



Application Development

Table of Contents

1	Instructions.....	3
1.1	Introduction.....	3
1.2	Rules	3
1.3	Evaluation	3
1.4	Deliverables	4
1.5	Tools	4
1.6	Documentation.....	4
2	Project Introduction	5
3	Application Structure.....	5
4	Layout and Design Specifications	5
5	Tasks	6
5.1	Database	6
5.1.1	Sample Data.....	6
5.2	Dashboard	7
5.2.1	Menu Bar	8
5.2.2	Dashboard Controls	8
5.3	Management	9
5.3.1	Manage Tournaments	9
5.3.2	Manage Teams	12
5.4	Manage Execution	15
5.4.1	Allocate Teams to Groups	16
5.4.2	Manage Games.....	17
5.4.3	Group Stage Games Generation.....	19
5.4.4	Round of 16 Games Generation	19
5.4.5	Quarter-Finals, Semi-Finals and Final Game Generation	21
5.5	Edit Game	22
5.5.1	Starting Grid.....	22
5.5.2	Start game	23
5.5.3	Events	24
5.5.4	Finish game.....	25
5.6	Tournament Overview.....	26
6	Mark Summary	28

1 Instructions

1.1 Introduction

This document describes the tasks to be performed for the Swiss ICT Skills Championship 2016 in the category Application Development. We advise you to read this document carefully and to have an overview of the provided media like attached images and text. In order to complete the tasks, you are free to use whichever of the provided tools you like.

You could find yourself in the situation that there is not enough time to complete all remaining tasks. You are allowed to skip the implementation of certain tasks in favor for others. However, it is expected that you deliver a smoothly running application at the end of the competition to facilitate the assessment of your work.

1.2 Rules

Please comply with the following rules throughout the competition:

- Accessing the internet is not allowed
- The use of any communication device is prohibited (mobile, tablet, etc.)
- The use of your own storage devices is prohibited (USB sticks / hard disks, etc.)
- Communication between other candidates is prohibited
- You are not allowed to bring books, notes, etc. to the competition

The non-compliance with these rules may result in the disqualification from the competition.

1.3 Evaluation

Each task is rated by testing the implementation of the required functionality. In addition, the following criteria applies to all tasks and has to match your solution:

- All written code must be properly formatted, indented and commented;
- SQL queries must be protected from any possible injections;
- No error should occur / proper exception handling is implemented;

1.4 Deliverables

All deliverables must be submitted in a zip file named "AppDev_Name_FirstName.zip" and it must be placed at the folder location indicated by the supervisor. The deliverables are **executable files**, **source code** and the **database** (db-files, backup or SQL-export). Please also fill out the requested information in the file *additionalInfo.txt* and include it in the zip file.

At the end of the competition you must also submit all documents and notes to the supervisor.

Only the contents of the zip file will be considered for the evaluation of the work. Only electronically delivered notes (*readme.txt or readme.rtf*) will be considered for evaluation. Notes on this document will not be used for the evaluation.

1.5 Tools

The tools available to perform the tasks in this document are:

- JDK 8
- NetBeans 8.1
- Eclipse Neon
 - WindowBuilder Pro
 - e(fx)clipse
- .NET Framework 4.6
- Microsoft Visual Studio Community 2015
- Microsoft SQL Server 2014 Express
 - JDBC Connector
 - Management Studio
- MySQL 5.7
 - JDBC / .NET Connector
 - Workbench
- LibreOffice 5.1
- Adobe Reader DC
- Notepad++
- Windows 10

1.6 Documentation

To perform the tasks of this project the following manuals are available:

- T-SQL
- MySQL
- .NET (Visual Studio)
- JDK
- JavaFX

2 Project Introduction



EUFA, the European Football Association, decided to renew their software tools which they need to manage the European football championships. You have been hired to develop one of the most important pieces of this system, the application where all the teams and matches are managed and the events of a game (like goals, yellow & red cards, substitutions, etc.) are recorded.

3 Application Structure

The graph below shows the navigation flow in the application. Each node represents a window. The requirements for each window will be mentioned later in this document.

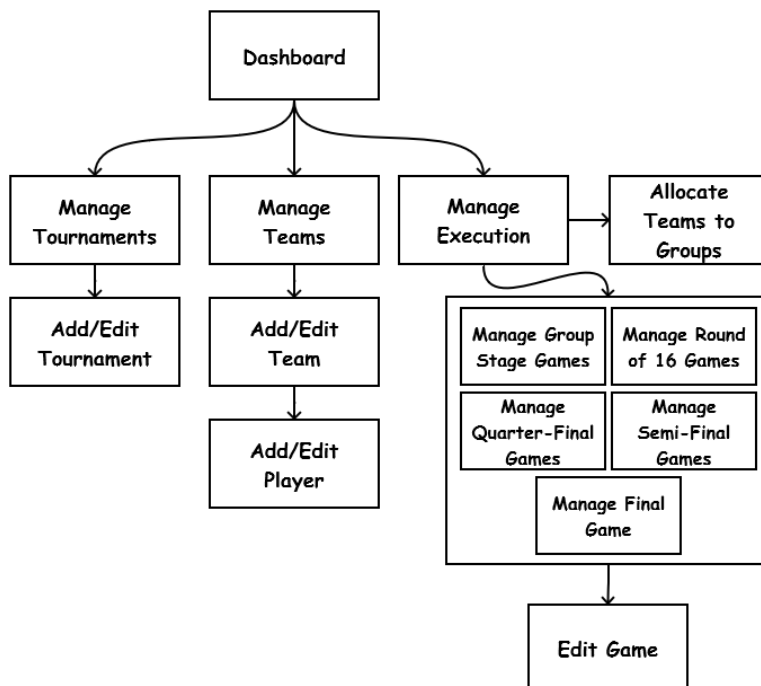


Figure 1: Application structure

4 Layout and Design Specifications

All components of this system must have a consistent look and feel that conforms to EUFA's corporate design. The style guide is provided as a separate handout.

