

ADMINISTRATIVE SYSTEM FOR THE DAYCARE "Roskilde"

Chivu Vlad

Computer Science 2021 – 2nd semester Project

1 | Page

Project Report

I Inception	3
Project Planning	3
Requirements workshop.....	4
1. Vision Document.....	4
1.1 Introduction	4
1.2 Positioning	7
1.3 Product Overview	9
1.4 Product features.....	10
II Elaboration	11
Use case Modeling.....	11
1. Use Cases	12
2. Use-case Diagram.....	13
3. System Sequence Diagram	14
4. Sequence Diagram.....	14
Design	15
1. Conceptual Model	15
2. EER Diagram	16
3. Design Model.....	17
APPENDIX A - Supplementary specification.....	19
Appendix B - Glossary	21
Appendix C - Gantt Chart	21

I Inception

Project Planning

Motto: "Plans Are Worthless, But Planning Is Everything." - D. D. Eisenhower



REQUIREMENTS WORKSHOP

- Analysis of the problem and possible solutions (analyze the business environment, delegation of responsibilities)
- A selection of use case scenarios will be discussed (agreement on critical elements to be developed first, constraints of any kind)

ELABORATION

- Continue work on selected use case scenarios
- Design phase (assign contracts to system events to better understand the system use and behavior and design software system components from conceptual classes established prior to this step)

CONSTRUCTION

- Implement the software system solutions in Java
- Testing phase

Requirements workshop

Motto: *"We want to have something done after every iteration."* - Anonymous

1. Vision Document

Version	Date	Description
1.0	14 th of March	First draft.

1.1 Introduction

Starting Questions:

1. *What is the point of having this service?*

Ellen Galinsky, a leading expert in this field says: *"Childcare is an invisible part of the economy."*

Childcare centers allow millions of parents to work, so they can produce value. And maybe even more importantly it is of great significance in the development of a child whose needs are many, to follow the footsteps of a specialized teacher when parents can't be around.

2. *So, who benefits from this?*

A STAKEHOLDER-GOAL ANALYSIS will clarify this and bring more light on the possible system features before starting to think at the user level. For now, we know we are to implement an IT solution at a Daycare Facility where an administrator is mostly involved with managing documents and operations but the scope of this may become bigger if we take into consideration the business environment which will discuss next.

STAKEHOLDER-GOAL ANALYSIS

Administrator:

- wants to save time on administrative tasks
- ease up the process of enrolling a new child
- update profiles (children, parents)
- multitude of records (authorized pickup records, staff records, meal records, etc...)
- possibility to print information easily.

Owner, Director:

- wants to feed information to parents
- provide a proper channel of communication considering child's development
- post news about activities provided
- engage kids

(the real challenge is to make time to provide undivided attention since every child has different needs, so what would be really helpful is a configurable flow of different activities from where the teacher can choose easily)

- getting fees on time (online payments)

(as any other business revenue is of great significance so you want a stable budget with planned payments)

- keep teachers happy

(they are the brand ambassadors of your enterprise; they might enjoy the multitude of tools to keep the kids engaged creating a proper environment for growing their capabilities. Parents will also be very grateful for the well-being of their kids)

- manage staff efficiently

(working hours, work profile, software to calculate their remuneration could be integrated)

- enhance quality of the childcare center

Teacher:

- have a task planner, easy to navigate through
- a reminder of the day-to-day goals to achieve
- print daily reports after the activity plan
- communication channel with management
- post photos and news
- child development profile updated automatically after each day

Parents:

- ✓ Is the website secure?

We plan that the website will be built with Secure Sockets Layer (SSL) certificate.

- ✓ Is the info backed up safely?

Yes, all of the information is automatically backed up on a daily basis. Data will be stored on Google Web Services cloud infrastructure.

1.2 Positioning

1.2.1 Problem statement

In this business much of the repetitive tasks have to be alleviated so you can save more time for the kids. Communication with parents is also important as well as educational techniques that should be integrated easily in the system and used to enhance the kids curiosity and discover his inclinations.

1.2.2 Competition

Competition is fierce as multiple solutions already exist, some of them offer free trials or free limited versions.

Reference to: <https://www.goodfirms.co/blog/best-free-childcare-software-for-your-business>

1.2.3 Business Opportunity

We can only differentiate if we manage to succeed in designing a management system which leaves space for integration with other specialized services which should go above and beyond. We should take the child development to another level, a service orientated system to be of big help to teachers who will use these techniques. We should also grow a continuous partnership with the parents, taking the load off of their shoulders but involving them in the process with seamless effort from their side.

