

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

รายงานความก้าวหน้าโครงการทางวิศวกรรม

(SENIOR PROJECT PROGRESS REPORT)

ชื่อโครงการ (ภาษาไทย) โปรแกรมค้นหาตำแหน่งเป้าหมายโดยใช้เครือข่ายแอดฮอค
(ภาษาอังกฤษ) Target location searching application using ad hoc network

เสนอโดย

ชื่อ - นามสกุล	เลขประจำตัว	ลายมือชื่อ
1. นางสาวกมลลักษณ์ สุขเสน	5230005721
2. นางสาวกรรณก ขาวอำไพ	5230008621

อาจารย์ที่ปรึกษาโครงการ

ลายมือชื่อ

ผู้ช่วยศาสตราจารย์ ดร. กุลธิดา โรจน์วิบูลย์ชัย

.....

ภาควิชาวิศวกรรมคอมพิวเตอร์

19 พฤศจิกายน 2555

สารบัญ

เรื่อง	หน้า
บทนำ.....	1
รายละเอียดการดำเนินงานที่ผ่านมา.....	4
1. ศึกษาการทำงานของเครือข่ายแอตฮอกและโปรโตคอล DECA.....	4
2. ศึกษาเครือข่ายแลนไร้สายและการเชื่อมต่อกับเครือข่ายแลนไร้สาย	8
3. การใช้งานเครือข่ายแอตฮอกบนระบบปฏิบัติการแอนดรอยด์.....	10
4. การเชื่อมต่อกับแลนไร้สายบนระบบปฏิบัติการแอนดรอยด์ เพื่อใช้ในการระบุตำแหน่ง	13
ความก้าวหน้าเมื่อเทียบกับกำหนดการที่วางไว้.....	18
1. แผนการดำเนินงานเดิม	18
2. การดำเนินงานในปัจจุบัน เมื่อเทียบกับแผนการดำเนินงานเดิม	19
3. การปรับเปลี่ยนกำหนดการ	20
แผนการดำเนินงานขั้นถัดไป	21
อุปสรรคและแนวทางแก้ไข	23
สรุปผลการดำเนินงาน.....	25
รายการอ้างอิง.....	27

บทนำ

โลกเราในปัจจุบันนี้เป็นยุคโลกาภิวัตน์ (Globalization) ซึ่งก็คือสังคมที่เต็มไปด้วยข้อมูลและข่าวสาร และเป็นยุคที่เป็นโลกของการติดต่อสื่อสารที่ไร้พรมแดน ทั้งนี้เพราะเทคโนโลยีการติดต่อสื่อสารมีความทันสมัยก้าวหน้าสามารถเชื่อมต่อโลกทั้งโลกได้ โดยไม่มีอุปสรรคด้านเวลาและระยะทาง ความก้าวหน้าในเทคโนโลยีที่ไม่เคยหยุดนิ่งนี้ส่งผลกระทบต่อการดำเนินชีวิตของมนุษย์ในทุกๆ ด้านไม่ว่าจะเป็นด้านการทำงาน การทำธุรกิจ การศึกษา การวิจัยพัฒนา หรือแม้กระทั่งความบันเทิงต่างๆ และเทคโนโลยีหนึ่งที่มีบทบาทอย่างมากในชีวิตประจำวันของมนุษย์ คือ ระบบเครือข่ายไร้สาย ซึ่งเป็นระบบการสื่อสารข้อมูลที่มีรูปแบบในการสื่อสารแบบไม่ใช้สาย แต่ใช้การส่งคลื่นความถี่วิทยุในย่านวิทยุ RF และคลื่นอินฟราเรดสำหรับการรับและส่งข้อมูลระหว่างอุปกรณ์แต่ละอุปกรณ์ ผ่านอากาศ, ทะลुकำแพง, เพดานหรือสิ่งก่อสร้างอื่นๆ นอกจากนั้นระบบเครือข่ายไร้สายก็ยังมีคุณสมบัติครอบคลุมทุกอย่างเหมือนกับระบบแลนแบบใช้สาย ซึ่งการใช้งานที่น่าสนใจที่สุดของเครือข่ายไร้สายก็คือ ความสะดวกสบายที่ไม่ต้องติดอยู่กับที่ และผู้ใช้สามารถเคลื่อนที่ไปมาได้โดยที่ยังสื่อสารอยู่ในระบบเครือข่ายด้วยเหตุนี้จึงมีการพัฒนาเทคโนโลยีที่สนับสนุนการใช้งานระบบเครือข่ายไร้สายในรูปแบบต่างๆ ที่สามารถใช้ได้ในอุปกรณ์พกพา (Mobile device) เช่น สมาร์ทโฟน และแท็บเล็ต เป็นต้น

เครือข่ายไร้สายแบบแอดฮอค (MANET : Mobile Ad hoc Networks) เป็นรูปแบบเครือข่ายที่ทำให้อุปกรณ์ต่างๆ สามารถติดต่อสื่อสารกันได้โดยไม่ต้องอาศัยโครงสร้างพื้นฐาน (Infrastructure) ทำให้สามารถใช้งานได้ในพื้นที่โดยไม่ต้องติดตั้งโครงสร้างพื้นฐาน นอกจากนี้แอดฮอควงมีความทนทานต่อความบกพร่องได้มากกว่าการสื่อสารที่อาศัยโครงสร้างพื้นฐานเป็นศูนย์กลางการทำงาน เนื่องจากมีรูปแบบการทำงานแบบกระจายและยังเสียค่าใช้จ่ายในการติดตั้งระบบน้อยกว่าระบบที่ต้องติดตั้งโครงสร้างพื้นฐานด้วย

ในการพัฒนาเครือข่ายแอดฮอกบนระบบปฏิบัติการแอนดรอยด์นั้น วิธีที่ใช้ในการจัดการเรื่องการรับและส่งข้อมูลจะอ้างอิงจากโพรโตคอล DECA ซึ่งเป็นโพรโตคอลที่ได้มีการทดลองเพื่อพิสูจน์ประสิทธิภาพการทำงานในเรื่องการกระจายข้อมูลว่ามีความสามารถในการกระจายข้อมูลได้อย่างทั่วถึง ใช้เวลาในการส่งข้อมูลต่ำ และไม่ใช้ทรัพยากรของเครือข่ายมาก จึงทำให้โพรโตคอลนี้สามารถทำงานได้ดีเมื่อเทียบกับโพรโตคอลอื่นๆ ผู้พัฒนาจึงนำโพรโตคอลนี้มาเป็นต้นแบบในการพัฒนาเครือข่ายแอดฮอกบนระบบปฏิบัติการแอนดรอยด์

การจัดทำโครงการวิศวกรรมเรื่องโปรแกรมค้นหาตำแหน่งเป้าหมายโดยใช้เครือข่ายแอดฮอก มีวัตถุประสงค์เพื่อพัฒนาต่อยอดโปรโตคอล Density-Aware Reliable Broadcasting in Vehicular Ad-hoc Network (DECA) ที่พัฒนาขึ้นมาสำหรับระบบปฏิบัติการแอนดรอยด์ โดยการนำโปรโตคอลดังกล่าวไปพัฒนาเป็น แอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ ทำให้อุปกรณ์พกพาที่ใช้ระบบปฏิบัติการแอนดรอยด์สามารถ เชื่อมต่อและรับส่งข้อมูลแบบแพร่กระจายที่มีความน่าเชื่อถือ โดยการพัฒนาแอปพลิเคชันสำหรับการค้นหา ตำแหน่งของบุคคลภายในอาคารนั้นจะเรียกใช้งานโปรโตคอล DECA ที่ได้รับการพัฒนาเป็นไลบรารีให้สามารถ เรียกใช้งานโปรโตคอลดังกล่าว ในการสร้างเครือข่ายแอดฮอกสำหรับติดต่อรับส่งข้อมูลระหว่างอุปกรณ์พกพาที่ใช้ ระบบปฏิบัติการแอนดรอยด์

เนื่องจากในปัจจุบันอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์นั้นมีหลากหลายรูปแบบ และแต่ละรูปแบบมี ความแตกต่างกันในด้านฮาร์ดแวร์ ดังนั้นการพัฒนาแอปพลิเคชันให้สามารถใช้งานได้กับทุกอุปกรณ์จึงเป็นไปได้ ยาก การวิจัยนี้ผู้พัฒนาจึงเลือกใช้โทรศัพท์มือถือ Google Nexus One ซึ่งถูกออกแบบมาสำหรับนักพัฒนา โดยเฉพาะและเป็นที่ยอมรับในท้องตลาด มาเป็นต้นแบบในการพัฒนาและใช้ในการทดสอบ โดยเป้าหมายของ โครงการนี้ คือ การพัฒนาแอปพลิเคชันสำหรับการค้นหาตำแหน่งของบุคคลภายในอาคารที่ทำงานได้อย่างถูกต้อง บนโทรศัพท์มือถือ Google Nexus One

