# Aplikácia pre správu ubytovania

## Okruhy prezentácie

1. **Aktéri, funkčné celky**  **a doplnkové funkcionality**
   * ***Aktéri systému:***
     + *Administrátor* - správa číselníkov, správa rolí (oprávnení), správa používateľov, správa harmonogramov, importy a exporty dát, správa dokumentov
     + *Pracovník* - správa vybraných číselníkov podľa kompetencií, správa a evidencia špecifickej agendy podľa kompetencií pracovníka (žiadosti o ubytovanie, ubytovanie, rezervácie), správa služieb (plánovanie, harmonogramy, opravy, kontroly, inventár, revízie), poplatky, importy dát, exporty dát, tvorba a vkladanie dokumentov
     + *Klient* - študent, študent externista, hosť (žiadosť o ubytovanie, rezervácia ubytovania, hlásenia porúch, úhrada platieb,...)
   * ***Dostupné rozhrania:***
     + *Spoločná aplikácia pre aktérov Administrátor a Pracovník* (prístup k jednotlivým funkcionalitám riadený cez oprávnenia / rolí) v podobe webovej aplikácie. Návrh obrazoviek, viď. mokups
     + *Samostatná aplikácia pre klienta* vo forme webovej aj mobilnej aplikácie (podpora pre IOS a Android)
   * **Výmena dát a ostatné služby**
     + Webové služby / API rozhrania pre účely výmeny dát (komunikácia s akademickými informačnými systémami, ekonomickými IS, riadenie prístupu do budov, LDAP,...)
     + Integrácia s platobnými bránami
     + Notifikačné servisy (interná systémová notifikácia, integrácia na SMS bránu, e-mailové notifikácie, PUSH)
     + Ručné importy a exporty dát z prehľadových obrazoviek
     + Správa dokumentov
2. **Technické riešenia**
   * **Architektúra systému**

Navrhovaná architektúra bude viacvrstvová, rozdelená do nasledovných hlavných vrstiev:

* + - Dátové úložisko - uchovávanie dát
    - Backend – komunikácia s dátovým úložiskom, biznis logika
    - REST API – komunikačná vrstva
    - Frontend – UI a prezentačná logika
    - Bezpečnostná vrstva – zabezpečenie aplikácie a prístupu k údajom
  + **Použité technológie**

Dôraz kladieme na Open Source technológie s cieľom minimalizovať licenčné náklady a maximalizovať flexibilitu. Hlavným vývojovým jazykom je Java.

* + - Databáza
      1. PostgreSQL – výkonný a spoľahlivý open-source databázový systém, preferovaný ako hlavné riešenie
      2. Oracle Database – možnosť použitia ako alternatíva v prostredí s existujúcou licenciou alebo infraštruktúrou
    - Perzistovanie údajov
      1. JPA (Java Persistence API) – štandardné API na prácu s databázou cez objekty
      2. Hibernate 6 – robustná implementácia JPA, zaisťuje efektívne mapovanie objektov na databázové tabuľky, poskytuje nástroje pre CRUD operácie
    - Backend a REST API
      1. Spring Boot – moderný framework pre vývoj Java backendu, umožňuje rýchly vývoj REST API s dôrazom na škálovateľnosť
      2. Swagger – automatizovaná dokumentácia REST API pre vývojárov aj testerov
    - Bezpečnosť
      1. Spring Security + JWT – zabezpečenie prístupov pomocou tokenov, role-based autorizácia a ochrana pred bežnými útokmi
      2. HTTPS – zabezpečená komunikácia medzi klientom a serverom šifrovaným protokolom
    - Frontend
      1. Nx (Monorepo systém) – Možnosť vyvíjať oddelene frontend pre študentov a správcov, ale zároveň zdieľať spoločné komponenty
      2. Redux Toolkit – zabezpečuje správu aplikačného stavu na strane klienta
      3. RTK Query - API komunikácia s backendom na webe aj v mobilnej aplikácii – rovnaké endpoints
      4. React – Hlavný framework pre webové rozhranie. Umožňuje tvorbu komponentového, moderného a responzívneho UI
      5. Tailwind CSS – Utility-first CSS framework pre rýchly a konzistentný dizajn
      6. I18next – Lokalizácia aplikácie
      7. Jest + React Testing Library – Jednotkové testy komponentov, logiky a UI
      8. React Native + Expo – Hlavný framework pre natívne mobilné aplikácie (Android + iOS). Zdieľanie logiky s webom.
  + **Zabezpečenie prevádzky, komunikácie**

Každá komunikácia prostredníctvom REST API prebieha využitím HTTPS protokolu. Autentifikácia používateľov cez interný prihlasovací modul. Možnosť externého modulu cez správu identít – IDM, LDAP a pod. Autorizácia operácii na základe oprávnení pridelených užívateľovi v systéme

* + **Klient ako Multiplatformová aplikácia**

Aplikácia pre klienta je vyvíjaná s dôrazom na multiplatformovú kompatibilitu a responzívny dizajn. Klient tak získa prístup do svojej časti aplikácie ako cez web, tak aj cez mobilné zariadenie Android a iOS

1. **Obsah dodávky** – možnosti dodávky systému:
   * Hotový sw + licencia na používanie + používateľská dokumentácia
   * Hotový sw + licencia + zdrojový kód + používateľská dokumentácia + dokumentácia k zdrojovému kódu
2. **Možné rozšírenia**
   * **Implementácia asistenta s využitím AI**  - integrovaný helpdesk pre používateľov
   * **Elektronické podpisovanie** – integrácia na sw pre el. podpisovanie (ubytovanice zmluvy a pod.)
3. **Otázky**
   * **Očakávané výstupy analýzy** **a návrhu aplikácie** – model tried, model procesov, usecases, testcases
   * **Očakávané výstupy testov** – štandardné funkčné testy podľa testcases, performantné testy, aká je očakávaná účasť a postup na akceptačných testoch
   * **Zavádzanie systému a Podpora**  - aká je predstava o priebehu školení používateľov, aký je očakávaný rozsah následnej podpory? Bežne poskytujeme podporu cez tiketový systém Jira, telefón a mejl v pracovných dňoch
   * **Zabezpečenie záloh nepretržitej prevádzky** – zabezpečenie náhradnej prevádzky v prípade výpadkov hlavnej prevádzky
   * **Zabezpečenie prevádzky s inštaláciou aplikácie pre správu ubytovania** – očakávané a navrhované spôsoby a prostriedky, zálohovanie nastavení a dát
   * **Lokalizácia prostredia systému** – aké jazyky a predstava o tom, kto zabezpečí podklady
   * **Tlač dokumentov** – formáty výstupov a správa tlačových šablón
   * **Financovanie** – predstava financovania pred a po odovzdaní produktu v rámci podpory, prípadne financovanie dodávky SW a HW podľa špecifických požiadaviek (napr. ak bude požadovaný komerčný DB server, Oracle, SMS brána, platobná brána a pod.)