1.2.4 Proposition

Our mission is to have a minimum viable product with enough features to prove it's worth.

After analyzing the goals and stakes, we decided that the application will be fully integrated in the cloud, meaning it will be a web application that will work on a remote server of a cloud provider. This way we do not have to worry about data because cloud providers have information backed up in multiple places.

Other decisive factors were:

- the requirements to run this system are fairly minimum you only need an installation of a browser; (no specialized personnel required for installation)
- the chance to connect to other services that we may want; (the design of the application makes it easy to plug them in our daycare management system to help with always changing business needs)
- a common platform that is to work on many devices around the daycare for other users too, like teachers who have lesson plans;
- the need to communicate over the internet with parents to keep them updated;
- if our needs increase, configuration is possible in the cloud where we can scale to our needs and pay for what we consume (computing power and database storage);
- and maybe one of the most important business needs, have a safe place for our data, not in one place but duplicated in many instances of different servers.

So, we place our service in the Software-as-a-Model (SaaS) delivery model, meaning that we will use the internet to deliver our application, which will run in a web browser. This means that all potential technical issues with data, servers and storage are managed by us the vendors of this service, freeing the customer of any tech concern. Customer needs just to login and continue its normal business.

1.3 Product Overview

Our management system will be composed of many services grouped into modules:

The *basic module* will include the first needs of the daycare management system which will be free to use. With this module the daycare can manage the onboarding of children, saving and retrieving the basic information about them and their parents and plan some activities for the day with an attendance list that is to be saved at the end. The full list of features that are coming at a small price is to be discussed after presentation of the initial offering.

An instance of the daycare will be created when signing up for the account after which a *dashboard* with different actions will be automatically created (classes of kids can be created here, staff can be added and assigned to classes, schedules can be worked on with activities, activities can be created with teachers assigned to them, (in a different module teachers can follow on their devices the teaching plan while in class) and an attendance list should also be generated.

The other modules to be integrated at a cost:

- ✚ *parent portal* with access to daycare activities and child records, easy billing, and payment processing, help with legal documentation, subsidy documentation, enrolment from home possible, send and receive files.
- ✚ *teacher portal* with access to multiple activities and reports that can be generated and uploaded to the cloud.

1.4 Product features

What we can offer now:

BASIC DAYCARE MANAGEMENT SYSTEM

- easy admin login
- register family information
- view of family information with parents and children
- register parents and children
- update and delete information

LIMITED CLOUD COMPUTING & STORAGE

- remote database that can be stored on personal computer too

If willing to subscribe:

