

ИНФОРМАЦИОНИ СИСТЕМИ

АЕРОДРОМ

Тамара Ђукић, 1051/2023
mi231051@alas.matf.bg.ac.rs

Тамара Јевтимијевић, 1045/2023
mi231045@alas.matf.bg.ac.rs

Милош Милаковић, 1052/2021
mi211052@alas.matf.bg.ac.rs

професор: др Саша Малков
асистент: Дара Милојковић

Београд 2023.



Садржај

1	Увод	3
2	Анализа система	3
2.1	Функционисање аеродрома	3
2.2	Учесници у систему	3
2.3	Коришћени дијаграми и алати	4
3	Процеси и случајеви употребе	5
3.1	Административни случајеви употребе	5
3.1.1	Уношење корисника у систем	5
3.1.2	Брисање корисника из система	6
3.1.3	Уношење авио-компаније у систем	9
3.1.4	Брисање авио-компаније из система	9
3.2	Резервација аеродрома (писте):	10
3.3	Праћење летова	14
3.4	Обрада захтева	14
3.4.1	Чекирање путника	14
3.4.2	Чекирање запослених	16
3.5	Одржавање авиона	18
3.5.1	Пријава кvara	18
3.5.2	Поправка кvara	18
3.5.3	Уношење потребних делова у систем	18
4	База података	18
5	Софтверска архитектура	18
6	Кориснички интерфејс	18

1 Увод

Овај рад представља пројекат из предмета *Информациони системи* на мастер студијама Математичког факултета, Универзитета у Београду. Рад описује информациони систем аеродрома. Свесни смо да у данашњем времену, много људи на дневном нивоу користи аеродроме, па је у складу са тим потребно да сваки аеродром има систем који савршено ради у сваком моменту. На пример, што се тиче наше домаће авио-компаније, Air Serbia је у првих шест месеци обавила 63% више летова и превезла 87% више путника него за исти период 2022. године [1]. Дакле, наша идеја је да развијемо систем, који ће се користити приликом рада аеродрома. У оквиру анализе система препознати су основни процеси и учесници у систему. За израду дијаграма у оквиру пројекта коришћени су следећи алати:

- Visual Paradigm

2 Анализа система

2.1 Функционисање аеродрома

Информациони систем, који развијамо, развијамо за потребе аеродрома. На аеродрому постоје две главне групе људи, а то су путници и радници. Како функционише аеродром? Путник дође на аеродром, чекира се, преда свој пртљак и оде на гејт да чека укрцавање. То звучи једноставно, али иза свега тога постоји један дугачак процес. Да би путник уопште могао да лети, потребан је авион. Да би авион постојао потребно је да постоји авио-компанија, која поседује тај авион и која има уговор са аеродромом. Под уговором са аеродромом, подразумева се то да авио-компанија може да користи тј. да резервише писту аеродрома и да је користи за полетање и слетање својих авиона. Даље да би ти авиони били савршено исправни у сваком моменту потребно је да постоје људи који ће одржавати и контролисати те авионе. Затим да би авиони безбедно летели постоје људи који ће их пратити и усмеравати, итд. Као што видимо систем једног аеродрома је веома комплексан и компликован. На све то, у сваком моменту, мора савршено да функционише.

На слици 1 налази се дијаграм, који приказује учеснике система и њихове послове.

2.2 Учесници у систему

Основна подела учесника је на **путнике, запослене и авио-компаније**. Запослени на аеродрому:

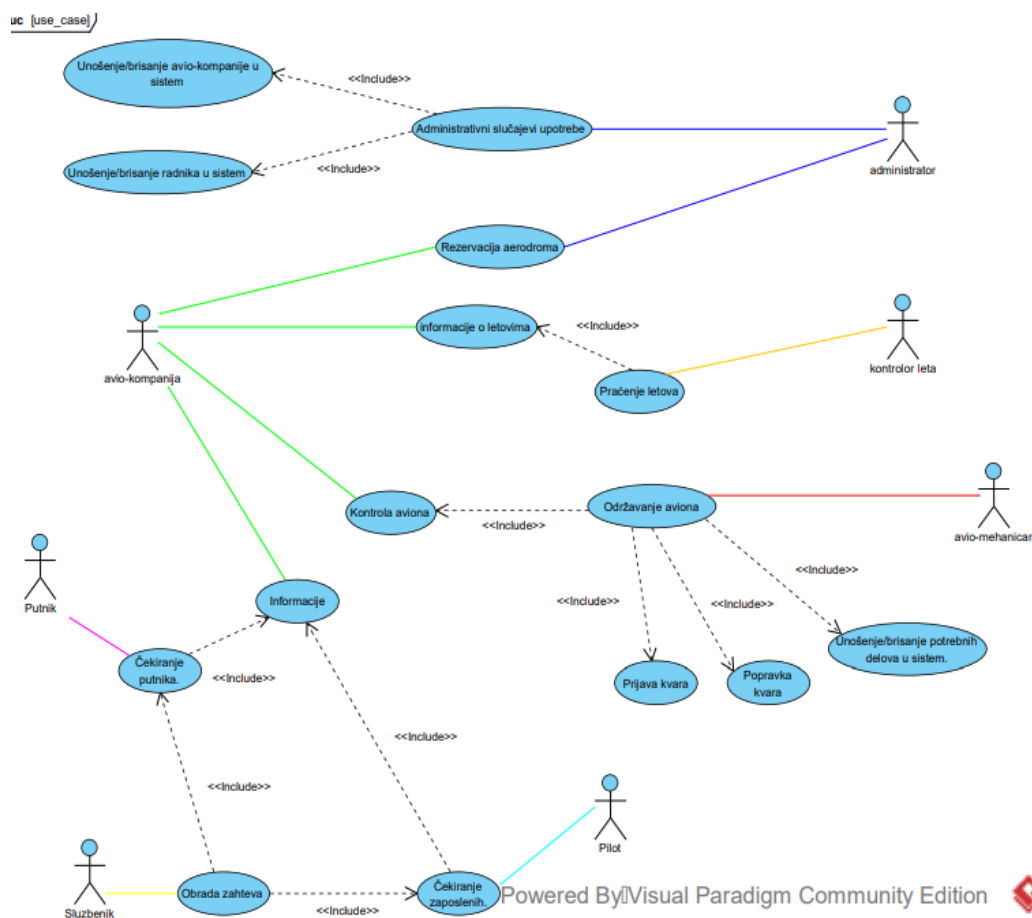
- Администратор - има улогу да одржава базу података
 - Администратор задужен за резервацију писте.
 - Администратор задужен за уношење и брисање запослених.
 - Администратор задужен за уношење и брисање авио-компанија.
- Авио-механичар - задужен за поправку, проверу и одржавање авиона;
- Службеник - задужен за опслуживање путника;
- Контролор лета - задужен за праћење и навигирање авиона.

2.3 Коришћени дијаграми и алати

Током израде рада коришћени су следећи дијаграми:

- UML дијаграми:
 1. Дијаграм случајева употребе;
 2. Дијаграм активности;

За израду свих UML дијаграма коришћен је алат *Visual Paradigm - Online/Community Edition* [2].



