

Best Projects Groups Year 2012-13

Sr. No.	Name of Students	Title	Area	Sponsoring Company
1	KULKARNI KALYANI N PATIL AKANKSHA SHAHA NIKITA VARPE JYOTI	Sysytem Call Analyze	System Programming, Linux	In-house
2	DIKSHA RAKWAL JAAE APTE TOPLE MITHILA ZAINAB KINKHABWALA	Indian sign language Detection using Microsoft Kinet	Application Development, AI	Persistent Systems Ltd.
3	HEBLE SOWMYA JOSHI NOOPUR KHANDEKAR MANGALA KULKARNI SAMHITA	Cloud Storage Provisioning Using Openstack-Swift	Cloud Computing, Openstack	Persistent Systems Ltd.
4	ISHA BHARTI JYOTI KUMAR KULKARNI KALYANI S KULKARNI RAJASI	Virtual Store Management	Mobile Application Development, Android, Java	Persistent Systems Ltd.
5	PATIL KSHIPRA PATIL SNEHA PHALKE NISHIGANDHA RANAWARE NEHA	Expiry based file Shares	Linux Kernel, FileSystems	Calsoft

System Call Analyser

Abstract:

We present System call analyzer, the best possible solution for minimizing this burden of kernel level programmers. The tool develops useful statistics and analysis of system calls and signals to help the system programmer. Statistics of all important system calls and signals according to user requirements are generated in the tabular as well as graphical formats. Some comparative studies like percentage error, time requirement of different system calls, input/output statistics, open/close statistics of files are also carried out to find out the efficiency and performance of system. System call analyzer traces and analyzes the interactions between kernel and different application programs of Linux and provides the useful statistics in graphical and user readable format. This will help kernel level users to better understand and to go to the root of the problems. Our tool also gives guidelines to normal programmers to increase the efficiency of his program by analyzing input/output interactions of the program.

ANUVAADAK- Indian sign language Detection using Microsoft Kinet

ABSTRACT

ANUVAADAK is an application to recognize the gestures of Indian Sign Language (ISL) using the Microsoft Kinect for Xbox. Our system has two modes - **Recognition** and **Tutorial**. The **Recognition** mode identifies the gesture of the user using the Hidden Markov Model (HMM) and translates the output in text and speech format with fairly accurate results. While the **Tutorial** mode has video lessons which helps the user to learn ISL. Also the user can perform that gesture and system will judge his accuracy. We offer a portable and a scalable application with an easy-to-use and interactive gesture based GUI. Our system aims to bridge the gap between people who use ISL and those who do not. The primary aim is to enable people with speech and hearing disorders to communicate with others.

Cloud Storage Provisioning System Using OpenStack Swift

ABSTRACT

Cloud computing delivers hardware and software computing resources as a service over network, typically the Internet. Infrastructure as a Service (IaaS) is cloud service model which offers computers as virtual machines and other resources like servers, storage, and network. Cloud storage is a model of networked online storage where data is stored in virtualized pools of storage- hosted by cloud providers.

Our project *Cloud Storage Provisioning System-SwiftStore Using OpenStack Swift* aims at developing cloud storage platform that focuses on high availability and disaster recovery. This platform can be accessed from web browser and storage account is mapped as a folder on the desktop. User can share and give controlled access to the files. User can select different storage plans. Administrator can monitor overall and individual usage and generate usage statistics. This project will be developed using OpenStack which is an Open Source cloud operating system. OpenStack controls large pools of compute, storage, and networking resources throughout the datacenter. OpenStack Swift is the Cloud Storage Project using clusters of servers capable of storing petabytes of data for highly available, distributed, consistent, scalable and cost-effective object store. Even though OpenStack Swift Storage has these features most end users won't use it directly because the interface is too primitive. That interface is designed for developers who can write programs to leverage the basic reading, writing and deletion of objects. Instead, end users need an access platform for OpenStack Swift storage that covers their basic needs of backup, access and controlled sharing. SwiftStore provides an access platform that satisfies these needs. SwiftStore provides a user-friendly access platform so that users can avail the benefits of OpenStack swift to get Storage-As-A-Service.

Thus this project harnesses the power of OpenStack Swift Object Storage to develop a cloud storage platform for high availability and disaster recovery.

Abstract

Virtual shopping is simulation of a shopping mall environment. A virtual store can be set up at public places such as metro stations by plastering the walls at public places with actual images of racks of shopping malls along with QR code of each item. Customers need to scan the QR code of the products they want to order using their Smartphone. People spend a considerable amount of time at metro stations or bus stations waiting for their train or bus to arrive. The time wasted in waiting can be utilized for shopping of daily necessities like vegetables or grocery. The products purchased are delivered at the customer's home within few hours.

We have developed an android Mobile application for users to shop at virtual store. When a customer scans the QR code of the product he wants to order, the details of the product stored in the QR code will be displayed on his Smartphone. Customer can confirm his order and make payment using PayPal or using the cash on delivery facility.

We have also developed a web application for enterprise management. Enterprise comprises of several stores distributed across the city. An order placed by the customer is fulfilled by the store which is nearest to his address of delivery. There are two types of users of web application. First is a super store administrator and the other is a store administrator. Each store administrator is a user of the web application which he will use for managing orders of the customers and stock of his store. Super store administrator will use the web application for managing all the stores, costing of the enterprise, complaint processing and employee management.

Expiry based file Shares

ABSTRACT

Enterprises want to empower their employees with greater information access, increased team collaboration, improved productivity and employee engagement. Enterprise employees are increasingly demanding better, easier and more efficient file sharing systems. But there are certain shared documents that are time sensitive or time-limited and must not be accessible (viewable or printable) after a certain time. Our system ensures that users can only view a document for a predefined time. This feature also ensures that only the authorized or current versions of documents are used after sharing because obsolete versions can no longer be accessed.

Existing file systems are capable of sharing the files without providing the facility of unsharing the file as per sharer's convenience. So the server who is sharing the file has to deal with complexities like remembering the files that are shared while unsharing. There are certain files sharing platforms which provide expiry based file sharing. They are created as user level daemon which continuously keeps checking expiry of file and because of polling it unnecessarily increases disk I/O.

Expiry based file sharing proposes solution to above problem in the form of filter which is created in kernel space. The expiry time is accepted from the sharer and file is unshared based on subsequent access after expiry. The system also provides security by restricting modification of expiry time only by the owner of shared file. Since system operates at kernel level overhead due to context switching and memory copies between the kernel and the user-space are reduced. Also wastage of CPU cycles due to polling is avoided.

Thus the system proposes an efficient and secured way of unsharing file after a predefined expiry time.