Project Report:Online Inventory Management System

Computer Science (University of Kerala)

Online Inventory Management System

PROJECT REPORT

Submitted by

HARIKRISHNAN R [65418802004]

to the University of Kerala
in partial fulfillment of the requirements
for the award of the degree of
Master of Science in Computer Science

DEPARTMENT OF COMPUTER SCIENCE

COLLEGE OF APPLIED SCIENCE, ADOOR

JULY 2020

INSTITUTE OF HUMAN RESOURCE DEVLOPMENT

COLLEGE OF APPLIED SCIENCE

Adoor, Kerala

CERTIFICATE

This is to certify that this bound volume is a bona fide record of the project work done on "**Online Inventory Management System"** by **HARIKRISHNAN R** (**65418802004**) of fourth semester M.Sc Computer Science in partial fulfillment of the requirement for the award of the Master's Degree in Computer Science from the University of Kerala.

Dr. Jayamohan M
Faculty In-charge

Mr. Ajayakumar T.
Head of the Department

Mrs. Latha P
Principal

External Examiner

DECLARATION

I hereby declare that the project entitled " **Online Inventory Management System"** has been carried out by me in partial fulfilment of the requirements for the award of M.Sc Computer Science, is a report of original project work done during the period of my study (2018-2020) in **College of Applied Science, Adoor** (Affiliated to University of Kerala),under the supervision and guidance of **Mr. Ajayakumar T.**(Head of Department).

Place: Adoor **HARIKRISHNAN R**

Date: / /2020

ACKNOWLEDGEMENT

Through the acknowledgement, we express our sincere gratitude to **GOD** and all those people who have associated with this assignment and have helped with it and made it a worthwhile experience. We take immense pleasure in ranking **Mrs. Latha P**, Principal of College of Applied Science, Adoor for her kindness that has enabled this project to be complete one.We wish express our deep sense of gratitude to **Mr. Ajayakumar T**(Head of Department) for their excellent guidance and constant encouragement to successfully complete this project.

We wish to express our heartfelt thanks to our internal guide **Dr. Jayamohan M**(Assistant Professor, Department of Computer Science) his valuable guidance, readiness to clear all doubts and for guiding in the right way to make this project a successful one.Also we extend our thanks to the various people who have shared their opinion and experience through which we received information crucial for our report.

Finally, yet importantly, we would like to express our heartfelt thanks to our beloved parents for their blessings, our friends

HARIKRISHNAN R

ABSTRACT

Online Inventory Management System (OIMS) is designed for the effective management of inventory control in various commodities. The project is to develop an online portal for managing the stock and shipping of vehicle spare parts as a case. This helps in controlling the centralized distribution and regionalized usage of various spare parts. Spare parts from the central store are carried to different sub stores of each district. It has to be done by creating intents by the sub stores for the parts which are needed for the specific vehicle which have to be repaired. Then the intent should be approved by the controlling authority. This project gives a better solution for the existing system. Lots of human effort is needed for maintaining the traditional existing system. This project can reduce those efforts to large extent and minimizes the effort for writing and keeping the accounts of inventory.

The proposed project is intended to reduce human effort and time for consolidating the stock accounts. The existing system is fully manual and hence error prone. This in turn wastes lots and lots of valuable time and money. Moreover when consolidation of reports is done manually by non-skilled staff it causes lot more problems. To avoid all these, a new system is proposed.

In this proposed system when all the related entries are done online by the staff regionally simultaneously with work and they are compiled centrally in the headquarters chances of error is avoided along with wastage of time.

CONTENTS

1. INTRODUCTION

1.1 PROBLEM STATEMENT

The project is to develop an online portal for managing the stock and shipping of parts, specifically vehicle spare parts as a case. The portal is intended to reduce human effort and time for consolidating the stock accounts. The system shall be easily accessible from any platform with maximum user-friendly design.

1.2 OBJECTIVES

The major objectives of the project are:

- Give the details of the central store and all the sub stores of Vehicle Spare parts outlets in various districts.
- Capture all details of spare parts available in various storehouse.
- Maintain inventory and spare management of store.
- Provide sufficient interactive screens to access and update all types of information.
- Provide a reliable ad consistent data store.

2. SYSTEM ANALYSIS

2.1 INTRODUCTION

System analysis is the most critical process of information development. In system analysis, the problem is identified, alternative solutions are evaluated and the most feasible solution is recommended. An initial investigation is performed to identify the current problems and solutions for the smooth functioning of the organization. Each module thoroughly studies and all the recommended for the project are gathered. The problem is split into module and is viewed at various angles .this lead to the evolution of project.

2.2BACKGROUND ANALYSIS

In existing system, all the operations are done manually and hence time consuming. The system requires lots of human effort for the proper and error free management of inventory. Any kind of mistakes during the manipulation of inventory accounts may lead to complex issues and thus leads to inefficiency and improper working of the system. Therefore the existing system is neither convenient nor efficient.

Drawbacks of existing system

- Highly time consuming.
- Requires lots of effort.
- Inconvenient and inefficient.
- Error prone and risky.

2.3 PROPOSED SYSTEM

The proposed project is intended to reduce human effort and time for consolidating the stock accounts. The existing system is fully manual and hence error prone. This in turn wastes lots and lots of valuable time and money. Moreover when consolidation of reports is done manually by non-skilled staff it causes lot more problems. To avoid all these, a new system is proposed.The proposed system will be developed in Python framework using MySQL as database management system.

The system helps in controlling the centralized distribution and regionalized usage of various spare parts. Spare parts from the central store are carried to different sub stores of each district. It has to be done by creating intents by the sub stores for the parts which are needed for the specific vehicle which have to be repaired. Then the intent should be approved by the

controlling authority. This project gives a better solution for the existing system. Lots of human effort is needed for maintaining the traditional existing system. This project can reduce those efforts to large extent and minimizes the effort for writing and keeping the accounts of inventory.

In this proposed system when all the related entries are done online by the staff regionally simultaneously with work and they are compiled centrally in the headquarters chances of error is avoided along with wastage of time.

All the drawbacks of the current system can be avoided using this system. Now there is no need to write and keep all the accounts of inventory. The proposed system gives a far better solution for the operations done in the store of Health transport wing. This project is done on considering the central store and the various sub stores to make their manually done processes online.

ADVANTAGES:

- Reduces lots of manual effort for keeping the accounts of store.
- It is totally free and user friendly.
- It is quick, easy and convenient.
- It reduces the risk of making errors.

  Easy for the controlling authority to check the inventory details from anywhere at any time.

2.4 FEASIBILITY STUDY

A feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, time and effort spent on it. Feasibility study lets the developer to foresee the future of the project and its usefulness.

Feasibility study involves eight steps:

- Form a project team and appoint a project leader.
- Prepare a system flow chart.
- Enumerate potential management system.
- Describe and identify characteristics of management system.

- Describe and evaluate performance and cost effectiveness of each management systems.
- Weight system performance and cost data.
- Select the best management system.
- Prepare and report final project directive and management.

There are three aspects in the feasibility study of the preliminary investigation:

- Operational Feasibility
- Economical Feasibility
- Technical Feasibility

OPERATIONAL FEASIBILITY

This system is operationally feasible, since the system is providing an attractive user interface to the operator/end user. This system is designed in such a way that the end user will not face any problem while working with this system. Validations are done in such a way that the system prevents wrong data from entering the system. Tool tips and message boxes are used in such a way that users or end user can easily operate the system.

ECONOMICAL FEASIBILITY

This system is economically feasible. The organization will not spend much for the development of the system. The cost for proposed system is overweighing the cost and effort involved in manually maintaining the existing system. The proposed system is cost effective. The cost of manual processing and paper work can be reduced considerably. The system also reduces administrative staff & other staff to do various tasks. Financial benefits exceed the cost for developing the system. The proposed system will increase profit and reduce expenditure.

TECHNICAL FEASIBILITY

This system is technically feasible. This system is web based and thus can be accessed through any browsers. The solution is practical since the objectives of the system development are achievable and realistic. The technology to be used is available, this includes use of scripting languages PHP, HTML and MySQL database to develop web based

applications. Since there is not much difficulty in getting the required resources for the development & deployment of the project. The proposed system is technically feasible.

2.5 HARDWARE REQUIREMENTS

Server Requirements:

The web server can be implemented in a rental space. However the optimum requirements for a separate server machine to install and run the portal is given below.

- Processor : Any x86/x64 based microprocessor
- Hard disk drive : Minimum of 80GB
- Memory : 512MB or Greater

Client Requirements:

Any PC with internet connectivity will serve the client side operations.

