



UNIVERSIDADE DO MINHO

APLICAÇÕES INFORMÁTICAS NA BIOMEDICINA  
MESTRADO EM ENGENHARIA INFORMÁTICA  
2019/2020

**Grupo: 1**

---

# A realização de urgências gerais num determinado hospital nacional

---

***Trabalho realizado por:***

Filipa Parente  
João Almeida  
Leonardo Jesus  
Nuno Valente

***Número de Aluno:***

A82145  
A75209  
PG39261  
A81986

18 de Dezembro de 2019

# Conteúdo

<b>Conteúdo</b>	<b>1</b>
<b>1 Introdução</b>	<b>2</b>
1.1 Plano Estrutural . . . . .	2
1.2 Descrição do problema . . . . .	2
<b>2 Resolução das questões propostas</b>	<b>4</b>
2.1 Criar um novo schema no MySQL Workbench denominado “BD_URG” . . . . .	4
2.2 Fazer o import de todos os dados no ficheiro urg_inform_geral.csv para um nova tabela denominada “urg_inform_geral” na base de dados BD_URG . . . . .	4
2.3 Desenhar e criar um modelo dimensional no formato de esquema em estrela no MySQL Workbench - EER . . . . .	5
2.4 Gerar o modelo físico associado ao modelo dimensional definido e criado no ponto anterior para uma base de dados denominada “DW_URG” . . . . .	6
2.5 Povoar o Data Warehouse na base de dados “DW_URG” a partir da tabela urg_inform_geral da base de dados BD_URG: . . . . .	6
2.5.1 Em SQL com uma script . . . . .	6
2.5.2 Com jobs no Talend . . . . .	7
2.5.3 Descrever e explicar as vantagens e as desvantagens entre os dois diferen- tes processos de povoamento utilizados (alíneas (a) e (b)) . . . . .	9
2.6 Defina e crie indicadores clínicos e de desempenho com o Microsoft Power BI (ou outro programa similar, por exemplo, Tableau) recorrendo ao Data Warehouse DW_URG implementado, justificando e descrevendo a relevância e a utilidade de cada um dos indicadores. . . . .	9
2.7 Sugira e descreva as diversas interfaces e as funcionalidade de uma aplicação informática que poderia incluir dashboards com os indicadores clínicos e de de- sempenho definidos, ou seja, de uma aplicação direcionada às urgência gerais no hospital nacional em questão, bem como das tecnologias que utilizaria para o seu desenvolvimento . . . . .	13

# 1 Introdução

## 1.1 Plano Estrutural

O seguinte documento visa elucidar os tópicos propostos para a elaboração do projeto prático, inserido na unidade curricular de Aplicações Informáticas na Biomedicina.

Numa primeira fase, na subsecção **Descrição do Problema (subsecção 1.2)**, começa-se por abordar o problema do projeto em si, bem como as tarefas a serem executadas para o seu sucesso.

Numa segunda fase, na secção **Resolução das questões propostas (secção 2)**, será explicitado o modo como cada tarefa/questão foi resolvida, através de pequenos textos com imagens ilustrativas. Esta secção está dividida em subsecções correspondentes às tarefas pedidas no enunciado.

Finalmente, na secção **Conclusões e Trabalho Futuro (secção 3)**, analisar-se-á o trabalho realizado, num cômputo geral, evidenciando as dificuldades sentidas durante a sua realização. Na mesma secção, serão sugeridas possíveis melhorias que poderão ser feitas futuramente.

## 1.2 Descrição do problema

Neste trabalho prático, inserido na unidade curricular de Aplicações Informáticas na Biomedicina, foi proposta o desenvolvimento de um Data Warehouse de um determinado hospital (não referido no enunciado).

Para o efeito, no enunciado são apresentadas as seguintes tarefas a serem executadas, com base no ficheiro .csv disponibilizado juntamente com o mesmo (*urg\_inform\_geral.csv*):

- Criar um novo schema no MySQL Workbench denominado “*BD\_URG*”;
- Fazer o import de todos os dados no ficheiro *urg\_inform\_geral.csv* para uma nova tabela denominada “*urg\_inform\_geral*” na base de dados *BD\_URG*;
- Desenhar e criar um modelo dimensional no formato de esquema em estrela no *MySQL Workbench* - *EER* Diagram. O modelo dimensional deverá ser constituído por uma tabela de factos que se liga às tabelas de dimensão definidas;
- Gerar o modelo físico associado ao modelo dimensional definido e criado no ponto anterior para uma base de dados denominada “*DW\_URG*”;
- Povoar o Data Warehouse na base de dados “*DW\_URG*” a partir da tabela *urg\_inform\_geral* da base de dados *BD\_URG*:
  - Em SQL com uma script;
  - Com jobs no Talend;
  - Descrever e explique as vantagens e as desvantagens entre os dois diferentes processos de povoamento utilizados (alíneas (a) e (b)).

- Definir e criar indicadores clínicos e de desempenho com o *Microsoft Power BI* (ou outro programa similar) recorrendo ao Data Warehouse DW\_URG implementado, justificando e descrevendo a relevância e a utilidade de cada um dos indicadores;
- Sugerir e descrever as diversas interfaces e as funcionalidade de uma aplicação informática que poderia incluir dashboards com os indicadores clínicos e de desempenho definidos, ou seja, de uma aplicação direccionada às urgência gerais no hospital nacional em questão, bem como das tecnologias que utilizaria para o seu desenvolvimento.

## 2 Resolução das questões propostas

### 2.1 Criar um novo schema no MySQL Workbench denominado “*BD\_URG*”

Para criar um novo schema no MySQL Workbench, o grupo recorreu ao script SQL ” *CREATE SCHEMA BD\_URG*”, resultando no seguinte schema:

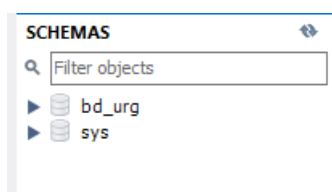


Figura 1: *schema BD\_URG no MySQL Workbench*

### 2.2 Fazer o import de todos os dados no ficheiro *urg\_inform\_geral.csv* para uma nova tabela denominada “*urg\_inform\_geral*” na base de dados *BD\_URG*

Para realizar o import dos dados, recorreu-se à ferramenta ” *Table Data Import Wizard*” presente no MySQL Workbench, tendo-se selecionado o ficheiro *urg\_inform\_geral.csv* (ficheiro, onde se faz o import dos dados).

