

TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG
BỘ MÔN MẠNG MÁY TÍNH & TRUYỀN THÔNG



LUẬN VĂN TỐT NGHIỆP

XÂY DỰNG TRÒ CHƠI CỜ TƯỚNG ONLINE TRÊN HỆ ĐIỀU HÀNH ANDROID

Sinh viên thực hiện:

Họ và tên: Trần Chí Tâm

MSSV : 1091443

Họ và tên : Phạm Thị Ánh Tuyết

MSSV : 1091643

Giáo viên hướng dẫn:

ThS. NGUYỄN CAO HỒNG NGỌC

MSCB : 002074



Học kỳ 1 2012-2013

TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG
BỘ MÔN MẠNG MÁY TÍNH & TRUYỀN THÔNG



LUẬN VĂN TỐT NGHIỆP

**XÂY DỰNG TRÒ CHƠI CỜ TƯỚNG
ONLINE TRÊN HỆ ĐIỀU HÀNH
ANDROID**

Sinh viên thực hiện:

Họ và tên: Trần Chí Tâm

MSSV : 1091443

Họ và tên : Phạm Thị Ánh Tuyết

MSSV : 1091643

Giáo viên hướng dẫn:

ThS. NGUYỄN CAO HỒNG NGỌC

MSCB : 002074

Cán bộ phản biện:

ThS. ĐOÀN HÒA MINH

ThS. LÂM CHÍ NGUYỄN

Luận văn được bảo vệ tại: Hội đồng chấm luận văn tốt nghiệp, bộ môn Mạng máy tính & truyền thông, khoa Công nghệ thông tin & truyền thông – Trường Đại học Cần Thơ vào ngày 05 tháng 12 năm 2012.

Mã số đề tài:

Có thể tìm hiểu luận văn tại:

- Thư viện Khoa công nghệ thông tin & truyền thông, trường Đại học Cần Thơ.
- Website: <http://www.cit.ctu.edu.vn/>



Học kỳ 1 2012-2013

LỜI CẢM ƠN



Trong suốt thời gian thực hiện đề tài luận văn tốt nghiệp Đại học “**Xây dựng trò chơi cờ tướng online trên hệ điều hành Android**”, chúng em luôn nhận được sự giúp đỡ quý báu từ quý thầy cô, anh chị, bạn bè và sự động viên, ủng hộ của người thân và gia đình để chúng em có thể hoàn thành đề tài của mình.

Trước tiên, em xin gửi lời chân thành cảm ơn đến cô Nguyễn Cao Hồng Ngọc – Giáo viên trực tiếp hướng dẫn chúng em thực hiện đề tài. Trong thời gian qua, cô đã sát cánh và đồng hành cùng chúng em, tận tình hướng dẫn, chỉ bảo và định hướng, tạo điều kiện tốt nhất để chúng em có thể hoàn thành đề tài.

Em cũng xin gửi lời cảm ơn đến quý thầy cô bộ môn Mạng máy tính & truyền thông – khoa CNTT & TT, đặc biệt là thầy Lâm Chí Nguyên, cô Triệu Thanh Ngoan, thầy Ngô Bá Hùng, thầy Nguyễn Trọng Nghĩa, cô Trần Thị Tố Quyên đã hỗ trợ, giải đáp thắc mắc, đưa ra ý kiến và định hướng cho chúng em thực hiện đề tài.

Xin gửi lời cảm ơn đến tất cả các quý thầy cô Trường Đại học Cần Thơ nói chung và quý thầy cô khoa CNTT & TT nói riêng, quý thầy cô đã truyền những kiến thức, kinh nghiệm quý báu cho chúng em trong suốt 4 năm qua để chúng em có đủ hành trang và nghị lực bước vào đời và có thể trở thành người có ích cho xã hội.

Xin cảm ơn các anh chị, đặc biệt là anh Lê Thành Tân, Đào Duy Tân – sinh viên khóa 34 khoa CNTT & TT và bạn bè đã nhiệt tình ủng hộ, truyền đạt kiến thức, kinh nghiệm của mình để chúng em có thể hoàn thành luận văn.

Một lần nữa xin gửi lời cảm ơn chân thành sâu sắc đến mọi người đã giúp đỡ, hỗ trợ, ủng hộ và động viên chúng em trong suốt thời gian làm đề tài.

Cần Thơ, ngày 19 tháng 10 năm 2012

Nhóm sinh viên thực hiện

Trần Chí Tâm

Phạm Thị Ánh Tuyết

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

[illegible]

Cần Thơ, ngày ... tháng ... năm 2012

Giảng viên hướng dẫn

ThS. Nguyễn Cao Hồng Ngọc

NHẬN XÉT CỦA HỘI ĐỒNG PHẢN BIỆN

This image shows a full page of white paper with horizontal dotted lines, typical of primary school writing paper. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Cần Thơ, ngày ... tháng ... năm 2012

Hội đồng phản biện

TÓM TẮT LUẬN VĂN



Ngày nay, sự phát triển vượt bậc của công nghệ thông tin đã mang đến rất nhiều lợi ích cho con người. Áp dụng công nghệ thông tin không những chỉ mang lại lợi ích về việc giảm chi phí mà còn tiết kiệm đáng kể thời gian, công sức... Nếu như sự ra đời của máy tính là bước ngoặt vĩ đại của nhân loại thì sự ra đời của thiết bị di động sẽ mở ra kỉ nguyên mới trong lĩnh vực công nghệ.

Nhu cầu giải trí của con người ngày càng tăng lên, đặc biệt là giải trí với game - xu hướng phát triển được dự đoán trong tương lai đã đưa chúng em đến với đề tài “**Xây dựng trò chơi cờ tướng online trên hệ điều hành Android**”. Với ý tưởng tự đề xuất, chúng em mong muốn tìm hiểu kiến thức mới, khả năng tư duy, kĩ năng lập trình và chuẩn bị hành trang sau khi tốt nghiệp. Thêm vào đó, đề tài có ý nghĩa thực tiễn, áp dụng vào một số giải đấu cờ nhỏ, hay phát triển thành phần mềm thương mại.

Từ việc tìm hiểu yêu cầu, phân tích, thiết kế, vận dụng lý thuyết vào cài đặt chương trình, kiểm thử và triển khai ứng dụng, đề tài đã áp dụng kiến thức về lập trình trên hệ điều hành Android kết hợp với lập trình mạng Socket TCP với giao thức tự thiết kế, lập trình song song và một số kiến thức khác để giải quyết vấn đề đặt ra.

Hệ thống gồm hai phần chính: Client và Server được xây dựng trên ngôn ngữ Java kết hợp với bộ xử lý song song MPJ tạo nên một chương trình hoàn chỉnh và mang tính ứng dụng cao. Ứng dụng game được thiết kế trên hệ điều hành Android từ 2.3 trở lên và được sử dụng khi đã kết nối đến Server. Vì thế người chơi có thể ở bất kì nơi đâu khi có kết nối mạng, nó sẽ rất tiện dụng khi tạo ra một ván cờ với sự thiết kế đồ họa đẹp mắt kết hợp với hiệu ứng âm thanh sinh động.

Sau 4 tháng thực hiện, đề tài hoàn thành và đạt được các yêu cầu cơ bản đã đặt ra. Tuy nhiên, không thể tránh khỏi những thiếu sót, chúng em rất mong nhận được ý kiến đóng góp từ quý thầy cô, anh chị, bạn bè và phản hồi của quý độc giả để có thể hoàn thiện chương trình hơn.

ABSTRACT



Nowadays, the strong development of information technology has been bringing people of many advantages. Applying information technology not only costs less but also saves time and labor significantly. If the birth of computers is a great turning-point, the birth of portable devices will open a new stage in technology fields.

With the development of society, the demand of recreation more and more increases, especially the game - the developing trend was predicted in the future, which makes us approach this thesis: “**Building online chineses chess game on Android operating system**”. With the idea which was offered by ourselves, we hope to search new knowledge, thinking ability, programming skill and prepare for baggage after we graduate. Moreover, this thesis has practical meaning in recreation, it can be applied for some small competitions, or develop into commercial software.

From learning requirements, analysis, design, applying theory for implementing program, testing and deploying application, in this thesis, we applied Android programming knowledge, socket network programming, combined multithreading programming, MPI parallel programming in order to solve problems.

Our system has two main parts: Client and Server were built by using Java language combine with parallel processor MPJ, it created a complete and practical program. Game application was designed on Android operating system from 2.3 and upper version and was used when it connected to Server. Therefore, players can play everywhere if it has network, it will be convenient when creating board with beautiful design and lively sound effect.

After nearly 4 months, the thesis was completed and obtained basic requirements. However, during doing thesis, it's impossible to avoid mistakes. We hope that we will be received opinions from teachers, relatives, friends and feedbacks from readers so that we can improve our program better.

DANH MỤC CÁC THUẬT NGỮ VIẾT TẮT

♦♦♦♦♦♦♦♦

Từ viết tắt	Tiếng Anh	Tiếng Việt
3G	3rd Generation	Công nghệ truyền thông thế hệ thứ 3
A2DP	Advanced Audio Distribution Profile	Công nghệ cho phép truyền âm thanh giữa các thiết bị qua Bluetooth
API	Application Programming Interface	Giao diện lập trình ứng dụng
APK	Application Package File	Gói tập tin ứng dụng
ARP	Address Resolution Protocol	Giao thức phân giải địa chỉ IP thành địa chỉ MAC
AVRCP	Audio/Video Remote Control Profile	Cách thức kết nối qua bluetooth
C++	C plus plus	Ngôn ngữ lập trình hướng đối tượng
CDMA	Code division multiple access	Đa truy nhập (đa người dùng) phân chia theo mã
CPU	Central Processing Unit	Đơn vị xử lý trung tâm
CSDL	Database	Cơ sở dữ liệu
DVM	Dalvik Virtual Machine	Máy ảo Dalvik
EVDO	Evolution-Data Optimized	Phát triển – Tối ưu hóa Dữ liệu
ICMP	Internet Control Message Protocol	Giao thức điều khiển thông điệp Internet
IGMP	Internet Group Management Protocol	Giao thức quản lý nhóm Internet
IRC	Internet Relay Chat	Mạng trò chuyện trực tuyến
ISO	International Standard Organization.	Tổ chức tiêu chuẩn hóa quốc tế
JVM	Java Virtual Machine	Máy ảo Java
MAC	Media Access Control	Điều khiển truy nhập môi trường

MPI	Message Passing Interface	Giao thức truyền thông điệp chương trình song song
MPJ	Java message passing	Giao thức truyền thông điệp trong ngôn ngữ Java
OS	Operating system	Hệ điều hành
OSI	Open System Interconnection Model	Mô hình tham chiếu hệ thống mở
ROM	Read-Only Memory	Bộ nhớ chỉ đọc
SD	Secure Digital	Thẻ nhớ
SDK	Software Development Kit	Bộ công cụ phát triển phần mềm
SMTP	Simple Mail Transfer Protocol	Giao thức truyền tải thư tín đơn giản
TCP	Transmission Control Protocol	Giao thức điều khiển truyền dẫn
TCP/IP	Transmission Control Protocol/Internet protocol	Bộ giao thức liên mạng
UDP	User Datagram Protocol	Giao thức dữ liệu người sử dụng
WVGA	Wide Video Graphics Array	Độ phân giải màn hình 480x800
XML	Extensible Markup Language	Ngôn ngữ Đánh dấu Mở rộng

XH□ NT 1. DANH MỤC CÁC BẢNG

♦♦♦♦♦♦♦

STT	Bảng	Tên Bảng	Trang
1	Bảng 2.1	So sánh 2 chế độ giao tiếp TCP và UDP	29
2	Bảng 2.2	Các quân cờ tướng	39
3	Bảng 3.1	Giá trị các quân cờ	48
4	Bảng 3.2	Bảng NGUOI_CHOI	60
5	Bảng 3.3	Bảng CAP_BAC	60
6	Bảng 3.4	Bảng VAN_CO	61
7	Bảng 3.5	Bảng LICH_SU_VAN_CO	61

DANH MỤC CÁC HÌNH ẢNH

♦♦♦♦♦♦♦♦

STT	Hình	Tên Hình	Trang
1	Hình 1.1	Các hệ điều hành trên điện thoại	1
2	Hình 2.1	Ứng dụng trên Android	9
3	Hình 2.2	Kiến trúc hệ điều hành Android	10
4	Hình 2.3	Quá trình chạy ứng dụng của Java VM và Dalvik VM	12
5	Hình 2.4	Vòng đời của Activity	15
6	Hình 2.5	Vòng đời của Service	16
7	Hình 2.6	Kho dữ liệu danh bạ	17
8	Hình 2.7	Sơ đồ hoạt động của Intent	17
9	Hình 2.8	Image View	19
10	Hình 2.9	Giao diện trên Android	20
11	Hình 2.10	OptionsMenu	20
12	Hình 2.11	ContextMenu	21
13	Hình 2.12	Dialog	21
14	Hình 2.13	Toast	22
15	Hình 2.14	Giao diện View	22
16	Hình 2.15	Mạng máy tính	23
17	Hình 2.16	Mô hình OSI	24
18	Hình 2.17	Mô hình OSI và giao thức TCP/IP	25
19	Hình 2.18	Mô hình nối kết Socket	27
20	Hình 2.19	Cơ chế giao tiếp giữa các máy tính thông qua Socket	28
21	Hình 2.20	Chế độ giao tiếp TCP và UDP	29
22	Hình 2.21	Thiết lập có nối kết TCP	30
23	Hình 2.22	Server xử lý song song	31
24	Hình 2.23	Các trạng thái của luồng	32
25	Hình 2.24	Quá trình đa luồng	33
26	Hình 2.25	Chương trình được thiết kế theo tập lệnh rời rạc	35
27	Hình 2.26	Chương trình được thiết kế song song	35
28	Hình 2.27	Mô hình tuần tự	36
29	Hình 2.28	Mô hình chuyển thông điệp	36
30	Hình 2.29	Hình ảnh chơi cờ	37
31	Hình 2.30	Bàn cờ tướng	38
32	Hình 2.31	Ma trận bàn cờ tướng	38
33	Hình 2.32	Các vị trí quân Tướng đi	40
34	Hình 2.33	Các vị trí quân Sĩ đi	40
35	Hình 2.34	Các vị trí quân Tượng đi	41
36	Hình 2.35	Các vị trí quân Tượng đi và trường hợp bị cản	41

37	Hình 2.36	Các vị trí quân Mã đi	42
38	Hình 2.37	Các vị trí quân Mã đi và trường hợp bị cản	42
39	Hình 2.38	Các vị trí quân Xe đi	43
40	Hình 2.39	Các vị trí quân Pháo đi	43
41	Hình 2.40	Các vị trí quân Pháo đi và ăn cờ đối thủ	43
42	Hình 2.41	Các vị trí quân Tốt đi	44
43	Hình 3.1	Kiến trúc chương trình	46
44	Hình 3.2	Sơ đồ nối kết Client và Server	47
45	Hình 3.3	Mô hình dữ liệu mức quan niệm CDM	59
46	Hình 3.4	Mô hình giao tiếp giữa Game Server và MPI node	62
47	Hình 3.5	Mô hình tổng quan	69
48	Hình 3.6	Sơ đồ giao tiếp Client-Server khi đăng ký	70
49	Hình 3.7	Sơ đồ giao tiếp Client-Server khi đăng nhập	71
50	Hình 3.8	Sơ đồ giao tiếp Client-Server khi yêu cầu chơi	73
51	Hình 3.9	Sơ đồ giao tiếp Client-Server khi đánh cờ	75
52	Hình 3.10	Sơ đồ giao tiếp Client-Server cầu hòa/xin thua	77
53	Hình 3.11	Sơ đồ chức năng của chương trình	79
54	Hình 3.12	Giao diện bắt đầu	79
55	Hình 3.13	Giao diện đăng nhập	80
56	Hình 3.14	Giao diện đăng ký	80
57	Hình 3.15	Giao diện danh sách người chơi	81
58	Hình 3.16	Giao diện bàn cờ	81
59	Hình 3.17	Giao diện Chat	82
60	Hình 3.18	Giao diện cài đặt	82
61	Hình 3.19	Giao diện luật chơi	83

MỤC LỤC



LỜI CẢM ƠN.....	i
NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN.....	ii
NHẬN XÉT CỦA HỘI ĐỒNG PHẢN BIỆN.....	iii
TÓM TẮT LUẬN VĂN.....	iv
ABSTRACT.....	v
DANH MỤC CÁC THUẬT NGỮ VIẾT TẮT.....	vi
CHƯƠNG 1. DANH MỤC CÁC BẢNG.....	viii
DANH MỤC CÁC HÌNH ẢNH.....	ix
MỤC LỤC.....	xi
TỔNG QUAN.....	1
1.1. ĐẶT VẤN ĐỀ.....	1
1.2. LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ.....	2
1.2.1. Hiện trạng game online.....	2
1.2.2. Các nghiên cứu và hướng giải quyết có liên quan.....	3
1.3. MỤC TIÊU - PHẠM VI ĐỀ TÀI.....	3
1.3.1. Mục tiêu.....	3
1.3.2. Phạm vi đề tài.....	4
1.4. PHƯƠNG PHÁP NGHIÊN CỨU GIẢI QUYẾT VẤN ĐỀ.....	4
1.4.1. Phương pháp nghiên cứu.....	4
1.4.2. Hướng giải quyết.....	5
1.4.3. Phân công công việc.....	5
1.5. BỐ CỤC ĐỀ TÀI.....	6
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT.....	7
2.1. GIỚI THIỆU VỀ LẬP TRÌNH TRÊN HỆ ĐIỀU HÀNH ANDROID.....	7
2.1.1. Tổng quan.....	7
2.1.2. Kiến trúc ứng dụng.....	10
2.1.3. Máy ảo Dalvik.....	11
2.1.4. Chu kì sống của ứng dụng.....	12

2.1.5. Các tiến trình.....	13
2.1.6. Công cụ lập trình.....	13
2.1.7. Các thành phần trên ứng dụng Android.....	14
2.1.8. Lập trình giao diện trên Android.....	18
2.2. TỔNG QUAN VỀ MẠNG MÁY TÍNH.....	23
2.2.1. Khái niệm.....	23
2.2.2. Các thành phần.....	23
2.2.3. Mô hình OSI.....	24
2.2.4. Giao thức TCP/IP.....	25
2.3. GIỚI THIỆU VỀ LẬP TRÌNH SOCKET.....	27
2.3.1. Khái niệm về socket.....	27
2.3.2. Cơ chế giao tiếp.....	27
2.3.3. Các chế độ giao tiếp	28
.....	29
2.3.1. So sánh giữa TCP và UDP.....	29
2.3.2. Xây dựng chương trình Socket ở chế độ có nối kết (TCP).....	30
2.4. LẬP TRÌNH LUỒNG (THREAD).....	31
2.5. TÍNH TOÁN LƯỚI VÀ LẬP TRÌNH SONG SONG.....	33
2.5.1. Hệ thống lưới và tính toán lưới.....	33
2.5.2. Lợi ích của tính toán lưới.....	34
2.5.3. Lập trình song song.....	34
2.5.4. Giới thiệu giao diện lập trình MPI.....	36
2.5.5. Gói lập trình MPJ Express.....	37
2.6. GIỚI THIỆU VỀ CỜ TƯỚNG.....	37
2.6.1. Nguồn gốc.....	37
2.6.2. Bàn cờ.....	38
2.6.3. Luật chơi.....	40
CHƯƠNG 3. NỘI DUNG – KẾT QUẢ NGHIÊN CỨU.....	46
3.1. XÂY DỰNG TRÒ CHƠI CỜ TƯỚNG ONLINE.....	46
3.1.1. Đặc tả chi tiết đề tài.....	46
3.1.2. Kiến trúc chương trình.....	47

3.1.3. Server	48
3.1.4. Client.....	48
3.1.5. Xây dựng bàn cờ.....	49
3.1.6. Giải thuật.....	51
3.2. XÂY DỰNG SERVER.....	58
3.2.1. Chức năng.....	58
3.2.2. Các thành phần của Server.....	58
3.2.3. Hướng giải quyết.....	59
3.3. XÂY DỰNG CLIENT.....	65
3.3.1. Chức năng.....	65
3.3.2. Vấn đề đặt ra.....	65
3.3.3. Hướng giải quyết.....	65
3.3.4. Các thành phần của chương trình.....	66
3.3.5. Thiết kế Service.....	68
3.3.6. Thiết kế giao diện và chức năng.....	69
3.4. CẤU TRÚC THÔNG ĐIỆP GIỮA CLIENT VÀ SERVER.....	70
3.4.1. Mô hình tổng quan.....	70
3.4.2. Đăng ký.....	71
3.4.3. Đăng nhập.....	72
3.4.4. Yêu cầu chơi.....	74
3.4.5. Chat và đánh cờ	76
3.4.6. Cầu hòa, xin thua.....	78
3.5. CHƯƠNG TRÌNH DEMO.....	80
3.5.1. Sơ đồ chức năng.....	80
3.5.2. Giao diện chương trình.....	80
3.5.3. Giao diện đăng nhập.....	81
3.5.4. Giao diện đăng ký.....	81
3.5.5. Giao diện người chơi online.....	82
3.5.6. Giao diện chơi cờ.....	82
3.5.7. Giao diện chat.....	83
3.5.8. Giao diện cài đặt.....	83

3.5.9. Giao diện trợ giúp.....	84
CHƯƠNG 4. KẾT LUẬN – HƯỚNG PHÁT TRIỂN.....	85
4.1. KẾT QUẢ ĐẠT ĐƯỢC.....	85
4.1.1. Kết quả đạt được.....	85
4.1.2. Ưu điểm.....	85
4.1.3. Hạn chế.....	86
4.2. HƯỚNG PHÁT TRIỂN.....	86
TÀI LIỆU THAM KHẢO.....	87
PHỤ LỤC A: CÀI ĐẶT PHẦN MỀM ĐỂ LẬP TRÌNH ANDROID.....	88
PHỤ LỤC B: CÀI ĐẶT MPJ EXPRESS.....	95

TỔNG QUAN

1.1. ĐẶT VẤN ĐỀ

Ngày nay, khi nhắc đến “Công nghệ thông tin” thì mọi người đã không còn gì xa lạ với nó nữa. Bởi lẽ, công nghệ thông tin đã được áp dụng trên hầu hết các lĩnh vực như kinh tế, kỹ thuật, giáo dục, văn hóa, thể thao và giải trí. Việc đẩy mạnh công nghệ thông tin là xu thế và là hướng phát triển trong tương lai, giúp thực hiện công việc một cách không chỉ thuận tiện mà còn nhanh chóng hơn rất nhiều.

Cùng với sự phát triển của của đất nước trong thời kì hội nhập, chất lượng cuộc sống con người ngày càng được tăng lên về nhu cầu vật chất lẫn tinh thần nhất là nhu cầu giải trí. Nắm bắt được nhu cầu đó, hàng loạt dịch vụ ra đời trong đó trong đó game online là hình thức phổ biến và được nhiều người ưa chuộng.

Bên cạnh đó, nếu như sự ra đời của máy tính là một thành tựu vĩ đại của nhân loại, thì sự ra đời của điện thoại giống như chìa khóa mở ra một thế giới mới trong lĩnh vực truyền thông đặc biệt là các thiết bị Smartphone tân tiến với nhiều loại hệ điều hành khác nhau. Người ta không còn phải sử dụng máy tính cồng kềnh nữa mà các chức năng của một máy tính hầu như đã được tích hợp trên điện thoại nhỏ gọn.



Hình 1.1. Các hệ điều hành trên điện thoại

(Nguồn <http://vnexpress.net>)

Thấy được nhu cầu thiết yếu trong việc giải trí, đặc biệt là game cùng với sự phát triển không ngừng của điện thoại di động, lập trình trên thiết bị di động – xu hướng được dự đoán sẽ phát triển trong tương lai, thêm vào đó là thông qua tìm hiểu và nghiên cứu, tác giả quyết định chọn **“Xây dựng trò chơi cờ tướng online trên hệ điều hành Android”** làm đề tài luận văn tốt nghiệp của mình.

Với ý tưởng tự đề xuất, tác giả sẽ nghiên cứu và xây dựng trò chơi cờ tướng online trên điện thoại di động chạy hệ điều hành Android với mong muốn tìm hiểu kiến thức mới kết hợp và vận dụng kiến thức đã học để giải quyết vấn đề thực tế. Bên cạnh đó, tác giả mong muốn sản phẩm của mình sẽ có ứng dụng thực tế. Tuy nhiên, trong quá trình nghiên cứu và giải quyết vấn đề, không thể tránh những điều sai sót, rất mong nhận được ý kiến đóng góp quý báu của thầy cô, bạn bè và sự phản hồi từ phía bạn đọc.

1.2. LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ

1.2.1. Hiện trạng game online

Trên thế giới cũng như ở Việt Nam, game online đã phát triển rất mạnh. Nhiều hãng game lớn luôn tung ra chiến lược quảng bá hình ảnh của mình bởi những game lureng lầy. Ở Việt Nam cũng vậy, VNG - FPT Online – VTC Game và AsiaSoft luôn là những nhà cung cấp game online hàng đầu.

Hiện nay trên thế giới đã có nhiều chương trình game online cho điện thoại và trang web chơi cờ phổ biến hiện nay là <http://www.chess.com>, trang web này đã hỗ trợ cho người dùng chơi cờ vua trực tuyến trên máy tính và điện thoại. Một trang web khác rất nổi tiếng là <http://xiangqiclub.com> hỗ trợ việc chơi cờ tướng online và thu hút rất nhiều người chơi từ khắp nơi trên thế giới. Ở Việt Nam, nhiều web game được xây dựng để phục vụ nhu cầu người chơi, đặc biệt là trang web <http://play.zing.vn> có nhiều loại trò chơi như cờ tướng, cờ vua, đánh bài ...vv. Tuy nhiên, <http://www.chess.com> chỉ hỗ trợ chơi cờ vua và chưa có chương trình chơi cờ tướng, còn <http://xiangqiclub.com> thì chỉ hỗ trợ chơi cờ tướng trên máy tính. Bên cạnh đó, <http://play.zing.vn> có đầy đủ các loại cờ và trò chơi dân gian khác và cũng chỉ hỗ trợ cho máy tính và cũng chưa có phiên bản giành cho điện thoại.

Android ra đời và phát triển mạnh mẽ kéo theo đó là hàng loạt các ứng dụng hỗ trợ. Android Market là nơi cho phép tải rất nhiều ứng dụng phong phú, game chạy trên hệ điều hành Android. Hiện nay, Android Market cũng có rất nhiều phần mềm hỗ trợ chơi cờ tướng, nhưng hầu hết là người chơi với máy và vẫn chưa tìm thấy phần mềm hỗ trợ chơi cờ tướng online.

1.2.2. Các nghiên cứu và hướng giải quyết có liên quan

Lập trình trên thiết bị di động và lập trình game luôn là những đề tài có tính hấp dẫn và thu hút sinh viên rất cao. Ở khoa CNTT & TT – Đại học Cần Thơ, trước đây cũng đã có nhiều sinh viên thực hiện đề tài luận văn lập trình game trên điện thoại di động. Một số nghiên cứu có thể điểm qua ở đây là:

- Luận văn tốt nghiệp Đại học **“Xây dựng trò chơi caro trên hệ điều hành trên Android”** của sinh viên Huỳnh Hải Đăng ngành Kỹ thuật phần mềm khóa 34, học kỳ 2 năm học 2011 - 2012. Đề tài đã được đánh giá là tốt, hoàn thiện hầu hết các chức năng, bao gồm chơi cờ với máy, chơi giữa hai người thông qua bluetooth, chơi qua wifi nhưng trò chơi là caro, chưa thể hiện được hết tư duy của người đánh cờ và sự phong phú của nước đi trong bàn cờ. Về trò chơi cờ tướng thì trước nay vẫn chưa có sinh viên nào thực hiện, nên tác giả lựa chọn cờ tướng sẽ là trò chơi trong đề tài luận văn của mình.

- Luận văn tốt nghiệp Đại học **“Xây dựng trò chơi đối kháng dựa trên giao thức IRC”** của nhóm sinh viên do thầy Ngô Bá Hùng hướng dẫn vào học kỳ 2 năm học 2010 - 2011. Đề tài xây dựng trò chơi đối kháng dựa trên giao thức IRC đã hoàn thành tốt nhưng còn một số hạn chế là thời gian chờ kết nối đến Server còn lâu.

1.3. MỤC TIÊU - PHẠM VI ĐỀ TÀI

1.3.1. Mục tiêu

Đề tài **“Xây dựng trò chơi cờ tướng online trên hệ điều hành Android”** nhằm xây dựng một chương trình game online và có thể ứng dụng trong thực tế. Ngoài ra, sản phẩm của đề tài phải là sản phẩm trò chơi cờ tướng online hoàn chỉnh giúp người chơi có thể giải trí, áp dụng vào việc thi đấu trong một số giải đấu nhỏ và xa hơn sẽ phát triển thành phần mềm thương mại.

1.3.2. Phạm vi đề tài

Đề tài sẽ xây dựng chương trình game cờ tướng online hoàn chỉnh. Phạm vi đề tài chủ yếu xoay quanh những nội dung sau:

- Xây dựng Client, giao diện chương trình ở phía người chơi sẽ được xử lý tương thích cho nhiều loại điện thoại có độ phân giải màn hình khác nhau. Cụ thể ở trong đề tài này sẽ được demo trên điện thoại SamSung Galaxy Y 5360 (hệ điều hành Android 2.3.6).

- Người chơi sẽ sử dụng điện thoại chạy trên hệ điều hành Android (từ 2.2 trở lên) cài chương trình được viết cho điện thoại. Để sử dụng các tính năng của chương trình người chơi phải đăng ký tài khoản. Sau khi có tài khoản, người chơi sẽ đăng nhập vào hệ thống để chọn bạn chơi, ván cờ sẽ được tạo ra khi cả hai bên chấp nhận.

- Xây dựng Server để xử lý thông tin được nhận từ chương trình được cài đặt trên điện thoại của người chơi. Thông tin được xử lý bao gồm người chơi nào, thông tin nước đi của từng người chơi, kiểm tra nước đi hợp lệ, giải thuật ăn quân cờ của đối phương, kiểm tra thắng thua, lưu thông tin điểm.

- Áp dụng kĩ thuật lập trình song song MPI vào việc xây dựng giải thuật kiểm tra chiếu tướng và tính toán thắng thua của bàn cờ.

1.4. PHƯƠNG PHÁP NGHIÊN CỨU GIẢI QUYẾT VẤN ĐỀ

1.4.1. Phương pháp nghiên cứu

Phương pháp nghiên cứu là một trong những yếu tố rất quan trọng quyết định sự thành công của đề tài. Trong thời gian 15 tuần, tác giả đã nghiên cứu và vận dụng kiến thức để xây dựng đề tài. Phương pháp nghiên cứu trong đề tài này như sau:

- Tìm hiểu yêu cầu cụ thể của đề tài.
- Phân tích yêu cầu của đề tài.
- Xác định rõ phạm vi của đề tài.
- Đánh giá hệ thống và lập kế hoạch công việc.
- Tìm hiểu lý thuyết liên quan đến đề tài.
- Vận dụng lý thuyết vào để xây dựng hệ thống cụ thể.
- Kiểm thử chương trình, sửa lỗi và hoàn thiện các chức năng.
- Báo cáo kết quả đạt được.

