**TRƯỜNG ĐẠI HỌC XÂY DỰNG HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN CÔNG NGHỆ PHẦN MỀM**

---------------🙦 🕮 🙤---------------



**BÁO CÁO**

**THỰC TẬP CÁN BỘ KỸ THUẬT**

|  |  |
| --- | --- |
| **Sinh viên thực tập** | Lương Bá Phan Anh |
| **Mã số sinh viên** | 0179166 |
| **Lớp** | 66PM3 |
| **Giảng viên phụ trách** | ThS. Phan Hữu Trung |
| **Bộ môn** | Công nghệ phần mềm |
| **Địa điểm thực tập** | Công ty cổ phần VCCorp |

**HÀ NỘI – 16/09/2021**

**Lời cảm ơn**

Trước tiên với tình cảm sâu sắc và chân thành nhất, cho phép em được bày tỏ lòng biết ơn đến tất cả các cấp lãnh đạo, anh chị em nhân viên đang công tác tại Công ty cổ phần VCCorp đã tạo điều kiện hỗ trợ, giúp đỡ em trong suốt quá trình học tập và nghiên cứu đề tài này. Trong suốt thời gian từ khi bắt đầu học tập tại trường đến nay, em đã nhận được rất nhiều sự quan tâm, giúp đỡ của quý Thầy Cô và bạn bè.

Với lòng biết ơn sâu sắc nhất, em xin gửi đến quý Thầy Cô ở Khoa Công nghệ thông tin trường Đại học Xây Dựng đã truyền đạt vốn kiến thức quý báu cho chúng em trong suốt thời gian học tập tại trường. Nhờ có những lời hướng dẫn, dạy bảo của các thầy cô nên đề tài nghiên cứu của em mới có thể hoàn thiện tốt đẹp.

Một lần nữa, em xin chân thành cảm ơn thầy Phan Hữu Trung – người đã trực tiếp giúp đỡ, quan tâm, hướng dẫn em hoàn thành tốt bài báo cáo này trong thời gian qua.

Bài báo cáo thực tập thực hiện trong khoảng thời gian gần 5 tuần. Bước đầu đi vào thực tế của em còn hạn chế và còn nhiều bỡ ngỡ nên không tránh khỏi những thiếu sót, em rất mong nhận được những ý kiến đóng góp quý báu của quý Thầy Cô để kiến thức của em trong lĩnh vực này được hoàn thiện hơn đồng thời có điều kiện bổ sung, nâng cao ý thức của mình.

Em xin chân thành cảm ơn!

**Mục lục**

[CHƯƠNG I. TỔNG QUAN VỀ CƠ SỞ THỰC TẬP 1](#_Toc82781767)

[1. GIỚI THIỆU CHUNG](#_Toc82781768) 2

[CHƯƠNG II. CƠ SỞ LÝ THUYẾT 3](#_Toc82781774)

[1. Java 3](#_Toc82781775)

[4. GIT 3](#_Toc82781778)

[5. MICROSOFT SQL SERVER 5](#_Toc82781779)

[6. SQL SERVER MANAGERMENT STUDIO 6](#_Toc82781780)

[CHƯƠNG III. NỘI DUNG NGHIÊN CỨU 7](#_Toc82781782)

[1. NỘI DUNG THỰC TẬP 7](#_Toc82781783)

[QUÁ TRÌNH THỰC HIỆN 7](#_Toc82781784)

[1.1. Tuần 1: Nghiên cứu tài liệu, tự học và làm bài tập về Java cơ bản. 7](#_Toc82781785)

[1.2. Tuần 2: Nghiên cứu tài liệu, tự học và làm bài tập về Java nâng cao. 8](#_Toc82781786)

[1.3. Tuần 3: SQl, liên kết CSDL. 14](#_Toc82781787)

[1.4. Tuần 4: Tìm hiểu lập trình phân tán, Hadoop. 19](#_Toc82781788)

[1.5. Tuần 5: Làm bài tập lập trình phân tán, Hadoop. 20](#_Toc82781789)

[CHƯƠNG IV. KẾT LUẬN VÀ KIẾN NGHỊ 21](#_Toc82781790)

[1. KẾT LUẬN 21](#_Toc82781791)

[2. KIẾN NGHỊ 21](#_Toc82781792)