Ensure all components comply in the following areas:

- Color scheme
- Borders and shading
- Font usage
- Header layout and design (incl. logo placement)
- Grouping of elements (into logical categories)
- Use of appropriate controls (e.g. dropdown lists to display lookup values from database)
- Layout/alignment of elements (labels, inputs, etc.)
- Overall professionalism – must be clear, concise and easy to use

5 Tasks

5.1 Database

All data of your product, apart from small configuration files, should be saved in a database. Please use the prepared database (name: *ictskills2016*) on the MySQL or MS-SQL server running on your virtual machine. You'll find the connection information on the desktop (*Database User and Password.txt*) and sample code, that shows how to connect to the different databases, can be found in the folder *Examples*.

You are free on how you design your database tables to fulfill the tasks in the following chapters, but you should stick to this rules:

- The database should be designed in 3rd normal form to ensure referential integrity and avoid duplicated data
- Use meaningful names for all database object (tables, columns, indexes, ...)
- Use a consistent naming scheme for your database objects (e.g. same prefixes for same type of objects)
- Use suitable data types and create an appropriate primary key on each table
- Create indexes where they are useful
- Create foreign key constraints to keep the data consistent

5.1.1 Sample Data

To facilitate your job, UEFA provides various sample data of tournaments, teams and players, which are stored as CSV files in the data directory.

File	Description
Tournaments.csv	All tournaments which took place within the last two years and their start and end dates
Teams.csv	All registered teams of the world with their country code, flag image reference and region
Players.csv	All players of the European Championship 2016 with their name, shirt number, date of birth and team membership

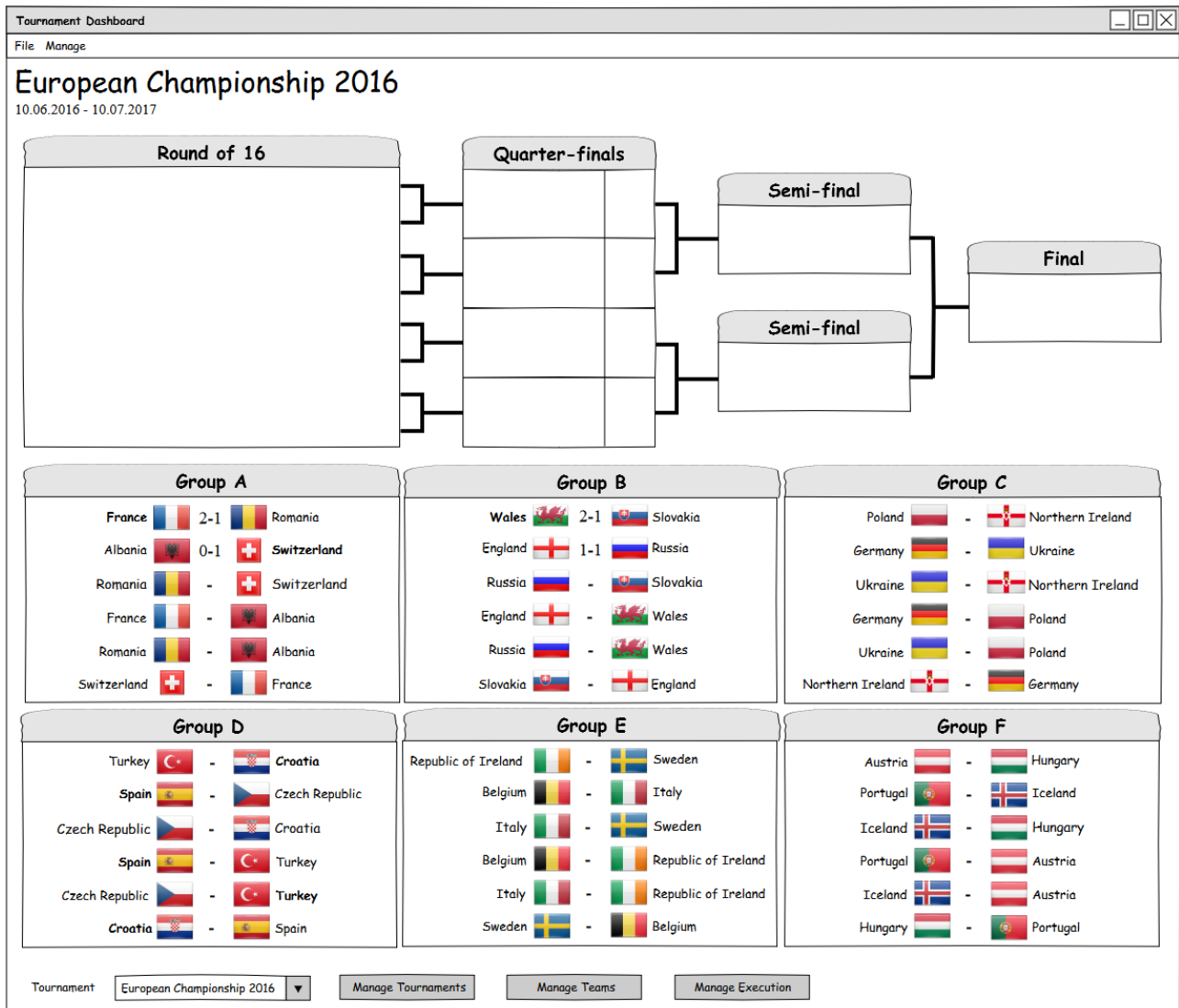
- Import all these files into the database in order to ensure the functionality of your application with real data
- The data structure of all these files can be changed in order to meet the requirements of chapter 5.1
- Tip: The sample data contains only players of countries which participated at the European Championships 2016. So it's recommended to work with the following countries to save time:

○ Albania	○ Hungary	○ Russia
○ Austria	○ Iceland	○ Slovakia
○ Belgium	○ Italy	○ Spain
○ Croatia	○ Northern Ireland	○ Sweden
○ Czech Republic	○ Poland	○ Switzerland
○ England	○ Portugal	○ Turkey
○ France	○ Republic of Ireland	○ Ukraine
○ Germany	○ Romania	○ Wales

5.2 Dashboard

The dashboard provides access to all parts of UEFA's tournament management application and shows the tournament overview of the current tournament.

