

1. Problem

There is a certain group of people who like watching Japanese animation (anime) and read Japanese comics (manga). Every user of this group has completed at least 100 titles (shows, movies, etc) and average user has about 200-300 titles completed. To avoid the unnecessary repetition of already watched titles (because it's hard to remember every title from 300-title list) users write their own anime and manga lists where they put all shows they watched and all manga they read. **The main problem is to create a database which could store data about users with anime and manga lists and basic application to interact with this database.**

Project name (for further usage in presentations, etc): “AniMaIndex”

List of objectives

Database

- Create an entity or group of entities to describe users.
- Users should have an ability to create separate lists of their favorite shows, books and artists.
- Create an entity or group of entities to describe anime and manga.
- Create entities which work with anime and manga-related entities for anime and manga lists.

Application

- All entities listed above can be viewed in this application.
- Provide instruments for simple CREATE and DELETE operations.
- Provide a simple search form.
- Application should use multi-user approach.

DBMS and platform

This database will be implemented for **MS Windows platform**, using **MS SQL Server as DBMS** and **C# as a programming language and LINQ** for implementation of user interface. The main reason behind this choice is that because this database is designed to work with operational systems made by Microsoft Corporation, it will be more convenient to use products of the same corporation for avoiding unnecessary problems during the project implementation. Another point is that this project won't be used for commercial purposes, so some of the license issues, which may force the developers to use opensource or freeware solutions, may be omitted.

2. Database description.

The whole conception of this database can be divided into 4 parts: **User Information, Anime, Manga and Common Information**. Also some of the tables may be related to more than one part.

User Information

This is group of entities which describes every user and it can be used for retrieving information about anime or manga list related to one particular user. It contains following tables:

1. **Users.** This table describes the user. It contains following attributes:
 - UserID (primary key). A unique id for every user.
 - NickName. Attribute with name or nickname of user.
 - Birthdate. Birthday date of user.
 - Gender. Male/ female/ unknown (in case if user doesn't want to write about this)
2. **FavStaff.** This table describes a user's list of favorite artists. It's also related to Staff table. It contains following attributes:
 - FavStaffID (primary key). A unique ID for every field.
 - UserID (foreign key). Taken from Users table.
 - StaffID (foreign key). Taken from Staff table.
3. **FavAnime.** This table describes a user's list of favorite anime shows. It's also related to Anime table. It's possible for user to add unwatched shows to favorites list.
 - FavAnimeID (primary key). A unique ID for every field.
 - TitleID (foreign key). Taken from Anime table.
 - UserID (foreign key). Taken from Users table.
4. **FavManga.** This table describes a user's list of favorite manga shows. It's also related to Manga table. It's possible for user to add unread manga to favorites list.
 - FavMangaID (primary key). A unique ID for every field.
 - MangaID (foreign key). Taken from Manga table.
 - UserID (foreign key). Taken from Users table.

Anime

This is a group of entities which describe anime and user's animelist. It contains following tables:

1. **Anime.** A table that contains all information about every anime title. It's quite common to use manga as a source for making anime show, therefore a connection to Manga table should be included:
 - TitleID (primary key). A unique field for every title.
 - AniYear.
 - MangaID (foreign key). Taken from Manga table.
 - StudioID (foreign key). Taken from Studio table.
 - TitleName. Attribute that describes title's name.
 - SerialType. OVA, ONA, Movie, Serial, etc.
 - GenreID (foreign key). Taken from Genres table.
 - ShortDescription. Short description of title's plot.
 - Episodes. How many episodes were aired?
 - AiredID. Is this show already aired?

2. **AnimeList.** A table which contains user's animelist.
 - ListID (primary key). A unique field for every list.
 - UserID (foreign key). Taken from Users table.
 - TitleID (foreign key). Taken from Anime table.
 - Score. Users score for one show.
 - StatusID (foreign key). Taken from UserStatus table.
 - EpsWatched. Episodes watched by user.