Software

- JAVA Runtime Environment (JRE)
- Eclipse IDE for JAVA Developer
- Android SDK (Software Development Kit)
- Android Development Tool (ADT)



Hardware

- Google Nexus One 2 เครื่อง ใช้ ROM CyanogenMod 7



รายละเอียดการดำเนินงานที่ผ่านมา

1. ศึกษาการทำงานของเครือข่ายแอตฮอกและโปรโตคอล DECA

จากการศึกษาหนังสือ AD HOC NETWORKS Technologies and Protocols [4] เครือข่ายไร้สายแอตฮอก คือ เครือข่ายไร้สายที่ไม่อาศัยโครงสร้างพื้นฐานในการติดต่อสื่อสารแต่จะใช้สัญญาณวิทยุ ทำให้ผู้ใช้งานแอตฮอกสามารถใช้งานขณะเคลื่อนที่ได้ แต่เนื่องจากขอบเขตการกระจายสัญญาณของเครือข่ายไร้สายมีข้อจำกัด กล่าวคือ ระยะทางระหว่างโหนดเชื่อมต่อที่ใกล้กันมาก หรือสิ่งแวดล้อมที่มีสิ่งกีดขวางมาก อาจส่งผลกระทบทำให้คุณภาพการส่งข้อมูลลดลงหรือขาดการเชื่อมต่อได้

ลักษณะการสื่อสารแบบแอตฮอก

ระบบเครือข่ายแอตฮอกมีลักษณะการติดต่อสื่อสารแบบเพียร์ทูเพียร์(Peer to Peer) หมายความว่า การติดต่อสัญญาณข้อมูลจากโหนดต้นทางไปยังโหนดปลายทาง มีการติดต่อกันระหว่างชั้นโปรโตคอลที่อยู่ในระดับเดียวกัน ลักษณะที่สำคัญอีกอย่างหนึ่งของเครือข่ายแอตฮอก คือ โหนดที่อยู่ภายในเครือข่ายมักมีการเคลื่อนที่ตลอดเวลา ทำให้รูปแบบของเครือข่าย (Topology) เปลี่ยนแปลงตลอดเวลา จึงเกิดปัญหาการเชื่อมต่อหลุดไปเมื่อโหนดมีการเคลื่อนที่ ซึ่งถือว่าเป็นข้อจำกัดของเครือข่ายไร้สายแอตฮอก นอกจากนี้ยังมีข้อจำกัดทางด้านแบนด์วิดท์ การเชื่อมโยงไร้สายอาจเจอปัญหาเรื่องสัญญาณรบกวน ความจุในการส่งสัญญาณแบบไร้สายที่มีน้อยกว่าแบบใช้สายมาก และข้อจำกัดทางด้านความปลอดภัยของข้อมูล

การค้นทางเส้นทาง(Routing)ในเครือข่ายไร้สายแอตฮอก

การค้นหาเส้นทางจากโหนดหนึ่งไปยังอีกโหนดหนึ่ง จะทำงานในระดับเน็ตเวิร์กเลเยอร์ (Network Layer) ประกอบไปด้วย

- อัลกอริธึมในการค้นหาเส้นทาง (Routing Algorithm)
- ฐานข้อมูล (Database) ที่ใช้เก็บเส้นทางในการส่งข้อมูลไปยังโหนดอื่นๆ
- โปรโตคอลในการค้นหาเส้นทาง (Routing Protocol) โดยสามารถแบ่งได้ 2 ประเภท ประเภทแรกคือ โปรโตคอลค้นหาเส้นทางแบบอาศัยตารางเส้นทางข้อมูล (Table-driven) และอีกประเภทคือโปรโตคอลค้นหาเส้นทางแบบไม่ใช้ตารางเส้นทาง (On-demand)

ซึ่งในโครงการนี้ผู้พัฒนาจะนำโปรโตคอล DECA ซึ่งเป็นโปรโตคอลประเภทค้นหาเส้นทางแบบไม่ใช้ตารางเส้นทาง (On-demand) มาพัฒนาเป็นแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์

โปรโตคอล DECA มีหลักการและรายละเอียดการทำงานดังนี้

- Store and forward: เมื่อมีการส่งข้อความผ่านไปตามเส้นทาง โหนดที่ได้รับข้อความจะทำการเก็บ (store) ข้อความนั้นไว้ก่อน จากนั้นจึงตรวจสอบว่ามีโหนดเพื่อนบ้านใดที่ยังไม่ได้รับข้อความ จึงค่อยส่งข้อความนั้นออกไป(forward)ยังโหนดเพื่อนบ้าน การทำงานในลักษณะนี้ทำให้โปรโตคอลสามารถรองรับการทำงานเชื่อมต่อเป็นช่วงๆได้ด้วย
- Preferred Node Selection Algorithm: เมื่อโหนดต้นทางต้องการจะกระจายข้อมูลจะพิจารณาเลือกโหนดที่มีความหนาแน่นของโหนดเพื่อนบ้านมากที่สุด จากนั้นจึงส่งข้อความให้โหนดที่ถูกเลือก เมื่อโหนดที่ถูกเลือกได้รับข้อความแล้วจะกระจายข้อความที่ได้รับซ้ำทันที เพื่อลดจำนวนการกระจายข้อความซ้ำในบริเวณเดิมและลดเวลาในการรอ
- Waiting Timeout Calculation: โหนดมีการตั้งเวลารอแบบสุ่ม เพื่อลดความเสี่ยงจากกรณีโหนดที่ถูกเลือกไม่ทำงาน ทำให้โหนดอื่นที่ไม่ถูกเลือกสามารถทำการกระจายข้อความซ้ำได้ และโหนดอื่นๆก็ยังสามารถส่งต่อข้อความได้

การออกแบบและพัฒนาโปรโตคอล DECA บนระบบปฏิบัติการแอนดรอยด์

การพัฒนาโปรโตคอล DECA สามารถแบ่งออกเป็น 5 Packages ซึ่งในแต่ละ Packages มีรายละเอียดการพัฒนาดังต่อไปนี้

1. Package deca_agent ประกอบด้วยโครงสร้างคลาสดังต่อไปนี้

- Deca_agent

เป็นคลาสสำหรับการจัดการการทำงานของโปรโตคอล DECA ซึ่งทำหน้าที่ต่างๆ ดังนี้

- แพร่กระจายข้อความจากโหนดตนเองไปยังโหนดผู้อื่น
- รับข้อความที่กระจายโดยโหนดผู้อื่น
- ตรวจสอบข้อความที่ตนเองได้รับกับข้อความที่เพื่อนบ้านได้รับ โดยจะส่งข้อความให้เพื่อนบ้านที่ได้รับข้อความไม่ครบ หรือส่ง beacon ให้เพื่อนบ้านเพื่อขอข้อความที่ตนเองไม่ได้รับ

- แพร์กระจายข้อความซ้ำทันทีเมื่อถูกเลือกให้เป็น prefer node
- แพร์กระจายข้อความซ้ำหากโหนดที่ถูกเลือกให้เป็น prefer node ไม่ทำหน้าที่
แพร์กระจายข้อความซ้ำ
- Deca_agent_thread
เป็น thread แยกออกมาเพื่อทำหน้าที่รับข้อความจากเพื่อนบ้าน
- PacketTimer
เป็นคลาสสำหรับจับเวลาว่าข้อความที่ได้รับมาจะหมดอายุเมื่อใด เมื่อข้อความหมดอายุ
ให้ลบข้อความนั้นออกจาก rtable
- NeighborTimer
เป็นคลาสสำหรับจับเวลาว่าเพื่อนบ้านหมดอายุเมื่อใด เมื่อเพื่อนบ้านหมดอายุให้ลบ
เพื่อนบ้านนั้นออกจาก ntable
- BeaconTimer
เป็นคลาสสำหรับจับเวลาในการส่ง beacon ครั้งต่อไป
- RebroadcastTimer
เป็นคลาสสำหรับจับเวลา back off time ของข้อความที่อยู่ใน ttable ว่าหมดอายุเมื่อใด
ถ้าหมดอายุก็ให้ rebroadcast ข้อความนั้น

