INFO30005 Project: Instructions

In this semester-long group project you will practice the skills and tools you learn in class by doing web design and development with a team.

Your task is to design and build a web application that meets the requirements provided to you in the separate Business Requirements document.

Figure 1: web dev team at work

Your Group

Projects will be conducted in groups of 5 students. All group members must be enrolled in the same weekly tutorial, and must attend the tutorial every week. (Absences must be justified in writing.) Groups will need to do substantial work outside tutorials as well.

Deliverables: overview

To help you make continual progress, and to provide multiple opportunities for feedback, the project is divided into three deliverables.

All of these are group (rather than individual) submissions. They sum to 50% of your whole-of-semester assessment.

#	Deliverable	Due	%
1	UI prototype + some HTML / CSS files	week 4	10
2	One complete feature: client + server + database	week 8	10
3	Complete web app + source code	week 11	30

Each deliverable is due by midnight Friday of the relevant week, Melbourne time.

Technologies and Tools

For equity reasons, you must use technologies taught in INFO30005 to build your web app.

- the web stack is: Node, Express, Handlebars, MongoDB
- supported by: authentication using Passport, database access using Mongoose
- the programming language is: JavaScript
- codebase is maintained on: GitHub
- web app is deployed to cloud platforms: Heroku, Atlas.

For prototyping we recommend and teach Adobe Xd, but will allow alternatives such as Figma on these conditions: alternatives will not be taught or supported, requirements for the deliverable and marking standards will not change.

Deliverable 1: details

Your first deliverable is:

- Design your web app's UI, and make a clickable prototype using e.g. Adobe Xd (7%)
- Implement two of the web pages in HTML and CSS (3%)

Make clickable prototypes of both the patient and clinician systems. Include all of the screens required to fulfill the business requirements. The patient prototype need only show phone-sized screens, while the clinician prototype need only show desktop-sized screens.

Implement in HTML/CSS the two static webpages "About diabetes" and "About this website". They should share the same external CSS file.

Submit via Canvas:

- a link (URL) to your Xd prototype
- a screenshot of your Xd project showing all artboards
- a link to your GitHub repo containing two HTML files and one CSS file.

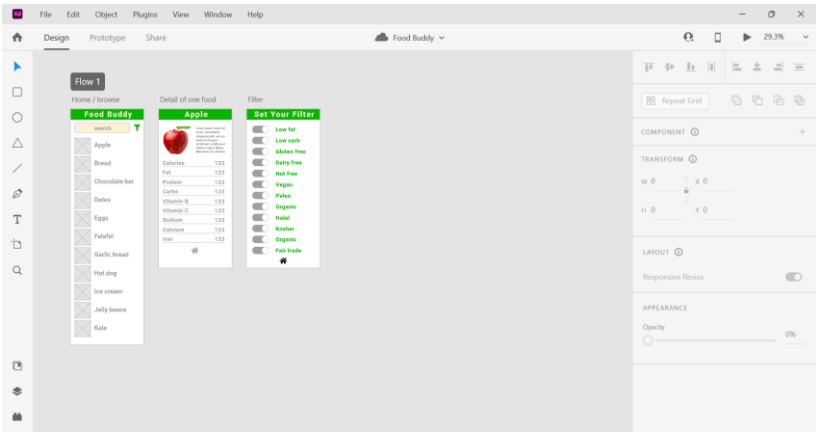

Figure 2: Example screenshot of artboards in an Xd project

Your work will be assessed according to these criteria:

- All business requirements are captured in the prototype
- Realistic data are shown on screens
- Interaction design and screen layout are of good quality
- HTML and CSS code is of good quality.

Deliverable 2: details

By the end of week 8, i.e. midnight on Friday 29th April, you should deliver one working end-to-end feature in your application, which enables a patient to enter some health data, and their clinician to see the data. For D2, your app should support the following scenario:

- Patient *Pat* has logged in and is viewing their home screen. Pat decides to enter today's blood glucose data, with an accompanying comment.
- This data is stored correctly in a MongoDB database on Atlas.
- Pat's screen changes to show that today's glucose data has been entered.
- Clinician *Chris* views their dashboard and sees the new data that Pat entered. The number is highlighted if it is outside Pat's safety threshold.

Note that many aspects of your app are not yet required in Deliverable 2:

- Logins (authentication via Passport) are not required in D2. Instead, you can simply hard-code Pat's and Chris's identities in your code.
- Validation of data entry is not required in D2. Your app can assume that Pat enters appropriate data.
- Software features described in the Requirements document but not listed on this page of the Instructions are not required in D2.
- You may manually place, in your database or screens, data other than what is entered in the scenario above; for example, Pat's safety thresholds, or other patient details for display in Chris's dashboard.

Your group should submit a single PDF that contains:

- Your group and tutorial details
- The URL of your patient home screen, running on Heroku
- The URL of your clinician dashboard screen, running on Heroku
- The URL of your relevant repo on GitHub.
- Tell us which commit in your repo you want to be marked. (It must be dated before the submission deadline. This is so you can keep building D3 while we mark D2.)
- This is a group deliverable. It should be submitted by just one member of the group.

Your work will be assessed according to these criteria:

- Markers can use your app to successfully step through the scenario above.
- UI design is of high quality. The patient UI works well on phone and desktop screens.
- HTML, CSS and JS code is high-quality.
- The database design (structure of data in MongoDB) is high-quality.

Deliverable 3 details

Details of D3 will be supplied later in the semester.