EASY BILLING AND PAYMENT PROCESSING

MANAGE CLASSES OF CHILDREN

MANAGE STAFF

GENERATE REPORTS

CUSTOMISATION

TECHNICAL SUPPORT

MORE STORAGE

UPDATES & UPGRADE

II Elaboration

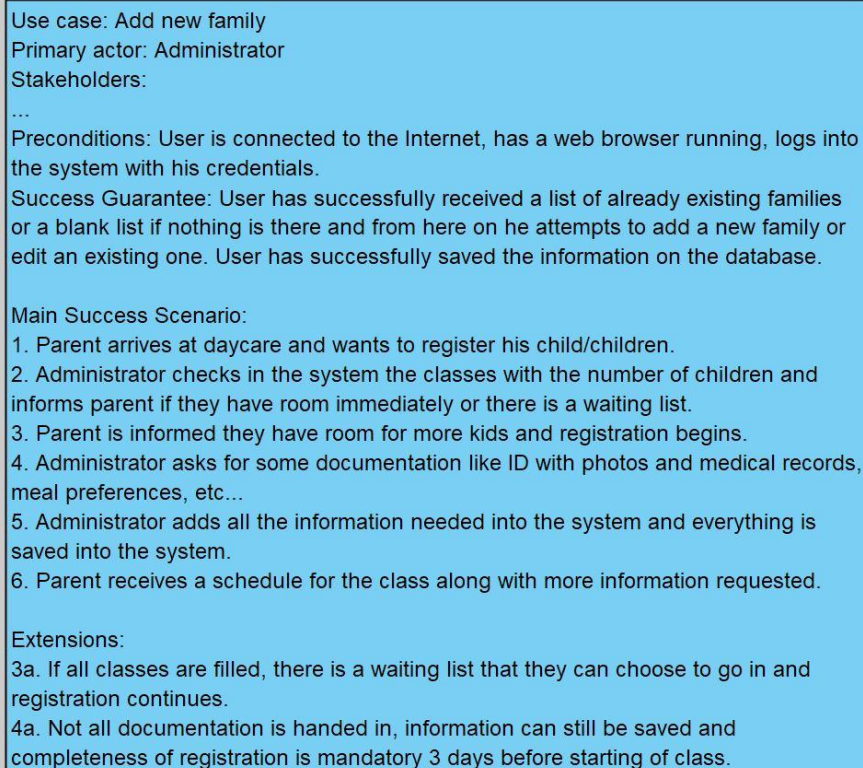
Use case Modeling

Motto: *"The indispensable first step to getting the things you want out of life: decide what you want."* - Ben Stein

Artifacts:

1. Use Cases
2. Use Case Model
3. System Sequence Diagram
4. Sequence Diagram

It is of great significance to act with many-eyes when modelling use cases, as we know from service design customer wishes carry different perspectives. Different by many factors which we can make optional and reusable leaving space for customization in this way if we design with respect to some of the most used practices.



Use case: Add new family
Primary actor: Administrator
Stakeholders:
...
Preconditions: User is connected to the Internet, has a web browser running, logs into the system with his credentials.
Success Guarantee: User has successfully received a list of already existing families or a blank list if nothing is there and from here on he attempts to add a new family or edit an existing one. User has successfully saved the information on the database.

Main Success Scenario:

1. Parent arrives at daycare and wants to register his child/children.
2. Administrator checks in the system the classes with the number of children and informs parent if they have room immediately or there is a waiting list.
3. Parent is informed they have room for more kids and registration begins.
4. Administrator asks for some documentation like ID with photos and medical records, meal preferences, etc...
5. Administrator adds all the information needed into the system and everything is saved into the system.
6. Parent receives a schedule for the class along with more information requested.

Extensions:

- 3a. If all classes are filled, there is a waiting list that they can choose to go in and registration continues.
- 4a. Not all documentation is handed in, information can still be saved and completeness of registration is mandatory 3 days before starting of class.

1. Use Cases

Use case: Add new family
Primary actor: Administrator
Stakeholders:

...

Preconditions: User is connected to the Internet, has a web browser running, logs into the system with his credentials.

Success Guarantee: User attempts to create a new family and has successfully saved the information on the system.

Main Success Scenario:

1. Parent arrives at daycare and wants to register his child/children.
2. Administrator checks in the system the classes with the number of children and informs parent if they have room immediately or there is a waiting list.
3. Parent is informed they have room for more kids and registration begins.
4. Administrator asks for some documentation like ID with photos and medical records, meal preferences, etc...
5. Administrator adds all the information needed into the system and everything is saved.
6. Parent receives a schedule for the class along with more information requested.

Extensions:

- 3a. If all classes are filled, there is a waiting list that they can choose to go in and registration continues.
- 4a. Not all documentation is handed in, information can still be saved and completeness of registration is mandatory 3 days before starting of class.

Use case: Manage Schedule
Primary actor: Administrator
Stakeholders:

...

Preconditions: User is connected to the Internet, has a web browser running, logs into the system with his credentials.

Success Guarantee: User attempts to create a new schedule for the class and has successfully saved the information on the system.

Main Success Scenario:

1. Administrator accesses Manage class, and a list of current classes should be listed.
2. Administrator selects one by double clicking on it, and then can proceed with editing information.
3. Administrator can add new schedule now.
4. Information about hours and activities must be completed.
5. Form is then submitted and a confirmation message should come back.

Extension:

- 3a. Administrator can edit the schedule if one is already present.

Use case: Add new class
Primary actor: Administrator
Stakeholders:

...

Preconditions: User is connected to the Internet, has a web browser running, logs into the system with his credentials.

Success Guarantee: User attempts to create a new class and has successfully saved the information on the system.

Main Success Scenario

1. Administrator accesses Manage Class, where will find a list of current classes if any.
2. Administrator then proceeds in creating a class, where she/he will have to type in some information about the class. Here we can have a maximum number of children grouped by age and having diverse activities. Next will have to assign a teacher and finally add kids to the class.
3. Administrator saves the information by submitting the form and a confirmation message should come back.