1.4.2. Hướng giải quyết

Từ việc tìm hiểu và làm rõ nội dung đề tài, tác giả đưa ra hướng giải quyết như sau:

- Áp dụng kỹ thuật lập trình ứng dụng trên hệ điều hành Android để xây dựng chương trình Client.
- Áp dụng kỹ thuật lập trình mạng bằng giao thức Socket TCP kết hợp với lập trình đa luồng để xây dựng Server. Server này đóng vai trò là nơi trung gian để giao tiếp giữa các Client và cũng là nơi tính toán các giải thuật.
- Áp dụng kỹ thuật lập trình song song bằng việc sử dụng thư viện lập trình song song MPJ Express để xây dựng giải thuật kiểm tra chiếu tướng và tính toán thắng thua cho bàn cờ.
- Sử dụng các công cụ phát triển ứng dụng cho Client và Server: Eclipse, Netbeans.

1.4.3. Phân công công việc

STT	Công việc	Người thực hiện	Thời gian	Đánh giá
1	Tìm hiểu lý thuyết	Trần Chí Tâm Phạm Thị Ánh Tuyết	Tuần 1 – 2	Tốt
2	Phân tích, định hướng công việc	Trần Chí Tâm Phạm Thị Ánh Tuyết	Tuần 3 – 5	Tốt
3	Thiết kế giao diện cho Client	Phạm Thị Ánh Tuyết	Tuần 4 – 6	Tốt
4	Viết chương trình Client	Phạm Thị Ánh Tuyết Trần Chí Tâm	Tuần 4 – 7	Tốt
5	Viết chương trình Server	Trần Chí Tâm Phạm Thị Ánh Tuyết	Tuần 4 – 9	Tốt
6	Cài đặt mpi xử lý song song	Trần Chí Tâm	Tuần 9 – 12	Tốt
7	Tổng hợp code	Phạm Thị Ánh Tuyết	Tuần 12	Tốt
8	Kiểm thử, sửa lỗi	Trần Chí Tâm	Tuần 12	Tốt
9	Tổng hợp tài liệu, viết báo cáo	Trần Chí Tâm Phạm Thị Ánh Tuyết	Tuần 14 - 15	Tốt

1.5. BỐ CỤC ĐỀ TÀI

Trong suốt thời gian làm đề tài, tác giả đã tổng hợp những kiến thức có liên quan đến đề tài. Để tạo sự thuận tiện cho người đọc, bài báo cáo này sẽ có bố cục phân chia thành 4 chương:

- Chương 1: Tổng quan - Đưa ra vấn đề, giới thiệu về sự cần thiết của đề tài. Bên cạnh đó, chương này sẽ đề cập đến lịch sử giải quyết vấn đề của đề tài để người đọc có thể nắm rõ. Tiếp theo đó là phân mục tiêu, phương pháp nghiên cứu cũng như hướng giải quyết sẽ được đề cập đến cho quá trình phát triển đề tài.
- Chương 2: Cơ sở lý thuyết - Từ việc nắm rõ một số vấn đề cơ bản ở chương 1, trong chương 2, tác giả sẽ đề cập chi tiết về lý thuyết có liên quan đến đề tài. Cụ thể, phần đầu giới thiệu về lập trình trên thiết bị di động trên hệ điều hành Android. Phần tiếp theo là lý thuyết về mạng máy tính, các giao thức của mạng máy tính, các phương thức giao tiếp giữa hai máy tính. Đặc biệt, tác giả sẽ chú trọng vào giao thức TCP/IP và cơ chế giao tiếp thông qua Socket, cũng như kiến thức lập trình song song MPI với việc sử dụng gói phần mềm hỗ trợ MPJ Express. Ở phần cuối sẽ giới thiệu đến độc giả trò chơi cờ tướng, quy tắc cũng như luật chơi của loại cờ phổ biến này.
- Chương 3: Nội dung và kết quả nghiên cứu - Từ việc nghiên cứu, phân tích đề tài ở chương 1 kết hợp với lý thuyết ở chương 2, chương 3 sẽ vận dụng và kết hợp vào xây dựng trò chơi cờ tướng online hoàn chỉnh. Trong chương này, các giai đoạn thực hiện, chi tiết kỹ thuật để xây dựng chương trình sẽ được trình bày cụ thể bao gồm mô hình kiến trúc hệ thống, cách thức xây dựng chương trình Client và Server, các thông điệp giao tiếp giữa Client và Server. Bên cạnh đó, chương trình demo sẽ được giới thiệu ở cuối chương.
- Chương 4: Tổng kết - Đánh giá – Chương này sẽ tổng kết lại những ưu điểm, kết quả đạt được, những mặt hạn chế còn tồn tại cũng như đề ra hướng phát triển trong thời gian sắp tới để chương trình có thể ngày càng được hoàn thiện hơn.

XH□□NT 2. CƠ SỞ LÝ THUYẾT

Trong chương 2, tác giả sẽ giới thiệu chi tiết về lý thuyết có liên quan đến đề tài. Nội dung bao gồm giới thiệu về lập trình trên hệ điều hành Android, mạng máy tính, đặc biệt là giao thức TCP/IP, kỹ thuật lập trình mạng, lập trình Socket kết hợp với lập trình đa luồng và lập trình song song MPI. Ở phần cuối chương, tác giả sẽ đề cập đến trò chơi cờ tướng bao gồm các quy tắc luật chơi của loại cờ phổ biến này.

2.1. GIỚI THIỆU VỀ LẬP TRÌNH TRÊN HỆ ĐIỀU HÀNH ANDROID

2.1.1. Tổng quan

a. Khái niệm

Android là một hệ điều hành trên điện thoại, nhưng Android còn được cho là một nền tảng mở cho thiết bị di động (gồm hệ điều hành, middleware và một số ứng dụng cơ bản) được phát triển bởi Google và dựa trên nền tảng Linux.

Android là một sản phẩm kết tinh từ ý tưởng của Khối Liên minh thiết bị cầm tay mở do Google dẫn đầu, với thành viên là các công ty hàng đầu về công nghệ và di động toàn cầu như Qualcomm, Intel, Motorola, Texas Instruments và LG Electronics, các nhà mạng như T-Mobile, Sprint Nextel, NTT DoCoMo và China Mobile. Google công bố hầu hết các mã nguồn của Android theo bản cấp phép Apache. Hệ điều hành Android bao gồm 12 triệu dòng mã; 3 triệu dòng XML, 2.8 triệu dòng mã C, 2.1 triệu mã Java và 1.75 triệu dòng mã C++.

Dựa trên ngôn ngữ Java và có thể sử dụng miễn phí bộ Android Software Development Kit (SDK), Android có cả một cộng đồng những nhà phát triển rất lớn viết các ứng dụng cho hệ điều hành của mình.

Về cơ bản Android sẽ hỗ trợ các chức năng sau:

- Màn hình cảm ứng (đa điểm).
- 3G, wifi.
- Trình duyệt dựa trên webkit.
- Tin nhắn (SMS) theo luồng.
- Định dạng MPEG-4, H.264, MP3, AAC.
- Bộ tăng tốc đồ họa 3D.

b. Lịch sử phát triển

Tháng 7 năm 2005, Google mua lại Android, Inc., một công ty nhỏ mới thành lập có trụ sở ở Palo Alto, California, Mỹ.

Tháng 9 năm 2007, InformationWeek đăng tải một nghiên cứu của Evalueserve cho biết Google đã nộp một số đơn xin cấp bằng sáng chế trong lĩnh vực điện thoại di động.

Tháng 9 năm 2008, chiếc điện thoại sử dụng Android đầu tiên chính thức được ra mắt. Đó là T-Mobile G1.

Từ tháng 10 năm 2008, hệ điều hành Android đã chính thức trở thành phần mềm mã nguồn mở. Theo đó, các công ty thứ ba được phép thêm những ứng dụng của riêng họ vào Android và bán chúng mà không cần phải hỏi ý kiến Google.

Từ lúc ra mắt phiên bản đầu tiên cho tới nay, Android đã có rất nhiều bản nâng cấp. Đa số đều tập trung vào việc vá lỗi và thêm những tính năng mới. Hiện tại các phiên bản chính của Android bao gồm:

- 1.0: Phiên bản đầu tiên, có mặt trên HTC Dream (T-Mobile G1).
- 1.1: Vá lỗi bảo mật, nhằm chế sự can thiệp vào cấu trúc ROM của máy.
- 1.5 (Cupcake): Hỗ trợ bluetooth A2DP và AVRCP, cập nhật về giao diện người dùng.
- 1.6 (Donut): Hỗ trợ tìm kiếm bằng giọng nói Voice Search, công nghệ CDMA/EVDO, màn hình WVGA.
- 2.0/2.1 (Eclair): Cải tiến lại giao diện người dùng, giới thiệu HTML5, hỗ trợ Exchange ActiveSync 2.5.
- 2.2 (Froyo): Nâng cấp tốc độ xử lý, giới thiệu engine Chrome V8 JavaScript, hỗ trợ Adobe Flash, thêm tính năng tạo điểm truy cập Wi-Fi.
- 2.3 (Gingerbread): Cải thiện giao diện người dùng, bàn phím ảo, thêm tính năng copy/paste, hỗ trợ công nghệ giao tiếp tầm sóng ngắn NFC.
- 3.0 (Honeycomb): Hệ điều hành dành riêng cho máy tính bảng với giao diện mới, hỗ trợ bộ xử lý đa nhân và xử lý đồ họa.
- 4.0.x (Ice-cream sandwich): Hệ điều hành mới nhất là sự kết hợp giữa Gingerbread và Honeycomb.

c. Thực trạng hiện nay

Android là hệ điều hành mở, do liên minh của Google phát triển (Google trực tiếp). Android đang lớn mạnh từng ngày và có khả năng sẽ vượt mặt Iphone OS trong dòng Smartphone. Bởi tốc độ xử lý nhanh, bộ nhớ SDCard lớn cho phép thực hiện công việc hay giải trí, nên điện thoại đã dần dần thay thế cho máy tính.



Hình 2.1. Ứng dụng trên Android

(Nguồn <http://google.com>)

Hệ điều hành Android với lợi thế là một hệ điều hành mã nguồn mở, được hỗ trợ bởi Google và đông đảo cộng đồng phát triển nên Android đang trở thành xu hướng phát triển của các hãng di động như HTC, LG, Samsung, Motorola, Kyocera, Sony Ericsson.

Trong bối cảnh thị trường Smartphone sôi động như hiện nay, các hệ điều hành luôn cạnh tranh với nhau để giành lấy thị trường. Hệ điều hành có thị phần cao nhất năm 2012 là Android.

Thêm vào đó, xu hướng công nghệ hiện nay đang coi Android là nền tảng để phát triển ứng dụng cho các thiết bị di động. Một bức tranh phác thảo toàn cảnh về Android cho thấy Android ra đời đã mang lại cho người dùng điện thoại sự cảm nhận khác biệt về một chiếc [Smartphone](#), chưa đầy 2 năm kho [phần mềm cho Android](#) đã lên đến còn số hơn 30.000 ứng dụng, khoảng 700.000 điện thoại Android được kích hoạt mỗi ngày.

2.1.2. Kiến trúc ứng dụng

Kiến trúc Android gồm 4 phần chính. Nhân của Android được phát triển dựa vào Kernel Linux 2.6. Mỗi tầng trong kiến trúc Android hoạt động dựa vào tầng bên dưới nó.

Hình 2.2. Kiến trúc hệ điều hành Android

(Nguồn <http://byeonely.tistory.com>)

a. Linux Kernel

Linux Kernel là lớp dưới cùng trong kiến trúc của hệ điều hành Android. Linux Kernel được phát triển từ nhân Linux phiên bản 2.6 với khoảng 115 bản vá có các chức năng cơ bản là quản lý bộ nhớ, giao tiếp phần cứng, thực hiện bảo mật, quản lý tiến trình. Linux Kernel còn có chức năng tạo lớp giao tiếp ảo với các tầng trên của nó.

b. Library và Android Runtime

Library: chứa một tập hợp các thư viện được viết từ ngôn ngữ C/C++ để cung cấp cho các thành phần khác của Android.

Android Runtime: là môi trường thực thi ứng dụng Android. Thực chất của Android Runtime là một máy ảo java để chạy các ứng dụng Java được lập trình.

c. Application Framework

Application Framework là thành phần phía trên của Library và Android Runtime với chức năng cung cấp một nền tảng mở và mạnh mẽ để phát triển ứng dụng Java:

- View UI: để xây dựng layout của ứng dụng bao gồm các đối tượng như List View, TextView, EditText, Button, Dialog, ...vv
- Content Providers: cho phép các ứng dụng truy cập dữ liệu từ các ứng dụng khác hoặc để chia sẻ dữ liệu của riêng ứng dụng.
- Resource Manager: cung cấp cách thức truy cập đến non-code resources như các asset, graphic, image, music, video...vv
- Notification Manager: cho phép tất cả các ứng dụng hiển thị thông báo trên hệ điều hành.
- Activity Manager: có chức năng quản lý vòng đời của các ứng dụng.

d. Applications

Applications là tầng trên cùng trong kiến trúc hệ điều hành Android. Applications là hệ thống phần mềm ứng dụng chạy trên Android. Các phần mềm ứng dụng này viết bằng ngôn ngữ Java và chạy trên máy ảo Dalvik Virtual Machine.

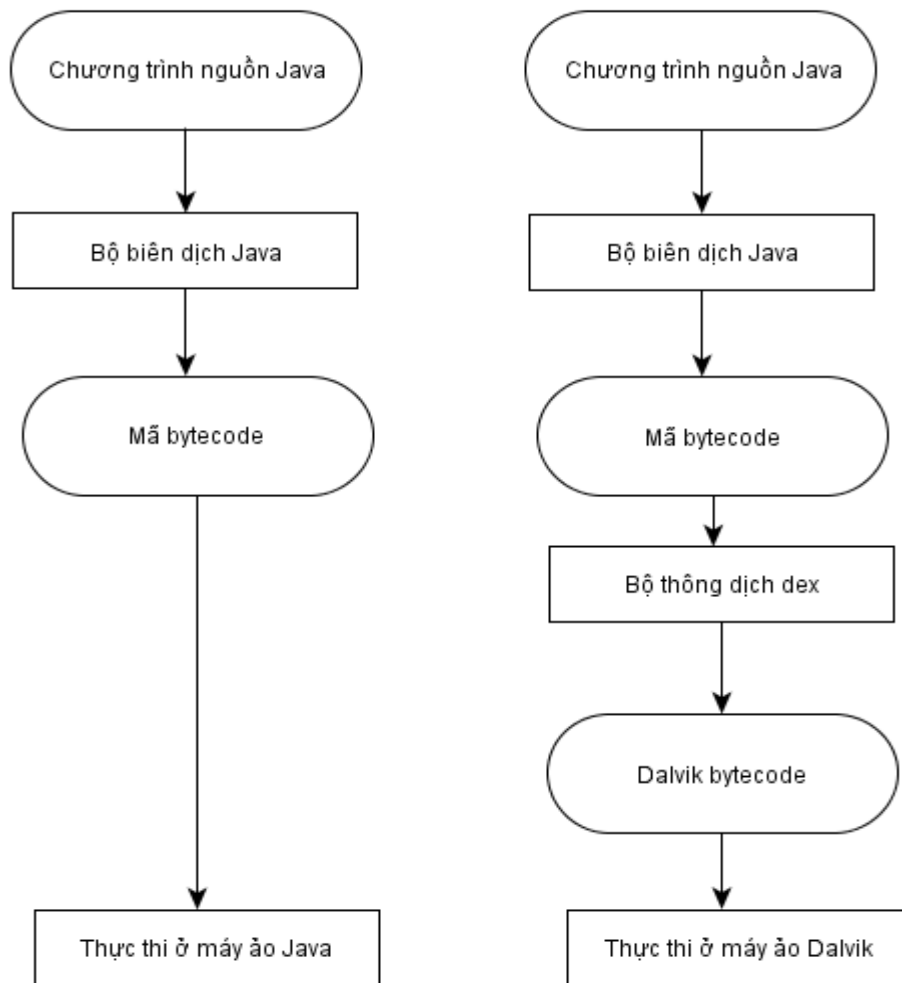
Applications còn tích hợp sẵn một số ứng dụng cơ bản như danh bạ, lịch, gọi điện thoại.

2.1.3. Máy ảo Dalvik

Máy ảo Dalvik (DVM) là một máy ảo chạy các ứng dụng được viết bằng Java. Chương trình nguồn Java sẽ được biên dịch thành tập tin chứa mã bytecodes, sau đó các tập tin này được biên soạn thành một tập tin dex. Tiếp theo đó, máy ảo Dalvik sẽ đọc và xử lý tập tin dex.

Thành phần quan trọng của hệ thống Android là máy ảo Dalvik được thiết kế cho Android với 2 nhiệm vụ chính là giúp chạy nhiều luồng xử lý đồng thời và đóng vai trò là một bộ nhớ hiệu quả.

Một trong những đặc điểm của Android là sử dụng máy ảo Dalvik. So với máy ảo truyền thống của Java, máy ảo Dalvik được xem là tốt hơn vì nó được viết chỉ dành riêng cho hệ điều hành Android sử dụng. Bên cạnh, máy ảo Dalvik còn là cơ sở cho việc chạy các ứng dụng (các tập tin chương trình .apk).



Hình 2.3. Quá trình chạy ứng dụng của Java VM và Dalvik VM

Dalvik cache là một bộ nhớ cache đơn giản được sử dụng bởi Dalvik và đó là kết quả của việc Dalvik làm tối ưu hóa các ứng dụng đang chạy. Một số ROM Android cho phép bạn di chuyển các bộ nhớ Dalvik cache vào thẻ SD của máy, để giải phóng bộ nhớ trong của máy.

2.1.4. Chu kì sống của ứng dụng

Trong một ứng dụng Android có chứa nhiều thành phần và mỗi thành phần đều có một chu trình sống riêng. Ứng dụng chỉ được gọi là kết thúc khi tất cả các thành phần trong ứng dụng kết thúc. Activity là một thành phần cho phép người dùng giao tiếp với ứng dụng, còn các thành phần không có khả năng tương tác với người dùng như là Service, BroadcastReceiver.

Nếu một Activity được tạm dừng hoặc dừng hẳn, hệ thống có thể bỏ thông tin của nó từ vùng nhớ bởi việc finish(), hoặc đơn giản “kill” tiến trình của nó. Khi nó được hiển thị lần nữa, nó sẽ phải được restart và phục hồi lại trạng thái trước. Khi một Activity chuyển đổi qua lại giữa các trạng thái, nó phải báo việc chuyển của nó bằng việc gọi hàm transition.

Tất cả các phương thức có liên quan chặt chẽ với nhau, có thể ghi đè lên nhau (override) để làm tương thích công việc trong ứng dụng khi thay đổi trạng thái. Tất cả các Activity bắt buộc phải có phương thức onCreate() để khởi tạo ứng dụng. Nhiều Activity sẽ cũng thực hiện onPause() ví dụ như để xác nhận việc thay đổi dữ liệu hoặc chuẩn bị dừng hoạt động.

2.1.5. Các tiến trình

Android có cơ chế quản lý các tiến trình (process) theo chế độ ưu tiên (priority). Các tiến trình có độ ưu tiên thấp sẽ bị giải phóng mà không hề có cảnh báo nhằm đảm bảo tài nguyên.

- Foreground process: là tiến trình của ứng dụng hiện thời đang được người dùng tương tác.
- Visible process: là tiến trình của ứng dụng mà Activity đang hiển thị đối với người dùng (phương thức onPause() của Activity được gọi).
- Service process: là Service đang hoạt động (running).
- Background process: là tiến trình của ứng dụng mà các Activity của nó không hiển thị với người dùng (phương thức onStoped() của Activity được gọi).
- Empty process: tiến trình không có bất cứ thành phần nào hoạt động. Theo chế độ ưu tiên thì khi cần tài nguyên, Android sẽ tự động đóng tiến trình (kill process), trước tiên là các Empty process.

2.1.6. Công cụ lập trình

Ngôn ngữ lập trình: Ứng dụng Android đa số được viết trên ngôn ngữ Java, bên cạnh đó, ứng dụng trên hệ điều hành Android cũng có thể được viết bằng một số ngôn ngữ khác như C/C++, C# và Python.

Quy trình phát triển phần mềm ứng dụng:

- Bước 1. Thiết kế (Design).
- Bước 2. Viết mã (Code).
- Bước 3. Biên dịch (Compile).
- Bước 4. Tiên kiểm tra (Preverify).
- Bước 5. Đóng gói (Package): Sinh ra tập tin APK.
- Bước 6. Triển khai (Deploy): Đưa các tài nguyên của ứng dụng dưới sự quản lý của bộ giả lập (emulator).
- Bước 7. Thực thi (Execute): Chạy ứng dụng với bộ giả lập.
- Bước 8. Gỡ lỗi (Debug): Nhận ra các lỗi và sửa lại mã nguồn.
- Bước 9. Phát hành (Publish): Phát hành ứng dụng tới người dùng

Android SDK là công cụ phát triển ứng dụng Android. Android SDK có thể được sử dụng để chạy và kiểm tra ứng dụng được phát triển. Thiết bị giả lập Android SDK được trang bị đầy đủ hầu hết các tính năng của một thiết bị thật. Tuy nhiên cũng có một số tính năng không được hỗ trợ như kết nối USB, camera, quay video, tai nghe, giả lập pin và bluetooth. Ngoài ra tốc độ khởi động của thiết bị giả lập thường rất chậm gây mất nhiều thời gian.

Công cụ lập trình: Eclipse for Java Developers hoặc Eclipse for Java and Report Developers.

2.1.7. Các thành phần trên ứng dụng Android

a. Activity

Activity là một thành phần ứng dụng cung cấp giao diện mà người dùng có thể tương tác với ứng dụng tương tự như window trong ứng dụng Desktop. Ví dụ như gọi điện thoại, chụp ảnh, gửi email, hiển thị bản đồ...

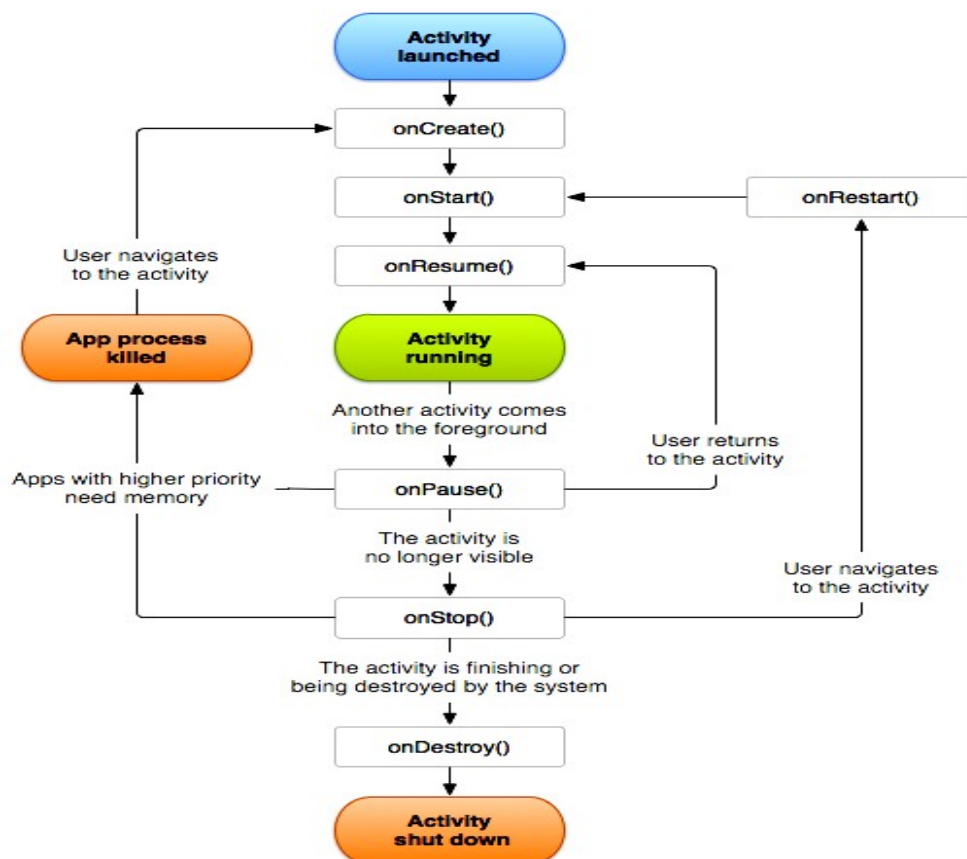
Activity là thành phần quan trọng nhất, nó đóng vai trò chính trong việc xây dựng các ứng dụng Android. Activity được quản lý theo dạng ngăn xếp (stack), tức là khi một Activity mới được khởi tạo, nó sẽ được xếp lên đầu của ngăn xếp và trở thành running activity, các Activity trước đó sẽ bị tạm dừng và chỉ hoạt động trở lại khi Activity mới được giải phóng.

Một Activity chủ yếu có 3 chu kỳ chính sau:

- Active hoặc running: Activity đang ở đầu ngăn xếp và được chạy trên màn hình. Người dùng có thể tương tác trực tiếp với Activity này bằng cách chạm vào màn hình.
- Paused: Activity được tạm dừng (paused) khi người dùng không thể tương tác nhưng vẫn trông thấy, tức là có một Activity mới ở trên nó nhưng không bao phủ đầy màn hình. Một Activity tạm dừng là còn “sống” nhưng có thể bị kết thúc bởi hệ thống trong trường hợp thiếu vùng nhớ.
- Stopped: Activity hoàn toàn bị bao phủ bởi Activity khác. Nó vẫn còn trạng thái và thông tin lưu trữ trong nó. Nhưng người dùng không thấy nó và thường bị loại bỏ trong trường hợp hệ thống cần vùng nhớ cho tác vụ khác.

Các phương thức trong chu kỳ sống của Activity: `void onCreate(Bundle savedInstanceState)`, `void onStart()`, `void onRestart()`, `void onResume()`, `void onPause()`, `void onStop()`, `void onDestroy()`.

Vòng đời của Activity:



Hình 2.4. Vòng đời của Activity

(Nguồn: [http:// developer.android.com](http://developer.android.com))

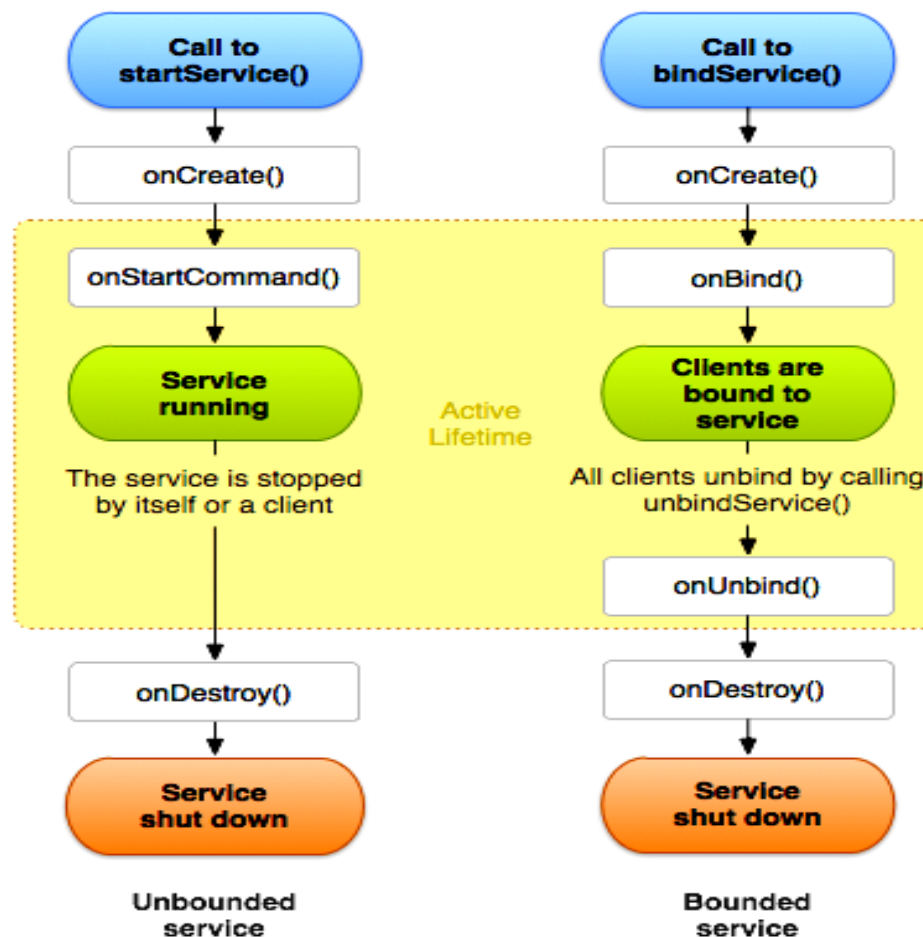
b. Service

Service là thành phần ứng dụng có thể thực thi ngầm, sử dụng để cập nhật dữ liệu, đưa ra các thông báo (Notification) và không cung cấp giao diện người dùng.

Một ứng dụng khác có thể khởi động Service và Service sẽ chạy ngay cả khi người dùng chuyển sang ứng dụng khác.

Thông thường, trong các chương trình trò chơi online, Service là thành phần trung tâm, nơi giao tiếp thực hiện việc truyền nhận dữ liệu từ Server. Thông qua đó, Service sẽ cập nhật các giao diện màn hình từ các Activity.

Vòng đời Service:



Hình 2.5. Vòng đời của Service

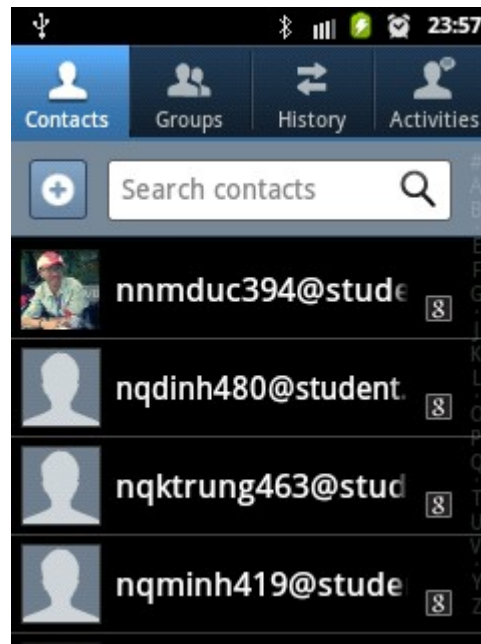
(Nguồn: [http:// developer.android.com](http://developer.android.com))

c. Content Provider

Content Provider có chức năng cung cấp tập cấu trúc dữ liệu từ ứng dụng này đến ứng dụng khác trong hệ điều hành Android.

Content Provider còn là kho dữ liệu chia sẻ, ví dụ: danh bạ, nhật ký cuộc gọi, cấu hình cài đặt.

