

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO GIỮA KỲ
MÔN PHÁT TRIỂN ỨNG DỤNG DI ĐỘNG

TÌM HIỂU SENSOR **VÀ XÂY DỰNG ỨNG DỤNG MINH HOẠ**

Người hướng dẫn : **GV. LÊ VĂN VANG**

Người thực hiện : **Trần Quang Vinh – 51900781**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO GIỮA KỲ
MÔN PHÁT TRIỂN ỨNG DỤNG DI ĐỘNG

TÌM HIỂU SENSOR **VÀ XÂY DỰNG ỨNG DỤNG MINH HOẠ**

Người hướng dẫn : **GV. LÊ VĂN VANG**

Người thực hiện : **Trần Quang Vinh – 51900781**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn chân thành đến GV. Lê Văn Vang, người thầy đã luôn hỗ trợ và hướng dẫn tận tình cho em trong suốt quá trình nghiên cứu và hoàn thành bài báo cáo giữa kỳ này.

Bên cạnh đó, em cũng xin gửi lời cảm ơn chân thành đến khoa Công nghệ thông tin, trường Đại học Tôn Đức Thắng, vì đã tạo điều kiện cho em được học tập và nghiên cứu môn học này. Khoa đã luôn sẵn sàng chia sẻ để đem tới những kiến thức bổ ích cho em, cũng như những kinh nghiệm trong quá trình tra cứu và tham khảo tài liệu. Điều đó đã giúp ích, không chỉ cho việc thực hiện và hoàn thành bài báo cáo, mà còn giúp ích cho việc học tập, rèn luyện trong suốt quá trình thực hành tại trường Đại học Tôn Đức Thắng nói chung.

Cuối cùng, sau khoản thời gian học tập trên lớp, em đã hoàn thành bài báo cáo giữa kỳ này, nhờ vào sự hướng dẫn, giúp đỡ và những kiến thức mà quý thầy cô đã mang đến. Do giới hạn về mặt kiến thức và khả năng lý luận, nên em vẫn còn nhiều thiếu sót và hạn chế. Kính mong sự chỉ dẫn và đóng góp từ quý thầy cô giảng viên để bài báo cáo này của em được hoàn thiện hơn. Hơn nữa, nhờ những lời góp ý từ quý thầy cô nhà trường và bạn bè, em sẽ hoàn thành tốt hơn ở những bài báo cáo tiếp theo trong tương lai. Em kính mong quý thầy cô và bạn bè – những người luôn quan tâm và hỗ trợ em – luôn tràn đầy sức khỏe và bình an.

XIN CHÂN THÀNH CẢM ƠN !

BÁO CÁO ĐƯỢC HOÀN THÀNH

TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Em xin cam đoan đây là sản phẩm nghiên cứu của riêng em và được sự hướng dẫn của GV. Lê Văn Vang. Các nội dung nghiên cứu, kết quả trong bài báo cáo này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá, đã được chính tác giả thu nhập từ các nguồn khác nhau, có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong bài báo cáo còn sử dụng một số nhận xét, đánh giá, cũng như số liệu từ các tác giả khác, cơ quan tổ chức khác, đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào, em xin hoàn toàn chịu trách nhiệm về nội dung báo cáo của mình. Trường Đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền, do em gây ra trong quá trình thực hiện (nếu có).

Tp. Hồ Chí Minh, ngày tháng năm 2023

Tác giả

(Ký tên và ghi rõ họ tên)

Trần Quang Vinh

PHẦN NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần nhận xét và đánh giá của giảng viên hướng dẫn

Tp. Hồ Chí Minh, ngày tháng năm 2023
(Ký tên và ghi rõ họ tên)

Phần nhận xét và đánh giá của giảng viên chấm bài

Tp. Hồ Chí Minh, ngày tháng năm 2023
(Ký tên và ghi rõ họ tên)

TÓM TẮT

Với sự phát triển mạnh mẽ của nền kinh tế, khoa học kỹ thuật ngày nay, một trong những thành tựu về phát minh khoa học làm thay đổi lịch sử, văn hoá và lối sống của con người trong giai đoạn cuối thế kỷ XIX, đầu thế kỷ XX, đó là thiết bị điện thoại di động.

Vào những năm 70 của thế kỷ XIX, chiếc điện thoại đầu tiên ra đời, mở ra kỷ nguyên phát triển mới nhằm thay thế phương thức liên lạc cũ là điện báo và thư từ. Năm 1973, chiếc điện thoại di động cầm tay đầu tiên được ra mắt bởi nhà khoa học *Martin Cooper*, đánh dấu một bước ngoặt mới trong việc phát triển về mặt công nghệ viễn thông. Cùng với sự phát triển vượt bậc của mạng *Internet*, nhiều công nghệ mới được tích hợp vào chiếc điện thoại di động làm bùng lên về khái niệm *smartphone* trên khắp thế giới, tạo nên một kỷ nguyên về điện thoại di động thông minh được lưu hành cho đến ngày nay. Nhiều “ông lớn” trong ngành nghiên cứu và sản xuất thiết bị điện thoại di động nổi lên ngày càng nhiều, một trong những cái tên nổi bật không thể kể đến, đó là *Apple*, *SamSung* và *Nokia*,...

