# CIS 634 OBJECT-ORIENTED SOFTWARE ENGINEERING

# COURSE PROJECT REPORT

# Employee Information System

# Logo, company name Description automatically generated

# Prepared by:

# Likith Roshan Chinnabathini (2786602)

# Sai Naveen Reddy Syamala (2788875)

# Kalyan Naga Venkata Ram Sai Annam (2788985)

# ABSTRACT

Employee Management System provides Employee information management system. Use in presentation of an information system for managing the staff data within a small company or organization. The system as such as it has been developed is called Employee Management System. It consists of functionally related GUI (application program) and database.

Employee information management system is a suitable one for running an organization in a successful manner. It permits recording all the points of interest of representatives to oversee them from anyplace. Grandeur employee management system is a suitable one for coordinating with human asset modules by tending to correct needs. An alternate preference is that it serves to keep individual record such as worker advantages, occurrence reports, evaluations, and so on.

Management of All Employee Information Data such as names, titles, addresses and salaries are a basic start. Salary and position history, reporting structures, performance appraisal histories, and other critical employee information. Management of all employee information. Company-related documents such as employee handbooks, emergency evacuation procedures, and safety guidelines. Benefits administration including enrollment, status changes, and personal information updating. In an ideal system, you can allow employees to look up and review their own information, including vacation tracking

***INDEX***

## S.NO. CHAPTER PAGE

1. [INTRODUCTION 1](#_TOC_250022)
2. [SYSTEM ANALYSIS 2](#_TOC_250021)
   1. [DEFINITION 2](#_TOC_250020)
   2. [DESCRIPTION OF PRESENT SYSTEM 2](#_TOC_250019)
   3. [PROPOSED SYSTEM 3](#_TOC_250018)
   4. [FEASIBILITY STUDY 4](#_TOC_250017)
3. [SYSTEM DESIGN 7](#_TOC_250016)
   1. [DEFINITION 7](#_TOC_250015)
   2. [OUTPUT DESIGN 8](#_TOC_250014)
   3. [INPUT DESIGN 9](#_TOC_250013)
   4. [LOGICAL DESIGN 10](#_TOC_250012)
4. [OVERVIEW OF LANGUAGE USED 13](#_TOC_250011)
   1. [PYTHON 13](#_TOC_250010)
   2. [JAVA VS. PYTHON IN 2019 14](#_TOC_250009)
   3. [JAVA VS. PYTHON TYPING 15](#_TOC_250008)
5. [SYSTEM SPECIFICATION 54](#_TOC_250007)
   1. [HARDWARE SPECIFICATION 54](#_TOC_250006)
   2. [SOFTWARE SPECIFICATION 55](#_TOC_250005)
6. [TESTING 56](#_TOC_250004)
   1. TESTING STRATEGY 56
      1. Unit Testing 56
      2. Integration Testing 57
      3. Functional Testing 58
      4. System Testing 59
      5. White Box Testing 60
      6. Black Box testing 60
      7. Other Testing Methodologies 61
7. [SCREENSHOTS 62](#_TOC_250003)
8. [PROGRAM 67](#_TOC_250002)
9. [FUTURE ENHANCEMENT 74](#_TOC_250001)
10. [CONCLUSION 75](#_TOC_250000)
11. BIBILOGRAPHY 76

# INTRODUCTION

The Employee Information Management has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardships faced by this existing system. Moreover, this system is designed for the particular need of the company to carry out operations in a smooth and effective manner. The application is reduced as much as possible to avoid errors while entering the data.it also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system.

Thus, by this all it proves it is user-friendly. Employee Information Management, as described above, can lead to error free, secure, reliable and fast management system.it can assist the user to concentrate on their other activities rather to concentrate on the record keeping.

Thus, it will help organization in better utilization of resources. The purpose of Employee Information Management system is to automate the existing manual system by the help of computerized equipment’s and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same.

The required software and hardware are easily available and easy to work with. The aim is to automate its existing manual system by the help of computerized equipment’s and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. Basically, the project describes how to manage for good performance and better services for the clients.

1

# SYSTEM ANALYSIS

#### DEFINITION

System Analysis is the detailed study of the various operations performed by the system and their relationships within and outside the system. Analysis is the process of breaking something into its parts so that the whole may be understood. System analysis is concerned with becoming aware of the problem, identifying the relevant and most decisional variables, analyzing and synthesizing the various factors and determining an optional or at least a satisfactory solution. During this a problem is identified, alternate system solutions are studied and recommendations are made about committing the resources used to the system.

#### DESCRIPTION OF PRESENT SYSTEM

The Employee Information Management have the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardships faced by this existing system

#### LIMITATIONS OF PRESENT SYSTEM

These tasks are time consuming. It may take one month or long. Customer segmentation Analysis using machine learning is very convenient because in the manual system there are lot of difficulties in conducting and managing a Customer segmentation.

#### PROPOSED SYSTEM

Employee Management System provides Employee information management system. Use in presentation of an information system for managing the staff data within a small company or organization. The system as such as it has been developed is called Employee Management System. It consists of functionally related GUI (application program) and database.

#### ADVANTAGES

The aim is to automate its existing manual system by the help of computerized equipment’s and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. Basically, the project describes how to manage for good performance and better services for the clients.. It resolves typical issues of manual examination processes and activities into a controlled and closely monitored work flow in the architecture of the application. This multi platform solution brings in by default, the basic intelligence and immense possibilities for further extension of the application as required by the user. The system makes it friendly to distribute, share and manage the examination entities with higher efficiency and easiness.

#### FEASIBILITY STUDY

A feasibility analysis usually involves a through assessment of the operational (need), financial and technical aspects of a proposal. Feasibility study is the test of the system proposal made to identify whether the user needs may be satisfied using the current software and hardware technologies, whether the system will be cost effective from a business point of view and whether it can be developed with the given budgetary constraints. A feasibility study should be relatively cheap and done at the earliest possible time. Depending on the study, the decision is made whether to go head with a more detailed analysis.

