

## 4.1. Záznamník terénnych dát

### Kľúčové slová

záznamník dát, súbor, CSV formát, formát času, web, chyba, ergonómia aplikácie, odchytyvanie chýb,

### Čo sa naučíme a čo si precvičíme

- Analyzovať požiadavky na funkcionality a ovládanie aplikácie pre konkrétne potreby.
- Testovať aplikácie na chyby a ošetriť chybové stavy.
- Vytvárať štruktúrované dáta vo formáte CSV.
- Zvoliť vhodnú dátovú reprezentáciu pre záznamy obsahujúce viac položiek.
- Zaznamenávať dáta do lokálneho aj do vzdialeného súboru.
- Používať geolokačný senzor, čas.

### Záznamník hustoty dopravy

Výhody mobilných zariadení už poznáme. Sú ľahko prenositeľné, nezávislé na externom zdroji energie, využívajú bezdrôtové pripojenie do siete, majú zabudované množstvo senzorov a pod. Nečudo, že vďaka týmto vlastnostiam upútali pozornosť aj terénnych výskumníkov a pozorovateľov.

Správa ciest pravidelne monitoruje hustotu dopravy. Vďaka tomu vie lepšie plánovať výstavbu nových ciest, predpovedať ich záťaž a v konečnom dôsledku znižovať ekologickú záťaž životného prostredia. Ich pozorovatelia stoja na vybraných úsekoch ciest a zaznamenávajú dopravné prostriedky, ktoré po ceste prechádzajú. Na záznam využívajú papierové záznamové hárky. Tento spôsob nie je veľmi efektívny, navyše je prácne spojiť záznamy od viacerých pozorovateľov. Oslovili nás, či by sme im vedeli navrhnúť a vyvinúť efektívnejší systém pre záznam dopravy. Pozorovateľ v záznamníku uvádza miesto pozorovania, čas záznamu, typ dopravného prostriedku a jeho smer.

### Analyzujeme zadaný problém

#### Otázky na zamyslenie

Aké dáta potrebujeme zaznamenávať?

Ktoré z týchto dát vieme získať automaticky a ktoré sú na rozhodnutí pozorovateľa?

V akom formáte budeme dáta zaznamenávať?

Kde budeme dáta zaznamenávať (ukladať)?

Akú funkcionality by mala výsledná aplikácia poskytovať?

Čo z predchádzajúceho vieme v prostredí AI2 implementovať a čo nie?

### Vysvetlíme si

CSV formát (1) (Comma-separated values) je jednoduchý súborový formát. Dáta sú zaznamenané vo formáte textu, pričom každý riadok súboru predstavuje jeden záznam. Položky záznamu v riadku sú oddelené čiarkou. Výhodou formátu CSV je, že je jednoduchý a vedia s ním pracovať rôzne aplikácie (napr. tabuľkový kalkulátor). Každý riadok by mal obsahovať rovnaký počet položiek. Ak by niektorá položka záznamu mala obsahovať čiarku, je potrebné celú položku uzavrieť do úvodzoviek.

Ukážka súboru `vyplaty.csv`, ktorý obsahuje zoznam ľudí a výšku ich platu:

```
Jožko,Mrkvička,"580,32"
Karol,Petrík,702
Danka,Šikovná,"987,65"
```

### Vysvetlíme si

Pri zaznamenávaní časovej značky by sme mali uvažovať, v akom formáte čas zaznamenať. Ak predpokladáme, že zaznamenané dáta o doprave budeme spracovávať v tabuľkovom tabulátore, mali by sme zvoliť formát, ktorý bude tabuľkový kalkulátor interpretovať ako čas, resp. ako dátum a čas. Ak preskúame, aké formáty času podporuje tabuľkový kalkulátor, nájdeme medzi nimi aj formát: `30. 8. 2018 7:39:51`. V tomto formáte by sme mohli zaznamenávať čas aj my.

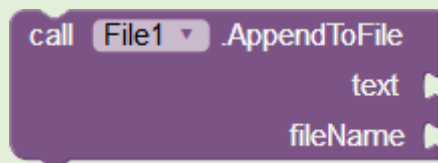
Pre prácu s časom použijeme komponent `Clock` (nájdeme ho v skupine `Sensors`) pomocou ktorého vieme pristupovať k systémovému času zariadenia (pozri: <http://ai2.appinventor.mit.edu/reference/components/sensors.html#Clock>). Súčasťou komponentu `Clock` je aj metóda pre formátovanie času (`FormatDateTime()`) ktorá upraví aktuálny čas podľa zadaného formátovacieho reťazca, resp. znakov tohto reťazca. (napr. hodina, mesiac apod.) Zoznam formátovacích znakov pre čas nájdeme na <https://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html>. Pre naše potreby môžeme čas formátovať nasledovne:



Výsledkom metódy `FormatDateTime()` je reťazec reprezentujúci čas v zadanom formáte. Výsledný reťazec použijeme ako súčasť záznamu.

