Software Design Document
for
EZ Tag System

Version 1.0 approved

Prepared by Isabel Almaguer, Carrie Dumit, Jason Jensen

<CIJ LLC>

March 24, 2016

Team Contribution

Name	Contributions
Jason Jensen	Implemented code, README file, created UML, reviewed documentation for grammar and corrected formatting.
Carrie Dumit	Created rough draft of documentation.
Isabel Almaguer	Completed documentation, reviewed final draft.

Table of Contents

1 Introduction

1.1 Document Description

This Detailed Design Document (DDD) provides design details for the ***EZTag*** Toll System. This DDD will be customized to be in accordance with software requirements specification, however it is based on the core ***EZTag*** product solution. The purpose of it is to specify the modules used in the EZTag System, Data Structures used, the files storage system, and a class diagram to show the relationships.

The overall scope of the DDD is to give an understanding of how to implement the backbone of an EZ Tag System. More specifically, we will be looking at the back end development of said system, which will include Lane System, Account Management System, Toll Schedule and Central Toll Management Systems. Stated details about the product's performance will be tested and outputs will be provided on a separate document dedicated only for testing the product (DTP). The project was conceived and will be developed by a team of three throughout the spring semester.

This provided documentation is intended to be a detailed guide for all readers that are interested in the back end development of an EZ Tag System, and all people involved in the primary functional areas of the system. It provides design details for our Toll System that may be read from front to back for a complete understanding of the project. The overall document is separated in sections, and designed to be read by specific topic to provide rapid access to material as needed. Toll management personnel and the developers in charge of maintenance of the product can read the specific sections, as well as the users who are interested in the details of the software.

Developer: The developer who wants to read, change, modify or add new requirements into the existing program, should consult this document and update the requirements in the proper manner. Therefore in order to properly modify the program, the changes made should be available in all phases of the process where the changes occurred and be stated in this documentation.

User: The user interested in the details of the program can review the diagrams and the specifications presented in order to obtain better understanding. In this manner the user can also determine if the software has all the suitable requirements.

Toll management personnel: The people in charge of the correct functioning of the entire system need this document to guide themselves and to execute the program correctly.

Summary:

The purpose of this document is to specify The modules used in the EZ Tag System, Data Structures used, the files storage system, and a class diagram to show the relationships.

The overall scope of this document is to give an understanding of how to begin implementing the backbone of an EZ Tag System. T\More specifically, we will be looking at the back end development of said system, which will include lane operations, cash management, toll schedule and central toll management systems. The project was conceived and will be developed by a team of three throughout the spring semester.

Our documentation is intended to be a detailed guide for all readers that are interested in the back end development of an EZ Tag System, and all people involved in the primary functional areas of the system.

Related Documentation:

- Software Requirements Specifications
- Detailed Test Plan

Acronyms:

- DDD- Detailed Design Document
- DTD- Detailed Test Plan
- VIS-Vehicle Identification System
- PMS-Personnel Management System
- LS-Lane System
- AS-Account System
- VLANs-Virtual Local Area Networks

1.2 System Overview

The program for controlling the EZTag toll system is based on three primary functional areas:

Account System (AS): Account Collection System based on the collection of fees solution, including lane hardware and software systems, Digital Toll Management, and Cash Management System.

Lane System (LS): Usage of time-stamps for the Lane Management System to determine fees.

Integration: Integrating the AS and LS through the controller system and data exchange between serialized files. Our system is able to support multiple accounts and types of accounts and collecting revenue based on these. An electronic identification will be implemented for Lane System also line direction are defined on software configurations to support specific operational modes.

The system performs cash and electronic toll collection.

It implements a **Virtual Scanner System (VSS)** – simulates a vehicle entering and exiting the toll system. This process enhances the vehicle classification to have accurate fees for each type of account. The cash collection is tracked and verified at each step along the way before making transactions and have them serialized. A collection of functions which provide accurate, reliable, and secure cash management eliminates the opportunities for revenue loss.

