King of the Hot Dog

Software Project Detailed Design

The project name：<u>King of the Hot Dog</u>

Team   members : <u>覃爽 李慧香 周曼曼 林锵</u>

Content

1.Introduction

1.1 Purpose

On the basis of the design of the outline of this manual, king of the hot dog game of the modules, procedures, subsystems are implemented on the level of the requirements and instructions.

The purpose is to:

(1) Provide basis for coders;

(2) Provide conditions for modification and maintenance;

(3) Project leader will arrange and control the whole process of development according to the requirements of the plan;

(4) Project test team will conduct periodic and summary performance testing and validation according to this plan.

(5) Product implementers in the software development team read and refer to this manual for coding and testing.

1.2 Explanation of terms and abbreviations

Technical terms:

(1) TensorFlow:it is a symbolic mathematical system based on data flow programming, which is widely used in various types of machine science. The programming implementation of the learning algorithm, its predecessor is the neural network algorithm library DistBelief of Google. TensorFlow mention. Available in four different versions of the Python language: CPU version (tensorflow), including GPU acceleration. Versions (tensorflow-gpu), and their daily compiled versions (tf-nightly, tf-nightly gpu)

(2) Pygame: it is a cross-platform Python module designed for video games, including graphics and sound. Based on SDL, it allows real-time video game development without being tied to low-level languages such as machine language

and assembly language.

Abbreviations:

(1) UML: unified modeling language (UML) is a standard modeling language for designing software blueprints

A standardized modeling language for software analysis, design, and programming

1.3 Intended readers of this manual:

(1) Software development team coders;

(2) Software tester;

(3) Teacher;

(4) Project leader and all participants.

1.4 Background

1.4.1 The development environment

The configuration required for product development and maintenance is shown in the table below：

Classification	Name	Version
The operating system	Windows	Windows10
Programming software	Python	3.6
Interface design	PS	Adobe Photoshop CC2017
Environment	PyCharm	2018.1.2

1.4.2 Runtime environment

The conditions required for software operation are shown in the following table:

Classification	Name	Note

The operating system	Windows	Default Windows
Hardware	CPU、The hard disk	Default minimum

1.5 Design and Rules of the Game

1.5.1 Game Mode

1) players select the game mode as "classic mode", "Infinite mode" and "limited time mode".

2) [Classic Mode] : initial 3 HP. The game ends if you eat the bomb or miss the hot dog 3 times.

3) [Time-limited Mode] : the user receives hot dogs within the specified time (60s), and the game ends when the time ends. If a bomb is eaten, the score is reduced by 10.

4) [Infinite Mode] : 3 health and 3 satiety points (full value 10).If the life value of bomb is decreased by 1, if the hot dog is eaten, the integral value and satiety will increase, and the decline speed of hot dog will accelerate with the increase of satiety. When the satiety reaches the maximum, it can be converted into 1 life value, that is, the life value will be increased by 1.When health is 0, the game ends.

1.5.2 Material Preparation

Select materials according to the requirements of the game: select the pictures of characters, bombs and backgrounds to be used, use Photoshop to synthesize and process the pictures into the picture styles and formats that meet the production requirements, download the appropriate music material for later use.

1.6 References

(1)Zhang Haifan. Introduction to software engineering. 5th edition. Tsinghua

university press

(2)Xiao gang et al. Writing with software documents. Tsinghua university press

2. Demand analysis

2.1 The functional requirements

2.1.1 Demand for design

The purpose of this game is to provide a new way of playing when people's life is boring. Let a person reach relaxed and happy, handle affairs efficiency double good condition. It can also train people's finger flexibility, reaction ability, judgment and observation.

