Chapter 1

Synopsis

1.1 Project Title

Automatic Shopping Cart with Advanced Billing System

1.2 Project Option

Internal Project

1.3 Internal Guide

Prof. Jameer Kotwal

1.4 Technical Keywords (As per ACM Key-words)

1. SIFT
2. Image processing
3. Tensor flow
4. Open CV
5. Euclidean distance

1.5 Problem Statement

To reduce the time consumption during shopping in malls is to be tackled using this project. The increasing crowd is creating a lot of problem during billing and hence this problem is to be solved efficiently.

1.6 Abstract

In today’s world shopping experience is becoming convenient as the world has become digitalized. This is one of the conveniences that will be providing with new and easy shopping experience. To reduce the waiting time in a long queue for the payment of purchased items, we are developing an automatic shopping cart with advanced billing system. The system scans the product placed into the cart and accordingly generates the lump sum amount. This system consists of screen, camera, microcontroller, etc. Products will be captured by camera and will be processed using image processing algorithm SIFT. The user has to place the product into the cart, camera will capture the image of the product and this image will be compared with the images stored in the database using SIFT. Once the image is matched with the database image, the product price will be displayed on the screen.

1.7 Goals and Objectives

* To reduce the time of scanning the products by replacing the scanner to image processing techniques.
* To reduce the long queues by implementing the SIFT algorithm.
* To increase the overall revenue of the shopping malls by increasing the ROI.

1.8 Relevant mathematics associated with the Project

System Description:

1. Input: Product, Database
2. Output: Total billing amount
3. Identify data structures, classes, divide and conquer strategies to exploit distributed/parallel/concurrent processing, constraints.
4. Functions : Insert products, Capture Images, SIFT implementation, Increment count, Decrement count, Get product amount, Display
5. Mathematical formulation: Where X are products and i is total number of products
6. Success Conditions: Products should be correctly identified and respective amount should be computed.
7. Failure Conditions: Products when removed are not detected and amount is not deducted.

1.9 Names of Conferences / Journals where papers can be published

IJSER (International Journal Of Scientific And Engineering Research)

UGCON

1.10 Review of Conference/Journal Papers sup-porting Project idea

Object Recognition in Shopping Cart

Authors: - Pradeep Gurunathan, Vishal Guruprasad, Ganveer N

Smart Shopping Cart for Automated Billing Purpose Using Wireless Sensor Networks

Authors: - Udita Gangwal, Sanchita Roy, Jyotsna Bapat

Smart Trolley In Mega Mall

Authors: - J. S. Awati, S. B. Awati

1.11 Plan of Project Execution

The project plan execution gives us the idea about the time period required to complete the respective tasks. Following is the PERT chart that explains our project timeline.

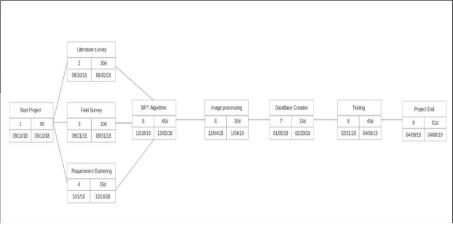


Figure 1.1: PERT chart

Chapter 2

Technical Keywords

2.1 Area of Project

Image Processing

2.2 Technical Keywords

SIFT

Image Processing

Tensor flow

Open CV

Euclidean distance

Chapter 3

Introduction

3.1 Project Idea

The basic idea is to give an efficient shopping way to customers. The main purpose is to reduce the waiting hours for billing in malls. This can be achieved by automation and hence we are automating the shop-ping cart. This will include the automatic scanning of products and generation of lump sum amount.

3.2 Motivation of the Project

The long queues in the shopping malls create impassiveness among the customers. This may cause the reduction in the ROI (Return on investment). Thus shopping malls are facing a lot of complications. Due to this the working or flow of the malls has to be changed. This is the main motivation behind this project.

3.3 Literature Survey

* Object Recognition in Shopping Cart

Authors :- Pradeep Gurunathan, Vishal Guruprasad, Ganveer N

The paper provides algorithm which successfully detects and identifies multiple grocery items using the Scale Invariant Feature Transform (SIFT) and image matching techniques. Integrating object detection techniques based on color into the algorithm would expand the type of detectable items rather than limiting the algorithm to items with unique labels and uniform shapes.

* Smart Shopping Cart for Automated Billing Purpose Using Wireless Sensor Networks

Authors :- Udita Gangwal, Sanchita Roy, Jyotsna Bapat

It describes the implementation of a reliable, fair and cost efficient Smart Shopping Cart using Wireless Sensor Networks and Image Processing techniques. It uses a passive sensor to reduce the communication requirement. The experimental set-up is tested for various test cases, with various products tested for all the possible cases mentioned in broadcast technique to communicate with the Base Station as each cart is associated with a unique ID. The system is cost-effective as it requires only one passive sensor (the load-cell) and a camera-based barcode scanner (which is way cheaper than any other type of barcode scanners) per cart.

* Smart Trolley in Mega Mall

Authors: - J. S. Awati, S.B.Awati

Microcontroller based design, has acquired the status of most happening field in electronics. This is highly specialized field that has the power of integrating thousands of transistors on single silicon chip. The microcontroller based trolley automatically follows the customer. Also it maintains safe distance between customer and itself. It gives number of products in trolley and total cost of the products on the spot.

Chapter 4

Problem Definition and scope

4.1 Problem Statement

To reduce the time consumption during shopping in malls is to be tackled using this project. The increasing crowd is creating a lot of problem during billing and hence this problem is to be solved efficiently.

4.1.1 Goals and objectives

Goal and Objectives:

* To reduce the time of scanning the products by replacing the scanner to image processing techniques.
* To reduce the long queues by implementing the SIFT algorithm.
* To increase the overall revenue of the shopping malls by increasing the ROI.

4.1.2 Statement of scope

The project cannot capture images of products whose size is greater than the size of the cart which in turn will create a problem for huge products such as bed, tables, cupboards and other furniture products.

The size of the cart plays an important role when we consider the number of products to be purchased. Thus, cart size will create a hindrance in number of products shopped.

4.2 Major Constraints

Python

4.3 Methodologies of Problem solving and efficiency issues

We are using waterfall model for our project estimation.

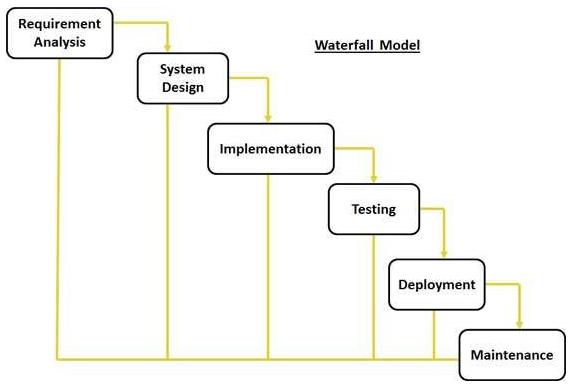


Figure 4.1: Waterfall model

1. Requirement gathering and analysis:

In this step of waterfall we identify what are various requirements are need for our project such are software and hardware required, database, and interfaces.

2. System Design:

In this system design phase we design the system which is easily understood for end user i.e. user friendly. We design some UML diagrams and data flow diagram to understand the system flow and system module and sequence of execution.

3. Implementation:

In implementation phase of our project we have implemented various module required of successfully getting expected outcome at the different module levels. With inputs from system design, the system is rst developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.

4. Testing:

The different test cases are performed to test whether the project module are giving expected outcome in assumed time. All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

5. Deployment of System:

Once the functional and non-functional testing is done, the product is deployed in the customer environment or released into the market.

6. Maintenance:

There are some issues which come up in the client environment. To x those issues patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as owing steadily downwards like a waterfall through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model phases do not overlap.

4.4 Applications

The major application of automatic cart is in shopping malls. The main purpose is to reduce the time required to scan the products in the queue of billing.

4.5 Hardware Resources Required

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameter | Minimum Requirement | |
|  |  |  |  |
| 1 | CPU Speed | 2 | GHz |
|  |  |  | |
| 2 | RAM | 3 GB | |
|  |  |  |  |
|  | Table 4.1: Hardware Requirements | | |

4.6 Software Resources Required

Platform :

1. Operating System: Windows
2. IDE: OpenCV
3. Programming Language: Python

Chapter 5

Project Plan

5.1 Project Estimates

5.1.1 Reconciled Estimates

1. Cost Estimate

The project cost can be found using any one of the model.

