

Manhattan School of Driving LLC, Management System

Date Here

By:
Author 1
Author 2
Author 3

Contents

Vision and Business Case 2

Introduction	2
Business Goals.....	2
Problem Statement.....	3
Business Case	4
Business Constraints	4
Executive Summary.....	4
Use-Case Models.....	5
Use Case UC1: Log-in To System.....	5
Use Case UC2: Student Search.....	5
Use Case UC3: Add a Lesson Taken.....	6
Supplementary Specifications.....	6
Introduction	6
Functionality	6
Usability	7
Reliability.....	7
Efficiency.....	7
Maintainability.....	8
Portability.....	8
Glossary.....	9

Vision and Business Case

Introduction

The Manhattan School of Driving LLC was established in 2008 on the upper east side of Manhattan. For the last 2 years, the school has devoted time to teach students from the age of 16 to 84 by providing instructions in a spotless, dual-brake, late model BMW. The school also offers many different driving courses such as the New York State 5-hour pre-licensing class in its comfortable, fully modern classroom. The 6-hour point and insurance-reduction class is also part of their curriculum.

Business Goals

During the last 2 years, the Manhattan School of Driving has successfully multiplied the number of students from its previous year, while forecasting a continuing growth for the next years to come. In order to guarantee such successful forecast, the school has set some goals and guidelines that they wish to accomplish. Some of these business goals are:

- Passing the DMV on-line training audit approval
- Providing more services for customers online

The Manhattan School of Driving is willing to invest time and money to accomplish their goals. Most importantly, by accomplishing these goals, customer and employee satisfaction will be reached.

Problem Statement

As the school continues to succeed and contract new students and employees, the company has faced a road block using their current paper based system. Keeping students records organized and accessible to both student and employees have become a hassle, error prone and cumbersome. Currently, the school has been able to manage its student and employee data with their current paper-base management system. It has been estimated that using the current paper-base system by the end of this year (2010) will no longer be an effective and efficient way to manage student and employee records. The following is a list of problems faced by students and employees due to the school's current paper-based system:

- Students don't have access to their own training records
- Employees must search piles of folders stored in file cabinets in order to retrieve a student's record
- All student and employee records are stored in file cabinets, unsecured and without backups
- Not all student records are filed correctly and inconsistent data exist
- There is no way to analyze the school data in order to generate valuable information or statistics about their students and performance
- Class schedules are kept in a notebook, which has been the reason for classes being scheduled incorrectly or having classrooms overbooked

