Information System to manage a spectacle Hall

Analysis report

Document content

I. Introduction

The purpose of this project is to develop an information system whose aim is to manage a spectacle hall. The application shall help the manager to present its spectacles and to sell tickets. In this way, customers could browse the spectacles and buy a ticket for the show they choose.

There are two principal goals in this project :

- The first one is pedagogic, in the context of my last year of my studies in the ENSG-Geomatics school, I will revise some concepts I could have forgotten. Indeed, my specialization in a pure IT formation requires me to get into the swing of development. This could also give some clues to guide my choice between two sub-specializations.

- The second aim is to enjoy conducting this project during my free time, this is the only way I could keep my motivation to put in place this application. It will essentially be applied by some humor in my examples. So I had to choose a funny particular case, this is what I did with the name of my spectacle Hall : “le p’tit Rex” which will propose incredible spectacles made by my friends. The rest of the application will stay very classic. Then, I will minimize the useless tasks that are uninteresting to me, the design of the presentation for example.

From the first of october, two months will be needed to get the job done. The first is devoted to the analysis of the application as presented in this document. The month of November will be dedicated to its development.

An analysis report should be released before the 30th of october, the application before the 31st of november.

II. technical analysis

A. main features

The application will have to complete some features, they are summarized below as user stories

- As a spectator, I want to buy one or several tickets
- As a spectator, I want to cancel a booking
- As a spectator, I want to choose my place on a plan

- As a manager, I want to manage several spectacles on several dates
- As a manager, I want to manage spectators who buy or cancel places
- As a manager, I want to stop the selling of a show if the date is complete
- As a manager, I want to propose half less places (in case of pandemic for example)

- As an artist, I want to watch the booking statistics for my spectacle

B. zoom on the users

Three users are made out, the manager who manages the spectacles, the spectator who buys tickets and the artist who can consult the booking for his spectacle. The use case diagram is presented below.

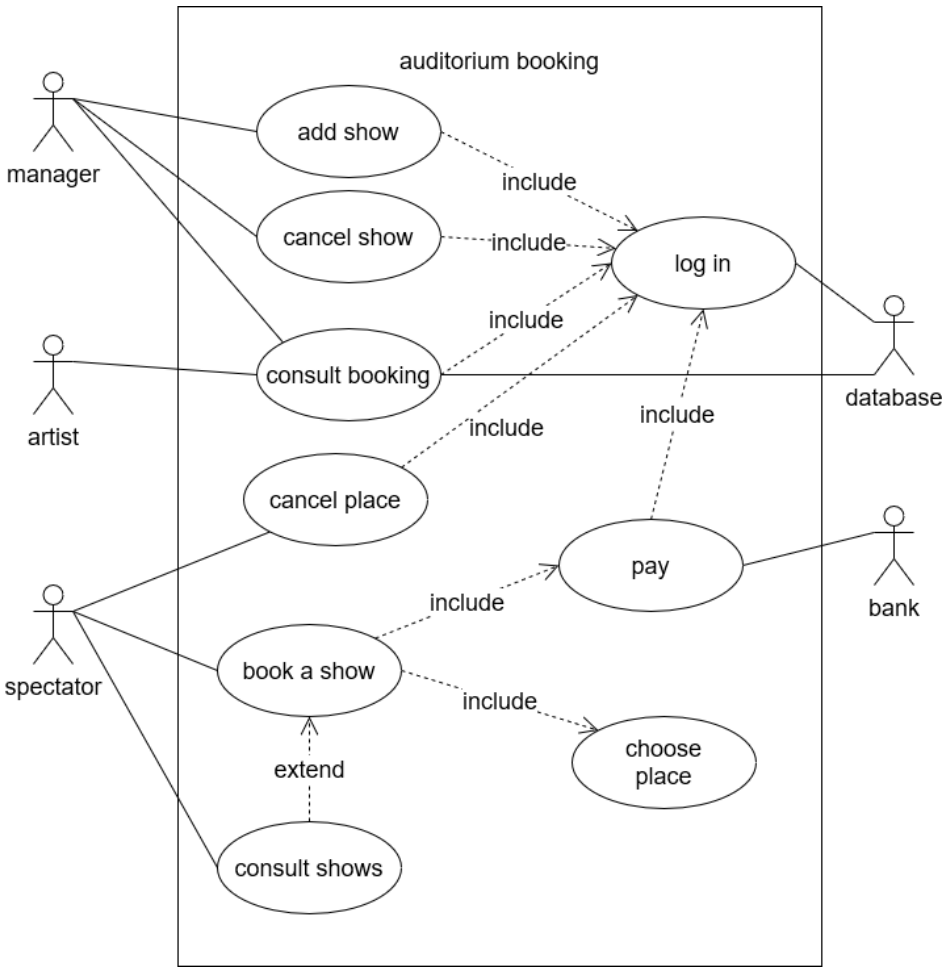

figure 1 : use case diagram

C. activities on the application

There are several activities to consider but we will focus on log-in, sign-up and ticket purchasing which are the main activities here. Their activity diagrams are presented beside.

