



## เรื่อง คู่มือผู้พัฒนา การติดตั้ง และแนวทางการปฏิบัติงานเป็นทีม

เสนอ

อ.ดร.บุญรัตน์ เผติมรอต

จัดทำโดย

นาย วิศวะ โชติพันธุ์พงศ์ 6220500725

นาย นักรบ สุขใจ 6220502141

นาย ไตรรัตน์ตรี เกษน้ำใส 6220503210

นาย เพชรกล้า กมลสุขยืนยง 6220504721

นาย ภควัตร บุญศรี 6220504747

รายงานนี้เป็นส่วนหนึ่งของวิชา Software Engineering 02204371-60

ภาคเรียนที่ 2 ปีการศึกษา 2564

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตกำแพงแสน

# เว็บระบบบริหารจัดการการฝึกงานแบบออนไลน์

## เอกสารประกอบโปรแกรมสำหรับผู้เขียนโปรแกรม (Technical Documentation)

### ที่มาและความสำคัญ

ด้วยระบบการยื่นเรื่องฝึกงานในปัจจุบันนั้น ยังต้องรอทางภาควิชาในการนำไฟล์เอกสารมาประกาศทางเว็บไซต์ นิสิตจึงไม่สามารถสืบค้นสถานประกอบการที่มีความประสงค์รับนิสิตฝึกงานล่วงหน้าได้ และยังยากต่อการติดตามผลเนื่องจากนิสิตจะต้องติดต่อเจ้าหน้าที่ภาควิชาเพื่อติดตามผลการยื่นคำร้องขอฝึกงาน ในส่วนของอาจารย์นั้นต้องจัดการเอกสาร Excel ซึ่งมีความยุ่งยาก และมีการแก้ไขที่บ่อยครั้ง ส่งผลให้การตรวจสอบข้อมูลการฝึกงานของนิสิตเพื่ออนุมัติการจบการศึกษามีความล่าช้าและไม่สะดวกรวดเร็ว

เพื่อออกแบบและพัฒนาระบบจัดการการฝึกงานแบบออนไลน์สำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ กำแพงแสน มหาวิทยาลัยเกษตรศาสตร์ และสามารถจัดทำฐานข้อมูลที่เกี่ยวข้องกับการฝึกงานของนิสิตภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ กำแพงแสน มหาวิทยาลัยเกษตรศาสตร์ และสามารถปรับปรุงการจัดการการฝึกงานภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ กำแพงแสน มหาวิทยาลัยเกษตรศาสตร์

### วัตถุประสงค์ของการพัฒนาระบบ

1. เพื่อออกแบบและพัฒนาระบบจัดการการฝึกงานแบบออนไลน์สำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ กำแพงแสน มหาวิทยาลัยเกษตรศาสตร์
2. เพื่อจัดทำฐานข้อมูลที่เกี่ยวข้องกับการฝึกงานของนิสิตภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ กำแพงแสน มหาวิทยาลัยเกษตรศาสตร์
3. เพื่อปรับปรุงการจัดการการฝึกงานภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ กำแพงแสน มหาวิทยาลัยเกษตรศาสตร์

## วิธีการใช้งาน

เว็บไซต์แบ่งการใช้งานแบ่งผู้ใช้ออกเป็น 3 ประเภท คือ

1. ผู้ใช้ นิสิต นักศึกษา คุณสมบัติ เป็นผู้ที่ศึกษาอยู่ใน ชั้นปีที่ สาม ต้องการที่จะไปฝึกงาน สามารถดาวน์โหลดเอกสาร และยื่นคำร้องขอไปฝึกงานได้ผ่านเว็บไซต์
2. ผู้ใช้ อาจารย์ สามารถ อนุมัติ หรือ ไม่อนุมัติ คำร้องไปฝึกงาน
3. ผู้ใช้ บุคคลทั่วไป สามารถ ดูประกาศต่างๆที่แจ้งเตือนในหน้าเว็บ และสามารถดูข่าวสารของบริษัทต่างๆผ่านทางหน้าเว็บไซต์ได้

