

PROJECT OVERVIEW FOR DELIVERABLE 1:

- UNDERSTAND MEAN STACK
 - MONGODB, EXPRESS, ANGULAR, NODE
 - ↳ How DO THESE WORK TOGETHER?
- MONGODB: USES MONGOOSE, NPM DEPENDENCY
 - ↳ USERS & OBJECTS ARE MODELLED, EXPORTED, CREATED & DESTROYED BY MODEL SCHEMAS
 - ⇒ CRUD DESIGN FOR DATABASE
- ~~API~~ NODE & EXPRESS: ALL API CONTROL OF APP:
 - TELL MODELS WHAT TO DO, CONTROL FLOW, ROUTING, FUNCTIONS, ETC.
- ANGULAR: FRONT-END APP CONTROL, SEND & RECEIVE CLIENT REQUESTS
 - INTERACTIONS CALL TO THE SERVER AND UPDATE USER VIEWS

AUTHENTICATION MODELS: USING LOCAL STRATEGY WITH PASSPORT

CURRENT ARCHITECTURE OVERVIEW

BACKEND: MONGODB

→ MODELS HANDLED BY MONGOOSE

ASSIGNMENTS: { TITLE, DESCRIPTION, TEACHERS, SECTION, SUBMISSIONS }

↳ FUNCTIONS: { UPDATETITLE(TITLE, CB), UPDATERDESCRIPTION(DESCRIPTION, CB) }

DOCUMENTS: { TITLE, GRADE, STATUS, SUBMITTED TO, STUDENT, GRAPH }

↳ FUNCTIONS: { UPDATEGRAPH(DATA, CB), UPDATESTATUS(STATUS, CB),
UPDATESUBMITTEDTO(ASSIGNMENT ID, CB), UPDATETITLE(TITLE, CB), UPDATEGRADE(GRADE, CB) }

SECTIONS: { TITLE, TEACHERS, STUDENTS, ASSIGNMENTS }

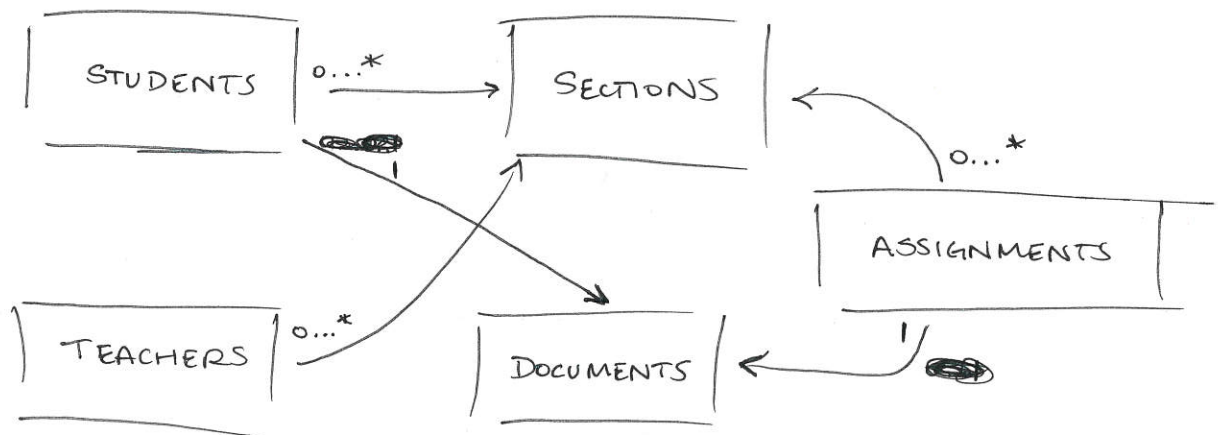
STUDENTS: { EMAIL, HASH, SALT, DOCUMENTS, SECTIONS }

↳ FUNCTIONS: { GENERATEJWT(), SETPASSWORD(PASSWORD), VALIDPASSWORD(PASSWORD) }

TEACHERS: { EMAIL, HASH, SALT, SECTIONS }

↳ FUNCTIONS: { GENERATEJWT(), SETPASSWORD(PASSWORD), VALIDPASSWORD(PASSWORD) }

MODEL COMMUNICATION ON BACK END:



APP MIDDLEWARE: EXPRESS

- SERVER RUNS ON PORT 5000 FROM app.js
- ↳ APP.JS LOADS PACKAGES, CONNECTS TO DATABASE, LOADS ALL MODELS, LOADS ROUTES, SETS A VIEW ENGINE FOR EJS, INITIALIZES MIDDLEWARE (LOGGER, BODY PARSER, COOKIE PARSER, PASSPORT), AND BINDS SERVER ROUTES. 3 ERROR FORWARDERS EXIST AT THE END OF THE APP (BUT DON'T ALWAYS CATCH!)

