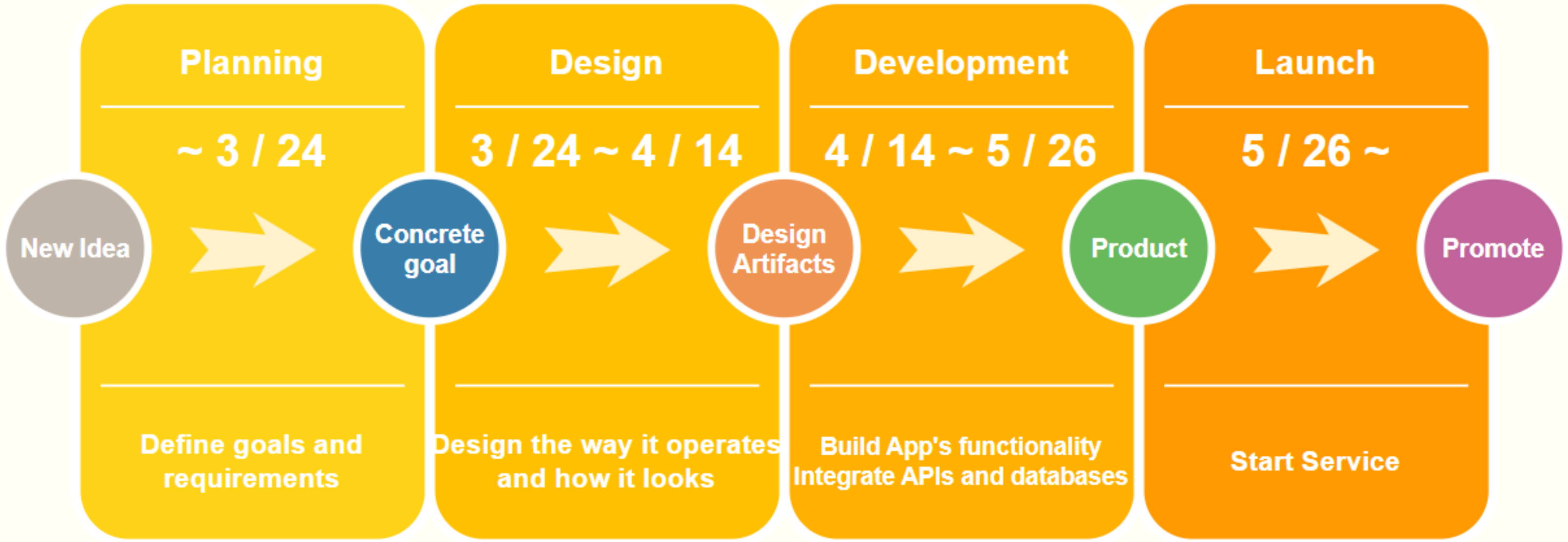
Midterm Presentation

Team C

Status of Progress



New Idea

Planning

~ 3 / 24

Define goals and requirements

Concrete goal

Design

3 / 24 ~ 4 / 14

Design the way it operates and how it looks

Design Artifacts

Development

4 / 14 ~ 5 / 26

Build App's functionality
Integrate APIs and databases

Product

Launch

5 / 26 ~

Start Service

Promote

Status of Progress (Continued)

Ideation, planning, shaping service, and making proposal (~ Mar 24)

UI design, Software architecture development. (~ Mar 31)

Collect data (validate crawling engine), make mobile app and server (~ May 5)

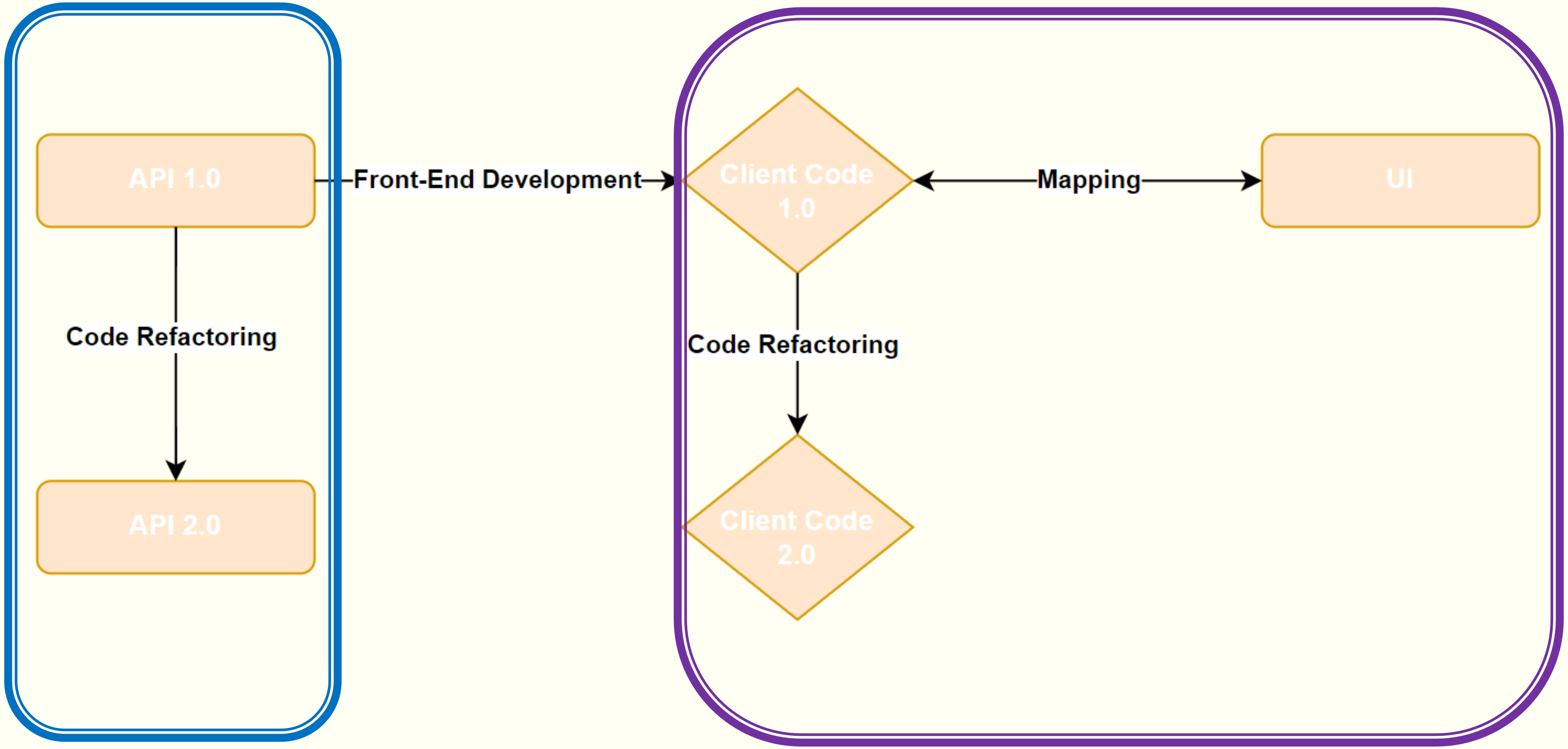
Code integration, acceptance test (~ May 19)