Implements a **Personnel Management System (PMS)** – Operational features that provide the ability to perform employee maintenance and scheduling of line time changes along with fee changes. This set of functions which are in charge of controlling the access to the system functionalities and to the tolling systems are only available to registered employees accounts. Only the people who are authorized to maintain the system and operate specific should be allowed access.

2 Architecture

2.1 Introduction

This is a large computer-based system for information processing. The information processed is distributed to all computer/components of the system and is intended to run on an integrated group of cooperating computer linked by a network.

In this system the sharing of hardware, such as sensors, is present and current processing is applied to enhance performance. Scalability will be applicable in this system, meaning that new resources can be added in the future if needed. It also has a fault tolerance since it has the ability to continue in operation after a fault has occurred after wrong input data. The major inputs expected will be the required information to register a new account and payments for tool fees.

The system provided a systematic user interface in which implements windows, icons, pop-up menus and button to gather all information necessary for each process.

The architecture applied is a client-server type, and will run on multiple platforms. The services are called by clients and the servers provide different services needed.

The overall system architecture for the tolling design uses the same lane layout in all lanes of the same type. This includes all mode lanes which support cash or automatic coin machine.

The key component of the lane layout is the VSS vehicle detection is implemented at this step so the classification can be obtained in order to process correct fees. The configuration and spacing of the “sensors”, and the associated processing algorithms, are designed to provide a fairly accurate system. The sensors are aligned on the lane stripes to detect vehicles in the tolling zone.

If we keep in mind that users from all over the city can access their accounts from home, a Local-area network would not be the best idea. The user can access their account and view their information from any computer, on the web, so a Wide-area network would have to be used.

Since this is a small scale implementation, there will only be one of each module. However, in a large scale real time implementation, each lane will be treated as its own module, processing its own thread, in order to operate in real time as required on a tolls system.

2.2 Data

This system utilizes serialized files to store account information.

2.2.1 Introduction

Data essential to inform the (LS) and (AS) about due toll and about updates on accounts for vehicle paid activity, is kept on serialized files. In this form all the typical

actions performed by the system are stored and can be retrieved in response to each event at the moment a vehicle enters a tolling zone.

Controller informs (LS) of vehicle presence, then (LS) adds vehicle to serialized files or the server looks up class and information (account) associated with the vehicle in (AS) Next toll due is displayed, transaction is made and thank you message displayed.

Transactions from lanes are posted to serialized files and retrieved when (AS) and Controllers make requests.

Once the system is up and running, updates will not be very often. Unless new technologies are introduced, the system will only need to be looked at every 4 months. Since we are taking in personally identifiable information, the system requires a password to log in to a unique account name assigned to each user. If a user does not recall username or password, they can receive that information back via email on their account.

2.2.2 Schema

At this time, no database will be implemented into our model.