2.6 SOFTWARE REQUIREMENTS

Server Requirements:

- Operating System : Any Operating System
- Database : MySQL
- Web server : Python Built - in Server
- Browser : Any web browser
- Server side scripting language :
- Front End : HTML, CSS, JavaScript
- Back End : Python 3.7 (Django) PostgreSql 12.1
- IDE : Pycharm / Spyder

3. SYSTEM MODELLING AND DESIGN

3.1 INTRODUCTION

System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

3.2 PROCESS DESIGN

Modules:

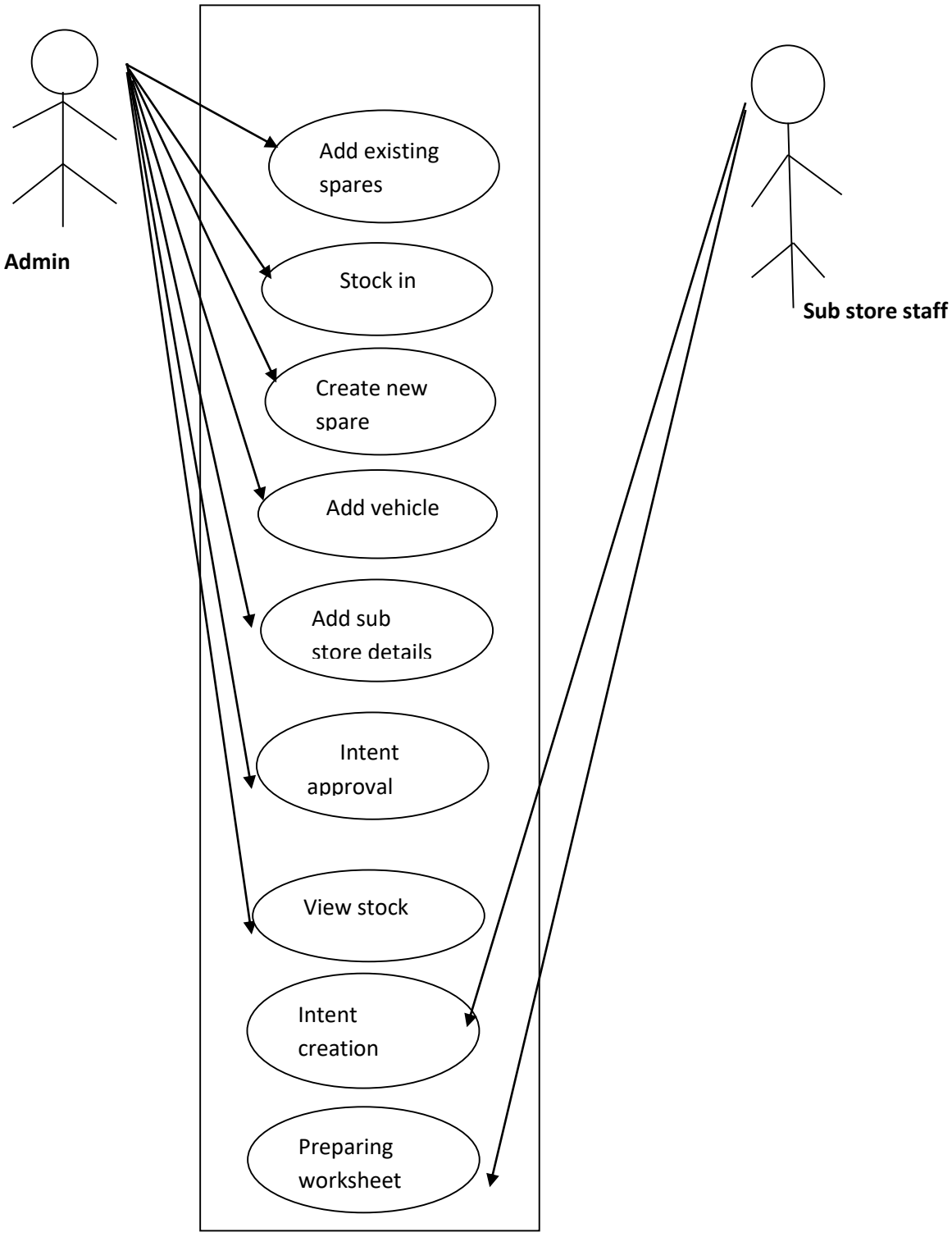
Admin login: The system is under supervision of admin who checks and manages the inventory of the store.

User login: Users are the staffs of different sub stores.

Vehicle Master: To capture all the details of the vehicles in the Health transport wing.

Inventory Management: To manage all the manipulations of inventory accounts.

3.3 USECASE DIAGRAM

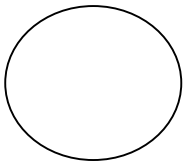


3.4 DATA FLOW DIAGRAM

A dataflow diagram shortly termed as DFD has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So it is a design phase that functionally decomposes the requirements specifications down to the lowest level of detail. It is the starting point of the design phase that functionally decomposes the requirements specifications down to the level of details.

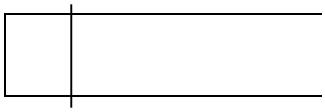
The merit of the DFD is that it can provide an overview of the data to be processed by the system, the data to be transformed, the files to be used and the flow of data along the system. It has illustrating the essential component of a process and the way of interaction.

DFDs mainly use the following symbol:



Circles are used to represent process that converts data into information.

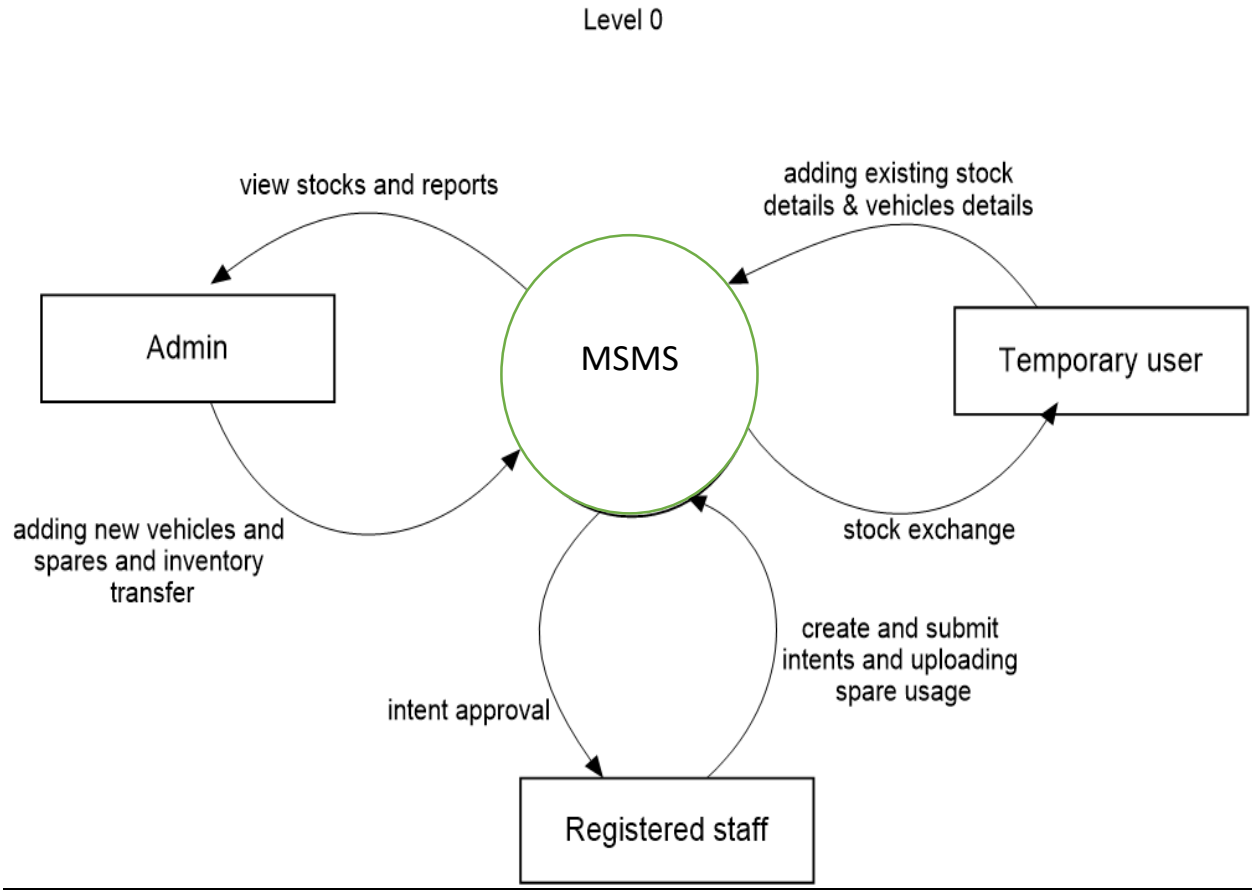
Rectangles are used to represent external entities, which are the sources of data that enter the system or the recipients of data that leave the system.