COCOMO-1 Model

COCOMO-2 Model

* Model -1: The basic COCOMO model computes software development efforts as a function of program size expressed in estimated lines of code.
* Model-2: The intermediate COCOMO model computes software development efforts as a function of program size and a set of cost drivers that include subjective assessment of the product, hardware, personnel, project attributes.
* Model-3: The advanced COCOMO model incorporates all characteristics of the intermediate version with an assessment of the cost drivers’ impact on each step of the software engineering process. Following is the basic COCOMO -2 models.

The basic COCOMO -2 model equations take form:

E=A(b)KLOCB(b)

D=C(b)ED(b)

Where E is the e ort applied in person months. D is development time in chronological month. KLOC is estimated number of delivered lines of code for the project. This project can be classified as Semidetached software project. The rough estimate of number of lines of this project is 9.072k.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Software Project | A(b) | B(b) | C(b) | D(b) |
|  |  |  |  |  |
| Organic | 2.4 | 1.05 | 2.5 | 0.38 |
|  |  |  |  |  |
| Semi-detached | 3.0 | 1.22 | 2.5 | 0.35 |
|  |  |  |  |  |
| Embedded | 3.6 | 1.20 | 2.5 | 0.32 |
|  |  |  |  |  |

Table 5.1: cocomo model

E=3.0\*(9.072)1.22

* 44.20 person- months D=2.5\* 44.35
* 9.40 months

Hence according COCOMO -2 model the time required for completion of the project is 9 ( 9.40) months.

Cost of Project:

Equation for calculation of cost of project using COCOMO - 2 model is:

C = D \* Cp

Where,

C = Cost of project

D = Duration in month

Cp = Cost incurred per person-month, Cp=Rs.1500/- (per person-month)

C = 9 \* 1500

= 13500/-

Hence according COCOMO - 2 model the cost of project is 13500/-

1. Time Estimates

The time estimate of this project is approximate 9.40 months.

5.1.2 Project Resources

PC, Operating System, Python, Dataset, Pycharm IDE , Processor, Hard disk, Keyboard, Mouse, RAM.

5.2 Risk Management

NP Hard: A problem is NP-hard if an algorithm for solving it can be translated into one for solving any NP-problem (nondeterministic polynomial time) problem. NP-hard therefore means "at least as hard as any NP-problem," although it might, in fact, be harder.

We start with the idea of a decision problem, a problem for which an algorithm can always answer yes or no. We also need the idea of two models of computer: deterministic and non-deterministic. A deterministic computer is the regular computer we always thinking of; a non-deterministic computer is one that is just like were used to except that is has unlimited parallelism, so that any time you come to a branch, we spawn a new process and examine both sides. Why the system is considered under NP-Complete at this stage? In development stage, project deals in NP HARD, and it is NP-complete, as heuristics have to be used to restrict the search space.

5.2.1 Risk Identification

For risks identification, review of scope document, requirements specifications and schedule is done. Answers to questionnaire revealed some risks. Each risk is categorized as per the categories mentioned in. Please refer table 5.2 for all the risks. You can refer following risk identification questionnaire.

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Risk Description | Probability |  | Impact |  |  |
|  |  |  |  |
| Schedule | Quality | Overall |  |
|  |  |  |  |
|  |  |  |  |  |  |  |
| 1 | Algorithm Study | Low | Low | High | High |  |
|  |  |  |  |  |  |  |
| 2 | Implementation | Low | Low | High | High |  |
|  |  |  |  |  |  |  |

Table 5.2: Risk Table

|  |  |  |
| --- | --- | --- |
| Probability | Value | Description |
|  |  |  |
| High | Probability of occurrence is | > 75% |
|  |  |  |
| Medium | Probability of occurrence is | 26 75% |
|  |  |  |
| Low | Probability of occurrence is | < 25% |
|  |  |  |

Table 5.3: Risk Probability definitions

|  |  |  |
| --- | --- | --- |
| Impact | Value | Description |
|  |  |  |
| Very high | > 10% | Schedule impact or Unacceptable quality |
|  |  |  |
| High | 5 10% | Schedule impact or Some parts of the project have |
|  |  | low quality |
|  |  |  |
| Medium | < 5% | Schedule impact or Barely noticeable degradation |
|  |  | in quality Low Impact on schedule or Quality can |
|  |  | be incorporated |
|  |  |  |

Table 5.4: Risk Impact definitions

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

|  |  |
| --- | --- |
| Risk ID | 1 |
|  |  |
| Risk Description | Algorithm Study |
|  |  |
| Category | Development Environment. |
|  |  |
| Source | Paper, Globalised Content Analysis. |
|  |  |
| Probability | Low |
|  |  |
| Impact | High |
|  |  |
| Response | Mitigate |
|  |  |
| Strategy | Strategy |
|  |  |
| Risk Status | Occurred |
|  |  |
|  |  |
| Risk ID | 2 |
|  |  |
| Risk Description | Implementation |
|  |  |
| Category | Requirements |
|  |  |
| Source | Software Design Specification i.e. Language. |
|  |  |
| Probability | Low |
|  |  |
| Impact | High |
|  |  |
| Response | Mitigate |
|  |  |
| Strategy | Better testing will resolve this issue. |
|  |  |
| Risk Status | Identified |
|  |  |
|  | Table 5.5: Risk Impact definitions |

5.3 Project Schedule

5.3.1 Project task set

Major Tasks in the Project stages are:

Task 1:Literature Survey Task 2: Field Survey

Task 3: Requirement Gathering

Task 4: Database Creation

Task 5: SIFT algorithm implementation

Task 6: Testing

5.3.2 Task network

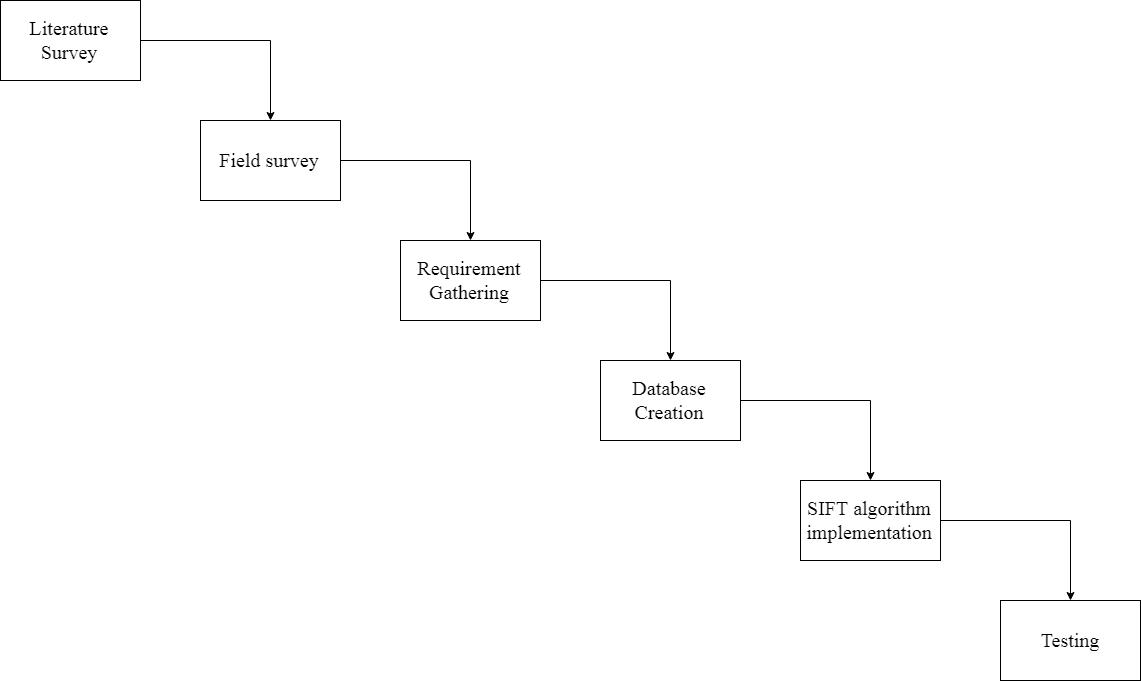


Figure 5.1: Timeline Network

5.3.3 Timeline Chart

Timeline chart explains the sequential time period required for implementing project tasks and the events associated with it.