Nắm bắt được những vấn đề trên và nhận thức được tầm quan trọng, cũng như sức ảnh hưởng lớn về thiết bị điện thoại di động thông minh và các công nghệ được tích hợp bên trong nó. Chính vì vậy, bài báo cáo giữa kỳ được chọn với đề tài : ***“Tìm hiểu Sensor và xây dựng ứng dụng minh hoạ”***.

Với kinh nghiệm và năng lực hiện tại, em mong muốn được tìm hiểu, tra cứu và đưa ra những nghiên cứu của mình về những công nghệ có trong một chiếc điện thoại thông minh. Từ những nghiên cứu về các nền tảng ứng dụng hiện tại và thông qua khảo sát các công nghệ của một chiếc điện thoại di động, Bài báo cáo được xây dựng dựa trên những kiến thức cơ bản được tìm hiểu và trích lọc cụ thể về các công nghệ của thiết bị điện thoại. Từ đó, áp dụng những kiến thức và ứng dụng thực tế, xây dựng nên một ứng dụng minh hoạ về một trong những công nghệ có trong điện thoại di động thông minh.

MỤC LỤC

LỜI CẢM ƠN	2
PHẦN NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN	4
TÓM TẮT	5
MỤC LỤC	6
DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT	7
DANH MỤC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ	7
CHƯƠNG I – TỔNG QUAN VỀ ĐỀ TÀI	8
1.1 Tổng quan về Sensor	8
1.1.1 Thuật ngữ Sensor	8
1.1.2 Cấu tạo của Sensor	9
1.1.3 Phạm vi ứng dụng của Sensor	10
1.2 Tổng quan về Android Sensors	11
1.2.1 Cảm biến Android (Android Sensors)	11
1.2.2 Hệ tọa độ cảm biến (Sensor Coordinate System)	11
1.2.3 Trình quản lý cảm biến (Sensor Manager)	12
1.3 Các loại Android Sensor phổ biến	13
1.4 Các bước làm việc với Android Sensor	14
CHƯƠNG II – XÂY DỰNG ỨNG DỤNG MINH HOẠ	15
2.1 Đặc tả ứng dụng	15
2.2 Thiết lập ứng dụng	15
2.3 Kết quả và đánh giá ứng dụng	21
TÀI LIỆU THAM KHẢO	22

DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT

- + *API* : *Application Programming Interface* – giao diện lập trình ứng dụng.
- + *Android SDK* : *Software Development Kit* – công cụ phát triển phần mềm của *Android*.
- + *GPS* : *Global Positioning System* – hệ thống định vị toàn cầu.
- + 2D và 3D : *D – Dimensional* – không gian 2 chiều, 3 chiều.

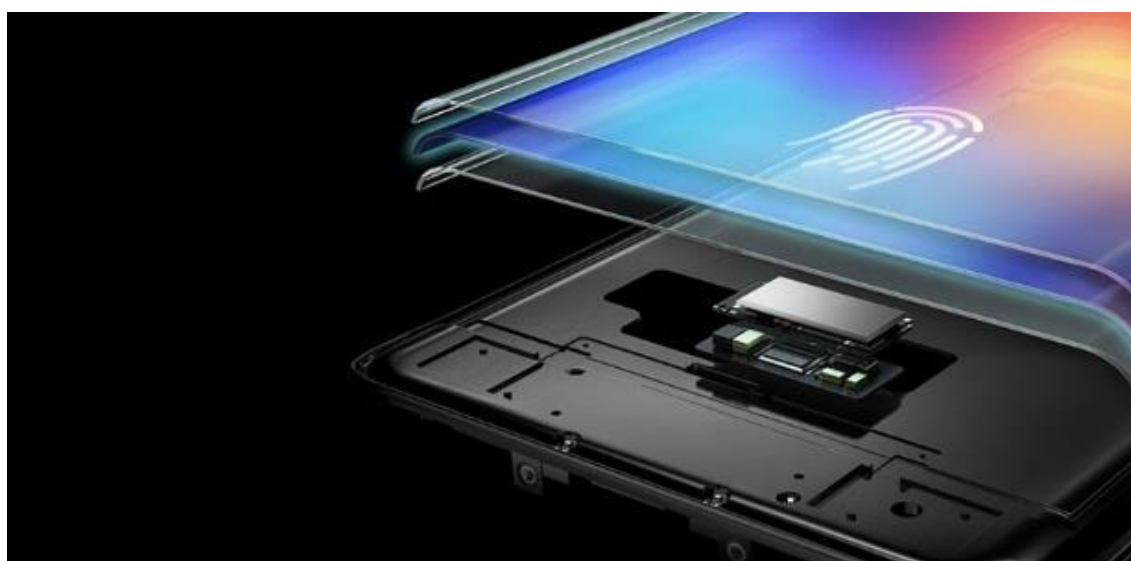
DANH MỤC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

