**TEACHER**

**Fields/Columns:** teacher\_id(int), nickname(char)

/api/teacher/

GET:

Parameters: none

Response: array of all teachers (all fields)

POST:

Parameters: Teacher.nickname

Action: Add new teacher with that nickname

Response: New Teacher object

/api/teacher/<teacher\_id>

GET:

Parameters: none

Response: teacher object matching <teacher\_id> (all fields)

POST:

Parameters: All teacher fields acceptable (optional)

Action: Edit teacher object matching teacher\_id, return teacher object.

Response: Edited Teacher object

**STUDENT**

**Fields/Columns:** student\_id(int), student\_name(char), preferred\_teacher1(foreignkey int), preferred\_teacher2(foreignkey int), preferred\_teacher3(foreignkey int), rejected\_teacher1(foreignkey int), rejected\_teacher2(foreignkey int), rejected\_teacher3(foreignkey int)

/api/student/

GET:

Parameters: none

Response: array of all students (all fields)

POST:

Parameters: Student.nickname

Action: Add new student with that nickname

Response: New Student object

/api/student/<student\_id>

GET:

Parameters: none

Response: student object matching <student\_id> (all fields)

POST:

Parameters: All student fields acceptable (optional)

Action: Edit student object matching student\_id, return student object.

Response: Edited Student object

**EVENT**

**Fields/Columns:** event\_id(int), start\_datetime(datetime), end\_datetime(datetime), duration(float), teacher(int foreignkey), student(int foreignkey), notes(char max\_length=200)

/api/event/

GET:

Parameters: None

Response: Return list of all events

POST:

Parameters: All event fields acceptable (optional), teacher\_id and student\_id are mandatory.

Action: Add new event with teacher and student, adds event to google calendar(primary)

Response: New event

/api/event/<event\_id>

GET:

Parameters: None

Response: Return single event matching event\_id

POST:

Parameters: All event fields acceptable(optional)

Action: Edit single event with given request fields

Response: Single edited event

**AVAILABILITY**

**Fields/Columns:** monday\_start(time), monday\_end(time), monday\_duration(float), tuesday\_start(time), tuesday\_end(time), tuesday\_duration(float), … , sunday\_start(time), sunday\_end(time), sunday\_duration(float), teacher(int foreignkey)

/api/availability/

GET:

Parameters: None

Response: Return list of all availability rows

POST:

Parameters: All availability fields acceptable (optional)

Action: Add new availability row

Response: New availability

/api/availability/<pk>

GET:

Parameters: None

Response: Return single availability row matching pk

POST:

Parameters: All availability fields acceptable(optional)

Action: Edit single availability with given request fields

Response: Single edited availability

/api/availability\_by\_teacher/<teacher\_id>

GET:

Parameters: None

Response: Return single availability row matching teacher\_id

POST:

Parameters: All availability fields acceptable(optional)

Action: Edit single availability with given request fields

Response: Single edited availability

**SKILLS**

**Fields/Columns:** skill\_1(bool), skill\_2(bool), …, skill\_15(bool)

Skills Key:

1. ACT-English, 2. ACT-Math, 3. ACT-Reading, 4. ACT-Science, 5. ACT-Writing, 6. Floor-ELA, 7. Floor-Math, 8. Pre-Calc, 9. Calculus, 10. Biology, 11. Chemistry, 12. Physics, 13. Earth/Space 14. History, 15. Spanish

/api/skill/

GET:

Parameters: None

Response: Return list of all skill rows

POST:

Parameters: All skills fields acceptable (optional)

Action: Add new skills row

Response: New skills

/api/availability/<pk>

GET:

Parameters: None

Response: Return single skills row matching pk

POST:

Parameters: All skills fields acceptable (optional)

Action: Edit single skills row with given request fields

Response: Single edited skills row

**GOOGLE CALENDAR EVENTS**

/api/gevent/

GET:

Parameters: None

Response: Return list of all google events from primary calendar

/api/gevent/<eventId>\* (eventId != event\_id, eventId is the gCal event’s ID)

GET:

Parameters: None

Response: Return single google event matching eventId

**Misc Notes**

SOCIAL\_AUTH\_GOOGLE\_OAUTH2\_KEY = '323423619559-orlpuuiaalb7sp3ooblt4mjmp32ffq1t.apps.googleusercontent.com'

SOCIAL\_AUTH\_GOOGLE\_OAUTH2\_SECRET = '49JMMebwB4NwwFPic3UZ9z3m'

SOCIAL\_AUTH\_GOOGLE\_SCOPE = [

'https://www.googleapis.com/auth/calendar',

'https://www.googleapis.com/auth/plus.login',

]

SOCIAL\_AUTH\_LOGIN\_REDIRECT\_URL = '/home/'

SOCIAL\_AUTH\_LOGIN\_URL = '/scheduler/login'

SECRET\_KEY = 's5#t%ojcd\*f6^^7%#fjh4\_-k2yv+s(bl(a47+a!5bh1ht\_bg4p'

Test User: [ZenonsTest@gmail.com](mailto:ZenonsTest@gmail.com)

Password: password@2

Example Datetime

"start\_datetime": "2016-05-12T06:04:57.735000Z"

Heroku Key

56c41dac7b3519214894221e