

BIENVENIDOS



PROYECTO DE FIN DE CICLO

APLICACIÓN PARA BECARIOS DE UST

ANA ISABEL FUENTES GÓMEZ
48157607-T

2º AÑO GRADO SUPERIOR DESARROLLO DE APLICACIONES MULTIPLATAFORMA
TUTOR: ANAHÍ MULA
IES LAS SALINAS – SESEÑA NUEVO

Índice

Abstract	2
Justificación	3
Introducción	3
Objetivos.....	3
Desarrollo	3
Ilustración 1. Diagrama de Gantt	4
Ilustración 2. Organización de Proyecto	4
Acerca de la organización	5
Análisis de requerimientos	5
Ilustración 3. Diagrama de tiempo de la realización de la aplicación.	5
Coste de la realización de aplicación	6
Ilustración 4. Costes de la aplicación.	6
Hardware	6
Sistemas Operativos y aplicaciones	6
Bases de datos y acceso a datos	7
Diseño de la aplicación	8
Ilustración 5. Diagrama de casos de uso, según el usuario.	8
Ilustración 6. Diagrama de casos de uso, según el Administrador.	8
Programación	9
Ilustración 7. Primer Activity.....	11
Ilustración 8. Segunda Activity.....	13
Ilustración 9. Tercera Activity.	14
Ilustración 10. Activity de Documentos.	16
Documentación	17
Tutorial de instalación.	17
Desarrollo de interfaces.....	19
Ilustración 11. Background	22
Ilustración 12. Diseño decoración.....	22
Sistemas de Gestión Empresarial.....	22
Conclusiones.....	23
Anexos	24
Ilustración 13. Permiso	24
Bibliografía.....	24

Solo dispone el documento de 25 páginas dado que este proyecto es de 25 horas y rellenar con más contenido seria innecesario para la explicación de la aplicación.

Abstract

Mi proyecto final consta de una aplicación para los becarios de la empresa en la que estoy realizando las prácticas de fin de grado, en concreto UST Global, nos hemos dado cuenta al realizar las practicas la falta de organización de la empresa con los nuevos becarios.

En dicha aplicación se recogerá toda la formación y todos los enlaces de interés a la hora de entrar en la empresa.

La aplicación constara de la información de los cursos que hemos recibido y de los enlaces de interés que nos facilitan de una manera organizada, ordenada y recurrente para así facilitar a los nuevos becarios de UST Global unas prácticas más organizadas a la hora de realizar las diferentes formaciones que solicita la empresa.

My final project consists of an application for the fellows of the company in which I am doing the end-of-degree practices, specifically UST Global, we have realized when doing the practices of the lack of organization of the company with the new fellows .

In this application, all the training and all links of interest will be collected when entering the company.

The constant application of the information from the courses we have received and the links of interest that they provide us in an organized, orderly and recurring way in order to provide new scholarships to UST Global with more organized practices when carrying out the different training requested by the company.

Justificación

La realización de este proyecto se basa en tener de una forma clara y organizada la información de los cursos que deben realizar los becarios a la hora de entrar a UST Global.

La idea surge tras mi experiencia como becaria en UST Global y comprobar de primera mano las carencias que tiene dicha empresa, como es la organización para becarios.

Es una aplicación creada por la falta de organización de los cursos que se les ofrece a los becarios, se ha desarrollado de una forma que sea útil tanto como becarios como ya empleados de UST Global, que pueden recurrir a la información a la hora de realizar aplicaciones para terceros.

Introducción

Principalmente he obtenido datos de los diferentes cursos que se solicitan para trabajar en UST Global, entre ellos son Appian y Testing.

Añadiremos también enlaces de interés como por ejemplo la web de Appian interna en UST Global, enlace a Kubera la red social de la empresa y además un apartado de empresas que tienen convenio con UST Global para los empleados.

Objetivos

Con esta aplicación se pretende tener de una manera organizada la aplicación, de esta manera se utilizará la aplicación como una guía para el becario de los cursos y enlaces de interés que debe disponer para trabajar en UST Global.

Desarrollo

El desarrollo ha sido realizado en dos fases:

En la primera fase nos hemos centrado en la memoria, en lo que se necesita para realizar la aplicación, así como los documentos, la organización, los análisis de requerimientos y todo lo necesario de la parte escrita de la aplicación.

También en la primera fase se ha realizado una presentación para la argumentación ante el tribunal de dicha aplicación.

En la segunda, nos centramos en cómo realizar una base de datos en la aplicación y en la realización de la aplicación en sí.

Para la segunda fase hemos consultado la documentación de SQLite para la realización de la base de datos.

DIAGRAMA DE GANTT

N° Actividad	Inicio	Final	19-may	20-may	21-may	22-may	25-may	26-may	27-may	28-may	29-may	01-jun	02-jun
MEMORIA	19/05/2020	21/05/2020											
ORGANIZACIÓN	20/05/2020	21/05/2020											
ANÁLISIS DE REQUERIMIENTOS	20/05/2020	21/05/2020											
SISTEMAS OPERATIVOS	22/05/2020	22/05/2020											
DISEÑO DE LA APLICACIÓN Y PROGRAMACIÓN	22/05/2020	22/05/2020											
DOCUMENTACIÓN	25/05/2020	25/05/2020											
DESARROLLO DE INTERFACES	25/05/2020	27/05/2020											
REVISION TUTOR	28/05/2020	28/05/2020											
REVISION TUTOR	29/05/2020	29/05/2020											
ARTE FINAL	01/06/2020	01/06/2020											
IMPRESIÓN DOCUMENTO	02/06/2020	02/06/2020											

Ilustración 1. Diagrama de Gantt

A la hora de realizar el diagrama de Gantt hemos usado una plantilla de Excel para crearlo.