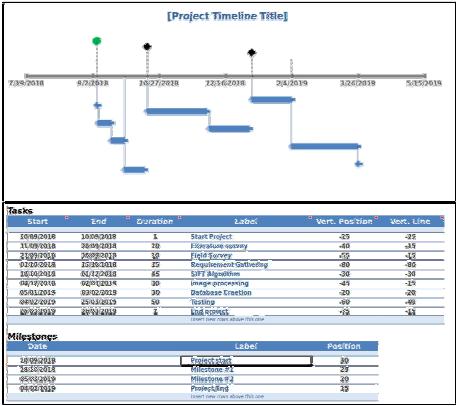


Figure 5.2: Timeline Network

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 5.4 | Team Organization | |  |  |  |  |  |
| 5.4.1 | Timeline Chart | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Sr. | Activity | Description |  |  | Proposed | Actual | Status |
|  | Planned |  |  |  |  |  |  |
| No. |  |  |  |  | Date | Date |  |
|  |  |  | | |  |  |  |
| 1. | Topic Search | Searching for topics | | | 18/07/18 | 25/07/18 | Complete |
|  |  | and finalizing one | |  |  |  |  |
|  |  |  | | |  |  |  |
| 2. | Literature Sur- | Searching & studying | | | 05/08/18 | 09/08/18 | Complete |
|  | vey |  |  |  |  |  |  |
|  |  | related papers | |  |  |  |  |
|  |  |  | | |  |  |  |
| 3. | Analyzing | Identifying software | | | 12/08/18 | 12/08/18 | Complete |
|  | Requirements | & hardware requirement |  | |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | | |  |  |  |
| 4. | Synopsis | Synopsis submission. | | | 19/08/18 | 22/08/18 | Complete |
|  |  |  | |  |  |  |  |
| 5. | Planning | Planning project | |  | 01/09/18 | 06/09/18 | Complete |
|  |  | activities |  |  |  |  |  |
|  |  |  |  | |  |  |  |
| 6. | Designing | Designing | different | | 19/09/18 | 03/10/18 | Complete |
|  |  | UML |  |  |  |  |  |
|  |  | diagrams |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| 7. | Design Report | Completion | of | first | 27/10/18 | 03/11/18 | Complete |
|  |  | phase report |  |  |  |  |  |
|  |  |  | |  |  |  |  |
| 8. | Phase One | Implementation | | of | 14/12/18 | 28/12/18 | Complete |
|  |  | first module |  |  |  |  |  |
|  |  |  | | |  |  |  |
| 9. | Phase Two | Implementation of | | | 07/01/19 | 21/01/19 | Complete |
|  |  | second module | |  |  |  |  |
|  |  |  | | |  |  |  |
| 10. | Phase Three | Implementation of | | | 22/01/19 | 05/02/19 | Complete |
|  |  | third module |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Activity | Description | Proposed | Actual | Status |
|  | Planned |  |  |  |  |
| No. |  |  | Date | Date |  |
|  |  |  |  |  |  |
| 11. | Phase Four | Implementation of | 15/02/19 | 01/03/19 | Complete |
|  |  | fourth module |  |  |  |
|  |  |  |  |  |  |
| 12. | System Testing | Perform testing using | 05/03/19 | 19/03/19 | Complete |
|  |  | open source tool |  |  |  |
|  |  |  |  |  |  |
| 13. | Final Report | Completion of final | 15/04/19 | 23/04/19 | Complete |
|  |  | report |  |  |  |
|  |  |  |  |  |  |

5.4.2 Team structure

|  |  |  |
| --- | --- | --- |
| SR | Team Member | Task |
| NO |  |  |
|  |  |  |
| 1. | Niranjani And Shrutika | Topic Search |
|  |  |  |
| 2. | Shraddha Waphare | Literature Survey |
|  |  |  |
| 3. | Rachana Patil | Synopsis |
|  |  |  |
| 4. | Niranjani Wagh | Planning |
|  |  |  |
| 5. | Shraddha Waphare | Design |
|  |  |  |
| 6. | Shrutika Phalke | Design Report |
|  |  |  |
| 7. | Rachana Patil | Report |
|  |  |  |
| 8. | Shrutika Phalke | Coding |
|  |  |  |
| 9. | Niranjani Wagh | Testing |
|  |  |  |
| 10. | Shraddha and Rachana | Final Report |
|  |  |  |

Table 5.6: Team Structure

Chapter 6

Software requirement specification

6.1 Introduction

The purpose of the document is to gather and analyze all ideas that have come up to de ne the system, its requirements with respect to consumers. Also, prediction and sorting is carried out with respect to how we hope this product will be used. This is done for the better understanding of the project, concepts that may be developed, and document ideas that are being considered, but this may be discarded as the product develops. In short, the purpose of this document is to provide a overview of our software product, its parameters and goals. This document describes the target audience and its UI, hardware and software requirements. It defines how our client, team and audience see the product and its functionality. Nonetheless, it helps any designer and developer to assist in software delivery life cycle (SDLC) processes.

6.1.1 Purpose and Scope of Document

The project cannot capture images of products whose size is greater than the size of the cart which in turn will create a problem for huge products such as bed, tables, cupboards and other furniture products. The size of the cart plays an important role when we consider the number of products to be purchased. Thus, cart size will create a hindrance in number of products shopped.

6.2 Usage Scenario

There are two users or actors which are a follows

Developer- Who will create project, will accept real time data, and Build project and also Deploy project.

End user-Who will use analyzed data for analysis purpose.

6.2.1 User profiles

The pro les of all user categories are described here.(Actors and their Description)

6.2.2 Use-cases

All use-cases for the software are presented. Description of all main Use cases using use case template is to be provided.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. No. | Use Case | Description | | Actors | Assumptions |
|  |  |  | |  |  |
| 1 | Purchase a product | any product | | user | Added to |
|  |  | purchased |  |  | cart |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| 2 | Product added to cart | amount | is | user | Added |
|  |  | displayed | |  |  |
|  |  | on screen | |  |  |
|  |  |  |  |  |  |
| 3 | Product removed from cart | amount | is | user | Deleted |
|  |  | deducted | |  |  |
|  |  |  | |  |  |
|  | Table 6.1: Use Cases | | |  |  |

6.2.3 Use Case View

Use Case Diagram.

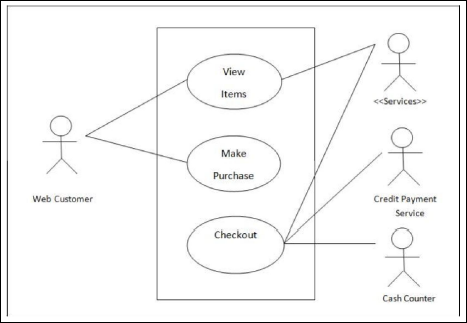


Figure 6.1: Use case diagram

6.3 Data Model and Description

6.3.1 Data Description

Data objects that will be managed/manipulated by the software are described in this section. The database entities or les or data structures required to be described. For data objects details can be given as below

6.3.2 Data objects and Relationships

Data objects and their major attributes and relationships among data objects are described using an ERD- like form.

6.4 Functional Model and Description

A description of each major software function, along with data flow (structured analysis) or class hierarchy (Analysis Class diagram with class description for object oriented system) is presented.

