TRUNG TÂM IVIETTECH

Address: 92 - Quang Trung - Tp. Đà Nẵng- Tel. 05113 888 279

Website: http://iviettech.vn, E-mail: [contact@iviettech.vn](mailto:contact@viettech.edu.vn)



LUẬN VĂN TỐT NGHIỆP

LỚP LẬP TRÌNH DI ĐỘNG IOS

**ĐỀ TÀI:**

**XÂY DỰNG ỨNG DỤNG HỖ TRỢ ĂN DẶM CHO BÉ**

**NGÀY BẢO VỆ: 28/09/2014**

HỌC VIÊN : ĐOÀN THANH HẢI

NGUYỄN THỊ ANH ĐÀO

GIÁO VIÊN HƯỚNG DẪN : NGUYỄN THANH LIÊM

ĐÀ NẴNG, 09/2014

LỜI CẢM ƠN

Bài báo cáo được hoàn thành dưới sự hướng dẫn và chỉ bảo tận tình của thầy Nguyễn Thanh Liêm. Em xin chân thành cảm ơn sự chỉ bảo của thầy.

Nhóm cũng rất cảm ơn các thầy cô, nhân viên của trung tâm Iviettech đã tận tình giảng dạy, truyền đạt những kiến thức quý báu và tạo điều kiện cho em hoàn thành báo cáo này.

Xin chân thành cảm ơn sự giúp đỡ, động viên của tất cả các bạn trong quá trình thực hiện báo cáo, cám ơn những anh chị trong công ty đã cùng làm việc với nhóm tạo điều kiện để em thực hiện đồ án.

Nhóm cũng muốn cảm ơn những người thân trong gia đình đã động viên giúp đỡ tạo điều kiện để em hoàn thành đồ án này.

Mặc dù đã cố gắng hết sức hoàn thành đồ án với tất cả sự nổ lực của bản thân, nhưng đồ án không tránh khỏi những thiếu sót. Nhóm rất mong nhận được sự thông cảm và chỉ bảo tận tình của các thầy cô và các bạn.

Đà Nẵng, Ngày 24 tháng 09 năm 2014

Nhóm học viên thực hiện

Đoàn Thanh Hải

Nguyễn Thị Anh Đào

**LỜI CAM ĐOAN**

*Nhóm học viên xin cam đoan :*

* 1. *Những nội dung trong luận văn này là do nhóm thực hiện dưới sự hướng dẫn trực tiếp của thầy Nguyễn Thanh Liêm.*
  2. *Mọi tham khảo dùng trong luận văn đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.*
  3. *Mọi sao chép không hợp lệ, vi phạm quy chế đào tạo, hay gian trá, nhóm xin chịu hoàn toàn trách nhiệm.*

MỤC LỤC

[MỞ ĐẦU 1](#_Toc399331178)

[I. Lý do chọn đề tài 1](#_Toc399331179)

[II. Mục đích và ý nghĩa 2](#_Toc399331180)

[III.2. Kết quả dự kiến 3](#_Toc399331181)

[IV. Nội dung đề tài 4](#_Toc399331182)

[CHƯƠNG I: CƠ SỞ LÝ THUYẾT 6](#_Toc399331183)

[I. Tổng quan hệ điều hành iOS 6](#_Toc399331184)

[I.1. Kiến trúc hệ điều hành iOS 7](#_Toc399331185)

[II. Điện toán đám mây và iCloud 10](#_Toc399331186)

[II.1. Tổng quan về điện toán đám mây 10](#_Toc399331187)

[II.2. iCloud 14](#_Toc399331188)

[III.Ngôn ngữ lập trình object-C 17](#_Toc399331189)

[III.1. Giới thiệu 17](#_Toc399331190)

[III.2. Thành Phần 17](#_Toc399331191)

[III.3. Thế mạnh 18](#_Toc399331192)

[IV.Tổng quan về Web service và SOAP 19](#_Toc399331193)

[IV.1. Web Service 19](#_Toc399331194)

[V. Tổng quan PHP và Cakephp framework. 28](#_Toc399331195)

[V.1. Tổng quan ngôn ngữ lập trình PHP 28](#_Toc399331196)

[V.2. Sơ lược về CakePhp 29](#_Toc399331197)

[VI. Công cụ phát triển. 32](#_Toc399331198)

[VI.1. Giới thiệu xCode và iOS SDK 32](#_Toc399331199)

[VII. Hệ thống ứng dụng “Ăn Dặm Cho Bé”. 37](#_Toc399331200)

[VII.1. Giới thiệu 37](#_Toc399331201)

[VII.3. Ưu nhược điểm của ứng dụng ăn dặm trên điện thoại 38](#_Toc399331202)

[VII.4. Các vấn đề khi xây dựng ứng dụng ăn dặm 39](#_Toc399331203)

[CHƯƠNG II: XÂY DỰNG ỨNG DỤNG 40](#_Toc399331204)

[I. Khảo sát tình hình thực tế 40](#_Toc399331205)

[II. Mô tả bài toán 42](#_Toc399331206)

[II.1. Cấu trúc hệ thống 42](#_Toc399331207)

[II.2. Yêu cầu chức năng 43](#_Toc399331208)

[II.3. Yêu cầu phi chức năng 47](#_Toc399331209)

[III.Phân tích và thiết kế chương trình 47](#_Toc399331210)

[III.1. Mô hình Use case 47](#_Toc399331211)

[III.2. Thiết kế giao diện 52](#_Toc399331212)

[III.3. Cơ sở dữ liệu 58](#_Toc399331213)

[III.3.1.Mô hình cơ sở dữ liệu Core Data 59](#_Toc399331214)

[CHƯƠNG III: TRIỂN KHAI ỨNG DỤNG 61](#_Toc399331215)

[I. Hướng dẫn cài đặt 61](#_Toc399331216)

[I.1.1. Yêu cầu tài nguyên 61](#_Toc399331217)

[I.1.2. Công cụ 61](#_Toc399331218)

[II. Minh họa chạy chương trình 61](#_Toc399331219)

[CHƯƠNG IV: KẾT LUẬN 69](#_Toc399331220)

[I. Đánh giá kết quả 69](#_Toc399331221)

[I.1. Kết quả đạt được 69](#_Toc399331222)

[I.2. Hạn chế 69](#_Toc399331223)

[II. Hướng phát triển 70](#_Toc399331224)

[PHỤ LỤC 71](#_Toc399331225)

[TÀI LIỆU THAM KHẢO 1](#_Toc399331226)

**MỤC LỤC HÌNH ẢNH**

[Hình 1: Các thiết bị chạy hệ điều hành di động iOS. 6](#_Toc399335083)

[Hình 2: Kiến trúc iOS 7](#_Toc399335084)

[Hình 3: Điện toán đám mây 11](#_Toc399335085)

[Hình 4: Điện toán đám mây và thiết bị 14](#_Toc399335086)

[Hình 5: Chuyển đổi từ mobile me qua iClound 15](#_Toc399335087)

[Hình 6: Chồng giao thức dịch vụ Web 20](#_Toc399335088)

[Hình 7: Kiến trúc của Web Service 22](#_Toc399335089)

[Hình 8: Tổng quan về WSDL 24](#_Toc399335090)

[Hình 9: Thông điệp sử dụng SOAP 26](#_Toc399335091)

[Hình 10: Cấu trúc của thông điệp theo dạng SOAP 26](#_Toc399335092)

[Hình 11: Mô hình MVC 31](#_Toc399335093)

[Hình 12: xCode và SDK iOS 32](#_Toc399335094)

[Hình 13: Cửa sổ đơn lẽ 33](#_Toc399335095)

[Hình 14: Jump Bar Xcode 34](#_Toc399335096)

[Hình 15: GUI Builder xCode 34](#_Toc399335097)

[Hình 16: Kéo thả kết nối giao diện và mã 35](#_Toc399335098)

[Hình 17: Tự phát sinh sửa lỗi 36](#_Toc399335099)

[Hình 18: Công cụ hỗ trợ 37](#_Toc399335100)

[Hình 19: Số liệu thống kê về tình trạng dinh dưỡng trẻ em dưới 5 tuổi qua các năm (1999-2013) 40](#_Toc399335101)

[Hình 20: Tình hình bú mẹ đến 6 tháng tuổi 41](#_Toc399335102)

[Hình 21: Tỉ lệ phần trăm suy dinh dưỡng năm 2013 41](#_Toc399335103)

[Hình 22: Cấu trúc hệ thống 43](#_Toc399335104)

[Hình 23: Sơ đồ phân rã chức năng của ứng dụng trên iOS 43](#_Toc399335105)

[Hình 24: Sơ đồ phân rã chức năng Web service 45](#_Toc399335106)

[Hình 25: Tác nhân MobileUser 47](#_Toc399335107)

[Hình 26: Tác nhân Nhân viên 48](#_Toc399335108)

[Hình 27: Tác nhân Admin 48](#_Toc399335109)

[Hình 28: Biểu đồ ca sử dụng mức 0 của tác nhân Người dùng 49](#_Toc399335110)

[Hình 29: Ca sử dụng xem, comment và rating công thức 50](#_Toc399335111)

[Hình 30: Ca sử dụng favorite 51](#_Toc399335112)

[Hình 31: Ca sử dụng xem, comment và rating công thức 51](#_Toc399335113)

[Hình 32: Ca sử dụng chỉnh sửa thông tin cá nhân 51](#_Toc399335114)

[Hình 33: Biểu đồ triển khai 52](#_Toc399335115)

[Hình 34: Giao diện prototype chức năng Login 52](#_Toc399335116)

[Hình 35: Giao diện prototype chức năng danh sách công thức 53](#_Toc399335117)

[Hình 36: Giao diện prototype chức năng danh sách Video 54](#_Toc399335118)

[Hình 37: Giao diện prototype chức năng danh sách Lưu ý 55](#_Toc399335119)

[Hình 38: Giao diện prototype chức năng Đổi mật khẩu 56](#_Toc399335120)

[Hình 39: Giao diện prototype chức năng xem thông tin người dùng 57](#_Toc399335121)

[Hình 40: Giao diện prototype chức năng Thông tin 58](#_Toc399335122)

[Hình 41: Mô hình cơ sở dữ liệu quan hệ SQL trên server 59](#_Toc399335123)

[Hình 42: Mô hình cơ sở dữ liệu quan hệ SQL trên server 60](#_Toc399335124)

[Hình 43: Chức năng Login 61](#_Toc399335125)

[Hình 44: chức năng danh sách công thức ăn dặm 62](#_Toc399335126)

[Hình 45: chức năng danh sách công thức ăn dặm yêu thích 63](#_Toc399335127)

[Hình 46: chức năng chi tiết công thức ăn dặm 64](#_Toc399335128)

[Hình 47: Danh sách công thức cần chú ý 65](#_Toc399335129)

[Hình 48: Danh sách video hướng dẫn nấu ăn dặm 66](#_Toc399335130)

[Hình 49: Chức năng thông tin người dùng 67](#_Toc399335131)

[Hình 50: Chức năng đổi mật khẩu 68](#_Toc399335132)

### MỞ ĐẦU

## I. Lý do chọn đề tài

Trong xã hội hiện nay, việc sở hữu một Smart Phone không còn là một điều quá khó khăn. Việc sử dụng SmartPhone có rất nhiều ích lợi từ việc các gói cước tốt hơn, đặc biệt là kết nối 3G ( 4G ở một số nước). Đây là điều kiện tốt giúp khách hàng thoải mái truy cập Internet với tốc độ cao và giá thành hợp lý. Smart phone còn có rất nhiều chức năng: lướt web, chơi game, xem phim trực tuyến, soạn email, sự dụng mạng xã hội… Ứng dụng bản đồ Google Map tích hợp có thể giúp bạn tiết kiệm được rất nhiều thời gian và chi phí giúp bạn di chuyển giữa 2 điểm bất kỳ trên bản đồ… Do đó việc thiết kế ứng dụng trên Smart Phone để giải quyết các công việc hằng ngày là rất cần thiết.

Khi con người quá bận rộn với công việc thì việc chăm con thủa ban đầu là một trong những nỗi lo. Việc chăm sóc con cái từ giai đoạn từ 4 tháng đến 12 tháng là giai đoạn quan trọng. Thời gian này các bậc cha mẹ thường không biết mình nên cho con tập ăn những món ăn gì? Cho con ăn sao đủ chất? Cho con ăn thế nào mới khoa học? Họ tìm đến các diễn đàn, các bài báo nhưng với tình hình hiện này khi các bài báo viết về việc cho con ăn dặm có khắp nơi trên các báo về gia đình, sức khỏe, chăm sóc mẹ và bé….nhưng lại không có sự tổng hợp khoa học. Các báo dẫn bài của nhau khiến cho việc tìm đúng thông tin khá vất vả. Có thể liệt kê nhược điểm của việc tìm kiếm thông tin về ăn dặm như sau:

* Quá nhiều thông tin trên các báo.
* Các báo dẫn bài của nhau khiến cho việc tìm kiếm bị trùng lặp.
* Bài viết trên báo giấy thì không xem được video hướng dẫn cụ thể.
* Các báo phần lớn chỉ hướng dẫn nhưng ít bài phân tích lợi và hại khi kết hợp thức ăn không khoa học.
* Bài viết ở báo giấy không có mục hỏi đáp trực tiếp để các bậc cha mẹ tham khảo lẫn nhau.
* Mất thời gian và cảm giác không tiện lợi đọc báo giấy. Hay quá nhiều thông tin trên báo mạng.