Nos hemos ayudado de una herramienta de gestión de proyectos que se llama Monday para el seguimiento del diagrama, es una herramienta sencilla de usar y se puede conectar con una cuenta de Slack o Google.

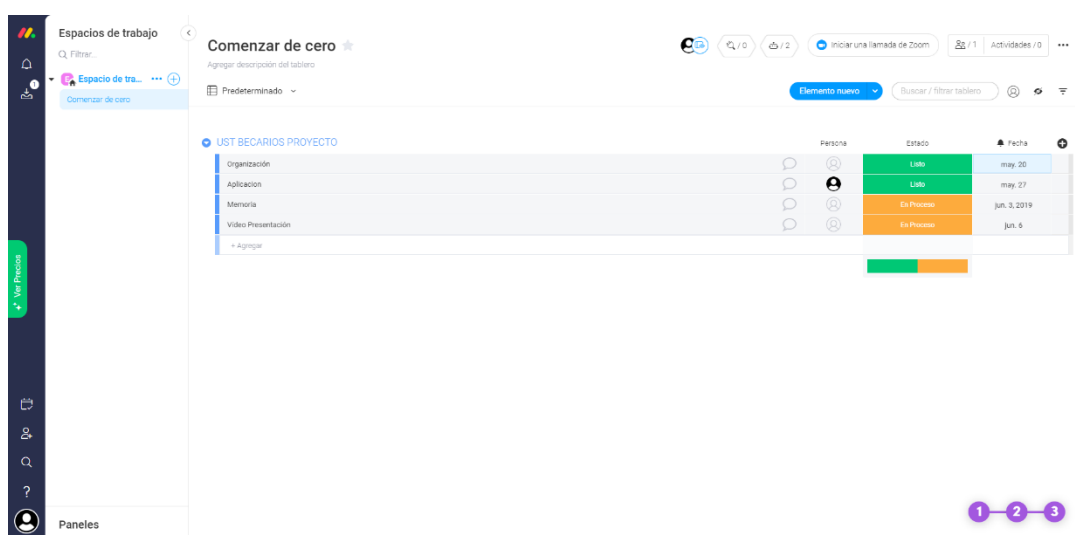


Ilustración 2. Organización de Proyecto

Nos ha venido muy bien esta herramienta para las pautas que hemos tenido que seguir respecto a la estructura y el tiempo necesario para realizar la aplicación.

Acerca de la organización

Somos una compañía digital a la vanguardia de las soluciones, plataformas y servicios más innovadores, un equipo de más de 25.000 profesionales repartidos a lo largo de los cinco continentes.

En nuestra geografía somos más de 1.500 empleados distribuidos entre España y LATAM coordinados desde la sede principal en Madrid.

Nuestros clientes están distribuidos por todo el mundo y entre todos los sectores de actividad: desde banca, servicios financieros, seguros, salud, retail, telco, media, energía, utilities y logística...

Contamos con los equipos informativos de mejor calidad y con los precios más competitivos del mercado para nuestros empleados, cuando el sueldo de un Programador Junior ronda los 1.600€.

La humanidad, la humildad y la integridad son parte de nuestro ADN y guían la forma en que trabajamos y nos relacionamos entre nosotros, con clientes, partners y en sí toda la comunidad global.

Análisis de requerimientos

La aplicación "UST BECARIOS" es una aplicación pensada internamente para los nuevos empleados de UST Global, en ella debe constar guías de realización de proyectos, enlaces de interés para la empresa, información y documentación oficial de los cursos a realizar y el enlace web a la página oficial de UST Global España.

Aquí se muestra el diagrama de Gantt de la realización la aplicación.

DIAGRAMA DE GANTT DESARROLLO APLICACIÓN "UST BECARIO"



Nº Actividad	Inicio	Final	18-may	19-may	20-may	21-may	22-may	22-may	26-may	27-may	28-may
Reunión para utilización de la información	18/05/2020	18/05/2020									
Planteamiento de la información	19/05/2020	19/05/2020									
Diseño sencillo de la aplicación	20/05/2020	20/05/2020									
Organización de la información	21/05/2020	22/05/2020									
Busqueda de informacion BBDD	22/05/2020	22/05/2020									
Pruebas	22/05/2020	26/05/2020									
Arte Final	27/05/2020	28/05/2020									

Ilustración 3. Diagrama de tiempo de la realización de la aplicación.

Coste de la realización de aplicación

Costes de la aplicación "UST Becarios"



COSTES	HORA	TOTAL
Desarrollador senior	27,5/h	1.485 €
Equipo "ORDENADOR DELL LATITUD 5400"		900 €
Publicacion Google Play		20 €
Coste Total		2.405 €

Ilustración 4. Costes de la aplicación.