## แนวคิดเกี่ยวกับการออกแบบโปรแกรม

ทางผู้พัฒนาเล็งเห็นปัญหาของผู้ใช้งานในการค้นหาสถานที่ฝึกงานและทำสภกิจที่มีความยุ่งยากข้อมูลและรายละเอียดไม่ครบตามความต้องการหรือต้องค้นหาจากหลายแหล่งข้อมูลเพื่อที่จะได้ข้อมูลที่ครบตามความต้องการ ทางผู้พัฒนาจึงจัดทำเว็บไซต์นี้ขึ้นมาเพื่อช่วยให้ผู้ใช้งานได้ค้นหาสถานที่ฝึกงานและดาวน์โหลดเอกสาร ได้สะดวกและรวดเร็วมากยิ่งขึ้น

## รายละเอียดภาษาและโปรแกรมที่ใช้

### ภาษา Java Script

ทางผู้พัฒนาใช้ภาษา java script ในรูปแบบ jsx สำหรับสร้างหน้าเว็บไซต์ ส่วน < head> ประกอบด้วยข้อมูล เช่น ชื่อ คำอธิบาย และลิงก์ไปยัง CSS ส่วน < body> ประกอบด้วยส่วนที่เป็นเนื้อหาของเว็บเพจ เช่น ข้อความ รูปภาพ พื้นหลัง สีตัวอักษร และแบบฟอร์มต่าง ๆ

### React

เป็น Library ที่ใช้ในการพัฒนาเว็บไซต์

## ภาษา CSS

ทางผู้พัฒนาใช้ภาษา CSS เพื่อพัฒนาลักษณะรูปแบบ ใส่พื้นหลัง หรือเพิ่มกรอบข้อความ การจัดเลย์เอาท์ และการทำให้ภาพลักษณ์ของเว็บสวยงามมากขึ้น

## โปรแกรม GitHub

เป็นเว็บบริการที่เก็บ Repository บนออนไลน์แพลตฟอร์ม พื้นที่ทางอินเทอร์เน็ต (hosting service) สำหรับเก็บการควบคุมการปรับปรุงแก้ไข (version control) โดยใช้ Git สามารถเข้าถึงและทำงานร่วมกับคนอื่นได้ผ่านหน้าเว็บไซต์ได้ทุกที่ทุกเวลา

## โปรแกรม Xampp

เป็นโปรแกรม Apache web server ไว้จำลอง web server เพื่อไว้ทดสอบเว็บไซต์

## โปรแกรม Visual Studio Code

เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไขและปรับแต่งโค้ด

## โปรแกรม phpMyAdmin

เป็นโปรแกรมที่ถูกพัฒนาโดยใช้ภาษา PHP เพื่อใช้ในการบริหารจัดการฐานข้อมูล Mysql แทนการคีย์คำสั่ง สามารถสร้างและลบ Database ได้

## Project Team lesson learn

การกำหนดขอบเขตของงาน ให้แคบลงเพื่อได้ทำงานได้ตรงประเด็น

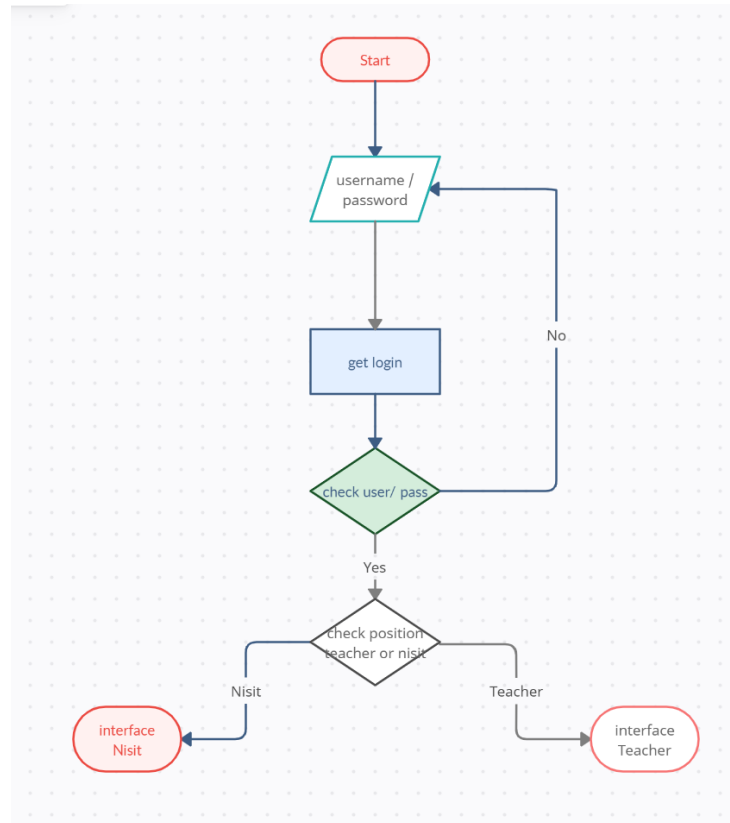
-แบ่งงานตามความสามารถและจุดเด่นของแต่ละคน