Hình 01 – Ảnh minh họa về cảm biến trong thiết bị điện thoại	8
Hình 02 – Ảnh minh họa về các thiết bị <i>Sensor</i>	9
Hình 03 – Ảnh minh họa về cấu tạo của <i>Sensor</i>	10
Hình 04 – Ảnh minh họa hệ tọa độ cảm biến trong <i>Android</i>	11
Hình 05 – Code giao diện cho ứng dụng minh họa	16
Hình 06 – Code main (1) cho ứng dụng minh họa.....	17
Hình 07 – Code main (2) cho ứng dụng minh họa.....	18
Hình 08 – Code main (3) cho ứng dụng minh họa.....	18
Hình 09 – Code main (4) cho ứng dụng minh họa.....	19
Hình 10 – Code main (5) cho ứng dụng minh họa.....	20
Hình 11 – Code main (6) cho ứng dụng minh họa.....	20
Hình 12 – Ảnh kết quả giao diện ban đầu của ứng dụng.....	21

CHƯƠNG I – TỔNG QUAN VỀ ĐỀ TÀI

1.1 Tổng quan về Sensor

Một trong những khác biệt lớn về công nghệ phát triển của các nền tảng thiết bị điện tử giữa máy tính và điện thoại di động, đó là các phần mềm cảm biến được tích hợp trong các thiết bị điện thoại di động thông minh. Việc sử dụng các phần mềm cảm biến này có thể tạo ra những ứng dụng hữu ích và tiện lợi, đáp ứng rất nhiều nhu cầu cần thiết trong đời sống của con người.



Hình 01 – Ảnh minh họa về cảm biến trong thiết bị điện thoại

1.1.1 Thuật ngữ Sensor

Thuật ngữ *Sensor* – cảm biến – là một thuật ngữ chuyên ngành để chỉ các loại cảm biến, đầu dò, công tắc hoặc một loại thiết bị có thể cảm nhận các vật thể, phi vật thể trong môi trường xung quanh. Một cụm từ để chỉ chung các loại thiết bị có khả năng đo lường và giám sát, cảm nhận những trạng thái hay quá trình vật lý, hoá học, sinh học của môi trường cần khảo sát và biến đổi thành tín hiệu điện để thu nhập thông tin về các trạng thái hay quá trình đó. Các loại *Sensor* chuyên dụng sẽ có khả năng hoạt động trong nhiều môi trường khắc nghiệt khác nhau và những môi trường khó tiếp cận tới.

Bên cạnh đó, các thiết bị *Sensor* còn có các bộ phận xử lý tín hiệu và cho ra các dạng tín hiệu khác nhau để người dùng có thể kết nối với các thiết bị khác, nhằm hỗ trợ trong việc phân tích dữ liệu và điều khiển thiết bị.

