**Application Design Document**

**APPLICATION REQUIREMENTS**

**This is an application that will provide the ability for an individual child to enter deadline details along with a parent user to enter the details for all the family. However, only have appointments and deadlines, do not add further items or categories\* The following user stories provide the basic requirements:**

**As a parent I wish to add details of dentist appointments for all in my family**

**As a parent I wish to see details of upcoming appointments for all in my family**

**As a parent I wish to change the details of any appointments for all in my family**

**As a parent I wish to add notes to an appointment that has taken place**

**As a parent I wish to cancel appointments for any in my family**

**As a child I wish to enter details of deadlines I need to meet**

**As a child I wish to see what deadlines I have coming up**

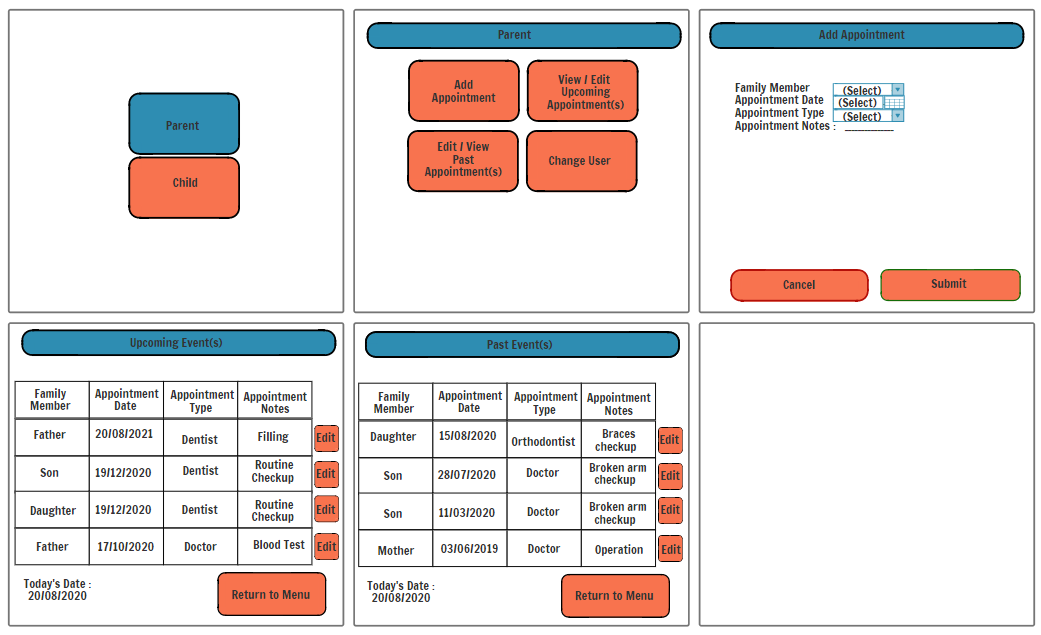
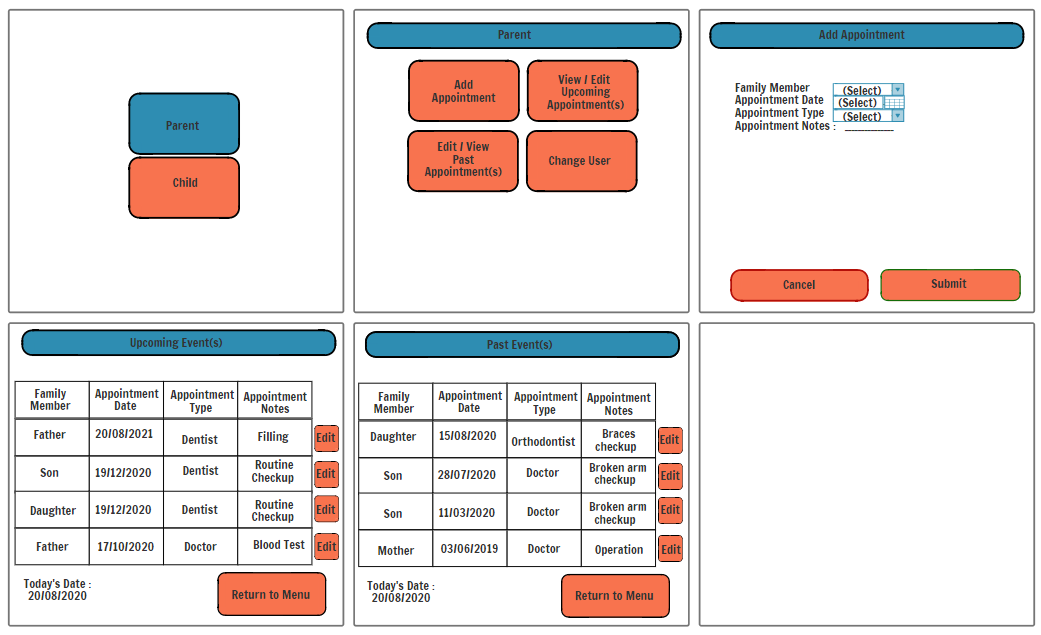
**As a child I wish to amend my deadline to show that I have met it**