-ช่วยเหลือการทำงานของสมาชิกในทีม เมื่อติดปัญหา

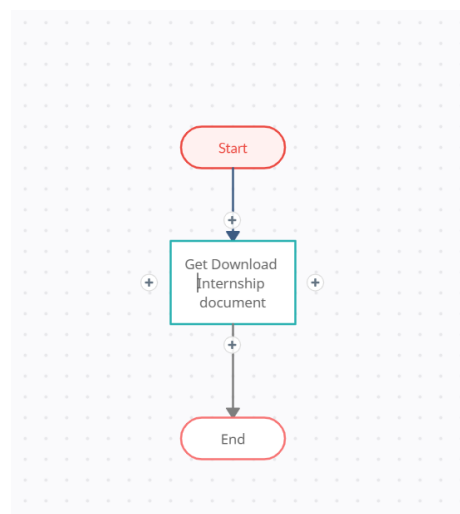
-การกำหนดผู้นำสำหรับการทำงานนั้นๆ

-กำหนดกระบวนการทำงานและแนวทางการปฏิบัติงาน

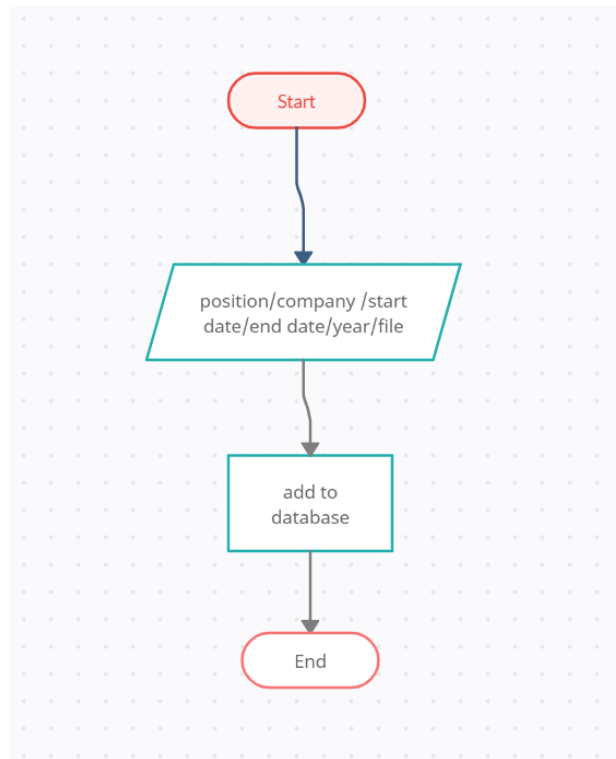
Login flow chart



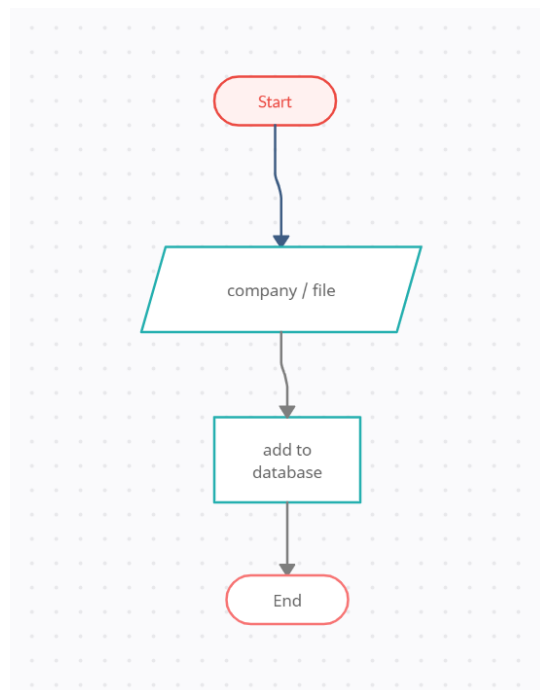
Download document flowchart



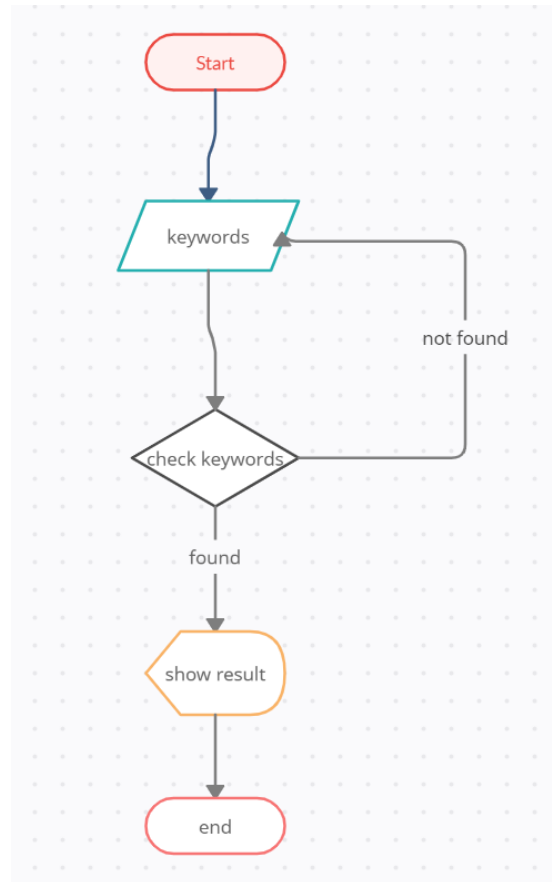
Add\_Nisit\_Internship flowchart



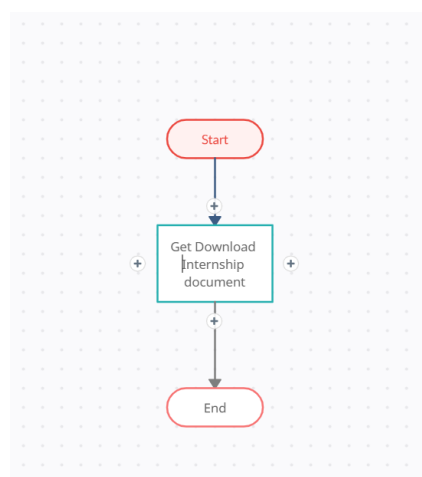
Apply Internship flowchart



Teacher search flowchart

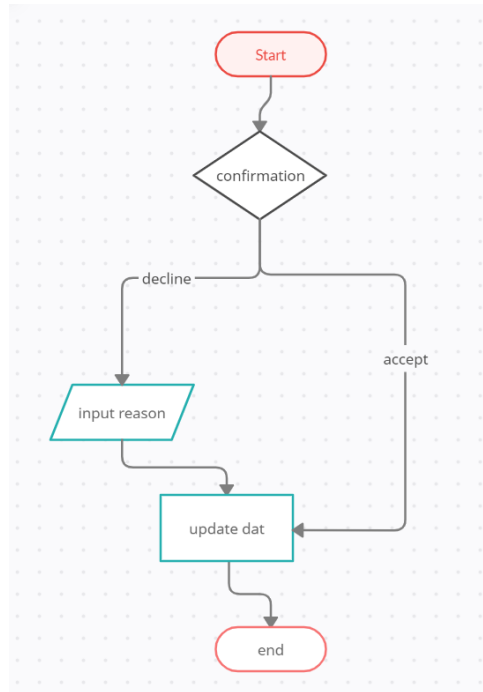


Teacher Download file flowchart





Update status for teacher flowchart



## Function ส่วน client

หน้า Login

ดึงข้อมูล user จาก Server มาใช้ตรวจสอบการ login

```
const [users, setUser] = useState([]);
const [admins, setAdmin] = useState([]);

if (start === 1)
  Axios.get("http://" + ip + ":3001/loginnisit").then((response) => {
    setUser(response.data);
    start = 0;
  });

if (end === 1)
  Axios.get("http://" + ip + ":3001/loginadmin").then((response) => {
    setAdmin(response.data);
    end = 0;
  });
```