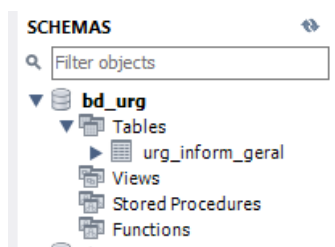


Figura 2: *tabela resultante da importação dos dados do ficheiro urg\_inform\_geral.csv*

### 2.3 Desenhar e criar um modelo dimensional no formato de esquema em estrela no *MySQL Workbench - EER*

De forma a realizar o modelo dimensional em formato de estrela, numa primeira fase fez-se uma análise dos atributos presentes na tabela *urg\_inform\_geral*.

Após a sua análise, procedeu-se à criação das tabelas da nova base de dados, a ser criada, já no MySQL Workbench-EER. Começou-se pela criação das tabelas de dimensão (*causa*, *data*, *especialidade*, *paciente*, *proveniencia*), finalizando o processo com a criação da tabela de factos (*factos*).

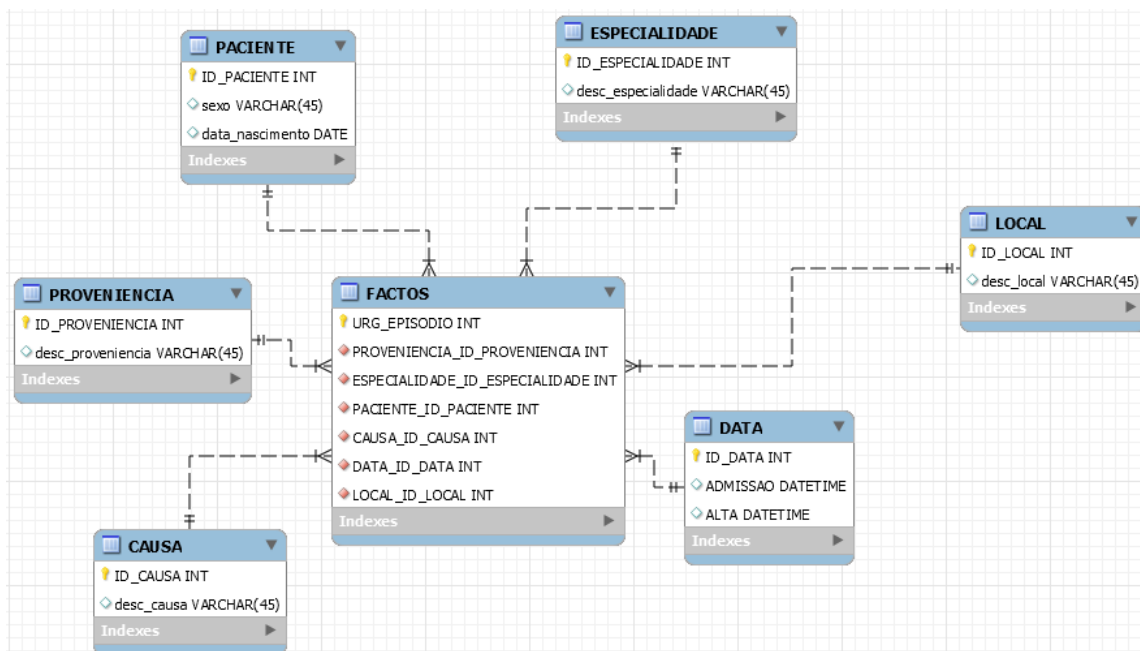


Figura 3: *Modelo dimensional, resultante da análise dos dados da tabela urg\_inform\_geral*

## 2.4 Gerar o modelo físico associado ao modelo dimensional definido e criado no ponto anterior para uma base de dados denominada “DW\_URG”

Para gerar o modelo físico, recorreu-se à ferramenta “Database > Forward Engineer” presente no MySQL Workbench, tendo-se selecionado o modelo dimensional criado em 2.3, denominando-o com o nome referenciado no enunciado do projeto.

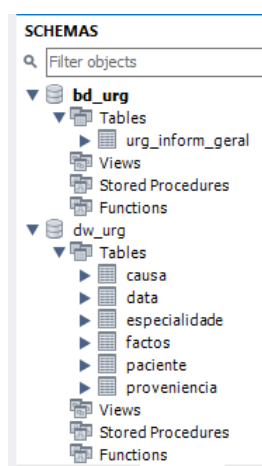


Figura 4: modelo físico dw\_urg gerado

## 2.5 Povoar o Data Warehouse na base de dados “DW\_URG” a partir da tabela urg\_inform\_geral da base de dados BD\_URG:

### 2.5.1 Em SQL com uma script

Para povoar a base de dados DW\_URG recorreu-se ao seguinte conjunto de queries SQL:

```
INSERT IGNORE INTO DW_URG.CAUSA(desc_causa)
  SELECT DISTINCT DES_CAUSA FROM urg_inform_geral;

INSERT IGNORE INTO DW_URG.ESPECIALIDADE(desc_especialidade)
  SELECT DISTINCT ALTA_DES_ESPECIALIDADE FROM urg_inform_geral;

INSERT IGNORE INTO DW_URG.PACIENTE(sexo,data_nascimento)
  SELECT DISTINCT SEXO, DTA_NASCIMENTO FROM urg_inform_geral;

INSERT IGNORE INTO DW_URG.PROVENIENCIA(desc_proveniencia)
  SELECT DISTINCT DES_PROVENIENCIA FROM urg_inform_geral;

INSERT INTO DW_URG.DATA(ADMISSAO,ALTA)
  SELECT DISTINCT DATAHORA_ADM, DATAHORA_ALTA FROM urg_inform_geral;
```