2.2.3 File and Data Formats

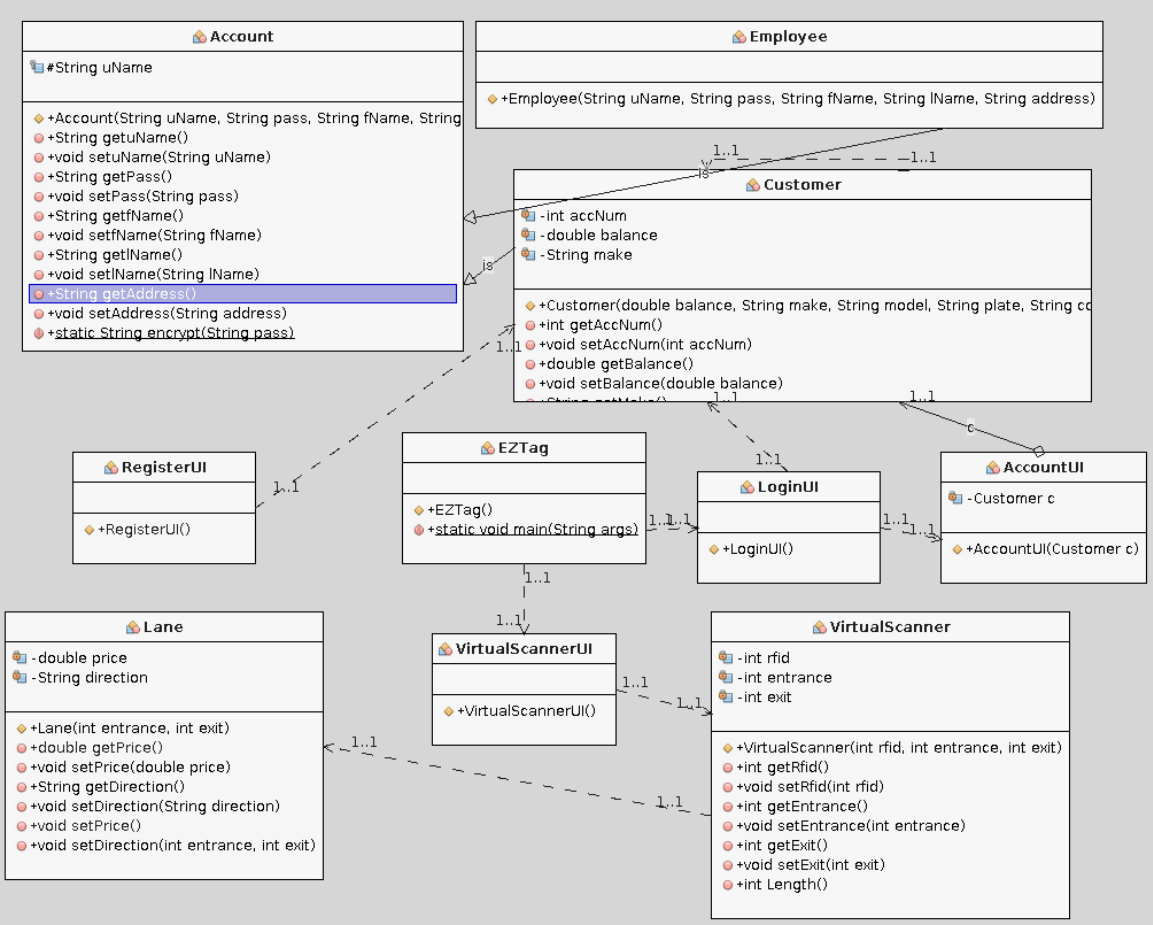
The (AS) collects required information to create account to later be used by (LS) operations. The personal data acquired from user, such as name, last name, driver's license and account number, is kept on serialized files, one file per account. This option of keeping all accounts on separate files was opted over keeping them all in one big file for the reason that it facilitates the retrieval of data for primary process such as processing payments and deletion of expired accounts.

2.3 Communication

The following diagram show the basic network topology for the mode lanes and tolling zones, respectively. The diagram shows a zone with 2 lanes to reflect two types of modes.

In the diagram its shown how the (LS) connects the lane equipment (sensor) to the controller. The communication is configured with multiple virtual local area networks (VLANs) to isolate network traffic associated with some equipment. Specific VLANs and port assignments will be developed during the installation and deployment stage of the project.

Diagram



In addition to network communications, there is a direct hard-wire connection between sensors and lanes. These wires are split, with one route connected to the (camera ? Include in hardware specifications?) ? , sensors and the other to the network to create event for capturing a image and record vehicle entrance.

2.4 Code

2.4.1 Introduction

We based this design on the idea that each user has an internet connection that can allow them to access their online EZ Tag account. We kept in mind that we would have to implement various integrated libraries to store account information. We also have to keep in mind that we will be require to put in some level of security measures to protect the sensitive information linked to each users account.

2.4.2 Modules

2.4.2.1 Module Account

This is the module in charge of storing all the customer's information such as passwords and usernames. This will be the parent class for the customer and employee classes. It is serializable so it information can be saved and retrieved. The only methods implemented are the accessors and mutators.

2.4.2.1.1 Internal Functions

Encrypt - encrypts the password using MD5.

2.4.2.2 Module AccountUI

This module accepts an account number, then opens and displays the specified account.

2.4.2.2.1 Internal Functions

2.4.2.3 Module Customer

This Module extends the account module. In addition to the information in the account module, it will store account number, balance, and card and payment information. In addition to the standard assignment, the constructor will generated a random nine digit account number.