# **CHƯƠNG I. TỔNG QUAN VỀ CƠ SỞ THỰC TẬP**

## **GIỚI THIỆU CHUNG**

**Tên công ty:** Công ty cổ phần VCCorp

**Trụ sở chính:** Toà nhà Center Building - Hapulico Complex, Số 1 Nguyễn Huy Tưởng, Thanh Xuân, Hà Nội.

**Điện** **thoại**: +84 24 7300 6886

**Website**: vccorp.vn

Được thành lập vào năm 2006, sau 18 năm phát triển, Công ty CP VCCorp (VCCorp) là một trong những công ty tiên phong trong lĩnh vực công nghệ và công nghệ cao ở Việt Nam. VCCorp đã xây dựng và ứng dụng thành công hạ tầng cloud computing có khả năng lưu trữ và tính toán lượng dữ liệu lớn (Big Data, Data mining) hỗ trợ cho nền tảng công nghệ quảng cáo trực tuyến Admicro, phục vụ trên 50 triệu người dùng (chiếm trên 90% tổng số người dùng Internet tại Việt Nam) với hơn 200 website uy tín, gần 30 website trong số đó thuộc sở hữu độc quyền, phục vụ hàng tỷ lượt xem mỗi tháng. Ngoài ra, VCCorp nằm trong top 3 công ty phát hành game trên nền tảng di động với thương hiệu Sohagame, và mới đây hệ sinh thái chuyển đổi số Bizfly đang được VCCorp phát triển và triển khai mạnh mẽ. Đồng thời VCCorp cũng là đơn vị xây dựng Lotus - mạng xã hội do người Việt Nam làm chủ.

Dẫn đầu thị trường Internet tại Việt Nam, VCCorp là người đi tiên phong trong tất cả các lĩnh vực phục vụ trên 50 triệu người dùng internet, 45 triệu người dùng mobile, hơn 2000 nhân sự đang làm việc tại Hà Nội và Thành Phố HCM. VCCorp đã xây dựng được một hệ sinh thái vững chắc trên thị trường.

# **CHƯƠNG II. CƠ SỞ LÝ THUYẾT**

## **Java**

Java là một trong những ngôn ngữ lập trình hướng đối tượng. Nó được sử dụng trong phát triển phần mềm, trang web, game hay ứng dụng trên các thiết bị di động.

Java được khởi đầu bởi James Gosling và bạn đồng nghiệp ở Sun MicroSystem năm 1991. Ban đầu Java được tạo ra nhằm mục đích viết phần mềm cho các sản phẩm gia dụng, và có tên là Oak.

Java được phát hành năm 1994, đến năm 2010 được Oracle mua lại từ Sun MicroSystem.

Java được tạo ra với tiêu chí “Viết (code) một lần, thực thi khắp nơi” (Write Once, Run Anywhere – WORA). Chương trình phần mềm viết bằng Java có thể chạy trên mọi nền tảng (platform) khác nhau thông qua một môi trường thực thi với điều kiện có môi trường thực thi

* Tương tự C++, hướng đối tượng hoàn toàn.
* Độc lập phần cứng và hệ điều hành.
* Cơ chế thu gom rác tự động.
* Đa luồng.
* An toàn và bảo mật.

## **GIT**

Git ([/ɡɪt/](https://en.wikipedia.org/wiki/Help:IPA/English)) là [phần mềm quản lý mã nguồn](https://vi.wikipedia.org/w/index.php?title=Ph%E1%BA%A7n_m%E1%BB%81m_qu%E1%BA%A3n_l%C3%BD_m%C3%A3_ngu%E1%BB%93n&action=edit&redlink=1) [phân tán](https://vi.wikipedia.org/w/index.php?title=Ph%C3%A2n_t%C3%A1n&action=edit&redlink=1) được phát triển bởi [Linus Torvalds](https://vi.wikipedia.org/wiki/Linus_Torvalds) vào năm [2005](https://vi.wikipedia.org/wiki/2005), ban đầu dành cho việc phát triển [nhân Linux](https://vi.wikipedia.org/wiki/H%E1%BA%A1t_nh%C3%A2n_Linux). Hiện nay, Git trở thành một trong các phần mềm quản lý mã nguồn phổ biến nhất. Git là phần mềm mã nguồn mở được phân phối theo giấy phép công cộng [GPL2](https://vi.wikipedia.org/wiki/Gi%E1%BA%A5y_ph%C3%A9p_C%C3%B4ng_c%E1%BB%99ng_GNU).

Git có khả năng chạy trên nhiều hệ điều hành khác nhau như [Linux](https://vi.wikipedia.org/wiki/Linux), [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), [Mac OSX](https://vi.wikipedia.org/wiki/OS_X) v.v..