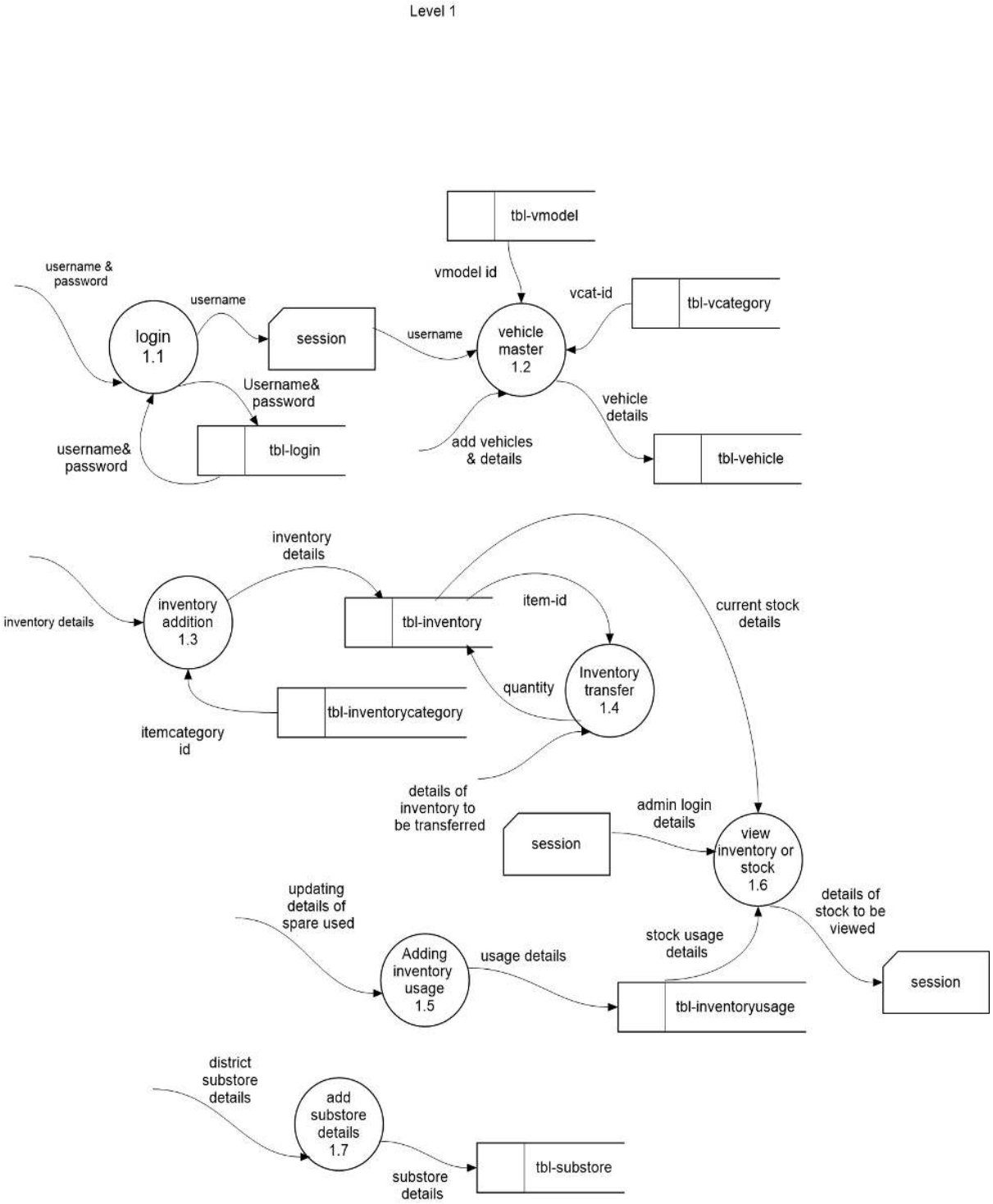
Open rectangles are used for representing databases.

Arrows are used to represent the data flow. Data flows represent the movements of data between other components.

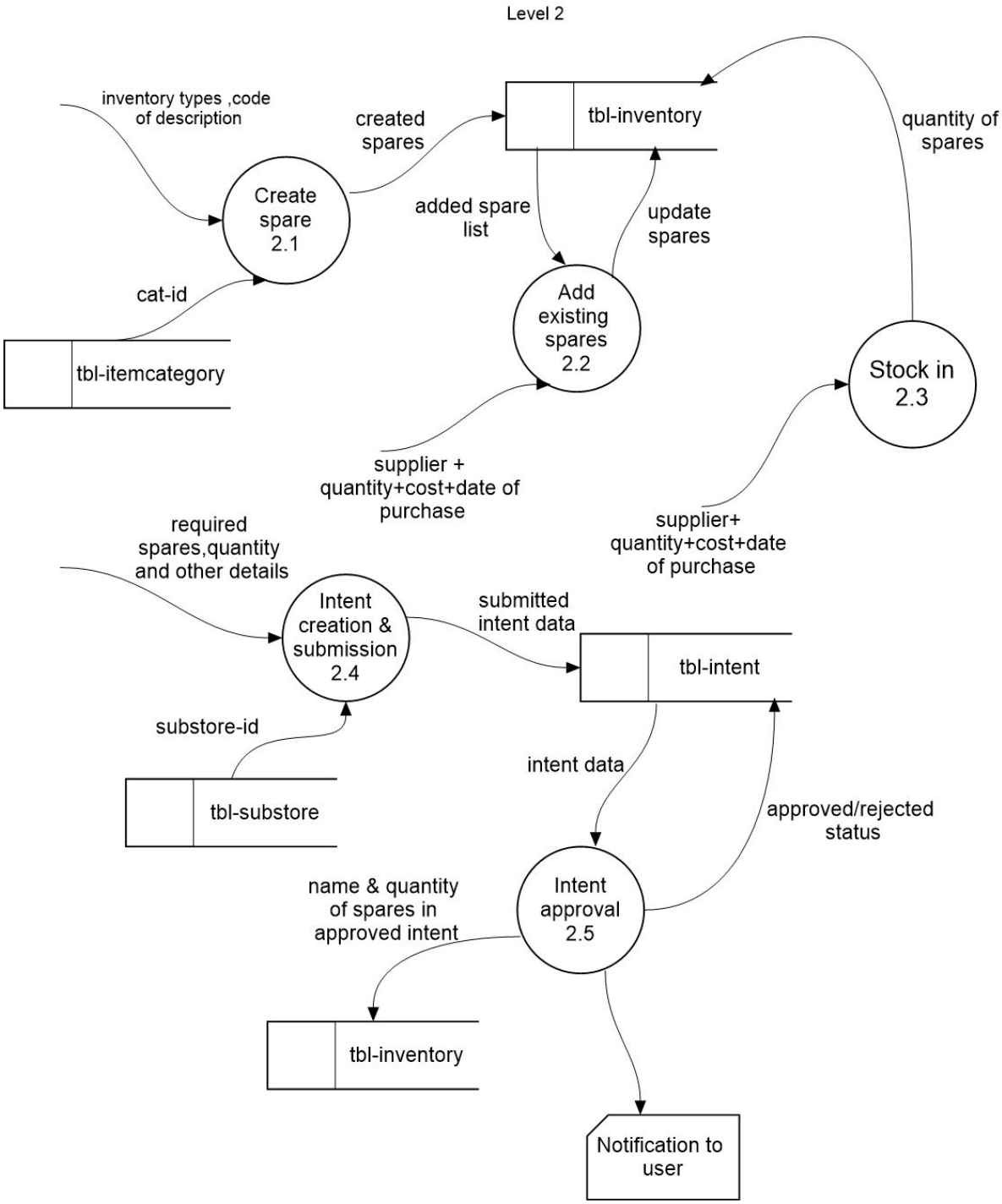
LEVEL 0



LEVEL 1



LEVEL 2



3.5. SYSTEM DESIGN

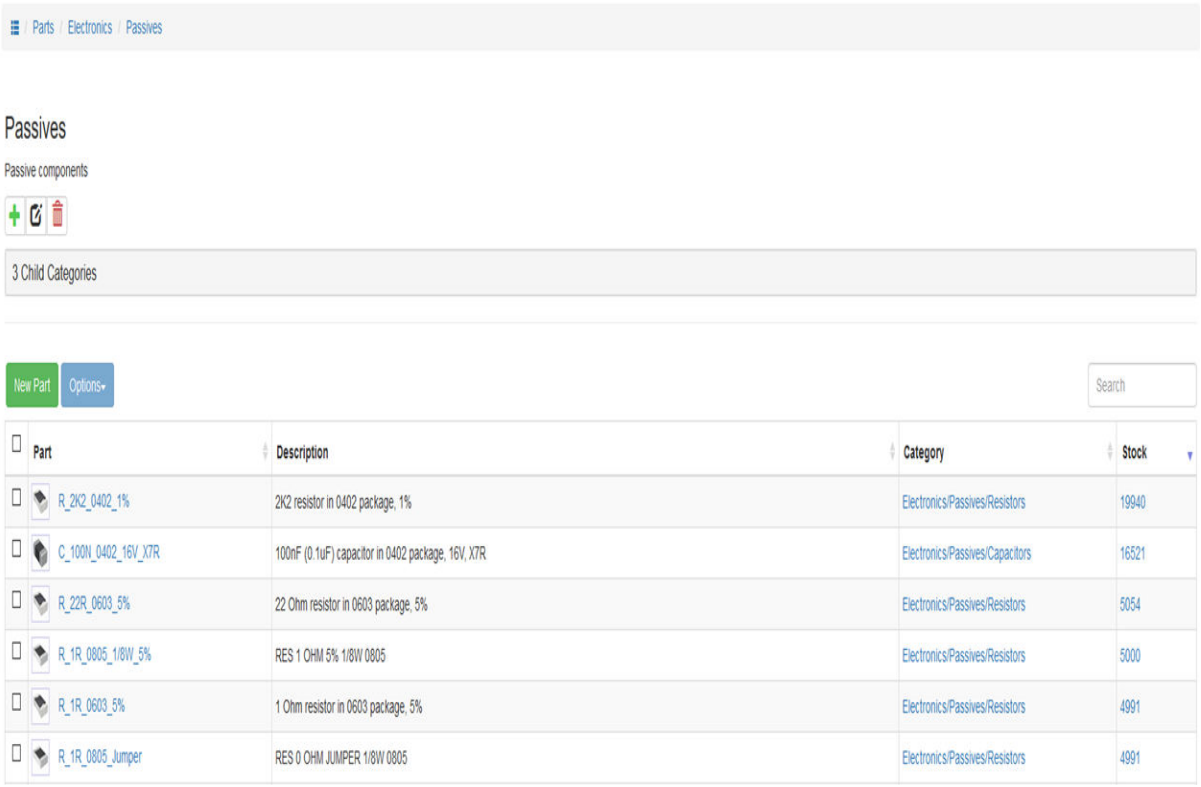
Part

The Part is the core element of the InvenTree ecosystem. A Part object is the archetype of any stock item in your inventory. Parts are arranged in heirarchical categories which are used to organise and filter parts by function.Part categories are very flexible and can be easily arranged to match a particular user requirement. Each part category displays a list of all parts *under* that given category. This means that any part belonging to a particular category, or belonging to a sub-category, will be displayed.