## II. Mục đích và ý nghĩa

#### II.1. Mục đích

Trong đề tài này nhóm đã nghiên cứu tìm hiểu nội dung cơ bản với những bước khi xây dựng ứng dụng trên nền tảng iOS.

Ứng dụng mà nhóm xây dựng nhằm mục đích hỗ trợ cho tất cả các bậc cha mẹ có một phần mềm thiết thực, tiện lợi. Đồng thời nâng cao khả năng làm việc nhóm cũng như năng cao tay nghề của mỗi cá nhân trong nhóm.

#### II.2. Ý nghĩa

Ứng dụng ra đời nhằm khắc phục những nhược điểm trên giúp các bậc cha mẹ có thể tìm đúng thông tin mình cần. Ứng dụng còn phân bố nội dung ra khá cụ thể thành nội dung như: công thức ăn dặm, ghi chú, video. Việc tích hợp xem chỉ số của con giúp bố mẹ xem con có đạt đúng tiêu chuẩn hay ngoài ra phần mềm còn có thể cho bố mẹ tham gia comment vào mỗi bài báo để việc lọc thông tin món ăn ngon được áp dụng ở các bố mẹ khác như thế nào. Ứng dụng còn cho biết lượt rating của mỗi món để bố mẹ lựa chọn vào số rating nếu họ cảm thấy những món ăn có lượt rating cao là đáng để nấu cho con ăn.

#### III.Nhiệm vụ thực hiện

#### III.1. Mục tiêu cụ thể

Điện thoại di động là thiết bị nhỏ gọn, được thiết kế phục vụ chủ yếu cho nhu cầu liên lạc và giải trí của người dùng khi “di chuyển”. Bộ xử lý cũng như khả năng lưu trữ của điện thoại di động kém xa so với máy để bàn. Trong môi trường xã hội ngày một nâng cao và phát triển, thêm vào đó đời sống con người ngày một nâng cao, vì thế những máy móc thiết bị ngày càng được các nhà sản xuất chú ý nhằm đáp ứng theo kịp nhịp độ sống con người hiện nay. Thời gian gần đây khi Apple thành công trong chuỗi sản phẩm iPhone và mang hệ điều hành iOS của mình trở nên phố biến tạo ra một sự thay đổi lớn trong lĩnh vực di động thông minh và nghành công nghiệp phần mềm. Vì vậy, nhiệm vụ của nhóm đặt ra trong đồ án tốt nghiệp lần này:

* Nghiên cứu công nghệ phát triển ứng dụng trên hệ điều hành di động iOS.
* Tổng hợp các vấn đề do Apple đặt ra khi phát triển ứng dụng di động triên

các thiết bị của Apple: iPhone, iPod Touch, iPad.

* Tìm hiểu các công thức nấu ăn dặm cho bé.
* Tìm hiểu các ghi chú trong việc nấu ăn dặm.
* Tìm hiểu các video hướng dẫn trong việc nấu ăn dặm.
* Xây dựng ứng dụng “Ăn Dặm Cho Bé” trên điện thoại iOS.

**III.2. Kết quả dự kiến**

Với mục đích đã được nêu ra, thì kết quả của nhóm cần có khi thực hiện đồ án tốt nghiệp này:

* Phân tích, thiết kế ứng dụng “Ăn Dặm Cho Bé” trên điện thoại di động iOS.
* Xây dựng chương trình “Ăn Dặm Cho Bé” trên điện thoại iOS.

**IV. Nội dung đề tài**

Cấu trúc báo cáo được thực hiện như sau:

* Chương I: Cơ sở lý thuyết.
* Hệ điều hành iOS và điện thoại iPhone.
* Điện toán đám mây và iCloud
* Ngôn ngữ lập trình object-C.
* Tổng quan về Web service.
* Tổng quan Php và Cakephp framework.
* Công cụ phát triển.
* Ứng dụng ăn dặm cho bé.
* Chương II: Xây dựng ứng dụng
* Khảo sát thực tế và mô tả bài toán.
* Phân tích thiết kế hệ thống.
* Thiết kế giao diện phần mềm.
* Thiết kế cơ sở dữ liệu.
* Chương III: Triển khai ứng dụng.
* Phương pháp xây dựng hệ thống.
* Cài đặt, thực thi chương trình.
* Kết quả demo.
* Thực thi chương trình.
* Hình ảnh kết quả demo về chức năng chương trình.
* Chương IV: Kết luận.
* Ưu điểm, hạn chế chương trình.
* Hướng phát triển.
* Phụ lục.
* Nguồn tài liệu tham khảo.

**CHƯƠNG I:** **CƠ SỞ LÝ THUYẾT**

1. **Tổng quan hệ điều hành iOS**

****

Hình 1: Các thiết bị chạy hệ điều hành di động iOS.

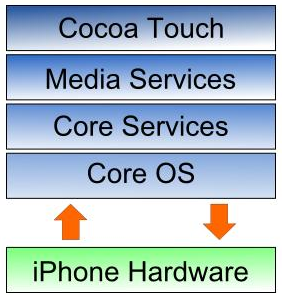
iOS là hệ điều hành chạy trên các thiết bị di động được sản xuất bởi Apple như iPhone, iPod touch,iPad... Hệ điều hành này quản lý hoạt động của cácthiết bị phần cứng và cung cấp tiện ích cũng như ứng dụng phục vụ cho hệ thống, ví dụ như: điện thoại, tin nhắn SMS, email, trình duyệt web Safari…

Hệ điều hành này được tiết lộ tại Hội nghị và Triển lãm Macworld diễn ra vào tháng 1 năm 2007 và được phát hành vào tháng 9 năm đó. Khi đó, hệ điều hành này chưa có tên riêng, nên chỉ đơn giản được gọi là “iPhone chạy OS X”. Vào ngày 17 tháng 10 năm 2007, Apple thông báo một bộ phát triển phần mềm đang được xây dựng và họ dự định sẽ đưa nó đến "tay của các nhà phát triển vào tháng 2". Ngày 6 tháng 3 năm 2008, Apple đã phát hành bản dùng thử đầu tiên, cùng với một cái tên mới cho hệ điều hành, đó là “iPhone OS”.

Tháng 6 năm 2010, Apple đổi cái tên iPhone OS thành iOS. Nhãn hiệu "IOS"

đã được Cisco dùng để đặt tên cho hệ điều hành của mình. Để trách các vụ kiện cáo, Apple đã xin giấy phép sử dụng nhãn hiệu iOS từ Cisco.

**I.1. Kiến trúc hệ điều hành iOS**



Hình 2: Kiến trúc iOS

Trong kiến trúc của iOS, ở cấp độ cao nhất, iOS hoạt động như một trung

gian giữa phần cứng cơ bản và các ứng dụng xuất hiện trên màn hình. Tại các

lớp thấp hơn của hệ thống là các dịch vụ cơ bản mà các ứng dụng cấp cao hơn sử

dụng để hoạt động.

**I.1.1. Tầng Cocoa Touch**

Cocoa là tập hợp các framework hướng đối tượng, cung cấp môi trường thực thi cho các ứng dụng chạy trên các hệ điều hành Mac OS và iOS. Cocoa là môi trường phát triển ứng dụng phổ biến và hiệu quả trong Mac OS, và là môi trường phát triển ứng dụng duy nhất trên iOS. Cocoa trong iOS được gọi là Cocoa Touch.

Nó bao gồm các bộ framework chính là:

UIKit: Framework này cung cấp các đối tượng và điều khiểnđể xây dựng giao

diện của các ứng dụng. Nó cũng giúp định nghĩa các hành vi của ứng dụng,xử lí các sự kiện, các thao tác của người dùng trên màn hình cảm ứng. Các tính năng khác có thể kể đến của framework này bao gồm:

* Quản lí vòng đời ứng dụng.
* Đa nhiệm.
* Local notification.
* Nhận diện các thao tác trên màn hình cảm úng
* Map Kit: Cung cấp các giao diện lập trình và thư viện để xây dựng các ứng dụng dựa trên bản đồ, định tuyến, xác định địa điểm hiện tại của người dùng.
* Message UI Framework: Cung cấp thư viện để soạn và gởi email.
* iAd Framework: Cho phép lập trình viên nhúng các banner quảng cáo vào ứng dụng.
* Push notification: Là dịch vụ chạy thường trực trên thiết bị, cho phép nhận thông báo được gởi bằng Push notification service ngay cả khi ứng dụng không được chạy.

**I.1.2. Tầng Media**

Cung cấp cho các ứng dụng iOS khả năng xử lí, khai thác hình ảnh, âm thanh, video, truyền hình. Các framework của tầng này bao gồm:

* Assets Library
* AV Foundation
* Core Audio
* Core Graphics
* Core MIDI
* Core Video
* Media Player
* Quartz Core
* OpenGL ES

**I.1.3. Tầng Core Service**

Đây là tầng chứa các dịch vụ cơ bản của hệ thống mà hầu hết các ứng dụng đều sử dụng. Các thành phần của lớp này bao gồm:

* High-level Features:
* In-app Purchase
* SQLite
* XML Support
* Block object
* Core Services Framework:
* Address Book
* Core Data
* Core Foundation
* Core Location
* Core Media
* System Configuration
* …

**I.1.4. Tầng Core OS**

Đây là lớp chứa các đặc trưng ở mức thấp (low-level features). Lập trình viên sẽ ít sử dụng trực tiếp lớp này (trừ khi thao tác với các thiết bị phần cứng khác hoặc các vấn đề bảo mật) mà sẽ sử dụng gián tiếp thông qua các lớp phía trên. Lớp này bao gồm các thành phần sau:

* Accelerate framework
* External Accessory framework
* Security framework
* System.

**II. Điện toán đám mây và iCloud**

**II.1. Tổng quan về điện toán đám mây**

Điện toán đám mây (tiếng Anh: cloud computing), còn gọi là điện toán máy chủ ảo, là mô hình điện toán sử dụng các công nghệ máy tính và phát triển dựa vào mạng Internet. Thuật ngữ "đám mây" ở đây là lối nói ẩn dụ chỉ mạng Internet (dựa vào cách được bố trí của nó trong sơ đồ mạng máy tính) và như một liên tưởng về độ phức tạp của các cơ sở hạ tầng chứa trong nó. Ở mô hình điện toán này, mọi khả năng liên quan đến công nghệ thông tin đều được cung cấp dưới dạng các "dịch vụ", cho phép người sử dụng truy cập các dịch vụ công nghệ từ một nhà cung cấp nào đó "trong đám mây" mà không cần phải có các kiến thức, kinh nghiệm về công nghệ đó, cũng như không cần quan tâm đến các cơ sở hạ tầng phục vụ công nghệ đó.

Theo tổ chức Xã hội máy tính IEEE "Nó là hình mẫu trong đó thông tin được lưu trữ thường trực tại các máy chủ trên Internet và chỉ được được lưu trữ tạm thời ở các máy khách, bao gồm máy tính cá nhân, trung tâm giải trí, máy tính trong doanh nghiệp, các phương tiện máy tính cầm tay, ...". Điện toán đám mây là khái niệm tổng thể bao gồm cả các khái niệm như phần mềm dịch vụ, Web 2.0 và các vấn đề khác xuất hiện gần đây, các xu hướng công nghệ nổi bật, trong đó đề tài chủ yếu của nó là vấn đề dựa vào Internet để đáp ứng những nhu cầu điện toán của người dùng. Ví dụ, dịch vụ Google AppEngine cung cấp những ứng dụng kinh doanh trực tuyến thông thường, có thể truy nhập từ một trình duyệt web, còn các phần mềm và dữ liệu đều được lưu trữ trên các máy chủ.



Hình 3: Điện toán đám mây

**II.1.1. Lịch sử**

Thuật ngữ điện toán đám mây xuất hiện bắt nguồn từ ứng dụng điện toán lưới (grid computing) trong thập niên 1980, tiếp theo là điện toán theo nhu cầu (utility computing) và phần mềm dịch vụ (SaaS).

Điện toán lưới đặt trọng tâm vào việc di chuyển một tải công việc (workload)

đến địa điểm của các tài nguyên điện toán cần thiết để sử dụng. Một lưới là một

nhóm máy chủ mà trên đó nhiệm vụ lớn được chia thành những tác vụ nhỏ để chạy song song, được xem là một máy chủ ảo.

Với điện toán đám mây, các tài nguyên điện toán như máy chủ có thể được định hình động hoặc cắt nhỏ từ cơ sở hạ tầng phần cứng nền và trở nên sẵn sàng thực hiện nhiệm vụ, hỗ trợ những môi trường không phải là điện toán lưới như Web ba lớp chạy các ứng dụng truyền thống hay ứng dụng Web 2.0.

**II.1.2. Đặc điểm**

Điện toán máy chủ ảo thường bị lẫn lộn với điện toán lưới, ("một dạng của điện toán phân tán trong đó tồn tại một 'siêu máy tính ảo', là sự bao gồm một cụm mạng máy tính, những máy tính liên kết mềm, hoạt động phối hợp để thực thi những tác vụ cực lớn"), điện toán theo nhu cầu (utility computing) ("khối những tài nguyên máy tính, như các bộ xử lý và bộ nhớ, trong vai trò một dịch vụ trắc lượng tương tự với các công trình hạ tầng kỹ thuật truyền thống chẳng hạn như điện lực hay mạng điện thoại") và điện toán tự trị (autonomic computing) ("những hệ thống máy tính có khả năng tự quản lý"). Thực ra nhiều hệ thống điện toán máy đám mây ngày nay được trang bị hệ thống lưới, có tính năng tự trị và được tiếp thị giống như những tiện ích, nhưng điện toán đám mây có thể được nhìn nhận như một bước phát triển tự nhiên tiếp theo từ mô hình lưới theo nhu cầu. Nhiều kiến trúc đám mây thành công có cơ sở hạ tầng không quy tập hay ít quy tập hay những hệ thống tiếp thị trong đó có mạng ngang hàng như BitTorrent và Skype và điện toán tình.

