**ISAD251 Report**

APEX811: Cody Robinson, Reece Davies,

Jaime Kershaw Brown, Erin Clifton

**Plymouth University**

**Nov 2018**

Contents

[Introduction 3](#_Toc531776258)

[1. Scenario 3](#_Toc531776259)

[2. Background information 3](#_Toc531776260)

[3. Assumptions made 3](#_Toc531776261)

[4. Functional requirements 4](#_Toc531776262)

[5. Project plan 5](#_Toc531776263)

[Development 6](#_Toc531776264)

[1. Method of Approach 6](#_Toc531776265)

[2. Entity Relationship Diagram 7](#_Toc531776266)

[3. Student App Development Screenshots 8](#_Toc531776267)

[4. Admin App Development Screenshots 10](#_Toc531776268)

[Appendix 11](#_Toc531776269)

[A) SQL Scripts 11](#_Toc531776270)

# Introduction

## Scenario

There is a need for a university system that allows stage-two computing students to apply for a placement year. This application will be tracking the student’s activity throughout the application process, from the start of stage-two until they have secured a placement. It will not focus solely on students, but also be capable of managing companies and the jobs available to students. Students will be capable of applying through a mobile app, whilst the university will be able to manage students and companies through the admin desktop app.

## Background information

Relating specifically to the requirements of the system, a student will be required to input their information in order to use the system. This includes: student ID, programme undertaking, full name, email address, mobile and landline telephone number, date of birth and home address. Furthermore, specific values must not be null and in the format that best suites the content. In order for a student to apply for a job, they are required to submit a CV and had it approved. Once this has been accomplished, they will be deemed ‘active’ and can start applying for jobs.

A company is capable of having one or more sites; the jobs provided by a company will be related to a specific site the company owns, and thus will be the location for the job. This includes the address and postcode. In order for the student to apply for a job, they must be informed about the specifics of the job, and therefore the job must include: a job title, a short description outlining the job, contact information for enquiries, the salary of the job, the start date for the placement, a closing date for applications and an application method. Specific jobs will also require information relating to the student’s application history, which will outline the dates and times of the student’s status throughout their previous applications.

## Assumptions made

* Each site will have its own independent email address and telephone number for students to contact enquiring about the job available.
* Each site can have multiple jobs.
* Each job can have multiple vacancies available.
* Each company should have at least one site.
* There is no limit for how many sites a company can have.
* Admin will be responsible for creating the students’ accounts.

## Functional requirements

**Admin Desktop Application**

**Students**

- Add a student to the database

- View a student’s information

- View lists of both active & inactive students

- View a list of active students who have not yet accepted an offer

- View a list of students who have accepted a placement offer

- View a report of a student’s application(s) status

- View a pie chart showing the proportion of placed to unplaced students

- View a stacked bar chart showing the number of active students placed & unplaced by programme

- View a calendar of job vacancy closing dates

- Amend student information

**Companies**

- Add a company to the database

- View a company’s information

- Update a company’s information

- Archive a company including the site information.

**Job Vacancies**

- Add a job vacancy

- View an individual job

- View a list of jobs (current & past job vacancies)

- View a list of jobs that close within the next 7 days

- Update a given job vacancy

- Archive a job vacancy

**Student Mobile Application**

**Jobs**

- View a list of current jobs (current & past job vacancies)

**Applications**

- Record an application made

- Add a new status of application

- View a report on the status of all applications and each application made

**Students**

- View a student’s information including

- Amend student information

## Project plan

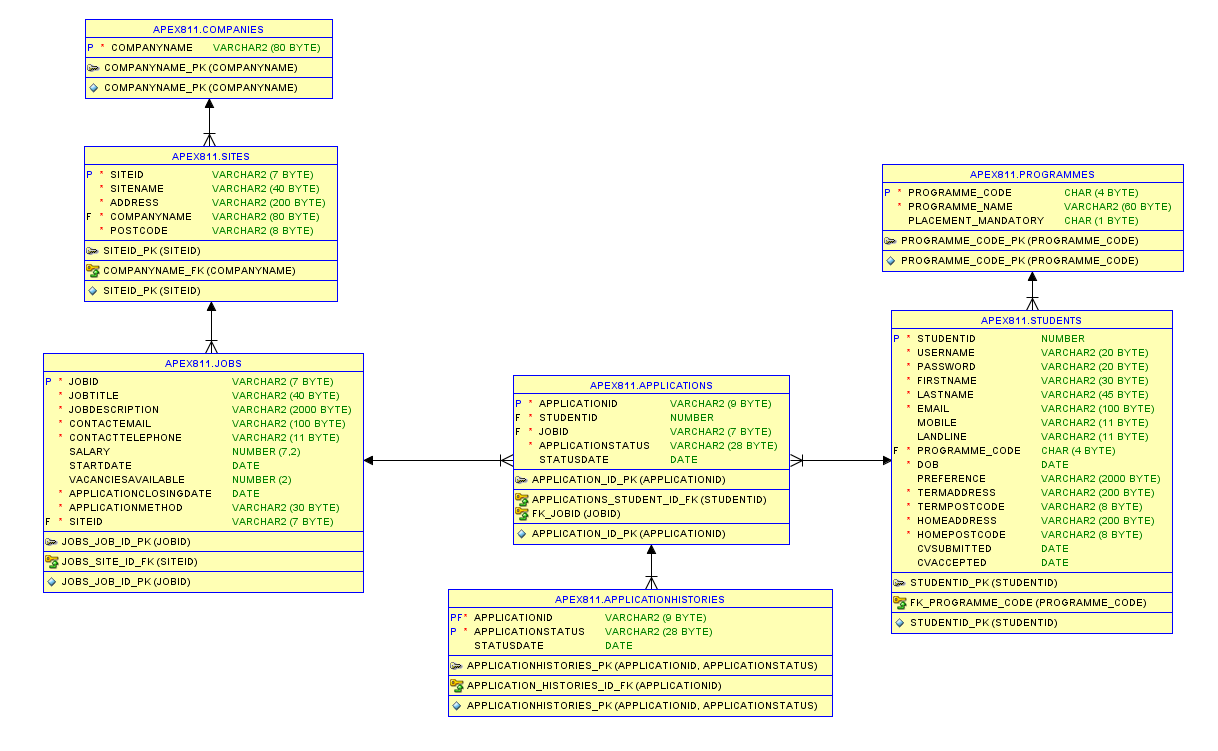
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Activity | Expected Start Date | Estimate Hours (Per Person) | Actual Start date | Actual Hours spent (Per Person) |
| Read through of Assignment brief, and discussion. | 22/10/2018 | 2 | 18/10/2018 | 2 |
| Create Project Plan for assignment. | 24/10/2018 | 5 | 31/10/2018 | 4 |
| Listed all the functional requirements. | 26/10/2018 | 2 | 23/10/2018 | 5 |
| List all attributes. | 29/10/2018 | 1 | 19/10/2018 | 1 |
| Normalise attributes into third normal form, alongside with declaring their table name and their data types. | 31/10/2018 | 6 | 19/10/2018 | 8 |
| Create relevant ERD for normalised data. | 06/11/2018 | 4 | 22/10/2018 | 3 |
| Design storyboard for front-end view of system. | 09/11/2018 | 3 | 31/10/2018 | 6 |
| Create SQL scripts to generate all tables in Oracle, alongside with constraints to relevant primary, composite and foreign keys. | 14/11/2018 | 5 | 29/10/2018 | 7 |
| Popularised the database with relevant information. | 23/11/2018 | 4 | 26/11/2018 | 4 |
| Create graphical user interface for front-view. | 27/11/2018 | 9 | 27/11/2018 | 15 |
| Create Report to review Assignment. | 07/12/2018 | 9 | TBC | TBC |