Figura 5: Queries SQL para o povoamento das tabelas de dimensão

Note-se que para povoar as tabelas de dimensão, recorreu-se ao seguinte esquema de query:

```
INSERT IGNORE INTO tabela_destino (atributo(s)_tabela_final)
  SELECT DISTINCT atributo(s)_tabela_inicial FROM tabela_inicial;
```

```
INSERT INTO DW_URG.FACTOS(URG_EPISODIO,PROVENIENCIA_ID_PROVENIENCIA,ESPECIALIDADE_ID_ESPECIALIDADE,
PACIENTE_ID_PACIENTE,CAUSA_ID_CAUSA,DATA_ID_DATA)
SELECT raw.URG_EPISODIO,pr.ID_PROVENIENCIA , e.ID_ESPECIALIDADE, pa.ID_PACIENTE, c.ID_CAUSA, d.ID_DATA
FROM BD_URG.urg_inform_geral raw
INNER JOIN DW_URG.CAUSA c on raw.DES_CAUSA=c.desc_causa
INNER JOIN DW_URG.DATA d on (raw.DATAHORA_ADM=d.ADMISSAO and raw.DATAHORA_ALTA=d.ALTA )
INNER JOIN DW_URG.ESPECIALIDADE e on raw.ALTA_DES_ESPECIALIDADE=e.desc_especialidade
INNER JOIN DW_URG.PACIENTE pa on (raw.SEXO=pa.sexo and raw.DTA_NASCIMENTO=pa.data_nascimento)
INNER JOIN DW_URG.PROVENIENCIA pr on raw.DES_PROVENIENCIA =pr.desc_proveniencia;
```

Figura 6: Query SQL para o povoamento da tabela de factos

Por sua vez, para povoar a tabela de factos foi necessário recorrer tanto às tabelas de dimensão já povoadas como à tabela original. Basicamente foi necessário associar o(s) atributo(s) pertencente(s) à tabela original com o(s) atributo(s) de cada tabela de dimensão e retirar o respetivo ID.

Finalmente, para inserir cada ID, já filtrado, utilizou-se o mesmo esquema de query, referido anteriormente.

### 2.5.2 Com jobs no Talend

Uma vez que o Talend é uma ferramenta (ETL - Extract, Transform, Load), serão aplicados os mesmos fundamentos (extrair do ficheiro .csv, tratar os dados e povoar) só que recorrendo a uma ferramenta diferente.

Assim, para povoar a base de dados procederam-se as seguintes etapas:

- criação de um job povoamento das tabelas de dimensão;
- criação de um job povoamento da tabela de factos.

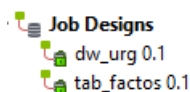


Figura 7: jobs criadas para o povoamento da DW\_URG



## Povoamento das tabelas de dimensão

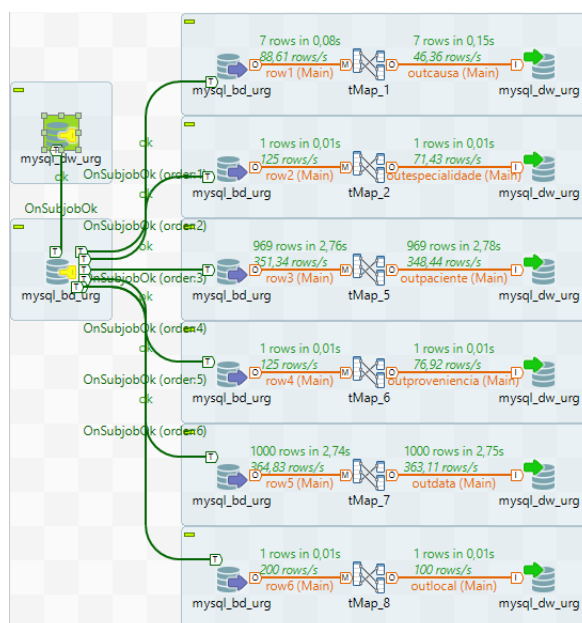


Figura 8: esquema do job "dw\_urg 0.1"

Neste job, utilizaram-se 4 tipos de componentes diferentes:

- **2 tDBConnection** uma (*mysql\_dw\_urg*) para estabelecer uma conexão com a base de dados onde se vão inserir os dados (*dw\_urg*) e outra conexão (*mysql\_bd\_urg*) com a base de dados onde se vão buscar os dados (*bd\_urg*).
- **6 tDBInput** para ler da base de dados já povoada (*bd\_urg*)
- **6 tMap** para mapear os atributos presentes na base de dados (*bd\_urg*) para os atributos respetivos das novas tabelas de dimensão a serem criadas.
- **6 tDBOutput** para inserir os dados nas tabelas de dimensão respetivas da BD (*dw\_urg*)

## Povoamento da tabela de factos

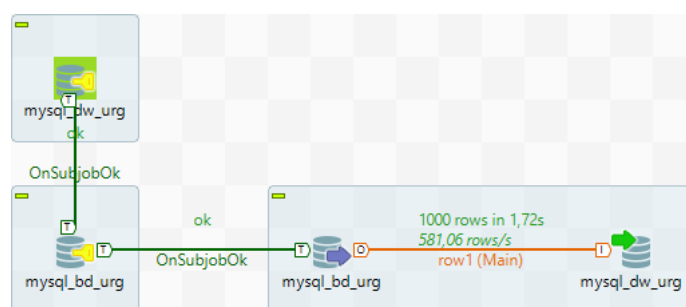


Figura 9: esquema do job "tab\_factos 0.1"

Neste job, utilizaram-se 3 tipos de componentes diferentes:

- **2 *tDBConnection*** uma (*mysql\_dw\_urg*) para estabelecer uma conexão com a base de dados onde se vão inserir os dados (*dw\_urg*) e outra conexão (*mysql\_bd\_urg*) com a base de dados onde se vão buscar os dados (*bd\_urg*).
- ***tDBInput*** para ler da base de dados já povoada (*bd\_urg*)
- ***tDBObject*** para inserir os dados na tabela de factos na BD (*dw\_urg*)

### 2.5.3 Descrever e explicar as vantagens e as desvantagens entre os dois diferentes processos de povoamento utilizados (alíneas (a) e (b))

Para descrever as vantagens e desvantagens dos respectivos processos de povoamento, utilizou-se dois critérios, modo de utilização e performance. Relativamente à performance, um dos aspetos essenciais é saber o tempo de execução para tal acção.