Final sign-off, deploy and release (~ June 9)

Collaborative Work

Back-End	강동헌	Database System Setting Crawling
Back-End	이수철	Log-In and Notification Encryption
Back-End	김태원	Database System Setting Mailing Authentification
Front-End	서창연	Application Development
Front-End	조예성	Application Development

Collaborative Work (Continued)



Collaborative Work (Continued)

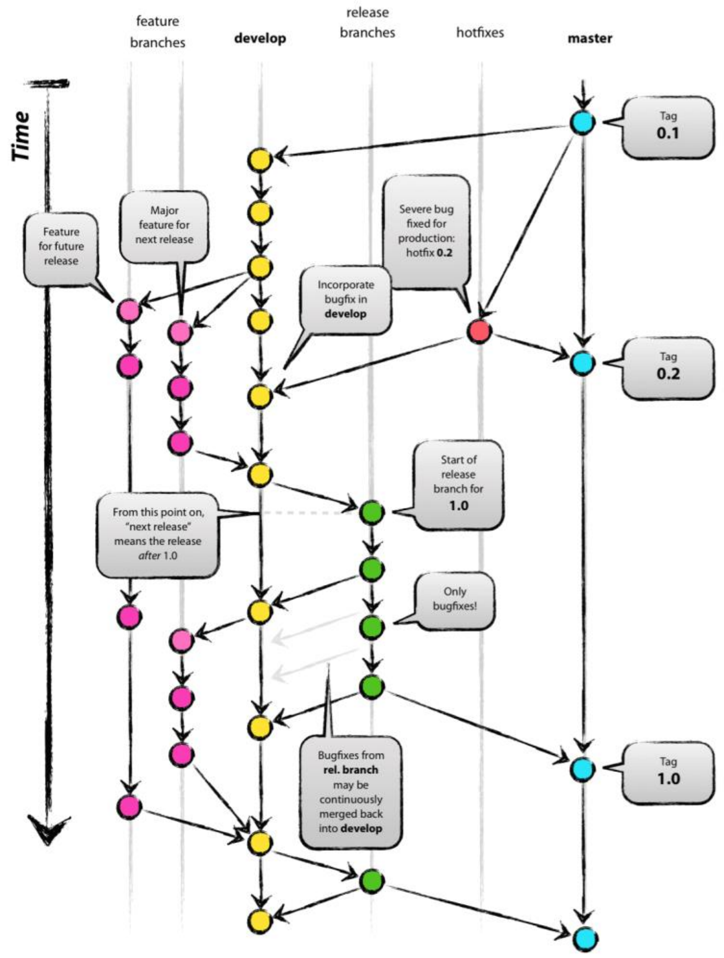

feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

Figure 1 Git Flow [1]