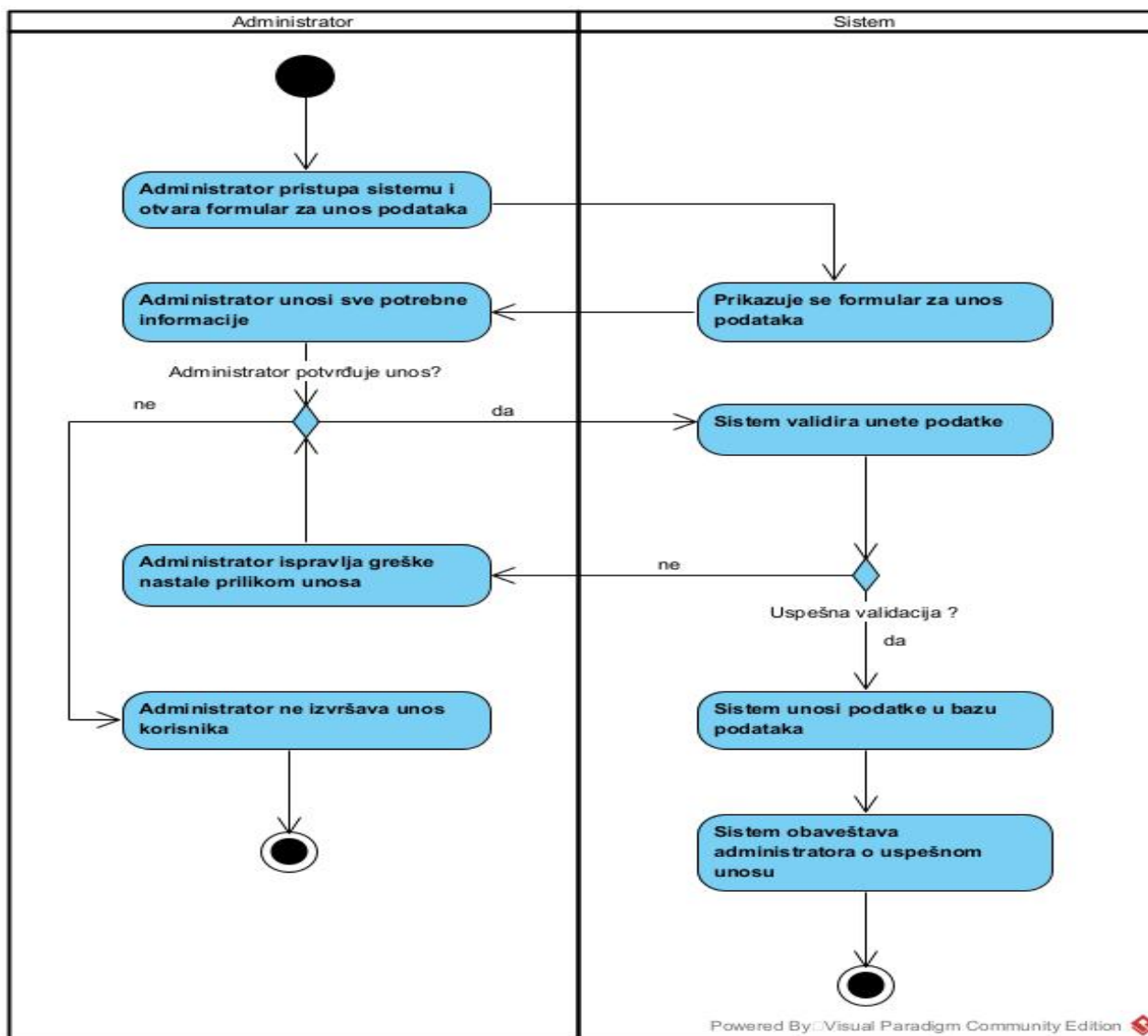
Слика 1: Дијаграм случајева употребе

3 Процеси и случајеви употребе

3.1 Административни случајеви употребе

3.1.1 Уношење корисника у систем

- **Кратак опис:** Администратор уноси информације о новом кориснику који у моменту уноса не постоји у систему. Систем прихвата информације и валидира их и враћа потврду о успеху или неуспеху уноса.
- **Учесници:**
 - *Администратор*
- **Предуслови:** Администратор има приступ систему и има информације о кориснику ког уноси.
- **Постуслови:** Корисник је успешно додат у систем.
- **Основни ток:**
 1. Администратор приступа систему и започиње поступак уноса корисника у систем тако што отвори формулар.
 2. Систем приказује форму за унос података.
 3. Администратор уноси све потребне информације у формулар.
 4. Администратор потврђује унос.
 5. Систем валидира унете податке.
 6. Систем уноси податке у базу података.
 7. Систем обавештава администратора да је операција успешно извршена.
- **Алтернативни токови:**
 - **A1. Администратор одустаје од уноса података.** Ако у кораку 4 администратор кликне на дугме одустани подаци се не шаљу систему и неће бити унети у исти.
 - **A2. Унети подаци о компанији нису исправни.** Ако у кораку 5 систем препозна да су неки од унетих података невалидни, избациће грешку и обавестити администратора које поље је неваљидно. Администратор исправља грешку и наставља даље од 4. корака главног тока.

