Game Database

# Database Design Document

Version 1.0

04/07/2018

Document Number: <document’s configuration item control number>

Contract Number: <current contract number of company maintaining document>

Table of Contents

[Database Design Document i](#_Toc511036827)

[Table of Contents ii](#_Toc511036828)

[List of Figures ii](#_Toc511036829)

[List of Tables ii](#_Toc511036830)

[1. Introduction 1](#_Toc511036831)

[1.1 Overview 1](#_Toc511036832)

[1.2 Scope 1](#_Toc511036833)

[2. Overview 2](#_Toc511036834)

[3. Assumptions/Constraints/Risks 3](#_Toc511036835)

[3.1 Assumptions 3](#_Toc511036836)

[3.2 Risks 3](#_Toc511036837)

[4. Design Decisions 4](#_Toc511036838)

[4.1 Key Factors Influencing Design 4](#_Toc511036839)

[4.2 Functional Design Decisions 4](#_Toc511036840)

[4.3 Database Management System Decisions 4](#_Toc511036841)

[4.4 Security and Privacy Design Decisions 4](#_Toc511036842)

[4.5 Performance and Maintenance Design Decisions 4](#_Toc511036843)

[5. Detailed Database Design 6](#_Toc511036844)

[5.1 Data Software Objects and Resultant Data Structures 8](#_Toc511036845)

List of Figures

No table of figures entries found.

List of Tables

## Introduction

### Overview

This document will detail the database for some entities that will belong in a video game. The database will be used to keep track of these entities and have the user add to these tables, to which they can use this for development of their game.

### Scope

The audience for this document is towards developers of video games, as this system will be used to add entities such as heroes, enemies and skills to the database and to each table.

The database will have specific fields when it comes to entering data so it doesn’t exceed what will be capable within the game that will be in development. Each field will have each data preference as to what should be entered, such as name fields only having text or such.

The database will be developed within Microsoft Access, as the organization of the database will be easier and development of the database will be easier than doing it entirely in SQL code.

## Overview

The idea of this database is to specifically to benefit game developers when it comes to making a game and this database should aid in the overall development by helping the user with developing and finalizing ideas for entities for their game.

The database will be designed to function simplistically, as it is supposed to be developed and run in Microsoft Access. The idea of the interface to have it in the program’s ‘Navigational Forms’, which act like a webpage and aims to be simplistic for the user of the database, rather than anything complex. Header tabs will be in place for the user to navigate between each from and the forms can be used by the user; entering values and specific entities against specific preferences.

Microsoft Access was the choice of software when it came to the system’s development, because of the fact that the system can be used by anyone who has access, therefore any user that has Access installed can just simply run the file without any problems and the database will remain the same as when developed.

## Assumptions/Constraints/Risks

### Assumptions

It should be noted that within the database, reports are developed beforehand, meaning that any data submitted within the forms can be updated into the report.

The user should be able within this system to access all the data stored within, as well as edit add and delete.

To run the system, the user will need to have a version of Microsoft Access 2007 or later due to the format. Any earlier version will not be able to run this system due to the different features and formats to that system.

### Risks

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Risk No. | Risk | Probability | Severity | Risk Effect | Mitigation |
| 1. | Scope Creep | 40% | Moderate | The original idea of the system may expand beyond the developers control. | Ideas will be set and seen that they are followed with no major change. Any minor change is allowed, but with careful consideration. |
| 2. | Hardware Failure | 35% | Significant | Issues to hardware or storage becoming corrupt or broken, leading to developing everything from scratch. | Backups of the system, storage peripherals and such will be used in case this risk does happen. |
| 3. | Unauthorized Accessed | 45% | Catastrophic | The database may be attacked and the data could be compromised. | Have security measures placed on the system, only allowing certain users access to the file. |

## Design Decisions

### Key Factors Influencing Design

The whole structure of the database basically has the entities fall under the different categories; the heroes, enemies, skills and the skills of the heroes and enemies respectively. It was important that the design of the database can accommodate these tables and have the data edited and to do this, forms and reports were created so the user can access and edit the data.

Simplicity within the system rather than the database was more the main goal, so the user wouldn’t have to focus too much of organization of the database and more on developing their work. The idea was to develop an interface that allowed for simple use and not just to organize databases. This should also make this system a reliable alternative to Excel or another spreadsheet software as well as other database software.

### Functional Design Decisions

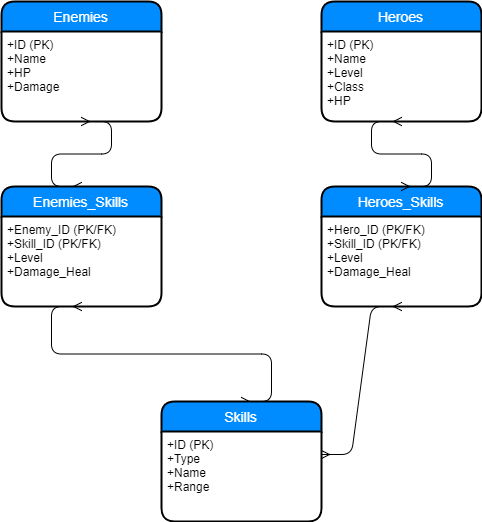
The database will function in homepage view, which is like how a webpage functions. Tabs will aid in navigation between different forms that correlate to the different tables in the database, as well as the different reports in the database.

The forms will allow for data input. This will allow for the user to edit data of each field, but preferences will be set so that the user doesn’t put in invalid inputs by accident and that the values are in respectable limits respective to the game in development. The way the user can add, update and remove data will be simple; just two buttons will help with entering the data of what the user wants. An add/update button will help with the user simply add or update, rather than having separate buttons which would unnecessarily confuse the user.

### Security and Privacy Design Decisions

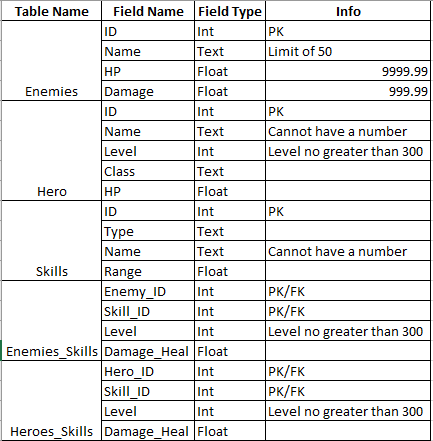
The database will have a small level security layer; the file will be password guarded. The reason will be so that only the users working on the game will have dedicated access to it. Should this be a weak attempt, user access control will help selecting accounts as to who will be able to open and use this system, thus providing more of a security layer so that any chance of unauthorized access will be very low.

## Detailed Database Design



Above is the ERD for the database system. Detailed here are the entities that are within the system and the fields for each entity, such as the ID, Name, etc. and all vary depending on the table that the entity has. The diagram here shows how each system relates to each other with the connection that defines the relationship as “many-to-many”.

I created the diagram using the site Draw.io, a site that allows the creation of charts. This site has many tools to help create numerous flowcharts of any kind as well as chart formatting, page settings and ability to link to your GitHub account and Repo. The site is entirely free without having to sign up. Using this diagram, I could create a database using the entities in said diagram.



The figure above is the data dictionary, containing the entities and fields designed in the ERD above. Each field in the table has different types that are associated with said fields. Detailed here are Int, Float and Text types that correlate to each field, as well as information that defines each preference to the fields, such as limits to health and levels, and points out each primary key; the IDs.

Appendix A: Data Flow Diagrams

Appendix B: Flowcharts