|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | **Purpose** | **Type** | **Input** | **Returns** |
| /courses | Get all courses in DB | POST | N/A | String that will be parsed |
| /sections | Get all sections for all courses in DB | POST | N/A | String that will be parsed |
| /sectionsforcourse \*\*\*\* | Get all section in DB for the specified course | POST | courseID |  |
| /userprefs | Get preferences for a user | POST | username | {"success":"true","username":"SprinkKing","courseload":"4","dayoff":"Monday","preferredTime":"Afternoons"} |
| /needed | Get needed courses for a user | POST | username | String that will be parsed  {"List":[{"name":"oop", "number":"248"},{"name":"oop2", "number":"249"}]} |
| /taken | Get taken courses for a user | POST | username | String that will be parsed  {"List":[{"name":"oop", "number":"248"},{"name":"oop2", "number":"249"}]} |
| /scheduler | Generate schedule | POST | username |  |
| /addcourse | Add a course to the DB | POST | {name:’’, number:’’} | False: if course with that id already exists  True: otherwise |
| /addsection | Add a section to the DB | POST |  | False: if section with that id already exists  True: otherwise |
| /removecourse | Remove a course from the DB | POST | courseID | N/A |
| /removesection | Remove a section from the DB | POST | {course:courseID, section:sectionID} | N/A |
| /editcourse | Edit course information | POST |  | N/A |
| /editsection | Edit section information | POST |  | N/A |
| /editpreferences | Edit user’s preferences | POST | {username:’’,cload:'',dayoff:'',preftime:''}} | {"success":"true","username":"JasonB","courseload":"5","dayoff":"Monday","preftime":"Mornings"} |
| /editneededcourses | Edit user’s needed courses | POST | {username:’’, json: '{"List":[{"name":"oop", "number":"248"},{"name":"oop2", "number":"249"}]}'} | N/A |
| /edittakencourses | Edit user’s taken courses | POST | {username:’’, json: '{"List":[{"name":"oop", "number":"248"},{"name":"oop2", "number":"249"}]}'} | N/A |
| /editusername | Change username for user | POST | {old:’’, new:’’} | False: new username exists already  True: otherwise  Now : result"=>"true","username"=>"$new") |
| /editemail | Change e-mail for user | POST | {username:’’, new:’’} | ("success"=>"true","username"=>"$old","email"=>"$new")  False: userna |
| /editpassword | Change password for user | POST | {username:’’, password:’’} | "success"=>"true","password"=>"$new")); |
| /email | Get user’s email | POST | username | E-mail as a String  Now: "result":"good","username":"JasonB","email":"newEnail@email.com"} |
| /login | Login verification | POST | {username:’’, password:’’} | {success:bool, isAdmin:bool}  Now: {"success":"true","username":"JASONB","isAdmin":"true"} |
| /register | Register user | POST | {username:’’, email:’’, password:’’} | False: username exists already  True: otherwise  Now : {"success":"false","username":"SprinkKing","error":"usernametakenalready"} |

Note:

Here is the new structure of the database for the preferences.

Smallint //courseload

enum('None', 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday') //dayoff

enum('Any', 'Mornings', 'Afternoons', 'Evenings') //preferred time

since they are stored as an enum in the db the choices on the user interface should reflect that.