- Create the *Dashboard* window as shown in the mockup below and as described in the following chapters.
- This window should be shown when the program is started.



Mockup 1: This example shows how the dashboard would have looked like after the 2nd day of the European Championship 2016

5.2.1 Menu Bar

Implement a menu bar with the following menu items:

File	Action
Exit	Close the application

Manage	Action
Manage Tournaments	Open the <i>Manage Tournaments</i> window (more instructions in chapter 5.3.1)
Manage Teams	Open the <i>Manage Teams</i> window (more instructions in chapter 5.3.2)
Manage Execution	Open the <i>Manage Execution</i> window (more instructions in chapter 5.4)

5.2.2 Dashboard Controls

Place the following controls on the Dashboard window.

Control	Description
Tournament title	Name of the currently selected tournament. The tournament can be switched by using the tournament combobox on the bottom of the screen.
Tournament duration	Displays the start date and end date of the currently selected tournament below the title.
Tournament Overview	Displays all match results and planned matches (more instructions in chapter 5.6)
Tournament combobox	Switches the current tournament that's displayed on the dashboard. Combobox items contains the name and the year of the tournament and are sorted by start date from nearest to oldest. Default selection: Tournament with the nearest start date
Manage Tournaments button	Open the <i>Manage Tournaments</i> window (more instructions in chapter 5.3.1)
Manage Teams button	Open the <i>Manage Teams</i> window (more instructions in chapter 5.3.2)
Manage Execution button	Open the <i>Manage Execution</i> window (more instructions in 5.4)

5.3 Management

5.3.1 Manage Tournaments

- Create the *Manage Tournaments* window as shown below. It is accessible over the corresponding button item on the dashboard.

ID	Tournament	Date
56	World Cup	12.06.14 - 13.07.14
89	AFC Asian Cup	09.01.15 - 31.01.15
87	Africa Cup of Nations	17.01.15 - 08.02.15
164	Copa América	03.06.16 - 26.06.16
267	European Championship	10.06.16 - 10.07.16

Buttons: Add, Edit, Delete, Close

- Add a table control to the window. All existing tournaments should be displayed in this table with the following visible columns and sorted by *start date*.

Column	Content
ID	Tournament ID
Tournament	Tournament name
Date	[Start date] – [End date] (Date format: DD.MM.YY)

- A user should be able to add a new tournament by clicking on the *Add* button.
- A user should be able to edit and delete an existing tournament by clicking on the corresponding buttons:
 - An error dialog should be shown, if the user clicks *Edit* or *Delete* and no tournament is selected.
 - A confirmation prompt should be shown after the user clicks *Delete*, to ask if the user really wants to delete the selected tournament and all associated matches.
- A user should be able to close the dialog by clicking on the *Close* button.

5.3.1.1 Add/Edit Tournaments

- Users should be able to add a new tournament and to edit an existing one by clicking on one of the corresponding buttons in the *Manage Tournaments* window.
- Create the *Add/Edit Tournament* window as shown below.






- The attribute *ID* has to be generated automatically and is not shown on the window.
- The fields *Tournament name*, *Start date*, *End date* are mandatory. The *Save & Close* button should be deactivated if one or more of these fields are empty or invalid.
- Use a date picker for the controls *Start date* and *End date* (date format: DD.MM.YY)
- Implement the following restrictions:
 - *Start date* must be before or equal to the end date.
 - *End date* must be after or equal to the start date.
- Improve the usability of the window:
 - *Start date*: Set the current date as default start date for adding a new tournament.
 - *End date*: Set current date + 1 month as default end date for adding a new tournament.
- Add a table control to the window. All existing teams should be displayed in this table with the following visible columns and sorted by *team name*.

Column	Content
Participation	A checkbox should be displayed which toggles the participation of a team if the user clicks on it.
Team	Team name

- A user should be able to select exactly 24 teams of the list. The current number of selected teams should be shown in a label below *Participating teams*, e.g. "(2/24 selected)".
- An error dialog should be shown, if the user has already selected 24 teams and tries to select more teams.
- An error dialog should be shown, if the user clicks *Save & Close* and less than 24 teams are selected.
- Improve the usability of the table form to handle the fact that there might be many teams:
 - Allow users to filter the list of teams by region.
 - The filter *All* is selected by default and so that all teams are shown in the table form.
 - By selecting another filter only teams of the selected region should be shown in the table form.
- A user should be able to save and close the dialog by clicking on the *Save & Close* button.
- A user should be able to close the dialog without saving by clicking on the *Close* button.

5.3.2 Manage Teams

- Create the *Manage Teams* window as shown below. It is accessible over the corresponding button on the dashboard.

ID	Flag	Team	Code
751		Angola	ANG
709		Bolivia	BOL
671		Colombia	COL
432		Moldova	MDA
427		Poland	POL

- Add a table control to the window. All existing teams should be displayed in this table with the following visible columns and sorted by *team name*.

Column	Content
ID	Team ID
Flag	Image of the team flag, use default.png if no flag of the team exists
Team	Team name
Code	Country code (3 characters)

- A user should be able to add a new team by clicking the *Add* button.
- A user should be able to edit and delete an existing team by clicking on the corresponding buttons:
 - An error dialog should be shown, if the user clicks *Edit* or *Delete* and no team is selected.
 - An error dialog should be shown, if the user tries to delete a team which has already participated in a tournament.
 - A confirmation prompt should be shown after the user clicks *Delete*, to ask if the user really wants to delete the selected team and all associated players.
- There are a lot of teams. Improve the usability of the table form:
 - Allow users to filter the list of teams by a search string for the team name. Only the teams which contains the search string in its name should be shown in the table form.
- A user should be able to close the dialog by clicking on the *Close* button.

5.3.2.1 Add/Edit Team

- Users should be able to add a new team and to edit an existing one by clicking on one of the corresponding buttons in the *Manage Teams* window.
- Create the *Add/Edit Team* window as shown below.