1. Collaborative Development

2. Consistent Workflow

3. Release Management

API Specification

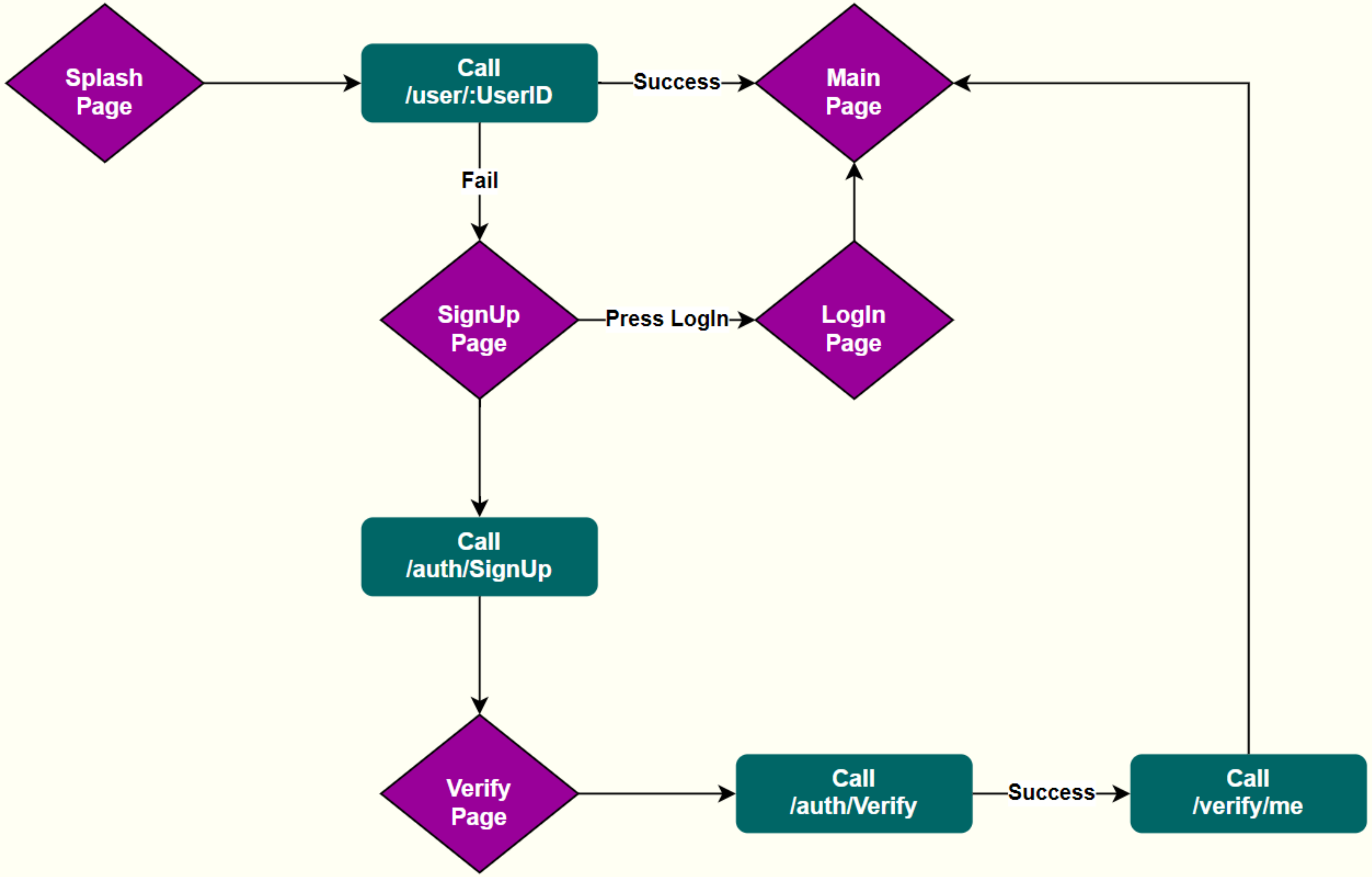
1. Authentication

/auth/signup	Add new users Send Post Request -> Get Success or Failure Message
/auth/signin	Login request Send POST Request -> Get JWT Token
/auth/verify/:confirmationCode	Request Email verification code authentication Send Get Request -> Get JWT