# Development

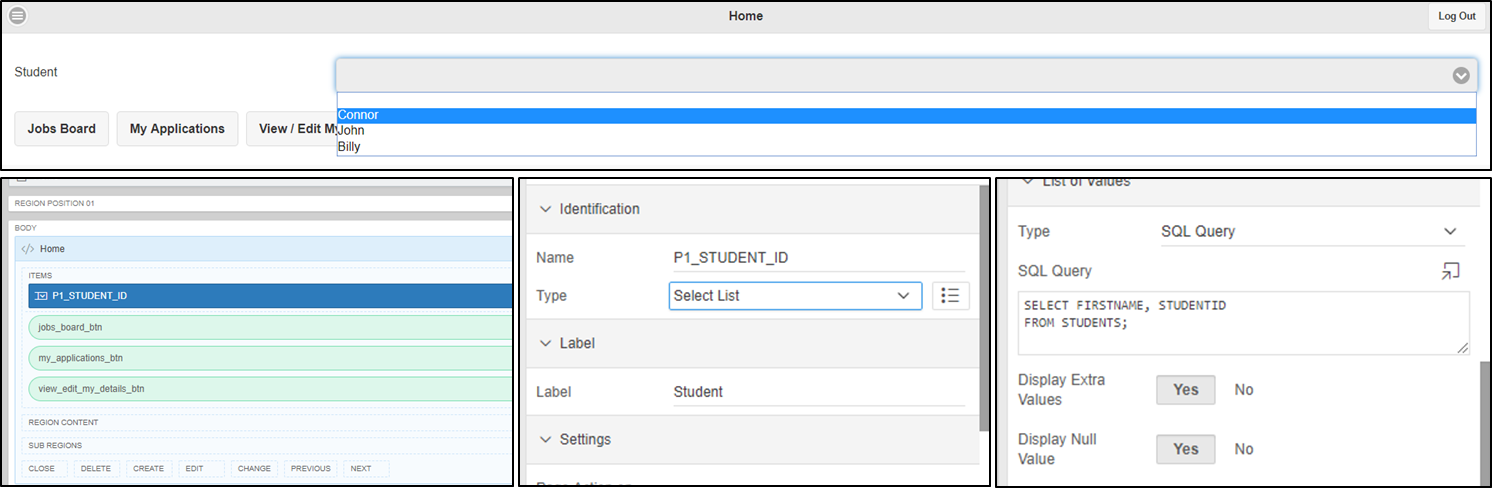
## Method of Approach

Once we had our four members for the group, we arranged a meeting in which we read through the assignment brief and discussed which tasks should be performed first with the best method to do so. Following this we started naming all the attributes that the system might contain, including the ones stated in the assignment brief. This may have not been the best approach as we did not initially create a project plan prior to starting the coursework. As a result, we were required to spend more time planning on what actions were left to do. This is because we were unsure of which tasks were complete, semi-complete or incomplete. Once the project plan was created, it acted as a foundation of what task must be worked upon, when it must be started, and the time spent on that specific task. Of course, certain tasks had complications and thus the actual amount of hours spent on it exceeded the estimate amount of hours. This was particularly problematic when developing the graphical user interface with APEX as we were unfamiliar with the software.