Tempos de execução do povoamento de cada tabela da *dw\_urg* (em segundos):

tabela ( <i>dw_urg</i> )	SQL( <i>script</i> )	Talend( <i>jobs</i> )
causa	0.016s	0.09s
paciente	0.016s	5.06s
proveniencia	0s	0.02s
especialidade	0s	0.01s
data	0.016s	4.94s
local	0s	0.02s
factos	1.094s	1.22s

Tempo total de processamento:

- Em SQL com uma *script* decorreu em 1.142 segundos.
- Com *jobs* no Talend decorreu em 11.36 segundos.

O *know-how* necessário para fazê-la em SQL com uma *script* é maior em relação aos *jobs* no Talend, pois não há nesse caso a necessidade possuir conhecimento de SQL avançado para construção de *queries*. Sobre a utilização, o Talend se destaca com sua usabilidade consequência da sua *user-friendly interface* em comparação a se utilizar de *script* o qual não detêm interface gráfica para auxiliar.

## 2.6 Defina e crie indicadores clínicos e de desempenho com o Microsoft Power BI (ou outro programa similar, por exemplo, Tableau) recorrendo ao Data Warehouse DW\_URG implementado, justificando e descrevendo a relevância e a utilidade de cada um dos indicadores.

Foram criados 6

A Figura 10 representa um resumo geral das informações acerca do hospital. Sua utilidade é proporcionar ao tomador de decisão uma visão holística sobre os recursos e demandas do



Figura 10: Indicador sobre Geral do Hospital

hospital. Dessa forma, o gestor ou tomador de decisão pode fazer questionamentos e nesse intuito foi gerado os demais indicadores a seguir.

Ao observa-se o número de casos de urgência, o gestor do hospital pode se questionar quais as causas das urgências e suas respectivas quantidades?

A Figura 11 é um indicador que tem como propósito responder tal questionamento. Ao visualizar o indicador é possível saber quais as causas mais decorrentes de atendimento na urgência. De posse dessa informação é possível criar estratégias no hospital para um melhor atendimento ou até mesmo elaboração de campanhas junto a população.

Nº Episódios de Urgência por Causa

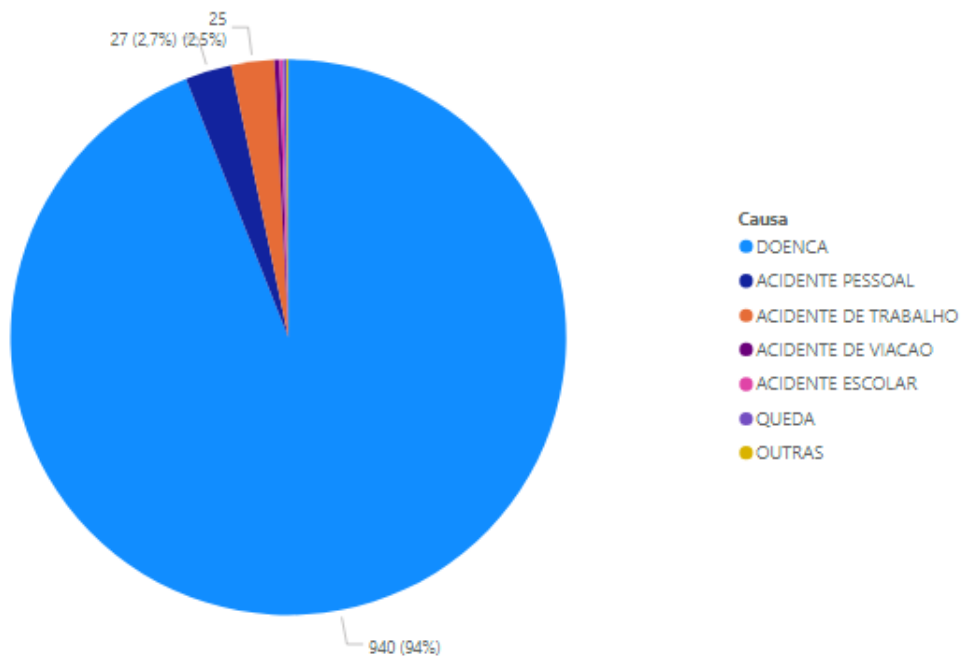


Figura 11: Indicador sobre Episódio por causa

Uma vez tenho conhecimento das causas dos episódios de urgências, pode-se questionar qual especialidade é mais solicitada para o atendimento? A Figura 12 o indicador o qual responde essa questão.

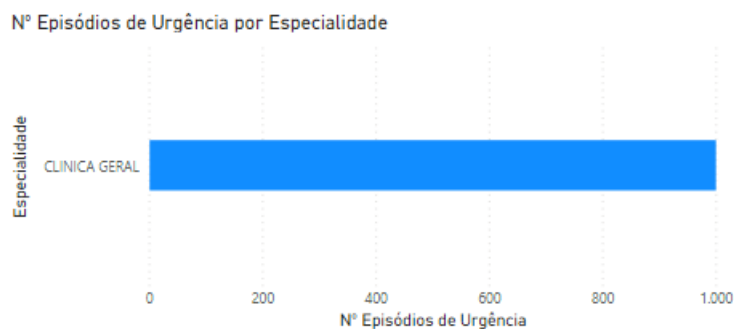


Figura 12: Indicador sobre Episódio por Especialidade

Pelo indicador da Figura 12 nota-se sobrecarga de atendimento em uma especialidade somente, por conseguinte essa informação é útil para saber qual especialidade necessitaria de mais recursos.

