

แบบรายงานความก้าวหน้าในการทำโครงการ

โครงการแพลตฟอร์มฝึกทักษะคณิตศาสตร์ระดับชั้นประถมศึกษาปีที่ 1 – 3 รายงานครั้งที่ 4

1. ภาพรวม (Summary)

งานด้านเทคนิค

งาน (Task)	นางสาววรรณนิษา พรหมบุตร	นายธนกฤต จันทร์านิมิตร	นายจักรดาว วณะทอง
การเช็คคำตอบว่าถูกหรือไม่ และการนับ ข้อถูก	9 ชั่วโมง	-	-
ศึกษาการบริหารจัดการบัญชีผู้ใช้	-	8 ชั่วโมง	-
ทดลองอัปโหลด API ขึ้น server	-	-	16 ชั่วโมง

งานทั่วไป

งาน (Task)	นางสาววรรณนิษา พรหมบุตร	นายธนกฤต จันทร์านิมิตร	นายจักรดาว วณะทอง
จัดพิมพ์เอกสาร	1 ชั่วโมง	1 ชั่วโมง	1 ชั่วโมง

2. ผลสัมฤทธิ์สำคัญในสัปดาห์ที่ผ่านมา (Highlights)

- 1. การเช็คคำตอบว่าถูกหรือไม่ และการนับข้อถูก
- 2. ทดลองอัปโหลด API ขึ้น server
- 3. ศึกษาการบริหารจัดการบัญชีผู้ใช้

3. รายละเอียดในการปฏิบัติงาน (Details)

3.1 การเช็คคำตอบว่าถูกหรือไม่ และการนับข้อถูก

สร้างฟังก์ชันการดึงข้อมูล

```
getdata(i){  
    i=this.index;  
  
    this.http.get(this.ex1Url)  
    .subscribe((data:any[])=>{  
        this.ex1=data[i];  
        this.ex1 = Array.of(this.ex1); ///when error [object]  
        console.log(data[i]);  
    });  
}
```

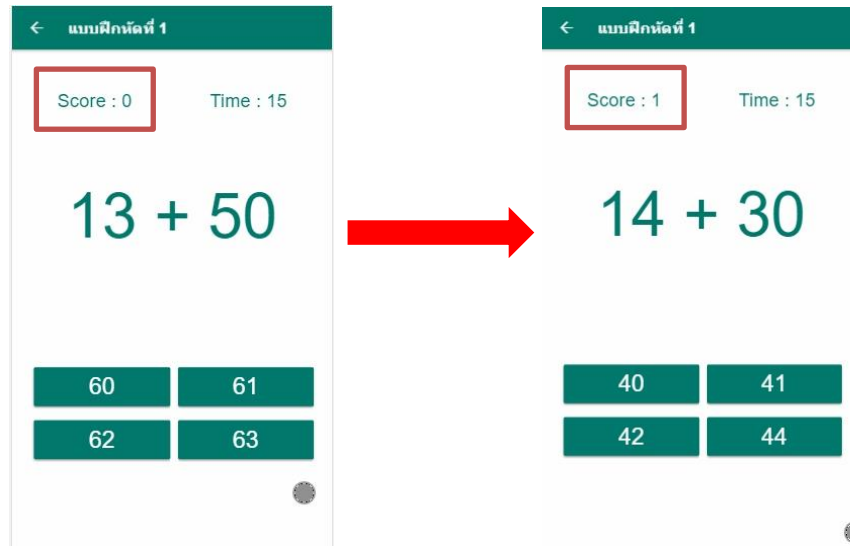
รูปที่ 1 การดึงข้อมูล

สร้างฟังก์ชันการเช็คคำตอบ

```
getanswer(i,j){  
    i=this.index;  
    var selectanswer = document.getElementById('test'+j).innerText;///get value from button  
    this.http.get(this.ex1Url)  
    .map((res:Response)=>res)  
    .subscribe(data=>{  
        this.answer=data[i].answer;  
        console.log(this.answer);  
  
        if(selectanswer == this.answer){  
            this.score = this.score+1;  
            // console.log(this.score);  
            this.index = this.index+1;  
            this.getdata(i+1);  
            // console.log('answer is true')  
        }else{  
            console.log('answer is false')  
            this.index = this.index+1;  
            this.getdata(i+1);  
        }  
    });  
}
```