Aquí se muestra los costes de la realización de la aplicación constando el coste de equipo, de empleados y servicio, se han tenido en cuenta solo los costes del proyecto, y no los costes de espacio de oficinas.

Hardware

Disponemos del equipo que mencionamos en "Costes de aplicación" es un equipo que dispone de un procesador Intel® Core™ i7-8665U de 8.ª generación, lo necesario para que nuestro programa Android Studio tenga un rendimiento sin problemas, no disponemos de un terminal físico para la realización de las pruebas, aunque no supone ningún problema, hemos usado el emulador que viene integrado en el programa en este caso el del terminal Nexus 4.

Sistemas Operativos y aplicaciones

La aplicación será desarrollada en Windows, ya que es el sistema operativo del que dispone la empresa.

Para la realización de la aplicación utilizaremos Android Studio, este ya tiene un emulador para poder ir testeando la aplicación sin necesidad de disponer de un terminal.

También hemos utilizado aplicaciones como Canva y Photoshop para el diseño de imágenes y background.

Bases de datos y acceso a datos

Para la base de datos de esta aplicación he utilizado SQLite, SQLite es una biblioteca escrita en lenguaje C que implementa un Sistema de gestión de bases de datos transaccionales SQL autocontenido, sin servidor y sin configuración. Es fácil de utilizar ya que no es una base de datos externa.

Para la creación de la base de datos debemos ir al archivo AndroidManifest.xml que hemos creado en la aplicación y añadir los permisos necesarios.

```
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

Tenemos que crear la clase Java que implemente la interfaz SQLiteOpenHelper, al implementarla se crea un constructor para la clase y los métodos.

Los métodos que se crean son onUpdate y onCreate, se encargan de actualizar el esquema de la base de datos y de crear las tablas de nuestra BBDD.

```
1 package com.example.ustglobalbecarios;
2
3 import android.content.Context;
4 import android.database.sqlite.SQLiteDatabase;
5 import android.database.sqlite.SQLiteOpenHelper;
6
7 import androidx.annotation.Nullable;
8
9 public class AdminSQLiteOpenHelper extends SQLiteOpenHelper {
10     public AdminSQLiteOpenHelper(@Nullable Context context, @Nullable String name, @Nullable SQLiteDatabase.CursorFactory factory, int version) {
11         super(context, name, factory, version);
12     }
13
14     @Override
15     public void onCreate(SQLiteDatabase db) {
16         db.execSQL("create table documentos (id int primary key, titulo varchar(50), enlace varchar(100))");
17     }
18
19     @Override
20     public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
```

```
public void sacarEnlace (String t, ImageView boton){
    AdminSQLiteOpenHelper admin = new AdminSQLiteOpenHelper( context: this, name: "documentos", factory: null, version: 1);
    SQLiteDatabase baseDatos = admin.getReadableDatabase();

    Cursor fila = baseDatos.rawQuery( sql: "SELECT enlace FROM documentos WHERE titulo = " + t + "'", selectionArgs: null);

    //La utilización de estos Toast, es para verificar si hacia la consulta y si recoge los datos

    //Toast.makeText(this, "Hace ka consulta", Toast.LENGTH_LONG).show();
    /*if(fila!=null && fila.getCount()>0){
        Toast.makeText(this, "Tiene datos", Toast.LENGTH_LONG).show();
    } else {
        Toast.makeText(this, "No tiene datos", Toast.LENGTH_LONG).show();
    }*/

    if(fila.moveToFirst()){
        // Toast.makeText(this, "Entra en el if", Toast.LENGTH_LONG).show();
        enlace = fila.getString( columnIndex: 0);
        //Toast.makeText(this, "coge el enlace: " + enlace, Toast.LENGTH_LONG).show();
    }