When a new project is proposed, it normally goes through feasibility assessment. Feasibility study is carried out to determine whether the proposed system is possible to develop with available resources and what should be the cost consideration. Facts considered in the feasibility analysis were

* + - Technical Feasibility
    - Economic Feasibility
    - Behavioral Feasibility

#### TECHNICAL FEASIBILITY

Technical feasibility includes whether the technology is available in the market for development and its availability. The assessment of technical feasibility must be based on an outline design of system requirements in terms of input, output, files, programs and procedures. This can be qualified in terms of volumes of data, trends, frequency of updating, cycles of activity etc, in order to give an introduction of technical system. Considering our project it is technical feasible. Customer segmentation Analysis and Customer segmentation Analysis Systems, with its emphasis on a more strategic decision making process is fast gaining ground as a popular outsourced function.

#### ECONOMIC FEASIBILITY

This feasibility study present tangible and intangible benefits from the project by comparing the development and operational cost. The technique of cost benefit analysis is often used as a basis for assessing economic feasibility. This system needs some more initial investment than the existing system, but it can be justifiable that it will improve quality of service.

Thus feasibility study should center along the following points:

* Improvement resulting over the existing method in terms of accuracy, timeliness.
* Cost comparison
* Estimate on the life expectancy of the hardware.
* Overall objective.

Our project is economically feasible. It does not require much cost to be involved in the overall process. The overall objective is in easing out the Customer segmentation Analysis processes.

#### BEHAVIORAL / OPERATIONAL FEASIBILITY

This analysis involves how it will work when it is installed and the assessment of political and managerial environment in which it is implemented. People are inherently resistant to change and computers have been known to facilitate change. The new proposed system is very much useful to the users and therefore it will accept broad audience from around the world.

# SYSTEM DESIGN

#### DEFINITION

The most creative and challenging face of the system development is System Design. It provides the understanding and procedural details necessary for the logical and physical stages of development. In designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. The first step is to determine how the output is to be produced and in what format. Second, input data and master files have to be designed to meet the requirements of the proposed output. The operational phases are handled through program construction and testing.

Design of the system can be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Thus, system design is a solution to “how to” approach to the creation of a new system. This important phase provides the understanding and the procedural details necessary for implementing the system recommended in the feasibility study. The design step provides a data design, architectural design, and a procedural design.

#### OUTPUT DESIGN

In the output design, the emphasis is on producing a hard copy of the information requested or displaying the output on the CRT screen in a predetermined format. Two of the most output media today are printers and the screen. Most users now access their reports from either a hard copy or screen display. Computer’s output is the most important and direct source of information to the user, efficient, logical, output design should improve the systems relations with the user and help in decision- making.

As the outputs are the most important source of information to the user, better design should improve the systems relations and also should help in decision-making. The output device’s capability, print quality, response time requirements etc should also be considered, form design elaborates the way the output is presented and layout available for capturing information. It’s very helpful to produce the clear, accurate and speedy information for end users.

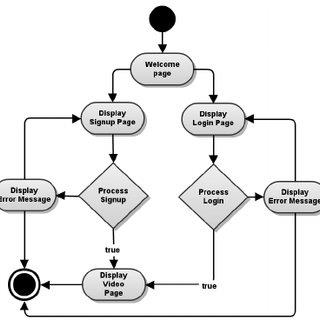
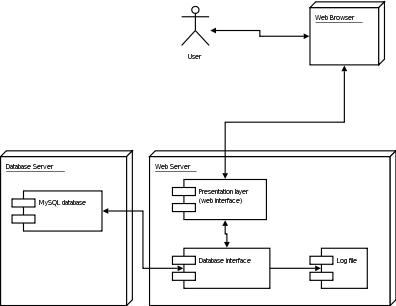
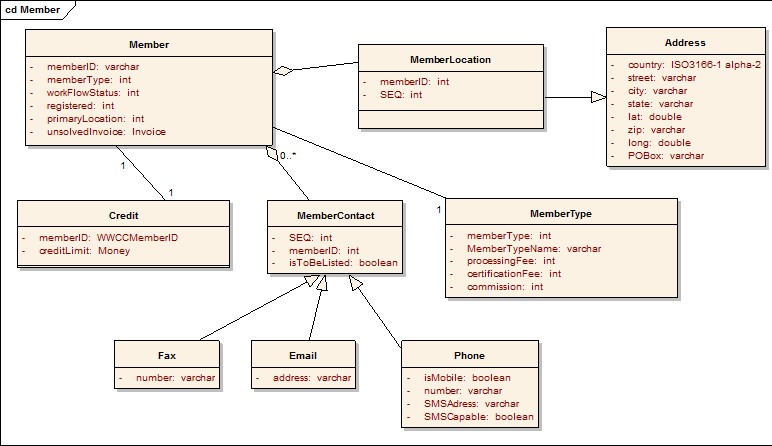
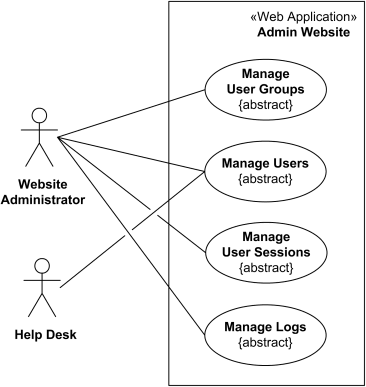
#### INPUT DESIGN

