# Baby, Build\*A\*Book Project Class Notes ((27 June 2018))

## Change to the Repo has been completed – ensure everything is updated

We have decided to change the Database set-up from a relational DB to a static DB. We will have two tables in the DB.

Themes: which will contain the themes and story page content (on theme per row in the table.

Books: which will contain rows of completed stories, essentially taking the variable inputs and the story content from the Themes table and building complete book per row.

We have agreed that we will use the simple page flip code and will have side by side pages. One with text (the story content) and one with the background image and avatar.

What we are working on between now and Saturday:

1. Building the input form section of the index.html (Risa)
2. Working on solving the avatar image overlay (Kimberly)
3. Style optimization (Kenzie)
4. Conversion of Index2.html to handlebar (Kenzie)
5. Review of current back-end code (Kimberly)
6. Begin looking into how to populate the BOOKS table (Ryan)
7. Add Finish or Submit or Generate Story Button at bottom of page (Kenzie)

Agreement for Saturday:

1. All person to perform commit, push and pull of the Repo
2. End to end walk through of the solution as it Saturday morning
3. Define the tasks for Saturday
4. Complete a Saturday mid-point check
5. Commit, Push, Pull and another review to task out work for Monday Class

Things we need to start thinking about:

* Writing the stories
* Populating the DB with stories
* Narrowing in on the API/s to use
* Integrating the APIs into the product
* Linking the input variables to the stories and into the BOOKs table
* Calling the Books table to append the story pages

Current Backlog Items:

1. Creating dropdown values which link to images for the Animal, so when the person inputs their favorite animal – the image of that animal would be pulled into the story
2. Match the Story Book Text to the favorite color selected from the options
3. Create a unique user login/password so stories can be saved saved and retrieved by a user