    boton.setOnClickListener(this);
    baseDatos.close();
}
```


En esta ultima captura mostramos como realiza las consultas la base de datos y como hemos utilizado los Toast para la comprobación de las consultas y de la obtención de datos de la base de datos.

Diseño de la aplicación

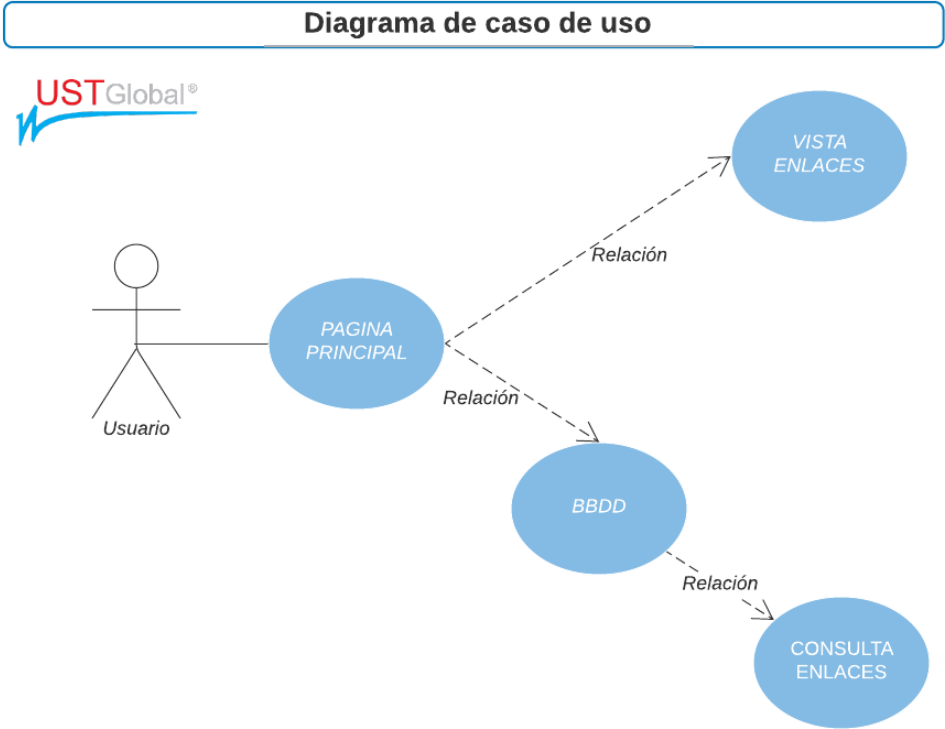


Ilustración 5. Diagrama de casos de uso, según el usuario.

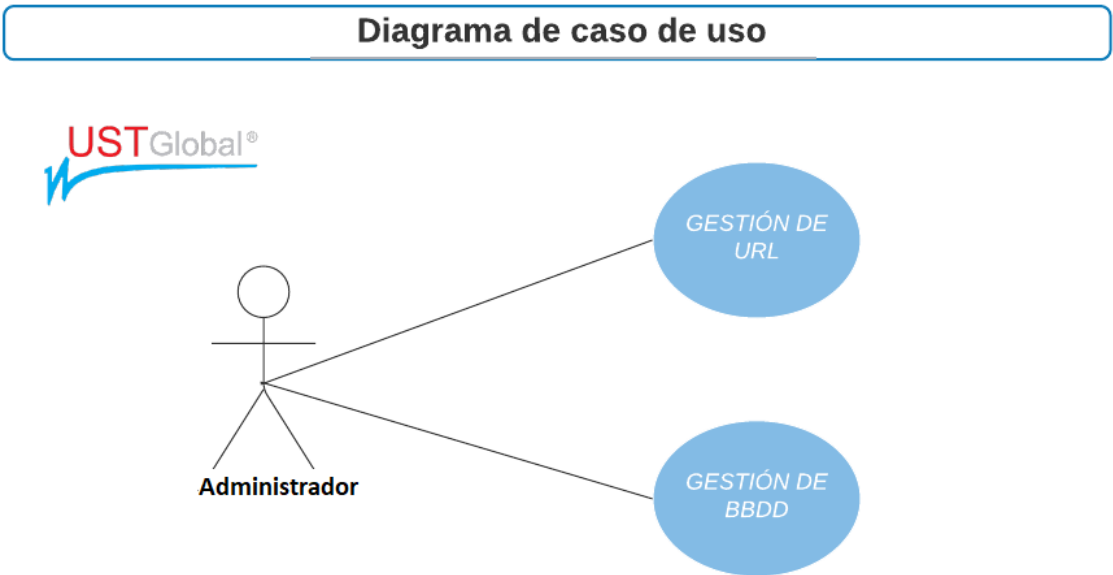


Ilustración 6. Diagrama de casos de uso, según el Administrador.

Programación

```
public class AdminSQLiteOpenHelper extends SQLiteOpenHelper {
    public AdminSQLiteOpenHelper(@Nullable Context, @Nullable String name,
    @Nullable SQLiteDatabase.CursorFactory factory, int version) {
        super(context, name, factory, version);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("create table documentos (id int primary key, titulo
        varchar(50), enlace varchar(100))");
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    }
}
```

Como se ha comentado anteriormente en el punto de la base de datos en esta parte del código de muestra de la utilización del metodo onCreate y onUpgrade.

```
public class MainActivity extends AppCompatActivity {
    Button siguiente;
    Button documentos;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        siguiente = findViewById(R.id.siguiente);
        documentos = findViewById(R.id.documentacion);

        cargarDatos(1, "Appian",
        "https://drive.google.com/file/d/1bwgnmsiCf6jfq1f5KqjgwDL6AvJt7ugX/view?usp=drives
        dk");
        cargarDatos(2, "Testing",
        "https://drive.google.com/drive/folders/14HyHaSVJMGPG5PjTwAaEaBa0vnbH01sD?usp=shar
        ing");
    }

    public void funcionAppian(View v){
        Intent = new Intent(Intent.ACTION_VIEW);
        intent.setData(Uri.parse("https://ustglobalspaindemo.appiancloud.com/suite/tempo/n
        ews"));
        if(intent.resolveActivity(getPackageManager())!= null){
            startActivity(intent);
        }
    }

    public void funcionEstrategno(View v){
        Intent intent3 = new Intent(Intent.ACTION_VIEW);
        intent3.setData(Uri.parse("https://www.estrategno.es/"));
        if(intent3.resolveActivity(getPackageManager())!= null){

```