A seguir suponhamos que o gestor e ou tomador de decisão do hospital necessitasse saber qual o número dos episódios de urgência por ano? A Figura 13 é indicador o qual se propõe responder a isso.

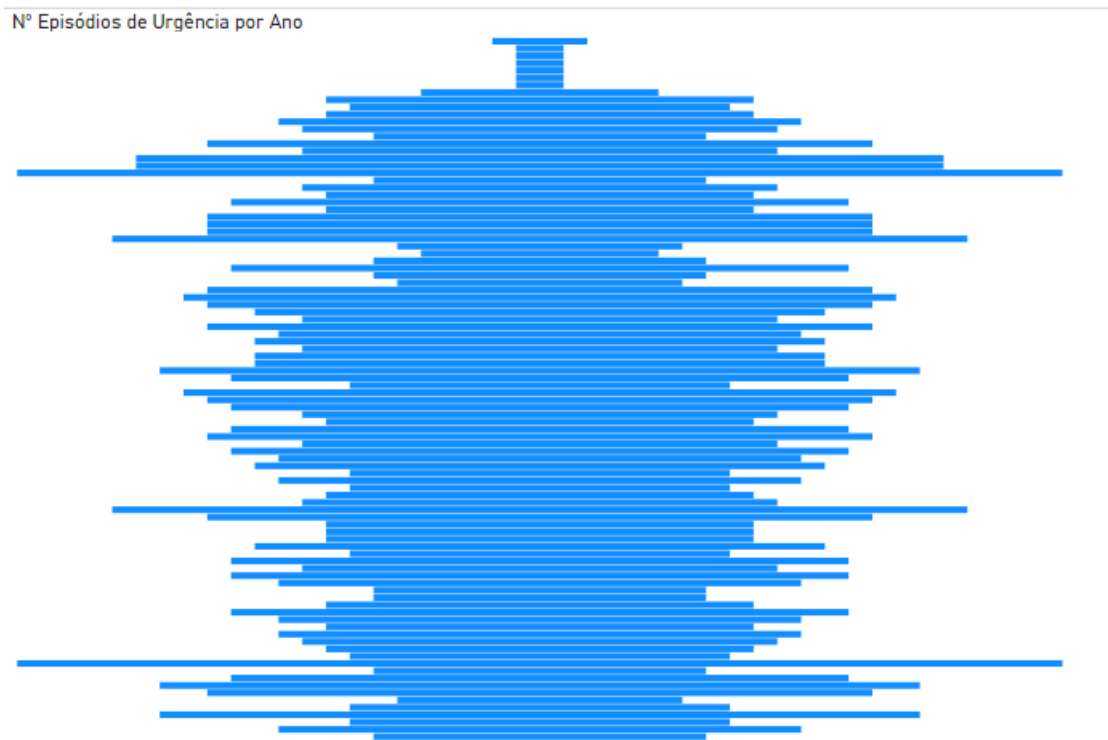


Figura 13: Indicador sobre Episódios por ano

Se for necessário uma análise mais detalhada, para saber qual o número de episódios de urgência decorreram em um determinado dia e quantos pacientes desses episódios foram para pacientes femininos ou masculino, o indicador da Figura 13 não consegue expressar essa informação.

Desta forma foi criado outro indicador ao qual traga essa informação na Figura 14. A qual exibe a distribuição do número de episódios e quais foram de pacientes do género masculino ou feminino.

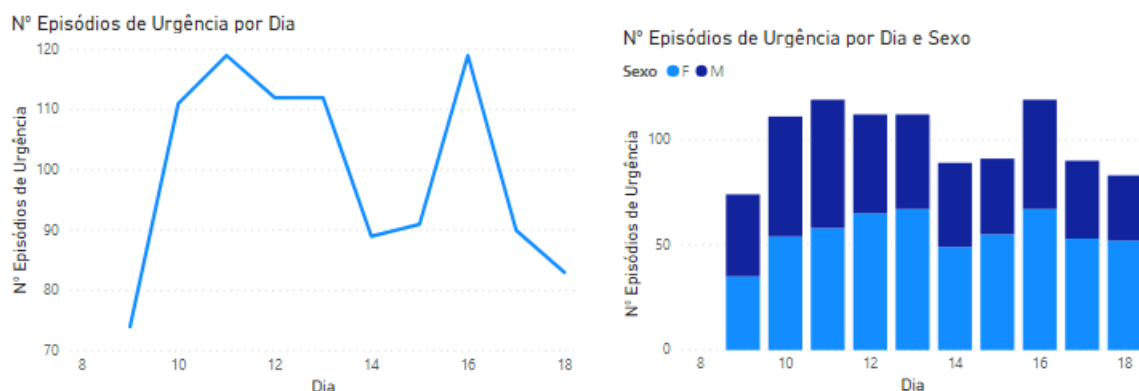


Figura 14: Indicador sobre Episódios por dia

Para saber qual o género predominante dentro do total de episódios ocorridos no hospital

deve-se analisar o número de episódios no qual o paciente em questão era do gênero masculino ou feminino. Já se o gestor e ou tomador de decisão do hospital quiser saber qual do gênero dos pacientes é mais representativo será necessário contabilizar o paciente unicamente. Pois um paciente pode ir mais de uma vez ao hospital e nesse caso contabilizar ele representa dois episódios de urgência, no entanto, só um paciente.