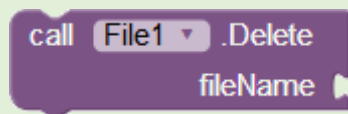
### Vysvetlíme si

Pre prácu so súborom použijeme komponent `File` (nájdeme ho v skupine `Storage`). Komponent `File` ponúka niekoľko užitočných metód:



Pripojí text k obsahu súboru. Ak súbor neexistuje, vytvorí ho. Ak potrebujeme do súboru zapísať znak konca riadku, použijeme znak `\n`. Nasledujúci text bude pokračovať v ďalšom riadku.

Ak názov súboru začína prefixom lomka `/`, pokúsi sa systém lokalizovať súbor na SD karte. Ak prefix `/` vynecháme, systém lokalizuje súbor súkromnom priestore našej aplikácie.



Zmaže súbor.

### Úloha 1

Vytvorte aplikáciu pre zaznamenávanie hustoty dopravy v dvoch smeroch. Aplikácia by mala umožniť zaznamenávať všetky dáta tak, ako to robia pozorovatelia do papierového záznamníka.

*Pomôcka: Premyslite si, ktoré hodnoty viete získať automaticky a ktoré sú na rozhodnutí pozorovateľa.*

*Pomôcka: Nezabudnite, že geolokačný senzor je potrebné po spustení aplikácie zapnúť.*

### Úloha 2

Pozorovateľ môže spraviť chybu a stlačiť nesprávne tlačidlo. Takýto záznam by zrejme znehodnotil prieskum hustoty dopravy.

Navrhните a implementujte spôsob, ako postupovať v prípade chybného zápisu (chyby pozorovateľa).

*Pomôcka: Prediskutujte vzájomne rôzne riešenia a ich efektívnosť.*

### Úloha 3

Pozorovateľ pri zaznamenávaní musí sledovať dopravu a zároveň pracovať s aplikáciou. Niekedy si nie je istý tým, či tlačidlo stlačil alebo nie. Navrhните a implementujte spôsob ako poskytnúť pozorovateľovi možnosť overiť si, či a aké tlačidlo stlačil.

### Vysvetlíme si

Pri aplikáciách ovládaných dotykom na obrazovke je problém v tom, že primárne nemáme spätnú väzbu po stlačení tlačidla (resp. po dotyku na obrazovke). Riešením je spustiť nejakú relevantnú reakciu, napr. zmenu obrazovky, zobrazenie textu alebo zavibrovanie zariadenia.

**call** **Sound1** **.Vibrate**  
**millisecs** Spustí vibrovanie zariadenia na zadaný počet milisekúnd.

#### Úloha 4

Pri zaškoľovaní pozorovateľov sa do súboru uloží množstvo testovacích záznamov. Zrejme by sa dali ignorovať použitím tlačidla `chyba`, ale nie je to veľmi praktické riešenie. Upravte aplikáciu tak, aby sa dali testovacie dáta v súbore zmazať.

#### Úloha 5

Nami navrhnutý záznamník pracuje spoľahlivo a pozorovateľom značne uľahčuje ich prácu. Správa ciest pri monitorovaní dopravy však využíva viacero pozorovateľov súčasne. Každý zaznamenáva hustotu dopravy na inom mieste. Pre výsledné spracovanie by bolo potrebné tieto záznamy spojiť v jednom, centrálnom bode.

Správa ciest má vytvorenú webovú stránku, ktorá čaká na dáta od pozorovateľov a tie potom uloží do súboru na serveri. V našej aplikácii potrebujeme vyriešiť to, ako dáta zo súboru `doprava.csv` prečítať a ako ich webovej stránke poslať.

Navrhnete a implementujete spôsob ako dáta zo súboru prečítať a poslať ich webovej stránke. Ak webová stránka dáta akceptuje, odošle späť odpoveď `ok`.

