VILNIUS UNIVERSITY
FACULTY OF MATHEMATICS AND INFORMATICS
INSTITUTE OF COMPUTER SCIENCE
DEPARTMENT OF COMPUTATIONAL AND DATA MODELING

Software engineering project

CompanyRates

Done by:
Henrikas Ragickas
Povilas Blusevičius
Akniyet Omirzak
Semih Yoltan
Mantas Aušra
Tadej Rola

Supervisor:
Virgilijus Krinickij

Vilnius
2018

Contents

1 Project team

TEAM NAME	TableDroppers
LEADER	Henrikas Ragickas

2 System description

2.1 Name

CompanyRates - CR

2.2 Purpose

The main purpose of this system is to allow users to view information and share their feedback about registered companies.

2.3 Description

The name of our system is CompanyRates and it allows users and companies to register their accounts on our system. The main idea is to have information of various companies in one place, which could help a person who is looking for job or specific services. That is why companies have to leave a description and put their contact information, such as name, address, phone number and logo in order to qualify. What is more, users, that register on our platform are able to post reviews and rate companies in order to reflect their work experience in the selected company.

2.3.1 Requirements

- Integrated development environment will be Visual Studio 2017, Visual Studio Code
- RESP API in WCF framework using C sharp programming language,
- Database Microsoft SQL Server,
- ORM technology Entity Framework,
- Web site HTML, CSS and JavaScript

2.3.2 Functionalities

- **Company Account registration:** To register company account, client must be owner of the company or have a permission to manage company‘s account. All registered accounts will be checked by administrator.
- **User Account registration:** To register User Account client has to provide all necessary data for his identification.
- **Log In:** All registered Users/ Companies can log into their accounts at any time.

- **Administrator session:** Administrator can connect to a special account that provides him with features to approve pending accounts or delete comments.
- **Description modification:** Information that is presented by company can be updated during any Log In session.
- **Comments:** Users can leave comments about companies. Comments can be rated by other users and can be deleted by administrator for improper language usage.
- **Rating:** Companies or comments can be rated with Thumbs up/ Thumbs down method.
- **Visitor session:** User without account can see registered companies and leave anonymous comments.
- **Connectivity to database:** Private user information (passwords, management data...) are stored in main database that can be accessed only with required rights.
- **Sorting system:** All companies are sorted by category to make it easier for user to find what he is looking for.
- **Check on map:** Location of companies will be shown on map.
- **Charts of different data:** For easier and more fluent companies comparison user will be provided with visual data represented by charts.

2.3.3 Environment

- REST API will run on Windows IIS,
- Web site will run locally,
- Microsoft SQL Server will be hosted on AWS

2.3.4 Customer perspective

CompanyRates will make it easier for you to find company that provides your desired services. The final version of our product should be able to give you best experience acknowledging most important facts about companies that interests you. You will find descriptions of various companies and reviews from their workers or customers in one place. Descriptions are constantly updated so you may never miss new offers from desired company. Application will provide you with modern search engine. You may find company by it‘s category or in a specific search system. Considering between companies will be easier than before because our rating system will show you the score of company that is based on customers grading and positive/ negative reviews will help you to make your final verdict for your decision. For full experience you‘ll have to register in our system. You can register yourself as a User or as a Company. Company account can manage its description and advertise it‘s offers. Registered accounts gains access to all features of this application. Otherwise you can just browse as a visitor. You still will be able to search for companies or leave anonymous comments but rating systems will not work for you.