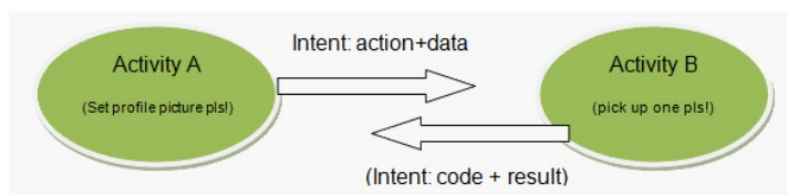
Content Provider là một tập các phương thức chuẩn mà các ứng dụng khác có thể truy xuất và lưu trữ dữ liệu của loại nó điều khiển.



Hình 2.6. Kho dữ liệu danh bạ

d. Intent

Intent dùng để truyền tải dữ liệu giữa các Activity, giữa Activity và Service, truyền tải và gửi các thông báo đi nhằm khởi tạo một Activity hay Service. Ngoài ra, Intent là một cơ cấu cho phép truyền thông điệp giữa các thành phần của 1 ứng dụng và giữa các ứng dụng với nhau.



Hình 2.7. Sơ đồ hoạt động của Intent

(Nguồn: [http:// developer.android.com](http://developer.android.com))

e. IntentFilter

IntentFilter là bộ lọc Intent, chỉ cho phép những Intent trùng khớp với nó đi qua.

Activity, Service và Broadcast Receiver sử dụng Intent Filter để thông báo cho hệ thống biết các dạng Implicit Intent mà nó có thể xử lý.

f. BroadcastReceiver

BroadcastReceiver có chức năng dùng để nhận các sự kiện mà các ứng dụng hoặc hệ thống phát đi.

Một BroadcastReceiver không có giao diện nhưng nó có thể thực hiện gọi một Activity hay là sử dụng NotificationManager để thông báo cho người dùng.

g. Notification

Notification có chức năng đưa ra các thông báo mà không làm Activity phải ngừng hoạt động.

2.1.8. Lập trình giao diện trên Android

a. Layout

Công dụng của layout là dùng để bố trí các đối tượng trên màn hình. Layout được khai báo bằng cách sử dụng xml hoặc viết code.

Có nhiều loại layout khác nhau:

- LinearLayout
- AbsoluteLayout
- TableLayout
- RelativeLayout
- FrameLayout

b. TextControls

TextControl là các đối tượng dùng để hiển thị hoặc điền thông tin dạng text.

Có hai loại textcontrols:

- TextView
- EditText

c. ButtonControls

ButtonControls là các đối tượng được sử dụng rất phổ biến trong ứng dụng Android.

ButtonControls Gồm có:

- Button
- ImageButton
- Checkbox
- RadioButton và RadioGroup

d. ImageView

ImageView dùng để hiển thị hình ảnh trên màn hình



Hình 2.8. Image View

e. ListView

List View không phải là một đối tượng cụ thể, nó được sử dụng như là nơi chứa một số đối tượng rồi hiển thị theo một cấu trúc đặc biệt và có thể xử lý từng thành phần trên danh sách này.



Hình 2.9. Giao diện trên Android
(Nguồn <http://google.com>)

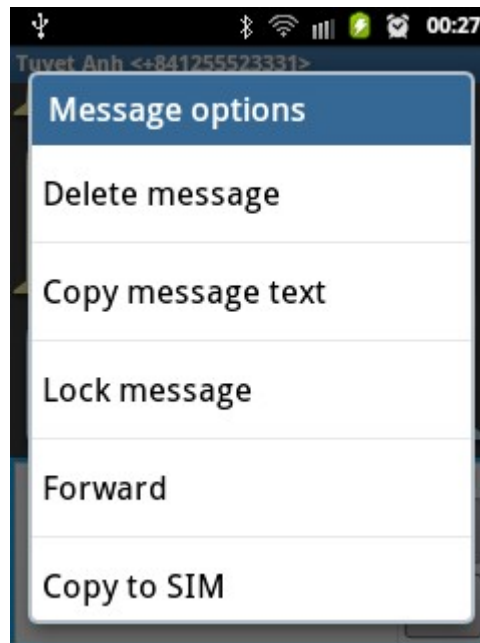
f. Menu

OptionsMenu: là menu xuất hiện khi ấn phím Menu trên thiết bị.



Hình 2.10. OptionsMenu

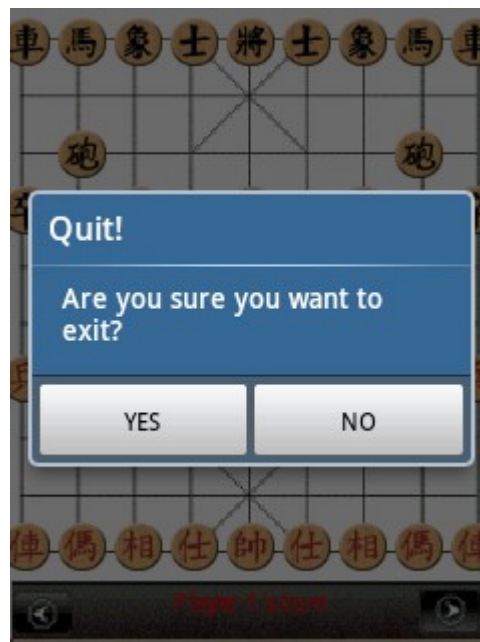
ContextMenu: là menu xuất hiện khi ấn lên đối tượng và giữ.



Hình 2.11. ContextMenu

g. Dialog

Dialog dùng để hiển thị thông báo dưới dạng hộp thoại.



Hình 2.12. Dialog

h. Toast

Toast hiển thị thông báo dưới dạng message.

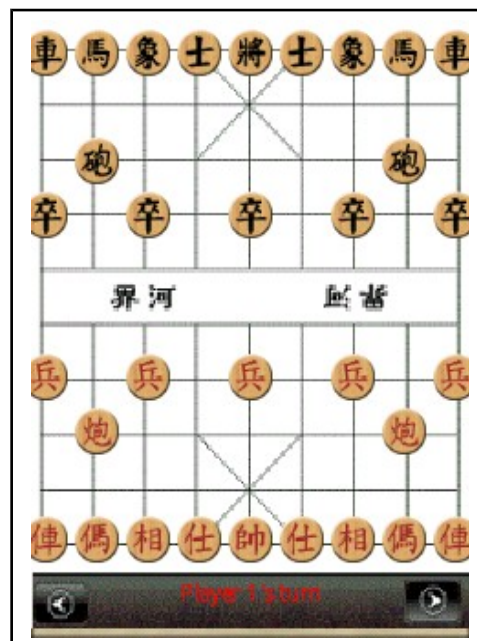


Hình 2.13. Toast

(Nguồn <http://android-codes-examples.blogspot.com>)

i. View

Đối tượng View dùng để lập trình trên giao diện mức thấp. Đối tượng View sử dụng đối tượng Canvas để vẽ lên màn hình. Chính vì vậy, nó thường được sử dụng trong việc thiết kế game.

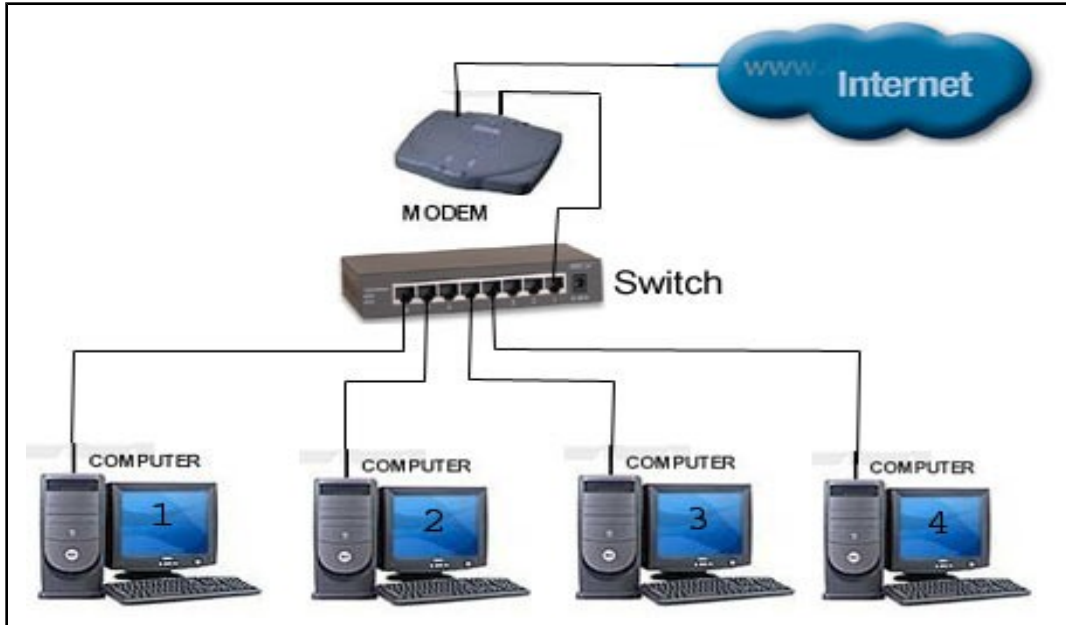


Hình 2.14. Giao diện View

2.2. TỔNG QUAN VỀ MẠNG MÁY TÍNH

2.2.1. Khái niệm

Mạng máy tính là mạng của hai hay nhiều máy tính được nối lại với nhau bằng một kiến trúc vật lý nào đó.



Hình 2.15. Mạng máy tính

(Nguồn <http://google.com>)

2.2.2. Các thành phần

Mạng máy tính bao gồm 3 thành phần: đường biên mạng, mạng đường trục và mạng truy cập.

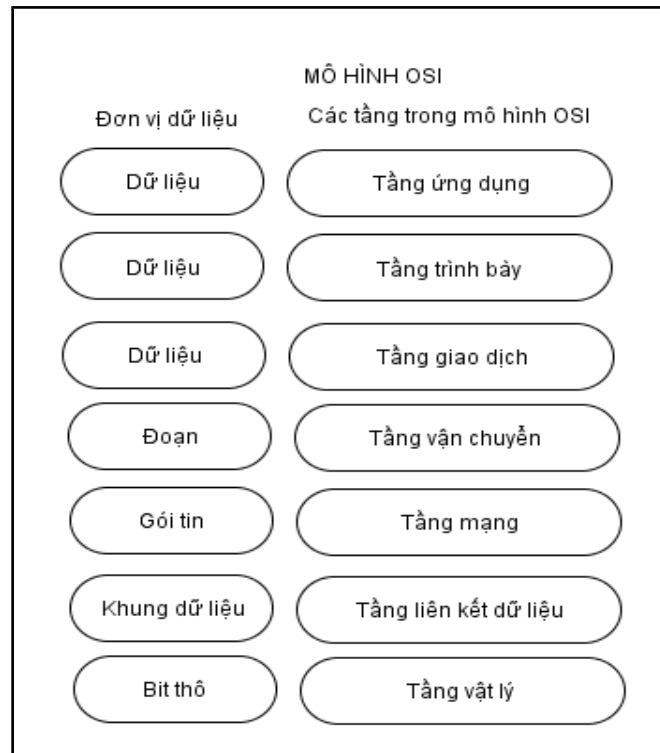
Đường biên mạng (Network Edge): bao gồm các máy tính và ứng dụng được tổ chức theo mô hình Client-Server, Peer2Peer.

Mạng đường trục (Network Core): là mạng nối các router đảm bảo truyền tin thông suốt giữa hai máy cách xa nhau.

Mạng truy cập (Access Network): nối các máy tính vào các router ngoài bìa.

2.2.3. Mô hình OSI

Mô hình OSI (Open System Interconnection Model) là mô hình tham khảo được phát triển bởi tổ chức tiêu chuẩn thế giới ISO (International Standard Organization).



Hình 2.16. Mô hình OSI

Theo mô hình OSI, mạng máy tính được chia ra làm 7 tầng:

- Tầng vật lý (Physical layer): đơn vị truyền nhận dữ liệu là bit (Bits), truyền tải các bit thô trên một kênh truyền vật lý.
- Tầng liên kết dữ liệu (Data link layer): đơn vị truyền nhận dữ liệu là khung(Frames). Tầng liên kết dữ liệu là kênh truyền nối trực tiếp hai máy tính, có vai trò thiết lập cơ chế phát hiện và xử lý lỗi, điều khiển dòng, giải quyết tranh chấp đường truyền.
- Tầng mạng (Network layer): đơn vị truyền nhận dữ liệu là gói tin (Packets), có chức năng chính là định tuyến và chuyển tiếp các gói tin, kiểm tra khắc phục tình trạng nghẽn đường truyền.

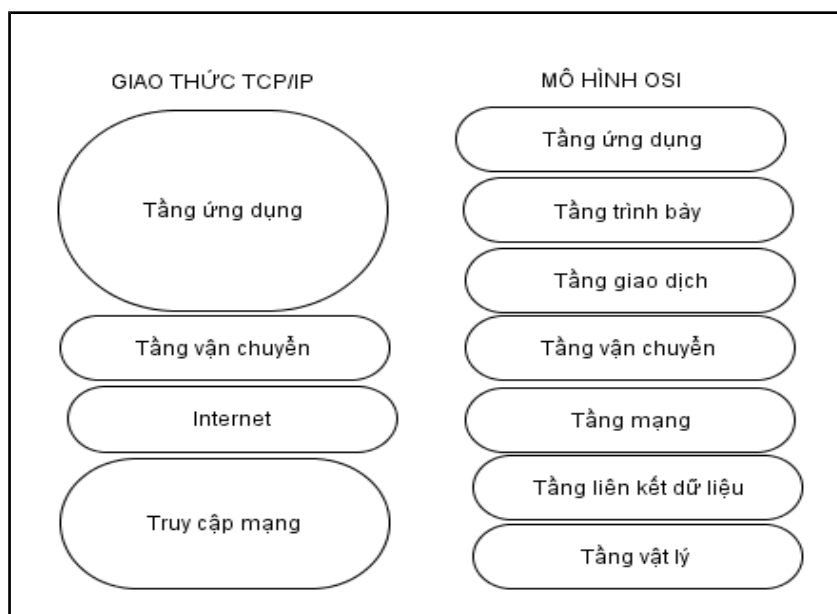
- Tầng vận chuyển (Transport layer): đơn vị truyền nhận dữ liệu là đoạn (Segments), truyền dữ liệu từ điểm tới điểm, kiểm tra gói tin mất, trùng lặp, đa hợp và phân hợp.
- Tầng giao dịch: đơn vị truyền nhận dữ liệu là dữ liệu (Data), quản lý các giao dịch, đồng bộ hóa dữ liệu.
- Tầng trình bày: đơn vị truyền nhận dữ liệu là dữ liệu (Data), chuẩn hóa dữ liệu trao đổi giữa các hệ thống khác nhau.
- Tầng ứng dụng: đơn vị truyền nhận dữ liệu là dữ liệu (Data), là các chương trình ứng dụng phần mềm dịch vụ. Ví dụ: email, web, ftp...

2.2.4. Giao thức TCP/IP

a. Giới thiệu

Mặc dù mô hình OSI được phân chia rất cụ thể và là mô hình rất nổi tiếng về mạng máy tính nhưng nó chỉ là mô hình tham khảo, không được áp dụng vào thực tế. Như vậy, các mạng máy tính ngày nay làm sao có thể trao đổi dữ liệu, đó là nhờ giao thức TCP/IP – Bộ giao thức chuẩn của mạng máy tính hiện nay.

Giao thức TCP/IP (Transmission Control Protocol/Internet Protocol) được phát triển từ mạng ARPANET và Internet và được dùng như giao thức mạng và vận chuyển trên mạng Internet. Họ giao thức TCP/IP hiện nay là giao thức nền tảng được sử dụng rộng rãi nhất để liên kết các máy tính và các mạng.



Hình 2.17. Mô hình OSI và giao thức TCP/IP

b. Các tầng của giao thức TCP/IP

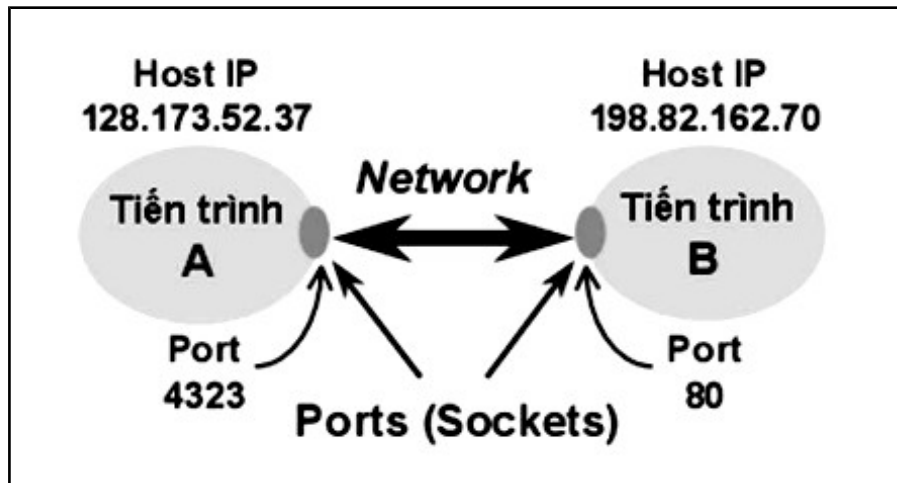
Giao thức TCP/IP được chia làm 4 tầng:

- Tầng truy cập mạng (Network Access layer) bao gồm một số giao thức như: ATM (Asynchronous Transfer Mode), Ethernet, Token Ring, Frame Relay,...
- Tầng Internet (Internet layer) gồm 4 giao thức:
 - + IP (Internet Protocol): có chức năng gán địa chỉ cho dữ liệu trước khi truyền và định tuyến chúng tới đích.
 - + ARP (Address Resolution Protocol): là giao thức phân giải địa chỉ IP thành địa chỉ MAC.
 - + ICMP (Internet Control Message Protocol): có chức năng thông báo.
 - + IGMP (Internet Group Management Protocol): có chức năng điều khiển truyền đa hướng (Multicast).
- Tầng vận chuyển (Transport layer) gồm 2 giao thức chính: TCP và UDP.
 - + TCP (Transmission Control Protocol): là giao thức cung cấp dịch vụ vận chuyển tin cậy bằng cách sử dụng cơ chế báo nhận, hướng nối kết theo kiểu truyền thông tin bằng cách phân luồng các byte. Bên cạnh đó, TCP còn là giao thức truyền song công, hỗ trợ cơ chế đa hợp.
 - + UDP (User Datagram Protocol): là giao thức cung cấp dịch vụ vận chuyển không nối kết không tin cậy.
- Tầng ứng dụng (Application layer): gồm nhiều giao thức ứng dụng như Tenet, FTP, SMTP, POP, IMAP, HTTP, DNS, SMNP...

2.3. GIỚI THIỆU VỀ LẬP TRÌNH SOCKET

2.3.1. Khái niệm về socket

Xét về góc độ mạng, Socket là 1 trong 2 điểm cuối của đường nối kết 2 chiều giữa 2 chương trình thực thi trên mạng.



Hình 2.18. Mô hình nối kết Socket

(Nguồn <http://google.com>)

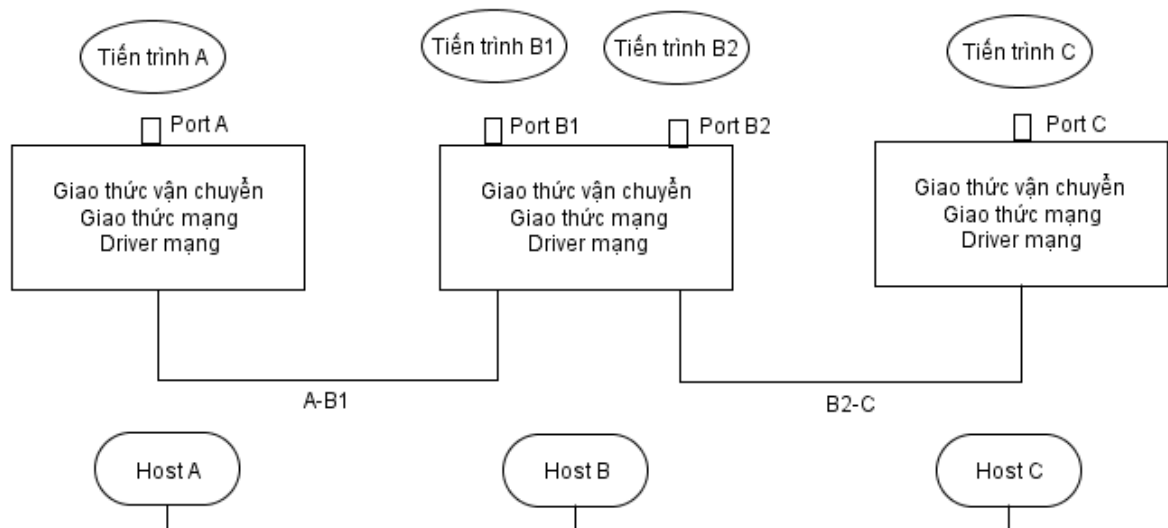
Xét ở góc độ người lập trình, Socket là giao diện lập trình ứng dụng (API) hay bộ thư viện hàm hỗ trợ, dùng để nối kết chương trình ứng dụng với lớp mạng trong hệ thống mạng TCP/IP.

Socket được giới thiệu lần đầu dưới hệ điều hành UNIX version 4.3 BSD.

2.3.2. Cơ chế giao tiếp

Trong cơ chế giao tiếp thông qua Socket, một trong hai quá trình phải công bố số hiệu cổng của Socket mà mình sử dụng để nhận và gửi dữ liệu.

Các quá trình khác có thể giao tiếp với quá trình đã công bố cổng cũng bằng cách tạo ra một Socket.



Hình 2.19. Cơ chế giao tiếp giữa các máy tính thông qua Socket

Trong Java, lớp Socket và ServerSocket hỗ trợ việc lập trình mạng giao tiếp giữa máy Client và Server.

Các bước cần có để tạo kết nối ở phía Client:

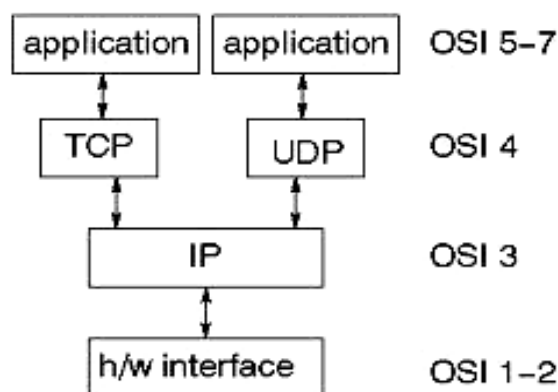
- Tạo một Socket để kết nối với Server.
- Định nghĩa các dòng nhập/xuất với Socket .
- Sử dụng các dòng nhập xuất để nhận/gửi dữ liệu.
- Đóng các dòng nhập/xuất và Socket khi không còn trao đổi.

Các bước cần có để tạo kết nối ở phía Server:

- Tạo ServerSocket lắng nghe ở một nào đó.
- Tạo một Socket accept nối kết với Client.
- Kết nối các dòng nhập/xuất với Socket .
- Sử dụng các dòng nhập xuất để nhận/gửi dữ liệu.
- Đóng các dòng nhập/xuất, Socket, ServerSocket.

2.3.3. Các chế độ giao tiếp

Trong lập trình Socket, có 2 chế độ giao tiếp, đó là TCP (Transmission Control Protocol) và UDP(User Datagram Protocol).



Hình 2.20. Chế độ giao tiếp TCP và UDP

(Nguồn <http://google.com>)

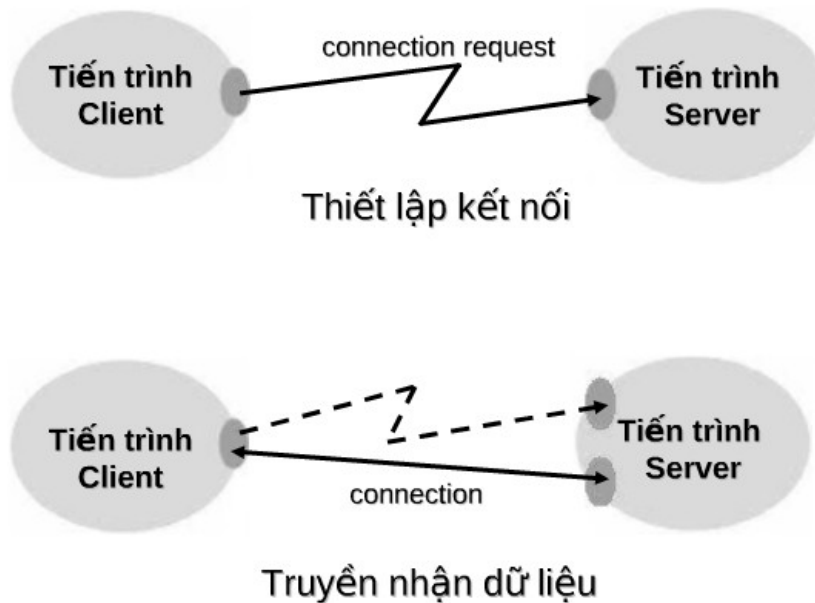
2.3.1. So sánh giữa TCP và UDP

TCP	UDP
TCP là chế độ giao tiếp có nối kết.	UDP là chế độ giao tiếp không nối kết.
Giữa 2 quá trình Client và Server luôn tồn tại kênh giao tiếp ảo.	UDP không tồn tại kênh giao tiếp ảo giữa 2 quá trình Client và Server
Đối với TCP, dữ liệu được gửi đi theo chế độ bảo đảm: có kiểm tra lỗi, truyền lại gói tin lỗi hay mất, bảo đảm thứ tự đến của các gói tin ...	Đối với UDP, dữ liệu được gửi đi theo chế độ không bảo đảm: không kiểm tra lỗi, không phát hiện và không truyền lại gói tin bị lỗi hay bị mất, không bảo đảm thứ tự đến của các gói tin ...
Chính vì gửi đi theo chế độ đảm bảo nên dữ liệu truyền đi chính xác và tốc độ truyền chậm hơn UDP.	Dữ liệu truyền đi trong chế độ giao tiếp UDP không chính xác bằng TCP và tốc độ truyền nhanh hơn.
Chế độ giao tiếp TCP thích hợp cho các ứng dụng cần độ chính xác cao: truyền file, thông tin điều khiển ...	Chế độ giao tiếp UDP thích hợp cho các ứng dụng cần tốc độ, không cần chính xác cao: truyền âm thanh, hình ảnh ...

Bảng 2.1. So sánh 2 chế độ giao tiếp TCP và UDP

(Nguồn: *Lập trình truyền thông – Khoa CNTT & TT – Đại học Cần Thơ*)

2.3.2. Xây dựng chương trình Socket ở chế độ có nối kết (TCP)



Hình 2.21. Thiết lập có nối kết TCP

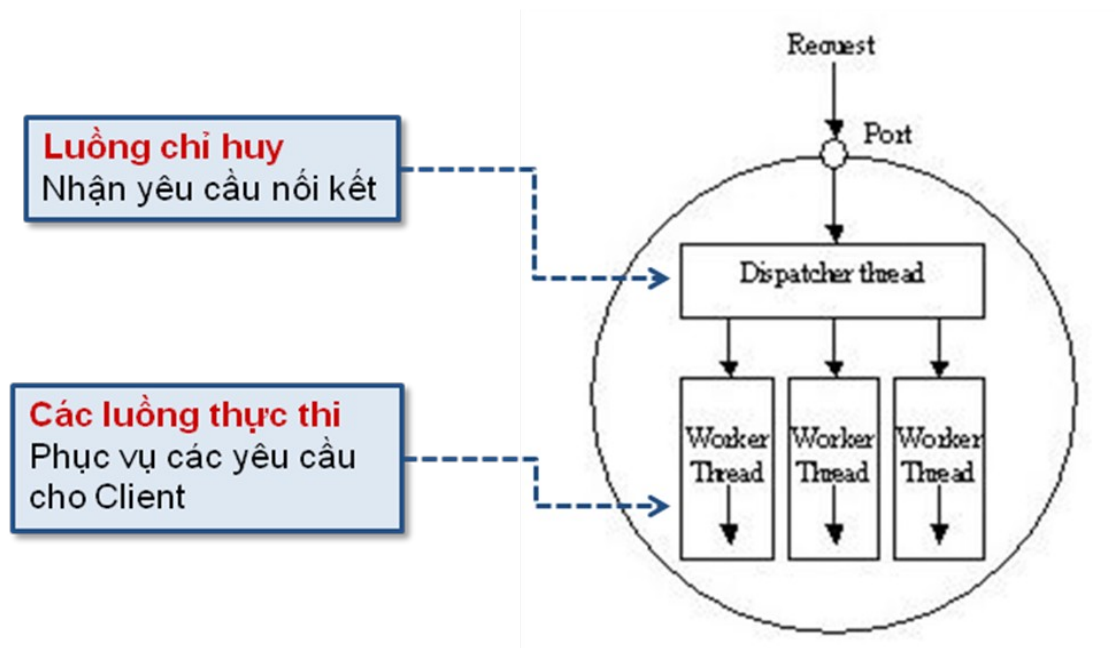
(Nguồn <http://google.com>)

Client:

- Mở một socket nối kết đến Server đã biết địa chỉ IP (hay địa chỉ tên miền) và số hiệu cổng.
- Lấy InputStream và OutputStream gán với Socket.
- Tham khảo protocol của dịch vụ để định dạng đúng dữ liệu trao đổi với Server.
- Trao đổi dữ liệu với Server nhờ vào các InputStream và OutputStream vừa lấy.
- Đóng Socket trước khi kết thúc chương trình.

Chương trình Server phục vụ song song được thiết kế gồm 2 phần:

- Phần 1: Xử lý các yêu cầu nối kết.
- Phần 2: Xử lý các thông điệp yêu cầu từ Client.



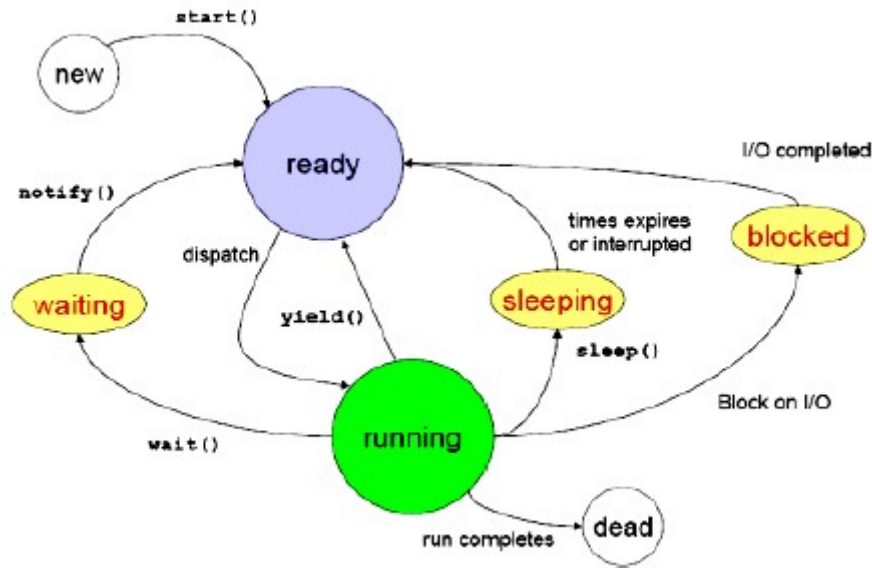
Hình 2.22. Server xử lý song song

(Nguồn Lập trình truyền thông – Khoa CNTT & TT – Đại học Cần Thơ)

- Phần 1: Lặp lại các công việc sau:
 - + Lắng nghe yêu cầu nối kết của Client.
 - + Chấp nhận một yêu cầu nối kết:
 - o Tạo kênh giao tiếp ảo mới với Client.
 - o Tạo phần 2 để xử lý các thông điệp yêu cầu của Client.
- Phần 2: Lặp lại các công việc sau:
 - + Chờ nhận thông điệp yêu cầu của Client.
 - + Phân tích và xử lý yêu cầu.
 - + Gửi thông điệp trả lời cho Client.
- Phần 2 sẽ kết thúc khi kênh ảo bị xóa đi.
- Phần 2 được thiết kế là 1 luồng (để có thể thực thi song song).