Use case: Register staff
Primary actor: Administrator
Stakeholders:

...

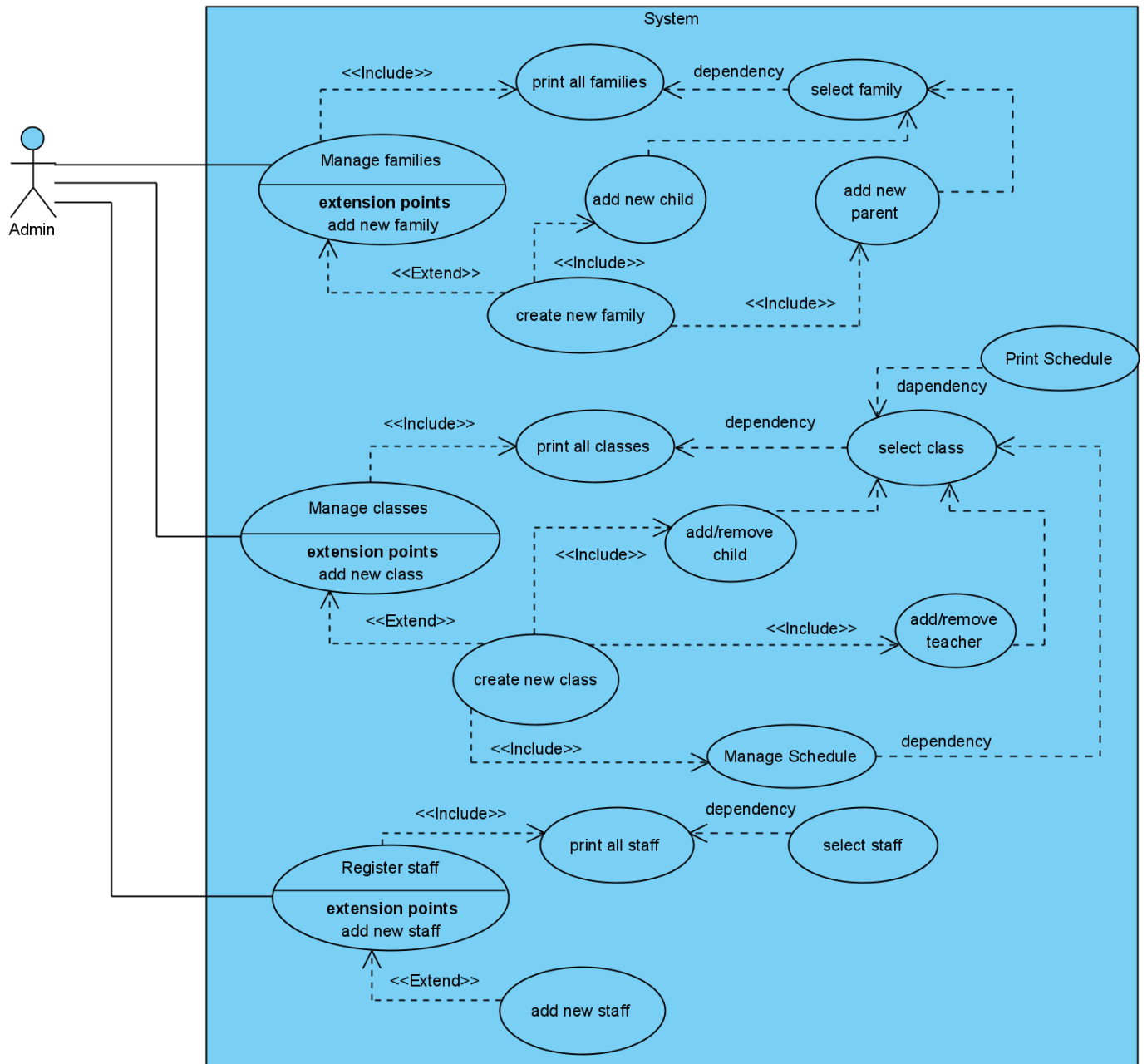
Preconditions: User is connected to the Internet, has a web browser running, logs into the system with his credentials.

Success Guarantee: User attempts to register a new staff member in this case a teacher, and has successfully saved the information on the system.