2.3.5 Prototype images

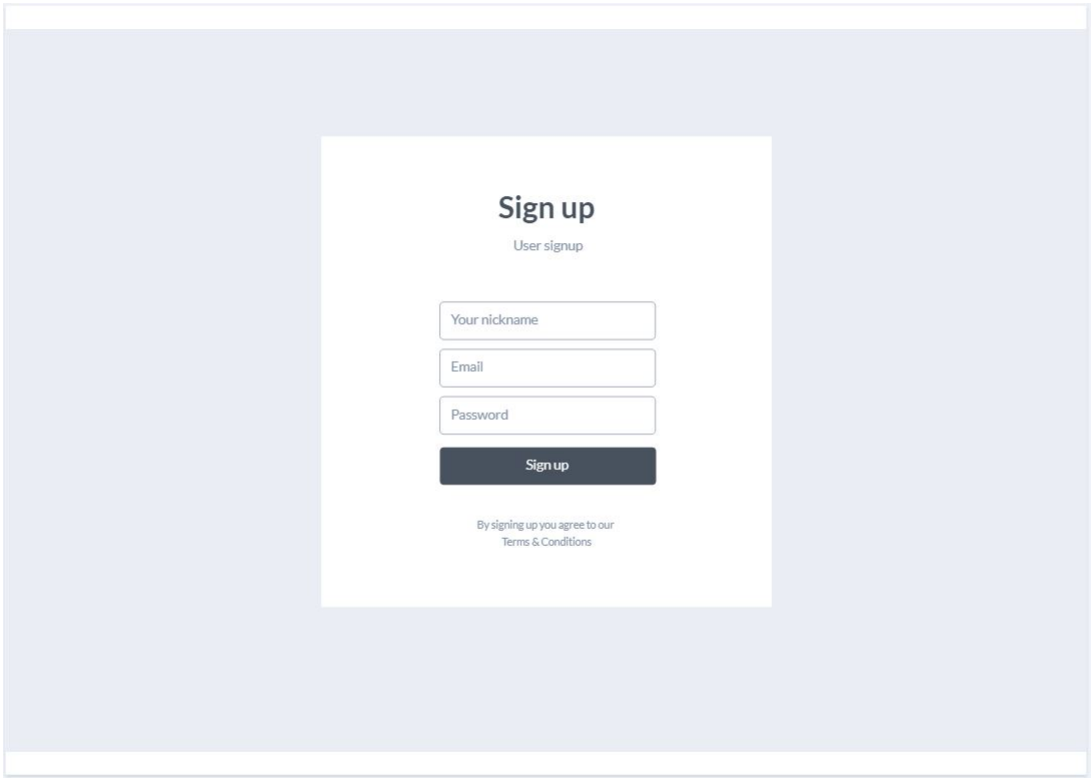

Figure 1. User signup

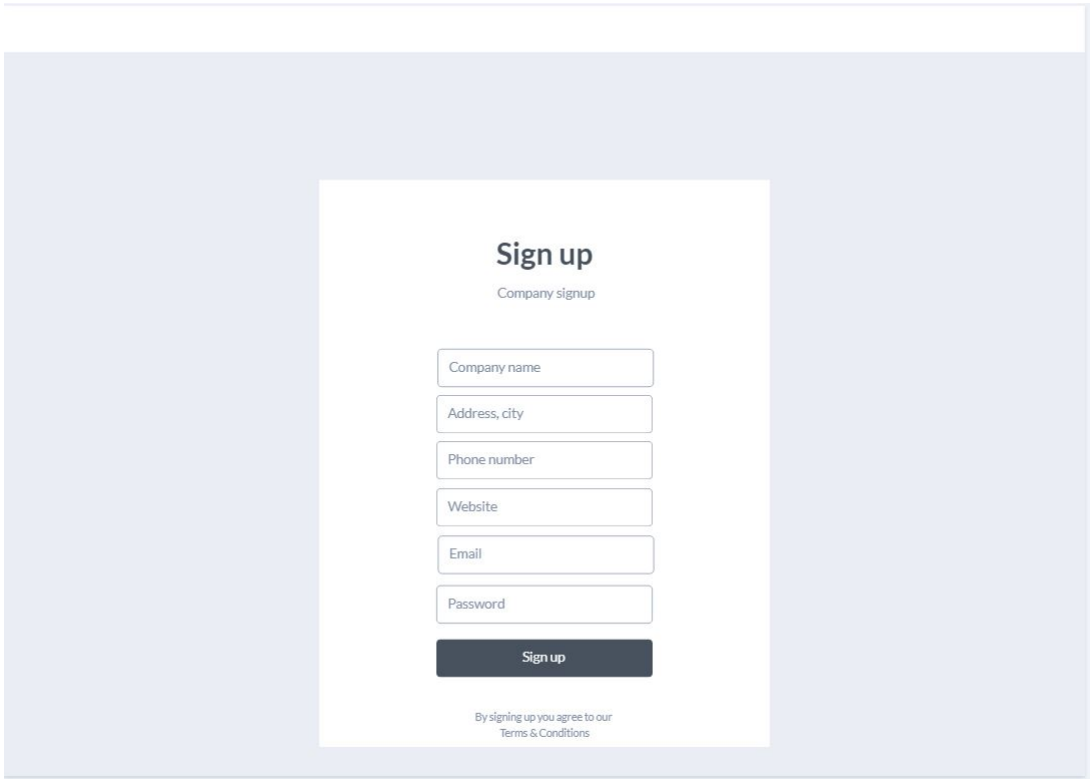

Figure 2. Company signup

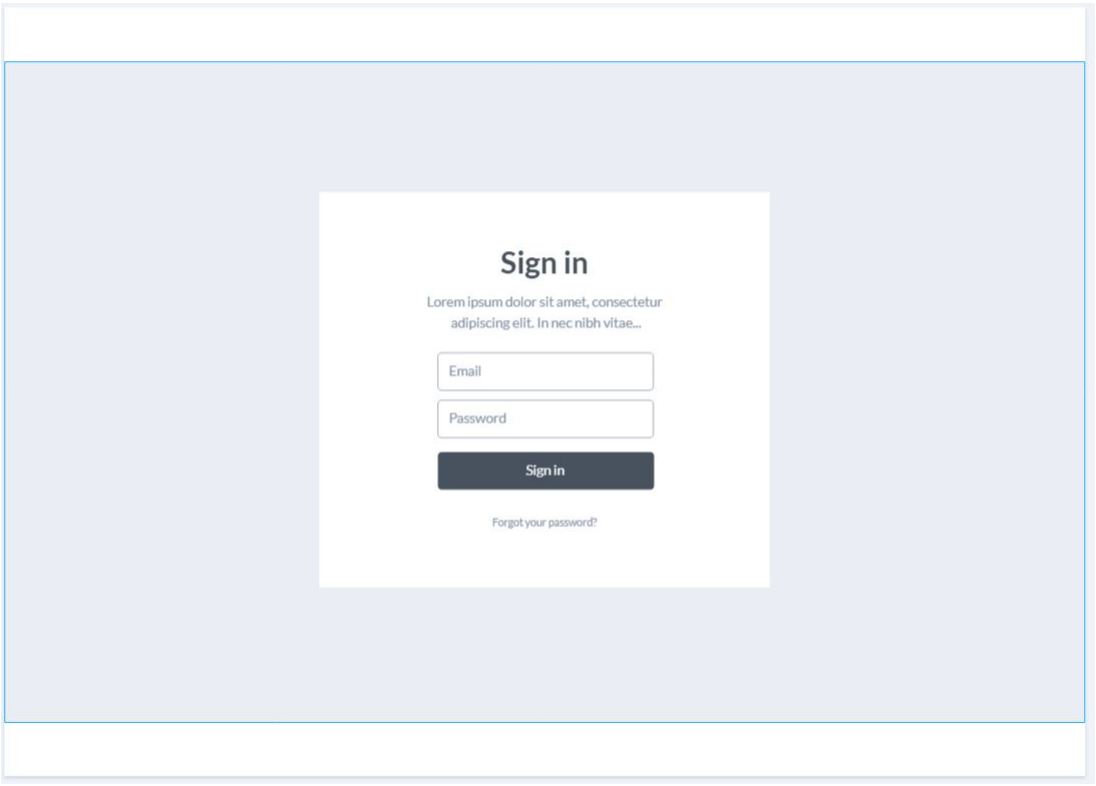

Figure 3. Login

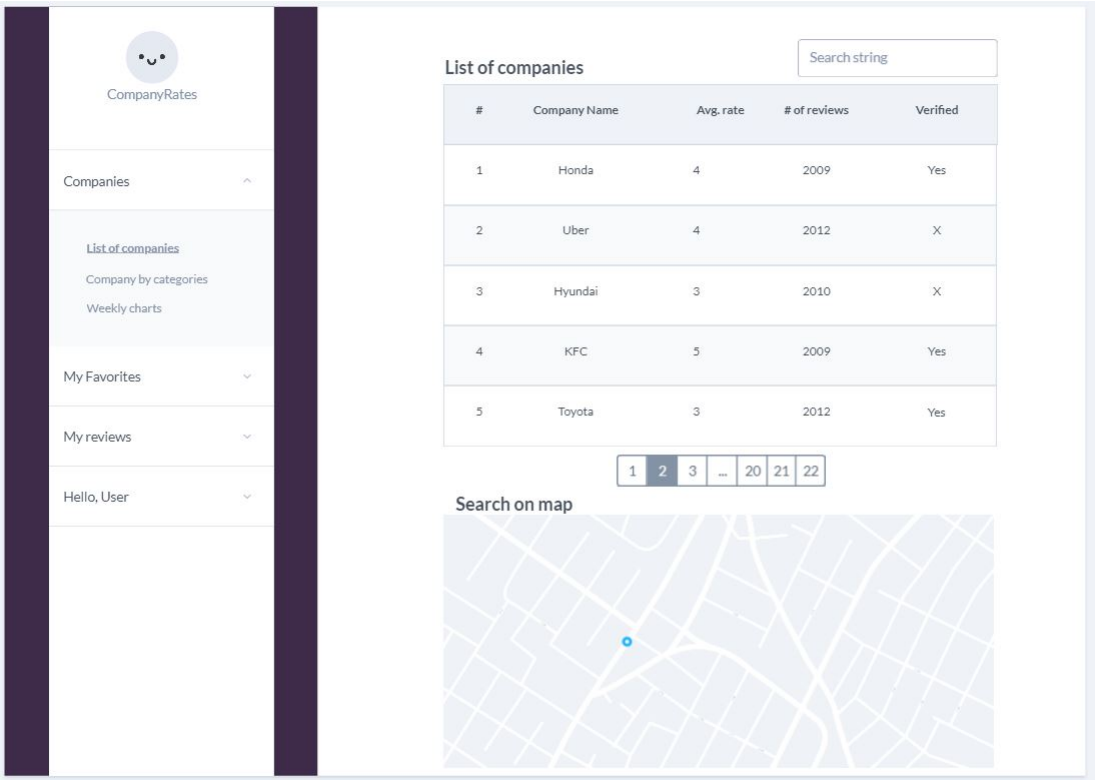

Figure 4. Home page as use

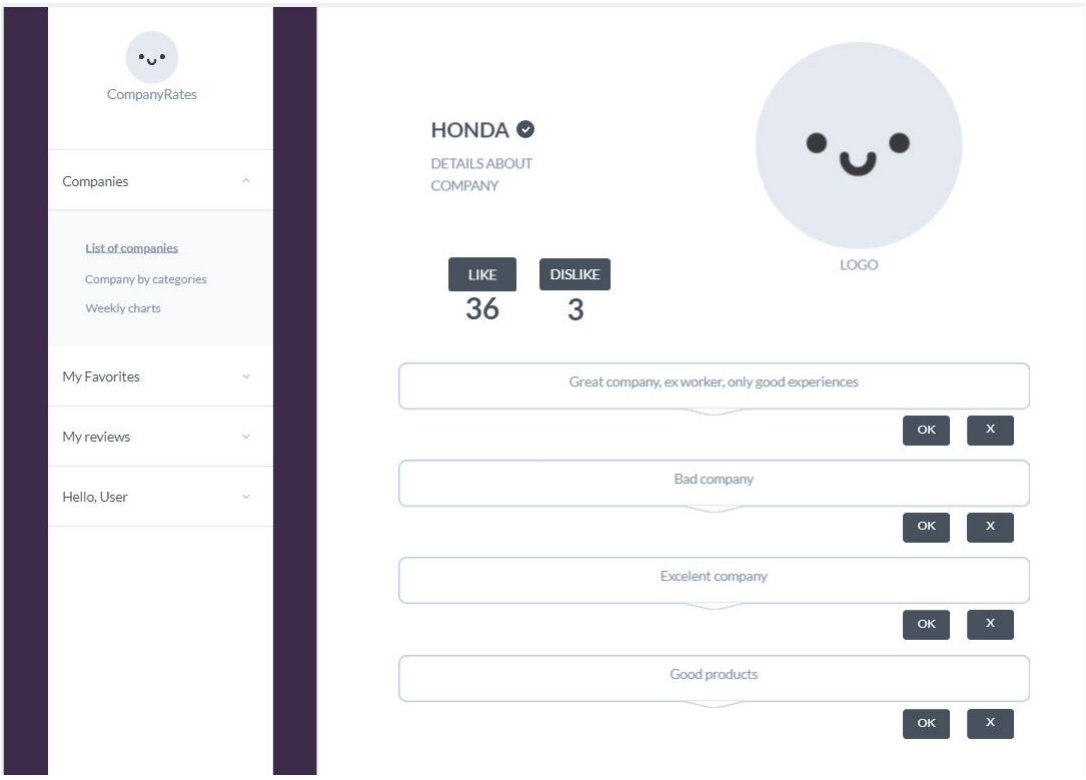

Figure 5. Company details with reviews

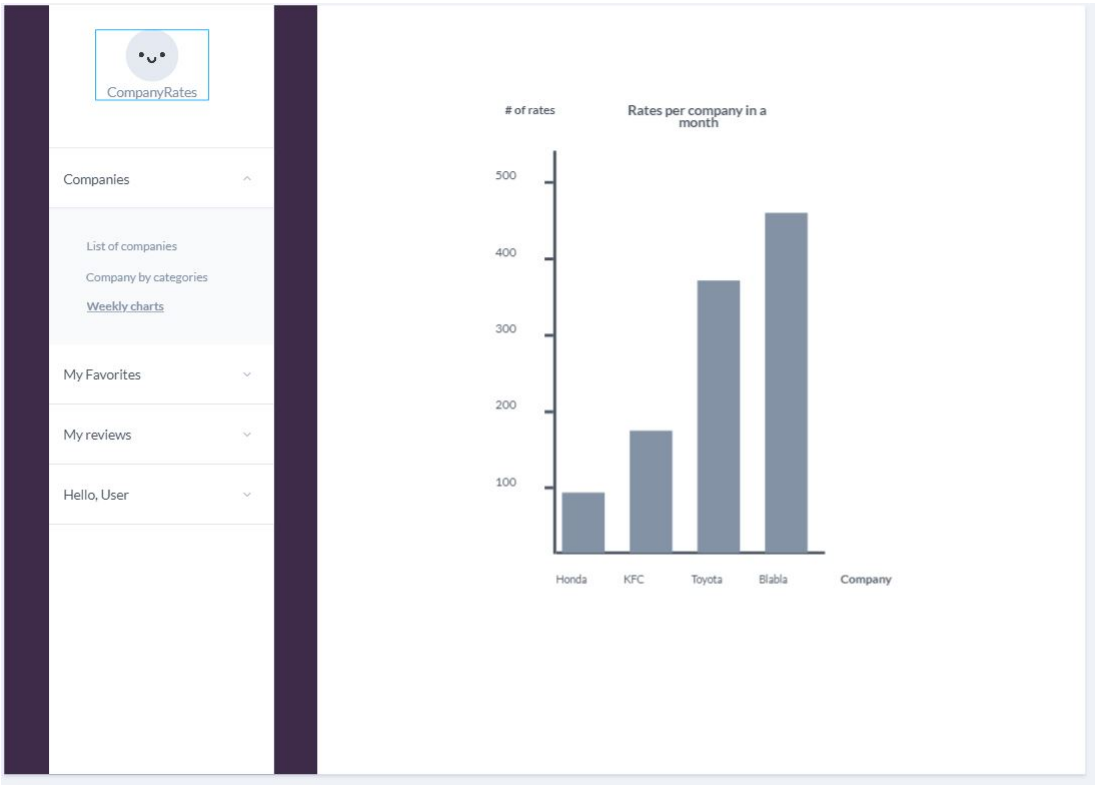

Figure 6. Charts of difernt data

3 System specification

3.1 Acting actors and objects

ACTORS	OBJECTS	ATTRIBUTES	FUNCTIONS
Administrator	1. Reviews 2. Company list 3. Platform users profiles 4. Company details	1.1 Admin can read and delete 2.1 Admin can read and delete 3.1 Admin can read and check 4.1 Admin can read and check	1.1 Admin have a full control of reviews on platform. It means he can delete any reviews that do not fit on platform 2.1 Admin can read company list and delete if companies have more than 1 account by announcing it to the owner 3.1 Admin can read platform users profile and check if all information are correct 4.1 Admin can read and check if all details are appropriate
Platfrom User (is the person who registered on our platform with an email address and password)	1. Reviews 2. Company list 3. Platform users profile 4. Company details	1.1 User can read and write 2.1 User can read 3.1 User can read and update 4.1 User can read	1.1 User can read other user's reviews and write own review 2.1 User can only read company list and choose needed service 3.1 User can only read other user's profile and update own profile 4.1 User can only read company details and choose needed one
Company user (is the person, that represents company, who registered on our platform with email address and password and details about company)	1. Reviews 2. Company list 3. Platform users profile 4. Company details	1.1 Company User can read 2.1 Company User can read 3.1 Company User can read 4.1 Company User can read and update	1.1 Company User can read user's reviews 2.1 Company User can read company list 3.1 Company User can read other user's profile 4.1 Company User can only read other company's details and update own company details
