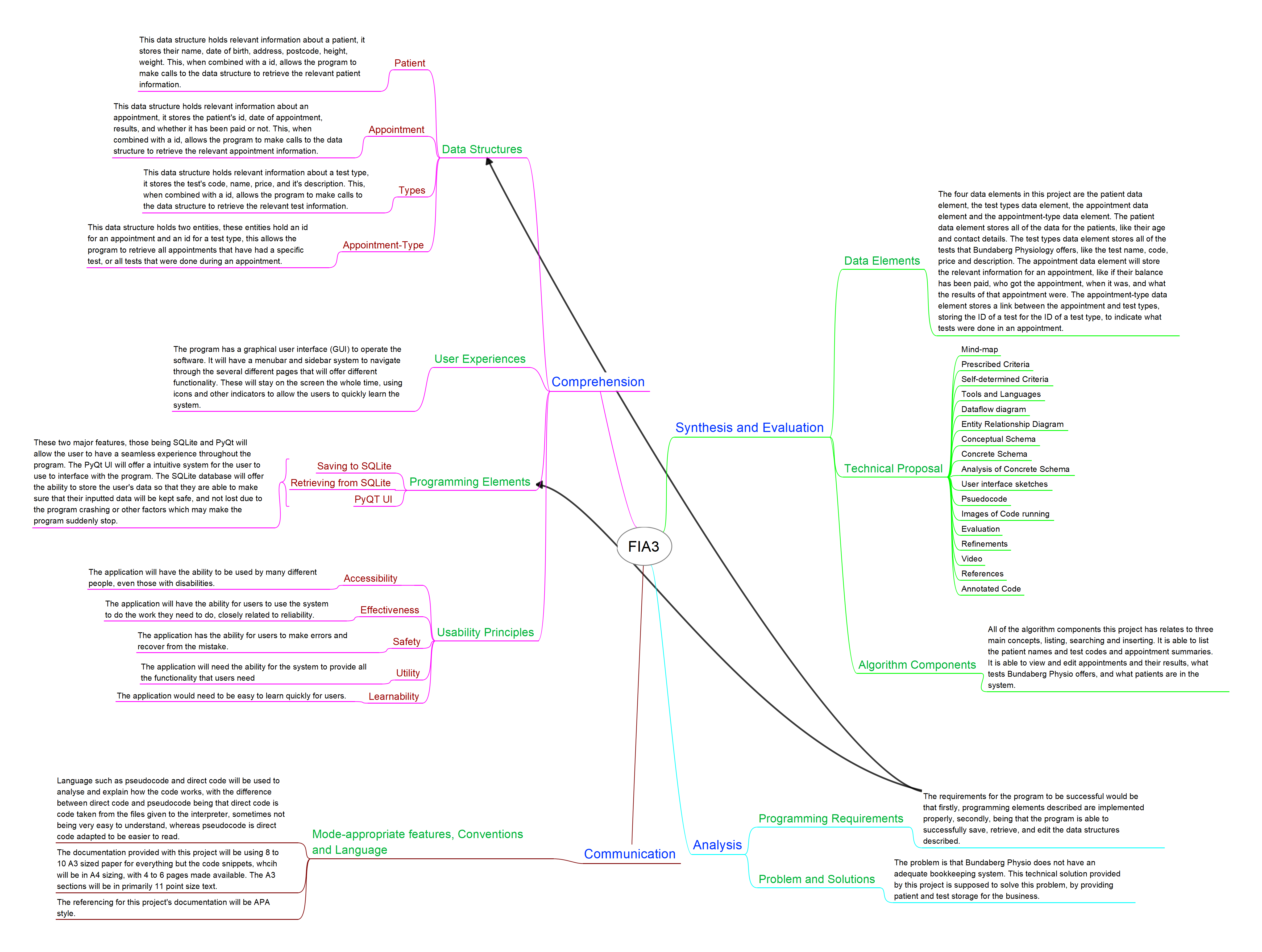
## Mind Map



## Dataflow Diagram

## Prescribed and Self-Determined Criteria

### Prescribed

Comprehending:

* The prototype must recognise and describe programming elements, user interface components and useability principles.
* The report must symbolise and explain programming information & ideas, data structures and interrelationships between user experiences & data of the prototype.