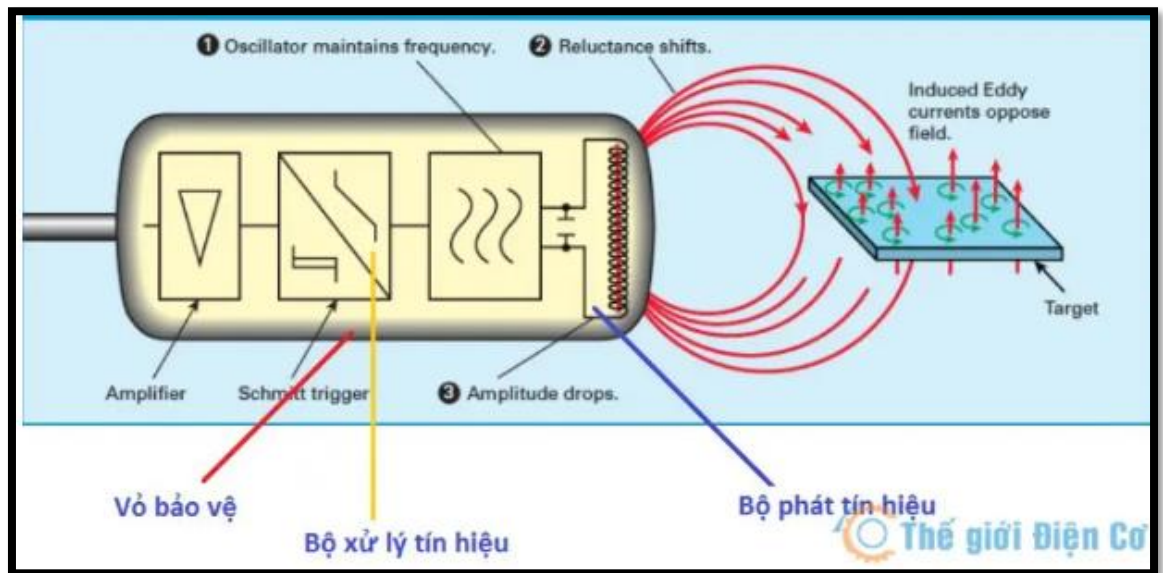


Hình 02 – Ảnh minh họa về các thiết bị Sensor

Việc kết hợp công nghệ *Sensor* đã thay thế hoàn toàn con người trong các ứng dụng đo lường và giám định các yếu tố vật lý, tác động đến quá trình làm việc. với độ chính xác cao, thời gian đáp ứng ngày càng nhanh, kèm theo việc hoạt động khá tốt trong môi trường khó tiếp cận. Những yếu tố đó đã góp phần cho các nhà phát triển cho ra những thiết bị tân tiến nhất để phục vụ không chỉ cho việc sản xuất, chế tạo trong ngành công nghiệp nói chung mà còn đáp ứng một số nhu cầu cần thiết trong đời sống con người hằng ngày nói riêng, như chụp ảnh, quay phim, bảo mật thiết bị hay nhận diện vật thể,...

1.1.2 Cấu tạo của Sensor

Có rất nhiều loại *Sensor* khác nhau trong ngành thiết bị điện tử ngày nay và mỗi loại đều có những tính năng riêng biệt. Nhưng, về mặt kỹ thuật, hầu hết đều có cấu tạo cơ bản giống nhau, được chia làm 3 thành phần chính :



Hình 03 – Ảnh minh họa về cấu tạo của Sensor

Phần 01 : vỏ bảo vệ cảm biến, có kết cấu bằng nhựa hoặc kim loại, tùy vào loại cảm biến và chức năng của nó. Phần vỏ có tác dụng bảo vệ các bộ phận chính bên trong của cảm biến.

Phần 02 : bộ phát tín hiệu, hay còn gọi là bộ phận cảm nhận của cảm biến. Ví dụ, cảm biến siêu âm sẽ phát ra sóng siêu âm, cảm biến nhiệt độ sẽ có phần đầu dò cảm nhận nhiệt độ của vật thể hoặc môi trường xung quanh.

Phần 03 : bộ xử lý tín hiệu, chuyển đổi tín hiệu nhận được từ bộ phát tín hiệu thành tín hiệu điện – còn được gọi là bộ não của cảm biến. Tín hiệu điện có thể là tín hiệu 4-20mA hoặc tín hiệu *ON – OFF*.

1.1.3 Phạm vi ứng dụng của Sensor

Thiết bị cảm biến là một sản phẩm công nghệ tiên tiến được sử dụng trong nhiều lĩnh vực của đời sống xã hội. Các ứng dụng của *Sensor* sẽ tùy thuộc vào môi trường xung quanh mà thiết bị cảm biến này được sử dụng, cách nhận biết dễ dàng và rõ ràng nhất chính là dựa vào việc phân loại các cảm biến theo tên của chức năng mà chúng mang lại. Ví dụ như cảm biến nhiệt độ dùng để đo nhiệt độ, cảm biến áp suất dùng để đo áp suất hay áp lực, cảm biến không khí, cảm biến âm thanh, cảm biến màu sắc, cảm biến tần số, cảm biến từ trường,...

1.2 Tổng quan về Android Sensors

1.2.1 Cảm biến Android (Android Sensors)

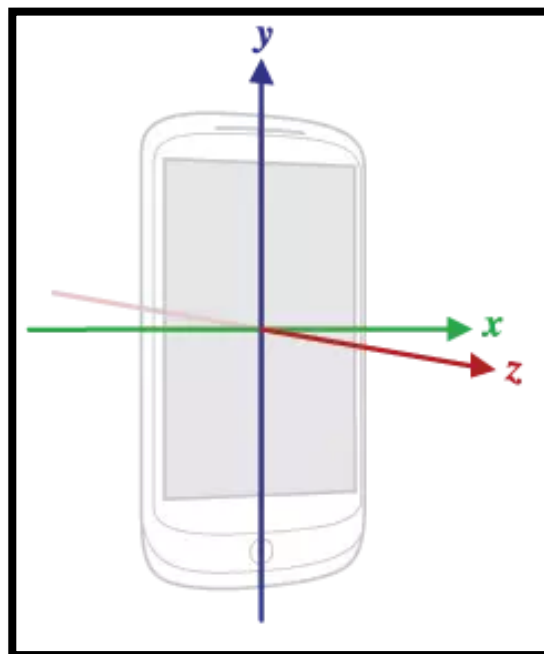
Android Sensor là một loại cảm biến có phần cứng được tích hợp trên các thiết bị điện thoại, có tác dụng phản hồi lại các hành động ở thế giới thực vào trong môi trường ứng dụng chạy trên thiết bị đó. Thiết bị *Android* cung cấp các phương tiện dùng để truy cập phần cứng của các loại cảm biến này, thông qua các *API*, cụ thể như *Multimedia Playback Library* hay *Specific Sensors Library* nhằm giám sát, theo dõi phương hướng và xác định gia tốc của thiết bị điện thoại.