Visitor (guest on platform, no user data)	1. Reviews 2. List of companies	1.1 Visitor can read 2.1 Visitor can read	1.1 Visitor can just read user's reviews 2.1 Visitor can just read list of company

Figure 7. Table of system actors, their objects, attributes and functions

3.1.1 Non-functional requirements

- Extensibility (taking future growth in consideration and adding new features)
- Reliability (preventing failures and being able to recover)
- Security (safeguarding the information and data from external and internal security threats)
- Usability (giving the best experience to users while using the system)
- Stability (performing and functioning consistently over period of time without breaking down)
- Performance (speed and e ciency)
- Availability (is it possible to access the system from all devices and locations)
- Compatibility (is the system compatible with all hardware and sofware)
- Safety (user authentication and database back up)
- Supportability (maintaining the system)

In CompanyRates one of the most important non fuctional requirements is usability, being user friendly and giving the best experience while using our system. To achieve this our system has to be stable, performing and functioning consistently over period of time without breaking down, as well as reliable, with ability to recover and prevent failures. It also has to have good performance, browsing has to be speedy and have modern search engine. Another important requirements are safety and security, that is why users have to be authorized and in order to gain access to all functionalities they have to login with username and password. Users from di erent locations and with di erent devices should also be able to use our system, meaning availability and compatibility have to be taken seriously. Lastly, in order to maximize the experience, CompanyRates has to be maintained and constantly updated and have extensibility, by taking future growth and adding new features in consideration.

3.2 Logical function groups

Package diagram:
This package diagram shows relations between platform users and functionalites. User can rate reviews and companies or write his own review. Company user can write reviews. Administrator can delete reviews All platform users can access connection packages. Visitors actions are hardly restricted.