**As a child I wish to amend my deadline to move it**

**As a child I wish to delete my deadline**

**\*This is not a full system, this is a minimum viable product (MVP) whereby only the user stories provided above are implemented. The following aspects are outside of the scope of the system: Other users. No other user stories for other users are required. Logging in is outside the scope of the system. You are not to spend time implementing a bespoke log in system.**

**User Story Boards**

**Parent**

The first stage of the user journey is that the user is give the option to pick between the child and parent users.

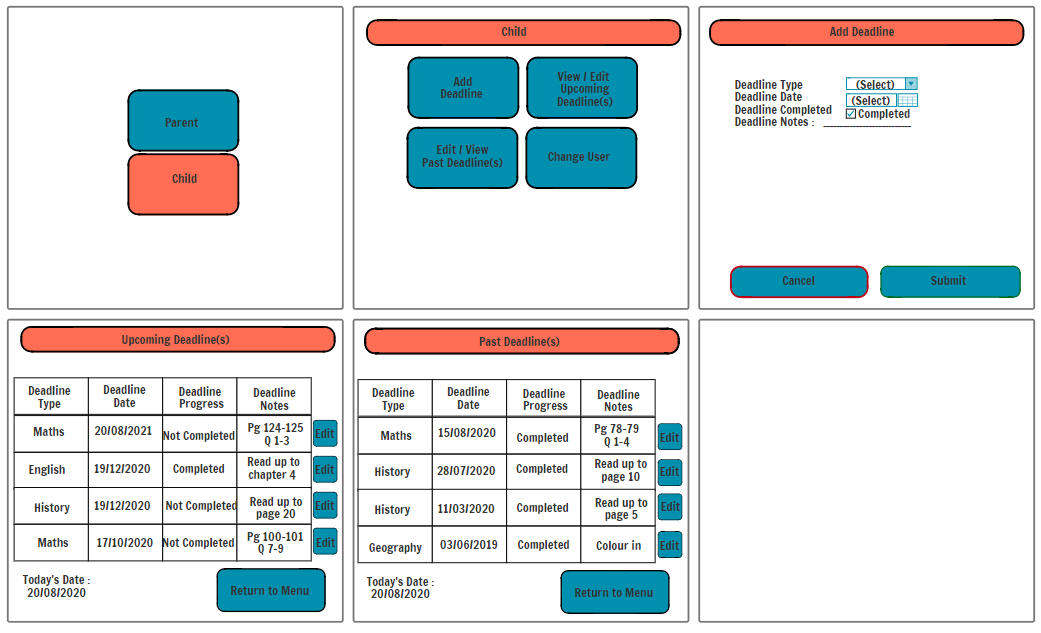
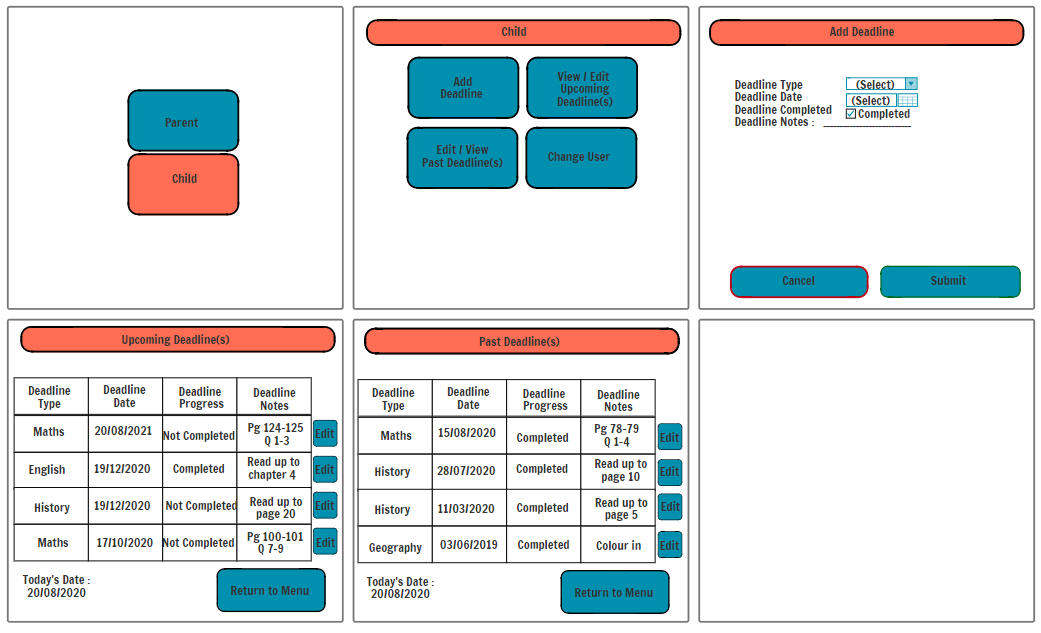
On selection of the parent user the parent user panel is shown, from here the parent choses depending on what they wish to do.