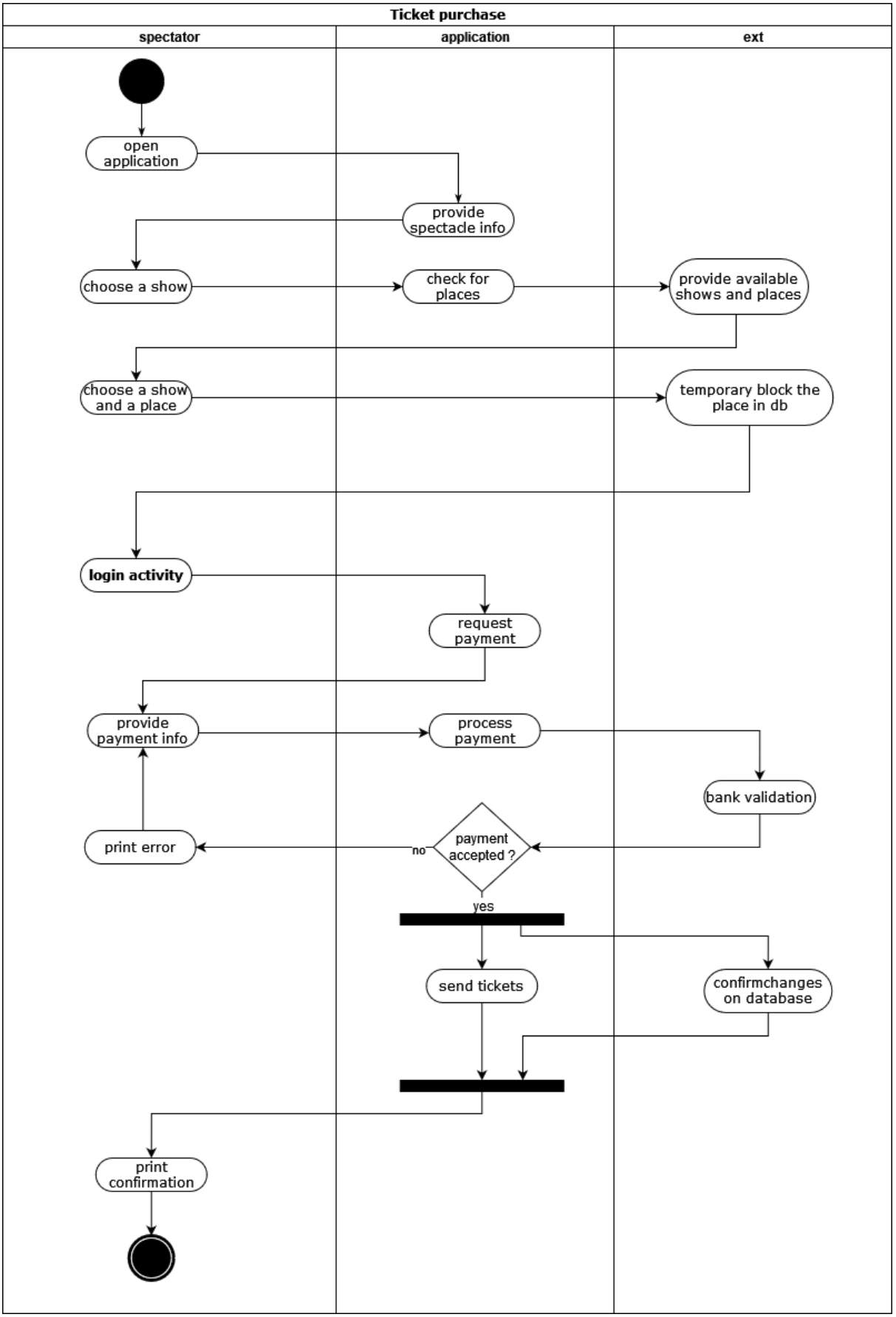

figure 2 : ticket purchase activity diagram

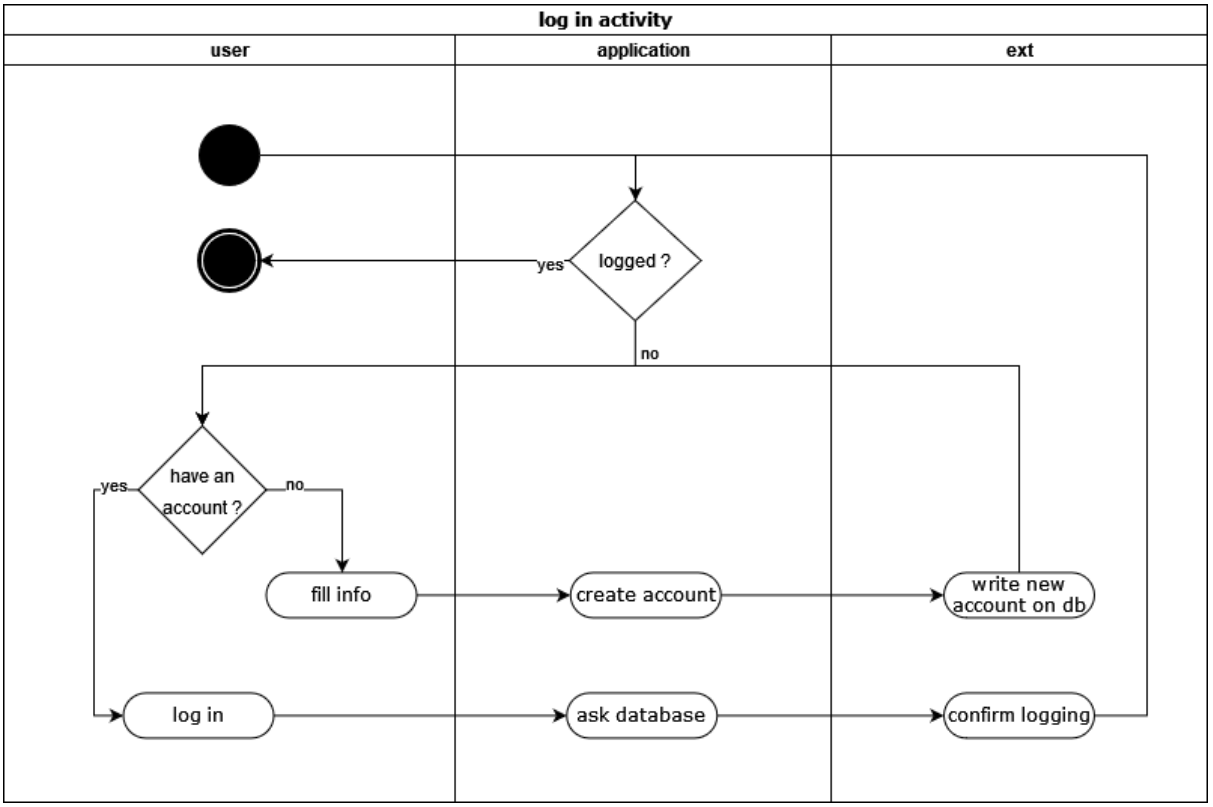

figure 3 : log-in and sign-up activity diagram

III. development analysis

A. development choices

It is clear that the application must be distributed on the internet to provide a web service available to the public. Then, a few possibilities are offered to develop this site :

- Jakarta EE is the java extension for business web services, it permits to provide a strong and well-secured website while taking advantage of java structure.
- LAMP is a most used architecture to develop websites, the access to a database is easier thanks to mySQL and php language.

Despite the robustness of Jakarta EE, the LAMP architecture is chosen because of the limited time. Moreover this little application hasn't the vocation to be deployed and used by many people.

B. class diagram

Application's architecture is developed around three packages :

- user : Describe the user features according to its role, Manager, Artist and Spectator classes inherit from the abstract class User
- spectacle : Deals with the management of the sp
- positioning : Describe the repartition of the spectacle hall with its places, there are three categories : the VIP places, in the Mosh pit and in Grand stands

Some classes are described in the database part.

