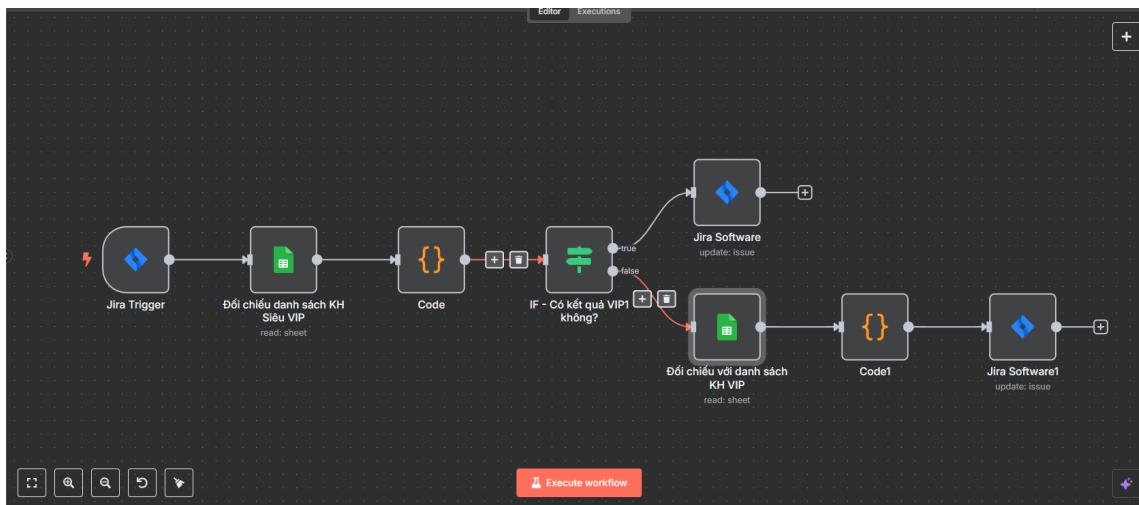


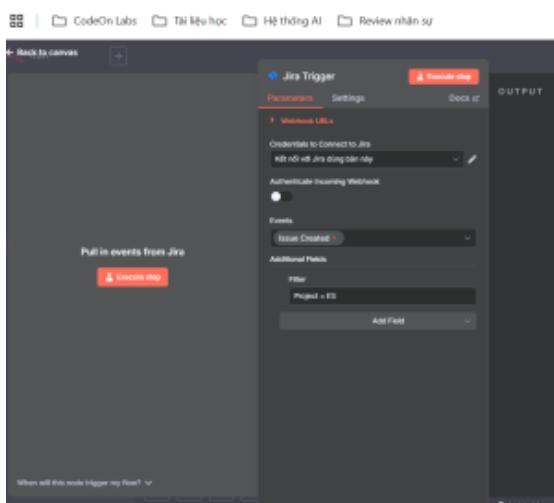
.1. Tự động phân loại khách hàng vV1.0



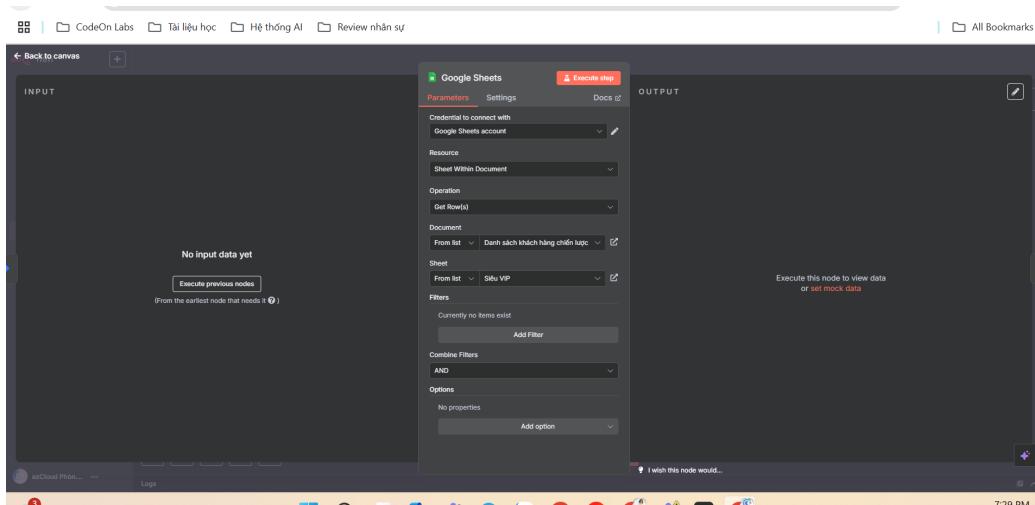
Node 1: Dùng Jira trigger để chạy mỗi khi có 1 issue mới được tạo

