

# MỘT SỐ VẤN ĐỀ VÀ CÁCH GIẢI QUYẾT TRONG LẬP TRÌNH ỨNG DỤNG CHUYỂN ĐỔI GIỮA HAI HỆ TOẠ ĐỘ WGS84 VÀ VN2000 TRÊN NỀN TẢNG ANDROID

Bùi Ngọc An  
Trường Đại học Thủy lợi

## Tóm tắt

*Việc chuyển đổi giữa các hệ thống toạ độ đóng vai trò quan trọng trong công tác khảo sát thiết kế công trình, thành lập bản đồ địa hình - địa chính, bản đồ chuyên đề, viễn thám, dẫn đường,... Những năm gần đây, việc tính toán chuyển đổi toạ độ chủ yếu được thực hiện trên máy tính để bàn hoặc Laptop, điều này khiến cho công việc không linh hoạt và gặp khó khăn trong công tác ngoại nghiệp. Để giải quyết vấn đề này, đồng thời giúp con người có thể làm việc mọi lúc mọi nơi và nâng cao năng suất lao động thì việc viết ra các phần mềm tính toán trên các thiết bị di động thông minh là rất cần thiết. Bài báo này giới thiệu phương pháp xây dựng ứng dụng tính chuyển giữa hai hệ toạ độ WGS84 và VN2000 trên nền tảng Android, đồng thời nêu ra một số vấn đề và hướng giải quyết trong quá trình lập trình ứng dụng chuyển đổi giữa hai hệ toạ độ trên.*

**Từ khóa:** Android; Chuyển đổi; WGS84; VN2000

## Abstract

***Some problems and solutions in programming application of transformation between the WGS84 and VN2000 co-ordinate on Android platform.***

*Transformation between co-ordinate systems plays an important role in surveying, creating topographic - cadastral maps, thematic maps, remote sensing and navigation. In recent years, this task has been mainly done on PC or laptop making field work inflexible and difficult. To help people be able to work anytime, anywhere and also improve labor productivity, developing application software of calculation on smart mobile devices is essential. This paper introduces the method of building a transformation application between the two co-ordinate systems WGS84 and VN2000 on the Android platform. The paper also discusses some problems and solutions in the process of application development.*

**Keywords:** Android; Transformation; WGS84; VN2000

## 1. Đặt vấn đề

Tính chuyển giữa hai hệ toạ độ WGS84 và VN2000 đóng vai trò quan trọng trong các công việc liên quan tới trắc địa - bản đồ, quản lý đất đai như: tìm mốc ngoài thực địa, xử lý tính toán số liệu đo GPS, hiển thị vị trí ranh giới lô đất theo toạ độ ghi trên giấy chứng nhận quyền sử dụng đất trên Google Map hoặc cán bộ thực địa muốn xác định vị trí cần

lấy mẫu trên bản đồ sẽ nằm ở đâu ngoài thực địa và nhiều mục đích sử dụng bản đồ khác,... Hiện nay, có nhiều phần mềm trên máy tính tính chuyển giữa hai hệ toạ độ, bên cạnh việc sử dụng máy tính, việc chuyển đổi này có thể thực hiện trên các ứng dụng điện thoại. Tuy nhiên, hầu hết các phần mềm trên điện thoại chưa đủ chức năng chuyển đổi đối với trường hợp múi chiếu 3° hoặc 6° đồng thời phương thức hiển thị cũng như nhập dữ liệu còn

khó khăn cho người dùng. Do vậy, việc xây dựng một ứng dụng có đầy đủ chức năng chuyển đổi và hiển thị điểm trên bản đồ Google Map là cần thiết và có ý nghĩa lớn trong thực tiễn sản xuất. Bài báo trình bày phương pháp tính chuyển giữa hai hệ tọa độ WGS84 và VN2000 trên nền tảng Android.