Analysing:

* The report must analyse the problem and information related to the technical proposal for a low-fidelity prototype digital solution.
* The report must determine user interface, data, programmed and solution requirements of the digital solution.

Synthesising and Evaluating:

* The report must synthesise information and ideas to determine user interface and programmed components for a digital solution.
* The user interfaces and programmed components of the digital solution must be generated.
* The report must evaluate impacts, components, and the digital solution against prescribed and self-determined criteria to make refinements and recommendations.
* The report must include decisions about and use mode-appropriate features, written language, and conventions for a technical audience.

### Self-Determined

* The application must maintain compatibility across major desktop operating systems, to allow for most compatibility.
* For the application, reliability must be a priority, there can’t be any bugs which prevent major features from working.
* The application must use python as it is a requirement, due to the use of PyQT5 and the need for the ability to produce prototypes quickly.
* The application needs to make it clear to what they are expected to do, if the experience confuses the user, it impedes the program altogether.

A picture containing text, diagram, parallel, plan

Description automatically generated

# Tools and Languages

Visual Studio Code

I am using Visual Studio Code as it is a powerful code editor with support for many languages, such as Python, which will be used in this project. It has support for actions like debugging and version control. The aim of Visual Studio code is to provide a slimmed down version of Visual Studio who developers with simpler workflows of mostly code, build then debug. It is also cross platform and has a wide range of extensions which can enhance the programming experience. These features are why it will be used over other tools like PyCharm for this project.

Word

Microsoft Word will be used in this project for word processing and note taking. Microsoft Word is a word-processing application that allows for the creation of simple and complex documents, some of the advantages of Microsoft Word is broad availability, simple UI, two-click error fixing, instant help feature, many dictionaries embedded, variation of document sizes, customisability and simple editing. This is being used over other tools like Libre Writer due to its seamless interconnection between other Microsoft products in the Microsoft 365 family. These features/benefits are why Microsoft Word will be used for this project.

Lucid chart

Lucid chart is a web-based diagram software, it was initially released in 2008, and allows users to visually draw, revise, and share diagrams. It is based on the latest web protocols, meaning no extra extensions are required to get started, it has an easy to understand drag and drop interface that allows users to go from beginning to finished within hours, much faster than doing it by hand. These features are why it will be used in this project for making the data flow diagram, over other tools like Draw.io.

PyQt5 + Designer

PyQt5 will be used in this project to develop the graphical user interface (GUI) for the prototype. PyQt5 is a powerful Python binding for the Qt framework. With PyQt5, it is possible to design interactive GUIs for the application. The versatility of PyQt5 allows for rapid prototyping, thus fulfilling both self-determined and prescribed criteria. Additionally, PyQt5 offers the flexibility to create reusable templates, making it possible to quickly produced subsequent widgets. These features are why PyQt5 will be used over other GUI interface tools like Tkinter.

Freeplane

Freeplane is a piece of software that allows for mind map (diagrams of connections between a set of ideas). It was forked from the FreeMind project in 2007 and is opensource under the GPL “version 2 or later” license. It is being used in this project for the simplicity of the main functions and the complexity of several functions allow it to create both simple and complex mind-maps. This will be used for creating the mind-map of this project. Freeplane will be used instead of other tools like Mind Meister for these reasons.