2. Package table ประกอบด้วยโครงสร้างคลาสดังต่อไปนี้

- Deca_rp_entry
ใช้สำหรับเก็บข้อมูลของข้อความที่ได้รับจากเพื่อนบ้านและข้อความที่ตนเองส่งออก
- Deca_rtable
ตารางสำหรับเก็บข้อมูล Deca_rp_entry
- Deca_nb_entry
ใช้สำหรับเก็บข้อมูลของเพื่อนบ้าน ซึ่งได้แก่ Address และจำนวนเพื่อนบ้าน
- Deca_ntable
ตารางสำหรับเก็บข้อมูล Deca_nb_entry
- Deca_temp_entry
ใช้สำหรับเก็บข้อมูลของข้อความที่มีการนับเวลาถอยหลังเพื่อแพร่กระจายซ้ำถ้าหากได้
ยินเพื่อนบ้านแพร่กระจายข้อมูลนั้นซ้ำแล้ว ข้อมูลนี้จะถูกลบออก

- Deca_ttable

ตารางสำหรับเก็บข้อมูล Deca_temp_entry

3. Package timer ประกอบด้วยโครงสร้างคลาสดังต่อไปนี้

- Deca_timer

ภายในมี Inner class ใช้สำหรับนับเวลาและสามารถตั้งเวลาที่ต้องการให้ expire ได้

- TimerThread

เป็น Thread สำหรับนับเวลาตามค่าที่ตั้ง เมื่อหมดเวลาแล้วจะทำงานในคำสั่ง expire()

4. Package packet ประกอบด้วยโครงสร้างคลาสดังต่อไปนี้

- Deca_packet

สำหรับสร้าง packet แล้วแปลงให้อยู่ในรูปแบบที่สามารถแพร่กระจายให้โหนดอื่นได้

- Deca_beacon_packet

สำหรับสร้าง Beacon packet ที่ใช้แลกเปลี่ยนข้อมูลระหว่างโหนดที่อยู่ติดกัน

- Deca_broadcast_packet

สำหรับสร้าง Broadcast packet ซึ่งเป็นข้อความที่ใช้โปรโตคอล DECA ในการแพร่กระจายข้อความ

5. Package util ประกอบด้วยโครงสร้างคลาสดังต่อไปนี้

- Deca_util

สำหรับอำนวยความสะดวกให้กับการทำงานของโปรโตคอล DECA มีฟังก์ชันสำหรับเปลี่ยนข้อมูล address จาก Integer เป็น String หรือจาก String เป็น Integer และฟังก์ชันคำนวณหา broadcast address

2. ศึกษาเครือข่ายแลนไร้สายและการเชื่อมต่อกับเครือข่ายแลนไร้สาย

เป็นเทคโนโลยีที่ใช้เชื่อมต่ออุปกรณ์สื่อสารเข้าด้วยกันโดยปราศจากการใช้สายสัญญาณในการเชื่อมต่อแต่จะใช้คลื่นวิทยุหรือคลื่นอินฟราเรดเป็นตัวกลางในการสื่อสารแทน โดยขอบเขตของสัญญาณครอบคลุมพื้นที่ประมาณ 100 เมตรในที่โล่งและประมาณ 30 เมตรในอาคาร ซึ่งระยะของสัญญาณจะได้รับผลกระทบจากสิ่งแวดล้อมรอบข้าง

หลักการทำงานของระบบเครือข่ายแลนไร้สาย (Wireless LAN)

การทำงานจะมีอุปกรณ์ในการส่งสัญญาณและกระจายสัญญาณ หรือที่เราเรียกว่า Access Point และมี PC Card ที่เป็น LAN Card สำหรับในการเชื่อมกับ Access Point โดยเฉพาะ การทำงานจะใช้คลื่นวิทยุเป็นการรับส่งสัญญาณ โดยมีให้เลือกใช้ตั้งแต่ 2.4 to 2.4897 Ghz

การเชื่อมต่อของระบบเครือข่าย Wireless LAN

Wi-Fi (Wireless Fidelity) เป็นองค์กรที่ทำหน้าที่ทดสอบผลิตภัณฑ์ Wireless LAN ภายใต้มาตรฐาน IEEE 802.11Wi-Fi ได้กำหนดการเชื่อมต่ออุปกรณ์ภายใน Wireless LAN ออกเป็น 2 ลักษณะ ได้แก่

- Infrastructure mode เป็นโหมดที่อนุญาตให้อุปกรณ์ Wireless LAN สามารถเชื่อมต่อกับเครือข่ายอื่นได้ ประกอบไปด้วยอุปกรณ์ 2 ประเภท ได้แก่

1. สถานีผู้ใช้ (Client Station) คือ อุปกรณ์คอมพิวเตอร์ที่สามารถเชื่อมต่อกับระบบ Wireless LAN ได้
2. สถานีแม่ข่าย (Access Point) คือ อุปกรณ์ซึ่งทำหน้าที่เชื่อมต่อสถานีผู้ใช้เข้ากับเครือข่ายอื่น โดยสามารถเชื่อมต่อได้ทั้งที่เป็นเครือข่ายแบบใช้สายและเครือข่ายแบบไร้สาย

- Ad hoc mode เป็นเครือข่ายแบบปิด คือ ไม่มีการเชื่อมต่อกับเครือข่ายอื่น เนื่องจากไม่มีสถานีแม่ข่าย อุปกรณ์ในเครือข่ายจะสื่อสารกันเองโดยตรงแบบ Peer-to-Peer

Received Signal Strength Indication (RSSI)

RSSI เป็นค่าที่ใช้บอกความแรงของสัญญาณวิทยุที่ได้รับในเทอมของพลังงานมีหน่วยเป็น dBm (decibels milliwatt) โดยค่า RSSI จะแปรผันตรงกับความแรงของสัญญาณ กล่าวคือ ถ้าค่า RSSI มีค่ามาก แสดงว่าสัญญาณที่ได้รับมีความแรงสูง นั่นคือตัวส่งและตัวรับอยู่ใกล้กัน ในทางกลับกันหากค่า RSSI มีค่าน้อยแสดงว่าสัญญาณที่ได้รับมีความแรงต่ำ นั่นคือตัวส่งและตัวรับอยู่ไกลกัน

ระยะทางการเชื่อมต่อของระบบ Wireless LAN

ภายในอาคาร

1. ระยะ 50 เมตร ได้ความเร็วประมาณ 11 Mbps
2. ระยะ 80 เมตร ได้ความเร็วประมาณ 5.5 Mbps
3. ระยะ 120 เมตร ได้ความเร็วประมาณ 2 Mbps
4. ระยะ 150 เมตร ได้ความเร็วประมาณ 1 Mbps

ภายนอกอาคาร

1. ระยะ 250 เมตร ได้ความเร็วประมาณ 11 Mbps
2. ระยะ 350 เมตร ได้ความเร็วประมาณ 5.5 Mbps
3. ระยะ 400 เมตร ได้ความเร็วประมาณ 2 Mbps
4. ระยะ 500 เมตร ได้ความเร็วประมาณ 1 Mbps

3. การใช้งานเครือข่ายแอตฮอกบนระบบปฏิบัติการแอนดรอยด์

การใช้งานเครือข่ายแอตฮอกบนระบบปฏิบัติการแอนดรอยด์ ไม่สามารถทำผ่าน Graphic User Interface ของระบบปฏิบัติการได้โดยตรง แต่สามารถทำได้โดยการพิมพ์คำสั่งผ่านทางเทอร์มินัล ซึ่งจะใช้คำสั่งตามลำดับที่แสดงดังต่อไปนี้

เริ่มจากเปลี่ยนสิทธิ์ของเราเป็น Superuser ก่อนโดยใช้คำสั่ง `su`

ต่อมาติดตั้งโมดูลชื่อว่า `bcm4329.ko` เพื่อให้เครื่องสามารถมองเห็นอินเทอร์เฟซ `eth0` ซึ่งเป็นอินเทอร์เฟซของ Wireless card ซึ่งการใช้งานเครือข่ายแลนไร้สายหรือ Wireless LAN จะใช้อินเทอร์เฟซนี้

`insmod /system/lib/modules/bcm4329.ko dhd_pkt_filter_enable=0`

หากต้องการตรวจสอบว่าโมดูลดังกล่าวถูกติดตั้งแล้วหรือไม่ให้ใช้คำสั่ง `lsmod` ซึ่งจะแสดงรายการโมดูลที่ติดตั้งบนเครื่องเรียบร้อยแล้ว

```
export PATH=/data/local/bin:$PATH
$ export PATH=/data/local/bin:$PATH
$ su
# lsmod
bcm4329 185125 0 - Live 0xbf038000
#
```