รูปที่ 2 ฟังก์ชันการเช็คคำตอบ

เมื่อตอบคำถามถูกจะมีคะแนนเพิ่ม 1 และดึงข้อมูลข้อถัดไปมาแสดง



รูปที่ 3 แสดงการทำงาน

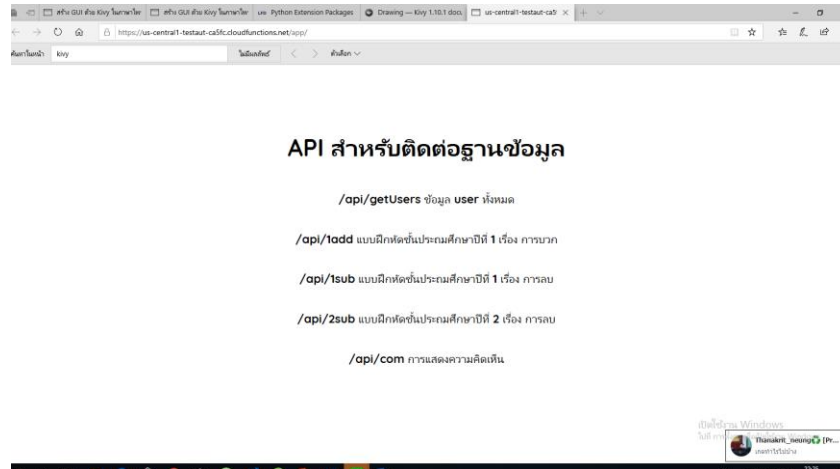
3.2 ทดลองอัปโหลด API ขึ้น server

ออกแบบ api สำหรับดึงข้อมูล node ต่างๆในfirebase และทำให้ apiสามารถ รับ พารามิเตอร์ ได้

```
// get users .child(uid) .equalTo(req.params.id).
app.get('/api/getUsers/:id', function(req, res) {
  var uid = "-LZj54S1LfKYwNYhrzLA";
  if (uid.length != 20) {
    res.json({message: "Error: uid must be 20 characters long."});
  } else {
    usersRef.orderByChild('ID').equalTo(req.params.id).once("value", function(snapshot) {
      if (snapshot.val() == null) {
        res.json({message: "Error: No user found", "result": false});
      } else {
        res.json( snapshot.val());
      }
    });
  }
});
```

รูปที่ 4 โปรแกรมการออกแบบ API

ทำการอัปโหลดตัว api ขึ้น server



รูปที่ 5 API

3.3 การบริหารจัดการบัญชีผู้ใช้

1) ดึงข้อมูลผู้ใช้จาก Firebase

เมื่อ sign-in เรียบร้อย user != null จะดึงข้อมูลผู้ใช้ออกมา ชื่อ, อีเมล

```
FirebaseUser user = FirebaseAuth.getInstance().getCurrentUser();  
if (user != null) {  
    String name = user.getDisplayName();  
    String email = user.getEmail();  
    Uri photoUrl = user.getPhotoUrl();  
    String uid = user.getUid();  
}
```

2) ดึงข้อมูล provider ของผู้ใช้

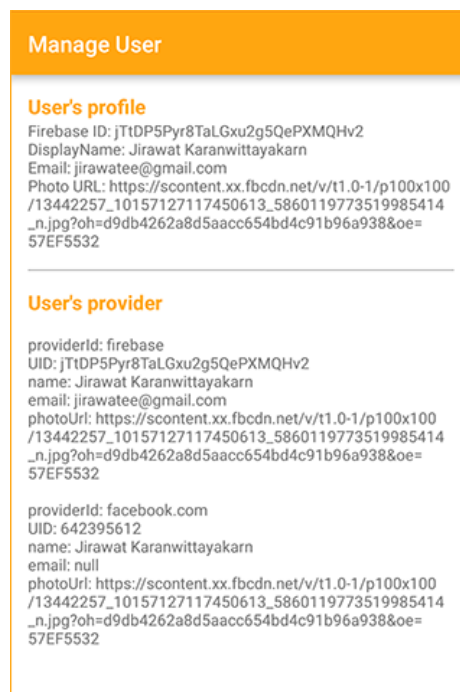
การ sign-in ของผู้ใช้จากบริการอื่นๆเช่น Google ต้องการดึง raw data ที่ได้จากผู้ให้บริการต่างๆ

```
FirebaseUser user = FirebaseAuth.getInstance().getCurrentUser();  
if (user != null) {  
    for (UserInfo profile : user.getProviderData()) {
```

```

// Id of the provider (ex: google.com)
String providerId = profile.getProviderId();
String uid = profile.getUid();
String name = profile.getDisplayName();
String email = profile.getEmail();
Uri photoUrl = profile.getPhotoUrl();
};
}
ตัวอย่าง

```



3) การอัปเดตข้อมูลผู้ใช้

อัปเดตข้อมูล DisplayName และ Photo URL ผู้ใช้ได้ด้วย updateProfile() สำหรับบัญชีผู้ใช้ด้วย email และ password

```

FirebaseUser user = FirebaseAuth.getInstance().getCurrentUser();
UserProfileChangeRequest profileUpdates = new
UserProfileChangeRequest.Builder()
    .setDisplayName("Jirawatee")
    .setPhotoUri(Uri.parse("http://placeholder.it/96x96"))

```

```

        .build();
    user.updateProfile(profileUpdates)
        .addOnCompleteListener(new OnCompleteListener<Void>() {
            @Override
            public void onComplete(Task<Void> task) {
                if (task.isSuccessful()) {
                    Log.d(TAG, "User profile updated.");
                }
            }
        });

```

The screenshot shows a mobile application interface titled "Manage User". It contains three main sections:

- User's profile:** Displays the message "User profile updated."
- User's provider:** Lists two providers:
 - firebase:** providerId: firebase, UID: EYrhCapfZ8NNwB5hAnHSJ01r6Zi2, name: Jirawatee, email: user@example.com, photoUrl: http://placeholder.it/96x96
 - password:** providerId: password, UID: user@example.com, name: null, email: null, photoUrl: null
- Update user's profile:** A section with a text input field containing "Jirawatee" and another text input field containing "http://placeholder.it/96x96". Below these fields is a yellow button labeled "UPDATE USER'S PROFILE".

4. แผนการปฏิบัติงานในสัปดาห์ต่อไป(plans)

- การเพิ่มข้อมูลการแสดงความคิดเห็นของผู้ปกครอง
- เช็กระบบการเข้าสู่ระบบว่ามีสถานะเป็นผู้ปกครอง ผู้เรียน และผู้สอน

5. บันทึกเพิ่มเติม(Open Issues)

ลงชื่อ อาจารย์ที่ปรึกษา
(อาจารย์สุระชัย พิมพ์สวัสดิ์)
วันที่.....