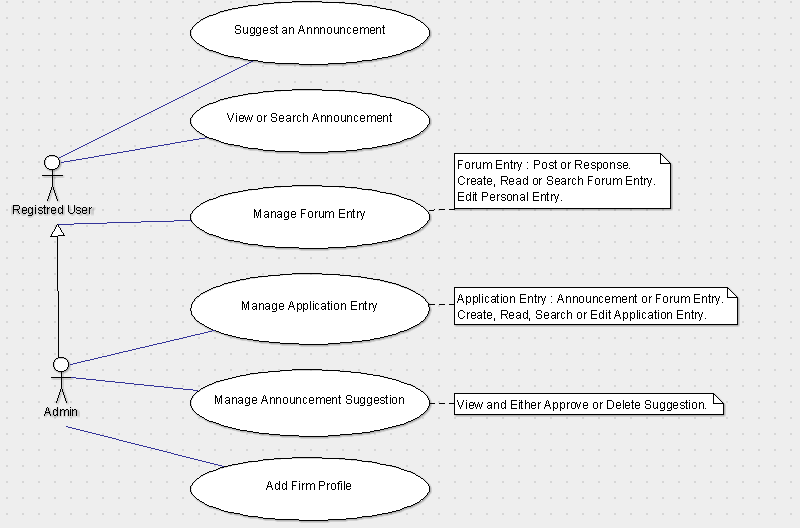
In the input design, user-originated inputs are converted into a computer-based system format. It also includes determining the record media, method of input, speed of capture and entry on to the screen. Online data entry accepts commands and data through a keyboard. The major approach to input design is the menu and the prompt design. In each alternative, the user’s options are predefined. The data flow diagram indicates logical data flow, data stores, source and destination. Input data are collected and organized into a group of similar data once identified input media are selected for processing.

In this software, importance is given to develop Graphical User Interface (GUI), which is an important factor in developing efficient and user friendly software. For inputting user data, attractive forms are designed. User can also select the desired options from the menu, which provides all possible facilities. Also the important input format is designed in such a way that accidental errors are avoided. The user has to input only just the minimum data required, which also helps in avoiding the errors that the users may make. Accurate designing of the input format is very important in developing efficient software. The goal of input design is to make entry as easy, logical and free from errors.

#### LOGICAL DESIGN

Logical data design is about the logically implied data. Each and every data in the form can be designed in such a manner to understand the meaning. Logical data designing should give a clear understanding & idea about the related data used to construct a form.





SYSTEM SPECIFICATION

## HARDWARE SPECIFICATION

CPU : CORE I3/I5/I7 PROCESSOR SPEED : 2 GHz COPROCESSOR : BUILT IN

TOTAL RAM : 8 GB

HARD DISK : 500 GB

KEYBOARD : 105 KEYS

MOUSE : LOGITECH MOUSE

DISPLAY : SGVA COLOR

## SOFTWARE SPECIFICATION

TECHNOLOGY : MACHINE LEARNING PROGRAMMING LANGUAGE : PYTHON

DISTRIBUTION : ANACONDA

OPERATING SYSTEM : WINDOWS 8 AND ABOVE

# TESTING

It is a process of establishing confidence that a program or system does what it is proposed of. Testing is the only way to assure the quality of software and it is an umbrella activity rather than a separate phase. This is an activity to be performed in parallel with the software effort and one that consists of its own phases of analysis, design, implementation, execution and maintenance.

#### TESTING STRATEGY

UNIT TESTING:

This testing method considers a module as single unit and checks the unit at interfaces and communicates with other modules rather than getting into details at statement level. Here the module will be treated as a black box, which will take some inputs and generate output. Outputs for a given set of input combination are pre- calculated and are generated by the module.

INTEGRATION TESTING:

Here all the pre-tested individual modules will be assembled to create the larger system and tests are carried out at system level to make sure that all modules are working in synchronous with each other. This testing methodology helps in making sure that all modules which are running perfectly when checked individually and are also running cohesion with other modules. For this testing we create test cases to check all modules once and then generated test combinations of test paths through out the system to make sure that no path is making its way into chaos.

### VALIDATION TESTING

Testing is a major quality control measure employed during software development. Its basic function is to detect errors. Sub functions when combined may not produce than it is desired. Global data structures can represent the problems.

Integrated testing is a systematic technique for constructing the program structure while conducting the tests. To uncover errors that are associated with interfacing the objective is to make test modules and built a program structure that has been detected by design. In a non-incremental integration all the modules are combined in advance and the program is tested as a whole. Here errors will appear in an endless loop function. In incremental testing the program is constructed and tested in small segments where the errors are isolated and corrected.

Different incremental integration strategies are top-down integration, bottom- up integration, regression testing.

### HIGH-ORDER TESTING (A.K.A. SYSTEM TESTING)

Modules are integrated by moving downwards through the control hierarchy beginning with main program. The subordinate modules are incorporated into structure in either a Breadth First manner or in a Depth First manner.

**THIS PROCESS IS DONE IN FIVE STEPS :**

* Main control module is used as a test driver and steps are submitted are all modules directly to main program.
* Depending on the integration approach selected subordinate is replaced at a time with actual modules.
* Tests are conducted.
* On completion of each set of tests another stub is replaced with the real module.
* Regression testing may be conducted to ensure that new errors have not been introduced.

This process continues from step 2 until entire program structure is reached. In top down integration strategy decision making occurs at upper levels in the hierarchy and is encountered first. If major control problems do exists early recognition’s is essential.

If Depth First integration is selected a complete function of the software may be implemented and demonstrated.

Some problems occur when processing at low levels in hierarchy is required to adequately test upper level steps to replace low-level modules at the beginning of the top-down testing. So no data flows upwards in the program structure.

**BOTTOM-UP INTEGRATION TESTING**

Begins construction and testing with automatic modules. As modules are integrated from the bottom-up, processing requirement for modules subordinate to a given level is always available and need for stubs is eliminated.

#### THE FOLLOWING STEPS IMPLEMENT THIS STRATEGY:

* + Low-level modules are combined in to clusters that perform a specific software sub function.
  + A driver is written to coordinate test case input and output.
  + Cluster is tested.
  + Drivers are removed and moving upward in program structure combines clusters.

Integration moves upward, the need for separate test drover’s lesions. If the top-levels of the program are integrated top-down, the number of drivers can be reduced substantially and integration of clusters is greatly simplified.

