

REPORT

General introduction to the agile and iterative methodologies.

“Agile – denoting “the quality of being agile, readiness for motion, nimbleness, activity, dexterity in motion. “
While each of the agile methodologies is different in itself and follow different practices, they all share a common idea and central values. These methods use the process of regular iteration with continuous feedback. They are all lightweight, as these development processes break a large phase of work into small iterations, not like the traditional waterfall-style processes. Agile methods focus on empowering people to work together and make appropriate decisions quickly and effectively.

Methodology 1

Extreme Programming

Extreme Programming development process is used where the information about the system to be developed is vague and incomplete and where the workforce of the teams are small to medium in size. Short iterations with frequent releases and rapid feedback help the system to grow in a stable way. Customer participation, communication and coordination help in guiding the changes in the project. Continuous integration and testing help in seeing the project as a whole. Collective ownership of the code, limited documentation and pair programming help in reducing the error that is generated. These and other are among the main characteristics of XP.

As the project is quite dependent of the owner of the product and brokers so the system should be both flexible and easy to understand and with the lowest amount to of documentation to go through in the shortest amount of time which can be achieved by Extreme programming.

As the requirements of the brokers might change along the line of the development of the system of the development should be done in tight iteration this is an integral part of extreme programming.

Advantages

1. It is iterative so specification can be changed. This ensures that the client gets what he wants even if the needs are changed over the period of time.
2. The developer can change the code in the middle of an iteration.
3. It incorporates pair programming.

Disadvantages

1. The task is performed in a strict priority order this order cannot be changed during an iteration.
2. It is focused on code rather than a design which is an important part of software projects
3. It does not measure code quality assurance.
4. XP will not work properly be the programmers are distributed geographically.

Methodology 2

Scrum

Scrum is a software development model used to manage complex software and product development. The iteration is of fixed size and fixed duration and are known as Sprints. Each sprint lasts from two to three weeks long, allow the team to release a newer version of the software product on a regular cadence. After every Sprint all stockholders and development team met in order to discuss the next iteration. It does not define any specific software development techniques for the implementation of a software. Scrum concentrates on how the team members should function in order to produce the system flexible in a constantly changing environment.

As it is mentioned OZES is a young business and there are not many employees this kind of development is good for small projects. Also, the initial plans seem clear from the case that was presented so it will be easy to start the work for the first few iterations. The priorities and the tasks are defined so it will be easy and efficient to apply this method.

Advantages

1. Short sprints and constant feedback, it's easier to cope with and apply changes.
2. Constant communication help in keeping the team aware of all the issues. by coding and testing the code in smaller chunks we can get continuous feedback of the system working.

Disadvantages

1. If there is no end date specified it can cause the risk of scope creep. Stakeholders may be tempted to keep requesting additional functionality.
2. You cannot make changes in the task in between an ongoing sprint.
3. If the initial goals are unclear, planning becomes difficult and sprints can take more time than originally estimated.

Methodology 3

Feature Driven Development

The project is divided into features. These features are small pieces of a complete project. Now all the features are listed in a list of backlog and are sorted on the basis of priority. After these, each feature is developed individually and then integrated into the project. With the help of FDD, you can create design, code, and code inspection schedules without going into elaborate paperwork. The focus is more on relying on people and their roles for development.

As the different features are well defined as the creation of the information system, sale item system and customer relationship it is easy to gather specification and develop a feature in the system. This kind of work can be handled by the feature Driven approach.

Advantages

1. Costing is easy to analysis as it depends on the number of features.
2. Time allocated can be managed on the basics on the number of features added or deleted and the end date can set to accuracy.
3. Good for long-term projects.

Disadvantages

1. If the high-level scope of the system is not clear this system will fail.
2. No written documentation, so there are no records or any proves.
3. There is a lot of dependencies that lie on the head programmer as he has to do all the work like coordinating lead designing and to lead the team.

Methodology 4

Kanban

In Kanban, the main emphasis is on continual delivery while not overstraining the development team. In Kanban system follow the simple three rules. Firstly, Visualize what to do today i.e. in this model we develop a plan for very short iteration and start the work. Limit the amount of work in progress. To progress flow i.e. if some work is completed the next higher thing in the backlog is taken up.

Kanban promotes continuous collaboration and encourages active, ongoing learning and improving by defining the best possible team workflow.