```

        startActivity(intent3);
    }
}
public void funcionMBA(View v){
    Intent intent4 = new Intent(Intent.ACTION_VIEW);
    intent4.setData(Uri.parse("https://thepowermba.com/es/"));
    if(intent4.resolveActivity(getPackageManager())!= null){
        startActivity(intent4);
    }
}
public void funcionOWeinars(View v){
    Intent intent5 = new Intent(Intent.ACTION_VIEW);

intent5.setData(Uri.parse("https://openwebinars.net/accounts/login/?next=/academia/cursos/"));
    if(intent5.resolveActivity(getPackageManager())!= null){
        startActivity(intent5);
    }
}
public void funcionEnglish(View v){
    Intent intent5 = new Intent(Intent.ACTION_VIEW);
    intent5.setData(Uri.parse("https://idiomasseif.com/"));
    if(intent5.resolveActivity(getPackageManager())!= null){
        startActivity(intent5);
    }
}

public void accionSiguiente(View v) {
    Intent intent1 = new Intent(MainActivity.this, segunda.class);
    startActivity(intent1);
}

public void acciondocumentos(View v) {
    Intent intent2 = new Intent(MainActivity.this, documentos.class);
    startActivity(intent2);
}

public void cargarDatos(int id, String pTitulo, String pEnlace){
    AdminSQLiteOpenHelper admin = new AdminSQLiteOpenHelper(this,
"documentos", null, 1);
    SQLiteDatabase baseDatos = admin.getWritableDatabase();

    String dataId = Integer.toString(id);
    String dataTitulo = pTitulo;
    String dataEnlace = pEnlace;

    ContentValues contenido = new ContentValues();

    contenido.put("id", dataId);
    contenido.put("titulo", dataTitulo);
    contenido.put("enlace", dataEnlace);

    baseDatos.insert("documentos", null, contenido);
    baseDatos.close();
    //Toast.makeText(this, "Abre la base", Toast.LENGTH_LONG).show();
}

```

Este código muestra el MainActivity que consta de dos partes una que es las funciones y acciones de botones y enlaces a url y otra parte es la parte de cargar los datos de la BBDD donde se especifica el contenido que tiene la BBDD.

Como se puede ver en el código llamamos funciones a la utilización de un testview como botón utilizando intent.



Ilustración 7. Primer Activity.

```
public class segunda extends AppCompatActivity {
    Button atras;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_segunda);

        atras = findViewById(R.id.atras);
    }

    public void funcionKubera(View v){
        Intent intent4 = new Intent(Intent.ACTION_VIEW);
        intent4.setData(Uri.parse("https://idp.ust-global.com/adfs/ls/?client-request-id=4833569f-c00f-0000-52ff-ac67b066e653&username=&wa=wsignin1.0&wtrealm=urn%3afederation%3aMicrosoftOnline&wctx=estsredirect%3d2%26estsrequest%3drQIIA21Ru24TQRtdsZ0ltgKJIgo6UrhCGnvuzOw8LFHY3l0ilIiXwVKaaJ-"));
    }
}
```

```

xFcdrdteCpKSioEhNE4kihYUQokKIALGmSos_AF1UiIoObEFBGXGKo_s4xbnn3ihCDRpV8gcMLxiTOAYcR
IvqH6Tr1bWTR2xz9ZNh3P_1eu9t_PXBS3Rln0V4b5D43qAWJAcTxHt5Psoa9fp88Xee9bw0GiX9Yb6Q1Hf
jJD3I6mEUe-
NBXvOy0dP3CJ0jNENoUnhMqKYAYBGwddPmRGjLccCmrnDBYQ6bk2QKqG1bSrRAYEFYRLVbUtqCKgtcxYVD
551uQ8vVzXbLFZy0iGRCa9exgTPp6LaUrmUJ4Th8Wli90xznPbqgJ00fRT8K5YXH3VGS5ZPihdJ5V6x6nD
0IuY8DSiTkjGsLUYw4wFRnvapr8KzopmMomE_PC-hb6UVUmwSL1fWjGvGhvGzhF4tzTN-
8_mjfhG9u33yZXbaSU-
Ns6U63xY7d_cfbjLZ0L112H0SdA81Azn09m_30y7fbN6TkKtg66iT3GQNODbRsWn0TPTdRM8vGR_K__WTa
eUqJXR-
pYWp2ADa4LoBYme6ApVyKpipNwz74XoVwA8VUQwrGRHMwbew70uFiR9wFihPRcp7dv1Cst81#"));
        if(intent4.resolveActivity(getPackageManager())!= null){
            startActivity(intent4);
        }
    }
    public void funcionTeams(View v){
        Intent intent5 = new Intent(Intent.ACTION_VIEW);
        intent5.setData(Uri.parse("https://www.microsoft.com/es-es/microsoft-
365/microsoft-teams/group-chat-software"));
        if(intent5.resolveActivity(getPackageManager())!= null){
            startActivity(intent5);
        }
    }
    public void funcionOutlook(View v){
        Intent intent5 = new Intent(Intent.ACTION_VIEW);