Event: Issue created

Project = ES



Node 2: Kết nối với file google sheet, trong sheet danh sách khách hàng Siêu Vip để đổi chiều xem khách hàng trong issue vừa tạo có thuộc nhóm Siêu Vip hay không



Node 3: Đoạn code dùng để nhận diện khách hàng Siêu VIP thông qua SĐT, email, domain, Summary, Organization, Description theo thứ tự ưu tiên như trên

```
// === Lấy dữ liệu từ Jira ===
const reporter = $('Jira Trigger').first().json.issue.fields.reporter;
const reporterEmail = reporter && typeof reporter === 'object' ? reporter.emailAddress : reporter;
const normalizedReporterEmail = String(reporterEmail || "").trim().toLowerCase();

const phoneFromJira = String($('Jira Trigger').first().json.issue.fields.customfield_12814 || "").trim();
const orgName = String($('Jira Trigger').first().json.issue.fields.customfield_10710 || "").trim().toLowerCase();
const summary = String($('Jira Trigger').first().json.issue.fields.summary || "").trim().toLowerCase();
const description = String($('Jira Trigger').first().json.issue.fields.description || "").trim().toLowerCase();
const issueKey = $('Jira Trigger').first().json.issue.key;

// === Hàm chuẩn hóa domain ===
const cleanDomain = (val) =>
  String(val || "")
    .replace(/@/, "") // Bỏ dấu @ nếu có
    .trim()
    .toLowerCase();

// === Lặp qua từng dòng từ Google Sheets ===
const rows = $input.all();

for (const row of rows) {
  const data = row.json;

  const email1 = String(data['Email 1'] || "").trim().toLowerCase();
  const email2 = String(data['Email 2'] || "").trim().toLowerCase();
  const domain1 = cleanDomain(data['Email Domain 1']);
  const domain2 = cleanDomain(data['Email Domain 2']);
  const phone1 = String(data['Phone Number 1'] || "").trim();
  const phone2 = String(data['Phone Number 2'] || "").trim();
  const hotelName = String(data['Hotel name'] || "").trim().toLowerCase();

  // 1. Email chính xác
  if (normalizedReporterEmail && (normalizedReporterEmail === email1 || normalizedReporterEmail === email2)) {
```

```

return {
  json: {
    result: 'VIP1',
    matchedBy: 'Email',
    issueKey
  }
};

// 2. Email domain
if (normalizedReporterEmail.includes('@')) {
  const domainFromJira = normalizedReporterEmail.split('@')[1];
  if (
    domainFromJira &&
    (domainFromJira === domain1 || domainFromJira === domain2)
  ) {
    return {
      json: {
        result: 'VIP1',
        matchedBy: 'Email Domain',
        issueKey
      }
    };
  }
}

// 3. Số điện thoại
if (phoneFromJira && (phoneFromJira === phone1 || phoneFromJira === phone2)) {
  return {
    json: {
      result: 'VIP1',
      matchedBy: 'Phone Number',
      issueKey
    }
  };
}

// 4. Hotel name khớp orgName, summary, hoặc description
if (
  hotelName &&
  (
    hotelName === orgName ||
    hotelName === summary ||
    hotelName === description
  )
) {
  return {
    json: {
      result: 'VIP1',
      matchedBy: 'Hotel name',
      issueKey
    }
  };
}

// Không khớp → không trả ra item nào
return [
  json: {
    result: 'Chuyển sang luồng tiếp theo',
    matchedBy: '',
  }
];

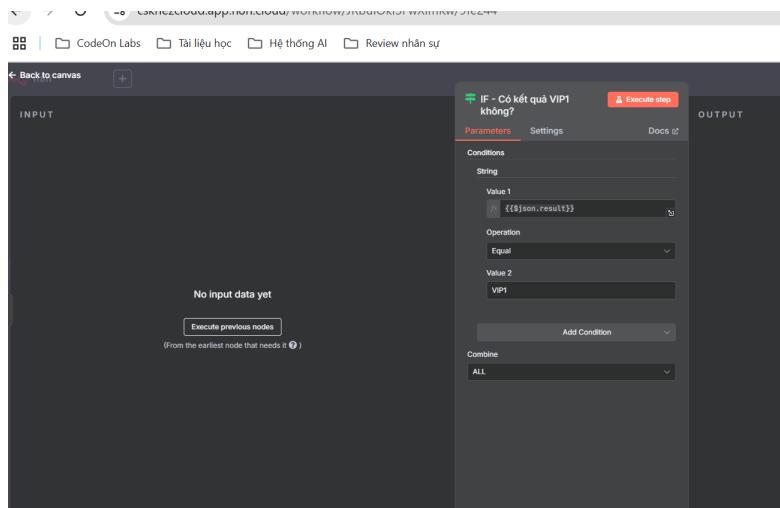
```

```
issueKey  
}  
];
```

