

4. Projekt Bazy Danych

4.1. Opis rozdziału

W tym rozdziale przedstawiono prosty projekt bazy danych dla naszej gry.

4.2. Dane logowania

Na początku zastanówmy się jakie dane będą nam potrzebne do zarejestrowania konta użytkownika? Z reguły jest tak, że podczas logowania podaje się login i/lub adres e-mail oraz hasło. I dane jakie powinniśmy pobierać od użytkownika to właśnie login, e-mail oraz hasło. Ponadto dodajmy pole, które będzie określało status konta czyli czy jest aktywne czy zablokowane oraz pole z datą do której konto ma być nieaktywne w przypadku zablokowania. Dane zbierzmy w tabeli *konto*.

Tabela 4.2.1. Schemat tabeli *konto*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	4	A_I, P_K
login	VARCHAR	20	
haslo	VARCHAR	40	
email	VARCHAR	255	
aktywny	INT	1	DEFAULT(1)
data	DATETIME		DEFAULT(0)

4.3. Dane królestwa

Następnie zbierzmy wszystkie dane jakie będą opisywać poczynania naszego króla krasnali. Możemy je umieścić w jednej tabeli, ale lepiej będzie stworzyć kilka mniejszych bo nie ma sensu wstawiać do jednej tabeli wszystkich parametrów takich jak posiadane surowce, poziomy

budynków oraz ilość jednostek. Zaczniemy więc od tabeli z surowcami. Struktura tabeli będzie bardzo prosta, mianowicie na początku będzie zawierać tylko jedno pole gdzie następnie dodamy kilka kolejnych kolumn które będą numerami poszczególnych surowców. Takie rozwiązanie pozwoli na dynamiczne dodawanie nowych surowców. Tak więc stwórzmy tabelę *surowce*.

Tabela 4.3.1. Schemat tabeli *surowce*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	P_K
<id surowca>	INT	11	DEFAULT(<liczba>)

Wszystkie pola id w tym podrozdziale określają id gracza, do którego należy budynek, jednostka czy surowiec. Następnie zajmijmy się tabelą zawierającą dane o budynkach gospodarczych oraz wojennych. Schemat tej tabeli będzie identyczny jak tabeli z surowcami. Tak więc tabela *budynki* będzie wyglądać tak:

Tabela 4.3.2. Schemat tabeli *budynki*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	P_K
<id budynku>	INT	2	DEFAULT(<0 - 1>)

Sprawa podobna z rozszerzeniem listy jak w poprzedniej tabeli. Na końcu tabela z jednostkami posiadanymi przez użytkownika. Schemat ten sam co poprzednio. Tabele nazwijmy *jednostki*.

Tabela 4.3.3. Schemat tabeli *jednostki*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	P_K
<id jednostki>	INT	11	DEFAULT(<liczba>)

Następnie dodajmy tabelę, która będzie przechowywać datę, która będzie określała czas kiedy rozbudowywany budynek będzie gotowy czy też rekrutowana jednostka zakończy szkolenie.

Tabela 4.3.4. Schemat tabeli *budynki_budowa*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	P_K
<id budynku>	DATETIME		

Tabela 4.3.5. Schemat tabeli *jednostki_rekrutacja*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	P_K
<id jednostki>	DATETIME		

Oraz tabelka która będzie przechowywać godzinę ostatnio odebranego surowca przez gracza.

Tabela 4.3.6. Schemat tabeli *surowce_wydobycie*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	P_K
<id surowca>	DATETIME		

4.4. Opis komponentów

Teraz stwórzmy tabele, które będą zawierać informacje o jednostkach oraz budynkach, takie jak cena, wymagania do rozbudowy czy zdobycia. Tak więc najpierw dodajmy tabelę z opisem budynków.

Tabela 4.4.1. Schemat tabeli *budynki_opis*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	A_I, P_K
nazwa	VARCHAR	20	
prod_pods	INT	5	DEFAULT(0)
prod_lvl	INT	5	DEFAULT(0)
opis	TEXT		
czas	INT	5	DEFAULT(1)
<id surowca>	INT	5	DEFAULT(0)

Gdzie:

- id* – określa identyfikator budynku,
- nazwa* – nazwa budynku,
- prod_podst* – produkcja budynku na podstawowym poziomie,
- prod_lvl* – produkcja zwiększana co poziom,
- opis* – pole zawierające krótki opis budynku,
- czas* – czas rozbudowy budynku na poziom,
- <id_surowca>* – rozbudowywana część tablicy która będzie zawierała informacje na temat surowców wymaganych do rozbudowy czy kupna budynku.