Kiến trúcĐại bộ phận hạ tầng cơ sở của điện toán đám mây hiện nay là sự kết hợp của những dịch vụ đáng tin cậy được phân phối thông qua các trung tâm dữ liệu (data center) được xây dựng trên những máy chủ với những cấp độ khác nhau của các công nghệ ảo hóa. Những dịch vụ này có thể được truy cập từ bất kỳ đâu trên thế giới, trong đó Đám mây là một điểm truy cập duy nhất cho tất cả các máy tính có nhu cầu của khách hàng. Các dịch vụ thương mại cần đáp ứng yêu cầu chất lượng dịch vụ từ phía khách hàng và thông thường đều đưa ra các mức thỏa thuận dịch vụ (Service level agreement). Các tiêu chuẩn mở (Open standard) và phần mềm mã nguồn mở (open source software) cũng góp phần phát triển điện toán máy chủ ảo.

Trước đây để có thể triển khai một ứng dụng (ví dụ một trang Web), bạn phải đi mua/thuê một hay nhiều máy chủ (server), sau đó đặt máy chủ tại các trung tâm dữ liệu (data center) thì nay điện toán đám mây cho phép bạn giản lược quá trình mua/thuê đi. Bạn chỉ cần nêu ra yêu cầu của mình, hệ thống sẽ tự động gom nhặt các tài nguyên rỗi (free) để đáp ứng yêu cầu của bạn. Chính vì vậy, có thể kể đến một vài lợi ích cơ bản của điện toán đám mây như sau :

Sử dụng các tài nguyên tính toán động (Dynamic computing resources) : Các tài nguyên được cấp phát cho doanh nghiệp đúng như những gì doanh nghiệp muốn một cách tức thời. Thay vì việc doanh nghiệp phải tính toán xem có nên mở rộng hay không, phải đầu tư bao nhiêu máy chủ thì nay doanh nghiệp chỉ cần yêu cầu “Hey, đám mây, chúng tôi cần thêm tài nguyên tương đương với 1 CPU 3.0 GHz, 128GB RAM…” và đám mây sẽ tự tìm kiếm tài nguyên rỗi để cung cấp cho bạn.

Giảm chi phí: Doanh nghiệp sẽ có khả năng cắt giảm chi phí để mua bán, cài đặt và bảo trì tài nguyên. Rõ ràng thay vì việc phải cử một chuyên gia đi mua máy chủ, cài đặt máy chủ, bảo trì máy chủ thì nay bạn chẳng cần phải làm gì ngoài việc xác định chính xác tài nguyên mình cần và yêu cầu. Quá tiện!.

Giảm độ phức tạp trong cơ cấu của doanh nghiệp: Doanh nghiệp sản xuất hàng

hóa mà lại phải có cả một chuyên gia IT để vận hành, bảo trì máy chủ thì quá tốn kém. Nếu outsource được quá trình này thì doanh nghiệp sẽ chỉ tập trung vào việc sản xuất hàng hóa chuyên môn của mình và giảm bớt được độ phức tạp trong cơ cấu.Tăng khả năng sử dụng tài nguyên tính toán: Một trong những câu hỏi đau đầu của việc đầu tư tài nguyên (ví dụ máy chủ) là bao lâu thì nó sẽ hết khấu hao, tôi đầu tư như thế có lãi hay không, có bị outdate về công nghệ hay không. Khi sử dụng tài nguyên trên đám mây thì bạn không còn phải quan tâm tới điều này nữa.

**II.2. iCloud**

iCloud là dịch vụ “tất cả trong một” được Apple trình làng vào ngày 6/6 vừa qua tại hội nghị WWDC. Về cơ bản, iCloud là dịch vụ hoạt động trên công nghệ “điện toán đám mây”, cung cấp cho người dùng cách thức mới để lưu trữ và truy cập dữ liệu từ mọi thiết bị của Apple. Mọi quá trình này đều được tiến hành một cách tự động.

Dưới đây là một vài thông tin cần thiết cho những ai quan tâm đến dịch vụ đám mây iCloud của Apple:



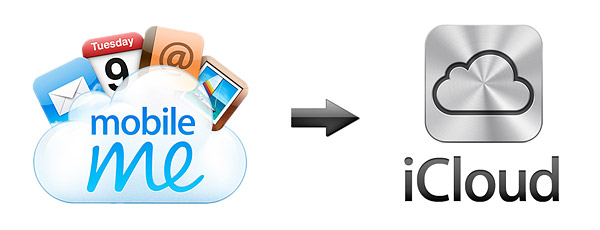
Hình 4: Điện toán đám mây và thiết bị

**Ngày phát hành chính thức**

Tương tự như iOS 5, iCloud sẽ được ra mắt vào mua thu năm nay, nhưng với các nhà phát triển ứng dụng thì đã có thể thử nghiệm iCloud ngay trên iOS 4.3.

**iCloud sẽ thay thế dịch vụ MobileMe**

Trong buổi giới thiệu tại hội nghị WWDC diễn ra vào thứ 2, Steve Jobs tuyên bố MobileMe (có giá 99 USD) sẽ không còn tiếp tục được Apple cung cấp. Ứng dụng MobileMe sẽ được xây dựng lại để trở thành iCloud.



Hình 5: Chuyển đổi từ mobile me qua iClound

Các dịch vụ trên Mobile Me như danh sách liên lạc, lịch, email… sẽ được lưu

trữ và chuyển đổi sang iCloud. Người dùng có thể chia sẻ lịch của mình với bạn bè và gia đình, trong khi đó hộp thư email sẽ được đồng bộ hóa giữa các thiết bị sử dụng iOS và máy tính sử dụng Mac OS X.

**Công cụ soạn thảo văn bản iCloud Documents**

Có rất nhiều tính năng khác nhau được cung câp trên iCloud, nổi bật trong đó là công cụ soạn thảo trực tuyến Documents, đây được xem là một đối thủ của công cụ Google Docs.

Với iCloud Documents, người dùng có thể upload các file văn bản lên iCloud rồi chia sẻ, chỉnh sửa và sử dụng ở bát kỳ đâu và tự động đồng bộ hóa trên các thiết bị nếu cần thiết.

**Tính năng chia sẻ ảnh Photostream của iCloud**

iCloud sẽ cho phép người dùng đồng bộ hóa các hình ảnh trên “đám mây”. Theo đó, các bức ảnh từ mọi thiết bị của Apple sẽ được upload và đồng bộ hóa trên iCloud. Đặc biệt các hình ảnh được đồng bộ hóa này sẽ không bị tính vào 5GB dung lượng miễn phí mà Apple cung cấp.

**Danh sách liên lạc trên iCloud**

Danh sách liên lạc trên các thiết bị cũng được upload và lưu trữ trên iCloud và

đồng bộ hóa trên mọi thiết bị của Apple. Mọi sự thay đổi trong danh sách liên lạc sẽ được tự động thay đổi theo ở các thiết bị.

**Sao lưu trên iCloud**

Tính năng sao lưu trên iCloud sẽ tự động sao lưu một cách bảo mật dữ liệu từ

các thiết bị sử dụng iOS hàng ngày thông qua Wifi khi các thiết bị này ở trạng thái nghỉ (sạc pin). Các nội dung được sao lưu, bao gồm các bản nhạc, ứng dụng và sách đã đặt mua, hình ảnh và video ghi lại từ thiết bị, thiết lập của các ứng dụng và của thiết bị.

Trong trường hợp người dùng thay đổi thiết bị (đổi mới iPhone, iPad…), chỉ

việc đăng nhập vào tài khoản của Apple trong quá trình thiết lập, iCloud sẽ phục hồi toàn bộ dữ liệu trên thiết bị cũ sang thiết bị mới.

**Lưu trữ dữ liệu**

Tương tự như các dịch vụ khác, tính năng lưu trữ trên iCloud cũng sẽ đồng bộ

hóa dữ liệu trên toàn bộ các thiết bị. Khi người dùng thay đổi nội dung của một file trên 1 thiết bị nào đó, iCloud sẽ tự động thực hiện thay đổi nội dung ở các thiết bị nào lại.

**Ứng dụng và sách trên iCloud**

Tương tự như việc mua nhạc của iTunes trên iCloud, các ứng dụng mua từ App Store và sách mua từ iBookstore bây giờ có thể được download và lưu trữ trên toàn bộ các thiết bị của Apple mà không cần phải trả thêm bất kỳ một khoản phí nào khác.

**III.Ngôn ngữ lập trình object-C**

**III.1. Giới thiệu**

Ngôn ngữ lập trình Objective-C là một ngôn ngữ lập trình hướng đối tượng được xây dựng chủ yếu dựa trên nền tảng ANSI C. Ngoài ra nó còn được mở rộng từ Smalltalk, một trong những ngôn ngữ lập trình hướng đối tượng đầu tiên. Objective-C được thiết kế với mục đích đưa vào C các tính năng hướng đối tượng một các đơn giản và dễ hiểu nhất.

Objective-C được thiết kế bởi Brad Cox khi ông làm việc cho công ty Stepstone vào đầu những năm 1980. Objective-C được thiết kế để phục vụ mục đích lập trình hướng đối tượng. Nó hoạt động giống như là một tập hợp các thành phần mở rộng rất mạnh mẽ của ngôn ngữ C. Objective-C kết hợp các đặc điểm ưu tú nhất của C và ngôn ngữ SmallTalk. Objective-C khá đơn giản để học và có đầy đủ các khả năng của một ngôn ngữ lập trình hướng đối tượng.Hiện tại, Objective-C trở nên phổ biến vì nó được Apple chọn làm ngôn ngữ lập trình cho hệ thống Mac và iPhone.

**III.2. Thành Phần**

Hầu hết các ngôn ngữ lập trình hướng đối tượng đều cung cấp:

* Một thư viện các Objects.
* Các công cụ phát triển cần thiết
* Hỗ trợ hướng đối tượng và các thư viện liên quan.

Là ngôn ngữ hướng đối tượng nên Objective-C giải quyết các vấn đề lập trình

dựa trên khái niệm các Object. Nó bao gồm 3 phần:

**III.2.1. Interface**

Interface của một lớp(class) thông thường được định nghĩa trong tệp tiêu đề với

đuôi .h. Nó chính là phần khai báo của một lớp.

**III.2.2. Implementation**

Mã nguồn của chương trình được viết trong phần cài đặt của một lớp và được

định nghĩa trong một tệp có đuôi .m. Đây là nó phần định nghĩa của lớp.

**III.2.3. Instantiation**

Sau khi khai báo và định nghĩa một lớp, chúng ta có thể thực thể hóa lớp này

bằng việc cấp phát bộ nhớ cho new object của lớp đó.

Tóm lại Objective-C là:

* Thành phần mở rộng của ngôn ngữ lập trình C.
* Là ngôn ngữ lập trình hướng đối tượng đơn giản nhưng mạnh mẽ.
* Là ngôn ngữ được Apple sử dụng để phát triển các ứng dụng cho hệ thống
* máy Mac và iPhone.

**III.3. Thế mạnh**

Ngôn ngữ lập trình Objective-C được chọn cho Cocoa framework vì một số lý

do dưới đây:

* Nó là ngôn ngữ lập trình hướng đối tượng và các chức năng cung cấp bởi Cocoa framework chỉ có thể được mang lại bởi các kỹ thuật hướng đối tượng.
* Nó là thành phần mở rộng của chuẩn ANSI C vì vậy các chương trình viết bằng C của framework này sẽ không bị mất đi tính năng nào và người dùng được hưởng các lợi thế của ngôn ngữ C.Với ngôn ngữ này, người dùng có thể lựa chọn cả lập trình hướng đối tượng và lập trình thủ tục khi cần thiết.
* Nó đơn giản và dễ học bởi cú pháp của nó khá ngắn gọn nên nó giúp cho lập trình viên đạt được hiệu quả mong muốn mà không gặp nhiều khó khăn.
* Nó là một ngôn ngữ mạnh bởi vì các quyết định có thể được đưa ra lúc biên dịch sẽ được trì hoãn cho tới khi chương trình chạy.
* Nó hỗ trợ ràng buộc động và mở tạo ra một cấu trúc đơn giản đối với giao diện tương tác người dùng.
* Nó cho phép phát triển các công cụ phát triển phức tạp. Một giao diện cho hệ thống run time tạo điều kiện cho việc tiếp cận các thông tin lúc run time giúp cho việc monitor ứng dụng viết bằng Objective-C.

**IV.Tổng quan về Web service và SOAP**

**IV.1. Web Service**

Dịch vụ Web (Web Service) được coi là một công nghệ mang đến cuộc cách

mạng trong cách thức hoạt động của các dịch vụ B2B (Business to Business) và

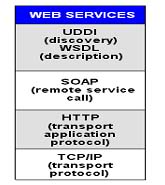
B2C (Business to Customer). Giá trị cơ bản của dịch vụ Web dựa trên việc cung cấp các phương thức theo chuẩn trong việc truy nhập đối với hệ thống đóng gói và hệ thống kế thừa. Các phần mềm được viết bởi những ngôn ngữ lập trình khác nhau và chạy trên những nền tảng khác nhau có thể sử dụng dịch vụ Web để chuyển đổi dữ liệu thông qua mạng Internet theo cách giao tiếp tương tự bên trong một máy tính.