Khái niệm cơ bản trong GIT:

* Repository
* Repository (nhà kho) hay được gọi tắt là Repo đơn giản là nơi chứa/cơ sở dữ liệu (database) tất cả những thông tin cần thiết để duy trì và quản lý các sửa đổi và lịch sử của dự án.
* Trong Repo có 2 cấu trúc dữ liệu chính là Object Store và Index. Tất cả dữ liệu của Repo đều được chứa trong thư mục bạn đang làm việc dưới dạng folder ẩn có tên là.git (không có phần tên trước dấu chấm).
* Object store
* Object store là trái tim của Git, nó chứa dữ liệu nguyên gốc (original data files), các file log ghi chép quá trình sửa đổi, tên người tạo file, ngày tháng và các thông tin khác. Git có bốn loại object là: Blobs, Trees, Commits, Tags
* Blobs: là file nhị phân có thể chứa được mọi loại dữ liệu bất kể là dữ liệu của chương trình gì.
* Trees: lớp đại diện cho thông tin thư mục như thông tin định danh của blob, đường dẫn, chứa một ít metadata chứa thông tin cấu trúc và các thư mục nhỏ có trong thư mục đó.
* Commits: Chứa metadata có thông tin về mọi thứ như tên tác giả, người tải lên (commiter), ngày tải lên, thông tin log...
* Tags: đánh dấu cho dễ đọc. Thay vì một cái tên dài như là 9da581d910c9c4ac93557ca4859e767f5caf5169, chúng ta có thể tên tag là Ver-1.0- Alpha. Dễ nhớ và dễ sử dụng hơn.
* Index
* Index là file nhị phân động và tạm thời miêu tả cấu trúc thư mục của toàn bộ Repo và trạng thái của dự án được thể hiện thông qua commit và tree tại một thời điểm nào đó trong lịch sử phát triển. Git là một hệ thống truy tìm nội dung (content tracking system).
* Index không chứa nội dung file mà chỉ dùng để truy tìm (track) những thứ mà bạn muốn commit.

## **MICROSOFT SQL SERVER**

Microsoft SQL Server là một [hệ quản trị cơ sở dữ liệu quan hệ](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng_qu%E1%BA%A3n_l%C3%BD_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u_quan_h%E1%BB%87) được phát triển bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft). Là một [máy chủ](https://vi.wikipedia.org/wiki/M%C3%A1y_ch%E1%BB%A7) [cơ sở dữ liệu](https://vi.wikipedia.org/wiki/C%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u), nó là một [sản phẩm phần mềm](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m) có chức năng chính là lưu trữ và truy xuất dữ liệu theo yêu cầu của các [ứng dụng phần mềm](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_%E1%BB%A9ng_d%E1%BB%A5ng) khác. Có thể chạy trên cùng một máy tính hoặc trên một máy tính khác trên mạng (bao gồm cả Internet).

Microsoft tiếp thị ít nhất một chục phiên bản Microsoft SQL Server khác nhau, nhắm vào các đối tượng khác nhau và cho khối lượng công việc khác nhau, từ các ứng dụng máy đơn nhỏ đến các ứng dụng Internet lớn có nhiều [người dùng đồng thời](https://vi.wikipedia.org/w/index.php?title=Ng%C6%B0%E1%BB%9Di_s%E1%BB%AD_d%E1%BB%A5ng_%C4%91%E1%BB%93ng_th%E1%BB%9Di&action=edit&redlink=1).

Lịch sử của Microsoft SQL Server bắt đầu với sản phẩm Microsoft SQL Server đầu tiên SQL SQL Server 1.0, máy chủ [16-bit](https://vi.wikipedia.org/w/index.php?title=16-bit&action=edit&redlink=1) cho [hệ điều hành OS/2](https://vi.wikipedia.org/wiki/OS/2) vào năm 1989 và kéo dài đến ngày hiện tại.

Các mốc quan trọng

MS SQL Server cho OS/2 bắt đầu như một dự án chuyển [Sybase SQL Server](https://vi.wikipedia.org/w/index.php?title=M%C3%A1y_ch%E1%BB%A7_SQL_Sybase&action=edit&redlink=1) sang OS/2 vào năm 1989, bởi [Sybase](https://vi.wikipedia.org/w/index.php?title=Sybase&action=edit&redlink=1), [Ashton-Tate](https://vi.wikipedia.org/w/index.php?title=Ashton-Tate&action=edit&redlink=1) và [Microsoft](https://vi.wikipedia.org/wiki/Microsoft).

SQL Server 4.2 cho NT được phát hành vào năm 1993, đánh dấu mục nhập vào [Windows NT](https://vi.wikipedia.org/wiki/Windows_NT).