As the OZES is a new company even if it has a predefined requirement but these can change and if it change's it can be easily incorporated in this development process. But this has a downside as it can overcomplicate things and may increase the time taken to do work. Or may also affect the whole project time.

Advantages

1. Rapid feedback loops improve the changes of less amount on error and higher-performance of the team as a whole.
2. The iteration is small of features are delivered faster
3. Changes can be made at any time in the development cycle
4. Reduces waste as the project requirements are always up to date so the team does not make anything that is not required.

Disadvantages

1. No roles are prescribed. Of if something's goes wrong there is less accountability on a specific person.
2. Less effective in shared resource systems
3. The quality of the product is low.
4. Due to freeness on the changes the team can complicate the task at hand.

Recommendation

Form my study of your companies' case and after understanding your need for the case study. I would like to recommend the Feature-based approach to development for the project because of the following reasons:

1. The features of each of the different system are well defined
-For example, in Sale Item System you need a feature to record data of buyers, owner, items, and broker.
2. As you are in the business of electrical parts for a while so there will be no drastic change in requirement.
3. It will be good to integrate a feature and test as all the system are mostly work independently of each other. Even though they use data from each other but for testing purposes on trial basics it can be independent.

4. As people are not used to changes to it will be easy to change slowly with one feature at a time rather than a new system as a whole like in waterfall model.
5. Costing is easy to analysis as it depends on the number of features.
6. Time allocated can be managed on the basics on the number of features added or deleted and the end date can set to accuracy.
7. Good for long-term projects.

MEMO

To: OZES Management team

From: Company you hired

Date: May 12, 2018

Subject: Importance of UCD (User Centric Design) in the new System.

Nowadays as most of the user of the system are well aware and educated to deal with human-computer interaction. Also looking down the road we can see a world that is flat due to the advancement in the technology.

In User-centric design, the main focus is on building a design that is able to fulfill users need. As each user will interact with the system for a different purpose and will use a different feature every time. But this requires the involvement of the user during the whole process of designing the system. As at the end it also give rise to a system that is well equipped to deal with the real –time user problem.

Ways in which UCD pays off:

1. If the user involvement is more the system developed in more user-friendly, this in term increases the changes of the product being sold to a greater extent.
2. Systems designers with help of user interaction with the system try to find the loopholes in the system which help them to see the system in a user context.
3. Putting designers with a user the system designed is keeping the user feel in the system and which help in keeping the privacy of the user and the quality of life intact.

4. By working with the different diversity of user the designer has to deal with a lot of problems and to work out ways in other to communicate to them all making the product an all-around success.
5. Engagement with users gives a better picture of the requirement analysis on the system.
6. Even if the user interaction increasing a cost a bit it takes this cost away by helping the designer a more refined design to work on which intern decrease the cost of redesigning the system in the future.

Customer Relationship Management or in short, CRM helps companies understand, as well as to develop approaches to deal with the needs of the current and future customer. However, studies show that 50% of the CRM system implementation fail. This situation is due to the following reasons no-ease-of-use, zero understanding from business point-of-view, wrong interpretation of needs, and no customer involvement. In order to get a wider view of the current CRM implementation, we have to understand the basis of human interaction with computer systems.

HCI (human-computer interaction) is the study on how the interaction of a user with computer system and application on computer help a system developer to attain information to what extent the user can access the computer's features. A significant number of major corporations and academic institutions now study HCI in order to make the user experience better in order to increase their sales. One important HCI factor is that every individual has a different mental picture of the system when they access the system or interact with the system. Every person learns in a different way and a different style. For example, "left-brained" and "right-brained" people. The major reason for Studying the HCL is of the following reason

1. Cultural and national differences
2. Quality of studying
3. User environment
4. Uses behaviour

As we are in the process of making a Customer relationship management (CRM) system the first and the foremost concern is the way the customer approach the system. Every user of the system wants a design that is easy to understand and deal with so to understand how a common man thinks we have to go to a view of the system they are comfortable with and to do so we have to take the help of User Centric Design. On the other hand, due to globalization, advancement in technology, increase in computer-based education of the people and for increasing the amount of the customers we should focus on the human-computerized interaction. This is why the system should consist of three major parts first being a computerized system of interaction, second being the system itself the Customer-Relationship Management and engulfing both of these inside a User-Centric Design.