6.4.1 Data Flow Diagram

Level 0 Data Flow Diagram

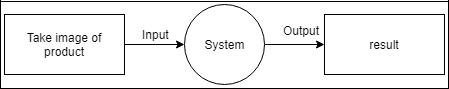


Figure 6.2: Level 0

Level 1 Data Flow Diagram

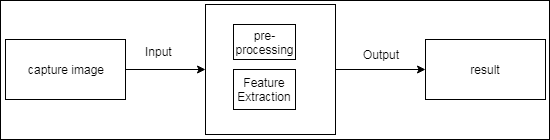


Figure 6.3: Level 1

Level 2 Data Flow Diagram

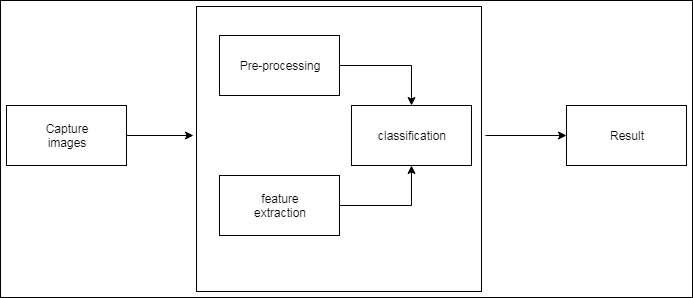


Figure 6.4: Level 2

6.4.2 Activity Diagram:

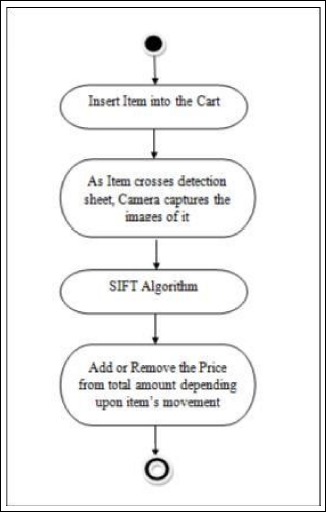


Figure 6.5: Activity Diagram

6.4.3 Non Functional Requirements:

Performance Requirement

* The performance of the functions and every module must be well.
* The overall performance of the software will enable the users to work efficiently.
* Performance of encryption of data should be fast.
* Performance of the providing virtual environment should be fast.

Safety Requirement-

The application is designed in modules where errors can be detected and fixed easily. This makes it easier to install and update new functionality if required.

Security Requirement-

All data will be encrypted using strong encryption algorithm and according to location encryption is done. Soft-ware Quality Attributes- Our software has many quality attribute that are given below:-

* Adaptability: This software is adaptable by all users.
* Availability: This software is freely available to all users. The availability of the software is easy for everyone.
* Maintainability: After the deployment of the project if any error occurs then it can be easily maintained by the software developer.
* Reliability: The performance of the software is better which will increase the reliability of the Software.
* User Friendliness: Since, the software is a GUI application; the output generated is much user friendly in its behavior.
* Integrity: Integrity refers to the extent to which access to software or data by unauthorized persons can be controlled.
* Security: Users are authenticated using many security phases so reliable security is provided.
* Testability: The software will be tested considering all the aspects.

6.4.4 State Diagram:

State Transition Diagram

Fig.6.6 example shows the state transition diagram of Cloud SDK. The states are represented in ovals and state of system gets changed when certain events occur. The transitions from one state to the other are represented by arrows. The Figure shows important states and events that occur while creating new project.

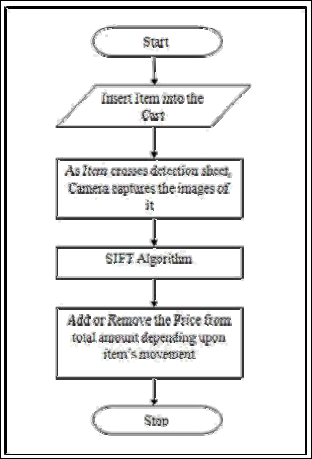


Figure 6.6: State transition diagram

Chapter 7

Detailed Design Document using Appendix A

7.1 Introduction

Detailed design of the system is the final design activity before implementation begins.

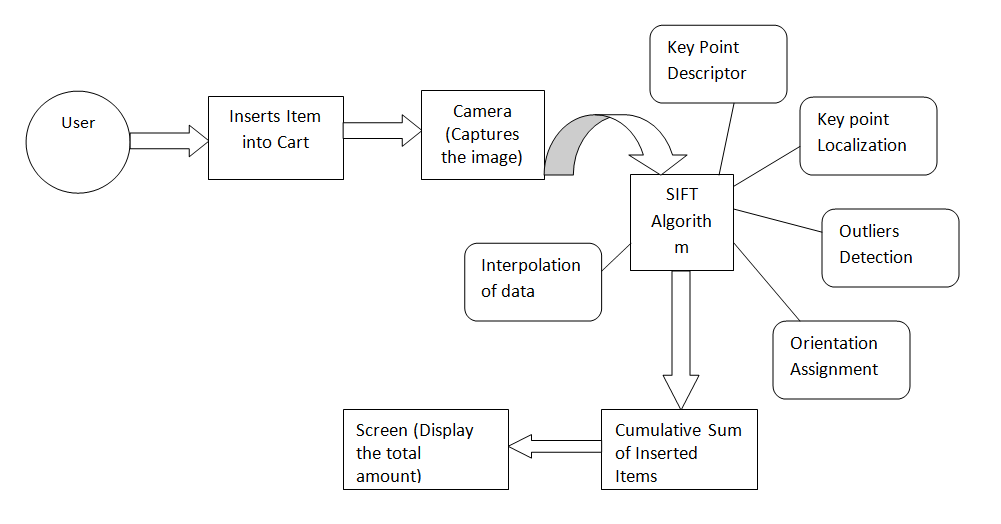


Fig 7.1 System Architecture

7.2.4 Database description

Database consists of all the detailed information regarding each and every product. Some of the information entities are Product ID, Product Name, Product Price and images of that particular product.

7.3 Component Design

7.3.1 Class Diagram

Class Diagram is a type of structured diagram that describes the structure of a system by showing the various classes in the system, their attributes, operations or functions and the relationships among objects.

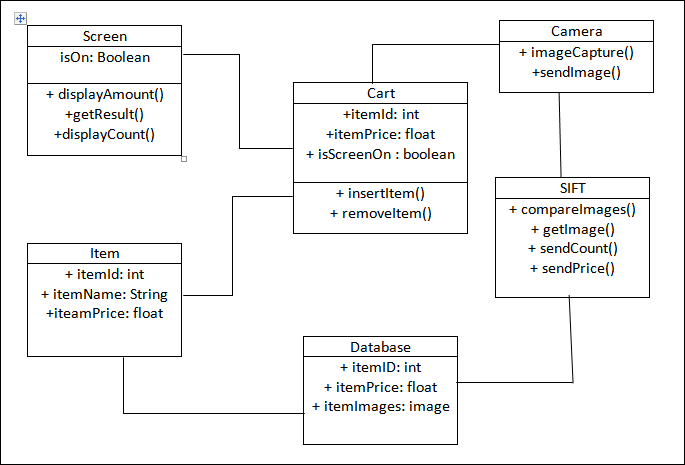


Figure 7.1: Class Diagram

Chapter 8

Project Implementation

8.1 Introduction

The purpose of the document is to gather and analyze all ideas that have come up to de ne the system, its requirements with respect to consumers. Also, prediction and sorting is carried out with respect to how we hope this product will be used. This is done for the better understanding of the project, concepts that may be developed, and document ideas that are being considered, but this may be discarded as the product develops. In short, the purpose of this document is to provide a overview of our software product, its parameters and goals. This document describes the target audience and its UI, hardware and software requirements. It defines how our client, team and audience see the product and its functionality. Nonetheless, it helps any designer and developer to assist in software delivery life cycle (SDLC) processes.

8.2 Tools and Technologies Used

8.2.1 Tools used

1. JIRA

JIRA is a tool developed by Australian Company Atlassian. It is used for bug tracking, issue tracking, and project management. The name "JIRA" is actually inherited from the Japanese word "Gojira" which means "Godzilla".

The basic use of this tool is to track issue and bugs related to your software and Mobile apps. It is also used for project management. The JIRA dashboard consists of many useful functions and features which make handling of issues easy.

8.2.2 Algorithm Details

SIFT algorithm:

SIFT is quite an involved algorithm. It has a lot going on and can become confusing, so I’ve split up the entire algorithm into multiple parts. Here’s an outline of what happens in SIFT.

1. Constructing a scale space This is the initial preparation. You create internal representations of the original image to ensure scale invariance. This is done by generating a "scale space".
2. LoG Approximation The Laplacian of Gaussian is great for finding interesting points (or key points) in an image. But it’s computationally expensive. So we cheat and approximate it using the representation created earlier.
3. Finding key points with the super fast approximation, we now try to find key points. These are maxima and minima in the Difference of Gaussian image we calculate in step 2
4. Get rid of bad key points Edges and low contrast regions are bad key points. Eliminating these makes the algorithm efficient and robust. A technique similar to the Harris Corner Detector is used here.
5. Assigning an orientation to the key points an orientation is calculated for each key point. Any further calculations are done relative to this orientation. This effectively cancels out the effect of orientation, making it rotation invariant.
6. Generate SIFT features finally, with scale and rotation invariance in place, one more representation is generated. This helps uniquely identify features.