**2. Phương pháp nghiên cứu**

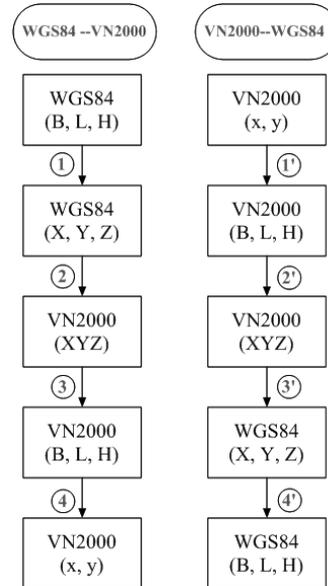
**2.1. Phương pháp tính chuyển giữa hai hệ tọa độ**

Bước 1: tính chuyển từ tọa độ trắc địa (B, L, H) sang tọa độ vuông góc không gian (X, Y, Z) trong hệ WGS84.

Bước 2: tính chuyển từ hệ WGS84 quốc tế về Elipsoid quy chiếu trong Hệ VN2000.

Bước 3: tính chuyển sang hệ tọa độ trắc địa sau khi đã có tọa độ vuông góc không gian VN2000.

Bước 4: tính chuyển sang hệ tọa độ phẳng với kinh tuyến trục của từng địa phương.



**Hình 1: Các bước tính chuyển giữa hai hệ tọa độ**

Quy trình tính chuyển VN2000 → WGS84 sẽ ngược lại so với tính chuyển WGS84→VN2000.

Để tính chuyển từ tọa độ trắc địa (B, L, H) sang tọa độ vuông góc không gian (X, Y, Z) và ngược lại sử dụng các công thức (2.1) và (2.2).

$$\begin{aligned} X &= (N + H) \cos B \cos L \\ Y &= (N + H) \cos B \sin L \\ Z &= \left( \frac{b^2}{a^2} N + H \right) \sin B \end{aligned} \tag{2.1}$$

$$\begin{aligned} B_{i+1} &= \arctg \left[ \frac{Z + e^2 N_i^2 \sin B_i}{\sqrt{X^2 + Y^2}} \right]; N_{i+1} = \frac{a}{\sqrt{1 - e^2 \sin^2 B_{i+1}}} \\ L &= \arctg \frac{Y}{X} \\ H &= \frac{Z}{\sin B_s} - N_s (1 - e^2) \end{aligned} \tag{2.2}$$

Với: a, b: Bán trục lớn, bán trục nhỏ Elipsoid WGS84

$$\begin{aligned} e &= \sqrt{\frac{a^2 - b^2}{a^2}}; e' = \sqrt{\frac{a^2 - b^2}{b^2}}; \\ N &= \frac{a}{\sqrt{1 - e^2 \sin^2 B}} \\ B_0 &= \arctg \left[ \frac{Z(1 + e'^2)}{\sqrt{X^2 + Y^2}} \right]; N_0 = \frac{a}{\sqrt{1 - e^2 \sin^2 B_0}} \end{aligned}$$

## Nghiên cứu

### **Chú ý:**

- Trường hợp không có H, thì coi H = 0;

- B được tính lặp đến khi nào kết quả giữa 2 lần gần nhau nhất giống nhau thì dừng lại (*thông thường lặp 5 lần*).

Để tính chuyển tọa độ vuông góc không gian (X, Y, Z) trong hệ WGS84 quốc tế về tọa độ vuông góc không gian (X, Y, Z) với Elipsoid qui chiếu trong Hệ VN2000 và ngược lại sử dụng công thức (2.3):

$$\begin{cases} X = \Delta X_0 + k(X' + \varepsilon_0 Y' - \psi_0 Z') \\ Y = \Delta Y_0 + k(-\varepsilon_0 X' + Y' + \omega_0 Z') \\ Z = \Delta Z_0 + k(\psi_0 X' - \omega_0 Y' + Z') \end{cases} \quad (2.3)$$

### Với:

- X, Y, Z là tọa độ vuông góc không gian trên hệ tọa độ cần tính chuyển sang, đơn vị là mét;

- X', Y', Z' là tọa độ vuông góc không gian trên hệ tọa độ tính chuyển sang, đơn vị là mét;

-  $\Delta X_0, \Delta Y_0, \Delta Z_0$  là tham số dịch chuyển gốc tọa độ, đơn vị là mét;

-  $\varepsilon_0, \omega_0, \psi_0$  là ba góc xoay trục tọa độ (góc xoay Öle), đơn vị là giây;

- k là hệ số tỷ lệ chiều dài giữa hai hệ.