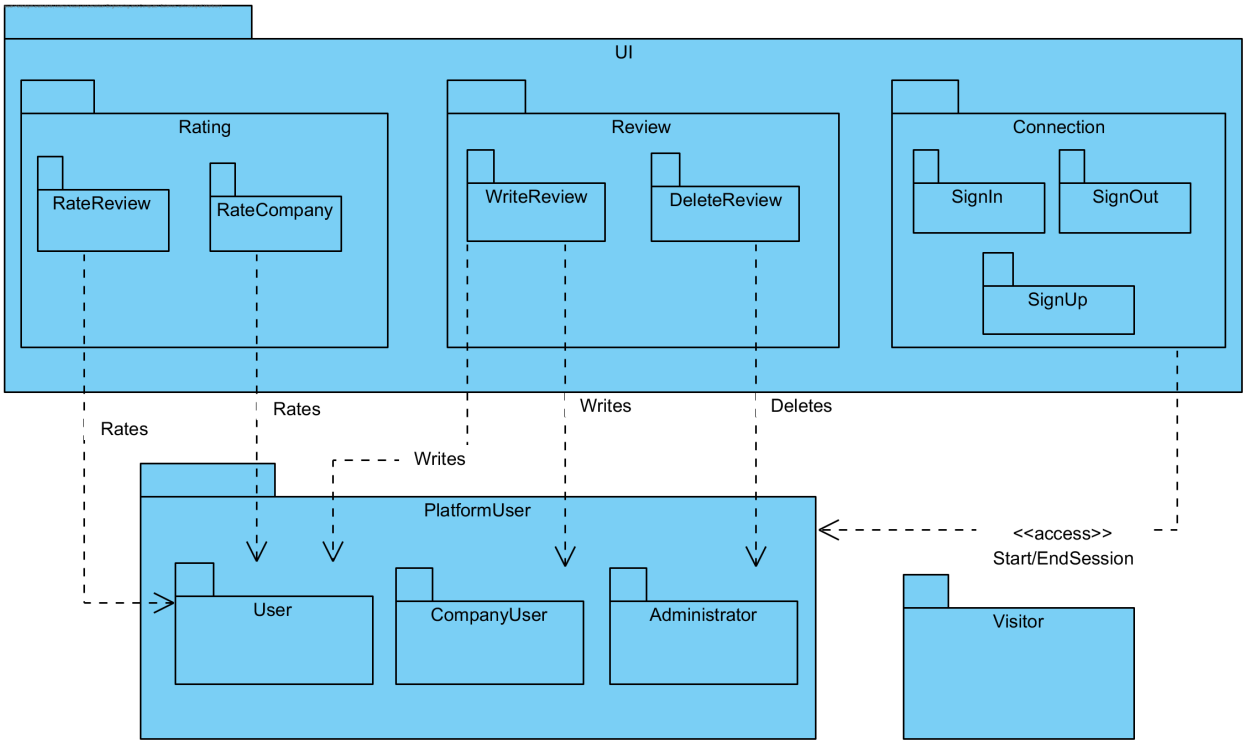

Figure 8. UML package diagram

3.3 Use cases

Use case diagrams model the functionality of a system using actors and use cases. Visitor on our platform, has access to functionalities such as Registration and Login (where Recover Password is extended functionality). Each actor in our system has ability to see full list of companies, including their reviews, ratings and details. Administrator has ability to approve Company User accounts and removing (harmfull) reviews. On the other side, Platform User has ability to write a review. And for his reviews, user can remove or edit them. Also, Platform User can update his profile details. Company User cannot write reviews, but can change details about their company, that are displayed on the platform.

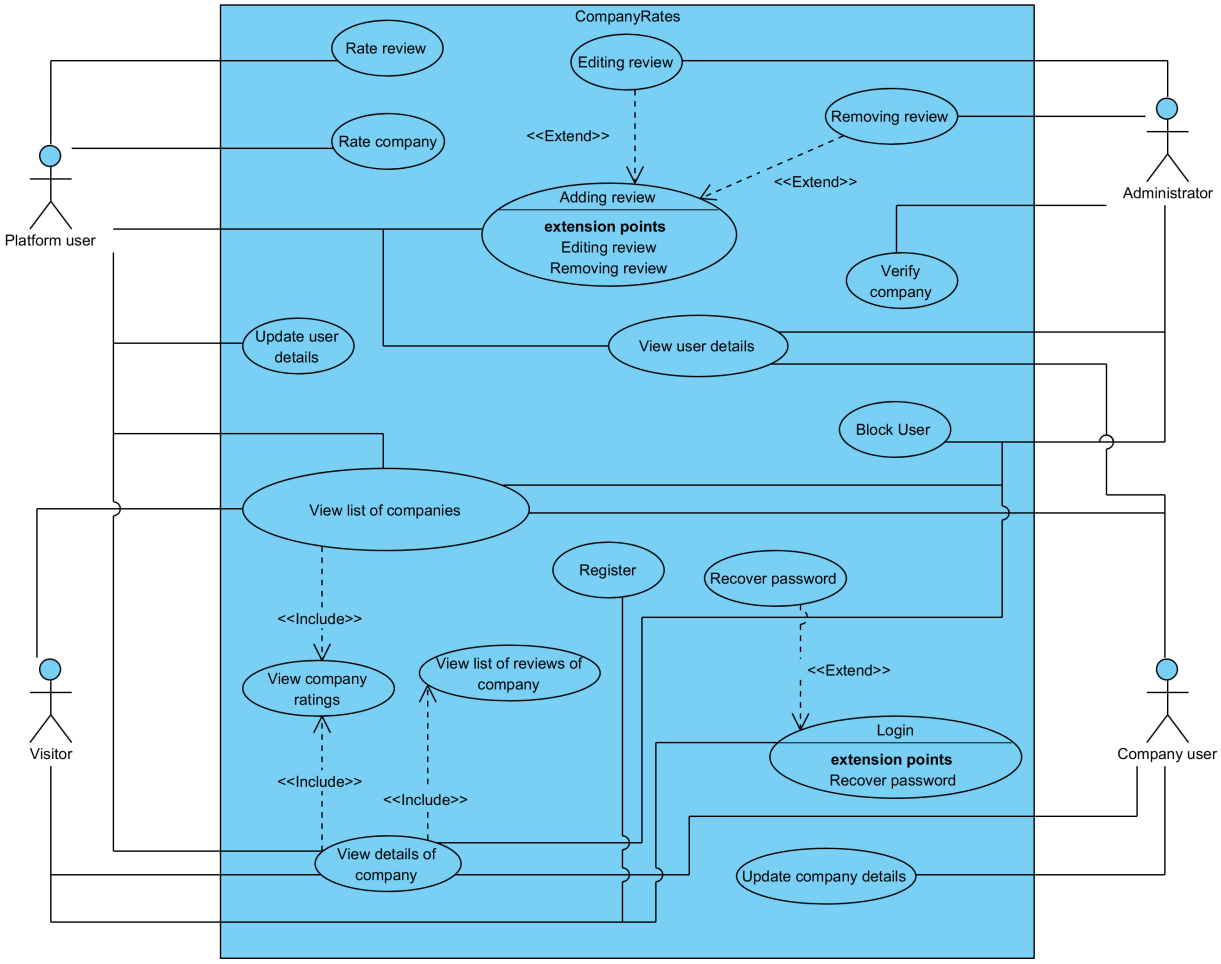

Figure 9. UML use-case diagram

This section represents system roles and system functionalities in a cross-reference table. We can see, that each of actors in our system, has access to function "View list of companies", "View Company ratings", "View details of company" and "View list of reviews of a company". By this table, we can easily identify, what are the roles in the system and who has allowances for specific operations.

Function/Actor	Platfrom User	Administrator	Company User	Visitor
Rate review				
Rate company				
Update user details				
View list of companies				
View company ratings				
View details of company				
View list of reviews of company				
Adding review				
Editing review				
Removing review				
View user details				
Verify company				
Recover password				
Register				
Login				
Update company details				
Blocking user				

Figure 10. User function matrix

3.4 Links between actors and objects, classes

Object diagram:

This object diagram displays how a review and its rating are created. First session is created using createSession method. Then Login method is used to login.Since is a platform user, he can create a review using createReview method. Once a review is submitted it can be rated with addReviewRating method.