According to the functional requirements, the product can be mainly divided into the following three functional modules, and the rule description of each functional module

As follows:

(1)Game control module: including control of the start of the game, pause and left and right movement of characters

(2)Mode selection module: the main function is mode selection. Players can choose classic mode, infinite mode and time-limited mode

(3)Main interface display module: mainly used for menu display, background image display and game entity

(4)Numerical control module: for timing and scoring

(5) Music function module: players can choose to open and close music

2.1.2 UI Design
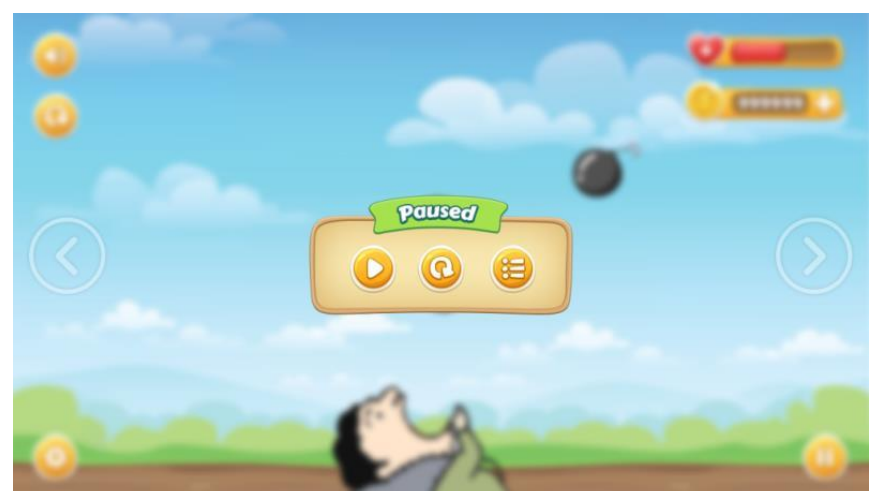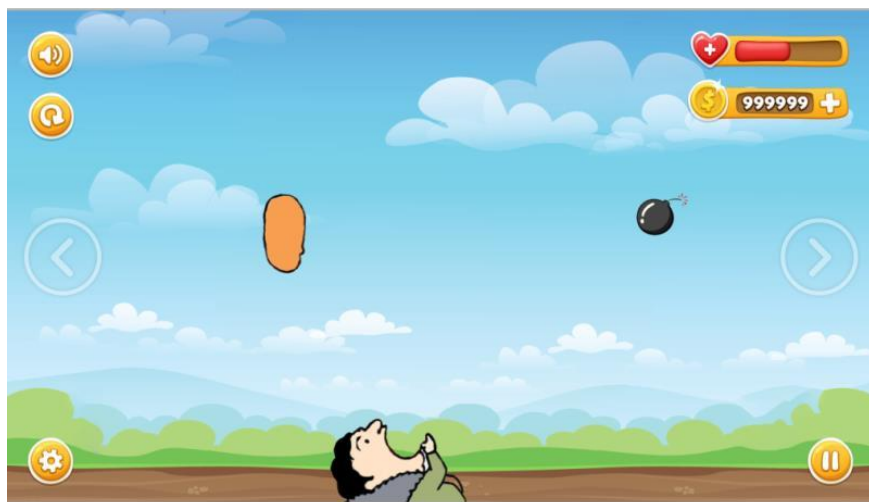
2.1.3 Functional chart
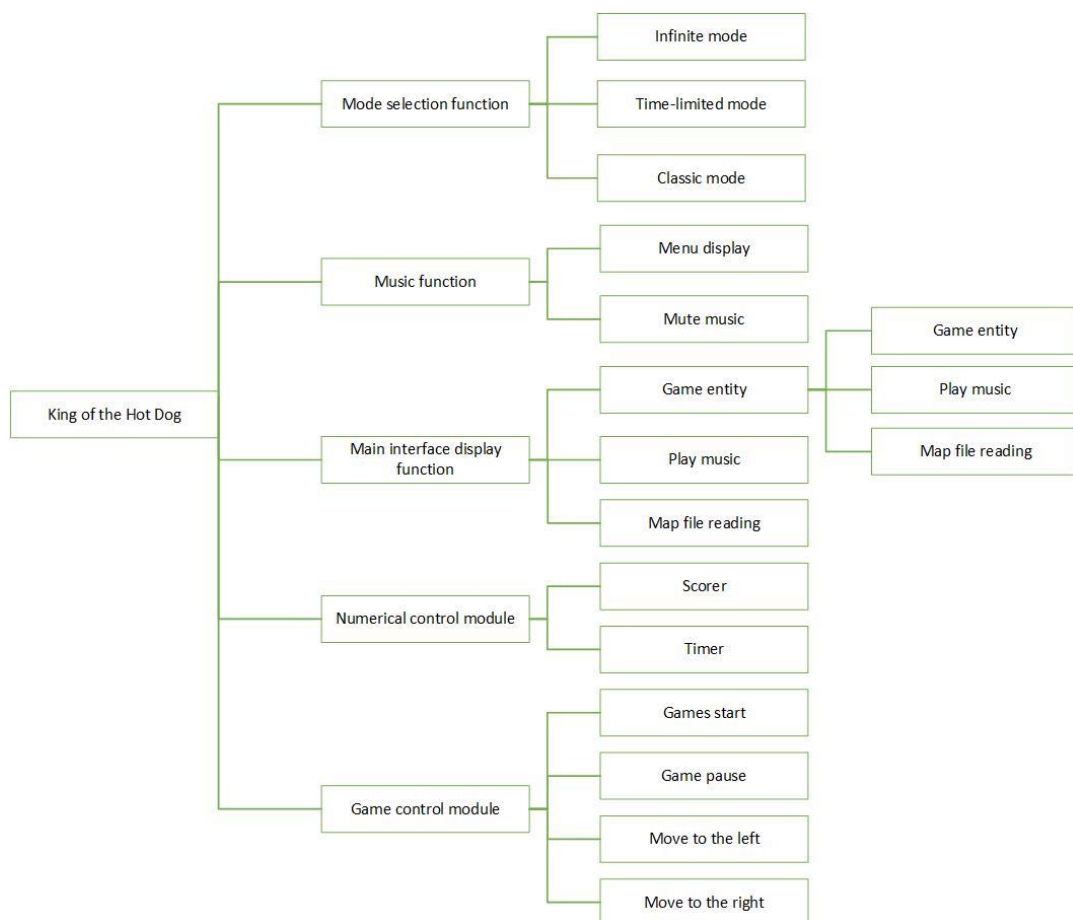
2.2 Non-functional requirements