#### Vysvetlíme si

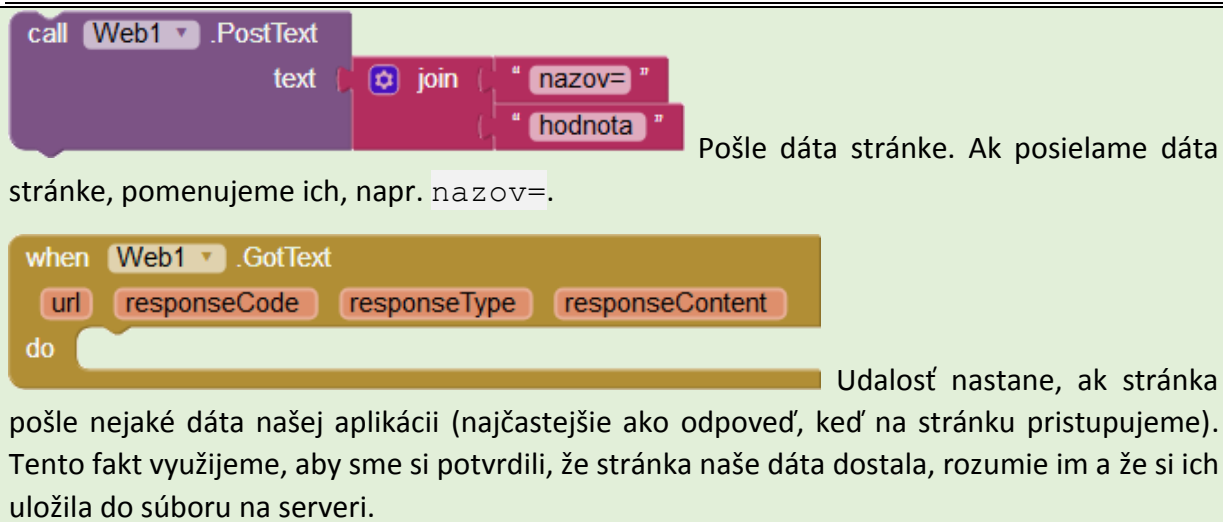
**call** **File1** **.ReadFrom**  
**fileName** Prečíta obsah súboru. Následne je vyvolaná udalosť `GotText()`.

**when** **File1** **.GotText**  
**text**  
**do** Udalosť nastane, ak prečítame obsah súboru. Obsah súboru je prístupný v premennej `text`.

#### Vysvetlíme si

Komponent `Web` (v skupine `Connectivity`) je určený pre komunikáciu s webovou stránkou. Komponent `Web` umožňuje posilať stránke dáta a rovnako od stránky dáta prijímať.

**set** **Web1** **.Url** **to** Nastaví adresu stránky, s ktorou chceme komunikovať.



call Web1 .PostText  
text join "nazov=" "hodnota"

Pošle dáta stránky. Ak posielame dáta stránky, pomenujeme ich, napr. nazov=.

when Web1 .GotText  
url responseCode responseType responseContent  
do

Udalosť nastane, ak stránka pošle nejaké dáta našej aplikácii (najčastejšie ako odpoveď, keď na stránku pristupujeme). Tento fakt využijeme, aby sme si potvrdili, že stránka naše dáta dostala, rozumie im a že si ich uložila do súboru na serveri.

### Čo by sme mohli viac preskúmať? Ako vylepšiť či rozšíriť záznamník dopravy?

Nasledujúce úlohy predstavujú možné vylepšenia aplikácie. Niektoré zvyšujú komfort používania aplikácie, iné riešia problémové situácie. Je možné, že niektoré z problémov ste postrehli pri predchádzajúcich úlohách a už ich vyriešili.

#### Úloha 6

Otestujte vami naprogramovanú aplikáciu v rôznych situáciách. Použite ju spôsobom, ktorý ste nepredpokladali pri jej vývoji. Správa sa aplikácia vždy korektne alebo ste ju dokázali dostať do chybného stavu? Vie aplikácia v každej situácii vykonať vami požadovaný úkon?

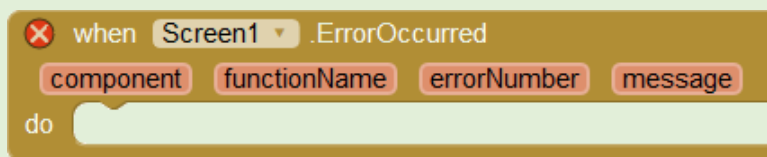
#### Úloha 7

Ak lokálny súbor neexistuje (pri prvom spustení aplikácie, po zmazaní testovacích dát), pri pokuse o odoslanie dát na server nastane chyba. Navrhните spôsob, ako tejto chybe predísť.

### Vysvetlíme si

Pri práci s aplikáciou môžu nastať chyby, ktoré je vopred ťažké predpovedať alebo im zabrániť. Výsledkom je, že aplikácia zobrazí nejakú chybovú, pre pozorovateľa nejasnú správu. Takéto správanie nie je žiadúce. Dobré naprogramovaná aplikácia by takéto chyby nemala zobrazovať. Používateľa by mala upozorniť vhodne zvoleným spôsobom a presne popísať, aká chyba nastala.