Tuy nhiên, công nghệ xây dựng dịch vụ Web không nhất thiết phải là các công nghệ mới, nó có thể kết hợp với các công nghệ đã có như XML, SOAP, WSDL, UDDI…

Với sự phát triển và lớn mạnh của Internet, dịch vụ Web thật sự là một công nghệ đáng được quan tâm để giảm chi phí và độ phức tạp trong tích hợp và phát triển hệ thống. Chúng ta sẽ xem xét các dịch vụ Web từ mức khái niệm đến cách thức xây dựng.

**IV.1.1. Giới thiệu công nghệ**

Dịch vụ Web gồm có 3 chuẩn chính: SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language) và UDDI (Universal Description, Discovery, and Integration). Hình 2 mô tả chồng giao thức của dịch vụ Web, trong đó UDDI được sử dụng để đăng ký và khám phá dịch vụ Web đã được miêu tả cụ thể trong WSDL. Giao tác UDDI sử dụng SOAP để nói chuyện với UDDI server, sau đó các ứng dụng SOAP yêu cầu một dịch vụ Web. Các thông điệp SOAP được gửi đi chính xác bởi HTTP và TCP/IP.



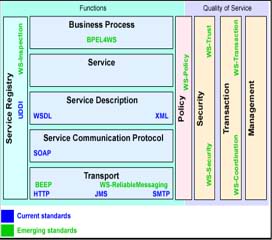
Hình 6: Chồng giao thức dịch vụ Web

Chồng giao thức dịch vụ Web là tập hợp các giao thức mạng máy tính được sử dụng để định nghĩa, xác định vị trí, thi hành và tạo nên dịch vụ Web tương tác với những ứng dụng hay dịch vụ khác. Chồng giao thức này có 4 thành phần chính:

- Dịch vụ vận chuyển (Service Transport): có nhiệm vụ truyền thông điệp giữa các ứng dụng mạng, bao gồm những giao thức như HTTP, SMTP, FTP, JSM và gần đây nhất là giao thức thay đổi khổi mở rộng (Blocks Extensible Exchange Protocol- BEEP).

- Thông điệp XML: có nhiệm vụ giải mã các thông điệp theo định dạng XML để có thể hiểu được ở mức ứng dụng tương tác với người dùng. Hiện tại, những giao thức thực hiện nhiệm vụ này là XML-RPC, SOAP và REST.  
- Mô tả dịch vụ: được sử dụng để miêu tả các giao diện chung cho một dịch vụ Web cụ thể. WSDL thường được sử dụng cho mục đích này, nó là một ngôn ngữ mô tả giao tiếp và thực thi dựa trên XML. Dịch vụ Web sẽ sử dụng ngôn ngữ này để truyền tham số và các loại dữ liệu cho các thao tác và chức năng mà dịch vụ Web cung cấp.

- Khám phá dịch vụ: tập trung dịch vụ vào trong một nơi được đăng ký, từ đó giúp một dịch vụ Web có thể dễ dàng khám phá ra những dịch vụ nào đã có trên mạng, tốt hơn trong việc tìm kiếm những dịch vụ khác để tương tác. Một dịch vụ Web cũng phải tiến hành đăng ký để các dịch vụ khác có thể truy cập và giao tiếp. Hiện tại, UDDI API thường được sử dụng để thực hiện công việc này.



Hình 7: Kiến trúc của Web Service

Trong đó, tầng giao thức tương tác dịch vụ (Service Communication Protocol) với công nghệ chuẩn là SOAP. SOAP là giao thức nằm giữa tầng vận chuyển và tầng mô tả thông tin về dịch vụ, cho phép người dùng triệu gọi một dịch vụ từ xa thông qua một thông điệp XML. Ngoài ra, để các dịch vụ có tính an toàn, toàn vẹn và bảo mật thông tin, trong kiến trúc dịch vụ Web, chúng ta có thêm các tầng Policy, Security, Transaction, Management.

#### IV.1.2. Các thành phần của Dịch vụ Web

**IV.1.2.1. XML – eXtensible Markup Language**

Là một chuẩn mở do W3C đưa ra cho cách thức mô tả dữ liệu, nó được sử dụng để định nghĩa các thành phần dữ liệu trên trang web và cho những tài liệu B2B. Về hình thức, XML hoàn toàn có cấu trúc thẻ giống như ngôn ngữ HTML nhưng HTML định nghĩa thành phần được hiển thị như thế nào thì XML lại định nghĩa những thành phần đó chứa cái gì. Với XML, các thẻ có thể được lập trình viên tự tạo ra trên mỗi trang web và được chọn là định dạng thông điệp chuẩn bởi tính phổ biến và hiệu quả mã nguồn mở.

Do dịch vụ Web là sự kết hợp của nhiều thành phần khác nhau nên nó sử dụng các tính năng và đặc trưng của các thành phần đó để giao tiếp. XML là công cụ chính để giải quyết vấn đề này và là kiến trúc nền tảng cho việc xây dựng một dịch vụ Web, tất cả dữ liệu sẽ được chuyển sang định dạng thẻ XML. Khi đó, các thông tin mã hóa sẽ hoàn toàn phù hợp với các thông tin theo chuẩn của SOAP hoặc XML-RPC và có thể tương tác với nhau trong một thể thống nhất.

**IV.1.2.2. WSDL – Web Service Description Language**

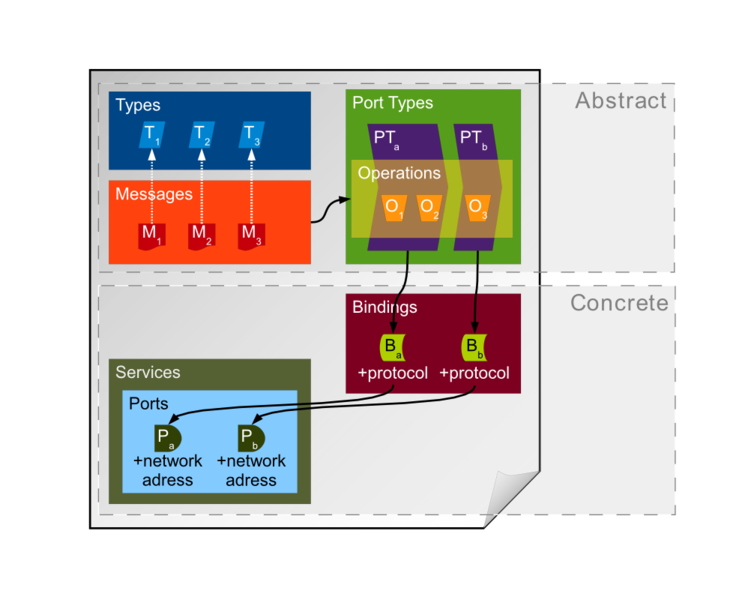
WSDL định nghĩa cách mô tả dịch vụ Web theo cú pháp tổng quát của XML, bao gồm các thông tin:

- Tên dịch vụ

- Giao thức và kiểu mã hóa sẽ được sử dụng khi gọi các hàm của dịch vụ Web

- Loại thông tin: thao tác, tham số, những kiểu dữ liệu (có thể là giao diện của dịch vụ Web cộng với tên cho giao diện này).

Một WSDL hợp lệ gồm hai phần: phần giao diện (mô tả giao diện và phương thức kết nối) và phần thi hành mô tả thông tin truy xuất CSDL. Cả hai phần này sẽ được lưu trong 2 tập tin XML tương ứng là tập tin giao diện dịch vụ và tập tin thi hành dịch vụ. Giao diện của một dịch vụ Web được miêu tả trong phần này đưa ra cách thức làm thế nào để giao tiếp qua dịch vụ Web. Tên, giao thức liên kết và định dạng thông điệp yêu cầu để tương tác với dịch vụ Web được đưa vào thư mục của WSDL.

WSDL thường được sử dụng kết hợp với XML schema và SOAP để cung cấp dịch vụ Web qua Internet. Một client khi kết nối tới dịch vụ Web có thể đọc WSDL để xác định những chức năng sẵn có trên server. Sau đó, client có thể sử dụng SOAP để lấy ra chức năng chính xác có trong WSDL.  


Hình 8: Tổng quan về WSDL

IV.1.2.3. Universal Description, Discovery, and Integration (UDDI)

Để có thể sử dụng các dịch vụ, trước tiên client phải tìm dịch vụ, ghi nhận thông tin về cách sử dụng và biết được đối tượng nào cung cấp dịch vụ. UDDI định nghĩa một số thành phần cho biết các thông tin này, cho phép các client truy tìm và nhận những thông tin được yêu cầu khi sử dụng dịch vụ Web.

- Cấu trúc UDDI :

* Trang trắng – White pages: chứa thông tin liên hệ và các định dạng chính yếu của dịch vụ Web, chẳng hạn tên giao dịch, địa chỉ, thông tin nhận dạng… Những thông tin này cho phép các đối tượng khác xác định được dịch vụ.
* Trang vàng – Yellow pages: chứa thông tin mô tả dịch vụ Web theo những loại khác nhau. Những thông tin này cho phép các đối tượng thấy được dịch vụ Web theo từng loại với nó.
* Trang xanh – Green pages: chứa thông tin kỹ thuật mô tả các hành vi và các chức năng của dịch vụ Web.
* Loại dịch vụ – tModel: chứa các thông tin về loại dịch vụ được sử dụng.

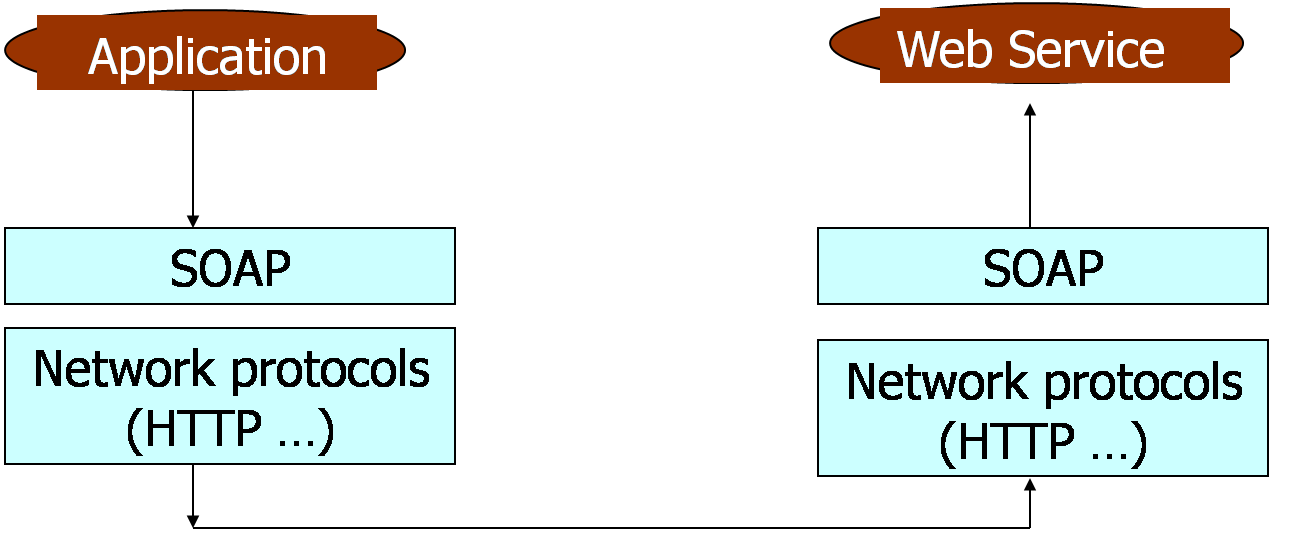
Những thông tin về dịch vụ Web được sử dụng và công bố lên mạng sử dụng giao thức này. Nó sẽ kích hoạt các ứng dụng để tìm kiếm thông tin của dịch vụ Web khác nhằm xác định xem dịch vụ nào sẽ cần đến nó.

**IV.1.2.4SOAP – Simple Object Access Protocol**

Các dịch vụ Web có thể truy xuất bằng một giao thức là Simple Object Access Protocol – SOAP. Nói cách khác chúng ta có thể truy xuất đến UDDI registry bằng các lệnh gọi hoàn toàn theo định dạng của SOAP.

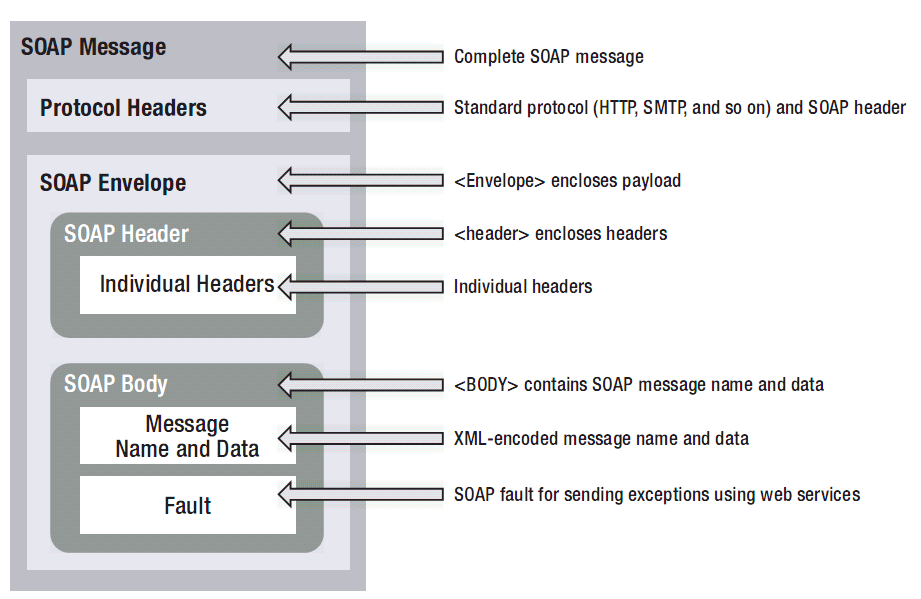
SOAP là một giao thức giao tiếp có cấu trúc như XML. Nó được xem là cấu trúc xương sống của các ứng dụng phân tán được xây dựng từ nhiều ngôn ngữ và các hệ điều hành khác nhau. SOAP là giao thức thay đổi các thông điệp dựa trên XML qua mạng máy tính, thông thường sử dụng giao thức HTTP.