8.3 Verification and Validation for Acceptance

Verification and validation are independent procedures that are used together for checking that a product, service, or system meets requirements and specifications and that it fulfills its intended purpose. These are critical components of a quality management system such as ISO 9000. The words "verification" and "validation" are sometimes preceded with "independent", indicating that the verification and validation is to be performed by a disinterested third party. "Independent verification and validation" can be abbreviated as "IV and V". "Validation. The assurance that a product, service, or system meets the needs of the customer and other identified stake-holders. It often involves acceptance and suitability with external customers.

Contrast with Verification. The evaluation of whether or not a product, service, or system complies with a regulation, requirement, specification, or imposed condition. It is often an internal process. Contrast with validation."

Chapter 9

Software Testing

9.1 Type of Testing

1. Unit Testing

It is the testing of individual software units of the application .it is done after the complexion of an individual unit before integration. Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

1. Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

1. White-box testing

In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases.

1. Black-box testing

Black-box testing treats the software as a and black box examining functionality without any knowledge of internal implementation. The testers are only aware of what the software is supposed to do, not how it does it.

9.1.1 Tools Used

JIRA

JIRA is a tool developed by Australian Company Atlassian. It is used for bug tracking, issue tracking, and project management. The name "JIRA" is actually inherited from the Japanese word "Gojira" which means "Godzilla". The basic use of this tool is to track issue and bugs related to your software and Mobile apps. It is also used for project management. The JIRA dash-board consists of many useful functions and features which make handling of issues easy.

9.2 Test Cases and Test Results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | |  |  |  |  |
|  |  |  |  |  |  |  |
| Test |  |  | Actual | Expected | Test Case |  |
| Case | Test Case | Test Case I/P | Criteria |  |
| result | Result |  |
| ID |  |  | (P/F) |  |
|  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Add |  |  |  |  |  |
| 001 | products to the | Any product | Accept | Accept | P |  |
|  | cart |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Detect the |  |  |  |  |  |
|  | product |  |  |  |  |  |
| 002 | and match | Any product | Accept | Accept | P |  |
|  | with |  |  |  |  |  |
|  | dataset |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Image |  |  |  |  |  |
| 003 | Identifica- | Any product | Accept | Accept | P |  |
|  | tion |  |  |  |  |  |
|  |  |  |  |  |  |  |
| 004 | Payment | Payment verified | Accept | Accept | P |  |
| verification |  |
|  |  |  |  |  |  |  |