Function getVal ทำหน้าที่ ตรวจสอบการ login ในระบบ

```
const getVal = () => {
  for (let i = 0; i < users.length; i++) {

    if (email === users[i].email && password === users[i].password) {
      const path = "/new/"+users[i].ln_id;
      navigate(path);
      return;
    }
  }
  for(let i = 0 ; i < admins.length ; i++){
    if (email === admins[i].email && password === admins[i].password) {

      if(admins[i].po_name === 'admin'){
        navigate("/new-admin" );
        return;
      }
      else if(admins[i].po_name === "teacher"){
        const path = "/new-t/"+users[i].la_id;
        navigate(path);
        return;
      }
    }
  }
};
```

หน้า New

ดึงข้อมูล Company จาก Server มาแสดงผล

```
const [companyList, setCompanyList] = useState([]);
const getCompany = () => {
  if (start === 1)
    Axios.get("http://" + ip + ":3001/company").then((response) => {
      setCompanyList(response.data);
      // alert(location.state.name);
      // alert(props.location.name)
      start = 0;
    });
};
```

หน้า Internship(nisit)

Function getInterndoc ทำหน้าที่ ดึงข้อมูล Internship\_Doc จาก Server มาแสดงผล

```
function getinterndoc() {
  Axios.get("http://" + ip + ":3001/internshipdocument").then((response) => {
    const fdata = response.data;
    const itemdata = [];

    for (let i in fdata) {
      if (fdata[i].ln_id === id) {
        itemdata.push(fdata[i]);
      }
    }
    setdata(itemdata);
  });
}
```

Function getInternresult ทำหน้าที่ดึงข้อมูล Internship\_Result จาก Server มาแสดงผล

```
function getinternresult() {  
  Axios.get("http://" + ip + ":3001/internshipresult").then((response) => {  
    const fdata = response.data;  
    const itemdata = [];  
  
    for (let i in fdata) {  
      if (fdata[i].ln_id === id) {  
        itemdata.push(fdata[i]);  
      }  
    }  
    setdataresult(itemdata);  
  });  
}
```

หน้า addInternship(nisit)

Function addinternship ทำหน้าที่ ส่งข้อมูลขึ้นไปเก็บบน Server

```
const addinternship = () => {

  if (cp_id == null || baseImage == null || id_position == null || id_year == null) {
    return
  }

  Axios.post("http://" + ip + ":3001/create/internshipdocument", {
    ln_id: ln_id,
    id_date: id_date,
    cp_id: cp_id,
    id_position: id_position,
    id_sdate: id_sdate,
    id_edate: id_edate,
    id_file: baseImage,
    id_tfile1: id_tfile1,
    id_tfile2: id_tfile2,
    id_status: id_status,
    id_confirm: id_confirm,
    id_year: id_year,
  }).then(() => {
    setinternship([
      ...internship,
      {
        ln_id: ln_id,
        id_date: id_date,
        cp_id: cp_id,
        id_position: id_position,
        id_sdate: id_sdate,
        id_edate: id_edate,
        id_file: baseImage,
        id_tfile1: id_tfile1,
        id_tfile2: id_tfile2,
        id_status: id_status,
        id_confirm: id_confirm,
        id_year: id_year,
      },
    ],);
  });
  alert("submit successful");
  navigate("/internship/"+id);
};
```

Function convertBase64 ทำหน้าที่ แปลงไฟล์ เป็น Base64 เพื่อส่งข้อมูลขึ้น Server

```
const convertBase64 = (file) => {  
  return new Promise((resolve, reject) => {  
    const fileReader = new FileReader();  
    fileReader.readAsDataURL(file);  
  
    fileReader.onload = () => {  
      resolve(fileReader.result);  
    };  
  
    fileReader.onerror = (error) => {  
      reject(error);  
    };  
  });  
};
```

หน้า addInternshipresult(nisit)