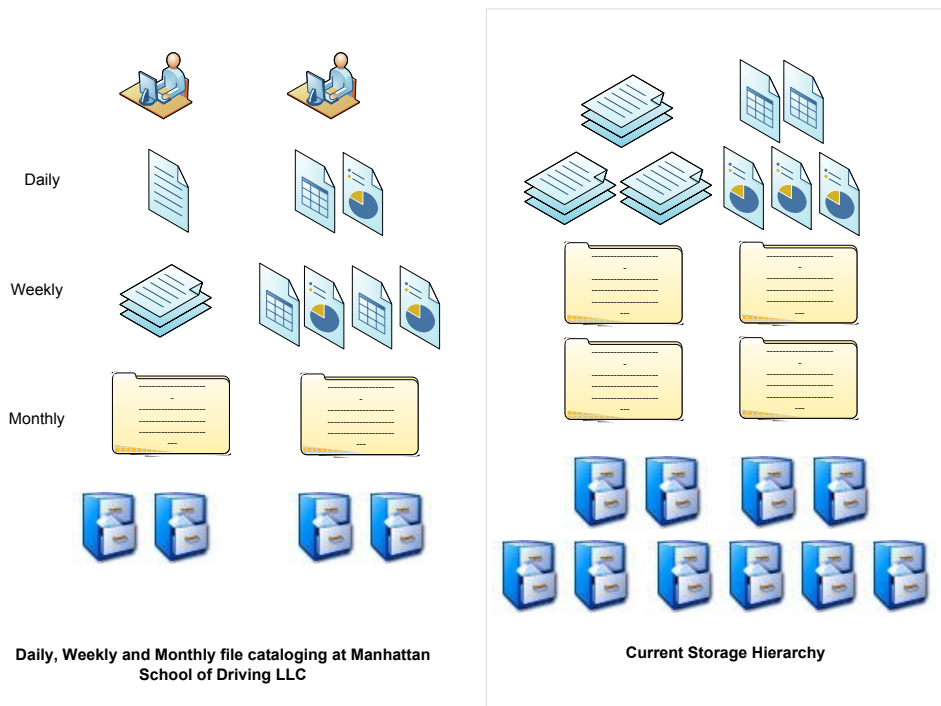


Figure 1: A visual representation of the file storage system currently in place for the Manhattan School of Driving LLC.

Business Case

Implementing the Manhattan School of Driving Management System has many positive outcomes for both the business and the developers. From the business point of view, a new system is definitely needed by the end of this year, the business success and its employees as well as customers satisfaction depends on this new system. Not all driving schools have been fortunate enough to have the large amount of success that Manhattan School of Driving is currently having. Because of this, not all driving schools need a system like the one being requested. Not only would Manhattan School of Driving be one of the few schools in the tri-state area with such advance management system but they will also set the standards for future driving schools and be one of the few to pass the DMV on-line training audit approval.

From the developer's point of view, the software system being requested by the business is not out of the ordinary. Many management systems have been built in the past, and some of these systems have shared their code to the public, allowing developers to reuse existing source code and modules that have been tested to work and proven efficient. Not only is this advantage for the developers but also to the business.

Business Constraints

With every project, for either a small or large business, there is always constraints that both the business and developer should be aware about. For this particular project, these are some possible identified constraints:

- Even if the change is made for the greater good and for the advantage of the greater population, there will always be a few that will dislike the change and will struggle adapting to these changes
- Not all customers and employees can be categorized as technically savvy. There will be a few customers (for example., elderly students) that may struggle learning and using the new system
- Not all students have computers outside the school, accessing their student records may still require either contacting the school directly or visiting the school
- Before the system can prove its advantages and success, the new management system will need to be advertised to all students. A system that is not in used have no positive effect
- The new management system will require a larger budget than the current paper-based system

Executive Summary

The Manhattan School of Driving Management System is meant to help the school reach its 2010 business goals. Accomplishing these business goals will guarantee that the school will remain as one of the best in the tri-state area. In order to accomplish its business goals, the school will have to go through a technological reform, which will include eliminating their current paper-based system and implementing a system that will better manage its growing needs. The current paper-based system has imposed many problems, some which have been described in the problem statement. Implementing the new management system should alleviate these problems and provide flexibility for future business growth.