On selection of the add appointment button the user is take to the add appointment page. The user is then to fill out the new appointment form which uses drop down menus to select Family member and Appointment type. Appointment date is selected use a date time selector. Appointment notes is a text field input for the user to add a few notes to do with the appointment.

On selection of the Edit/View Upcoming Appointments button the user is shown all upcoming appointments sorted by appointment date. They can use a scroller to scroll up and down through the different appointments. They can press an edit button that will allow them to edit an appointments data, this will update the table when the user presses the return to menu button.

On selection of the Edit/View Past Appointments button the user is shown all past appointments sorted by appointment date. They can use a scroller to scroll up and down through the different appointments. They can press an edit button that will allow them to edit an appointments notes, this will update the table when the user presses the return to menu button.

**Child**



The first stage of the user journey is that the user is give the option to pick between the child and parent users.

On selection of the child button the child user panel is shown. From here the child can chose how they wish to interact with their deadline data.

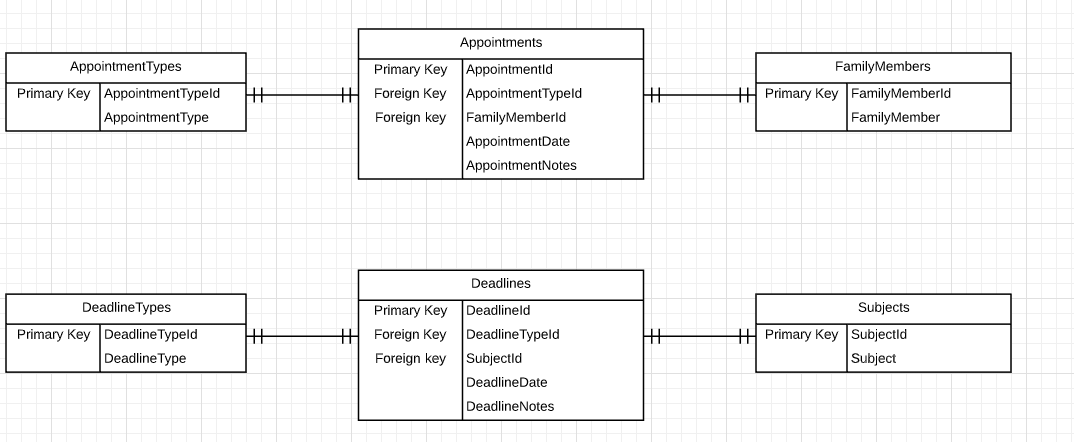
On selection of the Add deadline button the Add Deadline page is shown that allows the child to input a new deadline using widgets to easily facilitate this. Upon submission an alert is sent to the user to alert them of the success or failure of adding the new deadline. They are then returned to the child user panel.

On selection of Edit/View Upcoming Deadline(s) the user is present with all their upcoming deadlines. The blue button next to the specific deadline can be pressed to allow the user to edit the deadline and to select if they have completed it. This will update when the user returns to the menu.