Các hành động trong cảm biến được thực hiện một chiều và chúng chỉ cho phép thực hiện các hành động được mặc định sẵn. *GPS* cũng là một bộ cảm biến nhưng không được tích hợp vào nền tảng cảm biến trong *Android*.

Có 2 loại cảm biến chính :

- + Cảm biến phần cứng : là các thành phần vật lý làm nên điện thoại hoặc máy tính bảng. Chúng lấy dữ liệu bằng cách đo lường giá trị của môi trường xung quanh như gia tốc, từ trường,...
- + Cảm biến phần mềm : không phải là thiết bị vật lý, thay vào đó chúng bắt trước cảm biến phần cứng, lấy dữ liệu từ một hoặc nhiều cảm biến phần cứng.

1.2.2 Hệ tọa độ cảm biến (Sensor Coordinate System)



Hình 04 – Ảnh minh họa về hệ tọa độ cảm biến trong Android

Nhìn chung, khung cảm biến trong các thiết bị *Android* sử dụng hệ tọa độ với 3 trục tiêu chuẩn và giá trị của dữ liệu mà cảm biến thu nhập được sẽ được biểu diễn bởi 3 trục tọa độ này. Đối với hầu hết các loại cảm biến, hệ tọa độ được xác định liên quan đến màn hình của thiết bị khi thiết bị được giữ theo hướng mặc định (phương thẳng đứng).

Khi một thiết bị được giữ theo phương thẳng đứng, trục X sẽ nằm ngang và chỉ sang phải, trục Y thẳng đứng và hướng lên trên, còn trục Z sẽ hướng ra bên ngoài mặt màn hình. Điểm quan trọng nhất của hệ trục tọa độ cảm biến, đó là các trục không bị hoán đổi khi hướng màn hình của thiết bị thay đổi – nghĩa là, hệ tọa độ của cảm biến sẽ không bao giờ thay đổi ngay cả khi thiết bị di chuyển.

1.2.3 *Trình quản lý cảm biến (Sensor Manager)*

Để có thể truy cập vào các cảm biến có sẵn trong thiết bị, *Android* cung cấp cho người dùng một *Sensor Framework*, cho phép người dùng có thể sử dụng nhiều loại cảm biến (phần cứng và phần mềm).

Ngoài ra, còn dùng để kiểm tra xem thiết bị có hỗ trợ cảm biến cần truy cập và sử dụng được hay không, lấy dữ liệu *Sensor*, đăng ký hay hủy đăng ký tiếp nhận sự kiện mà cảm biến thay đổi,...với các *API* bao gồm :

- + *Package* : *android.hardware*
- + *Classes* : *SensorManager* – *android service*
Sensor – *specific sensor*
SensorEvent – *specific event of the sensor = data*

Trong đó, *SensorManager* và *Sensor* là 2 *class* cung cấp cho người dùng các phương thức để có thể truy cập và lấy dữ liệu từ các *Sensor*. Với các *Sensor* là các *chip* cảm ứng nằm trong thiết bị, cung cấp dữ liệu mà thiết bị cảm biến này đo đạc được cho hệ điều hành *Android*. Các *Sensor* này được quản lý chung bởi *SensorManager* – một dịch vụ của hệ thống.

SensorEvent – hệ thống sử dụng *class* này để tạo ra các sự kiện của đối tượng cảm biến, nó bao gồm các thông tin về dữ liệu cảm biến, loại cảm biến phát sinh, độ chính xác của dữ liệu.

SensorEventListener – tạo ra các *callback* để nhận các *SensorEvent* khi giá trị của cảm biến thay đổi.

Thông qua *SensorManager*, người dùng có thể :

- + Lấy danh sách các *Sensor* có trong hệ thống hiện tại.
- + Lấy đối tượng để làm việc trực tiếp với từng *Sensor*.
- + Đăng ký *listener* để xử lý sự kiện do các *Sensor* báo về.

1.3 Các loại Android Sensor phổ biến

Android Sensor được chia làm 3 nhóm chính :

- + Cảm biến gia tốc (cảm biến chuyển động) : cho biết về lực gia tốc và độ quay của thiết bị theo hệ trục tọa độ của cảm biến.
- + Cảm biến vị trí : cho biết về vị trí vật lý của thiết bị.
- + Cảm biến môi trường : cho biết thông tin về nhiệt độ, độ ẩm của môi trường.

Mỗi loại *Sensor* có những đặc điểm vật lý khác nhau, một số thuộc loại *virtual*, tức là kết quả được tính toán hoặc nội suy từ các nguồn khác nhau. Với *Android SDK* không có các *class* định sẵn cho từng loại *Sensor*, mà chỉ có các *TYPE* của *Sensor*, với dữ liệu trả về thuộc dạng *float* (trường hợp cảm biến với 1 đầu ra – cảm biến ánh sáng) hoặc thuộc dạng *float[]* (trường hợp cảm biến với nhiều đầu ra).