Table 9.2: System Test Case

Chapter 10

Results

10.1 Screen shots

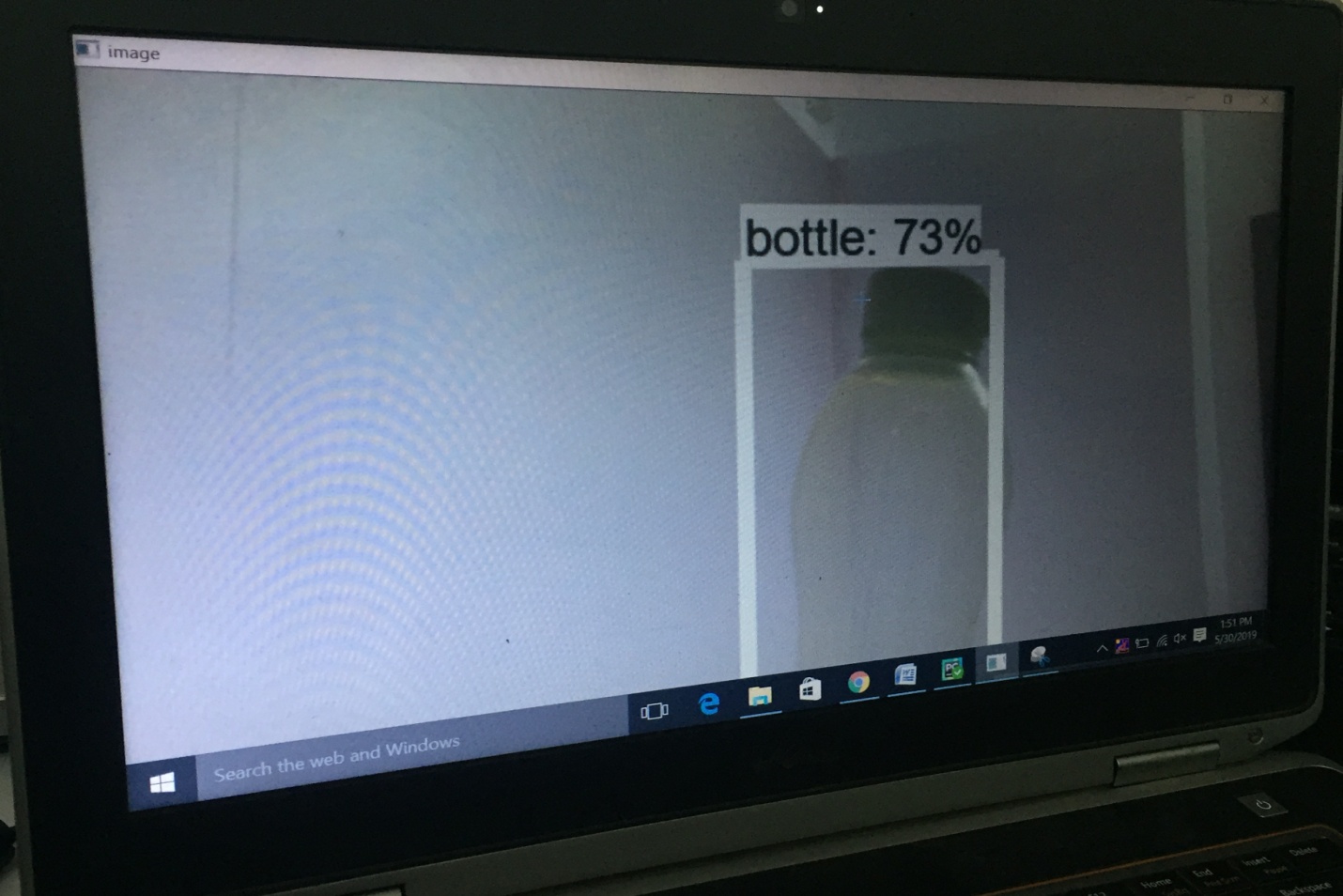


Fig 10.1 Object Detection

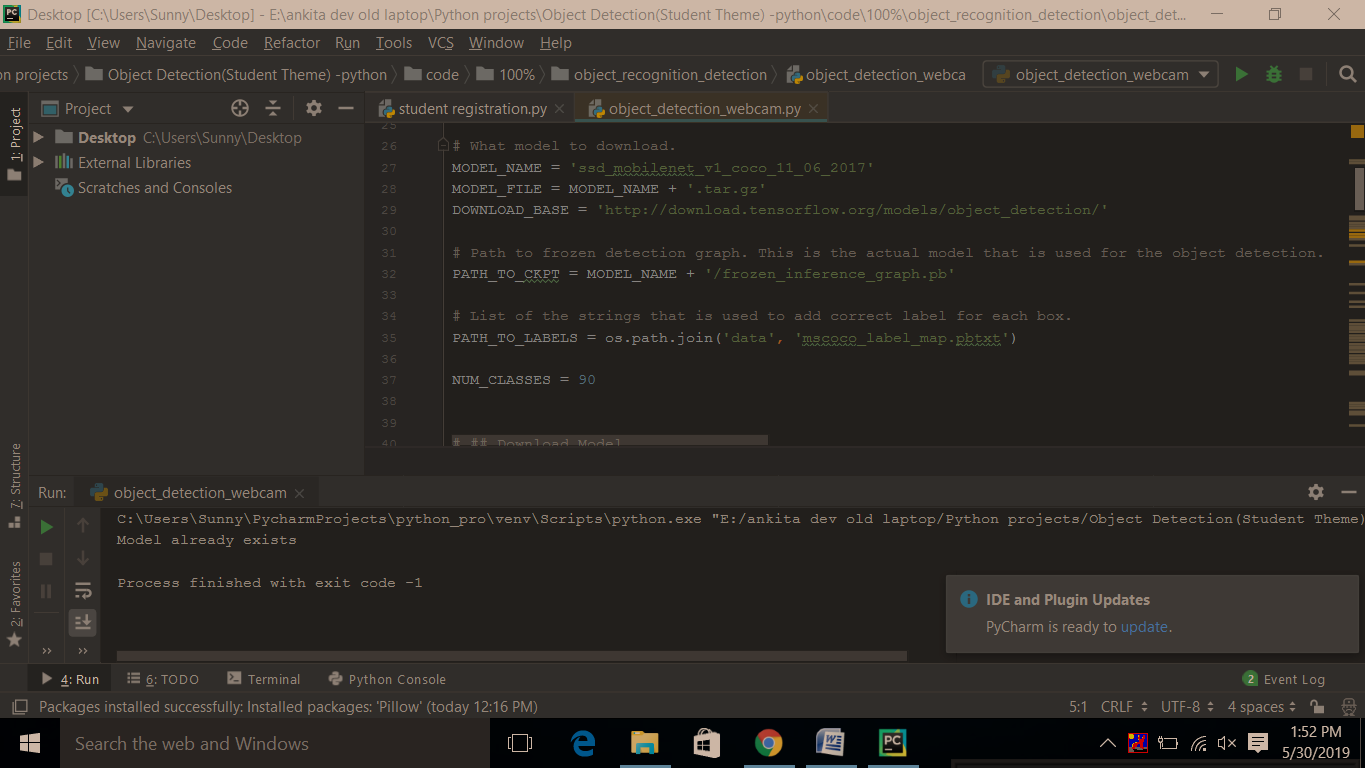


Fig 10.2 Camera Module

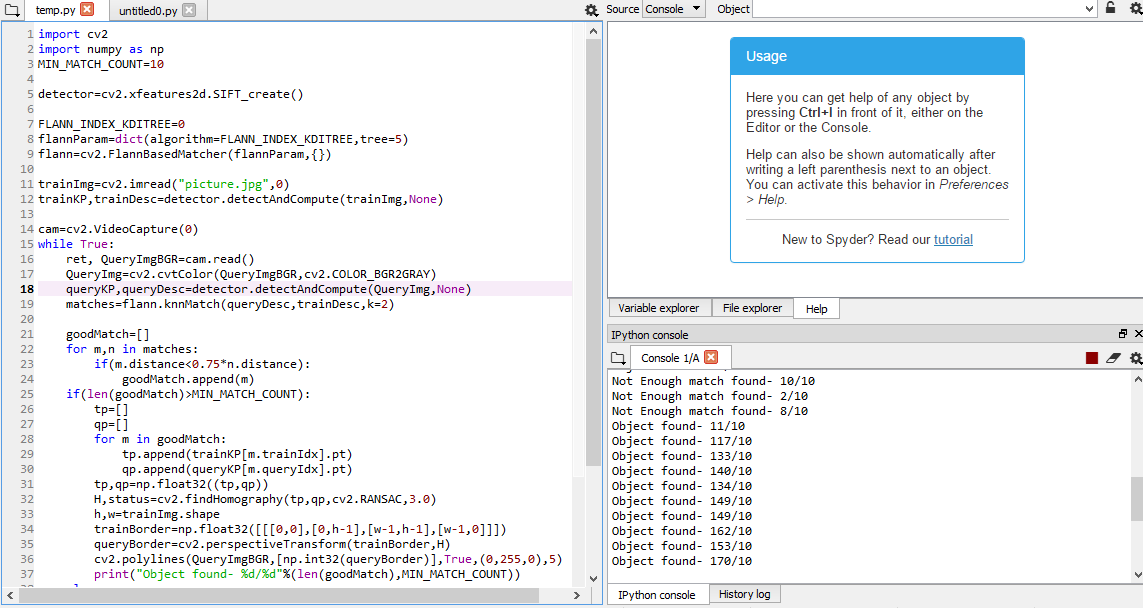


Fig 10.3 SIFT Implementation

10.2 Outputs

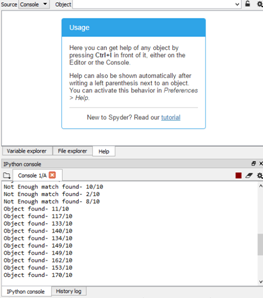


Fig 10.4 Key points Output

Chapter 11

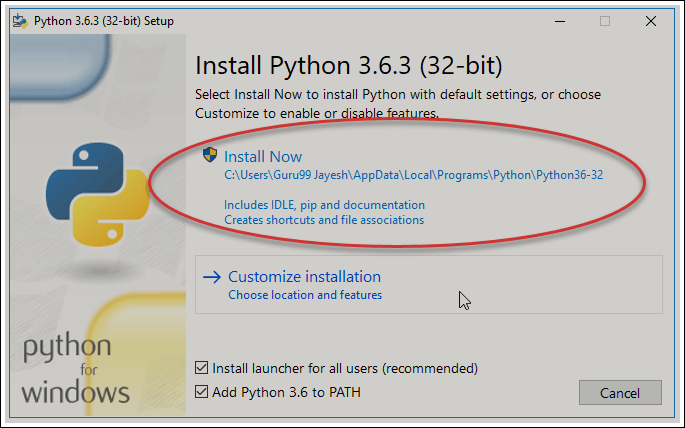
Deployment and Maintenance

11.1 Installation and un-installation

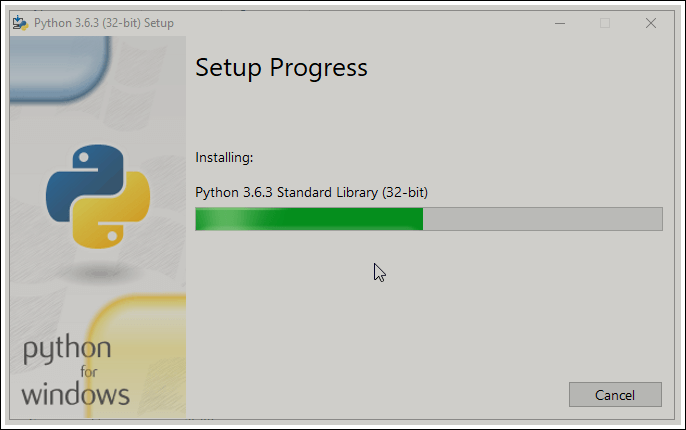
Step 1) To download and install Python visit the official website of Python http://www.python.org/downloads/ and choose your version. We have chosen Python version



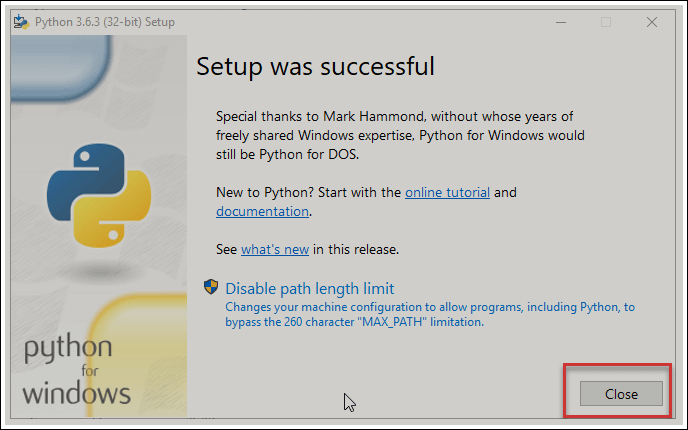
Step 2) Once the download is complete, run the exe for install Python. Now click on Install Now.



Step 3) You can see Python installing at this point.

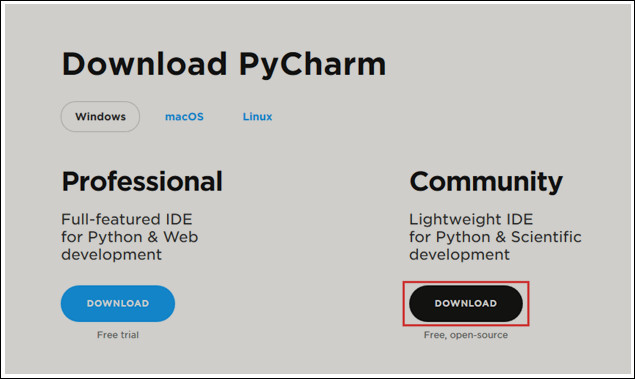


Step 4) When it finishes, you can see a screen that says the Setup was successful. Now click on "Close".



