1. **Развертывание проекта**
2. С нуля:

- Установить последнюю версию noedjs, добавить пути к node и к node\_modules в переменные среды (http://askubuntu.com/questions/426750/how-can-i-update-my-nodejs-to-the-latest-version)

- Установить модули фрэймворка express (консоль с админскими правами, npm i -g express-generator). В папке проекта в консоле ввести команду express.

- Установить основные модули node и экспресс (npm i --save express serve-favicon body-parser cookie-parser express-session express-react-view express-react-engine react react-dom winston nconf pg babel-preset-es2015 babel-preset-react)

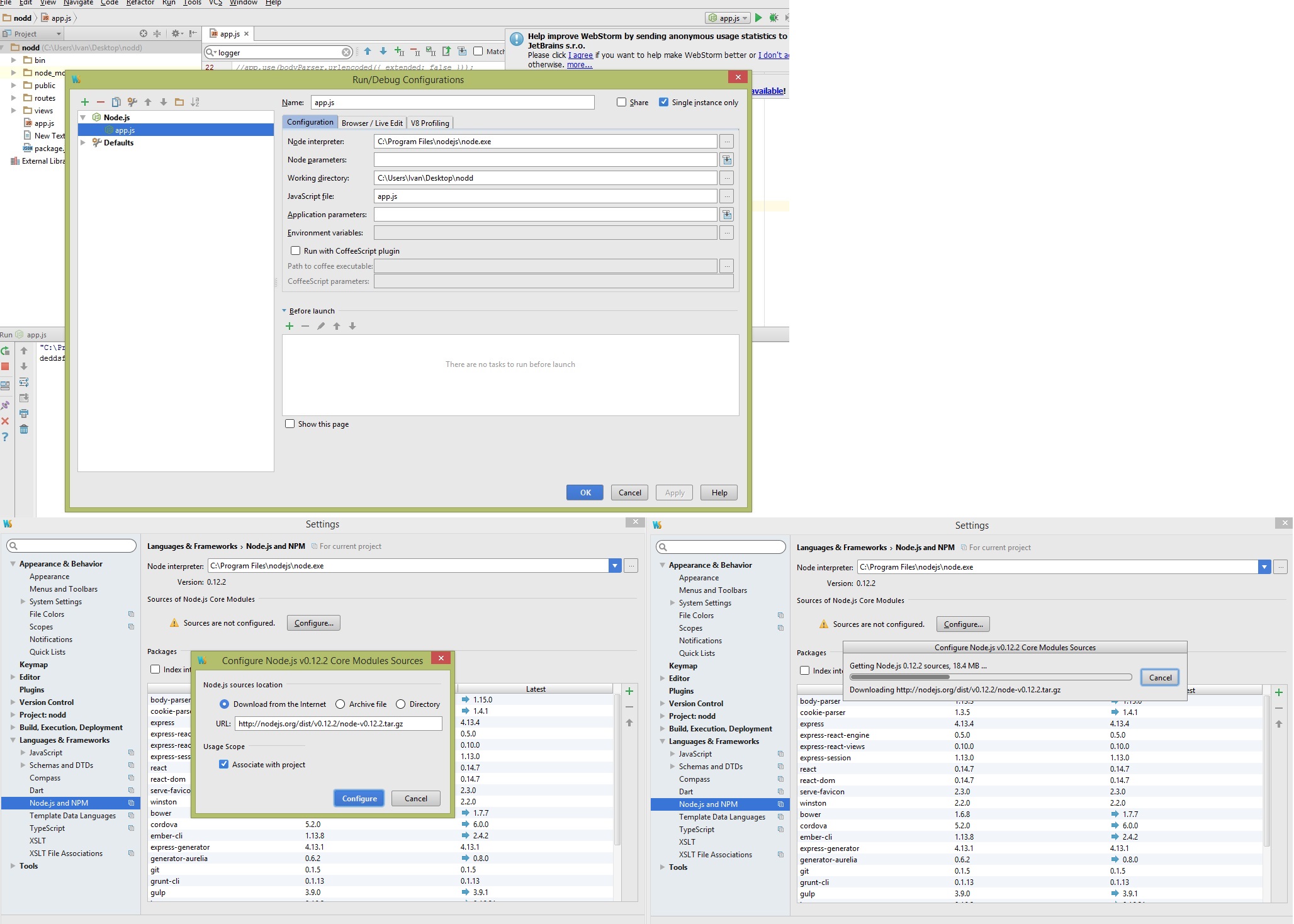
- bower init

1. Если проект загружен с репозитория и в packeges.json уже прописаны все нужные модули:

- В папке проекта в консоле npm install, bower install

- Если загружены модули node, котороые впоследствии были удалены из package.json, нужно удалить их из node\_modules и в консоле выполнить npm cahce clean

