



# UoC Inventory System

10.04.2019

—

Time Sponge Software

# Contents

<b>Contents</b>	<b>1</b>
<b>Goal</b>	<b>2</b>
<b>Project Specifications</b>	<b>2</b>
Initial Project Plan	2
Costings	6
Risk Assessment	7
<b>Stretch Goals</b>	<b>8</b>
<b>Team Roles</b>	<b>9</b>
<b>Long Term Goals</b>	<b>9</b>
<b>Design Documentation</b>	<b>11</b>
Problem Identification	11
Personas	11
User Journey	13
Crazy Eights	15
Low Fidelity	18
High Fidelity	23
Invision Usable Prototypes	29
<b>Development Cycle</b>	<b>30</b>
<b>GitHub Link</b>	<b>30</b>

## Goal

1. Create an inventory system so our client can loan out University devices more effectively

## Project Specifications

### Initial Project Plan

In this section, we take into consideration our essential and stretch goals in order to efficiently plan our undertaking of the task at hand. The idea is that we will follow these structured steps under a timescale to ensure at least the minimal functionality is included in the final project. Furthermore, our previously identified stretch goals will be also integrated into this plan but only after our essential functionality is completed to ensure a working and functional product is made available to our client. The time scale will be stretched over the next few weeks and looks as follows:

Task	Allocated Time	Front/Back	Reason	Priority
Discuss mapping of the App.	30 mins	Front	In order to efficiently design our app, we must first identify those screens.	High
Discuss mapping of the Web App.	30 mins	Back	In order to efficiently design our app, we must first identify those screens in accordance to the briefs additional requirements for back-end functionality.	High
Crazy 8's designs of App screens	1 hour	Front	To establish new design ideas and different ways of going about the screen is designed.	Medium
User Experience assessment	1 hour	Front	To analyse where design flaws are in an attempt to iron out affordances and make the app ultimately easier to use.	Medium

Design High Fidelity prototype on InVision.	1hr 30mins	Front	To visualise and finalise the working project without functionality.	High
Design a database conforming of appendix 1.	1hr 30mins	Back	To create an initial local database to upload to an ODBC SQL Server.	High
Create a blank app	5 mins	Front	To initialize production of the app	High
Add Firebase Libraries	45 mins	Front	To utilise the barcode scanning libraries	High
Integrate with android hardware camera	2 Hours	Front	To create functionality with the android camera for combined use with barcode scanning.	High
Create methods for processing barcodes using the firebase ml kit libraries	3 Hours	Front	To be able to process barcodes scanned using the camera	High
Add functionality to pass camera preview to the scanning library	2 hours	Front	Connecting previous 3 tasks together to gain a result.	High
Add graphic overlay to the camera preview screen	1 hour	Front	To ensure users capture the entire barcode in the viewfinder	High
Temporarily add toast or log to pass scanned code to, to aid development	20 mins	Front	To check if the barcode is scanning in the correct manner	Medium
Display popup with the item name and a confirmation button	2 hours	Front	To double check that barcode is on the correct item and user scanned the right thing	High

Display terms & conditions confirmation prompt	1 hour	Front	To ensure users understand and accept the terms of their loan	High
Add a confirmation button to confirm loan	1 hour	Front	To ensure an essential criterion measure is employed here to display what is necessary to the user.	High
Design UI for the app from high fidelity design.	2-4 days	Front	Following the design style	High
Install SQL Server 2016 on the client provided windows server	2 hours	Back	To give us a place to host all of the backend data for the system	High
Upload previously designed database to SQL server	1 hour	Back	As per previous item, gives us a place to access client data from	High
Create API to handle requests from app	2-5 days	Back	Ensures security of database so app can't be decompiled and ran with custom sql commands	Medium
Create SMTP email sending service on windows server	2 days	Back	So the system can send out reminder emails	High
Replace SQL statements and connection in app with API calls	1 day	Front	As with previous API item, to ensure security of the database	Medium
Create front end design of the web app (html / css / js) using ElectronJS	3-5 days	Front	So admin's have a way to check on the system. Desktop app with electron for ease of use and access	High
Create functional js for all api calls	1 day	Front/Back	To allow the API to manipulate the database server dependant	Medium

			on the type of call.	
Add support for hardware barcode scanner in the web app	2 days	Front/Back	Provides a way for admins to add items to the database	Medium
Add web app functionality to check items due to be returned soon or haven't been returned	2 days	Front	Lets admins check items are being returned properly and chase down people who haven't returned item	Medium
Port android app to IOS	1 - 1.5 weeks	Front	Cross platform support for students with IPHones	Medium
Give project to client to test.	2hrs	Front/Back	In order to gain feedback on areas that must be change/adapted/added in the next phase of the agile process.	Medium

## Key Technologies Being Used

We are using a number of key technologies in the development of the system. The main technologies are as follows:

- Kotlin
- Invision.com
- Proto.io
- Java
- Android SDK
- Firebase ML Kit
- SQL
- Asp.NetCore
- Konan
- REST API model
- JSon
- SMTP
- HTML, CSS, Javascript
- ElectronJS
- JQuery
- Kuery