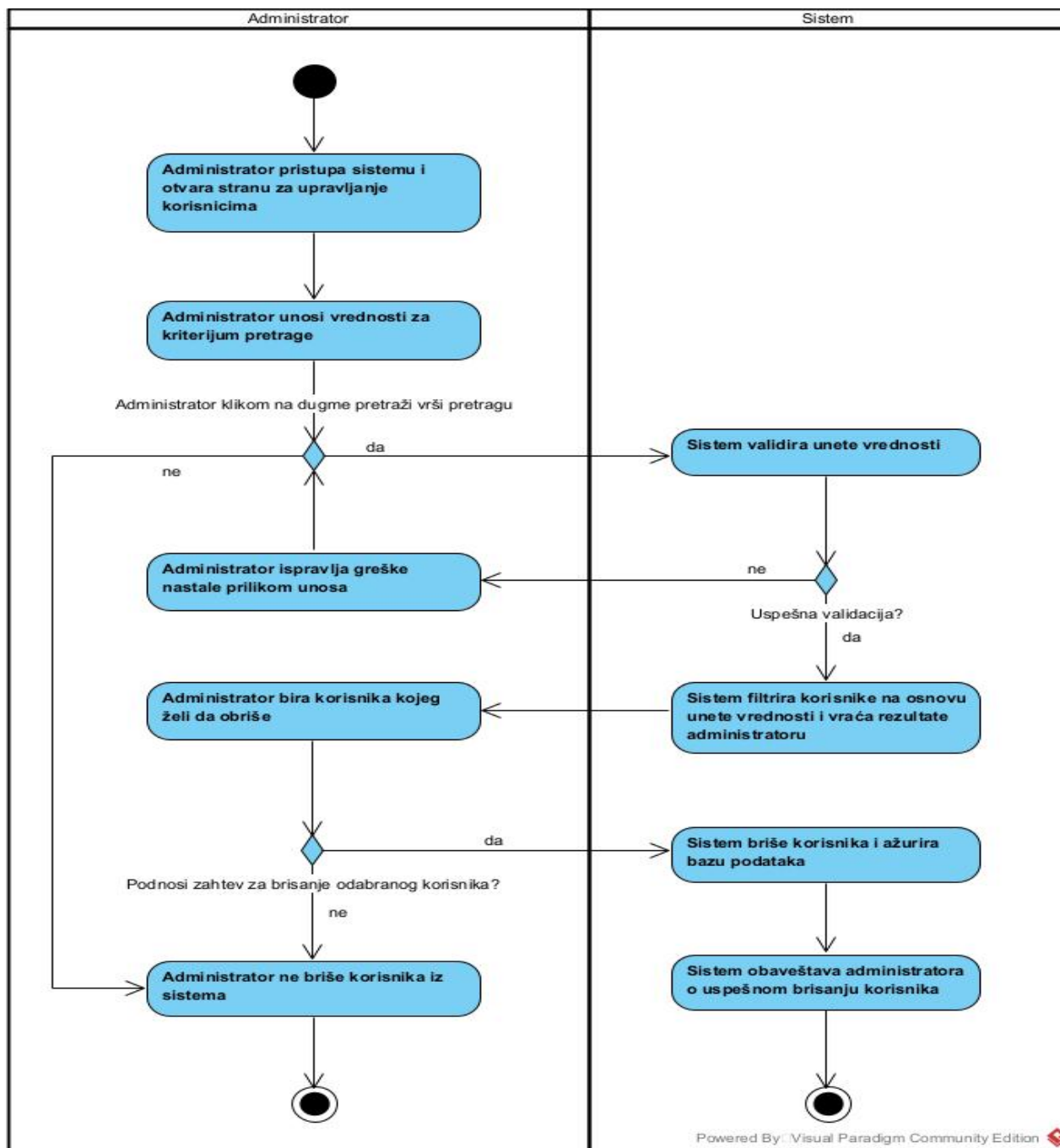


Слика 2: Дијаграм активности - Додавање корисника у систем

3.1.2 Брисање корисника из система

- **Кратак опис:** Администратор брише корисника који постоји у систему. Систем ажурира базу и враћа информацију о успешности захтева.
- **Учесници:**
 - Администратор
- **Предуслови:** Администратор има приступ систему и има информације о кориснику ког жели да обрише.

- **Постуслови:** Корисник је успешно обрисан из система.
- **Основни ток:**
 1. Администратор приступа систему и отвара страну за управљање корисницима.
 2. Администратор уноси вредност за критеријум претраге.
 3. Администратор кликом на дугме врши претрагу.
 4. Систем валидира унос.
 5. Систем на основу вредности за критеријум претраге филтрира кориснике и враћа резултате администратору.
 6. На основу добијених корисника, администратор бира одговарајућег и подноси захтев систему за брисање.
 7. Систем брише корисника и ажурира базу.
 8. Систем обавештава администратора да је операција успешно извршена.
- **Алтернативни токови:**
 - A1. **Администратор одустаје од уноса података.** Ако у кораку 3 или 6 администратор одустане од брисања корисника, захтев се не шаље систему и корисник неће бити обрисан.
 - A2. **Унети подаци за филтрирање нису исправни.** Ако у кораку 4 систем препозна да је вредност за филтрирање невалидна, избациће грешку и обавестити администратора о грешци. Администратор исправља грешку и наставља даље од 3. корака главног тока.



Слика 3: Дијаграм активности - Брисање корисника из система

3.1.3 Уношење авио-компаније у систем

- **Кратак опис:** Администратор уноси информације о новој авио-компанији која у моменту уноса не постоји у систему. Систем прихвата информације и валидира их и враћа потврду о успеху или неуспеху уноса.
- **Учесници:**
 - *Администратор*
- **Предуслови:** Администратор има приступ систему и има информације о авио-компанији коју уноси.
- **Постуслови:** Авио-компанија је успешно додата у систем.
- **Основни ток:**
 1. Администратор приступа систему и започиње поступак уноса авио-компаније у систем тако што отвори формулар.
 2. Систем приказује форму за унос података.
 3. Администратор уноси све потребне информације у формулар.
 4. Администратор потврђује унос.
 5. Систем валидира унете податке.
 6. Систем уноси податке у базу података.
 7. Систем обавештава администратора да је операција успешно извршена.
- **Алтернативни токови:**
 - **A1. Администратор одустаје од уноса података.** Ако у кораку 4 администратор кликне на дугме одустани подаци се не шаљу систему и неће бити унети у исти.
 - **A2. Унети подаци о авио компанији нису исправни.** Ако у кораку 5 систем препозна да су неки од унетих података невалидни, избациће грешку и обавестити администратора које поље је невалидно. Администратор исправља грешку и наставља даље од 4. корака главног тока.