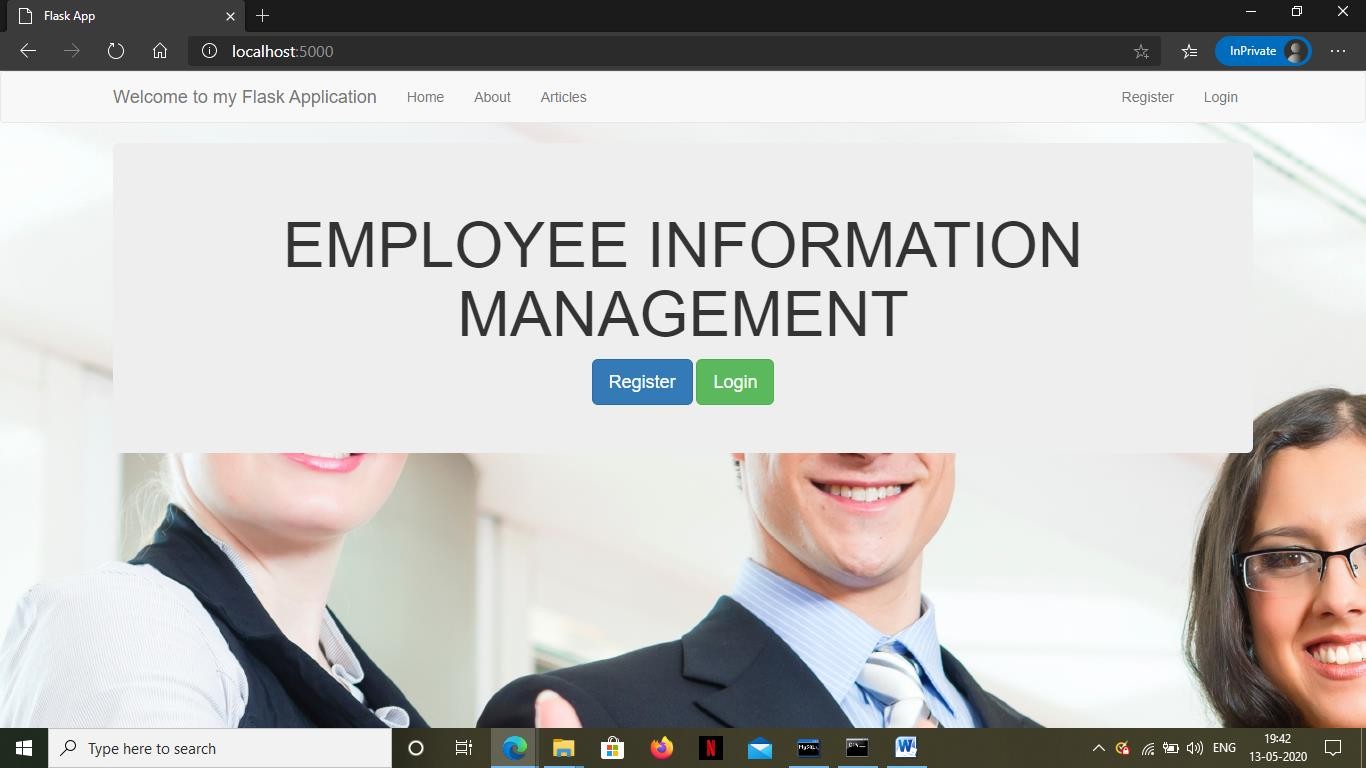
**REGRESSION TESTING**

Each time a new module is added as a part of integration as the software changes. Regression testing is an actually that helps to ensure changes that do not introduce unintended behavior as additional errors.

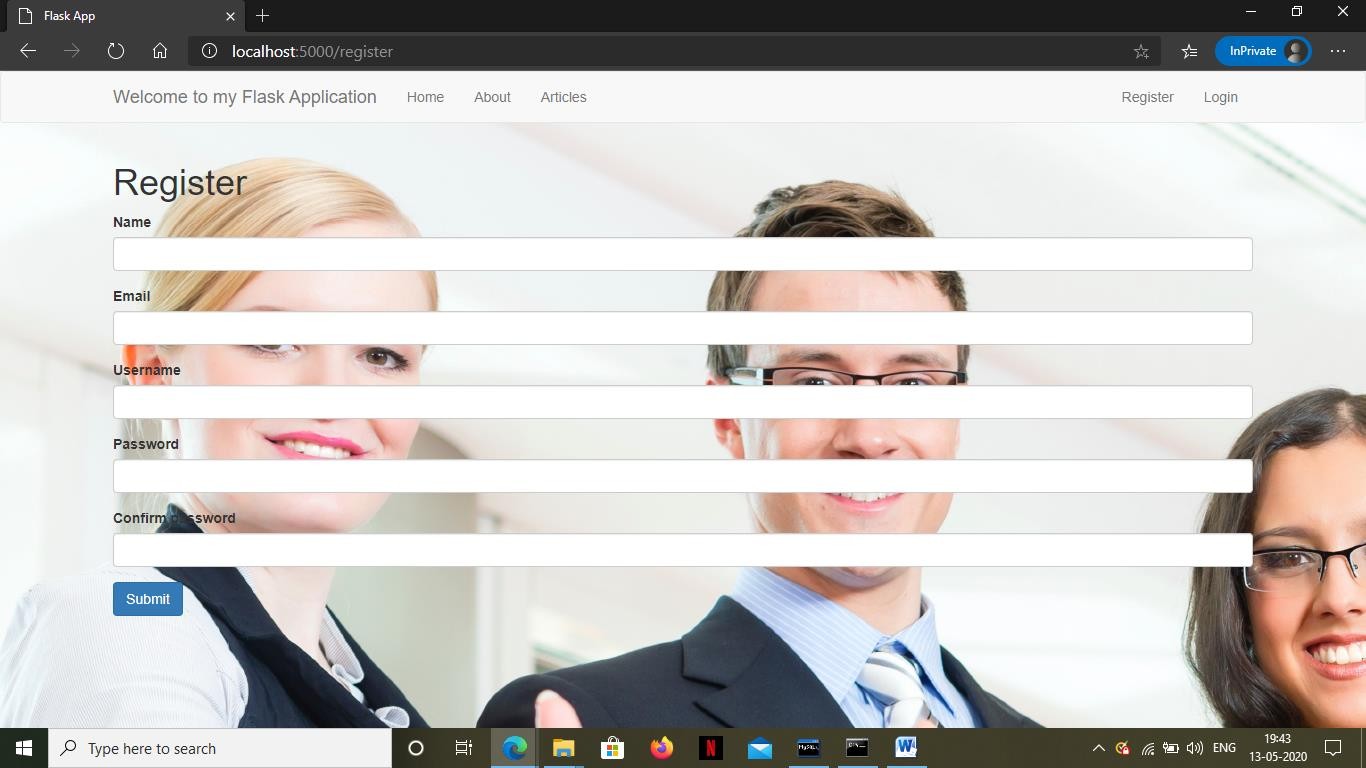
Regression testing may be conducted manually by executing a subset of all test cases and results for subsequent playback tools enables the software engineer to capture the test case and results for subsequent playback and compression. The regression suit contains different classes of test cases.