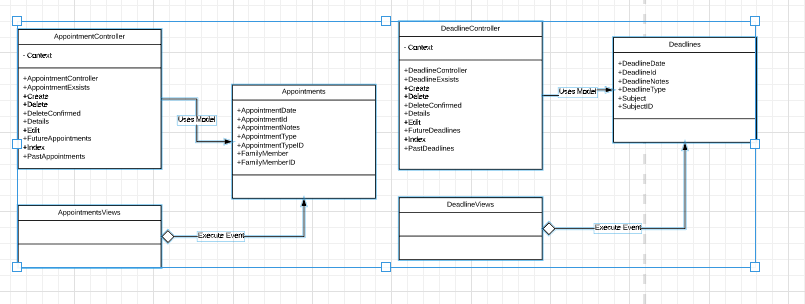
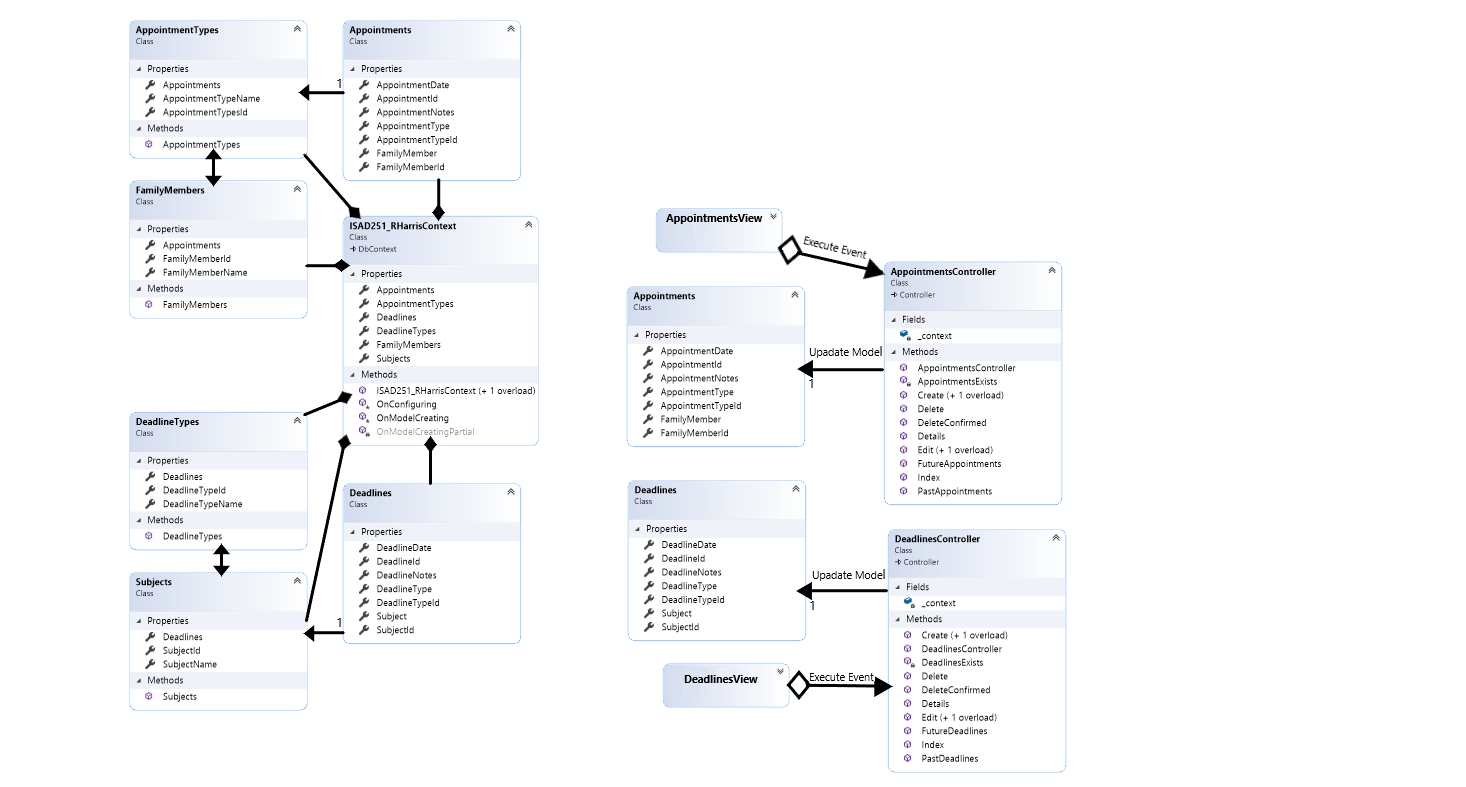
On selection of Edit/View Past Deadline(s) the user is present with all their past deadlines. The blue button next to the specific deadline can be pressed to allow the user to edit the deadline and to select if they have completed it. This will update when the user returns to the menu.

The user can select the change user button the return to the starting screen.

**ERD**



**UML**

****