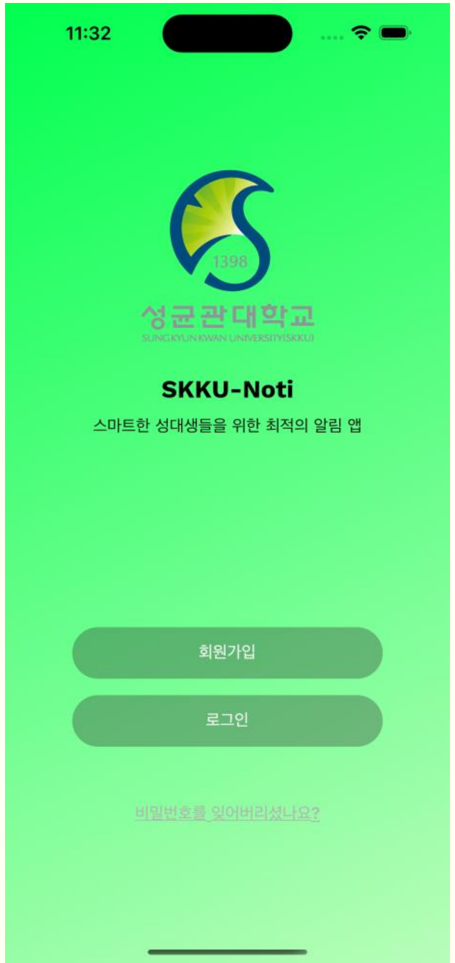
2. User

/user/me	Check existing user information Send GET Request -> Get tokens

Page Transition Flowchart



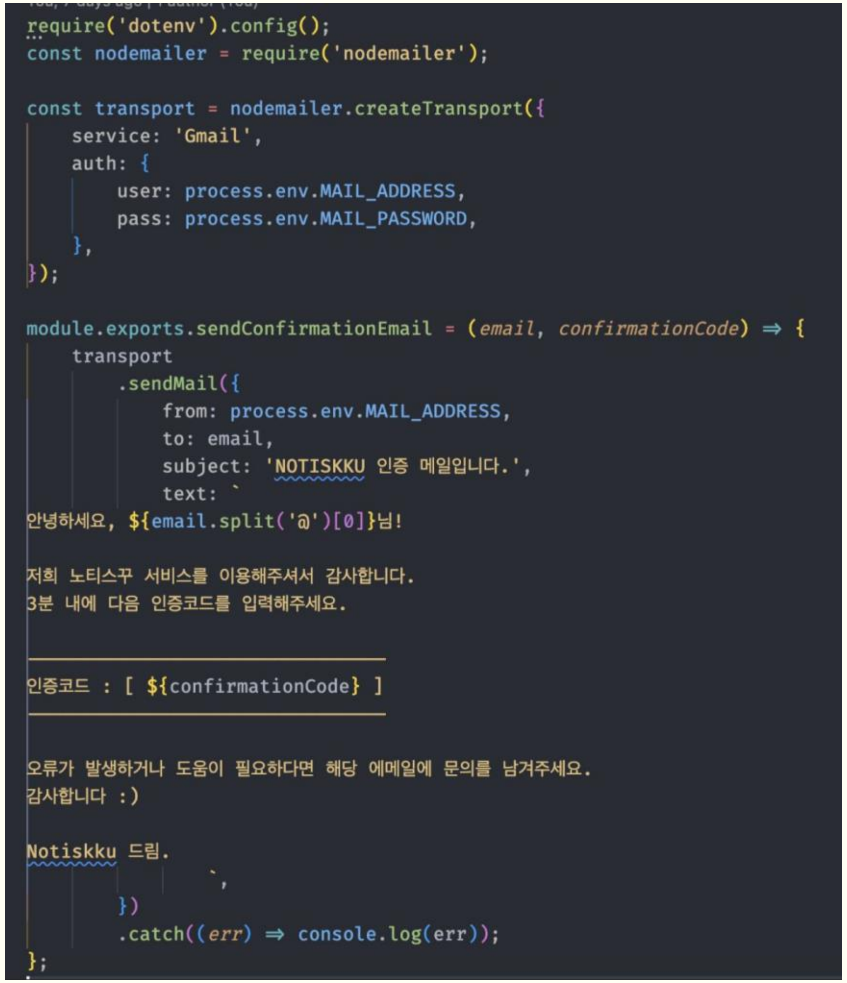
LogIn Page



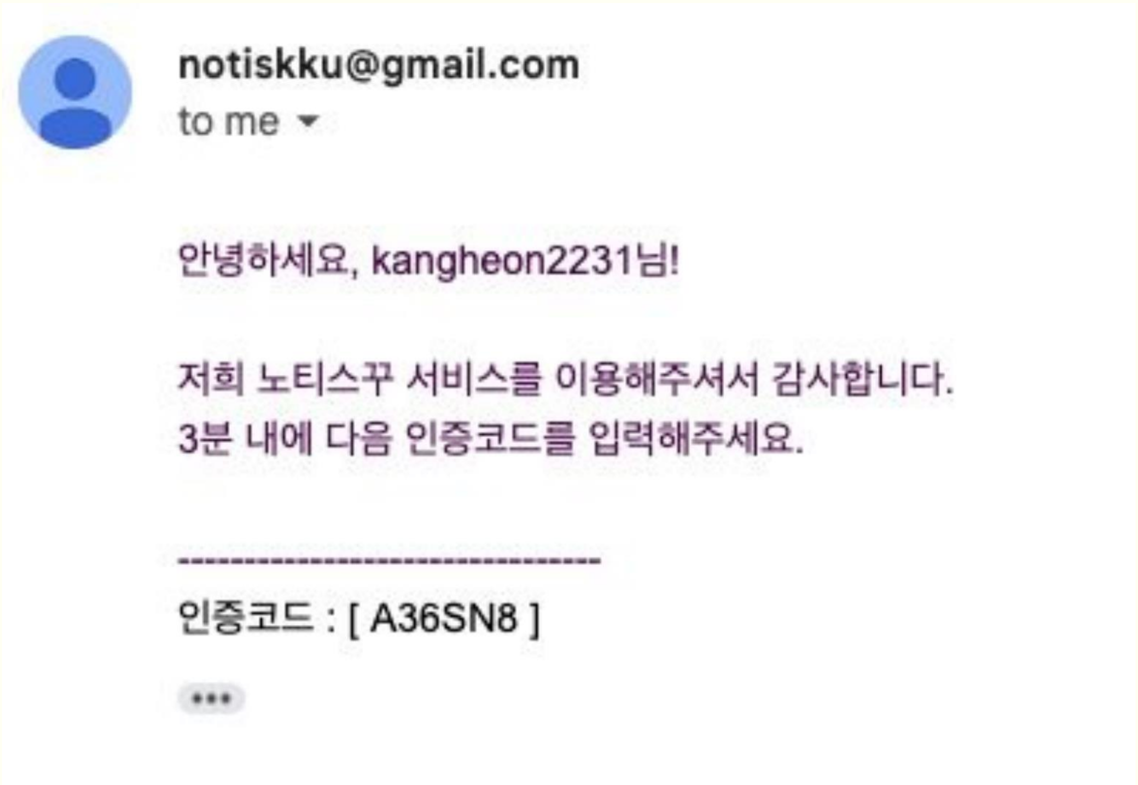
SKKU-Noti

스마트한 성대생들을 위한 최적의 알림 앱

API Specification (Continued)



```javascript
require('dotenv').config();
const nodemailer = require('nodemailer');

const transport = nodemailer.createTransport({
    service: 'Gmail',
    auth: {
        user: process.env.MAIL_ADDRESS,
        pass: process.env.MAIL_PASSWORD,
    },
});

module.exports.sendConfirmationEmail = (email, confirmationCode) => {
    transport
        .sendMail({
            from: process.env.MAIL_ADDRESS,
            to: email,
            subject: 'NOTISKKU 인증 메일입니다.',
            text: `
안녕하세요, ${email.split('@')[0]}님!

저희 노티스꾸 서비스를 이용해주셔서 감사합니다.
3분 내에 다음 인증코드를 입력해주세요.

────────────────────────────────

인증코드 : [ ${confirmationCode} ]
────────────────────────────────


오류가 발생하거나 도움이 필요하다면 해당 에메일에 문의를 남겨주세요.
감사합니다 :)

Notiskku 드림.
                `,
        })
        .catch((err) => console.log(err));
};
```



notiskku@gmail.com
to me ▾

안녕하세요, kangheon2231님!

저희 노티스꾸 서비스를 이용해주셔서 감사합니다.
3분 내에 다음 인증코드를 입력해주세요.

인증코드 : [A36SN8]

•••

CI / CD Pipeline

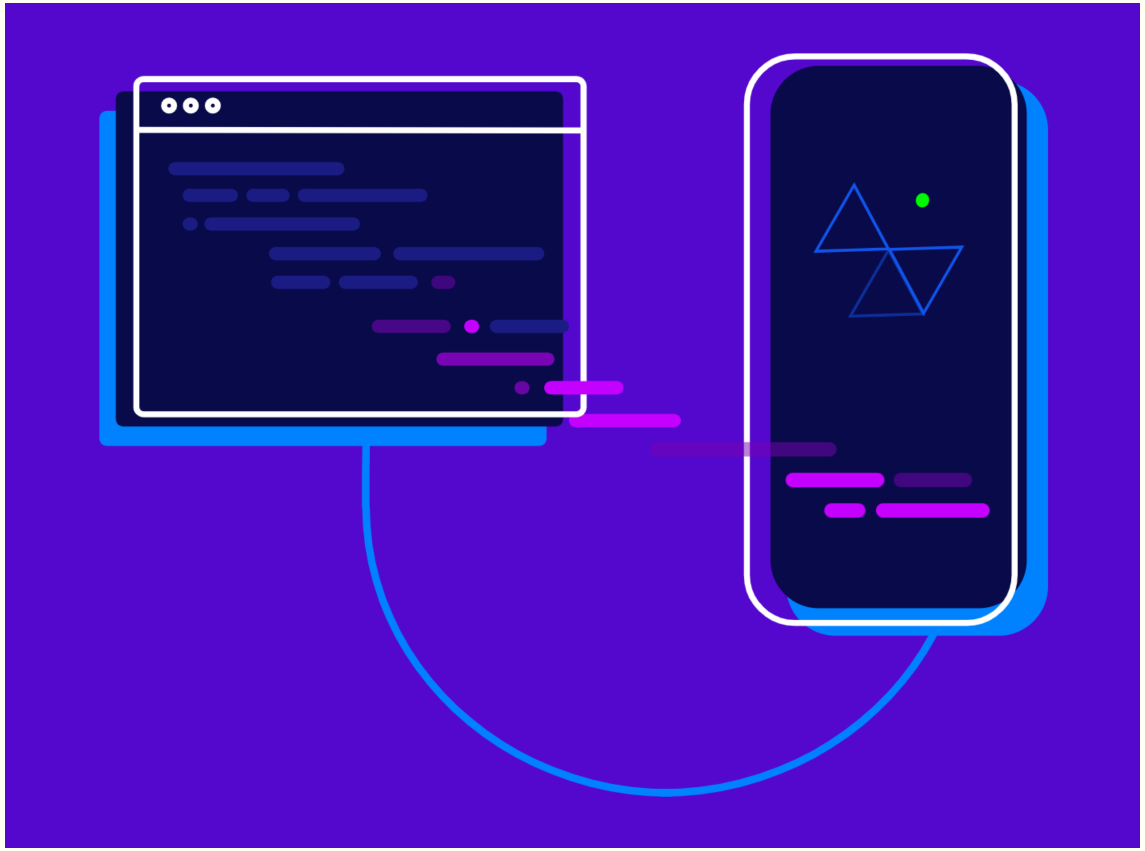
1. Different App Environment between tester and user

