:00, gdzie omawiamy sporne albo niejasne kwestie. Ustalamy jak gra ma wyglądać na poziomie konceptu.

Ponadto organizujemy coś w rodzaju scruma, gdzie każda osoba stara się powiedzieć:

- czym się zajmowała w ostatnim tygodniu,
- co zamierza zrobić / czy wie jakie zadania ma przed sobą,
- jakie rzeczy ją blokują i gdzie potrzebuje pomocy.

Pod koniec spotkania redagujemy dokument postępu prac i dokonujemy modyfikacji bieżących dokumentów wymagań/scenariusza jeśli tego wymagają.