Team name:

Country Code: Region:

Players (23)

Number	Name	Position
1	Sommer Yann	Goalkeeper
2	Lichtsteiner Stephan	Defender
3	Moubandje François	Defender
4	Elvedi Nico	Defender
5	von Bergen Steve	Defender
6	Lang Michael	Defender
7	Embolo Breel	Forward
8	Frei Fabian	Midfield
9	Seferović Haris	Forward

- The attribute *ID* has to be generated automatically and is not shown on the window.
- The fields *Team name* and *Country code* are mandatory. The *Save & Close* button should be deactivated if one or more of these fields are empty or invalid.
- A user should be able to select a flag for the team. Implement a “Flag Selector” for the user to choose one of the given flag images (see the provided files).
- Flag is not a mandatory field. Please use the default flag (*default.png*) in the whole application if no flag is selected.
- Use a combobox to select one of the following regions: Africa, America, Asia, Europe or Oceania.
- Implement the following restrictions:
 - *Team name* and *Country Code* have to be unique.
 - *Country Code* has to be exactly three characters long.
- Add a table control to the window. All existing players of this team should be displayed in this table form with the following visible columns and sorted by number.

Property	Content
Number	Shirt number
Name	Last name and First name
Position	Goalkeeper, Defender, Midfield, Forward

- A user should be able to add a new player to the list by clicking on the *Add* button.
- A user should be able to edit and delete an existing player by clicking on the corresponding buttons:
 - An error dialog should be shown, if the user clicks *Edit* or *Delete* and no player is selected.
 - A confirmation prompt should be shown after the user clicks *Delete*, to ask if the user really wants to delete the selected player
- The current number of existing players should be shown in the label: *Players (23)*.
- A user should be able to save and close the dialog by clicking on the *Save & Close* button.
- A user should be able to close the dialog without saving by clicking on the *Close* button.

5.3.2.2 Add/Edit Player

- Users should be able to add a new player and to edit existing ones by clicking one of the corresponding buttons in the *Add/Edit Team* window.
- Create the *Add/Edit Player* window as shown below.

- The attribute *ID* has to be generated automatically and is not shown on the window.
- All fields except *first name* are mandatory. The *Save & Close* button should be deactivated if one or more of the fields are empty or invalid.
- Use a combobox to select one of the following *positions*: Goalkeeper, Defender, Midfield, Forward.
- The *shirt number* has to be between 1 and 99. The *shirt number* is unique per team and an error message has to be displayed if a shirt number is assigned to more than one player.
- Use a date picker for the control *Date of birth*. (date format: DD.MM.YY).
- A new player should automatically be assigned to the selected team.
- A user should be able to save and close the dialog by clicking on the *Save & Close* button.
- A user should be able to close the dialog without saving by clicking on the *Close* button.

5.4 Manage Execution

- Create the *Manage Execution* window as shown below. It is accessible over the corresponding button item on the dashboard.

Manage Execution	
	Teams allocated
Allocate Teams to Groups...	Yes
	Finished
Manage Group Stage Games...	Yes
Manage Round of 16 Games...	No
Manage Quarter-Final Games...	No
Manage Semi-Final Games...	No
Manage Final Game...	No
Close	

- The *Teams allocated* column displays *Yes*, if all teams are allocated to groups, *No* if not.
- The *finished* column displays *Yes*, if all games are finished, *No* when any game of the corresponding game phase is not yet finished.
- Allocate Teams to Groups...* opens the *Allocate Teams to Groups* Dialog (see chapter 5.4.1).
- Manage Group Stage Games...* opens the *Manage Group Stage Games* dialog (see chapter 5.4.2).
- Manage Round of 16 Games...* opens the *Manage Round of 16 Games* dialog (see chapter 5.4.2).
- Manage Quarter-Final Games...* opens the *Manage Quarter-Final Games* dialog (see chapter 5.4.2).
- Manage Semi-Final Games...* opens the *Manage Semi-Final Games* dialog (see chapter 5.4.2).
- Manage Final Game...* opens the *Manage Final Game* dialog (see chapter 5.4.2).
- A user should be able to close the dialog by clicking on the *Close* button.

5.4.1 Allocate Teams to Groups

Create the following dialog

Allocate Teams to Groups

Allocate randomly

Group A

Shortname	Team
A1	France
A2	Romania
A3	Albania
A4	Switzerland

Group B

Shortname	Team
B1	
B2	
B3	
B4	

Group C

Shortname	Team
C1	
C2	
C3	
C4	

Teams

Belgium
Germany
Croatia
England
Iceland
Italy
...

Group D

Shortname	Team
D1	
D2	
D3	
D4	

Group E

Shortname	Team
E1	
E2	
E3	
E4	

Group F

Shortname	Team
F1	
F2	
F3	
F4	

Save & Close

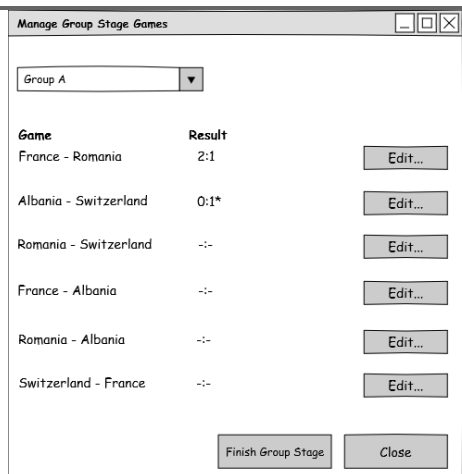

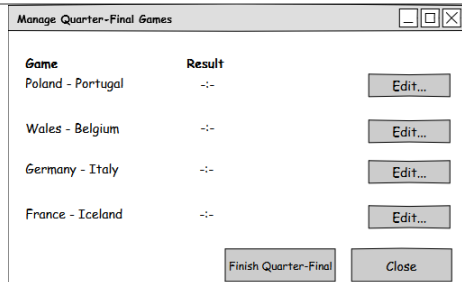
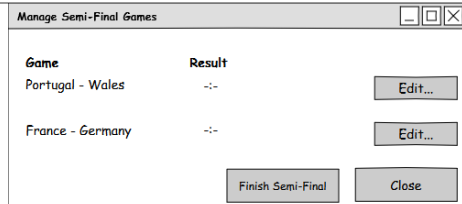
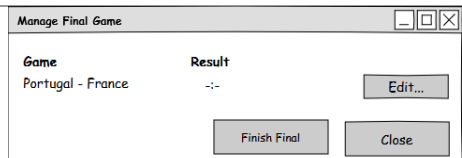
Close