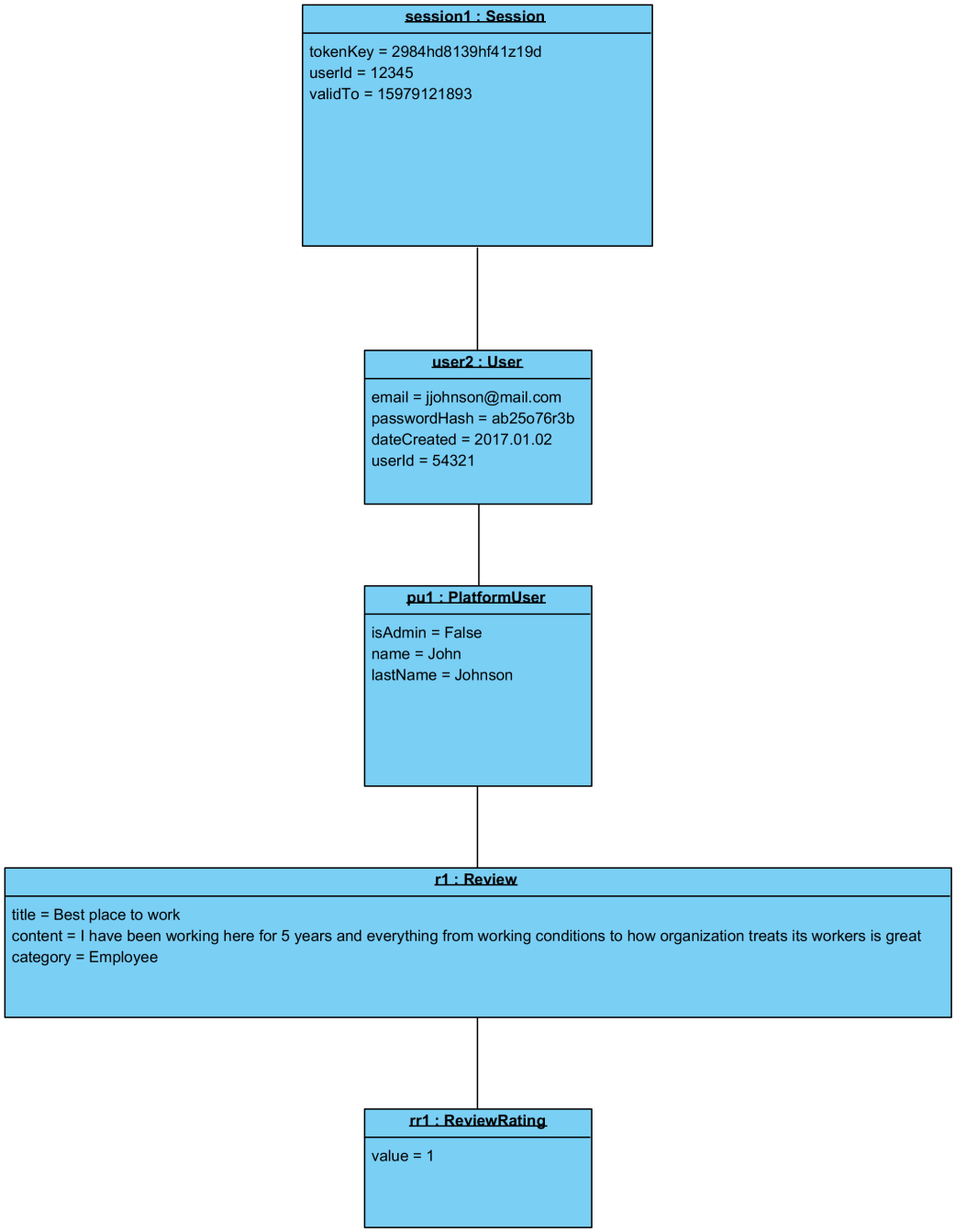

Figure 11. UML object diagram

Class diagram:

Class diagram represents static structure of system's classes. We have 2 di erent user types, that both inherit from base clas User. User have 3 common attributes. For each user, we will manage the login session. Every company instance has an array of reviews, and each Review belongs to one CompanyUser. On the other side, PlatformUsere can add and rate those reviews with 1 and -1. Thats why we have function, that sums up the total ReviewRating.

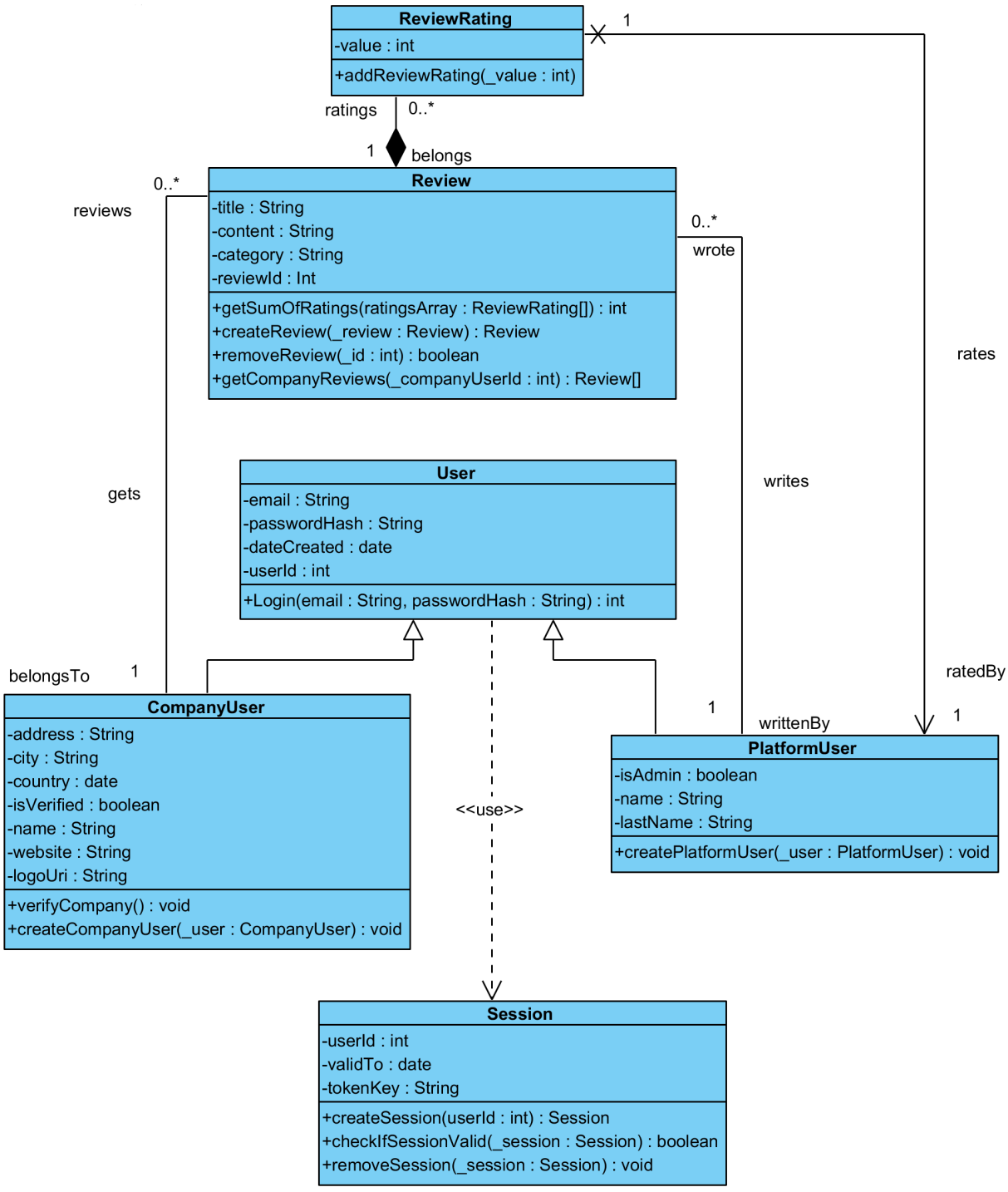

Figure 12. UML class diagram

3.5 Subsystems

Subsystem is a cross between a package and a class. It behaves like a package in that it groups other model elements, but it also behaves like a class because it has a specified behavior. In this subsystem we represent behaviour of three packages of classes. In our Subsystem diagram, we represents how our Authentication in our system works. We have realization elements from class diagrams, that are User (base class), and two derived classes (PlatformUser and CompanyUser). Also, because we are dealing with authentication, we have Sessio class. In specifications elements sections we have use cases related to these subsystem. Also we have operations section, where an actual methods are presented, and are realized by the class diagram methods.

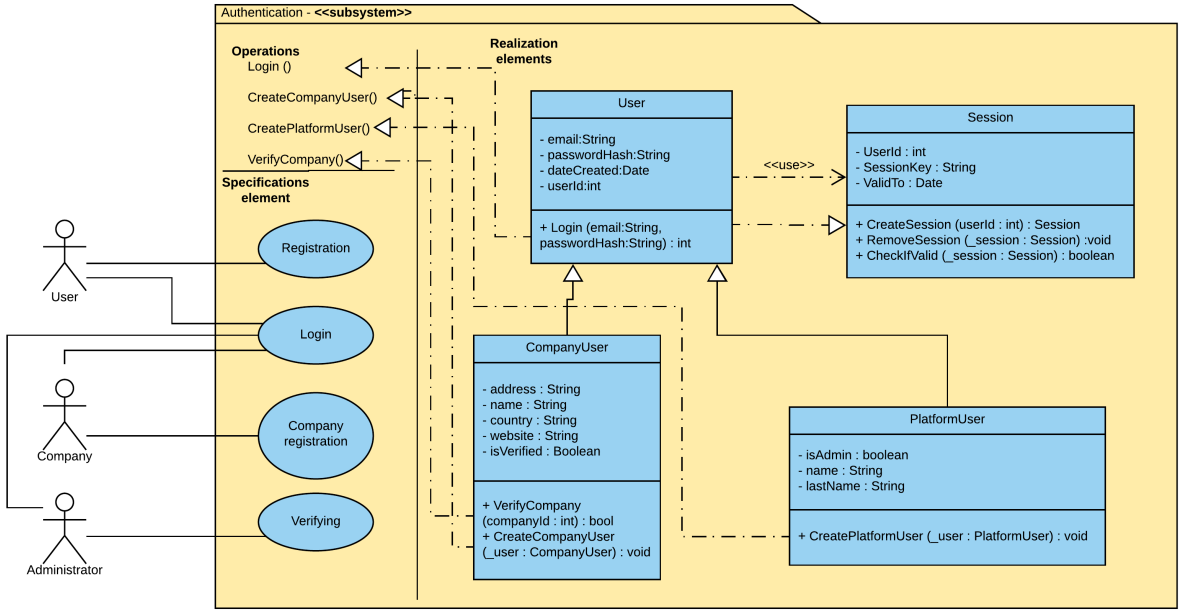

Figure 13. UML subsystem diagram

3.6 States of system elements and transitions

This state diagram describes behavior of systems that are used for company account registration functionality. In order to create Company Account user may have to fill a registration form that appears once "Register Company" button is pressed. Canceling registration will automatically delete registration form with all of its contents. If user submits his filled form then state of form will change into "pending for approval" and user will have to wait for interference of website administrator. From this point admin will have to check if information provided by user is valid and he will have a choice to approve it or reject it. If form is approved then Company Account will be created and user will be be able to reach it from login section. Otherwise if form is rejected then it will be deleted from database, user will be notified about failed registration and encouraged to try again.