- IOS
- MacOS
- Windows/Windows Server
- Debian Linux
- Swift

## Costings

<u>Initial Costs</u>	<u>Cost</u>		<u>Running Costs (During Project)</u>	<u>Cost</u>
Iphone 6s	£159.00		Wages per person x4	£10,800.00
Mac Book	£1,130.00		Pensions per person x4	£3,100.80
Server	£1,402.80		Server Maintenance per month	£70.00
Laptops x4	£2,450.00		Electricity per year	£9.80
Sticker Printer	£139.00		Internet Service Provider for 6 weeks	£2.30
Sticker Paper	£3.50		Travel Expenses for 6 weeks	£180.00
Brain Storming Utensils	£100.00		Water bill for 6 weeks	£40.85
Barcode Scanner	£70.00		Gas and Heating bill for 6 weeks	£5.54
<b>Total Initial Costs:</b>	<b>£5,454.30</b>		Food per week x4	£140.00
			<b>Total Running Costs During Project:</b>	<b>£14,349.29</b>
	<b>Grand Total of Project:</b>	<b>£19,803.59</b>		
<u>Running Costs (After Project Completion)</u>	<u>Cost</u>			
Electricity per year	£85.00			
Internet Service Provider per year	£19.99			
Server Maintenance per year	£840.00			
<b>Total Running Costs After Completion:</b>	<b>£944.99</b>			

From looking at the above spreadsheet we can now work out how much we need allocated to the project budget as we have worked out the minimum expense requirements for the 6 week project. The minimum expense requirement for the six weeks is a grand total of £19,803.59, this figure does not include the continued maintenance and running cost in order to keep the system up. We have estimated that it will cost £944.99 per year in order to keep the system running.

The initial start up cost of the project has been estimated at £5,454.30. We need to buy an Apple Macbook Pro in order to be able to programme and deploy the system for any IOS platform devices, this has been priced at £1,130.00. Because of this we also are also require to have an iphone in order to do user testing on the app with an IOS phone. We have found a secondhand iphone 6s priced at £159.00. We also require smaller expenses like the a barcode scanner, sticker printer and sticker paper in order to print out QR codes and barcodes that we can then stick one on to each device. A big main cost is buying a server to host the system. With a 6 core 12 thread server, 8gb ram and 512gb sata ssd we found one costing £1,402.80.

## Risk Assessment

Issue	Assessment	Actions	Probability	Impact
Technology	Potential loss of devices from software bugs	Put time aside to test the software thoroughly with test devices in a real world scenario	Low	High
Project specification	Could deviate from the original specification	Try to stick to the specification as much as possible. If the group needs to deviate add patch notes to describe how and why the product is different from the brief	Medium	Low
	We could over or under-estimate costing for the project	Evaluate as many sources of cost as possible before making any firm business decisions	Low	Low
	Detailed specifications we made may turn out to not be the best approach to the project	Evaluate and iterate to find a better approach	Medium	Low
Resources	The client cannot supply adequate resources to allow us to complete advanced development tasks	Work our way around that limitation to still deliver a successful, working project by changing project goals	Low	High
Client	Failure to complete the base functionality	We will set weekly goals and move resources around the group and assign group members to work together to get work done on time	Low	High
Programme	If code is not commented, programmers external to our group may struggle to understand it's functionality	Comment on all code as it's produced so our group and future programmers using the project	Medium	Medium



Security	If SQL statements are used in the app itself someone could access these commands and direct confidential data somewhere else.	We will use an API as a medium between the app and the server to reduce the amount of confidential amount of functions the app completes.	N/A	High
External factors	Loss of internet could result in a major loss of development and documentation time	<p>We will use desktop based applications which do not require internet access and we will create multiple local backups to be pushed to github as soon as we get internet access again</p> <p>We will also use 4G mobile hotspots from our phones incase we need internet access as quickly as possible</p>	Medium	High

## Stretch Goals

After completing the project requirements that are laid out in the initial project plan we will move on to completing these stretch goals. These stretch goals are based on the extra desirable requirements that were included in the project brief as well as extra research our group has conducted.

### Goals

- To expand the program to allow users to see the return date in the app. This would require the creation of a custom API. This would be a useful feature to add later on in the project as students may often forget their return dates and adding the return date to the app they used to book it out will hopefully reduce the amount of devices that are returned late. This will reduce the burden placed on our client as they will not have to wait for a long time to get devices back.
- Create a waiting list for students which will email students when devices become available. This stretch goal will benefit the students that will eventually use our system as they will be automatically notified about vacant devices that they want to loan out.
- Add a feedback form to gather opinions and new development ideas from students and staff. This added function would give the client more feedback of more functions and changes students would like to see in the system.

- Expand the API to contain stock levels and allow students to pre book devices up to 24 hours before they collect the device. Pre-booking devices would benefit students as they would be able to reserve items ahead of time incase they desperately needed a device but were not free to book it out at that time. They could reserve a device and collect when they are next free.
- Add a fault report system incase a device breaks whilst in use. This system would allow the client to recognise and fix issues with devices much more quickly. This digital report system will also reduce the amount of time taken to identify problems with devices and fix them as the student can report it directly to the client.

## Team Roles

James Hughes - Programmer

James Marszalek - Programmer

Alex Maden - Communicator / Design and Documentation / Paired Programming

David Sloane - Design and Documentation / Paired Programming

## Long Term Goals

Integrate the system into the Chester University student app:

This would reduce the cognitive function required to loan out equipment as the student would only have one app to use throughout their time at university. Furthermore the design of the loaning function can be adjusted to match the current university app to give students a similar feel and user experience. User testing would be required to make sure that the design and layout of the new integration has a positive impact on usability and user experience.