   for preventing reverse-engineering

2. Decryption of Log files for modifying distribution version

3. Different access address(Prod and Stag) for server-based testing

Fastlane : Automate Development Process



Automated App Configuration for diverse environments [3]

Current Progress in Back-End

1. Have built LogIn API

   Completed

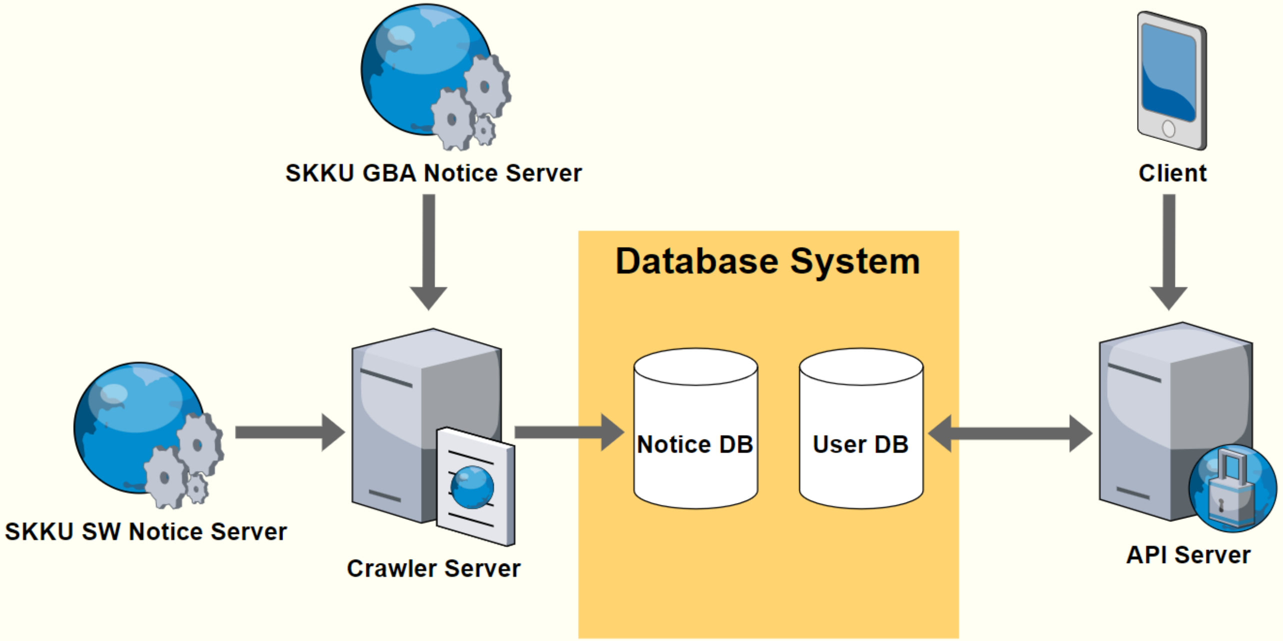
2. Running Crawler to scrape websites for Data

   Currently Crawling Data from GBA under permission

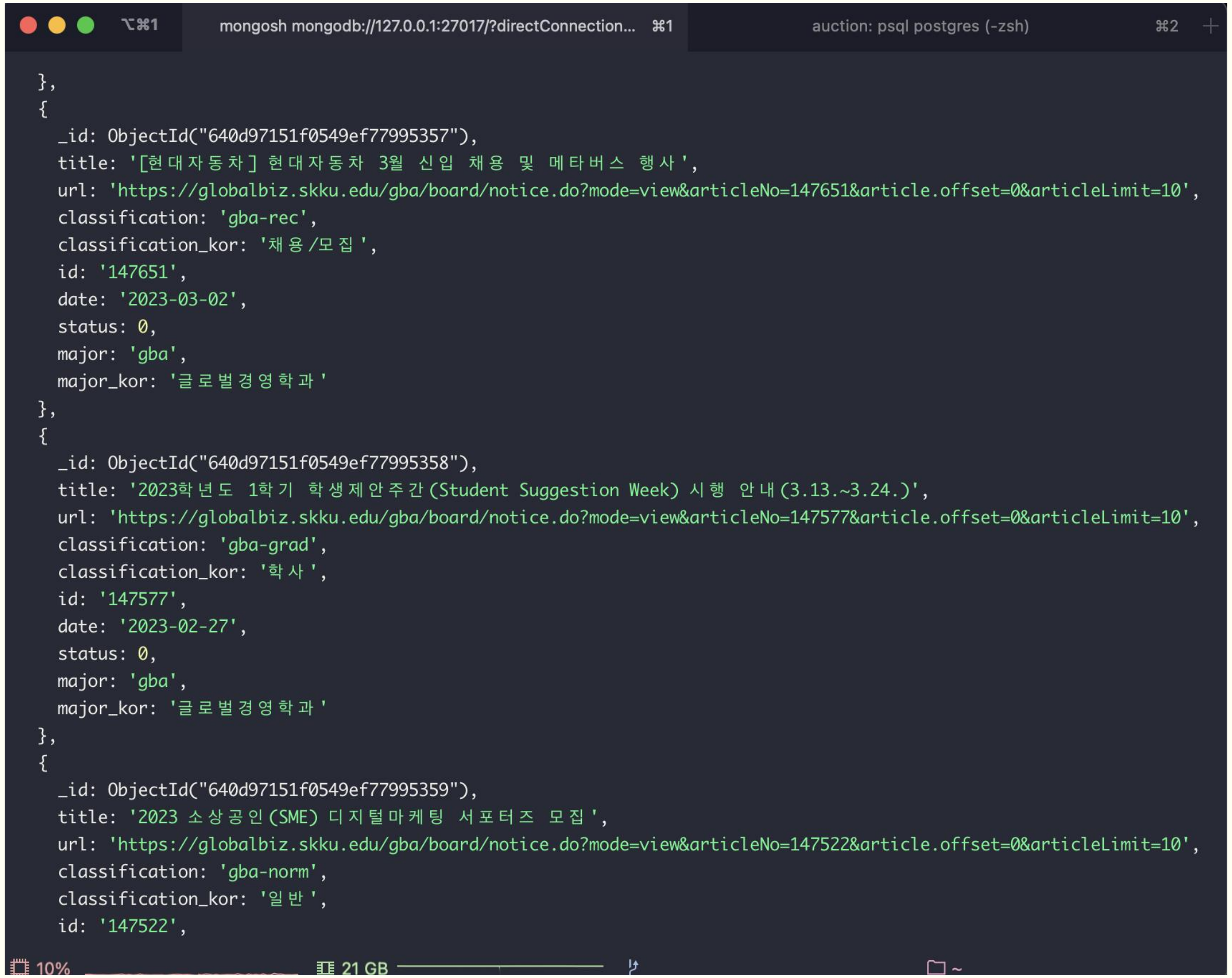
3. Have built pipeline to connect crawler to MongoDB