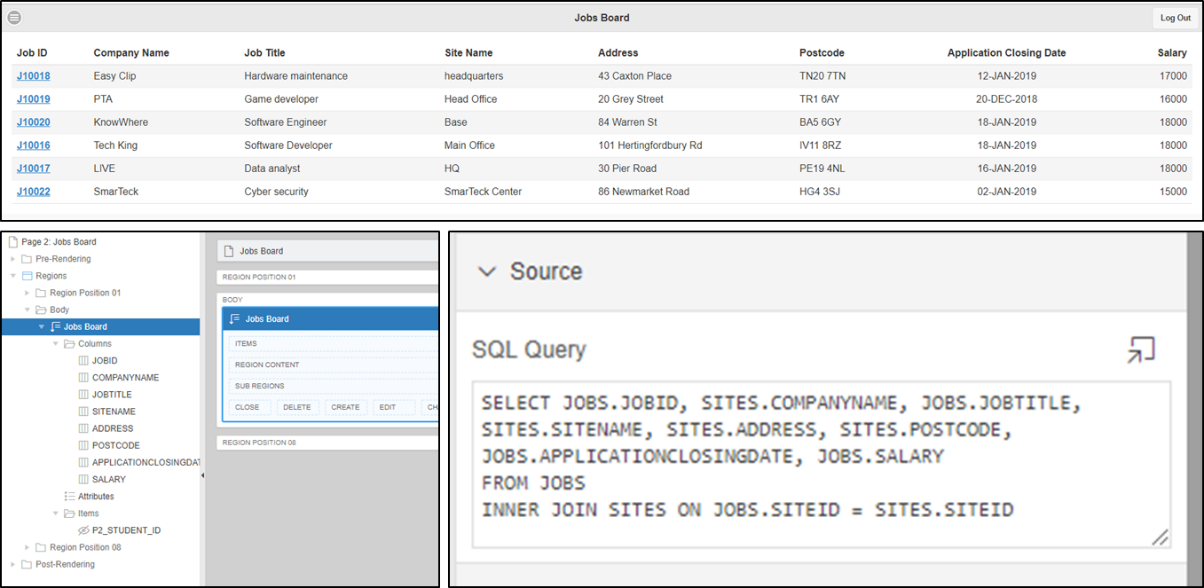
## Entity Relationship Diagram

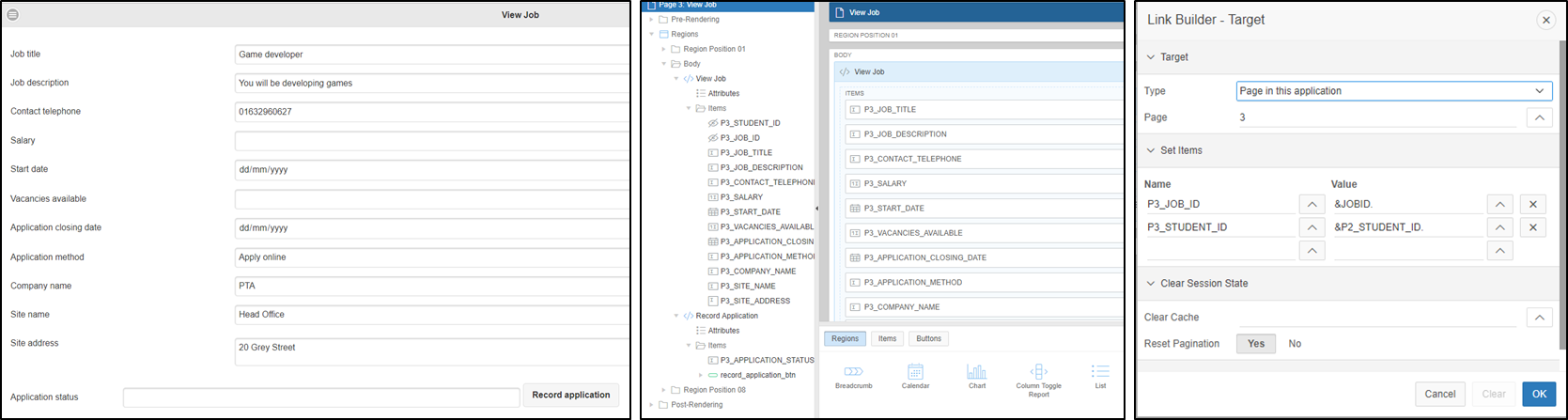


## Student App Development Screenshots



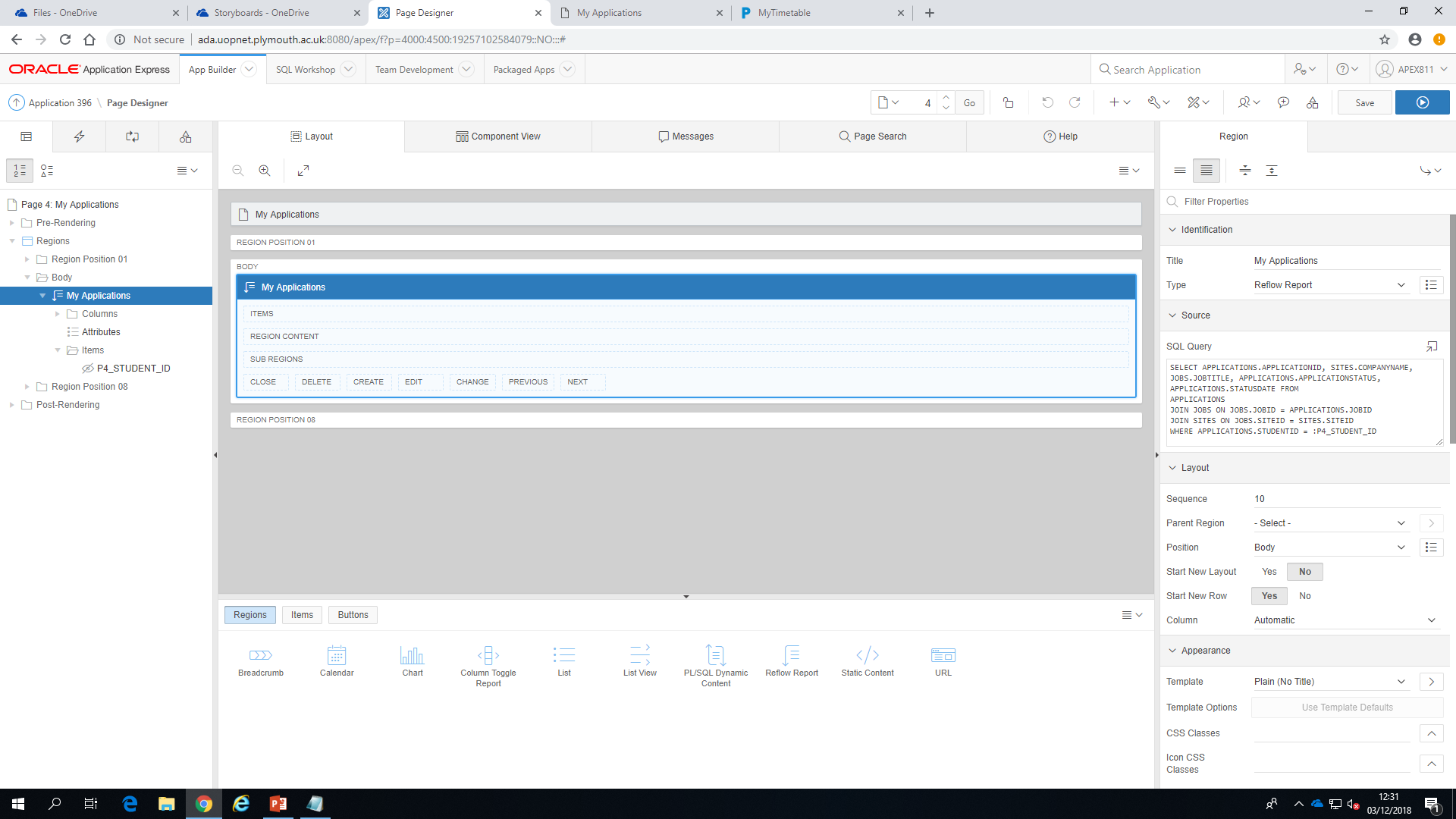
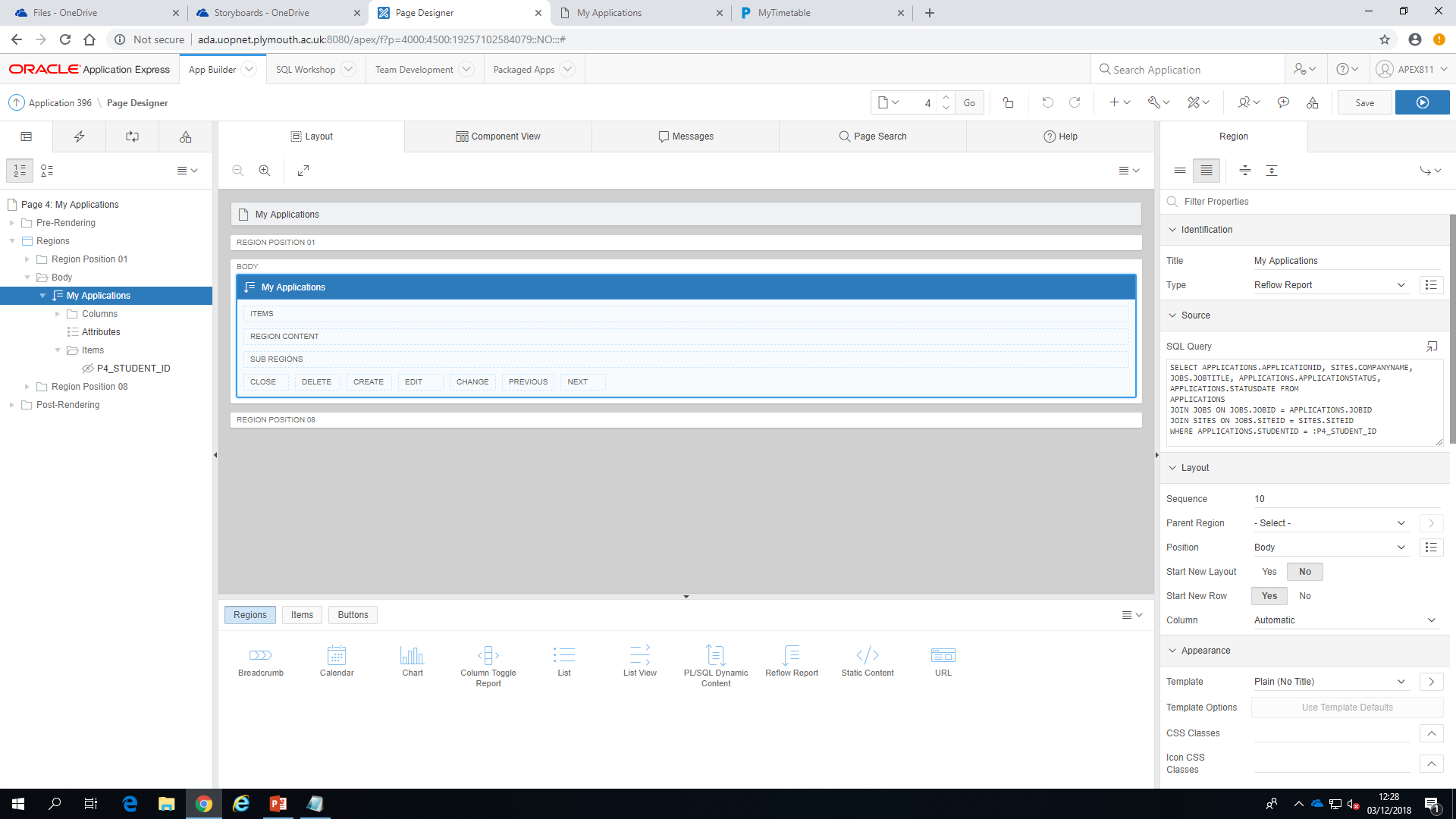