- With this dialog, it is possible to allocate the teams that are participating in the tournament to groups. The groups A – F are predefined. A tournament has always six groups. The teams can be placed into a group with a drag and drop operation.
- The *Teams* list shows all participating teams, sorted alphabetically in ascending order
- By clicking the button *Allocate randomly*, the teams are allocated to the groups in a random manner
- A team can only be allocated to one group
- It must be possible to reorder the teams within the groups
- It must be possible to change the group of a team
- If a team is placed in a group which already contains four teams, an error message must be shown
- *Save & Close* saves the allocation. If any team is not allocated to a group, an error message must be shown and the dialog remains open
- *Save & Close* generates the Group Stage games (see chapter 5.4.3)
- Once the Group Stage games are generated, the allocation cannot be modified anymore
- *Close* closes the dialog without a save operation

5.4.2 Manage Games

Create the dialogs to manage all the games in the tournament. Please check the table on the next page for additional information about the dialogs and for the mockups.

- If the defined precondition of a dialog is not met, display an error message and close the dialog
- Additional requirements for *the Manage Group Stage Games* dialog
 - A combobox lists all available groups in alphabetical order
 - When the dialog opens, Group A is preselected and the corresponding games with results and *Edit...* Buttons are displayed.
 - When the selected group changes, then the corresponding games are displayed
- Column *Result*:
 - If a game is finished, then the final result is shown
 - If a game is running but not finished, then the current result followed by a star (*) is shown
 - If a game has not yet started, then "-:-" is shown
- When the *Edit...* button is clicked, then the *Edit Game* dialog opens for the corresponding game, as described in chapter 5.5
- If not every game of a dialog is finished and the *Finish X* button (where X stands for the dialog's game phase) is clicked, then an error message is shown.
- If all games are finished and the *Finish X* button is clicked, the game phase is finished and the logic to generate the *next* games is executed.
- Once the game phase was successfully finished (by clicking the *Finish X* button), the games cannot be edited anymore

Dialog	Precondition	Logic to generate the games for this dialog	Mockup
Manage Group Stage games	Every team of the tournament must be allocated to a group	See chapter 5.4.3 Group Stage Games Generation	
Manage Round of 16 Games	All group stage games must be finished	See chapter 5.4.4 Round of 16 Games Generation	
Manage Quarter-Final Games	All round of 16 games must be finished	See chapter 5.4.5 Quarter-Finals, Semi-Finals and Final Game Generation	
Manage Semi-Final Games	All quarter-final games must be finished	See chapter 5.4.5 Quarter-Finals, Semi-Finals and Final Game Generation	
Manage Final Game	All semi-final games must be finished	See chapter 5.4.5 Quarter-Finals, Semi-Finals and Final Game Generation	

5.4.3 Group Stage Games Generation

- For every group A - F generate the following six games (for example for Group A):
 - A1 : A2
 - A3 : A4
 - A2 : A4
 - A1 : A3
 - A4 : A1
 - A2 : A3

5.4.4 Round of 16 Games Generation

- The teams ranked first and second of a group are qualified for the round of 16.
- Additionally, the best four teams ranked third are qualified
- To evaluate the ranking in a group or the best third-ranked teams use the following factors (ordered by significance)
 - Number of points from the group stage
 - Goal difference (goals scored – goals conceded)
 - Goals scored
 - If all of the above factors are same then use a random number generator to select a team
- Generate the following games for the round of 16:

Game	Home team		Away team
AF1	Second place group A	:	Second place group C
AF2	Winner of group B	:	Third place group A / C / D
AF3	Winner of group D	:	Third place group B / E / F
AF4	Winner of group A	:	Third place group C / D / E
AF5	Winner of group C	:	Third place group A / B / F
AF6	Winner of group F	:	Second place group E
AF7	Winner of group E	:	Second place group D
AF8	Second place group B	:	Second place group F

- Use the following table to determine the teams for the games AF4, AF2, AF5 and AF3:

Best ranked third-placed teams in groups	AF4: Winner of group A : Third place group	AF2: Winner of group B : Third place group	AF5: Winner of group C : Third place group	AF3: Winner of group D : Third place group
A, B, C, D	C	D	A	B
A, B, C, E	C	A	B	E
A, B, C, F	C	A	B	F
A, B, D, E	D	A	B	E
A, B, D, F	D	A	B	F
A, B, E, F	E	A	B	F
A, C, D, E	C	D	A	E
A, C, D, F	C	D	A	F
A, C, E, F	C	A	F	E
A, D, E, F	D	A	F	E

B, C, D, E	C	D	B	E
B, C, D, F	C	D	B	F
B, C, E, F	E	C	B	F
B, D, E, F	E	D	B	F
C, D, E, F	C	D	F	E

Please look at this example of the European Championships 2016 if you don't understand how the assignment of the best four third-ranked teams works:

- The following table contains all teams which ranked third in their group at the European Championships 2016:

Position	Group	Team	Points	Goals scored	Goals conceded
1	B	Slovakia	4	3	3
2	E	Republic of Ireland	4	2	4
3	F	Portugal	3	4	4
4	C	Northern Ireland	3	2	2
5	D	Turkey	3	2	4
6	A	Albania	3	1	3

- According to the rules above the best four third-ranked teams are Slovakia (group B), Republic of Ireland (group E), Portugal (group F) and Northern Ireland (group C).
- So now you have to check the table above and look for the row which contains "B, C, E, F" in the first column.