   Completed

SKKU-NOTI System

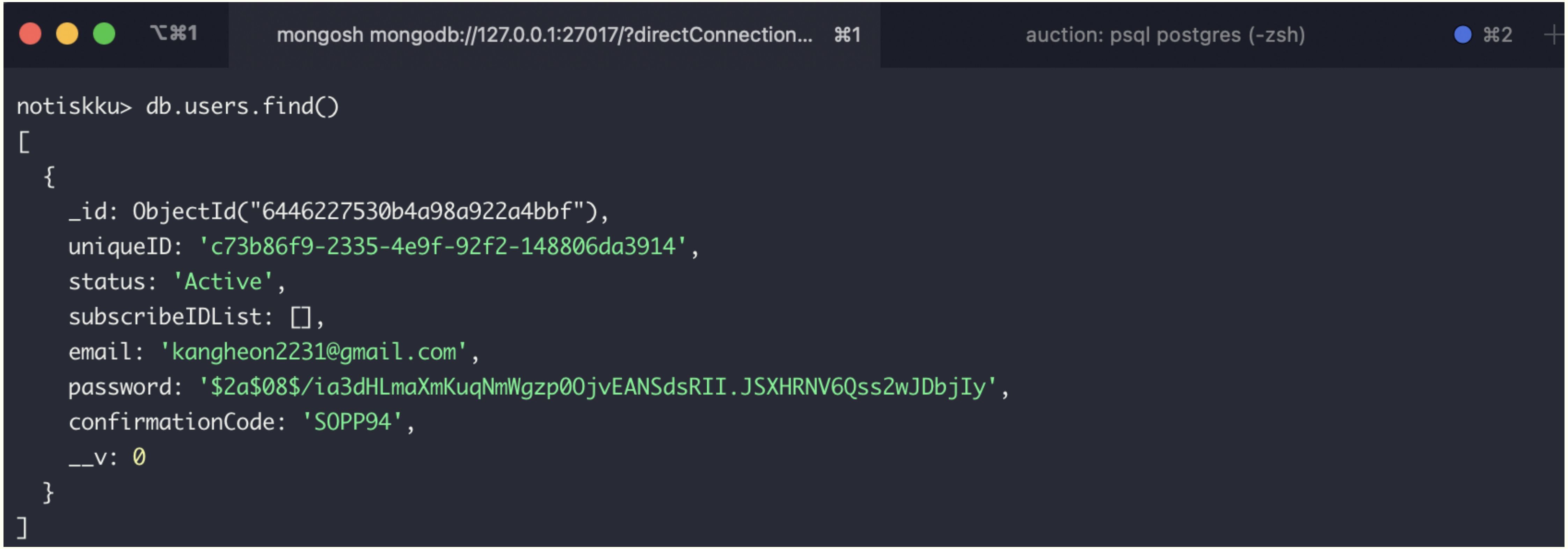


Notice DB


⌥⌘1
mongosh mongodb://127.0.0.1:27017/?directConnection... ⌘1
auction: psql postgres (-zsh) ⌘2
10%
21 GB
~

```
  },
  {
    _id: ObjectId("640d97151f0549ef77995357"),
    title: '[현대자동차] 현대자동차 3월 신입 채용 및 메타버스 행사',
    url: 'https://globalbiz.skku.edu/gba/board/notice.do?mode=view&articleNo=147651&article.offset=0&articleLimit=10',
    classification: 'gba-rec',
    classification_kor: '채용/모집',
    id: '147651',
    date: '2023-03-02',
    status: 0,
    major: 'gba',
    major_kor: '글로벌경영학과'
  },
  {
    _id: ObjectId("640d97151f0549ef77995358"),
    title: '2023학년도 1학기 학생제안주간(Student Suggestion Week) 시행 안내(3.13.~3.24.)',
    url: 'https://globalbiz.skku.edu/gba/board/notice.do?mode=view&articleNo=147577&article.offset=0&articleLimit=10',
    classification: 'gba-grad',
    classification_kor: '학사',
    id: '147577',
    date: '2023-02-27',
    status: 0,
    major: 'gba',
    major_kor: '글로벌경영학과'
  },
  {
    _id: ObjectId("640d97151f0549ef77995359"),
    title: '2023 소상공인(SME) 디지털마케팅 서포터즈 모집',
    url: 'https://globalbiz.skku.edu/gba/board/notice.do?mode=view&articleNo=147522&article.offset=0&articleLimit=10',
    classification: 'gba-norm',
    classification_kor: '일반',
    id: '147522',
```