SQL Server 6.0 được phát hành vào năm 1995, đánh dấu sự kết thúc hợp tác với [Sybase](https://vi.wikipedia.org/w/index.php?title=Sybase&action=edit&redlink=1); Sybase sẽ tiếp tục phát triển biến thể SQL Server của riêng họ, Sybase [Adaptive Server Enterprise](https://vi.wikipedia.org/w/index.php?title=Doanh_nghi%E1%BB%87p_m%C3%A1y_ch%E1%BB%A7_th%C3%ADch_%E1%BB%A9ng&action=edit&redlink=1), độc lập với Microsoft.

SQL Server 7.0 được phát hành vào năm 1998, đánh dấu việc chuyển đổi mã nguồn từ C sang C ++.

SQL Server 2005, được phát hành năm 2005, hoàn thành việc sửa đổi hoàn toàn mã Sybase cũ thành mã Microsoft.

SQL Server 2017, được phát hành vào năm 2017, bổ sung hỗ trợ Linux cho các nền tảng Linux này: [Red Hat Enterprise Linux](https://vi.wikipedia.org/wiki/Red_Hat_Enterprise_Linux), [SUSE Linux Enterprise Server](https://vi.wikipedia.org/w/index.php?title=M%C3%A1y_ch%E1%BB%A7_doanh_nghi%E1%BB%87p_Linux&action=edit&redlink=1), [Ubuntu](https://vi.wikipedia.org/wiki/Ubuntu) & [Docker Engine](https://vi.wikipedia.org/wiki/Docker_(ph%E1%BA%A7n_m%E1%BB%81m)).

## **SQL SERVER MANAGERMENT STUDIO**

SQL Server Management Studio (SSMS) là một phần mềm được khởi chạy lần đầu tiên với Microsoft SQL Server 2005, được sử dụng để định cấu hình, quản lý và quản trị tất cả các thành phần trong Microsoft SQL Server. Nó là sự kế thừa của Enterprise Manager trong SQL 2000 trở về trước. Công cụ này bao gồm cả trình chỉnh sửa tập lệnh và công cụ đồ họa hoạt động với các đối tượng và tính năng của máy chủ. [2] Một tính năng trung tâm của SSMS là Object Explorer, cho phép người dùng duyệt, chọn và hành động trên bất kỳ đối tượng nào trong máy chủ. [3] Nó cũng cung cấp một phiên bản Express riêng có thể được tải xuống miễn phí, tuy nhiên các phiên bản gần đây của SSMS hoàn toàn có khả năng kết nối và quản lý bất kỳ phiên bản SQL Server Express nào. Microsoft cũng kết hợp khả năng tương thích ngược cho các phiên bản SQL Server cũ hơn, do đó cho phép phiên bản SSMS mới hơn kết nối với các phiên bản SQL Server cũ hơn. Nó cũng đi kèm với Microsoft SQL Server Express 2012 hoặc người dùng có thể tải xuống riêng. Bắt đầu từ phiên bản 11, ứng dụng dựa trên Visual Studio 2010 shell, sử dụng WPF cho giao diện người dùng. Các phiên bản 18 trở lên dựa trên Visual Studio 2017. Isolated Shell. [4] Vào tháng 6 năm 2015, Microsoft đã công bố ý định phát hành các phiên bản tương lai của SSMS độc lập với các bản phát hành công cụ cơ sở dữ liệu SQL Server.