และใช้คำสั่ง `netcfg` ในการตรวจสอบว่าเครื่องสามารถมองเห็นอินเทอร์เฟซ `eth0` แล้วหรือไม่

```
export PATH=/data/local/bin:$PATH
$ export PATH=/data/local/bin:$PATH
$ su
# netcfg
lo      UP      127.0.0.1      255.0.0.0      0x0000
0049
gannet0 DOWN    0.0.0.0        0.0.0.0        0x0000
1082
dummy0  DOWN    0.0.0.0        0.0.0.0        0x0000
0082
ifb0    DOWN    0.0.0.0        0.0.0.0        0x0000
0082
ifb1    DOWN    0.0.0.0        0.0.0.0        0x0000
0082
rmnet0  DOWN    0.0.0.0        0.0.0.0        0x0000
1002
rmnet1  DOWN    0.0.0.0        0.0.0.0        0x0000
1002
rmnet2  DOWN    0.0.0.0        0.0.0.0        0x0000
1002
usb0    DOWN    0.0.0.0        0.0.0.0        0x0000
1002
sit0    DOWN    0.0.0.0        0.0.0.0        0x0000
0080
ip6tnl0 DOWN    0.0.0.0        0.0.0.0        0x0000
0080
eth0    UP      192.168.1.5    255.255.255.0  0x0000
1043
#
```

หลังจากติดตั้งโมดูลเสร็จเรียบร้อยแล้ว กำหนดค่าของอินเทอร์เฟซโดยใช้คำสั่ง

```
ifconfig eth0 up [IP Address]
```

เปลี่ยนมาใช้งานในโหมดแอดฮอกโดยใช้คำสั่ง

```
iwconfig eth0 mode ad-hoc essid DecaFriend
```

เมื่อพิมพ์คำสั่ง `iwconfig` จะเห็นว่าอินเทอร์เฟซ `eth0` กำลังอยู่ในโหมดแอดฮอก

```
export PATH=/data/local/bin:$PATH
$ export PATH=/data/local/bin:$PATH
$ su
# iwconfig
lo        no wireless extensions.

gannet0   no wireless extensions.

dummy0    no wireless extensions.

ifb0      no wireless extensions.

ifb1      no wireless extensions.

rmnet0    no wireless extensions.

rmnet1    no wireless extensions.

rmnet2    no wireless extensions.

usb0      no wireless extensions.

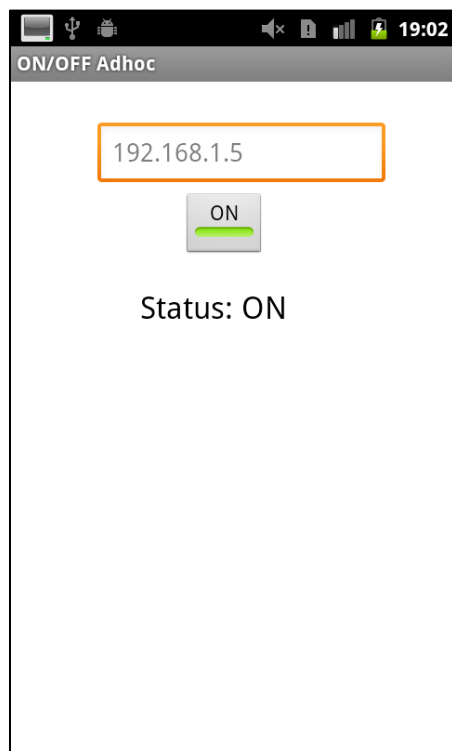
sit0      no wireless extensions.

ip6tn10   no wireless extensions.

eth0      IEEE 802.11-DS  ESSID:"DecaFriend"  Nicknam
e:""
          Mode:Ad-Hoc  Frequency:2.412 GHz  Cell: 56:
7C:44:1B:20:DC
          Bit Rate=72 Mb/s   Tx-Power:32 dBm
          Retry min limit:7   RTS thr:off   Fragment
thr:off
          Encryption key:off
          Power Managementmode:All packets received
          Link Quality=5/5   Signal level=-57 dBm   Noi
se level=-87 dBm
          Rx invalid nwid:0   Rx invalid crypt:0   Rx i
nvalid frag:0
          Tx excessive retries:0   Invalid misc:0   Mi
ssed beacon:0

#
```

การใช้งานเครือข่ายแอดฮอกเพื่อใช้ในการแพร่กระจายข้อมูลตำแหน่งที่อยู่ไปยังโหนดอื่นๆในเครือข่าย หากใช้การพิมพ์คำสั่งผ่านเทอร์มินัลเพื่อเปิดแอดฮอกจะเป็นการยุ่งยากและไม่สะดวกต่อการใช้งาน ผู้พัฒนาจึงสร้างแอปพลิเคชันที่มี Graphic User Interface ขึ้นมาใช้ในการทดสอบการสร้างเครือข่ายแอดฮอกบนอุปกรณ์ แสดงได้ดังภาพ



เมื่อคลิกที่ปุ่มสำหรับเปิด/ปิดแอตสอชเพื่อเปิดการใช้งานในเครือข่ายในโหมดแอตสอช จะได้เครือข่ายแอตสอช ชื่อว่า “DecaFriend” ดังภาพ



4. การเชื่อมต่อกับแลนไร้สายบนระบบปฏิบัติการแอนดรอยด์ เพื่อใช้ในการระบุตำแหน่ง

การพัฒนาแอปพลิเคชันเพื่อเชื่อมต่อกับเครือข่ายแลนไร้สาย

สำหรับการควบคุมการทำงานของอุปกรณ์สำหรับการเชื่อมต่อแลนไร้สาย (Wi-Fi) บนระบบปฏิบัติการแอนดรอยด์สามารถทำได้โดยเรียกใช้ผ่าน Object ที่สร้างจากคลาส WifiManager ซึ่งเป็นคลาสที่มีการพัฒนาไว้แล้ว เมื่ออิมพอร์ตคลาสนี้เข้ามาก็สามารถเรียกใช้งานได้เลยโดยไม่ต้องพัฒนาขึ้นมาใหม่ เมธอดสำคัญที่ผู้พัฒนาเรียกใช้ในการพัฒนาแอปพลิเคชันมีดังต่อไปนี้

- turnOnWiFi ()

ทำหน้าที่เปิดฟังก์ชันการทำงานของเครือข่าย Wi-Fi บนระบบปฏิบัติการแอนดรอยด์

```
Public void turnOnWiFi(){
    if(wifiManager.isWifiEnabled()==false){
        wifiManager.setWifiEnabled(true);
    }
}
```

- turnOffWiFi ()

ทำหน้าที่ปิดฟังก์ชันการทำงานของเครือข่าย Wi-Fi บนระบบปฏิบัติการแอนดรอยด์

```
Public void turnOffWiFi(){
    if(wifiManager.isWifiEnabled()==true){
        wifiManager.setWifiEnabled(false);
    }
}
```

- scan ()

ทำหน้าที่สแกนหาสัญญาณจากเครือข่ายแลนไร้สายที่ได้รับ เพื่อนำ MAC Address ที่ได้มาช่วยในการระบุตำแหน่ง

```
Public void scan(){
    filter = new IntentFilter();
    filter.addAction(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION);
    registerReceiver(wifiEventReceiver, filter);
}
```

โค้ดสำหรับ wifiEventReceiver ช่วยในการสแกนหา MAC Address ,ESSID และRSSI ซึ่งจะได้ผลการสแกนอยู่ในรูปแบบของโครงสร้างข้อมูลแบบลิสต์ และมีการเรียงลำดับตามความแรงของสัญญาณที่ได้รับจากเครือข่ายแลนไร้สาย โดยสัญญาณจากเครือข่ายที่แรงที่สุดจะอยู่ลิสต์ช่องที่ 0

```
Private BroadcastReceiver wifiEventReceiver = new BroadcastReceiver() {
    @Override
    Public void onReceive(Context arg0, Intent intent){
        if(intent.getAction().equals(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION)){
            List<ScanResult> list = wifiManager.getScanResults();
        }
        wifiManager.startScan();
    }
};
```

การพัฒนาแอปพลิเคชันเพื่อระบุตำแหน่ง

แอปพลิเคชันที่พัฒนาขึ้นมานั้นจะมีคลาส Localizer เป็นคลาสที่ทำหน้าที่เกี่ยวกับการบอกตำแหน่ง คลาสนี้จะมีเมธอดชื่อ findLocation ทำหน้าที่คำนวณตำแหน่งจากค่า MAC Address ที่ได้รับเข้ามา หลังจากนั้นจะคืนค่าเป็นข้อมูลประเภท String แสดงตำแหน่งที่อยู่จาก MAC Address ดังกล่าว

การพัฒนาแอปพลิเคชันในส่วนของการระบุตำแหน่งนั้น ผู้พัฒนาได้ทดลองใช้ Google Nexus One รับสัญญาณจากเครือข่ายแลนไร้สาย ณ ตำแหน่งต่างๆของหอพักพวงชมพูหญิง 1 และเก็บข้อมูลการรับสัญญาณ ดังกล่าว ได้ผลการเก็บข้อมูลดังนี้