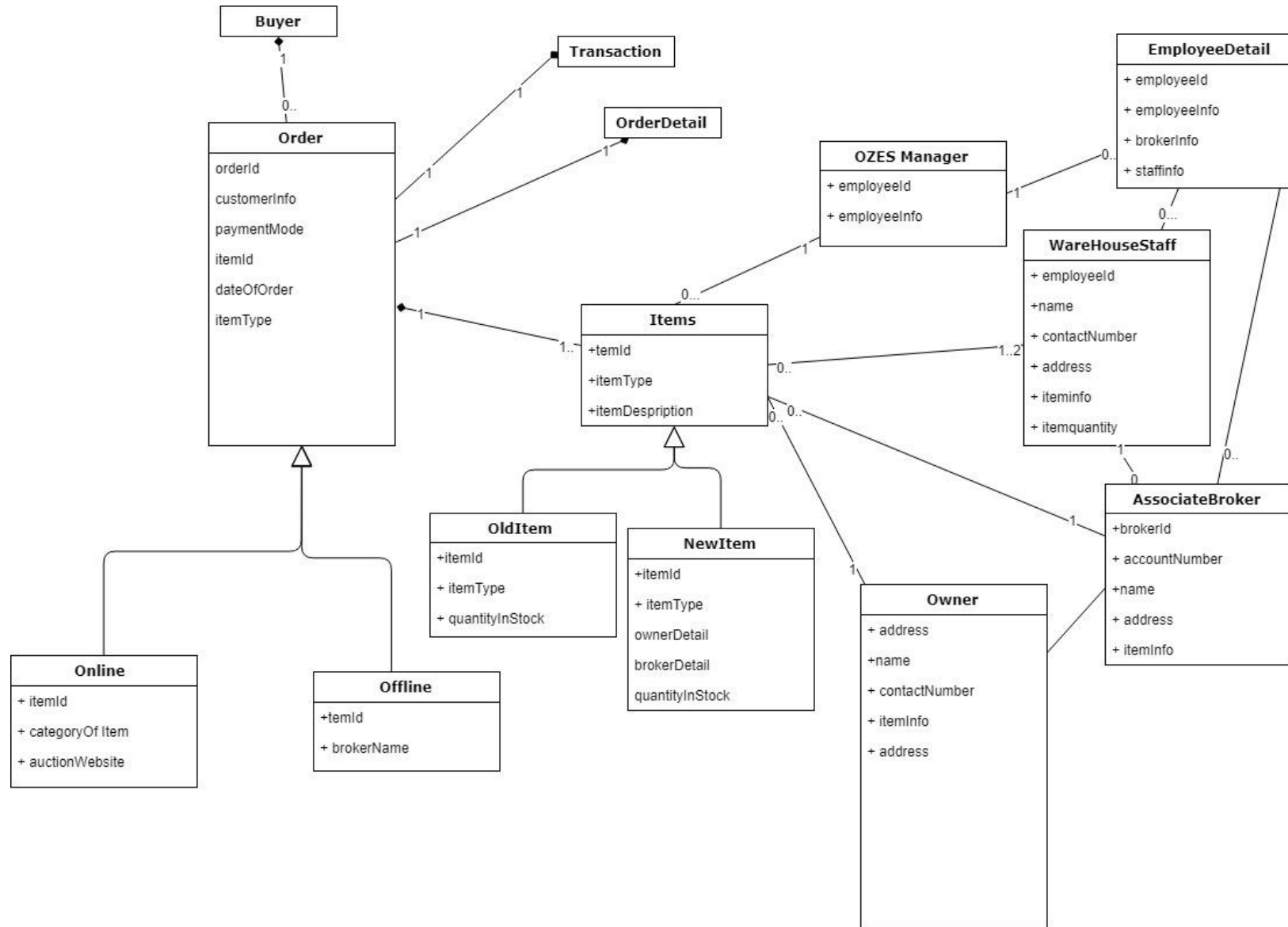
As we will be building an information system that will include a sales item System, a customer relationship system and warehouse management system and all of these systems has to interact with different categories of user which will increase the problem of diversity in age as well as mentality. As most of the broker will be middle to low class and the sellers or buyer of electrical equipment will be high class as these are expensive equipment. So, this will need a thorough analysis of user on all ground with the help of HCI. The system built should be flexible enough to work for all age group of people as it is concerning with electrical equipment that can be needed by all kinds of people. As the system has to go through different user the view of the system should be user-centric.