Các loại *Sensor* phổ biến hiện được *Android OS* hỗ trợ như :

- + *TYPE_ACCELEROMETER* : cảm biến gia tốc, đơn vị đo là m/s^2 .
- + *TYPE_GRAVITY* : cảm biến hấp dẫn (trọng lực) trong 3D, đơn vị đo là m/s^2 .
- + *TYPE_AMBIENT_TEMPERATURE* : cảm biến nhiệt độ môi trường, đơn vị đo là $^{\circ}C$.
- + *TYPE_GYROSCOPE* : cảm biến tốc độ góc quay trong 3D, đơn vị đo là rad/s .
- + *TYPE_LIGHT* : cảm biến ánh sáng, đơn vị đo là lx .
- + *TYPE_HEART_RATE* : cảm biến nhịp tim (mỗi phút).
- + *TYPE_LINEAR_ACCELERATION* : cảm biến gia tốc tuyến tính.
- + *TYPE_MAGNETIC_FIELD* : cảm biến lực từ tính tròn 3D, đơn vị đo là μT .
- + *TYPE_PRESSURE* : cảm biến áp suất không khí, đơn vị đo là $mbar$.
- + *TYPE_PROXIMITY* : cảm biến khoảng cách đến đối tượng, đơn vị đo là cm .
- + *TYPE_RELATIVE_HUMIDITY* : cảm biến độ ẩm, đơn vị đo là %.
- + *TYPE_ROTATION_VECTOR* : cảm biến xoay.
- + *TYPE_GAME_ROTATION_VECTOR* : cảm biến xoay 2D.
- + *TYPE_SIGNIFICANT_MOTION* : cảm biến chuyển động.

1.4 Các bước làm việc với Android Sensor

Để sử dụng *Sensor*, đầu tiên cần khởi tạo đối tượng của *SensorManager* class :

```
SensorManager sMgr;  
sMgr = (SensorManager)this.getSystemService(SENSOR_SERVICE);
```

Tiếp theo, khởi tạo đối tượng của lớp *Sensor*, bằng cách gọi *getDefaultSensor()* của lớp *SensorManager* để truy cập vào cảm biến :

```
Sensor light;  
light = sMgr.getDefaultSensor(Sensor.TYPE_LIGHT);
```

Sau khi *Sensor* được khai báo, người dùng có thể đăng ký *Sensor listener* khi muốn yêu cầu cần đăng ký sử dụng một loại cảm biến có trong *Android*. Việc này cho phép hệ thống thông báo đến người dùng khi dữ liệu của *Sensor* thay đổi. Với 2 *method* được sử dụng là *onAccuracyChanged()* và *onSensorChanged()*.

Một số lưu ý với *Sensor listener* đó là :

- + Nên đăng ký *Sensor listener* trong hàm *onResume()* và hủy đăng ký trong hàm *onPause()*, để tránh việc sử dụng không cần thiết và tiết kiệm *pin* cho thiết bị.
- + *onSensorChanged()* được gọi khi dữ liệu cảm biến thay đổi và có thể được gọi rất nhiều lần, chính vì vậy, nên hạn chế việc xử lý một cách nhiều *logic* trong hàm này.

```
sMgr.registerListener(this, light, SensorManager.SENSOR_DELAY_NORMAL);  
public void onAccuracyChanged(Sensor sensor, int accuracy) {  
}  
public void onSensorChanged(SensorEvent event) {  
}
```

Có thể lấy danh sách các *Sensor*, trả về một *Sensor List* bằng lệnh :

```
sMgr = (SensorManager)this.getSystemService(SENSOR_SERVICE);  
List<Sensor> list = sMgr.getSensorList(Sensor.TYPE_ALL);  
for(Sensor sensor: list){  
}
```

Tắt *Sensor* trong những tình huống không cần thiết để tránh thiết bị tiêu tốn năng lượng. Xử lý lỗi hoặc thay đổi độ nhạy của thiết bị.

CHƯƠNG II – XÂY DỰNG ỨNG DỤNG MINH HOẠ

2.1 Đặc tả ứng dụng

Tên ứng dụng minh họa : “Lắc số ngẫu nhiên”.

Ứng dụng được xây dựng dựa trên cảm biến gia tốc (*Accelerometer*) có trong *Android*.

Ban đầu, ứng dụng sẽ bắt đầu với con số 0, khi người dùng tác động lên thiết bị như lắc, xoay, sẽ tạo ra 1 lực tác động nhất định.