Best ranked third-placed teams in groups	AF4: Winner of group A : Third place group	AF2: Winner of group B : Third place group	AF5: Winner of group C : Third place group	AF3: Winner of group D : Third place group
B, C, E, F	E	C	B	F

- According to this row the following games will take place in the round of 16:

Game	Home team		Away team
AF1	Second place group A	:	Second place group C
AF2	Winner of group B	:	Northern Ireland (Third place group C)
AF3	Winner of group D	:	Portugal (Third place group F)
AF4	Winner of group A	:	Republic of Ireland (Third place group E)
AF5	Winner of group C	:	Slovakia (Third place group B)
AF6	Winner of group F	:	Second place group E
AF7	Winner of group E	:	Second place group D
AF8	Second place group B	:	Second place group F

5.4.5 Quarter-Finals, Semi-Finals and Final Game Generation

- The winner of the match is qualified for the next round (or wins the tournament if it is the final game)
- Generate the following games for the Quarter-Finals:

Game	Home team		Away team
QF1	Winner of AF1	:	Winner of AF3
QF2	Winner of AF2	:	Winner of AF6
QF3	Winner of AF5	:	Winner of AF7
QF4	Winner of AF4	:	Winner of AF8

- Generate the following games for the Semi-Finals:

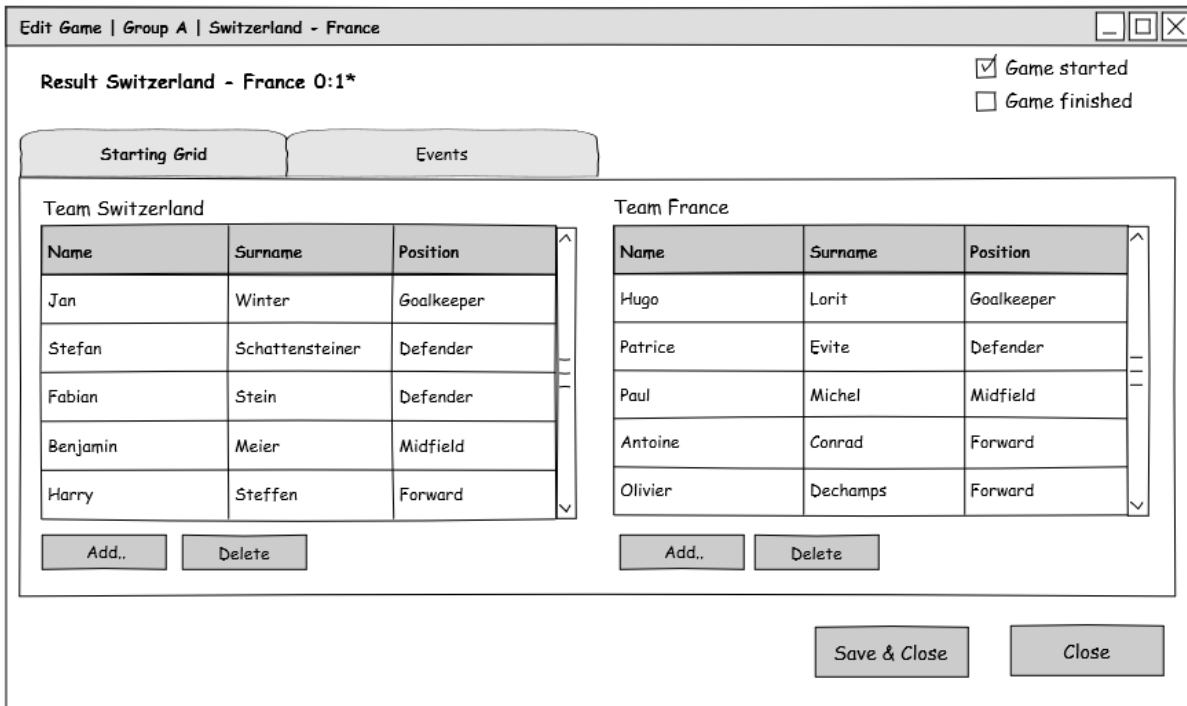
Game	Home team		Away team
SF1	Winner of QF1	:	Winner of QF2
SF2	Winner of QF3	:	Winner of QF4

- Generate the following game for the Final:

Game	Home team		Away team
TF1	Winner of SF1	:	Winner of SF2

5.5 Edit Game

Create the following dialog



Edit Game | Group A | Switzerland - France

Result Switzerland - France 0:1*

☒ Game started
☐ Game finished

Starting Grid Events

Team Switzerland

Name	Surname	Position
Jan	Winter	Goalkeeper
Stefan	Schattensteiner	Defender
Fabian	Stein	Defender
Benjamin	Meier	Midfield
Harry	Steffen	Forward

Add.. Delete

Team France

Name	Surname	Position
Hugo	Lorit	Goalkeeper
Patrice	Evite	Defender
Paul	Michel	Midfield
Antoine	Conrad	Forward
Olivier	Dechamps	Forward

Add.. Delete

Save & Close Close

- Show the result at the top according to the following description
 - If a game has not yet started, then "-:-" is shown
 - If a game is running but not finished, then the current result followed by a star (*) is shown
 - If a game is finished, then the final result is shown
- The dialog has two tabs, *Starting Grid* and *Events*
- *Save & Close* saves the Game
- *Close* closes the dialog without a save operation

5.5.1 Starting Grid

Add the following tab to the dialog

- The *Starting Grid* tab allows to manage the starting grid of each team
- If the game is not started, then the *Starting Grid* tab is preselected, when the dialog is opened
- The *Starting Grid* table, which shows the players of each team, is sorted the following way
 - First sorting column: *Position* in this specific order Goalkeeper, Defender, Midfield, Forward
 - Second sorting column: *Surname* in alphabetical order
- The titles of both tables must contain the related team name (e.g. Team Switzerland)
- For every team the *Starting Grid* must contain 11 players, where exactly one of these players must be a Goalkeeper. Otherwise a game cannot be started.
- *Add...* opens the *Add Player to Starting Grid* dialog, described in chapter 5.5.1.1
- An error dialog should be shown, if the user clicks *Delete* and no player is selected
- *Delete* opens a Message Box where the user is asked, if the selected player should be deleted
 - If the user clicks yes, then the row is deleted from the *Starting Grid* table
 - If the user clicks no, then nothing happens

5.5.1.1 Add Player to Starting Grid

Create the following dialog

- The *player* dropdown box shows the players of the team in alphabetical order, which are not already added to the *Starting Grid*
- The *position* dropdown box shows the following positions in this order: Goalkeeper, Defender, Midfield, Forward
- The default value of the *position* dropdown box should be the regular position of the player (as defined in chapter 5.3.2.2), but e.g. a regular Midfielder can also play as a Forward in a game.
- *OK* adds the Player
- *Cancel* closes the dialog without any change

5.5.2 Start game

- The user can set the game state to started by clicking the *Game started* checkbox
- Before a game can be started, the starting grids of both teams must be completely recorded (11 players, including exactly one Goalkeeper)
- When the checkbox *Game started* is clicked and the starting grids are not completely recorded, an error message should be displayed.
- A confirmation dialog should be displayed where the user has the possibility to cancel the action
- Once the game state has successfully been changed to *game started*, the editing of the starting grid should be prevented.