Non-functional requirements	Detailed requirements
Reliability	The game occupies the system resources little, the running platform request is not high
Performance and efficiency	Fast response time
Interesting	The game page tries to be beautiful and interesting to attract players
Ease of use	Simple operation, can meet the requirements of the public players
Compatibility	Compatibility as strong as possible

2.3 UML

2.3.1 Use case table

Game control module use case table

Use case name   Open the game

participants	The player
The target	Enables the player to start the game
Precondition	The game is running
Basic event stream	1. The player clicks the start button 2. Start the game

Use case name   Pause the game

participants	The player
The target	Allows players to pause the game
Precondition	The game is on
Basic event stream	1. The player clicks the pause button

	2.Pause the game

Use case name	Or so mobile
participants	The player
The target	Move the character left to right
Precondition	The game is on
Basic event stream	1. Start the game 2. Manipulate the left or right buttons to move the character to the left or right

Pattern selection module use case table

Use case name	Mode selection
participants	The player
The target	Players can choose which mode they want to challenge
Precondition	The game has been started and set in classic mode, unlimited mode, limited mode
Basic event stream	1. Players select the desired mode 2. The game enters the mode selected by the player

Numerical control module use case table

| Use case name	The scoring
participants	The player
The target	Keep track of your game score
Precondition	The game is started and the player has selected the desired mode and the score is started
Basic event stream	1. The scorecard records players' scores 2. The value of the player's game score is returned to the file

Use case name	Timing

participants	The player
The target	Record the game time of players
Precondition	The game has started and the player has selected the desired mode and the timer has started
Basic event stream	1. Timer records the game time of players 2. The value of the player's game time is returned to the file

Music function module use case table

Use case name	Close the music
participants	The player
The target	Add fun to the process
Precondition	Game start, music player system open
Basic event stream	1. The player clicks the music button 2. Music stops playing

