

Padification SDD

Date: 2017-07-14

Version: 0.4.3

McKenzie College SWTS2102

1. Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author | Ver. | Revision Notes |
| 2017-06-19 | William Gale | 0.1 | Heading layout for document. Integration of SQL TCP/IP configuration and a quick development tool outline prototype. |
| 2017-06-21 | Elie Godbout | 0.2 | Updated contents of Appendices; Added Visual aid & enhanced the process information. Added a List of Images to SDD. |
| 2017-06-23 | Elie Godbout | 0.3 | Added Steps for creation of the PADification database and prerequisites for the process in Appendices. |
| 2017-06-26 | Zachary Blue | 0.4 | Added images for downloading of OBDC and SQLCMD |
| 2017-06-26 | Elie Godbout | 0.4.1 | Updated the Steps for installations of ODBC and SQLCMD in Appendices. |
| 2017-07-13 | Elie Godbout | 0.4.2 | Added tools to the development tool outline, updated steps for create the PADification database and added steps to delete the PADification database. Added contents to section 2.8 Interface viewpoint. |
| 2017-07-14 | Elie Godbout | 0.4.3 | Added contents to sections 2.1 Introduction, 2.2 Context viewpoint, 2.3 Composition viewpoint, 2.4 Logical viewpoint, 2.5 Dependency viewpoint, 2.6 Information viewpoint and 2.9 structure viewpoint. |

1. Table of Contents

[I. Revision History 2](#_Toc487803660)

[II. Table of Contents 3](#_Toc487803661)

[III. List of Images 5](#_Toc487803662)

[IV. List of Tables 7](#_Toc487803663)

[1 Development Tool Outline 8](#_Toc487803664)

[2 Design Viewpoints 8](#_Toc487803665)

[2.1 Introduction 8](#_Toc487803666)

[2.2 Context Viewpoint 8](#_Toc487803667)

[2.2.1 User Use Case Diagrams 9](#_Toc487803668)

[2.3 Composition Viewpoint 9](#_Toc487803669)

[2.3.1 Deployment Diagram 9](#_Toc487803670)

[2.4 Logical Viewpoint 9](#_Toc487803671)

[2.5 Dependency Viewpoint 9](#_Toc487803672)

[2.6 Information Viewpoint 10](#_Toc487803673)

[2.6.1 Information gathering 11](#_Toc487803674)

[2.7 Patterns Viewpoint 11](#_Toc487803675)

[2.8 Interface Viewpoint 12](#_Toc487803676)

[2.8.1 Login screen 12](#_Toc487803677)

[2.8.2 Account creation screen 13](#_Toc487803678)

[2.8.3 Home screen 14](#_Toc487803679)

[2.8.4 Player collection 15](#_Toc487803680)

[2.8.5 Edit Monster screen 16](#_Toc487803681)

[2.8.6 Monster Book screen 17](#_Toc487803682)

[2.8.7 My Teams screen 18](#_Toc487803683)

[2.8.8 Edit team screen 2](#_Toc487803684)

[2.8.9 Community screen 3](#_Toc487803685)

[2.8.10 Team Ranking screen 3](#_Toc487803686)

[2.8.11 Account information screen 3](#_Toc487803687)

[2.9 Structure Viewpoint 3](#_Toc487803688)

[2.9.1 MonsterAttribute table 3](#_Toc487803689)

[2.9.2 MonsterType table 3](#_Toc487803690)

[2.9.3 ActiveSkill table 4](#_Toc487803691)

[2.9.4 LeaderSkill table 4](#_Toc487803692)

[2.9.5 AwokenSkill table 4](#_Toc487803693)

[2.9.6 AwokenSkillList table 5](#_Toc487803694)

[2.9.7 LatentSkill table 5](#_Toc487803695)

[2.9.8 LatentSkillList table 6](#_Toc487803696)

[2.9.9 MonsterClass table 6](#_Toc487803697)

[2.9.10 EvolutionTree table 8](#_Toc487803698)

[2.9.11 Player table 8](#_Toc487803699)

[2.9.12 Follower table 9](#_Toc487803700)

[2.9.13 MonsterInstance table 9](#_Toc487803701)

[2.9.14 Team table 10](#_Toc487803702)

[2.9.15 AwokenBadge table 10](#_Toc487803703)

[2.10 Interactions Viewpoint 11](#_Toc487803704)

[2.11 State Dynamics Viewpoint 11](#_Toc487803705)

[2.12 Algorithm Viewpoint 11](#_Toc487803706)

[2.13 Resources 11](#_Toc487803707)

[3 Appendices 11](#_Toc487803708)

[3.1 Tool Configuration 11](#_Toc487803709)

[3.1.1 Configure MSSQL DB for Remote Connections. 11](#_Toc487803710)

[3.1.2 Create the Database from .bat files 22](#_Toc487803711)

1. List of Images

[Figure 2.2‑1: User use case diagram 9](#_Toc487803712)

[Figure 2.8‑1: PADification Login screen 12](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803713)

[Figure 2.8‑2: PADification account creation screen 13](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803714)

[Figure 2.8‑3: PADification Home screen 14](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803715)

[Figure 2.8‑4: PADification Player collection screen 15](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803716)

[Figure 2.8‑5: PADification Edit monster screen 16](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803717)

[Figure 2.8‑6: PADification Monster book screen 17](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803718)

[Figure 2.8‑7: PADification My teams screen 18](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803719)

[Figure 2.8‑8: PADification Edit team screen 2](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803720)

[Figure 2.8‑9: PADification Account screen 3](file:///C:\Users\elieg\Documents\PADification\Docs\SDD\PADification%20SDD.docx#_Toc487803721)

[Figure 3.1‑1: Open SQL server Configuration 11](#_Toc487803722)

[Figure 3.1‑2: SQL server configuration; Protocols for MSSQLSERVER 11](#_Toc487803723)

[Figure 3.1‑3: SQL server Configuration Enable TCP/TP 12](#_Toc487803724)

[Figure 3.1‑4: SQL server Configuration Server services 12](#_Toc487803725)

[Figure 3.1‑5: SQL server Configuration restart MSSQLSERVER 12](#_Toc487803726)

[Figure 3.1‑6: MSSQL SMS New log in 13](#_Toc487803727)

[Figure 3.1‑7: MSSQL SMS New log in SQL Server Authentication 13](#_Toc487803728)

[Figure 3.1‑8: MSSQL SMS New log in name/password/Enforce Password Policy 13](#_Toc487803729)

[Figure 3.1‑9: MSSQL SMS New log in Server Roles 14](#_Toc487803730)

[Figure 3.1‑10: MSSQL SMS New log in Server Roles SysAdmin 14](#_Toc487803731)

[Figure 3.1‑11: MSSQL SMS Server Properties 15](#_Toc487803732)

[Figure 3.1‑12: MSSQL SMS Server properties Security 15](#_Toc487803733)

[Figure 3.1‑13: MSSQL SMS Server Properties SQL Server and Windows Authentication mode 16](#_Toc487803734)

[Figure 3.1‑14: Windows Firewall 16](#_Toc487803735)

[Figure 3.1‑15: Windows Firewall Advanced settings 17](#_Toc487803736)

[Figure 3.1‑16: Windows Firewall Inbound Rules 17](#_Toc487803737)

[Figure 3.1‑17: Windows Firewall Inbound Rules/New Rules 17](#_Toc487803738)

[Figure 3.1‑18: Windows Firewall Inbound Rules/New Rules/Port 18](#_Toc487803739)

[Figure 3.1‑19: Windows Firewall Inbound Rules/New Rules/Specified local port 18](#_Toc487803740)

[Figure 3.1‑20: Windows Firewall Inbound Rules/New Rules/Allow connection 19](#_Toc487803741)

[Figure 3.1‑21: Windows Firewall Inbound Rules/New Rules/Profile 19](#_Toc487803742)

[Figure 3.1‑22: Windows Firewall Inbound Rules/New Rules/Name 19](#_Toc487803743)

[Figure 3.1‑23: Windows Firewall Outbound Rules 20](#_Toc487803744)

[Figure 3.1‑24: Windows Firewall Outbound Rules/New Rules 20](#_Toc487803745)

[Figure 3.1‑25: Windows Firewall Outbound Rules/New Rules/Port 20](#_Toc487803746)

[Figure 3.1‑26: Windows Firewall Outbound Rules/New Rules/Specific remote ports 21](#_Toc487803747)

[Figure 3.1‑27: Windows Firewall Outbound Rules/New Rules/Allow connection 21](#_Toc487803748)

[Figure 3.1‑28: Windows Firewall Outbound Rules/New Rules/Profile 21](#_Toc487803749)

[Figure 3.1‑29: Windows Firewall Outbound Rules/New Rules/Name 22](#_Toc487803750)

[Figure 3.1‑30: Download Screen 22](#_Toc487803751)

[Figure 3.1‑31: ODBC Select Download 23](#_Toc487803752)

[Figure 3.1‑32: ODBC 1st Screen of Install 23](#_Toc487803753)

[Figure 3.1‑33: OBDC License Agreement 23](#_Toc487803754)

[Figure 3.1‑34: ODBC Feature Selection 24](#_Toc487803755)

[Figure 3.1‑35: ODBC Begin Installation 24](#_Toc487803756)

[Figure 3.1‑36: ODBC Install Screen 24](#_Toc487803757)

[Figure 3.1‑37: Notification of Restart 25](#_Toc487803758)

[Figure 3.1‑38: SQLCMD Download Page 25](#_Toc487803759)

[Figure 3.1‑39: SQLCMD Choose Download 25](#_Toc487803760)

[Figure 3.1‑40: SQLCMD First Page of Installation 26](#_Toc487803761)

[Figure 3.1‑41: SQLCMD Accept License Terms 26](#_Toc487803762)

[Figure 3.1‑42: SQLCMD Confirm Installation 27](#_Toc487803763)

[Figure 3.1‑43: SQLCMD Finish Installation 27](#_Toc487803764)

1. List of Tables

[Table 2.9‑1: Monster attribute 3](#_Toc487803765)

[Table 2.9‑2: Monster type 3](#_Toc487803766)

[Table 2.9‑3: Active skill 4](#_Toc487803767)

[Table 2.9‑4: Leader skill 4](#_Toc487803768)

[Table 2.9‑5: Awoken skill 4](#_Toc487803769)

[Table 2.9‑6: Awoken skill list 5](#_Toc487803770)

[Table 2.9‑7: Latent awoken skill 5](#_Toc487803771)

[Table 2.9‑8: Latent awoken skill list 6](#_Toc487803772)

[Table 2.9‑9: Monster class 7](#_Toc487803773)

[Table 2.9‑10: Evolution tree 8](#_Toc487803774)

[Table 2.9‑11: Player 8](#_Toc487803775)

[Table 2.9‑12: Follower 9](#_Toc487803776)

[Table 2.9‑13: Monster instance 10](#_Toc487803777)

[Table 2.9‑14: Team 10](#_Toc487803778)

[Table 2.9‑15: Awoken badge 10](#_Toc487803779)

# Development Tool Outline

|  |  |
| --- | --- |
| Development Language | Python 3.5.2 |
| Development Environment | Visual Studio 2015 |
| Documentation | Word 2016 |
| UML Design Tool | UMLet |
| Relational Database Environment | MSSQL 2014 |
| RDMS | Microsoft SQL server 2014 Management Studio |
| ERD Design Tool | ERD Concepts 7 |
| Python Modules | Tkinter  Pypyodbc  Pygubu  Pillow |
| Testing tools | Unity test |
| Test management tool | TestRail |
| API | Microsoft ODBC Driver 13.1 |
| Command-line tool | SQLCMD Utilities 13.1 |

# Design Viewpoints

## Introduction

In this section, we will explore the concepts and structure of the PADification product as well as the product’s intended purpose, audience, functionality, processes and maintenance requirements.

## Context Viewpoint

The PADification product is a software application designed to be a separate extension of an existing product known as Puzzles and Dragons. It serves to be a tool for players of the mobile video-game where they will be able to manage and modify monsters from the game to give players more options when it comes to planning combinations of monsters for teams and give players more information about the potential of each team they build all on a separate platform so players need not worry about compromising their personal Puzzles and Dragons accounts. Furthermore, the PADification product aims to allow players to share the discoveries they’ve made with others so the community may help one another improve their strategies when playing Puzzles and Dragons.

