

**Тема:** обратна връзка дипломна

**От:** Radostin Kostadinov <dpetrov@radostin.bg>

**Дата:** 25.01.2023 г., 15:59

**До:** Radostin Kostadinov <bg.radostin.kostadinov@gmail.com>

**СС:** Eurban Vangel <eurban.vangel@radostin.bg>, Eurban Vangel <eurban.vangel@radostin.bg>, Eurban Vangel <eurban.vangel@radostin.bg>, Eurban Vangel <eurban.vangel@radostin.bg>, Eurban Vangel <eurban.vangel@radostin.bg>, Eurban Vangel <eurban.vangel@radostin.bg>, Eurban Vangel <eurban.vangel@radostin.bg>

Здравей Радо,

пускам ти консолидирана обратна връзка от всички колеги относно дипломната ти. Надявам се да приемеш конструктивно критичните точки, за да изградиш правилните навици. Много от нещата са дребни и може да си ги коригираш когато имаш време за да ти е по-завършен проекта. Имай в предвид че портфолио от проекти винаги може да е полезно. *с дипломната и задачите като с работните си, имаш поне две*

Биле пописаните от екипите, а не само обратна връзка и бързи дребни неща при поименното й. Да се развлича по бързо. Обратната връзка е нещо което е много ценно, но когато условия да се даде, трудно се сваля от получателята страна и на много е по-лесно, а когато е по-лесно да се даде, трудно се сваля от получателята страна и на много е по-лесно.

Поздрави,

*Radostin*

### ### Back-end

- добре е че пишеш коментари, но е прието да се пишат на английски за да може всеки да ги разчете. Не знаеш кога кой проект от кого ще се продължи за работа и това трябва да ти е навик. Друга опция е да са дублирани английски + друг език.
- **основна грешка** при логин операцията връщаш хеширания парола на потребителя. Това първо е ненужно (за какво му е на фронтенда?) и второ е потенциален security breach. За целта се маха от обекта ръчно или модифицираш toJSON метода на mongoose (пропускаш/махаш паролата) и този метод се извиква преди всяко връщане на резултата.
- хеширането може да правиш използвайки hooks на mongoose
- на някои места валидираш бодито от заявката но не валидираш конкретните параметри и ги използваш директно. Може да се подава или липсващ или грешен тип. Като цяло валидациите винаги са и на ниво фронтенд (подказваш на юзъра и предпазваш бекенда) и на ниво бекенд (не разчиташ че фронтенда е твой или че не се ползва някой друг туул) независимо че ти в случая пишеш и двете, т.е. знаеш че си си проверил на едното място.
- когато стартираш приложението първо рънваш http сървър и след това чакаш конекция към базата. По-правилното е обратното - правиш X броя опити за конекция към базата през Y интервал и след последния ако не си се свързал спиращ, т.е. нищо не работи. Ако успееш да се свържеш тогава пускаш и http сървър и чакаш заявки. По този начин избягваш проблем при който получаваш заявка за промяна в базата, когато нямаш реално връзка.
- една препоръка за скелета - кода за определено api да не се пие при дефиниране на routes. Правят се отделни файлове обикновено кръстени controller, отделно кода в контролера се пише в services. Това са практики които спомагат при работа с повече хора всичко да е на точно определено място за да може всеки бързо и лесно да се ориентира, поддържа и тества. Можеш да разгледаш някои видеа по въпроса и гитхъб репота със скелети на приложения.

### ### Front-end

- интерфейса е свързан с темата на приложението, но не е много user-friendly, нещо на което по принцип се обръща доста внимание. Тук въпросът отива към основен проблем - нещо красиво в което не може да се ориентираш кой бутон за какво е и рискуваш да отблъснеш клиентите заради сложността или нещо просто но интуитивно, което обаче не привлича окото и съответно може да не задържи клиента достатъчно дълго за да разбере че е полезно.

Въпрос относно интерфейса - изцяло твое дело ли е или си взимствал дизайна?

- приложението е неизползваемо през телефон, но и с избраният интерфейс не виждаме как може да се коригира лесно. Като цяло при създаването е хубаво да се мисли и по този въпрос тъй като масово вече сайтове се отварят на телефони.

- една препоръка - в модерния уеб, както посочихме и по-горе телефоните се използват масово. В твой случай имаш много подобни или дори еднакви бутони и инпут елементи, които стилизираш поотделно. Избягвай дублиране или използване на много подобен код - мързела в това отношение е плюс и това важи и за фронтенда и за бекенда. В случая ще е по-добре да изкараш глобални стилизации и да ги използваш при нужда или още по-добре да си направиш къстъм бутон/инпут, който да се сменя според параметрите които подаваш.

### ### QA testing

- когато отвориш приложението те пренасочва към логин страницата (или която и да е друга), но ако презаредиш в този момент се чупи и хвърля грешка от Firebase че не намира index.html.

Подозираме че е проблем с конфигурацията на Firebase или Angular.

- нямаш favicon и title на страницата

- на места имаш полузакрит текст, като много вероятно е да зависи и от размера на монитора на който се гледа

- има места, на които изображенията свършват след като ги сменяш и се появява празно място - пример "Праисторията, времето преди създаването на писмеността"

- страницата "Състезания" не може да бъде избрана

- на "Профил" не се появява нищо, има и други страници които предполагахме просто са незавършени

- на моменти, за секунда се появява инфото от друга страница, преди да зареди тази, която съм кликнул

- когато се регистрираш и въвеждаш определени невалидни данни не те допуска до регистрация, но и не се показва съобщение защо и кои са невалидните данни. Също така с невалидни данни бутона за регистрация не светва, но след определен брой символи светва, но пак казва, че данните са невалидни без допълнителна информация

- през админския канал на "състезания и контролни" ми излиза просто един нестилизиран текст "competitions works!" така и с "Провери аватари" само че с "avatars-check works!"

- след като изтрия изображение, когато се опитвам да създам въпрос, после не мога да го кача отново

- въведените уроци се запазват и без натискането на бутона "Запази", а самия той не реагира на натискане

- при преглед на уроци, в полето уроци няма нищо показано

- когато сменяме от един урок на друг, изображенията не си отговарят, променят се и се смесват

- след като се логнеш, сесията не свършва