3.1.4 Брисање авио-компаније из система

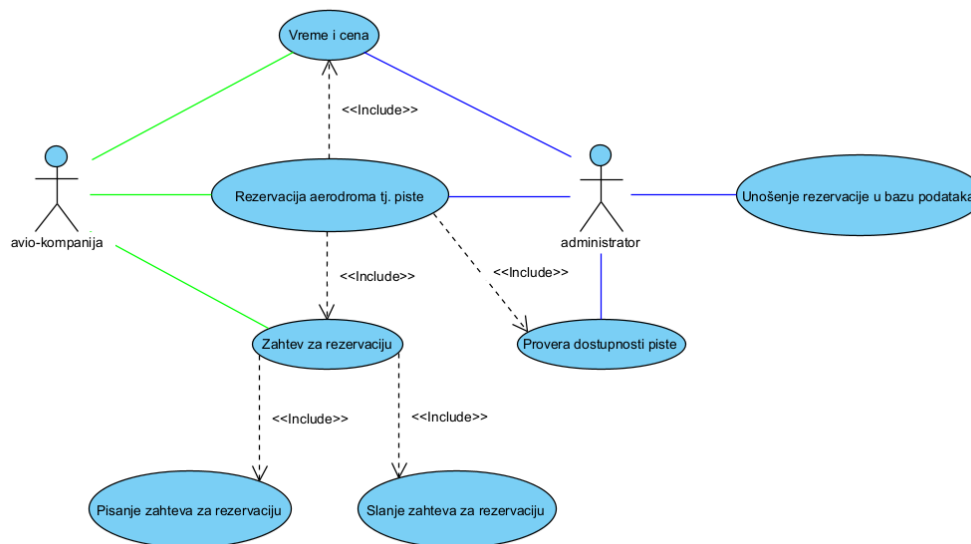
- **Кратак опис:** Администратор брише авио-компанију која постоји у систему. Систем ажурира базу и враћа информацију о успешности захтева.
- **Учесници:**
 - *Администратор*
- **Предуслови:** Администратор има приступ систему и има информације о авио-компанији коју жели да обрише.
- **Постуслови:** Авио-компанија је успешно обрисана из система.
- **Основни ток:**

1. Администратор приступа систему и отвара страну за управљање авио-компанијама.
2. Администратор уноси вредност за критеријум претраге.
3. Администратор кликом на дугме врши претрагу.
4. Систем валидира унос.
5. Систем на основу вредности за критеријум претраге филтрира авио-компаније и враћа резултате администратору.
6. На основу добијених авио-компанија, администратор бира одговарајућу и подноси захтев систему за брисање.
7. Систем брише авио-компанију и ажурира базу.
8. Систем обавештава администратора да је операција успешно извршена.

• **Алтернативни токови:**

- A1. **Администратор одустаје од уноса података.** Ако у кораку 3 или 6 администратор одустане од брисања авио-компаније, захтев се не шаље систему и корисник неће бити обрисан.
- A2. **Унети подаци за филтрирање нису исправни.** Ако у кораку 4 систем препозна да је вредност за филтрирање невалидна, избациће грешку и обавестити администратора о грешци. Администратор исправља грешку и наставља даље од 3. корака главног тока.

3.2 Резервација аеродрома (писте):



Слика 4: Дијаграм случаја употребе - Резервација аеродрома (писте)