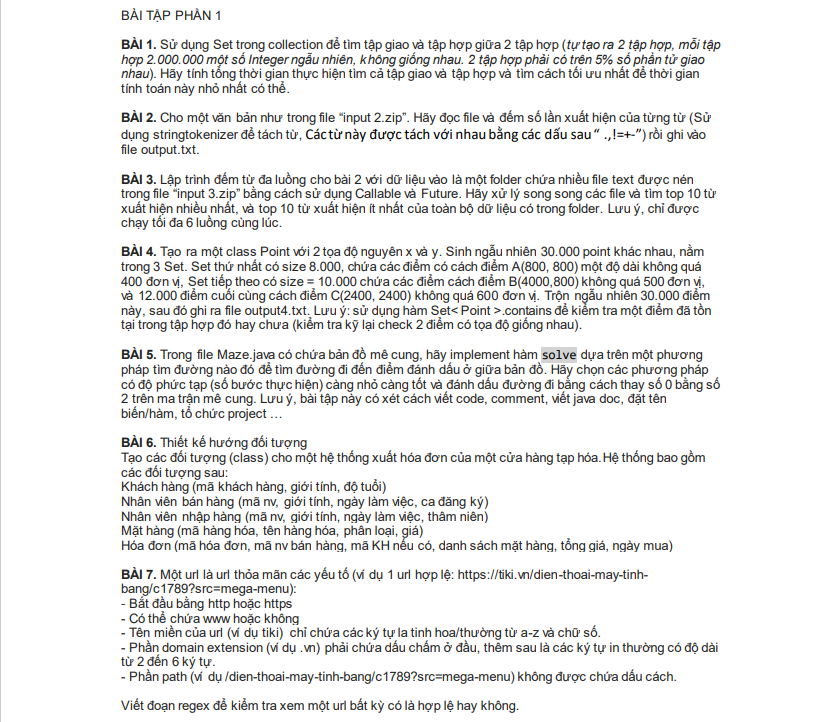
# **CHƯƠNG III. NỘI DUNG NGHIÊN CỨU**

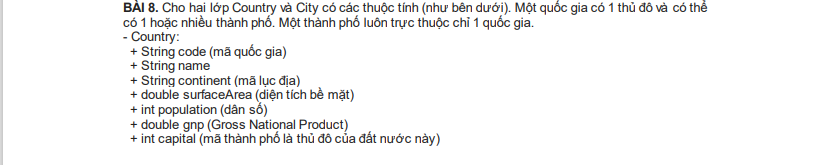
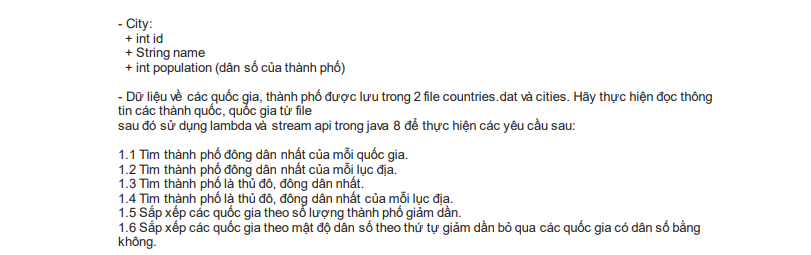
## **NỘI DUNG THỰC TẬP**

* Tìm hiểu chuyên sâu về ngôn ngữ lập trình Java.
* Cách sử dụng các framework hoặc các chương trình khác cùng Java.

## **QUÁ TRÌNH THỰC HIỆN**

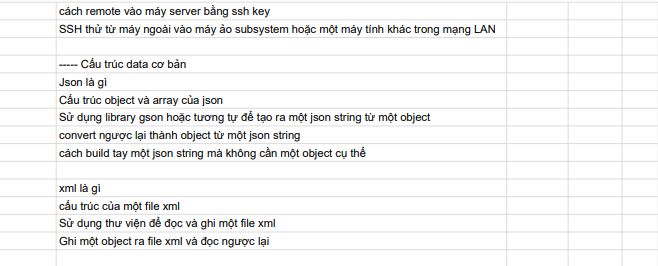
* 1. **Tuần 1: Nghiên cứu tài liệu, tự học và làm bài tập về Java cơ bản.**