2.4. LẬP TRÌNH LUỒNG (THREAD)

Trong hệ điều hành hiện đại, đơn vị sử dụng CPU là luồng (thread). Luồng cung cấp cơ chế tính toán và xử lý song song trong các ứng dụng. Mỗi luồng có riêng bộ đếm chương trình, trạng thái thanh ghi và ngăn xếp. Luồng sẽ chia sẻ không gian địa chỉ, do đó dễ dàng chia sẻ file, hiệu báo.



Hình 2.23. Các trạng thái của luồng

(Nguồn <http://google.com>)

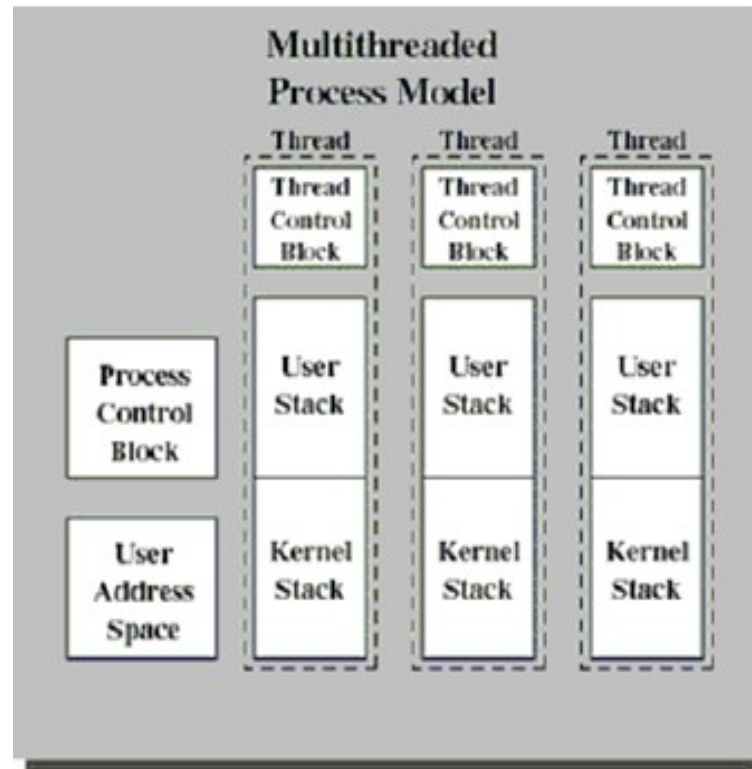
Trong ngôn ngữ lập trình Java, hỗ trợ cơ chế lập trình đa luồng bởi việc thừa kế lớp `java.lang.Thread` hoặc cài đặt lớp `Runnable`

Một số phương thức của lớp `java.lang.Thread`:

- `public static void sleep(long n)` throws `InterruptedException`: Blocked trong khoảng `n` mili giây.
- `public void run()`: thực thi luồng.
- `public final void suspend()`: chuyển từ Ready -> Blocked.
- `public final void resume()`: chuyển từ Blocked -> Ready.
- `public final void yield()`: chuyển từ Running -> Ready.

Các phương thức thừa kế từ `java.lang.Object`

- `public void wait(long n)` throws `InterruptedException`: dừng luồng trong `n` mili giây xác định.
- `public final void notify()`: đánh thức 1 luồng trong hàng đợi.
- `public final void notifyAll()`: đánh thức tất cả các luồng trong hàng đợi.



Hình 2.24. Quá trình đa luồng

(Nguồn <http://google.com>)

2.5. TÍNH TOÁN LƯỚI VÀ LẬP TRÌNH SONG SONG

Ngày nay, tính toán thông thường không thể đáp ứng được yêu cầu của các vấn đề tính toán phức tạp. Các siêu máy tính vẫn được phát triển cho đến ngày nay để phục vụ vào mục đích siêu tính toán. Tuy nhiên chi phí phải trả để có một chiếc siêu máy tính không hề nhỏ. Thêm vào đó, nhiều vấn đề thực tiễn được đặt ra như xử lý dữ liệu khổng lồ, nhu cầu tích hợp tài nguyên tính toán, tận dụng tài nguyên rồi cũng như yêu cầu tốc độ tính toán. Vì vậy, tính toán lưới ra đời được xem như là bước ngoặt lớn để giải quyết các vấn đề trên.

2.5.1. Hệ thống lưới và tính toán lưới

Lưới là một hệ thống phân tán bao gồm một tập các tài nguyên đa dạng có thể là các siêu máy tính, Clusters, hệ thống lưu trữ.

Tính toán lưới là sự cho phép ảo hóa các chức năng tính toán phân tán, các nguồn xử lý, băng thông mạng và khả năng lưu trữ để tạo ra một hệ thống đơn nhất

cho phép người sử dụng truy cập vào các tính năng điện toán rộng lớn. Một các dễ hình dung, tính toán lưới sử dụng tập hợp các tài nguyên qua mạng tốc độ cao để đáp ứng công việc tính toán.

2.5.2. Lợi ích của tính toán lưới

Tính toán lưới đang nổi lên như là phương tiện tập hợp tài nguyên tính toán chi phí thấp để giải quyết những bài toán lớn. Được dự đoán là kiến trúc tính toán của thế kỷ 21, tính toán lưới thực sự đã đem rất nhiều lợi ích cho con người.

Tính toán lưới giúp tiết kiệm thời gian và tài nguyên bởi việc kết hợp nhiều máy tính lại với nhau và một chương trình được chạy trên nhiều máy tính, cùng với việc tận dụng tài nguyên rỗi trong hệ thống mạng như bộ nhớ, CPU.

Bên cạnh đó, tính toán lưới còn cho phép xử lý song song. Việc kết hợp nhiều CPU cho phép chúng cùng xử lý song song một hay nhiều thành phần độc lập và tăng khả năng mở rộng chương trình.

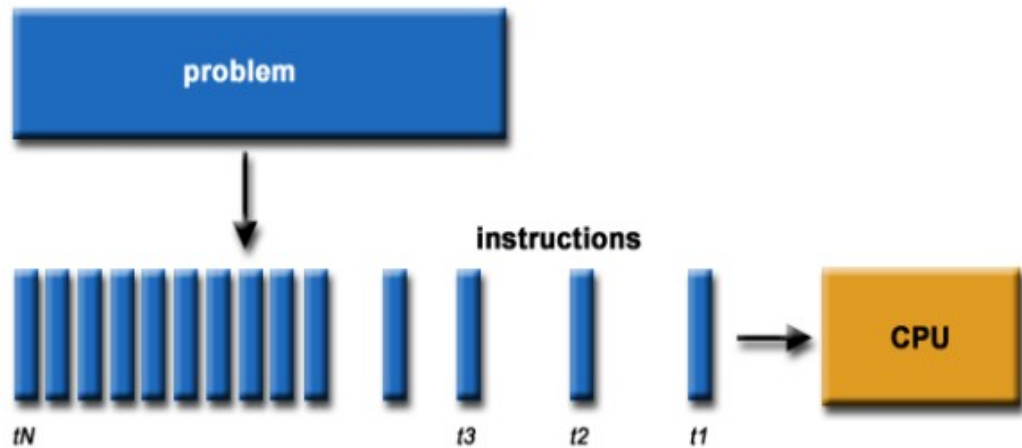
Một khả năng khác của tính toán lưới là cho phép tạo nên môi trường cộng tác rộng lớn, đồng nhất. Ngoài việc chia sẻ tài nguyên là các tập tin và cơ sở dữ liệu, tính toán lưới còn cho phép chia sẻ nhiều tài nguyên khác như thiết bị chuyên dụng, phần mềm, dịch vụ.

Ngoài ra, tính toán lưới giúp nâng cao độ tin cậy của hệ thống, cân bằng tài nguyên và dễ dàng quản lý.

2.5.3. Lập trình song song

Trong kĩ thuật lập trình trước đây, người ta thường thiết kế chương trình thực thi trên một máy tính đơn với một bộ vi xử lý. Điều đó mang đến sự dễ dàng trong việc thiết kế chương trình. Tuy nhiên, đối với những chương trình có lượng dữ liệu lớn, công việc tính toán phức tạp thì việc thực thi trên một máy tính sẽ tốn rất nhiều thời gian bởi khả năng của một máy tính là có giới hạn. Vì vậy, một kĩ thuật mới được ra đời là kĩ thuật lập trình song song ứng dụng tính toán lưới để giúp tiết kiệm thời gian và tăng tốc độ xử lý chương trình.

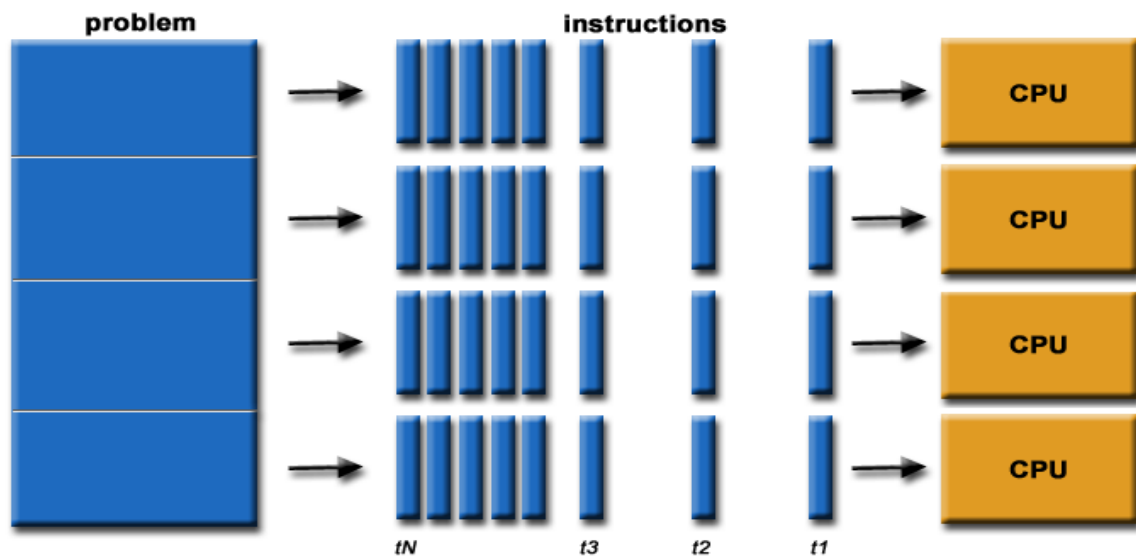
Nếu như các chương trình truyền thống được thiết kế và chạy trên một máy tính đơn với một tập lệnh rời rạc thì lập trình song song chương trình được thiết kế và chạy trên một hoặc nhiều máy tính với công việc được phân chia và thực thi song song.



Hình 2.25. Chương trình được thiết kế theo tập lệnh rời rạc

(Nguồn <https://computing.llnl.gov>)

Đối với chương trình được thiết kế theo tập lệnh rời rạc, vấn đề được chia ra thành vấn đề nhỏ và được xử lý tuần tự chỉ bởi một bộ xử lý.



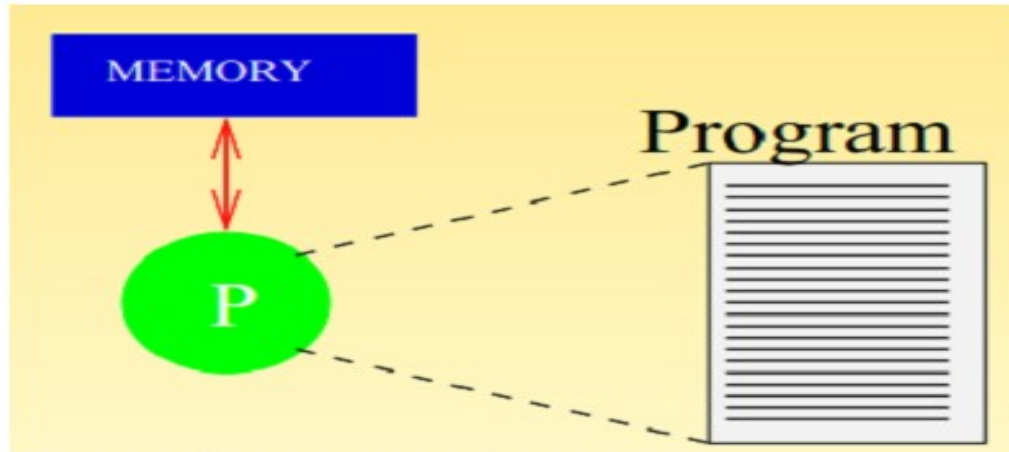
Hình 2.26. Chương trình được thiết kế song song

(Nguồn <https://computing.llnl.gov>)

Đối với chương trình được thiết kế song song, vấn đề lớn sẽ được chia ra thành nhiều vấn đề nhỏ và được xử lý song song bởi nhiều bộ xử lý.

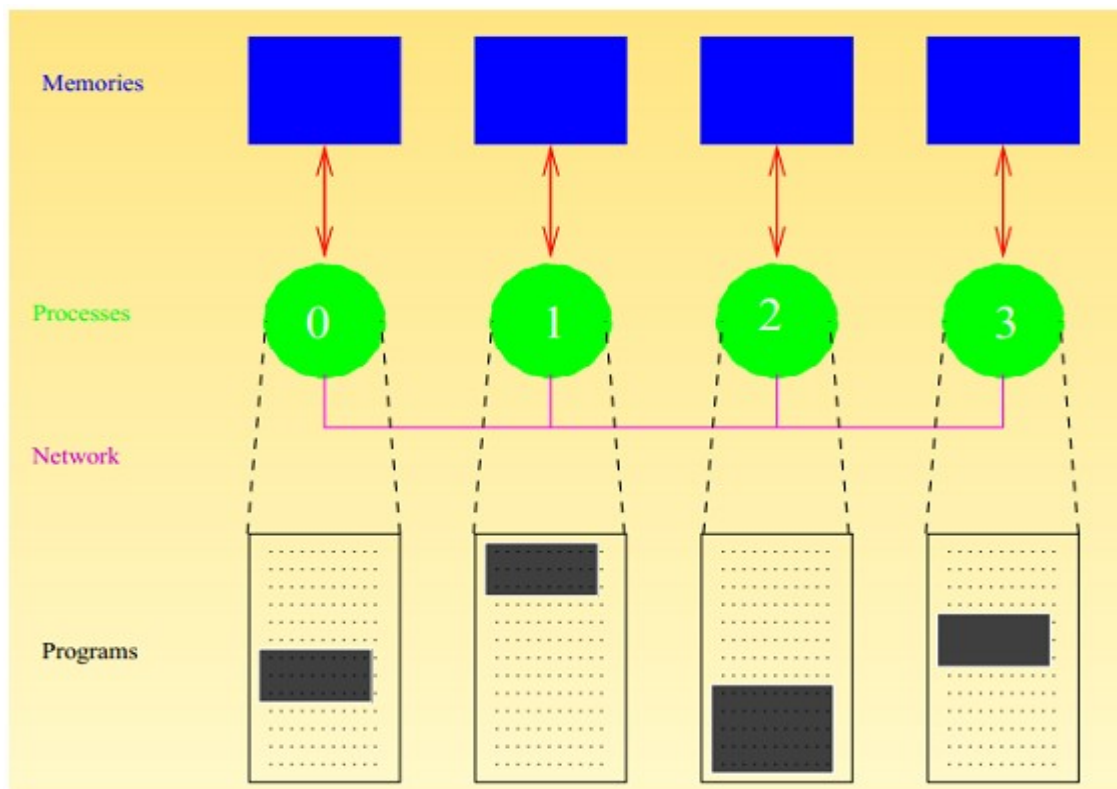
2.5.4. Giới thiệu giao diện lập trình MPI

MPI (Message Passing Interface) là chuẩn giao diện lập trình chuyển thông điệp. Với mục tiêu đặt ra là thực tế, hiệu quả và mềm dẻo, chuẩn giao diện lập trình MPI cung cấp giao diện lập trình song song trên hệ thống lưới.



Hình 2.27. Mô hình tuần tự

(Nguồn Tính toán lưới – Khoa CNTT & TT – Đại học Cần Thơ)



Hình 2.28. Mô hình chuyển thông điệp

(Nguồn Tính toán lưới – Khoa CNTT & TT – Đại học Cần Thơ)

2.5.5. Gói lập trình MPJ Express

MPJ Express là một thư viện lập trình của mpiJava 1.2 API được định nghĩa bởi diễn đàn Java Grande. MpiJava 1.2 API là một giao diện tương đương với MPI 1.1.

MPJ Express bao gồm các thư viện hỗ trợ lập trình song song dành cho ngôn ngữ Java và cả môi trường thực thi của nó. Dựa trên cơ chế chuyển thông điệp, MPJ Express cho phép thiết kế chương trình thực thi song song trên Clusters. MPJ Express còn hỗ trợ giao tiếp giữa các bộ xử lý độc lập của Clusters. MPJ Express được thiết kế cho mô hình lập trình bộ nhớ phân tán, tuy nhiên nó còn có thể hỗ trợ thực thi ứng dụng song song trên máy tính cá nhân mà không cần phải cài đặt Clusters.

Như đã đề cập, MPJ cung cấp 2 cách thiết kế chương trình song song đó là Multicore (trên một máy tính) và Clusters (trên nhiều máy tính).

2.6. GIỚI THIỆU VỀ CỜ TƯỚNG

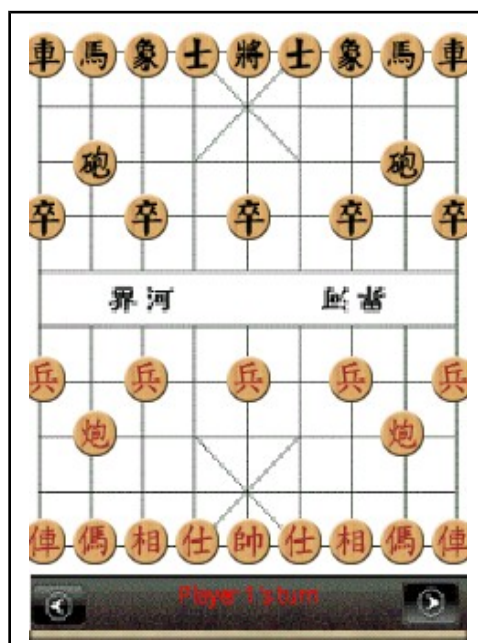
2.6.1. Nguồn gốc

Cờ tướng là loại cờ có từ khoảng thế kỷ VII. Cờ tướng được bắt nguồn từ Saturanga, một loại cờ cổ được phát minh ở Ấn Độ từ thế kỷ V đến thế kỷ VI (trước cờ tướng khoảng 200 năm). Saturanga được phát minh từ Ấn Độ, sau đó dần phát triển và phân hóa thành 2 loại cờ khác nhau. Đó là cờ vua khi đi về phía Tây (Châu Âu) và cờ tướng khi đi về phía Đông (Trung Quốc).



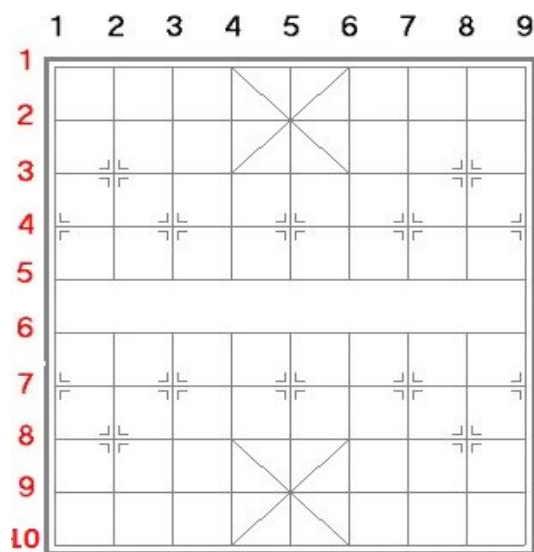
Hình 2.29. Hình ảnh chơi cờ
(Nguồn <http://google.com.vn>)

2.6.2. Bàn cờ



Hình 2.30. Bàn cờ tướng


Bàn cờ là một [hình chữ nhật](#) do 9 đường dọc và 10 đường ngang cắt nhau vuông góc tại 90 điểm hợp thành. Một khoảng trống gọi là sông nằm ngang giữa bàn cờ, chia bàn cờ thành hai phần đối xứng bằng nhau. Mỗi bên có một cung Tướng [hình vuông](#) do 4 ô hợp thành tại các đường dọc 4, 5, 6 kể từ đường ngang cuối của mỗi bên, trong 4 ô này có vẽ hai đường chéo xuyên qua.



Hình 2.31. Ma trận bàn cờ tướng

(Nguồn <http://google.com>)

Mỗi ván cờ lúc bắt đầu phải có đủ 32 quân, chia đều cho mỗi bên gồm 16 quân Trắng (Đỏ) và 16 quân Đen (Xanh). Quân cờ gồm 7 loại quân: Tướng, Sỹ, Tượng, Xe, Pháo, Mã, Tốt. Mỗi bên có thể viết khác nhau theo chữ hán nhưng giá trị và cách đi thì hoàn toàn giống nhau. Bảy loại quân có ký hiệu và số lượng cho mỗi bên như sau:

Tên quân cờ	Kí hiệu	Số lượng
Tướng	 	1
Sỹ	 	2
Tượng	 	2
Xe	 	2
Pháo	 	2
Mã	 	2
Tốt	 	5

Bảng 2.2. Các quân cờ tướng

2.6.3. Luật chơi

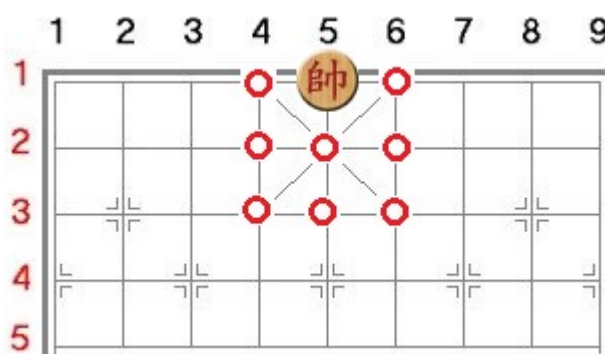
Ván cờ được tiến hành giữa hai người, một người cầm quân Trắng (Đỏ), một người cầm quân Đen (Xanh). Mục đích của mỗi người là tìm mọi cách đi quân trên bàn cờ theo đúng [luật](#) để chiếu bí Tướng của đối phương và giành thắng lợi.

Luật đi các quân cờ:

- Tướng:



Tướng chỉ được đi ngang hay đi dọc từng bước một trong phạm vi cung tướng, và không được đối mặt với quân tướng của đối phương khi không có quân nào ở chính giữa.

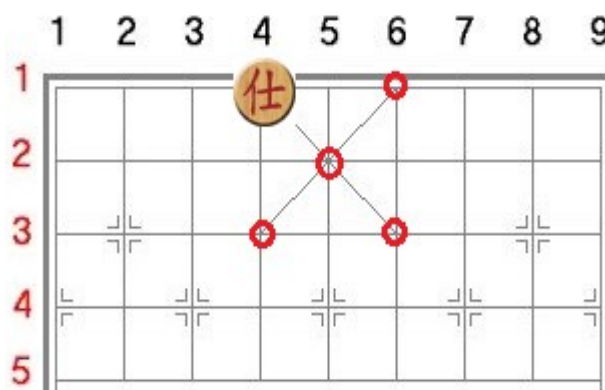


Hình 2.32. Các vị trí quân Tướng đi.

- Sĩ:



Quân Sĩ có vai trò bảo vệ cho Tướng. Quân Sĩ đứng ngay sát cạnh Tướng, chỉ đi từng bước một và đi theo đường chéo trong cung Tướng. Như vậy, quân Sĩ chỉ di chuyển đến được 5 điểm trên bàn cờ. Sĩ có chức năng trong việc bảo vệ Tướng, mất Sĩ được cho là nguy hiểm khi đối phương còn đủ 2 Xe hoặc dùng Xe, Mã, Tốt tấn công. Bỏ Pháo ăn Sĩ rồi dùng 2 Xe tấn công là chiến thuật thường thấy trong cờ tướng.

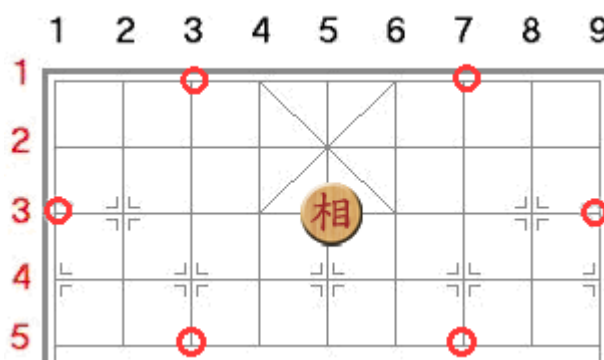


Hình 2.33. Các vị trí quân Sỹ đi.

- Tượng:

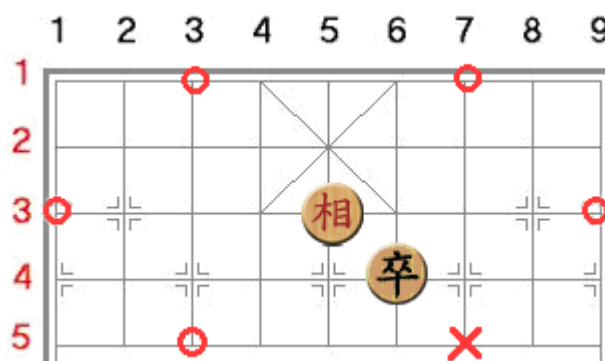


Quân Tượng đứng bên cạnh quân Sỹ và tương đương với Tượng trong cờ vua. Quân này đi theo đường chéo của hình vuông gồm 2 ô cờ. Quân Tượng không được qua sông và có nhiệm vụ cùng với quân Sỹ bảo vệ quân Tướng. Chỉ có 7 điểm mà Tượng có thể di chuyển tới và đứng ở đó.



Hình 2.34. Các vị trí quân Tượng đi.

Tượng sẽ không di chuyển được đến vị trí đã nêu nếu có 1 quân đặt tại vị trí giữa của hình vuông 2 ô. Khi đó ta gọi là Tượng bị cản và vị trí cản được gọi là “mắt Tượng”.

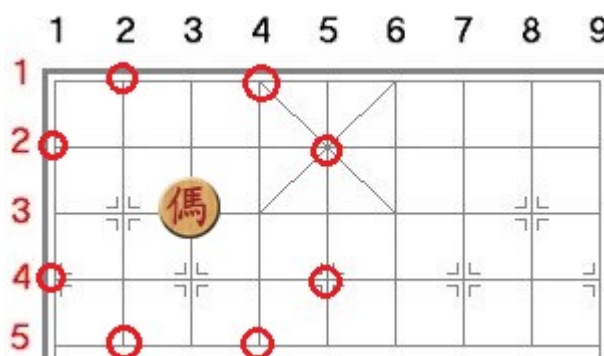


Hình 2.35. Các vị trí quân Tượng đi và trường hợp bị cản.

- Mã:

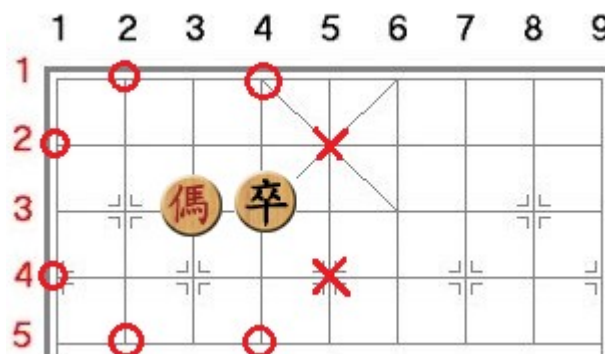


Quân Mã đứng bên quân Tượng. Quân này ăn tới vị trí chéo 2 ô so với nó.



Hình 2.36. Các vị trí quân Mã đi.

Quân Mã sẽ bị cản không di chuyển được đến vị trí đã nêu nếu có một quân cờ đặt nằm sát bên và gần vị trí mà quân Mã muốn tới.

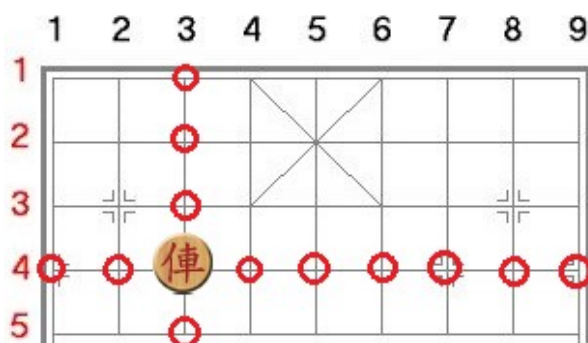


Hình 2.37. Các vị trí quân Mã đi và trường hợp bị cản.

- Xe:



Quân Xe đi và ăn theo một đường thẳng đứng hoặc ngang giống hệt quân Xe trong cờ vua. Quân Xe bắt đầu nước đi từ phía góc của bàn cờ và được xem là quân cờ mạnh nhất trong cờ tướng.

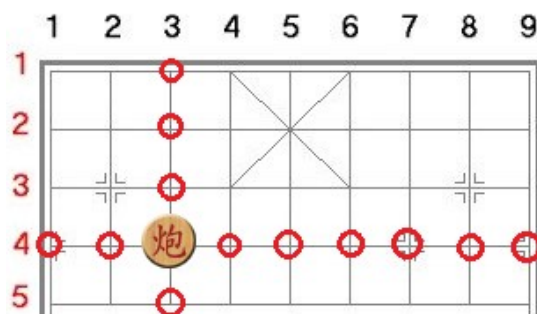


Hình 2.38. Các vị trí quân Xe đi.

- Pháo:

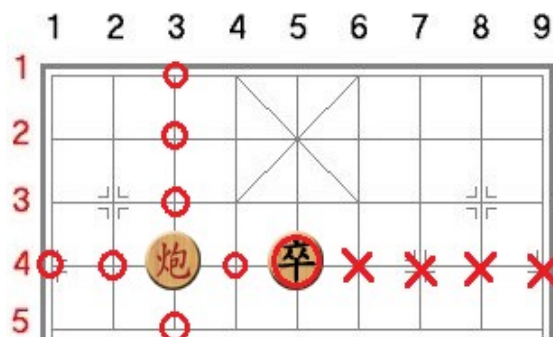


Quân Pháo có quy tắc đi giống quân Xe, theo chiều thẳng đứng hoặc ngang.