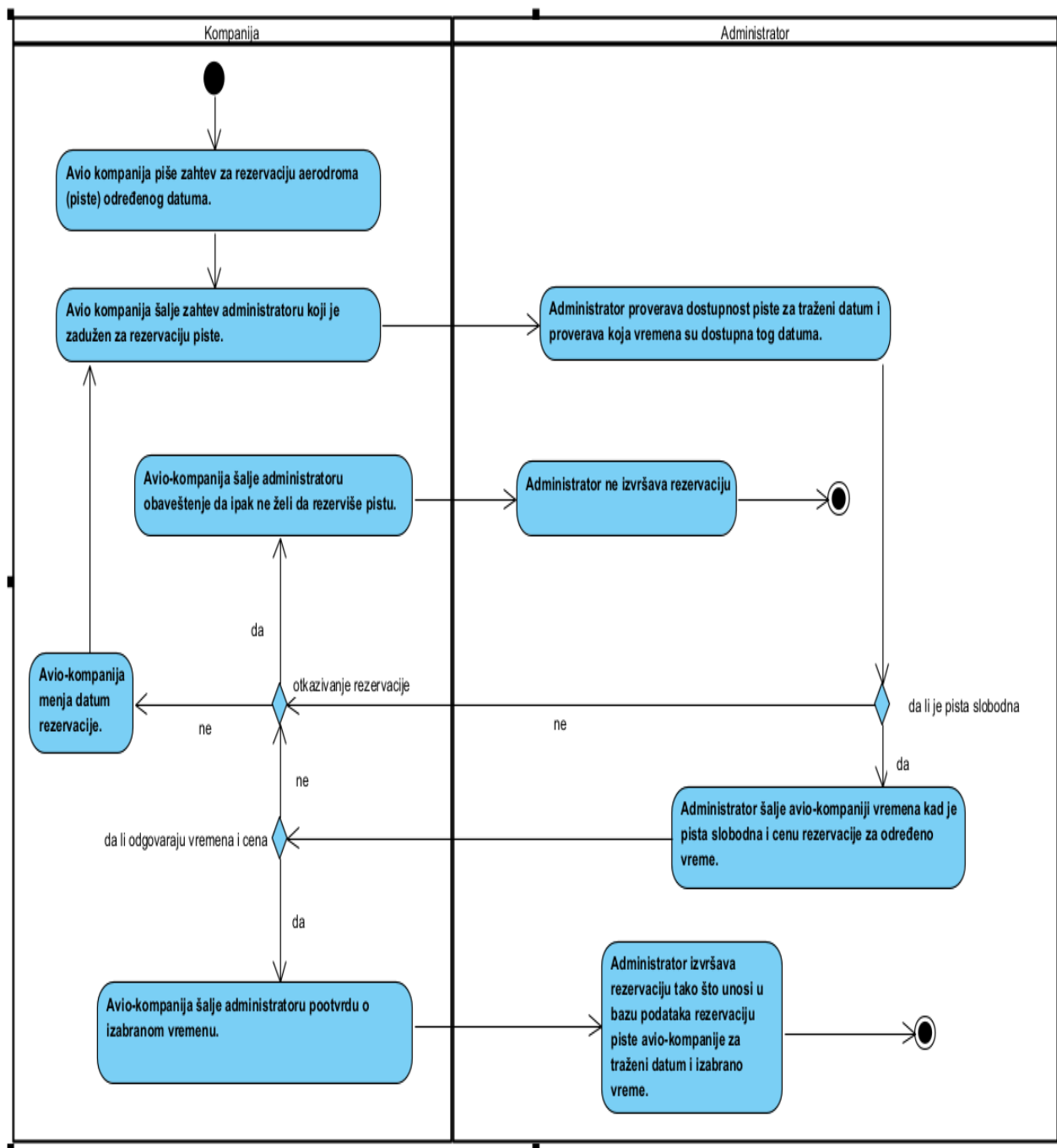
- **Кратак опис:** Авио-компанија контактира са администратором аеродрома задуженим за резервације писте у циљу резервисања писте за полетање или слетање својих авиона. Са администратором може да се договара око датума, времена као и цене резервисања писте.
- **Учесници:**
 - *Авио-компанија;*
 - *Администратор који је задужен за резервације писте.*
- **Предуслови:** Авио-компанија поседује барем један авион и регистрована је тј. да има уговор о пословању са аеродромом.
- **Постуслови:** Авио-компанија или јесте или није резервисала писту.

- **Основни ток:**

1. Авио-компанија пише захтев за резервацију аеродрома у ком наводи тачан датум кад жели да резервише писту.
2. Када је написала захтев, шаље га администратору аеродрома, који је задужен за резервације писте.
3. Администратор проверава доступност писте за тражени датум и проверава која времена су доступна тог датума.
 - (а) Ако не постоји ниједно време у које је писта доступна тог дана:
 - Авио-компанија отказује резервацију.
 - Авио-компанија мења датум и то шаље администратору.
 - (б) Ако постоји барем једно време у које је писта слободна, администратор шаље авио-компанији цену и време могуће резервације писте.
4. Авио-компанија разматра да ли цена и време одговарају или не.
 - (а) Ако не одговарају, авио-компанија разматра да ли жели да откаже резервацију или не.
 - Авио-компанија отказује резервацију.
 - Авио-компанија мења датум и то шаље администратору.
 - (б) Ако цена и време одговарају, администратор врши резервацију писте.
5. Комуникације између администратора и авио-компаније је завршена.

- **Алтернативни токови:**

- A1. **Унос неисправног датума.** Уколико у кораку 2 основног тока, администратор утврди да је послат погрешан датум, авио-компанија о томе бива обавештена и процес се наставља од корака 1 основног тока.