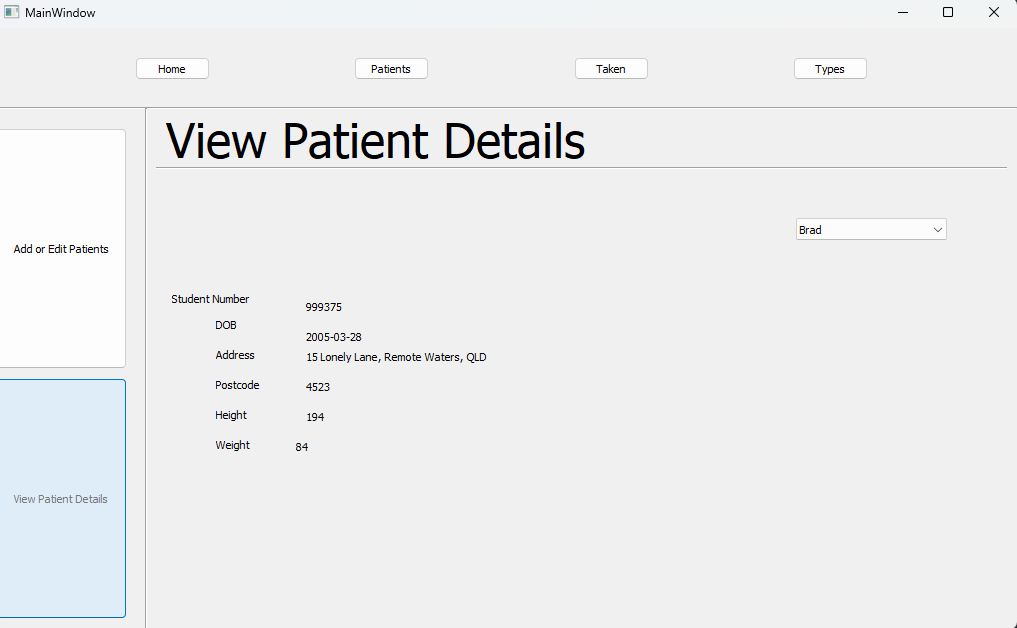
GitHub Desktop

GitHub Desktop will be used in this project for version control. The reason for using GitHub Desktop for version control over other solutions is that it has great integration with GitHub, the source control website I’ll use to store the code. GitHub Desktop takes the command-line git and puts most of its actions and uses into a GUI, preventing the need to touch a command line usually ever, when it’s related to git. It also has support for using different websites as the remote (code location on a server somewhere other than the local machine), allowing it to also be used with websites like GitLab. This is being used over other GUI git tools like SourceTree due to simple setup and easy to understand user interface.

Adobe XD

Adobe XD will be used in this project to produce wireframes for the planning stage of this prototype. Adobe XD is “a versatile page layout design application that lets users build realistic, working prototypes of web layouts for testing and collaboration.” (SOURCE HERE). Using Adobe XD to produce wireframes will allow to quickly revise and make changes to them, as well as allowing for creating templates that can be used to make other wireframes, allowing for quick production of more wireframes. This is being used over other tools like Figma due to being provided by the school and having good interconnection between other Adobe products.

# Evidence of program running

A screenshot of a computer

Description automatically generated with medium confidenceA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidence

A diagram of a diagram

Description automatically generated

# Concrete Schema

CREATE TABLE "Appointment" (

"id" INTEGER NOT NULL UNIQUE,

"patientid" INTEGER NOT NULL,

"date" TEXT NOT NULL,

"result" TEXT NOT NULL,

"paid" INTEGER NOT NULL,

PRIMARY KEY("id" AUTOINCREMENT),

FOREIGN KEY("patientid") REFERENCES "Patient"("id")

);

CREATE TABLE "AppointmentType" (

"appointment\_id" INTEGER NOT NULL,

"type\_id" INTEGER NOT NULL,

FOREIGN KEY("appointment\_id") REFERENCES "Appointment"("id")

);

CREATE TABLE "Patient" (

"id" INTEGER NOT NULL UNIQUE,

"name" TEXT NOT NULL,

"dob" TEXT NOT NULL,

"address" TEXT NOT NULL,

"post" TEXT NOT NULL,

"height" INTEGER NOT NULL,

"weight" INTEGER NOT NULL,

PRIMARY KEY("id" AUTOINCREMENT)

);

CREATE TABLE "Type" (

"id" INTEGER NOT NULL UNIQUE,

"code" TEXT NOT NULL UNIQUE,

"name" TEXT NOT NULL,

"desc" TEXT NOT NULL,

"price" INTEGER NOT NULL,

PRIMARY KEY("id" AUTOINCREMENT)

);

# Data Dictionaries

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Validation Rules | Example Data |
| Appointment |  |  |  |
| id | INTEGER | PRIMARY KEY AUTO INCREMENTING | 12 |
| patientid | INTEGER | NOT NULL | 14 |
| date | TEXT | NOT NULL | 2021-11-12 |
| result | TEXT | NOT NULL | Positive |
| paid | INTEGER | NOT NULL | 0 |