This would also mean it would be easy to expand the loaning system to all the university campuses to expand the reach of the system. Less technology inclined students may benefit from having the system already integrated into the app they are already familiar with.

Add RFID tags to each device so when scanned by students they can quickly rent devices or find out if specific devices are already booked:

Adding this function to the system would allow the system to detect where devices are when they are placed in RFID equipped charging docks. These docks would charge all the devices and be able to detect which device is in which dock. This would translate to a system which could give users exact locations on the devices they require or incase a device is unavailable on their campus it would tell the user if there is another device free on another campus that they can use.



Allow all students access to parkgate campus devices and their own campus of study devices:

This functionality would be added due to the fact parkgate campus contains the most popular library which is used by students from all the Chester University campuses. This means giving every Chester University student access to the devices on parkgate campus. This should expand the reach and usefulness of the loan system.

Implement different language settings as Chester University is a multicultural campus with many different nationality students:

Due to the fact that Chester University contains students from many different countries adding different language support to the system will make it usable by all students at Chester University. Also many students from different countries like to talk and communicate in their first language so adding the ability to use the system in their own language will be very beneficial.

Combine the book and device loaning systems together on the Chester Uni app:

Adding support in the loaning system for library books would combine all the University owned loanable items into one system, therefore streamlining the system.

RFID scanners in charging bays which will automatically sign the laptop back in and remove the loan from said students account:

Adding this automatic function to the system would reduce the workload placed on students and allow them to simply return items. Also this fits the busy working day that students have at university.

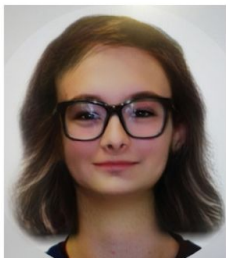
# Design Documentation

## Problem Identification

After our client outlined what he wanted the system to do we tasked ourselves with finding the best software solution to deliver to him. We decided to create a Web Application which can be used in a web browser but also on the desktop. This would be used directly by our client. Also we are creating an Android and IOS application. This would be used by students at the university. We then tasked ourselves with designing these two software systems incorporating as much User Experience and Interface Design theory as possible.

## Personas

### Pauline Overhill



#### Demographics

Age: 27  
Gender: Female  
Occupation: Systems Development Manager  
Status: Single  
Salary: £45,000  
Location: Chester England

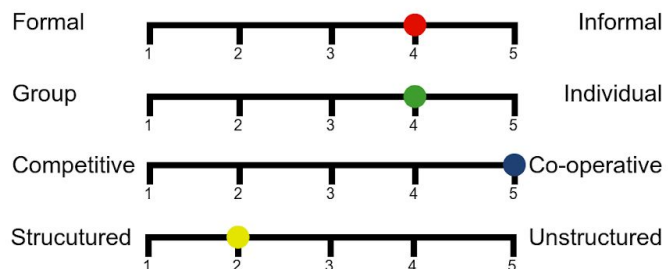
#### Responsibilities

Plan, organize, direct, control and evaluate the operations of projects  
  
Develop and implement policies and procedures for electronic data processing and computer systems operations  
  
Meet with clients to discuss system requirements, specifications, costs and timelines

#### Goals

She is goal-orientated within a strong leadership role. Her main concern is being able to keep track of all the loanable devices within her department. She is currently only using an excel spreadsheet to keep track of this, which she says is inefficient and ineffective.

She is looking for a more reliable method of loaning out the devices within her department and needs to be able to keep track of all return dates as well as sending out emails when devices are overdue.



## Len Sunner



### Demographics

Age: 30  
 Gender: Male  
 Occupation: Thornton  
 Science Park Student  
 Status: Single  
 Salary: N/A  
 Location: Chester England

### Pain Points

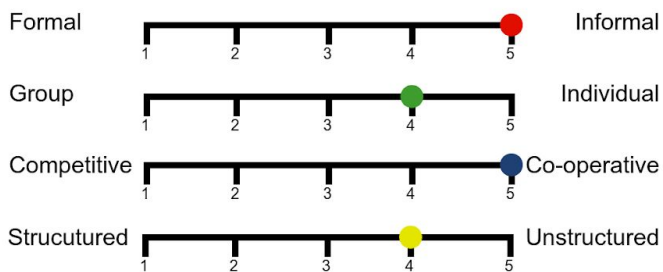
Having to wait for Pauline to get back to her office in order to be able to loan out a device

Not remembering when the return by date is for the device and accidentally going over thus incurring a fine.

Not knowing what devices are available or when a device you want to loan becomes available/not being able to reserve a device for when it becomes available