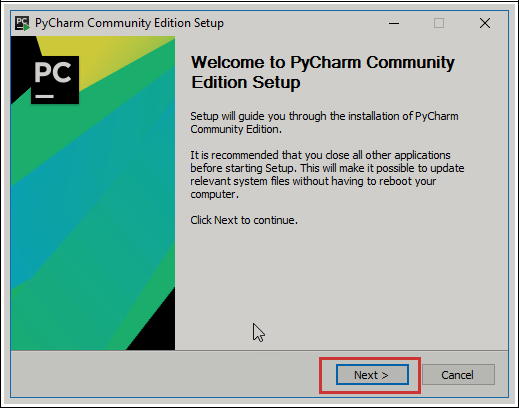
Downloading Pycharm:

Step 1) To download PyCharm visit the website www.jetbrains.com/pycharm/download/ and Click the "DOWNLOAD" link under the Community Section.

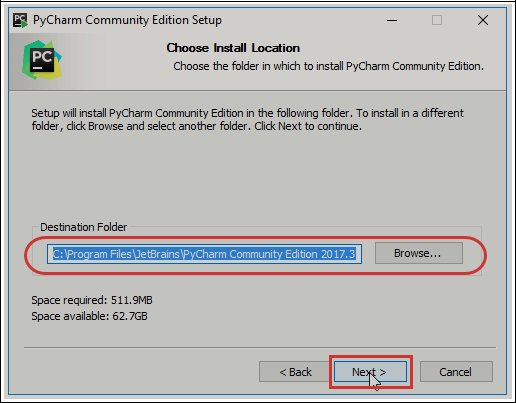


Step 2) Once the download is complete, run the exe for install PyCharm.

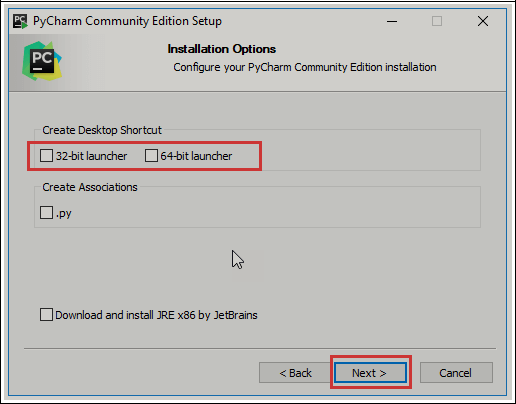
The setup wizard should have started. Click \Next".



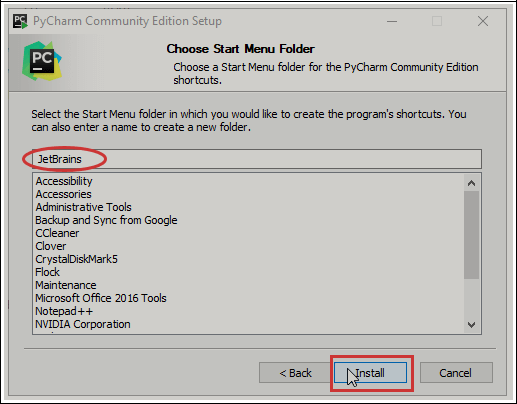
Step 3) On the next screen, Change the installation path if required. Click \Next".



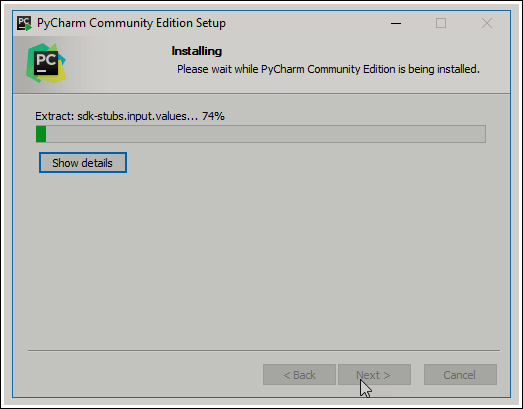
Step 4) On the next screen, you can create a desktop shortcut if you want and click on \Next".



Step 5) Choose the start menu folder. Keep selected JetBrains and click on \Install".



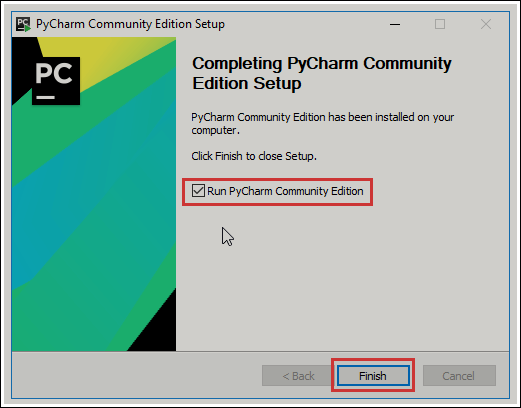
Step 6) Wait for the installation to finish.



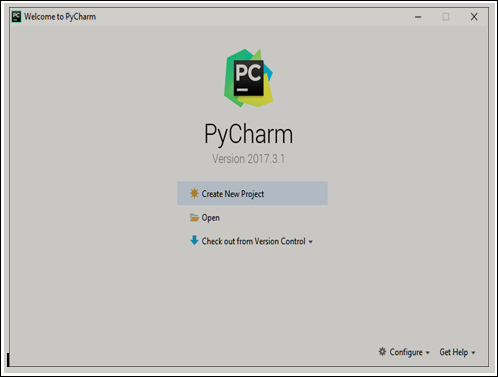
Step 7) Once installation finished, you should receive a message screen that

PyCharm is installed. If you want to go ahead and run it, click the \Run

PyCharm Community Edition" box first and click \Finish".



Step 8) After you click on "Finish," the Following screen will appear.



Chapter 12

Conclusion and future scope

Thus, a system called Automatic Shopping Cart with Advanced Billing Sys-tem will be developed using SIFT algorithm, Image Processing, Machine Learning. As we have overcome the two measure problems of computational time and physical wastage of time thus, the time complexity is reduced. These problems were overcome by using the image capturing technique in the shopping cart itself. The computational time could be greatly reduced by using above technologies. In turn, the waiting time of the user will be reduced and the shopping experience will be enhanced.

References

Object Recognition in Shopping Cart

Authors: - Pradeep Gurunathan, Vishal Guruprasad, Ganveer N

The paper provides algorithm which successfully detects and identifies multiple grocery items using the Scale Invariant Feature Transform (SIFT) and image matching techniques. Integrating object detection techniques based on color into the algorithm would expand the type of detectable items rather than limiting the algorithm to items with unique labels and uniform shapes.

Smart Shopping Cart For Automated Billing Purpose Using Wireless Sensor Networks

Authors :- Udita Gangwal, Sanchita Roy, Jyotsna Bapat

It describes the implementation of a reliable, fair and cost efficient Smart Shopping Cart using Wireless Sensor Networks and Image Processing techniques. The experimental set-up is tested for various test cases, with various products tested for all the possible cases mentioned in broadcast technique to communicate with the Base Station as each cart is associated with a unique ID.

Smart Trolley In Mega Mall

Authors :- J. S. Awati, S. B. Awati

Microcontroller based design, has acquired the status of most happening field in electronics. This is highly specialized field that has the power of integrating thousands of transistors on single silicon chip. The microcontroller based trolley automatically follows the customer. Also it maintains safe distance between customer and itself. It gives number of products in trolley and total cost of the products on the spot.

Annexure A

Laboratory assignments on Project Analysis of Algorithmic Design

A.1 Project problem statement feasibility assessment using NP- Hard, NP-Complete or satisﬁability issues using modern algebra and / co relevant mathematical models.

Mathematical Model

1. Input: Product, Database
2. Output: Total billing amount
3. Identify data structures, classes, divide and conquer strategies to exploit distributed/parallel/concurrent processing, constraints.
4. Functions : Insert products, Capture Images, SIFT implementation, Increment count, Decrement count, Get product amount, Display
5. Mathematical formulation:



Where X are products and i is total number of products

1. Success Conditions: Products should be correctly identified and respective amount should be computed.
2. Failure Conditions: Products when removed are not detected and amount is not deducted.

NP Hard:

A problem is NP-hard if an algorithm for solving it can be translated into one for solving any NP-problem (nondeterministic polynomial time) problem. NP-hard therefore means "at least as hard as any NP-problem," although it might, in fact, be harder.