3. **Studio.** Table about studios which made anime. One anime show usually is made by one particular studio.
 - StudioID (primary key). A unique ID for every studio.
 - StudioName. Studio name.
 - StudioPage. Studio's webpage link.

Manga

This is a group of entities that describes info about manga.

1. **Manga.** Table with all info about manga.
 - MangaID (primary key). A unique field for every manga.
 - PublisherID (foreign key). Taken from Publisher table.
 - MangaName. Manga name.
 - ShortDescription. Short Description.
 - Chapters. Chapters published.
 - Books. Books published.
 - PublishedYet. Was this manga published?
 - GenreID (foreign key). Taken from Genres table.

2. **MangaList.** Table with user's mangalist.
 - MangaListID (primary key). A unique field for every mangalist.
 - UserID (foreign key). Taken from Users table.
 - MangaID (foreign key). Taken from Manga table.
 - Score. User's score for one manga.
 - StatusID (foreign key). User's status on this manga.
 - ChaptersRead. Chapters read.
 - BooksRead. Books read.

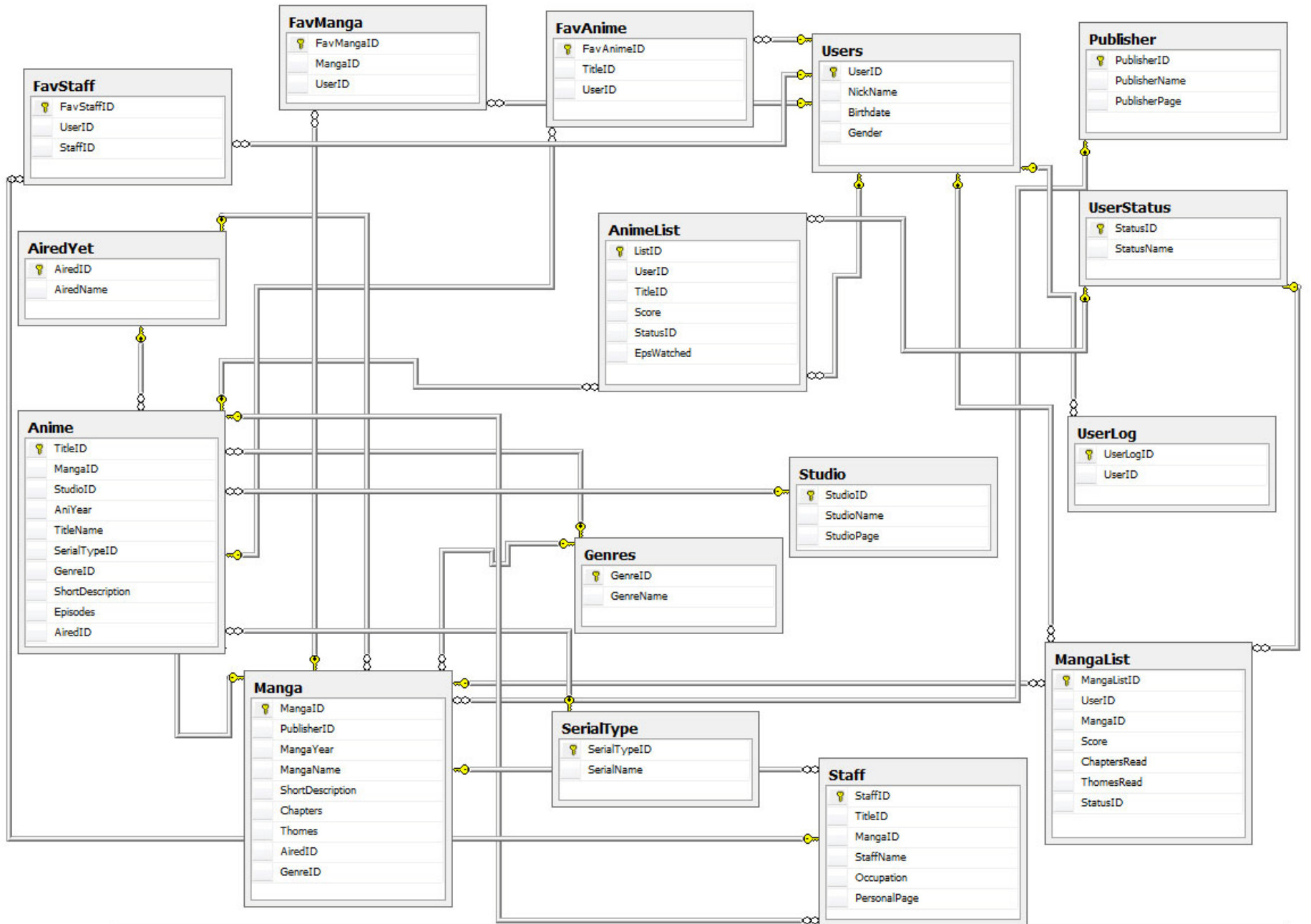
3. **Publisher.** Table about manga publishers.
 - PublisherID (primary key). A unique ID for ever publisher.
 - PublisherName. Publisher's name.
 - PublisherPage. A link to publisher's webpage.