### User Use Case Diagrams

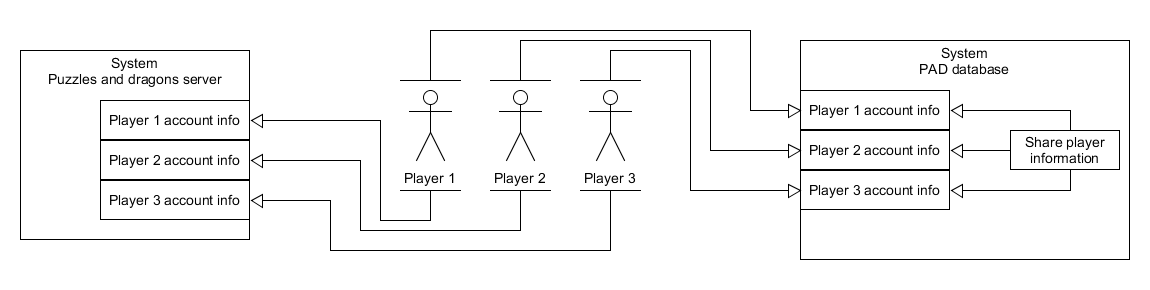


Figure ‑: User use case diagram

## Composition Viewpoint

The PADification Product is comprised of functions built using Python source code and MSQL server database management. The two, function together to allow players to manage a wide array of monsters, teams and other player information and saving all these modifications in a database. By using a UI design using Pygubu, Tkinter and Python functionality, the Python functions perform queries onto the PADification database to update its contents. The PADification database is create with queries all run by using a single batch file and the contents are populated automatically.

In the events an update is required, the Database will require a manual update to the tables and/or records. This could affect the functions found in the Python code as well, features of the UI may need to be redesigned alongside these modifications.

### Deployment Diagram

## Logical Viewpoint

The PADification product is inspired by the existing product known as Puzzles and Dragons, a mobile video-game developed by Gungho entertainment Inc. The PADification product aims to simulate aspects found in Puzzles and Dragons and greatly expand the functions. The calculations found in Puzzles and Dragons have been used in the PADification product to accurately replicate the features from the video-game. While Puzzles and Dragons is the inspiration and basis for the PADification product, the two systems do not interact directly.

## Dependency Viewpoint

The PADification is built upon interactions between MSQL server database and Python executable files all grouped in a Python project. The product will perform constant exchanges between the PADification database with the use of queries found in the Python code. The product requires a functioning PADification database before any of the executions can be performed. The product also requires the Python project sets the PADification.py file as the Startup file.

## Information Viewpoint

The PADification product includes a **Help** section for users. The help section will contain information about how to use the product. This section will contain the following;

* Required formatting for account creation
  + Your email is required to be an existing email address.
  + Your password can be populated with alphanumeric content.
  + You player ID will only allow numeric content. (If you have a Puzzles and Dragons account, we recommend using the same ID number.)
  + Your Username can be populated with alphanumeric values.
* Searching for monsters
  + Go to the Monster book section. Select search with the search field blank to view the list of all available monsters. Select the search field and input the monster name or ID number to search for that monster. You may also search for a range of monsters by inputting two ID numbers separated by a comma “,”. Search filters are available in order to restrict the types of monsters that will appear in the result. Monsters are displayed in limited numbers by pages. You can cycle through the pages by selecting the right and left buttons. You may input a page number in order to reach that page quickly.
* Adding monsters (normal and WishList)
  + Go into the Monster Book section. Search for the monster you wish to add to our collection. The monster will be displayed in the search result list. Select your desired monster. Select the **Add to collection** button. The monster will now be found in the Player collection section.
  + To add monsters as Wish listed monsters, go to the Monster Book section. Search for the monster you wish to add to our collection. The monster will be displayed in the search result list. Select your desired monster. Select the **Wishlist** button. The monster will now be found in the Player collection section under the Wishlist label. In addition, users have access to stat modifiers for the Wishlist monster. Changing these will affect the monster’s stats when transferred to the player collection.
* What are Wish listed monsters?
  + Wish listed monsters are monsters with a Wishlist label. These monsters are intended to be monsters a player desires but has yet to obtain.
* What is monster editing?
  + PADification allows users to customize the properties of their monsters. This feature was created so players can recreate the monsters they own and/or create monsters with properties they wish to achieve.
* Modifying monsters
  + From the Player collection screen, select the monster you wish to edit. Select the **Edit monster** button. This will send you to the Edit monster screen. In this screen, you will be able to modify almost all of a monster’s properties within the predetermined limits. By selecting the **Apply changes button**, you will save all the changes you have made to that monster. If you leave this page without applying your changes, the monster’s properties will default to the previous instance.
* Creating teams
  + Go to the My Team section. Select the **Add new team** or **Edit team** buttons. (Edit team will edit the team that is currently selected in the team list) This will send you to the Edit team screen. Select the slot you wish to add a monster into. Select the monster and it will be added to the slot. You may change the team name by selecting the field and inputting a new name. You may also add an awoken badge to your team by selecting the **awoken badge** button. Select the **Save team** button to save all changes done to your team. If you leave this page without saving the changes, the team will default to its previous instance.
* Searching for and viewing other players
  + Go to the community screen.
* Follower and following
  + If you find someone whom you feel you would like to see again, select the **Follow** button on their player view page. This will add them to your following list accessible from the Home screen. Players whom have decided to follow you will appear in the Follower list accessible from the Home screen.
* Team submit/ deleting teams
  + From the My Teams section, you may select the **Submit team** button. This will send your team instance to a repertoire of teams built by many different people using the PADification product. This will allow other people to rate your teams and compare them to other player’s teams.
* Team ranking
  + A team will only be able to receive a Rank after it has been submitted. If a team is removed from the ranking list or if the team is deleted, the Rank will be deleted. Ranks are the result of other people using PADification choosing to add rate teams they find in the Team Rank section. The Ranks allow players to showcase and discuss the potential of teams based on how many rates it has received.

A section in the home screen of the PADification product is reserved for displaying messages about the current progression and updates about the product.

### Information gathering

## Patterns Viewpoint

## Interface Viewpoint

The user interface for the PADification product is comprised of several screens all interactable with the use of a PC’s cursor and keyboard.

### Login screen



Figure 2.8‑1: PADification Login screen

The log in screen is the first screen to be displayed when users first execute the product. This screen serves as a security check for users as well as a primary way of differentiating each user by inputting unique combinations of credentials. This keeps the contents of PADification separate from each user so no overlap with each other’s information is encountered.

**Email & password fields**

These fields must be populated with not only existing information located in the PADification database, but the data inputted in each field must match the corresponding combination in the database in order to log in successfully. Any alterations will result in an unsuccessful login.

**Login button**

This button enables the check for appropriate login credentials. If both Email and password fields match a combination in the database, the user will proceed to that account’s information. If the data inputted in those two fields are invalid however, pressing the Login button will result in a message pop-up window stating invalid data has been inputted and the user will not be able to proceed.

**Create account button**

This button will send the user to the account creation screen.

### Account creation screen



Figure 2.8‑2: PADification account creation screen

On this screen, user will be able to create an account for PADification so they may login and use the application.

**Account creation credentials**

In order to create an account, users must input data in each of the following fields;

* Player ID
* Username
* Password
* Confirm password
* Email

Player ID must be numeric content only and is a way to represent the user among everyone else. Username adds alphanumeric content in order to represent the user in any way they choose. Password and confirm Password are important as they are required to access an account when logging in. Both fields must be exactly the same in order to complete an account creation. Email is the last critical field as it is used in conjunction with password for logging in.

**Create account button**

This button will check all the inputted credentials. If all requirements are met, a new PADification account will be created and the user will be sent to the login screen where they will be able to input their email and password to log into their account and use the PADification product. If the credentials are not satisfactory, the account will not be created and a message popup window stating the fields are invalid.

**Cancel button**

This button will send the user back to the Login screen and undo any changes done to the account creation fields.

### Home screen



Figure 2.8‑3: PADification Home screen

The home screen is the first screen to load once a player has logged in. From the Home screen, a player will have access to buttons relating to every other major screen available within the product. These buttons are known as the Navigation buttons;

* Home screen
* Player collection
* Monster Book
* Edit Team
* Community
* Team Rank
* Account

The Navigation buttons are present on all major screens titled by the same names as the titles of the buttons listed prior.

**PADnews window**

The home screen offers windows for giving information about the current events or updates to the product.

**Follower & Following windows**

Players will also have access to a list of other players under the follower window as well as players whom the player is currently following in the following window. By selecting a player profile in one of these windows, the user will be sent to that player’s community information screen.

### Player collection

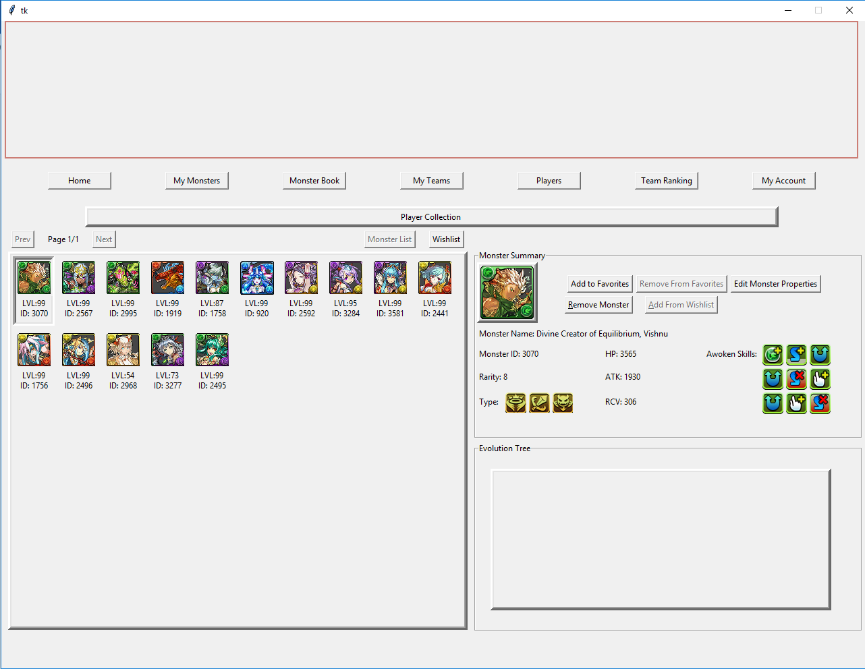


Figure 2.8‑4: PADification Player collection screen

The player collection screen is a major component of the PADification product. This screen contains a list of the instances of monsters added from the monster book screen by users. Selecting a monster will showcase that monster’s properties as well as a list of monsters relating to it by way of evolution or previous evolutions known as the evolution tree.

**Favorites/Add from Wishlist**

Depending on the specific monster selected, - this button will toggle between 2 purposes; enable/disable Favorites on a monster and adding a monster labelled as WishList monster to owned monster.

* As the **Favorites button**, it will give users the ability to set monsters under a “Favorite” label which will give them extra sorting properties as well as protect them from being deleted. When a Monster is already under the “Favorites” label, this button will serve as a way to undo the label and return the monster to an “owned” or simply a monster with no added sorting functionality.
* As the **Add from Wishlist button**, it will only be available when a monster has been added to the player collection from the monster book screen as a Wishlist monster meaning; a monster the player desires but has yet to obtain. This gives the monster a “Wishlist” label and by pressing this button, the monster will be changed to an “owned” status.

**Edit Monster properties**

This button will send the user to the edit monster screen.

**Remove/delete monster**

This button will allow a user to discard a monster from their list for whatever reason they may have. A prompt will appear when a user presses this button. This prompt will ask them to confirm the process before the monster can be deleted. Once a monster is deleted, its specific instance cannot be retrieved.