Function addinternship ทำหน้าที่ ส่งข้อมูลขึ้นไปเก็บบน Server

```
const addinternship = () => {  
  
  if (cp_id == null || baseImage == null ) {  
    return  
  }  
  Axios.post("http://" + ip + ":3001/create/internshipresult", {  
    ln_id: ln_id,  
    ir_date: id_date,  
    cp_id: cp_id,  
    ir_image: baseImage,  
    ir_status: id_status,  
  }).then(() => {  
    setinternship([  
      ...internship,  
      {  
        ln_id: ln_id,  
        ir_date: id_date,  
        cp_id: cp_id,  
        ir_image: baseImage,  
        ir_status: id_status,  
      },  
    ],  
  );  
});  
  
  alert("submit successful");  
  navigate("/internship/" + id);  
};
```



หน้า Internship\_Doc(teacher)

Function searcher และ Function searchfilter ทำหน้าที่ จัดเรียงข้อมูลและค้นหาข้อมูลตามความต้องการของผู้ใช้ Server

```
const searchbar = (key, data) => {
  if (key !== "") {
    const filteredData = data.filter((item) => {
      return Object.values(item)
        .join("")
        .toLowerCase()
        .includes(key.toLowerCase());
    });
    setInternshipdocumentList(filteredData);
  } else {
    setInternshipdocumentList(data);
  }
};

const searchfilter = (filter, key) => {
  if (filter == "ทั้งหมด") {
    Axios.get("http://" + ip + ":3001/internshipdocument").then(
      (response) => {
        // setInternshipdocumentList(response.data);
        searchbar(key, response.data);
        // alert(response.data)
      }
    );
  } else if (filter == "สถานะ") {
    Axios.get("http://" + ip + ":3001/internshipdocument/orderstatus").then(
      (response) => {
        // setInternshipdocumentList(response.data)
        searchbar(key, response.data);
        // alert(response.data)
      }
    );
  } else if (filter == "วันที่ยื่นขอ") {
    Axios.get("http://" + ip + ":3001/internshipdocument/orderdate").then(
      (response) => {
        // setInternshipdocumentList(response.data);
        searchbar(key, response.data);
        // alert(response.data)
      }
    );
  } else if (filter == "ปีการศึกษา") {
    Axios.get("http://" + ip + ":3001/internshipdocument/orderyear").then(
      (response) => {
        // setInternshipdocumentList(response.data)
        searchbar(key, response.data);
        // alert(response.data)
      }
    );
  }
};
```

Function updateFile ทำหน้าที่ อัปเดตสถานะของ internship แต่ละใบบน Server

```
const updateFile = (status, id_id) => {
  Axios.put("http://" + ip + ":3001/update/internshipdocument", {
    id: id_id,
    id_comment: newComment,
    status: status,
  });
};
```

หน้า Internship\_Result(teacher)

Function getResult ทำหน้าที่ ดึงข้อมูล internshipResult จากบน Server มาแสดงผล

```
const getResult = () => {
  if (start === 1)
    Axios.get("http://" + ip + ":3001/internshipresult").then((response) => {
      setinternshipResult(response.data);
      start = 0;
    });
};
```

Function `updateStatus` ทำหน้าที่ อัปเดตสถานะของ `internshipResult` แต่ละใบบน Server

```
const updateStatus = (status, id, a) => {  
  btn = a;  
  
  Axios.put("http://" + ip + ":3001/update/internshipresult", {  
    id: id,  
    status: status,  
    detail: comment,  
  }).then((response) => {});  
  if (btn == 1) {  
    refreshPage();  
  }  
};
```

## Function ส่วน Server

### Read company

ทำหน้าที่ดึงข้อมูลจากตาราง company ในดาตาเบสมาแสดงบนเว็บ

```
app.get('/company', (req, res) => {  
  db.query("SELECT * FROM company", (err, result) => {  
    if(err) {  
      console.log(err);  
    } else {  
      res.send(result);  
    }  
  });  
});
```

### Add company

ทำหน้าที่เพิ่มข้อมูลลงตาราง company ในดาตาเบส

```
app.post("/create/company", (req, res) => {  
  const id = req.body.id;  
  const name = req.body.name;  
  const link = req.body.link;  
  
  db.query(  
    "INSERT INTO company (cp_id, cp_name, cp_link) VALUES (?, ?, ?)",  
    [id, name, link],  
    (err, result) => {  
      if (err) {  
        console.log(err);  
      } else {  
        res.send("Values Inserted");  
      }  
    }  
  );  
});
```