User DB



⌥⌘1
mongosh mongodb://127.0.0.1:27017/?directConnection... ⌘1
auction: psql postgres (-zsh)
⌘2

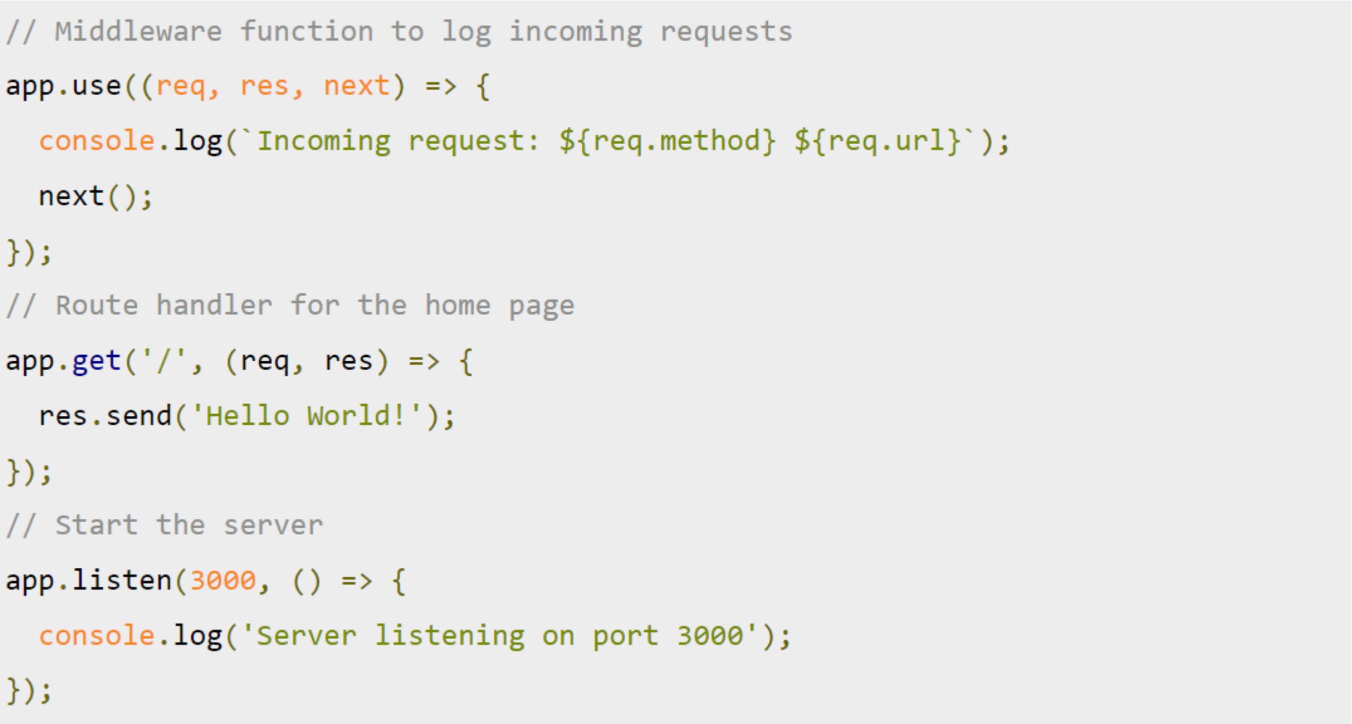
```
notiskku> db.users.find()
[
  {
    _id: ObjectId("6446227530b4a98a922a4bbf"),
    uniqueID: 'c73b86f9-2335-4e9f-92f2-148806da3914',
    status: 'Active',
    subscribeIDList: [],
    email: 'kangheon2231@gmail.com',
    password: '$2a$08$/ia3dHLmaXmKuqNmWgzp0OjvEANSdsRII.JSXHRNV6Qss2wJDbjIy',
    confirmationCode: 'SOPP94',
    __v: 0
  }
]
```