→ NPM IS CONFIGURED TO DO ALL BINDING AND STARTUP FROM BIN/WWW

1) PACKAGES LOAD ✓

2) MODELS LOAD

↳ SCHEMAS FOR MONGODB ARE INITIALIZED.

3) ROUTES LOAD

↳ THESE ARE NOT URLS. THEY ARE BACK-END ROUTES THAT DESCRIBE THE RELATIONSHIPS BETWEEN EACH COMPONENT OF CONTROL FLOW

4) VIEW ENGINE SET TO EJS

↳ ALLOWS ACCESS TO AN HTML MARKUP FILE AS THE ENTRY POINT TO THE APPLICATION.

↳ SENDS A NON-ANGULAR FRONT END COMPONENT TO CALL ALL ANGULAR COMPONENTS (THIS WAS VERY CONFUSING).

5) MIDDLEWARE INITIALIZATION

↳ BODY PARSER: ALLOWS POSTED JSON FILES TO BE READ

↳ EXPRESS.STATIC: SETS /PUBLIC TO STATIC FRONT-END

↳ COOKIE PARSER: TRACKS & READS COOKIES (FOR JSON WEB TOKENS)

↳ INITIALIZE PASSPORT

→ CONTAINS 2 AUTHENTICATION MODELS 'student-local' & 'teacher-local'

6) ERROR FORWARDING

↳ CATCH 404: PRINTS STACK TRACE TO USER SCREEN (BAD)

↳ DEV HANDLER FOR 500 (APP CRASH): PRINTS STACK TRACE FOR DEVELOPER

↳ PRODUCTION HANDLER: NO STACK TRACES LEAKED...

BACK-END ROUTES:

- INDEX.JS → SETS '/' AS ROOT & RENDERS INDEX.EJS
 - ↳ LOADS ALL ANGULAR COMPONENTS & STARTS FRONT END APP
 - USING A MARKUP FILE IS NOT AN ACCEPTABLE ENTERPRISE PRACTICE.
- ASSIGNMENTS.JS → PARAMETER : ASSIGNMENT IN BACK-END URLS
 - DEFINES ASSIGNMENT ROUTES.
 - /: ASSIGNMENT : POPULATES ASSIGNMENT TABLES
 - /: ASSIGNMENT/SUBMISSION : POPULATES SECTION TABLES W/ ASSIGNMENT TABLES
 - ↳ ALLOWS DOCUMENTS TO BE SUBMITTED TO ASSIGNMENTS
 - /: ASSIGNMENT/SUBMISSION/GRADE : UPDATE GRADE FOR ASSIGNMENT
 - ↳ MUST HAVE A SUBMITTED DOCUMENT.
 - /: ASSIGNMENT/TITLE : UPDATE TITLE
 - /: ASSIGNMENT/DESCRIPTION : UPDATE DESCRIPTION
- DOCUMENTS.JS → PARAMETER : DOCUMENT IN BACK-END ROUTES
 - DEFINES DOCUMENT ROUTES
 - /: DOCUMENT : POPULATES DOCUMENT TABLES
 - /SUBMISSIONS/:DOCUMENT : FIND SUBMITTED DOCUMENT FOR TEACHERS
 - /: DOCUMENT/GRAPH : UPDATES CYTOSCAPE GRAPH
 - /: DOCUMENT/TITLE : UPDATES DOCUMENT TITLE
 - /: DOCUMENT/DESCRIPTION : UPDATES DOCUMENT DESCRIPTION
 - /: DOCUMENT : DELETE FUNCTION REMOVES DOCUMENT FROM DB BY ID
- SECTIONS.JS → PARAMETER : SECTION IN BACKEND ROUTES
 - DEFINES SECTION ROUTES
 - /: SECTION : USED TO POPULATE SECTION TABLES
 - /: SECTION : DELETES SECTION BY ITS ID. FROM DB.
 - /: SECTION/STUDENTS/ADD : ADD A STUDENT BY EMAIL. TO SECTION