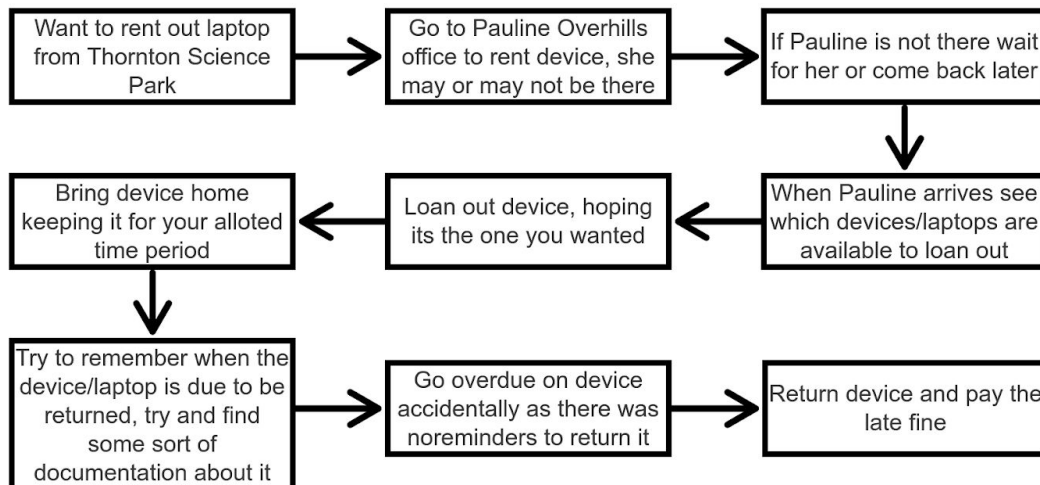
### Bio

Is a mature student at the university of Chester. His main campus of study is Thornton Science Park. Because he is a fulltime mature student most of his loan goes towards his rent and living costs, because of this he does not have enough money to justify spending it on a new laptop that he requires for his course. In order to fix/deal with this problem Len loans one out from Pauline Overhill at Thornton Science Park. Although he finds the loaning methods ineffective as he never knows/remembers the return date of said device.



## User Journey

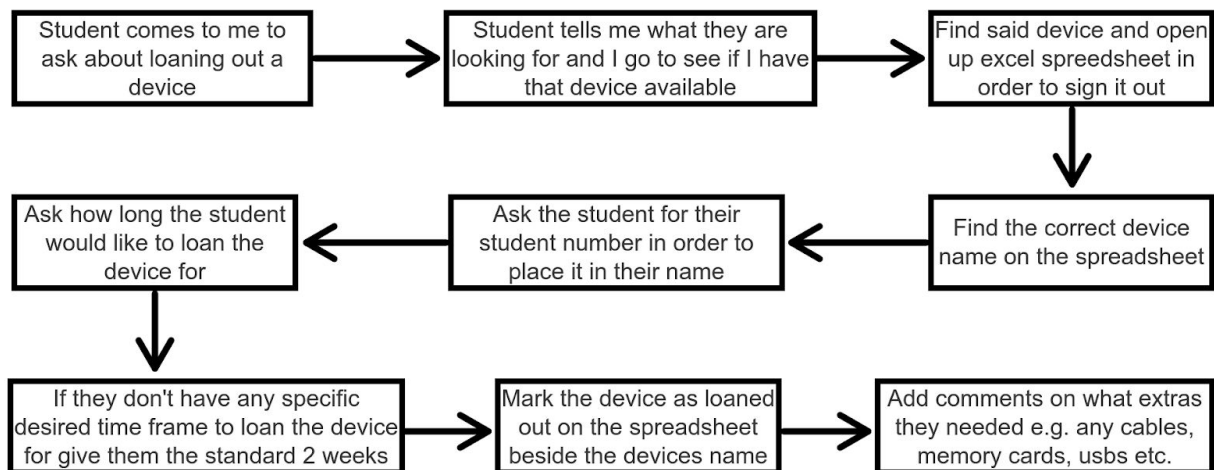
### User Journey - Len Sunner



#### Problems with Len Sunner's Current User Journey

- Pauline Overhill may or may not be in her office and if she's not then Len either has to wait for her to get back or come back later. If Pauline however isn't that day for some reason then no students can loan out devices.
- It's hard to remember when the device is due back as you are just given your due date and it is up to Len/the student to remember to bring it back. If he goes past the due date then he may incur late fines.
- Trying to remember to bring back any extra cables, usbs, memory cards etc. that he has loaned out with the device.
- Very rarely but sometimes Pauline Overhill may not have the device that Len Sunner or a student is looking for meaning that the student will then have to wait for it to arrive if Pauline orders one and if she doesn't order one then the student will have to find some sort of alternative.