Một client sẽ gửi thông điệp yêu cầu tới server và ngay lập tức server sẽ gửi những thông điệp trả lời tới client. Cả SMTP và HTTP đều là những giao thức ở lớp ứng dụng của SOAP nhưng HTTP được sử dụng và chấp nhận rộng rãi hơn bởi ngày nay nó có thể làm việc rất tốt với cơ sở hạ tầng Internet.



Hình 9: Thông điệp sử dụng SOAP

Cấu trúc một thông điệp theo dạng SOAP



Hình 10: Cấu trúc của thông điệp theo dạng SOAP

Thông điệp theo định dạng SOAP là một văn bản XML bình thường bao gồm các phần tử sau:

* SOAP Envelop: phần tử bao trùm nội dung thông điệp, khai báo văn bản XML như là một thông điệp SOAP.
* SOAP Header: chứa các thông tin tiêu đề cho trang, phần tử này không bắt buộc khai báo trong văn bản. Header còn có thể mang những dữ liệu chứng thực, những chứ ký số, thông tin mã hóa hay cài đặt cho các giao dịch khác.
* SOAP body: chứa các thông tin yêu cầu và thông tin được phản hồi.

Phần tử đưa ra các thông tin về lỗi -fault, cung cấp thông tin lỗi xảy ra trong qúa trình xử lý thông điệp.

* Protocol Header: Cho biết thông tin về các chuẩn giao thức được sử dụng.

Một SOAP đơn giản trong body sẽ lưu các thông tin về tên thông điệp, tham chiếu tới một thể hiện của dịch vụ, một hoặc nhiều tham số. Có 3 kiểu thông báo sẽ được đưa ra khi truyền thông tin: request message(tham số gọi thực thi một thông điệp), respond message (các tham số trả về, được sử dụng khi yêu cầu được đáp ứng) và cuối cùng là fault message (thông báo tình trạng lỗi).

Kiểu truyền thông: Có 2 kiểu truyền thông

- Remote procedure call (RPC): cho phép gọi hàm hoặc thủ tục qua mạng. Kiểu này được khai thác bởi nhiều dịch vụ Web.

- Document: được biết đến như kiểu hướng thông điệp, nó cung cấp giao tiếp ở mức trừu tượng thấp, khó hiểu và yêu cầu lập trình viên mất công sức hơn.  
Hai kiểu truyền thông này cung cấp các định dạng thông điệp, tham số, lời gọi đến các API khác nhau nên việc sử dụng chúng tùy thuộc vào thời gian và sự phù hợp với dịch vụ Web cần xây dựng.

Cấu trúc dữ liệu: Cung cấp những định dạng và khái niệm cơ bản giống như trong các ngôn ngữ lập trình khác như kiểu dữ liệu (int, string, date…) hay những kiều phức tạp hơn như struct, array, vector… Định nghĩa cấu trúc dữ liệu SOAP được đặt trong namespace SOAP-ENC.

## V. Tổng quan PHP và Cakephp framework.

## V.1. Tổng quan ngôn ngữ lập trình PHP

PHP (PHP: Hypertext Preprocessor) là ngôn ngữ kịch bản tầng server giống như ASP. Kịch bản PHP được thực hiện ở trên server. Nó hỗ trợ nhiều cơ sở dữ (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC...). Nó là mã nguồn mở và chạy được trên nhiều nền tảng khác nhau (Windows, Linux, Unix…). Tập tin PHP có thể chứa văn bản thuần túy, các thẻ HTML và các ngôn ngữ kịch bản như javascript. Nó luôn luôn trả về cho trình duyệt một HTML thuần túy. Do được tối ưu hóa cho các ứng dụng web, tốc độ nhanh, nhỏ gọn, cú pháp giống C và Java, dễ học và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một ngôn ngữ lập trình web phổ biến nhất thế giới. Rất nhiều trang web quy mô lớn sử dụng ngôn ngữ PHP như Facebook, Yahoo, Youtube, Wikipedia.

**Đặc điểm của PHP**

* PHP là ngôn ngữ kịch bản phổ biến nhất thế giới. Nó có nhiều ưu điểm: nhanh, ổn định, an toàn và là mã nguồn mở (miễn phí).
* PHP chạy được trên mọi nền tảng bao gồm Unix, Mac và Windows.
* PHP sử dụng ít tài nguyên của hệ thống, nó sử dụng như một mô-dun Appache nên thực thi nhanh. PHP cũng ổn định bởi vì cộng đồng PHP rộng lớn sẵn sàng làm việc với nhau để sửa chữa bất kỳ lỗi nào.
* Khả năng liên kết cao. Nó sử dụng các mô-đun hệ thống mở rộng để liên kết với hàng loạt các thư viện như đồ họa, XML, mã hóa… Lập trình viên cũng có thể viết mô-đun mở rộng, biên dịch thành tập tin thực thi và thêm vào PHP.
* PHP có rất nhiều giao diện máy chủ (Apache, IIS, Roxen, THTTPD and AOLserver…), giao diện cơ sở dữ liệu (MySQL, MS SQL, Informix, Oracle…).
* MySQL là một máy chủ cơ sở dữ liệu được thiết kế cho cả những ứng dụng nhỏ lẫn đồ sộ. Nó hỗ trợ các chuẩn của SQL (Structured Query Language) và chạy được trên nhiều nền tảng.

**V.2. Sơ lược về CakePhp**

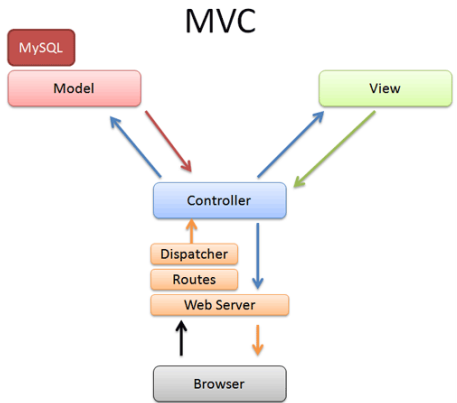
CakePHP là một nền tảng phát triển ứng dụng nhanh, mã nguồn mở miễn phí

và sử dụng ngôn ngữ lập trình PHP. Mục tiêu chủ yếu của nó là tạo ra một nền tảng có cấu trúc, cho phép bạn làm việc trên cấu trúc đó một cách nhanh chóng mà không mất đi sự uyển chuyển.

**Đặc điểm CakePHP**

* Cộng đồng năng động, thân thiện
* Việc cấp phép uyển chuyển
* Tương thích với PHP 4 và PHP 5
* Tích hợp sẵn CRUD (Khi làm việc với cơ sở dữ liệu)
* Hỗ trợ làm việc với cơ sở dữ liệu (scaffolding)
* Tự động sinh mã code
* Kiến trúc MVC
* Cho phép tạo ra các URL rõ ràng, dễ hiểu.
* Cung cấp khả năng bắt lỗi
* Cho phép tạo ra các bản mẫu (templating) nhanh chóng và uyển chuyển
* Các tiện ích giao diện cho AJAX, JavaScript, HTML Forms,...
* Có các công cụ xử lý Email, Cookie, Security, Session, yêu cầu (Request Handling)
* Danh sách điều khiển truy cập (ACL) uyển chuyển
* Cung cấp sẵn tiện ích xử lý dữ liệu (Data Sanitization)
* Cung cấp khả năng lưu tạm (Caching) uyển chuyển
* Cung cấp khả năng nội địa hóa (localization)

CakePHP sử dụng kiến trúc MVC (Model – View - Controller).



Hình 11: Mô hình MVC

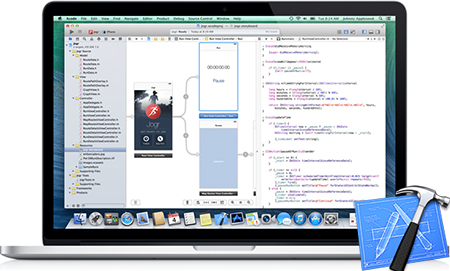
Models (Tầng dữ liệu): là một đối tượng hoặc một tập hợp các đối tượng biểu diễn cho phần dữ liệu của chương trình, ví dụ như các dữ liệu được lưu trong database, dữ liệu từ một hệ thống các ứng dụng khác như legacy system, file system, mail system…

* Views (Tầng giao diện): là phần giao diện với người dùng, bao gồm việc hiện dữ liệu ra màn hình, cung cấp các menu, nút bấm, hộp đối thoại, chọn lựa…, để người dùng có thể thêm, xóa. sửa, tìm kiếm và làm các thao tác khác đối với dữ liệu trong hệ thống.. Thông thường, các thông tin cần hiển thị được lấy từ thành phần Models.
* Controllers (Tầng điều khiển): chịu trách nhiệm xử lý các tác động về mặt giao diện, các thao tác đối với models, và cuối cùng là chọn một view thích hợp để hiển thị ra màn hình. Trong kiến trúc MVC, view chỉ có tác dụng hiển thị giao diện mà thôi, còn điều khiển dòng nhập xuất của người dùng vẫn do Controllers đảm trách.Xây dựng ứng dụng luyện thi trắc nghiệm tiếng Anh trên điện thoại di động iPhone

Ngoài ra, CakePHP cung cấp nhiều lớp với phương thức đa dạng giúp xử lý những công việc một cách dễ dàng và nhanh chóng.

## VI. Công cụ phát triển.

### VI.1. Giới thiệu xCode và iOS SDK

****

Hình 12: xCode và SDK iOS

Ra đời vào ngày 6 tháng 3 năm 2008, xCode là bộ công cụ hoàn chỉnh được phát triển bởi Apple Inc nhằm mục đích hổ trợ việc xây dựng ứng dụng trên OS X và iOS. Các công cụ cần cho quá trình phát triển ứng dụng đã được thiết kế và tích hợp để các công việc được thực hiện nhanh, dễ dàng, dễ sử dụng và hữu ích hơn bao giờ hết. IDE Xcode hiểu từng chi tiết của dự án, xác định những lỗi sai trong cả cú pháp và cả logic, và thậm chí sẽ sửa chữa mã sai.

Cùng với Xcode, SDK có chứa Simulator iPhone, một chương trình được sử dụng để mô phỏng giao diện của iPhone trên máy tính để bàn của nhà phát triển. Ban đầu được gọi Simulator Aspen, nó đã được đổi tên cùng với việc phát hành phiên bản Beta 2 của SDK. Lưu ý rằng xCode chỉ được hổ trợ để cài trên thiết bị Mac OS và không hổ trợ cho bất kỳ hệ điều hành nào khác như thông cáo từ phía Apple dành cho sản phầm mới nhất của họ:

*“SDK mới nhất, iOS 6,0 SDK trong Xcode 4.5, đòi hỏi phải có một máy Mac Intel chạy Mac OS X 10.7.4 "Lion" hoặc sau đó. Hệ điều hành khác, bao gồm cả Microsoft Windows và các phiên bản cũ của hệ điều hành Mac OS X, không được hỗ trợ”.*

1. **Các cửa sổ đơn lẽ**



Hình 13: Cửa sổ đơn lẽ

Là một người dùng mới tiếp xúc với phần mềm, điều đầu tiên bạn sẽ nhận thấy về Xcode là chứa nhiều cửa sổ được sử dụng để thực hiện các nhiệm vụ phát triển khác nhau hổ trợ bạn làm việc theo những chức năng nhất định.

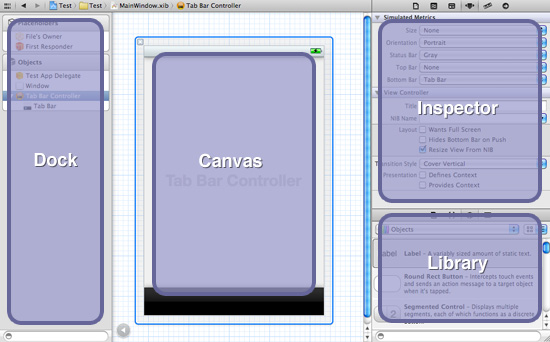
1. **Jump Bar**

****

Hình 14: Jump Bar Xcode

Ở phía trên của tất cả các cửa sổ soạn thảo là một thanh hiển thị các vị trí tương đối của các tập tin hiện tại của bạn. Nhấp vào bất kỳ vị trí trong đường dẫn để ngay lập tức nhảy đến bất kỳ tập tin ở cấp đó. Đây là Jump Bar, và nó hiệu quả cho việc bạn có thể muốn dành toàn bộ máy tính của bạn với việc lưu trữ mã nguồn, nhanh chóng nhảy từ tập tin vào tập tin.