# SCREENSHOTS

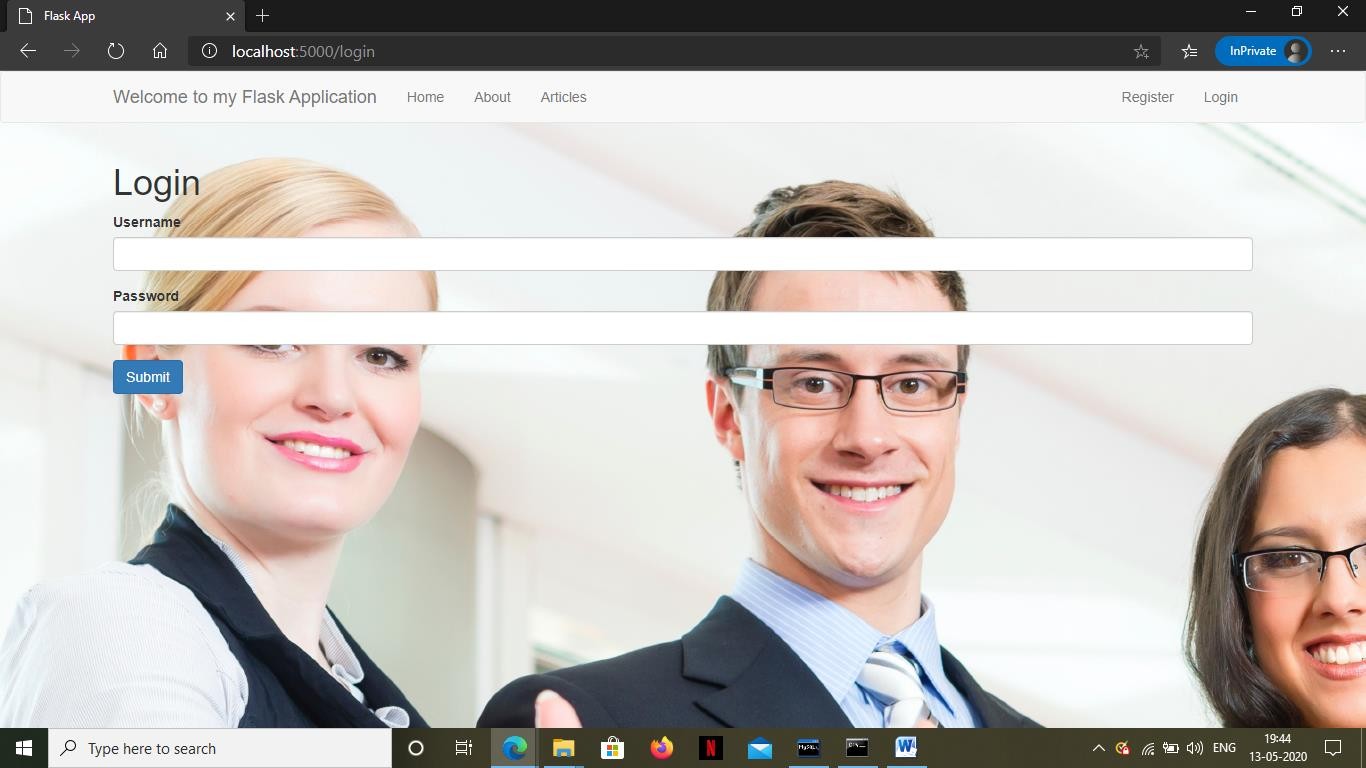
**HOME PAGE:**



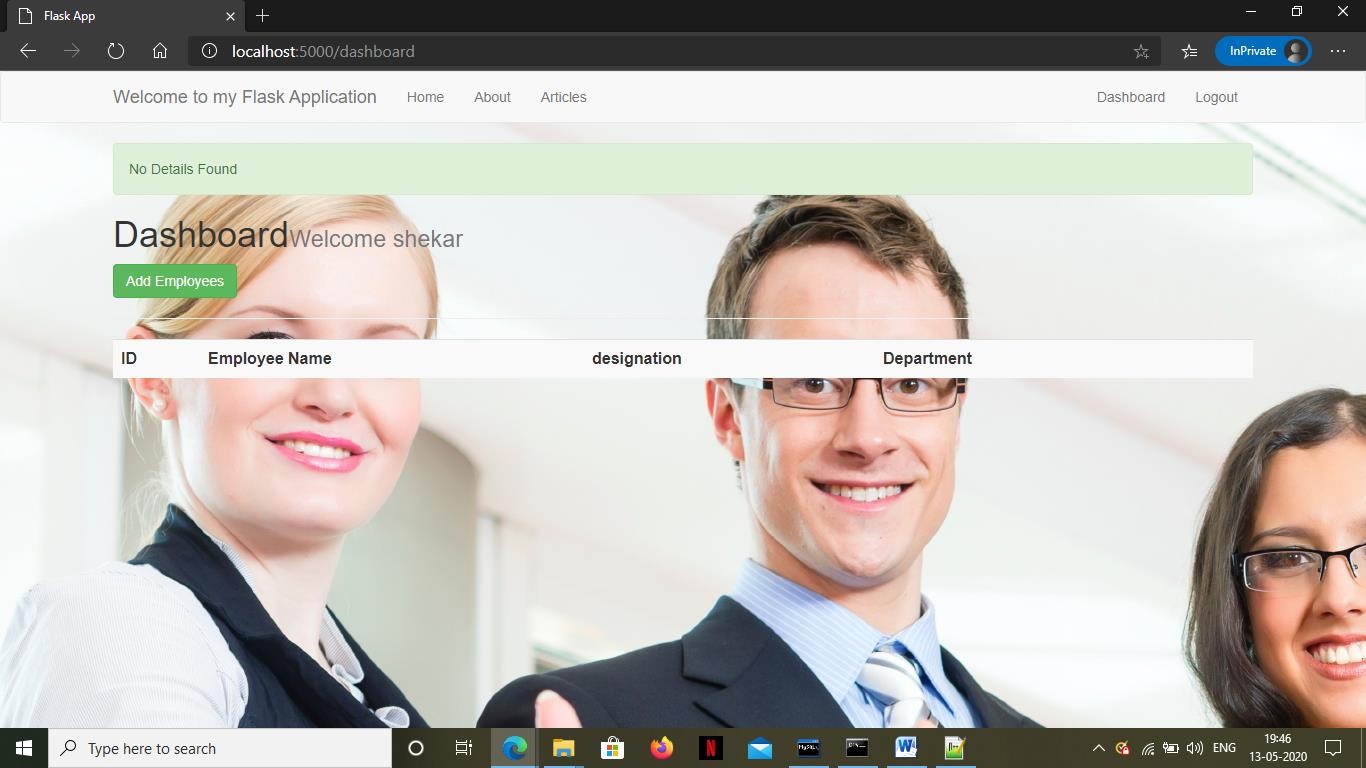
**REGISTER PAGE:**



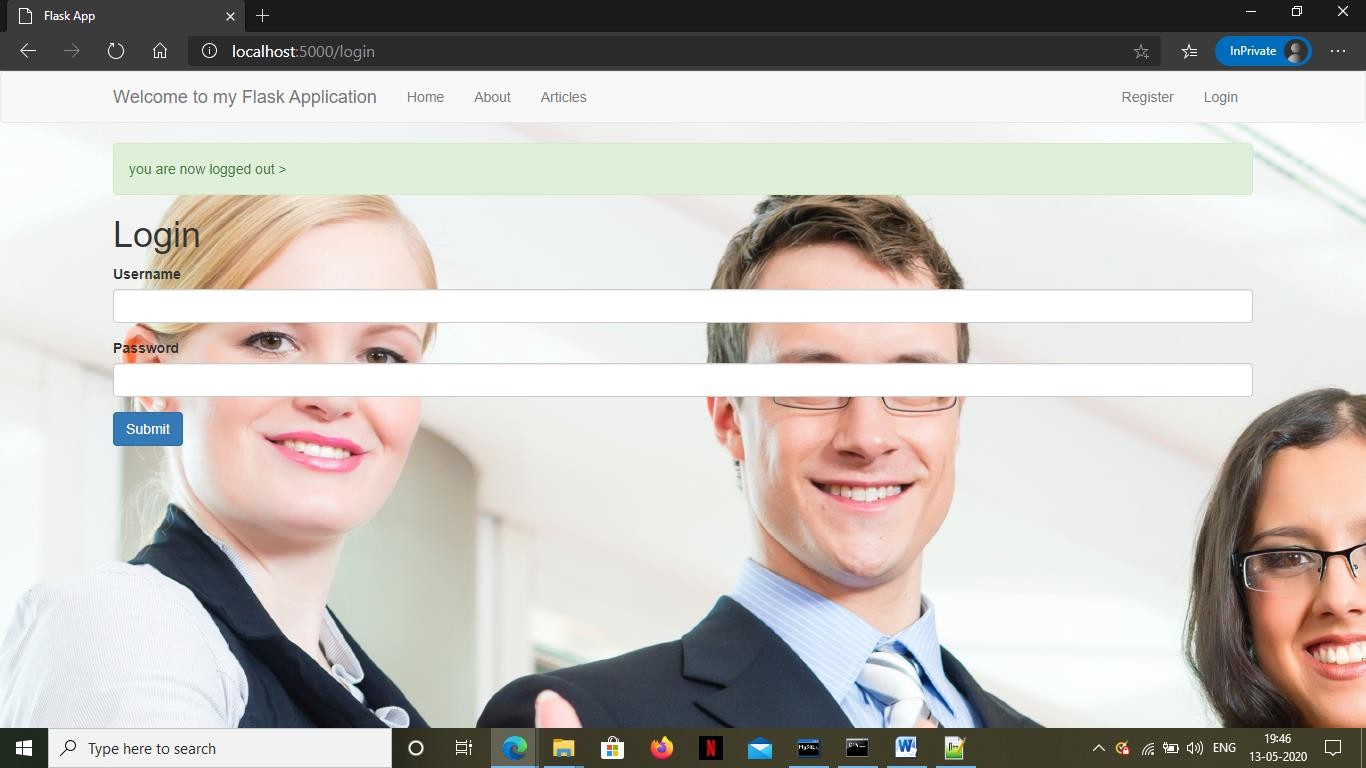
**LOGIN PAGE:**



**DASHBOARD:**



**LOGOUT PAGE:**



# PROGRAM

**//app.py**

from flask import Flask,render\_template, flash, redirect , url\_for , se ssion ,request, logging

from flask\_mysqldb import MySQL

from wtforms import Form, StringField , TextAreaField ,PasswordField , validators

from passlib.hash import sha256\_crypt from functools import wraps

app = Flask( name ) app.debug = True app.secret\_key='ilsttatmt1@'

#Config MySQL

app.config['MYSQL\_HOST'] = 'localhost' app.config['MYSQL\_USER'] = 'root' app.config['MYSQL\_PASSWORD'] = 'root' app.config['MYSQL\_DB'] = 'myflaskapp' app.config['MYSQL\_CURSORCLASS'] = 'DictCursor' #init MYSQL

mysql = MySQL(app)