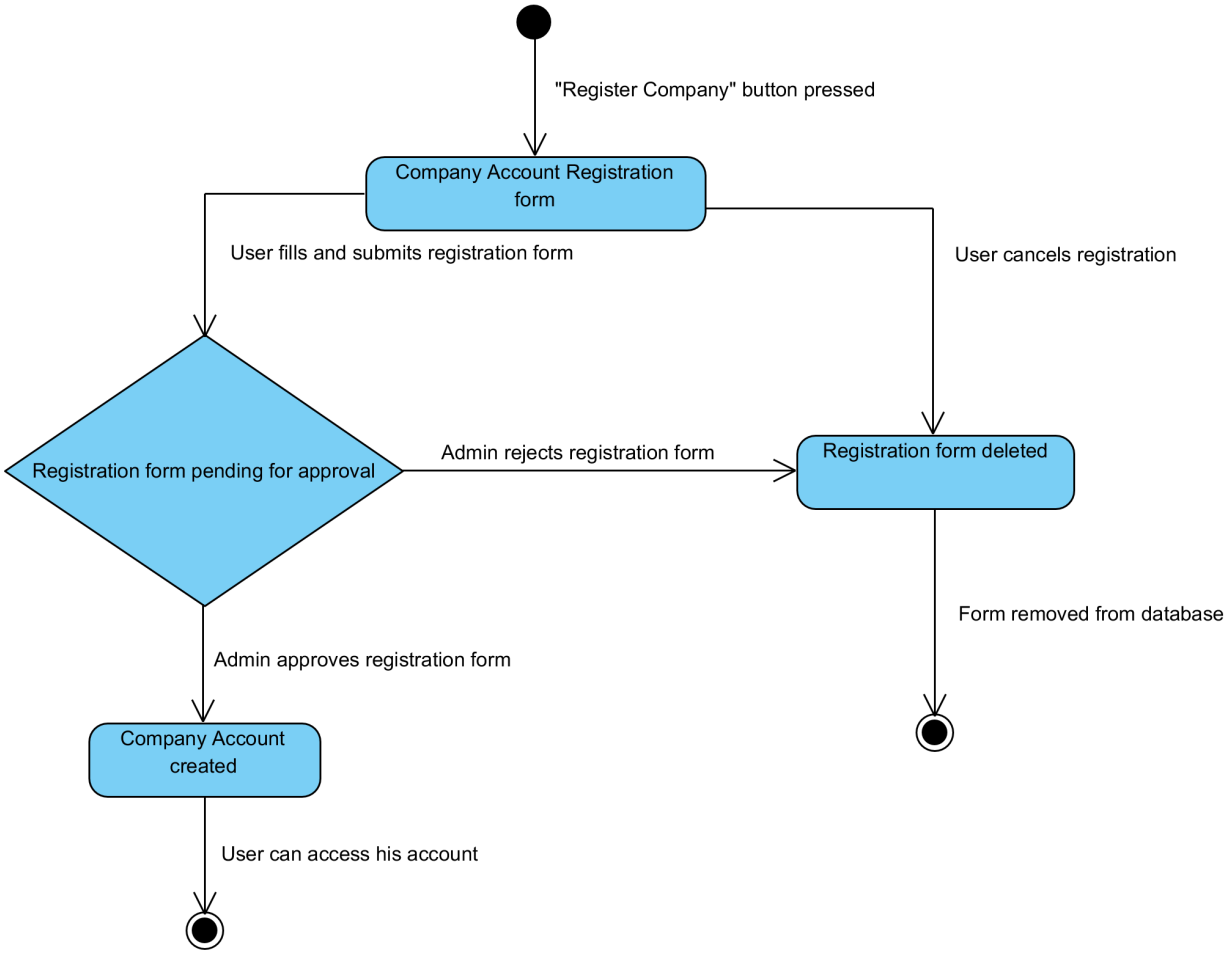

Figure 14. UML state diagram of Company account registration

3.7 Database structure

Database relationship diagram represents the database design. It containst di erent symbols and connections, that visualize two important information: Major entities withinh a system scope, and their relationship among these entities. We designed our diagram in a 3rd normal form. It means, that none attributes in the tables are functioanl dependent on anaother attribute in the same table. Each entity has at least one relationship with another entity. In our relationship diagram, we can not have many-many relationship, because it should represent new table in database. Each string attribute in the table is represented by data type "varchar", with usually fixed amount of char spaces. For the number attributes, we are using only variations of integers, becase we will not work with decimal numbers in our system.

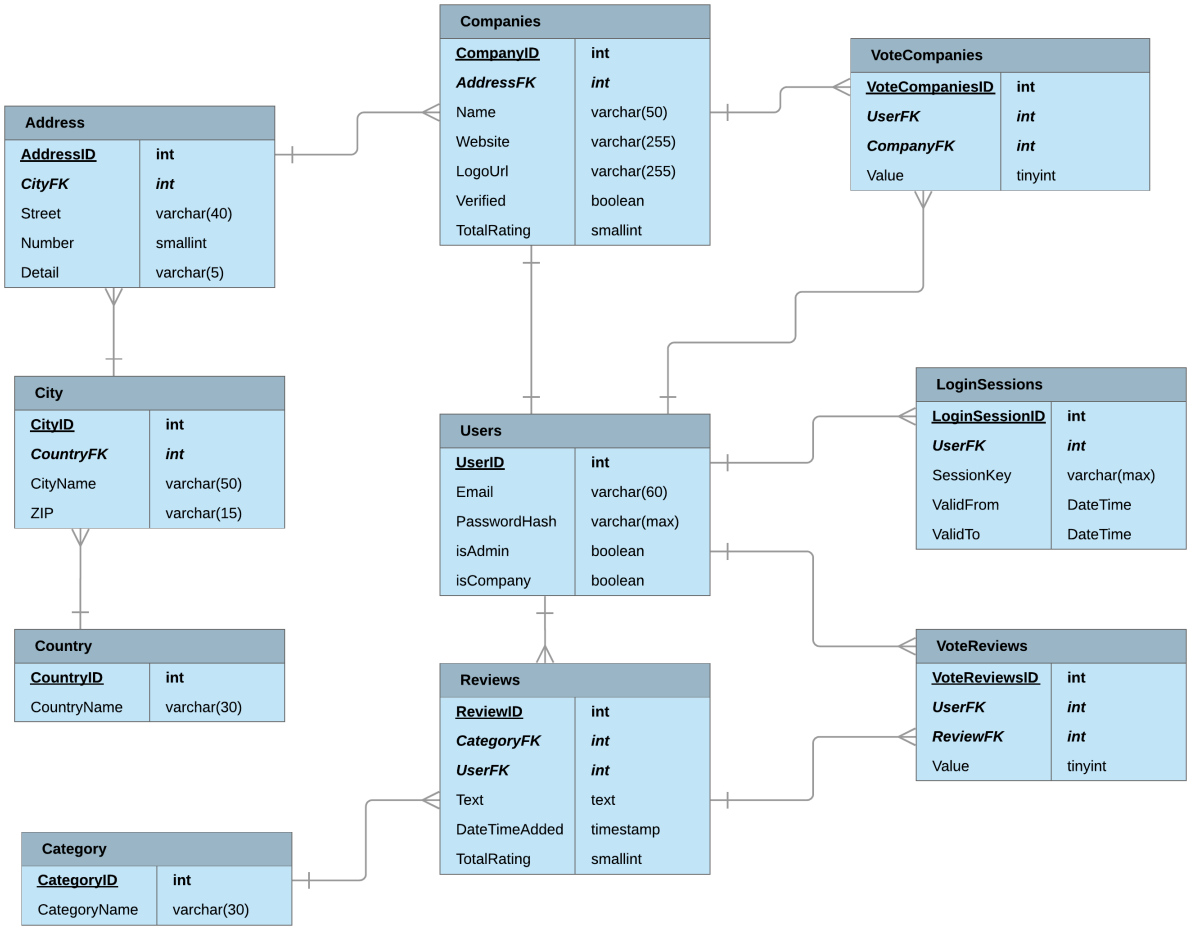

Figure 15. Relationship diagram, that represents tables and their attributes in relationship to other tables