## User Journey - Pauline Overhill

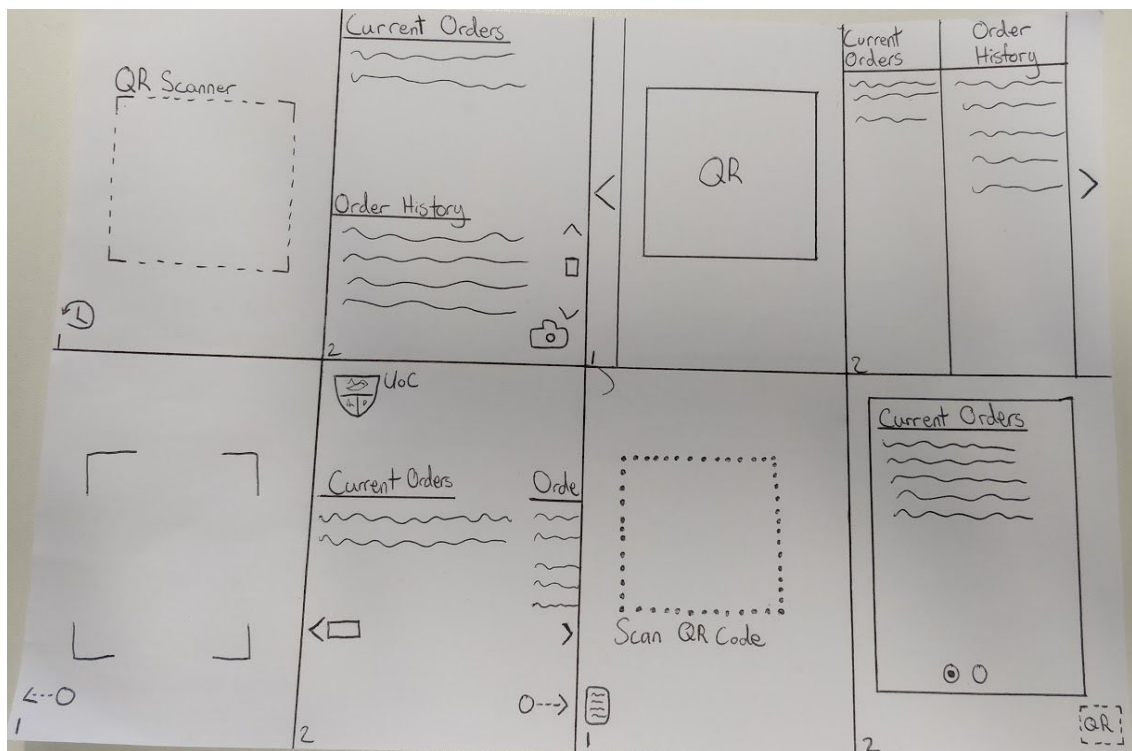


### Main Problems of Pauline Overhills Current User Journey

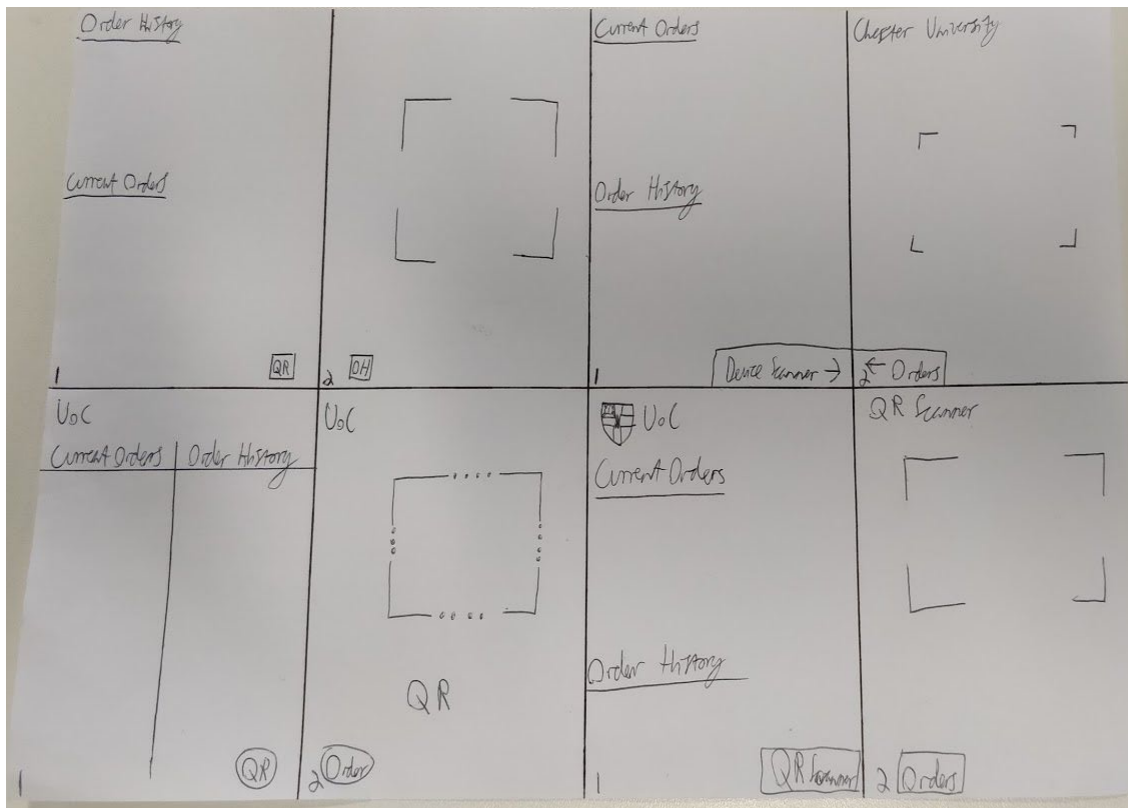
- Devices don't always come back on time and Pauline is too busy to sent out overdue emails.
- If Pauline is not in her office then students either have to wait for her to get back or cannot get the device altogether
- Only Pauline updates and edits the spreadsheet meaning it may get done late if the device is left back with the wrong person, this can potentially leave the spreadsheet redundant and out of date.
- Students are prone to forgetting to bring back extras such as cables, usbs, memory cards etc.
- Very rarely but sometimes Pauline does not have what the student is looking to loan out so she may have to order a brand new device in order to solve this problem.

## Crazy Eights

The first design technique our two designers used was crazy eights design sprints. We used this technique to create two separate pages of early prototypes for the student app. After completing the design sprint we decided on using elements of each others designs to produce the low fidelity prototypes.