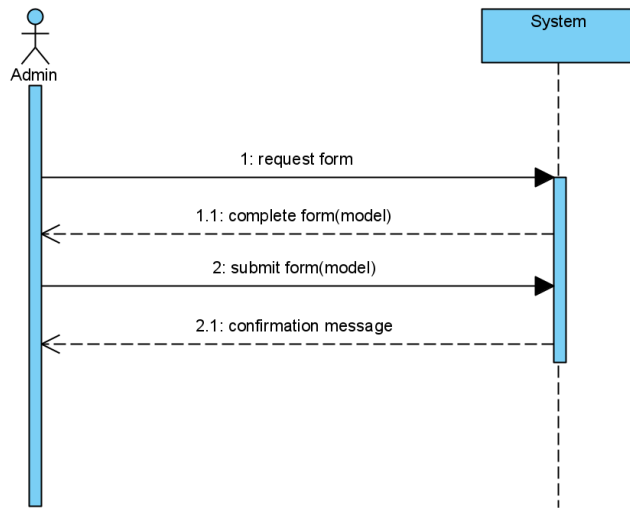
Main Success Scenario:

1. Administrator accesses Register staff and a list of already employed staff should be listed.
2. Administrator can easily add a new staff, where options for his information should be completed.
3. Administrator saves the information at completion by submitting the form and a confirmation message should come back.