Hình 2.39. Các vị trí quân Pháo đi.

Tuy nhiên, cách ăn quân cờ của Pháo khác Xe ở chỗ là Pháo sẽ ăn quân cờ đối phương bằng cách nhảy qua 1 quân cờ khác.

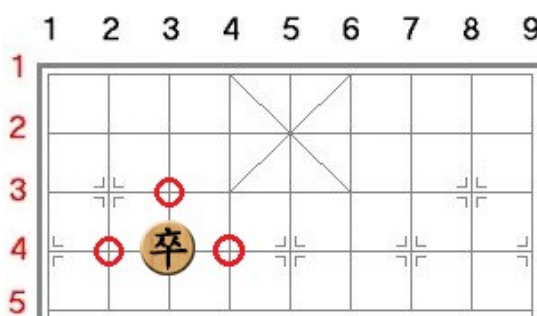


Hình 2.40. Các vị trí quân Pháo đi và ăn cờ đối thủ.

- Tốt:



Quân Tốt đi thẳng theo chiều đứng và có thể ăn quân từng bước một. Khi Tốt qua được sông, chúng có thể đi và ăn theo chiều ngang.



Hình 2.41. Các vị trí quân Tốt đi.

XH□□NT 3. NỘI DUNG – KẾT QUẢ NGHIÊN CỨU

Bằng cách vận dụng và kết hợp những kiến thức đã được trình bày trong chương 2, trong chương 3 này, tác giả sẽ vận dụng để xây dựng trò chơi cờ tướng online. Chi tiết việc xây dựng sẽ được đề cập đến bao gồm mô hình hệ thống, kiến trúc chương trình, cách thức xây dựng. Thêm vào đó, trong chương này sẽ trình bày chi tiết cấu trúc thông điệp giao tiếp giữa Client và Server. Việc thiết kế cơ sở dữ liệu cũng là một phần quan trọng trong đề tài. Cuối chương sẽ là phần giới thiệu chi tiết về các chức năng mà chương trình đã xây dựng được.

3.1. XÂY DỰNG TRÒ CHƠI CỜ TƯỚNG ONLINE

3.1.1. Đặc tả chi tiết đề tài

Trò chơi được xây dựng dành cho thiết bị có hệ điều hành Android phiên bản từ 2.2 Froyo trở lên (Demo trên Samsung Galaxy Y 5360 Android 2.3.6). Sau đó, người chơi phải cài chương trình ở phía Client vào thiết bị của mình.

Để có thể chơi cờ online thì máy của người chơi phải được kết nối Internet (wifi, 3G). Khi khởi động trò chơi, người dùng phải đăng ký tài khoản thông qua giao diện đăng ký. Khi đã có tài khoản người dùng sẽ đăng nhập vào Server. Nếu đăng nhập thành công, Server sẽ tạo ra 1 token tương ứng với người đăng nhập để quản lý cùng với đó là thông tin những ai đang đăng nhập.

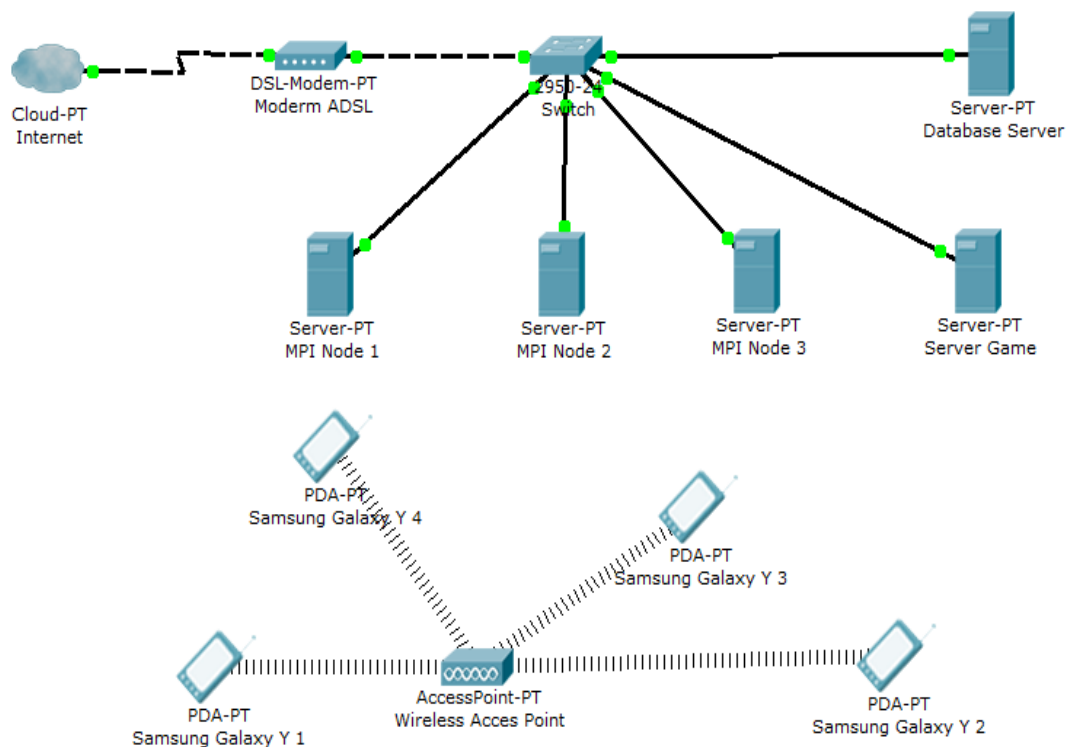
Dựa vào thông tin đó, người chơi có thể yêu cầu bất cứ người chơi nào khác đang đăng nhập để chơi cờ. Nếu người được yêu cầu đồng ý thì ván cờ được bắt đầu. Kể từ khi đăng nhập thành công, những thông tin được gửi từ Client đến Server sẽ gửi kèm theo token do Server tạo ra để đảm bảo thông tin gửi chính là do Client đó gửi. Để đảm bảo tính bảo mật, tất cả các thông tin dưới dạng plain text sẽ được mã hóa trước khi gửi lên đường truyền và giải mã khi nhận về bằng giải thuật mã hóa đối xứng. Khi ván cờ bắt đầu, người yêu cầu là người được đi trước. Hai người chơi lần lượt đi quân cờ của mình cho đến khi nào xác định được trạng thái thắng, thua hoặc hòa.

Khi người chơi đi quân cờ, Server sẽ nhận và xử lý bao gồm kiểm tra hợp lệ của nước đi, kiểm tra chiếu, xét thắng thua và gửi thông tin đến đối phương. Để tạo sự

công bằng và tránh phải chờ đợi trong khi chơi cờ, mỗi nước đi của mỗi quân cờ được quy định là 2 phút và tổng thời gian của mỗi người chơi ở một ván cờ là 20 phút. Nếu tới lượt người chơi nào mà người chơi đó trong vòng 2 phút không đi được nước cờ của mình hoặc hết tổng thời gian thì người đó sẽ thua và ván cờ sẽ kết thúc. Khi ván cờ kết thúc, hai người đang chơi có thể tiếp tục chơi hoặc thoát ra và tìm bạn chơi mới. Trong lúc chơi, người chơi có thể xin đi lại nếu đối thủ đồng ý, và cũng có thể cầu hòa hoặc xin thua. Nếu đang chơi mà người chơi thoát ra khỏi ván cờ thì sẽ thua ván cờ đó. Mỗi ván cờ thắng sẽ được cộng 50 điểm và thua sẽ bị trừ 50 điểm. Chương trình sẽ cập nhật lại số điểm và cấp bậc của người chơi khi ván cờ kết thúc.

Bên cạnh việc chơi cờ, để tạo hứng thú cho người chơi, chương trình còn cung cấp thêm tính năng chat giữa hai người chơi đang với nhau.

3.1.2. Kiến trúc chương trình



Hình 3.1. Kiến trúc chương trình

Từ việc phân tích và làm rõ nội dung đề tài, kiến trúc chương trình gồm có 2 phần đó chủ yếu đó là Client và Server.

3.1.3. Server

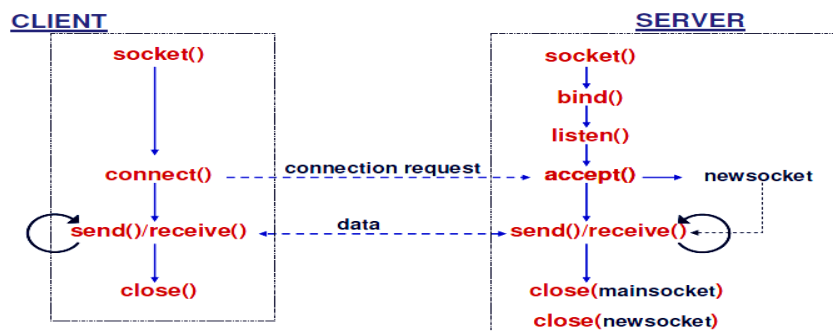
Server chịu trách nhiệm xử lý thông điệp gửi từ Client, tính toán và lưu trữ cơ sở dữ liệu. Một cách cụ thể:

- Database Server: lưu trữ cơ sở dữ liệu của người dùng bao gồm thông tin đăng ký, đăng nhập, thông tin người chơi như điểm, cấp bậc, bàn cờ.
- Game Server: đóng vai trò là nơi trung gian để giao tiếp giữa Client với nhau, quản lý kết nối, đồng bộ thời gian, nhận thông tin từ Client, tính toán nước đi hợp lệ, thông tin thắng thua và gửi thông tin đến Client. Game Server sẽ áp dụng kỹ thuật lập trình song song MPI để tính toán giải thuật thắng thua, kiểm tra chiếu tướng.
- MPI Node là nơi giao tiếp với Game Server để tính toán giải thuật thắng thua, kiểm tra chiếu tướng.

3.1.4. Client

Client đóng vai trò là người chơi. Cụ thể Client ở đây là sử dụng điện thoại Samsung Galaxy Y 5360. Client muốn chơi cờ bắt buộc phải có kết nối Internet thông qua Wifi hoặc 3G.

- Client có thể đăng ký, đăng nhập và tiến hành chơi cờ.
- Nhận thông tin cập nhật người chơi đang online được gửi từ Server.
- Gửi thông tin nước đi cho Game Server.
- Chờ nhận thông tin gửi từ Game Server và cập nhật lại đường đi cũng như giao diện đồ họa.
- Nhận kết quả thắng thua và hiển thị kết quả cho người chơi.



Hình 3.2. Sơ đồ nối kết Client và Server

3.1.5. Xây dựng bàn cờ

Áp dụng giải thuật, bàn cờ sẽ được chuyển sang ma trận 9x10. Mỗi quân cờ sẽ có một giá trị khác nhau trên bàn cờ, giá trị của quân cờ chính là giá trị của các phần tử trong ma trận.

Tên quân cờ	Client 1	Client 2
Tướng	5	-5
Sỹ	4	-4
Tượng	3	-3
Mã	2	-2
Xe	1	-1
Pháo	6	-6
Tốt	7	-7

Bảng 3.1. Giá trị các quân cờ

Vị trí nào trên bàn cờ không có quân cờ sẽ mang giá trị 0. Ví dụ, bàn cờ tướng lúc đầu sẽ được chuyển sang ma trận A[9][10] như sau:

1	2	3	4	5	4	3	2	1
0	0	0	0	0	0	0	0	0
0	6	0	0	0	0	0	6	0
7	0	7	0	7	0	7	0	7
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
- 7	0	- 7	0	- 7	0	- 7	0	- 7
0	- 6	0	0	0	0	0	-6	0
0	0	0	0	0	0	0	0	0
- 1	- 2	- 3	- 4	- 5	- 4	- 3	- 2	- 1

3.1.6. Giải thuật

Giải thuật tính toán cho ván cờ được thiết kế ở cả Client và Server. Phía Client được xét cho ván cờ chơi offline, phía Server được áp dụng trong các ván cờ online.

a. Khởi tạo bàn cờ:

Bàn cờ được khởi tạo bởi một mảng số nguyên 2 chiều 9 x 10. Các quân cờ mang giá trị tuyệt đối như sau: quân Xe – 1, quân Mã – 2, quân Tượng – 3, quân Sĩ - 4, quân Tướng – 5, quân Pháo – 6, quân Tốt – 7. Quân cờ bên thứ 1 sẽ mang giá trị dương và bên thứ 2 mang giá trị âm. Những ô không có cờ giá trị sẽ là 0.

```
void KhoiTaoBanCo() {  
    khai báo ma trận bàn cờ 9 x 10;  
    khởi tạo giá trị 0 cho tất cả các ô;  
    khai báo giá trị quân cờ ở các ô tương ứng;  
}
```

b. Xây dựng hàm có thể đi của các quân cờ

- Quân Xe: vị trí mà quân Xe đi tới hoặc ăn con cờ khác của đối thủ phải nằm tại vị trí thẳng hàng với nó (theo hàng dọc và hàng ngang) và không bị con cờ nào cản so với nó.

```
//Vị trí Xe đi được
boolean XeDi(int x1, int y1, int x2, int y2){
    if (vị trí không thay đổi) return false;
    if (vị trí mới không cùng nằm trên đường thẳng theo chiều
    ngang hoặc dọc) return false;
    if (vị trí mới cùng nằm trên đường thẳng theo chiều dọc){
        for(trên đường đi từ vị trí cũ đến vị trí mới){
            if(có quân cản) return false;
        }
    }
    if (vị trí mới cùng nằm trên đường thẳng theo chiều ngang){
        for (duyệt trên đường từ vị trí cũ đến vị trí mới){
            if (có quân cản) return false;
        }
    }
    return true; // Đúng ở các trường hợp còn lại
}
```

➤ **Quân Mã:** vị trí quân Mã đi và ăn là chéo 2 ô và không bị cản so với nó.

```
boolean MaDi(int x1, int y1, int x2, int y2){
    tính khoảng cách x của 2 vị trí;
    tính khoảng cách y của 2 vị trí;
    tính giá trị trung bình của x;
    tính giá trị trung bình của y;
    if (khoảng cách x bằng 0 hoặc khoảng cách y bằng 0 hoặc tổng
    khoảng cách x và y khác 3) return false; //Sai khi đi thẳng hàng và
    tổng khoảng cách khác 3
    if (đi theo chiều ngang và không bị cản theo chiều ngang)
    return true;
    if (đi theo chiều dọc và không bị cản theo chiều dọc)
    return true;
    return false;
}
```

- Quân Tượng: vị trí quân Tượng đi và ăn là chéo 4 ô và không bị cản so với nó.

```
boolean TuongDi(int x1, int y1, int x2, int y2){
    tính khoảng cách x của 2 vị trí;
    tính khoảng cách y của 2 vị trí;
    tính giá trị trung bình của x;
    tính giá trị trung bình của y;
    if (khoảng cách x khác 2 hoặc khoảng cách y khác 2) return
false; //Tượng phải đi chéo 2 ô
    if (có quân cản) return false;
    if(qua sông) return false;
    return true; // Đúng ở các trường hợp còn lại
}
```

- Quân Sĩ: vị trí quân Sĩ đi và ăn là chéo 1 ô và trong phạm vi của 2 bên (vị trí x từ 4 đến 6, bên 1 $y \leq 3$ và bên 2 $y \geq 8$).

```
Boolean SyDi(int x1, int y1, int X2, int Y2){
    int dx = |x1-x2|; // khoảng cách x của 2 vị trí
    int dy = |y1-y2|; // khoảng cách y của 2 vị trí

    if (x2<4 or x2>6) return false;
    if (y2>3 and y2<8) return false;
    // Sai nếu vị trí đến nằm ngoài giới hạn của nó
    if (dx!=1 || dy!=1) return false;
    // Sai nếu vị trí đến không phải chéo trong cùng ô

    return true; // Đúng ở các trường hợp còn lại
}
```

- Quân Tướng: vị trí quân Tướng đi và ăn là 1 ô (theo x hoặc y) mỗi quân Tướng chỉ đi trong giới hạn của mỗi bên (vị trí x từ 4 đến 6, bên 1 y ≤ 3 và bên 2 y ≥ 8).

```
boolean Tuong_Di(int x1, int y1, int x2, int y2){
    tính khoảng cách x của 2 vị trí;
    tính khoảng cách y của 2 vị trí;
    if(vị trí mới nằm ngoài cung tướng) return false; //Tướng
    không được đi ra ngoài cung tướng
    if (tổng khoảng cách x và y khác 1) return false; //Tướng
    phải đi thẳng 1 ô theo chiều ngang hoặc dọc
    return true; // Đúng ở các trường hợp còn lại
}
```

- Quân Pháo: vị trí quân Pháo đi giống như quân Xe – chỉ đi thẳng, còn vị trí quân Pháo ăn phải có 1 quân cờ ở giữa nó với vị trí quân cờ muốn ăn.

```
// Vị trí Pháo đi được
boolean PhaoDi(int x1, int y1, int x2, int y2){
    tính khoảng cách x của 2 vị trí;
    tính khoảng cách y của 2 vị trí;
    if(trường hợp pháo đi){
        đi như xe;
    }
    if(trường hợp pháo ăn){
        if(kiểm tra trên đường đi có đúng 1 quân cờ) return
    true;
    }
    return false; //Các trường hợp còn lại
}
```

- Quân Tốt: vị trí quân Tốt đi và ăn là chỉ cách nó 1 ô. Quân Tốt chỉ đi thẳng khi chưa qua sông và có thể đi thẳng, sang trái hoặc sang phải nếu đã qua sông.

```
boolean TotDi(int x1, int y1, int X2, int Y2){
    tính khoảng cách 2 vị trí;
    tính khoảng cách x của 2 vị trí;
    tính khoảng cách y của 2 vị trí;
    // Đi thẳng khi chưa qua sông, thẳng trái phải khi qua sông
    if (khoảng cách 2 vị trí khác 1) return false;//Tốt chỉ đi 1 ô
    // Nếu đi thẳng, trái và phải
    if (đi thẳng hoặc sang trái hoặc sang phải) {
        if(đi thẳng) return true;
        if((đi qua trái hoặc phải)&&(đã qua sông)) return true;
    }
    return false; // Sai các trường hợp còn lại
}
```

- Xét tổng hợp vị trí quân cờ có thể đi:

```
boolean CoTheDi(int x,int y,int x2, int y2){
    boolean ok = false;
    if (vị trí cũ và vị trí mới là hai quân cờ cùng 1 bên)
return false;
    switch(giá trị tuyệt đối con cờ){
        //Kiểm tra quân cờ muốn đi là quân cờ nào
        case 1:ok=this.XeDi(x1, y1, x2, y2); break;
        case 2:ok=this.NguaDi(x1, y1, x2, y2); break;
        case 3:ok=this.TuongDi(x1, y1, x2, y2); break;
        case 4:ok=this.SyDi(x1, y1, x2, y2); break;
        case 5:ok=this.Tuong_Di(x1, y1, x2,y2);break;
        case 6:ok=this.PhaoDi(x1, y1, x2, y2); break;
        case 7:ok=this.TotDi(x1, y1, x2, y2); break;
    }
    return ok; //Return có thể đi được hay không
}
```

- Xét chiếu: nếu bất kỳ quân cờ nào có thể đến tướng của quân cờ đối phương thì quân cờ đối phương bị chiếu.

```
boolean chieu(int x,int y){
    if (2 tướng đối mặt nhau) return true; // 2 tướng đối diện
    nhau
    for (chiều ngang bàn cờ){
        for (chiều dọc bàn cờ){//duyet qua tất cả con cờ của
            mình
                if (con cờ của mình có thể đến quân tướng của đối
                    phương)
                        return true;//Chiếu
        }
    }
    return false;//Không chiếu
}
```

- Xét nước cờ hợp lệ: nước cờ được xem là hợp lệ nếu quân cờ đang xét có thể đi đến vị trí đó mà không làm cho quân Tướng của nó bị chiếu.

```
boolean HopLe(int x1, int y1, int x2, int y2){
    đi thử từ vị trí x1 y1 đến vị trí x2 y2;
    for (chiều ngang bàn cờ){
        for (chiều dọc bàn cờ){
            if(quân Tướng của mình bị chiếu) return
            false;//Nước đi không hợp lệ
        }
    }
    dôi lại nước đi thử;
    return true; //Nước đi hợp lệ
}
```


- Xét thắng thua: nếu tất cả quân cờ của đối thủ không có quân cờ nào có thể đi được nữa, có nghĩa là không có quân cờ nào có nước đi hợp lệ thì bên đó thua.

```
boolean ThangThua(int x,int y){  
    for(chiều ngang bàn cờ){  
        for(chiều dọc bàn cờ){  
            if(quân cờ của đối thủ có nước đi hợp lệ) return  
            false;  
        }  
    }  
    return true;  
}
```

3.2. XÂY DỰNG SERVER

3.2.1. Chức năng

Server được thiết kế để quản lý và là nơi trung gian cho việc giao tiếp giữa các Client. Chức năng Server được xây dựng bao gồm:

- Tiếp nhận nối kết đến từ Client.
- Xử lý đăng ký, đăng nhập.
- Quản lý danh sách người chơi.
- Quản lý ván cờ.
- Xử lý song song và tính toán giải thuật cho các ván cờ.
- Lưu trữ cơ sở dữ liệu người chơi và ván cờ.
- Đồng bộ hóa thời gian.
- Cung cấp chức năng chat.

3.2.2. Các thành phần của Server

Server bao gồm các thành phần Database Server, Game Server, MPI node được nối kết thông qua mạng máy tính để có thể giao tiếp với nhau:

- Database Server: lưu trữ cơ sở dữ liệu (CSDL) của người dùng và ván cờ. Database Server sử dụng hệ quản trị cơ sở dữ liệu (HQTCSDL) MySql.
- Game Server: đóng vai trò là nơi trung gian để các Client giao tiếp. Có thể nói, Game Server đóng vai trò quan trọng quan trọng nhất trong hệ thống. Ngoài việc giao tiếp với Client, nó còn là nơi giao tiếp với các Server khác như Database Server để truy xuất dữ liệu và MPI node để tính toán giải thuật song song. Game Server sử dụng cơ chế lập trình Socket TCP kết hợp với lập trình đa luồng để xử lý yêu cầu của Client. Bên cạnh đó, Game Server sẽ sử dụng giao diện lập trình song song MPI cụ thể là gói thư viện hỗ trợ MPJ Express để xây dựng giải thuật tính toán thắng thua, xét chiếu tướng.

3.2.3. Hướng giải quyết

a. Game Server

Trong quá trình nghiên cứu và thực hiện, để xây dựng Game Server phục vụ Client nối kết, ngoài việc sử dụng Socket, vẫn còn một số kỹ thuật khác như RMI, WebServices vẫn đáp ứng được việc xây dựng Game Server. Tuy nhiên, so về tốc độ truyền dữ liệu, Socket là nhanh hơn các kỹ thuật còn lại. Thêm vào đó, Server cần phải quản lý người chơi một cách trực tiếp thông qua việc kiểm tra nối kết đó. Một phần nữa là trò chơi cần phải đáp ứng nhanh về tốc độ chương trình và hạn chế tối đa độ trễ. Từ việc đánh giá, so sánh các yếu tố, tác giả đã lựa chọn kỹ thuật lập trình Socket áp dụng vào xây dựng Game Server. Ngay cả khi xác định được kỹ thuật Socket thì vẫn còn tồn tại vấn đề, nếu xét về tốc độ, thì UDP sẽ là phương án lựa chọn đầu tiên. Tuy nhiên, cần phải xét thêm yếu tố về độ chính xác, thì Socket TCP đáp ứng tốt hơn so với UDP nên kỹ thuật Socket được áp dụng ở đây là Socket TCP.

- Server mở 1 ServerSocket duy nhất là ở cổng 20000.
- Server sẽ có 1 luồng chính luôn luôn lắng nghe nối kết từ Client vào.
- Mỗi khi có nối kết đến, Server sẽ Settimeout Socket là 10 phút, sau đó Server sẽ tạo 1 luồng khác để phục vụ cho Client.
- Mỗi Client chỉ có 1 luồng phục vụ nó, luồng này dùng để đăng ký, đăng nhập, đánh cờ, xin đi lại, chat, yêu cầu, cầu hòa, xin thua.
- Server sẽ sử dụng ArrayList để lưu người chơi đang online, lưu bàn cờ đang chơi.
- Khi Client đăng nhập thành công, Server sẽ tạo đối tượng người chơi và cập nhật vào danh sách người chơi.
- Để đảm bảo thông tin được nhận đúng từ người chơi, Server sẽ có cơ chế token để kiểm tra tính hợp lệ của Client bằng cách tạo ra token mỗi khi Client đăng nhập thành công và gửi về cho Client. Token là chuỗi được hệ thống quy định, có thể là chuỗi ngẫu nhiên. Trong đề tài này, token mặc định là sự kết hợp của chuỗi String **Socket** của Client + thời gian đăng nhập thành công của Client vào hệ thống. Kể từ khi token được tạo ra, mọi thông tin được gửi lên từ Client phải

kèm theo token để đảm bảo thông tin đó do ai gửi. Phía Server sẽ phân tích token để kiểm tra tính hợp lệ của Client.

- Khi có yêu cầu chơi từ một Client nào đó, và người chơi được yêu cầu chấp nhận thì ván cờ mới bắt đầu. Server sẽ tạo ra 1 bàn cờ mới rồi thêm vào danh sách bàn cờ. Kèm theo đó là tạo luồng đếm thời gian nước đi cho Client, mỗi nước đi là 2 phút., tổng thời gian mỗi bên là 20 phút. Nếu hết thời gian đi của mỗi lượt mà không đi hoặc hết tổng thời gian bên nào thì bên đó thua, và bàn cờ kết thúc.
- Trong khi đánh cờ có thể cầu hòa, xin thua hoặc đánh tới khi kết thúc ván cờ. Nếu phân định trạng thái thắng thua, thì Server làm công việc sau: Lưu trạng thái bàn cờ vào cơ sở dữ liệu, hủy luồng đếm thời gian, xóa bàn cờ trong danh sách bàn cờ. Ngoài ra, Server còn cập nhật lại thông tin người chơi. Ví dụ, khi chơi xong thì trạng thái của người chơi sẽ trở về trạng thái rảnh, điểm số tăng lên hoặc hạ xuống. Sau đó, Server sẽ gửi thông tin useronline lại để thông báo cập nhật điểm mới cho Client.

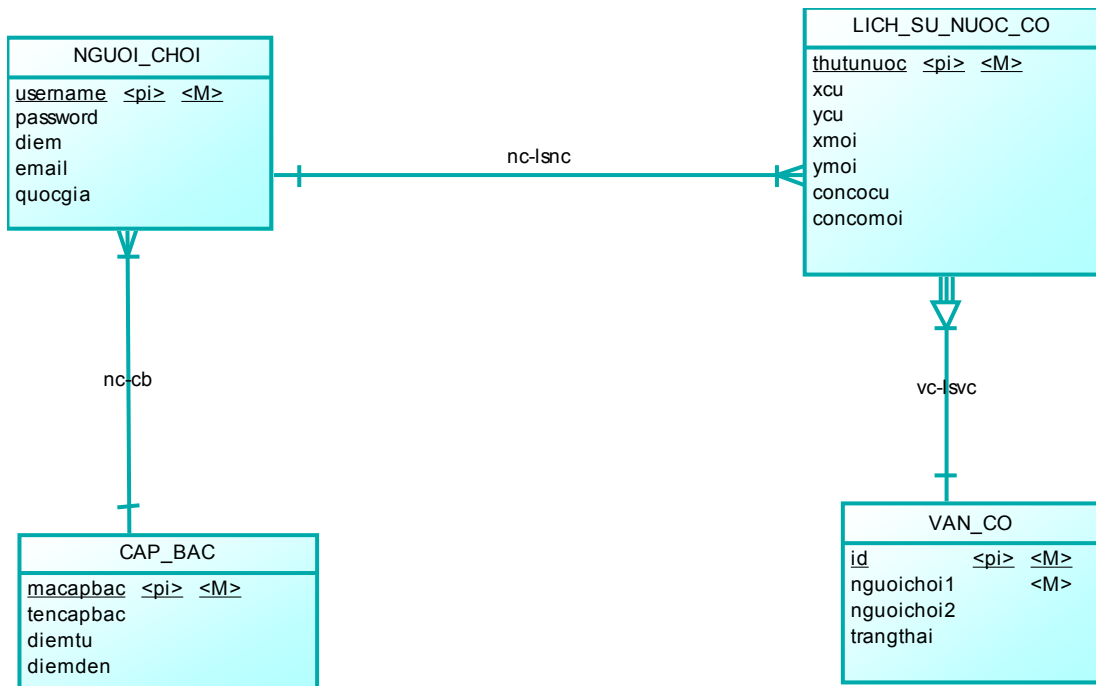
b. Database Server

Cơ sở dữ liệu là một phần rất quan trọng trong đề tài. Với mục đích quản lý người chơi, giúp người chơi có thể xem lại thành tích cá nhân, quản lý ván cờ, lưu lại lịch sử ván cờ giúp người chơi có thể xin đi lại, tất cả thông tin người chơi, bàn cờ sẽ được lưu ở cơ sở dữ liệu. Trong đề tài này, tác giả sử dụng hệ quản trị cơ sở dữ liệu Mysql lưu trữ thông tin. Mysql là phần mềm mã nguồn mở, dùng cho việc lưu trữ cơ sở dữ liệu vừa và nhỏ nên rất thích hợp đối với đề tài.

Để đáp ứng được yêu cầu của đề tài, cơ sở dữ liệu được thiết kế gồm 4 bảng:

- **NGUOI_CHOI**: lưu thông tin cá nhân của người chơi.
- **CAP_BAC**: các cấp độ của người chơi, mỗi cấp bậc được phân chia theo các khoảng điểm khác nhau.
- **VAN_CO**: quản lý các ván cờ đã được tạo, đánh dấu lịch sử và trạng thái của các ván cờ.
- **LICH_SU_VAN_CO**: lịch sử của từng ván cờ, lưu tất cả các nước đi trong một ván cờ.