บริเวณชั้น 2

ZONE D

58:bc:27:13:39:a8
58:bc:27:13:39:a5

58:bc:27:12:1f:d5
58:bc:27:12:1f:d8

ZONE C

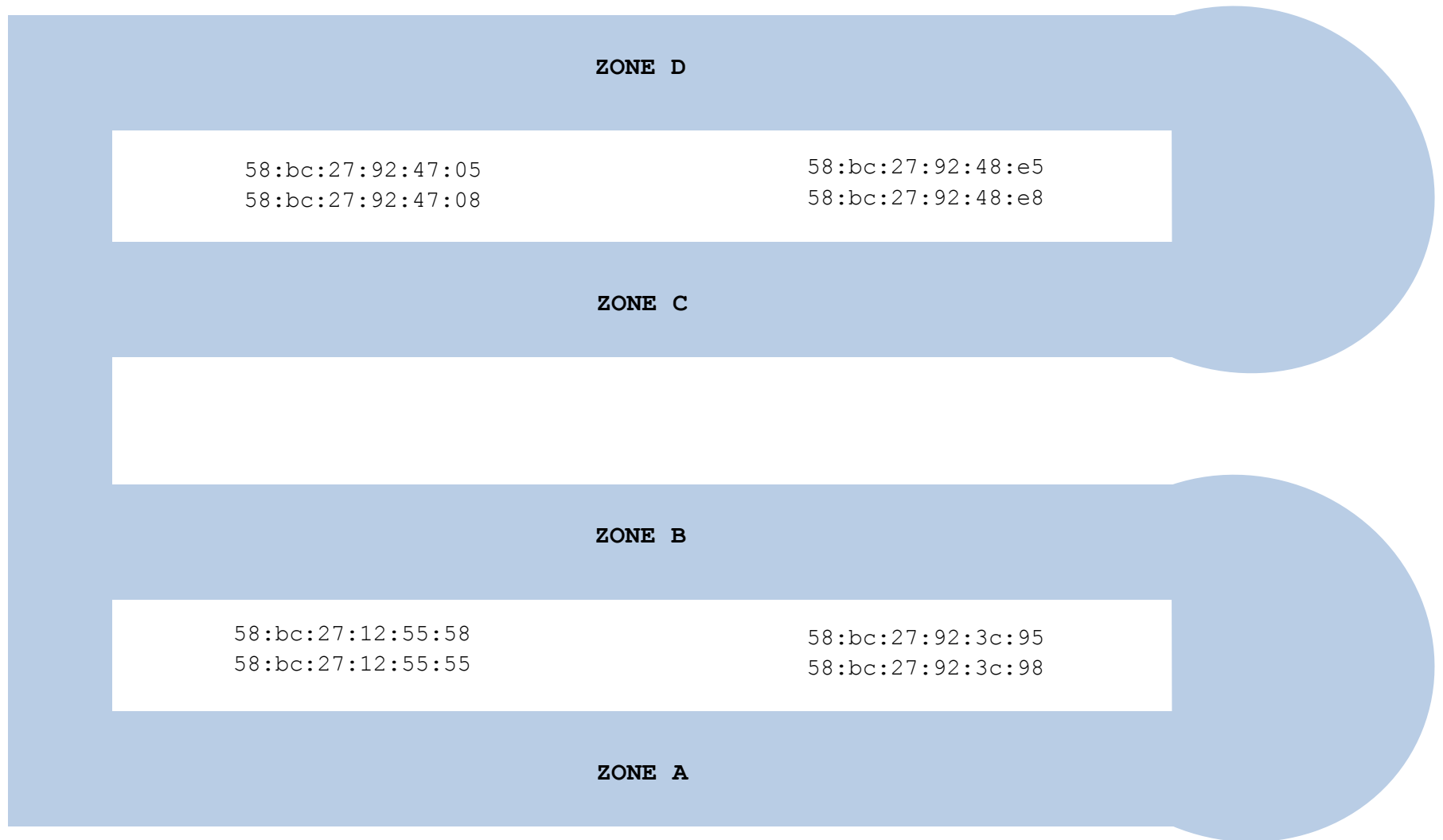
ZONE B

58:bc:27:12:1f:b5
58:bc:27:12:1f:b8

58:bc:27:12:44:45
58:bc:27:12:44:48

ZONE A

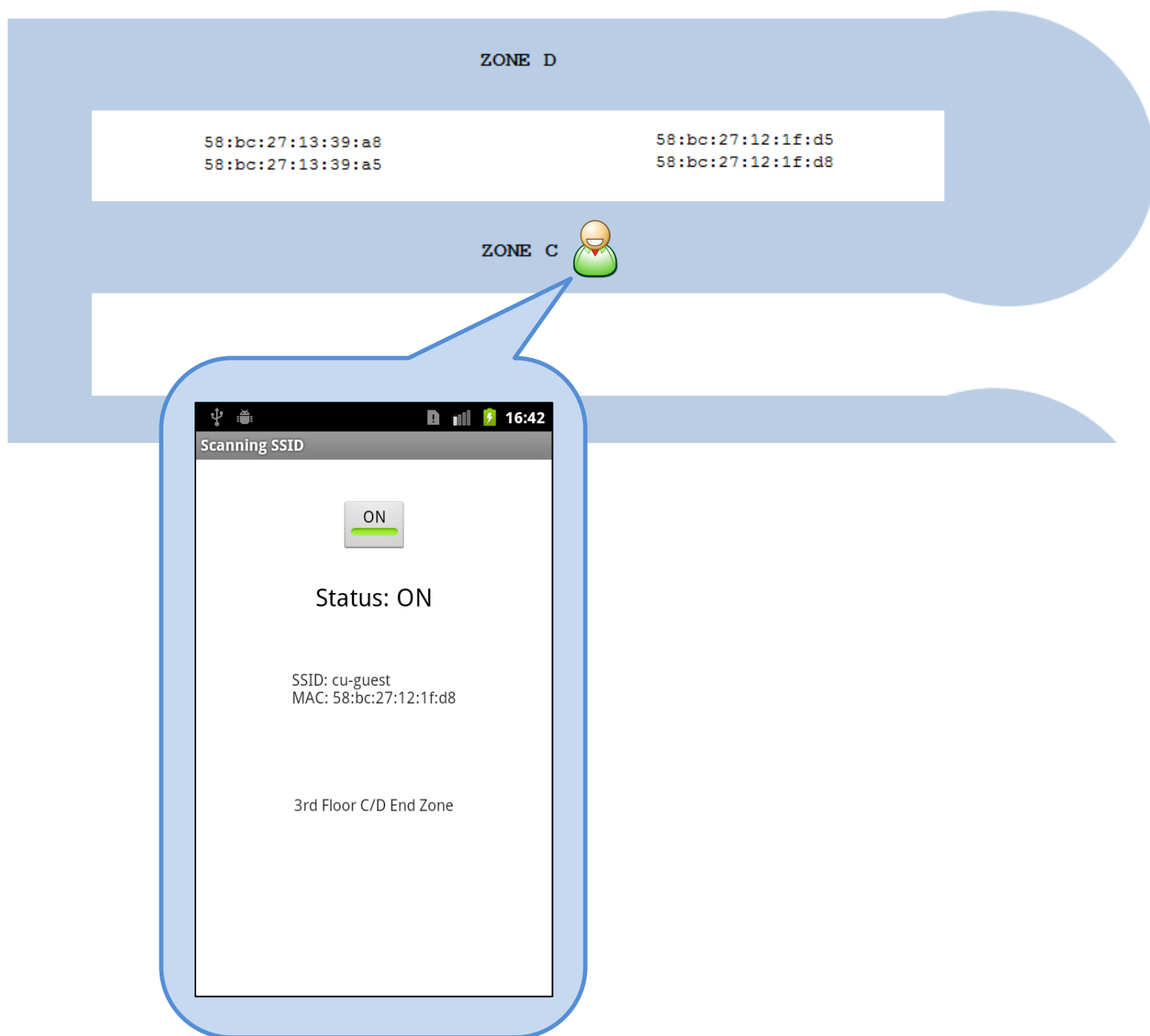
บริเวณชั้น 3



หมายเหตุ MAC Address ที่แสดงในภาพนั้นเป็น MAC Address ของ Access Point ที่ Google Nexus one จับสัญญาณได้แรงที่สุด