### Edit Monster screen

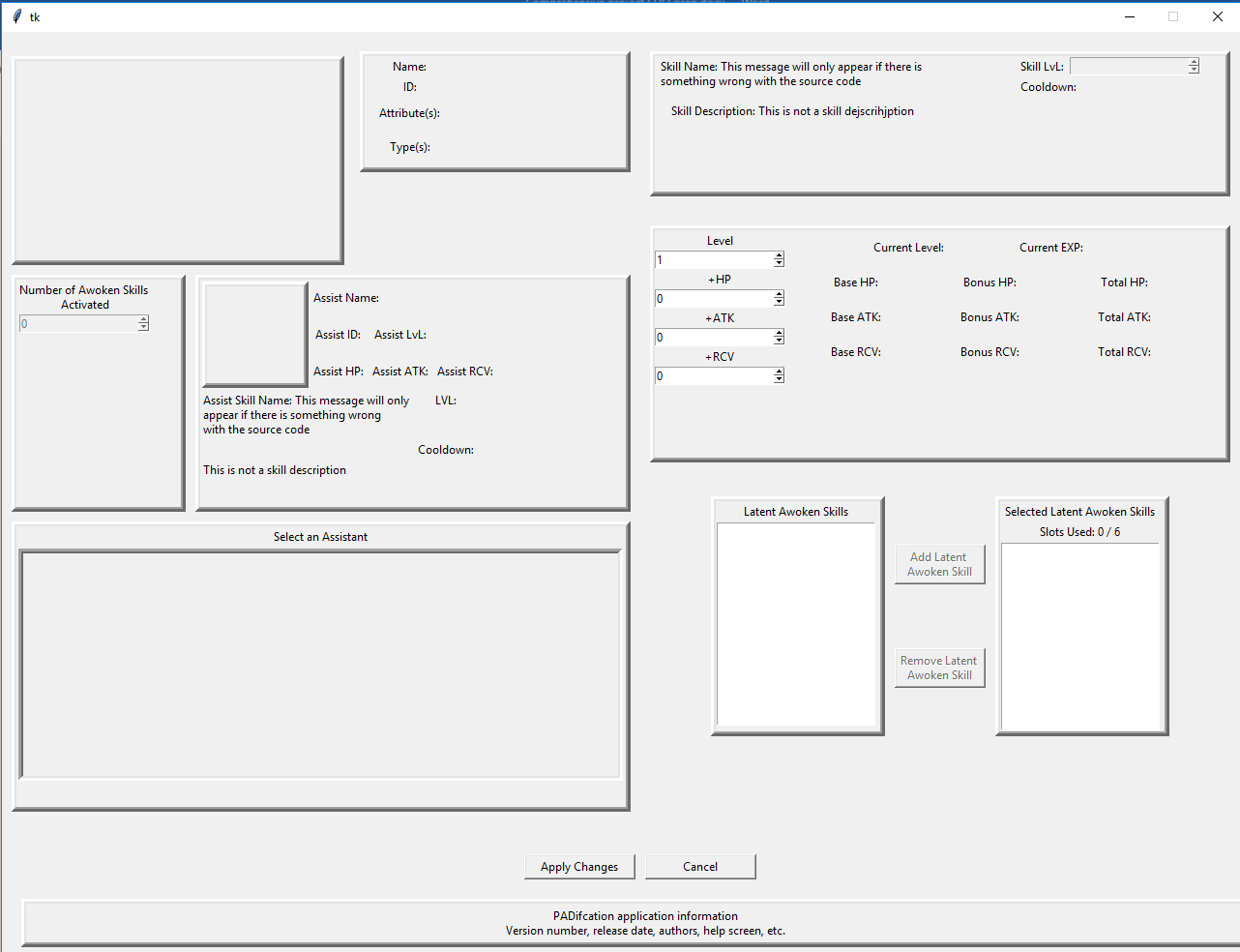


Figure ‑: PADification Edit monster screen

The edit monster screen allows users to not only view a monster’s properties, but to change them as they see fit. This screen offers users a view of the monster’s properties being;

* Name
* ID number
* Rarity
* Cost
* Active skill
* Leader skill
* Experience curve
* Coin value
* Monster points

The remaining properties of the monsters are available to be modified.

**Stats**

* Level: users can change the monster’s level from the monster’s minimum level to its maximum level.
* +HP: users can change the monster’s +HP stat from 0 to 99.
* +ATK: users can change the monster’s +ATK stat from 0 to 99.
* +RCV: users can change the monster’s +RCV stat from 0 to 99.

**Active Skill Level**

* Users can change a monster’s active skill level from its minimum level to its maximum level.

**Awoken Skills**

* Users can modify a monster’s list of Awoken skills. They will be able to decide how many Awoken skills will be awoken or not. (This feature will conform to the functionality of Puzzles and Dragons: Awoken skills can only be awoken in a linear fashion.)

**Latent Awoken Skills**

* Users can modify which Latent Awoken Skills are added to a monster within the limits of available slots. Users will also be able to activate the extra slot if they so choose.

**Assists**

* Users can modify the assist functions to a monster by using another monster from the user’s collection.

**Apply changes button**

* When a user performs changes to a monster, the changes will only be saved when a user presses this button. Otherwise the changes will be lost when a user changes screens or exits the product.

**Delete Monster button**

* This button will allow a user to discard a monster from their list for whatever reason they may have. A prompt will appear when a user selects this button. This prompt will ask them to confirm the process before the monster can be deleted. This process cannot be reverted.

**Return button**

* This button sends the user back to the player collection screen.

### Monster Book screen

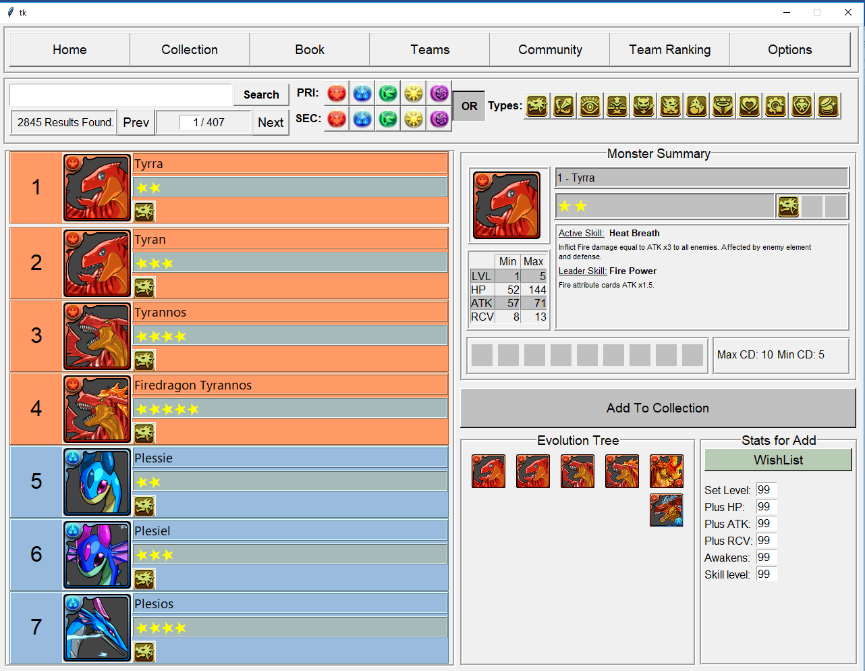


Figure 2.8‑6: PADification Monster book screen

The monster book screen allows users to view every monster available in the North American version of Puzzles and Dragons. The list contains interactable profiles of monsters. When a user selects a monster, that monster’s information will be displayed along with the evolution tree for the monster.

**Search bar**

Users can search for a specific monster by entering the monster’s name or ID number in the search bar and selecting the search button. Users also have access to a feature where they may input a specific page number in order to quickly jump to that page.

**Search filters**

Users have access to filters to help them search for specific monsters. They will be able to search by Attributes both primary and secondary as well as with the monster types.

**Hover features**

When a user hovers their cursor over icons such as monster type and awoken skills, a message will be displayed informing the user with the name of the specific icon they are hovering over. In addition, if a user hovers over the monster’s image, a larger fully detailed image will be displayed.

**Add monster to collection**

Users will be able to add the selected monster to their collection by clicking the “add monster to collection” button. This will send the selected monster to the collection under the “owned” label.

**Add monster to Wishlist**

Users will be able to add the selected monster to their collection by clicking the “add monster to collection” button. This will send the selected monster to the collection under the “Wishlist” label. In addition to adding a monster to the player collection, users will have access to a few properties to modify before adding the monster. These are the modifiable fields:

* Level
* +HP
* +ATK
* +RCV
* Awoken Skills awoken
* Active skill level

### My Teams screen

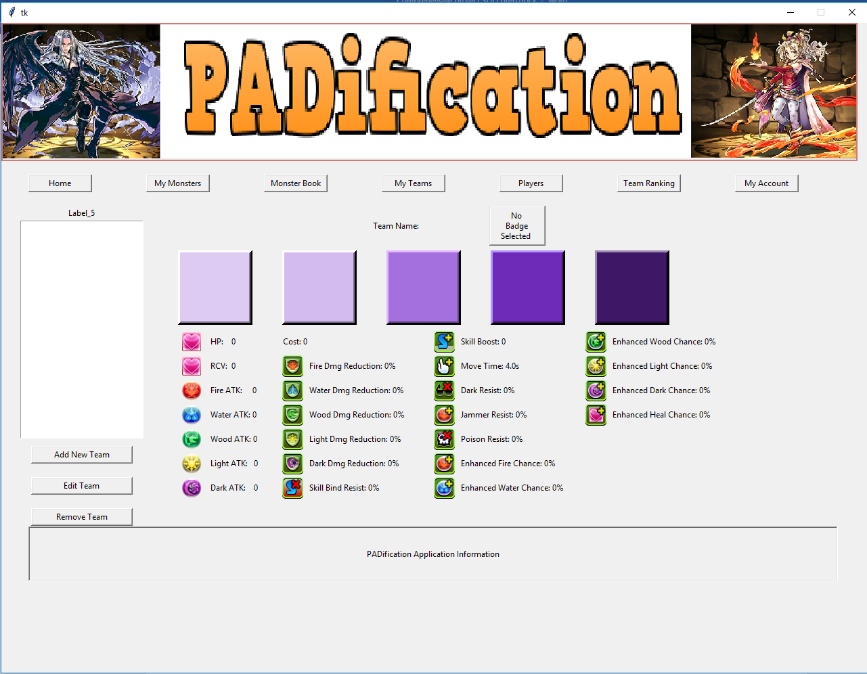


Figure 2.8‑7: PADification My teams screen

The My Teams screen allows users to set the monsters in their collections into teams. Users can view a team’s properties and the calculated stats of these properties based on the monster instances they’ve added to a given team.

**Cycle through teams**

Users have the ability to create numerous teams. Each team will be added to a list where users can select individual teams which will display their information. Selecting a team will also dictate the team which will be affected when users wish to Edit or delete a team.

**Create new team button**

This will send the user to a blank version of the edit team screen.

**Edit team button**

This will send the user to the edit team screen. The specific team will be determined by the team currently selected in the list.

**Delete team button**

This will remove the selected team from the user’s list permanently. When clicking the delete button a prompt will appear asking the user to confirm or deny the deletion of their team.

### Edit team screen



Figure 2.8‑8: PADification Edit team screen

The edit team screen allows users to mold teams, using the monsters from their collection, into any combination they wish.

**Team slots**

Users can select each of the 5 slots found on any team. When a slot is selected, users can then select a monster from their list. This will populate the slot with the monster selected and later the team’s properties. Users can also select the remove monster option to remove any instances in a slot. The same monster instance cannot be applied more than once in a single team.

**Team name**

Users are able to change a team’s name by inputting alphanumeric content in the Team name field.

**AwokenBadge button**

Users can choose to add an Awoken badge to their teams. Selecting this option will provide the user with a list of Awoken badges as well as an option to remove one if they so choose.

**Save Team button**

This button will save any changes done to a team. This button is also used to save a team instance when creating a new team.

**Delete team button**

This will remove the team from the user’s list permanently. When clicking the delete button a prompt will appear asking the user to confirm or deny the deletion of their team.

**Submit/retract team**

Users can submit the teams they create in order to have them available to the world and be given judgement. This judgement will declare the rank a team receives. This will help the community express how they feel about a given team. When a team has already been submitted, the button will become an unsub button. When pressed, the user will be prompted and informed that retracting a team will result in the loss of all ranks and the user will be asked to confirm or deny the retraction.

### Community screen

### Team Ranking screen

### Account information screen

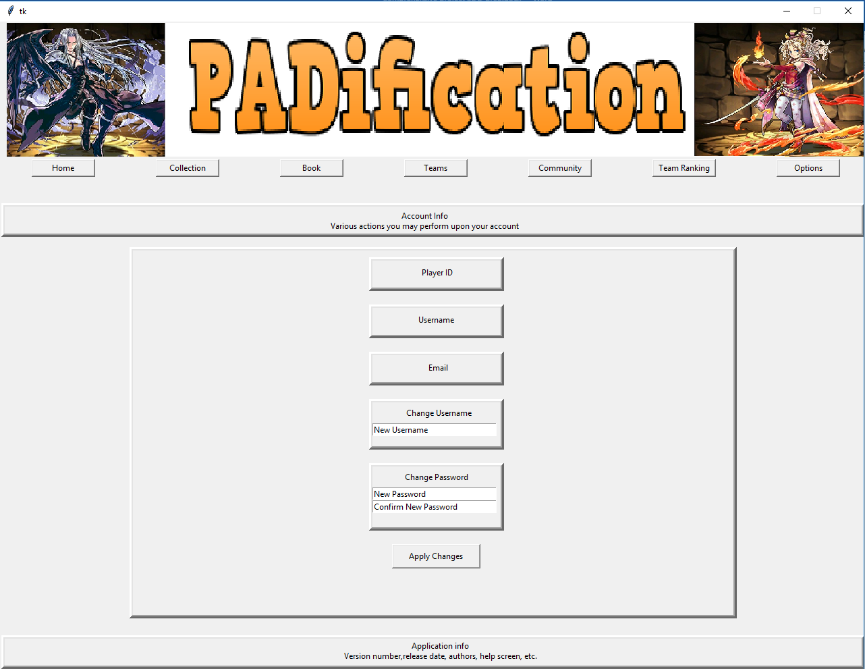


Figure 2.8‑9: PADification Account screen

The account information screen provides users with all the information about their account. These being:

* Player ID
* Email
* Password
* Username
* ProfileImage

Users are able to change their username, playerID, password and profile image here.