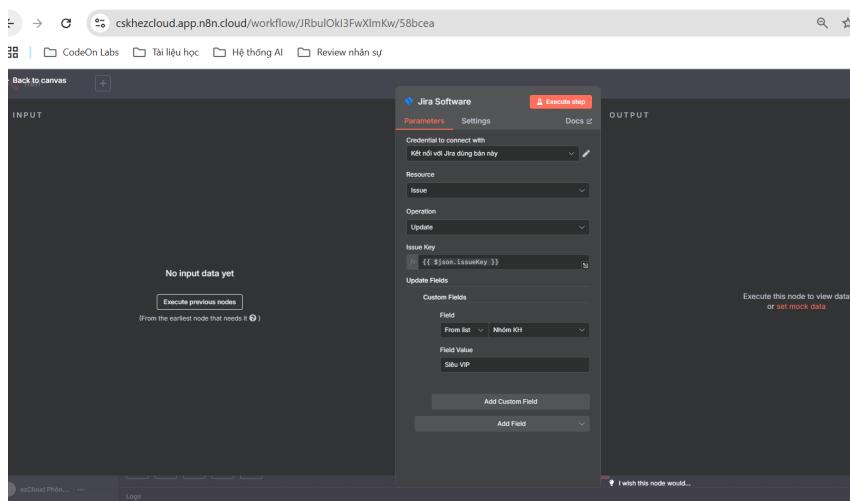
Node 4: Dùng node IF để trả ra kết quả:

Nếu kết quả trả ra là VIP 1 => Update lại trường Nhóm khách hàng trên Jira thành Siêu VIP

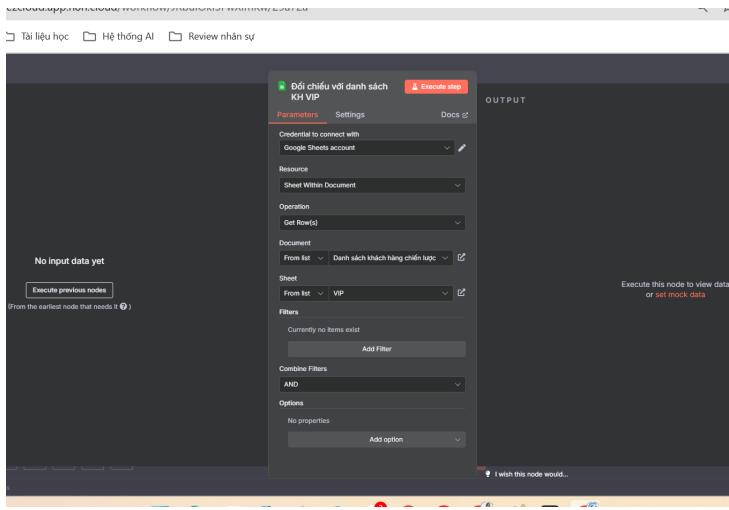
Nếu kết quả không trả ra VIP 1: chạy luồng thứ 2