Common Information.

This is group of entities used for both anime and manga parts.

1. **UserStatus.** User status for one title/ manga.
 - StatusID (primary key). A unique ID for every status.
 - StatusName. “Watching”, “Reading”, “Plan to watch”, “Dropped”, etc.
2. **Genres.** Info about genres.
 - GenreID (primary key). A unique ID for every genre.
 - GenreName. “Drama”, “Comedy”, “Shoujo-ai”, “Shounen-ai”, etc.
3. **Staff.** Info about anime and manga artists.
 - StaffID (primary key). A unique ID for every staff member.
 - TitleID (foreign key). Taken from Anime table.
 - MangaID (foreign key). Taken from Manga table.
 - StaffName. Person’s name.
 - Occupation. Voice actor, designer, director, etc.
 - PersonalPage. Person’s personal webpage.

Tables and their content may vary. Please check the updated database diagram on the next page before changing any tables.



User Guide

This is a short user guide which will cover all basic information you may need for using this application.

Contents

1. User and Admin. Main differences and description.
2. Getting started.
3. Possible problems.

1. As you may have already noticed, this application has two main “modes”. This is made to distinguish priorities between different types of users.
“Admin” users are those who have access to all kinds of main database content which can’t be generated by ordinary users. “Admin” can create new entries about anime, manga etc and manage user accounts. This type of user “sees” database content as it is, without replacing ID fields with appropriate names.
“User”, however, doesn’t need any of the features which “admin” mode can offer. The main purpose of “users” is to create and manage their own lists of shows, favorites etc. This type of user “sees” database content with replacing ID fields with appropriate names. All inner mechanisms are hidden.

In this application you can use both modes as you wish.

2. Getting started.
All ADD and DELETE forms are based on the same pattern. Thus there’s no need to show a guide for every form.

Adding new anime title to database (screenshot 1)

1. Go to **Admin > Anime**

2. Press **Load** button to load the current list of titles
3. Fill in fields below, please don’t leave the field **Year** blank or fill it with symbols
4. Press **Add New Anime** button
5. In case of success you will get a success message
6. Press **Load** button again. Your added title should appear.
7. To remove simply highlight the appropriate fields and press **Remove!** button. You will get an error message if something bad will occur

Adding new anime title to your list (screenshot 2)

1. Go to **User**, input your username which can be added via **Admin** mode
2. Go to **Lists: Anime**.
3. The process of adding to animelist is almost as same as adding new title in database. Press **Load** to load your current list.
4. Choose genre and press **Search**.
5. Highlight titles you want to add in the lower table and press **Add**
6. In case of success you’ll get a success message. Press **Load** to load your list again.
7. For removing see 7 from previous list.

3. Possible Problems

Most of the problems may come due to unavailability of cascade UPDATE and DELETE operations in LINQ. Thus, please be careful when you remove any information from database.