❖ **Mô hình dữ liệu mức quan niệm CDM:**



Hình 3.3. Mô hình dữ liệu mức quan niệm CDM

❖ **Mô hình dữ liệu mức luận lý LDM:**

- **NGUOI_CHOI** (username, password, diem, email, quocgia, macapbac)
- **VAN_CO** (id, nguoichoi1, nguoichoi2, trangthai)
- **CAP_BAC** (macapbac, tencapbac, diemtu, diemden)
- **LICH_SU_VAN_CO** (id, thutunuoc, username, xcu, ycu, xmoi, ymoi, concocu, concomoi)

❖ **Các ràng buộc tham chiếu:**

- **NGUOI_CHOI** (macapbac) → **CAP_BAC** (macapbac)
- **LICH_SU_VAN_CO** (id) → **VAN_CO** (id)
- **LICH_SU_VAN_CO** (username) → **NGUOI_CHOI** (username)

❖ Mô hình dữ liệu mức vật lý PDM:

BẢNG 3.2: NGUOI_CHOI

TT	Tên	Kiểu	Kích thước	Khóa chính	Duy nhất	Not Null	RBTV Khóa Ngoại	Diễn Giải
1	username	varchar	20	✓	✓	✓		Tên đăng ký
2	password	varchar	40			✓		Mật khẩu
3	diem	integer	11			✓		Điểm chơi
4	email	varchar	50		✓	✓		Địa chỉ email
5	quocgia	varchar	20					Quốc gia
6	macapbac	varchar	20				✓	Mã cấp bậc

BẢNG 3.3: CAP_BAC

TT	Tên	Kiểu	Kích thước	Khóa chính	Duy nhất	Not Null	RBTV Khóa Ngoại	Diễn Giải
1	macapbac	varchar	20	✓	✓	✓		Mã cấp bậc
2	tencapbac	varchar	20					Tên cấp bậc
3	diemtu	integer	11			✓		Điểm từ
4	diemden	integer	11			✓		Điểm đến

BẢNG 3.4: VAN_CO

TT	Tên	Kiểu	Kích thước	Khóa chính	Duy nhất	Not Null	RBTV Khóa Ngoại	Diễn Giải
----	-----	------	------------	------------	----------	----------	--------------------	-----------

1	id	integer	11	✓	✓	✓		Mã ván cờ
2	nguoichoi1	varchar	20			✓		Người chơi 1
3	nguoichoi2	varchar	20			✓		Người chơi 2
4	trangthai	integer	11					Trạng thái ván cờ

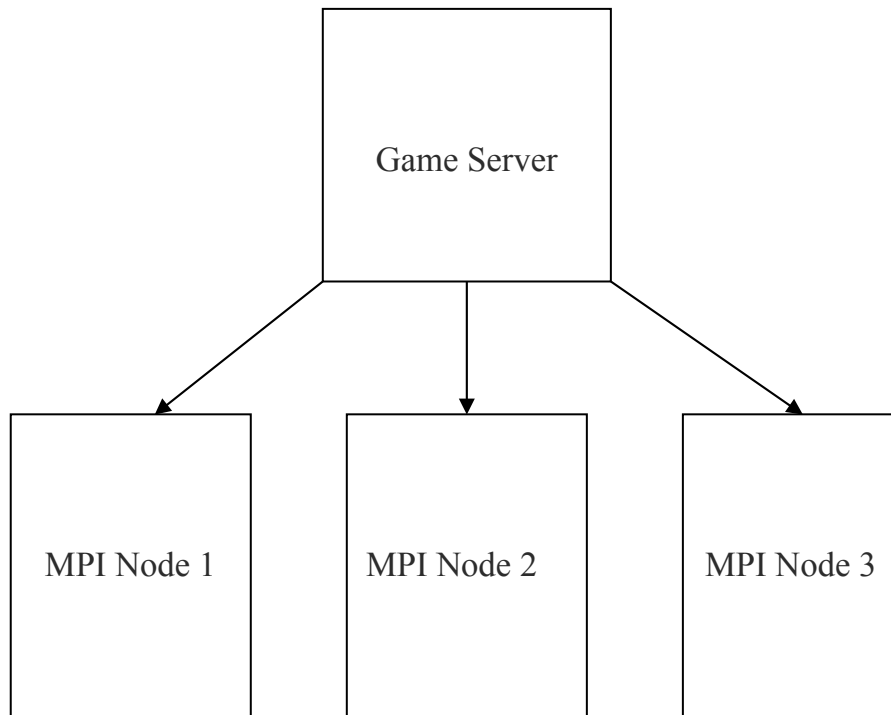
BẢNG 3.5: LICH_SU_VAN_CO

TT	Tên	Kiểu	Kích thước	Khóa chính	Duy nhất	Not Null	RBTV Khóa Ngoại	Diễn Giải
1	id	integer	11	✓		✓	✓	Mã ván cờ
2	thutunuoc	integer	11	✓		✓		Thứ tự nước đi
3	xcu	integer	11			✓		Vị trí X cũ
4	ycu	integer	11		✓	✓		Vị trí Y cũ
5	xmoi	integer	11			✓		Vị trí X mới
6	ymoi	integer	11			✓		Vị trí Y mới
7	concocu	integer	11			✓		Giá trị con cờ cũ
8	concomoi	integer	11			✓		Giá trị con cờ mới
9	username	varchar	20			✓	✓	Người đi

c. MPI Node

MPI Node đóng vai trò tạo thành một nhóm máy tính hay còn gọi là Clusters. MPI node là nơi tính toán giải thuật thắng thua ván cờ và kiểm tra chiếu. Được nối kết với Game Server thông qua mạng và sử dụng giao diện lập trình song song MPI với sự hỗ trợ của thư viện MPJ Express, MPI Node có thể nhận yêu cầu từ Game Server, tính toán và trả kết quả về Game Server thông qua cơ chế truyền thông điệp (Message Passing).

Tập hợp các MPI Node và Game Server sẽ tạo thành một Clusters tính toán song song. Trong đề tài, chương trình phía Server sẽ được phân chia thành 12 tiến trình con. Mỗi tiến trình sẽ có nhiệm vụ riêng để đảm bảo tận dụng được tài nguyên của hệ thống và tăng tốc độ tính toán của chương trình.



Hình 3.4. Mô hình giao tiếp giữa Game Server và MPI node

Chương trình được thiết kế ở Game Server, các công việc liên quan tính toán thắng thua và kiểm tra chiếu sẽ được đẩy xuống các node để thực thi. Các tiến trình sẽ thực thi song song. Do đó, tốc độ thực thi chương trình sẽ nhanh hơn chương trình được thiết kế như thông thường.

Cụ thể các tiến trình được thực thi:

- Tiến trình 0: thực thi ở Game Server, đóng vai trò trung gian, là nơi tiếp nhận và xử lý các yêu cầu từ Client. Thêm vào đó, tiến trình 0 còn truy xuất dữ liệu từ Database Server.
- Tiến trình từ 1 đến 9: tính toán thắng thua.
- Tiến trình 10: kiểm tra chiếu tương đối với người chơi 1.
- Tiến trình 11: kiểm tra chiếu tương đối với người chơi 2.

3.3. XÂY DỰNG CLIENT

3.3.1. Chức năng

Client đóng vai trò là người chơi, cụ thể là thiết bị chạy hệ điều hành Android. Ở phía Client, tác giả sẽ xây dựng chương trình trò chơi cờ tướng online trên hệ điều hành Android. Công việc cụ thể bao gồm:

- Viết một chương trình phần mềm cài đặt trên điện thoại Android.
- Ứng dụng có thể chơi offline và chơi online nếu có kết nối mạng thông qua 3G hoặc Wifi.
- Chương trình gồm các chức năng cơ bản: cài đặt, hướng dẫn, chơi mới hoặc kết thúc.
- Ngoài ra chương trình còn các chức năng của trò chơi online: đăng ký tài khoản, đăng nhập hệ thống, yêu cầu chơi cờ, chơi cờ trực tuyến, xin đi lại và chat với đối thủ trong khi chơi cờ.
- Trong khi chơi cờ, người chơi có thể xin thua, cầu hòa hoặc xin đi lại nếu được sự chấp nhận của đối thủ.

3.3.2. Vấn đề đặt ra

- Xây dựng giao diện tích hợp trên các loại màn hình khác nhau.
- Xây dựng các chức năng đáp ứng cho trò chơi.
- Xây dựng giải thuật kiểm tra nước đi hợp lệ của quân cờ, xét chiếu.
- Xây dựng giải thuật kiểm tra thắng thua.
- Xây dựng giao thức TCP Socket nối kết đến Server để chơi cờ và chat trực tuyến.
- Kiểm thử chương trình dành cho Client.

3.3.3. Hướng giải quyết

Ở phía Client, tác giả vận dụng kỹ thuật lập trình ứng dụng trên hệ điều hành Android kết hợp lập trình mạng Socket TCP, lập trình luồng để xây dựng chương trình phía Client:

- Thiết kế Service trên Android, được start() khi ứng dụng được mở. Service là chương trình chạy ngầm bên dưới, điều khiển giao diện màn hình (các Activity). Service có vai trò kết nối đến Server và tạo luồng luôn lắng nghe kết quả trả về từ Server và thông báo đến các Activity.
- Đặt thời gian Timeout cho mỗi Socket là 10 phút.
- Thiết kế Interface gồm tất cả các hàm mà tương tác với Service theo mẫu thiết kế (Design Pattern).
- Mã hóa các thông điệp trước khi gửi đi và giải mã khi nhận về.
- Thiết kế giao diện được tính theo tỉ lệ màn hình với đơn vị 1 ô là dp (với độ phân giải 160 px/inch thì 1 dp = 1 px).
- Bắt các sự kiện chạm màn hình theo chức năng của trò chơi.

3.3.4. Các thành phần của chương trình

a. Cấu trúc thư mục

Thành phần của chương trình gồm 6 package chứa các lớp của chương trình:

- Packet **chess** gồm các lớp:
 - + Class Chat: thể hiện giao diện khung chat của ván cờ.
 - + Class Profile: lớp lưu thông tin cá nhân và bạn chơi.
 - + Class Sound: lớp các hàm điều khiển âm thanh của chương trình.
- Packet **chess2user** gồm các lớp:
 - + Class ChessActivityOffline: lớp điều khiển giao diện bàn cờ chơi offline.
 - + Class ChessViewOffline: lớp giao diện bàn cờ offline.
 - + Class Chess: lớp khởi tạo bàn cờ offline.
- Packet **chessonline** gồm các lớp:
 - + Class ChessActivityOnline: lớp điều khiển giao diện bàn cờ chơi online.
 - + Class ChessViewOnline: lớp giao diện bàn cờ online.
 - + Class ChessOnline: lớp khởi tạo bàn cờ online.
- Packet **chinesechess** gồm các lớp:
 - + Interface ApplicationListener: gồm các hàm cập nhật giao diện.
 - + Class ChineseChessApplication: lớp ứng dụng truyền thông điệp từ Service đến các Activity.

+ Class ChineseChessService: dịch vụ chạy ngầm điều khiển giao diện chương trình.

- Packet **function** gồm các lớp:

+ Class Funtion: lớp các hàm thoát chương trình.

+ Class GetString: lớp các hàm tách chuỗi.

+ Class ShowDialog: lớp thừa kế dialog, dùng để xuất ra thông báo cho người dùng.

- Packet **interface** gồm các lớp:

+ Class Chose_Country: lớp thể hiện danh sách các quốc gia trong chức năng đăng kí tài khoản.

+ Class ListUser: lớp khởi tạo danh sách các người chơi đang online.

+ Class ListUserOnline: lớp cập nhật danh sách người chơi online.

+ Class MainChess: lớp menu chính của chương trình.

+ Class Rule: lớp thể hiện trợ giúp cho người chơi.

+ Class Setting: lớp chức năng cài đặt chương trình.

+ Class Signin: lớp chức năng đăng nhập.

+ Class Signup: lớp chức năng đăng kí.

b. Layout

Gồm 9 file xml giao diện của chương trình. Các file xml này được gọi trong các lớp Activity. Mỗi file chứa các đối tượng Button, Edittext, ImageView, Spinner được xử lý các sự kiện hoặc thay đổi nội dung, tính năng tại các Activity gọi tới nó.

c. Menu

Gồm 5 file xml menu của các giao diện chương trình. Các file xml này được gọi trong các lớp Activity. Chức năng của các file này là giúp tạo thuận tiện cho người sử dụng muốn thay đổi âm thanh hoặc chuyển qua khung chat mà không phải trở ra menu chính của chương trình.

d. Drawable và Raw

Chứa các hình ảnh và âm thanh cho các chức năng của chương trình.

e. AndroidManifest

Mỗi ứng dụng đều cần có tập tin này để mô tả những thông tin quan trọng của nó cho hệ điều hành biết. Các Activity, Application, Service được khai báo trong AndroidManifest, khi khởi động chương trình, hệ điều hành sẽ tìm kiếm thông tin trong tập AndroidManifest này để gọi Service hay Activity tương ứng.

3.3.5. Thiết kế Service

a. Class Interface

Lớp Interface là một lớp đặc biệt, lớp chỉ dùng để tạo các hàm, từ các lớp thừa kế đến lớp Interface sẽ định nghĩa cho các hàm này. Trong chương trình game cờ tướng được thiết kế ở đây sẽ tạo ra 7 hàm cập nhật giao diện chương trình là: updateMain(), updateChess(), updateSign(), updateSigup(), updateListUser(), updateChat(), updateThoiGian().

b. Class Application

Lớp Application quản lý danh sách các Activity chịu sự điều khiển của Service. Mỗi khi có sự thay đổi, Service sẽ thông báo đến tất cả các Activity thông qua lớp này. Các Activity sẽ cập nhật giao diện theo các hàm đã định nghĩa lại được thừa kế từ lớp Interface.

Lớp Service được start() khi bắt đầu chương trình, chịu chức năng điều khiển tất cả các giao diện đến người dùng. Tại đây, khởi tạo một Socket kết nối đến Server ở cổng 20000. Một luồng luôn lắng nghe các thông điệp từ phía Server rồi phân tích chuỗi sau đó đưa qua một luồng khác để cập nhật lại giao diện chương trình.

3.3.6. Thiết kế giao diện và chức năng

a. Đăng ký

Giao diện gồm các thành phần: hình nền – Background, các nút – Button, các ô nhập văn bản – EditText, danh sách – Spinner. Các thành phần được bao trong thanh cuộn – ScrollView và thiết kế theo một chiều nhất định dạng LinearLayout.

Để đăng kí, người chơi cần nhập đầy đủ các thông tin sau username – tên, emailaddress – địa chỉ mail, password – mật khẩu, repassword – nhập lại mật khẩu, chọn quốc tịch. Nếu đăng ký thành công sẽ chuyển sang giao diện đăng nhập để đăng nhập vào chương trình.

b. Đăng nhập

Đăng nhập: với tài khoản đã đăng kí thành công, nhập vào username – tên và password – mật khẩu rồi chọn **Sign In** để đăng nhập hoặc chưa có tài khoản thì chọn **Sign Up** để đăng kí.

c. Cài đặt

Giao diện gồm các thành phần: ô chọn tắt mở nhạc - CheckBoxPreference, chọn màu cờ (đỏ hoặc đen) – ListPreference.

d. Trợ giúp - Chat

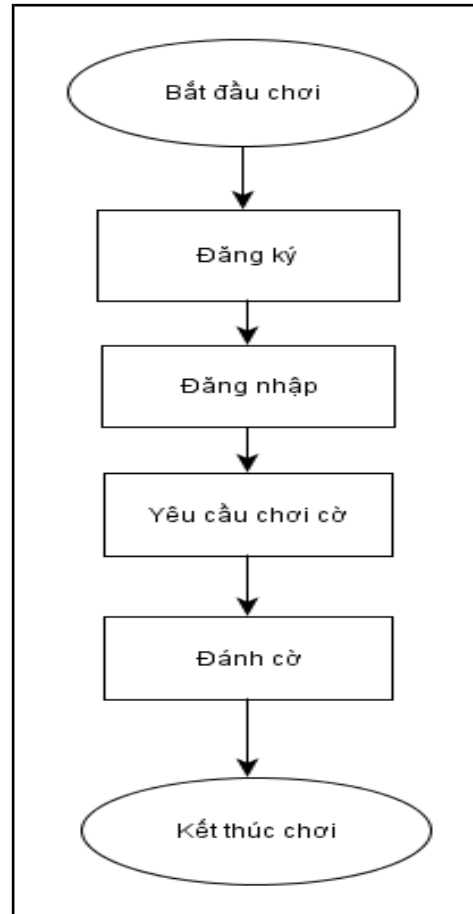
Trợ giúp giao diện là một hình nền – Background. Chat gồm một ô EditText để nhập nội dung chat, một Button để gửi nội dung chat và một EditText để hiển thị nội dung chat.

e. Bàn cờ

Bàn cờ được thiết kế bằng cách sử dụng đối tượng View. Giao diện được vẽ lại liên tục, nếu màn hình có chiều ngang nhỏ hơn chiều dọc thì chiều ngang của bàn cờ sẽ bằng với chiều ngang của màn hình và chiều dọc của bàn cờ sẽ bằng chiều ngang bàn cờ nhân với tỉ lệ bàn cờ, còn ngược lại chiều dọc của bàn cờ sẽ bằng chiều dọc của bàn cờ và chiều ngang của bàn cờ bằng chiều dọc chia cho tỉ lệ bàn cờ. Tỉ lệ bàn cờ ở đây được tính bằng chiều dài chia chiều rộng. Bàn cờ được chia làm 32 quân cờ được vẽ lên màn hình bằng phương thức drawBitmap.

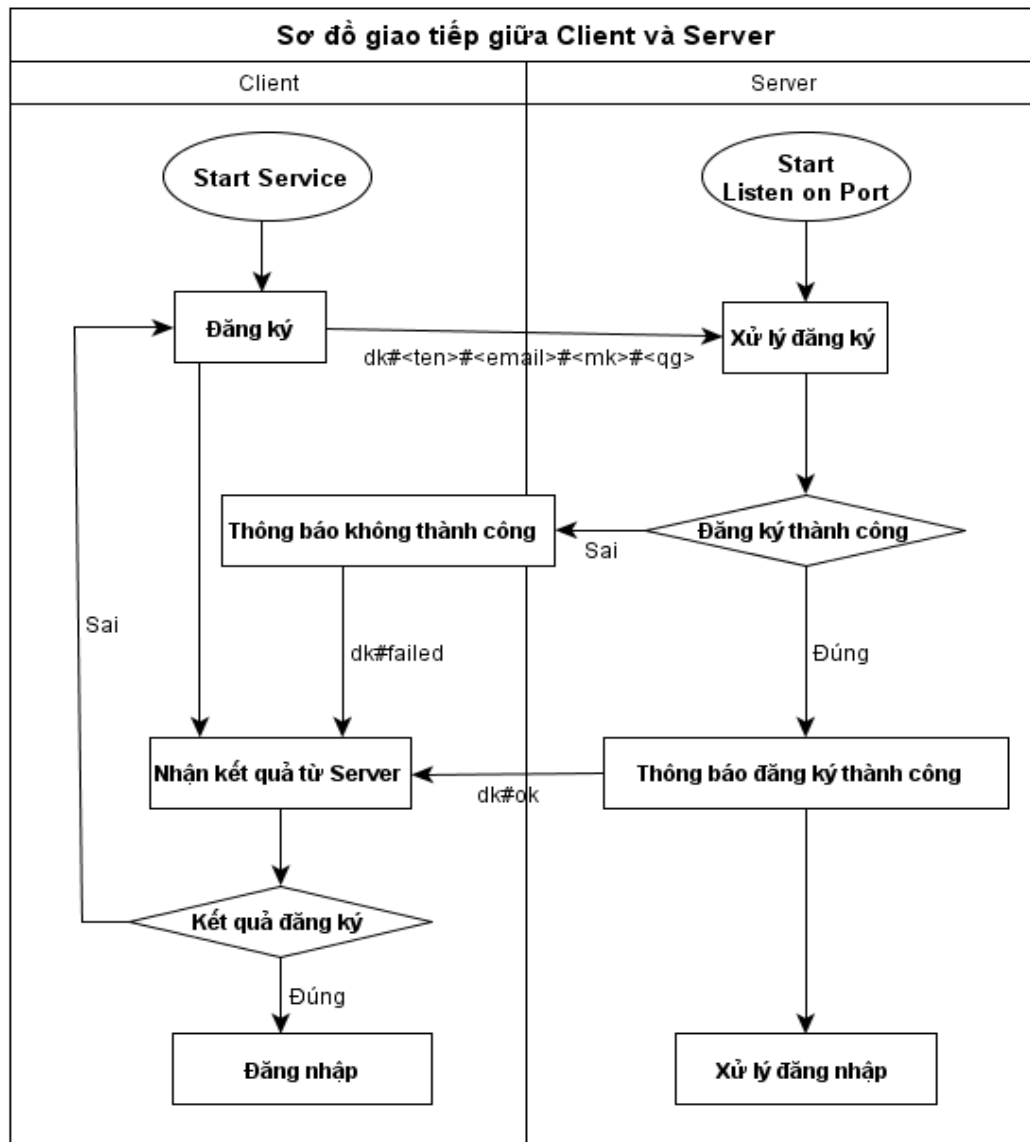
3.4. CẤU TRÚC THÔNG ĐIỆP GIỮA CLIENT VÀ SERVER

3.4.1. Mô hình tổng quan



Hình 3.5. Mô hình tổng quan

3.4.2. Đăng ký



Hình 3.6. Sơ đồ giao tiếp Client - Server khi đăng ký

❖ Client:

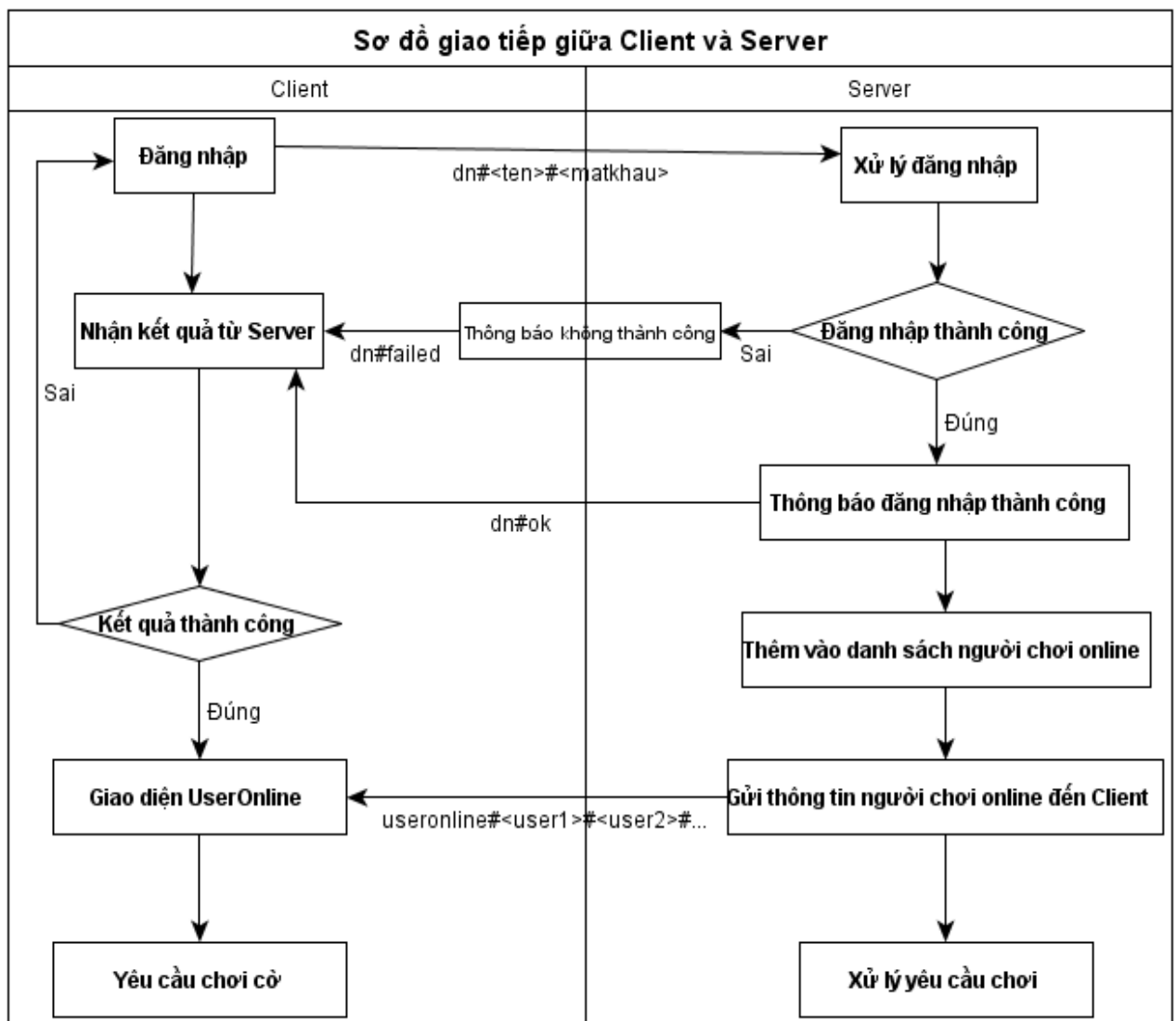
- **dk#tendangky#email#matkhau#tenquocgiadangky**: dùng để đăng ký tài khoản khi người chơi chưa có tài khoản. Ví dụ đăng ký tài khoản **tctam** có mật khẩu **tctam**, email là tctam443@student.ctu.edu.vn, quốc gia là Việt Nam thì cấu trúc thông điệp được gửi như sau:
dk#tctam#tctam443@student.ctu.edu.vn#VN

❖ Server:

- **dk#ok**: dùng để thông báo đăng ký thành công cho Client.

- **dk#failed:** dùng để thông báo đăng ký không thành công cho Client.
- **dk#failed#username:** dùng thông báo đăng ký không thành công trong hợp username đã tồn tại.
- **dk#failed#email:** dùng để thông báo đăng ký không thành công trong trường hợp email không hợp lệ.

3.4.3. Đăng nhập



Hình 3.7. Sơ đồ giao tiếp Client - Server khi đăng nhập

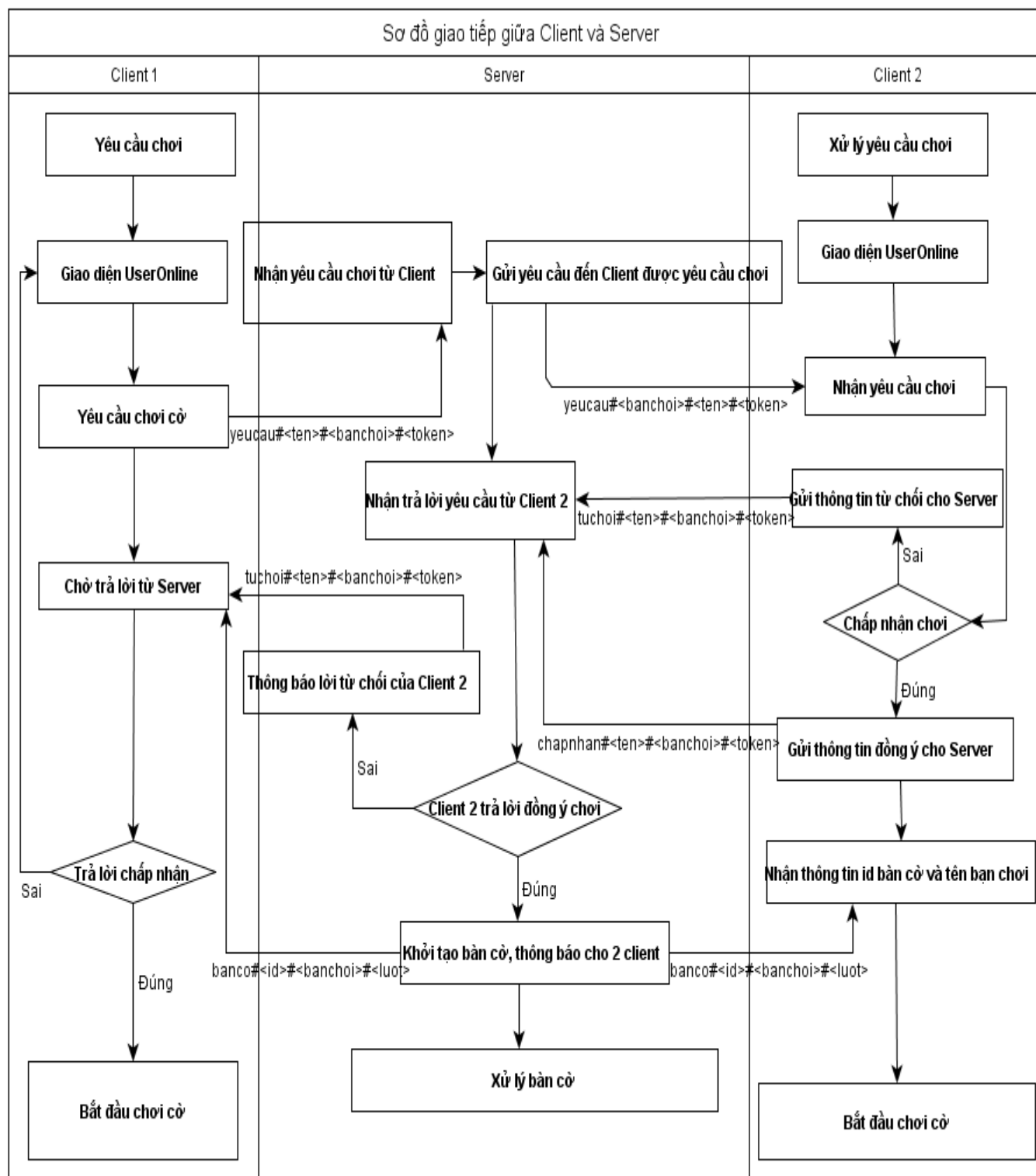
❖ Client:

- **dn#tendangnhap#matkhau:** dùng để đăng nhập. Ví dụ đăng nhập tài khoản **ptatuyet** có mật khẩu là **ptatuyet** thì cấu trúc thông điệp sẽ là **dn#ptatuyet#ptatuyet**.