intent5.setData(Uri.parse("https://login.microsoftonline.com/common/oauth2/authori
ze?client_id=00000002-0000-0ff1-ce00-
000000000000&redirect_uri=https%3a%2f%2foutlook.office.com%2fmail%2f&resource=0000
0002-0000-0ff1-ce00-
000000000000&response_mode=form_post&response_type=code+id_token&scope=openid&msaf
ed=0&client-request-id=87b10068-40b6-41f7-b4e1-
96e4c1966b1c&protectedtoken=true&nonce=637260941924587325.292ade88-9603-468c-adff-
adb5fc88cce7&state=Dcs5DoAwDABBB0reEhKcw_ZzckpIHC3Px8V0twYAVm3RjFeAcIDMXuIhGBNTwLS
jY0mD2Ur2wcbMzZY-p1LTbMytDTL6bu4u5-XOp77fDw"));
        if(intent5.resolveActivity(getPackageManager())!= null){
            startActivity(intent5);
        }
    }

    public void accionAtras(View v) {
        Intent intent2 = new Intent(segunda.this, MainActivity.class);
        startActivity(intent2);
    }

    public void accionSocial(View v) {
        Intent intent2 = new Intent(segunda.this, tercera.class);
        startActivity(intent2);
    }
}

```

Así es el código de la segunda pantalla, donde utilizamos también las funciones para los enlaces de los textview que en esta aplicación tienen la función de botón, la utilización de textview ha sido estética.

En la imagen muestra cómo se representa visualmente el activity.

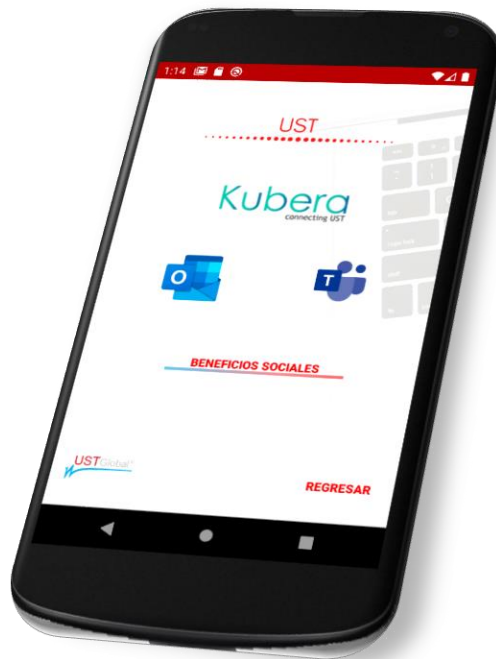


Ilustración 8. Segunda Activity.

```
Public class tercera extends AppCompatActivity {
    Button atrás2;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tercera);

        atrás2 = findViewById(R.id.atras);
    }
    public void funcionguarderia(View v){
        Intent intent3 = new Intent(Intent.ACTION_VIEW);
        intent3.setData(Uri.parse("http://www.chiquikid.com/"));
        if(intent3.resolveActivity(getPackageManager())!= null){
            startActivity(intent3);
        }
    }
    public void funcioncheque(View v){
        Intent intent4 = new Intent(Intent.ACTION_VIEW);
        intent4.setData(Uri.parse("https://www.familycheck.es/"));
        if(intent4.resolveActivity(getPackageManager())!= null){
            startActivity(intent4);
        }
    }
    public void funcionBBVA(View v){
        Intent intent5 = new Intent(Intent.ACTION_VIEW);
        intent5.setData(Uri.parse("https://www.fbbva.es/"));
        if(intent5.resolveActivity(getPackageManager())!= null){
            startActivity(intent5);
        }
    }
}
```

```

    }
    public void funcionGYM(View v){
        Intent intent5 = new Intent(Intent.ACTION_VIEW);
        intent5.setData(Uri.parse("https://www.basic-fit.com/es-es/gimnasios/madrid"));
        if(intent5.resolveActivity(getPackageManager())!= null){
            startActivity(intent5);
        }
    }
    public void accionAtras(View v) {
        Intent intent3 = new Intent(tercera.this, segunda.class);
        startActivity(intent3);
    }
}

```

En el tercer activity se sigue utilización las funciones que abren las url y los botones que realizan las acciones de navegación en la aplicación.



Ilustración 9. Tercera Activity.

```

public class documentos extends AppCompatActivity implements View.OnClickListener
{
    Button atras3;
    ImageView appian, testing;
    String enlace = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {

```

```

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_documentos);

        atras3 = findViewById(R.id.atras);
        appian = findViewById(R.id.imageView17);
        testing = findViewById(R.id.imageView18);

        sacarEnlace("Appian", appian);
        sacarEnlace("Testing", testing);
    }

    public void accionAtras(View v) {
        Intent intent4 = new Intent(documentos.this, MainActivity.class);
        startActivity(intent4);
    }

    public void sacarEnlace (String t, ImageView boton){
        AdminSQLiteOpenHelper admin = new AdminSQLiteOpenHelper(this,
"documentos", null, 1);
        SQLiteDatabase baseDatos = admin.getReadableDatabase();

        Cursor fila = baseDatos.rawQuery("SELECT enlace FROM documentos WHERE
titulo = '" + t + "'", null);
        //Toast.makeText(this, "Hace ka consulta", Toast.LENGTH_LONG).show();