**Bảng 1. Các tham số tính chuyển giữa hai hệ tọa độ WGS84 và VN2000 [3]**

Tính chuyển từ WGS84 sang VN2000		Tính chuyển từ VN2000 sang WGS84	
Tham số	Giá trị	Giá trị	Đơn vị
$\Delta X_0$	191,90441429	-191,90441429	Mét
$\Delta Y_0$	39,30318279	-39,30318279	Mét
$\Delta Z_0$	111,45032835	-111,45032835	Mét
$\varepsilon_0$	0,00928836	-0,00928836	Giây
$\omega_0$	-0,01975479	0,01975479	Giây
$\psi_0$	0,00427372	-0,00427372	Giây
k	0,999999747093722	1,000000252906278	

Để tính chuyển từ tọa độ trắc địa (B, L, H) sang hệ tọa độ phẳng với kinh tuyến trục của từng địa phương (x, y) và ngược lại thì sử dụng các công thức (2.4) và (2.5).

$$\begin{cases} x = m_0 (S + a_2 l^2 + a_4 l^4 + a_6 l^6 + \dots) \\ y = m_0 (b l + b_3 l^3 + b_5 l^5 + \dots) \end{cases} \quad (2.4)$$

$$\begin{cases} B = A_0 + A_2 y^2 + A_4 y^4 + A_6 y^6 + \dots \\ L = B_1 y + B_3 y^3 + B_5 y^5 + \dots \end{cases} \quad (2.5)$$

Với:  $l = L$  – Kinh tuyến trục của địa phương

$$S = a(1 - e^2) \left[ \beta_0 B - \frac{\beta_2}{2} \text{Sin}2B + \frac{\beta_4}{4} \text{Sin}4B - \frac{\beta_6}{6} \text{Sin}6B + \frac{\beta_8}{8} \text{Sin}8B \right]$$

$$\begin{cases} \beta_0 = 1 + \frac{3}{4}e^2 + \frac{45}{64}e^4 + \frac{175}{256}e^6 + \frac{11025}{16384}e^8 \\ \beta_2 = \frac{3}{4}e^2 + \frac{15}{16}e^4 + \frac{525}{512}e^6 + \frac{2205}{2048}e^8 \\ \beta_4 = \frac{15}{64}e^4 + \frac{105}{256}e^6 + \frac{2205}{4096}e^8 \\ \beta_6 = \frac{35}{512}e^6 + \frac{315}{2048}e^8 \\ \beta_8 = \frac{315}{16384}e^8 \end{cases}$$

$$a_2 = \frac{1}{2} N \text{Sin}B \text{Cos}B$$

$$a_4 = \frac{1}{24} N \text{Sin}B \text{Cos}^3 B (5 - t^2 + 9\eta^2 + 4\eta^4)$$

$$a_6 = \frac{1}{720} N \text{Sin}B \text{Cos}^5 B (61 - 58t^2 + t^4 + 720\eta^2 - 330\eta^2 t^2)$$

$$a_8 = \frac{1}{40320} N \text{Sin}B \text{Cos}^7 B (1385 - 311t^2 + 543t^4 - t^6)$$

$$b_1 = N \text{Cos}B$$

$$b_3 = \frac{1}{6} N \text{Cos}^3 B (1 - t^2 + \eta^2)$$

$$b_5 = \frac{1}{120} N \text{Cos}^5 B (5 - 18t^2 + t^4 + 14\eta^2 - 58\eta^2 t^2)$$

$$b_7 = \frac{1}{5040} N \text{Cos}^7 B (61 - 479t^2 + 179t^4 - t^6)$$

$$\begin{cases} t = \text{tg}B \\ \eta = e' \text{Cos}B \end{cases}$$

- $m_0 = 0.9999$  với múi  $3^\circ$
- $m_0 = 0.9996$  với múi  $6^\circ$
- y sẽ được cộng thêm 500000 mét.
- y được tính bằng:  $y = \frac{y}{m_0}$

