

ת.נ.מ - נ.ת.מ  
318403193 'יכ'נ'ס' א.ו.מ  
208638171 ג.ו.מ' ז.ו.מ

Pack 2 Snail Sna

# ערבל להקota במדמה

Pack 4 Snail Balrog Slime Pack 2 Slime Slim

Pack 2 Snail Slim

Pack 2 Pack 2 Pack 2 Balrog Balrog Slime Sna

Pack 3 Slime Snail Sna

Pack 3 Pack 3 Balrog Slime Snail Balrog Balrog

Pack 4 Snail Snail Slime Pack 2 Slime Slim

Pack 2 Snail Slim

2 Pack 3 Pack 2 Balrog Balrog Slime Slime Sna

Pack 2 Snail Sna

Pack 3 Pack 2 Balrog Slime Snail Balrog

Pack 5 Snail Snail Snail Slime Pack 2 Slime Slim

Pack 2 Snail Slim

Pack 2 Pack 2 Pack 2 Balrog Balrog Slime Sna

Pack 2 Snail Sna

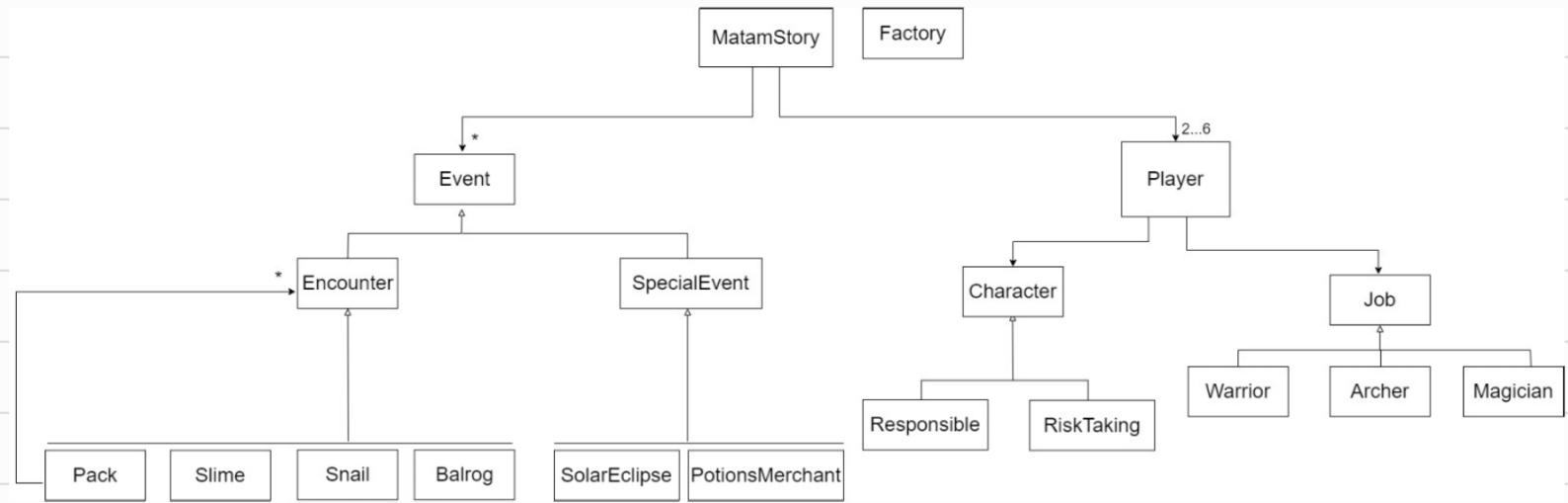
Pack 3 Pack 2 Balrog Slime Snail Balrog

Pack 4 Snail Snail Slime Pack 2 Slime Slim

Pack 2 Snail Sna

imgflip.com

1 Notice



2 Notice

רעיון פול אינטראקטיבי יוניברסלי

Encounter-ן מוגדר כ.pack - Composite (

הנראה כי הוא מוגדר כ多层次י Encounters, ומשתמש בComposite מודל

.Encounter דוחן מוגדר כキャラקטר ורוצח ורוצח

:MINION ANGEL IS ANGEL UNIVERSE - Strategy (2

Player רוצח MINION עלי לירוט בפער בין שמי שמי נושא Job -> (k

:ANGEL . מושג Job -> ANGEL ANGEL מושג לירוט

ANGEL ירכז מושג MINION ANGEL getCombat מושג לירוט

ANGEL מושג רוצח רוצח רוצח . Job מושג מושג לירוט

לפניהם ANGEL ירכז רוצח, רוצח בפער בין שמי שמי Job ->

player. getJob(). getCombat( level , force ) : 2

PotionsMerchant -> רוצח ANGEL ו Character מושג (2

ANGEL רוצח רוצח רוצח ANGEL רוצח רוצח רוצח

Character מושג רוצח ANGEL רוצח ANGEL רוצח

Player. getCharacter. buy(Player)

Player ANGEL Character ANGEL רוצח רוצח ANGEL רוצח

ANGEL רוצח רוצח ANGEL ANGEL ANGEL ANGEL ANGEL ANGEL

וילכ נאנו בז'ה יפכ . Player הוא נ"ל (ההנ'ת) מילבד  
. (פ'ר'ה הוא נ"ל (ההנ'ת) מילבד, כי (ה'ר'ה, ה'ר'ה ע'יק'ה)