Other errors may include incorrect input format and undiscovered problems about which we’d like you to tell us about.

Screenshots (see next page)

AniMaIndex v 0.5

	TitleID	MangaID	StudioID	AniYear	TitleName	SerialTypeID	GenreID	ShortDe
▶	1	1	1	2011	To aru majitsu no...	1	1	The adv
	2	1	1	1982	Name	1	1	Short de

Load!

Remove!

RM

testomedy

13

2001|

Tesuto

OVA

TestStud

Yes - Ongoing

Test Test

Add New Anime

>Admin

AniMaIndex v 0.5

	title_name	year	serial_type	genre	description	episodes	manga_name	studio_name
▶	To aru majitsu no ...	2011	OVA	testomedy	The adventures ...	3	Tesuto	TestStud

Load!Remove!

testomedySearch

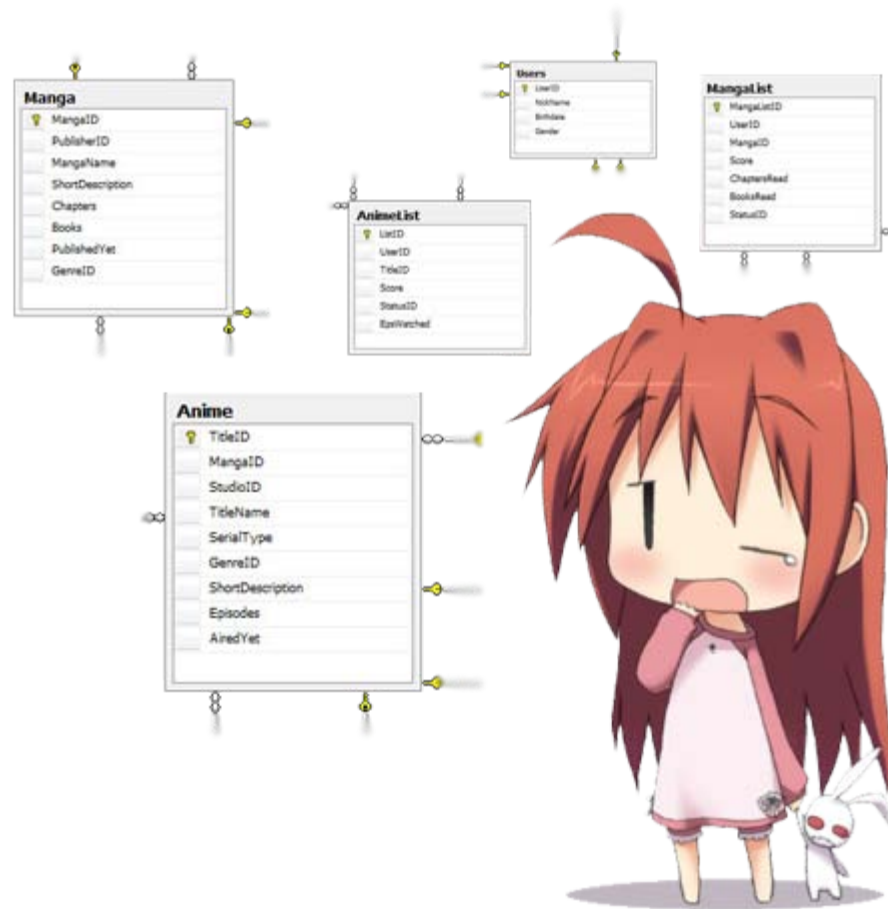
	manga_name	studio_name	year	title_name	serial_type	genre	description	episodes
	Tesuto	TestStud	2011	To aru majitsu no ...	OVA	testomedy	The adventures ...	3
	Tesuto	TestStud	1982	Name	OVA	testomedy	Short description	15
▶	Tesuto	TestStud	2001	RM	OVA	testomedy	Test Test	13

Add!

8
Watching
3

>User

You're logged as: TestUsr



Thanks for reading this guide to the last page and trying our application. We hope you liked it.