❖ Server:

- **dn#ok#token:** dùng để thông báo đăng nhập thành công cho Client và gửi thông tin **token** được tạo ra cho Client.
- **dn#failed:** dùng để thông báo đăng nhập không thành công.
- **dn#failed#username:** dùng để thông báo đăng nhập không thành công trong trường hợp user không tồn tại.
- **dn#failed#password:** dùng để thông báo đăng nhập không thành công trong trường hợp mật khẩu sai.
- **useronline#tenuser1|diemuser1|trangthaiuser1|quocgiauser1#tenuser2|diemuser2|trangthaiuser2|quocgiauser2#....:** gửi cho Client biết danh người chơi đã đăng nhập vào Server. Ví dụ, hiện tại đang có 2 người chơi là có tên đăng nhập là **tctam** và **ptatuyet** đã đăng nhập vào Server, người chơi **tctam** có điểm là 100, trạng thái là đang chơi cờ, quốc gia là Việt Nam và người chơi **ptatuyet** có điểm là 200, trạng thái là đang rỗi, quốc gia là Việt Nam thì Server sẽ gửi thông điệp cho Client là: **useronline#tctam#100|1|VN#ptatuyet|200|0|VN**

3.4.4. Yêu cầu chơi



Hình 3.8. Sơ đồ giao tiếp Client - Server khi yêu cầu chơi

❖ Client:

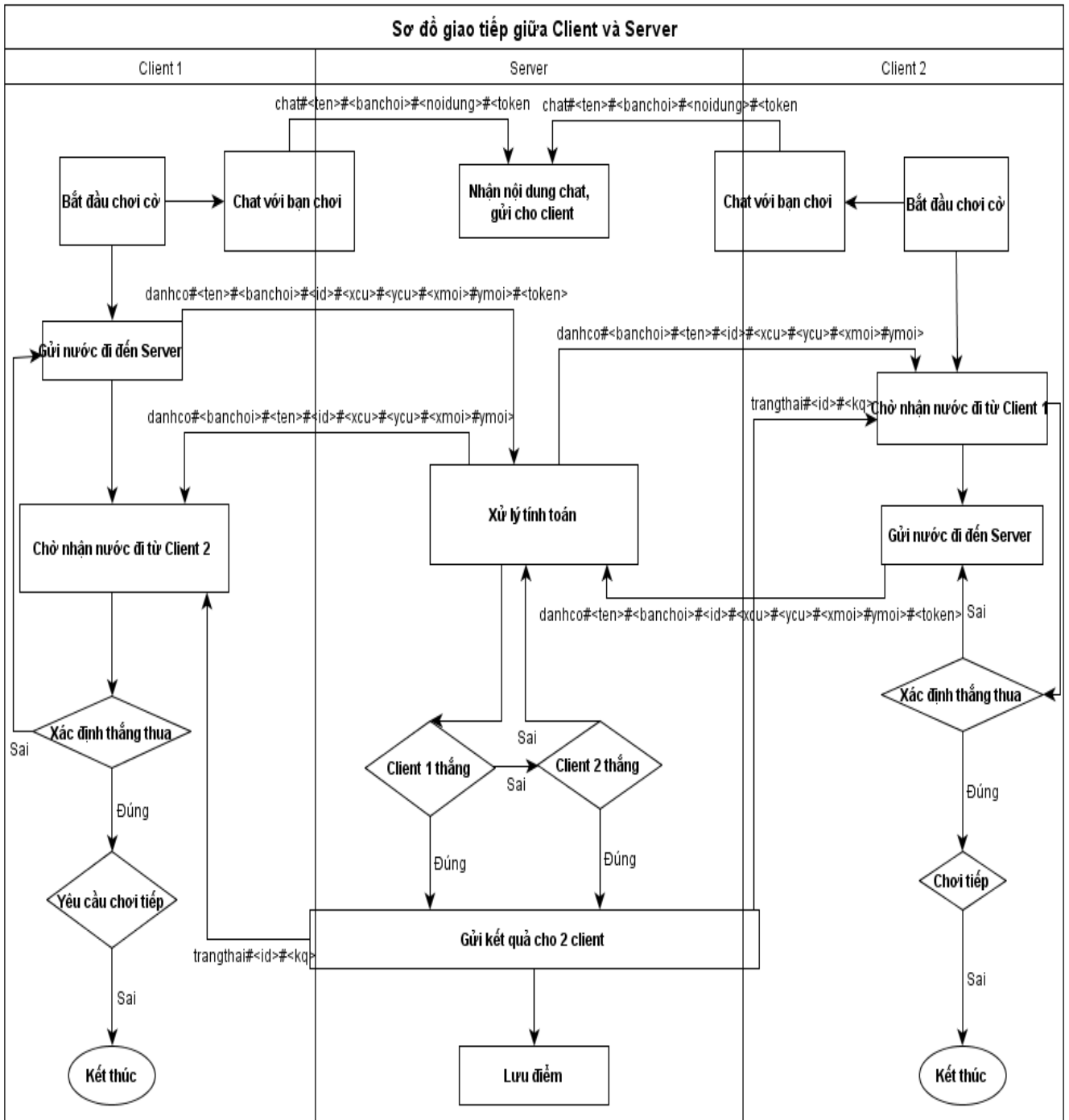
- **yeucau#tendangnhap#tendangnhapbanchoi#token:** dùng để gửi yêu cầu đến người chơi khác chơi khác đang đăng nhập chơi cờ với mình. Ví dụ đang đăng nhập dưới tài khoản **tctam**, token của **tctam** 12345 muốn yêu cầu người có tài khoản **ptatuyet** chơi cờ, thì người chơi **tctam** sẽ gửi thông điệp có cấu trúc như sau: **yeucau#tctam#ptatuyet#12345**.
- **chapnhan#tendangnhap#tendangnhapbanchoi#token:** dùng để chấp nhận yêu cầu chơi của người chơi khác. Ví dụ đang đăng nhập dưới tài khoản **tctam**, token của **tctam** 12345, người chơi có tài khoản **ptatuyet** gửi yêu cầu đánh cờ đến người chơi **tctam**, để chấp nhận thì người chơi **tctam** sẽ gửi thông điệp trả lời: **chapnhan#tctam#ptatuyet#12345**.
- **tuchoi#tendangnhap#tendangnhapbanchoi#token:** dùng để từ chối yêu cầu chơi của người chơi khác. Ví dụ đang đăng nhập dưới tài khoản **tctam**, token của **tctam** 12345, người chơi có tài khoản **ptatuyet** gửi yêu cầu đánh cờ đến người chơi **tctam**, muốn từ chối thì người chơi **tctam** sẽ gửi thông điệp trả lời là **tuchoi#tctam#ptatuyet#12345**.

❖ Server:

- **yeucau#tunguoichoi#dennguoichoi:** dùng để gửi yêu cầu chơi đến Client khi có người chơi khác yêu cầu. Ví dụ để thông báo yêu cầu đánh cờ từ **tctam** đến Client **ptatuyet** thì Server sẽ gửi thông điệp là **yeucau#tctam#ptatuyet**.
- **banco#id#tendangnhapdoithu#luotditruoc:** gửi thông tin id bàn cờ là **id**, thông tin đối thủ là **tendangnhapdoithu**, lượt đi trước **luotditruoc** (đi trước **luotditruoc** bằng 1, đi sau **luotditruoc** bằng 0), cho hai người chơi khi Server nhận được thông điệp chấp nhận chơi cờ. Ví dụ để gửi thông tin bàn cờ có id là **10** cho người chơi **tctam** rằng đối thủ của mình là **ptatuyet** và gửi cho người chơi **ptatuyet** rằng đối thủ của mình là **tctam**, **tctam** là người đi được đi nước cờ trước thì cấu trúc thông điệp gửi cho **tctam** là: **banco#10#ptatuyet#1**, cấu trúc thông điệp gửi cho **ptatuyet** là: **banco#10#tctam#0**

- **tuchoi#tunguoichoi#dennguoichoi:** gửi cho Client biết yêu cầu chơi của nó với Client khác đã bị từ chối. Ví dụ để thông báo **ptatuyet** đã từ chối chơi cờ với **tetam** thì Server sẽ gửi cho tetam thông điệp sau: **tuchoi#ptatuyet#tetam**

3.4.5. Chat và đánh cờ



Hình 3.9. Sơ đồ giao tiếp Client-Server khi đánh cờ

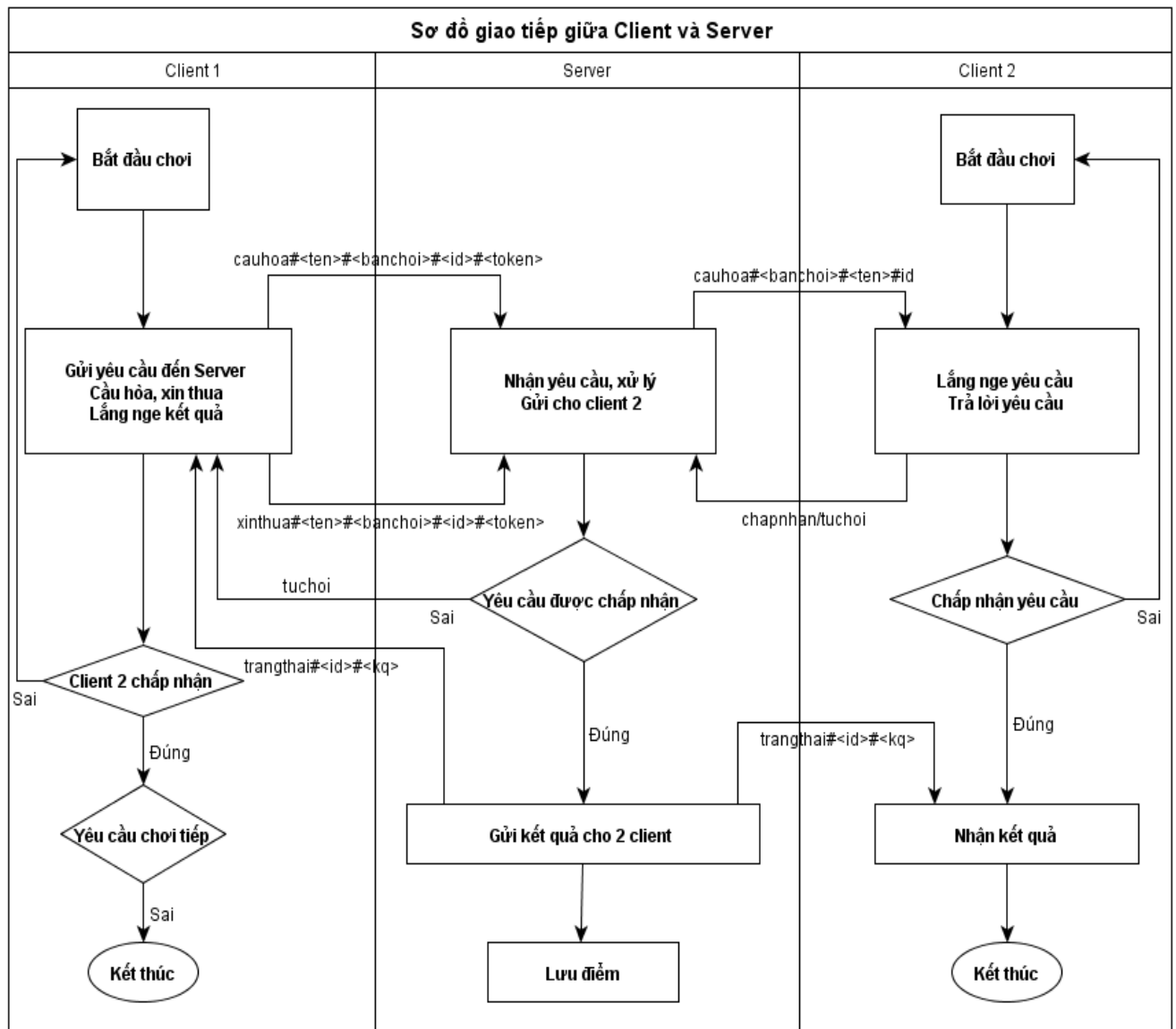
❖ Client:

- **chat#tendangnhap#tendangnhapbanchoi#noidungchat#token:** dùng để chat với người đang chơi với mình. Ví dụ tài khoản **tctam**, token của **tctam** là 12345, và đang đánh cờ với người chơi có tài khoản **ptatuyet**, người chơi **tctam** muốn gửi tin nhắn đến người chơi **ptatuyet** có nội dung “Hello” thì thông điệp là: **chat#tctam#ptatuyet#Hello#12345**.
- **danhco#tendangnhap#tendangnhapnguoichoi#id#vitrixcu#vitriycu#vitrixmoi#vitriymoi#token:** dùng để truyền tải thông tin nước đi con cờ từ người chơi **tendangnhap** đến người chơi **tendangnhapbanchoi**, ở bàn cờ có id là **id**, đi từ vị trí x cũ là **vitrixcu**, y cũ là **vitriycu** đến vị trí x mới là **vitrixmoi**, vị trí y mới là **vitriymoi** đến Server. Ví dụ, người chơi có tài khoản **tctam** đang chơi cờ với người chơi có tài khoản **ptatuyet**, bàn cờ có id là 10, token của **tctam** 12345, người chơi **tctam** đi quân “Tốt” ở vị trí x là 2, vị trí y là 3 đến vị trí x mới là 2, y là 4 thì sẽ gửi thông điệp như sau: **danhco#tctam#ptatuyet#10#2#3#2#4#12345**.

❖ Server:

- **chat#tunguoichoi#dennguoichoi#noidungchat:** để gửi nội dung chat từ **tunguoichoi** đến **dennguoichoi**. Ví dụ khi nhận được nội dung chat từ **tctam** gửi đến **ptatuyet** là “Please go quickly”, Server sẽ gửi đến **ptatuyet** cấu trúc thông điệp như sau: **chat#tctam#ptatuyet#Please go quickly**.
- **danhco#tunguoichoi#dennguoichoi#id#vitrixcu#vitriycu#vitrixmoi#vitriymoi:** dùng để gửi vị trí nước đi cập nhật của bàn cờ từ **tunguoichoi** đến **dennguoichoi** đang chơi cờ ở bàn cờ có id là **id**, quân cờ sẽ được cập nhật từ vị trí x là **vitrixcu**, y là **vitriycu** đến vị trí x mới là **vitrixmoi**, y mới là **vitriymoi**. Ví dụ **tctam** và **ptatuyet** đang chơi cờ ở bàn có id là 10, đến lượt **tctam** đi, **tctam** đi quân “Tốt” từ vị trí x là 4, y là 8 đến vị trí mới x là 4, y là 9 thì Server sẽ gửi thông tin về cho **ptatuyet** có dạng: **danhco#tctam#ptatuyet#10#4#8#4#9**

3.4.6. Cầu hòa, xin thua



Hình 3.10. Sơ đồ giao tiếp Client-Server cầu hòa/xin thua

❖ Client:

- **cauhoa#tendangnhap#tendangnhapbanchoi#id#token**: gửi thông điệp cầu hòa từ người chơi **tendangnhap** đến người chơi **tendangnhapbanchoi** ở bàn cờ có id là **id**. Ví dụ, hai người chơi có tên đăng nhập lần lượt là **tctam** và **ptatuyet** đang chơi cờ ở bàn cờ có **id** là 10, token của **tctam** 12345, người chơi **tctam** gửi yêu cầu cầu hòa người chơi **ptatuyet**, thì sẽ gửi thông điệp cho Server là:

cauhoa#tctam#ptatuyet#10#12345

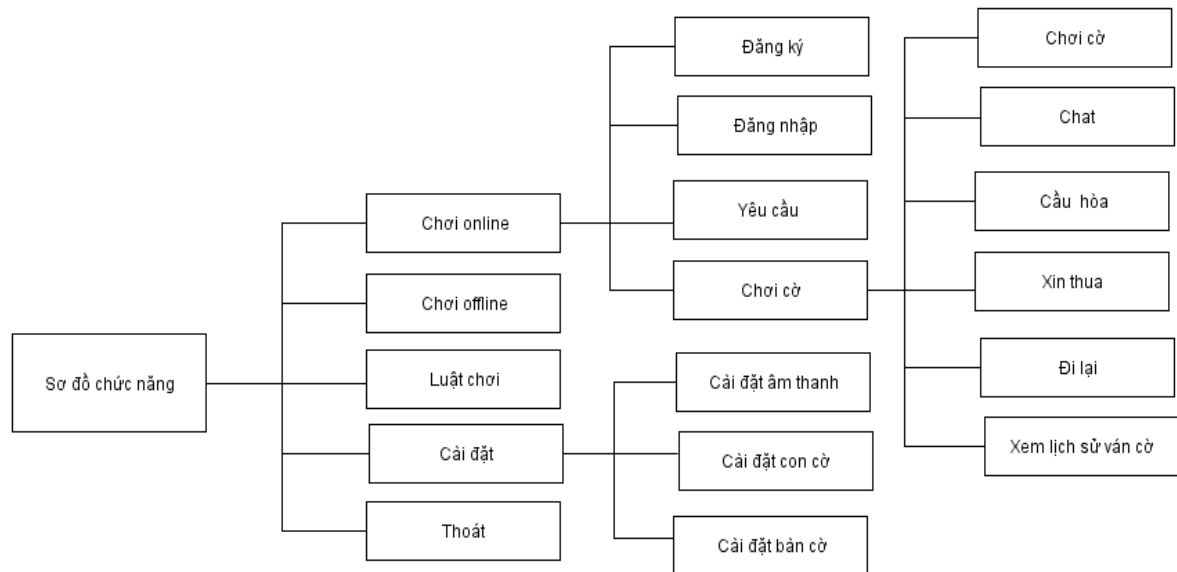
- **chapnhancauhoa#tendangnhap#tendangnhapbanchoi#id#token:** gửi thông điệp chấp nhận cầu hòa đến người chơi đang chơi với mình. Ví dụ, hai người chơi có tên đăng nhập lần lượt là **tctam** và **ptatuyet** đang chơi cờ ở bàn cờ có **id** là 10, token của **tctam** 12345, khi người chơi **ptatuyet** gửi yêu cầu cầu hòa đến người chơi **tctam**, nếu chấp nhận lời cầu hòa của người chơi **ptatuyet**, thì người chơi **tctam** sẽ gửi thông điệp là: **chapnhancauhoa#tctam#ptatuyet#10#12345**
- **tuchoicauhoa#tendangnhap#tendangnhapbanchoi#id#token:** gửi thông điệp từ chối cầu hòa đến người chơi đang chơi với mình. Ví dụ, hai người chơi có tên đăng nhập lần lượt là **tctam** có token là 12345 và **ptatuyet** đang chơi cờ ở bàn cờ có **id** là 10, khi người chơi **ptatuyet** gửi yêu cầu cầu hòa đến người chơi **tctam**, nếu từ chối lời cầu hòa **tctam** sẽ gửi thông điệp là: **tuchoicauhoa#tctam#ptatuyet#10#12345**
- **xinhua#tendangnhap#tendangnhapbanchoi#id#token:** gửi thông điệp xin thua đến bạn chơi. Ví dụ, hai người chơi có tên đăng nhập lần lượt là **tctam** có token là 12345 và **ptatuyet** đang chơi cờ ở bàn cờ có **id** là 10, người chơi **tctam** gửi yêu cầu xin thua người chơi **ptatuyet**, thông điệp là: **xinhua#tctam#ptatuyet#10#12345**

❖ Server:

- **cauhoa#tunguoichoi#dennguoichoi#id:** gửi thông điệp cầu hòa từ người chơi này đến người chơi khác khi hai người chơi này đang chơi cờ ở bàn cờ có **id** là **id**. Ví dụ, hai người chơi có tên đăng nhập lần lượt là **tctam** và **ptatuyet** đang chơi cờ ở bàn cờ có **id** là 10, người chơi **tctam** gửi yêu cầu cầu hòa người chơi **ptatuyet** lên Server, thì Server sẽ gửi thông điệp về cho người chơi **ptatuyet** là: **cauhoa#tctam#ptatuyet#10**
- **trangthai#id#noidungtrangthai:** dùng để thông báo kết quả ván cờ có **id** là **id** đến hai người chơi với nội dung là **noidungtrangthai** khi phân định thắng thua hoặc hòa, cụ thể **noidungtrangthai** có 3 giá trị là **1, 0, -1** tương ứng với thắng, hòa, thua.

3.5. CHƯƠNG TRÌNH DEMO

3.5.1. Sơ đồ chức năng



Hình 3.11. Sơ đồ chức năng của chương trình

3.5.2. Giao diện chương trình

➤ Menu chính của chương trình



Hình 3.12. Giao diện bắt đầu

3.5.3. Giao diện đăng nhập

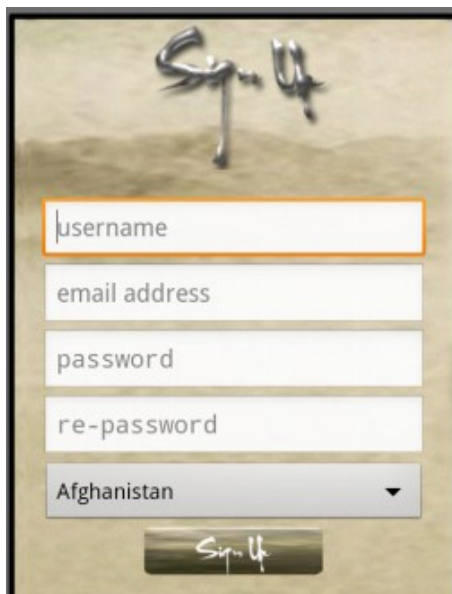
- Nếu chọn **Play Online** sẽ chuyển sang giao diện đăng nhập. Người chơi sẽ sử dụng tài khoản để đăng nhập vào chương trình.



Hình 3.13. Giao diện đăng nhập

3.5.4. Giao diện đăng ký

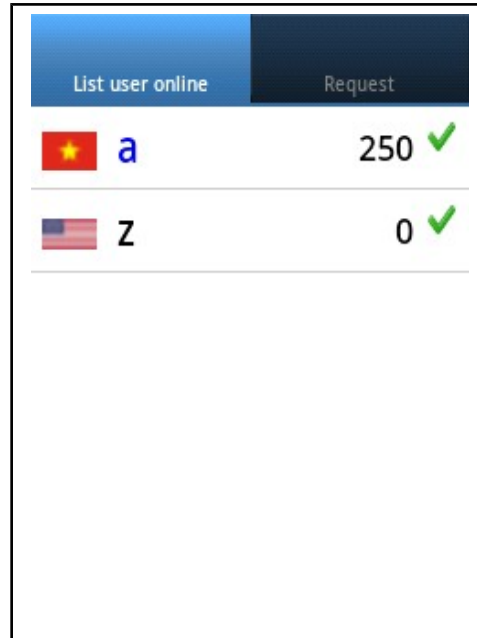
- Nếu chưa có tài khoản để đăng nhập, người chơi chọn **Sign Up** để đăng kí thành viên.



Hình 3.14. Giao diện đăng ký

3.5.5. Giao diện người chơi online

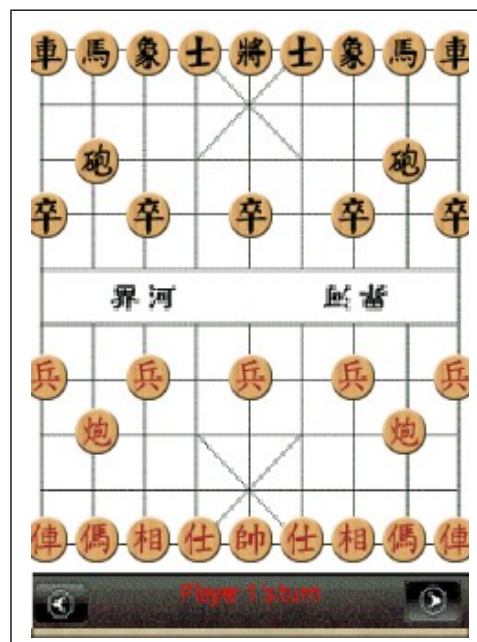
- Sau khi đăng nhập thành công, giao diện sẽ chuyển sang danh sách người chơi đang online. Người chơi muốn chơi cờ sẽ yêu cầu người chơi khác bằng cách chọn vào người chơi đó.



Hình 3.15. Giao diện danh sách người chơi

3.5.6. Giao diện chơi cờ

- Bàn cờ gồm 2 bên chơi được phân biệt bằng màu đen và đỏ.



Hình 3.16. Giao diện bàn cờ

3.5.7. Giao diện chat

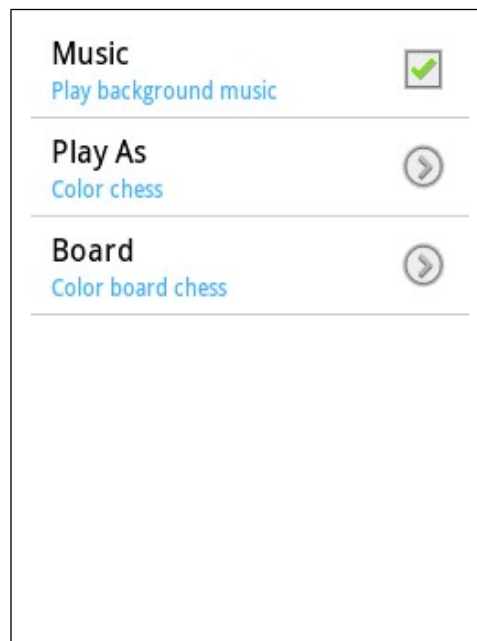
- Trong khi chơi cờ có thể vào menu **Chat** để trò chuyện với bạn chơi



Hình 3.17. Giao diện Chat

3.5.8. Giao diện cài đặt

- Chọn menu **Settings** để thiết lập chơi: tắt/mở nhạc nền hoặc chọn màu cờ đỏ/đen, chọn bàn cờ.



Hình 3.18. Giao diện cài đặt

3.5.9. Giao diện trợ giúp

- Chọn menu **Rule** để xem luật chơi cờ tướng.



Hình 3.19. Giao diện luật chơi

XH□□NT 4. KẾT LUẬN – HƯỚNG PHÁT TRIỂN

4.1. KẾT QUẢ ĐẠT ĐƯỢC

4.1.1. Kết quả đạt được

Sau 15 tuần thực hiện, đề tài đã được hoàn thành và đạt được những yêu cầu mà đề tài đã đặt ra. Tác giả đã thực hiện được những công việc sau:

❖ Server:

- Xây dựng Server trung gian là nơi giao tiếp giữa các Client và là nơi tính toán, lưu trữ.
- Thiết lập và xử lý đa luồng cho các ván cờ.
- Ứng dụng kĩ thuật lập trình song song MPI trong việc thiết kế các giải thuật.
- Quản lý người chơi.
- Quản lý các ván cờ.

❖ Client:

- Thiết kế giao diện cho các chức năng.
- Xây dựng chương trình nối kết đến Server: đăng ký, đăng nhập, xem danh sách người chơi online và chơi cờ.
- Chơi cờ online hoặc offline: xử lý các sự kiện trên màn hình cảm ứng và thiết lập các yêu cầu cơ bản cho game.
- Xử lý giao diện cho các độ phân giải màn hình khác nhau.

4.1.2. Ưu điểm

Sản phẩm chương trình là sản phẩm hoàn thiện có những ưu điểm sau:

❖ Về lý thuyết:

- Nắm vững được kiến thức cơ bản về lập trình Java.
- Nắm vững được những kiến thức cơ bản về lập trình trên hệ điều hành Android.
- Nắm vững được kiến thức về mạng máy tính, giao thức TCP/IP.
- Biết rõ về lập trình mạng Socket, cơ chế lập trình đa luồng và vận dụng vào để xây dựng đề tài.

- Áp dụng được kỹ thuật lập trình song song MPI vào chương trình.
- Tổng hợp được kiến thức trong quá trình học, vận dụng lý thuyết để giải quyết vấn đề thực tế.

❖ Về chương trình:

- Chương trình đã đáp ứng các yêu cầu mà phạm vi đề tài đã đặt ra.
- Chương trình đáp ứng đầy đủ các tính năng.
- Giao diện thân thiện với người dùng có thể tương thích với nhiều độ phân giải màn hình.
- Chương trình tốc độ xử lý nhanh và đáp ứng cho nhiều người dùng.
- Có thể áp dụng và triển khai vào thực tế.

4.1.3. Hạn chế

Bên cạnh những ưu điểm đạt được, trong quá trình thực hiện, đề tài cũng còn gặp một số mặt hạn chế:

- Chưa có điều kiện triển khai trên hệ thống lớn.
- Sử dụng giao diện lập trình mức thấp, lập trình đồ họa 2D, chưa sử dụng thư viện hỗ trợ nên giao diện còn chưa đẹp lắm.
- Chưa bổ sung một số tính năng mới như người chơi tự đánh với máy, hoặc đánh qua bluetooth.

4.2. HƯỚNG PHÁT TRIỂN

Để phát huy những ưu điểm và khắc phục những mặt hạn chế, sau khi hoàn thành đề tài, tác giả sẽ phát triển đề tài theo những hướng sau:

- Sử dụng thư viện OpenGL để cải tiến thành giao diện đồ họa 3D, tăng sức hấp dẫn của chương trình.
- Triển khai trên hệ thống thật và áp dụng vào thực tế.
- Bổ sung thêm tính năng chơi với máy, chơi qua bluetooth.
- Phát triển thành phần mềm thương mại.

TÀI LIỆU THAM KHẢO

- [1] <http://developer.android.com>
- [2] <http://stackoverflow.com>
- [3] <http://docs.oracle.com>
- [4] <http://www.droidviet.com>
- [5] <http://www.vietnamandroid.com>
- [6] <http://androidvietnam.blogspot.com>
- [7] <http://www.hpjava.org/mpiJava.html>
- [8] <http://congdongjava.com>
- [9] <http://mpj-express.org/>
- [10] <http://yinyangit.wordpress.com/>
- [11] <http://java.sun.com>
- [12] <http://vi.wikipedia.org>
- [13] <http://diendandroid.com>
- [14] **Grant Allen.** *Beginning Android 4.*
- [15] **Ts. Ngô Bá Hùng, Ths. Nguyễn Công Huy.** *Giáo trình Lập trình truyền thông.* Khoa CNTT & TT – Đại học Cần Thơ, năm 2008.
- [16] **Ks Hồ Quang Thái.** *Slide bài giảng Chuyên đề ngôn ngữ lập trình 2.* Khoa CNTT & TT – Đại học Cần Thơ, năm 2010.
- [17] *Slide bài giảng Tính toán lưới.* Khoa CNTT & TT – Đại học Cần Thơ, năm 2012.
- [18] **Lê Minh Quân.** *Luận văn tốt nghiệp Đại học: Tìm hiểu về nền tảng Android và xây dựng các bài thực hành lập trình cho thiết bị di động.* Khoa CNTT&TT, Trường Đại học Cần Thơ.
- [19] **ThS. Phan Tấn Tài, KS. Trần Nguyễn Minh Thái, KS. Nguyễn Thanh Hải, KS. Nguyễn Thị Thu An.** *Phân tích thiết kế hệ thống thông tin.* Khoa CNTT & TT – Đại học Cần Thơ, năm 2010.
- [20] **Trần Ngân Bình.** *Introduction to Databases.* Khoa CNTT & TT – Đại học Cần Thơ, năm 2008.

PHỤ LỤC A: CÀI ĐẶT PHẦN MỀM ĐỂ LẬP TRÌNH ANDROID

Công cụ để phát triển ứng dụng trên Android là gói Android SDK. Để có thể lập trình cho Android thì phải tích hợp gói này vào trong eclipse. Android SDK và Eclipse đều chạy trên nền java nên máy tính phải có cài đặt JAVA.

- JAVA: <http://www.java.com/en/download/manual.jsp>
- Android SDK: <http://developer.android.com/index.html>
- Eclipse: <http://www.eclipse.org/downloads/>

1. Hướng dẫn cài đặt Android với Eclipse:

➤ Download Android SDK:

Android SDK thực chất là tập hợp các công cụ và thư viện để phát triển các ứng dụng trên nền tảng hệ điều hành Android.

- **Bước 1:** Vào trang <http://developer.android.com/sdk/index.html> để tải Android SDK Starter.

Tùy thuộc vào hệ điều hành mà bạn chọn bản Mac, Linux hay Window.
Sau đây hướng dẫn tải bản cho Windows.

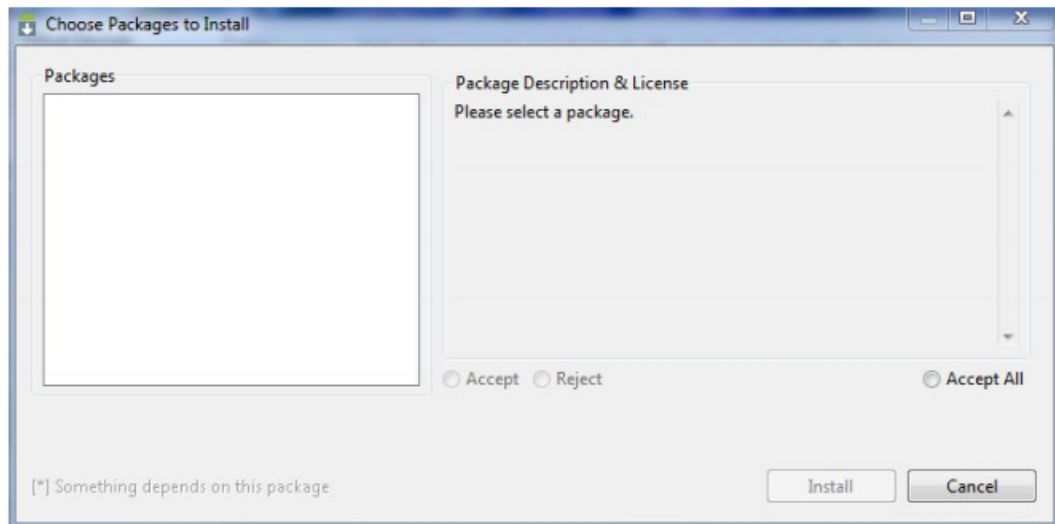
Platform	Package	Size	MD5 Checksum
Windows	android-sdk_r06-windows.zip	23293160 bytes	7c7fcec3c6b5c7c3df6ae654b27effb5
Mac OS X (intel)	android-sdk_r06-mac_86.zip	19108077 bytes	c92abf66a82c7a3f2b8493ebe025dd22
Linux (i386)	android-sdk_r06-linux_86.tgz	16971139 bytes	848371e4bf068dbb582b709f4e56d903

- **Bước 2:** Giải nén file zip vừa tải về. Chạy SDK Setup.exe.