        /*if(fila!=null && fila.getCount()>0){
            Toast.makeText(this, "Tiene datos", Toast.LENGTH_LONG).show();
        } else {
            Toast.makeText(this, "No tiene datos", Toast.LENGTH_LONG).show();
        }*/

        if(fila.moveToFirst()){
            // Toast.makeText(this, "Entra en el if", Toast.LENGTH_LONG).show();
            enlace = fila.getString(0);
            //Toast.makeText(this, "coge el enlace: " + enlace,
Toast.LENGTH_LONG).show();
        }

        boton.setOnClickListener(this);
        baseDatos.close();
    }

    @Override
    public void onClick(View v) {
        // Toast.makeText(this, "Entra en el metodo: " + enlace,
Toast.LENGTH_LONG).show();
        Intent = new Intent(Intent.ACTION_VIEW);
        intent.setData(Uri.parse(enlace));
        if(intent.resolveActivity(getPackageManager())!= null){
            startActivity(intent);
        }
    }
}

```


Como se muestra en el código es la parte que esta relacionada con la base de datos, en el cual aparte de las acciones de navegación de la aplicación se encuentra la consulta.

Se ha dejado comentado todas las comprobaciones que se han realizado para el funcionamiento de la base de datos.

Para poder acceder a los enlaces guardados en la base de datos, debe estar iniciada la cuenta de Google en el dispositivo, dado que los enlaces es en Google drive.



Ilustración 10. Activity de Documentos.



Utilización Aplicación

- ▶ En la pantalla principal el usuario puede acceder al apartado “cursos” y a la “documentación cursos”
- ▶ Al hacer click sobre los nombres de los diferentes cursos se abrirá una url de dicho curso.
- ▶ En el baton “siguiente” se pasara a la siguiente pantalla.



Utilización Aplicación

- ▶ Esta es la pantalla que se visualiza al seleccionar “Documención Cursos”



Utilización Aplicación

- ▶ En la segunda pantalla al pulsar al botón “siguiente” podemos encontrar los iconos de las aplicaciones que usa internamente UST Global.
- ▶ En esta pantalla encontramos el botón “regresar” y el botón “beneficios sociales”.



Utilización Aplicación

- ▶ En esta página se muestra la información de los beneficios sociales que ofrece la empresa los empleados de ust.

Desarrollo de interfaces

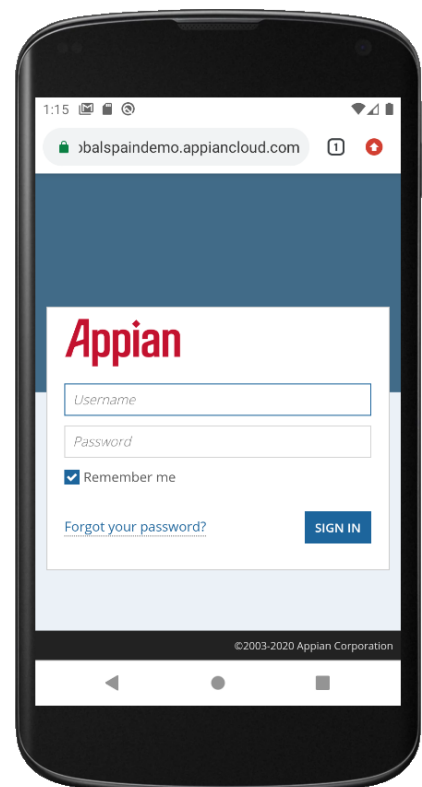


Este es el contenido que se muestra a la hora de seleccionar el botón de Appian, es un enlace a la página de Appian de UST.

El diseño de la aplicación es un diseño sencillo, en el que utilizo principalmente los colores rojo y azul, que es el color corporativo de la empresa.

En la página principal podemos encontrar los cursos y el contenido de la base de datos junto con un botón siguiente.

La distribución de los botones está colocada estratégicamente para que quede en el centro de la pantalla para un fácil acceso al usuario.





En la segunda página, podemos encontrar tres botones centrales los cual se acceden a la intranet y los servicios de Microsoft que utiliza UST Global, en este activity también seguimos usando los colores rojos, azul y el logo de UST.

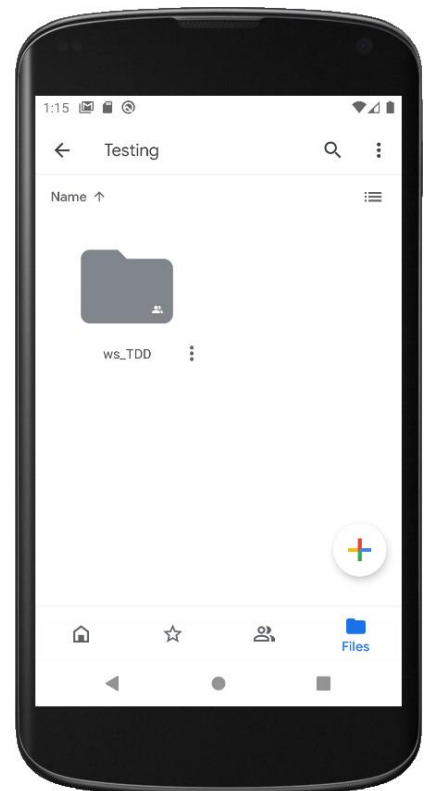
Este es el contenido que se muestra al seleccionar el botón de Beneficios sociales, este botón nos lleva a otro activity donde se recogen todos los beneficios que ofrece UST Global a sus trabajadores.





En la página principal también podíamos seleccionar el botón de “documentación cursos”, este botón está conectado a la base de datos y realiza las consultas, nos muestra dos botones Appian y Estratecno en los cuales se almacenan dos enlaces para acceder a los documentos.

