# **BED Assignment 01 Documentation**

## Team Overview

Class: **DISM/FT/2A/02**

Members:

1. Loh Kar Wei 1904204
2. Jess Kwek 1928934

## Requirements

* Mysql2
* Multer
* Express
* Body-parser
* bcrypt

## Setting Up

1. Install the required node modules specified in package.json file
   1. Npm install
2. Run setup.sql included in the file to setup and populate the database
   1. bed\_assignment.sql
3. Go into db.js to change the configurations for username and password for database

### Uploading Images

1. On postman, use form-data to send the post/put requests.
2. Using the key upload, set the file type to file and then choose the image that you wish to upload and upload!

### 

### Logging In

* Password automatically stored and hashed when creating new user
  + When creating new user using endpoint POST /users, have to include additional request body password for user password
    - Same when updating user information with put
* Endpoint used for logging in is POST /user/login
* Request body used is username and password
* If successful,
  + Status 200
  + Shows “Login successful, welcome user <username>
* If unsuccessful
  + Status 500
  + Internal server error

## Bonus features

* Use of asynchronous programming
* Writing image files locally to server storage
* Writing image binary into database and reconstructing them
* Login feature with password hashing and salting
  + Salt used differs for each user and stored together in database

## Database Tables

Schema Name: **sptravel**

Database Tables

|  |  |
| --- | --- |
| **Table Name** | **Columns** |
| users | * userid * primary key * auto increment * INT(11) * username * VARCHAR(255) * Unique key * email * VARCHAR(255) * profile\_pic\_url * VARCHAR(255) * password * VARCHAR(255) * salt * VARCHAR(255) * created\_at * timestamp |
| travel\_listings | * travel\_id * int(11) * auto increment * primary key * title * varchar(255) * description * text * price * int(11) * country * varchar(255) * travel\_period * varchar(255) * image\_url * varchar(255) |
| itinerary | * itinerary\_id * Int(11) * Auto increment * Primary key * day * int(11) * activity * varchar(255) * created\_at * timestamp * fk\_travel\_id * int(11) * foreign key * references travel\_id from travel\_listings * on delete cascade |
| reviews | * review\_id * int(11) * primary key * auto increment * fk\_travel\_id * int(11) * foreign key * references travel\_id from travel\_listings * on delete cascade * content * text * rating * int(11) * fk\_username * varchar(255) * foreign key * references username from users * on delete cascade * created\_at * timestamp |

## Contributions

|  |  |
| --- | --- |
| **Jess Kwek** | * Endpoint 1 to 6 * Writing and saving of image files * Asynchronous programming |
| **Loh Kar Wei** | * Endpoint 7 to 12 * Login feature with hashing and salting * Asynchronous programming |