****

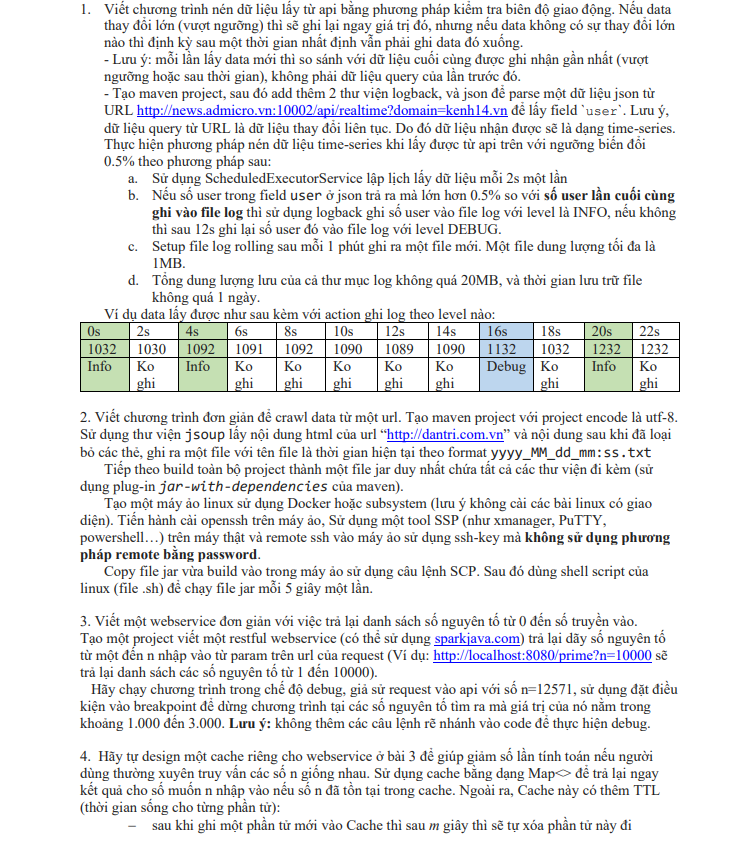
**** ****

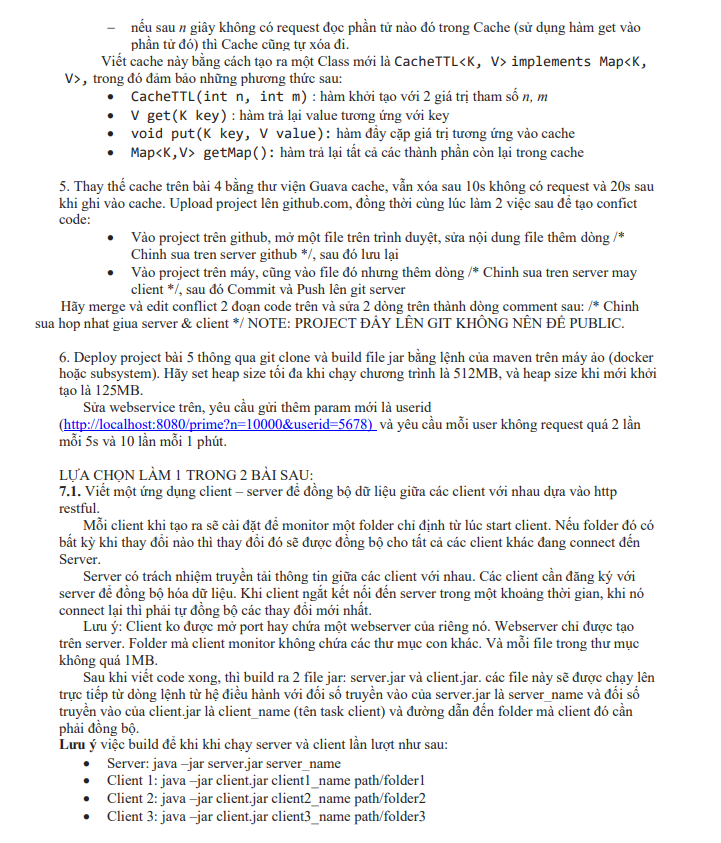
* 1. **Tuần 2: Nghiên cứu tài liệu về Java nâng cao.**