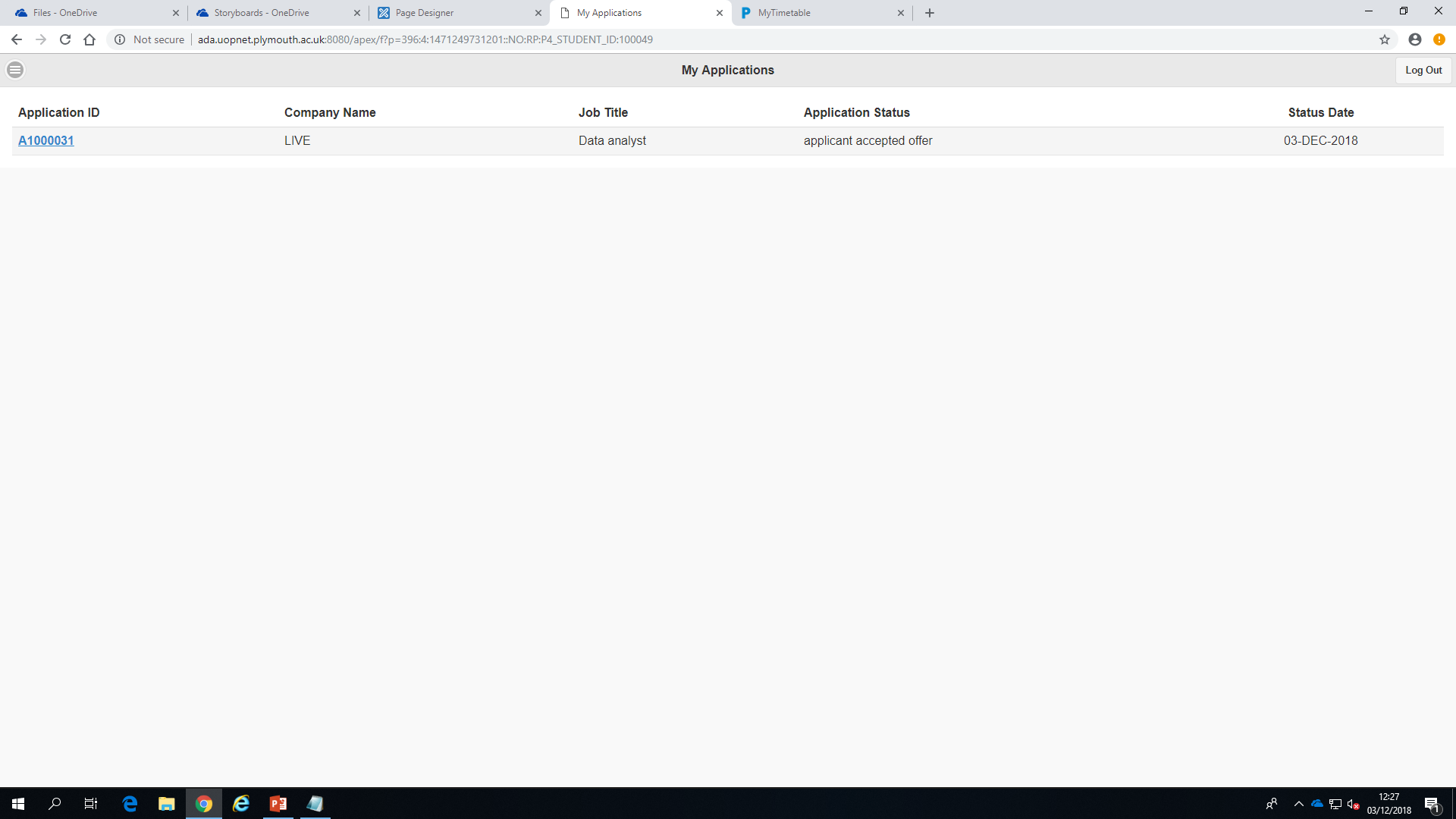
‘Home’ displays navigation buttons to take the user to other pages, which includes ‘Jobs Boards’, ‘My Applications’ and ‘View/Edit My Details’. Also contains a Select List that displays all the students’ names within the drop-down menu.