Middleware Design Pattern

1. Multiple design patterns available

2. Various Trial & error among Express designs

3. Chosen for easy maintenance and fast addition of functionalities

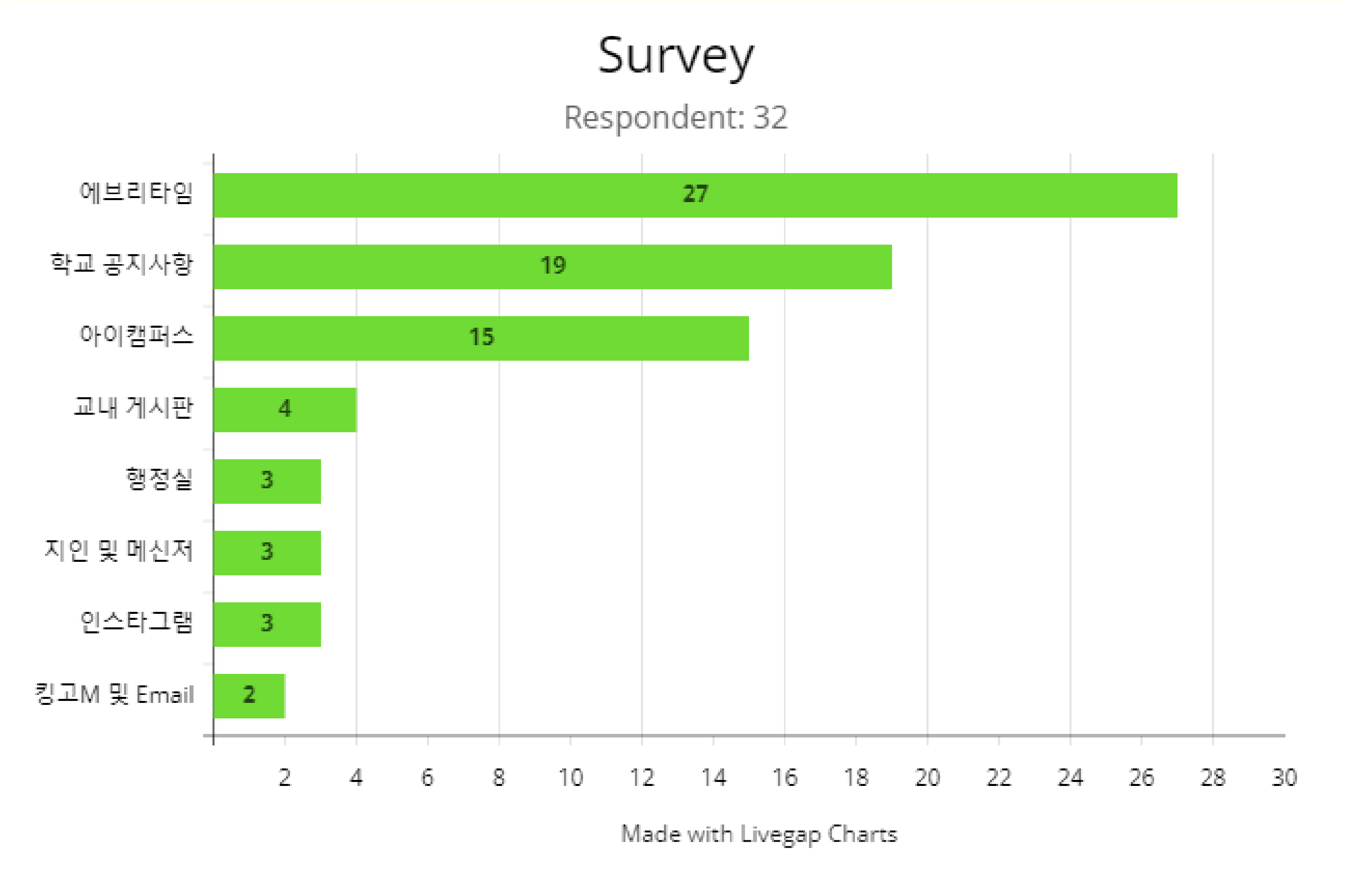


```
// Middleware function to log incoming requests
app.use((req, res, next) => {
  console.log(`Incoming request: ${req.method} ${req.url}`);
  next();
});
// Route handler for the home page
app.get('/', (req, res) => {
  res.send('Hello World!');
});
// Start the server
app.listen(3000, () => {
  console.log('Server listening on port 3000');
});
```

Figure 2 Middleware Functions [2]

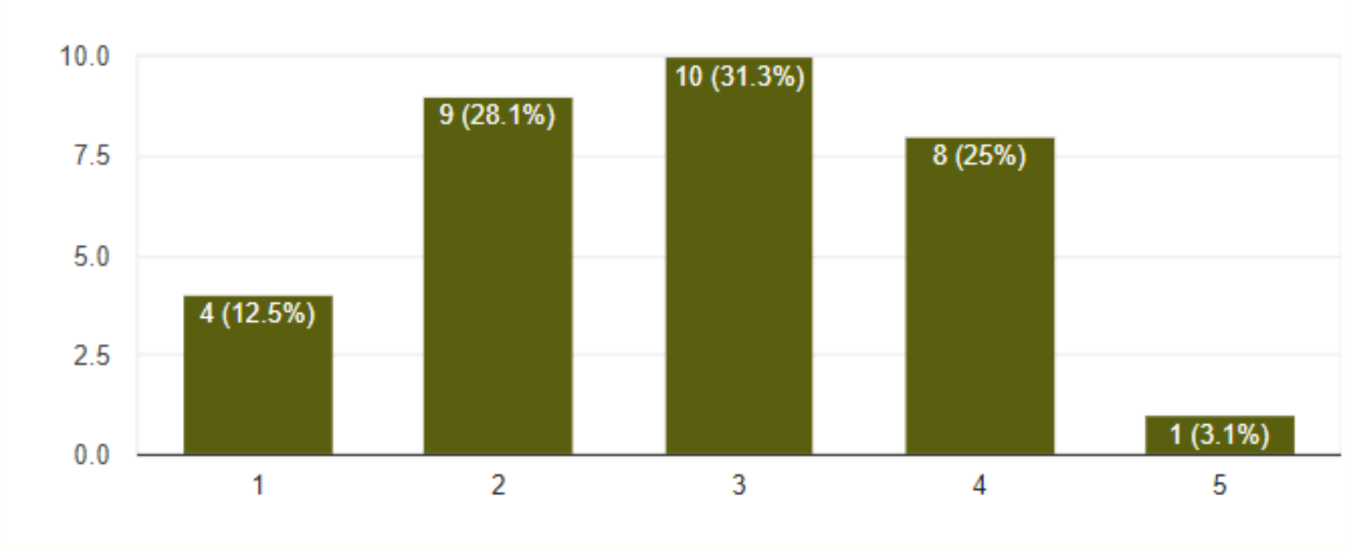
Project Milestone Review

From where do you usually get the notice information?



Project Milestone Review (Continued)

Then, are you satisfied with the current school notification system?



Project Milestone Review (Continued)

1. 학생들이 직접 알아보아야 하는 시스템인데 공지사항들이 정확히 어디에 나와있는 지 불명확하기 때문에 꽤 오랜 시간을 정보를 찾기 위해 할애해야 함

2. 학교 홈페이지, 각 해당 과 홈페이지, 학술정보관 홈페이지, 등등 너무 여러 군데에 올라와서 통합 포털이 있으면 좋겠다.

3. 신청기한이 지나고 알게 된 공지사항을 보게 된 경우가 꽤 있음

↓

Have these complaints been solved by our Application?

How to Deal with Challenges

Too many irrelevant announcements or Missing application deadline

↓

Notification System for only subscribed categories

No integrated portal for announcements with many websites with different announcements

↓

Crawl all information from several school notice systems

Expectation of Application

1. Save time and effort in checking multiple websites

2. Reduce the number of irrelevant announcements

3. Provide timely and relevant information to users

Reference

[1] 나동호, “우린 Git-flow를 사용하고 있어요”, https://techblog.woowahan.com/2553/

[2] LogRocket, “A guide to Node.js design patterns”, https://blog.logrocket.com/guide-node-js-design-patterns/#factory-method-pattern:~:text=in%20more%20detail.-,Middleware,-In%20Node.js

[3] Fastlane, “App automation done right”, https://fastlane.tools/