Project 2 Blueprint:

1. Node and Express Web Server
   1. Create database with user information – MySQL database + ORM
      1. Name, userID, password, balance, email, phone number
   2. GET and POST routing for retrieving and updating data
   3. Activities page – pay/request functions
2. Front End/UI
   1. Login page/create account
   2. forgot my password option
   3. validation – Back end
   4. activities (pay/request) screen
   5. handlebars
   6. routing
   7. Friends list – modal or screen for choosing
3. Project Management
   1. Folder structure planning and MVC Paradigm

New library, package, or technology

Nice to have:

1. show recent transactions per user

2. settings - edit profile/multiple themes

3. admin/user option

4. create groups with selectable admins among users

5. group pay

6. google locations api to generate advertisements

Suggested workflow for project 2

Broken down into three steps

Planning

Gap analysis

Execution and testing

Planning

Create a road map of what you would like to accomplish.

What APIs are going to be used if any?

How will the project be used (What problem will it solve)?

How will it be used?

User inputs (forms)?

User outputs (What information will be displayed to the user)?

Security (How will you maintain sensitive information private if any Optional)?

Error handling (How will you handle fringe cases)?

Performance (The speed at which it loads)

Scalability (Were this project to grow, how would you account for additional information)?

Data management (How will you handle persistent information in your DB)?

Anything else you think you would need to include.

Gap analysis

Create a gap analysis between what you know versus what you want to know and how you will get there.

Evaluate what you know what to do.

Evaluate what you know you don't know.

Evaluate what additional information you think you need to know.

Plan how to learn or gain better understanding over areas you don't understand.

Validate your plan

Evaluate the quality of your assumptions as you execute the project by documenting and testing your code iteratively.

How will you know each part of the project has been completed to an acceptable standard?

Justify each piece of code you write.

If your justification is wrong evaluate why to solidify your understanding.

Document as you move forward with the implementation of the plan.

Regularly test each piece of code you write as it is completed to verify it won't break your project.