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Validation Rules | Example Data |
| AppointmentType |  |  |  |
| appointmentid | INTEGER | FOREIGN KEY NOT NULL | 17 |
| typeid | INTEGER | FOREIGN KEY NOT NULL | 42 |

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Validation Rules | Example Data |
| Patient |  |  |  |
| Id | INTEGER | PRIMARY KEY AUTO INCREMENTING | 2 |
| name | TEXT | NOT NULL | Sam H |
| dob | TEXT | NOT NULL | 1999-11-12 |
| address | TEXT | NOT NULL | 17 Isaac Street, Bargara |
| post | TEXT | NOT NULL | 4670 |
| height | INTEGER | NOT NULL | 112 |
| weight | INTEGER | NOT NULL | 17 |

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Validation Rules | Example Data |
| Types |  |  |  |
| id | INTEGER | PRIMARY KEY AUTO INCREMENTING | 4 |
| code | TEXT | NOT NULL | XRY |
| name | TEXT | NOT NULL | X-Ray |
| desc | TEXT | NOT NULL | X-Ray for patient |
| price | INTEGER | NOT NULL | 117 |

# Conceptual Schema

Appointment (

id PK INT NOT NULL

patientid FK INT NOT NULL,

date TEXT NOT NULL

result TEXT NOT NULL

paid TEXT NOT NULL

)

AppointmentType (

appointmentid FK INT NOT NULL

typeid FK INT NOT NULL

)

Test Type (

id PK INT NOT NULL

code INT NOT NULL

name TEXT NOT NULL

desc TEXT NOT NULL

price INT NOT NULL

)

Patient (

id INT PK NOT NULL

name TEXT NOT NULL

studnum INT UNIQUE

dob TEXT NOT NULL

address TEXT NOT NULL

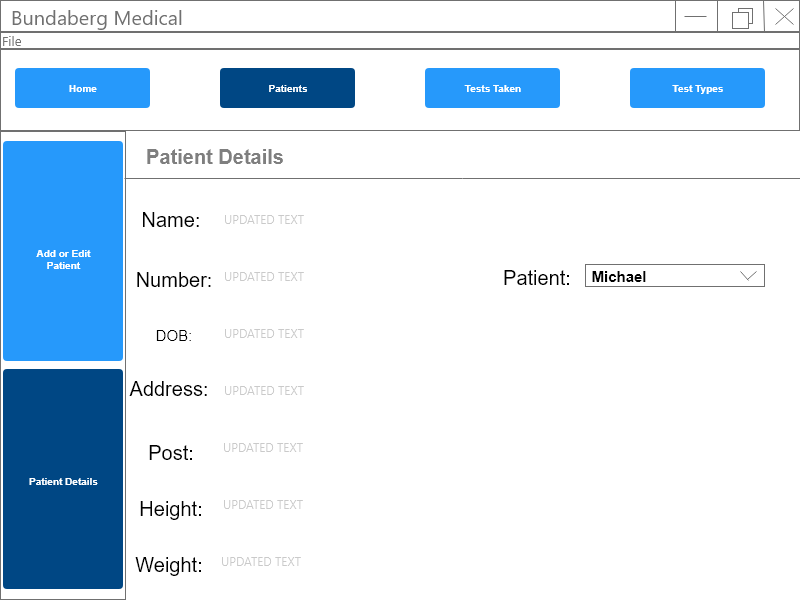
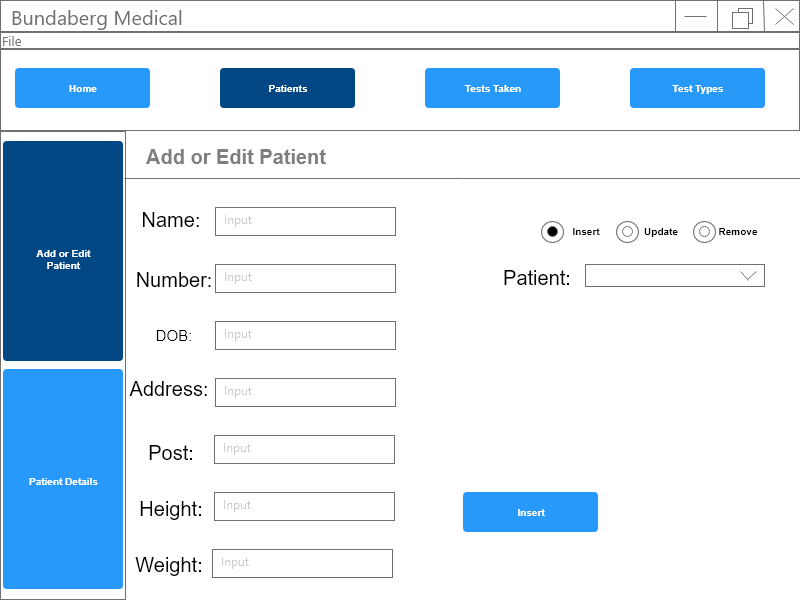
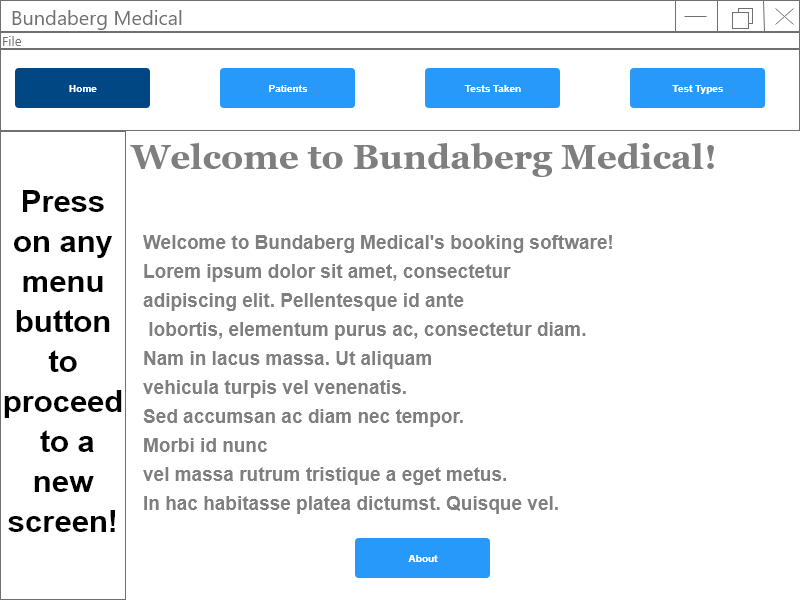
post TEXT NOT NULL