2.4.2.3.1 Internal Functions

Save - will open the an account file and write the current object to that file. If no file exist, it will create that file.

Verify- attempts to open a specialized account file and validates user name and password.

Open - Opens a specified account file, and maps the file information to a new customer object.

Generate Charge - Adjusts the balance based on calculated price and number of tolling plazas.

2.4.2.4 Module EZTag

This module is the main client controller; it is the GUI that launches first. There are four buttons, one that closes the GUI, one that opens the virtual Scanner UI, another that opens the login GUI, and the last one that opens the register GUI. Each time a button is pressed, it opens the GUI for that particular option.

2.4.2.4.1 Internal Functions

2.4.2.5 Module Employee

This module currently not implemented, however, it will hold extend the account module, and hold employee specific information.

2.4.2.5.1 Internal Functions

2.4.2.6 Module Lane

Lane module holds information regarding a toll lane, specifically the direction and price at various times of day.

2.4.2.6.1 Internal Functions

Set Price - currently only implemented for the Katy tollway, it accepts the direction a vehicle is traveling, then polls the systems to determine current time and set the price per gate accordingly.

Set Direction - calculates direction based on entrance and exit.

2.4.2.7 Module LoginUI

Log in button accepts the account number, username, and password. It then verifies that the requested account exist, and if it does, it passes the information to a new account UI object.

2.4.2.7.1 Internal Functions

2.4.2.8 Module RegisterUI

This module displays a GUI that requests and accepts information required to create a new account. It then passes that information to a new customer object and then calls the object to save its self as a serialized file.

2.4.2.8.1 Internal Functions

2.4.2.9 Module Virtual Scanner

This module is the main server side controller. It behaves as the event handler receiving the RFID, entrance and exit. It then calls the requisite functions to generate a charge for the appropriate account.

2.4.2.9.1 Internal Functions

Length - calculates the number of tolling plazas.

2.4.2.10 Module Virtual Scanner UI

This model will take the place of the sensor hardware for this implantation. It will accept the RFID for entrance and exit for vehicle and pass it to the Virtual Scanner.

2.4.2.10.1.1 Internal Functions

2.4.3 Interfaces

For this prototype, interfaces are not implemented. A possible interface will be the account module.

3 Operation

The program is expected to operate by implementing main operations that include definition of account types and direction and type of lane, which are requested through a controller in charge of making the program run smoothly.

Lane Operations implemented:

- Change operating mode of lane- function gives option to change direction or lane. Only accessible to registered employee.
- Opening and closing of lane- In case of emergency or accident this function should disable specific lanes.
- Assignment of toll schedule- Creates new schedules when exceptions need to be made.
- Determination of rates according to schedules - Has the ability to make exceptions on toll fees on holly days.
- Lane events summary-Displays daily transactions filtered by date, lane type.

Account Operations implemented:

- Transactions-Detailed transaction records amounts and payment types.

Account subclass (Employee)

Management- management and administration of employees within the system.

Accessible- Determine the level of accessibility to certain functionalities of the program for each use.

Controller functions:

- Making functions requests to specific operations.

3.1 User types

The (AS) will be in charge of managing the types of user and for creating accounts with specific functionalities for each user account. Each account is defined specifically for common user or customer, for employees with basic privileges, and for administrative personnel with maintenance privileges. The user interfaces dedicated for each user will be similar in respect to design and of data gathered for registering new user, collection of data will be the same, (personal data). Consistency is maintained throughout all user interfaces and only minor changes are implemented to dedicated to management personnel only.

Previous knowledge about how the system works is needed in order to execute certain function of the management account system. Employees who have access to those functionalities should be familiar with the hardware properties and how are they integrated to the system. The only security measure that are used to prevent access to those methods are the (ID) Identification number given at the moment the registration of

new employee occurs. And thus the set of functionalities reserved for all managerial accounts are accessible after performing adequate registration.

Change of schedule, change of tolling fees, deletion of accounts, and creation of personnel account are functions reserved to the management accounts only.

Registering, logging in, checking balance, making payment and requesting deletion of account are all functions accessible to all users but mostly performed by the regular client.