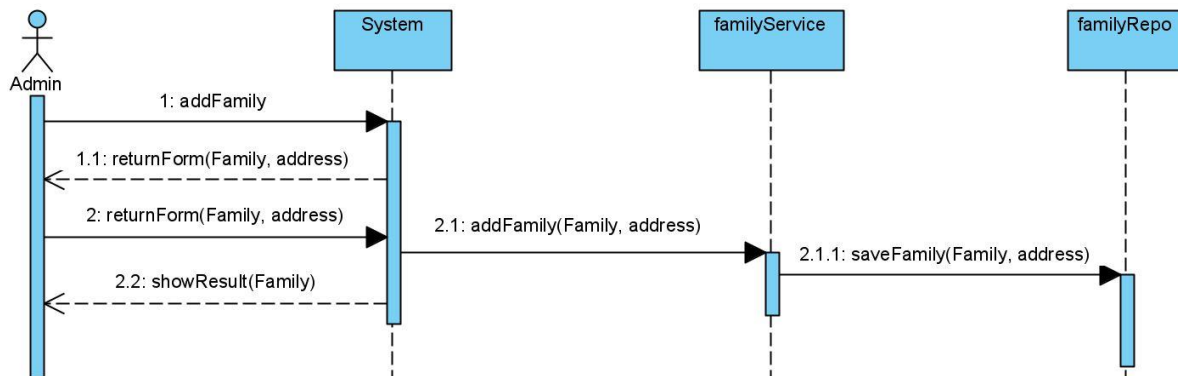
2. Use-case Diagram



3. System Sequence Diagram

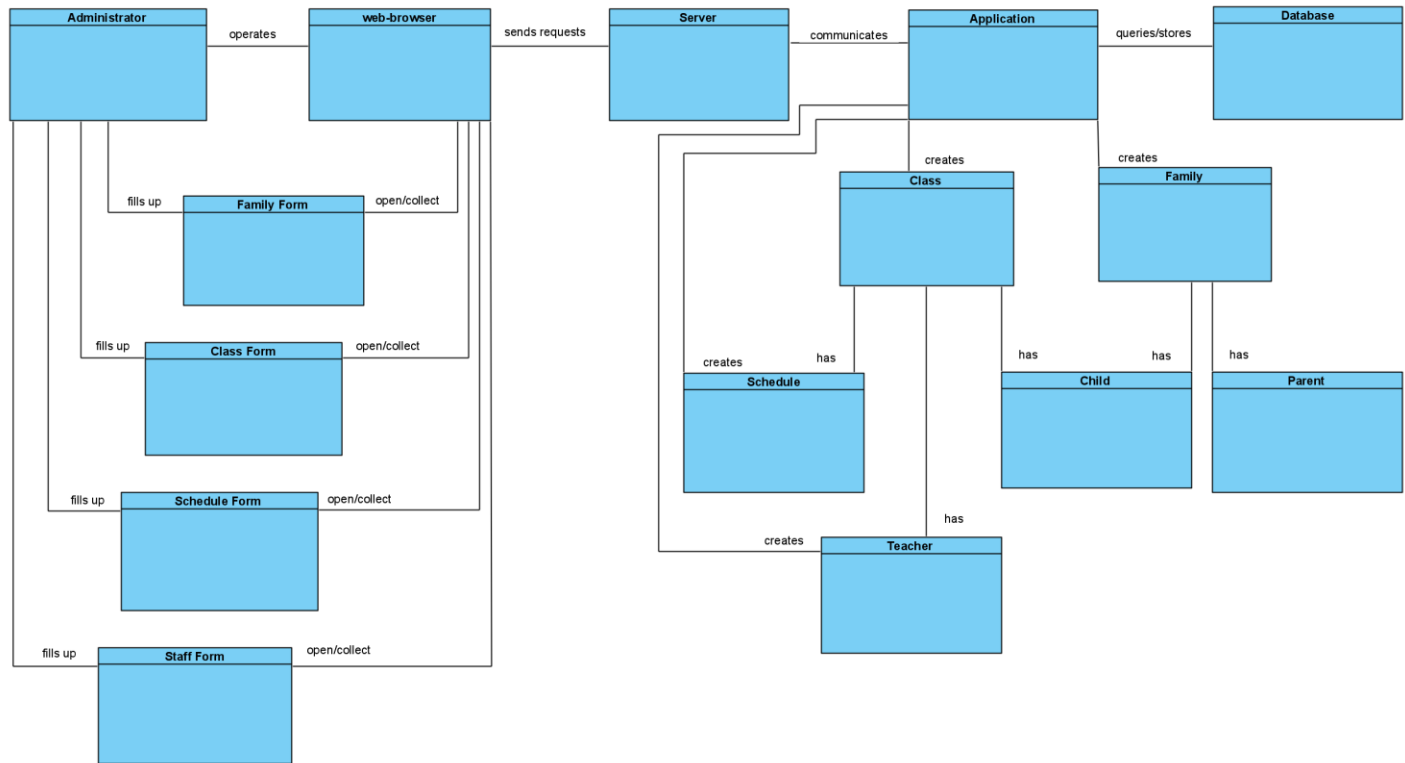


4. Sequence Diagram

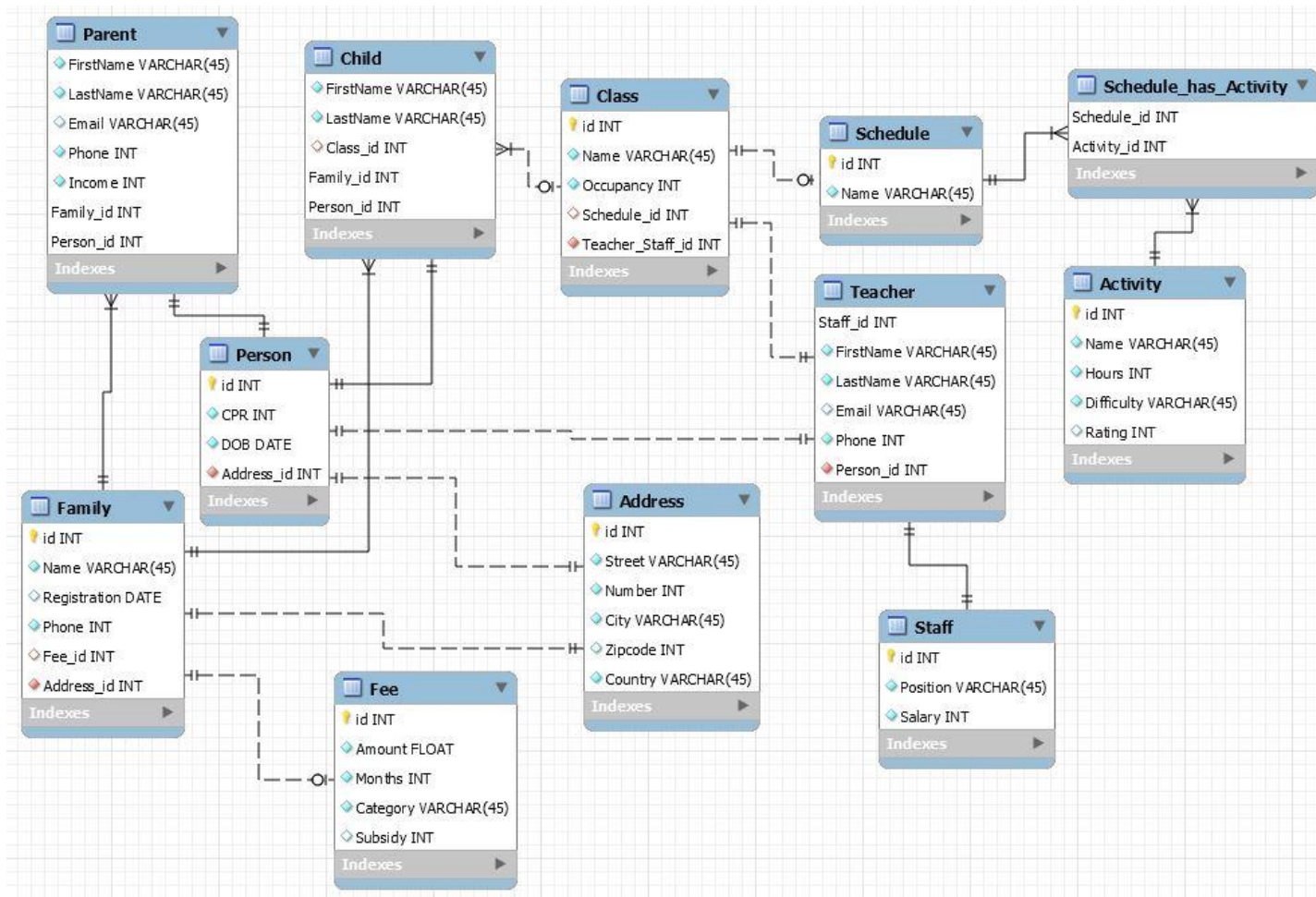


Design

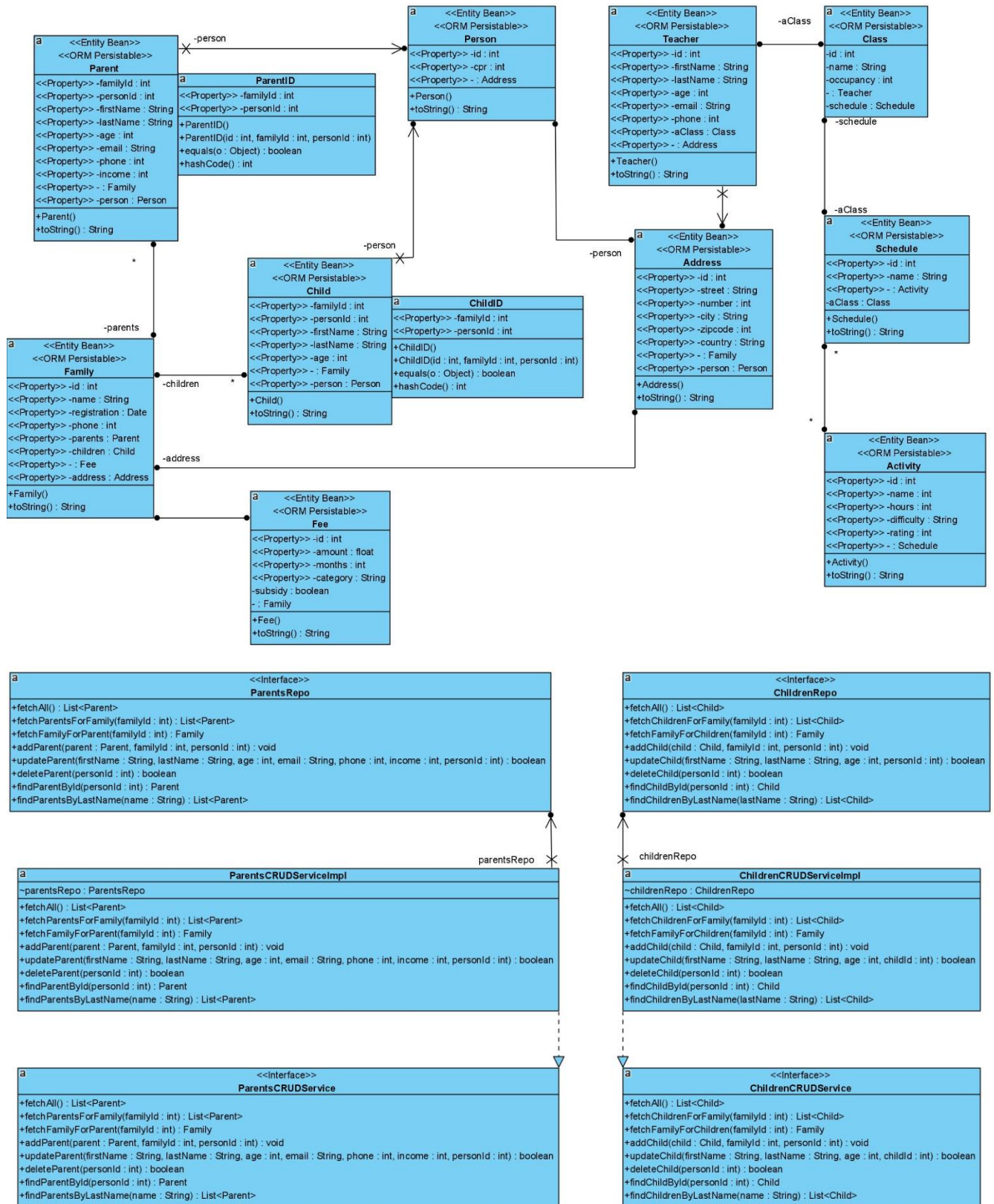
1. Conceptual Model



2. EER Diagram



3. Design Model



Work Report

1st Iteration 15/03/2021 (Requirements)

- we did some business analysis (Artifact: Stakeholders Goal Analysis & Vision Document).
- we defined the scope of the project and established the use case scenarios (features) that we are going to design and implement. (Artifact: Brief Use Cases).
- we decided the needs and technologies that we are going to use to implement our solution (Artifact: Supplementary Specification).

2nd Iteration 16/03/2021 (Modelling)

- we started diagramming the Use Case Model, a fully dressed Use Case, the System Sequence Diagram, followed by a first draft of the Conceptual Model.

3rd Iteration 17/03/2021 (Modelling)

- we started the first draft of our database model using MySQL Workbench. We used the EER model to generate the SQL.

4th Iteration 18/03/2021 (Software development)

- we began the construction phase; we followed the MVC pattern and created our POJOs, our repositories and our services and the controllers that are managing our endpoints.

5th Iteration 19/03/2021 – 21/03/2021 (Testing and Debugging)

- designed our views and started debugging; at the end we had our first working software but not ready for production, just a proof of concept.