1. **Bộ xây dựng giao diện tích hợp**

****

Hình 15: GUI Builder xCode

Xcode , giao diện Builder đã được hoàn toàn tích hợp trong IDE Xcode không có ứng dụng riêng biệt.

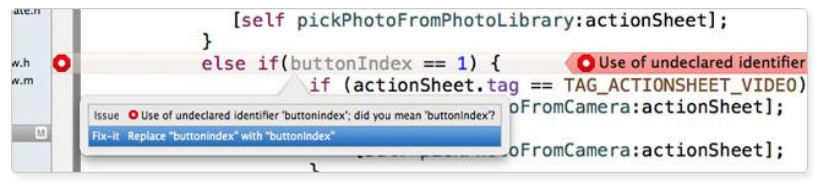
Lựa chọn một tập tin giao diện (.nib / .xib) trong dự án của bạn sẽ mở trình soạn thảo xây dựng giao diện trong Xcode. Mở khu vực tiện ích bên phải sẽ hiển thị đầy đủ của thanh tra giao diện, cũng như các thư viện của các điều khiển và các đối tượng giao diện người dùng. Kéo một điều khiển từ thư viện, và thả lên cửa sổ thiết kế, để bố trí giao diện phần mền trên hệ điều hành của bạn X hay iOS.

Bạn có thể kéo các kết nối trực tiếp từ thiết kế giao diện người dùng đến mã nguồn thông qua giao diện làm việc 2 cửa sổ. Chỉ đơn giản bằng cách kéo một kết nối đến mã hiện tại sẽ tạo ra hàng loạt hành động mới hoặc hành động mặc định cho bạn, chỉ cần kéo một không gian trống trong tập tin nguồn của bạn và Xcode sẽ tạo ra các mã mới.



Hình 16: Kéo thả kết nối giao diện và mã

1. **Tự động tìm và sửa lỗi**

****

Hình 17: Tự phát sinh sửa lỗi

Vượt ra ngoài việc chỉ báo cáo các lỗi, IDE là đủ thông minh để sửa chữa các vấn đề cho bạn. Trong nhiều trường hợp Xcode sẽ không chỉ báo cáo một lỗi, nó sẽ trình bày một giải pháp là tốt. Nhấp vào lỗi để xem các gợi ý sửa chữa của nó, chẳng hạn như điều chỉnh một chuyển nhượng để so sánh, sửa chữa một biểu tượng sai chính tả, hoặc phụ thêm một dấu chấm phẩy mất tích. Một phím tắt ngay lập tức sẽ có các lỗi sửa chữa, và cho phép bạn tiếp tục mã hóa.

Tìm kiếm và sửa chữa mã cùng lúc gõ là tính năng hổ trợ đắc lực cho người lập trình. Xcode sẽ phân tích qua hàng ngàn trường hợp mã tiềm năng, tìm kiếm địa điểm nơi mà mã có thể sẽ hợp lệ, sẽ sai trong 1 từ hoặc cụm từ hoặc trong cú pháp bạn đã gõ rồi đưa ra các trường hợp sửa lỗi cho bạn lựa chọn tức thì.

Cùng với chức năng đó, xCode còn cung cấp chức năng phân tích cả dự án. Tính năng này sẽ tìm thấy lỗi dài trước khi bạn kịp tìm ra nó.

1. **Các công cụ hổ trợ**

****

Hình 18: Công cụ hỗ trợ

Ngoài các tính năng được tích hợp sẳn xCode còn cung cấp 1 bộ cung cụ hổ trợ bạn đắc lực trong việc tìm kiếm mã lỗi, tìm nơi rò rĩ bộ nhớ trong mã, hay lỗi gây tắt chương trình bất thường.

**VII. Hệ thống ứng dụng “Ăn Dặm Cho Bé”.**

Để xây dựng ứng dụng ăn dặm phù hợp với thực tế, chúng ta cần khảo sát khái niệm ăn dặm, các bài báo viết về công thức, các bài chú ý khi thực hiện ăn dặm, độ tuổi phù hợp của ăn dặm.

**VII.1. Giới thiệu**

Ăn dặm là bước chuyển lớn từ chế độ ăn hoàn toàn bằng sữa sang chế độ ăn có thức ăn đặc. Đây là giai đoạn phát triển thiết yếu và thú vị đối với bé. Sữa vẫn sẽ tiếp tục là nguồn cung cấp dưỡng chất thiết yếu cho bé trong một thời gian nữa, nhưng khi bé đã trở nên lanh lợi hơn, thức ăn đặc sẽ trở thành phần chính yếu trong chế độ ăn uống cân bằng rất quan trọng đối với sức khỏe của bé.

Ăn dặm là một quá trình khá gian nan và thú vị đối với cả bố mẹ và bé. Đó là một tiến trình chuyển từ chế độ ăn dạng lỏng sang dạng sệt rồi đến dạng lợn cợn và sau cùng là dạng miếng.

Ăn dặm không đơn giản chỉ là cung cấp thêm dưỡng chất cho bé hay giúp bé ăn no, việc này còn giúp bé thực hành để phát triển và học hỏi. Bé sẽ học được cách ăn bằng muỗng (thìa) và làm quen với nhiều mùi vị và dạng thức ăn mới, quen dần với cảm giác có thức ăn trong miệng.

Trong vài tháng kế tiếp, sau khi đã làm quen với mùi vị và dạng thức ăn mới, bé sẽ phát triển kỹ năng nhai và nuốt. Điều này giúp phát triển cơ hàm để giúp bé hoàn thiện kỹ năng nói sau này.

**VII.3. Ưu nhược điểm của ứng dụng ăn dặm trên điện thoại**

**VII.3.1. Ưu điểm**

- Thuận tiện trong việc xem công thức. Có thể lưu về máy để xem offline bằng chức năng favorite.

- Dữ liệu sẽ được cập nhật liên tục.

-Có thể xem video hướng dẫn

- Xem được chỉ số của con

- Có thể comment công thức mình thích để trao đổi với người khác.

- Tổng hợp được bài viết tránh tình trạng dẫn nguồn của các báo dẫn đến trùng dữ liệu và mất thời gian của người dùng.

- Có thể xem đánh giá công thức bằng số rating.

**VII.3.2. Nhược điểm**

- Dữ liệu hiện tại còn ít

- Chưa có chức năng post bài của chuyên gia dinh dưỡng.

**VII.4. Các vấn đề khi xây dựng ứng dụng ăn dặm**

- Người lập trình chưa có nhiều kinh nghiệm chăm sóc con nhỏ nên việc hướng đên nhu cầu của các bậc cha mẹ còn thiếu sót.

- Hệ thống chỉ chạy trên di động chưa xây dựng được diễn dàn để có thể thu thập được nhiều kiến thức chăm sóc con của chuyên gia và người dùng là cha mẹ.

- Tạo ứng dụng thuận tiện cho việc xem công thức.

- Dễ dàng lưu để xem offline.

- Có thể trao đổi với người khác

- Chọn món theo số rating cao nhất

**CHƯƠNG II:** **XÂY DỰNG ỨNG DỤNG**

**I. Khảo sát tình hình thực tế**

Trong thời đại như hiện này, các bậc bố mẹ trẻ hầu như ít có kỹ năng cũng như kinh nghiệm chăm sóc con cái đặc biệt là với trẻ ở những tháng đầu đời. Vì cuộc sống quá bận rộn và việc chăm con cũng như chế độ dinh dưỡng không được dạy. Do đó, có nhiều cặp bố mẹ sẽ cho con ăn theo cách mình nghĩ là đúng và cho ăn không khoa học. Ví dụ như: cho con ăn quá bột đặc ngay từ đầu, cho con ăn số lượng quá lớn....

Chính chế độ dinh dưỡng không hợp lý dẫn đến chỉ số sức khỏe của các bé thường không đạt chuẩn.

|  |
| --- |
| http://viendinhduong.vn/FileUpload/Images/090114_041140do_thi_sdd_te_nam_2013.xls.jpg |

Hình 19: Số liệu thống kê về tình trạng dinh dưỡng trẻ em dưới 5 tuổi qua các năm (1999-2013)

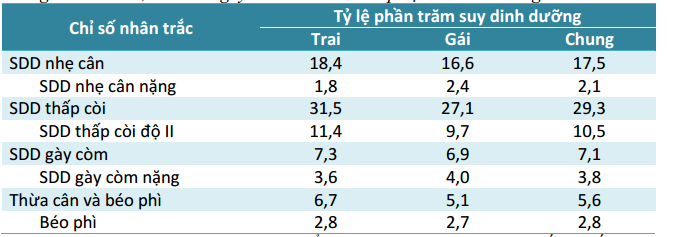
Theo báo cáo của tổng cục điều tra năm 2009 đến 2010 thì tỷ lệ nuôi con hoàn toàn bằng sữa mẹ cho đến 4 tháng tuổi vẫn còn thấp so với mục tiêu quốc gia giai đoạn 2001-2010. Tỷ lệ này đã giảm từ 31,1% vào năm 2000 xuống còn 28,8% vào năm 2010. Tỷ lệ trẻ được bú mẹ hoàn toàn cho đến 6 tháng tuổi là 19,6%.



Hình 20: Tình hình bú mẹ đến 6 tháng tuổi

**Tình trạng dinh dưỡng trẻ em**

- Tình trạng dinh dưỡng của trẻ đã được cải thiện sau 10 năm. Việt nam đã đạt được mục tiêu thiên niên kỷ về giảm tỷ lệ SDD thể nhẹ cân. Tuy nhiên, SDD thể nhẹ cân vẫn ở cấp YNSKCĐ trung bình; SDD thể thấp còi ở ranh giới giữa cấp độ YNSKCĐ trung bình và cao; SDD thể gày còm vẫn nằm ở cấp độ YNSKCĐ trung bình.

****

Hình 21: Tỉ lệ phần trăm suy dinh dưỡng năm 2013

- Tỷ lệ thừa cân-béo phì ở trẻ dưới 5 tuổi có xu hướng gia tăng, đặc biệt ở các thành phố lớn như Hà Nội và TP Hồ Chí Minh.

Chính những báo cáo tổng hợp nói trên cho thấy việc dinh dưỡng không hợp lý dẫn đễn tình trạng các bé sẽ dễ bị béo phì hoặc thiếu cân nặng cũng như chiều cao không đạt chuẩn.

1. **Mô tả bài toán**

Qua khảo sát tình hình thực trạng dinh dưỡng trên ta thấy có nhiều vấn đề còn bất cập, vấn đề đặt ra là phải xây dựng một ứng dụng ăn dặm cho bé có nhiều ưu điểm đễ hỗ trợ người dùng được nhiều hơn. Giúp người dùng tiếp cận kiến thức dễ dàng, tìm ra lỗi sai trong quá trình nấu ăn dặm cho bé, theo sát quá trình phát triển của bé để có những sửa đổi chế độ ăn cho bé phù hợp.

**II.1. Cấu trúc hệ thống**

Với các chức năng được mô tả trong phần giới thiệu thì hệ thống sẽ gồm 3 thành phần chính với các nhiệm vụ sau đây:

* Phần ứng dụng di động được phát triển trên nền tảng Iphone nhằm hổ trợ cho người sử dụng xem công thức, chú ý, video, chỉ số của con.
* Trang web cho nhà quản lý hổ trợ quản lý thư viện công thức, thư viện ghi chú, thư viện video, thư viện chỉ số của bé đồng thời quản lý thông tin người sử dụng.
* Phần Web Services được dùng để kết nối, lấy và cập nhật dữ liệu giữa ứng dụng di động với cơ sở dữ liệu.

Hình ảnh dưới đây sẽ minh họa cho cấu trúc của hệ thống

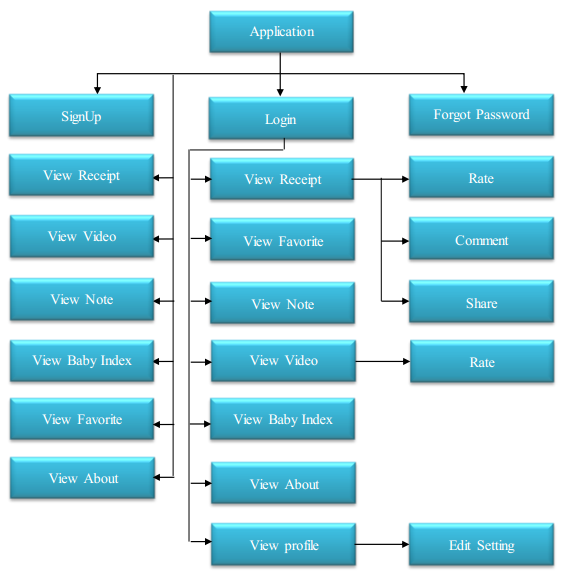


Hình 22: Cấu trúc hệ thống

**II.2. Yêu cầu chức năng**

**II.2.1. Ứng dụng trên thiết bị iOS**

Sau đây là danh sách các chức năng chính của chương trình.



Hình 23: Sơ đồ phân rã chức năng của ứng dụng trên iOS

**II.2.2. Web (Back-end)**

Phần Back-end có các chức năng tương ứng với 2 nhóm đối tượng chính là quản trị viên hệ thống và nhân viên nhập dữ liệu hệ thống. Trong đó:

+ Quản trị viên sẽ có quyền cao nhất, quản lý, cấu hình toàn bộ hệ thống bao gồm webservice.

+ Nhóm đối tượng nhân viên nhập dữ liệu hệ thống truy cập vào hệ thống web bằng user và pass đã được cung cấp nhằm thực hiện nhiệm vụ thêm và chỉnh sửa dữ liệu trắc nghiệm.