Event table

Event	Event Type	Trigger	Source	Use case /activity	Output	Destination
User click on the online catalog	External event	Inventory request	User	Viewing Products	List of all the products	User
Collecting Customer information	External event	Any order, selling Or inquiry	Brokers	Collecting information	name, address, email and phone contact details of customer	Information database
Reorder of inventory items	State Event	Inventory at low stock		Reorder inventory	Purchasing order	Supplier purchasing manager
Placing order	External event	Buy now option	User	order	Order details	User
Time to produce monthly sales report	Temporal event	"End of the month"		Monthly sales report	Sales Report	Manager
Find the position of placed order	External event	After placing the order	User	Track order	Position of the order	User

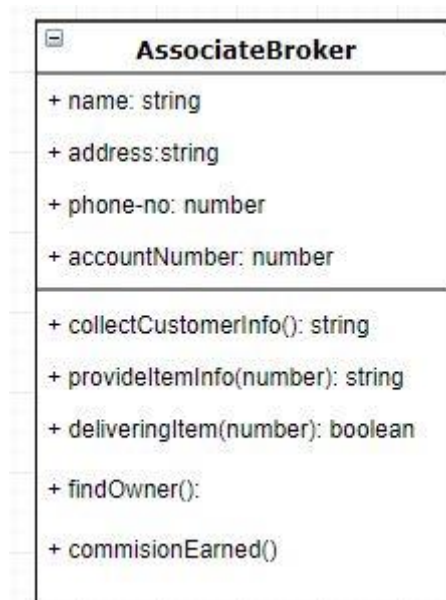
Information of broker	External event	Display broker information request	OZES employee	Employee information	List of brokers and there relevant information	OZES employee
Sending email of conformation	Temporal event	"After order is placed and on receiving order"		Sending emails	Information regarding the product the delivery date etc.	User
Buying second hand product from owner	External event	Request from the owner to sell product	Broker	Buying products Second hand	Product bought, owner information and item information	Broker
Sending the product bought from owner to OZES plant	External event	Broker drop the product at OZES plant	Broker	Receive the product at the inspection centre	Product delivered to OZES	Warehouse
Transfer money to brokers	Temporal event	"three days after the item is sold"		Payment to broker	Commission paid	Broker