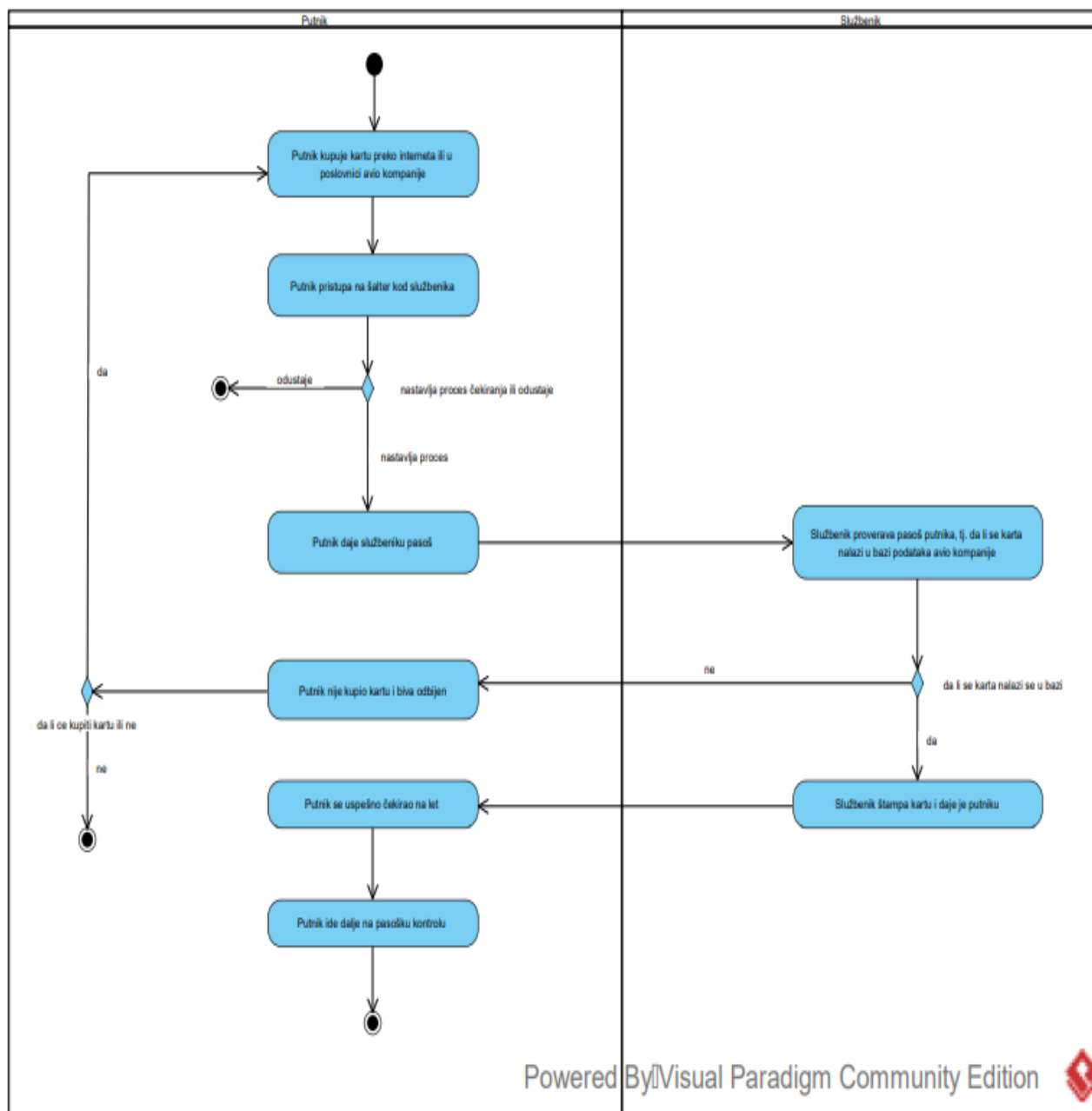
Слика 5: Дијаграм активности - Резервација аеродрома (писте)