## 2.2. Ngôn ngữ lập trình và phần mềm sử dụng lập trình

**Ngôn ngữ lập trình:** Có nhiều ngôn ngữ được sử dụng để lập trình ứng dụng di động như: Python, Java, Kotlin, C,... Tuy nhiên, nghiên cứu này lựa chọn Java vì đây là ngôn ngữ mã nguồn mở, dễ học, API phong phú, nhiều hỗ trợ từ cộng đồng sử dụng.

**Phần mềm sử dụng lập trình:** Mặc dù Eclipse dễ cài đặt và sử dụng nhưng lại ít được Google hỗ trợ hơn so với Android studio. Ngoài ra, Android studio có nhiều các module phục vụ lập trình được dễ

$$A_2 = -\frac{V_x^2 \text{tg}B_x}{2N_x^2}$$

$$A_4 = -\frac{A_2}{12N_x^2} (5 + 3t^2 + \eta_x^2 - 9\eta_x^2 t^2 - 4\eta_x^4)$$

$$A_6 = -\frac{A_2}{360N_x^4} (61 + 90t^2 + 45t^4 - 46\eta_x^2 - 252\eta_x^2 t^2 - 90\eta_x^4 t^4)$$

$$A_8 = -\frac{A_2}{20160N_x^6} (1385 + 3633\text{tg}^2 B_x + 4095\text{tg}^6 B_x)$$

$$B_1 = \frac{1}{N_x \text{Cos} B_x}$$

$$B_3 = -\frac{B_1}{6N_x^2} (1 + 2t^2 + \eta_x^2)$$

$$B_5 = \frac{B_1}{120N_x^4} (5 + 28t^2 + 24t^4 + 6\eta_x^2 + 8\eta_x^2 t^2)$$

$$B_7 = -\frac{B_1}{5040N_x^6} (61 + 662t^2 + 1320t^4 + 720t^6)$$

$$\begin{cases} V_x = 1 + \eta_x^2 \\ N_x = \frac{a}{\sqrt{1 - e^2 \sin^2 B_x}} \\ B_x = B_0 + \sin(2B_0) \cdot (k_0 + k_2 \sin^2 B_0 + k_4 \sin^4 B_0 + k_6 \sin^6 B_0) \end{cases}$$

$$\begin{cases} B_0 = \frac{x}{m_0 a_0 (1 - e^2)} \\ a_0 = 1 + \frac{3}{4}e^2 + \frac{45}{64}e^4 + \frac{350}{512}e^6 + \frac{11025}{16384}e^8 \\ k_0 = \frac{1}{2} \left( \frac{3}{4}e^2 + \frac{45}{64}e^4 + \frac{350}{512}e^6 + \frac{11025}{16384}e^8 \right) \\ k_2 = -\frac{1}{3} \left( \frac{63}{64}e^4 + \frac{1108}{512}e^6 + \frac{58293}{16384}e^8 \right) \\ k_4 = \frac{1}{3} \left( \frac{604}{512}e^6 + \frac{58293}{16384}e^8 \right) \\ k_6 = -\frac{1}{3} \left( \frac{26328}{16384}e^8 \right) \end{cases}$$

dùng, thuận tiện, hiệu suất cao. Do đó, phần mềm Android studio được ưu tiên sử dụng trong lập trình ứng dụng Android.