Model Class Diagram

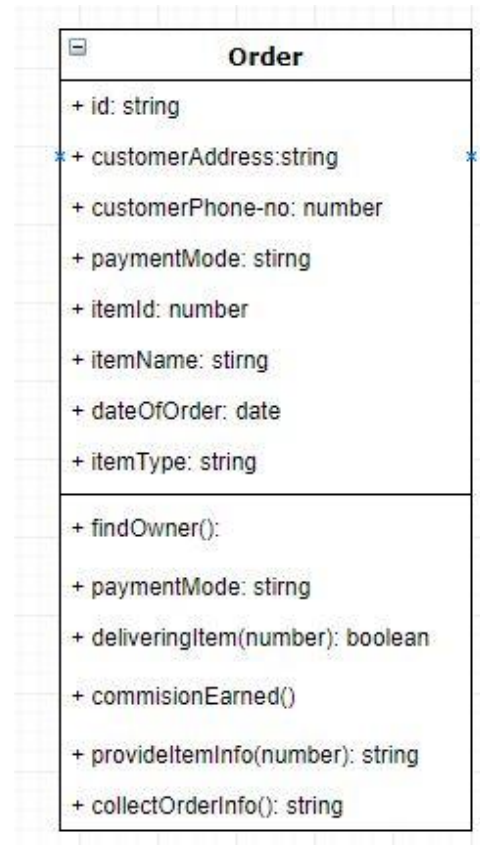


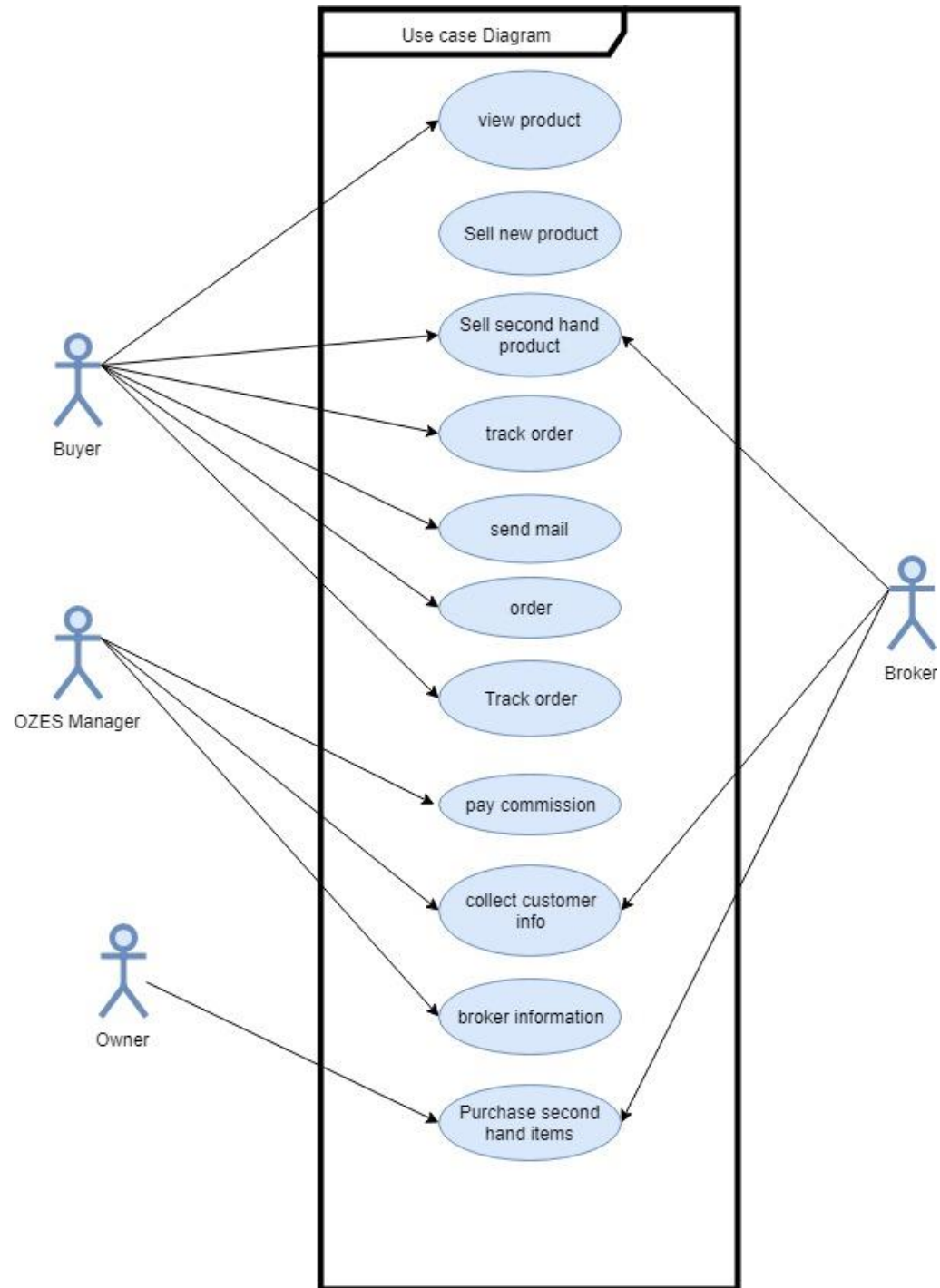
Class Diagram

1. Associate Broker



2. Order





Use case Description

Use case name	Sell second-hand items	
Scenario	To sell second hand items to customer	
Triggering event	If a customer call or click on order now button on a auction site	
Brief Description	Customer see the advertisement in the magazine or on an auction site then either call or provide the relevant information on the phone or online then makes a payment via card or in case of phone order pay to the broker.	
Actors	Customer , Broker	
Related use cases	Order if order is placed, collect customer info	
Stockholders	Accounting, Marketing, Sales.	
Precondition	Customer should have a phone or a computer with internet connection A mode of payment should be there	
Post conditions	Order must be placed Information must be stored Mail should be send	
Flow of activities	Actor	System

	<ol style="list-style-type: none"> 1. If the order is online then browse the item 2. Once the item is browser 3. Select the quantity of item 4. Click on the add to cart 5. Click on buy 	<ol style="list-style-type: none"> 1.1filter according to the given conditions 2.1show the description of selected items 3.1 check it the quantity is in the inventory 4.1 if Yes then add the selected items 5.1 ask for information 5.2 ask the mode of payment 5.3 verifies the user with one time password etc.
Exception Condition	<ol style="list-style-type: none"> 1.1no result found 3.1quantity not present 5.1 Information incomplete 5.2 verification not complete 	