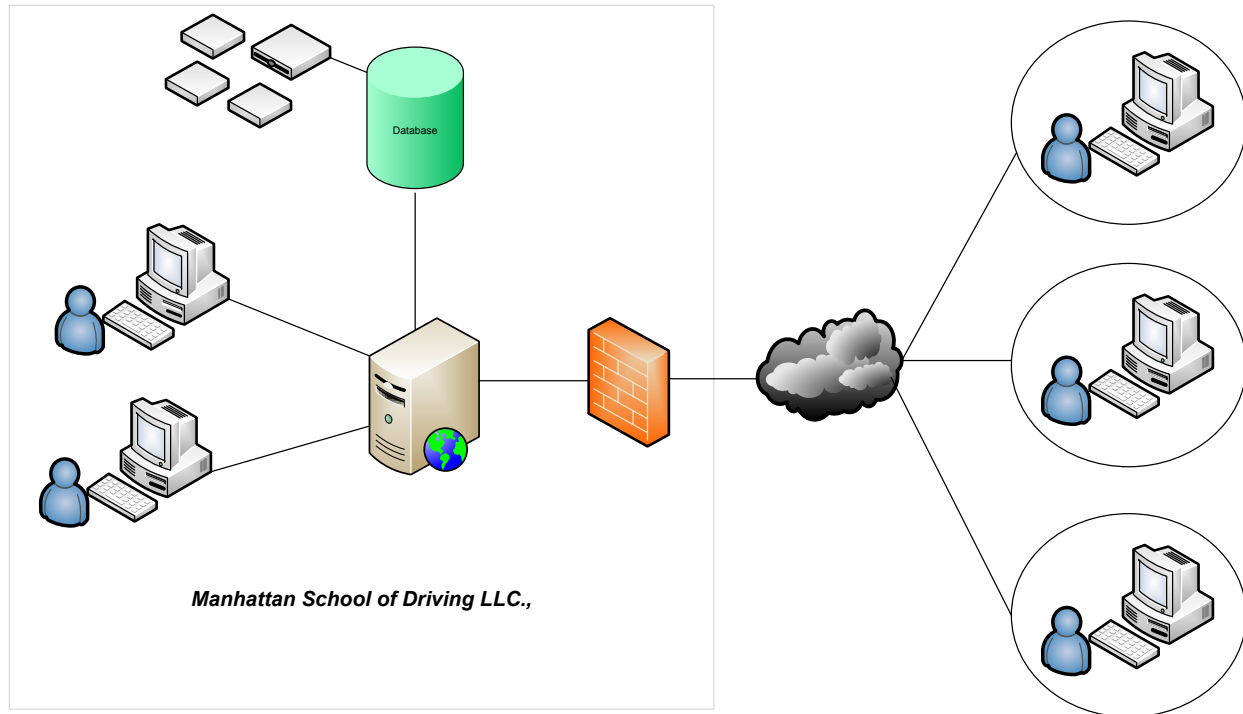


Figure 2: A visual representation of the technological reform

The advantages of implementing this system are many. Not only will the business get a system that they can use for many years to come, but this will also open a brand new market for all driving school if the system is implemented successfully. To the developer's advantage, the developers can make good use of software reusability for the current project and for future driving school management system projects as well.

Use-Case Models

Use Case UC1: Log-in To System

Primary Actor: Receptionist, Trainer, Student

Preconditions: Receptionist, Trainer or Student must have a username and password

Post conditions: Users log in to the system

Main Success Scenario:

1. Receptionist, Trainer or Student navigates to the system's web site using the link provided and a web browser that meets system requirements.
2. Receptionist, Trainer or Student enters his/her username and password and clicks OK.
3. Receptionist, Trainer or Student is redirected to his/her profile home screen.

Use Case UC2: Student Search

Primary Actor: Receptionist, Trainer

Preconditions: Receptionist or Trainer must log in to the system

Post conditions: Zero or many student records are display

Main Success Scenario:

1. Receptionist or Trainer clicks on 'Student Search' from the main menu and gets redirected to the Student Search page.
2. Receptionist or Trainer enters a combination of first name, last name or student number and selects Search.
3. A table containing zero or many records is displayed based on the search result.

Use Case UC3: Add a Lesson Taken

Primary Actor: Receptionist

Preconditions: Receptionist must log in to the system. A student must provide the receptionist with a 'Lesson Taken Receipt', signed by the trainer specifying the lesson type, lesson length, the lesson start time and lesson end time.

Main Success Scenario:

1. Receptionist searches for the student mentioned in the 'Lesson Taken Receipt'
2. The system displays a table list with the search results, which includes the student being searched
3. Receptionist clicks on the student name and is redirected to the student's page
4. From the 'Student Actions' menu, the Receptionist clicks on the 'Lesson Taken' link
5. The system redirects the client to a page where he/she can add a lesson taken and all its required information

Supplementary Specifications

Introduction

This document will serve as a repository for all requirements not captured in the use cases.