**Apply Changes button**

When users select this button, any changes that have been done to the modifiable fields will be saved. In the case of changing passwords, users will need to enter the new password and confirm the new password. If a user fails to match both password and confirm password fields, the changes will not be saved.

## Structure Viewpoint

In this section, we will identify and define each component of the PAD database in detail.

### MonsterAttribute table

Holds the 5 attributes assigned to each monster in Puzzles and Dragons

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| AttributeName | Identifies the name of the attribute as a string. | Varchar (50) - PK |

Table ‑: Monster attribute

* Table relationship:
  + This table is related to the MonsterClass table

### MonsterType table

Holds the 12 types assigned to each monster in Puzzles and Dragons

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| MonsterTypeName | Identifies the name of the types as a string. | Varchar (50) - PK |

Table ‑: Monster type

* Table relationship:
  + This table is related to the MonsterClass table

### ActiveSkill table

Holds all Active skills associated to monsters in Puzzles and Dragons

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| ActiveSkillName | Identifies the name of an Active skill as a string. | Nvarchar (100) - PK |
| ActiveSkillDesc | Describes the function of an Active skill using string content. | Nvarchar (MAX) – NOT NULL |
| ActiveSkillMaxLevel | Identifies the maximum numeric value in which an Active skill can be upgraded. | Int – NOT NULL |
| ActiveSkillMaxCoolDown | Identifies the initial numeric value needed before an Active skill can be used by a player. | Int – NOT NULL |

Table ‑: Active skill

* Table relationship:
  + This table is related to the MonsterClass table

### LeaderSkill table

Holds all Leader skills associated to monsters in Puzzles and Dragons

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| LeaderSkillName | Identifies the name of a Leader skill as a string. | Nvarchar (100) – PK |
| LeaderSKillDesc | Describes the function of a Leader skill using string content. | Nvarchar (MAX) – NOT NULL |

Table ‑: Leader skill

* Table relationship:
  + This table is related to the MonsterClass table

### AwokenSkill table

Holds all Awoken skills found in Puzzles and Dragons

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| AwokenSkillName | Identifies the name of an Awoken skill as a string. | Varchar (100) - PK |
| AwokenSkillDesc | Describes the function of an Awoken skill using string content. | Varchar (MAX) – NOT NULL |

Table ‑: Awoken skill

* Table relationship:
  + This table is related to the AwokenSkillList table

### AwokenSkillList table

Holds all Awoken Skills specific to each monster in Puzzles and Dragons

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| ASListID | Identifies a given Awoken skill list to its destined monster as an integer value. | Int - PK |
| AwokenSkillOne | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |
| AwokenSkillTwo | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |
| AwokenSkillThree | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |
| AwokenSkillFour | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |
| AwokenSkillFive | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |
| AwokenSkillSix | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |
| AwokenSkillSeven | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |
| AwokenSkillEight | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |
| AwokenSkillNine | Identifies the monsters’ Awoken skill using string content. | Varchar (100) – NULL |

Table ‑: Awoken skill list

* Table relationship:
  + This table is related to the MonsterClass table

### LatentSkill table

Holds all latent awoken skills found in Puzzles and Dragons

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| LatentSkillName | Identifies the name of a Latent skill as a string. | Varchar (50) - PK |
| LatentSkillDesc | Describes the function of a Latent skill using string content. | Varchar (MAX) – NOT NULL |
| LSSlotsReq | Identifies the numeric value required for a monster to equip a Latent skill. | Int – NOT NULL |

Table ‑: Latent awoken skill

* Table relationship:
  + This table is related to the LatentSkillList table

### LatentSkillList table

Determines each monsters’ availability for latent awoken skill slots

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| InstanceID | Identifies the monster when checking its slots as an integer value. | Int- PK |
| LatentSkillOne | Identifies if the slot is available/Identifies a monsters’ Latent skill as a string. | Varchar (50) – NULL |
| LatentSkillTwo | Identifies if the slot is available/Identifies a monsters’ Latent skill as a string. | Varchar (50) – NULL |
| LatentSkillThree | Identifies if the slot is available/Identifies a monsters’ Latent skill as a string. | Varchar (50) – NULL |
| LatentSkillFour | Identifies if the slot is available/Identifies a monsters’ Latent skill as a string. | Varchar (50) – NULL |
| LatentSkillFive | Identifies if the slot is available/Identifies a monsters’ Latent skill as a string. | Varchar (50) – NULL |
| LatentSkillSix | Identifies if the slot is available/Identifies a monsters’ Latent skill as a string. | Varchar (50) – NULL |
| ExtraSlot | Determines if a monster has access to a sixth Latent skill slot | Bit – NOT NULL – default: 0 |

Table ‑: Latent awoken skill list

* Table relationship:
  + This table is related to the MonsterClass table

### MonsterClass table

Holds records of every monster in Puzzles and Dragons

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| MonsterClassID | Identifies a monster via an integer value. | Int - PK |
| MonsterName | Identifies the monsters’ name as string content. | Nvarchar (100) – NOT NULL |
| Rarity | In-game representation of a monster’s value as an integer value. | Int – NOT NULL |
| PriAttribute | Identifies a monster’s primary attribute using a string. | Varchar (50) – NOT NULL |
| SecAttribute | Identifies a monster’s secondary attribute using a string. | Varchar (50) – NULL |
| MonsterTypeOne | Identifies one of the monster’s 3 possible type as a string. | Varchar (50) – NOT NULL |
| MonsterTypeTwo | Identifies one of the monster’s 3 possible type as a string. | Varchar (50) – NULL |
| MonsterTypeThree | Identifies one of the monster’s 3 possible type as a string. | Varchar (50) – NULL |
| ExpCurve | Identifies a monster’s maximum experience over the course of its growth. | Int – NOT NULL |
| MaxLevel | Identifies a monster’s highest point reachable in level as an integer value. | Int – NOT NULL |
| MonsterCost | Identifies a monster’s integer value used to determine how teams are assembled. | Int – NOT NULL |
| ASListID | Confirms whether or not a monster has an AwokenSkill list. | Int - NULL |
| LeaderSkillName | Identifies a monster’s Leader skill as a string. | Nvarchar (100) – NULL |
| ActiveSkillName | Identifies a monster’s Active skill as a string. | Nvarchar (100) – NULL |
| MaxHP | Identifies a monster’s highest point reachable in his health point stat as an integer value. | Int – NOT NULL |
| MinHP | Identifies a monster’s lowest point reachable in his health point stat as an integer value. | Int – NOT NULL |
| GrowthRateHP | Identifies how a monster’s health point stat will be influence as it continues to grow. | Real – NOT NULL |
| MaxATK | Identifies a monster’s highest point reachable in his attack stat as an integer value. | Int – NOT NULL |
| MinATK | Identifies a monster’s lowest point reachable in his attack stat as an integer value. | Int – NOT NULL |
| GrowthRateATK | Identifies how a monster’s attack stat will be influence as it continues to grow. | Real – NOT NULL |
| MaxRCV | Identifies a monster’s highest point reachable in his recovery stat as an integer value. | Int – NOT NULL |
| MinRCV | Identifies a monster’s lowest point reachable in his recovery stat as an integer value. | Int – NOT NULL |
| GrowthRateRCV | Identifies how a monster’s recovery stat will be influence as it continues to grow. | Real – NOT NULL |
| CurSell | Identifies the calculated Coin value a monster has. | Int – NOT NULL |
| CurFodder | Identifies the calculated Exp value a monster would present when used as Evo-material. | Int – NOT NULL |
| MonsterPointValue | Identifies the monster points received when selling a given monster as an integer value. | Int – NOT NULL |

Table ‑: Monster class

* Table relationship:
  + This table is related to the MonsterAttribute table, MonsterType table, AwokenSkillList table, ActiveSkill table, LeaderSkill table, Curve table, Player table and EvolutionTree table.

### EvolutionTree table

Holds all records for each monster’s required material in order to change into different monsters

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| NextMonsterID | Identification number for a monster after the evolution. | Int - PK |
| BaseMonsterID | Identification number for a monster before the evolution. | Int – NOT NULL |
| EvoMaterialIDOne | Identification number for a monster required to evolve another monster. | Int – NOT NULL |
| EvoMaterialIDTwo | Identification number for a monster required to evolve another monster. | Int - NULL |
| EvoMaterialIDThree | Identification number for a monster required to evolve another monster. | Int - NULL |
| EvoMaterialIDFour | Identification number for a monster required to evolve another monster. | Int - NULL |
| EvoMaterialIDFive | Identification number for a monster required to evolve another monster. | Int - NULL |
| Ultimate | Identifies the Evolution as Ultimate Evolution meaning a monster will be able to Devolve as well. | Bit – NOT NULL |

Table ‑: Evolution tree

* Table relationship:
  + This table is related to the MonsterClass table.

### Player table

Holds player information.

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| PlayerID | Identifies a player as an integer value. | Int – NOT NULL |
| Password | Validation expression for a player’s account. | Varchar (10) - NOT NULL |
| Email | Player email used to help uniquely identify the account. | Varchar (50) – PK NOT NULL |
| Username | Uniquely identifies the player’s account | Varchar (15) – NOT NULL |
| ProfileImage | Allows a monster image to be displayed as a profile picture. | INT - NULL |

Table ‑: Player

* Table relationship:
  + This table is related to the MonsterAttribute table, MonsterInstance table, Team table, MonsterClass table and Follower table.

### Follower table

Holds other people’s information for the player to access.

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| FID | Uniquely identifies the Instance of the Follower relationship. | Int – PK  Auto increment – 1  Default – 0 |
| Email | Used to identify the player as part of the relationship the Player & Follower Table. | Varchar (15) – NOT NULL |
| FollowerEmail | Identifies the Username of other players. | Varchar (15) –NOT NULL |

Table ‑: Follower

* Table relationship:
  + This table is related to the MonsterAttribute table, MonsterInstance table and Team table.

### MonsterInstance table

Holds the records of unique properties of monsters.

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| InstanceID | Identifies a specific monster entered into the database. | Int – PK  ID seed – 100,000,000  Auto increment - 1 |
| Email | Identifies the player related to his monsters in the database. | Varchar (50) – NOT NULL |
| MonsterClassID | Identifies the monster instance to an existing monster in PAD. | Int – NOT NULL |
| CurrentExperience | Identifies a monster’s current integer value representing its experience points. | Int- NOT NULL |
| PlusATK | Identifies a monster’s additional integer value added to the attack stat. | Int- NOT NULL |
| PlusRCV | Identifies a monster’s additional integer value added to the recovery stat. | Int- NOT NULL |
| PlusHP | Identifies a monster’s additional integer value added to the health point stat. | Int- NOT NULL |
| SkillsAwoke | Identifies the monster’s current active Awoken skills. | Int- NOT NULL |
| AssistMonsterID | Identifies a monster’s assist representative monster by integer value. | Int – NULL |
| SkillLevel | Identifies a monster’s current Active skill integer value. | Int – NULL |
| LSListID | Identifies a monster’s current Latent skill equip. | Int – NULL |
| Favorites | Identifies a monster’s current label as Favorited or no. | Bit – NOT NULL  Default – 0 |
| WishList | Identifies a monster’s current label as Wish Listed or no. | Bit – NOT NULL  Default – 0 |

Table ‑: Monster instance

* Table relationship:
  + This table is related to the Player table, MonsterClass table, LatentSkillList table and MonsterInstance table.

### Team table

Contains the fields needed to simulate the team building in Puzzles and Dragons.