Because this is a large scale system, the system should be able to handle thousands of users, and hundreds at once.

3.2 Scenarios

Registration Scenario: User is given the option to create a new account, and chooses between to option, Regular Account or Employee Account. If regular account is chosen then the user is prompted to enter his/her information such as: Drivers license , full name, account number, and a Ez tag number , a username and password are generated at the end. If user selects other option another window is displayed giving the option to choose between Administration or just employee. Same process is performed except that a Identification Number for employees is generated instead of an Ez Tag number.

Logging in Scenario: A window is displayed prompting the user to enter his/her username and password. Options to view transactions made and view balance are given.

Make payment Scenario: Controller communicates with (LS) about a car entrance to the tolling zone. (LS) performs operation of calculating toll fee and performs transaction, then it passes information to (AS) so it can update information.

Delete Account Scenario: User makes request to delete account, controller passes information to (AS), account system calls function to look up for number account and disables account. Information is updates and displays a message of successful operation.

3.3 Installation

This is a general indication of the different programs and utilities that will be installed to the servers and workstations supporting the ***EZTag*** Toll System. Provided major software components are the same ones mentioned throughout the document however not all subsystems will be specified. Specifically, it should be noted that changes might be made between now and the final installation of the software for the project.

The service-oriented architecture of the tolling system creates a uniform deployment and installation of the client and server environments within the system. All services and supporting framework libraries are installed to equipment connected to the network. Likewise, the client application which exposes modules based on the logged in user's employee type security settings, the lane management console, is installed to the general user workstations. Workstations with dedicated functionality, such as time management, and price management also have specific software installed to support their needs. The ***EZTag*** software includes all toll collection system products developed for any specific customer. The only thing needed to be installed is Java Platform, but it is safe to assume most users have that installed already.

4 Development

The software team consists of three members, each made use of source code change management (Github) to facilitate team code development process and to keep record of all steps of the process. We allowed explicit code check-out procedures with annotation of changes, and comments for each functionality added to the system. This control system provides code “adherence ” for indication of the latest version for use within the build system. Major code versions are maintained as well to allow rollback, reference and even comparison to a previous version.

The development of a primary Account class was necessary in order to be extended to other classes, Employee and Costumer, and to in order to test the functionalities of the Lane System class.

Because the software is a product, each new functionality build involves all of the code, regardless of how much has changed. ***EZTag*** will deliver notes to describe the changes made to this prototype version and list the new and modified modules and the end of spring semester. Our development time estimate, from the prototype and on, is one month. The entire system required a total of three months’ development time.

5 Miscellanea / Appendices

5.1 Conformance with standards

HCTRA security standards located within the privacy policy internal document.

5.2 Interoperability with other systems

The main external systems will be the web browsers the users would have to access to log on to their accounts.

5.3 Expandability

None implacable to this system.

5.4 Debugging

In addition to the internal debugging provided by the IDE, we utilized system out functions when the system did not perform as expected to reduce search overhead.

5.5 Security

To authenticate a user and allow access to the system a username and password will be required. A user with a regular account will only have access to their own account so will be able to create and modify their login information at will. An employee will have access to all accounts so an administrator will create an account and provide them with their username and a temporary password that will need to be changed.

Each account has certain personal information assign to it. Because of that, identity theft is the main possible threat we face with such a program. Although assigning random account numbers to each account can assist, data encryption will be implemented to keep accounts secure. Specifically all passwords will be encrypted with MD5.

5.6 Open Issues

We will need to map all of the total lanes, directions, times and changes as well as assigning each entrance and exit a unique ID. Unique accounts number needs to be generated, and we will have to verify that the account log in name is unique as well.

5.7 Glossary

UI: User interface the portion of the program that is displayed to the User.
INTERFACE: The program display.

EZ Tag: Name of the Harris County Toll Road Authority system.

5.8 Bibliography

"EZ Tag." *Login*. Harris Country, n.d. Web. 29 Mar. 2016.

"HOV / HOT Lanes (Express Lanes)." *METRO*. N.p., n.d. Web. 29 Mar. 2016.