5.5.3 Events

Add the *Events* tab to the dialog.

Edit Game | Group A | Switzerland - France

Result Switzerland - France 0:1*

☒ Game started
☐ Game finished

Starting Grid Events

Min.	Team	Event	Additional Information
27'	Switzerland	Yellow Card	Benjamin Meier
37'	France	Goal	Olivier Dechamps
64'	Switzerland	Substitute	In: Michael Kurz Out: Harry Steffen

Add... Edit... Delete

Save & Close Close

- The *Events* tab is only enabled if the game has been started.
- If the game is started, then the *Events* tab is preselected, when the dialog is opened
- *Add...* and *Edit...* opens the *Add/Edit Event* dialog, described in chapter 5.5.3.1.
- An error dialog should be shown, if the user clicks *Delete* and no event is selected
- *Delete* opens a Message Box where the user is asked, whether the event should be deleted
 - If the user clicks yes, then the row is deleted from the event table
 - If the user clicks no, then nothing happens
- The Event table should be sorted according to the *Min* column in ascending order

5.5.3.1 Add / Edit Event

Create the following dialog

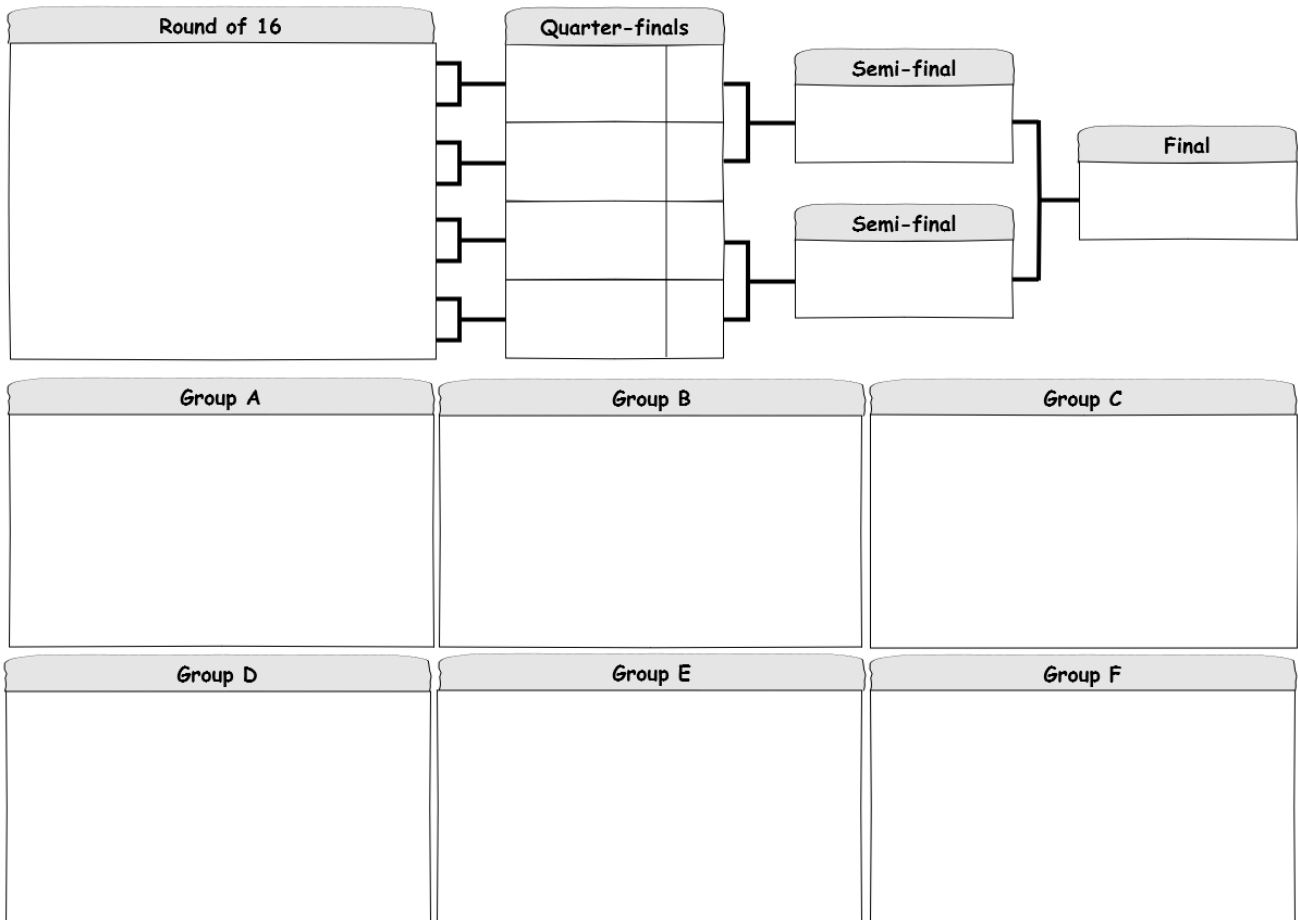
- The *minute* textbox only accepts numbers (without decimal point)
- The *Event* dropdown box shows the following events in this particular order: Goal, Foul, Free kick, Corner, Penalty, Substitute, Yellow Card, Red Card
- *Additional Information* allows any free text
- All fields are mandatory. The *OK* button should be deactivated if one or more of these fields are empty or invalid.
- *OK* adds / edits the Event
- *Cancel* closes the dialog without any change

5.5.4 Finish game

- The user can set the game state to finished by clicking the *Game finished* checkbox on the *Edit Game* window
- Once the game is finished, no events can be recorded anymore
- The game can only be finished, if it was started
- A confirmation dialog should be displayed where the user has the possibility to cancel the action
- After the group phase:
 - If a game ends in a draw the game has to be decided by a penalty shoot-out
 - Extend the confirmation dialog so that the user has to insert the result of the penalty shoot-out if both teams have the same number of goals at the end of the game
 - The *OK* button of the confirmation dialog should be disabled until a valid result of the penalty shoot-out has been inserted.