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| TeamInstanceID | Unique integer value used to identify a built team. | Int – PK  ID seed – 100,000  Auto increment - 1 |
| Email | Identifies the player related to heir teams in the database. | Varchar (50) – NOT NULL |
| TeamName | Identifies the team as a string. | Varchar (20) - NULL |
| LeaderMonster | Identifies a monster set in a team’s leader slot. | Int- NULL |
| SubMonsterOne | Identifies a monster set in a team’s sub slot. | Int- NULL |
| SubMonsterTwo | Identifies a monster set in a team’s sub slot. | Int- NULL |
| SubMonsterThree | Identifies a monster set in a team’s sub slot. | Int- NULL |
| SubMonsterFour | Identifies a monster set in a team’s sub slot. | Int- NULL |
| BadgeName | Identifies a Team’s selected Badge as a string. | Varchar (50) – NULL |

Table ‑: Team

* Table relationship:
  + This table is related to the Player table and Badge table.

### AwokenBadge table

Holds all Awoken Badges found in Puzzles and Dragons.

|  |  |  |
| --- | --- | --- |
| Field Name: | Description: | Properties: |
| AwokenBadgeName | Identifies the badge as a sting. | Varchar (50) - PK |
| AwokenBadgeDesc | Description of the badges’ effects. | Varchar (MAX) – NOT NULL |

Table ‑: Awoken badge

* Table relationship:
  + This table is related to the Team table.

## Interactions Viewpoint

## State Dynamics Viewpoint

## Algorithm Viewpoint

## Resources

# Appendices

## Tool Configuration

### Configure MSSQL DB for Remote Connections.

#### Purpose

Allowing MSSQL 2014 servers to Accept TCP/IP connections and SQL authentication.

#### Prerequisites

* MSSQL 2014 Installed
* Microsoft SQL Server Management Tool

#### Enable TCP/IP Connection

To enable the TCP/IP protocol in SQL Server 2014, follow these steps:

1. Open **SQL Server Configuration Manager** from:

for 64 bits: "**C:\Windows\SysWOW64\SQLServerManager12.msc**"

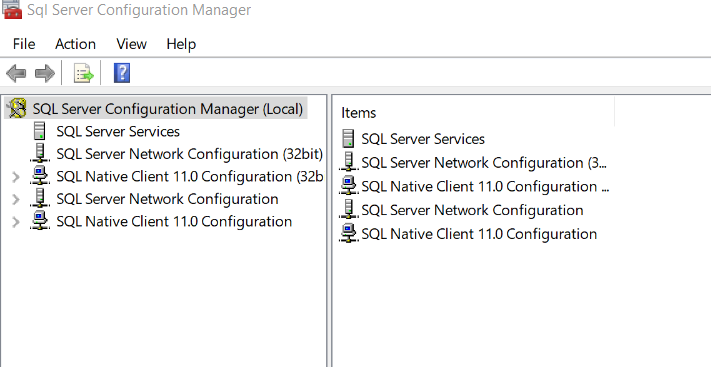
for 32 bits: "**C:\Windows\system32\SQLServerManager12.msc**"

Figure ‑: Open SQL server Configuration

1. Expand **SQL Server Network Configuration** and click on **Protocols for MSSQLSERVER**

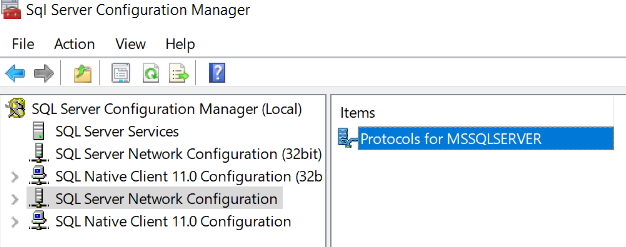


Figure ‑: SQL server configuration; Protocols for MSSQLSERVER

1. Right click on **TCP/IP** and choose **Enable**

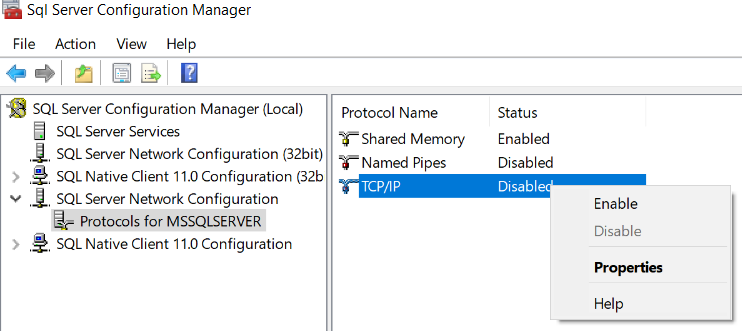


Figure ‑: SQL server Configuration Enable TCP/TP

1. Click **OK** on the Warning that the service will have to be restarted
2. Click on **SQL Server Services**

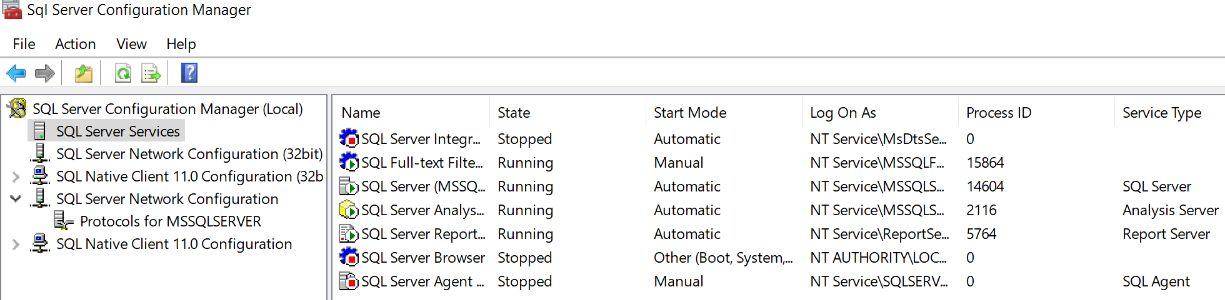


Figure ‑: SQL server Configuration Server services

1. Right click on **SQL Server (MSSQLSERVER)** and choose **Restart**

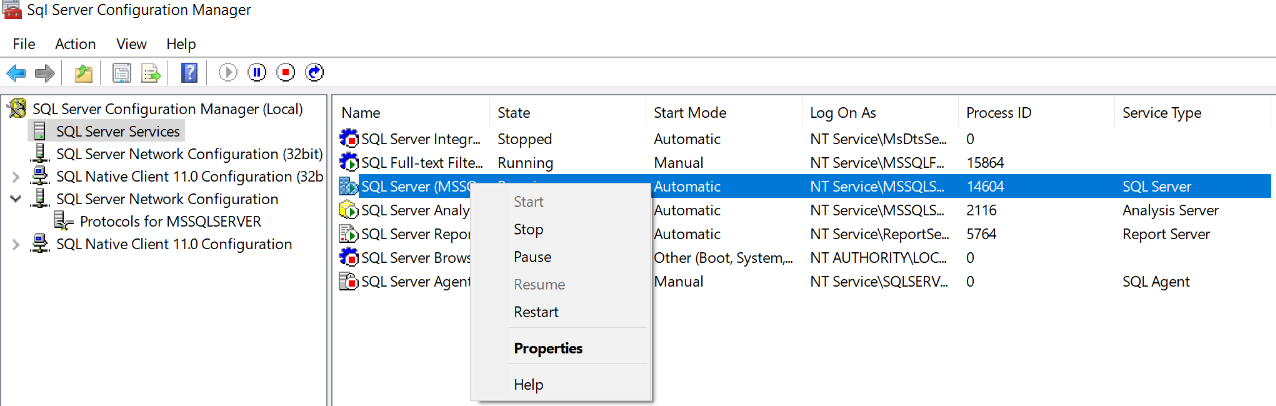


Figure ‑: SQL server Configuration restart MSSQLSERVER

#### Setup User for SQL TCP/IP Connection

To create a username to access the SQL Server Via TCP/IP:

1. Open **SQL Server Management Studio** and **Login**.
2. Expand **Security** in the **Object Explorer**.
3. Right Click **Logins** and Select **New Login**.