- + Khi lực tác động này lớn hơn 500 m/s^2 , ứng dụng sẽ sinh ra số ngẫu nhiên từ 1 đến 100 và hiển thị trên màn hình giao diện.
- + Khi lực tác động này nhỏ hơn 500 m/s^2 , ứng dụng sẽ không thay đổi số hiển thị, vẫn giữ nguyên giá trị số ngẫu nhiên trước đó và hiển thị trên màn hình giao diện.

2.2 Thiết lập ứng dụng

Bước [1] : tạo giao diện cho ứng dụng

- + Sử dụng *LinearLayout – vertical* làm *layout* chính cho giao diện.
- + Gồm 3 *TextView*, lần lượt hiển thị nội dung : chủ đề – số ngẫu nhiên – *speed*.


```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/linearLayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="Try to shake the Speed to 500"
        android:textSize="20dp"
        android:textStyle="bold"
        android:textColor="#e74c3c"/>

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="0"
        android:textStyle="bold"
        android:textSize="250dp"/>

    <TextView
        android:id="@+id/textView_speed"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="50dp"
        android:text="Speed : 0"
        android:textStyle="bold"
        android:textSize="30dp"
        android:textColor="#c0392b"/>
</LinearLayout>

```

Hình 05 – Code giao diện cho ứng dụng minh họa

Bước [2] : thiết lập ứng dụng

+ Các thư viện sử dụng trong ứng dụng.

```
package com.example.sensoraccelerometer;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Context;
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.os.Bundle;
import android.util.Log;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

import org.w3c.dom.Text;

import java.util.Random;
```

Hình 06 – Code main (1) cho ứng dụng minh họa

+ Khai báo các biến cần thiết

```
2 usages
public class MainActivity extends AppCompatActivity implements SensorEventListener {

    /*---Khai báo các biến cần thiết---*/
    3 usages
    |   SensorManager sensorManager;
    5 usages
    |   Sensor sensorAccelerometer;
    2 usages
    |   TextView textView;
    3 usages
    |   TextView textView_speed;
    3 usages
    |   LinearLayout linearLayout;
    2 usages
    |   float last_x, last_y, last_z;
    3 usages
    |   long last_time = 0;
}
```

Hình 07 – Code main (2) cho ứng dụng minh họa

- + Khởi tạo hàm *onCreate()*.
- + Xác định các biến với các *id TextView, LinearLayout*.
- + Khởi tạo và đăng ký sử dụng cảm biến của *Android*.

```
/*---main---*/
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    /*---find id---*/
    textView = findViewById(R.id.textView);
    textView_speed = findViewById(R.id.textView_speed);
    linearLayout = findViewById(R.id.linearLayout);

    /*---create and register sensor---*/
    sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
    sensorAccelerometer = sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
    if(sensorAccelerometer == null) {
        Toast.makeText(context, this, text: "Have no support sensor", Toast.LENGTH_SHORT).show();
    } else {
        sensorManager.registerListener(listener: this, sensorAccelerometer, SensorManager.SENSOR_DELAY_NORMAL);
    }
}
```

Hình 08 – Code main (3) cho ứng dụng minh họa

- + Khởi tạo hàm *onSensorChanged()*.
- + Lấy các chỉ số của hệ tọa độ cảm biến trong không gian 3D gán vào các biến *x, y, z*.
- + Xác định thời gian thực tế từ khi tác động lên thiết bị với biến *currentTime*.
- + Khi xác định được thời điểm xảy ra tác động, tiến hành tính toán lực tác động vào thiết bị với các tham số *x, y, z* – tham số của hệ trục tọa độ cảm biến.
- + Điều kiện để ứng dụng sinh ngẫu nhiên 1 số bất kỳ, đó là lực tác động *speed* phải lớn hơn 500 m/s². Với việc sinh số ngẫu nhiên được khởi tạo bằng hàm *randomNumber()*.

```

/*---sensor---*/
@Override
public void onSensorChanged(SensorEvent sensorEvent) {
    sensorAccelerometer = sensorEvent.sensor;
    if(sensorAccelerometer.getType() == Sensor.TYPE_ACCELEROMETER) {
        float x = sensorEvent.values[0];
        float y = sensorEvent.values[1];
        float z = sensorEvent.values[2];

        long currentTime = System.currentTimeMillis();
        if((currentTime - last_time) > 100){
            long time = currentTime - last_time;
            last_time = currentTime;

            float speed = Math.abs(x + y + z - last_x - last_y - last_z)/time * 10000;
            if(speed > 500) {
                randomNumber();
                textView_speed.setText("Speed : " + speed);
            } else {
                textView_speed.setText("Speed : " + speed);
            }
            last_x = x;
            last_y = y;
            last_z = z;
        }
    }
}

@Override
public void onAccuracyChanged(Sensor sensor, int i) {
}

```

Hình 09 – Code main (4) cho ứng dụng minh họa

- + Khởi tạo hàm *randomNumber()*.
- + Thiết lập hiệu ứng chuyển động sinh số cho ứng dụng với tính năng *Animation*.

```

/*---random number---*/
1 usage
private void randomNumber(){
    Random random = new Random();
    int number = random.nextInt( bound: 100) + 1;

    textView.setText("" + number);

    Animation animation = AnimationUtils.loadAnimation( context: this, R.anim.random_number);
    linearLayout.clearAnimation();
    linearLayout.startAnimation(animation);
}
}

```

Hình 10 – Code main (5) cho ứng dụng minh hoạ

- + Tạo *folder Anim* trong thư mục *res* chứa thông số hiệu ứng của ứng dụng với tên *file* được đặt ở đây là “*random_number.xml*”.