## Update company

หน้าที่อัปเดตข้อมูลของบริษัทลงในดาตาเบส

```
app.put("/update/company", (req, res) => {
  const id = req.body.id;
  const name = req.body.name;
  const link = req.body.link;

  db.query(
    "UPDATE company SET cp_name = ? , cp_link = ? WHERE cp_id = ?",
    [name, link ,id],
    (err, result) => {
      if (err) {
        console.log(err);
      } else {
        res.send(result);
      }
    }
  );
});
```

## Delete company

ทำหน้าที่ลบข้อมูลบริษัทออกจากดาตาเบส

```
app.delete("/delete/company", (req, res) => {
  const id = req.params.id;
  db.query("DELETE FROM company WHERE cp_id = ?", id, (err, result) => {
    if (err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```

Read loginnisit

ทำหน้าที่ดึงข้อมูลจากตาราง loginnisit ในดาตาเบสเพื่อมาใช้ในการ Login

```
app.get('/loginnisit', (req,res) =>{
  db.query("SELECT ln_id ,email , password ,ln_name FROM loginnisit", (err, result) => {
    if(err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```

Read loginadmin

ทำหน้าที่ดึงข้อมูลจากตาราง loginadmin ในดาตาเบสเพื่อมาใช้ในการ Login

```
app.get('/loginadmin', (req,res) =>{
  db.query("SELECT email , password , po_name FROM loginadmin INNER JOIN position AS p ON p.po_id = loginadmin.po_id", (err, result) => {
    if(err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```

Read internshipdocument

ทำหน้าที่ดึงข้อมูลจากตาราง internshipdocument ในดาตาเบสมาแสดงบนเว็บ

```
app.get('/internshipdocument', (req,res) =>{
  db.query("SELECT id_comment,id_id , ln_id , ln_name ,id_year , email ,id_date , cp_name , `id_position`,
`id_sdate`,`id_edate`,`id_file`,`id_status`,`id_confirm` FROM internshipDocument JOIN company USING(cp_id) JOIN loginNisit USING(ln_id)",
  (err, result) => {
    if(err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```

Read internshipdocument

ทำหน้าที่ดึงข้อมูลจากตาราง internshipdocument ในดาตาเบสมาแสดงบนเว็บ โดยจะแสดงแค่รหัสனிสิตแค่ชื่อ

```
app.get('/internshipdocument', (req,res) =>{
  db.query("SELECT ln_id , ln_name FROM internshipDocument JOIN loginNisit USING(ln_id)", (err, result) => {
    if(err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```

Read internshipdocument/orderstatus

ทำหน้าที่ดึงข้อมูลจากตาราง internshipdocument ในดาตาเบสมาแสดงบนเว็บ โดยเรียงตามสถานะของการอนุมัติ ได้แก่ pending , disapprove และ approve ตามลำดับ

```
app.get('/internshipdocument/orderstatus', (req,res) =>{
  db.query("SELECT id_comment,id_id , ln_id , ln_name ,id_year , email ,id_date , cp_name
, `id_position`, `id_sdate`,`id_edate`,`id_file`,`id_status`,`id_confirm`
FROM internshipDocument JOIN company USING(cp_id) JOIN loginNisit USING(ln_id) ORDER BY id status DESC",
(err, result) => {
    if(err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```

Read internshipdocument/orderdate

ทำหน้าที่ดึงข้อมูลจากตาราง internshipdocument ในดาตาเบสมาแสดงบนเว็บ โดยเรียงตามวันที่ส่งเอกสารล่าสุด

```
app.get('/internshipdocument/orderdate', (req,res) =>{
  db.query("SELECT id_comment,id_id , ln_id , ln_name ,id_year , email ,id_date , cp_name , `id_position`,
`id_sdate`,`id_edate`,`id_file`,`id_status`,`id_confirm` FROM internshipDocument
JOIN company USING(cp_id) JOIN loginNisit USING(ln_id) ORDER BY id id DESC", (err, result) => {
    if(err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```



Read internshipdocument/orderyear

ทำหน้าที่ดึงข้อมูลจากตาราง internshipdocument ในดาตาเบสมาแสดงบนเว็บ โดยเรียงตามปีการศึกษาของ นิสิต