We start with the idea of a decision problem, a problem for which an algorithm can always answer yes or no. We also need the idea of two models of computer: deterministic and non-deterministic. A deterministic computer is the regular computer we always thinking of; a non-deterministic computer is one that is just like were used to except that is has unlimited parallelism, so that any time you come to a branch, we spawn a new process and examine both sides. Why the system is considered under NP-Complete at this stage? In development stage, project deals in NP HARD, and it is NP-complete, as heuristics have to be used to restrict the search space.

Project status:

On the basis of the above deﬁnitions, our problem statement is solvable in polynomial time by deterministic Turing machine hence it is said to be NP- complete.

Annexure B

Laboratory assignments on Project Quality and Reliability Testing of Project Design

1. Functional Dependency graphs and relevant Software modeling methods

USE CASE DIAGRAM

A use case diagram in the Uniﬁed Modeling Language (UML) is a type of behavioral diagram deﬁned by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

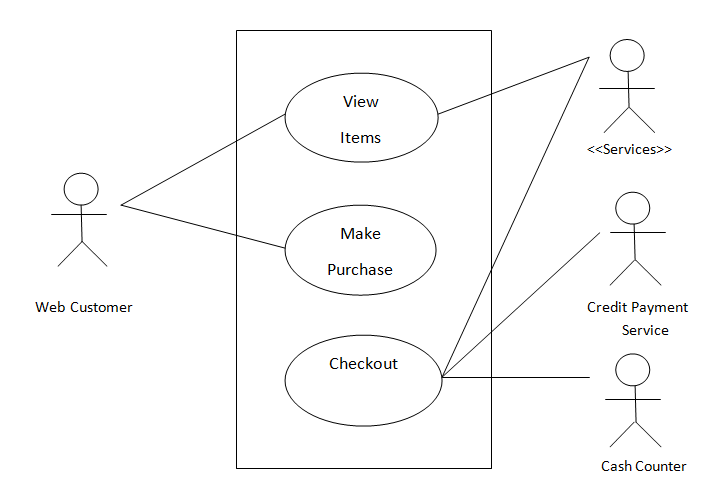


Fig. Use Case Diagram

CLASS DIAGRAM

In software engineering, a class diagram in the Uniﬁed Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system’s classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

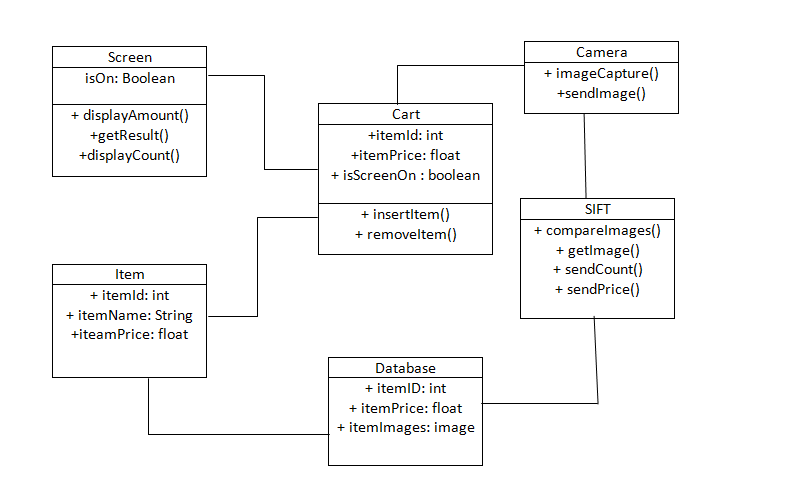


Fig. Class Diagram

ACTIVITYDIAGRAM

Activity diagrams are graphical representations of work ﬂows of stepwise activities and actions with support for choice, iteration and concurrency. In the Uniﬁed Modeling Language, activity diagrams can be used to describe the business and operational step-by-step work ﬂows of components in a system. An activity diagram shows the overall ﬂow of control.

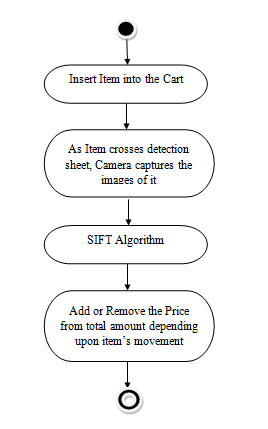


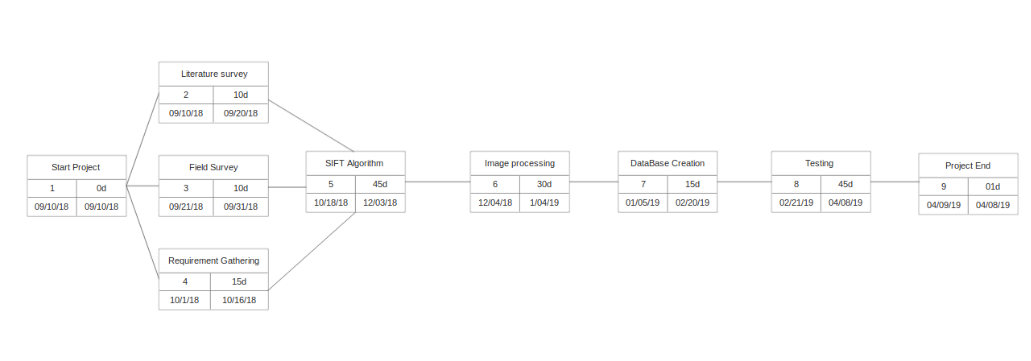
Fig. Activity Diagram

2. Testing the Problem Statement using general test data

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | |  |  |  |  |
|  |  |  |  |  |  |  |
| Test |  |  | Actual | Expected | Test Case |  |
| Case | Test Case | Test Case I/P | Criteria |  |
| result | Result |  |
| ID |  |  | (P/F) |  |
|  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Add |  |  |  |  |  |
| 001 | products to the | Any product | Accept | Accept | P |  |
|  | cart |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Detect the |  |  |  |  |  |
|  | product |  |  |  |  |  |
| 002 | and match | Any product | Accept | Accept | P |  |
|  | with |  |  |  |  |  |
|  | dataset |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Image |  |  |  |  |  |
| 003 | Identifica- | Any product | Accept | Accept | P |  |
|  | tion |  |  |  |  |  |
|  |  |  |  |  |  |  |
| 004 | Payment | Payment verified | Accept | Accept | P |  |
| verification |  |
|  |  |  |  |  |  |  |

Annexure C

PROJECT PLANNER

Fig C.1 Project Planner

Annexure D

Reviewers Comments of Paper Submitted

1. Paper Title:

Automatic Shopping Cart with Advanced Billing System

2. Name of the Conference/Journal where paper submitted:

UGCON

3. Paper accepted/rejected: Accepted

Annexure E

Plagiarism Report

Plagiarism report

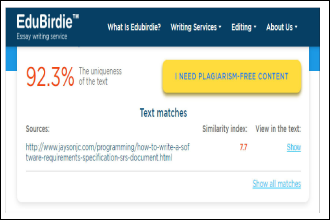


Figure B.1: Plagiarism report

Annexure F

Information of Project Group Members



1. Name: Niranjani Wagh

2. Date of Birth: 29 August 1997

3. Gender: Female

4. Permanent Address: Mohan nagar, Chinchwad

5. E-Mail: nirajaniwagh111@gmail.com

6. Mobile/Contact No.: 7743996388

7. Placement Details: Placed

8. Paper Published: Yes



1. Name: Shrutika Phalke

2. Date of Birth: 27 May 1998

3. Gender: Female

4. Permanent Address: Nigdi

5. E-Mail: phalkeshruti27@gmail.com

6. Mobile/Contact No.: 8407958890

7. Placement Details: Placed

8. Paper Published: Yes



1. Name: Shraddha Waphare

2. Date of Birth: 30 March 1998

3. Gender: Female

4. Permanent Address: Man

5. E-Mail: shraddhawafare30@gmail.com

6. Mobile/Contact No.: 7709357526

7. Placement Details: Placed

8. Paper Published: Yes



1. Name: Rachana Patil

2. Date of Birth: 19 March 1997

3. Gender: Female

4. Permanent Address: Chinchwad

5. E-Mail: rachanapatil475@gmail.com

6. Mobile/Contact No.: 8856849696

7. Placement Details: Placed

8. Paper Published: Yes