## 2.3. Một số vấn đề gặp phải trong quá trình lập trình và cách giải quyết

```
//Hàm chuyển từ XYZ (VN2000) --> XYZ (WGS84)
//Chuyển từ Z2000 về Z84
public double Z2K2Z84(double X2000, double Y2000, double Z2000) {
    double Z84 = (-dx0) + kphay*(X2000+(-e0)*Y2000-(-phi0)*Z2000);
    return Z84;
}
//Chuyển từ Y2000 về Y84
public double Y2K2Y84(double X2000, double Y2000, double Z2000) {
    double Y84 = (-dy0) + kphay*(Y2000-(-e0)*X2000+(-w0)*Z2000);
    return Y84;
}
//Chuyển từ Z2000 về Z84
public double Z2K2Z84(double X2000, double Y2000, double Z2000) {
    double Z84 = (-dz0) + kphay*((-phi0)*X2000-(-w0)*Y2000+Z2000);
    return Z84;
}
//Kết thúc hàm chuyển từ XYZ(2000) --> XYZ(WGS84)
```

**Hình 2: Ví dụ về đơn giản hoá kết quả trả về**

**Vấn đề thứ nhất:** Khi tính toán liên quan tới toạ độ, thông thường kết quả trả về ở dạng mảng hai chiều (do toạ độ bao

## Nghiên cứu

gồm 2 hoặc 3 giá trị X, Y, Z). Như vậy, khi viết các hàm con tính toán, việc trả về giá trị mảng (array) sẽ gặp nhiều khó khăn và phức tạp, đồng thời việc đọc giá trị trả về của hàm này để dùng cho hàm con khác cũng gặp nhiều rắc rối. Để giải quyết vấn đề này, cách đơn giản nhất là viết hàm con tính toán cho từng giá trị tọa độ, lúc này giá trị trả về chỉ ở dạng đơn lẻ, thuận tiện cho các việc xuất hoặc lấy dữ liệu vào các hàm khác.

*Vấn đề thứ hai:* Việc tính toán toán tử lũy thừa trong java cũng gặp phải một số khó khăn do phải khai báo biến cho toán tử thừa số và toán tử số mũ. Vì vậy, cần viết một hàm để việc tính toán lũy thừa được dễ dàng hơn.

*Vấn đề thứ ba:* Thư viện toán học phục vụ tính toán lượng giác trong Java làm việc với giá trị là radian, trong khi

quá trình thực hiện tính toán và kết quả đưa ra dưới định dạng độ, phút, giây. Do đó, việc tính toán với giá trị độ phút giây cần phải chuyển sang định dạng radian. Mặt khác, khi đã có kết quả, cần chuyển đổi để hiển thị dưới dạng độ, phút, giây. Trong quá trình chuyển đổi để hiển thị sẽ gặp phải một số trường hợp sau:

- Hiển thị:  $15^{\circ} 59' 60''$  hay  $15^{\circ} 60' 00''$  thay vì  $16^{\circ} 00' 00''$

- Khi lấy phần nguyên của độ sẽ bị làm tròn lên. Ví dụ 15.6 sẽ bị làm tròn lên 16. Để giải quyết, cần chuyển đổi dữ liệu qua lại từ double về int với câu lệnh: **int biến integer = (int) biến double**. Thao tác này được gọi là ép kiểu dữ liệu.

- Để độ phút giây được thể hiện đủ 3 chữ số phần độ, 2 chữ số phần phút, 4 chữ số phần giây thì cần sử dụng hàm *String.format()*.