2.3.2 Use case diagram
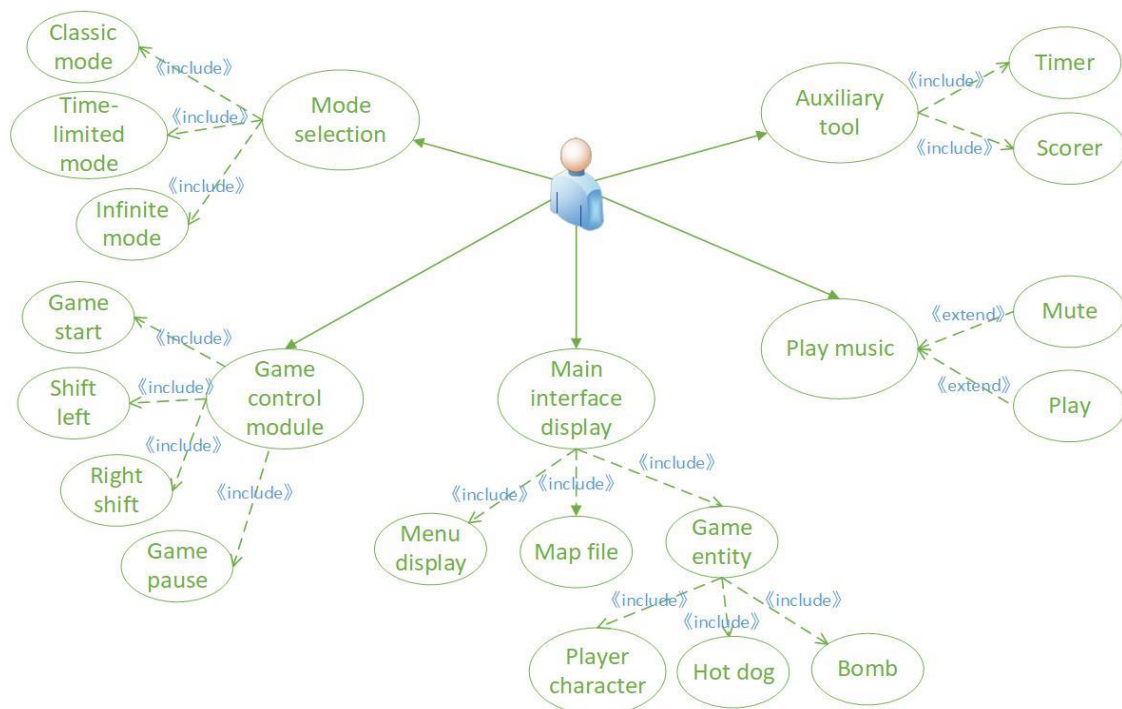
3.Architecture design

3.1 Basic design concept and process flow
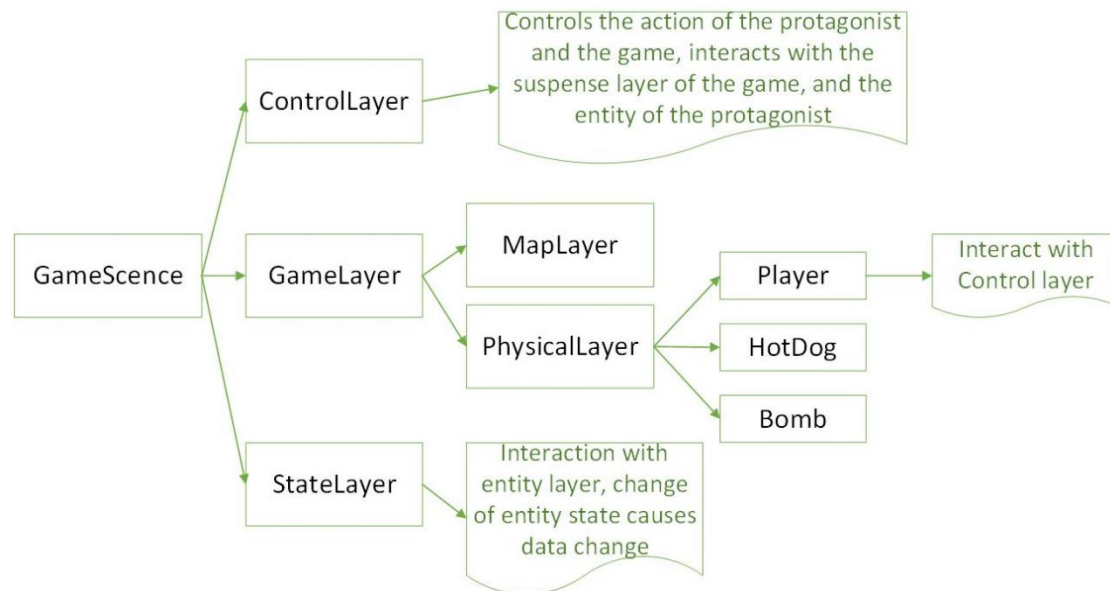
3.2 Software modules

3.2.1Module naming rules

·   Main interface display module：image_display

·   Game control module：game_control

·   Numerical control module：numerical_control

3.3 Function description of each module

·   Main interface display module：

Method names	Create_object
function	Read object parameters to

	generate object groups
The input parameters	The object parameters
The return value	Set of objects
Call a function	NULL

Method names	Updata_object
function	Control object image state, monitor object collision
The input parameters	Object parameters, screen parameters
The return value	NULL
Call a function	check_collide

Method names	Updata_screen
function	Control screen, numerical display
The input parameters	Screen parameters, numerical parameters
The return value	NULL
Call a function	pygame.display

· Game control module:

Method names	check_events
function	Take input events, give feedback, and control roles
The input parameters	Events, role parameters