การทดสอบเบื้องต้นพบว่าการใช้วิธีการนี้ในการระบุตำแหน่ง สามารถระบุตำแหน่งได้ค่อนข้างละเอียด และมีความแม่นยำในระดับหนึ่ง และผู้พัฒนาจะทำการทดสอบประสิทธิภาพในการระบุตำแหน่งโดยใช้แอปพลิเคชันอย่างละเอียดอีกครั้งในการดำเนินงานครั้งถัดไป

ตัวอย่างการทำงานของแอปพลิเคชันในการระบุตำแหน่ง สามารถแสดงได้ดังภาพ



ความก้าวหน้าเมื่อเทียบกับกำหนดการที่วางไว้

1. แผนการดำเนินงานเดิม

วัน/เดือน/ปี	การดำเนินงาน
11 กรกฎาคม 2555	ส่งหัวข้อโครงการ
12 กรกฎาคม – 19 กรกฎาคม 2555	กำหนดขอบเขตของการวิจัย
20 กรกฎาคม – 3 สิงหาคม 2555	ศึกษาเครือข่ายแอตฮอกและโปรโตคอล DECA
6 สิงหาคม – 10 สิงหาคม 2555	ศึกษาเครือข่ายแลนไร้สายและการเชื่อมต่อของเครือข่าย
13 สิงหาคม – 17 สิงหาคม 2555	ศึกษาการพัฒนาแอปพลิเคชันบนแอนดรอยด์
20 สิงหาคม – 27 สิงหาคม 2555	ศึกษางานวิจัยที่เกี่ยวข้องและเครื่องมือที่จำเป็นในการวิจัย
28 สิงหาคม – 7 กันยายน 2555	ทำรายงาน Project Proposal
10 กันยายน 2555	ส่งรายงาน Project proposal ต่อคณะกรรมการ
11 กันยายน – 14 กันยายน 2555	เตรียม Slide สำหรับนำเสนอ Project proposal
17 กันยายน 2555	สอบ Project proposal
18 กันยายน – 23 ตุลาคม 2555	พัฒนาการส่งข้อมูลบนเครือข่ายแอตฮอกและการระบุตำแหน่ง
24 ตุลาคม – 7 พฤศจิกายน 2555	พัฒนาแอปพลิเคชันบนแอนดรอยด์
8 พฤศจิกายน – 9 พฤศจิกายน 2555	ทำรายงานความก้าวหน้าของโครงการ
14 พฤศจิกายน 2555	ส่งรายงานความก้าวหน้าของโครงการ
13 พฤศจิกายน – 15 พฤศจิกายน 2555	เตรียม Slide สำหรับนำเสนอความก้าวหน้าของโครงการ
16 พฤศจิกายน 2555	นำเสนอความก้าวหน้าของโครงการต่อคณะกรรมการ

2. การดำเนินงานในปัจจุบันเมื่อเทียบกับแผนการดำเนินงานเดิม

วัน/เดือน/ปี	การดำเนินงาน
11 กรกฎาคม 2555	ส่งหัวข้อโครงการ
12 กรกฎาคม – 19 กรกฎาคม 2555	กำหนดขอบเขตของการวิจัย
20 กรกฎาคม – 3 สิงหาคม 2555	ศึกษาเครือข่ายแอตสอกและโปรโตคอล DECA
6 สิงหาคม – 10 สิงหาคม 2555	ศึกษาเครือข่ายแลนไร้สายและการเชื่อมต่อของเครือข่าย
13 สิงหาคม – 17 สิงหาคม 2555	ศึกษาการพัฒนาแอปพลิเคชันบนแอนดรอยด์
20 สิงหาคม – 27 สิงหาคม 2555	ศึกษางานวิจัยที่เกี่ยวข้องและเครื่องมือที่จำเป็นในการวิจัย
28 สิงหาคม – 7 กันยายน 2555	ทำรายงาน Project Proposal
10 กันยายน 2555	ส่งรายงาน Project proposal ต่อคณะกรรมการ
11 กันยายน – 14 กันยายน 2555	เตรียม Slide สำหรับนำเสนอ Project proposal
17 กันยายน 2555	สอบ Project proposal
18 กันยายน – 27 ตุลาคม 2555	พัฒนาการส่งข้อมูลบนเครือข่ายแอตสอกและการระบุตำแหน่ง
27 ตุลาคม – 5 พฤศจิกายน 2555	พัฒนาแอปพลิเคชันบนแอนดรอยด์
6 พฤศจิกายน – 7 พฤศจิกายน 2555	ทำรายงานความก้าวหน้าของโครงการ
8 พฤศจิกายน 2555	ส่งรายงานความก้าวหน้าของโครงการให้อาจารย์ที่ปรึกษาตรวจแก้
19 พฤศจิกายน 2555	ส่งรายงานความก้าวหน้าของโครงการ
19 พฤศจิกายน – 22 พฤศจิกายน 2555	เตรียม Slide สำหรับนำเสนอความก้าวหน้าของโครงการ
23 พฤศจิกายน 2555	นำเสนอความก้าวหน้าของโครงการต่อคณะกรรมการ

การดำเนินงานที่ผ่านมา ผู้พัฒนาได้นำโปรโตคอล DECA ที่ได้รับการพัฒนาเป็นไลบรารี ให้สามารถเรียกใช้งานโปรโตคอลดังกล่าวในการสร้างเครือข่ายแอตสอกเพื่อติดต่อรับส่งข้อมูลระหว่างอุปกรณ์พกพาที่ใช้ระบบปฏิบัติการแอนดรอยด์มาอิมพลีเมนต์ได้สำเร็จ แต่เนื่องจากพบปัญหาเกี่ยวกับอุปกรณ์ wireless card บนโทรศัพท์มือถือ Google Nexus One ที่ไม่สนับสนุนการใช้งานเครือข่ายแลนไร้สายในโหมด adhoc พร้อมกับโหมด managed จึงทำให้การดำเนินงานในขั้นตอนการพัฒนาการส่งข้อมูลบนเครือข่ายแอตสอกและการระบุตำแหน่งล่าช้ากว่ากำหนดการที่วางแผนไว้ประมาณ 4 วัน และเริ่มทำรายงานความก้าวหน้าของโครงการเร็วขึ้นเพื่อให้อาจารย์ที่ปรึกษาโครงการทำการตรวจแก้ไข นอกจากนี้กิจกรรมจุฬาวិชาการที่จัดขึ้นในสัปดาห์ที่

ต้องส่งรายงานความก้าวหน้าของโครงการตามกำหนดการเดิม จึงทำให้มีการเลื่อนกำหนดการส่งรายงานความก้าวหน้าของโครงการออกไปอีก 1 อาทิตย์

3. การปรับเปลี่ยนกำหนดการ

เนื่องจากการพัฒนาการส่งข้อมูลบนเครือข่ายแอตฮอกและการระบุตำแหน่ง เป็นขั้นตอนสำคัญในโครงการนี้ จึงต้องมีการทดสอบการส่งข้อมูลบนเครือข่ายแอตฮอกและการระบุตำแหน่งให้มีความถูกต้องและแม่นยำ และเพื่อให้การแก้ไขนั้นเสร็จเรียบร้อย จึงมีการเปลี่ยนแปลงแผนการดำเนินงานในขั้นตอนนี้ จากนั้นจึงเริ่มพัฒนาแอปพลิเคชันสำหรับค้นหาตำแหน่งของบุคคลภายในอาคาร โดยสามารถแสดงแผนการดำเนินงานใหม่ได้ดังนี้