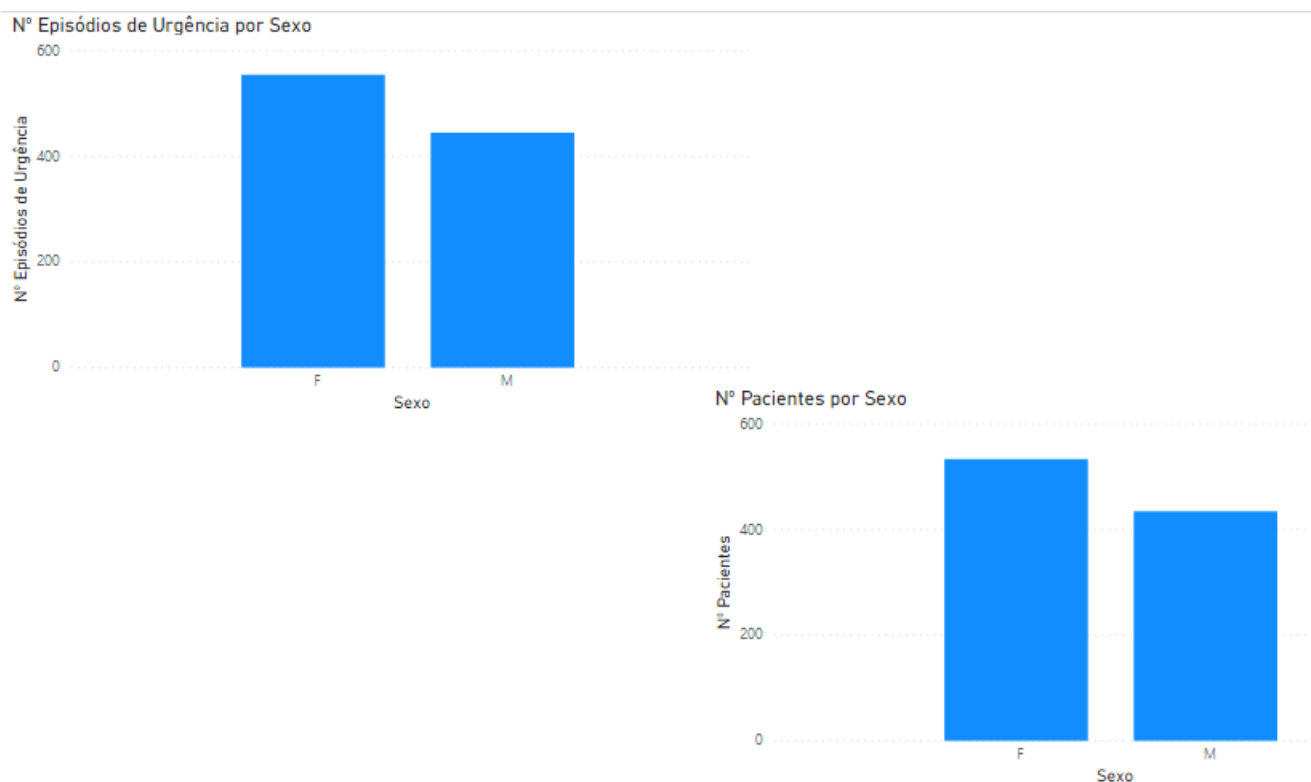


Figura 15: Indicador sobre Género

A Figura 15 demonstra o número de episódios pelo sexo do paciente. De qual modo mostra o número de pacientes por sexo o qual o hospital teve ao realizar todos atendimentos de urgência.

## 2.7 Sugira e descreva as diversas interfaces e as funcionalidade de uma aplicação informática que poderia incluir dashboards com os indicadores clínicos e de desempenho definidos, ou seja, de uma aplicação direcionada às urgência gerais no hospital nacional em questão, bem como das tecnologias que utilizaria para o seu desenvolvimento

Um aplicativo para gestão de hospital. Foi criado um modulo para os gestores e tomadores de decisão do hospital terem acesso aos indicadores e dashboards a respeito das informações do hospital. Em especial os mockups descritos abaixo são referentes ao acesso ao modulo para os gestores e tomadores de decisão.

Na Figura 20 é descrito a tela inicial do aplicativo, de igual forma é apresentado a Tela de Login do mesmo. Após inserir os dados para aceder ao aplicativo. O aplicativo irá carregar

acesso aos dados e os dashboards para os gestores terem acesso de forma instantânea.