Grafika z miniaturką budynku będzie zapisywana do katalogu *_resources* w odpowiednim katalogu. Nazwa grafiki będzie taka sama jak jej identyfikator np.: „*1.png*”.

Podobnie będzie wyglądać tabela z opisem jednostek:

Tabela 4.4.2. Schemat tabeli *jednostki_opis*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	A_I, P_K
nazwa	VARCHAR	20	
opis	TEXT		
czas	INT	5	DEFAULT(1)
zycie	INT	3	DEFAULT(1)
atak	INT	3	DEFAULT(1)
obrona	INT	3	DEFAULT(1)
inteligencja	INT	3	DEFAULT(1)
s_ <i><id_surowca></i>	INT	4	DEFAULT(0)
b_ <i><id_budynku></i>	INT	2	DEFAULT(0)

Gdzie:

- id* – określa identyfikator jednostki,
- nazwa* – nazwa jednostki,
- opis* – pole zawierające krótki opis jednostki,
- czas* – czas rozbudowy budynku na poziom,
- zycie* – ilość punktów życia jednostki,

atak – ilość punktów ataku jednostki,

obrona – ilość punktów obrony jednostki,

inteligencja – ilość punktów inteligencji jednostki,

s_<id_surowca> – rozbudowywana część tablicy która będzie zawierała informacje na temat surowców wymaganych do rekrutacji jednostki,

b_<id_budynku> – rozbudowywana część tablicy która będzie zawierała informacje na temat poziomów budynków po osiągnięciu których będzie możliwa rekrutacja jednostki.

W przypadku surowców oraz budynków wymaganych do rekrutacji jednostki schemat nazewnictwa jest taki, że znaki przykładowy surowiec o identyfikatorze 1 będzie nazwany w tej tabeli *s_1*, a budynek o tym samym identyfikatorze będzie nazwany *b_1*. Pozwoli to wyeliminowanie kolizji nazw.

Grafika z miniaturką jednostki będzie zapisywana do katalogu *_resources* w odpowiednim katalogu. Nazwa grafiki będzie taka sama jak jej identyfikator np.: „*1.png*”.

Następnie tworzymy tabelę zawierającą opis surowców:

Tabela 4.4.3. Schemat tabeli *surowce_opis*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	A_I, P_K
nazwa	VARCHAR	20	
opis	TEXT		

Gdzie:

id – określa identyfikator surowca,

nazwa – nazwa surowca,

opis – pole zawierające krótki opis surowca.

Grafika z miniaturką surowca będzie zapisywana do katalogu *_resources* w odpowiednim katalogu. Nazwa grafiki będzie taka sama jak jej identyfikator np.: „*1.png*”.

4.5. Raporty

Kolejną tabelą będzie tabela zawierająca raport z bitwy. Zastosujemy tutaj bardzo prostą tabelkę zawierającą tekstowy opis przebiegu walki bez rozbijania tego na mniejsze elementy.

Tabela 4.5.1. Schemat tabeli *raport*

Nazwa pola	Typ	Długość	Dodatkowe
id	INT	11	A_I, P_K
id_napastnika	INT	11	
id_ofiary	INT	11	
zwyciezca	INT	1	0 – napastnik wygrał, 1 – napastnik przegrał
opis	TEXT		
data_roz poczenia	DATETIME		
data_zakonczenia	DATETIME		

Tabela bardzo prosta, id rywali będą stosowane do sortowania czy też wyświetlania raportu u konkretnych graczy, pole zwyciezca określa tylko czy napastnik wygrał i na tej podstawie będzie u konkretnego gracza wygenerowany odpowiednio raport o wygranej lub o przegranej.

4.6. Poczta

Do poczty użyjemy także prostej tabelki.

Tabela 4.6.1. Schemat tabeli *poczta*

Nazwa pola	Typ	Długość	Dodatkowo
id	INT	11	A_I, P_K
id_nadawcy	INT	11	
id_odbiorey	INT	11	
temat	VARCHAR	255	
tresc	TEXT		
data	DATETIME		

4.7. Uzupelnienie tabel

Teraz należało by uzupełnić tabelki na podstawie opisu z pierwszego rozdziału. Czyli kilka insertów i baza będzie gotowa do działania z naszą grą. W pliku „roz4.rar” znajduje się pełen kod dodający i uzupełniający tabele w bazie danych.