The return value	NULL
Call a function	Updata_object

Method names	check_play_button
function	Accept input events, control keys
The input parameters	Events, game parameters
The return value	NULL
Call a function	

· Numerical control module：

·

Method names	Get_score
function	Monitor the event and calculate the score
The input parameters	Events, count
The return value	score
Call a function	

Method names	show_score
function	Accept the number and display
The input parameters	Score

The return value	
Call a function	

4. Data structure design

4.1 Logical structure design

The task of logical structure design is to transform the basic e-r diagram designed in the conceptual structure design stage into a logical structure consistent with the data model

The steps of logical structure design are as follows:

(1) convert the conceptual structure into a general relationship, mesh and hierarchical model

(2) transform the transformed relational, mesh and hierarchical models into data models

(3) optimize the data model

4.2 E-R diagram

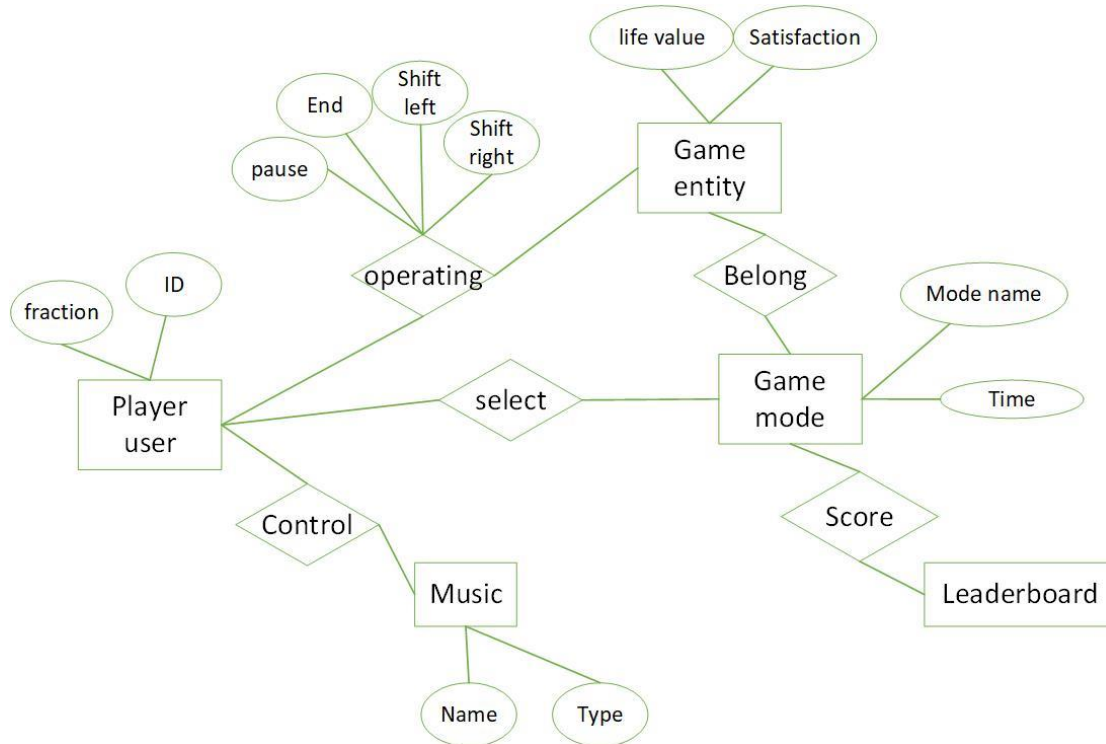The e-r diagram of the game project is as follows:

Converted from e-r diagram to relational mode, as follows:

Player user (ID, score)

Game entities (health, satiety, game mode)

Game mode (mode name, time)

Music (name, type)

4.3 Physical structure design essentials

The physical structure refers to the access method, storage structure and storage location of the database on the storage medium. Physical structure design: refers to the process of designing the best physical environment suitable for the logical structure, i.e. the storage mode, by designing the optimal access method, storage structure and location of the logical structure as well as the reasonable selection of storage medium, etc. according to the results of logical structure design.

(1)Select the access method. Access methods refer to the methods and techniques for users to access database data. Common storage methods include balanced tree, clustering and hash index.

(2)Design the storage structure. Storage structure refers to the index of the logical structure and the data type supported by DBMS, the storage type and length of the determined data items as well as the storage structure of tuples, i.e., data files and The specific storage structure of its data items on the media.

(3)Determine the storage location. Storage location refers to the specific storage location of database files and index files on the media.

(4)Select the storage medium. A storage medium is a physical storage device used to store files. Include, disk, CD, etc.

5. System error handling design

5.1 Error messages

Because the input information does not meet the requirements, for the soft error;

Due to hardware errors (such as: hardware error, etc.), called a hard error;

For data, test documents, to provide the corresponding privacy measures Settings.

Wrong type	Children	The reason for the error
Image display error	The picture is not displayed correctly	Parameter error
The numerical error	Failed to correctly record the score	Object collision detection error
Database error	The connection	Connection timeout
		Connection is broken
	Database itself	Database code error
		Database overflow
System partial customization error	Input error	User input username format error
Connection error	Internal link error	Interface response timeout

5.2 Remedial measures

Indicate the workarounds that may be taken after the failure, including:

(1)For soft errors, first judge the user input information. If the user information is wrong or does not exist after the judgment, give the corresponding error prompt statement and return it to the user.

(2)For hard errors, the error type is not too many and relatively clear, so you can output the corresponding error statement in the possible error, and reset the program, and finally return to the picture supply stage.

(3)Backup technology a backup technology to be used to establish copies to be enabled when the original data is los.t And booting techniques, such as periodically recording disk information.

(4)Recovery the recovery and restart technology that will be used in the startup technology description to enable the software to resume execution from the point of failure. Or a way for software to run from scratch.

6.System maintenance design

The system designed is small, the design time is tight, and there is no extra maintenance module. The system maintenance is relatively simple, basically do not have what big problem, basically be the basic maintenance of the database of docking mark and time is ok. To facilitate maintenance, three types of logs should be designed: system operation log, operation log and error log. Three log roots according to the importance of different levels of the way to take the file and database, system maintenance and management personnel can be very easy monitor the operation of the system.