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו לינ'ו ג'

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו לינ'ו ג'  
לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו לינ'ו ג'  
לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו לינ'ו ג'  
לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו לינ'ו ג'

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו - Command (3)

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו - Command (3)  
לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו - Command (3)

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו - Command (3)

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו - Command (3)

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו - Command (3)

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו - Factory (4)

לפ'ר'ה יפכ נאנו בז'ה שטרכ'ן strategy-הו - Factory (4)

13) INTRODUCE POLYMORPHISM BY ADDING IS\_ABLE  
 WHICH IS USED FOR. 1311' POLYMORPHIC /& /  
 String & IT IS USEFUL FOR THE CLASS WHICH  
 CAN FIND, CREATE 'IF IT IS'. AND WHICH  
 CAN HANDLE THE SAME FOR THIS CASE.  
 3) USE OF POLYMORPHISM

3 interface  
 : A1A1N1 P1R1N1, M1N1P1R1N1, S1T1N1

Class Rouge: Job {

public:

Rouge();

Rouge(const Rouge&) = delete;

Rouge& operator=(const Rouge&) = delete;

std::uniqueptr<Job> clone() override; }

13) IF YOU WANT TO USE THE SAME JOB CLASS FOR  
 ANOTHER CLASS, YOU CAN USE THE SAME CLASS  
 BUT WITH DIFFERENT METHODS. FOR EXAMPLE,  
 IF YOU WANT TO USE THE SAME CLASS FOR  
 ANOTHER CLASS, YOU CAN USE THE SAME CLASS  
 BUT WITH DIFFERENT METHODS. FOR EXAMPLE,

ACNE ALREADY USES THE SAME CLASS AS  
 .Factory - 2

13) IF YOU WANT TO USE THE SAME CLASS AS  
 FOR THE Job CLASS, YOU CAN USE THE SAME CLASS AS  
 getCombat & OR Warrior

!151



Void Encounter:: Play (Player &) {

    Player.getJob().Fight(\*this, Player &);  
}

void Job:: Fight (Encounter& encounter, Player& player) {  
    Job job;

    if (job == Rouge) {  
        Rouge.fight(encounter, player);  
    }

    else if (job == Ninja) {  
        Ninja.fight(encounter, player);  
    }

    else if (job == Rouge) {  
        Rouge.fight(encounter, player);  
    }

    else if (job == Ninja) {  
        Ninja.fight(encounter, player);  
    }

    else if (job == Mage) {  
        Mage.fight(encounter, player);  
    }

    else if (job == Thief) {  
        Thief.fight(encounter, player);  
    }

    else if (job == Wizard) {  
        Wizard.fight(encounter, player);  
    }

    else {  
        cout << "Unknown job" << endl;  
    }

    cout << "Job selected" << endl;

    cout << "Job selected" << endl;

    cout << "Job selected" << endl;

    cout << "Job selected" << endl;

## Class DivineInspiration : SpecialEvent {

public:

DivineInspiration();

DivineInspiration( const DivineInspiration& ) = delete;

DivineInspiration& operator=( const DivineInspiration& ) = delete;

Void play( Player & ) override;

}

רָאשׁוֹן הַיּוֹם בְּכָל־עֲמָדָה יָקֵנָה  
מִלְחָמָה כִּי־מִנּוּן נֶאֱמַר לְפָנָיכֶם וְכֵן

בְּמִזְבֵּחַ תַּעֲמֹד וְיָדְךָ כְּפָרָה תַּעֲשֶׂה  
לְפָנָיךְ אֱלֹהִים וְעַל־עַצְמָתְךָ תַּעֲמֹד  
בְּמִזְבֵּחַ תַּעֲמֹד וְיָדְךָ כְּפָרָה תַּעֲשֶׂה  
לְפָנָיךְ אֱלֹהִים וְעַל־עַצְמָתְךָ תַּעֲמֹד

Void Player::setJob( uniqueptr<Job> newJob ) {

this->Job = std::move( newJob );  
}

רְאֵיתָ לְפָנֵיךְ Factory::getRandomJob() מֵעַד  
'גָּדוֹלָה שְׁלֵמָה וְעַד אֵלֶיךָ מֵעַד  
size -> יָהִי לְפָנֶיךָ זָמָן לְמַלְאָה  
קִנְאָה יְהִי לְפָנֶיךָ זָמָן לְמַלְאָה  
. וְעַד לְסָאנְדָה זָמָן לְמַלְאָה uniqueptr < Job > יְהִי  
. Factory -> 'גָּדוֹלָה שְׁלֵמָה וְעַד אֵלֶיךָ

: void DivineInspiration::play() מֵעַד  
רְאֵיתָ לְפָנֶיךָ זָמָן לְמַלְאָה

Void DivineInspiration::Play( Player & player ) {

std::unique\_ptr<Job> newJob = Factory::getRandomJob();

player.setJob( std::move( newJob ) );

printDivineInspirationMessage( Player, player.getJob().getName() );

}

!ASI