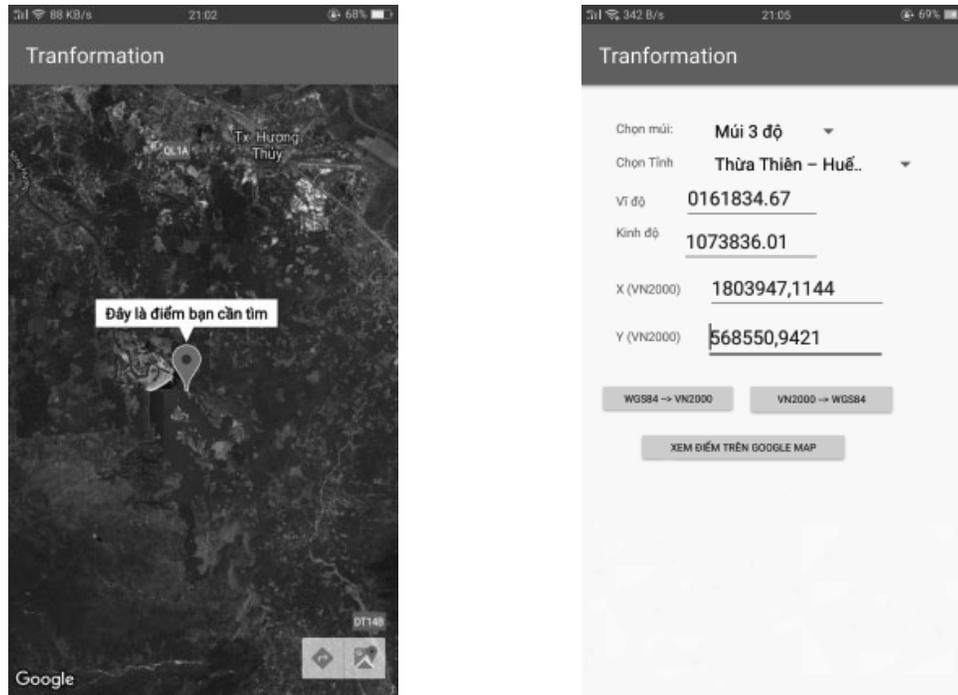
```
public double ex (double coso, double somu){
    double luythua = Math.pow(coso, somu);
    return luythua;
}
double N = a/(Math.sqrt(1-(ex(e, somu: 2)*ex(Math.sin(B), somu: 2)))));
```

**Hình 3: Hàm con tính toán lũy thừa**

```
// Hàm hiển thị giá trị độ phút giây
public String hienthi (double goc) {
    double do1 = goc;
    int do2 = (int)do1;
    double d = (double)do2;
    double phut1 = (goc-d)*60;
    int phut2 = (int)phut1;
    double p = (double)phut2;
    double g = (((goc-d)*60)-p)*60;
    if(g >= 59.9999) {
        g = g - 59.9999;
        p = p + 1;
        d = d;
        if (p >= 59.9999) {
            p = p - 59.9999;
            d = d + 1;
            g = g;
            String hienthi = String.format("%3.0f",d) + " " + String.format("%2.0f",p) + " " + String.format("%6.3f",g);
        }
    }
    String hienthi = String.format("%3.0f",d) + " " + String.format("%2.0f",p) + " " + String.format("%6.3f",g);
    return hienthi;
}
// Kết thúc hàm đổi độ phút giây
```

**Hình 4: Hàm định dạng hiển thị độ phút giây**

### 3. Kết quả và thảo luận



**Hình 5: Kết quả xây dựng ứng dụng**

Xây dựng được ứng dụng tính chuyển giữa hai hệ tọa độ VN2000 và WGS84 với các trường hợp múi 3 độ và 6 độ, kèm theo kinh tuyến trực của từng tỉnh thành, hiển thị đúng format và đảm bảo không bị lỗi định dạng; hiển thị điểm tính chuyển lên bản đồ Google map.

Kết quả nghiên cứu được đánh giá độ chính xác bằng cách so sánh kết quả tính

chuyển 5 điểm tại khu vực Thừa Thiên Huế với phần mềm Geotool của Bộ Tài nguyên và Môi trường. Kết quả tại Bảng 1 cho thấy: ứng dụng hoàn toàn đảm bảo độ chính xác và có thể sử dụng rộng rãi; kết quả tính chuyển giữa hai phần mềm có tồn tại sai số nhỏ là do quá trình tính lặp khác nhau, tuy vậy điều này không ảnh hưởng lớn tới kết quả và độ chính xác của kết quả tính chuyển.