Functionality

Logging and Error Handling

The management system should log the following events:

- Failed log-in attempts
- Failed data-backups

Security

The usage of the management system should be available to receptionist, trainers and students who have active usernames and passwords. There is no alternate access to the management system without proper system authentication using the system's main website. This includes no backdoor access from developers to any user (authorized or unauthorized).

Usability

Human Factors

The Manhattan School of Driving Management System will be used in an office environment by employees (receptionist and trainers) and students. The system will also be available outside the office environment. Since this is a web based application, students and employees can access the system online by typing the system URL link on their web browser.

The target audience is employees and students which may vary in age from 16 to 84. Since this web based application will be used by many different people with different ages, the system must be user friendly and easy to learn. The term “User Friendly” can be very vague and generic, so a formal definition can consist of the following:

- Adequate Text Size – The text size used by the system should be large enough to read from arm’s length.
- Descriptive Menus – All menus must have a descriptive name that is useful in indicate its purpose. For example, Log-off, would obviously indicate that you would like to terminate your session and Print, would indicate that you would like to print the page currently in view.
- Pleasing Colors – Some colors may be hard for certain people to see. For example, a bright red or pink might not be pleasant to someone’s eyes while a color that is too light might be hard for someone to notice.
- Tool-Tips – certain forms may require a “tip” to indicate how the data should be entered on the system or what the purpose is for a particular button or field box. Where ever it might seem required, a tool-tip may be placed to help the user navigate and use the system.

Reliability

The Manhattan School of Driving Management System must behave in a reliable manner in its working environment. A “reliable manner” is a specified level of performance that is determined only by specified conditions. These specified conditions will be mentioned in further detail during the elaboration phase. For now we will consider the following characteristic:

- Maturity – the system should avoid failure as a result of faults in the software by being in a state of being fully developed
- Fault Tolerance – the system should be capable of maintaining a specified level of performance in cases of software faults or of infringement of its specified interface
- Recoverability – the capability of the software product to re-establish a specified level of performance and recover the data directly affected in the case of a failure
- Compliance – the capability of the software product to adhere to standards, conventions or regulations relating to reliability

Efficiency

The system should be capable of providing appropriate performance, relative to the amount of resources used, under stated conditions. The specified conditions will be mentioned in further detail

- Time Behavior – the system should be capable of providing appropriate response and processing times and throughput rates when performing its function, under stated conditions
- Resource Utilization – the system should use an appropriate amount of resources when the system performs its function under stated conditions
- Compliance – the system should adhere to standards or conventions relating to efficiency

Maintainability

The system should be modifiable. Modifications may include corrections, improvements or adaptation of the system to changes in environment, and in requirements and functional specifications. The following characteristics will define if a system is maintainable:

- Changeability – the system should easily enable a specified modification to be implemented
- Stability – the system should be able to avoid unexpected effects from modifications of the software
- Testability – the system should easily enable modified software to be validated
- Compliance – the system should adhere to standards or conventions relating to maintainability

Portability

The system should be developed in a way that it is easy for someone to transfer it from one environment to another. The following characteristics will define if a system is portable:

- Adaptability – the system should be able to adapt to different specified environments without applying actions or means other than those provided for this purpose for the system considered
- Install ability – the system should be capable of being installed in a specified environment
- Co-existence – the system should co-exist with other independent software in a common environment sharing common resources
- Replace ability – the system should be used in place of another specified software product for the same purpose in the same environment
- Compliance – the system should adhere to standards or conventions relating to portability

Glossary

Term	Definition	Aliases
Management System	is a collection of procedures used to manage work flow in a collaborative environment	
DMV	Department of Motor Vehicle	
Backdoor	is a method of bypassing normal authentication, securing remote access to a computer, obtaining access to plaintext, and so on, while attempting to remain undetected	
Function	is a structured representation of the functions, activities or processes	Action, Event
Environment	state which provides software services for processes or programs while a computer is running	infrastructure