App inventor obsahuje jednoduchý mechanizmus pre odchyťovanie chýb. Pomocou metódy `ErrorOccured()` komponentu `Screen` vieme odchytiť a zareagovať na chybný stav.



when Screen1 .ErrorOccured  
component functionName errorNumber message  
do


#### Úloha 8

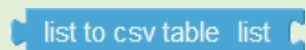
Pri odosielaní dát na server odosielame aj chybné dáta (ak pozorovateľ urobil chybný záznam). Výhodnejšie by bolo uložené dáta analyzovať a chybné záznamy vynechať. Navrhните

a implementujte spôsob ako z lokálnych dát vynechať chybné záznamy a rovnako aj záznamy informujúce o chybe.

### Vysvetlíme si

Štruktúrované dáta môžeme reprezentovať rôznymi spôsobmi. O formáte CSV sme hovorili v predchádzajúcej časti. Tento formát sa hodí skôr pre súbory. Pri práci s dátami CSV je vhodné ich transformovať do zoznamu – každý záznam (riadok) z CSV súboru bude samostatným prvkom zoznamu.

 `list from csv table text` CSV text skonvertuje do zoznamu. Jednotlivé záznamy (riadky) textu sú transformované do samostatných položiek zoznamu.

 `list to csv table list` Skonvertuje zoznam do textu v CSV formáte. Jednotlivé položky zoznamu sú transformované do samostatných riadkov v texte.

### Úloha 9

Pozorovateľ mohol spraviť aj ďalší typ chyby. Tlačidlo pre zmazanie chybného záznamu stlačil viackrát než ako bol počet záznamov pred tým. Pri pokuse o odstránenie chybných záznamov (pred odoslaním na server) sa pokúsime odstrániť záznam z prázdneho zoznamu. To samozrejme spôsobí chybu. Navrhните a implementujte spôsob ako tejto chybe predísť.

### Úloha 10

Pri úprave záznamov sa môže stať, že výsledkom je prázdny zoznam. Súbor záznamov síce obsahoval nejaké záznamy, ale obsahoval aj informáciu o tom, že tieto záznamy sú chybné. Výsledkom je, že na server posielame prázdny reťazec (žiadne záznamy). Navrhните spôsob ako tomuto zabrániť a ako informovať pozorovateľa, že nie sú žiadne záznamy pre odoslanie na server.

### Čo sme sa naučili

- analyzovať požiadavky používateľov a implementovať ich do návrhu aplikácie,
- testovať aplikáciu s cieľom detegovať chybné stavy a ošetriť tieto chybné stavy,
- popísať dáta v štruktúrovanom formáte CSV,
- ukladať a čítať dáta zo súboru,
- konvertovať dáta z formátu CSV do zoznamu a naopak,
- komunikovať z webovou stránkou.

### Sebahodnotiaca karta

	s výdatnou pomocou učiteľa	s pomocou učiteľa	samostatne
Viem vytvoriť program, ktorý zapisuje do súboru			
Viem vytvoriť program, ktorý zapisuje dáta do súboru v CSV formáte			
Viem vytvoriť program, ktorý zistí systémový čas			
Viem vytvoriť program, ktorý formátuje systémový čas			

Viem vytvoriť procedúru s parametrami			
Viem ako zistiť, že nastala chyba pri vykonávaní programu			
Viem vytvoriť program, ktorý pošle dáta webovej stránke			
Viem konvertovať dáta z CSV formátu do zoznamu a naopak			

## Bibliografia

1. **prispievatelia Wikipédie**. Comma-separated values. *Wikipédia, Slobodná encyklopédia*. [Online] 2017. [https://sk.wikipedia.org/wiki/Comma-separated\\_values](https://sk.wikipedia.org/wiki/Comma-separated_values).
2. **Refsnes Data**. PHP 5 Tutorial. *W3Schools Online Web Tutorials*. [Online] <https://www.w3schools.com/php/default.asp>.

## Register pojmov

### C

Clock  
FormatDateTime, 2

### F

File, 3  
AppendToFile, 3  
Delete, 3  
GotText, 4  
ReadFrom, 4  
formát  
csv, 2

### C

chyba  
odchyťovanie. *Pozri* Screen:ErrorOccurred

### L

list  
list from csv table, 6  
list to csv table, 6

### S

Screen  
ErrorOccurred, 5  
Sound  
Vibrate, 4

### W

Web, 4  
GotText, 5  
PostText, 5  
Url, 4