```
app.get('/internshipdocument/orderyear', (req, res) => {
  db.query("SELECT id_comment, id_id , ln_id , ln_name , id_year , email , id_date , cp_name , `id_position`, `id_sdate`, `id_edate`, `id_file`, `id_status`, `id_confirm` FROM internshipDocument JOIN company USING(cp_id) JOIN loginNisit USING(ln_id) ORDER BY id_year DESC", (err, result) => {
    if(err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```

Add internshipdocument

ทำหน้าที่เพิ่มข้อมูลลงตาราง internshipdocument ในดาตาเบส

```
app.post("/create/internshipdocument", (req, res) => {
  const ln_id = req.body.ln_id;
  const id_date = req.body.id_date;
  const id_year = req.body.id_year;
  const cp_id = req.body.cp_id;
  const id_position = req.body.id_position;
  const id_sdate = req.body.id_sdate;
  const id_edate = req.body.id_edate;
  const id_file = req.body.id_file;
  const id_tfile1 = req.body.id_tfile1;
  const id_tfile2 = req.body.id_tfile2;
  const id_status = req.body.id_status;
  const id_confirm = req.body.id_confirm;

  db.query(
    "INSERT INTO internshipdocument (ln_id , id_date , cp_id , id_year , `id_position`, `id_sdate`, `id_edate`, `id_file`, `id_tfile1`, `id_tfile2`, `id_status`, `id_confirm`) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)",
    [ln_id , id_date, cp_id, id_year , id_position, id_sdate, id_edate, id_file , id_tfile1, id_tfile2, id_status, id_confirm],
    (err, result) => {
      if (err) {
        console.log(err);
      } else {
        res.send("Values Inserted");
      }
    }
  );
});
```

## Update company

หน้าที่อัปเดตข้อมูลของใบคำร้องขอฝึกงานลงในดาตาเบส

```
app.put("/update/internshipdocument", (req, res) => {
  const id = req.body.id;
  const status = req.body.status;
  const id_comment = req.body.id_comment;

  db.query(
    "UPDATE internshipdocument SET id_status = ? , id_comment = ? WHERE id_id = ?",
    [ status ,id_comment ,id],
    (err, result) => {
      if (err) {
        console.log(err);
      } else {
        res.send(result);
      }
    }
  );
});
```

## Read internshipresult

ทำหน้าที่ดึงข้อมูลจากตาราง internshipresult ในดาตาเบสมาแสดงบนเว็บ

```
app.get('/internshipresult', (req,res) =>{
  db.query("SELECT ir_id, ln_id , ln_name ,email , cp_name , ir_date , ir_image ,ir_status FROM internshipresult
  JOIN company USING(cp_id) JOIN loginNisit USING(ln_id)", (err, result) => {
    if(err) {
      console.log(err);
    } else {
      res.send(result);
    }
  });
});
```

Add internshipresult

ทำหน้าที่เพิ่มข้อมูลลงตาราง internshipresult ในดาตาเบส

```
app.post("/create/internshipresult", (req, res) => {
  const ln_id = req.body.ln_id;
  const cp_id = req.body.cp_id;
  const ir_date = req.body.ir_date;
  const ir_status = req.body.ir_status;
  const ir_image = req.body.ir_image;

  db.query(
    "INSERT INTO internshipresult (ln_id,cp_id ,ir_date,ir_status,ir_image) VALUES (?,?,,?,?)",
    [ln_id,cp_id,ir_date, ir_status, ir_image],
    (err, result) => {
      if (err) {
        console.log(err);
      } else {
        res.send("Values Inserted");
      }
    }
  );
});
```

Update company

หน้าที่อัปเดตผลการอนุมัติการฝึกงานลงในดาตาเบส

```
app.put("/update/internshipresult", (req, res) => {
  const id = req.body.id;
  const detail = req.body.detail;
  const status = req.body.status;

  db.query(
    "UPDATE internshipresult SET ir_status = ?, ir_detail = ? WHERE ir_id = ?",
    [ status ,detail,id],
    (err, result) => {
      if (err) {
        console.log(err);
      } else {
        res.send(result);
      }
    }
  );
});
```