Each part category also shows a list of sub-categories which exist underneath it.

The category part list provides an overview of each part:

- Part name and description
- Part image thumbnail
- Part category
- Part stock level



Stock

Stock Location¶

A stock locations represents a physical real-world location where *Stock Items* are stored. Locations are arranged in a cascading manner and each location may contain multiple sub-locations, or stock, or both.

Stock Item¶

A *Stock Item* is an actual instance of a *Part* item. It represents a physical quantity of the *Part* in a specific location.

Stock Item Details¶

The *Stock Item* detail view shows information regarding the particular stock item:

Part - Which *Part* this stock item is an instance of

Location - Where is this stock item located?

Quantity - How many items are in stock?

Supplier - If this part was purcahsed from a *Supplier*, which *Supplier* did it come from?

Supplier Part - Link to the particular *Supplier Part*, if appropriate.

Last Updated - Date that the stock quantity was last updated

Last Stocktake - Date of most recent stocktake (count) of this item

Status - Status of this stock item

Stock Tracking¶

Every time a *Stock Item* is adjusted, a *Stock Tracking* entry is automatically created. This ensures a complete history of the *Stock Item* is maintained as long as the item is in the system.

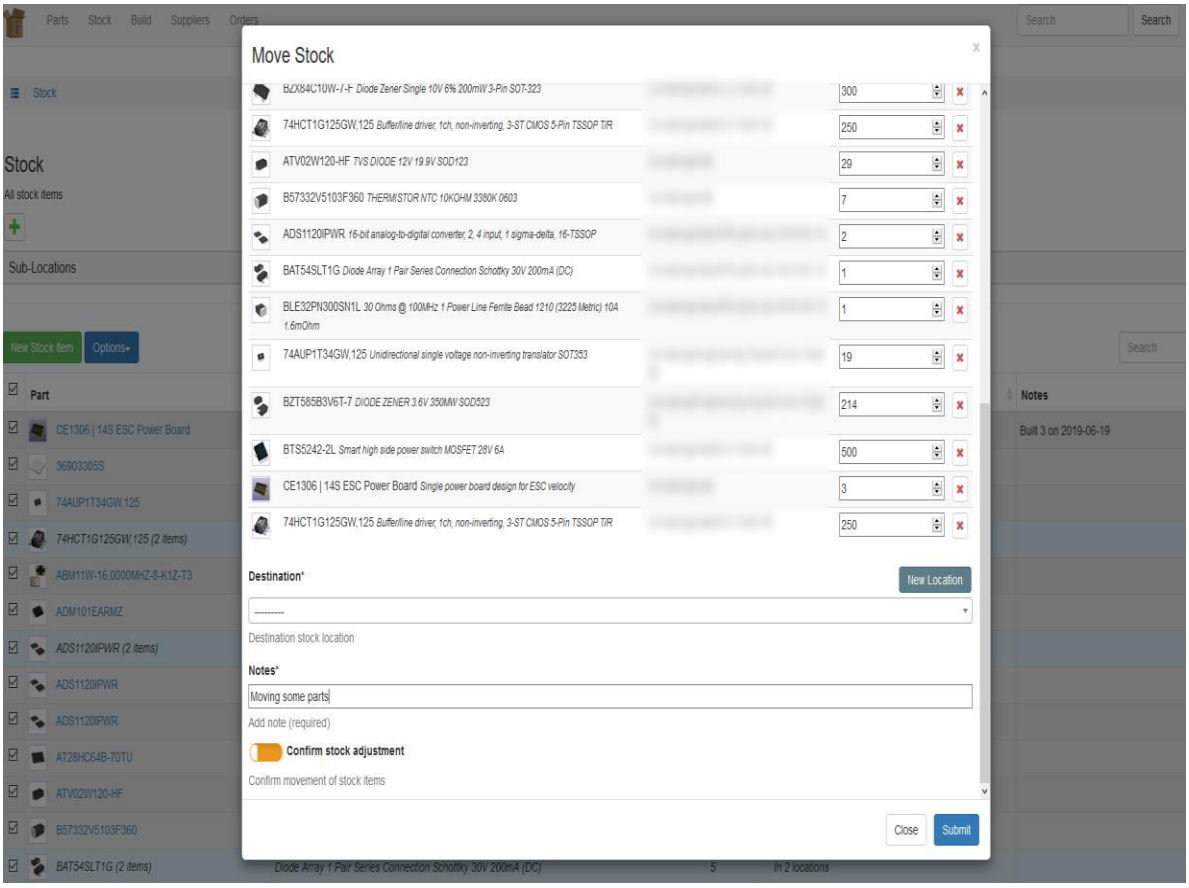
Each stock tracking historical item records the user who performed the action.

Stock Adjustments¶

Inventory provides simple yet powerful management of stock levels. Multiple stock adjustment options are available, and each type of adjustment is automatically tracked to maintain a complete stock history.

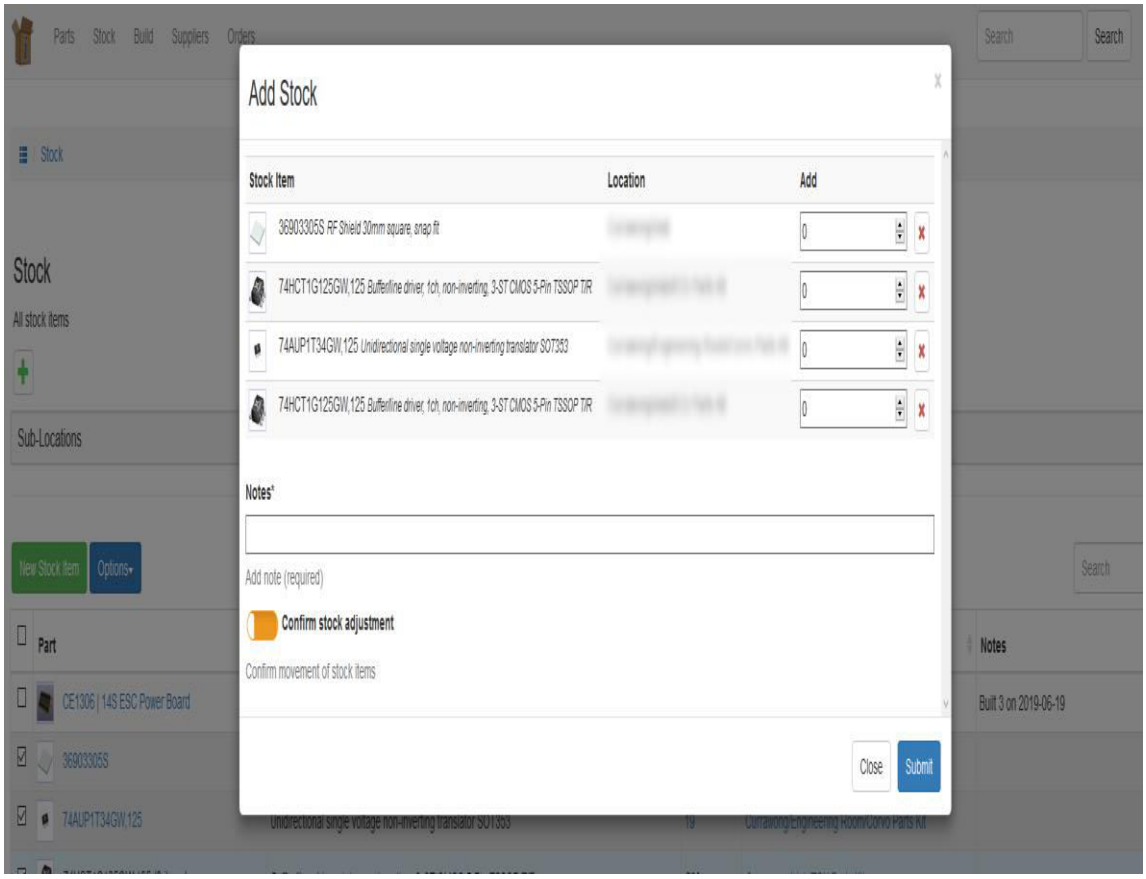
Move Stock¶

Multiple stock items can be moved to a new location in a single operation. Each item is moved to the selected location, and a stock tracking entry is added to the stock item history.



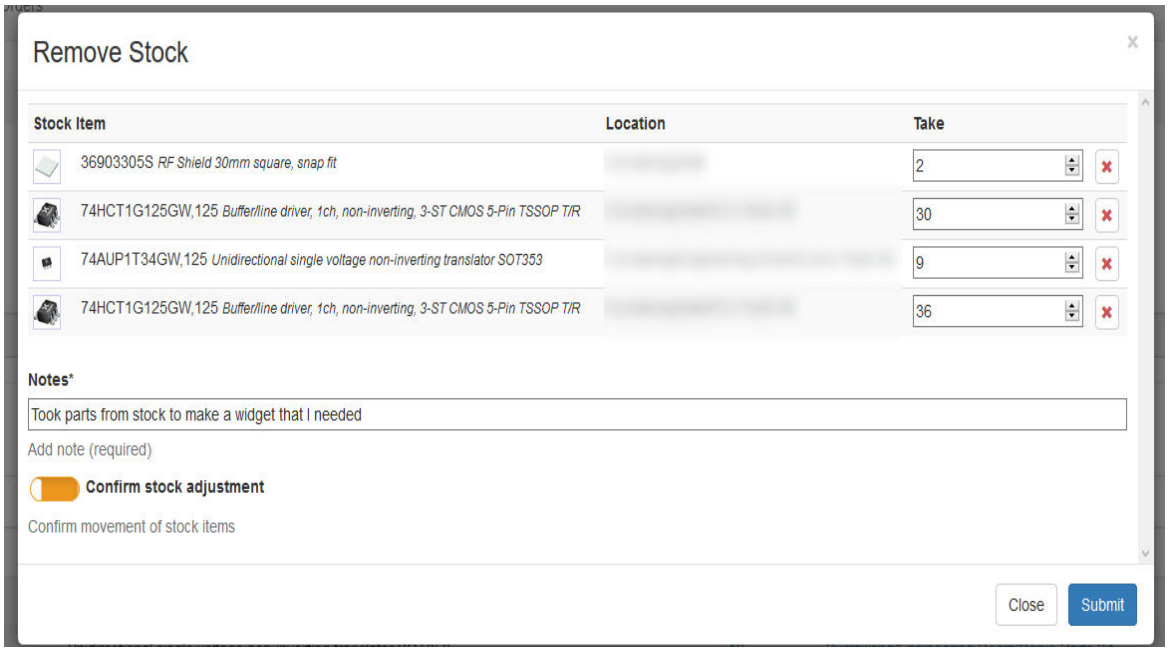
Add Stock¶

Add parts to a stock item record - for example putting parts back into stock. The in-stock quantity for each selected item is increased by the given amount.