3.3 Праћење летова

3.4 Обрада захтева

3.4.1 Чекирање путника

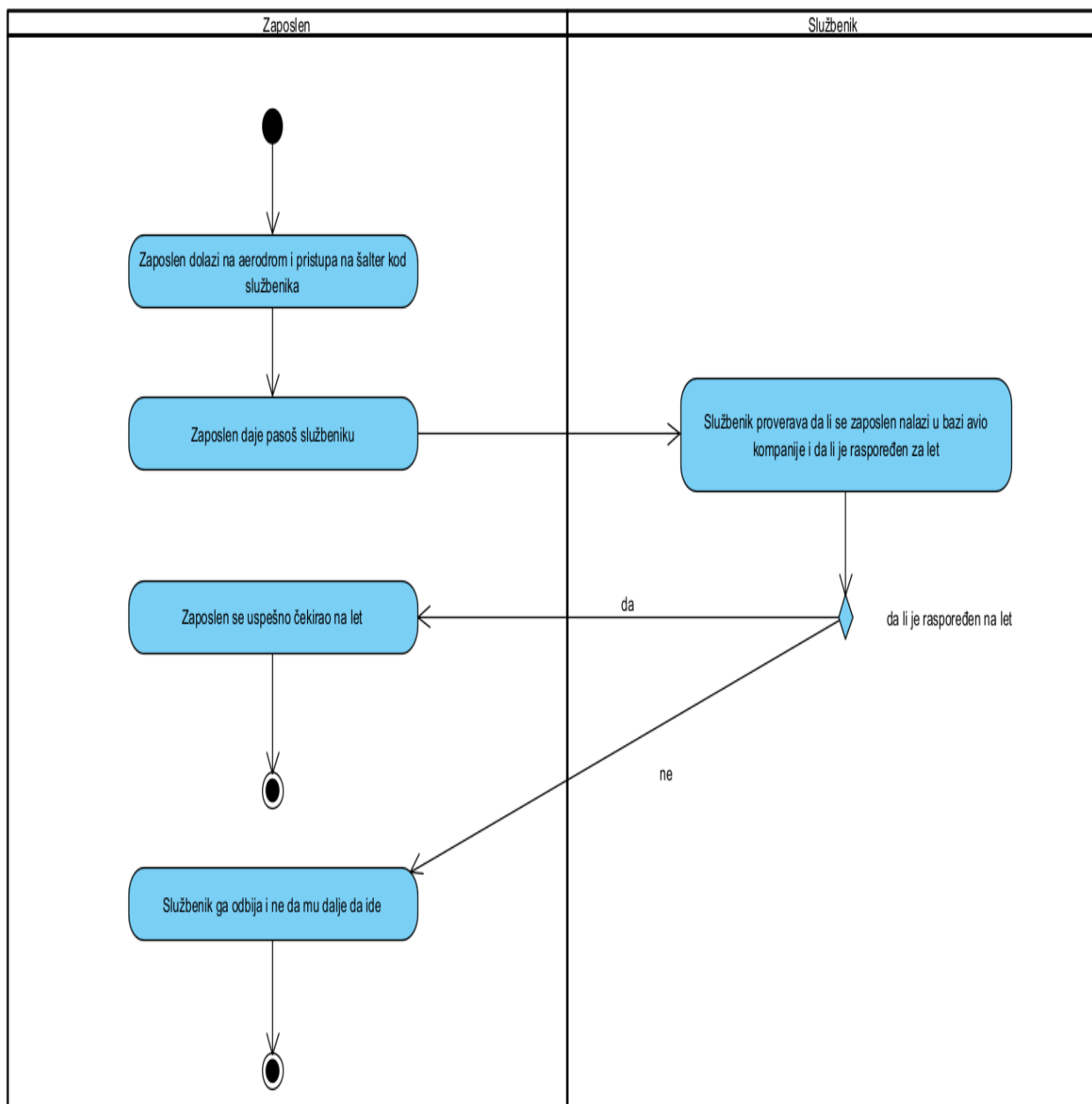
- **Кратак опис:** Путник долази на шалтер аеродрома са купљеном картом, пасошем и опционо пртљагом. Службеник проверава исправност наведених докумената, и при успешној радњи, корисник добија карту за лет и иде даље на пасошку контролу.
- **Учесници:**
 - Путник
 - Службеник
- **Предуслови:** Путник је купио карту (преко интернета или директно у пословници) и са пасошем и опционо пртљагом долази на аеродром.
- **Постуслови:** Путник се успешно чекирао на лет, добио карту и може да настави даље ка пасошкој контроли.
- **Основни ток:**
 1. Путник на интернету или у пословници купује карту за одређени лет.
 2. Путник долази на аеродром и приступа на шалтер код службеника.
 3. Путник даје службенику пасош и уколико има пртљак оставља га на траци.
 4. Службеник проверава пасош путника и на тај начин проверава и да ли се карта налази у бази података авио компаније, тј. да ли путник има купљену карту.
 - (а) Ако се карта налази у бази података, службеник је штампа и даје је путнику.
 - (б) Ако се карта не налази у бази, то значи да је путник није ни купио и он бива одбијен.
 5. Ако је све исправно прошло, путник добија карту што значи да се успешно чекирао на конкретан лет.
 6. Путник наставља даље на пасошку контролу.
- **Алтернативни токови:**
 - A1. **Путник одустао од лета:** Ако у кораку 2. путник из неких својих личних разлога одустане од лета основни ток се аутоматски ту завршава.



Слика 6: Дијаграм случаја употребе - Чекирање путника на лет

3.4.2 Чекирање запослених

- **Кратак опис:** Запослен долази на шалтер аеродрома са пасошом и опционим пртљагом. Службеник проверава исправност наведених докумената, и при успешној радњи, запослен иде даље на пасошку контролу.
- **Учесници:**
 - *Запослен*
 - *Службеник*
- **Предуслови:** Запослен је са пасошом и опционо пртљагом дошао на аеродром.
- **Постуслови:** Запослен се успешно чекирао на конкретан лет и може да настави даље ка пасошкој контроли.
- **Основни ток:**
 1. Запослен долази на аеродром и приступа на посебан шалтер код службеника.
 2. Запослен даје службенику свој пасош.
 3. Службеник, на основу пасоша, проверава да ли се запослен налази у бази података авио компаније и да ли је распоређен на конкретан лет.
 - (а) Ако је распоређен, службеник одобрава запосленом да крене даље.
 - (б) Ако није распоређен, службеник одбија запосленог и не пропушта га даље.
 4. Ако је све исправно прошло, запослен се успешно чекирао на конкретан лет.
 5. Запослен наставља даље ка пасошкој контроли.



Слика 7: Дијаграм случаја употребе - Чекирање запослених на лет

3.5 Одржавање авиона

3.5.1 Пријава квара

3.5.2 Поправка квара

3.5.3 Уношење потребних делова у систем

4 База података

5 Софтверска архитектура

6 Кориснички интерфејс

Литература

- [1] BBC news na srpskom. *Srbija i avio-saobraćaj: „Godina istorijskog rekorda“, zašto kasne letovi sa beogradskog aerodroma’*, 18. avgust 2023.
- [2] *Visual Paradigm*.