• STUDENT.A.JS

- /REGISTER : IF TEACHER ACCOUNT UNCHECKED, CREATE ~~USER~~ ^{STUDENT} FROM PROVIDED CREDENTIALS
- /LOGIN : PASSPORT JWT AUTHENTICATION IF FIELDS ARE FILLED
- /EMAIL : GETS STUDENT'S EMAIL
- /DOCUMENTS/REMOVE : REMOVE A DOCUMENT FROM STUDENT LIST
- /SECTIONS/ADD : ADD SECTION TO STUDENT'S SECTION LIST
- /SECTIONS : GET ALL SECTIONS CONTAINING ASSIGNMENTS

• TEACHER.A.JS

- /REGISTER : SAME AS STUDENT, WITH TEACHER ACCOUNT CHECKED.
- /LOGIN : SAME AS STUDENT
- /EMAIL : SAME AS STUDENT
- /SECTIONS/REMOVE : DELETE A SECTION

NEW STRUCTURE PLAN:

- REIMPLEMENT THE FILE STRUCTURE ACCORDING TO THE REST API:

CAPSTONE

- APP // BACK-END
 - └ MODELS
 - └ ROUTES
- NODE_MODULES
- PUBLIC // FRONT END
 - └ APP
 - └ CONTROLLERS
 - └ VIEWS
 - └ PAGES
 - └ INDEX.HTML // REPLACES INDEX.EJS
 - └ APP.JS // HTML5MODE IS SET TO TRUE
 - └ ROUTES.JS // REMOVES W2UI & UI ROUTER
- └ ASSETS // ALL IMPORTED COMPONENTS
 - └ CSS
 - └ FONTS
 - └ JS // ANGULAR, ^{→ REPLACE WITH GRIDS} ~~BOOTSTRAP~~, CYTOSCAPE, JQUERY
- PACKAGE.JSON // DEPENDENCIES
- SERVER.JS // NODEMON SERVER.JS WILL START APP
 - // CONTAINS BOOTING COMPONENTS & LOADS EXPRESS.

NEW STRUCTURE BACKEND:

- ALL ORIGINAL MODELS WILL STILL EXIST.
- ADD ADMIN.JS: WILL AUTHENTICATE TEACHERS BY SLU ID.

CHANGES:

- STUDENTS & TEACHERS AUTHENTICATION METHODS (GENERATE JWT, ETC)

WILL NEED TO BE CONVERTED INTO INDIVIDUAL MODULES (THIS WILL CAUSE A CASCADE OF APP RESTRUCTURING IN ALL OTHER COMPONENTS).

- WE SHOULD STILL INCLUDE A LOCAL STRATEGY, AS PER ENTERPRISE AUTHENTICATION STRUCTURE

- SLU'S SERVER WORKS, BUT WE CAN USE ANY DATABASE (EVEN LOCAL). I'VE BEEN USING MLAB.

- ADD A PRE (SAVE) MODULE FOR ENCRYPTION IN SCHEMAS.

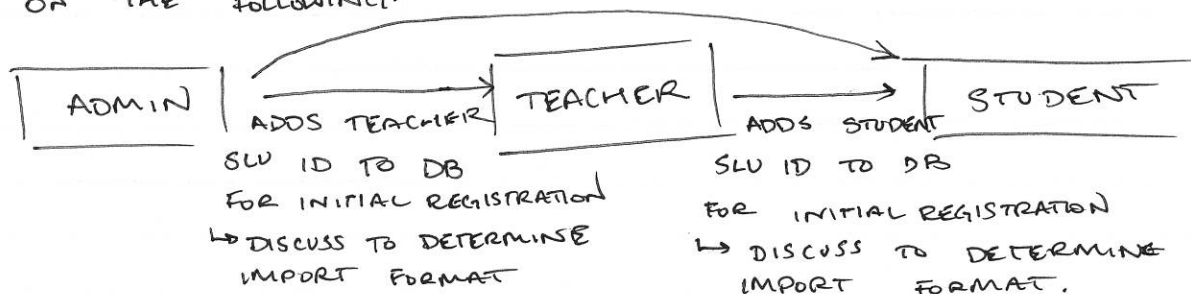
↳ BCrypt IS MORE SECURE THAN JWT pbkdf2Sync HASH/SALT

NEW STRUCTURE MIDDLEWARE:

- INDEX.EJS MUST BE REMOVED, AS PER ENTERPRISE APP DEVELOPMENT
- ALL ROUTES WILL BE INITIALIZED WITH /API TO DIFFERENTIATE BETWEEN ANGULAR AND BACKEND ROUTES.
- INITIALLY, 404'S ARE NOT ALLOWED. ANY WRONG URL WILL REDIRECT TO HOME PAGE WITH A WARNING BANNER.