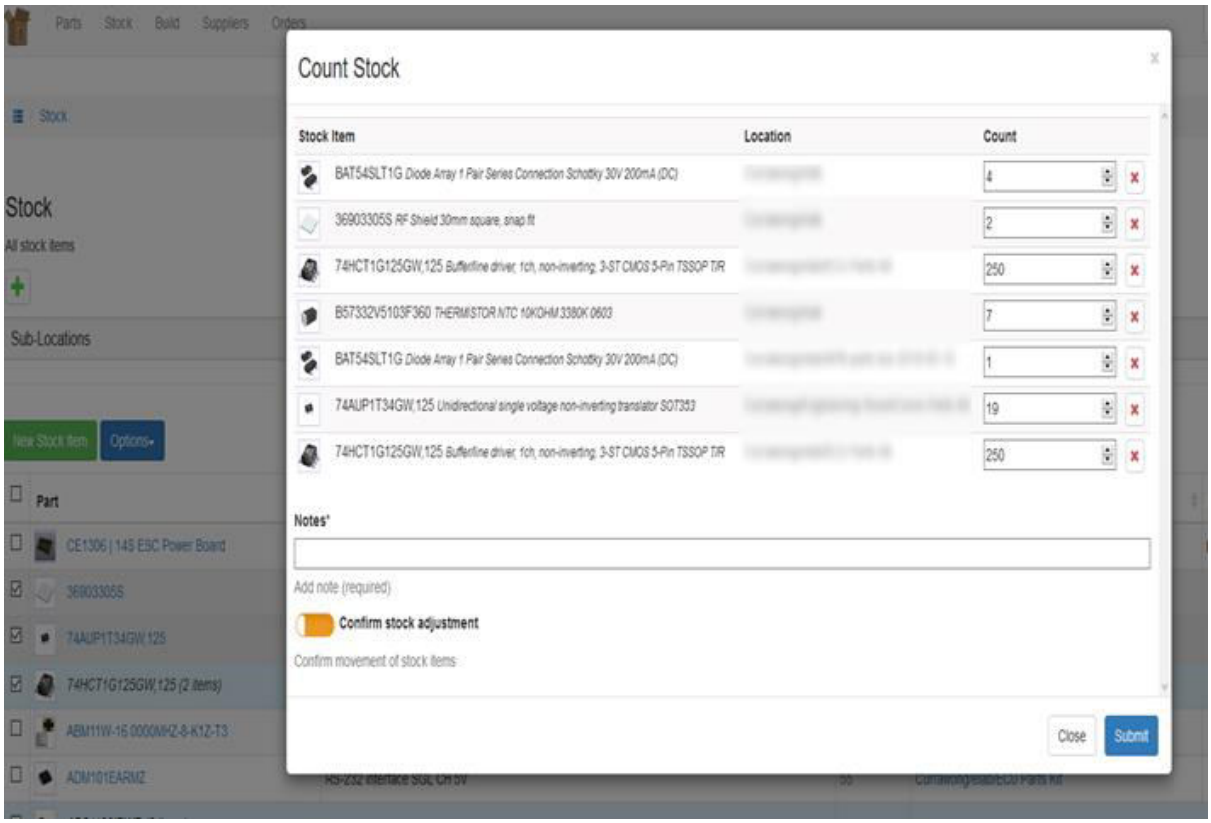
Remove Stock¶

Revove parts from a stock item record - for example taking parts from stock for use. The in-stock quantity for each selected item is decreased by the given amount.



Count Stock¶

Count stock items (stocktake) to record the number of items in stock at a given point of time. The quantity for each part is pre-filled with the current quantity based on stock item history.



3.6 DATABASE DESIGN

1. TABLE NAME: *tbl_login*

Description: Login details

Primary Key: u_usname

FIELD NAME	TYPE	REMARKS	DESCRIPTION
u_usname	varchar(50)	Primary key	Username of user/admin
u_pswd	varchar(150)	Not null	Password of user/admin
u_type	int(4)	Not null	Usertype
s_stid	int(3)	Not null	Sub Store ID

2. TABLE NAME: *tbl_substore*

Description: Sub store details

Primary Key: s_stid

FIELD NAME	TYPE	REMARKS	DESCRIPTION
s_stid	int(3)	Primary key	Sub store ID
s_subname	varchar(150)	Not null	Sub store name
s_addr	varchar(150)	Not null	Address
s_phone	BigInt(12)	Not null	Phone Number
s_head	varchar(20)	Not null	Head of sub store
s_loc	int(20)	Not null	Pin code

3. TABLE NAME*: tbl_itemcategory*

Description: Categories of spare items

Primary Key: itc_catid

FIELD NAME	TYPE	REMARKS	DESCRIPTION
itc_catid	int(10)	Primary key	Item category ID
itc_catname	varchar(25)	Not null	Item category name

4.TABLE NAME: *tbl_vehiclecategory*

Description: Categories of vehicle

Primary Key:v_catid

COLUMN NAME	TYPE	REMARKS	DESCRIPTION
v_catid	int(20)	Primary key	Vehicle category ID
v_catname	varchar(50)	Not null	Vehicle category name

5.TABLE NAME: *tbl_vehiclemodel*

Description: Vehicle model

Primary Key: v_mid

COLUMN NAME	TYPE	REMARKS	DESCRIPTION
v_mid	int(20)	Primary Key	Vehicle model ID
v_model	varchar(25)	Not null	Vehicle model Name

6.TABLE NAME:*tbl_vehicle*

Description: Details of vehicle

Primary Key: v_vno

COLUMN NAME	TYPE	REMARKS	DESCRIPTION
v_vno	varchar(30)	Primary Key	Vehicle number
v_catid	varchar(30)	Foreign Key	Vehicle category ID
v_mid	varchar(20)	Foreign Key	Vehicle model ID
v_ftype	varchar(30)	Not null	Vehicle fuel Type
v_chno	varchar(30)	Not null	Vehicle chasis number
v_engno	varchar(20)	Not null	Vehicle engine number
s_stid	int(3)	Foreign Key	Sub store ID

7.TABLE NAME: *tbl_inventory*

Description: Inventory details

Primary Key: in_itemid

COLUMN NAME	TYPE	REMARKS	DESCRIPTION
in_spname	varchar(150)	Not null	Spare parts name
in_itemid	varchar(10)	Primary Key	Spare item ID
itc_catid	int(3)	Foreign Key	Item category ID
in_descpn	varchar(150)	Not null	Item description
v_mid	varchar(10)	Foreign Key	Vehicle model ID
in_supply	varchar(150)	Not null	Supplier
in_qty	int(100)	Not null	Quantity
in_cost	float	Not null	Item cost
in_dop	date	Not null	Date of purchase

8.TABLE NAME: *tbl_intent*

Description: Intent data

Primary Key: int_id

COLUMN NAME	TYPE	REMARKS	DESCRIPTION
int_id	int(10)	Primary Key	Intent ID
int_itemid	int(25)	Foreign Key	Item ID
int_qty	int(15)	Not null	Quantity
v_vno	varchar(30)	Foreign Key	Vehicle number
int_status	int(2)	Not null	Status
s_stid	int(10)	Foreign Key	Store ID
int_dor	date	Not null	Date of request