‘Jobs Board’ contains a reflow report which uses a SQL query to display all relevant information of a job in a table. The user can click on the ‘Job ID’ to be taken to a separate page showing all information specific to the job. Useful for information that does not fit into the table, such as the job description.

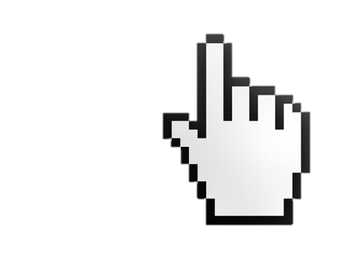
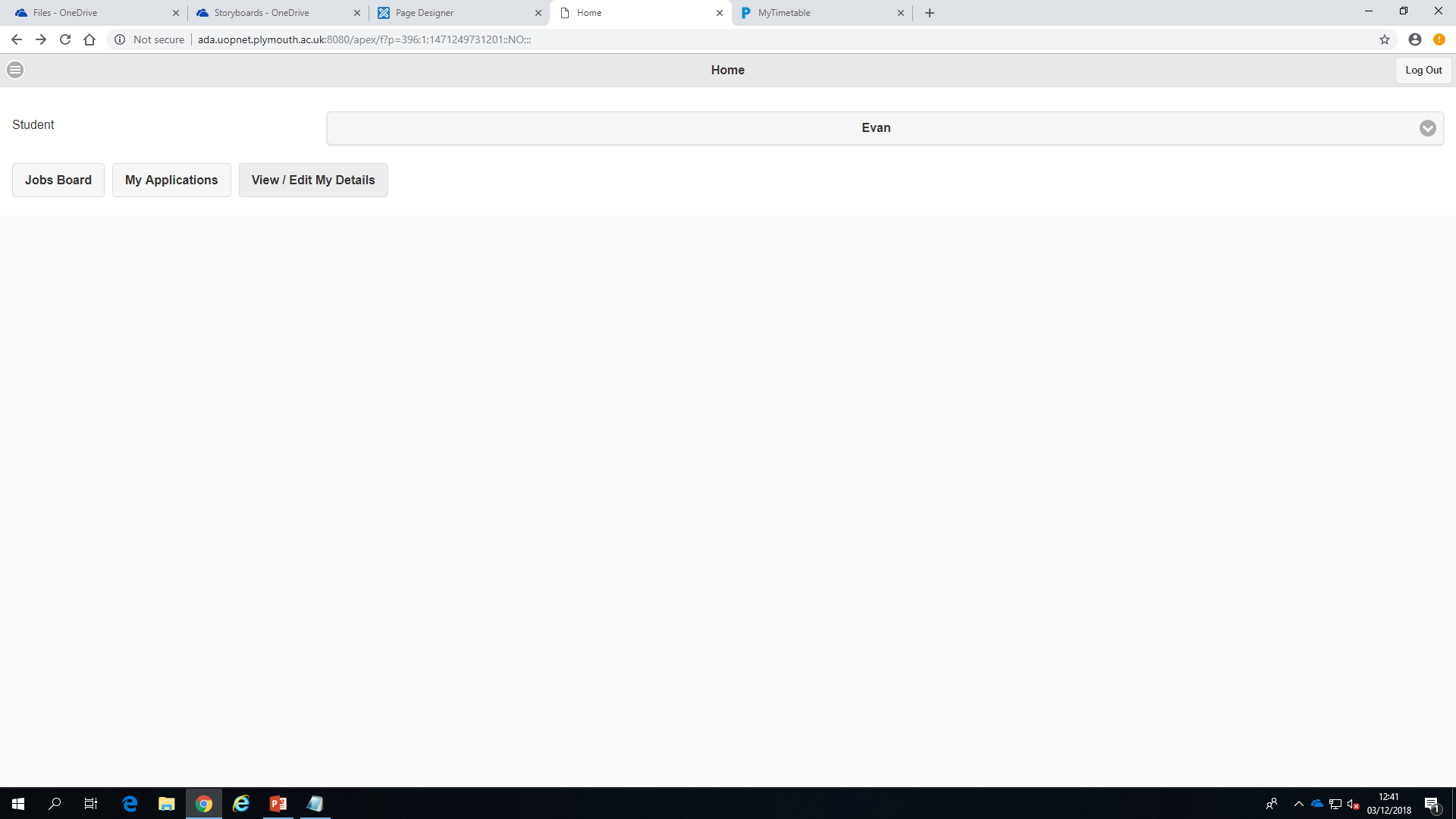
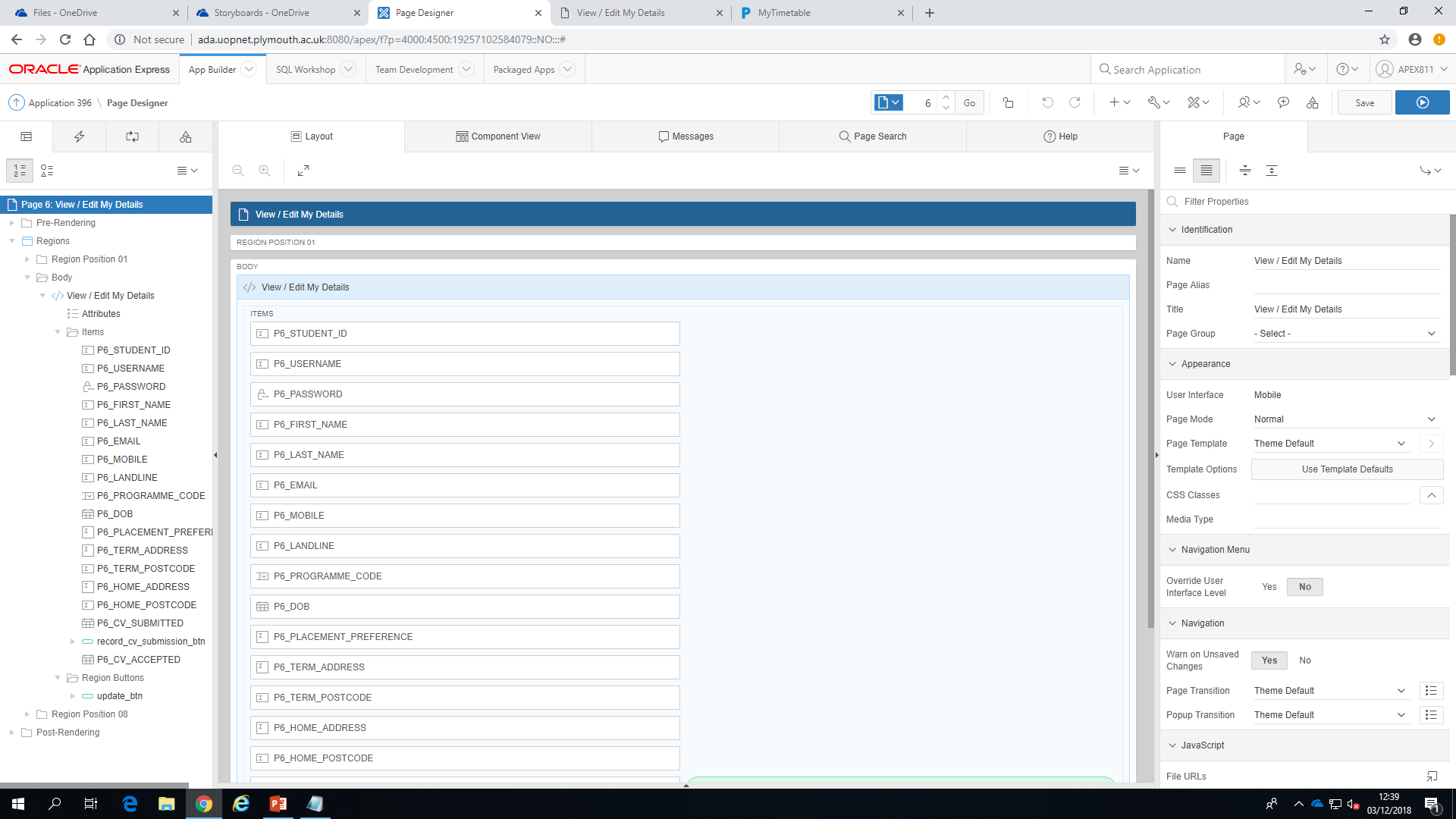