21/03/2021 – 23/03/2021

- more software development, testing and reverse engineering;

APPENDIX A - Supplementary specification

Introduction

This section documents the requirements and specifics not captured in the uses cases.

We must mention that since we are not ready for production this application is not fully secured. It will be tested while running on a Tomcat webserver provided by Spring Boot framework along with some other dependencies of the project that will be mentioned in the *.pom file*. We will be using Maven for dependency management. We will also rely on a remote instance of a MySQL database running on gearhost. This architecture of the program won't change the way the application will work in production.

So, for the purpose of testing and showcasing this system we will use the local environment as a webserver. From inside IntelliJ IDE we can generate a *.jar file*, which can run on any machine with Maven and Java installed on. Later on, the application can be containerized along with all its dependencies in a Docker container and deployed on a server and accessed from any web-browser with no installation on the user's end.

Functionality

- Easy to use, user needs just to login to the website.
- Pluggable Business Rules depending on the Company wishes.

Usability

- A common interface for all pages, options are visible and remain in the same place throughout the website.

Reliability

- The risk remains if internet is down, and this is a bottleneck; the database although is permanently updated and kept safe in the cloud.

Performance

- Application is performing generally well in terms of speed; majority of errors are handled, still improvements can be made here.

Supportability

- The application is to work on all major web browsers that support HTML5.

Implementation

- The application is to be built with Java 11 and Spring Boot

Open-Source Components

- MySQL database
- Remote database hosting (gearhost)
- The pages are served with Tomcat
- web application hosting (Google or Heroku)

Tools

- IntelliJ IDE
- Maven for dependency management

Domain Rules

- User authentication required
- Once browser window is closed the user is automatically logged out

Legal issues

- Making sure all terms and conditions of GDPR regarding data processing are satisfied

Appendix B - Glossary

SSL is the standard security technology for establishing an encrypted link between a web server and a browser. This link ensures that all data passed between the server and our customer's browsers remain private and integral. SSL is an industry standard for securing online transactions so is used by online banking and payments sites and.

Google Web Services is one of the most secure cloud computing environments available with highly secure data centers utilizing state-of the art electronic surveillance and access control systems, including 24×7 trained security guard protection.

Appendix C - Gantt Chart