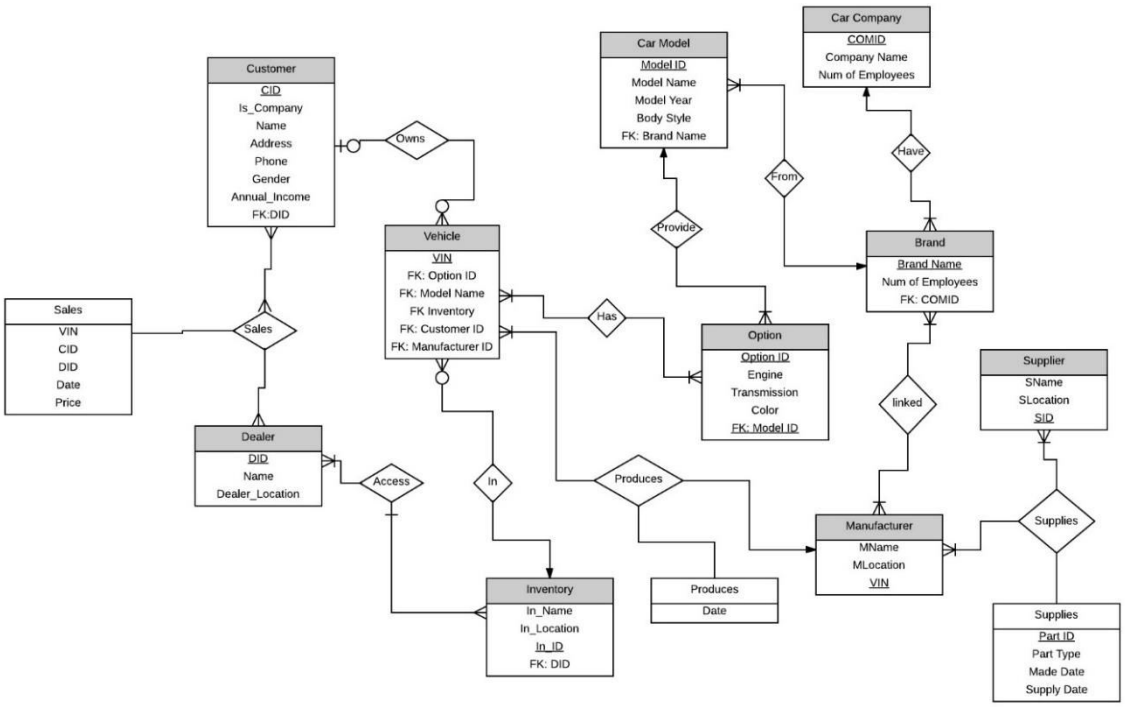
9.TABLE NAME: tbl_worksheet

Description: Worksheet to show inventory usage

Primary Key:w_id

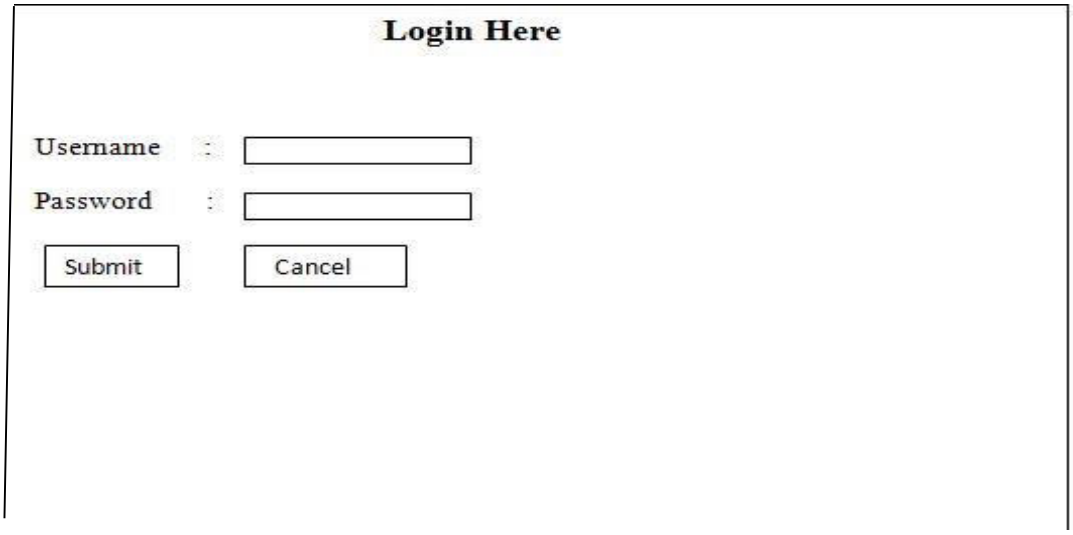
COLUMN NAME	TYPE	REMARKS	DESCRIPTION
w_id	int(4)	Primary key	Worksheet ID
s_stid	int(10)	Foreign Key	Store ID
v_vno	int(25)	Foreign Key	Vehicle number
int_dor	date	Not null	Date of request
int_itemid	int(10)	Foreign Key	Item ID
w_used	int(10)	Not null	Item used
w_damage	int(10)	Not null	Item damaged