Có thể gặp thông báo lỗi Fetching <https://dl-sl...> Failed to fetch... Close thông báo này lại.

Tiếp theo cửa sổ **Choose Packages to Install** xuất hiện.

Nếu cửa sổ này trống rỗng -> Cancel. -> Quay về cửa sổ Android SDK and AVD manager -> Chọn Setting, đánh dấu vào ô Force <https://...> -> Chọn Available Packages.

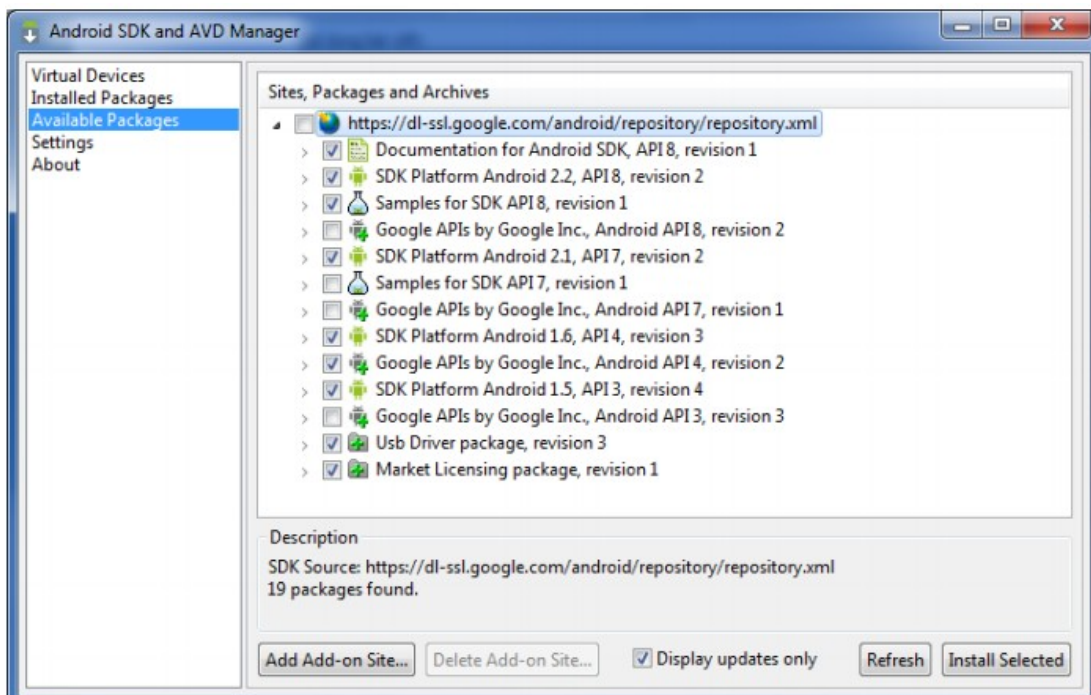


- **Bước 3:** Đánh dấu các **Packages** muốn tải.

Có thể tải hết chỉ chọn những Packagex cần thiết để sử dụng như mình (USB drivers chỉ dành cho sử dụng Windows và muốn phát triển ứng dụng test bằng điện thoại thật).

-> Install Selected

-> Install



➤ **Tích hợp Android SDK vào Eclipse:**

- **Bước 1:** Tải Eclipse nếu chưa có. Có thể dùng 1 trong 2 bản sau: Eclipse for Java Developers, hoặc Eclipse for Java and Report Developers.

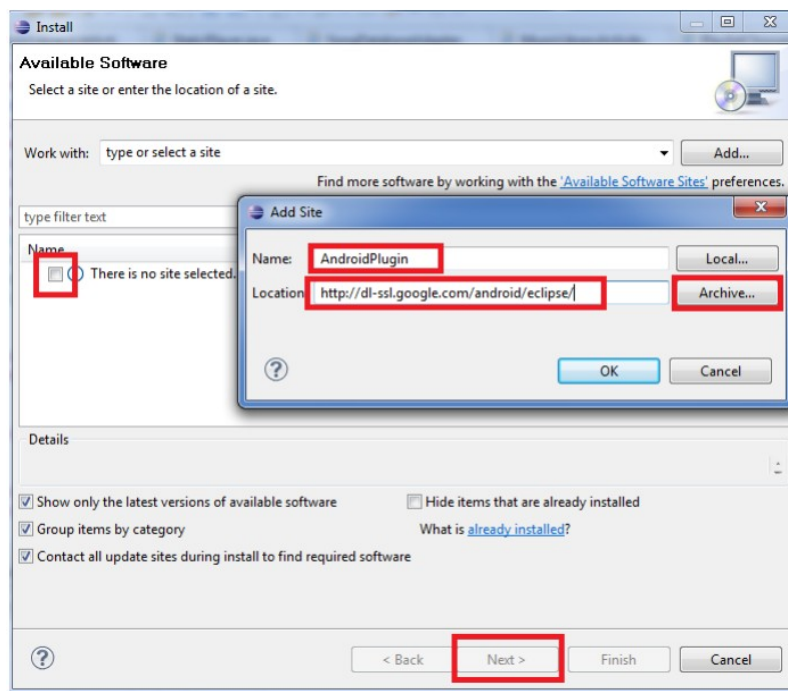
- **Bước 2:** Khởi chạy Eclipse, vào Help -> Install new softwares.

Chọn Add, gõ vào ô Name tên bạn muốn và Location gõ vào địa chỉ để tải về ADT: <https://dl-ssl.google.com/android/eclipse/> hoặc

<http://dl-ssl.google.com/android/eclipse/> nếu https không hoạt động. Ngoài ra cũng có thể tải thẳng ADT về máy theo link <http://dl.google.com/android/ADT-0.9.7.zip>, chọn Archive và browse tới file này (lưu ý không giải nén)

-> OK

-> Check vào phần dưới ô Name (sẽ hiện ra dòng Developer Tools).

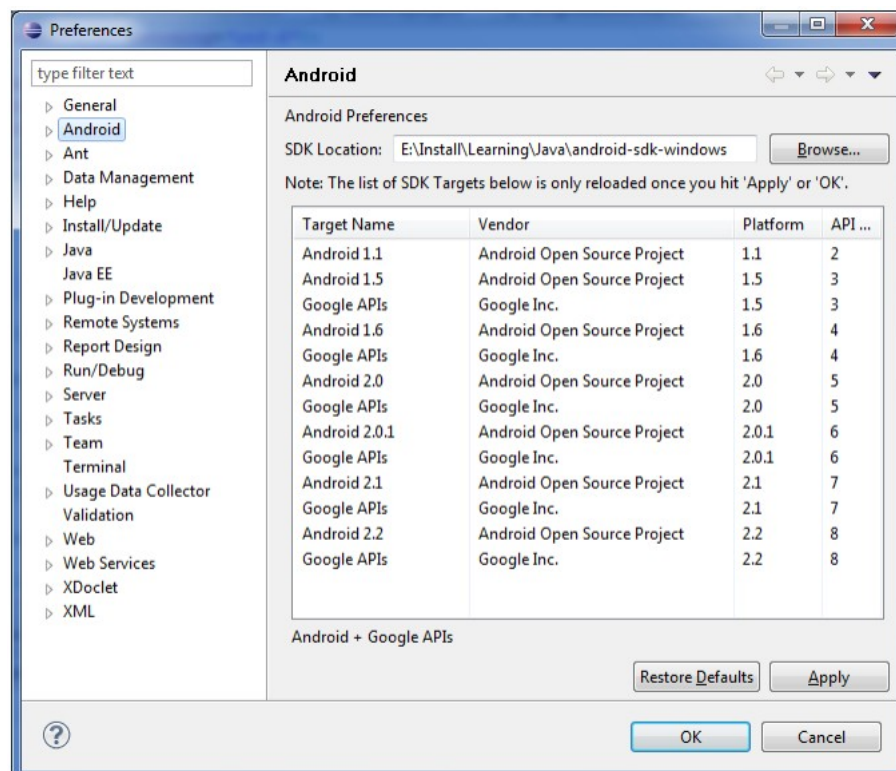


- **Bước 3:** Tiếp theo chọn Next, tiếp tục Next và Accept, Next, sau cùng là Finish.

- **Bước 4:** Eclipse -> Windows -> Preferences -> Android

Nhấn nút Browse và chỉnh đường dẫn tới thư mục của Android SDK đã tải lúc trước.

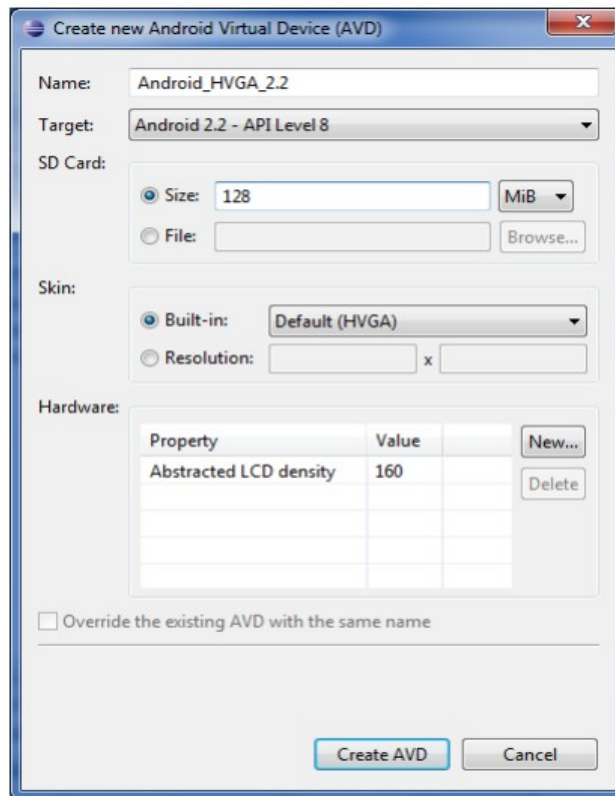
-> Apply -> OK



2. Android Virtual Device (Emulator):

AVD là máy ảo cho developer phát triển ứng dụng. Để tạo 1 AVD thực hiện các bước sau:

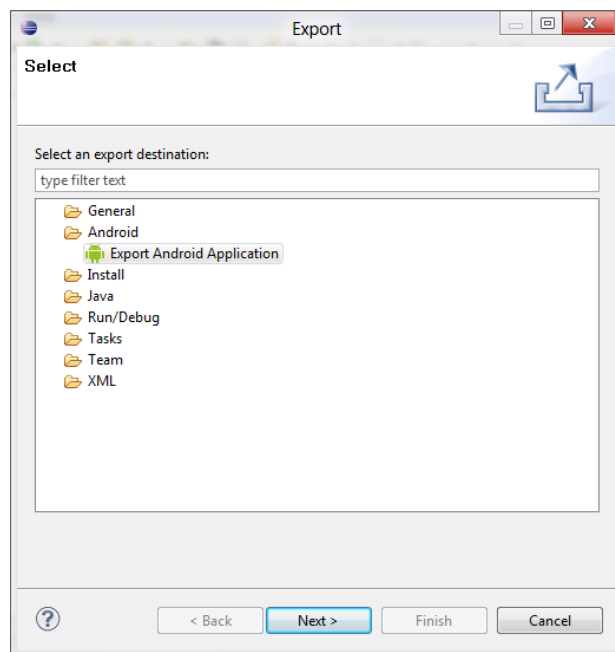
- **Bước 1: Windows -> Android SDK and AVD Manager -> Virtual Devices** chọn New.
- **Bước 2:** Cửa sổ Create new AVD hiện ra, thiết lập cài đặt cho AVD:
 - ◆ Name: Đặt tên cho AVD.
 - ◆ Target: Chọn phiên bản hệ điều hành.
 - ◆ SD Card: Cấp Size SD card ảo cho AVD, hoặc chỉnh tới file đã có sẵn. Nhiều AVD có thể dùng chung 1 Sdcard (chỉ cần tạo lần đầu, các lần sau chỉnh đường dẫn tới file đó).
 - ◆ Skin: có thể để Default (HVGA) hoặc chọn kích cỡ màn hình theo ý muốn: HVGA (phân giải 320-480 như G1, G2, i5700...), QVGA (240-320 như HTC Wildfire...), WVGA854 (480-854 như Milestone, NexusOne...)
- **Bước 3:** Create AVD.



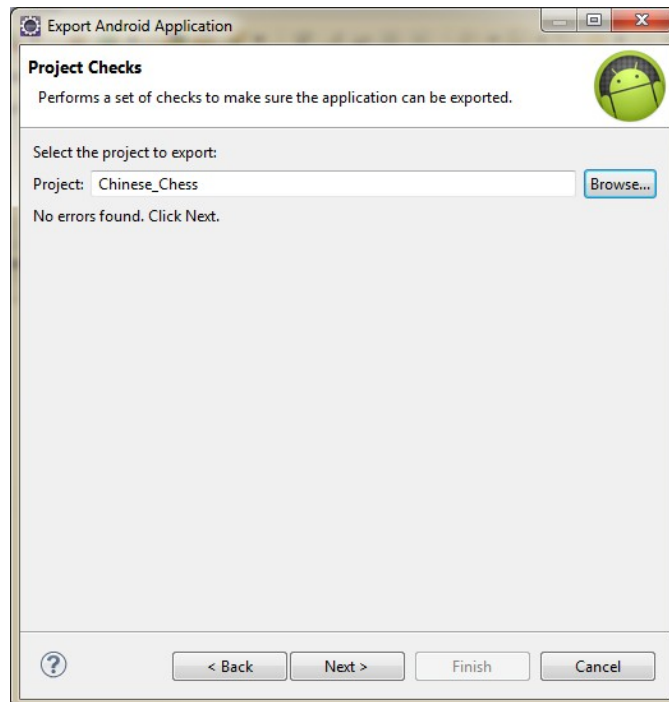
3. Đóng gói ứng dụng thành tập tin .apk

Để đóng gói project thành tập tin APK thì trước tiên cần đảm bảo project đó không có lỗi (error). Trong Eclipse thực hiện như sau:

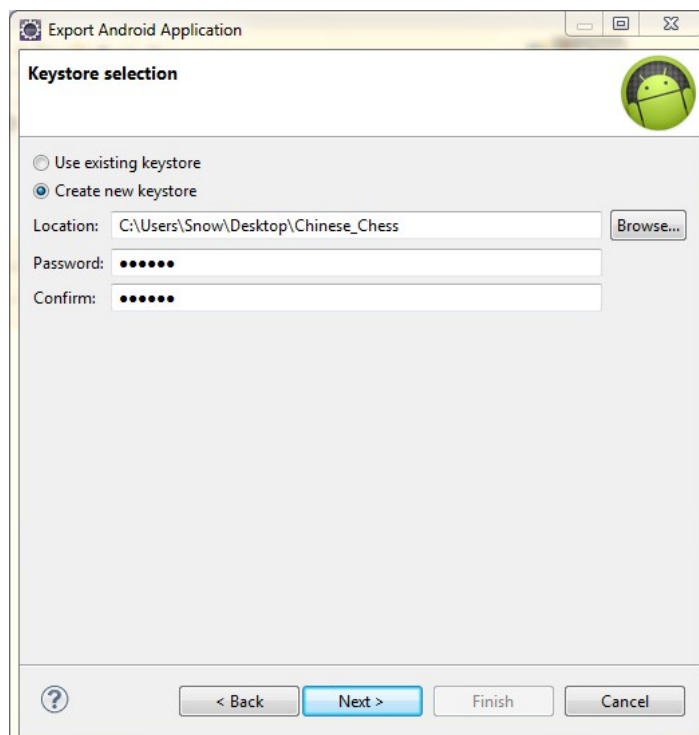
- **Bước 1:** Nhấp phải chuột lên tên project rồi chọn Export, trong hộp thoại xuất hiện chọn **Android > Export Android Application** rồi nhấn **Next**.



- **Bước 2:** Trong ô **Project** mặc định sẽ là project cần đóng gói, nếu không phải thì nhấn nút **Browse** để chọn lại, tiếp tục nhấn **Next**.



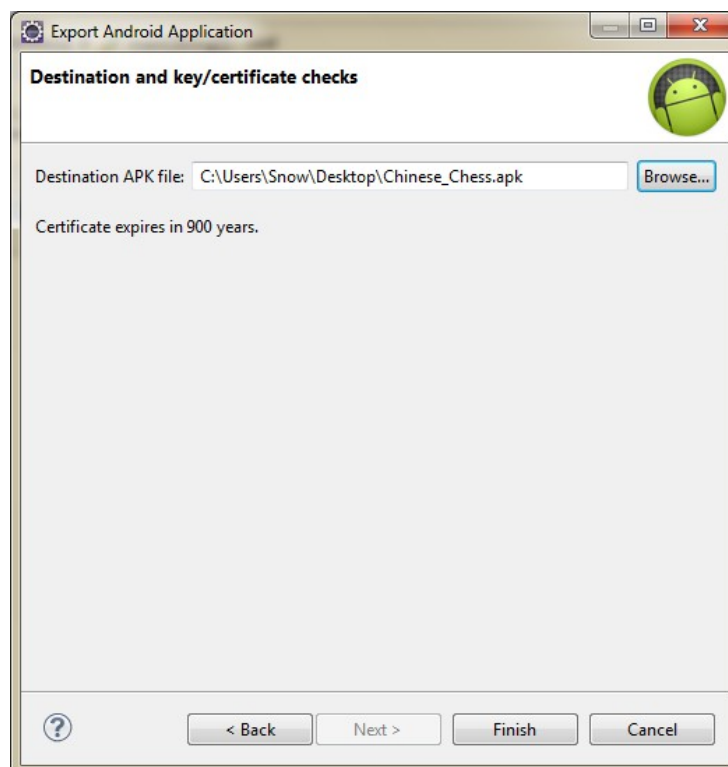
- **Bước 3:** Chọn mục **Create new keystore**, phần **Location** chọn nơi lưu và đặt tên cho **keystore**, phần **Password** (ít nhất 6 ký tự) và **Confirm** nhập mật khẩu, nhấn **Next**.



- **Bước 4:** Cửa sổ tiếp theo nhập các thông tin cần thiết theo sự hướng dẫn của chương trình, rồi nhấn **Next**.



- **Bước 5:** Trong ô **Destination APK** file chọn nơi lưu và đặt tên cho gói APK, cuối cùng nhấn **Finish**.



PHỤ LỤC B: CÀI ĐẶT MPJ EXPRESS

1. Download file:

Đăng ký thông tin tại trang web <http://mpj-express.org/> sau đó download file nén **mpj-v0_38.zip** tại địa chỉ <http://sourceforge.net/projects/mpjexpress/>

2. Yêu cầu:

- Java từ 1.5 trở lên.
- Apache ant 1.6.2 trở lên

3. Cài đặt:

Cấu hình phần mềm MPJ Express có 2 cách: Cách thứ nhất là cấu hình **Multicore** – chạy trên 1 máy, cách thứ 2 là cấu hình **Clusters** – thực thi chương trình MPJ Express trên những Clusters hoặc mạng các máy tính.

➤ Tạo chương trình MPJ Express với tên **HelloWorld.java**

```
import mpi.*;

public class HelloWorld {

    public static void main(String args[]) throws Exception {

        MPI.Init(args);

        int me = MPI.COMM_WORLD.Rank();

        int size = MPI.COMM_WORLD.Size();

        System.out.println("Hi from <"+me+">");

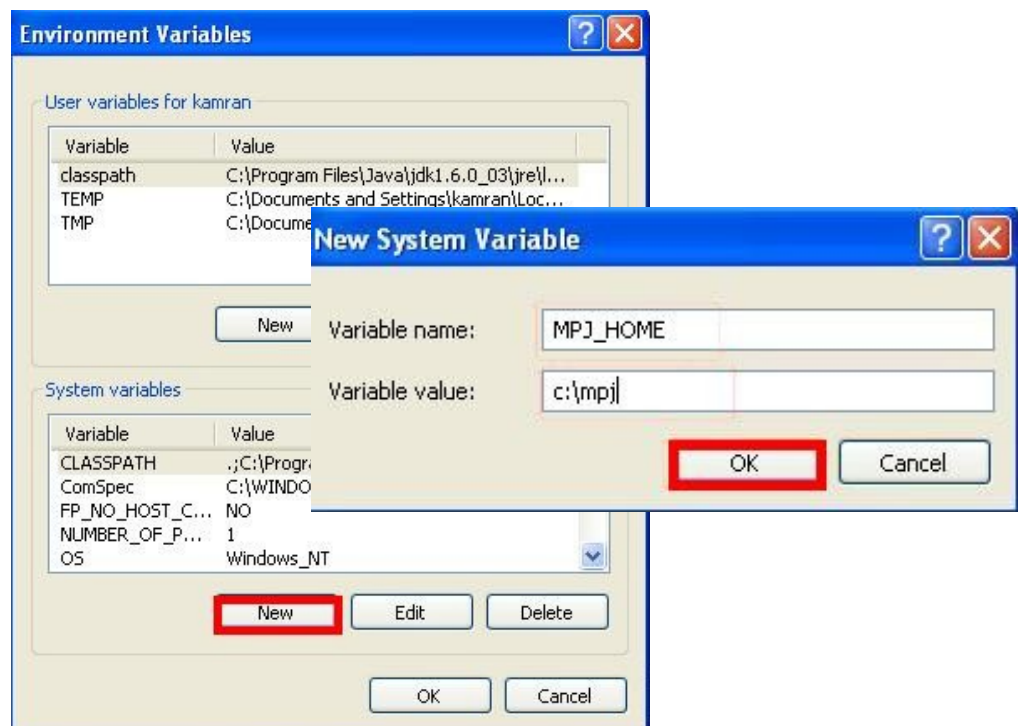
        MPI.Finalize();

    }
```

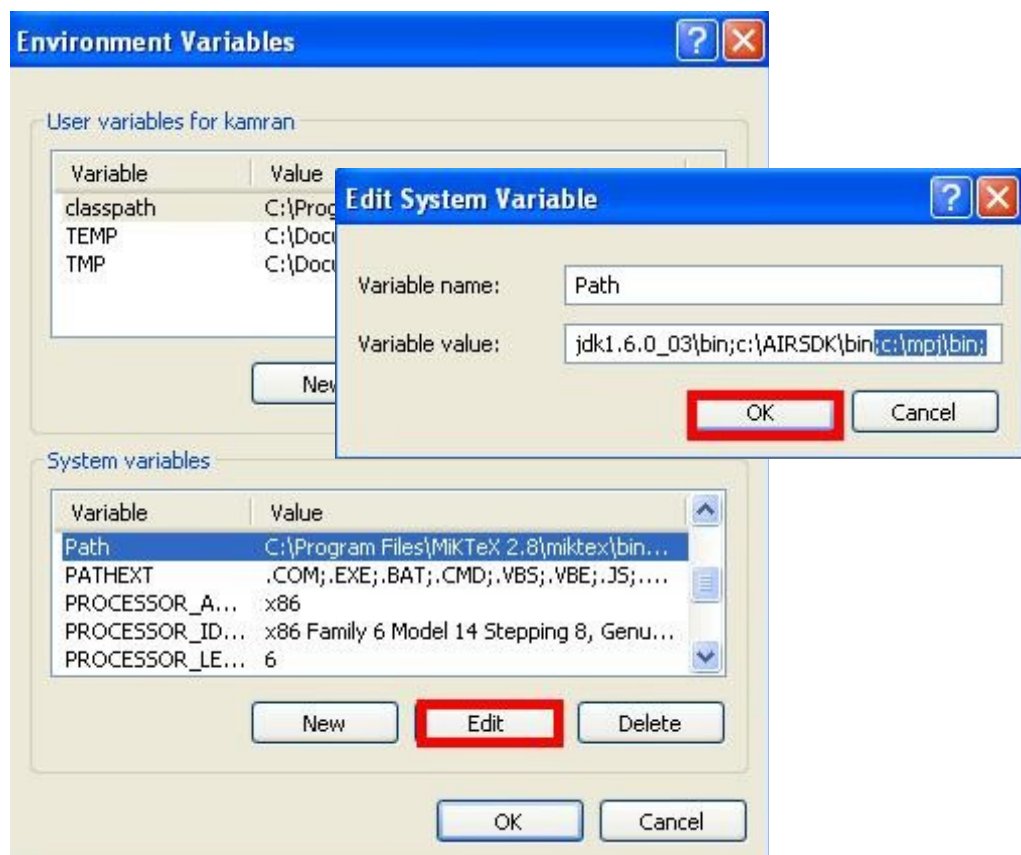
➤ Cấu hình **Multicore**

- Giải nén file MPJ Express và lưu lại với tên ở ổ C:\mpj
- Cài đặt biến môi trường:
 - MPJ_HOME là C:\mpj
 - PATH là C:\mpj\bin

- ♦ MPJ_HOME là C:\mpj



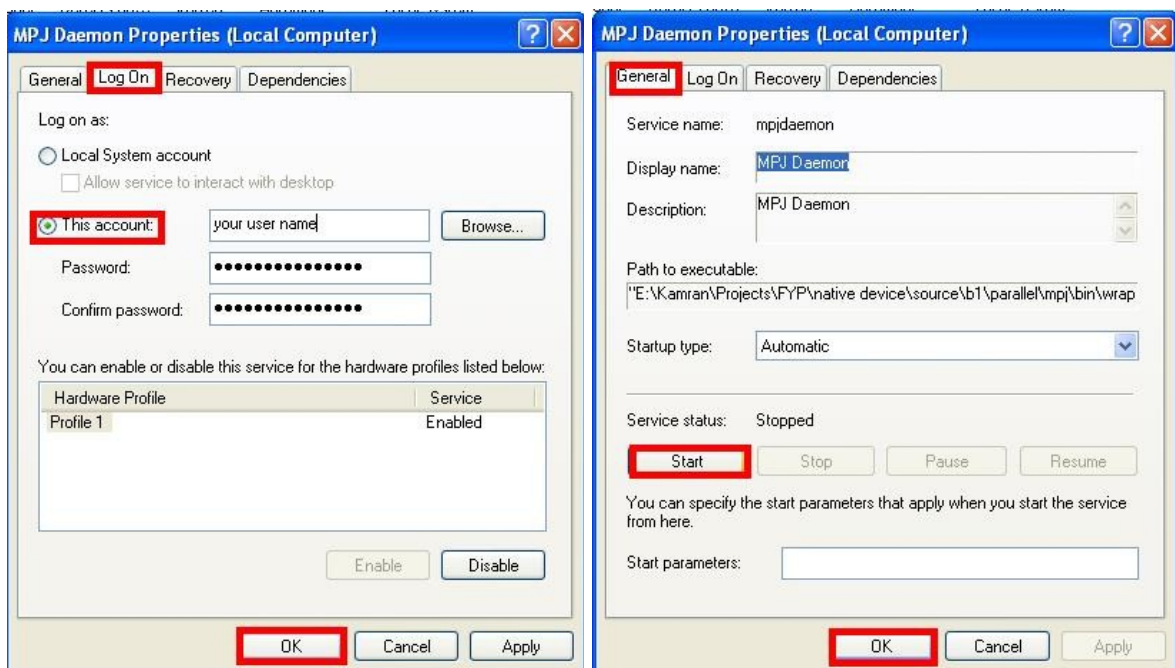
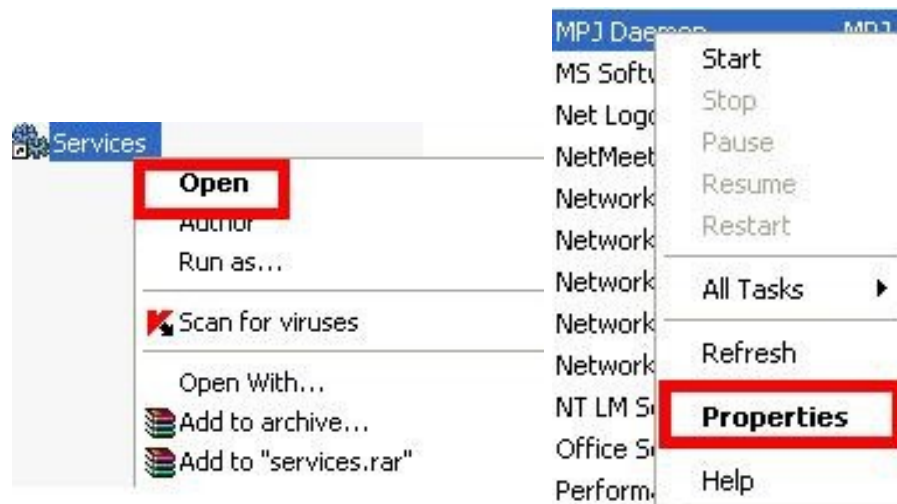
- ♦ PATH là C:\mpj\bin



- Test thử một bộ test trong file MPJ Express bằng lệnh trong cmd
`mpjrun.bat -np 2 -jar %MPJ_HOME%/lib/test.jar`
- Để thực thi một chương trình **HelloWorld.java** thực hiện 2 bước:
 - o Biên dịch chương trình:
`javac -cp .;%MPJ_HOME%/lib/mpj.jar HelloWorld.java`
 - o Thực thi chương trình với 2 tiến trình:
`mpjrun.bat -np 2 HelloWorld`

➤ **Cấu hình Clusters**

- Giải nén file MPJ Express và lưu lại với tại ổ C:\mpj
- Cài đặt biến môi trường như cấu hình Multi-core:
 - o MPJ_HOME là C:\mpj
 - o PATH là C:\mpj\bin
- Tạo folder **mpj-user** chứa các chương trình MPJ Express cần thực thi.
- Tạo file **machines** chứa địa chỉ các máy node .
- Start Service MPJ Daemons:
 - o Run `%MPJ_HOME%/bin/installmpjd-windows.bat` hoặc chạy file `C:\mpj\bin\installmpjd-windows.bat` với “Run as Administrator”.
 - o Vào Control-Panel->Administrative Tools -> Services -> MPJ Daemon và Start the Service. Khi Start Service phải chọn “Log on as” với tài khoản của administrator.



- Để thực thi một chương trình **HelloWorld.java** thực hiện 2 bước:

- o Biên dịch chương trình:

```
javac -cp .;%MPJ_HOME%/lib/mpj.jar HelloWorld.java
```

- o Thực thi chương trình với 2 tiến trình:

```
mpjrun.bat -np 2 -dev niodev HelloWorld
```