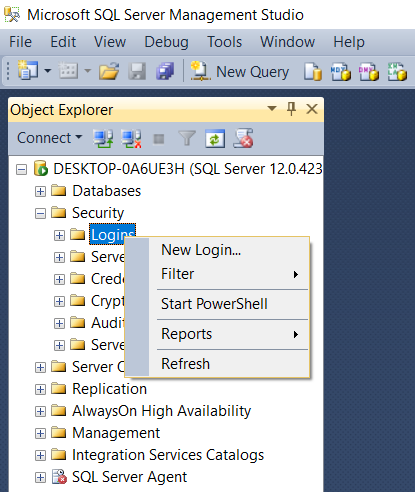


Figure ‑: MSSQL SMS New log in

1. In the Login Creation Screen first select **SQL Server Authentication**.

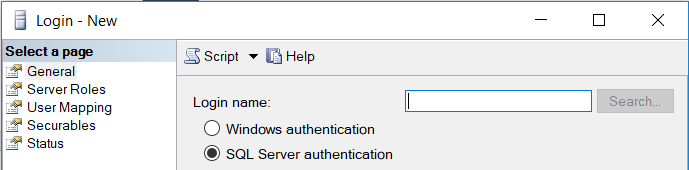


Figure ‑: MSSQL SMS New log in SQL Server Authentication

1. Enter the New **Login Name** and **Password / Confirm Password**.
2. Uncheck **Enforce Password Policy**.

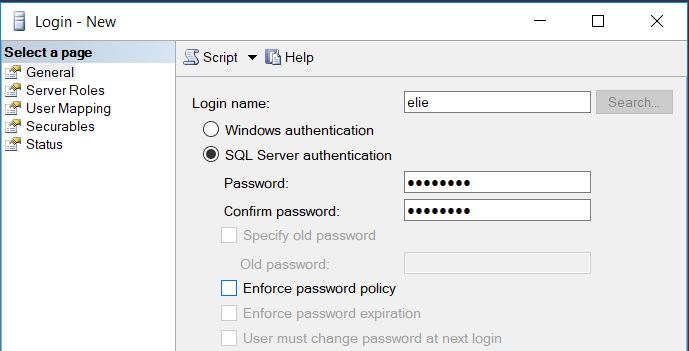


Figure ‑: MSSQL SMS New log in name/password/Enforce Password Policy

1. Select **Server Roles** at the top left.

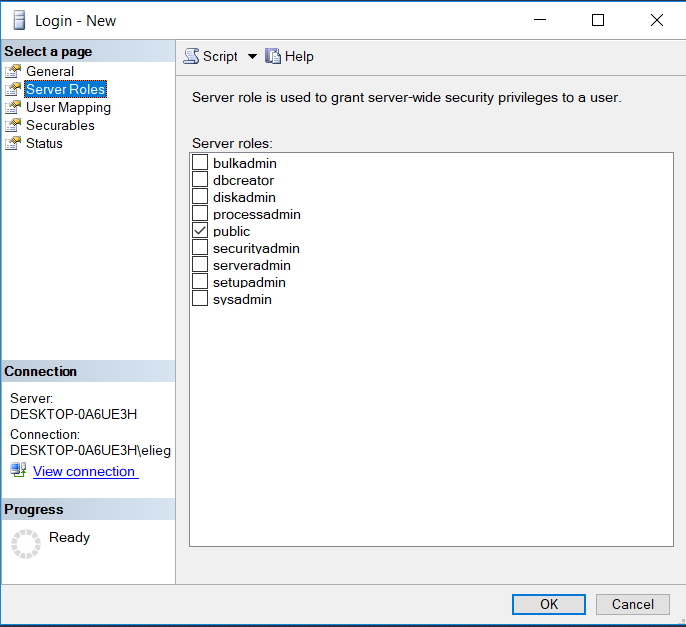


Figure ‑: MSSQL SMS New log in Server Roles

1. Select the role **SysAdmin**. "If more constraints are needed, creating custom Roles is possible."
2. Click **OK**

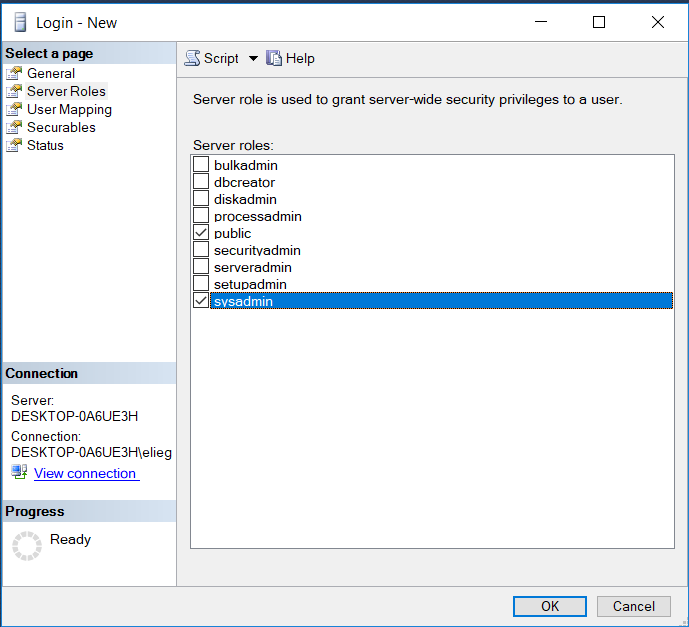


Figure ‑: MSSQL SMS New log in Server Roles SysAdmin

1. Right Click on **server** in **Object** **Explorer** \*first item\* and Select **Properties**

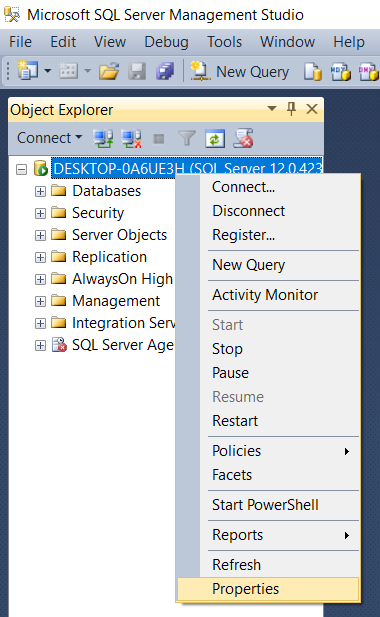


Figure ‑: MSSQL SMS Server Properties

1. Select **Security** on the left

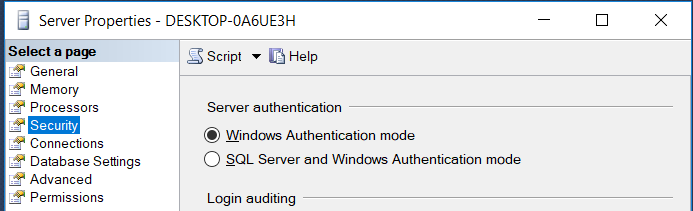


Figure ‑: MSSQL SMS Server properties Security

1. Select **SQL and Windows Authentication Mode** and Press **ok**

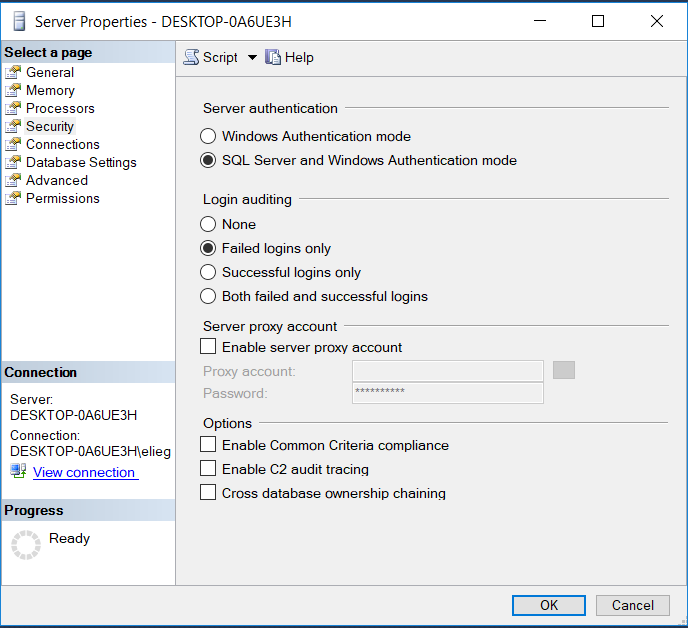


Figure ‑: MSSQL SMS Server Properties SQL Server and Windows Authentication mode

1. Restart the SQL server.

#### Setup Firewall Exceptions:

To allow connection through the Windows Firewall

1. Open **Windows Firewall**

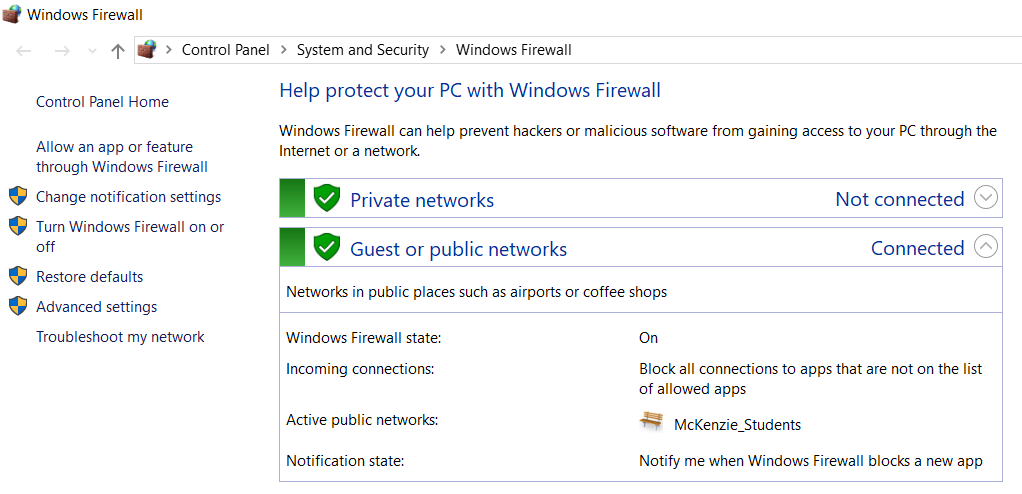


Figure ‑: Windows Firewall

1. Select **Advanced settings** on the left.

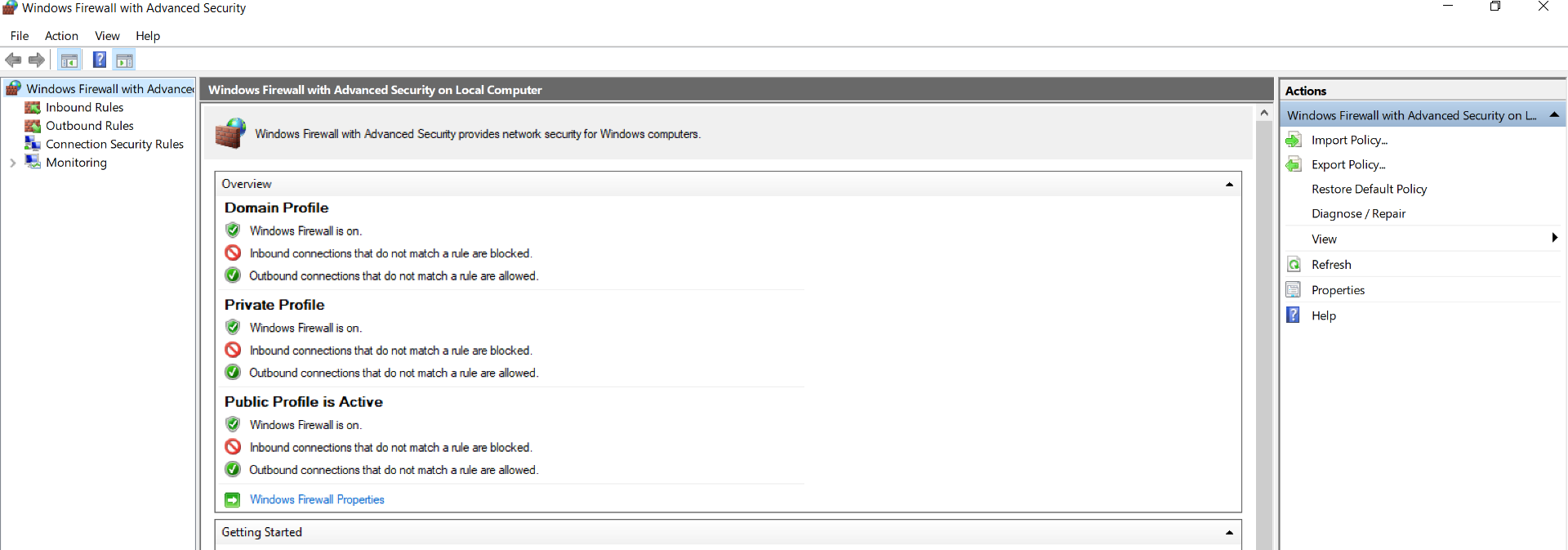


Figure ‑: Windows Firewall Advanced settings

1. Select **Inbound Rules** on the Left.

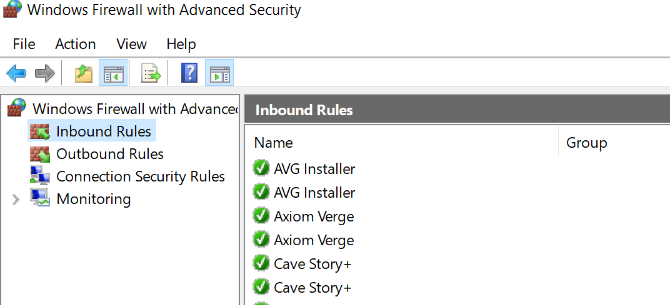


Figure ‑: Windows Firewall Inbound Rules

1. Select **New Rule** on the Right.

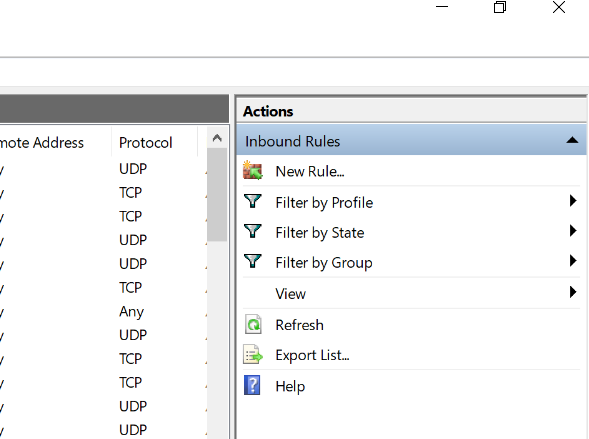


Figure ‑: Windows Firewall Inbound Rules/New Rules

1. Select **Port** and hit **Next >**

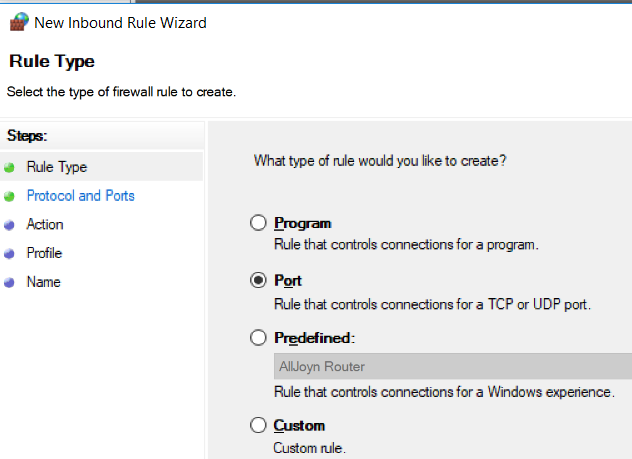


Figure ‑: Windows Firewall Inbound Rules/New Rules/Port

1. Set **Specific Local Ports** to **1433** and hit Select **Next >**