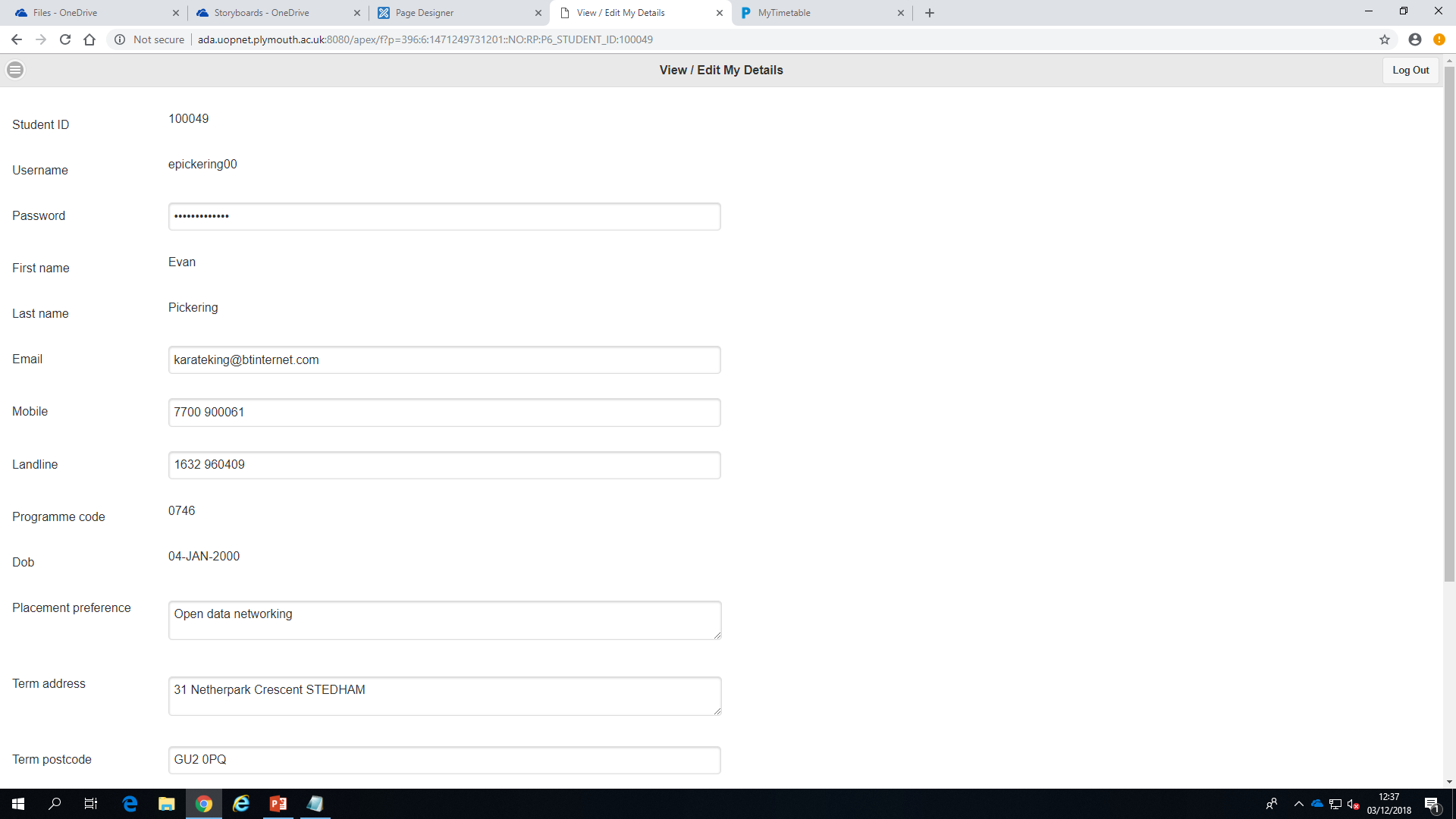


‘View Job’ contains Static Content with columns relevant to the Job’s details. In order to transfer the information across from ‘Jobs Board’ to ‘View Job’, the settings of the column had to be ‘Link’, which will navigate the user to the ‘View Job’ page. The text fields must be automatically filled with the appropriate information when this happens, and therefore we were required to use the ‘Set Name’ property.