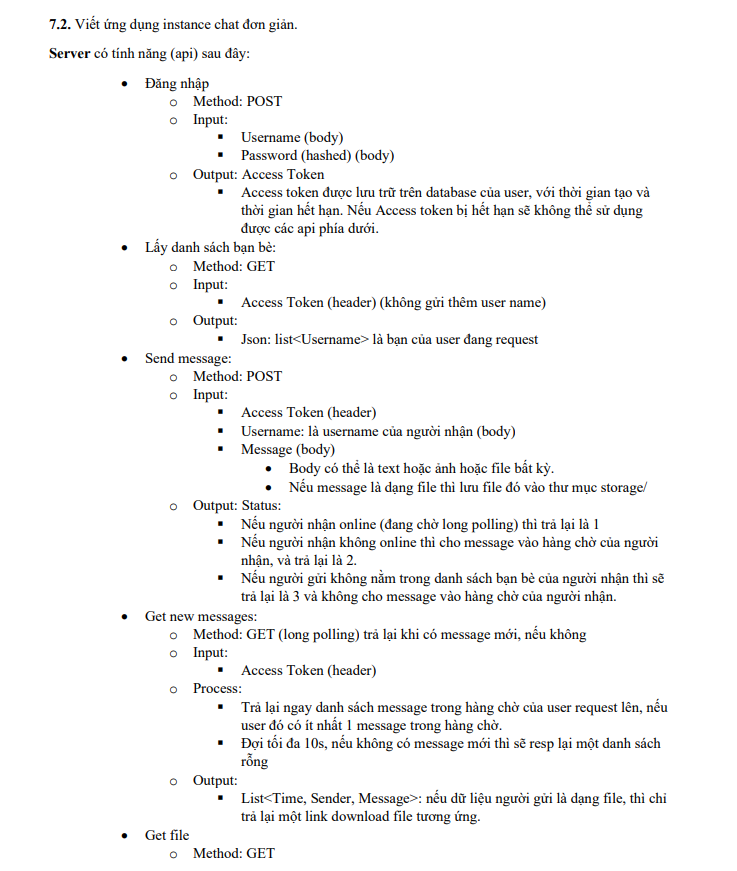
****

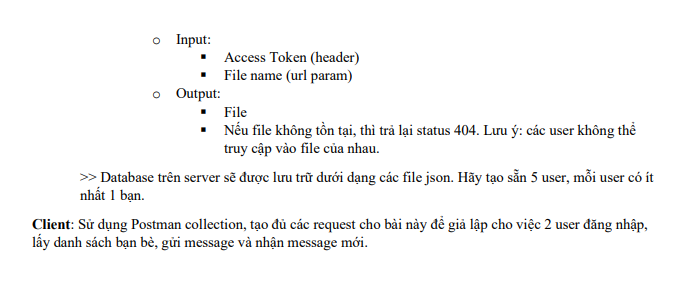
**** ****

****

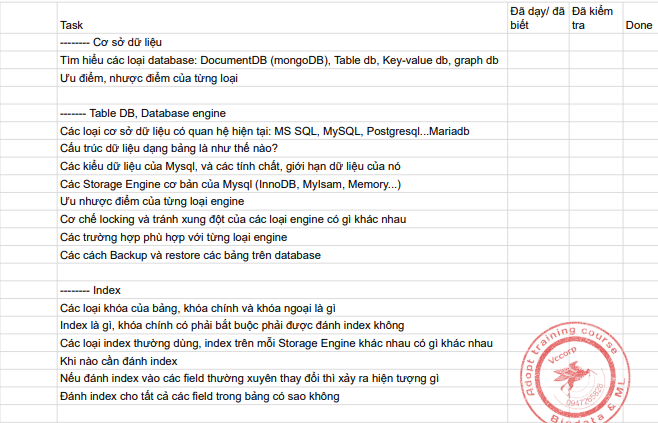
****

****

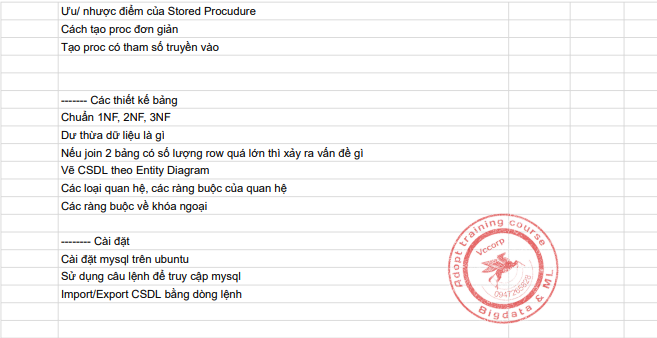
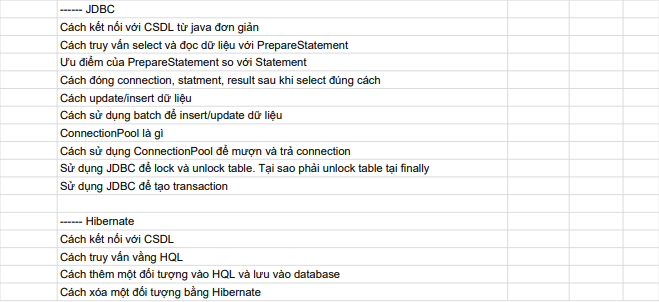
****