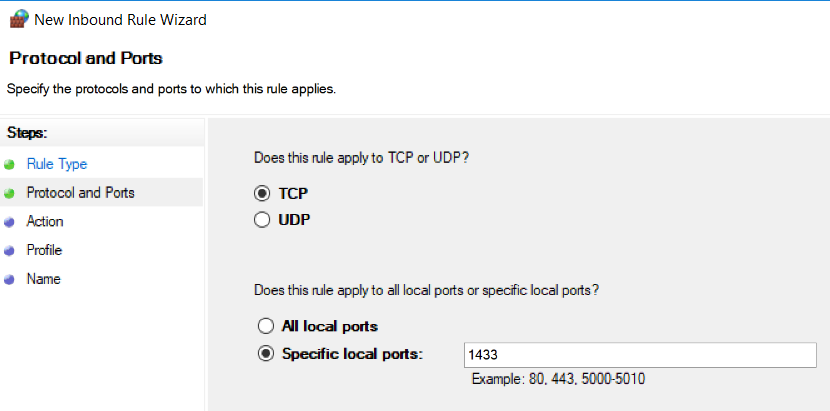


Figure ‑: Windows Firewall Inbound Rules/New Rules/Specified local port

1. Select **Allow the connection** and hit **Next >**

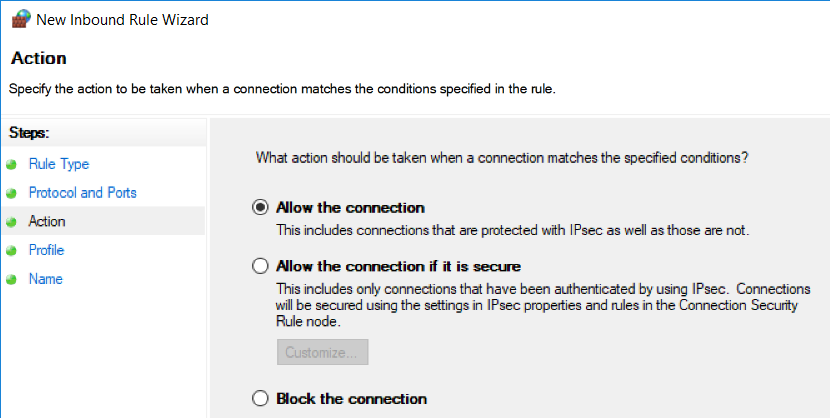


Figure ‑: Windows Firewall Inbound Rules/New Rules/Allow connection

1. Select all check boxes and hit **Next >**

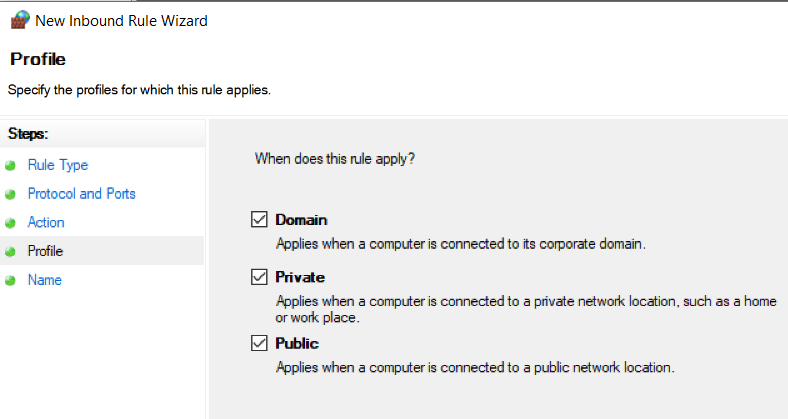


Figure ‑: Windows Firewall Inbound Rules/New Rules/Profile

1. Name **SQL IN** and hit **Finish**

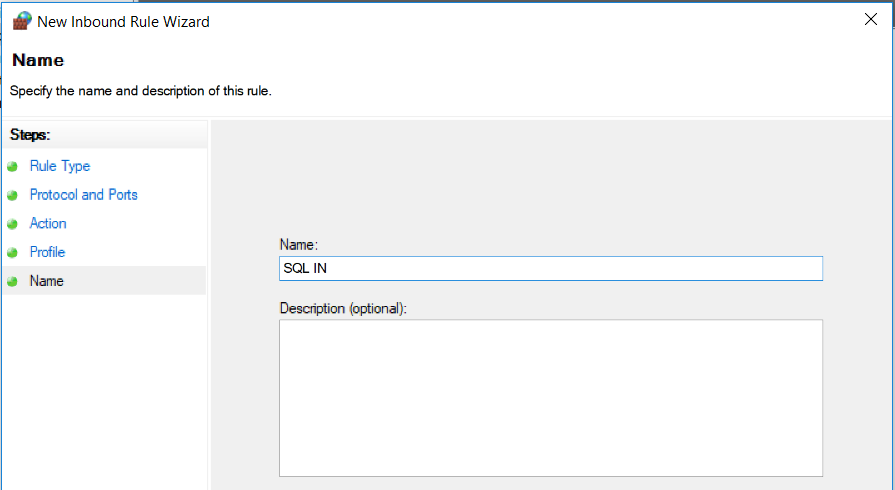


Figure ‑: Windows Firewall Inbound Rules/New Rules/Name

1. Select **Outbound Rules** on the Left.

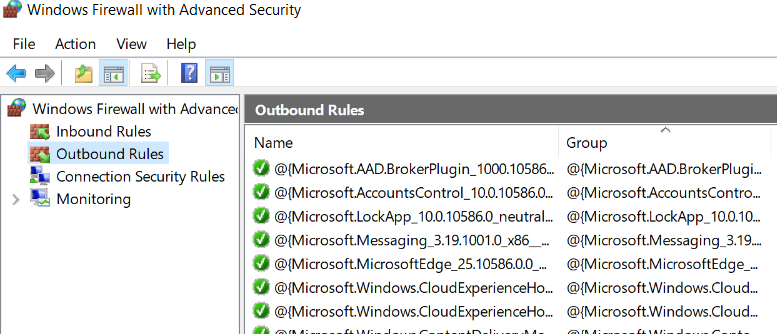


Figure ‑: Windows Firewall Outbound Rules

1. Select **New Rule** on the Right.

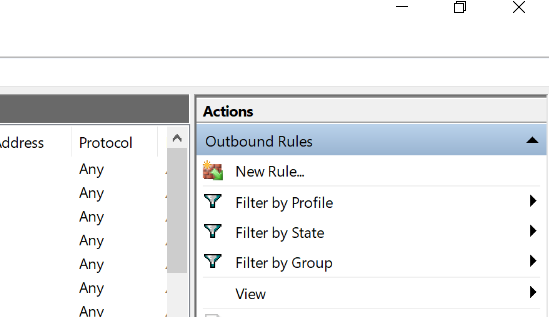


Figure ‑: Windows Firewall Outbound Rules/New Rules

1. Select **Port** and hit **Next >**

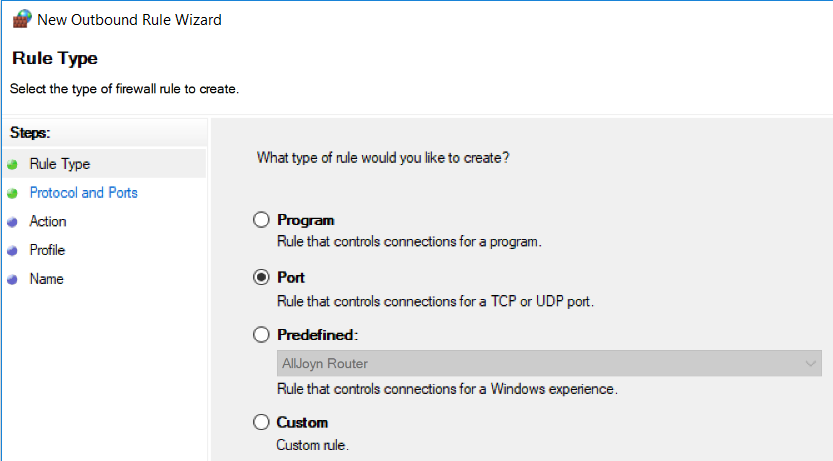


Figure ‑: Windows Firewall Outbound Rules/New Rules/Port

1. Set **Specific Remote Ports** to **1433** and hit Select **Next >**

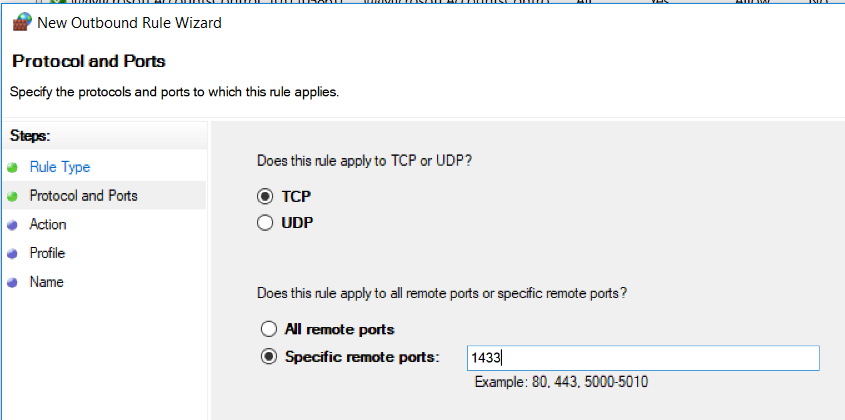


Figure ‑: Windows Firewall Outbound Rules/New Rules/Specific remote ports

1. Select **Allow the connection** and hit **Next >**

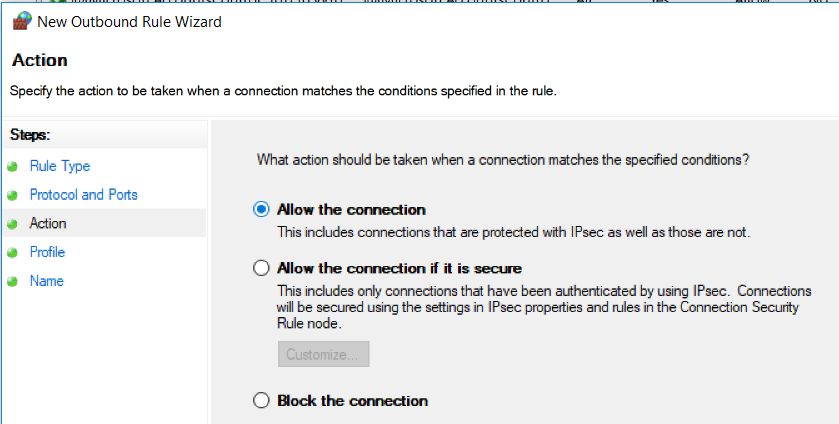


Figure ‑: Windows Firewall Outbound Rules/New Rules/Allow connection

1. Select all check boxes and hit **Next >**

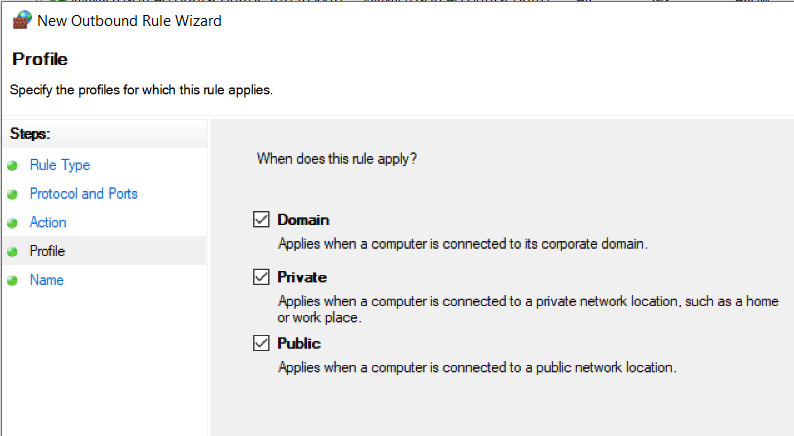


Figure ‑: Windows Firewall Outbound Rules/New Rules/Profile

1. Name **SQL OUT** and hit **Finish**



Figure ‑: Windows Firewall Outbound Rules/New Rules/Name

### Create the Database from .bat files

#### Prerequisites:

* Microsoft SQL server management studios
* Install Microsoft ODBC 13.1
* Install SQLCMD utilities

#### Install Microsoft ODBC 13.1

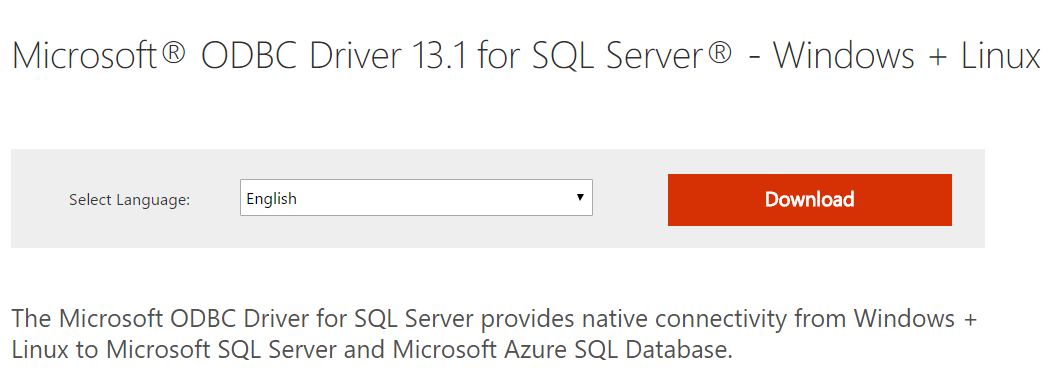
1. Search for SQLCMD Utilities 13.1 or use the link below to reach the website and hit **Download.** <https://www.microsoft.com/en-us/download/details.aspx?id=53339> 

Figure ‑: Download Screen

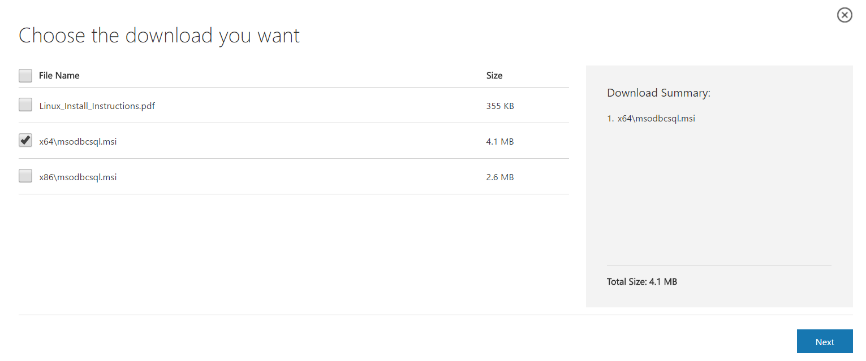
1. Check the 2nd box titled: **x64\msodbcsql.msi** and hit Next. 