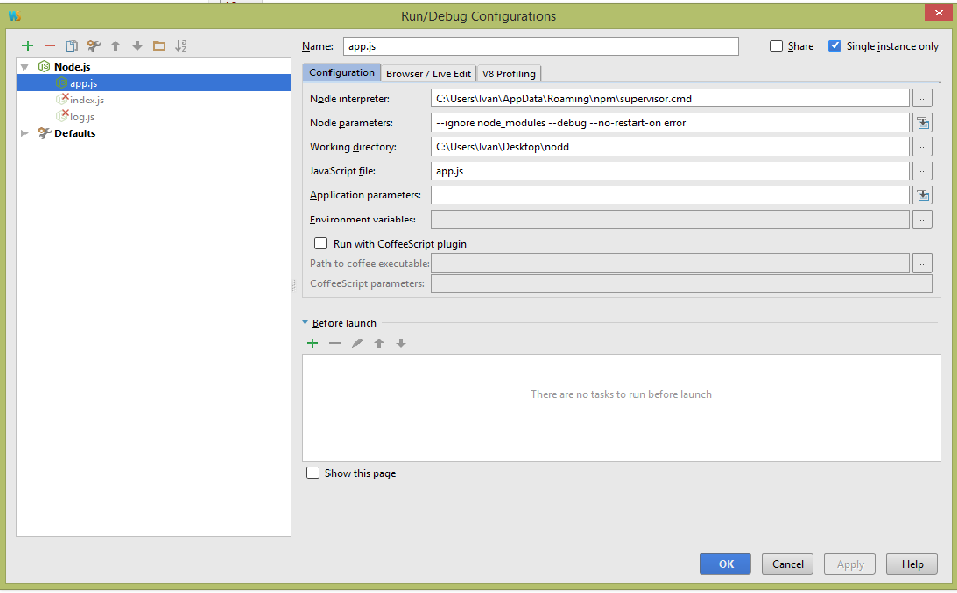
3) Настройка вебсторма: (--debug --ignore node\_modules --no-restart-on error /usr/local/bin/supervisor или /usr/bin/node)



1. **Настройка средств nodejs по автоматическому перезапуску приложения (**live-reload**) при изменении кода и средств дебага:**
2. Supervisor:

- Консоль из-под админа, npm i -g supervisor

- Использование: supervisor xxx.js --ignore node\_modules (игнор сторонних модулей для экономии ресурсов) для старта проекта под супервизором, либо назначить его точкой входа в проект в вебсторме C:\Users\Ivan\AppData\Roaming\npm\supervisor.cmd.



- Особенность: супервизор отслеживает изменения через зависимости (если добавляются новые файлы, а перезапуск супервизора не происходит), поэтому, ели в проект добавлять файлы, которые, как модули, ни к чему не подключены, нужно перезапустить проект, чтобы супервизор их подцепил.

1. Debug:

- хак убунты sudo apt-get install nodejs-legacy

- Консоль из-под админа: npm i -g node-inspector

- Использование: запуск приложения с параметром –debug заставит node слушать специальный порт на предмет команд отладки.

node-inspector – это модуль, который, когда мы его запускаем, подключается к этому порту и предоставляет удобный веб-интерфейс для отладки сервера через этот порт. То есть: запускаем сервер с параметром –debug, запускаем в консоле node-inspector, он покажет url, по которому можно производить отладку в браузере, например, - <http://127.0.0.1:8080/debug?port=5858>. Остановка будет осуществляться по брэйкпоинтам и строке debugger;, вставленной в код.

- Можно включить в отладчике старт дебага на ошибке. --debug-brk сразу стопорит поток выполнения. Или через вебсторм.

\*NODE\_DEBUG "http net" node xxx.js - дебаг нода через консоль

1. **Установка базы postgresql, liquidbase**

- поставить postgres с официального сайта (в комплекте простенькая админка и дрова) http://www.postgresql.org/download/windows/

- создать через pgAdmin базу engin (насчет новой схемы пока вопрос, нужно ли)

- хак пароля для убунту:

sudo -u postgres psql postgres

alter user postgres with password 'postgres';

- скачать jdbc-драйвер и скачать ликвидбэйс (https://github.com/Heigvd/Wegas/wiki/Using-Liquibase-and-DB-changes):

https://jdbc.postgresql.org/download.html (JDBC4 Postgresql Driver, Version 9.4-1208)

http://www.liquibase.org/download/index.html (папку можно кинуть куда удобно, хоть на диск С)

- *если проект пустой:*

*liquibase --url="jdbc:postgresql://localhost:5432/engin" --changeLogFile="chlg1.xml" --username=postgres --password=ParPar52 generateChangeLog*

- развернуть ликвидбэйс:

liquibase.properties и postgresql-9.4.1208.jre6.jar кинуть в папку ликвида, скачанную с www.liquibase.org

run.bat.example в папку ликвида, находящуюся в проекте, в ней же должны быть ченджлоги

(https://habrahabr.ru/post/179425/)

rollback version-1.0 вместо update для ролбэка

1. **Установка монго-базы**
2. Скачать монго с сайта <https://www.mongodb.org/downloads#production>
3. Установить
4. Выполнить команду md data\db в нужной директории для создания папки под базу данных
5. Запустить mongod (сервер) перед стартом сервера или началом работы с монго-базой.

**Полезные ссылки:**

http://design.tutsplus.com/tutorials/use-adobe-illustrator-to-create-a-clean-website-layout--vector-596

http://create.adobe.com/2014/11/12/adobe\_illustrator\_cc\_for\_web\_designers.html

https://helpx.adobe.com/illustrator/how-to/export-css.html

http://sketchapp.me/dizajnery-google-o-rabote-v-sketch-i-ego-i-budushhem/

https://habrahabr.ru/post/226025/

http://seldo.com/weblog/2011/08/11/orm\_is\_an\_antipattern

http://docs.sequelizejs.com/en/latest/api/associations/belongs-to-many/

http://vincit.github.io/objection.js/#introduction

http://stackoverflow.com/questions/10060265/sequelize-how-to-import-definitions-from-an-existing-database

http://stackoverflow.com/questions/27117337/exclude-route-from-express-middleware

<https://wiki.postgresql.org/wiki/Why_PostgreSQL_Instead_of_MySQL_2009>

https://nodejs.org/en/docs/es6/

http://jsfiddle.net/3hLm7tet/5/

http://jsfiddle.net/3hLm7tet/12/