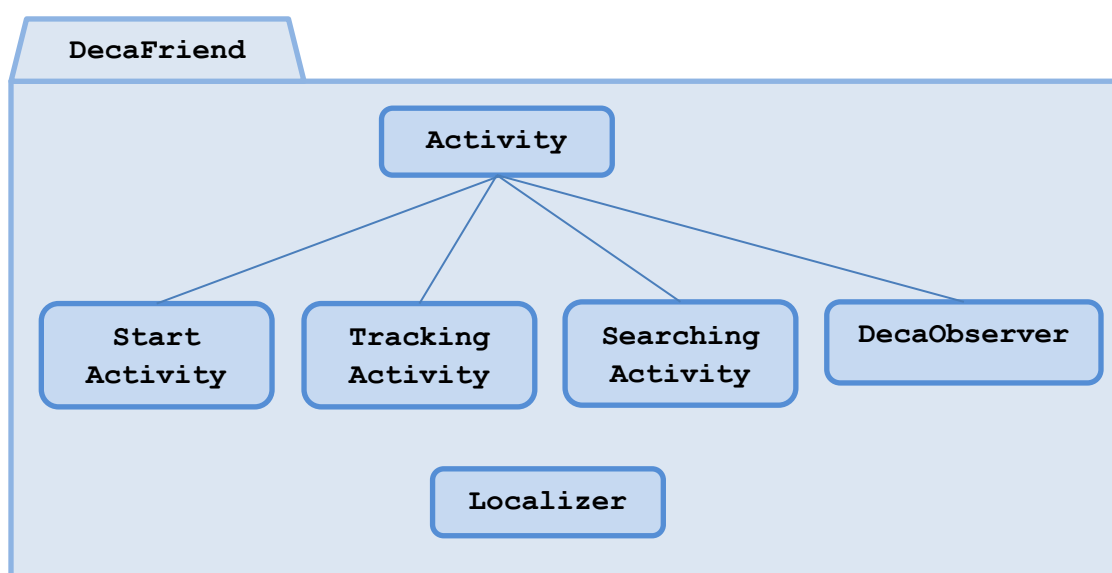
24 พฤศจิกายน – 17 ธันวาคม 2555	พัฒนาแอปพลิเคชันบนแอนดรอยด์ต่อเนื่องจากเดิม
18 ธันวาคม – 28 ธันวาคม 2555	ทดสอบการทำงานของแอปพลิเคชัน
31 ธันวาคม 2555 -14 มกราคม 2556	แก้ไขข้อผิดพลาดของแอปพลิเคชัน
15 มกราคม – 18 มกราคม 2556	ทำรายงานฉบับสมบูรณ์
21 มกราคม 2556	ส่งรายงานฉบับสมบูรณ์ให้อาจารย์ที่ปรึกษาตรวจและแก้ไข
22 มกราคม – 1 กุมภาพันธ์ 2556	แก้ไขรายงานฉบับสมบูรณ์
4 กุมภาพันธ์ 2556	ส่งรายงานฉบับสมบูรณ์ที่อาจารย์ที่ปรึกษาตรวจแก้และลงนามแล้ว
5 กุมภาพันธ์ – 14 กุมภาพันธ์ 2556	ทำ Poster presentation
8 กุมภาพันธ์ 2556	สอบ Senior Project
15 กุมภาพันธ์ 2556	นำเสนอ Poster presentation
22 กุมภาพันธ์ 2556	ส่งรายงานฉบับสมบูรณ์พร้อมแฟ้มข้อมูล

แผนการดำเนินงานขั้นถัดไป

การปรับปรุงประสิทธิภาพในการระบุตำแหน่งของแอปพลิเคชัน

จากการสืบค้นข้อมูลผ่านทางอินเทอร์เน็ต เพื่อหาวิธีในการปรับปรุงประสิทธิภาพในการระบุตำแหน่งของแอปพลิเคชันให้ดียิ่งขึ้น พบว่าสามารถเรียกดูค่า RSSI ด้วยการเขียนโปรแกรมภาษาจาวาและใช้ค่า RSSI ดังกล่าวในการวิเคราะห์ระยะห่างจาก Access Point ที่ได้รับสัญญาณ ผู้พัฒนาแอปพลิเคชันจึงมีแนวคิดที่จะนำค่า RSSI มาช่วยในการปรับปรุงประสิทธิภาพของแอปพลิเคชันให้มีความแม่นยำในการระบุตำแหน่งมากยิ่งขึ้น

การออกแบบแอปพลิเคชันสำหรับการแพร่กระจายและค้นหาตำแหน่งบนระบบปฏิบัติการแอนดรอยด์



- Activity ทำหน้าจัดการเกี่ยวกับการทำงานของแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์
 - StartActivity ทำหน้าที่ตั้งค่าเริ่มต้นในการใช้งานแอปพลิเคชัน
 - TrackingActivity ทำหน้าที่ระบุตำแหน่งของผู้ใช้งานและแพร่กระจายข้อความไปยังผู้ใช้งานอื่นๆในเครือข่าย
 - SearchingActivity ทำหน้าที่ติดตามและค้นหาตำแหน่งผู้ใช้งานอื่นๆที่อยู่ในเครือข่าย
 - DecaObserver จะถูกเรียกใช้งานเมื่อได้รับข้อความใหม่
- Localizer ทำหน้าที่ระบุตำแหน่งจากข้อมูลของสัญญาณที่ได้รับจาก Access Point

การทดสอบประสิทธิภาพในการทำงานของแอปพลิเคชันสำหรับการแพร่กระจายและค้นหาตำแหน่ง

ทดสอบความสามารถของแอปพลิเคชันในการระบุตำแหน่งและแพร่กระจายตำแหน่งให้กับโหนดอื่นๆ และทดสอบการรับข้อความที่ระบุตำแหน่งจากโหนดอื่นๆที่อยู่ในระบบ โดยการทดสอบนี้ผู้พัฒนาจะใช้โทรศัพท์มือถือ Google Nexus One ในการทดสอบ

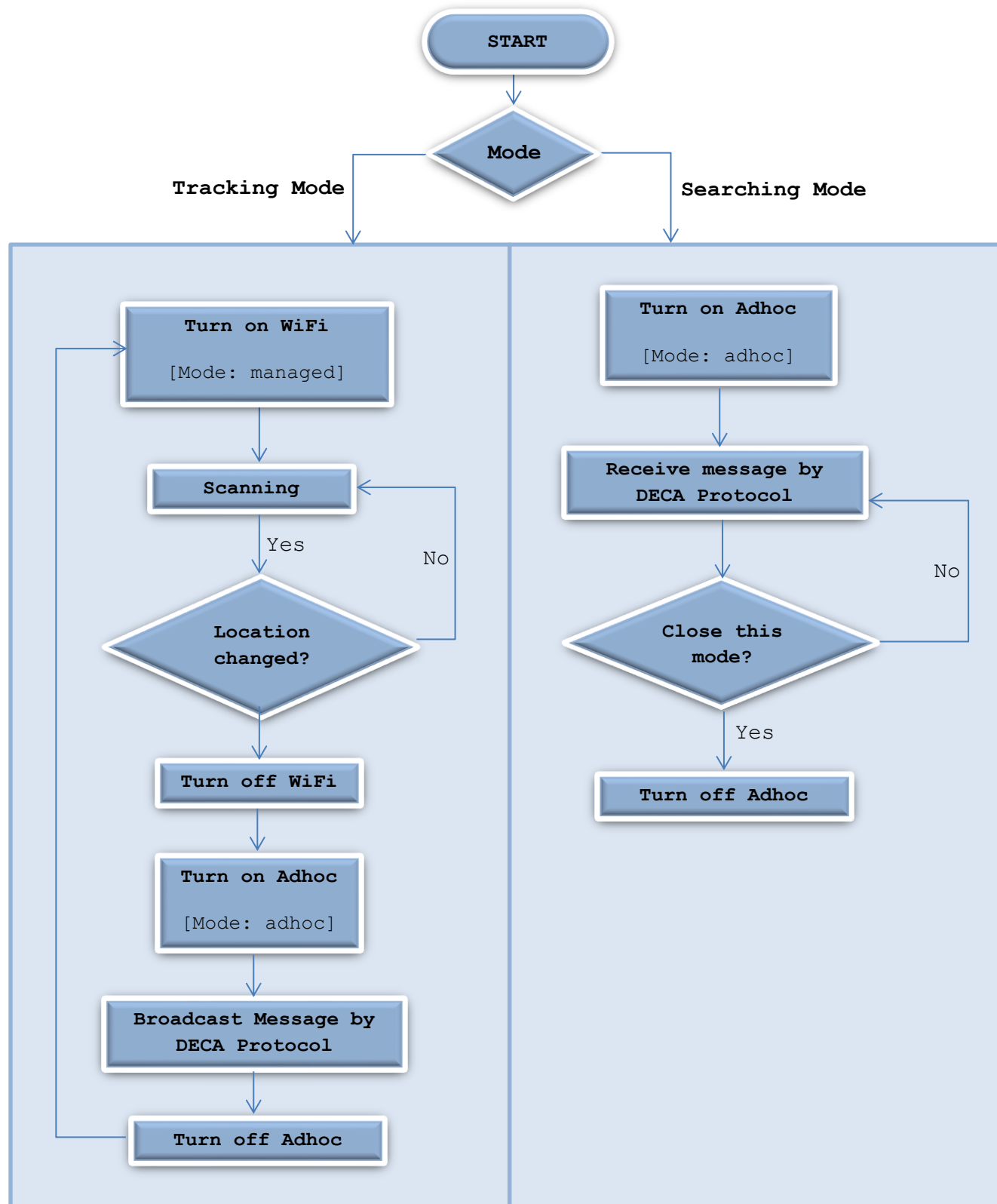
อุปสรรคและแนวทางแก้ไข

เมื่อเริ่มต้นทำโครงการ ผู้พัฒนามีเป้าหมายที่จะพัฒนาแอปพลิเคชันสำหรับค้นหาตำแหน่งของบุคคลภายในอาคารเจริญวิศวกรรม ระหว่างชั้น 17 ถึงชั้น 20 แต่เนื่องจากการค้นหาตำแหน่งของบุคคลจำเป็นต้องอาศัยข้อมูลสัญญาณจากสถานีแม่ข่ายในการระบุตำแหน่ง ซึ่งสัญญาณจากสถานีแม่ข่ายระหว่างชั้น 17 ถึงชั้น 20 นั้นค่อนข้างรบกวนกันเป็นอย่างมาก ทำให้ไม่สามารถระบุตำแหน่งของบุคคลได้อย่างถูกต้อง ผู้พัฒนาจึงเปลี่ยนสถานที่ทำการทดลองแอปพลิเคชันไปเป็นหอพักพวงชมพูหญิง 1 ซึ่งสัญญาณจากสถานีแม่ข่ายไม่รบกวนกัน จึงทำให้สามารถระบุตำแหน่งของบุคคลได้อย่างถูกต้องและแม่นยำมากกว่า