3.7 E R DIAGRAM

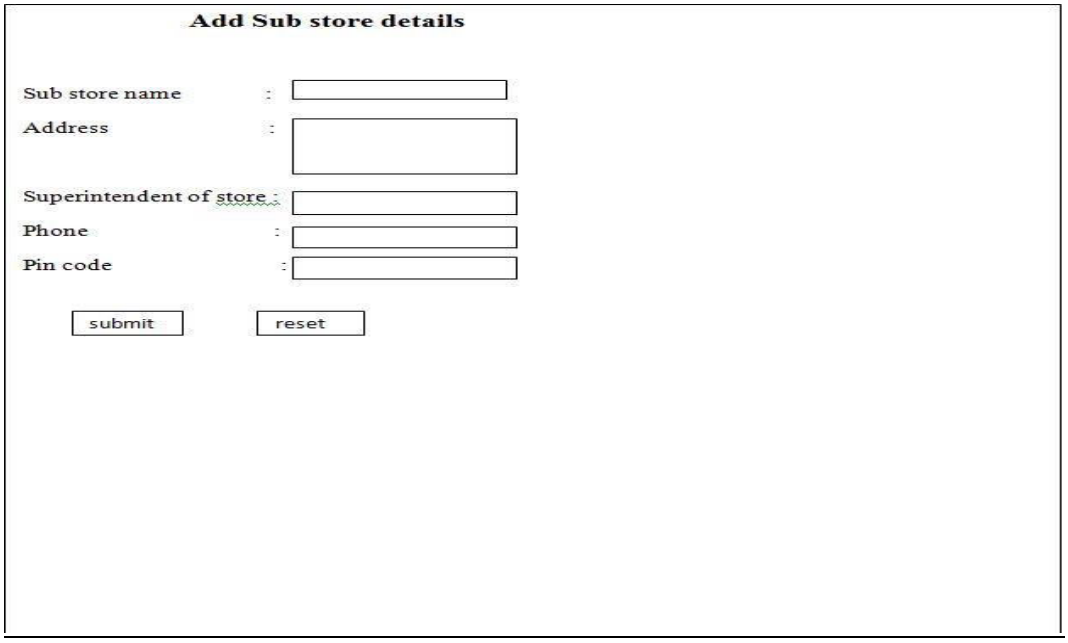


3.8 USER INTERFACE DESIGN

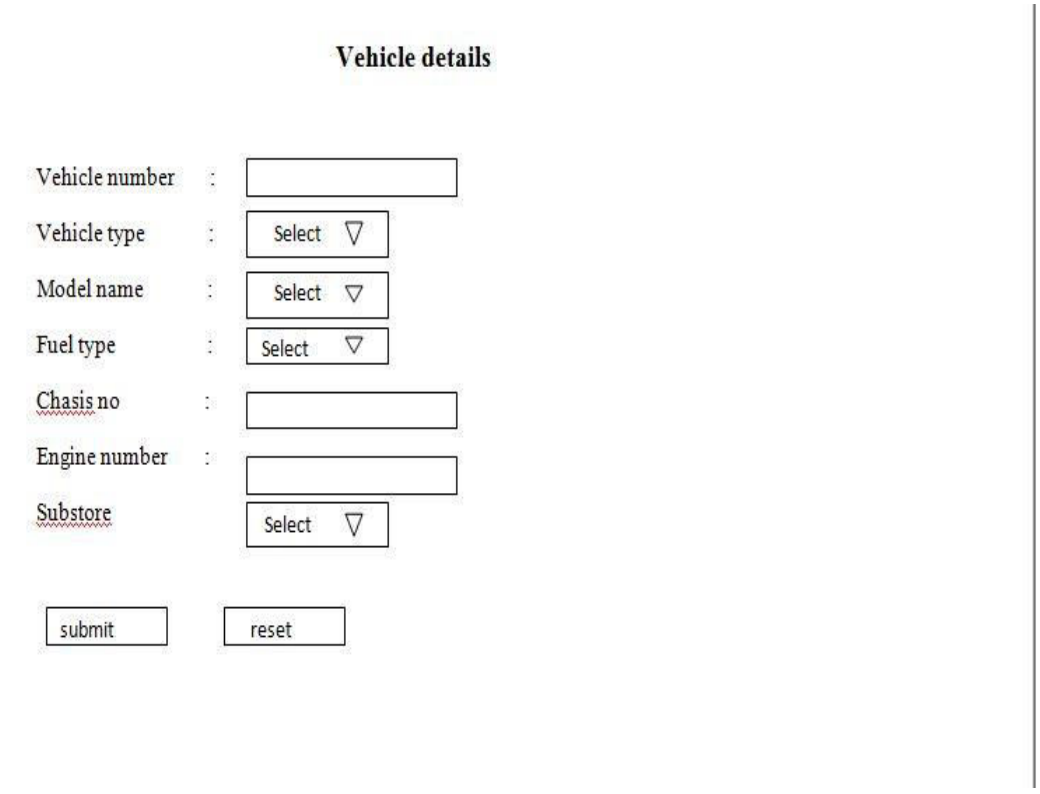
LOGIN



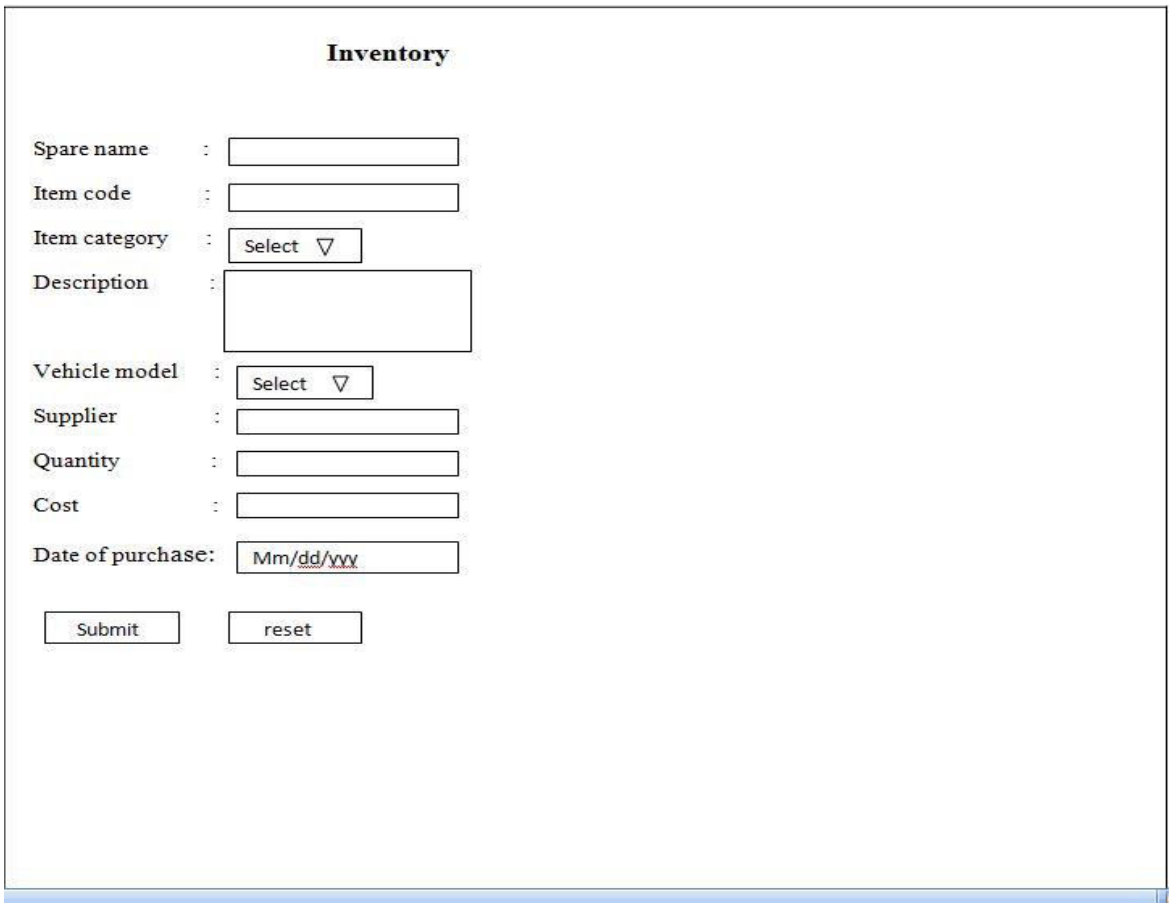
FOR ADDING SUB STORE DETAILS



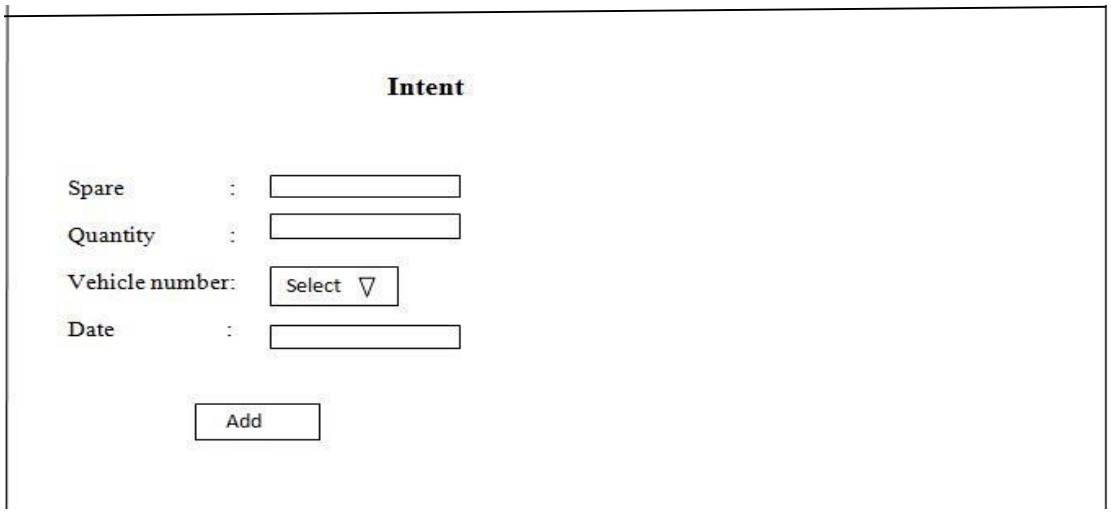
FOR ADDING VEHICLE DETAILS



FOR ADDING INVENTORY DETAILS



FOR ADDING INTENT DETAILS



4. DEVELOPMENT AND CODING

4.1 INTRODUCTION

A code is an ordered collection symbols to provide unique identification of data. Codes can be used by people who do not with data processing. The goal of the coding or programming phase is to translate the design of the system produced during the design phase into code in a given programming language, which can be executed by a computer and that performs the computation specified by the design. The coding phase affects both testing and maintenance profoundly. As we saw earlier, the time spent in coding is a small percentage of the total software cost, while testing and maintenance consume the major percentage. Thus it should be clear the goal should be reduced the cost of later phrases, even if it means that the cost of this phase has to increase. In other words, the goal during this phase is not to simplify the job of the programmer.

1. **Code Optimization**

Code optimization aims at improving execution efficiency of a program. This is achieved in two ways:

- Redundancies in a program are eliminated.
- Computations in a program are rearranged or rewritten to make it execute efficiently.

The optimization must not change the meaning of a program. The "Online Student Data Manipulation" optimizes the code by using the optimization techniques such as dead code elimination and frequency reduction. Thus improves the execution efficiency.

2. **Validation**

Validation means observing the behavior of the system. The verification and validation means that will ensure that the output of a phase is consistent with its input and that the output of the phase is consistent with the overall requirements of the system. The college admission manager performed validation by verifying the output of each phase. This is done to ensure that it is consistent with the required output. If not we apply certain mechanisms for repairing and thereby achieved the requirement.

4.2 DEVLOPMENT TOOLS

Python

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

The Python interpreter and the extensive standard library are freely available in source or binary form for all major platforms from the Python Web site, https://www.python.org/, and may be freely distributed. The same site also contains distributions of and pointers to many free third party Python modules, programs and tools, and additional documentation.

The Python interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable from C). Python is also suitable as an extension language for customizable applications.

Python is a general purpose and high level programming language. You can use Python for developing desktop GUI applications, websites and web applications. The simple syntax rules of the programming language further makes it easier for you to keep the code base readable and application maintainable. There are also a number of reasons why one should prefer Python to other programming languages.

1) Readable and Maintainable Code

While writing a software application, you must focus on the quality of its source code to simplify maintenance and updates. The syntax rules of Python allow you to express concepts without writing additional code. At the same time, Python, unlike other programming languages, emphasizes on code readability, and allows you to use English keywords instead of punctuations. Hence, you can use Python to build custom applications without writing additional code. The readable and clean code base will help you to maintain and update the software without putting extra time and effort.

2) Multiple Programming Paradigms

Like other modern programming languages, Python also supports several programming paradigm. It supports object oriented and structured programming fully. Also, its language

features support various concepts in functional and aspect-oriented programming. At the same time, Python also features a dynamic type system and automatic memory management. The programming paradigms and language features help you to use Python for developing large and complex software applications.

3) Compatible with Major Platforms and Systems

At present, Python is supports many operating systems. You can even use Python interpreters to run the code on specific platforms and tools. Also, Python is an interpreted programming language. It allows you to you to run the same code on multiple platforms without recompilation. Hence, you are not required to recompile the code after making any alteration. You can run the modified application code without recompiling and check the impact of changes made to the code immediately. The feature makes it easier for you to make changes to the code without increasing development time.

4) Robust Standard Library

Its large and robust standard library makes Python score over other programming languages. The standard library allows you to choose from a wide range of modules according to your precise needs. Each module further enables you to add functionality to the Python application without writing additional code. For instance, while writing a web application in Python, you can use specific modules to implement web services, perform string operations, manage operating system interface or work with internet protocols. You can even gather information about various modules by browsing through the Python Standard Library documentation.

5) Many Open Source Frameworks and Tools

As an open source programming language, Python helps you to curtail software development cost significantly. You can even use several open source Python frameworks, libraries and development tools to curtail development time without increasing development cost. You even have option to choose from a wide range of open source Python frameworks and development tools according to your precise needs. For instance, you can simplify and speedup web application development by using robust Python web frameworks like Django, Flask, Pyramid, Bottle and Cherrypy. Likewise, you can accelerate desktop GUI application

development using Python GUI frameworks and toolkits like PyQT, PyJs, PyGUI, Kivy, PyGTK and WxPython.

6) Adopt Test Driven Development

You can use Python to create prototype of the software application rapidly. Also, you can build the software application directly from the prototype simply by refactoring the Python code. Python even makes it easier for you to perform coding and testing simultaneously by adopting test driven development (TDD) approach. You can easily write the required tests before writing code and use the tests to assess the application code continuously. The tests can also be used for checking if the application meets predefined requirements based on its source code.

7) Error Handling

When an error occurs, the interpreter prints an error message and a stack trace. In interactive mode, it then returns to the primary prompt; when input came from a file, it exits with a nonzero exit status after printing the stack trace. (Exceptions handled by an except clause in a try statement are not errors in this context.) Some errors are unconditionally fatal and cause an exit with a nonzero exit; this applies to internal inconsistencies and some cases of running out of memory. All error messages are written to the standard error stream; normal output from executed commands is written to standard output.