‘My Applications’ is very similar to the ‘Jobs Board’ page, in which it contains a reflow report which extracts specific information from the database and structures it in a table. The relevant headings are ‘Application ID’, ‘Company Name’, ‘Job Title’, ‘Application Status’ and ‘Status Date’.



‘View/Edit My Details’ displays the user’s details; this is done by using similar methods for ‘View Job’ page.



## Admin App Development Screenshots

# Appendix

## SQL Scripts

CREATE TABLE "APEX811"."COMPANIES"

( "COMPANYNAME" VARCHAR2(80 BYTE) CONSTRAINT "COMPANYNAME\_NN" NOT NULL ENABLE,

CONSTRAINT "COMPANYNAME\_PK" PRIMARY KEY ("COMPANYNAME")

);

CREATE TABLE "APEX811"."SITES"

( "SITEID" VARCHAR2(7 BYTE) DEFAULT ON NULL 'S'||LTRIM(TO\_CHAR("APEX811"."SEQ\_SITE\_ID"."NEXTVAL",'09999')) NOT NULL ENABLE,

"SITENAME" VARCHAR2(40 BYTE) CONSTRAINT "SITENAME\_NN" NOT NULL ENABLE,

"ADDRESS" VARCHAR2(200 BYTE) CONSTRAINT "ADDRESS\_NN" NOT NULL ENABLE,

"COMPANYNAME" VARCHAR2(80 BYTE) CONSTRAINT "SITE\_COMPANYNAME\_NN" NOT NULL ENABLE,

"POSTCODE" VARCHAR2(8 BYTE) CONSTRAINT "SITE\_POSTCODE\_NN" NOT NULL ENABLE,

CONSTRAINT "SITEID\_PK" PRIMARY KEY ("SITEID"),

CONSTRAINT "SITE\_POSTCODE\_CHK" CHECK (postcode LIKE '%\_\_\_ \_\_\_') ENABLE,

CONSTRAINT "COMPANYNAME\_FK" FOREIGN KEY ("COMPANYNAME")

REFERENCES "APEX811"."COMPANIES" ("COMPANYNAME") ENABLE

) ;

CREATE SEQUENCE "APEX811"."SEQ\_SITE\_ID" MINVALUE 1 MAXVALUE 99999 INCREMENT BY 1 START WITH 10016 NOCACHE NOORDER CYCLE NOKEEP NOSCALE GLOBAL ;

CREATE TABLE "APEX811"."JOBS"

( "JOBID" VARCHAR2(7 BYTE) DEFAULT ON NULL 'J'||LTRIM(TO\_CHAR("APEX811"."SEQ\_JOB\_ID"."NEXTVAL",'09999')) NOT NULL ENABLE,

"JOBTITLE" VARCHAR2(40 BYTE) CONSTRAINT "JOBS\_JOB\_TITLE\_NN" NOT NULL ENABLE,

"JOBDESCRIPTION" VARCHAR2(2000 BYTE) CONSTRAINT "JOBS\_JOB\_DESCRIPTION\_NN" NOT NULL ENABLE,

"CONTACTEMAIL" VARCHAR2(100 BYTE) CONSTRAINT "JOB\_EMAIL\_NN" NOT NULL ENABLE,

"CONTACTTELEPHONE" VARCHAR2(11 BYTE) CONSTRAINT "JOB\_TELEPHONE\_NN" NOT NULL ENABLE,

"SALARY" NUMBER(7,2),

"STARTDATE" DATE,

"VACANCIESAVAILABLE" NUMBER(2,0),

"APPLICATIONCLOSINGDATE" DATE CONSTRAINT "JOBS\_APPLICATION\_CLOSING\_DATE\_NN" NOT NULL ENABLE,

"APPLICATIONMETHOD" VARCHAR2(30 BYTE) CONSTRAINT "JOBS\_APPLICATION\_METHOD\_NN" NOT NULL ENABLE,

"SITEID" VARCHAR2(7 BYTE) CONSTRAINT "SITEID\_NN" NOT NULL ENABLE,

CONSTRAINT "JOBS\_JOB\_ID\_PK" PRIMARY KEY ("JOBID"),

CONSTRAINT "JOB\_EMAIL\_CHK" CHECK (contactemail LIKE '%\_@\_\_\_%.\_\_%') ENABLE,

CONSTRAINT "JOB\_MOBILE\_CHK" CHECK (contacttelephone NOT LIKE '%[^0-9]%') ENABLE,

CONSTRAINT "JOBS\_SITE\_ID\_FK" FOREIGN KEY ("SITEID")

REFERENCES "APEX811"."SITES" ("SITEID") ENABLE

) ;

CREATE SEQUENCE "APEX811"."SEQ\_JOB\_ID" MINVALUE 1 MAXVALUE 99999 INCREMENT BY 1 START WITH 10024 NOCACHE NOORDER CYCLE NOKEEP NOSCALE GLOBAL ;

CREATE TABLE "APEX811"."PROGRAMMES"

( "PROGRAMME\_CODE" CHAR(4 BYTE),

"PROGRAMME\_NAME" VARCHAR2(60 BYTE) CONSTRAINT "PROGRAME\_NAME\_NN" NOT NULL ENABLE,

"PLACEMENT\_MANDATORY" CHAR(1 BYTE),

CONSTRAINT "PROGRAMME\_CODE\_PK" PRIMARY KEY ("PROGRAMME\_CODE")

) ;

CREATE TABLE "APEX811"."STUDENTS"