เนื่องจากการพัฒนาแอปพลิเคชันสำหรับค้นหาตำแหน่งของบุคคลภายในอาคาร จำเป็นต้องอาศัยข้อมูลจากสถานีแม่ข่ายในการระบุตำแหน่งของบุคคล จึงต้องมีการเชื่อมต่อเครือข่ายแลนไร้สายเพื่อรับสัญญาณจากสถานีแม่ข่าย (Access Point) ทั้งหมดที่อยู่ในบริเวณนั้น แต่เนื่องจากแอปพลิเคชันที่พัฒนาขึ้นจะใช้การกระจายข้อมูลตำแหน่งของบุคคล (Broadcast) ผ่านเครือข่ายแอดฮอค ซึ่งจำเป็นต้องใช้อุปกรณ์ Wireless card ที่มีเพียงอันเดียวของโทรศัพท์มือถือ Google Nexus One และจากการศึกษาข้อมูล[5] ในการเชื่อมต่อเครือข่ายแอดฮอคเมื่อโทรศัพท์มือถืออยู่ในโหมดแอดฮอคแล้วจะไม่สามารถเชื่อมต่อกับสถานีแม่ข่ายได้ เพราะการเชื่อมต่อกับสถานีแม่ข่ายก็จำเป็นต้องใช้อุปกรณ์ Wireless card เช่นเดียวกัน จึงทำให้ไม่สามารถทำการรับสัญญาณจากสถานีแม่ข่ายพร้อมกับแพร่กระจายข้อมูลไปยังโหนดอื่นๆได้ ภายใต้ทรัพยากรที่มีอยู่อย่างจำกัด ผู้พัฒนาจึงจำเป็นต้องคิดลำดับการทำงานของแอปพลิเคชันขึ้นมาใหม่เพื่อแก้ไขปัญหาดังกล่าว โดยลำดับการทำงานของแอปพลิเคชันเป็นดังนี้

เมื่อผู้ใช้งานเข้าใช้งานแอปพลิเคชัน แอปพลิเคชันจะเปิดการเชื่อมต่อแลนไร้สายให้อยู่ในโหมด managed เพื่อรับสัญญาณจากสถานีแม่ข่าย หากเมื่อใดก็ตามที่แอปพลิเคชันพบว่าผู้ใช้งานมีการเปลี่ยนตำแหน่งจะปิดการเชื่อมต่อในโหมด managed หลังจากนั้นจะเปิดการใช้งานเครือข่ายในโหมด adhoc เพื่อแพร่กระจาย (Broadcast) ตำแหน่งของตนเองไปยังโหนดอื่นๆในระบบด้วยโปรโตคอล DECA หลังจากแพร่กระจายข้อความเสร็จจะกลับมาสู่โหมด managed เพื่อค้นหาสัญญาณจากสถานีแม่ข่ายสำหรับระบุตำแหน่งอีกครั้ง

แต่ถ้าหากผู้ใช้ต้องการค้นหาตำแหน่งของผู้ใช้คนอื่นๆที่กำลังใช้งานแอปพลิเคชัน สามารถทำได้โดยการเปิดโหมดค้นหา การเปิดโหมดค้นหาจะเป็นการเปิดใช้งานเครือข่ายในโหมด adhoc เพื่อรับตำแหน่งของผู้ใช้คนอื่นๆที่ได้มีการแพร่กระจายข้อความระบุตำแหน่งมา



สรุปผลการดำเนินงาน

ปัจจุบันผู้พัฒนาได้ดำเนินงานมาจนถึงขั้นตอนที่ 17 จากแผนการดำเนินงานทั้งหมด 28 ขั้นตอน โดยขั้นตอนที่สำเร็จแล้ว 17 ขั้นตอนแรกได้แก่

- ส่งหัวข้อโครงการ
- กำหนดขอบเขตของการวิจัย
- ศึกษาเครือข่ายแอตตอกและโปรโตคอล DECA
- ศึกษาเครือข่ายแลนไร้สายและการเชื่อมต่อของเครือข่าย
- ศึกษาการพัฒนาแอปพลิเคชันบนแอนดรอยด์
- ศึกษางานวิจัยที่เกี่ยวข้องและเครื่องมือที่จำเป็นในการวิจัย
- ทำรายงาน Project Proposal
- ส่งรายงาน Project proposal ต่อคณะกรรมการ
- เตรียม Slide สำหรับนำเสนอ Project proposal
- สอบ Project proposal
- พัฒนาการส่งข้อมูลบนเครือข่ายแอตตอกและการระบุตำแหน่ง
- พัฒนาแอปพลิเคชันบนแอนดรอยด์
- ทำรายงานความก้าวหน้าของโครงการ
- ส่งรายงานความก้าวหน้าของโครงการให้อาจารย์ที่ปรึกษาตรวจแก้
- ส่งรายงานความก้าวหน้าของโครงการ
- เตรียม Slide สำหรับนำเสนอความก้าวหน้าของโครงการ
- นำเสนอความก้าวหน้าของโครงการต่อคณะกรรมการ

ขั้นตอนที่กำลังดำเนินการอยู่คือ

- พัฒนาแอปพลิเคชันบนแอนดรอยด์ต่อเนื่องจากเดิม

ขั้นตอนที่จะดำเนินการถัดไปคือ

- ทดสอบการทำงานของแอปพลิเคชัน
- แก้ไขข้อผิดพลาดของแอปพลิเคชัน
- ทำรายงานฉบับสมบูรณ์
- ส่งรายงานฉบับสมบูรณ์ให้อาจารย์ที่ปรึกษาตรวจและแก้ไข
- แก้ไขรายงานฉบับสมบูรณ์
- ส่งรายงานฉบับสมบูรณ์ที่อาจารย์ที่ปรึกษาตรวจแก้และลงนามแล้ว
- ทำ Poster presentation
- สอบ Senior Project
- นำเสนอ Poster presentation
- ส่งรายงานฉบับสมบูรณ์พร้อมแฟ้มข้อมูล

ผู้พัฒนาได้ศึกษาการเขียนแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ และได้เริ่มเขียนแอปพลิเคชันไปบางส่วนแล้ว ผู้พัฒนาจึงมีความพร้อมที่จะดำเนินงานในขั้นตอนถัดไป ซึ่งก็คือการทดสอบแอปพลิเคชันและแก้ไขข้อผิดพลาดของแอปพลิเคชัน และผู้พัฒนาคาดว่าโครงการนี้จะเสร็จสมบูรณ์ตามแผนการดำเนินงาน

รายการอ้างอิง

- [1] Google Inc. *Android Developers* [online]. [cited 23 Aug, 2012]. Available from:
<http://developer.android.com/index.html>
- [2] N. Na Nakorn, and K. Rojviboonchai, "DECA: Density-Aware Reliable Broadcasting in Vehicular Ad-Hoc Networks," *IEEE the 7th ECTI-CON2010*, Chiang Mai, Thailand, May 19-21, 2010.
- [3] N. Na Nakorn, and K. Rojviboonchai, "Efficient Beacon Solution for Wireless Ad-Hoc Networks," *the 7th JCSSE2010*, Bangkok, Thailand, May 12-14, 2010.
- [4] Prasant Mohapatra and Srikanth V. Krishnamurthy. *Ad Hoc Networks: Technologies and Protocols*. Springer, 2005.
- [5] TechTarget. Infrastructure mode and ad hoc mode at the same time? [online]. [cited 30 Oct, 2012]. Available from: <http://searchmobilecomputing.techtarget.com/answer/Infrastructure-mode-and-ad-hoc-mode-at-the-same-time>