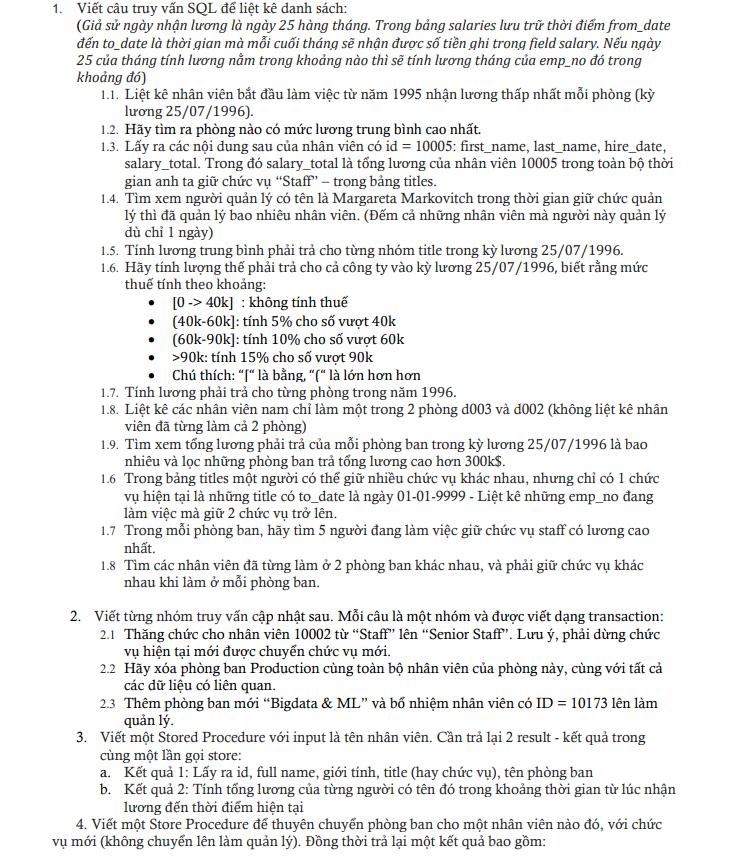
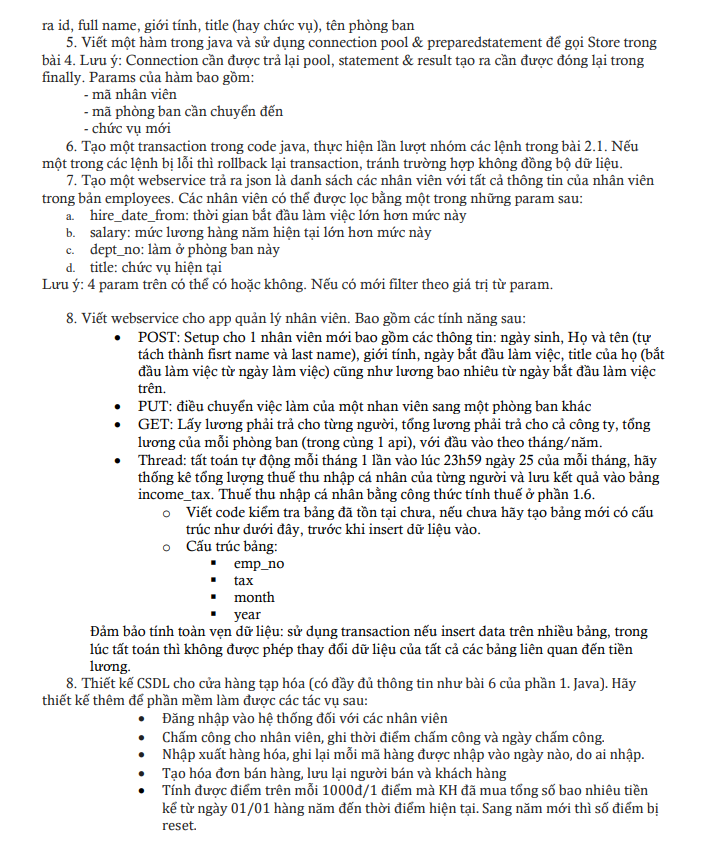
****

* 1. **Tuần 3: SQl, liên kết CSDL**

****

****

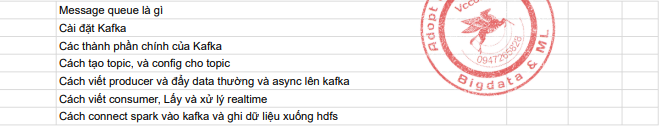
**** ****

**** ****

* 1. **Tuần 4: Tìm hiểu lập trình phân tán, Hadoop:**

****

****

****

* 1. **Tuần 5: Làm bài tập lập trình phân tán, Hadoop:**

****

# **CHƯƠNG IV. KẾT LUẬN VÀ KIẾN NGHỊ**

## **KẾT LUẬN**

Sau kì thực tập vừa rồi em đã được trải qua 1 quá trình rèn luyện khả năng tư duy và giải quyết vấn đề, bên cạnh đó là việc thích nghi và học hỏi những thứ mới như:

* Học được kĩ năng làm việc nhóm, ý thức và kỉ luật trong môi trường làm việc.
* Tích lũy thêm được rất nhiều kiến thức về ngôn ngữ lập trình Java và các công nghệ liên quan.
* Nâng cao kĩ năng code cũng như debug.
* Giải quyết xung đột khi pull hay push source code từ repo git về hay merge branch.

## **KIẾN NGHỊ**

Kính mong nhà trường có thể bổ sung thêm các kiến thức sát với thực tế hơn

như sử dụng Git, công nghệ đang phát triển mạnh, các mô hình làm việc doanh

nghiệp thực tế, ...

Ngoài ra, nhà trường có thể tạo điều kiện cho sinh viên thực tập sớm hơn để có

thể tiếp xúc với doanh nghiệp sớm hơn, tạo tiền đề cho việc định hướng của sinh

viên càng sớm càng tốt, và có thể bổ sung kiến thức hỗ trợ cho quá trình học

tập, làm đồ án, ... thay vì về cuối mới có thể thực tập hay gần ra trường vẫn

chưa rõ định hướng bản thân.