**Project Outline**

**BY Nov 24:**

Frontend:

1. front page with login
2. Register page
3. Catalog

Backend:

Setup Sqlite DB

1. -maintain user information
   1. username(primary key)
   2. password
   3. address
   4. Purchase history(String)
2. -keep track of shoe stock

Front-end

* Front page
* login/registration View
  + When a user tries to checkout they get prompted to log in or to register if not already signed in
  + once a user logs in they stay logged in throughout the session unless they sign out
* Catalog of shoes View
  + Filter by brand, price
  + View stock of shoes
  + Images of shoes
* Purchase history view
* Shoe view
  + For when the user clicks on a shoe while browsing a dialog
* Customer account view
  + See user information (name, address)
  + Access purchase history view
  + Option to sign out from this account
* Admin view
  + Delete users
  + View all previous checkouts
  + View stock of shoes
  + Change stock of shoes
  + Change price of shoes
* Checkout view
  + Order completed view with all the checkout details
  + After checkout the new cc number, billing and shipping addresses become the default for that user
* We need a banner that displays all the different navigation buttons to get to the different views above

Manual Data Collection

* Each person should focus on a brand and collect pictures of 4 different shoes with 1-3 colourways each for a total of 10 pictures per person
* Just name them based on model and colour/colourway and we can figure out how to use them later
* Image can have whatever background
* Brand focus:
  + Steven - Nike
  + Matthew - Addidas
  + Neven - Puma
  + Eric - Under Armour

Backend

* DAO package:
  + Use interface ShoesDAO
  + Connect to database using class ShoesDAOImpl
* Database will be a sqlite to start with so we can work on it easier,
  + if multiple people need to access it at once then we can change it to a SQL server
* Database tables:
  + PURCHASES - contains all the purchases that have been done
    - **Purchase ID # (primary key)** rand
    - *Username (foreign key)*
    - items\_ids (string) contains all the ids purchased items
    - Credit card number
    - Billing/Shipping address
    - Purchase\_Date
  + USER - contains all the users’ data and their admin access
    - **Username (primary key)** string with regex matching to check for valid characters, must be at least 8 characters long
    - Password string with regex matching to check for valid characters must be at least 8 characters long
    - isAdmin boolean value indicating whether they have the ability to modify the database tables
    - Default Billing/Shipping address
    - Default credit card number
  + SHOETYPE - each row represents one pair of shoes with a model, colour, brand
    - **Shoe ID # (primary key)**
    - Model string
    - Colourway string
    - Brand string
    - Price float
    - ImageName (UNIQUE string) stores the file name for the image of the shoes
  + SHOESTOCK - holds the stock of each specific shoes in each size
    - *Shoe ID # (foreign key)*
    - **StockID (primary key)** formed by concatenating the shoe ID with the size number
    - Shoe size - will be represented by
    - stock/ quantity - int value cannot go below 0 , default value 0