#Articles = Articles () @app.route('/')

def index():

return render\_template('home.html')

@app.route('/about') def about():

return render\_template('about.html')

@app.route('/articles') def articles():

#create cursor

cur = mysql.connection.cursor()

#get articles

result = cur.execute("SELECT \* FROM articles") articles = cur.fetchall()

if result > 0:

return render\_template('articles.html',articles=articles) else:

msg = 'No Employees Found'

return render\_template('articles.html',msg=msg) #close connection

cur.close()

@app.route('/article/<string:id>/') def article(id):

#create cursor

cur = mysql.connection.cursor()

#get article

result = cur.execute("SELECT \* FROM articles WHERE id = %s",[id]) #result = cur.execute("SELECT \* FROM articles")

article = cur.fetchone()

return render\_template('article.html',article=article)

class RegisterForm(Form):

name = StringField('Name',[validators.Length(min=1,max=50)]) username = StringField('Username',[validators.Length(min=4,max=25)]

)

email = StringField('Email',[validators.Length(min=4,max=25)]) password = PasswordField('Password', [ validators.DataRequired (),v

alidators.EqualTo('confirm',message ='passwords do not match')]) confirm = PasswordField('Confirm password')

@app.route('/register', methods=['GET','POST']) def register():

form = RegisterForm(request.form)

if request.method == 'POST' and form.validate(): name = form.name.data

email = form.email.data username = form.username.data

#password = sha256\_crypt.encrypt(str(form.password.data)) password = str(form.password.data)

# Create crusor

cur = mysql.connection.cursor()

cur.execute("INSERT INTO users(name,email,username,password) VA LUES(%s,%s,%s,%s)",(name,email,username,password))

# commit to DB mysql.connection.commit() #close connection cur.close()

flash("You are now Registered and you can login" , 'success') redirect(url\_for('login'))

return render\_template('register.html',form=form)

# user login

@app.route('/login',methods =['GET','POST']) def login():

if request.method == 'POST': #Get Form Fields

username = request.form['username'] password\_candidate = request.form['password']

# Create cursor

cur = mysql.connection.cursor() #Get user by username

result = cur.execute("SELECT \* FROM users WHERE username = %s"

,[username])

if result > 0:

# Get Stored hash

data = cur.fetchone() password = data['password']

# Compare Passwords

#if sha256\_crypt.verify(password\_candidate,password): if password==password\_candidate:

#Passed session['logged\_in'] = True

session['username'] = username

flash('You are now logged in ','success') return redirect(url\_for('dashboard'))

else:

error = 'Username not found'

return render\_template('login.html',error=error) #close connection

cur.close()

else:

error = 'Username not found'

return render\_template('login.html',error=error) return render\_template('login.html')

#check if user logged in

def is\_logged\_in(f): @wraps(f)

def wrap(\*args,\*\*kwargs):

if 'logged\_in' in session: return f(\*args, \*\*kwargs)

else:

flash('Unauthorized, please login','danger') return redirect(url\_for('login'))

return wrap

#logout @app.route('/logout') @is\_logged\_in

def logout():

session.clear()

flash('you are now logged out ','success') return redirect(url\_for('login'))

# Dashboard @app.route('/dashboard') @is\_logged\_in

def dashboard():

#create cursor

cur = mysql.connection.cursor() #result=0

#get articles

result = cur.execute("SELECT \* FROM articles") articles = cur.fetchall()

if result > 0:

return render\_template('dashboard.html',articles=articles) else:

msg = 'No Details Found'

return render\_template('dashboard.html',msg=msg) #close connection

cur.close()

class ArticleForm(Form):

title = StringField('Employee Name',[validators.Length(min=1,max=50

)])

])

)

body = TextAreaField('Designation',[validators.Length(min=1,max=50) author = StringField('Department',[validators.Length(min=1,max=50)]

#Add Article

@app.route('/add\_article', methods=['GET','POST']) @is\_logged\_in

def add\_article():

form = ArticleForm(request.form)

if request.method == 'POST' and form.validate(): title = form.title.data

body = form.body.data author=form.author.data

# Create a cursor

cur = mysql.connection.cursor() #execute

cur.execute("INSERT INTO articles(title,body,author) VALUES(%s,

%s, %s)",(title, body,author)) #commit to db mysql.connection.commit()

#close connection cur.close()

flash('created ','success')

return redirect(url\_for('dashboard'))

return render\_template('add\_article.html',form=form) #Edit Article

@app.route('/edit\_article/<string:id>', methods=['GET','POST']) @is\_logged\_in

def edit\_article(id): # Create cursor

cur = mysql.connection.cursor() #get article by id

result = cur.execute("SELECT \* FROM articles WHERE id = %s", [id]) article = cur.fetchone()

#get form

form = ArticleForm(request.form)

#populate article form fields form.title.data = article['title'] form.body. data = article['body'] form.author. data = article['author']

if request.method == 'POST' and form.validate(): title = request.form['title']

body = request.form['body'] author = article['author']

# Create a cursor

cur = mysql.connection.cursor() #execute

cur.execute("UPDATE articles SET title=%s, body=%s WHERE id = % s" , (title,body,id))

#commit to db mysql.connection.commit()

#close connection cur.close()

flash('Updated ','success')

return redirect(url\_for('dashboard'))

return render\_template('edit\_article.html',form=form) #Delete article

@app.route('/delete\_article/<string:id>', methods=['POST']) @is\_logged\_in

def delete\_article(id): # Create cursor

cur = mysql.connection.cursor()

#Execute

cur.execute("DELETE FROM articles WHERE id = %s",[id]) #Commit to DB

mysql.connection.commit() #close connection

cur.close()

flash('Deleted ','success')

return redirect(url\_for('dashboard'))

if name ==' main ': app.secret\_key='ilsttatmt1@' app.config['SESSION\_TYPE'] = 'filesystem' sess.init\_app(app)

app.debug = True app.run()

# FUTURE ENHANCEMENT

### In wake of the new and related trends, it is imperative for frequent upgrades to a new models and algorithms to make it easier according to address new business needs.