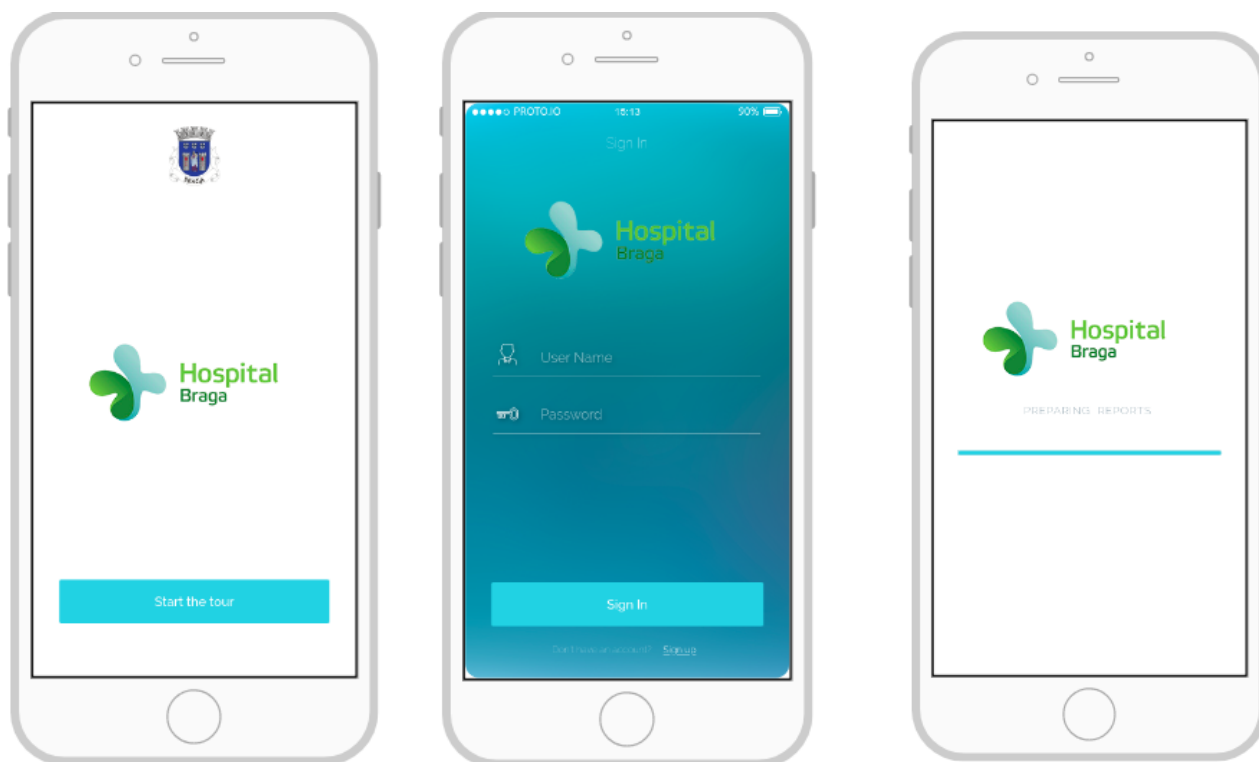


Figura 16: Tela Inicial, Login e Load

O utilizador será levado a tela para inserir os dados de usuário para aceder aos indicadores e dashboards do hospital. Logo após terá acesso aos dashboards e indicadores do hospital. Esse aplicativo tem como propósito dar suporte e manter informado os gestores diariamente sem a necessidade desses estarem fisicamente no hospital.

Logo após aceder ao aplicativo os gestores e tomadores de decisão terão acesso aos dashboards contendo os indicadores do hospital. Foram gerados 3 dashboards contendo os indicadores respectivos relacionados a específicos ao respectivo dashboard.

O primeiro dashboard, na Figura 17, contém indicadores em relação ao género do paciente. Trazendo a relação de cada género com outras variáveis a saber número de pacientes atendidos por exemplo.

O segundo dashboard, na Figura 18, refere-se a indicadores relacionados directamente a relação temporal do atendimento ao pacientes e o género dos mesmos. Trazendo conhecimento do dia a dia do hospital em relação ao atendimento como número de episódios de emergências.

O terceiro dashboard, na Figura 19, refere-se a indicadores relacionados a especialidades. Verificando qual especialidade está demandando mais recurso do hospital. Assim podendo administrar os recursos da forma mais efectiva.

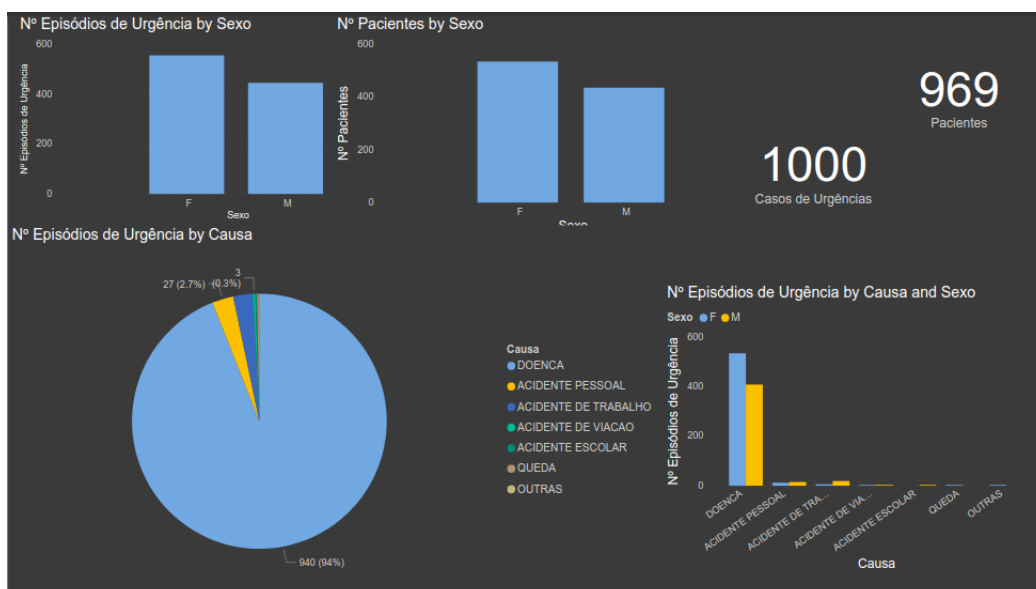


Figura 17: DashBoard 1

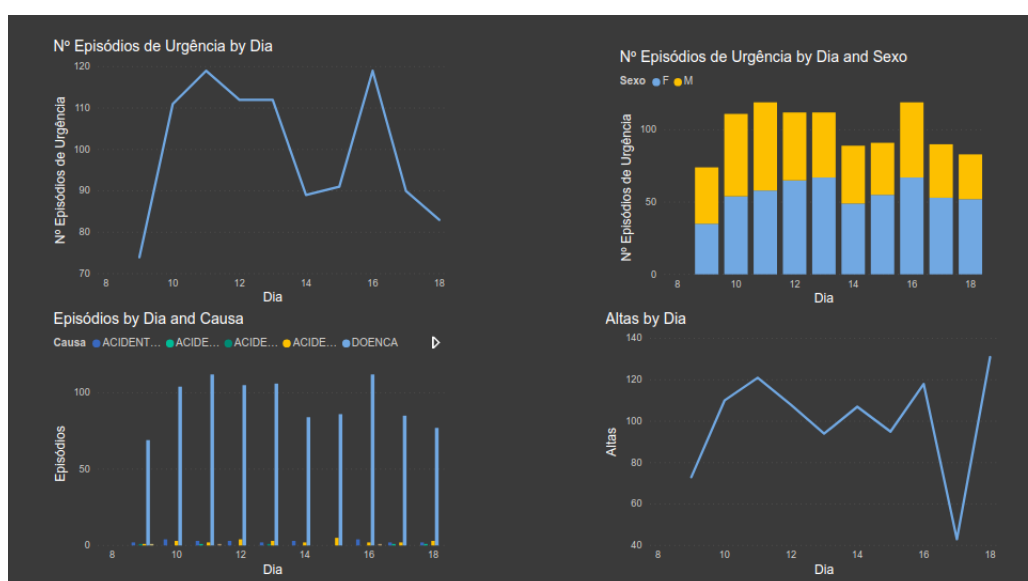


Figura 18: DashBoard 2

Com esses dashboards descritos acima, os gestores podem ser acessíveis por meio dos seus dispositivos mobile. Desta forma, consegue trazer agilidade e otimização do tempo entre a tomada de decisão e o conhecimento da situação a qual necessita da decisão.