Next we completed a crazy eight design sprint for the Web Application. Due to the more limited structure that this type of software imposed on our designers we decided to combine resources and both work on one page of designs. Below are the designs we produced. Firstly we designed four prototype screens for the devices screen. This screen will show all the current devices owned by the University and it will allow our client to add new devices, loan out devices and edit the current devices. Next we designed the loan screen which will show all the devices that are currently loaned out along with their details, the student number of the student who has them and the dates that the device was loaned and when they are due back in. These aspects are all key pieces of functionality that our client requires.

UoC Inventory		Devices	Loans	Devices Page
Add Device +				
~~~~~	~~~~~	~~~~~	Loan	Edit
~~~~~	~~~~~	~~~~~	Loan	Edit
~~~~~	~~~~~	~~~~~	Loan	Edit

UoC Inventory		Devices	Loans	Devices Page
1	~~~~~	~~~~~	~~~~~	
2	~~~~~	~~~~~	~~~~~	
3	~~~~~	~~~~~	~~~~~	
4	~~~~~	~~~~~	~~~~~	
Loan				Edit
				+Add Device

UoC Inventory		Devices	Loans	Devices Page
Edit	~~~~~	~~~~~	~~~~~	Loan
Edit	~~~~~	~~~~~	~~~~~	Loan
Edit	~~~~~	~~~~~	~~~~~	Loan

UoC Inventory		Devices	Loans	Devices Page
				+Add Device
~~~~~	~~~~~	~~~~~	~~~~~	Loan
~~~~~	~~~~~	~~~~~	~~~~~	Loan

UoC Inventory		Devices	Loans	Loans Page
~~~~~	~~~~~	~~~~~	~~~~~	Return
~~~~~	~~~~~	~~~~~	~~~~~	Return
~~~~~	~~~~~	~~~~~	~~~~~	Return

UoC Inventory		Devices	Loans	Loans Page
1	~~~~~	~~~~~	~~~~~	
2	~~~~~	~~~~~	~~~~~	
3	~~~~~	~~~~~	~~~~~	
4	~~~~~	~~~~~	~~~~~	

UoC Inventory		Devices	Loans	Loans Page
Return	~~~~~	~~~~~	~~~~~	
Return	~~~~~	~~~~~	~~~~~	
Return	~~~~~	~~~~~	~~~~~	

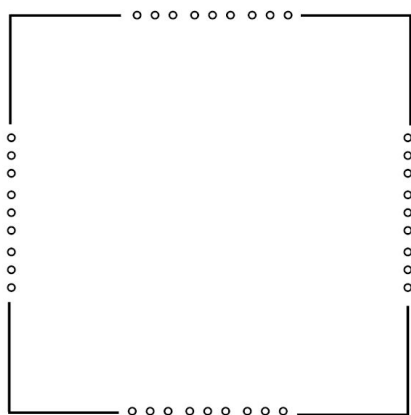
  

UoC Inventory		Devices	Loans	Loans Page
Devices	1	~~~~~	~~~~~	
Loans	2	~~~~~	~~~~~	
	3	~~~~~	~~~~~	
	4	~~~~~	~~~~~	
				Return

## Low Fidelity

Next our designers developed their own individual ideas from the crazy eights design sprints they conducted. Each designer made 2 prototype designs for the QR scanning screen and the current loans and loan history screen. These prototypes are shown below.

### UoC



### QR

Loans

### UoC

Current Loans

Loan History

QR

QR Scanner

UoC  
Crest

UoC

Current Loans



Loan History



Loans

QR





Scan QR Code

## Current Loans

01 \_\_\_\_\_  
02 \_\_\_\_\_  
03 \_\_\_\_\_  
04 \_\_\_\_\_





## Current Loans

01 \_\_\_\_\_  
02 \_\_\_\_\_  
03 \_\_\_\_\_

QR Scanner



## Loan History

01 \_\_\_\_\_ ^  
02 \_\_\_\_\_  
03 \_\_\_\_\_  
04 \_\_\_\_\_  
05 \_\_\_\_\_  
06 \_\_\_\_\_  
07 \_\_\_\_\_  
08 \_\_\_\_\_ v