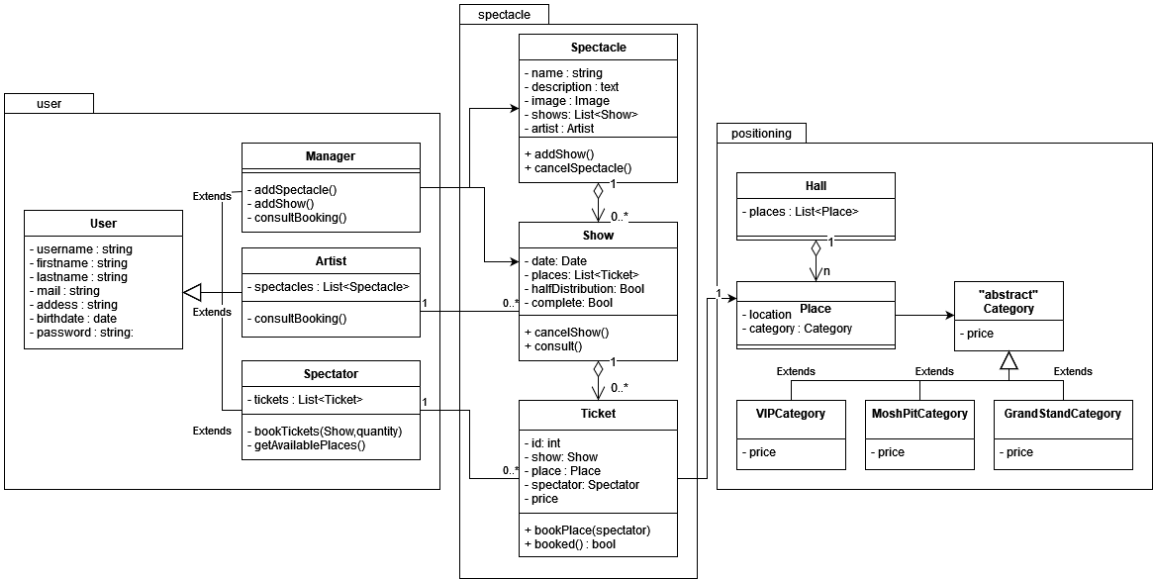

figure 4 : class diagram

C. database structure

A database has to be implemented and completed to use the application, its structure is described in the database diagram below.

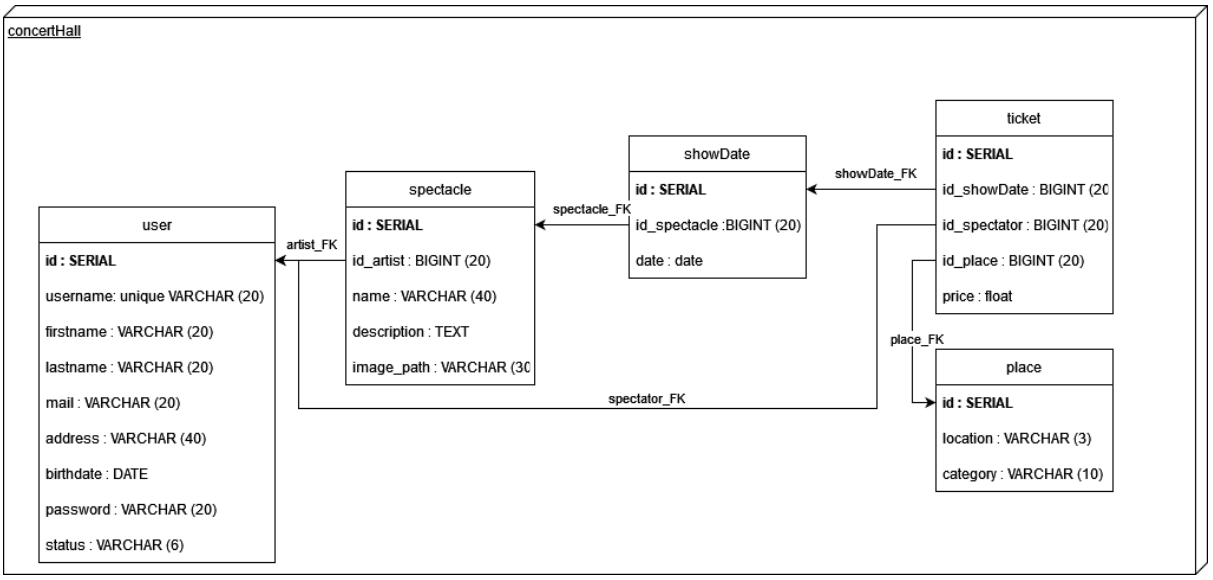

figure 5 : database diagram

- User table describes the user, its id and username which have to be unique and some information about the user. Its status can be “ARTIST”, “ADMIN” or “SPECTA”
- Spectacle table deals with a spectacle which will be implemented several times, it has an associated artist, a name, a description and a image path which will be displayed
- showDate table is an implementation of a spectacle, this is why it has a date and a spectacle reference
- place table is the list of any places in the spectacle hall, it has a location like “D45” and a category
- ticket table is the booking of a showDate by an user with an associated place

D. component diagram

The component diagram offers a global view of the application. The diagram below presents 8 main packages.

- the www package is the root of the website, it contains an index to welcome users and redirect them
- the auth package permit to authenticate by logging-in or signing-up
- the account package offers a personal space to manage its information, show bucket or disconnect
- the admin package is accessible if the user is connected as admin, it permits to manage spectacles
- the boxoffice package permits to consult a spectacle, show its show dates and purchase a ticket
- the app package is an oriented object which is implemented as the class diagram
- the accessDB package permit to ask requests to the database
- the database concertHall was presented previously