# CONCLUSION

The project has been successfully completed. The goal of the system is achieved and the problems are solved. This project is developed in this manner that is user friendly and required help is provided at different levels

# BIBLIOGRAPHY

#### REFERENCES

1. Paul K. Asabere and Forrest E. Huffman. “Price Concessions, Time of the Market, and the Actual Sale Price of Homes”. In: Journal of Real Estate Finance and Economics 6

(1993), pp. 167–174. URL: https://link.springer.com/article/10.1007/ BF01097024.

1. Leo Breiman. “Random forests”. In: Machine learning 45.1 (2001), pp. 5–32.
2. Rochard J. Cebula. “The Hedonic Pricing Model Applied to the Housing Market of

the City of Savannah and Its Savannah Historic Landmark District”. In: The Review of Regional Studies 39.1 (2009), pp. 9–22. URL: journal.srsa.org/ojs/index. php/rrs/article/download/182/137.

1. Consumer Housing Trends Report 2016. Zillow Group. Accessed: 11/10/2017. 2016.

URL: https : / / www . zillow . com / research / zillow - group - report - 2016-13279/.

1. Harris Drucker et al. “Support vector regression machines”. In: Advances in neural

information processing systems. 1997, pp. 155–161.

1. Gang-Zhi Fan, Seow Eng Ong, and Hian Chye Koh. “Determinants of House Price:

A Decision Tree Approach”. In: Urban Studies 43.12 (2006), pp. 2301–2315. URL: journals.sagepub.com/doi/pdf/10.1080/00420980600990928.

1. Tin Kam Ho. “Random decision forests”. In: Document analysis and recognition, 1995.,

proceedings of the third international conference on. Vol. 1. IEEE. 1995, pp. 278– 282.

1. Daniel R. Hollas, Ronald C. Rutherford, and Thomas A. Thomson. “Zillow’s estimates of single-family housing values.” In: Expert Systems with Applications 78.1 (2010). URL[: http://www.freepatentsonline.com/article/Appraisal-](http://www.freepatentsonline.com/article/Appraisal-) Journal/220765044.html.
2. Gu Jirong, Zhu Mingcang, and Jiang Liuguangyan. “Housing price based on genetic

algorithm and support vector machine”. In: Expert Systems with Applications 38 (2011), pp. 3383–3386. UR[L: http://www.sciencedirect.com/science/](http://www.sciencedirect.com/science/) article/pii/S0957417410009310.

1. Kelvin J. Lancaster. “A New Approach to Consumer Theory”. In: The Journal of Political

Economy 74.2 (1966), pp. 132–157. ISSN: 0303-2647. DOI: 10.1.1.456.4367&

rep=rep1&type=pdf. URL[: http://www.jstor.org/stable/1828835.](http://www.jstor.org/stable/1828835)

1. Number of houses sold in the United States from 1995 to 2016. [www.statista.com](http://www.statista.com/). Accessed:

11/10/2017. URL: htt[ps://www](http://www.statista.com/statistics/219963/).sta[tista.com/statistics/219963/](http://www.statista.com/statistics/219963/) number-of-us-house-sales/.

1. Quick Facts: Resident Demographics. National Multifamily Housing Council. Accessed:

11/11/2017. 2017. URL: [http://www.nmhc.org/Content.aspx?id=4708.](http://www.nmhc.org/Content.aspx?id=4708)

1. Random Forests by Leo Breiman and Adele Cutler. URL: htt[ps://www](http://www.stat.berkeley/).sta[t.berk](http://www.stat.berkeley/)e[ley.](http://www.stat.berkeley/)

edu/˜breiman/RandomForests/.

1. Sherwin Rosen. “Hedonic Prices and Implicit Markets: Product Differentiation in Pure Competition”. In: The Journal of Political Economy 82.1 (1974), pp. 34–55. URL:

[http://people.tamu.edu/˜](http://people.tamu.edu/)ganli/publicecon/rosen74.pdf.

1. Hasan Selim. “Determinants of house prices in Turkey: Hedonic regression versus

artificial neural network”. In: Expert Systems with Applications 36 (2009), pp. 2843– 2852. URL: [www.sciencedirect.com/science/article/pii/S0957417408000596.](http://www.sciencedirect.com/science/article/pii/S0957417408000596)

1. G. Stacy Sirmans, David A. Macpherson, and Emily N. Zietz. “The Composition of

Hedonic Pricing Models”. In: Journal of Real Estate Literature 13.1 (2005), pp. 3–43.

URL[: http://www.jstor.org/stable/44103506?seq=1#page\_scan\_tab\_](http://www.jstor.org/stable/44103506?seq=1&page_scan_tab_) contents.

1. Alex J Smola and Bernhard Sch¨olkopf. “A tutorial on support vector regression”.

In: Statistics and computing 14.3 (2004), pp. 199–222.

1. Danny P. H. Tay and David K. H. Ho. “Artificial Intelligence and the Mass Appraisal of Residential Apartments”. In: Journal of Property Valuation and Investment

10.2 (1992), pp. 525–540. UR[L: http://www.emeraldinsight.com/doi/abs/](http://www.emeraldinsight.com/doi/abs/) 10.1108/14635789210031181.

1. The Price of Overpricing: How Listing Price Impacts Time on Market. Zillow. Accessed: 03/06/2018. 2016. URL: htt[ps://www.zillow.com/research/ove](http://www.zillow.com/research/overpricingimpacts-)rpr[icingimpact](http://www.zillow.com/research/overpricingimpacts-)s-

time-market-12476/.

1. Trulia Estimate. Trulia. Accessed: 11/11/2017. 2017. URL: https://www.trulia. com/trulia\_estimates/.
2. Vladimir N. Vapnik and Alexey Ya Chervonenkis. “On a class of algorithms of learning pattern recognition.” In: Automation and Remote Control 25.6 (1964).