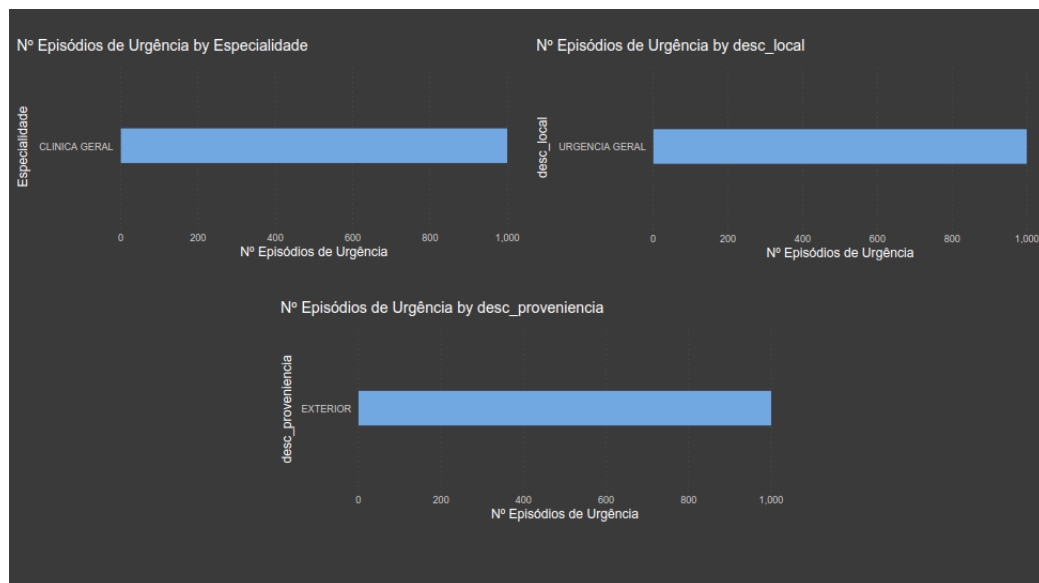


Figura 19: DashBoard 3

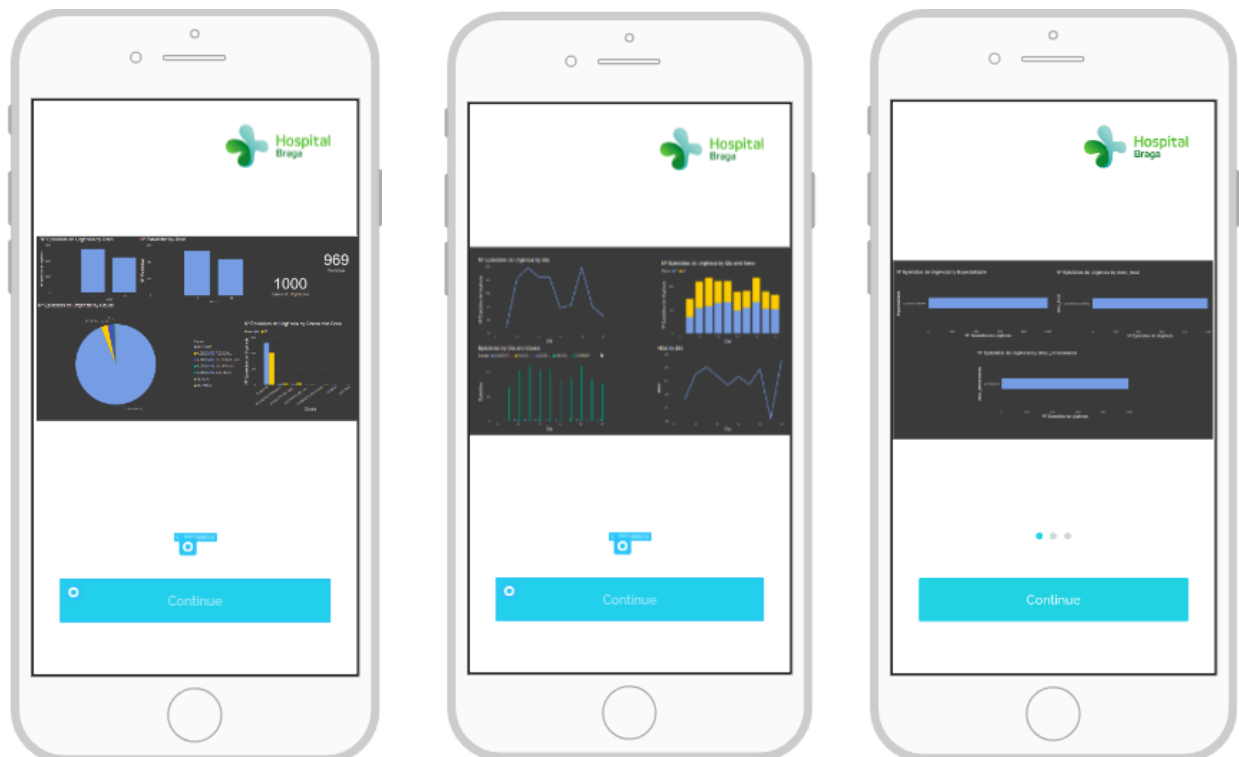


Figura 20: Telas dos Dashboards