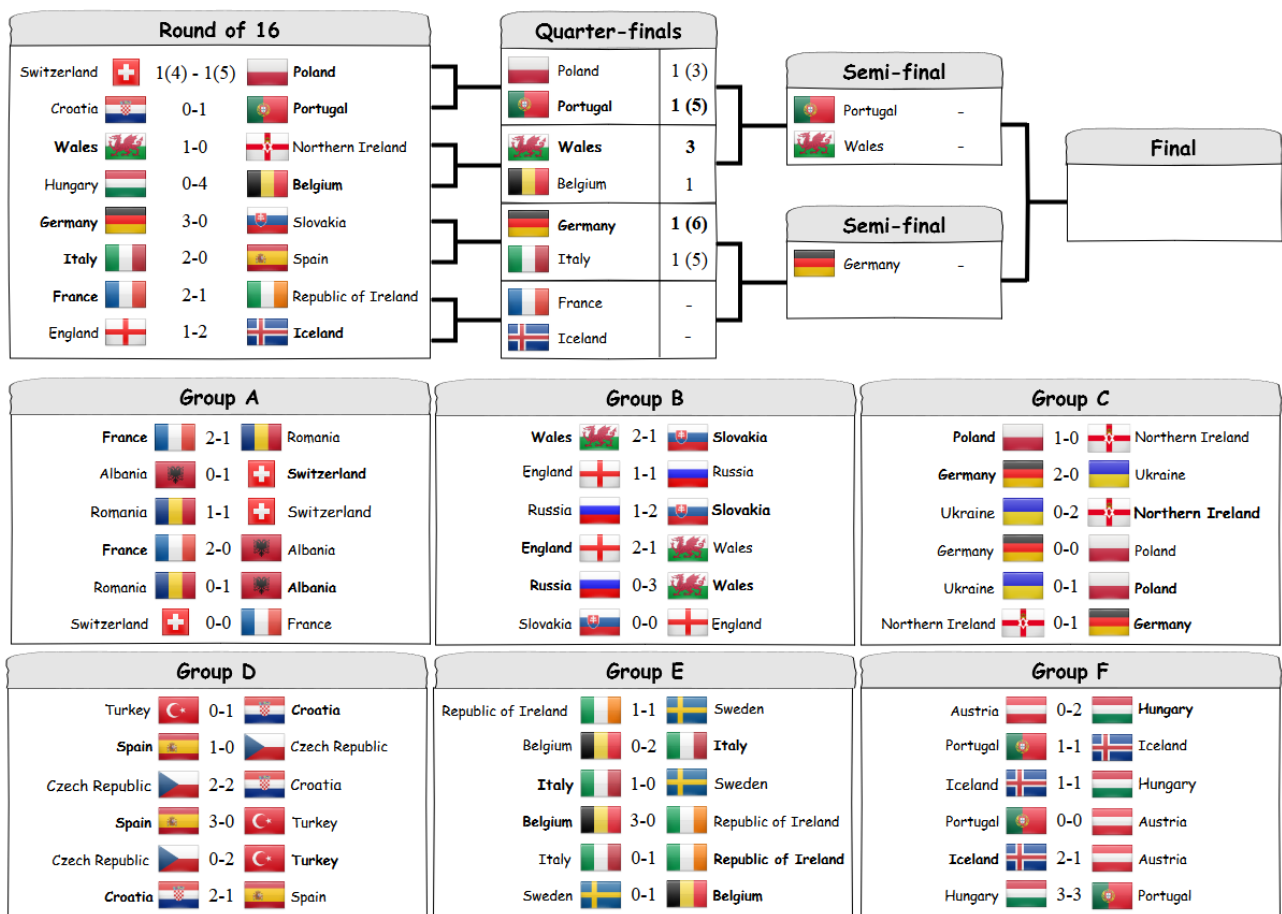
5.6 Tournament Overview

Use the data from the previous tasks to create the tournament overview on the dashboard as shown below.



Mockup 2: Empty tournament overview

- If the teams are not completely allocated to groups, all boxes should be empty (see Mockup 2: Empty tournament overview)
- Display the corresponding games and their results in the boxes
 - Display the two teams, including their flags
 - Display the score of the match, if the match is already finished. Otherwise display a dash character (“-“)
 - The name of the winning team should be written in bold. If the game is not finished, or the match ended in a draw, none of the teams should be written in bold.
 - After the group phase
 - display the achieved penalty-points in brackets, if the game was decided by a penalty shoot-out (e.g. Switzerland 1 (4) – 1(5) **Poland**)
 - Only display the teams in the boxes that are already qualified for the respective stage (e.g. see Mockup 3: Tournament overview during quarter-finals, for the second semi-final game only Germany is qualified at that time, because the quarter-final game between France and Iceland has not been finished yet).



Mockup 3: Tournament overview during quarter-finals

6 Mark Summary

SECTION / CRITERION	POINTS
5.1 Database	
Database design	3.5
Data import	3
5.2 Dashboard & 5.6 Tournament Overview	
Implementation of the userinterface	8.5
Dashboard functionality	7
5.3 Management (Tournament & Teams)	
Implementation of the userinterface	15
Basic data retrieval & functionality	21.25
5.4 Manage Execution	
Implementation of the userinterface	12.5
Basic data retrieval & functionality	22.75
Game generation & team assignment	10.25
5.5 Edit Game	
Implementation of the userinterface	12.5
Basic data retrieval & functionality	14.5
General	
Styleguide and Overall impression	9
TOTAL	139.75