**Bảng 2. Bảng so sánh kết quả tính chuyển**

Điểm				Ứng dụng		Geotool		Sai số	
	Vĩ độ			X	Y	X	Y	X	Y
	Kinh độ								
1	16	19	10.10	1805033.477	567721.359	1805033.478	567721.359	0.001	0.000
	107	38	8.18						
2	16	19	24.75	1805480.041	566499.179	1805480.041	566499.179	0.000	0.000
	107	37	27.05						
3	16	18	43.63	1804216.783	566706.078	1804216.783	566706.079	0.000	0.001
	107	37	33.89						
4	16	18	34.67	1803947.232	568550.826	1803947.232	568550.826	0.000	0.000
	107	38	36.01						
5	16	18	8.11	1803125.348	566847.263	1803125.348	566847.263	0.000	0.000
	107	37	38.53						

## Nghiên cứu

Mặc dù có khả năng chuyển đổi qua lại giữa hai hệ tọa độ nhưng ứng dụng này chưa cho phép lưu lại lịch sử chuyển đổi của các điểm hoặc chuyển đổi cùng lúc nhiều điểm dưới dạng tệp. Mặt khác, trong bài báo, cần phải xem xét sự khác biệt giữa hệ tọa độ địa lý và hệ tọa độ trắc địa khi xác định tọa độ từ Google map để tiến hành chuyển đổi về VN2000. Giữa hai hệ tọa độ này có sự khác biệt nhau về vĩ độ; do vậy để nâng cao độ chính xác cần sử dụng cả dữ liệu đo trọng lực tại từng khu vực tính chuyển.

### **4. Kết luận**

Bài báo đã xây dựng thành công ứng dụng chuyển đổi qua lại giữa hai hệ tọa độ WGS84 và VN2000 cho thiết bị di động thông minh sử dụng hệ điều hành Android. Ứng dụng này có ưu điểm dễ dàng nhập liệu và chuyển đổi được cả hai trường hợp múi chiếu ( $3^\circ$  và  $6^\circ$ ). Kết quả nghiên cứu cũng đã đề cập đến ba vấn đề gặp phải trong quá trình lập trình ứng dụng và đưa ra phương hướng giải quyết. Tuy nhiên, ứng dụng cũng còn tồn tại một

số hạn chế về các tính năng chuyển đổi đồng thời số lượng lớn các điểm, sai lệch về hệ tọa độ địa lý và hệ tọa độ trắc địa. Trong tương lai cần nghiên cứu thêm các tính năng như: hiển thị các lớp bản đồ chuyên đề lên Google Map, thu thập dữ liệu theo hành trình di chuyển hoặc quản lý bản đồ cá nhân trên ứng dụng.

### **TÀI LIỆU THAM KHẢO**

- [1]. Android Studio Tutorial: [https://www.tutorialspoint.com/android/android\\_studio.htm](https://www.tutorialspoint.com/android/android_studio.htm);
  - [2]. Quyết định số 83/2000/QĐ-TTg ngày 12/7/2000 của Thủ tướng Chính phủ về việc *áp dụng Hệ quy chiếu và Hệ tọa độ quốc gia VN2000*.
  - [3]. Quy định số 1123/ĐĐBBĐ-CNTĐ ngày 26 tháng 10 năm QĐ BTNMT về *sử dụng các tham số tính chuyển từ hệ tọa độ quốc tế WGS84 sang hệ tọa độ quốc gia VN2000 và ngược lại*.
  - [4]. Thông tư số 973/2001/TT-TCĐC ngày 20/6/2001 của Tổng cục Địa chính *hướng dẫn áp dụng hệ quy chiếu và hệ tọa độ quốc gia VN2000*.
- BBT nhận bài: 18/5/2020; Phản biện xong: 11/6/2020; Chấp nhận đăng: 26/6/2020