CHANGES:

- ALL VIEWS ARE CONTAINED IN ANGULAR
- CONFIG CONTAINS KEYS & PASSPORT
 - ↳ KEYS STORE CLIENTID, CLIENT SECRET, HASH ~~SECRET~~ SECRETS, DBURIS, ETC.
 - ↳ PASSPORT WILL UTILIZE MULTIPLE STRATEGIES STRUCTURE
 - GOOGLE OAUTH AND A LOCAL STRATEGY WILL EXIST
 - REGISTRATION REQUIRES SECTION ENROLLMENT BASED ON THE FOLLOWING:



→ THIS SCHEMA REQUIRES:

- ROUTES TO ALLOW USER CREATION & DELETION
 - ↳ ADMIN WILL BE A DB USER AND CANNOT BE REMOVED WITHOUT EDITING THE SOFTWARE. (DOCUMENTATION FOR KEYS)
- DB AUTHENTICATED USER LIST TO ALLOW USER REGISTRATION, CHECKING AGAINST SLUID.
 - PASSING TO PROFILE ROUTES
- ALL ROUTES WILL BE PLACED IN API.JS UNDER APP/ROUTES DIRECTORY; PARAMS WILL BE USED FOR ORGANIZATION.

NEW STRUCTURE FRONT END:

- ALL ANGULAR WILL BE REWRITTEN USING ANGULAR V1.6.8
 - ↳ STATES NEED TO BE REMOVED. ROUTING WILL BE BASED ON A LOCAL ROUTING STRATEGY, AS PER ENTERPRISE APP STRUCTURE
 - ↳ AUTHENTICATION "MODULE" WILL NEED TO BE BROKEN INTO SEVERAL MODULES BASED ON BACK END AUTHENTICATION STRATEGY, THEN FEED TO A MAIN AUTHENTICATION CONTROLLER
 - HTML5MODE IS REQUIRED BY OAUTH API CONSOLE (NO URL FRAGMENTS)
- CHANGES: CURRENT STRUCTURE IS UNSUSTAINABLE...

- ALL ANGULAR WILL BE REWRITTEN...
 - ↳ CONTROLLERS, FACTORIES, SERVICES, AND DIRECTIVES WILL BE DESIGNED ACCORDING TO REST API
 - ↳ ROUTING WILL USE \$ROUTE PROVIDER INSTEAD OF THE VERY CONFUSING UI-ROUTER/STATE IMPLEMENTATION THAT CURRENTLY EXISTS.
- ↳ HTML WILL NOT USE BOOTSTRAP, AS PER ENTERPRISE STRUCTURE
 - HTML WILL BE REWRITTEN USING OUR OWN CSS.
- ↳ SCRIPTS WILL UTILIZE LOCAL ASSETS INSTEAD OF CDN, AS PER ENTERPRISE STRUCTURE

TO DO: FEBRUARY 12:

QUESTIONS TO ADDRESS:

- HOW ARE USERS (TEACHERS & STUDENTS) PLACED IN SECTIONS ON, SAY MYSLU BLACKBOARD?
- WHAT DATA CAN WE IMMEDIATELY ACCESS WITH A SLU ID?
- WHERE DO EMMETT & IMAD STAND?
 - ↳ I NEED THEM TO WORK ON ANGULAR WHILE I RESTRUCTURE BACK-END
- → FIGURE OUT HOW TO PULL REQUEST ON REFRESH IN HTML5 MODE
- WHERE DOES THE WEBPAGE FORWARDING URL STAND?

FORMAT .CSV FOR DATA PULL.

↳ CREATE A TABLE WITH NECESSARY INFO

→ FORWARDING ADDRESS

→ UPLOAD CURRENT DIRECTORY STRUCTURE

↳ GITLAB??