Typing the interrupt character (usually Control-C or Delete) to the primary or secondary prompt cancels the input and returns to the primary prompt. 1 Typing an interrupt while a command is executing raises the KeyboardInterrupt exception, which may be handled by a try statement.

8) Numerous libraries and frameworks

A huge advantage of Python is the wide selection of libraries and frameworks it offers. Your time-to-market will improve if you leverage them, since you won't be coding features manually.

There's a Python library for everything:

- data visualization,

- machine learning,
- data science,
- natural language processing,
- complex data analysis.

There's a variety of frameworks to choose from, depending on our needs, such as:

- Django,
- Flask,
- Pyramid,
- Twisted,
- Falcon.

9) Performance

One of the biggest criticisms of Python is the runtime, relatively slow when compared to other languages. However, there's a workaround to this specific challenge.

When performance takes priority, Python gives you the ability to integrate other, higher-performing languages into your code. Cython is a good example of such a solution. It optimizes your speed without forcing you to rewrite your entire code base from scratch.

Besides, the priciest resource isn't CPU time; it's your developers' time. Reducing your time-to-market should therefore always take precedence over fast runtime execution.

10) Easy maintenance

Python is intuitive to read, because it resembles actual English. This makes the language effortless to decipher and maintain.

Additionally, Python has a clear syntax and doesn't require as many lines of code as Java or C to give you comparable results.

11) Reliable scalability

Scalability is unpredictable. You never know when your user numbers surge and you find yourself prioritizing the ability to scale over anything else.

That's why Python is such an optimal choice, with its reliability and scalability. Some of the biggest players on the web, like YouTube, have bet on Python for that very reason.

12) Speed

Building an MVP with Java can take months because of its high code complexity and volume. Consequently, projects written in Java often go on for years and demand more developers on the team.

Python doesn't have any of these problems, thanks to its lightning-fast development speed. You can build an MVP with Python in mere weeks, finish the whole project in a matter of months, and use only a handful of developers for the job.

Beating deadlines is Python's specialty.

13) Resources

Development in Java is a bigger investment all around; it requires more time and money. If you have a lot of those on your hands, you should be perfectly satisfied with Java.

Python is less expensive, which is why for most projects it's the preferred choice. Remember, just because something costs more doesn't automatically make it better.

Trending technologies

No programming language is better suited for trending technologies than Python. Whether it's artificial intelligence or machine learning, Python's design and features give it an advantage over all other languages for these relatively new purposes. The main reason why Python's been adopted as the go-to language for trending technologies is its extensive AI/ML library support.

Advantages of Python over PHP

1. **Versatility**

From data analytics through machine learning models to powerful web apps, there is little that Python can't do. In terms of versatility, it beats PHP hands down.

2. **Structure**

Since there have been fewer releases of Python than PHP, it tends to be more organized, secure, and easier to maintain.

3. **Popularity**

Given its application in areas such as artificial intelligence, machine learning, and the Internet of Things, as well as a vast array of uses that remain beyond the scope of PHP, Python has enjoyed a huge popularity spike in recent years.

Even though PHP has traditionally been more popular than Python, it's been gradually losing traction of late.

Advantages of Php over Python

1. **Features**

PHP comes with more out-of-the-box features than Python. The latter does, however, make use of plenty of libraries to make up for that slight inconvenience.

2. **Ease of installation**

PHP is easier to install on any platform than Python, unless you stick exclusively to Linux.

Is Python or PHP better for your software project?

Even though both Python and PHP are mature, well-established languages that enjoy widespread use and support, they couldn't be more different from each other when it comes to their syntax and philosophy.

However, when faced with a choice between the two, the decision should always come down to the individual requirements of your project.

If you're after a website, a blog, or a simple web service, the end result will most likely be the same whether you go with Python, PHP, or any other leading web development technology. You shouldn't be able to notice any difference in terms of performance, speed, or user design.

However, if the scope of your project is more varied and includes, for instance, machine learning, data analytics, or the Internet of Things, you should pick Python.

Django

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source.

Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes reusability and "pluggability" of components, less code, low coupling, rapid development, and the principle of don't repeat yourself.[7] Python is used throughout, even for settings files and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models.

Ridiculously fast.

Django was designed to help developers take applications from concept to completion as quickly as possible.

Reassuringly secure.

Django takes security seriously and helps developers avoid many common security mistakes.

Exceedingly scalable.

Some of the busiest sites on the Web leverage Django’s ability to quickly and flexibly scale.

The core Django framework can be seen as an MVC architecture.[6] It consists of an object-relational mapper (ORM) that mediates between data models (defined as Python classes) and a relational database ("Model"), a system for processing HTTP requests with a web templating system ("View"), and a regular-expression-based URL dispatcher ("Controller").

Also included in the core framework are:

- a lightweight and standalone web server for development and testing
- a form serialization and validation system that can translate between HTML forms and values suitable for storage in the database
- a template system that utilizes the concept of inheritance borrowed from object-oriented programming
- a caching framework that can use any of several cache methods
- support for middleware classes that can intervene at various stages of request processing and carry out custom functions
- an internal dispatcher system that allows components of an application to communicate events to each other via pre-defined signals
- an internationalization system, including translations of Django's own components into a variety of languages
- a serialization system that can produce and read XML and/or JSON representations of Django model instances
- a system for extending the capabilities of the template engine
- an interface to Python's built-in unit test framework
- Django REST framework is a powerful and flexible toolkit for building Web APIs.

Bundled applications[edit]

The main Django distribution also bundles a number of applications in its "contrib" package, including:

- an extensible authentication system
- the dynamic administrative interface
- tools for generating RSS and Atom syndication feeds
- a "Sites" framework that allows one Django installation to run multiple websites, each with their own content and applications

- tools for generating Google Sitemaps
- built-in mitigation for cross-site request forgery, cross-site scripting, SQL injection, password cracking and other typical web attacks, most of them turned on by default[18][19]
- a framework for creating GIS applications

MySQL:

Relational database systems are the most important database system used in the software industry today. One of the most outstanding systems is MySQL. MySQL is a database management system developed and marketed by Microsoft. The most important aspects of MySQL are,

- MySQL is easy to use.
- MySQL scales from a mobile laptop to symmetric multiprocessor system.
- MySQL provides data warehousing features that until now have only been available in Oracle and other more expensive DBMS.

MySQL uses services of Linux or Windows to offer new or extended database capabilities, such as sending and receiving message and managing login security. The MySQL administrator's primary tool for interacting with the system is Enterprise Manager. The Enterprise Manager has two main purposes: Administration of the database server and Management of database objects. MySQL Query Analyzer provides a graphical presentation of the execution plan of a query and an automatic component that suggests which index should be used for a selected query. This interactive component of MySQL performs the task like:

- Generating and executing Transact-SQL statements.
- Storing the generated Transact-SQL statements in the file.
- Analyzing execution plans for generated queries.

Graphically illustrating the execution plan for a selected query.

5. TESTING AND IMPLEMENTATION

5.1 INTRODUCTION

Software testing is critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing begins by testing program modules separately, followed by testing "bundled" modules as a unit. A program module may function perfectly in isolation but fail when interfaced with successively larger up to the system test level. The following methods of testing were carried out to assure the correctness and reliability.

UNIT TESTING

All during the system design activity, basic program module are tested. At this stage program modules are tested .At this stage programmers usually makeup their own data. Unit testing with test data is necessary, of course, but it is not sufficient. Although it is important know if the logic included in a program works properly, conditions that are not included in the program are also considered. In this mainly syntax and logical errors of the programs are tested.

INTEGRATED TESTING

As modules pass unit test, they are integrated for testing. Programs are invariability related to one another and interact in a total system. Each program is tested to see whether it confirms to related programs in the system. Each portion of the system is tested against the entire module with both test data and live data before the entire system is ready to be tested.

VALIDATION TESTING

Validation succeeds when the software function in a manner the user wishes. Validation refers to the process of using software to live environment in order to find errors. During the course of validation system failure may occur and sometimes coding has to be changed according to the requirement. Thus the feedback from the validation phase generally produces changes in the software.

5.2 IMPEMENTATION

Implementation is one of the most important task in a project. Implementation is the face, in which one has to be cautious, because all the efforts under taken during this project will be fruitful only if the software is properly implemented according to the plans made.

Implementation is the stage in the project were theoretical design is turned in to a working system. The crucial stage is achieving successful new system and given to the users confidence in that system will work effectively and efficiently.

6. CONCLUSION

There are a bunch of websites which deal component sales and service online. But most of them are either confined to promote and sell the products of a particular company or brand, or selling our al types of products, not only parts. This project address this scenario. A customer can find and purchase parts of his choice at one place. If the product of one company is not available in his reach, he can move on to th next brand, not the next distance.

Utmost care has been taken to design the screens to make the customers' choices the easiest. Tables are designed to meet all requirements identified during system study. Besides, all the tables are normalized to reduce redundancy. The system addresses all issues identified during study, design development phases.

Though the primary methodology used in this project is the conventional software development life cycle (SDLC) method, attempts has been made to incorporate agile technology during design and development.

BIBLIOGRAPHY

Web Resources

1. https://www.digitalocean.com/community/tutorials/how-to-build-a-universal-application-with-nuxt-js-and-django

2. https://www.postgresql.org/

3. https://www.django-rest-framework.org/

Books

5. Russo et.al. “Agile technology in Open source Development”,IGI Global, 2016.

6. Pankaj Jelotte, “An integrated Approach to Software Engineering”, Narosa Publishing House, 2012.

APPENDIX

A1.SCREENSHOTS