Como comentábamos antes al pulsar en “Estratecno” se realiza la consulta y abre la url que esta almacena en la BBDD.



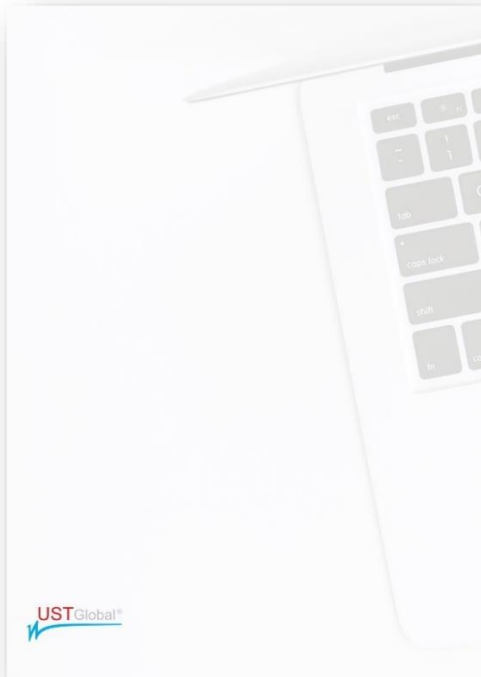


Ilustración 11. Background

Estos son los dos diseños de Background que han sido realizados con la aplicación Canva para implementar a la aplicación y mejorar su diseño.



Ilustración 12. Diseño decoración.

Con el programa Canva también hemos realizado un diseño mucho mejor del que nos ofrece el programa Android Studio para algunos botones y diseño de la memoria.

Sistemas de Gestión Empresarial

Somos una compañía digital a la vanguardia de las soluciones, plataformas y servicios más innovadores con lo cual disponemos de un sistema ERP interno creado por nuestros programadores.

El sistema ERP que disponemos en nuestra empresa que nos ayuda a :

- **Optimizar los procesos de la empresa.**
La elaboración de facturas, que se puede automatizar y hacer de forma más rápida y sin errores.

- **Favorecer el acceso a la información.**
Cuando la información es accesible para los distintos departamentos que intervienen en los procesos empresariales, se eliminan errores y tareas duplicadas.
- **Tomar decisiones de forma más rápida.**
El acceso a la información veraz sobre tu empresa te permitirá tomar decisiones basadas en datos reales.
- **Tener un mayor control.**
En el caso de que se cometa un error, el sistema ERP, gracias a la trazabilidad, permitirá detectarlo y corregirlo de forma rápida y sencilla.
- **Aumentar la seguridad.**
La información sobre la empresa y sus actividades estará protegida y, además, el programa ERP generará copias de seguridad de forma automática.
- **Mejorar la atención al cliente.**
Los datos que se obtendrán de cada cliente permitirán dar un mejor servicio y conocer sus necesidades en cada momento.
- **Reducción de los costes empresariales.**
El programa ERP reduce errores y elimina duplicidades, por lo que también se reducen los costes de las organizaciones.
- **Aumento de la rentabilidad.**
Todas las ventajas que hemos detallado anteriormente tienen una ventaja final u objetivo: el aumento de la rentabilidad en la empresa.

Conclusiones

En conclusión, es una aplicación creada para organizar y ayudar a los empleados con su trabajo, en el que pueden consultar su formación, los enlaces y beneficios sociales de la empresa.

Es una aplicación totalmente funcional y con una interfaz sencilla para la fluidez que se necesita a la hora de trabajar en una empresa que consta de diferentes proyectos en un breve periodo de tiempo.

La realización de la aplicación no ha sido compleja, y los problemas que ha podido dar la realización de la base de datos han sido solucionado con las comprobaciones pertinentes.

La aplicación tiene su correcto funcionamiento a la hora de empezar a trabajar con ella.

Anexos

Dispongo del permiso necesario para realizar esta aplicación y adjunto una fotografía del correo.



Alvaro Jusedado Cañete(UST,ES)
Para Isabel Fuentesgomez(UST,ES)

Respondió a este mensaje el 14/05/2020 16:08.

Responder

Responder a todos

Hola Isabel.

No he tenido tiempo antes, ningún problema.

Yo no he visto la formación de testing pero no creo que haya problemas. Y en cuanto a la de Appiam tampoco. Aquí si te digo que no tenemos más formación, únicamente la oficial de Appian.

Ilustración 13. Permiso

Bibliografía

<https://es.wikipedia.org/wiki/SQLite>

<https://developer.android.com/?hl=es>

<https://online.visual-paradigm.com/es/diagrams/solutions/free-use-case-diagram-tool/>

<https://ustglobal.es/>

<https://ust-global10.monday.com/>