Phần báo cáo dưới đây sẽ giới thiệu các chức năng chính của các đối tượng trên.

**1. Quản trị viên**

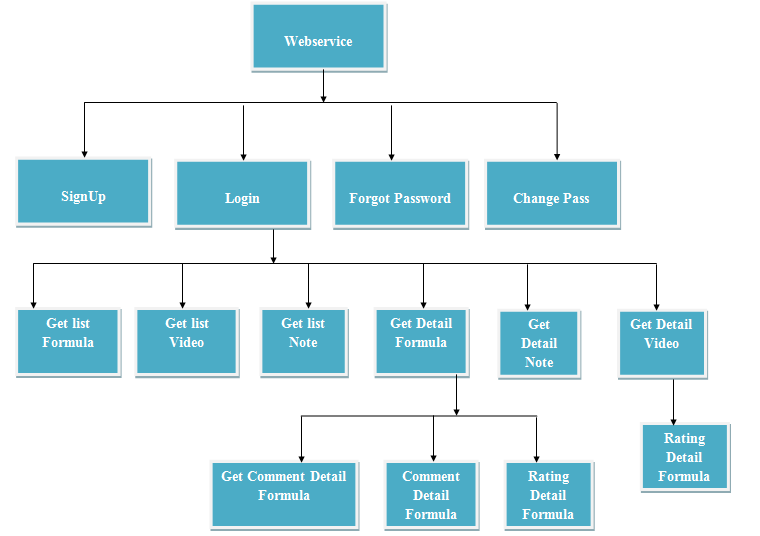
* Quản lý thành viên
* Quản lý toàn bộ dữ liệu trắc nghiệm.

**2. Nhân viên nhập dữ liệu**

* Quản lý tài khoản
* Nhập dữ liệu trắc nghiệm.
* Kiểm tra trùng dữ liệu (xem câu mới nhập đã tồn tại chưa).
* Quản lý dữ liệu đã nhập

#### II.2.3. Web service

Web service nhằm mục đích sử lý yêu cầu và truy vấn database trả về kết quả cho ứng dụng trên thiết bị di động (iOS) nên sẽ có các chức năng theo sơ đồ phân rã sau:



Hình 24: Sơ đồ phân rã chức năng Web service

* Hàm SignUp: Đăng ký thành viên mới với thông tin username, email và password, phone, giới tính, tên của bé, ngày sinh của bé, avatar. Trả về kết quả userid và mã kết quả.
* Hàm Login: Tiến hành login cho user với username và password sau đó sinh mã kết quả trả về.
* Hàm Forgot Password: Cho người dùng lấy lại mật khẩu với thông tin email. Kết quả sẽ được gửi vào mail người dùng.
* Hàm Change Password: Cho phép người dùng thay đổi mật khẩu hiện tại với 4 thông tin truyền vào là CurrentPass, NewPass, ConfirmPass, UserID. Kết quả trả về là mã kết quả.
* Hàm Get List Formula: Lấy danh sách các công thức nấu ăn dặm. Kết quả trả về là danh sách các món.
* Hàm Get List Video: Lấy danh sách các video hướng dẫn nấu ăn dặm. Kết quả trả về là danh sách các video.
* Hàm Get List Note: Lấy danh sách các chú ý khi nấu ăn dặm. Kết quả trả về là danh sách các chú ý.
* Hàm Get Detail Formula: Lấy chi tiết của công thức nấu ăn dặm theo ID. Kết quả trả về là chi tiết của công thức.
* Hàm Get Detail Video: Lấy chi tiết của video nấu ăn dặm theo ID. Kết quả trả về là chi tiết của video.
* Hàm Get Detail Note: Lấy chi tiết của chú ý ăn dặm theo ID. Kết quả trả về là chi tiết của chú ý.
* Hàm Get Comment Detail Formula: Cho người dùng được phép xem comment công thức theo ID của công thức.
* Hàm Comment Detail Formula: Cho người dùng được phép comment công thức theo ID của công thức.
* Hàm Rating Detail Formula: Cho người dùng được phép rating công thức theo ID của công thức.

### II.3. Yêu cầu phi chức năng

Ứng dụng được áp dụng cần một số yêu cầu phi chức năng như sau :

* Chạy nhanh.
* Tính chính xác.
* Giao diện đẹp, bố trí hợp lý, dễ dàng sử dụng, không gây nhàm chán cho người dùng.
* Tối ưu hóa dung lượng có thể, vì dung lượng bộ nhớ điện thoại không như máy tính.
* Server thi trực tuyến chạy ổn định.
* Khi không kết nối mạng vẫn đảm bảo các chưc năng hoạt động tốt.

**III.Phân tích và thiết kế chương trình**

**III.1. Mô hình Use case**

Ca sử dụng thể hiện sự tương tác đặc trưng giữa người dùng và hệ thống, tập

hợp tất cả các ca sử dụng của hệ thống sẽ mô tả tất cả các trường hợp mà người

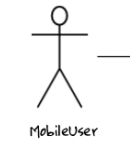
dùng có thể tương tác với hệ thống.

**III.1.1. Nhận dạng các tác nhân (actor)**

Hệ thống gồm 3 tác nhân chính là người dùng (trên thiết bị Iphone), nhân viên

nhập dữ liệuvà quản trị viên.

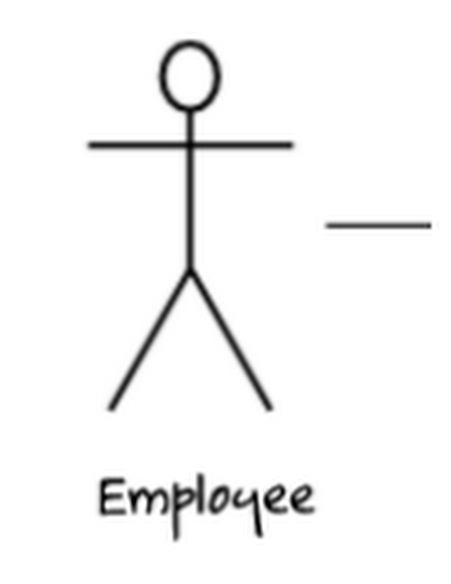
1. **Người dùng**

****

Hình 25: Tác nhân MobileUser

Người dùng là người sử dụng các chức năng chính của hệ thống trên các thiết bị Iphone. Họ có nhu cầu luyện kỷ năng tiếng anh thông qua các phần thi trắc nghiệm từ phần mềm cung cấp.

1. **Nhân viên nhập dữ liệu**

****

Hình 26: Tác nhân Nhân viên

Nhân viên nhập dữ liệu là người được thuê có kiến thức về dinh dưỡng cho trẻ với trách nhiệm nhập dữ liệu câu hỏi vào hệ thống và đảm bảo độ chính xác của thông tin.

1. **Quản trị viên**

****

Hình 27: Tác nhân Admin

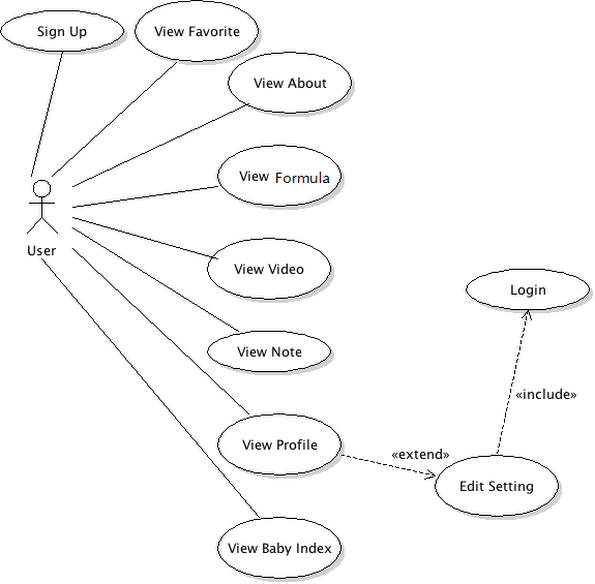
Quản trị viên là người quản lý hệ thống, có thể sửa chữa tất cả thông tin của

người sử dụng, người nhập liệu. Quản lý các thiết lập của các thể loại trắc nghiệm

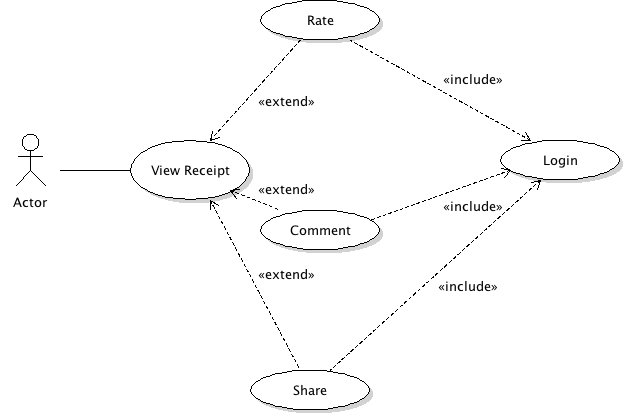
cũng như đảm bảo webservice hoạt động.

**III.1.2. Biểu đồ Use case của ứng dụng trên iOS**

1. **Use Case mức 0**

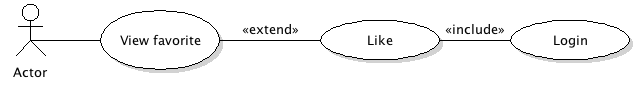
****

Hình 28: Biểu đồ ca sử dụng mức 0 của tác nhân Người dùng

Ca sử dụng: Ca sử dụng xem, comment và rating công thức 

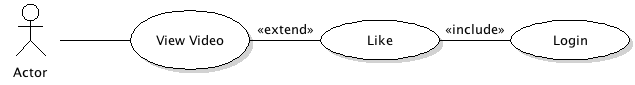
Hình 29: Ca sử dụng xem, comment và rating công thức

Ca sử dụng: Ca sử dụng favorite

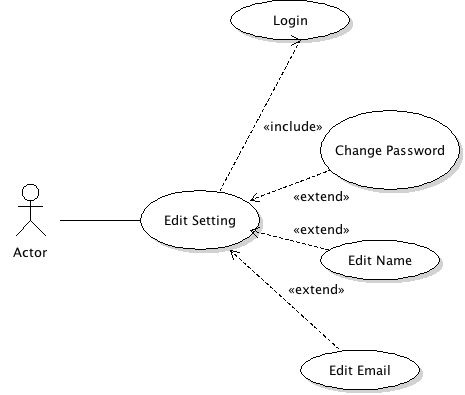


Hình 30: Ca sử dụng favorite

Ca sử dụng: Ca sử dụng favorite



Hình 31: Ca sử dụng xem, comment và rating công thức

Ca sử dụng: Chỉnh sửa thông tin cá nhân

Hình 32: Ca sử dụng chỉnh sửa thông tin cá nhân

**III.1.3.Biểu đồ triển khai**

****

Hình 33: Biểu đồ triển khai

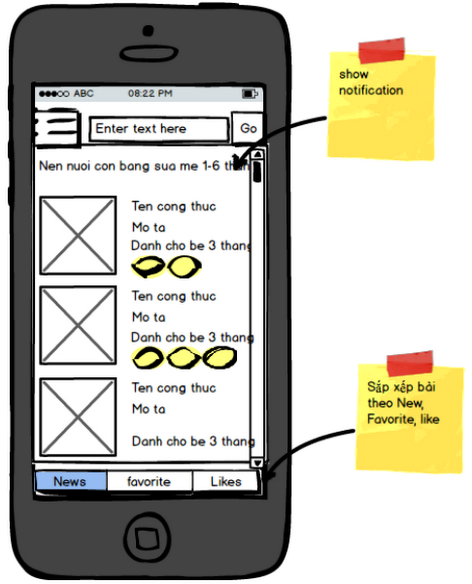
* **Interface** : Hiện thị giao diện.
* **Process** : Xử lý các tác vụ.
* **Database** : Cơ sở dữ liệu.

**III.2. Thiết kế giao diện**

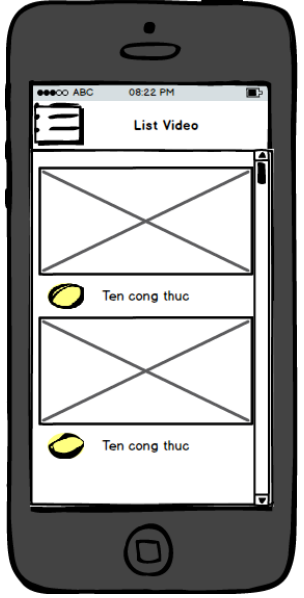
Trong chương trình này nhóm sử dụng triển khai giao diện Prototype với MockUp. Sau đây là một số màn hình chính.