height INT NOT NULL

weight INT NOT NULL

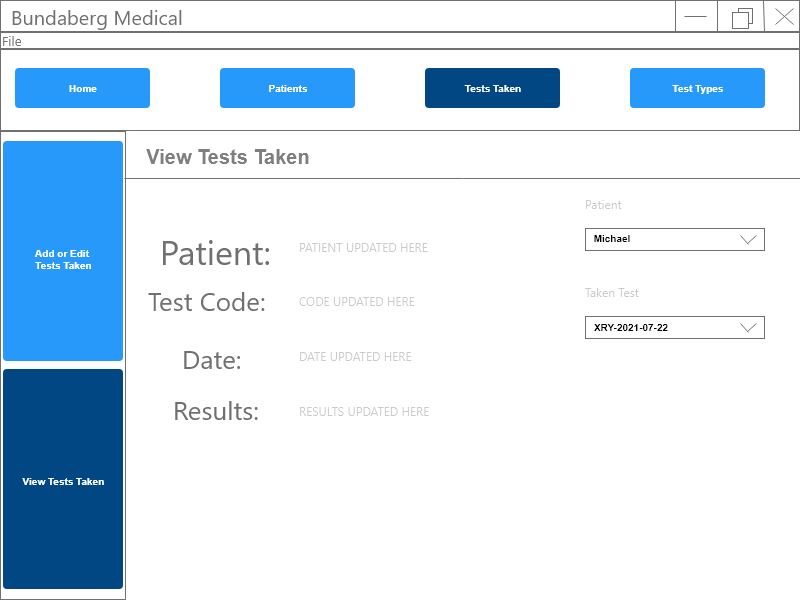
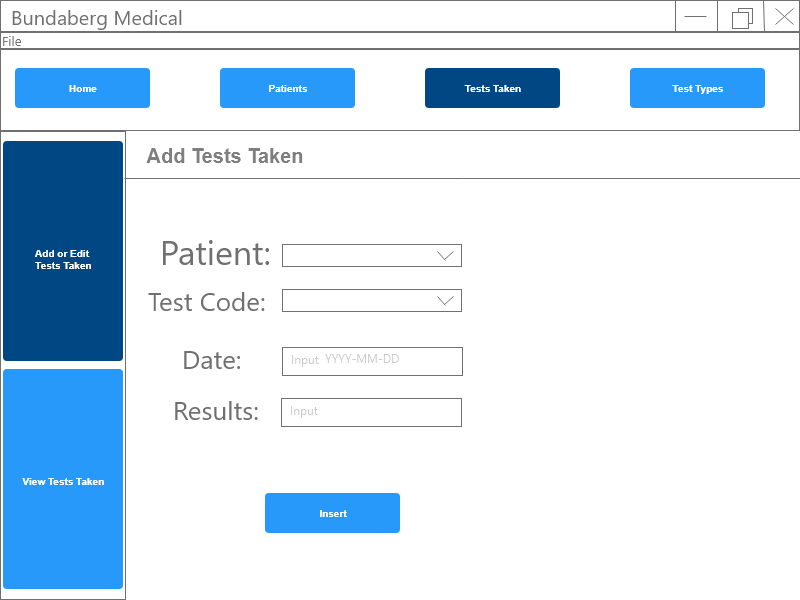
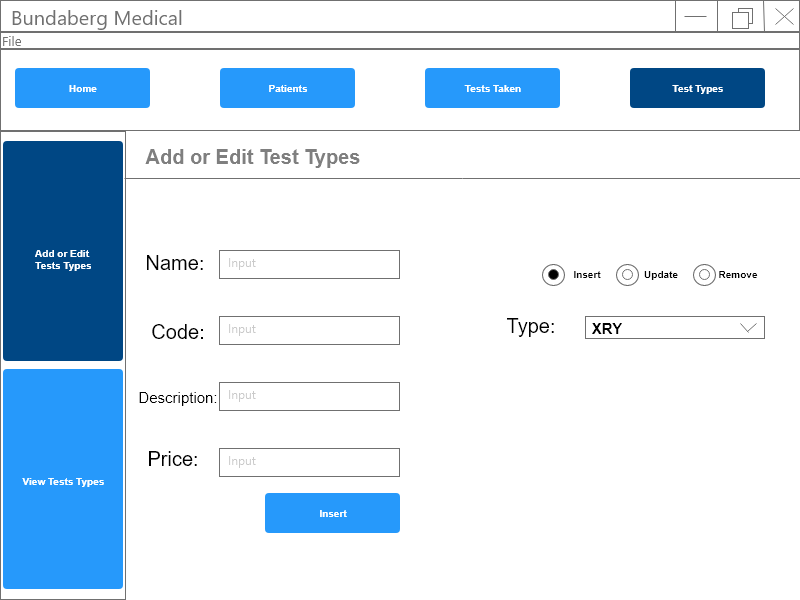
);

# Wireframe

A screenshot of a medical test

Description automatically generated with low confidence

System provides all of the functionality the previous text system offered – related to **utility** useability principle.



Taken tests update immediately upon changing of patient and text updates upon changing of test – apart of **effectiveness** useability principal.

Clearly labelled buttons – related to **accessibility.** useability principle.

Combo-box with test codes already filled in – prevents any non-existing tests – part of **safety** useability principal.

Radio buttons with an option already selected – so that no mode selected cannot happen – part of useability principal **safety**.

Same UI for side and menu buttons – related to **learnability** useability principle.

# References

Docherty, K., Graham, J., & Russell, A. (2018). Nelson Digital Solutions for QCE units 104. <https://cengage.com.au/product/title/nelson-digital-solutions-for-qce-units-1-4-st/isbn/9780170420655>.

Freeplane team. (n.d.). Home—Freeplane documentation. Retrieved May 23, 2023, from <https://docs.freeplane.org/>.

GitHub. (n.d.). Getting started with GitHub Desktop. GitHub Docs. Retrieved May 23, 2023, from <https://ghdocs-prod.azurewebsites.net/en/desktop/installing-and-configuring-github-desktop/overview/getting-started-with-github-desktop>.