Figure ‑: ODBC Select Download

1. Proceed to install the application.
   * **Warning:** your PC will have to restart after the changes have been made.



Figure ‑: ODBC 1st Screen of Install



Figure ‑: OBDC License Agreement

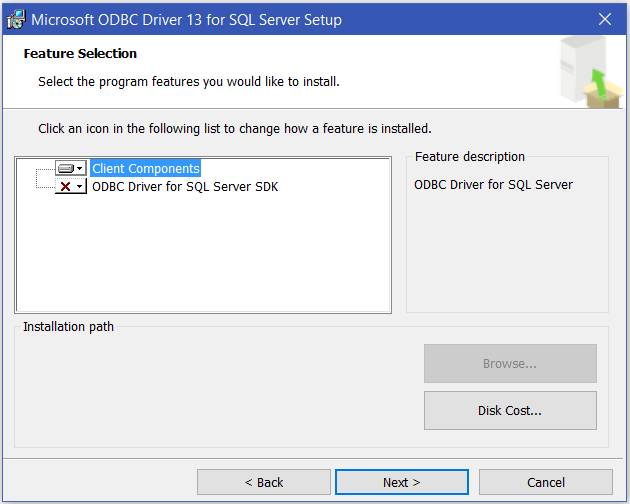


Figure ‑: ODBC Feature Selection

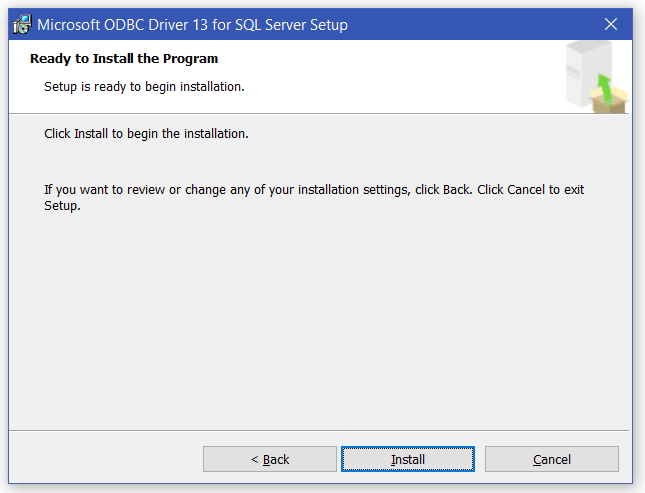


Figure ‑: ODBC Begin Installation

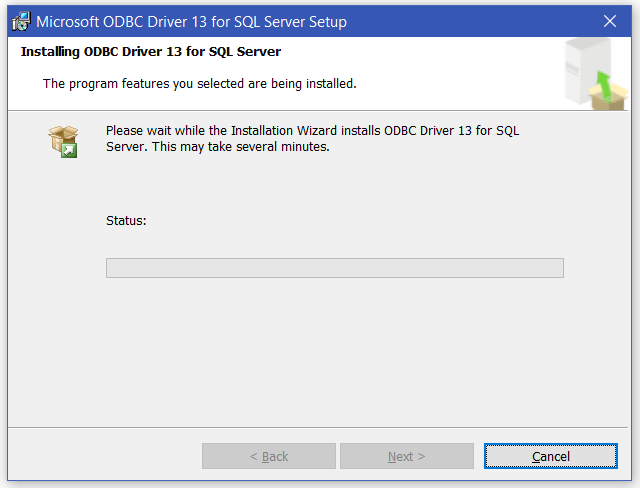


Figure ‑: ODBC Install Screen

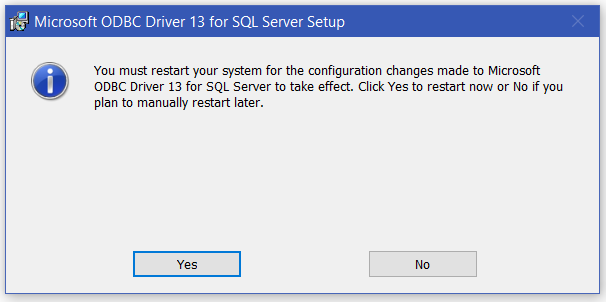


Figure ‑: Notification of Restart

#### Install SQLCMD utilities

1. Search for SQLCMD Utilities 13.1 or use the link below to reach the website and hit **Download.** <https://www.microsoft.com/en-us/download/details.aspx?id=53591>

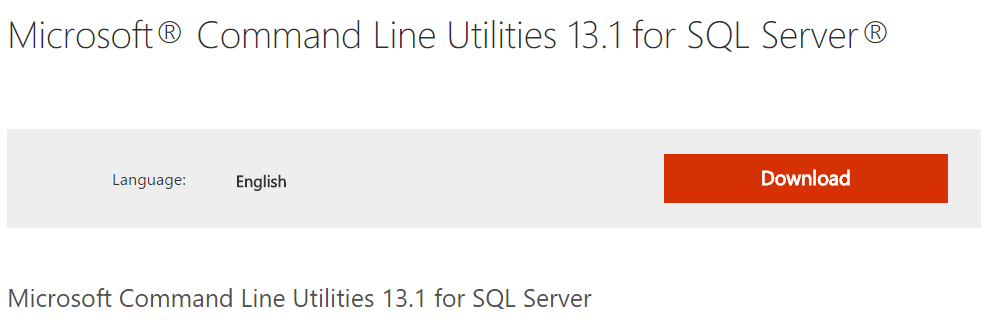


Figure ‑: SQLCMD Download Page

1. Check the 2nd box titled: **Command Line Utilities MSI files\x86\MsSqlCmdLnUtils.msi** and hit Next.

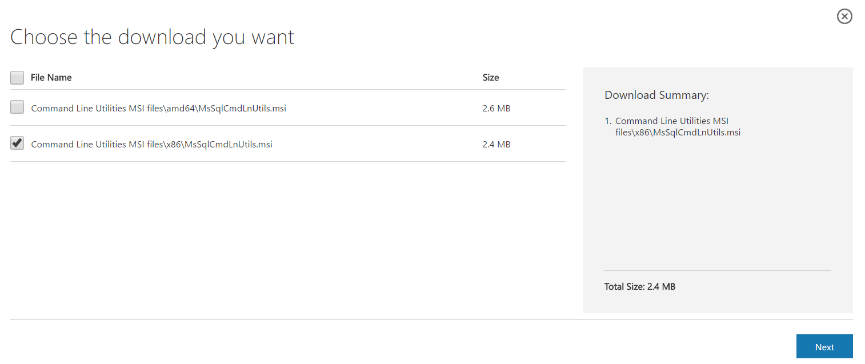


Figure ‑: SQLCMD Choose Download

1. Proceed to install the application.

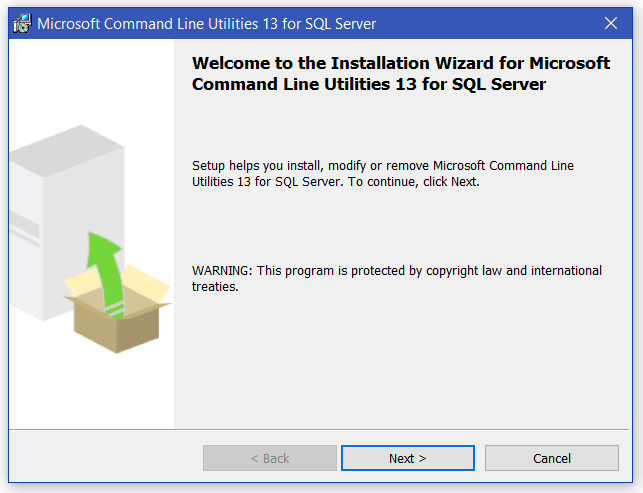


Figure ‑: SQLCMD First Page of Installation



Figure ‑: SQLCMD Accept License Terms

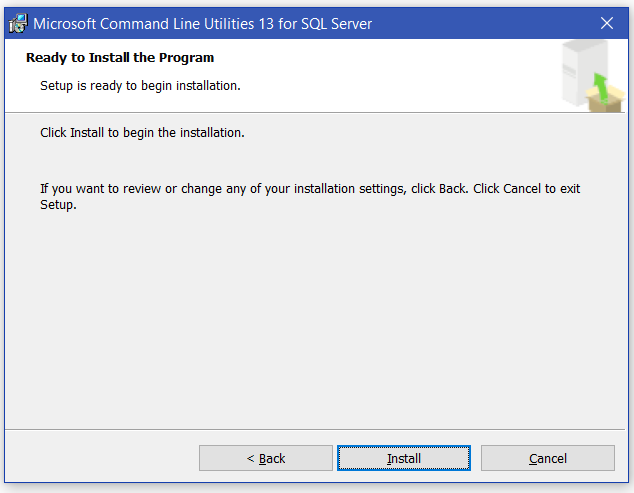


Figure ‑: SQLCMD Confirm Installation

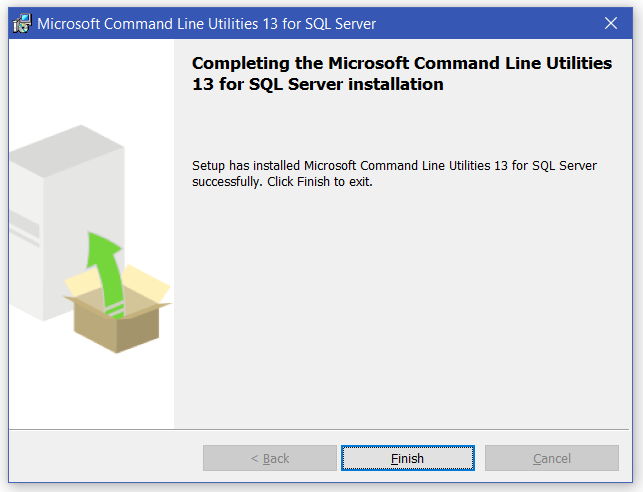


Figure ‑: SQLCMD Finish Installation

#### Create The database

1. Find the **Create\_DB.bat** file on your PC
2. Ensure the following files are located within the same directory as **Create\_DB.bat**
   * Generate-PADification-DB.sql
   * Table-Create.sql
   * LatentSkill-Inserts.sql
   * AttributeAndType-Inserts.sql
   * ActiveSkill-Inserts.sql
   * LeaderSkill-Inserts.sql
   * AwokenSkill-Inserts.sql
   * AwokenSkillList-Inserts.sql
   * MonsterClass-Inserts.sql
   * EvolutionTree-Inserts.sql
   * AwokenBadges-Inserts.sql
3. **Double click** the file and the process will begin automatically.
4. **Press the Return key** when prompted to continue the process.

#### Delete The database

**NOTE**: In order to delete the database, you must ensure the following

* + Close any .sql files associated to the PADification database
  + Refresh the database

1. Find the **Drop\_DB.bat** file on your PC
2. Ensure the following file is located within the same directory as **Drop\_DB.bat**
   * Drop-PADification-DB.sql
3. **Double Click** the file and the process will begin automatically
4. **Press the Return Key** when prompted to continue the process.