## UoC Logo and green background

Devices Tab Loans Button

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_

Loan Button Edit Button

Add Device Button

## UoC Logo and green background

Devices Tab Loans Tab

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_

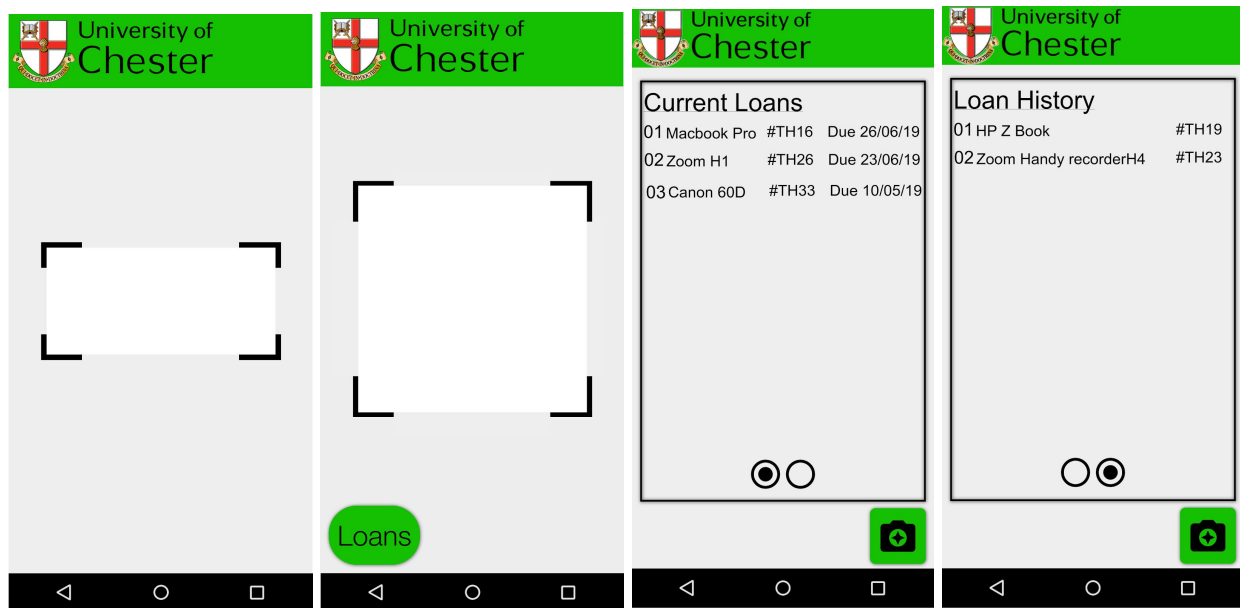
Return

Return

Return

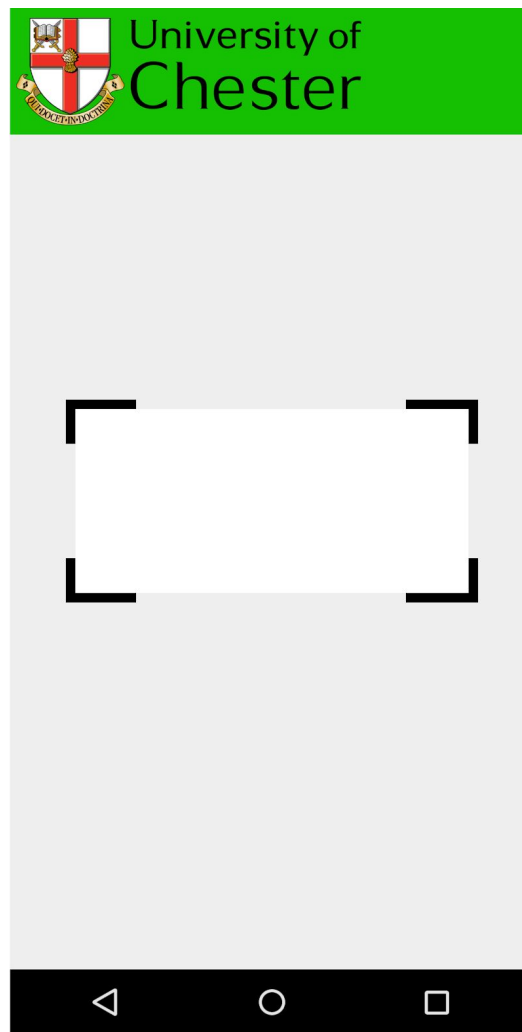
## High Fidelity

Next we combined our low fidelity design ideas together utilising User Experience design theory to create our high fidelity prototypes. Firstly we created the student app high fidelity prototypes. These are based around the colour scheme of green and grey outlined as a requirement by our client. The University of Chester logo features prominently in the top banner of the app.

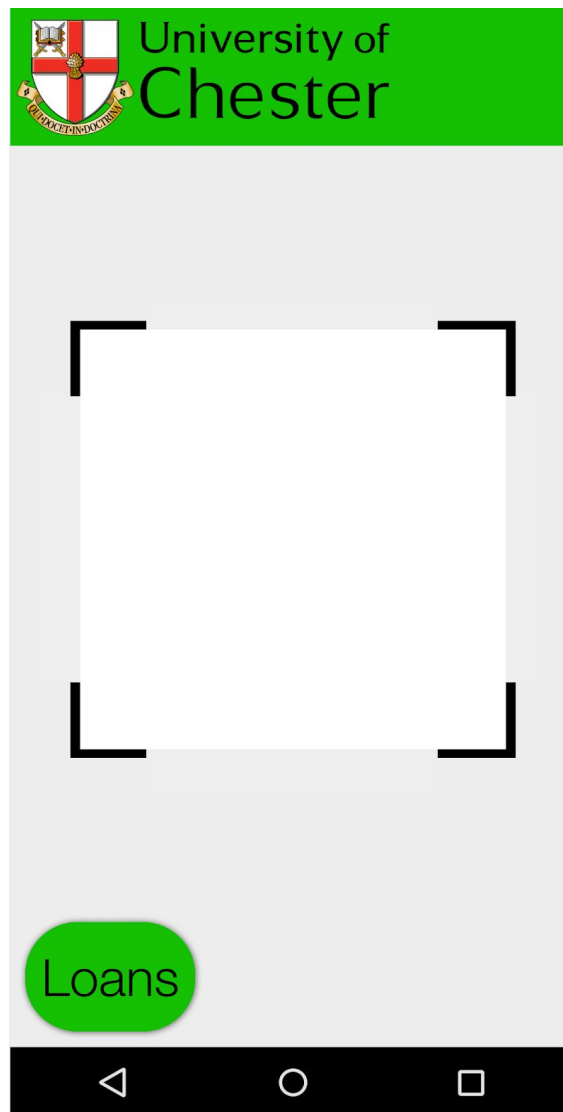




As the app opens a loading screen will display which will have a University of Chester approved image on it. This is to initialise the camera. Next the first screen we designed will show. This screen is to automatically scan the students barcode on their student ID. This screen and the QR reader screen have right angled edges around the camera square. This is common practice for camera apps which have a specific focus area such as QR scanner apps. Also the first two screens feature blurred out backgrounds as indicated by the grey background. Furthermore we don't need a button to take an image on either of our camera screens as our app will automatically scan the barcode or qr code when it is placed inside the box. Once the student has scanned their ID barcode the app will log them in and the app will take them to the QR scanning screen.