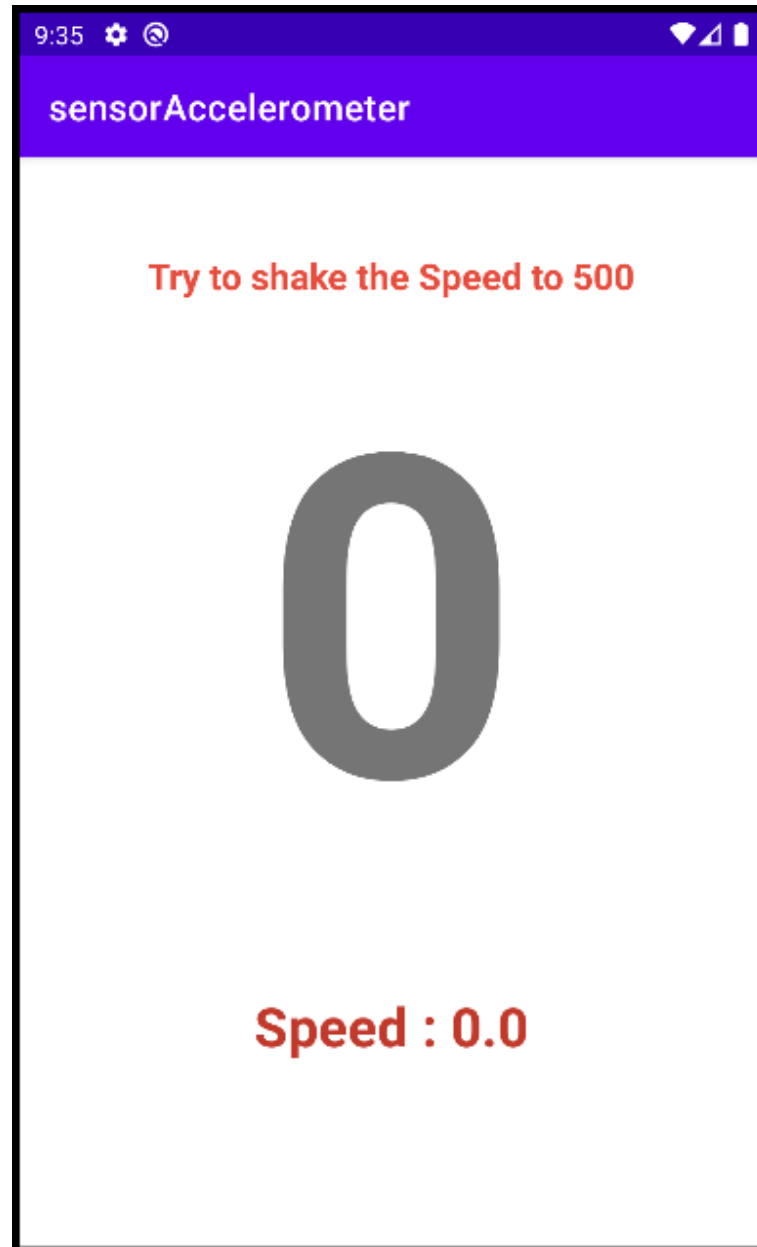
```

<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:duration="800"
        android:fromXScale="15.0"
        android:fromYScale="-20.0"
        android:toXScale="15.0"
        android:toYScale="15.0"
    />
</set>

```

Hình 11 – Code main (6) cho ứng dụng minh hoạ

2.3 Kết quả và đánh giá ứng dụng



Hình 12 – Ảnh kết quả giao diện ban đầu của ứng dụng

Đánh giá ứng dụng :

- + Ứng dụng còn nhiều hạn chế về mặt chính xác của việc tính toán lực tác động vào thiết bị khi sử dụng tính năng cảm biến gia tốc (*sensor accelerometer*) của *Android*.
- + Giao diện còn thô sơ, đơn giản, tốc độ cập nhập thông số khi giá trị được thay đổi còn khá chậm.

TÀI LIỆU THAM KHẢO

- [1] <https://123docz.net/document/2626256-bai-6-sensor.htm>
- [2] <https://viblo.asia/p/tim-hieu-ve-sensor-trong-android-6J3ZgkxgZmB>
- [3] <https://viblo.asia/p/so-luoc-ve-sensors-trong-android-mPjxMe3xv4YL>
- [4] <https://thegioidienco.vn/sensor.html>