Node 5: Update Nhóm KH thành Siêu VIP nếu kết quả trả ra là VIP 1



Node 6: Nếu kết quả không trả về VIP 1, đổi chiểu tiếp vớ file google sheet ở sheet KH VIP để tìm VIP 2



Node 7: Đoạn code dùng để nhận diện khách hàng VIP thông qua SĐT, email, domain, Summary, Organization, Description theo thứ tự ưu tiên như trên

```
// === Lấy dữ liệu từ Jira ===
const reporter = $('Jira Trigger').first().json.issue.fields.reporter;
const reporterEmail = reporter && typeof reporter === 'object' ? reporter.emailAddress : reporter;
const normalizedReporterEmail = String(reporterEmail || "").trim().toLowerCase();

const phoneFromJira = String($('Jira Trigger').first().json.issue.fields.customfield_12814 || "").trim();
const orgName = String($('Jira Trigger').first().json.issue.fields.customfield_10710 || "").trim().toLowerCase();
const summary = String($('Jira Trigger').first().json.issue.fields.summary || "").trim().toLowerCase();
const description = String($('Jira Trigger').first().json.issue.fields.description || "").trim().toLowerCase();
const issueKey = $('Jira Trigger').first().json.issue.key;

// === Hàm chuẩn hóa domain ===
const cleanDomain = (val) =>
  String(val || "")
    .replace(/@/, '') // Bỏ dấu @ nếu có
    .trim()
    .toLowerCase();

// === Lặp qua từng dòng từ Google Sheets ===
const rows = $input.all();

for (const row of rows) {
  const data = row.json;

  const email1 = String(data['Email 1'] || "").trim().toLowerCase();
  const email2 = String(data['Email 2'] || "").trim().toLowerCase();
  const domain1 = cleanDomain(data['Email Domain 1']);
  const domain2 = cleanDomain(data['Email Domain 2']);
  const phone1 = String(data['Phone Number 1'] || "").trim();
  const phone2 = String(data['Phone Number 2'] || "").trim();
  const hotelName = String(data['Hotel name'] || "").trim().toLowerCase();

  // 1. Email chính xác
  if (normalizedReporterEmail && (normalizedReporterEmail === email1 || normalizedReporterEmail === email2)) {
    return {
      json: {
        // Logic xử lý khi tìm thấy khách hàng VIP
      }
    }
  }
}
```

```

        result: 'VIP2',
        matchedBy: 'Email',
        issueKey
    }
};

}

// 2. Email domain
if (normalizedReporterEmail.includes('@')) {
    const domainFromJira = normalizedReporterEmail.split('@')[1];
    if (
        domainFromJira &&
        (domainFromJira === domain1 || domainFromJira === domain2)
    ){
        return {
            json: {
                result: 'VIP2',
                matchedBy: 'Email Domain',
                issueKey
            }
        };
    }
}

// 3. Số điện thoại
if (phoneFromJira && (phoneFromJira === phone1 || phoneFromJira === phone2)) {
    return {
        json: {
            result: 'VIP2',
            matchedBy: 'Phone Number',
            issueKey
        }
    };
}

// 4. Hotel name khớp orgName, summary, hoặc description
if (
    hotelName &&
    (
        hotelName === orgName ||
        hotelName === summary ||
        hotelName === description
    )
){
    return {
        json: {
            result: 'VIP2',
            matchedBy: 'Hotel name',
            issueKey
        }
    };
}

// Không khớp → không trả ra item nào
return [];

```

Node 8: Update Nhóm KH thành VIP nếu kết quả trả ra là VIP 2