On this screen another camera square is used along with the blurred background. Once the user aligns a QR code of a device they wish to loan out the app will automatically scan it and pop up with a terms and conditions screen which the user must accept before the loan can start. This device will then be added to their current loan screen with the information associated with the loan. A prominent green button is located in the bottom left of the screen. This button is designed using the pill shape which is part of the modern material design language. This button takes you to the students loan screen. This screen contains their current loans and their loan history.



On this screen there is a card system for showing the student their current loans and loan history. The user can either swipe on the card section or press the two round buttons at the bottom of the card. These buttons are also used to show the user which page they are on and which way they can swipe or press to get to the next card. In the bottom right of the screen there is a camera button. The rounded edges further extend the use and aesthetic of the material design language. This button indicates to the user the location of the adjacent screen and which screen will come up when they click it.




Next we created the high fidelity prototypes for the Web App which our client will use directly. We created two main screens for our client to interact with, the devices and loans screen. On the devices screen our client will be able to add devices, loan devices out to students through the web app and edit the devices.


University of  
Chester

Devices
Loans
Add Device +

Name	ID	Serial Number	Mid Number	Loaned out (Yes/No)
MacBook Pro	#TH16	0284749210a	3fehbb	Yes
Dell XPS 13	#TH18	3954584nd22	48ndj22	No
Canon 60D SLR	#TH33	4839209ap11	4143ffd	No
HP Z Book	#TH19	48392hjds11	2917djs	Yes

Loan
Edit


University of  
Chester

Devices
Loans
Add Device +

Name	ID	Serial Number	Mid Number	Loaned out (Yes/No)
MacBook Pro	#TH16	Name:		<input checked="" type="checkbox"/> Yes
Dell XPS 13	#TH18			No
Canon 60D SLR	#TH33	Device ID:		No
HP Z Book	#TH19			Yes

Loan
Edit

Serial Number:  
Mid Number:  
Add

Devices

Loans

Add Device +

Name	ID	Serial Number	Mid Number	Loaned out (Yes/No)
MacBook Pro	#TH16	Device Name:		<input checked="" type="checkbox"/> Yes
Dell XPS 13	#TH18			No
Canon 60D SLR	#TH33	Device ID:		No
HP Z Book	#TH19			Yes

Student ID:

Loan

Loan

Edit

Devices

Loans

Add Device +

Name	ID	Serial Number	Mid Number	Loaned out (Yes/No)
MacBook Pro	#TH16	Select the device you want to edit		<input checked="" type="checkbox"/> Yes
Dell XPS 13	#TH18	Name:		No
Canon 60D SLR	#TH33			No
HP Z Book	#TH19	Device ID:		Yes

Serial Number:


Mid Number:

Add Updates

Loan

Edit

On the loan screen our client will be able to see all the current loans and they can return items.



University of

Chester

Devices

Loans

Name	ID	Loan Date	Due Date	Student ID	
MacBook Pro	#TH16	2/5/19	16/5/19	1703829	Return
Dell XPS 13	#TH18	5/5/19	19/5/19	1732819	Return
Canon 60D SLR	#TH33	7/5/19	21/5/19	1749110	Return
HP Z Book	#TH19	9/5/19	23/5/19	1602983	Return

## Invision Usable Prototypes

Below is a link to the onedrive folder containing the working web based prototypes for the App and the web application.

[https://outlooklivechesterac-my.sharepoint.com/:f:/g/personal/1704985\\_chester\\_ac\\_uk/EswBQ7I37vBHksCHUGBI3mkB11nsoMnZnCzosS-hr0II\\_A?e=oHyWx1](https://outlooklivechesterac-my.sharepoint.com/:f:/g/personal/1704985_chester_ac_uk/EswBQ7I37vBHksCHUGBI3mkB11nsoMnZnCzosS-hr0II_A?e=oHyWx1)

## Development Cycle

### Week 1:

Started to implement the Android app backend functionality in Android Studio

Creating barcode scanner using Android Studio

Mocked up a prototype database using Microsoft Access

Created QR codes for all the devices using QR Generator

Set up the SMTP relay using server manager on a Windows remote desktop

Created all the initial project documentation

### Week 2:

Created the working SQL Server

Set up the SMTP server

Conducted design sprints to create the first prototype designs for the App and Web application

Created low and high fidelity prototypes for the App and Web application on proto.io

### Week 3:

Created the Front end UI for the Android app

Created interactive prototypes of the app and web app on invision.com

Added more detail to the project file

Created the Inventory API in Visual Studio using Asp.NetCore

Started creating the HTML, CSS and Javascript for the Web application using Atom

Completed the basic functionality for the web app

Completed User Journeys for the current inventory system

## GitHub Link

<https://github.com/james1997hughes/UoC-CompSci-Inventory>