Lucidchart. (n.d.). Intelligent diagramming. Lucidchart. Retrieved May 23, 2023, from <https://www.lucidchart.com>.

Microsoft. (n.d.). Documentation for Visual Studio Code. Retrieved May 23, 2023, from <https://code.visualstudio.com/docs>.

Template.net. (n.d.). Microsoft Word - What is Microsoft Word? Definition, uses. Template.Net. Retrieved May 23, 2023, from <https://www.template.net/office/microsoft-word/>.

The Python Foundation. (n.d.). What is Python? Executive summary. Python.Org. Retrieved May 23, 2023, from <https://www.python.org/doc/essays/blurb/>.

Adobe. (n.d.). What is Adobe XD and what is it used for. What is Adobe XD used for? <https://www.adobe.com/products/xd/learn/get-started/what-is-adobe-xd-used-for.html>.

Limited, R. C. (n.d.). PyQt5: Python bindings for the Qt cross platform application toolkit. Retrieved June 1, 2023, from <https://www.riverbankcomputing.com/software/pyqt/>

|  |  |  |
| --- | --- | --- |
| Criteria | Successful? | Notes |
| The prototype must recognise and describe programming elements, user interface components and useability principles. | Yes. | Described in the mind map under data sources and under programming elements. |
| The report must symbolise and explain programming information & ideas, data structures and interrelationships between user experiences & data of the prototype. | Yes. | Described in the mind map under algorithm components and data structures. |
| The report must analyse the problem and information related to the technical proposal for a low-fidelity prototype digital solution. | Yes. | Described in the mind map under Problems and Solutions. |
| The report must determine user interface, data, programmed and solution requirements of the digital solution. | Yes. | Described in Prescribed and Self Determined Criteria. |
| The report must synthesise information and ideas to determine user interface and programmed components for a digital solution. | Yes. | Described in the mind map under algorithm components and data elements. |
| The user interfaces and programmed components of the digital solution must be generated. | Yes. | Found within the annotated code, provided MP4 of program and evidence of program running photos. |
| The report must evaluate impacts, components, and the digital solution against prescribed and self-determined criteria to make refinements and recommendations. | Yes. | Provided within this section. |
| The report must include decisions about and use mode-appropriate features, written language, and conventions for a technical audience. | Yes. | Described in the mind map under mode-appropriate features, Conventions and Language. |
| The application must maintain compatibility across major desktop operating systems, to allow for most compatibility. | Yes. | Provided prototype works with the same outcomes across operating systems. |
| For the application, reliability must be a priority, there can’t be any bugs which prevent major features from working. | Yes. | Provided prototype doesn’t have any breaking bugs that prevent it from being used properly. |
| The application must use python as it is a requirement, due to the use of PyQT5 and the need for the ability to produce prototypes quickly. | Yes. | Python was used to produce the prototype. |
| The application needs to make it clear to what they are expected to do, if the experience confuses the user, it impedes the program altogether. | Yes. | It’s clear within the program, indicators and hints are present. |

BEGIN

OUTPUT Home

INPUT current screen

IF current screen IS View Patient Details THEN

WHILE current screen is View Patient Details THEN

INPUT patient

OUTPUT patient details

ENDWHILE

ELSEIF current screen IS Add or Edit Patient Details THEN

WHILE current screen IS Add or Edit Patient Details THEN

INPUT patient

OUTPUT patient details

INPUT user values

SET selected patient TO modified patient from user

ENDWHILE

ELSEIF current screen IS View Tests Taken THEN

WHILE current screen IS View Tests Taken THEN

INPUT patient

OUTPUT patient tests

INPUT test taken

OUTPUT test information

ENDWHILE

ELSEIF current screen IS Add Taken Test THEN

WHILE current screen IS Add Taken Test

INPUT new test

INPUT patient

SET patient in tests taken TO appended test

ENDWHILE

ELSEIF current screen IS View Test Types THEN

WHILE current screen IS View Test Types THEN

INPUT test code

OUTPUT test code information

ENDWHILE

ELSEIF current screen IS Add or Edit Test Types

WHILE current screen IS Add or Edit Test Types THEN

INPUT test type

OUTPUT test information

INPUT user values

SET selected test TO modified test from user

ENDWHILE

ENDIF

END