3.8 Activity procedures

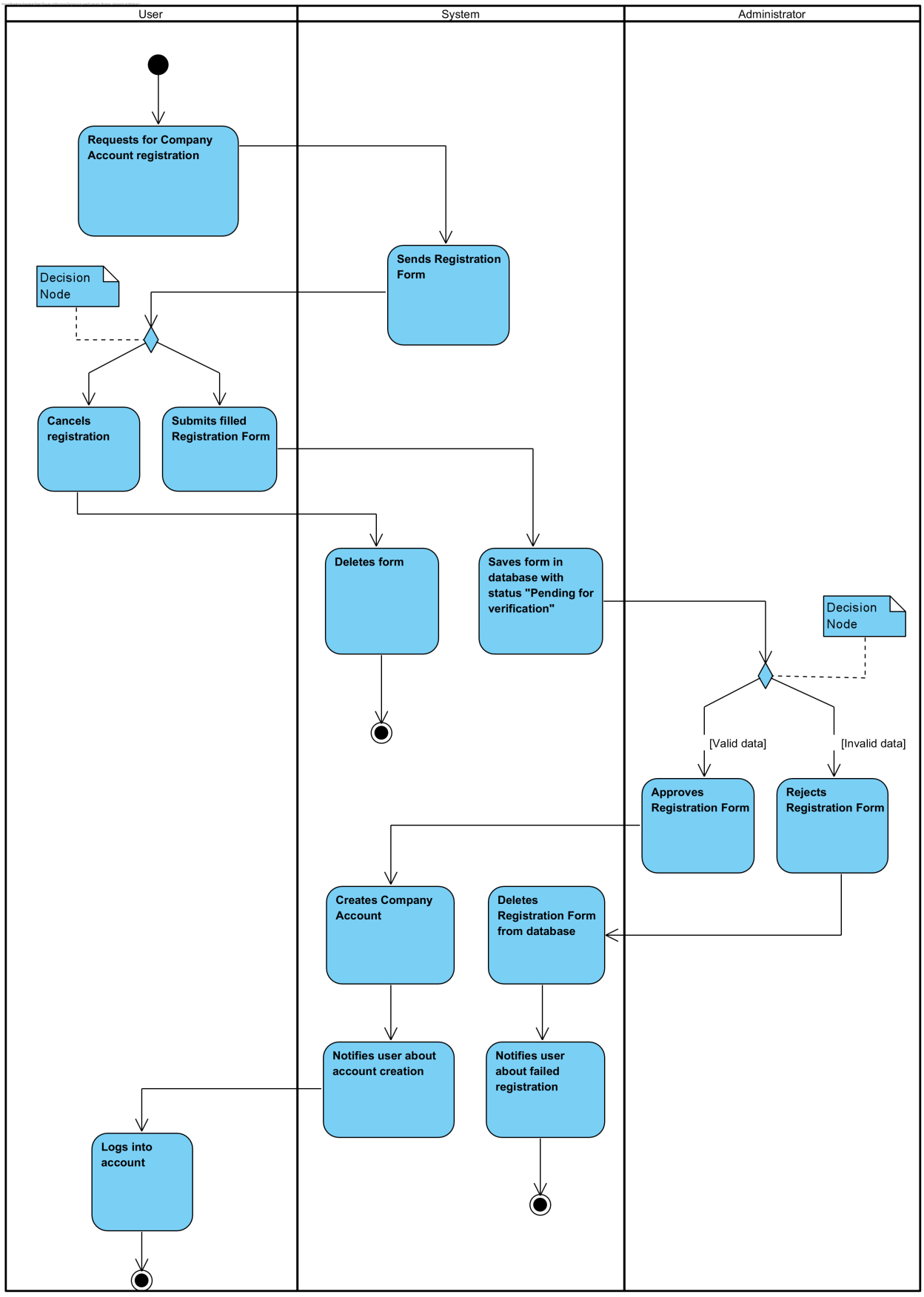

Figure 16. UML activity diagram for registering as a company user

DIAGRAM 1 DESCRIPTION:

This activity diagram is used to graphically represent workflow and possible choices at di erent levels of company account registration functionality. Company Rates platform provides its users with an ability to register their Company Account. To use this functionality user should select company registration button from webpage home screen. Once this function is triggered, system will provide registration form for user to fill. It is necessary to fill every field of this form, otherwise submit button will stay inactive. During registration user can cancel it at any time, it will cause automatic deletion of all his inserted data. If all fields were filled and submit button were activated, then system will send filled form to database where it will be waiting for verification. To qualify for verification all data must be valid because administrator will check it and decide if this account satisfies platforms rules. In case of invalid information, administrator will reject submitted form and then system will delete it from database and also will send a notification for user about failed registration. If submitted form had all fields filled correctly, then administrator will apply this form and system will create an account by submitted data. Once account is created, user may log into it at any time.

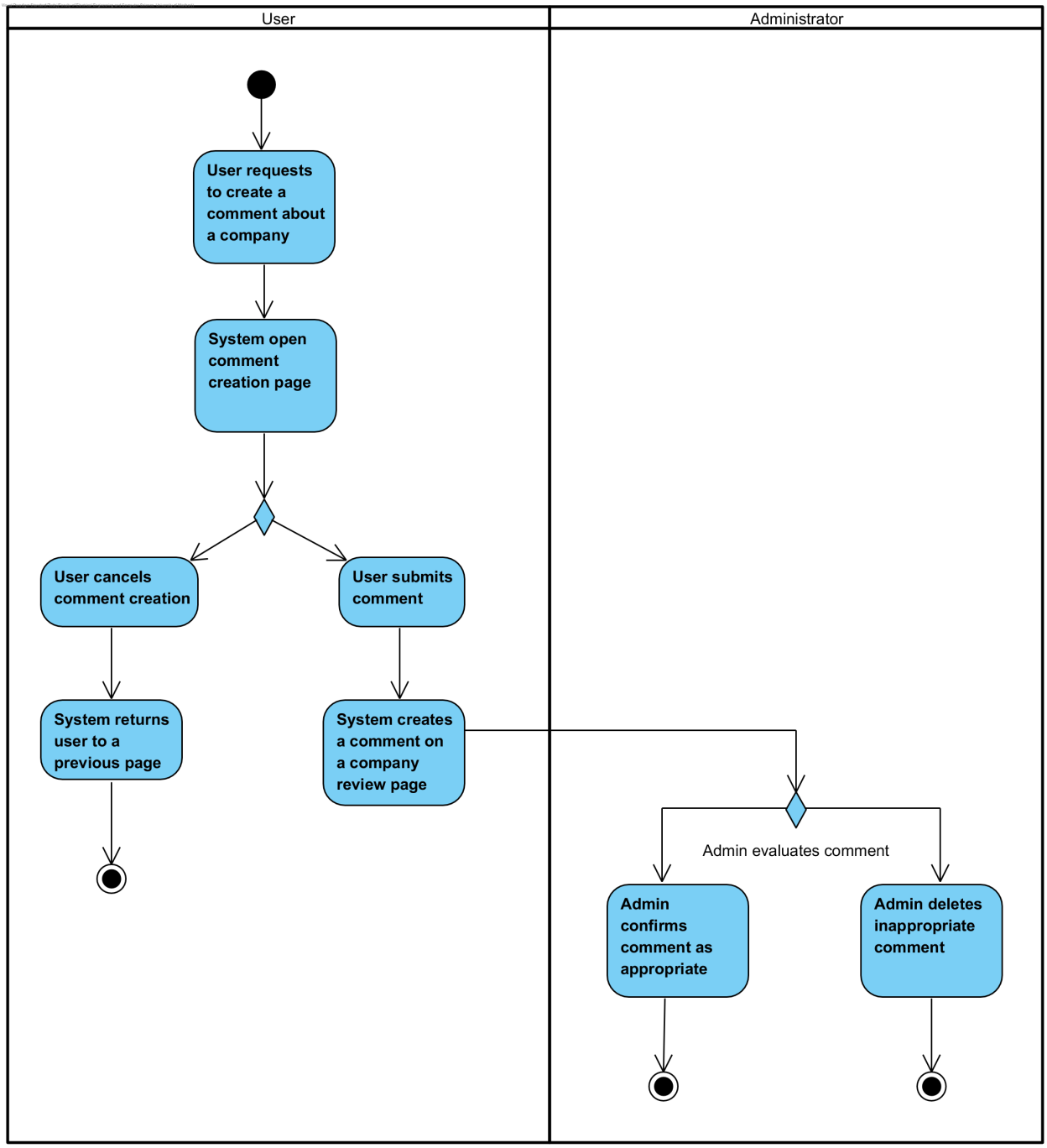

Figure 17. UML activity diagram for adding a review to a company

DIAGRAM 2 DESCRIPTION:

An activity diagram visually presents a series of actions and a flow of control in a system. This activity diagram represents how a platform user can submit a comment about a company. After reading company description, platform user is allowed to create a comment about that company. Once a user selects to write a comment he is prompted with a window in which he can write his comment. If a user chooses to cancel writing, he will be brought back to company review page, otherwise user can choose to submit his comment and it will appear under company description. Once a comment is submitted administrator can read it and decide whether it is appropriate. If a comment is seen as inappropriate it will be deleted.

3.9 System software components (modules) and data flows

Component diagrams are used to visually represent how components are wired together to form larger components or software systems. In this diagram we have represented three most important components of our system. SearchEngine can be used by registered users and visitors. In order to use this functionality, user must perform Search action on website. Once it is done, database will provide user with required data. CompanyRegistration functionality can be accessed from website directly. User will be asked to fill the form and once it is done, the form will be transferred to a database where admin will decide if this form meets all requirements. If admin approves form then account will be created and added to list of already existing accounts. LoginSession requires Authentication component to approve users inserted credentials. To do that, this component sends inserted data to database where authentication process are performed by accessing LoginCredentials component. In order to use PostingReview interface, user must authenticate himself with LoginSession. CommentSection component allows user to post reviews. Every new review is transferred to a database where authors user information is assigned for comment. Posted comment is provided for CommentSection component which uploads it into a database. Rating component can be used only by users who already authenticated themselves. This component sends the value of rating to database(+1/-1). In database value is added to current rating and user is identified to prevent him from rating more than one time.

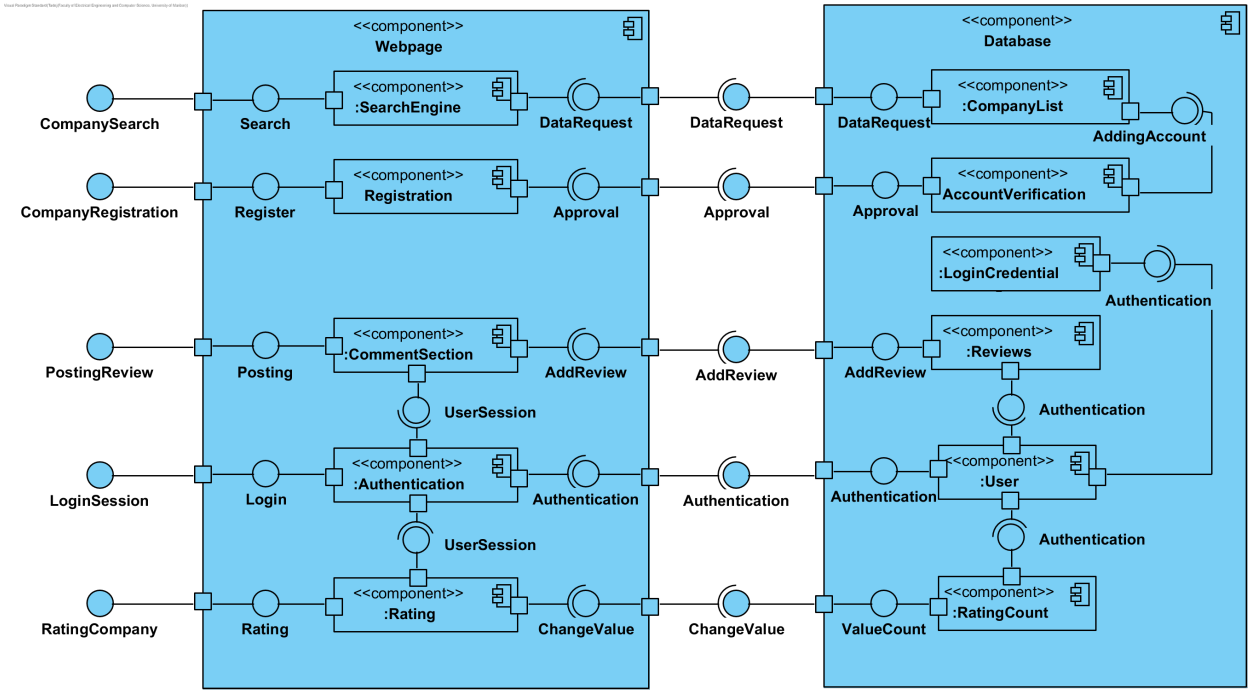

Figure 18. UML component diagram

3.10 Operating environment and hardware components

The Deployment Diagram in the UML models the physical deployment of artifacts on nodes. The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have subnodes, which appear as nested boxes. This diagram shows the static deployment view of our Company Rate system. In the diagram we have nodes: Client, Github and Azure Cloud. Execution Environment Github contains artifact “Code repository” and another node “Github Pages” where HTML, CSS and JS files run. Same with Azure cloud, which contains Database Server with “Microsoft SQL Server” artifact and IIS Server with “EntityFramework.exe” and “CompanyRatesAPI.dll” artifacts. Diagram also shows connection between pieces.

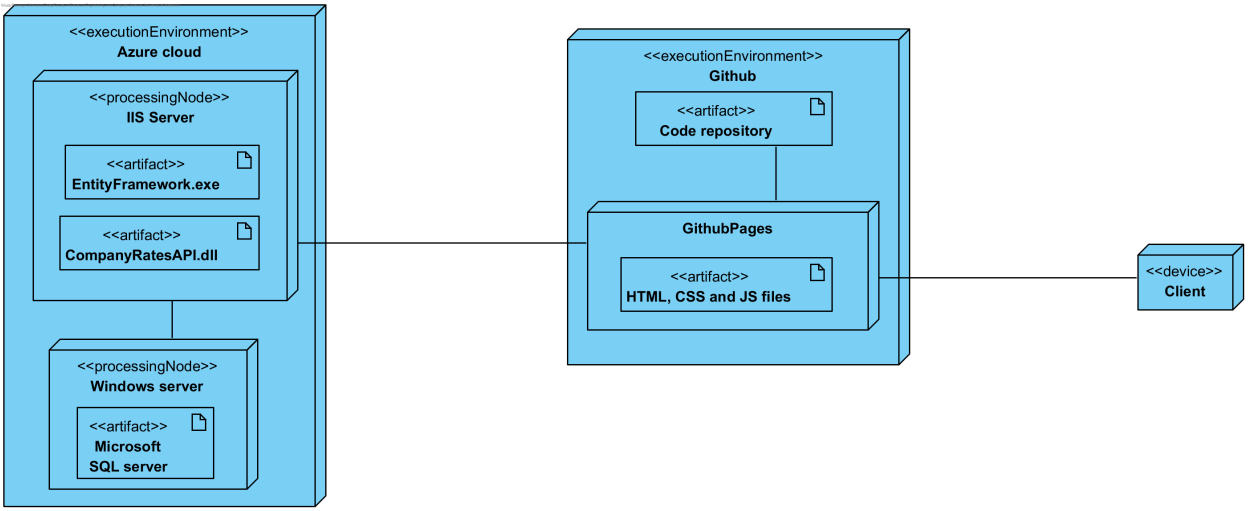

Figure 19. UML deployment diagram

3.10.1 Test cases

We prepared 2 di erent test scenarios and build them using Selenium API in C sharp programming language.

TEST CASE 1:
1. Open Index page
2. Click on Register
3. Click on Register COMPANY account
4. Enter details and register
5. Logout
6. Login as admin
7. Go to registered company
8. Approve company
9. Logout

TEST CASE 2:
1. Open index page
2. Open company
3. Click on Add review
4. Login as user
5. Go to company
6. Click add Review
7. Enter details
8. Rate company
9. Logout