( "STUDENTID" NUMBER DEFAULT ON NULL "APEX811"."SEQ\_STUDENT\_ID"."NEXTVAL" NOT NULL ENABLE,

"USERNAME" VARCHAR2(20 BYTE) CONSTRAINT "USERNAME\_NN" NOT NULL ENABLE,

"PASSWORD" VARCHAR2(20 BYTE) CONSTRAINT "PASSWORD\_NN" NOT NULL ENABLE,

"FIRSTNAME" VARCHAR2(30 BYTE) CONSTRAINT "FIRSTNAME\_NN" NOT NULL ENABLE,

"LASTNAME" VARCHAR2(45 BYTE) CONSTRAINT "LASTNAME\_NN" NOT NULL ENABLE,

"EMAIL" VARCHAR2(100 BYTE) CONSTRAINT "STUDENT\_EMAIL\_NN" NOT NULL ENABLE,

"MOBILE" VARCHAR2(11 BYTE),

"LANDLINE" VARCHAR2(11 BYTE),

"PROGRAMME\_CODE" CHAR(4 BYTE) CONSTRAINT "PROGRAMME\_CODE\_NN" NOT NULL ENABLE,

"DOB" DATE CONSTRAINT "DOB\_NN" NOT NULL ENABLE,

"PREFERENCE" VARCHAR2(2000 BYTE),

"TERMADDRESS" VARCHAR2(200 BYTE) CONSTRAINT "STUDENT\_TERMADDRESS\_NN" NOT NULL ENABLE,

"TERMPOSTCODE" VARCHAR2(8 BYTE) CONSTRAINT "STUDENT\_TERMPOSTCODE\_NN" NOT NULL ENABLE,

"HOMEADDRESS" VARCHAR2(200 BYTE) CONSTRAINT "STUDENT\_HOMEADDRESS\_NN" NOT NULL ENABLE,

"HOMEPOSTCODE" VARCHAR2(8 BYTE) CONSTRAINT "STUDENT\_HOMEPOSTCODE\_NN" NOT NULL ENABLE,

"CVSUBMITTED" DATE,

"CVACCEPTED" DATE,

CONSTRAINT "STUDENTID\_PK" PRIMARY KEY ("STUDENTID"),

CONSTRAINT "STUDENT\_EMAIL\_CHK" CHECK (email LIKE '%\_@\_\_\_%.\_\_%') ENABLE,

CONSTRAINT "STUDENT\_MOBILE\_CHK" CHECK (mobile NOT LIKE '%[^0-9]%') ENABLE,

CONSTRAINT "STUDENT\_LANDLINE\_CHK" CHECK (landline NOT LIKE '%[^0-9]%') ENABLE,

CONSTRAINT "STUDENT\_TERMPOSTCODE\_CHK" CHECK (termpostcode LIKE '%\_\_\_ \_\_\_') ENABLE,

CONSTRAINT "STUDENT\_HOMEPOSTCODE\_CHK" CHECK (homepostcode LIKE '%\_\_\_ \_\_\_') ENABLE,

CONSTRAINT "FK\_PROGRAMME\_CODE" FOREIGN KEY ("PROGRAMME\_CODE")

REFERENCES "APEX811"."PROGRAMMES" ("PROGRAMME\_CODE") ENABLE

) ;

CREATE SEQUENCE "APEX811"."SEQ\_STUDENT\_ID" MINVALUE 1 MAXVALUE 999999 INCREMENT BY 1 START WITH 100042 NOCACHE NOORDER CYCLE NOKEEP NOSCALE GLOBAL ;

CREATE OR REPLACE EDITIONABLE TRIGGER "APEX811"."TRG\_CV\_ACCEPTED"

BEFORE UPDATE OF CVACCEPTED ON STUDENTS FOR EACH ROW

BEGIN

IF :OLD.CVSUBMITTED IS NULL

THEN :NEW.CVACCEPTED := NULL;

END IF;

END;

/

ALTER TRIGGER "APEX811"."TRG\_CV\_ACCEPTED" ENABLE;

CREATE OR REPLACE EDITIONABLE TRIGGER "APEX811"."TRG\_DOB"

BEFORE INSERT OR UPDATE OF DOB ON STUDENTS FOR EACH ROW

BEGIN

IF (:NEW.dob + NUMTOYMINTERVAL(18,'YEAR')) >

SYSDATE THEN

RAISE\_APPLICATION\_ERROR(-18, 'Student must be at least 18 years old');

END IF;

END;

/

ALTER TRIGGER "APEX811"."TRG\_DOB" ENABLE;

CREATE TABLE "APEX811"."APPLICATIONS"

( "APPLICATIONID" VARCHAR2(9 BYTE) DEFAULT ON NULL 'A'||LTRIM(TO\_CHAR("APEX811"."SEQ\_APPLICATION\_ID"."NEXTVAL",'0999999')) NOT NULL ENABLE,

"STUDENTID" NUMBER CONSTRAINT "STUDENT\_ID\_NN" NOT NULL ENABLE,

"JOBID" VARCHAR2(7 BYTE) CONSTRAINT "JOB\_ID\_NN" NOT NULL ENABLE,

"APPLICATIONSTATUS" VARCHAR2(28 BYTE) CONSTRAINT "APPLICATIONSTATUS\_NN" NOT NULL ENABLE,

"STATUSDATE" DATE,

CONSTRAINT "APPLICATION\_ID\_PK" PRIMARY KEY ("APPLICATIONID"),

CONSTRAINT "APPLICATIONS\_STUDENT\_ID\_FK" FOREIGN KEY ("STUDENTID")

REFERENCES "APEX811"."STUDENTS" ("STUDENTID") ENABLE,

CONSTRAINT "FK\_JOBID" FOREIGN KEY ("JOBID")

REFERENCES "APEX811"."JOBS" ("JOBID") ENABLE

) ;

CREATE SEQUENCE "APEX811"."SEQ\_APPLICATION\_ID" MINVALUE 1 MAXVALUE 9999999 INCREMENT BY 1 START WITH 1000029 NOCACHE NOORDER CYCLE NOKEEP NOSCALE GLOBAL ;

CREATE OR REPLACE EDITIONABLE TRIGGER "APEX811"."TRG\_APPLICATION\_UPDATE"

BEFORE UPDATE OF APPLICATIONSTATUS, STATUSDATE ON APPLICATIONS FOR EACH ROW

BEGIN

INSERT INTO APPLICATIONHISTORIES

(

APPLICATIONID, APPLICATIONSTATUS, STATUSDATE

)

VALUES

(

:OLD.APPLICATIONID, :OLD.APPLICATIONSTATUS, :OLD.STATUSDATE

);

END;

ALTER TRIGGER "APEX811"."TRG\_APPLICATION\_UPDATE" ENABLE;

CREATE TABLE "APEX811"."APPLICATIONHISTORIES"

( "APPLICATIONID" VARCHAR2(9 BYTE),

"APPLICATIONSTATUS" VARCHAR2(28 BYTE) CONSTRAINT "APPLICATIONHISTORIESSTATUS\_NN" NOT NULL ENABLE,

"STATUSDATE" DATE,

CONSTRAINT "APPLICATIONHISTORIES\_PK" PRIMARY KEY ("APPLICATIONID", "APPLICATIONSTATUS"),

CONSTRAINT "APPLICATION\_HISTORIES\_ID\_FK" FOREIGN KEY ("APPLICATIONID")

REFERENCES "APEX811"."APPLICATIONS" ("APPLICATIONID") ENABLE

) ;