****

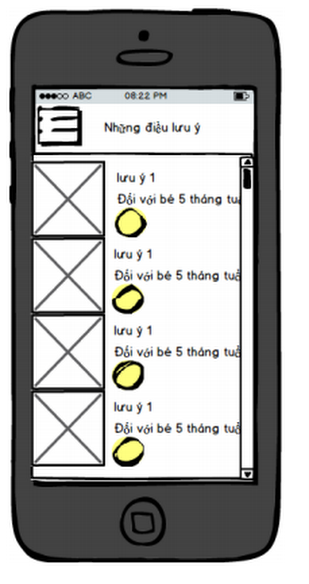
Hình 34: Giao diện prototype chức năng Login

****

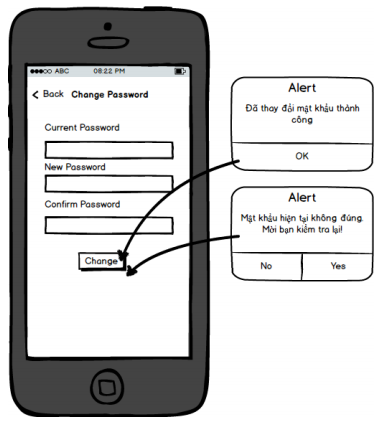
Hình 35: Giao diện prototype chức năng danh sách công thức

****

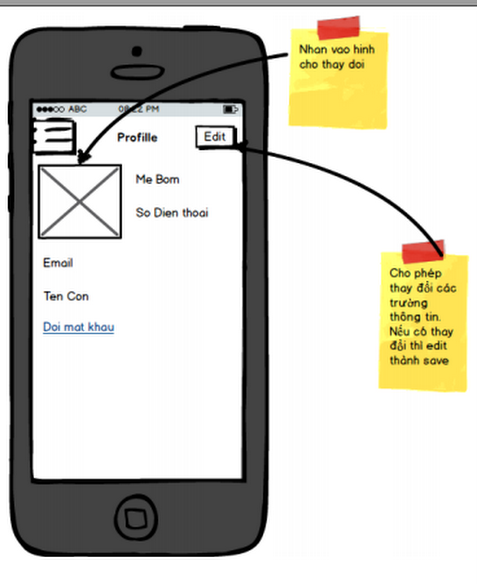
Hình 36: Giao diện prototype chức năng danh sách Video



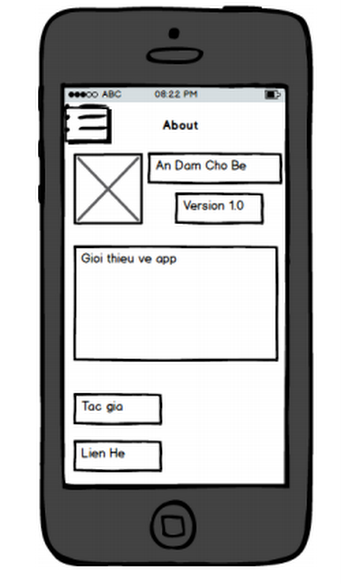
Hình 37: Giao diện prototype chức năng danh sách Lưu ý



Hình 38: Giao diện prototype chức năng Đổi mật khẩu



Hình 39: Giao diện prototype chức năng xem thông tin người dùng



Hình 40: Giao diện prototype chức năng Thông tin

**III.3. Cơ sở dữ liệu**

Qua nghiên cứu các tính năng và yêu cầu của ứng dụng, hệ thống sẽ gồm 2 cơ sở dữ liệu bao gồm :

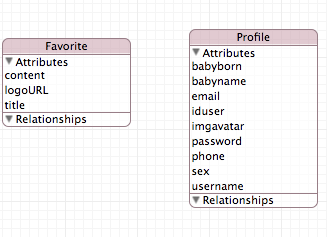
**1. Cở sở dữ liệu offline :** Sử dụng Coredata trên iOS hổ trợ để lưu trữ một lượng khá lớn dữ liệu có sẵn để người dùng có thể thi trong trường hợp không thể kết nối internet, số lượng các bảng cần thiết và quan hệ là giống hệt với CSDL được đặt trên server nhưng có giảm một vài trường không cần thiết nhằm tối ưu cho hệ thống, đồng thời vì giới hạn của những thiết bị di động.

**2. Cơ sở dữ liệu online :** Sử dụng mySql, một hệ CSDL rất mạnh mẽ, thông

dụng hiện nay và miễn phí. CSDL thiết kế đặt trên server có thêm bảng

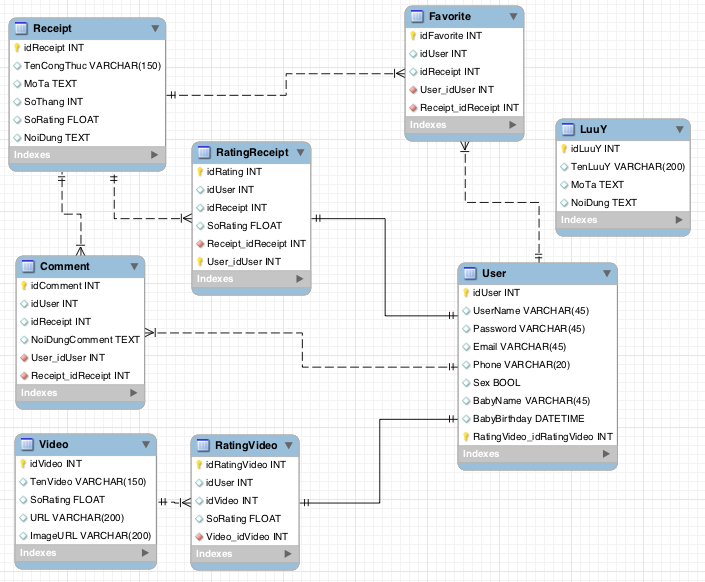
Admins để chứa thông tin cho Admin đăng nhập quản lý hệ thống.

**III.3.1.Mô hình cơ sở dữ liệu Core Data**

****

Hình 41: Mô hình cơ sở dữ liệu quan hệ SQL trên server

**III.3.2.Mô hình cơ sở dữ liệu quan hệ SQL trên server**

****

Hình 42: Mô hình cơ sở dữ liệu quan hệ SQL trên server

## CHƯƠNG III: TRIỂN KHAI ỨNG DỤNG

## I. Hướng dẫn cài đặt

### I.1.1. Yêu cầu tài nguyên

* Windows XP hay Vista, Windown 7.
* Mac OS X 10.6 trở lên.
* Thiết bị di động iOS 4.0 trở lên.

### I.1.2. Công cụ

* xCode (có thể download tại đây: https://developer.apple.com/xcode/)

**II. Minh họa chạy chương trình**

****

Hình 43: Chức năng Login

Chức năng danh sách công thức.

****

Hình 44: chức năng danh sách công thức ăn dặm

Chức năng danh sách công thức ăn dặm yêu thích.



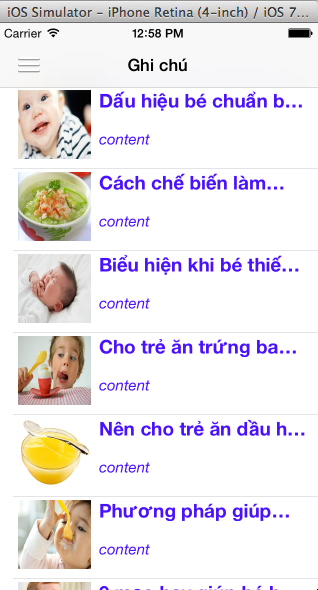
Hình 45: chức năng danh sách công thức ăn dặm yêu thích

Chức năng xem chi tiết công thức.



Hình 46: chức năng chi tiết công thức ăn dặm

Chức năng danh sách ghi chú khi nấu ăn dặm.



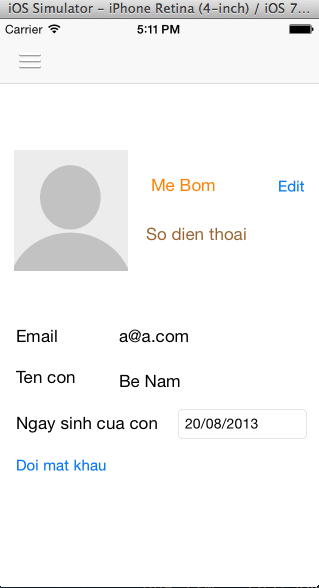
Hình 47: Danh sách công thức cần chú ý

Chức năng danh sách video hướng dẫn.



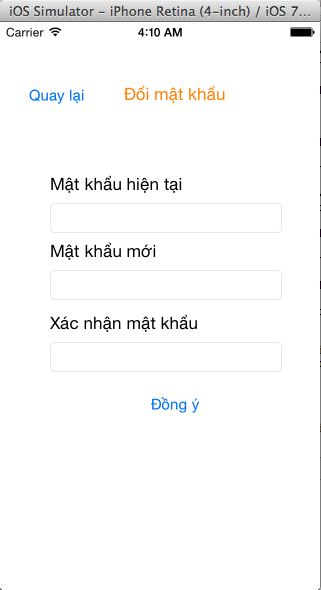
Hình 48: Danh sách video hướng dẫn nấu ăn dặm

Chức năng xem thông tin của người dùng.



Hình 49: Chức năng thông tin người dùng

Chức năng đổi mật khẩu của người dùng.



Hình 50: Chức năng đổi mật khẩu

**CHƯƠNG IV: KẾT LUẬN**

**I. Đánh giá kết quả**

Hiện tại chưa có ứng dụng ăn dặm cho bé trên di động (Khảo sát trên iOS lẫn Android). Nên ý tưởng đưa ra sẽ có sức hút nhất định với các bậc cha mẹ có con nhỏ. Tuy nhiên, cũng không thể tránh được thiếu sót. Dưới đây, chúng ta cùng nhìn lại những kết quả cũng như hạn chế.

**I.1. Kết quả đạt được**

Trong thời gian học tập và nghiên cứu làm đồ án tốt nghiệp, nhóm đã nhận được nhiều sự giúp đỡ nhiệt tình của thầy cô cũng như các bạn sinh viên để hoàn thành đề tài, qua quá trình làm đồ án tốt nghiệp xây dựng ứng dụng ăn dặm cho bé trên thiết bị iOS nhóm đã thu được kết quả sau:

* Cách thức xây dựng một ứng dụng dựa trên nền hệ điều hành iOS
* Cách thức xây dựng webservice và đưa lên host.
* Xây dựng được ứng dụng ăn dặm cho bé trên điện thoại di động
* Chương trình có nhiều chức năng thuận tiện cho người dùng
* Ứng dụng hỗ trợ các chức năng share, comment để người dùng có thể đánh giá và trao đổi với các bố mẹ khác.
* Giao diện chương trình được thiết kế tiện lợi.

**I.2. Hạn chế**

Tuy đã đạt được những kết quả nhất định, nhưng chương trình vẫn có những hạn chế.

* Nguồn dữ liệu chưa được phong phú.
* Vẫn còn những tính năng chưa được thêm để làm ứng dụng hấp dẫn và thiết thực hơn. Như thiết kế comment ở cả chắc năng xem "Lưu ý".

**II. Hướng phát triển**

Chăm sóc con là một nhiệm vụ thiêng liêng của các bậc bố mẹ. Do đó, họ sẽ rất quan tâm đến chệ độ ăn dinh dưỡng của con ngay cả những tháng đầu đời. Vì vậy, khi thực hiện đề tài nhóm mong đợi có thể góp phần vào việc đưa ra một cách tham khảo đáng tin cậy của các bậc bố mẹ. Ngoài ra, những hướng phát triển tiếp theo sẽ là:

* Xây dựng ứng dụng có chương trình đẹp hơn về dao diện, lôi cuốn về nội dung.
* Phát triển ứng dụng trên hệ các dòng điện thoại android, Window Mobile và web.
* Bổ sung thêm những tính năng mới hỗ trợ người dùng:

+ Thêm chức năng comment ở Lưu ý.

+ Thêm chức năng rating cho lưu ý

+ Thêm chức năng cho người dùng post công thức của riêng họ.

+ Thêm xem chỉ số của con và biểu đồ

* Ngân hàng dữ liệu phong phú.
* Xây dựng ứng dụng có giao diện đẹp.

**PHỤ LỤC**

**PHỤ LỤC A. Môi trường và công cụ sử dụng để phát triển ứng dụng**

* **Môi trường phát triển ứng dụng**

|  |  |  |
| --- | --- | --- |
| STT | Tên | Mô tả |
| 1 | Hệ điều hành máy tính | Mac OS 10.8 |
| 2 | Hệ điều hành di động | iOS 7.0 trở lên |

Bảng 1. Môi trường phát triển ứng dụng

* **Công cụ sử dụng để phát triển ứng dụng**

|  |  |  |
| --- | --- | --- |
| **STT** | Tên | Mô tả |
| **1** | xCode | 5.1.1 |
| **2** | Adobe Photoshop | 6.0 |
| **3** | Notepad++ |  |
| **4** | Edraw UML Diagram | 5.1 |
| **5** | **MySQL workbench** | **5.2CE** |

Bảng 1. Môi trường phát triển ứng dụng

**PHỤ LỤC B. Các từ viết tắt và thuật ngữ riêng**

|  |  |
| --- | --- |
| **KÝ HIỆU** | **MÔ TẢ** |
| LBS | Location – Base Services |
| HĐH | Hệ điều hành |
| IDE | Intergrated Development Environment |
| VM | Virtual Machine |
| IDL | Interface Description Language |
| iOS | Hệ điều hành trên các thiết bị di động của Apple |
| IPC | Interprocess Communication |
| GPS | Global Positioning System |

Bảng 2. Các từ viết tắt

**TÀI LIỆU THAM KHẢO**

**[1] Trang web** [**https://www.google.com.vn**](https://www.google.com.vn)

**[2] Trang web http://www.raywenderlich.com**

**[3] Trang web http://www.youtube.com**

**[4] Trang web http://www.dinhduongchobe.org**

**[4] Trang web http://** [**http://phunutoday.vn**](http://phunutoday.vn)

**[5] Giáo trình xcode của trung tâm Iviettech.**

**[6] Báo cáo tốt nghiệp của thầy Nguyễn Thanh Liêm.**